Title

Next-Generation Liquefaction (NGL) Case History Database Structure

Journal

GEOTECHNICAL EARTHQUAKE ENGINEERING AND SOIL DYNAMICS V: LIQUEFACTION TRIGGERING, CONSEQUENCES, AND MITIGATION, 2018-June(290)

ISSN

0895-0563

Authors

Brandenberg, Scott J

Kwak, Dong Youp

Zimmaro, Paolo

et al.

Publication Date

2018-06-07

DOI

10.1061/9780784481455.040

Peer reviewed

Next-Generation Liquefaction (NGL) Case History Database Structure

Scott J. Brandenberg, Ph.D., P.E., M. ASCE, [1] Dong Youp Kwak, Ph.D.,[2]
Paolo Zimmaro, Ph.D., M. ASCE [3] Yousef Bozorgnia, Ph.D., P.E., F. ASCE,[4]
Steven L. Kramer, Ph.D., M. ASCE,[5] Jonathan P. Stewart, Ph.D., P.E., F. ASCE[6]

[1]University of California, Los Angeles, Dept. Civil & Environmental Engineering; Los Angeles, CA, 90095; e-mail: sjbrandenberg@ucla.edu
[2]RMS, Inc., Newark, CA, USA, 94560; e-mail: dongyoup.kwak@rms.com
[3]University of California, Los Angeles, Dept. Civil & Environmental Engineering; Los Angeles, CA, 90095; e-mail: pzimmaro@ucla.edu
[4]University of California, Los Angeles, Dept. Civil & Environmental Engineering; Los Angeles, CA, 90095; e-mail: yousef.bozorgnia@ucla.edu
[5]University of Washington, Dept. Civil & Environmental Engineering; Seattle, WA, 98195; e-mail: kramer@uw.edu
[6]University of California, Los Angeles, Dept. Civil & Environmental Engineering; Los Angeles, CA, 90095; e-mail: jstewart@seas.ucla.edu

ABSTRACT

This paper presents the organizational structure (schema) of the Next-Generation Liquefaction (NGL) relational database. The schema describes the tables, fields, and relationships among the tables, and provides an important resource for users who wish to interact with the database by writing queries. Structured relational databases are not commonly utilized in the natural hazards community, where file repositories are more commonly used and often called "databases". This paper also discusses what a relational database is, and why this approach was adopted for the NGL project.

INTRODUCTION

The Next-Generation Liquefaction (NGL) project is a multi-year community-based effort consisting of three components: (1) a transparent, open-source, community database of liquefaction case histories, (2) supporting studies for effects that should be captured in models but that cannot be constrained by case history data, and (3) model development (Stewart et al. 2016). This paper addresses the structure of the case history database that is accessible via a web interface at http://www.uclageo.com/NGL/ (last accessed 02/06/2018). Registered users can upload, view, and download data. The database was developed using the My Structured Query Language (MySQL) relational database management system (RDBMS). The web interface was developed using PHP: Hypertext Preprocessor (PHP), Hypertext Markup Language 5 (HTML5), and Javascript and also utilizes the Environmental Systems Research Institute (ESRI) Arc Geographic Information System (ArcGIS) Application Program Interface (API) and the Leaflet Javascript API to organize the data geo-spatially.

The essential data requirements for the NGL database were developed over a few years through a series of national and international community workshops. A draft version of the database was presented in a workshop at the University of California, Berkeley, in July 2017, and this paper presents an updated version of the database reflecting community input. While the database structure, as described here, is essentially complete, the database itself has only begun to be populated. The task of populating the database is an ongoing community task being overseen by the NGL Database Working Group consisting of the first author (chair), K. Onder Cetin, Kevin W. Franke, and Robb E.S. Moss.

The NGL database is a relational database, which differs from what many in the natural hazards community intend when they use the term "database". Often, data are organized into file repositories, which strictly-speaking should not be called databases. This paper first briefly describes relational databases, and explains why this approach was adopted for NGL. The paper then presents the organizational structure of the database, describing the tables, fields, and relationships among tables, which is called a schema. We anticipate that this paper will serve as an important resource for future users of the database who wish to write queries to extract data using the Structured Query Language (SQL).

