Experience and Outcomes Organizing a Hackathon in the Physical Sciences

Aaron Jezghani†
Georgia Institute of Technology
ajezghani3@gatech.edu

Jason Fry†
Eastern Kentucky University
jason.fry@eku.edu

ABSTRACT

Despite its growing importance in physical sciences, research computing with cluster resources remains difficult to access and sustain, especially in long-term, multi-institutional projects. Challenges include site-specific workflows, evolving software stacks, and rapid changes in hardware post-Generative AI. The Nab collaboration, conducting a precision test of the Standard Model at Oak Ridge National Laboratory, hosted a hackathon to address these issues. Over four half-days, 25 participants engaged in training and collaborative problem-solving across four priority areas, supported by mentors and structured sessions. Post-event surveys showed improved computational knowledge and strong interest in recurring events. This paper shares insights from organizing the hackathon and discusses scalable strategies for computational training in experimental research.

KEYWORDS

Data Science, Hackathon, Cluster Computing

1 INTRODUCTION

The physical sciences have historically garnered attention for their impressive experimental setups and apparatuses: from tools such as macromolecular crystallography used in the development of new drugs [9] to the stunning imagery captured by the James Webb Science Telescope, scientists and engineers have demonstrated incredible ingenuity in developing tools to explore all scales of the universe around us. In spite of the primary focus on the design and outcomes from the experiment itself, most efforts today include a heavy component of computational effort, whether through the use of computer simulations to understand the operation of the experiment, or in the analysis of the data taken along the way. As an example of the ubiquitousness of research computing as a critical component of science and engineering, Figure 1 shows the top 25 domains served by the NSF XSEDE and ACCESS programs from July 2021 through July 2025, with nearly all cycles going towards science and engineering [2]; specifically, the physical science domains consume roughly 75% of all cycles.

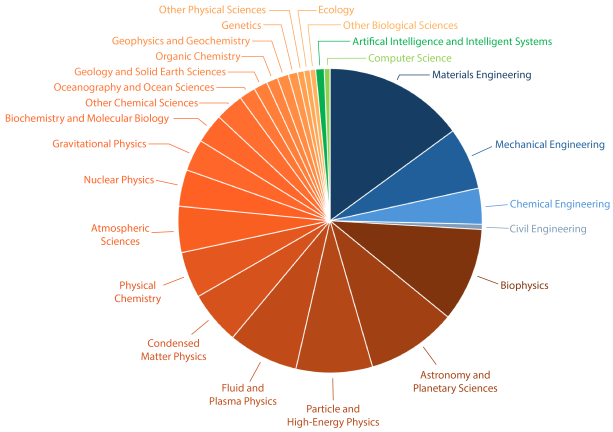


Figure 1: Top 25 domains by CPU hours across XSEDE/ACCESS systems from July 2021 through July 2025.

Note: Although engineering (blue) represents the greatest consumption of resources by a single domain, the physical sciences (orange) represent roughly 75% of all consumed cycles. Nuclear physics ranks 10th *in total consumption.*

In spite of the large volume of resources consumed by domain science users, they are not always the most efficient users in their work. Multiple reasons can explain this apparent contradiction, but one of the most likely culprits lies in the fact that as experimentalists in the domain sciences, the experimental apparatuses and the mastery of the theories in their specific domain are prioritized over the computational tools leveraged for research. For example, despite the domain of nuclear physics ranking 10th overall in CPU cycle consumption, a sample program from the National Nuclear Physics Summer School shows that only 3 of 80 hours in the program explore computational methods [11]; while service providers know the invaluable contribution of big computational resources for science and engineering, the systems are a tool to produce output for many users. Consequently, additional approaches are necessary to engage and onboard users in these domains.

University and national HPC centers offer training across a spectrum of topics, ranging from basic cluster orientation to more advanced topics like distributed and GPU programming. For example, Pittsburgh Supercomputing Center has tallied more than 24,000 attendees for its workshops on parallel, distributed, and big data computing, and feedback has been overwhelmingly positive [16]. However, the same paper acknowledges that the resources are presented in a simplified fashion, and the examples are presented for a general audience, so the specific translation to another computing

---

† On behalf of the Nab collaboration.

https://doi.org/10.22369/issn.2153-4136/17/1/9

resource or for a specific field of science may still present a barrier for the end user.

