PROJECT SUMMARY

Overview:

Modern research is inescapably digital, with data and publications most often created, analyzed, and stored electronically, using tools and methods expressed in software. While some of this software is general-purpose office software, a great deal of it is developed specifically for research, often by researchers themselves. This type of research software is essential to progress in science, engineering, and all other fields, but it is not developed in an efficient or sustainable way. The researchers, while well-versed in their discipline, generally do not have sufficient training and understanding of best practices that ease development and maintainability and that encourage sustainability and reproducibility. In response, this project will conceptualize a US Research Software Sustainability Institute. URSSI will go beyond resources like GitHub, cutting across existing activities funded by NSF and beyond, directly and indirectly positively impacting all software development and maintenance projects across all of NSF. URSSI conceptualization will include workshops and a widely-distributed survey that will engage important stakeholder communities to learn about the software they produce and use, and the ways they contemplate sustaining it. Communication will be a key component of this project, with newsletters, a web site, survey outputs, and social media used to provide broad dissemination and engagement. The workshops, survey, and community management approach will allows us to iteratively build on existing, extensive understanding of the challenges for sustainable software and its developers. Our team has accumulated hundreds of person-years of combined experience by thinking, researching, and living scientific software; we will combine this with feedback from the broader community. The results will create an eager supportive community, a concrete institute plan configured to offer valued services, and a published survey and data that demonstrates community need.

Intellectual Merit:

URSSI conceptualization will validate and address at least three classes of concerns (functioning of the individual and team, the research software, and the research field itself.) Addressing these concerns in URSSI will improve existing software and will promote the systematic, high quality release of previously sequestered ad-hoc software, which in turn will transform many science and engineering fields where knowledge is now locked away in individual laboratories or is only shared via method papers that cannot directly be used by others. Our conceptualization plan for workshops and a survey is based on the paths blazed by other successful software institutes (SGCI, MolSSI, and SSI). We have unparalleled expertise in this area, as software developers, software users, and researchers in the field of software, and our advisory board includes leaders of other software institutes and centers. We will leverage our existing collaborations to efficiently pursue this project, with most of the resources going to participant support, to expand both the community and our knowledge of its needs, to plan the best possible URSSI.

Broader Impacts:

Software underlies many of the national economic advances of the last 60 years, from better weather models that enable more productivity to better designed products that lower material usage and costs to the foundations of the Internet to deep learning systems used to optimize all segments of business, e.g., manufacturing, production, and services. Most of these advances originated in research software, initially via algorithms and licensing, and more recently via open source code. The early Internet protocols and implementation were developed as part of a physics project; the first graphical web browser was developed by an undergrad; and the initial routines that became Google search were developed by graduate students. Improving the process of generating and sustaining research software thus has clear implications for the future of software in general. URSSI conceptualization will focus on the entire research software ecosystem, including the people who create, maintain, and use research software. We will address how URSSI could formalize, diversify, and improve the pipeline under which students enter universities, learn about and contribute to software, then graduate to full-time positions where they make use of their software skills. Looking at both the front end and the interior of this pipeline, one topic will be how to use lessons learned by the computer science community to increase the diversity of those entering research software development and to retain diversity over their university careers.

1 Introduction

Modern research is inescapably digital, with data and publications most often created, analyzed, and stored electronically [1]. The processes by which these objects are created, analyzed, and stored use tools and methods that are expressed in software. While some of this software is general-purpose office software (e.g., for email, text, slide/presentation, spreadsheet), a great deal of it is developed specifically for research, often by researchers themselves [2]. Examining papers in *Nature* in January and February 2016, we find that 20 of 23 papers explicitly mention software, with each paper mentioning an average of 6 software tools. Of the 115 software tools that are mentioned in these articles, 108 are research software. Research software is essential to progress in science, engineering, humanities, and all other fields. Indeed, during the period from 1995 to 2016, the NSF has made 18,592 awards totaling $9.6 billion that topically reference "software" in their abstracts (see Figure 1.)

In many fields, most research software is produced within academia, by academics, ranging in experience and status from students, postdocs, staff members, to faculty. The academic environment in which this software is developed, maintained, and used is quite chaotic with regards to the software development lifecycle. This is partially because the academic environment and culture have developed over hundreds of years, while software has only recently become important, in some fields over the last 60+ years, but in many others, just in the last 20 or fewer years [4]. Further, it has only been recently that frameworks such as science gateways, software repositories, and virtualization have been widely available to significantly lower the barriers to sharing such software.

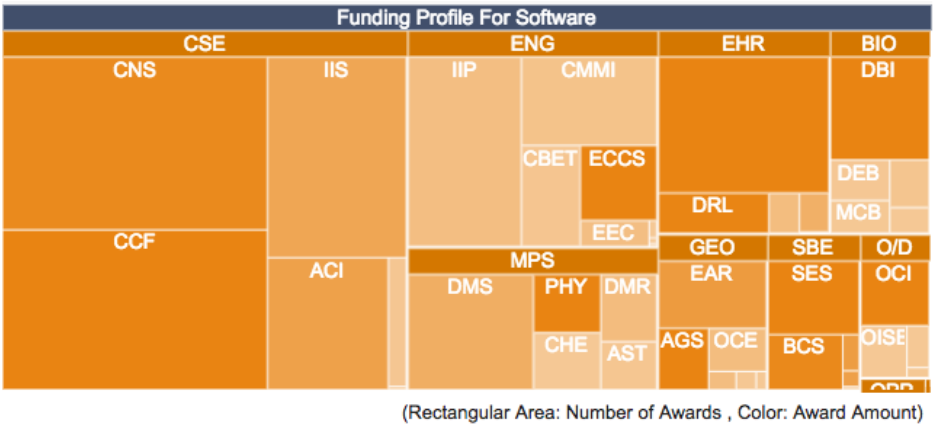

Figure 1: Funding profile for 18,592 NSF project awards from 1995-2016 that topically reference "software" in their abstracts, totaling $9.6 billion. Rectangular area: number of awards; color: award amount. (Data from [3].)

While much research software is developed in academia, important components are also developed in national laboratories and industry. Wherever research software is created and maintained, it might be open source (most likely in academia and national laboratories) or it might be commercial/closed source (most likely in industry, although industry also produces and contributes to open source.)

The open source movement has created a tremendous variety of software, including software used for research and software produced in academia. This plethora of solutions is not easy to find and use for researchers out-of-the-box [5]. Standards and a platform for categorizing software for communities is lacking and leads often to novel developments instead of reusing solutions [6]. Three primary classes of concern are pervasive across research software that have stymied it from achieving maximum impact:

- Functioning of the individual and team: issues such as training and education, ensuring appropriate credit for software development, enabling publication pathways for research software, fostering satisfactory and rewarding career paths for people who develop and maintain software, and increasing the participation of underrepresented groups in software engineering.
- Functioning of the research software: supporting sustainability of the software; growing community, evolving governance, and developing relationships between organizations, both academic and industrial; fostering both testing and reproducibility, supporting new models and developments (e.g., agile web frameworks, Software-as-a-Service), supporting contributions of transient contributors (e.g., students), creating and sustaining pipelines of diverse developers.

- Functioning of the research field itself: growing communities around research software and disparate user requirements, cataloging extant and necessary software, disseminating new developments and training researchers in the usage of software.

The goal of this conceptualization project is to create a roadmap for a *US Research Software Sustainability Institute* (URSSI) to minimize or at least decrease these types of concerns. To do this, the two aims of the URSSI conceptualization are:

1. To bring the research software community together to determine how to address the issues about which we have already learned (described below). In some cases, there are already subcommunities working together on a specific problem, including those that we are part of, but those subcommunities may not be working with the larger community [WSSSPE1]. This leads to a risk of developing solutions that solve one issue but don't reduce (or even that deepen) other concerns.
2. To identify additional issues URSSI should address and yet-to-be-identified relevant communities, how we should address or coordinate with them, and to determine how to prioritize all the issues in URSSI.

