



*Enabling the Network for  
Earthquake Engineering Simulation*



QA-2005-000A

# Data Retrieval Tools

## Test Plan

Jun Peng and Kincho H. Law

Stanford University

Last Modified: 2005-07-21 Version: 1.0

## Table of Contents

1	About This Test .....	4
1.1	Overview (Test Goals) .....	4
1.2	Components Covered by This Test Plan .....	4
1.3	Test Environment and Conditions .....	4
1.4	Reference, Definitions and Acronyms.....	4
2	Approach .....	5
2.1	Types of Testing .....	5
2.2	Test objectives.....	5
2.3	Performance .....	5
3	Features to Be Tested and Not to Be Tested .....	5
3.1	Features to Be Tested .....	5
3.2	Features Not to Be Tested .....	5
4	Test Execution .....	6
4.1	Test Configurations.....	6
4.2	Test Data .....	8
4.3	Test Cases .....	8
4.4	Test Reports.....	9



## Table of Figures

Figure 1 – The Main Page of the Data Browser .....	9
Figure 2 – Detailed Display of the Project MiniMOST .....	10
Figure 3 – Detailed Display of the Task miniMOST_at_UIUC .....	10
Figure 4 – Detailed Display of the InfrastructureSetup .....	11
Figure 5 – Access of Files Showing the InfrastructureSetup .....	11
Figure 6 – Detailed Display of the Task miniMOST_at_UIUC .....	12
Figure 7 – Detailed Display of the EventGroup .....	12
Figure 8 – Detailed Display of the Event miniMOST_test_0228 .....	13
Figure 9 – Access of SensorReading for the Event miniMOST_test_0228 .....	13
Figure 10 – Main Page to Retrieve MiniMOST Data Saved in MySQL .....	14
Figure 11 – Display of “Summary” for MiniMOST Experiment .....	14
Figure 12 – Display of “Instrumentation” for MiniMOST Experiment .....	15
Figure 13 – Main Page to Retrieve Composite Frame Test Data Saved in OWL .....	16
Figure 14 – Display of “Specimen” for the Composite Frame Test .....	16
Figure 15 – Access of Drawings for the “Specimen” .....	17
Figure 16 – Display of “Loading System” for the Composite Frame Test .....	17
Figure 17 – Access of Drawings for the “Loading System” .....	18
Figure 18 – Main Page of the Report Generator .....	19
Figure 19 – Main Interface of Report Generator for MiniMOST Experiment .....	19
Figure 20 – Access of “Specimen” for MiniMOST Data Saved in OWL .....	20
Figure 21 – Access of “Specimen” for MiniMOST Data Saved in MySQL .....	20
Figure 22 – Main Interface of Report Generator for the Composite Frame Test .....	21
Figure 23 – Access of “Infrastructure” Data for the Composite Frame Test .....	21
Figure 24 – Access of “Loading” Data for the Composite Frame Test .....	22
Figure 25 – Access of “Instrumentation” Data for the Composite Frame Test .....	22



# 1 About This Test

## 1.1 Overview (Test Goals)

The goal of this test is for testing the functionality and usability of the developed data retrieval tools. The tools are designed to retrieve project data according to a reference data model developed for shake table and large-scale structural tests (see report *NEESgrid-2004-40: Reference NEESgrid Data Model*). The web interfaces to the data retrieval tools will be the main testing environment where the functionality of the data retrieval tools can be observed. The test results (observed behavior of software tools) can be compared with the expected behavior, which is specified in the report *Data Retrieval Tools Software Design Specification*. The data retrieval tools can pass this test only when the observed behavior matches the expected behavior.

## 1.2 Components Covered by This Test Plan

This test focuses on testing the functionalities of the following software components that are developed to browse and retrieve data related to a project:

- **Project browser** – developed to browse the data and metadata that are related to a particular project.
- **Data retrieval tool** – developed to facilitate retrieving related data from the data repository and to present the data to the users according to a pre-defined file hierarchy.
- **Report generator** – implemented to help organizing project data according to a file hierarchy specified in a report structure.

## 1.3 Test Environment and Conditions

The following external tools and libraries are employed for the execution of these data retrieval tools:

- Jena API to read, write and manipulate OWL files.
- MySQL database to store data in relational form.
- MySQL Connector J/3.0 (JDBC implementation) to connect Java programs with the MySQL database.
- Java Servlet for retrieving data from the data repository and dynamically generating web pages.

## 1.4 Reference, Definitions and Acronyms

- **OWL** – Web Ontology Language. OWL is employed to explicitly represent the meaning of terms and the relationships between those terms defined in the



reference data model (*NEESgrid-2004-40: Reference NEESgrid Data Model*). OWL is represented in XML format. <http://www.w3.org/2001/sw/WebOnt/>.

- **Jena** – A Semantic Web Framework for Java. Jena provides a programmatic environment for OWL. By using Jena, OWL documents can be parsed and interpreted. <http://jena.sourceforge.net/>.
- **JDBC** – Java Database Connectivity. JDBC is part of the Java Development Kit (JDK) that defines an API (application program interface) to support standard SQL access to databases from Java programs. <http://java.sun.com/products/jdbc/>.

## 2 Approach

### 2.1 Types of Testing

The usability and functionality of the developed data retrieval tools can be tested without prior knowledge of the internal implementation. After a data server is setup and configured, the data retrieval tools can be accessed via a web interface. Manual testing can be carried out directly using the web interface to browse and retrieve the project data.

### 2.2 Test objectives

The data retrieval tools need to be tested to verify (1) if the functionalities of the data retrieval tools match the specification; (2) if there is any obvious glitch or broken link; (3) how the data retrieval tools are to be used.

### 2.3 Performance

System performance will not be tested for current version of the implementation.

## 3 Features to Be Tested and Not to Be Tested

### 3.1 Features to Be Tested

The features to be tested include the project browser, a set of data retrieval tools, and the report generator.

### 3.2 Features Not to Be Tested

As of this writing, the design of the data repository for NEES has yet to be finalized. Therefore, the developed reference data model has not been integrated with the data repository and the data retrieval tools cannot access data from the repository. Instead, a file system and/or a MySQL database system are used to serve as data storage mechanism for the developed data retrieval tools. The integration with NEES data repository will not be tested.



## 4 Test Execution

### 4.1 Test Configurations

This section describes the setup and the configurations of the external tools and libraries that are listed in Section 1.3. Please be noted that the following installation instructions are for Windows 2000 or Windows XP systems only. The steps to create or change Windows **Environment Variables** are:

1. Right-click on 'My Computer' at the desktop.
2. Click on 'Properties' and then choose 'Advanced' from the top menu bar.
3. Click on the button 'Environment Variables...'
4. There are two windows: one is for 'System Variables' and the other is for "User variables'. Normally we want to change 'System Variables'.
5. If we want to create a new environment variable, for example "%CLASSPATH% = C:\java", we can click on the button 'New...'. On the pop-up window, enter "CLASSPATH" to the text bar named 'Variable Name' and "C:\java" to the text bar named 'Variable Value'.
6. If we want to change an existing environment variable, for example adding "D:\Tomcat" to the variable "%PATH%", we can double-clicking the variable "PATH" from the 'System Variables' window. Then on the pop-up window, add "D:\Tomat" to the text bar named 'Variable Value'.

