Preface

On November 8, 2004, a 16-rack, 16,384-node Blue Gene®/L supercomputer was crowned as the fastest supercomputer in the world on the 24th TOP500 list (*http://www.top500.org*). IBM was regarded as the most innovative computer manufacturer. The sustained LINPACK benchmark number for this Blue Gene/L was 70.72 teraflops (a trillion floating-point operations per second), or 77% of its peak performance of 91.75 teraflops.

This double issue describes the work of the Blue Gene/L supercomputer project, which is aimed at expanding the horizon of high-performance computing to unprecedented levels of scale and performance. Blue Gene/L is the first supercomputer in the Blue Gene family. The full Blue Gene/L consists of 64 racks containing 65,536 high-performance compute nodes; i.e., it is four times the size of the Blue Gene/L system that won the 2004 TOP500 race. Each node (nodes and chips are the same in the Blue Gene system) contains two embedded 32-bit PowerPC® processors. Furthermore, the same chip that is used for compute nodes is also used for the 1,024 I/O nodes. A three-dimensional torus network and a collective network are used to interconnect all nodes. The full system contains 33 terabytes of main memory; it is designed to achieve 183.5 teraflops peak performance using one of the processors of each node for computation and the other processor for communication, and 367 teraflops using both processors for computation.

One of the key objectives in Blue Gene/L design is to achieve cost/performance comparable to the COTS (Commodity Off The Shelf) approach, while at the same time incorporating a processor and network combination so powerful that it revolutionizes the performance of supercomputer systems. In 1999, IBM formulated a vision to design such a computer. When the idea was shared with the Lawrence Livermore National Laboratory (LLNL), they saw that part of their computing needs could be satisfied with the Blue Gene approach in a cost-effective way that has never been achieved before. Since then, IBM and LLNL have worked as partners to develop the system.

Hardware

The first paper in this issue is an overview of the Blue Gene system design whose leading author is Dr. Alan Gara, the chief architect of the Blue Gene project. This paper outlines not only the important architectural features, but also the purpose of the system design.

A key differentiator of the Blue Gene family is its low-power design. In order to maximize the performance of a rack in a thermally limited regime, we chose to use the most power-efficient processor. A moderate-frequency (700-MHz), low-power, embedded PowerPC processor consistently outperforms the high-frequency, high-power microprocessors at the rack level by a factor of 2 to 10 times. Once this low-power design point was adopted, it was up to the packaging team to assemble as many processors as possible in a given rack to realize high rack-level performance. Thus a 1,024-node rack was shoehorned into a compact volume of 2 ft $\times$ 3 ft $\times$ 6 ft. The packaging design, along with the cabling together of up to 64 racks, is described in the paper by Coteus et al.

The paper by Wait invites us to look at the design of a dual-floating-point unit (FPU), called the double hummer, which is associated with each processor. This double-precision FPU has a wide datapath that enables quadword (128-bit) load and store; at 700 MHz, it achieves 2.8 gigaflops. Since there are two processors and two double hummers per chip, each Blue Gene/L node (chip) can execute 5.6 gigaflops.

The paper by Ohmacht et al. describes the design of a sophisticated memory subsystem that meets not only the latency and bandwidth requirements of two processors and two double hummers, but also those of networks that communicate with other processors, hosts, and RAID subsystems via Gigabit Ethernets.

What distinguishes the Blue Gene family from a commercial cluster is powerful networks which connect all of the processors. All compute nodes are connected by a three-dimensional torus, which is detailed in the paper by Adiga et al. This communication backbone provides the point-to-point interconnection. Most scientific problems, using the domain decomposition technique, rely heavily on the three-dimensional nearest-neighbor communication which is fulfilled by this network. For collective communication such as reduction and broadcast, we provide a separate network known as the *collective network*. By integrating the internode networks, we can take advantage of the same generation of technology used in the processor logic; i.e., these networks scale naturally with chip frequency. Furthermore, the off-chip drivers and receivers can be optimized to consume less power than those of industry-standard networks.

The system-on-a-chip approach integrates 95 million transistors which are used to build two processors, two double hummers, an L2 cache, a 4-MB embedded DRAM L3, a torus, a collective network, Gigabit Ethernet, and a JTAG network on a single chip. The next three papers describe key aspects of the Blue Gene/L compute chip which are essential to ensure the proper functionality of the chip. The paper by Bright et al. documents the chip synthesis, timing, and physical design. The paper by Haring et al. describes the infrastructure

for control, testing, and bring-up, and the paper by Wazlowski et al. details the verification strategy.

The paper by Giampapa et al. describes the Blue Gene/L advanced diagnostics environment. This bottom-up diagnostic kernel software was vital in the bring-up procedure, since it exercised all of the functionality of the chips. In contrast, the system software started from a top-down design, because its overriding concern was to present users with a familiar environment, including system calls, operating system, programming languages, and various libraries for communication, mathematical functions, and diagnostics.

The paper by Iyer et al. provides the technology background of the IBM 0.13-μm-generation embedded DRAM (dynamic random access memory). In comparison with SRAM-based cache, which is typically 0.5–1 MB, the on-chip L3 cache was enlarged to 4 MB by incorporating embedded DRAM technology.

The Columbia QCD (Quantum Chromodynamics) team, under the leadership of Professor Norman Christ, and the IBM Blue Gene team have been collaborating since early 1999. The last paper in the hardware section, by Boyle et al., describes the Blue Gene "predecessors," QCDSP and QCDOC.

