A Curriculum for Petascale Computing

Thomas Hacker[†‡] Gary Bertoline[*†]

†Computer and Information Technology, ‡Discovery Park Cyber Center, *Computer Graphics Technology
College of Technology, Purdue University, West Lafayette, IN

1 Introduction

A group of faculty at Purdue is working to develop a curriculum for high performance computing and cyberinfrastructure [1]. Current curriculum efforts are focused on developing and delivering university courses in bioinformatics systems development, grid computing and cyberinfrastructure, algorithms, biomedical informatics, high performance computing systems, and scientific and information visualization. While this effort will address *current* needs for high performance computing and visualization, the recent development of petascale computing - which we define as petaflops of computation attached to petabytes of storage and high speed networks – will require a cohort of highly skilled developers, systems administrators, and science gateway experts who will need training and experience beyond the level available today. Petascale computing represents a new frontier in supercomputing with significant engineering, technology computer science, visualization, and software engineering challenges that must be overcome. In this paper, we propose an enhanced petascale curriculum to address the petascale challenge.

2 The Petascale Frontier

Terascale systems emerged in the late 1990's. At that time, teraflop scale performance could be achieved with thousands of processors running housed in a few dozen cabinets. The Blue Horizon at San Diego Supercomputer Center was a system typical of terascale systems of the era. Blue Horizon contained 1,152 Power3 processors in 42 racks, and consumed approximately 12.5 kW of power per rack. If the growth in computing power over past decade followed Moore's Law alone, systems today would provide around 128 TF of computing power. However, petascale systems under construction today are two orders of magnitude faster and more complex than terascale systems. One example of a recent petascale system is the 0.5 petaflop Ranger system at TACC at the University of Texas at Austin. Ranger is a 504 teraflop system based on Sun blades. Ranger contains 15,744 quad core processors (62,976 cores) in 3,936 nodes, 3 PB of storage, and consumes several megawatts of electricity. The demands of petascale systems for high performance storage, communications, and data centers collectively represent a frontier in performance that must be reached and surpassed to reach levels needed for petascale computing. The demands of petascale systems go beyond challenges in computer engineering. Distilling kilobytes of knowledge from petabytes of data is a difficult challenge today for terascale systems. The output from petascale simulations and data analysis from petascale systems are so voluminous and complex that advanced visualization technologies will be needed to interpret and visualize the data [2].

Petascale systems are more complex than terascale systems. This increased complexity is reflected in the increased number of processors per system, processor density, power consumption, and the effects of scale on reliability, performance, operating systems, and data centers that house production petaflop systems. Petascale computing pushes the envelope of computing to the point where latent factors, such as reliability and power consumption, which did not make a significant impact on computing at a terascale level, becomes a serious problem affecting the usability of petascale systems. These latent effects include: i) power consumption; ii) effects of component reliability on the overall reliability of a system built from thousands of these components; iii) models for writing software designed to run across thousands of processors; iv) the effects of operating systems noise and performance on application performance; and v) the problem of storing, moving, and using the petabytes of data processed by petascale computing systems. Adding to these complexities are the problems of integrating new technologies for petascale computing, such as field programmable gate arrays (FPGAs) and graphics processing units (GPUs), into large-scale computer systems.

Petascale systems are far more intricate and powerful than terascale systems of the past. Operating and utilizing these systems to their full potential requires a higher level of knowledge and skill than is needed today for teraflop systems. An example that illustrates this increased complexity is the tremendous storage capacity and network performance needed to match the computational power of petaflop scale systems. Using a conservative estimate of 200 MB/sec per teraflop of computing, a petascale storage system must deliver data at a rate of 196 GB/second. To match this data rate, data transport and networking systems must be able to transfer data at an aggregate bandwidth of 1.6 terabits per second. Storing only a day's worth of data at this rate will require over 2.7 PB of storage, which is far beyond the capacity of the majority of storage systems used today at universities. Achieving this level of performance is not optional – it is a necessary part of a

complete petascale computing environment.

3 A CURRICULUM FOR PETASCALE COMPUTING

To meet the demands of petascale computing, we propose that the high performance computing community needs an expanded curriculum that expands training and education in several key areas. These areas include: parallel computation and algorithm development; petascale systems architecture and operating systems; petascale systems operations and systems administration; petascale visualization, and science gateway development. The goals of a petascale computing curriculum are to: i) provide training to address the immediate problems of petascale computing; ii) provide course content along with project experience that will remain relevant over a long period of time and establish a foundation for future learning and research; and iii) build on existing practice and knowledge in high performance computing. The curriculum we propose features core courses, as well as specialization tracks in petascale parallel application development, petascale systems, petascale visualization, and broadening access to HPC systems through systems such as science gateways.

