Means of Achieving Cross-Program Focus, Coordination, and Technology Transfer

Shirley Browne
University of Tennessee

Jack Dongarra
University of Tennessee and Oak Ridge National Laboratory

Geoffrey Fox
Syracuse University

Ken Kennedy
Rice University

June 7, 1995

Abstract

This paper is in response to the National Science and Technology Council (NSTC) Committee on Information and Communications R&D (CIC) call for white papers on the CIC's Strategic Implementation Plan for America in the Age of Information. We agree with the identification of strategic focus areas and the need for efficient coordination of activities. However, the Strategic Plan provides only a high level description of how the focusing and coordination will be carried out through the use of information technology. The Strategic Plan emphasizes the importance of accelerating the maturation and broad deployment of technologies, but again nowhere is it spelled out how this acceleration will be achieved. We propose the use of domain-specific modeling of complex hardware and software systems, coupled with formation of domain-specific but interoperable resource and knowledge bases for achieving the needed focus, coordination, and technology transfer. We further propose the selection of a particular strategic focus area, namely high performance/scalable systems, and a particular application category, namely high performance applications, in which to initially investigate the effectiveness of the proposed approach. If this initial focus is successful, the lessons learned and the methods and tools produced will be applicable to the other focus areas.

1 Statement of the Problem

Much success has been achieved during the first years of the federal High Performance Computing and Communications (HPCC) program, which began in 1992. Research has shown that parallel computation is effective for a large class of scientific and engineering applications. The high speed and cost effective performance of parallel computing have been demonstrated. However, parallel computing can only realize its full potential and be commercially successful if it is accepted in the real world of industry and government applications.

Many of the central parallel algorithms needed in industry have already been developed by the research community. Many artifacts produced by HPCC research programs, for example technical reports and software, are being made widely available on the World Wide Web. Attempts are being made, for example by the National HPCC Software Exchange (NHSE), to collect, organize, and catalog this information and to exercise quality control. Searchable indexes for distributed databases are being constructed, and robots and agents are being developed that can forage for and filter the vast amount of information being produced. Such tools are not sufficient, however, for answering the basic question of which software models and components are suitable on which parallel machines for solving a particular problem. Indexes allow users to do keyword or perhaps even semantic searching. Agents and filters are programmed to operate based on user profiles. However, users have little or no structure to rely on in constructing queries and profiles. Even where collection management and quality control is being attempted, there is little guidance on how software components and other artifacts should be selected, further developed or refined, and organized so as to achieve the desired broad deployment and technology transfer goals. Furthermore, the emphasis should not be on individual users, but on stakeholder communities in the broad context of interagency, interdisciplinary, and cross academic and industry cooperation.

Thus, the problem is to determine how to guide the production of, and to capture and manage the diversity of information and artifacts produced by the HPCC program, so as to create resource and knowledge bases that will be truly useful for focusing and coordinating research and development and technology transfer efforts. Merely collecting and warehousing artifacts in a haphazard manner will be ineffective, no matter how sophisticated are the search tools and agents developed to search the resulting collection. Rather, criteria must be established to guide the development and management of domain-specific resource and knowledge bases so as to meet the strategic needs of stakeholder communities. In addition, support structures consisting of effective and user friendly access and retrieval methods are needed.

2 The Domain Modeling Approach

A *problem domain* is defined in D. Shapere [8] as follows:

> In a given community, items of real-world information come to be associated as bodies of information or problem domains, having the following characteristics:
>
> 1. deep or comprehensive relationships among the items of information are suspected or postulated with respect to some class of problems, and
> 2. the problems are perceived as significant by the members of the community.

Domain analysis is defined in Arango [1] as "the process of creating models that make it possible to reason about the domain, and to predict, explain, or derive facts about the domain that are difficult to observe directly." Prediction might involve establishing the feasibility of an application, or estimating the cost of evolving an existing design. Explanation might involve clarifying the structure and function of an application.