Another opportunity for researchers is to participate in OpenHackathons events, which are sponsored and supported by OpenACC members spanning industry, academia, and national labs [14]. Since 2014, the organization has hosted bootcamps and hackathons that pair researchers with expert mentors to optimize and scale their specific applications. While the outcomes from these events is exceptional, the commitment of mentors and resources, as well as the time required, limit the scalability of these hackathons. As such, the events have a limit on the number of teams, and the application process is competitive, which can present as a barrier to researchers still developing their workflows.

In summary, there exists a gap for many users across the physical sciences, which is problematic for the efficient use of research computing platforms, as they represent the largest consumer of resources. In this paper, we present an internally organized and hosted hacakthon to address this challenge head on, and summarize the outcomes and feedback so that it can serve as a model for others interested in taking a similar approach.

2 THE NAB EXPERIMENT

Proposed in 2007, the Nab experiment is currently taking production data at Oak Ridge National Laboratory is a precision test of the Standard Model of Particle Physics based on the decay of free neutrons [3, 7] . A beam of unpolarized neutrons enters a 7 m magnetic spectrometer, which measures the phase space of neutron β-decay through direct measurement of the resultant electron and proton in pixelated silicon detectors above and below the decay region. The goal for the experiment is to provide the most precise determination of two parameters in the equation describing the neutron decay rate, which can help resolve contention between physics models and observed results, as well as provide a sensitive probe for new physics.

Unlike complementary efforts such as high-energy experiments conducted using the Large Hadron Collider at CERN, where the primary challenge is generation of the greatest quantity of relativistic particle collisions that yield statistically significant exotic physics signals above background levels, high-precision experiments such as Nab emphasize a rigorous understanding of the various sources of experimental error to appropriately bound. For the purposes of Nab, the demand for computing is two-fold:

- simulations of billions of decay events, which currently require several hundred thousand CPU-hrs each, must be run many times to quantify the experimental uncertainties, and
- rigorous analysis of multiple petabytes of simulation and experimental data, which must be replayed repeatedly as analysis algorithms are continuously refined.

2.1 The Collaboration

As shown in Figure 2, the collaboration today includes 51 active members from 16 institutions across 4 countries. To date, 18 PhD students have published dissertations as part of the Nab effort, with another 16 currently conducting research. Additionally, a number of high school and undergraduate students have contributed to the collaboration, as well as several post-doctoral researchers, academic faculty, and staff scientists. The effort of the collaboration leverages a hybrid model, with the main experimental work conducted at the lab and simulation, analysis, remote shift work, and small-scale studies occurring at collaborator home institutions.

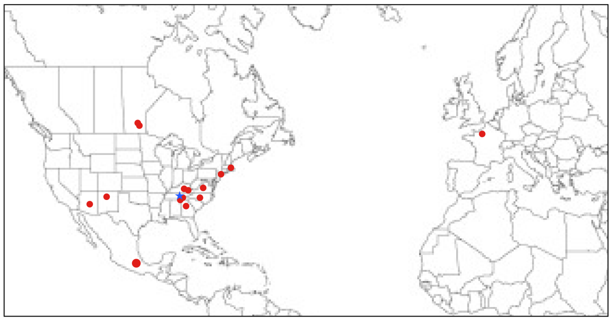


Figure 2: The Nab collaboration.

Note: The Nab collaboration is presently comprised of researchers from 16 institutions across 4 countries (red dots), and the experiment takes place at Oak Ridge National Laboratory (blue star).

Computing resources for Nab are distributed across various institutions. In addition to near-line analysis servers with GPUs and the physics division cluster, collaborating universities provide storage and computational cycles to support simulation and analysis efforts. In particular, computing effort on the Phoenix cluster at Georgia Tech (GT) is enabled by the institute-supported tier provided to all faculty as well as an Institute for Data Science and Engineering (IDEaS) allocation for CPU and GPU cycles plus 250TB of local storage for batch processing of the experimental data [8, 10]; cluster usage at GT since 2024 has totaled 1.6M CPU-hours and 6.2k GPU-hours.

