

High Performance Computing: Infrastructure, Application, and Operation

Byung-Hoon Park* and **Youngjae Kim**

Computing and Computational Sciences Directorate, Oak Ridge National Laboratory, Oak Ridge, TN, USA
parkbh@ornl.gov, kimy1@ornl.gov

Byoung-Do Kim

Advanced Research Computing, Virginia Tech, Blacksburg, VA, USA
bdkim@vt.edu

Taeyoung Hong and Sungjun Kim

Supercomputing Center, Korea Institute of Science and Technology Information, Daejeon, Korea
tyhong@kisti.re.kr, sjkim@kisti.re.kr

John K. Lee

Appro Inc., Milpitas, CA, USA
johnlee@appro.com

Abstract

The last decades have witnessed an increasingly indispensable role of high performance computing (HPC) in science, business and financial sectors, as well as military and national security areas. To introduce key aspects of HPC to a broader community, an HPC session was organized for the first time ever for the United States and Korea Conference (UKC) during 2012. This paper summarizes four invited talks that each covers scientific HPC applications, large-scale parallel file systems, administration/maintenance of supercomputers, and green technology towards building power efficient supercomputers of the next generation.

Category: Embedded computing, Green computing

Keywords: High performance computing; Supercomputer; Parallel file system; Green technology

I. INTRODUCTION

A supercomputer is a computer that integrates the most advanced technologies available today, thus considered to be at the cutting edge of computational capability. Scientists and engineers incessantly try to push the frontiers

with technological innovations and renovations. High performance computing (HPC) applies supercomputers to highly compute intensive problems of national concerns (a narrower definition of HPC may have a different meaning. However, we use HPC and supercomputing interchangeably in this paper). In this regard, HPC tech-

Open Access <http://dx.doi.org/10.5626/JCSE.2012.6.4.280>

<http://jcse.kiise.org>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 21 October 2012, Revised 28 October 2012, Accepted 4 November 2012

*Corresponding Author

nology is generally acknowledged for its imperative importance to national security and global competition. Table 1 lists the fastest supercomputers for the past five years. As shown, the United States, China, and Japan have exerted much effort to produce faster supercomputers, taking over the No. 1 spot in turns. The race seems to remain competitive. The United States has already laid out a plan to produce a machine capable of performing 10^{18} floating point operations per second (or exaFLOPS) by the end of this decade, i.e., machine 1,000 times faster than Roadrunner [1, 2]. Japan, China, Russia, and the European Union also plan to produce the exascale machine in the near future.

HPC has become an indispensable part of scientific communities changing the research paradigm. More and more scientists rely on highly intensive computation to posit/validate new theories, and to make long-sought discoveries. However, for many scientists, HPC is still a new and alienated topic. The United States and Korea Conference (UKC) of 2012 organized the HPC session for the first time ever to bridge this gap. The session was intended to introduce HPC to a broader community, and highlight several important aspects that comprise HPC. Five invited talks were presented during the session, four of which addressed technical areas of HPC and the last one reported the launching of the HPC special interest group. This paper is a summary of the four technical presentations.

The paper is organized as follows. It first reviews practical issues in conducting large-scale scientific computing projects; how interdisciplinary collaborations can create synergetic outcomes not otherwise possible, and some of the challenges faced by scientists in the new era of computing. Second, issues in provisioning I/O systems are discussed. In particular, it introduces lessons learnt from provisioning and maintaining the Spider file and storage system of the Oak Ridge Leadership Computing Facility (OLCF), and discusses the next-generation parallel file system architecture to meet the ever growing I/O requirement of the upcoming 20-petaflop Titan supercomputer