Domain analysis attempts to reduce the complexity of a domain by imposing a coherent organization. This organization may take the form of controlled vocabularies, classification systems, entity relationship models, or formal knowledge representation and reasoning systems. Thus, both *taxonomic modeling* and *conceptual modeling* techniques are useful in deriving the organization of a domain.

In some cases, problem domains and the communities they are associated with arise naturally – for example, in numerical analysis and physics. In the case of numerical analysis, the GAMS classification scheme [2] has been developed, and the Netlib repository has been populated with software artifacts [3]. The highly structured nature of the mathematics domain makes further domain modeling largely unnecessary, at least when the users are mathematicians. The physics community has developed over the years a culture of sharing research results through preprints, and this tradition has evolved into the popular electronic preprint archives maintained at Los Alamos National Laboratory. These archives have grown to include some areas of mathematics in addition to physics. The archives do not use controlled vocabularies for indexing, but they are divided into subject areas in which participants speak a common language (e.g., quantum physics, superconductivity, functional analysis).

In other cases, concentrated effort is required to define and explicitly scope a domain. This will be especially true for problem domains that cross organizational and disciplinary boundaries. In such cases, differences between organization-specific and discipline-specific terminology and conceptual models must be reconciled. Clearly, such cross-organization and cross-disciplinary domain modeling will be one of the major hurdles for Grand Challenge and National Challenge problems.

For domains that do not arise naturally, it will be necessary to determine candidate project stakeholders and analyze their characteristics, interests, and relationships. *Candidate project stakeholders* are parties whose commitment and buy-in are critical to the success of domain projects [9]. Committed stakeholders then participate in the domain modeling activities of setting project objectives, identifying commonality and variability between systems, and identifying useful legacy and exemplar systems. The results of domain modeling are used to engineer a resource base and resource base access infrastructure that meet the needs of project stakeholders. That is, a descriptive domain model is used to produce a prescriptive resource base model that drives ongoing development and management of the resource base and its access infrastructure [9]. The access methods may be quite domain-specific – for example, expert help systems that draw on structured domain knowledge.

3 The Selected Focus Area

Of the six Strategic Focus Areas listed in the Strategic Implementation Plan, the High Performance/Scalable Systems area is the best candidate for implementing and testing the domain modeling approach for achieving cross-program focus and coordination. Considerable progress has already been achieved in this area in exploiting high performance systems to solve computationally intensive and interdisciplinary scientific and engineering problems. Some preliminary domain modeling has been carried out as a followup to the Caltech Concurrent Computation Program (C^3P) [7], and a resource base, namely the National HPCC Software Exchange, is currently under construction [4].

The biggest obstacle to the widespread adoption of high performance technologies is not the availability of the hardware, but the need to develop effective software for exploiting the hardware and achieving scalability. The function of software is to map problems onto computers. In reality, there is not a single mapping step, but rather a sequence of mapping steps between complex systems. For example, for scientific and engineering problems, the sequence starts with 1) a basic question, and proceeds through 2) a theory or approach, 3) a tractable model, 4) a numerical formulation of the model, 5) a high level software solution for a virtual computer, and 6) a low level software solution for an actual computer [7]. The different steps involve different types of experts – e.g., subject area specialists, applied mathematicians, computer scientists, and electrical engineers. Different disciplines have been built up around different complex systems and different steps of the computational process. Correspondingly different terminology is often used to describe related issues.

In general, a *complex system* may be viewed as a space, or data domain, that evolves deterministically or probabilistically with time. For example, a computer is a complex system in which the space is the collection of processing nodes, communication channels, and peripherals, and the time is quantized physical time. Two seemingly different problems can sometimes be formulated numerically with isomorphic complex systems at step 4, implying that parallelization issues for both software and hardware are essentially the same for the two problems.

By analyzing the spatial and temporal properties of high performance complex systems, Fox et.al. [7] have derived a classification of these systems and defined a partial order on them that can be used to predict the likelihood of success of mappings between them. The broad classes of complex systems are the following:

1. *Synchronous* complex systems are a direct generalization of SIMD parallel computers. They consist of more or less identical members that evolve synchronously with identical time evolution algorithms.

