UCRL-JC-132362
PREPRINT

The Tri-Lab Data Models and Format (DMF) Project: Parallel I/O and Data Exchange

L. M. Cook, C. M. Matarazzo

This paper was prepared for submittal to the
1998 Nuclear Explosives Development Conference
Las Vegas, NV
October 25-30, 1998

October 1, 1998

Lawrence Livermore National Laboratory

This is a preprint of a paper intended for publication in a journal or proceedings. Since changes may be made before publication, this preprint is made available with the understanding that it will not be cited or reproduced without the permission of the author.

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

The Tri-Lab Data Models and Format (DMF) Project: Parallel I/O and Data Exchange

Presenters: L.M Cook, C.M. Matarazzo
Lawrence Livermore National Laboratory

A central goal of the ASCI program is to push simulation and modeling for Science-based Stockpile Stewardship to unprecedented levels. ASCI applications will use extremely high-fidelity models, on the order of one billion cells, to generate terabytes of raw data. Such vast amounts of data produced by these supercomputing applications will overwhelm scientists, whose efforts to understand their results are hindered by inadequate visualization and data management tools. Much of the Scientific Data Management (SDM) effort concerns managing the large and complex data emerging from these simulation codes. One particular area for which commercial and scalable solutions do not exist is in Parallel I/O and data exchange between simulations. To address these needs, the Tri-lab Data Models and Formats effort of the SDM project is developing capabilities to enable the capturing and sharing of simulation data.

