Web-based Grid PSE for MD Simulations using Grid PSE Builder

Naotaka Yamamoto[1], Futoshi Shimizu[2], Mitsuo Yokokawa[1], Satoshi Sekiguchi[1], Hideo Kaburaki[3], Motoyoshi Kurokawa[4], and Ryutaro Himeno[4]

Grid Technology Research Center, AIST[1],
Tsukuba Central 2, Umezono 1-1-1 Tsukuba, Ibaraki
305-8568 Japan

Department of Materials Science and Engineering, The Ohio State University[2],
467 MacQuiggLab, 105 West Woodru® Avenue,
Columbus, OH 43210

Center for Promotion of Computational Science and Engineering,
Japan Atomic Energy Re-search Institute[3],
Tokai, Naka, Ibaraki, 319-1195, Japan.

Advanced Center for Computing and Communication,
RIKEN[4], Hirosawa 2-1, Wako, Saitama,
351-0198, JAPAN

naotaka@ni.aist.go.jp, m.yokokawa@aist.go.jp, s.sekiguchi@aist.go.jp,
shimizu@koma.jaeri.go.jp, kaburaki@sugar.tokai.jaeri.go.jp, motoyosi@riken.jp,
himeno@postman.riken.go.jp

Abstract

In large-scale scientific simulations, we sometimes have access to supercomputers and large-scale cluster systems of appropriate computer centers at remote sites as a way of obtaining large scale computer resources. In this case, some difficulties in using such resources due to differences in rules and job execution at each computer center have been found. Differences among remote sites to be accessed might contribute a delay in research and development, since it takes time for the application-side researcher to get to know a new computer environment well.

In order to provide a common environment to users, we have constructed an MD Portal or a web-based PSE for parallel MD simulations using Grid PSE Builder, a portal construction tool for PSE developers on the Grid. Users can obtain common "look and feel" interfaces to carry out their MD simulations through a Grid PSE, regardless of differences in computer resources. Multi-site computer environments are obscured from user's eyes.

The MD Portal has two components; one is an MD kernel component using the parallel MD

Stencil library, and the other is an image generator component for drawing snapshots of the results of MD simulations. Users can submit their jobs with their own parameters, check job status, and manage result files from their jobs through a web browser with secure connections. The image generator provides an animation of the results. We have confirmed the effectiveness of the Grid PSE by applying it to an MD simulation for calculating the intrinsic transformation of the vacancy dislocation loop in copper crystal.

Keyword: Grid, Grid PSE, Grid Portal, Molecular Dynamics

I. Introduction

A large amount of computer resources consisting of CPU time and memory capacity is required for large-scale scientific simulations. Researchers in computational science sometimes have access to various supercomputers and large-scale cluster systems at appropriate computer centers located in remote sites as a way to obtain large-scale computing resources. Each center usually has its own policy and rules for use of its computer facilities, and its computer environment, consisting of compilers, software libraries, and software tools, is slightly different from those of the other centers. This diversity annoys users in computational science who are not familiar with various types of computer systems, and it brings its own research delay. If a common and useful environment is built as a uniform interface from the users' point of view, users could be released from the complicated rules at various computer centers and concentrate the research itself.

One of the solutions is a problem solving environment (PSE), which provides all the computational resources needed to solve a target class of problems. This PSE is then offered to users who have insufficient knowledge of how to use various types of computer systems. Users can readily make use of the systems through the PSE for their simulations and save the valuable time for exploring their true problem, avoiding the complicated procedures that may be required to use the systems. Much effort has been expended by researchers and engineers in computer science to provide a well-considered PSE. However, there are few systems implemented on the Grid environment.

The Grid is a new environment for computation achieved by connecting multiple computer resources. It promises to supply a larger amount of computational resources, and has the potential to become a world-wide distributed computing environment. The Globus Toolkit [1], which is a *de facto* standard for building the Grid environment and provides a certain level of security among distributed resources, can be used for the construction of a Grid PSE. Though the Grid seems to be a strong tool for researchers in computational science in that it may be able to provide unlimited computer resources, it is difficult for such researchers to use a new environment efficiently because they are not computer science experts and are not familiar with the Grid. If an adequate environment, e.g. a form of web portal, for solving their problems on the Grid were provided, this environment would be useful for researchers in various fields. In order to demonstrate the effectiveness of such environments, we have developed a web-based Grid PSE building toolkit called Grid PSE Builder[2], and two web-based portals or PSEs have been constructed as experiments. First, a weather forecast grid portal for performing long-term weather forecast simulations with an S-model [3], and second, a computational fluid dynamics portal for commercial application code [4] were built.

