March 2024

Volume 15 Issue 1

JOCSE

Journal Of Computational Science Education

Promoting the Use of Computational Science Through Education

ISSN 2153-4136 (online)

Volume 15, Issue 1 **March 2024**

Journal Of Computational Science Education

The Journal Of Computational Science Education (JOCSE), ISSN 2153-4136, published in online form, is a supported publication of the Shodor Education Foundation Incorporated. Materials accepted by JOCSE will be hosted on the JOCSE website and will be catalogued by the Computational Science Education Reference Desk (CSERD) for inclusion in the National Science Digital Library (NSDL).

Subscription: JOCSE is a freely available online peer-reviewed publication which can be accessed at `http://jocse.org`.

Contents

Volume 15, Issue 1

Introduction to Volume 15, Issue 1

David Joiner, Editor
Kean University
djoiner@kean.edu

Volume 15, Issue 1 of the Journal of Computational Science Education offers a comprehensive overview of current trends and innovations in High-Performance Computing (HPC) and cybersecurity education. This issue emphasizes the importance of innovative educational strategies, the shift toward online and hybrid learning models, the role of community building in creating inclusive educational environments, the necessity of aligning educational content with practical application, and the critical need for sustainability and evolution in educational programs. Through these themes, the issue reflects a broader commitment to developing a diverse, skilled workforce capable of navigating the challenges and opportunities of these rapidly evolving fields, underscoring the significance of adaptability, inclusivity, and real-world relevance in shaping the future of HPC and cybersecurity education.

This issue includes articles from the tenth annual Best Practices in High Performance Computing Training and Education meeting at SC23, one article from the ninth annual BPHTE meeting, and a submitted student paper.

Purwanto et al. describe the DeepSecure training program to bolster cybersecurity. Multiple authors discuss efforts to manage hardware and architecture access for resource-constrained learners, including Johnston et al.'s description of using OpenHPC based virtual systems, Raoofy et al.'s paper on the BEAST Lab course on modern architecture and accelerators, and Seegerer and Nakahara's paper on approaches to teaching quantum computing.

Other papers focus on the challenge of scaling the delivery of training resources to a wide audience, including Vivas et al.'s paper on HPC Education in Colombia's summer schools, Reid et al.'s presentation of the HPC Carpentry program, Mehringer et al.'s description of the HPC Ed pilot project to share and federate repositories of training objects, and Crosby et al. detail a Cross-Institutional Research Engagement Network to train facilitators to expand the outreach of research programs at the University of Tennessee Knoxville and Arizona State University. Filinger and Cohen describe results from the "Understanding the Skills and Pathways Behind Research Software Training" BoF at ISC'23. Lastly, multiple papers focus on examples of student projects and training programs, including a student paper by Yazdani et al. on using neural networks for materials science, Adeniji et al.'s paper on multiple projects using Unity for Virtual Reality-based science visualization, Bautista and Sukhija's paper on a new Data Science certificate program at the National Energy Research Scientific Computing Center, and Leung et al. describe the HPC Bootcamp program with the Department of Energy.

We thank all of our contributors, editors, reviewers, and readers and look forward to hearing more from you in future issues!

Sincerely,
Dave Joiner

DeapSECURE Computational Training for Cybersecurity: Progress Toward Widespread Community Adoption

Wirawan Purwanto
Old Dominion University
wpurwant@odu.edu

Bahador Dodge
Old Dominion University
bdodg001@odu.edu

Karina Arcaute
Old Dominion University
karcaute@odu.edu

Masha Sosonkina
Old Dominion University
msosonki@odu.edu

Hongyi Wu
The University of Arizona
mhwu@arizona.edu

ABSTRACT

The Data-Enabled Advanced Computational Training Program for Cybersecurity Research and Education (DeapSECURE) is a non-degree training consisting of six modules covering a broad range of cyberinfrastructure techniques, including high performance computing, big data, machine learning and advanced cryptography, aimed at reducing the gap between current cybersecurity curricula and requirements needed for advanced research and industrial projects. Since 2020, these lesson modules have been updated and retooled to suit fully-online delivery. Hands-on activities were reformatted to accommodate self-paced learning. In this paper, we summarize the four years of the project comparing in-person and online only instruction methods as well as outlining lessons learned. The module content and hands-on materials are being released as open-source educational resources. We also indicate our future direction to scale up and increase adoption of the DeapSECURE training program to benefit cybersecurity research everywhere.

KEYWORDS

Parallel computing, big data, machine learning, cybersecurity, non-degree training, hands-on, online training

1 INTRODUCTION

The world that we live in today relies heavily on connected computers and mobile devices. Furthermore, many physical instruments are now connected to form the "internet-of-things". As such, the significance of cybersecurity cannot be underestimated. Cybersecurity in practice consists of many different tools, techniques and policies to protect and defend computing systems from potential attacks, as well as detect and mitigate attempted attacks. The research and development of novel cybersecurity tools and techniques have become more dependent on advanced cyberinfrastructure (CI) due to increasing complexity of the cyber systems being defended, as well as the growing intensity and sophistication of cyberattacks. As an area of study, cybersecurity is a multi-disciplinary field which draws from areas such as computer science and engineering, information technology, mathematics, business, law, social science, psychology, and more. At present, however, standard curricula used in many colleges and universities lack inclusion of advanced CI techniques to strengthen cybersecurity analysis, research, and development. This lack exists only in cybersecurity as a stand-alone discipline, but also in many of its "upstream" disciplines mentioned earlier. As a result, skill and knowledge gaps exist among students who are being trained to work in research areas related to cybersecurity.

With funding from the National Science Foundation (NSF), the School of Cybersecurity at Old Dominion University (ODU) developed DeapSECURE (short for *Data-Enabled Advanced Computational Training Platform for Cybersecurity Research and Education*) as an innovative, non-degree CI training program tailored for cybersecurity students and researchers. The DeapSECURE training program was created to address major curricular gaps in cybersecurity education in the areas of advanced computing. This non-degree training program consists of six modules that cover a broad range of CI topics: high performance computing (HPC), big data analytics (BD), neural networks (NN), machine learning (ML), parallel programming (PAR) and cryptography for privacy-preserving computation (CRYPT) [16]. These techniques are used extensively in state-of-the-art cybersecurity research and practice.

The goals, approach and philosophy of the DeapSECURE training program has been described in detail our earlier paper [18]. DeapSECURE emphasizes hands-on experience to fortify and connect theoretical materials with real-world applications. The primary goal of DeapSECURE lessons is to "crack open" the tough nuts of CI methods and concepts through practical use cases and codes. Application examples in the modules are carefully selected to engage learners in the field of cybersecurity and aim to train current and future researchers, engineers and practitioners with advanced techniques and skills necessary to carry out cybersecurity research and industrial projects. In a way similar to that adopted by the Carpentries [3], we leverage real cybersecurity problems (i.e. scenarios) and datasets as a way to introduce and practically learn the CI techniques through workshops. The CI technique unfolds as the lesson progresses through a series of computer codes employed to work out the solution. Quite frequently, important concepts are directly demonstrated to learners by these codes, followed by the explanation on the spot. All the lesson materials are available openly on

https://doi.org/10.22369/issn.2153-4136/15/1/1

the DeapSECURE website [16]. Workshops based on DeapSECURE lesson materials are not meant to replace comprehensive educational means such as semester-long courses; neither are the lessons intended to serve as a complete overview or an in-depth treatise on the CI topics. Rather, they are meant to give an initial practical exposure to CI and to provide learners with the first "stepping stones" to their further learning of CI for their own purposes (e.g. research). Exercises and activities in the lessons encourage learners to try, explore, and experiment with the CI tools. This training program has been in continuous development since 2018. By 2022, the lesson modules have been improved and road-tested through at least four workshop iterations.

In this paper, we present the complete conversion of all the DeapSECURE lessons from the in-person format to fully virtual delivery: the changes implemented to adapt the lessons for virtual workshops, the experience of conducting the workshops online, and the learners' feedback and reaction to the online format. We will also describe our effort of training students to become workshop teaching assistants (WTAs) and content developers, which we consider to be an important next-step to sustain the training program beyond NSF funding. The rest of the paper is organized as follows: Section 2 describes the first two years, when training was conducted in-person only. Section 3 outlines training adaptation to the online delivery and introduces our approach to training WTAs to assist the development process. In Section 4, we note on the statistics of the learners and their perception of the transition of DeapSECURE from in-person to online training. Finally, we discuss the availability of open-source training modules (Section 5) and a future roadmap in Section 6.

2 FIRST- AND SECOND-YEAR DEVELOPMENT

First year (Y1, 2018–2019 academic year)—The lesson modules of DeapSECURE were developed from scratch in the first year of the program. The unique component of DeapSECURE—combining exposure of state-of-the-art research and hands-on training on CI techniques—was developed through intensive engagement with cybersecurity researchers at ODU [18]. The hands-on component was designed with the use of HPC (i.e., parallel computers) in mind, since HPC will allow students to eventually scale up their computation when working with many challenging, real-world cybersecurity research problems. The training materials were developed collaboratively by the project's principal investigators (PIs) and WTAs using Gitlab for codes, lessons and data repositories, as well as Google Drive for document sharing and workflow coordination among team members. Three Ph.D. students assisted in the development of the lessons and the hands-on parts of the workshops. Assessments had been an integral part of the training program since its inception, utilizing pre- and post-workshop surveys, as well as focus group interviews. Findings from assessments helped drive continuous improvement of the program.

The six modules were offered twice in Y1, first as a series of workshops during the 2018-2019 academic year, and second as a week-long summer institute in June 2019. (These were in-person events, but the entire sessions were recorded with the support of ODU Distance Learning for learners' review and/or future purpose of creating a repository of video learning resources.) Each workshop lasted for three hours, which included a 30-minute cybersecurity research presentation by ODU faculty members. The bulk of the workshop consisted of hands-on introduction of CI methods using the participatory live coding method as adopted by the Carpentries [12], where the instructor narrated the method and typed on his/her own computer screen, and learners were to follow the same steps on their own computers, following the instructor's projected screen. The hands-on activities of the workshops were at that time carried on ODU's Turing HPC cluster, primarily on the UNIX terminal interface. The first-year program, the contents of the workshop materials, the demographic of the learners, and the initial assessment results were described in detail in Ref. [18].

The training program was widely advertised to ODU student body, particularly to cyber-related fields (cybersecurity, electrical and computer engineering, computer science, and modeling & simulation study programs). There were close to 50 sign-ups received; they were all accepted to the program. During the academic year, student attendance varied greatly through the semester (between 11 and over 30) based on their course workloads. Participation in summer institution was more consistent (17–21), presumably due to the absence of other commitments and contiguous workshop days. The workshops were generally well received, and students were exposed to state-of-the-art cybersecurity research topics and modern CI methods, both of which were not in the students' general awareness prior to this training. There were notable challenges in the hands-on sessions, however, due to the diversity of the participants' backgrounds as well as their computer programming experiences [17]. In particular, the command-line interface posed difficulty for many learners, who had not been familiar with such a mode of interaction with computers.

Second year (Y2, 2019–2020)—Key changes were introduced to the lessons and the workshop delivery [17], taking the lessons learned from the first year's workshop experience. Firstly, the modules were grouped into two distinct groups: (1) compute-intensive modules (HPC, CRYPT, PAR); (2) data-intensive modules (BD, ML, NN). The data-intensive modules were completely rewritten to use Pandas [13] as the data analytics toolkit (in Y1, the BD module used PySpark, which is a more difficult framework to use), together with scikit-learn [14] and Keras [4]. In an effort to streamline the lessons, a single cybersecurity use case was used for the three lesson modules, leveraging the SherLock smartphone security dataset [11]. This resulted in a more focused attention to three cybersecurity themes as the backdrop to introduce the CI techniques in the lessons: (1) spam email analysis; (2) computation with homomorphically encrypted data; (3) mobile device security. Additional hands-on sessions named "hackshops" were introduced (one session for every workshop) to provide opportunities for further hands-on learning, guided by the WTAs. Table 1 shows the lesson modules of DeapSECURE after all the changes had been completed in Y2. All the DeapSECURE lessons and their resources are available openly at DeapSECURE's project website [16].

While the training was still widely announced to any interested students at ODU, acceptance to the workshop was limited to those who had experience in writing simple computer programs (100 lines or less). This resulted in a smaller cohort at the beginning. The workshop format, structure, and length remained the same as the previous year. The workshop dates were compressed towards

Table 1: The DeapSECURE Lesson Modules (since Fall 2019)

Module	Lesson Description	Hands-on Activities	Toolkits
HPC	Introduction to HPC and how to access, use and program HPC systems	Analyzing countries of origin from a large collection of spam emails; using parallel processing on HPC to speed up data processing	UNIX shell commands, SLURM
CRYPT	Advanced cryptography for privacy-preserving computation	AES ciphertext cracking; “King Oofy” privacy-preserving census; Paillier encryption of bitmap image data	AES-Python [19], Python-paillier [5]
PAR	Parallel programming with MPI	Parallelization of image Paillier encryption	mpi4py [6], Python-paillier
BD	Big data (BD) analytics	Processing, cleaning, analyzing, and visualizing large SherLock dataset	Pandas, Matplotlib, Seaborn
ML	Machine learning (ML) modeling	Classification of smartphone apps based on system utilization data using classic ML methods	scikit-learn [14]
NN	Neural networks (NN) for deep learning modeling	Building neural networks to classify smartphone apps	TensorFlow [2] and Keras [4]

the beginning of the semesters (amounting to 3 workshops per semester) in an effort to improve retention. We were able to secure a large classroom with tables for collaborative work in small groups, which greatly improved hands-on learning. For the data-intensive module, we devised a hackish way to run Jupyter (a web-based interactive Python environment) on Turing HPC compute nodes and forward the output to learner's computers. While this was a great improvement over using vanilla Python / IPython interface, the set up procedure was very challenging for most learners, resulting in lost time. The assessment results were discussed in Ref. [17], comparing attendance and a subset of knowledge acquisition from both the first and second years. The hands-on part of the workshop was particularly well received by many learners. While there were indications of somewhat better outcome in the second year (e.g. attendance, learner's satisfaction rate), we still noticed challenges particularly in the area of knowledge acquisition from the workshops.

3 THIRD-YEAR DEVELOPMENT: FROM IN-PERSON TO ONLINE WORKSHOPS

The COVID-19 pandemic hit shortly after the second-year workshop series was completed. This forced the DeapSECURE team to change the structure and format of the workshops and make them ready for online delivery. The team conducted a pilot online workshop in the summer of 2020, using Zoom videoconferencing platform for synchronous instruction, Jupyter for hands-on activities, and Slack (a group-based messaging platform) for communications among team members and learners during and after each workshop. By this time, an Open OnDemand instance [9] has been set up for the newer Wahab cluster, which enabled convenient access to Jupyter environment. Based on the experience and lessons learned from this pilot workshop, we proceeded to convert all the DeapSECURE lesson modules to the online delivery format in the third year.

3.1 Lesson Format Redesign

While it is still possible to emulate a Carpentries-style hands-on instruction using Zoom, there are several challenges with this format: (1) It is difficult for instructors to get the sense where the learners are, and whether they are able to follow or have difficulties, since most learners tend turn their cameras off and be quiet in Zoom; (2) From the past years, the full hands-on learning of a DeapSECURE module could not be completed within the 3-hour time frame of the workshop, leading to incomplete knowledge delivery. Remote learning tends to be a self-directed process, where learners needs to have more autonomy in driving their own learning process; therefore a suitable online training format should account for this, while compensating the known challenges.

In the online format, Jupyter was the platform of choice for nearly all the modules except the first one (HPC), where command-line interaction on a UNIX shell was a major and essential part of the lesson. In Y3, a major effort was spent in producing Jupyter notebooks for the online workshops, between 2–3 notebooks per lesson module. The Jupyter notebooks were an abridged version of the web-based, Carpentry-style lessons produced by this project [16]. But unlike the web-based lessons, which contain mostly completed codes, these Jupyter notebooks contain partially completed codes which are to be completed by the learners as they are going through the notebooks. In this regard, we deviated from the teaching model of the Carpentries, which typically “unfolds” the computer codes from complete scratch. (Carpentries-style lessons are like textbooks, but they are generally intended for the instructors while preparing for their teaching, although a motivated learner can definitely use these lessons to learn hands-on computing skills independently.) This is an important design consideration that we took in order to make the notebooks usable for self-paced learning. One major challenge with online hands-on workshops is that learners can easily get lost when they fall behind the instructor. In an in-person workshops, instructors can easily identify learners that face difficulties from their gestures and facial expressions—something that is very hard to sense in a virtual workshop because most learners turn off their cameras. With their own notebooks, learners would have a way to catch up the missed part. The technical steps for creating Jupyter notebooks for online workshops was described in Ref. [7].

In addition to converting the lessons to the Jupyter format, much work was dedicated to improving and tuning the parts of the lessons to fill the gaps, devise better approaches to teach the concepts and/or skills. For every lesson, we sifted through the episodes and parts and identify the most salient parts of the concepts, codes, and exercises that will be included in the notebooks. Details that are important but too long to be included in the notebooks are referenced using links to the web-based lessons. This process was done to focus learners' attention only on those critical parts of the CI knowledge and skills:

(1) The HPC module was reworked to introduce basic parallel processing of independent tasks using only shell scripts (the previous version jumped directly to using GNU parallel, which did not give learners an opportunity to observe how the domain decomposition was performed).
(2) The CRYPT module guides learners to encrypt and decrypt data using homomorphic encryption (Paillier) as well as the standard AES encryption; compares and contrasts their strengths, limitations, as well as computational costs.
(3) In the PAR module, emphasis was placed on basic MPI "verbs" such as send, receive, broadcast and barrier; then followed by the step-by-step MPI parallelization of a simple "map-reduce"-style computation.
(4) The BD module focuses on the basic data processing building blocks (e.g. select, filter, sort, groupby, aggregate operations), followed by data wrangling and exploratory data analysis.
(5) In the ML and NN module, a greater priority was devoted to the key steps in a standard machine learning workflow, neural-network model construction, as well as basic model hyperparameter tuning. Full implementation of ML and NN on HPC became optional activities for learners that are keenly interested in the method.

All of these are the indispensable, rudimentary principles of the CI methods, which are the key opener for learning and utilizing these techniques.

3.2 Online Workshop Delivery

In the **third year (Y3, 2020–2021)**, three workshops (HPC, CRYPT, PAR) were conducted throughout the academic year, whereas the three data-intensive workshops (BD, ML, NN) in the summer of 2021. The extensive work of conversion to the online format caused delay in the scheduling of the workshops. We did not offer hackshops in the third year due to limitations in time and resources. Since learners have their own copies of notebooks, we expect that they should be able to continue learning after leaving the workshops.

The online workshops were carefully planned out, including the strict time allocation for every part therein. We still used a three-hour format (not including breaks) per workshop. The three-hour instruction was broken up to three one-hour sessions with short breaks in-between, each of which was a mix of a lecture and a hands-on work on Jupyter (or UNIX shell). The 30-minute cybersecurity research guest lectures were omitted in the online workshops conducted in the third year. Instead, faculty and advanced-stage Ph.D. students gave somewhat longer lectures with an overview of the CI methods, which included a brief overview of their own state-of-the-art cybersecurity research applications.

The Zoom breakout room feature was used to conduct the hands-on sessions in smaller groups (around 4-6 learners each). Each breakout room had a WTA that guided the learners through the notebooks. The original intent of using breakout rooms for hands-on learning was to encourage learners to open up and discuss the hands-on materials; but this generally did not occur. Initially, the learners went through the notebook on their own, which resulted in very slow pace and nearly. In latter workshops, based on a learner's input, the WTAs would actually share their Jupyter screens, talking over the materials while actively working through the code cells in their own notebooks (somewhat similar, but not identical to the Carpentries, because our Jupyter notebooks contain partially completed codes). Three breakout rooms were initially defined, designated "beginner/novice", "intermediate", and "advanced". Participants were assigned to each room based on their self-assessed computing skill levels that was self-assessed by the learners when signing up for the training. Later on, this procedure was changed to allow learners to choose any breakout room that they thought was appropriate for their skill levels. This freedom turned out to be boone for some participants: they felt they were able more comfortable at learning by choosing the appropriate level.

During Y3 workshops, we employed Kahoot online quiz platform [1] to provide additional opportunities for learners to be socially involved. With Kahoot, we did ask questions that were more specific, such as specific function names or call syntax, how a certain computation or action was programmed in Python, in addition to general questions.

3.3 Training Workshop Teaching Assistants

DeapSECURE was developed and piloted at ODU with the aim of eventually serving the community of cybersecurity research everywhere in the U.S. and beyond. There needs to be an effort to produce trainers and lesson developers to pave the way for continual development of DeapSECURE lessons and for scaling the training beyond ODU. We laid the groundwork toward this by establishing a framework to onboard and train WTAs in an ongoing basis. During the project duration, we have witnessed high turnover of the WTAs, although the 1–2 core WTAs remained with the project for at least two years. To ensure continuity and fast onboarding of new WTAs, we not only utilized collaborative lesson development practices and tools but also developed initial phases of the so-called "train-the-trainer" program, in which WTAs themselves acted as trainees first, by working through the lesson modules already existing (e.g. through the same Jupyter notebooks given to the workshop learners). They are then onboarded to the collaborative development methodology (Git/Gitlab, Jupyter, Jekyll). Afterwards, they can be brought into the ongoing collaborative work of developing, improving, and/or polishing the lesson structure and contents. We have also developed project wiki to document as much team knowledge in a single place, allowing latter WTAs to pick up the existing knowledge independently. We started this WTA training in Y2, where we trained and onboarded four Ph.D. students into the role of lesson developers [17]. In Y3, many of these Ph.D. students had taken other interests or responsibilities, and we had two existing Ph.D. students along with two undergraduate students. We worked closely with these students to perform the conversion to online workshops by producing the Jupyter notebooks for learners. This process has allowed us to successfully work with short-term WTAs, who were able contribute remotely, even if they are from a different university. At the end of the fourth year, we have trained a WTA from the University of Virginia to help us complete the web-based lessons for final release. Within a couple of weeks, the student was able to meaningfully contribute to the lesson modules and be well versed in the lesson materials using Jupyter notebooks. To have student-contributors from other Virginia institutions was a forward-looking decision toward the expansion of the project activities, as described in Section 6.

4 ASSESSMENTS AND LESSONS LEARNED

Training assessments were conducted in all the training workshops conducted by the program, whether in-person (Y1 and Y2) or online (Y3) as well as in mixed mode during the Summer of 2022 (Y4, elaborated later). Collected assessment information includes demographic data, opinion questions (perception) about the workshops, and pre- (PRE) and post-workshop (POST) knowledge questions. The knowledge questions measured general, high-level knowledge on the CI topics, instead of focusing on toolkit-specific or programming issues. The questionnaires in both years were largely the same (some minor changes were implemented along the way to improve knowledge testing), which enabled us to compare the effectiveness of our mid-project changes. In this paper we will focus only on certain demographic data and learners' perception about the workshops. (Analysis and study on the knowledge questions will be a topic of an upcoming publication.) In particular, we examine two opinion questions asked of the learners in both the second and third years. This may give us an insight into the contrast between the in-person and virtual formats.

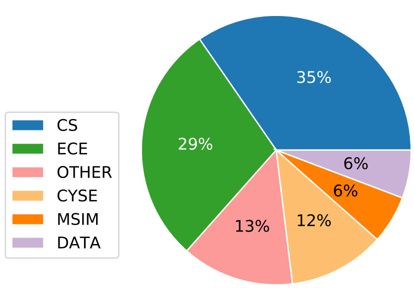


Figure 1: Distribution of Y3 workshop participants according to their academic majors. (Source: [7])

4.1 Learners' Profile

Figure 1 shows the distribution of the learners based on their academic majors in Y3. Not surprisingly, computer science (CS), electrical and computer engineering (ECE), and cybersecurity (CYSE) were the top three majors, collectively accounting for more than

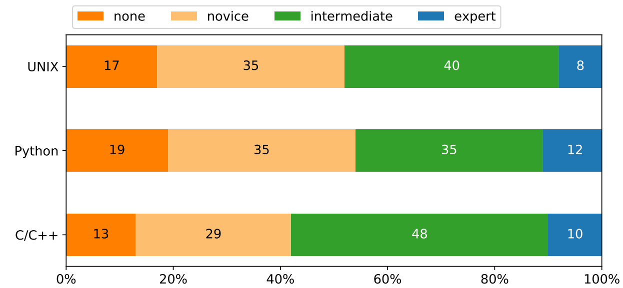


Figure 2: Distribution of programming skill levels in key programming languages (Unix shell, Python, and C/C++), self-assessed by the learners in Y3. (Source: [7])

75% of the learners. Other majors that are less prevalent include computational modeling and simulation engineering (MSIM) and data science (DATA). The OTHER category contains non-STEM majors and STEM majors representing less than 2% of participants per major, such as math and physics.

During the registration process, learners were asked to self-identify their skill levels (none, novice, intermediate, or expert) on Unix, Python, and C/C++. (This question was asked because we were interested to see if this factor would have any bearings in their perception of the workshops and their learning effectiveness.) Figure 2 shows the results of this questionnaire in Y3. Many participants were novice or intermediate in each programming tool, but a considerable number of them self-identify as intermediates for C/C++; this is likely due to C++ being taught as a required course for Engineering and computer-related majors at ODU.

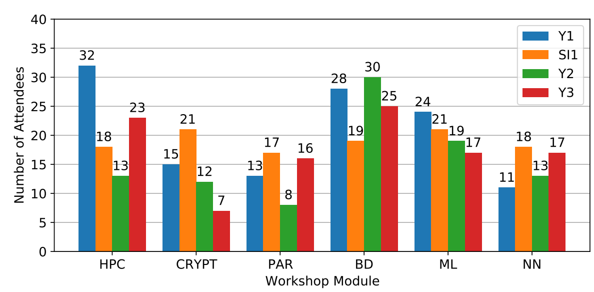


Figure 3: Number of learners attending individual workshops, reported for all the four complete rounds of DeapSECURE workshops (workshop series in Y1, Y2, Y3, as well as a summer institute [SI] at the end of Y1). (Source: [7])

4.2 Comparison of Workshops with In-person and Online Delivery

The attendance statistics of the DeapSECURE workshops is reported Fig. 3. We compare the attendance of the all-virtual workshops (Y3) with the other in-person training events (workshop series in Y1 and Y2, as well as a summer institute at the end of Y1). During Y3, where all workshops were delivered virtually, between 7 to 30 participants attended each workshop. Again, the workshop attendance

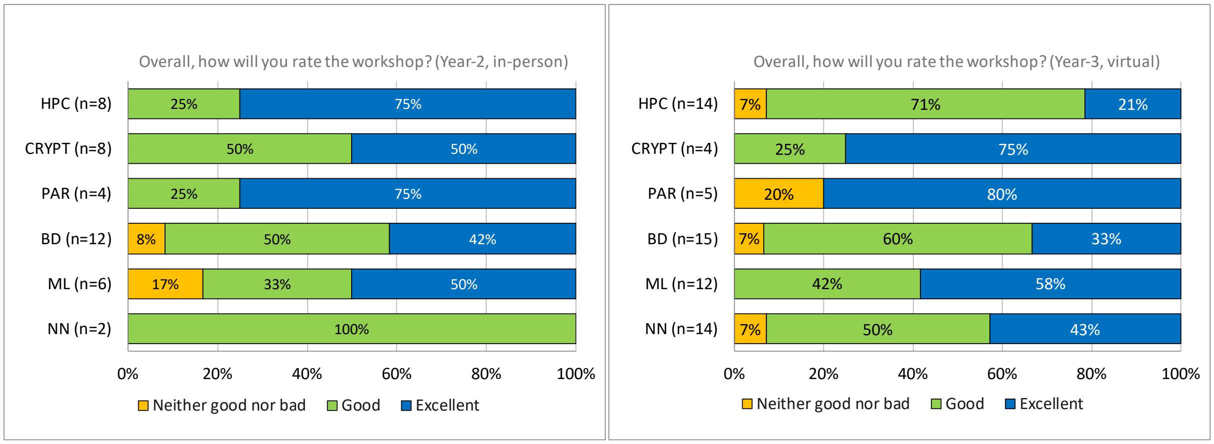


Figure 4: Percentage ratings of the six workshops given in Year 2 (left) and Year 3 (right) in response to the survey question "*Overall, how will you rate the workshop?*". The y-axis provides the workshop abbreviations (see Table 1, column 1) followed by the number n of opinions in parentheses.

was notably better and more consistent in the summer (the last three workshops in Y3) than during the academic year. The same attendance tendency was observed during the years of in-person delivery.

To get a comparative insight about the two delivery modes, online and in-person, we examine here an opinion question asked of the learners in *both* the second (Fig. 4, left) and third (Fig. 4, right) years. There were no radical differences in answers to the opinion and open-ended questions among the different years, where delivery changed from in-person to virtual (online). Note that, despite the low numbers of respondents, the comparisons in Fig. 4 are still fair since the low numbers are consistent across all the workshops (see the n-values) with slightly more responses received in Y3, which also corresponds to Y3 workshops having somewhat more attendees on average than those of Y2. From Fig. 4, note that, some workshops were consistently rated higher than others across the two years. For example, the ML rating was higher than that of BD in both years and PAR was higher than CRYPT. These relative ratings might correlate with (1) the perceived final applicability of the lessons to the cybersecurity task at hand, and (2) the continuity of the module materials. For example, the hands-on activities in the ML module led to the inferences for smartphone apps, whereas in the BD module, the activities mainly involved data handling and exploratory analysis. For the PAR module, the use of Python-paillier, the same tool to which the learners were introduced in CRYPT, might have contributed to the former's higher rating. The HPC module was rated lower in Y3; learners were split whether the lesson was too easy or too hard. The HPC module included a quick overview of UNIX shell commands, which topic is very hands-on in nature and require much practice to master. From the survey, we discovered the following: Because this module was taught using command-line interface, it might have been very challenging to learners who never used such a interface before, yet for others who had used shell for a period of time, this overview might have been considered a waste of time. This observation seems to support the notion that a command-line-based workshop is significantly harder to do virtually than in-person. The ML and NN modules received slightly higher ratings in Y3, which might have been due to the improved lessons in Y3.

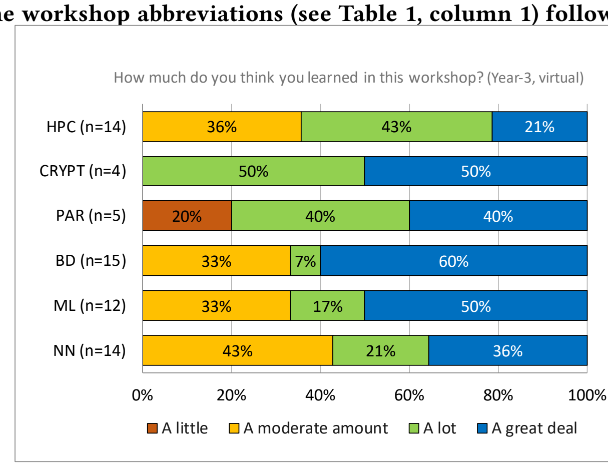


Figure 5: Percentage ratings of the six workshops given in Y3 in response to the survey question "*How much do you think you learned in this workshop?*". The y-axis provides the workshop abbreviations (see Table 1, column 1) followed by the number n of opinions in parentheses.

Learners' preferences are also reflected in Fig. 5, which shows Y3 outcomes to the opinion question: "How much do you think you learned in this workshop?" It is interesting to note that the BD module, which spent much time on the tedious handling of data in pandas, received a larger percentage of the highest ratings (compared with its *overall* rating in Fig. 4, right panel). In particular the highest ratings were 60% vs 33%, respectively. Of all the modules, the lowest rating was given for the PAR module by 20% of learners. We reckon that, in general, there may be two possible reasons for the perception of learning only little or moderately: (1) The concepts and hands-on material are very new, so that learners cannot keep up in absorbing exercises with respect to their applicability; (2) Conversely, the topics taught might have been quite familiar to learners, so that the concepts taught and exercises fill only small gaps in learner's knowledge and skill. In the case of the PAR module, the first reason is much more plausible because this module considers parallel programming with Message Passing Interface (MPI) (see Table 1), which is a very advanced topic typically taught only to upperclassmen and graduate students.

The responses from surveys and knowledge questions were instrumental in driving the iterative improvements of the training through its four years of development. We are aware that the responses to the opinion questions such as those reported in Figs. 4 and 5 do have their limitations; in particular, they are subject to respondents' biases, including their educational backgrounds, computing skills, etc. Nevertheless, they may give useful indicators on the areas needing improvement. In cases where improvements are needed, a focus group interview with the survey respondents might be valuable.

We found additional insights by analyzing responses to two open-ended questions: "What is most valuable about this training?" and "What is least valuable about this training?" (which will thereafter be abbreviated as "most valuable" and "least vaulable"). Responses from the open-ended questions in the post-workshop survey were analyzed by scanning for keywords (i.e. "hands-on") or themes (i.e. topic-related keywords like "encryption") and quantified. On Y2, there were 40 and 38 responses to the "most valuable" and "least valuable" open-ended questions, respectively. For Y3, the number of responses were 30 and 29. In general, learner's feedback consistently showed that participants enjoyed the hands-on component of the training, which evolved and was augmented over the project years. For both Y2 and Y3, 58% and 27% of respondents mentioned the hands-on training as the most valuable aspect of the training (with "Jupyter notebooks" repeatedly mentioned in Y3). On Y2, 38% of responses cited programming- or coding-related aspects (i.e. learning about different Python operations) as the most valuable. On Y3, most of the responses (40%) point to the topic or exposure to the training as the most valuable aspect. It should be mentioned that on Y3, 13% of responses indicated that the teaching assistants were the most valuable part of the training. Learners were generally happy with the training, as majority of respondents indicated that nothing was the "least valuable" part of the training in (73% in Y2, 68% in Y3). Upon further analysis on the "least valuable" responses, we found the following: Challenges with pace or insufficient time (14% in Y2, 7% in Y3); The material was difficult (10% in Y2, 3% in Y3). It is encouraging that the pace and level of materials seemed to have improved in Y3, based on learners' perception.

4.3 Lessons Learned

Through the four years of improving the training and conducting workshops, we have gained a number of important lessons. In terms of participation and attendance, there is no doubt that offering this training as a summer institute leads to the best level of engagement and learning, as students are completely focused on the training for a concentrated period of time. In the future, however, it might help to provide additional engagement opportunities in the year that follows the summer institute by offering seminars on cybersecurity research topics that leverage CI techniques, or small group meetups to work on specific challenges utilizing CI techniques. In general, unless there is a research driving needs, the students' participation and engagement will be somewhat limited to general literacy on CI.

Another important lesson learned is related to the timing of the workshop. It seems that devoting a whole-day workshop might be more appropriate for each DeapSECURE lesson module, to allow sufficient time to work through the notebooks. It is very important, however, to provide a way for learners to check-in at various stages, in order to keep up with their progress. This could be an important change that we will implement in the coming year.

Based on the level of materials presented in DeapSECURE lessons, the prerequisite for participation may need to be raised up so that learners will be able to engage with the presented CI techniques much more effectively. While currently we simply required participants to self-evaluate if they were able to write a 100-line code (or less), it may be better to require them to have command-line experience and Python programming experience. This can be satisfied, for example, by completing both the Software Carpentry's "Unix Shell" [8] and "Plotting and Programming with Python" [10] lessons prior to enrolling to the DeapSECURE training.

5 OPEN-SOURCE RELEASE AND COMMUNITY ADOPTION

The fourth and last year of DeapSECURE under the funding from NSF (**Y4, 2021–2022**) was spent completing all the lesson modules and hands-on materials, and releasing them open-source. The Big Data module has been completely released [15]; other modules are under review and refinement to become open-source. We expect to release all the data-intensive modules by the end of 2022. All the lessons will be released using CC-BY-4.0 license and all the codes with MIT license, compatible with lessons from the Carpentries.

We had two outreach activities to gauge the community interest in a training in the cross-cutting areas of cybersecurity and HPC. Firstly, a pilot workshop was conducted in the Fall of 2021, targeting students across Virginia, leveraging a blend of BD and ML lessons to teach students the basics of data analytics and machine learning. Secondly, we also conducted a small "community interest survey" gathering input and interest by faculty and researchers across Virginia on DeapSECURE training. From both the survey and the pilot workshop, we discovered great interest in adopting and leveraging DeapSECURE beyond ODU. We gathered nine responses from the community interest survey, with many indicating an interest in adopting DeapSECURE lessons for their own instruction. Some of the respondents would like to have hands-on workshops offered at their institutions, and/or customize DeapSECURE lessons for teaching.

In the Summer of 2022, a three-day summer institute was held to teach the DeapSECURE data-intensive modules to students in the Cybersecurity Research Experience for Undergraduate Students (REU) program at ODU. This institute was well received, and the lessons taught were instrumental in bringing the REU students up-to-speed with their summer research activities involving artificial intelligence and machine learning. During this institute, several Ph.D. students who were not part of the DeapSECURE team were quickly onboarded to teach the materials to the REU students. We were encouraged with how quickly the Ph.D. students were able to assimilate the lesson materials and step up to teach them. The success of this institute is an indicator that (1) the DeapSECURE lessons and teaching methodology have matured to the point that they are ready for a wide-range of instructors to take up and teach to others, and (2) graduate students may become quickly proficient in teaching the lessons.

6 SUMMARY AND FUTURE DIRECTION

Here we summarize major impacts of the DeapSECURE training on student future and carriers:

- Over the years DeapSECURE workshops had been offered, a number of students became interested in learning CI techniques in-depth and followed this up by taking formal HPC- and BD-related courses at ODU.
- At least one undergraduate student decided to pursue a M.S. degree in cybersecurity after attending the DeapSECURE workshop series (summer institute).
- DeapSECURE has been instrumental in augmenting REU students' interest in cybersecurity with HPC skills (Summer of 2019, 2021, 2022). This impact was evidenced by their final project posters, some of which embodied AI/ML work carried out on ODU's HPC cluster.
- A number of DeapSECURE learners (both undergraduate and graduate students) had continued their interest in CI/cybersecurity intersection by becoming WTAs in subsequent years. Their participation as WTAs afforded them intensive training in programming, in using state-of-the-art software development tools and methodologies, in team work, and in pedagogy (teaching) [7].

We plan to expand the current project both in-depth and in-breadth manners with the overarching goal to produce a community of practice (CoP) of next-generation cybersecurity researchers and scholars who are well-versed in leveraging CI technologies and methods—such as HPC, big data, AI, advanced cryptography and privacy protection, parallel computing. In particular, we plan to provide training for learners of different levels (depth) such as faculty, researchers, and graduate students by leveraging our initial experiences in training WTAs and expanding this to a full-fledged "train-the-trainer" program that is designed to be synergistic with research, teaching, and learning activities carried out by faculty, postdocs, and graduate students. We also plan to incorporate the the training modules into curriculum/instructional material fabric in various institutions in Virginia and beyond (thereby increasing the breadth of training application). In addition, we will closely engage and collaborate with Virginia Commonwealth Cyber Initiative (CCI) to strengthen, expand, and enrich the CI training program and scale up the effort and impact to state-wide and beyond. The project team will collaborate closely with CCI and its members from higher education institutions, industry, government, and non-governmental and economic development organizations.

ACKNOWLEDGMENTS

The development of DeapSECURE training program is supported by NSF CyberTraining grant #1829771. The workshops utilized Turing and Wahab HPC clusters provided by ODU Research Computing Services, part of Information Technology Services. Wahab cluster was acquired in part using NSF Major Research Instrumentation grant #1828593. We gratefully acknowledge our current and prior workshop teaching assistants: Liuwan Zhu, Qiao Zhang, Jacob Strother, Rosby Asiamah, and Orion Cohen for their excellent contribution to the training materials and to the teaching of the workshops in Y3 and Y4. We also acknowledge Rui Ning, Kazi Aminul Islam, Md. Shibly Sadique, and Joseph Zalameda for their guest contributions in teaching the DeapSECURE workshops in Y3 and Y4. We thank the ODU Distance Learning for their support in recording the workshop sessions, and the anonymous reviewers of the extended abstract, whose comments helped sharpen the presentation of this paper.

REFERENCES

[1] 2020. Kahoot! Game-based Learning Platform. https://kahoot.com

[2] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/ Software available from tensorflow.org.

[3] Erin Becker and Fran cois Michonneau. 2022. The Carpentries Curriculum Development Handbook. https://cdh.carpentries.org/

[4] François Chollet and Keras team. 2015. Keras. https://keras.io.

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

[6] Lisandro D. Dalcin, Rodrigo R. Paz, Pablo A. Kler, and Alejandro Cosimo. 2011. Parallel distributed computing using Python. *Advances in Water Resources* 34, 9 (2011), 1124 – 1139. https://doi.org/10.1016/j.advwatres.2011.04.013

[7] Bahador Dodge, Jacob Strother, Rosby Asiamah, Karina Arcaute, Dr. Wirawan Purwanto, Dr. Masha Sosonkina, and Dr. Hongyi Wu. 2022. DeapSECURE Computational Training for Cybersecurity: Third Year Improvements and Impacts. http://www.modsimworld.org/papers/2022/MSVSCC_2022_InfrastructureSecurityMilitary.pdf

[8] Gabriel A. Devenyi (Ed.), Gerard Capes (Ed.), Colin Morris (Ed.), Will Pitchers (Ed.), Greg Wilson, Gerard Capes, Gabriel A. Devenyi, Christina Koch, Raniere Silva, Ashwin Srinath, and ... Vikram Chhatre. 2019. swcarpentry/shell-novice: Software Carpentry: the UNIX shell, June 2019 (Version v2019.06.1). (July 2019). http://doi.org/10.5281/zenodo.3266823

[9] 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 (XSEDE16)*. ACM, New York, NY, USA, Article 43, 7 pages. https://doi.org/10.1145/2949550.2949644

[10] Allen Lee, Nathan Moore, Sourav Singh, and Olav Vahtras (eds). 2018. Software Carpentry: Plotting and Programming in Python. (2018). http://github.com/swcarpentry/python-novice-plotting

[11] Yisroel Mirsky, Asaf Shabtai, Lior Rokach, Bracha Shapira, and Yuval Elovici. 2016. SherLock vs Moriarty: A Smartphone Dataset for Cybersecurity Research. In *Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security (AISec '16)*. ACM, 1–12. https://doi.org/10.1145/2996758.2996764

[12] 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

[13] The pandas development team. 2020. pandas-dev/pandas: Pandas. https://doi.org/10.5281/zenodo.3630805

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* 12 (2011), 2825–2830.

