NGL Database

Overview and Discussion

Scott J. Brandenberg, Ph.D., P.E.
Civil and Environmental Engineering, UCLA
Dong Youp Kwak, Ph.D.
RMS, Formerly UCLA
Paolo Zimmaro, Ph.D.
Civil and Environmental Engineering, UCLA

Outline

- Overview of NGL database vision and mission.
- Why a relational database instead of traditional data repository?
- Database current status and future directions.
- Vision for how developers will interact with NGL database.

NGL Database Objective

- Develop a database of liquefaction case histories, and provide a geo-spatial web interface to the data.
- Permit users to upload data to, and download data from the database.
- Permit users to view with data in the web application, and eventually perform calculations using the data (will require DesignSafe integration).

What is a Database?

- Common definition used by engineers:
  - A collection of data.
  - Examples include experimental data archived in DesignSafe, or the ground motion records made available through the various NGA projects.
- However, this is a data repository, not a database, according to the computer science community
- The word “database” refers to a relational database (e.g., MySQL, MS Access).

What is a Relational Database?

- A structured body of related information.
- Data are organized as a set of tables formally described by a “schema”.
- Tables are related to each other by shared fields called “keys”.
  - Primary Key: A unique identifier for each record
  - Foreign Key: A field in one table that identifies a record in another table. Used to relate two tables.
- Databases are typically accessed using the *structured query language* (SQL).

Example Database

Table: users

name	company	company_address	url1	url2
Joe	ABC	1 Work Lane	abc.com	xyz.com
Jill	XYZ	1 Job Street	abc.com	xyz.com

- Problems:
  - What if a future user wants to list three URL's? Would need to add columns, and many users would have null fields.
  - Multiple users might work for the same company. No need to duplicate company info for every user.

Example Database

- Solve problems by dividing into multiple related tables:
  - Assign primary key to every field.
  - Eliminate fields that do not depend on the key, and create new table for those fields.
  - Relate fields in separate tables using foreign key.

Third Normal Form

Table: users

userId	name	companyID
1	Joe	1
2	Jill	2

Table: companies

companyID	company	company_address
1	ABC	1 Work Lane
2	XYZ	1 Job Street

Table: urls

urlId	relUserId	url
1	1	abc.com
2	1	xyz.com
3	2	abc.com
4	2	xyz.com

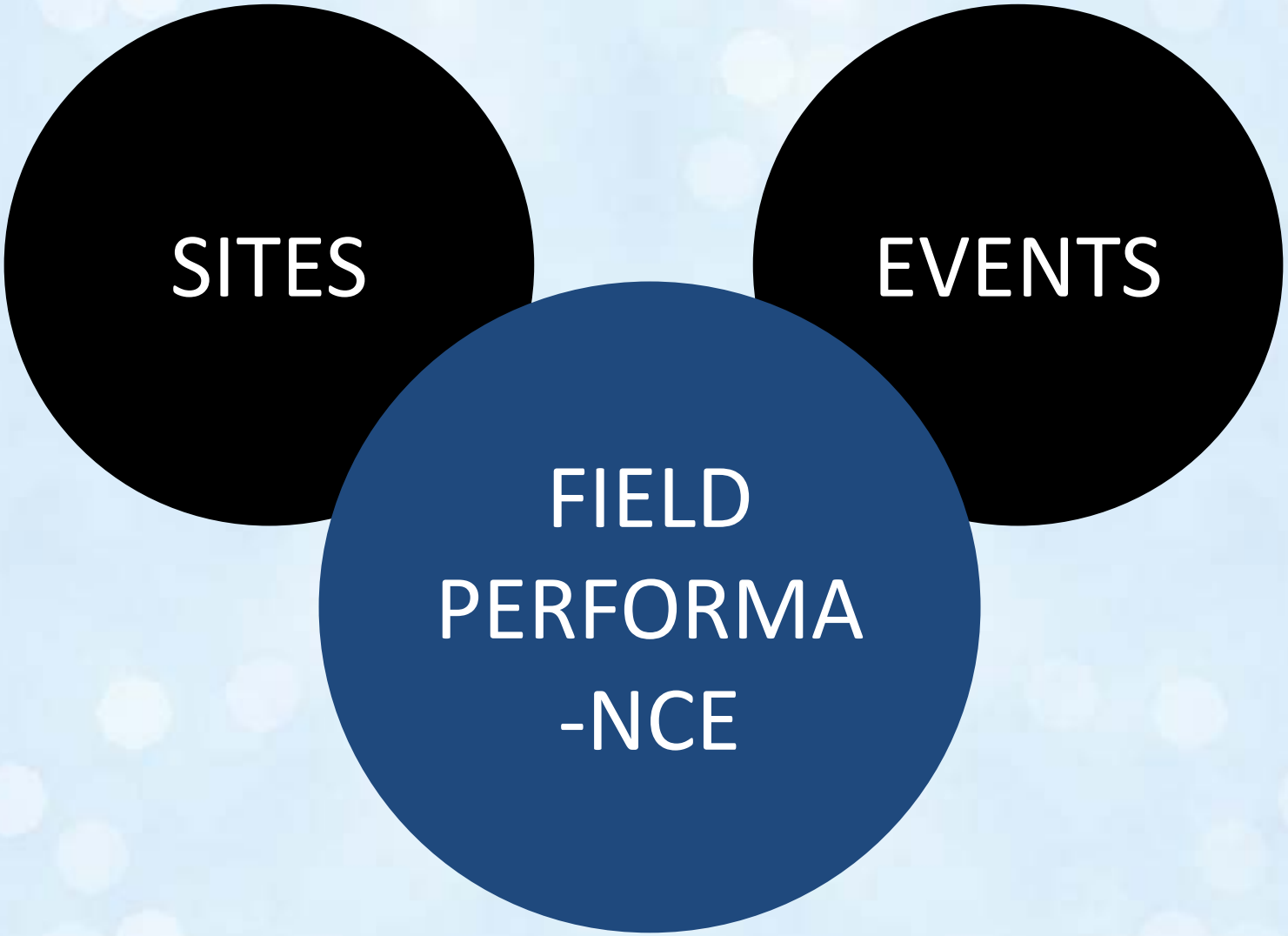
Why is NGL a Relational Database?

