REVIEW

National Science Review
3: 36–48, 2016
doi: 10.1093/nsr/nww001
Advance access publication 20 January 2016

COMPUTER SCIENCE

Special Topic: High Performance Computing

High-performance computing environment: a review of twenty years of experiments in China

Zhiwei Xu[1,*], Xuebin Chi[2] and Nong Xiao[3]

ABSTRACT

A high-performance computing environment, also known as a supercomputing environment, e-Science environment or cyberinfrastructure, is a crucial system that connects users' applications to supercomputers, and provides usability, efficiency, sharing, and collaboration capabilities. This review presents important lessons drawn from China's nationwide efforts to build and use a high-performance computing environment over the past 20 years (1995–2015), including three observations and two open problems. We present evidence that such an environment helps to grow China's nationwide supercomputing ecosystem by orders of magnitude, where a loosely coupled architecture accommodates diversity. An important open problem is why technology for global networked supercomputing has not yet become as widespread as the Internet or Web. In the next 20 years, high-performance computing environments will need to provide zettaflops computing capability and 10 000 times better energy efficiency, and support seamless human-cyber-physical ternary computing.

Keywords: cyberinfrastructure, e-Science environment, middleware, supercomputing

[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China; [2]Supercomputer Center of Chinese Academy of Sciences, Beijing 100190, China and [3]State Key Laboratory of High Performance Computing, National University of Defense Technology, Changsha 410073, China

*__Corresponding author.__ E-mail: zxu@ict.ac.cn

Received 24 September 2015; **Revised** 22 December 2015; **Accepted** 29 December 2015

INTRODUCTION

High-performance computing (HPC), also called supercomputing, has become an essential tool for science and engineering. A high-performance computing environment (HPCE) is a system that connects users' applications and supercomputers that belong to multiple institutions. An advanced HPCE connects scientific computing users, data, applications software, middleware, and supercomputer centers and integrates them into a single research environment. This is a crucial system that turns supercomputing resources into a nationwide, sometimes even worldwide, productive environment by providing integration, usability, efficiency, sharing, and collaboration capabilities. An HPCE is also called an e-Science environment, especially in Europe. It is synonymous with the term 'cyberinfrastructure' used by the US National Science Foundation (NSF). Another name often used is a computational grid, or computing grid.

An example of a worldwide HPCE is the Worldwide LHC Computing Grid (WLCG) in high-energy physics. Its mission is 'to provide global computing resources to store, distribute and analyse the ∼30 Petabytes of data annually generated by the Large Hadron Collider (LHC) at CERN' [1,2]. It integrates resources from over 170 supercomputing centers in 40 countries, and handles over 1 million computational jobs per day for 1700 scientist users. The WLCG played an important role in the discovery of the Higgs boson (Nobel Prize in 2013).

There are many examples of nationwide HPCEs, such as the Extreme Science and Engineering Discovery Environment (XSEDE) in the USA [3,4], European Grid Infrastructure (EGI) [5], Japanese Computational Grid Research Project NAREGI [6], and China National Grid (CNGrid) [7,8].

China initiated activities to build a nationwide HPCE in 1995. Over the past 20 years, despite numerous difficulties, a set of coherent nationwide efforts persisted to enable the growth of the supercomputing field in China. Coherence in this instance means that these efforts were more or less aimed at achieving a similar goal. Drawing from these 20-year

experiments, this review presents important lessons related to the research agenda, technology choice, and science policy. These lessons are organized into three observations and two open problems:

(i) Observation 1 (research agenda): HPCE helps to grow China's supercomputing ecosystem by orders of magnitudes. A persistent research agenda for a nationwide HPCE, instead of ad hoc and isolated projects to develop supercomputers and their applications, is crucial for building a healthy supercomputing ecosystem. We present concrete evidence of this observation.

(ii) Observation 2 (technology): Loose coupling accommodates diversity. In scientific computing, there is diversity in science drivers, technology advances, management models, and user preferences. This diversity is inevitable, and an HPCE has to cope with it. An important technology design decision is that the HPCE technology stack should be loosely coupled to provide flexibility.

(iii) Observation 3 (policy): International cooperation is essential, even for a nationwide HPCE such as CNGrid.

(iv) Open Problem 1 (policy): What is a feasible institution to enable a sustainable HPCE?

(v) Open Problem 2 (technology): The worldwide scientific community benefits from a global Internet and global Web, but why is a global HPCE not widespread yet?

HPC will become increasingly more important in the coming decades. We offer perspectives for HPCE for the next 20 years (2015–2035), focusing on three questions: Will there be zettaflops system capability? Can we achieve 10 trillion computational operations per joule (10 TOPJ) energy efficiency? Can we build, by 2035, a seamless environment for human-cyber-physical ternary computing?

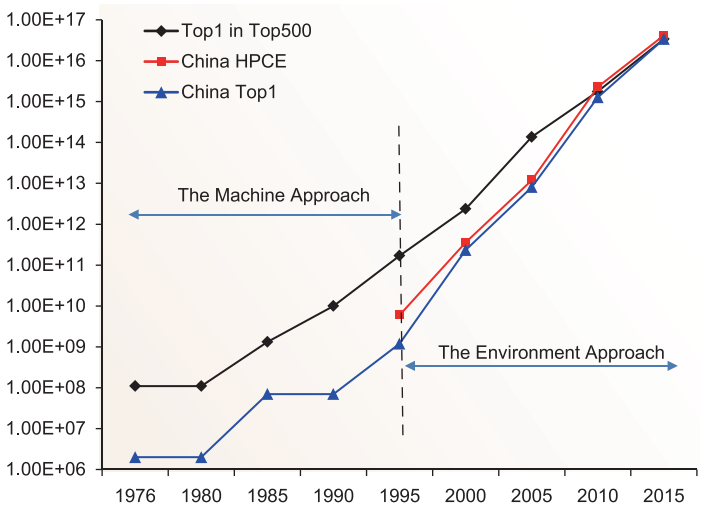

Figure 1. Linpack performance (flops) of Top1 in Top500, China HPCE, and China Top1.

HPCE HELPS TO GROW AN HPC ECOSYSTEM

Two approaches for supercomputing development

When setting the country's supercomputing research agenda, China followed two approaches in the 39 years between 1976 and 2015. (The year 1976 was significant in the world because Cray-1 was announced, which heralded the modern supercomputing era. It was significant in China because the Cultural Revolution ended in 1976, and China started to reemphasize science and technology development.) We call them the machine approach and the environment approach. The machine approach was the main practice in China before 1995. It was ad hoc to some extent, featuring isolated projects for building supercomputers and their applications. This machine approach was flexible but lacked a coherent long-term direction for the nation's supercomputing field. After 19 years, China's supercomputing field experienced significant growth, but continued to lag behind the world level (see Fig. 1 and Table 1). For instance, in 1995, the peak speed of a single world-level supercomputer had already exceeded 235 gigaflops [9]. More significantly, the Information Wide Area Year (I-WAY) experiment was demonstrated at the International Conference on Supercomputing in 1995, and provided a wide-area visual supercomputing environment (SCE) by interconnecting 17 sites [10], which heralded grid computing [11]. By contrast, there were only five disconnected public supercomputer centers in China nationwide—not counting private supercomputer centers within companies—and their peak speed totaled approximately 22 gigaflops. These systems were mostly running C/Fortran programs with little parallelism. Only a few scientific papers using computational results on these supercomputers were published in peer-reviewed journals and presented at international conferences. There were many external reasons for this lag, such as a severe lack of funding and human resources. However, the ad hoc, supercomputer-centric, machine approach showed disadvantages.

