March 2025

Volume 16 Issue 1

# JoCSE

Journal Of Computational Science Education

Promoting the Use of Computational Science Through Education

ISSN 2153-4136 (online)

# Contents

***Volume 16, Issue 1***

# Introduction to Volume 16, Issue 1

David Joiner, Editor
Kean University
djoiner@kean.edu

This issue of the Journal of Computational Science Education highlights contributions from the Best Practices in High Performance Computing Training and Education (BPHTE) workshop at SC24. The BPHTE 2024 program demonstrates significant growth in computational science education, expanding both its scope and scale. The selected papers showcase a wide range of advancements in high-performance computing (HPC) education, workforce development, and training methodologies for researchers, educators, and students.

The featured articles in this issue include:

(1) **Barrios et al.** discuss strategies for integrating diverse stakeholders into HPC ecosystems through transdisciplinary collaboration and curriculum development.
(2) **Mehringer et al.** present updates on the HPC-ED project, a community-driven CyberTraining catalog designed to improve resource discovery and sharing.
(3) **Filinger et al.** present a new approach to structuring learning pathways for HPC education, capturing training needs and progression.
(4) **Purwanto et al.** describe T3-CIDERS, a train-the-trainer program focused on cybersecurity research and HPC-based cyberinfrastructure training.
(5) **Suleman et al.** analyze the usability of HPC services at the University of Cincinnati, emphasizing accessibility and workforce development.
(6) **García Mesa and Speyer** explore institutional strategies for supporting large language models (LLMs) in research and education.
(7) **Reid et al.** provide updates on HPC Carpentry and its formalization as an official Carpentries lesson program.
(8) **Wang and Ponce** introduce a modular, hands-on approach to teaching computer networking through captive portal technologies.
(9) **Johnston et al.** document the evolution of a student-led HPC Special Interest Group (SIG) and its impact on training and Student Cluster Competition success.
(10) **Finch et al.** describe a computational skills training program for undergraduate researchers in molecular engineering, focusing on Python, HPC, and machine learning.
(11) **Tsoukalas** presents an automated batch scheduler feedback system for HPC users, providing real-time job monitoring and performance optimization.

These contributions reflect the dynamic and evolving landscape of HPC education, addressing both technical skill development and the broader challenges of training and engagement within the HPC community.

We encourage you to submit your work to the Journal of Computational Science Education. Computational science education continues to play a critical role in equipping students and professionals with the skills needed to tackle complex scientific and engineering problems. Whether you are developing new instructional materials, evaluating educational initiatives, or exploring innovative teaching methodologies, your contributions can help advance the field. We also encourage submissions from students describing their experience and results implementing computational science solutions. Student submissions should include reflections from the student and mentor on how those problems could be replicated for student training.

Additionally, if you have expertise in computational science education, consider volunteering as a reviewer to support our peer review process. Together, we can strengthen the HPC education community and foster new opportunities for learning and discovery.

Sincerely, Dave Joiner

# Integration of Actors in HPC Ecosystems: Transdisciplinarity, Interdisciplinarity and Interaction

Carlos J. Barrios H.
SC3UIS, Universidad Industrial de Santander
cbarrios@uis.edu.co

Sergio A. Gelvez C.
SC3UIS, Universidad Industrial de Santander
sagelcor@uis.edu.co

Luis A. Torres N.
SC3UIS, Universidad Industrial de Santander
luatorni@uis.edu.co

## ABSTRACT

Interest in High-Performance Computing (HPC) has surged, driven by the demand for skills to utilize advanced computing methods. These methods include managing vast amounts of data, implementing complex algorithms, and developing Artificial Intelligence (AI) applications. HPC ecosystems play a crucial role in tackling intricate scientific and engineering problems. However, integrating various stakeholders demands a deep understanding of collaboration and innovation within HPC settings. This proposal explores how different stakeholders—from students across diverse fields to scientists and policymakers—have been integrated at various levels. This integration is facilitated by introducing new formal courses, incorporating relevant topics into existing curricula, and other related activities.

## KEYWORDS

Education, HPC, Training, Best Practices

## 1 INTRODUCTION

The enhancement of High Performance Computing (HPC) skills is vital for multiple professional sectors, extending beyond advanced scientific and technological domains to encompass a diverse array of fields and applications. With the ongoing expansion of artificial intelligence (AI) applications, acquiring knowledge and competencies in parallel computing across various levels and disciplines is imperative, whilst promoting interactivity among them to enhance conception, development, and utilization. Institutions and communities pursue various initiatives [1] and recommendations tailored to specialized communities and interest groups [2], involving individuals in transformative experiences focused in implementations addressed to build knowledge or innovations [1, 3].

These individuals can be categorized in several ways, primarily as participants interacting with High-Performance Computing (HPC) resources and possessing distinct requirements [12]. They may be classified into three distinct groups: HPC system engineers, who integrate, monitor, tune, optimize, and evaluate the performance of HPC hardware and software; these professionals are also referred to as DevOps professionals [2]. The second group consists of HPC software engineers, who are responsible for developing parallel programs, debugging and analyzing application performance, and optimizing HPC software cycles. The third group encompasses computer scientists, who, while classified as users, require specialized skills for HPC, such as coding, executing, and efficiently deploying parallel programs. These three groups engage in continuous interaction and are required to share common knowledge. It is pertinent to note, however, that these categories are not uniform development spaces; rather, they can vary in depth based on individual interests and levels of interaction. On the other hand, it is important to understand that the actors are not isolated but are in constant interaction and that these actors, at their base, can be part of other disciplines (and not only at the user level), which implies transdisciplinarity and interdisciplinarity [4, 7].

This proposal outlines an experience centered on integrating actors within the HPC ecosystem, considering various factors: interaction spaces, transdisciplinary and interdisciplinary approaches, competency levels, and knowledge depth. Over more than a decade, this experience has been gathered through diverse activities, including the Supercomputing and Distributed Systems Camping School (SCCAMP) [3], specialized certifications with the technology industry, mainly conducting workshops and seminars with NVIDIA under the NVIDIA Deep Learning Institute[4] and formal courses and certification training at Universidad Industrial de Santander (from the Spanish acronym UIS, Universidad Industrial de Santander) [5], facilitated by the High Performance and Scientific Computing Center (SC3UIS, from the Spanish acronym of Supercomputación y Cálculo Científico de la Universidad Industrial de Santander)[6]. This effort has led to the identification and development of effective practices in training ecosystem participants, significantly impacting the advancement of computing in both the academic sector and related communities.

## 2 APPLYING KEY CONCEPTS

In our proposal, we outlined two significant concepts: transdisciplinarity and interdisciplinarity. Transdisciplinarity encompasses integrating knowledge from various fields and stakeholders, underscoring the importance of collaboration in addressing complex issues. Within High-Performance Computing (HPC), this approach involves computational scientists, domain experts, and end-users.

[1]For example, the HPC Forum https://www.hpcuserforum.com/
[2]Such as the HPC Certification Forum https://www.hpc-certification.org/, and the Education Chapter on ACM Special Interest Group in HPC https://sighpceducation.hosting.acm.org/

[3]More about SCCAMP in https://www.sc-camp.org
[4]More information about NVIDIA DLI in https://www.nvidia.com/en-us/training/
[5]For more information about the UIS: https://www.uis.edu.co
[6]More information about SC3UIS in: http://www.sc3.uis.edu.co


https://doi.org/10.22369/issn.2153-4136/16/1/1

Interdisciplinarity, conversely, merges knowledge from diverse disciplines to tackle specific problems, fostering collaboration to develop innovative computational techniques and applications within HPC ecosystems. We wish to emphasize the critical significance of interactions both within and across disciplines, as previously explained, which transcends a mere multidisciplinary approach. Consequently, interactions are paramount, as this article will elaborate on.

## 2.1 Implementing Transdiciplinarity

In order to implement transdisciplinarity, as defined in the preceding discussion, the establishment of specialized courses tailored to distinct groups of undergraduate and graduate programs was proposed. Initially, we developed the following undergraduate courses: High Performance and Scientific Computing, which is available to all disciplines; Introduction to Parallel Computing, designed specifically for students in the Systems and Informatics Engineering program; and a high-performance computing course that was first offered to postgraduate students of the university's School of Systems Engineering and Computer Science. After a period of three years, this course was subsequently made available to all postgraduate programs at the university, including both master's and doctoral courses. The courses commence simultaneously, and while there are dedicated sessions for each course, they also incorporate shared challenges and activities to ensure interaction and foster a unified perspective on complex problems across diverse disciplines and levels.

The different activities carried out during these years have been diverse: for example, a collaborative mid-course project developed in specific sessions with students from the (3) different courses to address an evaluated challenge in distributed memory programming with MPI, or the final project of the course, developed in slightly larger teams including students of different levels.

Over time, we identified consequential aspects to implement this transdisciplinarity: inclusion of different actors (not necessarily direct actors of the programs in science and engineering of the University or the academic community), collaborative learning environments (i.e., platforms or collaborative *fun* activities) [11], vision on transformation knowledge and a systemic approach [5]. With these aspects, the courses were evolving towards popularization activities in which the students participated: for example, hackathons, activities framed within science and technology festivals with the participation of high school students or inhabitants of the neighborhoods surrounding the university, and students of human and health sciences disciplines.

Figure 1 shows the participants in different activities: on the top left, microbiology students in the data analytics course with R; on the bottom left, participants in the smart vehicle challenge based on NVIDIA Jetson; and on the right, children from the supercomputing nursery in schools and neighborhoods surrounding the university.

The transdisciplinary initiative fostered a strong interdisciplinarity. In the next section, we will outline how this prompted the creation of specialized courses for postgraduate students. These courses focus on managing large-scale data within microbiology and bioinformatics, utilizing parallel processing. Additionally, we will discuss proposed courses in physics and chemical engineering, as well as developments in digital humanities [9].

**Figure 1: Participants in transdisciplinary activities**

## 2.2 Developing Interdisciplinarity

In our approach, interdisciplinarity refers to an educational strategy that integrates insights and methods from multiple disciplines to address complex problems or questions that a single field cannot adequately solve. This approach is increasingly recognized as essential in academic research and practical applications, as it allows for a more comprehensive understanding of nuanced issues.

Clearly, in the academic university environment, there is more familiarity with interdisciplinarity, and that is why we develop it further in this section. Although there are clearly aspects that can be considered equally transdisciplinary, the focus given allows us to identify the important activities and characteristics:

- Complex Problem Solving: many challenges require interdisciplinary research and actions to synthesize knowledge from various fields. For this reason, different specific courses and seminars were created addressed to specific domains. In the case of the computer science students (undergraduate or postgraduate), their technical skills were directed towards understanding specific scientific formalisms for different areas of interest (in physics, for example, with quantum computing).
- Innovate Solutions: by merging traditional educational notions, interdisciplinary work can lead to innovative solutions that are more effective than those derived from a single discipline. This is particularly relevant in business careers and humanities. And for the creation of entrepreneurial spirits in students.
- Cognitive Development: interdisciplinary instruction enhances cognitive abilities by encouraging them to integrate concepts from different disciplines. This fosters skills such as perspective-taking, which is crucial for understanding diverse viewpoints and approaches to problem-solving. This aspect is closely related to the resolution of complex problems.

- Multidisciplinary Approach: this aspect involves the integration and collaboration of knowledge, skills, and methodologies from multiple academic disciplines. In the university, it is particularly valuable to improve soft skills such as communication and the growth of deep scientific knowledge.
- Adaptative Knowledge: The adaptive knowledge management process systematically and intentionally collects and selects information and connects it to the right people at the right time to increase the likelihood that students, researchers or decision-makers see, understand, and use the knowledge [6]. This is an essential pillar for multi-level transdisciplinary activities.
- Collaborative Skills: To achieve collaboration, for example, it is necessary to establish a common language, given by good practices, aspects related to paradigms in programming languages, and, of course, modeling and simulation (as well as systemic approaches).

Several strategies may be utilized to identify these characteristics. For instance, phased integration—a structured methodology for interdisciplinary research—can encompass phases such as comparing disciplines, comprehending their methodologies, and engaging in cross-disciplinary thinking. This approach fosters the establishment of a cohesive framework for assimilating diverse insights. Consequently, it facilitated the development and design of the curriculum. The educational programs at our university ought to be structured to promote collaboration among various disciplines. This may involve the creation of modules that necessitate students to leverage knowledge from multiple fields in order to address real-world challenges. In light of this, the topics covered in formal courses within the domains of science and engineering include, but are not restricted to, the utilization of tools and techniques that incorporate high-performance computing support, thereby enriching the entire ecosystem. Such was the case in courses such as Fluid Dynamics for Mechanical Engineering, Numerical Methods in Engineering, Computer Architecture for Computer Science, and Data Analytics for Human Sciences.

The table 1 below illustrates examples of different topics and subjects in courses related to specific degrees at UIS, at both undergraduate and postgraduate levels. In the first column, example topics shared across various courses are listed in the second column, alongside their corresponding academic programs in the third column. For instance, the topic of Parallel Programming Paradigms appears in two courses with different focuses: Parallel Programming, which is available for all degrees (including both undergraduate and graduate), and Parallel Computing, offered only for the undergraduate program in Systems Engineering (Sys. Eng.). Within this topic, we explore shared memory programming (using OpenMP), distributed memory programming (with MPI), and hybrid programming with directives (such as OpenACC or CUDA). Regarding the second topic, Parallel and Distributed Computation, we find that it is taught in Numerical Analysis for the Sys. Eng. program, as well as in Numerical Methods in Chemical Engineering (Chem. Eng.) and Fluid Dynamics in Mechanical Engineering. Another intriguing topic worth noting involves GPGPUs, which is covered in both postgraduate and undergraduate courses, such as Computer Architecture and Parallel Computing for Sys. Eng., HPC for the MSc. in Computer Science (C.S.), and Advanced Computing Architecture for the MSc. in Electronic Engineering.

**Table 1: Topics, Courses and Involved Programs**

| Topics | Course | Career |
|---|---|---|
| Parallel Programming Paradigms | Parallel Programming<br>Parallel Computing | All<br>Sys. Eng. |
| Parallel and Distributed Computation | Numerical Analysis<br>Numerical Methods<br>Fluid Dynamics | Sys. Eng.<br>Chem. Eng.<br>Mec. Eng. |
| Data Reduction and Visualization | Data Analytics<br>Computing Tools<br>Data Analysis | Biology<br>Social Work<br>Medicine |
| Debugging, Profiling, and Monitoring | Algorithms and HPC<br>Parallel Computing<br>Parallel Programming<br>Data Analytics | Physics<br>Sys. Eng.<br>All<br>Biology |
| GPGPUs Systems | Computer Architecture<br>Parallel Computing<br>HPC<br>Advanced Computer Architecture | Sys. Eng.<br>Sys. Eng.<br>MSc. in C. S.<br>MSc. in Electronic Enginnering |

These courses are integrated, featuring a collaboration between professors and engineers from SC3UIS. In all instances, specific activities and sessions—both theoretical and practical—are conducted with participants from various courses, disciplines, and study levels. This approach fosters a unified language and coordinated efforts among the diverse participants[7].

However, implementing transdisciplinary and interdisciplinary approaches can be complex due to critical challenges: Coordination and Collaboration through the different programs require significant coordination among diverse professors and stakeholders, which can be time-consuming and resource-intensive. Institutional Support, acquiring buy-in from the administration, and securing funding for interdisciplinary and transdisciplinary initiatives can be complex. Skill Development can also be complex because the parties need to develop leadership and cooperation skills to navigate the collaborative nature of interdisciplinary and transdisciplinary work.

## 3 INTEGRATING ACTORS IN HPC ECOSYSTEMS

Dynamic exchanges exist between different actors within the HPC ecosystem. Effective interaction fosters communication, knowledge sharing, and collaborative problem-solving. It is essential for building trust and understanding among stakeholders, which is crucial for successfully integrating and implementing HPC solutions. By uniting diverse perspectives through transdisciplinary and interdisciplinary approaches, HPC ecosystems can tackle complex

[7]The courses developed, and the associated material can be consulted (mainly in Spanish originally) at: https://wiki.sc3.uis.edu.co/

problems more effectively. This integration allows for exploring innovative solutions that may not emerge within isolated disciplines and enhanced problem-solving. Figure 2 shows the different elements impacted by each one of the roles in the HPC environment in levels, from infrastructure to applications.

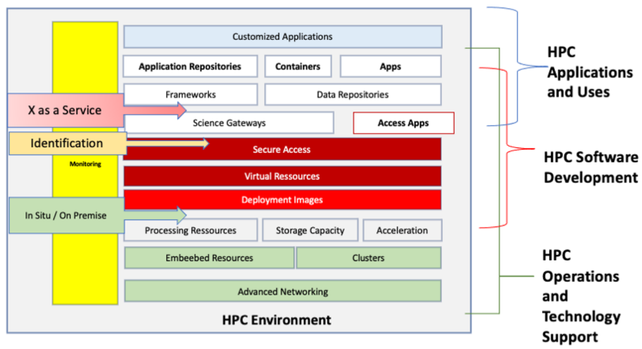



**Figure 2: HPC Roles in the HPC Environment.**

Figure 2 presents three primary actors in advanced computing projects: HPC technology and operations engineers, HPC software developers, and HPC scientists. These groups continually interact to meet their objectives and ensure the effective deployment and performance of applications in these initiatives. Additionally, specific skills are necessary for engaging with various components of an HPC system. For example, the scientific end user, who relies on the HPC environment for research support, customizes application usage and ensures accurate execution and deployment. In contrast, the developer establishes development and image deployment environments for scientists while facilitating communication with both users and DevOps actors. Furthermore, the administrator (DevOps), as depicted in the figure, offers operational support directly from the architecture figure 2, the integrating element is monitoring and performance evaluation.

Collaborative actions can lead to a more efficient use of resources, including computational power, funding, and human expertise. By pooling resources and knowledge, actors can achieve more significant outcomes than they could individually. This pooling can be considered resource optimization. Furthermore, collaboration among various stakeholders can stimulate innovation and transfer technology from research to practical applications. This is particularly important in HPC, where advancements in computational methods can significantly impact various industries.

Integrating actors from diverse backgrounds helps build capacity within the HPC ecosystem. It enhances participants' skills and competencies, preparing them to address future challenges collaboratively. Transdisciplinary approaches ensure that HPC research effectively addresses societal needs and challenges. By involving community stakeholders, researchers can align their work with real-world issues, enhancing the relevance and impact of their findings.

For example, in courses developed for undergraduate and graduate students in areas other than computer science, emphasis is placed on the proper use of resources and appropriate interaction with developers and administrators [5]. In computer science courses that involve implementing development and execution environments and/or supporting platforms aimed at end users, the need for interaction is raised to understand requirements and ensure customization and good performance, for example, by following good practices for monitoring and analyzing the performance of HPC environments.

Monitoring and observing the status of both high-level code and low-level processes allows, for example, identifying faults, efficient execution environments, load balancing, and time management without affecting accuracy. On the other hand, from this interaction, new common areas of research have been developed, thinking about sustainability: observing, for example, software quality, useful lifetime, energy efficiency, learning curves, and even, very recently, frugality[10].

On the other hand, training courses have been created that can be considered as specific modules of the different formal courses, for example, using the training given by NVIDIA within the academic programs of the NVIDIA DLI (or now via academy NVIDIA Academy programs. Likewise, activities within Summer schools we have hosted on campus or co-organized with our university in different years or even in specific tutorials.

The interaction between different courses, students, and levels has allowed the recognition of various actors and their roles and competencies and the development of shared spaces around good practices, as will be discussed in the following section. This generates challenges and allows the implementation of diverse strategies according to the interests and skills sought, searching integration.

## 4 CHALLENGES AND STRATEGIES

Academic disciplines are characterized by distinct cultures, terminologies, and methodologies, which impede effective communication and collaboration. Surmounting these differences necessitates deliberate efforts to cultivate mutual understanding. To transcend traditional barriers, incentives for cooperation must be aligned with interdisciplinary objectives.

Managing interactions among diverse actors can be complex and is a significant challenge. It requires effective coordination and leadership. Establishing clear communication channels and collaborative frameworks is essential for successful integration and credibility. For better or worse, the academic community is hierarchical and intellectually elitist, where authority in knowledge is necessary. Inclusion is another crucial challenge, as the technology community continues to be very uninclusive, not only in terms of gender but also in terms of culture and nationality.

Ultimately, we recognize the significance of sustainability within collaborative initiatives is duly acknowledged. The challenge of ensuring the resilience of collaborative efforts can be substantial. Continuous engagement and steadfast commitment from all stakeholders are imperative for sustaining momentum and realizing long-term objectives. This notion of sustainability encompasses not only ecological aims but also the advancement and well-being of humanity. Effective collaboration necessitates equity, integration of policies, and partnerships with industry players across various strata, including large corporations and local communities, as well as an array of expertise and resources. While collaboration presents considerable advantages, it also introduces certain challenges, including coordinating diverse stakeholders, managing conflicts, and securing enduring commitments. Collaborative initiatives remain one of the most efficacious approaches to addressing the intertwined challenges of sustainability (In contexts of emerging economies, where culture, production and technological development are involved, collaborative practices strengthen sustainability [8].

Given these challenges and the nature of the training activity, the student is placed in an environment tailored to their interests, needs, and motivations. Figure 3 shows the developmental spaces.

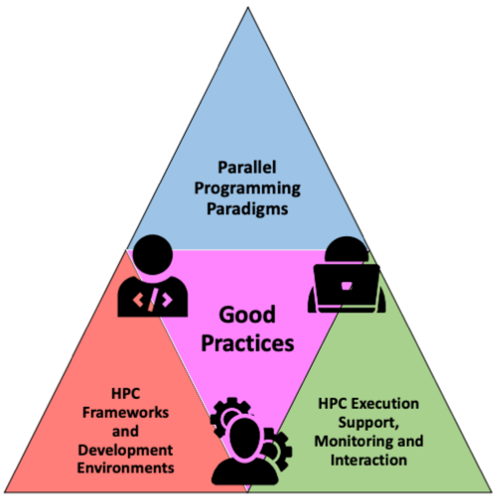



**Figure 3: Skills Development Spaces. The three main actors are represented by icons clockwise from the top left: HPCRSE, HPCUSE, and HPCTOE.**

The designated spaces are systematically structured around specific competencies, as shown in Figure 3, incorporating a shared lexicon and methodologies that foster enhanced communication and collaboration among all participants, irrespective of their experience levels, while cultivating trust in the instructors. Depending on the activities undertaken, various technical proficiencies and teamwork capabilities can be advanced within this ecosystem, whether through formal educational courses or specialized university programs. For individuals engaged in scientific disciplines or students from fields beyond computer science, we recommend courses concentrating on parallel programming paradigms and tools for task execution. For software developers, the curriculum underscores the utilization of frameworks, development environments, debugging techniques, and profiling methodologies. In DevOps, the courses provide an in-depth exploration of system architecture, support for high-performance computing execution, monitoring practices, and collaborative interactions.

## 5 CONCLUSION

Integrating a diverse range of participants within High-Performance Computing (HPC) ecosystems is crucial, fostering the synthesis of varied motivations through the principles of transdisciplinarity, interdisciplinarity, and collaboration. This integration is vital for effectively addressing complex challenges. HPC initiatives have the potential to enhance problem-solving capabilities, optimize resource utilization, and stimulate innovation by encouraging collaboration among a diverse group of stakeholders. Similarly, HPC—and generally any computational or technological training—requires establishing positions around sustainability and its impact on individuals and society. However, to realize the full potential of such collaborative efforts, it is essential to overcome cultural, institutional, and coordination challenges. A more cohesive and impactful HPC ecosystem can be created by underscoring the importance of collaboration and shared objectives.

## REFERENCES

[1] Kenneth A. Frank, Yong Zhao, William R. Penuel, Nicole Ellefson, and Susan Porter. 2011. Focus, Fiddle, and Friends: Experiences that Transform Knowledge for the Implementation of Innovations. *Sociology of Education* 84, 2 (2011), 137–156. https://doi.org/10.1177/0038040711401812 arXiv:https://doi.org/10.1177/0038040711401812

[2] Ramtin Jabbari, Nauman bin Ali, Kai Petersen, and Binish Tanveer. 2016. What is DevOps? A Systematic Mapping Study on Definitions and Practices. In *Proceedings of the Scientific Workshop Proceedings of XP2016 (XP '16 Workshops)*. Association for Computing Machinery, New York, NY, USA, Article 12, 11 pages. https://doi.org/10.1145/2962695.2962707

[3] Stelios Karozis, Xenia Ziouvelou, and Vangelis Karkaletsis. 2023. Mapping the national HPC ecosystem and training needs: The Greek paradigm. *The Journal of Supercomputing* 79, 10 (Feb. 2023), 10691–10705. https://doi.org/10.1007/s11227-023-05080-y

[4] Alexander T. Latinjak, Alain Morin, Thomas M. Brinthaupt, James Hardy, Antonis Hatzigeorgiadis, Philip C. Kendall, Christopher Neck, Emily J. Oliver, Małgorzata M. Puchalska-Wasyl, Alla V. Tovares, and Adam Winsler. 2023. Self-Talk: An Interdisciplinary Review and Transdisciplinary Model. *Review of General Psychology* 27, 4 (2023), 355–386. https://doi.org/10.1177/10892680231170263 arXiv:https://doi.org/10.1177/10892680231170263

[5] Teemu Leinonen and Eva Durall. 2014. Design Thinking and Collaborative Learning. *Comunicar* XXII (01 2014). https://doi.org/10.3916/C42-2014-10

[6] Reed Milewicz and Miranda Mundt. 2022. Building Bridges: Establishing a Dialogue Between Software Engineering Research and Computational Science. arXiv:cs.SE/2201.04007 https://arxiv.org/abs/2201.04007

[7] Keisuke Okamura. 2019. Interdisciplinarity revisited: evidence for research impact and dynamism. *Palgrave Communications* 5 (2019), 1–9. https://api.semanticscholar.org/CorpusID:207960568

[8] Michele M. O. Pereira, Rita C. Arantes, Luiz G. R. Antunesand Linda C. Hendry, Leonardo P. Deboç a, Marilia B. Bossle, and Luiz M. Antonialli. 2021. Sustainability Initiatives and Collaborative Practices: A Study of Emerging Economy Suppliers. *Latin American Business Review* 22, 4 (2021), 359–391. https://doi.org/10.1080/10978526.2021.1930550 arXiv:https://doi.org/10.1080/10978526.2021.1930550

[9] Melissa Terras, Julianne Nyhan, and Edward Vanhoutte (Eds.). 2016. *Defining digital humanities*. Routledge, London, England.

[10] Wim Vanderbauwhede. 2023. Frugal Computing - On the need for low-carbon and sustainable computing and the path towards zero-carbon computing. *CoRR* abs/2303.06642 (2023). https://doi.org/10.48550/ARXIV.2303.06642 arXiv:2303.06642

[11] Elena Verdú, Luisa Regueras, and Juan Pablo De Castro. 2008. An analysis of the research on adaptive Learning: The next generation of e-learning. *WSEAS Transactions on Information Science and Applications* 5 (06 2008), 859–868.

[12] J. Škrinárová and E. Vesel. 2016. Model of education and training strategy for the high performance computing. In *2016 International Conference on Emerging eLearning Technologies and Applications (ICETA)*. 315–320. https://doi.org/10.1109/ICETA.2016.7802080

# HPC-ED: Building a Sustainable Community Driven CyberTraining Catalog

Susan Mehringer
Center for Advanced Computing, Cornell University
shm7@cornell.edu

Katharine Cahill
New Jersey Institute of Technology
katharine.cahill@njit.edu

Charlie Dey
Texas Advanced Computing Center
charlie@tacc.utexas.edu

Brian Guilfoos
Ohio Supercomputer Center
guilfoos@osc.edu

David Joiner
Kean University
djoiner@kean.edu

Richard Knepper
Center for Advanced Computing, Cornell University
rich.knepper@cornell.edu

John-Paul Navarro
Argonne National Lab
navarro@anl.gov

Jeaime H. Powell
Omnibond Systems
jeaime@omnibond.com

Mary P. Thomas
University of California, San Diego
mpthomas@ucsd.edu

Zilu Wang
Center for Advanced Computing, Cornell University
zw427@cornell.edu

Jiesen Zhang
University of California, San Diego
jiz147@ucsd.edu

## ABSTRACT

HPC-ED is working to improve discovery and sharing of CyberTraining resources through the combination of the HPC-ED CyberTraining Catalog, an effective and flexible interface, thoughtful metadata design, and active community participation. HPC-ED encourages authors to share training resource information while retaining ownership and allows organizations to enrich their local portals with shared materials. By basing the architecture on an established, flexible framework, HPC-ED can provide a range of solutions people and organizations can employ for sharing and discovering materials.

In this paper we describe the initial pilot phase of the project, where we prototyped the HPC-ED catalog, established an initial metadata set, provided documentation, and began using the system to share and discover materials. We gathered community feedback through a variety of means, and are now planning an implementation phase based on evolving our architecture and tools to meet community needs and feedback through improved interfaces and tools designed to address a range of preferences.

## KEYWORDS

Education, Training, Community Engagement, HPC, Cyberinfrastructure, Metadata, Globus


https://doi.org/10.22369/issn.2153-4136/16/1/2

## 1 INTRODUCTION

A wealth of CyberTraining material exists, but locating appropriate resources can be challenging. HPC-ED's primary goal is to improve discovery and sharing of CyberTraining resources through the HPC-ED CyberTraining Catalog, an effective and flexible interface, and active community participation. HPC-ED enables training material owners to share metadata describing their resources, allows organizations to enrich their local portals with shared materials, and provides individuals with the ability to share and discover training resources. The catalog is based on Globus Search [6], an established, flexible framework; this allows HPC-ED to provide a range of solutions people and organizations can employ.

During the initial stage of the project, we prototyped the HPC-ED catalog, developed a set of basic metadata [15], provided instructions for sharing and discovering materials, and began populating the catalog using different sources. We gathered community feedback through surveys, presentations, tutorials, and hackathons, explored collaborations and engaged with projects interested in utilizing the catalog. Currently, the project is evolving to address community needs and feedback; we are adapting interfaces and building tools to address a range of preferences, including downloading filtered search results and using Jupyter Notebooks to share resources. Details of the pilot project can be found in [30].

Building on recent improvements, our immediate goals are to implement an improved approach for hosting the HPC-ED catalog in production mode, expand its user base, and continue to improve tools and interfaces, such as integrating Large Language Models (LLMs) to generate relevant and contextually appropriate metadata and content and supporting a new approach to share material metadata (see Section 3). By implementing a federated catalog with a robust interface, HPC-ED can broaden the reach of training materials, increase their usage, and expand local learning resources.

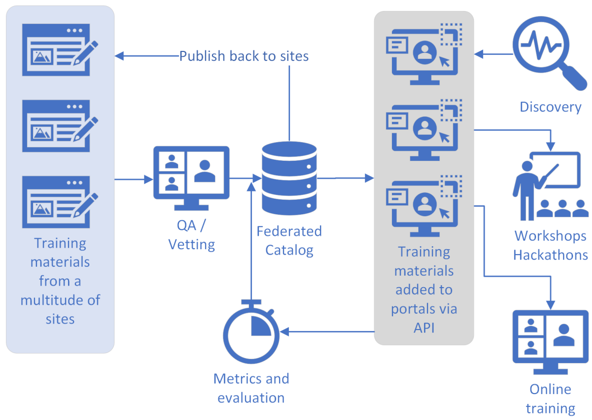



Figure 1: Pilot project architecture: Key features include: materials remain in original location; only metadata describing the materials are stored; individuals or smaller organizations can share their materials more broadly.

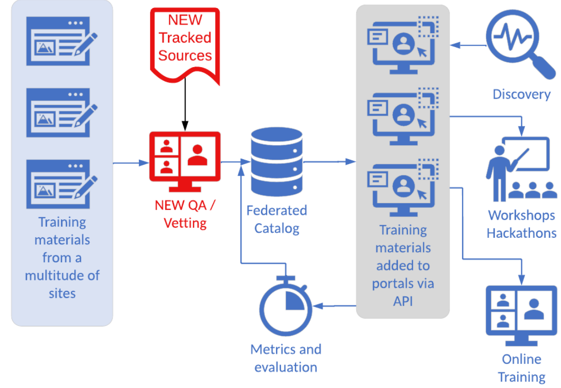



Figure 2: Implementation project architecture: key change is a decentralized ***pull model*** with indirect interaction with the HPC-ED catalog, providing improved security, quality checks, and enabling tool development.

## 2 PILOT: PROOF OF CONCEPT

The pilot project and results are highlighted in this section. Details about the project can be found on the project web page [14] and additional progress and results can be found in a recent paper [30].

### 2.1 Architecture

The pilot project architecture is designed to enable CyberTraining material sharing and discovery. The community currently relies on finding resources on local training catalogs and discovery portals hosted at larger research or commercial organizations, or by using internet search engines. The HPC-ED Pilot architecture (see Figure 1) leverages and builds on the strengths and flexibility of organization-specific training catalogs and portals while enabling individuals to share their materials and discover training materials from all HPC-ED catalog participants.

An overview of the process as implemented in the pilot project is presented in Figure 1. Key features of this architecture are that materials remain in their original location; only metadata describing the materials are stored in the HPC-ED catalog; individuals or smaller organizations can share their materials more broadly; thoughtful metadata design results in more appropriate search results; and the robust software base used is stable and provides excellent flexibility for mid-level tools.

Using Globus Search commands, materials can be shared or discovered. Sharing materials is done by compiling HPC-ED specified metadata in JSON (JavaScript Object Notation [17]) format and sharing it with the HPC-ED catalog. Sharing to the pilot catalog requires authorized credentials, which are issued by the HPC-ED project. Searching the catalog is done by issuing a Globus Search command or by using any web portal that is populated with catalog materials. Authorization credentials are not needed to search public HPC-ED catalogs. Sharing and discovery is described in detail on the project wiki page. [16]

### 2.2 HPC-ED Pilot Metadata

The HPC-ED project is committed to ensuring that the digital objects stored in the repository are findable and reusable. This is accomplished by following the Findable, Accessible, Interoperable and Reusable (FAIR) principles, [5]. The HPC-ED Pilot project website describes the metadata that is being used for an initial demonstration of capability. [15] We discuss our implementation plans in Section 5.2. This includes custom metadata and existing sets including: Metadata required by the Globus search, the Research Data Alliance (RDA) ontology for recommended minimal metadata set for training materials [12], the set of HPC-ED metadata, and the Dublin Core Learning Resource Type. [3]

HPC-ED uses Globus search tools, which require a JSON formatted document that is used to share information on resources and events to the HPC-ED catalog. [6] Details on the fields in the JSON document can be found in the HPC-ED Metadata Description and the Globus Search Overview. Currently there are 9 required metadata fields and 12 optional fields, including Title, Description, Authors, Publisher, Type, Language, Cost, Format, License, Target Group, Expertise Level, Certification details, and very importantly, Persistent Identifiers, Tags, or Keywords. One optional field is "keywords," which can be used to add additional search metadata.

