Common Modeling Data: New Approach in Designing Integrated Natural Disaster Simulation Software

1. Introduction:

Roles of GIS in developing and managing modern cities are apparent. Several layers of information are massed in GIS which can be used into to or alone to achieve a better understanding of prospective of current city situation.

Although the basic layer of GIS is urban elements (e.g. buildings), lack of studies on these element's behavior under different natural disasters are noticeable.

The authors have been developing integrated earthquake simulation (IES) environment as a candidate of simulation-based earthquake disaster prediction system. In recent studies[1),2),3)] developing and applicability of such system are illustrated which is aimed on earthquake disaster simulation and prediction.

In order to increase capability of IES in coping with different type of input data (e.g. a different source of GIS data, FEM-Based data) and utilizing these data as a feed of other layers, an alternative software design is proposed.

With the proposed method, authors tried developing step toward Integrated Natural Disaster Simulation (INDS), which is a super-class of IES. In this paper is tried to show how IES design can be improved such that different GIS data, simulation programs, and visualization techniques can be used with more consistency between different layers of software.

The contents of the present paper are as follows: In section 2, briefly IES structure is described and compared by INDS structure. In section 3, implementing Common Modeling Data (CMD) is represented.

2. Review of IES and Introduction of INDS

IES simulation was done by three operational layers, Strong Ground Motion (SGM) simulator, Simulation Program (SP), and Visualization layers.

Connections between these three layers are established by Wrapper (Mediator)[2)] which essentially behaves as an interpreter of data, which should be used by IES layers. Wrapper is responsible of reading GIS data, converting to specific type of structure and executing Simulation Program (SP), and in the last step reading and changing results of SP, somehow the visualization layer can visualize it in different visualization environment. The schematic view of IES layers is depicted in Fig. 1.

With analogy to IES the integrated natural disaster simulation environment can be defined to consider different sources of natural hazards and their effects on infrastructures (e.g. building, bridges, pipelines, etc.), human reactions against disasters (e.g. evacuation, etc) and disaster mitigation (e.g. recovery and rehabilitation planning).

The proposed INDS structure can be described with one kernel class and four aggregated abstract classes like Data, Hazard simulation, Disaster simulation and Visualization classes.

Kernel class manages initialization, connection, and execution of each aggregated layers. Data layer reads different source of input data like GIS (Geographical Information System) or FEM-Based (Nodes-Elements) data and changes them to appropriate form which is called CMD (CMD will be described in next section) since after. Then Kernel sends CMD data for hazard simulation class, which simulate the natural hazards like earthquake, flood, or volcano and store the results of each simulation based on the CMD standard.

Several kinds of third party simulation programs can be plugged in Disaster class. In final stage, visualization class can read results of Hazard and Disaster simulations and visualize simulated environment.

Figure 1 points out the necessity of having CMD in INDS. Each layer in IES produce different types of data that is inconsistence with other layer's input data.

Wrapper as an interpreter has to tackle with converting each of them to appropriate format as an input data for others. Therefore, by increasing variation of the input data resources, simulation programs and visualization tools wrapper's structure get more complicated. This complication is essentially in contradiction with KISS (Keep It Simple, Straightforward) software development principals.

INDS uses CMD concept that is inspired from FEM program input file format. Node and Element are used

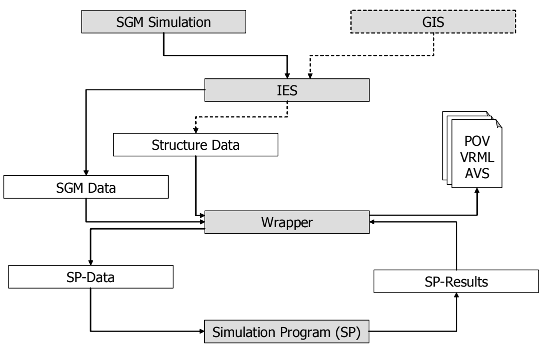

Figure 1. Schematic of IES.

