OCSDO
OFFICE OF THE CHIEF
SCIENCE DATA OFFICER

2024 SOFTWARE FOR NASA SMD WORKSHOP REPORT

NASA Science Mission Directorate

May 7-9, 2024

TABLE OF CONTENTS

Workshop agenda, recordings, and presentations can be found here.

EXECUTIVE SUMMARY

The 2024 Software for the NASA Science Mission Directorate Workshop was the first workshop of its kind in over 10 years. The numerous attendees and high level of interaction in this hybrid workshop portrayed the untapped interest and energy in the NASA SMD community about software. Over 100 takeaways were collected from the hosted discussions in four key areas - Communication, Communities, Funding and Clarity - as summarized in this report. One takeaway was representative of all four categories:

> *“Software is not hardware; it is organic and needs a different model. You often don't know which components will be Open-Source reusable until later in the development cycle.”*

For the communication category, this takeaway motivates the reduction of silos by shifting towards a model that better supports community collaboration on common challenges and a more streamlined and improved software release process. For the communities category, the same statement points to forming communities of practice with varying scopes and a new approach to recognition and incentives for open-source software contributions. Concerning funding, this statement motivates a more sustained and flexible funding model that better supports the software foundation needed for NASA’s long-term success, including the collaboration and infrastructure a good foundation requires. The new approach to clarity motivated by this statement calls for significant changes to the software release process and related policies to streamline compliance, align those policies with the open-source science culture NASA is promoting and with each other, and simplify use of the cloud.

It is time to recognize software as a foundational component of NASA with an organic nature not properly supported by current approaches. Different models are needed in all four areas to shift the NASA SMD software community and governance structures into a more efficient, open, and collaborative ecosystem - one that enables ground-breaking science and daring exploration into the coming decades

Key Takeaways

In addition to the takeaway mentioned in the previous section, ten key takeaways and trends were noted in the analysis of the >100 takeaways gathered from workshop attendees. They are presented in brief form here, with no implied ranking, and discussed in the sections below.

- Training, education, related tools, and increased alignment are needed for NPR 7150.2, NPR 2210.1, and related policies and software practices.
- The Software Release Process needs significant streamlining, clarity, and improvement, including simplified processes for the release of open-source software and legacy codes and improved uniformity across centers.
- A new Open-Source Software Fellow rotating position, recruited from NASA scientists or software developers involved in writing open-source software, is needed to provide input into software-related policies, processes, trainings, and practices.
- NASA’s software catalogs need improvement, consolidation, and a user-friendly search interface.
- Increased support of Communities of Practice will help improve the current lack of communication and collaboration within and across divisions and increase feedback on software-related policies and processes.
- The community that attended this workshop considered the workshop beneficial in several ways and should meet every 1-2 years in a similar fashion.
- A more sustained, flexible, and consistent funding model is needed to increase the long-term benefits and usefulness of NASA-funded software.
- Purposefully designed funding pathways are needed to better incentivize contributions to open-source software and collaborations across missions, divisions, centers, and organizations.
- Key infrastructure is needed to better support software lifecycle management and cyber-compliance.
- The software community needs guidance and increased exposure to the cloud, cloud-optimized data formats, and NASA’s HPC resources (e.g., workshops that use these resources) to increase the community’s use of these resources.

MEETING SUMMARY

Introduction

The 2024 Software for the NASA Science Mission Directorate Workshop was the first NASA-wide workshop focused on software in over ten years. The workshop aimed to explore the current opportunities and challenges for software relevant for activities funded by the NASA Science Mission Directorate (SMD). This includes increasing the visibility of software being developed and igniting wider collaborations on those challenges between centers, missions, data repositories, and sciences. The official objectives of the workshop were to:

- Bring together those interested in software for the NASA Science Mission Directorate to facilitate new collaborations across centers and sciences.
- Identify opportunities and challenges in different stages of the software lifecycle.
- Identify software being developed, areas of expertise, resources, and best practices for the NASA software community.
- Provide community feedback to prioritize paths forward.