Keywords: data management, simulation, data models, data formats, data interchange, I/O, parallel I/O

Introduction

Scale and complexity of data are the major challenges when trying to capture and share ASCI simulation data. Complexity comes from the computational algorithms employed and the broad range of discrete representations – such as simple grids to grids of general polyhedra. For data to flow smoothly among physics codes and between phases of calculation and analysis, the data must be modeled in a machine- and application-neutral way. To meet this goal, the Tri-lab ASCI DMF effort is jointly developing a common data model for ASCI simulation data along with common application programming interfaces (APIs). The data model is based on fundamental topological principles that provide a comprehensive foundation for the complex data objects used in scientific applications.

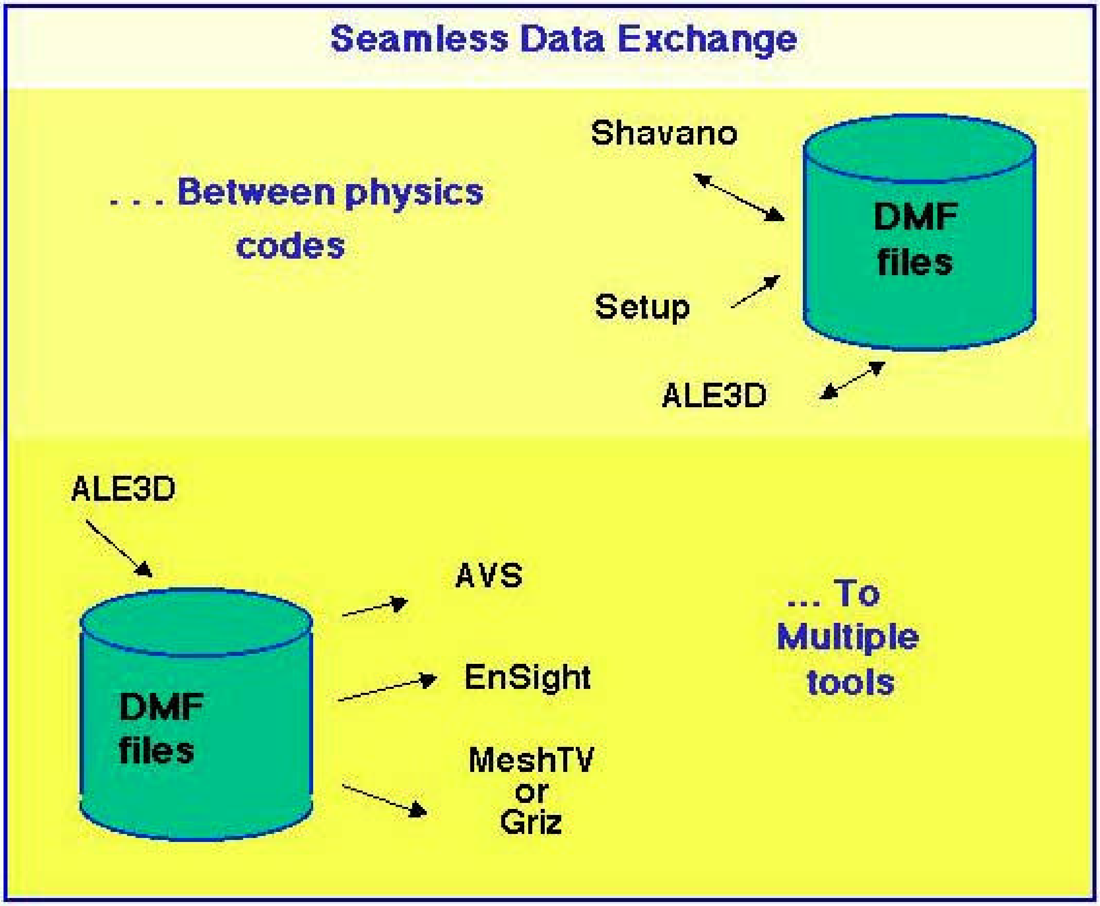
This paper and accompanying poster describe the work and strategy of the DMF effort. The goal of this effort is to provide a comprehensive solution for fast, portable serial and parallel I/O, enabling data shareability and application and tool interoperability for scientific data. It is a fundamental technology in that it provides common means to describe our scientific data, thus enabling and encouraging tools to be built at a higher level of common abstractions.