2.2 Software Stack

Like many nuclear and particle physics experiments, the Nab software stack was built around CERN's analysis and simulation packages, `ROOT` and `Geant4`, respectively [1, 5]. With decades of software engineering baked into both applications, `ROOT` and `Geant4` both seek to address the challenges of efficiency, providing numerous template classes and interfaces to libraries for accelerated algorithms, efficient data structures, and routines for parallel and distributed processing. However, these capabilities present a double-edged sword, as building with the necessary dependencies is a non-trivial effort. In light of these challenges, Docker-based images and Spack installation instructions are provided; by default, however, these solutions are provided as separate containers, and enabling platform-specific plugins can still prove challenging. Additionally, the initial installation of `Geant4` provides only the libraries necessary to compile the Nab simulation, which itself introduces additional dependencies and complexities.

The Nab collaboration has developed an additional suite of tools for simulation and analysis for use in conjunction with the aforementioned packages:

- `Delta-Rice`, an HDF5 plug-in for compression of digitized waveform data, allows for extremely high-throughput compression and decompression [13],

- `nabPy`, a Python-based analysis stack with routines for waveform processing and physics analysis [12], and
- `NESSE`, a Python utility for detector response simulation [15].

Given the challenges to prepare the full swath of simulation and analysis software, an Apptainer container was developed to provide a portable, reproducible environment with the requisite software for simulation and analysis efforts for Nab. Starting from a base Ubuntu 20.04 image, `Geant4` with support for `ROOT` data structures, as well as a compiled version of the Nab Simulation, was built and made available to collaborators. However, the use of the container was not well-documented, and the additional suite of Nab-specific packages was not included in the build, rendering it limited for use beyond a basic particle simulation. See Section 4.1 for improvements of the container made during the hackathon.

2.3 Motivations for Targeted Training

Although not formally tracked, internal communications have long indicated challenges in using cluster resources and software tools by various collaborators. Despite explicit requests for GT cluster access by 46 individuals, Figure 3 shows that only 21 users have accumulated any time since January 1, 2024. Furthermore, 85% of allocated CPU-hrs can be traced to just 2 of those users, while 70% of allocated GPU-hrs are the work of a single individual. Monthly utilization summaries collectively show utilization efficiency across the majority of users is less than 10% of theoretical limits.

As will be described in Section 3.1, participants were provided an opportunity for feedback following the collaboration hackathon. In particular, one question asked respondents to rank their skills in various topics before participation in the event. Figure 4 shows the results from the 15 individuals who provided feedback. Notably, skills such as Python scripting or cluster access via remote command-line interface or browser gateways ranked highest, while confidence in the Nab software stack, and particularly emergent GPU and AI workflows, were demonstrably lower.

In response to the increasing urgency for more effort and results from both simulation and analysis, the collaboration was encouraged by sponsoring agencies to address gaps in capability as expeditiously as possible. As such, Nab leadership decided to organize a dedicated event to facilitate knowledge sharing, especially to newer members, as well as expend effort on outstanding tasks.

3 THE NAB HACKATHON: “NABATHON”

Following the efficacy of the OpenHackathons format and the successes of participating teams, the Nab hackathon was fashioned similarly with a few modifications. First, the event was designed to be inclusive of all skill-levels; beyond development and optimization of specific applications, the event was meant to introduce new collaborators to the available tools and reinforce software development and maintenance practices across all . Second, the length of the event, both in number of days as well as the length of each day, had to be reduced due to other scheduled activities. Lastly, the results were inclusive of physics outcomes as well as improvements to specific software packages and computational workflows.

3.1 Organization

Organization of the hackathon began approximately six weeks in advance of the event. Based on the experimental schedule, data-taking campaigns, and outstanding tasks, senior personnel from the collaboration identified simulation and analysis priorities to assign to subteams for the event. For each task, one or more mentors with expertise in the physics or software tools were identified to shepherd the subteam effort; additionally, one mentor with general cluster knowledge provided guidance across all subteams as relevant. This list of tasks was then circulated to the collaboration as a Google spreadsheet, asking members to identify subteams to which they were conducting synergistic activity and could contribute.