Since 1995, a more systematic approach has been gradually adopted in both China and the world. The research agenda has become to build, upgrade, and maintain a nationwide HPCE, wherein developing

Table 1. Growth trends of top supercomputers in China and the world before and after 1995.

	1976 to 1995		1995 to 2015	
Flops growth	Speed increase	Annual growth rate	Speed increase	Annual growth rate
China	600 times	40%	28 million times	136%
World	1550 times	47%	0.2 million times	84%

supercomputers and applications has become part of the research objectives. An HPCE also needs to connect and integrate supercomputer centers, scientific data, and middleware, and to offer services to a growing number of users.

Figure 1 and Table 1 show that after 1995, HPC development in China and the world both significantly accelerated. Although there are many contributing factors, we argue that, at least for China, an HPCE helps to grow China's supercomputing ecosystem and the environment approach has advantages. Measured by speed (floating-point operations per second, or flops) on the Linpack benchmark [9], the annual growth rate was 40% in China and 47% worldwide before 1995. After 1995, the annual growth rate was 136% in China and 84% worldwide.

HPCE helps to form a core community: CNGrid

More significant than the growth in raw computing power, Table 2 demonstrates that China's HPC users and scientific outcomes also experienced an increase of two orders of magnitude in the 20 years from 1995 to 2015.

The main reason that the environment approach is more effective than the machine approach is that the former enables the formation of a nationwide supercomputing community, and consequently, a growing supercomputing ecosystem. This community has two circles. At its core is CNGrid, funded by China's Ministry of Science and Technology, with matching funding from the Chinese Academy of Sciences (CAS), Ministry of Education, National Natural Science Foundation of China, and local governments. The outer circle, called CNGrid+, is a larger community that consists of SCEs within companies—both private and state-owned—universities, research institutes, and regional supercomputing centers.

Community forming is very important for the development of the national supercomputing field. Community building takes time, and it should not exclude innovation, dissident voices, and diversity, which are important for science and technology development. Nevertheless, the advantages of forming a community outweigh its disadvantages. We list three main benefits: a common direction, platform, and school, with evidence provided in Table 2.

A common direction: A community eases consensus building, especially to establish a long-term direction. By contrast, lacking a community could result in a set of conflicting Brownian movements. The community's efforts could accumulate into an accessible articulation to gain the support of the public and decision makers, resulting in executable national policies. An example is China's 'National Planning Framework for Mid- and Long-Term Science and Technology Development (2006–2020)', in which petaflops supercomputing was chosen as one of the 62 national priority directions [12]. This choice was made in 2005 after 2 years' intensive deliberation and consultation among thousands of scientists from all fields of science and technology.

Table 2. HPCE evolution in China, 1995–2015.

Attribute	1995	2000	2005	2010	2015
HPCE mode	Isolated	Interconnected	Grid services	Grid services	GS+domains
No. of sites	5	7	8	14	17
OS	Self-made	AIX, Linux	Linux	Linux	Linux
Middleware	N/A	NHPCE	CNGrid GOS	CNGrid GOS	CNGrid SCE
Total speed	0.01 Tflops	0.6 Tflops	18 Tflops	3.4 PFlops	62.6 PFlops
Total disk	0.08 TB	5.4 TB	200 TB	17.6 PB	34.6 PB
No. of apps	100	Hundreds	10 domains	450	500
No. of users	Dozens	Hundreds	Hundreds	Thousands	Thousands
Publications	A few	Dozens	Dozens	Hundreds	Hundreds
PhD awardees	A few	15	30	45	50
Funding/year	12 M yuan	20 M yuan	90 M yuan	440 M yuan	770 M yuan

This was possible because, at least partially, China's HPCE helped to create an HPC community in the decade from 1995 to 2005, and the community voiced its consensus. This historical decision provided a national research direction and steady funding for the HPC field's development in China for 2005–2015.

A common platform: A community provides a forum for scientist users, software developers, and hardware builders to learn from one another, and to codesign the systems and HPCE. For instance, users' requirements have heavily influenced the choice of supercomputer architectures, which in the past 15 years have converged to Linux clusters. This evolution to open systems fundamentally transformed China's supercomputing field, which before 1995, were mostly monopolized by proprietary systems sold by giant companies.

A common school: A community over an HPCE provides a hands-on platform to train young people. As shown in Table 2, the number of power users and HPC-related PhD awardees has steadily increased.

Table 2 shows in detail the growth of China's core HPCE and the CNGrid community over the past 20 years. The mode of HPCE evolved from nonexistence (ad hoc, isolated supercomputer centers) to interconnected but not managed HPC centers to a nationwide environment providing grid services with a single-system image to general-purpose grid services with support for multiple science domains. The number of HPC centers, also called HPCE sites, increased from 5 in 1995 to 17 in 2015. Supercomputer architecture converged to Linux clusters. The middleware for managing the HPCE evolved from nonexistence to rudimentary National High-Performance Computing Environment software to a Web services-based CNGrid GOS [8,13] to a simplified, more lightweight SCE today. The total computing speed and total storage capacity increased 5 million and 0.4 million times, respectively. The number of applications grew from around 100 small Fortran and C programs to thousands of Fortran/C/Java/Python programs, including hundreds of large programs. In addition to user-developed software, applications have included both commercial and open source software packages. Application software for over 10 application domains has been developed and deployed, ranging from a digital observatory for astronomy, gene sequencing, and climate computing to automobile simulation. The annual number of published peer-reviewed scientific papers supported by HPCE grew from only fewer than 10 in 1995 to hundreds in 2015.

LOOSE COUPLING ACCOMMODATES DIVERSITY

Technology must provide for usage diversity

Over the past 20 years, HPC users in China practiced multiple, different ways of using supercomputing resources, and this trend is continuing. This diversity seems inevitable, and technology has to cope with it. In fact, Richard Karp recently highlighted that this diversity may be fundamental from a scientist user's viewpoint. He summarized four generations of the relationship between computing and sciences [14,15], shown in Table 3. An HPCE needs to support all generations.

From a technology perspective, among the 20-year experiments and experiences of developing and using HPCE in China, we can identify four ways of performing supercomputing, and thus, four types of technology needs, as shown in Fig. 2. They are formed along two dimensions. The horizontal dimension regards whether users' applications are executed on a single site or multiple sites, where a 'site' is another name for a supercomputing center. The vertical dimension differentiates control, that is, whether the system is owned by and, thus, managed by one institution (centralized) or multiple institutions (decentralized). In more technical terms, a decentralized system has multiple administrative domains and a centralized system has one administrative domain.

The most familiar type is a traditional supercomputer center, which is a centralized, single-site system. Users apply for an account from the site administrator to use resources, which include the user's account, home directory, job queues, work directory (scratch space), software, data, and various resource quotas. Even today, many HPC users in China prefer this usage mode.

Table 3. Richard Karp's four phases of relationship between computing and sciences.