The following lists the steps to install and configure **JDK** and **Java Servlet**:

1. Installation and configuration for **JDK** (Java Development Kit)
  - 1.1. Download a free JDK package from <http://java.sun.com/>.
  - 1.2. Install the downloaded JDK package to your preferred location, such as C:\Program Files\Java.
  - 1.3. Change the Windows environment variable %JAVA\_HOME% to reflect the location where JDK package is installed, such as 'C:\Program Files\Java'.
2. Installation and configuration for Apache Tomcat **Java Servlet**
  - 2.1. Download a free Java Servlet package from <http://jakarta.apache.org/tomcat/>
  - 2.2. Install the downloaded package to a location such as  
%TOMCAT% = C:\Program Files\Apache Group\Tomcat4.0\
  - 2.3. Change the %CLASSPATH% environment to include:  
CLASSPATH = C:\Program Files\Apache Group\Tomcat4.0\common\lib\servlet.jar
  - 2.4. Change the configuration file '%TOMCAT%\bin\setclasspath.bat' by updating the line: 'set CLASSPATH=%CLASSPATH%;%JAVA\_HOME%\lib\tools.jar'

If **OWL files** are to be employed as data storage, the following lists the steps to install and configure **Jena**:

1. Download a free Jena package from <http://jena.sourceforge.net/>.



2. Unzip and install the downloaded Jena package to your preferred location, such as C:\Program Files\Jena.
3. Set a Windows environment variable %JENAROOT% to reflect the location where Jena package is installed, such as C:\Program Files\Jena.
4. Set the Java classpath (Windows environment variable %CLASSPATH%) to include all the JAR files in the %JENAROOT%/lib directory
  - JENAROOT\lib\jena.jar
  - JENAROOT\lib\xercesImpl.jar
  - JENAROOT\lib\xml-apis.jar
  - JENAROOT\lib\icu4j.jar
  - JENAROOT\lib\concurrent.jar
  - JENAROOT\lib\jakarta-oro-2.0.5.jar
  - JENAROOT\lib\antlr.jar
  - JENAROOT\lib\junit.jar
  - JENAROOT\lib\commons-logging.jar
  - JENAROOT\lib\log4j-1.2.7.jar
  - JENAROOT\lib\rdf-api-2001-01-19.jar
5. Test the Jena installation by running "test.bar" on a Windows Command Prompt.

If MySQL database is to be employed as data storage, MySQL database can be downloaded, installed and configured as follows:

#### 1. Installation and configuration for **MySQL** Database

- 1.1. Download a free MySQL database system from <http://www.mysql.com/>, the preferred version is 4.1.
- 1.2. Install the downloaded MySQL package to your preferred location, such as 'C:\Program Files\mysql'.
- 1.3. Setup password for the default user 'root'.
 

```
mysql> SET PASSWORD FOR root@localhost=PASSWORD('new_password');
```
- 1.4. Delete the default user " " for security reason:
 

```
mysql> DELETE FROM user WHERE Host='localhost' AND User="";
```
- 1.5. Create a new username/password and grant all privileges:
 

```
mysql> GRANT ALL PRIVILEGES ON *.* TO username@localhost IDENTIFIED BY 'password' WITH GRANT OPTION;
```

#### 2. Installation and configuration for **MySQL Connector J/3.0** (JDBC)

- 2.1. Download a free JDBC package from <http://www.mysql.com/>.
- 2.2. Unzip the downloaded package to your preferred location, such as 'D:\jdbc'.
- 2.3. Set the Java classpath (%CLASSPATH%) to include the JAR file.  
For example: CLASSPATH = %CLASSPATH%;D:\jdbc\mysql\_jdbc\mysql-connector-java-2.0.14-bin.jar



## 4.2 Test Data

The data from two pre-recorded earthquake experiments are used in this test. These two experiments are a miniMOST experiment at University of Illinois at Urbana-Champaign and a composite structural frame test at University of Minnesota:

**MiniMOST @ UIUC:** The miniMOST experiment equipment was designed to show the capability of the major service components for the NEESgrid using a small-scale physical experiment setup. An experiment has been carried out at UIUC by using miniMOST equipment.

**Composite Frame Test @ U of Minnesota:** The primary portion of this experiment is a cyclic loading test on one two-story, one-bay specimen, which represents the bottom two stories of a six-story prototype structure at approximately one-third scale. The steel frame was comprised of W5x19 wide flange columns and W8x13 wide flange beams. The columns and beams were joined through partially-restrained connections, which are comprised of top and seat angles and double web angles.

The data from these two experiments are ingested according to the developed reference data model (see report *NEESgrid-2004-40: Reference NEESgrid Data Model*). For the object-oriented model, data are ingested using Protégé, which is an open-source software for ontology development (see <http://protege.stanford.edu>). In the two examples, the metadata are saved in two files named **neesmeta\_1.2.pprj** and **neesmeta\_1.2.owl**.

For the relational data model, SQL scripts are used to create database schema and to ingest data into the MySQL database. Examples of SQL script can be found in **create.sql**, which is used for creating database schema, and **ingestion.sql**, which is used for ingesting miniMOST experiment data. Please be noted that all the aforementioned files are included in the source code release. These files can also be downloaded from <http://eil.stanford.edu/neesgrid/>.

## 4.3 Test Cases

The developed data retrieval tools can access data from both experiments that are saved in either OWL files or MySQL database. In this test, we will conduct and report the results for the test cases that are marked in the following table:

	Project Browser	Data Retrieval Tool	Report Generator
miniMOST / OWL	XX		XX
miniMOST / MySQL		XX	XX
Composite / OWL		XX	XX





## 4.4 Test Reports

The following reports the results of the test cases. These tests are conducted via a web browser, and selected screen shots are presented.

### 1. Project Browser / miniMOST / OWL

Figure 1 shows the main page of the project browser with a list of saved projects. When we click on a particular project, say miniMOST-1, the details of the project will be shown on the browser, as illustrated in Figure 2.

As defined in the reference data model, a Project is a collection (organized group) of Tasks designed to achieve specific goals and objectives. Following the model, we can navigate and access all the Tasks that belong to the Project. Figure 3 shows the details of a particular Task named miniMOST\_at\_UIUC. One property (or a slot) of a Task object is InfrastructureSetup, which models the assembly and arrangement of the PrimaryEquipment used for a specific Task. We can access the details of the InfrastructureSetup object by clicking on the highlighted button as shown in Figure 3.

Figure 4 shows the details of the InfrastructureSetup, which is a collection of text descriptions, documents (in the format of Word, PDF, Excel, etc.), figures and drawings stored as files. Files are saved in a web server and their URIs are saved as metadata. The files can be dynamically downloaded and shown on a web browser, as illustrated in Figure 5.