Figure 2 illustrates the key factors for creating the roadmap: the issues faced, the target groups, and the engagement with the target groups to achieve our goals during the conceptualization phase.

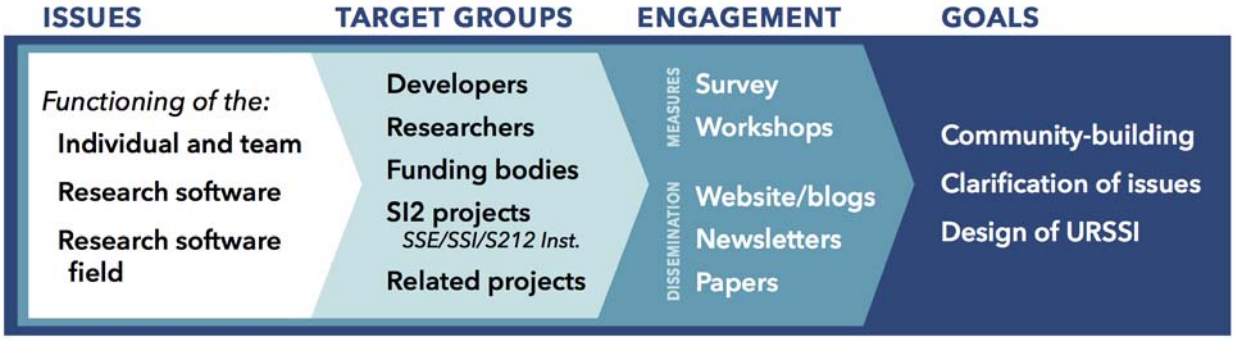

Figure 2: Key factors for URSSI conceptualization.

To conceptualize URSSI, our plan for workshops and a survey follow the paths blazed by other successful software institutes, both in the US (the Science Gateways Community Institute [SGCI], the Molecular Science Software Sustainability Institute [MolSSI]) and in the UK (the Software Sustainability Institute [SSI]), and our PIs and senior personnel have unparalleled expertise in this area, as software developers, software users, and researchers in the field of software. Additionally, our advisory board includes leaders of other software institutes and centers. We will leverage our existing connections and collaborations to efficiently pursue this project, with most of the resources going to participant support, to expand both the community and our knowledge of its needs, to plan the best possible URSSI. Diversity of perspectives, fields, and backgrounds is important in growing a robust research software community. We intend that each workshop and the survey pool have appropriate representation of women, racial/ethnic minorities, persons with disabilities, and other individuals who have been traditionally underrepresented in science, technology, engineering and mathematics (STEM).

To meet the goals of Aim 1, we need to understand existing challenges faced by the research community. Several surveys of the research community conducted in the last few year have provided effective insight. Their results emphasize the extensive use of and dependence on software and unmet needs for training in software development, as well as outlining who is doing software development and how it is being done.

Initial results of a March 2017 survey of members of the National Postdoctoral Association conducted by NCSA show that 95% of respondents (N=208) use research software. Of all the respondents, 63% could not do their work without research software, while 31% could continue their work but with more effort, and 6% would continue without a difference. Of the 197 respondents who use research software, the corresponding totals are 65%, 32%, and 3% (see Figure 3.) 28% of all the respondents develop their own software, while 72% use software developed by others. From all the respondents, 30% have received formal training

in software development, including some who also taught themselves, while an additional 16% have only taught themselves, leaving 54% with no formal or informal training. Of the 59 respondents who develop their own software, the corresponding totals are 58%, 32%, and 10%.

Also, we note that the traditional perception of postdocs as recent graduates may be flawed: only 16% of the respondents indicated they have worked in research 5 or fewer years, while 58% have done so for 6-10 years, and the remaining 25% have worked in research more than 10 years.

Similarly, a recent survey of 704 NSF PIs working in biological big data [7] found that large majorities (90% or more) need updated analysis software and training on basic computing and scripting, either immediately or in the next 3 years. And 60-80% had unmet needs in training on scaling to clouds and HPC, and multi-step analysis workflows. Another recent survey and social network analysis of NIH BD2K-funded Key Personnel indicated that approximately 30% of key contributors were not producing publications, but rather other scholarly contributions – in particular software [8].

Finally, a survey on bioinformatics infrastructure and training in 2015 [9] with 272 responses elucidated that 85.1% select the software they use because it is the best for the job and 77.6% additionally because of good documentation. On the other hand, 12.5% complained that the documentation often does not suffice to make an informed decision, and 23% complained about there being too many dependencies to install the software smoothly or even at all. The participants used free text to answer about challenges in training bioinformaticians and quite a few answers suggested that best practices for software engineering tailored to students in a multidisciplinary environments are needed. 8% directly referred to a lack of Unix skills and/or the fear of their students of using the command line. Some suggested that education in software engineering be left to experts in the field, and similarly, that education in the life sciences be left to experts in that field, to best benefit the students in gaining valuable knowledge in both disciplines.

Figure 3: Initial survey results of postdocs about research software. (Survey currently in progress at U. Illinois.)

1.1 Broader Impacts

Our broader impacts fall into two categories: improved economic and societal outcomes, and increasing the number and diversity of researchers with software development skills.

Software underlies many of the national economic advances of the last 60 years, from better weather models that enable more productivity, to better designed products that lower material usage and costs, to the foundations of the Internet, and more recently to deep learning systems used to optimize all segments of business, from manufacturing and production to services [10]. Most of these software advances are inherently tied to research software, initially via algorithms and sometimes licensing, and more recently via code in open source projects. For example, the Internet protocols and initial implementation were developed at CERN as part of a physics project, the first graphical web browser was developed at NCSA, and the initial software prototyping of what became Google's search facility was developed at Stanford. Similarly, software using physics-based research sits behind life-like animation and games, and data analysis and machine learning software enables today's financial systems.

These discoveries and outcomes were made possible by researchers with both knowledge in their domain and software development skills. We can increase the types and number of questions that are addressed with software by training more researchers in the skills needed to do software development and data analysis and supporting their careers.

By making data accessible and putting the software skills and perspectives in the hands of all researchers, we allow them to answer their own questions and capture their passion and expert knowledge. When we limit the number or types of people who do this work, we lose that curiosity, that drive, that expert knowledge. Integral to this proposal will be inclusion of these diverse groups, so that we can increase accessibility and training opportunities and scale software development skills along with our ever-increasing rates of data production.

The URSSI conceptualization workshops and survey will focus on the entire research software ecosystem, including the people and underrepresented groups who create, maintain, and use research software. Related to this, we will address how URSSI could formalize, diversify, and improve the pipeline under which students, particularly from underrepresented groups, enter universities, learn about and contribute to software, then graduate to full-time positions where they may make use of their software skills. Looking at both the front end and the interior of this pipeline, one workshop and survey topic will be how to use lessons learned by the computer science community to increase the diversity of those entering research software development and to retain members of underrepresented groups.

1.2 Solicitation-Specific Review Criteria

Rationale: URSSI will be a cross-cutting hub and set of resources, positively impacting all existing activities that include software development and maintenance across all NSF directorates, including NSF SSEs, SSIs, and the two existing S2I2s (SGCI, and MolSSI). This conceptualization proposal is the first formal step toward defining and creating that hub.

Communities and Software: The workshops and surveys will use the PIs' and senior personnel's diverse backgrounds to reach across many research and software development communities, the software they produce and use, and the ways they sustain it.

Engagement: The workshops will provide focused engagement, with the PIs and senior personnel using their diverse experiences to reach across many communities, the survey will provide broad inputs and some engagement, and the newsletters and output documents will provide broad engagement, particularly with groups underrepresented in STEM.

Impact: By addressing the three classes of concerns (functioning of the individual, the research software, and the research field itself,) an effectively designed URSSI will provide a better pipeline for research software developers, with more diverse entrants on all levels and better career paths inside. It will make research software better and more sustainable, and will promote research software to a higher standing in the research enterprise among all stakeholders.