2. *Asynchronous* complex systems correspond to a general distributed memory MIMD parallel architecture. These systems contain disparate members that evolve dynamically over time with changing interconnect patterns and usually different evolution algorithms for each member.

3. *Loosely asynchronous* complex systems are an intermediate case between synchronous and asynchronous systems. They have generally disparate members evolving with possibly different temporal algorithms. However, the members are synchronized every so often, typically at the end of an iteration or time step.

4. *Spatially disconnected* complex systems (also known as *embarrassingly parallel*) have no connection between different members in the spatial structure of the system. A spatially disconnected system can be parallelized in a very straightforward manner, no matter what its temporal structure, by using any MIMD machine.

The majority of large scale scientific and engineering problems are synchronous or loosely synchronous. However, many real world industrial and government applications consist of asynchronous event-driven simulations. Many real world examples are a combination of the above systems, and such combinations are termed *compound* systems. In many compound systems, synchronous or loosely synchronous modules are combined into an asynchronous hierarchical structure. Examples of compound systems are the integrated design of a new aircraft and the integrated battle management problem. Many major industrial and government applications will not involve simulation, but rather the large-scale storage and access of information, followed by major processing, for example the interpretation of data from NASA's "mission to planet Earch" where the processing is large-scale image analysis.

A partial ordering has been defined on a two-dimensional complex system space, consisting of the temporal and spatial dimensions, that essentially reflects the generality of the systems in each dimension. The existence and direction of a relation between two systems in the partial order represents whether or not a mapping from one to the other can realistically be achieved. Although it has not yet been rigorously defined, an intuitive distance measure between systems in the space reflects the expected difficulty of a mapping. The goal of high level parallel software systems, such as data parallel compilers, is to shorten the distance between the numerical formulation of a problem and its expression as a parallel program.

The classification and analysis of complex systems described above demonstrates the usefulness of domain modeling, both taxonomic and conceptual, for high performance applications. Analysis of commonality and variability between different systems follows from classifying them and positioning them in the two-dimensional complex system space. The resulting model can be used to drive the development of parallel systems software and basic reusable components, and to create an access infrastructure for these resources.

4 Implementation Plan

Results of an initial survey of potential industrial applications of high performance computing systems were reported in [5]. Some industrial opportunities include the following:

- seismic analysis of new oil fields and reservoir simulation of existing fields;
- fluid flow simulations of aircraft, general vehicles, engines, airconditioners, and other turbomachinery; integration of structural analysis with the computational fluid dynamics of airflow;
- integrated design and manufacturing systems;
- design of new drugs for the pharmaceutical industry by modeling new compounds;
- simulation of electrical and gas distribution systems to optimize production and response to failures;
- support for geographic information systems including real-time analysis of data from satellite sensors;
- command and control in the military area and decision support in civilian applications.

These applications have been classified according to the taxonomy described in the preceding section, and they have been evaluated for relevance of HPCC technology and existence of relevant legacy or "dusty deck" systems.

The National HPCC Software Exchange (NHSE) has been established and has collected a large amount of information related to the Advanced Software Technology and Algorithms (ASTA) and High Performance Computing Systems (HPCS) components of the HPCC Program [4]. The information is in the form of software, technical reports and papers, and informational HTML documents. Over two hundred software packages have been collected and cataloged in the categories of scientific and engineering applications, data analysis and visualization, numerical computation, and parallel processing tools. A software submission and review process has been set up that will exercise quality control over the software collection by means of peer review.

The Northeast Parallel Architectures Center (NPAC) at Syracuse University has established the InfoMall HPCC Technology Transfer Program that is designed to bring the results of the federal HPCC and NII initiatives into the real world [6]. An example of the InfoMall approach is the formation of the Multidisciplinary Analysis and Design Industrial Consortium (MADIC), a group of large firms that share interests in the analysis and design of large-scale, complex products, such as aircraft and automobiles. HPCC technologies are expected to provide a new generation of design tools for such products that make use of large-scale, high-resolution simulations executing in distributed, heterogeneous computing environments that include workstations, vector supercomputers, and massively parallel processors. MADIC includes companies such as Grumman, General Motors, Lockheed, Boeing, United Technologies, and General Electric, and has been established as NASA's advisory group for its computational aeroscience HPCC program.

