Journal of Applied Mechanics Vol.10, pp.573-579 (August 2007) JSCE

Improvement of IES Wrapper for Plug-in of Advanced Simulation Programs

Sivasithamparam Nallathamby*, Hori Muneo** and Yoshiaki Ariga***

* Master student. Earthquake Research Institute. Tokyo University (1-1-1 Yayoi, Bunkyo, Tokyo, 113-0032)
** Professor. Earthquake Research Institute. Tokyo University (1-1-1 Yayoi, Bunkyo, Tokyo, 113-0032)
*** Senior Researcher. Chigasaki Research Institute, J-Power (1 Chigasaki, Chigasaki, Kanagawa, 253-0041)

A new IES wrapper is constructed for an advanced simulation program, UNIVERSE, for dam seismic response analysis. Modifying the program structure of the member functions and finding an alternative of generating visualization data are the two key strategies in constructing the wrapper. The performance characteristics of the new wrapper such as scalability and modularity are compared with those of the previous one. The unified visualization of a virtual city in which a large arch dam is located is made, to demonstrate the performance of the new wrapper which handle huge amount of UNIVERSE results.

Key Words: *wrapper, IES (integrated earthquake simulation), unified visualization*

1. Introduction

Development of a simulation-based earthquake disaster prediction system is a challenge of earthquake engineering of next generation. Modern society is rapidly changing and it is not sufficient to prevent reoccurrence of past earthquake disasters. The simulation-based disaster prediction system is aimed at providing possible disasters which have not been experienced. The system will be a huge computer system since its target is to simulate all three phases of earthquake hazard and disasters, namely, simulation of an earthquake, structure responses, and actions against earthquake disasters.

The authors have been developing integrated earthquake simulation (IES) [1), 2), 3), 4)] as a candidate of the simulation-based earthquake disaster prediction system. The current concern of developing IES is the improvement of simulation for seismic structure responses. This is because the number of building and structure types is huge and each type has distinct advanced analysis method which computes non-linear seismic responses. These methods need to be plugged into IES.

In order to include various kinds of buildings and structures, IES makes use of a wrapper. For a given simulation program (SP) of a particular building or structure, a wrapper plays the role of an interpreter so that SP is linked to IES without changing SP and IES; see Fig. 1. Wrappers have been built for several non-linear SP's of RC or steel piers, ground molds, and wooden houses. These wrappers, however, are at a primitive level and cannot be applied to more advanced SP's which require larger numerical computation since they are used in actual design of structures and buildings.

This paper reports the improvement of wrapper so that an advanced SP will be plugged into IES. UNIVERSE[6), 7), 8)], which has been used for seismic design of dam structures, is used as an example of such an advanced SP, and a wrapper for UNIVERSE is constructed. The contents of the paper are as follows: In Section 2, the structure of IES is briefly explained, focusing the role of wrapper. In Section 3, strategies of improving a wrapper for UNIVERSE are discussed. These strategies will be of primary importance in constructing a wrapper for

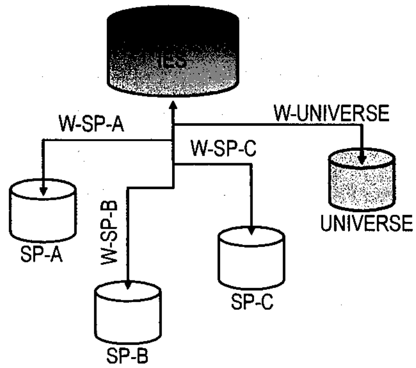

Fig.1. Schematic view of IES; connection of SP through wrapper; W stands for a wrapper.

other advanced SP's. Finally, in Section 4, the construction of the wrapper for UNIVERSE is explained. Comparison is made for the new wrapper and previous ones to verify the strategy of improving wrapper.