- One option would be to store case history data in a file repository. This is problematic for a few reasons:
  - To extract data (e.g., latitude and longitude) to populate the web app, a large number of files would need to be opened and read, which is inefficient and slow.
  - Data repositories often require repeated information, increasing the possibility of inconsistencies in repeated fields (e.g., NGA Flatfiles) or lost relationships among data quantities.

NGL Database Progress

- Database structure has been created, and is continuously being revised.
- Web interface has been developed.
- Database has been populated with 63 case histories by several different researcher groups.
- We have learned quite a bit about data structure and user experience, and are nearing the release of the second version of the database and web app.

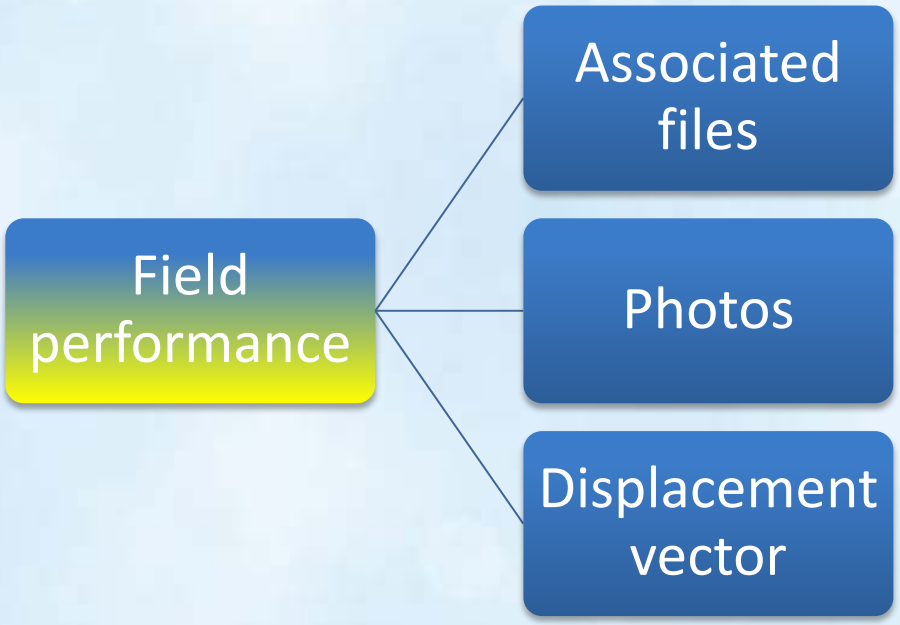
NGA Objects



- A site is a collection of data selected by a user.
- Consists of specific locations where boreholes, CPT soundings, test pits, or geophysical measurements are performed.
- Includes laboratory tests performed on samples.

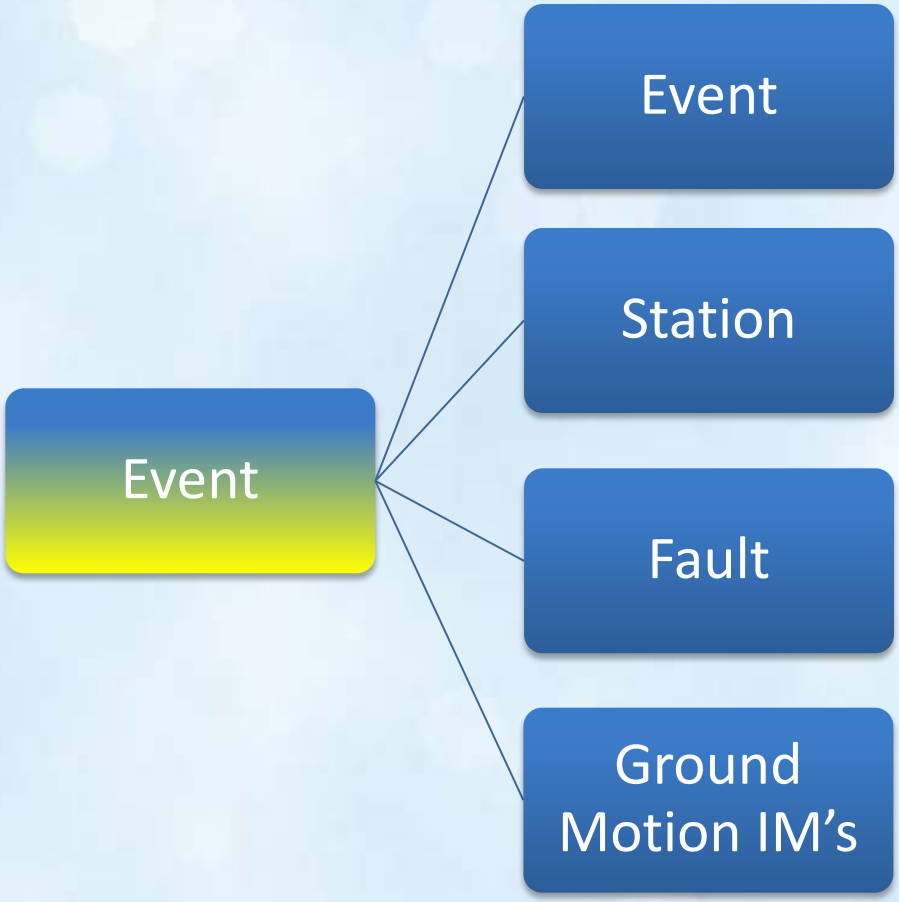
Field Performance Database Structure

- Field performance database consists of associated files (e.g., LiDAR point clouds), photos, and displacement vectors.