Over 150 people attended the hybrid workshop, with nearly 100 people attending in person and at least half as many people attending virtually. This report briefly describes the workshop logistics, summarizes an analysis of the takeaways, and provides suggested paths forward to improve the NASA software landscape in four categories: communication, communities, funding and clarity. Links to additional materials supporting this report are available in the Reference section.

Description

The in-person component of the hybrid 2024 Software for the NASA Science Mission Directorate Workshop was held at NASA headquarters in downtown Washington, D. C. on May 7-9, 2024. A significant effort was made to equalize the interaction capabilities for virtual and in-person attendees by incorporating several technologies and approaches. The highlighted talks were chosen to align with the most highly prioritized software categories for the attendees indicated during registration (see the next section) and to provide focus on NASA-specific software policies. These talks provided context for the attendees to dynamically contribute and select discussion topics for the remainder of the workshop. This structure enabled the NASA software community to focus their discussions on the topics of most importance to them and provide input into NASA's future priorities in alignment with those interests. More details on the dynamic approach and the supporting technologies are available in the 'Logistics Lessons Learned' document in the supporting materials.

Statistics

The 2024 Software for the NASA Science Mission Directorate Workshop was well attended despite the short timeframe for registration and abstract submission. Advertising was also limited to the efforts and reaches of the program and executive committee members. Those efforts were generally limited to NASA centers with few exceptions due to the NASA-specific purpose of the workshop. Registration for in-person attendance and abstract submission was open for about one month, with registration for virtual attendance open for approximately two months. In that timeframe, 326 unique registrations were received, with roughly 70% of registrants having a NASA identity, 113 requesting in-person attendance, and 87 of the total, including a verified abstract. Due to facility limitations, additional requests to attend in-person were not possible to fulfill despite multiple requests. The high demand for the workshop indicates a significant unanswered need in the NASA-funded software community.

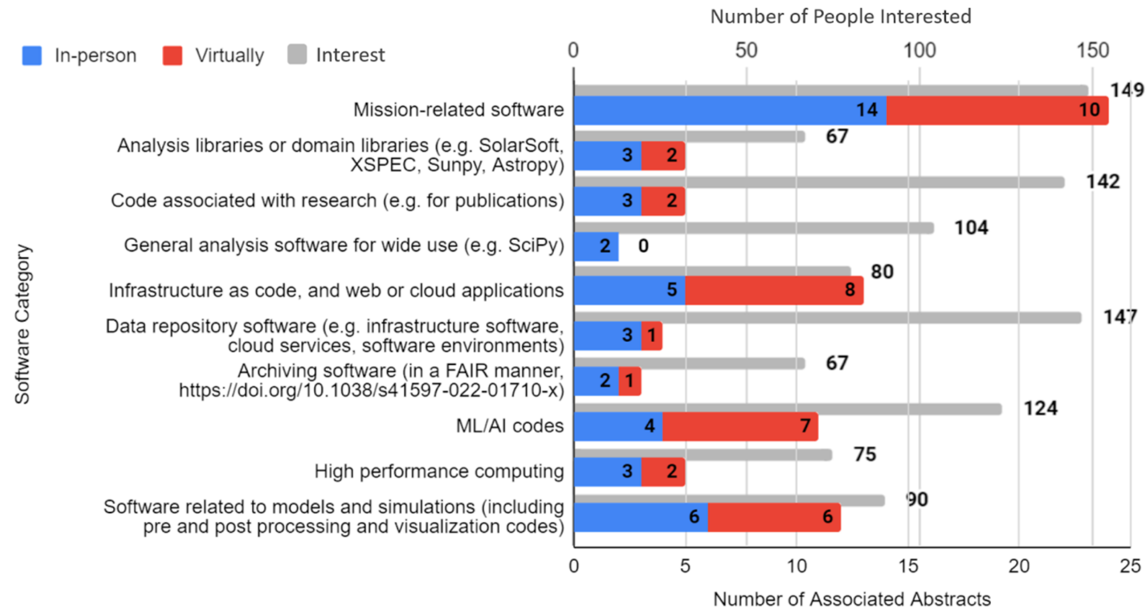

Figure 1*: Distribution of the items associated with each software category. The ten software categories are listed on the left. The number of votes for each software category are indicated by the gray bars measured with the top horizontal axis. Blue bars indicate an in-person abstract and red bars indicate a virtual abstract for the given software category, both measured by the lower horizontal axis. The numbers at the end of each bar indicates the value of that item.*