Before starting this paper, we should emphasize that a wrapper used in this paper is slightly different from a mediator which is usually used in software engineering. A mediator is used as an interface for classes or member functions which are constructed in a complicated program system; users can easily use these complicated system with the help of the mediators. The wrapper in this paper is for a kernel of the system and plays a role of an interpreter for a specific simulation programs.

2. Brief Summary of IES

IES is a simulation system for the following three phases of earthquake: 1) generation and propagation of a seismic wave and strong ground motion; 2) response and damage of structures subjected to strong ground motion; and 3) human and social actions against earthquake disasters such as crisis management or recovery (Fig. 2 depicts the IES system). While each simulation has its own purpose, the three simulations are related to each other.

IES consists of GIS[1] (Geographical Information System) and the three groups of numerical simulation. GIS provides data to construct computer models which include underground structures. The second phase, the simulation of structural responses, uses numerous SP's, with suitable computer models being provided. IES has several visualization tools with which the results of SP's are visualized for each structure or for the entire city.

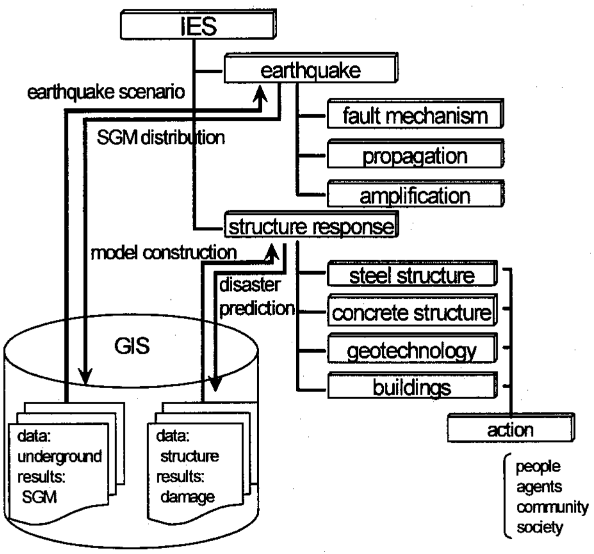

Fig. 2. Schematic view of IES.

As mentioned, a wrapper plays the role of an interpreter between IES and SP. The structure of a wrapper is simple. It has only one data for a target structure (target), and seven member functions, as follows:

1. a target is found by FindStructure(), together with a site of SGM (Strong Ground Motion) being given by SetSite();
2. structural and SGM input data of SP are respectively provided by PutStructureData() and PutSGMData()
3. execution is made by ExecuteSP() and results of SP are obtained by GetResults();
4. the results are converted to data for visualization of IES by VisualizeResult().

Figure 3 provides a perspective of the wrapper's role. For simplicity, GIS-Data and SGM-Data are converted to SP-Data, and SP-Result to V-Data, where V stands for visualization.

3. Strategy of Improving Wrapper

In developing a large computer system, it is generally expected that system follows the KISS (Keep It Simple, Straightforward) principle, so that developers or users have easier access to the system. The major advantages of following the KISS principle are

1. to keep the system largely free from defects;
2. to easily and quickly detect defects;
3. to simplify the logic;
4. to have a well-structured code.

As the system becomes larger, these four advantages become more significant.

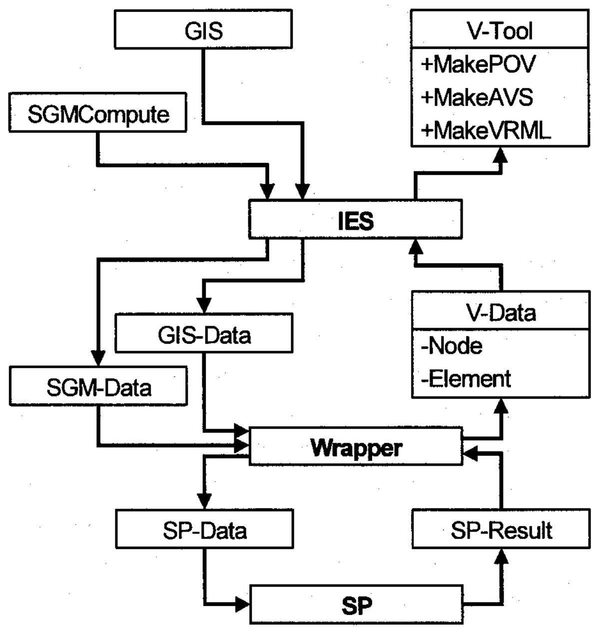

Fig.3. Roles of mediator as interpreter between IES and SP converting various data.