Event Database Structure

- Event database consists of event info (M_w, style of faulting, etc.), station info, fault geometry info, and ground motion IM's.
- NGAWest2 and NGASub datasets will be included in addition to other necessary events.
- Events uploaded by super-users



Site Table

SITE_ID	INT(6)	UNSIGNED AUTO_INCREMENT	Unique ID for the table SITE
USER_ID	INT(6)	UNSIGNED	Unique ID for the table USER
SITE_NAME	VARCHAR(30)		Site name may be defined based on the name of the location.
SITE_LAT	VARCHAR(30)		Latitude of the site (e.g., center of the site) in decimal degree following WGS84 system
SITE_LON	TIMESTAMP		Longitude of the site (e.g., center of the site) in decimal degree following WGS84 system
SITE_ELEV	VARCHAR(30)		Surface elevation with respect to mean sea level
SITE_GEOL	VARCHAR(30)		Description of surface geology. If available, indicate geoloty unit in parentheses [e.g., Alluvial plain (Qa)].
SITE_REM	VARCHAR(30)		Remark
SITE_STAT	BOOL		For upload status (1 = submitted / 0 = temporal)
SITE_REVW	BOOL		For review status (1 = reviewed / 0 = not reviewed)

Location Table

LOCA_ID	INT(6)	UNSIGNED AUTO_INCREMENT	Unique ID for the table LOCA
SITE_ID	INT(6)	UNSIGNED	Unique ID for the table SITE
LOCA_LAT	FLOAT(10)		Latitude of activity in decimal degree following WGS84 system
LOCA_LON	FLOAT(10)		Longitude of activity in decimal degree following WGS84 system
LOCA_TYPE	VARCHAR(10)		Type of activity. Four options: - Borehole - Cone penetration test (CPT) - Test pit - Geophysical investigation
LOCA_GL	VARCHAR(10)		Surface elevation with respect to mean sea level.
LOCA_STAR	VARCHAR(20)		Start date of activity.
LOCA_ENDD	VARCHAR(20)		End date of activity.
LOCA_REM	VARCHAR(1000)		Remark

SPT Table

ISPT_ID	INT(6)	UNSIGNED AUTO_INCREMENT	Unique ID for the table ISPT
LOCA_ID	INT(6)	UNSIGNED	Unique ID for the table LOCA
ISPT_TOP	VARCHAR(10)		Depth of the top of main blows
ISPT_TPEN	VARCHAR(10)		Penetration depth from depth top (main drive only; e.g., 30 cm)
ISPT_NVAL	VARCHAR(5)		Number of blow counts for main drive
ISPT_ERAT	VARCHAR(10)		Hammer drop energy ratio
ISPT_MECH	VARCHAR(100)		Hammer drop system. Example: - Rope-cathead Trip - Semi-automatic - Automatic Another system can be described.
ISPT_METH	VARCHAR(100)		Method (i.e., standard) followed if different from ASTM D1586-11.
ISPT_REM	VARCHAR(1000)		Remark

