Research on Numerical Simulation Metadata

Hong Chen, Fang Xia, Lei Song

***Abstract*—In many research fields of numerical simulations, programs of scientific computing often produce a large amount of simulation simulation data with complex structure and lacking for explaining information. It is a fatal bottleneck for scientists to organize and manage such large-scale simulation data. This paper takes typical numerical simulation procedure and its result data as the application backgrounds, provides the definition and classification of metadata which faces to the numerical simulation, puts forward the method to distinguish the characteristics of various of data documents. The functions of collecting metadata online and storage them automatically are realized by imbedding thread task module in the interface of access data. I.e. the metadata are stored into the database while the calculation results in each step of outputting time are sent to the file system. In addition, a scientific calculating metadata management prototype system was designed, the users can browse and inquire about the metadata through webs, and further obtain and analyze the distributed numerical simulations data, carry on the inquiries about space scope and physical quantity ranges and its visualization in which they are interested, thus offers the users the services which are top-graded, simple to use and with data as its core. Simultion results can be look over and trace conveniently and real-timely by the researchers in this field, and the analysis and assessment ability to the calculation result can be improved efficiently.**

***Index Terms* — scientific-computing, numerical-simulation, metadata, data management**

I. Introduction

THE scientific computing is a technology by means of numerical computation to simulate complicated and continuously-altered scientific phenomena. In the research fields of hydromechanics and FEM numerical computing, programs of scientific computing do discrete sampling to data in the continuous space through series of time steps to generate scientific data with mesh structure[1]. Utilizing visualization post-processing analysis tools, researchers can analyze scientific data and correctly evaluate and forecast the simulating effects of numerical computing to physical phenomena.

Along with the improvement of high-performance computers and the parallel computing, the numerical simulation is trying to do more refined simulations to more complicated phenomena. The precision of simulation depends on the spacing of discrete points. In large-scale scientific computing, number of selected discrete points is so huge that the numerical simulating process needs great computing capacities, and number of simulation data-output files and size of each file have increased on a large scale. In the large-scaled scientific calculating data, not only the file is numerous, the single data file is large, but also the inner structure of the data file is complex. When the original calculating result data is stored, the relative explaining information about these data, i.e. metadata must be collected and stored at the same time.

Metadata can help the field researching persons to effectively organize, store, inquiry, inter visit and analyze the large amount of data and information in millions of bets scale, improve the ability of effective storage, index and the data moving of the system, so that to solve the question of managing and sharing the data. But being different from the metadata in other application field, because the numerical simulation produces many large and complex data set, its metadata collection work is also very comprehensive, collect the metadata by hands is easy to make mistakes and in low efficiency. In process of numerical calculations, the characteristics of various datasets are extracted real-timely and automatically with output of simulation results, and metadata are collected for numerical simulations. It becomes an important and difficult program to manage simulation results with scientific computing metadata.

Manuscript received March 06, 2011; revised March 24, 2011. This work was supported in part by CAEP Science & Technology Fund (2009A09005) .

Hong Chen, Fang Xia, Lei Song are all with the Institute of Applied Physics & Computational Mathematics, Beijing, China, 100094 (TEL: 0086-13910773233, Email: chenhong@iapcm.ac.cn; 0086-13718558589, xiafang@iapcm.ac.cn; 0086 TEL: 0086 13910773233, Email: chenhong@iapcm.ac.cn;13601375829, songlei@iapcm.ac.cn).

II. Numerical Simulation Metadata

It is impossible for a large-scale scientific computing without parallel I/O capacity offered by the high-performance file system; however, the file system is short of flexible data organization and querying capacity. It is because that we can only find layer directory structure and filenames in the file system that cannot provide meta-data, especially the meta-data of data contents. On the other hand, the database management system (DBMS) provides a convenient and advanced interface as well as data querying functions. It is transplantable, expandable, usable and maintainable. But it cannot meet the performance demands of large-scale scientific applications running in super computers[6]. Adopting meta-data technology, we have designed and developed a scientific data management system to integrate high-performance file system and database management system. We use high-performance file system to save original simulating data-set; meanwhile, we use commercial database management system to save meta-data relevant to applications. In this way the capacity of saving, searching and moving data in large-scale scientific computing is improved, and problems of data management and data share are solved.