Phase	Name	Main characteristics (computing is used for)
I	Numerical analysis	Solving equations that model physical phenomena
II	Computational science	Simulation and visualization of the physical world
III	e-Science	Managing massive experimental data and collaboration via the Internet
IV	Computational lens	Computing as a universal way of thinking

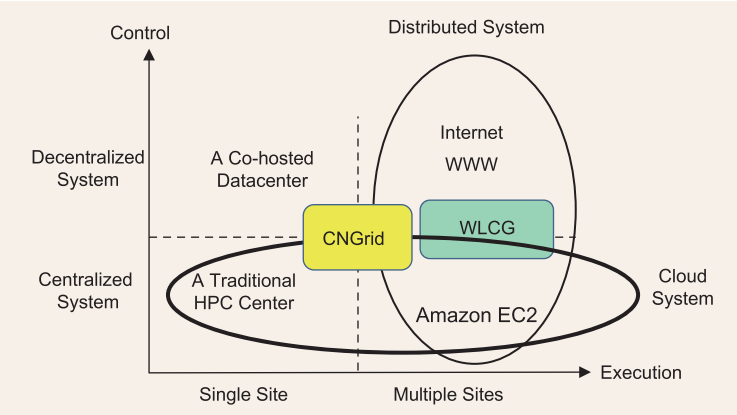

Figure 2. Four types of systems for HPC.

Decentralized single-site systems are mostly used in industry. A typical example is a cohosted datacenter, where the resources are allocated to different institutions with long-term contracts. The institutions manage and operate their own regions of resources.

The most popular centralized multisite system in industry is the cloud computing system, such as the Amazon EC2 cloud [16]. Such a system is owned and managed by a single institution, such as Amazon. Users can rent resources for computing, such as a cluster of virtual machine instances and S3 storage space. An application in such a centralized HPC system may execute on multiple sites or within a single site. Such HPC clouds have become increasingly popular in recent years because they usually have a viable business model and are easier to use and manage than federating resources from multiple institutions. High-speed interconnects, such as InfiniBand, and accelerators, such as GPUs, are added to HPC clouds to provide higher performance.

The most popular decentralized multisite systems are the Internet and World Wide Web (WWW). The WLCG is a multisite system that consists of over 170 supercomputing centers in 40 countries. However, the WLCG has both decentralized and centralized features. It is decentralized in the sense that its supercomputing centers are owned and managed by more than 100 institutions, which volunteer to contribute their resources for the common research agenda in high-energy physics. It is centralized in the sense that these resources, once contributed, are controlled and administrated in a common way, with a single-system image. For instance, the 1700 scientist users have the same global user account scheme, computational jobs are scheduled centrally, and petabytes of data are managed in one data space. A cross-site middleware platform supported by the European Middleware Initiative provides single-system image functionality and central management [17].

The Supercomputing Center of the Chinese Academy of Sciences (SCCAS) has developed a RESTful web interface called SCEAPI to enable access to various resources such as computing queues, applications, and data. SCEAPI allows scientists to securely connect to non-WLCG computers to analyze data for ATLAS experiments at CERN without a complicated WLCG middleware setup.

By a common understanding, a nationwide HPCE needs to interconnect and manage users, data, and programs hosted in multiple supercomputer centers that are geographically distributed. However, many users still prefer the single-site usage modes. Thus, the HPCE in China is designed to enable all four modes in Fig. 2. In Table 4, we list the main generations of progressively more sophisticated technology. Note that older technology is not necessarily worse technology for a user. In addition, these four generations are not inclusive, that is, later technology does not necessarily include earlier technology, and they reflect different user preferences. In fact, all four generations are needed and in use today.

HPCE 0 is characterized by isolated sites. Some sites are not even on the Internet, or blocked by firewalls. The HPCE 0.1 mode only allows users to enter a supercomputer site to conduct on-site computing. The HPCE 0.2 mode allows users to use a client machine to access resources remotely at a supercomputer site through protocols such as SSH and SFTP. A user may access multiple sites via the same client device, but the sites themselves are not interconnected. Some sites even force users to log in through a dynamic passcode counter for security reasons.

HPCE 1 features interconnected sites. The I-WAY project in 1995 was probably the first interconnected HPC environment, or HPCE 1.1. If there are n sites, a user needs to send out n resource request applications for n user accounts, one for each

Table 4. Evolution of four generations of the HPCE.

Generation	Characteristics
HPCE 0.0	**Isolated supercomputer sites**
HPCE 0.1	On-site computing
HPCE 0.2	Remote computing
HPCE 1.0	**Interconnected environment (interconnected sites)**
HPCE 1.1	Multiple resource requests, no platform support
HPCE 1.2	One resource request, multiple accounts, platform support
HPCE 2.0	**Federated environment (federated sites)**
HPCE 2.1	Targeted for one application domain
HPCE 2.2	Supporting multiple application domains
HPCE 3.0	**Ternary computing**, extending HPCE to the physical world and human society

site. Then the user can develop multisite resource sharing and collaboration capabilities at the application level, but with no platform support. In HPCE 1.2, some platform support is provided through middleware for cross-site resource sharing and collaboration. Often-used middleware is GridFTP [18] for intersite data transfer. Furthermore, a user only needs to send out one resource request application to the environment, which covers the desired resources from all sites.

HPCE 2 features federated sites. Federation in this instance means that the sites each set aside some resources to be managed by the environment. An important feature is that a global, environment-level user account system is available, which enables a user to use resources in all sites. Some type of single-system image is also available. Although the sites still belong to different institutions or organizations, a virtual organization is set up to control and manage the environment resources. In HPCE 2.1, the environment is designed mainly for one application domain. A good example is the WLCG for high-energy physics. In HPCE 2.2, the environment needs to support multiple application domains. As an example, the CNGrid environment today is designed to support three communities, in addition to general-purpose scientific and engineering computing. The three application domains are drug discovery, movie rendering, and industrial simulation.

With HPCE 3.0, the cyberinfrastructure is no longer just for cyberspace. It is extended to human society and the physical world for human-cyber-physical ternary computing. This is a future trend.

Loosely coupled architecture demonstrates flexibility

An HPCE should support general-purpose scientific and engineering computing, in addition to domain-specific systems, over multiple sites belonging to different organizations. This is a challenging task. Over the 20-year practice of China's HPCE development, four technical guidelines have emerged. The central concept is loosely coupled system architecture, that is, the HPCE system architecture needs to allow the corunning of multiple technical stacks.

Guideline 1: Users are different, sites are different, and communities are different. An HPCE needs to recognize and respect these differences. It is difficult or even impractical to force all stakeholders them to follow a single management policy or single usage mode.

Guideline 2: The HPCE needs to provide site-level, environment command line-level, and Web portal-level interfaces. Many users only want to run applications and are not concerned about optimized performance. They are often satisfied with using an environment-level Web portal interface. Power users are concerned about performance and efficiency, and are willing to take time to assess system details. They need command line interfaces at both the environment and site levels.

Guideline 3: HPCE middleware is there to help, not impede. The HPCE system architecture must allow users and sites to bypass HPCE middleware when developing and executing applications. The HPCE can provide information and knowledge, not just management.

Guideline 4: The HPCE is comprised of two circles: at the core is the CNGrid itself, and the outer circle is the CNGrid+. The CNGrid is managed by the same middleware, but allows users to also access a site directly. The CNGrid+ has additional users and sites that can be managed differently, but share information and knowledge with CNGrid. They can even leave CNGrid later to form a subcommunity of a nationwide HPCE.