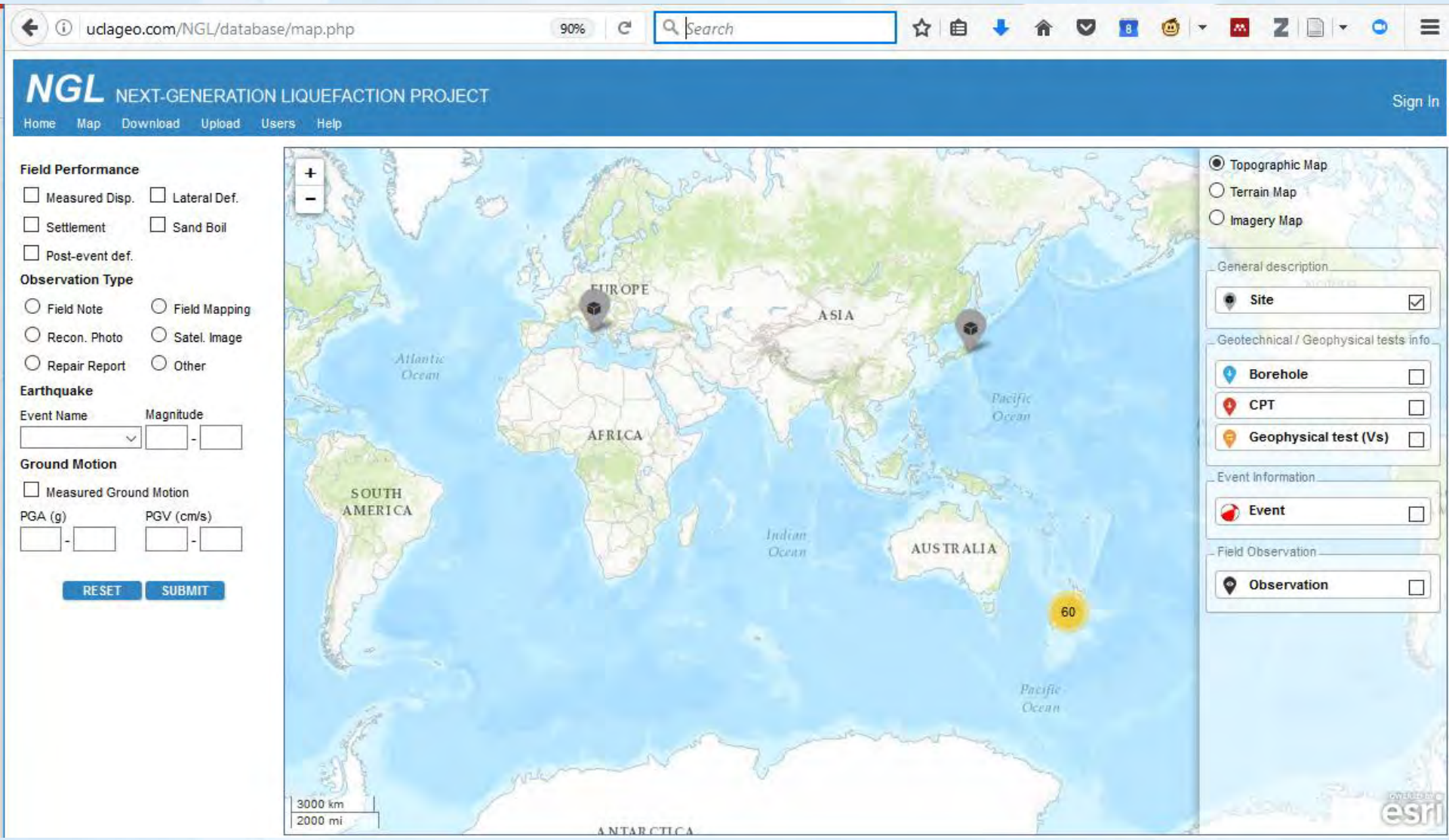
NGL Schema

- Some tables have structured data fields (e.g., ISPT has depth, blow count, energy ratio, etc.).
- The database also accepts files of any type that users wish to upload (e.g., geology maps, LiDAR point clouds, etc.) using the BLOB database format.

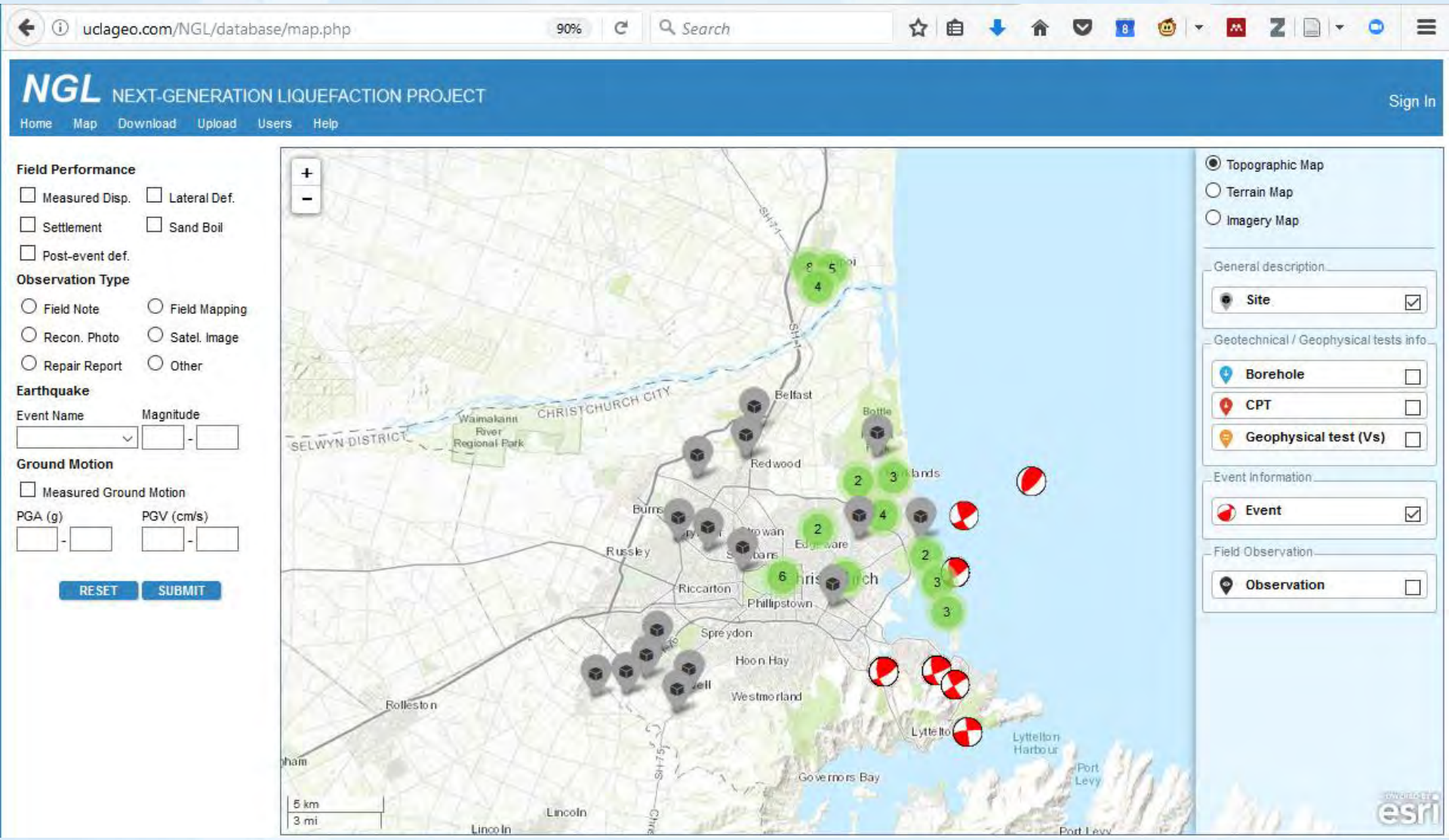
Example Table for Files

Field	LOCF_ID	INT(6)	UNSIGNED AUTO_INCREMENT	Unique ID for the table LOCF
Field	LOCA_ID	INT(6)	UNSIGNED	Unique ID for the table LOCA
Field	LOCF_DESC	VARCHAR(1000)		Detailed description of the associated file (e.g., original data, profile image).
Field	BLOB_ID	INT(6)	UNSIGNED	Unique ID for the table BLOB