Table 1. The world's fastest supercomputers between 2008 and 2012

Year	Name	Peak speed (petaflops)	Location
2012	Sequoia (IBM)	16.32	Lawrence Livermore National Laboratory, USA
2011	K-computer (Fujitsu)	10.51	RIKEN Advanced Institute of Computational Science, Japan
2010	Tianhe-1A	2.566	National Supercomputing Center, China
2009	Jaguar (Cray)	1.75	Oak Ridge National Laboratory, USA
2008	Roadrunner (IBM)	1.026	Los Alamos National Laboratory, USA

and other computing resources. Third, monitoring system health through log data analysis is introduced. As an example, two years of event logs generated from the Tachyon2 system of Korea Institute of Science and Technology Information (KISTI) is described. Fourth, power consumption of the supercomputer, an urgent and difficult problem towards building the exascale system is discussed. In particular, this paper discusses how commodity technology can be utilized to appease the power consumption problem. Finally, discussions on the next generation machines conclude the paper.

II. SCIENTIFIC HPC APPLICATIONS

Computational science is an effort to solve science and engineering problems using mathematical models and analysis techniques. Often times, HPC catalyzes computational simulation in terms of scale and speed, allowing scientists to tackle problems of unprecedented scale. A list of areas where HPC plays an important role includes, but is not limited to, chemistry, climate science, computer science, geoscience, astrophysics, biology, nuclear fusion, physics, and various engineering fields. In fact, the advance of supercomputers is largely accredited to the ever-increasing demand for computational power in these fields. Many applications in these fields scale up to a large computational domain in order to solve, typically millions of grid points, which normally require tens of millions CPU hours to run.

Due to its large scale by nature, utilizing HPC resources in scientific research requires a multidisciplinary, collaborative effort in order to complete the workflow of research projects. For example, a bioinformatics project would require expertise from the fields of biology, data mining, statistics, and HPC. This approach is becoming an ideal model for transformative knowledge based study. The HPC research community is also facing technical challenges: 1) the advancement of hardware has already outpaced software development especially in parallelism and performance optimization [3]; 2) data being created through the HPC is exponentially increasing; 3) as stated above, the interdisciplinary and collaborative research environment demands a higher level of understanding in a wider spectrum of other science domains than ever before.

Performance engineering of applications in the HPC environment is also a tall order as the scales of system hardware and parallel applications are rapidly increasing. There are many performance-profiling tools that assist in the optimization of HPC applications, but most of the tools are developed for experts with advanced knowledge in optimization tools that enable application developers to understand the mechanisms of performance engineering in the HPC environment [4-6].

Another noticeable trend in computational science research is collaborative work in various fields, as HPC has become a

critical component of modern science and is fundamentally transforming how research is done. In this new environment, scientists no longer work exclusively within a single domain, but instead conduct their work with a more collaborative and multidisciplinary approach. This new research modality demands more than scientific experts; it also requires effective communication and management. Even with the dramatic advance of supercomputers with regards to speed and capacity, the way they are operated and used is still in its infancy. To efficiently utilize a supercomputer for an application, a computational scientist is expected to possess not only a keen understanding of the parallel algorithms but also the run-time environment of the target system. A modern supercomputer consists of sophisticated subsystems such as a high speed interconnect, parallel file system, and graphics accelerators. It is therefore quite difficult for an ordinary end user to utilize a system as many end users are not properly educated and software and system interfaces are not intuitive and clear.

HPC empowers the computational science and engineering research by providing an unprecedented scale of computational resource as well as data analysis and visualization capacity. It will continue to play a key role for the advances of scientific research. In this section, the importance of sharing knowledge and effective communication among various domains, and continuous education are discussed. Synergic outcomes of a multidisciplinary collaboration that addresses these crucial issues will be unlimited.

III. PARALLEL FILE SYSTEM

As discussed in the previous section, large-scale scientific applications generally require significant computational power and often produce large amounts of data. An inefficient I/O system therefore creates a serious bottleneck degrading the overall performance of applications and the system. In general, however, the performance of an I/O system and its capacity does not scale proportionally to meet the need. This section discusses the parallel file system as a solution to the issue by introducing the Lustre file system in the Oak Ridge National Laboratory (ORNL) [7].

