Technical Editor:
Marcin Paprzycki
Dept. of Computer Science and Statistics
Univ. of Southern Mississippi
Southern Station 1506
Hattiesburg, MS 39406-1506
m.paprzycki@usm.edu

High Performance Cluster Computing

Reviewed by Lalit M. Patnaik, Indian Institute of Science

High Performance Cluster Computing:
Architectures and Systems, Vol. 1
Programming and Applications, Vol. 2
Rajkumar Buyya, ed.
881 and 700 pages
$54.00 each
Printice Hall PTR
Upper Saddle River, New Jersey
1999 and 2000
0-13-013784-7 and 0-13-013785-5

Because high-performance cluster computing is a relatively new area, few books successfully cover the topic. Rajkumar Buyya provides an authoritative overview of the field and its relevant state-of-the art research directions. Both volumes stem from his interaction with leading researchers, offering in-depth coverage for scientists and engineers engaged in the research, development, and application of high-performance computing systems. (Buyya's Web site, www.dgs.monash.edu.au/~rajkumar/cluster/index.html, offers a wealth of additional information.) Graduate students should also find these two books to be extremely useful, especially when exploring research topics.

Volume 1: Architectures and systems

Volume 1 contains 36 chapters divided into four parts. Each part features an introductory chapter that deals with key issues, making this volume more accessible to readers with minimal background.

Part I focuses on requirements and general issues, discussing how to set up and administer a cluster and offering detailed information about security policies and system monitoring. Part II covers the latest trends in networking, protocols, and I/O. In particular, Chapter 9 offers detailed information on high-speed networks, discussing design issues, Fast Ethernet, high-performance parallel interfaces, and asynchronous transfer modes. Chapter 14 discusses load balancing over networks, and Chapter 15 covers multiple-path communication. Chapter 17 delves into an interesting trend in parallel computing—distributed shared memory—discussing network performance as well as design issues at length.

Part III covers process scheduling, load sharing, and balancing. Chapters 20 through 25 extensively deal with job and resource management systems, scheduling parallel jobs on clusters, load sharing, parallel program scheduling, dynamic load balancing, and mapping and scheduling on heterogeneous systems.

Part IV discusses representative cluster systems such as Beowulf-class Linux clusters; COMPaS, a Pentium Pro PC-based SMP cluster; the BSP-Based Adaptive Parallel Processing System; and MARS, an adaptive parallel programming environment. A particularly interesting topic is a scalable and highly available cluster Web server. Chapter 36 presents key issues for building a Web server that scales with increased computing requirements, efficiently using cluster technology and allowing online maintenance while providing high availability of Web servers.

The in-depth coverage of so many topics makes *High Performance Cluster Computing: Architectures and Systems* unique, and the bibliographies concluding each chapter are a great resource for recent publications and URLs. However, the book would have been more helpful had it provided advanced research-level problems at the end of each chapter.

Volume 2: Programming and applications

Buyya separates the 29 chapters of Volume 2 into three parts, concluding each chapter with an extensive bibliography, as he did in Volume 1.

Part I deals with programming environments and development tools. The first two chapters concisely cover concepts related to parallel programming models, paradigms, languages, and environments. Chapters 3 and 4 discuss the most popular message-passing environments, MPI and PVM, and three software packages

that link these two together—PLUS, PACX-MPI, and PVMPI. Part I also covers topics such as active objects, the distributed shared data system, and the Library for Parallel Systems, discussing how to implement distributed applications on different architectures and heterogeneous computers. Chapter 9 goes over the Tuple space paradigm through Linda and Chapter 10 elegantly explains how to debug parallel code. The last chapter of Part I is particularly important, focusing on Web operating systems and system infrastructures for efficiently developing wide-area applications.

Part II focuses on Java in high-performance computing, discussing different models that support distributed object-oriented computation. Chapter 12 has excellent coverage of distributed-object models such as RMI, CORBA, DCOM, and Voyager and discusses parallel programming models. Chapter 13 presents the hot topic of Web-based parallel computing with Java, along with a case for the JET platform.

Part III presents newly developed and newly applied algorithms and applications for a cluster environment. Chapter 17 is of great interest because it discusses load balancing over heterogeneous workstations, offering performance results. Part III also explains time management in parallel simulation with synchronization protocols, algorithms, and hardware system simulations with the network enabled parallel simulator as a test bed. In addition, it details the implementation of several applications such as Lazzy evaluation of parallel ray tracing, image retrieval, database and knowledge base systems, climate ocean modeling, biomedical system modeling, and computational fluid dynamics simulation.

High Performance Cluster Computing: Programming and Applications is suitable for graduate students with a strong background in software for high-performance computers. It is ideal for a one-semester course with selected topics chosen from Part III. Almost all chapters include sections highlighting current research challenges; unfortunately, the book would have been more interesting and helpful had it come with a CD containing important programming examples. Also, unlike Volume 1, at times the chapters seem disjointed. Such things happen when several authors offer contributions on diverse topics, but a minor reorganization of the chapters could have overcome this problem.