Overview of ASCI DMF Effort

Data Models and Formats (DMF) is an ASCI program collaboration between Problem-Solving Environment (PSE), Applications and external contributors. The main role of DMF is to explore and establish data models and schema at the various levels required to adequately describe data entities, and to provide the tools and libraries for permanently storing and retrieving these entities.

The DMF group aims to develop and promote technology that will allow simulation data not only to be stored and retrieved efficiently, but to be readily shared between applications and tools.

This work was performed under the auspices statement of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract No.W-7405-Eng.-48.



To accomplish this end, DMF hopes to achieve the following goals by working with developers and users at the labs, and with colleagues in industry and academia. These goals are described as follows:

- Common conceptual data models
- Common libraries and tools
- Common data schema and formats
- Common Application Programming Interfaces

Some measure of success in reaching data shareability can be achieved by making progress in any or all of these areas. Of the four, common models are perhaps the most important because they allow data encoded in completely different forms to be related by a common semantic interpretation.

Data Models

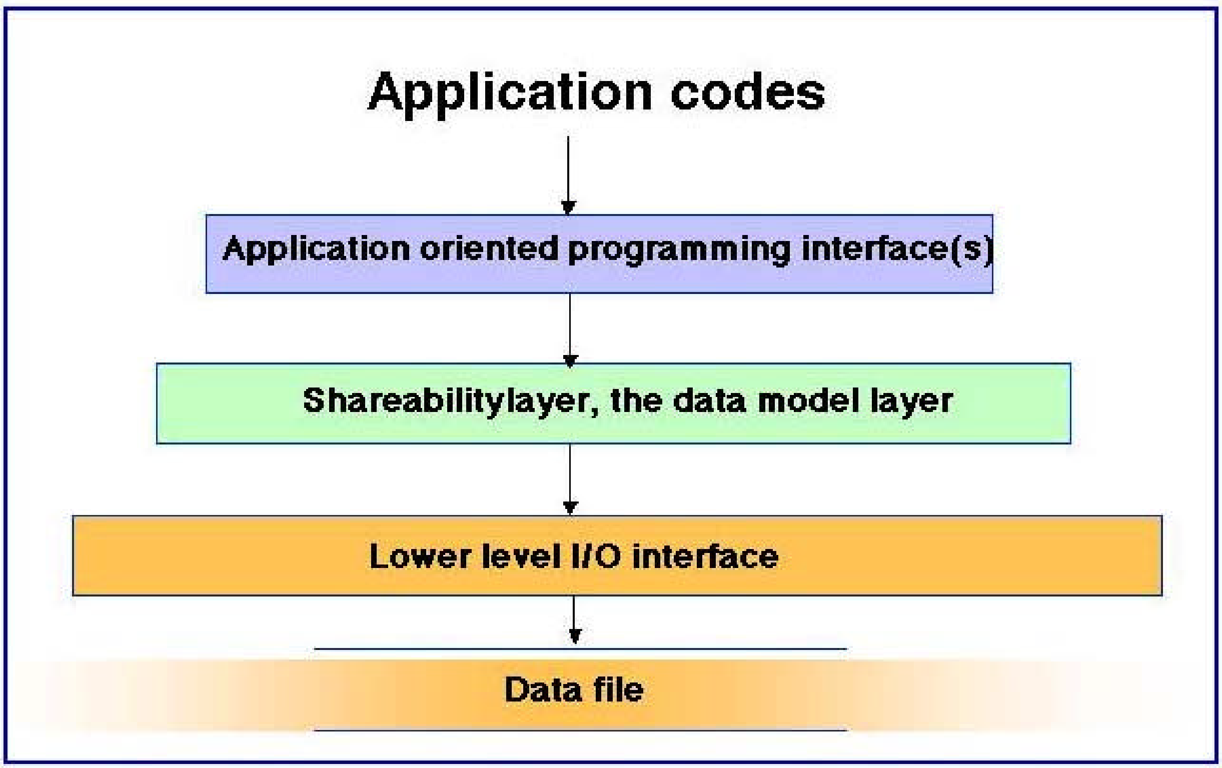
Data models are critical for the ASCI program where many applications will produce and consume one another's data, and where the computer and storage architectures will be changing rapidly. A data model is an abstract concept that allows one to describe the essential features of a class of data. Data models can be based on data structures, usage or meaning (semantics). Without a data model, data elements are just numbers (or, even worse, just bits). Within a closed set of operations or applications, data can be manipulated without much attention given to how one might model its form, meaning, or usage. However, whenever data must be shared, queried, or operated on from an outside application, a model is imperative.

The foundation for the current DMF data modeling effort is the mathematical concept of fiber bundles. A fiber bundle is an abstraction borrowed from topology that provides a good general foundation for describing computational data fields. Most scientific variables are conceptually functions or mappings of some kind. Fiber bundles are a generalization of mappings broad enough to encompass just about any conceivable mapping no matter how bizarre.

It is because the DMF work is based on a mathematical foundation that we believe we can achieve data shareability. Most descriptions of physical systems use common mathematical abstractions (fields, function spaces, differential equations, …). Data is shareable if we describe it using these abstractions; that is what is accomplished by using the fiber bundle concepts. The DMF data model describes the data in an abstract mathematical way as opposed to the ad hoc descriptions that would result from different numerical analysis or software engineering processes.

Architecture for DMF

At this time, the architecture for ASCI data management is built up from several layers of increasing specialization and complexity. Each level has a data model associated with it.