Approach: The workshops and survey will allow us to iteratively build from our existing experience about the challenges for sustainable software and its creators and maintainers to add both focused and broad inputs from a set of diverse participants and communities, leading to an eager community, a concrete institute plan, and a published survey and data.

Qualifications: The PIs and senior personnel have a wide range of experience across many disciplines, including software, sociology, computer science, software engineering, information science, data science, mathematics, entrepreneurship, standards, and NSF disciplines funded by BIO, CISE, ENG, GEO, MPS, and SBE. The PIs have management experience with projects of this size, and have successfully worked together previously (in smaller groups) to build communities and lead community-oriented projects. The advisory board includes leaders of 3 software institutes, 2 national centers, a university data science institute, 2 industrial software leaders, and an expert in collaboration and technology, specifically the scientific software ecosystem.

2 Potential URSSI Structure and Organization

In this section, we briefly describe our current vision for the actual US Research Software Sustainability Institute. While establishing and operating this institute is not the immediate goal of this proposal, the work in this proposal is intended to be a step along the path toward that institute, and we feel it is important for readers to understand more of the path than just this initial step.

2.1 Challenges

Research software is essential to progress in science, engineering, humanities, and all other fields. It has led to Nobel prizes (e.g., Perlmutter, Schmidt, and Riess in Physics in 2011; Karplus, Levitt, and Warshel in Chemistry in 2013; Englert and Higgs in Physics in 2013) and likely will again, such as for the discovery of gravitational waves by LIGO. Here, numerical relativity simulations of Einstein's equations were extensively used to calibrate models that enabled the identification and characterization of the signatures of black hole mergers in LIGO's highly noisy data. The detection and subsequent validation of this discovery using high performance and high throughput resources was carried out with open source data analysis and workflow management software that is publicly available in LIGO's Algorithm Library. And the LIGO Open Science Center made the data segments containing the first three detected gravitational wave transients publicly available, along with a Jupyter notebook and a tutorial on signal processing with LIGO data.

Software also enables tens to hundreds of thousands of individual researchers to work on distinct aspects of grand challenges, and to have their work build into a set of knowledge, for example in bioinformatics, where computer scientists develop workflow tools, and bioinformaticians develop components, which then are combined together to solve problems. And all six research ideas in the recent NSF "10 Big Ideas" will depend on research software to make progress: "Understanding the Rules of Life: Predicting Phenotype," "Work at the Human-Technology Frontier: Shaping the Future," "Windows on the Universe: The Era of Multi-messenger Astrophysics," "Navigating the New Arctic," "Harnessing Data for 21st Century Science and Engineering," and "The Quantum Leap: Leading the Next Quantum Revolution."

Based on our experience in software development, maintenance, and usage, as well as our involvement in past SI2 PI meetings [11,12], a number of research software sustainability challenges are already known to us. Some of these include:

- Projects need to understand and plan how to start, grow, and continue, including understanding and applying sustainability models, and governance practices.
- Developers need to learn and use best practices for development (coding and maintenance), deployment, and operations.
- Many stakeholders need to track software usage & impact, which are difficult to measure and interpret.
- Developers funded by NSF, DOE, NIH, and others, including internationally, face similar and related problems to each other, but in many cases mostly communicate with others funded by their agency, and may meet in smaller groups (e.g., PI meetings) where they do not have the opportunity to develop an understanding of common problems across the field, or common solutions.
- The public and the general scientific community do not have a good understanding of the role of software in research, leading to miscast priorities for it, and insufficient recognition of its developers and maintainers.
- Students are not required to be trained in software skills, though in many fields, they are as likely to need them as the lab skills they are routinely taught.
- Science is increasingly viewed as being in a state of crisis due to lack of reproducibility, much of which goes beyond software, but some of which can be addressed through better software development, publication, and sharing practices.
- Women, persons with disabilities, and three racial and ethnic groups—African Americans, Hispanics, and American Indians or Alaska Natives—are underrepresented in science and engineering [13]. For example, while women were about 50% of the US population ages 18 to 64 in 2014, both the number and proportion of computer sciences degrees earned by women declined in 2014 to 18.1% of Bachelor's degrees, 28.8% of Master's degrees and 20.8% of Doctorates. According to the latest Census Bureau projections, minorities will account for 56% of the U.S. population by 2060, with the largest growth projected in the numbers of Hispanics, Asians, and persons of multiple races. It is crucial to increase the participation of underrepresented groups increase the participation of underrepresented groups both to provide their members with equal opportunities, and because it is important for science to have diverse participants and their viewpoints.

The UK's Software Sustainability Institute [14], which was established in 2010 to serve the the UK's research software community, has successfully addressed a number of these challenges in the UK. These include:

- Improving the digital skills of the UK research base by acting as the UK coordinators for Software Carpentry and Data Carpentry, providing effort to promote and facilitate training events, including supporting the development and training of new instructors. This has led to over 2,000 learners being trained, and lessons in computational and data skills being established in centers for doctoral training.
- Working with research software projects to improve their practice directly. This has led to the improved simulation of anti-viral drugs [15]; efficient modeling of the UK's biomass yield [16]; enabling software to improve its turnover and achieve financial stability [17]; and broadening of the user and developer communities for codes internationally [18–20].
- Getting software on the research agenda, by running workshops focused on key community challenges such as software credit, career paths and reproducibility, and contributing to governmental and funder policy. This has led to recognition in issued policy around the need for investment in research software, and the establishment of the Research Software Engineers Association, RSE Leaders Network, and EPSRC RSE Fellowships, creating new career paths for those developing research software in UK research organizations.
- Developing the next generation of passionate, informed, active champions for research software who raise and address issues specific to their own disciplines and organizations. For example, over the last five years, the SSI established a network of over 100 Institute Fellows by providing career development opportunities and support, creating champions who continue to campaign for better software practice as they advance in their careers. This has led to Fellows rising to senior positions in professional societies, research organizations, and policy departments where they can advance culture change. We hope to replicate a similar effort with URSSI.

We believe that using the UK SSI as a model will allow us to achieve similar success in addressing the challenges we have identified.

2.2 Potential Responses

We envision that the URSSI implementation could address some of the challenges identified above as follows. We note these are only starting points for conversations; which challenges URSSI will address will ultimately be determined by the overall community we intend to serve, and the conceptualization process to be implemented under this proposal.

- Help software projects determine and apply appropriate models, possibly including counseling ((similar to SGCI's Incubator service) on sustainability models, open source governance practices, and application of strategic business practices (such as "startup thinking"); matchmaking between projects and funding opportunities; instigating a Research Projects Carpentry organization (like Software Carpentry but to teach research project skills).

- Train new developers and document best practices for experienced developers, possibly including working with K-20 educators to build curriculum modules for software, often inside domain science coursework, mentoring, creation of proven templates for new research software projects, and document models for developing software in a high turnover environment.

- Influence policies, possibly including developing mechanisms to index and measure software usage/impact, adapting and developing templates for university hiring/promotion that take software into account, encouraging journals to adopt software citation principles, developing example language for funding agencies for CFPs for open source software and software management plans, and encouraging institutions to use them.

- Sustain software, possible with staff members at partner institutions who work for short periods on projects proposed to the institute then competitively selected (a la XSEDE ECSS).