While introducing wrapper is a logical solution in order to plug various SP's into IES, the structure of wrapper should be improved by critically reviewing the data structure and functions of the wrapper. The first two advantages of the KISS principle are not relevant to improving wrapper; introducing wrapper itself may be regarded as these advantages because defects only appear in new wrappers. Regarding to the third advantage, the current wrapper structure is slightly complicated in converting SP's results to visualization data. Of course, it is quite logical to convert SP's results to data of a common structure, which are actually intermediate data of the visualization process since they are re-converted to data of specific form that are used for a particular visualization tool such as POV (Persistence of Vision), VRML (Virtual Reality Modeling Language) or AVS (Advanced Simulation System). Regarding to the fourth advantage, the wrapper's role of providing input data should be more flexible so that data which are not stored in GIS can be used. In general, an advanced SP requires input data for material properties which are not available in GIS; it is planed that IES has some libraries for this purpose, but it is not realized.

Based on the above discussion, the authors make the following two strategies of improving wrapper as 1) modifying the program structure of the member functions that provide input data to SP; and 2) finding an alternative of generating visualization data as intermediate data. It is expected that the flexibility of the member functions is increased by the first strategy so that specific input data which are not in GIS can be used easily. The second strategy will contribute to simplify the logic of providing data for visualization tools. One alternative of generating intermediate data is to directly generate data of specific format which can be used by a visualization tool. It is certainly true that this does not appear a right choice since the work requested to a wrapper is increased, i.e., making several files for each visualization tool instead of one file for the intermediate data. However, the results of an advanced SP, the amount of which is often huge, are well-structured so that they can be used for SP's own or related visualization. Thus, converting such well-structured data of the results to several data of specific format is not so laborious.

An actual task of modifying the program structure of the member functions inputting data is to make data which are not in GIS available to be handled by a wrapper. For given several data files, which are not related to GIS, a wrapper will be able to generate certain input data. A key point in generating data of specific format for visualization tools is to keep consistency among several visualization member functions as well as to reduce the computational time.

The simple structure of the new wrapper is shown in Fig. 4. Adding to GIS, a new element of Structure Data appears; this element is a source of specific data necessary for each advanced SP. Also, instead of producing V-Data in Fig. 3, the new wrapper directly produces files for POV, VRML or AVS. These files are the result of converting SP-Result.

4. Construction of Wrapper for UNIVERES

4.1 Characteristics of UNIVERSE

UNIVERSE [6),7),8)] is a 3-D non-linear dynamic FEM program which analyzes a dam-foundation-reservoir system; it is actually a commercial package which has been using by Electric Power Company Co., Ltd (J-Power). This simulation program has a capability to analyze not only body and foundation of a dam, but also interaction between the dam and the reservoir (which store water). The program consists of quasi-static analysis, eigenvalue analysis and dynamic analysis. These three analyses are sequentially executed.

The input of UNIVERSE is similar to that of general FEM programs; the geometric form of a structure is described with nodal points, elements, and material properties are described as well. Strong ground motion is given as an input external force at one site, which is described as time series data, although it is not necessary to input earthquake motion for quasi-static analysis and eigenvalue analysis.

The output of UNIVERSE is various files which have computational results of stress, strain, displacement, acceleration. These results are given as a time history. The spatial resolution is 100 Hz, even though the actual computation can change the time increment.