There are some grid portal building toolkits which build web-based portals such as GridPort [5], GPDK [6], and GridSpeed [7]. These toolkits can be used to make conventional applications over Grid resources via web interfaces. However, these toolkits not only require special knowledge of web programing and security issues, but also include potential security vulnerabilities. Their authentication and authorization methods have no credential delegation mechanisms. GridSpeed provides generation of a proxy certificate with credential delegation using the MyProxy [8] toolkit. However, users have to install the Globus Toolkit on their client computer, so that MyProxy can generate a proxy certificate using the Globus Toolkit. In order to solve these problems and inconveniences, we have developed a Grid sign-on tool to provide authentication and authorization between a portal server and a user client as a part of Grid PSE Builder.

In this paper, we show the details of a web-based PSE for MD simulations constructed using Grid PSE Builder. An overview of Grid PSE Builder and the processing flow of the PSE are described in section 2. In section 3 and 4, the details of the MD Portal and an experiment involving the PSE and an MD simulation for copper crystal are presented, respectively. We provide conclusions and an overview of future plans for Grid PSE Builder in section 5.

II. Grid PSE Builder

A. *Overview of a Grid PSE*

An overview of the Grid PSE built using the Grid PSE Builder is shown in Figure 1. The hatching rectangle shows the core of the Grid PSE, called Grid PSE Server, which has a **Java Servlet engine**, a **single-sign-on session manager**, a **job dispatching and data staging manager**, and a **user accounting DB** server. A user of this Grid PSE can access the Grid PSE Server through any web browser and a Java Web Start toolkit installed on his machine. The right side components in this figure represent the back-end application providing servers on the Grid environment. The **job dispatching and data staging manager** connects back-end servers using **globus-job-run**[1] and/or **globus-url-copy** commands for dispatching a job and/or transferring a file, respectively. Furthermore, an https protocol is used between the user and the Grid PSE Server connection, then all communications are encrypted. If user's job accounting information is needed, elapsed job time, CPU time, and disk usage can be stored in the **user accounting DB**.

[1] In this paper, although we show **globus-job-run** as the command used for the execution a job, the **globus-job-submit** command is used, in the actual execution of the job.

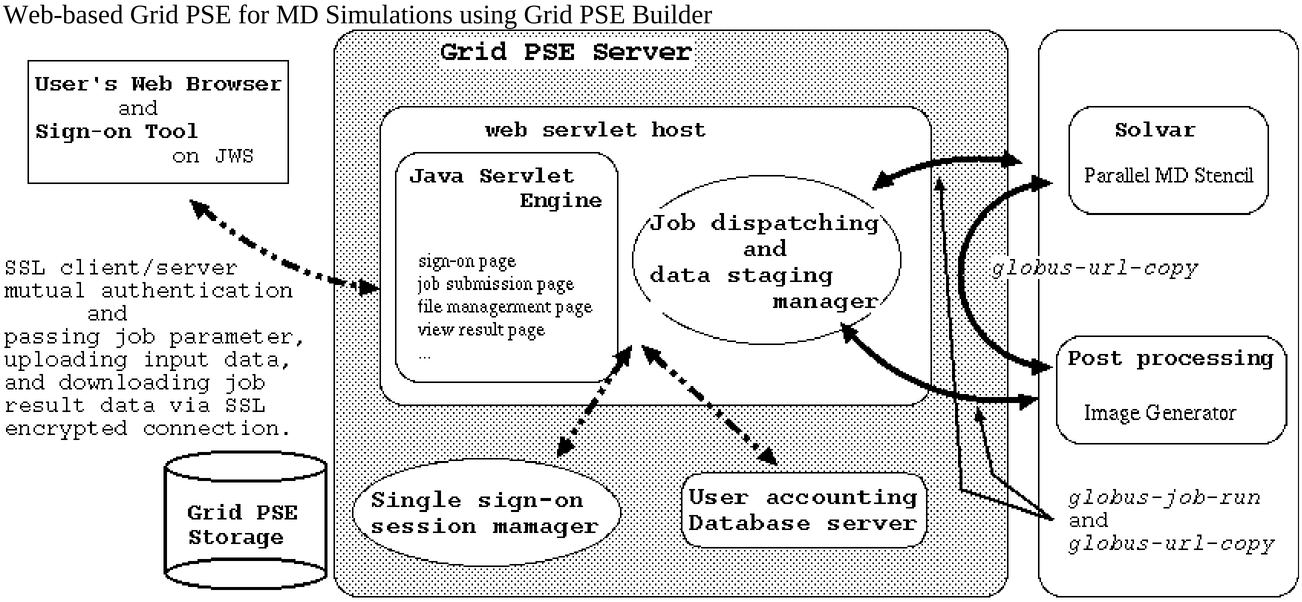

Figure 1: Job submission page for an MD simulation on a Grid PSE.