WHAT IS A "DATABASE"?

The word "database" is often used by the natural hazard engineering community in a rather loose manner to mean a collection of data. However, this is not the definition widely agreed upon by the computer science community. Rather, a relational database (RDB) is a structured body of related information organized into inter-related tables formally described by a schema. Tables are related to each other by shared fields called "keys", where a *primary key* is a unique identifier for each record, and a *foreign key* is a field in one table that identifies a record in another table.

To illustrate the benefit of an RDB, consider the hypothetical information contained in Table 1 describing two different earthquake events that were each recorded by two ground motion stations. Each row corresponds to a specific ground motion record, and information about the earthquake must be repeated each time a new record is inserted into the table. Repeating information in a table presents the possibility for data inconsistencies because a user might accidentally type the event name, magnitude, or other fields incorrectly. Furthermore, the earthquake magnitude would need to be updated at potentially many different positions within the table if it happened to be revised at some point in the future . An RDB eliminates such problems by organizing all relevant information in different tables and defining relationships among them. This results in a structure that avoids repetitions and null fields.

Table 1. Hypothetical earthquake event, station, and ground motion information.

Event Name	Magnitude	Epicentral Latitude	Epicentral Longitude	Station Name	V_{S30} (m/s)	R_{jb} (km)	PGA (g)
Westwood Hills	6.3	34.0689	118.4452	Factor Building	380	2	0.84
Westwood Hills	6.3	34.0689	118.4452	Santa Monica Courthouse	215	14	0.28
Hollywood Valley	7.2	34.1027	118.3404	Factor Building	380	20	0.61
Hollywood Valley	7.2	34.1027	118.3404	Santa Monica Courthouse	215	30	0.32

The data in Table 1 contains three different types of information:

- Event: event name, magnitude, latitude and longitude of epicenter;
- Station: recording station name and time-averaged shear wave velocity in the upper 30m (V_{S30});
- Information specific to a recorded ground motion: distance to surface projection of fault, R_{jb}, and ground motion intensity, PGA.

The information should therefore logically be organized into three tables, as illustrated in Tables 2-4. Each entry is assigned a unique primary key, which in this case is an integer under entry names Event_id, Station_id, and Motion_id. The event name could potentially be the primary key for the Event table because the event name is unique. However, integers facilitate faster searches than long character strings, and it's possible that the event name could be modified at some point in the future if, for example, the epicenter location is modified. For these reasons, introducing an integer primary key is common practice. In addition to a primary key, the Motion table contains foreign keys that relate a particular motion to a particular event and to a particular recording station.

Dividing the information among three tables may seem unnecessarily complicated, but the structure presented in Tables 2-4 offers significant benefits over that in Table 1. Information for each event and station is entered only once, which eliminates the possibility for inconsistencies due to data entry errors, and also eliminates the need for updating multiple cells when an entry needs to be modified. The benefit of this structure may not seem significant for the small dataset used in this example, but it is easy to imagine the benefits realized for large datasets containing thousands of ground motion records from hundreds of events. The data presented in Tables 2-4 is said to be in the "third normal form" (Codd, 1972) because all of the entries are non-transitively dependent on the primary key. This means that each column entry can be derived by knowledge of the primary key, and that no column logically depends on any other column besides the primary key.

Table 2. Earthquake event table.

Event_id (Primary Key)	Event Name	Magnitude	Epicentral Latitude	Epicentral Longitude
1	Westwood Hills	6.3	34.0689	118.4452
2	Hollywood Valley	7.2	34.1027	118.3404

Primary Key
Foreign Key

Table 3. Recording station table.

Station_id	Station Name	V_{S30} (m/s)
1	Factor Building	380
2	Santa Monica Courthouse	215

Table 4. Ground motion table.

