Fifteen Years of International HPC Summer School

Scott Callaghan
Statewide California Earthquake Center
University of Southern California
Los Angeles, USA
scottcal@usc.edu

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

Ilya Zhukov
Forschungszentrum Jülich
Jülich, Germany
i.zhukov@fz-juelich.de

Hermann Lederer
Max Planck Computing and Data Facility
Garching, Germany
lederer@mpcdf.mpg.de

John Urbanic
Pittsburgh Supercomputing Center
Pittsburgh, PA, USA
urbanic@psc.edu

Abstract

Since 2010, the International HPC Summer School (IHPCSS) has trained more than 1,500 graduate students and early-career researchers in high-performance computing (HPC). Originally a European-US collaboration, it now includes Japan, Canada, Australia and South Africa, and serves around 80 students and 40 staff each year. This paper examines the evolution of the technical program, which initially focused on domain-specific scientific applications and MPI/OpenMP programming, but later expanded to include emerging technologies like GPU acceleration, Python for HPC, big data analytics, and AI/ML. It also discusses the challenges IHPCSS faced - technical, logistical, and demographic - and how they were addressed through real-time HPC access, mentorship, and adaptable sessions for mixed-skill audiences. IHPCSS continues to provide inclusive and high-quality trainings around the world by integrating new technologies and responding to participant feedback, while maintaining the core principles of HPC.

CCS Concepts

• **Computing methodologies → Massively parallel and high-performance simulations**; • **Social and professional topics → Computational science and engineering education**; **Informal education**.

Keywords

high performance computing, education and training, computational modeling, workforce development

ACM Reference Format:
Scott Callaghan, Weronika Filinger, Ilya Zhukov, Hermann Lederer, and John Urbanic. 2025. Fifteen Years of International HPC Summer School. In *Practice and Experience in Advanced Research Computing (PEARC '25), July 20–24, 2025, Columbus, OH, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3708035.3736011

PEARC '25, Columbus, OH, USA

ACM ISBN 979-8-4007-1398-9/25/07
https://doi.org/10.1145/3708035.3736011

1 Introduction

High performance computing (HPC) is a dynamic field, vital in enabling cutting-edge research in many science and engineering fields. However, training new users to use HPC tools and technologies remains difficult. Many HPC users are domain specialists who are not introduced to HPC until graduate school or later, and connecting users with training opportunities and the broader HPC community can be challenging.

With this in mind, the International HPC Summer School (IHPCSS, or Summer School) was founded in 2010 to provide HPC training opportunities to graduate students and early-career researchers. IHPCSS is a multi-institutional collaboration that has been bringing together about 80 students and 40 staff for a week of training and community-building for 15 years, encouraging all attendees to benefit from the power of collaboration. The Summer School is currently organized by institutions representing Canada (Digital Research Alliance of Canada), the US (NCSA and TACC), the UK (EPCC), Europe (EuroHPC JU via the HPC SPECTRA project), Japan (RIKEN R-CCS), Australia (Pawsey), and South Africa (NICIS CHPC and NITheCS).

In previous papers, we have described the IHPCSS mentoring program [1] and community-building frameworks [2]. Here, we will describe the evolution of the technical program in response to student feedback and to changing demands of the field, and our efforts to continue to seek the best formats and topics to make the IHPCSS a valuable experience for attendees. In Section 2 we will provide background on the IHPCSS. In Section 3 we explain our framework for improvement. In Section 4 we describe the evolution of the student population, and in Section 5 we describe the philosophy behind and content of the technical program. In Section 6 we discuss some of the challenges we have confronted, and we conclude in Section 7.

2 Background on the IHPCSS

The purpose of the IHPCSS is to prepare students to apply HPC techniques in various research disciplines and to foster international and interdisciplinary collaboration. This differs from other HPC training programs, which may be aimed at more advanced topics (e.g. Argonne Training Program on Extreme-Scale Computing[9]), have a more specific emphasis (e.g. Jülich Quantum Computing

Summer School[3]), or be focused on a single region (e.g. CHPC Coding Summer School[4]). The idea was born at the joint DEISA [14] and TeraGrid [15] workshop held in July 2009 at the Pittsburgh Supercomputing Center that focused on improving international scientific collaborations. Since the first event in 2010, IHPCSS has grown into a collaborative effort bringing together students, mentors, and educators from Europe and USA (since 2010), Japan (2013), Canada (2014), Australia (2023) and South Africa (2024). Despite several changes to funding sources over the years, the IHPCSS keeps growing and adapting, showcasing the power of international collaboration. Over the last 15 years, about 1,500 participants (including students, mentors and educators) contributed to making the School a successful and nurturing experience for all involved. Past students have returned as returning mentors and sometimes as staff members to give back to the community.