By 2015, the CNGrid HPCE had 17 sites and over 3000 users, of which 40% accessed resources through CNGrid SCE middleware. The CNGrid+ spun off many private HPC environments. They began as members of CNGrid, developed their networked computing knowledge base, and then left CNGrid to form their own HPC environments. For instance, Beijing Genomic Institute (BGI) was a member of CNGrid in 2000–2008. BGI utilized resources in CNGrid to quickly become a genome 'sequencing superpower' [19], and to develop its rice genomics information service system, BIG-RIS [20]. BGI is now one of the leading genome sequencing institutions in the world. Another example is China Aviation Corp (AVIC), which developed its HPCE prototype [21] as a project in CNGrid, but later ran the developed HPCE production system as an intracompany computational grid. These HPCEs became private mainly because their computations involved proprietary software and data, or private information.

Loosely coupled architecture also enables innovations, sometimes extending beyond the boundary of traditional HPC, to enter emerging areas such as cloud computing and big data computing. For instance, built on experiences in HPCE, Tsinghua University's HPC site now offers all students and faculty members a cloud account, with storage space and computing power. The CNGrid GOS team developed open source software such as CCIndex [22], RCFile [23], and DataMPI [24,25], for big data computing.

Although loosely coupled architecture allows users to bypass HPCE middleware, implementing

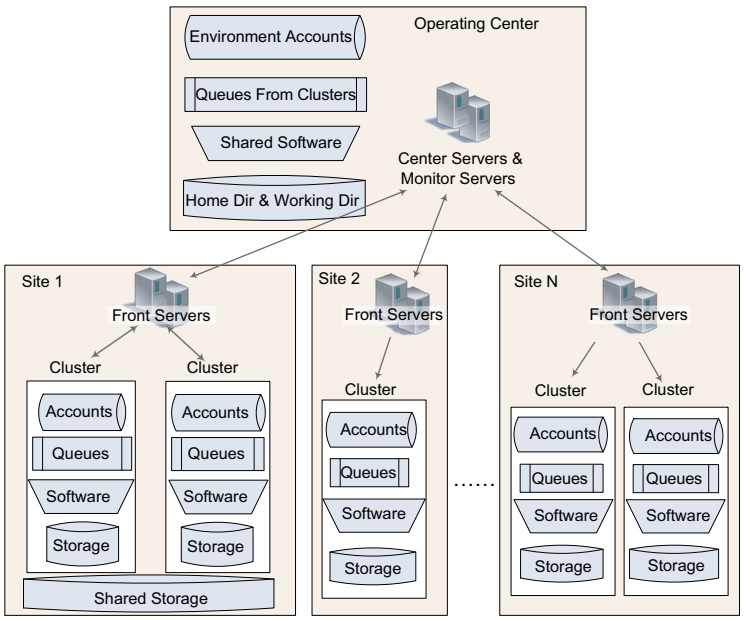

Figure 3. The CNGrid architecture as seen by an environment user.

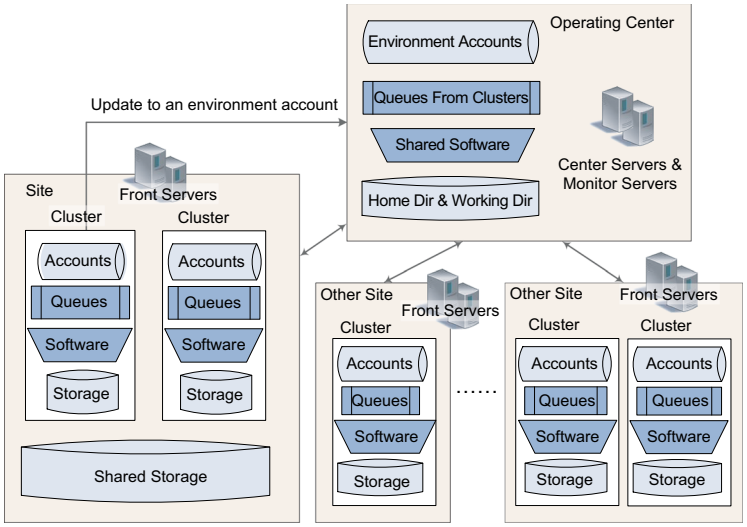

Figure 4. The CNGrid architecture as seen by a site user.

middleware has advantages. Three types of benefits for scientific users are observed from CNGrid operations and research: environment abstraction, site augmentation, and cross-site resource sharing, as discussed below with application examples.

Figure 3 illustrates the 'environment abstraction' benefit, that is, an HPCE user can see and use an abstract collection of resources from all sites through a familiar interface, although actual executions of computations occur in the sites.

The CNGrid HPCE offers each user two types of accounts: an environment account and a set of site accounts, one for each granted site. Through the environment account, the user sees a uniform user interface that hides site heterogeneity and offers site location transparency. An HPCE user can use either the Web portal or command line method to use resources, with the same environment account name and password.

The HPCE user sees an abstract collection of permitted job queues; software, encapsulated as services; and storage spaces in all participating clusters of the sites. A special storage space is set aside to hold all users' environment accounts and home directories. HPCE users execute the same operating steps or commands as they would if they used a cluster, but the resources used can be on any site. For example, the HPCE command line 'bsub -n 128 -q normal vasp' means submitting a job to the queue 'normal' utilizing 128 cores and VASP software, even though the executing clusters may have different job management systems or VASP environments and commands.

When any cluster in HPCE is down, users can transparently use other clusters without needing to apply for another user account or read new cluster manuals. In September 2010, one cluster was down for almost one month because of water cooling system damage. As a result of an HPCE, users were able to use other clusters, and most of their jobs were executed without interruption.

Figure 4 illustrates the 'site augmentation' benefit. More precisely, the benefit is 'site user augmentation', that is, a traditional HPC site user can be augmented by HPCE middleware by bringing in extended job queue and software resources. The site user sees an augmented site, with more job queues and software than the existing site originally offered.

The important factor is that a site user can be elevated to an HPCE user by the HPCE administrator. Then the user can remain as a user of the existing site, supported by the familiar site team, but use job queues and software in other sites. A recent example occurred in 2014, when a user of the SCCAS site constantly complained that his jobs suffered from a very long pending time. This scientist has been studying the effect of aluminum on the protein structure using quantum computation on Gromacs software. The SCCAS site elevated him by providing an HPCE account. He could then submit Gromacs jobs to other sites. Now, he is still one of most active users at the SCCAS site.

The third benefit is cross-site resource sharing. An example is RamGrid, a service-oriented wide-area memory sharing scheme. This technique can utilize memories in multiple sites of an HPCE to form a giant memory so that application data can reside in memory without going to disk. All this can be performed in a distributed, heterogeneous, and dynamic environment as a significant

Table 5. A partial list of open source software contributed by China HPCE.