ISBN: 978-988-19251-4-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

A. Basic Characteristics of Simulation data

The mesh is the outcome of discretely decomposing the continuous space. A certain continuous domain in a plane, a surface or a 3-D space, according to a certain rule, can be decomposed into a non-intersectant polygon set or a polyhedron set, and in this way the mesh is formed. The decomposed polygon or polyhedron structure is called the mesh unit. By scientific computing, a set of discrete data obtained in the center, in vertexes or on the surface of a mesh unit form a simulation data file. By a series of time steps a numerical computing program generates a series of simulation data files.

Usually simulation data are composed of mesh structure data and physical properties data. Furthermore mesh structure data include geometrical data and topological data. Therefore a mesh has four basic characteristics: geometrical property, topological structure, physical property and time-varying property.

Geometrical property: Discrete sampling points in the continuous space correspond to nodes (or vertexes) of the mesh. The geometrical data determine the positions of the corresponding mesh in the computing space, namely, the coordinates are used to identify the positions of the mesh in the computing space. Dimension of the geometrical space of the mesh is optional. When a mesh is visualized in a geometrical space, the 1-D mesh is connecting line-segments, the 2-D mesh is connecting polygons, and the 3-D mesh is connecting polyhedrons.

Topological structure: The mesh topological data are used to depict the space geometric shape, namely, the topological structure formed by several mesh units constructed by a certain connection order. They have types of points, line-segments, triangles, polygons, tetrahedrons, hexahedron, pyramids and cuneate shapes. The regulation of the mesh is determined by the regulation of nodes of mesh units and the topological regulation. Different types of meshes determine if the mesh nodes coordinates and topological information are visible or invisible.

Physical property: The physical property data of the mesh show the distribution of simulated physical variables in the multi-dimension space. It can be defined in mesh nodes or at the center of the mesh unit. The physical property data vary from different numerical simulation applications. For example, in the simulating computing of hydromechanics, the variables to be computed are pressure, density, temperature and velocity. And physical variables are divided into the scalar quantity, the vector and the tensor.

Time-varying: Usually, the numerical computing saves and deals with simulation data according to each time step. For time-regulated simulation data it is not necessary to save time coordinates because the mesh is formed by regulated time interval. For time-varying data with step interval, however, it is necessary to save time coordinates along with properties. The time-varying mesh is usually used in the simulation to a dynamic process, for example, the interaction of liquid substances and solid substances can all lead to a distortion (collision of cars), or natural phenomena such as weather changes. In these circumstances it is needed to save coordinates of vertexes. If the topological structure of the mesh is altered, the corresponding topological index information also needs to be saved.

B. Definition

It is known that the literal definition of the meta-data is “data about data”. However, in different professional fields meta-data have different definitions due to the different source, application and expressing form of the data. In the scientific computing field, the meta-data are basic information to describe characteristics of original data-set in a numerical simulation. It has expanded itself from a describing and data-indexing method to one of the indispensable tools and methods including data discovery, data transformation, data management and data usage in the complete information process on networks[7].

As for simulation data, the simulation data type determines their constructive mode, and the distribution of physical variables can be on nodes or units. If describing information about mesh types and distribution types of physical variables is added, it can avoid problems of different file formats resulted from different mesh types. In addition, we have provided meta-data describing geometrical characteristics and physical distribution characteristics of simulation data space, realized the goal of using unified data format to save simulation data. The Fig.1 shows the hierarchical structure of simulation data’s meta-data.

Usually, simulation data’s core meta-data are number of physical blocks of the mesh, number of physical

