

**Evaluating the Impact of Programming Language Features on the
Performance of Parallel Applications on Cluster Architectures**

Jan Prins, Bill Pugh, P. Sadayappan, Chau-Wen Tseng

To appear at the *16th Workshop on Languages and Compilers for Parallel
Computing (LCPC03)*, College Station, TX, 2-4 October 2003

Abstract

Developing parallel programs is known to be difficult, particularly for distributed-memory architectures. Parallel programming languages attempt to reduce the amount of user effort needed by providing high-level language features for specifying functions such as interprocessor communication and synchronization. However, using these language features may impose a performance overhead, depending on characteristics of the compiler, run-time system, and of the parallel architecture. In this paper, we evaluate the impact of programming language features on the performance of parallel applications on modern parallel architectures, particularly for the demanding case of sparse integer codes. We compare a number of programming languages (OpenMP, MPI, UPC, Global Arrays) on both shared and distributed-memory architectures. We find that language features can make parallel programs easier to write, but cannot hide the underlying communication costs for the target parallel architecture. Some features such as fine-grain remote accesses are inherently expensive. Powerful compiler analysis and optimization can help reduce software overhead. To avoid large reductions in performance, language features must avoid interfering with compiling local computations.