Software name	Software description
DCFS [32]	Dawning Cluster File System: a file system for Linux clusters (http://www.ncic.ac.cn/dcfs/)
OpenBLAS [33,34]	An optimized BLAS (Basic Linear Algebra Subprograms) library (http://www.openblas.net/)
OpenCVCL [35]	An optimized OpenCV kernel for GPU, merged into OpenCV (http://opencv.org/)
yaSpMV [36]	An optimized sparse matrix-vector multiplication routine for GPU (https://code.google.com/p/yaspmv/)
Hadoop+ [37]	A heterogeneous MapReduce framework (https://github.com/ict-carch/hadoop-plus)
Loongcc [38]	A C compiler for Loongson series MIPS-compatible CPUs (http://svn.open64.net)
FunctionFlow [39]	A C++ parallel programming library with a simple interface (https://github.com/AthrunArthur/functionflow)
LiveRender [40]	A cloud gaming system (https://github.com/llfjfz/LiveRender)
NightWatch [41]	A malloc system with extensions for CPU cache management (https://github.com/grtoverflow/PC-Malloc)
Mammoth [42]	A memory-centric MapReduce system (https://github.com/mammothcm/mammoth; https://issues.apache.org/jira/browse/MAPREDUCE-5605)
Frog [43]	An asynchronous graph processing system for GPU (https://github.com/AndrewStallman/Frog)
Giraffe [44]	A scalable distributed coordinator (https://github.com/haohonglin/Giraffe)
TripleBit [45]	A graph database supporting SPARQL and transactions (http://grid.hust.edu.cn/triplebit)
PathGraph [46]	A parallel graph computing system for one compute node (http://grid.hust.edu.cn/triplebit)
CCIndex [22]	An complementary clustering index system for multiattribute range queries on distributed ordered tables (https://github.com/ICT-Ope/CCIndex_HBase_0.90.0)
RCFile [23]	A row-column file placement scheme for a data warehouse (https://en.wikipedia.org/wiki/RCFile)
DataMPI [24,25]	A key-value communication library extending MPI for big data (http://DataMPI.org)

improvement over disk I/O performance. Experiments on an eight-site HPCE demonstrated that the RamGrid technique could provide Internet memory services [26].

INTERNATIONAL COOPERATION IS ESSENTIAL

Although CNGrid is a nationwide HPCE environment mainly for China, we found that international cooperation is essential in the following three aspects:

First, China's HPC field is part of the international HPC community. Interactions among scientists and engineers are very important for the healthy development of the worldwide HPC field, especially when the world community joins forces, going in the same direction.

Second, China's HPCE development has benefitted from international cooperation projects. We participated in the WLCG [1,2], the European Commission's XtreemOS project [27], and the UK's e-Science projects [28], and collaborated with US cyberinfrastructure projects such as Globus [29] and TeraGrid [3]. We learned the importance of supporting application domains from the NanoHub project [30].

Third, China's HPCE development contributes to the worldwide HPC community. Hundreds of graduate students educated in the process of developing China's HPCE now work in the HPC field worldwide. Scientific data produced by China's HPCE are used by colleagues worldwide. Early examples are rice genome data [31] and rice information services [20], which have been widely used and cited. China's HPCE has also contributed open source software to the worldwide community, a partial list of which is shown in Table 5.

OPEN PROBLEMS

What is a sustainable development model for an HPCE?

We summarized in Fig. 1 and Tables 1 and 2 that the environment approach in 1995–2015 helped to grow China's supercomputing field. China's sustainable development and scientific advance will benefit if the growth trends continue in 2015–2035. For instance, there is still a severe lack of HPC human resources in China's IT industry today. People who understand HPC are in high demand, whether the knowledge is in HPC systems architecture, computational models, parallel algorithms, performance optimization, or coding for MPI, OpenMP, or GPU application programs. Can we increase China's HPC users by two orders of magnitude as we did in the 20 years from 1995 to 2015? That is, can we have

100 000 HPCE users by the year 2035? This is not an unrealistic goal because many students in China's science and engineering schools are already using GPUs in their personal computers and local servers to speed up computation.

There is no evidence that the growth trend will continue for the next 20 years. In particular, the environment approach is not guaranteed to be adopted in the future. In fact, we face the danger that future development of China's supercomputing field may degenerate into the isolated sites (HPCE 0.0) situation of 1995, back to the machine approach.

The problem is that we have not found a sustainable model, especially a sustainable institution model for the long-term development of the supercomputing field in China. By contrast, the space program in China has such a long-term, sustainable development model.

There is related work and practice worldwide, especially in Europe, the USA, and Japan. The USA seems to have a relatively comprehensive model. The federal government plays a leadership role, with participation from industry and local governments. Approximately every 10 years, the US federal government issues a binding strategy document. The most recent White House executive order was announced in July 2015 and set up a coordinated 20-year strategy called the National Strategic Computing Initiative (NSCI). At the agency level, an interesting example is how NSF supports HPCE, which is called cyberinfrastructure by NSF. Cyberinfrastructures are 'research environments that support advanced data acquisition, data storage, data management, data integration, data mining, data visualization and other computing and information processing services distributed over the Internet beyond the scope of a single institution' [47]. Before 2004, NSF supported cyberinfrastructures by project. Between 2005 and 2012, NSF established a temporary Office of Cyberinfrastructure to perform the role. After 2013, this office became a permanent Division of Cyberinfrastructure.

China can learn from all the above initiatives, but has to find its own model to suit its situation. Two central questions are as follows: How should the central government play a leadership role in setting up a long-term national strategy? What institution should be established to conduct the strategy?

Why not a global HPCE like the Internet and WWW?

Both the Internet and WWW were mainly created and implemented by the public research community, and they are both 'beyond the scope of a single institution'. They reached global, massive public use in fewer than 20 years. Today, we have a single, global Internet and a single, global Web used by billions of ordinary citizens.

If we set I-WAY [10] as the world's first HPCE, 20 years have passed since its invention. Thus, why is HPCE not yet widespread? Why is there not a single, global HPCE, or cyberinfrastructure? What makes HPCE different from the Internet and Web? What is missing?

We heard many anecdotal 'reasons'. However, research is greatly needed to gain a systematic and scientific understanding. We briefly discuss three common complaints below.

An HPCE is supposed to provide resource sharing and collaboration capabilities among multiple sites. However, resources in most HPC centers in China and worldwide are already heavily booked, leaving no resources available for sharing, for instance, the utilization of the CAS Supercomputing Center's machines is above 75%. Another common complaint is that the current HPCE technology stack is too heavy and complex and difficult to learn, use, and operate. This was made worse by the Web services movement that started in the early 2000s, when big companies arrived and pushed Web services standards, such as the simplest example called WS-I [48], and this had an overall negative effect on HPCE development. More recent HPCE software has shed much of the complexity and has opted for the simpler REST architectural style [49]. Yet, another complaint is that the current HPCE lacks cloud computing's elasticity and agility, such that users can almost instantly expand their computing resources from 10 cores to 10 000 cores in a few minutes or even seconds. This is made possible by the fact that a cloud is centralized, owned, and operated by a single institution. An HPCE is a federation of multiple sites 'beyond the scope of a single institution'. [47] It is not as easy to realize such elasticity.

OUTLOOK AND PERSPECTIVES

What will happen in the next 20 years? What will we see when looking back in 2035? We offer perspectives on three important issues: Will we see zettaflops system capability? Could this zettaflops capability be provided with 10 TOPJ (trillion operations per joule) energy efficiency? Can we build, by 2035, a seamless environment for human-cyber-physical ternary computing?

