Note: This cover sheet is not to exceed one page.

ALFRED P. SLOAN FOUNDATION
sloan.org | proposal guidelines

PROPOSAL COVER SHEET

PROJECT INFORMATION

Principal Investigator

Daniel S. Katz
Chief Scientist, NCSA
Research Associate Professor, CS, ECE, iSchool
1205 W Clark St, Urbana, IL 61820
217-244-8000
dskatz@illinois.edu

Grantee Organization: U. of Illinois Urbana-Champaign
Amount Requested: $350,000
Requested Start Date: 1 September 2022
Requested End Date: 31 August 2024
Project URL: TBD

PROJECT GOAL

The work to be done in this policy project will address the question of "How can policy be used to increase the sustainability and impact of research software in the scholarly research community?"

OBJECTIVES

The project has three objectives: 1) making positive policy changes, 2) understanding why such changes succeed and fail in this context, and 3) planning for future policy work based on these lessons.

PROPOSED ACTIVITIES

This project will hire a postdoc to work with the PI to accomplish the project tasks over two years:

1. Assemble and maintain a list of potential policy activities (with elements of both policy research and policy advocacy) and the effort required for each
2. Accomplish some number (9 are initially planned) of these policy activities.
3. Document the lessons from these policy activities and using this to plan and propose an expanded and improved research policy center

EXPECTED PRODUCTS

1) List of potential policy activities; 2) documentation about the lessons from accomplished policy activities and plan for an expanded and improved research policy center; 3) documented existing career paths for individuals creating research software; 4) communications list for those interested in career paths; 5) documents on starting and running URSSI chapters, and 3 such chapters; 6) documented examples of successful use of individual contributions to public goods to gain academic promotion; 7) documented examples of reviewers prioritizing grant proposals that reuse and contribute back to maintenance of public infrastructure; 8) documented cases of discovery where software was particularly fundamental; 9) tweets, blog posts and editorials based on these cases; 10) public living summary of funding opportunities for software maintenance; 11) updated DISCOVER event cookbook

EXPECTED OUTCOMES

This project will make progress towards: 1) Establishing career paths (including titles and evaluation criteria for hiring and promotion); 2) Improving the measurement of impact of individuals, especially in activities that are inherently collaborative like software development; 3) Incentivizing contributions to public goods / infrastructure within the academic credit model; 4) Better recognizing the value and importance of software; 5) Improving funding opportunities and stable funding for maintenance of software that is important but doesn't have a generic market and/or isn't considered novel in the eyes of the average funder/reviewer; 6) Increasing the diversity of the development and maintenance community to achieve the diversity of the overall US population.

II Proposal Body

1. What is the research question and why is it important?

Software pervades all parts of modern scientific research, including data analysis and inference as well as computational science. One would be hard pressed to find an area of research that is not impacted by software. Recent surveys in the US and UK show that 90-95% of researchers rely on research software, and 63-70% of them cannot continue their work if this software was to stop functioning (Hettrick et al. 2014, Nangia & Katz 2017a). Much of this software is developed by researchers for researchers, as the contemporary scientific process demands the development of new methods in tandem with the demands of new discoveries and fields. However, despite its importance, a large proportion of research software is developed in an ad hoc manner, with little regard for the high standards that are characteristic of other research activities. As a result, the research software ecosystem is fragile and the source of numerous problems that plague modern computational science (Carver et al. 2018).

Researchers today are under intense pressure to demonstrate expertise in their chosen domains while also trying to maintain a working current knowledge of digital skills such as software engineering. This combination is unsustainable for most researchers. With little bandwidth to keep up with best practices or sufficient recognition of software development as a scholarly activity, much research software is developed in a manner that makes it wholly unsustainable, despite the obvious role that it plays in modern research, for a multitude of reasons. Academic promotion and tenure, even in institutions with liberal policies, consider peer-reviewed publications to be the primary metric of progress in most disciplines. Even when the impact of software is made clear, it is usually not considered a traditional scholarly activity, making it very challenging to get credit (OECD 2020). There is no shortage of horror stories of academics who have built demonstrably impactful software, only to be denied tenure. Even outside the tenure track, only a few academic jobs offer meaningful career progression for

software work. A second reason, strongly correlated with the lack of recognition of software, is the lack of training opportunities. Many research software engineers are self taught. Others learn programming from bootcamps and workshops rather than in traditional academic coursework. Overworked academics are unable to take advantage of such opportunities and therefore develop software using outdated practices. Lastly, even when software is recognized as having impact, funding agencies rarely fund maintenance and ongoing development of such work, leading to reinvention rather than reuse (https://chanzuckerberg.com/rfa/essential-open-source-software-for-science/).

The US Research Software Sustainability Institute (URSSI) team (Carver et al. 2018) was initially funded by NSF in 2017 to engage in a series of conceptualization activities designed to gain a deeper understanding of why research software is so unsustainable and what can be done about it. Through numerous discussions with diverse groups of researchers, we brainstormed challenges and solutions that are highly scalable and impact a large swath of researchers.

The URSSI team is now in the process of creating an openly-governed community organization to move forward on a set of initial activities and to support future activities, aimed at increasing the sustainability of research software as created, maintained, and used by US institutions and researchers. This includes three projects that are being submitted to the Sloan Foundation:

1. this one on policy, including the guidelines, rules, and practices related to research software created by institutions such as universities, laboratories and industry where research software is created, maintained, and used; those created by government and private funders who support the development and maintenance of research software; those created by publishers and others in the publishing ecosystem who support the publishing, sharing, indexing, and crediting of research software; and those created by

scholarly and professional societies (e.g. for astronomers, physicists, computer scientists, linguists, sociologists, research software engineers) that create community norms among the people who work with research software,

2. one led by Karthik Ram on community development, and
3. one led by Kyle Niemeyer on education,

as well as one project that was formally submitted to NSF after initial positive discussion with NSF program officers, led by Nic Weber on incubating research software projects.

Our vision is that the open URSSI organization will be the primary place where these projects, plus others to come in the future, communicate and coordinate, and that the URSSI governance body (initially the five NSF-funded URSSI conceptualization PIs—Carver, Gesing, Katz, Ram, Weber—plus Niemeyer, with a structure to be developed to change this over time based on community inputs and new projects) will act as advisors for URSSI-aligned projects, including this one.

During the period of this set of projects, the URSSI governance body will consider sustainability of URSSI as a core activity (community, community manager, governance, outreach) as well as projects that could potentially be separately funded (policy, diversity, education, incubator, etc.) with the goal of creating a sustainability plan that could be enacted with additional short-term support.