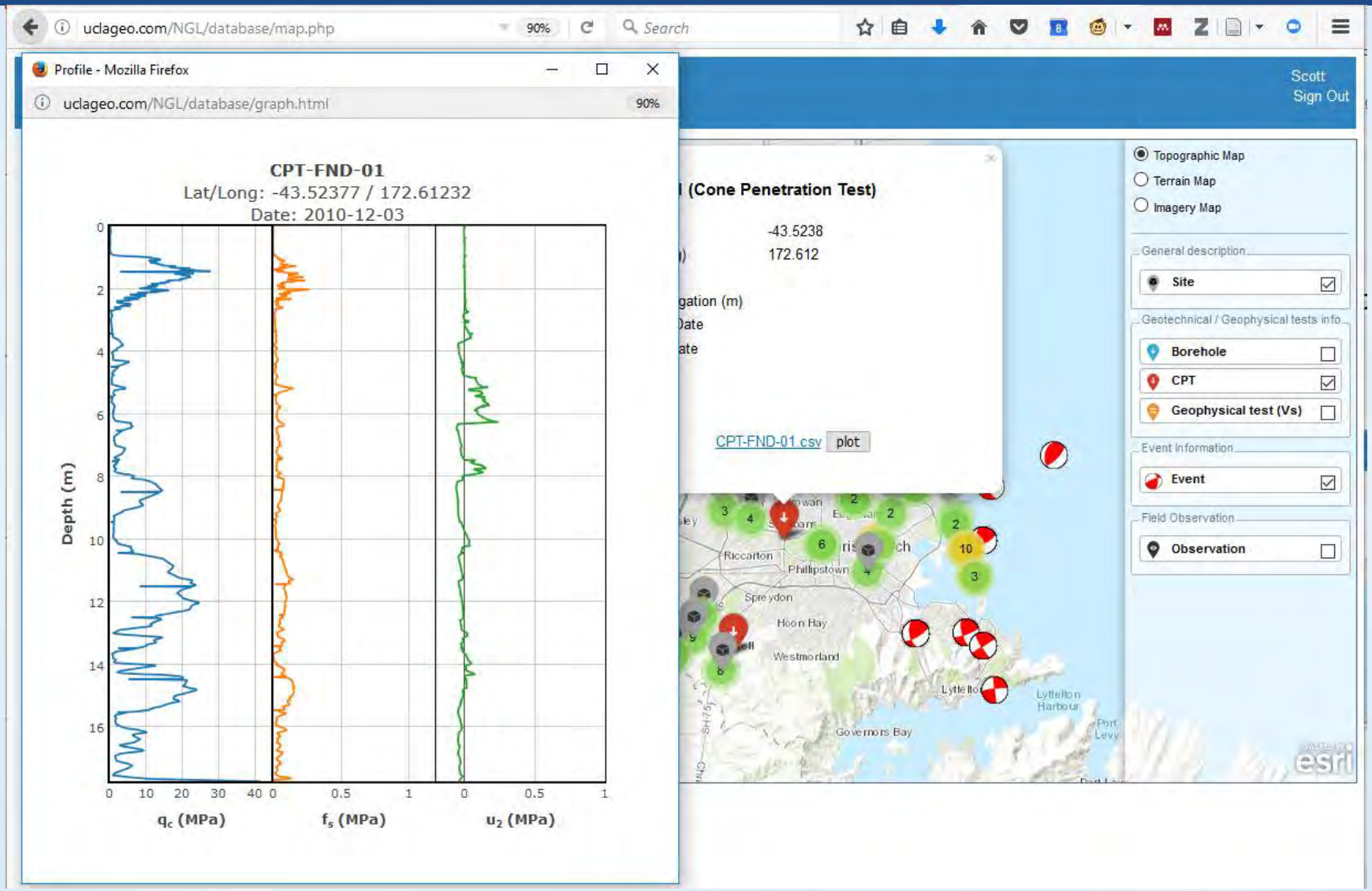
Web Interface: Map



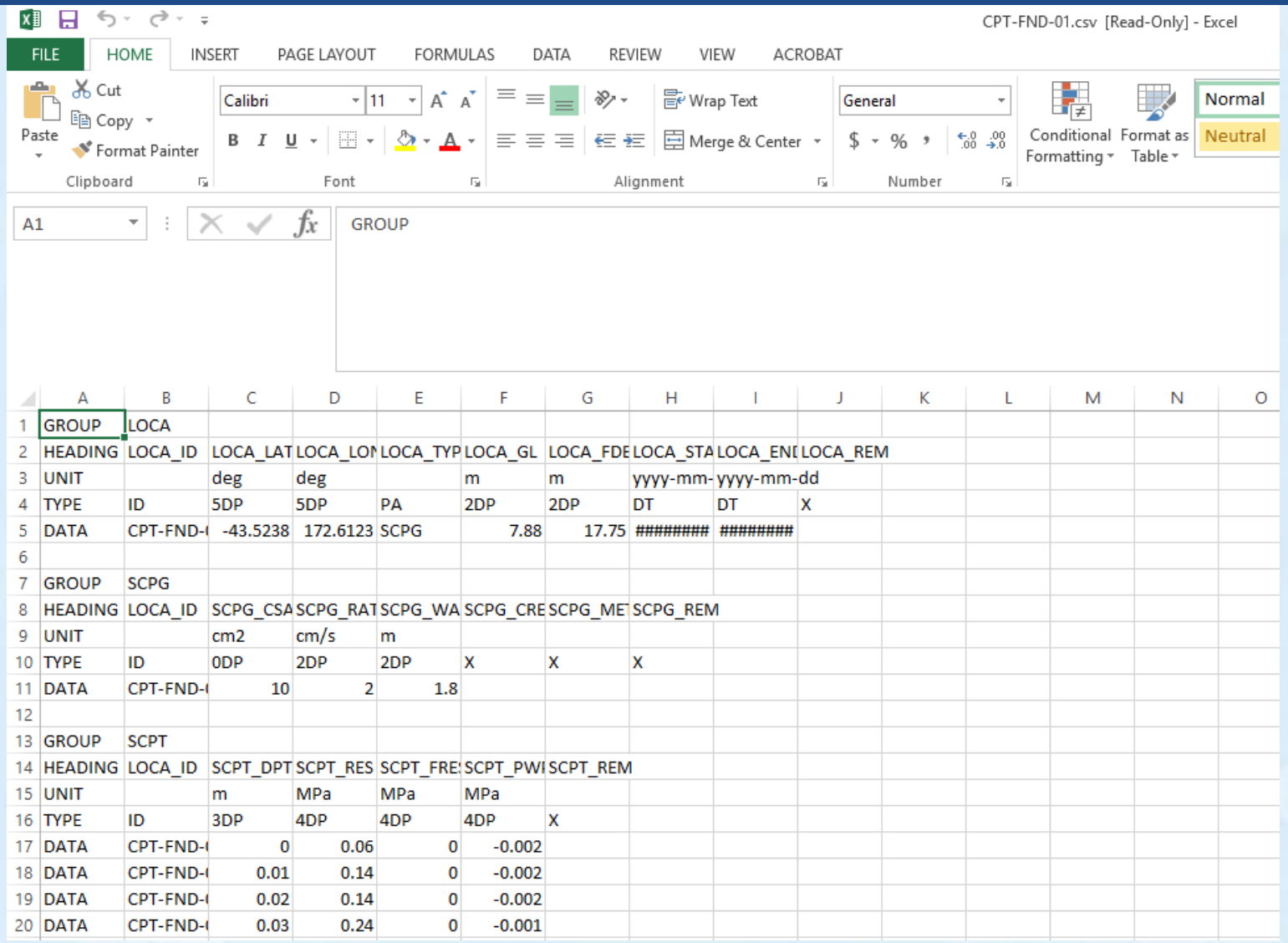
Web Interface: Map



Web Interface: Map

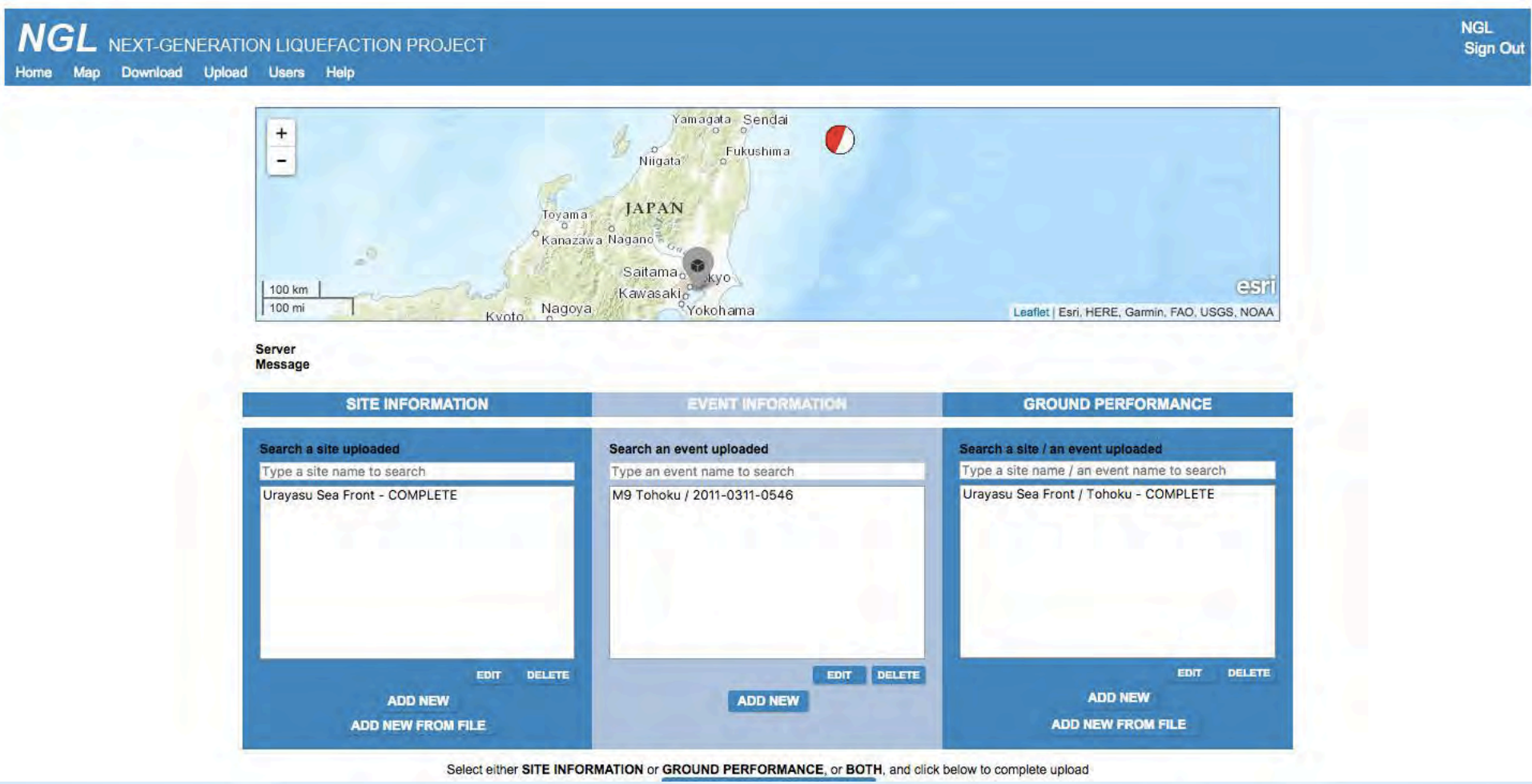


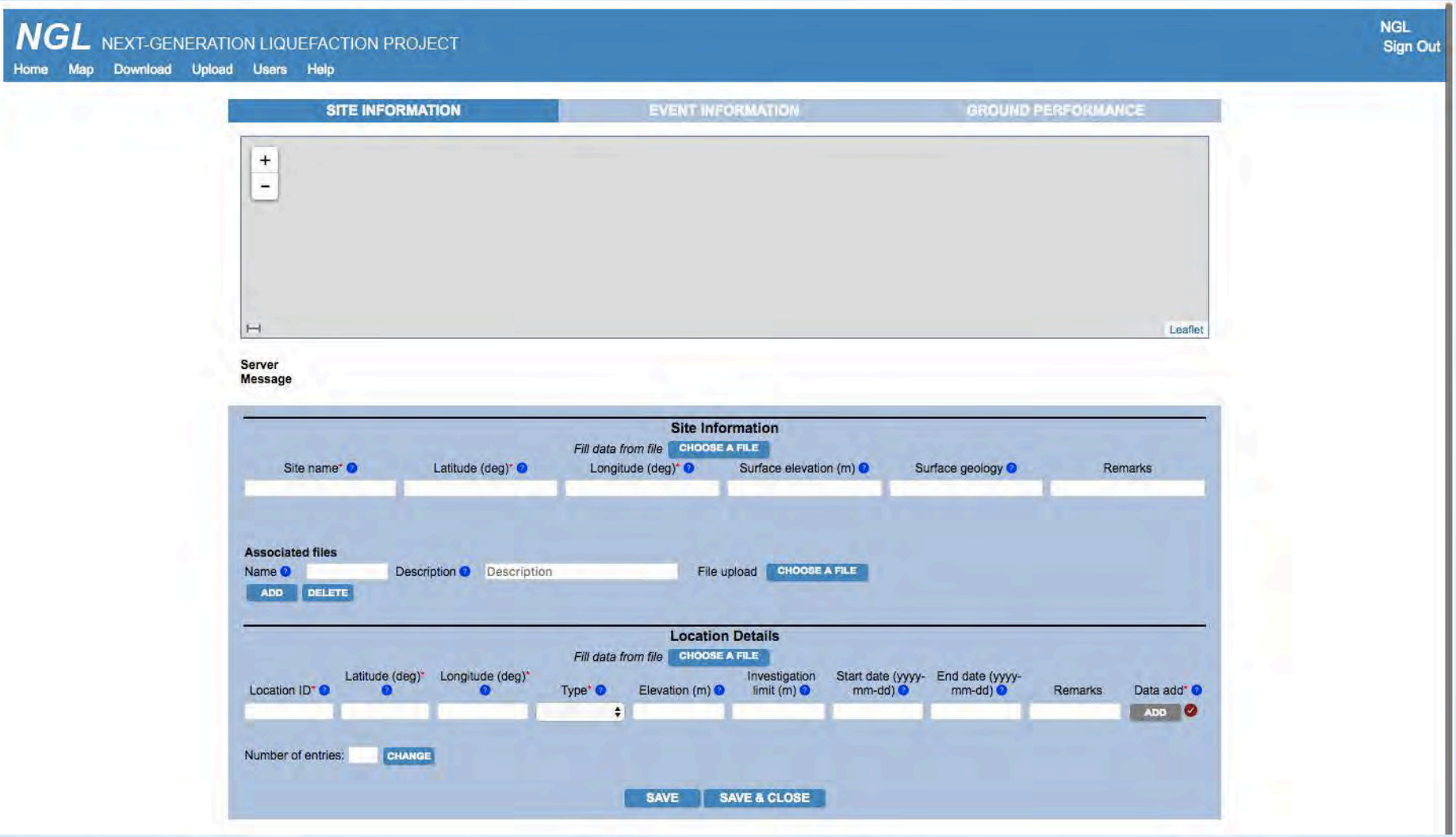
Web Interface: Download



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	GROUP	LOCA													
2	HEADING	LOCA_ID	LOCA_LAT	LOCA_LOI	LOCA_TYP	LOCA_GL	LOCA_FDE	LOCA_STA	LOCA_ENI	LOCA_REM					
