Final Report

EVALUATION OF EXISTING DEEP FOUNDATION PERFORMANCE USING THE FDOT DATABASE TO IMPROVE CURRENT DESIGN METHODOLOGIES

UF Project No.: 4910 4554 035-12

Contract No.: BD545, RPWO# 17

Submitted to:

Mr. Richard Long
Research Center
Florida Department of Transportation
Tallahassee, FL 32399-0450

Project Manager: Peter Lai

Principal Investigator: Michael McVay
Co Authors: Marc Hoit
Erica Hughes
Mark Styler
Thai Nguyen

July 13, 2005

DISCLAIMER

“The opinions, findings and conclusions expressed in this publication are those of the authors and not necessarily those of the Florida Department of Transportation or the U.S. Department of Transportation.

Prepared in cooperation with the State of Florida Department of Transportation and the U.S. Department of Transportation.”

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Evaluation of Existing Deep Foundation Performance Using FDOT Database to Improve Current Design Methodologies		5. Report Date July 2005 6. Performing Organization Code 49010 4554 035-12
7. Author(s) M.C. McVay, M. Hoit, E. Hughes, M. Styler, T. Nguyen		8. Performing Organization Report No.
9. Performing Organization Name and Address University of Florida Department of Civil Engineering 345 Weil Hall / P.O. Box 116580 Gainesville, FL 32611-6580		10. Work Unit No. (TRAIS) 11. Contract or Grant No. BD545, RPWO# 17
12. Sponsoring Agency Name and Address Florida Department of Transportation Research Management Center 605 Suwannee Street, MS 30 Tallahassee, FL 32301-8064		13. Type of Report and Period Covered Final Report 14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The Florida Department of Transportation funds thousands of insitu and geotechnical laboratory tests every year. Bridge foundations often require load tests during construction. Electronically preserving the resulting data enables reusability in future FDOT projects, including bridge maintenance.

Utilizing an internet based geotechnical database to digitally preserve the data enhances the value of the data in many significant ways. Based online, geotechnical consultants will be able to directly upload results derived under FDOT contracts. A relational database enables searching of the data in order to form meaningful and relevant datasets. A geotechnical XML schema was developed to handle the communications between the users and the database.

The XML schema was used to develop Microsoft Excel spreadsheets that could upload and download from the database. The geotechnical XML schema will also promote the development of commercial software for consultants.

The database was populated with various completed projects within the state of Florida. This was accomplished with the previously mentioned Excel spreadsheets. This data was subsequently used to calibrate LRFD resistance factors for the BSI developed FB-Deep software program.

17. Key Word	18. Distribution Statement
Database, Security, XML Schema, Web Interface, LRFD, Driven Pile, Excel Spreadsheets	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
		191	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

ACKNOWLEDGEMENTS

The authors wish to thank the Florida Department of Transportation (FDOT) for providing the financial support under Contracts BC-354 RPWO#45 for making this study possible. Also the assistance of Mr. Peter Lai (Assistant State Geotechnical Engineer), Dr David Horhota (State Materials Geotechnical Engineer) and Mr. Larry Jones (State Geotechnical Engineer) as well as the State Geotechnical Engineers for individual project information is great fully acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

Currently, the Florida Department of Transportation (FDOT) conducts thousands of insitu tests (SPT, CPT, etc.), as well as laboratory strength and compressibility tests each year for existing (e.g. maintenance), planned or under construction infrastructure components (roads, bridges, etc.). In addition, FDOT installs thousands of piles and drilled shafts for bridge widening and/or replacement of existing structures (e.g. bridges, sign/lighting, walls, etc.). Significant amounts of the latter data (e.g., insitu, laboratory, as built, etc.) are reusable on existing (i.e. maintenance), and future projects (e.g. widening, nearby structures, etc.), as well as improve future design and construction. For instance, load testing of deep foundations (i.e. conventional top down, dynamic-statnamic, or bottom up – Osterberg) under current construction may be used improve future LRFD resistance factors, ϕ. Similarly, variability of measured pile/shaft capacities in multiple Florida sites may help in identifying the safe maximum LRFD resistance factor, ϕ for non-redundant foundations.

With the development of personal digital assistants (PDA), rugged laptops, etc., the FDOT is moving away from paper records of field (i.e. construction), and laboratory (e.g. strength tests) data to electronic formats, databases (i.e. LIMS, PEEDS, etc). Also, the data recorded or generated by one group (e.g. design) may be useful to others (e.g. construction, maintenance, etc.). Consequently, there is great need to standardize transportation data for use within the DOT, i.e. district to district, as well as by different consultants, software vendors, etc.

Since it is expected that viewing, transferring, etc. will be occurring over the Internet, the use of XML (eXtensible Markup Language) would be language of choice, given its standardization by browsers (Microsoft), viewers (Adobe), etc.   In addition, XML schema are currently being developed a number of vendors (Bentley, Infotech, etc.) for multiple transportation arenas (CADD, Maintenance, etc.). Recently, NCHRP (National Cooperative Highway Research) has awarded a multiyear contract (NCHRP 20-64) to develop XML schemas for the transportation field (i.e. TransXML). Unfortunately, the latter did not consider the Geotechnical Area within Civil Engineering.

This effort concerns the development of Geotechnical XML schema, Excel spreadsheets to collect Geotechnical Laboratory and Field Data (Insitu, and Bridge Substructures) as well as the development of a web based FDOT database with multiple levels of security.   The transferring of data in and out of the database, i.e. Excel spreadsheets, existing software (e.g. FB-Deep), etc was handled with a DLL which parses the XML data with the login security. The FDOT database was subsequently used to evaluate the current deep foundation axial capacity design software, FB-Deep, by developing LRFD, resistance factors, ϕ, for variable reliabilities, β.   A detailed scope of work follows.

1.2 SCOPE

The research focused on the following four tasks:

Task 1 – Develop a data structure and XML schema for FDOT's Bridge Substructure Data. The latter is setup in a hierarchy relationship. The highest hierarchy is the project ID followed by the bridges, and holes. The holes represent insitu data (e.g. SPT, pressuremeter, etc.) as well as samples from which laboratory testing (i.e. classification, soils and rock strengths, etc.) is preformed. The bridge has associated substructure information: piers, piles, and shafts. The latter includes both design (e.g. cross sections, etc.) or as built data (lengths, driving information, load tests, etc.) recovered during construction. All of the information within the Data Dictionary has an associated XML schema for transfer from software (i.e. Excel Spreadsheets, etc.) to the FDOT database.

Task 2 - Excel spreadsheets were to be written for reducing all the data from the laboratory (e.g., compressibility, strength, etc.) or the field (e.g., Osterberg, Statnamic Tests, etc.). Generally, the data recorded and reduced (e.g., pile/shaft capacities, soil/rock properties, etc.) are printed and turned in as a report during the design or construction phase depending on the activity. However, with the proposed architecture, the information (e.g., Excel spreadsheet) may be uploaded directly by the consultant to the database for later use on this or another project. Four general purpose Excel spreadsheets were to be developed for this process: 1) Insitu testing (SPT, CPT, Pressuremeter, etc.); 2) Design (Pile/Shaft geometries, analysis, etc.); 3) As Built (Pile/Shaft geometries, construction methods, etc.); and 4) Load Testing (Driving, Osterberg, Statnamic, etc.). Each of the spreadsheets may be used to load or retrieve data from the database.

Task 3 – Development of a database in SQL with Microsoft IIS in the .NET environment taking advantage of ODBC connections. The data was to be arranged in Tables having hierarchy as identified in the Data Dictionary (Project, Bridge, Hole, etc.). The web server was to needed for all file intelligence and security, and ODBC was used for the database connection. A DLL was written to parse the Geotechnical XML schema data (Task 1), as well as security when communicating with the database.

Task 4 - All of the FDOT Deep Foundation Data currently in Microsoft Access files, Spreadsheets, etc. was to be subsequently uploaded into the new Internet accessible (FDOT server) database. Using the DLL developed in Task 3, the FDOT FB-Deep Software would be used access the new database for insitu data, and predict pile capacities. Also using the recorded Load Test capacities in new database, along with the predicted FB-Deep axial capacities new FDOT LRFD resistance factors, ϕ, for various reliabilities, β will be found.

The following chapters describe the work accomplished for the identified tasks. In addition, the Appendices include the complete Geotechnical Data XML Schema, developed Excel spreadsheets, as well as new FDOT Database Projects.

CHAPTER 2

GEOTECHNICAL DATA DICTIONARY & XML SCHEMA

2.1 INTRODUCTION

Prior to 1991, Geotechnical Data transfer between consultants, highway agencies, laboratories, etc. within Europe and Asia, occurred through a multitude of formats, which was dependent on the vendor. In 1991, the Association of Geotechnical and Geoenvironmental Specialist (AGS) was formed to establish an interchange format to allow transfer of data between systems (software, databases, etc.) with minimal change to the systems themselves. The AGS format is an ASCII flat file format employing a Data Dictionary approach using Groups and Fields to delineate specific elements of data. For instance, under a project would occur all associated information: hole details, strata details, and laboratory data. The latter is referred to as a hierarchical approach, allowing for as little or as large a quantity of information transfer as necessary, minimizing the likelihood of lost information, e.g. widow or orphaned data - SPT, and laboratory information with no physical location. Recently (2005), AGS has begun transferring the ASCII file format to a XML schema, which is GML compliant (AGS version 3.1). As identified earlier, EXtensible Markup Language (XML) is used throughout the Internet, for browsers (Microsoft), viewers (Adobe), etc.

Within the United States, there currently doesn't exist an agency/standard like AGS for Geotechnical Data. However, within the US, a number of state and federal agencies have initiated research (NCHRP 20-64) to develop XML schema for transportation related data (TransXML). Recently, a number of DOTs (Florida, Virginia, Maryland, etc), Federal Agencies (FHWA, EPA, etc.), and National Organizations [COSMOS (Consortium of Organizations for Strong-Motion Observation Systems)] have started the process of developing a national consensus for a Geotechnical Data Dictionary with associated XML schema (e.g. FHWA pooled fund study). For instance, shown in Figure 2.1 is the Geotechnical Data Structure/Model proposed by COSMOS (Benoit, 2004) for Geotechnical Field/Laboratory Information. Note the hierarchical similarities of Fig. 2.1 with AGS structure. Unfortunately, neither deal with bridge pier substructure geometries, construction or load testing. Consequently, the latter is the focus of section 2.2, with an insitu and laboratory data/structure similar to the AGS and COSMOS approach.

2.2 DEEP FOUNDATION DATA STRUCTURE AND DICTIONARY

For successful foundation design, and construction, the FDOT records/monitors (QA/QC) significant amounts of deep foundation information. For instance, insitu boring, laboratory strength, as well as construction load testing is typically performed on every site. A significant amount of this data is of value for improved design (LRFD resistance factors), maintenance (e.g. scour), nearby construction (i.e. insitu, laboratory) or even potential future bridge widening (e.g. as built), etc. To collect the data and provide accessibility, the following data structure and names (dictionary) was developed.

Insitu testing, i.e. SPT borings, CPT data, etc, is connected with a hole, which is associated with subsurface investigation for a specific project as shown in Figure 2.2.

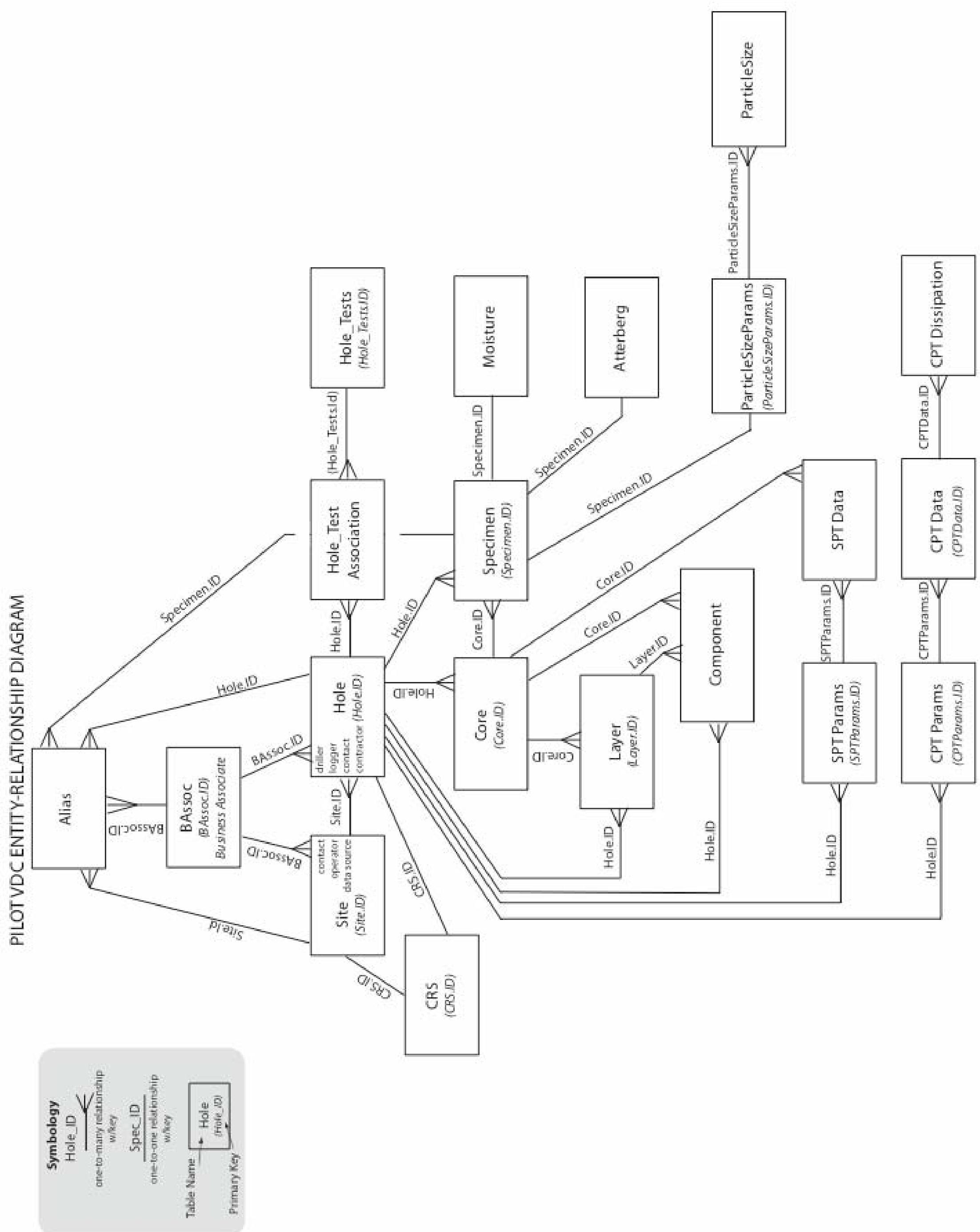


Figure 2.1 Geotechnical Data Structure, COSMOS (Benoit, 2004)

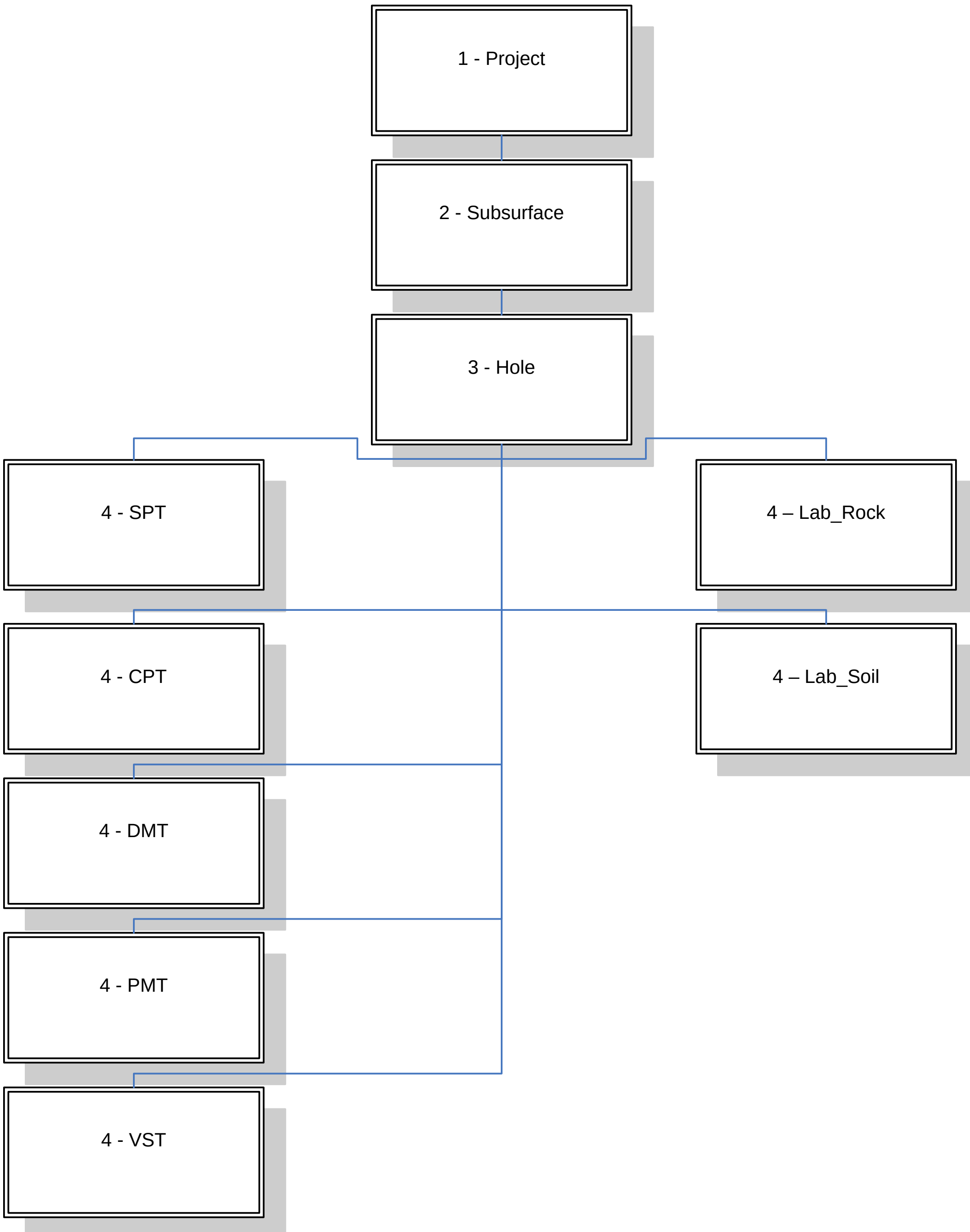


Figure 2.2 FDOT Insitu and Laboratory Data Structure

Also associated with a specific hole, Fig 2.2, are laboratory samples obtained from the SPT sampler, Shelby Tube (clay), core barrel (rock), etc. Presented in Figure 2.3 are the various types of Insitu Data collected in the database: Standard Penetration Testing (SPT), Cone Penetration Tests (CPT), Pressuremeter Tests (PMT) and Vane Shear Tests (VST). Also collected for each test is the associated metadata, which identifies types of spoons, cones, etc. used to obtain the data or reduce it. Evident is the hierarchal structure of the data, i.e. Sample → SPT → Hole → Project.

Presented in Figures 2.4 and 2.5 are the laboratory tests for both soil and rock collected in the database and its coupled structure. Of strong interest are the soil classification, strength, as well as rock strengths. New features are the Young's Modulus of the rock and scour potential of the rock from erosion laboratory experiments.

Shown in Figure 2.6 is the bridge, pier, pile, shaft and field-testing data structure for a specific project. The structure follows the organization of a bridge, as well as the chronological construction process. For instance, specific pile/shafts elements are components of a pier, and associated with any pile/pier are its design and construction information. Of interest, Figure 2.7 are the loads, geometry, etc. used for design, whereas from construction, final lengths, soil/rock conditions, and capacities are of interest. In the case of capacities, the database covers, pile driving capacity assessment (PDA, CAPWAP), Osterberg, Statnamic, as well as conventional top down static load testing. Figure 2.8 shows the top down conventional load test data structure, and shown in Figure 2.9 is the data structure for the Osterberg test. The complete data structure for the FDOT Deep Foundation Database is given in Appendix A.

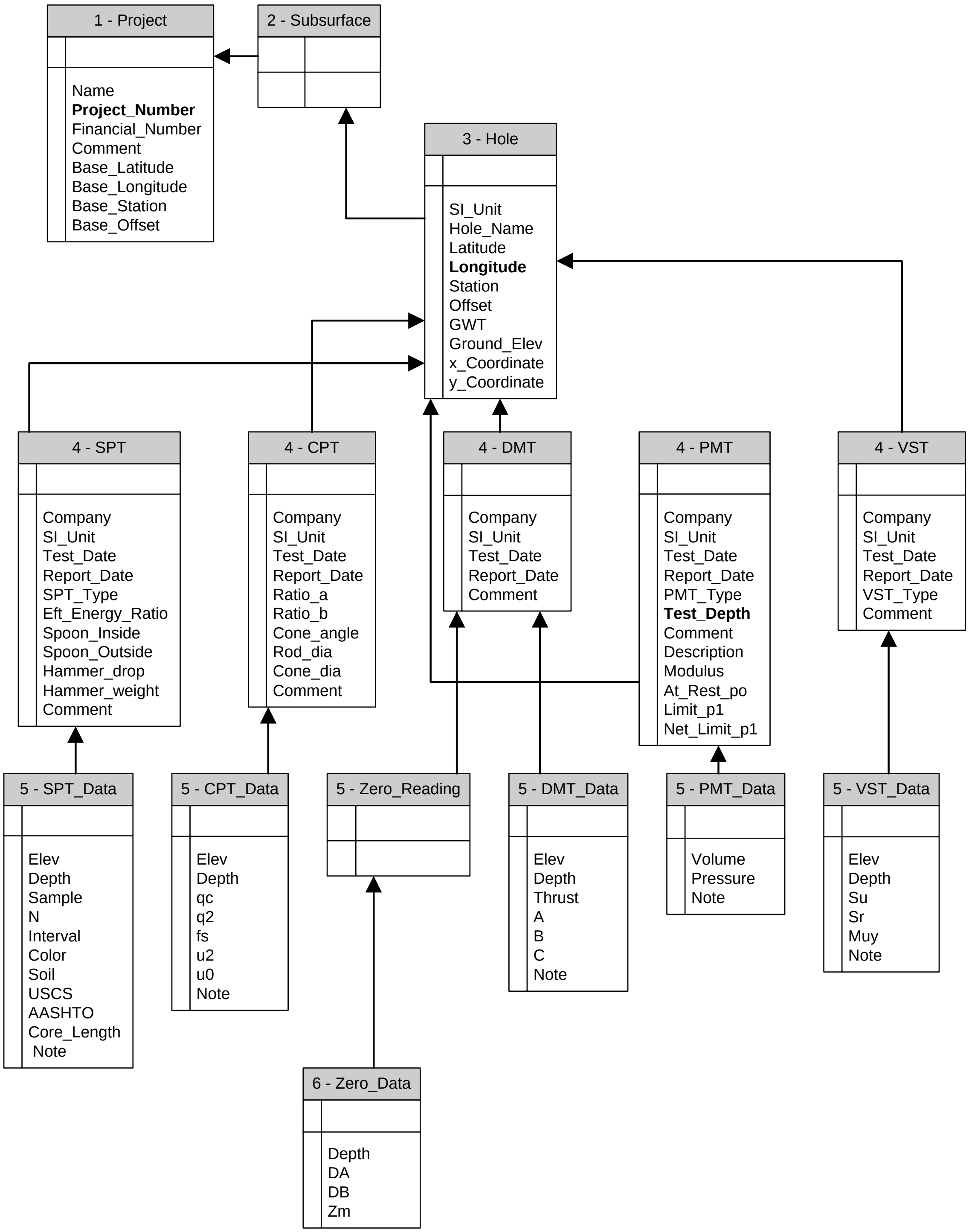


Figure 2.3 Insitu Data (SPT, Cone, DMT, and VST) Structure

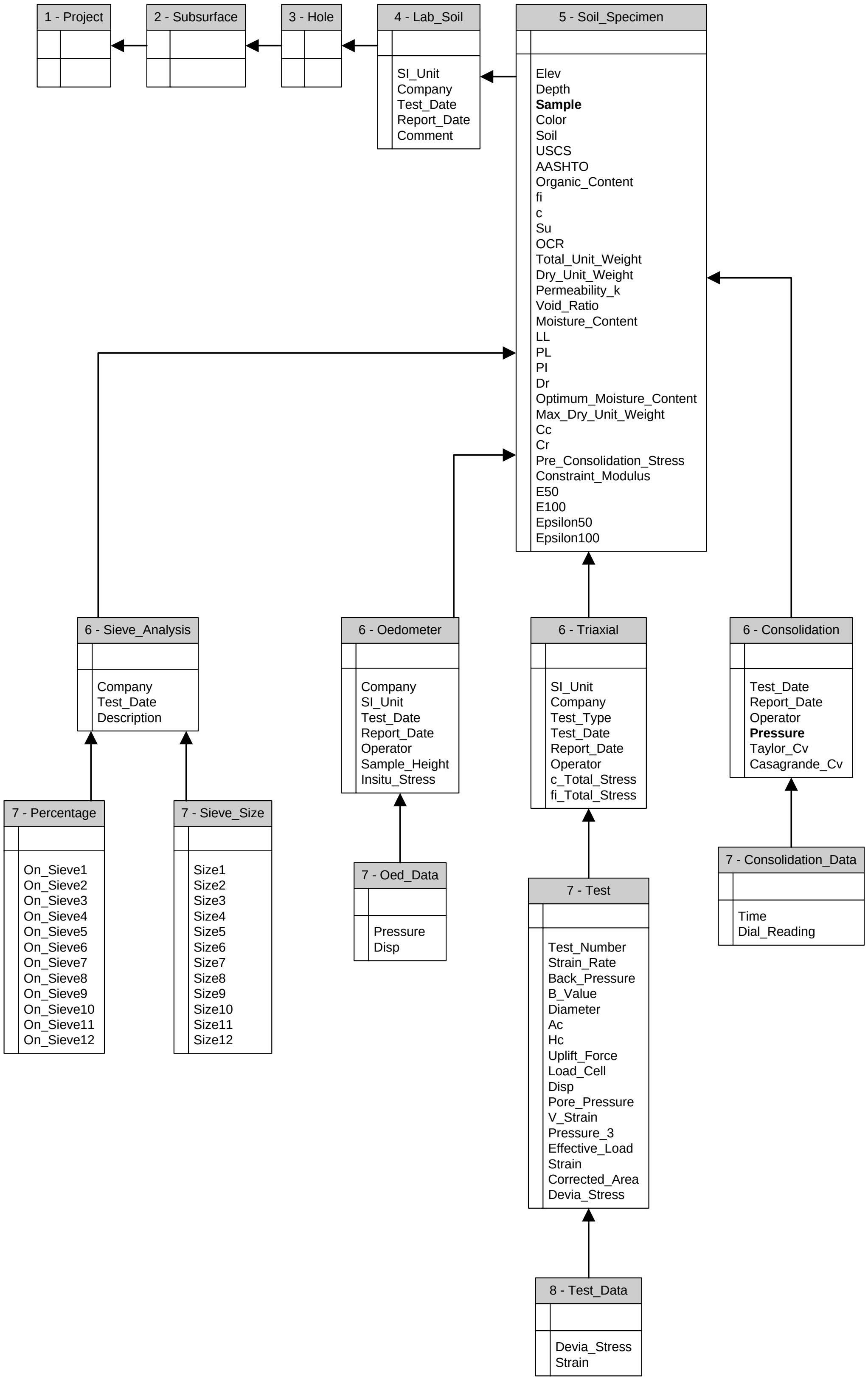


Figure 2.4 Laboratory Soil Test Data Structure

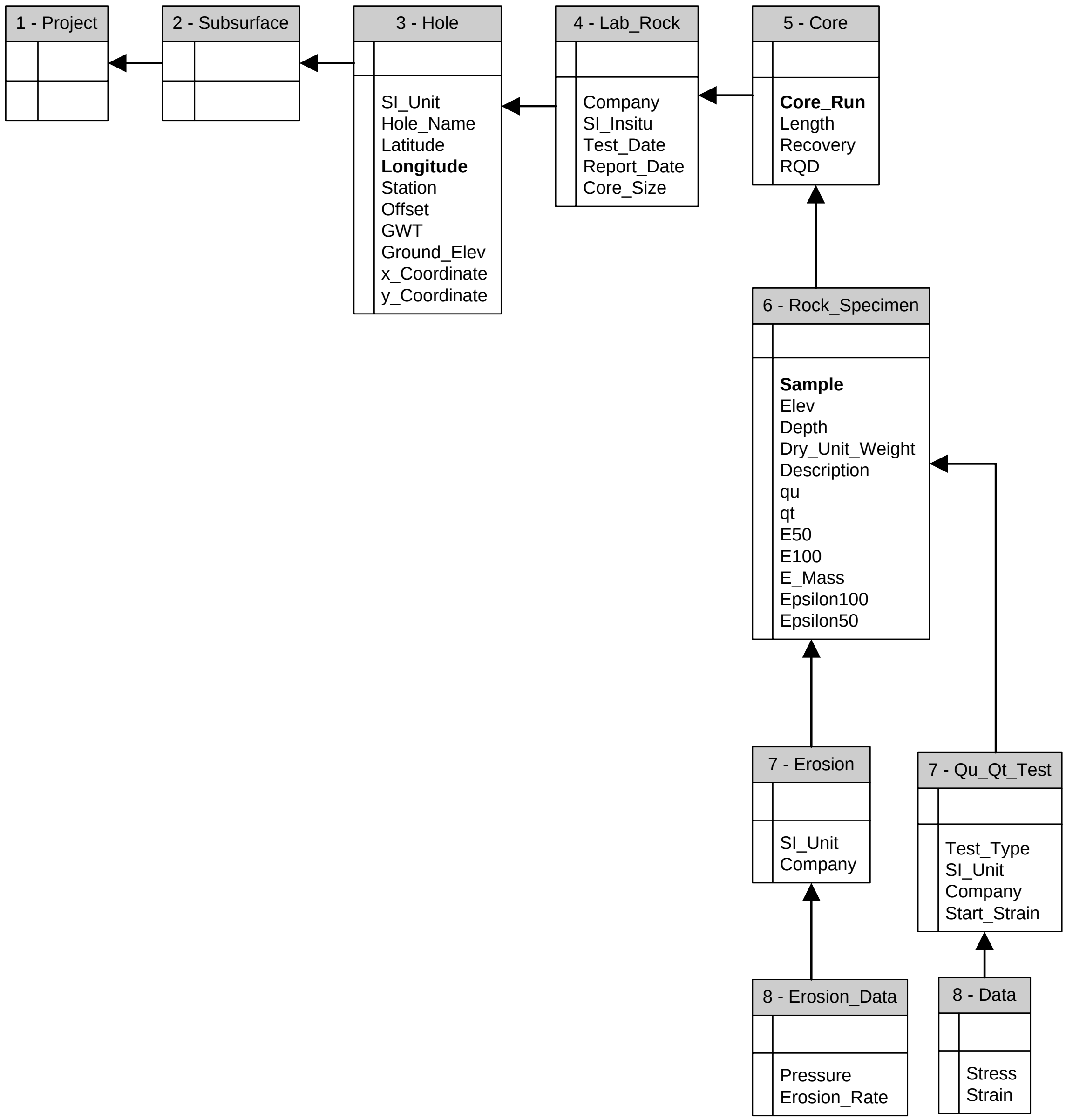


Figure 2.5 Laboratory Rock Testing Data Structure

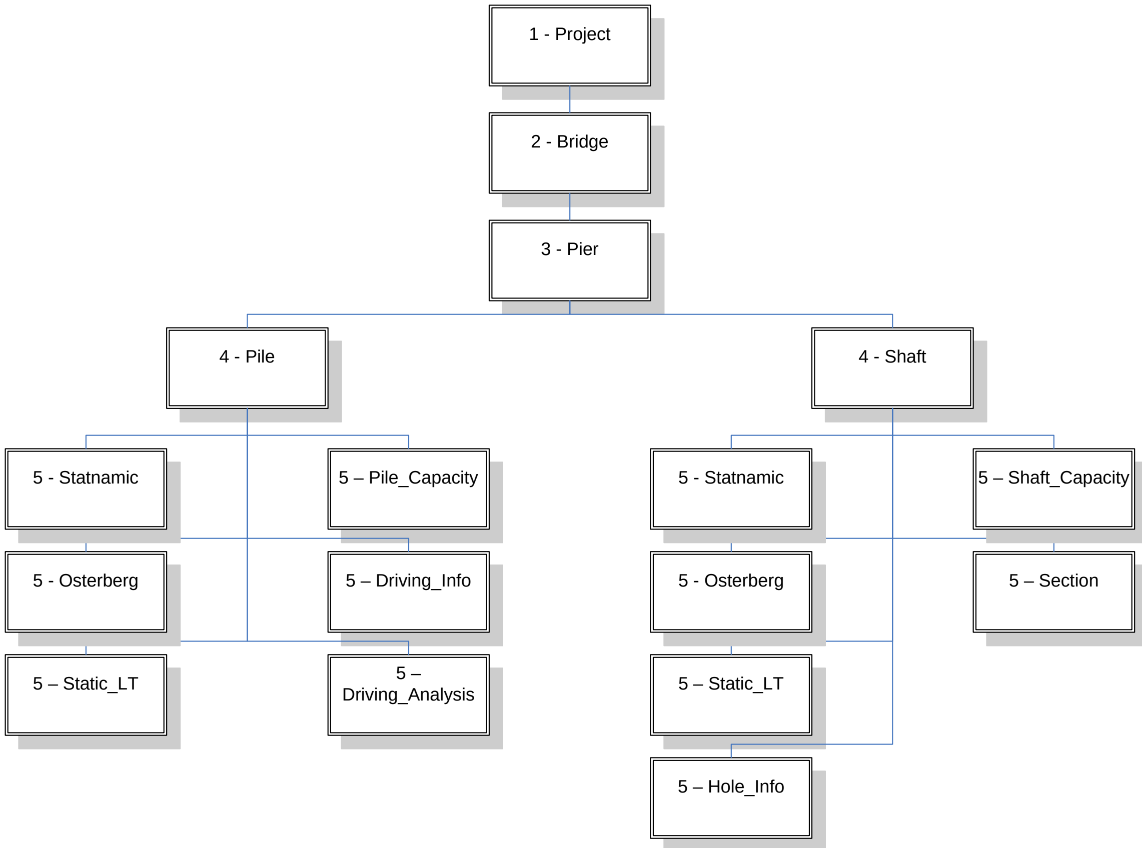


Figure 2.6 FDOT Bridge-Pier-Pile/Shaft-Testing Data Structure

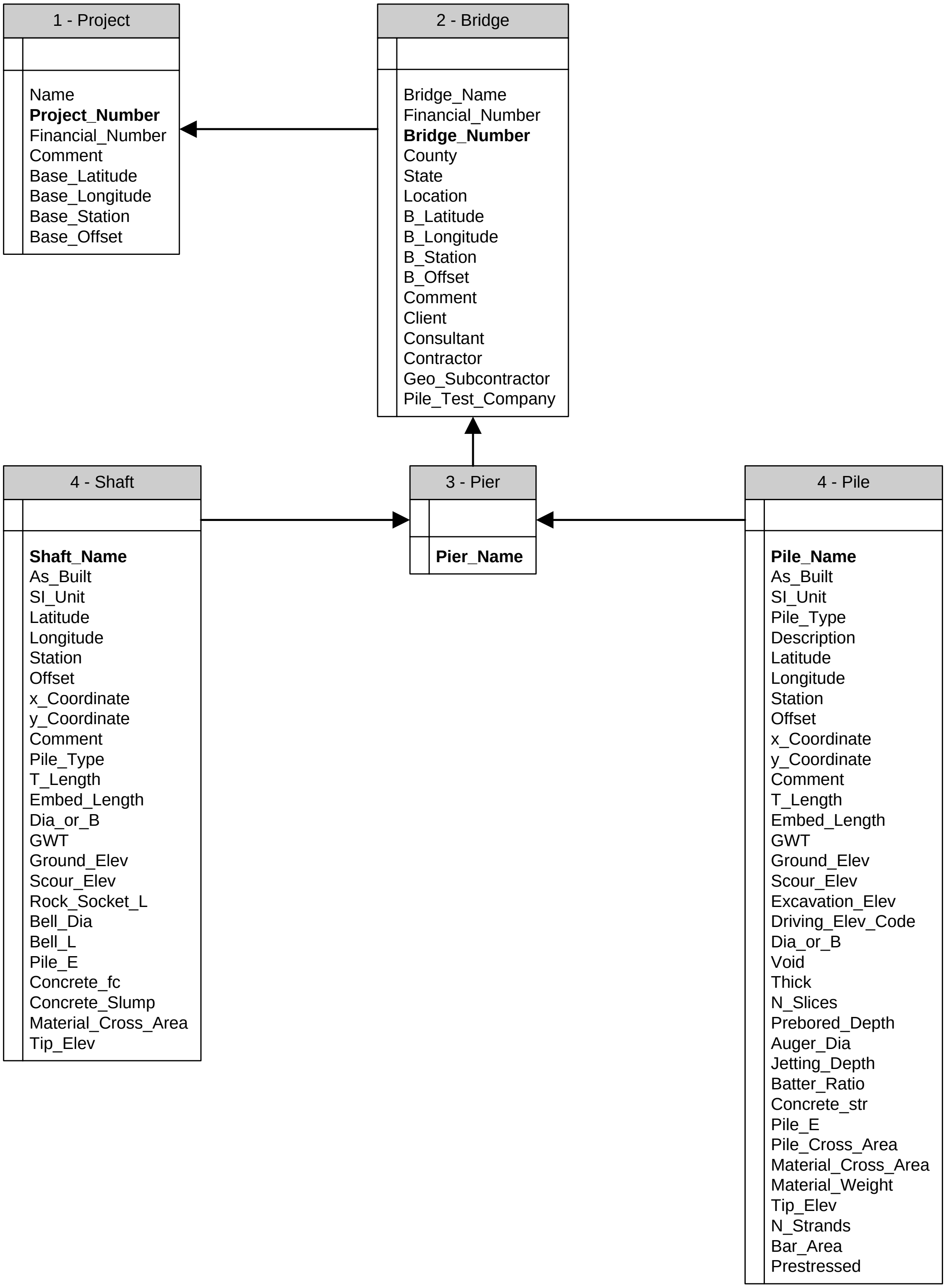


Figure 2.7 Bridge-Pier-Pile/Shaft Design and As Built Information

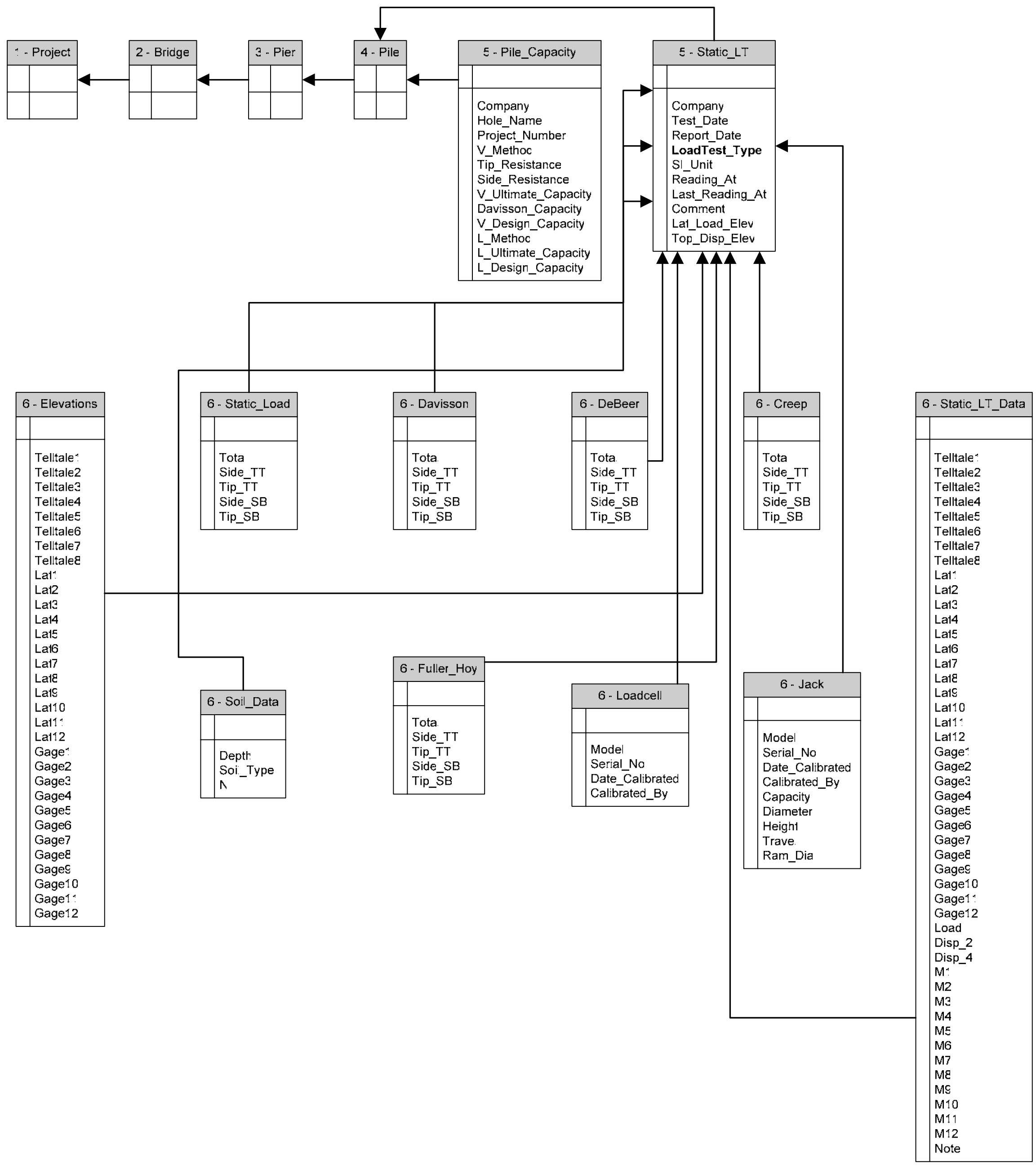


Figure 2.8 Data Structure for Conventional Static Top Down Load Test

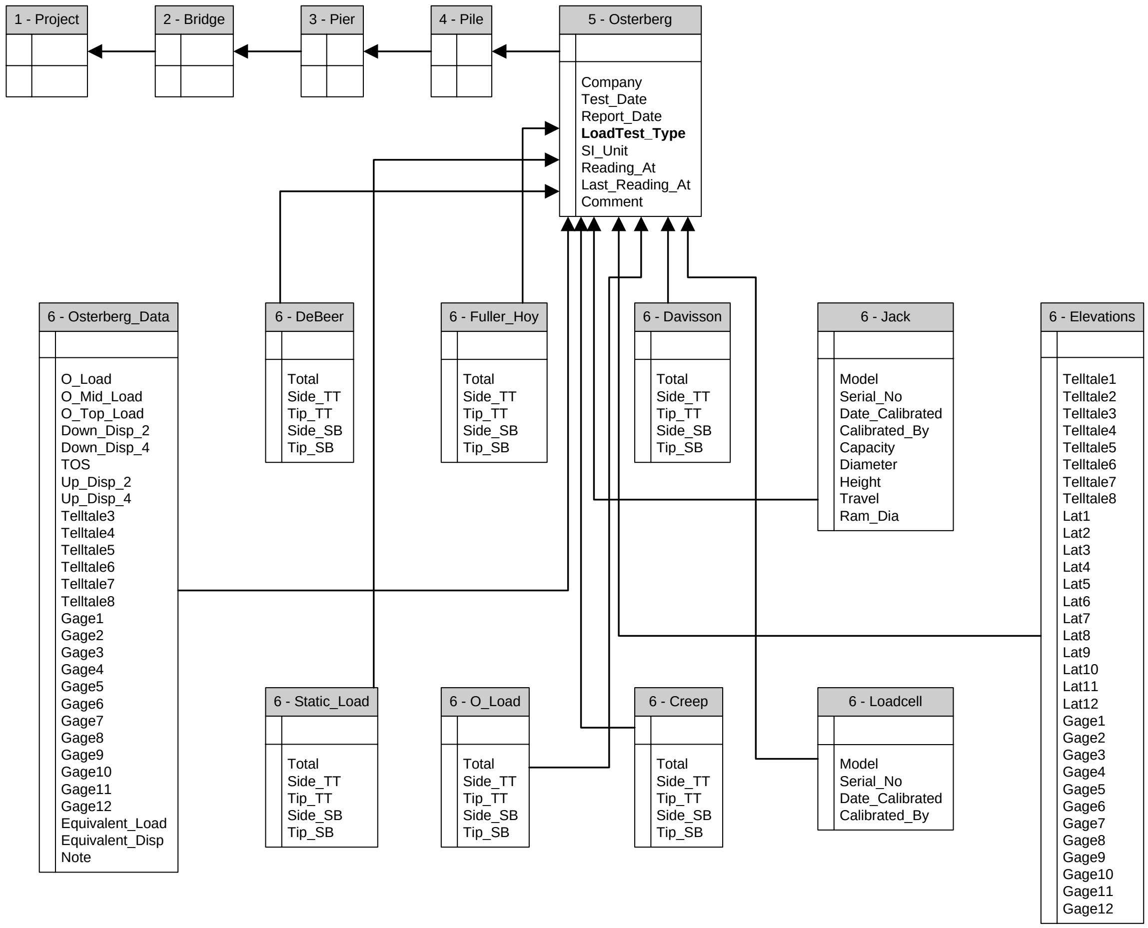


Figure 2.9 Osterberg Bottom Up Data Structure

2.3 DEEP FOUNDATION XML SCHEMA

Once the data structure/dictionary (section 2.2) for the database has been established, the XML tags identifying the data and associated metadata (i.e. cone dimensions, SPT energy, etc.) may be formalized.

XML elements (Figure 2.10) consist of a start and closing tag, multiple optional attributes, optional character data content, and sub-elements. The **start** and **end tags** (Fig. 2.10) names must be unique and are case-sensitive. The **element** can be a container for other elements or it can contain **character data** (Fig 2.10). An **attribute** is similar to an array and contains information associated with the tag.