Prior to the start of the hackathon, portions of weekly collaboration analysis calls were dedicated to discussion of specific details of the hackathon tasklist. Logistics for the hybrid event, participant expectations, and cluster access were also addressed during this period. Zoom meeting invitations were prepared for remote attendees, while co-located attendees were able to reserve a lab conference room for the event. Additionally, sample simulation macros and datasets were prepared to provide a baseline for effort during the hackathon.

The event was scheduled as four half-days over a period of one week, with an introductory Day 0 on Thursday for cluster onboarding and final subteam selection, and three hacking days starting the following Monday. Participants were encouraged to engage asynchronously before the first hacking day to familiarize themselves with the cluster resources and subteam tasks. In addition to existing Phoenix cluster access, participants were given temporary accounts on the instructional cluster [4, 6], which provided the same software stack and hardware architectures, but with more conservative resource limits to reduce queue wait times for teaching purposes.

Day 0 was scheduled from 9 am - 1 pm and content was crowd-sourced from existing GT training, course content, and publicly-available community content; hosted on the Nab collaboration internal git repository with links for recording of day 0 content. The git repository also housed directories for training and homework for participants. The scrum and final presentations were accessed through a in shared google drive with templates for each that included skeleton structure for content and recommended presentation time.

At the start of Days 1 and 2, teams presented progress, goals, and obstacles as part of morning scrums. Afterward, the teams hacked from roughly 9:20 am - 1 pm. At the end of Day 3, teams gave a longer presentation reporting on their final results and status of assigned activity. Additionally, an anonymous survey was disseminated via Microsoft Forms to participants to provide feedback on the efficacy of the event, including both the organization and structure as well as the technical components.

3.2 Subteam Efforts

In pre-planning for the hackathon during collaboration analysis calls, three main efforts were chosen and subteams were assigned. The three subteams were simulation and pipeline, a-fitting, and magnetometry. A spreadsheet was sent around for everyone participating to sign up for the various groups and subgroups and expert mentors were identified for each subgroup. The simulation and

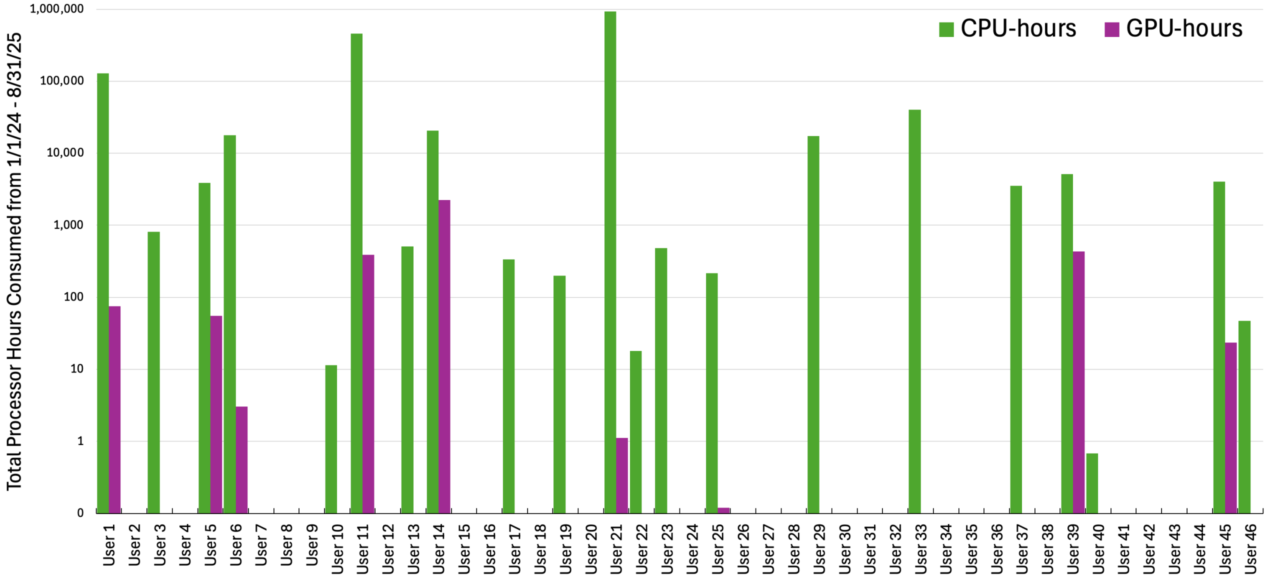


Figure 3: CPU-hrs and GPU-hrs consumed by Nab collaborators on the Phoenix cluster.