Simplified Architectural View

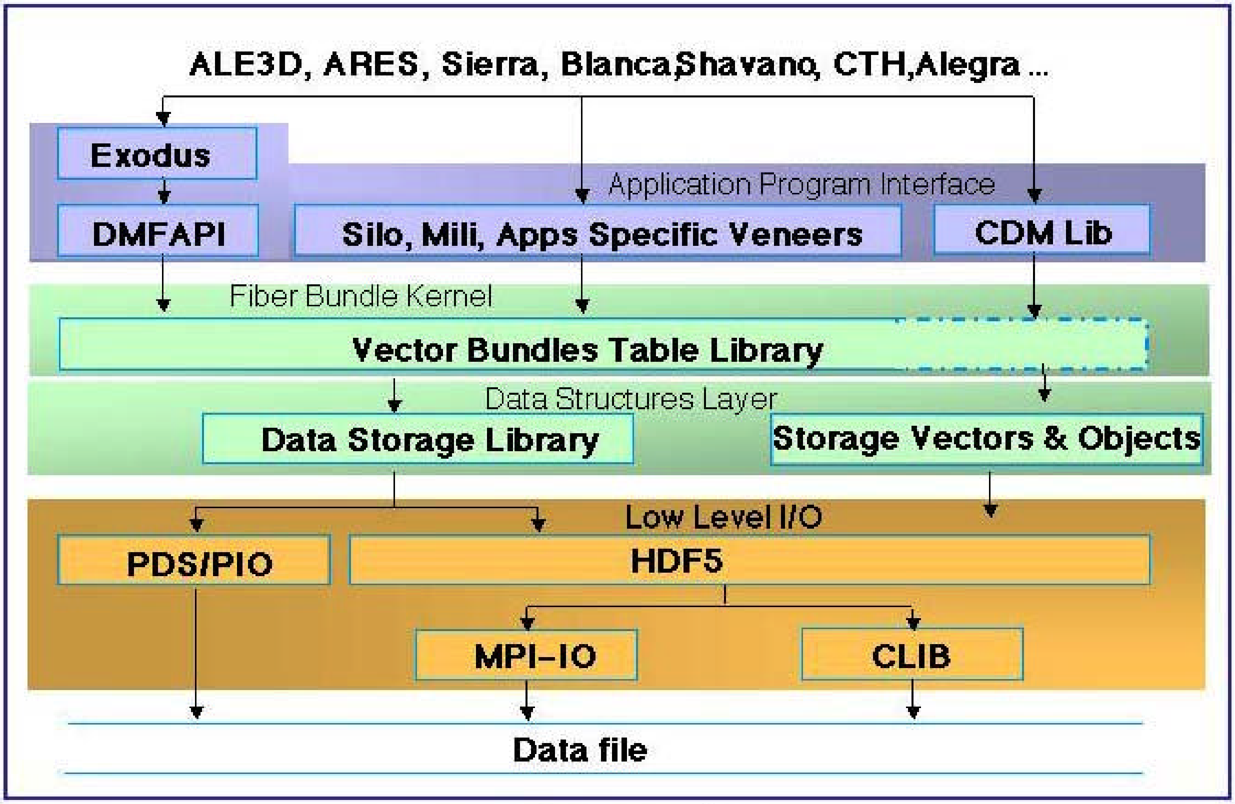
At the basic I/O level, we have adopted HDF5, the low-level I/O system from the National Center for Supercomputing Applications (NCSA). This level handles the machine independence, subsetting and data conversion capabilities. To achieve parallel I/O capabilities, HDF has been implemented on top of MPI-IO, a standard interface for parallel I/O operations. The data model at this level involves single or multiple streams of bytes written independently or collectively.

At the next data structures level, we have defined models of simple *ad hoc* structures that are common in every programming and data management application: intrinsic data types (ints, floats, strings, etc.), arrays, records, and tables. This layer provides an abstraction barrier to the underlying I/O subsystem allowing for different low-level libraries as necessary.

At the next higher level, the fiber bundle kernel layer, the fiber bundle concepts are implemented providing generalized representations. At this layer we describe data in terms of a well-defined mathematical model (cell, field, map). It is this layer that we believe will provide the highest level for data shareability. The DMF group will publish an API at this layer for tool developers to use.