To meet the ever-growing demand of scientists, the capability of a supercomputer is upgraded every few years. For example, Jaguar of the OLCF at ORNL is now under an upgrade phase. In 2010, when it was ranked as the world's No. 1 supercomputer, Jaguar had 18,688 compute nodes with 300 TB of memory. More specifically, it had 224,256 compute cores that together produced 2.3 petaflops. As of February of 2012, Jaguar's compute nodes were replaced with AMD 16-core Opteron processors, and a total of 299,008 cores with 600 TB of memory. Currently, NVIDIA Kepler graphics processing unit (GPU) accelerators are being added into 14,592 compute nodes.

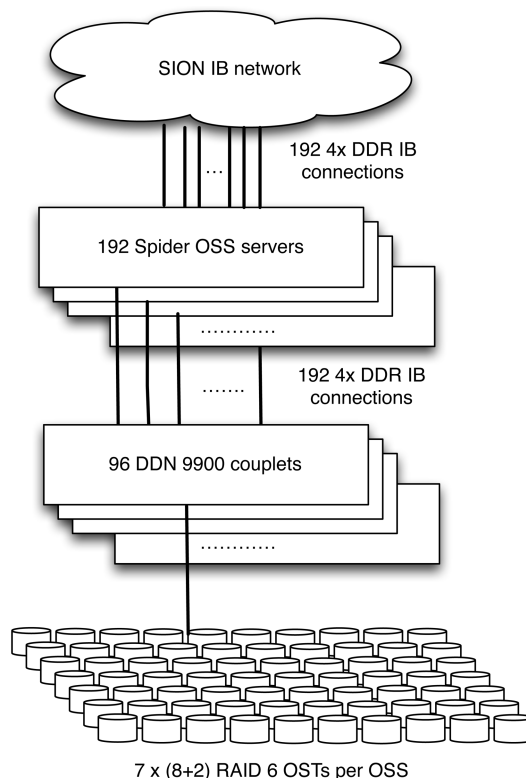


Fig. 1. Spider file system. SION: scalable I/O network, IB: InfiniBand, DDR: double data rate, OSS: object storage server.

Once completed, the upgraded machine will be called Titan, which aims to offer over 20 petaflops. In regards to the capability of Jaguar, at least in terms of flops, Titan will be 10 times as powerful as the current one, and the I/O need of the upgraded system is also expected to increase as much.

Among several choices in file systems, Lustre is probably the most widely adopted for supercomputers. As of June of 2012, fifteen of the top 30 supercomputers use Lustre including No. 1 IBM Sequoia [8]. Its popularity comes from its inherent scalability. Lustre is an open source parallel distributed file system. It separates metadata and actual file data. The entire namespace is stored on metadata servers (MDSs), whereas file data are kept in object storage targets (OSTs) that are managed by an object storage server (OSS). In other words, MDSs contain information about a file such as its name, location, permission, and owner. The actual file data are stored across multiple OSTs. In summary, the main advantage of Lustre (and parallel file system in general) is that it offers a global name space through MDSs and distributes files across multiple nodes through OSSs, thus providing load balancing and scalability.

Spider, the main file system of OLCF, is a Lustre-based storage cluster of 96 DDN 2A9900 RAID controllers providing an aggregate capacity of over 10 petabytes

from 13,440 1-terabyte SATA drives [7]. The overall Spider architecture is illustrated in Fig. 1. Each controller has 10 SAS channels through which the backend disk drives are connected. The drives are RAID 6 formatted in an 8+2 configuration requiring disks to be connected to all ten channels. The current configuration connects fourteen disks per channel, thereby each controller is provisioned with 4 tiers and overall each couplet has 28 RAID 6 8+2 tiers. Each controller has two dual-port 4x DDR IB host channel adapters (HCAs) for host side connectivity. Access to the storage is made through the 192 Lustre OSSs connected to the controllers over InfiniBand (IB). Each OSS is a Dell dual socket quad-core server with 16 GB of memory. Four OSSs are connected to a single couplet with each OSS accessing 7 tiers. The OSSs are also configured as failover pairs and each OSS connects to both controllers in the couplet. The compute platforms connect to the storage infrastructure over a multistage IB network, referred to as scalable I/O network (SION), connecting all of OLCF.

