The ASCE Civil Engineering Conference and Exposition, Nashville, TN, USA, November 12-15, 2003

A Software Framework for Internet-Enabled Nonlinear Dynamic Structural Analysis

Jun Peng and Kincho H. Law
Department of Civil and Environmental Engineering, Stanford University

Abstract

This paper describes the research and prototype implementation of an Internet-enabled software framework that facilitates the utilization and the collaborative development of a finite element structural analysis program by taking advantage of object-oriented modeling, distributed computing, database and other advanced computing technologies [1, 2]. This new framework allows users easy access to the analysis program and the analysis results by using a web-browser or other application programs, such as MATLAB. In addition, the framework serves as a common finite element analysis platform for which researchers and software developers can build, test, and incorporate new developments.

Introduction

The system architecture of the Internet-enabled collaborative framework is schematically depicted in Fig. 1. A component-based modular architecture is utilized for the design to support multiple parties and applications and the interaction among these participants. The *Analysis Core* module is built upon an object-oriented finite element program, which is named OpenSees (Open System for Earthquake Engineering Simulation) [3]. New element and material technologies, as well as new analysis strategies and solution strategies can be brought into the *Analysis Core* module to enhance the functionalities of OpenSees. The *User-Interaction* module provides an interface to facilitate the access to the software platform and the analysis results. The *Registration and Naming Service* is provided for on-line services to register to the core so that these services can be found during an analysis. Two approaches are provided for remote access to element services residing in different locations. The *Distributed Element Service* is intended to provide a communication link to remote element services where the element code is executed. The *Dynamic Linked Element Service* is implemented to provide a flexible way of dynamically binding elements to the core in real time. Last but not least, the *Database Interface* module can take advantage of a persistent data storage that further enhances data access and facilitates post-processing of analysis results.

The Internet-enabled collaborative framework can provide greater flexibility and extensibility than traditional structural analysis programs, which are typically packaged individually. The mechanics of the collaborative model is illustrated in Fig. 2. In the proposed framework, the users build their structural model by using a web-based model-building service on the client site. The model then can be sent to the analysis core by using the Internet as a communication channel. Upon receiving the analysis model and other related information, the core server authenticates the user's identity and starts performing a structural analysis on the received model. During the analysis, elements that are available in the core can be accessed locally from the static element library, whereas other elements are obtained from online element services. In order to find the required elements, the *registry* is queried to find the on-line element services, which have already been pre-registered with the core platform. After the analysis is completed, part of the results will be returned to the user by generating a dynamic web page in the user's web browser.

There is a standard interface/wrapper for communicating the element with the object-oriented analysis core. To introduce new elements into the analysis core generally composes of creating subclasses of *Element* class whose common interface is defined in the analysis kernel. After the development process is finished, the new element code can be compiled with the core platform and become part of the static element library. In addition to the traditional way of building element library for new element development, the new elements can also be developed in the form of on-line element service. We propose two types of on-line element services: distributed element service and dynamic shared library service respectively. These two services can be differentiated based on where the actual computation code resides.

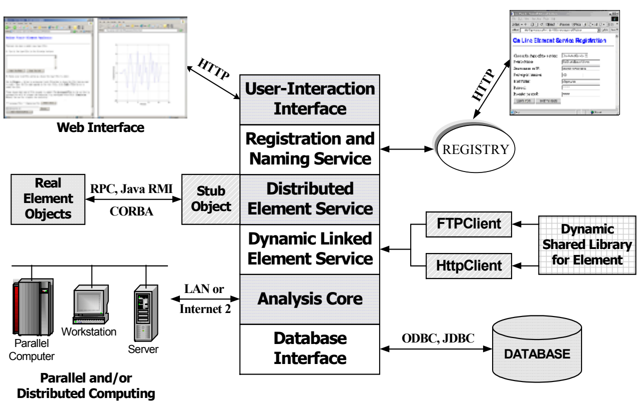

Figure 1 – Collaborative System Modules

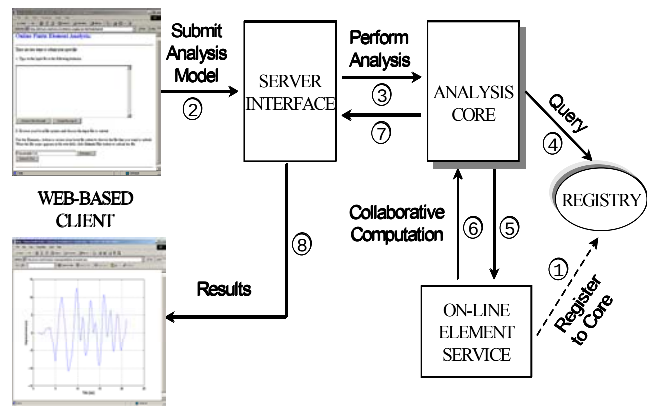

Figure 2 – Mechanics of the Framework

The collaborative finite element software framework also includes data and project management functionalities [4]. A commercial off-the-shelf (COTS) database system is employed to store selected analysis results and to provide flexible data management and data access. The Internet is utilized as a data delivery vehicle and a data query language is developed to provide an easy-to-use mechanism to access the needed analysis results from readily accessible sources in a ready-to-use format for further manipulation. In the data access system, a selective data storage scheme is introduced to provide flexible support for the tradeoff between the time used for reconstructing analysis domain and the disk space used for storing the analysis results. The user has the flexibility to specify storing only the required and needed data, all the other analysis results can be accessed through OpenSees core with certain re-computation during the postprocessing phase of an analysis. Data are organized internally within the OpenSees core based on an object-oriented model. Data saved in the COTS database are represented in three basic data types: Matrix, Vector, and ID. For external data representation, XML (eXtensible Markup Language) is chosen as the standard for representing data in a platform independent manner. To support the interaction with both humans and other application programs, a data query language is defined to provide support for data retrieval and postprocessing functionalities. The engineering data access system gives great flexibility and extendibility to the data management in finite element programs and can provide additional features to enhance the applicability of FEA software. Finally, a simple project management scheme is developed to allow the users to manage and to collaborate on the analysis of a structure.

References

[1] J. Peng, F. McKenna, G. L. Fenves, and K. H. Law. "An Open Collaborative Model for Development of Finite Element Program," *Proceedings of the Eighth International Conference on Computing in Civil and Building Engineering (ICCCBE-VIII)*, Palo Alto, CA, pp. 1309-1316, 2000.

[2] J. Peng and K. H. Law. "A Prototype Software Framework for Internet-Enabled Collaborative Development of a Structural Analysis Program," *Engineering with Computers*, 18(1):38-49, 2002.
[3] F. McKenna. *Object-Oriented Finite Element Programming: Frameworks for Analysis, Algorithm and Parallel Computing*, Ph.D. Thesis, Department of Civil and Environmental Engineering, University of California at Berkeley, Berkeley, CA, 1997.
[4] J. Peng, D. Liu, and K. H. Law. "An Engineering Data Access System for a Finite Element Program," *Advances in Engineering Software*, 34(3):163-181, 2003.