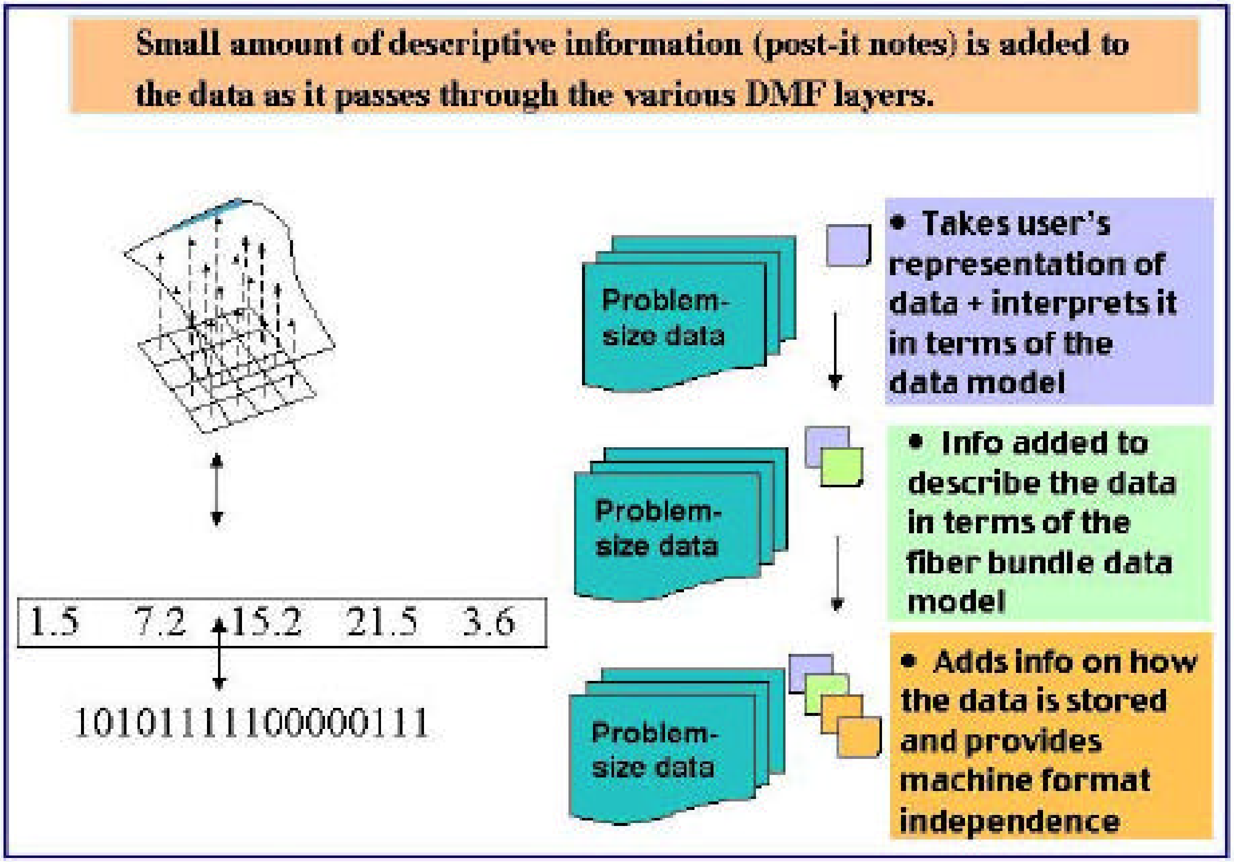
At the highest layer in the architecture are the application programming interfaces. These interfaces perform the mapping from the application problem domain to the fiber bundle data model. This layer helps to shield the application users from the complexity of the fiber bundle and uses concepts which are familiar to the specific problem domain.

The architecture has matured from the bottom up. That is, the most robust layers begin with HDF and get less mature as you approach the application interfaces. We are experimenting with several different APIs at the highest level: CDMLIB, a mesh and field object-oriented class library; DMF API, a common API for computational mechanics; and more customized and thin application-specific veneers. The DMF architecture is applicable to a wide range of scientific domains and we may find that many domain-specific APIs are developed at the highest level to achieve support for the widest-possible scientific audience.