3	UNIT		deg	deg		m	m	yyyy-mm-	yyyy-mm-dd						
4	TYPE	ID	5DP	5DP	PA	2DP	2DP	DT	DT	X					
5	DATA	CPT-FND-(	-43.5238	172.6123	SCPG	7.88	17.75	########	########						
6															
7	GROUP	SCPG													
8	HEADING	LOCA_ID	SCPG_CSA	SCPG_RAT	SCPG_WA	SCPG_CRE	SCPG_MET	SCPG_REM							
9	UNIT		cm2	cm/s	m										
10	TYPE	ID	0DP	2DP	2DP	X	X	X							
11	DATA	CPT-FND-(	10	2	1.8										
12															
13	GROUP	SCPT													
14	HEADING	LOCA_ID	SCPT_DPT	SCPT_RES	SCPT_FRE	SCPT_PWI	SCPT_REM								
15	UNIT		m	MPa	MPa	MPa									
16	TYPE	ID	3DP	4DP	4DP	4DP	X								
17	DATA	CPT-FND-(	0	0.06	0	-0.002									
18	DATA	CPT-FND-(	0.01	0.14	0	-0.002									
19	DATA	CPT-FND-(	0.02	0.14	0	-0.002									
20	DATA	CPT-FND-(	0.03	0.24	0	-0.001									