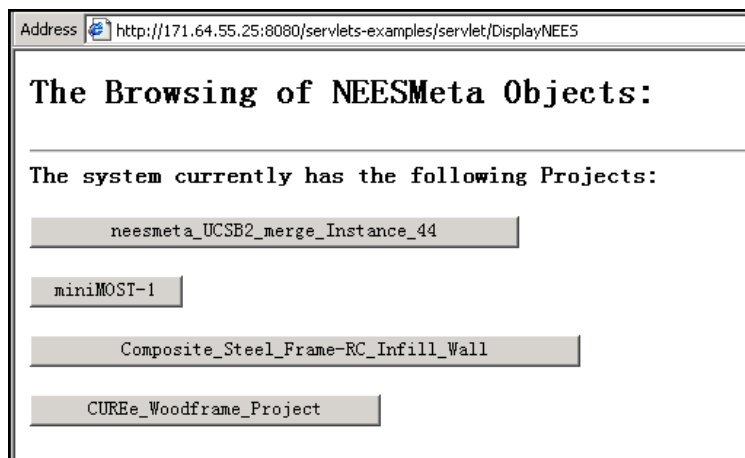


Figure 1 – The Main Page of the Data Browser



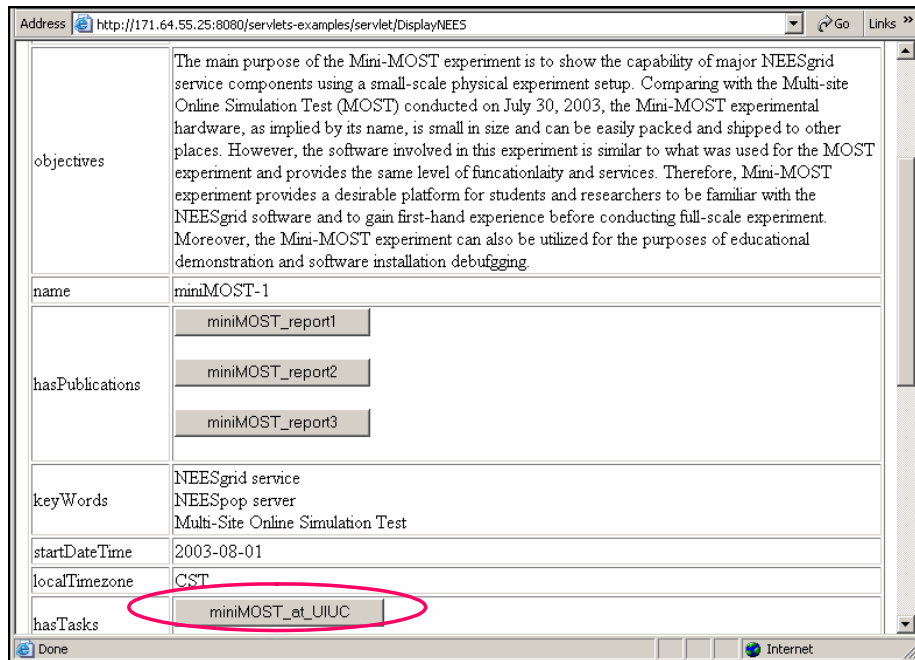


Figure 2 – Detailed Display of the Project MiniMOST

Each Task in a project may contain one or more EventGroups. The EventGroup object can be accessed by clicking the highlighted button shown in Figure 6. The details of a particular EventGroup object named miniMOST\_UIUC\_EventGroup\_2004 are shown in Figure 7.

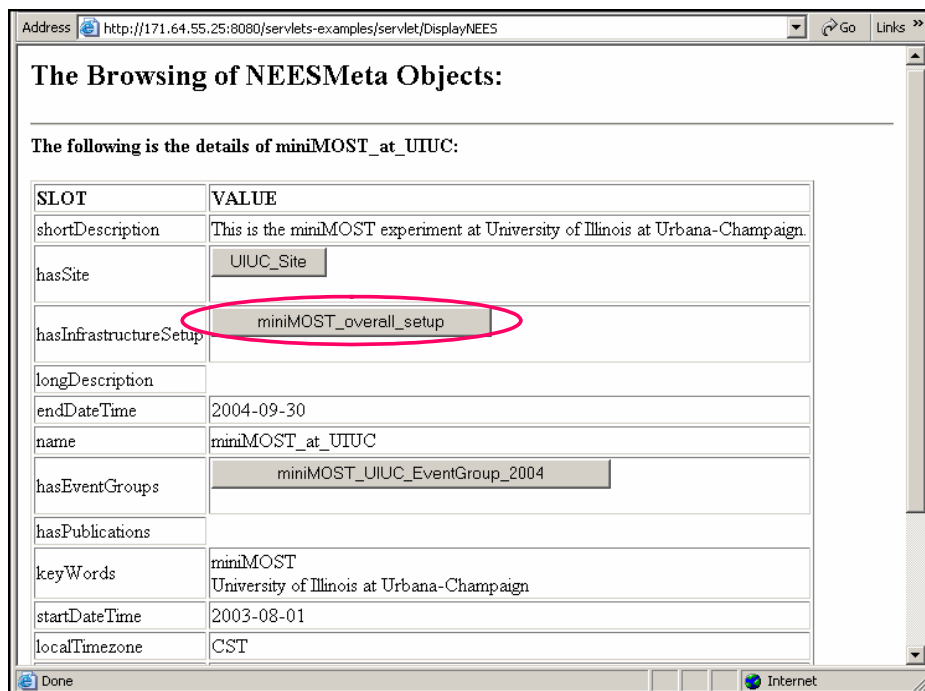


Figure 3 – Detailed Display of the Task miniMOST\_at\_UIUC



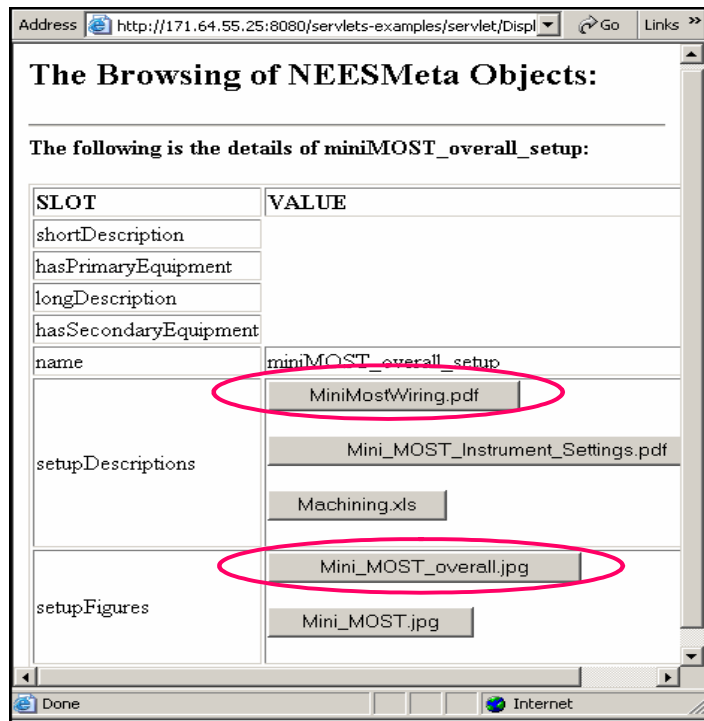
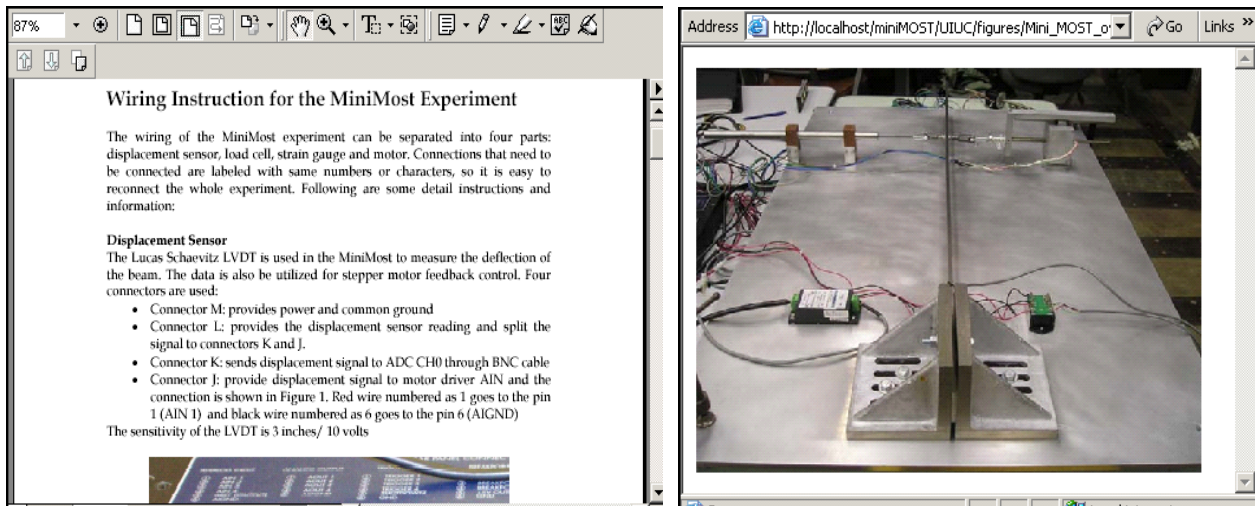