Motion_id	Event_id	Station_id	R_{jb} (km)	PGA (g)
1	1	1	2	0.84
2	1	2	14	0.28
3	2	1	20	0.61
4	2	2	30	0.32

NGL DATABASE STRUCTURE

One goal of the NGL project is to develop a transparent, open source, community database of case histories of liquefaction, ground failure, and non-ground failure (Stewart et al. 2016). This section describes the NGL case history database and its organization. In NGL, a case-history consists of three components: (1) geotechnical/geological site characterization, (2) observed field performance, including evidence for liquefaction and its effects, ground failure, or non-ground failure, and (3) earthquake event and ground motion information. The NGL database consists of 43 tables (10 for general information, 24 for site characterization, 5 for field performance observations, and 4 for earthquake events). Its structure is described by the database schema which represents the blueprint of how the database is constructed. The schema also defines relationships among tables through a formal definition of primary and foreign keys. The current version of the NGL schema has been refined through a community-based effort performed in the last two years via project coordination meetings and public workshops.

Figure 1 defines the content of various type of tables in the database. The *Users* table contains information about NGL database account holders, with USER_ID as the primary key. Along with individual users, information for each component of the database (site, observation, and event), can be accessed and modified by members of a research team (groups of one or more users) with permissions to access information uploaded by members of the team. Tables *Site Member* (MEMS), *Observation Member* (MEMO), and *Event Member* (MEMV), can be considered as junction tables, as they set the relationship between users and the research team(s) they belong to. A *site* represents a broad area for which related information such as site investigation and post-earthquake observations are available. Although sites are assigned a latitude and longitude for the purpose of plotting them on a map, they may occupy an area rather than a point, which is often required in the documentation of case histories due to spatial variations of field performance (e.g., across the domain of a lateral spread) and geotechnical conditions. The site's geodetic coordinates are used only to plot the site on a map.

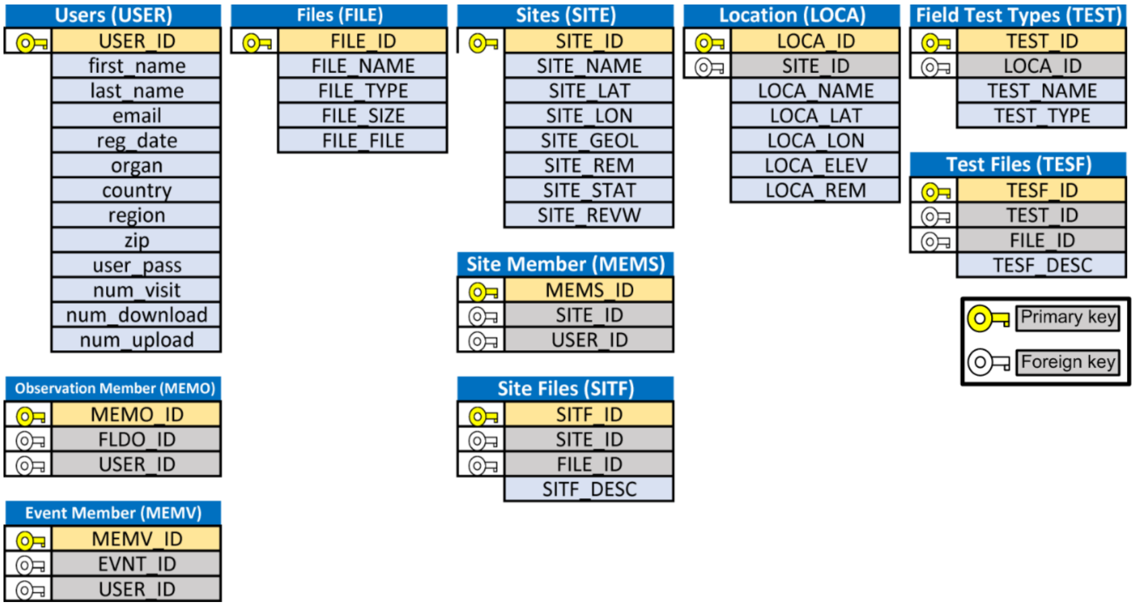

Figure 1. Relational database structure for general information.