The work to be done in this specific policy project will address the question of “How can policy be used to increase the sustainability and impact of research software in the scholarly research community?” It will have two components, 1) direct policy work, aimed at both a) making positive policy changes and b) understanding why such changes succeed and fail in this context, and 2) planning for future policy work based on these lessons.

Note that the work aimed at making positive policy changes is neither lobbying nor formal advocacy, but rather gathering data about existing practices and policies, analyzing it, and then making the data and analyses available to stakeholders in the research software landscape.

The direct policy work is based on URSSI planning done over the last few years, as documented in URSSI's implementation plan (Ram et al. 2022), and specifically in Chapter 7 (Policy). The overall (long-term) goals for the URSSI policy work, which this project will begin, are:

1. In funding agencies, direct funding of software maintenance and other software sustainability activities is a core part of the mission, e.g., at NSF, this includes all program officers across all directorates. (Again, note that this project will not lobby or formally advocate for such policies with members of the federal government, but we expect that the data and analyses that we will disseminate in the scholarly community will be read by some program officers who will share it with their colleagues and will use it in their internal policymaking.)
2. In universities and academic fields, positions for people developing and maintaining research software are available, recognized, and rewarded.
3. In publishing, support and recognition for software as a core part of scholarly research is the norm. (This includes the recognition that software is as valuable to the research community as the results themselves, that processes exist to evaluate software in papers, expectations for authorship for software developers are clear.)
4. In industry, sharing best practices, coordinating efforts, and contributing to open source software projects is the norm.
5. Open source software is recognized as a key element of open science and reproducibility.
6. The community that builds and maintains research software is diverse and inclusive.

2. What is the state of the research on this question?

The primary work that uses policy to impact the research software world has been performed by the (UK) Software Sustainability Institute (SSI) (Crouch et al. 2013) over the last 13 years, with additional work happening in funding agencies and organizations, such as Sloan and Moore Foundations, the Chan-Zuckerberg Initiative, and government agencies including the European Commission, UKRI, NSF, and DOE. Policy has been a theme of the SSI from the start: “If we're to change the way that researchers deal with software, work is required […] on influencing the policy that motivates all of the stakeholders in the research software community.” (Crouch et al. 2013). Policy here includes understanding the sociotechnical context in which research software is created, maintained, and used, and attempting to change this context to promote desired outcomes (e.g., more sustained research software, less redevelopment, better software, better research), rather than attempting to make changes at the level of individuals or individual software projects.

This work is somewhat orthogonal to the larger discussion around science policy and funding (Sarewitz 1996, Pielke 2007), though it contains elements of Pielke’s Issue Advocate and Honest Broker of Policy Alternatives modes of science engagement. Regarding the role of software in research, it uses the Issue Advocate approach to attempt to influence policymakers to understand the importance of research software and to better support it. However, regarding how to do this most effectively, it uses the Honest Broker approach to discover and present multiple alternative methods that different stakeholders can choose between.

The SSI has used various elements of its programs to discover potential places where policy is impeding good research software outcomes. From the topics that are discovered, policy activities are planned that include 1) data gathering and analysis to understand the policy

challenge and how it affects research software work, and 2) campaigns to disseminate this information to influence stakeholders who can change that policy.

One successful example is the SSI work that has led to the Research Software Engineer (RSE) title and movement, and also to funded fellowships for RSEs from UK funding agencies EPSRC and then STFC. One initial activity in the SSI discovered that in 2012, people who today we would call RSEs were listed under 194 different titles and didn't have stable careers or good career paths. This recognition led to a lot of community building and follow-on advocacy, which in turn led to the RSE movement, with a professional society in the UK (Society of Research Software Engineering) with community of >3500 people, a US organization (US Research Software Engineer Association) with ~1000 members, and 7 other national/multinational RSE organizations globally. Alongside this RSE work, the SSI collected and analyzed data about the importance of software to EPSRC grants. (Hetrick 2014, Hettrick 2018). The SSI then used this data and the growing RSE movement to work with the EPSRC to create Research Software Engineering Fellowships (later also supported by STFC). This was done by identifying and working with the advisory groups and focus groups that the funders relied on, and getting the community to align their goals. This work has also led to a general acceptance that software is a vital part of the research and research infrastructure. The importance of software is raised in various policies and reports from UKRI (the organization above EPSRC, STFC, etc.) A tangible example of this is the recent software for research communities call (UKRI 2021), which specifically aimed at RSEs, and received over 200 expressions of interest in the initial phase.

A second example from the SSI is work on capturing and counting software as part of the UK's research assessment process (the Research Excellence Framework, or REF). The SSI worked with the UK government to open this process in the most recent cycle to allow software to be a type of output that researchers and their universities could put forward as a product on which they would be assessed. However, in practice, less than 0.1% of submitted outputs were

software. To address this, the SSI has created guidance for submitters and reviewers to encourage software, and also has run a campaign called the Hidden Ref to crowdsource submission of important work that was not in the form of a publication, and then to run a competition to award prizes to the best of these submissions (Hettrick 2021).

Another third example, which Katz has co-led, is work on software citation. In 2016, the FORCE11 Software Citation working group published a set of software citation principles (Smith et al. 2016) that justified the need for these principles based on data about existing software mentions in publications and desired recognition of software developers. Following this publication, the follow-on FORCE11 Software Citation Implementation working group has been creating communities focused around various classes of stakeholders and working to influence these stakeholders and their policies. These include publishers (Katz & Chue Hong et al. 2021), repositories that store software (Task Force on Best Practices for Software Registries 2020), and software development sites (GitHub, 2021.)

Overall, there is a long history of research and guidance on policy changes, with Alinsky (1971) as a collection of some of the material developed during the 1960s, though this of course was based on part in union and other organizing over the preceding century. Cerna (2013) summarizes a set of theories on policy change and discusses various types of policy implementation. This project will qualitatively and quantitively highlight problems with current policies related to research software, and use a combination of community building of those stakeholders affected by policy (software developers, software users, disciplinary communities) to influence the stakeholders who create and implement policy (funders, governments, publishers, hiring institutions) and competition between these policy creation stakeholders to create incremental change, using existing networks (e.g., Research Software Alliance's (ReSA) Funders Forum, International Association of Scientific, Technical and Medical Publishers (STM)). In particular, this project will seek areas in which nudges can be implemented at low

cost and with large affect (Thaler and Sunstein 2008). As previously mentioned, the project will use the honest broker approach to suggest and explain multiple of these possible changes, in the context of advocacy around the importance of research software.

An example of a successful nudge is the 2013 change by NSF allowing biosketch entries to contain “products” instead of “publications.” This change was intended to make clear to proposers, reviewers, and the research community, that “products may include, but are not limited to, publications, data sets, software, patents, and copyrights.” (NSF 2013) This has led to NSF proposers highlighting software that they have developed as part of the evidence of their ability to do new software-focused research.