Figure 4 – Detailed Display of the InfrastructureSetup



(a) MiniMostWiring.pdf

(b) Mini\_MOST\_overall.jpg

Figure 5 – Access of Files Showing the InfrastructureSetup



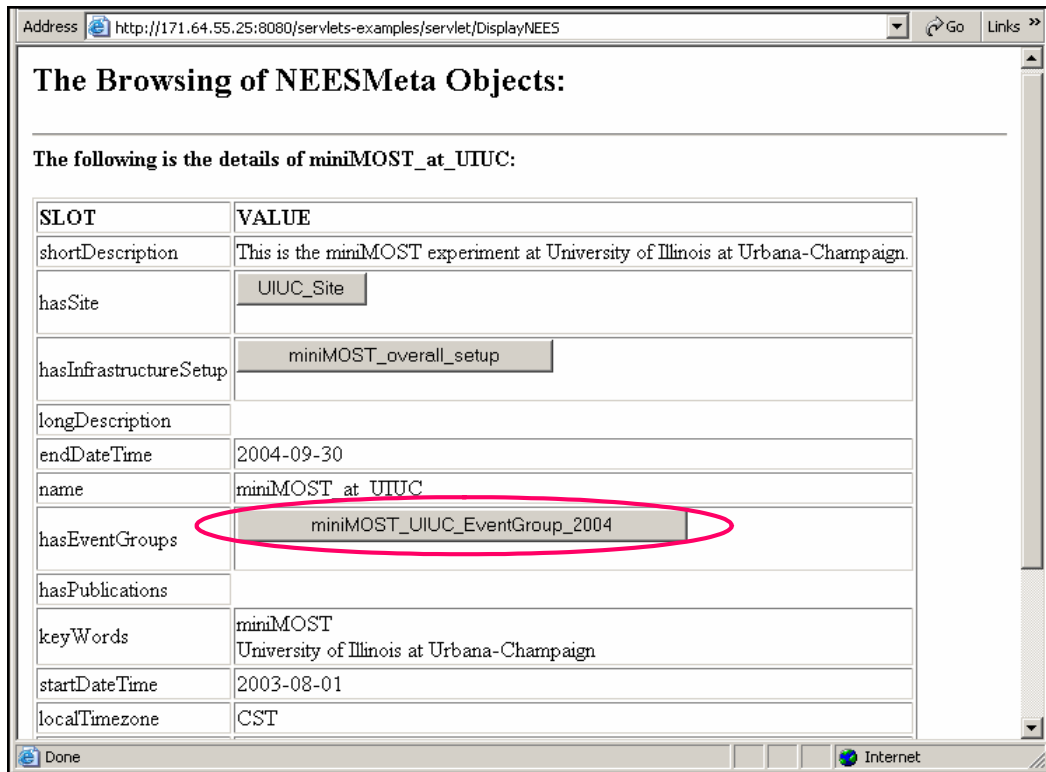


Figure 6 – Detailed Display of the Task miniMOST\_at\_UIUC

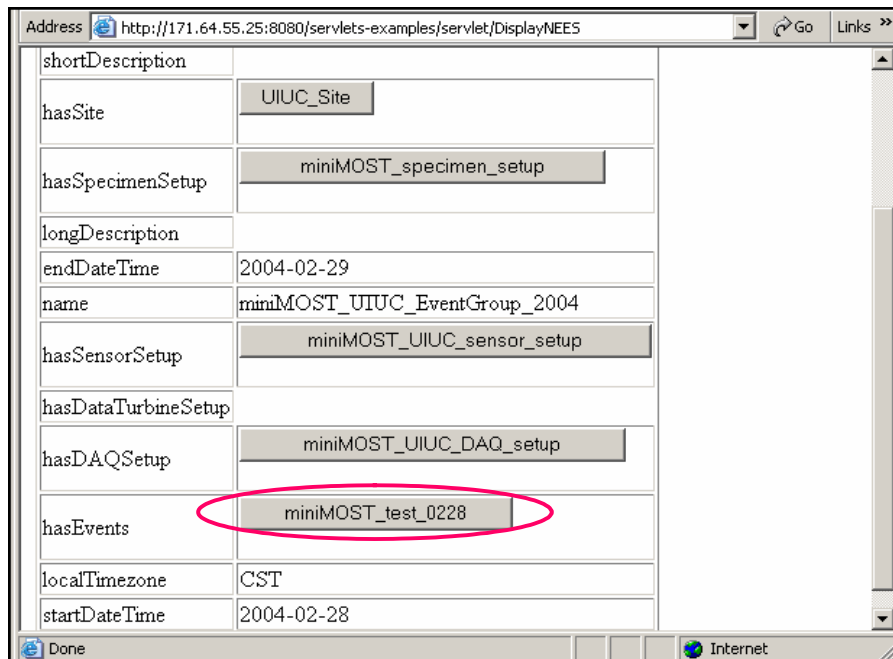


Figure 7 – Detailed Display of the EventGroup



An EventGroup is defined as a collection of Events, each of which can be accessed from the EventGroup object. The details of an Event named miniMOST\_test\_0228 are shown in Figure 8. An Event, which is the atomic level of an Activity, refers to a single run of an experiment or a simulation. Experimental results, such as SensorReading, can be accessed from an Event object, as shown in Figure 9.

SLOT	VALUE
shortDescription	An event with one-acurator miniMOST setup.
hasSite	UIUC_Site
testType	pseudo dynamic
hasWaveFormSetup	
longDescription	
endDateTime	2004-02-29T02:11:14.78099
name	miniMOST test 0228
hasOutputData	miniMOST_test_0228_results
startDateTime	2004-02-28T20:15:49.57800
loadTime	1.035167

Figure 8 – Detailed Display of the Event miniMOST\_test\_0228

```

Active channels: LVDT, StrainGage, LoadCell
Channel units: m,microstrain,N

Time      LVDT      StrainGage      LoadCell
2004-02-28T20:15:49.57800      -0.000072      -20.202637      0.737591
2004-02-28T20:15:58.03099      -0.000063      -17.211914      0.836224
2004-02-28T20:16:05.59299      -0.000053      -18.432617      0.840723
2004-02-28T20:16:13.01499      -0.000077      -17.395020      0.836664
2004-02-28T20:16:20.43699      -0.000060      -17.456055      0.734262
2004-02-28T20:16:28.12500      -0.000063      -17.456055      0.836754
2004-02-28T20:16:36.76499      -0.000060      -17.944336      0.735809
2004-02-28T20:16:44.31199      -0.000086      -20.080566      0.838325
2004-02-28T20:16:52.07800      -0.000079      -17.028809      0.838468
2004-02-28T20:16:59.50000      -0.000098      -18.920898      0.933884
2004-02-28T20:17:06.98399      -0.000107      -16.296387      0.928718
2004-02-28T20:17:14.51499      -0.000121      -18.920898      1.034670
2004-02-28T20:17:21.90599      -0.000126      -15.747070      1.031043
2004-02-28T20:17:29.31199      -0.000147      -16.540527      1.036738
2004-02-28T20:17:36.75000      -0.000144      -17.578125      1.031765
2004-02-28T20:17:44.28099      -0.000149      -15.136719      1.035167
2004-02-28T20:17:52.71799      -0.000135      -15.319824      1.038448
2004-02-28T20:18:00.13999      -0.000147      -17.333984      1.035462
2004-02-28T20:18:07.56199      -0.000102      -17.150879      0.932433
2004-02-28T20:18:15.07800      -0.000058      -19.042969      0.836102
    
```

Figure 9 – Access of SensorReading for the Event miniMOST\_test\_0228



## 2. Data Retrieval Tool / miniMOST / MySQL

Figure 10 shows the main page of the data retrieval tool in accessing experiment data that are saved in MySQL. A user can choose a project and the category of interest. Once the selection is processed, the retrieved data will be shown on the browser. For example, Figure 11 shows the data related to category "Summary" and Figure 12 shows the data related to category "Instrumentation".