Note: Of the 46 accounts that were requested, only 21 have any trackable activity on the cluster, with most effort dominated by just a few users.

pipeline team focused on the `geant4` and `NESSE` simulations with the goal of creating a pipeline between the two. The magnetometry group focused on the analysis on the mapping of the complex magnetic fields in the Nab spectrometer. This included traditional analysis in python and C++ as well as a neural network approach. The a-fitting group's focus was on developing new algorithms to fit the complex 2D phase space that Nab measures as well as further developing existing algorithms.

4 OUTCOMES AND FEEDBACK

As described in section 3.1, a survey was created and sent to all participants of the hackathon with 15 respondents to assess pre-event and post-event proficiency. Overall, the survey showed that the participants found the event useful and the collaboration agreed hackathons would be useful going forward on as annual or semiannual events.

4.1 Technical Accomplishments

Figures 4 and 5 show the participants proficiency in various computing areas before and after the hackathon. We can see that the mean reported values of the all the areas increased and the Nab-based areas increased significantly. Improvements were readily apparent in the Nab software stack, the use of software containers, and the development of more effective workflows with Slurm. However, little improvement was demonstrated for GPU and AI activity, which can be attributed to a lack of organized effort and focus on those two particular topics.

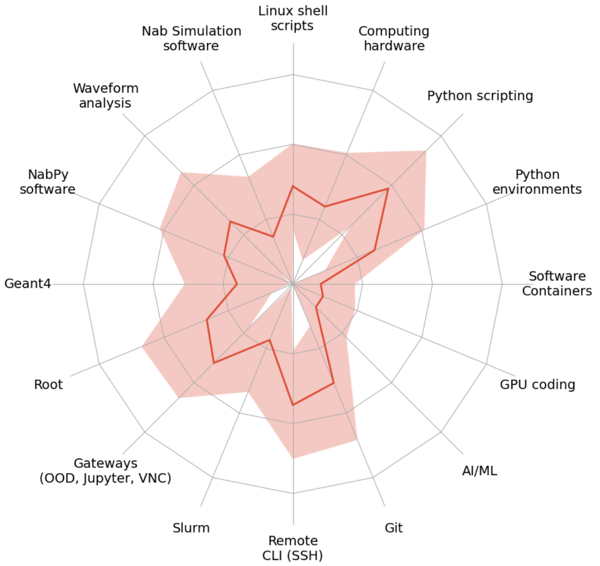


Figure 4: Self-reported proficiency by participants prior to the hackathon event.

Note: Radial distance represents participant expertise in a particular topic, with the line and band representing the mean reported value plus or minus one standard deviation, respectively. Despite moderately high confidence in fundamental skills such as Python scripting, version control, and basic command line interface, respondents did not express strong confidence in neither the Nab software stack nor novel workflows using GPUs or AI.

While the survey results indicated pre-event and post-event proficiency, the subteams presented physics achievements in the final slide decks. In the simulation and pipeline group, the team was able to marry two previously separate software packages to form a pipeline between the two. This was one of the main objectives of the hackathon and the team produced documentation of