A challenge for the repository lies in the area of searching and discovery, especially when the number of objects is large. The metadata and keywords used by the HPC-ED project will be used to create an HPC training materials taxonomy, knowledge graph, and eventually an ontology. This data will be used to support a variety of clients to share and discover materials. These can be used to create an LLM for discovering training materials.

### 2.3 Communication and Community

To facilitate effective communication and participation in the community, a dedicated website and wiki were established as part of the project. The website provides a comprehensive overview of the project, including details of recent and upcoming events. The wiki serves as a repository for procedural documentation, which is regularly updated to ensure clarity and accessibility, featuring resources such as a quick start guide.

In addition to these resources, a Slack channel and an ACCESS affinity group were created to offer support to users testing the new tools. Regular emails have been distributed throughout the project year to keep mailing list subscribers informed of new developments, features, and future events. An informational session was held in April for people interested in utilizing the publication and sharing tools. During this session, processes were reviewed and feedback was sought. For those unable to attend, the session was recorded and made available on the project website to ensure broad accessibility.

Project documentation is provided on the HPC-ED web site [14] which includes a project overview, community activities, events, and contact information. Technical documentation for sharing and discovering HPC-ED catalog materials are available on the HPC-ED wiki pages [16]. The wiki includes a process overview, a full description of the required and optional metadata, along with detailed and quick start documents on how to share and discover materials. The recording of an early tester online event is available online [13].

HPC-ED is seeing growing interest in engagement from people and organizations, and coordination with projects as a result of meetings, presentations, and hackathons 5.1. Recent presentations include *Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* [30], *Assessing Shared Material Usage in the High Performance Computing (HPC) Education and Training Community* [19], *The HPC Federated Learning Catalog* [18], and *Scaling HPC Education* [20]. In addition, a tutorial *Publishing and Discovering CyberTraining Materials Across the HPC and CI Research Communities* [21] was held at PEARC24 [24].

With our objectives for the pilot project successfully testing the concept met, effort has begun on initiating an action plan for implementation.

## 3 IMPLEMENTATION: IMPROVING USABILITY AND SCALING

The HPC-ED *implementation plan* includes a decentralized architecture that retains flexibility and adds security with the integration of several new features: implementing a distributed model for sharing of data by contributing partners, adding security by controlling which partner data is added to the catalog, validating and vetting partner and shared metadata, developing different sharing and discovery approaches for different types of partner capabilities, and engaging in collaborations to develop new tools and build community. These new features are described the sections below.

### 3.1 Architecture

We are shifting to a new architecture, as shown in Figure 2, that will be more flexible, secure, and easier to use. The primary change is that rather than giving all community members direct access to share resource metadata with the HPC-ED catalog, we are shifting to a decentralized *pull model* where contributors will not directly share metadata with the HPC-ED catalog. Instead, the relevant data will be pulled from local catalogs to the HPC-ED catalog. When the data is pulled, identity information will be added, and quality checks will be conducted. In addition, only the data that is necessary for the HPC-ED catalog will be pulled. This change has many advantages: the community member does not need to obtain a provider ID or create unique subject tags as they will be automatically created by the pull software, the community member can choose the automatic sharing interval (e.g. daily, weekly, or monthly), and each person or site can choose the method that best suits their needs. By removing the ability to share directly with the catalog, the project benefits from an interim step that allows security and quality checks to be applied.

This architectural shift enables a variety of sharing and discovery options, such as those described in the following sections.

### 3.2 Sharing Training Material from Local Catalogs

The new HPC-ED architecture supports a decentralized data collection system (see Figure 3). The HPC-ED architecture supports the creation of local catalogs created by different organizations and projects. When registered, the system will identify relevant contributions, and then will pull, inspect and validate the material. Examples of different contribution mechanisms are described below.

For organizations with multiple resources to share, but choose not to maintain a local catalog, the system can accept shared materials from a variety of document formats. A few examples are provided below.

*3.2.1 Spreadsheets:* A spreadsheet template is provided that includes dropdown selections, help tabs, and common values, all intended to simplify choosing appropriate metadata to describe the shared materials. The spreadsheet can either be shared once by email to the project email address, or, if shared with the project via a google spreadsheet, it would be used to update the shared material at the selected periodicity. This method also simplifies the process by removing the requirement to provide a provider ID and unique subject fields.

*3.2.2 Simple Text Files.* The system architecture can accept multiple file formats, including a text file. In this example, we use "key=value" pairs based on the HPC-ED required metadata. A snippet of the code is shown in Figure 4.

### 3.3 Discovery Clients

The federated catalog has a search API that is client agnostic. [15] Currently the search query returns the JSON formatted metadata information. Other query mechanisms will be created as part of the implementation phase of the project.

*3.3.1 Download search results to a web page.* A simple method to incorporate select materials from the catalog into a local web site is to download those results directly to an html file. This method has been implemented on the development portal [23]. After using the filters available on the portal to find the desired materials, a button on the page shares a downloaded web page with the results that can be incorporated into a local web site.

*3.3.2 A Web Application to Discover Materials from a Local Portal.* A collaborative effort is underway to provide a web application

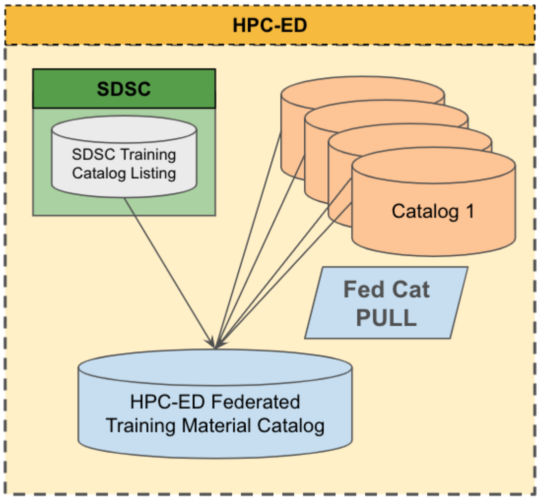



**Figure 3: Diagram demonstrates how resource material is pulled from remote catalogs based on validation criteria and permissions.**

```
"Title": "Expanse Webinar: Introduction to Neural Networks, Convolution Neural Networks and Deep Learning on Expanse",
"Abstract": "This webinar will be a quick introduction and overview of neural networks, convolution networks...",
"Authors": "Paul Rodriguez, Ph.D. ",
"Keywords": [ "Expanse", "HPC Training", "Industry"],
"Resource_URL_Type": "URL",
"Learning_Resource_Type": "recorded lesson",
"Start_Datetime": "2022-05-19T18:00:00.000000",
"URL": "https://education.sdsc.edu/training/interactive/202205_ExpanseWebinar-Intro-to-Neural-Networks-and-Deep-Learning-on-Expanse"
```

**Figure 4: Example showing the SDSC Training Materials text file used to share materials, consisting of KEY=VALUE pairs, where KEY is based on the HPC-ED metadata. [15]**

that can be used on any local site to let visitors search the HPC-ED catalog without leaving their local community portal. This would be available to download from GitHub in the form of an iframe plug-in.

*3.3.3 Training Materials Web Page.* The discovery example in Figure 5 shows the SDSC Training Materials client (Webhost) discovering relevant metadata stored in the catalog (via a python script titled "intvid.py") to display training material to the different clients. The diagram shows the specific case of retrieving videos from YouTube [25] based on video url information contained in the federated catalog.

## 4 COLLABORATIONS

The collaborative efforts of multiple organizations have contributed significantly to the advancement of this project. ACCESS Support and ACCESS Operations have played a crucial role in metadata development and hosting a pilot catalog search portal. Future plans include the integration of ACCESS training materials into the repository and the provision of our search tools to ACCESS users, including a proposed NAIRR (nairr.org) training catalog collection. This aligns with the project's goal of enhancing metadata interoperability and improving user discovery.

The ACM SIGHPC Education chapter has emerged as a valuable partner for dissemination through its Education committee. This collaboration facilitates the broader distribution of project resources and findings within the high-performance computing education community. Additionally, the Outreach committee is developing a web application to integrate the repository into the chapter's website, which will enhance accessibility and user engagement.

These partnerships have been instrumental in addressing key challenges in metadata management, including analyzing metadata requirements, adopting appropriate schemas, and creating metadata content. The collaborations also support the project's focus on improving search functionality and providing web services, which are essential for enhancing the discoverability and utility of digital resources. We will continue to reach out through our website and email updates and outreach opportunities to connect with the HPC Education and Training Community and identify new collaborations.

## 5 QUALITY ASSURANCE

Our quality assurance plan is focused on allowing for both human collection and automated collection of quality assurance and review information on catalog items. Defining quality variables in digital libraries can contain a large number of dimensions [7, 31]. After a review of different potential quality variables, we chose 4 items that would help us to maintain a process easy enough for the end user to encourage participation and enough data to help with the overall goal of connecting end users to materials hosted by catalog maintainers. These 4 items include 2 human contributed review items, and 2 automated items.

For human contributed review metadata, we will focus on whether the end user found materials that were accurate and well crafted (content correctness), and whether the search process helped the user to find what they were looking for (search relevance). As our goal is to provide a federated search across catalogs, our focus is at the item level and not the collection level. Reviews will ask end users two questions, with a Likert scale response (displayed as star ratings) and an optional text exposition of the user's answer. First, "Did you find what you were looking for?" Second, "How would you rate the quality of the materials you accessed?" Likert scale ratings will allow for a no response option in cases where a response is not applicable (A user who did not find what they were looking for, for example, may not choose to follow up with a statement on material quality.)

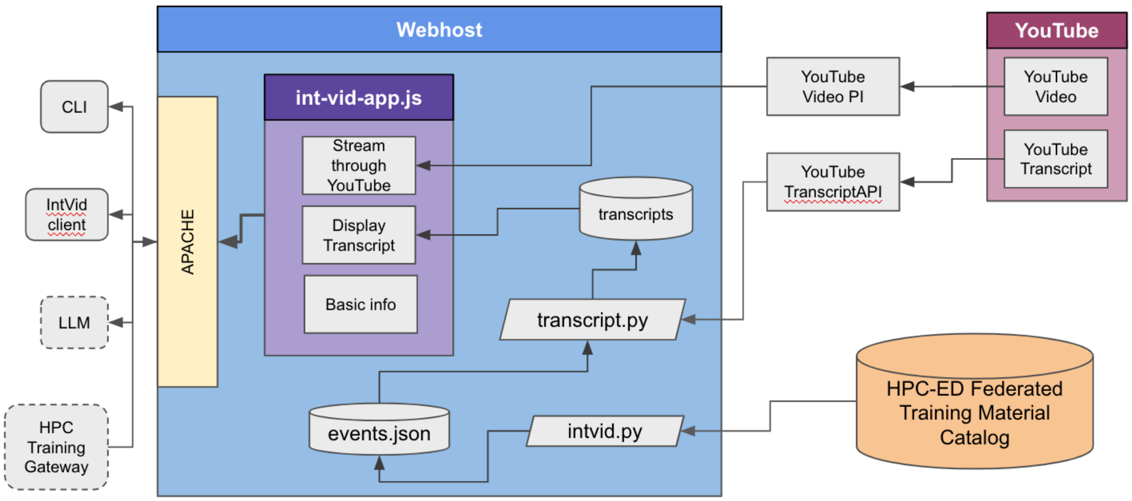



**Figure 5: Discovery example showing the SDSC Training Materials client discovering relevant metadata stored in the catalog to display training material to the different clients. Diagram shows specific case of retrieving videos from YouTube [25]**

For automated review metadata, we will focus on availability and accessibility, as measured by standard uptime measures which can be checked periodically and ADA compliance, for which there are multiple online tools available which have been used for studies in educational settings (See for example [29]).

Review metadata will be stored in a single Globus collection managed by the project, to ensure consistency of metadata format, and to allow for one to many relations between items and reviews. Reviewer identity will be managed and collected through the user's CI login, and stored by ORCID ID. For automated review metadata, a list of datetime values for which the URI was checked along with response will be stored, and ADA compliance will include metadata on the datetime of each check, the compliance tool used, and the result.

We will develop JavaScript based web applications that can be embedded into both item pages and into search results.

## 5.1 Workshops/Hackathons

Hackathons have become valuable tools in high-performance computing (HPC) education and training, providing an intensive, hands-on environment for participants to engage with real-world computational challenges. These events foster collaboration and innovation by bringing together students, professionals, and researchers to solve complex problems using HPC resources. Through hackathons, participants not only deepen their technical skills in parallel computing, data analysis, and algorithm optimization, but also develop critical soft skills such as teamwork, problem-solving, and communication. This immersive learning experience accelerates the development of expertise in HPC, making it a highly effective model for educational and training programs in the field.

A central repository of high-performance computing (HPC) training materials can significantly enhance the effectiveness and impact of HPC hackathons. By providing participants with ready access to comprehensive resources, such as tutorials, code libraries, case studies, and best practices, the repository ensures that both novice and experienced participants can quickly upskill and troubleshoot during the event. This centralized resource hub allows teams to focus on applying knowledge to real-world HPC challenges, rather than spending time searching for learning materials. Furthermore, the repository promotes consistency in training, helping hackathon organizers standardize the educational experience and ensure that participants have the foundational knowledge needed to maximize their productivity and success during the hackathon.

Hackathons can play a pivotal role in improving the development of high-performance computing (HPC) training materials by providing a real-world testing ground for educational content. During these events, participants encounter various challenges and gaps in their understanding, which can reveal shortcomings in existing materials. This feedback from participants, especially in terms of the resources they found most helpful or lacking, offers valuable insights for refining and expanding training content. Additionally, the collaborative and fast-paced nature of hackathons often leads to the creation of new tools, workflows, and techniques, which can be incorporated into future HPC training modules. The dynamic environment of a hackathon thus fosters continuous improvement of training resources, ensuring that they remain relevant and aligned with the evolving needs of the HPC community.

As a means to grow the data contained in and the community around the HPC-ED catalog, workshops, tutorials, and hackathons have been held at multiple cyberinfrastructure-related events. The events began with mini-workshops to engage with the community in a fashion that both spurred conversation and gathered terminology data from the audience. Later workshops and tutorials provided overviews for alpha and beta users. Additionally, a student-centered hackathon saw the creation of example web applications built with the HPC-ED catalog API.

*5.1.1 Mini-Workshops.* Initial community engagement with the HPC community began through the use of mini-workshops in which terminology linked to categorization was gathered. Two such events included the SGX3 Faculty Hackathon [9] and later Gateways Conference [26]. The workshops prompted the audience members to define overloaded terms common to cyberinfrastructure in the community. The results were collected with the interactive whiteboard application Google Jamboard [8]. The results were displayed during the sessions with commentary to induce conversation and dissemination further. The audience members produced 211 definitions across six common terms. The results will aid in future HPC-ED categorization methodologies.

*5.1.2 Hackathon.* In June of 2024 the HPC-ED API was selected by three undergraduate student groups as the focus of their hackathon projects. The event, SGX3's HackHPC@ADMI24 [11] hosted by the The Association of Computer Science Departments at Minority Institutions (ADMI) [22] and funded by Science Gateways Community Institute [28] through SGX3 [1] for workforce development. The three teams were comprised of four students and a team mentor. In summary, the teams created web applications that implemented HPC-ED catalog search and publication [10]. Deliverables included GitHub repositories, presentations, and posters of which two were submitted to the Gateways24 Conference [27].

*5.1.3 Workshops and Tutorials.* Several workshops and tutorials have been conducted focusing on community usage and education. From workshops with concept dissemination to alpha user tutorials, HPC-ED has used both virtual and in-person events for directed community engagement. One such event took place at the Practice & Experience in Advanced Research Computing PEARC24 [24] annual conference as a tutorial titled *Publishing and Discovering Cybertraining Materials Across the HPC and CI Research Communities* [21]. The session was attended by over 30 educators/trainers in the HPC field. The tutorial included the use of a developmental catalog that was populated by the audience members. As true alpha users, the engagement both highlighted future enhancements and features through conversations and discussions fostered during the event. The audience members provided real-time feedback as a very diverse sample of clients.

## 5.2 Metadata and Metrics

As described in Section 2.2, the federated catalog metadata is designed to enable training material owners to share materials by publishing metadata associated with those materials to the federated catalog. In the pilot phase, we adopted a "minimal" metadata set approach, details of which can be found on the HPC-ED metadata description web page [15]. These metadata terms were used to conduct simple searches.

In the implementation phase, we will extend the existing metadata fields to include the following categories: metadata that describes the training material, its access methods, and educational characteristics; metadata that identifies the publisher and source of the training material so that when an individual selects a specific training item they can be directed to the source catalog that published that material in order to browse all available information and to access that training material; metrics metadata about educational material reviews, ratings, and access metrics. Information about the metrics will be drawn from the quality assurance part of the project, as described earlier in Section 5.

Defining, categorizing, and standardizing the metadata will be a significant effort. Where possible, we will identify and use existing metadata sets, taxonomies, and ontologies. [2, 4, 32] Where needed, we will add new terms to these existing ontologies and work with existing communities to update them or to extend the HPC-ED training materials ontology.

# 6 FUTURE WORK

Looking to future directions for the HPC-ED project, we will extend interface functionality, expand the types and quantity of catalog offerings, and pursue additional outreach opportunities to raise awareness of the project. The primary value of any directory of training materials is in the breadth and quality of the offerings available, and our immediate plan is to make adding entries to the catalog an easier task. There are still a number of organizations who have expressed interest sharing their materials but are waiting until the process is easier. Improving the process to add materials is a critical element to supporting the growth of the catalog and bringing additional partners to the project.

As the process for listing more entries in the catalog improves, we will continue to work to build awareness of the project and the potential for discovering and sharing more materials. This includes meetings within the national cyberinfrastructure community that bring together trainers and learners (PEARC, SCxy, and other meetings), as well as reaching out through the CyberTraining program in order to support these projects in making their training offerings easier to discover and share.

With additional items discoverable through HPC-ED, providing fully-functional quality assurance for catalog listings will be a necessity to ensure that users are directed to items that are relevant, salient, and effective. HPC-ED will incorporate a familiar content ratings system ("one to five stars") that allows users to contribute scores for materials appearing in the catalog, and will experiment with leveraging AI tools to provide summaries and quality ratings.

Looking further to extending the capabilities of the project, we will make use of the idea-generating potential of Hackathons and Workshops where the HPC-ED team can participate and help identify refinements and improvements to the catalog interface options, search, quality assurance, and integrations. Our team will also look to the capabilities of LLMs for providing better training results and summaries of content. Throughout the process of supporting sharing and discovery of training materials by a wider group of trainers and learners, we will continue to note the contributions of users of the catalog in order to identify capability improvements and extensions.


# ACKNOWLEDGMENTS

Our thanks for collaborative efforts go to the SIGHPC Education Chapter [1]. This project is funded through the NSF CyberTraining:


[1]SGX3 was funded by the National Science Foundation under award number 2231406

Pilot: HPC ED: Building a Federated Repository and Increasing Access through CyberTraining (#2320977); and supported by multiple NSF grants (#2230127, #2017767, #2320934, #2112606). We appreciate being granted access to NSF resources and services including: the NSF ACCESS Track 3 Award: COre National Ecosystem for CyberinfrasTructure (CONECT) (#2138307); and several HPC resources (SDSC Expanse #1928224 and TACC Stampede # 1663578).

## REFERENCES

[1] ACM. 2023. Association for Computing Machinery: SIGHPC Education chapter. https://sighpceducation.acm.org/
[2] Ayesha Ameen, Khaleel Ur Rahman Khan, and B. Padmaja Rani. 2012. Creation of Ontology in Education Domain. In *2012 IEEE Fourth International Conference on Technology for Education*. 237–238. https://doi.org/10.1109/T4E.2012.50
[3] Dublin Core. 2043. LRMI Learning Resource Type vocabulary. Retrieved September 8, 2024 from https://www.dublincore.org/specifications/lrmi/concept_schemes/learningResourceType
[4] Dong Dai, Yong Chen, Philip Carns, John Jenkins, Wei Zhang, and Robert Ross. 2019. Managing Rich Metadata in High-Performance Computing Systems Using a Graph Model. *IEEE Transactions on Parallel and Distributed Systems* 30, 7 (2019), 1613–1627. https://doi.org/10.1109/TPDS.2018.2887380
[5] Leyla Garcia, Bérénice Batut, Melissa L. Burke, Mateusz Kuzak, Fotis Psomopoulos, Ricardo Arcila, Teresa K. Attwood, Niall Beard, Denise Carvalho-Silva, Alexandros C. Dimopoulos, Victoria Dominguez Del Angel, Michel Dumontier, Kim T. Gurwitz, Roland Krause, Peter McQuilton, Loredana Le Pera, Sarah L. Morgan, Päivi Rauste, Allegra Via, Pascal Kahlem, Gabriella Rustici, Celia W.G. Van Gelder, and Patricia M. Palagi. 2020. Ten simple rules for making training materials FAIR. *PLoS Computational Biology* 16, 5 (2020), 1–9. https://doi.org/10.1371/journal.pcbi.1007854
[6] Globus. 2024. Globus Search Service. https://docs.globus.org/api/search/.
[7] Marcos André Gonçalves, Bárbara L. Moreira, Edward A. Fox, and Layne T. Watson. 2007. "What is a good digital library?" – A quality model for digital libraries. *Information Processing & Management* 43, 5 (2007), 1416–1437. https://doi.org/10.1016/j.ipm.2006.11.010 Patent Processing.
[8] Google. 2024. Google Jamboard. https://jamboard.google.com
[9] HackHPC. 2023. CiberInfrastructure. https://hackhpc.github.io/facultyhack-gateways23/assets/slides/pre-event_training/FacultyHack-HPC-ED.pdf
[10] HackHPC. 2024. Hackathon Teams. https://hackhpc.github.io/sgx3admi24/teams.html
[11] HackHPC. 2024. SGX3's Association of Computer Science Departments at Minority Institutions (ADMI) Hackathon. https://hackhpc.github.io/sgx3admi24/
[12] N. J. Hoebelheinrich, K. Biernacka, M. Brazas, L. J. Castro, N. Fiore, M. Hellström, E. Lazzeri, E. Leenarts, P. M. Martinez Lavanchy, E. Newbold, A. Nurnberger, E. Plomp, L. Vaira, and A. van Gelder, C. W. G. & Whyte. 2022. Recommendations for a minimal metadata set to aid harmonised discovery of learning resources. https://doi.org/10.15497/RDA00073
[13] HPC-ED. 2024. Early Tester Training. https://www.youtube.com/watch?v=fD9g5z5qZFQ
[14] HPC-ED. 2024. HPC-ED Documentation. https://hpc-ed.github.io/
[15] HPC-ED. 2024. HPC-ED Metadata. https://github.com/HPC-ED/HPC-ED.github.io/wiki/Metadata-Description
[16] HPC-ED. 2024. HPC-ED Wiki Documentation. https://hpc-ed.github.io/wiki
[17] JSON. 2024. JavaScript Object Notation. https://www.json.org/
[18] Richard Knepper. 2024. The HPC Federated Learning Catalog. In *Thirty-First International Conference on Learning*.
[19] Susan Mehringer, Kate Cahill, Scott Lathrop, Charlie Dey, Mary Thomas, and Jeaime H Powell. 2023. Assessing Shared Material Usage in the High Performance Computing ( HPC ) Education and Training Community. In *Sixth Workshop on Strategies for Enhancing HPC Education and Training (SEHET23)*. https://doi.org/10.22369/issn.2153-4136/14/2/4
[20] Susan Mehringer, Mary P Thomas, Charlie Dey, Kate Cahill, David Joiner, Richard Knepper, and Jeaime H Powell. 2023. Scaling HPC Education. In *Tenth SC Workshop on Best Practices for HPC Training and Education BPHTE23*. Denver, 41–46. https://doi.org/10.22369/issn.2153-4136/15/1/8
[21] John-Paul Navarro, Katharine Cahill, Susan Mehringer, Je'Aime Powell, and Richard Knepper. 2024. Publishing and Discovering Cybertraining Materials Across the HPC and CI Research Communities. https://github.com/HPC-ED/PEARC-Tutorial-2024/tree/main
[22] Association of Computer Science Departments at Minority Institutions. 2024. What is ADMI? https://admiusa.org
[23] ACCESS Operations. 2024. HPC-ED Developers Portal. https://search-pilot.operations.access-ci.org/.
[24] PEARC. 2024. PEARC24. https://pearc.acm.org/pearc24
[25] SDSC. 2024. Interactive Video Tutorials. https://education.sdsc.edu/training/interactive.
[26] SGX3. 2023. Science Gateways 23 Annual Conference. https://sciencegateways.org/gateways2023
[27] SGX3. 2024. Gateways24. https://sciencegateways.org/gateways2024
[28] SGX3. 2024. SGX3 - A Center of Excellence to Extend Access, Expand the Community, and Exemplify Good Practices for CI Through Science Gateways. https://sciencegateways.org/
[29] Tatiana I Solovieva and Jeremy M Bock. 2014. Monitoring for Accessibility and University Websites: Meeting the Needs of People with Disabilities. *Journal of Postsecondary Education and Disability* 27, 2 (2014), 113–127.
[30] Mary P Thomas, Susan Mehringer, Katharine Cahill, Charlie Dey, Brian Guilfoos, David Joiner, John-Paul Navarro, Jeaime H Powell, and Richard Knepper. 2024. Building a Federated Catalog for CyberTraining Materials: The HPC-ED Pilot Project. In *Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (New York, NY, USA), Vol. 1. ACM, 1–5. Issue 1. https://doi.org/10.1145/3626203.3670586
[31] Hong Xie. 2006. Evaluation of digital libraries: Criteria and problems from users' perspectives. *Library and Information Science Research* 28, 3 (2006), 433–452.
[32] Aolong Zhou, Kaijun Ren, Xiaoyong Li, Wen Zhang, Xiaoli Ren, and Kefeng Deng. 2021. Semantic-based discovery method for high-performance computing resources in cyber-physical systems. *Microprocessors and Microsystems* 80 (2021), 103328. https://doi.org/10.1016/j.micpro.2020.103328

# Exercise: Design a Learning Pathway

Weronika Filinger
EPCC, The University of Edinburgh
w.filinger@epcc.ed.ac.uk

Jeremy Cohen
Imperial College London
jeremy.cohen@imperial.ac.uk

Samantha Wittke
CSC- IT Center for Science
samantha.wittke@csc.fi

## ABSTRACT

Despite the quantity of existing training materials, acquisition and development of the specialist skills required for High Performance Computing (HPC) is not straightforward enough to address the needs of the growing, diversifying and constantly evolving HPC community. The HPC education and training community is exploring different approaches that could facilitate the uptake and progression of technical skills - one of those new approaches is focused on defining and formalising learning pathways. In this lightning talk we will briefly present an exercise designed as a starting point for capturing and outlining learning pathways for the HPC community. This exercise was run for the first time during the ISC'24 BoF on "Developing a Sustainable Future for HPC and RSE Skills: Training Pathways and Structures" and was accompanied by a Mentimeter survey to evaluate its effectiveness. The summary of the survey results is also included.

## KEYWORDS

Learning Pathways, HPC Training, Research Software Skills

## 1 INTRODUCTION

To address the educational needs of the growing HPC community we need new approaches that will not only ensure multiple entry points to training for new HPC users but also enable continuous professional development for all members of the HPC community, regardless of their role. These 'learning pathways' are not meant to replace any of the existing training efforts but rather complement them. The learning pathways provide two major benefits, first they encourage a shift from teaching topics in a linear manner toward concept based learning pathways that better align with adult learning models. Secondly, learning pathways are an important component that is needed to make training content more FAIR - findable, accessible, interoperable and reusable. Developing interoperable, citable and persistent training materials is key to creating personalised learning pathways that directly correspond to the training needs and job requirements of HPC community members, especially in the exascale era.

Data that was collected during the community driven BoF sessions at ISC'23 and SC'23 presented a snapshot of the educational HPC landscape and illustrated some of the challenges associated with creation of learning pathways [1, 2]. The learning pathways need to be personalised enough to be useful, but it's impossible for training providers to outline and support every possible learning journey. Therefore, the goal of the ISC'24 BoF was to capture example learning pathways and explore general trends that could be used to guide the design of other pathways, which was done through the exercise described below. The exercise and survey results are available at: https://zenodo.org/records/11395712.

## 2 EXERCISE DESIGN

The exercise sheet (A3) designed for the session consists of two fields at opposite ends that need to be filled in - starting point and learning objective - and a set of bubbles containing skills from 6 different categories that should be connected to create a path. The skill categorisation is based on the skill tree developed by the HPC Certification Forum (HPC-CF) [3] and includes these categories:

- Use of HPC Environment
- HPC Knowledge
- Software Development
- Performance Engineering
- System Administration
- Big Data Analytics

Due to the space limitation on paper, not all of the skills defined by HPC-CF (v. CS-1.0) have been used. Nonetheless, we felt that running this as a paper-based exercise would help participants to engage more with the topic than trying to use online tooling. The participants were asked to define a starting point and learning objective and then to create a learning pathway connecting the two using the listed skills. The learning path could be outlined in any way that worked for the participants - they could annotate the existing content, add skills, specify order or co-dependencies between skills etc. The example pathway, developed by one of the authors, is shown in Figure 1. The goal of the exercise was to collect as many different learning pathways as possible and to test if a similar template could be developed further to facilitate design of new pathways for specific skill sets or roles. This meant some of the pathways were focused on specific skill sets while others covered 20+ years of work experience. Some were linear and others had multiple loops, branches and merging points between skills.

Many of the participants expressed their interest in repeating this exercise within their institutions or even with their user base. Depending on the intended scope and audience of the exercise, it is recommended to clearly define the desired granularity of starting points, learning objectives and the pathways. This will help to narrow the scope and design something useful for the participants.

## 3 SURVEY RESULTS

The second part of the ISC'24 BoF session was used to understand how participants felt the exercise went, using an online Mentimeter survey. The survey was designed to be fully anonymous and consisted of 10 questions, including multiple choice, Likert scale and


https://doi.org/10.22369/issn.2153-4136/16/1/3

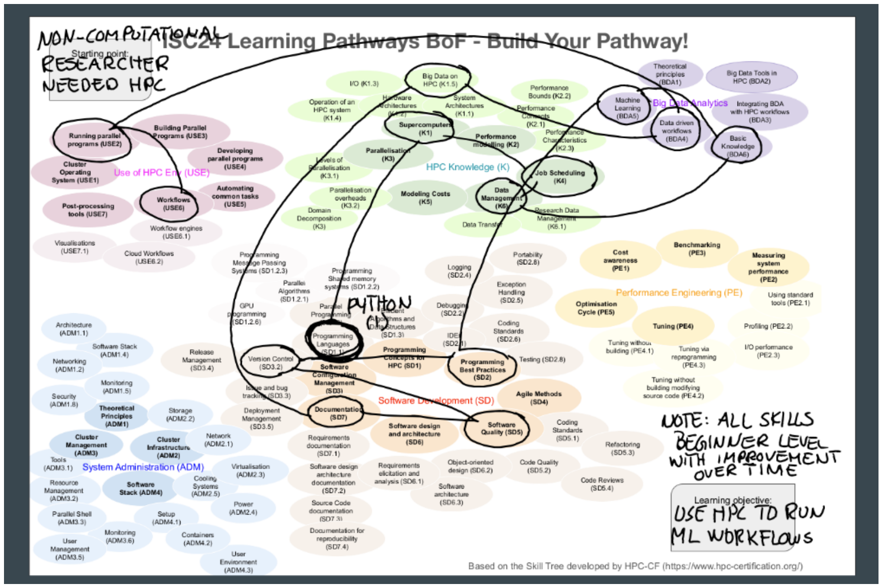



**Figure 1: An example learning pathway created by Samantha Wittke to test the design of the exercise.**

open questions. Out of about 70 people in the room at the ISC'24 session up to 39 provided responses to at least one question.

The first question asked the participants to describe their role in the educational context, from a set of options, and allowed multiple answers - 23 declared themselves as educators/trainers, 18 as training content providers, 8 as training managers, 19 as learners and 12 as having another role. Out of 39 respondents 25 indicated this was their first time attempting to design a learning pathway. The participants strongly agreed that the exercise illustrated the challenges of learning pathway design and that designing learning pathways is hard - both statements got the average score of 4.1 out of 5. They also generally agreed that the listed skills provided enough scaffolding to build useful paths and that the skill granularity was sufficient - both scored 3.3. Most also wished for more example pathways to exist - score 3.9.

Most people started their design process at the 'starting point' (17), some started with the learning objective (9) and the rest started somewhere in the middle i.e. with a specific skill (11). Most people also stated that the path they designed includes branching and merging points, or circular relations. Only 3 said their path was linear. Some of the design challenges included: determining the appropriate starting and ending point, deciding on the skill order, grasping the scope of possibilities, keeping the path relevant and meaningful, thinking about inter-dependencies and skill levels, and identifying missing skills. Many participants also believed their pathway was personal, task driven and could be hard to follow in different contexts e.g. working environment, role. It was also noted that learning takes time and there are always competing priorities.

Finally, when asked what could be done to make learning pathways easier to discover and navigate, some of the answers included: providing problem to skill maps, a catalogue of pathways that is searchable/browsable by starting/ending points or roles, better alignment with career options, more higher level examples that can be customised to fit the individual's needs, and collecting all relevant resources in a single location. Most of the answers aligned well with many on-going efforts within the HPC community, clearly demonstrating how important they are.

## 4 NEXT STEPS

Subsequently, the exercise was also run at another BoF session at PEARC'24. Across both sessions, the authors collected over 60 example pathways that need to be analysed. Additionally, the exercise is going to be refined and hopefully used in different contexts to further the community understanding of how the learning pathways should be designed and used. Anyone interested in reusing this exercise or adapting it to their own needs is free to do so under the CC-BY v. 4.0 license.

## 5 ACKNOWLEDGEMENTS

JC and WF acknowledge support from the UNIVERSE-HPC project under UKRI-EPSRC grant EP/W035731/1.