As part of the registration process, attendees indicated the top three software categories and lifecycle stages of most importance to them. The distribution of their answers is shown by the gray bars in Figure 1 for the software categories. Similar plots are available in the appendix for the software lifecycle stages. Surprisingly, the software categories (and lifecycle stages) associated with the abstract submissions did not follow a similar distribution except for the mission-related software category (see the colored and gray bars in Figure 1). This could be an effect of factors limited to this workshop, such as the limited scope of advertising and the small timeframe for registration, or also factors specific to NASA, such as the funding landscape for software development in each category or lifecycle stage, or even a lack of collaboration opportunity through workshops similar to this one. No information on potential causes for these differences was obtained from the attendees.

The distribution of the software categories for abstract submissions and attendees' interests provides context for the takeaways analyzed in the following section. The interests of attendees tended to prioritize issues concerning mission-related software, data repository software, code associated with research, and ML/AI codes, which propagated through the takeaways and into the resulting paths forward represented in this report.

TAKEAWAY ANALYSIS

Analysis Method

Immediately following the conclusion of the workshop, the available program committee members chose four categories to represent the main themes of the takeaways for the workshop: Communication, Communities, Funding and Clarity. The program committee plus one member of the executive committee then worked to:

1. Categorize the 128 takeaways into these four categories,

2. Predict the needed effort and expected impact of the takeaways,

3. Select a few of the takeaways in each category as ‘quick wins’ (e.g. high impact + low effort),

4. Summarize the trends of the remaining takeaways in each category.

The spreadsheet used for the analysis is available in the Takeaway Themes Categorization file in the supporting materials below.

All committee members agreed that one takeaway was representative of all four categories: “***Software is not hardware, it is organic and needs a different model. You often don't know which components will be Open-Source reusable until later in the development cycle.***” The same statement is also true when comparing software to data, publications, and other components of work supported by NASA. Software contributors need a wider communication capability than those currently available, a less siloed community than the few that currently exist, a more flexible funding model than those currently possible, and increased clarity on the relevant NASA policies. For example, the push for open data has resulted in new related structures such as data management templates and specified funding opportunities. Now, as NASA pushes for open code as part of the transition to Open Science, similar advances are needed for software, and the efforts already begun should be amplified.

The subsections below provide the insights obtained during the review of the takeaways in each category and the top prioritized suggestions based on the effort-vs-impact analysis performed.

Communication

The need for increased communication using a variety of approaches was readily apparent in the analysis of the takeaways relevant to this category.

- The TOPS curriculum has been an outstanding success but should be expanded to include training for the most discussed topics at this conference, NPR-7150.2, NPR-2210.1, and software practices. Training and collaborative opportunities beyond TOPS could be very impactful for these topics, such as summer schools and collaboration workshops between NASA divisions. These workshops can also help the community transition to the cloud and leverage NASA's HPC resources to control costs.

- Many takeaways focused on streamlining and improving the software release process (SRP), with emphasis on better documentation of the release process itself, detailed examples of the requisite documents and processes for each class and type of software under SPD-41A, defining software metrics, and automating standard forms and documents such as the software management document. Large Language Models could automate these time-consuming documents.

- NASA's software catalogs need improvement and consolidation, including a discovery page enabling search of all NASA-related open-source software repositories, to decrease duplication of effort.

- Incentives need to be provided to encourage developing and contributing to OSS projects to increase practitioners' competitiveness and likelihood in securing future funding.

Finally, these efforts should enable the NASA software community and associated missions and projects to reduce silos, duplication of efforts and cost for common features and capabilities, and shift software development focus to new mission-specific capabilities not addressed by existing OSS projects.

