Advances in Engineering Software 42 (2011) 398–407

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

Short Communication

Web-based post-processing visualization system for finite element analysis

Wei-Chu Weng

Department of Systems Engineering and Naval Architecture, National Taiwan Ocean University, Keelung, Taiwan, ROC

ARTICLE INFO

Article history:
Received 29 January 2010
Received in revised form 25 October 2010
Accepted 8 March 2011
Available online 7 April 2011

Keywords:
Web system
OpenGL
Database
Finite element analysis
Post-processing
Structural static analysis

ABSTRACT

In this paper, we propose and implement a website of post-processing system for finite element analysis (WebDFEA). Finite element analysis is a computer-aided engineering tool and is popular for static/dynamic structure analysis. It includes three processing systems where post-processing system is to graphically demonstrate the analysis result of a structure model analyzed by finite element method. WebDFEA performs as a website. It is cross-platform because it can auto-detect a client computer platform and auto-download proper OpenGL API for drawing computer graphics. It can draw precise graphics on webpage which can be free controlled by the mouse as a manner in professional software. A database server is involved to store finite element model data and its analysis result. The graphic user interface (GUI) of WebDFEA is a flexible GUI comprising three parts: the switch buttons designed by HTML, the display board and the color bar both developed in Java. The three components are independent and cooperative with each other. They can be recombined without running errors for different purposes. A ship hull section with half a hatch is chosen as the study case to test WebDFEA website. Its finite element model comprises 11,442 triangle elements (shapes). The timeframe starting when WebDFEA is connected to the end when the model is demonstrated is acceptable.

1. Introduction

Finite element analysis (FEA) is a computer-aided design software and is usually used for structural static/dynamic analysis. Basically, there are three sequential steps for processing an analysis: preprocessing, analysis, and post-processing. During preprocessing, users have to firstly create a geometric model. A geometric model is a 2D/3D geometric graphics. The geometric model is then meshed into a number of small analytical elements for analysis. A finite element model is thereby generated and more analytical elements generated result in more accurate analysis. The analysis step uses finite element method to analyze the finite element model. The post-processing is to graphically present the analysis result based on the finite element model.

Software has three kinds of different running framework: PC-standalone, Internet-based, and web-based. PC-standalone and Internet-based software need installation for usage while the latter can communicate with remote servers through Internet. Web-based software, however, does not require local installation but users need to be connected to the Internet via a web browser. The main advantage of web-based software is that users do not need to re-install the software when their computers break down. Web-based software also provides enough central storage for user's working data and files. In addition, developers can easily maintain and upgrade web-based software without inconveniencing users.

Most professional CAD/CAE/CAM software are either PC-standalone or Internet-based. However, more recently, many researchers have successfully developed web-based systems to facilitate CAD/CAE/CAM technologies that are applicable to today's distributed production environment. For example, Bouzakis et al. have proposed a web-based framework and implemented it as a web service for designers and manufacturers to communicate for manufacturing a workpiece [1]. Lee et al. have presented a web service system to efficiently integrate and manage resources distributed over heterogeneous platforms [2]. Chassiakos and Sakellaropoulos have presented a web-based system facilitating the effective management of construction information and the communication between project members; this information is large in amount, varies in nature, and dispersed over several spaces [3]. Cheng and Fen have presented a prototype of a web-based system that helps scientists collaborate and explore their work in an interactive, visual, and experimental environment [4]. Marante et al. have presented a web system to simulate damaged concrete frame structures with cracking plastic hinges [5].

In CAD/CAE/CAM software, computer graphics is an important function for demonstrating two-dimensional (2D) or three-dimensional (3D) products or designs. One of the most popular tools for developing computer graphics applications is Open Graphics Library (OpenGL). OpenGL is a standard specification defining a cross-language, cross-platform API (application programming

E-mail address: pw@mail.ntou.edu.tw

0965-9978/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.advengsoft.2011.03.003

interface) for writing applications that produce 2D and 3D computer graphics [6]. It is an open standard API available on most operating systems and is usually used for developing professional software for computer-aided design, virtual reality, scientific visualization, information visualization, flight simulation, computer games, etc. Several OpenGL applications have been successfully developed and presented, to date, in academic disciplines, and a number of them are web-based.

