

MPI for Java

Bryan Carpenter, Vladimir Getov, Glenn Judd,
Tony Skjellum, Geoffrey Fox and others

Introduction and Background

- The existing communication packages in Java
 - RMI, API to BSD sockets - are optimized for Client/Server programming
- The symmetric model of communication is captured in the MPI standard - MPI-1 and MPI-2
- A Java-MPI API specification is urgently needed to enable the development of portable JavaGrande applications using MPI

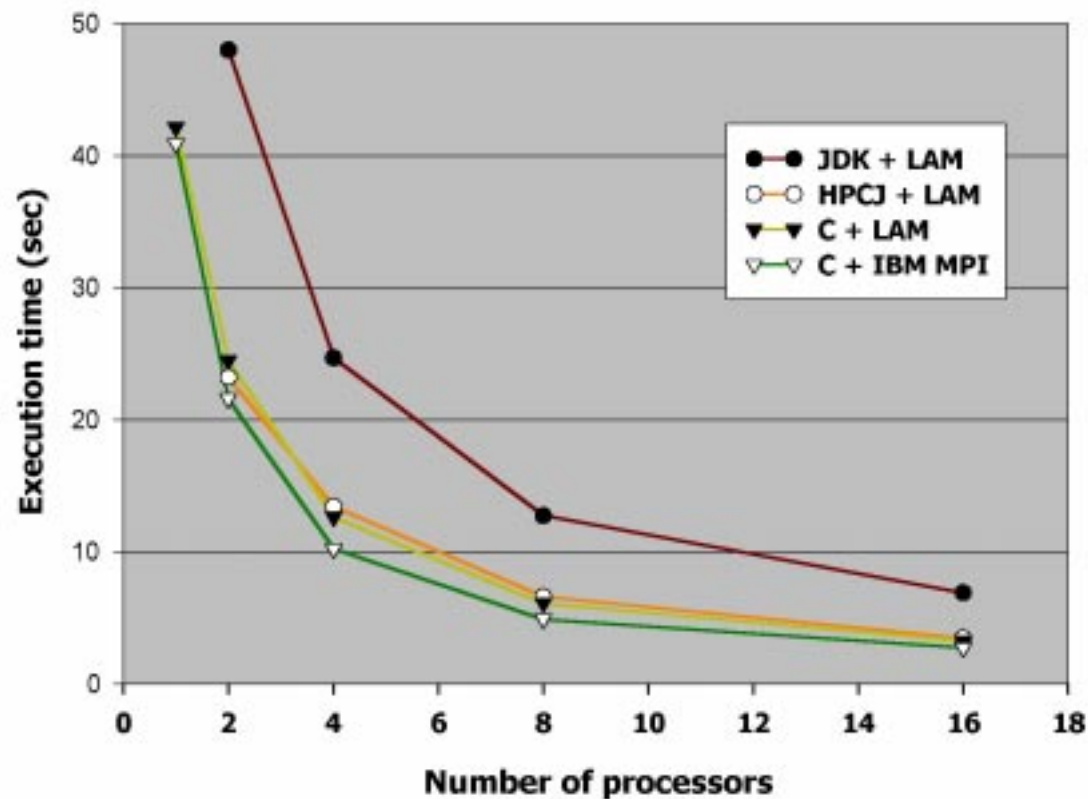
Current Status

- **mpiJava** - Modelled after the C++ binding for MPI. Implementation through JNI wrappers to native MPI software. <http://www.npac.syr.edu/projects/pcrc/HPJava/>
- **JavaMPI** - Automatic generation of wrappers to legacy MPI libraries. C-like implementation based on the JCI code generator. <http://perun.hscs.wmin.ac.uk/JavaMPI/>
- **MPIJ** - Pure Java implementation of MPI closely based on the C++ binding. A large subset of MPI is implemented using native marshaling. <http://ccc.cs.byu.edu/DOGMA/>

Current Status (contd.)