- Coordinate and collaborate with the other efforts on scientific software productivity and sustainability efforts through joint workshops, initially with the US Department of Energy (DOE) laboratory community, and possibly by organizing contributions to the Guide to the Software Engineering Body of Knowledge (SWEBOK Guide) [21] focused on specific communities.
- Communicate software work to both the science (and research) community that directly benefits from the software as examples of success, as well as the wider science community and the public, working with education & public outreach (EPO) parts of computational and data science centers, museums, publishers, etc.
- Improve career paths, career sustainability, and workforce pipelines, including by increasing the participation of underrepresented communities, and possibly including developing RSE-like career paths for developers, and funding URSSI fellowships.
- Provide strong ethical and legal guidance for licensing and distribution of software, knowledge, and data products (including awareness of reuse issues around unlicensed products), possibly including supporting community coordination and review on licensing and sustainability plans, providing templates and examples of licensing documentation and software management plans.
- Create a Responsible Conduct of Research Code for software developers and users, analogous to the Hippocratic Oath for medicine. The institute would offer faculty the opportunity to educate their students in software practices that if followed, would allow the persistence of the research products beyond the life of the student who produced it and increase the impact of research at large. The center would offer a service to train and monitor the application of such practices, similar to CTSC audits for security, to help research group leaders gain an understanding of how "sustainable" their groups are based on good software practices.
- Develop concrete measures of reproducibility and provide guidance for researchers to define and implement reproducibility in the context of their scientific domains. Given the diversity of standard and tooling, the URSSI could offer advice on automation and provenance tracking, availability of data and software, software documentation and engineering, and issues related to copyright.
- Identify and evaluate existing "point" solutions for the above challenges; determine which models (such as ECSS-style support or Software Carpentry-style educational activities) fit within an integrated, cohesive institute effort.

2.3 URSSI and SI2 coordination

In addition to coalescing the US research software community to address the challenges above, URSSI will also act as the point of coordination for the NSF SI2 program and its awardees, as well as related NSF projects as recommended by SI2 program management. URSSI will maintain a list of active and expired SI2 awards and the software they produce, to encourage potential users and collaborators to work with this set of software with the knowledge that it is being funded by NSF. URSSI will also lead the annual SI2 PI meeting, with the set of organizers changing each year, including both URSSI staff and other SI2 PIs. Finally, URSSI will gather SI2-specific concerns from the

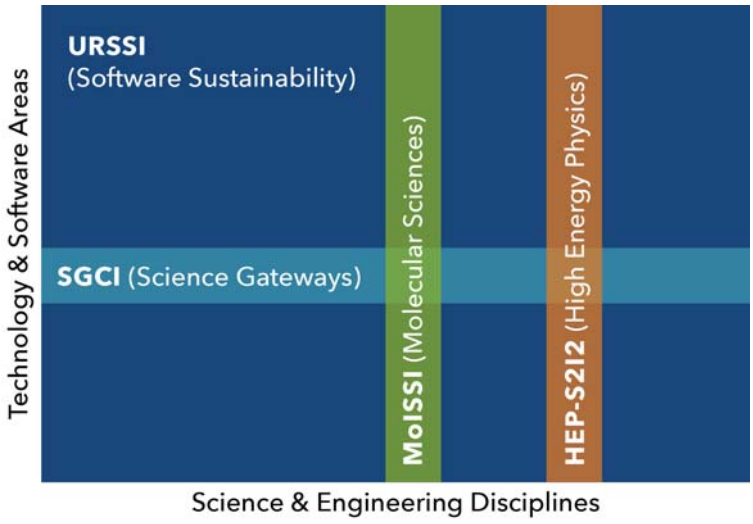

Figure 4: URSSI and other S2I2 projects in the science-technology space

SI2 PIs to understand which are common, and prioritize communication of them with NSF, including sharing responses with the SI2 PI community.

As a broad software institute, URSSI will also coordinate and work with other SI2 institutes (S2I2s), as shown in Figure 4. While we specifically include the two current institutes and the one active conceptualization as examples, we also intend to work with other conceptualizations that are awarded under this 2017 SI2 solicitation. Where other institutes cover the space of technologies and science disciplines, URSSI will defer to the other institutes. For example, when projects go the SGCI that are not purely gateways, URSSI and the SGCI will work together to answer their questions. And URSSI will take the lead in coordinating on common areas over time, so that the other institutes can focus on their unique challenges. But where there are no other institutes, URSSI will attempt to cover the technology (software) and science (user) space.

2.4 The Expected URSSI Impact

After 5 years of URSSI, we expect that the research software world will be different as a consequence, with the impact felt in at least three ways. The first will be through direct participation with research software teams, resulting in:

- A set of software across a variety of fields that has been improved by URSSI consultants and URSSI-trained developers; such packages will acknowledge these efforts and the URSSI contributions will be seen as a “quality stamp?? for these packages.
- Software projects advised through URSSI will reach higher levels of sustainability, as measured by increased engagement with research communities, increased measured usage and citation, and expanded development communities (development and maintenance supported by more individuals, at a larger number of institutions and organizations.)

Second, the general services that URSSI offers will provide resources to the entire research software community, such as:

- Best practices for Software as a Service for research software that will be established and disseminated to the research community. This may include greater understanding of industry practices as well as how these need to be adapted and modified for research software and the development models commonly used for research software.
- Workforce development activities that result in an increase in the number of highly skilled developers who are working on research software systems and an increase in their diversity in research domains, education, gender, and ethnicity.
- Software training will be common in many departments in universities, in the same way lab training is now common, and more universities will offer software training in a non-curricular method. URSSI will be aligned with Software and Data Carpentry and other “carpentry?? efforts to promote the training and shared experiences needed for successful projects of all scales, not just successful individual researchers. URSSI will be the “host of last resort?? for synchronous and asynchronous training material, and will work with large disciplinary communities to customize content for them. Almost all such material will be openly shared, via CC-BY or more permissive licenses.

Third, as a formal institute and through its involvement with the research software community, URSSI will be an important voice on issues regarding the elevation of research software as a recognized intellectual contribution. Specifically, we hope that outcomes of our work will be as follows:

- Software will be discussed more and will receive increased attention. Specifically, more NSF announcements and solicitations will be aware that software is a key element of research, and will have instructions for proposers to describe software and its disposition, and for reviewers to judge it. Additionally, we expect that experts in all aspects of the software lifecycle will be invited to participate in review of proposals. We expect similar improvements in DOE, NIH, and private foundation opportunities. URSSI will be seen as the central US entity that promotes and improves research software. The URSSI web site will be a central place for researchers to find out about the latest tools and newest opinions on the state of research software, and URSSI-organized and URSSI co-sponsored events will be where research software developers meet. Members of the URSSI staff will be invited keynote speakers at a variety of general and discipline-specific conferences and workshops.

- Software will be seen as a valid research product. Software metrics will be an accepted part of faculty hiring and promotion decisions in universities, and more universities will provide opportunities for software professionals to build careers. URSSI-suggested language will be commonly found in recommendation letters. Software will be viewed in parallel with data and other non-traditional scholarly products.

3 URSSI Conceptualization Process and Activities

The aims for this conceptualization project are:

1. To bring the research software community together to determine how to address the issues about which we already know (described above).
2. To identify additional issues URSSI should address, how we should address them, and to determine how to prioritize all the issues in URSSI.

To address the two URSSI conceptualization aims, our activities comprise a set of workshops, a broad survey, and a set of community engagement activities, scheduled as shown in Figure 5. These will lead to the following outcomes and outputs by the end of the project:

1. A diverse and engaged community that supports the idea of URSSI and understands how it will benefit their work
2. The survey data and a paper analyzing it, with the paper published as a preprint and submitted for journal or conference publication
3. A public (licensed as CC-BY and published as a preprint or technical report) strategic plan for an URSSI implementation project, including discussion of community and research questions that the S2I2 will interact with and support; specific relevant software; sustainability challenges for the software and for the institute; development, deployment, and usage processes; required infrastructure and technologies; the required organizational, personnel and management structures and operational processes, and corresponding budget and effort estimates; mechanisms for human resource development that proactively address; and potential risks
4. An NSF final report on the conceptualization project

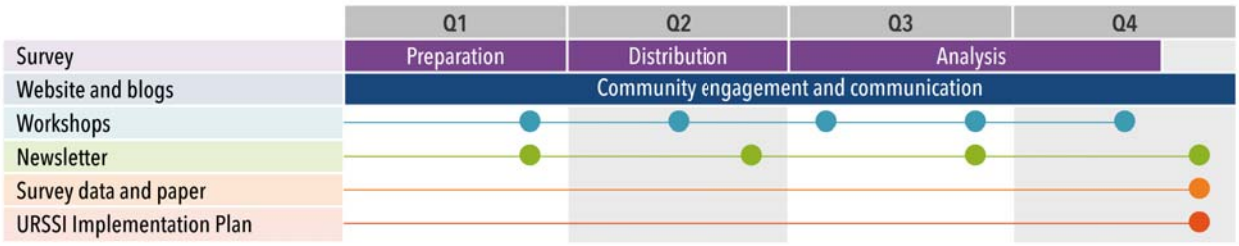

Figure 5: Timeline of the Conceptualization of the URSSI, showing public engagement and outcomes.