The content of the early Schools was broadly divided into two categories: 1) challenges by scientific discipline and 2) HPC programming techniques and tools. The focus changed over time in response to advances in technology, the growth of HPC as an interdisciplinary field, and diversification of the community. As students and their needs changed, the content and presenters of IHPCSS also changed, while keeping the theme of interdisciplinary collaboration at its core. The program of the most recent School (IHPCSS'24) consisted of the following type of sessions: technical lectures, practical hands-on sessions, panels, programming challenge, poster reception, structured and unstructured mentoring, and social events. This paper focuses on the technical sessions and how they have evolved in response to changes in demographics and advancement of HPC and its applications.

3 Framework for Evolution

The Summer School is able to adapt and evolve to better address community needs thanks to the interaction between the evaluation process, which enables the planning committee to receive thorough and honest student feedback, and the planning process, which every year examines the feedback from the previous year and implements changes to improve the student experience. Every year, student feedback is diligently collected and then distilled into a set of recommendations. The planning committee carefully considers every recommendation and implements changes that are expected to improve the student experience. We describe these two processes below.

3.1 Evaluation Process

To evaluate its effectiveness and relevance, since the very beginning, IHPCSS has implemented a rigorous evaluation process to ensure continuous improvement of the format and content of the School. The evaluations are conducted by an independent team (i.e. not involved in organizing or running the event itself) to avoid bias. The evaluation team has typically been affiliated with the US partner organization (TeraGrid, XSEDE, NCSA). They conduct pre-, mid- and post-event surveys and hold focus groups during the event, and following the event prepare a quantitative and qualitative summary of the results and specific recommendations for the next year. For example, after the 2024 IHPCSS, the evaluation report recommended modifying the schedule to reduce fatigue. In response to student feedback, many changes were introduced to better address the shift in technical skills, background knowledge, and research interests of the participants, which will be described in more detail in Section 5.

3.2 Planning Process

The planning committee, consisting of representatives from all partner organizations, starts its work the preceding autumn. The committee meets biweekly initially, changing to weekly a few months before the event. Decisions are made by consensus on the basis of the input of the subcommittees and the partner organizations. Planning process includes many phases, corresponding to local venue selection and logistics, advertisement, application and selection process, designing program and inviting speakers, student registration and follow-up communications, travel arrangements, content upload to virtual learning platform and refining guidelines for mentors and speakers. Based on the evaluation report from the previous year's event, changes to the technical program are discussed and implemented. For example, in 2025 we adjusted the schedule to reduce the length of the first two days based on the evaluation recommendation.

4 Attendee Selection and Demographics

The profile of the average Summer School student has changed over the last 15 years, a result of changes both in the simulation and modeling community and in how applicants are selected. These changes are described below.

4.1 Application Process

Each year, the partner organizations individually determine how many students they are able to fund. The total number of students is kept around 80 (increased to 90 in 2025) to ensure high-quality interactions and personal attention. To identify the specific candidates who can best benefit from the School, applicants complete an application in December or January that includes questions about their areas of research, HPC experience, motivation to attend, and interest in learning about various topics covered by the Summer School.

The selection process should be fair and objective. To this end, we have implemented a number of changes over time. Originally, applicants were asked to rate their proficiency (beginner, intermediate, advanced, expert) on a variety of HPC topics (MPI, OpenMP, command-line terminal, etc.). The result was often a considerable number of over- and under-qualified participants, indicating that self-reported abilities may be unreliable. Hence, the questions were changed to ask about the frequency of usage instead (daily, weekly, monthly, occasionally, never). This is a more objective measure and we have found it better reflects the capabilities of the applicants. We also ask how the applicant's lab, department, and community could benefit from their attendance. Since we cannot accept everyone who might benefit, we seek to expand the reach of the Summer School by encouraging attendees to share their knowledge at their home institutions.

Most organizational partners use a common application process, with occasional exceptions due to funding restrictions. Each organizational partner is responsible for the review and selection process for attendees from their geographic region. To ensure consistency in technical abilities between applicants, a common rubric was established. Most partners use multiple reviewers on each application to provide numerical ratings. Applications are pseudonymized before being reviewed, since this has been found to help achieve a fair distribution of selected participants. In addition to the rubric, each organization has the flexibility to adopt additional criteria in the selection process. Different organizations may have varying factors to consider, such as gender, institution, and geographical diversity. For example, in 2024 Pawsey set a "soft" limit of maximum two participants per institution, which allowed them to support students from five different states in Australia.

The application process is typically very competitive. In 2024, 461 applications were received for 84 seats, for an acceptance rate of 18%. The review process is completed by March, and applicants are promptly notified so they can make travel arrangements and arrange visa applications if necessary. We also maintain a wait-list if students initially invited are unable to attend.

4.2 Applicant and Attendee Demographics

The data presented in this section was gathered from the evaluation reports described in Section 3.1[5][6][10][11][12]. Note that no Summer School was held in 2020 due to COVID-19, and IHPCSS'21 was fully virtual.