3. Why is the proposer qualified to address the research question for which funds are being sought?

Katz is a global leader in research software sustainability (e.g., Katz & Proctor 2014, Katz 2018, Druskat & Katz 2018, Druskat et al. 2021, Katz & Carver et al. 2021, Katz & McHenry et al. 2021, Katz & Ramnath 2021, Ramnath & Katz 2021), leading or co-leading projects and community activities to highlight the role of research software, its developers and maintainers, and their incentives and career paths (e.g., Research Software Alliance (ReSA, https://www.researchsoft.org), US Research Software Engineer Association (US-RSE, https://us-rse.org), FORCE11 Software Citation Working Group (https://force11.org/group/software-citation-working-group/), FORCE11 Software Citation Implementation Working Group (https://force11.org/groups/software-citation-implementation-working-group/), RDA/ReSA/FORCE11 FAIR 4 Research Software (FAIR4RS) WG (https://www.rd-alliance.org/groups/fair-4-research-software-fair4rs-wg)), as well as presenting talks, papers, and blog posts on this topic. This is based on his 35 years of work in research software, in both applications (e.g. in electromagnetics, Katz 1988, Katz et al. 1991) and tools

(e.g., in workflow systems, Deelman et al. 2005, Wilde et al. 2011, Babuji et al. 2019), as well as his work as an NSF program officer responsible for research software as scientific infrastructure from 2012-2016, where he realized that simply funding software directly was neither scalable nor sufficient, and that a more effective way of addressing research challenges was to bring in a sociological view and to attempt to change practices through influencing the behavior of the research community (NSF 2014).

As a co-PI of the URSSI conceptualization project (Carver et al. 2018), Katz led the activities to plan the policy part of the URSSI Institute (Ram et al. 2022). These included a set of community workshops that iteratively developed ideas for this area and then budgeted this work, which is the source of the methodology and specific activities described in the next two sections.

Katz has also been involved in a number of previous data gathering and analysis activities, such as the 2017 and 2018 surveys of the international RSE community (Philippe et al. 2018, Philippe et al. 2019), a survey done of the US research software community for URSSI planning (summary of results in Section 2.3 of Ram et al. 2022, full results in an accepted PeerJ Computer Science paper, Carver et al. 2022), an analysis of public data archiving policies in academic publishing ecology journals (Sholler et al. 2019), and a study of software mentions in Nature (Nangia & Katz 2017b).

Finally, Katz has been involved in setting policy and analyzing existing programs in his work at NSF and as NCSA Chief Scientist, and as a reviewer (e.g., LBNL 2019) and advisor of other activities (e.g., Center for Computing in Engineering & Sciences (CCES) at University of Campinas, ReproNim: A Center for Reproducible Neuroimaging Computation, SBGrid Consortium.)

4. What is the research methodology?

This proposal aims to accomplish multiple tasks, which will be challenging given limited resources. Fortunately, the full URSSI plan's policy component was scoped at 3 FTEs of staff and about 0.25 FTE of leadership annually for five years, so this proposed project, with 1 FTE of staff and 0.08 FTE of leadership annually for two years has a good starting list of activities to draw from, which is the basis of the work proposed in this section. (See Appendix V.6 for details about the full URSSI implementation plan's policy work.)

The tasks (and fraction of work) this project will accomplish are

1. Assembling a list of potential policy activities (with elements of both policy research and policy advocacy) and the effort required for each (5%)
2. Accomplishing some number of these activities (80%)
3. Documenting the lessons from these activities and using this to plan and propose an expanded and improved research policy center (15%)

Policy activities will aim to achieve the following outcomes, though this initial project will only be able to make partial process in doing so. (Also, note that each activity will be planned to make at least some initial impact during this project, though in some cases, the majority of the impact will occur after this project ends.)

1. Establish career paths (including titles and evaluation criteria for hiring and promotion).
2. Improve the measurement of impact of individuals, especially in activities that are inherently collaborative like software development.
3. Incentivizes contributions to public goods / infrastructure within the academic credit model
4. Better recognize the value and importance of software.
5. Improve/increase funding opportunities and stable funding for maintenance of software that is important but doesn't have a generic market and/or isn't considered

novel in the eyes of the average funder/reviewer. Today “lumpy” project funding (projects that are competitively funded for fixed periods, often with gaps between funded project periods) means that maintenance/sustainability can’t be reliably folded into project costs.

6. Increase the diversity of the development and maintenance community to achieve the diversity of the overall US population.

All of these tasks will be done publicly and in the context of the URSSI community and its governance, as discussed in Section 1, meaning that the governance body members other than PI Katz (Carver, Gesing, Ram, Weber, and Niemeyer) will serve as advisors for this project (all three tasks), and the URSSI community will be invited to participate in Task 1, with the outputs of all tasks openly shared with the full URSSI community and the public. The initial URSSI community consists of participants in URSSI workshops and electronic communications during the NSF-funded conceptualization activity, and the Sloan-funded URSSI project being proposed by Ram will further develop and grow this community.

Because community changes require community input and community activity, and because this project is resource limited, it will seek to involve collaborators, including:

- Representatives from organizations that represent other research support roles (e.g., librarians, data stewards, RSEs), to work together to promote all such roles
- Representatives from open-source communities, particularly those that are already effective and working at scale
- People working in science policy, such as in AAAS and the National Academies (in particular, AAAS policy fellows might be interested in helping on time-constrained activities that match the purpose and period of their fellowship)

- Representatives from organizations and companies serving OSS and RSE communities (e.g. NumFOCUS, Code for Science & Society, GitHub[1])
- Representatives from organizations that focus on diversity and inclusion in academia, to encourage them to include software-focused roles where possible
- People from the European Commission regarding European Open Science Cloud (EOSC), etc.
- Representatives from organizations like Research Software Alliance (ReSA) Research Data Alliance (RDA), Academic Data Science Alliance (ADSA), FORCE11

5. What is the work plan?

Initially, this project will assemble a list of potential policy activities (with elements of both policy research and policy advocacy) and the effort required for each. This list will be public, and the initial work will be done with the advisory committee. Having the list be public will offer the opportunity for public inputs and public participation in tasks, in some cases without participation from this project. The set of potential activities will be seeded with the full list from the policy chapter (7) of the URSSI plan (Ram et al. 2022), which includes both activities that were deemed potentially feasible with 3 FTEs of effort over 5 years, as well as activities that were not feasible at that level and would require additional resources. Generating and maintaining this list will be an ongoing activity, at 5% effort over the project duration. The list is intended to be a community resource, to encourage others to define, propose, and carry out some of the activities, as part of the larger URSSI community.