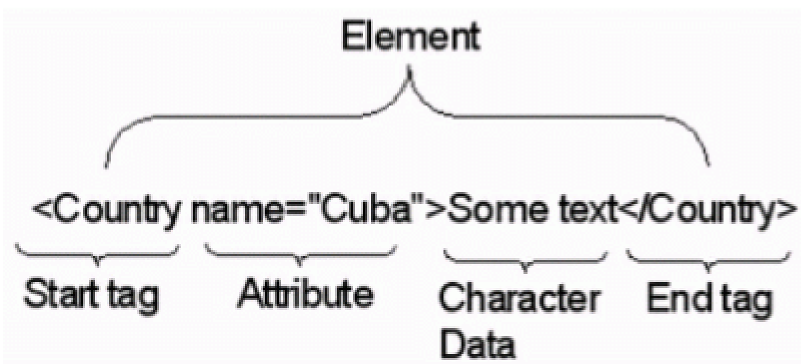


Figure 2.10 Standard XML tag

An example of XML tags for SPT data for Boring B-1, at Station 221+90 in Port Orange Florida (Project No. 79180-3502) is given in Figure 2.11. Evident is the legibility of the tags, and associated data. Also, note the nesting of the XML tags as identified in the data structure of Figure 2.3 for SPT data. The first line of Figure 2.11 is called the DTD or Document Type Definition (DTD). The purpose of a Document Type Definition is to

```
<?xml version="1.0" ?>
 - <Project Name="Port Orange Intercostal and Relief Bridges" Project_Number="79180-3502">
   - <Subsurface>
     - <Hole SI_Unit="False" Hole_Name="B-1" County="Volusia" Ground_Elev="6.3" Station="221+90" Offset="-20">
       - <SPT SI_Unit="False">
           <SPT_Data Elev="4.8" Depth="1.5" N="13" Predescript="brown" Soil="shelly Sand" />
           <SPT_Data Elev="0.3" Depth="6" N="16" Predescript="brown" Soil="shelly Sand" />
           <SPT_Data Elev="-4.7" Depth="11" N="1" Predescript="gray, with clay" Soil="shelly Sand" />
           <SPT_Data Elev="-9.7" Depth="16" N="29" Predescript="compact" Soil="shelly Sand" />
           <SPT_Data Elev="-14.7" Depth="21" N="26" Predescript="trace shell" Soil="Sand" />
         </SPT>
       </Hole>
     </Subsurface>
   </Project>
```


Figure 2.11 Example XML Tags: SPT Data for Port Orange, Project Number 79180-3502 define the legal building blocks of an XML document. It defines the document structure with a list of legal elements. A complete list the DTD for all the XML tags, attributes, etc. for the database is given in Appendix B. Next, a number of Excel spreadsheets showing lab and field data reduction as well as uploading or downloading of the data to the FDOT database through the web are presented.

CHAPTER 3

EXCEL DATASHEETS

3.1 INTRODUCTION

As identified in chapter 2, the FDOT database basically covers the following areas: Insitu, Laboratory, Design, Construction, and Load Testing. Since each may be performed by different consultants/engineers, it was decided to split the data collection/analysis into five separate spreadsheets: 1) Insitu; 2) Laboratory; 3) As Built (Construction); 4) Design; and 5) Load Testing. Each spreadsheet is capable of recording raw data, reducing it when necessary (e.g. rock strengths, moduli, etc.), and uploading it into the database. Similarly, a user (e.g. FDOT personnel, consultant, contractor, etc.) is capable of downloading the data for a specific project into any spreadsheet from the database with the appropriate user ID and password. All of the Excel spreadsheets are open source, easily modifiable and free. A discussion of each follows.

3.2 INSITU DATA

The Insitu Excel Spreadsheet covers all typical FDOT insitu testing: Standard Penetration Testing (SPT), Cone Penetration Testing (CPT), DilatoMeter Testing (DMT), Pressuremeter Testing (PMT), and Vane Shear Testing (VST) as shown in Figure 3.1. The "general" tab dialogue sheet (Fig. 3.1) identifies the types of insitu data (e.g. SPT, CPT, etc.) available over a project site as well as their physical location relative to one another. Note the locations may be by Station Number, GPS, or XY coordinates.

Associated with Insitu Data (Fig. 2.3) are Project Name and Number, Bridge Number, Financial Number, and any comments.

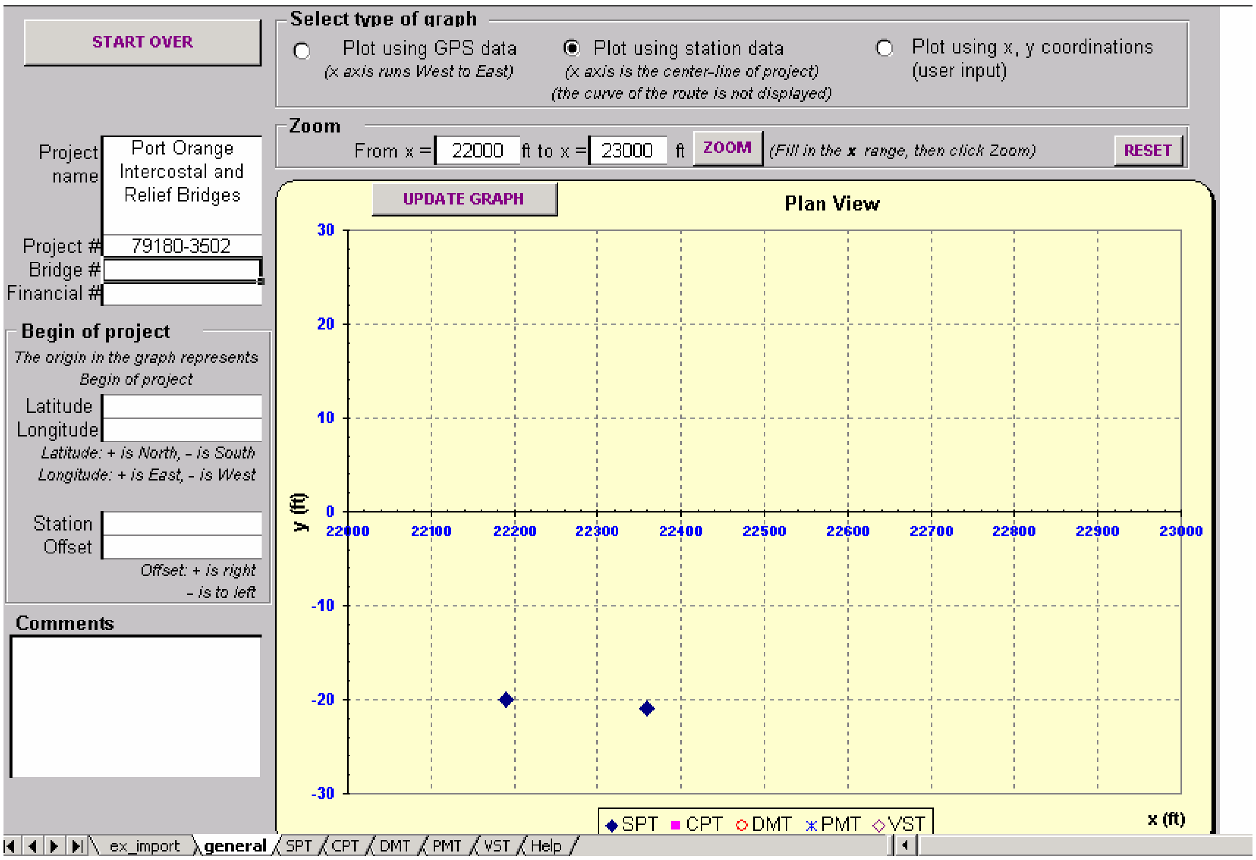


Figure 3.1 Insitu Spreadsheet: General Information

Shown in Figure 3.2 is the data collected for one SPT boring. Different borings are selected from the slide bar on the top left of the sheet (i.e. 1 of 2). On the left is metadata describing the equipment: spoon, hammer, drop system, energy, etc. In the middle are the depth, elevation, SPT “N” values, and soil descriptions. The drop down menu on soil description is the primary soil type as identified in FDOT's Soil and Foundation Manual. On the right is the laboratory classification (USCS & AASHTO), which would be populated if the data were downloaded from the database (i.e. includes laboratory data). Not shown on the right is a plot of SPT "N" values vs. depth.

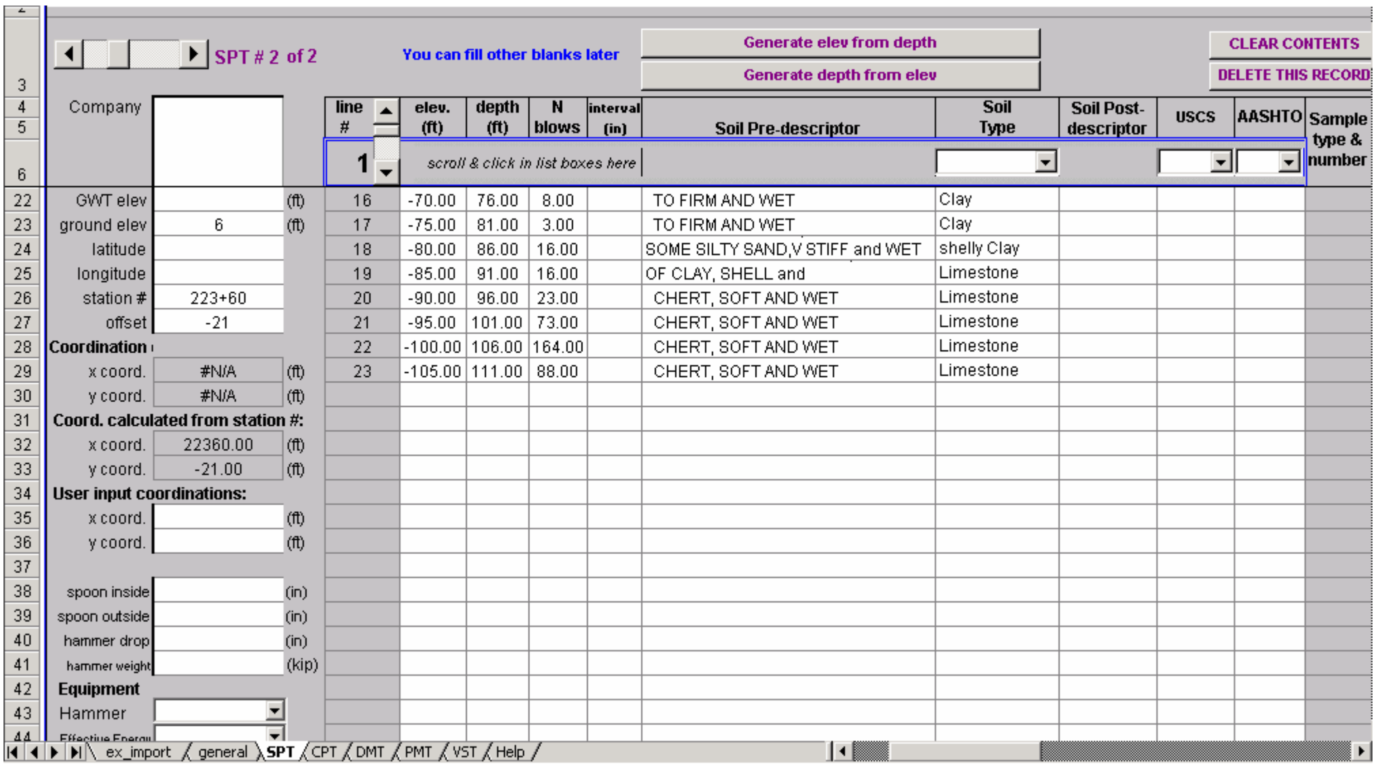


Figure 3.2 SPT Data: Spoon Dimensions, Depth, Blow Count, and Soil Description

Presented in Figure 3.3 is the datasheet for the CPT information. Again on the left is the metadata: Project information, cone sizes, etc., along with a slide bar (top left) identifying a specific sounding on the site. Shown in the middle is the raw data, and on the right is a plot tip resistance and sleeve resistance vs. depth. Presented at the top of the sheet are automatic import buttons to import data from Hogentoggler, Pagani, etc. data files.

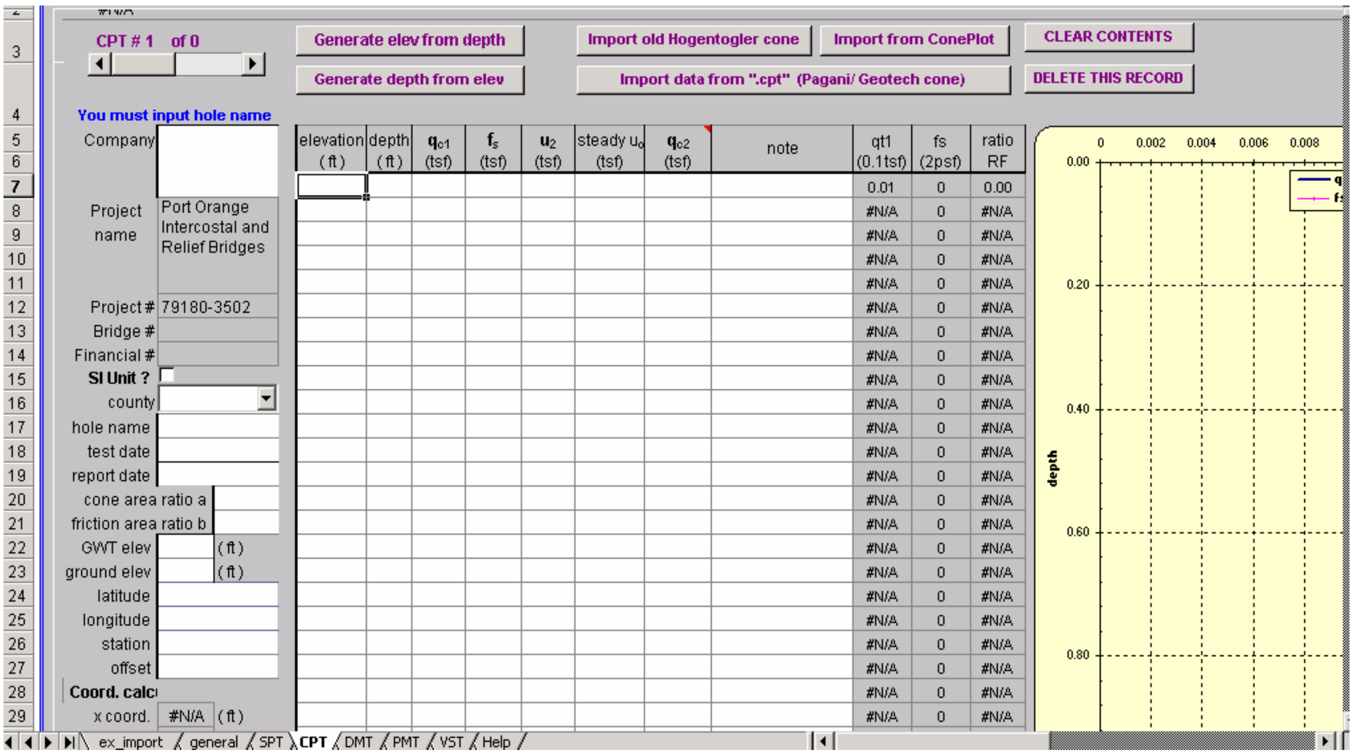


Figure 3.3 CPT: Depth, Tip Resistance, Skin Friction, Friction Ratio, and Pore Pressure

Shown in Figure 3.4 is the DMT Data Sheet. Standard DMT information on initial reading, i.e. delta A & B, Z_m is presented on the left for a specific sounding (top left). Depth, Thrust, A, B, and C are shown in the middle, and a plot of A and B is given on the right. Note, all user input are shown with white cells, all computed output (i.e. K_d, I_d, and Ed) are shown with grey cells.

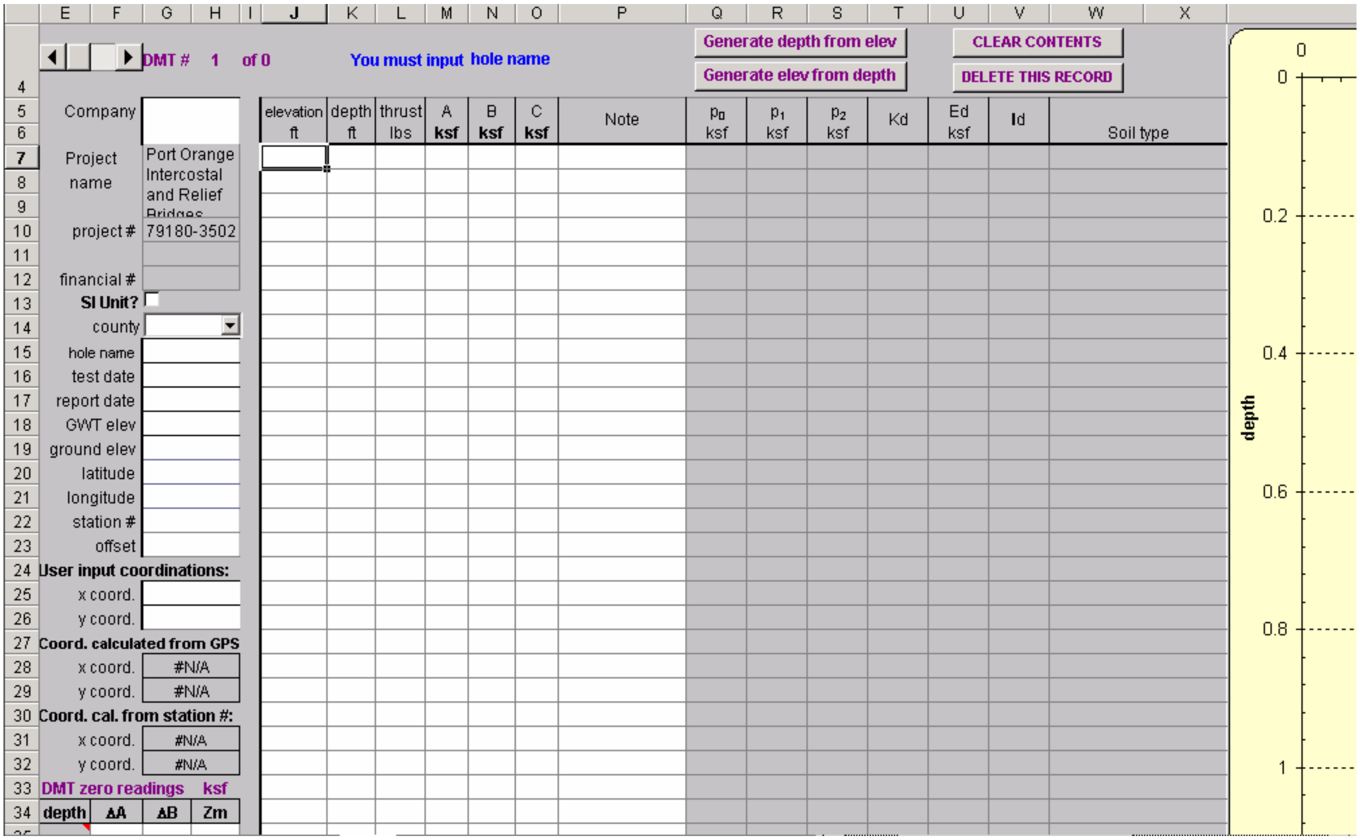


Figure 3.4 DMT Data: Zero Readings, Thrust, A, B, and C Readings vs. Depth

Presented in Figure 3.5 is the Pressuremeter Tab Data input. Required information is the hole location, and corrected pressure vs. volume information for a specific test at a given depth. Note all Insitu information (SPT, CPT, etc.) allows for either SI or English set of units.

3.3 LABORATORY TESTING

Presented in Figure 3.6 is the general-purpose tab for the Laboratory Collection/Analysis Spreadsheet. It allows the user to open/save an existing file from hard disk or the database, as well as upload collected/reduced data to the database.

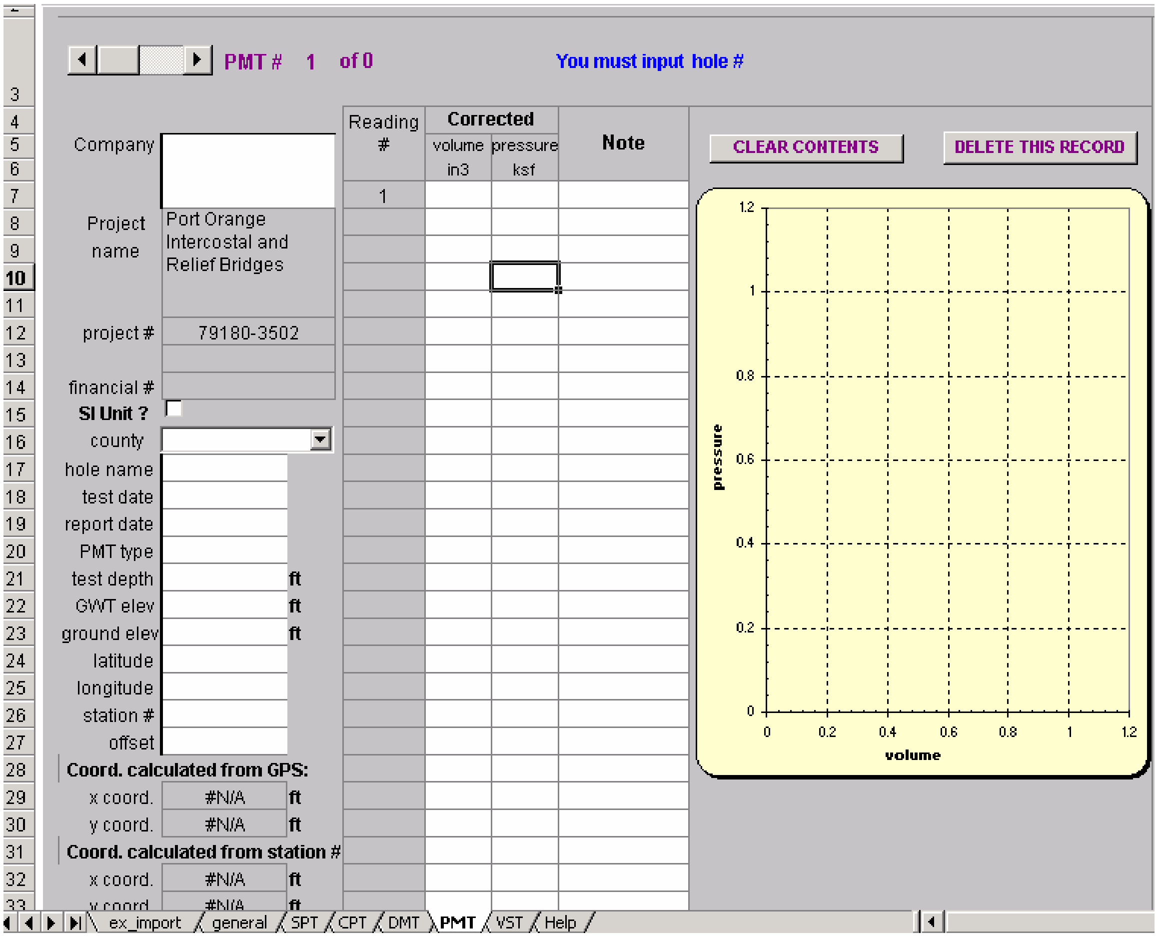


Figure 3.5 Pressuremeter Data: Pressure vs. Volume

Shown in Figure 3.7 is the general soil information referenced to a specific sample, hole, bridge, project and financial number. The soil classification, strength, compressibility, permeability, organic content, etc. is determined in the other tab sheets, but entered here for general comparison. For instance, for soil classification, the grain size laboratory sieve analysis would be performed and entered in Figure 3.8. Using the latter with the Atterberg limits entered in Fig. 3.7, the USCS and AASHTO soil classification would be entered by the engineer/technician in Soil Tab sheet for a given sample and depth.

In the case of soil compressibility of clays, a 1-D consolidation or oedometer test would be preformed. For a given load increment, the Tab Consolidation Data sheet, Figure 3.9 would be used to record the time vs. dial reading. From deformation, the void ratio change as a function of effective stress is obtained and plotted in the e vs log P curve, Figure 3.10, in the Oed. Tab sheet. Based on the void ratio vs. effective stress

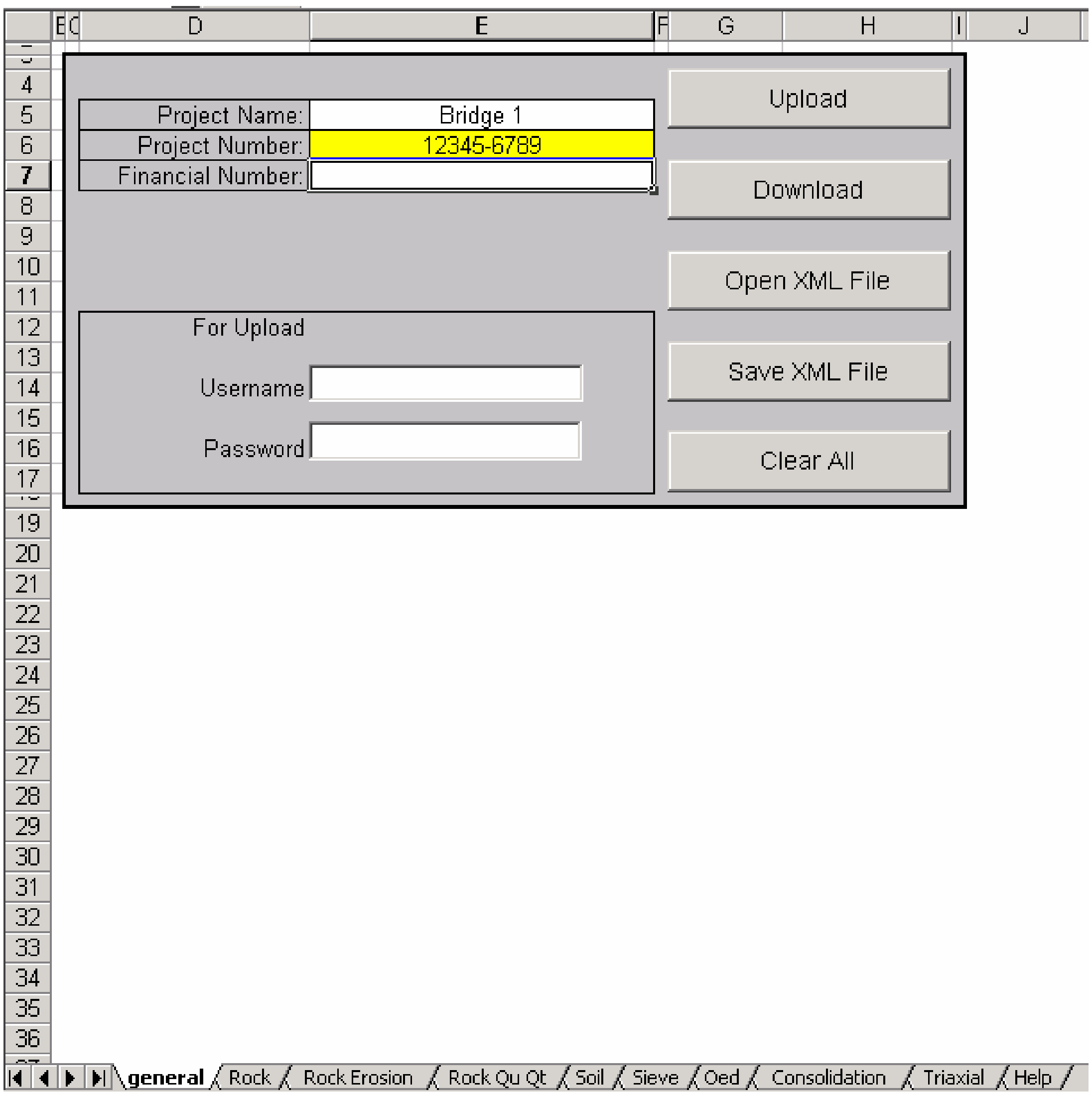


Figure 3.6 Laboratory General Sheet: Opening, Saving, Uploading & Downloading Data

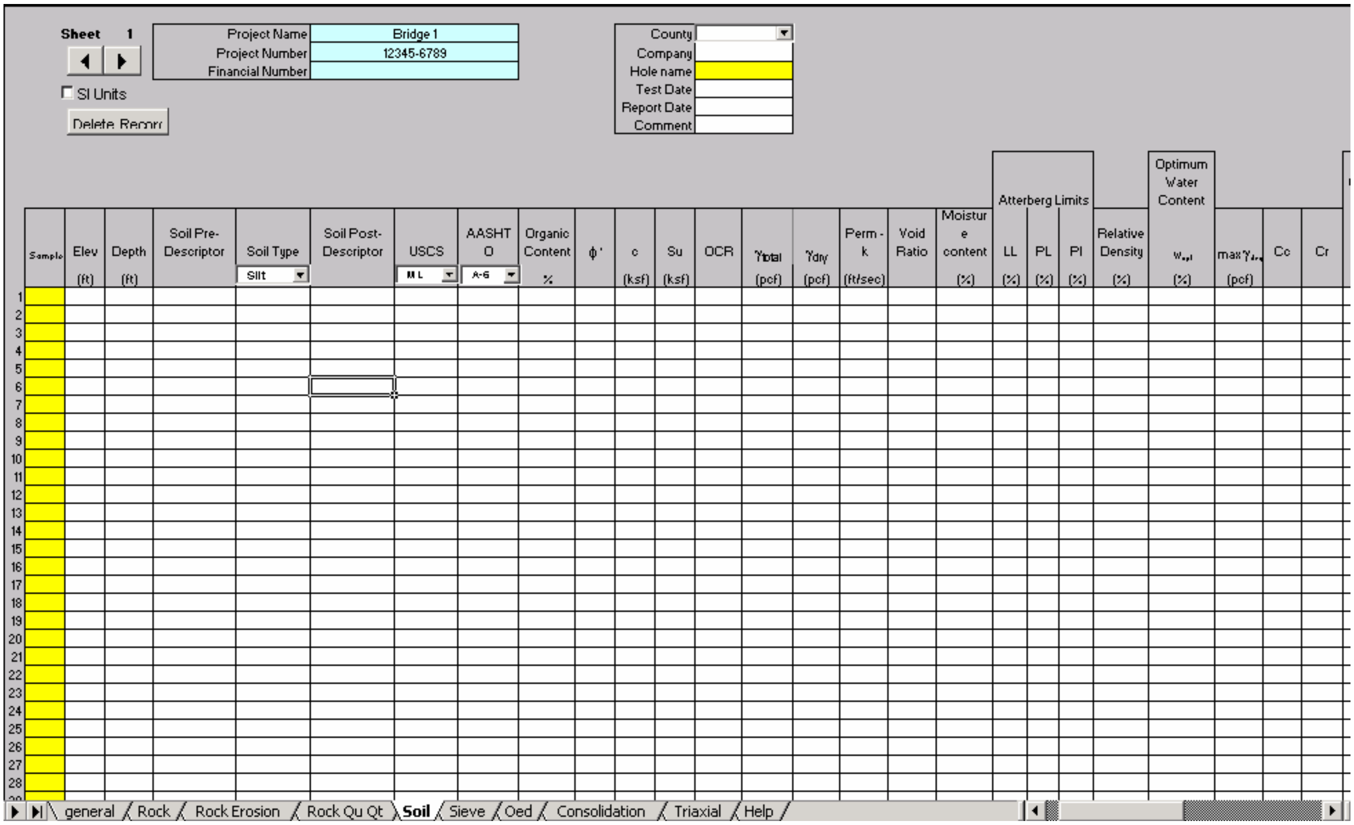


Figure 3.7 Soil Sheet: Classification, Strength, Compressibility and Permeability

plot, the Field Coefficients of Compressibility, Cc, and Swell, Cs are found and recorded in the Soil Tab sheet, Fig. 3.7.

For soil strength, Fig 3.7, the user has multiple options for triaxial testing, i.e. CU, CD, etc. as shown in Figure 3.11. The use of confining pressure (i.e. cell pressure), pore pressure (i.e. backpressure), saturation assessment (i.e. B value), etc. are selectable options. The hole name, sample name, test type, and load cell at failure are required for the triaxial test. These required input cells are identified in yellow in the Excel sheet. The Tab sheet will automatically generated the stress path of the loading in either MIT or

Cambridge p-q space. After recording the stress paths, the user enters the effective strength (i.e. ϕ', c') or total strength parameters (i.e. Su) in the soil sheet, Fig. 3.7.

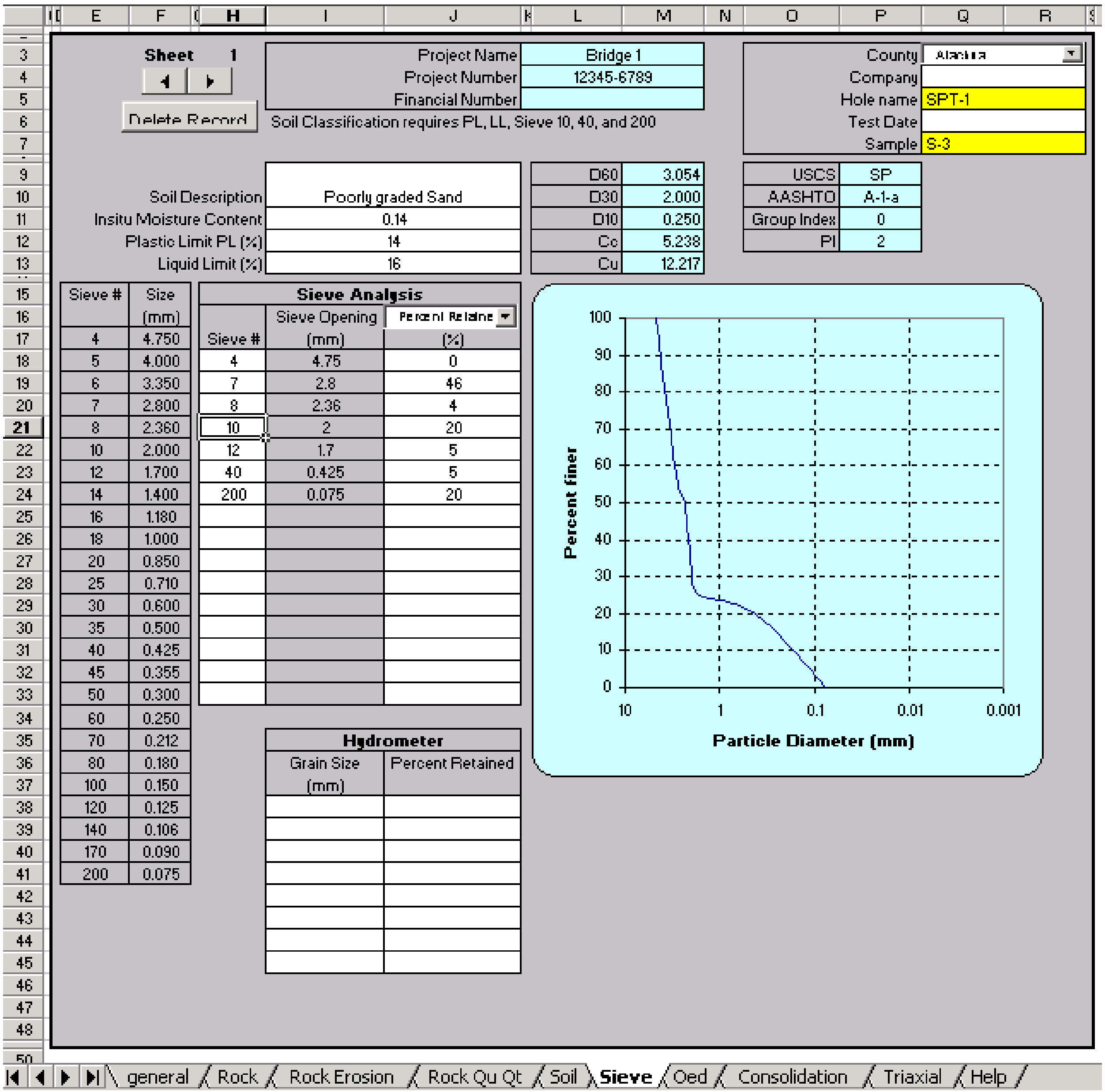


Figure 3.8 Grain Size Sieve Analysis Data

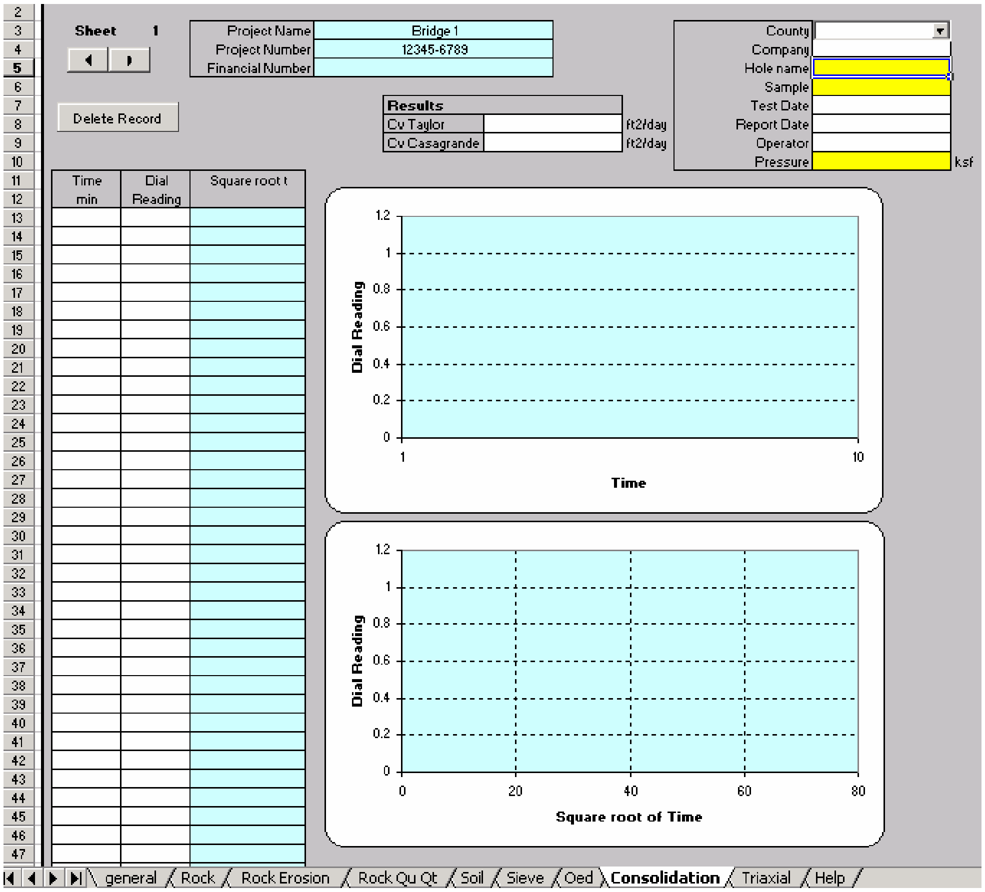


Figure 3.9 1-D Consolidation Deformation vs. Time Data

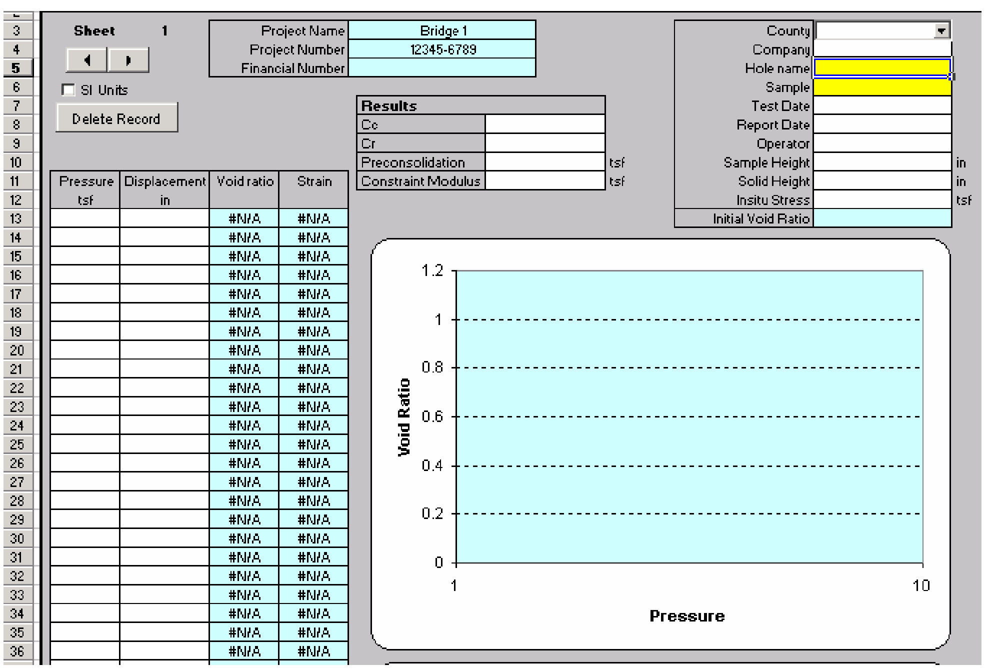


Figure 3.10 1-D Void Ratio vs. Vertical Effective Stress Data

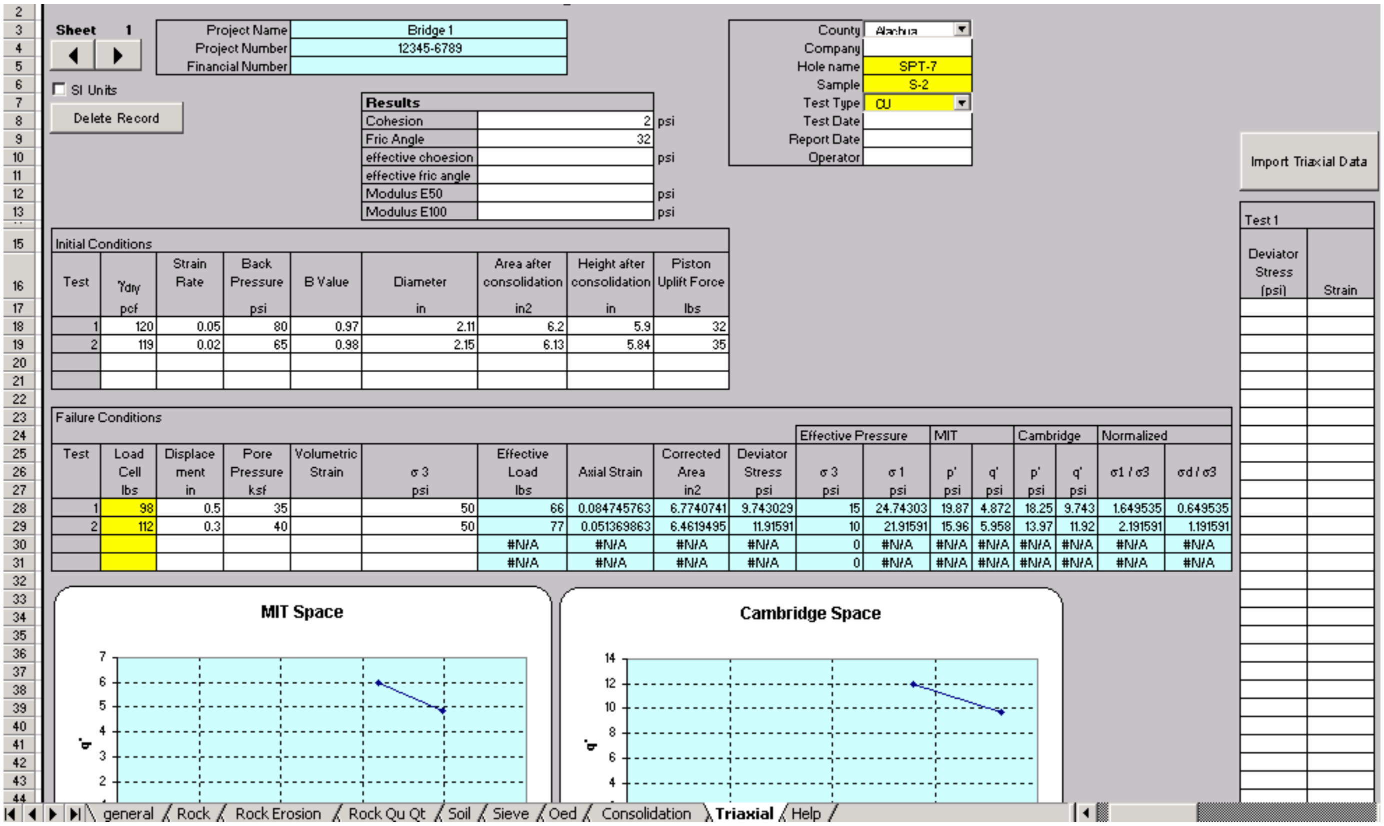
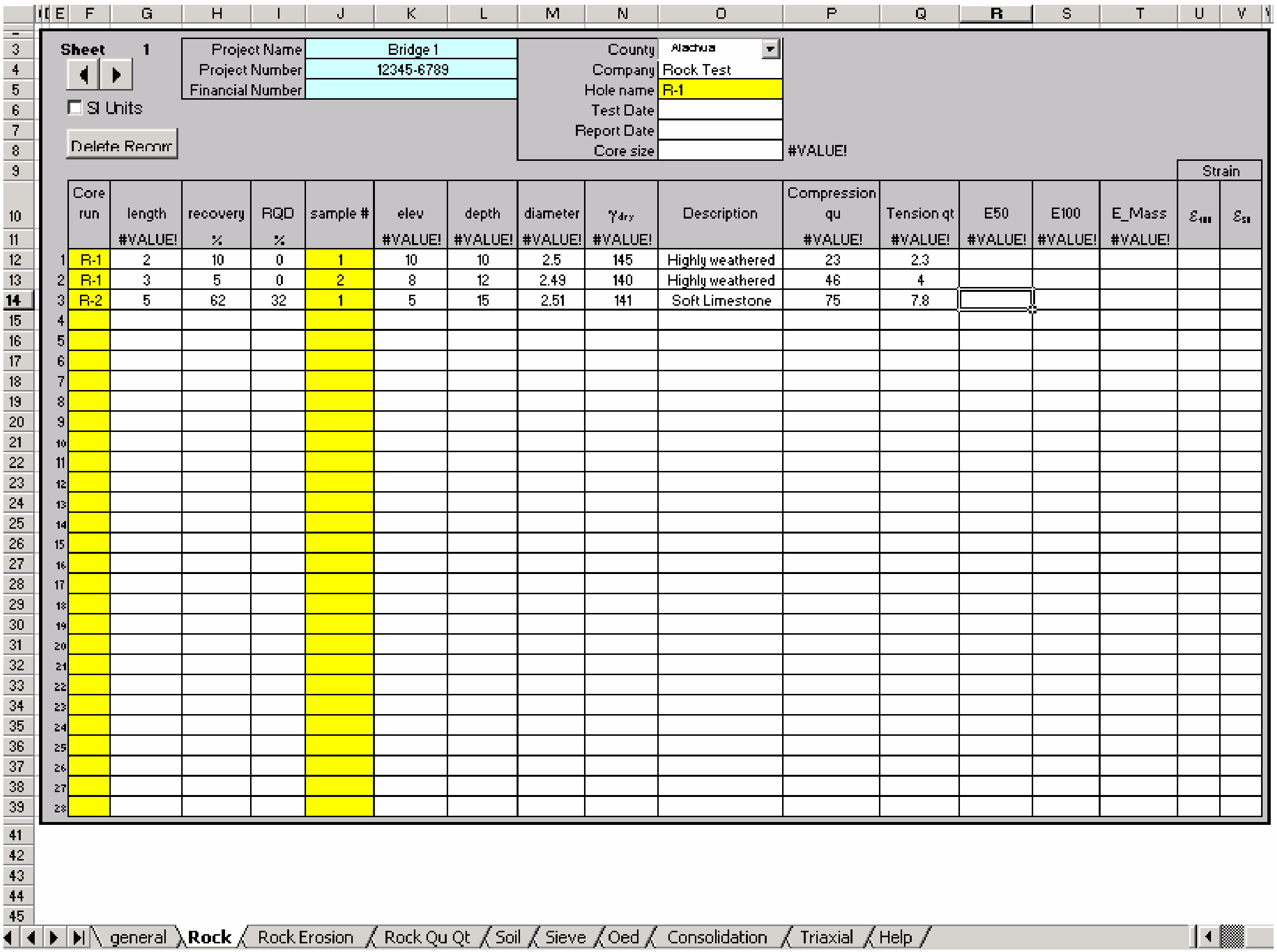


Figure 3.11 Laboratory Triaxial Strength Data/Analysis

The rock tab sheet, Fig 3.12, records standard laboratory rock data by sample: length, diameter, Elevation, Unit Weight, Recovery, RQD, Strength (Qu, Qt), Compressibility (Young's Modulus), and erosion characteristics. Specifics of each test are given in separate Tab sheets. For instance, the rock erosion results are recorded in Figure 3.13, and compressibility and strength in Figure 3.14. The rock strength sheet, Figure 3.14, requires the user to identify type of test, i.e. qu and qt, as well as stress and strain response of the sample. From stress vs. strain, the Young's Modulus of a rock sample is determined. The user (engineer/technician) needs to enter the computed properties in Figure 3.12.