A *location* is a specific geo-referenced point within a site where site investigations are performed or an observation is made. Many locations may be assigned to a single site. A single location may contain more than one field investigation. Each individual field investigation type is assigned as a single entry in the *Test* table. As an example, at a given location, a borehole and a downhole test may be performed. They will share the same location (i.e. the same coordinates), but they will belong to separate test table entries. In many cases, a file may be associated with a site or a location. The file itself is stored in the *Files* table; such files utilize the MySQL Binary Large OBject (BLOB) file type. BLOB file types are typically used to store large blocks of binary data such as pictures or videos. The *Site Files* and *Test Files* tables contain metadata about the files stored in the *Files* table. As an example, a geologic map describing the site can be uploaded to the *Files* table and associated with the site through the *Site Files* table. Keeping BLOB types in separate tables from other data fields can make queries operate more efficiently.

Figure 2 shows the database schema for tables describing site characterization information. For any location, users can input information about a borehole, test pit, sample, stratigraphy, water table elevation, cone penetration test, invasive geophysical test (cross-hole, downhole, suspension log), non-invasive geophysical test (surface wave methods), and soil description. Each of these fields has the TEST_ID as a foreign key. For any sample, users can enter information about a laboratory specimen or penetration resistance measurement. For a particular specimen, users can enter information about index tests, grain size distribution, Atterberg limits, relative density, or other lab tests. For an invasive geophysical test such as a downhole or suspension logger test, users input a profile of shear wave velocity and/or p-wave velocity. For a non-invasive geophysical test such as spectral analysis of surface waves (SASW), users must input the measured dispersion curve, and may also input any number of profiles consistent with the measured dispersion curve.

A subsection of the NGL database describes earthquake events, recording stations, fault segmentation, and recorded ground motions. It mirrors in some respects the data structure used in the NGA-West2 (Bozorgnia et al. 2014) and NGA-Subduction (Kishida et al. 2017) projects. Further details about interactions between the NGA databases and the NGL project will be provided in subsequent publications.

Event information may only be uploaded by Event Members, which is deliberately maintained as a small group to ensure proper coordination between the NGL and NGA data products. Regular users wishing to add an event can submit requests to NGL project staff. The Event Members designation only applies to event information (i.e., regular users will be able to upload information in all other sections of the database, including on site characterization and field performance information). Currently all 173 NGA-West2 and NGA-Subduction events have been uploaded, along with eight earthquakes from the Canterbury earthquake sequence.

Figure 3 contains tables for field performance information. The *Observations* table (FLDO) contains foreign keys for an event and a site because observations are made at a specific site following a specific event. Multiple earthquakes may shake a single site, giving rise to multiple observations. Users can enter a ground motion intensity measure that occurred at the site during a specific event in the *Ground Motion Intensity Measure at Site* (GMIM) table. Users must specify whether an entered ground motion intensity measure was directly measured at the site, interpolated from nearby ground motions using a spatial interpolation procedure such as kriging, estimated from a ground motion model, or obtained in another manner. These methods all exhibit different levels of uncertainty, which is important when developing models to describe liquefaction behavior. Users should select whether surface manifestation of liquefaction (or lack of evidence) was observed. This information is stored in the *Liquefaction Manifestation* table (FLDM). In case of liquefaction with surface manifestation (i.e. sand boil, lateral spreading, settlement, or structural damage), users can upload observations as files, such as field-maps or photos in the *Observation File* (FLDF) table. Furthermore, ground displacement vectors that are commonly measured (e.g., from lateral spreads or flow slides) can be entered in the *Displacement Vectors* table (FLDD).

USER INTERACTION

The NGL database is being populated as of this writing. The data uploading process is occurring through the interface developed at uclageo.com/NGL/. Aside from upload and download functionality, certain data visualization functionality is provided through the web interface. For example, users are able to plot site investigation data and earthquake event information, and view observation data.

We anticipate a higher-level of data access and analysis functionality will be required during the model development phase of the project. Accordingly, the database will be mirrored to DesignSafe (Rathje et al. 2017), where users will be able to query the database using, for example, Python scripts via Jupyter notebooks (Perez and Granger 2007).