## REFERENCES


[1] Weronika Filinger, Julia Mullen, and Jeremy Cohen. 2024. Developing HPC Learning Pathways: Challenges and Recommendations. In *Practice and Experience in Advanced Research Computing* (Providence, RI, USA) *(PEARC '24)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3626203.3670575

[2] Weronika Filinger, Julia Mullen, Jeremy Cohen, Ann Backhaus, and Samantha Wittke. 2024. Building HPC Learning Pathways: Understanding our Community. In *Practice and Experience in Advanced Research Computing* (Providence, RI, USA) *(PEARC '24)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3626203.3670513

[3] Julian Kunkel, Weronika Filinger, Christian Meesters, and Anja Gerbes. 2020. The HPC Certification Forum: Toward a Globally Acknowledged HPC Certification. *Computing in Science and Engineering* 22, 4 (2020), 110–114. https://doi.org/10.1109/MCSE.2020.2996073

# $T^3$-CIDERS: Fostering a Community of Practice in CI- and Data-Enabled Cybersecurity Research Through A Train-the-Trainer Program

Wirawan Purwanto
Old Dominion University
wpurwant@odu.edu

Mohan Yang
Texas A&M University
mohanyang@tamu.edu

Peng Jiang
University of Nebraska Omaha
pjiang@unomaha.edu

Masha Sosonkina
Old Dominion University
msosonki@odu.edu

Hongyi Wu
The University of Arizona
mhwu@arizona.edu

## ABSTRACT

We present a training program named $T^3$-CIDERS, the "Train-The-Trainer approach to fostering cyberinfrastructure (CI)- and Data-Enabled Research in CyberSecurity." $T^3$-CIDERS is a train-the-trainer program for advanced cyberinfrastructure (CI) skills that is designed to be synergistic with research, teaching, and learning activities in cybersecurity and cyber-related disciplines. The participants, termed "future trainers" (FTs), are trained in effective instructional design and CI hands-on materials from "DeapSECURE", developed in a previous CyberTraining program. $T^3$-CIDERS aims to enhance cybersecurity research and education through broader adoption of advanced CI techniques such as artificial intelligence, big data, parallel programming, and platforms like high-performance computing (HPC) systems. $T^3$-CIDERS includes pre-training, a weeklong summer institute, ongoing learning engagements, and local training activities. The FTs conduct local training tailored to the needs at their respective home institutions. They will also develop a new CI training module (called "Module X") based on the observed common needs in the cybersecurity research community. Community building is integral to $T^3$-CIDERS as its overarching goal. The first cohort of FTs who took the 2024 summer institute comprises faculty members, researchers, and students representing multiple states.

## KEYWORDS

Cyberinfrastructure, Cybersecurity, Train-the-trainer, Pedagogy, HPC, Parallel Computing, Big Data, Machine Learning

## 1 INTRODUCTION

Cyber-enabled technologies have revolutionized the 21st century society by increasing efficiency, convenience, and productivity in every walk of life. With the increasing volume and sophistication of cyberattacks, researchers, engineers, and practitioners heavily rely on advanced cyberinfrastructure (CI) techniques to assess cyber risks, identify and mitigate threats, and achieve defense in depth. CI capabilities such as artificial intelligence (AI), machine learning, big data, as well as advanced CI platforms, e.g., the cloud and high-performance computing (HPC) have become integral to support the state-of-the-art cybersecurity research and implementations.

Despite the widespread availability of CI and its importance in strengthening the security of cyber systems, there is still a significant gap in the preparedness of the cybersecurity workforce to leverage CI. In research, the lack of preparedness in CI often hampers, or frustrates, students from entering research areas that rely heavily on CI, such as AI/cybersecurity intersection and privacy-preserving machine learning techniques. To fill this gap and broaden CI adoption in research and education in cybersecurity, we developed the "Train-The-Trainer Approach to Fostering Cyberinfrastructure (CI)- and Data-Enabled Research in CyberSecurity" ($T^3$-CIDERS) training program [9]. The goal of $T^3$-CIDERS is to accelerate state-of-the-art research and development in cybersecurity and related fields by (1) preparing competent trainers to broaden the utilization of advanced CI in these disciplines; and (2) fostering a CI-enabled cybersecurity research community of practice. As a train-the-trainer program for advanced CI, $T^3$-CIDERS is designed to be synergistic with research, teaching, and learning activities of its participants.

The $T^3$-CIDERS program is a year-long engagement for each cohort of participants, referred to as *Future Trainers* (FTs). The FTs are chosen from faculty, researchers, students, and practitioners with interests in cybersecurity and cyber-related fields. They will receive training on the basic skills in key CI areas and effective pedagogical methods to further promulgate these CI skills to others in their academic communities. The FTs are clustered into units ("FT units"), each of which consists of one faculty or researcher and one or two students. As part of the requirements of the program, these FT units will collaboratively prepare and conduct CI training activities at their respective institutions. The activities of $T^3$-CIDERS encourage collaborations and community building among FTs as a CI- and cybersecurity-focused community of practice.

$T^3$-CIDERS leverages and builds upon "DeapSECURE" introductory CI training modules [5–8], which covers fundamental CI topics: HPC, Big Data, Machine Learning, Deep Learning, Cryptography, and Parallel Programming. As a minimum prerequisite to participate in $T^3$-CIDERS, FTs need to have working knowledge of computer programming (in any language); preferably, they should also


https://doi.org/10.22369/issn.2153-4136/16/1/4

have experience working with one or more of the aforementioned CI topics.

## 2 OVERVIEW OF THE T$^3$-CIDERS PROGRAM

### 2.1 Program Structure

T$^3$-CIDERS is designed to train and engage the FTs through be a year-long program. This comprehensive approach consists of six major phases:

**Phase 1: Pre-training** — A month-long, self-paced, virtual training on CI skills baseline, covering HPC, Unix shell, Jupyter, and Pandas. This phase ensures that all participants have a foundational understanding of essential CI concepts and tools.

**Phase 2: Summer Institute** — An intensive, week-long, in-person training that covers an overview of CI techniques (Machine Learning, Deep Learning, Cryptography, and Parallel Programming), as well as pedagogy/training methodology which includes educational foundations, assessment, hands-on lesson planning, and simulated teaching practice. It also includes discussion panels with experts aimed to broaden the FTs' perspective on pressing issues in cybersecurity practice and education.

**Phase 3: Monthly Learning Engagement** — After the summer institute, the FTs will engage in year-long virtual meetings (about once a month) to further their CI learning journey, share experiences in CI-augmented teaching and/or research, connect with peers and cyber experts. This ongoing engagement fosters an inclusive and supportive community of practice and encourages continuous professional development.

**Phase 4: Local Training** — As part of their enrollment with T$^3$-CIDERS, FT units are required to conduct local training activities that are tailored to their academic communities, ensuring the dissemination of CI knowledge and skills.

**Phase 5: K-12 Outreach** — FT units have the opportunity to spread awareness of cybersecurity, research in cybersecurity, and the role of CI to support cyber research in local K-12 schools, inspiring students to be the next generation of cybersecurity professionals. This phase is optional, but FTs are strongly encouraged to reach out to the communities around them in this way.

**Phase 6: "Module X"** — A community-driven initiative where FTs and the project team will collaboratively design and develop new CI training module(s) to address additional training gaps in the preparation of students for cybersecurity research.

The pre-training and summer institute serve as a short-term "boot camp" to the FTs. The monthly learning engagement is an effort to foster the connections among FTs that were established during the summer institute and provide continuous learning opportunities. The local training activities, K-12 outreach, and "Module X" represent outcomes and sustained activities facilitated by the T$^3$-CIDERS program.

### 2.2 CI Skills Baseline

T$^3$-CIDERS equips FTs with understanding of baseline CI skills that can be further taught to their own learners, who most likely are novice to CI. The baseline skills are provided by adapting the DeapSECURE's lesson modules[6] covering HPC, Big Data, Machine Learning, Deep Learning, Cryptography, and Parallel Programming. DeapSECURE lessons emphasizes hands-on introduction to these topics and incorporates research datasets and use cases to provide strong relevance to cybersecurity. Table 1 shows the six lesson modules, the key contents of each lesson, the software tools introduced, and the target cybersecurity application used through-out the lesson. DeapSECURE modules integrate the principle of authentic learning [3]: Certain challenges (problems), carefully selected within the target applications, motivate the introduction of the CI methods and/or techniques. The learners are guided through-out a series of hands-on activities (similar to Carpentries-style live coding method [4]) to leverage these methods and address the posed challenges. In this way, CI topics are introduced in a way that is relevant to cybersecurity applications.

The DeapSECURE lessons were originally designed for non-degree, non-credit, bootcamp-style training targeting novice learners who may not know the CI topics at all. The lessons provide a gentle onramp by covering only the most critical concepts and techniques that novices must learn to become competent learners and practitioners on their own. The lesson materials include Carpentries-style online e-textbooks, Jupyter notebooks, and hands-on files (sample scripts and templates, datasets, etc.). as explained further below. In addition to providing baseline CI skills, these lessons also serves as teaching resources, from which the FTs can draw and/or adapt to suit their own learners' needs.

### 2.3 Training Infrastructure

The T$^3$-CIDERS program leverages the following platforms as its infrastructure: (1) ODU's "Wahab" HPC cluster for the hands-on learning environment; (2) Canvas learning management system to present the lesson materials to the FTs; (3) Discord instant messaging platform, for discussions among FTs and with the training team; (4) LinkedIn and X (Twitter) as the social media for public outreach and engagement; (5) Qualtrics for data collection and surveys; (6) Gitlab for collaborative hands-on lesson development.

## 3 THE 2024 COHORT OF FUTURE TRAINERS

The 2024 summer institute took place in July 29–August 2 at the Norfolk campus of Old Dominion University. It drew together 10 faculty/researchers and 12 students with varied backgrounds. These FTs represented multiple states (Washington, Texas, Virginia, and New York), reflecting the program's broad geographic reach.

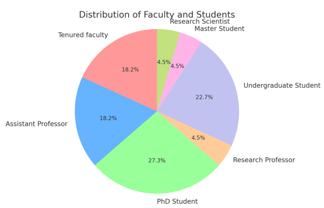
### 3.1 Participant Background

Figure 1 highlights the distribution of participants by role and background. As shown in Figure 1a, 36% of the total participants were tenured or assistant professors, while 9% held positions as research scientists or professors. Among the student participants, 31% were graduate students, and 23% were undergraduate researchers who were attracted to the program for its academic and professional opportunities. As displayed in Figure 1b, 60% of participants identified as male, 30% as female, 10% as others.

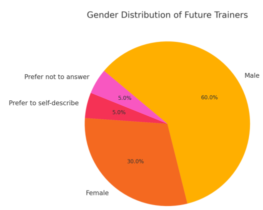
The major distribution of participants (Figure 1c) illustrates strong alignment of our proposed program to the research interests of our participants. Of the 22 participants, 8 are majoring in

**Table 1: Currently available DeapSECURE's CI lesson modules**

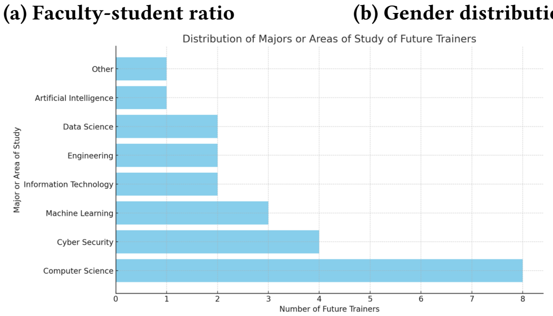
| Module Name | Description | Toolkits | Target Application |
|---|---|---|---|
| Introduction to HPC | Intro to HPC platform and basic parallel processing | UNIX shell commands, SLURM job scheduler | Analysis of a spam collection |
| Dealing with Big Data | Processing, cleaning, analyzing, and visualizing big data sets | Pandas, Matplotlib, Seaborn | Data preparation and smartphone app classification using Sherlock dataset |
| Machine Learning | Workflow of building, training and validating machine learning models | Scikit-learn | |
| Deep Learning using Neural Networks | Building, training and validating neural networks | KERAS, TensorFlow | |
| Cryptography for Privacy-Preserving Computation | Symmetric cryptography; homomorphic encryption for privacy-preserving computation | AES-Python [10], PyCryptodome [1], Python-Paillier [2] | Image encryption with Paillier crypto |
| Parallel & High-Performance Programming | Parallel programming with MPI | mpi4py | |




**(a) Faculty-student ratio**




**(b) Gender distribution**




**(c) Major distribution**

**Figure 1: The statistical overview of 2024 cohort**

Computer Science, 4 in Cybersecurity, and another 4 in Artificial Intelligence/Machine Learning (AI/ML). Additionally, there are 2 participants each in Data Science, Engineering, and Information Technology. The mutual focus in CI- and Cybersecurity-related topics serves as the cornerstone in building a community of practice of next-generation cybersecurity researchers and scholars in leveraging CI technologies to accelerate state-of-the-art research and development in cybersecurity and cybersecurity-related fields.

## 3.2 Program Details

*3.2.1 Pre-Training.* Prior to the summer institute, FTs were onboarded to Canvas as the platform to host lesson materials, and to Discord (a group messaging platform used to facilitate discussions, ongoing collaborations, and community building). Each FT was required to complete the two pre-training modules, which helped them to get familiar with the HPC, as well as the prerequisite knowledge for the summer institute. Two virtual Q&A sessions were held to answer questions related to access to the ODU HPC cluster, and the technical challenges encountered in the pre-training modules.

*3.2.2 Summer Institute.* The Summer Institute for the 2024 Cohort was designed to provide an intensive, hands-on learning experience, combining technical lectures with practical activities, pedagogical training, and collaborative and interactive lesson design sessions. Participants engaged with well-designed cutting-edge topics while applying their knowledge through hands-on sessions and collaborative discussion. Additionally, pedagogical training is involved throughout the week, emphasizing the importance of effective teaching in the technical fields. Each FT unit has their own studio time in the afternoon to prepare a draft lesson plan for a local training activity that will be further developed and executed in the 2024–2025 academic year. On the last day of the summer institute, each FT unit presents a 15-20 teaching demo to showcase the lesson plans they built during the studio times, coupling with the pedagogical methods introduced in the summer institute.

*3.2.3 Professional Developments.* Throughout the week-long workshop, we organized two panel discussions and a field trip to bridge the gap between academic learning and real-world cybersecurity practices. The first panel featured industry partners from Hampton Roads, who provided a showcase of cutting-edge cybersecurity techniques and technologies. They discussed the challenges faced in real-world cybersecurity operations and highlighted advanced CI techniques being used to solve these problems. This session offered participants valuable insights into the practical application of their skills in industry settings. The second panel brought together high school teachers who had participated in the NSA GenCyber Summer Camps hosted at ODU in the past two years. These educators shared their experiences teaching cybersecurity concepts to their own students, offering a unique perspective on K-12 education. Our FTs engaged in productive discussions with the teachers, these exchanges provided our participants with a deeper understanding of how cybersecurity education is being approached at the K-12 level and the opportunities for integrating emerging technologies into the curriculum. The field trip to Old Dominion

University's ITS data center offered participants a firsthand look at the infrastructure supporting high-performance computing and cybersecurity efforts, allowing them to observe real-world applications of the concepts discussed throughout the workshop.

*3.2.4 Monthly Engagement.* After the conclusion of the summer institute and continuing through the end of each cohort year (April 2025), the project team will maintain regular engagement with the FTs through monthly virtual events. The initial engagements will primarily focus on lesson plan development. The project team will monitor progress from each FT unit on their lesson development and local training planning, providing feedback and support to help them refine their plans until the local training is implemented. From 2025, monthly research webinars will be hosted for both FT units and the broader community, allowing faculty to present their research findings. Additionally, we plan to invite external speakers to discuss recent advancements in HPC, education, cybersecurity research, etc, extending the influences to a wider community.

### 3.3 Assessment and Impacts

We plan to conduct a multiple-level and multiple-phased evaluation approach before, during, and after the program. The evaluation will include both formative and summative plans with a mix-method design for data triangulation. T$^3$-CIDERS employs multiple methods such as pre-survey, daily reflection and feedback form, post-institute survey, interviews, and final (post-) survey to assess the program's contents and effectiveness, FTs' attitude, learning, and teaching plan. (Since the cohort is still ongoing as of the time of writing, the evaluation has been conducted only partially.)

*3.3.1 Before the summer institute.* A pre-survey was sent out to the FTs and collected their feedback before the first day of the summer institute. It includes questions to identify FT's demographic information, characteristics, reactions toward the program, motivation, needs, and attitudes that need to be considered in designing and redesigning the program. It also provides insights into FT's target learners and helps us further tailor the content and instruction.

*3.3.2 During the summer institute.* During the summer institute, observational notes were taken to track the FTs' behavior, interactions with peers, and engagements. This helped in assessing participant involvement and learning application. FTs were encouraged to submit daily reflections, documenting their experiences and thoughts during the program. These sessions help ensure that learning is an iterative process, with participants and our project team, continually improving their understanding and teaching methods.

*3.3.3 After the summer institute.* Following the completion of the summer institute, a post-institute survey was launched to collect participants' reaction with the training they received thus far. Furthermore, after each FT unit conducts their local training or outreach, a survey will be distributed to their attendees to gather feedback and assess the effectiveness of their training efforts. At the end of the cohort period, a post-survey will be conducted to assess participants' satisfaction with the entire program, their views on the relevance of the content, and their confidence in applying the skills they acquired. These results will be compared with the pre-survey to identify any shifts in attitudes and technical abilities that occurred throughout the program. As each cohort year comes to a close, we will begin examining the Return on Expectations (ROE). This will involve conducting semi-structured interviews with participants, the program coordinator, and training mentors to gather insights into their experiences, the program's structure, activities, content, and instructors.

## 4 CONCLUSION

T$^3$-CIDERS is a promising initiative that addresses the growing need for advanced CI expertise in cybersecurity research and education. By providing comprehensive training, fostering a supportive community of practice, and promoting sustained engagement, the program is effectively preparing the next generation of cybersecurity professionals to tackle emerging challenges. The positive feedback from the 2024 cohort, the anticipated local training activities during the 2024–2025 academic years and beyond, as well as the ongoing development of "Module X" demonstrate the program's potential for long-term impact.

## 5 ACKNOWLEDGMENTS

Funding for this program comes from the U.S. National Science Foundation CyberTraining grants #2320998 and 2320999. The authors graciously acknowledge the support from ODU School of Cybersecurity for hosting the 2024 summer institute, and ODU Research and Cloud Computing group for the use of their Wahab HPC cluster for conducting the CI training activities. Dr. Nitin Sukhija from Slippery Rock University of Pennsylvania reviewed the pre-training materials.

## REFERENCES


[1] Simon Arneaud, Nevins Bartolomeo, and Thorsten E. Behrens *et al.* [n. d.]. PyCryptodome. https://pycryptodome.org/

[2] CSIRO's Data61. 2013. Python Paillier Library. https://github.com/data61/python-paillier

[3] Jan Herrington and Ron Oliver. 2000. An Instructional Design Framework for Authentic Learning Environments. *Educational Technology Research and Development* 48, 3 (2000), 23–48. https://doi.org/10.1007/BF02319856

[4] Alexander Nederbragt, Rayna Michelle Harris, Alison Presmanes Hill, and Greg Wilson. 2020. Ten quick tips for teaching with participatory live coding. *PLoS Comput. Biol.* 16 (2020), e1008090. Issue 9. https://doi.org/10.1371/journal.pcbi.1008090

[5] Wirawan Purwanto, Bahador Dodge, Karina Arcaute, Masha Sosonkina, and Hongyi Wu. 2024. DeapSECURE Computational Training for Cybersecurity Students: Progress Toward Widespread Community Adoption. *The Journal of Computational Science Education* 15 (2024), 2–9. Issue 1.

[6] Wirawan Purwanto, Issakar Doude, Yuming He, Jewel Ossom, Qiao Zhang, Liwuan Zhu, Jacob Strother, Rosby Asiamah, Bahador Dodge, Orion Cohen, Masha Sosonkina, and Hongyi Wu. 2022. DeapSECURE Lesson Modules. https://deapsecure.gitlab.io/lessons/

[7] Wirawan Purwanto, Yuming He, Jewel Ossom, Qiao Zhang, Liuwan Zhu, Karina Arcaute, Masha Sosonkina, and Hongyi Wu. 2021. DeapSECURE Computational Training for Cybersecurity Students: Improvements, Mid-Stage Evaluation, and Lessons Learned. *The Journal of Computational Science Education* 12 (2021), 3–10. Issue 2.

[8] Wirawan Purwanto, Hongyi Wu, Masha Sosonkina, and Karina Arcaute. 2019. DeapSECURE: Empowering Students for Data- and Compute-Intensive Research in Cybersecurity through Training. In *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning) (PEARC '19)*. ACM, New York, NY, USA, Article 81, 8 pages. https://doi.org/10.1145/3332186.3332247

[9] Wirawan Purwanto, Mohan Yang, Peng Jiang, Masha Sosonkina, Hongyi Wu, Jael Perales, and Kayla Curtis. 2024. T3-CIDERS Project Website. https://sites.wp.odu.edu/t3-ciders/

[10] Bo Zhu. 2015. A pure Python implementation of AES. https://github.com/bozhu/AES-Python.git

# The Use of High-Performance Computing Services in University Settings: A Usability Case Study of the University of Cincinnati's High-Performance Computing Clusters

Mahmoud Junior Suleman, MSc.
University of Cincinnati
sulemamr@mail.uc.edu

Amy Latessa, PhD.
University of Cincinnati
latessak@mail.uc.edu

Shane Halse, PhD.
University of Cincinnati
halsese@mail.uc.edu

Jess Kropczynski, PhD.
University of Cincinnati
kropczjn@mail.uc.edu

## ABSTRACT

High-performance computing (HPC) clusters are powerful tools that can be used to support a wide range of research projects across all disciplines. However, HPC clusters can be complex and difficult to use, limiting their accessibility to researchers without a strong technical background. This study used a mixed method to investigate ways to make HPC clusters more accessible to researchers from all disciplines on a university campus. A usability study of 19 university researchers was conducted to understand the needs of HPC users and identify areas where user experience could be improved. Our findings reveal the need to build a customized graphical user interface HPC management portal to serve users' needs and invest in workforce development by introducing an academic credit-based High-Performance Computing Course for students and partnering with other faculties to introduce special programs, e.g., Student Cluster Competitions which would draw more student interest.

## KEYWORDS

High Performance Computing, Computing Resources for Academic Researchers, Usability, Think Aloud, Accessibility

## 1 INTRODUCTION

High-performance computing (HPC) has become an essential tool to advance scientific discovery over the last two decades, and it's an area where researchers actively create larger systems to accommodate new modes of scientific discovery with complex workflows. Advocating for large-scale scientific programming and HPC have become more essential to achieving national goals considering the discoveries made by academic researchers and industry professionals who contribute significantly to national development and further increase the importance of adequately educating the next generation [14]. National labs, academic institutions, and industries need scientists and staff who understand high-performance computing (HPC) and the complex interconnections across individual topics in HPC. However, domain science and computer science undergraduate programs need to provide more educational resources and are far from conveying the interdisciplinary and collaborative nature of the HPC environment [10].

Academic institutions that are research (R1) based invest resources to acquire and build centers to manage HPC Systems on campus to enable scientists to improve and foster their research capabilities. Over the past few years, some HPC centers have encountered challenges in encouraging scientists to use HPC resources effectively for their research. There is an HPC expertise and knowledge gap because very few educators have the skill set to use HPC Systems made available on campus, making it difficult for non-stem students to learn and access available HPC resources [17].

HPCs are used across various fields, making it challenging for students to understand their applications. Launching HPC applications is complex, requiring multiple components and specialized skills [30]. Using HPCs effectively demands a broad knowledge base and significant practice. Typical HPC curricula often focus on STEM students, covering topics like computer literacy, programming, parallel computing, version control, and debugging [31]. Students from non-computing departments need extra training to use HPCs effectively, hindering their research progress. To improve the accessibility and usability of local HPC resources, we conducted a usability study to determine the factors hindering their adoption by students and faculty. Toward this goal, this paper aims to address the following overarching research question: *How can HPCs be made more accessible for use across disciplines in institutes of higher education?*

## 2 BACKGROUND

### 2.1 Evolution of High-Performance Computers

Over the years, there has been rapid growth in computing and communications technology; the past decade has witnessed a proliferation of robust parallel and distributed systems and an ever-increasing demand for the practice of high-performance computing (HPC). HPCs have moved into the mainstream of computing. They have become a key technology in determining future research and development activities in many academic and industrial branches. They must cope with very tight timing schedules when solving large and complex problems [11].


https://doi.org/10.22369/issn.2153-4136/16/1/5

In 1985, the National Science Foundation established a partnership between five research centers: the San Diego Supercomputer Center (SDSC) at the University of California San Diego, the Pittsburgh Supercomputer Center (PSC) at the University of Pittsburgh, the National Center for Supercomputing Applications (NCSA) at the University of Illinois Champagne-Urbana, the Cornell Theory Center at Cornell University [29]. In the last decade, there has been significant advancement in the reliability and performance of computing elements such as processors, disks, and network devices. Computational power has increased in desktops and laptops by the availability of processors; these reliable and robust off-the-shelf computational elements have also spurred a new generation of high-performance computing systems [4].

## 2.2 The Value of High-Performance Computing Centres on University Campuses

Cyberinfrastructure includes computing systems, data storage, instruments, repositories, visualization, and people, connected by high-speed networks for scholarly innovation as defined by Stewart et al. [20]. Research shows a campus supercomputer positively impacts research output; Apon's work analyzed the ROI of cyberinfrastructure, finding that HPC investment benefits research productivity in fields like Chemistry, Physics, and Civil Engineering [1]. Apon's studies show statistical analyses of cyberinfrastructure's impact [2]. Indiana University calculated the ROI of 3 cyberinfrastructures, finding cost savings compared to commercial cloud alternatives [19]. Scrivner developed XDMOD-VA, a visualization plug-in for XDMOD, to present value proposition metrics for HPC centers [18]. This demonstrates the cost-effectiveness of supercomputers for most educational institutions.

## 2.3 Usability of High-Performance Computing in STEM and other disciplines

High-performance computing (HPC) is increasingly essential in higher education, HPCs offer new opportunities but pose challenges in teaching. Educators and researchers need to adopt innovative teaching approaches to leverage HPC effectively. Computing Education faces new demands, requiring motivated students and adaptive learning systems like SAIL- a System for Adaptive Interest-based Learning [9]. The Higher Education System needs comprehensive training to address the limitations of integrating technology into teaching and the constraints of using supercomputers. HPC is crucial for STEM and other disciplines despite challenges like limited experience and high costs [6]. Supercomputing education fosters innovation and creates a new generation of professionals. It's essential for various fields that use computational tools. Many educational institutions are integrating computer science concepts into basic training, raising questions about the importance of supercomputing education for students' careers [8]. ICT integration is crucial for modernizing education and transitioning to a knowledge society. Supercomputers will be vital for solving complex data problems in STEM and non-STEM disciplines [6].

## 2.4 Demystifying High-performance computing for non-Linux users: Introducing Remote computing desktops.

Implementing web-based portals for end users to manage and use HPCs has been remarkable. It has improved users' time navigating around HPCs to complete a task. This is not only limited to the time HPC centers spend training new users, especially non-Linux users, to adapt to the Command Line Interface (CLI) to complete their tasks. Not only have end users been relieved, but system admins and research computing facilitators have also been relieved; these portals give them a one-stop shop to manage these clusters and easily support end users. In this era of computing, more emphasis on graphical and intuitive interfaces is essential to clear barriers away from researchers from all respective fields to use computing resources with ease and not deal with rigid legacy system designs. Mastering the CLI is an essential skill, but it becomes intimidating for researchers who want to use the High-Performance computing resources of a university [23].

According to [25], Over 75% of users say that desktop services are either moderately or extremely important for their ability to use HPC resources. During training sessions at Indiana University, they observed that researchers who were constant Linux users who did not know about HPCs were enthusiastic about learning and trying out the HPC system and wanted to know more about its capabilities and features. However, they mostly complained about the Job Schedulers, e.g., SLURM and longer wait times for their jobs to be in the queue. Non-Linux researchers/users have more reservations about using HPC systems; most had no choice but to learn the command line and how to eventually use the job schedulers due to the nature of their research and the size of their data sets. In Summary, most HPC Users were demotivated to use the HPC system for their research. This concluded that the command line and the batch job scheduling are significant barriers for potential HPC users [25].

*2.4.1 Web-based Portals for Managing High-Performance Computing Clusters.* Web-based portals for HPCs were introduced in the 1990s with Java applets, making it easy to access HPC resources remotely using a variety of use cases. This identified the key features and functional and non-functional requirements of these portals [5]. Web-based management portals aid in reducing barriers that limit the adoption of high-performance computing (HPC), which addresses an important challenge affecting the STEM community. Through the Cyber-Infrastructure grant, the NSF has supported Higher Education institutions and National Supercomputing Labs to develop web-based HPC management portals to help researchers with little computer science skills adapt and foster the use of HPCs in their research [26].

Some benefits of having Web-based portals include but are not limited to:

(1) Increasing the use of HPC resources among disciplines that are not well represented in the community but desire the need for HPC resources.
(2) It is also beneficial to HPC stakeholders because it is easier to adopt advanced features to monitor and visualize system needs and upgrades.

(3) This creates a working model for training and workforce development to expand the cyberinfrastructure space for computer science. In 2016, the first version of Open On-demand launched the Open On-demand project, funded through the NSF Cyberinfrastructure grant ([27]) to make accessing HPC resources easier. The Open on Demand Project was an Open-source software project hosted by the Ohio Super Computing Center that enables HPC centers to install and deploy advanced web and graphical interfaces for their users [12].

## 2.5 Case Study: Advanced Research Computing Centre at the University of Cincinnati

The University of Cincinnati's Advanced Research Computing (ARC) Centre offers a readily accessible hybrid CPU/GPU computing cluster, supporting computational and data science researchers while developing a highly competitive workforce amongst the university community.

[3]. ARC partners with researchers to utilize the core HPC services and resources suite. With ARC's resources, researchers advance theoretical knowledge and expand the realm of discovery, generating leading-edge research and applications suitable for innovation and commercialization in line with the University of Cincinnati's "Next Lives Here" strategic direction. The center has a sustainable high-performance computing (HPC) infrastructure with technical support that leverages HPC services to accelerate the time to discovery and enables sophisticated and increasingly realistic modeling, simulation, and data analysis, which helps to bridge users to the local, regional, and national HPC ecosystem [3].

# 3 METHODS

To understand user needs at the High-Performance Computing Center, we combined methods used by [16]. We conducted an online survey, qualitative think-aloud interviews, and a heuristic evaluation. These methods provided comprehensive feedback on user needs and design effectiveness. The survey and interviews collected user insights, while the heuristic evaluation assessed adherence to usability principles. All methods were IRB-approved.

## 3.1 Research Participants

The selection of participants in this research was guided by the study's nature and desired outcomes, as outlined in [22]. Our target population for the survey were students and faculty members of the University of Cincinnati, especially current and future non-Linux researchers who intend to use or use the resources of the High-Performance Computing cluster.

The targeted survey audience was the Science, Technology, Engineering, and Mathematics (STEM) students and faculty members. We chose not to limit our audience to a particular group and included all potential users who were interested in using the cluster regardless of their technical skills or background. This diversified our participants, which gave us more information to feed into our future design.

**Participants were recruited in two stages or categories:**

- **Current University Researchers:** HPC researchers at the University of Cincinnati, both current and potential users.
- **Students (Undergraduate, Masters, PhD):** STEM students from various departments, including Engineering, Information Technology, Medicine, Art, and Planning, as well as non-STEM students.
- **Faculty:** Faculty teaching courses requiring HPC resources and principal investigators leading research labs.
- **Staff:** University staff using or potentially using HPC resources for departmental work.

## 3.2 Data Collection

For quantitative data, a 27-question survey was shared with participants by casting a wide net to all the users of the cluster, either past or present. These questions were carefully designed to capture feedback from highly skilled and novice users with little skills for genuine feedback on their needs and issue severity.

For Qualitative data, we analyzed and interpreted data, identified themes, and understood the study's phenomenon. We collected cognitive and constructive feedback through a think-aloud activity. [21].

*3.2.1* ***Think-Aloud Activity.*** The method of collecting think-aloud data involves participants spontaneously verbalizing words that come to mind as they complete an activity [28]. This makes it essential to conduct usability testing, as it allows users to verbalize their thinking as they use a new system. This allows evaluators to infiltrate the minds of users and acknowledge individual differences.

To further understand users' needs in-depth, We selected five users out of the respondents of the Survey who voluntarily opted in, three of whom were super users, and two non-superusers. We opened the conversation for users to have cognitive thinking, which allowed them to expand their thoughts while using the portal; we further asked questions in two categories, for Superusers and Non-Superusers:

- **Superusers** were asked what connection methods to the cluster they preferred and whether they needed an intuitive Graphical User Interface portal. Also, what were their needs for the graphical user interface portal? Furthermore, what could be done to improve the usability of the portal?
- **Non-Superusers**, mostly Non-Linux users, were asked about their concerns with the HPC portal and what they expected from the HPC cluster to avoid using the CLI.

In order not to influence users' thoughts and have their ideal participation while offering their honest thoughts spontaneously, we did not prompt users during the conversation; however, to avoid too much silence or stiltedness, we had signs with important information, e.g., keep talking, Go on, that is interesting, etc. For pre-orientation and housekeeping, we dedicated the first fifteen minutes of the session to explain how the think-aloud activity would be facilitated, and an overview of the 'think-aloud' method was provided in a pre-interview facilitator script [28].

Our participants were users who were allowed to reflect more on the best options that worked for them when using the cluster and why they preferred them. After that, we focused on the features of each connection method to understand their perceptions and whether they had HPC experience from their previous organizations or personal use. We also observed and noted their behaviours in real time.

*3.2.2 Heuristic Evaluation.* In 1990, a published article by web usability pioneers Jakob Nielsen and Rolf Molich defined Heuristic Evaluation as a set of principles used in "Improving a Human-Computer Dialogue" [15]. Heuristics evaluation is a process to systematically determine or certify a design or product's usability; researchers review the product's interface and compare it with other usability principles, and the results are accompanied by recommendations to improve the current system [7].

We based our Heuristic evaluation on the ten most fundamental principles that were given by the Nielsen Norman Group, which are listed in 1.


**Figure 1: Nielsen and Molich's 10 User Interface Design Heuristics [7].**

A group of examiners evaluated the University of Cincinnati's High-performance Computing cluster interface's user interface according to a set of heuristics rules based on 1.

We observed users' actions and interpreted their connection to usability challenges. We recorded the evaluation results and the evaluator's comments about the user interface; however, there was no need to interpret the evaluator's actions [? ]. We provided a severity rate scale of 5 for each violation to measure the severity of design problems; at the end of the heuristic evaluation, we rated these design problems accurately to reflect the responses and evaluation of our participants. The schema for the severity ratings is listed below :

- **0** - don't agree that this is a usability problem
- **1** - cosmetic problem
- **2** - minor usability problem
- **3** - Major usability problem; important to fix
- **4** - usability catastrophe; imperative to fix

*3.2.3 Data Analysis .* Due to the nature of our study, we employed a mixed methods approach to collecting data, which involved both qualitative (survey) and quantitative (Think-aloud activity) methods. According to [24] mixed methods offer confirmation and complementarity feedback that "enables the researcher to simultaneously answer confirmatory and exploratory questions, and therefore verify and generate theory simultaneously. We found similarities between topics discussed by the participants, which were generated as themes by performing a Qualitative Thematic analysis.