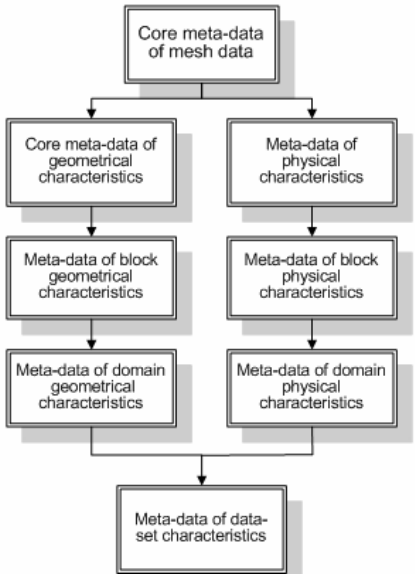

Fig. 1. Simulation data’s meta-data structure

characteristics quantities, simulation time steps, physical time, storage signs of geometrical information, and path marks. The meta-data of geometrical characteristics of blocks are mesh types, number of mesh domains, and ranges of physical coordinates of mesh nodes. For unstructured mesh there are ranges of logical coordinates of mesh nodes. The meta-data of geometrical characteristics of mesh domains are similar to the meta-data of geometrical characteristics of blocks, mainly including ranges of logical coordinates and physical coordinates of mesh nodes in the mesh domain. The meta-data are used to describe distribution characteristics (number of units or nodes) and types (scalar quantity, vector

ISBN: 978-988-19251-4-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

or tensor) of certain physical quantities. The meta-data of physical characteristics of blocks and of mesh domains are to describe value ranges within the mesh range. When the meta-data of the range of geometrical space and value ranges of physical variables are available, it is helpful for the range query of simulation data and convenient for users to find their interested data. The meta-data of data-set characteristics are mainly used in describing the file-storage characteristics of data, such as dimensions of arrays, data types, and storage positions.

C. Classification

According to the process of producing data by the numerical simulation procedure, we classify the metadata, which describes the characteristics of the data, into three types: application metadata, calculation metadata and document metadata.

(1) Application metadata is the explaining information about the numerical simulation application procedure, including the name of the application procedure, the serial number of the edition, the space dimension of the solving problem, the category of the physical model, the serial number of the subject, the designers of the physical model, the designers of the mathematical model, the designers of the numerical simulation procedure, the key words, the security and the describing information about the application procedure etc.

(2) Calculation metadata is the explaining information about the working environment and conditions of the numerical simulation procedure, including the following contents of the application procedure: the host computer, the account number, the user's name, the starting and ending time of the procedure, the number of CPU used in the operation of the procedure, net scale, the staring and ending time of the output data, input parameter file, catalogue of the output data file, the size of the result file, the key operation parameter, the describing information about the operation, and so on.

(3) Document metadata is the explaining information about the result data documents, including the field data information (such as the coordinate system, the name of the physical quantity, the unit of the physical quantity, the index, category, time information, etc.) and the storing information about the data (such as the type of the data, the dimension, the data space, compression, byte order and so on). In addition, we defined the maximum and minimum metadata (such as the coordinate range and the physical range) of the stored data, convenient to index and look for the net data rapidly by using space data indexing technology in the future.

The numerical simulation data has obvious producing characteristics and consuming characteristics, the application and calculation metadata show the producing characteristics of the data, I.e. the relevant information about data producing and origin, they are helpful for the studying personnel to understand the real meaning of the data. In our concrete realization, we take this information as the outer feature of the calculation result, and keep it in an overall metadata file. Document metadata shows the consuming characteristic of the data. I.e. the relevant information about the storing method and operation method of the data what is concerned by the data user. Generally, such information is taken as the attributive characteristic of all kinds of the numerical simulation data entities, and is kept in each calculating result file, enabling the calculating result file has self-describing characteristic.

D. Automatic extraction of meta-data

The working quantity of extracting meta-data is very huge in numerical simulations because large-scale data-sets are generated during computing. Furthermore, under the need of real time storage of numerical simulation programs, some meta-data to describe contents of original data, ranges of physical variables in the certain mesh domain, for example, must be extracted at real time during the process of program running. The solution to this problem is to use a special meta-data definition interface after the original data output.

The numerical simulation data-sets are organized by layers, and each layer has its own special meta-data. As shown in Fig.3, API is defined by the meta-data of geometrical data and physical data. In the preparing stage of outputting computing data-sets, we must configure meta-data on the model layer, meta-data of geometrical data and meta-data of physical data. After that, when writing numerical simulation data-sets to the file system, the corresponding characteristics of data-sets are extracted. The meta-data corresponding to several original data files are automatically organized and stored in a database table. Thus users can do queries, accesses and shares of the data-sets in the file system by SQL sentences through meta-data stored in the database.

