RLF 5700.12
(01/98)

U. S. DEPARTMENT OF ENERGY
Richland Operations Office

**FIELD WORK PROPOSAL**

| 1. Work Package Number | 2. Contractor Project Number | 3. Date Prepared | 4. Budget and Reporting Code |
|---|---|---|---|
| K86695 | 28646 | 03-01-98 | KJ-01-01-03-0 |

5. Package Title
Extending The Scientific Template Library with Shared Memory NUMA Programming Model

| 6. Headquarters/Operations Office Program Manager (Name: Last, First, Middle Initial) | 7. Headquarters Organization | 8. DOE-HQ Organization Code |
|---|---|---|
| Scott, Mary-Anne (301) 903-6368 | ER | ER-31 |

| 9. DOE Operations Office Reviewer (Name: Last, First, Middle Initial; Phone: area code-7 digit #) | 10. DOE Operations Office | 11. DOE Organization Code |
|---|---|---|
| Brown, Dennis A (509) 372-4030 | Richland Operations Office | RL |

| 12. Principal Investigator (Name: Last, First, Middle Initial; Phone: area code-7 digit #) | 13. Contractor Name | 14. Code |
|---|---|---|
| Nieplocha, Jarek (509) 372-4469 | Battelle Memorial Institute<br>Pacific Northwest National Laboratory | 48 |

15. Work Proposal Description (Approach, Anticipated Benefit, in 200 Words or Less)

In this proposal, we build upon a set of powerful and proven paradigms for managing the complex memory hierarchy in MPPs, and fully integrate that functionality into key areas of the ACTS toolkit, greatly extending its capabilities for applications requiring shared-memory communication and for computer systems having non-uniform memory access architecture (essentially all present and future MPPs). Our approach combines the strengths of the shared-memory-programming model with the scalability and performance of the distributed-memory model. What has emerged is a highly successful Global Arrays model (GA), whose one-sided communications model yields clean code with outstanding performance and scalability, as well as compatibility with standards and common linear algebra packages. Current applications span a wide range, including molecular dynamics, computational chemistry, graphics rendering, and security value forecasting. GA is used by major centers and applications development groups nationwide and worldwide. In the spirit of a standards-based approach, this toolkit will be integrated with other ACTS toolkits in a standards compliant and portable fashion, so that applications can readily exploit the combined advantages of these systems.

| 16. Contractor Subprogram Manager | | 17. DOE Operations Office Review Official | |
|---|---|---|---|
| (Signature) | (Date) | (Signature) | (Date) |

18. Detail Attachments:

- ☒ A. Purpose
- ☒ B. Approach
- ☒ C. Technical Progress
- ☒ D. Future Accomplishments
- ☒ E. Relationships to Other Projects
- ☐ F. Explanation of Milestones
- ☐ G. Publications
- ☒ H. Other: Enhanced Proposal
- ☐ I. Waste Management

RLF 5700.12R
(01/98)

**WORK PROPOSAL REQUIREMENTS FOR OPERATING/EQUIPMENT OBLIGATIONS AND COSTS**

| Contractor Name | Work Package No. | Contractor Project No. | Date Prepared |
|---|---|---|---|
| Battelle Memorial Institute<br>Pacific Northwest National Laboratory | K86695 | 28008 | 03-01-98 |

| | FY 1998 | FY 1999 | FY 2000 Requested | FY 2000 Authorized | FY 2001 | FY 2002 | FY 2003 | FY 2004 |
|---|---|---|---|---|---|---|---|---|
| 19. Staffing (in Staff Years) | | | | | | | | |
| A. Scientific | .7 | .7 | .7 | | | | | |
| B. Other Direct | 0 | 0 | 0 | | | | | |
| C. Total Direct | .7 | .7 | .7 | | | | | |
| 20. Operating Expense (in Thousands) | | | | | | | | |
| A. Total Obligations* | 146 | 267 | 270 | | | | | |
| B. Total Costs | 146 | 267 | 270 | | | | | |
| 21. Equipment (in Thousands) | | | | | | | | |
| A. Obligations | | | | | | | | |
| B. Costs | | | | | | | | |

| 22. Tasks or Milestones | Proposed Dollars FY 1998 | FY 1999 | FY 2000 |
|---|---|---|---|
| | | | |

Footnotes:

*Reflects $117K additional funding in FY1999-2000 for NPAC Syracuse required to implement work described in the enhanced proposal (Section H). The requested funding for PNNL is the same in either case.

RLF 5700.12A
(01/98)

U. S. DEPARTMENT OF ENERGY
Richland Operations Office

**Field Work Proposal (Continuation Sheet)** 

| Contractor Name | Work Package No. | Contractor Project No. | Date Prepared |
|---|---|---|---|
| Battelle Memorial Institute<br>Pacific Northwest National Laboratory | K86695 | 28008 | 03-01-98 |

## A. Purpose

The ACTS project combines strengths of multiple programming tools and environments to be used for development of application codes that address complex, multidisciplinary scientific and technical problems faced by DOE. Global Array shared memory NUMA programming model has been proven effective in development of scalable and efficient application codes in several application areas relevant to the DoE mission. We propose to further extend capabilities of Global Arrays to address needs of computational chemistry, molecular dynamics, and ground water flow modeling applications, and integrate it with other components of the ACTS toolkit such as PADRE, Nexus, Tau, and PETSc.