We employed a mixed methods approach to collecting data and focused more on participant engagements, which influenced our research: both positive and negative feedback, participants describing experiences with using the HPC Cluster during our conducted aloud activity, and ideas to design an intuitive, customized graphical user Interface.

After manually reviewing collected data, which included surveys, transcribed audio, and videos, the data were categorized into codes and potential themes. For thematic data analysis by [13], we grouped the themes into broad sub-themes in relation to the research questions and analyzed the data with regard to the identified themes related to the research questions.

## 4 RESULTS

In this section, we present our research findings and in-depth analysis of participant recommendations.

### 4.1 Description of Users and Results from the Survey

The survey had 21 responses; two were voided, and the remaining 19 responses were from 11 super users and 8 non-super users. Most users were graduate students, followed by faculty and staff. Superusers had more experience with HPC clusters than non-superusers. Most users used the cluster once or twice a week for research projects. Key reasons for using the cluster included research projects, personal projects, grant requirements, thesis/dissertation, and class projects. Accessibility, data security, and cost-effectiveness were major factors in choosing the on-campus cluster. 47% of respondents didn't need UNIX/LINUX training, while 38% did. 42% used FTP for data transfer, 25% used an API, and 16.5% used Globus. Details about the participant demographics are shown in Table 5.

### 4.2 Identified Codes and Themes from the Think-Aloud activity

For our Thematic Coding analysis, as suggested by [16], we analyzed data from the Think aloud activity and created a thematic coding scheme to identify common patterns in the experiences of

Table 1.1: Participant Demographics and Descriptive Reported by Expertise

| Descriptor | All Participants | Super Users | Novice Participants |
|---|---|---|---|
| *N* | 19 | 11 | 8 |
| No. of Years at University | | | |
| 1 – 3 | 9 | 2 | 5 |
| 3 – 5 | 4 | 3 | 3 |
| 5 – 10 | 6 | 6 | 0 |
| Role at University | *n* (%) | *n* (%) | *n* (%) |
| Faculty Researcher or PI | 3 (1.5) | 3 (10.0) | 0(0) |
| Masters Student | 4 (2.2) | 1 (2.5) | 3(7.5) |
| PhD Student | 11 (5.78) | 7 (6.3) | 5(3.7) |
| Staff | 1 (0.52) | 1 (10.0) | 0(0) |
| Previous Use of HPC | *n* (%) | *n* (%) | *n* (%) |
| Never Used | 0(0) | 0 (0) | 0(0) |
| Seasonal User (several uses total) | 1 (0.54) | 1 (10.0) | 0(0) |
| Once per Month | 3 (1.59) | 2 (6.3) | 1(3.7) |
| Two or Three Times per Week | 10 (5.26) | 5 (5.0) | 5(5.0) |
| Everyday | 5 (2.6) | 3 (6.0) | 2(4.0) |
| Type of Project | | | |
| Personal Project | 4 | 3 | 1 |
| Class Project | 1 | 0 | 1 |
| Thesis or Dissertation | 4 | 3 | 1 |
| Internally Funded Research | 7 | 3 | 4 |
| Externally Funded Research | 3 | 2 | 1 |

**Figure 2: Participant Demographics and Descriptive Reported by Expertise**

our participants using the HPC Cluster; after identifying key themes and concepts mentioned by our participants via the interview, we searched for patterns to in these codes after reaching saturation point. We identified key themes/codes from our participants, which we explained in the subsections below. There were five interviewees for the Think-aloud activity, and to be more specific in identifying them, we referenced participants as P1, P2, P3, etc. Three superusers and two non-supers users participated in the Think-aloud activity. However, to incorporate qualitative responses from our survey participants, we referenced them as S1, S2, S3, etc.

*4.2.1 Theme 1: Connection Methods to the HPC Cluster .* For theme (1), we explored user preferences regarding connection methods (e.g., CLI, SSH, Open On-Demand) to the HPC Cluster. Generally, users relied on instructions provided by the Advanced Research Computing Center (ARCC) [3] through their website or the Research Computing Facilitator. The Facilitator served as a crucial resource for users facing challenges or requiring direct information about the cluster (e.g., specific software, GPU/CPU allocations, job queues).

This reliance is exemplified by the following comment from a participant (P1): *"It was OK because I think I was following some of the instructions. You see, IT had some instructions on that. About like first users. So it was. I had thought that that was pretty smooth."*

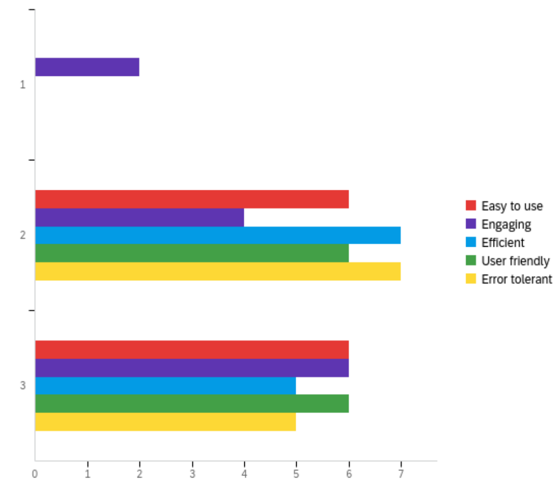



**Figure 3: A Visualized report of respondents ratings of the current HPC Web Portal**

Users found the Advanced Research Computing Center's website provided sufficient instructions for connecting to the cluster. On-campus connections were generally smooth due to the University's network resources. However, off-campus access via VPN was disliked by some (P1, P2, P3) due to perceived hindrance to collaboration, despite its security benefits.

All users favored the Open OnDemand interface. Non-superuser users (P3, P4) relied on it exclusively due to a lack of UNIX/Linux skills required for CLI or SSH access. Open OnDemand also streamlined file transfers and provided access to necessary software (e.g., Matlab for P1).

Both superusers and non-superusers appreciated Open OnDemand's intuitive interface, particularly for first-time or less experienced users. The quarterly *Introduction to HPC* training was also highly valued, serving as an onboarding tool for superusers and providing basic command and cluster usage knowledge for non-superusers.

*4.2.2 Theme 2: Jobs Management (Job Scheduling, Queue Management and Execution of Jobs) .* Job submission and queue management are essential steps when using the HPC Cluster. All jobs on the cluster would achieve the desired results based on the accuracy and correctness of your Job submission file. The Primary tool for Job Scheduling, queue management and execution of Jobs on the HPC Cluster is called "The Slurm Workload Manager." SLURM takes some information about the requirements of resources and sends these calculations to the compute nodes that run the jobs to satisfy the criteria or requirements. It ensures resources are allocated to all users fairly and based on priority.

Two ways to submit jobs on the HPC Cluster currently are using SLURM via the Command Line Interface "CLI" or ARCC Desktop on Open On-Demand.

Our participants (P1-P4) appreciated using the ARC Desktop through the Open on-Demand (Interactive Desktop) portal, which made it easy and accessible to submit their jobs by quickly uploading your Python file. This also allows users to select their preferred resources, (e.g., Number of hours, Number of nodes and partitions); it also has an email notifications option for when your session starts if selected. Open On-demand has a File Transfer button, Job Management, shell access and the ARCC Desktop.

SLURM is an excellent tool for Job scheduling, queuing and submission; However, our participants, both superusers and non-superusers, appreciated working with an interactive graphical user interface desktop as expressed in the following quotes from study participants:

**P4** *"I think using the SLURM platform is a little bit challenging for me as a beginner. The Open On-Demand is more helpful and straight forward."* and *"I think the On-Demand platform helped me a lot when I used the advanced research computing for my dissertation."*

**P3** *"Too difficult to or like I can still run things and get an output, so it's so that so I put it in the middle.Helpful, but it was still usable, so it wasn't too difficult to use. I think it was.Sometimes it does seem that something more interactive could be. Well, like, I feel like I'd be in the middle. Like it wasn't too difficult, but I feel like there are some aspects about like, troubleshooting that could be easier. I with the way that I've been studying up. So I kind of just like send a job and then."*

*4.2.3 Theme 3: Navigating the HPC Web portal .* The core reasons users choose the web portal to access the cluster other than other preferred methods were ease of access, efficiency and optimized performance. Our results from the Think-aloud activity showed that both Super and Non-superusers preferred to use the interactive graphical user interface portal since it came with File Transfer, Job Management, shell access and the ARCC Desktop features. This made it very accessible for users to interact with the portal for their activities.

Though it was evident users preferred the Interactive Desktop, the current Graphical user interface had some cosmetic issues our participants identified and recommended to make the portal more accessible and usable as demonstrated through the following representative quotes on this theme:

**P4** *"You know, Welcome to the ARC?, you know follow these steps one upload your data too you know and then this is how to use open on demand. I think just having that on this front page there's other things I think would make it better but."*

**P3** *"I'm using about the same amount of cores every time, so I'm typically it's like I'm. I'm running the same setup every time, so it seems like going in and then having to type in the stuff I usually have.* and *"Visual Studio code or Mathematica through a portal on open on demand I there's a I'd say there's a decent likelihood I would use it."*

## 4.3 Results from Heuristic Evaluation

A heuristic evaluation was conducted to assess the severity of usability issues in the Interactive web-based Graphical User Desktop for managing the HPC Cluster. The evaluation identified specific violations directly impacting functionality and general violations related to visual and functional design that could lead to usability problems. The violations and their severity ratings are presented in Figure 5.

In total, 18 problems with a severity rating of 0 or higher were found, with 8 having a severity of 2 or higher. The main issues involved complex connection methods and user interface design problems, such as large blank spaces and the lack of an intuitive job status button. Other identified issues were minor cosmetic problems.


**Figure 4: A Text Analysis of Keywords from the Quantitative Data**

Table 1.2: Results of Heuristic Evaluation.

| Violations by Task | Heuristic | Severity |
|---|---|---|
| **Task 1** Going through long procedures to get an account set up | User Control and Freedom | 2 -minor usability problem |
| **Task 2** Users reported difficulty knowing the status of a job and whether an error occurred after submission | Visibility of System Status | 4 -usability catastrophe; imperative to fix |
| **Task 3** No violations | none | N/A |
| **Task 4** we found violations with page alignments, consistency in terms of alignments and fonts | Atheistic and Minimalist Design | 1- Cosmetic problem |
| **Task 5** we found violations were users didn't know either if their submitted jobs were running or an estimated completion time | Flexibility and efficiency of use | 4 - usability catastrophe; imperative to fix |
| **Task 6** Users could not edit their already submitted jobs | User Control and Freedom | 2 - minor usability problem |
| **Task 7** Landing page underutilized, could contain tips or documentation section | Help and Documentation | 1- cosmetic problem |
| **Task 8** Users not able to use other software installed on the Cluster on the web portal | Consistency and Standards | 4- usability catastrophe; imperative to fix |
| **Task 9** Users wanted a more intuitive and interactive web portal | Aesthetic and minimalist design | 3 - major usability problem; important to fix |

**Figure 5: Results of Heuristic Evaluation.**

## 5 RECOMMENDATION AND DISCUSSION

Our results show that users of the HPC Cluster managed by the Advanced Research Computing Center were satisfied with the services provided by ARCC, as shown in Figure 3. This further proved that

HPCs played a significant role in fostering the research capabilities of researchers at the University of Cincinnati. Although there were multiple alternatives for users to connect to the clusters, Open on-Demand was the preferred connection method for job submission, file transfer, and managing job queues.

To address our research question of how HPCs can be made more accessible across disciplines in institutes of higher education, specifically at the University of Cincinnati as our case study and also providing standardized HPC tools, we recommend taking the following actions:

- **Develop a customized graphical user interface (GUI) web HPC management portal**. Considering the research and user environment, this would be tailored to users' needs at the University of Cincinnati. Open On-Demand is an excellent HPC portal, but it is a generic system that other HPC centers use to manage their clusters. However, designing an accessible and customized GUI portal for the University of Cincinnati would improve users' needs and make their use of the High-Performance Computing cluster more accessible.
- **Allow the HPC to be internet-facing**. This would allow users to connect to the cluster off-campus, benefiting researchers who need to access the cluster from home or other locations. Also, this goes further to enable researchers, especially industry companies that require the cluster's services, to integrate third-party systems, e.g. APIs, to improve their research needs. The clusters can be more secure by implementing 2-factor authentication methods, e.g., the current DUO system used university-wide to serve security needs.
- **Workforce Development** To increase workforce development, ARCC should increase the frequency of training sessions and introduce Advanced Level and speciality training for users. Users want to acquire academic credits when these courses are taken, which we believe will encourage novice and frequent users to attend the training sessions. Training can be made mandatory regardless of your experience, especially new users, as an onboarding method for new users of the cluster; this would help to ensure that all users, irrespective of their experience level, have the skills they need to use the HPC cluster effectively.
- **Student Cluster Competition** ARCC should introduce the student cluster competition on campus, primarily partnering with the School of Information Technology (SoIT) and Engineering Faculty during the annual expo; this would be a means of encouraging users to draw more interest to using the High-Performance Computing Cluster.

Feedback from our survey respondents gave a general picture of the current usage of the HPC Cluster. Graduate students from departmental or research labs use the Cluster for their dissertations and funded research. This proves the importance of designing a customized HPC web portal for the University of Cincinnati to make the cluster more accessible for researchers across all disciplines. This would make onboarding and usage of the cluster very easy and direct. Training plays a pivotal role in onboarding users with no experience and users with some experience with Linux/UNIX. However, though some users have experience with Unix, using a cluster requires some special skills, which is beneficial for all users. Frequent training and introducing special programs like the "Student Cluster Competition" will be an excellent initiative for ARCC to solidify its name across the University of Cincinnati ecosystem.

## 6 LIMITATIONS AND FUTURE WORK

Our primary limitation was participant recruitment. Many superusers were unavailable for various reasons, and non-super-users were often hesitant about being recorded on video while performing basic tasks. Greater incentives might be necessary to encourage user participation in future data collection. In future work, we would expand our research criteria and participant pool to several universities in various regions to encompass the broader HPC community. This broader focus would support our goal of making HPCs accessible to researchers across all disciplines.

## 7 CONCLUSION

In this research, we conducted a usability study on how to make HPCs accessible to users across all disciplines, and our case study involved the users of the Advanced Research Computing Center at the University of Cincinnati. Our results highlight several factors that back the need for the Advanced research computing cluster to build a customized, intuitive HPC management web portal. Overall, our results indicated that both super and non-superusers preferred using the Web-based HPC Management portal for job submission and queue management, and implementing our recommendation and future work would improve the ease of access to the cluster and increase the confidence of our non-superusers since the portal would make it easy for them to manage the cluster and implement several technologies through a web-based portal. A critical contribution from this study to workforce development is to expand the training sessions, which have been very instrumental to many users; this training is a resource to a lot of first-time users, especially non-superusers and a few of our respondents, who were on-boarded through the training and have been using the cluster which has made them superusers. This is a suitable means to encourage the use of HPCs across all disciplines, which addresses the challenge in our research question.

## REFERENCES


[1] Amy Apon, Linh Ngo, Michael Payne, and Paul Wilson. 2014. Assessing the effect of high performance computing capabilities on academic research output. *Empirical Economics* 48 (02 2014). https://doi.org/10.1007/s00181-014-0833-7

[2] Amy W. Apon, Linh B. Ngo, Michael E. Payne, and Paul W. Wilson. 2015. Assessing the effect of high performance computing capabilities on academic research output. *Empir. Econ.* 48, 1 (Feb. 2015), 283–312. https://doi.org/10.1007/s00181-014-0833-7

[3] Advanced Research Computing at UC. 2019. Advanced Research Computing at UC. https://arc.uc.edu/citing-the-arc/

[4] Michael A. Bauer. 2007. High Performance Computing: The Software Challenges. In *Proceedings of the 2007 International Workshop on Parallel Symbolic Computation (PASCO '07)*. Association for Computing Machinery, New York, NY, USA, 11–12. https://doi.org/10.1145/1278177.1278180

[5] Patrice Calegari, Marc Levrier, and Paweł Balczyński. 2019. Web Portals for High-performance Computing: A Survey. *ACM Trans. Web* 13, 1 (Feb. 2019), 1–36. https://doi.org/10.1145/3197385

[6] Álvaro Fernández, Camino Fernández, José-Ángel Miguel-Dávila, and Miguel Á. Conde. 2021. Integrating supercomputing clusters into education: a case study in biotechnology. *J. Supercomput.* 77, 3 (March 2021), 2302–2325. https://doi.org/10.1007/s11227-020-03360-5

[7] Interaction Design Foundation. 2021. What is Heuristic Evaluation?

[8] Michail N. Giannakos. 2014. Exploring Students Intentions to Study Computer Science and Identifying the Differences among ICT and Programming Based Courses. *Turkish Online Journal of Educational Technology - TOJET* 13, 4 (Oct. 2014), 36–46. https://eric.ed.gov/?id=EJ1043181

[9] Abid Haleem, Mohd Javaid, Mohd Asim Qadri, and Rajiv Suman. 2022. Understanding the role of digital technologies in education: A review. *Sustainable Operations and Computers* 3 (Jan. 2022), 275–285. https://doi.org/10.1016/j.susoc.2022.05.004

[10] Stephen Harrell, Hai Nam, Verónica Larrea, Kurt Keville, and Dan Kamalic. 2015. Student Cluster Competition: A Multi-disciplinary Undergraduate HPC Educational Tool. https://doi.org/10.1145/2831425.2831428

[11] HPCC 2021. 2021. The 23rd IEEE int'l Conference on HPC and Communications. Retrieved May 26, 2023] from http://www.ieee-hpcc.org/2021

[12] David E. Hudak, Douglas Johnson, Jeremy Nicklas, Eric Franz, Brian McMichael, and Basil Gohar. 2016. Open OnDemand: Transforming Computational Science Through Omnidisciplinary Software Cyberinfrastructure. In *Proceedings of the XSEDE16 Conference on Diversity, Big Data, and Science at Scale*. Association for Computing Machinery, New York, NY, USA, Article 43, 7 pages. https://doi.org/10.1145/2949550.2949644

[13] Barikisu Issaka. 2021. *Alienating: how the portrayal of Muslim women in US media affects Muslim women's social identities*. Ph.D. Dissertation. https://krex.k-state.edu/handle/2097/41509

[14] Peter Kogge, S. Borkar, Dan Campbell, William Carlson, William Dally, Monty Denneau, Paul Franzon, William Harrod, Jon Hiller, Stephen Keckler, Dean Klein, and Robert Lucas. 2008. ExaScale Computing Study: Technology Challenges in Achieving Exascale Systems. *Defense Advanced Research Projects Agency Information Processing Techniques Office (DARPA IPTO), Techinal Representative* 15 (01 2008).

[15] Rolf Molich and Jakob Nielsen. 1990. Improving a human-computer dialogue. *Commun. ACM* 33, 3 (1990), 338–348.

[16] Svati Murthy, Jess Kropczynski, and Shane Halse. 2022. Understanding Decision-Making Needs of Open Government Data Users. https://doi.org/10.24251/HICSS.2022.331

[17] Rajendra K. Raj, Carol J. Romanowski, John Impagliazzo, Sherif G. Aly, Brett A. Becker, Juan Chen, Sheikh Ghafoor, Nasser Giacaman, Steven I. Gordon, Cruz Izu, Shahram Rahimi, Michael P. Robson, and Neena Thota. 2020. High Performance Computing Education: Current Challenges and Future Directions. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education (ITiCSE-WGR '20)*. Association for Computing Machinery, New York, NY, USA, 51–74. https://doi.org/10.1145/3437800.3439203

[18] Olga Scrivner, Gagandeep Singh, Sara Bouchard, Scott Hutcheson, Ben Fulton, Matthew Link, and Katy Borner. 2018. XD Metrics on Demand Value Analytics: Visualizing the Impact of Internal Information Technology Investments on External Funding, Publications, and Collaboration Networks. *Frontiers in Research Metrics and Analytics* 2 (01 2018). https://doi.org/10.3389/frma.2017.00010

[19] Craig A. Stewart, David Y. Hancock, Wernert Julie, Matthew R. Link, Nancy Wilkins-Diehr, Therese Miller, Kelly Gaither, and Winona Snapp-Childs. 2018. Return on Investment for three Cyberinfrastructure facilities: a local campus supercomputer; the NSF-funded Jetstream cloud system; and XSEDE (the eXtreme Science and Engineering Discovery Environment). https://scholarworks.iu.edu/dspace/handle/2022/22590 [Online; accessed 26. May 2023].

[20] Craig A. Stewart, Stephen Simms, Beth Plale, Matthew R. Link, David Y. Hancock, and Geoffrey C. Fox. 2010. What is Cyberinfrastructure? *ACM* (Oct. 2010). https://scholarworks.iu.edu/dspace/handle/2022/12967

[21] Anna Sutton. 2016. Heasuring the Effects of Self-Awareness: Construction of the Self-Awareness Outcomes Questionnaire - PMC. Europe's journal of psychology. https://doi.org/10.5964/ejop.v12i4.1178

[22] Jane Sutton, AnnaSutton and Zubin Austin. 2015. Qualitative Research: Data Collection, Analysis, and Management. In *The Canadian journal of hospital pharmacy (vol. 68,3)*. The Canadian journal of hospital pharmacy, Article , numpages = 68,3,.

[23] High Performance Systems. 2023. HPC everywhere: A quick-access web portal for all things HPC-related at IU.

[24] Tasbakkori Teddlie, C. 2023. *A. (2003). Major Issue and Controversies in the Use of Mixed Methods in Social and Behavioral Sciences. In C. Teddlie, & A. Tasbakkori (Eds.), Handbook of Mixed Methods in Social and Behavioral Sciences (pp. 3-50). Thousand Oaks, CA SAGE. - References - Scientific Research Publishing*. Ph.D. Dissertation. https://www.scirp.org/(S(lz5mqp453edsnp55rrgjct55))/reference/referencespapers.aspx?referenceid=2220642

[25] Abhinav Thota, Le Mai Weakley, Ben Fulton, H. E. Cicada Brokaw Dennis, Laura Huber, Scott Michael, Winona Snapp-Childs, Stephen Lien Harrell, Alexander Younts, Daniel T. Dietz, Christopher Phillips, and Xiao Zhu. 2019. Research Computing Desktops: Demystifying Research Computing for Non-Linux Users. In *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (Learning) (PEARC '19)*. Association for Computing Machinery, New York, NY, USA, Article 54, 8 pages. https://doi.org/10.1145/3332186.3332206

[26] U.S. National Science Foundation. 2023. Frameworks: Software NSCI-Open OnDemand 2.0: Advancing Accessibility and Scalability for Computational Science through Leveraged Software Cyberinfrastructure. Retrieved May 13, 2023] from https://www.nsf.gov/awardsearch/showAward?AWD_ID=1835725 NSF Award Search - Award # 1835725.

[27] U.S. National Science Foundation. 2023. SI2-SSE: Open OnDemand: Transforming Computational Science through Omnidisciplinary Software Cyberinfrastructure. Retrieved May 13, 2023] from https://www.nsf.gov/awardsearch/showAward?AWD_ID=1534949 NSF Award Search - Award # 1534949.

[28] Maaike J. Van den Haak, Menno D. T. de Jong, and Peter Jan Schellens. 2003. Retrospective vs. concurrent think-aloud protocols: Testing the usability of an online library catalogue. *Behaviour & IT* 22, 5 (Sept. 2003), 339–351. https://doi.org/10.1080/0044929031000

[29] XSEDE. 2022. A Brief History of High-Performance Computing (HPC) - XSEDE Home - XSEDE Wiki. Retrieved 04/08/2022) from https://confluence.xsede.org/pages/viewpage.action?pageId=1677620

[30] Il-Chul Yoon, Alan Sussman, and Adam Porter. 2005. And Away We Go: Understanding the Complexity of Launching Complex HPC Applications. In *Proceedings of the Second International Workshop on Software Engineering for High Performance Computing System Applications (SE-HPCS '05)*. Association for Computing Machinery, New York, NY, USA, 45–49. https://doi.org/10.1145/1145319.1145333

[31] Jill Zarestky and Wolfgang Bangerth. 2014. Teaching High Performance Computing: Lessons from a Flipped Classroom, Project-Based Course on Finite Element Methods. 34–41. https://doi.org/10.1109/EduHPC.2014.10

# Education and Support of Large Language Models in a Research Institution

Juan José García Mesa
Arizona State University
jgarc111@asu.edu

Gil Speyer
Arizona State University
speyer@asu.edu

## ABSTRACT

As the capabilities of large language models (LLMs) continue to expand, with more accurate and powerful models being released monthly, researchers and educators are increasingly eager to incorporate these tools into their work. The growing demand for this technology reflects its transformative potential in natural language and its impact on scientific research. However, as more users seek to harness the power of LLMs, the need to provide comprehensive education and scalable support becomes ever more critical. Our institution has recognized this challenge and developed a support framework to educate users through regular educational events, consultations, and project support. To address the growing need for LLM support, we have implemented several key strategies, including deploying Jupyter Lab sessions using Open OnDemand for seamless HPC access and integrating cloud-based solutions via Jetstream2. We provide insights into our approach, detailing how we empower researchers and educators to leverage the capabilities of LLMs in their diverse applications.

## KEYWORDS

Large Language Models, OpenOnDemand, Jetstream2

## 1 INTRODUCTION

The rapid advancement of large language models (LLMs) has sparked significant interest among researchers and educators, driven by the transformative potential in natural language processing and their broad applicability across various scientific disciplines. Our institution has recognized the challenges and opportunities presented by this evolving landscape and developed a support framework that addresses the educational and technical needs of our community. In response to external inquiries and the increasing interest observed during recent presentations about our efforts, we provide an encompassing view of our practices for supporting LLMs, focusing on the strategies and technologies that enable users to harness the full capabilities of these models.

Key technologies that form the backbone of our support infrastructure include Open OnDemand [6], Jupyter Notebooks [8], and Jetstream2 [5]. In addition, we leverage Python packages such as LangChain for building scalable LLM applications, Hugging Face for accessing a vast repository of pre-trained models [10], and Gradio for creating intuitive interfaces for LLM-driven applications [1]. Combined with the technical expertise of maintainers and facilitators from our university staff, these tools create a robust ecosystem that empowers researchers and educators to explore new frontiers in their work.

Here we detail our approach, highlighting the methods and tools we employ to educate users, facilitate LLM adoption, and provide ongoing support for a wide range of projects. By sharing our experiences, we hope to contribute valuable insights to the broader community engaged in similar efforts.


https://doi.org/10.22369/issn.2153-4136/16/1/6

## 2 TEACHING AND SUPPORT OPPORTUNITIES

### 2.1 Building Foundational Knowledge

To facilitate the effective use of LLMs, our institution conducts a series of teaching workshops throughout the year designed to introduce users to both the theoretical and practical aspects of these models. These workshops cover:

- An overview of the available HPC resources, how to request resources, and best practices for optimal usage
- Requesting a Jupyter lab session with GPUs and adequate memory using Open OnDemand
- Question-answer chatbot use cases using a Jupyter Notebook
- Retrieval-augmented generation
- How to submit batch queries for large datasets
- Fielding questions and providing guidance

Through these workshops, participants gain the necessary skills to initiate their own projects employing LLMs in their research and teaching activities. In addition, our department organizes day-long events each academic semester to display the research technology tools available and showcase success stories within our community, including a high number of cases that leverage artificial intelligence applications. Researchers from our institution who have benefited from using the available services present their work to the community.

### 2.2 Consultation and Project Support

Beyond workshops, we engage in collaborations with faculty and researchers, offering specialized support for projects that use advanced LLM capabilities. Our involvement includes:

- Consultation and planning: assisting in the design of experiments and the selection of appropriate models.
- Resource provisioning: allocating resources, including GPUs, ensuring that large-scale LLM experiments are executed efficiently.
- Skill development: ongoing training and mentorship to help users adapt to evolving LLM technologies.

- Execution and results: in limited cases, we work alongside the researcher during the majority of the project and assist in running the experiments.

Often these projects are exploratory and having an experienced facilitator helps advance progress. Furthermore, these commitments ensure that complex LLM-based projects receive the sustained support they need to succeed.

# 3 FRAMEWORK

## 3.1 Institution-Supported Web Interface

Open OnDemand, an NSF-funded open-source HPC portal, plays a crucial role in our user support strategy. This web-based interface simplifies the use of HPC resources that require or are enhanced by a graphic interface, creating an accessible option for users to launch Jupyter lab sessions and run LLMs.

To further streamline user experience and system efficiency, we provide pre-downloaded models. This approach reduces the need for users to fetch and load models individually, which not only simplifies their workflow but also helps us, as maintainers, to troubleshoot issues more effectively. Additionally, pre-downloading models reduce the memory footprint on the system, ensuring a more stable and efficient environment for all users. In line with this effort to streamline resources, we recommend the use of permanent databases that are created once and queried multiple times, using technologies such as ChromaDB.

We also offer existing materials that implement a user-friendly interface to run question-and-answer chats [9] and retrieval-augmented generation [4]. By lowering the barrier to entry, OpenOnDemand enables a wider range of users to take advantage of our LLM support infrastructure.

## 3.2 Command Line Interface for Batch Queries

Jupyter notebooks provide intuitive and interactive access to LLMs. However, this approach lacks scalability. For users with command-line experience, we provide Python and sbatch template scripts to process LLM queries in parallel.

## 3.3 Cloud Computing with Jetstream2

Our institution also leverages cloud-based resources through Jetstream2, a national science and engineering cloud funded by the NSF and made accessible through the ACCESS [2] and Campus Champions [3] programs. Jetstream2 offers 8 petaFLOPS of supercomputing power, designed to simplify data analysis and support AI-driven research. It provides researchers with on-demand access to interactive computing resources, including a library of virtual machines and shared software for creating customized research environments. This user-friendly platform enables researchers to build personalized virtual machines or private computing systems. Key benefits include:

- Flexibility: provisioning virtual machines tailored to the specific needs of LLM projects, including custom configurations for GPU and memory.
- Availability: reduced waiting time for the compute nodes.
- Usability: Jetstream2 allows the creation of pre-defined images that users can deploy without requiring setup.
- Collaboration: facilitating collaborative efforts across institutions through shared access to cloud-based environments.

Offered as a specialty request, the integration of Jetstream2 with our existing HPC framework provides a flexible, scalable solution that complements our on-premise resources. Notably, this resource provides the means to operationalize proof-of-concept projects initially developed on Sol, ASU's flagship supercomputer [7]. Additionally, Jetstream2 offers the capability to provision sustained instances that are continually accessible, ensuring the long-term stability of research tasks.

## 3.4 Infrastructure

A well-known issue in supporting artificial intelligence applications is the resource-intensive nature of the workflows that translates to a high demand for resources, especially GPUs. The combined request for these resources in a large research institution can often create bottlenecks that result in long waiting times. At our institution, we have implemented several measures to alleviate this situation that focus on improving efficiency. We advise our users to choose the most fitting node partition and quality of service flag to increase the pool of possible nodes where their jobs can land. Specifically for GPUs, we offer the ability to specify different models (i.e., Nvidia A30 vs Nvidia A100 40GiB vs Nvidia A100 80GiB vs first available). We also leverage multi-instance GPU (MIG) slices, which facilitate efficient resource utilization and work well for quantized models. When applied to our Nvidia A100 GPUs, the 20GiB MIG slices allow comfortably running LLMs with up to 8 billion parameters.

# 4 DISCUSSION

Users of LLMs range from novices wanting to evaluate their potential applications for their research to experts training foundational models or fine-tuning existing ones. The former presents a use case where education and plug-and-play materials are key, offered by our regularly schedule workshops and by-appointment consultations. The latter, and ultimately all cases present a heavy demand for resources. We mitigate this by taking advantage of MIG slices, a curated hierarchy of node and GPU specifications, and offering consultations to researchers wanting to optimize their computations.

# 5 CONCLUSION

Supporting large language models at our institution, from proof-of-concept to sustained instances, requires a multifaceted effort that combines educational initiatives, sustained project collaboration, and a wide array of advanced computational resources. By leveraging tools such as Open OnDemand and Jetstream2, commonly and easily implemented in centralized systems, and an active outreach program that educates users about efficiency best practices, we empower our users to explore the full potential of LLMs in their work, contributing to the broader academic and research community.


# ACKNOWLEDGMENTS

The authors acknowledge Research Computing at Arizona State University for providing computing and storage resources that contribute to the deployment and maintenance of the systems described within this paper [7].

## REFERENCES


[1] Abubakar Abid, Ali Abdalla, Ali Abid, Dawood Khan, Abdulrahman Alfozan, and James Zou. 2019. Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild. arXiv:cs.LG/1906.02569 https://arxiv.org/abs/1906.02569

[2] Timothy J. Boerner, Stephen Deems, Thomas R. Furlani, Shelley L. Knuth, and John Towns. 2023. ACCESS: Advancing Innovation: NSF's Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support. In *Practice and Experience in Advanced Research Computing 2023: Computing for the Common Good (PEARC '23)*. Association for Computing Machinery, New York, NY, USA, 173–176. https://doi.org/10.1145/3569951.3597559

[3] Marisa Brazil, Dana Brunson, Aaron Culich, Lizanne DeStefano, Douglas Jennewein, Tiffany Jolley, Timothy Middelkoop, Henry Neeman, Lorna Rivera, Jack Smith, and Julie Wernert. 2019. Campus Champions: Building and sustaining a thriving community of practice around research computing and data. In *Practice and Experience in Advanced Research Computing 2019: Rise of the Machines (Learning) (PEARC '19)*. Association for Computing Machinery, New York, NY, USA, Article 78, 7 pages. https://doi.org/10.1145/3332186.3332200

[4] Juan Jose Garcia Mesa and Gil Speyer. 2024. Supplementary materials for Enhancing the application of large language models with retrieval-augmented generation for a research community. https://doi.org/10.5281/zenodo.13328766

[5] David Y. Hancock, Jeremy Fischer, John Michael Lowe, Winona Snapp-Childs, Marlon Pierce, Suresh Marru, J. Eric Coulter, Matthew Vaughn, Brian Beck, Nirav Merchant, Edwin Skidmore, and Gwen Jacobs. 2021. Jetstream2: Accelerating cloud computing via Jetstream. In *Practice and Experience in Advanced Research Computing 2021: Evolution Across All Dimensions (PEARC '21)*. Association for Computing Machinery, New York, NY, USA, Article 11, 8 pages. https://doi.org/10.1145/3437359.3465565

[6] Dave Hudak, Doug Johnson, Alan Chalker, Jeremy Nicklas, Eric Franz, Trey Dockendorf, and Brian L McMichael. 2018. Open OnDemand: A web-based client portal for HPC centers. *Journal of Open Source Software* 3, 25 (2018), 622.