Drawing on the above experiences, we propose to facilitate cross-program focus, coordination, and technology transfer with the following implementation plan:

- Identification of problem domains for industrial and government applications of high performance scalable computing systems. Each problem domain will consist of a class of problems and associated stakeholder communities and organizations.

- Formation of industrial and government consortiums, such as MADIC, that set project objectives and collaborate on projects of mutual interest.

- Application of domain modeling to the complex systems within a domain, resulting in a descriptive domain model and a prescriptive resource base model.

- Collection and organization of the resource base for each domain within domain-specific repositories. The NHSE will provide tools and assistance for setting up and maintaining these repositories.

- Access to and interoperation between domain-specific repositories via the NHSE.

5 Summary and Conclusions

The Strategic Plan stresses the need for cross-agency focus and coordination and for cooperation between academia, government, and industry. The Plan identifies six Strategic Focus Areas for achieving these goals and describes seven broad R&D activity areas for supporting the focus areas. Research in the support software and tools area is supposed to provide "improved understanding of the design, creation, and validation of complex software systems." We argue, however, that modeling of complex software systems should precede and drive the development and acquisition of appropriate support software and tools, rather than the other way around. Furthermore, to be effective, the modeling should be carried out in the context of a particular problem domain, and tool development and acquisition should be aimed at meeting the needs of stakeholder communities associated with that domain.

According to the Strategic Plan, the goals of research in the intelligent systems and information management areas are to develop intelligent databases that locate appropriate resources for meeting users' needs and to move toward cognitive systems that use knowledge-based information management techniques. However, two decades of artificial intelligence research have failed to show that construction of knowledge-based systems can be completely automated. We argue that as much or more research and development will need to be invested in the production of taxonomic and conceptual domain models as in the production of the intelligent system and information management tools that will use these models.

References

[1] G. Arango. Domain analysis methods. In W. Schäfer, R. Prieto-Díaz, and M. Matsumoto, editors, *Software Reusability*, pages 17–49. Ellis Horwood, 1994.

[2] R. F. Boisvert, S. E. Howe, and D. K. Kahaner. GAMS: A framework for the management of scientific software. *ACM Trans. Math. Softw.*, 11(4):313–355, Dec. 1985.

[3] S. Browne, J. Dongarra, S. Green, K. Moore, T. Rowan, and R. Wade. Netlib services and resources. Technical Report UT-CS-94-222, University of Tennessee Computer Science Department, Feb. 1994.

[4] S. Browne, J. Dongarra, T. Rowan, and K. Kennedy. Management of the National HPCC Software Exchange – a virtual distributed digital library. In

Proc. Digital Libraries '95, Austin, Texas, June 1995. Preliminary version available as University of Tennessee Technical Report UT-CS-95-287.

[5] G. C. Fox. Parallel computing in industry – an initial survey. Technical Report SCCS-302b, Northeast Parallel Architectures Center, 1992.

[6] G. C. Fox, W. Furmanski, K. A. Hawick, and D. M. Leskiw. Exploration of the InfoMall concept. Technical Report SCCS 634, Northeast Parallel Architectures Center, Aug. 1994.

[7] G. C. Fox, R. D. Williams, and P. C. Messina. *Parallel Computing Works*. Morgan Kaufmann Publishers, 1994. Accessible online at `http:www.infomall.org/npac/pcw/`.

[8] D. Shapere. Scientific theories and their domains. In F. Suppe, editor, *The Structure of Scientific Theories*, pages 519–565. University of Illinois Press, 1977.

[9] M. Simos, R. Creps, C. Klingler, and L. Lavine. Organization Domain Modeling (ODM) guidebook, version 1.0. Technical Report STARS-VC-A023/011/00, Software Technology for Adaptable Reliable Systems, Mar. 1995.