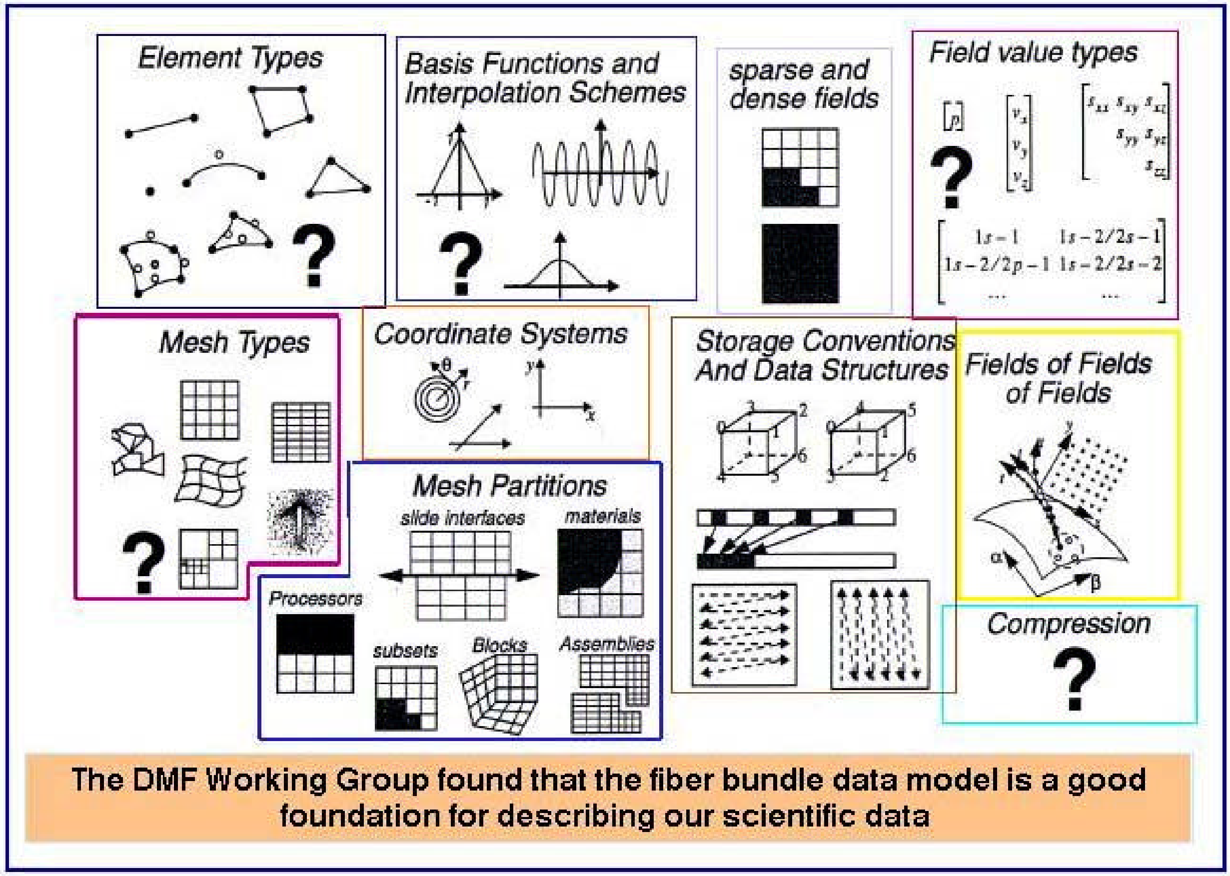
Detailed Architectural View

Given the many layers in the implementation of the DMF architecture, one could be concerned with the performance penalty that may arise. The DMF committee is well aware of the importance of fast I/O solutions for our simulation codes and we have focused on achieving the best possible performance form the outset. Although the architecture is complicated, no problem-sized (large) data is moved through the layers. At each layer, descriptive information, meta-data, is added to provide richer descriptions of the data. For example, in addition to the size and shape of an array holding the values of a field, the field itself is described with topological and algebraic-type information. The meta-data is small in size, but it enables a wealth of operations to be defined on the data.