[7] Douglas M. Jennewein, Johnathan Lee, Chris Kurtz, Will Dizon, Ian Shaeffer, Alan Chapman, Alejandro Chiquete, Josh Burks, Amber Carlson, Natalie Mason, Arhat Kobwala, Thirugnanam Jagadeesan, Praful Barghav, Torey Battelle, Rebecca Belshe, Debra McCaffrey, Marisa Brazil, Chaitanya Inumella, Kirby Kuznia, Jade Buzinski, Sean Dudley, Dhruvil Shah, Gil Speyer, and Jason Yalim. 2023. The Sol Supercomputer at Arizona State University. In *Practice and Experience in Advanced Research Computing (PEARC '23)*. Association for Computing Machinery, New York, NY, USA, 296–301. https://doi.org/10.1145/3569951.3597573

[8] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, et al. 2016. Jupyter Notebooks–a publishing format for reproducible computational workflows. In *Positioning and power in academic publishing: Players, agents and agendas*. IOS press, https://ebooks.iospress.nl/doi/10.3233/978-1-61499-649-1-87, 87–90.

[9] Dhruvil Shah, Gil Speyer, and Jason Yalim. 2023. Centralized provisioning of large language models for a research community. 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, 704–707. https://doi.org/10.1145/3624062.3624147

[10] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. arXiv:cs.CL/1910.03771 https://arxiv.org/abs/1910.03771

# HPC Carpentry: Recent Progress and Incubation Toward an Official Carpentries Lesson Program


Andrew Reid
National Institute of Standards and Technology
andrew.reid@nist.gov

Trevor Keller
National Institute of Standards and Technology
trevor.keller@nist.gov

Alan O'Cais
University of Barcelona
alan.ocais@cecam.org

Annajiat Alim Rasel
BRAC University
annajiat@bracu.ac.bd

Wirawan Purwanto
Old Dominion University
wpurwant@odu.edu

Jane Herriman
Lawrence Livermore National Laboratory
herriman1@llnl.gov

Benson Muite
Kichakato Kizito
benson_muite@emailplus.org

Marc-André Hermanns
RWTH Aachen University
hermanns@itc.rwth-aachen.de



## ABSTRACT

The HPC Carpentry project aims to develop highly interactive workshop training materials to empower novices to effectively leverage HPC to solve scientific and technical problems in their domains. Modeled after The Carpentries training programs, the project's goal is to develop foundational HPC skills and a sense of empowerment, rather than expertise. The workshop setting provides learners with hands-on experience that elicits confidence working with HPC systems and provides sufficient vocabulary to make subsequent self-study more effective.

In a major milestone, the steering committee is leading HPC Carpentry through the formal incubation process to become an official Carpentries lesson program alongside the existing Software, Data, and Library Carpentry programs. This achievement is the product of significant work over the past several years, incorporating valuable materials from many contributors. Our most recent focus has been developing materials for a user workshop. We begin with an introduction to the command-line shell (using Software Carpentry's Unix Shell lesson), followed by our Introduction to HPC lesson, covering remote access and resource management. We end with a newly developed lesson on HPC workflow management, which walks learners through the execution of a scaling study on an HPC system, emphasizing both the benefits and limitations of the system for domain applications. This workshop program was recently run in full at the Lawrence Livermore National Laboratory.

Future plans include building a developer workshop, reconnecting with disparate contributors, and engaging with the broader community through regular open conference calls and outreach.




https://doi.org/10.22369/issn.2153-4136/16/1/7



## KEYWORDS

Cyberinfrastructure, Training, Pedagogy, HPC, Parallel Computing, Big Data, Non-degree Training, Hands-on


## 1 BACKGROUND

HPC Carpentry [12] is an informal training project with a mission to provide a set of lessons aimed at introducing the basic "know-how" of running applications on high-performance computing (HPC) resources to new audiences, including investigators from fields which are not traditional users of HPC systems, as well as novice users from fields in which HPC is commonly used. Eventually, the project's goal is to empower HPC novices to effectively leverage HPC to solve scientific and technical problems in their respective domains. The project paves the way for the potential users from non-traditional HPC disciplines to tap into HPC resources for their data analysis, modeling, and simulation needs while remaining relevant for beginners from the traditional HPC disciplines.

The project is patterned after the broader Carpentries [11] project, and adopts many of that organization's priorities, including an emphasis on accessibility and open-source materials. The current project is the product of significant work over the past several years, incorporating valuable materials from many contributors.

### 1.1 History

Efforts we have drawn on include materials from Compute Canada (now the Digital Research Alliance of Canada), and Peter Steinbach's "HPC In A Day" lessons. The community had numerous important discussions at a birds-of-a-feather event at the SC17 conference, at CarpentryCon in 2018, and another birds-of-a-feather session at the SC18 conference.

The current leadership team coalesced around the content at the end of 2019, when competing demands absorbed the attention of many previous contributors. During the height of the COVID-19 pandemic, the new team had useful discussions with the Carpentries curriculum development team at CarpentryCon 2020@Home, where the idea of seeking to become a new lesson program alongside Software Carpentry, Data Carpentry and Library Carpentry began to be seriously considered. Several of our lessons benefited from learner feedback following workshops [15], especially a semester-long HPC workshop piloted at BRAC University [7]. This proof-of-concept began in 2020 to provide the students of Parallel, Distributed, & High-Performance Computing (HPC) and Distributed Computing Systems their first exposure to HPC through these open-source lessons.

In 2021, a number of HPC Carpentry lessons were moved into the Carpentries Incubator, and the strategic plan for workshop development began to come into focus. To gather community perspectives, a bird of a feature session was held at SC'21 [17]. The project had a significant presence at CarpentryCon 2022, including a sprint that began serious development on the workflow lesson

Since the SC23 conference session on HPC Carpentry – A Scalable, Peer-Reviewed Training Program to Democratize HPC Access [18], we have become much better connected with the broader HPC education community, which has already led to new contributors and workshop opportunities.

## 2 LESSON DEVELOPMENT EFFORTS

The recent focus of HPC Carpentry has been the development of a complete workshop program for new HPC users.

### 2.1 HPC User Workshop

The HPC User workshop consists of three lessons and targets HPC novices. Though we see variety in learner profiles, an example learner might come to an HPC user workshop with some background in a domain science and a research problem that requires the use of HPC or cluster resources; that user may or may not have former programming experience in a laptop or desktop environment. The workshop aims to cultivate basic HPC skills and a sense of self-efficacy for post-workshop learning; additionally, key themes of the workshop are automation, reproducibility, and reusability.

We begin with an introduction to the command-line interface using Software Carpentry's Unix Shell lesson [16]. This lesson gives learners a basic familiarity with command-line operations in the shell, the structure of Unix commands, and the use of arguments to control the behavior of the command, as well as basic operations on hierarchical file systems.

This lesson is followed by our Introduction to High Performance Computing lesson [13], covering remote access and resource management. Starting where the Unix Shell lesson leaves off, it begins by introducing clusters and how to connect to them over SSH. Learners then explore remote resources and differences from traditional desktop machines, *e.g.*, distinguishing between "login" and "compute nodes". They learn to launch and investigate jobs on compute nodes using Slurm, the resource manager used throughout this lesson, and to customize their software environment through the module system. This lesson also introduces how to execute parallel tasks over distributed resources using an MPI-parallelized mini-application called 'amdahl' that mimics the scaling behavior of a parallel program retaining serial sections.

We end with a newly developed lesson on HPC workflow management [14], which walks learners through the specification and execution of a scaling study on an HPC system, emphasizing both the benefits and limitations of HPC systems for domain applications. Instead of writing a series of Slurm job scripts that run the `amdahl` binary with different parameters, learners automate a workflow that sweeps their scaling study's parameter space using a workflow tool (either Snakemake or Maestro).

This set of three lessons is sufficient to offer a two-day hands-on workshop in a format similar to that of the Software Carpentry or Data Carpentry workshops.

### 2.2 HPC Developer Workshop

The project plans to develop more advanced training resources for HPC developers. Whereas target learners for the HPC User Workshop may only need to run pre-built software for their work and research, target learners for the HPC Developer Workshop are expected to have software development tasks and experience programming outside an HPC environment. In this workshop, instructors will have the option to substitute the workflow lesson with a coding exercise in a parallel framework such as MPI, for example.

The content for the HPC Developer Workshop's lesson on parallel programming will likely grow from existing sources. For example, several relevant lessons, including an introduction to parallel programming using the Chapel programming language, already exist within the Carpentries Incubator. A lesson using Programming Big Data with R (pbdR) is also in development [9]. Furthermore, we have received engagement from HPC community members at large who explore the potential of merging their in-house lessons into HPC Carpentry's lesson portfolio. This is still an ongoing effort and engagement with the community.

### 2.3 Customizing Workshop Materials

Lesson material customizations may be desired to better reflect the configuration of a workshop host's HPC resources or to better address the needs of a particular learner audience. There are two principle ways that the lessons can be adapted to local resources.

Firstly, the HPC Intro lesson has a "snippet library", which allows the rendered lesson to be customized to use the correct resource manager for a site (Slurm, GridEngine, or Torque) and to present the correct machine names and site-specific command arguments. Making these customizations reduces cognitive load for the Instructor during the workshop and provides correct commands for learners revisiting the content afterwards.

Secondly, these lessons are modular. Individual modules can be dropped if they are inappropriate for a particular site. Because these lessons are primarily meant to be presented in real time to learners, instructors can and should "read the room", or consult with helpers, to decide when to slow down or dive deeper into particular parts of the material.

In keeping with Carpentries best practice, all of this material is open-source and freely licensed (CC BY-4.0). We share the Carpentries' commitment to continuous improvement of the lessons through learner and instructor feedback. This mechanism also allows others from the community to adapt and re-use the material as they see fit.

## 2.4 Building Community

Additionally, we plan to boost our engagement both with the Carpentries and the HPC community through specific interactions at Carpentries events, such as the CarpentriesConnect'24 in Heidelberg, Germany. Beyond the core curriculum—proposed in the Lesson Program Incubator—we want to discuss with the community, which other lessons exist, are in development, or even planned, that are relevant to the HPC Carpentry, and how to integrate them within HPC Carpentry. To enable such discussion, lesson development is done in the open on GitHub and there are bi-monthly online calls. This will be highly valuable for enabling a broader topic range for the proposed HPC Developer Workshops and ensures in-depth engagement of additional instructors and lesson developers as well as keeping the available topics relevant.

# 3 RECENT WORKSHOPS

The complete HPC Carpentry workshop program for new users was recently offered online for the BioNT project in February of 2024 [10] and at Lawrence Livermore National Laboratory in June of 2024. The HPC Intro lesson was offered as a stand-alone unit at the National Institute of Standards and Technology in July of 2024. Previous workshops were held at University College Dublin, BRAC University, Helmholtz Einstein International Berlin Research School in Data Science (HEIBRiDS), University of Mauritius, Florida International University, Delft University of Technology, National Institute of Standards and Technology, and EPFL CECAM.

Feedback from these workshops has been crucial in improving the lesson material.

## 3.1 Gathering Attendee Feedback

In keeping with the Carpentries' methods, HPC Carpentry workshop feedback is solicited in three primary ways. First, prior to the workshop, a pre-workshop survey gathers information about the attendees' skill levels and familiarity with programming concepts. The instructors can then attempt to present workshop content at the level best suited to each unique audience. Second, throughout the workshop, users are asked to use colored sticky notes to indicate whether they are keeping up with the instructor or if they require assistance catching up from one of the non-lecturing instructors circulating the room. At the end of each in-person session, instructors will also sometimes prompt their learners to leave written comments on sticky notes (typically, one good thing, one bad thing). These methods help to capture learner feedback with little overhead and while it is still fresh, presenting the instructors with realtime feedback on the pacing and clarity of their instruction as well as the quality of the content. Third, a post-workshop survey gathers information about the attendees' workshop experience and their level of comfort with the concepts treated in the workshop. These post-workshop responses allow us to modify our material and approach for future workshops.

## 3.2 Lessons Learned

From the run-up to the BioNT workshop, we discovered some important version-specific updates in the workflow tool we chose to use (Snakemake), which required significant modifications to the draft lesson to accommodate.

From the Lawrence Livermore workshop, we heard from learners that the process of building up a workflow configuration file is sensitive to a loss of context: if a learner misses a step, it's hard to recover, because the next version of the file depends on a consistent prior version. Having a shared virtual notepad with "checkpoint" versions of the relevant files can help learners recover the needed context and not lose the thread of the lesson. The instructor team used such a notepad to catch learners up in real time during the latter part of the LLNL workshop, as well as using the `give` tool to transfer files in their entirety [6].

The NIST lesson was run for members of that institution's Summer Undergraduate Research Fellowship program. Learners self-selected based on a published invitation, and the instructor team found it beneficial to teach the lesson more slowly and with deeper attention to the details of the commands than had been the case at preceding Livermore workshop. This provided a very striking example of adapting the lesson to the room, which is a significant benefit of live workshops.

## 3.3 HPC Infrastructure

One important issue we are facing in offering HPC Carpentry workshops is the need for HPC infrastructure for learners to use during the workshop. While some HPC site operators have their own HPC systems to conduct their own workshop, others, particularly from under-resourced institutions, do not have their own HPC resources. More recently, we have acquired support from Jetstream2 through ACCESS to set up a "standard" reference HPC Carpentry cluster in a virtual-machine-based environment. The cluster set-up has been prepared in an automated fashion using the Magic Castle tool [4], built on the Terraform cloud provisioning tool [5]. This effort could pave the way to allow instructors to configure their own clusters, irrespective of the existence of a local HPC cluster in their own institution. This provisioning scheme is itself open-source, and significantly lowers the barrier to HPC operations for anyone in the community.

A cluster alone is not enough, one also needs a software stack. Magic Castle provides the user with a complete HPC cluster software environment including a Slurm scheduler, a Globus Endpoint, JupyterHub, LDAP, DNS, and over 3000 research software applications compiled by experts. With respect to a scientific software stack, Magic Castle provides two primary options: the Digital Research Alliance of Canada (the Alliance) software stack [2], and the European Environment for Scientific Software installations (EESSI) [3]. Both of these are distributed via CernVM-FS [1] and can be easily utilised by the participants after the workshop.

# 4 CARPENTRIES LESSON PROGRAM INCUBATION

In a major milestone, the steering committee is leading the project through the formal incubation process towards becoming an official Carpentries lesson program alongside the existing Software, Data, and Library Carpentry programs. This process is expected to last for about 18 months, with expected completion in December 2025. We hope to establish a strong community, a useful set of lessons and a good governance process by that time.

## ACKNOWLEGMENTS

We acknowledge the infrastructure support hosted on Jetstream2 cloud environment. Allocation to Jetstream2 was provided through ACCESS, which is funded by the US National Science Foundation.

Alan O'Cais is co-funded by the European High-Performance Computing Joint Undertaking (JU) and countries participating in the project under grant agreement No 101093169 (MultiXscale), and the European Union under grant agreement number 101100604 (BioNT).

The semester long trainings for the students of Parallel, Distributed, & High-Performance Computing (HPC) at BRAC University [7] through HPC Carpentry Workshops were made possible since 2020 using the HPC resources sponsored by the European initiative LearnHPC [8] and the recent JetStream2 arranged by Alan O'Cais, Intel Dev Cloud, Amazon Cloud and other resources arranged by Benson Muite, Oracle Cloud arranged by Annajiat Alim Rasel.

## REFERENCES

[1] Jakob Blomer, Carlos Aguado-Sánchez, Predrag Buncic, and Artem Harutyunyan. 2011. Distributing LHC application software and conditions databases using the CernVM file system. In *Journal of Physics: Conference Series*, Vol. 331. IOP Publishing, Bristol, United Kingdom, 042003.

[2] Maxime Boissonneault, Bart E. Oldeman, and Ryan P. Taylor. 2019. Providing a Unified Software Environment for Canada's National Advanced Computing Centers. In *Practice and Experience in Advanced Research Computing 2019: Rise of the Machines (Learning) (PEARC '19)*. Association for Computing Machinery, New York, NY, USA, Article 51, 6 pages. https://doi.org/10.1145/3332186.3332210

[3] Bob Dröge, Victor Holanda Rusu, Kenneth Hoste, Caspar van Leeuwen, Alan O'Cais, and Thomas Röblitz. 2023. EESSI: A cross-platform ready-to-use optimised scientific software stack. *Software: Practice and Experience* 53, 1 (2023), 176–210. https://doi.org/10.1002/spe.3075

[4] Félix-Antoine Fortin, Etienne, Frédéric Fortier-Chouinard, Maxime Boissonneault, Charles Coulombe, Erik Sundell, Simon Guilbault, Alan ÓCais, Étienne Parent, R. P. Taylor, jboschee, Kenneth Hoste, Adam Huffman, Chris Want, Darren Boss, Justin Lagüe, Marie-Helene Burle, Quoding, and Pier-Luc St-Onge. 2024. Magic Castle. https://doi.org/10.5281/zenodo.10953488

[5] Mitchell Hashimoto. 2014. Terraform. https://www.terraform.io

[6] Shawn Instenes. 2017. give/take. https://github.com/hpc/give

[7] B. Muite, A. A. Rasel, M. I. Hafiz, and M. A. Rahman. 2020. Two Decades Making an Impact, Brac University Annual Report 2020. , 34 pages. https://www.bracu.ac.bd/sites/default/files/resources/ar/Brac%20University%20Annual%20Report%202020-compressed.pdf

[8] A. O'Cais. 2021. LearnHPC - Scalable HPC Training. https://www.learnhpc.eu/

[9] G. Ostrouchov, W.-C. Chen, D. Schmidt, and P. Patel. 2012. Programming with Big Data in R. http://r-pbd.org/

[10] The BioNT Project. 2024. An introduction to High Performance Computing. https://www.cecam.org/workshop-details/an-introduction-to-high-performance-computing-1270

[11] The Carpentries Project. 2024. The Carpentries. https://carpentries.org

[12] The HPC Carpentry Project. 2024. HPC Carpentry: Teaching Basic Skills for High-Performance Computing. https://www.hpc-carpentry.org/

[13] The HPC Carpentry Project. 2024. hpc-intro: Introduction to High-Performance Computing. https://carpentries-incubator.github.io/hpc-intro/

[14] The HPC Carpentry Project. 2024. hpc-workflow: HPC Workflow Management with Snakemake. https://carpentries-incubator.github.io/hpc-workflows/

[15] The HPC Carpentry Project. 2024. Past Workshops related to HPC Carpentry. https://www.hpc-carpentry.org/past-workshops/

[16] The Software Carpentry Project. 2023. Software Carpentry: the UNIX shell. https://doi.org/10.5281/zenodo.7886938

[17] A. Reid, T. Keller, A. O'Cais, and A. A. Rasel. 2021. HPC Carpentry: Introducing New Users to HPC. https://sc21.supercomputing.org/presentation/index-id=bof161&sess=sess403.html

[18] A. Reid, A. O'Cais, T. Keller, W. Purwanto, and A. A. Rasel. 2023. HPC Carpentry – A Scalable, Peer-Reviewed Training Program to Democratize HPC Access. https://sc23.supercomputing.org/proceedings/workshops/workshop_pages/ws_bphpcte106.htmlandhttps://jocse.org/downloads/jocse-15-1-6.pdf

# Integrating Captive Portal Technology into Computer Science Education: A Modular, Hands-On Approach to Infrastructure

Lianting Wang
University of Toronto Scarborough
lianting.wang@alumni.utoronto.ca

Marcelo Ponce
University of Toronto Scarborough
m.ponce@utoronto.ca

## ABSTRACT

In this paper, we present an educational project aimed to introduce students to the technology behind *Captive Portals* infrastructures. For doing this, we developed a series of modules to emphasize each of the different aspects and features of this technology. The project is based on an open source implementation which is widely used in many computer network courses, making it well-suited and very appealing for instructors and practitioners in this field.

## KEYWORDS

Computer Networks, Captive Portal, Computer Science Education

## 1 INTRODUCTION

A *Captive Portal* is a form of network connectivity where access to the Internet is restricted until certain requirements are satisfied. It presents the access-point provider's requirements through an interface, typically a web-page type, such as reading advertisements, accepting acceptable usage policies, or providing some form of credentials [6].

Nowadays, Captive Portals are ubiquitous in our lives. Whether we are in airports, hotels, or restaurants, Captive Portals have always provided an easy way to authenticate our identity and access the Internet. Although the time spent interacting with Captive Portal interfaces is short and in some cases considered a formality, the technology encompasses a rich set of network protocols and authentication processes. HPC and distributed compute systems can use this approach to validate users and access to resources, e.g. gateways to specialized portals, disclose systems' policies, etc. Automated pop-ups from a mobile phone or computer device further illustrate the versatility and complexity of this technology. These features will stimulate people's curiosity and encourage them to further explore the field of computer network security and infrastructure [1].


https://doi.org/10.22369/issn.2153-4136/16/1/8

The pervasive nature of Captive Portals in everyday scenarios presents a unique opportunity for computer science educational exploration, especially in the field of computer networks. Students are already familiar with the user-oriented aspects of the technology and are standing on the threshold of deeper exploratory learning. By demystifying the core principles of Captive Portals, we can transform students' casual encounters with this technology into profound learning experiences. This educational journey promises not only to help students gain a deeper understanding of network communications and its related protocols, but also to foster critical thinking about the security measures inherent in seemingly mundane digital interactions.

The teaching case model proposed in this paper is an innovative educational tool designed to explore Captive Portal technology in depth through hands-on practice, thereby promoting students' comprehensive understanding of basic network technologies. By simulating the operation of a typical Captive Portal, students are able to intuitively learn and understand the core technical principles in networking, including the reasons for the use of media access control (MAC) addresses, the role of address resolution protocol (ARP) requests, the working mechanism of switches, the processing of domain name system (DNS) requests, the methods of applying web servers, and the key differences between HTTP and HTTPS. This type of learning is not limited to academic exploration; it also helps students relate their theoretical knowledge to the construction and operation of networks in their daily lives, leading to a more intuitive understanding of how networks work.

This model emphasizes the close integration of technology and life, encouraging students to become more engaged in their studies by increasing their interest, while also developing their practical application skills [7]. Through this model, students are able to not only grasp the technical fundamentals of the Captive Portal system, but also critically assess its security and further understand the importance of network security and infrastructure. This pedagogical approach seeks to bridge theoretical knowledge with its practical application, enabling students to apply what they have learned to solve real-life network problems they may encounter, thereby developing their comprehensive understanding and application skills.

## 2 DESIGN PRINCIPLES

### 2.1 Introduction to Design Principles

The Captive Portal instructional project is underpinned by a set of core design principles aimed at maximizing the educational value for students. These principles are crafted to ensure a blend of practical learning experiences with a strong theoretical foundation. By engaging students in hands-on activities involving Software-Defined Networking (SDN) and Captive Portal technologies, the project bridges the gap between conceptual knowledge and real-world applications. The overarching goal is to foster a deep understanding of network communication principles, system architecture, and the challenges of implementing network-based applications.

## 2.2 Extensive Coding Instructions

Central to our approach is the provision of detailed coding instructions and guidelines. Before embarking on code development, students are required to engage with comprehensive documentation that outlines the system's architecture, design considerations, and specific module requirements. This preparatory step is designed to shift the focus from minutiae to the broader system perspective, encouraging students to appreciate the interconnectedness of system components. Supplementary resources, including helper functions and baseline code, are provided to streamline the implementation process, enabling students to concentrate on solving more complex problems.

## 2.3 Modular Design

Acknowledging the diverse focus areas and difficulty levels required across different courses, the Captive Portal project is inherently modular. This flexibility allows instructors to tailor the project to fit educational objectives, selecting modules that align with specific aspects of networking, such as SDN or DNS. Moreover, the modularity supports scalability in complexity, making it suitable for a wide range of student expertise. Such a design not only enhances the learning experience but also empowers educators to customize the curriculum to meet students' needs more effectively.

## 2.4 Critical Thinking and Problem-Solving

The tiered difficulty levels within the modular design introduce a unique opportunity for critical evaluation. Students are challenged to assess the adequacy of the current module design, promoting critical thinking. This evaluative process deepens their understanding of the system, encouraging a thoughtful consideration of design efficiency, scalability, and performance. Through this, students develop a keen eye for optimizing solutions, a skill critical in the fields of HPC and technology development [3].

## 2.5 Stable Development Environment

Recognizing the complexity of setting up a development environment for such a diverse project, we provide a pre-configured virtual environment. This environment includes all necessary dependencies and tools, ensuring uniformity across all modules. By eliminating the setup barriers, students can immediately focus on the learning objectives and practical experience, rather than grappling with configuration challenges.

## 2.6 Test-Driven Development

Embracing best practices in software development, the project emphasizes test-driven development (TDD). For each module, comprehensive test suites are provided, guiding students through the development process. This approach not only facilitates immediate feedback on the correctness of implementations but also instills a disciplined coding methodology [4]. By integrating TDD, students learn to prioritize functionality and reliability from the outset, a practice that significantly benefits their future professional endeavors.

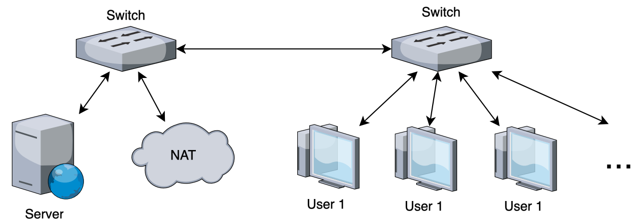



**Figure 1: Basic structure for a typical Captive Portal configuration.**

# 3 IMPLEMENTATION

For our project we decided to use an open-source tool named *mininet* –https://mininet.org/– [5, 9], which has been widely used and adapted in many computer networks courses due to its versatility, flexibility and open approach. Employing *mininet* as a *software defined network* (SDN) solution, we developed our main infrastructure for the teaching modules and educational projects.

Fig. 1 depicts a typical infrastructure for a Captive Portal deployment, where users and servers are connected through two Switches.

Using SDN technology, we can manage connections at the link layer, directing users with unique MAC addresses either to the network address translation (NAT) or to the Captive Portal server. However, since the network layer focuses solely on delivering data to the correct IP address without regard to the connection's status, we employ specific techniques to reroute user requests to the appropriate server.

One possible method is the so-called *DNS Spoofing* – see Fig. 2. In this case when a user tries to access an external website, their device sends out a DNS request to find the website's IP address. Normally, this would lead them to the intended website, but we intervene by using a special DNS server. This server directs all domain name requests to the Captive Portal server, ensuring the user encounters the authentication page regardless of their intended destination.

Another approach we use is *IP Forgery*. With this technique, when a user‘s device sends a DNS request, we provide the correct IP address. However, when the user attempts to access the website, we intercept the web request and respond with an HTTP redirect message that redirects the user to the Captive Portal server's domain name. The user‘s second connection will connect to the Captive Portal server.

Regardless of the specific methodology used, the framework of the entire project is fixed, as shown in Figs. 3 and 4.

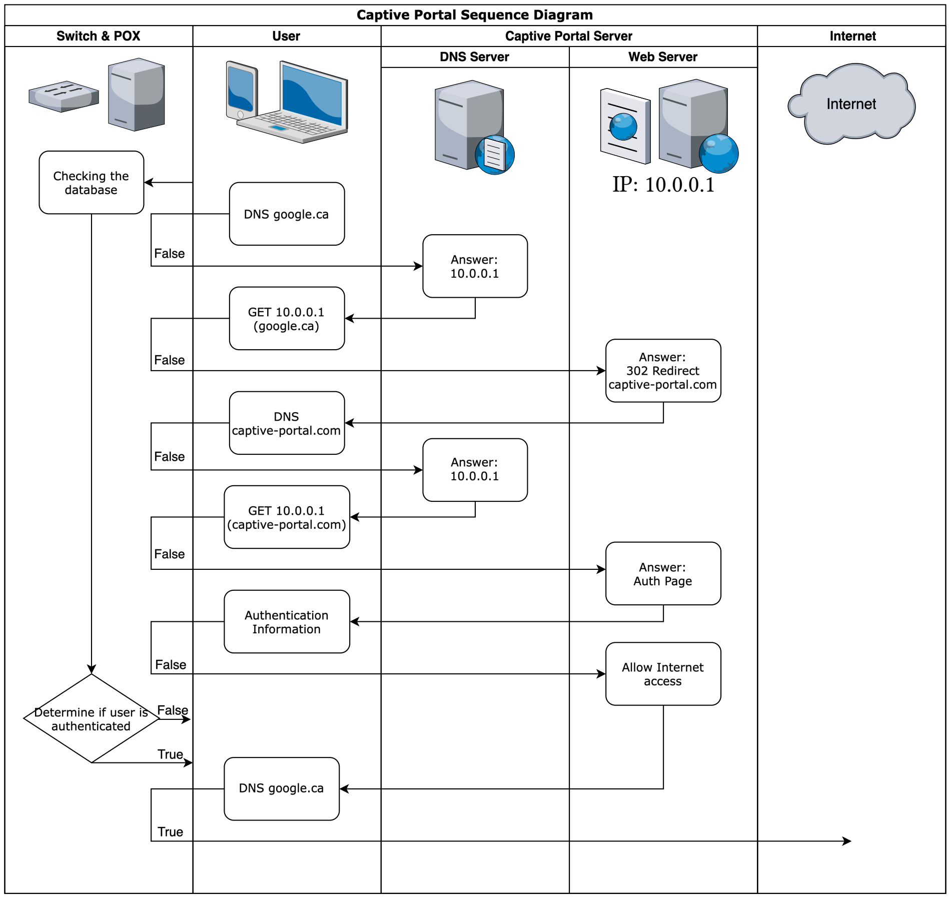



**Figure 2: Sequence Diagram of Captive Portal with DNS Spoofing Technology**

## 4 MODULES

In this section we describe the main modules/projects we have originally designed with the goal of guiding interested readers through the different instances of the Captive Portal technology.

The details of the implementations, along with specific examples, demos and any software and configuration details are available in our public repository, https://github.com/Lianting-Wang/Captive-Portal-Education.

Further details about these are also available in Appendix A.

- **Module 1 - TCP Sever and Client**: This module lays the foundational understanding of TCP connections, vital for implementing captive portal projects. It aims to solidify students' network programming skills and introduce them to Python coding for network applications.
- **Module 2 - SDN Switch**: Focuses on understanding Software-Defined Networking (SDN) controllers and the manipulation of network traffic. Through hands-on practice with the POX SDN Controller, students will learn how to program switches and process network packets.
- **Module 3 - DNS Server**: Introduces the functionality and implementation of DNS servers within the context of captive portal technology. This module covers different approaches to handling DNS requests, emphasizing hands-on experience with DNS manipulation.
- **Module 4 - Web Server**: Explores the deployment and configuration of a local web server to interact directly with network requests, crucial for the operation of a captive portal.
- **Module 5 - Mininet Script**: Delves into network simulation using Mininet, teaching students to construct and modify network topologies to route user requests through specific servers, mimicking a captive portal environment.

## 5 CODE PROVISIONING TOOLS

To facilitate a streamlined educational experience in Captive Portal development, we have implemented a comprehensive suite of code provisioning tools, designed to integrate seamlessly into academic settings. This suite includes a meticulously crafted flowchart –see Fig. 5– and an accompanying website, which together serve as a central hub for instructional resources and starter code packages tailored to various teaching objectives. These tools, in addition to other elements, like demos and tutorials can be found in our publicly available repository:

https://github.com/Lianting-Wang/Captive-Portal-Education.

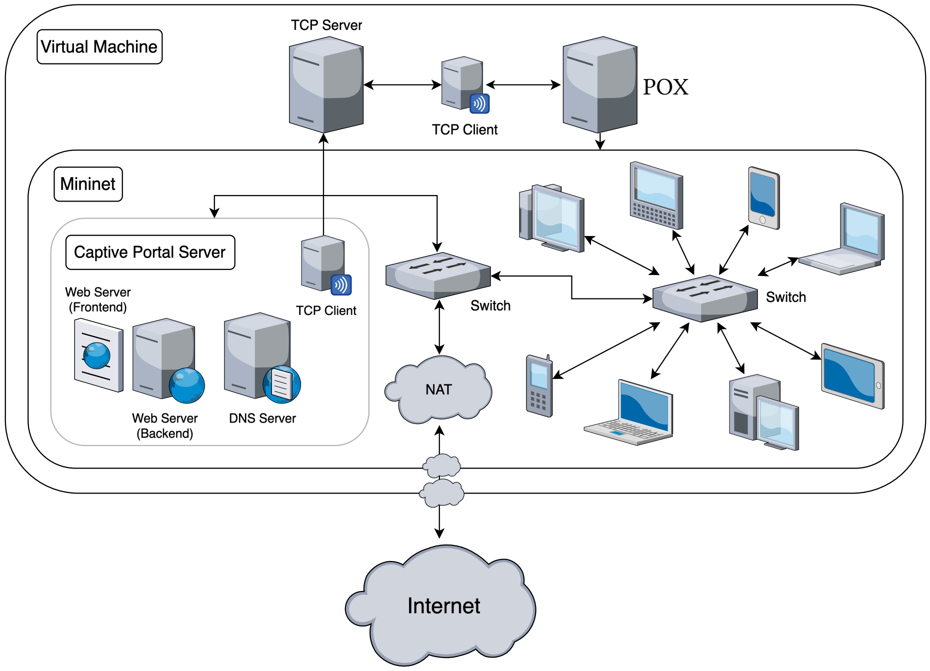



**Figure 3: Framework of Captive Portal project.**

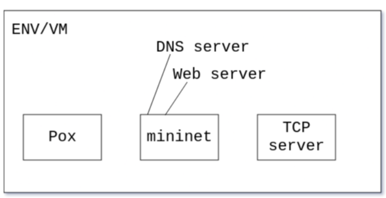



**Figure 4: Simplified view of the main components of the Captive Portal setup as described in Fig. 3: POX controler, mininet SDN emulator, and TCP server.**

In addition to that, we have developed an automated script to assist users with the setup process, as well as, an automated test script, which also verifies whether the environment is configured correctly.

## 5.1 Flowchart Design

The flowchart, Fig. 5, is engineered to guide educators through a structured selection process, enabling them to identify and choose the most appropriate instructional materials and starting code bases that align with their specific teaching goals and the learning outcomes desired for their students. This visual tool simplifies the complexity of course design by categorizing materials based on topics, difficulty levels, and the progression of concepts essential to Captive Portal development.

