하이브리드 구조실험을 위한 데이터 모델

Data Model for Hybrid Structural Experiments

이 창 호† 토마스 마룰로* 리차드 소스**
Lee, Chang-Ho Thomas Marullo Richard Sause

(논문접수일 : 2009년 3월 30일 ; 심사종료일 : 2009년 7월 30일)

요 지

하이브리드 구조실험에서는 구조물을 여러 개의 부분구조물로 나누어서 실험과 해석을 하고 이의 결과를 합쳐서 전체적인 구조물의 거동을 파악한다. 이러한 방법은 진동대 실험과 비교하여 구조물의 크기제한의 영향을 받지 않는 유사동적 실험에 효과적이다. 하이브리드 구조실험과정에서 발생된 데이터와 관련 정보를 저장하고 검색할 수 있는 컴퓨터시스템을 만들기 위해서는 하이브리드 구조실험과 관련된 정보를 체계화시켜서 구성하는 작업이 선행되어야 한다. 본 논문은 하이브리드 구조실험에 관련된 정보를 표현하는 데이터 모델을 제시하고 있는데, 이 데이터 모델은 포괄적인 구조실험 정보를 표현하는 데이터 모델의 하나인 리하이 모델에서 하이브리드 실험부분을 개선한 것이다. 하이브리드 구조실험에서의 부분구조물들을 표현하기 위하여 실험모델 클래스와 해석모델 클래스를 정의하였고, 이러한 클래스들의 정보교환을 조정하는 클래스를 정의하였으며, 제한된 범위의 시스템을 구현하여 객체들 간의 연결 상태를 파악할 수 있도록 하였다. 본 논문에서 기술한 데이터 모델은 구조실험자와 연구자들이 사용할 수 있는 하이브리드 구조실험 정보를 저장하는 컴퓨터 시스템을 개발하는데 적용할 수 있을 것으로 사료된다.

핵심용어 : 데이터 모델, 하이브리드실험, 구조실험, 리하이 모델

Abstract

The hybrid approach for structural experiments decomposes a structure into independent substructures that can be tested or simulated. The results from the decomposed substructures are combined to predict the behaviors of the entires structure. The hybrid approach is especially useful for the hybrid pseudo-dynamic tests that overcome the limitations of size of a test structure present in a shaking table test. The development of a computer system for the hybrid experiment requires a data model that formally organizes the information involved in the hybrid experiments. This paper provides the data model for representing the information involved in the hybrid experiments, by modifying the classes and attributes for the hybrid experiments in the Lehigh Model that is one of the data models for structural experiments. The data model for the hybrid experiments includes the classes for the physical substructures being tested and the analytical substructures being analyzed, and the simulation coordinator managing the overall experiments. Some objects for classes are implemented as an example to show the links among the classes. The data model presented in this paper can be applied for developing a computer system that helps structural engineers and researchers store, share, and access the information for the hybrid experiments.

Keywords : *data model, hybrid experiment, structural experiment, Lehigh Model*

1. Introduction

The structural experiments are conducted to examine the behaviors of structural components, connections, and assemblies. The information involved in the structural experiments is often complicated and

---

† 책임저자, 정회원 · 한경대학교 건축학부 부교수
Tel: 031-670-5275 ; Fax: 031-674-8656
E-mail: chlee@hknu.ac.kr
* IT Manager, ATLSS Center, Lehigh University, PA, USA
** Joseph T. Stuart Professor of Structural Engineering, ATLSS Center, Lehigh University, PA, USA

• 이 논문에 대한 토론을 2009년 12월 31일까지 본 학회에 보내주시면 2010년 2월호에 그 결과를 게재하겠습니다.

hard to understand, and makes it difficult to efficiently access, share, and use the information. There have been several research efforts to organize and represent the information involved in structural experiments. Some efforts were to provide databases for existing structural test data. Examples are the SAC Design Information Database(http：//www.sacsteel.org/, 2000) and the PEER Structural Performance Database(http://nisee.berkeley.edu/spd, 2004). Other efforts were to develop data models to help researchers manage the information and the data using predefined hierarchies and categories of information. Numbers of data models have been developed in support of the George E. Brown Jr. Network for Earthquake Engineering Simulation (NEES; http://nees.org/, 2009), which is a shared national network of 15 experimental facilities in the United States, and is operated until 2014 for ten years supported by the NEES Consortium, Inc.(NEESinc). The Oregon State Model(Oregon State University and Network Alliance for Computational Science and Engineering, 2003) was developed to tsunami wave basin experiments. The Stanford Model(i.e., the Reference NEESgrid Data Model; Peng et al., 2004) was primary for shake table experiments. The NEEScentral Model(NEESit, 2007) describes the general categories of information related to structural experiments. The Lehigh Model(Lee et al., 2008) was intended to represent more detailed and flexible representation of test conditions and tests.

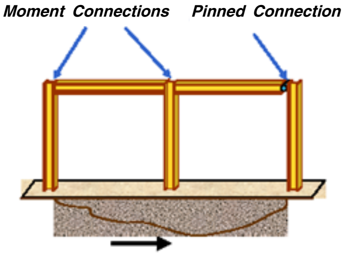
Previously developed data models for structural experiments have provided researchers with tools to efficiently access and understand the details of the structural experiments. In particular, the NEEScentral Model is actively used for the many projects by the NEES related researchers. However, there exist some limitations for representing the full sets of information involved in all kinds of structural experiments, since the organizations of the information differs with the kinds of experiments. For example, the organization of a hybrid structural experiment can be quite different from that of a typical structural experiment. The representation of the organization of the hybrid structural experiment was explored in the Lehigh Model(Lee et al., 2008). This paper discusses more details of the hybrid structural experiments.