Because scientific data are object entities composed by many data-sets that save space information and physical information respectively, all types of data-sets have common characteristics. For example, for space data-sets which belong to a same physical block, names of physical blocks and mesh types are their common characteristics. On the other hand, each data-set saving original data has its own particular characteristics; for example, they might be storage forms (order and dimension of arrays), data types, or the byte-sorting information in a file. By the way, we also define meta-data of maximum and minimum data values (coordinates range and physical value range, for instance) in order to do index and fast search to the simulation data in the future, using space data index technology. Therefore, the definition of meta-data that describe characteristics of scientific data contents can not only support the self-description of scientific data, but also realize the efficient data access in the object-oriented mode.

Thus users can write and exact data using the I/O property of parallel files, and they do not need to care about I/O details of the real execute file. Through meta-data physicians can efficiently select their interested data to do analyses.

III. SCIENTIFIC DATA MANAGEMENT

In order to support the data transplant and sharing, we have adopted the object-oriented method to build up a multi-layer data model, and to develop a data access interface supporting

ISBN: 978-988-19251-4-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

high-level applications (including scientific computing, visualization or other forms data analyses).

A. Simulation data Model

Considering the above-mentioned characteristics of numerical computing data and because of the requests of data scale, complexity and access type, the usual relational model cannot provide good functional supports since it is not suitable for the structure of multi-dimension, irregularity and multi-layer (the structure however is often used by scientific data-sets), and it is also not suitable for the access type relevant to computation requests. As shown in Fig.2, according to the principle of dividing simulation data units, we have brought forward a multi-layer data model. This model can organize multi-dimension and multi- variable data field, and support the description, storage and operation to large scale scientific data. In this way the complexity of data operation and storage can be obviated, problems of frequent and complicated data-format transformation among different numerical computing applications can be solved, so that the construction capacity of large-scale data parallel processing and the analysis capacity are enhanced.

The hierarchical structure of scientific data is similar to the directory tree-structure of the UNIX file system. The group object "model" in the root is used to organize simulation data in the physical domain simulated by the computing program. The simulation data include mesh space geometrical data and physical data. These data are organized by the "geometrical information" and the "physical-variable information" of the group object, and are managed on each layer according to the mesh dividing principle of "physical block - mesh domain". Different physical blocks have different mesh types; for example, some meshes have a uniform structure, and some are unstructured mesh. As a part of the physical block mesh, the mesh domain is introduced to satisfy the need of outputting computing results of the typical parallel computing programs using domain-dividing and load-balancing techniques.

The model manages the data generated from the computing program in a directory. The group object and the data-set object are two basic data objects. The group object is a container to organize data and to set up complex data management with other data objects by means of linkage. On the other hand the data-set object keeps the original data arrays. The property object includes describing data of objects such as data names, creators and physical parameters. Data-sets save original data and describing data (i.e. meta-data). Meta-data include objects such as data size information, binary sort order, and information that is needed in reading, writing and explaining data. The property object, which belongs to meta-data too, can be added to group objects or data-set objects to give them descriptions and explanations.

B. Access Interface For Scientific Data

The access interface for high-level scientific data supports functions of saving and reading scientific data in applications. It consists of two main parts: the data-writing interface and the data-reading interface. And the Application Programming Interface (API) between original data of writing/reading simulation data (physical variables, nodes coordinates, unit-connection information etc.) and meta-data is constructed. In order to meet the need of saving and reading original data and their meta-data, every part of interface is composed of the file interface, the original data interface and the meta-data interface. Because it is necessary to manage geometrical data and physical data separately, it is also needed to define an interface for meta-data of these two types of data.

The Fig.3 shows us the flowing of a computing program to output hierarchical simulation data in normal circumstances.