Over the years, the demographics of the applicants have evolved. One metric is the country of applicant's institution, as seen in Figure 1. There have been some fluctuations in the number of applicants from the respective regions over the years, which may be linked to the location or dissemination of the event. However, the total number of applications is consistently above 300, showing a high level of interest in the Summer School and competition for acceptance. Additionally, while Figure 1 shows only applications from eligible institutions (i.e. those based in one of the partner countries or region), each year the Summer School receives applications from people from ineligible institutions, indicating interest in the School globally.

The years 2013–2017 had a proportion of 75%–80% male attendees, while subsequent years were in the 65%–70% range. Figure 2 shows applicants' gender identity for 2018 through 2024. The stacked bar chart compares applications on the left (ranging from 322 to 501) versus attendees on the right (ranging from 55 to 74), with totals displayed above each column. The gender distribution is shown as percentages, with blue representing male (approximately 60-70%), pink representing female (roughly 20-40%), and yellow representing other (2% to 3%). 2024 had the highest percentage of non-male-identified attendees, at 43%. Conscious efforts to improve gender diversity in our student population through wider-reaching and more inclusive advertising, an application form designed to be less intimidating (e.g. frequency of use questions as described above, minimizing stereotype threat by placing demographic questions at the end), and additional consideration during the selection process, have proven effective.

5 Technical Program

The content of IHPCSS can be divided into: 1) technical program, 2) mentoring program[1] and 3) networking and social events (including welcoming reception, poster sessions, and other formal and informal social outings). Below we describe the technical program in more detail.

5.1 Philosophy

The purpose of the IHPCSS has changed slightly since its inception. Originally, the organizers felt that the greatest impact could be had by selecting the applicants with the most extensive HPC experience. However, the emphasis has since shifted to identifying applicants with moderate HPC knowledge, as the IHPCSS can most benefit students seeking a broad overview rather than a deep dive into a single topic. Additionally, through the content of the applications and the results of the evaluations, we have found that student experience has shifted over time. 15 years ago, most students with HPC experience had written their own parallel programs, typically using MPI; now, many students are using parallel codes written by others, but are seeking help with other aspects of HPC, such as machine learning, I/O performance, or GPU porting.

High performance computing is not a monolithic topic, but rather a broad landscape with many different tools and technologies. By necessity, a week-long program must choose to only focus on a few. The IHPCSS focuses on those elements which 1) will be most useful to early-career scientists in their research and 2) will inform attendees about new trends in HPC so they can stay up-to-date. For example, several years ago a session in Big Data and ML was added to reflect changes in the field. The evaluation process is also key in identifying the knowledge attendees expect to gain, allowing the IHPCSS to be tailored accordingly.

Technical content is presented to the students primarily though hands-on and lecture-style activities. In the hands-on sessions, the goal is to have as much time devoted to student exercises as possible so that students can get real experience with the tools and, hopefully, later apply them in their own work. The evaluation results show that attendees appreciate this hands-on format. The lecture-style sessions usually provide an overview of key concepts, tools, techniques and best practices. Some examples include test-based development, software engineering practices, workflow tools, and optimizing parallel I/O. We encourage instructors to provide real-world examples from their own work so that students can see how these tools are actually used.

The next section summarizes the key changes to the technical program over the years.

5.2 Evolution of Technical Content

In the first Summer School in 2010, the technical content consisted of: mixed MPI/OpenMP programming, algorithmic approaches, numerical libraries and data intensive computing (fast I/O, visualization, CPU performance analysis and profiling). Since then, besides keeping content up to date, we have made a number of changes:

- **2011**: added Workflow Tools.
- **2012**: added GPU programming and Software Engineering.
- **2013**: added Heterogeneous Computing and Directive-Based Programming.
- **2014**: added Python for HPC; split Parallel programming into two parallel tracks: (1) MPI/OpenMP and (2) Accelerator programming (primarily OpenACC).
- **2017**: added High Performance Data Analytics.

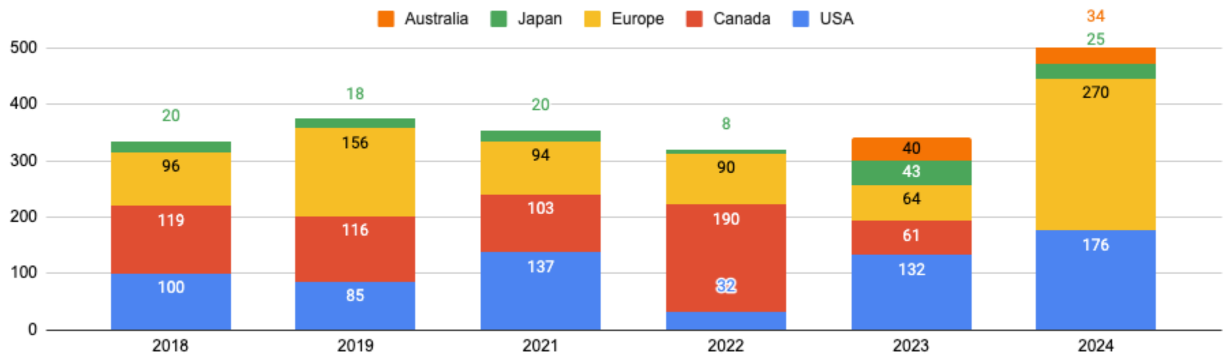

Figure 1: Location of home institution for IHPCSS applicants, 2018-2024. Only the applicants from eligible countries are shown.