This paper begins by explaining the characteristics of and interactions in the hybrid structural experiments. The information and interactions involved in the hybrid structural experiments are represented by using the classes and the attributes which have been revised from the Lehigh Model. Then, some objects of the classes are implemented for an example hybrid structural experiment. Finally, some concluding remarks for developing a complete data model for the hybrid structural experiments are presented.

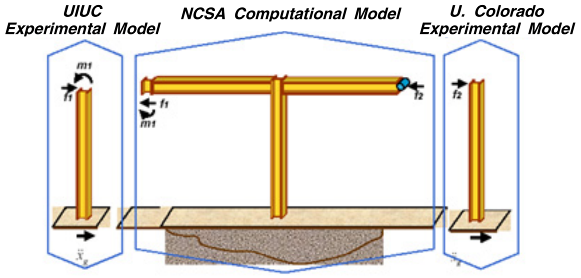
2. Hybrid Experiments

2.1 Characteristics of Hybrid Experiments

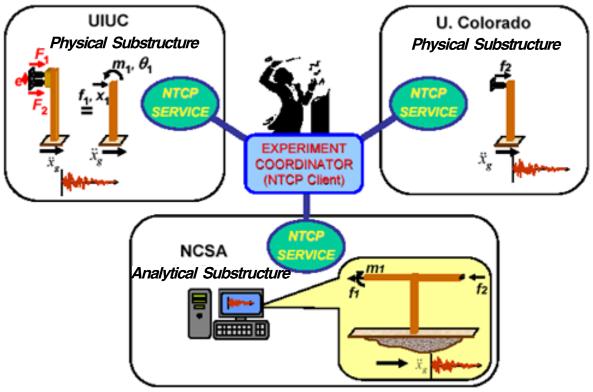
Structural engineers have traditionally investigated the behaviors of structures with either physical experiments or computational simulations. A hybrid approach decomposes a structure into independent substructures that can be located at different sites, tested or simulated separately, and integrated (Dermitzakis et al., 1985). The hybrid approach is especially useful for the hybrid pseudo-dynamic tests that avoid fabricating of an entire structure and overcome the limitations of size and mass of a test structure present in a shaking table test(Pearlman et al., 2004; Lehigh University, 2007). In Fig. 1, the framework for the hybrid pseudo-dynamic test technique was applied in the Multi-site Online Simulation Test(MOST; The MOST Experiment, 2003). A structure in Fig. 1a is a two-bay single story steel frame. The structure is divided into two experimental substructures and one analytical substructure in Fig. 1b. Two experimental substructures are tested at the University of Illinois of Urbana-Champaign(UIUC) and the University of Colorado. One analytical substructure is simulated at the National Center for Supercomputing Applications(NCSA). The three substructures are integrated by the NEESgrid Teleoperations Control Protocol(NTCP) server, and the simulation coordinator provides the overall



(a) Two-bay single story steel frame



(b) Divided substructures



(c) Integration of substructures

Fig. 1 Hybrid structural testing
〔after Pearlman et al.(2004)〕

management of the experiment(Fig. 1c).

The information involved in the hybrid experiment shown in Fig. 1 is organized differently from the typical experiment. Fig. 2 shows the comparison of the typical and hybrid experiments. For the typical experiments, one typical experimental task involves a number of the test conditions. Each test condition involves a number of tests, and each test involves a data set. For the hybrid experiments, one hybrid experimental task involves a simulation coordinator, a number of physical substructures, and a number of analytical substructures. The simulation coordinator involves a number of simulations. Each physical substructure involves a number of physical substructure tests, and each analytical substructure a number of analytical computations. Each simulation involves a data set, each physical substructure test a data set, and each analytical substructure a data set. It is seen that a hybrid experimental task should have an organization of the simulation coordinator and the physical and analytical substructures.

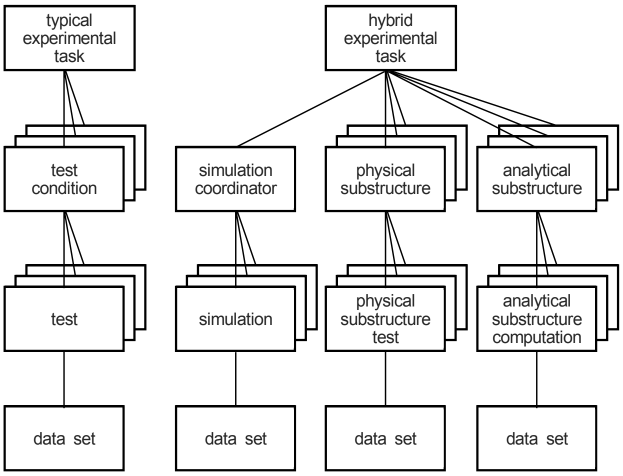


Fig. 2 Comparison of typical and hybrid experimental tasks

2.2 Interactions in Hybrid Experiments

Fig. 3 shows the interactions among a simulation coordinator, two physical substructures, and an analytical substructure. In the figure, the simulation coordinator involves two simulations, each physical substructure two tests, and the analytical substructure two computations. A simulation is related with one of the physical substructure tests and one of the analytical substructure computations. For example, simulation 1 is related with physical substructure 1 test 1, physical substructure 2 test 1, and analytical substructure computation 1. Simulation 1 sends the commands to and receives the feedbacks from physical substructure 1 test 1, physical substructure 2 test 1, and analytical substructure computation 1. These interactions require the proper links among simulation 1, physical substructure 1 test 1, physical substructure 2 test 1, and analytical substructure computation 1(Fig. 4). The links and other information are represented using the classes of the Lehigh Model.