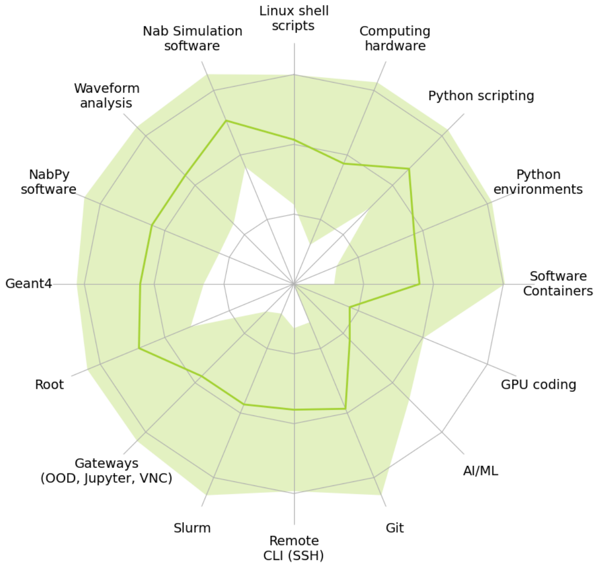
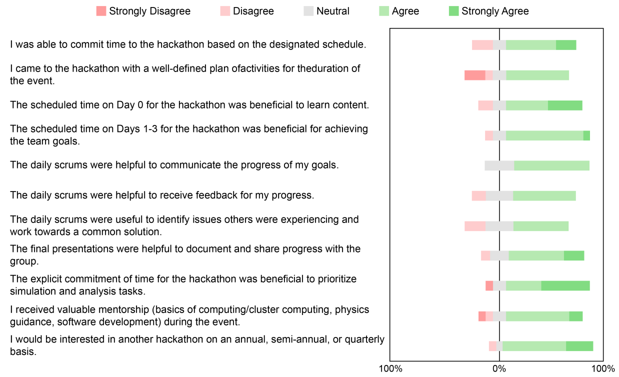


Figure 5: Self-reported proficiency by participants following the hackathon event.

Note: Increasing distance from the center represents increasing expertise. The line represents the mean for the data while the band shows one standard deviation about the mean. Respondents reported on average significant improvements in collaboration software, Slurm resource allocation, and software containers; however, the wide band reflects a broad range of confidence from participants.



I was able to commit time to the hackathon based on the designated schedule.

I came to the hackathon with a well-defined plan ofactivities for theduration of the event.

The scheduled time on Day 0 for the hackathon was beneficial to learn content.

The scheduled time on Days 1-3 for the hackathon was beneficial for achieving the team goals.

The daily scrums were helpful to communicate the progress of my goals.

The daily scrums were helpful to receive feedback for my progress.

The daily scrums were useful to identify issues others were experiencing and work towards a common solution.

The final presentations were helpful to document and share progress with the group.

The explicit commitment of time for the hackathon was beneficial to prioritize simulation and analysis tasks.

I received valuable mentorship (basics of computing/cluster computing, physics guidance, software development) during the event.

I would be interested in another hackathon on an annual, semi-annual, or quarterly basis.

Figure 6: Participant survey results.

Note: According to survey respondents, the overall perception of the event was quite positive. Notable detractors include a lack of explicit effort, the efficacy of the daily scrums, and the quality of mentorship, but in theory these could be addressed with better planning. Despite this, respondents agreed that recurring events would be invaluable for the collaboration.

the changes implemented. The group also produced an 10^8 event simulation in 18 hours using 500 CPUs through slurm. Now the collaboration has a set of slurm templates for submitting large simulation and analysis jobs. In the magnetometry group, the team used the dedicated time to devote to developing analysis algorithms of the magnetic field data. This analysis produced a data-driven approach to making the next measurements of the magnetic field. The a-fitting group focused on fitting algorithms in python (such as `lmfit`) and compared speed and accuracy of different fitting infrastructure as well as documenting all the existing fitting algorithms of the collaboration.

4.2 Reflection on the Hackathon

The survey also included a question about what computational resources were utilized during the hackathon, which allowed multiple responses: personal device, GT instructional compute cluster, and GT Phoenix compute cluster. Out of the 15 participants, 5 of the participants worked exclusively on their personal devices for the event. Of the remaining 10 that took advantage of GT compute clusters, 5 used both Phoenix and the instructional cluster, 3 used only the instructional, and 2 used only Phoenix cluster.

In the open-ended question asking for any additional feedback, multiple respondents indicated the need for a better job organizing subteams and tasks, especially the explicit definition of deliverables at the end of the hackathon. Another respondent also suggested that the Day 0 content could be more impactful for participants if organized as a hands-on workshop rather than presented via slide decks. Yet another person suggested that increased in-person participation would be beneficial, although logistically, this is perhaps the most challenging component to address given the distribution of collaboration members. Nonetheless, 13 of the 15 respondents concurred that there should be another hackathon event within the next 6 to 12 months, with commentary that the event was particularly effective for advancing their research efforts within the collaboration.