Grid PSE also provides **Grid PSE Storage** and third party file transfer. When users use a data file for input of job submission, they can select it from anywhere on job submission page and can specify anywhere they like as the output data area.

B. Registration of Applications

We strongly suggest that all applications be registered with the Grid PSE. However, when a person who wants to register an application does not have an advanced knowledge of web programming, security vulnerabilities will occur in the Grid PSE. In order to solve this problem, the Grid PSE provides automatic generation of a job submission web page to users by means of an application-described XML file. Figure 2 shows a job submission page generated by an application-described XML file (Fig. 3).

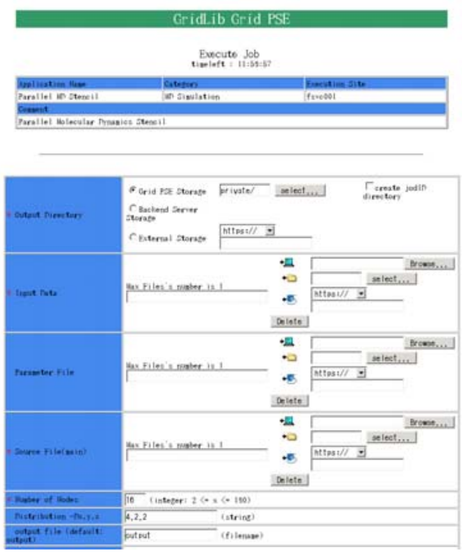

Figure 2: Job submission page for an MD simulation on a Grid PSE.

```
<application>
   <appname>date</appname>
   <appcomment>
      Parallel Molecular Dynamics Stencil
   </appcomment>
   <argspec>mdStencil.pl
     %InputFile% %SourceFile% %nprocs%
     ... and other options in here ....
   </argspec>

   <arglist>
     <args use="required">
       <title>Input Data</title>
       <file name="InputFile" max="1" />
     </args>
     <args use="required" option="-c">
       <title>Source File(main)</title>
       <file name="SourceFile" max="1" />
     </args>
     <args use="required" option="-n">
      <title>Number of Nodes</title>
      <text name="nprocs" size="3"
            maxlength="3" value="16">
        <constraint>
          <type value="integer" />
          <minInclusive value="2" />
          <maxInclusive value="160" />
        </constraint>
      </text>
     </args>
     ... and other options in here ....
```

Figure 3: A part of an application-described XML file for MD stencil job submission.

In an application-described XML file, the command executed on a server and details of the command line arguments can be described. Dependencies of parameters, and the kinds of input data (string, number, filename, etc.) can be specified in this XML file. When the argument is a numerical value, it is also possible to specify the range of this value, and then incorrect value input by users can be prevented. When a filename is specified in this XML file, the **data staging manager** invokes **globus-url-copy** to send a file to the back-end server from user's local machine, Grid PSE Storage, or other storage server.

C. Single sign-on on Grid PSE

The Globus Toolkit provides secure communication by means of a mutual authentication and credential delegation mechanism using GSSAPI in the SSL communication. These security mechanisms are called GSI, Grid Security Infrastructure. In the Grid environment based on the Globus Toolkit, an authentication is made by means of the SSL mutual authentication, and authorization is made by delegating a proxy certificate. While a proxy certificate is valid, arbitrary jobs can be performed on a server. The credential is implemented as an X.509 proxy certificate which is dynamically generated when a connection is established between a client and a server by requesting the delegation of a credential from a client. The proxy certificate is just an ordinary X.509 certificate. However, the issuer of the proxy certificates differs from that of an ordinary X.509 certificate. A user certificate is used as an issuer instead of the general Certificate Authority.

We have designed and implemented the Grid Sign-on Tool shown in Fig. 4 for providing a single sign-on mechanism between a user's client and Grid PSE Server. This tool is provided as part of Grid PSE Builder and used with a **single sign-on session manager** within the Grid PSE Server. The Grid Sign-on Tool provides a seamless authentication and credential delegation mechanism between the web environment and the Grid resources. Note that the Grid Sign-on Tool does not require a Globus environment on the client machine at all. This is also a great advantage for end users. With the Grid Sign-on Tool, the user can generate a proxy certificate in a fully PKI-compliant way, using only currently existing web-based technologies. Figure 5 shows the flow of authentication and authorization between the Grid Sign-on tool and Grid PSE Server and the details of this flow are listed below.