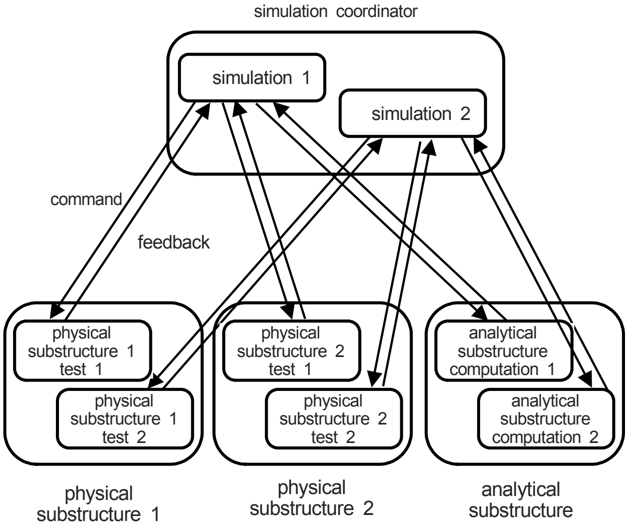


Fig. 3 Interactions in hybrid experiment

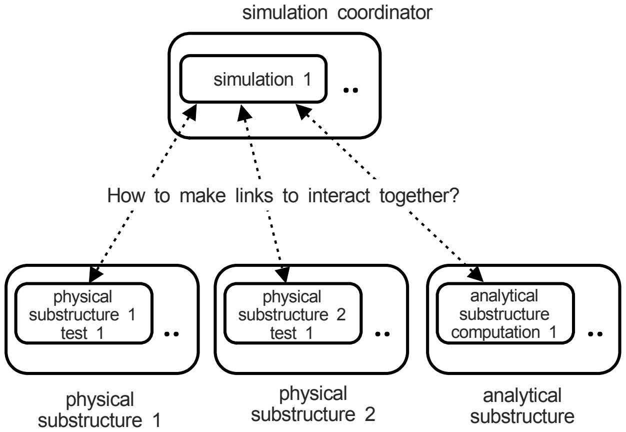


Fig. 4 Need for links in hybrid experiment

3. Class Hierarchy in Lehigh Model

The Lehigh Model(Lee et al., 2008) was developed to represent large-scale structural experiments including pseudo-dynamic tests and hybrid pseudo-dynamic tests. The model includes a number of classes and attributes. Fig. 5 shows the class hierarchy of the main classes of the model. The classes and attributes in the figure and later figures are represented using a modified entity-relationship diagram developed for entity-based product and process models(Hong et al., 1994; Lee et al., 1998). Each rectangle in Fig. 5 indicates an entity category(referred to as a class). Each attribute of the class is shown below the rectangle with a horizontal bar. If the attribute is singe-valued, the bar ends with an empty circle, and if the attribute is multi-valued, the bar ends with a black circle. The value set of an attribute(the set of possible values for the attribute) is represented in square brackets. The attribute type is identified in parentheses. Attributes of a class are classified into two main types : (1) "data-valued" attributes(DVA) whose values are alphanumeric or otherwise indecomposable; and (2) "object entity-valued" attributes(OEVA) whose values refer to other classes.

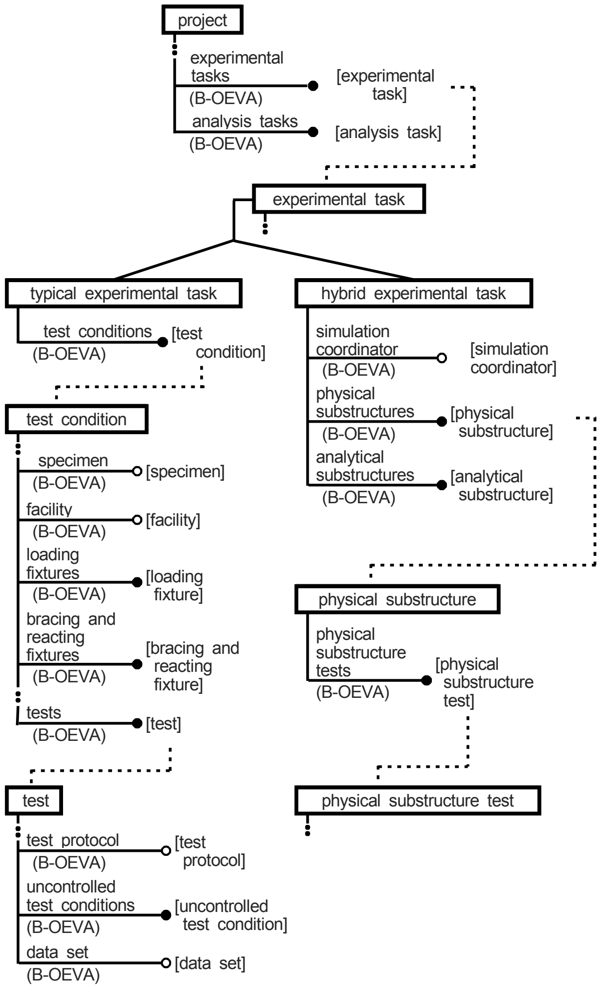


Fig. 5 Class hierarchy in Lehigh Model
(after Lee et al.(2008))

The class hierarchy of the Lehigh Model shown in Fig. 5 includes four main levels of classes : (1) the project class level, (2) the experimental task class level, (3) the test condition class level, and (4) the test class level. The project class at the top level represents the entire research project, and includes the attributes for the experimental tasks and the analysis tasks. The experimental task class at the second level is a generalization of the typical experimental task and the hybrid experimental task. The test condition class at the lower level provides the information on the specimen, the facility, and the setup of the specimen. The test class at the bottom level includes the data set.