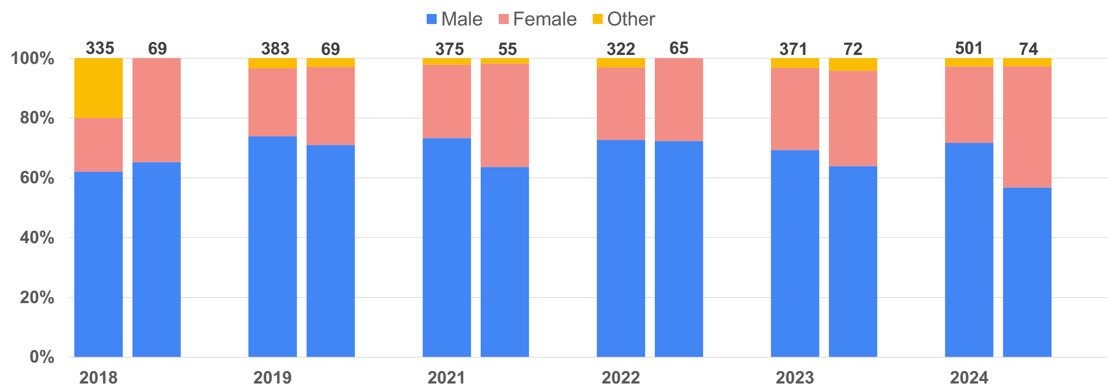

Figure 2: Gender identity for IHPCSS applicants and attendees, 2018-2024. The left bar represents applicants and the right bar attendees. "Other response" includes blanks, "Prefer not to disclose", "non-binary", and others. In the 2018 applications, the question was optional and many chose not to respond.

- **2019**: added Deep Learning; split Performance Analysis and Optimization into single-node and multi-node.
- **2021**: added Machine Learning. This Summer School was presented virtually in two different time slots due to COVID-19.
- **2023**: changed parallel tracks into 1) MPI and 2) OpenMP (instead of OpenACC).
- **2024**: added session on Containers and merged Data analytics and ML sessions into one.

To find time in the schedule for these new topics, the Summer School was extended from 4 days from 2010-12 to 5 days. The science talks (detailed in Section 5.6) were progressively cut back and finally removed. Some sessions which were originally plenary are now offered in parallel (for example, Performance Optimization is now in parallel with Python for HPC). We have also removed sessions which are low-rated in the evaluation or less well suited to the Summer School environment; for example, Numerical Algorithms, though relevant, was difficult to cover in a domain-agnostic way during a short lecture and thus was removed.

5.3 Prerequisites

To fully benefit from the IHPCSS, participants are expected to have specific technical qualifications. Students must be able to use the Linux command line. Familiarity with compiled programming languages such as C, C++, or Fortran is essential, as most examples from the hands-on sessions use these languages. Python is only used in a few sessions but making other sessions friendlier for Python users is being considered.

Although extensive experience with HPC is not essential, participants should have a working knowledge of HPC tools and methodologies to effectively integrate advanced computational techniques into their research. Some level of parallel programming experience, or at least a strong interest in developing parallel programming skills, is highly recommended as the curriculum emphasizes the application of parallel computing concepts.

Over the years, the requirements for the IHPCSS have evolved in response to feedback from participants and organizers. After several iterations, the prerequisites have stabilized to ensure that they are in line with the objectives of the IHPCSS and the diverse backgrounds of the participants. However, slight variations in requirements from year to year are normal, particularly as the curriculum and tracks are updated to reflect emerging trends and technologies.

5.4 Agenda in 2024

In 2024 IHPCSS offered a well-structured program covering a wide range of topics, providing participants with both fundamental knowledge and advanced skills in HPC and Big Data[8], and is quite different from the agenda of the first IHPCSS in 2010[7]. The 2024 agenda included the following topics:

- **Parallel Methodology:** this lecture gives participants an overview of different programming paradigms commonly used in HPC. It also sets context for the technical content of the event and provides information to help participants choose the right sessions from those run in parallel.
- **Shared memory programming:** using OpenMP, participants learn how to use multiple cores on a modern CPU to increase computational efficiency.
- **Distributed memory programming:** participants explore advanced MPI techniques and learn how to improve performance in distributed systems, critical for multi-node scaling.
- **GPU programming:** this session covers the programming of GPUs, a key enabler to accelerate scientific computing and machine learning tasks.
- **Big Data and Machine Learning:** this topic allows participants to gain skills in analyzing large data sets and integrating machine learning.
- **Parallel I/O:** participants learn how to distribute read and write operations across multiple processes, essential for speeding up scientific simulations and data analysis.
- **HPC Python:** participants learn how to leverage Python for computationally intensive tasks.
- **Performance Analysis and Optimization:** participants learn to identify and resolve performance bottlenecks using profiling and tracing tools.
- **Software Engineering:** participants learn best practices for developing and maintaining scientific software to ensure reliability, scalability and collaboration. High quality software is essential for long-term scientific success.
- **Scientific Visualization:** participants learn to use 3D visualization tools to effectively present large datasets, a key skill for communicating scientific results clearly and persuasively.
- **Workflow Tools:** knowledge in this area enables the efficient management and automation of scientific workflows by orchestrating tasks across multiple processes to improve productivity and reproducibility.
- **Numerical Libraries:** participants learn how to integrate already implemented numerical algorithms to save development time and improve performance.

This agenda has evolved through several iterations, with an initial focus on balancing CPU and GPU programming to reflect the core requirements of HPC applications. However, as new technologies such as big data, machine learning and deep learning increasingly influence the computing landscape, the agenda has been adapted to reflect these transformative trends.

The parallel computing program has been structured into two distinct tracks, with the option for participants to indicate their preferred track during the application process. Track 1 focuses on shared-memory parallelism and accelerator programming, utilizing OpenMP for single-node programming optimization and GPU offloading. This track is useful for newcomers to parallel programming, requiring only basic knowledge of compiled languages. Track 2, on the other hand, concentrates on advanced distributed-memory programming using MPI, and is designed for participants with existing MPI experience, i.e. knowledge of point-to-point and basic collective operations. This advanced track delves into collective operations and derived data types. Following these preliminary tracks, the program also covers additional topics listed above. It is important to note that all participants, irrespective of their initial track selection, have the opportunity to attend any subsequent sessions.

5.5 Hands-On Exercises

At IHPCSS, hands-on sessions are delivered in a variety of formats to balance engagement and efficiency. Some sessions follow a "demo mode", where speakers demonstrate examples, allowing participants to observe and later follow instructions independently. This approach requires only a few additional helpers as experienced speakers control the workflow. In sessions where participants are periodically given steps to follow, helpers with HPC experience and some knowledge in the subject provide support. Alternatively, some hands-on sessions involve little direct instruction. Attendees work through detailed exercises, with helpers available for questions. To facilitate this, the "sticky note" method can be effective - participants place a red note on their laptop to indicate they need help, or a green note to indicate they are on track, making it easy for helpers to see who needs assistance. We have found that regardless of the approach used, simplicity is key due to time constraints, and presenters should check in with students frequently to ensure they are following along.

We have tried running some sessions where participants can choose topics on the fly (e.g., through polls). However, this approach forces the presenter to improvise, and we have found it can lead to unstructured and less focused sessions.

5.6 Science Talks

The Summer School started with a balance between science-related sessions, and technical and programming-related presentations. Science presentations covered a variety of computational modeling disciplines such as astrophysics, earth sciences/climate research, life and materials science, and plasma physics. Speakers were leaders in their respective field, and were requested to emphasize the general computational challenges in their discipline so that the material would be accessible to all students, regardless of scientific background. According to the post-event surveys, this goal was difficult to achieve; frequently, the science sessions were the lowest rated sessions by students. For example, in 2023, the last year science talks were held, they had an average rating of 3.58 (on a 5-point scale), while all other technical sessions had an average rating of 4.05. Many students did not see the point of attending talks outside of their domain, and presenters often struggled to find a good balance between scientific background and discussion of their HPC challenges.

The initial Schools were very focused on tools and techniques used in different scientific domains, but over time the focus shifted to more transferable skills. To make space in the agenda for new topics, the number of science talks was reduced from the original eight to four by 2023. In 2024, science talks were removed entirely. However, exposing students to real-life examples of domain science researchers overcoming HPC challenges and including scientists as part of the mentoring staff remained valuable. Therefore, the science talks were replaced by two panels. The first one focused on computational challenges faced in different disciplines, whereas the second one discussed the future of HPC and computational science. The students liked having the opportunity to engage with the experts from different domains, and evaluated both panels highly. We will continue offering the panel sessions during the future events.

5.7 Programming Challenge

The programming challenge originated from a Parallel Programming Bootcamp developed for the National Science Foundation[1], and was driven by the need to address the complexities of teaching hybrid parallel programming, which traditionally requires students to have knowledge of multiple programming models. The overlap between instructors from the Bootcamp and IHPCSS allowed the challenge to be adapted for the Summer School's needs.