Zettaflops computing capability by 2035

In China, the installation of 100-petaflops systems by 2016 is scheduled for the CNGrid HPCE. President Obama of the USA recently issued an executive order to create an NSCI, which will 'create systems that can apply exaflops of computing power to exabytes of data', probably around 2023. What about 2035? Will the world see zettaflops computing capability? Should China set zettaflops supercomputing on zettabytes of data as a national research priority?

This is a serious question, not idle speculation. In 2012, the CAS launched a New-Generation Information and Communication Technology strategic priority project, where a core component was cloud-sea computing systems for zettabytes of data [50]. The Chinese Academy of Engineering recently started a community consultation process to identify potential national priority research directions of engineering technology by 2035 [51]. One of the six candidates in the information technology area is zettaflops supercomputing. However, Thomas Sterling, a coauthor of the influential book *Enabling Technologies for PetaFLOPS Computing*, conjectured that 'we will never reach zettaflops' [52].

We believe that zettaflops supercomputing on zettabytes of data should be a long-term national research priority because there are scientific and societal needs, especially in intelligent computing with multiscale, high-dimensional data in a human-cyber-physical universe. However, the capability could be best provided by an HPCE, not a single supercomputer system. The systems architecture, programming model, and application frameworks could be quite different from today's SCEs. Research should start now.

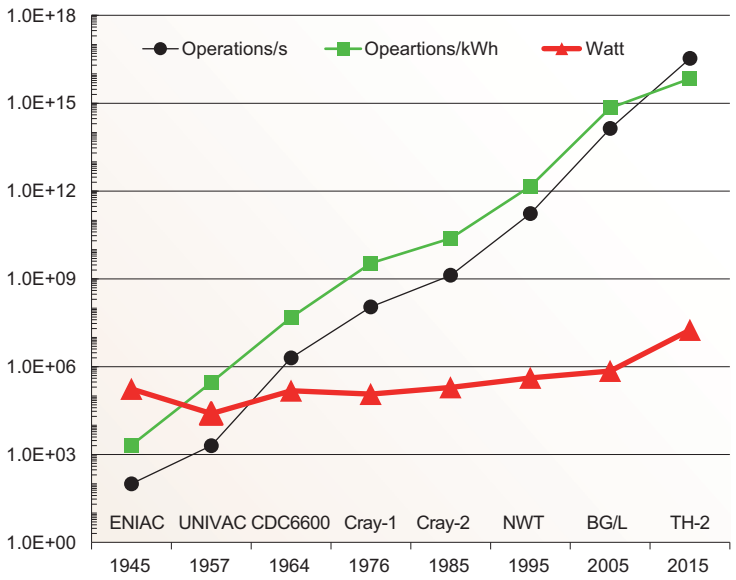

Figure 5. The trends of speed, energy efficiency, and power consumption of the world's fastest computers over the past 70 years (1945–2015).

10 000 times improvement of energy efficiency

The modern HPC era started in 1976 with the introduction of the Cray-1 supercomputer. When one looks back carefully at the history of the past four decades, two major phases may be observed in HPC systems development, each lasting roughly 20 years. The first is called the performance-first phase, lasting roughly from 1976 (Cray-1) to 1994. The most important priority of HPC systems development in this phase was performance, or flops speed. Factors such as systems cost and application scope were of secondary consideration. Energy consumption was considered insignificant.

The second phase is called the scalability-first phase, spanning from 1994 (IBM SP-2) to the present. The most important priority of HPC systems development in this phase is scalability, including market scalability and systems scalability. The worldwide overall HPC market revenue grew from US$2 billion in 1990 to approximately US$25 billion in 2015, according to the market research firm IDC. The application scope significantly expanded. An important feature of systems architecture to support application market expansion is the convergence to clusters, so that a big system can scale down to smaller systems. This increased product volume thus improves the performance/cost ratio. For high-end applications, a system can scale up and scale out to provide more parallelism. The number of cores per system increased from 140 in 1995 to 31 million in 2015.

Now we are entering a third phase: the efficiency-first phase. In the next 20 years, supercomputing systems research needs to increase energy efficiency by 10 000 times, in addition to continuing performance and scalability advances. The reason is illustrated in Fig. 5, which lists the speed (operations executed per second), energy efficiency (operations executed per kilowatt hour), and system power consumption (watt) of the world's fastest computers of the past 70 years. We can observe a disturbing recent trend. For 60 years, the energy efficiency improved at the same rate as the speed. However, in the past 10 years, this has changed, and energy efficiency improvement now lags behind speed improvement.

The research community should reverse this trend by setting a bold goal of achieving energy efficiency of 10 tera operations per joule (10 TOPJ), or 10 tera operations per second per watt (10 TOPS/W) by 2035. Today, CPUs can deliver

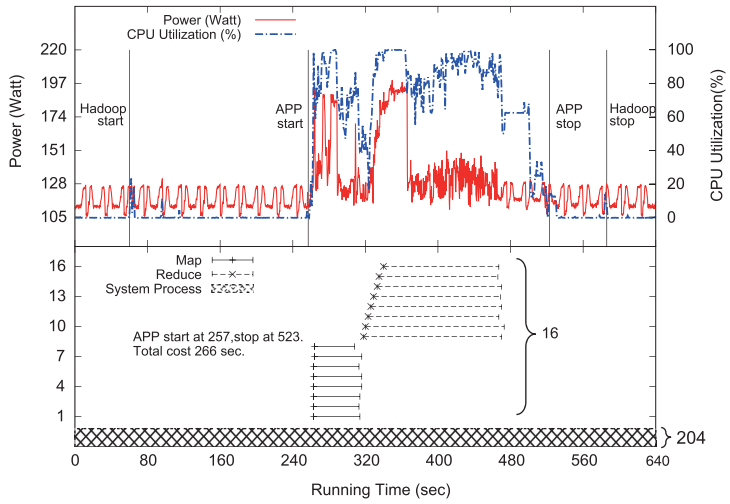

Figure 6. The traces of system power, CPU utilization, and task processes when sorting 4 GB of data on one cluster node.

1 GOPS/W and GPUs can perform 10 GOPS/W. Thus, we need to increase energy efficiency by 1000 to 10 000 times. A promising direction is aggressive specialization, which means dynamically changing the hardware to match application needs. In the area of deep learning applications, there is research that demonstrates the feasibility of over 1 TOPJ at both the microchip level and subsystem level [54,55]. It is a significant technology challenge to achieve 10 TOPJ at the whole system level, while maintaining programmability and usability.

When we consider the need to handle big data, the efficiency issue becomes more urgent. Scientific data computing systems often store data in file systems (unstructured), while data computing systems within Internet services also organize data in a more structured way, such as in relational databases, or a semi-structured way, such as in NoSQL databases or key-value stores. Big data computing systems today consider four nonfunctional metrics and have made significant progress. The first is fault tolerance, that is, how well are faults handled. Now, big data computing systems provide good-enough availability for managing 100 PB to 1 EB of data. The second is usability, that is, how easy is it to write a scalable application. Big data computing application frameworks such as Hadoop enable a user to write a sort program using only a few dozens of lines of code, and the code does not need to change with the data size (1 GB, 1 TB, 1 PB, etc.) or system size (1 node, 100 nodes, 1000 nodes, etc.). The third is scalability, that is, how much data can be handled. Now over 1 EB of data can be managed. The fourth is speed. Now, 10 PB data can be sorted in hours.