[15] Wirawan Purwanto and DeapSECURE Team. 2022. Open-Source Release of the DeapSECURE 'Big Data' Lesson Module to the Community. https://deapsecure.gitlab.io/posts/2022/02/release-big-data-lesson/.

[16] 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/

[17] 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. https://doi.org/10.22369/issn.2153-4136/12/2/1

[18] 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

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

Benchmarking Machine Learning Models on a Dielectric Constant Database for Bandgap Prediction

Mohammad Hadi Yazdani
Mork Family Department of Chemical Engineering & Materials Science
University of Southern California
yazdanim@usc.edu

Paulo S. Branicio
Mork Family Department of Chemical Engineering & Materials Science
University of Southern California
branicio@usc.edu

Ken-ichi Nomura
Mork Family Department of Chemical Engineering & Materials Science
University of Southern California
knomura@usc.edu

ABSTRACT

In this study, we investigate the performance of several regression models by utilizing a database of dielectric constants. First, the database is processed using the Matminer Python library to create features, and then divided into training, validation, and testing subsets. We evaluate several models: Linear Regression, Random Forest, Gradient Boosting, XGBoost, Support Vector Regression, and Feedforward Neural Network, with the objective of predicting the bandgap values. The results indicate superior performance of tree-based ensemble models over Linear Regression and Support Vector Regression. Additionally, a Feedforward Neural Network with two hidden layers demonstrates comparable proficiency in capturing the relationship between the features generated by Matminer and the bandgap target values.

KEYWORDS

Supervised Learning, Linear Regression, Random Forest, Gradient Boosting, XGBoost, Support Vector Machine, Neural Network

1 INTRODUCTION

The field of Materials Informatics represent a data-centric methodology aimed at accelerating innovations in materials design and discovery [9]. Currently, an array of open-source software is available for materials scientists and engineers, facilitating the integration of informatics into their research. Notably, Matminer–an open-source Python library designed for materials informatics–has gained popularity due to its extensive suite of tools for data extraction and analysis, robust feature extraction capabilities, and open APIs that provide unrestricted access to online databases of materials data [10].

The Matminer dielectric constant dataset [8] is a comprehensive repository of data encompassing the dielectric properties of over 1,000 inorganic compounds as well as additional attributes such as formation energy, band gap, and melting point. The dielectric constant, also known as the relative permittivity, quantifies the capacity of a material to store electrical energy when subjected to an electric field, making it a crucial parameter for materials in the realms of electronics and energy storage. This dataset is a valuable resource for materials science and engineering researchers focused on developing novel materials for applications such as capacitors, solar cells, and sensors. The dataset was generated using Density Functional Perturbation Theory utilizing the Perdew-Burke-Ernzerhof (PBE) functional.

https://doi.org/10.22369/issn.2153-4136/15/1/2

Density Functional Theory (DFT) is a powerful quantum mechanical theory that accurately describes many material properties at their ground state. Density Functional Perturbation Theory (DFPT) [1] builds upon DFT to incorporate the effects of an external perturbation, such as changes in the electronic structure induced by an external electric field. This extension enables the calculation of a wide range of material properties, such as dielectric constants, phonon frequencies, and piezoelectric coefficients. PBE functional developed by Perdew, Burke, and Ernzerhof, based on the generalized gradient approximation (GGA) [7] is widely used to create reliable materials datasets.

Band gap is a fundamental concept in materials science and solid-state physics that plays a crucial role in determining the electrical and optical properties of a material. It is defined as the energy gap between the top of the valence band and the bottom of the conduction band within a material. Materials with wider band gaps are typically insulators, whereas those with a narrow or nonexistent band gap act as semiconductors or conductors, respectively. Understanding the band gap of a material is essential for designing and optimizing a wide range of electronic and photonic devices, as it determines how the material responds to electrical and optical stimuli.

2 METHODS

Linear Regression (LR) is a widely used statistical method for modeling the relationship between a dependent variable and one or more independent variables, specifically capturing their linear correlation. Although LR exhibits a comparatively high model bias, it remains extensively utilized in practical applications ranging from stock price forecasts to the analysis of experimental data, largely due to its strong generalizability and interpretability. Furthermore, LR serves as the cornerstone for numerous sophisticated regression methods, rendering it an indispensable instrument for data analysts and researchers across diverse disciplines.

Random Forest (RF) [2] is a machine learning algorithm widely utilized for both regression and classification tasks. It operates by constructing a set of many decision trees, each generated from a subset of features, thereby ensuring a diverse population of models.

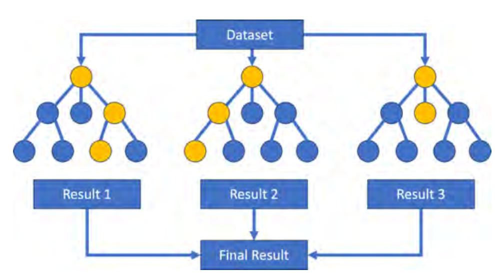


Figure 1. Random Forest model

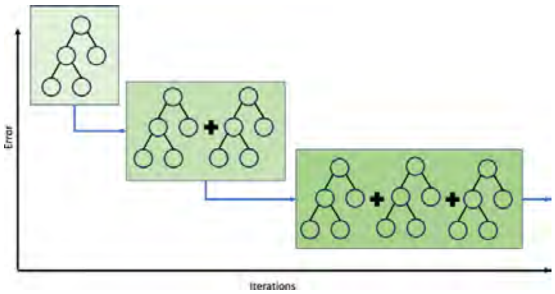


Figure 2. Gradient Boosting model

An RF model aggregates the predictions made by each individual tree, using either the mean or mode as its final prediction. RF has been very popular due to its capability of handling high-dimensional datasets with numerous and diverse features. Another significant advantage of RF is its robustness against overfitting, which is a common issue when a single decision tree is trained on a complex dataset.

Gradient Boosting (GB) [5] is a machine learning technique employed for both regression and classification tasks. Similar to RF, the goal of GB is to generate numerous decision trees to cover a large model population. However, GB distinguishes itself by constructing trees sequentially: each new tree is built to correct the errors made by the previous ones. This is achieved by fitting the new tree to the negative gradient of the loss function, which represents the direction in which the model should be adjusted to improve accuracy. This iterative process continues until the model reaches a predefined level of precision. GB's capacity to manage complex datasets and yield highly precise predictions has made it a favored algorithm in diverse domains such as natural language processing, computer vision, and recommendation systems.

XGBoost [3], short for eXtreme Gradieng Boosting, enhances traditional gradient boosting methods through a suite of algorithmic improvements. These enhancements accelerate model training and increase predictive accuracy. XGBoost incorporates several regularization algorithms, such as Shrinkage and Column Subsampling, which help prevent overfitting during tree generation and improve its overall generalization capabilities. Additionally, XGBoost is designed to exploit modern CPU and GPU architectures for computational efficiency. The combination of these enhancements makes XGBoost a highly desired tool for various applications, such as customer behavior prediction in marketing and medical data analysis in healthcare.

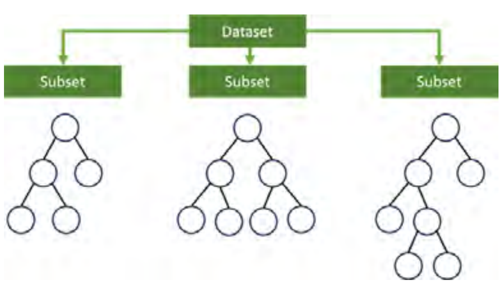


Figure 3. XGBoost model

Support Vector Machine (SVM) [4] is a machine learning algorithm designed to optimize the margin between the decision boundary and the nearest data points, known as support vectors, to improve its predictive generalizability on new data. Support Vector Regression (SVR) is a popular regression model based on SVM algorithm that has been implemented in numerous machine learning libraries. SVR utilizes kernel functions, such as the radial basis function (RBF) or a polynomial function, to model non-linear relationships between input features and target variables effectively. These kernel functions facilitate the mapping of input data to a higher-dimensional space, where linear separation is possible. The robustness of SVR makes it suitable for diverse applications across fields such as finance, engineering, and biology.

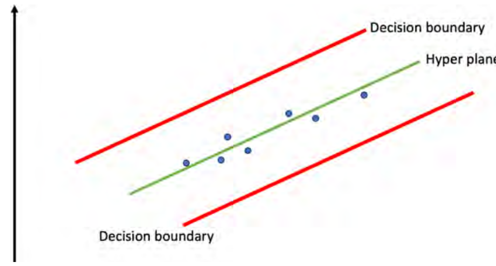


Figure 4. Support Vector Regression model

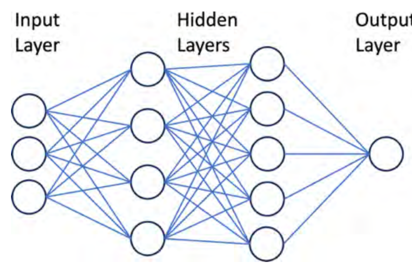


Figure 5. Feedforward Neural Network model

A Feedforward Neural Network (FFNN) [6] is a machine learning architecture that comprises multiple layers of nodes or neurons. These layers include an input layer that takes in data, followed by several hidden layers that process the data sequentially, and an output layer that delivers the final prediction. Each neuron in one layer is connected to neurons in the subsequent layer through weights. These weights are iteratively adjusted during the training phase to minimize the discrepancy between the prediction of the network and the actual data. FFNN are renowned their ability to learn complex patterns within datasets and make accurate predictions, making them a popular choice for many machine learning tasks such as image and speech recognition, natural language processing, and financial forecasting.

At first, the dielectric constant dataset from Matminer is used, and the bandgap feature is designated as the target variable. Supplementary input features comprise the chemical formula, the refractive index (denoted as n), the space group (an integer specifying the crystallographic structure of the material), the structure (presented as a pandas Series defining the structure of the material), the number of sites (nsites, representing the number of atoms in the unit cell of the calculation), the volume of the cell, among others.

To enrich the dataset with additional features, specific featurizers from the Matminer library are employed. These featurizers are algorithms designed to extract meaningful information from the raw data, transforming it into quantifiable attributes that can be utilized by machine learning models to improve their predictive performance. The following featurizers are used:

1. *matminer.featurizers.composition.ElementProperty:* This feature extractor calculates elemental properties such as atomic number, atomic mass, atomic radius,

electronegativity, and so on, for a given chemical composition.

2. *matminer.featurizers.structure.DensityFeatures:* This feature extractor calculates various features related to the density of a crystal structure, such as the total volume of the unit cell, the packing fraction, and the Voronoi volume of each atom.
3. *matminer.featurizers.structure.CoulombMatrix:* This feature extractor calculates a matrix of pairwise interactions between atoms in a crystal structure, based on their charges and distances from each other.
4. *matminer.featurizers.composition.OxidationStates:* This feature extractor calculates the most likely oxidation states of each element in a given chemical composition, based on the electronegativity and coordination number of each element.
5. *matminer.featurizers.structure.ElectronicRadialDistribution*: Function: This feature extractor calculates the distribution of electron density around each atom in a crystal structure, as a function of radial distance from the atom.

The dataset comprises 166 potential features and is partitioned into training, validation, and testing sets with proportions of 70%, 15%, and 15%, respectively.

The Scikit-learn library is employed for the construction of the linear regression models. The R^2 score and the Root Mean Square Error (RMSE) are utilized as metrics for model evaluation. For the RF model, the ensemble comprises 1,000 trees, as indicated by the number of estimators. The GB model incorporates 100 estimators, adopts a learning rate of 0.2, and a maximum depth of 5, with each new tree intended to enhance the performance of the model incrementally. The XGBoost model is set with 50 estimators. For the SVR model, the RBF kernel is chosen with a gamma of 8×10^{-7} and a margin of error or epsilon of 0.1. The RBF kernel is favored for its efficiency in mapping input features into a higher-dimensional space. The FFNN is designed with two hidden layers, containing 128 and 64 units, respectively, a dropout rate of 0.1, and a mini-batch size of 16. The FFNN is trained using the Adam optimizer over 200 epochs.

3 RESULTS AND DISCUSSION

The outcomes of the six models are delineated in Table 1.

Table 1. Comparative performance of six regression models.

Model	Training		Validation		Test	
	R^2	RMSE	R^2	RMSE	R^2	RMSE
LR	0.732	0.831	0.582	1.081	N/A	N/A
RF	0.975	0.239	0.752	0.833	0.780	0.708
GB	0.999	0.045	0.804	0.740	0.818	0.644
XGBoost	0.999	0.029	0.783	0.778	0.818	0.643
SVR	0.943	0.384	0.535	1.140	0.405	1.164
FFNN	0.978	0.091	0.819	0.504	0.820	0.395

From the results shown in Table 1 one can note that the LR model is inadequate for capturing the non-linearity inherent in the actual data, as evidenced by its inferior results on the validation set. While the SVR yielded satisfactory outcomes during the training phase, its performance on the validation and testing sets is suboptimal. In contrast, the tree-based ensemble models, i.e., RF, GB, and XGBoost, exhibited superior performance, underlying their capability and efficiency. The FFNN shows demonstrate a comparable performance to the ensemble models displaying robust results on the testing subset.

4 CONCLUSIONS

This investigation assessed six regression models applied to a dielectric constant materials dataset, with an emphasis on predicting bandgaps. The results indicate that LR and SVR models yield the least satisfactory results for this application. In contrast, the GB and XGBoost methods as well as the FFNN architecture delivered the most accurate predictions. This demonstrates their superior capacity to learn complex input-output relationships, making them well-suited for tasks requiring high accuracy and the analysis of extensive datasets.

All ML models were implemented on Jupyter Notebook, which substantially increased the student's engagement and comprehension of the algorithms in this project. With the interactive platform and GUI interface, the student could easily evaluate the significance of input parameters in the machine learning algorithm. This project also honed students' analytical skills and hands-on experiences, providing a profound awareness of their potential for materials informatics and real-world engineering applications.

ACKNOWLEDGEMENTS

K.N. is partially supported by the NSF grant OAC-2118061.

REFERENCES

[1] Stefano Baroni, Stefano de Gironcoli, Andrea Dal Corso, and Paolo Giannozzi. 2001. Phonons and related crystal properties from density-functional perturbation theory. *Reviews of Modern Physics* 73, 2, 515.

[2] Leo Breiman. 2001. Random forests. *Machine Learning* 45, 5-32.

[3] Tianqi Chen, and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *KDD'16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794.

[4] Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. 1996. Support vector regression machines. In *Advances in Neural Information Processing Systems* 9. papers.nips.cc/paper_files/paper/1996

[5] Jerome H. Friedman. 2001. Greedy function approximation: A gradient boosting machine. *Annals of Statistics* 29, 5, 1189-1232.

[6] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT press.

[7] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. 1997. Generalized gradient approximation made simple. *Phys. Rev. Letters* 77, 18, 3865.

[8] Ioannis Petousis, David Mrdjenovich, Eric Ballouz, Miao Liu, Donald Winston, Wei Chen, Tanja Graf, Thomas D. Schladt, Kristin A. Persson, and Fritz B. Prinz. 2017. High-throughput screening of inorganic compounds for the discovery of novel dielectric and optical materials. *Scientific Data* 4, 1, 1-12.

[9] Rampi Ramprasad, Rohit Batra, Ghanshyam Pilania, Arun Mannodi-Kanakkithodi, and Chiho Kim. 2017. Machine learning in materials informatics: Recent applications and prospects. *Computational Materials* 3, 1, 54.

[10] Logan Ward, Alexander Dunn, Alireza Faghaninia, Nils E.R. Zimmermann, Saurabh Bajaj, Qi Wang, Joseph Montoya, Jiming Chen, Kyle Bystrom, Maxwell Dylla, Kyle Chard, Mark Asta, Kristin A. Persson, G. Jeffrey Snyder, Ian Foster, and Anubhav Jain. 2018. Matminer: An open source toolkit for materials data mining. *Computational Materials Science* 152, 60-69.

Bridging the Quantum Gap: Addressing Challenges in Training Individuals in Quantum Computing Using Self-Guided Learning Resources

Stefan Seegerer
IQM Quantum Computers
stefan.seegerer@meetiqm.com

Mikio Nakahara
IQM Quantum Computers
mikio.nakahara@meetiqm.com

ABSTRACT

The convergence of quantum technologies and high-performance computing offers unique opportunities for research and algorithm development, demanding a skilled workforce to harness the quantum systems' potential. In this lightning talk, we address the growing need to train experts in quantum computing and explore the challenges in training these individuals in quantum computing, including the abstract nature of quantum theory, or the focus on specific frameworks. To overcome these obstacles, we propose self-guided learning resources that offer interactive learning experiences and practical framework-independent experimentation for different target audiences.

KEYWORDS

quantum computing, interactive learning environments, education

1 INTRODUCTION

As quantum technologies inch closer to practical applications, there is a pressing need to train a new generation of experts capable of harnessing the power of quantum computing systems effectively and advancing the research around quantum computing. As anticipated use-cases of quantum computing overlap strongly with existing applications of high-performance computing (HPC) [3], there is increased demand in the HPC world in providing training and learning opportunities to support a workforce equipped with the knowledge and skills to exploit quantum systems' potential [6].

In this paper, we delve into the challenges faced in training people in quantum computing such as the abstract nature of quantum theory, including superposition and entanglement, which pose a steep learning curve for individuals transitioning from classical computing paradigms. To address these challenges, we introduce the concepts that guided the development of a comprehensive learning platform aimed at facilitating the training of individuals in quantum computing. The proposed self-guided learning resources provide interactive and immersive learning experiences, encompassing theoretical foundations and practical implementations.

We believe that the self-guided learning resources can help bridge the gap in quantum computing education, enabling individuals to acquire the necessary skills to contribute to the emerging quantum landscape. Through a combination of accessible learning resources, practical experimentation, and a collaborative ecosystem, this platform can empower learners to harness the full potential of quantum computing and accelerate the advancement of quantum technologies in the HPC domain.

https://doi.org/10.22369/issn.2153-4136/15/1/3

2 CHALLENGES IN PROVIDING QUANTUM COMPUTING EDUCATION

Educating a wide range of high-performance computing (HPC) users in the field of quantum computing poses several challenges. Quantum computing is a rapidly evolving field that combines elements of physics, computer science, and mathematics and is considered to be counterintuitive, making it difficult for individuals to grasp its fundamental concepts.

2.1 Diverse Backgrounds

HPC users come from various backgrounds, including computer science, physics, mathematics, engineering, chemistry, meteorology and material science with initiatives to expand to the vast array of social, behavioral, or economics disciplines [8]. Educating such a diverse audience requires finding common ground and conveying quantum concepts in a manner that is accessible and understandable to individuals with different backgrounds. This is especially true in view of the frequent use of physics vocabulary in the field of quantum computing.

2.2 Abstract Nature of Concepts

Quantum computing operates on principles that deviate significantly from classical computing. Concepts like superposition, entanglement, and quantum algorithms can be abstract and counterintuitive [11], making them challenging to grasp for newcomers. Translating these abstract concepts into tangible and relatable examples is crucial to facilitating understanding among a wide range of users.

2.3 Rapid Technological Advancement

Quantum computing is a fast-paced field with frequent breakthroughs and advancements. This challenge is compounded when educative initiatives focus on learning a specific quantum computing framework or application programming interface (API) that is subject to change. Quantum computing frameworks such as Qiskit, Cirq, or Forest, are essential tools for developing and running quantum algorithms. However, these frameworks often undergo updates, improvements, and even major revisions to accommodate advancements in hardware, algorithmic breakthroughs, or community feedback. As learners show difficulties in semantic transfer on constructs with different syntax but same semantics [9], relying solely on a specific framework or API can limit adaptability and hinder the ability to keep up with the rapid evolution of the field.

3 EDUCATING A WIDE RANGE OF USERS IN QUANTUM COMPUTING USING SELF-GUIDED LEARNING RESOURCES

Different self-guided learning resources (e.g. [4, 10]) have been proposed to address the challenge of providing quantum computing education to a wider audience either by offering direct learning opportunities or by being used as an additional resource in in-person training. However, many of these options tend to presuppose prior knowledge, resemble technical documentation for specific frameworks, or delve excessively into advanced physics concepts. These characteristics can impede the accessibility and comprehension for beginners and non-experts. To this end, we have developed self-guided learning resources available through IQM Academy[1] addressing the aforementioned challenges as outline below.

3.1 Focusing the Underlying Principles and Concepts of Quantum Computing

It is crucial for learners to develop a solid foundation in the underlying principles and concepts of quantum computing, allowing them to understand the fundamental building blocks that drive various frameworks and APIs and that stay relevant in the long term. By focusing on core concepts and gaining a broader understanding of quantum computing principles, users can adapt to changes in frameworks and APIs more effectively, enabling them to navigate the evolving landscape of quantum computing with greater flexibility and agility.

3.2 Low Floors and High-ceilings

In order to cater to *diverse backgrounds* of learners, the resources should cater different levels and be easy to pick-up for beginners, but also provide challenging content for advanced learners – this is often referred to as having low floor and high-ceilings [5] and can be facilitated through fine-grained modularization. Following the style of exploratory learning [7], IQM Academy provides interactive applets that invite learners to first experience the concepts hands-on, which also helps to overcome the *abstract nature of concepts*.

3.3 Using Different Frameworks to Highlight Concepts

In learning, code serves particularly as a tool for thought during the learning process, allowing individuals to mold and articulate ideas. However, fixating too much on particular aspects of a programming framework can obstruct the understanding of the fundamental concept. Therefore, for any curriculum striving for profound learning and transfer, it is crucial to deliberately integrate strategies that promote transfer [1]. By avoiding excessive focus on a specific framework and instead utilizing implementations in different frameworks presented next to each other to illustrate the concept (see fig. 1), we enable a comparison of various ways to express the same idea. This approach facilitates a deeper understanding and appreciation of the underlying principles in the context of *rapid technological advancements* [2].

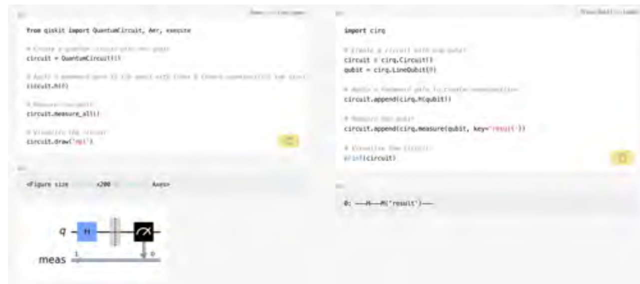


Figure 1: Parallel support for Qiskit, cirq and QASM (not displayed) fosters deeper understanding of underlying ideas.

4 CONCLUSION

While educating a wide range of HPC users in quantum computing may be challenging, it is an essential endeavor to unlock the full potential of this technology. With concerted efforts and effective educational strategies, we can bridge the gap and empower a broader community to explore, understand, and contribute to the exciting world of quantum computing. IQM Academy is addressing this educational need by following above concepts and is available for free.

REFERENCES

[1] Shuchi Grover. 2021. Teaching and assessing for transfer from block-to-text programming in middle school computer science. *Transfer of learning: Progressive perspectives for mathematics education and related fields* (2021), 251–276.
[2] Gongbing Hong, Jenq-Foung Yao, Chris Michael, and Lisa Phillips. 2018. A multilingual and comparative approach to teaching introductory computer programming. *Journal of Computing Sciences in Colleges* 33, 4 (2018), 4–12.
[3] Travis S Humble, Alexander McCaskey, Dmitry I Lyakh, Meenambika Gowrishankar, Albert Frisch, and Thomas Monz. 2021. Quantum computers for high-performance computing. *IEEE Micro* 41, 5 (2021), 15–23.
[4] Q-CTRL LTD. 2023. Black Opal: Learn quantum computing. https://q-ctrl.com/black-opal
[5] Francesco Maiorana. 2019. Interdisciplinary Computing for STE (A) M: a low Floor high ceiling curriculum. *Innovations, Technologies and Research in Education* 37 (2019).
[6] Stefano Mensa, Emre Sahin, George Williamson, and Robert J Allan. 2023. An Educational and Training Perspective on Integrating Hybrid Technologies with HPC Systems for Solving Real-World Commercial Problems. *Journal of Computational Science* 14, 1 (2023).
[7] John Rieman. 1996. A field study of exploratory learning strategies. *ACM Transactions on Computer-Human Interaction (TOCHI)* 3, 3 (1996), 189–218.
[8] Jim Samuel, Margaret Brennan-Tonetta, Yana Samuel, Pradeep Subedi, and Jack Smith. 2021. Strategies for democratization of supercomputing: Availability, accessibility and usability of high performance computing for education and practice of big data analytics. *arXiv preprint arXiv:2104.09091* (2021).
[9] Ethel Tshukudu and Quintin Cutts. 2020. Understanding conceptual transfer for students learning new programming languages. In *Proceedings of the 2020 ACM conference on international computing education research*. 227–237.
[10] James R Wootton, Francis Harkins, Nicholas T Bronn, Almudena Carrera Vazquez, Anna Phan, and Abraham T Asfaw. 2021. Teaching quantum computing with an interactive textbook. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 385–391.
[11] Esteban Zapirain, Stella Maris Massa, and Juan P Cardoso. 2019. Quantum Penny Flip: an open source serious game for quantum computing assessment. In *XIV Congreso Nacional de Tecnología en Educación y Educación en Tecnología (TE&ET 2019),(Universidad Nacional de San Luis, 1 y 2 de julio de 2019).*

[1] IQM Academy is available via https://www.iqmacademy.com.

Expanding Horizons: Advancing HPC Education in Colombia through CyberColombia's Summer Schools

Aurelio Vivas
Universidad de los Andes
Argonne National Laboratory
aa.vivas@uniandes.edu.co

Carlos E Alvarez
Tecnológico de Monterrey
carlose.alvarez@tec.mx

Jose M Monsalve Diaz
Argonne National Laboratory
jmonsalvediaz@anl.gov

Esteban Hernandez
CyberColombia
esteban.hernandez@cybercolombia.org

Juan G. Lalinde-Pulido
Universidad EAFIT
jlalinde@eafit.edu.co

Harold Castro
Universidad de los Andes
hcastro@uniandes.edu.co

ABSTRACT

High-performance computing (HPC) is an important tool for research, development, and the industry. Moreover, with the recent expansion of machine learning applications, the need for HPC is increasing even further. However, in developing countries with limited access to the HPC ecosystem, the lack of infrastructure, expertise, and access to knowledge represents a major obstacle to the expansion of HPC. Under these constraints, the adoption of HPC by communities presents several challenges. The HPC Summer Schools are an initiative of CyberColombia that has taken place over the past 5 years. It aims to develop the critical skills, strategic planning, and networking required to make available, disseminate, and maintain the knowledge of high-performance computing and its applications in Colombia. Here we report the results of this series of Summer Schools. The events have proven to be successful, with over 200 participants from more than 20 institutions. Participants span different levels of expertise, including undergraduate and graduate students as well as professionals. We also describe successful use cases for HPC cloud solutions, namely Chameleon Cloud.

1 INTRODUCTION

Supercomputers are of paramount importance in solving critical challenges in many fields. Some of these are atmospheric simulation, genome sequencing, and cybersecurity, to name a few [15]. In Colombia, this area is developing at a slower pace in comparison to leading countries such as the USA, Japan, and European countries. Colombia only spent 0.29% of its Gross domestic product (GDP) on Research and Development according to the UNESCO statistics [23]. This level of investment is 0.92% behind South American countries such as Brazil, which is one of the leading Latin American countries in supercomputing technology. Differences in investment levels on the development of supercomputers [10] and adoption of HPC are some of the factors that heavily influence this gap. On the other hand, Colombia is facing technological, economic, and educational challenges despite the efforts made by universities, and public and private institutions to promote the development of this area. However, initiatives that provide access to HPC resources such as SCALAC [12], RedCLARA [12], and HPC Americas Collaboration [9] still thrive.

Education in HPC is not only a critical concern in Colombia, but also in supercomputing leading countries [10, 19]. There is a gap between the speed at which new technologies are developed compared to the speed at which they are adopted. The HPC Summer Schools presented in this paper are an initiative of CyberColombia[1] [16], that aims to develop critical skills in HPC, strategic planning for HPC centers, and provide networking opportunities for the national community. We focus on these objectives as we consider them to increase the availability of knowledge of high-performance computing and its applications in Colombia. We also make a great effort in the establishment of collaborations and virtual organizations around common practices, tools, and data usage. Learning from the experience of already developed HPC communities worldwide, we emphasize multi-institutional collaborations, considering actors from industry, academia, and the international community.

The upcoming sections are organized as follows: Section 2 presents the main Colombian High-Performance Computing education and training event, the High-Performance Computing Summer School (HPCSS). Here we briefly describe (i) the main goals of this initiative, (ii) how the initiative has managed to persist during the pre-pandemic, pandemic, and post-pandemic periods, and (iii) what makes our initiative different to already existing ones. Section 3 describes (i) the timeline of events and decisions consolidating the HPCSS and statistics depicting the progression of the initiative. The latest version of the initiative, HPCSS 2023, which we consider the most successful, is detailed in Section 4. This work concludes in Section 5, where learned lessons and future directions are discussed.

2 SUMMER SCHOOLS

The High-Performance Computing Summer School (HPCSS) initiative was primarily motivated by already established non-local HPC education and training programs such as the Supercomputing Camp (SC-Camp) [22] and the Argonne Training Program on Extreme-scale Computing (ATPESC) [7]. Nevertheless, we have created a

https://doi.org/10.22369/issn.2153-4136/15/1/4

[1] https://cybercolombia.org

program that closely matches the country's current ecosystem and difficulties and it is tailored to resolve them. Over time, we have adopted methodologies, such as the multi-site and virtual delivered training, established on scientific environments like XSEDE [24] (nowadays ACCESS [6]) to enhance the effectiveness and reach of the Summer Schools program.

2.1 Objectives and Differences

Our summer schools have mainly two objectives: (i) bring awareness in Colombia of the broad array of research and career opportunities in areas supported by HPC, and (ii) promote cooperation between **academia**, **industry**, and **international entities**. In a nutshell, our program seeks to generate HPC expertise needed in Colombia [14] while broadening the community.

There are currently several HPC initiatives worldwide that target HPC education. The National Science Foundation in the United States of America created the Engineer Discovery Environment (XSEDE) [24]. This program ran from 2011 to 2022, and its focus was on sharing education and infrastructure for advanced services such as supercomputers. XSEDE offered a variety of online lectures and other training distributed across XSEDE sites. This program was then replaced in 2022 by the Advanced Cyberinfrastructure Coordination Ecosystem: Services and Support (ACCESS) [6], which currently has a similar idea, but it has been focused on increasing HPC resources and community development. Despite the success of these programs, they have been created mostly to tackle the needs of the community in the United States. Likewise, the US Department of Energy's leadership computing facilities (e.g., NERSC [4], ALCF [1], OLCF [18], etc.) run a series of training programs that include long training sessions (e.g., ATPESC). However, these programs are made to increase the knowledge of already existing users of their facilities, as well as encourage users with applications that are ready to be ported to these machines to apply for allocations. Unfortunately, these criteria create a really high bar for many researchers in Colombia who are just learning to use these systems.

Worldwide, there are also other supercomputing training programs. The International SuperComputing Camp [22], a non-profit event for training HPC, is held annually in different parts of the world. The International HPC Summer School [3], sponsored by different organizations around the world, has been an excellent option since 2010. More recently, the Latin American Introductory School on Parallel Programming and Parallel Architecture for High-Performance Computing organized by the International Center for Theoretical Physics (ICTP), focuses on parallel and distributed computing for scientists. Additionally, a large number of initiatives across multiple other countries and institutions [2, 5].

Despite the large number of options currently in the market, our program has a fundamental difference that makes it valuable. First, most of the already existing programs are created for those scientists who already have a need for HPC. Many of these participants have already been exposed to the idea of High-Performance Computing and are looking to expand their knowledge. As previously mentioned, the low investment in HPC in Colombia has resulted in limited exposure to the area of HPC to the large majority of scientists, students, and professionals. Furthermore, there is a lack of a strong HPC community in the country that does not have the necessary grounds to grow. Our focus is to tackle these problems. Our low entry bar allows many to participate.

Furthermore, we strongly encourage the interaction of members from different institutions. Thus, helping to increase the visibility and sense of community for these individuals, providing an experience that they would not have in their own institutions. The emphasis on networking aims to increase connections across institutions and individuals that could result in longer-term collaborations. In general, we see ourselves as facilitators of HPC in Colombia, and we expect that our participants can, in the future, take part in more advanced programs as those mentioned earlier.

2.2 Summer School Evolution

The annual HPCSS series program started in 2018. It represents the main high-performance computing informative and training event organized and delivered by CyberColombia. Our organization works in collaboration with local universities serving as hosts for the events and national (Colombian) and international institutions such as Universidad de los Andes (academia), Argonne National Laboratory (research), and NVIDIA (industry) participating as speakers or mentors.

The lockdown generated as a response to the SARS-CoV-2 pandemic posed great challenges to the development of activities of many organizations, and CyberColombia was not an exception. However, it also opened the opportunity to test communication technologies that we had not considered up to that moment and expand our international collaborations. For this reason, we will divide the development of the HPCSS into 3 consecutive periods of time: From 2018-2019 (pre-pandemic), from 2020-2022 (pandemic), and from 2023 onward (post-pandemic).

During the pre-pandemic period, the summer schools consisted of two main parts: (i) informative talks given by international speakers on various HPC-related topics, and (ii) practical workshops supervised by both professors from academic institutions and outreach staff from research institutions or industry. These first iterations of the summer school were 5-day events. Each day, we started with a keynote speaker who would help to motivate the materials presented during the day. Following, a series of hands-on tutorials were conducted. The topics of the tutorials were mostly introductory rather than driven by particular use cases. Among some of the topics that were covered are: introduction to HPC infrastructure, introduction to C and C++, introduction to job scheduling technologies (e.g., Torque, Slurm, PBS, etc.), introduction to parallel programming (e.g., OpenMP, OpenACC, Pthreads, etc.), introduction to accelerators programming (e.g., CUDA, OpenACC), and introduction to distributed Programming (e.g., MPI).

In the pandemic period, the events were switched from an in-person modality to a fully virtual experience. However, this transition did not come free of challenges when switching the talks and tutorials to a virtual environment. First, selecting a proper video conferencing software. Not only there were personal preferences in the attendees and speakers, but also different organizations and participating institutions had their own set of rules and preferences with respect to this matter. As a result, different sessions required us to switch from one videoconferencing software to another. Additionally, we also noticed that it was more difficult to keep the audience engaged in a virtual environment. Consequently, tutorials had lower interaction and participation from the attendees, diminishing the learning process. Another challenge brought by a virtual event was a reduction of networking opportunities between attendees. In-person sessions force participants to be face-to-face during several parts of the event. In contrast, virtual attendees do not require an active engagement, thus reducing the incentive to discuss ideas in a casual way or to engage in random conversations. Despite our efforts, most participants did not have the initiative to participate. Finally, virtual sessions proved to be more exhausting for participants and speakers. Maintaining the same position in front of a device considerably increased fatigue.

However, the virtual settings opened certain opportunities that were not possible before. The lack of a venue, catering, and travel expenses for speakers makes virtual meetings less costly to organize. For participants, it is also easier to attend, as the time and costs of traveling and accommodation are no longer present. These factors were reflected in lower registration fees, a factor that was particularly important for attendees with low-income levels in the country. As a result, the 2020, 2021, and 2022 versions of the HPCSS featured an increased level of international participation from both attendees and speakers/mentors. Furthermore, the scholarships, usually offered to a few participants who could not afford the registration, now were available for a larger number of people. Additionally, under normal circumstances, high-impact researchers and professionals would have a harder time scheduling a multi-day trip to the country. The lower time and personal commitment of speakers increased the chance to secure more impactful content. Finally, prior to the pandemic, we used different university venues that offered their facilities for our event. However, this reduced the visibility of CyberColombia as the main organizing partner. It also meant that other institutions would be more cautious about advertisement and participation. By removing dependencies on physical venues, CyberColombia, and our initiatives, were decentralized, allowing us to be a more neutral actor in the country and further increasing our collaborations with a wider range of institutions.

During the pandemic, the 5-day format was maintained, but the topics of the talks shifted from HPC core concepts to its applications. From this point on summer schools were more application oriented. We maintained the HPC focus, but, by considering a primary topic and a hot or emerging topic, we increased participation from domain-specific sectors. For the 2020 version, the main topic was HPC in Data Science and Artificial Intelligence, while the emerging topic was HPC against COVID. In 2021, a substantial amount of talks and tutorials were conducted in Convergence HPC, IA, and Big Data; but some of them were treated as emerging topics such as Quantum Computing. The 2022 version was concentrated on Space exploration and High-performance computing. While most of the tutorials during this period remained focused on the basics of HPC, new topics like the use of Matlab for HPC and quantum computing were also present.

Lastly, the post-pandemic era and the latest edition of the HPCSS was held in 2023. It will be addressed in more detail in Section 4.

3 SUMMER SCHOOLS IN NUMBERS

The progression of CyberColombia events over time has shaped what nowadays is called the HPCSS. The commitment to introduce Colombians to the world of HPC started in 2016 with a workshop held at Universidad Distrital de Colombia. In 2017, Universidad Distrital de Colombia welcomed students from multiple universities, fostering a diverse, inclusive, and cooperative learning environment. In 2018, the initiative was formally constituted as the HPCSS series, with the first edition taking place at Universidad de los Andes.

Also in 2017 the Earlham Institute, Colciencias, BRIDGE Colombia, and GROW Colombia launched the C3 Biodiversidad (C3) initiative; "aiming to promote a research cyberinfrastructure in Colombia for the analysis of the natural and agricultural biodiversity" [13]. In 2019, C3, Grow Colombia, and organizers of early versions of the Summer School series joined efforts to create CyberColombia. The result was a more formalized version of the HPCSS program that helped us tackle challenges in data-intensive science in Colombia from critical research areas including computational biology [16] [14]. From 2020, the HPCSS turned into a virtual event as explained in section 2. The following sections present statistics characterizing the estimated impact of the above-mentioned events in the progression of the HPCSS over time. Metrics on participation in attendance and mentoring, as well as some demographic data through the years, are described.

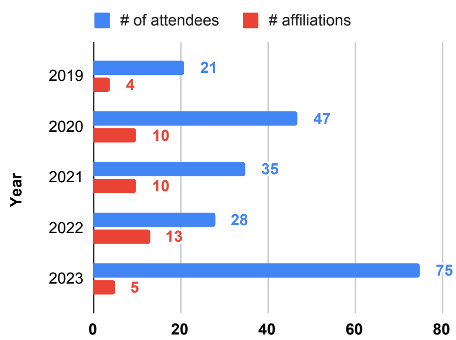
3.1 Participation Over Time

Figure 1 illustrates attendance and participation of speakers/mentors over time, the number of participating organizations (affiliations) in every case is also depicted. Note that in 2019, the event took place in its traditional, in-person format, whereas in 2020-2021 (pre-pandemic) and beyond (starting from 2022, post-pandemic), it transitioned to a virtual format.

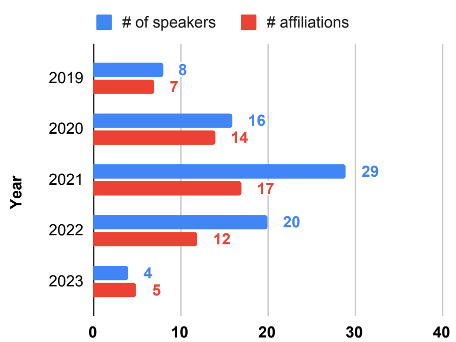
As seen in Figure 1a, the beginning of virtual experiences in 2020, encouraged attendance at a wide variety of events given the ease of access, savings in transportation times, and cost reductions. Nevertheless, the transition to the virtual experience had three associated drawbacks, impacting participation during periods 2021 and 2022. First, we lost the student-university interaction. Second, the large exhaustion of virtual meetings among participants discouraged people from taking part in these events. Third, our focus on specialized topics forced us to open new doors thus limiting our ability to reach out to a larger number of participants.

Despite receiving virtual support from the hosting university, challenges persisted due to students not being in an optimal learning environment and the inability to effectively gauge students' emotions and progress during talks and tutorials.

More so, the full-day week-long program turned out to be difficult for participants. We noticed a high dropout rate in the last days of training, especially during the 2021 and 2022 versions. We attributed the high desertion rate to the considerable burden of commitments acquired by the participants given the virtual experience and the psychological effects generated by the confinement situation during the pandemic. The above-mentioned situation led us to reformulate our program from one week to a three days comprised format and maintain the virtual experience for the latest version of the event, HPCSS 2023.