First, the program determines output times by controlling parameters according to time steps. In each output time, after configuring meta-data of model layer, meta-data of geometrical data, and meta-data of physical data, the program creates a specified file that is named according to the nominating rule. Then, it is time to output geometrical information and physical- variable information of the

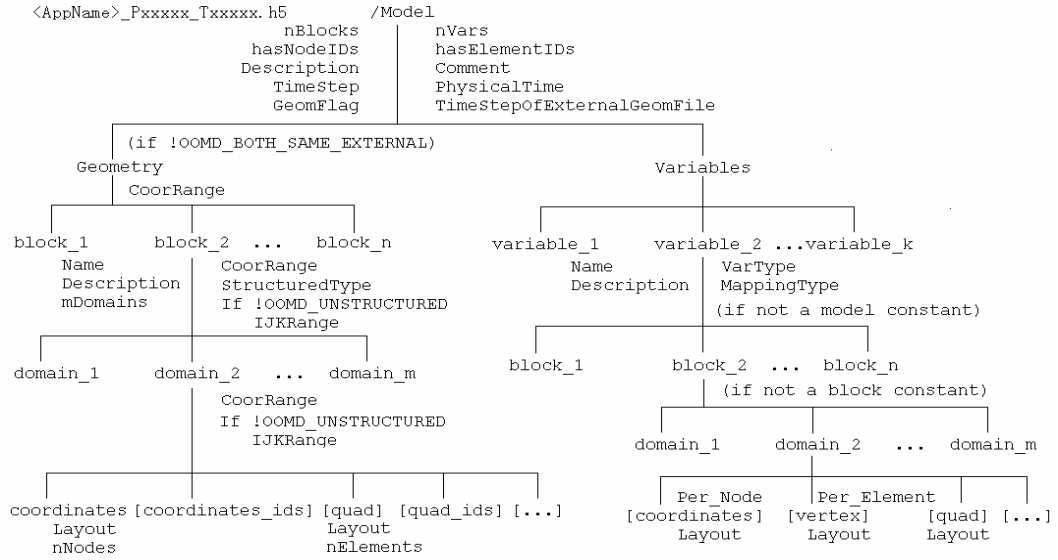

Fig. 2. Simulation data model

ISBN: 978-988-19251-4-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

corresponding mesh type. Finally, it will close the output file to end the output process. Here the geometrical information is the construction information including coordinates of mesh nodes or mesh units; it is also called the connection information of nodes. In this process one file is generated in one output time. For constant geometrical information only one output is needed, i.e. it is not necessary to output at each output time. For parallel programs, each process outputs its own file independently. To manage output files generated from a multi-process and multi-time program, it is possible to keep management information in an external abstract file. The information may be filenames, number of output process, starting and ending time step of the output, and the time step interval of the output.

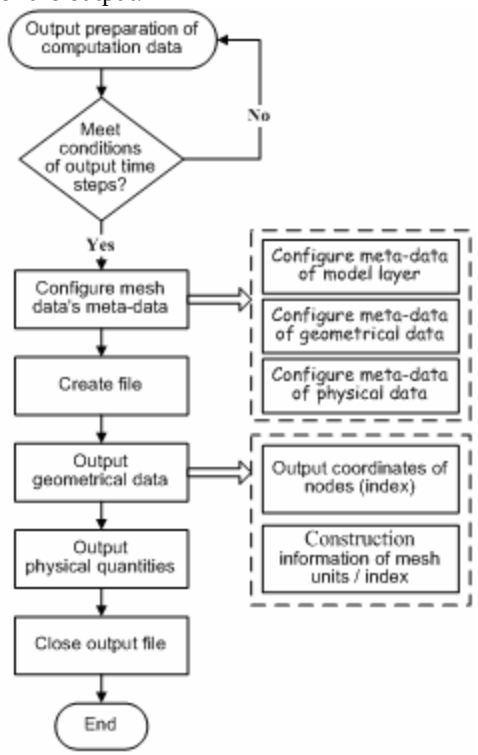

Fig. 3. Output process of simulation data

The input process of simulation data is the contrary process of the output process. It is simply as follows. First, it is to determine hich files are needed to open in accordance with the nominating rule. Then, it executes the operation of opening file. Afterwards, it reads mesh geometrical data or physical-variable data in certain physical blocks or mesh domains in the file. Finally, the file will be closed after reading process is finished.

Based on the multi-layer scientific data model, data is obtained in an I/O command-driven mode. Simple and common Application Programming Interface (API) are provided to users, by which data I/O functionality can be easily implemented in application codes by means of defining data-field objects and call their methods, instead of allocating storage directly for computing results and using low-level I/O methods to store them.