4. Classes for Hybrid Experiments

The hybrid experimental task class shown in Fig. 5 represents the information involved in the hybrid experiments, and includes a single-valued attribute for the simulation coordinator, the multi-valued attribute for the physical substructure, and the multi-valued attribute for the analytical substructures. The value sets of the attributes refer to the simulation coordinator class, the physical substructure class, and the analytical substructure class in Fig. 6.

4.1 Classes for Simulation Coordinator

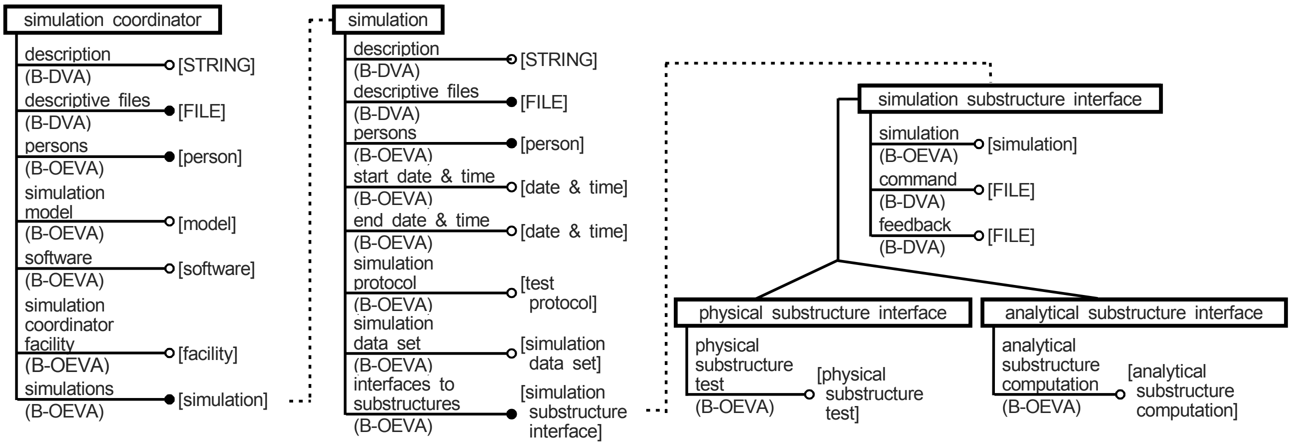
The simulation coordinator class in Fig. 6a represents the information involved in the simulation coordinator of a hybrid experiment. The simulation coordinator class includes the attributes for the simulation model and the simulation coordinator facility, and the multi-valued attribute for a number of the simulations. The value set of the simulations attribute refers to the simulation class. The simulation class includes the attributes for the start date & time, the end date & time, and the data set. The multi-valued attribute for the interfaces to substructures makes the links for the simulation with the physical and analytical substructures. The value set of the attribute refers to the simulation substructure interface class which is a generalization of the physical substructure interface class and the analytical substructure interface class which are used for the links with the physical substructures and the analytical substructures, respectively. The attributes for the simulation, the command, and the feedback of the simulation substructure interface class are inherited by the physical substructure interface class and the analytical substructure interface class.

4.2 Classes for Physical Substructure

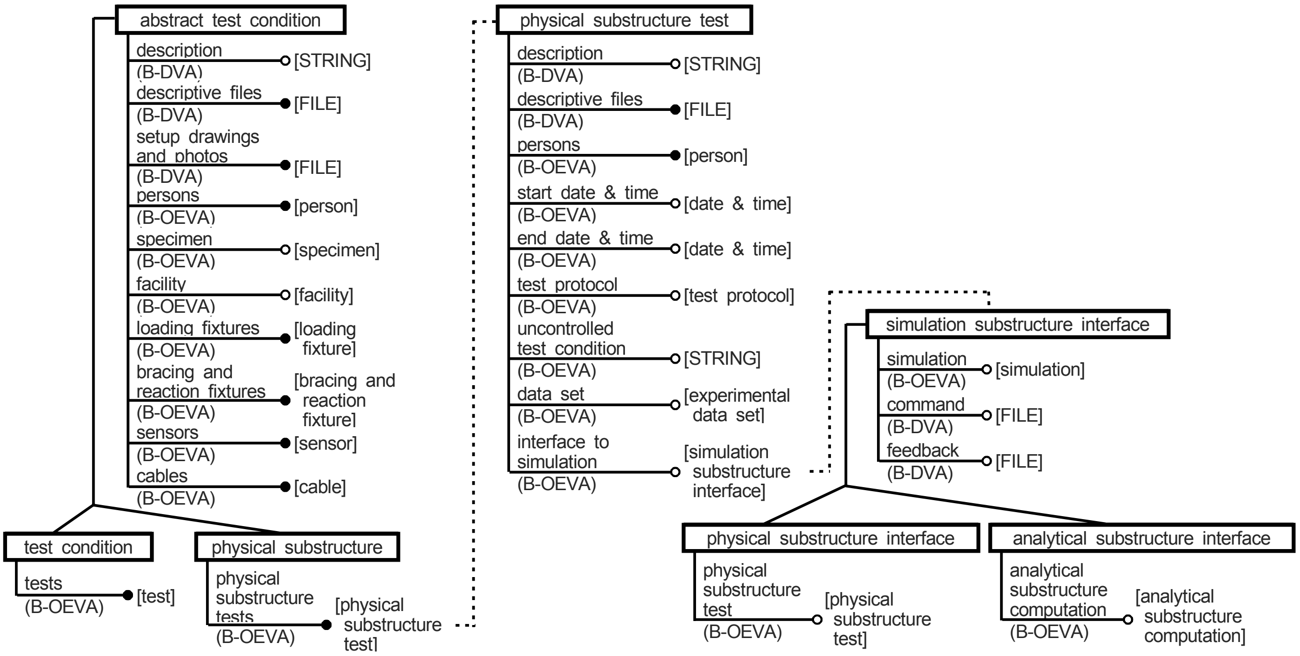
In Fig. 6b, the abstract test condition class is a generalization of the test condition class and the physical substructure class. The test condition class is for the typical experiments, and the physical substructure class for the hybrid experiments. The attributes for the specimen and the facility of the abstract test class are inherited by the test condition class and the physical substructure class. The value set of the multi-valued attribute for the physical substructure tests refers to the physical substructure test class. The physical substructure test class includes the attributes for the start date & time, the end date & time, and the data set. The single-valued attribute for the interface to simulation makes the link for the physical substructure with the simulation. The value set of the attribute refers to the simulation substructure interface class which is the same as in Fig. 6a.