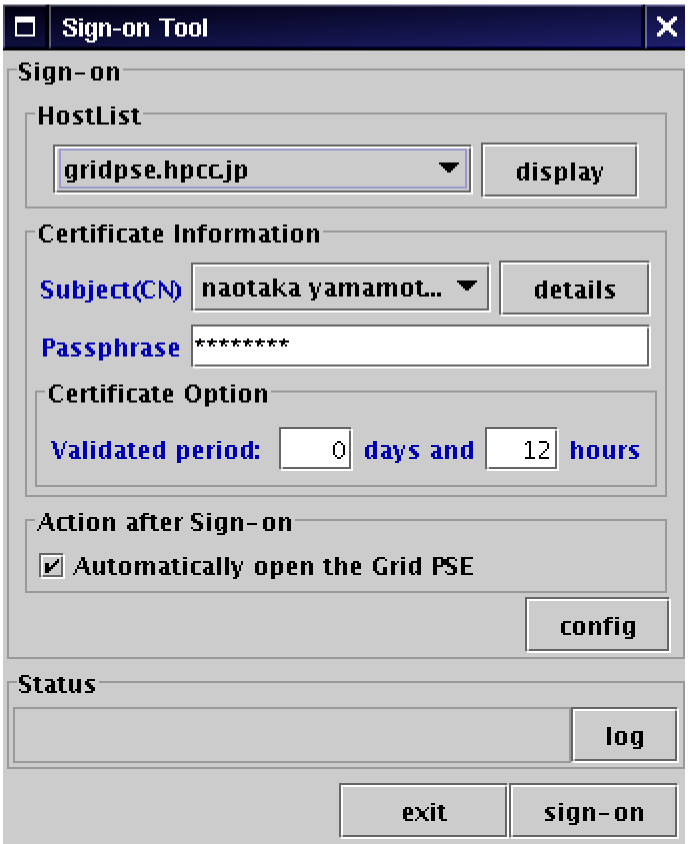

Figure 4: A view of the Grid Sign-on Tool.

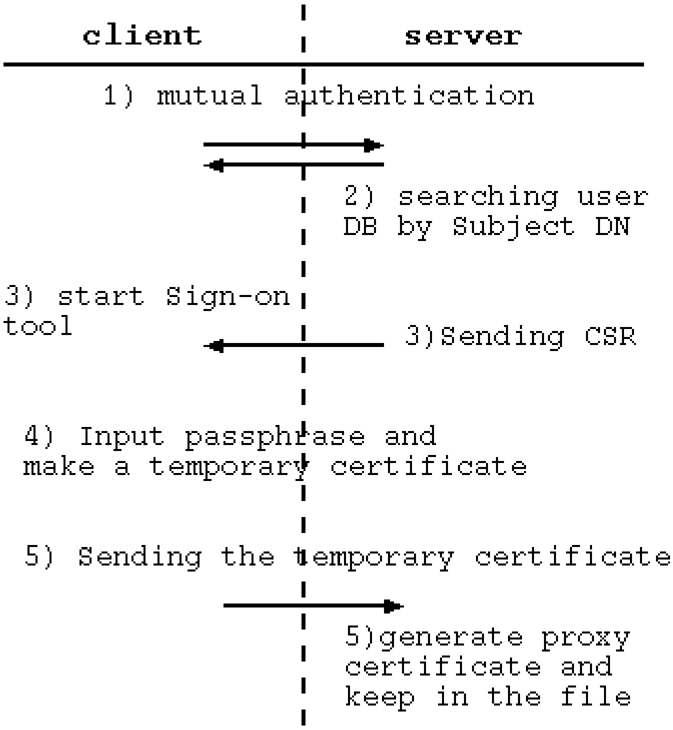

Figure 5: The mutual authentication procedure between the Grid Sign-on Tool and Grid PSE Server.

0. A PKCS#12 formatted client certificate packed with a user certificate and private key is imported onto a client web browser. On the server side, a web server program must be configured for using SSL and requiring a user certificate via the SSL protocol when a connection is requested.
1. An initialization of a connection is performed between the client web browser and the server process using the SSL protocol. Then, the web server process requests a user certificate from the client. The client certificate and the server certificate are exchanged between the client and server processes. An SSL mutual authentication is created if each signature of the certificates is verified.
2. After the SSL mutual authentication is established, whether a user can access the Grid PSE is chekced on the server by searching for a user's Subject Distinguished Name(DN) on a **user accounting DB**. If the Subject DN is found, a user has permission to use these services.
3. Then the server invokes the Grid Sign-on Tool, which is a Java Web Start application, on the client through the web browser, and generates a new temporary private/public key pair on the server. Then the server process makes a CSR from its public key and sends it to the Grid Sign-on Tool running on the client. When a user starts the Grid Sign-on Tool for the first time, the client PKCS#12 certificate is registered on the tool.
4. The Grid Sign-on Tool requests the user to input a passphrase for the client private key. The Grid Sign-on Tool decrypts the client private key and signs the CSR with a passphrase input by the user to make a temporary certificate. Inputting a passphrase will give the impression of ``being authenticated'' to the user.
5. Finally, the Grid Sign-on Tool sends the temporary certificate back to the server. For GSI authentication, the server concatenates the client certificate, the new temporary private key, and the temporary certificate into a proxy certificate file. A