The challenge was designed as an optional activity with several key objectives in mind: to validate knowledge, to encourage the use of practical skills, to promote teamwork, and to build confidence in applying these skills to research. These objectives aimed to create a holistic learning experience that went beyond theoretical knowledge. The first implementation of the IHPCSS challenge focused on optimizing a Laplace solver, requiring participants to scale their solution across multiple nodes an HPC system. They were provided with baseline implementations in MPI, OpenMP, and OpenACC (in C, Fortran, and Python) and had to optimize either MPI+OpenMP or MPI+OpenACC (until 2023) for maximum performance, measured by walltime within the solver, while still matching the reference output.

The current implementation of the challenge focuses on optimizing and parallelizing a given PageRank algorithm [13], a graph-based method for ranking the influence of web pages using iterative propagation and a damping factor. Unlike the first challenge, participants compete to maximize iterations within a fixed runtime, using OpenMP for shared memory and MPI for distributed memory programming. We have found that maximizing iterations is a powerful feedback mechanism for students when evaluating optimization strategies, and it also makes grading solutions easy. Teams of 1-3 students can submit multiple solutions before the Thursday midnight deadline.

There are strong constraints imposed to give all students a fair opportunity. Submissions must retain the original software ecosystem and core algorithm. External libraries and changes to timer placement are strictly forbidden. Automated correctness checks validate results. Winners are chosen in multiple categories; for the current challenge, MPI+OpenMP using CPUs and MPI+OpenMP using GPUs. The winner in each category is determined based on the best performance across multiple execution runs. The winners are announced on Friday, the last day of the Summer School.

The IHPCSS programming challenge successfully balances multiple objectives to create an inclusive and effective learning environment. It accommodates different skill levels, from beginners to experienced programmers, while maintaining optional competitive elements. Recent additions such as "Fun Fact" awards, which celebrate interesting discoveries or creative approaches during the challenge, have made the event more inviting and enjoyable. The challenge provides a structured environment with clear rules and multiple programming tracks, encouraging teamwork and the exchange of ideas. The challenge has always had little formal time in the already packed agenda, and runs largely on the enthusiasm of students and instructors who steal that time from sleep and other activities. The end result is a special bond within and among teams and visible pride at the trophy award ceremony.

6 IHPCSS Challenges

Over the years, the content of IHPCSS has evolved to better address the needs of the participants and to reflect the changes in HPC hardware and the resulting changes in programming methodologies. The content and delivery style have been refined to support a positive and effective learning environment for increasingly diverse cohorts of students. Some of those changes introduced new challenges, while others made existing issues harder to mitigate. Most challenges are the result of 1) new HPC hardware, tools and techniques, 2) the participants' diverging skills, interests and priorities and 3) bringing together over 100 strangers in a single location for a week.

6.1 Technical Issues

Keeping the "HPC" in the IHPCSS is an important part of the program philosophy, and distinguishes the program from others. Core to this is the use of actual HPC platforms for as much of the hands-on sessions as possible. This brings greater credibility to the topics being taught, builds confidence in the students in navigating these increasingly intimidating environments, minimizes issues with complex software stacks, and empowers students to continue working on these platforms beyond the Summer School. Reducing the potential for software issues becomes more important every year, as modern software installations, especially in the world of machine learning, have become very deep and fragile. Debugging these during the program's packed agenda would compromise the student experience. Allowing the instructors to create these environments in advance of the event helps minimize student software installation.

As a result, the primary requirement for students is just SSH remote access. This can be easily enabled and tested in advance and requires only a terminal program on their laptop. Some sessions may require other tools, but this can be limited to lightweight packages with limited requirements, while installing software packages like Spark or deep learning containers can be left to the experts on the HPC platforms, where they will also perform much better. More complex software dependencies would require dedicated time for fixing installations at the beginning of the session, so such configurations are generally avoided.

[1] U. S. National Science Foundation, https://www.nsf.gov

The HPC platform dependency requires an ongoing commitment, and the centers that have supported the program have provided both continuity to the program from year to year, as well as real-time support during the event. This responsive support has been critical more than once, and is impressive given that the HPC platform is often in an inconvenient time zone, as much as 12 time zones away from the School location twice in recent years.

6.2 Satisfying Prerequisites

IHPCSS participants come from diverse scientific and technical backgrounds, resulting in a cohort with widely varying programming and software development skills. One of the main trends in recent years was an increase in the number of applications and attendees from the life science domains. The programming needs and technical backgrounds of computational biologists and physicists tend to be very different, so finding the right balance between introductory, intermediate, and advanced materials is challenging.

The school is meant to be accessible and inclusive, so providing students with an opportunity to prepare is important. Ideally, each session should provide well-defined technical and skill prerequisites that students are expected to know. However, this has been a challenge on both sides; presenters do not always have time to prepare additional content in a time frame that is actionable for students, and in turn students do not have much time to prepare. The only prerequisites set for the IHPCSS participants before recent events were: 1) getting access to the HPC system used for sessions, 2) preparing a set of slides and a presentation for the poster session, 3) connecting with their mentor and mentoring group, and 4) installing any software required for specific sessions. In reality, many participants complete the last two points during the event.