4.2 Modification of member functions

Modification of the member function structure must be made, based on the analysis of the data structure of UNIVERSE. UNIVERSE needs four input files for computation, and these files are usually made by hand, checking information of a target dam; this information is now stored in three

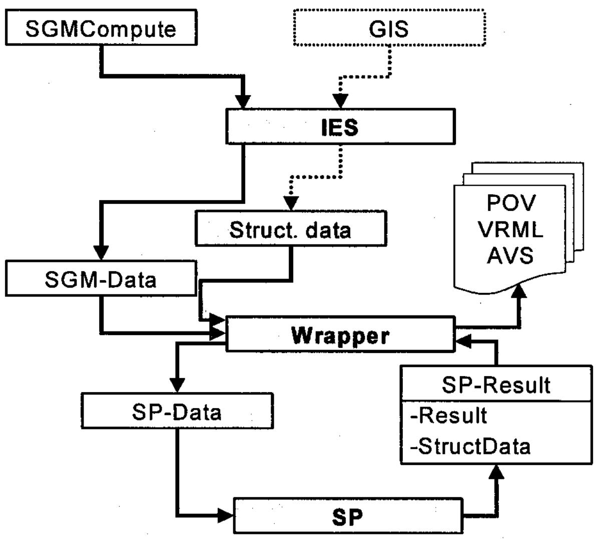

Fig. 4. Roles of new mediator as interpreter between IES and SP converting various data.

files which is made by third party. A new wrapper for UNIVERSE thus modifies the member function of PutStructureData() so that this function can make input files for UNIVERSE from stored data in three files.

PutStructureData() is a function which plays a role of converting data between the data for the dam structure information and the data actually input to UNIVERSE. The structure information is to describe nodes, elements and material properties; information should not be changed, but converted to data expressed in a specific format of UNIVERSE.

It seems that modifying PutStructureData() is trivial. However, this modification is essential when a user of IES changes the number of nodes or elements in order examine the numerical computation itself. Such change is now made by human hands. Once the function structure is modified, this change can be made by PutStructureData(); data input to UNIVERSE are automatically changed by means of PutStructureData() for a common structure information.

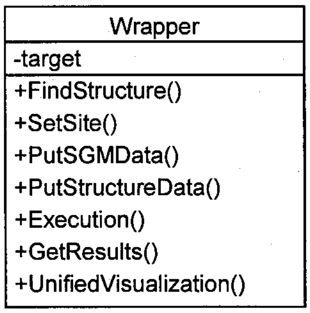

Fig. 5. Class diagram of wrapper; data and member functions.

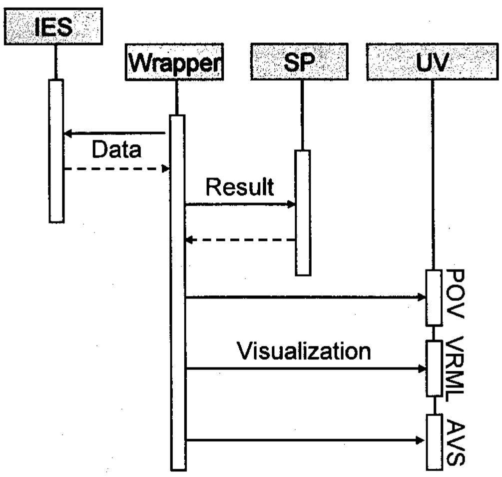

Fig. 6. Data flow sequence diagram for new wrapper. (UV: Unified Visualization)

4.3 Generating data of specific visualization tool

A new member function of the UNIVERSE wrapper, called UnifiedVisualization(), is added to the new wrapper. This function is similar to the IES's member function that converts the results of each SP to POV/AVS data of a common format; the IES's member function actually combine all data of SP's results and visualize seismic responses of all structures that are simulated. This is called unified visualization.