Communities

The greatest takeaway from the Communities theme is that NASA should prioritize the development of communities of practice (CoP) both within and across the various divisions. Communities can help communicate information to their members (horizontally), but also to NASA leadership (vertically). This allows all involved to better appreciate the Open-Source Software philosophy, and why it is important for NASA moving forward. It also allows suggested policy changes, such as recognition and incentives for those that use open source and publish it, to be discussed and disseminated to the appropriate personnel.

Additionally, the community that formed to run this workshop should continue to meet at least every other year, if not, every year to keep momentum going.

Funding

NASA's Science Mission Directorate (SMD) has established several funding programs aimed at enhancing the accessibility and impact of scientific research through open-source initiatives. For example, the High Priority Open-Source Science (HPOSS) program focuses on developing new technologies and capacity-building materials to advance open science practices. However, while programs like HPOSS and Supplements for Support for Open-Source Tools, Frameworks, and Libraries (OSTFL) provide valuable support, there is a recognized need for more sustained and flexible funding models that go beyond the project-driven approach currently predominant in NASA's funding structure. This approach often fails to adequately support the long-term development and maintenance of generalized, reusable software that could benefit multiple missions across SMD.

To effectively sustain the move towards open-source software, there is a critical need for investment in key infrastructure, particularly in the areas of software lifecycle management and cyber-compliance. The establishment of software Communities of Practice (CoPs) across NASA would be instrumental in fostering collaboration and raising awareness about emerging technologies. These communities could address current challenges, such as the lack of consolidated licenses and the high costs associated with acquiring necessary software tools. Furthermore, SMD must ensure that widely used tools receive consistent support, both in terms of funding and development resources. Implementing a more streamlined and efficient process for distributing funds across divisions and centers, with an emphasis on incentivizing cross-mission collaboration, would be a significant step forward.

The creation of a new funding structure that supports multi-mission and cross-organizational development is essential for the success of NASA's open-source initiatives. Just as the Deep Space Network (DSN) has become indispensable for maintaining communication with countless missions across the solar system, software has similarly evolved into a foundational capability that underpins the development and operation of these missions. Like the DSN, software is now critical infrastructure, deserving of dedicated funding to ensure its sustainability and ongoing support for the success of current and future missions.

By continuing to incorporate multi-mission and cross divisional support as well as code reuse into the selection criteria for funding opportunities, NASA can drive greater efficiency and innovation across its missions. Additionally, ensuring that open-source funding is not tied exclusively to current grants or specific missions will allow for more flexibility in achieving broader scientific goals. This approach will help bridge existing gaps between early-stage projects funded by programs like HPOSS and more mature software efforts, ultimately leading to a more cohesive and robust open-source ecosystem within NASA.

Clarity

Analysis of the takeaways relevant to the Clarity category produced four main issues, with related suggestions, each issue by increasing effort.

Issue	Recommendations
Streamline and clarify the steps to release open-source software (OSS).	• Expand exemptions from the SRP to include software written by civil servants and contractors, primarily supports a publication, or legally vendors third party code. • Provide software-specific training with examples for each step of the process. • Automate the software release process (SRP) by reducing duplicate data entries, providing API access to the New Technology Report and software release form. • Embed open-source software packaging and distribution experts into the Software Release Authority and intellectual property counsel organizations, ideally as an “Open-Source Software Fellow” rotating position that is recruited from NASA scientists or software developers who are involved in writing OSS. • Make the SRP experience more uniform across centers. • Improve software processes through education, not primarily compliance enforcement.

The current SRP makes it nearly impossible to formally release and continually improve high-value legacy software developed by NASA centers (e.g. HEASoft).	• A pragmatic and flexible approach is needed to better support such software.
There is a need for increased awareness, clarification, and alignment of policies related to open-source software, including SPD-41a and NPR 7150.2	• NPR 7150.2 should be modified to align with SPD41a. • Create TOPS training course(s) for policies around open software, how to ensure compliance, and how to interpret and align with third-party open-source licenses. • Update policies to conceive of and fund software as a capability. • Add clear and concise guidance for software management plans to recognize and document contributions to existing software projects and ecosystems. • Simplify external contributions by providing clear and concise instructions and removing the NASA ownership requirement. • Implement incentives and recognition for OSS-related accomplishments.
Guidance for cloud-based resource usage is needed.	• The use of cloud-optimized data file formats should be encouraged, and data formats should be standardized whenever possible. • Software should be built using reusable and reproducible packages and software architecture to enable data access and analysis on a variety of compute platforms.