(a) Number of attendees over time



(b) Number of speakers over time

Figure 1: Participation over time

Figure 1b shows the number of speakers and speakers' affiliations. The variation across versions can be attributed to different factors. The virtual experience also encouraged speakers' and mentors' participation and promoted a larger diversity of academia, industry, and public and private institutions. Moreover, the change in focus area across the different versions increased or reduced the need for different speakers as well as our ability to recruit impactful ones.

Virtual events also provided a better opportunity to include more speakers and institutions, as mentioned in section 2. However, this virtual-only mode also significantly reduced the speakers-attendees interaction, finally reducing the impact of their participation in the attendee's professional development and networking opportunities.

We made significant progress on strengthening the speakers-attendees relationship on the last version of the event, HPCSS 2023 which is detailed in Section 4.

3.2 Participation Over Time per Sector

Tables 1 and 2 illustrate the number of HPCSS attendees and the number of institutions participating with speakers and/or mentors for tutorials. Note that both the total number of attendees and participating institutions over time are disaggregated across sectors. *Academia* includes universities; *research* includes research laboratories and institutions; *industry* includes national and multi-national companies; and *other* includes other public, private, and non-profit organizations.

Table 1: Attendance per Sector Overtime

Year\Sector	# academia	# research	# industry	# other
2019	14	-	-	7
2020	30	5	-	12
2021	21	-	1	13
2022	26	1	1	-
2023	75	-	-	-

Table 2: Institutions per Sector Overtime

Year\Sector	# academia	# research	# industry	# other
2019	6	-	1	-
2020	5	3	6	-
2021	5	4	7	1
2022	2	4	6	
2023	5	-	-	-

When comparing Tables 1 and 2, it is worth noting the difference in participation across sectors in attendance against institutions giving talks or mentoring tutorials. While institutions across sectors are more diverse, i.e., the attendance force is highly concentrated in academia.

Although our future goal is to make a more diverse attendance across sectors, academia is where we have found potential for the HPCSS program to make a meaningful and far-reaching impact. This is because academia is where future research and industry workforce develop fundamental technical and critical skills, in the realm of a learning environment, to face more specific, and complex problems in research laboratories and the industry.

Finally, the diversity in participating institutions has served two vital purposes: firstly, it has reaffirmed the sector's commitment to developing education in HPC in Colombia. Secondly, it has enabled us to keep the school's curriculum updated and relevant, incorporating cutting-edge tools and real-world applications.

3.3 Inclusivity

In order to incentivize the participation of underrepresented communities and guarantee inclusivity across underrepresented groups, we offer open-to-all scholarships for attendance on every version of the summer school. Acceptance was granted to participants (general public) who demonstrated in their applications how the summer school program would benefit their interest in career opportunities or strengthen ongoing research. We also consider the need to increase the participation of underrepresented groups in STEM, such as women in science communities, and low-income communities in the country. Although we slightly increased female attendance, the gap between men and women who participate in this kind of event is still high, as shown in Figure 2. Nevertheless, we understand that gender equality in science is a progressive transformation process.

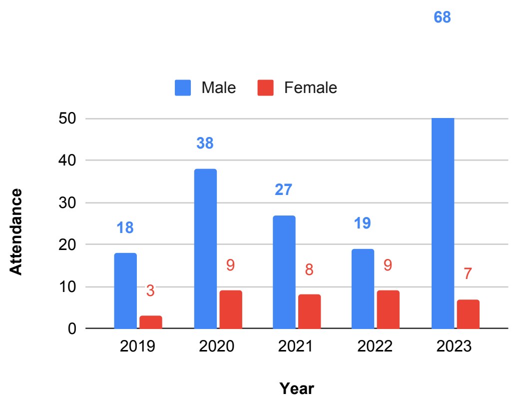


Figure 2: Attendance gender

3.4 Influence on Attendance and Collaboration

The Summer School has impacted more than 200 students in more than 20 institutions, as shown in Figure 3. This achievement owes itself to the collaborative effort of 27 organizations comprising universities, research institutions, and the industry, as described in Figure 4 and Table 3.

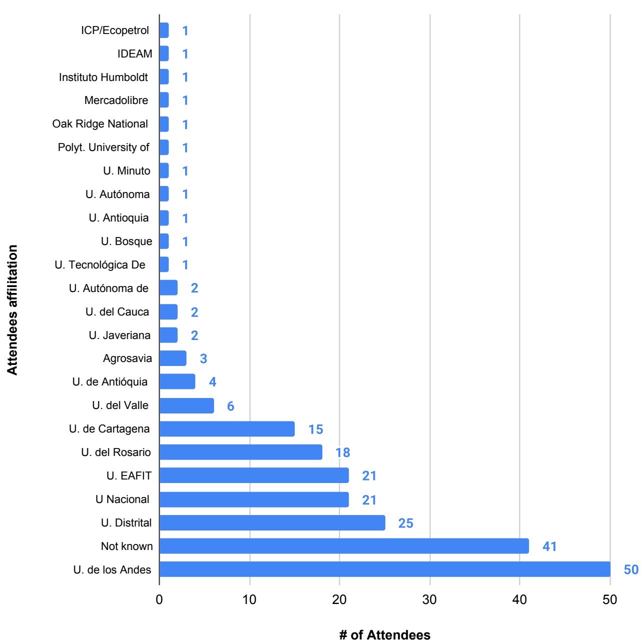


Figure 3: Number of attendees per institution (2019-2023)

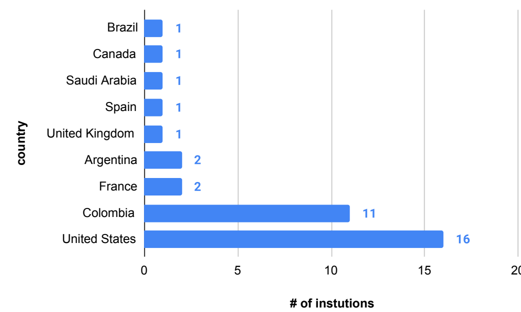


Figure 4: Number of Institutions Participating (with Speakers or Mentors) per Country (2019-2022)

Table 3: List of Institutions Participating (with Speakers or Mentors) per Country (2019-2022)

Institutions	Country
UNESP Center for Scientific Computing	Brazil
NVIDIA	United States
Universidad de los Andes	Colombia
Universidad del Valle	Colombia
Universidad Industrial de Santander	Colombia
KAUST Supercomputing Core Laboratory	Saudi Arabia
University of Delaware	United States
IBM	United States
University of Tennesse	United States
Pittsburgh SuperComputer Center	United States
AWS	United States
10x Genomics	United States
PSL	Colombia
Oak Ridge National Laboratory	United States
Argone National Laboratory	United States
University of Buenos Aires	Argentina
Loyola University	United States
MathWorks	United States
Argonne National Laboratory	United States
Coiled	United States
Universidad del Rosario	Colombia
National Aeronautics and Space Administration (NASA)	United States
Universidad Distrital	Colombia
Cybercolombia	Colombia
Renata	Colombia
Earlham	United Kingdom
Microsoft	United States
Barcelona Supercomputing Center	Spain
Universidad Industrial de Santader	Colombia
ATOS	France
McMaster University	Canada
iMMAP Colombia	Colombia
USGS Geologycal Survey	United States
UbiHPC	Colombia
European Space Agency (ESA)	France
DYMAXION LABS	Argentina

4 HPC SUMMER SCHOOL 2023: A DISTRIBUTED EVENT

As a result of our experiences before and after the SARS-CoV-2 lockdowns, we were able to use both remote and in-person tools to plan and carry out the HPCSS initiative in 2023. The possibility of creating a hybrid remote-in-person event offered a way to scale the initiative by including more actors, particularly universities. These multiple institutions were willing to act as decentralized venues for the event, connecting all of them remotely but maintaining the in-person character of the workshops within each venue. Similar approaches have been taken by XSEDE and related programs.

A big challenge for this modality is to procure centralized access to the hardware and software used to run the tutorials. Decentralization has the potential to reduce coherence and coordination, ultimately diminishing the experience for participants. After evaluating multiple options including in-house solutions, request allocations to leadership computing facilities, and cloud, we decided to use Chameleon Cloud [2].

Chameleon Cloud [17] is a *"large-scale, deeply reconfigurable experimental platform built to support Computer Sciences systems research"* [8], and it provides testbeds as instruments for research, as well as education. CyberColombia's HPCSS initiative, being an educational project, benefits greatly from the resources offered by Chameleon Cloud, providing the centralization of computing and software that the distributed modality requires.

In this section, we describe the methodology behind this last version of the Summer School. Here we explain the general organization, the program at a glance, supporting material, the challenges encountered, and how we managed to solve them.

4.1 Organization

Figure 5 provides a visual representation of the general structure of the Colombian HPCSS 2023. For this version, we partnered with five universities located in Colombia's primary metropolitan areas (Bogota, Medellin, and Cartagena). Faculty professors at these universities, referred to as *leaders*, were responsible for (1) encouraging participation in the event at their respective universities; (2) selecting students (up to 15) as well as a group of *volunteers* ranging from one to eight; and (3) providing a suitable space at their institution for students to receive the training and interact with each other.

CyberColombia, as the primary organizer, conducted two meetings with the leaders prior to the event. In the first meeting, the methodology and organization of the event were explained. In addition, individual sessions were scheduled to carry out network tests for the transmission (i.e., audio and visualization) and connection to the HPC platform. In the second meeting, the aforementioned tests were carried out in the rooms designated for the student's training. These technical tests enabled us to assess the transmission delay between universities and identify any connection issues from the universities to the HPC on Cloud infrastructure.

Furthermore, CyberColombia provided training materials and facilitated access to the HPC on Cloud platform in advance for volunteers, ensuring that these participants were well-informed about the content and adequately prepared to address minor issues and questions.

During the event, a team of four CyberColombia speakers and mentors, alongside two highly experienced volunteers, delivered both theoretical and hands-on content via Zoom, while also offering continuous support through Slack. Active support for students was presented by leaders of each institution and volunteers in the host sites along with the remote team. Professors at host sites not only reinforced the online-delivered content in person but also gauged students' comprehension and involvement with the material. During the event, they provided valuable feedback on what was the environment at each site, allowing us to adapt our progress accordingly. To say it simple, faculties and volunteers were our eyes on site.

[2]https://chameleoncloud.org/

Other responsibilities of each institution included: catering, collection of the registration fee, dealing with administrative procedures and permits, and generation of the necessary certificates.

4.2 Program at a glace

Table 4 depicts the HPCSS 2023 Daily schedule. The summer school program featured two discussion forums along with five introductory training that included both theory and hands-on tutorials.

Table 4: HPCSS 2023 - Daily schedule

TIME	DAY 1	DAY 2	DAY 3
8:00 - 900	Welcome	Colombia in HPC: Discussion Forum	Landscape of Supercomputing in Colombia: Discussion Forum
9:00 - 9:30		Coffee break	
9:30 - 12:30	Track 0 Parallel Computing Fundamentals	Track 2 Paralle Programmin with OpenMP	Track 4 Parallel and Distributed Programming with Python
12:30 - 13-30		Lunch	
13:30 - 16:30	Track 1 C++ 101	Track 3 Distributed Programming with MPI	Networking / Closing

4.3 Tutorial topics and material

The objective of this iteration of the summer school was to introduce the participants to the general concepts of parallel programming, as well as the basic hardware and software tools that are used. The tutorial is planned for 20 hours, divided into 3 days. The topics presented to the participants are divided into sections as follows:

Introduction to HPC. To motivate the whole program, the participants were first introduced to Moore's proposal for the growth of computational power in time and its limits. We described limitations posed by energy consumption, heat dissipation, and the physical limits of miniaturization of the transistor. The concepts of concurrent, parallel, and distributed computing were then introduced as solutions to the computational scaling problem. The concepts of process and thread were also presented here, along with common metrics used for assessing the success of parallel solutions such as speed-up, weak scaling, and strong scaling. This introductory section was then finalized with some examples and exercises using Python's threading module. We have chosen Python as it is a widely known language with a simple syntax, while the threading module allows the participants to manually create and destroy threads, consolidating the concept of the fork/join model as a basis for many parallel tasks.

Introduction to C++. The basic frameworks commonly used in HPC are OpenMP and MPI. However, before we can discuss them, we need to introduce C++, a language with solid and up-to-date implementations of these frameworks as well as a staple in the field of HPC. Many new programmers have not been introduced to this language. Thus, we used this tutorial section to introduce C++'s syntax and the basic structure of code. In particular, this section introduced the strong-typed nature of C++, the basic syntax for conditionals, loops, and functions, as well as the concepts of references and pointers. These latter are frequently used in memory copying operations in HPC, especially during accelerator programming.

Parallel computing with shared memory. In this section, the fork/join model was explored further. This section explained how to

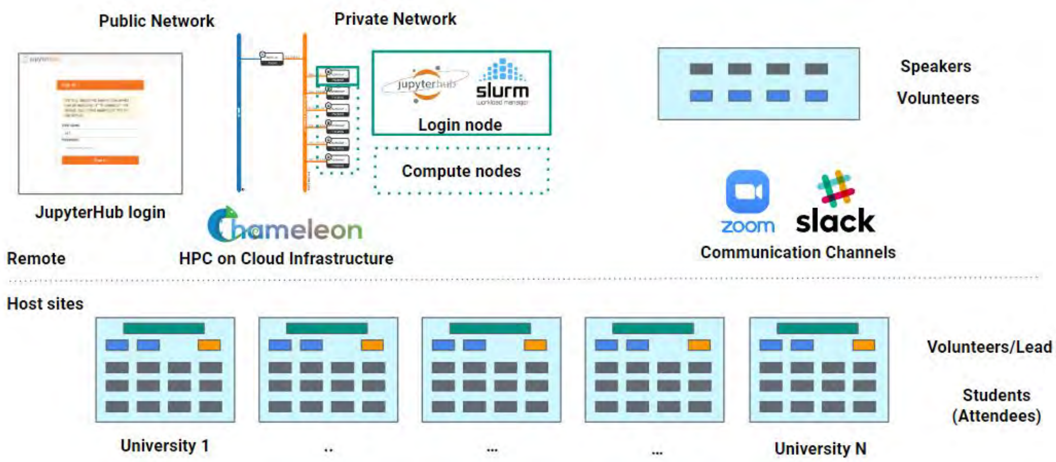


Figure 5: Colombain Organization of the High-Performance Computing Summer School 2023

implement parallel solutions in C++ using the OpenMP framework and deploy them on multi-core CPUs. The participants learned how to fork and join the flow of the program by creating threads with the #pragma omp parallel directive, as well as creating parallel loops and tasks. Several examples and exercises were provided to consolidate these concepts. The participants are encouraged to estimate the speedup of their parallel solutions. The tutorial used for this section is publicly available at [11].

Distributed computing. The concept of distributed computing was explored using C++ and MPI. The participants learned that to use several machines or nodes concurrently, a message protocol between nodes is needed. Using examples and hands-on exercises, participants learned how to initialize MPI to create the processes and perform point-to-point communications to send and receive messages between individual nodes.

Parallel computing with Python. Data science (DS) and artificial intelligence (AI) have gained great predominance. A great amount of the computational work in this field is done using Python. In this final module, the participants had the opportunity to work with Dask [21], a Python framework that allows the scaling of popular Python modules used for DS and AI. Dask performs the scaling by parallelizing the Python code, allowing the programmer to work with large amounts of data.

4.4 Technical Aspects

Teaching fundamental skills in parallel and distributed computing, the building blocks of High-Performance Computing, presents unique pedagogical challenges [19, 20]. These challenges extend beyond just delivering content to students; they also encompass the crucial task of ensuring access to the appropriate technological tools for a comprehensive and effective learning experience integrating both theory and practice.

When it comes to enabling the appropriate technological tools, the primary concern resolves around access to a flexible, configurable, and user-friendly HPC infrastructure. On this matter, our teaching infrastructure has shifted over different HPC solutions, providing considerable improvements every year.

Back in 2018, our main education infrastructure was based on an in-house university cluster. This required direct coordination and constant support from the facilities team. Challenges like account management, network restrictions, and unsupported software were common. Other solutions involved shared time in cloud computing services such as Amazon AWS and Google Cloud. However, the credits were often not enough. Access required additional steps for the users and the final result was not representative of leadership computing facilities worldwide (e.g., no batch scheduling).

Most recently, we deployed all our infrastructure using Chameleon Cloud [17]. This solution has proven to be the right middle point between resembling HPC ecosystems and having enough control over the configuration. Chameleon Cloud enabled us to allocate bare-metal HPC computing nodes located in leading HPC computing facilities such as Argonne National Laboratory and Texas Advanced Computing Center. The allocated instances are provisioned using ready-to-use Ansible playbooks, developed and maintained by CyberColombia for the HPCSSs. The provisioned software stack typically includes Slurm (workload management system), C/C++ and C-lang (compilers) with support for OpenMP, CUDA, OpenMPI, JupyterHub (for user-HPC interaction), and Jupyter notebooks (to deliver theoretical and practical content). Recently, we updated our software stack to include data analytics frameworks such as Dask; comprising interfaces like Dask-jobqueue, enabling the interaction of the framework with the workload management system.

Jupyter notebooks have garnered significant momentum as a powerful tool for interactive "human-in-the-loop" based research. However, care should be taken in the way they are used for teaching, particularly because the format does not resemble a genuine user-HPC system interaction. Nonetheless, for training and education, they substantially reduce the learning barriers.

As the HPC system is managed with the Slurm scheduler to provide isolation between different executions and deliver the performance expected in parallel and distributed applications, we did not write executable code directly into the notebook's cells. Instead, the notebooks provide the guidelines to edit pre-filled source files, then source files are compiled and sent to the scheduler through customized commands inserted on notebooks' cells. These commands provide interactive job submissions, timing, and memory and CPU usage tracking. This approach allowed us to maintain an interactive learning experience without delving too deeply into the intricacies of job submission. The primary goal of the summer school is for students to grasp the potential of parallel and distributed computing.

5 LESSONS LEARNED AND FUTURE DIRECTIONS

From the experience conducting HPC training events in Colombia, we concluded the following lessons learned: First, although virtual events stimulate participation in both speakers/mentors and attendees, this approach presents several drawbacks in the long term such as the lack of interaction among participants (student to student, students to mentors, and students to university), the lack of an appropriate learning environment, the lack of proper supervision of progress, among others. To overcome these challenges, we reintroduced in-person events while simultaneously sustaining remote content delivery, allowing us to expand our reach across multiple universities. Professors, on-site volunteers, and remote experienced volunteers played essential roles in providing students with a real-time, supportive learning experience. Secondly, in prior editions of the event, we observed that a significant number of participants lacked a fundamental understanding of HPC, leading to feelings of overwhelm and frustration when exposed to advanced materials and use cases that were beyond their current understanding. To address this concern, we reformulated our program into five tracks: Track 0. Parallel Computing Fundamentals, Track 1. C++ 101, Track 2. Parallel Programming with OpenMP, Track 3. Distributed Programming with MPI, and Tack 4. Parallel and Distributed Programming in Python. Finally, HPC in cloud solutions such as Chameleon Cloud played a paramount role in decentralizing our training efforts and amplifying our overall impact.

Future works will focus on tracking students' progress beyond the summer school and incentivizing their participation in the initiative, enabling them to contribute to the expansion of our programs across various locations and sectors, in different modalities (monthly talks, short tutorials, work-in-progress showcases).

ACKNOWLEDGEMENTS

We extend our gratitude to the numerous institutions, speakers, and mentors whose support has been instrumental in advancing CyberColombia's initiatives, along with all attendees who have contributed to our endeavors. Special appreciation goes to the Red Nacional Académica de Tecnología Avanzada (RENATA) for their invaluable partnership in the development of our latest version of the HPCSS. Furthermore, we express our thanks to Chameleon Cloud, backed by the National Science Foundation, for supporting our last two events. Finally, this research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy (DOE) Office of Science and the National Nuclear Security Administration. This research used resources from the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

REFERENCES

[1] Argonne Leadership Computing Facility . 2023. Training and Events | Argonne Leadership Computing Facility. https://www.alcf.anl.gov/events. (Accessed on 09/08/2023).
[2] Association for Computing Machinery (ACM) . 2023. HPC. https://europe.acm.org/seasonal-schools/hpc. (Accessed on 09/08/2023).
[3] IHPCSS . 2021. International HPC Summer School (IHPCSS). https://www.ihpcss.org/. (Accessed on 09/08/2023).
[4] National Energy Research Scientific Computing Center . 2023. NERSC Training Events. https://www.nersc.gov/users/training/events/. (Accessed on 09/08/2023).
[5] Partnership for Advanced Computing in Europe (PRACE). 2019. Training portal. https://training.prace-ri.eu/. (Accessed on 09/08/2023).
[6] ACCESS. 2023. ACCESS: Advancing Innovation. https://access-ci.org/. (Accessed on 09/08/2023).
[7] Argonne Training Program on Extreme-scale Computing. 2013. Argonne Training Program on Extreme-Scale Computing. https://extremecomputingtraining.anl.gov/. (Accessed on 09/06/2023).
[8] Chameleon Cloud. 2023. About Chameleon. https://www.chameleoncloud.org/about/chameleon/. (Accessed on 12/07/2023).
[9] AMERICAS HPC Collaboration. 2022. AMERICAS HPC Collaboration. https://americashpccollaboration.net/. (Accessed on 08/09/2022).
[10] ASC Community. 2018. *The Student Supercomputer Challenge Guide: From Supercomputing Competition to the Next HPC Generation*. Science Press.
[11] Jose M Monsalve Diaz CyberColombia. 2023. OpenMP Tutorial in Jupyter Notebooks. https://github.com/josemonsalve2/openmp_tutorial. (Accessed on 08/09/2023).
[12] SCALAC: Sistema de Computación Avanzada para Latino América y el Caribe. 2022. RedCLARA Home. https://scalac.redclara.net/index.php/es/. (Accessed on 08/09/2022).
[13] Jose De Vega, Monica C Munoz-Torres, Jorge Duitama, Narcis Fernandez-Fuentes, Graham J Etherington, and Robert P Davey. 2019. C3-Biodiversidad Consortium: A Cyberinfrastructure to Accelerate the Understanding and Preservation of the Colombian Biodiversity. In *Plant and Animal Genome XXVII Conference (January 12-16, 2019)*. PAG.
[14] Jose J De Vega, Robert P Davey, Jorge Duitama, Dairo Escobar, Marco A Cristancho-Ardila, Graham J Etherington, Alice Minotto, Nelson E Arenas-Suarez, Juan D Pineda-Cardenas, Javier Correa-Alvarez, et al. 2020. Colombia's cyberinfrastructure for biodiversity: Building data infrastructure in emerging countries to foster socioeconomic growth. *Plants, People, Planet* 2, 3 (2020), 229–236.
[15] S.L. Graham, M. Snir, and C.A. Patterson (Eds.). 2005. *Getting Up to Speed: The Future of Supercomputing*. The National Academies Press.
[16] E. Hernández, C.E. Álvarez, C.A. Varela, J.P. Mallarino, and J. De Vega. 2021. CyberColombia: a Regional Initiative to Teach HPC and Computational Sciences. *ACI Avances en Ciencias e Ingenierías* 13, 2 (2021), 9.
[17] K. Keahey, J. Anderson, Z. Zhen, P. Riteau, P. Ruth, D. Stanzione, M. Cevik, J. Colleran, H. Gunawi, C. Hammock, J. Mambretti, A. Barnes, F. Halbach, A. Rocha, and J. Stubbs. 2020. Lessons learned from the Chameleon testbed. In *Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC '20)*. USENIX Association, Berkley, CA, USA, 219–233. https://doi.org/10.5555/3489146.3489161
[18] Oak Ridge Leadership Computing Facility . 2023. Training – Oak Ridge Leadership Computing Facility. https://www.olcf.ornl.gov/for-users/training/. (Accessed on 09/08/2023).
[19] 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, et al. 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*. 51–74.
[20] Diego A. Roa Perdomo, Paige C. Kinsley, Jose M. Monsalve Diaz, and Michael E. Papka. 2023. DEMAC – A Platform for Education in High-performance Computing, Bridging the Gap Between Users and Hardware. (2023). [Manuscript submitted for publication].
[21] Matthew Rocklin et al. 2015. Dask: Parallel computation with blocked algorithms and task scheduling. In *Proceedings of the 14th python in science conference*, Vol. 130. SciPy Austin, TX, 136.
[22] Supercomputing Camp (SC-camp). 2023. Supercomputing Camp 2023. https://sc-camp.org/2023/index.html. (Accessed on 09/06/2023).
[23] UNESCO Institute for Statistics. 2022. Research and development expenditure. https://datos.bancomundial.org/indicator/GB.XPD.RSDV.GD.ZS. (Accessed on 08/09/2022).
[24] XSEDE. 2022. XSEDE: Extreme Science and Engineering Discovery Environment. https://www.xsede.org/. (Accessed on 09/08/2023).

BEAST Lab: A Practical Course on Experimental Evaluation of Diverse Modern HPC Architectures and Accelerators

Amir Raoofy
Technical University of Munich
Leibniz Supercomputing Centre
amir.raoofy@lrz.de

Bengisu Elis
Technical University of Munich
bengisu.elis@tum.de

Vincent Bode
Technical University of Munich
vincent.bode@tum.de

Minh Thanh Chung
Ludwig Maximilian University of Munich
minh.thanh.chung@ifi.lmu.de

Sergej Breiter
Ludwig Maximilian University of Munich
sergej.breiter@nm.ifi.lmu.de

Maron Schlemon
German Aerospace Center
maron.schlemon@dlr.de

Dennis-Florian Herr
Technical University of Munich
deflo.herr@tum.de

Karl Fuerlinger
Ludwig Maximilian University of Munich
karl.fuerlinger@ifi.lmu.de

Martin Schulz
Technical University of Munich
Leibniz Supercomputing Centre
schulzm@in.tum.de

Josef Weidendorfer
Technical University of Munich
Leibniz Supercomputing Centre
josef.weidendorfer@lrz.de

ABSTRACT

Giving students a good understanding how micro-architectural effects impact achievable performance of HPC workloads is essential for their education. It enables them to find effective optimization strategies and to reason about sensible approaches towards better efficiency. This paper describes a lab course held in collaboration between LRZ, LMU, and TUM. The course was born with a dual motivation in mind: filling a gap in educating students to become HPC experts, as well as understanding the stability and usability of emerging HPC programming models for recent CPU and GPU architectures with the help of students. We describe the course structure used to achieve these goals, resources made available to attract students, and experiences and statistics from running the course for six semesters. We conclude with an assessment of how successfully the lab course met the initially set vision.

KEYWORDS

High Performance Computing, Computer Architecture, Accelerator Architectures, Computer Science Education

1 INTRODUCTION

As Tier-0 compute centers in Europe move towards exascale, the education of students and users alike is gaining significant importance in order to be prepared to utilize the potential of these systems. Various valuable training materials and courses have been developed, including broad community training initiatives in the US (e.g., efforts in ECP [13, 14]) and Europe (e.g., activities in PRACE [8] and European Union's Horizon 2020 research and innovation program). These programs mainly cover mainstream HPC programming models, frequently target specific domains (e.g., ModSim or AI) for researchers beyond computer science, and often focus on platform-specific tools for better compute facility usage. A more general education for computer scientists from the architectural point of view is missing, which is, despite recent training methods and approaches, often lacking even in advanced university curricula. Those often stay either at the theoretical level, or focus only on high-level programming aspects of HPC systems.

Based on our experience working with HPC end-users in our compute center, we consider getting practical experience, in particular with focus on the micro-architecture of modern HPC platforms and programming models, a key aspect in HPC education, especially for computer science students and future HPC experts. This also must include architectural details, like differing processing elements and complex memory hierarchies.

In 2020, we introduced a new *practical lab* course focusing on the experimental evaluation of HPC architectures, called the BEAST Lab, after the BEAST testbed at LRZ which is used in the lab (BEAST is short for Bavarian Energy, Architecture, and Software Testbed). It is primarily targeted at Computer Science and Computer Engineering (CS and CE) students and has a strong focus on micro-architecture of modern HPC architectures and systems.

The BEAST Lab shares the spirit of the recent training methods and programs [4, 5, 9, 12] in providing resource access to students. Our primary teaching approach and educational strategy is to enable students to acquire a deep understanding of micro-architectural

https://doi.org/10.22369/issn.2153-4136/15/1/5

aspects of HPC platforms through practical experience on modern HPC platforms and experimental evaluation of their micro-architectural properties. To reach this goal, BEAST[1], the *Bavarian Energy, Architecture, and Software Testbed*, which is the LRZ test environment with the latest computer technologies available, enabling research and exploration of new computer technologies and (HPC) systems, plays a central role. In addition, the BEAST Lab aims to have a broader impact by enhancing collaborative research, utilizing specialized hardware, and involving students in HPC hardware evaluation and programming, through a multi-institute structure organized by LRZ, thereby complementing lectures and seminars at co-organizing universities, the Technical University of Munich and the Ludwig Maximilian University of Munich.

The BEAST Lab uses a multitude of diverse hardware platforms available in BEAST with the intention of enabling insights by highlighting the architectural differences as well as the differences in programming models designed to target this diverse hardware in practice. Therefore, the course is structured with practical assignments and projects requiring minimal implementation effort, but focusing on experimental evaluations, that teach how to conduct detailed performance analysis and interpret them. This enables students to explore and understand the architectural differences through their measurements. The assignments cover programming experiments on compute-bound and memory-bound kernels on CPUs and GPUs using common HPC node-level programming models (OpenMP, CUDA, and HIP), instruction-level and memory parallelism, branch prediction, and NUMA effects on CPUs.

The projects cover the same type of topics and experiments but need more effort in programming, enabling multi-threading and various optimizations such as low-level manual vectorization. These projects target a selection of conventional and fairly new applications such as multigrid solver [20], matrix profile computation [17], and interpolation kernels [18].

Finally, to assess the success of the lab course and to ensure that the lab material stays aligned with the teaching goals, we establish a survey for students, where the participants are asked to evaluate whether the goals are met. This also helps the instructors to gradually improve the course over time. In addition to quality assurance, the surveys serve as feedback from students on their experience on various hardware, software, tools and programming models. We collected these surveys over four semesters and discussed the results in the paper, e.g., that students voted OpenMP offloading the most intuitive and easy-to-learn model for GPU programming.

Overall in this paper, we make the following contributions:

- We discuss the benefits of the BEAST Lab from the perspective of organizing parties as well as students.
- We describe the teaching approach, including the course structure, resources made available to students, assignments, projects, and grading scheme.
- We summarize lessons learned from organizing the BEAST Lab for six semesters and provide insights from student surveys on experiences with various hardware and programming models collected over four semesters.

[1]BEAST is a wordplay: the more systems are coming in, the more the testbed environment looks like a beast to tame to get results. The corresponding logo shows a lion head as beast, relating to the lions in the Bavarian coat of arms (cf. Figure 1).

The survey shows that in most cases the students could explain measurements on the specific architectures and that overall the BEAST Lab helped the majority of students to properly evaluate and better understand modern HPC architectures, making this a highly successful lab course that has become a permanent offering.

2 MOTIVATION

The BEAST Lab was incepted with two primary goals in mind: filling a practical gap in educating students to become HPC experts, as well as acquiring experience and understanding the stability and usability of emerging HPC programming models for recent CPU and GPU architectures with the help of students. In addition to these aspects, the BEAST Lab is also designed with a broader impact in mind to strengthen the collaborative research between academic institutes and the Leibniz supercomputing center in Munich.

Motivation for the Compute Center: Evaluation of modern HPC hardware technologies and emerging HPC programming models is a critical research area for the LRZ, especially in the era of the Cambrian explosion of computing and novel architectures [15]. This research helps LRZ to acquire practical experience on the various modern systems, technologies, and approachability of programming models as well as the maturity of software and toolchains, and gain expertise to provide users with suggestions, recommendations, and best practices on how to write sustainable parallel codes for future generation machines in computing centers. However, the extensive variety of modern architectures requires allocating lots of internal resources. The BEAST Lab helps to attract, educate, and engage new students in this research area (similar to [22]). We received more than 20 inquiries from students to continue research after participating in the BEAST Lab. Additionally, the BEAST Lab helps motivate research teams and PhD candidates to engage in the evaluation of modern HPC architectures. 30 researchers (mainly PhD candidates) from the co-organizing universities are conducting research on BEAST; 5 of them actively contribute to the development of the BEAST Lab.

Motivation for Educators: The CS and CE study programs at TUM and LMU provide lectures around computer architecture both on a basic level in the first year of Bachelor as well as on a deeper level as selective courses in Master tracks. The latter discusses concepts, approaches, and design choices of the micro-architecture, mostly covering CPUs on a theoretical level, such as pipelining, out-of-order execution, or the memory hierarchy. The BEAST Lab helps to fill the practical gap in CS students' education regarding complex concepts and intricacies in the micro-architecture of modern HPC processors and accelerators. It drives understanding to observe the performance differences of code variants in practice, backed by discussion on the micro-architectural properties responsible for the observed effects. Furthermore, the variety of architectures available in BEAST allows for insightful comparisons, highlighting different design choices.

Motivation for Students: For students, the lab course provides the following benefits: 1) access to rather expensive, modern, and often rare HPC hardware, 2) gaining practical hands-on experience on these modern systems, and 3) engaging in state-of-the-art HPC hardware and system software research after gaining experience in the BEAST Lab. Additionally, the BEAST Lab also features invited talks by mainstream HPC vendors through which students can hear a light vendor roadmap and an overview of cutting-edge hardware and software technologies. We also provide a tour of the LRZ site, including a visit to the infrastructure and the compute cube of SuperMUC-NG, offering the students a unique chance to visit a top-tier HPC system.

3 AVAILABLE INFRASTRUCTURE FOR LAB

To stay up to date with new architectures and components, to contribute to the research in shaping the future of HPC systems, as well as to educate future HPC experts, the LRZ has established the "Future Computing" program funded by the Bavarian State Government: the cornerstone of this program is the development of test environments with the latest computer technologies, the "Bavarian Energy, Architecture and Software Testbed" or BEAST for short (Figure 1).

The "Future Computing" Program: LRZ serves academic researchers from Bavaria and Germany, mostly running scientific simulation codes from natural sciences. The main goal of the "Future Computing" program is to match available future technologies with the demands of LRZ users, starting with a set of benchmarks that reflect the typical application mix running on current systems. These benchmarks are then evaluated on the various architectures in the BEAST system to get an understanding of future technology and to answer questions like, "Which systems provide the best acceleration and efficiency for user codes considering a given budget?". The resulting insights are of significant help for the procurement of upcoming systems. Another important goal is to check the claims of vendors about the stability and versatility of the provided software stack, including support for parallel programming models. Finally, BEAST helps enable the porting/adaptation of in-house system tools for new architectures.

Figure 1: BEAST Testbed system at LRZ.

3.1 BEAST Hardware

Our lab course uses BEAST as the main education resource and provides the students with access to the latest HPC processors and accelerators. Through the access and working on BEAST testbed, students get a chance to work on processors and accelerators with similar architecture and capabilities to the top systems in Top500 list [6], including Frontier, Fugaku, LUMI, and Leonardo. The BEAST testbed offers a variety of architectures from different vendors and exposes different instruction sets. However, in the lab course, the following systems are used intensively.

Intel Icelake + Volta 100: Each node of this type features a two-socket Intel Xeon Platinum 8360Y CPUs with an Icelake micro-architecture. 72 cores are distributed over the two sockets (in two NUMA domains), where each is equipped with one NVIDIA Tesla V100 GPU (to be moved to A100 GPUs in the upcoming semester). We are providing access to two nodes of this type to our students.

ARM Thunder X2 + Volta 100: The second system type consists of two nodes with Thunder X2 CN9980 ARM CPUs. Each has two sockets with 32 cores per socket and four-way hyperthreading. Two V100 NVIDIA GPUs are connected to each node.

AMD Rome + AMD MI100: The third system type features two nodes with AMD EPYC 7742 Rome CPUs of Zen2 micro-architecture, each containing 64 cores distributed on two sockets. Each node is equipped with two AMD MI100 GPUs (MI50 GPUs were used in the early stages of the lab course). In the upcoming semesters, we are moving towards using AMD Milan systems with the Zen3 micro-architecture and AMD MI200 GPUs.

Fujitsu A64FX:. The fourth system features Fujitsu A64FX nodes with 64-bit Arm architecture with 512-bit vector implementation of ARMv8 SVE SIMD instruction set extensions. Each node has 48 cores distributed on 4 NUMA domains and a total of 32 GB HBM memory. There is no simultaneous multi-threading support.

3.2 BEAST Software Environment

BEAST offers a similar environment, such as the production systems of LRZ, running SLES and a SLURM instance (the scheduler is, however, not currently used in the lab as we allocate dedicated machines). However, as BEAST is part of the LRZ research infrastructure, many researchers from Munich universities and internal LRZ users actively use the infrastructure, and consequently, the software environment and configuration are subject to change based on the needs of the researchers.

For this reason, we use Spack [11] to maintain a dedicated software stack for the Lab course (of course, this stack can be reused for other purposes as well). For instance, various compilers with different backends, as well as tools (e.g., LIKWID), are installed using Spack. Other system software and tools, such as device drivers, vendor compilers, and runtime libraries, are installed system-wide and regularly updated. We also maintain a system-wide instance of Data Center Data Base (DCDB) [16] to monitor the outbound power consumption of nodes gathered from PDUs, which is used in parts of the lab for experiments on energy efficiency.

4 THE LAB STRUCTURE AND TEACHING APPROACH

Focus and Topic Coverage: BEAST Lab gives students opportunities to work on modern computing architectures. In particular, the course helps students understand how micro-architecture technologies affect applications' performance. The involved assignments and projects in BEAST Lab cover various aspects, such as thread-level parallelism, instruction-level parallelism, vectorization, memory access patterns, caches-memory hierarchy, NUMA effects, branch prediction, and power consumption. These are integrated through the general and experimental questions of assignments. Each assignment is linked to a micro-benchmark referring to the aspects that can impact overall performance. Students can perform experiments on CPU and GPU architectures.

Teaching Approach: Our teaching approach is primarily based on comparing architectural effects among different architectures. Students are assigned to work in a group to complete the assignments and projects. In each assignment, the groups work on a specific topic and conduct experiments on BEAST hardware. Before conducting experiments, they may have a conjecture about the results. After experiments, each group of students summarizes the results in a report and analyzes the performance evaluation of different architectures. Students are expected to understand architectural knobs, tuning, and optimization potentials affecting performance. We do not require students to put extra effort into programming from scratch. Instead, we provide the reference codes for the tasks and instructions for conducting experiments on BEAST platforms. With this approach, most of the time can be spent on investigating different optimization ideas and analyzing performance results.

Another aspect to consider for reducing the programming effort on the student side is the programming models. These models should make it easy to port from a sequential program to a parallel version and port to different platforms and architectures in BEAST. Therefore, we rely on OpenMP as the primary programming model for teaching as it is almost the only vendor-neutral option supporting all the deployment and acceleration in all BEAST platforms, including heterogeneous programming on GPUs and CPUs. Other advanced and specialized programming models, such as CUDA, or HIP, are also introduced to students. However, we consider these as bonuses if students try to make an effort.

Lecture Organization & Assignment Structure. The lab is structured into two main parts: In Part 1, students hear lectures on micro-architectural topics and work on weekly assignments. In Part 2, they work on projects with more complex coding tasks. Part 1 is organized with the weekly lab sessions as follows: sessions begin with a short lecture on a specific topic related to the upcoming assignment. This is followed by student presentations covering their experience and results from the previous assignment. In the end, we introduce the next assignment. In Part 2, the sessions mainly include an introduction to the project, open group progress discussions, and project presentations.

In the first lecture, we introduce the course organization and provide supporting materials to the students. This includes an overview of the available hardware and their architecture, software environment, and compilers in BEAST. Further weekly lectures cover an introduction to OpenMP, GPU architecture and OpenMP target offloading, memory hierarchy of multi-core CPUs, pipelining and branch prediction, and tools for tracing and profiling.

Overview of assignments: Following the first introductory lecture, BEAST LAB begins with a warm-up assignment providing instructions about accessing the system remotely and loading related libraries and compilers to familiarize students with the usage of our platform. After the warm-up assignment, seven assignments cover the mentioned topics. Each assignment is divided into smaller

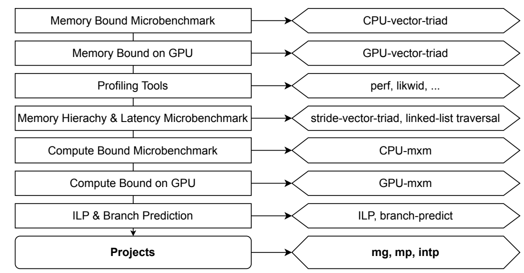


Figure 2: Overview of assignments & projects in BEAST Lab.

parts with specific tasks and questions, supposed to incrementally improve the understanding of the topics. Students are asked to explain the performance of certain codes and modifications, supported by appropriate visualizations. The assignment overview and order are shown in Figure 2. Their content is described as follows.