## 5.2 Website Interface

Complementing the flowchart, the website offers an intuitive, user-friendly platform where educators can effortlessly download pre-packaged instructional materials and starting code. This eliminates the traditionally time-consuming process of manual compilation and customization of teaching resources. Educators can make selections based on the guidance provided by the flowchart, with the website offering detailed descriptions, download links, and additional resources for each package.

## 5.3 Key Features

- Customization: Educators can customize packages to suit the unique needs of their curriculum, allowing for flexibility in teaching advanced topics or focusing on foundational skills.
- Accessibility: The website ensures easy access to up-to-date materials, fostering an adaptive learning environment that can quickly respond to the evolving landscape of Captive Portal technologies.

# 6 DISCUSSION

The transition to environments supportive of Apple's M-series chips has unveiled challenges, notably in virtualization capabilities crucial for educational settings. Our ongoing efforts to integrate Docker as a foundational component for our instructional environment have encountered obstacles, particularly with the integration of the X Window System. Despite these hurdles, we remain optimistic about resolving these issues, ensuring a seamless experience across diverse hardware platforms.

An alternative to be considered for Apple silicon, would be alternative emulation environments like the one offered by EMU

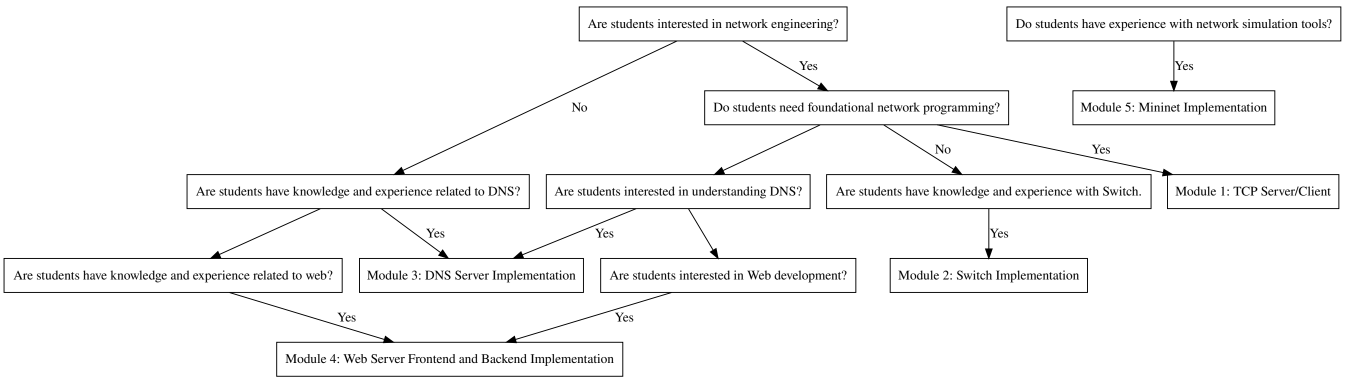



**Figure 5: Captive Portal education modules *recomendation system* – flowchart used to determine what modules are recommended to use based on the desired educational targets and goals. Also available as an interactive tool in the web-interface from the Captive Portal educational repository.**

(https://getutm.app/) or UTM (https://www.qemu.org/). In these cases, it would require porting the whole infrastructure associated to the SDN layer and additional components of the Captive Portal implementation.

In the past, we have also students attempting a fresh installation and compilation of mininet on native Mac OS with relative success although this it not complexly clear is a solution that implies a substantial efforts commitment as well as is not clear whether it will be sustained in time.

We believe this is important information to share from educators perspective, as it has proven to be an additional annoyance for some students to deal with. The palliative solution we offer some of our students is to use Linux based lab. machines, which can be accessed in-situ as well as remotely – for in which cases, we will teaching technologies such as remote connectivity via ssh, X-forwarding, VNC and even VPN due to universities policies and cyber-security best practices [8].

In addition to platform compatibility concerns, our evaluation of Mininet –sometimes also considered as a lightweight network simulation tool– has revealed limitations in its capacity to simulate hardware intricacies accurately. This shortfall has prompted a quest for more sophisticated alternatives capable of delivering a purer simulation experience. Our aim is to enrich the educational experience by offering students hands-on exposure to configuring a variety of network protocols from the ground up. This endeavor seeks to bridge theoretical knowledge with practical application, enhancing the comprehensiveness of our instructional approach.

## 7 CONCLUSIONS

In this work we present a module-based project aiming to familiarize students with Captive Portal technologies. We do so, by using a series of small problems employing the mininet SDN tool.

Our project is open source and hosted in publicly available repository, emphasizing relevant aspects of its open source approach, such as reproducibility, modularity and sustainability. Furthermore, because of its openness, it is possible to continue to grow this project, by adding more modules, accepting contributions from the community, etc.

In conclusion, the development and implementation of a Captive Portal model for educational purposes embodies a concrete contribution to enrich computing education. By integrating comprehensive technical content related to computer networks, this model not only illuminates the operational essence of Captive Portals but also paves the way for additional educational practices in the field of computing and information technology.

## REFERENCES

[1] Suzan Ali, Tousif Osman, Mohammad Mannan, and Amr Youssef. 2019. On Privacy Risks of Public WiFi Captive Portals. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2019 International Workshops, DPM 2019 and CBT 2019, Luxembourg, September 26–27, 2019, Proceedings* (Luxembourg, Luxembourg). Springer-Verlag, Berlin, Heidelberg, 80–98. https://doi.org/10.1007/978-3-030-31500-9_6

[2] Ramon dos Reis Fontes and Christian Esteve Rothenberg. 2016. Mininet-WiFi: A Platform for Hybrid Physical-Virtual Software-Defined Wireless Networking Research. In *Proceedings of the 2016 ACM SIGCOMM Conference* (Florianopolis, Brazil) *(SIGCOMM '16)*. Association for Computing Machinery, New York, NY, USA, 607–608. https://doi.org/10.1145/2934872.2959070

[3] Anita Heijltjes, Tamara Van Gog, Jimmie Leppink, and Fred Paas. 2015. Unraveling the effects of critical thinking instructions, practice, and self-explanation on students' reasoning performance. *Instructional Science* 43 (2015), 487–506.

[4] Christopher G. Jones. 2004. Test-driven development goes to school. *J. Comput. Sci. Coll.* 20, 1 (oct 2004), 220–231.

[5] Faris Keti and Shavan Askar. 2015. Emulation of Software Defined Networks Using Mininet in Different Simulation Environments. In *2015 6th International Conference on Intelligent Systems, Modelling and Simulation*. 205–210. https://doi.org/10.1109/ISMS.2015.46

[6] K Larose, D Dolson, and H Liu. 2020. RFC 8952: Captive Portal Architecture.

[7] Beatriz Pérez and Ángel L. Rubio. 2020. A Project-Based Learning Approach for Enhancing Learning Skills and Motivation in Software Engineering. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education* (Portland, OR, USA) *(SIGCSE '20)*. Association for Computing Machinery, New York, NY, USA, 309–315. https://doi.org/10.1145/3328778.3366891

[8] Marcelo Ponce and Ramses van Zon. 2023. Cybersecurity Training for Users of Remote Computing. *The Journal of Computational Science Education* 14 (Nov. 2023), 10–17. Issue 2. https://doi.org/10.22369/issn.2153-4136/14/2/3

[9] Zuo Xiang and Patrick Seeling. 2020. Chapter 11 - Mininet: an instant virtual network on your computer. In *Computing in Communication Networks*, Frank H.P. Fitzek, Fabrizio Granelli, and Patrick Seeling (Eds.). Academic Press, 219–230. https://doi.org/10.1016/B978-0-12-820488-7.00025-6

## A MODULES DETAILS

The following provides details about the initial set of modules created for the Captive Portal education project.

| **Module 1: TCP Server and Client** |
|---|
| This module lays the foundational understanding of TCP connections, vital for implementing captive portal projects. It aims to solidify students' network programming skills and introduce them to Python coding for network applications. |

*Objectives*

- Comprehend the architecture and function of TCP servers and clients.
- Understand the reason for using TCP to exchange information in Captive Portal.
- Have a general understanding of the entire Captive Portal framework.

| Pre-requisites | Implementations | Learning Outcomes |
|---|---|---|
| Basic understanding of TCP packets and python scripting. | Set up a TCP client in Mininet and enable it to communicate with a TCP server hosted in the virtual machine. | Students will gain hands-on experience with network programming, enhancing their understanding of how data is reliably transmitted across a network. |

| **Module 2: SDN Switch** |
|---|
| Focuses on understanding Software-Defined Networking (SDN) controllers and the manipulation of network traffic. Through hands-on practice with the POX SDN Controller, students will learn how to program switches and process network packets. |

*Objectives*

- Achieve proficiency in utilizing the POX controller to program a network switch.
- Explore the feasibility of modifying a standard switch to act as a captive portal.

| Pre-requisites | Implementations | Learning Outcomes |
|---|---|---|
| Basic knowledge of networking concepts, including switches, MAC addresses, and packet flow. | Students are provided with a basic POX code framework and tasked with implementing a learning switch using POX, thus applying their theoretical knowledge in a practical setting. | Enhanced understanding of SDN controllers and the ability to manipulate network traffic for specific applications such as captive portals. |

| **Module 3: DNS Server** |
|---|
| Introduces the functionality and implementation of DNS servers within the context of captive portal technology. This module covers different approaches to handling DNS requests, emphasizing hands-on experience with DNS manipulation. |

*Objectives*

- Learn to uniformly respond to all DNS queries with a specific IP address.
- Develop skills to parse DNS requests, make requests on behalf of the user, and return genuine responses.
- Modify DNS request destination IPs using iptables for direct server responses.

| Pre-requisites | Implementations | Learning Outcomes |
|---|---|---|
| Basic knowledge of UDP packets and DNS protocols. | Students can explore one of three different DNS configuration methods, from simple IP redirection to complex request handling and response modification. | Students will understand how DNS servers can be manipulated to redirect or handle requests in a captive portal setup. |

**Module 4: Web Server**

Explores the deployment and configuration of a local web server to interact directly with network requests, crucial for the operation of a captive portal.

*Objectives*

- Set up and configure a local web server.
- Use iptables to redirect web requests to the local server.

| Pre-requisites | Implementations | Learning Outcomes |
|---|---|---|
| Basic understanding of web server operation and network request handling. | Students will configure iptables to redirect web requests to a local server, ensuring direct response capability. | Practical skills in web server setup and configuration, with an emphasis on interaction with network requests in a captive portal context. |

**Module 5: Mininet Script**

Delves into network simulation using Mininet, teaching students to construct and modify network topologies to route user requests through specific servers, mimicking a captive portal environment.

*Objectives*

- Understand network structure and the necessary modifications for captive portal functionality.
- Learn to use Mininet for creating and testing network topologies.

| Pre-requisites | Implementations | Learning Outcomes |
|---|---|---|
| Basic knowledge of network topology and mininet. | Students have the flexibility to design the network according to the requirements of the captive portal setup they envision. | Ability to simulate and test network configurations, crucial for the development and deployment of captive portal technologies. |

# B MANUAL CONFIGURATION & GUI ATTACHMENT

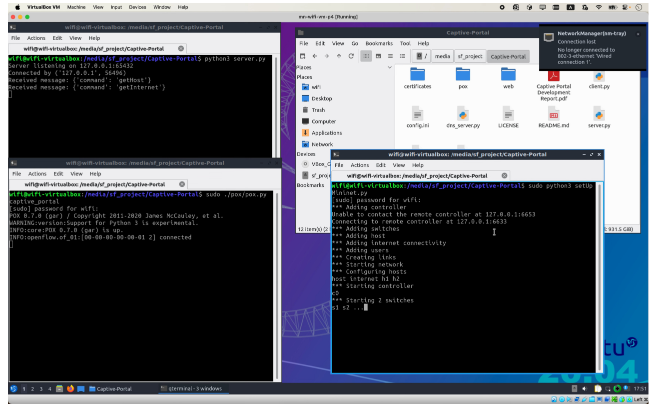
For advanced courses and users of Mininet who wish to gain deeper insights into the technology and procedural setup, the following steps provide a brief introduction to run Mininet with a graphical user interface and additional networking tools:

(1) Initiate the virtual environment that hosts the Mininet installation.
(2) Start the POX controller and server setup as shown in Fig. 6a.
(3) Open additional terminals or sessions to run the corresponding web and DNS servers as in Fig. 6b.
(4) Launch an X-terminal to run a web-browser and optionally a network packet sniffer, such as `wireshark`, as illustratged in Fig. 6c.

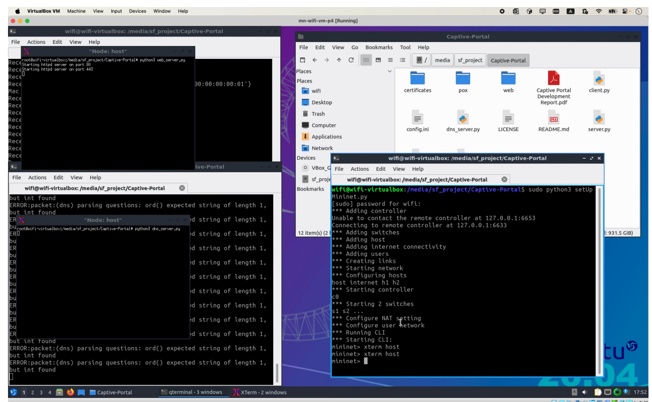
This process allows users to manually configure and customize their Mininet setup, including the option to run graphical user interfaces (GUIs), trigger browser pop-ups typical of Captive Portals, or capture network traffic using `Wireshark`[1]. It also enables users to explore other Mininet base configurations, such as wireless mininet setup [2].

However, it is important to note that these steps are meant to demonstrate what the final product will look like once students have completed all or part of their project code, regardless of how many modules the students complete, the final product will be a fully functional Captive Portal, it is not an exact reflection of how students will test their code.

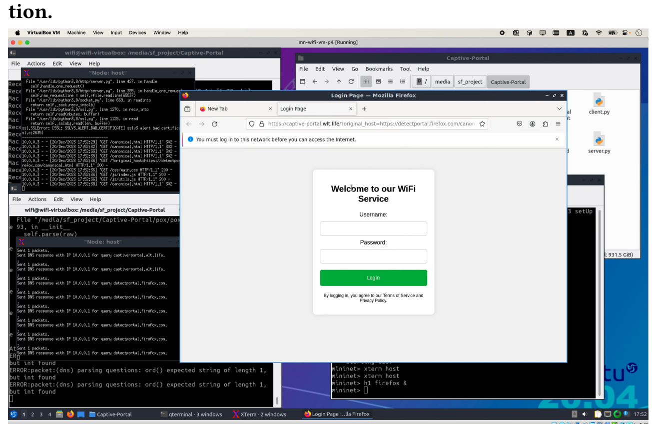
Notice that to properly demonstrate these capabilities, some additional configuration steps are required beyond what is shown in the images. These additional steps, as well as more specific configuration details, will be provided in the project's GitHub repository.



**(a) Configuration setup in a VM for running mininet as a Captive Portal.**



**(b) Setup of DNS server with the mininet Captive Portal configuration.**




**(c) Demonstration of the DNS server interception of network traffic via a web-browser request.**

**Figure 6: Demonstration of the sequence of steps required to setup and execute the DNS spoofing technology to forge the Captive Portal allowance.**

[1] https://www.wireshark.org/

# From Student SIG to Success: *The Journey of a Student HPC Special Interest Group Towards Sustainable Training and Success in Student Cluster Competitions*


Bryan Johnston
CSIR
bjohnston@csir.co.za

Lara Timm
CSIR
ltimm@csir.co.za

David Macleod
CSIR
dmacleod@csir.co.za

John Poole
Center for Human Genetics, Clemson University
jopoole@clemson.edu

Lily de Melo
University of the Witwatersrand
lily.demelo@outlook.com

Sayfullah Jumoorty
University of the Witwatersrand
sayfullah.jumoorty@live.com

Jonathan Faller
University of the Witwatersrand
jono.faller@gmail.com

Reinhard Jansen van Vuuren
University of the Witwatersrand
reinhardjvv@outlook.com

Michael Beukman
University of Oxford
mbeukman@robots.ox.ac.uk



## ABSTRACT

Developing a sustainable High Performance Computing (HPC) workforce pipeline remains a global priority. As access to HPC resources continues to improve in Africa, the shortage of available skilled HPC personnel is a significant impediment to the adoption of advanced research computing infrastructure. South Africa has several workforce training initiatives aimed at developing career HPC system administrators, yet, apart from the annual national Student Cluster Competition (SCC) – an initiative of South Africa's Centre for High performance Computing (CHPC) – there are no other formal training programmes available for the undergraduate student community.

Each year since the inception of the CHPC-SCC the University of the Witwatersrand ("Wits") has entered at least one team. Through the implementation of an array of student-led training approaches using decommissioned HPC hardware, Wits has enjoyed continued success at these SCC events. Moreover, Wits students have been part of the national teams that have achieved six top-3 finishes in international Student Cluster Competitions.

This paper provides an overview of the student HPC Special Interest Group (SIG) formed at the University of the Witwatersrand that focuses on delivering HPC training to the undergraduate student community. The paper outlines the approach towards growing and maintaining the interest group, including teaching and learning strategies to prepare Wits students for Student Cluster Competitions. Insights into the challenges experienced and lessons learned are discussed, particularly with respect to sustainable workforce development. These insights could help develop a structured framework for creating effective and sustainable HPC special interest groups, centred around student involvement.



## KEYWORDS

Student Cluster Competition, HPC Workforce Development, HPC Education, HPC Training, Africa


## 1 INTRODUCTION

Sustainable HPC workforce development is widely recognised as a critical component of effective advanced research computing [6, 11]. Until recently, South Africa did not possess a strong workforce pipeline for HPC. Following the inception of South Africa's first national HPC facility, the Centre for High Performance Computing (CHPC) in 2007, a concerted effort needed to be made towards HPC awareness and outreach initiatives.

As a driver for HPC outreach aimed at undergraduate students at national universities in South Africa the CHPC's national Student Cluster Competition (CHPC- SCC) was launched in 2012. Through the SCC pipeline, South Africa has enjoyed notable success in the ISC Student Cluster Competition, co-organised by the HPC-AI Advisory Council and ISC Group (ISC-SCC).

This paper showcases the University of the Witwatersrand's student HPC Special Interest Group (SIG). The approaches taken towards sustainable student HPC training, the lessons learned, the challenges encountered, and a brief overview of plans for expanding the student SIG model for other universities in South Africa are discussed.

### 1.1 Importance of HPC Outreach

It is well established that both outreach and education play important roles in accelerating optimal use of advanced research computing resources and ensuring a sustainable workforce pipeline [3]. Hernandez et al. note that the resources encountered during one's collegiate education shapes the tools and techniques that professionals employ throughout their careers [6]. It is important to


https://doi.org/10.22369/issn.2153-4136/16/1/9

expose undergraduate students to HPC concepts as early as possible in their degrees, not only to enhance accessibility to HPC skills and resources, but to inspire undergraduate students to make informed decisions in their course selection to enable the pursuit of careers in HPC.

### 1.2 Student Cluster Competition

The SCC concept began in 2007 [1] during the SC Conference Series [2] as an educational tool to immerse undergraduate students in HPC [4, 13]. Through the competitive format of the student event, the SCC plays a dual role of HPC outreach and HPC workforce development by introducing participating teams to conventional HPC challenges. Since the initial SC07 event, the SCC has expanded beyond the USA to other international HPC conferences, including Africa [3], Asia [4], and Europe [5].

### 1.3 CHPC Student Cluster Competition

South Africa's national SCC exposes undergraduate students to HPC in an exciting format and aims to motivate students to consider future careers in HPC.

The CHPC-SCC involves a National Selection Round, held annually in the July winter break, where twenty teams (each of four eligible students) participate in a week of intensive HPC system administration training. The training encompasses both theoretical lectures and practical hands-on labs. The Selection Round covers topics relating to HPC fundamentals, culminating in a final theoretical cluster design project and presentation. The top ten performing teams from the Selection Round progress to the National Finals, held annually in December at the CHPC's national conference [6]. During this round each team is provided with physical hardware and is tasked with assembling a small-scale HPC system and undertaking several benchmarking and performance challenges. The overall winning team of four is crowned the national champions and serves as the foundation of the team to represent South Africa at the ISC-SCC. From the remaining teams, a further two students, as well as two reserves, are selected to complete the national team.

Prior to competing in ISC-SCC, the team travels to the USA for a week of focused technical training and cluster design preparation.

Through its Student Cluster Competition initiative, the CHPC not only develops technical skills in HPC but also cultivates a strong HPC community. Motivating students to pursue careers in scientific computing contributes to a sustainable HPC workforce pipeline in South Africa.

## 2 WORKFORCE DEVELOPMENT

There are three primary considerations to ensure the sustainable use of advanced research computing:

(1) A user community with the need for, and ability to use, advanced research computing resources.
(2) A skilled technical workforce to deploy, administer, and maintain the resources.
(3) A pipeline to replenish the workforce and retain the core competencies.

[1] https://www.youtube.com/watch?v=NhSWMSp096c
[2] https://supercomputing.org/
[3] https://scc.chpc.ac.za/
[4] http://www.asc-events.org/StudentChallenge/index.html
[5] https://isc-hpc.com/program/student-cluster-competition/
[6] https://chpcconf.co.za/

**Table 1: CHPC Results at ISC-SCC 2013-2020**

| Year | Result | Year | Result |
|---|---|---|---|
| ISC 2013 | 1st | ISC-2017 | 2nd |
| ISC-2014 | 1st | ISC-2018 | 3rd |
| ISC-2015 | 2nd | ISC-2019 | 1st |
| ISC-2016 | 1st | ISC-2020 | 2nd |

### 2.1 Building A Workforce Pipeline Through Outreach

The CHPC restricts SCC participants to undergraduate students in the early stages of their degrees to amplify and maximise early curriculum intervention and outreach impact [11].

Prior to the COVID-19 pandemic, from 2012 to 2019, the CHPC-SCC met its national outreach objectives by introducing more than 600 undergraduate students to HPC. In addition, the South African national teams achieved international success at the ISC-SCC, placing in the top three in each of these years (see Table 1). Students from Wits fielded 28 of the 56 (50%) international team members.

### 2.2 Achieving Outreach Through Student Competition

Adopting a competition model for building a scientific computing workforce provides numerous benefits:

- Competition provides an engaging opportunity for students to work together in teams, strengthening their social and collaborative skills.
- Team assignments closely mirror professional HPC centre scenarios and require students to think critically and work autonomously [4, 5].
- Working closely in their teams against a community of their peers, students are motivated to excel through the high stakes of elimination and reward.

Additionally, the CHPC-SCC is an effective outreach tool because of the attractive opportunity to be selected for the ISC-SCC team. The ISC-SCC team is supported to travel to the USA for additional training and then again to ISC to compete and experience one of the largest HPC conferences.

## 3 BUILDING THE WITS HPC SIG

Many HPC workforce training initiatives catering for students are organised in a top-down fashion. In contrast, Wits formed a student special interest group dedicated towards HPC training for undergraduate students that is primarily run by the student volunteers [2].

The SIG has matured to perform three primary roles:

(1) Preparing and selecting the University's participating teams for the CHPC-SCC.
(2) Delivering HPC training to undergraduates, with a focus on practical skills that are not typically addressed in the standard Computer Science curriculum.

(3) Developing the student organisers through the experience of leading and training their younger peers.

## 3.1 The Wits SIG Origin Story

Wits had an established history with HPC prior to the creation of the CHPC in 2007. In 2010, the Wits Mathematical Sciences Support team (MSS) formed several interest groups, including the Wits HPC Interest Group. The group was created to facilitate collaboration and expand access to scientific computing, with a specific focus on leveraging HPC to address complex scientific challenges faced by researchers at the university.

A pivotal moment came with the introduction of the CHPC-SCC in 2012. In the first year of the competition, students entered as individuals and the CHPC assigned them to teams of four. The inaugural winning team included three Wits students. In preparation for ISC'13 the team sought additional computational resources from Wits. Their subsequent victory at ISC'13 brought recognition and secured hardware donations, catalysing the growth of the HPC Interest Group.

Prior to 2021 the group's primary focus was competition preparation and mentorship for the CHPC-SCC. In 2021, the group shifted focus towards a more formal training model incorporating lectures.

The group continues to evolve, relying heavily on mentorship from former ISC participants, under the guidance of the School of Computer Science and Applied Mathematics.

# 4 A MODEL FOR A STUDENT-LED HPC TRAINING PIPELINE

While numerous HPC student interest groups exist at universities, there is a scarcity of case studies of student SIGs addressing an HPC training pipeline. Pertinently, the Wits SIG, with the Mathematical Sciences Support team, has developed a training model that has been consistently tried and tested.

Due to the continual succession in leadership, the SIG's structure and format has seen incremental evolution over many years. The model presented in this paper focuses on the current approach used by the SIG.

## 4.1 Creating Awareness for the SIG

A fundamental criterion of success for an organised group is its ability to recruit and maintain members. As members graduate and leave the SIG, a consistent recruitment effort is needed to replenish the membership of the group.

Previously, interest in the SIG was created by past students of the CHPC-SCC targeting undergraduate student mailing lists at various faculties. While this attracted some interest, it was primarily the encouragement from past students who had competed in the CHPC-SCC that enticed their peers to participate during the winter vacation period.

As the SIG has grown into something more than a competition recruitment strategy, new methods of advertising and awareness creation have been introduced. These include: a SIG Discord channel, Instagram account, Linkedin page, WhatsApp group, posters, a website [7], and an exhibition stand at the Science Career Fair. Invitations to join these groups or channels are distributed within university mailing lists by current members and on other social platforms within the university. While creating awareness in this way has brought in some interest, students report that the most interest and participation is still gained by word of mouth.

## 4.2 SIG Structure and Leadership

The SIG is student-run, with a leadership structure that is more collaborative than hierarchical. There are no formal titles; the group functions on a volunteer basis, driven by its members. Each year, students who were part of the previous year's ISC-SCC team take on the role of organisers, guiding the next cohort of students. Organisers take on a variety of responsibilities, including developing content for talks, managing social media, and overseeing team selection and preparation. This flexible structure has proven effective, allowing the group to easily change its methods to suit the current circumstances. Since there isn't a single figure of authority determining the group's direction, members are more innovative and open to exploring new ideas, where decisions are made collectively.

## 4.3 Training and Activities

Prior to 2020, the exclusive focus of the SIG was to prepare students for excelling at the CHPC-SCC [2]. A decision was made in 2020 to extend the reach of the SIG to cater for a wider range of interests. Subsequently, general HPC-themed talks and practicals were introduced to expand the group's mission to provide broader HPC education to all interested students. The expanded talks and workshops are publicly available and hosted on a dedicated SIG GitHub repository [8].

The SIG follows the two semesters of the academic year, with weekly meetups held during the campus lunch hour. The first semester focuses on introductory-level HPC talks, with each talk accompanied by a tutorial relating to the content. Presenting content in this format aims to equip students with the practical skills required to use a particular tool or to perform a particular technique immediately. Following the preparatory talks, team selection for the CHPC-SCC Selection Round is concluded with a test (outlined in Section 4.4).

The second semester begins after the CHPC-SCC Selection Round concludes, and the SIG's focus shifts to broader topics with more emphasis on preparing any progressing teams for the CHPC National Finals (outlined in Section 4.5).

The SIG has explored various formats for delivering the training, including traditional SIG-led lectures, guest lectures from HPC experts, practical-styled interactive sessions, and tutorial-styled sessions that combine lectures with practical exercises.

Following feedback and further research into effective practical HPC training, the most recent efforts have focused on tutorial-style sessions, where interactive exercises aim to enhance students' engagement and learning while reducing the volume of lectures presented to participants [10, 12]. For instance, the SIG recently ran

[7] https://github.com/WitsHPC
[8] https://github.com/WitsHPC/HPC-InterestGroup

a hackathon event that tasked students with solving a highly parallelisable programming challenge in the fastest possible runtime. [9]

### 4.4 Team Selection

Wits commits to sending two teams to the CHPC-SCC Selection Round every year. Team selection begins with a timed open-book placement test, which consists of extensive Linux-based and shell-scripting questions. The highest-scoring students are selected to represent the university. To ensure well-balanced teams, the organisers review students' motivations and transcripts. Once teams are formed, SIG organisers act as mentors, dedicating time to assist with cluster design, setup, and testing at Wits. The SIG has investigated preliminary hands-on training before team selections to allow a larger pool of students an opportunity to be selected by bridging the technical knowledge gap. To complement this, the SIG holds regular practical tutorials that are also considered when selecting teams.

Most recently, the group has experimented with a project instead of a placement test. The project utilises the HPC Ecosystems Project's OpenHPC 2.x virtual lab [8] to set up a local virtual 3-node cluster using Vagrant and VirtualBox. Students had four weeks to run a small custom parallel application on their virtual clusters, including build and run scripts, to simulate aspects of a student cluster competition. Students were scored based on the functionality of their scripts, with a significant allocation to the fastest runtime. The project had several advantages over the placement test: it attracted better-prepared applicants, and it further enhanced their preparation for the Selection Round; however, there were also some drawbacks: participation rates were lower than usual, and organisers had to manually test the run scripts to ensure no hardware advantages impacted the results. Moving forward, the aim is to find a balance between tests and projects to optimise participation and learning outcomes.

The implementation of a multifaceted selection process increases the likelihood of choosing motivated and balanced student teams. Where some students may not have achieved top scores in the placement test, they have an opportunity to demonstrate their willingness to learn and improve through their performance in the hands-on practicals.

### 4.5 Team Preparation

Following the CHPC-SCC Selection Round, the SIG focuses attention towards preparing the Wits teams progressing to the CHPC National Finals. To facilitate hands-on practice, the teams are provided with access to HPC training systems. The training systems are built from decommissioned HPC production nodes [7, 9].

Hands-on access to physical systems provides a comprehensive learning experience to gain the technical skills required to build, deploy, and optimise an HPC cluster. These skills include, inter alia, operating system installation, network configuration, package management, and Linux shell.

Using their own deployed systems, the teams learn to compile software from source and to run distributed applications. During this training phase, the SIG mentors serve as guides for the students, giving them high-level technical advice and objectives. With this guidance, the teams learn how to identify and control factors affecting application and system performance. The extensive hands-on time produces teams well-versed in competition expectations and capable of solving problems rapidly and effectively [1].

[9] https://github.com/WitsHPC/HPC-InterestGroup/tree/main/assorted/competitions/2023_cuda/problem

## 5 CURRICULUM

One goal of the SIG is to be as inclusive as possible. To achieve this, training material assumes no prior knowledge or experience in any of the areas relevant to HPC. Accordingly, considerable emphasis is placed on introduction to essential concepts. More advanced topics are left to later stages of the SCC training or for further self-study to avoid overwhelming beginners.

The lessons are designed to be modular and include sufficient theoretical context to provide fundamental understanding. Focus is placed on developing practical skills and providing insights into best practices. This philosophy of learning by doing is well suited to the practice of HPC system administration and performance tuning. The curriculum equips students to excel in the Selection Round of the CHPC-SCC and is shown in Table 2, classified by topic and assessment type.

The curriculum has evolved over multiple iterations through contributions by Wits SCC participants based on areas of interest and available expertise. It accommodates a variety of teaching modalities (online, in-person, asynchronous, talks, practicals, labs, and quizzes) across a wide range of topics at different levels of depth and inherent difficulty.

From 2021 the core course materials are consolidated in the public GitHub repository which contains the SIG talks, tutorials, and materials for self-study. Some of the material is maintained privately out of necessity (for instance, the SIG quizzes). This includes more advanced material, including solutions that are relevant to CHPC-SCC preparations, and content that is central to the team's strategy at ISC-SCC. The GitHub approach facilitates collaboration and accessibility and lends itself to iterative development. The repository contains both original work as well as material collected or reformatted from publicly available sources. Quizzes, student submissions, and student discussions are hosted on Moodle LMS.

## 6 STATISTICS

The SIG facilitates a virtual community through Discord, and membership stands at 366 past and present students as of April 2024.

**Table 3: Wits SIG results in SCC**

| | 1st | 2nd | 3rd | Total Events |
|---|---|---|---|---|
| CHPC-SCC (National) 2014-2023 | 7 | 5 | 4 | 11 |
| ISC-SCC (International) 2013-2023 | 2 | 2 | 1 | 11 |

Table 3 shows the performance of Wits participants at the CHPC-SCC and ISC-SCC events. The ISC (International) row shows results from national teams where Wits comprises more than half of the members. The Total Events column reflects the number of SCC

**Table 2: Wits SIG HPC Curriculum (1st and 2nd Semester)**

| **Lecture** | **Topics Covered** | **Tutorial (T) or Quiz (Q), if any.** | |
|---|---|---|---|
| Introduction to HPC | Fundamental concepts of HPC and its applications. | - | - |
| Linux and shell | Basic shell commands and their usage in Linux. | Use the terminal to answer questions | Q |
| Hardware Basics | Understanding hardware components and how to make informed decisions about cluster design. | Technical questions about hardware, requiring investigation of hardware specifications and computing centres | Q |
| Compiling Applications | Building and running applications from source (compiler flags, linking libraries). | Compile a simple application (e.g., \`htop\`). | T |
| Parallel Computing and MPI | A high-level overview of MPI, OpenMP, and CUDA. | Compile and run a simple MPI-enabled program. | T |
| Advanced HPC System Tools | Containerisation and module management. | Hands-on implementation of these tools. | T |
| Benchmarking$^{\alpha}$ | Understanding benchmarking applications. | Run HPL. | T |
| Scripting$^{\alpha}$ | Overview of scripting and best practices. | Script a small application installation. | T |
| Version Control$^{\alpha}$ | Using Git and GitHub effectively. | - | - |
| Makefiles$^{\alpha}$ | Creating and using Makefiles. | - | - |
| HPC Best Practices$^{\alpha}$ | Tools, techniques, and approaches relevant to reliable and efficient operation of HPC clusters. | - | - |

**Key: $^{\alpha}$ denotes 2nd semester**

events, noting that Wits had multiple teams at each CHPC-SCC. The strong results reflect the SIG's effectiveness in supporting world-class competencies in HPC system administration and optimisation.

From 2014 to 2023 teams from Wits have won the National Round seven times, with the non-winning Wits teams securing a further five silver and four bronze medals (Wits traditionally sends two teams to the National Round). Over the past ten years, more than half of all members of the CHPC national teams competing in ISC-SCC have been students from the Wits SIG.

Most of the SIG membership originates from two University schools: the School of Computer Science and Applied Mathematics; and the School of Electrical and Information Engineering. The SIG comprises of first to fourth-year students, with the majority being in their second year of study. The organising committee are in their third year of study or later.