3.1 Core Courses

Training in science, technology, engineering, mathematics, or computer science is a necessary building block of a petascale curriculum. Building on this foundation, there are additional core university courses needed for further progression in any of the three petascale curriculum areas. This set of core courses includes: grid computing, numerical analysis and statistics, algorithms, high performance computing systems (focusing on cluster architectures), and database systems.

3.2 Applications Development and Computation Area

Application development and parallel computation require a highly refined set of skills and experiences in analyzing scientific problems, decomposing the problem domain into smaller pieces amenable to parallel computing, and scaling, tuning, and benchmarking codes to improve application performance.

The development cycle of scientific applications that efficiently scales on a large parallel system passes through several phases. Most applications begin with a serial code running on a single workstation, and mature over several years of intensive analysis and incremental improvement into a highly tuned parallel application that scales well up to hundreds and thousands of processors. The process of parallelizing code, which includes analysis of the problem for domain decomposition, identifying the sections of the code that can be parallelized, measuring the speedup, and tuning the code, is an involved and time-consuming process that requires several years of training and experience to reach the level of skill necessary to successfully parallelize code. This process is difficult today for systems that can scale up to a few thousand processors. Only the most mature codes today can efficiently scale up to thousands of processors. Scaling up to tens of thousands of processors in petascale system is a significant challenge, and specialized training in both the algorithmic and mathematical aspects of parallel computing as well as the systems and physical characteristics of the systems is needed. Thus, for application development, we believe that a university course sequence that includes numerical analysis, serial and parallel algorithms, operating systems, networking and communications, software engineering, and training in biology, physics, chemistry, or one of the engineering disciplines will be needed to prepare students to successfully develop efficient high performance applications for petascale systems.

Developing applications for petascale systems will require attention to several areas in particular that have a significant effect on the use of petascale systems. The first is multicore programming, which involves the design of parallel applications for the tens of thousands of cores that are used in petascale systems. This is an active area of research in curriculum design today in computer science and information technology, and developing codes that can run well on multicore systems requires new ways of programming that are inherently parallel. The second area is reliability and fault tolerance. Petascale systems built from thousands of commodity components will experience several component failures per hour, and applications designed to fully exploit petascale systems need built-in resiliency mechanisms to detect and recover from component and software failures, regardless of the source. The final area is scientific workflows and data flow oriented programming. Many applications use open source and commercial libraries, packages, and codes. The ability to break down a problem into discrete computing tasks linked by data flows and to build a solution that couples existing packages with new algorithms and workflows will be a critical skill necessary for developing petascale applications.

3.3 Petascale Systems

The second area, Petascale Systems, is focused on the design, operation, and administration of petascale systems. This area has five main components: data center design and operation, systems design, communications, operating systems, data systems, and novel architectural elements (such as FPGAs) for petascale computing.

The first component focuses on data center design and operation. The tremendous power and cooling demands of petascale systems is driving the development of data centers that can consume a sizeable fraction of the electrical output of a power plant. Data centers designed for petascale systems are evolving to become an integral part of the system architecture on the same level as storage and networking. Education and training in planning, designing, and operating petascale data centers efficiently and safely, and the development of control systems for power and cooling coupled with petascale systems will be an important part of maximizing useful work and minimizing power consumption of petascale systems.

The second component is design, which involves the engineering and development of petascale supercomputing systems. Although petascale systems will be available as "turn-key" systems from vendors, a deep understanding of the computer engineering and architecture of petascale systems, as well as the tradeoffs and benefits of different design approaches are needed by users, owners, and operators of petascale systems.

The third component, communications, is one of the most important aspects of a petascale systems curriculum. Parallel computing inherently involves the sharing of data among processors, instruments, and data repositories, which are all connected with high speed networks. Detailed knowledge and skills in the design, tuning, operation, and use of high performance communication systems, ranging in scale from cluster level to wide-area networks, is a critical part of achieving efficient petascale performance. Communications can be further subdivided into several partitions: cluster level communication, which focuses on parallel interconnections such as infiniband and myrinet; local area and storage area networking, which involves the integration of instruments, facilities, and data sources on a campus level; and wide-area networking, which is the most critical part of building a successful cyberinfrastructure.

In addition to data centers, systems design, and communication, the fourth component of the petascale systems area is operating systems. Operating systems form the core of a parallel computing system, and a high level of expertise in installing, optimizing, maintaining, and modifying operating systems will be needed. Many large systems today use specialized low-impact operating systems, such as Catamount [3], that consume as few system resources as possible to ensure excellent application performance. Petascale systems will also require light-weight operating systems, as well as support for virtualization to allow developers to optimize operating systems for maximum application performance.