(1) **Vector Triad on CPU.** This assignment uses a vector triad to show how the available bandwidth of different levels in the memory hierarchy affects performance in memory-bound applications in dependence on their arithmetic intensity. The micro-benchmark is analyzed in sequential and parallel execution. Topics covered include compiler auto-vectorization, caching, thread pinning and OpenMP loop scheduling effects, NUMA effects and first touch policy.
(2) **Vector Triad on GPU.** Assignment 2 introduces students to OpenMP offloading in the context of GPUs. The tasks require analysis and evaluation of the GPU architecture with a focus on the memory hierarchy, i.e., the OpenMP thread and team scheduling on GPU processing elements, as well as the importance of memory coalescing and data transfers between host and device, which are explored.
(3) **Profiling Tools for CPU and GPU.** This assignment introduces students to various profiling tools for CPUs (Perf, Likwid, PAPI) and GPUs (nsys, ncu, rocprof, THAPI). Students learn to identify performance bottlenecks using code instrumentation and tools to measure performance events.
(4) **Memory Hierarchy and Memory Access Latency.** We use vector triad (a stride variant), a linked-list traversal, and a ping-pong code to introduce detailed performance aspects of the memory hierarchy and parallelism, such as prefetching, cache line efficiency, NUMA effects, cache coherence, and core-to-core latency.
(5) **Matrix Multiplication on CPU.** A dense matrix multiplication is used as a compute-bound micro-benchmark. We focus on memory access patterns, loop scheduling with OpenMP parallelization, cache blocking, (auto-)vectorization, and compiler optimizations. Additionally, we encourage the students to use roofline model for each architecture to further analyze their experiments on BEAST.
(6) **Matrix Multiplication on GPU.** Assignment 6 covers performance aspects of data transfers between host and device, matrix multiplication cache blocking and thread scheduling on GPUs, and introduces the CUDA programming model. Besides, students can measure energy efficiency using DCDB.

(7) **Pipelining and Branch Prediction.** This assignment introduces codes to measure properties of key aspects in modern pipelined microprocessor architectures, such as instruction latency, throughput, and branch prediction. Branch prediction capacities are analyzed via different types, i.e., conditional jumps, indirect jumps, and returns.

Final Projects: At the end of the lab course, student groups work on projects that put their knowledge and experience to the test. These projects are structured similarly to the weekly assignments but include more implementation workload around the topics discussed in the lab. Groups bid on the available platforms for their project and further implement specified applications targeting CPUs or GPUs using OpenMP, and optionally CUDA and HIP. They further and evaluate their implementations according to the provided instructions and report the results in a final presentation.

The content of these projects is described as follows:

(1) Multigrid is a well-established iterative approach for solving linear equations mainly associated with partial differential equations. This approach leverages a hierarchy of grids to accelerate the convergence of the iterative solver. In this project, students are provided with a sequential implementation of a full multi-grid solver that is based on Jacobi smoother, and it includes restriction and prolongation steps, as well as the typical so-called V and W cycle schemes. Groups implement vectorization, parallelization, and GPU offloading for target codes and investigate performance bottlenecks and efficiency.
(2) Automatic optimization and vectorization enabled by OpenMP directives or compiler flags may not improve performance in complex cases with complex access patterns. In such cases, manual implementation of SIMD is often required, potentially necessitating restructuring the algorithms to tap into parallel processing capabilities. In this project, students extend the implementation of a sinc interpolation function [18] with OpenMP as well as manual SIMD and evaluate their solutions on several CPU platforms in BEAST.
(3) In the last project, students work on a time series mining application, specifically matrix profile computation [17]. We provide students with the source code for the serial implementation of matrix profile computation. Students implement it by enabling efficient multi-threading, vectorization, and offloading and further study it in their selected target platforms in BEAST.

Teaching Support Structure, Student Feedback, and Evaluation Scheme: In terms of teaching support and student evaluation, we require students to work in groups with Git for their collaboration, assignment/project submission. For the discussion among students' groups, a common Zulip channel is created. Tutors also get involved in this channel to support students. Depending on the content of each assignment and project, deadlines can be in one or two weeks. After each assignment/project, a few groups are selected to present the reports of their performance results. Following that, we can question and discuss in detail how students understand the results and performance aspects corresponding to each micro-architecture.

5 CHALLENGES

Organizing the practical course with the above-mentioned structure comes with various organizational and technical challenges. These challenges include:

(1) Structuring suitable assignments and projects that are aligned with the primary goals of the lab course,
(2) Regular cross-institution coordination and planning,
(3) Coping with the gradual changes in hardware and software platforms, which requires additional effort in keeping a dedicated software environment for the assignments and projects to rely upon while also maintaining an ever-changing research software stack and environment,
(4) Better usage of scheduler for student jobs (overlapping allocations).

6 EVALUATIONS OF BEAST LAB SUCCESS

Over the course of four semesters between 2021 and 2023, we have asked students to evaluate their experiences in the BEAST Lab. While the two universities conduct generic course evaluations, we provided a special feedback form in addition that focuses on the syllabus and learning aspects specific to the BEAST Lab. The types of questions include rating on a scale, selecting from categories, and providing free-text answers. In total, we have received 60 responses, of which 12 are from students at the bachelor's level (all enrolled in computer science) and 48 at the master's level (in computer science and adjacent disciplines, such as computational science & engineering, robotics, cognition & intelligence, or games engineering).

6.1 Student Experience

On average, our students report having 4.2 years of programming experience, which is slightly above their average length of enrollment. Many students have previously taken courses in the subject field, with a majority (58%) having attended the lectures *Parallel Programming or Programming of Supercomputers*, 22% having attended the lecture *Advanced Computer Architecture*, 8% having attended a lab related to High Performance Computing or GPUs, and 12% having attended no prerequisite courses at all. Despite this, many students report having only little experience with both OpenMP and OpenMP Offloading. On a 1-5 scale (5 very experienced and 1 novice), the average student self-reports an experience of 2.4 for OpenMP and an experience of 1.3 for OpenMP Offloading, with 38% and 79% of students having had no respective prior experience at all, raising the question of how effective the adjacent courses teach OpenMP as a parallel programming technique.

6.2 Merits of OpenMP

OpenMP is by far the most popular parallelization technology in the lab, with almost two-thirds of students preferring its directive-based approach to a computing-kernel paradigm (25%) such as the one found in CUDA. Coming from a background of comparatively little experience, students find that OpenMP is reasonably easy to learn rated as 3.5/5, with OpenMP Offloading being somewhat more difficult at 2.8/5 on a scale of 1 very easy 5 very hard. This is likely due to the additional requirement of having to handle data movement. OpenMP loop constructs were rated as the most intuitive API for parallelization on CPU, followed by intrinsics

Table 1: Students' Opinions of Implementing Parallel Code on CPU and GPU on a 1-5 Scale.

	Debugging	Profiling	Difference
CPU	2.7	3.0	**0.3**
GPU	1.9	2.4	**0.5**
Difference	**-0.8**	**-0.6**	

Note: Generally, students prefer pro-filing over debugging and working on CPU over working on GPU.

before OpenMP SIMD. On GPUs, twice as many students preferred writing OpenMP Offloading code overwriting CUDA code, with barely any students preferring HIP. Figure 3 shows the top reasons that students like OpenMP and the most often cited criticisms. As expected, the ease of implementation and the learning curve are seen as big pluses for using OpenMP for on-node parallelization. However, students often find it difficult to understand what their code or the OpenMP library is doing at run time, which is reinforced by the observation that students also find it difficult to debug their parallel programs. To this day, despite the maturity of OpenMP and shared memory parallelism in general, these still appear to be common pain points for parallel program developers.

6.3 CPU and GPU

As part of the lab course, students work with both CPU and GPU technology. Unsurprisingly, students preferred working on CPU rather than GPU (with GPUs scoring around 20% lower in the ease of debugging criterion and 15% lower in the ease of profiling criterion) and found that finding bottlenecks was generally easier than debugging a program (Table 1). With a rating of only 1.9/5, debugging on GPU was by far the most painful for students. This is likely due to the fact that GPU debugging tools are not nearly as prevalent as their CPU counterparts. The easiest task was profiling on the CPU, but with a rating of 3.0/5 it can hardly be said that this is a seamless experience.

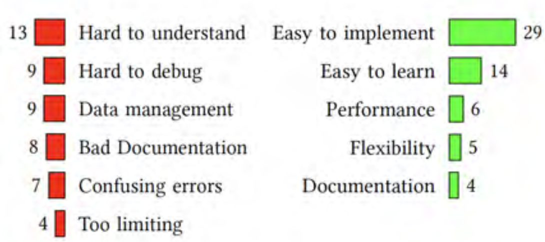


Figure 3: Advantages and disadvantages of using OpenMP according to student opinion. The biggest appeals of using OpenMP for students are how easy and fast it is to parallelize code and the learning curve. However, students often find it difficult to understand what their code or the OpenMP run-time is doing.

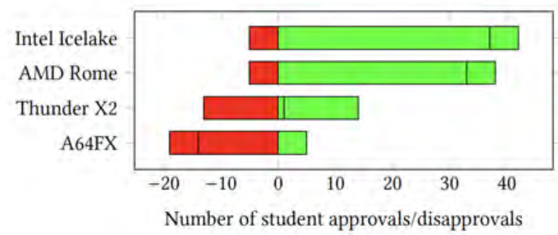


Figure 4: Architecture popularity among students. The x86-64-based systems (Icelake and Rome) outclass the ARM-based systems (A64FX and X2), with the A64FX, in particular, being the least popular. being the least popular.

6.4 Profilers

Students typically spend around 54% of their time parallelizing their application until they reach their first correct implementation (which includes debugging), and around 42% of their time in optimization (including profiling). In general, students find profilers reasonably useful when attempting to optimize code (3.5/5). The most popular profiler was Linux's Perf (3.4/5), with Likwid (3.1/5) and the GPU profilers (2.9/5) following suit.

6.5 Architectures

Throughout the lab, students work with four types of machines available in the BEAST cluster. One of the motivations for the LRZ to conduct the lab is to learn about the merits of different systems from a user's perspective, with students serving as stand-ins for actual scientific users. According to Figure 4, the students have a very clear opinion on which systems they like. The two x86-64-based systems, Intel Icelake and AMD Rome, come out far ahead of the ARM-based systems, Thunder X2 and the A64FX. Icelake wins out slightly over Rome, though this may be attributed to the system being slightly more recent, but both systems have a good approval rating of 62% and 55% of participants, respectively. The opinion on the Thunder X2 machines is neutral, with approximately equal counts of approval and disapproval leading to an approval rating

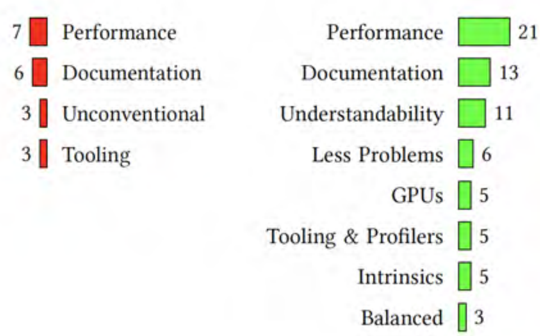


Figure 5: Top reasons why students like or dislike architectures found in BEAST. The most important factors for students are performance and available documentation.

of 2%. The biggest loser in terms of student opinion is clearly the A64FX, with a final approval rating of -23%, which is particularly unfortunate because the BEAST cluster has more A64FX machines available than any other type, which means crowding should have been less of an issue.

Figure 5 gives us clues as to why the approval ratings are the way they are. The top reason why students liked or disliked architecture was the performance they were able to achieve. For most students, this weighs in favor of x86-64 and against the A64FX, likely due to its unusual architecture. Documentation was the second most important factor, both in favor and against systems, with students remarking on the good documentation available for Icelake machines in particular. Furthermore, the ability to understand what was happening in hardware was important for students. In the end, when asked whether they think that they can choose which architecture to use given an algorithm, students were fairly confident (3.2/5) that they would be able to make a good decision. When asked whether they think they will be able to explain performance results, students were also fairly confident (3.3/5) that they would be able to explain the cause. Additionally, when asked what other architectures they would have been interested in, popular candidates included RISC-V systems, FPGAs, and AI-specific architectures.

6.6 Lab Evaluation

Last but not least, we asked students to evaluate the course itself. In general, students describe the difficulty of understanding the algorithms for the assignments as medium (3-5 hours) and for the projects as difficult (approximately 2 days). Some students felt that they could have benefited from more in-depth introductions to the algorithms, and it was found that self-guided research was the primary source of information for understanding concepts and developing solutions ahead of the provided course material and API/standards documentation. Further, students felt that the lab could be improved by additional feedback to students as well as sample solutions, indicating the desire to reduce the open-endedness slightly in favor of a more rigidly structured course. Overall, many participants thought that the course matched their expectations fairly well (3.7/5) and that the exercises helped them understand modern computer architecture (3.8/5). Many were also interested in further student work such as a thesis or guided research.

Throughout the semesters of the BEAST Lab, we gained a lot of experience with running the lab course and the surrounding infrastructure. We believe this course is unique among the offerings at our university, and based on student engagement in class and the provided feedback, we know that students have the opportunity to learn practical aspects of high performance computing not covered in the theoretical courses. We therefore believe that the BEAST Lab is a win for both the students taking the course and the staff supporting it.

7 RELATED WORK

Various valuable training programs, materials, and courses have been developed by the HPC community. In order to locate the scope of the BEAST Lab, we look at recent programs from three main perspectives: 1) the primary educational purpose, 2) the target audience, and 3) the teaching approach.

The Supercomputing Institute internship program at Los Alamos National Laboratory [22] provides a basis in cluster computing for undergraduate and graduate students, offering an educational experience to students and serves as an important recruitment tool for HPC field. RWTH Aachen University offers a software lab [19] to students, in which the students develop parallel code using modern tools while focusing on High-Performance Computing foundations and parallel programming skills. This lab also introduces a self-paced learning approach where status surveys, developer diaries, and group competitions are used to motivate and track students' progress. The German National Supercomputing Center HLRS has expanded its academic training programs to include courses for students, teachers, and professionals in response to the increasing demand from broader application domains and relevance of high-performance computing and simulations beyond academia[7]. Kokkos is presented as a generic parallel programming model suitable for the education of a broader audience, including academia, in the recent work of Sandia National Laboratory [3], where they introduce the best practices obtained from giving virtual classes on Kokkos. Michigan State University uses a flipped classroom model and a “hands-on” approach in teaching parallel programming to undergraduates in targeting STEM fields [4]. College of Meteorologic Oceanography at National University of Defense Technology in China offers a parallel computing course [2] tailored for atmospheric science majors, addressing challenges faced by non-computer science students in understanding parallel scalability. Texas A&M High Performance Research Computing (HPRC) [1] explored educational approaches in response to COVID-19 pandemic using virtual sessions and peer-learning environment.

Overall, in most cases, the existing programs focus on introducing the basics or mainstream HPC programming models and tools. Also, existing programs mainly target undergraduate or graduate student education or consider interdisciplinary domains (e.g., ModSim or AI) and, therefore, target a broad community of researchers beyond the computer science discipline.

The educational challenge of teaching high-performance computing in the face of rapid heterogeneous hardware innovation and adoption renders parts of textbooks obsolete [10]. Reed College uses a diverse heterogeneous hardware and software environment for computer science majors. Artificial Intelligence National Laboratory of Hungary, in collaboration with NVIDIA Deep Learning Institute, discusses challenges in accelerated heterogeneous parallel computing and deep learning education and presents the structure of their Instructor-led Workshops [12]. Magic Castle [9] of Jülich Supercomputing Centre in Germany creates the supercomputer experience in public or private Clouds enabling scalable HPC training through provisioning of virtual supporting infrastructures. Indiana University uses Jetstream National Science Foundation Cloud infrastructure [5], to provide practical HPC training experiences for both HPC administrators and users, covering concepts from basic command-line usage to advanced cluster management. FreeCompilerCamp.org of Lawrence Livermore National Laboratory [21] is an open online platform designed to train researchers in developing OpenMP compilers primarily to address the lack of training resources for researchers who are involved in the compiler and language development around OpenMP. Another worth-mentioning work is on teaching methods and hardware platforms used by

Purdue Research Computing for educating HPC system administrators [23]. Overall, we can conclude that providing resources to users for training is a central piece of the teaching method in most of the existing programs. Also, looking at ECP and PRACE education programs, we conclude that existing programs often target platform-specific tools and optimizations with the intention of better utilization of specific computing facilities.

BEAST Lab shares the spirit of recent training methods and programs in providing resource access to students and extends the scope of the micro-architecture of modern HPC platforms and programming models.

8 LESSON LEARNED, SUCCESS STORIES, AND DISCUSSIONS

Here we summarize a few of the lessons we learned from running the lab:

- most students registering for the lab are interested in recent technology. Getting access to expensive data center hardware motivates them to invest a lot of time in assignments.
- often it is difficult for students to come up with explanations for observed effects. Weekly meetings with a good mix of background information and in-depth discussion of detailed explanations are important for understanding.
- students always work more before deadlines. A job scheduler would not allow students to keep up with deadlines, so measurements may come from congested and overloaded systems. It is important to split up student groups to work on different hardware to reduce this effect.
- next to simple micro-benchmarks (Part 1), working on more complex code is important to see the bigger picture on the challenges of well-tuned HPC code. Examples: just tuning one kernel is not helpful (Amdahl's law will kick in early); on GPU it is important to keep data structures as long as possible on the accelerator.

We can identify two types of success stories. First, we see that Part 1 of the lab (discussing results of micro-benchmarks) helps students to get good results on optimizing real-world codes (Part 2 of the lab). Second, a lot of students come back for student work (bachelor's/master's theses) around BEAST. It is also nice to hear from nearside researchers that students cite their participation in the BEAST lab when searching for final theses. Finally, it is nice to have a colleague helping advise the BEAST lab now who joined the lab as a student before.

However, from the perspective of LRZ, it is important to discuss whether specific goals could be achieved. Especially valuable are experiences about (1) the maturity of software stacks, (2) the ease of getting into parallel programming with a given API/framework, and (3) whether programming models allow users to get the expected performance. In regards to (1), we wanted to understand the maturity of support for ARM. It was good to see that students had no issues with ARM compilers; the ecosystem around ARM seems to be so well supported nowadays that switching from x86 to ARM as host architecture for large HPC systems is not any risk. In regards to (2), we wanted to know if it is reasonable to start with OpenMP GPU support to get users into GPU programming. Experience from the lab shows definitely that it is beneficial for users to have an easy start. Once they see their code running on GPUs, it is relatively easy to dig deeper. However, the latter is required to get reasonable performance (3). Results from OpenMP Offloading often were disappointing.

9 CONCLUSION AND OUTLOOK

The BEAST Lab is a success story for all participating institutions, the universities, and the data center. While it is known among students to be challenging, we see students acquiring a lot of insights about CPU and GPU architectures, enabling them to come up with significant speedups for given HPC application codes. The lab successfully attracts good students. Furthermore, the "Future Computing" program benefits from the lab. First, students ask to do their Bachelor/Master theses with LRZ. Second, a lot of valuable experience is gathered with the help of the lab: it is helpful to know that, e.g., migrating users to a new ISA or to GPUs for the next leadership system can be done without much risk. And the experience becomes a guide for what kind of user training is required for successful migration to the next system.

REFERENCES

[1] Joseph Bungo and Daniel Wong. 2021. Bringing GPU Accelerated Computing and Deep Learning to the Classroom. *The Journal of Computational Science Education* 12 (Feb. 2021), 21–21. Issue 2.

[2] Juan Chen, Brett A. Becker, Youwen Ouyang, and Li Shen. 2021. What Influences Students' Understanding of Scalability Issues in Parallel Computing? *The Journal of Computational Science Education* 12 (Feb. 2021), 58–65. Issue 2.

[3] Jan Ciesko, David Poliakoff, Daisy S. Hollman, Christian C. Trott, and Damien Lebrun-Grandié. 2020. Towards Generic Parallel Programming in Computer Science Education with Kokkos. In *2020 IEEE/ACM Workshop on Education for High-Performance Computing (EduHPC)*. 35–42.

[4] Dirk Colbry. 2021. The Design of a Practical Flipped Classroom Model for Teaching Parallel Programming to Undergraduates. *The Journal of Computational Science Education* 12 (Feb. 2021), 41–45. Issue 2.

[5] Eric Coulter, Richard Knepper, and Jeremy Fischer. 2019. Programmable Education Infrastructure: Cloud resources as HPC Education Environments. *The Journal of Computational Science Education* 10 (Jan. 2019), 107–107. Issue 1.

[6] Jack Dongarra and Piotr Luszczek. 2011. *TOP500*. Springer US, Boston, MA, 2055–2057. https://doi.org/10.1007/978-0-387-09766-4_157

[7] Tibor Döpper, Bärbel Große-Wöhrmann, Doris Lindner, Darko Milakovic, et al. 2021. Expanding HLRS Academic HPC Simulation Training Programs to More Target Groups. *The Journal of Computational Science Education* 12 (Dec. 2021), 13–26. Issue 3.

[8] PRACE Training Events. 2023. Partnership for Advanced Computing in Europe. https://prace-ri.eu/training-support/training/. Accessed: 2023-08-10.

[9] Félix-Antoine Fortin and Alan Ó Cais. 2022. Magic Castle —Enabling Scalable HPC Training through Scalable Supporting Infrastructures. *The Journal of Computational Science Education* 13 (April 2022), 21–22. Issue 1.

[10] Eitan Frachtenberg. 2021. Experience and Practice Teaching an Undergraduate Course on Diverse Heterogeneous Architectures. In *2021 IEEE/ACM Ninth Workshop on Education for High Performance Computing (EduHPC)*. 1–8.

[11] Todd Gamblin, Matthew LeGendre, Michael R. Collette, Gregory L. Lee, Adam Moody, Bronis R. de Supinski, and Scott Futral. 2015. The Spack package manager: bringing order to HPC software chaos. In *SC '15: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 1–12. https://doi.org/10.1145/2807591.2807623

[12] Balint Gyires-Toth, Icsl Oz, and Joe Bungo. 2023. Teaching Accelerated Computing and Deep Learning at a Large-Scale with the NVIDIA Deep Learning Institute. *The Journal of Computational Science Education* 14 (July 2023), 23–30. Issue 1.

[13] Yun (Helen) He and Rebecca Hartman-Baker. 2022. Best Practices for NERSC Training. *The Journal of Computational Science Education* 13 (April 2022), 23–26. Issue 1.

[14] O. Marques and A. Barker. 2020. Training Efforts in the Exascale Computing Project. *Computing in Science &; Engineering* 22, 05 (2020), 103–107.

[15] Satoshi Matsuoka. 2018. Cambrian Explosion of Computing and Big Data in the Post-Moore Era. In *Proceedings of the 27th International Symposium on High-Performance Parallel and Distributed Computing (HPDC '18)*. Association for Computing Machinery, New York, NY, USA, 105. https://doi.org/10.1145/3208040.3225055

[16] Alessio Netti et al. 2019. From Facility to Application Sensor Data: Modular, Continuous and Holistic Monitoring with DCDB. In *Supercomputing 2019 (SC'19)*.

[17] Amir Raoofy, Roman Karlstetter, Dai Yang, Carsten Trinitis, and Martin Schulz. 2020. Time Series Mining at Petascale Performance. In *High Performance Computing: 35th International Conference, ISC High Performance 2020, Frankfurt/Main, Germany, June 22–25, 2020, Proceedings*. Springer-Verlag, Berlin, Heidelberg, 104–123.

[18] Maron Schlemon, Martin Schulz, and Rolf Scheiber. 2022. Resource-Constrained Optimizations For Synthetic Aperture Radar On-Board Image Processing. In *2022 IEEE High Performance Extreme Computing Conference (HPEC)*. 1–8.

[19] Christian Terboven, Julian Miller, Sandra Wienke, and Matthias S. Müller. 2020. Self-paced Learning in HPC Lab Courses. *The Journal of Computational Science Education* 11 (Jan. 2020), 61–67. Issue 1.

[20] Ulrich Trottenberg, Cornelis W. Oosterlee, and Anton Schüller. 2001. *Multigrid*. Texts in Applied Mathematics. Bd., Vol. 33. Academic Press, San Diego [u.a.]. With contributions by A. Brandt, P. Oswald and K. Stüben.

[21] Anjia Wang, Alok Mishra, Chunhua Liao, Yonghong Yan, and Barbara Chapman. 2020. FreeCompilerCamp.org: Training for OpenMP Compiler Development from Cloud. *The Journal of Computational Science Education* 11 (Jan. 2020), 53–60. Issue 1.

[22] J. Lowell Wofford and Cory Lueninghoener. 2020. The Supercomputer Institute: A Systems-Focused Approach to HPC Training and Education. *The Journal of Computational Science Education* 11 (Jan. 2020), 73–80. Issue 1.

[23] Alex Younts and Stephen Lien Harrell. 2020. Teaching HPC Systems Administrators. *The Journal of Computational Science Education* 11 (Jan. 2020), 100–105. Issue 1.

HPC Carpentry—A Scalable, Peer-reviewed Training Program to Democratize HPC Access

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

Alan Ó Cais
University of Barcelona
alan.ocais@cecam.org

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

Wirawan Purwanto
Old Dominion University
wpurwant@odu.edu

Annajiat Alim Rasel
Brac University
annajiat@gmail.com

ABSTRACT

The HPC Carpentry lesson program is a highly interactive, hands-on approach to getting users up to speed on HPC cluster systems. It is motivated by the increasing availability of cluster resources to a wide range of user groups, many of whom come from communities that have not traditionally used HPC systems. We adopt the Carpentries approach to pedagogy, which consists of a workshop setting where learners type along with instructors while working through the instructional steps, building up "muscle memory" of the tasks, further reinforced by challenge exercises at critical points within the lesson. This paper reviews the development of the HPC Carpentry Lesson Program as it becomes the first entrant into phase 2 of The Carpentries Lesson Program Incubator. This incubator is the pathway for HPC Carpentry to become an official lesson program of The Carpentries.

KEYWORDS

HPC, training, open source, lesson

1 INTRODUCTION

HPC Carpentry is a set of lessons whose goal is to introduce the ins and outs of running applications on HPC resources to new audiences, including investigators from fields which are not traditional users of high-performance computing, as well as novice users from fields in which HPC is commonly used.

For historical reasons, we have a variety of lessons at varying stages of maturity, and are in the process of crafting some of these lessons into workshops or tracks, which can be presented together to bring learners up to speed on modern HPC resources.

1.1 The Carpentries

We take advantage of pedagogical methods built up by The Carpentries [3]. This organization was founded 25 years ago to solve a related problem, that of getting scientists to engage better with code development best practices, to help investigative teams capture knowledge generated by possibly-short-term participants, and to improve reproducibility of computational science.

The fundamental idea of Carpentries workshops is two-fold. Firstly, within the workshop setting, instructors and learners live-code the lesson steps together, with a strong emphasis on ensuring that instructors explain the process in detail, including real-time post-mortems on typos and errors. Learners exercise the correct workflow, building up "muscle memory" of the process, and also get an organic sense of what common errors look like, and how to recover from them. Secondly, the workshop material is open-source, and feedback gathered after a workshop, from learners, instructors, and organizers, and can be translated into modifications to the lessons, so that they are continuously improved. Because the lessons are shared across the Carpentries community, there are many eyes on the content, and "bugs" are quickly found and corrected for all future users.

The limited time of the workshop setting admits modest but important pedagogical goals – learners are expected to come away not as experts in the material they have just worked through, but rather with a solid grounding in what success looks like, search terms to drive later discovery, and continued access to the lesson material itself.

2 HPC CARPENTRY LESSON PROGRAM

HPC Carpentry addresses a similar need to The Carpentries, namely on-boarding novice HPC users. Many of the same constraints that motivated The Carpentries strategy are similar – there is no room in anyone's curriculum for a formal course in HPC usage, but the material is unfamiliar, and HPC interactions are unlike other ways that computers are used. The same strengths of The Carpentries – the workshop setting and live-typing technique – help novice HPC users just as they do novice users of Git, Bash, or Python.

2.1 A Brief History

Birds of a Feather sessions at SC17 [13] and SC21 [11] demonstrate that, for some time, there has been significant interest among the HPC community in The Carpentries approach to generating and delivering training content.

The HPC Carpentry GitHub organisation was created in 2017 as part of HPC lesson development efforts by Compute Canada (now Digital Research Alliance of Canada). There were also other ongoing HPC lesson development efforts such as the HPC Parallel

*Presenter, all authors contributed equally to this submission.

https://doi.org/10.22369/issn.2153-4136/15/1/6

novice lesson [12]. At CarpentryCon 2018, HPC Carpentry had 2 sessions [2, 9] (each with ~40 participants), where much of the discussion centered around how to merge existing efforts and form a single group of collaborators to drive the lesson development forward.

Over the past few years, in working towards Carpentries lesson-program incubation status, the current team has identified components which can be organized into thematic Carpentries-style workshops. The themes identified are a user workshop, taking learners from an introduction to the shell through to running a simple application on a cluster, and a developer workshop, taking learners through the execution of a code example for a parallel framework, such as MPI. At the same time, we do not wish to devalue or discard the niche or advanced material which we have access to, and which does not neatly fit into the workshop themes. Identifying and developing the workshop-appropriate material has been one of the significant challenges of the past few years.

Over the course of all of this development, a number of teams have forked lesson material on GitHub at various points in its development. We recently reached out to many of these teams to try to ascertain whether there were valuable additions to the material that had been added, that we might want to capture, and to try to identify opportunities for re-integration or collaboration. This effort promises to rekindle some prior relationships, and we are hopeful that this will help us serve a broader community than would otherwise be reached, but re-integration has certainly been another significant challenge.

2.2 Lessons in the Program

The current user-oriented lesson program of HPC Carpentry consists of three lessons:

- **"The Unix Shell"** [1] – This lesson is included directly from Software Carpentry. To quote from the lesson itself: "Use of the shell is fundamental to a wide range of advanced computing tasks, including high-performance computing."
- **"Introduction to High Performance Computing"** [5] – The most highly developed of our lessons, which takes learners from basic shell use to dispatching parallel jobs on an HPC cluster, and includes careful feedback-driven descriptions of various jargon terms.
- **"HPC Workflow Management with Snakemake"** [4] – Currently under development, this lesson is meant to follow the "Introduction to HPC" in a workshop setting. It takes the user from dispatching jobs to performing a workflow-managed scaling study on a reference executable which illustrates Amdahl's law (using Snakemake [10] as the workflow tool).

As mentioned earlier, for historical reasons, there are a number of other lessons in The Carpentries format, created and maintained by the wider community, which encapsulate valuable material but are not part of the initial workshop plan. Examples include a lesson on the Chapel program language [6] and the use of containers [7]. The Carpentries lesson incubator provides a venue where relevant new material can be identified, and possible new contributors brought into the community.

2.3 Continuous Improvement Through Continuous Assessment

The Carpentries model seeks to actively gather learner feedback to foster lesson improvement. Each delivery of the lesson program includes a pre- and post- workshop surveys for each Carpentries Workshop to help evaluate the effectiveness of the lesson and identify opportunities for further improvement.

2.4 Portability

All of our lessons are maintained in public GitHub repositories. The lesson "Introduction to High Performance Computing" currently allows for extensive customisation for use at a particular site. For the purposes of providing community access to HPC resources, we are planning to converge our lessons such that the default content refers to a "reference cluster," built in the cloud, that meets our general requirements and can be easily reproduced (and hence scaled out). Adapting the lessons to a specific site, for workshops that benefit from this, will be still be possible.

2.5 Supporting Infrastructure

We currently have access to cloud resources, where we use Magic Castle [8] to create instances of "HPC clusters" that can serve the requirements of the lesson program. We anticipate that our relationship with The Carpentries will help us to acquire and manage additional cloud resources, either in-kind or via funding, which should facilitate future workshops.

3 FUTURE DIRECTIONS

Our near-term strategy at this point is to complete and begin teaching the material for the two workshops mentioned above, one for HPC users, and one for developers. The workshops will both start with the existing "The Unix Shell" lesson, maintained by The Carpentries, followed by an "Introduction to High Performance Computing". Afterwards, the workshops will diverge, with the "user" workshop continuing with the "HPC Workflow Management with Snakemake" lesson, and the "developer" workshop continuing with a lesson in the operation of a parallel framework, such as MPI.

3.1 The Carpentries Lesson Program Incubator

With the ongoing support of The Carpentries, it is hoped that the HPC user-oriented workshop described above will be the first entrant into Phase 2 of The Carpentries Lesson Program incubation process. Incubation is a 3 phase process with the final phase being adoption as an official lesson program of The Carpentries (alongside Software Carpentry, Data Carpentry and Library Carpentry).

3.2 Development of Additional Lessons

Aside from the lessons intended for workshop integration, we also plan to continue to welcome and disseminate contributions on general HPC topics. We are aware that HPC is used for many things, so we can imagine application-specific lessons for popular HPC applications, or lessons for more compact and expressive languages for HPC applications.

3.3 Engagement with HPC Education and Training Communities

As we enter a new incubation phase with The Carpentries, we wish to also increase our engagement with the wider HPC education and training communities. The continuous improvement model we use relies on a steady stream of instructors, workshops and integration of feedback from those workshops. Connection with this group can help to streamline this process.

4 CONCLUSION

HPC Carpentry is entering an exciting phase, where seeds that were planted years ago are starting to bear fruit. HPC Carpentry has honed an initial lesson program to the point where it is being considered as an official lesson program of the Carpentries. It has addressed, and found workable solutions for, complex issues such as access to HPC resources for training purposes, and site-specific customisation of community-maintained lessons.

ACKNOWLEDGMENTS

The authors would like to thank The Carpentries for their ongoing engagement with this effort. Alan Ó Cais is supported by Digital Europe Programme under Grant No. 101100604 (BioNT).

REFERENCES

[1] Inigo Aldazabal Mensa, Harriet Alexander, James Allen, Areej Alsheikh-Hussain, Daniel Baird, Piotr Banaszkiewicz, Pauline Barmby, Rob Beagrie, Trevor Bekolay, Evgenij Belikov, Jason Bell, Kai Blin, John Blischak, Simon Boardman, Maxime Boissoneault, Jessica Bonnie, Andy Boughton, Ry4an Brase, Amy Brown, Dana Brunson, Orion Buske, Abigail Cabunoc Mayes, Daniel Chen, Kathy Chung, Gabriel A. Devenyi, Emily Dolson, Jonah Duckles, Rémi Emonet, David Eyers, Filipe Fernandes, Hugues Fontenelle, Francis Gacenga, Matthew Gidden, Ivan Gonzalez, Norman Gray, Varda F. Hagh, Michael Hansen, Emelie Harstad, Adina Howe, Fatma Imamoglu, Damien Irving, Mike Jackson, Emily Jane McTavish, Michael Jennings, Dan Jones, Alix Keener, Kristopher Keipert, Tom Kelly, Jan T. Kim, W. Trevor King, Christina Koch, Bernhard Konrad, Sherry Lake, Doug Latornell, Philip Lijnzaad, Eric Ma, Joshua Madin, Camille Marini, Kunal Marwaha, Sergey Mashchenko, FranÃ§ois Michonneau, Ryan Middleson, Jackie Milhans, Bill Mills, Amanda Miotto, Sarah Mount, Lex Nederbragt, Daiva Nielsen, Aaron O'Leary, Randy Olson, Adam Orr, Nina Therkildsen, Kirill Palamartchouk, Adam Perry, Jon Pipitone, Timothée Poisot, Hossein Pourreza, Timothy Povall, Adam Richie-Halford, Scott Ritchie, Noam Ross, Halfdan Rydbeck, Mahdi Sadjadi, Pat Schloss, Bertie Seyffert, Genevieve Shattow, Raniere Silva, Sarah Simpkin, John Simpson, Byron Smith, Nicola Soranzo, Ashwin Srinath, Daniel Standage, Meg Staton, Peter Steinbach, Marcel Stimberg, Bartosz Telenczuk, Florian Thoele, Tiffany Timbers, Stephen Turner, Jay van Schyndel, Anelda van der Walt, David Vollmer, Jens von der Linden, Andrew Walker, Josh Waterfall, Ethan White, Carol Willing, Greg Wilson, Donny Winston, Lynn Young, and Lee Zamparo. 2016. Software Carpentry: The Unix Shell. https://doi.org/10.5281/zenodo.57544

[2] Alan Ó Cais and Daniel Smith. 2018. Breakout 8: HPC Carpentry. https://github.com/carpentries/carpentrycon/tree/master/CarpentryCon-2018/Sessions/2018-05-31/05-Breakout-8-HPC-Carpentry. Breakout session at CarpentryCon 2018.

[3] The Carpentries. 2023. The Carpentries teaches foundational coding and data science skills to researchers worldwide. https://carpentries.org/.

[4] HPC Carpentry. 2023. HPC Workflow Management with Snakemake. https://carpentries-incubator.github.io/hpc-workflows/.

[5] HPC Carpentry. 2023. Introduction to High Performance Computing. https://carpentries-incubator.github.io/hpc-intro/.

[6] HPC Carpentry. 2023. Introduction to High-Performance Computing in Chapel. https://www.hpc-carpentry.org/hpc-chapel/.

[7] Pawsey Supercomputing Centre. 2023. Containers on HPC and Cloud with Singularity. https://pawseysc.github.io/singularity-containers/.

[8] Félix-Antoine Fortin, etiennedub, Frédéric Fortier-Chouinard, Maxime Boissonneault, Charles Coulombe, Erik Sundell, Simon Guilbault, Adam Huffman, jboschee, R. P. Taylor, Étienne Parent, Chris Want, Darren Boss, Justin Lagüe, Kenneth Hoste, Marie-Helene Burle, Quoding, and Pier-Luc St-Onge. 2023. ComputeCanada/magic_castle: Magic Castle 12.6.0. https://doi.org/10.5281/zenodo.8096727

[9] Christina Koch and Peter Steinbach. 2018. Workshop 5: HPC Carpentry. https://github.com/carpentries/carpentrycon/tree/master/CarpentryCon-2018/Sessions/2018-06-01/05-Workshop-5-HPC-Carpentry. Workshop at CarpentryCon 2018.

[10] Felix Mölder, Kim Philipp Jablonski, Brice Letcher, Michael B. Hall, Christopher H. Tomkins-Tinch, Vanessa Sochat, Jan Forster, Soohyun Lee, Sven O. Twardziok, Alexander Kanitz, Andreas Wilm, Manuel Holtgrewe, Sven Rahmann, Sven Nahnsen, and Johannes Köster. 2021. Sustainable data analysis with Snakemake. *F1000Research* 10 (April 2021), 33. https://doi.org/10.12688/f1000research.29032.2

[11] Andrew Reid, Trevor Keller, Alan Ó Cais, and Annajiat Alim Rasel. 2021. HPC Carpentry: Introducing New Users to HPC. https://sc21.supercomputing.org/app/uploads/2021/11/SC21-Final-Program-211114.pdf. BoF session at SC21.

[12] Peter Steinbach. 2021. psteinb/hpc-parallel-novice: fork to work. https://doi.org/10.5281/zenodo.4525377

[13] Andrew Turner, Christina Koch, Tracy Teal, Robert Freeman Jr, Chris Bording, and Martin Callaghan. 2017. HPC Carpentry - Practical, Hands-On HPC Training. https://sc17.supercomputing.org/index.html%3Fpost_type=page&p=5407&id=bof125&sess=sess359.html. BoF session at SC17.

Using Unity for Scientific Visualization as a Course-based Undergraduate Research Experience

Idunnuoluwa Adeniji
Kean University
Argonne National Laboratory
iadeniji@kean.edu

Lark Bancairen
Kean University
bancaila@kean.edu

Cymantha Blackmon
Kean University
blackmoc@kean.edu

Michael Casarona
Kean University
casaromi@kean.edu

Melissa Menzel
Kean University
menzelm@kean.edu

Matthew G. Niepielko
Kean University
mniepiel@kean.edu

Leonard Bielory
Kean University
Rutgers University – Center of Environmental Prediction
Hackensack Meridian School of Medicine
lbielory@kean.edu

Nan Perigo
Kean University
nperigo@kean.edu

Joseph Insley
Argonne National Laboratory
Northern Illinois University
Insley@anl.gov

David Joiner
Kean University
djoiner@kean.edu

ABSTRACT

We have developed a series of course-based undergraduate research experiences for students integrated into course curriculum centered around the use of 3D visualization and virtual reality for science visualization. One project involves the creation and use of a volumetric renderer for hyperstack images, paired with a biology project in confocal microscopy. Students have worked to develop and test VR enabled tools for confocal microscopy visualization across headset based and CAVE based VR platforms. Two applications of the tool are presented: a rendering of *Drosophila* primordial germ cells coupled with automated detection and counting, and a database in development of 3D renderings of pollen grains. Another project involves the development and testing of point cloud renderers. Student work has focused on performance testing and enhancement across a range of 2D and 3D hardware, including native Quest apps. Through the process of developing these tools, students are introduced to scientific visualization concepts, while gaining practical experience with programming, software engineering, graphics, shader programming, and cross-platform design.

KEYWORDS

Unity, Visualization, course-based undergraduate research experiences

1 INTRODUCTION

Significant study has been made in the impact on the student experience of course-based undergraduate research experiences (CUREs), in which activities focused on following the research process and performing inquiry is used in addition to or in place of traditional laboratory activity [4]. Positive benefits of CUREs have been seen in professional identity, research skills, project ownership, and higher retention [2, 3, 8, 11].

Our context for implementation is a computational science and engineering program at Kean University, a regional university in northern New Jersey. The program is part of a 5-year combined BS/MS honors program in the School of Integrative Science and Technology.

The projects presented in this paper involve development or enhancement of software in Unity Game Engine [10]. The four CUREs that the computational science students have been working on presented in this paper center around volumetric rendering of hyperstack data, and performance testing and implementation of point cloud renderers in Unity using VR hardware.

2 METHODS

2.1 Hyperstack Image Rendering

VR is increasingly being used for visualization of hyperstack data in scientific and medical imaging [9]. It has shown potential for use specifically with confocal microscopy along with other data [16].

We have worked with students implementing a technique for rendering hyperstacks by using transparency and shading within Unity's rendering pipeline, as opposed to traditional raytracing, for a process that is performant, easy to implement, and is cross platform compatible across laptop and VR hardware.