generated proxy certificate is used in a job execution and/or file transfer to communicate between the user and the Grid environment.

III. A portal site for MD Simulations

A web-based portal site for MD simulations was developed using the Grid PSE Builder. It has two application components, a parallel MD stencil library and an Image Generator. A user can readily make MD simulations by describing the parameters for the job at the portal site. In this section, two application components and the structure of the MD portal site are presented.

A. *Parallel MD Stencil Library*

Recently MD simulation is very important to understand the characteristics of molecules and it requires a tremendous amount of computer resources, especially CPU time and memory capacity. Several efforts have been made to carry out the simulation fast and, as a result, parallel computing was developed to be useful for high speed simulations.

In the parallelization of the MD simulation program, the data structures kept positions and velocities of particles, also computing procedures such as calculations of forces and updates of particle positions, were partitioned and allotted to each processor. One of the key issues to be taken into account for parallel computation is the communication between processors. The communication time, consisting of the data transfer time and the start-up time, always tends to reduce the parallel efficiency. Moreover, load imbalance, which is seen in the case where the task on one processor has been done and the processor has to wait for the other processors to finish their jobs, degrades parallel efficiency. Since the method of partitioning particle data and computational procedures is closely related, a good parallel MD program should be formulated in such a way as to decrease the communication data load and frequency, and to realize a good load balance among processors.

There are two typical parallelization methods for MD simulations: the particle decomposition method and the spatial decomposition method. In the particle decomposition method, particles are, first globally numbered, then particles are partitioned into multiple sets, and finally each set is allotted to a processor. Load balance among processors can be kept uniform by allotting the same number of particles to each processor. In the spatial decomposition method, particles belonging to a partitioned space are allotted to each processor. Since inter-processor communication is restricted to local areas for short-range MD simulations, good scalability in parallel computing is obtained by the latter method.

The Parallel Molecular Dynamics Stencil (PMDS) is an assembly of subroutine programs for executing parallel short-range MD simulations [10, 11]. PMDS consists of parallel model programs which include two decomposition methods (particle- and spatial- decomposition), cutoff schemes (Verlet neighbor list and cell partitioning), time integration schemes (Velocity Verlet, Beeman, and 5-value Gear), and methods for sampling statistical ensembles (Nose-Hoover thermostat and Parrinello-Rahman constant-stress methods). PMDS is written in the C language using the MPI library for parallelization, and is designed to separate and conceal the parallel algorithms, such as inter-processor communications, so that the parallel programming for force calculation can be done in the same way as sequential programming. This makes it easy to revise and to debug the program depending on physical models employed using a familiar debugger.

PMDS can be linked with a main program to make an MD simulation and any potential can be incorporated easily into the MD application bydescribing it in the main program.

B. Image Generator

Image Generator is created using PVSLIB which is a real time drawing library. It can draw multiple particles by referencing location data for atoms which are held in snapshot files generated by the MD simulations. Three dimensional locations and coordination numbers of simulated atoms kept in a snapshot file are used for creating a visualized image file. And Image Generator outputs an image according to input parameters specified by users. It is possible to specify the size of displayed particles, a viewpoint in the three dimensional space, an angle for the image, and the colors of particles which correspond to the coordination number of an atom.

Image Generator can be implemented on any application server separate from the MD simulation. Since snapshot files are obtained one by one in the course of MD simulations, Image Generator can make visualized image files in jpeg format at any time, corresponding to the snapshot files.

C. A portal site for MD Simulations

We constructed a Grid PSE for our MD simulation using Grid PSE Builder. It includes two application components, Parallel MD Stencil (PMDS) and Image Generator. An application-described XML file for PMDS (Fig. 3) has 20 input parameter fields with a file upload dialog for a main program source code. Therefore a compilation is performed before an execution of the MD simulation after one job submission (Fig. 2).