Discussion Topic Analysis

In addition to the takeaways from the discussions at the workshop, attendees contributed 47 suggested discussion topics in the registration form (see the 'Other Discussion Topics' tab of the takeaways spreadsheet). Of these, 22 were discussed at the workshop. Ten of the remaining topics aligned with one of the categories discussed above and four were specific to a science topic or similar (e.g. software for hazards). The remaining 11 discussion topics aligned along three common themes: (1) Best practices for software (e.g. verification and validation), (2) Incorporating FAIR and Open Science principles into software (e.g. long-term preservation of software), and (3) HPC-related and modeling topics. The second topic is of timeliness as NASA and other agencies work towards supporting FAIR and Open Science for software in their policies and proposal requirements, warranting a cross-agency collaboration with inclusion of relevant expertise (e.g. the Open Modeling Foundation, CodeMeta, Software Heritage, and established software communities of practice). In addition, there were also topics within each category of software that would warrant their own discussions such as the unique challenges for flight software or the modeling community.

Post-Workshop Survey Results

A post-workshop survey was advertised to attendees at the end of the workshop and twice afterwards. Exactly 50 responses were obtained during the three weeks following the workshop (see Supporting Materials for details). The responses were overwhelmingly positive, with 47 of the respondents indicating benefit or potential benefit from the workshop (questions 1-6), and 48 indicating some level of confidence that NASA benefited from the workshop (question 7). In support of objective 1 for the workshop, 33 of the 50 respondents reported forming a new collaboration, several of which were across different NASA centers or disciplines (questions 1 and 2). Positive answers in support of objective 3 were reported from 46 respondents (questions 3-6), with 40 of those respondents reporting gaining at 3 out of the 4 possible benefits from the workshop (excluding forming new collaborations). Perhaps the most telling statistic from that survey was that an overwhelming majority of respondents (48) indicated a desire for additional workshops like these (question 8), with just over half of the respondents requesting annual workshops.

---

[1] https://doi.org/10.5281/zenodo.12665701
[2] https://codemeta.github.io/
[3] https://www.softwareheritage.org/

Paths Forward

Each of the four key objectives of the workshop were fulfilled by the new collaborations and opportunities discovered by the attendees, and by the dynamically chosen discussions and the resulting takeaways analyzed here. Additionally, the feedback from the attendees obtained from this workshop converges towards several paths forward.

- First, the NASA software community is eager to collaborate openly on software in ways that are currently made difficult by funding, policies, the software release process, and other constraints - all barriers that must be decreased to improve NASA's effectiveness.

- Second, the community is unified in their requests for more opportunities and methods to communicate about and discover software, whether through a publicly available discussion forum focused on NASA-funded software, through focused workshops or hackathons, larger workshops like this one, or an improved software search interface.

- Third, there is an overwhelming call for improvement in the understandability of resources for the NASA software release process and related policies, such as an AI-powered Q&A interface to guide one through a given policy to determine what is needed for compliance.

- Fourth, there was almost unanimous agreement on repeating this workshop routinely with some improvements in the logistical execution.

The 2024 Software for the NASA Science Mission Directorate Workshop was the first NASA-wide workshop focused on software in over ten years. The unexpected high response rate, participation, and positive feedback led to the potential paths forward presented here to improve numerous aspects of software for the NASA Science Mission Directorate.