Our focus with student research projects has been on two sets of data, the first involving the formation of primordial germ cells in fruit fly embryos. Students were given the task of developing tools to count the number of cells in the embryo, with cells overlapping in 3D and requiring a 3D object detection approach. The second data set is a developing database of 3D scans of pollen grains. While multiple databases exist of 2D images of pollen grains for the purpose of both human and AI training for pollen detection, there are far fewer databases of 3D pollen scans, a notable exception being the recently introduced 3D Pollen Project [19]. As part of a project to provide 3D images of common US pollen species, our students use Unity to render

https://doi.org/10.22369/issn.2153-4136/15/1/7

hyperstack images of pollen grains to increase availability of 3D pollen data.

2.2 Point Cloud Rendering

An increasing amount of scientific data is rendered as point clouds, both due to the use of higher quality 3d scanners as well as traditional glyph renderings of large point-based computations. Creating point clouds in Unity has traditionally used geometry shaders [15]. In cases where geometry shaders are not supported, some software has instead used point shaders with a pointsize parameter [17]. Other work in the area has focused on techniques for efficient point decimation for faster loading and display, with typical renderings done at a level of one million points [5].

We set out to have students compare different techniques for point cloud rendering, as well as perform cross platform testing. Students were assigned the task initially of working with the Keijero point cloud model, and testing it with data sets of varying size, as well as across platforms, testing on Windows, Android, and OS X using where possible Direct X, OpenGL, Vulkan, and Metal graphics APIs, using both PC, Oculus Rift, Oculus Quest, and Oculus Quest 2 hardware. Over the course of this project, additional geometry-based shaders were developed and tested as well.

3 CURES PROJECTS

3.1 Hyperstack Image Rendering Projects

3.1.1 Primordial Germ Cells rendering network viewer

This project has focused on creating multi-user spaces in VR to view and share 3D models of primordial germ cell data from a confocal microscope. Primordial Germ Cells (PGCs) are embryonic precursors that pass on both genetic and epigenetic information to succeeding generations [5]. While broadly applicable to any hyperstack image data, the particular application is driven by a project developing automated analysis tools for detecting cells, and the need to validate results in a 3D view. We have pursued a multi-user environment to allow remote viewing of data by multiple simultaneous viewers, based on Photon 2 Unity Networking. A web-based front end lets users upload these images and other pertinent data, and configuration files for a pre-built unity scene are automatically created. Additional tools are also in development to allow users more fine-grained control over the generated VR models, including the ability to save changes. By advancing these elements, the project aims to offer a comprehensive, user-friendly platform for 3D hyperstack image model visualization and collaboration.

3.1.2 3D Pollen Database

Seasonal allergic rhinitis (SAR) is a common inflammatory condition caused by pollen grains released by trees, grasses, weeds, or molds [6]. Many people are affected by the cold-like symptoms caused by these various pollen species [12]. Therefore, streamlining the identification and distribution of real-time pollen conditions is important because it can provide allergy sufferers with useful information to help reduce pollen exposure. We apply confocal microscopy to capture diversities in pollen grain structures, which can be further used in 3D analysis of pollen structures. Whereas other types of microscopes can only allow the external characteristics of samples to be seen, confocal microscopy allows a sample to be imaged in slices along its z-axis which are then used to create 3D and cross-sectional images. This allows the images to not only display the external morphologies and characteristics that are unique to each pollen species, but also the internal morphologies and characteristics.

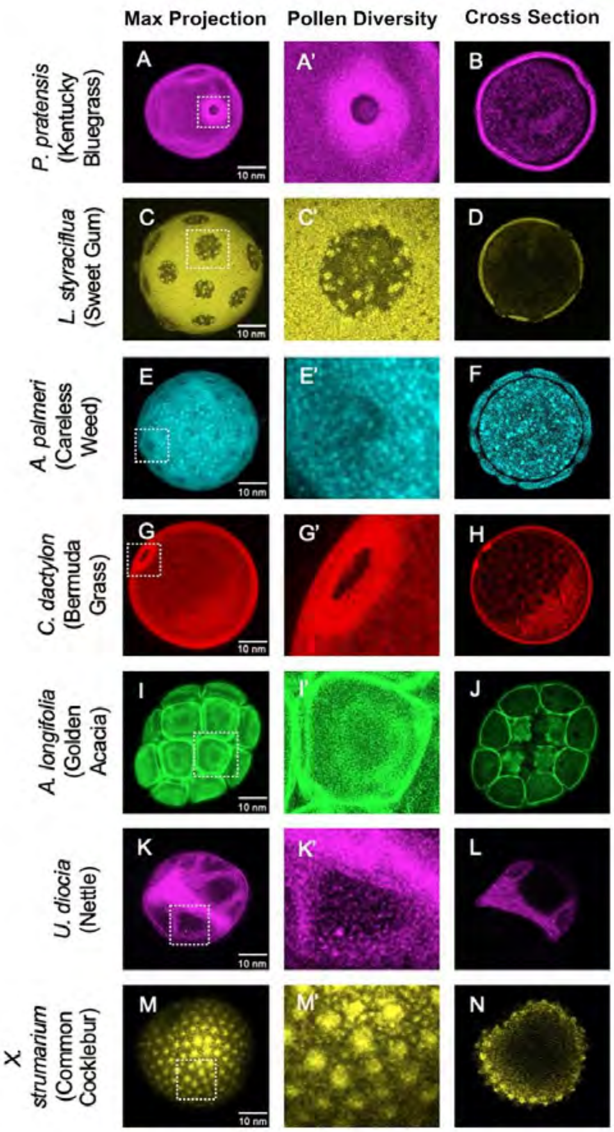


Figure 1. Confocal microscopy captures diversity in pollen structures. A-M) Max project confocal images (first column), captures differences in species' structures. A′-M′) Enlarged sections (second column) of the broken white boxes in the first column highlight key features for each pollen species. A-M) Image cross sections (third column), reveals additional differences between pollen species' structures.

The initial data used is shown in Figure 1. Each pollen species shown can be identified by their distinctive characteristics. Specifically, Kentucky Bluegrass (A) has one aperture, Sweet Gum (C) has many holes with spikes, Careless weed (E) has many circular indentations, Bermuda Grass (G) has one aperture and a disc shape, Golden Acacia (I) is split into many cube-shaped sections, Nettle (K) has an irregular shape, and Common Cocklebur (M) has small spikes on the exterior.

The goal of this project is to collect, database, and disseminate multiple 3D renderings of pollen grains for further analysis.

3.1.3 *Hyperstack rendering clipping shader implementation.*

A portion of Unity game and object rendering relies heavily on the computer graphics shader attached to objects to apply per-pixel calculations on GPU hardware and to allow for modification of the look and feel of objects based on shading and lighting effects. Shaders are used to create a wide variety of visual effects for both static and dynamic user interfaces. Built-in Unity shaders are typically focused on the needs of gaming, and visualization can often be simplified or made more computationally efficient by directly writing shader code that allows per-pixel (fragment) calculation of viewed effects based on data at known points (vertices).

In the case of our hyperstack images, a custom vertex-fragment shader is used to add transparency to pixel values below a threshold and modify the transparency of viewed pixels based on their intensity value.

Clipping is a standard technique used in 3D visualization software. It can be implemented in Unity at the shader level provided the origin of the clip plane in world space, the normal of the clip plane relative to the clip plane origin, and the position of the pixel rendered in world space.

This student project was to implement and test the efficiency of clipping in the shader with a combination of a dot product and the CG clip function, or in a more advanced approach a step operation in place of clip to create a variable that can be used to create a more blended view (e.g., rendering the positive side of the clip plane with a different transparency value). This has been added to our custom shader in the hyperstack renderings in the two projects mentioned above.

4 Point Cloud Rendering Projects

4.1.1 *HACC simulation visualization*

Traditional large scientific data exploration predominantly relies on 2D and 3D visualization tools. However, a transformative shift is occurring as lower cost virtual reality (VR) hardware emerges, offering immersive experiences. This study showcases the development of a data pipeline from conventional visualization tools like Paraview to Unity Game Engine and the Oculus Quest 2 headset.

The primary goal is to explore this transition, with a focus on enabling a more comprehensive understanding of complex datasets.

5 STUDENT PROJECT RESULTS

5.1 Hyperstack Image Rendering

Student work on hyperstack image rendering focused on techniques for image preparation for rendering. Our pipeline was to open the hyperstacks in Image J and save each z-slice as an individual jpeg. Images were imported into the Unity editor as RGBA 32 bit sprites with alpha as transparency and read/write selected. The images were then projected onto quads, using an unlit 2 sided transparent shader. The shader was designed to clip any pixels below a set threshold and apply an alpha level proportional to the total brightness of the pixel. The quads were spaced in the scene evenly according to their height in *z*.

5.1.1 *Primordial Germ Cells rendering network viewer*

A network-based visualization tool has been created, with a web-based management interface so that users can easily download the visualization tool, as well as create and manage cell models. This will eventually allow for additional custom analytics tools to be run simultaneously with the model creation so that prevalent cell data can be highlighted through automated analysis and then confirmed visually. The scene created for the networked viewer is displayed in Figure 2.

Figure 2. 3D cell model in networked visualization application. Cell images are loaded into the lab environment and compiled into a 3D model. The user interface (UI) located on the board displays all relevant cell information as well as the slider interface that is able to transform the 3D cell model. The lab scene provides a comfortable environment for working with the model as well as provides a professional feeling to the application.

5.1.2 *3D Pollen Database*

An early rendering is shown in Figure 3, with data from a rendering of a Ragweed pollen grain. Input data for rendering was 1576x1576x60, with 100+ FPS performance on a TensorBook w/ GeForce 3080. Other performance measures were 30+ FPS for an Oculus Quest Pro connected with link cable, and 20+ FPS for a native Quest Pro app with data downgraded to 1024x1024x60. Of note was that as we moved from a high-end workstation to a native Quest app, images larger than 1024x1024 noticeably degraded performance.

Further images were limited to 1024x1024, and communication with the biology team resulted in additional student projects focused on production of pollen data. Figure 4 shows a scan of a Golden Acacia pollen grain. Input data for rendering was 1024x1024x90, with 300+ FPS performance on a TensorBook w/ GeForce 3080. Other performance measures were 60+ FPS for an Oculus Quest Pro connected with link cable, and 40+ FPS for a native Quest Pro app.

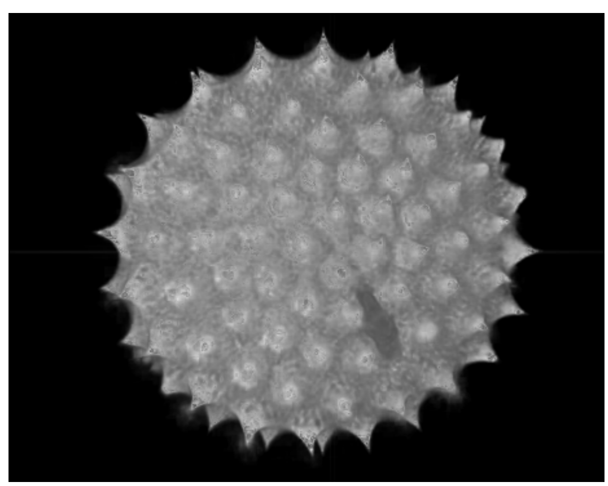


Figure 3. Hyperstack rendering of confocal scan of a Ragweed pollen grain.

In summary, the study found that confocal microscopy can be used to produce detailed images of pollen grain species and provide 3D images that can be reconstructed in Unity.

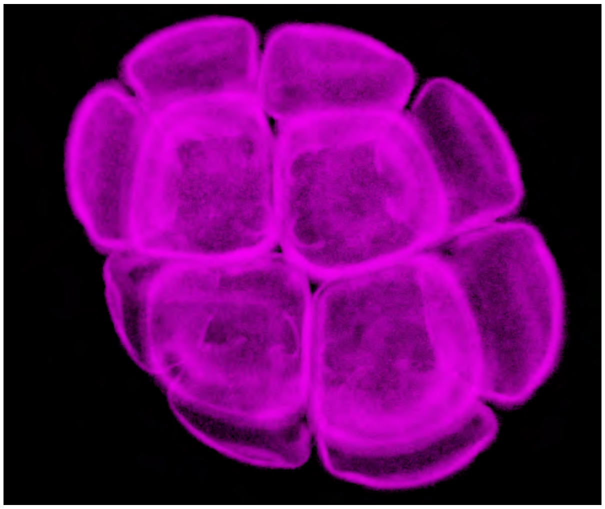


Figure 4. Hyperstack rendering of confocal scan of a Golden Acacia pollen grain.

5.1.3 Hyperstack rendering clipping shader implementation

A clip plane effect was added to the shaders used in the hyperstack viewer, and results can be seen in Figure 5. Clipping was implemented by calculating the world space position of each vertex $\vec{p}_{ws}$ in the shader, and setting shader properties for the base (in world space coordinates) $\vec{b}_{ws}$ and normal $\hat{n}$ (relative to the base) of the clipping plane, at which point the clip function could be applied to $(\vec{p}_{ws} - \vec{b}_{ws}) \cdot \hat{n}$. Addition of clipping plane had no impact on performance with the GeForce 3080 test, Link Cable, or native Quest app.

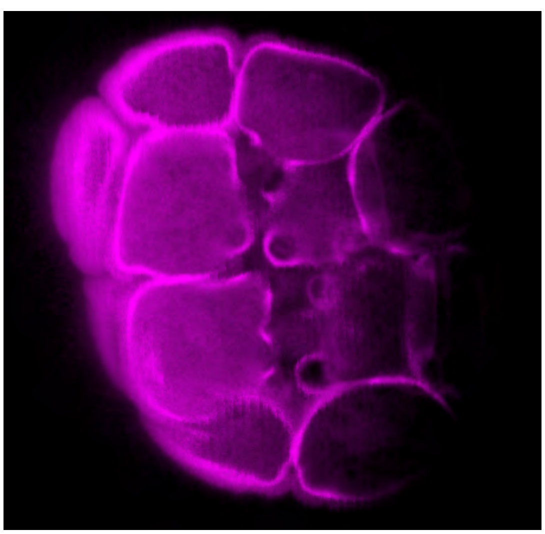


Figure 5. Hyperstack rendering of confocal scan of a Golden Acacia pollen grain with clipping plane applied.

5.2 Point Cloud Rendering

One challenge with developing point cloud rendering approaches that work across a wide range of platforms, including Windows, Mac, and VR headsets, is the difference in graphics API availability for those platforms, with DirectX being windows specific, Vulkan working on Windows or Android, OpenGL supported by Windows, Android, and Intel-based Macs, and Metal supported by Macs. Metal is the only graphics API supported for the M series of Macs [1].

Figure 6 shows a test of the point cloud rendering process, applied to a 10,000,000-point Sierpinski Trapezoid calculated using an iterated function system. The cloud rendered on a TensorBook with a GeForce 3080 at 100+ FPS. Similar performance on other hardware include Quest Pro connected with link cable and 10 million points at 30+ FPS. A natively compiled Quest Pro app with 1,000,000 points ran at 10+ FPS.

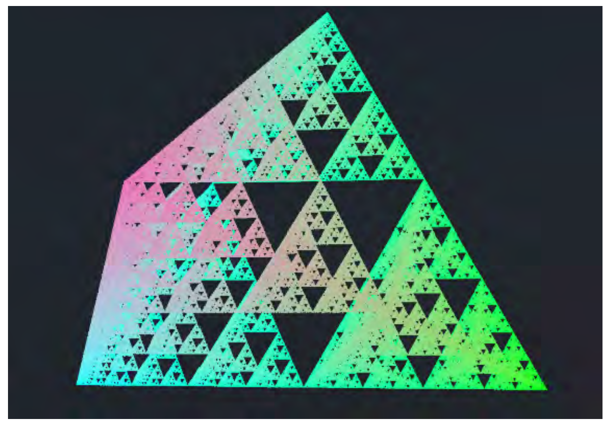


Figure 6. Point cloud of a 10-million-point Sierpinski Tetrahedron

Geometry shaders have proven to have wide cross platform compatibility and good performance on all hardware tested except for M series chip-based Macs. Future student projects will investigate workarounds for rendering on newer Macs restricted to the Metal graphics API.

5.2.1 HACC simulation visualization

This project applied virtual reality to the identification of patterns and clusters within point cloud data, in particular a set of data generated by a Hardware/Hybrid Accelerated Cosmology Code (HACC) simulation (Figure 7, Figure 8).

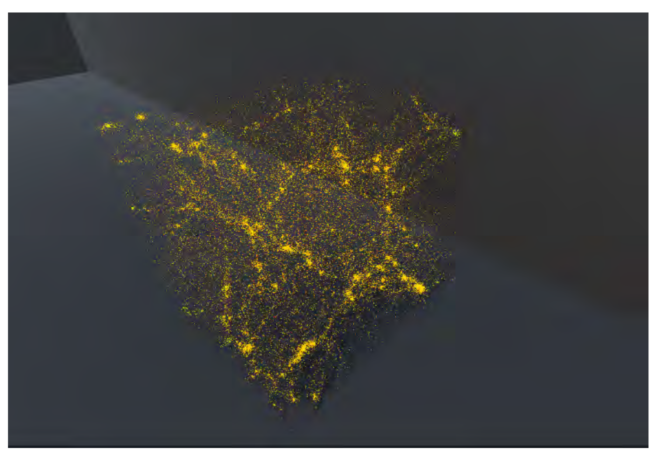


Figure 7. Scene view of HACC data in the VR environment

In addition to enhancing scientists' comprehension of their datasets, this study emphasizes the integration of VR into the broader field of large data exploration, which includes features like data interaction, manipulation, and in-depth analysis. This study employs a custom import script designed for transferring particle data generated by a HACC simulation into Unity. The files we work with contain a sample of approximately 4 million particles (out of up to 2 trillion simulated), from which we load a selected subset. The point cloud data is presently rendered as a point topology mesh using a custom geometry shader.

Through the immersion of users in virtual environments, this study significantly amplifies the identification of patterns and clusters when compared to conventional methods. In addition, we achieved an average PC frame rate of approximately 1012.3 frames per second (FPS) when visualizing 10,000 particles on an Alienware Aurora R15 with a GeForce 3090. When considering the entire dataset of approximately 4 million particles, we attained an average PC frame rate of approximately 235.9 FPS. Beyond static visuals, a custom time lapse technique animates data, providing insights into pattern evolution.

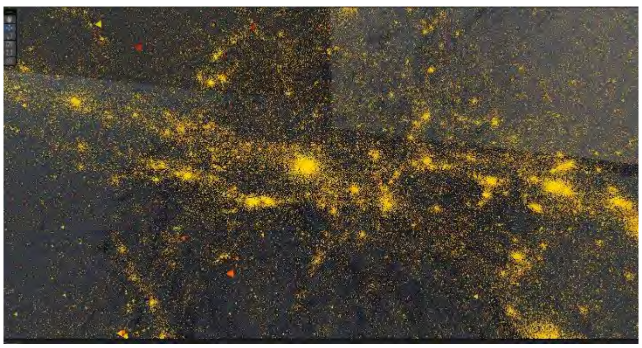


Figure 8. Headset view of HACC data in the VR environment

6 DISCUSSION

VR driven scientific visualization can provide a source of rich, authentic research-based activities for students learning skills in computational science. As a real-world example for students, scientific visualization using Unity in VR has many features we have found beneficial to students. Unity's C# language is similar to Java, which is the primary language used for our students' computer science coursework.

In training students to use Unity for visualization projects, we have developed an offering of our freshman research course sequence focused on Unity programming. Materials used in this are a combination of public materials available at https://learn.unity.com [18], in particular the ubiquitous "Roll-A-Ball" lesson, as well as hosted materials written for the course, available at https://joinerda.github.io/ [14], in particular the "Hello Unity," "Using GetComponent," and "Lorenz Butterfly" tutorials, which allow for covering of Unity in a computational science context. Typically, students then brainstorm ideas on projects. Past course offerings have included projects on using VR to hand count primordial germ cells, as well as non-visualization projects such as creating a lab safety simulation. When adding in discussions of VR, class projects have focused on the concepts of grabbing, rotating, scaling, and activating (pressing) objects – activities that are ubiquitous across modern VR toolkits.

Interested students continue in sophomore, junior, and senior years with independent research, working specifically on visualization projects.

7 REPRODUCIBILITY

For adding virtual reality hardware support to our Unity models, we are using the XR Interaction Toolkit (XRI), which provides a player object that can be used with a variety of VR systems. Students are instructed to use XRI version 2.3.2 or later, and to include the samples when installing, as the samples includes a working player controller and locomotion system that they can quickly copy and paste from the sample scene into their scene rather than configuring from scratch. Students need the ability to plug their headset into their laptop in order to install the apps they create, so at least one USB port needs to be available, and an appropriate cable. Students use the personal edition of Unity Game Engine, though depending on the use case there are also options to request a Unity license for education through Unity's grant program. Students use the community edition of Microsoft Visual Studio C# compiler.

For installing to Quest, Unity needs to be configured for Android build. Recent versions of Unity do not require extensive additional installation of Android toolkits, and installation can be managed solely through the Unity Hub. ASTC Texture compression is selected in the build settings.

Dependencies for XR Interaction Toolkit include the AR Foundation, XR Plugin Management, Oculus XR Plugin, and OpenXR Plugin (if using Oculus Link through a cable) packages. Version information and package requirements are as follows: Unity Version for tests: 2021.3.8f1, AR Foundation 4.2.8, Oculus XR Plugin 3.0.2, OpenXR Plugin 1.4.2, XR Interaction Toolkit 2.3.2, Android Graphics API OpenGLES3.2, Linear color space, OpenXR Plugin used for PC, Oculus Plugin used for Android, Multi-Pass Rendering for both plugins.

Drosophila primordial germ cells were labeled and detected using immunofluorescence and was carried using polyclonal anti-Vasa (Boster Bio, cat #DZ41154) and secondary Alexa Fluor 568 (anti-rabbit, ThermoFisher, cat #A10042) as previously described in detail [7]. Confocal microscopy was carried out using the Leica STELLARIS 5 white light laser system and Leica LIGHTNING following previously established protocols [7]. For pollen images, dry pollen samples were mounted in ProLong Glass (Life Technologies) and imaged using pollen's autofluorescent properties that are produced when exposed to UV.

ACKNOWLEDGMENTS

We thank Kean University, The Center for Aerobiological Research, and Kean's Center for Biological Imaging for providing pollen slides and imaging facilities. Work on this project was supported by NSF IUSE-HSI 2247157 and the primordial germ cell project was supported by NIH R15HD102960. This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

REFERENCES

[1] Apple Developer Forum. 2022. Are OpenGL and OpenCL supported on Apple silicon? Retrieved from https://developer.apple.com/forums/thread/694866.

[2] Aysha Alneyadi, Iltaf Shah, and Syed Salman Ashraf. 2019. An innovative bioanalytical research project course to train undergraduate students on liquid chromatography–

mass spectrometry, *Biochem. Mol. Biol. Educ.* 47, 3, 228–238.

[3] Sara E. Brownell, Mary Pat Wenderoth, Roddy Theobald, Nnadozie Okoroafor, Mikhail Koval, Scott Freeman, Cristina L. Walcher-Chevillet, and Alison J. Crowe. 2014. How students think about experimental design: novel conceptions revealed by in-class activities. *BioScience* 64, 2, 125–137.

[4] Alaina J. Buchanan and Ginger R. Fisher. 2022. Current status and implementation of science practices in course-based undergraduate research experiences (CUREs): A ystematic literature review," *CBE—Life Sci. Educ.* 21, 4, ar83.

[5] Ryan M. Cinalli, Prashanth Rangan, and Ruth Lehmann. 2008. Germ cells are forever. *Cell* 132, 4, 559–562.

[6] Arianna Dondi, Salvatore Tripodi, Valentina Panetta, Riccardo Asero, Andrea Di Rienzo Businco, Annamaria Bianchi, Antonio Carlucci, Giampaolo Ricci, Federica Bellini, Nunzia Maiello, et al. 2013. Pollen-induced allergic rhinitis in 1360 Italian children: Comorbidities and determinants of severity," *Pediatr. Allergy Immunol.* 24, 8, 742–751.

[7] Dominique A Doyle, Florencia N Burian, Benjamin Aharoni, Annabelle J Klinder, Melissa M Menzel, Gerard Carlo C Nifras, Ahad L Shabazz-Henry, Bianca Ulrich Palma, Gisselle A Hidalgo, Christopher J Sottolano, et al. 2023. Germ granule evolution provides mechanistic insight into drosophila germline development. *Mol. Biol. Evol.* 40, 8, msad174.

[8] Jennifer C. Drew and Eric W. Triplett. 2008. Whole genome sequencing in the undergraduate classroom: outcomes and lessons from a pilot course. *J. Microbiol. Biol. Educ.* 9, 1, 3–11.

[9] Corentin Guérinot, Valentin Marcon, Charlotte Godard, Thomas Blanc, Hippolyte Verdier, Guillaume Planchon, Francesca Raimondi, Nathalie Boddaert, Mariana Alonso, Kurt Sailor, et al.. 2022. New approach to accelerated image annotation by leveraging virtual reality and cloud computing. *Front. Bioinforma.* 1, 777101.

[10] John K. Haas. 2014. A history of the unity game engine. Retrieved from https://digitalcommons.wpi.edu/iqp-all/3207

[11] D. I. Hanauer, J. Frederick, B. Fotinakes, and S. A. Strobel. 2012. Linguistic analysis of project ownership for undergraduate research experiences. *CBE—Life Sci. Educ.* 11, 4, 378–385.

[12] Abdullah Aburiziza, Mohammed A Almatrafi, Aishah Saud Alonazi, Mawaddah Hani Zatari, Samah Ali Alqouzi, Rasha Abdulaziz Mandili, Wedad Taher Hawsawi, and Rehab Hejji Aljohani. 2008. The prevalence of nasal symptoms attributed to allergies in the United States: Findings from the burden of rhinitis in an America survey. *Allergy Asthma Proc.* 29, 6, 600–8.

[13] Elias Neuman-Donihue, Michael Jarvis, and Yuhao Zhu. 2023. FastPoints: A state-of-the-art point cloud renderer for Unity. Retrieved from https://arxiv.org/abs/2302.05002

[14] David Joiner. n.d. David Joiner – Computational Science Educator. Retrieved from https://joinerda.github.io/

[15] Santana Núñez, José Miguel, Trujillo Pino, Agustín Rafael, Ortega Trujillo, and Sebastián Eleazar. 2019. Visualization of large point cloud in unity. *Eurographics Tech. Rep. Ser.* Retrieved from http://hdl.handle.net/10553/70567

[16] C. Stefani, A. Lacy-Hulbert, and T. Skillman. 2018. ConfocalVR: immersive visualization for confocal microscopy. *J. Mol. Biol.* 430, 21, 4028–4035.

[17] Keijiro Takahashi. 2017. Pcx. Retrieved from https://github.com/keijiro/Pcx.

[18] Unity. 2023. Learn game development w/ Unity; Courses & tutorials in game design, VR, AR, & Real-time 3D. Retrieved from https://learn.unity.com/.

[19] Oliver J. Wilson. 2023. The 3D Pollen Project: An open repository of three-dimensional data for outreach, education and research. *Rev. Palaeobot. Palynol.* 312, 104860.

Scaling HPC Education

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

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

Kate Cahill
Ohio Supercomputer Center
khill42@gmail.com

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

David Joiner
Kean University
djoiner@kean.edu

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

John-Paul Navarro
University of Chicago, Argonne National Lab
navarro@anl.gov

Jeaime H. Powell
Texas Advanced Computing Center
jpowell@tacc.utexas.edu

ABSTRACT

Throughout the cyberinfrastructure community there are a large range of resources available to train faculty and young scholars about successful utilization of computational resources for research. The challenge that the community faces is that training materials abound, but they can be difficult to find, and often have little information about the quality or relevance of offerings. Building on existing software technology, we propose to build a way for the community to better share and find training and education materials through a federated training repository. In this scenario, organizations and authors retain physical and legal ownership of their materials by sharing only catalog information, organizations can refine local portals to use the best and most appropriate materials from both local and remote sources, and learners can take advantage of materials that are reviewed and described more clearly. In this paper, we introduce the HPC ED pilot project, a federated training repository that is designed to allow resource providers, campus portals, schools, and other institutions to both incorporate training from multiple sources into their own familiar interfaces and to publish their local training materials.

KEYWORDS

education, training, community engagement, survey

1 INTRODUCTION

We introduce the HPC ED pilot project, a federated training repository of vetted and tested training that is designed to allow resource providers, campus portals, schools, and other institutions to both incorporate training from multiple sources into their own familiar interfaces and to publish their local training materials. As a needs assessment prior to proposing HPC ED, the project team conducted a survey in October-November 2022 of providers of HPC training materials and related resources. The results [6, 18] showed that most respondents were both interested in, and able to, work toward community efforts to share and discover materials (see Section 2.) The HPC ED federated repository has been designed to identify and vet training resources from this broad set of offerings and to provide metadata and characterization that support successful discovery and utilization of training resources, and their incorporation into portals for research computing groups at universities, HPC centers, schools, domain-centered institutions and elsewhere. HPC ED also provides an API so that local centers will be able to include content identified in the repository and offer it to their local communities side-by-side with local offerings. In addition, local centers can upload training offerings that have been vetted and share with the broader computational science community.

HPC ED extends and enhances the ability of universities, departments, research computing groups, HPC centers, and domain-specific collaborations to discover and incorporate relevant and proven training materials into their own websites, portals, and science gateways, and to contribute to the federated repository. Leveraging of the federated training repository allows communities access to advanced CI-related training materials without the burden of creating and maintaining large sets of materials, and facilitate the professional development of individuals served by those institutions. In addition to providing sets of training materials that are commonly used for local activities, the federated repository will facilitate discovery of new materials that add to the overall catalog.

In this paper we describe our efforts to identify the needs of the HPC training community via our surveys (Section 2) which provides motivation for the HPC ED project. Section 3 presents an overview of the architecture of the Federated Training Catalog, our approach to developing the metadata, and an API for publishing and discovering the catalog content, and our commitment to maintaining community driven standards of quality. Section 4 describes our efforts to date to train our community to utilize the catalog by hosting workshops and hackathon and working with early adopters,

*Both authors contributed equally.

https://doi.org/10.22369/issn.2153-4136/15/1/8

who will provide important feedback for the Federated Training Catalog system. The paper concludes with a discussion that by our plans to help build a community of HPC trainers, we will help to ensure the longevity of the materials and metadata used for the catalog (Section 5).

2 SURVEY RESULTS OVERVIEW

The survey [6, 18] conducted in 2022 was conducted to learn if there was a benefit to improving how HPC training materials are shared and discovered. The survey showed that while community members are successful at finding materials, there are many barriers that make it difficult; the most cited reason was difficulty in finding materials at the right depth or level, as shown in Table ??.

Table 1: What Barriers Have You Encountered When Searching for Materials?

Barriers encountered	Responses
I can't find materials on the topic I need	35
I can find materials on the topic, but not at the depth or level I need	72
I find too many materials, and I can't effectively sort through them all	44
I am aware of specific appropriate materials, but search engines don't list them in the top results	26
Other	28
Total	93

In Figure 1, we explore whether respondents are both interested in making data discovery easier, and able to provide metadata. We see in the lower right quadrant that very few respondents want to make finding data easier, but lack the ability. The greatest number of respondents, in the upper right quadrant, have both the interest and ability. This shows great potential in the community moving forward with solutions. Altogether, the results imply that the community sees the potential for improving discovery of materials and many have the interest and ability to contribute to a solution.

3 FEDERATED ARCHITECTURE

We are building an architecture designed to enable organized and collective HPC training material sharing and discovery across the national and international research and education community. This community currently relies on two approaches for HPC training material discovery. First, larger research projects and education-focused organizations often have local training catalogs and discovery portals where members of their community discover materials selected for relevance and value to their community. These catalogs may contain locally produced training material or manually selected and entered information about external training resources. Commercial organizations, such as LinkedIn Learning, also provide their own catalogs and training discovery environments. Second, internet search engines like Google and streaming services like YouTube are often used to discover HPC training materials.

Figure 1: Results of two survey questions: Does your organization want to make it easier for the public to find your training and education materials? & Would your organization be willing and able to share your training and education materials in a public catalog by providing metadata about your materials in a standard format?

At the core HPC ED is a federated training catalog that (1) leverages and builds on the strengths and flexibility of organization-specific training catalogs and portals, (2) addresses the many deficiencies of search engines and streaming media services, (3) enables every individual looking for HPC education material, whether they are in an organization providing local training material or not, to discover training material across all federated training catalog participants,

HPC ED provides an API where resources can be published to the catalog and queries can be made to identify and retrieve content via the metadata system. Organizations that produce training materials and events will be able to publish into the Federated Catalog and reach a greater audience (for example MathWorks leveraging the catalog to publish documentation and events). Conversely, organizations that are curating appropriate materials for their local community can browse the catalog and add resources to their local portal. Organizations can both share their material and discover new materials to be added to local portals.

An overview of the federation process is presented in Figure 2. In this workflow, Training Developers/Curators use the API to submit a request to publish their materials and events into the Federated Catalog. Once approved, the information is entered into the catalog and made available to the public on the catalog site via the API query interface. This catalog of resources has the potential to encompass thousands of training resources and events that will be made available through an API that allows sites to add resources to their local portals and share resources with the catalog.

Using the HPC ED API, organizations seeking training are able to browse the catalog for materials and add them to their local portals and information systems. In this way, they can present complete training material sets without the difficulty of creating and maintaining them consistently over time. By leveraging a federated model, the training community can highlight the best possible training resources and emphasize competencies developed individually.

The base technology for the repository has been tested and is complete, and is currently running behind the https://software.xsede.org and the https://research.illinois.edu sites, both of which provide information about HPC software and research resources respectively. The former includes training materials on a much broader basis (including LinkedinLearning.com resources). HPC ED provides additional quality assurance of resources and integration into HPC learning roadmaps. The project team will establish a similar repository for training materials that collects information about the materials: location information, title, and metadata about content and topics. A feature of this system is that training materials remain in their original location and are discovered via the repository itself or from within an HPC Center website that uses the HPC ED repository via an API.

3.1 Metadata, Taxonomy and Ontologies

Working with existing community efforts to build a set of standardized minimal HPC training metadata is critical for publishing and discovering training information effectively. The Research Computing and Data community is active in this area, but the current lack of consistent metadata for HPC training is a major barrier to effective discovery. A standard taxonomy of HPC/CI training concepts, developed by an HPC training community, would make materials more easily searchable and discoverable, more readily adoptable and it will support the FAIR principles (Findable, Accessible, Interoperable, and Reusable) [11] for sharing of training materials. The HPC ED federated catalog builds on a foundation of standardized minimal HPC training metadata for publishing and discovering training information and will merge and standardize the HPC learning metadata from among our partnering organizations into a common metadata schema. [22] [15] [19] [20] [21] [9]

We propose that our effort begin with two types of metadata for elements in the federated repository: first, metadata that describes the training material, its access methods, and educational characteristics, including Title, Description, Authors, Publisher, Type, Language, Cost, Format, License, Target Group, Expertise Level, Certification details, and very importantly, Persistent Identifiers, Tags, or Keywords; and second, 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 to browse all available information and to access that training material. Additionally, we will start with the Research Data Alliance (RDA) “Recommendations for a minimal metadata set to aid harmonized discovery of learning resources” that addresses many of the use cases and needs around basic training sharing and discovery and supports FAIR practices" [12]. Note that once the community can agree on this metadata, we can begin exploring or connecting to other HPC based taxonomies or ontologies efforts.

A key outcome of this project is the formation of and participation in an HPC/CI training materials ontology community (e.g., an NSF ACCESS Affinity Group [2] [4]). This is described in Section 5 We are in the process of forming a Metadata Team of collaborators who will iteratively review community schema standards and identify discovery terms that need to be added to the schema, and post periodic schema upgrades for public use. This activity will be most likely organized through the ACCESS MATCH Affinity program.[4] The "HPC Ontologies and Metadata" Affinity group was created in Sept, 2023.[3]

Defining, categorizing, and standardizing the metadata is a significant effort. There are currently several efforts in the HPC/CI area that are building ontologies that describe the HPC ecosystem, but there is no single/primary metadata set, taxonomy, or ontology [7] [8] [14] [17] [23] [16] [24]. This is because of the complexities of the hardware, software, other components, organizations, local admin policies, etc. Where possible, we will identify and use existing, common metadata sets, taxonomies, and ontologies. Where needed, we will identify and add new terms to these existing ontologies and work with existing communities to update them or to develop the HPC/CI training materials ontology.

3.2 Sharing/Publishing Materials

To enable individuals, organizations, and projects with local training material to share and publish, HPC ED provides a secure publishing RESTful API. Using this API, any authorized organization will be able to automatically publish standard metadata from their local training materials catalog to the Federated Catalog. Affiliated organizations will implement automated publishing once, and rerun/synchronize frequently to refresh published information about their local training materials. Published training metadata will be stored in the Federated Catalog where it is aggregated with training metadata from other organizations. We will track who published each training element, perform basic quality checks, and inform the publishers of metadata quality issues.

We foresee challenges to the long-term maintenance of the project, such as how to handle material that is no longer being updated or is no longer accessible. We plan to address these challenges using quality assurance (see Section 3.4), and also by tracking when metadata was published and most recently refreshed. Working with an advisory board and the HPC community, the HPC ED project team will conduct periodic quality reviews of content to assess for relevance and correctness of materials. Publishers will be required to automatically refresh their metadata, and if they fail to do so within a configurable and reasonable interval, their metadata will be expired in the Federated Catalog. To support projects, organizations, and researchers that have training materials but do not have local training catalogs, we plan to coordinate with the ACCESS Support (e.g. MATCH) project [5] which already offers a way to publish training materials into the ACCESS reference material catalog.

3.3 Discovery of Materials

To enable organizations, projects, and individuals to discover published training material we will implement a federated training discovery/search RESTful API. This API will provide advanced

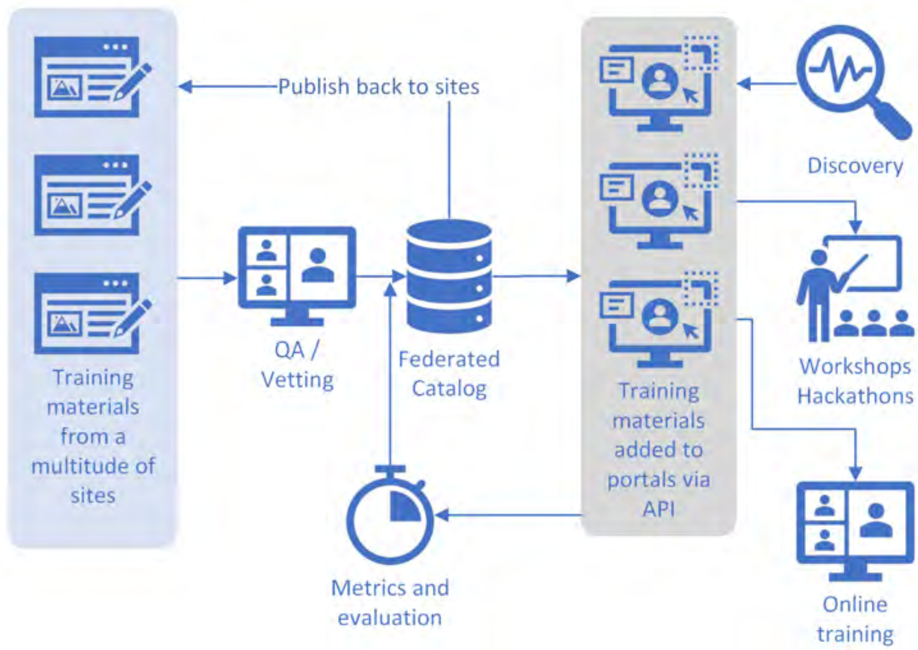


Figure 2: Training Developers/Curators use the API to publish their materials and events into the Federated Catalog. Organizations seeking training use the API to browse the catalog for materials and add them to their local portals.

search capabilities. For example, it will enable individuals to perform precise searches on specific metadata values, find materials from a particular organization, author, or targeted skill level, as well as perform more advanced key terms-based searches that rely on our related HPC training material ontology and taxonomy work. Advanced searching will ensure that the most relevant results are returned for search terms and enable relevance rankings based on known relationships between terms.

Projects and organizations with their own catalogs and training discovery portals or websites will be able to present their users with training materials published and shared by other organizations through the Federated Catalog. They will be able to do this by building into their websites the ability to query the catalog using our APIs and present their users with federated training search results.

To support projects, organizations, and researchers that do not have local training portals we hope to partner with the ACCESS Support (MATCH) project to implement a way for those individuals to discover training through a MATCH community-wide portal.

3.4 Quality Assurance

For the HPC training community, building a repository is not enough; we must also assure the accuracy of metadata of items shared through the federation. This includes verifying the status and nature of materials, validating their accuracy, and accrediting that metadata associated with the materials is appropriate [13]. Some of these processes can be fully automated, others assisted by artificial intelligence techniques, and some are simply human labor. QA will include processes that use automation where appropriate and build on the crowd-sourced nature of input from users of the repository and at workshops and hackathons through rating information from the community on the material in the catalog including a 5-star rating system, and monitoring the existence and uptime of links.

An additional component of the review process will center on material metadata. While review metadata that is collected will at the simplest level implement a star rating and user comments, additional feedback collected will focus on whether the metadata being displayed accurately reflects the catalog item. This can include descriptive information, such as author, title, and source, as well as audience level and content description. By sharing back this review information with catalog maintainers, we will provide a value add for adopters of the federation. Catalog maintainers will be able to opt-in to the review system.