In this project, the first specific policy activity that will be performed is thus

1. Generating and maintaining the public activity list

[1] Note that GitHub's policy unit is actively working on issues around research software impact and plans to build community activity around this area (Cihon 2022).

In addition, an activity that will be performed based on the other activities is

2. Documenting the lessons from these activities and using this to plan and propose an expanded and improved research policy center (4 months of work, plan to be delivered at end of project)

Other activities will be determined with the advisory committee (and the public). The starting point for this set is the following 9 activities:

Career paths, in collaboration with US-RSE and other RSE organizations:

3. Document existing (known successful/viable, known failures) career paths for individuals creating research software (2 months of work, deliver in month 6) (Note that this activity is described in more detail in Appendix V.4 - Empirical Research Methods)
4. Create and maintain a communications list (e.g., mail, slack) for those interested in career paths, with list to be supported by URSSI community manager) (low level of effort, deliver in month 3)
5. In collaboration with URSSI community manager and US-RSE, help start organizational “chapters” of research software developers (aka URSSI or US-RSE chapters?) at existing universities / societies / organizations; create materials, e.g., guidance on how to set up a chapter, how to align it to local activities, and how to run it. These chapters could talk about training, do consulting for problems, host hacky hours, study groups, software days, and come together in a larger event, perhaps a regional workshop or an URSSI-wide conference. Overall, this helps, grows, and establishes the community (and make connections that could help chapter members meet the right person for their next career moves). (2 months of work, deliver in month 15)

Incentivizing contributions to public software, in collaboration with ReSA, SSI, RSE organizations:

6. Gather examples of successful use of individual contributions to public goods/infrastructure to gain academic promotion (both case studies and language used in research statement and letters of recommendation), gather anecdotes about how contributing to projects has increase value within the traditional reward system (e.g., more collaborators and papers; increased opportunities to meet/work with new/old collaborators), share such examples via website and publicize via blogs and editorials, (6 months of work, deliver in month 18)
7. With ReSA, study funder policies to understand how and when reviewers prioritize grant proposals that reuse, build-upon and contribute back to maintenance of public research software, highlight and share good examples (3 months of work, deliver in month 21)

   (Note that this activity is described in more detail in Appendix V.4 - Empirical Research Methods)

Recognizing the value and importance of software:

8. Find science/discovery cases where software was particularly fundamental, particularly digging into software that is not so generally well-known (2 months of work, deliver in month 13)
9. Publicize science/discovery cases where software was particularly fundamental, focusing on demonstrating impact of software, via tweets, blog posts, editorials (2 weeks of work per opportunity, roughly every 6 months)

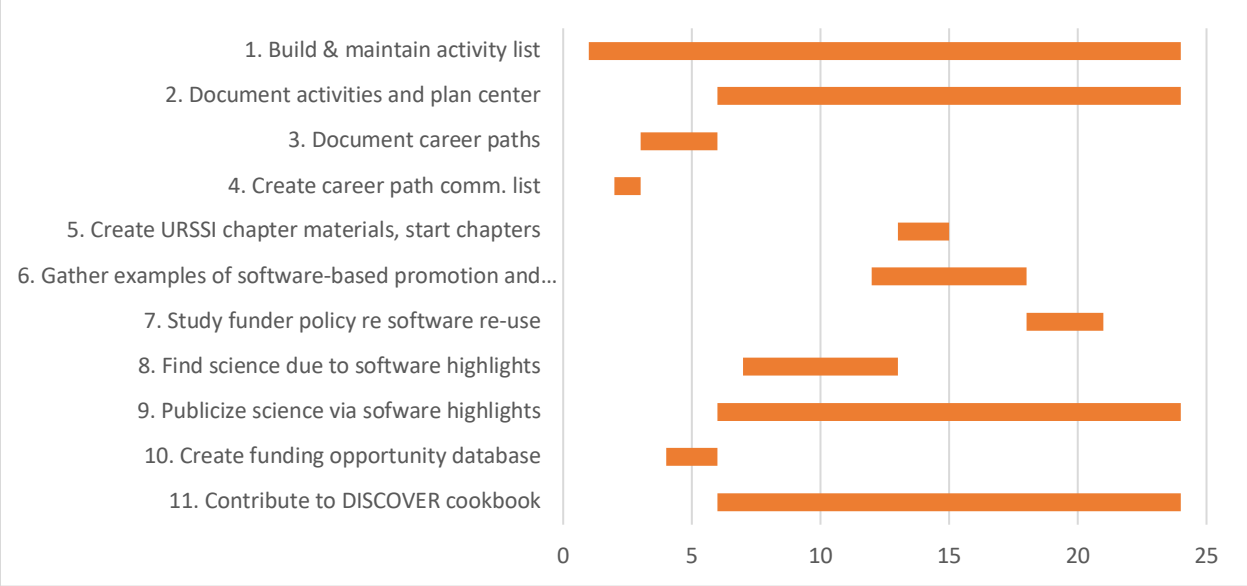
Funding opportunities for software maintenance, in collaboration with ReSA and others, including CZI EOSS participants, NumFOCUS, and CS&S:

10. Review the landscape of funding opportunities for software maintenance (and gather data about them) and provide a public summary, then keep the summary up to date. (1 month of initial work, deliver initial site in month 6, ongoing work to keep up to date)

Increase diversity of software community, in collaboration with NumFOCUS:

11. Contribute to DISCOVER event cookbook update; advertise updated DISCOVER event cookbook (1 month of work, deliver updated cookbook in year 2 (date to be determined by NumFOCUS DISC committee)

The following chart shows the timing of the activities under the initial plan, by project month.



As previously mentioned, this set of activities will likely change for a variety of reasons, including inputs from the advisory committee, the status of ongoing activities (this plan is almost certainly too ambitious unless an exceptional postdoc is hired), as well as new opportunities that arise (and are captured in the list of potential activities) and changing conditions, such as the availability of collaborators interested in specific activities. The postdoc will dedicate about 5% effort maintaining the list of potential activities (activity 1 above), which includes advertising its public availability, adding activities that arise in community discussions or are suggested by the

PI or advisory committee, and reviewing potential additions and edits from the community (i.e., reviewing and merging pull requests to the GitHub document). Potential changes in activities to be performed (from the starting list of activities above) will be discussed in advisory committee meetings (to be held virtually) 6, 12, and 18 months into the project. The advisory committee will be asked to come to a consensus on potential changes, with the PI making the final decision.