Figure 10 – Main Page to Retrieve MiniMOST Data Saved in MySQL

SLOT	VALUE
projectID	100001
title	miniMOST-1
contractID	CMS-0117853
localTimeZone	CST
startDate	2003-08-01
endDate	2004-09-30
keywords	NEESgrid service; NEESpop server; Multi-Site Online Simulation Test
description	This is the Mini-MOST experiment.
objectives	The main purpose of the Mini-MOST experiment is to show the capability of major NEESgrid service components using a small-scale physical experiment setup. Comparing with the Multi-site Online Simulation Test (MOST) conducted on July 30, 2003, the Mini-MOST experimental hardware, as implied by its name, is small in size and can be easily packed and shipped to other places. However, the software involved in this experiment is similar to what was used for the MOST experiment and provides the same level of functionality and services. Therefore, Mini-MOST experiment provides a desirable platform for students and

Figure 11 – Display of "Summary" for MiniMOST Experiment



The following files are related to **Instrumentation Setup**:

SLOT	VALUE
sensorSetupID	3000021
name	miniMOST_UIUC_sensor_setup
description	Instrument Wiring and Settings for Mini-MOST Experiment
eventGroupID	300001

The related files are as follows:

SLOT	VALUE
fileID	30000011
name	LVDT1.jpg
type	JPEG Image
description	A photo of the LVDT sensor
authors	UIUC
dateModified	2003-12-08 02:12:00
uri	<a href="http://171.64.55.25/miniMOST/UIUC/figures/LVDT1.jpg">http://171.64.55.25/miniMOST/UIUC/figures/LVDT1.jpg</a>

SLOT	VALUE
fileID	30000012

Figure 12 – Display of “Instrumentation” for MiniMOST Experiment

### 3. Data Retrieval Tools / Composite Frame Test / OWL

Data retrieval tools can also be used to retrieve and organize data saved in OWL files. As shown in Figure 13, a main page is provided to retrieve the experiment data from the composite frame test. Based on the category of interest, certain related data are retrieved and shown on the browser. Figure 14 shows the data related to category “Specimen”. Detailed information about “Specimen” can be accessed from this page, such as a particular drawing of the specimen shown in Figure 15. Similarly, description and detailed information about “Loading System” can be retrieved, as shown in Figure 16 and Figure 17.



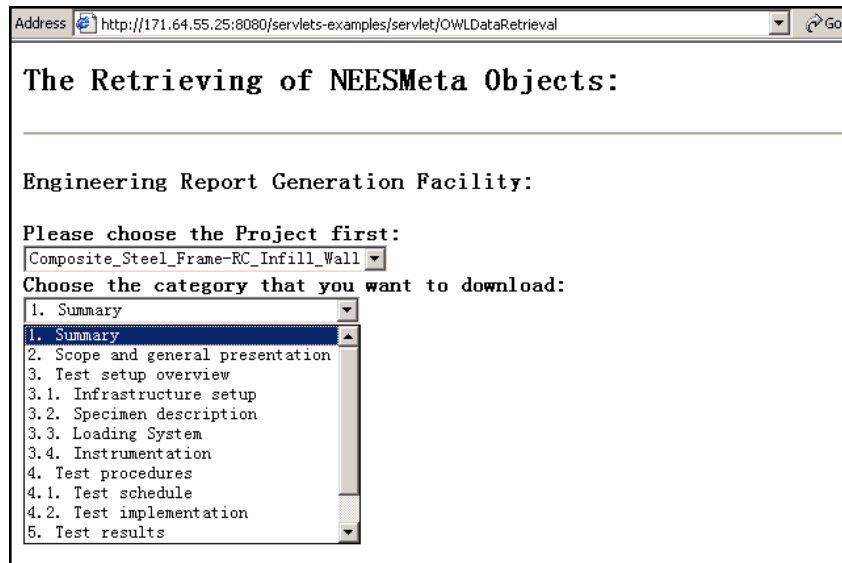


Figure 13 – Main Page to Retrieve Composite Frame Test Data Saved in OWL

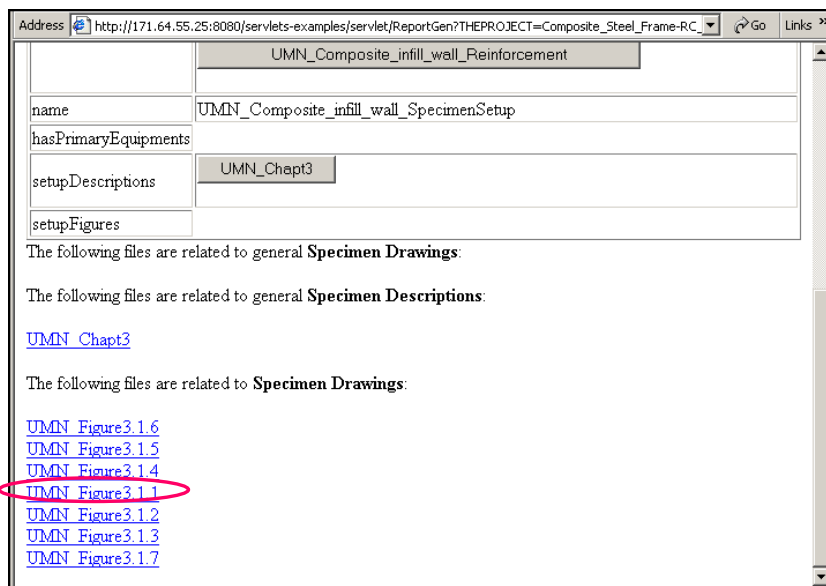


Figure 14 – Display of “Specimen” for the Composite Frame Test





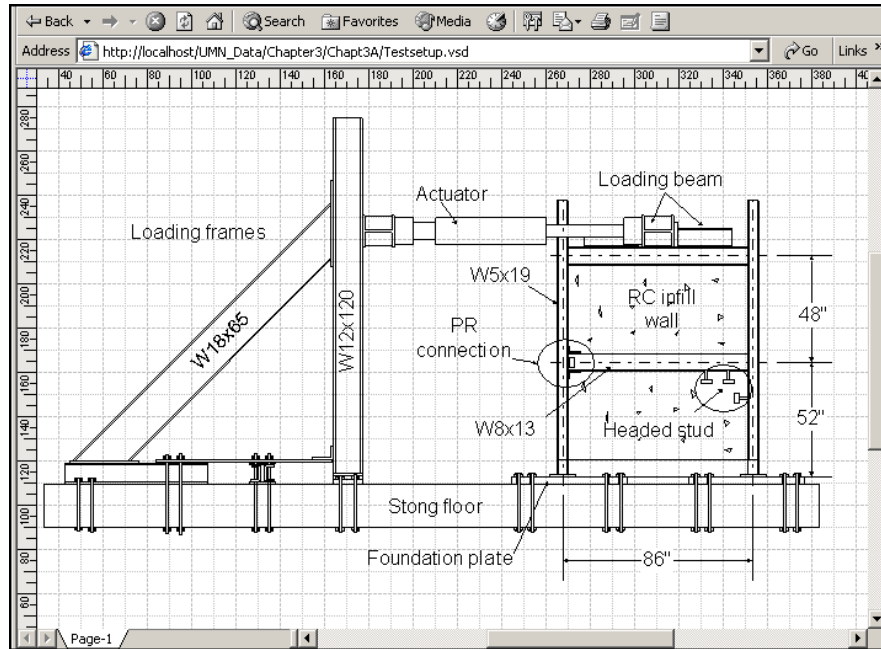


Figure 15 – Access of Drawings for the “Specimen”

**Engineering Report Generation Facility:**

The following are related to **Loading** of EventGroups:

The following is the details of UMN\_Composite\_infill\_wall\_LoadHistory:

SLOT	VALUE
shortDescription	The cyclic loading history on the composite infill wall
hasWaveForm	UMN_LoadHistory