4 PROJECT TRAINING

4.1 Workshops and Hackathons

The 2022-2026 National Science Foundation Strategic Plan [10] notes that development of human capital must begin with training that embeds generations of technical expertise followed by cultural/ community capital. For the HPC ED project, that means leveraging workshops in a phased manner to engage with MSIs, non-research, and academic research institutions, industry vendors, and research organizations to ingest training resource data needed to fill and maintain the catalog. Additionally, hackathons (time-scoped deliverable-driven events), and later Birds of a Feather

(BoF) sessions at targeted community conferences (see Section 5 will engage the collective human capital to maintain, evaluate, and identify community technical training needs. To clarify, the use of the workshops, hackathons, and BoFs for the purpose of collecting, maintaining, and building both the resources connections and fostering community engagement with the Federated Catalog each will have targeted outcomes within the three training delivery phases. These workshops, hackathons, and BoFs will be held virtually and, when applicable, in person.

4.2 Early Adopters

Early adopters will be encouraged to participate at all levels of development of the Federated Training Repository system, and will provide valuable feedback for needed changes. This group will serve as alpha clients and will define and/or redefine methodologies, processes, and initial user interface templates based on their required experiences. The reason this will be targeted for a workshop is to allow the development team rapid turnaround from critical path concerns identified by the user group. A diverse set of early adopters is essential to ensure broader engagement, current knowledge resources, and post- proposal funding resource opportunities respectively ensuring the culture of the project is inclusive. The project team has received letters of commitment from a number of sites willing to be eager adopters and help contribute to the holdings of the repository. These projects include enhancing coursework at MSI institutions, training Cyberinfrastructure Professional (CIP), hosting training catalogs at our local institutions, and collaborations with various ACCESS projects.

5 BUILDING AND SUSTAINING COMMUNITY

A community-wide project can only thrive when it has input, feedback, and use by the community. We have organized avenues for community engagement and communication on the project activities, which include our training program described above, and holding meetings with key stakeholders. A working group has been formed within the ACM SIGHPC Education Chapter to discuss metadata standards for sharing materials across all interested organizations. A BoF was held at PEARC23 to gather community input of what is needed to make existing training materials to be more findable, accessible, interoperable and reusable (FAIR)[21] for the whole community to benefit from them. Two key outcomes resulted from this meeting: (1) the participants wanted to meet as often as possible, synchronizing with other meetings; and (2) an affinity group within ACCESS [3] was recently created as a result of the discussions at this BoF. Future events include meeting at SC23, and the Science Gateways 2023 Annual Conference, and other meetings. For more information, visit the HPC ED website, located at: https://github.com/HPC-ED/HPC-ED.io.

We have a number of partners who confirmed their interest and intention in integrating their training resources as early adopters, who are committed to helping us to grow the repository and to collect feedback on the product and procedures. We share regular updates through the HPC ED Google Group mailing list and newsletter to allow those who want to know about activities can be kept updated. To join the mailing list, send an email to hpc-ed@googlegroups.com.

ACKNOWLEDGEMENTS

Our thanks for collaborative efforts and events go to the SIGHPC Education Chapter [1]. We want to acknowledge the use of several NSF funded resources and services including: the SDSC Expanse project (#1928224); the TACC Stampede System (# 1663578); NSF CyberTraining: Pilot: HPC ED: Building a Federated Repository and Increasing Access through Cybertraining (# 2320977); the Extreme Science and Engineering Discovery Environment (XSEDE) (NSF award #ACI-1548562); and the NSF ACCESS Track 3 Award: COre National Ecosystem for CyberinfrasTructure (CONECT) (#2138307);

REFERENCES

[1] 2023. SIGHPC Education chapter. https://sighpceducation.acm.org/
[2] ACCESS. 2023. ACCESS Advanced Cyberinfrastructure Coordination Ecosystem. https://access-ci.org/
[3] ACCESS. 2023. ACCESS HPC Ontologies and Metadata" Affinity group. https://support.access-ci.org/affinity_groups
[4] ACCESS. 2023. ACCESS Support Affinity Groups. https://support.access-ci.org/affinity_groups
[5] ACCESS. 2023. ACCESS User Support. https://support.access-ci.org/
[6] K Cahill, D Joiner, S Lathrop, S Mehringer, and A & Navarro, J-P & Weeden. 2022. Final Results: National Survey on Educational and Training Materials Repositories. https://www.cac.cornell.edu/about/pubs/Survey2022.pdf
[7] Gabriel G. Castañé, Huanhuan Xiong, and Dapeng Dong & John P. Morrison. 2018. An ontology for heterogeneous resources management interoperability and HPC in the cloud. Future Generation Computer Systems 88 (2018), 373–384. https://doi.org/10.1016/j.future.2018.05.086
[8] 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
[9] M. Emani and X Liao, C. & Shen. 2021. HPCFAIR: An Infrastructure for FAIR AI and Scientific Datasets for HPC Applications. https://custom.cvent.com/DCBD4ADAAD004096B1E4AD96F3C8049E/files/event/1fe48ee7ca1949c0b6ebd5f4a81c3d5f/04c21e2a6378405586b3e7ce51570e0b.pdf
[10] National Science Foundation. 2022. U.S. National Science Foundation 2022-2026 Strategic Plan. https://www.nsf.gov/pubs/2022/nsf22068/nsf22068.pdf
[11] 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
[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] David Joiner, Steven Gordon, Scott Lathrop, Marilyn McClelland, and D. E. Stevenson. 2005. Applying Verification, Validation, and Accreditation Processes to Digital Libraries. In Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL '05). Association for Computing Machinery, New York, NY, USA, 382. https://doi.org/10.1145/1065385.1065485
[14] R.M. Keller, D.C. Bemos, R.E. Carvalhol, D.R. Hall, S.J. Rich, I.B. Sturken, and S.R. Swanson, K.J. & Wolfe. 2004. SemanticOrganizer: A Customizable Semantic Repository for Distributed NASA Project Teams. , 5 pages. https://ntrs.nasa.gov/api/citations/20040084377/downloads/20040084377.pdf
[15] Richard M Keller, Daniel C Bemos, Robert E Carvalhol, David R Hall, Stephen J Rich, Ian B Sturken, Keith J Swanson, and Shawn R Wolfe. 2004. SemanticOrganizer : A Customizable Semantic Repository for Distributed NASA Project Teams. , 15 pages. https://doi.org/10.1007/978-3-540-30475-3_53
[16] C. Liao, P.-H. Lin, G. Verma, T. Vanderbruggen, M. Emani, and X. Nan, Z. & Shen. 2021. HPC Ontology: Towards a Unified Ontology for Managing Training Datasets and AI Models for High- Performance Computing. , 69-80 pages. https://www.osti.gov/servlets/purl/1832325
[17] L. Ma, J. Mei, Y. Pan, K. Kulkarni, and A. Fokoue, A. & Ranganathan. 2007. Semantic Web Technologies and Data Management. https://www.w3.org/2007/03/RdfRDB/papers/ma.pdf
[18] 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. The Journal of

Computational Science Education (2023).
[19] Shodor. 2023. HPC University Resources Page. Retrieved September 8, 2023 from http://hpcuniversity.org/resources/search/
[20] Diana Tanase, David A. Joiner, and Jonathan Stuart-Moore. 2006. Computational science educational reference desk: A digital library for students, educators, and scientists. D-Lib Magazine 12, 9 (2006), 0–4. https://doi.org/10.1045/september2006-tanase
[21] G Verma, M Emani, C Liao, P Lin, T Vanderbruggen, X Shen, and B Chapman. 2021. HPCFAIR: Enabling FAIR AI for HPC Applications. https://www.osti.gov/biblio/1830466
[22] Alexander Willner, Mary Giatili, Paola Grosso, Chrysa Papagianni, Mohamed Morsey, and Ilya Baldin. 2017. Using Semantic Web Technologies to Query and Manage Information within Federated Cyber-Infrastructures. Data 2, 3 (2017). https://doi.org/10.3390/data2030021
[23] L. Youseff and D. Butrico, M. & Da Silva. 2008. Toward a Unified Ontology. https://ieeexplore.ieee.org/document/4738443
[24] 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

Understanding Community Perspectives on HPC Skills and Training Pathways

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

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

ABSTRACT

The "Understanding the Skills and Pathways Behind Research Software Training" BoF session run at ISC'23 provided an opportunity to bring together a group of attendees interested in approaches to enhance skills within the Research Software Engineering community. This included looking at options for understanding and developing pathways that practitioners can follow to develop their skills and competencies in a structured manner from beginner to advanced level. Questions discussed included: How can we highlight the existence of different training opportunities and ensure awareness and uptake? What materials already exist and what's missing? How do we navigate this largely undefined landscape? In short: how does one train to become an RSE?

One of the interactive parts of this session was based around a live, anonymous survey. Participants were asked a number of questions ranging from their role in the training community to how easy they feel it is to find/access training content targeting different skill levels. They were also asked about challenges faced in accessing relevant content, combining it into a coherent pathway, and linking training content from different sources. Other questions focused on discoverability of material and skills that are most commonly overlooked. The number of respondents and responses varied between questions, with 24 to 50 participants engaging and providing 32 to 59 replies.

The goal of this lightning talk is to present findings, within the context of the community wide effort to make the training materials more FAIR - findable, accessible, interoperable and reusable.

KEYWORDS

Research Software Engineering, Training, Community, FAIR, Survey Results

1 INTRODUCTION

At ISC High Performance 2023, representatives from UNIVERSE-HPC [5], The Carpentries [1], CodeRefinery [2] and the HPC Certification Forum [4] organised a BoF session to discuss the skills and learning pathways behind Research Software Training. The goal was to trace current research software learning pathways and to understand the community perspective on difficulties with finding appropriate training content and creating routes for gaining specialist RSE skills. Beside a series of short presentations and discussions, the session also included an interactive, live and anonymous survey run through Mentimeter. Here we focus on the survey results.

2 METHODOLOGY

The session was attended by approximately 70 people with up to 50 session participants responding interactively to the questions in the room (the questions could only be answered when displayed on the screen during the workshop). There was no pre-registration for the session so we have no details of the profile of audience members or which of the participants in the room chose to respond to which questions. The session attendees were asked 9 questions, and the number of respondents and responses varied between questions, with 24 to 50 participants engaging and providing 32 to 59 replies. Questions were of two types, those requiring only a numeric score and those with the option for open-ended text responses.

3 RESULTS

The first question asked the audience to define their role(s) in the context of training, selecting as many options as appropriate. 47 attendees answered, selecting 94 options, with 16 using training materials as a learning resource, 28 being a trainer, 8 training others, 27 developing training, and 15 attending the session out of general interest. Therefore, more than half the attendees were actively involved in training development and/or delivery. The next two questions (see Fig. 1) asked attendees about the ease of finding/accessing training materials online and within their organisations.

The responses – scored between 1 (easy) and 5 (difficult) – show that many people feel finding beginner level resources within their own institutions is harder than finding equivalent resources online (difficulty level 2.4 vs. 1.5). The difference in the average scores for the difficulty of finding intermediate level training resources is smaller, with a difficulty of 2.9 within their organisation and 2.6 online. Finally, the average scores for the advanced level training are the same at 3.6. Overall, the attendees feel it's relatively easy to find or access beginner-level training online (1.5), while it is much harder to find or access similar content within their institutions - the score of 2.4 seems high. Intermediate training material is harder to find and access overall. There is no difference in the perception of how accessible advanced level training is - it's on average relatively hard to find both online and within the attendees' organisations.

The next question asked the attendees to score how difficult it is (1 - easy, 5 - difficult), in their opinion, to combine existing training materials into coherent pathways. The average score of 3.7 from the 49 respondents confirms that it is relatively hard to combine material from different sources. Another two questions (7 & 8) were focused on challenges associated with finding relevant

https://doi.org/10.22369/issn.2153-4136/15/1/9

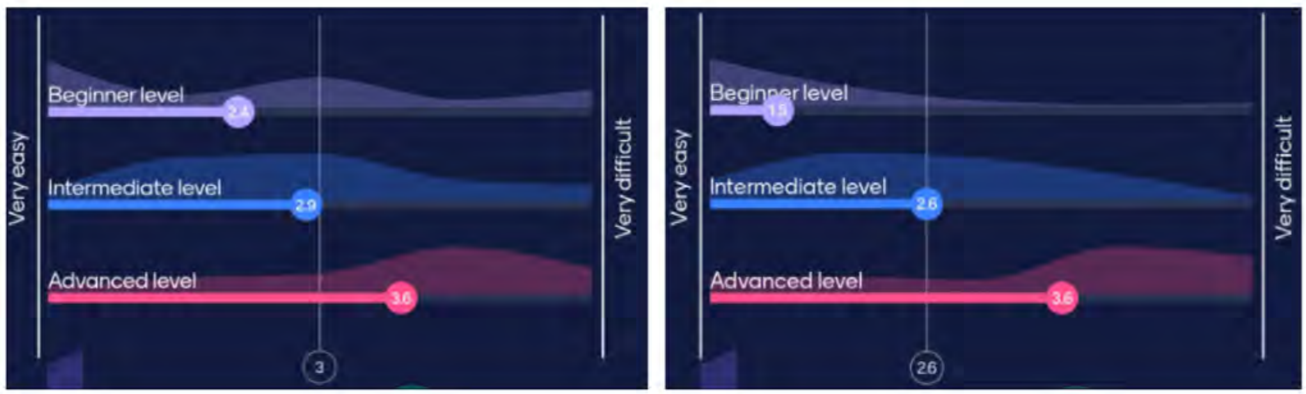


Figure 1: Audience's perception of how easy (1) or difficult (5) it is to find training content at beginner, intermediate and advanced levels within their organisations (left) and online (right). (Slides: Mentimeter.com)

training content and linking it into coherent pathways. 30 respondents provided 38 answers which can be divided into two main categories: accessibility, and appropriateness or relevance of available training materials. The accessibility challenges can be further divided – categories include: too many information sources, the financial cost of accessing content, access to hardware resources and outdated content. Examples from the relevance category include: trivial, too specific or irrelevant examples and difficulties with finding content at the right level and assessing its appropriateness. The main issues with linking content into pathways are centred around accessibility, incompatibility and teaching challenges. Accessibility issues include: broken links, disappearing pages, poor documentation, licensing and materials being either too specific or lacking useful examples. Some of the incompatibilities highlighted were: no consistent notation or terminology, competing standards (e.g. programming languages, research practices), incompatible or different packages/versioning. The teaching issues are related to overheads of producing good quality material not being recognised, time and expertise needed to verify online content, knowledge gaps and content overlaps.

Attendees were also asked about the most frequently overlooked research software skills, with 38 respondents providing 59 answers roughly fitting into categories that include: documentation, project management, communication skills, software engineering skills / best practices (e.g. testing, CI/CD, profiling, debugging), reusability and reproducibility, and personal skills like workload planning, critical thinking and independence. 30 respondents provided 40 answers on the question of how to make training content easier to discover. The main prevailing answer was providing a single point of access, in the form of curated repositories, a training portal or a centralised list, with coherent and consistent tagging. Other ideas included collaborations, organised user forums, raising awareness, dissemination in cooperation with educational institutions and advertising via social media. Additionally, the community should not only create incentives for improving skills, but ensure people can dedicate more time to learning/training.

Finally, attendees were asked about the skill(s) they would like to learn next. There were 35 answers from 20 respondents, mostly fitting into 6 categories: languages/models, GPU skills, parallel computing, ML/AI, quantum computing and algorithms, and general software skills.

All responses collected during this live survey are available as a digital artifact on Zenodo [3]. The authors hope the collected data will be useful to the whole community.

4 CONCLUSIONS

Overall, there are less training materials covering intermediate and advanced topics and beginner level material, despite its abundance, is not always easy to use and re-use. Linking materials from different sources to create a coherent learning pathway tends to be very difficult. The breadth of skills required by our community is constantly growing, making it impossible for every institution to provide the relevant training. Ensuring that training materials are created in a way that makes them easy to share and re-purpose should take priority over constantly developing new content that may well already exist elsewhere. The responses discussed here provide additional evidence and enforce the need for community-wide effort to make training materials more FAIR - findable, accessible, interoperable and reusable. This material and its associated presentation represent another step in that direction.

ACKNOWLEDGMENTS

The authors thank Anja Gerbes, Maciej Cytowski and Matias Jääskeläinen for presenting at the "Understanding the Skills and Pathways Behind Research Software Training" BoF, and the audience members for their participation in the session. WF and JC acknowledge support via the UNIVERSE-HPC project (EPSRC grant EP/W035731/1).

REFERENCES

[1] The Carpentries. 2023. https://carpentries.org/
[2] CodeRefinery. 2023. https://coderefinery.org/
[3] Weronika Filinger and Jeremy Cohen. 2023. Challenges in Research Software Training - survey results. https://doi.org/10.5281/zenodo.8321376
[4] HPC Certification Forum. 2023. https://www.hpc-certification.org/
[5] UNIVERSE-HPC. 2023. Understanding and Nurturing an Integrated Vision for Education in RSE and HPC. https://www.universe-hpc.ac.uk/

Intro to HPC Bootcamp: Engaging New Communities Through Energy Justice Projects

Mary Ann Leung
Sustainable Horizons Institute
mleung@shinstitute.org

Katharine Cahill
Ohio Supercomputer Center
kcahill@osc.edu

Rebecca Hartman-Baker
Lawrence Berkeley National Lab
rjhartmanbaker@lbl.gov

Paige Kinsley
Argonne National Lab
pkinsley@anl.gov

Lois Curfman McInnes
Argonne National Lab
mcinnes@anl.gov

Suzanne Parete-Koon
Oak Ridge National Lab
paretekoonst@ornl.gov

Sreeranjani "Jini" Ramprakash
Argonne National Lab
ramprakash@anl.gov

Subil Abraham
Oak Ridge National Lab
abrahams@ornl.gov

Lacy Beach Barrier
Sustainable Horizons Institute
lbeach@shinstitute.org

Gladys Chen
Oral Roberts University
gladyschenyx@gmail.com

Lizanne DeStefano
Georgia Institute of Technology
ldestefano6@gatech.edu

Scott Feister
California State University Channel Islands
scott.feister@csuci.edu

Sam Foreman
Argonne National Lab
foremans@anl.gov

Daniel Fulton
Lawrence Berkeley National Lab
DFulton@lbl.gov

Lipi Gupta
Lawrence Berkeley National Lab
lipigupta@lbl.gov

Yun (Helen) He
Lawrence Berkeley National Lab
yhe@lbl.gov

Anjuli Jain Figueroa
DOE Office of Economic Impact and Diversity
anjuli.jainfigueroa@hq.doe.gov

Murat Keceli
Argonne National Lab
kecelim@anl.gov

Talia Capozzoli Kessler
Georgia Institute of Technology
talia.capozzoli@ceismc.gatech.edu

Kellen Leland
Oak Ridge National Lab
lelandkr@ornl.gov

Charles Lively
Lawrence Berkeley National Lab
CharlesLively@lbl.gov

Keisha Moore
Sustainable Horizons Institute
kmoore@shinstitute.org

Wilbur Ouma
Ohio Supercomputer Center
wouma@osc.edu

Michael Sandoval
Oak Ridge National Lab
sandovalma@ornl.gov

Rollin Thomas
Lawrence Berkeley National Lab
rcthomas@lbl.gov

Alvaro Vazquez-Mayagoitia
Argonne National Lab
vama@alcf.anl.gov

ABSTRACT

The U.S. Department of Energy (DOE) is a long-standing leader in research and development of high-performance computing (HPC) in the pursuit of science. However, we face daunting challenges in fostering a robust and diverse HPC workforce. Basic HPC is not typically taught at early stages of students' academic careers, and the capacity and knowledge of HPC at many institutions are limited. Even so, such topics are prerequisites for advanced training programs, internships, graduate school, and ultimately for careers in HPC. To help address this challenge, as part of the DOE Exascale Computing Project's Broadening Participation Initiative, we recently launched the *Introduction to HPC Training and Workforce Pipeline Program* to provide accessible introductory material on HPC, scalable AI, and analytics.

We describe the *Intro to HPC Bootcamp*, an immersive program designed to engage students from underrepresented groups as they learn foundational HPC skills. The program takes a novel approach

https://doi.org/10.22369/issn.2153-4136/5/1/10

to HPC training by turning the traditional curriculum upside down. Instead of focusing on technology and its applications, the bootcamp focuses on energy justice to motivate the training of HPC skills through project-based pedagogy and real-life science stories. Additionally, the bootcamp prepares students for internships and future careers at DOE labs. The first bootcamp, hosted by the advanced computing facilities at Argonne, Lawrence Berkeley, and Oak Ridge National Labs and organized by Sustainable Horizons Institute, took place in August 2023.

KEYWORDS

high-performance computing, training, bootcamp, energy justice, diversity, workforce development

1 WHY AN INTRO TO HPC BOOTCAMP?

The U.S. Department of Energy (DOE) employs a mission-driven team science approach to basic and applied research, and the DOE national laboratories are renowned leaders in research and development (R&D) in high-performance computing (HPC). DOE's investments have pushed the growth of computational and data-enabled science and engineering as essential drivers of scientific and technological progress, in conjunction with theory and experiment. And yet, the DOE national laboratories remain somewhat of a hidden gem; they are not well-known among the emerging talent pool. This limited visibility, coupled with urgent workforce development and training challenges, call for interventions aimed at raising awareness and engaging a broader set of people in the mission-driven team science of DOE laboratories.

1.1 DOE HPC Workforce Challenges

The DOE labs face critical HPC workforce challenges, similar to those in science and engineering generally [28], but exacerbated due to the inter- and multidisciplinary nature of the work and the reliance on an understanding of advanced and high-performance computing [12]. As stated by the DOE Advanced Scientific Computing Advisory Committee (ASCAC) Workforce Subcommittee [5],

> "All large DOE national laboratories face workforce recruitment and retention challenges in the fields within Computing Sciences that are relevant to their mission. ... Future projections indicate an increasing workforce gap and a continued underrepresentation of minorities and females in the workforce unless there is an intervention."

Addressing these workforce challenges requires broad community collaboration to change computational science's culture and profile to match the changing demographics of the future workforce. Impactful DOE-wide programs, lab-specific regional initiatives, and activities in the wider computing community are making headway. Additional work is underway within the Broadening Participation Initiative [2, 11, 30] launched by the DOE Exascale Computing Project (ECP) [9, 26], which includes three complementary thrusts: (1) Establishing an *HPC Workforce Development and Retention (HPC-WDR) Action Group*, to foster a supportive and inclusive culture in DOE labs and communities; (2) expanding the *Sustainable Research Pathways (SRP)* [27] internship and workforce development program as a multi-lab cohort of students from underrepresented groups (and faculty working with them), who collaborate with DOE lab staff on world-class HPC projects; and (3) building an *Introduction to HPC Training and Workforce Pipeline Program* to provide accessible introductory material on HPC, scalable AI, and analytics to a broader audience. The Intro to HPC Bootcamp is one facet of the third thrust.

1.2 HPC Training Challenges

The idea of engaging a broader group of people earlier in their academic careers compelled us to rethink the existing DOE HPC training curricula. Broadening participation in this context translates to several dimensions of diversity: demographic, academic discipline, generational, and others. The challenge is in reaching new communities who have little exposure to HPC and may not know why or how HPC can address issues they care about. The future workforce will be comprised of Millennials, Gen Z, women, people of color, and others, who strongly value having a positive social impact on their communities. Further complicating the training challenge, the basics of HPC are not typically taught at the early stages of students' careers, and the capacity and knowledge of HPC at many institutions are limited [13, 15, 32]. Even so, such topics are prerequisites for advanced opportunities such as internships, the Argonne Training Program for Extreme-Scale Computing [1], the DOE Computational Science Graduate Fellowship Program [8], and ultimately for careers in HPC.

1.3 Raising Awareness and Engaging a Broader Set of People in Mission-driven Team Science

Thus, introductory HPC training must employ a fundamentally different approach that addresses the values and needs of this broader audience. The Intro to HPC Bootcamp's novel approach engages students in energy justice projects, where they learn HPC fundamentals using culturally relevant pedagogy, while gaining exposure to exciting career opportunities in computing sciences at DOE national labs.

We envision the Intro to HPC Bootcamp as an accessible on-ramp for students early in their academic careers, who will subsequently be prepared to complete internships and other HPC training offerings at the DOE labs and continue to graduate school for eventual careers in the computing sciences. As illustrated in Figure 1, the following sections describe our goals and approach, diversifying team science, bootcamp innovations, and related topics.

2 GOALS AND APPROACH

The first Intro to HPC Bootcamp [19], hosted by the advanced computing facilities at Argonne, Lawrence Berkeley, and Oak Ridge National Labs and organized by Sustainable Horizons Institute, was held August 7–11, 2023 at Lawrence Berkeley National Lab. Sixty students [20] worked in groups supported by fourteen trainers [22] and ten peer mentors [21] on seven energy justice projects [23] that explored issues related to the social impact of climate risk and resilience, solar power, sustainable cities, and energy usage. The bootcamp goals focused on engaging new communities in HPC in order to:

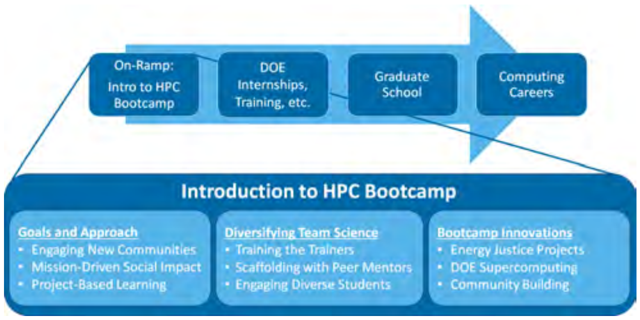


Figure 1: Overview of the Intro to HPC Bootcamp.

- Raise general awareness of the value, benefits, and rewards of work in HPC, especially at DOE labs;
- Engage more people from historically underrepresented groups, including those with no (or limited) HPC exposure;
- Include a broader set of disciplines in the HPC community; and
- Increase the available talent to work in HPC and computing sciences, where we have urgent needs for skilled researchers, research software engineers, systems administrators, and more.

Thus, as shown in Figure 1, the bootcamp can be considered as an accessible on-ramp to HPC training, internships, and ultimately careers in the computing sciences at DOE national labs.

2.1 Engaging New Communities

Changing the HPC workforce culture and profile to match the evolving demographics of the emerging workforce requires intentional approaches to engaging new communities and addressing the current state of very low numbers of people from underrepresented groups. In addition, many people in the developing workforce are driven by a desire to have a positive social impact on their communities [7]. The combination of these factors suggests employing culturally relevant pedagogy, including: creating a safe learning space, fostering a sense of belonging, challenging and engaging learners in problem solving, providing checkpoints and proctors, and enabling students to see where they are going throughout the learning process. This approach requires trainers to be intentional about building trust with learners—sharing their experiences and providing examples of breakthrough science accomplishments and projects that are led or contributed to by researchers with diverse backgrounds and demographics.

We therefore designed the Intro to HPC Bootcamp to provide a culturally relevant curriculum through social impact projects that are driven by DOE's mission and help to build foundational skills in HPC, scalable AI, and analytics. Accomplishing this required turning the traditional HPC curriculum upside down. Instead of teaching technology and its applications, the bootcamp uses project-based learning to engage students to answer social impact questions through energy justice projects while gaining hands-on experience using state-of-the-art computational and data science tools and techniques.

As shown in Figure 2, the five-day bootcamp framework begins by building community and establishing a friendly learning environment, motivating learners through challenging problems in science and society, and then introducing foundations of computation, HPC, and mathematics, while exposing students to the exciting R&D in high-performance computing at DOE labs. Capstone projects are introduced on the first day, and learners work throughout the week in small groups toward a final presentation at the conclusion of the bootcamp. The bootcamp agenda and plenary lectures are available via the bootcamp website [19]. This approach is motivated by the programs Advanced Computing for Social Change [14] and Computing4Change [6].

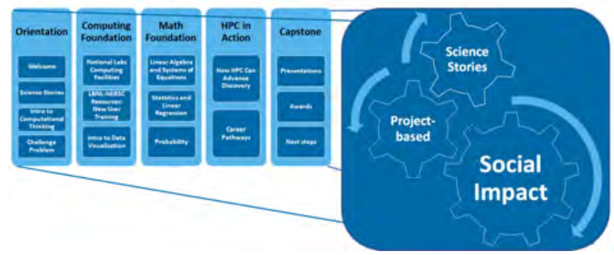


Figure 2: Framework for engaging new communities in high-performance computing: Solving problems with mission-driven social impact.

2.2 Mission-driven Social Impact

Research and development throughout DOE national laboratories are driven by the mission of ensuring America's security and prosperity by addressing its energy, environmental, and nuclear challenges through transformative science and technology solutions [39]. The topic of *energy justice* is an umbrella for a wide range of work with compelling social impact. The topics of projects in the Intro to HPC bootcamp were inspired by the Justice40 Initiative [25], whose implementation at DOE is led by the DOE Office of Economic Impact and Diversity [10]. By applying the tenets of justice (procedural, distributional, recognition, and restorative justice), energy justice [24] can promote policies that center community needs like reducing energy burden, avoiding disproportionate environmental impacts, ensuring equitable distribution of benefits of energy generation, creating opportunity for reliable access to clean energy, and encouraging democratic participation and enterprise creation in the energy system. DOE's Justice40 framework outlines eight such policy priorities that can better address the energy systems needs of overburdened communities.

Energy justice recognizes that the benefits of energy technologies have not been equally distributed across all Americans, often leaving out Black, Brown, Indigenous, and low-income communities. Underlying structural inequalities have resulted in development projects with higher rates of pollution, lower adoption of renewables, negative health impacts, and a higher energy burden in these communities. Energy justice is a complex issue with economic, racial, geographic, and social implications covering issues from energy affordability and access to infrastructure development. As we embark on an energy transition demanded by climate change concerns and fairness, we need to incorporate energy justice in the earliest stages of R&D and workforce development to enable a more just technology development and deployment.

2.3 Project-based Learning

Project-based learning in computational science [34] engages students through an active educational approach focused on real-world problems. It involves collaborative project work where students utilize computational tools to address complex challenges, fostering a deeper understanding of both subject matter and practical applications. This method introduces authentic issues mirroring research, industry, and societal contexts, motivating students by demonstrating the tangible impact of their skills. Project-based learning emphasizes critical thinking, problem-solving, and active participation—empowering students to drive their own investigations, make decisions, and take ownership of projects, thereby fostering a sense of agency and responsibility. Collaborative teamwork is highlighted, mirroring the collaborative nature of scientific research. Students gain proficiency in using computational tools through hands-on projects, with the iterative process fostering skills in inquiry, reflection, and resilience. Instructors and mentors offer personalized guidance, enhancing the learning experience, while presentations and communication of project outcomes refine students' ability to convey technical concepts. Overall, project-based learning in computational science equips students with practical skills, nurturing curiosity, critical thinking, and readiness to contribute to scientific and technological advancements.

3 DIVERSIFYING TEAM SCIENCE

Numerous studies have shown that diverse organizations, teams, and communities perform more creatively and effectively—and thus are demonstrably more innovative and productive [18, 33]. In the case of the Intro to HPC Bootcamp, diversity of team science took on a variety of forms. We employed a diverse set of trainers, organizers, peer mentors, and students on several dimensions of diversity, as described below.

Collaboration on the Intro to HPC Bootcamp is a partnership among experts in advanced computing, computational science, energy justice, workforce development, education, training, social science, and program evaluation. With collaborators from multiple DOE national laboratories, Sustainable Horizons Institute, the DOE Office of Economic Impact and Diversity, and academic partners, the team's breadth of experience has been essential for devising this first-of-a-kind program to introduce HPC at DOE labs in the context of mission-centered energy justice projects, with emphasis on engaging early-career students from underrepresented groups. The diversity of the bootcamp trainers, project leaders, and organizers is another facet of the diversity of this effort.

3.1 Training the Trainers

A pivotal step in developing the bootcamp program was hosting a "Train the Trainers" workshop in spring 2023, which served as a platform for promoting effective team collaboration. Through two days of in-person interactions, the workshop fostered strong interrelationships among the bootcamp leadership team and introduced the bootcamp concept to project leaders and trainers.

Workshop participants discussed the foundational principles of energy justice and inclusivity, enriched by a panel of organizers and mentors with extensive experience in social justice-focused computational science workshops. A key focus of the workshop was training the trainers in the art of modifying existing HPC training materials to suit the unique requirements of the bootcamp pedagogy, while also imparting effective coaching techniques to ensure students' successful engagement with the bootcamp's social challenge projects. The forum provided an opportunity for the team to deliberate on anticipated outcomes of the bootcamp, delineating the essential takeaways for students, along with post-bootcamp opportunities for their continued growth. Notably, the workshop stimulated the generation of innovative HPC project ideas aligned with the theme of energy justice, laying the foundation for a series of compelling energy justice projects to be explored in the bootcamp.

3.2 Scaffolding with Peer Mentors

Inspired by the approach used in the Advanced Computing for Social Change workshops, we engaged peer mentors [21] to provide support to students during the bootcamp. Peer mentors addressed technical questions, while also providing guidance on collaboration, presentations, and workshop expectations. Their contributions to the bootcamp were considerable, from providing a manual for mentors to use as a guide, to being a conduit of information between the students and the organizers. For example, peer mentors helped to foster a supportive and responsive environment by conveying to the project leaders and trainers questions that arose in conversations with students. The peer mentors facilitated discussions and tutorial sessions, while working closely with trainers and project groups. Most peer mentors were invited to apply to be a mentor, having had previous mentoring experiences; others were invited from the student applicants.

3.3 Engaging Diverse Students

Recruiting a diverse set of bootcamp participants early in their academic careers was central to the Intro to HPC Bootcamp success. Following is a description of the student profiles and background as well as our methods for recruiting and engaging them.

Demographics. The bootcamp partcipants were comprised of a highly diverse group of students on several dimensions. As shown in Figure 3, almost sixty percent of the sixty bootcamp participants were female, nearly a third were Hispanic/Latinx, thirteen percent had a disability, and nearly half were first-generation scholars (the first in their family to attend college). The racial distribution of students, also shown in Figure 3, illustrates an unusual representation for HPC—with Black or African Americans at twenty-eight percent (the largest racial group), followed by twenty percent Asian, twenty percent multiple race, and eighteen percent Caucasian.

As shown in Figure 4, the student academic profile included nearly eighty percent undergraduates, fourteen percent masters students, five percent students from community colleges, and just three percent doctoral students. Institution types ranged from eight Historically Black Colleges and Universities (HBCU), twenty Hispanic Serving Institutions (HSI), ten Asian American Native American Pacific Islander Serving Institutions (AANAPISI), three community colleges, thirteen liberal arts colleges, twenty-nine public/state universities, and thirteen high research institutions. Note that some institutions are counted in multiple types. In terms of academic focus, while approximately forty percent of students are studying

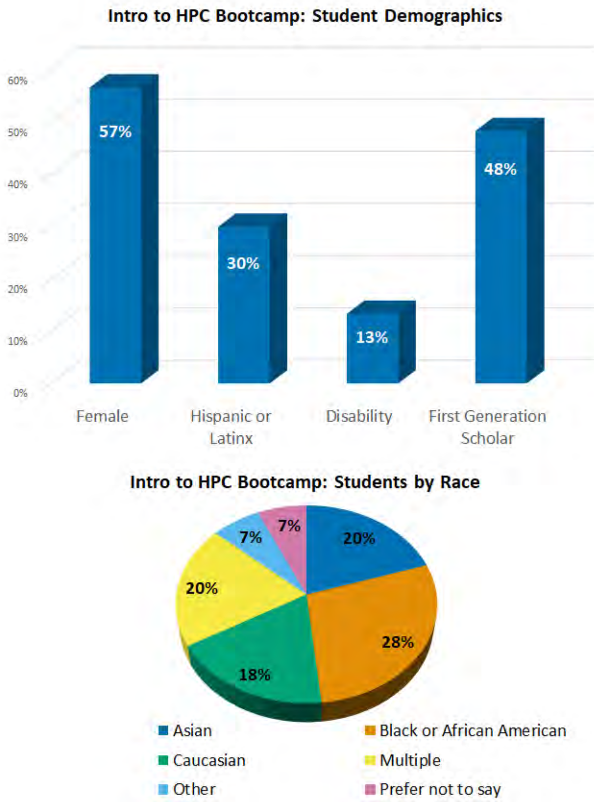


Figure 3: Demographics of bootcamp participants.

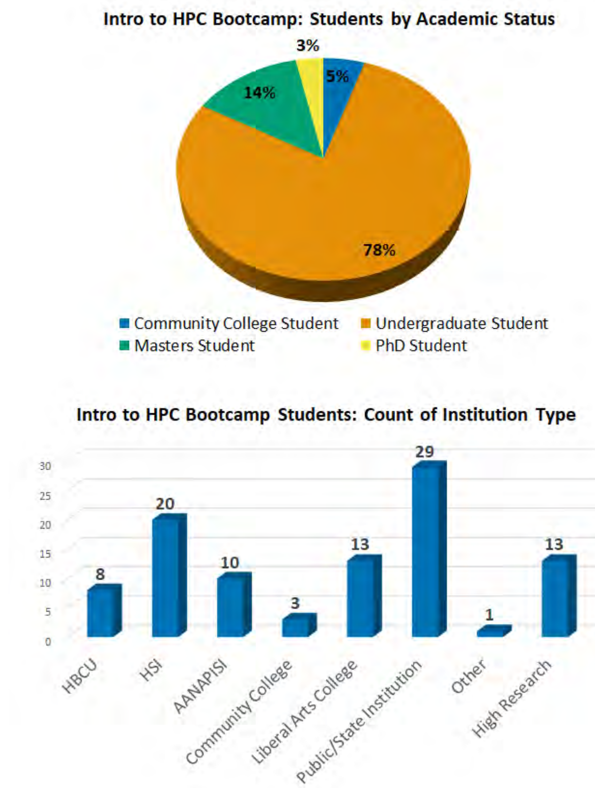


Figure 4: Academic status and institution type of bootcamp participants.

computer science, Figure 5 shows a wide distribution of domains in science, engineering, and mathematics.

Student recruitment. We achieved this demographic profile by broadly recruiting students from underrepresented groups to apply to the bootcamp, emphasizing the unique opportunity to advance understanding of *both* energy justice and HPC, while learning about career opportunities at DOE national labs. Although applications did not require CVs or letters of recommendation (with the goal of removing barriers for students to apply), applicants were asked to provide information about their motivation for participating in the bootcamp and their experience in computing, including R, Python, and HPC. This approach appears to have strongly resonated with the target community, as we received applications from several hundred students at US-based institutions, even though we could accommodate only sixty learners in this pilot session.

HPC skills. Typical skills needed by students to become interested and involved in HPC-based research include subsets of any number of the following: experience working with a programming language, exposure to parallel programming models and parallel thinking, exposure to techniques for visualizing and managing large data sets, experience with machine learning, and exposure to computational science. Requirements for the bootcamp were some experience in computing *and* an interest in energy justice topics. Because the bootcamp focused on introducing HPC, we did not expect applicants to have significant prior HPC exposure. We were able to engage a few applicants whose HPC skills exceeded the program scope as *peer mentors* in the bootcamp.

Student travel and stipend. The bootcamp fully supported travel, lodging, and meals for participants in the 5-day program. Participants received a modest stipend of $500 after completion of the bootcamp. We consider a stipend to be essential—making it possible for students who truly need the earnings from their usual jobs to sacrifice a week's employment in order to attend the bootcamp; without the stipend, some students could not participate.

4 BOOTCAMP INNOVATIONS

The centerpiece of the Intro to HPC Bootcamp was tackling energy justice projects using DOE supercomputing, while building community among the students, peer mentors, and lab staff.

4.1 Energy Justice Projects

Throughout the bootcamp, students worked in twelve small groups on projects, led by trainers and peer mentors. Bootcamp projects [23] were developed to teach HPC and AI tools in the context of wide-reaching energy justice problems, an example of our intentional design of melding social impact with technical skills development.

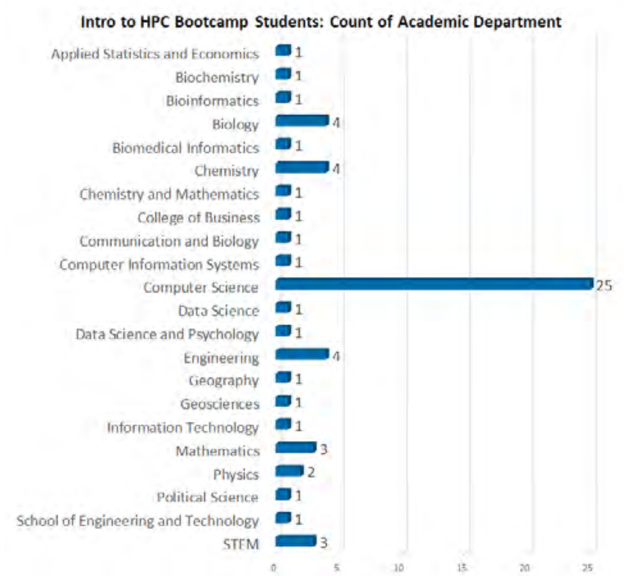


Figure 5: Academic departments of bootcamp participants.

The following are key themes in social impact and HPC technologies for the seven bootcamp projects.