Marti et al. have developed a computer-aided system using C++ for teaching computer graphics [7]. Chen and Cheng have developed a drawing system using Ch programming language, where users can enter text commands to draw computer graphics on a web browser in real time [8]. Massire et al. have presented a web-based RNA (ribonucleic acid) demonstration system using Ch [9]. Bloem and Bailey have developed a visualization system of regional weather data using Ch, where the weather data is accessed from an Internet-based weather service [10]. Shimoide et al. have developed a 3D visualization system using Java and Java OpenGL package (JOGL) for demonstrating protein shapes as computed by hydrodynamics [11]. Hobona et al. have presented a web-based 3D geospatial data system developed using Java and the Java3D package [12]. Chen and Lin have presented an Internet-based finite element analysis (FEA) system on a parallel computing environment developed using Java [13]. All the systems mentioned above are web-based except the ones in [7,13]; two are associated with database systems [9,10]; some are developed for education [7,8]; some for bio-informatics [9,11]; and some for weather and geographical information [10,12].

Few studies have focused on web systems with computer graphics for CAD/CAE/CAM. Except, Song et al. have successfully developed an Internet-based system for reverse engineering processes [14] and a web-based system for styling processes [15]. Both the systems can demonstrate 3D computer graphics using OpenGL by Microsoft technologies and products. In this paper, we propose a web system for the post-processing of finite element analysis and called WebDFEA. The 4th letter D in the word WebDFEA indicates "demonstration" and is inserted to prevent from being interpreted as a whole web-based finite element analysis system. WebDFEA is developed especially by focusing on its graphic user interface and website characteristics.

Website is a web system. From the user's perspective, website is popular not only because it is easy to use (friendly user interface) but also because users can access their updated and long-term stored data by using any on-line computers. Based on this, WebDFEA is developed as a website, i.e. software webalization. The word webalization indicates the transformation from an original software framework to website framework. Before development, we summarize website's common characteristics as below:

- Web browser is websites' common window and webpage is their user interface.
- Website is apparently cross-platform. It can be demonstrated on various types of computer platforms.
- Most websites can be browsed without further installing extra software. In general, necessary software is automatically installed. However, users do not like frequently auto-installations except necessary software provided by trustable companies such as interpreters or runtime environments.
- Many websites provide storage service for storing users' data. Database management system (DBMS) is one of the most popular tools for managing users' data. DBMS can store and manage various forms of data such as sound tracks, movies or any document files.
- Websites can fast demonstrate multimedia information composed of texts, sound tracks, movies, images/pictures or animations, etc. At least, users can stand for the time delay even with heavy Internet traffic.

As a webalized professional software, WebDFEA must be able to demonstrate precise 2D/3D computer graphics with professional graphic user interface (GUI) and perform website common characteristics. This paper is an extension of the author's preliminary work [16] where a web-based system was developed to demonstrate precise 3D graphics on web browser and all model data are stored in a remote database server.

WebDFEA has a web server hosting the GUI webpage and a database server that stores the graphic data of the finite element model and its analysis result. We chose Java as the development tool because it is a free and platform-independent programming language. In addition, Java can be used to develop applications embedded in webpage. Furthermore, there exists JOGL package (exactly the OpenGL library) for professional development by Java.

In this paper, first of all, we briefly introduce finite element analysis by using a practical example in Section 2. Some images captured from commercial software ANSYS are demonstrated for later comparison to WebDFEA. Then, we describe the design methodology for WebDFEA in Section 3 and demonstrate WebDFEA performance in Section 4. At last, we make some conclusion in Section 5.

2. Finite element analysis

Finite element analysis is used for static or dynamic structural analysis using the finite element method. Static structural analysis is mainly employed to analyze the stress distribution of a structure for safety. A section of a ship hull with half a hatch has been chosen as an example [17]. As shown in Fig. 1, its geometric model is constructed with commercial software ANSYS. Each of the two hatch corners has a small hole. The red arrows indicate the direction of external forces. The geometric model must be meshed to a finite element model for analysis, as shown in Fig. 2. The two black areas are hatch corners where they are meshed into more analytical elements for a more accurate analysis result.

An analytical element is a small 2D/3D geometric shape with nodes predefined in the analysis software. The finite element method calculates the stress for every node in the finite element model. A triangle analytical element has three nodes while a cubic analytical element has two types for choice: one with eight nodes and the other with 20 (one more node on every edge). Before analysis, every used analytical element has material parameters to be set such as density and Young's modulus. Furthermore, users have to specify how the finite element model is constrained and what kind of external forces are loaded. After analysis, the post-processing step demonstrates the stress distribution by coloring the finite element model. Red represents the largest stress and blue represents the smallest stress (Fig. 3)[1]. Fig. 4 shows a magnified view of the hatch corner.

3. WebDFEA system

In order to develop WebDFEA website, our design methodology is summarized as below:

- WebDFEA adopts OpenGL API (application programming interface) to demonstrate precise 2D/3D computer graphics on webpage. It is a high-level graphics library (directly used by developers) which can draw precise computer graphics and is usually applied to professional software development.

[1] For interpretation of color in Figs. 1–14, the reader is referred to the web version of this article.