The design of the SGCI used inputs from focus groups, workshops, and a large survey. Based on the SGCI team?s positive experience in gathering requirements and building a vibrant community around this institute, we assume that these methods will also benefit the overall URSSI vision and broaden its communities. Even though there was a clear vision for SGCI in 2009 developed via an EAGER grant, important aspects and priorities changed through the community involvement in the conceptualization process, e.g., instead of developing a software framework, the SGCI became technology agnostic and now supports multiple mature frameworks under active development. We expect similar changes to our current vision to design an institute that fulfills the needs of our communities.

3.1 Workshops

Workshops are the core of our conceptualization project, and we will use about 60% of the budget for participant support costs to allow the largest possible number to participate. We will begin with a 3-day general community workshop (to be held in Month 3 at Berkeley), with about 90 attendees, aimed at building community support, determining areas where URSSI can impact the community, and areas where more discussion is needed. The PIs and senior personnel will work together (led by Ram and Katz) to define the topics and invited attendees of this initial meeting, with guidance from the advisory committee members.

Based on six previous Workshops on Sustainable Software for Science: Practice and Experience (WSSSPE) [22–24] with 200-300 total distinct attendees and SI2 PI meetings with another 200 attendees, we have a good starting point for understanding potential topics, such as were used in WSSSPE4: development and community, professionalization, training, credit, software publishing, software discovery and reuse, and reproducibility and testing. We will issue a call for submissions with a set of topics closely related to these, and also offer submitters the opportunity to propose their own topics. All submissions will be in the form of extended abstracts, to be presented in groups (by topics) similar to panels, but with questions and discussion not focused on the particular presentations but on the more general topics themselves. The goal of this workshop will be to determine what topics the community considers well-understood, where there is work to be done but we know how to start the work, and to determine what topics are not well-understood, where we need more discussion before we can start implementing solutions.

The results of this workshop, in particular the areas where more discussion is needed, will be used to determine the subjects for 3 smaller 2-day workshops (in Portland, Chicago, and Baltimore, in Months 5, 7, and 9), each with 30-40 attendees. The goal of these workshops is again building community support in these areas and focusing the work that URSSI will be planned to do. Depending on the targeted issues, a workshop might focus on a single topic, a set of closely-related topics, or a set of diverse topics. Workshop attendees will include URSSI conceptualization staff, invited participants, end-users, and call for submission respondents.

The PIs will ensure that all workshops have diverse attendees, including those from MREFCs, large and small NSF projects, industry, labs, varied disciplines, as well as the more traditional diversity measures, such as gender, ethnicity, etc. The consortium will aim at increased participation of underrepresented communities through reaching out to organizations already involved with XSEDE and the SGCI, and through groups that focus on women and minorities. With this in mind, members of minority serving institutions will receive priority in the selection process for invited talks. Our goal is to spend at least 25% of the participants support budget on bringing in participants from underrepresented groups.

We have allocated sufficient funds in the budget to support the full travel costs of the expected minimum number of attendees for all workshops. With some participants local to each workshop city, we should be able to support close to the maximum numbers we discuss above, including some key international invitees, such as representatives from the UK SSI. In particular, we want to ensure that sufficient travel support is available to all members of underrepresented groups who we ask to participate.

3.2 Survey

The survey effort will be led by PI Carver, with support from the PIs and Senior Investigators. The diverse expertise of the team, including expertise in survey design/execution, domain science, and computer science/software engineering, will allow us to build an effective survey to capture key inputs from the community. We will also build upon the lessons learned from recent surveys on similar topics (including a survey conducted by the Science Gateways Community Institute [25] and an in-progress U Illinois survey of the National Postdoctoral Association, as well as a survey from the UK Software Sustainability Institute [26].)

The survey effort will begin early in the grant period. In Months 1-3 we will design the survey. The survey will focus on gathering input on the key challenges, as described earlier in the proposal. The goal will be to understand both current practices and future needs in each of these areas. By capturing current practices, we will be able to identify successful solutions employed by segments of the community and share those with the larger community. By capturing needs, we will be able to ensure that design of the URSSI Institute addresses those needs. The survey questions will explicitly address the following topics (with specific questions developed during the first part of the project period):

1. Sustainability models and governance
2. Tracking software use
3. Communicating the value of software work to external constituents
4. Career paths
5. Software development practices
6. Training needs , both for students and for professionals
7. Software licensing

We will ensure that additional topics can organically arise from the responses.

In Months 4-6, we will distribute the survey to a random sample of NSF PIs and via the networks of the PIs and Senior Personnel, to ensure broad coverage of relevant stakeholders. By distributing the survey widely, we will ensure that our process captures the input from stakeholders who do not attend the workshops, yet still have valuable contributions to the direction of the project. In Months 7-11, we will analyze and document the survey results.

In addition to being useful for the planning of the URSSI Institute, we envision that the results of the survey will be of interest to others in the community. Therefore, in Month 12, we plan to make the results available to the community in multiple ways: (1) we will include the results in the final project report, (2) we will publish the dataset in an open access format, and (3) will publish the analyzed results as an open access paper. Finally, we envision this survey serving as a baseline for a series of longitudinal surveys to track changes in the community over the course of the URSSI Institute.

3.3 Community engagement and management

Community engagement and management will be led by PI Gesing. Five main measures, which complement and are interconnected with each other, are planned for the conceptualization phase:

1. The workshops will be the initial measure to reach out to the community and initiate discussions, brainstorm, and create results to identify the main areas of the anticipated institute. These will extend our current vision of the institute and will include requirements and visions from a broad community served by research software.
2. The next major measure for the outreach is the planned survey. It will have several phases and we intend to reach out with the online survey to a much broader audience then we can do in person at the workshops, especially at an international scale. We aim to synergize by asking the attendees of the first workshop to answer the questions of the first draft of the survey and also to review the survey to propose further questions or would like to add potential topics.
3. A web presence is crucial nowadays for a successful community project; we will set up a website with information on the goals of the conceptualization phase and the developing vision of the intended institute. It will include information about the project itself and its progress, as well as how it fits in the NSF sustainability landscape and in the research landscape on research software. The website will include news on the project that will be updated regularly with all major steps and achievements, such as workshop reports and iterations of the institute plan.
4. In addition to the more static part of the website, we will publish our own regular blogs and provide options for contributor blogging (publishing at least two blogs a month, guided by the challenges and solutions surfaced in the workshops and survey) and commenting. We will also set up email lists so that people can subscribe to receive regular (every 3 months) newsletters discussing project accomplishments and news, also posted on the website. The email lists will start with the audience of the workshops, and will be advertised for others to join.
5. The fifth measure is reports and papers on the collected data and on the outcome of the workshops and the survey. The reports will be openly accessible and we plan to submit about two papers to conferences or related events, and give presentations about the project.

We are already at a solid starting point for reaching out to the communities to which the PIs and senior personnel of this project are connected. To ensure we address underrepresented groups in STEM, we will reach out to organizations already involved with XSEDE and the SGCI, groups that focus on women and minorities, and organizations with minority membership such as the National Medical Association and to

chairpersons of relevant departments at the over 100 Historically Black Colleges and Universities in the USA. The five measures will extend and formalize the community that is particularly interested in the goals and visions of building the URSSI institute, and will strengthen the final URSSI concept.

3.4 Final Meeting

A final 2-day working meeting of the PIs, senior personnel, and interested advisory committee members will be used to write the two final project outputs: the final report and the URSSI strategic plan (see §3 introduction.) This meeting will provide good focused discussions and contributions among a relatively small set of community representatives. The final team meeting will be held in Berkeley in Month 11.