The cyclic loading history is shown in Figure 3.5.1. A total of 25 loading cycles were divided into 9 loading groups. Each group had 3 loading cycles except for the last two groups, which had 2 loading cycles each. The cyclic loading history was modified from SAC protocol for testing of steel beam-column connections and other steel specimens (SAC, 1997). In order to capture the characteristics of crack formation and Fig. 3.5.1 Loading History progression in the RC infill wall, 3 loading group were applied before the total drift reached 0.5%, and these first 3 loading groups was controlled by the peak lateral load, with 40 kips for Group 1, 80 kips for Group 2 and 120 kips for Group 3. The loading history of the remaining 6 loading groups was controlled by the total drift of the specimen, with 0.5% for Group 4, 0.75% for Group 5, 1.0% for Group 6, 1.25% for Group 7, 1.5% for Group 8 and 1.75% for Group 9. During these 6 loading groups, the increment of the total drift was set at 0.25% instead of the 0.5% or 1% in SAC protocol (SAC, 1997) because the specimen exhibited

Figure 16 – Display of “Loading System” for the Composite Frame Test



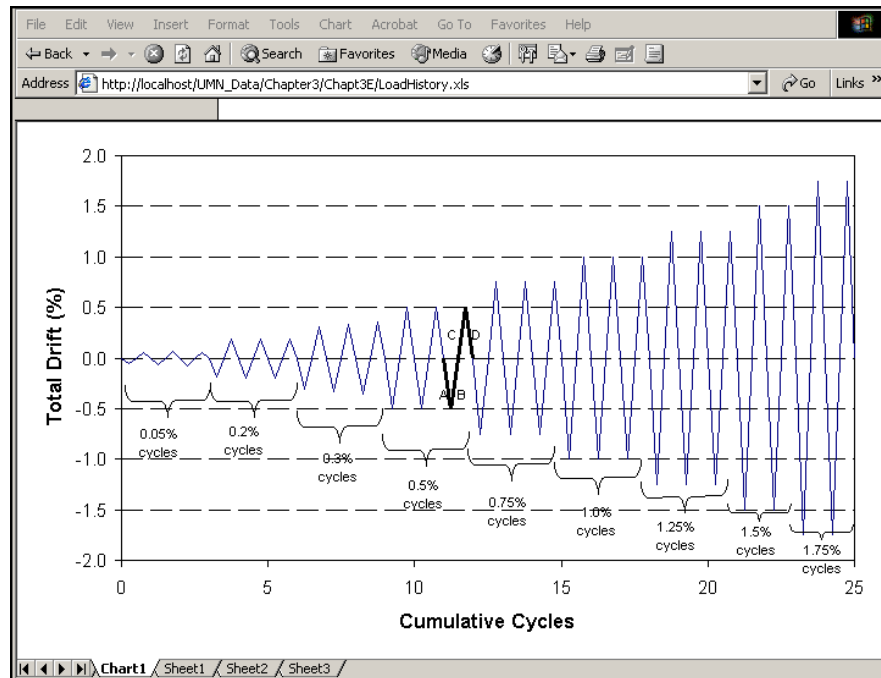


Figure 17 – Access of Drawings for the “Loading System”

#### 4. Report Generator

The main page for the Report Generator is shown in Figure 18. The same interface is used for accessing the data saved in either OWL files repository or MySQL database. Once a particular project is chosen, the main interface of the report generator will be shown. As illustrated in Figure 19, the left frame of the main interface displays buttons which represent the categories as defined in the report file hierarchy. Upon clicking on a particular button, the right frame displays the content of that category. For example, Figure 19 shows the miniMOST experiment data related to category “Summary”.

Besides retrieving the project data from the data repository, the Report Generator also supports the scenario where project data are saved in different types of repositories. For instance, miniMOST experiment data are saved in OWL files as well as in a MySQL database. Figure 20 shows the interface to access miniMOST experiment data saved in OWL files, whereas Figure 21 shows the interface to access miniMOST experiment data saved in MySQL database.

Another test case is using the Report Generator to access the composite structural frame test data saved in OWL files. The main interface to access the test data is shown in Figure 22. The content of a particular category can be directly accessed. Figure 23 shows the data related to category “Infrastructure”, Figure 24 shows the data related to category “Loading”, and Figure 25 shows the data related to category “Instrumentation”.