5 CONCLUSION

Despite the prominence of users from the physical sciences on university and national research computing platforms, many experimentalists de-prioritize computational training and knowledge for attention to hands-on effort and domain expertise. Following a popular model for project-oriented hackathons, the Nab collaboration recently organized an event to streamline new student onboarding, expedite software development, and improve resource utilization efficiency. While multiple participants commented that the hackathon organization could be improved, with more detailed task identification and subteam effort, participant responses were largely positively to the event. Self-reported technical proficiency in cluster utilization, hardware, and software improved in many areas, while the general sentiment was favorable for recurring events to continue the successes following the 4 day event.

We feel that similarly self-organized events can provide a scalable and reproducible mechanism to enhance the computational abilities for other experimentalists who consume large quantities of computer resources. In addition to collaborations organizing such events to drive their own efforts, disjoint groups with common interests, particularly across multiple universities, can collectively arrange for their own hackathons to maximize the return on investment, without the barrier to entry they may encounter in other events. Additionally, we can imagine that this may reduce the overall support burden for cluster administrators, as researchers are exposed to the tools for the specific problems

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation (NSF) Contract PHY-2412846. This research was supported through research cyberinfrastructure resources and services provided by the Partnership for an Advanced Computing Environment (PACE) at the Georgia Institute of Technology, Atlanta, Georgia, USA.

REFERENCES

[1] 2003. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506, 3 (2003), 250–303. https://doi.org/10.1016/S0168-9002(03)01368-8

[2] ACCESS CI Database. 2025. ACCESS-CI XDMoD: CPU Hours: Total: by Field of Science: 2021-07-01 to 2025-07-31. https://xdmod.access-ci.org/#tg_usage?node=statistic&realm=Jobs&group_by=fieldofscience&statistic=total_cpu_hours

[3] Ricardo Alarcon, Septimiu Balascuta, Andreas Klein, W. Scott Wilburn, Michael T. Gericke, John R. Calarco, F. William Hersman, Albert Young, James D. Bowman, T. Vince Cianciolo, Seppo I. Penttila, Krzysztof P. Rykaczewski, Glenn R. Young, Vladimir Gudkov, Geoffrey L. Greene, Robert K. Grzywacz, L. Peter Alonzi, Stefan Baeßler, Maksim A. Bychkov, Emil Frlez, Anthony Palladino, and Dinko Pocanic. 2007. Precise Measurement of the Neutron Beta Decay Parameters "a" and "b". http://nab.phys.virginia.edu/nab_proposal.pdf

[4] Mehmet Belgin, Trever C. Nightingale, David A. Mercer, Fang Cherry Liu, Peter Wan, Andre C. McNeill, Ruben Lara, Paul Manno, and Neil Bright. 2018. ICE: A Federated Instructional Cluster Environment for Georgia Tech. In *Proceedings of the Practice and Experience on Advanced Research Computing: Seamless Creativity (PEARC '18)*. Association for Computing Machinery, New York, NY, USA, Article 16, 7 pages. https://doi.org/10.1145/3219104.3219112

[5] Rene Brun and Fons Rademakers. 1997. ROOT — An object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 389, 1 (1997), 81–86. https://doi.org/10.1016/S0168-9002(97)00048-X New Computing Techniques in Physics Research V.

[6] J. Eric Coulter, Michael D. Weiner, Aaron Jezghani, Matthew Guidry, Ruben Lara, Fang (Cherry) Liu, Allan Metts, Ronald Rahaman, Kenneth Suda, Peter Wan, Gregory Willcox, Deirdre Womack, and Dan (Ann) Zhou. 2023. ICE 2.0: Restructuring and Growing an Instructional HPC Cluster. In *Proceedings of the SC '23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis (SC-W '23)*. Association for Computing Machinery, New York, NY, USA, 591–597. https://doi.org/10.1145/3624062.3624131