The new wrapper's UnifiedVisualization() produces distinct files each of which is used by a specific visualization tool of POV, VRML or AVS; see Fig. 4. Furthermore, these can be used to generate the unified visualization, just by combing files or data which are produced by other wrappers.

To enhance the computational speed and to increase the number of wrappers which produce data for the unified visualization, MPI parallel programming is implemented to IES; currently, an independent program written in MPI format is made for this purpose, and it will be merged into IES. A simple algorithm is made; one CPU is assigned to process one file which is made by certain wrapper, including the wrapper for UNIVERSE. In this manner, no confliction is made among CPU's. As mentioned, this algorithm is simple but effective as follows:

```
MPI_Status status;
int  Process,  myrank;
MPI_Init(&argc, &argv);
MPI_Comm_rank(MPI_COMM_WORLD,&myrank);
MPI_Comm_size(MPI_COMM_WORLD, &Process);
 filenos=new int[Process+1];
 filepercpu=nofiles/Process;
 remain=nofiles%Process;
 for(i=1; i<Process-remain; i++)
   filenos[i]=filepercpu;
   filenos[0]=0;
   for(i=filenos[myrank]; i<filenos[myrank+1]; i++)
     { //Reading_files(); }
MPI_Finalize();
```

The CPU time comparison is shown in the Table 1 for the above simple algorithm. It shows MPI parallel computation has significant improvement in the computation time when increase the number of node.

Table 1. Comparison of parallel computation time

No. of nodes	Time/(Sec)
4	1775.62
6	1361.42
12	544.35
18	195.15
Without MPI	2236.1

4.4 Class diagram of new wrapper

Figure 5 shows the class diagram of the new wrapper for UNIVERSE. A sequence diagram is shown in Fig. 6. This figure describes the data which are transmitted from the wrapper to the simulation program and to the visualization tool; UV stands for the unified visualization that covers the whole buildings and structures within a target area. The new member function is the replacement of VisualizeResult(); as planed in Section 3, an alternative of generating intermediate data is achieved, and this new member function can contribute efficient computation of the unified visualization.

In Fig. 7, the program structure of UnifiedVisualization() is presented; it has two layers for data sets, a basic layer for the original results of UNIVERSE and a layer for data which replace the previous intermediate data of the visualization process. In this layer, nodes, elements and values are actual data which are made by pointing the data in the basic layer. In this manner, the consistency of the data which are used by several visualization tools is achieved by the minimum effort; the possibility that the original results computed by UNIVERSE are altered is quite small, even though the amount of the results is huge.

In software engineering, scalability, modularity and interoperability are the desirable properties of a system or process. A system like wrapper or a set of wrappers is easily evaluated by them. These characteristics are used as an index with which previous and new wrappers are compared.

Scalability means ability to either handle growing amount of data in a graceful manner, or to be readily enlarged. MPI parallel computation is applied to new wrapper, in order to meet the demand to handle growing amount of SP's results having without a bottle neck. On other hand, no parallel computation technique to enhance its scalability is applied to the previous wrapper.

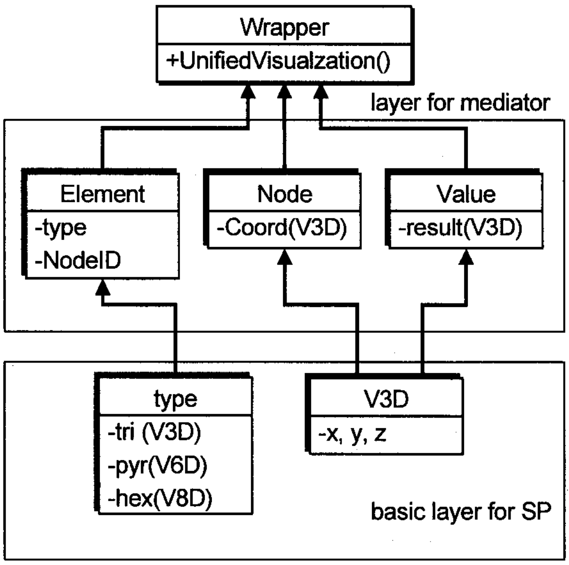

Fig. 7. Layered program structure for member function of UnifiedVisulzation().