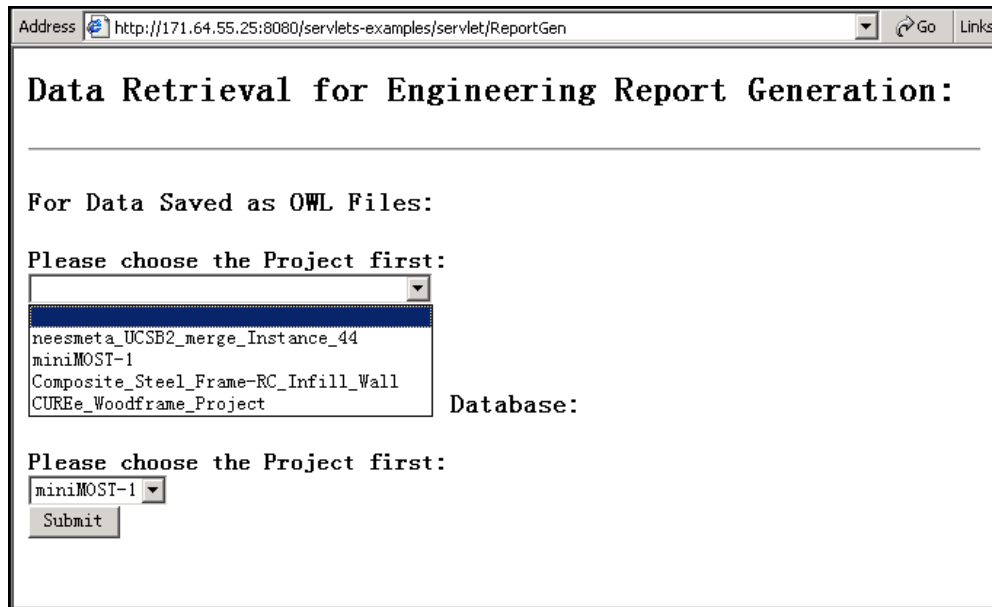


Figure 18 – Main Page of the Report Generator

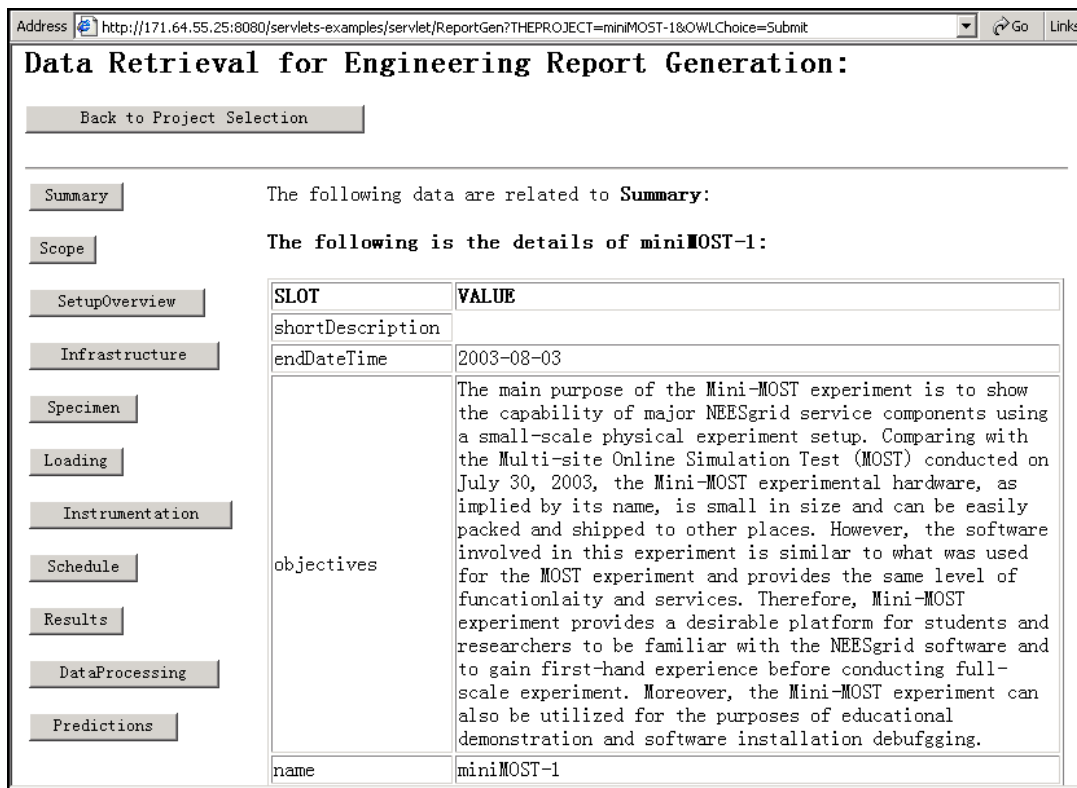


Figure 19 – Main Interface of Report Generator for MiniMOST Experiment



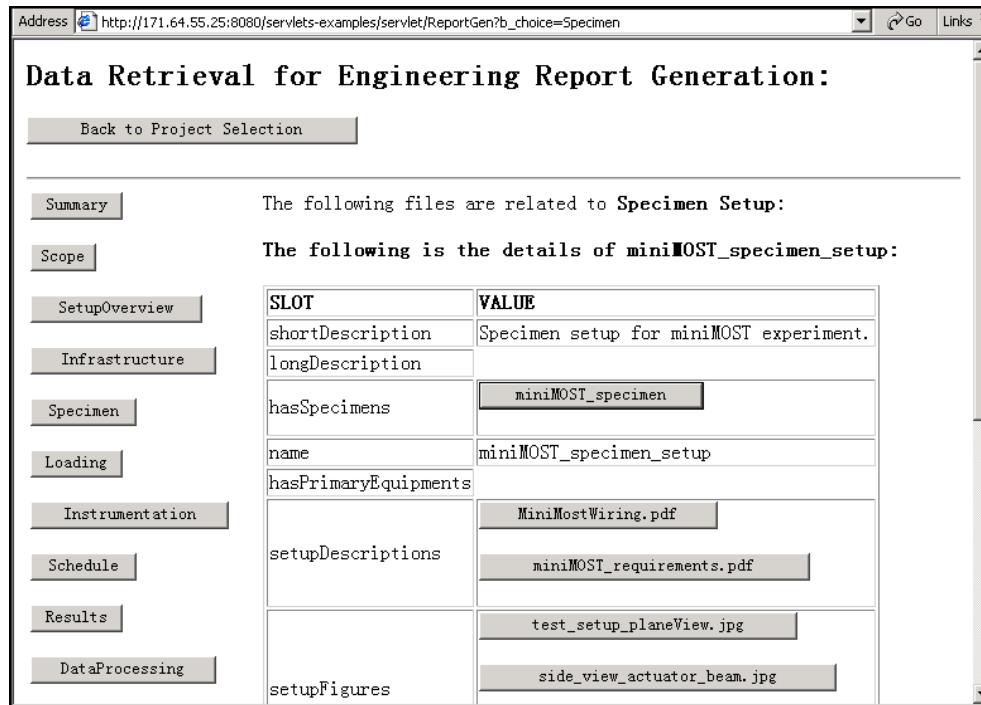


Figure 20 – Access of “Specimen” for MiniMOST Data Saved in OWL

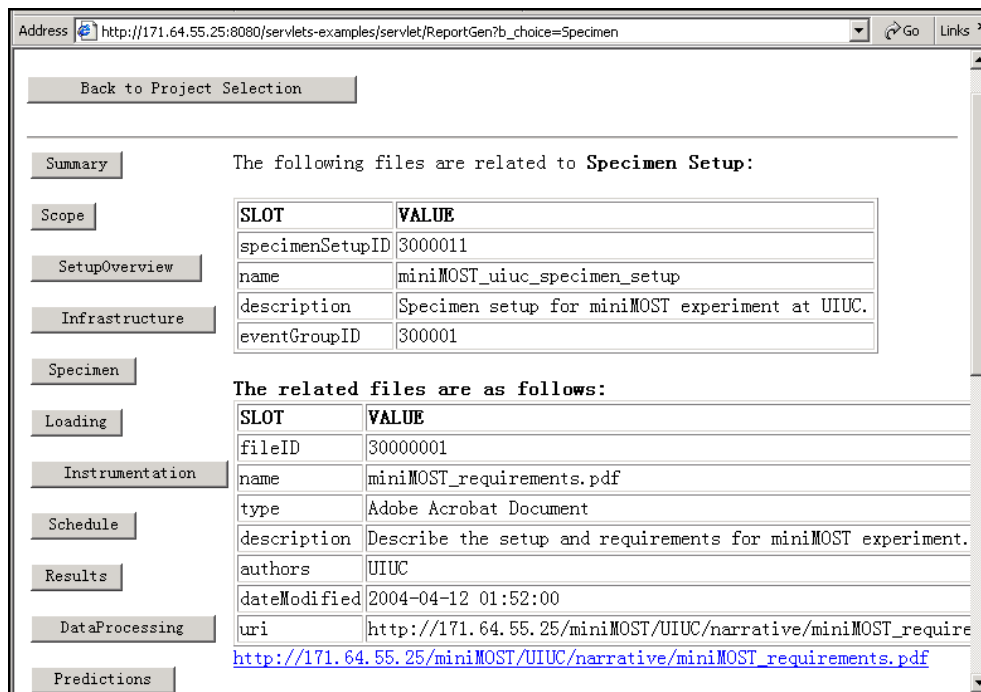


Figure 21 – Access of “Specimen” for MiniMOST Data Saved in MySQL