Understanding I/O bandwidth demands is a prerequisite step to deploy an I/O system of the next generation. This process, technically called I/O workload characterization, aims to project future I/O demands by creating a parametric I/O workload model using observed I/O data of the current system. A proper validation of the model is also a crucial step in this process. In fact, design of the storage system should seamlessly integrate three steps to avoid either under or over provisioning of I/O systems: 1) collection of data, 2) design and construction of the system, and 3) validation of the system.

To design and provision the next Spider system of 20 petabyte scale, OLCF monitors a variety of metrics from the back-end storage hardware such as bandwidth (MB/sec) and input/output operation per second (IOPS). More specifically, a custom-built utility tool was developed that periodically collects statistical data from DDN controllers. The period of sampling can be set to 2, 6, or 600 seconds. All data is archived in a MySQL database for further analysis. To characterize workloads, OLCF currently studies five metrics in depth based on knowledge and expertise collected over four years of operation experience.

- I/O bandwidth distribution
- Read to write ratio
- Request size distribution
- Inter-arrival time of I/O request
- Idle time distribution

A workload characterization study using DDS's performance data revealed interesting lessons [9]: 1) I/O bandwidth can be modeled in a Pertomodel, which means storage bandwidth is not efficiently utilized, 2) read requests (42%) are closely as many as write requests (58%), and 3) peak bandwidth is observed at a 1 MB request.

Current processor technology is moving fast beyond the era of multi-core towards many-core on-chip. The

Intel 80 core chip is an attempt at many-core single chip powered data centers. The growing processing power demands the development of memory and I/O subsystems, and in particular disk-based storage systems, which remain unsolved. Recent advances in semiconductor technology have led to the development of flash-based storage devices that are fast replacing disk-based devices. Also, the growing mismatches between the bandwidth requirements of many-core architectures and that provisioned through a traditional disk based storage system is a serious problem.

IV. MONITORING SUPERCOMPUTER HEALTH THROUGH LOG DATA ANALYSIS

KISTI has a long history of HPC in Korea. Founded as the computer laboratory of the Korea Institute of Science and Technology (KIST) in 1967, KISTI provides national computing resource services for scientists and industry. In 1988, it deployed the first supercomputer of Korea, Cray-2S in 1988. Ever since, KISTI has operated more than a dozen world-class supercomputers and accumulated a great deal of knowledge on maintaining the systems. However, as the size and complexity of a supercomputer grows at an unprecedented rate, the center experiences system faults more frequently than ever. Whereas faults are unavoidable, understanding their occurrence behaviors from various perspectives is essential to minimize system loss and the applications running therein. To this end, KISTI analyzed two years of its main system event logs to characterize faults behaviors in an attempt to identify important fault types either atomic or composite and extract event signatures essential to assess system health.

Tachyon2, the mainframe of KISTI, consists of 3,200 compute nodes that are stored in 34 Sun Blade 6048 racks. Each rack includes four shelves and each has 24 x6275 blades. Each blade has two quad-core Intel Nehalem processors and 24 GB of memory, and a 24 GB CF flash module. Networking between nodes is done through IB. Four QDR IB constitute the non-blocking IB network. All compute nodes are connected to eight Sun Datacenter IB 648 switches and other infra nodes are connected to six Sun Datacenter IB 36 switches. For the file system, 36 Sun x4270 servers and 72 J4420 provides 59 TB of user home space and 874 TB of scratch spaces. These spaces are made available to users through the Lustre file system. Overall, Tychon2 serves approximately 1,000 scientists from more than 20 different science domains.