Because students often fail to complete recommended refresher content, that content is often recapped at the beginning of the actual session. Moving forward, we hope to improve our process for both obtaining prerequisites from presenters and emphasizing to students the importance of completing these activities in advance.

6.3 Meeting Student Expectations

Students, understandably, attend the Summer School hoping to apply what they learn to their codes and projects. Given the diverse backgrounds of the participants, ensuring the right balance between technical and non-technical content is one challenge, and finding the balance between the breadth and depth of technical content is another. Also, how much focus should be put on science and how much on research software engineering? In attempt to make the School more applicable to students' work, the first several Summer Schools included a "bring-your-own-code" session in which students could discuss technical problems with staff members and hopefully get new ideas. However, a number of challenges were encountered. It was difficult for the students to communicate their coding problem effectively enough to receive useful feedback within the allocated time. This was especially difficult for students used to having scientific discussions not focused on technical details. Finding a staff member with the right level of expertise was also challenging, as students were not always skilled at correctly identifying the nature of their technical barrier. As a result, students often ended up working on the programming challenge or a hands-on activity rather than interacting with staff. More structure, or soliciting problem descriptions from students ahead of time so that staff and students can better match up, would help the bring-your-own-code session be more successful.

6.4 Creating an Inclusive and Supportive Environment

IHPCSS covers a lot of technical and non-technical content in a week, and instructors do not have time to verify the effectiveness of their teaching and adapt it during the event. Therefore, to fully benefit from the content, students need to be proactive in asking questions. Creating an inclusive and supportive environment conducive to learning within the first few hours of the event is challenging. Significant effort has been put into creating the Code of Conduct and Health and Safety Policy [2], and the event starts with a session focused on mentoring and context setting to better manage participants' expectations. The returning mentors (former students returning to IHPCSS as mentors) are effective at breaking down the student-staff barrier by sharing their personal stories of challenges and growth and setting the tone for the event as an open and safe space where anyone can ask for advice - technical or not. Additionally, care is taken to collect and accommodate additional requests or preferences, such as preferred name for the event badge, dietary or learning needs, etc. The students are encouraged to interact with each other and the staff members extensively both during formal and informal contexts.

6.5 Time Management

The IHPCSS agenda is packed with scheduled sessions, informal mentoring and other interactions; in 2024, 48 hours of activities were scheduled over 4.5 days. Every year, new topics are being suggested, so ensuring there is a sufficient number of long, restorative breaks can be a challenge. Not many break and time-keeping issues were reported before the COVID-19 pandemic, however, at the first in person event in 2022 it was a significant issue. The agenda that worked well before proved to be too tight at IHPCSS'22. Many of the presenters ran over time, cutting into the break time and not leaving sufficient time for participants to attend to their needs. Since IHPCSS'23, all coffee breaks are set to be 30 minutes and lunch breaks are 90 minutes. There are also breaks before any evening events. Each session now has a chair, whose job is to ensure their session starts and ends on time. Additionally, for sessions that are run in parallel in non-adjacent rooms, additional time is added to the agenda to provide time for students to move between rooms without missing the start of the next session.

6.6 Local Logistics

In the 15 years of the IHPCSS the Summer School has been hosted in 13 different cities and 14 different locations (IHPCSS'20 was canceled, IHPCSS'21 was online). Both IHPCSS'19 and IHPCSS'24 were hosted by RIKEN in Kobe (in two different venues); all other events had a different local host and coordinator. Effectively communicating venue and event requirements to new local staff every year is critical to holding a successful Summer School. Local hosts have no "insider knowledge" while the organizing committee heavily relies on implicit "institutional knowledge". To date, this communication is piecemeal; planning notes from the previous years exist but are not organized to be shared directly with local staff. This includes everything from room requirements to meal arrangements to name tags to the evening social event. Although this approach has been largely effective thanks to a list of "often-forgotten" items, we are working to formalize the requirements for the Summer School, reducing the risk that something will be forgotten because the wrong person missed a planning call.

7 Conclusion

International, multi-institutional and even multi-continental collaborations are difficult to initiate and maintain. Despite the on-going funding challenges and the expanding number of academic partner organizations, IHPCSS is an excellent example of what can be achieved with the power of collaboration. Each year, an overwhelming number of applications is received relative to the total number of available seats. Since 2010, there was only one year (2020) when IHPCSS did not take place, making IHPCSS'25 its 15th anniversary.