4.3 Classes for Analytical Substructure

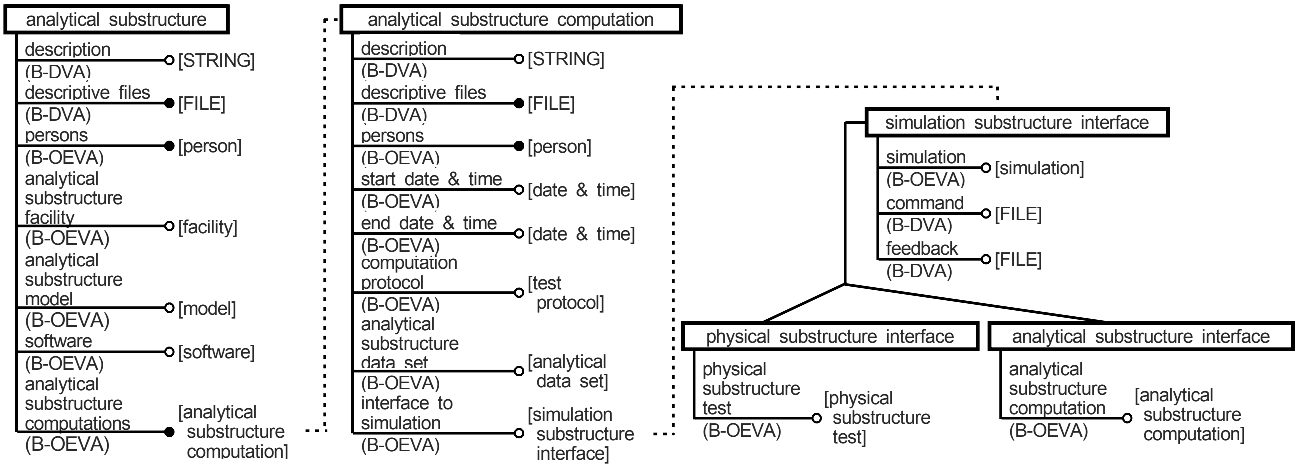
The analytical substructure class in Fig. 6c represents the information involved in the analytical substructures of a hybrid experiment. The analytical substructure class includes the attributes for the facility and the analytical substructure model, and the multi-valued attribute for a number of the analytical computations. The value set of the analytical substructure computations attribute refers to the analytical substructure computation class. The analytical substructure computation class includes the attributes for the start date & time, the end date & time, and the data set. The single-valued attribute for the interface to simulation makes the link for the analytical substructure with the simulation. The value set of the attribute refers to the simulation substructure interface class which is the same as in Figs. 6a and 6b.



(a) Classes for simulation coordinator



(b) Classes for physical substructure



(c) Classes for analytical substructure

Fig. 6 Classes for hybrid experiments

4.4 Organization of Classes for Hybrid Experiments

The classes for the simulation coordinator, the physical substructure, and the analytical substructure shown in Figs. 6a, 6b, and 6c indicate an underlying concept of the organization. A simulation and its related physical substructure tests and analytical substructure computations are linked through the simulation substructure interface class which is a generalization of the physical substructure interface class and the analytical substructure interface class. Linked through the same class, it allows a simple and easy understanding of the classes for the hybrid experiments.

5. Application Example

The classes for the hybrid experiments shown in Fig. 6 are applied to an example hybrid experiment performed at the Real-Time Multi-Directional(RTMD) testing facility at the ATLSS Center at Lehigh University. For the space limitation, only a small part of the hybrid experiment is presented in this section. The example structure is subject to the earthquake loading, and is manipulated by a simulation coordinator with one physical substructure and one analytical substructure. The objects for the example hybrid experiment are shown in Fig. 7, and the implementation of the objects is shown in Fig. 8.

5.1 Example Objects for Hybrid Experiment

Fig. 7 shows the objects for the example hybrid experiment. Each rounded rectangle in the figure