The job submission page for Image Generator is also provided by an XML application to describe command line arguments. If Image Generator is selected at the job selection stage, a web page appears prompting parameters required for Image Generator. Users can select one job ID from a list box and input a file prefix word which is usually specified in the MD simulations. Image Generator can make image files in jpeg format by searching snapshot files on the MD simulation server. Then, these files are directly transmitted to Image Generator from the server. All snapshot files can be used for making a sequence of images or an animation after the end of the MD simulation. Moreover, an animation can be created at any time during the simulation by taking the snapshot files which are obtained at the time Image Generator is invoked, because the image generator is implemented independent of the MD kernel component and can be run on a different back-end server.

IV. Simulation Results

A. Grid Environment

Two Linux servers, AIST Gfarm Cluster I [11] and a visualization server are utilized as the back-end servers for each of the two applications, PMDS and Image generator, respectively. AIST Gfarm Cluster I is part of a proven distributed file system called Gfarm which is a reference implementation of the Grid Datafarm architecture. The specification of the Grid Datafarm was proposed for and discussed at the Global Grid Forum. Though PMDS can be carried out on multiple Grid computing resources in parallel with MPICH-G library or something similar, only one cluster system is used in the experiment. A node of AIST Gfarm Cluster I consists of an SMP with two 2.8MHz Xeon processors and RAID disk system with 4 disk drives. The visualization server is a workstation located at RIKEN, a research institute for science and technology in Japan. The portal server is located on a high-speed metropolitan area network called Tsukuba-WAN, of which the network bandwidth is 10Gbps, and is connected to a NAS server with 1TByte disk capacity to handle the large size of result files. The servers are located at the different places within a 50km distance. They are connected to the Internet and form a single virtual organization based on the Grid. A user can perform an MD simulation and visualization via a web browser without being conscious of the internal structure of the Grid PSE.

B. MD simulation of the extension process of a Frank Loop

It is known that metal material will be hardened due to irradiation by particles such as neutrons. Plastic deformation of a metal crystal is explained by a movement based on a dislocation which is a line defect. When metal is irradiated, the atomic vacancy and the interstitial atom will be generated inside a crystal. These point defects will gather and cluster in the shape of a hexagon or triangle, forming a fixed dislocation called a Frank loop. The fixed dislocation might become an obstacle to the moving dislocation. To elucidate the mechanism of the irradiation hardening phenomenon in metal, we carried out an MD simulation of the extension process of the Frank loop using the Grid PSE. The parameters for the simulations are specified on the web page.

As an initial condition, atomic vacancy has been arranged in the shape of a hexagon cluster, and then into the perfect copper crystal (a face-centered cubic lattice). Then the system is relaxed using the embedding atomic method (EAM) potential for describing the interaction between atoms. The results of an MD simulation shown in the coordination number representation, in which the atoms for the coordination number of 12 (the value of a perfect crystal) are omitted, is shown in the figure on the right in Fig. 6. Here, a structure similar to the stacking fault tetrahedron is seen. The figure on the left in Fig. 6 is the result for the Lennard-Jones potential. It was found that the self-organization process is sensitive to the model of inter-atomic interaction.

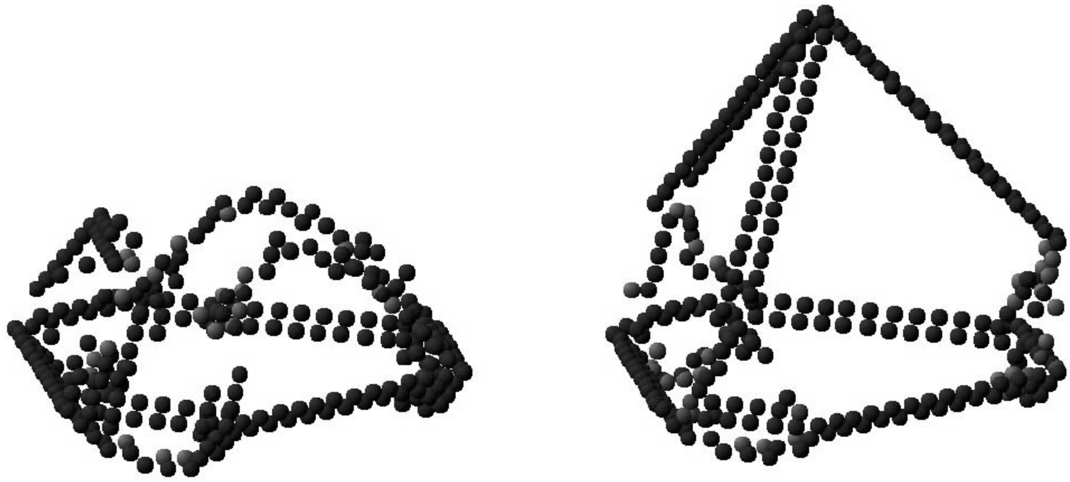

Figure 6: Visualized images. Left: using Lennard-Jones(LJ) potential, Right: using Embedded Atom Method(EAM) potential

With this application, we can study various models of metal systems by uploading a program which describes the atomic interaction in the system. We also can easily compare these results utilizing visualized images via a web browser on a PC.