Sheet 1

Project Name: Bridge 1
Project Number: 12345-6789
Financial Number:
SI Units
Delete Record
County: Alachua
Company: Rock Test
Hole name: R-1
Test Date:
Report Date:
Core size: #VALUE!

	Core run	length	recovery	RQD	sample #	elev	depth	diameter	γ_{dry}	Description	Compression qu	Tension qt	E50	E100	E_Mass	Strain ε_{ui}	Strain ε_{si}
		#VALUE!	%	%		#VALUE!	#VALUE!	#VALUE!	#VALUE!		#VALUE!	#VALUE!	#VALUE!	#VALUE!	#VALUE!		
1	R-1	2	10	0	1	10	10	2.5	145	Highly weathered	23	2.3					
2	R-1	3	5	0	2	8	12	2.49	140	Highly weathered	46	4					
3	R-2	5	62	32	1	5	15	2.51	141	Soft Limestone	75	7.8					

general | Rock | Rock Erosion | Rock Qu Qt | Soil | Sieve | Oed | Consolidation | Triaxial | Help

Figure 3.12 Rock Data: Recovery, RQD, Unit Weight, Strength, & Compressibility

3.4 AS BUILT PILE/SHAFT DATA

Information of great use for maintenance, as well as future bridge widening (design/construction) is the as built data from new construction, especially individual piles/shafts data for every bridge pier. This information is currently stored in either paper or pdf format; however the FDOT construction group is implementing a PC based pile/shaft technician software to electronically record as built information. Consequently, it is envisioned that the latter information may be uploaded directly into the FDOT Deep

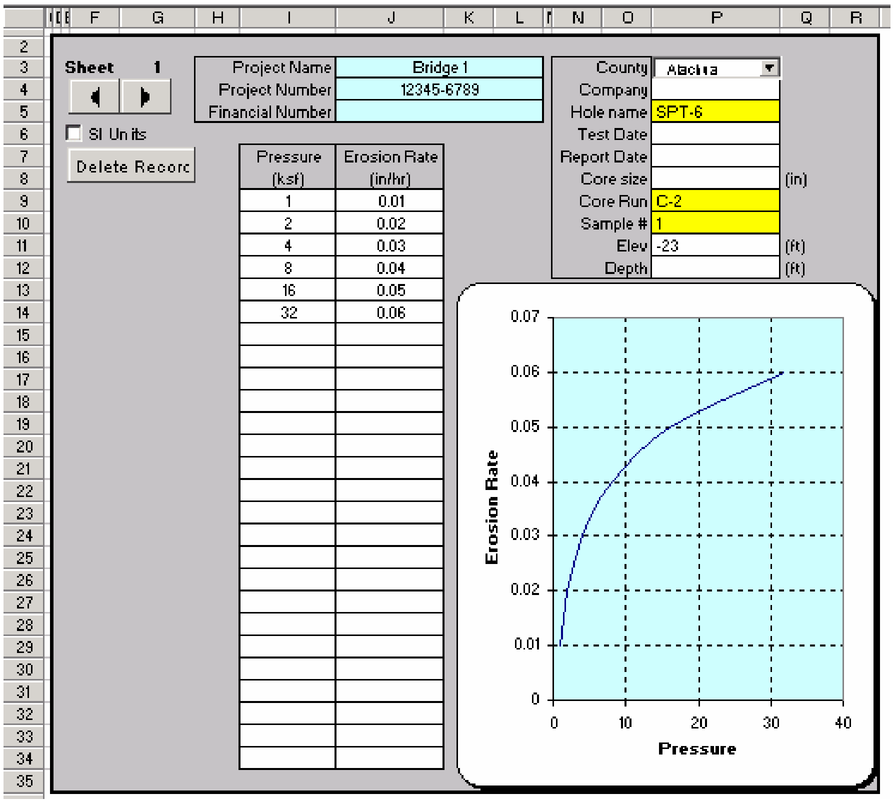


Figure 3.13 Rock Erosion Tab Sheet

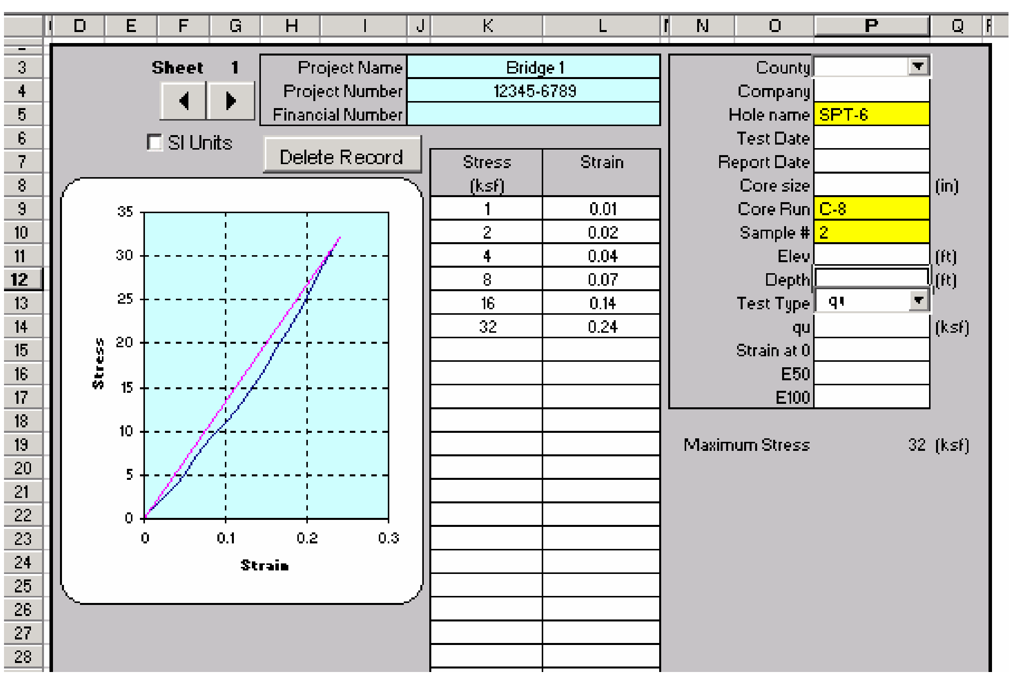


Figure 3.14 Rock Qu and Qt Tab Sheet

Foundation Database as part of the construction process. The following is a description of the recorded information, as given in As Built Excel Spreadsheet.

Shown in Figure 3.15 are list of piles and shafts associated with a project, bridge, and connecting piers, along with their geometry (length, width, etc.) and locations. A

Project name: **Example not real bridge** Project number: **55555-4444**

Pile name	Pier #	Pile type	desc-ription	D	latitude	longitude	station	offset	ground elev.	total length	embed length	pile driven
11	1	Concrete	closed end	36	29.60013	-81.8379	22+26	-9	-1	100	90	97
11	2	H Steel	closed end				22+10			40	35	
12	3	Pipe Steel	closed end	200			22+30	3		50	45	

Project name: **Example not real bridge** P

Shaft name	Pier #	Drilled method	latitude	longitude	
Barrette	None	slurry and/or casing	29.60015	-81.8379	2
Shaft 2	None	dry	29.6002	-81.838	

ex-im / General / list / Piles / Shafts / Help

Figure 3.15 As Built Piles/Shafts, Pier No., Station, Lengths and Widths

graphical view of all piles and shafts are presented in the "General" Tab sheet of Figure 3.16. Also presented in that sheet are the names of the contractor, design consultant, bridge location, as well as Load Test Company if field-testing is performed.

Presented in Figure 3.17 is the tab sheet identifying the pile geometry, materials, driving system, and driving-record.   Data collected is the information specified in "Florida Department of Transportation's Standard Specifications for Road and Bridge

Construction" handbook. Of interest are final pile lengths (based on construction), pile splices if they occurred, and pile driving record for future bridge maintenance, design, etc.

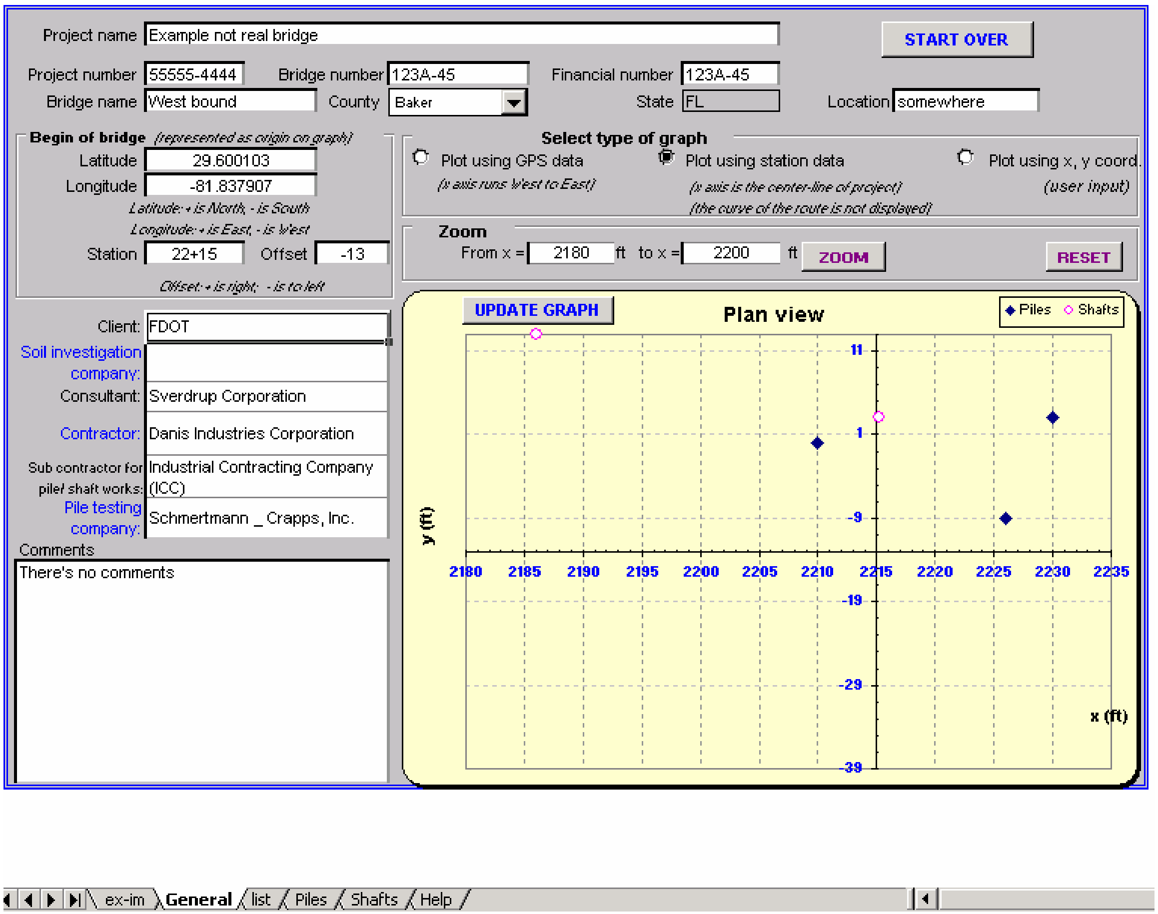


Figure 3.16 As Built Location of Piles/Shafts

In the case of drilled shaft construction, the "Shafts" tab sheet, Figure 3.18, identifies the construction process (wet hole, case, etc.), geometry of shaft, steel areas, concrete strengths and volumes, as well as rock quality for borings below the tip if founded in limestone.

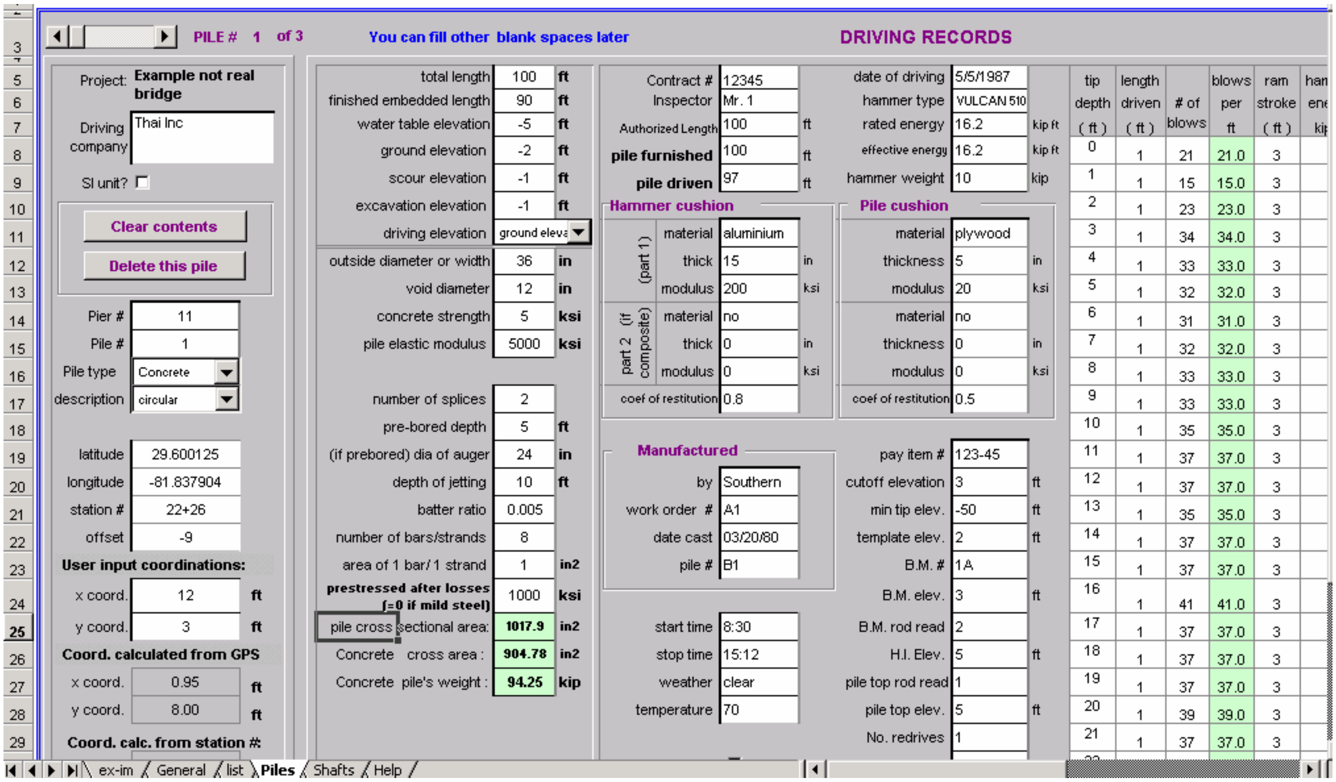


Figure 3.17 Pile Geometry, Materials, Driving System, and Driving Record

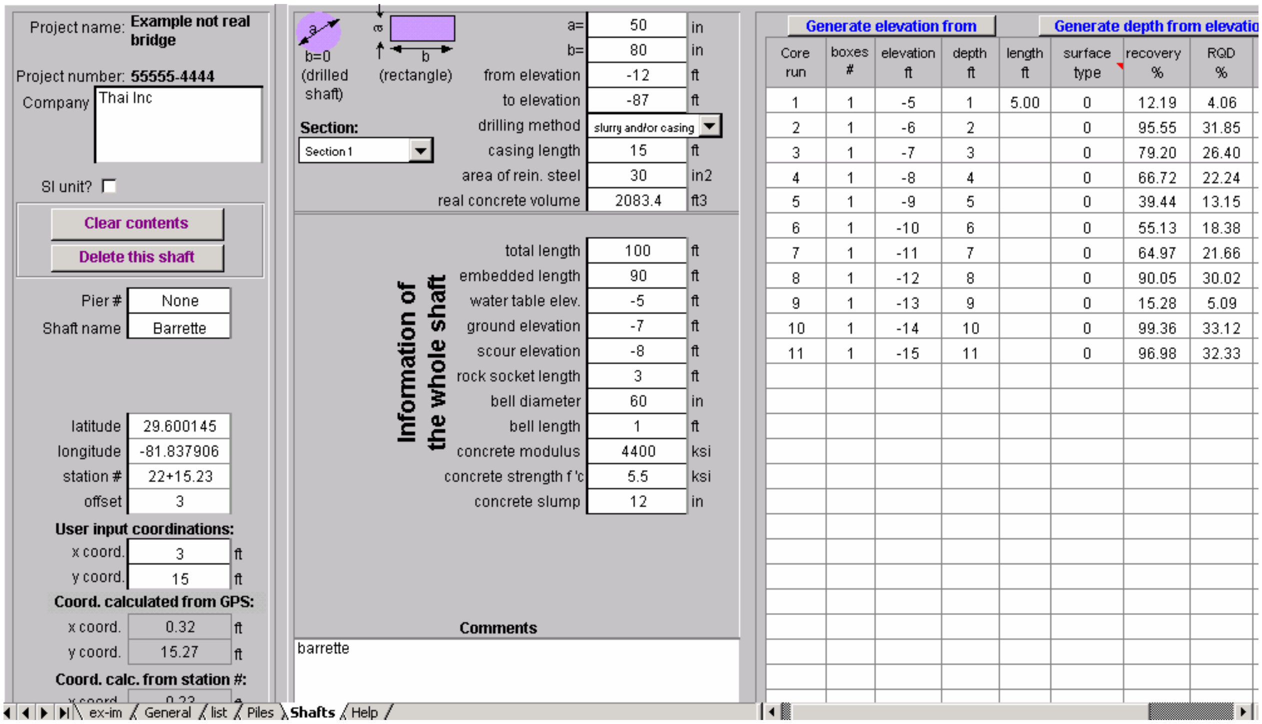


Figure 3.18 As Built Drilled Shaft Geometry, Construction Process & Rock Quality

The user has the option of loading existing design pile/shaft information from a file or the database into the spreadsheet modifying it for as built conditions, with the “ex_imp” tab sheet, Figure 3.19. In addition, since the pile/shaft technician may not have access to the Internet on the jobsite, he/she has the option of storing the as built information in an XML file, Figure 3.19 for later uploading into the database.

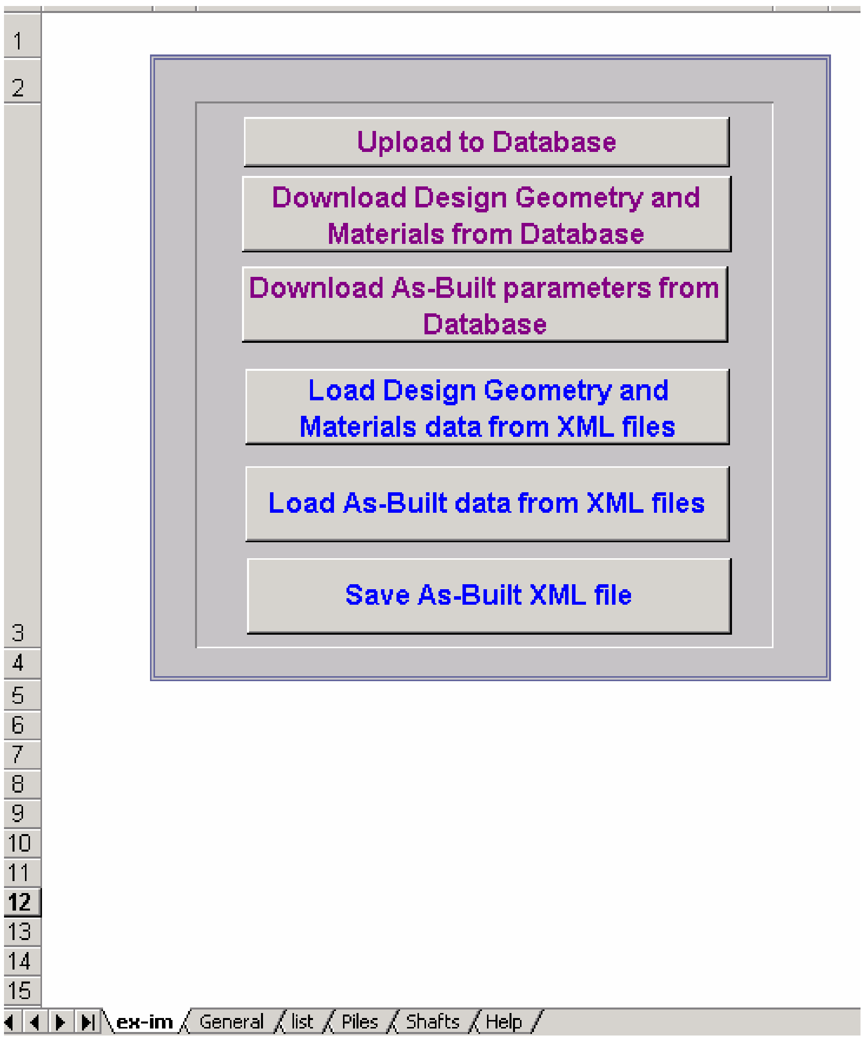


Figure 3.19 Uploading and Downloading As Built and Design Information

3.5 DESIGN PILE/SHAFT DATA

Shown in Figure 3.20 is the soil and pile property information for axial pile analysis from the Design Excel spreadsheet. The user inputs the boring number, pile tip and top elevations and FB-Deep soil descriptions (i.e. soil #s 1,2, etc.) based on SPT soil descriptions. Since each pile in the design may have a specific SPT boring associated with it, the user has an option of downloading different borings from the database, (download SPT button), or a XML file, Fig. 3.20. The user/engineer is required to input “Soil Type” (i.e. FB-Deep descriptors), pile geometry (i.e. cross, length, etc.), as well as elevations, for each pile (top right, Fig. 3.20).

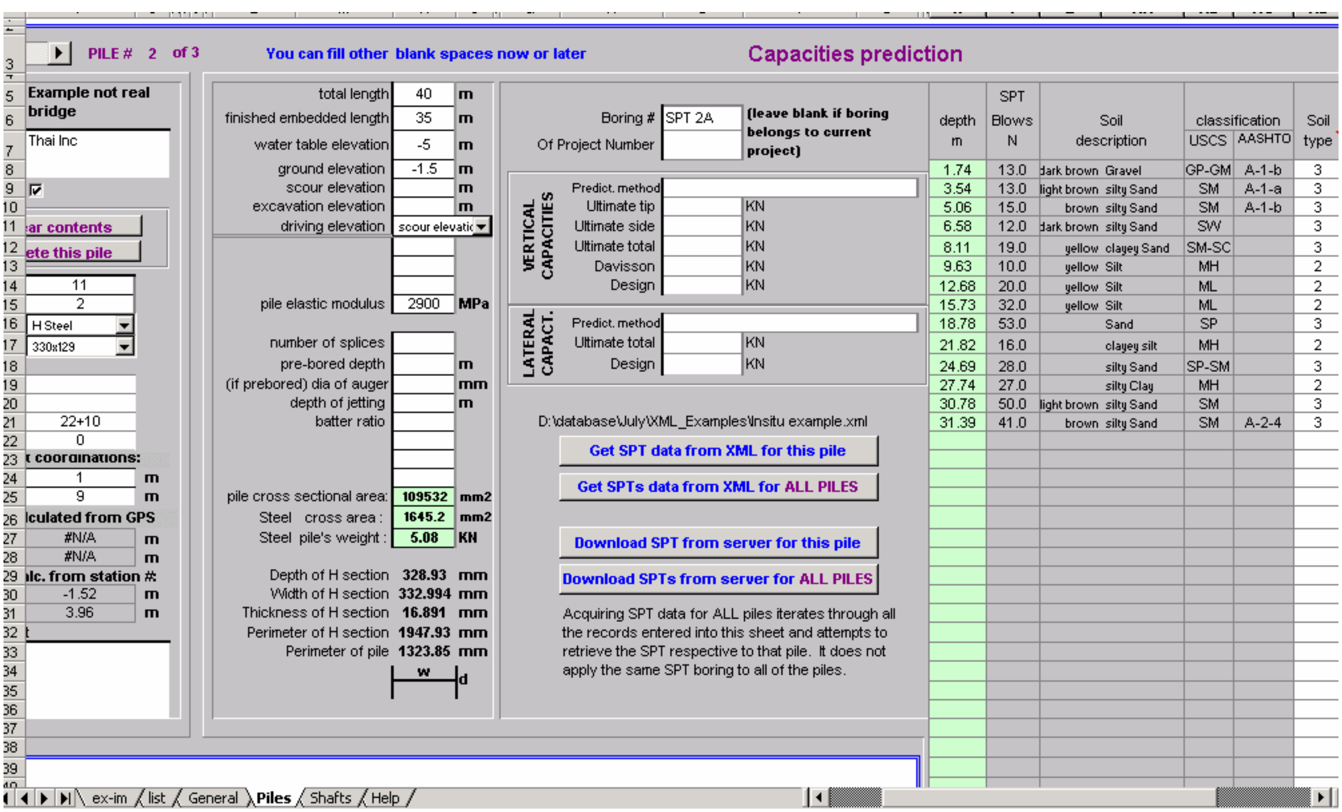
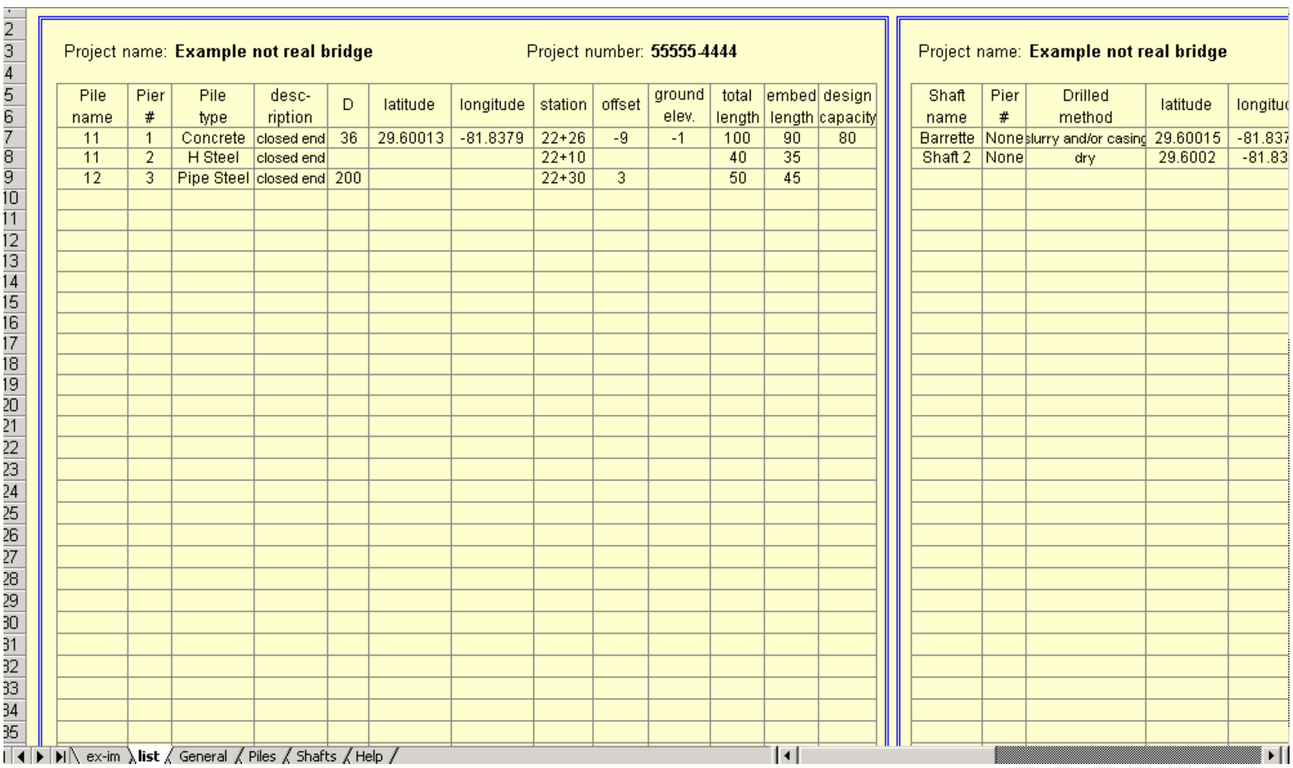


Figure 3.20 Design Pile Information: Location, Boring Information, FB-Deep Data

Presented in Figure 3.21 are the complete list of piles and shafts on a project. The latter may be created within the spreadsheets, e.g. Fig.3.20, read from a file, or downloaded from database (i.e. future viewing). The “List” tab identifies all the geometries of the piles, and shafts, as well as their locations (i.e. station numbers), and elevations.



Project name: **Example not real bridge** Project number: **55555-4444**

Pile name	Pier #	Pile type	desc-ription	D	latitude	longitude	station	offset	ground elev.	total length	embed length	design capacity
11	1	Concrete	closed end	36	29.60013	-81.8379	22+26	-9	-1	100	90	80
11	2	H Steel	closed end				22+10			40	35	
12	3	Pipe Steel	closed end	200			22+30	3		50	45	

Project name: **Example not real bridge**

Shaft name	Pier #	Drilled method	latitude	longituc
Barrette	None	slurry and/or casing	29.60015	-81.837
Shaft 2	None	dry	29.6002	-81.83

ex-im list General Piles Shafts Help

Figure 3.21 List of Pile and Shaft Geometries and Locations for a Project

Shown in Figure 3.22 is a plan view of all the piles and shafts on a specific project. The pile and shaft locations may be displayed by station number, GPS coordinates, or with x-y coordinates. Generally, for design station numbers are employed, however for future maintenance, bridge widening, etc, it is envisioned that GPS coordinates would be more useful. Also the construction contractor and Pile Testing Company shown in Fig. 3.22 are only viewable after construction and when downloaded from the database.

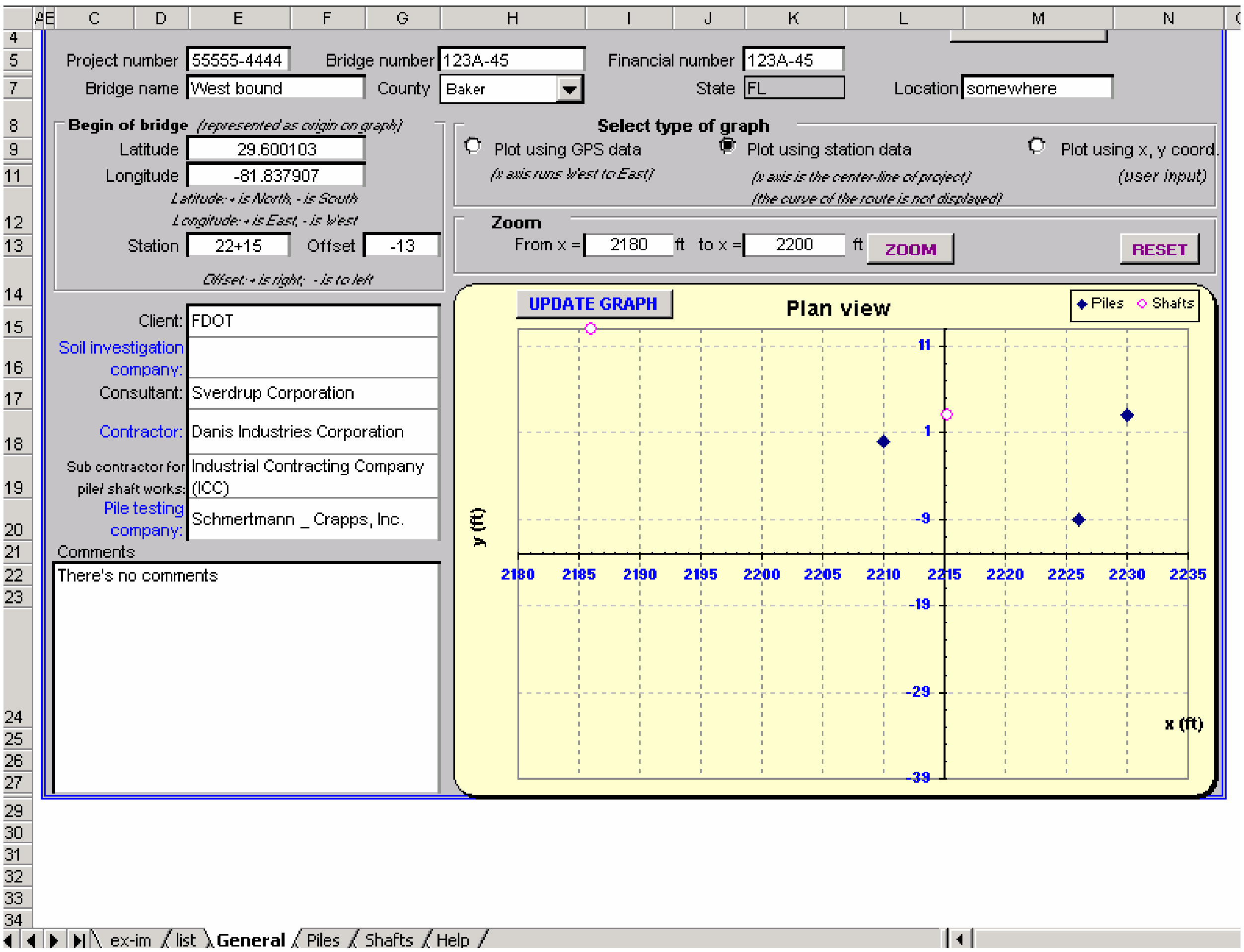


Figure 3.22 Plan View of Pile and Shaft Locations for a Specific Project

Presented in Figure 3.23 is the “Shafts” design Tab. The sheet includes individual geometries for each shaft on the project (i.e. diameter, steel areas, shaft lengths, and elevations), as well as soil and rock properties from the nearest boring. The latter information is saved to a file or the database. FB-Deep which is capable of reading the data from a file or database, performs the capacity assessment, i.e. skin and tip resistance, and writes to the database or XML file. If available, the spreadsheet will display the results on the right of “shaft” tab sheet, or the user may enter his/her own capacities (i.e. non-FB-Deep Analysis).

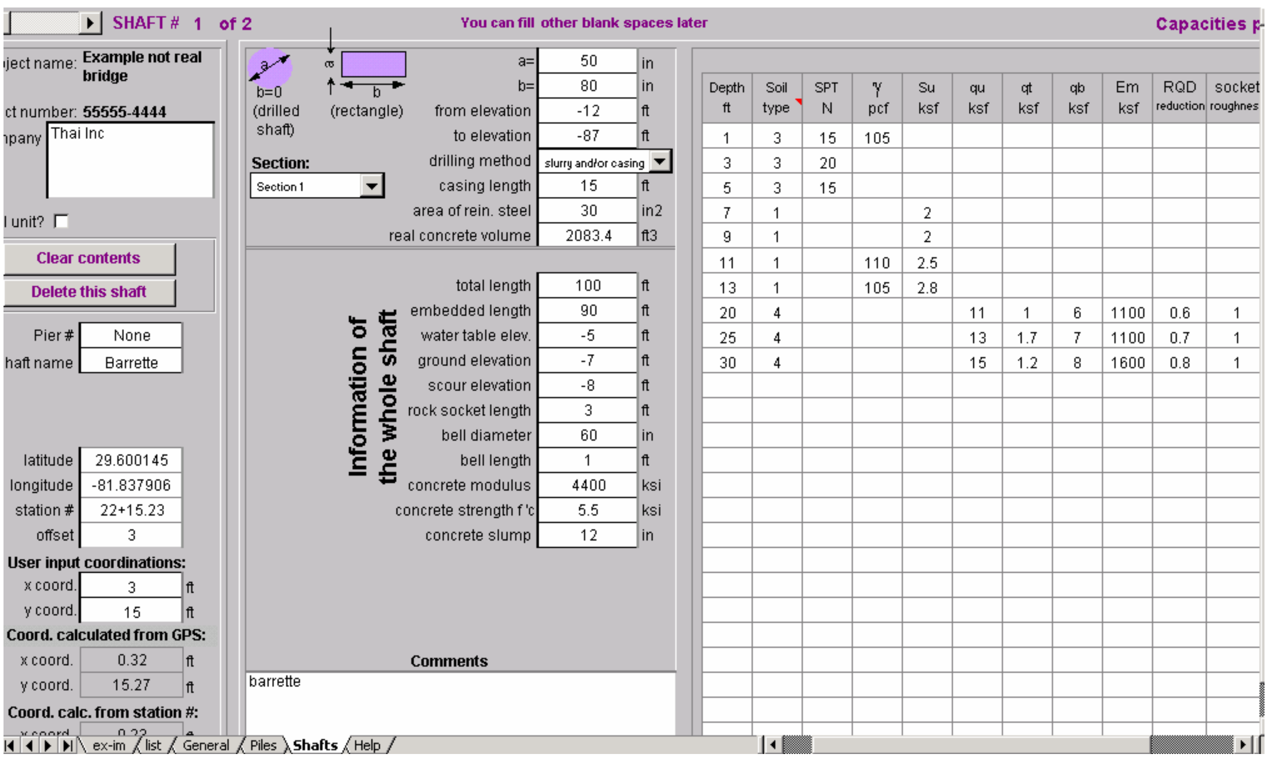


Figure 3.23 Design Shaft Information: Location, Boring Information, FB-Deep Data

After entering all the geometries, properties, and capacities of the piles and drilled shafts on the project, the user has the option of either saving the data to an XML file or uploading it to the database, Figure 3.24. Saving to a file would be typical when all the pile/shaft design information hasn't been collected. Downloading would be used to obtain either Insitu (e.g. SPT), Laboratory (rock strengths) Data, or completed past project design data.

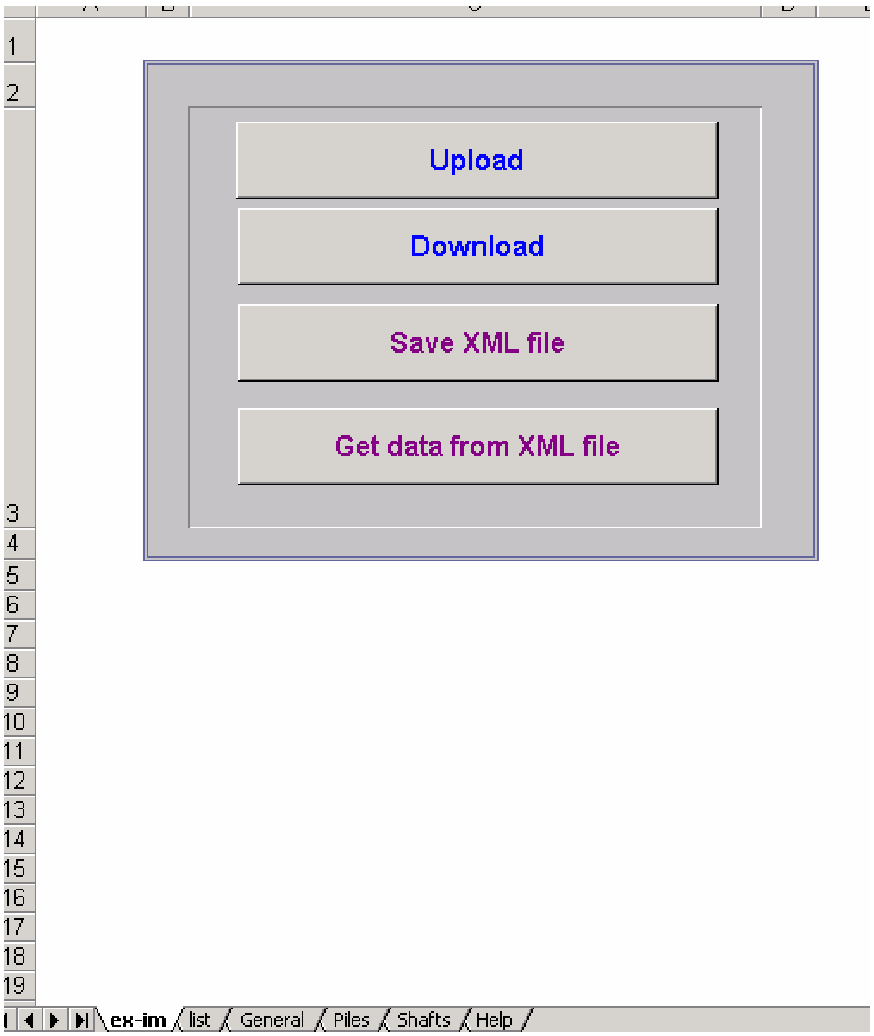
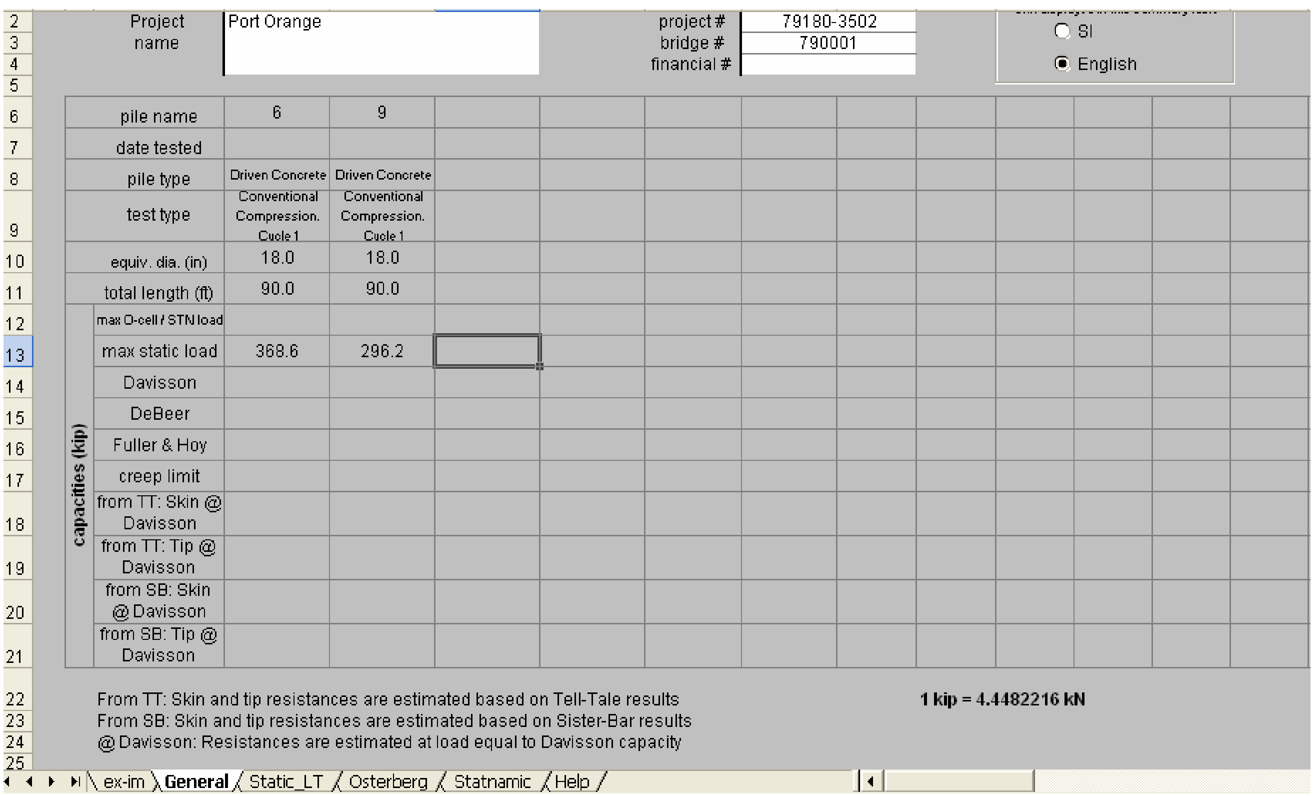


Figure 3.24 Saving or Reading Pile/Shaft Design Data

3.6 LOAD TESTING DATA/ANALYSIS

In the case of field load-testing: Conventional Top Down, Osterberg, or Statnamic, the Excel Spreadsheet “Load_Test” was developed to record all raw data (i.e. loads, settlements, telltales, strains, etc.), as well as plotting load vs. settlement, skin friction, and estimated pile/shaft capacities. Shown in Figure 3.25 is the “General” Tab which provides a list of piles/shafts, tested, length, width, maximum load applied, Static Capacity (Davisson, Debeer, etc.), and distribution of skin and tip resistance for a project.


Project name: Port Orange
project #: 79180-3502
bridge #: 790001
financial #:
SI
English

pile name	6	9
date tested		
pile type	Driven Concrete	Driven Concrete
test type	Conventional Compression. Cycle 1	Conventional Compression. Cycle 1
equiv. dia. (in)	18.0	18.0
total length (ft)	90.0	90.0
capacities (kip): max O-cell / STN load		
max static load	368.6	296.2
Davisson		
DeBeer		
Fuller & Hoy		
creep limit		
from TT: Skin @ Davisson		
from TT: Tip @ Davisson		
from SB: Skin @ Davisson		
from SB: Tip @ Davisson		

From TT: Skin and tip resistances are estimated based on Tell-Tale results
From SB: Skin and tip resistances are estimated based on Sister-Bar results
@ Davisson: Resistances are estimated at load equal to Davisson capacity
1 kip = 4.4482216 kN

Figure 3.25 Pile/Shaft Geometries, Maximum Loads and Capacities

Shown in Figure 3.26 is the raw or recorded data for conventional top down static load tests performed on a project. The right portion of the sheet identifies the load test number, location, type of pile/shaft, location of telltale or strain gages within pile/shaft, and calibration of load cell. The middle of the sheet, Fig. 3.26 presents the load applied to the top of the pile/shaft and its associated displacement. The right side provides the telltale movement or strains at various elevations for each applied top load.