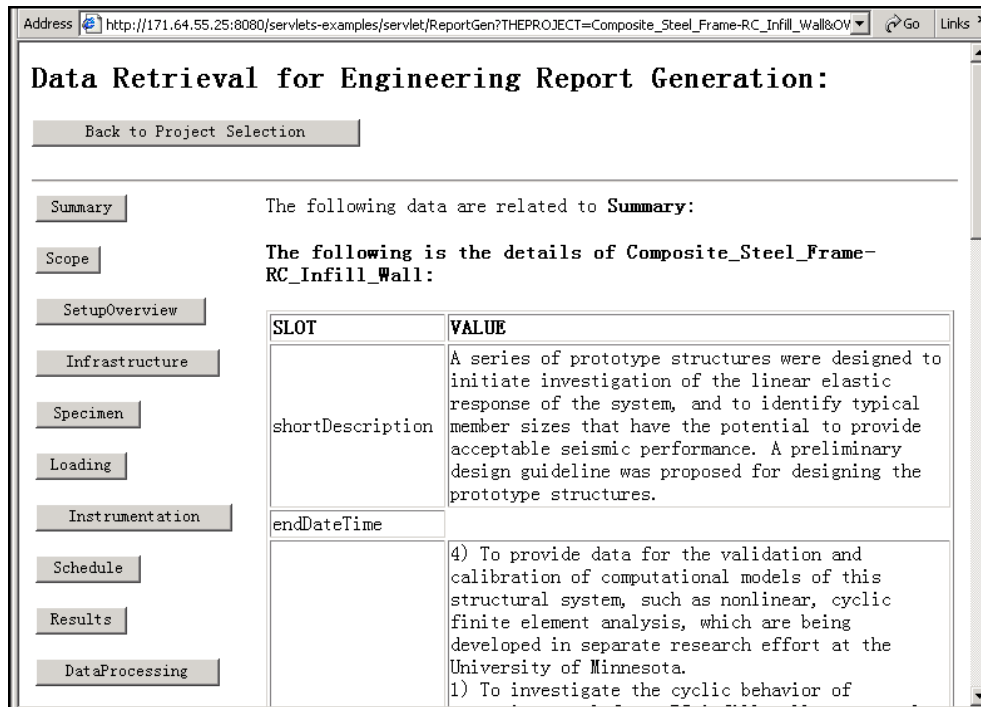


Figure 22 – Main Interface of Report Generator for the Composite Frame Test

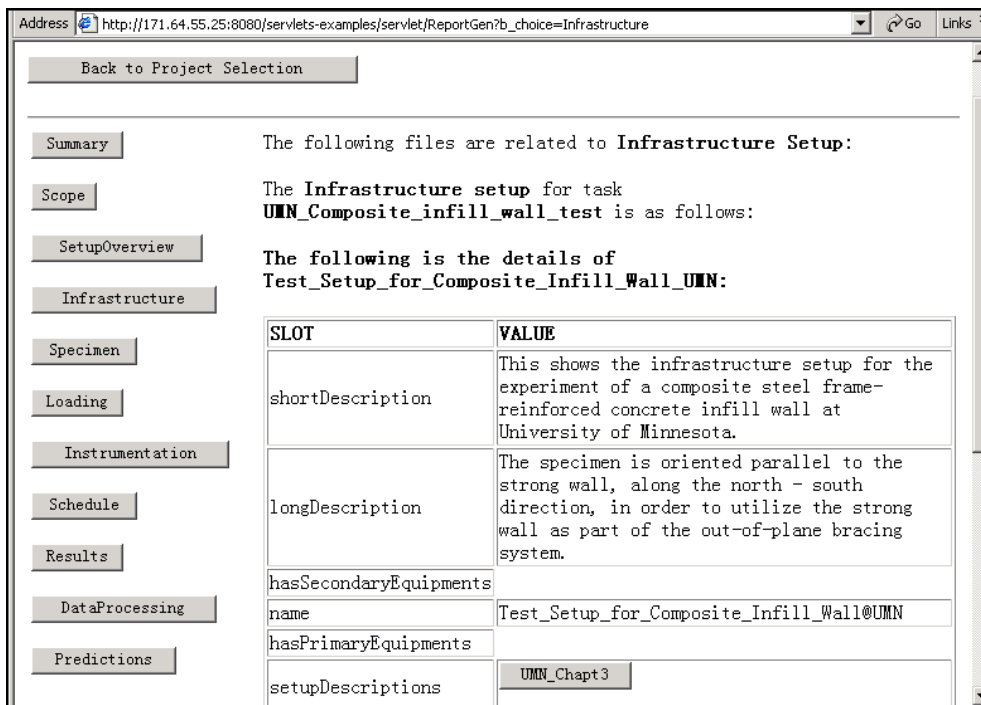


Figure 23 – Access of “Infrastructure” Data for the Composite Frame Test



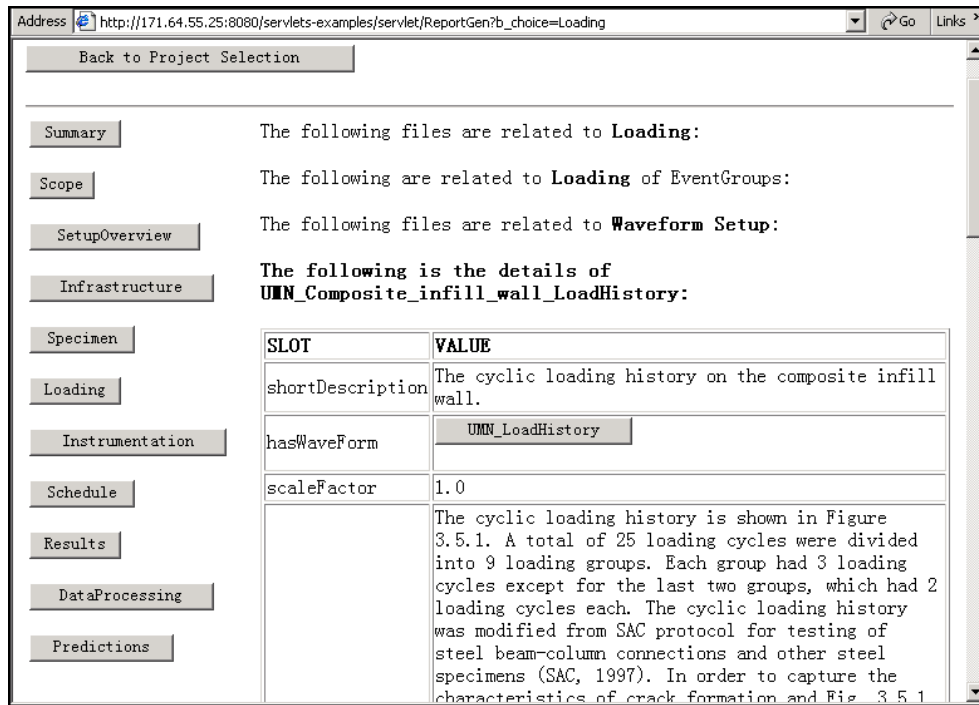


Figure 24 – Access of “Loading” Data for the Composite Frame Test

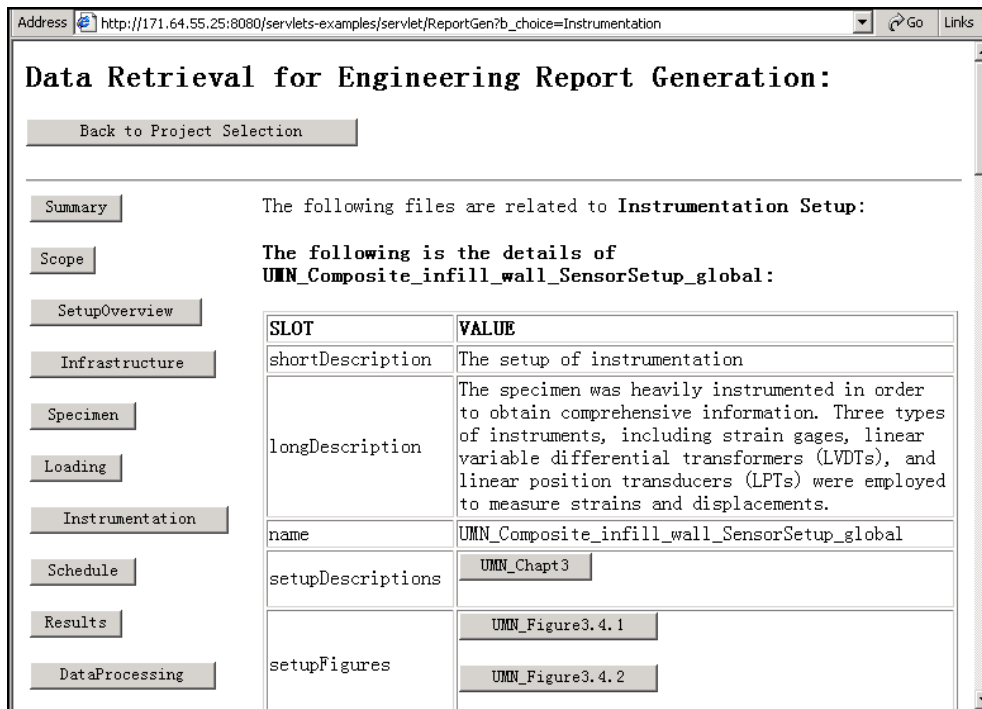


Figure 25 – Access of “Instrumentation” Data for the Composite Frame Test