ACKNOWLEDGEMENTS

We thank the participants of the workshop for their contributions, presentations, and meaningful discussions that have contributed to this report. We thank the staff at NASA Headquarters for supporting the workshop and enabling the event. The authors of the workshop report are indicated by a “*” in the following list. Rebecca Ringuette, the workshop lead, produced the main body of the text, committee members analyzed and summarized the four categories of takeaways and provided comments and suggestions on the full text, and the executive committee performed final edits and other supporting tasks. The program committee members contributing to the analysis and summary of each of the four takeaway categories are as follows:

- Communication: *Jon Jenkins and Ashish Acharya*
- Communities: *Josh Steele and Crystal Gummo*
- Funding: *Demitri Muna and Paul Ramirez*
- Clarity: *Leo Singer and Amanda Saravia-Butler*

Workshop Lead:

Rebecca Ringuette* (ADNET Systems, Inc./NASA GSFC)

Program Committee:

Ashish Acharya* (University of Alabama, Huntsville/NASA MSFC)
Crystal Gummo* (ADNET Systems, Inc. / NASA Langley Research Center)
Jon Jenkins* (NASA ARC)
Maria Kuznetsova (NASA GSFC)
Paul Ramirez* (Jet Propulsion Lab)
Amanda Saravia-Butler* (KBR Space Biology/NASA ARC)
Leo Singer* (NASA GSFC)
Josh Steele* (Johns Hopkins APL)

Executive Committee:

Steven Crawford* (NASA HQ)
Brian A. Thomas* (NASA GSFC)
Mark Parsons (University of Alabama, Huntsville / NASA MSFC)
Demitri Muna* (Agile Decision Sciences / NASA HQ)

Workshop Support:

Holly Norton, Paul Fertitta, Paul Tate, Iman Nasif (Booz Allen Hamilton)

Report Formatting:

Amanda Adams (NASA MSFC), Jeamay Palo (Science Systems & Applications Inc.)

Workshop Sponsor: Office of the Chief Science Data Officer, Science Mission Directorate, NASA

SUPPORTING MATERIALS

Workshop website with event recordings and agenda:
https://science.data.nasa.gov/news/software-workshop-2024/

Workshop Zenodo community with presentations, posters, and other conference materials:
https://zenodo.org/communities/2024softwareforthenasasmdworkshop/records

Additional support material is available at https://zenodo.org/records/13968054.

Analysis of registration and abstract information referenced in this report:

Ringuette, R., Crawford, S., Thomas, B., Parsons, M., & Muna, D. (2024). 2024 Software for the NASA SMD Workshop: Logistics Materials. Zenodo. https://doi.org/10.5281/zenodo.12668746

Items included in this Zenodo submission:

- A pdf of this report
- Logistics Lessons Learned
- Post-workshop Survey form
- Post-workshop analysis spreadsheet
- Takeaway Categorization analysis spreadsheet

APPENDIX A

This appendix contains two plots similar to Figure 1 but for the software lifecycle stages.