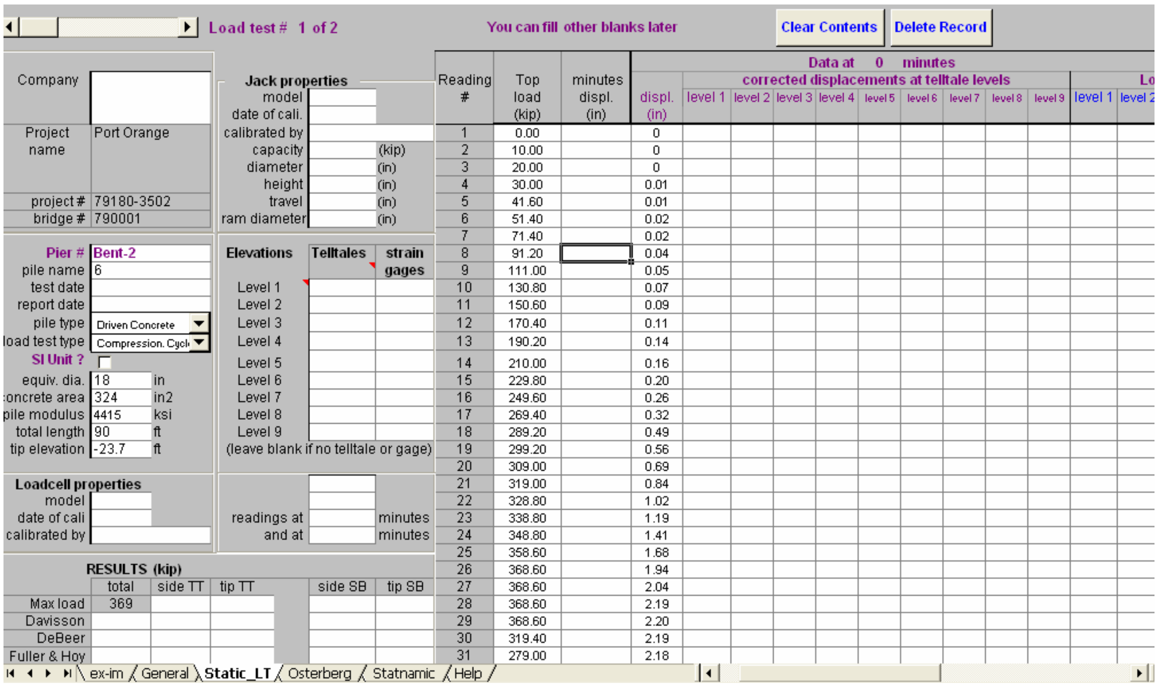


Figure 3.26 Top Down Conventional Static Load Test Data

Presented in the right of “Static_LT” sheet is the plot of load vs. displacement of the top of the pile/shaft, Figure 3.27, as well as the location of telltales/strain gages, and the distribution of the load along the length of the pile/shaft for individual top loads. For capacity estimation, Davisson is shown in the top right of Fig. 3.27 and Debeer is estimated from the log Load vs. log Displacement plot shown in the lower right.

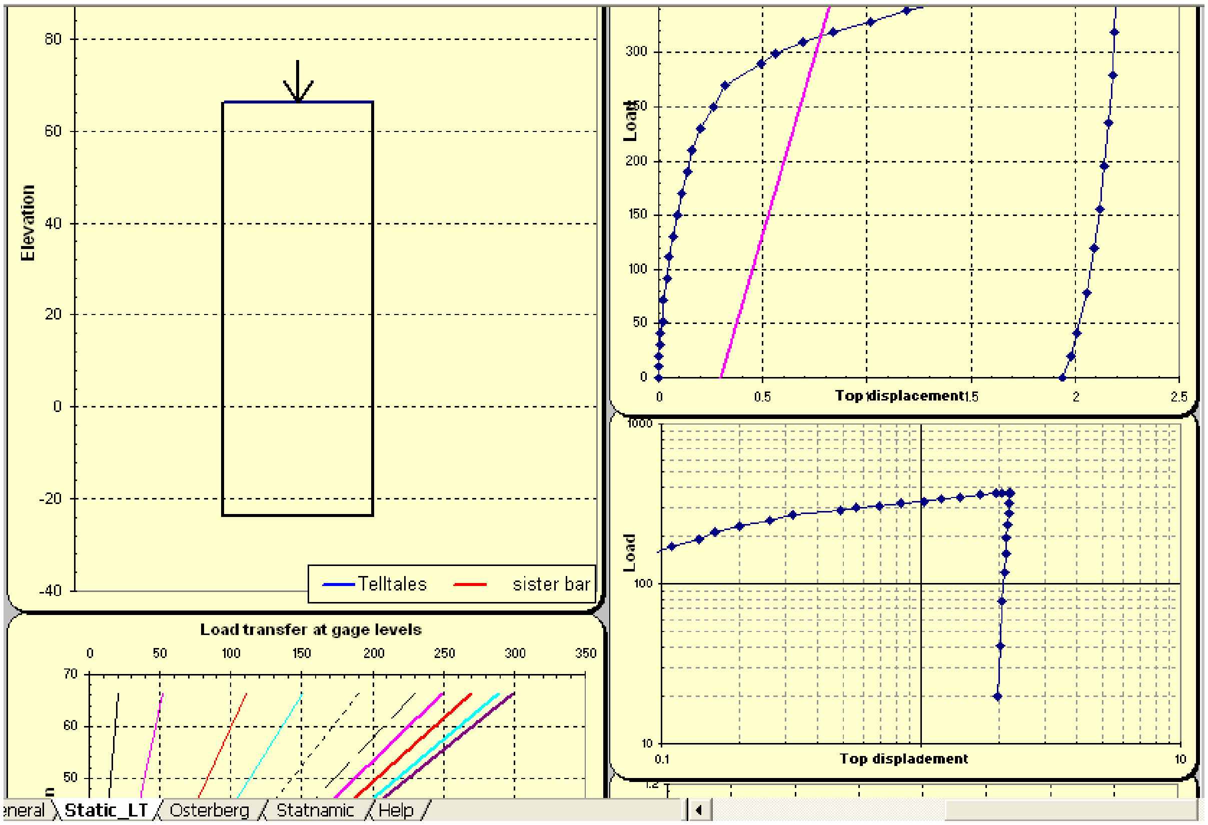


Figure 3.27 Static Gage Locations, Load vs. Displacement, and Load vs. Depth

For Osterberg Testing, the "Osterberg" Tab, Figure 3.28, was developed to collect and reduce the data. In the right side of the sheet, the pile/shaft geometry, Osterberg cell dimensions (height, and travel), and calibration are recorded. Also identified are instrumentation (telltales, and strain gages), as well as their location within the pile/shaft. In the middle and right side of Figure 3.28 are the recorded telltale movements and strains for individual applied cell loads. For "Osterberg" Tab, the far right of the sheet, Figure 3.29, displays the upward and downward movement of the pile/shaft with cell load, as well as the distribution of load within the pile/shaft as a function of applied cell load.

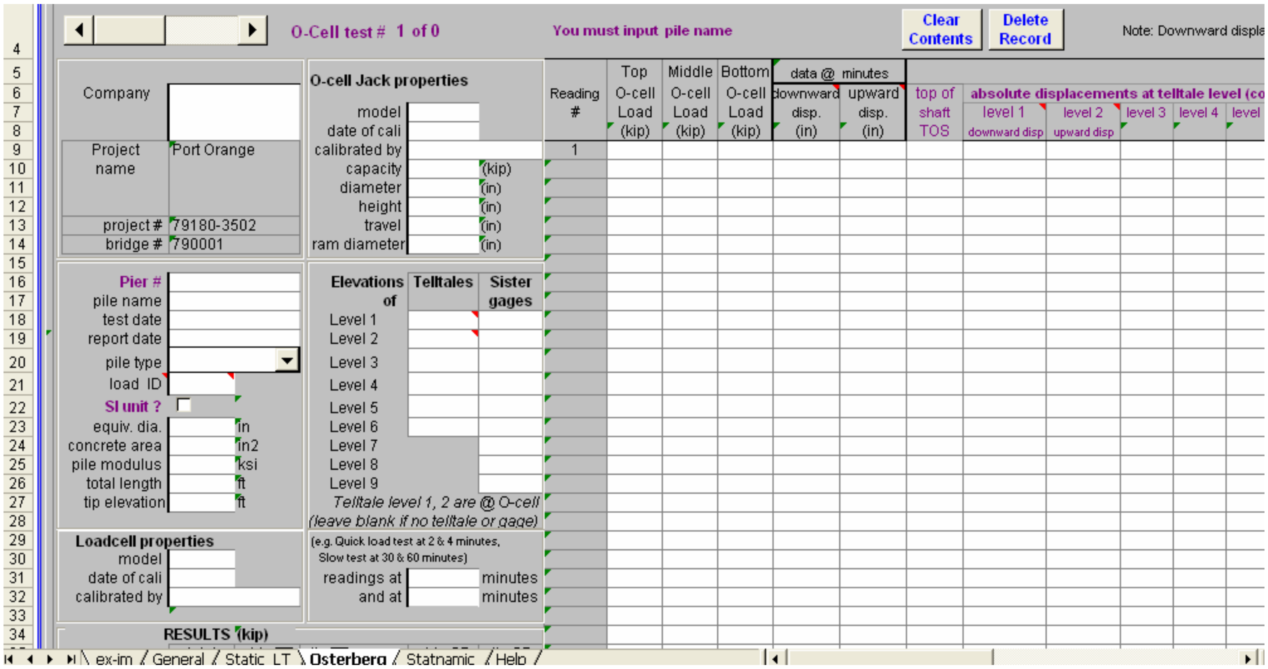


Figure 3.28 Osterberg Testing, Applied Loads, Telltale Movements, and Strains in Shafts

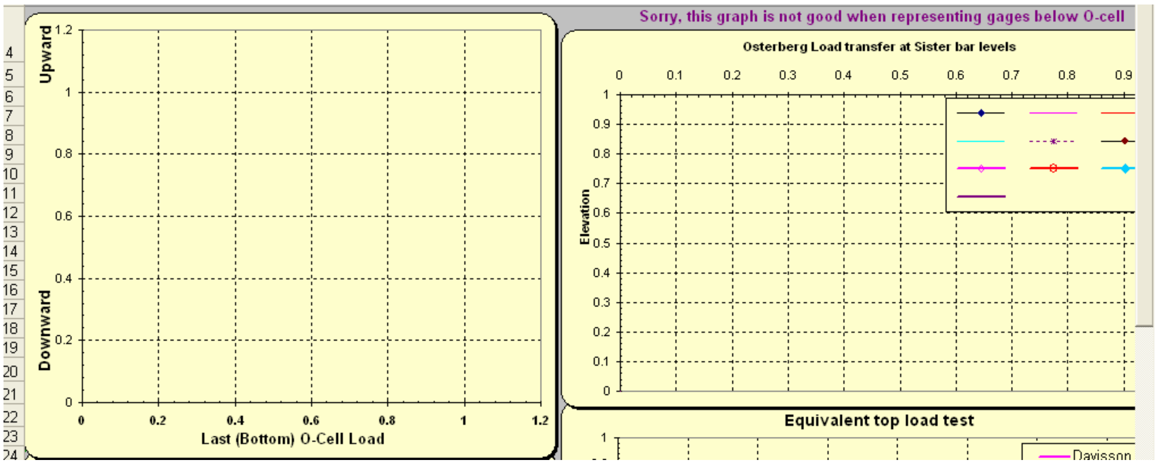


Figure 3.29 Plots of Pile/Shaft Movements, and Loads within Pile/Shaft with Cell Loads

For Statnamic field testing, the “Statnamic” Tab sheet, Figure 3.30 was written to collect and reduce the data. The right side of the sheet identifies the test number, Pile/Shaft location, strain gage location, and reaction mass. The middle of the sheet records the top acceleration, velocity, and strains within pile/shaft for applied dynamic load (i.e. Statnamic load) as a function of time.

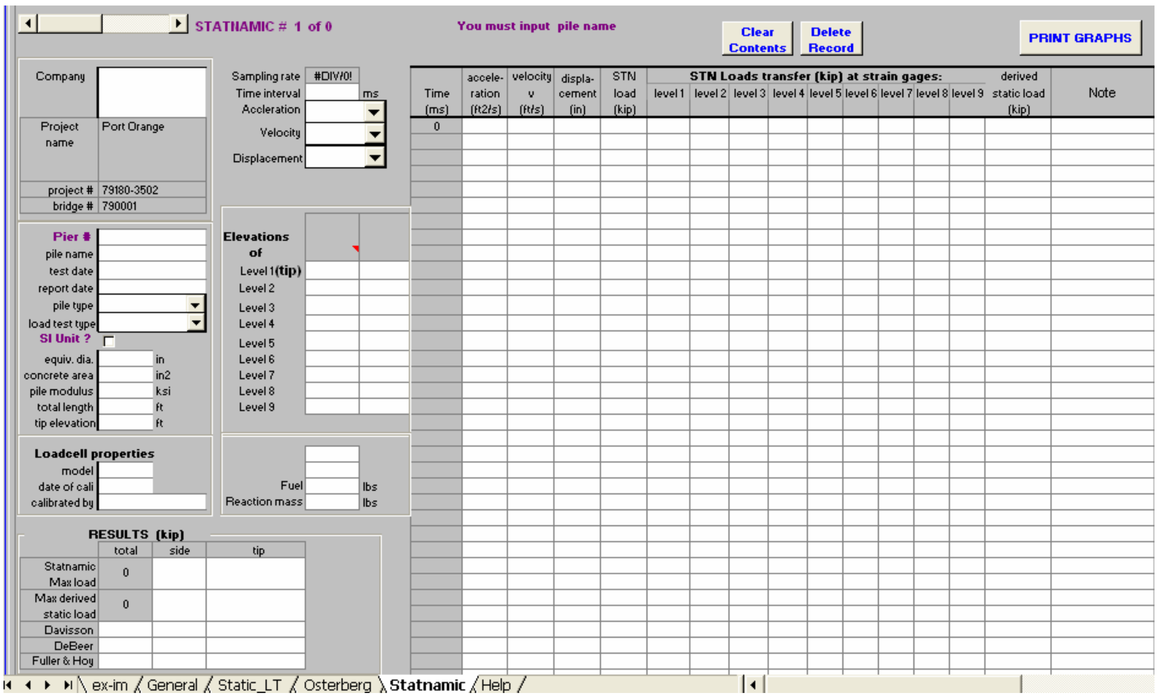


Figure 3.30 Statnamic Load, Velocity, Telltale Movements, and Strains in Shafts

Shown in the far right of the “Statnamic” sheet, Figure 3.31 are the instrumentation locations, and plots of dynamic load vs. top displacement as well, as back computed static load (Statnamic-damping-inertia) vs. top displacement of pile/shaft. The static load is estimated from the “Unloading Point Method” if instrumentation is placed only at top (i.e. load cell and accelerometer), and from the “Segmental Unloading Point Method”, if strain gages are located along the length of the pile/shaft. From the static load vs. displacement plot, the pile/shaft capacity is determined and displayed in the right of the sheet.

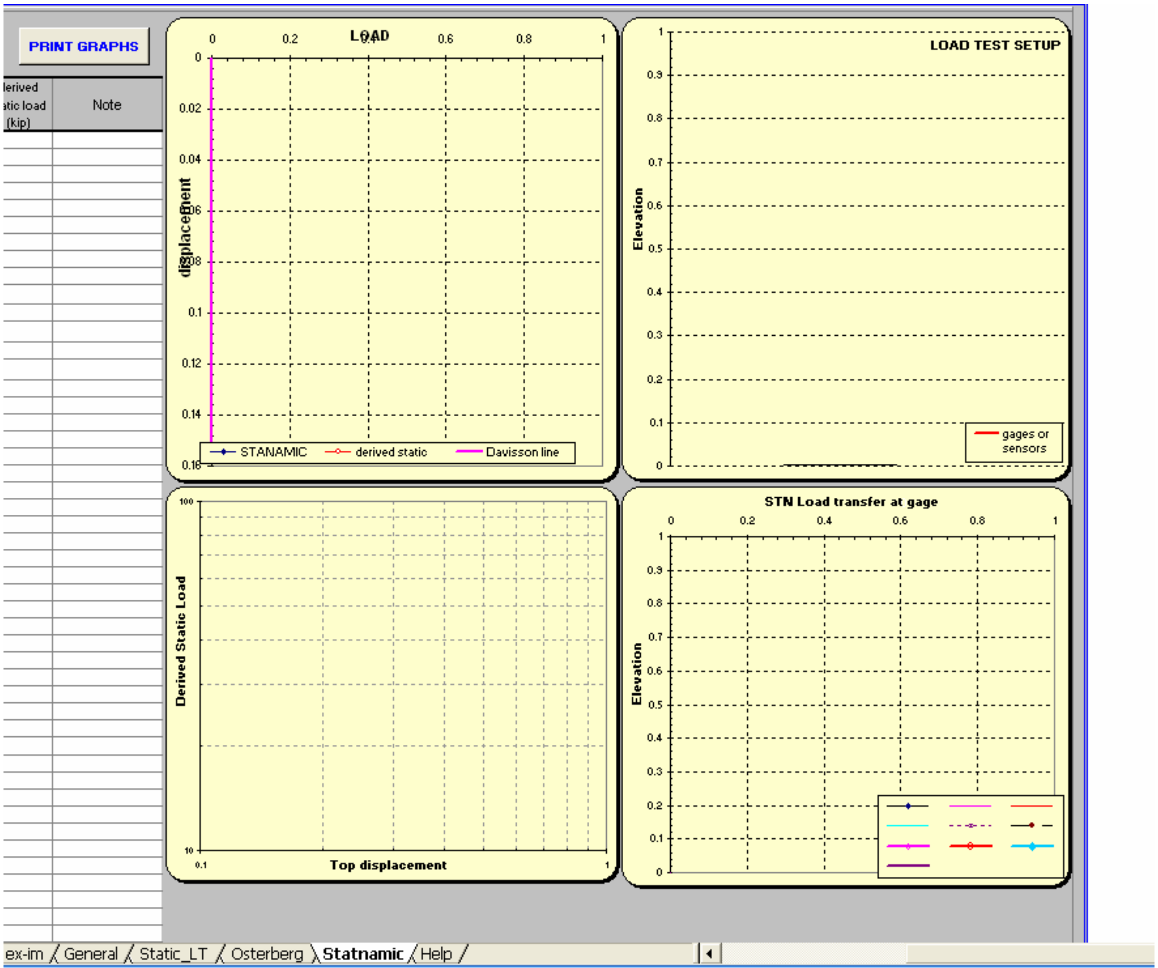


Figure 3.31 Gage Locations, Static, & Dynamic load vs. Top Displacement of Pile/Shaft

The raw data, i.e. loads, displacements, strains, etc., as well as the back computed capacities, skin friction and end bearings determined from Static Load Testing, Osterberg, and Statnamic Testing may be uploaded to the database from the “ex_im” tab sheet shown in Figure 3.32. Note, the spreadsheet may also be used to download existing field load testing data from a project with the “Download” button in Fig. 3.32. Also, since data recovery may occur in the field where Internet access is unavailable, the user may save the data to a file and later load it, Fig 3.31, for uploading to the database.

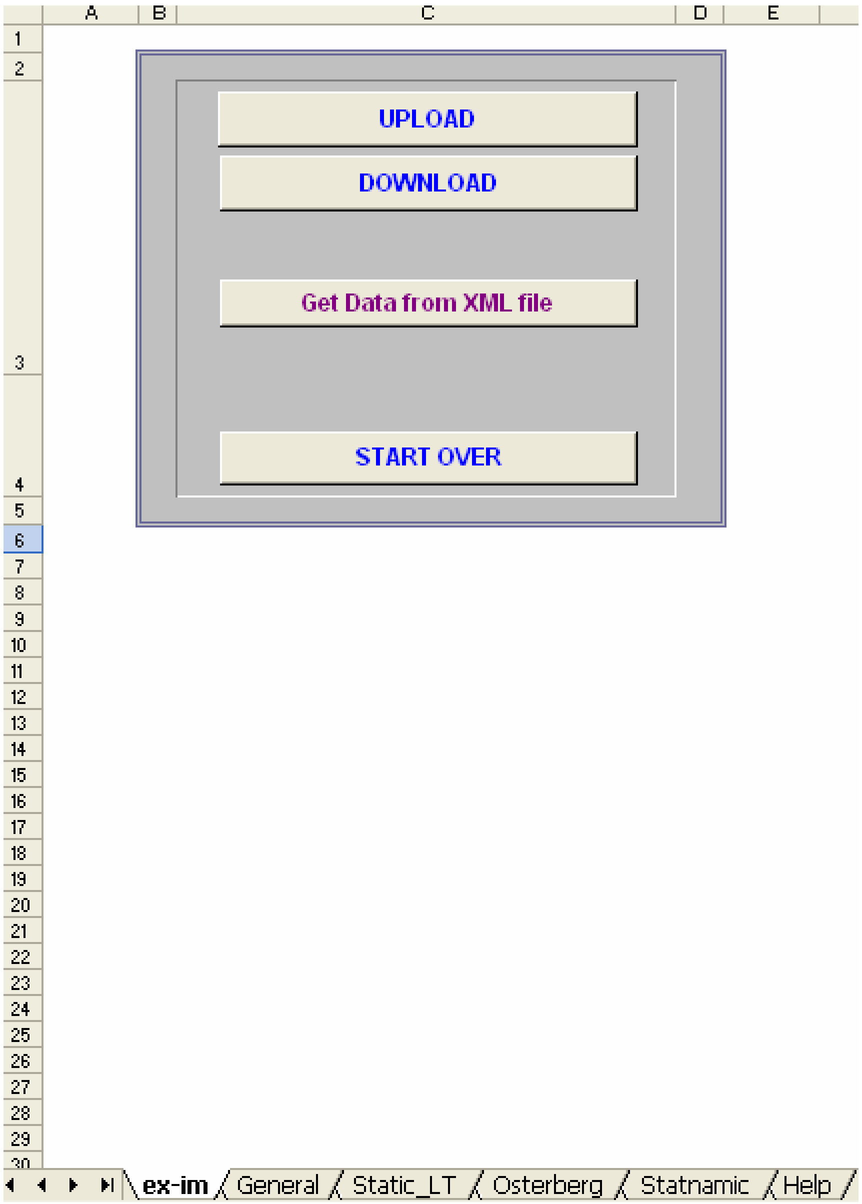


Figure 3.32 Upload and Download Field Load Testing Data

CHAPTER 4
GEOTECHNICAL DATABASE WITH XML TRANSFER

4.1 OVERVIEW

For this project a web based database which stores and provides access to Geotechnical data for deep foundation information was developed. As part of the research, data transfer capability was developed which allows information to be deposited and retrieved (uploaded and downloaded) from the database through XML transfer files. The design allows for the ability to store the various data related to DOT bridge foundations inside a database for easy querying and retrieval at a later time. The system also allows for multiple software groups to easily interact with the database by converting data to the standard XML schema in hierarchical format and submitting it with a DLL through the Internet. Inherent in the design is a security system that allows only the users with the proper credentials to post data and the ability to lock individual data after finalization. The latter involves a three-stage approval process which allows the entered data to be reviewed and approved and as well as locked against change, i.e. creating permanent records.

4.2 DATABASE STRUCTURE

An internal database format was developed to handle storing and retrieving the data through the XML transfer. While the format is similar to the XML schema, it is not the same. The underlying database structure is independent of the transfer format. The software takes care of uploading the XML transfer file and converts it into the required information used by the database. The database format corresponds to the XML schema's hierarchy but the XML tags are independently named. The XML schema is dynamically created based on the tag definitions stored inside the database. The database also has the ability to support multiple XML formats with the names of the tags being different. Currently, only the UF/FDOT schema is implemented. The structure of both the database and the XML schema is contained in the xml_list.doc file. When a change is made to the schema, the xml_list.doc is updated. The file can be found at: http://bsi-web.ce.ufl.edu/db/.

In order to know what table data in the underlying database should be posted to (updated), there are identifying fields (similar to Primary Keys) for most XML tags. For instance, every submission must contain <Project> and this tag must contain the identifying "Project_Number" that is unique to only their Project. Without including "Project_Number", the submission will be rejected. The identifying fields are represented in xml_list.doc by ***(id)*** to left of the attribute. Element groups that are not intended to be updated but always overwritten (ex: SPT_Data group) do not contain an identifier. The original element records are deleted when the new records are submitted each time a submission is made. These types of tags have their submissions locked and approved at the parent level (ex: SPT data is entered and then locked against change). When an item is locked all of the records that fall under its hierarchy are also locked and unable to be updated.

4.3 VIEW DATA

There are multiple ways to view the data stored in the database. The two currently implemented are through a DLL or a web browser. The universal DLL written for this project can be called from most applications in the MS windows environment. The DLL source code is also available for those wishing to build their own DLLs. The DLL has two functions: upload and download. To view the database information, a user may use the download feature. After activating the DLL, a login prompt appears for secure access. Only fields and records that the particular user account has been given access to view will be displayed. The initial tree is called a "skeleton tree" because only the information that identifies the records is displayed, not all of the attributes (data). This is to prevent transferring more data than necessary over the internet and to expedite the retrieval of desired information. When the user selects the node (part of the data) of interest and pushes the 'Get Attributes' button, a tree with all of the parent nodes' attributes and the selected nodes' children is displayed (all the requested data is returned). The nodes that are neither parent nor child of the selected node are discarded. The user can then choose 'Import' and the XML is imported into the program that called the DLL. This DLL can be called from almost any program. The Microsoft Excel files presented in chapter 3 save and retrieve information from the database in this manner.

Another way to view the data is through the web interface which can be found at: http://fdot.ce.ufl.edu/fdot/dot_login.aspx. The interface also contains the ability to handle the security functions (ex: creating users, projects, etc.) for the database. After the user logs in they have the ability to 'View My XML Tree' which takes them to a similar tree as viewed in the DLL. Besides the default view, which corresponds to the DLL's version, they also have the ability to view their results in table format. This allows for them to view certain tags (ex: SPT_Data) in table format rather than the tree view. The table format puts data like SPT data into a more recognizable table form. The table format also allows for the user to modify individual attributes (data) directly without having to submit an XML document through the DLL. Like using the DLL, you choose the node that they wish to modify or view and click on the generated link. The link will take the user to a form (table view) with their data elements values that are able to be modified. When they push submit, their changes will be posted (updated). This form would also be where security approval and review would take place.

4.4 TECHNICAL OVERVIEW OF DATABASE AND APPLICATIONS

The FDOT system contains 7 main components: the database, the XML schema, the XMLTrans DLL, the ProcessUserSecurity program, the Excel interface, the web interface, and the core program that handles the conversion, processing and the security validation between the XML formatted data that the DLL and the web interfaces use and the database format. A discussion of each follows.

4.4.1 Database

The database currently exists as a SQL Server 2000 database. The database organization, while very similar in structure to the XML schema, is independent. The hierarchy must be very close to the XML schema's so that the conversion between using a series of unique attribute identifiers translates to the database identifiers and parents will be created prior to children needing to link to them. The tables that are responsible for handling the conversion of data between the XML schema and the database organization are:

- **Data_Relationships**: This table keeps track of the relationships between the data tables and their primary keys. It is accessed heavily during the code that creates the xsl conversion files and the ProcessUserSecurity program.
- **XML_Elements**: This table handles the direct relationship between an element name and the corresponding table name. The 'Data_Table' field represents the database table that the element matches up to. All of the element's attributes must have corresponding fields inside the database table. The 'Std_Format' field represents what the element name will be in the XML. The table may be expanded later to support the conversion to multiple XML schemas. Currently, only the 'Std_Format' that represents the UF/FDOT XML format is implemented.

  The 'Shell_Element' field is a Boolean field that identifies whether or not the element has any element or if it is a 'shell' that is only meant to organize other elements. An example of a shell element would be the Subsurface element under Projects. It has no attributes of its own and is only meant to contain the Hole element.

  The 'Share_Table' field is an indicator if the element relates to a table that another element also relates to. A shell element still needs to relate to a table for the hierarchy conversions to work properly even if it has no attributes. Its 'Data_Table' would be set to its Parent element's 'Data_Table' and it would have 1/True for the 'Share_Table' field. The parent element would NOT have 'Share_Table' set to True since it needs to be called first and then any sub-elements will be processed. There are many cases where an element would have 'Share_Table' as True but not be a shell element. In these instances the elements that are sharing the table all have a 1:1 relationship and a user with security for one of the elements would definitely have access to the sibling elements. The multiple elements are only being used for organizational purposes to separate the data into logical blocks. An example of this would be the Statnamic child elements (Loadcell, Elevations, STN_Load, etc.).

- **XML_Element_Links**: This table manages the relationship between all of the elements. An element's 'Element_ID' is matched up to its parent element's 'Element_ID' in the table's 'XML_Parent_ID' field. The only element with no parent is the root element: Project. Every XML document processed must have Project element or it will not be considered. This table **IS** the XML schema. If an element is not listed in this table then the xsl conversion files will not know it exists.
- **XML_Attributes**: The table contains all of the attributes that elements can have. The table matches an attribute name to a field name but not to a field of a particular table. In this respect, an attribute can be reused if another element's attribute of the same name matches up to a table with the corresponding table's field name that is also the same as the original element's table's field name. The attribute's name is represented in the 'Std_Format' field. This table could also be expanded to support multiple XML schemas. The table field's name is 'Data_Field' and the table is understood to be the attribute's element's table.

- **XML_Element_Attributes**: This table's main responsibility is to relate an element to an attribute. There are a number of options an element's attribute has. If the attribute is to be displayed in the skeleton version of the XML tree then the 'DisplayInTree' field is set to 1/True (*the skeleton version is designed to have the smallest XML file possible but still show enough information of interest for the user to choose an element that they then want the full information for*). If the attribute is an Identifier for this element (*ex: Project_Number for the Project element*) then set 'Identifier' field to 1/True. The 'VarType' of the identifier must also be specified so that it is handled properly during the database calls (*String Values: str, Integer values: int, Decimal Values: float, Boolean Values: bln*). If the attribute is a Database ID field (*ex: Project_ID*) then set 'DBID' to 1/True. The Database ID field attributes are only used by the DLL to easily filter the XML files for the user (*more details will be discussed in the DLL section*). If this attribute is one of the attributes that the DLL can search by then set 'SearchField' to 1/True. And the last field, 'OrderBy', controls the order that attributes are displayed. The higher the 'OrderBy' number the lower it will be displayed. This allows us to make sure that certain attributes will be listed after others and before others in the XML trees and forms.

The Database hierarchy relates fairly closely to the XML schema but is organized with security in mind. Each table has a series of fields that represent if the levels of approval for a record: Submit_Lock, Review_ID, Approve_ID (represents District level approval), and DOT_ID. The user who submits the data has their UserID recorded in the table's Submit_ID field and the date is recorded in the Submit_Date field. When a user decides that their data is final and should not be modified anymore then the Submit_Lock field becomes 1/True. The other levels of security can be used if desired.

4.4.2 UF/FDOT XML Schema

All uploads and downloads from the database are in XML. The format of the XML files are defined by the settings in the database as mentioned in the XML schema document XML_list.doc located at http://bsi-web.ce.ufl.edu/db/. The root element is 'GML'. All data elements are located inside this element. The root data element is 'Project'. All data must belong to a Project and be within the Project element. Table 4.1 presents a segment from the XML_list.doc that describes the schema.

Table 4.1 XML Schema and Relationship Level

Level	Name	relationship to prev level	detail see page
0	**GML**	**1**	
1	**Project**	**1-∞**	**5**
2	**Subsurface**	**1-1**	**5**
3	**Hole**	**1-∞**	5

The Level represents the element's location in the hierarchy. The Project element is the only element at Level 1 since all data must be under a Project. Subsurface is Level 2 and is located inside the Project table. The **1-∞** next to the Project shows that the GML tag can contain multiple Project elements. This means that the user can upload data for multiple Projects at the same time. The Excel spreadsheets currently do not have a way to handle multiple Projects but that is only a limitation of the spreadsheets, not of the application that listens for transfers. The **1-1** next to Subsurface shows that there can only be 1 Subsurface tag inside each Project. However, Table 4.1 also shows that the Subsurface element can contain multiple Hole elements. Below in Table 4.2, is an excerpt from Page 5 of XML_list.doc that shows some detail information of the Project element.

Table 4.2 Project Element Information

level	name	type		tag	attr	description	unit
0	**GML**		**1**	**x**			
1	**Project**		**1-∞**	**x**			
	Name	str			x	project name	
(id)	***Project_Number***	***str***			***x***	***project number***	
	Financial_Number	str			x		
. . .							
3	**Hole**		**1-∞**	**x**			
	SI_Unit	boolean			x	either "true" or "false"	
(id)	***Hole_Name***	***str***			***x***	***boring name (or #)***	
	Latitude	real			x	GPS coordination of the hole	
	Longitude	real			x		

The ***(id)*** next to Project_Number shows that ‘Project_Number’ is the Identifier attribute. This means that the Project element **must include** a ‘Project_Number’ attribute in order to be processed. An Identifier must also be unique under that element’s parent. Since Project has no parent (with regards to the data hierarchy, GML is only there to form a valid XML file) there can not be any to Projects with the same identifier. Hole’s identifier is ‘Hole_Name’. Hole is inside a shell element ‘Subsurface’ that contains no attributes so one looks at Subsurface’s parent Project. There can only be **one** Hole with a particular Hole_Name under a Project. For instance the XML:

<GML><Project Project_Number=’TestProject’><Subsurface><Hole
Hole_Name= ‘TestHole’/></Subsurface></Project></GML>

The XML above is valid. There is only one Hole ‘TestHole’ that belongs to Project ‘TestProject’. The XML above would locate the particular Hole in question and be able to update its information.

<GML><Project Project_Number=’TestProject2’><Subsurface><Hole Hole_Name= ‘TestHole’/></Subsurface></Project></GML>

The XML above is also valid. There is a Hole ‘TestHole’ that belongs to another Project ‘TestProject2’. These are different Holes. This is why you must make sure that all of your element tags contain the appropriate identifiers. All identifiers must be present to make sure that the correct records are processed. If an identifier is missing, it will not be processed. You must also pay attention to the ‘type’ description. This represents what format the data must be in to be processed correctly. If you have an attribute that should be in ‘real’ format (decimal number) and it contains a string, the process will fail.

4.4.3 The XMLTrans DLL

The DLL has two main purposes: uploading and downloading. It was developed to provide an easy interface to the database accessible from any program as follows.

4.4.3.1 Uploading - The upload component has been designed to send an XML string (created by the application calling the DLL) from the application to the database server’s waiting listener. The server will take care of validating the user’s security access and converting the XML to the database format. The application can choose to have a log returned after the processing is complete. In order for the application to use the DLL, the user must download and install the DLL. The most current DLL is available at http://bsi-web.ce.ufl.edu/db. The current one is labeled version2_0_04.zip and was last modified 11/9/04. If someone is setting up an application to use the DLL then after it is installed, they must add a reference to the DLL inside their application. In Excel this was done through the Visual Basic Editor and then in the menu Tools>References and adding XMLTrans. The commands below are all one needs to send the information to the DLL.

```
Dim obj As New FastXML
Dim y As String
y = obj.UploadXML(uName, uPass, ThisData, True)
```

uName: Username with security to update the data being sent
uPass: Password that corresponds to the Username
ThisData: complete XML string
True/False: whether or not the application wants an upload log returned. If the programmer enters ‘True’ a message box will be created from the DLL and contains a link to the Upload Log.

For the upload process, the application is also responsible for handling and the entering of the User’s credentials as well as handing them to the DLL.

4.4.3.2 Downloading - To download through the component the application will be asked to log in through the DLL interface. If the credentials are valid, the user account will be shown a skeleton tree of the information they have the credentials to view. If they have not been granted view access to particular fields for an element then their values will be cleared. If they have not been given access to entire elements, those elements will not be shown at all. If the user has extensive access, the processing and sending of the skeleton XML file may take a while. The size of the XML file is the main reason a skeleton file is used. This ensures that only the information that was deemed necessary and the most helpful for identifying an item of interest will be downloaded through the Internet to the DLL and displayed in the tree. It is recommended that the user drill down through the tree until they find the item of interest before they choose to ‘Get Attributes’. This ensures that their XML file is not larger than it needs to be since when they select the node (item) and choose 'Get Attributes' all attributes for that item's parents **AND** all attributes for **any and all children** elements will be sent. If the user chooses 'Get Attributes' at the Project level for instance, the returned tree will contain **ALL** elements and attributes under that Project (as far as your security access will allow). After the user 'Get Attributes' for their chosen node they can browse through the tree with the full collection of attributes. If they wish to return the results to the application that called the DLL, they would hit 'Import'. The commands below are used to access the download component:

```
Dim obj As New FastXML
TreeStr = obj.GetTree
Do Until obj.IsTreeReady = 0
   'Wait for tree to be ready
Loop
```

The Do Loop was added so that the application would not continue processing until the tree was ready.

The downloading interface also has the option to **Search** for particular items of interest. The items the user can search by are specified by the database's 'SearchField' flag. When the DLL receives the skeleton DLL, the search fields are collected along with the tree. Currently the search keywords must be the complete value of the attribute in question (no wildcards are assumed). This feature will be expanded in future updates of the database.

The DLL code will be made available so that anyone can make their own DLL to handle the transferring of the XML information if their application is unable to use the one provided. The provided DLL is designed to handle any type of XML schema the server sends it and has no schema information programmed inside of it. The latter ensures that any changes to the schema, will not require any changes to the DLL.

4.4.4 ProcessUserSecurity Program

In order for the XML trees to be displayed quickly through the DLL and web interfaces, the security settings are saved in a database table, Access_XMLOut_Security. The table includes for each user, which database tables, records, and fields they have security to view. The ProcessUserSecurity Program creates these records in the Access_XMLOut_Security table. The program needs to be called anytime a record is inserted that the User should have security to view (for updates since no record is added that is not in the current list, the program does not need to be called). The Program is designed to process all users that are currently in the waiting table 'UserSecuritytoProcess' OR processes a particular user by sending the program a command argument. The program is not web accessible and can only be called by a user on the server OR by the uploading process of the listener program OR by XML tree pages if they are listed in the waiting table. After an insert is made to a table, the uploading process adds the user to the 'UserSecuritytoProcess' table and then adds all users to the table that also have security to view the same table and record that was just created. After all of the uploading is done the upload process then calls the ProcessUserSecurity program to run asynchronously so that the user who did the uploading does not wait for the ProcessUserSecurity to finish.

When the program is called, it first checks whether an argument was sent. It is expecting "UserID:#" where # is the Database identifier for a user. If a UserID was sent then only that user's security is processed. Otherwise, all users listed in the waiting table (UserSecuritytoProcess) are processed. After a user's security is processed they are deleted from the UserSecuritytoProcess table.

When a user accesses one of the XML tree interfaces (web or DLL), the page checks whether the user is listed in the waiting table. They only have their security processed once, even if they are listed multiple times. This ensures that if another user has done a series of inserts to data that the current user has the security to view, their security will be processed and updated but only once and only when they actually need it to be. This cuts down on needless processing for user security. If a user does not access any of the tree interfaces for weeks, months, etc. and then they finally do view the page, their security will only be processed that first time when they view the tree and will have avoided being processed for all those other instances records were added. Also, if they return to the tree the next day and no records have been added, they will quickly access their relevant records and not have to wait for the security to be processed.

4.4.5 Excel Interfaces

Each Excel interface has the ability to upload and download data however, only the data that the Excel file is designed to handle will be uploaded or imported from the DLL.

4.4.6 Web Interface

The Web Interface includes the XML tree page and also the interface for assigning security access to user accounts. The interface is located at: http://fdot.ce.ufl.edu/fdot/dot_login.aspx

4.4.6.1 XML Tree Page - When the user accesses http://fdot.ce.ufl.edu/fdot/xml_tree2.aspx they will be asked to log in. While logging in, the user has an option to select how they will want their filtered results returned to them. The options are: XML Results (the standard format, similar to the DLL's tree), Table Results, and Excel File Results.

4.4.6.2 Table Results- After the user drills down through the tree (by clicking on the + signs) and clicks on their chosen node/element, the results will be processed. The function 'ProcessSelection_TableView' is called. The function creates an XSL file dynamically that will filter the selected element and create an appropriate Form based on the element's content. After the xsl file is created, it will be applied to filter the full XML file and return the form as its end result. A link to this form will then appear after processing is completed. The form allows for a user to modify an individual record (that is not part of a _Data or _Log element) so that the user will not need to process the entire XML for a small change. This form is also used to Lock the data by a Submitter. When a Submitter chooses to Lock their data, only their account will be able to update the data fields. All of those items sub-elements are also unavailable for updating. Another way to make the data unable to be edited would be for a Reviewer, District Approver, or a DOT Admin to approve/lock the data. Those accounts (Reviewer/District/DOT) would also use the same form generated from the XML tree to approve/lock data.

If the user chooses an element that contains a Data element (ex: SPT contains a collection of SPT_Data elements) then the _Data collection of elements will be displayed in table format. _Data and _Log elements are not able to be updated individually since they are uploaded in a group and have no Identifier attribute to locate which one of the _Data/_Log records should be updated. When an update is made, the original _Data/_Log records are deleted and the new ones are inserted.

Certain elements have special features in their form. One element that has a special feature is the Hole element. At the bottom of the generated form, there is a link to the PDF file upload form for a Hole element. As long as the Hole and Project are not locked, the user can overwrite the PDF file associated with the Hole record.

4.4.6.3 Excel Results- An Excel file is dynamically generated with the selected node's attributes on one worksheet and any _Data, _Log sub-elements on another worksheet.

4.4.7 Core Program

The core program is a Visual Studio .NET 2003 Project written in Visual Basic .NET. There are 4 main pages that are used: xmlin.aspx, xmlout.aspx, xml_tree.aspx, and xml_tree2.aspx.

- **XMLIn.aspx**: This page handles the processing of uploaded data. The submission from the DLL and the generated form pages are processed as a form post. The Username and Password are immediately checked to make sure that the credentials validate to an existing user. If the credentials are valid the XML string is processed and saved as a file on the server. Afterwards, the WriteXMLFile function is called to convert the XML file from the XML schema format to the database schema format (coincides with the database table structure). The ReadXML function is then called that process the XML file and loads all of the information into a series of arrays. These arrays keep track of the XML Identifiers (ex: Project_Number) and how they relate to the Database Identifiers (ex:Project_ID). For instance, the full XML Identifier list to locate a particular Pier record would be:

Project_Number=TestingProject,Bridge_Number=111,Pier_Name=TestingPier. Those Identifiers would then match up to a particular Pier record with the Database Identifier being Pier_ID=7. After the Arrays are built with all of the hierarchy and identifying information the ProcessSQL function is called. ProcessSQL will process the arrays and as a record is updated will update the arrays with the Database Identifier information. Then when the next element/record is processed it will know its parent element's Database Identifier. That way when we go to look up the information for that element we only need to find where FieldName=XML Identifier Value and Parent = Parent Database Identifier (ex: Select * from Piers where Pier_Name=TestingPier and Bridge_ID=3). The Arrays also keep track of whether that item is locked and then when a child is attempted to be processed it immediately cancels the operation. The ProcessSQL function calls a number of security related functions that make sure that only the fields/attributes the user has access to update will be updated and records their ID and Date at time of submission. The Date that is used for an update/insert will be the same for the entire update session. A log is continuously updated throughout the process and can be returned after completion. If an upload is attempted with an element that contains a unique identifier that does not currently exist under its parent, a new record will be created automatically (if they have the security for the XML identifier field). After an insert is made (a new item entered), the user's ID will be added to the UserSecurityToProcess table. After the entire upload session the ProcessUserSecurity function will be called.

- **XMLOut.aspx**: This page creates all of the xsl conversion files dynamically from the XML tables. These files are called during any type of conversion between the database format and the XML schema formatted XML files. The page is currently set up to just access the page and all of the necessary files (full, skeleton, etc.) are created. Anytime new XML elements or attributes are added to the schema, the xmlout.aspx page needs to be processed and the created xsl files need to be moved over to the server.

- **XML_Tree.aspx**: This page is called by the DLL to deliver the XML tree. There are 2 Phases of processing between the page and the DLL, Phase 0 and Phase 1. At Phase 0, the User's credentials are validated only. If the user is valid, the page is redirected to Pending.html which the DLL is expecting. The DLL then sends another request with Phase=1 and the User's credentials again and the tree is processed. This 2 Phase process is done to allow the DLL to notify the User what is happening and why there may be a delay. There is no wasted time if the User is not logging in correctly; they know that the wait is for the results. When the tree is ready, the page redirects to the final XML file with the name being a GUID. The DLL then sends that GUID later to notify the server whether a result should be filtered and what the user selected. Each time the final result is delivered to the DLL by redirecting the current page to the final XML file.

- **XML_Tree2.aspx**: This page is used with the web interface. It allows for multiple options of viewing the XML results as discussed in the Web Interface section.

4.4.8 Security Design

The key tables that control the Access Security are: Access_UserAccounts, Access_Tables, Access_Levels, Access_Jobs, Access_Job_Levels, Access_UserSecurity.

- **Access_UserAccounts**: This table contains all of the User's information: Name, Email, Company, Password, etc. An email address may have multiple user accounts and a user account **must** have an email address. When a user account is created by a DOT Admin, a generated email is sent to the user. The user always has the ability to log in to the user interface and modify the password. If they are not able to log in correctly because they have forgotten either their username or password, they can fill out the Forgot Password form and have their account information sent to them by filling out the associated email address or the username. The table also tracks when the user was created and what user account created the user.

- **Access_Tables**: This table matches up with an Access_Level (ex: Projects_View, Access_ID=1) that states which tables and fields that Access_Level have the authority to view. An Access_Level can relate to multiple Access_Table settings. Access_Level Projects_View's Access Table record would look like:
    - **Access_ID = 1 (Projects_View)**
    - **Access_Table = Projects**
    - **Access_Field_Name = <NULL> (means all fields if none specified)**
    - **Access_Flag = 2 (View, The Access_Flag values are defined in the Access_Flags table)**
- **Access_Levels**: This table allows for a collection of table and field settings to be represented by Access_ID. This means that multiple Jobs will be able to use the same settings over and over again, simplifying the database as a result.
- **Access_Jobs**: This table contains the names and descriptions for each Job that a User can be assigned (ex: Lab Rock Poster).
- **Access_Job_Levels**: This table relates the Access_Jobs to the Access_Levels. A Job can contain multiple Access Levels.
- **Access_UserSecurity**: This table relates a User account with an Access Job for a particular Project. It also tracks when the security setting was created for the User and which User set the security.

