Parallel Java: An API for Teaching and Developing Parallel Programs in 100% Java

Alan Kaminsky
Rochester Institute of Technology

Follow this and additional works at: http://scholarworks.rit.edu/other

Recommended Citation
Alan Kaminsky. Parallel Java: An API for teaching and developing parallel programs in 100% Java. ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE 2006), Houston, TX, USA, March 3, 2006, faculty poster.
PARALLEL JAVA

AN API FOR TEACHING AND DEVELOPING PARALLEL PROGRAMS IN 100% JAVA

Alan Kaminsky, Department of Computer Science, Golisano College of Computing and Information Sciences, Rochester Institute of Technology

PARALLEL COMPUTING

- Shared memory multiprocessor (SMP) parallel computer
  - Multiple threads, each running on its own CPU
  - Threads share variables in main memory
  - OpenMP standard for SMP programming

- Cluster parallel computer
  - Multiple processes, each running on its own processor
  - Processes communicate data by message passing through the network
  - MPI standard for cluster programming

- All CS students need to learn parallel computing
  - Computers shifting to multicore chips (SMPs)
  - Multicore chip based computers require parallel programs for full performance
  - OpenMP and MPI are C/Fortran based and are not object oriented
  - Java is becoming the language of choice for teaching programming
  - Middleware is needed for parallel programming in Java: ParallelJava!

CODING WITH PARALLEL JAVA

PJ program for computing the Mandelbrot Set on an SMP parallel computer

PARALLEL JAVA PERFORMANCE

Output image (N x N pixels)

Features of Parallel Java

- PJ itself is written in 100% Java (JDK 1.5)
- PJ's SMP parallel features inspired by OpenMP
  - Parallel thread teams
  - Parallel loops with selectable scheduling
- Parallel sections and section groups
- Variable scoping: shared, thread local
  - Reduction variables with arbitrary operations
- PJ's cluster parallel features inspired by MPI
  - PJ middleware automatically runs a program on multiple processors of the cluster
  - Message passing of primitive types and non-primitive types (Java Object Serialization)
  - Message passing of arrays and matrices, or arbitrary portions thereof
  - Message passing operations: send, receive, sendReceive, broadcast, scatter, gather, allGather, reduce (others tba)
  - Reduction with arbitrary operations
- PJ supports hybrid SMP cluster parallel programming
- PJ facilitates teaching parallel programming
  - Students who know Java find it easy to write parallel programs in PJ
  - PJ teaches the concepts of OpenMP and MPI in a 100% Java setting
  - PJ comes with an extensive library of example programs for SMP and cluster computers

For further information

- Alan Kaminsky, ark@cs.rit.edu
- PJ Library (GNU GPL licensed): http://www.cs.rit.edu/~ark/
- Parallel Computing I courseware: http://www.cs.rit.edu/~ark/531/

PJ authors: Alan Kaminsky and Luke McOmber
Presented at SIGCSE 2006, Houston, 03-Mar-2006