- **AI-powered equity analysis of renewable energy**
  - *Social*: Equitable energy law, accessible energy justice info
  - *Tech*: Machine learning, large language models, web scraping
- **Energy justice analysis of climate data**
  - *Social*: Geographic climate disparities, variable correlations
  - *Tech*: Data visualization, big data wrangling
- **Solar power for affordable housing through computational design of low-cost/high-efficiency solar cells**
  - *Social*: Sustainable solar cells, renewable energy optimization
  - *Tech*: Machine learning, big data analysis
- **Energy cost for disadvantaged populations and methods of energy efficiency and optimization in computing systems**
  - *Social*: Power sustainability in HPC, equitable energy affordability and distribution
  - *Tech*: Top500 and Green500 energy efficiency ratings, HPC resource utilization
- **Understanding the impact of HPC center energy usage on low-income and minority populations**
  - *Social*: Power sustainability in HPC, equitable energy affordability and distribution
  - *Tech*: Code optimization, Top500 energy efficiency ratings, strategies for HPC resource utilization and allocation
- **Power outages and inequities in energy access for medically vulnerable populations**
  - *Social*: Weather impact on energy access, energy access for medically vulnerable populations
  - *Tech*: Data visualization, parallel wrangling of large data sets
- **Socioeconomics of power outages and heatwaves**
  - *Social*: Power outage data, weather impact on energy access, discrepancies in weather event impact across populations
  - *Tech*: Data visualization, statistical analysis of large data sets, MPI for python

To address the energy justice questions posited by the projects, bootcamp participants employed publicly available data sets, including climate data for national lab projects such as ClimRR [4] and EAGLE-I [35], HPC Top500 and Green500 power consumption statistics [16, 37], a dye-sensitized solar-cell device database [3], the HHS emPOWER Program [17], and U.S. census data [38].

4.2 DOE Supercomputing

Participants received access to the NERSC supercomputer Perlmutter [29], ranked 8th on the June 2023 Top500 List [36], providing them a suite of powerful computational tools to work on their energy justice projects. In contrast to training accounts usually provided for user training events, NERSC created a regular allocation project using the Director's Reserve Pool, so that students could continue Perlmutter access through the end of the current allocation year (until mid-January 2024). Students received detailed instructions about applying for a Perlmutter account and setting up multi-factor authentication. Multiple office hours were held prior to the bootcamp to help resolve any account and login issues, so that students were ready to work on projects immediately during the bootcamp. The students also had the opportunity to view Perlmutter in action during a tour of NERSC. Due to the large size of the student group and the noise level inside the machine room, a presentation about the NERSC facility and machine room took place before the tour, and an engaging Q&A session took place after the tour.

Students largely employed Jupyter Notebooks for bootcamp projects. NERSC staff prepared a bootcamp-specific kernel for use with the Notebooks, thereby streamlining access to necessary Python and other data analytics and plotting packages. GPU compute node reservations were made in advance; usage in a shared mode by multiple students was integrated smoothly upon launching the Notebooks.

A single GitHub repository provided easy access to all bootcamp project materials and presentations. The data sets and Jupyter Notebooks for bootcamp projects also were made available on the Perlmutter file system in each project's workspace.

4.3 Community Building

Community building served as an important theme and thread throughout the bootcamp, with the aim of supporting the goals of culturally relevant pedagogy: providing a safe learning environment, fostering a sense of belonging, and helping participants envision themselves not just "fitting in" but, also developing productive, rewarding careers.

Staff and leaders of DOE national labs spoke to the students about their career paths and motivations, oftentimes describing some of their challenges and nonlinear pathways. During a DOE staff panel, students learned about research applications of HPC and engaged in questions/answers about HPC careers.

Networking and relationship-building opportunities were built in formally and informally throughout the bootcamp, giving participants opportunities to connect with others sharing common interests in computing, energy justice, career paths, and other topics. Students met with other participants, with peer mentors who are a little further along in their academic journeys, and with DOE lab staff who have careers in HPC. During lunch each day, students and invited speakers discussed topics ranging from energy justice to whether to pursue a Ph.D., to work-life balance.

On the final bootcamp day, each of the twelve project groups presented an overview of the team, project goals, approach, insights gained during the bootcamp, and ideas for future work. Each bootcamp participant provided feedback to other groups on their presentation, and lively questions and discussion followed each presentation.

5 EVALUATION INSIGHTS

Bootcamp participants were surveyed both before and after the bootcamp by an external evaluator, employing instruments developed with a utilization-focused evaluation design [31]. The pre-survey, a formative tool to assist in pre-bootcamp planning, was distributed electronically to 60 participants, who all responded (100 percent). The post-survey included both qualitative and quantitative items related to participants' satisfaction with the bootcamp, as well as the perceived impact on participants' technical skills, understanding of energy justice, and future research or career attainment. The post-survey was distributed electronically to 60 participants at the conclusion of the bootcamp, and 54 responded (90 percent).

Preliminary feedback indicates that by focusing on project-based HPC for energy justice topics, the bootcamp captured the interest of a wide range of students from underrepresented groups, exposing them to the power of applying HPC to challenges in science and society. Further, participants indicated interest in pursuing a career in HPC or energy justice (Table 1). The bootcamp also introduced students to the impactful HPC research underway at DOE labs, providing pathways for future internships, education, networking, and more; at the conclusion of the bootcamp, participants indicated interest in pursuing a career at a DOE lab (Table 1).

Table 1: Interest in Careers

Key: 1: Not interested at all / 2: Somewhat interested / 3: Neutral
4: Interested / 5: Very interested

How interested are you in ...	1	2	3	4	5	Mean	SD
continuing to pursue HPC in a future career	1 (2%)	3 (6%)	4 (7%)	21 (39%)	25 (46%)	4.22	0.95
energy justice in a future career	1 (2%)	5 (9%)	8 (8%)	20 (37%)	20 (37%)	3.98	1.04
a career at a DOE lab	1 (2%)	1 (4%)	2 (4%)	16 (30%)	33 (61%)	4.44	0.88

In preliminary post-survey feedback, participants referenced the wide range of participant proficiency in technical fields such as coding, and some indicated it presented a challenge during groupwork. In order to balance the strengths and weaknesses of each group, participants suggested that groups should be intentionally comprised of members that capture a variety of skillsets. Additionally, participants identified time constraints as a challenge to the bootcamp, sharing that more time would have been preferable to complete their project. As some participants suggested, condensing the bootcamp lectures or expanding the length of the bootcamp, in general, would allow for more extensive groupwork.

6 LESSONS LEARNED AND NEXT STEPS

This pilot *Intro to HPC Bootcamp* represents our first iteration of an innovative approach to HPC training—emphasizing mission-driven social impact using project-based pedagogy and real-life science stories to prepare students for internships and future careers at DOE labs. Others could adapt and customize this model according to their specific needs and communities.

Looking ahead, the team is working to refine the bootcamp model and expand its reach through other modalities such as academic curricula and online learning, especially targeting students from underrepresented communities. At a high level, these measures could include, but are not limited to:

- **Partnering with faculty members:** To develop course modules for flexible use in campus settings, including through integration with subject-matter courses in the physical and social sciences
- **Creating an asynchronous online learning component:** To accommodate various schedules and learning styles and supplement the synchronous team and project-based components
- **Providing multiple offerings:** Working toward offering the bootcamp multiple times per year, considering regional emphasis
- **Broadening involvement of DOE lab staff:** To enhance the depth and breadth of expertise available to participants.
- **Offering training to progressively build HPC skills:** To bridge the gap for beginners (such as bootcamp alumni) and prepare them for more advanced HPC training and internships

Partnerships with faculty at colleges and universities, especially minority-serving institutions without on-site HPC research staff, offer a scalable avenue to connect with diverse students. We are piloting this strategy, and plans to expand are under way.

Acknowledging the diverse commitments and schedules of potential participants, we are considering a hybrid approach including an asynchronous learning component to complement the synchronous team-oriented core activities and prepare students who have less experience with programming. While asynchronous offerings on their own do not naturally support the community building aspect of the bootcamp, we believe that a a hybrid approach could enable participants to access introductory content at their own pace, while also engaging synchronously for group work on HPC projects and making personal connections with peers and lab staff.

To meet the urgent demand for HPC training and engagement, we are working toward offering the bootcamp multiple times per year, considering enhancements such as regional emphasis and avenues for more local lab involvement. Also, development of additional training modules would enable pathways for bootcamp alumni and others to progressively build HPC skills. Goals include reaching more participants and building a sustainable pipeline of talent for DOE national labs.

As we look ahead, these strategic directions will guide us in furthering the impact and accessibility of the Intro to HPC Bootcamp, as a core pillar of the ECP Broadening Participation Initiative. By embracing innovative approaches, emphasizing flexibility, and increasing our offerings, we aspire to cultivate a workforce that reflects the diversity of our society and leverages the power of HPC to address critical challenges in energy justice and beyond.

ACKNOWLEDGMENTS

We sincerely thank Linda Akli for insightful discussion about training strategy. We thank all Intro to HPC Bootcamp participants and peer mentors for their contributions to the program.

This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231 using NERSC award DDR-ERCAP0025729.

REFERENCES

[1] Argonne Training Program on Extreme-Scale Computing (ATPESC) 2023. https://extremecomputingtraining.anl.gov.
[2] A. Barker, D. Martin, M. A. Leung, L. C. McInnes, J. Ramprakash, J. White, J. Ahrens, E. Draeger, S. Fomby, Y. Ghadar, R. Gupta, M. Halappanavar, M. Heroux, C. Kelly, M. Miller, H. A. Nam, S. Peles, D. Rouson, D. Turner, T. Turton, D. Brown, and V. Taylor. 2021. A multipronged approach to building a diverse workforce and cultivating an inclusive professional environment for DOE high-performance computing. Response to the Stewardship of Software for Scientific and High-Performance Computing RFI, https://doi.org/10.6084/m9.figshare.17192492.
[3] Edward Beard and Jacqueline Cole. 2022. Dye-sensitized Solar Cell Database. (2022). https://doi.org/10.6084/m9.figshare.13516220.v1
[4] Center for Climate Resilience and Decision Science (ClimRR) 2023. ClimRR Data Catalog. https://disgeoportal.egs.anl.gov/ClimRR/?page=Data-Catalog.
[5] B. Chapman, H. Calandra, S. Crivelli, J. Dongarra, J. Hittinger, S. Lathrop, V. Sarkar, E. Stahlberg, J. Vetter, and D. Williams. 2014. DOE Advanced Scientific Advisory Committee (ASCAC): Workforce Subcommittee Letter. https://dx.doi.org/10.2172/1222711.
[6] Computing 4 Change 2021. https://www.sighpc.org/for-our-community/computing4change.
[7] Deloitte 2021. The Deloitte Global 2021 Millenial and Gen Z Survey. https://www2.deloitte.com/content/dam/Deloitte/mk/Documents/about-deloitte/2021-deloitte-global-millennial-survey-report.pdf.
[8] DOE Computational Science Graduate Fellowship (CSGF) Program 2023. https://www.krellinst.org/csgf.
[9] DOE Exascale Computing Project (ECP) 2023. https://www.exascaleproject.org.
[10] DOE Office of Economic Impact and Diversity 2023. https://www.energy.gov/diversity/office-economic-impact-and-diversity.
[11] ECP Broadening Participation Initiative 2023. https://www.exascaleproject.org/hpc-workforce.
[12] Roscoe Giles et al. 2020. Transforming ASCR after ECP. https://science.osti.gov/-/media/ascr/ascac/pdf/meetings/202004/Transition_Report_202004-ASCAC.pdf.
[13] Scott Feister and Elizabeth Blackwood. 2022. HPC Workforce Development of Undergraduates Outside the R1. *The Journal of Computational Science Education* 13 (Dec. 2022), 8–11. Issue 2. https://doi.org/10.22369/issn.2153-4136/13/2/2
[14] K. Gaither, R. Gomez, L. Akli, R. Mendenhall, M. Bland, S. Fratkin, L. Rivera, and L. DeStefano. 2017. Advanced Computing for Social Change: Educating and Engaging Our Students to Compete in a Changing Workforce. In *PEARC17: Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact.* https://doi.org/10.1145/3093338.3093391
[15] Steven I. Gordon and Katharine Cahill. 2020. The State of Undergraduate Computational Science Programs. *The Journal of Computational Science Education* 11 (April 2020), 7–11. Issue 2. https://doi.org/10.22369/issn.2153-4136/11/2/2
[16] Green500 List 2023. https://www.top500.org/lists/green500/2023/06/.
[17] HHS emPOWER Map 2023. Medicare At-Risk Populations by Geography. https://empowerprogram.hhs.gov/empowermap/.
[18] Dame Vivian Hunt, Dennis Layton, and Sara Prince. 2015. *Why diversity matters.* Technical Report. McKinsey and Company.
[19] Intro to HPC Bootcamp 2023. https://shinstitute.org/introduction-to-high-performance-computing-bootcamp.
[20] Intro to HPC Bootcamp: Participants 2023. https://shinstitute.org/intro-to-hpc-participants.
[21] Intro to HPC Bootcamp: Peer mentors 2023. https://shinstitute.org/intro-to-hpc-peer-mentors.
[22] Intro to HPC Bootcamp: Project leaders and trainers 2023. https://shinstitute.org/intro-to-hpc-project-leaders-and-trainers.
[23] Intro to HPC Bootcamp: Projects 2023. https://shinstitute.org/intro-to-hpc-energy-justice-projects.
[24] Kirsten Jenkins, Darren McCauley, Raphael Heffron, Hannes Stephan, and Robert Rehner. 2016. Energy justice: A conceptual review. *Energy Research and Social Science* 11 (2016), 174–182. https://doi.org/10.1016/j.erss.2015.10.004
[25] Justice 40 Program, DOE Office of Economic Impact and Diversity 2023. https://www.energy.gov/diversity/justice40-initiative.
[26] D. Kothe, S. Lee, and I. Qualters. 2019. Exascale Computing in the United States. *Computing in Science and Engineering* 21, 1 (2019), 17 – 29. https://doi.org/10.1109/MCSE.2018.2875366.
[27] M. A. Leung, S. Crivelli, and D. Brown. 2019. Sustainable Research Pathways: Building Connections across Communities to Diversify the National Laboratory Workforce. in Collaborative Network for Engineering and Computing Diversity (CoNECD), Washington, D.C..
[28] National Center for Science and Engineering Statistics (NCSES) Directorate for Social, Behavioral and Economic Sciences, National Science Foundation. 2023. Diversity and STEM: Women, Minorities, and Persons with Disabilities. https://ncses.nsf.gov/wmpd.
[29] NERSC Perlmutter System 2023. https://www.nersc.gov/systems/perlmutter/.
[30] S. Parete-Koon, M.A. Leung, J. Ramprakash, and L.C. McInnes. 2022. Exascale Computing Project's Broadening Participation Initiative. In *Ninth SC Workshop on Best Practices for HPC Training and Education (with SC22).*
[31] M. Q. Patton. 2000. Utilization-focused Evaluation. 49 (2000). https://doi.org/10.1007/0-306-47559-6_23
[32] 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
[33] David Rock and Heidi Grant. 2016. Why Diverse Teams Are Smarter. *Harvard Business Review* (2016).
[34] Namsoo Shin, Jonathan Bowers, Joseph Krajcik, and Daniel Damelin. 2021. Promoting computational thinking through project-based learning. *Discip Interdscip Sci Educ Res* 3, 7 (2021). https://doi.org/10.1186/s43031-021-00033-y
[35] Varisara Tansakul, Aaron Myers, Sarah Tennille, Matthew Denman, Alec Hamaker, Jonathan Huihui, Kyle Medlen, Karl Allen, Daniel Redmon, Supriya Chinthavali, Mark Coletti, Joshua Grant, Matt Lee, Dakotah Maguire, Scott Newby, Chelsey Stahl, Budhu Bhaduri, and Jibo Sanyal. 2023. EAGLE-I Power Outage Data 2014 - 2022. https://doi.org/10.13139/ORNLNCCS/1975202
[36] Top500 List 2023. https://www.top500.org/lists/top500/2023/06.
[37] Top500 Statistics 2023. https://www.top500.org/statistics/.
[38] United States Census 2019. County Population Totals: 2010-2019. https://www.census.gov/data/datasets/time-series/demo/popest/2010s-counties-total.html.
[39] U.S. Department of Energy 2023. https://www.energy.gov/about-us.

Cross-Institutional Research Engagement Network (CIREN): Initial Project Goals and Objectives in Support of Training, Mentoring, and Research Facilitation

Lonnie D. Crosby
National Institute for Computational Sciences (NICS)
University of Tennessee
lcrosby1@utk.edu

Gil Speyer
Research Technology Office
Knowledge Enterprise
Arizona State University
speyer@asu.edu

Marisa Brazil
Research Technology Office
Knowledge Enterprise
Arizona State University
marisa.brazil@asu.edu

ABSTRACT

The Cross-Institutional Research Engagement Network (CIREN) is a collaborative project between the University of Tennessee, Knoxville (UTK) and Arizona State University (ASU). This project's purpose is to fill critical gaps in the development and retention of cyberinfrastructure (CI) facilitators via training, mentorship, and research engagement. Research engagements include projects at the CI facilitator's local institution, between CIREN partner institutions, and through NSF's ACCESS program. This lightning talk will detail the training curriculum and mentorship activities the project has implemented in its first year as well as plans for its future research engagements. Feedback is welcome from the community with respect to project directions, best practices, and challenges experienced in implementing this or similar programs at academic institutions.

KEYWORDS

Cyberinfrastructure Facilitator, Training, Mentorship, High-Performance Computing, Machine Learning, Artificial Intelligence

1 INTRODUCTION

The Cross-Institutional Research Engagement Network (CIREN) is a collaborative project between the University of Tennessee, Knoxville (UTK) and Arizona State University (ASU) under National Science Foundation (NSF) grants OAC-2230106 and OAC-2230108. This project's purpose is to fill critical gaps in the development and retention of cyberinfrastructure (CI) facilitators via training, mentorship, and research engagement. Through these core activities of training and mentorship, CIREN aims to lower the barriers to the recruitment of CI facilitators and provide pathways to their continued development and retention via engagement in local, regional, and national research projects. CIREN recognizes the collaborative nature of the facilitator/researcher relationship and aims to enable CI facilitators to utilize their unique skills and interests to enable transformative research discoveries. For the purposes of the CIREN grant, a facilitator is an individual collaborating with researchers to enhance their research program, to enable discovery, and to produce innovative scientific impact via advanced cyberinfrastructure tools, skills, and technologies [2].

https://doi.org/10.22369/issn.2153-4136/15/1/11

2 TRAINING CURRICULUM

New CIREN facilitators are required to complete 256 hours of training and mentorship activities including both synchronous and asynchronous courses, a 6-month mentoring program, a final presentation, and elective courses in high-performance computing, machine learning, and/or artificial intelligence topics. This curriculum can be broken down into six main themes including

1) CIREN Training,
2) CyberAmbassadors Certification,
3) Community,
4) Project Management,
5) Mentorship Program, and
6) Elective Courses.

Table 1 shows each theme with its number of training hours and included topics.

The core material including the "CIREN Training" category is original content created and curated by the project. This material also serves a dual role as recruitment material for new CI Facilitators and Researchers. Other topics such as the "CyberAmbassadors Certification" leverages already existing content created by the CyberAmbassadors project [3]. CIREN includes trained CyberAmbassador Facilitators who deliver and curate this content. All other training content is provided by outside sources including LinkedIn Learning [6], campus HPC centers, and ACCESS [1]. Additionally, the "Elective Courses" in high-performance computing, machine learning, and artificial intelligence is provided by outside sources including Coursera [4], NVIDIA Deep Learning Institute [7], and tutorials provided at professional conferences such as the US-RSE [9], Super Computing [5], and PEARC [8].

3 MENTORSHIP PROGRAM

The CIREN mentorship program pairs new CIREN Facilitators with an experienced facilitator/mentor who will work with them on their first CIREN research engagement. This program guides and coaches the facilitator through the collaborative project pipeline including 1) the review of potential projects, 2) project intake interviews, 3) creation of the project work plan, 4) regular check-ins during the project, and 5) project reporting. The mentorship program culminates after the initial project with a final presentation on the project and its results.

Table 1: CIREN Training and Mentorship Curriculum Themes, Hours, and Topics

Theme	*Hours*	*Topics*
CIREN Training	3	Overview, Project Management, and Continuous Training and Development
CyberAmbassadors Certification	9	Communication, Teamwork, and Leadership
Community	4	Project Intake Interviews, Campus and ACCESS resources
Project Management	3	Project Management Foundations
Mentorship Program	217	Complete a 6-month research engagement with a facilitator mentor and final presentation.
Elective Courses	20	High-performance computing, machine learning, and artificial intelligence
Total	**256**	

4 RESEARCH ENGAGEMENTS

Research engagements include projects at the facilitator's local institution, between CIREN partner institutions, and through NSF's ACCESS program [1]. Engagements are expected to be six months in duration and chosen via a biannual competitive proposal process where facilitators review new research project proposals and choose those that align best with their skills and interests. CIREN project leadership also leverages this proposal process to identify needed skills and incorporates those in the future recruitment and/or continuing development of CIREN facilitators. In alignment with CIREN's focus areas, research engagements in any area of high-performance computing, machine learning, or artificial intelligence are welcome. The ideal research engagement project is oriented towards expanding, enhancing, or otherwise augmenting an existing research program with new capabilities. These capabilities can include new research directions or advance current research goals. Engagements can also include proof-of-concepts or other higher risk pursuits that may not align well within existing research, experience, or funding, but, if successful ,could be incorporated into the research program and future grants.

5 CIREN FACILITATORS

In addition to new facilitators going through the CIREN mentorship program, CIREN includes more experienced facilitators who engage in research engagements without a mentor. Additionally, these facilitators have opportunities to become mentors for new facilitators in the future. During this project, facilitators are expected to spend 20% of their time in continuing training and development. They spend the remaining 80% of their time on research engagement projects. Their overall project load is about 1 project per 25% of total project effort.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grants OAC-2230106 and OAC-2230108. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] ACCESS. n.d. Home – Access. Retrieved August 11, 2023 from https://access-ci.org/.

[2] ACI-REF. 2022. Key Terms and Definitions. Retrieved August 11, 2023 from https://aci-ref.github.io/facilitation_leading_practices/definitions/.

[3] Astri Briliyanti, Julie Rojewski, T. J. Van Nguyen, Katy Luchini-Colbry, and Dirk Colbry. 2019. The CyberAmbassador Training Program. 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 86, 1–6. https://doi.org/10.1145/3332186.3332218

[4] Coursera Inc. 2023. Coursera | Degrees, Certificates, & Free Online Courses. Retrieved September 8, 2023 from https://www.coursera.org/.

[5] The International Conference for High Performance Computing, Networking, Storage, and Analysis. n.d. Home • SC23. Retrieved September 8, 2023 from https://sc23.supercomputing.org/.

[6] LinkedIn Corporation. 2023. Online Learning Platform for Businesses | LinkedIn Learning. Retrieved September 8, 2023 from https://learning.linkedin.com/.

[7] NVIDIA Corporation. 2023. Deep Learning Institute and Training Solutions | NVIDIA. Retrieved September 8, 2023 from https://www.nvidia.com/en-us/training/.

[8] PEARC Conference Series. 2022. PEARC Conference Series - Practice and Experience in Advanced Research Computing. Retrieved September 8, 2023 from https://pearc.acm.org/.

[9] US-RSE. 2023. US-RSE'23 | The First US-RSE Conference Ever! Retrieved September 8, 2023 from https://us-rse.org/usrse23/program/tutorials/.

Data Analytics Program in Community Colleges in Preparation for STEM and HPC Careers

Elizabeth Bautista
NERSC
Lawrence Berkeley National Laboratory
ejbautista@lbl.gov

Nitin Sukhija
Department of Computer Science
Slippery Rock University of Pennsylvania
nitin.sukhija@sru.edu

ABSTRACT
Students in community colleges are either interested in a quick degree or a skill that allows them to hop onto a career area while minimizing debt. Attending a four-year university can be a challenge for financial costs or academic reasons, and acceptance can be competitive. Today's job market is challenging in hiring and retaining diverse staff. More so within the High Performance Computing (HPC) or a government laboratory. Industry offers higher salaries, potentially better benefits, or opportunities for remote work, factors that contribute to the challenge of attracting talent. At the National Energy Research Scientific Computing Center (NERSC) at Lawrence Berkeley National Laboratory, site reliability engineers manage the HPC data center onsite 24x7. The facility is a unique and complex ecosystem that needs to be monitored in addition to the normal areas such as the computational systems, the three-tier storage, the supporting infrastructure, the network and cybersecurity. Effective monitoring requires the understanding of data collected from the heterogeneous sources produced by the systems and facility. With so much data, it is much easier to view the data in graphic format and NERSC uses Grafana to display their data. To encourage interest in HPC, NERSC partnered with Laney College to create a Data Analytics Program. Once Laney faculty learns how to teach the classes toward a certificate program, they fill a need for their students to build the skill in data analytics toward a career or to continue toward a four-year degree as transfer students. This also fills a gap where the nearby four-year university has a long waitlist. This paper describes how NERSC partners with to create a pipeline toward a data analytics career. This is the follow-up program to creating a pathway into HPC and Science, Technology, Engineering and Mathematics (STEM) [3].

KEYWORDS
Site Reliability Engineer, HPC Education, HPC Training, Diversity, Inclusion, STEM, community college, data analytics.

1 INTRODUCTION
Everything today requires data or a visual representation of the data. According to a 2020 study at the Massachusetts Institute of

Technology, the demand for "data scientists" is expected to grow in the future as more data is being accumulated. However, even academia is confused because there is no standard on how to train a workforce in this area. Is Data Science a series of principles, a skillset or an "umbrella term" that encompasses a series of required expertise? Is it a specific discipline? What are the jobs associated with this type of degree [7].

ttps://doi.org/10.22369/issn.2153-4136/15/1/12

At NERSC, the Operations Technology Group (OTG) staff are the 24x7 onsite site reliability engineers who are the first responders to anything that occurs in the data center. Although the position does not necessarily require a college degree or certifications, the job description does require knowledge of system administration of HPC systems, local and wide area networking, a three-tier onsite storage and data center facility management at minimum for staff to be successful.

Further, one of the Laboratory's mission statements is widening Diversity and Inclusion in these areas and has programs that support internship programs in underserved communities as well as recruit staff from a wide area of disciplines with the idea that these skills can be transferable into science research. However, according to a 2022 Lab study of our workforce demographics, we continue to see a small percentage of staff who are underrepresented such as Black/African American, American Indian/Alaska Native, Asian, Hispanic or Latino, etc. We see an even smaller percentage of women in these demographics especially within the Lab Senior Leadership roles. The numbers demonstrate a compelling story [5, 9].

NERSC is one of the largest facilities in the world devoted to providing computational resources and expertise for basic scientific research. NERSC currently supports close to 10,000 users globally across almost 1,000 scientific projects [1]. As NERSC moves toward exascale, how will we increase diversity and inclusion percentages in this area and also continue to recruit the much needed talent to fill positions that are also in high demand in our neighboring Silicon Valley?

Part of the solution was to partner with the local community college in the neighborhood to create a potential pipeline by influencing students' curriculum and education with much needed support for these underserved students as well as providing a place for them to practice their skills hands-on.

This paper documents the process for creating a pipeline where data analytics students can get an education, are provided training and hands-on projects [2]. Section 2 will explain the academic program itself and the creation of the internship program. Section 3 provides the background on the type of training they will be provided by the internship program, in addition to the program at school. Section 4 will provide logistics such as sourcing funding streams and our experiences in the first year of the program. Section 5 will provide case studies of positive outcomes. Section 6 will provide lessons learned, future work and final thoughts to continue the program.

2 BACKGROUND
With so much information and a list of the different type of programs in academia, we decided to focus on the type of training needed by the employers in the San Francisco, Oakland, Berkeley, and Alameda areas, the areas surrounding the school. A survey from the school's industry partners resulted in a need for more training in these skill areas: Data Analyst, Data Engineers, and Database Administrator. Although the survey included other skill areas such as Machine Learning Engineer, Data Scientist, Data

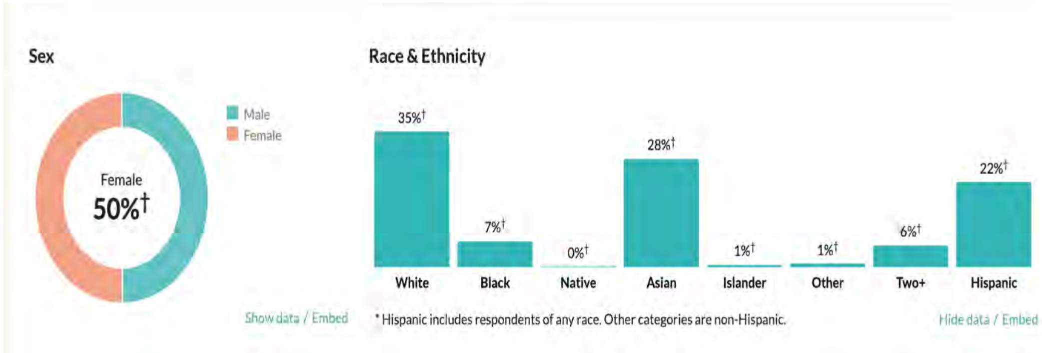


Figure 1: Workforce demographics survey, Lawrence Berkeley National Lab, 2022

Architect, Statistician, Business Analyst, and Data and Analytics Manager, the school felt that the latter required more foundation than they could provide, therefore, they focused on the three skillsets first mentioned.

A second survey found that although these jobs are in high demand by the area employers, it requires a very specific skill. For students, this could be daunting even just imagining undertaking such training, especially when they have low confidence in their science and math skills. However, these skills are obtainable with the right education, hands-on training and support of the students from peers and faculty. After all, we are training the next generation of data analysts.

Because we also had a focus on diversity and inclusion, we wanted to recruit students into the program who are underrepresented in the data science career areas. In the case of NERSC, the area they wanted to serve was the San Francisco, Oakland, Berkeley, Alameda Metro area with an underrepresented population of 4,579,599. According to a workforce survey by the Human Relations department at LBNL in 2022, less than 10% of the Bay Area workforce are within the underrepresented population of Black, Pacific Islander, Other, Native American and two plus who work in STEM fields. Within these groups, there is still only 50% or less women, especially in the senior management levels [5] See Figure 1.

We were also hoping to fill a gap in the closest four-year university offering a data science degree had a wait list of at least 200 students in the last five years that includes the pandemic. Students who are accepted as freshmen into the four-year university could be accepted into the data science program; however, due to the wait list, a community college transfer student would also have to join the waitlist. Filling this gap through teaching the classes at the community college level could give these transfer students a chance to enter the program in their junior year.

2.1 Train the Teacher

One of the early steps is to train the faculty on how to teach various classes in the program. Through various contacts with the University of California, Berkeley (UCB), Electrical Engineering and Computer Science (EECS) department, we decided that the best training to get is from the trainer themselves. Therefore, the faculty who have committed to teach at the community college, entered various UCB extension programs to go through the classes themselves.

Three faculty members started the program in the prior spring semester 2022 before we launched the classes in the fall, September 2023. By the fall program start, they would have taken two semesters, spring and summer, of classes and should be able to teach a series of classes in the fall.

2.2 Create the Program

The program itself would not be finalized until the middle of the summer prior to the launch. This is due to funding which I will cover in Section 4. Parts of the program that we would submit toward a certificate are already being taught and we will just integrate the data science classes.

As such, this is the program that was created and presented beginning September 2023, noted by Figure 2. Though the data science foundation classes would not be taught until the end of their degree program, most students who added the data science program would have already completed most of the pre-requisite classes in the prior year. Thus, they were second year community college students. Most of these students came from the cohort of students from the previous year who entered another program to prepare them for STEM classes.

Database Programming with SQL
Introduction to Microsoft Excel for Business
Introduction to Computer Programming
Introduction to Computational Thinking with Data
Introduction for Computer Science
Microcomputer Assembly Language
Introduction to Artificial Intelligence and Machine Learning
Object Oriented Programming Using C++
Java Programming or Python Programming
Data Structures and Algorithms
Structure and Interpretation of Computer Programs
Foundations in Data Science
Introduction to Statistics

Figure 2. Data Science certificate program

These classes vary from three to five semester units. The program requires a minimum completion of 27 units - 31 units to acquire an Associate of Science (AS) degree with a specialization in data science. These classes are also transferable into the University of California system as credits toward their freshman and sophomore years. In the University of California, Berkeley, it would be acceptable as pre-requisites to classes they would need to take to earn a Bachelor of Science (BS) degree with a concentration in data science.

2.3 Support of Students

Understanding that some of the students who enter the program may have been challenged in math and science through the high school level, the college decided that we need to provide some tutorial support for the students. Thus, one faculty member was transferred from the previous STEM program into this program to assist in supporting the students with their homework, concepts explanation, any lab they need and generally with their homework. In addition, upper classmen students from the math and science departments were recruited to help during the tutorial lab. Industry partners were also asked to volunteer time to help the students at their discretion. Due to the wide variety of volunteers who assist in the tutorial lab, the support program can be available from 9:00 a.m. through 9:00 p.m. Monday through Friday, depending on the needs of the student and availability of tutors. After 5:00 p.m., tutorial times are on a scheduled basis.

2.4 Internship Program

The internship program was modeled after the prior program for STEM preparation. We used most of the same industry partner employers. The students work 40 hours during a thirteen-week summer program that required for the employer to provide a hands-on training class for at minimum forty hours. Further, they provide a project where the student can do the work that they just learned to do in the eleven weeks after. In the last week, the students need to prepare a poster presentation that will be explained to several of the company staff. In lieu of the poster, they will submit a paper for review to a conference within one year. The preparation of the submission will be approved by the student's supervisor and faculty advisor. The employer commits to send the student to the conference if the paper is accepted for presentation. More on how this travel will be funded in Section 4.

3 TRAINING BY THE EMPLOYER

Each employer we recruited committed to providing the necessary training, a very specific skill in the data science area that a student can learn within forty hours and put that training into practice on a project that they will work on for the next eleven weeks in their organization.

For NERSC, they participated in the programs pilot cohort of students. In the summer of 2023, the students from the program took a class on Grafana [6], an open-source software that visualizes data for a time-series database into graphs and visualizations that can be queried, alerted on and explore metrics and logs. NERSC uses Grafana to do exactly this from the heterogeneous data in their data [10] warehouse called the Operations Monitoring and Notification Infrastructure (OMNI) [4].

NERSC hired two trainers from Grafana Enterprise to teach the students how to create visualizations from the data. Each student was assigned a mentor and had a project that would create

Figure 3: Slurm dashboard for perlmutter

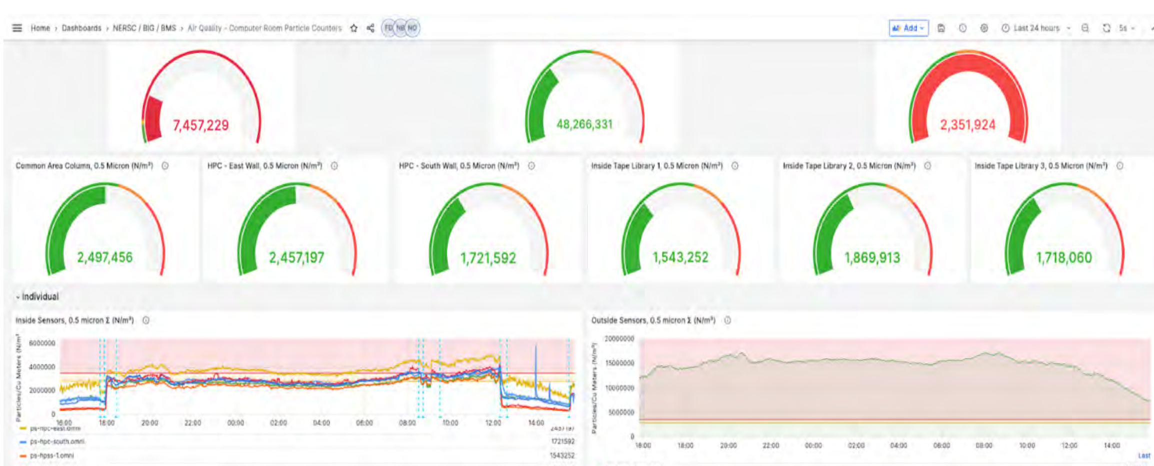


Figure 4: Dashboard of particle count in tape libraries

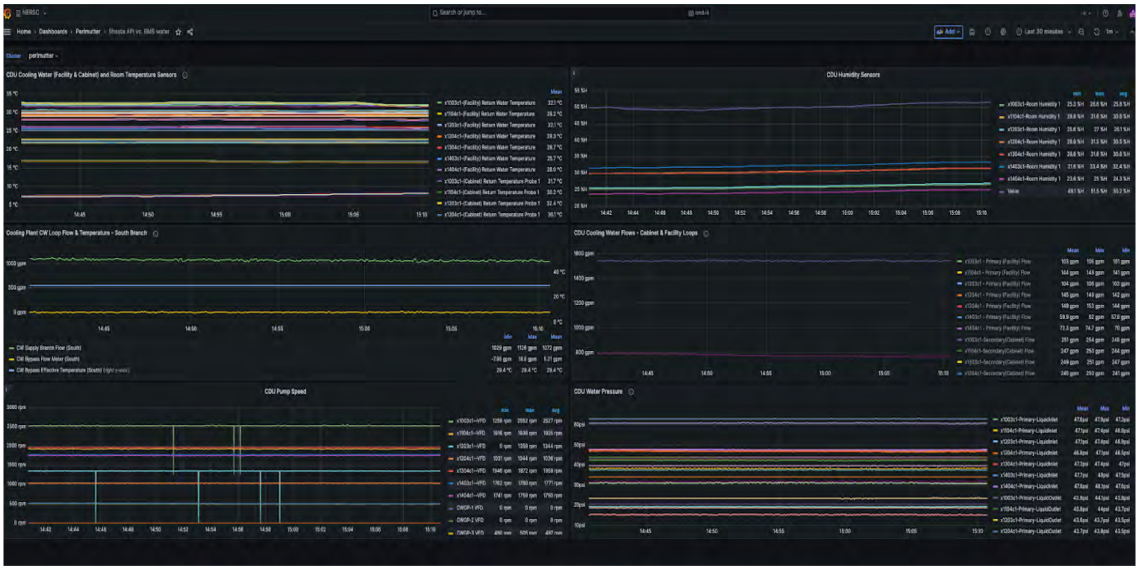


Figure 5. Pipe water pressure

a series of graphs to showcase the data, to "have the data tell a story". This supplemented hands-on training as well as their academic background formed a foundation for a successful experience for the students.

4 FUNDING AND OTHER CHALLENGES

The program needed to be funded. Though some activities already had their own funding streams, such as the classes already being taught, the new classes needed basic funding, especially for the tutorial lab.

Luckily, there was a grant that the school was able to obtain that would cover three years of the program from inception. Some of the funding also provided for salaries, travel and housing costs for students who are out of the area for the internship program. This only used up less than 10% of the funding in the first year since the main focus was to assist the metro area students. However, beyond three years of funding, it will be a challenge to continue unless they can secure more funding which is the challenge because, it was easy to acquire startup funding, but without proof of success, funding can be a challenge.

However, first we have the challenge of our program being accepted for certification. At the time of the conference presentation of this work, we were only in the first semester after launch. To acquire certification for the program, we must show a success rate of at least two semesters and have a plan for the second year of the program. We are now currently in the second semester and it looks like it could be successful and there is a plan for the third and fourth semesters.

Some of the faculty teaching the program had a difficult time helping the students understand the concepts and they were not quite confident that the students will successfully pass. After a few weeks, the faculty members were encouraged that the tutorial program seems to be helping a lot, therefore, they felt much better by the end of the fall semester to see that they had an 85% pass rate of the classes.

Though the program had some faculty, as the program expands, the college will need more faculty to teach classes. At this point, the administration is currently challenged to recruit other faculty to commit to taking the extension classes and be able to teach in the program. They may have to hire from external to the college however, funding will be an issue. There is a challenge of recruiting more employers to participate in the internship program. While the program was able to support the salaries, travel and housing, that part of the funding was only available in year one and not for year two or three. Though they still have a commitment from the existing employers, as the program expands, more employers need to participate in providing internships. It is a continuing challenge to entice an employer to hire from the community college instead of the four-year university. Further, with the Metro Area having a reputation for being a tech area, most employers expect to hire four-year university students. The pandemic has changed that for the Metro Area, and we are attempting to educate the industry that investing in a community college student can be far rewarding with potential longevity especially in the underrepresented groups who are from the area.

In a positive note, the students who participated in the pilot program have said that it was the most rewarding experience that they've ever had. Participating in the internship was most valuable and they feel they now have a skill that they offer to employers.

5 CASE STUDIES

This section will discuss the positive outcomes of three students in the program at NERSC.

Student #1 is the first student who has entered college from an immigrant family. He has taken 2 years of courses in a community college and took classes that were part of the pilot program. After Grafana training, he was able to create various dashboards, one of which was used during the acceptance of NERSC's Perlmutter system, as noted in Figure 3.

While it may be difficult to see the dashboard clearly because it is so full of information, there are two graphics that show system utilization, noted by 86% and where you see the 4, these graphs show that jobs were running on the nodes. There are other details but this is a good summary of what the SRE staff need to see on the system.

Student #2 was a student who was also in his third year of community college and was preparing to get a transfer to a four-year university. Like the first student, he was also the first college student in the family from immigrant parents.

The Metro Area can experience high smoke levels from fires during the summer but can also extend into early fall. Because of the design of the data center, we have no chillers, instead use evaporative cooling system, hot air recycling and air movement to keep the data center cool, therefore, any pollutants external to the facility can impact the air within the facility. Such as the case in late September 2023 where there were a series of fires more than fifty miles away, and high winds blew smoke toward the data center impacting our tape libraries, as noted in Figure 4.