API is easily accepted by users because it has reduced modifications to the original numerical simulation program, and it is not necessary to consider output by users. On the other hand, the data organization and storage are determined by API, being independent of computing researchers. Data management personnel take over the power of computing outputs, and they can directly obtain operations to save and manage data formats as well as their results. And meta-data, which are necessary for data management, are inserted in or drawn out from the data output. The unified data access interface is provided for computing programs and visualization programs. In addition, because of defining and employment of unified and self-describing data format, scientific data can be efficiently organized and stored in different platforms. Therefore, they can be shared among different numerical simulations without transformation of data.

On the basis of HDF5 software library[4], we have built up the library of C functions for simulation data access interface, by increasing semanteme information in files to describe various types of simulation data. And a FORTRAN interface is also set up for FORTRAN77 programs. The function library supports common mesh types including uniform mesh, rectangular mesh, curve mesh and unstructured mesh.

Through real applications of numerical simulating the interaction between equivalent particles and laser[2] and the molecular dynamics[5], we have found out many its advantages. First, in real applications users only need to use a few API interfaces to do output and read/write simulation data. Comparing with directly using HDF5 library, the code-writing time can be reduced. Second, for outputting different types of simulation data, it is only need to modify several interface parameters so that codes can be used repeatedly; it would be easy to realize the modularization or templates of the code. Third, there are few output interferes to computing programs because the data organization and storage are determined by scientific data access API. Other reasons for this are that data management personnel take over the power of computing output, and they can directly obtain operations and results needed in data format storage and management. And meta-data, which are necessary for data management, are inserted in or drawn out from the data output. Finally, the unified data access interface and the unified data format are more convenient for developing common data transformation tools and analysis tools. The simulation data share is realized among computing programs or between the computing program and the visualization program. In addition, because the unified, self-described and cross-platform binary data format is adopted, the original data and simulation data's meta-data can obtain high-efficient storage and access in the high-performance file system.

C. Simulation data's meta-data management system

On the basis of PostgreSQL[8] database, we have primarily established the simulation data's meta-data management prototyping system.

Besides offering powerful data-searching functions for users, PostgreSQL provides geometrical data type and array type, and supports geometrical operations of space objects (computing if two objects will intersect, for example) and the search of multi-dimension array elements. Because the meta-data of describing space ranges of simulation data in various physical blocks and mesh domains are recorded, it is

ISBN: 978-988-19251-4-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

easy to query these data in the database and to identify positions of the goal data rapidly. The tool of meta-data is a Web interface in the client side developed by Python (see Fig. 4). By means of Web browsers users can easily access the remote simulation data's meta-data.

Users of simulation data management system basing on the meta-data can be divided into three typical types: data creator, manager and user. The data creator generates data by transparent and traditional modes; namely, he runs numerical simulation programs to generate data-sets of computing results. These data-sets are still stored in the file system. Therefore, the data creator needs an efficient method to save and maintain data. Data manager, however, is a total new role. He produces meta-data to describe numerical simulation data-sets, and stores these meta-data to the database management system. Data users, within their access powers, in accordance with the physical description habits (not computer language) and through the interface provided by meta-data management system, do combined queries to position, access, obtain and use numerical simulation data-sets more rapidly, more completely and more efficiently, and do data analyses and program reset.

IV. Conclusion

When catching and sharing simulation data generated from scientific computing programs, the scale and complexity of data are main problems. The complexity of data comes from the universality of computing and discrete representations. Aimed at smoothly transforming data among physical variables, computing and analyses, data must be modeled in a middle way between computer and applications. The simulation data model put forward by us not only considers the relation between physical system and mathematics, but also thinks over the relation between emulational universal mathematical entity and discrete representations of computer arithmetic. We must ensure the semanteme connection between them can be kept, and it can be used in the processes of data transform and management. With the API interface offered by us, computation and visualization applications can realize a unified simulation data access in the simulation data model. When we save the simulation data's meta-data into databases and build up the scientific data's meta-data management system, the technological advantages of database query can be put into full play, and the filing and searching of large-scale simulation data can be realized. Since the original data of mesh are still saved in the file system and kept in file mode, the data I/O property of programs is guaranteed. Meanwhile, our work has clearly shown that the unified storage of scientific computing data and the meta-data which describe data contents are key points of realizing simulation data share and access.