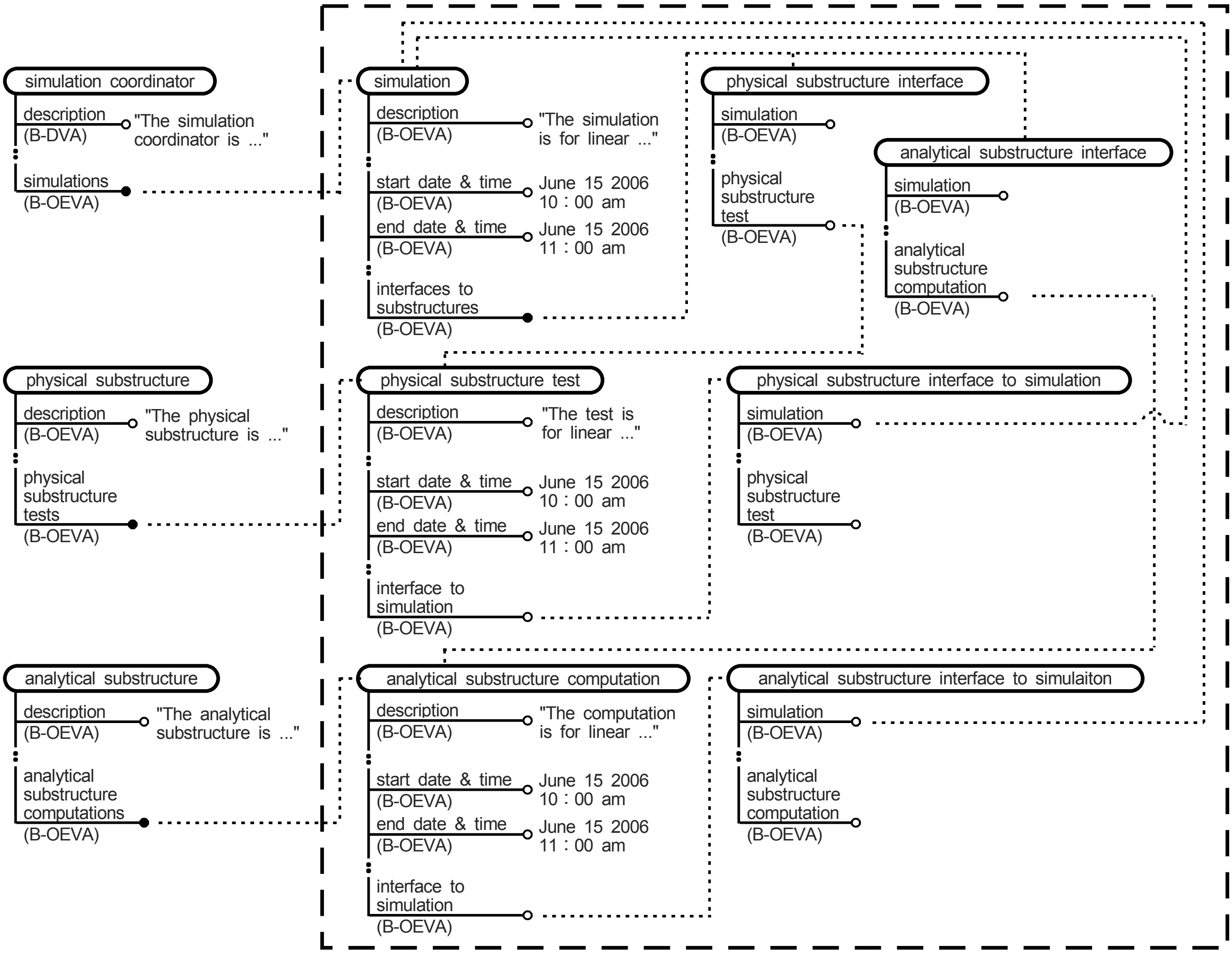
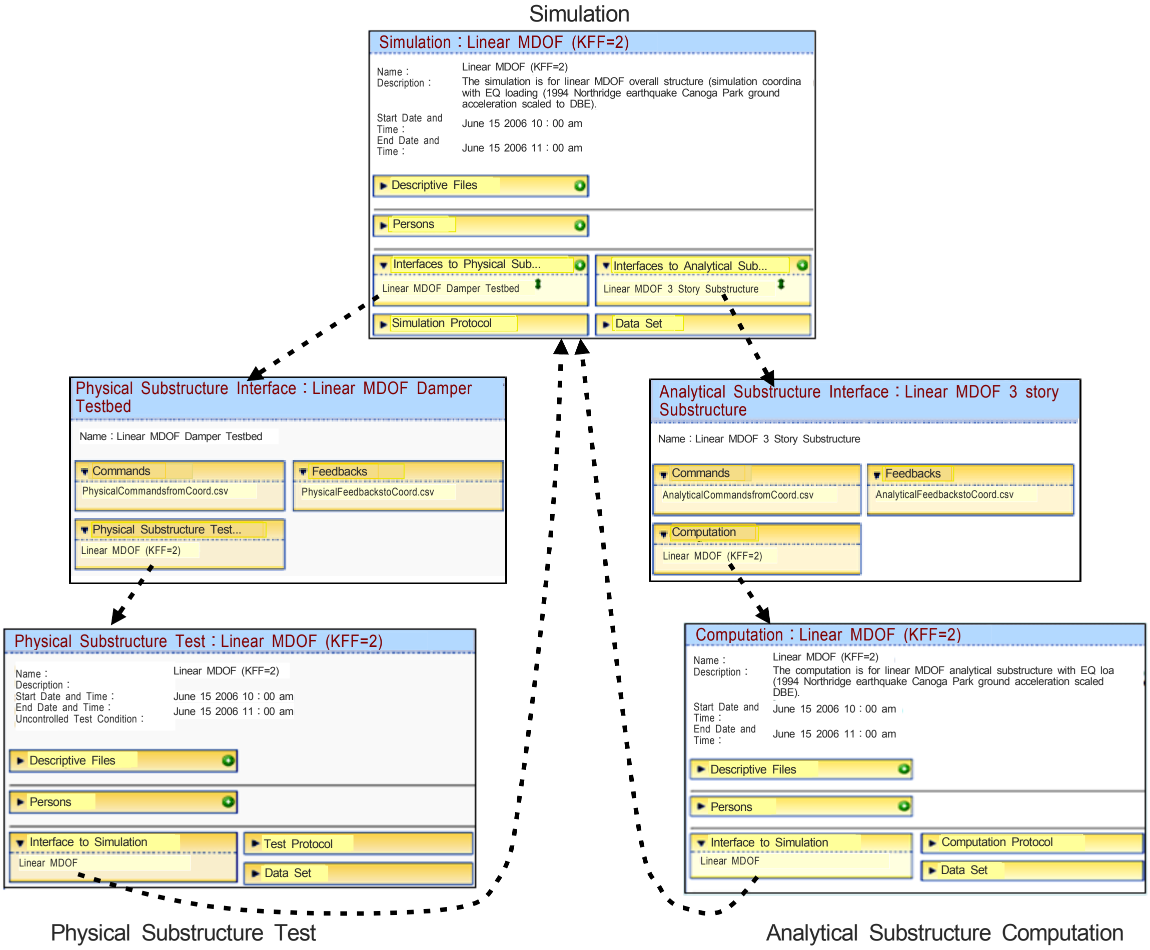