6. What will be the output from the research project?

Based on the 11 activities above, the following outputs are planned:

1. The public policy activity list, stored on GitHub.
2. Documentation about the lessons from these policy activities and a plan for an expanded and improved research policy center. This will be produced in collaboration with the URSSI governance body and shared with potential funders.
3. Documented existing (known successful/viable, known failures) career paths for individuals creating research software, publicized via blog posts and made available via GitHub for suggestions and contributions.
4. A communications list (either email, slack) for those interested in career paths.
5. Documents on URSSI chapters, such as guidance on how to set up a chapter, how to align it to local activities, and how to run it, along with 3 such instantiated chapters.
6. Documented examples of successful use of individual contributions to public goods/infrastructure to gain academic promotion (studies and language used in research statement and letters of recommendation); anecdotes about how contributing to projects has increase value within the traditional reward system (e.g., more collaborators and papers; increased opportunities to meet/work with new/old collaborators), publicized via website, blogs, and editorials.

7. Documented (anonymized) examples of how and when reviewers prioritize grant proposals that reuse, build-upon and contribute back to maintenance of public research software, publicized via ReSA website and blogs.
8. Documented cases of science/discovery where software was particularly fundamental, particularly featuring software that is not so generally well-known, stored on GitHub.
9. Tweets, blog posts and editorials based on documented cases of science/discovery where software was particularly fundamental
10. A public summary of funding opportunities for software maintenance, maintained on GitHub
11. Updated DISCOVER event cookbook, based on current NumFOCUS plan to lead this activity, distributed by NumFOCUS.

7. What is the (summary) justification for the amount of money requested?

The budget for this award is very simple, with direct labor costs consisting of 1 month annually of PI Katz's time and 1 year annually of a postdoc, plus associated fringe for both, and NCSA's IT infrastructure support fee, which has a fixed charge per work month to support computer, data, and network equipment for NCSA employees.

In addition to labor, the budget includes $20,000 of direct travel costs, as it will be essential for the postdoc to become involved in the community and to present in-progress work to build community interest and support. This travel is expected to include events such as the two major annual events sponsored by the SSI (the spring collaborations workshop (CW) on software sustainability and the fall RSE conference); a planned annual US-RSE conference; events around scholarly publishing and sharing of digital products such as the FORCE11 (https://force11.org) conference, the FORCE11 Scholarly Communications Institute (FSCI), and

bi-annual RDA meetings; meetings around information and computer science such as the iConference, CSCW, ASIS&T, and JCDL; workshops that will occur around research computing and data (RCD) professionalization and education such as those organized by CaRCC (https://carcc.org) and ADSA (https://academicdatascience.org); and disciplinary meetings that include a software focus depending on the postdocs experience and particular event agendas, such as AAS (https://aas.org), AGU (https://www.agu.org), etc.

In addition to the direct costs, the budget includes 20% overhead.

8. What other sources of research support does the proposer have in hand or has he/she applied for to support the research team?

There is no other formal research support for this work, or to support the planned postdoc. As previously stated, this work is a part of the planned URSSI implementation, and the planned URSSI governance committee will advise this project, under institutional support or under the support of other Sloan or NSF awards that are optimistically expected. Katz's previous work in this area has been supported by carving out some fraction of the partial funding for his roles at NCSA as Assistant Director for Scientific Software and Applications and as Chief Scientist, in addition to his unfunded volunteer activities.

9. What is the status and output of current and/or previous Sloan grants?

Katz has previously been financially supported on two small Sloan grants, as PI of one for $20,000 (https://sloan.org/grant-detail/8006) that partially supported the Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4), and as co-PI of one for $143,919 (https://sloan.org/grant-detail/9227) to support a US-RSE Community building workshop that has been delayed to this year due to Covid. The WSSSPE4 workshop had a published report (Katz et al. 2018) and instigated nine working groups to explore ideas that

arose in the workshop on topics such as best practices for developing research software, funding models for RSEs, academic industrial collaboration, software engineering, metrics, training, credit, software publishing, and software communities. Many of these discussions have contributed to positive changes in the field, such as software citation, various journals that review and publish research software, RSE career paths, etc. The RSE community is eagerly looking forward to the US-RSE community building workshop, which is being designed as a working meeting to plan and get to >80% outcomes on activities to grow the community, with the remaining effort needed to complete these activities to happen in the months after the workshop.

In his role as ReSA steering committee chair, Katz has also been involved in two Sloan grants to Code for Science & Society to support ReSA activities though he was not directly funded in either. In the first, for $86,000 (https://sloan.org/grant-detail/9557), ReSA provided project management to the FAIR for Research Software Working Group (FAIR4RS WG) on the development of community-endorsed FAIR (Findable, Accessible, Interoperable, Reusable) principles for research software. This project completed its two deliverables on time: an engagement plan; and a report summarizing a community-agreed definition of the FAIR principles for research software (Chue Hong et al. 2021, Chue Hong et al. 2022). In the second, now underway for $100,000 (https://sloan.org/grant-detail/9719), ReSA is establishing the Research Software Funders Forum as a collaboration of funding organizations committed to supporting research software, and those who develop it, as fundamental and vital to research. A community manager has been hired, Terms of Reference have been created, and three monthly meetings have occurred in Jan – Mar 2022, with additional meetings scheduled.

V.1 Required Appendix – List of Citations

Alinsky, Saul D. (1971). Rules for Radicals: A Pragmatic Primer for Realistic Radicals. Random House.

Babuji, Y., Woodard, A., Li, Z., Katz, D. S., Clifford, B., Kumar, R., Lacinski, L., Chard, R., Wozniak, J. M., Foster, I., Wilde, M., & Chard, K. (2019). Parsl. In Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing. HPDC '19: The 28th International Symposium on High-Performance Parallel and Distributed Computing. ACM. https://doi.org/10.1145/3307681.3325400

Berente, N., Howison, J., Cutcher-Gershenfeld, J., King, J. L., Barley, S. R., & Towns, J. (2017). Professionalization in Cyberinfrastructure. In SSRN Electronic Journal. Elsevier BV. https://doi.org/10.2139/ssrn.3138592

Berente, N., Howison, J., King, J.L., Ahalt, S. and Winter, S. (2018, June 9). Organizing and the Cyberinfrastructure Workforce. SRN. https://ssrn.com/abstract=3260715

Carver, J. C., Cosden, I. A., Hill, C., Gesing, S., & Katz, D. S. (2021). Sustaining Research Software via Research Software Engineers and Professional Associations. In 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS). 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS). IEEE. https://doi.org/10.1109/bokss52540.2021.00016

Carver, J. C., Gesing, S., Katz, D. S., Ram, K., & Weber, N. (2018). Conceptualization of a US Research Software Sustainability Institute (URSSI). In Computing in Science & Engineering (Vol. 20, Issue 3, pp. 4–9). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/mcse.2018.03221924

Carver, J. C., Weber, N., Ram, K., Gesing, S., Katz, D. S. (2022 - accepted). A Survey of the State of the Practice for Research Software in the United States. PeerJ Computer Science.