## B. Approach

This is an ongoing project. Key personnel: Jarek Nieplocha (60%), Robert Harrison (10%).

## C. Technical Progress

This project started in January 1998.

FY 1998

- Develop Aggregate Remote Memory Copy Interface (ARMCI), a portable one-sided communication engine optimized for noncontiguous and strided data transfers. It will be used in the multidimensional GA and contributed to other projects including PADRE.
- Develop a prototype implementation of multidimensional global arrays.
- Begin integration with PADRE.

FY 1999

- Full implementation of multidimensional global arrays.
- Optimize ARMCI for clusters of symmetric multi-processors (SMP) using Nexus.
- Incorporate Tau tracing and performance analysis into Global Arrays.
- Develop asynchronous interfaces to the GA one-sided operations to facilitate split-phase communication.
- Interoperability between GA and PADRE.

FY 2000

- Add support for sparse data structures.
- Exploit OpenMP on SMP nodes of clusters and MPPs.
- Develop interfaces to PETSc.

The Enhanced Proposal (section H) extends the FY1999 and FY2000 milestones in ways useful to the ACTS project and DOE applications.

U. S. DEPARTMENT OF ENERGY
Richland Operations Office

**Field Work Proposal (Continuation Sheet)**



| Contractor Name | Work Package No. | Contractor Project No. | Date Prepared |
|---|---|---|---|
| Battelle Memorial Institute<br>Pacific Northwest National Laboratory | K86695 | 28008 | 03-01-98 |

## D. Future Accomplishments

Based on discussions at the ACTS workshop in January 1998 and input from DOE application developers, we refined priorities for this project to be: (a) support for higher dimensional arrays and (b) integration of GA with other ACTS toolkit (PADRE, Nexus, Tau). The multidimensional arrays will be implemented on top of our portable one-sided communication library ARMCI. ARMCI will be extracted from and generalized based on the existing one-sided communication engine in GA. Unlike vendor-specific non-portable one-sided communication facilities (SHMEM, LAPI, Remote DMA, etc.) ARMCI will be optimized for low-latency high-bandwidth non-contiguous and strided data transfers such as those used in operations on multidimensional distributed arrays. ARMCI as a standalone library will be contributed to other projects such as PADRE and Parallel Runtime Consortium (PCRC) which also need one-sided communication on distributed arrays.

Interoperability of GA with other ACTS libraries and systems will provide additional value to applications by combining the strengths of several libraries: the GA NUMA-oriented shared-memory programming environment for Fortran and C codes, object-oriented C++ system for data-parallel operations on grids and multidimensional arrays in Padre and A++/P++, combined with Tau tracing and performance analysis. Nexus will be exploited in ARMCI to provide a messaging context orthogonal to (and compatible with) MPI to implement one-sided communication on clusters of workstations. Further steps in the integration task will be accomplished by developing interfaces to PETSc to enable applications codes such as ground water modeling to access numerical solver capabilities of this library in the GA environment.

Additional capabilities such as asynchronous interfaces to facilitate split-phase communication and support for sparse storage in GA will be provided to support needs of the applications in the computational chemistry area.

In addition to the capability extensions and interoperability work, a significant effort will be directed toward optimizing the performance of GA on the forthcoming system architectures such as ASCI. One of the important components of this task will be exploitation of SMP nodes and OpenMP standard.

## E. Relationships to Other Projects

Collaboration with other ACTS projects such as PADRE, Nexus, Tau, and PETSc is necessary to accomplish goals of this project. Close interactions with the application developers in the DoE community is important to maintain the traditional focus of GA on the application needs. The application areas that influence this project include computational chemistry (including the DoE Grand Challenge Actinides project) and ground water flow modeling.

## H. Enhanced Proposal

In the process of refining the objectives and approach to this project, we have developed an alternative approach to the work that will provide enhanced functionality in areas useful to DOE applications. With the additional funding requested, we propose to extend the scope of work based on contributions from the NPAC Syracuse. The Adlib runtime library from NPAC provides capabilities that could be used to facilitate implementation of the multidimensional GA extensions and collective communication functions optimized for multidimensional distributed arrays. By leveraging on Adlib (combined with ARMCI), we could implement these extensions with less effort and, with the same level of funding at PNNL as in the original proposal, commit resources to:

- providing additional features (based on Adlib) such as: group array descriptors and block cyclic array decomposition, and
- extending interoperability of GA with other ACTS components such as Globus and Nexus to support the Mirrored Array metacomputing extensions of GA in the computational grid environments. This work is independent of Adlib.

| Contractor Name | Work Package No. | Contractor Project No. | Date Prepared |
|---|---|---|---|
| Battelle Memorial Institute<br>Pacific Northwest National Laboratory | K86695 | 28008 | 03-01-98 |

In the following subsections, we summarize Adlib capabilities and describe the extended project scope that would be addressed under this proposal.