For the study, three log sources, *syslog*, *conman*, and *opensm*, each reporting different aspects of system status, are studied. Table 2 lists some events extracted by the study. These events were generated by a hybrid approach: manual investigation by domain experts and the text/data mining approach to cluster log entries. These events have been confirmed to be important for monitoring system health and possible root causes of some faults. It should

Table 2. Event examples and their log source

Log type	Event
<i>syslog</i>	Soft lockup
	Lustre error
	I/O error
	Backend file system error
	Machine check exception Segfault
<i>conman</i>	Oops
	Kernel panic
<i>opensm</i>	Node state unknown
	Link state change

be noted that some events are of composite type, and can thus be further decomposed into sub events. For example, a Lustre error can be classified based on the component that generates it such as OSS, OST, MDS, OSC (compute node), etc. Some events convey insightful information about system status if they are viewed from a *sender/target* perspective. A Luster event, as another example, typically reports a problem between peers, and thus contains three pieces of information: reporting component, target component, and content. The classification of events in this manner is useful to assess the health of the system when few components are the source of the problem and a tsunami of log events is generated. Through an appropriate aggregation of the information, a tsunami of events may collectively represent a simple and clean view of the system. Some visual aiding tools for this purpose are also available [10].

V. CONSTRUCTION OF GREEN SUPERCOMPUTERS

The US Department of Energy recently awarded ten million dollars to companies like Intel, AMD, NVIDIA, and Whamcloud to initiate research and development to build exascale supercomputers. These companies will attempt to advance processor, memory, and I/O system technologies required to build exascale computers. However, even with advanced modern technologies, to build a computer capable of executing 10^{18} flops is not an easy task. In the center of technological barrier lies power consumption. An ordinary supercomputer consumes a significant amount of power. Most of the energy consumed is transformed into heat, which also requires additional power for cooling. Approximately, it costs three to four million dollars per year to operate a 10 petaflops machine.

Tightly interconnected microprocessors (and graphics accelerators) constitute modern supercomputers. For the past several decades, microprocessors have become smaller and more powerful (Moore's Law [11]). Some studies demonstrate that iPad2 is as fast as the Cray-2 supercom-

puter [12], the worlds' fastest supercomputer in the mid-1980s. However, such a trend seems to be hitting a wall – the clock speed of a microprocessor is stuck at 3 GHz. It is mainly due to the fact that the amount of transistors needed to achieve higher clock speeds will generate unimaginable amounts of heat; in other words they are *power hungry*. This essentially indicates that it is impractical to expect that a supercomputer would advance at the same rate considering the amount of power required to support the projected scale.

Blue Waters is a supercomputer that will be deployed at the National Center for Supercomputer Application (NCSA). Blue Waters, in operation, will consume 15 mW for the performance of 10 petaflops. Linearly projected, ten times as much power will be needed to construct an exascale machine (exa is 1,000 peta). As a result, some people jokingly claim to build a sizable nuclear power plant next to each supercomputer center.

There exist evolutionary and revolutionary steps to address power consumption problems leveraging commodity technologies. The evolutionary approaches are efforts to save energy through current design improvements. For example, a redesigned platform of a blade can save 20–30% of power for individual blade servers by making a group of blade servers share power supplies and cooling. Also, a more efficient power distribution can save more energy. Traditional AC power distribution systems in North America provide 480 V 3-phase power into a facility. The voltage is typically converted down to 208 V 3-phase during which an energy loss of 3% to 5% is incurred. Also, operating power supplies at 208 V also results in efficiency loss. Efficiency gains are thus realized when supplying 480 V directly into the rack, avoiding the voltage conversion. Furthermore, operating at 277 V not 208 V could gain an additional 2% efficiency gain. In summary, 480/277 V AC power delivery is the most efficient power distribution method in North America.

Liquid cooling is a revolutionary approach to solve the problem of power consumption. As in an automobile, liquid circulates through heat sinks attached to processors inside a rack, and transfers the heat to the ambient air. Then the liquid is cooled and circulates through the heat sinks again. Comprehensive power monitoring should be preceded in order to achieve a detailed view on power consumption. For this, system architecture should be designed to monitor the power consumptions of individual nodes, fans, and platform management devices independently. This will provide capability to estimate the power consumption of each component accurately over time so that provisioning and/or capping power at an individual or a group of components becomes possible.