In the first semester, weekly talks usually attract 20 students, with as many as 30 students participating in the hands-on tutorials. After the winter break (which coincides with the conclusion of the CHPC-SCC Selection Round), weekly participation drops significantly to around 10 participants on average.

# 7 CHALLENGES

The predominant challenge faced by the SIG is its sustainability. Sustainability is affected by three primary factors: group membership attrition, leadership succession, and remote delivery.

## 7.1 Attrition

The SIG experiences a sustained reduction in attendance throughout the course of the year. Feedback alludes to three reasons for student withdrawal:

(1) After final team selection, the unplaced students lose motivation and subsequently withdraw.
(2) As the academic year progresses, students prioritise academic coursework over the SIG.
(3) As topics become more advanced, student interest in the SIG diminishes.

## 7.2 Succession

As a student body, the organising committee is constantly affected by members graduating and leaving, which necessitates recurring recruitment of new mentors. Ensuring continuity in leadership is a recurring challenge.

To mitigate this challenge, promising students are selected to co-organise with more senior members, who take on a more direct mentorship role. The mentorship model provides a gradual exposure to leadership roles while spreading the workload for the current leadership. Once the inevitable graduation of senior students occurs, the replacement organisers already have the requisite experience.

### 7.3 Remote Delivery

Remote delivery was adopted during the COVID-19 pandemic, transitioning all SIG activities to online platforms. While the improved accessibility enabled broader outreach and scalability of the SIG's activities, it led to lower engagement levels compared to in-person sessions. Consequently, the SIG delivers most talks in-person on campus. When the goal of an event is to reach a wider audience, the sessions are conducted online.

## 8 RISKS

The SIG's primary appeal remains the opportunity to participate in the CHPC-SCC and ultimately to compete in the ISC-SCC. The regular success of the Wits teams at the National Finals encourages new students to enrol. With the current succession model of past students returning to mentor the new cohort, the strength of the SIG is closely correlated to the performance of the previous year's teams in the SCC. It is possible that if a cohort fails to emulate the success of the previous year, there could be a reduction in the strength of the mentorship for the subsequent year. Further, it is possible that a relatively poor performance may also discourage students from joining the next year.

While the CHPC-SCC sets diversity requirements for participating teams, the Wits SIG leadership struggles to retain female representation which may pose a threat to long-term retention of female involvement in the SIG.

The SIG enjoys support from the University through access to HPC resources, but the absence of any formal funding does leave the SIG heavily reliant on the continued goodwill of the University for hardware and training resources. In particular, the use of years-old repurposed decommissioned HPC hardware is prone to hardware failures, leading to a diminishing supply of available HPC resources for hands-on practice.

## 9 LESSONS LEARNED AND REFLECTIONS

Past competitors returning as mentors has proven to be effective in leveraging the firsthand experience gained in the competition, ensuring future teams remain well-prepared and competitive. To strengthen knowledge transfer resilience the SIG has a strong culture of opensource documentation, storing past and current resources in the SIG GitHub.

Under the guidance of past organisers, the leadership handover process has proven crucial to equipping the incoming organising team. Since the new organisers have been participants in the group's activities in the preceding year, they are familiar with the responsibilities and methods of execution.

Notwithstanding all efforts to mitigate interruption in leadership succession, the fact remains that the SIG is operated on a voluntary basis by students whose primary responsibilities are their studies. While the SIG remains primarily led by student volunteers, there is a risk to the SIG sustainability. At the risk of volunteers withdrawing, despite the strengths of a student SIG, the group's longevity is inextricably linked to sustained institutional support.

The current model for recruiting new SIG members depends on the CHPC-SCC to attract students to participate. While there is no indication that ISC-SCC or CHPC-SCC will end, the SIG is cognisant of this inter-dependence.

A perfunctory assessment of the SIG's success at both the CHPC-SCC and ISC-SCC could lead to the conclusion that the Wits SIG is a successful model for building an effective student HPC pipeline. While this may indeed be true, such an assessment fails to consider circumstances at Wits which may be contributory factors to this success. For one, Wits is among the top-ranked universities in Africa and is recognised as one of the top Computer Science and Engineering universities in South Africa [14]. The strength of the Wits academic programme likely attracts good students, and these good students will excel against their peer institutions.

The Wits SIG owes its origins and much of its ongoing popularity to the incentive of participating in the ISC-SCC. Reaching an international cluster competition like ISC-SCC is well beyond the means of most institutions. Likewise, even hosting a local SCC requires significant investment in time, resources, and funding.

## 10 FUTURE WORK

A natural consequence to students not being selected for the CHPC-SCC is a drop in engagement with the SIG. The organisers hope to improve engagement by undertaking several interventions.

Hosting additional internal competitions throughout the year will enable more students to experience the competition component, resulting in improved student engagement and retention in the SIG.

The introduction of long-term projects may also produce stronger participation and engagement. By encouraging students to consistently invest time throughout the year, they will build incrementally towards a concrete but ambitious end goal.

While local HPC resources are currently reserved for the use of the SCC teams, the goal is to expand access for the general (non-SCC) SIG members.

To maintain engagement throughout the year, the organisers would like to introduce certificates of completion for students who complete the SIG training. Additional non-technical content, such as career panels, will be included in future SIG activities. Organisers currently contribute to the broader HPC community through presentations and collaboration with the HPC Ecosystems Project [7, 9]. The SIG intends to expand its involvement in this project through the development of additional training content for the community.

## 11 CONCLUSION

The University of the Witwatersrand HPC Special Interest Group is a student group that seeks to provide advanced scientific computing skills to undergraduate students. The SIG focuses on three areas: exposing undergraduate students to general HPC education; preparing teams for Student Cluster Competitions; and equipping students with technical skills in HPC.

The Wits SIG has fostered HPC knowledge and technical skills among the student community for more than ten years. The evidence of both local and international student cluster competition achievements may offer some indication of the strength of the SIG's efforts.

This case study aims to provide insight into how to build a sustainable and successful student HPC community. At the very least, aspects of the Wits student SIG model might be of use to others seeking to contribute to the enhancement of HPC education and workforce development.

## ACKNOWLEDGEMENTS

The impact of the Wits student SIG is the result of years of concerted contributions by many individuals. From members of the Mathematical Sciences Support Team to the Student Cluster Competition convenors, past and present, including all the generous sponsors and enablers for the CHPC-SCC and ISC-SCC. In particular, Brian Maistry, Shunmuga Pillay, Senzo Mpungose, Kay Dhlamini, Eugene de Beste, Nick Thorne, Matthew Cawood, Israel Tshililo, Sakhile Masoka.

## REFERENCES

[1] Lluc Alvarez, Eduard Ayguadé, and Filippo Mantovani. 2018. Teaching HPC Systems and Parallel Programming with Small-Scale Clusters. *2018 IEEE/ACM Workshop on Education for High-Performance Computing (EduHPC)* (2018), 1–10. https://api.semanticscholar.org/CorpusID:61811423

[2] Michael Beukman, Sayfullah Jumoorty, Jonathan Faller, Eugene Beste, and Bryan Johnston. 2024. Second Life: Enabling a Student Special Interest Group through Decommissioned HPC Systems. 1–4. https://doi.org/10.1145/3626203.3670573

[3] Caughlin Bohn. 2022. HPC Outreach and Education at Nebraska. In *Practice and Experience in Advanced Research Computing 2022: Revolutionary: Computing, Connections, You (PEARC '22)*. Article 63, 4 pages. https://doi.org/10.1145/3491418.3535128

[4] Stephen L Harrell, Hai Ah Nam, Veronica G. Vergara Larrea, Kurt Keville, and Dan Kamalic. 2015. Student Cluster Competition: A Multi-disciplinary Undergraduate HPC Educational Tool. Los Alamos National Laboratory (LANL). https://www.osti.gov/biblio/1227249

[5] Stephen L. Harrell, Preston M. Smith, Doug Smith, Torsten Hoefler, Anna A. Labutina, and Trinity Overmyer. 2011. Methods of creating student cluster competition teams. In *Proceedings of the 2011 TeraGrid Conference: Extreme Digital Discovery (TG '11)*. Association for Computing Machinery, Article 50. https://doi.org/10.1145/2016741.2016795

[6] Jose Angel Hernandez, Marcelina Nagales, and Paul Van der Mark. 2024. Education and Outreach Strategy Design for High-Performance Computing Centers: Lessons from Florida State University's Research Computing Center. In *Practice and Experience in Advanced Research Computing 2024: Human Powered Computing (PEARC '24)*. Association for Computing Machinery, Article 84, 4 pages. https://doi.org/10.1145/3626203.3670587

[7] Bryan Johnston. 2019. HPC Ecosystems Project: Facilitating Advanced Research Computing in Africa *(PEARC '19)*. Association for Computing Machinery, Article 107, 3 pages. https://doi.org/10.1145/3332186.3333264

[8] Bryan Johnston, Lara Timm, Eugene de Beste, and Mabatho Hashatsi. 2024. Let's Get Our Heads Out of the Clouds: A scalable and sustainable approach to HPC Training Labs for Resource Constrained Environments and anyone else stuck in the Clouds. *The Journal of Computational Science Education* 15 (2024), 64–71. Issue 1. https://doi.org/10.22369/issn.2153-4136/15/1/13

[9] Bryan Johnston, Lara Timm, David Macleod, and John Poole. 2024. Ten Years of the HPC Ecosystems Project - Transforming HPC in Africa for the past decade. In *Practice and Experience in Advanced Research Computing 2024: Human Powered Computing (PEARC '24)*. Association for Computing Machinery, Article 35, 8 pages. https://doi.org/10.1145/3626203.3670537

[10] Jennifer K. Knight and William B. Wood. 2005. Teaching more by lecturing less. *Cell Biology Education* 4 (2005), 298–310. https://doi.org//10.1187/05-06-0082/ASSET/IMAGES/LARGE/298-FC01.JPEG

[11] Elizabeth Leake. 2020. Student Success from 'Scratch': CHPC's Proof is in the Pudding. Retrieved September 5, 2024 from https://www.hpcwire.com/2020/08/07/student-success-from-scratch-chpcs-proof-is-in-the-pudding/

[12] Anthony McCourt and Caitriona Carr. 2010. Improving Student Engagement and Retention Through Small Group Tutorials. *Innovation in Teaching and Learning in Information and Computer Sciences* 9, 1 (2010), 61–77. https://doi.org/10.11120/ital.2010.09010061

[13] Scientific Computing World. [n. d.]. When students compete, the community wins. Retrieved September 8, 2024 from https://www.scientific-computing.com/feature/when-students-compete-community-wins

[14] Times Higher Education. 2024. World University Rankings: University of the Witwatersrand. Retrieved September 9, 2024 from https://www.timeshighereducation.com/world-university-rankings/university-witwatersrand

# Computational Skills Training for Undergraduate Researchers in Molecular Engineering

Kristen Finch
University of Washington
finchkn@uw.edu

Ryan Beck
University of Washington
rbeck4@uw.edu

Xiaosong Li
University of Washington
xsli@uw.edu

Nam Pho
University of Washington
npho@uw.edu

Xiao Zhu
University of Washington
xiaozhu@uw.edu

## ABSTRACT

In June 2024, the University of Washington's (UW) Clean Energy Institute (CEI) and Molecular Engineering and Materials Center (MEMC) in partnership with UW Research Computing (RC) prepared complimentary training for a group of 25 Research Experience for Undergraduates (REU) participants. Workshop participants had completed zero to four years of post-secondary education and came from 17 colleges and universities across eight states with 29% currently attending 2-year programs. On average, 14 students attended a given workshop. The program included four targeted workshop offerings, spanning essential skills in computational science and advanced topics: (1) Python via Jupyter, (2) Command Line Interface (CLI) and high performance computing (HPC), (3) Gaussian and Quantum Espresso, and (4) data analysis using linear and logistic regression as well as neural networks. The program's effectiveness was evaluated with a post-workshop survey. Survey results indicated most participants had little prior experience in these topics but indicated the content was relevant for their current and future aspirations. The survey showed some students agreed with statements indicating that learning objectives were met, but overall scores and open responses indicated areas for improvement. In the future, the CLI and HPC session will be converted from one to two sessions and the material in the applied Gaussian and Quantum Espresso demonstrations reduced. The program's materials are reproducible and publicly accessible, compatible with most academic HPC clusters. Our program addressed a wide range of training and education needs within computational science, emphasizing practical skills and interdisciplinary applicability.

## KEYWORDS

Undergraduate Research, REU Workshop, Student Training, High Performance Computing, HPC Education, Jupyter Notebook, Linux, Command Line, SLURM, Gaussian, Quantum Espresso, Molec-ular Engineering, Machine Learning


https://doi.org/10.22369/issn.2153-4136/6/1/10

## 1 INTRODUCTION

The UW Clean Energy Institute (CEI) and Molecular Engineering and Materials Center (MEMC) are combining innovative research with comprehensive training to foster broader engagement in STEM fields. As part of their grant-funded activities, CEI and MEMC partnered with UW Research Computing (RC) to provide training for REU students residing on campus in Seattle, WA, USA for an immersive 12-week research experience. Students were recruited for their interest and promise in chemical and molecular engineering. Four workshops were delivered in June 2024, covering a range of topics from fundamental computing and data analysis to more advanced and applied techniques (Figure 1). The topics were carefully curated to support on-boarding and build technical capacity relevant to the students' research projects, as well as to equip them with skills that will be valuable in their future coursework and careers. The workshops primarily utilized the UW RC High-Performance Computing (HPC) cluster, Hyak, and Hyak's implementation of the Open OnDemand [14] platform, ensuring hands-on experience with widely adopted tools. The effectiveness of the workshops was evaluated through a post-workshop survey, which provided the RC team with valuable insights for refining future iterations of the workshop series. These lessons will inform the development of future workshops aimed at enhancing HPC skills across the UW research community. Additionally, the materials and methods developed for this series offer a template that can be adapted for other academic research computing centers, paving the way for broader implementation and outreach in STEM education.

**Figure 1: A photo from Workshop 3 of the series, "Using Gaussian for Computational Chemistry."**

## 2 OBJECTIVES

The objective of these workshops was to introduce data analysis tools and HPC, as well demonstrate how more advanced and scientific domain-specific software utilizes the job scheduling infrastructure and computing power delivered through HPC. The workshops offered were as follows, and the team developed specific objectives for each workshop (WS):

**WS 1** *Basic Python Programming*
- Understand the basics and importance of Jupyter Notebooks [12] towards reproducible and transparent scientific research.
- Navigate the basics of Python as a scripting and programming language.

**WS 2** *Using Hyak, Linux Operating System (OS)*
- Navigate a file system and explore files using Linux commands.
- Understand the purpose of a job scheduler like SLURM in HPC [28].

**WS 3** *Using Gaussian for Computational Chemistry*
- Create and modify input file(s) for quantum chemistry calculations, and become familiar with output files from Gaussian [7] and Quantum Espresso [9, 10].
- Submit and monitor computational chemistry jobs to an HPC cluster.

**WS 4** *Data Analysis in Python with scikit-learn and PyTorch*
- Apply linear and logistic regression models to data and interpret the results.
- Train a neural network in Python and understand training parameters.
- Tune network architecture parameters to improve performance and articulate the various metrics to evaluate a machine learning model.

Our final objective was to make the materials available for future iterations and independent study by the student participants.

## 3 METHODS

### 3.1 Format

Each workshop was three hours, live, hands-on, and delivered in Seattle, WA, USA on the main UW campus. Students were sent calendar invitations and reminder emails with links to training materials at least 24 hours in advance of each workshop. In addition, virtual office hours were held the day prior to each workshop to offer support, whether workshop related or beyond.

Workshops utilized a variety of tools to deliver content. WS 1 used Jupyter Lab and Notebooks via Hyak's Open OnDemand [14] environment delivered through each participant's browser. Additionally, each participant was guided through the installation of Anaconda Navigator for later independent study during their REU. WS 2 was delivered using participants' local shell terminals. All commands were demonstrated on Hyak, which runs Rocky Linux 8. Hyak is the UW tri-campus supercomputer with over 30K compute cores, 500 GPU accelerators, and 2.5 PB of parallel file systems storage operating on a shared condo model. The CEI has dedicated resources that are primarily used for research throughout the year and were temporarily reallocated for education and outreach during these workshops. WS 3 used a combination of submitting jobs and editing scripts in the Hyak shell, then generating inputs and viewing Gaussian and Quantum Espresso results with Hyak's OnDemand virtual desktop. WS 4 used Google Colab, demonstrating an additional platform for using Python-based computational notebooks in the cloud.

### 3.2 Training Materials

Training materials were prepared and made publicly accessible via a GitHub repository or as a web-based tutorial on Hyak's documentation website. The program's materials are reproducible, compatible with most academic HPC clusters, and in-line with FAIR computational workflow specifications [11]. The format is flexible and could be altered to focus on any HPC-compatible discipline-specific software.

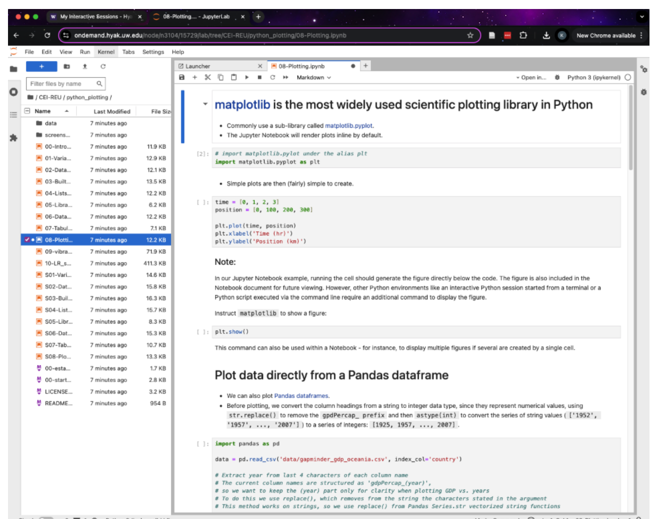



**Figure 2: Jupyter Lab and Notebooks prepared for WS 1 accessed via Hyak Open OnDemand. Python was taught via live demonstration with a blank notebook. The photo shows notebooks that were prepared from Software Carpentry's lesson, "Plotting and Programming in Python" for the participant's independent study.**

WS 1 sampled exercises from Software Carpentry's [26, 27] lesson, "Plotting and Programming in Python." Selected sections were made into Jupyter Notebooks with and without solutions for participant independent study, but the demonstration was live from a fresh Notebook without solutions. WS 1 was delivered via Hyak's OnDemand platform, which ran a containerized version of Jupyter's datascience-notebook. (Figure 2).

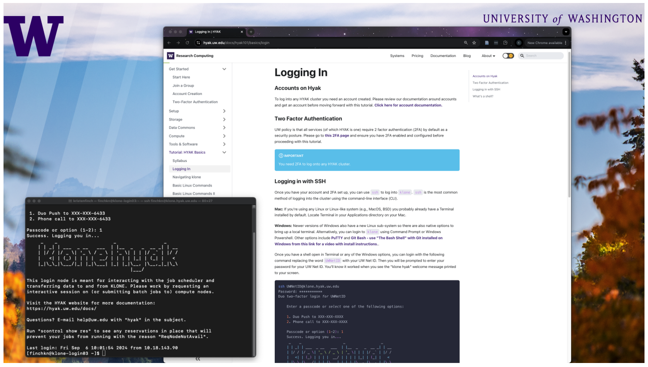



**Figure 3: Hyak's documentation website showing materials for WS 2 as well as a shell logged into Hyak's third generation cluster, klone.**

WS 2 introduced Hyak's file system (Figure 3), and some exercises to demonstrate Linux commands were sampled from Software Carpentry's [26, 27] lesson, "The Unix Shell." SLURM interactive jobs, batch jobs, and array jobs were demonstrated using Apptainer [16, 17] and a containerized version of Locator neural network [1] and publicly available data from black cottonwood trees [8].

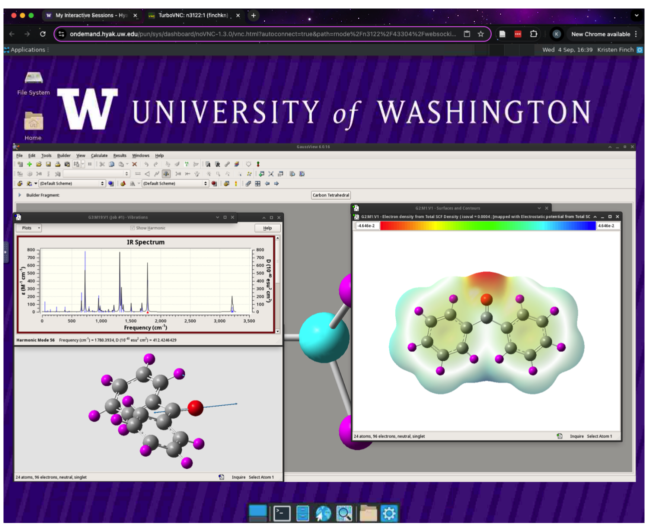



**Figure 4: GaussView accessed via Hyak Open OnDemand virtual desktop feature, TurboVNC, showing an example molecule used during WS 3 from the photochemCAD database (benzophenone). The upper left panel shows the calculated infrared absorption spectrum (and the "normal mode" responsible for the highlighted peak). The right panel shows the electrostatic potential and the electronegativity of the oxygen atom.**

WS 3 introduced quantum chemistry software on Hyak. The GaussView [4] graphical user interface was delivered via Hyak's OnDemand platform (Figure 4), and used to set up both Gaussian [7] and Quantum Espresso [9, 10] calculations. Job types included geometry optimizations and band plots were shown. In addition, the photochemCAD database [24] was used to compare experimental and calculated absorption spectra. Exercises for WS 4 were prepared as Jupyter Notebooks and delivered with Google Colab (Figure 5).

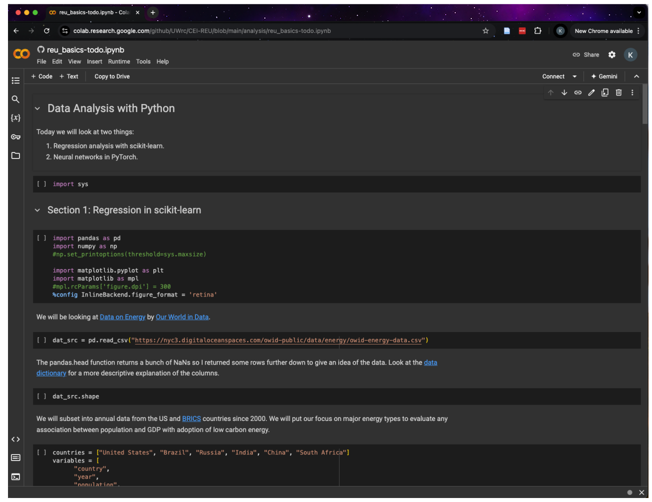



**Figure 5: Google Colab view of Jupyter Notebook prepared for WS 4 to demonstrate some data analysis basics as well as linear and logistic regression and training a neural network.**

WS 4 utilized a data set from Our World in Data on statistics regarding country-level clean energy adoption [23]. Trends were predicted using linear regression to predict future clean energy utilization and classification done using logistic regression with scikit-learn [21] to predict what factors would contribute towards a country adopting clean energy (or not). A shallow, three-layer artificial neural network (ANN) was generated with PyTorch [20] but the framework allows for the attendee to modulate ANN depth, layer width, and test optimization algorithms to experiment with different ANN architectures. All these aspects help to demonstrate the effort that goes into tuning an ANN to assess performance improvements relative to logistic regression. Model performance metrics were explored, including $R^2$ and Receiver Operating Characteristic (ROC) curves [6], as quantitative measures for model comparison.

## 3.3 Accessibility

With the exception of one student, all students brought a personal computer to the workshop and installed the required software to participate. One student was provided a laptop for use by UW Information Technology. No other accommodations were requested.

## 3.4 Evaluation and Analysis

All workshop objectives were translated into post-workshop survey statements to evaluate if the objective was met, and we refer this category of survey response as, "objective agreement ranking scores" because participants were asked to rank their level of agreement with the statements from 0 (strongly disagree) to 5 (strongly agree). In addition, the survey collected information about the participants educational experience, prior knowledge about workshop topics, and their assessment of the relevance of the topics towards their current and future studies. We also requested positive and negative feedback short responses.

Our analysis included data visualization and rudimentary statistical analysis of participant agreement rankings using R [22] via a Tidyverse container (V4.4.1) from the Rocker Project [2, 19]. For the purposes of this discussion, we evaluated a ranking ≥ 2.5 as indicating that an objective had been met.

## 4 RESULTS

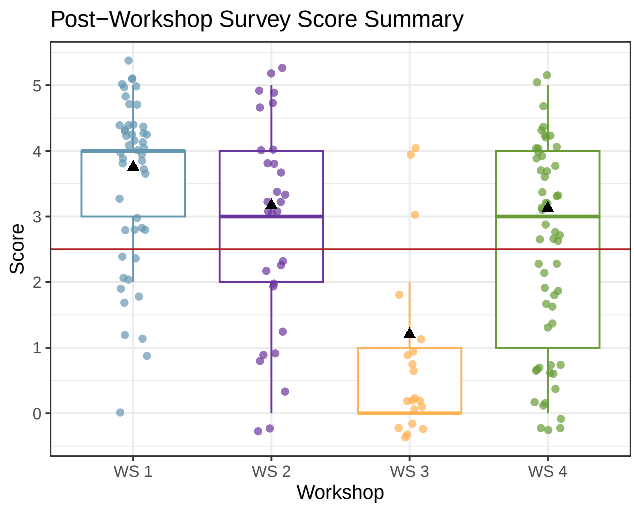



**Figure 6: Box plot shows the distribution of objective agreement ranking scores for each workshop (WS). Points indicate individual scores from post-workshop survey respondents, which were used to generate the box plot summaries. These points and the box plots were color coded by workshop. Black triangular points indicate the average overall score for each workshop on a scale from 0 (not helpful) to 5 (extremely helpful). The red line at 2.5 is our threshold for if workshop objectives were met. *NOTE: Objective agreement ranking scores were collected on a discrete scale. In this plot, points were "jittered" for better visualization of the data.***

Thirteen participants responded to the post-workshop survey. The prior education experience breakdown was 23.1% for having completed 0-1 years of post secondary education, 46.2% for 1-2 years, 15.4% for 2-3 years, and 15.4% for 3-4 years. The two workshops demonstrating Python (WS 1 and WS 4) were evaluated with the highest relevance for participants future research and professional development, scoring on average 3.50 and 3.25, respectively on a scale of 0 (not certain of the relevance) to 5 (high relevance or likely frequent and regular usage in the future).

The overall average objective agreement score was 2.66 ($SD$ = 1.64) indicating that some objectives were met yet suggesting some areas for improvement in future iterations. By workshop, the average objective agreement ranking scores were: 3.50 ($SD$ = 1.25) for WS 1, 2.83 ($SD$ = 1.62) for WS 2, 0.90 ($SD$ = 1.33) for WS 3, and 2.48 ($SD$ = 1.53) for WS 4. Participants' overall workshop rating on a scale from 0 (not helpful) to 5 (extremely helpful) was evaluated independent of the objective agreement rankings, and participants ranked each workshop higher than the agreement rankings scores suggest: 3.75 ($SD$ = 0.622) for WS 1, 3.17 ($SD$ = 1.17) for WS 2, 1.20 ($SD$ = 1.30) for WS 3, and 3.12 ($SD$ = 1.64) for WS 4. Additionally, the respondents agreed that the instructors were understanding of their needs (3.65 on average; $SD$ = 1.20), but short responses requesting critical feedback and improvements indicated that the pace at which the material was covered should be decreased for better learning and retention.

## 5 DISCUSSION

Although the average objective agreement ranking for the workshop series as a whole was leaning in agreement that our objectives were met, we have reflections to share along with improvements for future iterations:

(1) WS 1 was largely successful, but the wrong venue was selected. The classroom was too small and lacked climate control, especially important for a summer course. Being the first workshop, at least 20 participants attended and stayed for the entire three hour session. Subjectively, there were palpable feelings of excitement and energy in this first session. However, due to first time delivery issues and other logistical issues around the infrastructure, it is possible this momentum was blunted as the relatively high attendance did not carry over to subsequent workshops. Learning basic Python with Jupyter Notebooks is generally enjoyable, which was reflected by the positive feedback.

(2) WS 2 was in part successful, but it was clear that combining basic Linux skills with advanced concepts included in demonstrating SLURM was not effective for this group. Like with Python basics, Linux basics are fun and easy to follow, but the jump to SLURM was not well executed. This particular discussion of SLURM also utilized a software container, which exposed the students to additional, advanced content. We have since improved this training with simpler and easier to consume examples. For future iterations, WS 2 should be split into a session focused solely on Linux CLI and a separate session on SLURM and parallel computing to address learning fatigue.

(3) Feedback from participants and instructor observations also indicated that the pacing of WS 3 needed to be slowed and include fewer distinct examples, allowing more time to be spent on each part of the demonstration. One participant commented that while the workshop PDF included a step-by-step guide, it was missing some vital commands that would have helped participants catch up when they missed a step and consequently fell behind the presentation.

(4) Since WS 1 and WS 4 were both using Jupyter Notebooks, in future iterations, we will deliver these workshops in sequence to increase continuity with these tools. During WS 4, in-class engagement in discussion suggesting that the participants were following the course content, but the range of agreement ranking scores was high, indicating not all participants were reached.

We think that learning fatigue was present in all workshops due to the three hour format. An alternative pedagogical delivery suggests balancing our approach with more active self-directed learning [3, 25]. Outside of the workshop series, students were participating in a immersive research experience with new colleagues and independent projects, further adding stress on their time, energy, and focus. Hosting more frequent but shorter sessions could be more effective for this demographic as there has been a generational shift in attention spans and emphasis on brevity [18, 25].

## 6 FUTURE PLANS

It is worth exploring if a format that connects the workshops around a single dataset or product could improve participant engagement and the overall learning experience. For example, a single dataset or problem would allow any "output" generated from a prior workshop be the "input" to the next. Should a student attend every session, the material would make sense not only in terms of the presented computational skill but it would loosely resemble a journey an experienced researcher might take to explore a hypothesis. A final hackathon could bring together students to write an academic paper unifying the workshop materials and results, and provide a tangible outcome for the participants in addition to capacity building.

In addition, we aim to broaden the applicability of this content and extend this integrated training model beyond the REU event. By applying this approach to various science and engineering disciplines, we can provide a cohesive learning experience that supports students in building computational skills progressively. This template may also be beneficial in reaching fields that lack training advanced computing and data literacy but may benefit from technological advanced in computer vision and AI. Introducing these disciplines to a structured, data-driven learning process can address training gaps and inspire avenues for research. Our goal is to create a versatile training platform at University of Washington that fosters interdisciplinary collaboration and advances students' skills across a wide range of scientific and engineering domains.

## 7 CONCLUSIONS

Many universities have long had a research computing, high-performance computing, or similar teams for decades and only relatively recently adopted data science analogs [5, 13, 15]. As noted previously in this paper, the research computing team has intentions to refine and redeploy the material used in this REU workshop. The UW RC team is developing a regular training schedule for its research community, drawing from both accepted best-practices nationally as well as custom modules. We explored how packaging this combination of pre-existing and custom educational materials would be received by a diverse undergraduate research audience and provided the lessons and feedback for other teams to learn from. Given the sample size ($n = 25$), we acknowledge there could be limited external validity of any conclusions. However, we believe it is important to provide these voices toward the ongoing national discussions around computational skills training for research trainees in higher education.

## ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (NSF) through the UW Molecular Engineering Materials Center, a Materials Research Science and Engineering Center (DMR-2308979). Computations were facilitated through the use of advanced computational, storage, and networking infrastructure provided by the shared facility supported by the Clean Energy Institute (CEI) via the UW Hyak Supercomputer.

## REFERENCES

[1] CJ Battey, Peter L Ralph, and Andrew D Kern. 2020. Predicting geographic location from genetic variation with deep neural networks. 9 (2020), e54507. https://doi.org/10.7554/eLife.54507

[2] Carl Boettiger and Dirk Eddelbuettel. 2017. An Introduction to Rocker: Docker Containers for R. *The R Journal* 9, 2 (2017), 527–536. https://doi.org/10.32614/RJ-2017-065

[3] Katarzyna Czabanowska, Jos Moust, André Meijer, Peter Schröder-Bäck, and Herma Roebertsen. 2012. Problem-based Learning Revisited, introduction of Active and Self-directed Learning to reduce fatigue among students. 9, 1 (2012). https://doi.org/10.53761/1.9.1.6

[4] Roy Dennington, Todd A. Keith, and John M. Millam. 2019. GaussView Version 6. Semichem Inc. Shawnee Mission KS.

[5] Rae Earnshaw. 2019. *Data Science Institutes and Data Centers.* Springer International Publishing, 93–108. https://doi.org/10.1007/978-3-030-24367-8_7 Series Title: Advanced Information and Knowledge Processing.

[6] Tom Fawcett. 2006. An introduction to ROC analysis. 27, 8 (2006), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

[7] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman, and D. J. Fox. 2016. Gaussian~16 Revision C.01. Gaussian Inc. Wallingford CT.