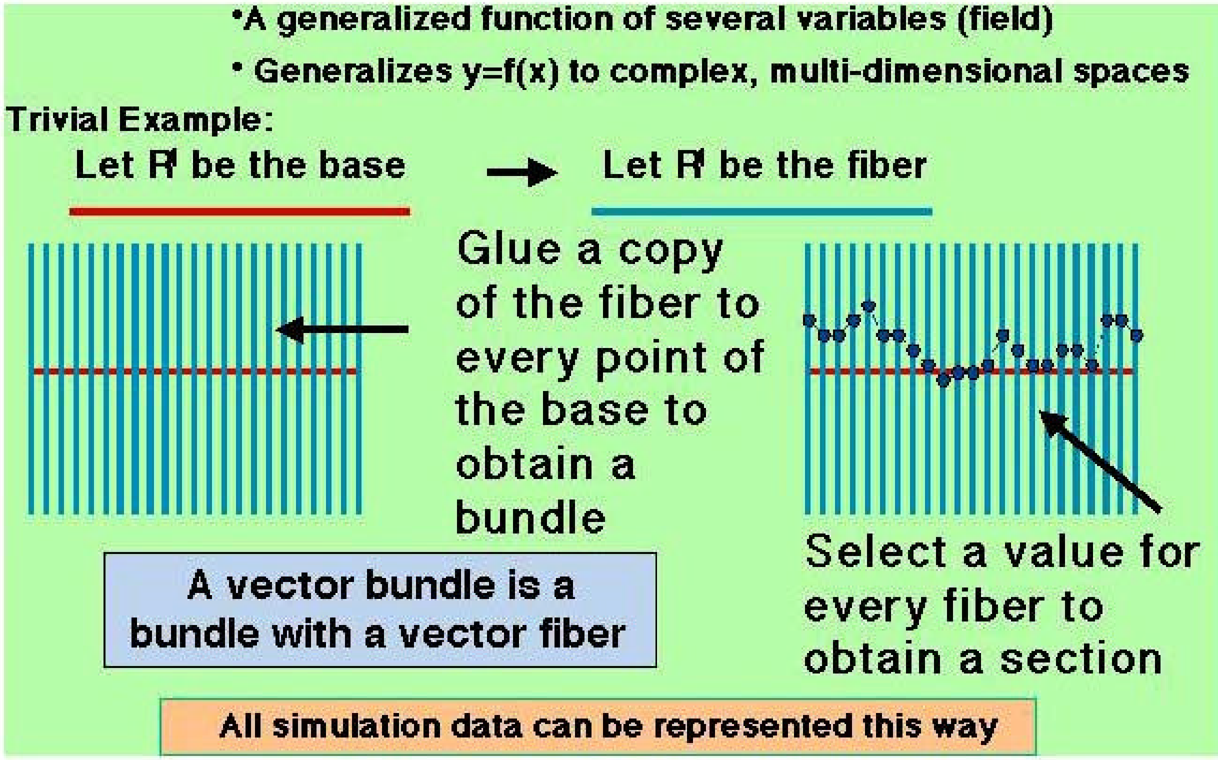


Fiber Bundles

Describing data in a general way is challenging. The ASCI DMF group has found that the fiber bundleconcepts are a good generalized way to describe the wide-range of ASCI computational science data. The fiber bundle concepts provide a common way to describe our data. The DMF working group has looked at and successfully described a wide range of data examples using the fiber bundle concepts. Data examples include element types, basis functions, sparse and dense fields, field value types, mesh types, coordinate systems, mesh partitions (slide interfaces, material interfaces, subsets, assemblies, processor partitioning), storage conventions, time histories and fields of fields.



The fiber bundle data model provides a framework for us to preserve (with the data) important attributes of fields and to operate on the data using this information. With this data model, we can describe the topology and geometry of the domain (base space), maintain local and global coordinates, describe properties of the range (fiber space) and include local and global mappings.



Trivial Example of a Fiber Bundle

The mathematics of the fiber bundle has enabled us to define objects that are useful in describing a wide-range of scientific data, to specify schema and meta-data important in capturing this description, and to define useful operations on this data.

Low-Level I/O

HDF5 from NCSA has been adopted as the primary low-level I/O subsystem for the DMF architecture. It is a file format and I/O library for high performance computing and scientific data management. HDF5 supports cross platform portability of the library and format, as well as ease of access for scientists and software developers. HDF5 is a completely new version of HDF, a format and library that have successfully supported scientific research and software development in many different user communities in high performance computing, including NASA, the Department of Energy (DOE) research laboratories, and commercial visualization software.

HDF5 can store very large numbers of very large data objects, such as multidimensional arrays, tables, and computational meshes. The scientific data objects stored within an HDF5 file are encoded in a self-describing manner and easily allow for future extensions of the file format.

HDF5 offers improved performance by giving users control over how data is stored. Among the storage options available are *data compression*, *dataset chunking*, and the ability to store an HDF5 file as a *family of files*. These options can be mixed and matched according to the needs of a particular application. For instance, it is possible to have a compressed, chunked dataset.