4.5 WEB INTERFACE OVERVIEW

The Web Interface has been designed for the DOT Administrators to be able to create users, create projects, and assign users access security rights to projects. The typical user can also access the web interface to view/modify their account settings (name, email, etc.) and view their security settings for projects. Users are notified by email anytime a change is made to their account information or security settings. The database also records what user account (DOT Admin or otherwise) created a user, when the user was created along with the user that assigned a particular security job role to a user for a particular project and when the assignment was made. An email address is required for all user accounts.

4.5.1 Log In to the Web Interface

At the website address: http://fdot.ce.ufl.edu/fdot/dot_login.aspx, the user will see the following Log In Form, Figure 4.1.

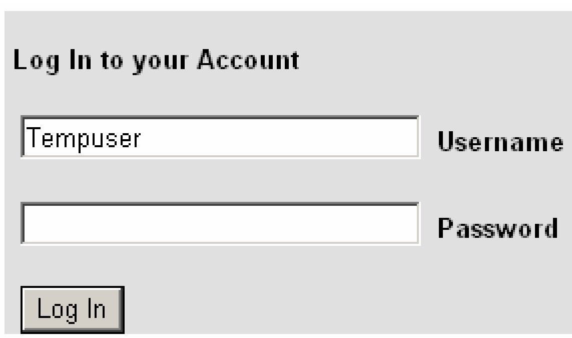


Figure 4.1 Log In Form

The user should enter the Username that they were emailed when an administrator created their user account. At the time of user creation, a randomly generated password is created and emailed to the user's email address.

If the user does not remember their credentials correctly, they will be shown this message:

The user should click on the 'Forgot Password' link as shown above to access the Forgot Password Form. The form is shown in Figure 4.2

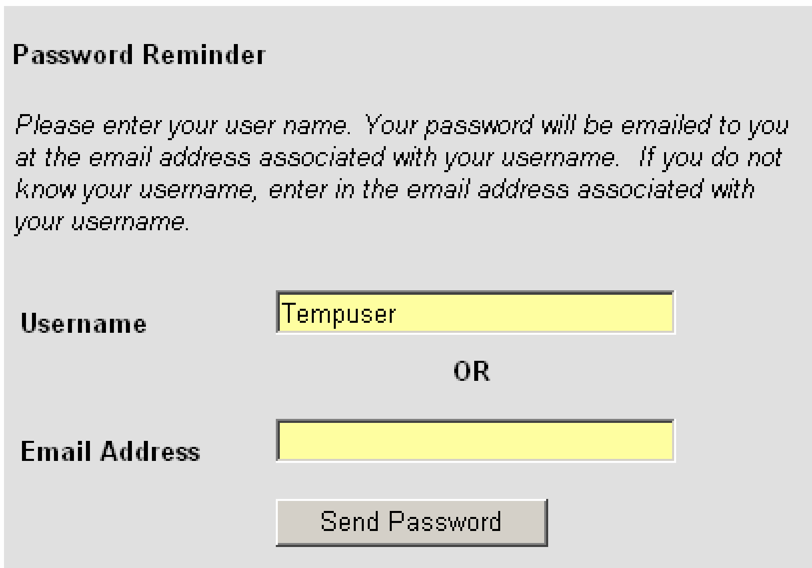


Figure 4.2 Password Reminder Form

As described in the form, the User can either use their Username (if they know it) or their email address to retrieve their account credentials. The information will be emailed to the email address they specify or the email address associated with the username they enter.

When a user account with DOT Admin security settings logs in, they see the option panel show in Figure 4.3.

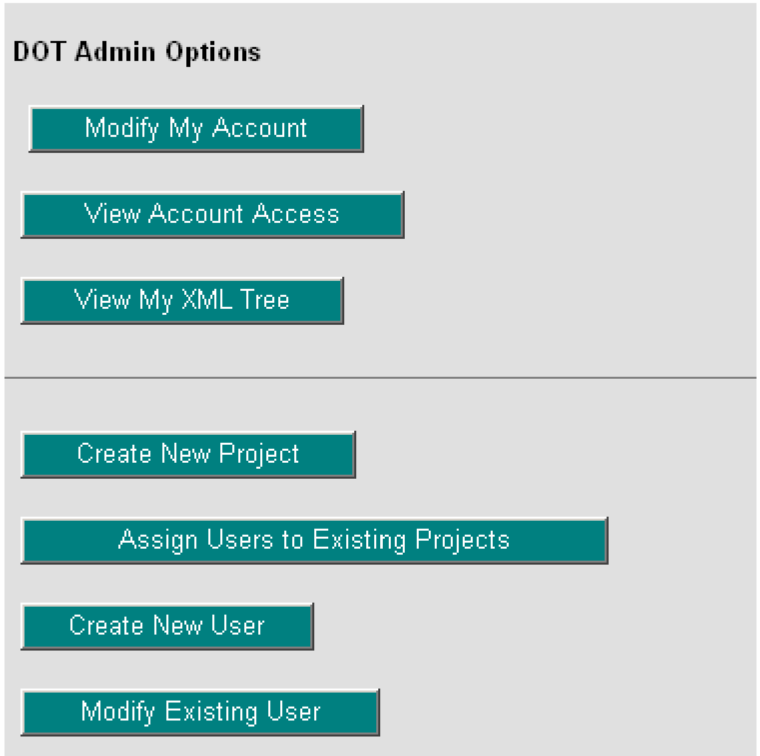


Figure 4.3 Administrative Logon Form

When a standard user successfully logs in their option panel appears as in Figure 4.4.

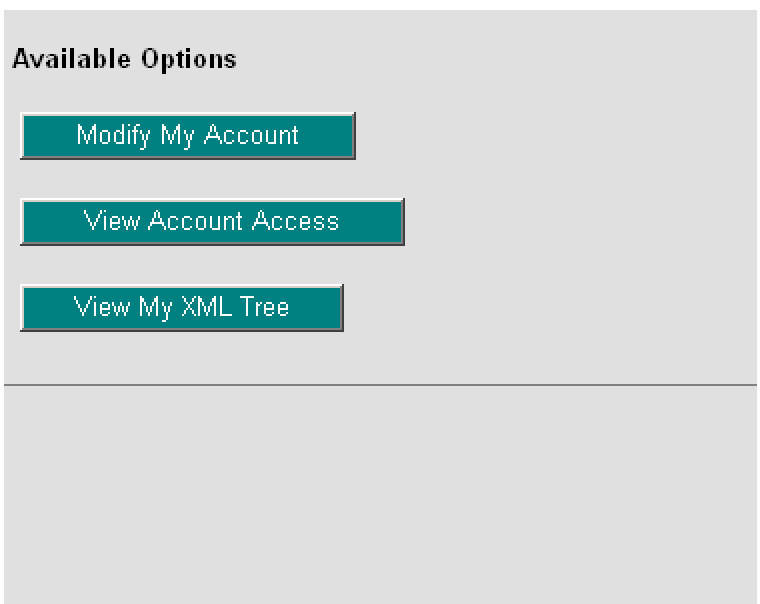


Figure 4.4 Standard User Option Panel

The top portion of the form with the three options: Modify My Account, View Account Access, and View My XML Tree are available options for all user accounts.

4.5.2 Modifying Personal User Account Information

Every user account is able to modify their information (Name, Company, Password, etc.) from the user option panel, Fig. 4.4, and selecting the 'Modify My Account' button. This takes them to the following form, Figure 4.5.

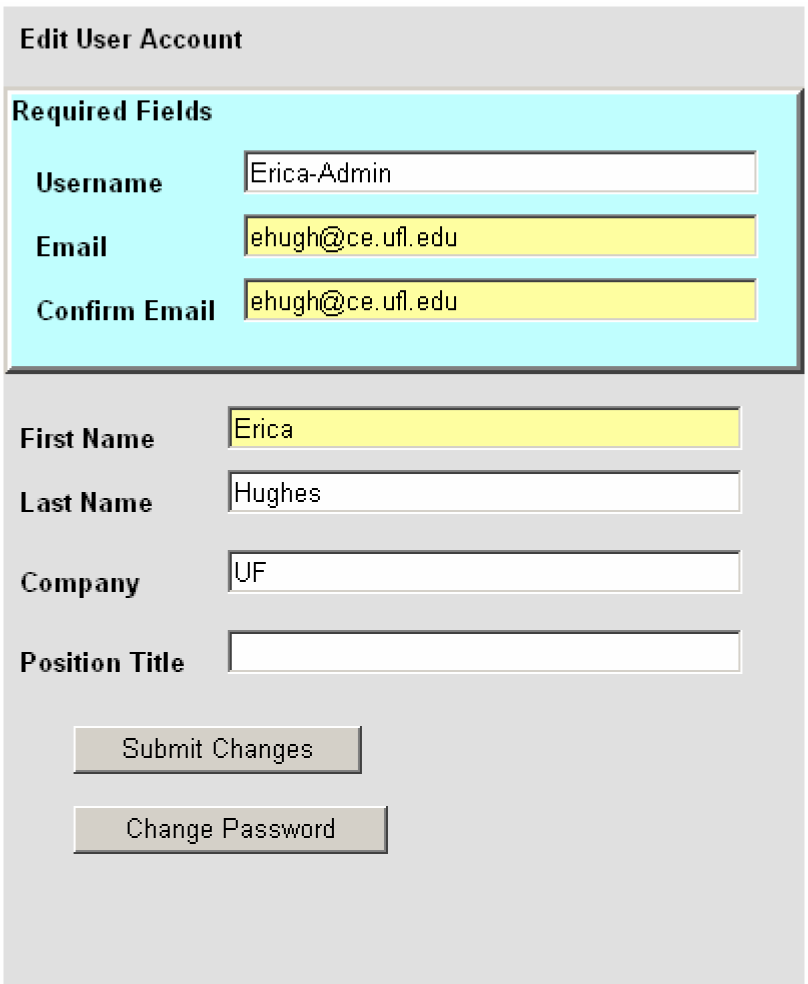


Figure 4.5 User Account Information

The 'Change Password' button will take them to a separate form for changing their password, Fig 4.6. At this screen, if they choose to 'Change Password' it will **NOT** submit any changes they have made unless the "Confirm Password" dialogue box isn't filled in.

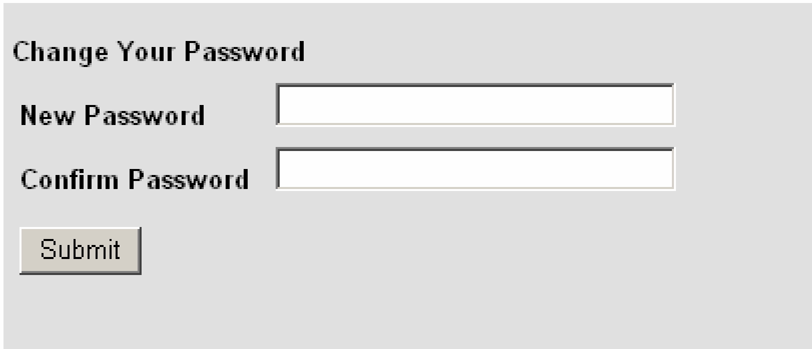


Figure 4.6 Change Password Form

If the 'New' and 'Confirm' passwords match, the password will be immediately changed when the user clicks the 'Submit' button.

4.5.3 Viewing Personal Security Access

If the user clicks the 'View Account Access' button on their options page, Fig. 4.4, they will see the following form, Figure 4.7.

Figure 4.7 User Security Access

The panel shows what security job role the user is assigned to (they can be assigned to multiple roles) for each Project that they have security for. If the user would like the details regarding a job role listed, they should click on the hyperlinked job name. If the user had clicked on 'Holes Full Poster', Fig. 4.7, they would have a new window opened with the below information, Figure 4.8

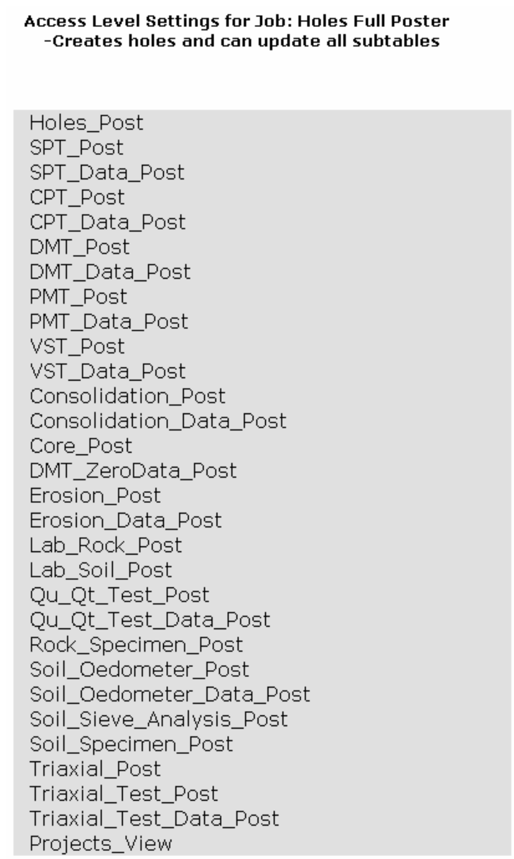


Figure 4.8 Project Access Control

The top part in bold gives a description of the job role and the type of access it has. The list below represents what Access Levels the Job corresponds to. These are the actual settings the user has for the particular project. The top item in the list 'Holes_Post' identifies that the user has posting access (full submitter rights) to the Holes table.

A DOT Admin account which has rights for **all** projects will only see the message: **Your account is currently a DOT Admin account.**

4.5.4 Viewing Personal XML Tree

If the user would like to view their XML tree data (all items that they have authority to view represented in a data tree) then they would click the 'View My XML Tree' button from their options page, Fig. 4.4. It will take them directly to http://fdot.ce.ufl.edu/fdot/xml_tree2.aspx. The user should not have to log in again if their Session has not expired. The XML tree page can be used by the users to make minor corrections to their existing records. It is also used by administrators (DOT/District/Reviewer) to approve/lock a record. To access those forms, the administrators would need to view the XML tree as 'Table Results'. They would then drill down through the appropriate nodes until they found the item that they wanted to approve/lock and click on it (not the + sign but the description text). After the form has completed being generated from the filtered results, a link will be made available at the top of the page labeled 'Your Table File'. A sample approval form for a Project is shown in Figure 4.9. The figure shows the user account 'Erica-Holes' has submitted Reviewer Approval for this Project which prevents any user from making any changes to the Project record or any of its subtables (Bridges, Holes, SPT, etc.). The Project still awaits DOT Approval which is the final stage. An intermediary approval level can also be used, the District Approval. These levels can be used however desired and any user account could be set as the approver for a particular security level for a particular item (ex: District Approver for Hole records under the TestingErica Project). If any of the levels of approval are set then the data is locked and unable to be modified.

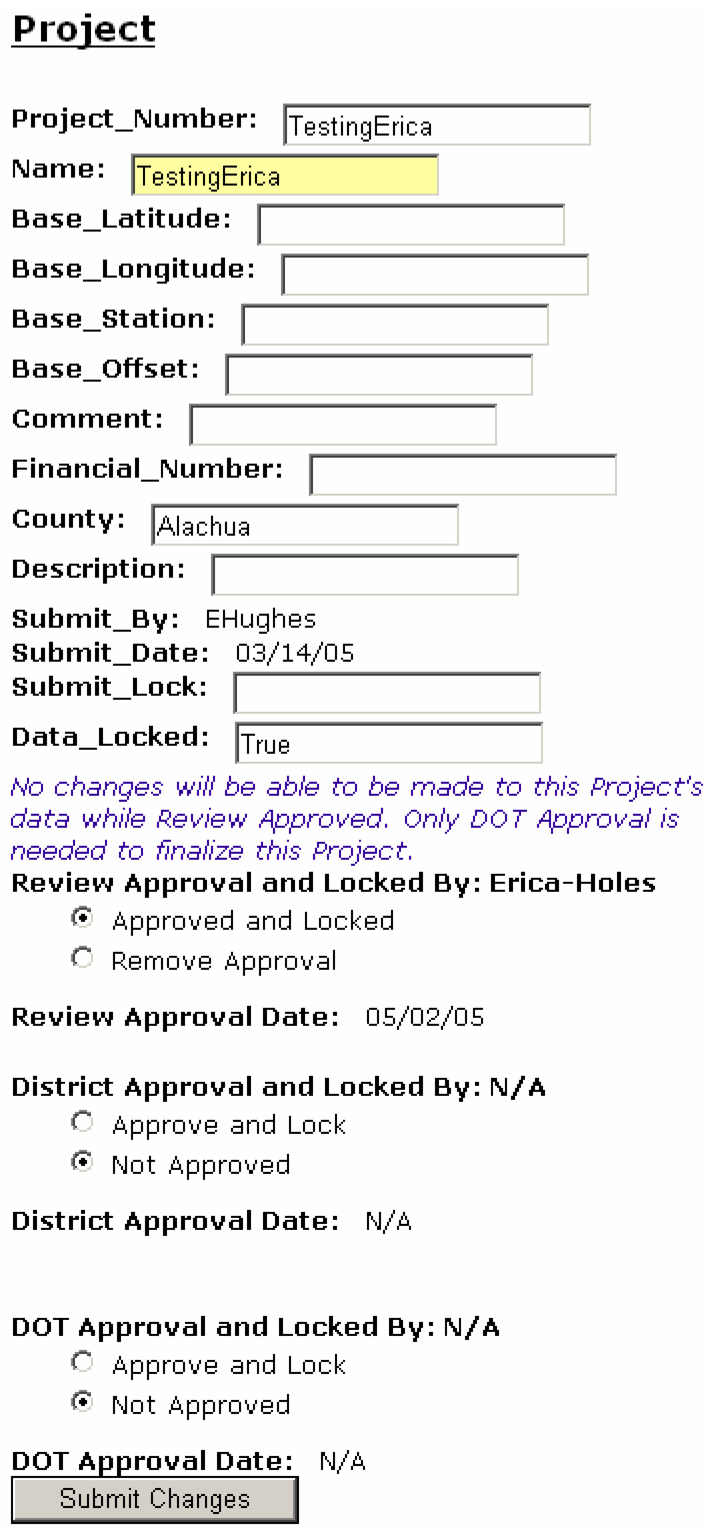


Figure 4.9 Project Approval Screen

4.5.5 Create New Project – DOT Admins

Only DOT Admins can create new projects. The DOT Admin is then responsible for assigning a user to administer the Project who would then be able to assign other users to individual roles for the Project. The DOT Admin will need to click on the 'Create New Project', Fig. 4.3 from their options panel as shown (Fig. 4.10).

Figure 4.10 Creating a New Project

Selecting Create a New Project, Fig. 4.10, will result in the DOT Admin generating the following create project screen, Fig. 4.11,

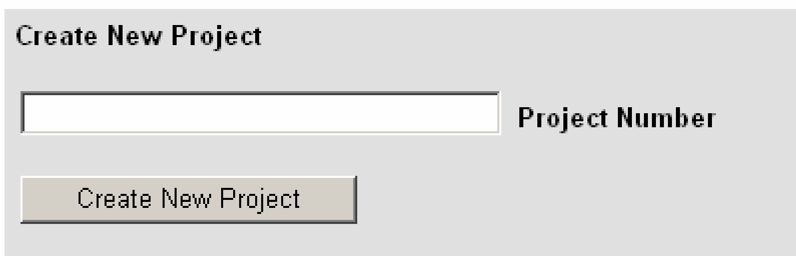


Figure 4.11 Screen to Create New Project

The DOT Admin will be required to enter a unique Project Number for this Project. No two Projects may have the same Project Number. The project will be instantly created after the admin clicks the 'Create New Project' button.

After the Project is created, Fig. 4.11, the DOT Admin will also automatically display the new user screen for the Project, Fig. 4.12. Until a user account is assigned, no one will be able to update/view the project.

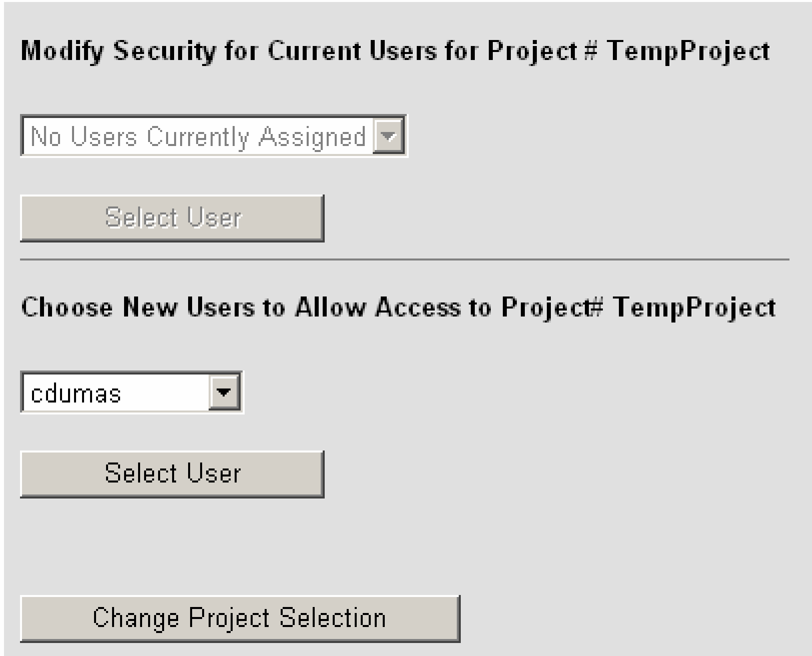


Figure 4.12 Selecting Users for Projects

This form, Fig. 4.12 is also used to assign user accounts to existing projects. The top portion represents Users that already have some type of security access for this Project. The bottom section is all of the users that do **not** have any type of security access specified for this Project. This helps administrators easily identify who already has access and who still needs it. Since this is a new Project, there are currently no users assigned and the DOT Admin will need to select a user from the bottom pulldown. When the admin has selected a user they would like to assign to the new project, they click the 'Select User' button for the appropriate section, Fig. 4.12. They will then be directed to the User's project access screen, Figure 4.13.

Figure 4.13 A User's Access to Project Information

Here the Administrator will check the boxes for each job role they would like the user 'Erica-Holes' to be assigned for Project# 'TempProject'. The Admin can make changes to these settings at anytime and quickly remove or increase access. If the Admin would like to have more information about a particular job role, they can click on the hyperlinked job name. It will take them to the same type of page as described in the Access Levels section above. To finalize the security assignment, the admin will need to click the 'Submit Selections' button. After a security setting is created or modified for a user, an email is sent to the user's email address notifying them of any additions, removals and what their current access is. They are also notified of the Admin's username that made the changes. The database also records what admin account created a security setting for a user and at what time.

After assigning a user specific access, the Admin clicks on submit selection, Fig. 4.13, and they then be given the option to assign another user to the same project or return to the options page, Figure 4.14.

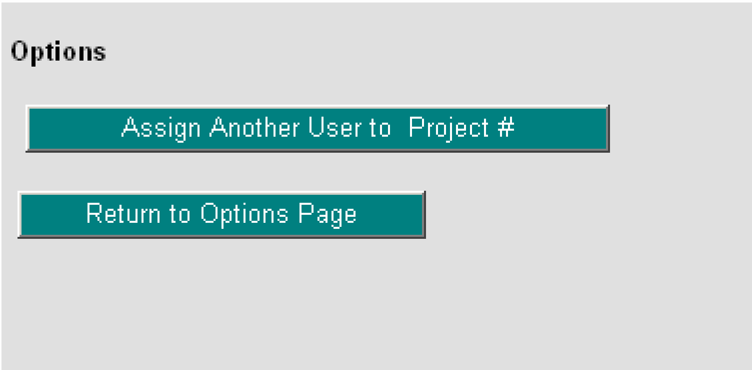


Figure 4.14 Admin User Options Screen

4.5.6 Create New User Account

To create a user account, the DOT Admin would click on the ‘Create New User’ button on the options page, Fig. 4.3 as shown (Figure 4.15).

Figure 4.15 Creating New User Account

A new screen, Figure 4.16, similar to what is used to modify an individual account is opened.

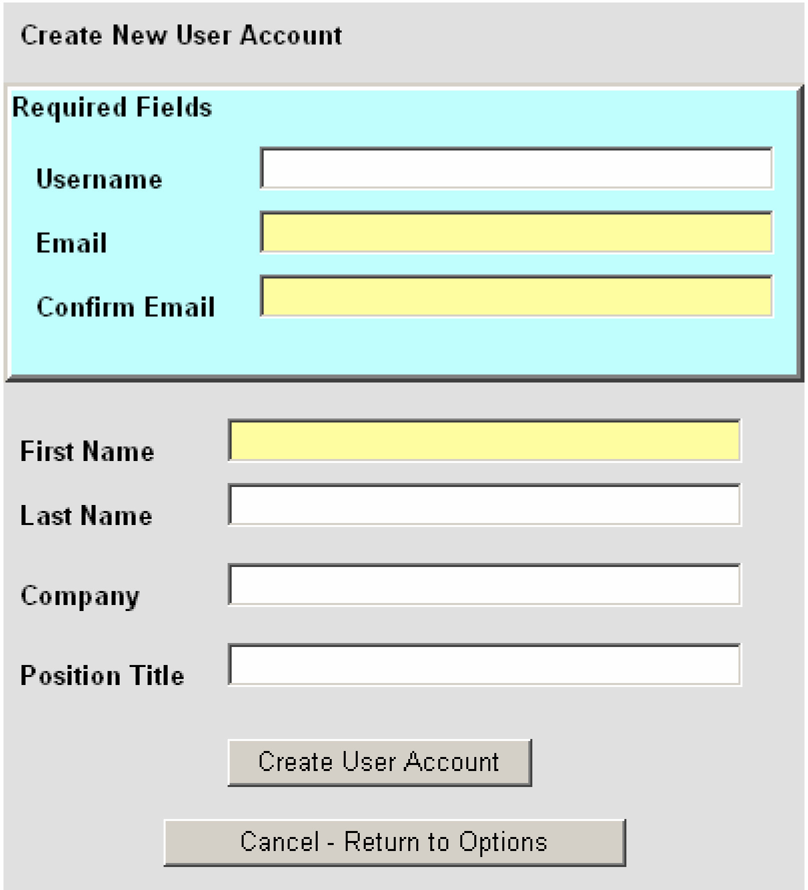


Figure 4.16 Administrative Creation of a New User

The only required fields are the email address (which must be confirmed) and the username that is to be added must be entered. This username must not already be in use. An error will be displayed if the email address is not valid or if the username is already in use. If the username is already in use, the Admin may wish to 'Cancel – Return to Options' (bottom of Fig. 4.16) and view the account information for the user in question. When the Admin has entered all of the required information and any desired additional information (Name, Company), they will need to click the 'Create User Account' button to finalize the creation. After the account is created, a password is randomly generated and all of the account information is emailed to the user along with directions on how to access the web interface and a reminder that they can change their password. The database also records what Admin account created the new user and at what time.

After the New User is created, the Admin will be given the opportunity to immediately assign the New User to a Project, Figure 4.17, create another user, or return to the Administrator Options Screen, 4.3.

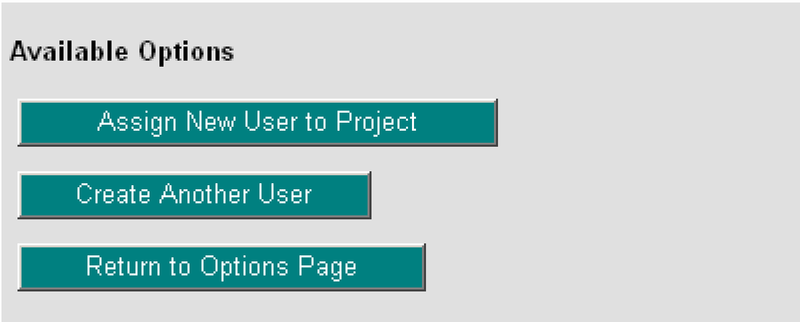


Figure 4.17 Administration Screen after New User Creation

4.5.7 Assign Users to Projects

If the DOT Admin has just created a New User, they may wish to “Assign New User to Project” from Fig. 4.17. Selecting “Assign New User to Project” button from Fig. 4.17 will open the Project Number or Name screen, Figure 4.18.

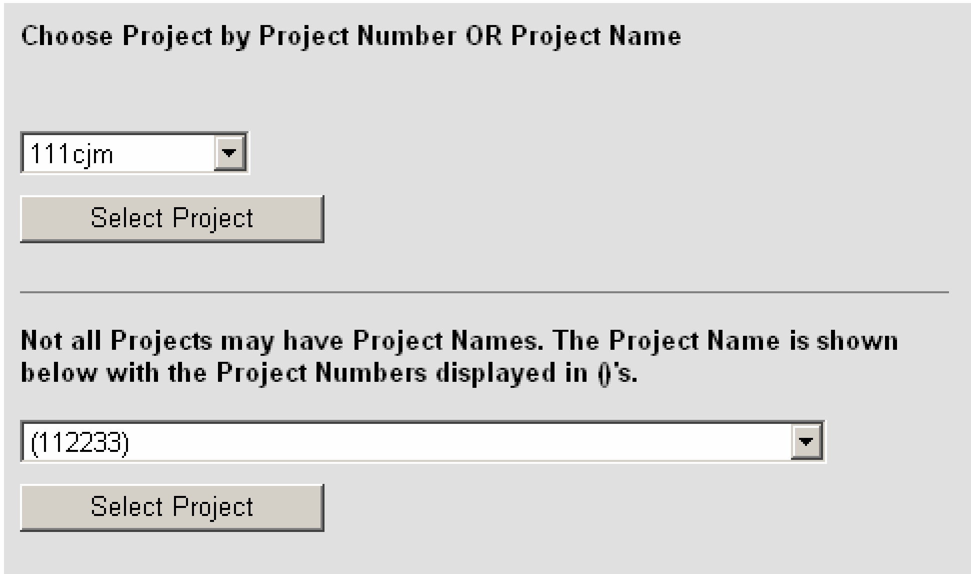


Figure 4.18 Project Number or Name Selection Screen

Note the Admin could get to the same screen by clicking the 'Assign Users to Existing Projects' button on their options page, Fig 4.3 shown in Figure 4.19.

Figure 4.19 Assignments of Users to Existing Projects Screen

The Projects are listed by Project Number in the top section and by 'Project Name – Project Number' in the bottom section, Fig. 4.20. A Project Number is also shown in parenthesis. If a Project does not have a Project Name (it is not a required field), then only the Project Number will be shown. Projects without Project Name's appear at the top of the list since the sort is by Project Name and NULL appears alphabetically before any letter.

After identifying the appropriate Project, the Admin will click the 'Select Project' button, Fig. 4.18, and the following screen, Figure 4.20 is displayed.

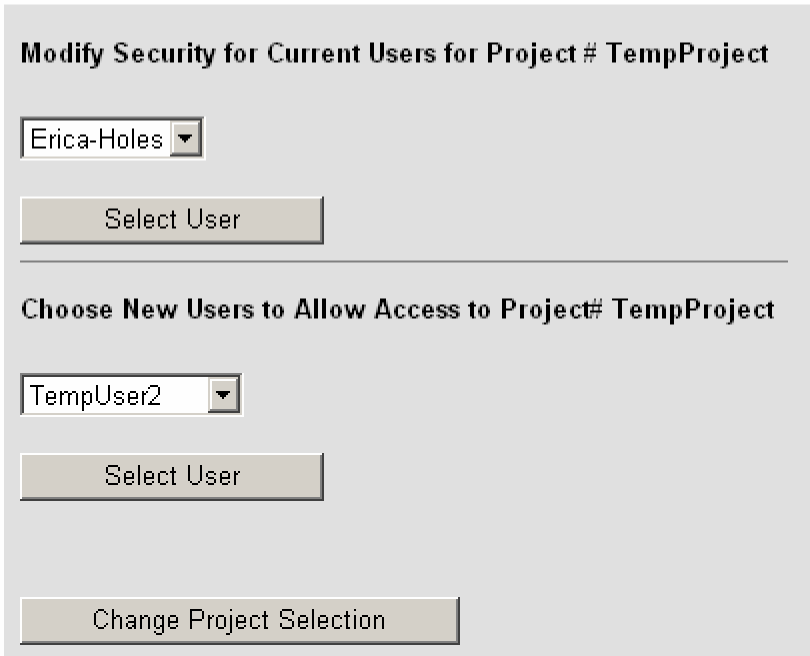


Figure 4.20 Modifying Project Security by Adding Users

If the Admin is assigning a New User account and if the account has just been created then the user will be automatically displayed in the bottom section of Users that do not already belong to the current Project. The Admin also has the opportunity to change existing user security for this project by selecting a user from the top section.

When the Admin has chosen his/her user to assign/modify security access for this project, they then will click the 'Select User' button in the appropriate section. For example, if Admin chooses the 'Erica-Holes' account that was assigned security to earlier to modify, then the following screen, Figure 4.21 is shown.

Figure 4.21 Project Access Assigned to a New User

To remove the assigned security access, the Admin simply unchecks, Fig. 4.21, the job roles that they no longer want the user to have access to within the project. They may also wish to check additional boxes to add security job roles to a given user. When the Admin is satisfied with their changes, they must click the 'Submit Selections' button, Fig. 4.21.

If the Admin had chosen a New User to assign different security roles to, a panel identical to the one in Fig. 4.21, would be displayed but a number of job roles would be pre-checked or selected.

4.5.8 Modify An Existing User

To modify an existing user (account information or security settings), the Admin would click on the 'Modify Existing User' button on the Admin options page, Fig. 4.3, as shown in Figure 4.22.

Figure 4.22 Modifying Existing Account Information

The latter selection will open the "User Account to Modify" screen displayed as shown in Figure 4.23,

Figure 4.23 User Account to Modify Screen

The Admin has a pull down window, Fig. 4.23 containing all existing usernames. After the admin has selected a user to modify, they will click the 'Select User' button, Fig. 4.23. A screen similar to initial user creation screen Fig. 4.24 is displayed as in Figure 4.16. This screen allows for account modification where the admin may change the user's username or change his/her associated email address, company information, etc.

If Admin has made any changes to the Account Information that needs to be saved, then the 'Submit Changes to User Information' button must be clicked. Subsequently, an email will be sent to the user's email address with the associated changes that the Admin has made.

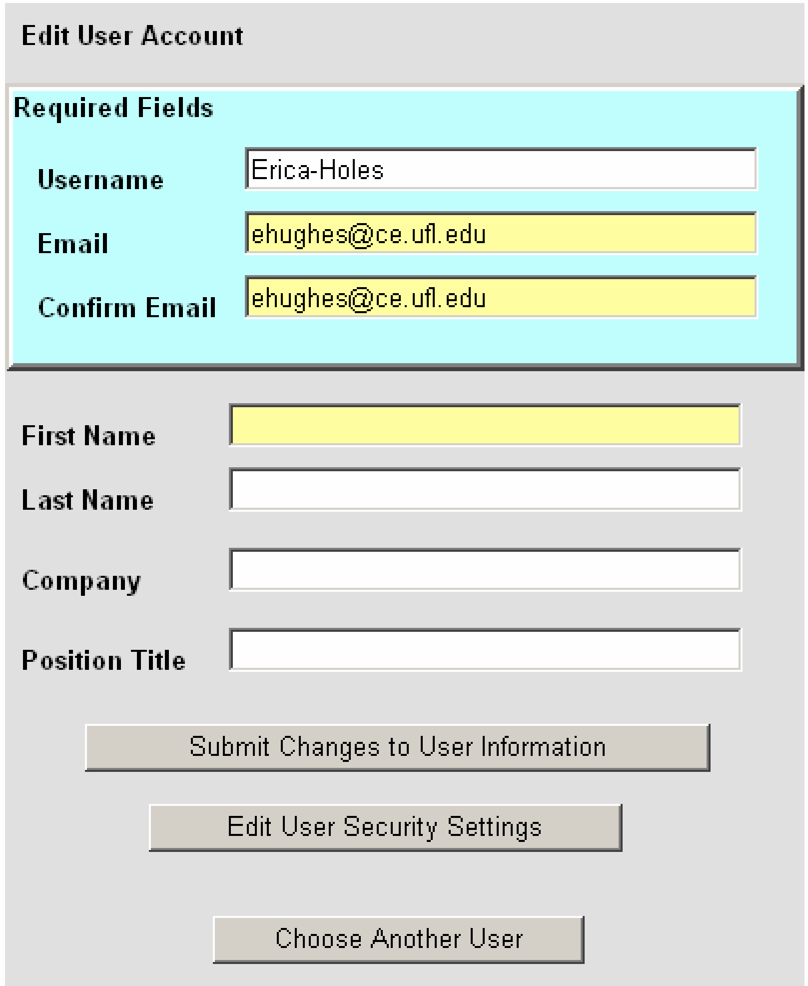


Figure 4.24 User Account Modification Screen

If the Admin would like to change the security settings for a user, the 'Edit User Security Settings', Fig. 4.24, at the bottom of the form must be selected. Subsequently, the project security screen, Figure 4.25 is displayed. The top section of the security screen, Fig. 4.25 represents all Projects that the current user has access. The bottom section represents all Projects that the current user does **not** have existing access. The form displayed when clicking "Select Project" will be identical for either the creation of a new Project, or for modify an existing user access, Fig. 4.21. For instance, if "Select Project" for an existing user project is selected, then Figure 4.26 is displayed.

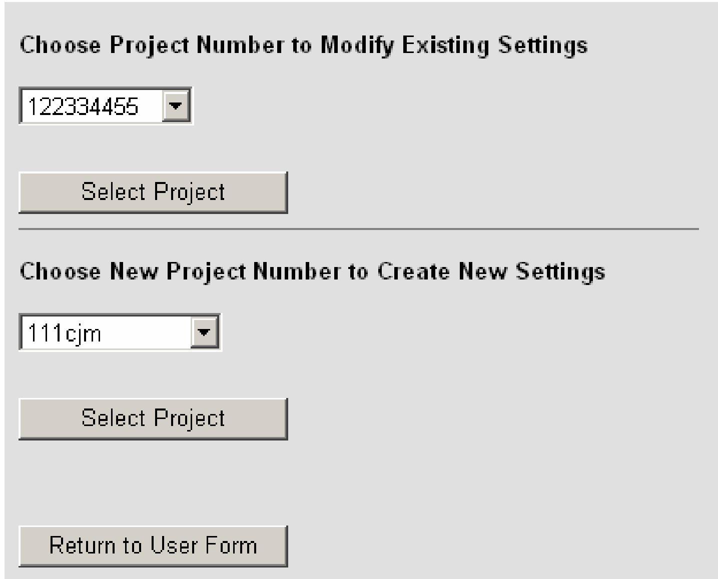


Figure 4.25 Project Security Settings

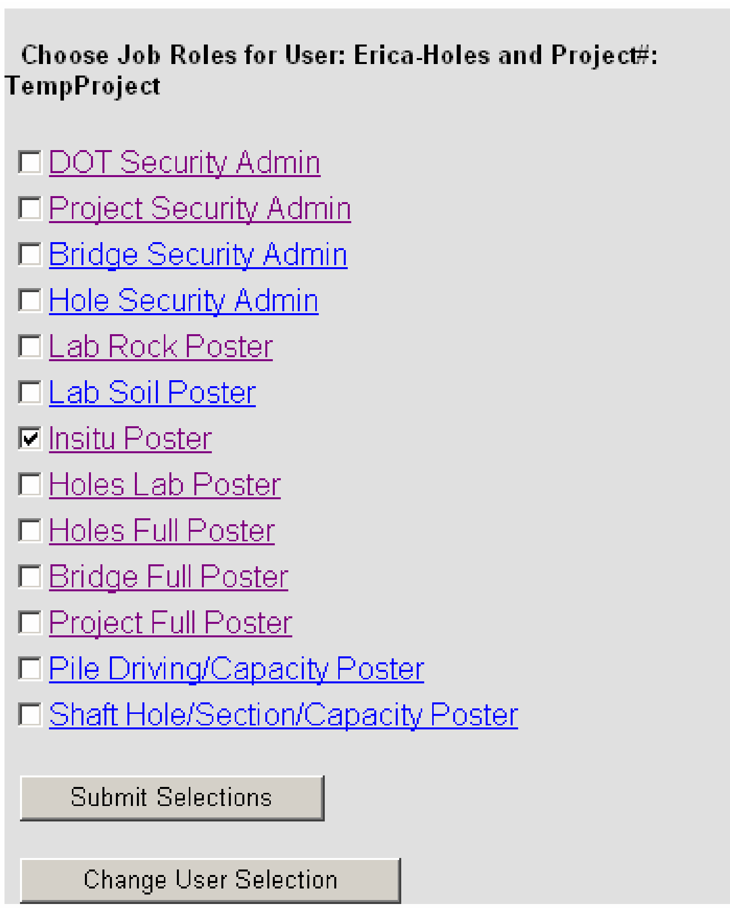


Figure 4.26 A User's Specific Data Project Access

A new project for a current user, bottom Fig. 4.25, will have the same project access information, Fig. 4.26, but have no job roles already checked/selected.

4.5.9 Log Out or Log In As Another User Account

Any User account can log out of the interface at any time. At the top of all forms/pages (except for the options panel, Fig. 4.3), the user has the following navigation links, Figure 4.27

Figure 4.27 Navigation Link on General User Account

Above all the links on DOT Admin Account is the following screen, Figure 4.28,

Figure 4.28 Navigation Link on DOT Admin Account

Once a user clicks the 'Log Out' link they are immediately logged out, their sessions are destroyed and they are directed to the log in page. The user may then initiate access as another user account, i.e. logging in, or disconnect Internet access altogether.

CHAPTER 5

LRFD CALIBRATION OF FB-DEEP

5.1 INTRODUCTION

SPT boring logs and pile load test data from projects within the state of Florida have been collected and uploaded into the new FDOT database. Each pile has a corresponding SPT log which was used in the FB-Deep program to predict the Davisson failure load. This prediction was compared to the measured value from the pile load test. These results were used to calibrate the LRFD resistance factors for Florida soils in the FB-Deep computer program.

The reduction of the data contained within the FDOT database is preceded by a detailed discussion of the Davisson failure limit, the FB-Deep prediction method, and the LRFD resistance factor reliability calibrations.

5.2 DAVISSON FAILURE CRITERIA

The Florida Department of Transportation specifies the Davisson capacity as the failure criteria for a pile. The Davisson capacity is reached when the axial movement of the top of the pile equals or exceeds one of the following values:

$$x = \frac{PL}{AE} + (0.15 + \frac{d}{120}) \qquad (d < 30") \qquad \text{Eq. 5.1}$$

$$x = \frac{PL}{AE} + (0.15 + \frac{d}{30}) \qquad (d \geq 30") \qquad \text{Eq.5.2}$$

where,

x = Axial displacement at the top of pile

d = Diameter or width of the pile

P = Load applied to the pile

L = Total length of the pile

A = Cross-sectional area of the pile

E = Elastic modulus of the pile

There are two components to the Davisson equations (Eq. 5-1, Eq. 5-2). The first component is the elastic shortening of the pile. The second component is a specified offset equal to the allowable movement of the pile within the soil mass.

A load test is required in order to measure the Davisson capacity of a pile. During the load test, the applied axial loads and total pile deformations at the top are plotted, i.e. load vs. settlement. Next, the elastic shortening of pile is plotted. It forms a straight line that initiates at zero and linearly increases with a slope of L/AE. The intersection of the load settlement curve and line parallel to elastic shortening line offset by 0.15 + D/125 or D/30 depending one pile size, corresponds to the Davison failure load or the pile capacity.

5.3 PREDICTING DAVISSON CAPACITY WITH FB-DEEP

FB-Deep is a computer program developed by the Florida Department of Transportation and maintained by the Florida Bridge Software Institute. FB-Deep is and outgrowth of SPT 89, SPT91 and SPT-97, with the inclusion of drilled shafts and recently the inclusion of large diameter pipe piles. For this research, FB-Deep was modified to interact with the FDOT database by reading and writing XML files using the specified

FDOT XML schema, Appendix B. Subsequently, FB-Deep was used to predict the Davison capacity of all the prestressed concrete piles in the database using SPT (Standard Penetration Test) boring informaiton in the database.

FB-Deep estimates the Davisson capacity as the total resistance provided by the skin friction, Table 5.1, plus one third of the resistance of the ultimate end bearing Table 5.2. The skin friction and end bearing are estimated using blow counts and soil types provided by the SPT boring log.

Table 5.1 FB-Deep Side Friction Equations

Soil Type	Description	Ultimate Unit Side Friction (TSF)
1	Plastic Clay	$f = \frac{2N(110-N)}{4006.6}$ Eq. 5.3
2	Clay-Silt-Sand mixtures, very silty sand, silts and marls	$f = \frac{2N(110-N)}{4583.3}$ Eq. 5.4
3	Clean Sands	$f = 0.019N$ Eq. 5.5
4	Soft limestone, very shelly sand	$f = 0.01N$ Eq. 5.6

*N is the uncorrected SPT blow count from a representative boring log for the pile.

The skin friction along the whole length of the pile is equal to:

$$Q_s = \sum f_N * length_N * perimeter \qquad \text{Eq. 5.7}$$

For Eq.5.7, unit skin friction, f_N, is calculated for each SPT N value from Table 5.1 for appropriate soil type.

An assumption made by the FB-Deep program is that the end bearing failure is controlled by the soil 3.5B below and 8B above, with B equal to the pile diameter or width. An exception to this assumption is when the bearing layer is weaker than the overlying layer. In this case it is assumed that the upper limit of the end bearing contributing soil is the boundary between the layers. N values should be interpolated at the tip, 3.5B below, and 8B.

Table 5.2 FB-Deep Mobilized End Bearing Equations

Soil Type	Description	Mobilized Unit End Bearing Capacity (TSF)
1	Plastic clay	$q = \frac{0.7N}{3}$ Eq. 5.8
2	Clay-silt-sand mixtures, very silty sand, silts and marls	$q = \frac{1.6N}{3}$ Eq. 5.9
3	Clean sands	$q = \frac{3.2N}{3}$ Eq. 5.10
4	Soft limestone, very shelly sand	$q = \frac{3.6N}{3}$ Eq. 5.11

The end bearing equation is more complicated than the side friction, but the same definition for layers applies. Layers change at N value elevations, and the average N values enclosing the layer controls the layer capacity. The previous equations divide the end bearing by 3 in order to approximate the resulting end bearing when the skin friction has been fully mobilized. The following equation is used to estimate the mobilized end bearing:

$$Q_t = \frac{\sum_{8B_{above}} f_{layer} * \frac{lenght_{layer}}{8B} + \sum_{3.5B_{below}} f_{layer} * \frac{length_{layer}}{3.5B}}{2} * A \qquad \text{Eq.5.12}$$

The end bearing is found by taking the weighted average above and below the tip. These values are then added together and divided by two. This result is a stress that is then multiplied by the tip cross-sectional area in order to estimate the mobilized end bearing resistance force in tons.