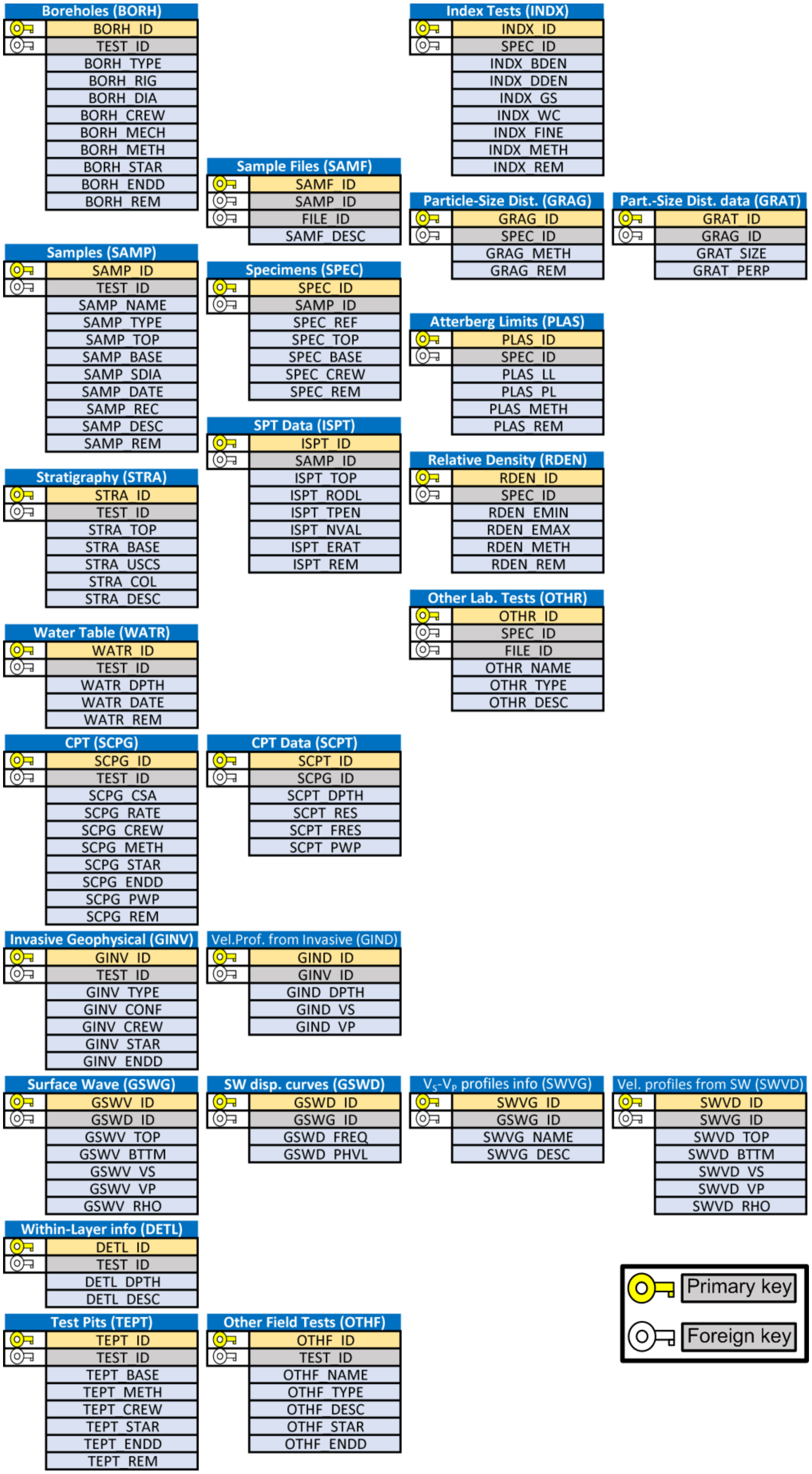

Figure 2. Relational database structure for site characterization.

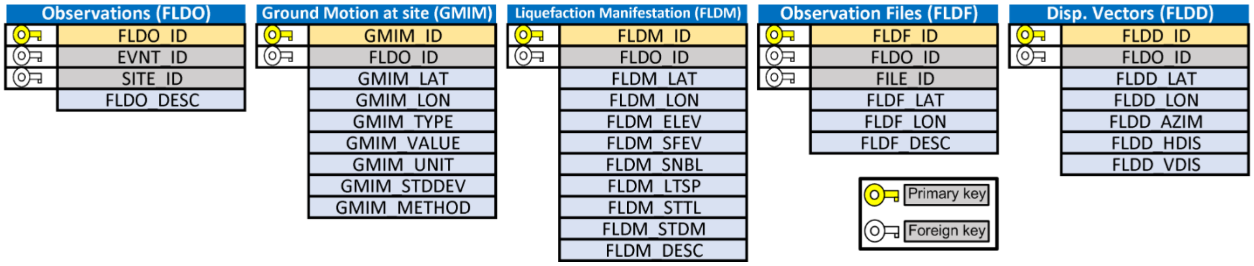

Figure 3. Relational database structure for post-earthquake observations.