- JMPI - MPI Soft Tech Inc. have announced a commercial effort under way to develop a message passing environment for Java.
- Others
- Current ports - Solaris (both WS clusters and SMPs), AIX (both WS clusters and SP2), Windows NT clusters, Origin-2000, Fujitsu AP3000, and Hitachi SR2201.
- Java + MPI codes - growing variety including full applications

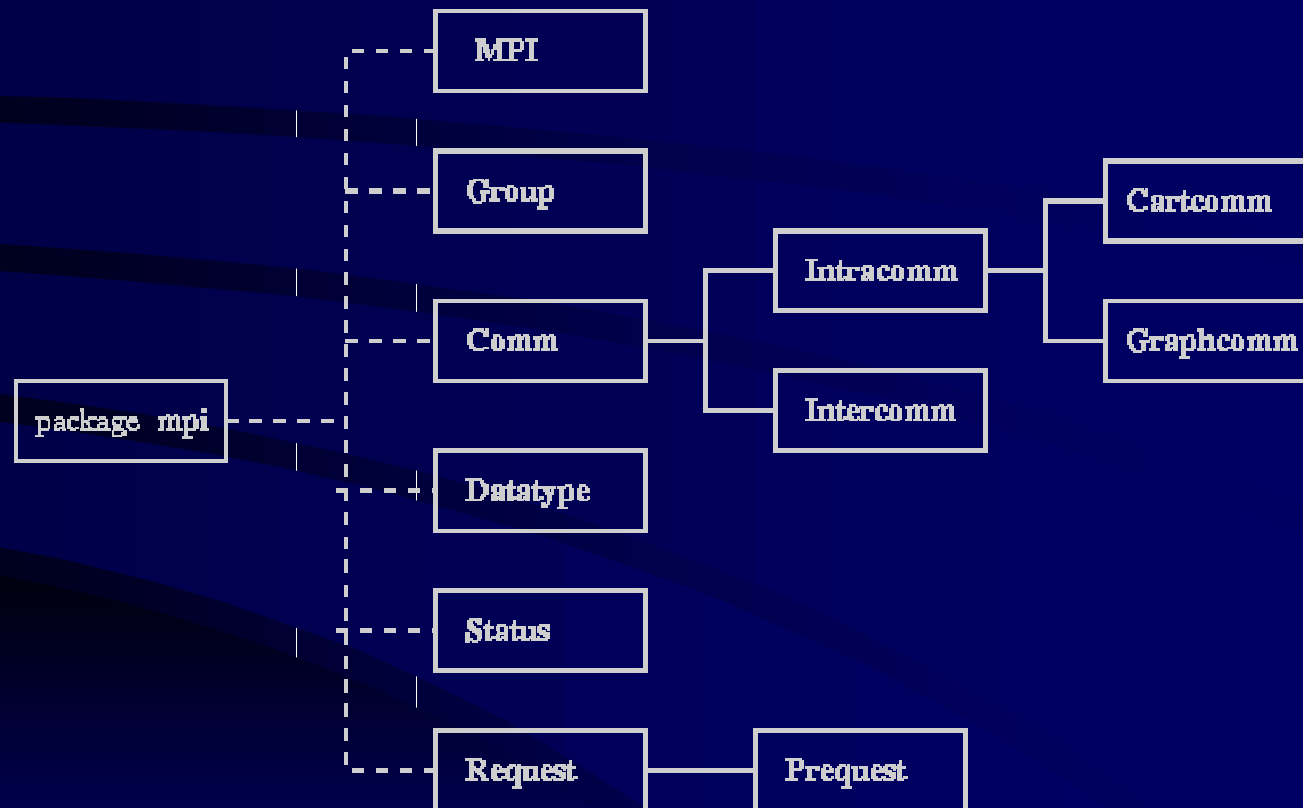
JavaMPI: NPB IS kernel on IBM SP2



Java-MPI API Specification

- First phase in our work on MPI for Java.
- Builds on MPI-1 Specification and the current Java Specification.
- Provide immediate standardization for common message passing programs in Java
- Provide a basis for conversion between C, C++, Fortran and Java.
- Eventually, support for aspects of MPI-2 as well as possible improvements to the Java language.

mpiJava Class Hierarchy



Advanced Message Passing for Java

- Second phase in our work on MPI for Java
- Current task - to offer a first principles study of MPI-like services in an upward compatible fashion
- Purpose - performance and portability
- Fundamental look at data marshaling
- Preference for Java-natural mechanisms

Further Actions

- Standard Java-MPI Specification - Draft document at www.javagrande.org
- Discussion and comments to javagrande@npac.syr.edu
- Closer contact with C/Fortran MPI Forum community
- Final document - next spring

Further Actions (contd.)

- Java-MPI wrapper(s) publicly available
- Intelligent code generator of wrappers
- Pure Java MPI implementation
- Test suite
- Java-MPI benchmarks
- Advanced MPI for Java research