3.5 People and Roles

URSSI conceptualization will be led by the PIs:

- Karthik Ram, PI of rOpenSci, which provides open-source tools to help address the technical and social challenges associated with open science, centered around the use of R in data science.
- Jeffrey Carver, Assoc. Prof. in Computer Science at U. Alabama, works in empirical software engineering for science and engineering.
- Sandra Gesing, Research Asst. Prof in Computer Science & Eng. and the Center for Research Computing at Notre Dame, focuses in her research on usability and re-usability of research software via science gateways and on reproducibility of science via computational workflows. As computational scientist of the CRC, she designs and consults projects on software frameworks. She is part of the SGCI and works in the Community engagement and Exchange and Incubator service areas. She founded the successful European workshop series IWSG (International Workshop on Science Gateways) in 2009 and still guides it.
- Daniel S. Katz, Asst. Director for Scientific Software and Applications at NCSA, coordinating research software development and maintenance, and research application support. Katz previously led the NSF ACI software cluster and is a software developer of workflows and other parallel and distributed tools. He led the organization of 4 WSSSPE workshops (2 at SC14/15 and 2 standalone, see §3.6) He also co-leads the FORCE11 Software Citation Working Group (see §3.6), and studies how software can be treated as a scholarly product.
- Nicholas Weber, Asst. Prof. in Information Sciences at U Washington, uses ethnographic work in the context of scientific collaboration to contribute to the fields of Science and Technology Studies (STS) and Computer Supported Cooperative Work (CSCW).

Our senior personnel are:

- Wolfgang Bangerth, Colorado State University, is a computational scientist and mathematician who leads the deal.II and ASPECT open source projects with together many hundred users and dozens of contributors from across many disciplines.
- Anshu Dubey, Argonne National Laboratory & University of Chicago, is interested in research software lifecycle and process design for sustainability. She has led the development of FLASH, a scientific community code for multiple domains. She also led a previous S2I2 conceptualization effort for a software institute on abstractions for sustainability of high performance computing scientific codes.
- Melissa Haendel, Oregon Health & Science University, leads large-scale semantic data integration efforts to enable biological mechanistic discovery (Monarch Initiative, NIH Data Translator). She has also been coordinating work on contribution roles for tracking non-traditional scholarly contributions and their implementation in biosketch systems such as SciENcv. Finally, she is an active proponent of open science, and coordinates cross-community groups within FORCE11, the Biocuration society, and the Biden Blue Ribbon Panel on open data sharing.
- Michael A Heroux, St John's University and Sandia National Laboratories, is interested in all aspects of scientific software productivity and sustainability. He leads the Trilinos scientific libraries project, and several other scientific computing software projects. He leads the scientific and math libraries effort for the Department of Energy's Exascale Computing Project.

- Kathryn Huff, University of Illinois Urbana-Champaign, Blue Waters Assistant Professor, Dept. of Nuclear, Plasma, and Radiological Engineering. Her research focuses on modeling and simulation of advanced nuclear reactors and nuclear fuel cycles. Through a history of leadership within the Hacker Within, Software Carpentry (SC&DC, see §3.6), SciPy, the Journal of Open Source Software, and other initiatives she strives to advocate for best practices in open, reproducible scientific computing.
- Suresh Marru is Deputy Director of Science Gateways Research Center, Pervasive Technology Institute, Indiana University. His research focuses on Cyberinfrastructure development and teaches Architecture courses using Open Source technologies. He coordinates Google Summer of Code within Apache Software Foundation.
- Kate Mueller is the Managing Director of the Center for Social Research, which is dedicated to improving the quality and efficiency of social research including survey design and administration, and a Concurrent Professor of Law at the University of Notre Dame.
- Jarek Nabrzyski is the director of the Center for Research Computing at the University of Notre Dame, where he oversees a group of 45 full time research programmers, computational scientists, HPC engineers and data scientists.
- Kyle Niemeyer, Oregon State University, works in computational modeling of combustion and chemically reacting fluid flows, with applications in energy systems, transportation, and aerospace.
- Marlon Pierce, Director of the Science Gateways Research Center at Indiana University. He manages the XSEDE Science Gateways program and is a Co-PI of the Science Gateways Community Institute (SGCI, see §3.6.)
- Ariel Rokem, University of Washington, Senior Data Scientist at the eScience Institute. His research and software development work focuses on tools for processing biomedical images. He is also a Software Carpentry instructor and instructor trainer (SC&DC, see §3.6) and directs the Neurohackweek summer school in neuroimaging and data science. He co-PIs the NSF-funded West Big Data Hub.
- Arfon Smith, Space Telescope Science Institute, leads the Data Science Mission Office (DSMO), and is Editor-in-Chief of the Journal of Open Source Software (JOSS, see §3.6.)
- Tracy Teal, Executive Director of Data Carpentry, a non-profit organization building community and training researchers in data analysis and management skills to enable data-driven discovery. (SC&DC, see §3.6.)
- Matthew Turk, University of Illinois Urbana-Champaign.
- Rick Wagner, Argonne National Laboratory & University of Chicago, is the Globus Professional Services Manager, working with research projects to integrate the Globus Saas and PaaS cloud products.
- Michael Zentner, Purdue University, directs the HUBzero science gateway platform, leads the Science Gateway Community Institute's Incubator program, is co-PI on the nanoHUB science gateway project, is an Entrepreneur in Residence at Purdue's Foundry, and is a serial entrepreneur.

Our senior personnel also include as Advisory Committee members:

- Richard Arthur, General Electric Global Research, is responsible for the vision, strategy, and coordination of computing research from embedded devices through supercomputing.
- Michelle Barker, Deputy Director, Research Software Infrastructure for Nectar, Australia's National eResearch Collaboration Tools and Resources project.
- Philip E. Bourne, former NIH AD for Data Science and incoming Director of the University of Virginia Data Science Institute.
- Neil Chue Hong, PI of UK Software Sustainability Institute (SSI, see §3.6.)
- Daniel Crawford, PI of the Molecular Sciences Software Institute (MolSSI, see §3.6.)
- James Howison, Assistant Professor at UT Austin studying open source software development and the development of software in science as examples of collaboration.
- Kurt Schwehr, Head of Ocean Engineering and GIS Data Engineer for Oceans at Google and an affiliate Faculty in CCOM/JHC, Computer Science and Earth Science at University of New Hampshire.

- Jeff Spies, co-founder and CTO of Center for Open Science.
- Nancy Wilkins-Diehr, PI of the Science Gateways Community Institute (SGCI, see §3.6.)

3.6 Management Team

The PIs will be responsible for the project, working together with the senior personnel to implement the activities, using electronic communication to obtain feedback and guidance from the advisory committee. While all PIs will be active in planning all activities, Ram and Katz will lead the first workshop and the final team meeting. Carver and Gesing will lead the design and analysis of a survey to collect feedback on preferences, current practices, and attitudes of software developers across a range of scientific disciplines. Gesing will also be responsible for setting up the website, blogs and newsletters. Mueller and Weber will also conduct ethnographic work at workshops and within a select group of scientific software development communities. Both sets of activities, survey and ethnographies, will contribute to a richer understanding of the motivations, norms, social structures, and collective practices of the diverse communities that URSSI hopes to reach. Two to three senior personnel will co-lead in organizing the smaller workshops, to be determined after the first workshop based on the needed foci.

The PIs and senior personnel have a wide range of experience across many disciplines, including software, sociology, computer science, software engineering, information science, data science, mathematics, entrepreneurship, and NSF disciplines funded by BIO, CISE, ENG, GEO, MPS, and SBE, and have previously been involved in many software and community building projects. Members of the URSSI Conceptualization team are already collaborating in a number of software community projects, as shown in Figure 6. The leads for the projects are also mentioned in §3.5.