CONCLUSION

This paper presents the organizational structure of the NGL relational database by providing a list of tables, fields within those tables, and relationships among the tables. The adoption of a relational database management system is a departure from the more common file repository approach adopted by many engineers in the natural hazards community. The RDBMS provides significant benefits over traditional file repositories.

ACKNOWLEDGEMENTS

Financial support for the NGL project is provided by California Department of Transportation (Caltrans) through the Pacific Earthquake Engineering Research Center (PEER), and by the U.S. Nuclear Regulatory Commission (NRC) through the Southwest Research Institute (SWRI). The authors would like to thank Tom Shantz of the Caltrans, Thomas Weaver of NRC, and John Stamatakos and Miriam Juckett of SWRI for their support of the project. The authors acknowledge the contributions by Mike Liu, Sjoerd Van Ballegooey, Kristin Ulmer, Russell Green, Mike Greenfield Christine Beyzaei, and Jonathan Bray for populating the database. We would also like to acknowledge the participants of the NGL workshop for their valuable input on the schema.

REFERENCES

Bozorgnia, Y., Abrahamson, N. A., Al Atik, L., Ancheta, T. D., Atkinson, G. M., Baker, J. W., Baltay, A., Boore, D. M., Campbell, K. W., Chiou, B. S.-J., Darragh, R., Day, S., Donahue, J., Graves, R. W., Gregor, N., Hanks, T., Idriss, I. M., Kamai, R., Kishida, T., Kottke, A., Mahin, S. A., Rezaeian, S., Rowshandel, B., Seyhan, E., Shahi, S., Shantz, T., Silva, W., Spudich, P., Stewart, J. P., Watson-Lamprey, J., Wooddell, K., and Youngs, R. (2014). "NGA-West2 research project," *Earthquake Spectra*, 30, 973–987.

Codd, E.F. (1972). Further normalization of the data base relational model. In Data Base Systems, Courant Institute Computer Science Symposia Series, vol. 6, R. Rustin, Ed., Prentice-Hall, Englewood Cliffs, N.J.

Kishida, T., Bozorgnia, Y., Abrahamson, N.A., Ahdi, S.K., Ancheta, T.D., Boore, D.M., Campbell, K.W., Darragh, R.B., Magistrale, H., and Stewart J.P. (2017). "Development of

NGA-Subduction database," *Proc. 16th World Conf. on Earthquake Eng.*, Santiago, Chile (Paper No. 3452).

Pérez, F. and Granger, B.E. (2007). "IPython: A System for Interactive Scientific Computing, Computing in Science and Engineering", 9(3), 21-29, doi:10.1109/MCSE.2007.53. URL: http://ipython.org

Rathje, E.M., Dawson, C., Padgett, J.E., Pinelli, J.-P., Stanzione, D., Adair, A., Arduino, P., Brandenberg, S.J., Cockeril, T., Esteva, M., Haan, F.L. Jr., Hanlon, M., Kareem, A., Lowes, L., Mock, S., and Mosqueda, G. (2017). "DesignSafe: A new cyberinfrastructure for natural hazards engineering." Natural Hazards Review. 18(3).

Stewart, J.P., Kramer, S.L., Kwak, D.Y., Greenfield, M.W., Kayen, R.E., Tokimatsu, K., Bray, J.D., Beyzaei, C.Z., Cubrinovski, M., Sekiguchi, T., Nakai, S., and Bozorgnia, Y. (2016). "PEER-NGL project: Open source global database and model development for the next-generation of liquefaction assessment procedures." *Soil Dyn. Earthquake Eng.*, 91, 317–328.