The fifth and final component is data systems. As mentioned earlier, achieving petascale performance depends on breaking through the storage frontier of high performance and high capacity. Designing and operating systems of this scale today is difficult and requires specialized training in: parallel file systems; parallel and scientific databases; streaming data systems and workflows, streaming SQL, and the integration of sensors and instruments. Maximizing data reuse will be an important challenge for petascale computing. Enormous amounts of power and cooling will be consumed running applications that produce output data. Since less power is needed to store these results on disk, users of petascale systems will need to learn to reuse petascale data. Adopting approaches developed by library community to manage the entire data life cycle, which includes indexing and cataloging data, data archiving, and active curation of data sets, will be an important addition to the curriculum.

A final component of this area is the exploration, integration, and use of novel architectures such as field programmable gate arrays (FPGAs) and graphics processing units (GPUs) into petascale systems architectures.

The components that comprise petascale systems could be included as course modules in university courses, or presented as modules in a tutorial. Collectively, material for this area could be included in an upper level introductory course and an advanced graduate level course.

3.4 Petascale Visualization

Extracting meaning from large-scale simulations created from petascale computing that produces hundreds of terabytes or more of data is a challenge facing researchers. Computer graphics and visualization specialists must be trained in new critical areas of focus that includes parallel and distributed visualization technologies, support for complex datasets, and discovery enabling visualization and analysis techniques [3].

Parallel and distributed visualization at the petascale level requires training in parallel visualization and rendering techniques, GPU processor accelerated visualization techniques, in-situ data processing and visualization, and remote, collaborative data analysis and visualization. Training in support for complex data sets includes visualization of adaptive mesh data, organization of datasets, and time-varying data visualization. Discovery enabling visualization and analysis training would include tracking and visualizing the evolution of features in time varying data, feature detection and tracking, error and uncertainty in data visualization, and user interface development for visualization frameworks that integrates into the discovery workflow.

3.5 Science Gateways

The final curriculum area is focused on broadening the use of petascale computing to groups that traditionally have not used supercomputing, or who cannot invest the time needed to develop and parallelize applications and can benefit from the use of applications developed and shared by others. One example of an emerging application that is scaling up to petascale that could effectively be provided through a science gateway is NEMO and OMEN, developed by Professor Gerhard Klimeck at Purdue University, which was recently awarded a National Science Foundation PetaApps award. Science gateways are an attractive approach to proving access to a petascale system for a large community of users, and well designed systems (such as the Purdue nanoHUB [4]) have demonstrated success in reaching tens of thousands of users – many of whom have never previously used supercomputers. The Science Gateway area requires skills and knowledge in several areas: computer graphics; human interface design; web programming; educational technology and integration with learning management systems; and scientific workflows and database systems. The curriculum for this

area must focus on usability as well as maximizing the education and research impact of science gateways for communities ranging from middle school science courses to global science collaborations involving thousands of researchers using cyberinfrastructure.

6 CONCLUSIONS

Petascale computing is a new frontier for high performance computing that will help to drive advances in engineering, technology, computer science, and the domain sciences. Training developers and users of petascale computing will require a deepening and broadening of current curriculum approaches for high performance computing. In this paper, we proposed an enhanced petascale curriculum to prepare students to develop and use new technologies to successfully meet the challenges in this new frontier.

REFERENCES

[1] Thomas Hacker, John Springer, Shannon Schlueter, and Michael Kane, "Developing a Curriculum for High Performance Computing and Cyberinfrastructure Education", Proceedings of the American Society for Engineering Education Conference for Industry and Education Collaboration, February, 2008, New Orleans, LA.

[2] Kwan-Liu Ma, Robert Ross, Jian Huang, Greg Humphreys, Nelson Max, Kenneth Moreland, John D. Owens, and Han-Wei Shen, "Ultra-Scale Visualization: Research and Education", Journal of Physics: Conference Series 78, SciDAC, IOP Publishing, pp. 1-6, 2007.

[3] Trammell Hudson and Ron Brightwell "Network Performance Impact of a Lightweight Linux for Cray XT3 Compute Nodes", SC'2006 Conference CD, IEEE/ACM SIGARCH, November 2006.

[4] G. Klimeck, M. Korkusinski, H. Xu, S. Lee, S. Goasguen, and F. Saied, "Building and Deploying Community Nanotechnology Software Tools on nanoHUB.org and Atomistic simulations of multimillion-atom quantum dot nanostructures", Proceedings of the 5th IEEE Conference on Nanotechnology, Vol 2, pp. 807, 2005.