VI. DISCUSSION AND CONCLUSION

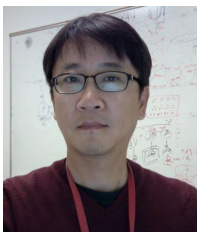
HPC has changed the way scientific research is con-

ducted, and its role will continue to increase. However, for many scientists, HPC is still a distant or uncomfortable area to access. This brief paper intends to fill such a gap by introducing the four areas of HPC: HPC applications, file systems, system health monitoring through log data analysis, and green technology. The paper is by no means a comprehensive review on HPC; many important areas are not covered such as the parallel programming environment, hybrid system, fault tolerant system, and cloud computing, to name a few.

Computational demands of scientists have advanced HPC, pushing the limit of its capacity and capability. Current efforts are exerted to build exascale systems. As always, there are many technological challenges, and both incremental and revolutionary solutions have been proposed to address diverse issues. Apart from the areas this paper discussed, the architecture and software programming paradigm will undergo dramatic changes creating a wide spectrum of research and engineering challenges. More specifically, hybrid architectures that blend traditional CPUs with GPUs or many-integrated cores (MICs) will be explored extensively requiring different programming paradigms, respectively. System and software resiliency is another crucial issue. A supercomputer of current days consists of millions of CPU cores with billions of threads running. It is thus considered norm that exascale systems with their greatly increased complexities will suffer various faults many times a day, which will severely afflict the successful run of an application. Overall, these problems are equally important and should be collectively addressed and scrutinized in order to reach the goal.

REFERENCES

1. J. Dongarra, P. Beckman, T. Moore, P. Aerts, G. Aloisio, J. C. Andre, et al., "The international exascale software project roadmap," *International Journal of High Performance Computing Applications February*, vol. 25, no. 1, pp. 3-60, 2011.
2. K. Alvin, B. Barrett, R. Brightwell, S. Dosanjh, A. Geist, S. Hemmert, M. Heroux, D. Kothe, R. Murphy, J. Nichols, R. Oldfield, A. Rodrigues, and J. S. Vetter, "On the path to exascale," *International Journal of Distributed Systems and Technologies*, vol. 1, no. 2, pp. 1-22, 2010.
3. J. Diamond, M. Burtscher, J. D. McCalpin, B. D. Kim, S. W. Keckler, and J. C. Browne, "Evaluation and optimization of multicore performance bottlenecks in supercomputing applications," *Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software*, Austin, TX, 2011, pp. 32-43.
4. M. Burtscher, B. D. Kim, J. Diamond, J. McCalpin, L. Koesterke, and J. Browne, "PerfExpert: an easy-to-use performance diagnosis tool for HPC applications," *Proceedings of the ACM/IEEE Conference for High Performance Computing, Networking, Storage and Analysis*, New Orleans, LA, 2010, pp. 1-11.
5. N. Tallent, J. Mellor-Crummey, L. Adhianto, M. Fagan, and M. Krentel, "HPCToolkit: performance tools for scientific computing," *Journal of Physics: Conference Series*, vol. 125, no. 1, 2008.
6. O. A. Sopeju, M. Burtscher, A. Rane, and J. Browne, "AutoSCOPE: automatic suggestions for code optimizations using PerfExpert," *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, Las Vegas, NV, 2011, pp. 19-25.
7. G. M. Shipman, D. A. Dillow, H. S. Oral, Y. Kim, D. Fuller, J. Simmons, and J. Hill, "A next-generation parallel file system environment for the OLCF," *Proceedings of the Cray User Group Conference*, Stuttgart, Germany, 2012.
8. Top500 supercomputer sites, <http://www.top500.org>.
9. Y. Kim, R. Gunasekaran, G. M. Shipman, D. A. Dillow, Z. Zhang, and B. W. Settlemyer, "Workload characterization of a leadership class storage cluster," *Proceedings of the 5th Petascale Data Storage Workshop*, New Orleans, LA, 2010, pp. 1-5.
10. B. H. Park, G. Kora, A. Geist, and J. Heo, "RAVEN: RAS data analysis through visually enhanced navigation," *Proceedings of the Cray User Group Conference*, Edinburgh, UK, 2010.
11. G. E. Moore, "Cramming more components onto integrated circuits," *Proceedings of the IEEE*, vol. 86, no. 1, pp. 82-85, 1998.
12. J. Dongarra and P. Luszczek, "Anatomy of a globally recursive embedded LINPACK benchmark," *Proceedings of the IEEE High Performance Extreme Computing Conference*, Waltham, MA, 2012.