Web Interface: Upload

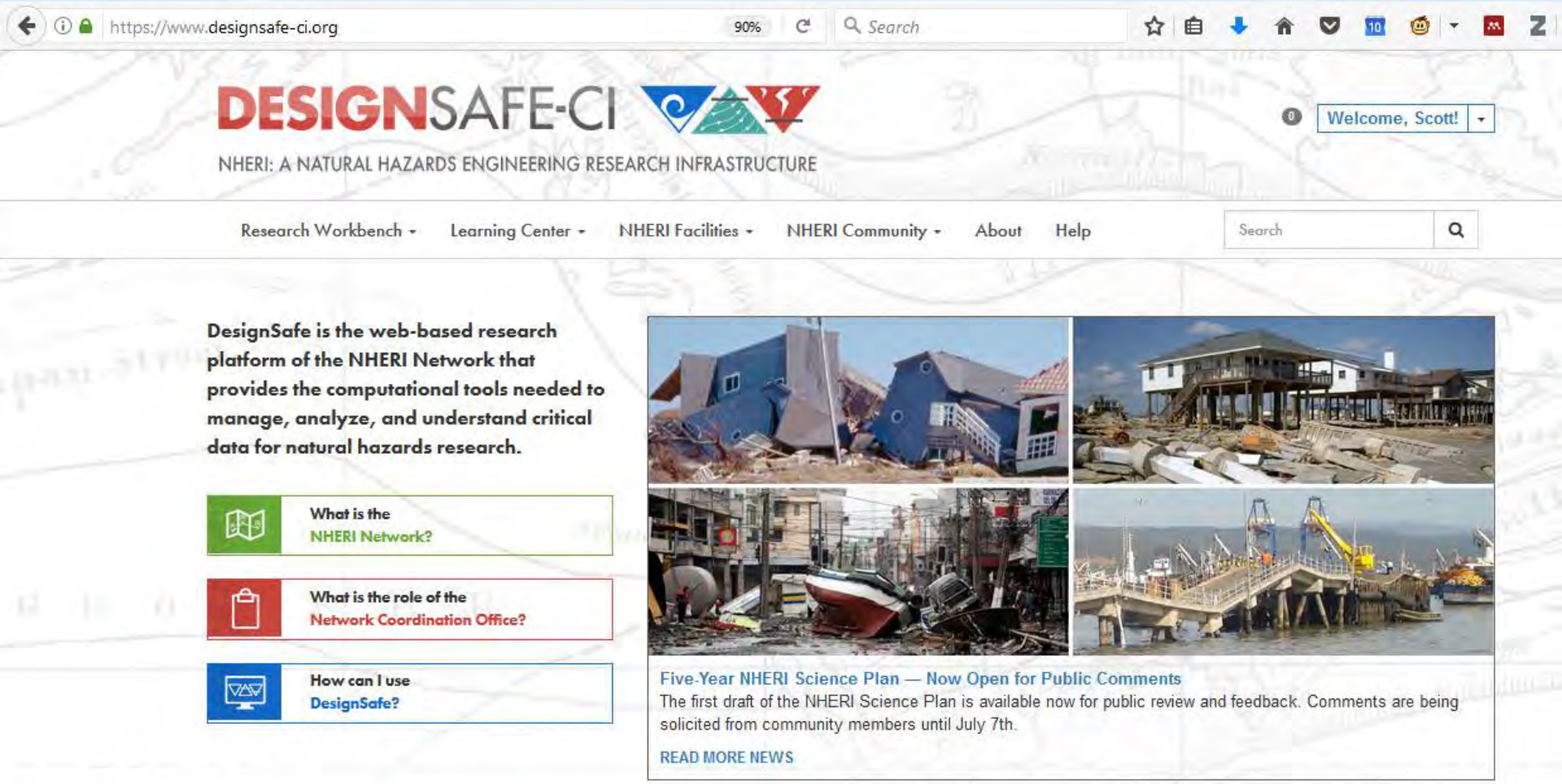



NGL NEXT-GENERATION LIQUEFACTION PROJECT
Home Map Download Upload Users Help
NGL
Sign Out
SITE INFORMATION
EVENT INFORMATION
GROUND PERFORMANCE
+
-
Leaflet
Server
Message
Site Information
Fill data from file
CHOOSE A FILE
Site name*
Latitude (deg)*
Longitude (deg)*
Surface elevation (m)
Surface geology
Remarks
Associated files
Name
Description
Description
File upload
CHOOSE A FILE
ADD
DELETE
Location Details
Fill data from file
CHOOSE A FILE
Location ID*
Latitude (deg)*
Longitude (deg)*
Type*
Elevation (m)
Investigation limit (m)
Start date (yyyy-mm-dd)
End date (yyyy-mm-dd)
Remarks
Data add*
ADD
Number of entries:
CHANGE
SAVE
SAVE & CLOSE

Integration with DesignSafe

- NGL is currently hosted at uclageo.com, which is a domain I own and maintain.
- We are integrating with DesignSafe to leverage data security, staff support, and the ability for developers to interface with the data in the cloud.
- The amount of data is going to be too large to operate effectively using old-fashioned methods (i.e., download all data and process on a local computer).

Integration with DesignSafe



Integration with DesignSafe

- Likely to keep website hosted at UCLA, and mirror database at DesignSafe.
- Could transition web app to DesignSafe too. URL would be ngl.designsafe-ci.org.
- Users will be able to interact with database using MySQL queries in Matlab, or Jupyter notebooks running Python and/or R kernels.

Integration with DesignSafe

jupyter ExampleQuery Last Checkpoint: 04/25/2017 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python [conda root]

Code

In [7]:

```
import pymysql
import csv
import numpy
import xlrd
conn = pymysql.connect(host='127.0.0.1', port=3306, user='root', passwd='root', db='ngawest2')
cursor = conn.cursor()

MySQLCommand1 = "SELECT event_name FROM event WHERE event_id=1"
cursor.execute(MySQLCommand1)
event_name = cursor.fetchone()[0]
print("Event Name = " + event_name)

MySQLCommand2 = "SELECT pga FROM motion INNER JOIN event ON motion.event_id = event.event_id WHERE event.event_
cursor.execute(MySQLCommand2)
pga = cursor.fetchone()[0]
print("PGA = " + str(pga))

conn.commit()
conn.close()
```

```
Event Name: Helena, Montana-01
PGA = 0.15702
```

In [ ]:

Questions / Discussion?