Software

Software has often been considered the Achilles heel of supercomputers. One of the most challenging aspects of the Blue Gene/L system design has been the development of software that can be scaled to the unprecedented levels of more than a hundred thousand processors. The first paper in the Software section, by Moreira et al., gives an overview of the programming and operating environment on Blue Gene/L and describes how the software team dealt with the unique challenge posed by the scale of Blue Gene/L.

The dual-floating-point unit that was added to each processor of the Blue Gene/L compute node was designed by architects in consultation with the compiler developers and algorithm designers for high-performance numerical libraries. The next paper, by Chatterjee et al., describes the hardware–software co-design of the dual-floating-point unit. It explains the rationale for an interesting variant of a SIMD (single-instruction multiple-data) architecture used for this dual FPU, and details how the compiler and expert library developers exploit its instruction set.

Blue Gene/L supports the popular Single Program Multiple Data programming model, with explicit message passing among parallel processors using the *Message Passing Interface* (MPI) standard. Hence, the performance and scalability of the MPI implementation are key to parallel application performance on Blue Gene/L. The IBM team collaborated with researchers from the Argonne National Laboratory in the development of a highly scalable MPI library that has been tuned to exploit the powerful Blue Gene/L networks. The paper by Almási et al. describes the design and implementation of the MPI library for Blue Gene/L.

Developers of parallel applications invariably face surprises and new challenges as they attempt to achieve high performance for their applications on any new platform. Performance tools help the application developers understand the performance bottlenecks and ways to improve the performance of their codes on the given system. The paper by Martorell et al. describes performance tools that have been developed for Blue Gene/L.

Any parallel system, especially a massively parallel machine such as Blue Gene/L, can be used to support several parallel applications simultaneously. A job scheduler provides a convenient interface to the multiple end users that wish to use the parallel machine for their jobs, and helps utilize that precious computational resource effectively. The fifth paper in the Software section, by Aridor et al., describes resource allocation and job scheduling for Blue Gene/L.

Many scientific applications employ a fast Fourier transform (FFT) algorithm during some phase in their computation. Hence, a high-performance FFT library is considered a key element in the utilities provided on a high-performance machine. The paper by Lorenz et al. describes an effective self-tuning approach that has been customized for Blue Gene/L and used to obtain high levels of FFT performance on the Blue Gene/L compute node.

Science and applications

A major objective for the Blue Gene project is to apply this unprecedented scale and performance of computing to study truly challenging problems in science, where such high-performance computing can achieve significant impact. From the outset, the Blue Gene team recognized the need to have such a thrust to provide a rigorous end goal that could be achieved by only this unprecedented scale of computing, as well as to provide the focus and guidance for difficult design choices. Significant leaps in scale and performance have often benefited from such bold application challenges.

To provide such motivation, the Blue Gene project chose a problem in life sciences. Stated briefly, the goal is to study the mechanism behind protein folding, a seemingly ubiquitous occurrence in biology that has defied detailed understanding in terms of the physics and chemistry of the process. Computational chemists have over the last few decades pursued several approaches,

from very detailed atomistic simulations to quite abstract physical and thermodynamic models that attempt to describe aspects of the protein-folding process. A petaflop-scale computer would begin to yield meaningful timescales for protein simulation and allow scientists to connect with fast timescale experimental observations of the protein-folding phenomenon. In addition, the atomistic simulation techniques and the processing capability of Blue Gene could be applied to study other problems in computational sciences (for instance, in life sciences—the behavior of large biomolecular complexes) or to characterize the molecular systems in quantitative detail with a very accurate representation to facilitate the development of faster, more abstracted representations for molecular simulations.

The first paper in the Science and applications section, by Germain et al., describes Blue Matter, a software application framework and environment which was developed from scratch, with the primary motivation of achieving optimal performance on Blue Gene/L while providing a robust environment for atomistic molecular dynamics simulations using any of the several popular force fields.

Eleftheriou et al. focus on the implementation of fast Fourier transforms on Blue Gene/L. FFTs are often a significant fraction of many computational physics and computational chemistry problems. In this instance of atomistic molecular simulations, FFTs are a necessary and significant part of the computation cost in obtaining accurate long-range electrostatic forces. This paper provides a novel slab decomposition approach to multidimensional FFTs that scales well to thousands of nodes.

Obtaining the best performance for the application requires tailored and optimized implementation of the frequently used mathematical functions in the software library. The paper by Enenkel et al. presents a description of the mathematical functions of particular relevance to molecular dynamics and their implementation on Blue Gene/L.

The size of the molecular system and the attendant geometric complexity, coupled with the need to observe and categorize events in simulation of unprecedented timescales, call for new analysis techniques. The paper by Suits et al. describes the approach to analysis of large-timescale molecular dynamics trajectories, a problem of particular relevance with Blue Matter on Blue Gene/L.

The paper by Bhanot et al. addresses the issue of mapping the computational problem onto hardware in a manner that provides optimal performance and scaling for the application. They employ a simulated annealing approach to optimize the layout of tasks onto the 64,000 nodes of Blue Gene/L.

This double issue of the *IBM Journal of Research and Development* is dedicated to the hardware and software teams of the Blue Gene/L project. Their innovative hard work has opened a new chapter in the history of supercomputer design, performance, and application.

George L.-T. Chiu
Senior Manager
Advanced Hardware Systems
IBM Research Division

Manish Gupta
Senior Manager
Emerging Systems Software
IBM Research Division

Ajay K. Royyuru
Senior Manager
Computational Biology Center
IBM Research Division

Guest Editors