Modularity is the property of computer programs that measures the extent to which they have been composed out of separate parts called modules. A modular approach is essential in programming wrappers, where a large-scale system composed of modules each of which has a specific purpose and communicates with each other, in order to increase the system's overall performance. Here are six core design operators essential for a modular system: [5]

1. Splitting – Modules can be made independent
2. Substituting – Modules can be substituted and interchanged
3. Excluding – Existing modules can be removed to build a usable solution
4. Augmenting – New modules can be added to create new solution
5. Inverting–The hierarchical dependencies between modules can be rearranged
6. Porting – Modules can be applied to different contexts

Though previous and new wrappers are coded in object oriented programming to meet the above core issues, authors have paid more attention to make disposable code which is composed of a number of small, self-contained units in new wrapper.

Interoperability[10] is the ability of a system to work with other systems without any special effort on the part of the user. Interoperability concept in IT is "The network is the computer." To reach this ability, a system should have a loosely coupled distributed systems built by interconnecting multiple work-stations through a local area network (LAN). Distributed C++ [9] has the capability to work through LAN in work-stations. Authors are currently not interested in Distributed C++ because of a computer cluster which has enough capability to handle IES to a sufficient level.

The new wrapper has improvement over the previous one in scalability and modularity. In interoperability, however, no improvement is made. Summary of the comparison of the new and previous wrappers is presented in Table. 2.

Table 2. Comparison of previous and new mediators.

Index	Previous	New
Modularity	Medium	High
Scalability	Low	High
Interoperability	Very low	Very low

5. Performance of Wrapper for UNIVERSE

5.1 Performance via visualization

The performance of the new wrapper for UNIVESE can be mainly related to visualization or unified visualization. Visualization can be an effective method for understanding numerical results which are extracted from a massive volume of numerical data. It is also essential for the presentation to general public or local government officials.

Authors have applied several visualization tools, such as POV (Persistence of Vision), VRML (Virtual Reality Modeling Language) and AVS (Advanced Visual System) to make visualization images with different intension is explained below.

POV-ray[11)] is a ray-tracing program which works with its simulated camera and traces rays backwards out into the scene. A user specifies the location of the camera, light sources, and objects as well as the surface texture properties of objects, their interiors (if transparent) and any atmospheric media such as fog, haze, or fire. Ray-tracing programs can be applied for image analysis.

There are two difficulties in making POV files. First, triangular element can only be used. Other type of element can not be used in the mesh2()[11)] function (mesh2 is a object among the tool box of povray to represent mesh). In data writing class for POV, it has an additional member function to convert all other type of elements into triangular element. Second, it has some bugs when handle with very huge amount data. POV-ray produces some white color elements irrelevant to the assigned element color. Authors have found that if data amount is huge the number of precision of nodal coordinate is limited for POV-ray.

VRML[12)] has been designed to create a friendlier environment for the World Wide Web. A VRML file resides on a website just like an HTML file. VRML incorporates 3-D shapes, colors, textures and to produce a virtual world that a user could walk and fly through. VRML is an interpreted language. That is, commands written in text are interpreted by the browser and displayed on the user's monitor. With the help of VRML, IES results can be uploaded to internet without any effort.

AVS[13), 14)] is industry leading, interactive data visualization software which gains critical insight from all types of data. For our visualization purpose Micro-AVS is used in windows environment. It is good for general purpose application. It can correspond to whatever the physical amount of data which produced by UNIVERSE simulation program.

5.2 Application to a Real Dam

The wrapper for UNIVERSE is used to simulate seismic response of an existing arch dam [5)]. This dam is a parabolic thin arch dam which was constructed in 1970's. The height of the dam is 116.5m and the crest length of the dam is 311m.