Byung-Hoon Park

Byung-Hoon Park is a research staff member at Computer Science Research Group of Computer Science and Mathematical Division, Oak Ridge National Laboratory. He received B.S. degree in computer science from Yonsei University, and M.S. and Ph.D. degree from Washington State University, respectively. He has participated in a number of DOE and DHS funded projects such as SciDAC Scientific Data Management Center, Coordinate Infrastructure for Fault Tolerant Systems, Biological Text Mining, and High-throughput Data Analysis and Modeling for Genomes to Life, and BioEnergy Center. His research interest includes fault tolerant computing and system health monitoring through log data analysis, anomaly detection from big data databases, and computational biology.



Byoung-Do Kim

Byoung-Do Kim is a Deputy Director of HPC in Advanced Research Computing at Virginia Tech. His research interests include large-scale applications performance engineering and optimization in HPC environment, development of computational models for various parallel computing systems, and computational fluid dynamics and heat transfer. He earned his Ph.D. degree in Aeronautics and Astronautics Engineering from Purdue University in 2002, and worked for National Center for Supercomputing Applications (NCSA) at University of Illinois at Urbana-Champaign and Texas Advanced Computing Center (TACC) at The University of Texas at Austin.



Youngjae Kim

Youngjae Kim is an I/O Systems Computational Scientist for Center for Computational Sciences at Oak Ridge National Laboratory. He is responsible for various research and development projects for data storage, data management and data analysis. He received the B.S. degree in computer science from Sogang University, Korea in 2001, the M.S. degree from KAIST in 2003 and the Ph.D. degree in computer science and engineering from Pennsylvania State University in 2009. His research interests include operating systems, parallel I/O and file systems, storage systems, emerging storage technologies, and performance evaluation. He is currently an adjunct professor in the school of electrical and computer engineering at Georgia Institute of Technology.



Taeyoung Hong

Taeyoung Hong is a senior researcher at KISTI. He received his B.S. and M.S. degree in physics from Sungkyunkwan University. He is currently responsible for administrating "Tachyon" supercomputer system. He has been involved in evaluating, acquiring, installing and operating supercomputers for KISTI and other government agencies for 9 years. His research interest includes effective monitoring and management of large-scale HPC system, parallel file system, high performance networking, and optimization of MPI and openMP applications.



Sung Jun Kim

Sung Jun Kim received his B.S. degree in Computer Science from Hannam University, Daejeon, Korea in 2000. And his M.S. degrees was given in Computer Science and Engineering from Hannam University in 2002. Since 2002, he is a senior researcher at Supercomputing Center in Korea Institute of Science and Technology (KISTI). His recent research interests include handling large amounts of log data, monitoring system and network status.



John K. Lee

John K. Lee joined Appro in 2001. John provides Operations leadership strategy direction to Appro in partnership with the CEO and senior staff. In this role, he heads the Operations Group planning, coordination and growth initiatives for manufacturing and deployment of Appro's products and complex supercomputing solutions. He is also responsible for leading Appro HPC Servers, Storage and Supercomputer Solutions Product Development Group. This portfolio provides the foundation for Appro's converged HPC Solutions, the company's strategy for next-generation supercomputers that enables customers to increase agility, lower costs of operations and drive innovation into their businesses. Prior to his role at Appro, Mr. Lee served in both Sales and Service Management capacities at multiple storage and telecom companies. Mr. Lee graduated from University of California, Los Angeles.