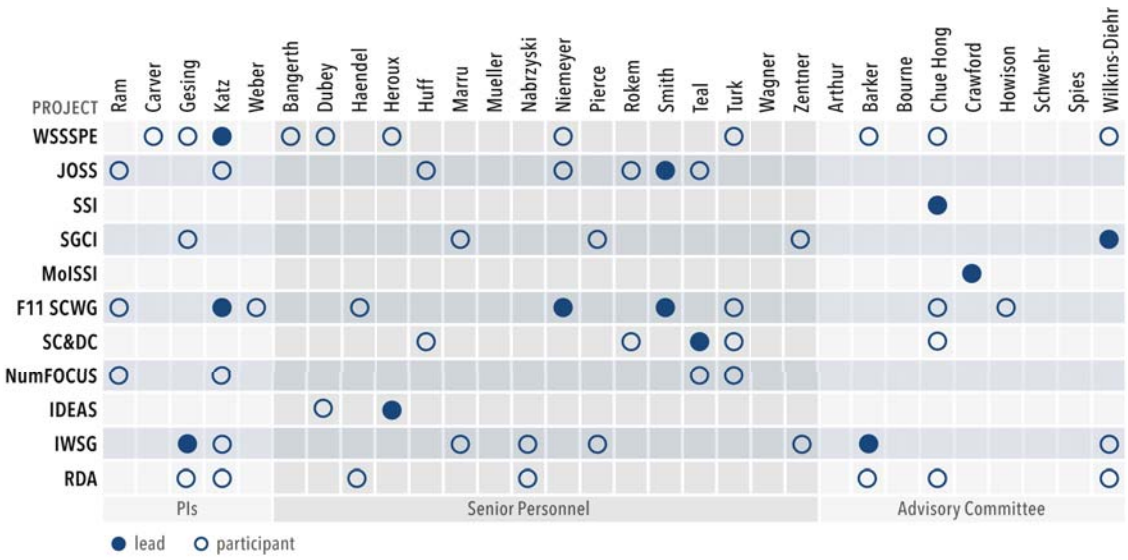

Figure 6: Existing collaborations of URSSI team in software community projects. WSSSPE = Workshop on Sustainable Software for Science: Practice and Experiences; JOSS = Journal of Open Source Software; SSI = UK Software Sustainability Institute; SGCI = Science Gateways Community Institute; MolSSI = Molecular Science Software Institute; F11 SCWG = FORCE11 Software Citation Working Group; SC&DC = Software & Data Carpentry; NumFOCUS = NumFOCUS project leaders and board members; IDEAS = DOE IDEAS Software Productivity project; IWSG = International Workshop on Science Gateways; RDA = Research Data Alliance.

In addition, the PIs and senior personnel work with and are funded by a variety of public and private agencies, including in addition to NSF, DOE, NIH, the Moore Foundation, the Sloan Foundation, and the Mellon Trust. In particular, Ram, Weber, and Rokem are associated with two of the three Moore-Sloan Data Science Environments, and Turk is a 2014 Moore Investigator in Data-Driven Discovery. In addition, Haendel co-leads the NIH NCATS Data Translator initiative, and Heroux co-leads the DOE IDEAS software productivity project. We believe that this conceptualization and the later institute will be of strong interest to

these other public agencies and private foundations, and that a coalition of funders could be assembled by NSF to share this institute and its funding.

4 Results from Prior NSF-funded work

Karthik Ram: Ram has not received NSF support with a start date in the past five years.

Jeffrey Carver: 1445344, Amount: $190,411, Dates: 8/1/14–7/31/17, Title: EAGER: Collaborative Research: Making Software Engineering Work for Computational Science and Engineering: An Integrated Approach. Intellectual Merit: This project has supported the study of peer code review in scientific software and the organization of multiple International Workshops on Software Engineering for Computational Science and Engineering [27]. Broader Impacts: This work has facilitated the interaction between software engineers and computational scientists and has produced the following publications [28, 29]).

Sandra Gesing: Gesing's closest related NSF award is the SGCI. 1547611, Amount awarded to date: $6,599,000, Dates: 8/1/16–7/31/21, Title: S2I2: Impl: The Science Gateways Community Institute (SGCI) for the Democratization and Acceleration of Science. Intellectual Merit: By being easily accessible via the Web, science gateways expand and democratize access to supercomputers, telescopes, sensor networks, unique data collections, collaborative spaces that enable the multidisciplinary collaborations needed to solve complex problems, and analysis capabilities. Thus, science gateways expand and broadening participation in science. The SGCI will speed the development and application of robust, cost-effective, sustainable gateways to address the needs of scientists and engineers across the sciences. The work of the institute will increase the number as well as the effectiveness and usability of gateways to science and engineering. Broader Impacts: The SGCI broadens access to advanced CI resources by increasing the number and robustness of science gateways across NSF domains. It will bring research-grade tools into the classroom and allow citizen scientists to participate in meaningful ways. Institute services helps broaden the impact of clients' gateway projects. With its training programs, the Institute will increase the number of gateway developers and increase inclusion of underrepresented minorities. Outputs: one peer-reviewed paper [30], two peer-reviewed abstracts [31, 32], one journal special issue [33], an organized conference with 120 participants [34], an active blog, a monthly newsletter (1300 subscriptions,) a monthly webinar, and 36 presentations at meetings, related conferences or universities.

Daniel S. Katz: Katz was an NSF program officer from 5 to 1 year ago; his current NSF awards were made within the last year and do not yet have outputs. One representative award is 1659702, Amount $360,000, Dates: 3/1/17–2/29/20, Title: REU Site: INCLUSION - Incubating a New Community of Leaders Using Software, Inclusion, Innovation, Interdisciplinary and OpeN-Science. Intellectual Merit: This project will build and improve open source software across a variety of areas, which will enable a wide variety of new scientific and engineering knowledge. Broader Impacts: The project immerses a highly motivated and diverse cohort of students in a guided research process that enables them to experience the thrill of discovery and to adopt the role of scientists as one which is authentically theirs, inspiring them to pursue research careers.

Nicholas Weber: Weber has not received NSF support with a start date in the past five years.

5 Conclusions

Data, discoveries, and publications in modern research are quite frequently created, analyzed, and stored electronically using tools and methods expressed in software, often developed by academic researchers. However, software development in academia suffers from a number of factors related to the academic environment, including at least three classes of concerns that are pervasive across research software and that have stymied it from achieving maximum science impact: 1) functioning of the individuals and teams who develop and maintain research software; 2) functioning of the software itself, including usability, sustainability, and reproducibility; and 3) functioning of the field of research software more generally.

In this conceptualization project, we will validate and address the concerns, planning an institute that will provide a better pipeline for research software developers, with more diverse entrants and better career paths. The institute will also make research software better and more sustainable, and it will promote research software to a higher standing in the research enterprise among all stakeholders.

References

[1] National Science Foundation. A vision and strategy for software for science, engineering, and education: Cyberinfrastructure framework for the 21st century, 2012. NSF 12-113, http://www.nsf.gov/publications/pub_summ.jsp?ods_key=nsf12113.

[2] Jo Erskine Hannay, Carolyn MacLeod, Janice Singer, Hans Peter Langtangen, Dietmar Pfahl, and Greg Wilson. How do scientists develop and use scientific software? In *2009 ICSE Workshop on Software Engineering for Computational Science and Engineering*, pages 1–8, May 2009.

[3] Deep insights anytime, anywhere. http://www.dia2.org.

[4] Ian Foster. 2020 computing: A two-way street to science's future. *Nature*, 440(7083):419, 2006.

[5] Lucas N. Joppa, Greg McInerny, Richard Harper, Lara Salido, Kenji Takeda, Kenton O'Hara, David Gavaghan, and Stephen Emmott. Troubling trends in scientific software use. *Science*, 340(6134):814–815, 2013.

[6] James Howison, Ewa Deelman, Michael J. McLennan, Rafael Ferreira da Silva, and James D. Herbsleb. Understanding the scientific software ecosystem and its impact: Current and future measures. *Research Evaluation*, 24(4):454, 2015.

[7] Lindsay Barone, Jason Williams, and David Micklos. Unmet needs for analyzing biological big data: A survey of 704 NSF principal investigators. *bioRxiv*, 10.1101/108555, 2017.