However, keeping the IHPCSS current and relevant for students is an ongoing effort. By integrating the feedback we receive from the evaluation process into planning for the next year, student experience plays a direct role in shaping the Summer School for future attendees. Our experience underscores the importance of a robust evaluation process, so that organizer preferences are balanced by direct feedback. As we look to the 2025 Summer School and beyond, we are confident to be able offer again a highly compelling program, reflecting changes in the state-of-the-art and refined through past feedback, and enabling attendees to benefit from the power of collaboration of highly competent and engaged partners from five continents.

Acknowledgments

We would like to thank all the staff and committee members who have contributed to the IHPCSS since 2010. We thank Elsa Gonsiorowski, Ludovic Capelli, Ramses van Zon, Simon Wong, Matthias Brust and Jolanta Zjupa for their helpful suggestions. We thank Toshiyuki Imamura, Jay Alameda, and Maciej Cytowski for their help describing the review and selection process. We thank all past and current sponsors. IHPCSS'25 is possible thanks to NCSA, TACC, EuroHPC JU and the HPC SPECTRA project, Digital Research Alliance of Canada, RIKEN R-CCS, PAWSEY, EPCC, NICIS CHPC and NITheCS. This research was supported by the Statewide California Earthquake Center (Contribution No. 14153). SCEC is funded by NSF Cooperative Agreement EAR-2225216 and USGS Cooperative Agreement G24AC00072-00.

References

[1] Scott Callaghan, Elsa Gonsiorowski, Weronika Filinger, and Ilya Zhukov. 2024. Developing a Successful HPC Mentoring Program at the IHPCSS *(PEARC '24)*. Association for Computing Machinery, New York, NY, USA. doi:10.1145/3626203.3670519

[2] Scott Callaghan, Elsa Gonsiorowski, Weronika Filinger, and Ilya Zhukov. 2024. IHPCSS: Building Community for Short Training Schools *(PEARC '24)*. Association for Computing Machinery, New York, NY, USA. doi:10.1145/3626203.3670570

[3] Jülich Supercomputing Centre. 2024. Summer School 2024. https://www.fz-juelich.de/en/ias/jsc/systems/quantum-computing/juniq-facility/training/summer_school_2024. [Accessed 06-02-2025].

[4] NICIS CHPC. 2025. CHPC Coding Summer School - 2025 (27 January 2025 - 7 February 2025): Overview · CHPC Events (Indico). https://events.chpc.ac.za/event/145/overview. [Accessed 06-02-2025].

[5] Lizanne DeStefano and Lorna Rivera. 2018. *Evaluation of the 2018 International High Performance Computing Summer School (IHPCSS).* Technical Report. University of Illinois at Urbana-Champaign. https://hdl.handle.net/2022/29630

[6] Lizanne DeStefano and Lorna Rivera. 2019. *Evaluation of the 2019 International High Performance Computing Summer School (IHPCSS).* Technical Report. University of Illinois at Urbana-Champaign. https://hdl.handle.net/2022/29634

[7] IHPCSS. 2010. European-US Summer School on HPC Challenges in Computational Sciences. https://ihpcss.org/past-events/documents/SC10-Program.pdf. [Accessed 23-06-2025].

[8] IHPCSS. 2024. Agenda - International HPC Summer School 2024. https://ss24.ihpcss.org/agenda/. [Accessed 07-02-2025].

[9] Argonne National Laboratory. 2025. Argonne Training Program on Extreme-Scale Computing. https://extremecomputingtraining.anl.gov/. [Accessed 06-02-2025].

[10] Tonya Miles, Julie Wernert, and Lizanne DeStefano. 2021. *EVALUATION OF THE 2021 INTERNATIONAL HIGH PERFORMANCE COMPUTING SUMMER SCHOOL (IHPCSS).* Technical Report. Indiana University. https://hdl.handle.net/2022/29623

[11] Tonya Miles, Julie Wernert, and Lizanne DeStefano. 2022. *EVALUATION OF THE 2022 INTERNATIONAL HIGH PERFORMANCE COMPUTING SUMMER SCHOOL (IHPCSS).* Technical Report. Indiana University. https://hdl.handle.net/2022/29622

[12] Tonya Miles, Julie Wernert, and Lizanne DeStefano. 2023. *EVALUATION OF THE 2023 INTERNATIONAL HIGH PERFORMANCE COMPUTING SUMMER SCHOOL (IHPCSS).* Technical Report. Indiana University. https://hdl.handle.net/2022/29621

[13] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank Citation Ranking: Bringing Order to the Web.* Technical Report 1999-66. Stanford InfoLab. http://ilpubs.stanford.edu:8090/422/ Previous number = SIDL-WP-1999-0120.

[14] Wikipedia contributors. 2023. Distributed European Infrastructure for Supercomputing Applications — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Distributed_European_Infrastructure_for_Supercomputing_Applications&oldid=1191589597. [Online; accessed 1-March-2024].

[15] Wikipedia contributors. 2023. TeraGrid — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=TeraGrid&oldid=1177919988. [Online; accessed 1-March-2024].