Fig. 7 Example objects for hybrid experiment



Name : Linear MDOF (KFF=2)
Description : The simulation is for linear MDOF overall structure (simulation coordina with EQ loading (1994 Northridge earthquake Canoga Park ground acceleration scaled to DBE).
Start Date and Time : June 15 2006 10 : 00 am
End Date and Time : June 15 2006 11 : 00 am

Name : Linear MDOF Damper Testbed

Name : Linear MDOF 3 Story Substructure

Name : Linear MDOF (KFF=2)
Description :
Start Date and Time : June 15 2006 10 : 00 am
End Date and Time : June 15 2006 11 : 00 am
Uncontrolled Test Condition :

Name : Linear MDOF (KFF=2)
Description : The computation is for linear MDOF analytical substructure with EQ loa (1994 Northridge earthquake Canoga Park ground acceleration scaled DBE).
Start Date and Time : June 15 2006 10 : 00 am
End Date and Time : June 15 2006 11 : 00 am

Fig. 8 Implementation of objects for hybrid experiment 〔after Marullo (2007)〕

denotes an object. The simulation coordinator object in Fig. 7 is from the simulation coordinator class in Fig. 6a. Similarly, the physical substructure object and the analytical substructure object in Fig. 7 are from the physical substructure class in Fig. 6b and the analytical substructure class in Fig. 6c. This applies to the other objects in Fig. 7.

The simulation coordinator object in Fig. 7 includes the simulations attribute whose value refers to the simulation object. The simulation object includes the attributes for the start date & time and the end date & time. The simulation object also includes the attribute for the interfaces to substructures whose values refer to the physical substructure interface object and the analytical substructure interface object. The physical substructure interface object is linked with the physical substructure test object, and the analytical substructure interface object is linked with the analytical substructure computation object.

The physical substructure object in Fig. 7 includes the attribute for the physical substructure tests whose value refers to the physical substructure test object. The physical substructure test object includes the attributes for the start date & time and the end date & time. The physical substructure test object also includes the attribute for the interface to simulation whose value refers to the physical substructure interface to simulation object. The physical substructure interface to simulation object is linked with the simulation object.

The analytical substructure object and it′s lower level objects in Fig. 7 are similarly formulated. The analytical substructure interface to simulation object is linked with the simulation object.

The simulation object, the physical substructure test object, and the analytical substructure computation object in Fig. 7 are linked together and have the same start date & time and the same end date & time. The simulation object, the physical substructure test object, the analytical substructure computation object, and their lower level objects, enclosed by the dotted-line box, are implemented in Fig. 8.

5.2 Implementation of Objects

The example hybrid experiment performed at Lehigh University was implemented as an internet application backed by a database system. A Red Hat Linux server running Apache provides the framework for the software. The software package is being developed by integrating PHP, JavaScript, XML, and MySQL code. XML and MySQL provides the structure of the data and JavaScript dynamically create HTML that is presented to the user through a web browser.

Fig. 8 shows some of the displays of the implemented system(Marullo, 2007) for the objects enclosed by the dotted-line box in Fig. 7. The implementation by Marullo was originally for the Lehigh Model(Lee et al., 2008), and the implementation shown in Fig. 8 is particularly for the hybrid experiments which were revised from the Lehigh Model. The simulation object in Fig. 7 is shown as the "Simulation" window in Fig. 8. The physical substructure interface object and the analytical substructure interface object in Fig. 7 are shown as the "Physical Substructure Interface" window and the "Analytical Substructure Interface" window in Fig. 8. The physical substructure test object and the analytical substructure computation object in Fig. 7 are shown as the "Physical Substructure Test" window and the "Computation" window in Fig. 8.

The "Simulation" window in Fig. 8 includes the name and description items and other items. The "Interfaces to Physical Sub...(Substructures)" item leads to the "Physical Substructure Interface" window. The "Physical Substructure Interface" window includes the "Physical Substructure Test..(Tests)" item which leads to the "Physical Substructure Test" window. The "Physical Substructure Test" includes the "Interface to Simulation" item which leads to the "Simulation" window. Similarly, the "Interfaces to Analytical Sub...(Substructures)" item of the "Simulation" window leads to the "Analytical Substructure Interface" window, which is linked with the "Computation" window. The "Computation" window is linked with the "Simulation" window.

6. Discussion and Conclusions

The hybrid approach for structural experiments is useful for the hybrid pseudo-dynamic tests that decompose a structure into independent substructures, and overcome the limitations of a test structure (Dermitzakis et al., 1985; Pearlman et al., 2004). Based on the hybrid approach, a hybrid experimental task involves a number of physical substructures and a number of analytical substructures that are managed by a simulation coordinator. This paper has presented the representation of the information involved in the hybrid experiments, using the classes and attributes of the Lehigh Model(Lee et al, 2008).