[7] Jason Fry, Ricardo Alarcon, Stefan Baeßler, Septimiu Balascuta, Libertad Barrón Palos, Thomas Bailey, Kevin Bass, Noah Birge, Alexander Blose, Dennis Borissenko, James D. Bowman, Leah J. Broussard, Albert T. Bryant, James Byrne, John R. Calarco, Jimmy Caylor, K. Chang, Timothy Chupp, T. Vince Cianciolo, Christopher Crawford, Xinjian Ding, Margaret Doyle, Wenjiang Fan, W. Farrar, Nadia Fomin, Emil Frlež, Michael T. Gericke, Michelle Gervais, Ferenc Glück, Geoffrey L. Greene, Robert K. Grzywacz, Vladimir Gudkov, Joshua Hamblen, Christopher Hayes, Chelsea Hendrus, Takeyasu Ito, Aaron Jezghani, Huangxing Li, Mark Makela, Nicholas Macsai, Juliette Mammei, Russell Mammei, Madelyn Martinez, David G. Matthews, Mark McCrea, Patrick McGaughey, C. David McLaughlin, Paul Mueller, D. van Petten, Seppo I. Penttilä, David E. Perryman, Rüdiger Picker, Josh Pierce, Dinko Počanić, Yu Qian, John Ramsey, Glenn Randall, Grant Riley, Krzysztof P. Rykaczewski, Americo Salas-Bacci, Sepehr Samiei, E. Mae Scott, Thomas Shelton, Skye K. Sjue, Alexander Smith, Erick Smith, Eric Stevens, Jonathan Wexler, Ryan Whitehead, W. Scott Wilburn, Albert Young, and Bryan Zeck. 2019. The Nab experiment: A precision measurement of unpolarized neutron beta decay. *EPJ Web Conf.* 219 (2019), 04002. https://doi.org/10.1051/epjconf/201921904002

[8] IDEaS. 2025. Institute for Data Engineering and Science. https://research.gatech.edu/data

[9] Mariusz Jaskolski, Zbigniew Dauter, and Alexander Wlodawer. 2014. A brief history of macromolecular crystallography, illustrated by a family tree and its Nobel fruits. *The FEBS Journal* 281, 18 (2014), 3985–4009. https://doi.org/10.1111/febs.12796 arXiv:https://febs.onlinelibrary.wiley.com/doi/pdf/10.1111/febs.12796

[10] Aaron Jezghani, Semir Sarajlic, Michael Brandon, Neil Bright, Mehmet Belgin, Gergory Beyer, Christopher Blanton, Pam Buffington, J. Eric Coulter, Ruben Lara, Lew Lefton, David Leonard, Fang Cherry Liu, Kevin Manalo, Paul Manno, Craig Moseley, Trever Nightingale, N. Bray Bonner, Ronald Rahaman, Christopher Stone, Kenneth J. Suda, Peter Wan, Michael D. Weiner, Deirdre Womack, Nuyun Zhang, and Dan Zhou. 2022. Phoenix: The Revival of Research Computing and the Launch of the New Cost Model at Georgia Tech. In *Practice and Experience in Advanced Research Computing 2022: Revolutionary: Computing, Connections, You (PEARC '22)*. Association for Computing Machinery, New York, NY, USA, Article 13, 9 pages. https://doi.org/10.1145/3491418.3530767

[11] Jinfeng Liao, Walter Pettus, W. Michael Snow, and Scott Wissink. 2024. National Nuclear Physics Summer School. https://ceem.indiana.edu/events/nnpss/program.html

[12] David G. Mathews. 2025. pyNab Repo. https://gitlab.com/NabExperiment/pyNab

[13] David G. Mathews, Christopher B. Crawford, Stefan F. Baessler, Noah Birge, Leah J. Broussard, Francisco M. Gonzalez, Leendert Hayen, Aaron Jezghani, Huangxing Li, Russell R. Mammei, August Mendelsohn, Glenn Randall, Grant V. Riley, and Danielle C. Schaper. 2024. Delta-Rice: A HDF5 Compression Plugin optimized for Digitized Detector Data. *Journal of Open Source Software* 9, 98 (2024), 6598. https://doi.org/10.21105/joss.06598

[14] OpenHackathons. 2025. OpenHackathons Webpage. https://www.openhackathons.org/s/

[15] Robert J. Taylor. 2025. Nab Event Shape Simulation Effort (NESSE) Repo. https://github.com/rjtayl/nesse

[16] John Urbanic, Thomas Maiden, and Valerie Rossi. 2024. Evaluating the Wide Area Classroom After 24,000 HPC Students. arXiv:physics.ed-ph/2402.03343 https://arxiv.org/abs/2402.03343