

# Visualization and Debugging in a Heterogeneous Environment

Adam Beguelin, Carnegie Mellon University and Pittsburgh Supercomputing Center  
Jack Dongarra, University of Tennessee and Oak Ridge National Laboratory  
Al Geist, Oak Ridge National Laboratory  
Vaidy Sunderam, Emory University

**T**he emergence of a wide variety of commercially available parallel computers has created a software dilemma. Will it be possible to design general-purpose software that is both efficient and portable across these new parallel computers? Moreover, will it be possible to provide programming environments sophisticated enough for explicit parallel programming to exploit the performance of these new machines? For many computational problems, the design, implementation, and understanding of efficient parallel algorithms can be a formidable challenge. Additional issues of synchronization and multiple-task coordination make efficient parallel programs more difficult to write and understand than efficient sequential programs. Parallel programs are often less portable than serial codes because their structure may depend critically on the hardware's specific architectural features (such as how it handles data sharing, memory access, synchronization, and process creation).

The computing requirements of many current and future applications, ranging from scientific computational problems in the material and physical sciences to simulation, engineering design, and circuit analysis, are best served by concurrent processing. Multiprocessors can frequently address the computational requirements of these high-performance applications, but other aspects of concurrent computing are not adequately addressed when conventional parallel processors are used.

For instance, software aspects, including program development methods, scalable programs, profiling tools, and support systems, require significant development. While hardware and architectural advances in parallelism have been rapid, the software infrastructure has not kept pace, resulting in unsystematic and ad hoc approaches to the implementation of concurrent applications. In recent years, several research groups have focused on various aspects of this shortcoming, producing significant developments in programming paradigms, data partitioning, algorithms, languages, and scheduling.

Heterogeneous networks of computers ranging from workstations to supercomputers are becoming commonplace in high-performance computing. Until recently, each computing resource on the network remained a separate unit, but now hundreds of institutions worldwide are using the Parallel Virtual Machine<sup>1</sup> soft-

**A monitoring tool and a graphical interface working on top of the PVM software can help programmers make better use of heterogeneous networks of computers.**