The hybrid experimental task class includes a single-valued attribute for the simulation coordinator, a multi-valued attribute for the physical substructure, and a multi-valued attribute for the analytical substructure. The simulation coordinator class includes the multi-valued attribute for the simulation, the physical substructure class the multi-valued attribute for the physical substructure test, and the analytical substructure class the multi-valued attribute for the analytical substructure computation. The simulation class, the physical substructure test class, and the analytical substructure computation class are linked through the simulation substructure interface class. These classes for the hybrid experiments are applied to an example hybrid experiment. The objects for interactions among a simulation, a physical substructure test, and an analytical substructure computation are demonstrated and implemented.

Compared with previous data models, the data model presented in this paper has the following contributions and benefits :

(1) The hybrid experimental task class is developed

as a specialization of the experimental task class which also includes the typical experimental task class(Fig. 5). The physical substructure class is developed as a specialization of the abstract test condition class which also includes the test condition class(Fig. 6). The classes for the hybrid experiments are closely related with those for the typical experiments in one data model.

(2) The hybrid experimental task includes the simulation coordinator at a site, the physical substructures at multiple sites, and the analytical substructures at multiple sites(Fig. 5). The use of common attributes(Fig. 6) helps communications among the simulation coordinator, physical substructures, and analytical substructures.

(3) The simulation class, the physical substructure tests, and the analytical substructure computation class are linked together through the simulation substructure interface class(Fig. 6). Use of the same class simplifies the representation of the links.

(4) The simulation class, the physical substructure tests, and the analytical substructure computation class have the direct and cross-links(Figs. 6, 7, and 8). For the simulation, the related physical and analytical substructures can be found in the simulation substructure interface class. For the physical and analytical substructures, the related simulation can be also found in the simulation substructure interface class.

(5) The simulation interface class includes the attributes for the commands and the feedbacks that can hold the information of interactions among the simulation, the physical substructure test, and the analytical substructure computation (Figs. 6 and 8).

The data model in this paper have provided the main classes and attributes for the hybrid experiments, but additional research is required to refine the classes and attributes for practical use. The attributes for the simulation coordinator, the physical substructure, and the analytical substructure need to be described in detail. The details of the commands and feedbacks among the simulation, the physical substructure test, and the analytical substructure computation also need to be described in detail. Other ways of the links for the simulation, the physical substructure tests, and the analytical substructure computations can be possibly considered.

The data model presented in this paper is believed to have provided the basic but essential descriptions of the data model for the hybrid experiments that other data models have not explicitly addressed. The concepts can be applied into other data models for structural experiments. The implementation for example objects can be expanded for a more practical system that helps researchers access, share, and use the information on the hybrid structural experiments.

Acknowledgements

This paper is based upon work supported by the National Science Foundation under Grant No. CMS-0402490 (NEES Consortium Operation), within the George E. Brown, Jr. Network for Earthquake Simulation (NEES) program. The authors are grateful to Professor James M. Ricles, Lehigh University, for his assistance with this work. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Dermitzakis, S.N., Mahin, S.A. (1985) Development of Substructuring Techniques for On-Line Computer Controlled Seismic Performance Testing, *Report UBC/EERC-85/04*, Earthquake Engineering Research Center, University of California, USA.

Hong, N.K., Sause, R. (1994) Concepts and Notation for Integrated Structural Design:Product and Process Models, *ATLSS Report No. 94-13*, Advanced Technology for Large Structural Systems(ATLSS) Center, Lehigh University, Bethlehem, PA, USA.

http://nees.org/ (2009) Website for George E. Brown, Jr. Network for Earthquake Engineering

Simulation(NEES), USA.

http://nisee.berkely.edu/spd/ (2004) Website for PEER Structural Performance Database, Pacific Earthquake Engineering Research Center, Berkeley, CA. USA.

http://www.sacsteel.org/ (2000) SAC Steel Project Website, SAC Steel Project, Richmond, CA, USA.

Lee, C.H., Chin, C.H., Marullo, T., Bryan, P., Sause, R., Ricles, J.M. (2008) Data Model for Large-Scale Structural Experiments, *Journal of Earthquake Engineering*, 12, pp.115~135.

Lee, C.H., Sause, R., Hong, N.K. (1998) Overview of Entity-Based Integrated Design Product and Process Models, *Advances in Engineering Software*, 29(10), pp.809~823.

Lehigh University (2007) *Real-Time Multi-Directional(RTMD) Earthquake Simulation Facility User's Guide*, Lehigh University, Bethlehem, PA. USA.

Marullo, T. (2007) A Data Model for Large-Scale Structural Laboratory Experiments Developed at the Real-Time Multi-Directional(RTMD) Testing Facility at the ATLSS Center at Lehigh University, *Fifth NEES Annual Meeting*, Snowbird, Utah, USA.

NEESit (2007) *NEEScentral User's Guide 1.7*, George E. Brown, Jr. Network for Earthquake Engineering Simulation(NEES), USA.

Oregon Sate University and Network Alliance for Computational Science and Engineering (2003) *NEES Database and Metadata Structure, Version 1.3*, white paper, George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES), USA.

Pearlman, L., Kesselman, C., Gulapalli, S., Spencer, B.F., Futrelle, J., Ricker, K., Foster, I., Hubbard, P., Severance, C. (2004) Distributed Hybrid Earthquake Engineering Experiments: Experiences with a Ground-Shaking Grid Application, *Proc. 13th IEEE Symposium on High Performance Distributed Computing(HPDC-13)*.

Peng, J., Law, K. (2004) Reference NEESgrid Data Model, *Technical Report NEESgrid-2004-40*, George E. Brown, Jr. Network for Earthquake Engineering Simulation(NEES), USA.

The MOST Experiment (2003) *Technical Report*, George E. Brown, Jr. Network for Earthquake Engineering Simulation(NEES), USA. (http://it.nees.org/documentation/pdf/MOST_document_v_1.0.pdf).