ANNUAL REPORT FOR AWARD # 0096236

Geoffrey C Fox ; *Florida State University*
Data Parallel SPMD Programming Models from Fortran to Java

Participant Individuals:

Senior personnel(s) : Bryan Carpenter
Graduate student(s) : Sung-Hoon Ko; Han-Ku Lee; Sang Boem Lim

Participants' Detail

Partner Organizations:

Other collaborators:

Xiaoming Li, Peking University, China.
Co-author of original proposals for HPspmd.
Carpenter visited him in Peking for lectures earlier this year.

Vladimir Getov, University of Westminster, UK.
Organized the Java Grande Message-Passing Working Group, in which we were involved.

Mark Baker, University of Portsmouth, UK.
Collaborated on Java MPI. Investigating Jini-based approaches.

Other participants in the Java Grande Forum.

Activities and findings:

Research and Education Activities:

We have been developing a model for parallel programming that integrates high-level, data-parallel features from languages like High Performance Fortran (HPF) with well-established, library-based approaches to programming distributed memory parallel computers. In particular we have been implementing and testing these ideas in an environment called HPJava, built around an 'HPspmd' programming language extended from Java.

An early version of an HPJava translation system was reported last year. This year, in the light of experiences with the experimental system, we undertook a major rewrite of the translator. The main goal was to implement an improved translation scheme that we hope will provide realistic performance. We also took the opportunity to bring the standard of checking and error-reporting in our HPJava front-end closer to what would be expected in a practical tool. The new translator is approaching completion, and we hope to be able report results of early benchmarks at a forthcoming workshop (ref 1). The current translation scheme is documented in detail in (ref 2). As a by-product of the efforts to produce an efficient translator, many details of the language definition have been refined and clarified, by now we can more more definitely characterize the HPJava language. HPJava is a strict extension of Java. It incorporates all of Java as a subset (including, for example, nested classes). Any existing Java class library can be invoked from an HPJava program without recompilation. HPJava adds to Java a concept of multi-dimensional, distributed arrays, closely modelled on the arrays of High Performance Fortran. Sequential multi-dimensional arrays--essentially equivalent to Fortran 95 arrays--are available as a subset of the HPJava distributed arrays. Regular sections of distributed arrays are fully supported. The multidimensional arrays can have any rank, and the elements of distributed arrays can have any standard Java type, including Java class types and Java array types.

A translated and compiled HPJava program is a standard Java class file, which will presumably be executed by a collection of JIT-enabled Java Virtual Machines. All externally visible attributes of an HPJava class--e.g. existence of distributed-array-valued fields or method arguments--can be transparently reconstructed from Java signatures stored in the class file. This makes it possible to build libraries operating on distributed arrays, while maintaining the usual portability and compatibility features of Java. The libraries themselves can be implemented in HPJava, or in standard Java, or as JNI interfaces to other languages. The HPJava language specification carefully documents the mapping between distributed arrays and the standard-Java components they translate to.

Although HPJava incorporates HPF-like distributed arrays, it does not incorporate HPF-like 'sequential' semantics for manipulating these arrays. Besides features designed to support library interfaces, HPJava DOES add a handful of high-level features designed to support direct programming with distributed arrays, including a distributed loop construct called ‘overall’. To directly support lower-level SPMD programming, it also provides a complete set of inquiry functions that allow the local array segments in distributed arrays to be manipulated directly, if preferred.

Findings:

Continuing developments encourage our belief that the HPspmd approach can efficiently and elegantly combine several important paradigms for parallel computing, and our hope that the HPJava system in particular has the potential to become a practical tool for 'scientific' computing in the near future.

We are encouraged by reports at recent Java Grande and JavaOne, conferences claiming that--on important commodity platforms--Java JIT performance is now approaching parity with C and Fortran compilers, and also by recent discussions in Java Grande Forum meetings highlighting the pressing need for preprocessors to introduce multi-dimensional arrays and other 'scientific' features into Java.

Training and Development:

A series of lectures on HPspmd and related issues was developed and delivered by Carpenter at Peking University in January 2000. The lectures are online at http://aspen.csit.fsu.edu/pss/HPJava/beijing.html

Several graduate students have worked on the project. Sung-Hoon Ko received his Ph.D. degree on work related to this project in August 2000.

Two graduate students are currently occupied full-time in developing and testing the HPJava translator and associated software, and eventually expect to complete Ph.D.'s on related materials.

Outreach Activities:

Java Grande Forum http://www.javagrande.org is related and a very successful community activity.

Journal Publications:

Bryan Carpenter, Geoffrey Fox, Sung Hoon Ko and Sang Lim, "Object Serialization for marshaling data in a Java interface to MPI", *Concurrency: Practice and Experience*, vol. 12, (2000), p. . Accepted

Bryan Carpenter, Vladimir Getov, Glenn Judd, Anthony Skjellum, Geoffrey Fox, "MPJ: MPI-like message passing for Java", *Concurrency: Practice and Experience*, vol. 12, (2000), p. . Accepted

Book(s) of other one-time publications(s):

Sung Hoon Ko, "Java for High Performance SPMD and Message Passing Programming" , bibl. Syracuse University Graduate School Dissertation, (2000). *Thesis* Degree awarded.

Other Specific Products:

Software (or netware)

mpiJava -- Java wrappers for MPI.

It is downloadable from the mpiJava home page: http://aspen.csit.fsu.edu/pss/HPJava/mpiJava.html

Has been downloaded by approx 400 people.

Software (or netware)

The HPJava translation system, described elsewhere in this report

It will be made available as open source software at the end of the project (if not sooner).

Internet Dissemination:

http://aspen.csit.fsu.edu/pss/HPJava/

HPJava project home page, including mpiJava project home page.

Includes HPJava language definition (see the document "Parallel Programming in HPJava"), mpiJava API definition, downloadable software for mpiJava, lectures on data parallel, SPMD programming, other papers and presentations.

Contributions:

Contributions within Discipline:

The HPspmd model attempts to integrate some of the best features from HPF and MPI-style, SPMD programming with libraries, in a practical and elegant way. These ideas will be captured in an environment based on the most successful language of recent times for Internet programming -- Java.

Contributions to Education and Human Resources:

4 graduate students were actively involved in the project at Syracuse. One recently graduated. 2 have transferred with the project to Florida State University.

We have extensively involved students in the ACM Java Grande conferences which were initiated from related research several years ago.

Contributions to Resources for Science and Technology:

(Not directly using Java for scientific applications, but somewhat related) in the last year the senior personnel on the project (Fox and Carpenter) developed a two-part, Java-based, inter-disciplinary, graduate course on information technology for the School of Computational Science and Information Technology at Florida State University:

http://aspen.csit.fsu.edu/it1spring01/
http://aspen.csit.fsu.edu/it2spring01/

Special Requirements for Annual Project Report:

Unobligated funds: less than 20 percent of current funds

Categories for which nothing is reported:
Participants: Partner organizations
Contributions to Other Disciplines
Contributions Beyond Science and Engineering
Special Reporting Requirements
Animal, Human Subjects, Biohazards

Submit
Return