5. Conclusions

Some advantages of Grid PSE have been identified through the experiment in building a portal site and applying it to an actual MD simulation.

1. Since a portal site can be accessed through a web browser and the Grid Sign-on Tool, users can make MD simulations and see their results at any time and from any places without worrying abount information disclosure.
2. Compilation of the main program followed by execution of MD simulations can be performed as consecutive jobs.
3. Since any job and the resulting files are related to the Job ID on the Grid PSE, it is easy to handle result files by referring to the job ID. Then comparison of two or more job result images can also be carried out.
4. Since visualization of application jobs is performed without transmitting calculation results to Grid PSE Storage or a user client machine, the cost of file transfers is dramatically decreased.

It is clear that the Grid PSE is extremely efficient in performing research in computations. If Grid PSE Builder is used, an application provider can build a Grid PSE easily without advanced programming skills. Moreover, the Grid PSE Builder provides user friendly interfaces throughout the Internet.

Acknowledgements

We are very thankful to the members of Grid PSE Builder development team who gave us valuable comments on portal site construction. We also thank the members of the Grid Technology Research Center, AIST, for their cooperation in this work. We are also thankful to Mr. Hisashi Eguchi of the Tsukuba WAN NOC team who helped with use of NAS equipment.

References

[1] The Globus Alliance: http://www.globus.org/
[2] M. Hirano, N. Yamamoto, H. Takemiya, Y. Tanaka, S. Itoh, and S. Sekiguchi: Grid PSE Builder: A Framework for Building Web-based Distributed PSE on Grid, *Proceedings of the Seventh International Conference on High Performance Computing and Grid in Asia Pacific Region* (HPCAsia'04), pp. 34 - 41 (2004).
[3] H. Takemiya, K. Shudo, Y. Tanaka, and S. Sekiguchi: Developing a Simulation System for Atmospheric Prediction on a Grid Environment, *SIG*, Vol. 44}, pp. 23 - 33 (2003) *(in Japanese)*.
[4] N. Yamamoto: Implementation of a CFD Portal System by the Grid PSE Builder, *AIST Today*, No. 10 Autum, pp. 30 (2003).
[5] M. Thomas, S. Mock, J. Boisseau, M. Dahan, K. Muller, and D. Sutton: The gridport toolkit architecture for building grid portals. *In Proceedings of the 10th IEEE Intl. Sympo. on High Perf. Dist. Comp* (2001).
[6] J. Novotony: The grid portal development kit. *CONCURRENCY AND COMPUTATION PRACTICE & EXPERIENCE*, 14(13-15), pp. 1129 - 1144 (2002).
[7] T. Suzumura, H. Nakada, S. Matsuoka, and H. Casanova: *Gridspeed: A web-based grid portal generation server.*} http://www.gridspeed.org/.
[8] J. Novotny, S. Tuecke, and V. Welch: An Online Credential Repository for the Grid: MyProxy. *Proceedings of the Tenth International Symposium on High Performance Distributed Computing* (HPDC-10), IEEE Press, August (2001).
[9] F.Shimizu, H.Kimizuka, H.Kaburaki, and C.Arakawa: Development of Parallel Molecular Dynamics Stencil, Trans. JSCES, #20020015 (2002). *(in Japanese)*
[10] Parallel Molecular Dynamics Stencil: http://stencil.koma.jaeri.go.jp/.
[11] O. Tatebe, H. Ogawa, Y. Kodama, T. Kudoh, S. Sekiguchi, S. Matsuoka, K. Aida, T. Boku, M. Sato, Y. Morita, Y. Kitatsuji, J. Williams, J. Hicks,: The Second Trans-Pacific Grid Datafarm Testbed and Experiments for SC2003, *Proceedings of 2004 International Symposium on Applications and the Internet Workshops (SAINT 2004 Workshops)*, (2004).

Naotaka YAMAMOTO received a B.S. and an M.S. from Science University of Tokyo in 1997 and 1999 respectively. In 1999, he joined Science University of Tokyo, in Tokyo, Japan. He is now a research scientist in Grid Infraware team of Grid Technology Research Center at AIST (National Institute of Advanced Industrial Science and Technology). He is currently studying user interfaces and middlewares for the Grid technology.