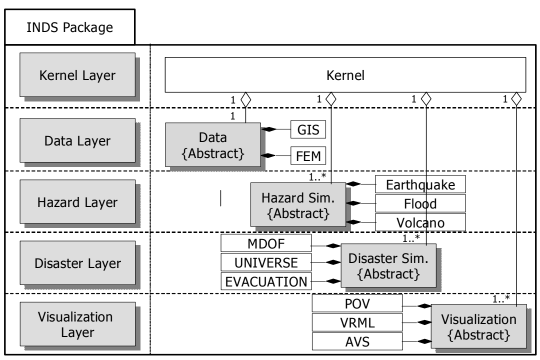

Figure 2. INDS Class Diagram.

by FEM programs as fundamental descriptors of a geometrical shape. The structure and implementation of INDS is described in following section.

3. Common Modeling Data (CMD)

To establish appropriate data communication between each layer, common modeling data is developed. Common Modeling Data concept is a solution to describe data in a general form to be understood by INDS layers. Then each layer can generate a model based on CMD. In addition, CMD is used as a communication data between each layer instead of different type of data that was changed by wrapper to proper format in IES.

Figure 3 shows component of CMD. It consists of two classes, Shape and Data. Shape class carry geometric description of data. Shape is aggregated with Node and Element class.

Data class is a container for information that is generated before, during or after executing different layers. Data class associates with Shape class. As an example after executing Hazard simulation layer for earthquake hazard, earthquake wave can be put in the data format, which can be used by other layers as an input data.

Shape and Data classes have Input( ) and Output( ) member functions which can write or read their attributes with a coherent format. Hence, it is not to carry all the information in memory. Different layers can access to other layers information just by using these member functions.

Communication diagram between INDS layers are illustrated in fig 4. Kernel class communicates with others by CMD and just in start and end of execution, INDS sends two

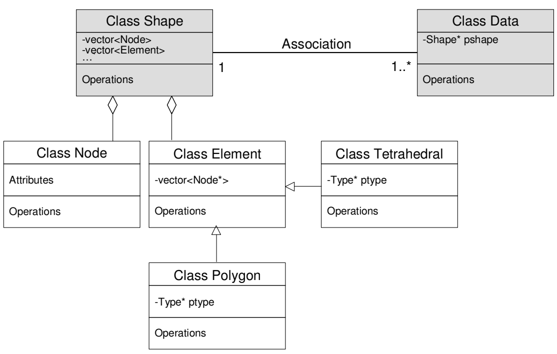

Figure 3.CMD package components.

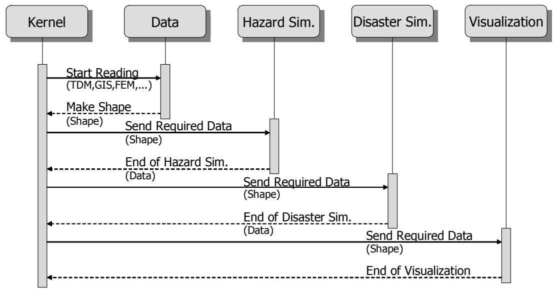

Figure 4.Comunication Diagram between INDS layers.

controlling signal.

Considering huge amount of data processing and analysis that INDS is dealing with, INDS demands high computational effort. Therefore using high performance computation platforms is inevitable. By means of proposed layered structure INDS and CMD in kernel level which prepare job-distinction, INDS can be execute on shared memory computers (Workstation), Distributed Memory Computers (Cluster) or Grid Computation Networks.

4. Conclusion:

A layered-structure, for integrated natural disaster simulation (INDS) is proposed. In addition, to establish inter-intra communication between these layers Common Modeling Data (CDM) concept presented as a substitution of Wrapper (Mediators) in previous studies. With the proposed design, using different type of high performance computing techniques is possible in kernel level.

5. References:

1. M. Hori., Introduction to Computational Earthquake Engineering, Imperial College Press 2006.
2. S. Nallathamby, M. Hori, Y. Ariga., Improvement of IES Wrapper for Plug-in of Advanced Simulation Programs, Journal of Applied Mechanics Vol.10, pp-573-579, 2007.
3. T. Ichimura, M. Hori., Development of prototype of Integrated Earthquake Disaster Simulator using digital city and strong ground motion simulator with high resolution, 13th Conference on Earthquake Engineering, No.1418,2004.