Special corrections have been developed to account for the concept of critical depth. Critical depth is based on the assumption that the pile tip must be embedded a certain depth within the bearing layer in order for the previously defined end bearing value to be fully realized. The critical depth ratio for each soil type can be found in Table 5.3.

Table 5.3 Critical Depth Ratios in FB-Deep

Soil Type	Description	Critical Depth Ratio (D/B)
1	Plastic Clay	2
2	Clay-silt-sand mixtures, very silty sand, silts and marls	4
3	Clean sands (N<=12)	6
3	Clean sands (12<N<=30)	9
3	Clean sands (N>=30)	12
4	Soft limestone, very shelly sands	6

The critical depth ratio is multiplied by the pile diameter or width in order to calculate the critical depth of embedment within the bearing layer.

The following equation is used to correct the end bearing:

$$q = q_{LC} + \frac{D_A}{D_C}(q_T - q_{LC}) \qquad \text{Eq. 5.13}$$

where,

q = Corrected unit end bearing at the pile tip

q_{LC} = Unit end bearing calculated at the layer change

q_T = Unit end bearing calculated at the pile tip

D_A = Actual embedment depth in bearing layer

D_C = Critical embedment depth in bearing layer

The skin friction within the bearing layer is also corrected with respect to the critical depth. The following equation is used to correct the skin friction within the embedment layer if the critical depth is not realized and the overlying layer is weaker:

$$CSFBL = \frac{SFBL}{q_T}\left(q_{LC} + \frac{D_A}{2D_C}(q_T - q_{LC})\right) \qquad \text{Eq. 5.14}$$

where,

CSFBL = Corrected side friction within the bearing layer

SFBL = Uncorrected side friction within the bearing layer

q_{LC}, q_T, D_A, D_C as previously defined

The skin fiction within the critical depth of the bearing layer is also reduced when the overlying layer is weaker and the critical depth is reached. In other words, if the overlying layer is weaker, the skin friction within the critical depth will always be reduced. This corrected value must be added to the value calculated using the length of the pile beyond the critical depth. In this case, the corrected skin friction within the critical depth is calculated with the following equation:

$$CSFACD = \frac{USFACD}{q_{CD}}(q_{LC} + 0.5(q_{CD} - q_{LC})) \qquad \text{Eq. 5.15}$$

where,

CSFACD = Corrected side friction within the critical depth

USFACD = Uncorrected side friction from the top of the bearing layer to the critical depth

q_{CD} = Unit end bearing calculated at the critical depth

q_{LC} as previously defined

FB-Deep estimates the Davisson capacity as the sum of the skin friction above the bearing layer, the corrected skin friction within the bearing layer, and the corrected mobilized end bearing.

5.4 LRFD

The probability of failure can never equal zero when any amount of uncertainty exists within the scope of the engineered solution. Load and resistance factor design (LRFD) was developed to account for the uncertainty within the design process. Previous design codes increased the resistance capacity of a design by a factor of safety in an attempt to account for the uncertainty. Engineering judgment was used to account for the quantity and quality of the available geotechnical data, the applied design method, and relevant design codes, in order to derive an appropriate factor of safety.

Load and resistance factors used with the LRFD design codes correspond to a specific probability of failure. As previously stated, the probability of failure will never be 0% when any amount of uncertainty exists. Uncertainty will always exist in a geotechnical design.

The principal LRFD equation is as follows:

$$\Phi R_n \geq \eta \sum \gamma_i Q_i \qquad \text{Eq.5.16}$$

where,

Φ = Resistance factor

R_n = Nominal resistance

η = Importance, redundancy, and ductility modifier. Typically 0.95 to 1.

γ_i = Load factor

Q_i = Load

The LRFD resistance factor can be calibrated using reliability theory with a dataset consisting of measured and predicted values. More applicable resistance factors can be realized by creating subsets of the data that contain certain similarities. For example, an LRFD resistance factor is more applicable when derived with a consistent prediction method. This is due to the varying amounts of conservatism that exist among the multitudes of available prediction methods. Another way to improve the quality of the resistance factor is to limit the subset to a specific geological area. LRFD factors developed within the state of Florida will be more applicable than ones developed using data from all over the globe.

The resistance factor can be derived from a dataset using the following equation:

$$\Phi = \frac{\lambda_R \left(\gamma_D \frac{Q_D}{Q_L} + \gamma_L \right) \sqrt{\frac{1 + COV_{QD}^2 + COV_{QL}^2}{1 + COV_R^2}}}{\left(\lambda_{QD} \frac{Q_D}{Q_L} + \lambda_{QL} \right) e^{\beta_T \sqrt{\ln\left[\left(1+COV_R^2\right)\left(1+COV_{QD}^2+COV_{QL}^2\right)\right]}}} \qquad \text{Eq. 5.17}$$

where,

Φ = Resistance factor

λ_R = Resistance method bias

γ_D = Dead load factor (1.25)

Q_D/Q_L = Dead to live load ratio

γ_L = Live load factor (1.75)

COV_{QD} = Coefficient of variation of the dead load

COV_{QL} = Coefficient of variation of the live load

COV_R = Coefficient of variation of the method bias

λ_{QD} = Dead load bias factor

λ_{QL} = Live load bias factor

β_T = Target reliability index

- λ_R, the resistance method bias, is found by averaging the bias of each set of measured and predicted data. The bias is calculated as follows:

$$\lambda = \frac{measured}{predicted} \qquad \text{Eq. 5.18}$$

and,

$$\lambda_R = \frac{\sum \frac{measured}{predicted}}{N} \qquad \text{Eq. 5.19}$$

where,

N = number of items within the dataset

A bias of 2 would mean that the measured Davisson load was twice the predicted amount.

- γ_D, the dead load factor, is based on the unknowns that might be associated with the applied dead load. For this analysis it will be set to an assumed value of 1.25.
- Q_D/Q_L, the dead to live load ratio, is a variable within this study. The resulting resistance factor will be a function of this ratio.
- γ_L, the live load factor, is based on unknowns associated with the applied live load. For this analysis it will be set to an assumed value of 1.75.
- COV_{QD}, the coefficient of variation of the dead load, defines the statistical variation of the applied dead load from the mean. A coefficient of variation is equated as:

$$COV = \frac{\sigma}{m} \qquad \text{Eq. 5.20}$$

where,

σ = the standard deviation

m = the mean

In this study COV_{QD} will be set to an assumed value of 0.14.

- COV_{QL}, the coefficient of variation of the live load, will be set to an assumed value of 0.18.
- COV_R, the coefficient of variation of the method bias, will be calculated from the available data within the database. A bias will be calculated for each measured

and predicted pair within the dataset. COV_R will equal the standard deviation of the biases divided by λ_R, the resistance method bias.

- λ_{QD}, the dead load bias factor, will be set to an assumed value of 1.03. Similar to the resistance bias factor, it is equal to the measured weight divided by the predicted weight of the dead load.
- λ_{QL}, the live load bias factor, will be set to an assumed value of 1.15.
- β_T, the target reliability index, is a variable within the resistance factor equation. The resistance factor will be a function of the target reliability index. The reliability index can be used to approximate the probability of failure. The derivation of the reliability index assumes that the applied load (Q) and the resulting resistance (R) can be approximated as gaussian random variables. The probability density function for a gaussian random variable is defined as:

$$f_X(x) = \frac{e^{\frac{-(x-m)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \qquad \text{Eq. 5.21}$$

where,

m = statistical mean

σ = statistical standard deviation

The gaussian random variable represents a probability distribution. Integrating between two points on the curve yields the probability of a random value falling between those limits. As such, the total area under the curve must equal 1.

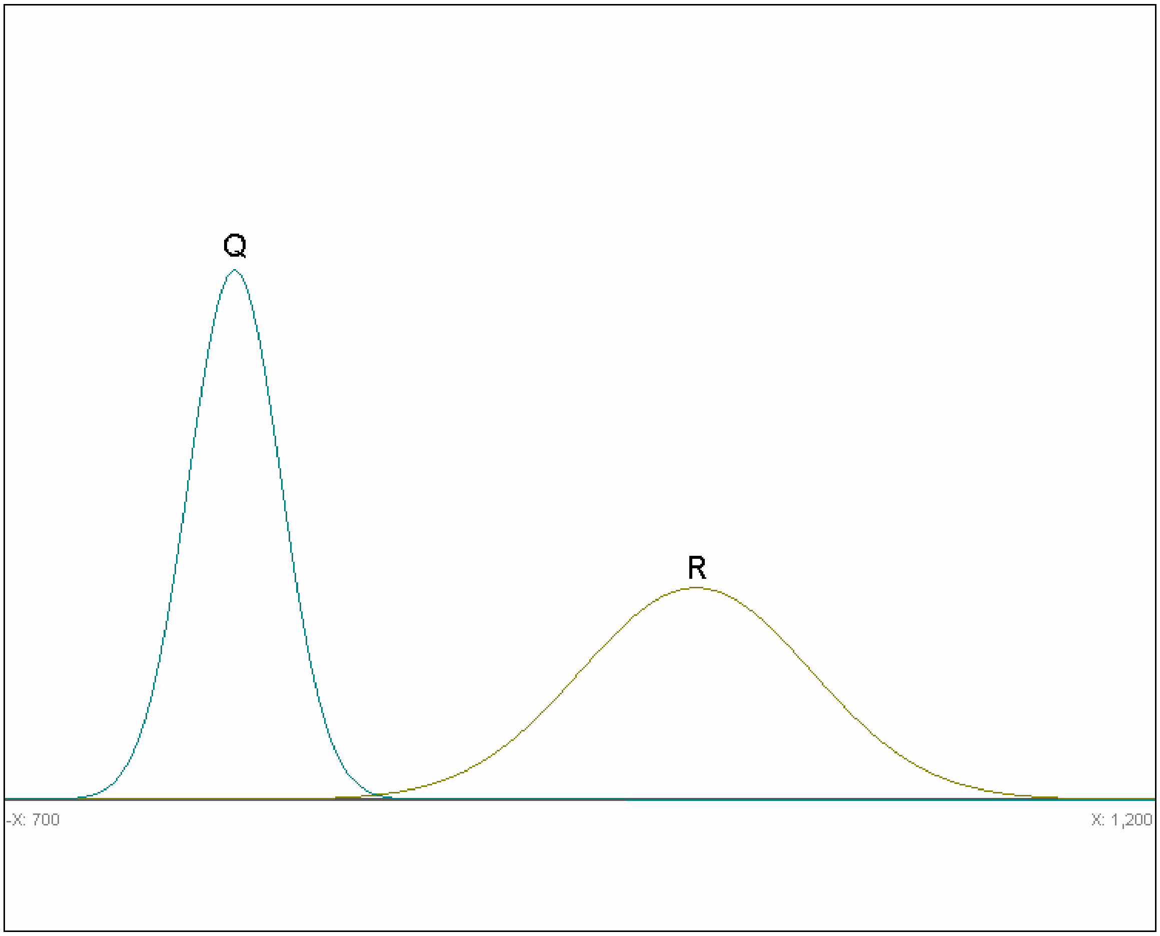


Figure 5.1 Probability density functions for normally distributed load and resistance

A typical plotting of load (m=800, σ=20) and resistance (m=1000, σ=50) gaussian random variables is shown in Figure 5.1. In this system the factor of safety equals 1000/800 = 1.25. However, failure will occur when the load is larger than the resistance. The overlap of the load and resistance curves in Figure 5.1 shows the potential for failure within the design. However, the probability of failure is not equal to the area under the intersection of the Q and R gaussian curves. It is equal to the probability that the resistance is less than the load. In order to evaluate the probability that the resistance is less than the load, a normally distributed function of the two random variables is developed as follows:

$$G(R,Q) = R - Q \qquad \text{Eq. 5. 22}$$

A lognormal distribution of this function would be:

$$g(R,Q) = \ln(G(R,Q)) = \ln(R) - \ln(Q) = \ln\left(\frac{R}{Q}\right) \qquad \text{Eq. 5.23}$$

According to Eq. 5.23, failure will occur when g(R,Q) < 0. This function is plotted below in Figure 5.2.

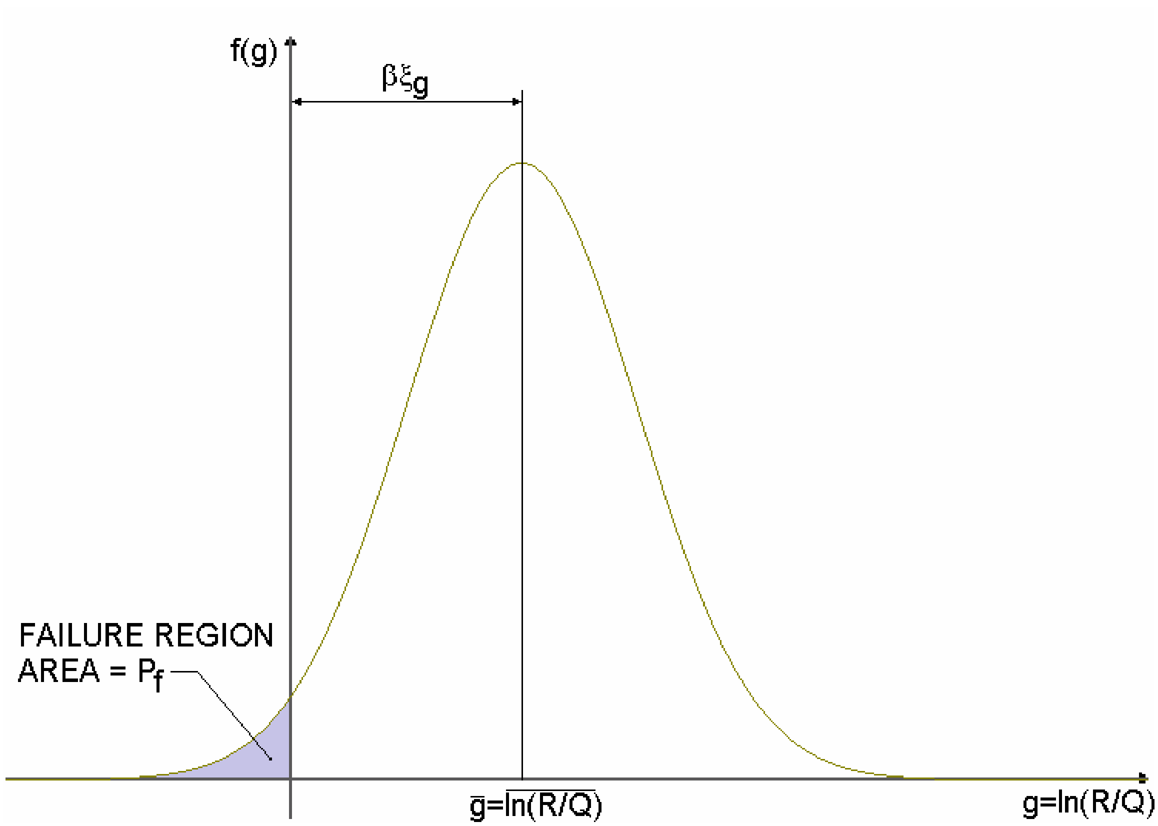


Figure 5.2 Plot of Eq. 5.23

The reliability index equals the ratio of the lognormal mean of g(R,Q), and the lognormal standard deviation ξ_g. As the lognormal standard deviation increases, the curve flattens and the failure region area increases. As the lognormal mean decreases the curve shifts to the left and the failure probability region increases. As previously mentioned, β_T, the target reliability index, is a variable within the resistance factor equation. The resistance factor will be a function of this reliability index.

Rosenblueth and Esteva, 1972, developed the following approximation of the probability of failure:

$$p_f = 460e^{-4.3\beta} \qquad \text{Eq. 5.24}$$

Where β is the reliability index. This equation is only applicable when the reliability index is between 3 and 5. Baecher as referenced in NCHRP Report 507 (2004), calculated "exact" probabilities of failure for various reliability indexes. Table 5.4 approximates the solution to the Rosenbleuth and Estevas equation and provides the corresponding Baecher solution for a given reliability index.

Table 5.4 Approximate probability of failure given the reliability index

β	Rosenbleuth and Estevas' p_f	Baecher's p_f	Percent Error (%)
2	8.469E-02	2.275E-02	272.3
2.5	9.865E-03	6.210E-03	58.9
3	1.149E-03	1.350E-03	-14.9
3.5	1.339E-04	2.327E-04	-42.5
4	1.559E-05	3.169E-05	-50.8
4.5	1.816E-06	3.401E-06	-46.6
5	2.116E-07	2.871E-07	-26.3
5.5	2.464E-08	1.904E-08	29.5
6	2.871E-09	9.901E-10	189.9

5.5 FLORIDA PRESTRESSED CONCRETE PILES

All of the driven prestressed concrete piles in the earlier FDOT Microsoft Access Deep Foundation Database were entered into the new Internet Based FDOT Deep Foundation Database (Chapter 2) through the Excel "Load Test" spreadsheets (Chapter 3). Insitu SPT data within the earlier database as well other available SPT data from old Geotechnical Reports, Plans, etc. was uploaded into the new FDOT Deep Foundation Database through the Excel "Insitu" spreadsheets (Chapter 3).

A total of 56 prestressed concrete piles, Table 5.5 which had both top down static load tests and reached Davisson Failure Capacity were considered for the LRFD study. Also shown in Table 5.5 are the main soil types around each pile.

Next the FB-Deep software was modified to read and write the XML schema tags (Appendix B), and to upload and download the Insitu and Laboratory Data (i.e. piles and shaft design) with the DLL program discussed in Chapter 4. The latter handles security (i.e. username, password), as well parsing the data between the software and the database. Subsequently, the soil boring SPT N values were downloaded, assigned FB-Deep soil descriptors (i.e. 1,2,3, etc – Table 5.1), analyzed, and the estimate Davisson capacities and soil descriptors were uploaded back into the database.

Shown in Figure 5.3 is a plot of predicted Davisson capacities from FB-Deep vs. measured Davisson capacities from load tests. Also to identify the match for various soil types, i.e. Table 5.1, three subsets were created: 1) piles with more than 75% cohesive material along their length; 2) piles with more than 75% cohesionless material along their length, and 3) piles with less than 75% cohesive and less than 75% cohesionless. Plots of each measured vs. predicted pile capacities for each major soil descriptions are shown in

Figures 5.4, 5.5, and 5.6. Evident from the plots, each of the predominate soil types does a good job of predicting Davisson, Capacities; however, the mixed soil, Figure 5.6 shows that FB-Deep is conservative, i.e. under predicts capacities

Using the predicted Davisson capacities (Table 5.5) with the measured Davisson capacities (Table 5.5), the load test bias, Eq. 5.18 was found, for each test. Subsequently, LRFD resistance factors for whole database were found as well for the 3 major soil descriptors described. A description of the assessment follows.

Table 5.5 Measured versus predicted pile capacities

Project	Pile	Prediction FB-Deep Tons	Measured Davisson Tons	Bias	Cohesionless (%)	Cohesive (%)
Acosta - 72160-3506	F6	476.51	338	0.709	29	71
Acosta - 72160-3506	G13	284.98	558	1.958	88	12
Acosta - 72160-3506	H2	97.71	280	2.866	17	83
Apalachicola Bay - 49010-3536	TS-5	145.07	262	1.806	16	84
Apalachicola Bay - 49010-3536	TS-16	136.94	406	2.965	16	84
Apalachicola Bay - 49010-3536	TS-23	412.72	404	0.979	18	82
Apalachicola Bay - 49010-3536	TS-27	413.75	490	1.184	44	56
SR20 Blountstown - 47010-3519	20	307.14	825	2.686	65	35
SR20 Blountstown - 47010-3519	21A	699.62	550	0.786	12	88
SR20 Blountstown - 47010-3519	22	532.32	600	1.127	17	83
Apalachicola River - 49010-3533	14	345.51	425	1.230	72	28
Apalachicola River - 49010-3533	25	121.77	290	2.382	55	45
Apalachicola River - 49010-3533	3	293.85	385	1.310	87	13
Blackwater - 58002-3449	14L	509.32	300	0.589	90	10
Blackwater - 58002-3449	20L	506.91	420	0.829	100	0
Buckman - 72001-3462	TS-13	411.36	490	1.191	32	68
Buckman - 72001-3462	TS-19	495.19	530	1.070	58	42
Buckman - 72001-3462	TS-24	354.38	555	1.566	62	38
Buckman - 72001-3462	TS-29	421.54	530	1.257	45	55
Choctawhatchee - 60040-3527	P-5	390.8	612	1.566	87	13
Choctawhatchee - 60040-3527	P-11	482.94	715	1.481	71	29
Choctawhatchee - 60040-3527	P-17	311.17	756	2.430	65	35
Choctawhatchee - 60040-3527	P-23	329.55	330	1.001	71	29
Choctawhatchee - 60040-3527	P-29	453.77	458	1.009	68	32
Choctawhatchee - 60040-3527	P-35	350.02	729	2.083	73	27
Choctawhatchee - 60040-3527	P-41	455.94	703	1.542	41	59
Choctawhatchee - 60040-3527	FSB-26	295.86	480	1.622	74	26
Choctawhatchee - 60040-3527	FSB-3	100.86	249	2.469	82	18

Dodge Island - 87000-3675	6	601.2	620	1.031	0	100
Howard Franklin - 15190-3446	S-3	353.39	920	2.603	61	39
Howard Franklin - 15190-3446	S-4 lng	683.48	430	0.629	80	20
Howard Franklin - 15190-3446	S-1	344.53	510	1.480	81	19
Port Orange - 79180-3502	Bent 2	132.76	139	1.047	100	0
Port Orange - 79180-3502	Bent 1	94.91	101.5	1.069	100	0
Vilano - 78030-3546	TS-1	378.53	553	1.461	89	11
Vilano - 78030-3546	TS-2	349.05	696	1.994	81	19
Sunshine Skyway Site 10	p13	335.45	480	1.431	14	86
Sunshine Skyway Site 13A	p18	148.78	300	2.016	95	5
Sunshine Skyway Site 1A	20	304.56	275	0.903	0	100
Sunshine Skyway Site 1B	24	394.68	376	0.953	0	100
Sunshine Skyway Site 3	p6	325.79	483	1.483	27	73
Escambia River 48140-3509	5	264.28	425	1.608	100	0
White City Bridge - 51020-3514	TP3	106.38	315	2.961	44	56
White City Bridge - 51020-3514	TP6	128.61	230	1.788	21	79
Edison Bridge - 12001-3513	1a	316.5	500	1.580	84	16
Edison Bridge - 12001-3514	2a	298.85	285	0.954	91	9
Edison Bridge - 12001-3515	2b	261.4	242	0.926	90	10
Edison Bridge - 12001-3516	2c	178.49	117	0.655	90	10
Edison Bridge - 12001-3517	3a	274.53	305	1.111	74	26
Edison Bridge - 12001-3518	3b	216.96	280	1.291	72	28
Edison Bridge - 12001-3519	3c	120.21	110	0.915	66	34
Edison Bridge - 12001-3520	4a	389.97	475	1.218	77	23
Edison Bridge - 12001-3521	4b	245.35	215	0.876	74	26
Edison Bridge - 12001-3522	4c	98.55	140	1.421	68	32
Edison Bridge - 12001-3523	5a	417.59	495	1.185	93	7
Matanzas River - 72002-3509	14	294.09	535	1.819	47	53

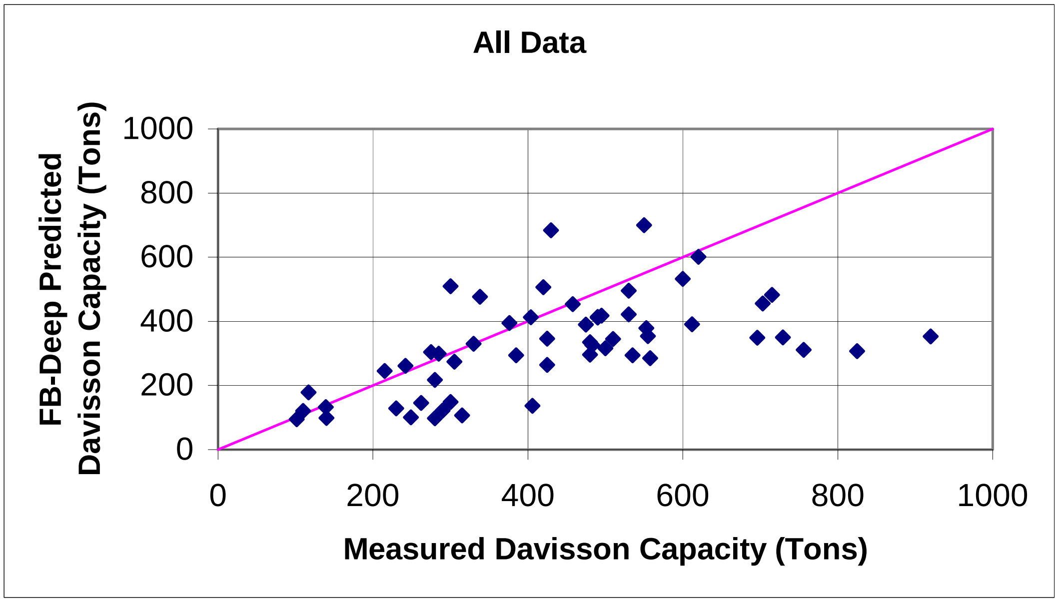


Figure 5.3 Comparison of measured versus predicted capacities, All Soils

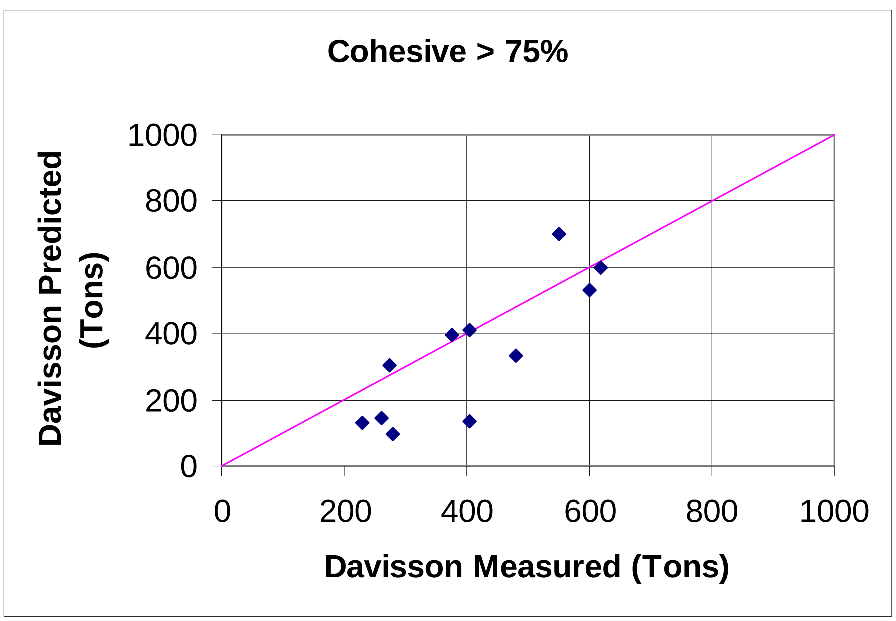


Figure 5.4 Measured versus predicted capacities, More than 75% Cohesive

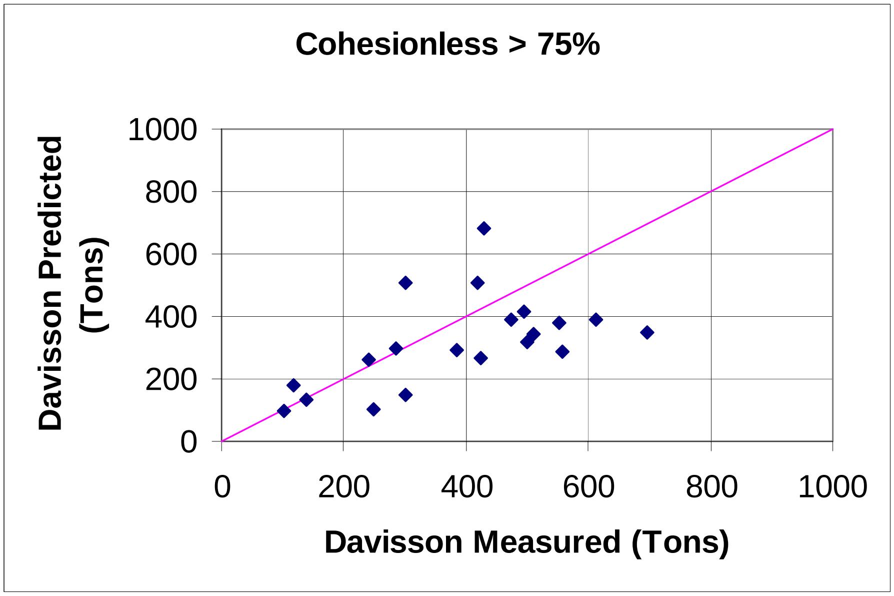


Figure 5.5 Measured versus predicted capacities, More than 75% Cohesionless

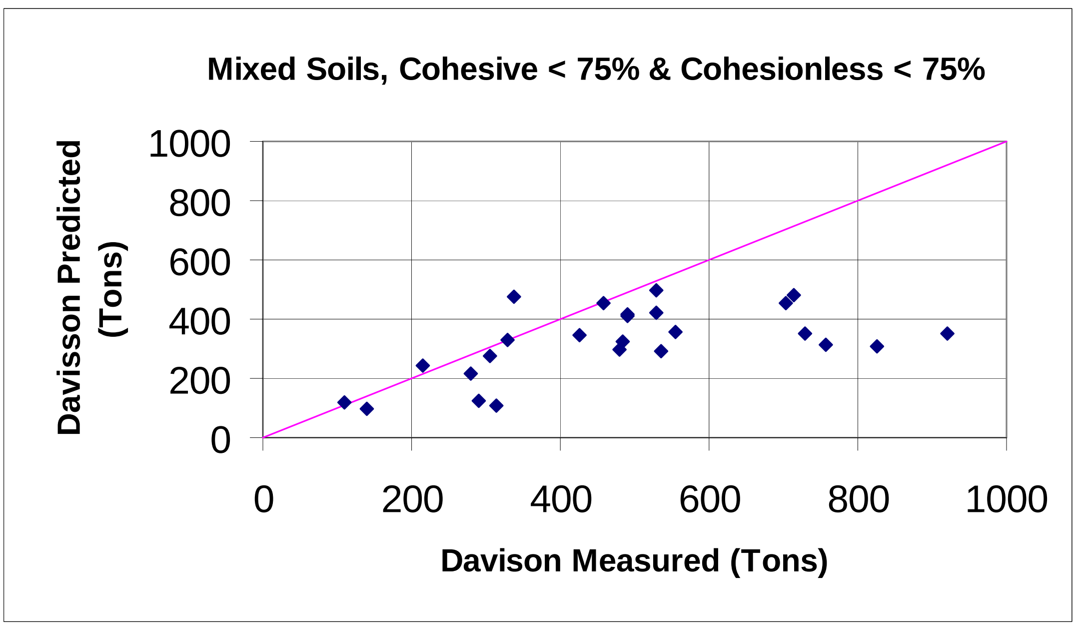


Figure 5.6 Measured vs. Predicted Capacities, Mixed Soils, Neither Cohesive nor Cohesionless greater than 75%

5.6 LRFD DATA ANALYSIS

The average bias, λ_R of all the data in Table 5.5 was 1.53. The standard deviation of the data was 0.662. This yields a calculated variance of 0.433. The resistance factor can be calculated with the following equation:

$$\Phi = \frac{\lambda_R\left(\gamma_D \dfrac{Q_D}{Q_L} + \gamma_L\right)\sqrt{\dfrac{1 + COV_{QD}^2 + COV_{QL}^2}{1 + COV_R^2}}}{\left(\lambda_{QD}\dfrac{Q_D}{Q_L} + \lambda_{QL}\right)e^{\beta_T\sqrt{\ln\left[\left(1+COV_R^2\right)\left(1+COV_{QD}^2+COV_{QL}^2\right)\right]}}} \qquad \text{Eq. 5.25}$$

where,

Φ = Resistance factor

λ_R = Resistance method bias = 1.47

γ_D = Dead load factor = 1.25

Q_D/Q_L = Dead to live load ratio

γ_L = Live load factor = 1.75

COV_{QD} = Coefficient of variation of the dead load = 0.14

COV_{QL} = Coefficient of variation of the live load = 0.18

COV_R = Coefficient of variation of the method bias = 0.422

λ_{QD} = Dead load bias factor = 1.03

λ_{QL} = Live load bias factor = 1.15

β_T = Target reliability index

Shown in Table 5.6 are the predicted LRD resistance factors, ϕ, for the whole database, using Eq. 5.25 for different dead load to live load ratios (Q_D/Q_L), and reliability, β_T, values

Table 5.6 Computed Resistance Factors (Φ)

β	QD / QL										
	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
2	0.74	0.72	0.70	0.70	0.69	0.68	0.68	0.67	0.67	0.67	0.67
2.25	0.66	0.64	0.63	0.62	0.61	0.61	0.60	0.60	0.60	0.59	0.59
2.5	0.59	0.57	0.56	0.55	0.55	0.54	0.54	0.53	0.53	0.53	0.53
2.75	0.52	0.51	0.50	0.49	0.49	0.48	0.48	0.48	0.47	0.47	0.47
3	0.46	0.45	0.44	0.44	0.43	0.43	0.43	0.42	0.42	0.42	0.42
3.5	0.37	0.36	0.35	0.35	0.34	0.34	0.34	0.34	0.33	0.33	0.33
4	0.29	0.28	0.28	0.28	0.27	0.27	0.27	0.27	0.27	0.26	0.26
4.5	0.23	0.23	0.22	0.22	0.22	0.21	0.21	0.21	0.21	0.21	0.21
5	0.18	0.18	0.18	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17

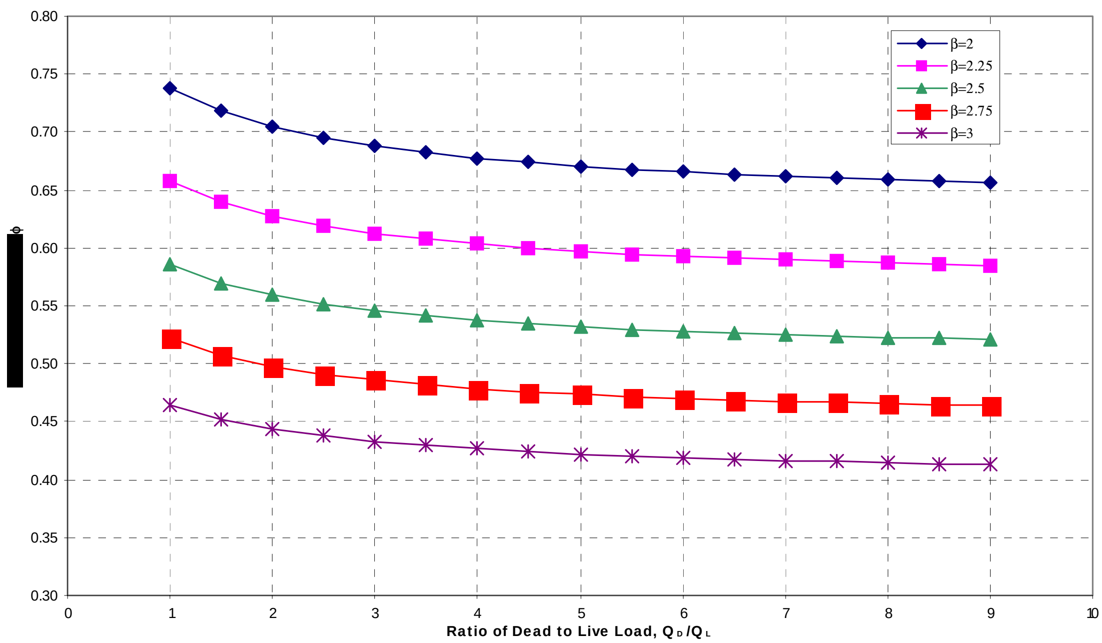


Figure 5.7 FB-Deep Reliability Based Calibration of LRFD Resistance Factor, All Data

Shown in Figure 5.7 are plots of LRFD resistance factors,ϕ, for different Q_D / Q_L for the various reliability values, β, given in Table 5.6.

Table 5.7 Analysis of Mean and COV Based on Percentages of Cohesive Soil

Cohesive (%)	λ_R	Std	COV
55	1.512	0.748	0.494
60	1.430	0.715	0.500
65	1.430	0.715	0.500
70	1.448	0.740	0.511
75	1.512	0.774	0.512
80	1.485	0.810	0.545
85	1.021	0.246	0.241

Table 5.7 provides the method bias, standard deviation, and covariance for the various minimum percentage of cohesive material. For example, the first row corresponds to the subset of data in which a minimum 55% of the material was cohesive. The 75% minimum cohesive plot is provided in Figure 5.4. Table 5.8 is similar to Table 5.7, only with respect to the minimum percentage of cohesionless material. The data plotted at a minimum of 75% is given in Figure 5.5.

Table 5.8 Analysis of Mean and COV Based on Percentages of Cohesionless Soil

Cohesionless (%)	λ_R	Std	COV
55	1.415	0.559	0.395
60	1.425	0.564	0.396
65	1.309	0.462	0.353
70	1.330	0.476	0.358
75	1.327	0.517	0.389
80	1.372	0.517	0.376
85	1.227	0.447	0.364

Table 5.9 Analysis of Mean and COV Based on Mixed Soil Percentages

Mixed (%)	λ_R	Std	COV
55	1.819	0.562	0.309
60	1.745	0.700	0.401
65	1.955	0.677	0.346
70	1.736	0.692	0.399
75	1.557	0.624	0.401
80	1.520	0.619	0.407
85	1.617	0.657	0.406

Table 5.9 presents the analysis of mean COV for mixed soil percentages. Table 5.9 needs to be interpreted slightly different than Table 5.8 and 5.7. In Table 5.9, the percentage refers to the maximum percent of cohesive or cohesionless material. For example, the 60% row corresponds to the subset of data in which neither the cohesive nor cohesionless material accounts for more than 60% of the soil along the length of the pile.

Tables 5.10, 5.11, and 5.12 give the calculated resistance factors (Φ) for target reliability indexes (β) of 2, 2.5, and 3 respectively. The percentages refer to the minimum amount of soil. The value under the cohesive column at 55% corresponds to a dataset containing all piles in which at least 55% of the soil along the pile was cohesive. The percentage on the mixed column corresponds to the dataset at which neither soil type exceeds the given percentage.

Table 5.10 Resistance factors, Φ, corresponding to a reliability index (β) of 2

β = 2 (%)	Cohesive	Cohesionless	Mixed
55	0.608	0.688	1.036
60	0.569	0.691	0.838
65	0.569	0.689	1.040
70	0.563	0.693	0.838
75	0.588	0.652	0.749
80	0.541	0.690	0.722
85	0.653	0.631	0.769

Table 5.11 Resistance factors, Φ, corresponds to a reliability index (β) of 2.5

β = 2.5	Cohesive	Cohesionless	Mixed
55	0.469	0.551	0.858
60	0.438	0.554	0.671
65	0.438	0.561	0.850
70	0.432	0.563	0.671
75	0.450	0.524	0.599
80	0.409	0.557	0.576
85	0.554	0.512	0.614

Table 5.12 Resistance factors, Φ, corresponds to a reliability index (β) of 3

β = 3	Cohesive	Cohesionless	Mixed
55	0.362	0.442	0.711
60	0.337	0.444	0.536
65	0.337	0.457	0.694
70	0.331	0.458	0.537
75	0.345	0.421	0.479
80	0.309	0.450	0.460
85	0.471	0.415	0.490

Presented in Figure 5.8 are FB-Deep Reliability Based Calibration of LRFD Resistance Factors, ϕ, for cohesive Soils greater than 75% (Tables 5.10 – 5.12) for different reliability, β, and bridge dead load to live load ratios (i.e. Q_D / Q_L). Similarly Figure 5.8 is FB-Deep Resistance Factors, ϕ, for piles with 75% or greater cohesionless material along its length. Figure 5.9 is FB-Deep Reliability Based Calibration of LRFD Resistance Factor, ϕ, for soils with neither 75% nor greater Cohesive nor Cohesionless Soils along their length. Evident due to the mixed soil conservative nature, i.e. Figure 5.6, they results in the highest LRFD resistance factors.

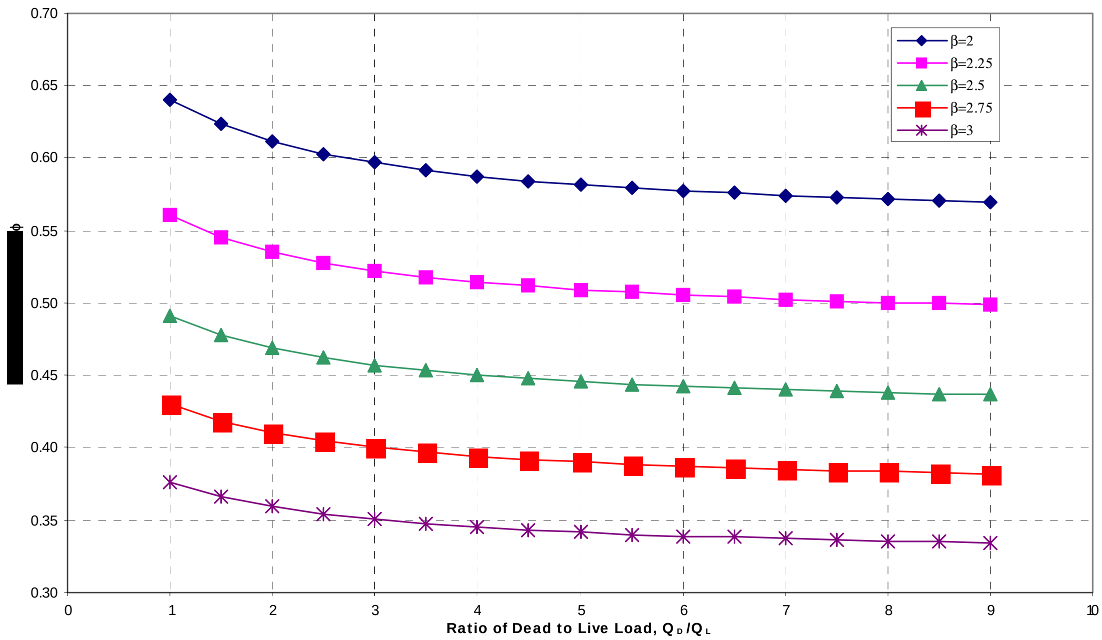


Figure 5.8 FB-Deep Reliability Based Calibration of LRFD Resistance Factor, Cohesive Soils greater than 75%

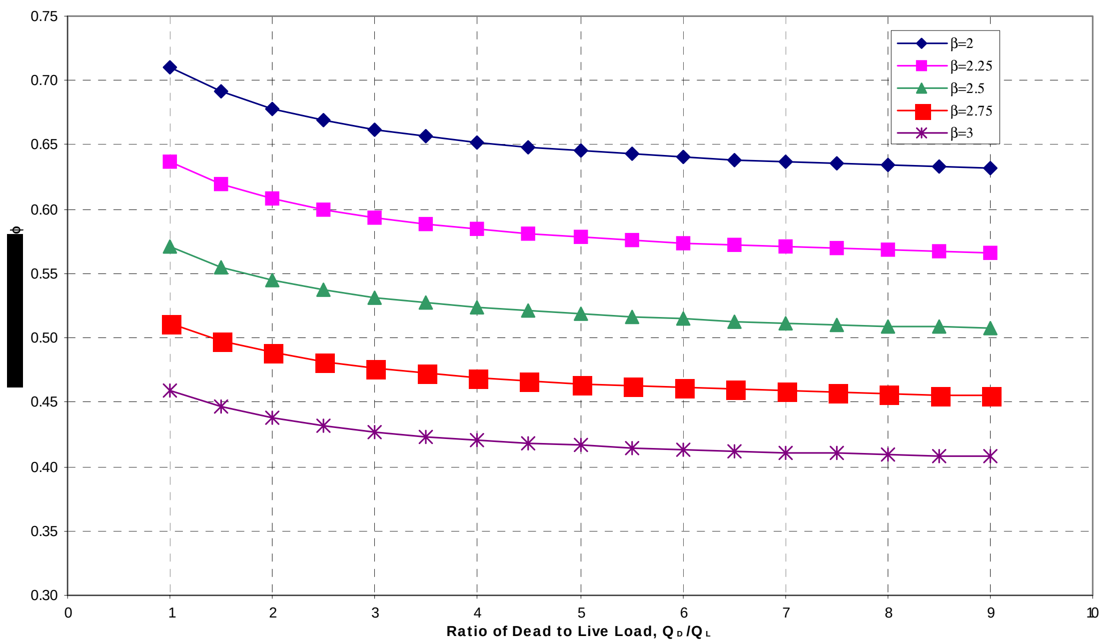


Figure 5.9 FB-Deep Reliability Based Calibration of LRFD Resistance Factor, Cohesionless Soils greater than 75%

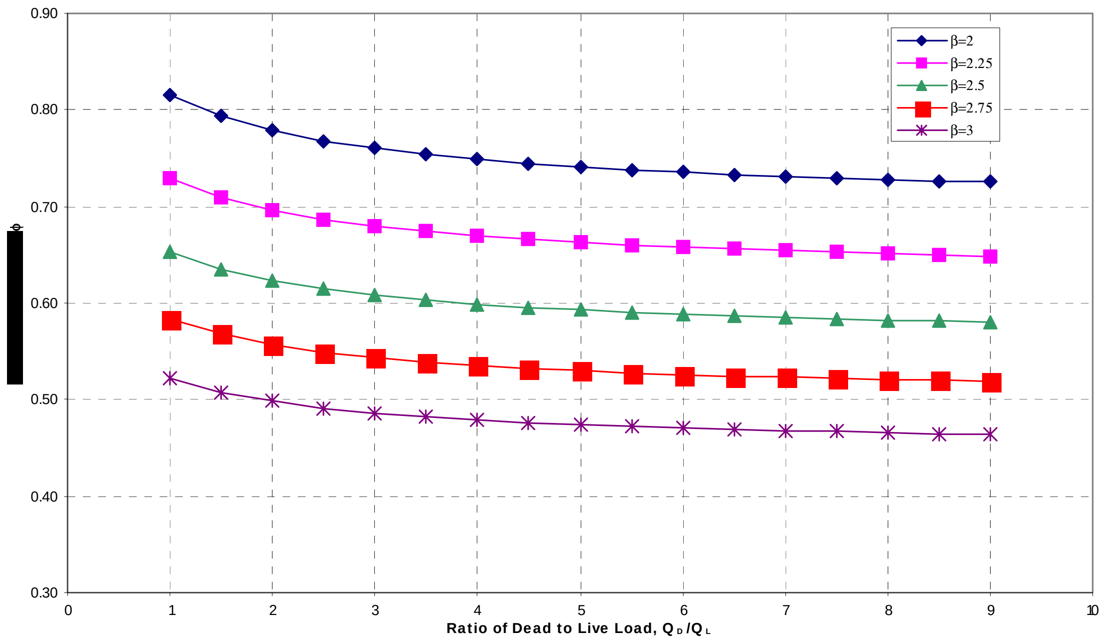


Figure 5.10 FB-Deep Reliability Based Calibration of LRFD Resistance Factor, Neither Cohesive nor Cohesionless Soils greater than 75%

5.7 LRFD PRESTRESSED CONCRETE PILE CONCLUSIONS

The calibration of LRFD resistance factors in this study was based on the Davisson failure limit and the FB-Deep computer program. The dataset used in this calibration has been uploaded into the FDOT database. Resistance factors can be calibrated to a factor of safety using the following equation:

$$FS = \frac{\gamma_d \frac{Q_D}{Q_L} + \gamma_L}{\phi\left(\frac{Q_D}{Q_L} + 1\right)} \qquad \text{Eq. 5.26}$$

This equation can be used to back calculate an equivalent factor of safety for a given resistance factor. Solving this can demonstrate that the design developed using LRFD does not deviate significantly from previously developed designs applying ASD.