Cerna, Lucie. (2013). The Nature of Policy Change and Implementation: A Review of Different Theoretical Approaches. OECD. https://www.oecd.org/education/ceri/The%20Nature%20of%20Policy%20Change%20and%20Implementation.pdf

Chue Hong, N. P., Katz, D. S., Barker, M., Lamprecht, A.-L., Martinez, C., Psomopoulos, F. E., Harrow, J., Castro, L. J., Gruenpeter, M., Martinez, P. A., Honeyman, T., et al. (2021). FAIR Principles for Research Software (FAIR4RS Principles). Research Data Alliance. https://doi.org/10.15497/RDA00065

Chue Hong, N. P., Katz, D. S., Barker, M., Lamprecht, A-L, Martinez, C., Psomopoulos, F. E., Harrow, J., Castro, L. J., Gruenpeter, M., Martinez, P. A., Honeyman, T., et al. (2022). FAIR Principles for Research Software version 1.0. (FAIR4RS Principles v1.0). Research Data Alliance. DOI: https://doi.org/10.15497/RDA00068

Cihon, P. (2022, January 22). Open source creates value, but how do you measure it? https://github.blog/2022-01-20-open-source-creates-value-but-how-do-you-measure-it/

Crouch, S., Hong, N. C., Hettrick, S., Jackson, M., Pawlik, A., Sufi, S., Carr, L., De Roure, D., Goble, C., & Parsons, M. (2013). The Software Sustainability Institute: Changing Research Software Attitudes and Practices. In Computing in Science & Engineering (Vol. 15, Issue 6, pp. 74–80). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/mcse.2013.133

Deelman, E., Singh, G., Su, M.-H., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, G. B., Good, J., Laity, A., Jacob, J. C., & Katz, D. S. (2005). Pegasus: A Framework for Mapping Complex Scientific Workflows onto Distributed Systems. In Scientific Programming (Vol. 13, Issue 3, pp. 219–237). Hindawi Limited. https://doi.org/10.1155/2005/128026

Druskat, S., & Katz, D. S. (2018). Mapping the Research Software Sustainability Space. In 2018 IEEE 14th International Conference on e-Science (e-Science). 2018 IEEE 14th International Conference on e-Science (e-Science). IEEE. https://doi.org/10.1109/escience.2018.00014

Druskat, S., Katz, D. S., & Todorov, I. T. (2021). Research Software Sustainability and Citation. In 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS). 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS). IEEE. https://doi.org/10.1109/bokss52540.2021.00008

European Commission, Directorate-General for Research and Innovation, Mendez, E. (2020). Progress on open science: towards a shared research knowledge system: final report of the open science policy platform (R,Lawrence, editor). Publications Office. https://doi.org/10.2777/00139

GitHub. (2021). About CITATION files. https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-citation-files

Gugnani, A., & Misra, H. (2020). Implicit Skills Extraction Using Document Embedding and Its Use in Job Recommendation. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, Issue 08, pp. 13286–13293). Association for the Advancement of Artificial Intelligence (AAAI). https://doi.org/10.1609/aaai.v34i08.7038

Hettrick, Simon. (2012). What happened at the Collaborations Workshop? https://software.ac.uk/news/2012-03-27-what-happened-collaborations-workshop

Hettrick, Simon. (2014). It's Impossible to Conduct Research Without Software, Say 7 Out of 10 UK Researchers. https://www.software.ac.uk/blog/2014-12-04-its-impossible-conduct-research-without-software-say-7-out-10-uk-researchers.

Hettrick, Simon. (2018). softwaresaved/software_in_research_survey_2014: Software in research survey (1.0). Zenodo. https://doi.org/10.5281/zenodo.1183562

Hettrick, Simon. (2021). The ‘Hidden REF’ will highlight research’s unacknowledged heroes. Times Higher Education. https://www.timeshighereducation.com/blog/hidden-ref-will-highlight-researchs-unacknowledged-heroes

Katz, Daniel S. (1988). The Hypercube Implementation of the Finite-Difference Time-Domain (FD-TD) Method for Electromagnetic Wave Scattering. Unpublished. Available via https://doi.org/10.6084/m9.figshare.681783.v1

Katz, D. S. (2018). Fundamentals of Software Sustainability. In NumFOCUS Summit, New York, New York. Available via https://www.slideshare.net/danielskatz/fundamentals-of-software-sustainability

Katz, D. S., Carver, J. C., Hong, N. P. C., Gesing, S., Hettrick, S., Honeyman, T., Ram, K., & Weber, N. (2021). Addressing Research Software Sustainability via Institutes. In 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS). 2021 IEEE/ACM International Workshop on Body of Knowledge for Software Sustainability (BoKSS). IEEE. https://doi.org/10.1109/bokss52540.2021.00013

Katz, D. S., Niemeyer, K. E., Gesing, S., Hwang, L., Bangerth, W., Hettrick, S., Idaszak, R., Salac, J., Chue Hong, N., Núñez-Corrales, S., Allen, A., Geiger, R. S., Miller, J., Chen, E., Dubey, A., & Lago, P. (2018). Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4). In Journal of Open Research Software (Vol. 6, Issue 1, p. 10). Ubiquity Press, Ltd. https://doi.org/10.5334/jors.184

Katz, D. S., Chue Hong, N. P., Clark, T., Muench, A., Stall, S., Bouquin, D., Cannon, M., Edmunds, S., Faez, T., Feeney, P., Fenner, M., Friedman, M., Grenier, G., Harrison, M., Heber, J., Leary, A., MacCallum, C., Murray, H., Pastrana, E., … Yeston, J. (2021). Recognizing the value of software: a software citation guide. In F1000Research (Vol. 9, p. 1257). F1000 Research Ltd. https://doi.org/10.12688/f1000research.26932.2

Katz, D. S., McHenry, K., Lee, J. S. (2021). Research Software Sustainability: Lessons Learned at NCSA. In 54th Hawaii International Conference on System Sciences, http://hdl.handle.net/10125/71494

Katz, D. S., Piket-May, M. J., Taflove, A., & Umashankar, K. R. (1991). FDTD analysis of electromagnetic wave radiation from systems containing horn antennas. In IEEE Transactions on Antennas and Propagation (Vol. 39, Issue 8, pp. 1203–1212). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/8.97356