[8] A. Geraldes, S. P. Difazio, G. T. Slavov, P. Ranjan, W. Muchero, J. Hannemann, L. E. Gunter, A. M. Wymore, C. J. Grassa, N. Farzaneh, I. Porth, A. D. Mckown, O. Skyba, E. Li, M. Fujita, J. Klapste, J. Martin, W. Schackwitz, C. Pennacchio, D. Rokhsar, M. C. Friedmann, G. O. Wasteneys, R. D. Guy, Y. A. El-Kassaby, S. D. Mansfield, Q. C. B. Cronk, J. Ehlting, C. J. Douglas, and G. A. Tuskan. 2013. A 34K SNP genotyping array for Populus trichocarpa: Design, application to the study of natural populations and transferability to other Populus species. 13, 2 (2013), 306–323. https://doi.org/10.1111/1755-0998.12056

[9] P Giannozzi, O Andreussi, T Brumme, O Bunau, M Buongiorno Nardelli, M Calandra, R Car, C Cavazzoni, D Ceresoli, M Cococcioni, N Colonna, I Carnimeo, A Dal Corso, S de Gironcoli, P Delugas, R A DiStasio Jr, A Ferretti, A Floris, G Fratesi, G Fugallo, R Gebauer, U Gerstmann, F Giustino, T Gorni, J Jia, M Kawamura, H-Y Ko, A Kokalj, E Küçükbenli, M Lazzeri, M Marsili, N Marzari, F Mauri, N L Nguyen, H-V Nguyen, A Otero de-la Roza, L Paulatto, S Poncé, D Rocca, R Sabatini, B Santra, M Schlipf, A P Seitsonen, A Smogunov, I Timrov, T Thonhauser, P Umari, N Vast, X Wu, and S Baroni. 2017. Advanced capabilities for materials modelling with QUANTUM ESPRESSO. *Journal of Physics: Condensed Matter* 29, 46 (2017), 465901. http://stacks.iop.org/0953-8984/29/i=46/a=465901

[10] Paolo Giannozzi, Stefano Baroni, Nicola Bonini, Matteo Calandra, Roberto Car, Carlo Cavazzoni, Davide Ceresoli, Guido L Chiarotti, Matteo Cococcioni, Ismaila Dabo, Andrea Dal Corso, Stefano de Gironcoli, Stefano Fabris, Guido Fratesi, Ralph Gebauer, Uwe Gerstmann, Christos Gougoussis, Anton Kokalj, Michele Lazzeri, Layla Martin-Samos, Nicola Marzari, Francesco Mauri, Riccardo Mazzarello, Stefano Paolini, Alfredo Pasquarello, Lorenzo Paulatto, Carlo Sbraccia, Sandro Scandolo, Gabriele Sclauzero, Ari P Seitsonen, Alexander Smogunov, Paolo Umari, and Renata M Wentzcovitch. 2009. QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter* 21, 39 (2009), 395502 (19pp). http://www.quantum-espresso.org

[11] Carole Goble, Sarah Cohen-Boulakia, Stian Soiland-Reyes, Daniel Garijo, Yolanda Gil, Michael R. Crusoe, Kristian Peters, and Daniel Schober. 2020. FAIR Computational Workflows. *Data Intelligence* 2, 1-2 (01 2020), 108–121. https://doi.org/10.1162/dint_a_00033 arXiv:https://direct.mit.edu/dint/article-pdf/2/1-2/108/1893377/dint_a_00033.pdf

[12] Brian E. Granger and Fernando Perez. 2021. Jupyter: Thinking and Storytelling With Code and Data. *Computing in Science & Engineering* 23, 2 (March 2021), 7–14. https://doi.org/10.1109/MCSE.2021.3059263
[13] Tony Hey and Anne Trefethen. 2003. e-Science and its implications. 361, 1809 (2003), 1809–1825. https://doi.org/10.1098/rsta.2003.1224
[14] Dave Hudak, Doug Johnson, Alan Chalker, Jeremy Nicklas, Eric Franz, Trey Dockendorf, and Brian McMichael. 2018. Open OnDemand: A web-based client portal for HPC centers. *Journal of Open Source Software* 3, 25 (May 2018), 622. https://doi.org/10.21105/joss.00622
[15] Nicholas W. Jankowski. 2007-01. Exploring e-Science: An Introduction. 12, 2 (01 2007-01), 549–562. https://doi.org/10.1111/j.1083-6101.2007.00337.x
[16] Gregory M. Kurtzer, cclerget, Michael Bauer, Ian Kaneshiro, David Trudgian, and David Godlove. 2021. *hpcng/singularity: Singularity 3.7.3.* https://doi.org/10.5281/zenodo.4667718
[17] Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. 2017. Singularity: Scientific containers for mobility of compute. 12, 5 (2017), e0177459. https://doi.org/10.1371/journal.pone.0177459 Publisher: Public Library of Science.
[18] Michael Z. Newman. 2010. New media, young audiences and discourses of attention: from *Sesame Street* to 'snack culture'. 32, 4 (2010), 581–596. https://doi.org/10.1177/0163443710367693
[19] Daniel Nüst, Dirk Eddelbuettel, Dom Bennett, Robrecht Cannoodt, Dav Clark, Gergely Daróczi, Mark Edmondson, Colin Fay, Ellis Hughes, Lars Kjeldgaard, Sean Lopp, Ben Marwick, Heather Nolis, Jacqueline Nolis, Hong Ooi, Karthik Ram, Noam Ross, Lori Shepherd, Péter Sólymos, Tyson Lee Swetnam, Nitesh Turaga, Charlotte Van Petegem, Jason Williams, Craig Willis, and Nan Xiao. 2020. The Rockerverse: Packages and Applications for Containerisation with R. *The R Journal* 12, 1 (2020), 437–461. https://doi.org/10.32614/RJ-2020-007
[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. *PyTorch: an imperative style, high-performance deep learning library.* Curran Associates Inc., Red Hook, NY, USA.
[21] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* 12 (2011), 2825–2830.
[22] R Core Team. 2024. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/
[23] Hannah Ritchie, Max Roser, and Pablo Rosado. 2020. Renewable Energy. *Our World in Data* (2020). https://ourworldindata.org/renewable-energy.
[24] Masahiko Taniguchi, Hai Du, and Jonathan S. Lindsey. 2018. PhotochemCAD 3: Diverse Modules for Photophysical Calculations with Multiple Spectral Databases. 94, 2 (2018), 277–289. https://doi.org/10.1111/php.12862 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/php.12862.
[25] Shirine Voller, Eddie Blass, and Vicki Culpin (eds.). 2011. *The Future of Learning.* Palgrave Macmillan UK. https://doi.org/10.1057/9780230306356
[26] G. Wilson. 2006. Software Carpentry: Getting Scientists to Write Better Code by Making Them More Productive. 8, 6 (2006), 66–69. https://doi.org/10.1109/MCSE.2006.122 Conference Name: Computing in Science & Engineering.
[27] Greg Wilson. 2025. Software Carpentry web site. Retrieved December 2010 from http://software-carpentry.org Main web site for Software Carpentry, replacing http://swc.scipy.org..
[28] Andy B. Yoo, Morris A. Jette, and Mark Grondona. 2003. SLURM: Simple Linux Utility for Resource Management. In *Job Scheduling Strategies for Parallel Processing*, Gerhard Goos, Juris Hartmanis, Jan Van Leeuwen, Dror Feitelson, Larry Rudolph, and Uwe Schwiegelshohn (Eds.). Vol. 2862. Springer Berlin Heidelberg, Berlin, Heidelberg, 44–60. https://doi.org/10.1007/10968987_3 Series Title: Lecture Notes in Computer Science.


# A ARTIFACT DESCRIPTION

This appendix includes artifact associated with **Computational Skills Training for Undergraduate Researchers in Molecular Engineering**.

The artifacts described there are training materials prepared for the workshop series described, and *do not include computational results.* To access the workshop materials, follow the links presented below. During the workshop series held at University of Washington, participants were given a sponsored UWNetID, which is required to use UW's research computing cluster Hyak. Although the workshop series focused on UW resources and infrastructure, the materials are transformable to similar systems.

*A.0.1 Artifact Check-list.*

- **Publicly available workshop series repository on GitHub:** https://github.com/UWrc/CEI-REU.git
  - python_plotting sub-directory - these materials accompany WS 1 "Basic Python Programming" held on June 20, 2024.
    - Exercises and explanations used to build the Jupyter notebooks were sampled from Software Carpentry's "Plotting and Programming in Python."
      - Notebooks starting 00-08 can be followed workbook style by running the code blocks to demonstrate presented concepts. This set of Notebooks excludes the solutions to the Exercises.
      - Notebooks starting S00-S08 are duplicates of the Notebooks starting **00-08**, but this set of Notebooks include the solutions to the Exercises.
      - Notebooks starting 09-10 provide plotting examples from Gaussian outputs.
  - using_hyak sub-directory - these materials accompany WS 2 "Using Hyak, Linux Operating System" (OS) held on June 25, 2024. The walk-through tutorial is available on Hyak's documentation website discussed below, but we also share relevant files used in the tutorial via the repository so that they could be more broadly available.
    - locator_NN_job.slurm and locator_NN_array.slurm - template single and array job submission scripts.
    - data sub-directory
      - potr_genotypes.txt - sample data (adapted from [8].
      - potr_m_pred(0-4).txt - sample data (adapted from [8])
  - Using_Quantum_Chemistry_Software sub-directory - these materials accompany WS 3 "Using Gaussian for Computational Chemistry" held on June 27, 2024.
    - Chem_Software_Hyak.pdf - walk through tutorial that demonstrates how to: (1) use the GaussView graphical user interface, (2) create and submit Gaussian jobs, and (3) create and submit Quantum Espresso jobs.
    - ethane.sh file - this file will allow slurm submission of the ethane.gjf file.
    - Pseudopotential .upf files - extra files used in the Quantum Espresso portion of the workshop.
  - analysis sub-directory - these material accompany WS 4 "Data Analysis Basics in Python with scikit-learn and PyTorch" held on July 1, 2024.
    - Complete Jupyter Notebook
    - Jupyter Notebook with "TODO" or in-session code boxes for demonstration.
- **Publicly available tutorial on UW RC's Hyak documentation website:**
  - https://hyak.uw.edu/docs/hyak101/basics/syllabus provides a walk through tutorial with the following sections: Syllabus, Logging In, Navigating Klone, Basic Linux Commands, Basic Linux Commands II.
  - https://hyak.uw.edu/docs/hyak101/basics/syllabus_slurm provides a walk through tutorial for SLURM, which has been updated with additional examples. The "Advanced SLURM" sections were what was originally presented in addition to Linux CLI Basics for WS 2.
- **Additional publicly available resources:**
  - WS 2 used a containerized version of Locator Neural Network [1]. Locator Neural Network is available on GitHub: https://github.com/kr-colab/locator.git
  - The Dockerfile for the containerized version of Locator Neural Network used during WS 2 is available on GitHub:

https://github.com/finchnSNPs/Docker_kr-colab_locator
  - The Docker container used during WS 2 to run Locator Neural Network is available on DockerHub: https://hub.docker.com/repository/docker/finchnsnps/locator/general
- **Software and Code licenses:**
  - Bash, SLURM, and Quantum Espresso are published under the terms of the GNU General Public License.
  - Gaussian16 and Gaussview 6.1.1 are Copyright (c) 1988-2017, Gaussian, Inc. All Rights Reserved, and were used under a license held by the Li Lab at University of Washington.
  - Jupyter is a Copyright (c) Project Jupyter Contributors, and distributed under the terms of the 3-clause BSD License (https://jupyter.org/governance/projectlicense.html).
  - Locator Neural Network [1] is a copyright 2019 of C. J. Battey and released under a Non-Profit Open Software License 3.0 (NPOSL-3.0); (https://github.com/kr-colab/locator/blob/master/LICENSE.txt).
  - matplotlib's license is based on the Python Software Foundation license (https://matplotlib.org/stable/project/license.html).
  - NumPy's is a Copyright (c) 2005-2024, NumPy Developers. The license is publicly available (https://numpy.org/doc/stable/license.html).
  - Python is under the Python Software Foundation license agreement (https://docs.python.org/3/license.html#psf-license).
  - Pytorch is a Copyright (c) 2016-present, Facebook Inc. (https://github.com/pytorch/pytorch/blob/main/LICENSE)
  - scikit-learn was used under Copyright (c) 2007-2024 The scikit-learn developers, and distributed under the terms of the 3-clause BSD License (https://github.com/scikit-learn/scikit-learn/blob/main/COPYING).
  - scipy was used under Copyright (c) 2001-2002 Enthought, Inc. 2003-2024, SciPy Developers. It can be used under a BSD 3-clause (https://github.com/scipy/scipy/blob/main/LICENSE.txt).
  - Some of the data and exercises used in WS 1 and WS 2 are under the Copyright of Software Carpentry and are made available under the Creative Commons Attribution license (CC BY 4.0).
- **Data licenses:**
  - Our adaptation of *Populus trichocarpa* genotype data and locations [8] are licensed under a CC0 1.0 Universal (CC0 1.0) Public Domain Dedication license. Original genotyping results available on DRYAD (https://doi.org/10.5061/dryad.1051d).
  - All visualizations, data, and code produced by Our World in Data are completely open access under the Creative Commons BY license.

# HPC Andragogy: Automating Batch Scheduler Feedback

Kyriakos Tsoukalas
Colgate University
ktsoukalas@colgate.edu

## ABSTRACT

This paper proposes a monitoring system that emails feedback to users about submitted jobs and has the capability to stop and resubmit jobs to a batch scheduler. The proposed system has been implemented for a small supercomputing environment with a mix of high-performance and high-throughput computing jobs. User feedback includes alerts for over- and under-utilization of CPU and physical memory. This paper also discusses how predefined system thresholds were chosen and proposes three algorithms. An algorithm for the proposed monitoring system and two algorithms for the prediction of CPU and physical memory utilization. The latter algorithms are based on users' input of the identification string (job ID) of a similar job that should have finished execution without errors. Lastly, a git repository is shared to make the code accessible for review.

## KEYWORDS

HPC, HTC, Andragogy, Batch Scheduler, Feedback, Automation

## 1 INTRODUCTION

Batch scheduling on systems that combine High-Performance Computing (HPC) and High-Throughput (HTC) Computing has many challenges to overcome. The batch scheduler has a number of differing goals based on the mix of HPC and HTC jobs. Furthermore, resource optimization involves stohastic processes such as the demand for resources, the dynamic availability of resources, and the variability of job processing duration. Resource fragmentation is a particular challenge in systems with both HPC and HTC loads [3]. Moreover, different computer clusters may have different operating systems and support different scientific software.

An important factor in resource fragmentation is users' requested resources for their jobs. This paper proposes a system for monitoring the utilization of resources in comparison to the resources allocated to each running job. The proposed monitoring system automates feedback via email when a running job is over- or under-utilizing its allocated resources. The purpose of the proposed system is twofold. Firstly, to stop jobs that grossly under-utilize allocated resources and attempt to automatically resubmit them requesting appropriate resources. Secondly, to alert users for jobs that over-utilize or marginally under-utilize resources, prompting them to review the statistics for each job with misallocated resources. The focus of resource allocation extends beyond processing individual jobs. Variability in CPU and physical memory utilization affects all jobs submitted to a batch scheduler. Furthermore, possibly for long periods of time, each job's maximum and minimum resource utilization may vary greatly from the average resource utilization. Ideally, jobs that have known preparatory steps, such as moving files or preprocessing data, should be executed in a chain of jobs as opposed to a single job. Each job in the chain with an appropriate allocation of resources. Thus, the Colgate Supercomputer emails users when a predefined discrepancy (threshold) between allocated (requested) and utilized resources of CPU and physical memory has been exceeded.

The feedback emails aim to let researchers know when their job submissions are over- and under-utilizing allocated (requested) CPU and physical memory resources. The andragogical aspect of the proposed monitoring system is an early familiarization of new users of a supercomputer to the important aspect of resource allocation. More importantly, familiarization with the fact that the Colgate Supercomputer allows over-utilization when less resources have been requested and allocated. The overall goal is that users receive frequent feedback in order to learn to be more conservative with their requests for resources. Having knowledge that their jobs will be allowed to utilize more resources when then are able to, while a batch scheduler will not be able to allocate any allocated but unused resources to newly submitted jobs.

## 2 PROPOSED MONITORING SYSTEM

### 2.1 Rationale

The proposed system has been implemented on the Colgate Supercomputer [1]. Contemplate the following supercomputing academic context. A supercomputer is used by faculty, staff, and students alike, both for research and educational activities. The same supercomputer processes both HPC and HTC jobs, which include interactive applications and web services. Users of the supercomputer submit jobs that are queued to be processed while processing duration varies from a few minutes to weeks.

In the previously aforementioned context, it is extremely important to decrease the discrepancy between allocated (requested) and utilized resources. Firstly, to increase the number of jobs processed daily, weekly, monthly, and yearly. Secondly, to decrease the number of jobs competing for the same resources. Therefore, it is important to inform the users of a supercomputer about the difference between over-allocating resources versus under-allocating resources.

Over-allocating resources results in inability to schedule more jobs because there are unused but allocated resources that could have been allocated to newly submitted jobs. Under-allocating resources results in the possibility that multiple jobs may experience slower processing due to competing for the same resources. However, a batch scheduler may be configured to not fill a node when


https://doi.org/10.22369/issn.2153-4136/16/1/11

CPU or memory utilization is higher that allocated, unless there is no other node left with enough available (unallocated) resources. Moreover, jobs could be allowed to utilize more than the allocated resources when there are more available.

## 2.2 Proposed Algorithms

A git repository is shared to make the code accessible for review [10]. Algorithms 1 and 2 are used to predict utilization of CPU cores and physical memory, respectively, given the identification string (job ID) of a job with similar characteristics that has finished without errors. Many users tend to submit groups of jobs at a time, which have similar characteristics, such as the same processing script but with different parameters. Algorithm 3 describes the proposed monitoring system. The proposed monitoring system could run as a CRON job (a time-based job scheduler in Unix-like systems).

**Algorithm 1** Average Used CPU cores

**Input:** Job IDs, status, owner, and used CPU percentage
**Output:** Average Used CPU cores
$C_{max} \leftarrow 3$ ▷ Max counter
$I_{max} \leftarrow 11$ ▷ Max iterations
$I \leftarrow 0$ ▷ Iterations
$C \leftarrow 0$ ▷ Counter
ID ← (Job ID) ▷ Job ID
$T_{cpu} \leftarrow 0$ ▷ Total used CPU percentage
$J_{cpu} \leftarrow 0$ ▷ Used CPU percentage
**while** $C < C_{max}$ **and** $I < I_{max}$ **do**
    **if** job with ID has status = F **and** owner = user **then**
        $T_{cpu} \leftarrow T_{cpu} + J_{cpu}$
        $I \leftarrow I + 1$
    **else**
        $C \leftarrow C + 1$
    ID ← ID − 1
**Print:** round $\left(\frac{T_{cpu}}{100I}\right)$

**Algorithm 2** Average Used Physical Memory

**Input:** Job IDs, status, owner, and used physical memory
**Output:** Average Used Physical Memory
$C_{max} \leftarrow 3$ ▷ Max counter
$I_{max} \leftarrow 11$ ▷ Max iterations
$I \leftarrow 0$ ▷ Iterations
$C \leftarrow 0$ ▷ Counter
ID ← (Job ID) ▷ Job ID
$T_{mem} \leftarrow 0$ ▷ Total used physical memory in bytes
$J_{mem} \leftarrow 0$ ▷ Used physical memory in bytes
**procedure** ConvertMemory($m$)
    **if** $m >$ 1**gb then**
        **Print:** $m_{gb}$gb
    **else if** $m >$ 1**mb then**
        **Print:** $m_{mb}$mb
    **else if** $m >$ 1**kb then**
        **Print:** $m_{kb}$kb
    **else**
        **Print:** $m$
**while** $C < C_{max}$ **and** $I < I_{max}$ **do**
    **if** job with ID has status = F **and** owner = user **then**
        $T_{mem} \leftarrow T_{mem} + J_{mem}$
        $I \leftarrow I + 1$
    **else**
        $C \leftarrow C + 1$
    ID ← ID − 1
ConvertMemory($T_{mem}$)

**Algorithm 3** Monitoring System

**Input:** Running jobs info ($\mathcal{F}$), logged job IDs ($\mathcal{L}$), job status, owner, used walltime, used CPU percentage, and used physical memory
**Output:** Action log
$c \leftarrow 2$ ▷ Threshold for CPU core difference
$m \leftarrow 8$ GB ▷ Threshold for physical memory difference
**procedure** AllRunningJobs($\mathcal{F}$)
    **for each** Job `Job_With_ID` $\in \mathcal{F}$ **do**
        Process `Job Log` for ID
        **if** CPU cores requested - used > 2 **then**
            Increase counter in `Job Log`
            **if** counter > 1 **then**
                Stop `Job_With_ID` and email `Owner`
                Submit new job with `Used_CPU_Cores`
**procedure** AllLoggedJobs($\mathcal{L}$)
    **for each** Job `Job_With_ID` $\in \mathcal{L}$ **do**
        **if** 0 < CPU cores requested - used < $c$ **then**
            Email `Owner`
        **if** CPU cores requested - used < 0 **then**
            Email `Owner`
        **if** 0 < Physical memory requested - used > $m$ **then**
            Email `Owner`
        **if** Physical memory requested - used < 0 **then**
            Email `Owner`
AllRunningJobs($\mathcal{F}$)
AllLoggedJobs($\mathcal{L}$)

# 3 EMAIL FEEDBACK AUTOMATION

The Colgate Supercomputer automates feedback by monitoring all running jobs to alert users about under- or over- utilization of resources. Note that calculations should take into consideration how memory is reported (for example in MiB or MB). The monitoring system does not assign weights based on elapsed processing time of jobs, because the andragogical aspect of the proposed system aims to familiarize users with the implications of over-allocating resources. Email is a common medium for sending textual feedback.

The automation requires the capability to send emails. A common program for sending emails in Linux operating systems is mailx.

Figure 1 shows a basic Grafana dashboard for CPU utilization and Figure 2 shows a basic Grafana dashboard for memory utilization [5].

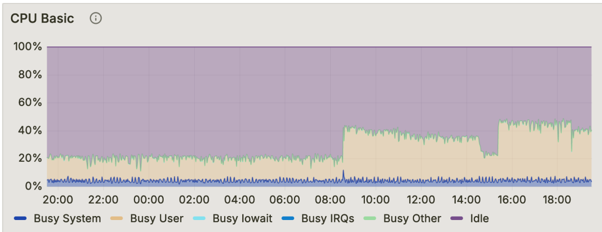



**Figure 1: Example Grafana Dashboard for CPU Utilization**

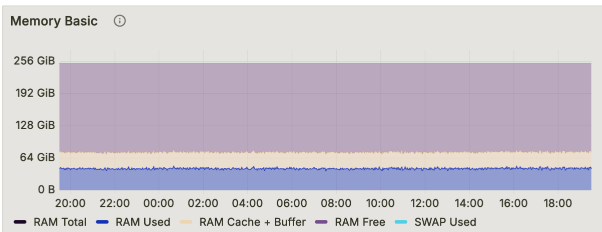



**Figure 2: Example Grafana Dashboard for Memory Utilization**

## 3.1 Email Feedback Templates

*3.1.1 Example for Major CPU Under-utilization.* The email subject is "CPU under-utilization Alert: Job xxxxx.<clustername> on Node n03 was stopped".

Job ID: xxxxx.<clustername>
Node: n03
Owner: <username>
Elapsed Time: 0h45m
Requested CPU cores: 31
Unused CPU cores: 4
Suggested CPU cores: 27
Metadata: qstat -xf xxxxx.<clustername>
Monitor: Monitor: https://monitor.domain.edu

This is an automated message.

Job xxxxx.<clustername> on Node n03 requested 31 CPU cores but was using an average of 2723% CPU after 45 minutes. Job xxxxx.<clustername> was stopped but resubmitted requesting 27 CPU cores. The new job's ID is yyyyy.<clustername>. It is suggested to request 27 CPU cores for future similar jobs instead of 31. Please review both jobs.

*3.1.2 Example for Minor CPU Under-utilization.* The email subject is "CPU under-utilization Alert: xxxxx.<clustername> on Node n03".

Job ID: xxxxx.<clustername>
Node: n03
Owner: <username>
Elapsed Time: 1d4h24m
Requested CPU cores: 31
Unused CPU cores: 2
Suggested CPU cores: 29
Metadata: qstat -xf xxxxx.<clustername>
Prediction of CPU cores: calccpu xxxxx.<clustername>
Monitor: https://monitor.domain.edu

This is an automated message.

Job xxxxx.<clustername> on Node n03 used 2903% CPU, therefore it is suggested to request 29 CPU cores for future similar jobs instead of 31. Please review your job.

*3.1.3 Example for non Multi-threading.* The email subject is "CPU under-utilization Alert: xxxxx.<clustername> on Node n03".

Job ID: xxxxx.<clustername>
Node: n03
Owner: <username>
Elapsed Time: 5h27m
Requested CPU cores: 3
Unused CPU cores: 2
Suggested CPU cores: 1
Metadata: qstat -xf xxxxx.<clustername>
Monitor: https://monitor.domain.edu

This is an automated message.

Job xxxxx.<clustername> on Node n03 used 36% CPU and did not multi-thread. When a job is not multi-threading it cannot use more that 1 CPU core and you will be notified when a job used more than the requested CPU cores. Therefore it is important to request only 1 CPU core for future similar jobs. Please review your job.

*3.1.4 Example for CPU Over-utilization.* The email subject is "CPU over-utilization Alert: xxxxx.<clustername> on Node n03".

Job ID: xxxxx.<clustername>
Node: n03
Owner: <username>
Elapsed Time: 13h13m
Requested CPU cores: 10
Unused CPU cores: -7
Suggested CPU cores: 17
Metadata: qstat -xf xxxxx.<clustername>
Prediction of CPU cores: calccpu xxxxx.<clustername>
Monitor: https://monitor.domain.edu

This is an automated message.

Job xxxxx.<clustername> on Node n03 used an average of 1707% CPU, therefore it is suggested to request 17 CPU cores for future similar jobs instead of 10. Please review your job.

*3.1.5 Example for Memory Under-utilization.* The email subject is "Memory under-utilization Alert: xxxxx.<clustername> on Node n03".

Job ID: xxxxx.<clustername>
Node: n03
Owner: <username>
Elapsed Time: 13h37m
Requested physical memory: 83886080kb
Unused physical memory: 50gb
Suggested physical memory: 35gb
Prediction of physical memory: calcmem xxxxx.<clustername>
Metadata: qstat -xf xxxxx.<clustername>
Monitor: https://monitor.domain.edu

This is an automated message.

Job xxxxx.<clustername> on Node n03 used 29gb of physical memory and 60gb of virtual memory, therefore it is suggested to request 35gb of physical memory for future similar jobs instead of 83886080kb. Please review your job.

*3.1.6 Example for Memory Over-utilization.* The email subject is "Memory over-utilization Alert: xxxxx.clustername on Node n03".

Job ID: xxxxx.<clustername>
Node: n03
Owner: <username>
Elapsed Time: 26h13m
Requested physical memory: 30gb
Unused physical memory: -21gb
Suggested physical memory: 55gb
Prediction of physical memory: calcmem xxxxx.<clustername>
Metadata: qstat -xf xxxxx.<clustername>
Monitor: https://monitor.domain.edu

This is an automated message.

Job xxxxx.<clustername> on Node n03 used 51gb of physical memory and 83gb of virtual memory, therefore it is suggested to request 55gb of physical memory for future similar jobs instead of 30gb. Please review your job.

## 4 PREDEFINED SYSTEM THRESHOLDS

On the Colgate supercomputer, jobs utilizing 1 to 3 CPU cores are considered HTC. There is a grey zone of CPU utilization between 4 and 7 CPU cores. A job which requires more that 12 hours of processing and an average of 4 CPU cores or more is considered HPC. Utilization of 8 or more CPU cores is also considered HPC. The goal of achieving a fair resource allocation between HPC and HTC job loads is a complicated matter.

The threshold for stopping a running job has been set to 3 cores. During job processing, jobs are not allowed to use, on average, less than 2 of the total CPU cores allocated. The proposed monitoring system will stop such jobs and try to resubmit them using the average CPU utilization within 25-45 minutes of job processing.

The proposed monitoring system will email alerts to users for jobs that use more or less than 8GB of physical memory than the memory requested and thus allocated by the batch scheduler.

## 5 EARLY RESULTS

The proposed monitoring system has been employed for about a year on the Colgate Supercomputer. The early findings from its use informed the supercomputer's documentation and a progressive refinement of the system during its trial period. The important findings from the trial period are presented in this section.

New users who submitted jobs requesting many CPU cores but their jobs did not multi-thread, received feedback emails. As a result, their subsequent submissions quickly changed to requesting just 1 CPU core, but also 2 to testing multi-threading. Testing multi-threading is not necessary as the proposed system will notify users when a job of theirs used more than the requested CPU cores. The relevant email alert was modified to clarify this capability in order to reduce unused but allocated CPU cores.

Some experienced users reached out to discuss the metric of average CPU utilization with regard to the maximum utilization during job execution. A result of these discussions was to update the documentation of the Colgate Supercomputer to emphasize the difference and how to interpret the relevant metadata given by the batch scheduler per job.

Another discussion with experienced users was about the differentiation between physical and virtual memory. Many users request physical memory that the batch scheduler allocates, even though their jobs end up using only a fraction of the allocation. Although the availability of large amounts of physical memory may help alleviate this problem, it remains an important issue in HPC andragogy. The relevant email feedback template was edited to clearly state the use of both physical and virtual memory.

The capability of the proposed monitoring system to stop and attempt to resubmit jobs that are under-utilizing allocated resources has been received with some skepticism from older users. Nevertheless, the system was able to attract attention to the important difference between under-allocating and over-allocating resources, and to the fact that jobs are allowed to use more resources than what was allocated by the batch scheduler.

Researcher training for supercomputing environments is not an easy endeavor. In a given academic institution, there can be a lot of different computing backgrounds as well as levels of experience with scientific software. The proposed monitoring system provides feedback as needed, rather than requiring users to undergo software carpentry and batch scheduler training to become familiar with a supercomputing environment, especially considering that many users either do not read the available documentation or only skim through it.

The most important result was that while new users were becoming familiar with the Colgate Supercomputer, any submitted jobs were restricted from excessively over-allocating resources.

## 6 DISCUSSION

In an effort to separate HPC and HTC jobs, the Colgate Supercomputer runs interactive applications on dedicated batch scheduling queues. Most interactive applications will run with a predefined number of CPU cores and physical memory. Achieving some level of separation between HTC and HPC job loads will reduce complexity and may result in fewer unused allocated resources while keeping higher job cancellation thresholds.

The thresholds chosen for the trial period, 3 CPU cores and 8GB of physical memory, would ideally be lowered in the future, as the resubmission process becomes more efficient. A common problem is that users could delete or rename files used in a job after its submission.

Formal training [4], informal training [7, 8], and educational activities [6] are based on proven paradigms, however they lack granularity due to the fact that learning is grouped in training modules, learning topics, or instructional activities. Consistent feedback for experiential learning provides increased granularity by matching one to one the job submissions of users of HPC/HTC systems.

A useful addition to the proposed monitoring system would be to automate a monthly report to each user that would present statistics about under- and over- utilization of requested resources. Supercomputing users would benefit from automatically receiving reports that highlight steps to improve subsequent job submissions, regarding resource requests.

Batch schedulers, such as Slurm [9], provide the statistics to be leveraged by the proposed monitoring system, hence shell scripting was used to implement the proposed monitoring system for the trial period. The approach is considering small supercomputers with less than 500 nodes. Newer resource managers have been proposed for supercomputers with more than 10k nodes [2]. A future integration with a large language model (LLM) would benefit from a python implementation.

The proposed monitoring system could assign predefined labels to each discrepancy between allocated (requested) and utilized resources (such as CPU cores and physical memory). An unsupervised machine learning algorithm, such as Bayesian learning, could then be employed to create user profiles regarding resource requests.

## 7 CONCLUSION

The proposed monitoring system offers a solution for managing resource misallocation leveraging the fault tolerance in scenarios of resource over-utilization. Supercomputing users can learn to be more conservative with their resource requests when there is little to no penalty for over-utilizing resources allocated by a batch scheduler.

Additionally, the proposed system is designed to efficiently reclaim resources that are allocated but under-utilized, particularly in an environment that integrates both HPC and HTC workloads. By addressing both over-utilization and under-utilization, the proposed system ensures more effective resource management, and promotes HPC andragogy.

Current work includes a refactoring of the system's code to query metadata from both PBS Pro and the Slurm batch schedulers. It also includes the development of Graphical Processing Unit (GPU) resource monitoring.

Future work includes integrating the monitoring system with Colgate's Supercomputer's AI chatbot which assists with documentation and coding. This integration aims to develop pipelines for more complex resource utilization predictions by analyzing previous job scripts as well as the scripts to be submitted. Parsing new scripts will also enable raising alerts about job dependencies and required POSIX permissions.

## REFERENCES


[1] Colgate University. 2024. *Colgate Supercomputer (Partially supported by NSF grant OAC-2346664).* https://rcd.colgate.edu Accessed: 2024-08-12.

[2] Yiqin Dai, Yong Dong, Kai Lu, Ruibo Wang, Wei Zhang, Juan Chen, Mingtian Shao, and Zheng Wang. 2022. Towards scalable resource management for supercomputers. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis* (Dallas, Texas) *(SC '22)*. IEEE Press, Article 24, 15 pages.

[3] Oliver Freyermuth, Peter Wienemann, Philip Bechtle, and Klaus Desch. 2021. Operating an HPC/HTC cluster with fully containerized jobs using HTCondor, Singularity, CephFS and CVMFS. *Computing and Software for Big Science* 5, 1 (2021), 9.

[4] Kai Himstedt, Nathanael Hubbe, Julian Kunkel, Hinnerk Stuben, Deutsches Klimarechenzentrum, Thomas Ludwig, Stephan Olbrich, Matthias Riebisch, Sandra Schröder, and Markus Stammberger. 2018. An HPC certification program proposal meeting HPC users' varied backgrounds. *Concept paper https://wr. informatik. uni-hamburg. de/research/projects/pecoh/start, DKRZ and RRZ* (2018).

[5] janakverma. 2024. *Linux Exporter Node.* https://grafana.com/grafana/dashboards/14513-linux-exporter-node/ Accessed: 2024-09-09.

[6] Bryan Johnston, Lara Timm, and Mabatho Hashatsi. 2023. Delivering Digital Skills across the Digital Divide. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis (SC23).*

[7] Julia Mullen, Chansup Byun, Vijay Gadepally, Siddharth Samsi, Albert Reuther, and Jeremy Kepner. 2017. Learning by doing, High Performance Computing education in the MOOC era. *J. Parallel and Distrib. Comput.* 105 (2017), 105–115. https://doi.org/10.1016/j.jpdc.2017.01.015 Keeping up with Technology: Teaching Parallel, Distributed and High-Performance Computing.

[8] Julia Mullen and Lauren Milechin. 2023. A Data Driven Approach to Informal HPC Training Evaluation. In *Practice and Experience in Advanced Research Computing.* 378–381.

[9] Nikolay A. Simakov, Robert L. DeLeon, Martins D. Innus, Matthew D. Jones, Joseph P. White, Steven M. Gallo, Abani K. Patra, and Thomas R. Furlani. 2018. Slurm Simulator: Improving Slurm Scheduler Performance on Large HPC systems by Utilization of Multiple Controllers and Node Sharing. In *Proceedings of the Practice and Experience on Advanced Research Computing: Seamless Creativity* (Pittsburgh, PA, USA) *(PEARC '18)*. Association for Computing Machinery, New York, NY, USA, Article 25, 8 pages. https://doi.org/10.1145/3219104.3219111

[10] Kyriakos Tsoukalas. 2024. *Supercomputer Monitoring System.* https://github.com/tsoukalask/jobcheck Accessed: 2024-09-09.

March 2025

Volume 16 Issue 1

ISSN 2153-4136 (online)