The wrapper for UNIVERSE has played the role of an interpreter, by handing input data and output results of UNIVERSE completely, satisfying all requirements of IES, as well. Static and dynamic images of the dam for a designated earthquake response are produced.

Just for presentation, the authors have considered a non-realistic situation that the dam is located in the center of a city; the city model is made with the aid of available GIS data. The results of UNIVERSE dynamic analysis are gathered by the wrapper's member function UnifiedVisualization(), and the time history of displacement for all the buildings in the city are combined for the unified visualization; see Fig. 8. These images are generated by POV, each snapshots view point is changed with the aid of moving simulated camera. UNIVERSE computed time history (dynamic analysis) displacement of dam in some particular nodes only, on the other hand maximum displacement of dam is computed to whole nodes. Since the dam and buildings have quite different scale of displacement due to their dimension, the displacement of the dam and the building is multiplied by a factor of 6 and 20 respectively.

a) t = 5 [sec]

b) t = 20 [sec]

Fig. 8. Snapshots of unified visualization through wrapper.

6. Concluding Remarks

A new wrapper is constructed for UNIVERSE, following the two strategies that are made based on the KISS principle. The performance of the new wrapper is improved, regarding to scalability and modularity. With the aid of this wrapper, IES is able to handle a huge amount of UNIVERSE input data and results, in order to make the unified visualization of a virtual city which includes an arch dam in it. The scalability that is achieved by applying MPI parallel programming will be advantageous when more advanced SP's which execute large scale computation and .produce huge results.

References

1. M. Hori., *Introduction to Computational Earthquake Engineering*, Imperial College Press, 2006.
2. F. Yang, T. Ichimura and M. Hori, Earthquake simulation in virtual metropolis using strong motion simulator and Geographic Information System, *Journal of Applied Mechanics*, JSCE, Vol.5, pp. 527-534, 2002.
3. T. Ichimura and M. Hori, Development of prototype of Integrated Earthquake Disaster Simulator using digital city and strong ground motion simulator with high resolution, 13th *World Conference on Earthquake Engineering*, No. 1418, 2004.
4. T. Ichimura, H. Itami, T. Samo, M. Hori, and N. Yamaguchi, Construction of digital city Kobe & a basic discussion on application of IES approach to earthquake simulation, *Journal of Structural Engineering*, JSCE, Vol. 51A, pp. 513-520, 2005.
5. C. Y. Baldwin and K. B. Clark, *Design Rules*: the power of modularity- volume 1, The MIT Press, 2000.
6. Y. Ariga, Study on quantitative evaluation of dynamic properties of dams by 3-D reproduction analyses, Thesis for Doctorate of Saitama University, 2001.
7. Y. Ariga and H. Watanabe, Reproduction analysis of real behavior of existing arch dam during the 1995 Hyogoken-Nanbu Earthquake, 13th *World Conference on Earthquake Engineering*, No. 405, 2004.
8. Y. Ariga, Z. Cao and H. Watanabe, Development of 3-D dynamic analysis method for coupled dam-joints-foundation-reservoir system, 13th *World Conference on Earthquake Engineering*, No. 412, 2004.
9. H. Carr, Distributed C++, Ph.D. Dissertation, the University of Utah, 1994.
10. B. Bukovics, NET 2.0 *Interoperability Recipes: A problem-Solution Approach*, après, 2006.
11. POV-Team. Introduction to POV-Ray for POV-Ray Version 3.6.1. (Available on www.povray.org).
12. A. L. Ames, D. R. Nadeau, and J. L. Moreland, *VRML 2.0 Sourcebook*, Second Edition, John Wiley & Sons, Inc., , 1997.
13. Advanced Visual System Inc., *AVS/Express Developer's Reference*, Release 3.0, Waltham, 1996.
14. Advanced Visual System Inc., *AVS User's Guide*, Waltham, 1992.

(Accepted on April 12/2007)