**H1: Adlib background**

Adlib (Array Distribution library) is a library for describing and manipulating distributed data arrays. It was first designed and prototyped in the SHPF project at Southampton, UK [1]. The library was redesigned, extended and reimplemented at NPAC, Syracuse in the framework of Parallel Consortium Runtime Consortium (PCRC) project [2]. PCRC has been funded by DARPA and supported a collaboration of compiler and runtime groups from Syracuse, Maryland, Rice and Indiana Universities, and several other institutes. The new library was delivered in PCRC as the "NPAC runtime kernel" [3]. Adlib provides:

- Translation of global array subscripts to processor location and local offset.
- Enumeration of the locally held blocks of a given array.
- Enumeration of all (local and non-local) blocks in a given array (which may be a section or patch of some parent array).

These functions directly support translation of data parallel languages with distributed arrays. They are also used in the runtime communication libraries, and can potentially be used directly by application programmers. Additional Adlib capabilities relevant to GA include:

- Group array descriptors, defining a restricted subset of processors over which elements of the array can be distributed. In collaboration with the originators of the Kelp system [4] (supported by NPACI), the group array field was generalized to allow arrays to be distributed over arbitrary subgroups of the original process set. It addresses problems distributed over arbitrary collections of regular meshes, each mesh distributed over a distinct subset of processors (multiblock problems).
- Comprehensive set of collective communication and arithmetic functions on distributed multidimensional arrays. They are similar to those of the Multiblock Parti library. However, Adlib provides a more general array model, and offers additional collective operations to support the Fortran 90 array-programming style. Like Parti and Chaos systems, Adlib bases all its collective operations on reusable communication schedules and provides support for irregular problems through its collective gather and scatter operations.
- Additional collective operations directly generalizing all transformational intrinsic function of Fortran 90/95 to distributed arrays. These operations include many kinds of reduction functions and some matrix operations.

The Adlib kernel is based on the portable MPI standard, and written in a portable subset of C++. The current version has been tested on IBM SP2, DEC alpha cluster, Sun UltraSparc cluster, SGI Challenge, Beowulf cluster (LINUX) and Windows NT.

**H2. Approach**

The key person at NPAC involved in this project is Bryan carpenter, a developer of Adlib. NPAC will be involved in optimizing and integrating Adlib into GA.

**H3. Technical Progress**

The proposed collaboration of PNNL and NPAC will extend the list of progress items in the original Section C:

<u>FY 1999</u>

- Support for additional distribution types: cyclic and block cyclic.
- Optimize collective communication operations on distributed arrays.

| Contractor Name | Work Package No. | Contractor Project No. | Date Prepared |
|---|---|---|---|
| Battelle Memorial Institute<br>Pacific Northwest National Laboratory | K86695 | 28008 | 03-01-98 |

FY 2000

- Add group array descriptors.
- Exploit Globus in computational grid environments based on Mirrored Array extensions of GA.

**H4. Future Accomplishments**

Extending the scope of work would provide additional value to DOE applications and improve interoperability of GA with other ACTS components.

Additional distribution types are useful in GA interfaces with parallel linear algebra packages such as ScaLAPACK and PLAPACK that employ block cyclic distribution for static load balancing. Such interfaces could be simplified or the cost of redistributing the data could be avoided if could applications adopt the same distribution type as used the linear algebra operations. The ability of creating selected global arrays on subset of processors is useful for load balancing in some applications, for example molecular dynamics.

Mirrored Array extensions of GA [5] were developed at PNNL for the wide-area-network supercomputing environments and used successfully in the SCF chemistry application running on two homogenous supercomputers [5,6]. With help of Globus and Nexus, these extensions could be offered in the computational grid environments that comprise multiple heterogenous systems. In addition, the Globus resource allocation and interfaces could greatly improve the usability of such environments to the GA applications.

In addition, using an object oriented C++ implementation of Adlib in GA could improve interoperability of GA with other ACTS libraries, many of them implemented in C++.

**H5: References**

1. John Merlin, Bryan Carpenter and Tony Hey, "shpf: a Subset High Performance Fortran compilation system", Fortran Journal, pp 2-6, March, 1996.
2. PCRC Consortium, http://www.npac.syr.edu/projects/pcrc/index.html
3. G. Zhang, B. Carpenter, G. Fox, X. Li, X. Li and Y. Wen, "PCRC-based HPF Compilation", 10th International Workshop on Languages and Compilers for Parallel Computing, 1997, To appear in Lecture Notes in Computer Science.
4. J. Merlin, S. Baden, "Implementing SPMD-HPF using SHPF and KeLP", http://www.vcpc.univie.ac.at/~jhm/shpf-kelp/
5. J. Nieplocha and R. J. Harrison, Shared Memory NUMA Programming on I-WAY, Proc. of IEEE High Perform. Distr. Comp. HPDC-5, 1996. (HPDC-5 Best Paper award)
6. J. Nieplocha, R.J. Harrison, and I.T. Foster, Explicit Management of Memory Hierarchy, in Advances in High Performance Computing, Eds. J. Kowalik, L. Grandinetti and M. Vajtersic, Kluwer Academic, 1997.