Futoshi SHIMIZU received a B.E., an M.E. and a D.Eng. from University of Tsukuba in 1990, 1992 and 1996 respectively. He joined Center for Promotion of Computational Science and engineering of Japan Atomic Energy Research Institute, in Tokyo, Japan. He engaged in development of a parallelized molecular dynamics simulation code (PMDS). From 2004, he is a Visiting Scholar at Department of Materials Science and Engineering, the Ohio State University, USA. He is currently studying the atomistic simulation to research on mechanical properties of the metals.

YOKOKAWA Mitsuo received B.Sc and M.Sc degrees in Mathematics from University of Tsukuba in 1982 and 1984, respectively. He also received D.Eng. degree in Computer Engineering from University of Tsukuba in 1991. He joined the Japan Atomic Energy Research Institute in 1984 and was engaged in high performance computation in nuclear engineering. In 1995, he worked at the Science and Technology Agency of Japan to make a plan of the Earth Simulator project. He was engaged in the development of the Earth Simulator from 1996 to 2002. He has joined the National Institute of Advanced Industrial Science and Technology (AIST) in 2002 and he is Deputy Director of the Grid Technology Research Center of AIST. He was a visiting researcher of the Theory Center at Cornell University in US from 1994 to 1995. He is a member of the Information Processing Society of Japan and the Japan Society for Industrial and Applied Mathematics.

Satoshi SEKIGUCHI was born in 1959, received B.S. from Department of Information Science, Faculty of Science, the University of Tokyo in 1982, and M. SE. from University of Tsukuba in 1984 respectively. He joined Electrotechnical Laboratory, Agency of Industrial Science and Technology in 1984 to engage research in high performance and parallel computing widely from the computer architecture, compiler, numerical algorithm, performance evaluation as well as its applications. He served as the deputy director of Research Institute of Information Technology, AIST in 2001, and is currently the founding director of Grid Technology Research Center (GTRC), AIST since 2002. He is a member of IEEE, SIAM, IPSJ, and is a chair of the SIGHPC.

He also had been serving as one of the steering committee members of the Global Grid Forum (GGF) till 2003, and now a member of GGF advisory committee. Since the dawn of grid era, he has been one of technology and community leaders, who is in particular one of the PIs of the Ninf project since 1995 being developed as a reference implementation of current GridRPC GGF standard draft, the founder of the Asia Pacific Grid partnership (ApGrid), and chairing Japan Grid Consortium (JpGrid).

Hideo KABURAKI received a Doctor of Science for applied physics from Tokyo Institute of Technology(1980). Entered Japan Atomic Energy Research Institute and engaged in research projects for high temperature gas-cooled reactor and atomic laser isotope separation (1980-1993). Engaged in computational science project and worked on multiscale materials modeling and molecular dynamics simulations(1994-present). He is now Principal Scientist, Group Leader, Deputy Director Center for Computational Science and Engineering Japan Atomic Energy Research Institute.

Motoyoshi KUROKAWA received a B.E. from the University of Electro-Communications, Tokyo, in 1997. He received a M.S. in Information Science and a Ph.D. in Information Science from the Japan Advanced Institute of Science and Technology, Ishikawa, Japan, in 1999 and 2002 respectively. In 2002, he joined RIKEN (The Institute of Physical and Chemical Research), in Saitama, Japan. He is currently a technical scientist in the Advanced Center for Computing and Communication at RIKEN. He is a system administrator for RIKEN supercomputer and studies parallel and distributed systems, parallel CFD and GRID.

Ryutaro HIMENO received a B.E. and an M.E. from Kyoto University in 1977 and 1979 respectively. He received a D.Eng. from the University of Tokyo in 1988. In 1979, he joined the Central Research Laboratories of Nissan Motor Co., in Yokosuka, Japan. He researched the application of computational fluid dynamics to car's aerodynamics. From 1984 to 1986, he served as a researcher at the Institute of Space and Astronautical Science in Tokyo. In 1998, he joined RIKEN (The Institute of Physical and Chemical Research) and he is now the Director of the Advanced Center for Computing and Communication at RIKEN. He is also a Visiting Professor at the University of Tokyo and at Saitama University. He is currently studying the aerodynamics of baseball and the biomechanical simulation of the human body. He was a winner of the Nikkei Scientific Visualization Contest in 2000, and received the JSME Computational Mechanics Division Award in 1997 and the JSME Youth Engineer Award in 1988. He has also won the NICOGRAPH Paper Award in 1993, the CRAY Research Inc. Giga FLOPS Award in 1990, and other awards.