[8] Melissa Haendel. Credit where credit is due: acknowledging all types of contributions. In *9th Conference on Open Access Scholarly Publishing (COASP)*, 2016. https://www.slideshare.net/mhaendel/credit-where-credit-is-due-acknowledging-all-types-of-contributions.

[9] Nicholas Loman and Thomas Connor. Bioinformatics infrastructure and training survey. *fighshare*, October 2015. https://doi.org/10.6084/m9.figshare.1572287.v2.

[10] President's Council of Advisors on Science and Technology. Designing a digital future: Federally funded research and development in networking and information technology, 2010. http://web.archive.org/web/20161219102833/https://www.whitehouse.gov/sites/default/files/microsites/ostp/pcast-nitrd-report-2010.pdf.

[11] Frank Timmes, Matthew Turk, Stan Ahalt, Shaowen Wang, Ray Idaszak, Richard Brower, Chris Lenhardt, and Karl Gustafson. 2016 Software Infrastructure for Sustained Innovation (SI2) PI workshop. Technical report, USA, 2016. Available from: http://cococubed.asu.edu/si2_pi_workshop_2016/ewExternalFiles/nsf-si2piw_2015.pdf.

[12] Frank Timmes, Stan Ahalt, Matthew Turk, Ray Idaszak, Mark Schildhauer, Richard Brower, Chris Lenhardt, and Karl Gustafson. 2015 Software Infrastructure for Sustained Innovation (SI2) principal investigators workshop. Technical report, USA, 2015. Available from: http://cococubed.asu.edu/si2_pi_workshop_2016/ewExternalFiles/SI2_PI_2016_report_final.pdf.

[13] National Science Foundation. Women, Minorities, and Persons with Disabilities in Science and Engineering, 2017. https://www.nsf.gov/statistics/2017/nsf17310/.

[14] Stephen Crouch, Neil Chue Hong, Simon Hettrick, Mike Jackson, Aleksandra Pawlik, Shoaib Sufi, Les Carr, David De Roure, Carole Goble, and Mark Parsons. The Software Sustainability Institute: Changing research software attitudes and practice. *Computing in Science & Engineering*, 15(6):74–80, 2013.

[15] Christopher J. Woods, Katherine E. Shaw, and Adrian J. Mulholland. Combined quantum mechanics/molecular mechanics (QM/MM) simulations for protein–ligand complexes: Free energies of binding of water molecules in influenza neuraminidase. *The Journal of Physical Chemistry B*, 119(3):997–1001, 2015. PMID: 25340313.

[16] Software Sustainability Institute. Biofuel research potential grows with updated software. https://www.software.ac.uk/resources/case-studies/biofuel-research-potential-grows-updated-software.

[17] Software Sustainability Institute. Helping a research project transform into a business service. https://www.software.ac.uk/resources/case-studies/helping-research-project-transform-business-service.

[18] Software Sustainability Institute. Building a firm foundation for solid mechanics software. https://www.software.ac.uk/blog/2014-11-21-building-firm-foundation-solid-mechanics-software.

[19] Gillian Law. Magnetic imaging software now FABBERlously easy to use, 2014. https://www.software.ac.uk/blog/2014-10-17-magnetic-imaging-software-now-fabberlously-easy-use.

[20] Software Sustainability Institute. Improving climate modelling and making it accessible to new users. https://www.software.ac.uk/improving-climate-modelling-and-making-it-accessible-new-users.

[21] Pierre Bourque and R.E. Fairley, editors. *Guide to the Software Engineering Body of Knowledge, Version 3.0*. IEEE Computer Society, 2014. http://www.swebok.org.

[22] Daniel S. Katz, Sou-Cheng T. Choi, Hilmar Lapp, Ketan Maheshwari, Frank Löffler, Matthew Turk, Marcus Hanwell, Nancy Wilkins-Diehr, James Hetherington, James Howison, Shel Swenson, Gabrielle Allen, Anne Elster, Bruce Berriman, and Colin Venters. Summary of the first workshop on sustainable software for science: Practice and experiences (WSSSPE1). *Journal of Open Research Software*, 2(1), 2014.

[23] Daniel S. Katz, Sou-Cheng T. Choi, Nancy Wilkins-Diehr, Neil Chue Hong, Colin C. Venters, James Howison, Frank J. Seinstra, Matthew Jones, Karen Cranston, Thomas L. Clune, Miguel de Val-Borro, and Richard Littauer. Report on the second workshop on sustainable software for science: Practice and experiences (WSSSPE2). *Journal of Open Research Software*, 4(1):e7, 2016.

[24] Daniel S. Katz, Sou-Cheng T. Choi, Kyle E. Niemeyer, James Hetherington, Frank Löffler, Dan Gunter, Ray Idaszak, Steven R. Brandt, Mark A. Miller, Sandra Gesing, Nick D. Jones, Nic Weber, Suresh Marru, Gabrielle Allen, Birgit Penzenstadler, Colin C. Venters, Ethan Davis, Lorraine Hwang, Ilian Todorov, Abani Patra, and Miguel de Val-Borro. Report on the third workshop on sustainable software for science: Practice and experiences (WSSSPE3). *Journal of Open Research Software*, 4(1):e37, 2016.

[25] Katherine A. Lawrence, Michael Zentner, Nancy Wilkins-Diehr, Julie A. Wernert, Marlon Pierce, Suresh Marru, and Scott Michael. Science gateways today and tomorrow: positive perspectives of nearly 5000 members of the research community. *Concurrency and Computation: Practice and Experience*, 27(16):4252–4268, 2015. CPE-15-0033.R1.

[26] Simon Hettrick, Mario Antonioletti, Les Carr, Neil Chue Hong, Stephen Crouch, David De Roure, Iain Emsley, Carole Goble, Alexander Hay, Devasena Inupakutika, Mike Jackson, Aleksandra Nenadic, Tim Parkinson, Mark I Parsons, Aleksandra Pawlik, Giacomo Peru, Arno Proeme, John Robinson, and Shoaib Sufi. Uk research software survey 2014, December 2014. Available from: https://doi.org/10.5281/zenodo.14809.

[27] International workshops on software engineering for computational science and engineering. http://se4science.org/workshops.

[28] Dustin Heaton and Jeffrey C. Carver. Claims about the use of software engineering practices in science: A systematic literature review. *Information and Software Technology*, 67:207 – 219, 2015.

[29] Jeffrey C. Carver, Neil Chue Hong, and Selim Ciraci. The 4th International Workshop on Software Engineering for HPC in Computational Science and Engineering. *Computing in Science Engineering*, 19(2):91–95, Mar 2017.

[30] Sandra Gesing, Nancy Wilkins-Diehr, Maytal Dahan, Katherine Lawrence, Michael Zentner, Marlon Pierce, Linda Hayden, and Suresh Marru. Science Gateways: The Long Road to the Birth of an Institute. In *Proc. of HICSS-50 (50th Hawaii International Conference on System Sciences)*, 2017.

[31] Sandra Gesing, Maytal Dahan, Linda Hayden, Katherine Lawrence, Suresh Marru, Marlon Pierce, Michael Zentner, and Nancy Wilkins-Diehr. The Science Gateway Community Institute – Supporting Communities to Achieve Sustainability for their Science Gateways. In *Proc. of WSSSPE4 (4th Workshop on Sustainable Software for Science: Practice and Experiences)*, 2016.

[32] Sandra Gesing, Michael Zentner, Maytal Dahan, and Katherine Lawrence. Gateways to Science: Harnessing Big Data and Open Data for Precision Medicine. In *Book of Abstracts – 2nd Personalized Medicine Conference*, pages 55–57, 2017.

[33] Sandra Gesing, Nancy Wilkins-Diehr, Michelle Barker, and Gabriele Pierantoni. Special Issue on Science Gateways. *J. Grid Comput.*, 14(4):495–703, 2016.

[34] *Proc. of Gateways 2016 Conference*. 2016. `https://gateways2016.figshare.com/`.