Katz, D. S., & Proctor, D. (2014). A Framework for Discussing e-Research Infrastructure Sustainability. Journal of Open Research Software, 2(1), e13. DOI: http://doi.org/10.5334/jors.av

Katz, D. S., & Ramnath, R. (2021). Principles of Research Software Sustainability. figshare. https://doi.org/10.6084/M9.FIGSHARE.14138036

LBNL. (2019). CSA Strategic Plan. https://cs.lbl.gov/about/strategy/#review-panel

Liu, J., Ng, Y. C., Wood, K. L., & Lim, K. H. (2020). IPOD: A Large-scale Industrial and Professional Occupation Dataset. In Proceedings of the 2020 ACM Conference on Computer Supported Cooperative Work and Social Computing Companion (CSCW'20) (pp. 323–328). https://doi.org/10.48550/arXiv.2005.02780

Mimno, D., & McCallum, A. (2008). Modeling Career Oath Trajectories. Cornell. https://mimno.infosci.cornell.edu/papers/07-69.pdf

Nangia, U., & Katz, D. S. (2017a). Track 1 Paper: Surveying the U.S. National Postdoctoral Association Regarding Software Use and Training in Research. Figshare. https://doi.org/10.6084/m9.figshare.5328442.v3.

Nangia, U., & Katz, D. S. (2017b). Understanding Software in Research: Initial Results from Examining Nature and a Call for Collaboration. In 2017 IEEE 13th International Conference

on e-Science (e-Science). 2017 IEEE 13th International Conference on e-Science (e-Science). IEEE. https://doi.org/10.1109/escience.2017.78

NSF. (2013). GPG Summary of Changes. NSF 13-1. https://www.nsf.gov/pubs/policydocs/pappguide/nsf13001/gpg_sigchanges.jsp

NSF. (2014). Dear Colleague Letter - Supporting Scientific Discovery through Norms and Practices for Software and Data Citation and Attribution. NSF 14-059. https://www.nsf.gov/pubs/2014/nsf14059/nsf14059.jsp

OECD. (2020). Building digital workforce capacity and skills for data-intensive science. OECD Science, Technology and Industry Policy Papers. 90. https://doi.org/10.1787/e08aa3bb-en.

Pielke, Roger S., Jr. (2007). The Honest Broker: Making Sense of Science in Policy and Politics. Cambridge University Press.

Philippe, O., Hammitzsch, M., Janosch, S., van der Walt, A., van Werkhoven, B., Hettrick, S., … Henwood, S. (2018, March 27). softwaresaved/international-survey: Public release for 2017 results (Version 2017-v1.2). Zenodo. http://doi.org/10.5281/zenodo.2574123

Philippe, O., Hammitzsch, M., Janosch, S., van der Walt, A., van Werkhoven, B., Hettrick, S., … Sinha M. (2019, March 6). softwaresaved/international-survey: Public release for 2018 results (Version 2018-v.1.0.2). Zenodo. http://doi.org/10.5281/zenodo.2585783

Ram, Karthik, Carver, Jeffrey, Gesing, Sandra, Katz, Daniel S., Weber, Nic. (2022). URSSI Implementation Plan. https://plan.urssi.us

Ramnath, R., & Katz, D. S. (2021). Case Studies in Scientific Software Sustainability. figshare. https://doi.org/10.6084/M9.FIGSHARE.14153303

Sarewitz, Daniel. (1996). Frontiers of Illusion: Science, Technology, and the Politics of Progress. Temple University Press.

Sholler, D., Ram, K., Boettiger, C., & Katz, D. S. (2019). Enforcing public data archiving policies in academic publishing: A study of ecology journals. In Big Data & Society (Vol. 6, Issue 1, p. 205395171983625). SAGE Publications. https://doi.org/10.1177/2053951719836258

Smith, A. M., Katz, D. S., Niemeyer, K. E. & FORCE11 Software Citation Working Group (2016). Smith, Arfon M., Daniel S. Katz, Kyle E. Niemeyer,. 2016. Software citation principles. In PeerJ Computer Science (Vol. 2, p. e86). PeerJ. https://doi.org/10.7717/peerj-cs.86

Task Force on Best Practices for Software Registries: Monteil, A., Gonzalez-Beltran, A. Ioannidis, A., Allen, A., Lee, A., Bandrowski, A., Wilson, B. E., Mecum, B., Fan Du, C., Robinson, C., Garijo, D., Katz, D. S., Long, D., Milliken, G., Ménager, H., Hausman, J., Spaaks, J. H., Fenlon, K., … , T. Morrell. (2020). Nine Best Practices for Research Software Registries and Repositories: A Concise Guide. arXiv 2012.13117 [cs.DL]. https://arxiv.org/abs/2012.13117.

Thaler, Richard H., & Cass R. Sunstein. (2008). Nudge: Improving Decisions about Health, Wealth, and Happiness. Yale University Press.

UKRI. (2021). Funding opportunity: Software for research communities. https://www.ukri.org/opportunity/software-for-research-communities/

Wilde, M., Hategan, M., Wozniak, J. M., Clifford, B., Katz, D. S., & Foster, I. (2011). Swift: A language for distributed parallel scripting. In Parallel Computing (Vol. 37, Issue 9, pp. 633–652). Elsevier BV. https://doi.org/10.1016/j.parco.2011.05.005

V.4 Required Appendix – Empirical Research Methods

Each activity will have its own method as appropriate for the activity. Two examples of the methods we will use are for Activity 3, “Document existing (known successful/viable, known failures) career paths for individuals creating research software”, and Activity 7, “With ReSA, study funder policies to understand how and when reviewers prioritize grant proposals that reuse, build-upon and contribute back to maintenance of public research software, highlight and share good examples”

Activity 3, “Document existing (known successful/viable, known failures) career paths for individuals creating research software”

Understanding the structure and dynamics of a job market is typically the domain of labor economists and organizational scholars. For example, in the domain of IT professionals and software engineers previous work has relied upon job market artifacts, such as CVs, job titles, and position advertisements, to describe how professional skills translate to career advancement (Mimno & McCallum 2008, Gugnani & Misra 2020, Liu et al. 2020). These methods of analysis are largely computational, using topic models to describe a general structure of job markets for software engineers. But, these methods are poorly suited for analyzing job transitions, or understanding the mobility of a software professional between or within specific industries. Previous work in the professionalization of cyberinfrastructure developers has similarly used descriptive case studies (e.g., drawing upon XCEDE and RENCI) to demonstrate a need for more strategic professional development support (Berente et al. 2017, Berente et al. 2018). While these previous studies provide a valuable demonstration of how descriptive work can inform career development policy, there is a gap in understanding how, and in what ways, research software careers are shaped by the institutional structures in which research software engineers are employed.