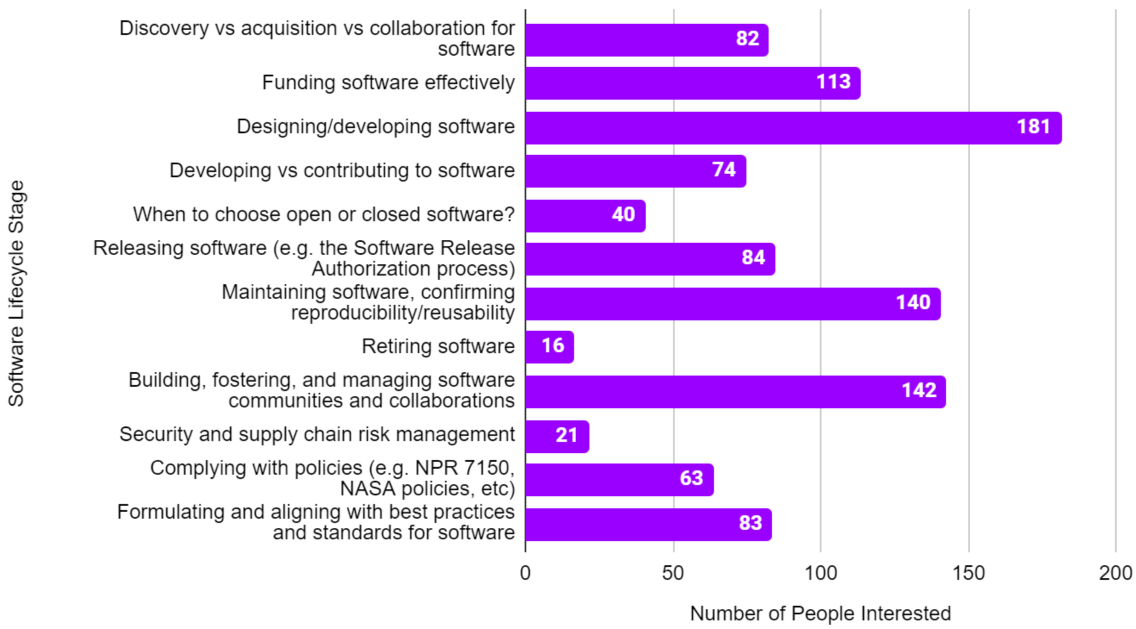

***Figure A.1**: Distribution of the importance of software lifecycle stages. The twelve software lifecycle stages are listed on the left with the number of votes for each software category indicated by the purple bars on the right. The numbers at the end of each purple bar indicates the number of votes the software lifecycle stages received. Each registrant indicated roughly three lifecycle stages each. An alternate version of the table with percentages is available in the RegistrationAnalysis.xlsx file in Ringuette et al. (2024).*

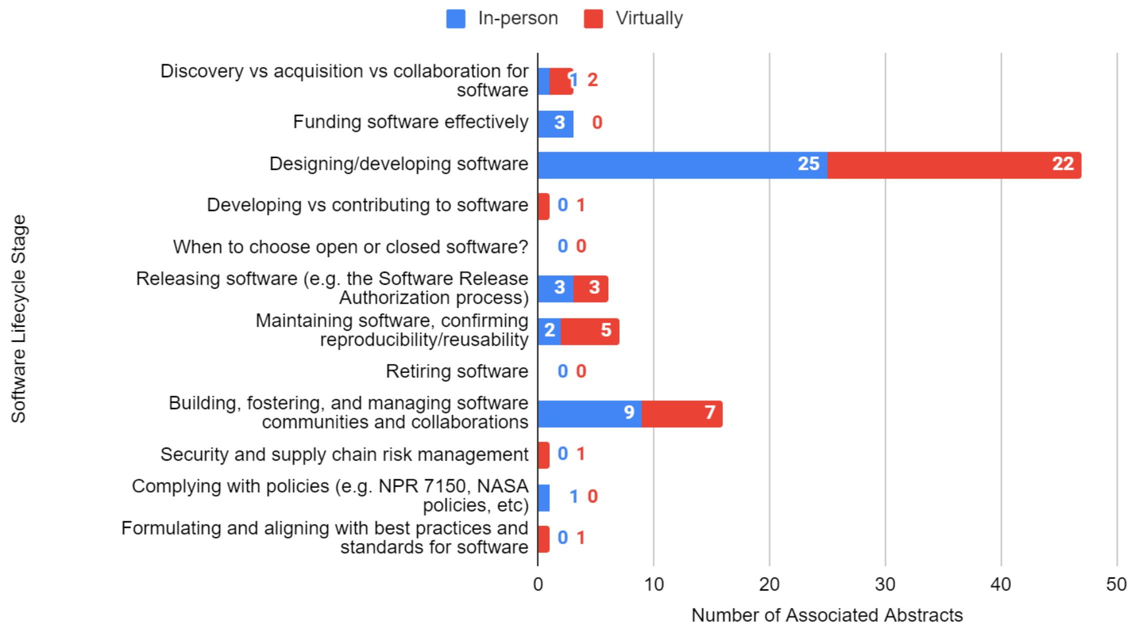

Figure A.2*: Distribution of the software lifecycle stages associated with abstract submissions. The twelve software lifecycle stages are listed on the left with the number associated abstracts indicated by the bars on the right. Blue bars indicate in-person attendance by the presenting author, and red bars indicate virtual attendance. The numbers at the end of each bar indicates the number of associated abstracts each software lifecycle stage received of each attendance type. This plot is available in the AbstractAnalysis.xlsx file in Ringuette et al. (2024).*