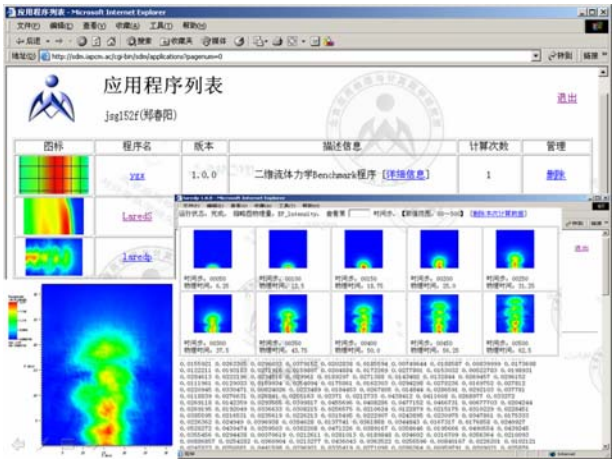

Fig. 4. Simulation data management system

References

[1] Byung S. Lee. Modeling and Querying Scientific Simulation Simulation data. *University of Vermont, Department of Computer Science, Technical Report CS-02-7, February 2002.*

[2] Chen Hong, Zhang Xia, Xia Fang, Zhang Aiqing, Song Lei. A Data Model And I/O Performance Improvement For 3-D Plasma Simulations With Particle. *Computer Engineering and Applications，Sep 2004 Vol.40.*

[3] Jerry A. Clarke, Raju Namburu. The eXtensible Data Model and Format for Interdisciplinary Computing. *Department of Defense High Performance Computing Modernization Program USERS GROUP CONFERENCE 2001.*

[4] *http://hdf.ncsa.uiuc.edu/HDF5*

[5] Xiaolin Cao, Zeyao Mo. Parallel Computation for Molecular Dynamics Simulation Based on Cell-Block Data Structures. *Computational Physics, 2004, 21 (5): 377～385.*

[6] Jim Gray, David T. Liu, . Scientific Data Management in the Coming Decade. Microsoft Research Technical Report, MSR-TR-2005-10.

[7] Wang Juanle, Sou Songcai, Xie Chuanjie. Analysis and design of metadata standard structure for geosciences data sharing.*Geography and Geo-Information Science,2005,21(1):16-18.*

[8] *http://www.postgresql.com*

[9] The ASCI Scientific Data Management, http://www.ca.sandia.gov/asci-sdm/

[10] Scientific Data Management Data Models and Formats, http://www.ca.sandia.gov/asci-sdm/

[11] Victor P. Holmes, Wilbur R. Johnson, David J. Miller. Integrating Metadata Tools with the Data Services Archive to Provide Web-based Management of Large-Scale Scientific Simulation Data. *Annual Simulation Symposium 2004: 72-79*

[12] J. No, R. Thakur, D. Kaushik, L. Freitag, A. Choudhary. Scientific Data Management System for Irregular Applications. *the Eighth International Workshop on Solving Irregular Problems in Parallel (Irregular 2001), April 2001*

[13] *http://hdfeos.gsfc.nasa.gov*

[14] Tahsin Kurc,Umit Catalyurek, Xi Zhang, Joel Saltz. A Simulation and Data Analysis System for Large Scale. *Data-Driven Oil Reservoir Simulation Studies. SC2002*

[15] Alan Sussman, Beomseok Nam. Improving Access to Multi-dimensional Self-describing Scientific Datasets.*CCGrid2003.*

[16] M. Valle, J. Favre, E. Parkinson, A. Perrig, and M. Farhat, Scientific Data Management for Visualization Implementation Experience. *Proceedings Simulation and Visualization 2005*.

[17] Xiaosong Ma, Marianne Winslett, John Norris, Xiangmin Jiao, Robert Fiedler. GODIVA: Lightweight Data Management for Scientific Visualization (draft).*the 20th International Conference on Data Engineering (ICDE 2004).*

ISBN: 978-988-19251-4-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)