Our activity will use empirical methods to describe career pathways for research software engineers (or similar roles with different titles) who work in national labs and academia. While there are already some existing investigations of RSE careers underway by national RSE associations (Philippe et al. 2018, Philippe et al. 2019), we seek to complement this work by focusing on how institutional policies (e.g., job titles, career levels) shape a career path. To accomplish this, we will:

- Gather job title descriptions from
    - 5 Federally funded research and development centers (FFRDCs)
    - 15 R1 universities
- We will then survey ~100 individuals at these institutions who hold a job title at and above an entry level software position
    - The survey will focus on collecting education, training, and career advancement of research software engineers in these roles
    - For SEs holding advanced job titles, we will specifically ask about
        - Their career trajectory (e.g., what roles they have previously held and for how long)
        - Tenure advancement (e.g., how well does a job title reflect previous, current, and future work)
        - Career sentiment (e.g., how do individuals feel about advancing their career in a given job title - What are potential limitations, ceilings for advancement as a RSE, etc.)
        - Examples of people at their institution that they think have been particularly successful in their career, how they define success, and why they think this is the case

        - Examples of people at their institution that they think have been particularly unsuccessful in their career, how they define failure, and why they think this is the case
        - Demographics
- Based on the survey results
    - We will describe the variety of job titles
    - We will describe tenure, career path, career sentiment, demographics, and characteristics associated with success and failure in this career.
- Data collection/analysis outcome
    - A report that presents empirical results about the career paths that exists, and the experiences of individuals that are new and have advanced through careers (based on job titles), and that describes policy implications of our findings.
- Dissemination
    - We will provide the surveyed participants with the report, and we will use a combination of blogs, editorial pieces, and presentations/papers to inform the wide community of our findings.
- Path to impact
    - The surveyed individuals can use this report with their management to highlight issues in their institution, and to provide data of what other institutions are doing. US-RSE, CaRCC, and ASDA and their members can similarly use this report, both directly with their institutions and to compare with ongoing and future activities on RCS professionalization, which have been supported by US funders such as NSF, NIH, and DOE, who are

interested in how these types of roles are developed and filled, particularly as tied to the research that depends on these roles that these funders support, as well as issues related to diversity and equity in these positions.

Activity 7, "With ReSA, study funder policies to understand how and when reviewers prioritize grant proposals that reuse, build-upon and contribute back to maintenance of public research software, highlight and share good examples"

The Research Software Alliance (ReSA) hosts a Funders Forum, for funders of research software, including government, philanthropic, and other funders of research software. It currently has representatives from about 20 funding organizations and is growing. The terms of reference of this group include:

> The Research Software Funders Forum is a collaboration of funding organisations committed to supporting research software and those who develop it, as fundamental and vital to research.
>
> The Research Software Funders Forum aims to provide a formal mechanism to increase:
>
> - Sharing of funding practices for research software and the people that develop and maintain it, and learning about those practices, to encourage reflection and advancement.
> - Consideration of how to address key research software community challenges, both technical and social (such as diversity, equity and inclusivity) to achieve the significant cultural change needed across the research sector globally.

- Expansion of networks to increase sense of community and identification of collaboration opportunities.
- Consideration of opportunities to achieve long-term sustainability for research software, especially those that are part of key global infrastructures.

Given this, the Funders Forum is a natural community for this proposed activity. To accomplish it, we will interview members of the Funders Forum, some of whom may want to help with organizing and running the interviews and analyzing the data, likely including representatives from the Australia Research Data Commons and the Digital Research Alliance of Canada. We will:

- Interview Funders Forum members (10-20 interviews, depending on member willingness)
    - The qualitative, semi-structured interviews will focus on formal review criteria and practices, observed reviewer behaviors in proposals that include software development, and examples and justifications
    - We will specifically ask open-ended, guided questions about
        - How funder-wide policies and criteria are used by reviewers to review grant general proposals that include reuse, building-upon and contributing back to maintenance of public software infrastructure
        - How policies and criteria specific to particular funding calls that focus on research software are used by reviewers to review grant proposals that reuse, build-upon and contribute back to maintenance of public software infrastructure

        - Policies related to subawards or other contributions to community software in proposals
        - Examples of funded projects that funders consider exemplary as developing or contributing to public software infrastructure
        - Evidence and arguments that have been used within the funding organization related to research software funding opportunities, including those that can publicly shared
    - Interviews will be recorded if the participant agrees, with notes taken in all cases
- The interview results will be coded by the postdoc, checked by the PI independently coding a subset of the interviews, and iterated until codes are generally agreed upon.
    - Given the small number of interviews, this will be done by hand
    - We will then use search for, name, and report on themes
- Data collection/analysis outcome
    - A report that presents qualitative results about formal review policies, informal review practices, and outcomes, along with comments about how research software funding is justified by funders, and any public evidence that is used
- Dissemination
    - We will publicize the report via the Funders Forum and ReSA
- Path to impact
    - Our experience is that the Funders Forum members are eager to learn from each other, and understand how to use documents such as this report within their organization to change policies and practices, with the direct aim of

better supporting research software and the indirect aim of better supporting research.

V.5 Required Appendix – Information Products

Section 6 of the proposal body lists the planned outputs of the project. Here, we discuss general policies about these outputs, which are based on working in the open, encouraging collaboration, using tools to automatically repeat processes, and preserving source materials.

Documents will be developed in the open, typically on GitHub, Overleaf, or Google docs. These documents will be CC-BY licensed, and open to contributions, comments, and collaboration. Any published documents will be at least green open access, with pre-prints made available, in addition to the working versions that they were derived from.

Some materials developed by others with the participation of this project will be hosted by those others, but are expected to follow similar practices. For example, the DISCOVER events cookbook is hosted on the NumFOCUS website, with the source from which it was generated being available on GitHub, available for collaboration.

The project website will be public, generated from GitHub, with the website source also being public on GitHub. This website will be used primarily as an index to documents held both on the repository/website and elsewhere. Because GitHub content is archived by Software Heritage, archival versions of this content will be preserved.

Collected and analyzed data will generally be stored on GitHub, and at release points, will be archived in Zenodo. Data will also be licensed as CC-BY. When raw data includes personally identifiable information, it will only be stored locally, with an anonymized/de-identified version shared instead.

Software developed in this project will include analysis scripts and programs. These will be licensed as MIT or similar, and stored in the projects GitHub repository, and archived by Software Heritage. We will strive to make all analyses automated and reproducible, and refer to this software from any published analyses.

The PI will be responsible for all of the practices described here, and the postdoc will follow them, with training, if needed, provided by the PI.