You can see the top level of the dashboard that shows three circles and two are red. This shows that two out of three libraries have a high particle count and is in the critical stage. At this point, library 3, to the right, was shut down to save moving parts, since smoke particles can potentially scratch the moving parts and void the warranty. That said, we needed to find out when this high particle count occurred because it happened very quickly.

The student created the third panel and fourth panel of graphs. The left one shows when the spike of high particles occurred by time in comparison to the panel on the right, that shows high particle count external to the facility. In this way, we can see when the high particle count occurs, put an alert on it and be able to monitor the graphs so we can mitigate it in the data center.

Student #3 was a female student who previously graduated from a four-year university with a marketing degree. However, after one year, she has not been successful in finding a job in her area due to the intense competition post pandemic. Therefore, she decided to enter a community college data science program to learn new skills but minimize her debt.

After the training program she created these dashboards that show the speed of the water in the pipes that help cool the facility on two loops that cool the ambient air around the equipment, as noted in Figure 5.

While the dashboard is difficult to read because it is dense with information, the important graphic is that all these lines are horizontal. Should there be a spike, then the facility will have an issue with their evaporative cooling. Pipes that flow water go through the HPC system and they need to be at constant speed, pressure and temperature. When the speed varies or the pressure varies or the temperature rises, an alarm will alert the SRE on duty who will look at these graphs to find more information to mitigate the situation.

In the three students' experiences, the internship and training helped the grow professionally and develop a new skill. Further, these graphs were put into production and are used daily by the SRE's who work in the data center control room 24x7. The students understand that the work they did has value and gives them confidence that they can do the same for another employer.

6 FINAL THOUGHTS AND FUTURE WORK

We've seen the success of just three students at NERSC, however, there were approximately 20 students in the pilot program. The initial cohort who started in September 2023 have 42 students who are looking forward to a summer internship in 2024.

It has been NERSC's experience that learning how to analyze and visualize data is a skill that can be easily learned given that the student has the mathematical or programming foundation. The students are mentored to help understand what the data represents, what is important to show, and eventually create the dashboards that can tell a story to the SRE that helps them mitigate any kind of incidences Funding and finding the right faculty to teach the classes will be continuing challenges but it could be overcome.

Future work involves expanding the program so that we can intake more students, recruiting more employers for internships and exploring multi-semester internships for students who are more senior in the program. NERSC plans to continue to collaborate with Laney College, Oakland, in creating the next step, which is the second year of the Data Science program. For the SRE's at NERSC, OMNI is the center of a monitoring infrastructure that allows them to "see" the health and status of the facility [8].

Visualizing the data allows them to determine issues early, diagnose the problem quickly and come to a resolution as soon as possible in order to continue to serve their global users who use their HPC system and facility. This is the next step in creating a staffing pipeline for new talent and to fulfill the diversity and inclusion mission of the Lab.

ACKNOWLEDGMENTS

This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operated under Contract No. DEAC02-05CH11231.

REFERENCES

[1] Elizabeth Bautista, Melissa Romanus, Thomas Davis, Cary Whitney, and Theodore Kubaska. 2019. Collecting, monitoring, and analyzing facility and systems data at the national energy research scientific computing center. In *ICPP Workshops '19: Workshop Proceedings of the 48th International Conference on Parallel Processing*. August 2019, Kyoto, Japan, 1-9.

[2] Elizabeth Bautista and Nitin Sukhija. 2021. Employing directed internship and apprenticeship for fostering HPC training and education. *Journal of Computational Science 12*, 2, 33-36.

[3] Elizabeth Bautista and Nitin Sukhija. 2023. Creating pathways in disadvantaged communities towards STEM and HPC. *Journal of Computational Science 14*, 2, 2-5.

[4] Elizabeth Bautista, Nitin Sukhija, and Siqi Deng. 2022. Shasta log aggregation, monitoring and alerting in HPC environments with Grafana Loki and ServiceNow. In *Proceedings of the 2022 IEEE International Conference on Cluster Computing (CLUSTER).* September 2022, Heidelberg, Germany. 602-610.

[5] Berkeley Lab. Workforce Demographics. Retrieved from https://diversity.lbl.gov/berkeley-lab-workforce-demographics-fy2022/.

[6] Grafana Labs. Retrieved from https://grafana.com/.

[7] Raphael A. Irizarry. 2020. The role of academia in data science education. *Harvard Data Science Review 2*, 1.

[8] Emad Mushtaha, Saleh Abu Dabous, Imad Alsyouf, Amr Ahmed, and Naglaa Raafat Abdraboh. 2022. The challenges and opportunities of online learning and teaching at engineering and theoretical colleges during the pandemic. *Ain Shams Engineering Journal 13*, 6.

[9] Public Policy Institute of California. California's Need for Skilled Workers. Retrieved from https://www.ppic.org/publication/californias-need-for-skilled-workers/.

[10] Cary Whitney, Thomas Davis, and Elizabeth Bautista. 2016. NERSC Center-wide Data Collect. Retrieved from https://cug.org/proceedings/cug2016_proceedings/includes/files/pap101s2-file1.pdf.

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

Bryan Johnston
Advanced Computer Engineering Lab
CSIR
bjohnston@csir.co.za

Lara Timm
Advanced Computer Engineering Lab
CSIR
ltimm@csir.co.za

Eugene de Beste
Advanced Computer Engineering Lab
CSIR
edebeste@csir.co.za

Mabatho Hashatsi
Advanced Computer Engineering Lab
CSIR
mhashatsi@csir.co.za

ABSTRACT

In this paper, we present an approach to hands-on High Performance Computing (HPC) System Administrator training that is not reliant on high performance computing infrastructure. We introduce a scalable, standalone virtual 3-node OpenHPC-based training lab designed for Resource Constrained Environments (RCE's) that runs on a participant's local computer. We describe the technical components and implementation of the virtual HPC training lab and address the principles and best practices considered throughout the design of the training material.

KEYWORDS

SC Proceedings, High Performance Computing, CSIR, NICIS, CHPC, HPC Ecosystems Project, OpenHPC, Workforce development, Virtual Training, System Administrator Training.

1 INTRODUCTION

The training of new and existing HPC practitioners is recognized as a priority in the global HPC community [23]. The *HPC Ecosystems Project* is an initiative of the South African Department of Science and Innovation to address this priority. This is achieved through the distribution of repurposed HPC resources in Africa and subsequent training of HPC System Administrators to manage these systems [18].

Within the global HPC community, the common approach to delivering HPC System Administrator training is through physical face-to-face engagements. The hands-on component of HPC training is facilitated through leveraging existing HPC infrastructure or cloud-based services that simulate an HPC environment [2, 7–10, 40].

During its early stages (2014-16), the HPC Ecosystems Project followed the same approach: face-to-face training engagements using HPC resources allocated from a production HPC system or cloud-based platform[1]. Over time, the HPC Ecosystems Project has migrated much of its training material to digital delivery platforms[2] and has favored the use of virtual resources, in the form of local Virtual Machines (VMs) for the hands-on training component. This adaptation was made in response to the challenges experienced with the traditional approach towards HPC System Administrator training; namely relating to the use of physical HPC infrastructure or remotely provided HPC resources. We prioritize the delivery of HPC System Administrator training that does not require a constant internet connection and aims to provide downloadable resources to facilitate offline learning.

This paper will describe our approach towards offering HPC System Administrator training that overcomes these challenges and presents a more sustainable and scalable solution.

We emphasize that this is a training lab intended to teach HPC System Administrators how to configure and build an OpenHPC cluster from scratch. There are numerous localized virtual OpenHPC cluster solutions available publicly, but these are pre-configured virtual clusters that are not suitable for learning how to deploy an OpenHPC cluster. Going forward, we will refer to any allocated HPC resource, whether it is a remotely accessible HPC or a cloud-hosted service, collectively as a 'hosted HPC resource'; since these resources offer the same hands-on HPC experience for System Administrator training.

2 MERITS OF VIRTUAL MACHINES AS A TEACHING RESOURCE

A Virtual Machine (VM) is a software-simulated computer environment that behaves like a physical computer system that accesses the host computer's resources from software called a hypervisor. Virtualization allows the creation and running of multiple VMs on a single physical computer system. The hosting computer system for VMs can be a single laptop computer or a cloud-based system. Since the revival in popularity of VMs in the mid-2000s owing to the ability to operate VMs on commodity hardware [32, 33], significant research has been done on evaluating the impact of VMs as teaching resources.

Staubitz et al. [38] evaluated the use of VMs for hands-on exercises in courses and concluded that properly managed VMs can provide many benefits for scalable and online teaching, including:

- Reducing support costs ("The amount of effort and time the course provider has to invest to deliver the exercise or to help

https://doi.org/10.22369/issn.2153-4136/15/1/13

[1] The ACE Lab's internal private OpenNebula (pre-2020) and OpenStack cloud (2020+)

[2] HPC Ecosystems GitLab Training Repository; HPC Ecosystems YouTube Training Videos; OpenHPC 101 GitLab Pages

users with the setup") by utilizing mainstream desktop hypervisors and automated VM deployment tools;
- Reducing setup costs by preparing a 'golden image' in advance with all tools and software tailored for the course coordinator's needs which can then easily be replicated on demand for participants. The course coordinator can control the behavior of the tools that are curated in the central VM image.
- Locally hosted VM's can reduce costs for the course provider since computation is performed on the participants' machines.

Staubitz et al. [38] found that VMs are an effective way to simulate a real environment at greater scale for training labs. Students are trained for real world applications because they experience and use a system as it would typically be used by a professional. The adoption of VMs as a teaching tool for hands-on technical training has long been accepted as an acceptable and effective tool.

3 HPC SYSTEM ADMINISTRATOR TRAINING

3.1 Classifying the HPC System Administrator Audience

For our case study, we considered HPC System Administrator Training to be classified into two target categories: for Resource Constrained Environment (*RCE*), or for traditional scientific computing environments (*non-RCE*). We adopted the metrics below to classify an RCE or non-RCE. Sites classified as an RCE are shown to experience significant challenges when following a traditional training approach. Within this classification, an RCE lacks at least two of the following:
- Access to advanced computing infrastructure that can be purposed for scientific computing workloads.
- Computing resources that are available on-premises or via remote access (such as institutional partnerships or remote hosting).
- Access to stable and performant internet and infrastructure.
- Technology service levels that are considered reliable and generally uninterrupted (such as network connectivity and power supply).

In alignment with the goals of the HPC Ecosystems Project, the designed training approach aims to overcome the challenges faced when providing HPC System Administrator training in RCEs, where a traditional approach is not always appropriate or effective.

3.2 Hosted HPC Resources for Training

Through our own experience in both delivering and participating in many international HPC training programs that facilitate training through hosted HPC resources, we have identified several underlying limitations:
- Training class size is still limited by the limited availability of compute resources;
- The training lab's existence is dependent on the timed availability of advanced computing infrastructure (of any measure);
- The teaching resources and practical compute environments are ephemeral;
- The transitory life of the training resources limits time flexibility for participants before the resources are inevitably destroyed and the learning opportunity is over.

In particular, the class capacity restrictions associated with limited compute resources reduce the effective scalability of the HPC System Administrator training. Likewise, the reliance on timed availability of HPC resources along with the ephemeral properties of these training resources limits accessibility and long-term impact. Where hosted HPC resources are integrated into the training, any failure with the cloud host, while rare, will impact the entire class of participants. Additionally, it is undesirable that a participant's hard-fought gains during the training are at risk of being destroyed on the remotely hosted environment before they have concluded working with the resources. There is generally no straightforward mechanism for participants to reference their original training environment, or even to progress further on their original training once the training period is officially concluded. Simply put, to address the need for scalable and sustainable HPC training, short-lived cloud labs are short-lived answers.

3.3 Training HPC System Administrators in Resource Constrained Environments

As the HPC Ecosystems Project expanded into more than thirty African partner sites, so the need for scaling out of the training into these partner countries expanded. Specifically, many of the international partner sites were receiving their first HPC deployments and required HPC System Administrator training [35]. The traditional model of provisioning HPC or cloud-hosted training resources was not a viable approach to facilitating HPC technical training, given that many partner institutions met the classification of an RCE.
- Sites did not have HPC or cloud resources to use for provisioning lab resources.
- Many sites lacked reliable internet connectivity to facilitate remote access to hosted HPC resources in South Africa.

Our virtual HPC Training Lab is designed to provide an effective scalable and sustainable HPC training platform with these constraints in mind.

3.4 Training HPC System Administrators in non-Resource Constrained Environments

It is reasonable to assume that non-RCE's enjoy a richer and more effective training experience because they have access to advanced computing infrastructure, be it on-premises or remotely; however, we have indicated several shortcomings to the learning experience that are explicitly related to hosted HPC resources, such as the ephemeral nature of the resources and the limits to the class size that are directly linked to the resource capacity of the remote resource. We note that non-RCE's are not shielded from these training shortfalls.

4 THE CASE FOR A TAKEAWAY LAB

There is a need for truly scalable and sustainable HPC training to meet the training priorities of the HPC community [2, 23]. Based on our experiences in aspiring to deliver sustainable and scalable HPC training through hosted HPC resources, we have learned that the answer is not going to lie exclusively "in the cloud." While our solution was originally developed for a specific HPC community within a specific geographic region, we believe there is global utility for the principles of an on-host virtual computing cluster to facilitate HPC training. Our involvement with international HPC workshops suggests that many trainees globally would benefit from an HPC lab resource that lives on indefinitely after a course is concluded. Specifically, it would be preferable for participants to have the freedom to choose whether they wish to keep their HPC resource to continue their learning experience after a training course is concluded.

To overcome the fundamental barriers to effective HPC training labs associated with the traditional hosted model, we have considered lessons learned from E-Learning and Massive Open Online Course (MOOC) methodologies to develop an effective HPC training lab alternative. The resultant training lab is a virtual HPC lab that is reproducible, self-paced and emulates a basic three-node computing cluster on a trainee's local machine with an indefinite lifespan and without the need for any high-end computing resources or cloud infrastructure.

5 A MODEL FOR HPC TRAINING

The HPC Ecosystems virtual HPC lab does not require high-end computing resources or cloud infrastructure and is available on demand in the participant's preferred schedule because it is all hosted on their personal computer.

The development of the virtual lab considered best practices in the implementation of educational technology and followed previous lessons learned to provide a robust and effective learning platform for HPC System Administrator training [1, 2, 8, 11, 19, 36, 38].

Our intention is not to present our virtual HPC lab as the definitive ensemble of technical tools for all future HPC System Administrator training. Rather, we wish to showcase the principles underpinning our approach to promote a new model for scalable HPC System Administrator training.

5.1 Online Delivery vs. Online Training

Digital delivery has enabled our training solution to become available, scalable, and globally accessible. Although the training material is hosted online, this is simply to facilitate delivery of the training on demand and for free. The hands-on virtual lab is, in principle, capable of being modified for offline distribution and operation[1] and is designed to mitigate needing special HPC systems or infrastructure for training [13, 26].

Research indicates that online content delivery can be as good as, if not better than, on-premises course participation [8, 21]. "Online students learned as much as students in the Traditional version" [20] and [21] observed "the online students actually achieved superior learning outcomes despite spending less time."

The virtual training lab is intended to be easily scaled up to a virtual classroom environment to allow for blended learning delivery where needed. Without proposing dramatic changes to the existing in-person content, simply including an online rendition of the existing presentations and workshops could potentially be as effective as the current on-premises approach but open accessibility to a wider audience [20–22].

Given that the existing traditional face-to-face delivery mechanisms are time constrained with defined start and stop time windows, the potential for improved learning in shorter time with online content would suggest online delivery leads to improved impact of teaching HPC System Administrator skills [21].

6 TECHNICAL COMPONENTS

Commensurate with accessibility and sustainability principles, our virtual HPC lab uses commonly available opensource resources and differs from traditional HPC workshops in several ways:

1. Content is available in two formats - video tutorials and an interactive guide, available through YouTube and GitLab respectively;
2. The lab uses the OpenHPC[2] software stack to create the HPC environment.
3. *There is no cloud* – the HPC components are provisioned by Hashicorp Vagrant[3] and deployed using Oracle VirtualBox to emulate a 3-node virtual cluster that can run on any computer with at least 8GB of RAM.[4] Vagrant ensures consistency and reproducibility of the lab VMs;
4. The training lab has the potential to be modified to operate entirely offline (all components are downloadable);
5. The training is self-paced and is available on demand, with no cut-off deadline for access to the HPC resources;
6. The training lab has fewer restrictions to infrastructure modifications
7. There is a freedom to fail – the localised lab can use snapshots to preserve development states;
8. The virtual cluster remains on the participant's computer until they choose to remove it;
9. The finished virtual management node can be tweaked to manage physical HPC infrastructure.

The virtual HPC lab is available to anyone wishing to configure a basic 3-node HPC system or to practice / learn the OpenHPC software stack. Since all resources are run on the trainees' local machines, there is no need to troubleshoot remote connectivity or to create cloud credentials. Significantly, since every HPC resource is localized to each participant, there is no limit to the number of participants able to partake in the training.

Perhaps the strongest arguments in favor of a remotely hosted HPC resource are the comparatively poor performance of a locally hosted VM and the requirement for participants to have access to local computing systems with sufficient memory and disk space to run the virtual HPC labs. We do not consider performance to be a reason for concern since we are focused on delivering an accessible learning experience rather than a performant HPC cluster. Participant feedback indicates 4GB RAM is sufficient to operate the HPC Lab – the VirtualBox VM's are collectively allocated 7GB of RAM, and the finished lab occupies 6GB of local disk space; we have not yet encountered a participant who has been unable to complete the virtual lab – if necessary, the virtual HPC cluster can be reduced to a 2-node cluster, which will allocate 4GB of RAM through VirtualBox.

6.1 Support Materials

A scalable training lab ensures that content is available to any and all numbers of participants. We took note that MOOCs provide a means for scalable education but require some attention to potential challenges in delivering successfully at scale [29]. We also

[1] We have prototyped an offline solution but it has not yet formally been released.

[2] OpenHPC

[3] Vagrant by HashiCorp (vagrantup.com)

[4] Each compute node consumes 3GB of RAM and the **smshost** consumes 1GB of RAM.

considered [31] insights into the use of a cloud-based virtual lab to "develop scalable, maintainable, and shareable contents that minimize technical hurdles while still exposing students to critical concepts in cluster computing". When delivering our training, we have selected delivery services that support scalable delivery and downloadable content for offline consumption.

6.1.1 Virtual Lab Guide

The primary lab guide instructs users on how to deploy 3-node virtual cluster using Vagrant, VirtualBox, and OpenHPC. In line with established education principles, the efficacy of training content is enhanced with additional documentation beyond the core deployment steps for the virtual HPC [3]. To this end, the guide includes general high-level principles and context-relevant supplementary information and elaborations of the HPC software stack components. We use GitLab to host the training guide and training material and deliver the guide using GitLab Pages.[1]

6.1.2 Virtual Lab Video Guide

The video content and the documentation are constructed with similar detail such that the media can be used independently or collectively, depending on the specific needs of the participant. While the primary objective is to deliver a virtual HPC lab, it is evident that the most effective delivery of the content will have accompanying reference documentation [3]. When considering an appropriate delivery method for the video content, we noted [30] observations that many of the conferencing technologies they tested had significant constraints on numbers of connections. Noting that YouTube offers free access, free hosting, and offline download capabilities, YouTube is a potential platform to deliver training [5]. In line with these findings we chose to host the video content on YouTube to offset any delivery constraints.

YouTube allows for videos to be downloaded for offline viewing and is designed to be compatible with most modern devices (phones, tablets, mobile devices, etc.). The option to provide for offline video and document downloads is vital for accessibility for African partners, especially where electricity and network reliability is uncertain. Additionally, YouTube caters for automated captions in multiple languages, broadening accessibility for non-English speaking participants [19, 27]. While we have employed YouTube videos to enhance the learning experience for the virtual lab, discussion around the specifics of the format and construction of the video content is beyond the scope of this case study.

6.2 OpenHPC

OpenHPC is the standard HPC software stack for the HPC Ecosystems Community and the selection of OpenHPC is not arbitrary – it is a popular HPC software stack due to its simplification of implementing traditional HPC software components[34]. It is widely used with an active and growing community of users and contributors [15, 24, 34, 35, 37].

6.3 No Cloud

The motivation for a solution that is not dependent on a hosted HPC environment has already been explored in previous sections; suffice to say, we sought a solution that would provide at least an equivalent technical experience to one delivered by a remote resource. We settled on a combination of VirtualBox managed by *Hashicorp Vagrant*[2] that emulates many of the benefits of a traditional hosted delivery platform while avoiding the problems related to remote / hosted resources indicated earlier. Since all resources are provisioned through automated deployment and all run on the trainees' local machines, there is no need to troubleshoot remote connectivity or to create cloud credentials, or to spend any time with configuring or provisioning remote compute resources. Since everything is local to each participant, there is no limit to the number of participants that can undertake the training in the same cohort. From a training perspective, any catastrophic-level failures in the localized virtual HPC resource will be limited to an individual participant and not an entire class.

The lab has been verified to work on host systems with at least 8GB of RAM (each compute node requires 3GB of RAM and the management node requires 1GB of RAM). Survey results from participants indicate that some have reported success with deploying the 3-node virtual cluster with 4GB host systems. The deployed virtual cluster occupies 6GB of local storage.

6.3.1 VirtualBox and Vagrant

When selecting a type-2 Hypervisor that can be installed on participants' local machines, VirtualBox was selected because it is a widely used Hypervisor with support for the primary computer Operating Systems (Linux, Windows, Mac). We were cognisant of three problems observed by Helsing et al. [12] when implementing a localized virtual lab with VirtualBox –

1. the lack of user skills;
2. the long latency in installing and configuring the environment;
3. and the lack of resources and material to help solve technology issues in the virtual lab.

Noting that the participants will have heterogeneous computer environments, we introduced Vagrant as a virtual machine environment management tool that ensures configuration parity to address the three problems raised by Helsing and Staubitz [12, 38] in addition to managing the unpredictability of heterogeneous environments. Vagrant offers a solution to the "configuration drift". Since it is free and supports all major platforms, Vagrant ensures parity for all virtual lab participants and significantly reduces the complexity of installing and configuring the virtual machine environment for the virtual lab [4, 38]. Staubitz et al. showed that Vagrant "reduces the friction for creation, distribution, setup and update of Virtual Machines" and "can reduce hosting costs, improve the user experience for learners compared to traditional virtualization software and thus reduce support efforts on the course provider's side."

6.4 Potential for Offline Delivery

The HPC lab's virtual machine environment is deployed and stored locally on the participant's computer and the virtual cluster nodes are fully accessible without any internet connection. In principle, with the necessary Linux repositories synchronized offline and the GitLab repository cloned to the local machine, the entire training lab can plausibly run without an internet connection.

6.5 Self-paced and On Demand

Training labs—either in-person or remote—have additional accessibility and availability considerations with respect to the cost and time factors associated with the training. Even where training

[1] GitLab Pages | GitLab

[2] Vagrant | HashiCorp Developer

may be offered for free, there might be a cost factor associated with traveling to the event, or the time cost of participating during work hours. HPC tutorials that allow remote participation may resolve the issue of travel costs, but these tutorials will still be governed by a time schedule (and in some cases, running across a different time zone). The need for on demand material is borne from the fact that people seeking training may not find any in a time zone suitable to them [29]. Accordingly, our lab material is available on demand and is time zone agnostic, with participants able to access the material at their convenience, and at no cost.

6.6 Fewer Restrictions

Since the virtual machine training environment is stored locally, the participants have the control to modify the VM infrastructure if desired, such as network interface parameters, RAM, CPU core count, etc. On machines with sufficient resources, additional virtual nodes can be added to the virtual cluster – none of this would be possible on a cloud-hosted HPC lab without an internet connection or without appropriate administrator permissions.

6.7 Freedom to Fail

A benefit of many virtual environments is the ability to perform snapshots—an automated point-in-time backup of the virtual infrastructure that typically captures the state of the virtual resources which can be restored at a later stage. Besides the fact that most remotely served HPC labs disable the snapshot feature, it would in any event be of limited benefit since the remote resources are ephemeral. Since the HPC Ecosystems virtual HPC lab is hosted on local machines, participants can secure their workload with snapshots whenever they wish, which encourages the principle of "freedom to fail" [25, 39].

6.8 Persistent Image

Beyond the benefits of the snapshot feature, the virtual HPC lab remains persistent on the local machine indefinitely. Participants can perform the training at their own pace, or to explore further upon completion of the official training. Notwithstanding the utility in having a persistent image for further exploration, we acknowledge that it is advisable to keep images up to date and we advise participants to regularly update their software stack.

6.9 Virtual to Physical

The transition from the virtual HPC cluster to a live physical system is trivial—participants can either start over and repeat the process on physical infrastructure or they can take the documented steps to link the completed virtual management node to physical resources, which is a relatively straightforward procedure. The ability to transition from the virtual lab to a production system with only slight modifications to the virtual cluster brings substantial value to the persistent image.

7 EXISTING ALTERNATIVES

In addition to tutorials offered at HPC conferences (such as ISC, SC, and PEARC), there are many documented training and learning opportunities for HPC System Administrators facilitated in the Northern Hemisphere. For example:

- The TACC Institute Series
- Linux Cluster Institute workshops
- PRACE Training Portal

By contrast, it is difficult to locate documented public HPC System Administrator training opportunities occurring on the African continent. Under these conditions, accessibility to face-to-face HPC System Administrator training is affected by geographical location. When training events may include an online component, the issue of time zone differences may also be a factor.

7.1 OpenHPC Training

The official OpenHPC guide states that it is intended for *experienced Linux System Administrators* [34], with current training for OpenHPC conducted as short hands-on workshops during conferences hosted primarily in the Northern Hemisphere. Based on the information presented in Table 1, there have been approximately **fifteen hours of hands-on tutorials** conducted by the OpenHPC leadership community to date, with each session usually lasting approximately three hours of dedicated hands-on training.

Some other limited training on OpenHPC is provided at research institutions through internships or workshops. For instance, Wofford et al. [40] describes a 10-week internship program hosted at Los Alamos National Laboratory that includes aspects of hands-on OpenHPC experience. Outside of the engagements listed and the limited training at research institutions, OpenHPC support is informally provided through the OpenHPC virtual group.

Table 1: List of Known Formal Training for OpenHPC

Event	Duration in hours	Year & Month	Description
SC20	3	2020-11	Tutorial
SC19	1	2019-11	Birds of a Feather
PEARC19	*3.5*	2019-07	Tutorial
ISC 2019	*1*	2019-06	Birds of a Feather
HPCKP'19	0.5	2019-06	Presentation
SC18	1 unknown	2018-11	Birds of a Feather Tutorial
DAAC 2018	unknown	2018-11	Presentation
Open Source Summit 2018	3	2018-08	Tutorial
MVAPICH'18	unknown	2018-08	Presentation
DevConf.CZ 2018	0.25	2018-01	Presentation
SC17	1	2017-11	Birds of a Feather
MVAPICH'17	unknown	2017-08	Presentation
PEARC 17	3.5	2017-07	Tutorial
ISC 2017	1	2017-06	Birds of a Feather
HPCKP 2017	0.5	2017-06	Presentation
SC16	1	2016-11	Birds of a Feather
MVAPICH 2016	unknown	2016-08	Presentation
ISC 2016	1	2016-06	Birds of a Feather
FOSDEM 2016	0.45	2016-01	Presentation

8 RESULTS AND IMPACT

Since the launch of the OpenHPC1.3.x Virtual Lab in October 2020, there have been 226 participants trained in six formal online workshops hosted to date, with more than 5,500 views of the accompanying online training videos. Notably, the Virtual HPC

Lab has attracted participants from outside of Africa, attesting to the global relevance as on demand virtual OpenHPC Training Lab.

8.1 Evaluation & Assessment

Regular participant assessment is identified as a key component to enhancing online learning efficacy [29]. Similar HPC Training workshops collect "attendee evaluations for each workshop session to ensure topical relevance, assess the instructors, and support a continual improvement process for the instructional materials." [2] or daily 'sticky note' surveys and end-of-term longer, formal surveys [40] To measure the effectiveness of the Training Lab, both quantitative and qualitative evaluation must be measured.

8.1.1 Quantitative Feedback

Quantitative evaluation of the HPC Ecosystems Virtual HPC Lab can be measured by:

- The number of Virtual Clusters successfully activated on the software layer monitoring agent that is associated with the Virtual Lab;
- The number of engagements with GitLab, YouTube, and the OpenHPC Virtual Lab guide.

Although the listed measures provide definitive *values*, the metrics themselves may not be meaningful without additional information, such as the number of *unique* users attempting the training material, or the number of attempts (and revisits) to the training material by a single participant over time. To track each participant of the training, a registration form is incorporated for the formal online training events, but this does not include measures for informal ad hoc consumption of the material. We intend to incorporate additional quantitative metrics to measure time-to-completion as well as the success rate of the virtual cluster deployments.

8.1.2 Qualitative Feedback

Participants are encouraged to provide regular feedback to enhance the quality of the content being delivered in future iterations (and possibly to allow for quick-response adjustments to issues with the course).

8.2 Risks

When the HPC System Administrator training is not conducted face-to-face, it will be prone to the common risk factors associated with exclusively online courses, such as high drop-out rates, isolation in an online course, lack of proximity to physical hardware, and lack of dedicated technical support for troubleshooting [6, 16, 17, 20, 25, 26, 29]. Findings by Michinov et al. [28] indicate that there is a correlation between lack of participation in discussion forums and low performance in online learning environments; to see an improvement, participants should be encouraged to participate in online discussions.

With both online and virtual learning, participants do not have the opportunity to see, touch or interact with actual hardware components [26]. Cahill et al. [6] advise providing additional instructional resources along with the online content to reduce limitations of online-based training.

Further risks relate to our specific approach to the digital delivery of the training material; the training model requires for the resources to be installed on a local machine:

- Participants are expected to have computers with adequate specifications;
- An internet connection is necessary to connect to the digital delivery;
- Where internet connection is not available, the coordination of shipping the modified offline virtual HPC lab potentially to international sites.

9 CONCLUSION

The move to virtual content delivery for HPC System Administrator training has enabled the HPC Ecosystems Project to reach a wider and larger audience in a fraction of the usual time. In our formal training events, we have observed that the localised virtual HPC lab enables many participants that do not complete the training during the allotted period to successfully deploy their virtual HPC systems later in their own time.

Cytowski et al. [8] observe that there is no universal learning solution – "various solutions and platforms need to be carefully selected for different groups of interest" – accordingly, much of what is asserted in this case study was initiated with the narrow project scope of HPC Ecosystems community members in mind. We acknowledge that hosted HPC resources are also helpful in facilitating HPC training and there is certainly merit in adopting a hosted platform for certain types of training or audiences.

In all considerations, in order to deliver a solution that meets the requirements of the HPC Ecosystems audience (who are all considered RCE's), emphasis has been placed on scalability, sustainability, and self-sufficiency.

While cloud-computing resources can be used for virtual labs [2] these can prove costly and inaccessible for participants with internet connectivity constraints, such as those present in RCE's. Catering for offline interaction with a virtual lab not only expands accessibility but offers limitless scalability since each participant will be hosting their own virtual cluster infrastructure.

Through the development of a virtual training lab that does not rely on a remote cloud for HPC resources, we believe we have identified a sustainable and scalable solution for practical HPC training that reaches further than the HPC Ecosytems community: an accessible on-demand self-paced virtual HPC lab where the HPC resources remain available indefinitely on a user's local machine.

10 FUTURE WORK

An updated version of the Virtual HPC Lab (OpenHPC 2.x) is ready to launch in 2023Q3. Further work is underway to develop additional HPC modules that can be treated as additional standalone courses or used as 'bolt-on' modules to the foundational virtual 3-node cluster that is deployed in the Virtual HPC Lab.

Future planned OpenHPC modules include OpenOnDemand [14]. We currently have a group of supporters and contributors from numerous countries who are offering time and resources towards developing our future content and we always welcome more!

ACKNOWLEDGMENTS

John Poole, Mopeli Khama, Anton Limbo, Rammolenyane Lethaha, Sakhile Masoka, Israel Tshililo, Matthew Cawood, Themba Hlatshwayo, Zintle Sanda, Zama Mtshali.

REFERENCES

[1] Olatunde Abiona, Clement Onime, Stefano Cozzini, and Sebsibe Hailemariam. 2011. Capacity building for HPC Infrastructure setup in Africa: The ICTP experience. In *2011*

IST-Africa Conference Proceedings, May 2011, Gaborone, Botswana, 1-8.

[2] David Akin, Fang Cherry Liu, Amiya Kumar Maji, Henry Neeman, Resa Reynolds, Andrew Sherman, Michael Showerman, Jenett Tillotson, John Towns, George William Turner, et al. 2017. Linux clusters institute workshops: Building the HPC and research computing systems professionals workforce, In *EventHPC Systems Professionals Workshop, Proceedings of HPCSYSPROS 2017*, Denver Colorado, November 2017.

[3] Maha Bali. 2014. MOOC pedagogy: Gleaning good practice from existing MOOCs. *MERLOT Journal of Online Learning and Teaching 10*, 1.

[4] Abhishek Bhattacharya, Shagun Paul, and Arunima Jaiswal. 2014. Operating systems: Basic concepts and challenges against virtualization. *International Journal of Computer Applications 102*, 4, 975–8887.

[5] Sloane C. Burke, Shonna Snyder, and Robin C. Rager. 2009. An assessment of faculty usage of YouTube as a teaching resource. *Internet Journal of Allied Health Sciences and Practice 7*, 1, 8.

[6] Katharine J. Cahill, Scott Lathrop, and Steven Gordon. 2017. Building a community of practice to prepare the HPC workforce. *Procedia Computer Science 108*, 2131-2140.

[7] Carolyn Connor, Amanda Bonnie, Gary Grider, and Andree Jacobson. 2017. Next generation HPC workforce development: The computer system, cluster, and networking summer institute. *In Proceedings of EduHPC 2016: Workshop on Education for High-Performance Computing,* Salt Lake City Utah, November 2016, 32-39.

[8] Maciej Cytowski, Christopher Harris, Luke Edwards, Karina Nunez Mark Gray, and Aditi Subramanya. 2019. HPC education and training: An Australian perspective. *Journal of Computational Science Education 10*, 1, 48-52.

[9] Alex Younts and Stephen Lien Harrell. 2020. Teaching HPC systems administrators. *Journal of Computational Science Education 11*, 1, 100-105.

[10] Stephen Lien Harrell, Hai AhNam, Veronica Vergara Larrea, Kurt Keville, and Dan Kamalic. Student cluster competition: A multi-disciplinary undergraduate HPC educational tool. In *EduHPC '15: Proceedings of the Workshop on Education for High-Performance Computing*, Austin Texas, November 2015, 1-8.

[11] Stewart Hase and Chris Kenyon. 2001. Moving from andragogy to heutagogy: implications for VET. In *Proceedings of Research to Reality: Putting VET Research to Work: Australian Vocational Education and training Research Association* (AVETRA), Adelaide SA, March 2001.

[12] Joseph Helsing, Paulette Lewis, and Edward Warga. 2013. Teaching tools, applications, and infrastructure for digital curation through the use of a virtual lab. In *iConference 2013 Proceedings*, University of North Texas, February 2013. 781–784.

[13] Violeta Holmes and Ibad Kureshi. 2015. Developing high performance computing resources for teaching cluster and grid computing courses. *Procedia Computer Science 51*, 1, 1714–1723.

[14] 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. *The Journal of Open Source Software* 43, 1–43.

[15] Satrio Husodo, Jacob Chappell, Vikram Gazula, Lowell Pike, and James Griffioen. 2019. Slicing and dicing OpenHPC infrastructure: Virtual clusters in OpenStack. In In *PEARC '19: Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines, Chicago Illinois,* July 2019, 1-6.

[16] Alexandru Iosup and Dick Epema. 2014. An experience report on using gamification in technical higher education. In *SIGCSE '14: Proceedings of the 45th ACM technical symposium on Computer science education,* Atlanta Georgia, March 2014, 27–32.

[17] Evelyn S. Johnson, Michael J. Humphrey, and Keith W. Allred. 2017. Online learning and mentors: Addressing the shortage of rural special educators through technology and collaboration. *Rural Special Education Quarterly 28*, 2,17–21.

[18] Bryan Johnston. 2019. HPC Ecosystems Project: Facilitating advanced research computing in Africa. In *PEARC '19: Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines, Chicago Illinois,* July 2019, 1-3.

[19] Troy Jones and Kristen Cuthrell. 2011. YouTube: Educational potentials and pitfalls. *Computers in the Schools 28*, 1.

[20] David Joyner. 2018. Toward CS1 at scale: Building and testing a MOOC-for-credit candidate. In *L@S '18: Proceedings of the Fifth Annual ACM Conference on Learning at Scale,* London, June 2018, 1-10.

[21] David Joyner and Melinda Mcdaniel. 2019. Replicating and unraveling performance and behavioral differences between an online and a traditional CS course. In *CompEd '19: Proceedings of the ACM Conference on Global Computing Education*, Chengdu China, May 2019, 157–163.

[22] David A. Joyner, Charles Isbell, Thad Starner, and Ashok Goel. 2019. Five years of graduate CS education online and at scale. In *CompEd '19: Proceedings of the ACM Conference on Global Computing Education,* Chengdu China, May 2019, 16–22. DOI:https://doi.org/10.1145/3300115.3309534.

[23] Julian Kunkel, Kai Himstedt, Nathanael Hübbe, Hinnerk Stüben, Sandra Schröder, Michael Kuhn, Matthias Riebisch, Stephan Olbrich, Thomas Ludwig, Weronika Filinger, Jean-Thomas Acquaviva, Anja Gerbes, and Lev Lafayette. 2019. Towards an HPC certification program. *The Journal of Computational Science Education 10*, 1, 88–89.

[24] Chet Langin. 2019. From BigDog to BigDawg transitioning an HPC cluster for sustainability. In *PEARC '19: Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines*, Chicago Illinois, August 2019.

[25] Joey Lee and Jessica Hammer. 2011. Gamification in Education: What, How, Why Bother? *Academic Exchange Quarterly 15,* 2, 146.

[26] Pedro López and Elviro Baydal. 2017. On a course on computer cluster configuration and administration. *Journal of Parallel and Distributed Computing 105*, 127–137.

[27] Justin R. Louder, Suzanne Tapp, Larry Phillippe, and Jackie Luft. 2016. Video captioning software to help comply with

ADA accessibility requirements. *Community College Enterprise 22*, 1, 71–76.
[28] Nicolas Michinov, Spohie Brunot, Olivier Le Bohec, Jacques Juhel, and Marine Delaval. 2011. Procrastination, participation, and performance in online learning environments. *Computers & Educatyion 56,* 1, 243-252.
[29] Julia Mullen, Weronika Filinger, Lauren Milechin, and David Henty. 2019. The impact of MOOC methodology on the scalability, accessibility and development of HPC education and training. *The Journal of Computational Science Education*. 10, 1, 67-73. DOI:https://doi.org/10.22369/issn.2153-4136/10/1/11.
[30] Henry Neeman, Horst Severini, Dee Wu, and Katherine Kantardjieff. 2010. Teaching high performance computing via videoconferencing. *ACM Inroads*. 1, 167–71.
[31] Linh B. Ngo and Jeff Denton. 2019. Using CloudLab as a scalable platform for teaching luster computing. *Journal of Computational Science Education 10,* 1, 100-106.
[32] Mendel Rosenblum. 2004. The reincarnation of virtual machines: Virtualization makes a comeback. *Queue 2*, 5, 32-40.
[33] Mendel Rosenblum and Tal Garfinkel. 2005. Virtual machine monitors: Current technology and future trends. *Computer 38*, 5, 39–47. DOI:https://doi.org/10.1109/MC.2005.176.
[34] Karl W. Schulz, C. Reese Baird, David Brayford, Yiannis Georgiou, Gregory M. Kurtzer, Derek K. Simmel, Thomas L. Sterling, Nirmala Sundararajan, and Eric Van Hensbergen. 2016. Cluster computing with OpenHPC. Retrieved from https://api.semanticscholar.org/CorpusID:63310875.
[35] Jimmy Shapopi, Anton Limbo, and Michael Backes. 2023. Namibia's first high performance computer. *South African Computer Journal 35*, 1, 4–18. DOI:https://doi.org/10.18489/SACJ.V35I1.1189.
[36] Elizabeth Shoop, Malcolm Kane, Richard Brown, Devry Lin, Eric Biggers, and Maura Warner. 2012. Virtual clusters for parallel and distributed education. In *SIGCSE '12: Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, Raleigh North Carolina, March 2012, 517-522.
[37] Christopher Simmons, Karl Schulz,. and Derek Simmel. 2020. Customizing OpenHPC. In *PEARC '20 Companion: Proceedings of the Conference on Practice and Experience in Advanced Research Computing*, Portland Oregon, July 2020.
[38] Thomas Staubitz, Maximilian Brehm, Johannes Jasper, Thomas Werkmeister, Ralf Teusner, Christian Willems, Jan Renz, and Christoph Meinel. 2016. Vagrant virtual machines for hands-on exercises in massive open online courses. *Smart Innovation, Systems and Technologies*. 59, 363–373.
[39] Andrew Stott and Carman Neustaedter. 2013.Analysis of gamification in education. Retrieved from http://clab.iat.sfu.ca/pubs/Stott-Gamification.pdf.
[40] J. Lowell Wofford and Cory Lueninghoener. 2019. The Supercomputing Institute: A systems-focused approach to HPC training and education. *Journal of Computational Science and Education 11,* 1, 73-80.

March 2024
Volume 15 Issue 1
ISSN 2153-4136 (online)