However, the design and implementation of big data computing systems today have not paid sufficient attention to efficiency. The result is that current data computing systems have much worse efficiency than HPC systems. Let us consider two efficiency metrics. Speed efficiency is equal to sustained speed divided by the system's peak speed. Energy efficiency is equal to sustained speed per watt, which is equal to the number of payload operations executed per joule. Payload operations are the actual work operations of the application program, not including other operations executed by the system for management or other overheads. For instance, when solving a linear system of equations of size N using the Linpack program, the number of payload operations is equal to $2N^3/3 + O(N^2)$ double-precision floating-point operations [9].

Table 6. Speed efficiency and energy efficiency when running the Hadoop sort and Linpack benchmarks on one cluster node.

Benchmark	Speed efficiency	Energy efficiency
Hadoop sort	0.002%	0.0000155 GOPS/W
Linpack	94.5%	0.726 GOPS/W

Figure 6 shows traces of system power, CPU utilization, and task processes when sorting 4 GB of data on one cluster node. The node hardware contains two four-core E5620 CPUs, 16 GB memory, and 150 GB hard disk. The system software consists of Hadoop 0.20.2 over CentOS 5.3. The application software (APP) executed is the standard Hadoop sort benchmark.

Three surprising results can be observed: (i) there are only 16 payload processes but 204 system processes running; (ii) more than 50% of system power is consumed when no work is being performed (the CPU utilization is 0); and (iii) power consumption varies widely even when the CPU utilization is in the high region (80%–100%).

The most surprising result is listed in Table 6. On the same cluster node, the HPC Linpack benchmark achieved a speed efficiency of 94.5% and an energy efficiency of 0.726 GOPS/W. However, the Hadoop sort data computing benchmark's corresponding numbers are 47 000 times worse. There is an efficiency gap of five orders of magnitude between HPC and big data computing. In-memory data computing systems such as Spark can speed up Hadoop by about six to ten times. There are still thousands of times of improvement potential.

A seamless environment for human-cyber-physical ternary computing

The future HPC usage mode may change too. Scientists may access HPC resources not only from a desk-

Figure 7. A video monitoring system deployed in the core protection area of the Qinghai Lake nature reserve, networked to the CNGrid backend.

top or laptop, but from tablets, smartphones, or even sensor devices. The HPCE we see today is mostly within cyberspace. We may see a trend to extend the HPC environment to the physical world. In fact, today's CNGrid already has environmental science applications, where sensor devices are used together with the HPCE to collect and analyze data for bird migration patterns in the Qinghai Lake Nature Reserve (See Fig. 7) [56]. This is a rudimentary human-cyber-physical ternary computing scenario: sensors and wireless communication technology extend the HPCE infrastructure to the physical world, the backend supercomputer performs data analytics, and scientist users orchestrate and steer field research and backend computation tasks.

Can we have, by 2035, a seamless environment for human-cyber-physical ternary computing [57]? In such an environment, the scientist users, cyberspace, and physical world all become resources and research targets in a new type of HPCE. We can expect that some type of seamless intelligence [58] will become available for scientific research by 2035. Such a seamless environment would not only enable high-end scientific research, but also benefit scientific and engineering experiments for students in high schools and universities.

ACKNOWLEDGEMENTS

The first author would like to thank Drs Gordon Bell, Jonathan Koomey, Dag Spicer, and Ed Thelen for helping to provide data on the first three machines in Fig. 5. The unit for energy efficiency (operations per kilowatt hour) follows Dr Koomey's convention [53]. Data on Cray systems are from the Cray company website. Data on the last three systems are from Top500.org. The authors would like to thank Fan Liang of the Institute of Computing Technology, CAS for providing the raw experimental data. The authors would also like to thank Dr Xiaoning Wang of the CNGrid Operating Center for providing raw data related to CNGrid. This research is supported in part by the Hi-Tech Research and Development (863) Program of China (2013AA01A209), the Natural Science Foundation of China (61532016), the Strategic Priority Program of the CAS (XDA06010401) and the Guangdong Talents Program of China (201001D0104726115).

REFERENCES

1. *The Worldwide Large Hadron Collider Computation Grid*. http://wlcg.web.cern.ch/ (16 September 2015, date last accessed).
2. Geddes, N. The Large Hadron Collider and grid computing. *Phil Trans R Soc A* 2012; **370**: 965–77.
3. *The Extreme Science and Engineering Discovery Environment*. https://www.xsede.org/ (16 September 2015, date last accessed).
4. Towns, J, Cockerill, T and Dahan, M *et al*. XSEDE: accelerating scientific discovery. *Comput Sci Eng* 2014; **16**: 62–74.
5. *The European Grid Infrastructure*. http://EGI.eu/ (16 September 2015, date last accessed).
6. Matsuoka, S, Shimojo, S and Aoyagi, M *et al*. Japanese computational grid research project: NAREGI. *Proc IEEE* 2005; **93**: 522–33.
7. *China National Grid*. http://www.cngrid.org/ (16 September 2015, date last accessed).
8. Xie, X, Xiao, N and Xu, Z *et al*. CNGrid Software 2: service oriented approach to grid computing. In: *Proceedings of the IFIP International Conference on Network and Parallel Computing*. 2005, 14–21.
9. *The List of the 500 Most Powerful Computer Systems Worldwide*. http://top500.org/ (16 September 2015, date last accessed).
10. DeFanti, T, Foster, I and Papka, M *et al*. Overview of the I-WAY: wide area visual supercomputing. *Int J Supercomput Appl* 1996; **10**: 123–30.
11. Foster, I and Kesselman, C (eds) *The Grid 2: Blueprint for a New Computing Infrastructure*. San Francisco: Morgan Kaufmann Publishers, 2004.
12. *China's National Planning Framework for Mid- and Long-Term Science and Technology Development (2006–2020)*. http://www.most.gov.cn/kjgh/kjghzcq/ (16 September 2015, date last accessed).
13. Xu, Z, Li, W and Zha, L *et al*. Vega: a computer systems approach to grid computing, *J Grid Comput* 2004; **2**: 109–20.
14. Karp, RM. Understanding science through the computational lens. *J Comput Sci Technol* 2011; **26**: 569–77.
15. Xu, Z and Tu, D. Three new concepts of future computer science. *J Comput Sci Technol* 2011; **26**: 616–24.
16. Juve, G, Rynge, M and Deelman, E *et al*. Comparing FutureGrid, Amazon EC2, and Open Science Grid for scientific workflows. *Comput Sci Eng* 2013; **15**: 20–9.