However, LRFD is not equivalent to ASD since an approximate probability of failure can be defined for a given design. Eq. 5.26 has been solved with a dead to live load ratio of 4 for various references in Table 5.13.

Table 5.13 Comparison of Resistance Factors

Reference	β=2		β=2.5		β=3	
	φ	FS	φ	FS	φ	FS
FB-Deep - All Soils	0.68	1.99	0.54	2.50	0.43	3.14
FB-Deep - Cohesive Soils	0.59	2.29	0.45	3.00	0.34	3.97
FB-Deep - Cohesionless Soils	0.65	2.08	0.52	2.60	0.42	3.21
FB-Deep - Mixed Soils	0.75	1.80	0.60	2.25	0.48	2.81
Calibrating Resistance Factors in the Load and Resistance Factor Design for Florida Foundations - SPT94	0.78	1.73	0.66	2.04	0.57	2.37
Singletary, William - SPT94	0.71	1.90	0.67	2.01	0.48	2.81
Thai, Nguyen - SPT97	0.55	2.45	0.43	3.14		
	φ			FS		
AASHTO SPT (1994)	0.45			3.00		

As expected, a resistance factor derived from a reliability based calibration of FB-Deep with Florida projects improves upon the AASHTO recommendations. The resistance factors derived in this study are comparable to similar calibrations preformed in previous studies.

It is recommended that the resistance factors be periodically recalibrated as the FDOT database grows during its implementation. The XML schema provides a means to programmatically interact with the database to achieve this.

CHAPTER 6

SUMMARY AND CONCLUSIONS

6.1 BACKGROUND

A number of State DOTs (Virginia, Ohio, Maryland, California, Minnesota, etc.) have established or are in the process of establishing Internet based Geotechnical Databases. Some include SPT Boring data with laboratory data [VDOT: GDBMS (Geotechnical Database Management System)] others are developing virtual data center interfaces, i.e., links (URLs) to multiple databases, [Consortium of Organizations for Strong-Motion Observation Systems (COSMOS,2004) and by the Pacific Earthquake Engineering Research Center Lifelines Program (PEER)]. Since most users are spread throughout a state or country, access to the databases are generally through the Internet. Standard Languages to display and/or describe data is XML (eXtensible Markup Language). Existing and ongoing efforts are focused on the development of data dictionaries or format standards for the XML interfaces between software (e.g., gINT, etc.) and backend databases (Oracle, SQL, etc.) for transportation applications. Examples of the data dictionaries are TransXML (NCHRP 20-64), and AGS-XML (2004). Unfortunately, none of the current XML schemas or data dictionaries describe Bridge Foundation Design, or As Built information.

Within Florida, the current design, and construction data for DOT bridge foundations are archived in formats, which may not be queried or mined. For instance, in construction, AASHTOWare's SiteManager (*2004*) stores much of the data in PDF or Tiff formats. The latter is a result of field practices, such as 1) recording construction processes on paper, e.g., pile-driving records (blow counts vs. penetration, pile lengths, etc.); and/or 2) sub contractors performing site work (e.g., pile load testing: Statnamic, Osterberg) with deliverables (i.e., final report) in paper format. However, with the development of PDAs and/or field laptops, the electronic capture of pile driving records has now begun (e.g., FDOT Pile Technician Software). In addition, most if not all current laboratory data (e.g., soil classification, rock strength, etc.) and/or field load testing (e.g., Statnamic, Osterberg, etc.) is collected or reduced with spreadsheets (e.g., Microsoft's Excel).

Given the standardization of transportation data [i.e., TransXML(*2004*): 1) Survey/Roadway Design, 2) Transportation Construction and Materials, 3) Highway Bridge Structures, and 4) Transportation Safety], the data which is captured and analyzed on spreadsheets (i.e., load tests, lab data, etc.), borings (e.g., Autocad, gINT, etc.) and in construction (e.g., Pile Technician software), an Internet based database dealing with Bridge Substructures, is now viable. The need for this completely integrated database is important, since: 1) information collected in the field (e.g., borings) and lab (e.g., classification, strength, scour ability, etc.) may be used by multiple consultants in design (e.g., capacity: FB-Deep, PL-Aid, scour: Hec 18 & Hec 20, etc.); 2) it will provide quality control and assessment during construction; 3) supply input for maintenance, rehab, and scour software (e.g., AASHTOware's Pontiss, FHWA's Hyrisk, etc.); 4) make available data for updating LRFD resistance factors for piles, shafts, etc.; and finally 5) provide location for storing information on substructure modification (i.e., bridge widening, repairs, etc.). It is envisioned that the database would follow the bridge substructure from "birth" (design, construction), and through its useful "life."

A web based database and associated DLL software for security, XML schema for data transfer, as well as Excel spreadsheets for uploading and downloading of the data has been developed. To demonstrate the use of the database, LRFD assessment of Resistance Factors, ϕ, for prestressed concrete piles with the FB-Deep software was carried out. A discussion of each follows.

6.2 DATABASE STRUCTURE

Shown in Figure 6.1 is the layout of the current FDOT's Bridge Substructure Relational Database. The database employs a tree structure with information stored in tables with various hierarchy and interdependence. The Table with the highest hierarchy is the project ID table, Fig. 6.1, identified by any of the following: 1) project financial number, 2) project name, or 3) project State Job Number. Associated with the Project Tables are 1) bridges, 2) Insitu data, 3) Laboratory data, and 4) Load Test Results, Fig. 6.1. Note, multiple bridges may be associated with the same project or job number. Within the Bridge Table, Fig. 6.1, are the Bridge Numbers, used in maintenance. Associated with a given bridge are its Pier Tables, Fig. 6.1, which have individual piles and/or shaft tables. Each pile/shaft table contains information (data not shown) concerning as-built (i.e., size, properties, material: steel concrete, etc.). For an "as-built =false" entry in Pile/Shaft Table, Fig. 6.1, the information is for design and not the final constructed pile/shaft information. In the case of "as-built=true," the information was obtained during the construction process, e.g., Pile Technician software. Other tables associated with an individual "as-built=true" pile/shafts are the Load Test Tables, Fig. 6.1: 1) Static Top Down; 2) Osterberg bottom up, 3) Statnamic, as well as 4) PDA/CAPWAP pile driving analyses. Note, each of the Tables shown in Fig. 6.1 contains multiple information, which is discussed later.

The Subsurface (soil/rock) information identified as Hole Tables has a similar architecture as COSMOS/PEER (*2004*). The Hole Tables may have any of the following information: 1) Standard Penetration (SPT), 2) Cone Penetration (CPT), 3) Dilatometer Modulus

(DMT), 4) Vane Shear (VST) Tests and 5) rock core or Shelby tube samples. Note, the Hole Tables may be assigned to a given Project ID or multiple Project IDs, Fig. 6.1. The multiple IDs allow the reuse of the data over the "Life" of the structure, e.g., bridge widening, rehabilitation.

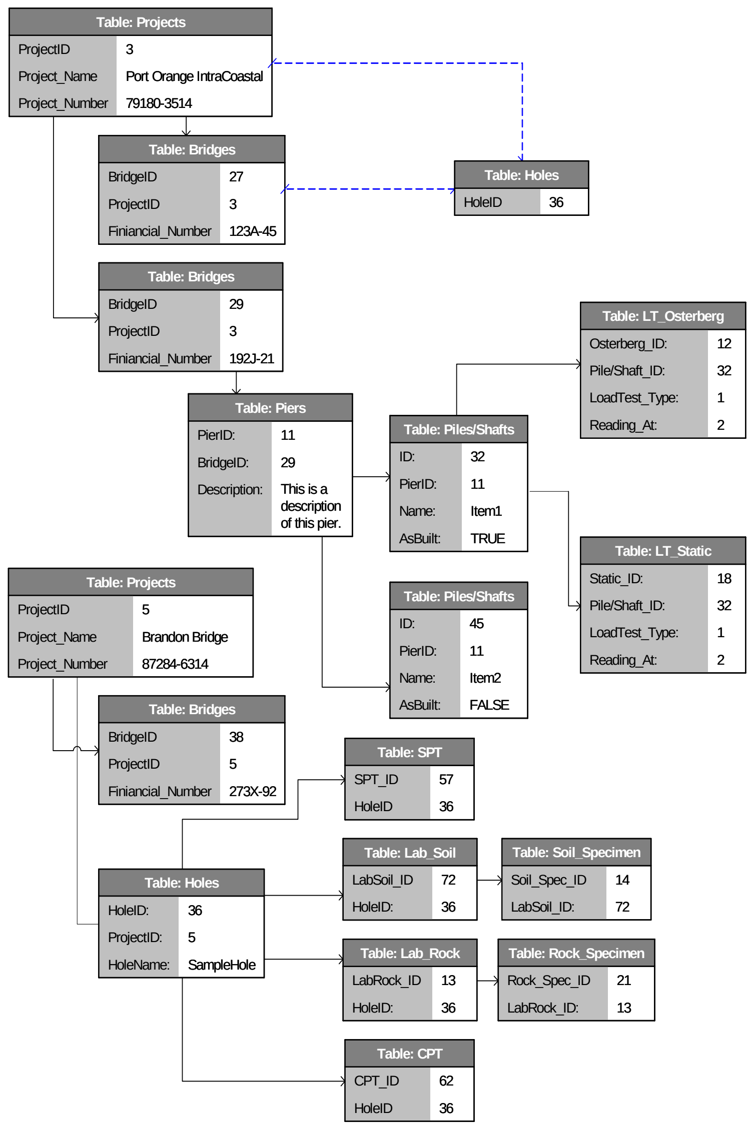


Figure 6.1 FDOT Deep Foundation Database Structure

Attached to the Hole Tables are Specimen Tables and associated laboratory test results. The latter could include simple tests, i.e., grain size, Atterberg, etc. for Soil Classification (Unified or AASHTO Soil Classification), or the more involved compressibility (i.e., oedometer) or strength testing. In the case of soil strength, both drained and undrained triaxial as well as unconfined compression is available. For rock, the RQD Recovery ratio, the percent recovery, as well as split tension, unconfined compression, and scour potential are recorded. Also recorded from the rock's strength testing are its stress-strain behavior, and secant Young's Modulus.

6.3 DATABASE SECURITY

There are four levels of user security for the database. The intent was to follow the standard FDOT review and acceptance levels. All the levels are hierarchical and have authority over the lower levels within the tree. The lowest level (0) is considered the data originator and is the only account allowed to change the entered (i.e. uploaded) data. Figure 6.2 shows the current security levels for the data tables within the database.

Users at each level can be assigned access to individual tables and values by an FDOT administrator through the web interface, Figure 6.3. The access can include read only and lock or update, read and lock for level 0. Once the lower level is locked, the higher level has review and locking authority over the lower level. However, only the data originator (level 0) can modify the data. A discussion of access by any user is given in chapter 4.

6.4 DATABASE ACCESS

The database is written using Microsoft IIS in the .NET environment taking advantage of ODBC connections. All of the file intelligence and security is handled by the web server. ODBC handles the database connection and allows for the use of a variety of databases.

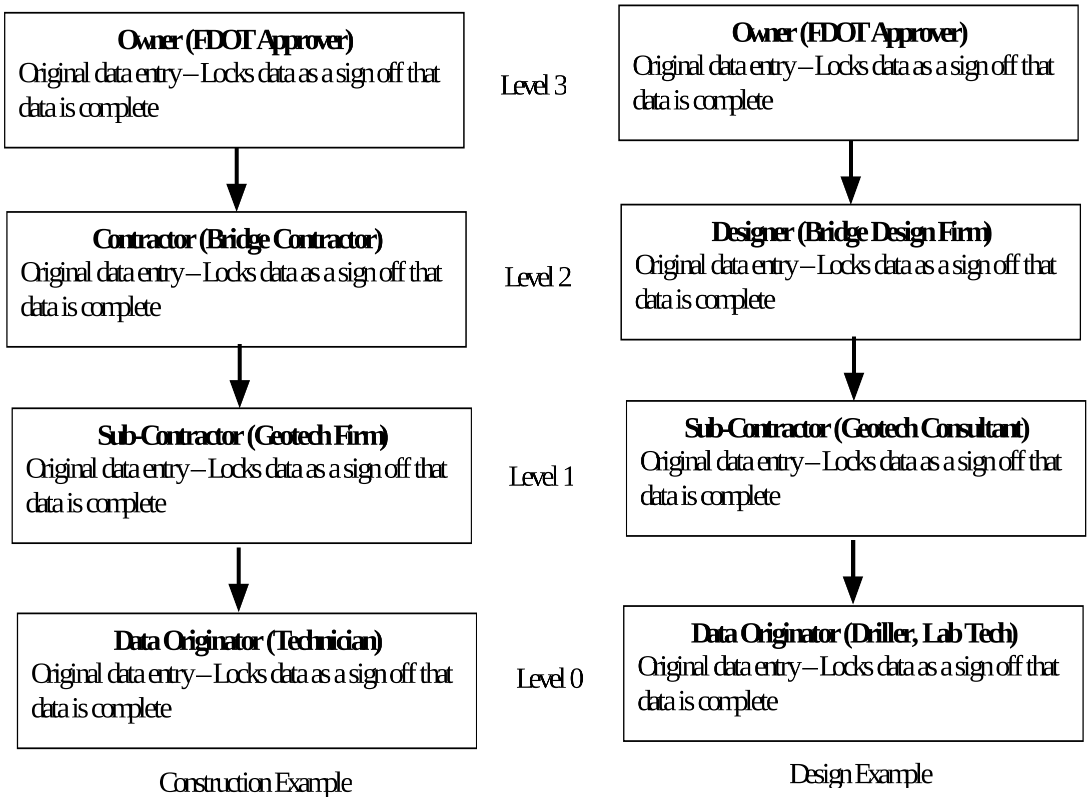


Figure 6.2 FDOT Administration, Contractors, and Consultant Data Security

Database interaction is handled completely through the transfer of XML files using the web server. Data is retrieved from the database by first selecting a portion of the data tree and then requesting a download. The selected data is sent to the user as an XML file following the defined XML schema or format. The XML format of the transferred file is flexible. The database has been designed to accommodate a number of different XML schemas. These schemas are defined through a table in the database consisting of XML tags and an XSL style sheet that defines the format of the file. Currently, only the UF-FDOT schema, Appendix B is implemented.

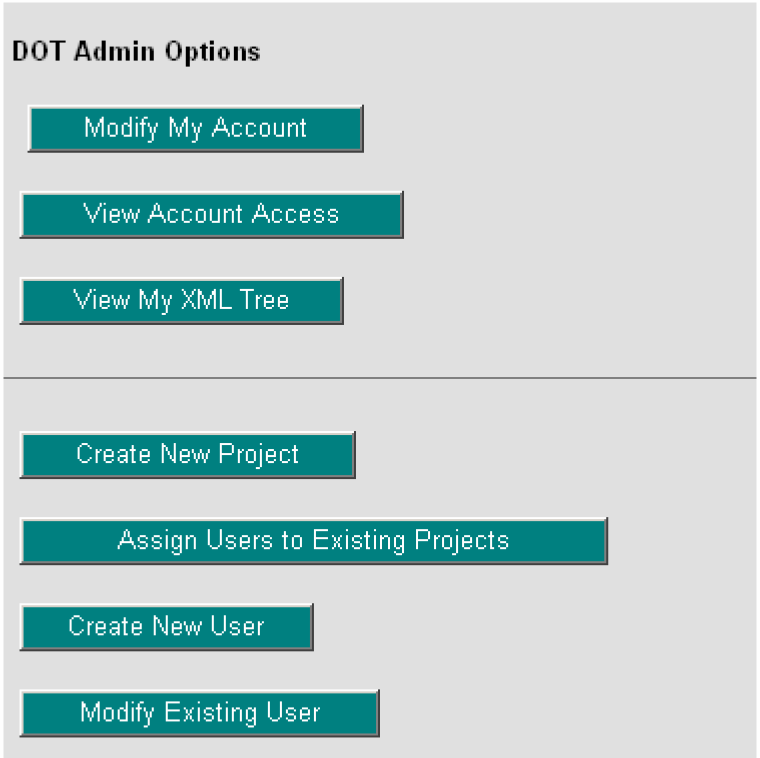


Figure 6.3 FDOT Administrator Project Access Screen

Entering data into the database is also handled through the XML file exchange and the web server. The same format used for sending out from the database is used to enter data into the database. The web server receives the XML file, validates security for each data items and then updates the appropriate pieces of data.

Security is integrated into the structure. Users can be given access to different projects and components under a project, Fig 6.3. For example, a user may be given access to Bridge 1 under Project 1 with complete access to all data. This same user may only be given access to the pile data under project 2, bridge 3. This access can be limited to specific values in any table.

There is also a history mechanism for the data. Basic data is appended to the table, rather than inserted, and flagged as the current version; older versions are tagged sequentially to allow roll-back. Detailed data such as SPT test data or Statnamic tests are replaced and not appended in order to keep the data storage at a reasonable level.

The database structure has been developed with expansion in mind. The XML file structure is controlled by a table and XSL views. Additional tables can be added following the defined structure of project hierarchy.

6.5 DATABASE-SOFTWARE INTERACTION

Since the goal of the database is for the user to interact through application software, a number of methods have been created to make this process simpler. There are two delivered tools for interacting with the data: a web browser and a DLL callable from any software application.

The web browser allows users to search the data tree, select a branch and have data delivered to the browser. There are two forms of delivery available, 1) save the data to a XML file, or view the file using pre-defined XSL style sheets. The style sheets allow the data to be displayed in tables and will be important for reviewers and data acceptance. The file save option allows the XML file to be saved to a computer for use by any software application.

The DLL interface was created to simplify the connection for software applications. DLL's are a universal way for any programming language to access the database. Most software applications have a built in programming language. For example, MS Excel, AutoCAD, and many others use VBA (Visual Basic for Applications). Other software applications that are written in C, C++, Java or others can call the DLL and transfer data to and from the database.

The DLL provides two functions: Send an XML file to the Database or receive a data tree, select the requested data and pass the resulting XML file to the software application. The source code for the DLL is also provided as well as examples for connections using Excel.

Neither the DLL nor the browser is required to interact with the database. Since it is controlled through the web server, all security and interactions are handled by the server and accessed through the XML files. Any application that sends or receives the correct form of the XML files and is authenticated through the user login can access the data.

6.6 EXCEL SPREADSHEETS

Excel spreadsheets are typically used for reducing data from the laboratory (e.g., compressibility, strength, etc.) or the field (e.g., Osterberg, Statnamic Tests, etc.). For instance, Figure 6.4 shows the recording of laboratory soil/rock data, and Figure 6.5 shows the reduction of a conventional top down static load test. In the latter case, pile/shaft capacity estimates (i.e., Davisson, & Debeer), as well as unit skin friction alongside the shaft were back computed from the strain gages within the spreadsheet. Generally, the data recorded and reduced (e.g., pile/shaft capacities, soil/rock properties, etc.) are printed and turned in as a report during the design or construction phase depending on the activity. However, with the proposed architecture, the information (e.g., Excel spreadsheet) may be uploaded directly by the consultant to the database (Fig. 6.1) and locked (Fig. 6.2) for later use on this project or another project.

Presented in Figure 6.6 is the layout of currently available Excel Spreadsheets for use with the database (input/output). They are divided into five general groups: 1) Design, 2) As Built, 3) Insitu , 4) Laboratory, and 5)Field Testing. Generally each group, (i.e., Design, As Built, etc.) is a separate Excel file with multiple sheets (Fig. 6.6 tree structure).

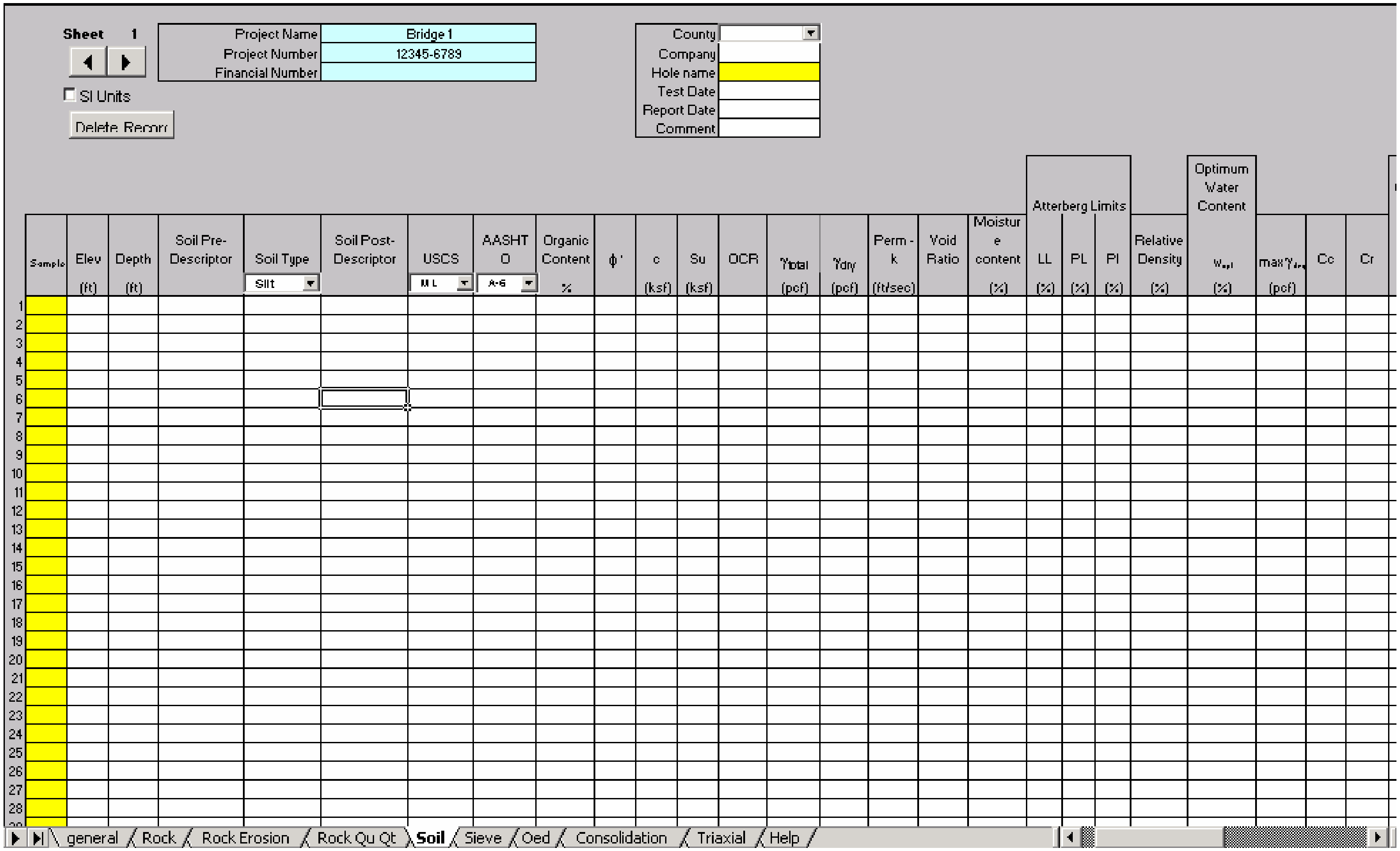
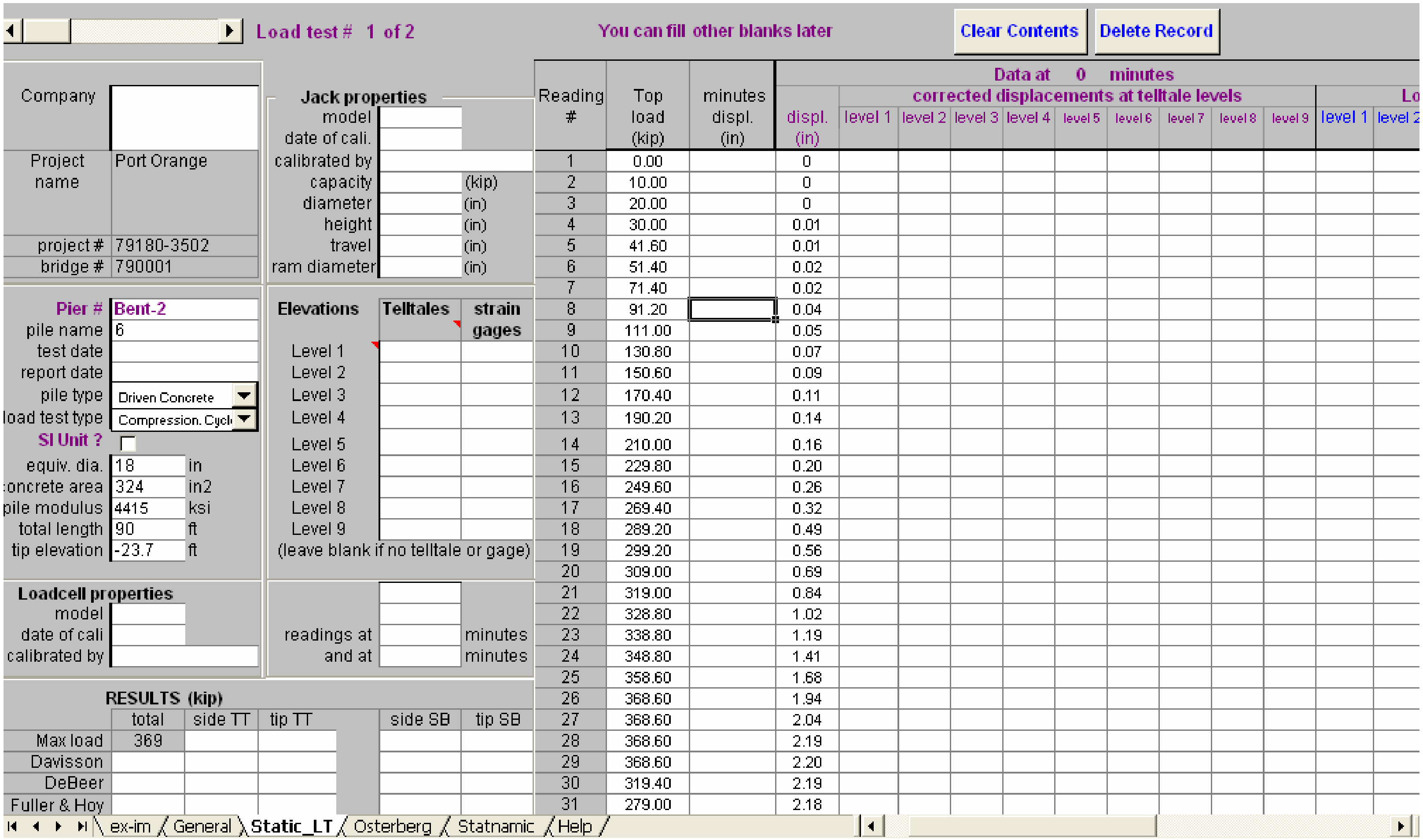


Figure 6.4 Laboratory Soil/Rock Excel Spreadsheet



Reading #	Top load (kip)	minutes displ. (in)	displ. (in)
1	0.00		0
2	10.00		0
3	20.00		0
4	30.00		0.01
5	41.60		0.01
6	51.40		0.02
7	71.40		0.02
8	91.20		0.04
9	111.00		0.05
10	130.80		0.07
11	150.60		0.09
12	170.40		0.11
13	190.20		0.14
14	210.00		0.16
15	229.80		0.20
16	249.60		0.26
17	269.40		0.32
18	289.20		0.49
19	299.20		0.56
20	309.00		0.69
21	319.00		0.84
22	328.80		1.02
23	338.80		1.19
24	348.80		1.41
25	358.60		1.68
26	368.60		1.94
27	368.60		2.04
28	368.60		2.19
29	368.60		2.20
30	319.40		2.19
31	279.00		2.18

Figure 6.5 Top Down Conventional Static Load Test Excel Spreadsheet

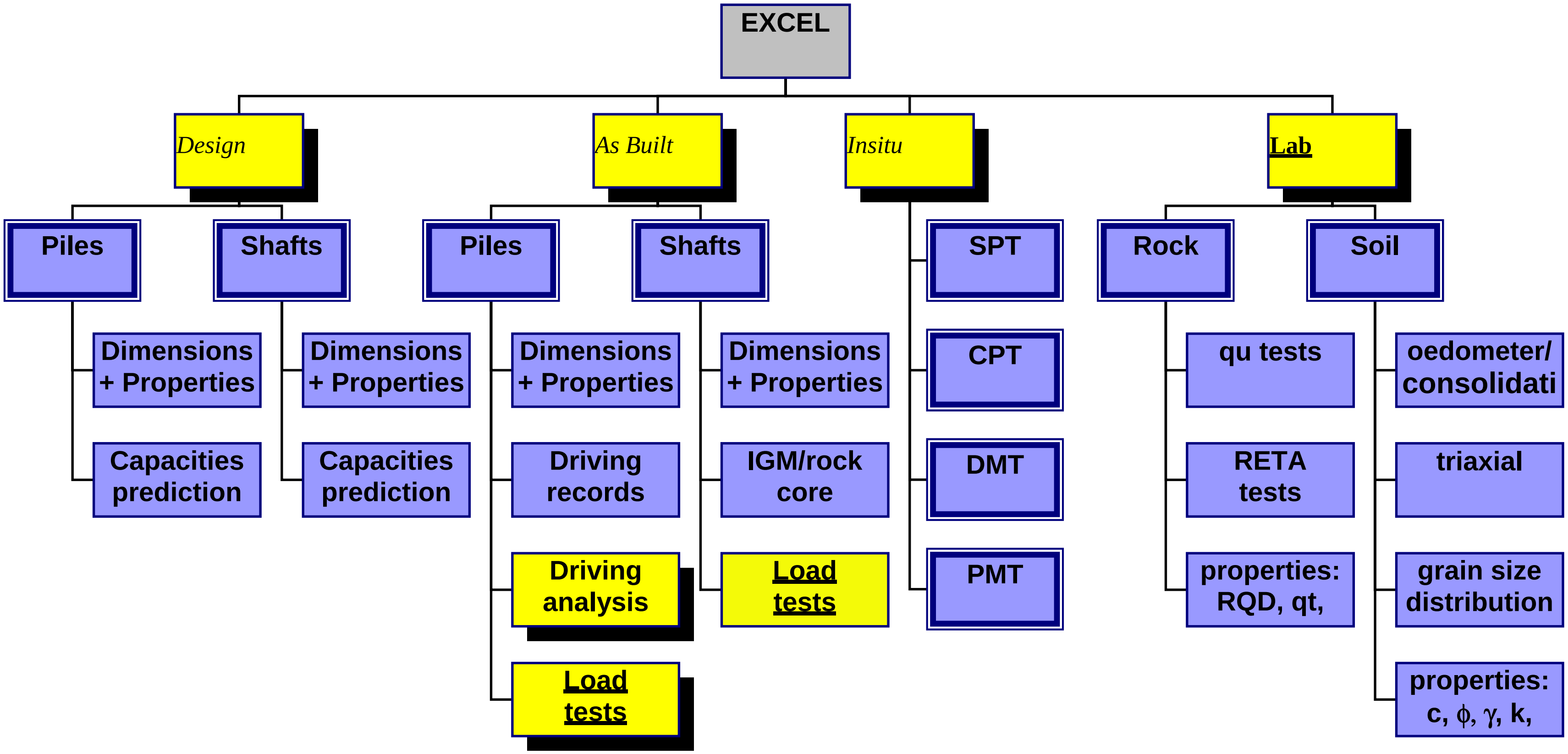


Figure 6.6 Excel Spreadsheets for Uploading and Downloading Data to FDOT Deep Foundation Database

For field testing, separate Tab sheets within Field Testing are provided for 1) Conventional Top Down Static Loading, 2) Osterberg Bottom Up Loading, and 3) Statnamic Dynamic Loading. The separation of the different Excel sheets are required to deal volume of data, as well as the fact that different contractors may perform different tasks on a project. Note, the layout, analysis, etc. of both the Statnamic and Osterberg Spreadsheets have been reviewed by a number of FDOT consultants. A by product of the Excel spreadsheets (Design, As Built, etc.), and the database is the quality control and quality assurance (QA/QC) that the FDOT can exert on consultants, contractors, etc. for successful bridge design/construction.

6.7 LRFD RESISTANCE FACTORS FOR FB-DEEP

All of the driven prestressed concrete piles in the earlier FDOT Microsoft Access Deep Foundation Database were entered into the new Internet Based FDOT Deep Foundation Database (Chapter 2) through the Excel "Load Test" spreadsheets (Chapter 3). A total of 56 prestressed concrete piles, Table 5.5 which had both top down static load tests and reached Davisson Failure Capacity were considered for the LRFD study. The FB-Deep software was modified to read and write the XML schema tags (Appendix B), as well as upload and download the Insitu and Laboratory Data (i.e. piles and shaft design) with the DLL program discussed in Chapter 4. Subsequently, FB-Deep was used to predict the FDOT failure capacity of the piles using the new database. Using the predicted and measured capacities, the LRFD resistance factors, ϕ, were computed for different bridge live to dead load ratios, and reliability, β, values.

As expected, the resistance factor derived from the reliability based calibration of FB-Deep with Florida projects varied from 0.45 to 0.70 depending on reliability values, Table 5.7, which is an improvement over current AASHTO recommendations for SPT testing. The resistance factors derived in this study are comparable to similar calibrations preformed in earlier studies (Singletary, McVay, etc.). It is recommended that the resistance factors be periodically recalibrated as the FDOT database grows during its implementation.

6.8 CONCLUSIONS

The implementation of a flexible and extendable web accessible database has been described. It has been implemented for the Florida Department of Transportation in order to be an active repository for Geotechnical Deep Foundation data. The key concept of the database is its accessibility through the Internet with security, and universal XML schema (Appendix B). Any application can retrieve or submit data to the database provided the user has the proper security and parser (DLL).

The application centric view of the database makes it useable for multiple functions; data warehouse, work flow repository and a research data collection. The data warehouse option moves the concept of electronic storage of data into an active form of storage. All stored data can be actively used by any application without recoding. The workflow repository option allows the database to be the storage location for data to be shared during project development and construction and maintenance. An engineer, technician contractor, etc. can put their results into the database and allow other engineers or contractors direct access to the data in a form viewable or usable by other software, web viewers, etc. Finally, the research option allows owners, researchers and others to query database from anywhere to find, collect and analyze data for their use.

This type of storage for direct use has been the goal of many data storage efforts. The web and current database technology offers the tools and capability to bring this vision into practice. Setting transfer standards (i.e. XML Geotechnical Schema) will allow the sharing of data and improve the overall quality of results for major engineering projects.

CHAPTER 7

REFERENCES

1. COSMOS/PEER (2001), "Invited Workshop on Archiving and Web Dissemination of Geotechnical Data", October 4-5, Richmond, CA, COSMOS Publication CP-2001/03.

2. National Cooperative Highway Research Program (NCHRP) Project 20-64, *XML Schemas for Exchange of Transportation Data*, URL site: http://www.transxml.com

3. Association of Geotechnical Specialists (AGS) (1992), "Electronic Transfer of Geotechnical Data from Ground Investigations," United Kingdom.

4. American Association of State Highway and Transportation Officials (1994), LRFD Bridge Design Specification, SI Units, First Edition.

5. McVay, M., Kuo, C., and Singletary, W. (1998), Calibrating Resistance Factors in the Load and Resistance Factor Design for Florida Foundations, Florida Department of Transportation.

6. Nguyen, Thai (2001), Load and Resistance Factor Design for Driven Piles Based on Static Methods, University of Florida, Gainesville, FL.

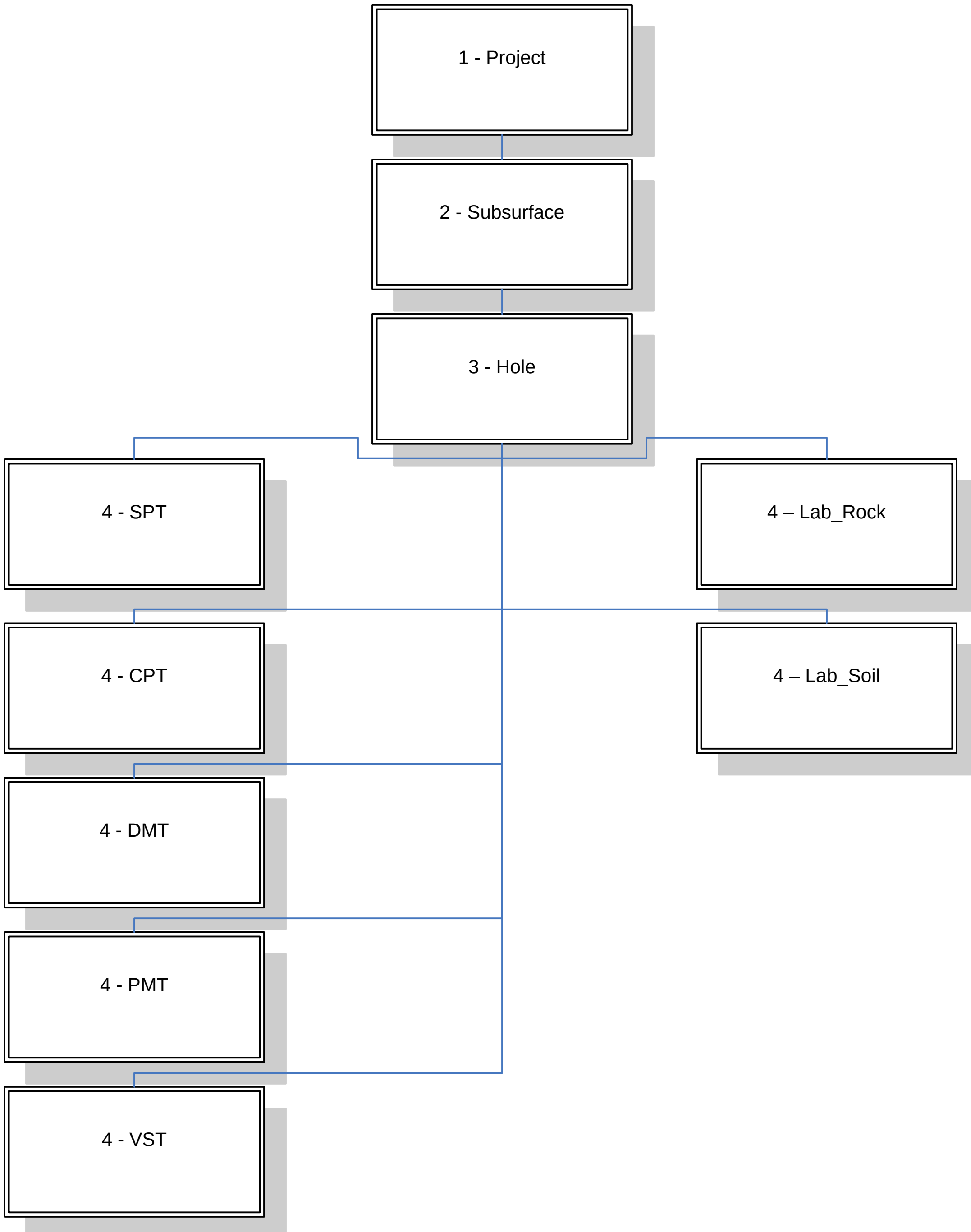
7. Rosenblueth, E. and Esteva, L. (1972), "Reliability Basis for some Mexican Codes", ACI Publication SP-31, American Concrete Institute, Detroit MI.

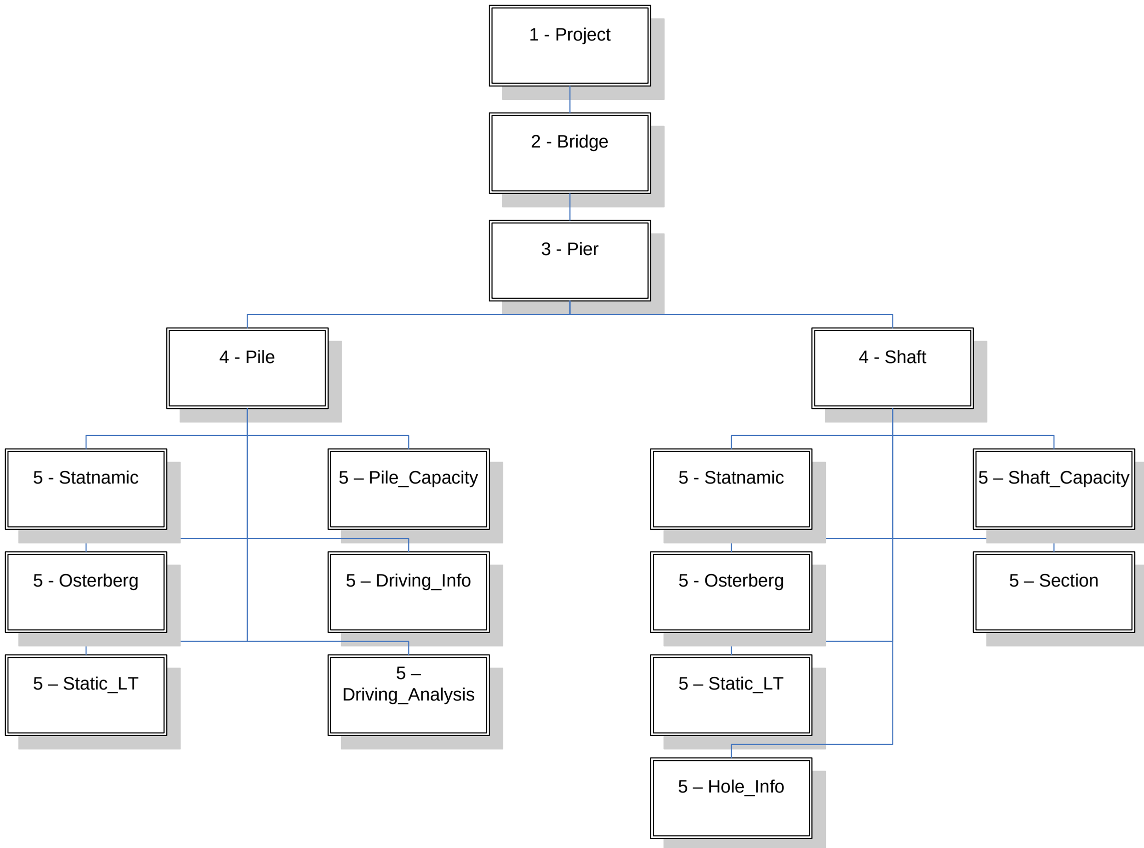
8. Singletary, William (1998), Calibration of Load and Resistance Factor Design (LRFD) Resistance Factors for Geotechnical Designs in the State of Florida, University of Florida, Gainesville, FL.

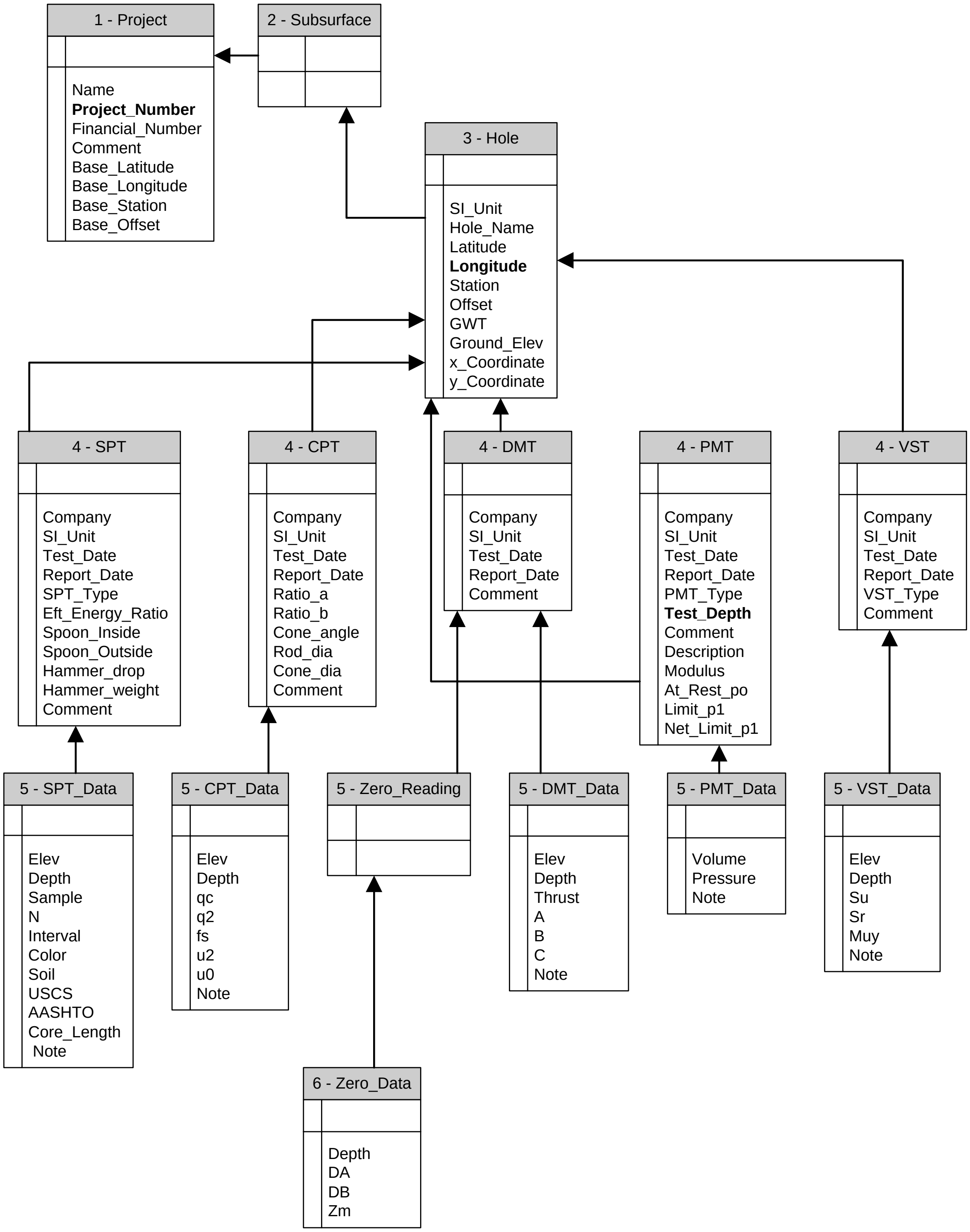
9. NCHRP Report 507 (2004), Load and Resistance Factor Design (LRFD) for Deep Foundations, Transportation Research Board, Washington D.C.