The HDF5 library is designed to support flexible high performance I/O on massively parallel systems. For instance, subsetting operations are optimized "under the hood" for performance. The library contains a low-level layer that can read and write to a number of different forms of "storage," such as UNIX standard I/O, MPI-IO, or internal memory. The current version of the library has been implemented using the normal UNIX I/O, as well as ROMIO, an implementation of MPI-2 from Argonne National Laboratory.

The HDF library is still under development, but early indications are that performance will be well within acceptable performance requirements for its most demanding applications. HDF5 can support all of the applications that the older version of HDF supports, but its particular strength is in supporting applications that require high performance. The first full release of

HDF5 occurred in November, 1998, but the beta version of the library has already been used successfully by several applications, including the ASCI DMF implementation and a digital library application involving very large indexes of small text phrases drawn from 20GB of text data.

Summary

Today our simulation data is more complex and massive than ever before. For example, we are using 3D, AMR, unstructured, and domain-decomposed data constructs. There is an increased need to leverage tool development efforts because tools are expensive and complex to build. Also, there has long been a desire among users to be able to easily use many different commercial tools with their data and that the same tool can operate on all of their data from different simulation codes. The DMF work is a key piece in making this a reality.

The ASCI DMF effort provides a solution for parallel I/O and data exchange. We have designed and implemented a flexible system that encapsulates common functionality into different layers and provides abstraction barriers when necessary. We are using standardized interfaces where possible- specifically MPI-IO for parallel I/O and the new HDF5 library for our lowest level I/O library and file format. In addition, the layered architecture approach allows us to implement existing I/O library APIs (such as Exodus, and Silo) at the highest layer, making use of the new infrastructure for shareability. This strategy will preserve our investment in existing tools and provide a smoother transition to the DMF system.

The DMF solution provides for important capabilities for our users and strives for fast I/O performance. Scientists will have a parallel I/O solution and can create data that is portable across platforms and is not restricted, for example, by platform file size limitations. Users can specify rich data descriptions in a compact way, thereby allowing many different views of the same raw data. For example, for the same arrays of data, one could have the following views: domain decomposition, assembly-part description, and element description. Because of the ability to create detailed data descriptions and data subsetting capabilities, the user can write and read only the data of interest. An analyst can decide to write the pressure field for all nodes at a specific time frequency and the stress field on only the nodes on the interface of two materials at a different time frequency. These sophisticated subsetting capabilities provide important storage efficiencies and give the user a mechanism to focus on the data of interest. We believe these capabilities will become increasingly more important as the size of the simulation results increase.

The other major goal of the DMF effort is to provide tool interoperability for scientific data and seamless data exchange between physics codes. Shareable data will alleviate the need to convert data to use a variety of tools or with many simulation codes. Reducing multiple copies of data is extremely important with ASCI problem-size data since even one copy of the data will strain the computational and storage resources available. The DMF effort to develop a well-defined data model for ASCI simulation data and its implementation in the fiber bundle kernel is essential to achieving the desired interoperability and shareability. Tools will be implemented at the fiber bundle kernel layer and will use the information provided at this layer for the common data descriptions to provide interoperability.

A first prototype release of the DMF libraries was delivered in October 1998 and they are currently being deployed with several different applications within the Department of Energy ASCI Laboratories. Additional releases of the software are planned throughout FY99.

Additional Information

For additional information, refer to the ASCI scientific Data Management (SDM) public web-page at http://www.ca.sandia.gov/asci-sdm. You can contact the DFM executive committee members as follows:

John Ambrosiano, LANL SDM lead and DMF technical lead,ambro@lanl.gov
David Butler, Limit Point Systems, dmbutler@limitpt.com
Linnea Cook, LLNL applications area, cook13@llnl.gov
Mike Folk, HDF5 project management, mfolk@ncsa.uiuc.edu
Celeste Matarazzo, LLNL SDM lead, celestem@lllnl.gov
Mark Miller, LLNL DMF technical lead, miller86@llnl.gov

Larry Schoof, SNL DMF technical lead, laschoo@sandia.gov