17. *The European Middleware Initiative*. http://www.eu-emi.eu/ (16 September 2015, date last accessed).
18. Subramoni, H, Lai, P and Kettimuthu, R *et al*. High performance data transfer in grid environment using GridFTP over InfiniBand. In: *Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*. 2010, 557–64.
19. Normile, D. Beijing Genomics Institute: from standing start to sequencing superpower. *Science* 2002; **296**: 36–9.
20. Zhao, W, Wang, J and He, X *et al*. BGI-RIS: an integrated information resource and comparative analysis workbench for rice genomics. *Nucl Acids Res* 2004; **32**: D377–82.
21. Cui, D and Cheng, W. Advanced manufacturing grid supporting aviation industry extensive collaborative design and simulation. In: *Proceeding of the 26th International Congress of Aeronautical Sciences*. 2008, 1–6.
22. Zou, Y, Liu, J and Wang, S *et al*. *CCIndex: A Complemental Clustering Index on Distributed Ordered Tables for Multi-dimensional Range Queries*. IFIP NPC, 2010, 247–61.
23. He, Y, Lee, R and Huai, Y *et al*. *RCFile: A Fast and Space-Efficient Data Placement Structure in MapReduce-Based Warehouse Systems*. ICDE, 2011, 1199–208.
24. Lu, X, Liang, F and Wang, B *et al*. *DataMPI: Extending MPI to Hadoop-Like Big Data Computing*. IPDPS, 2014, 829–38.
25. Chao, L, Li, C and Liang, F *et al*. *Accelerating Apache Hive with MPI for Data Warehouse Systems*. ICDCS, 2015.
26. Chu, R, Xiao, N and Zhuang, Y *et al*. A distributed paging RAM grid system for wide-area memory sharing. In: *Proceedings of the 20th IEEE International Parallel and Distributed Processing Symposium*, 2006, 25–9.
27. Coppola, M, Jégou, Y and Matthews, B *et al*. Virtual organization support within a grid-wide operating system. *IEEE Internet Comput* 2008; **12**: 20–8.
28. Hey, T and Trefethen, A. The data deluge: an e-science perspective. In:Berman, F, Fox, G and Hey, AJG (eds). *Grid Computing: Making the Global Infrastructure a Reality*. Wiley, 2003.
29. Foster, I. Globus toolkit version 4: software for service-oriented systems. *J Comput Sci Technol* 2006; **21**: 513–20.
30. Klimeck, G, McLennan, M and Brophy, SP *et al*. nanoHUB.org: advancing education and research in nanotechnology. *IEEE Comput Sci Eng* 2008; **10**: 17–23.
31. Yu, J, Hu, SN and Wang, J *et al*. A draft sequence of the rice genome (Oryza sativa L. ssp. indica). *Science* 2002; **296**: 79–92.
32. Xiong, J, Wu, S and Meng, D *et al*. Design and performance of the dawning cluster file system. *IEEE International Conference on Cluster Computing* (Cluster2003).
33. Zhang, X, Wang, Q and Zhang, Y. Model-driven level 3 BLAS performance optimization on Loongson 3A processor. In: *Proceeding of the IEEE 18th International Conference on Parallel and Distributed Systems*. 2012, 684–91.
34. Wang, Q, Zhang, X and Zhang, Y *et al*. AUGEM: automatically generate high performance dense linear algebra kernels on x86 CPUs. In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. November 2013.
35. Jia, H, Zhang, Y and Long, G *et al*. GPURoofline: A model for guiding performance optimizations on GPUs. In: *Proceeding of the 18th International European Conference on Parallel and Distributed Computing*. 2012, 920–32.
36. Yan, S, Li, C and Zhang, Y *et al*. yaSpMV: yet another SpMV framework on GPUs. In: *Proceeding of the 19th ACM Symposium on Principles and Practice of Parallel Programming*. 2014, 107–18.
37. He, W, Cui, H and Lu, B *et al*. Hadoop+: Modeling and Evaluating the Heterogeneity for MapReduce Applications in Heterogeneous Clusters. In: *Proceedings of the 29th International Conference on Supercomputing*. ACM, 2015, 143–53.
38. Zhou, S, Liu, Y and Lu, F *et al*. Open64 on MIPS: porting and enhancing Open64 for Loongson II. Open64 Workshop at CGO, 2009.
39. Fan, X, Jin, H and Zhu, L *et al*. Function flow: making synchronization easier in task parallelism. In: *Proceedings of the 2012 International Workshop on Programming Models and Applications for Multicores and Manycores*. 2012, 74–82.
40. Lin, L, Liao, L and Tan, G *et al*. LiveRender: a cloud gaming system based on compressed graphics streaming. In: *Proceedings of the 22nd ACM International Conference on Multimedia*. 2014, 347–56.
41. Guo, R, Liao, X and Jin, H *et al*. NightWatch: integrating lightweight and transparent cache pollution control into dynamic memory allocation systems. In: *Proceedings of the 2015 USENIX Annual Technical Conference*, 2015, 307–18.
42. Shi, X, Chen, M and He, L *et al*. Mammoth: gearing Hadoop towards memory-intensive MapReduce applications. *IEEE T Parall Distr* 2015; **26**: 2300–15.
43. Shi, X, Liang, J and Di, S *et al*. Optimization of asynchronous graph processing on GPU with hybrid coloring model. *The 2015 ACM Symposium on Principles and Practice of Parallel Programming*.
44. Shi, X, Lin, H and Jin, H. *GIRAFFE: A Scalable Distributed Coordination Service for Large-scale Systems*. Madrid, Spain: IEEE Cluster, 2014.
45. Yuan, P, Liu, P and Wu, B *et al*. TripleBit: a fast and compact system for large scale RDF data. *PVLDB* 2013; **6**: 517–28.
46. Yuan, P, Zhang, W and Xie, *et al*. *Fast Iterative Graph Computation: A Path Centric Approach*. SC, 2014.
47. *Cyberinfrastructure Definition*. https://en.wikipedia.org/wiki/Cyberinfrastructure (16 September 2015, date last accessed).
48. *Web Services Standards*. http://www.ws-i.org/ (16 September 2015, date last accessed).
49. Fielding, RT and Taylor, RN. Principled design of the modern Web architecture. *ACM Trans Internet Technol* 2002; **2**: 115–50.
50. Xu, Z. Cloud-sea computing system: towards thousand-fold improvement in performance per watt for the coming zettabyte era. *J Comput Sci Technol* 2014; **29**: 177–81.
51. *China Engineering Technology 2035 Foresight Survey*. http://www.cae.cn/ (16 September 2015, date last accessed).
52. *Thomas Sterling: 'I Think We Will Never Reach Zettaflops'*. http://www.hpcwire.com/2012/05/07/thomas_sterling:_i_think_we_will_never_reach_zettaflops_/ (16 September 2015, date last accessed).
53. Koomey, J, Berard, S and Sanchez, M *et al*. Implications of historical trends in the electrical efficiency of computing. *IEEE AnnHis Comput* 2010; **33**: 46–54.
54. Chen, Y, Luo, T and Liu, S *et al*. DaDianNao: a machine-learning supercomputer. *Proceedings. of the 47th IEEE/ACM International Symposium on Microarchitecture (MICRO-47)*, 2014:609–22.
55. Chen, Y, Chen, T and Xu, Z *et al*. DianNao family: energy-efficient hardware accelerators for machine learning. *Accepted to appear in Communications of the ACM*.
56. Luo, Z, Zhou, Y and Li, J *et al*. Cyberinfrastructure for joint research in Qinghai Lake nature reserve. *Procedia Environ Sci* 2011; **10**: Part B: 1781–90.
57. Xu, Z and Li, G. Computing for the masses. *Commun ACM* 2011; **54**: 129–37.
58. Alkhatib, H, Faraboschi, P and Frachtenberg, *et al*. What will 2022 look like? The IEEE CS 2022 report. *IEEE Comput* 2015; **48**: 68–76.