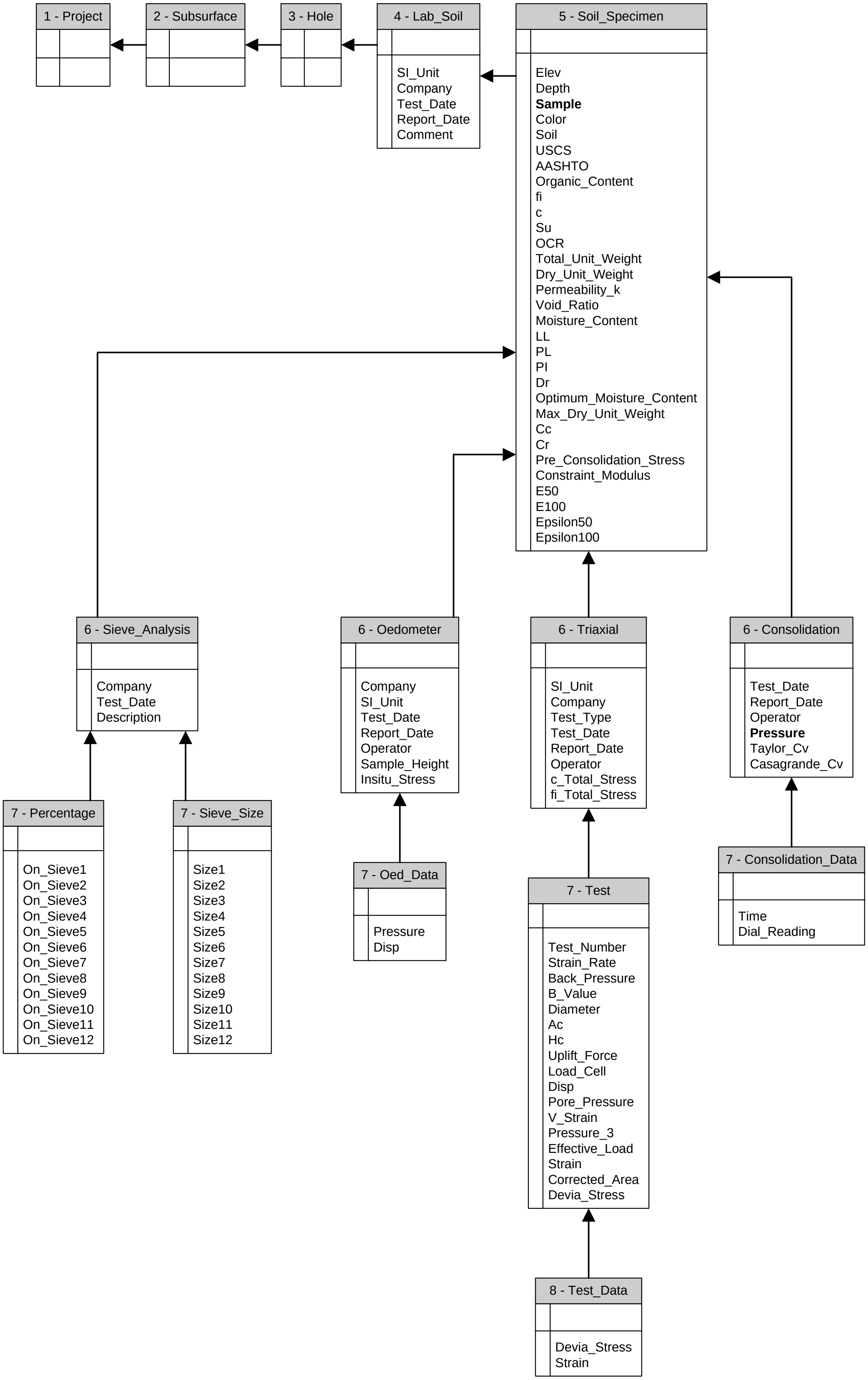
Appendix A

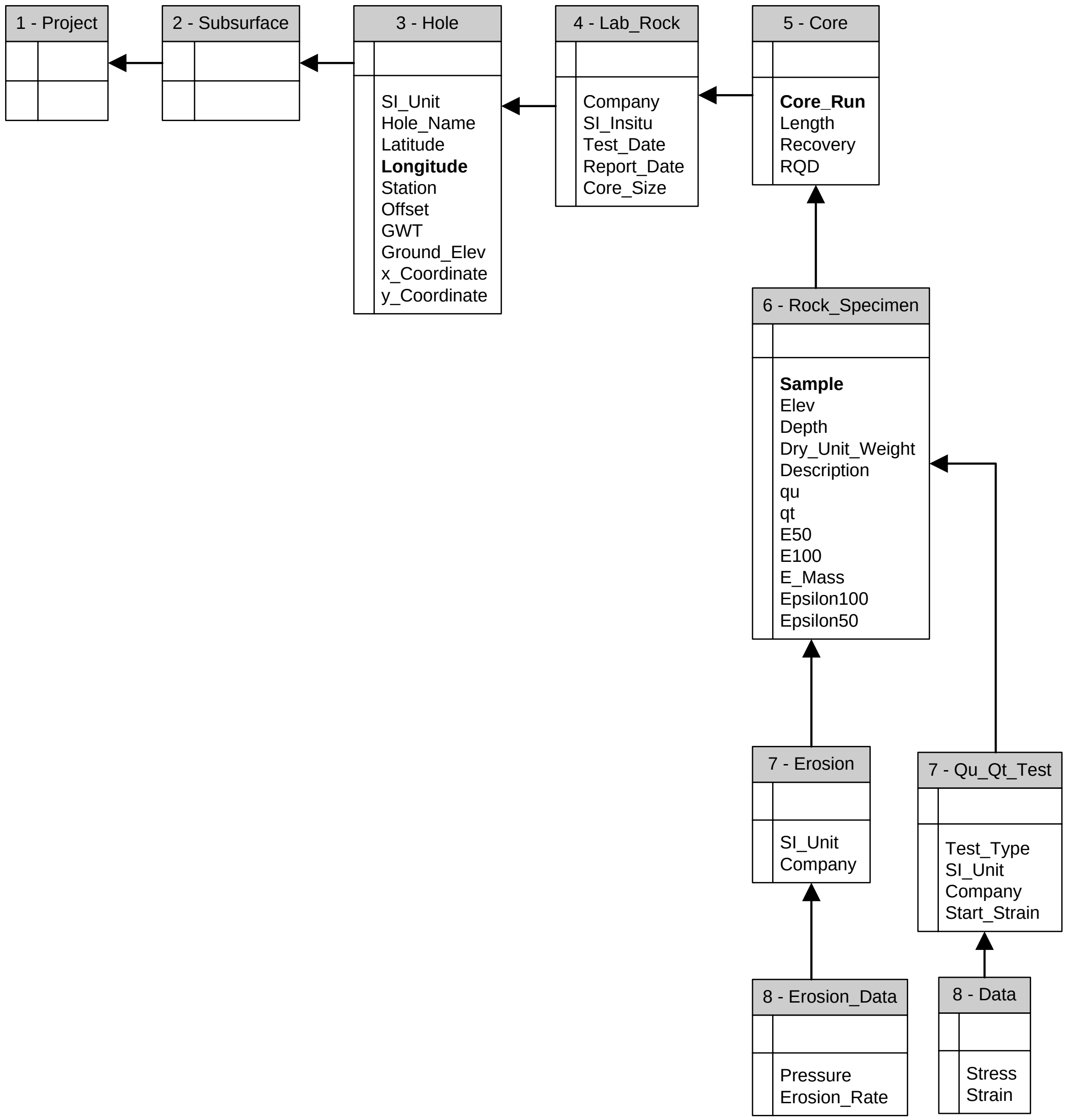
FDOT DATA STRUCTURE AND DICTIONARY

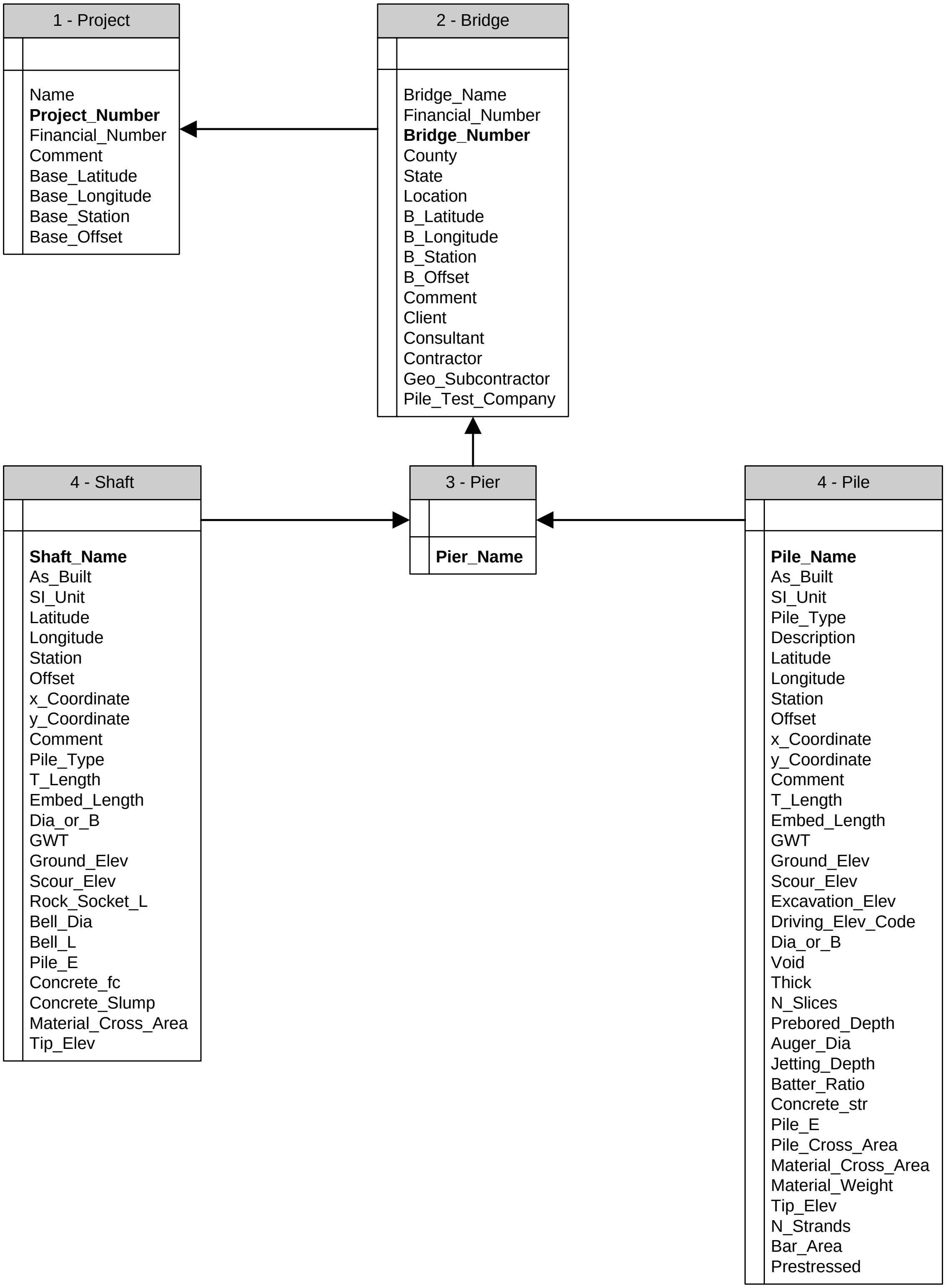

1 - Project
2 - Subsurface
3 - Hole
4 - SPT
4 – Lab_Rock
4 - CPT
4 – Lab_Soil
4 - DMT
4 - PMT
4 - VST

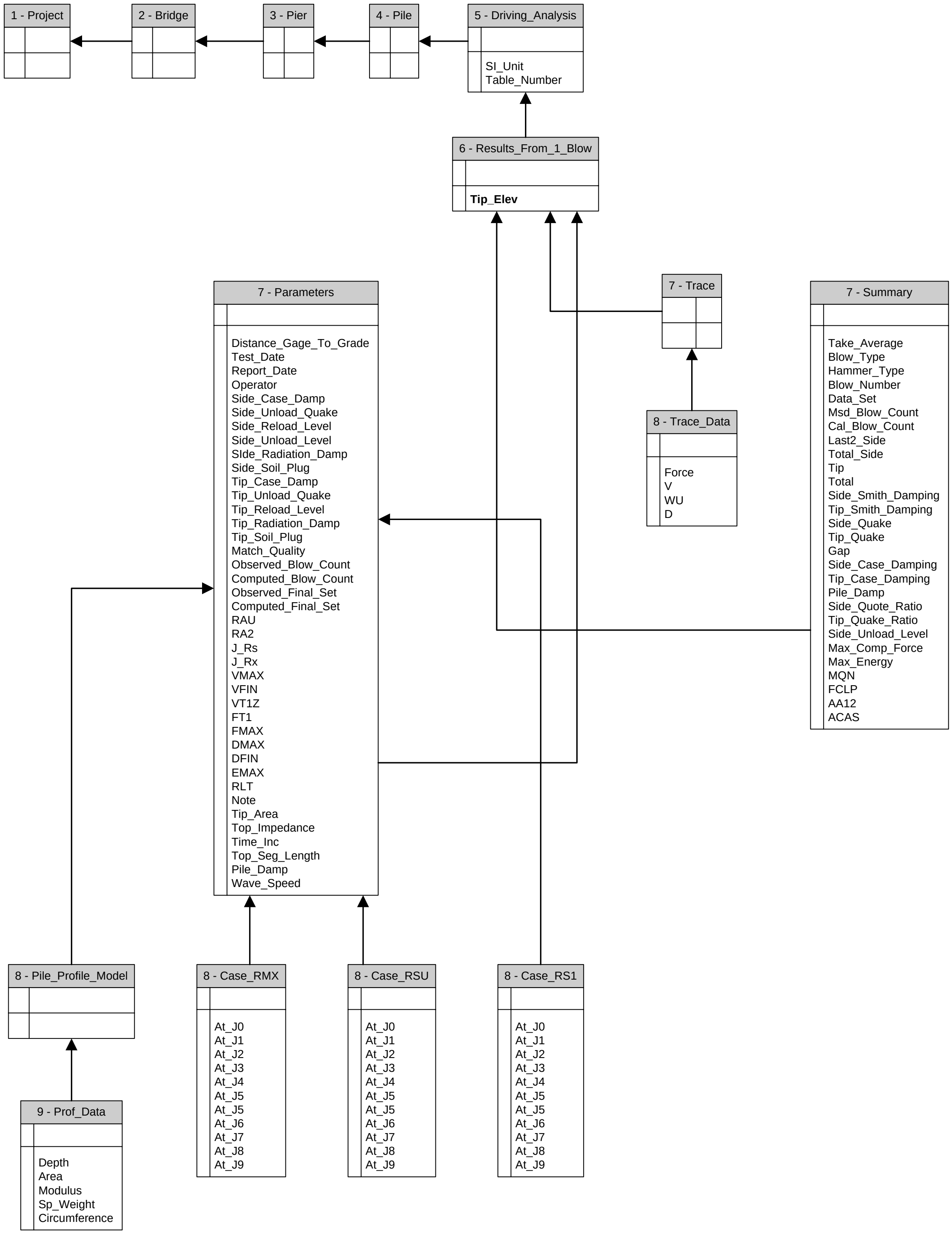

1 - Project
2 - Bridge
3 - Pier
4 - Pile
4 - Shaft
5 - Statnamic
5 – Pile_Capacity
5 - Osterberg
5 – Driving_Info
5 – Static_LT
5 – Driving_Analysis
5 - Statnamic
5 – Shaft_Capacity
5 - Osterberg
5 – Section
5 – Static_LT
5 – Hole_Info

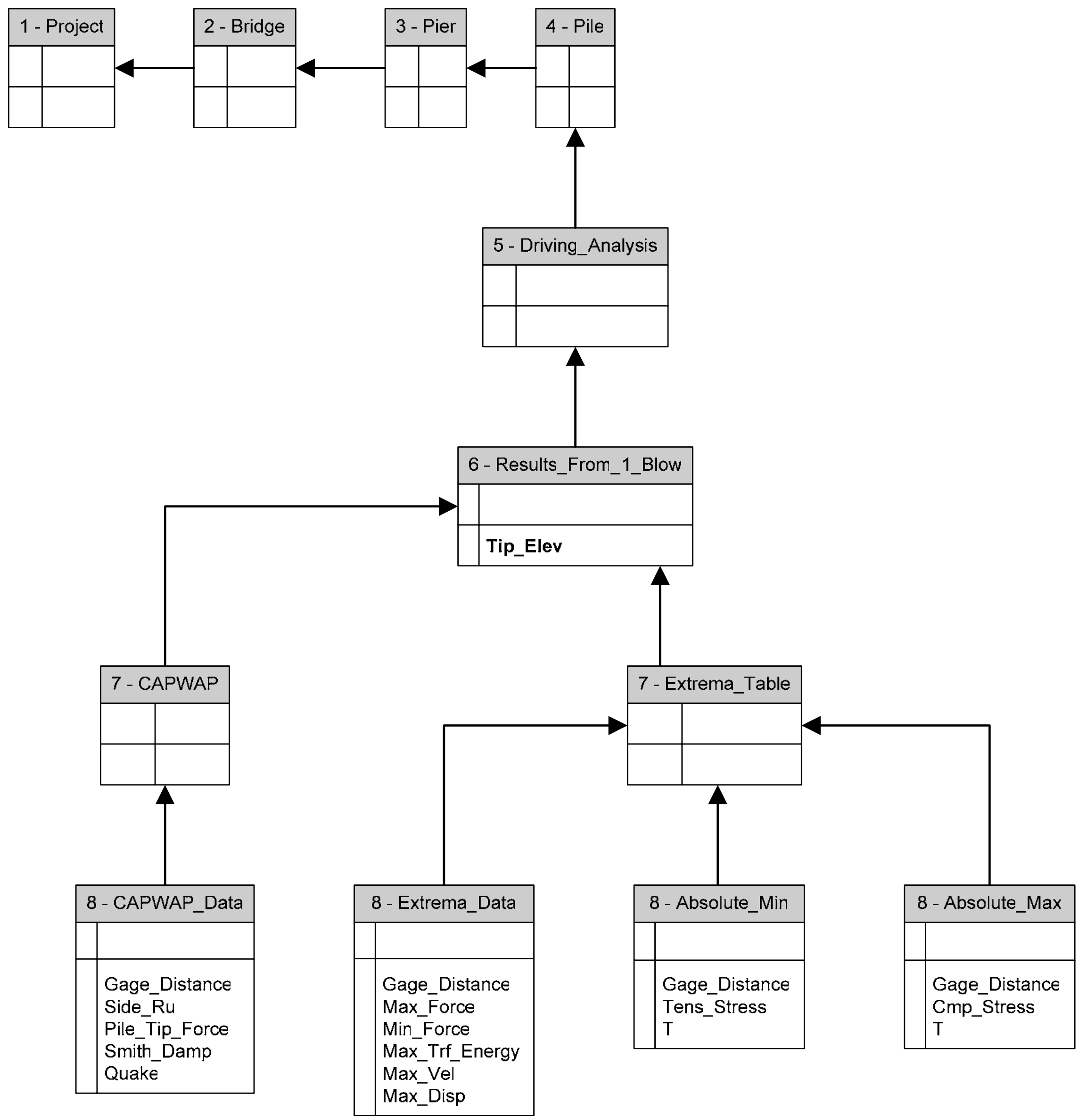

1 - Project
Name
Project_Number
Financial_Number
Comment
Base_Latitude
Base_Longitude
Base_Station
Base_Offset
2 - Subsurface
3 - Hole
SI_Unit
Hole_Name
Latitude
Longitude
Station
Offset
GWT
Ground_Elev
x_Coordinate
y_Coordinate
4 - SPT
Company
SI_Unit
Test_Date
Report_Date
SPT_Type
Eft_Energy_Ratio
Spoon_Inside
Spoon_Outside
Hammer_drop
Hammer_weight
Comment
4 - CPT
Company
SI_Unit
Test_Date
Report_Date
Ratio_a
Ratio_b
Cone_angle
Rod_dia
Cone_dia
Comment
4 - DMT
Company
SI_Unit
Test_Date
Report_Date
Comment
4 - PMT
Company
SI_Unit
Test_Date
Report_Date
PMT_Type
Test_Depth
Comment
Description
Modulus
At_Rest_po
Limit_p1
Net_Limit_p1
4 - VST
Company
SI_Unit
Test_Date
Report_Date
VST_Type
Comment
5 - SPT_Data
Elev
Depth
Sample
N
Interval
Color
Soil
USCS
AASHTO
Core_Length
Note
5 - CPT_Data
Elev
Depth
qc
q2
fs
u2
u0
Note
5 - Zero_Reading
5 - DMT_Data
Elev
Depth
Thrust
A
B
C
Note
5 - PMT_Data
Volume
Pressure
Note
5 - VST_Data
Elev
Depth
Su
Sr
Muy
Note
6 - Zero_Data
Depth
DA
DB
Zm

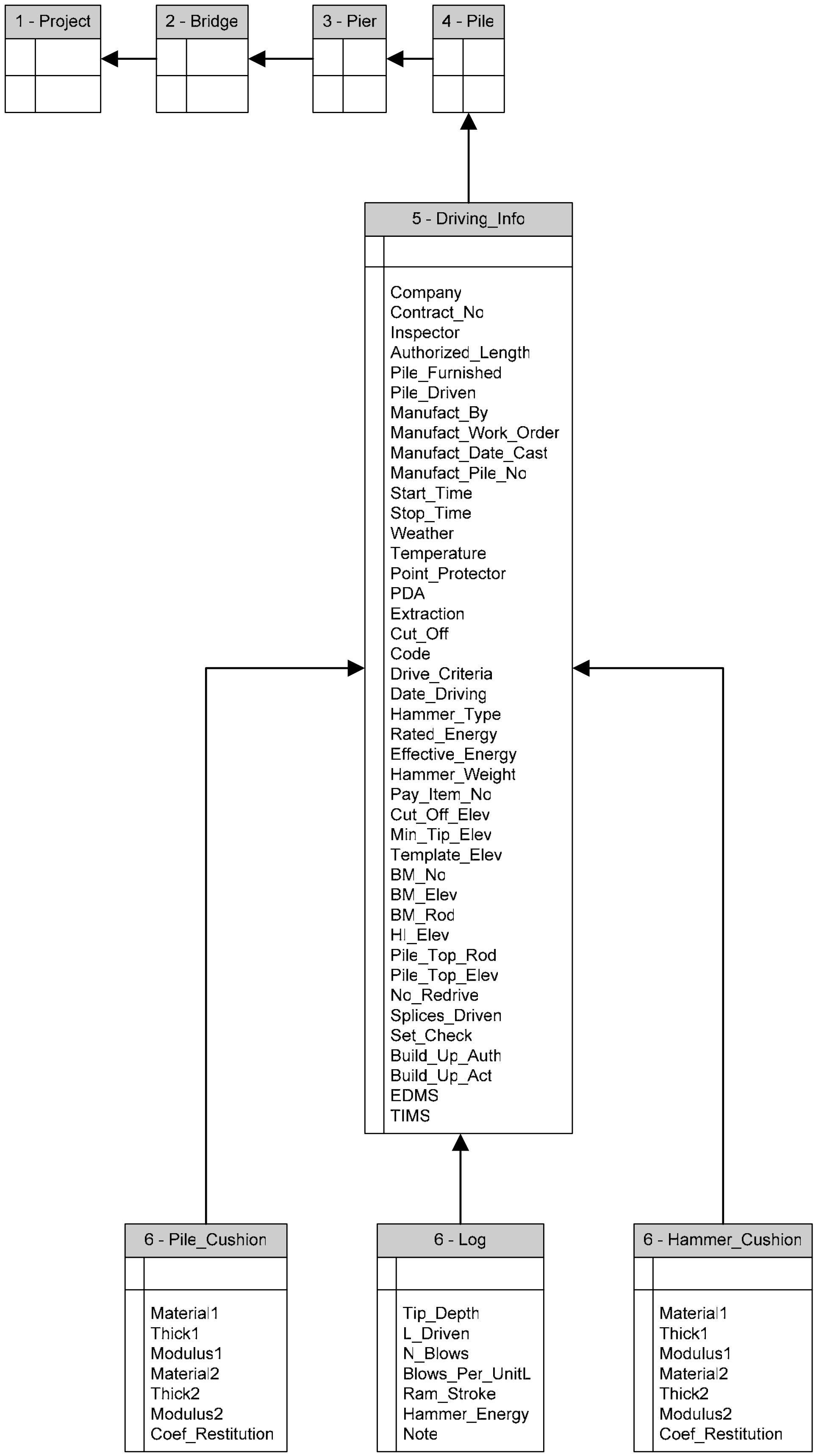

1 - Project
2 - Subsurface
3 - Hole
4 - Lab_Soil
SI_Unit
Company
Test_Date
Report_Date
Comment
5 - Soil_Specimen
Elev
Depth
Sample
Color
Soil
USCS
AASHTO
Organic_Content
fi
c
Su
OCR
Total_Unit_Weight
Dry_Unit_Weight
Permeability_k
Void_Ratio
Moisture_Content
LL
PL
PI
Dr
Optimum_Moisture_Content
Max_Dry_Unit_Weight
Cc
Cr
Pre_Consolidation_Stress
Constraint_Modulus
E50
E100
Epsilon50
Epsilon100
6 - Sieve_Analysis
Company
Test_Date
Description
6 - Oedometer
Company
SI_Unit
Test_Date
Report_Date
Operator
Sample_Height
Insitu_Stress
6 - Triaxial
SI_Unit
Company
Test_Type
Test_Date
Report_Date
Operator
c_Total_Stress
fi_Total_Stress
6 - Consolidation
Test_Date
Report_Date
Operator
Pressure
Taylor_Cv
Casagrande_Cv
7 - Percentage
On_Sieve1
On_Sieve2
On_Sieve3
On_Sieve4
On_Sieve5
On_Sieve6
On_Sieve7
On_Sieve8
On_Sieve9
On_Sieve10
On_Sieve11
On_Sieve12
7 - Sieve_Size
Size1
Size2
Size3
Size4
Size5
Size6
Size7
Size8
Size9
Size10
Size11
Size12
7 - Oed_Data
Pressure
Disp
7 - Test
Test_Number
Strain_Rate
Back_Pressure
B_Value
Diameter
Ac
Hc
Uplift_Force
Load_Cell
Disp
Pore_Pressure
V_Strain
Pressure_3
Effective_Load
Strain
Corrected_Area
Devia_Stress
7 - Consolidation_Data
Time
Dial_Reading
8 - Test_Data
Devia_Stress
Strain

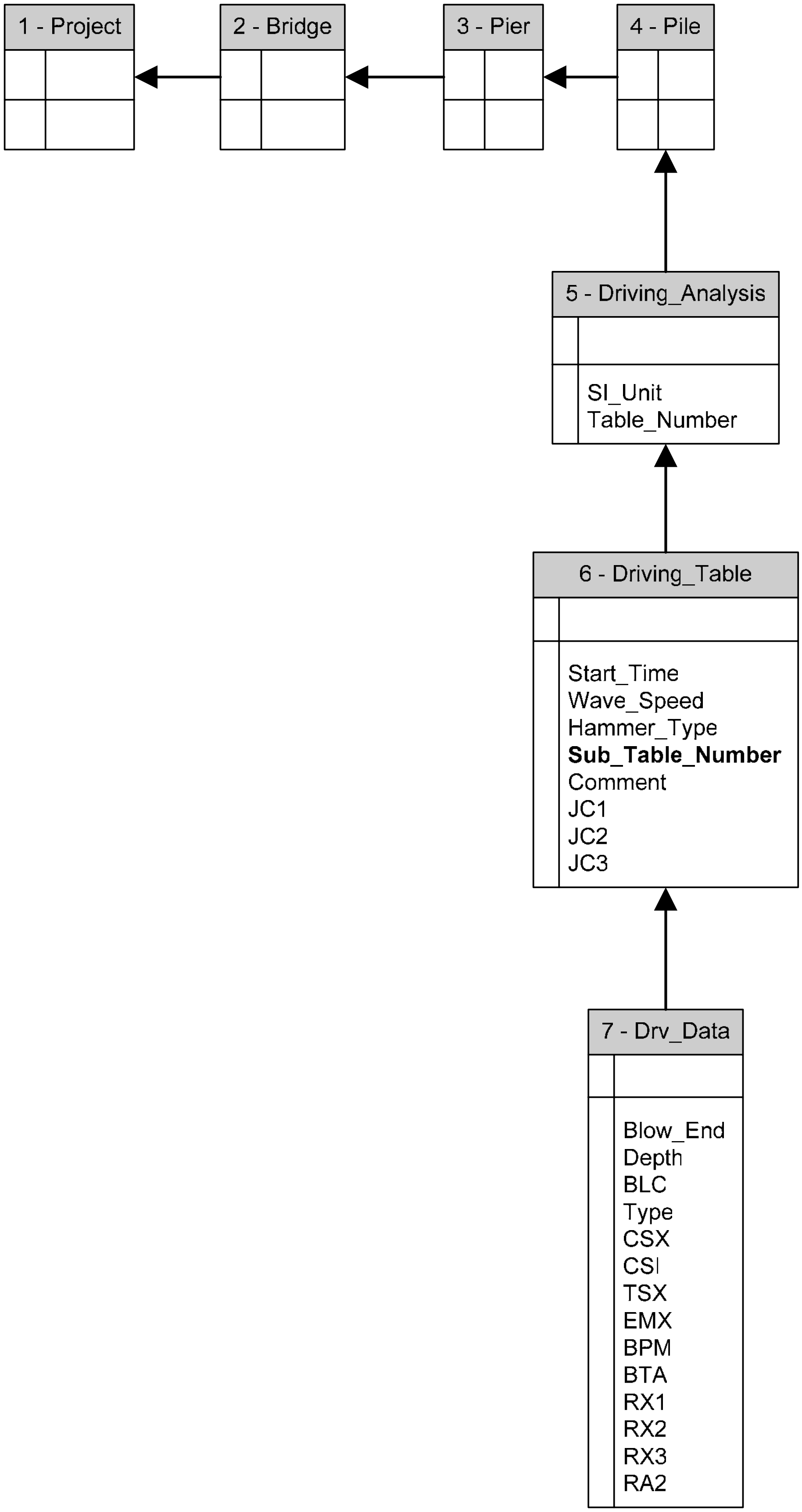

1 - Project
2 - Subsurface
3 - Hole
SI_Unit
Hole_Name
Latitude
Longitude
Station
Offset
GWT
Ground_Elev
x_Coordinate
y_Coordinate
4 - Lab_Rock
Company
SI_Insitu
Test_Date
Report_Date
Core_Size
5 - Core
Core_Run
Length
Recovery
RQD
6 - Rock_Specimen
Sample
Elev
Depth
Dry_Unit_Weight
Description
qu
qt
E50
E100
E_Mass
Epsilon100
Epsilon50
7 - Erosion
SI_Unit
Company
7 - Qu_Qt_Test
Test_Type
SI_Unit
Company
Start_Strain
8 - Erosion_Data
Pressure
Erosion_Rate
8 - Data
Stress
Strain

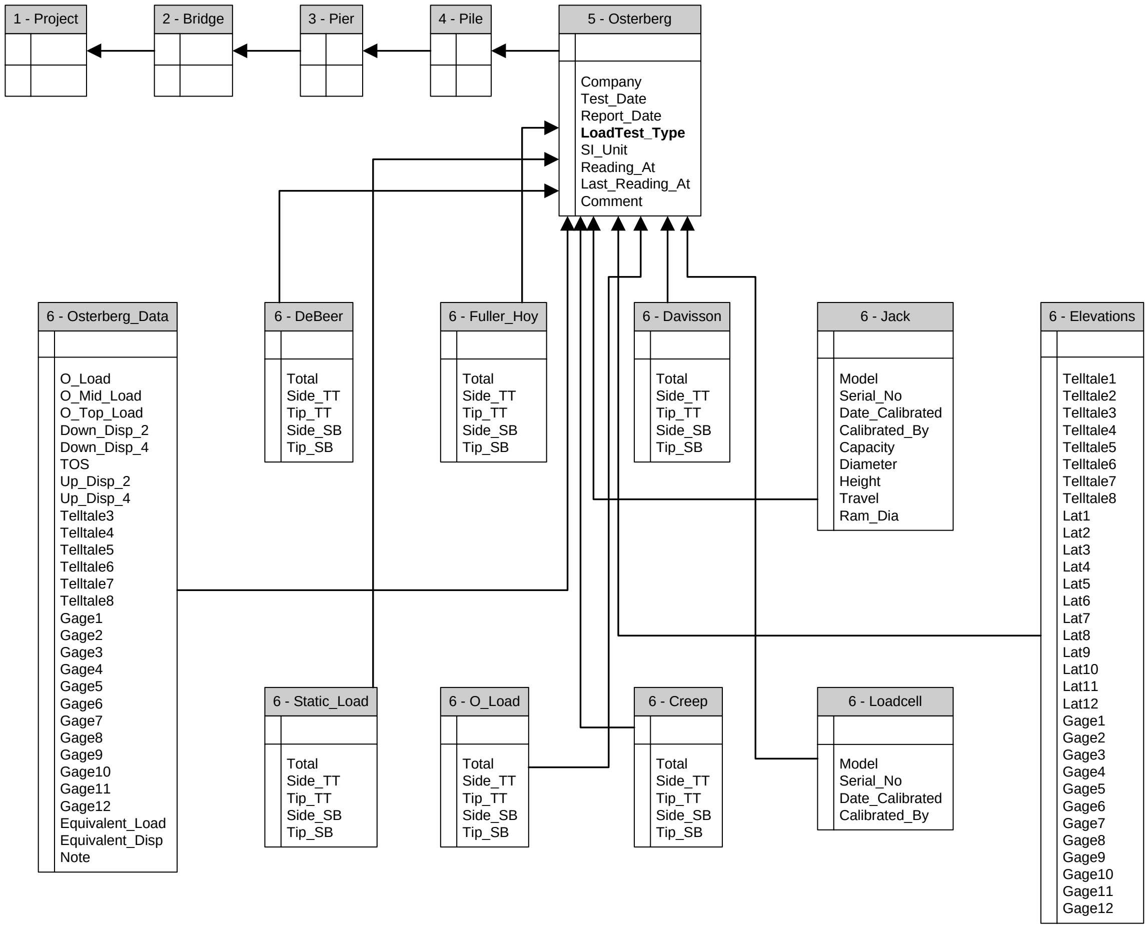

1 - Project
Name
Project_Number
Financial_Number
Comment
Base_Latitude
Base_Longitude
Base_Station
Base_Offset
2 - Bridge
Bridge_Name
Financial_Number
Bridge_Number
County
State
Location
B_Latitude
B_Longitude
B_Station
B_Offset
Comment
Client
Consultant
Contractor
Geo_Subcontractor
Pile_Test_Company
4 - Shaft
Shaft_Name
As_Built
SI_Unit
Latitude
Longitude
Station
Offset
x_Coordinate
y_Coordinate
Comment
Pile_Type
T_Length
Embed_Length
Dia_or_B
GWT
Ground_Elev
Scour_Elev
Rock_Socket_L
Bell_Dia
Bell_L
Pile_E
Concrete_fc
Concrete_Slump
Material_Cross_Area
Tip_Elev
3 - Pier
Pier_Name
4 - Pile
Pile_Name
As_Built
SI_Unit
Pile_Type
Description
Latitude
Longitude
Station
Offset
x_Coordinate
y_Coordinate
Comment
T_Length
Embed_Length
GWT
Ground_Elev
Scour_Elev
Excavation_Elev
Driving_Elev_Code
Dia_or_B
Void
Thick
N_Slices
Prebored_Depth
Auger_Dia
Jetting_Depth
Batter_Ratio
Concrete_str
Pile_E
Pile_Cross_Area
Material_Cross_Area
Material_Weight
Tip_Elev
N_Strands
Bar_Area
Prestressed

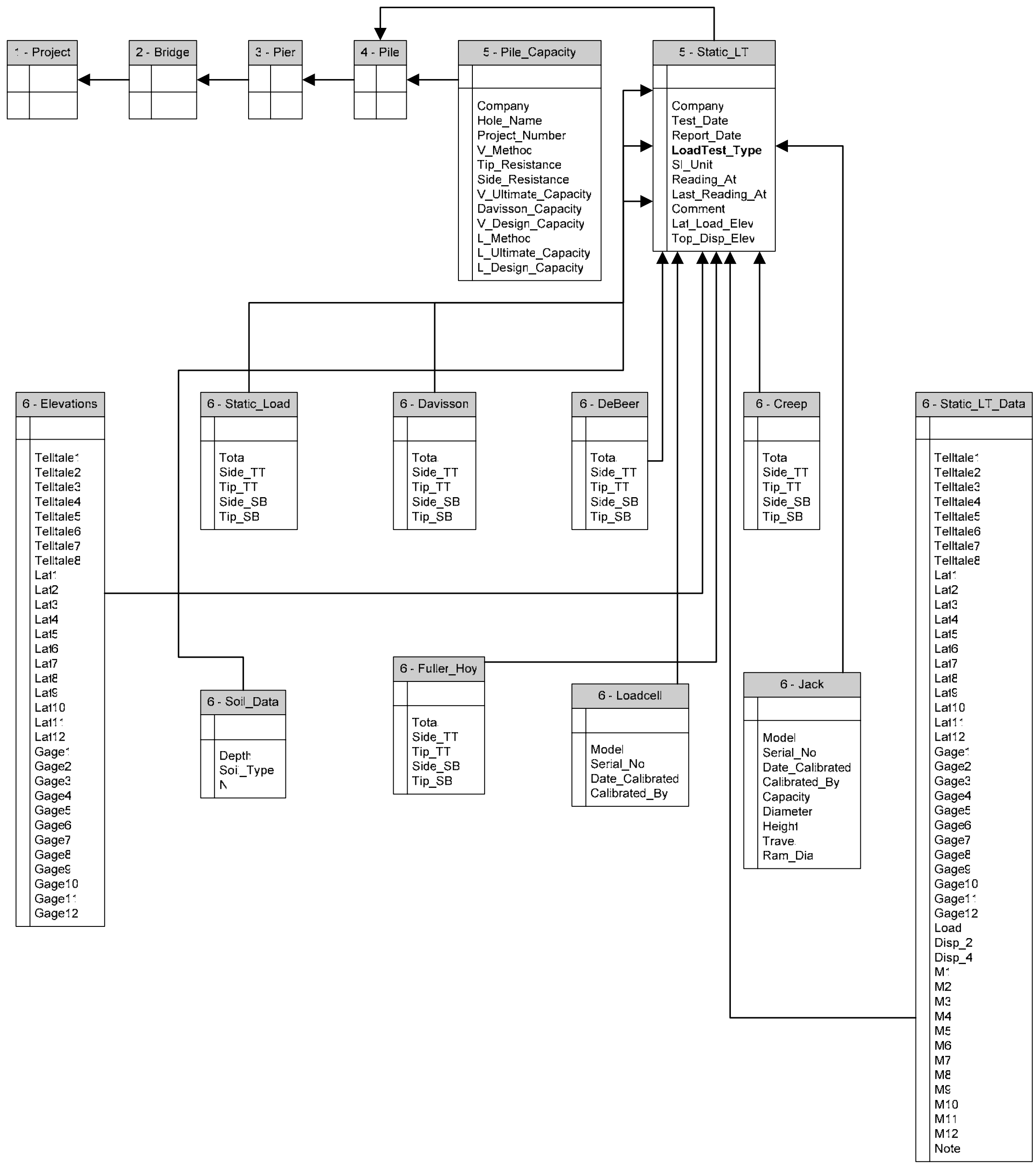

1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Driving_Analysis
SI_Unit
Table_Number
6 - Results_From_1_Blow
Tip_Elev
7 - Parameters
Distance_Gage_To_Grade
Test_Date
Report_Date
Operator
Side_Case_Damp
Side_Unload_Quake
Side_Reload_Level
Side_Unload_Level
SIde_Radiation_Damp
Side_Soil_Plug
Tip_Case_Damp
Tip_Unload_Quake
Tip_Reload_Level
Tip_Radiation_Damp
Tip_Soil_Plug
Match_Quality
Observed_Blow_Count
Computed_Blow_Count
Observed_Final_Set
Computed_Final_Set
RAU
RA2
J_Rs
J_Rx
VMAX
VFIN
VT1Z
FT1
FMAX
DMAX
DFIN
EMAX
RLT
Note
Tip_Area
Top_Impedance
Time_Inc
Top_Seg_Length
Pile_Damp
Wave_Speed
7 - Trace
8 - Trace_Data
Force
V
WU
D
7 - Summary
Take_Average
Blow_Type
Hammer_Type
Blow_Number
Data_Set
Msd_Blow_Count
Cal_Blow_Count
Last2_Side
Total_Side
Tip
Total
Side_Smith_Damping
Tip_Smith_Damping
Side_Quake
Tip_Quake
Gap
Side_Case_Damping
Tip_Case_Damping
Pile_Damp
Side_Quote_Ratio
Tip_Quake_Ratio
Side_Unload_Level
Max_Comp_Force
Max_Energy
MQN
FCLP
AA12
ACAS
8 - Pile_Profile_Model
9 - Prof_Data
Depth
Area
Modulus
Sp_Weight
Circumference
8 - Case_RMX
At_J0
At_J1
At_J2
At_J3
At_J4
At_J5
At_J5
At_J6
At_J7
At_J8
At_J9
8 - Case_RSU
At_J0
At_J1
At_J2
At_J3
At_J4
At_J5
At_J5
At_J6
At_J7
At_J8
At_J9
8 - Case_RS1
At_J0
At_J1
At_J2
At_J3
At_J4
At_J5
At_J5
At_J6
At_J7
At_J8
At_J9

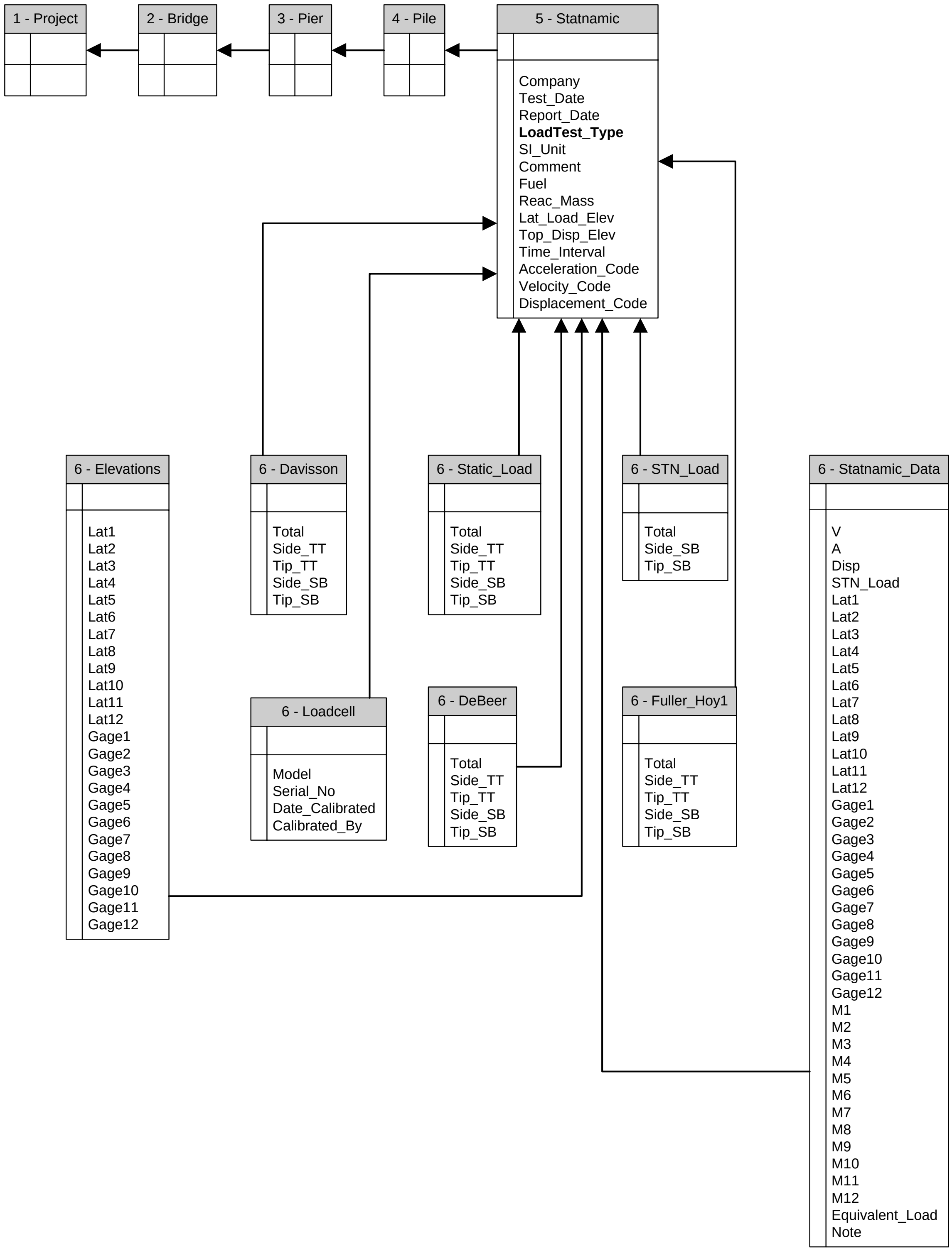

1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Driving_Analysis
6 - Results_From_1_Blow
Tip_Elev
7 - CAPWAP
7 - Extrema_Table
8 - CAPWAP_Data
Gage_Distance
Side_Ru
Pile_Tip_Force
Smith_Damp
Quake
8 - Extrema_Data
Gage_Distance
Max_Force
Min_Force
Max_Trf_Energy
Max_Vel
Max_Disp
8 - Absolute_Min
Gage_Distance
Tens_Stress
T
8 - Absolute_Max
Gage_Distance
Cmp_Stress
T

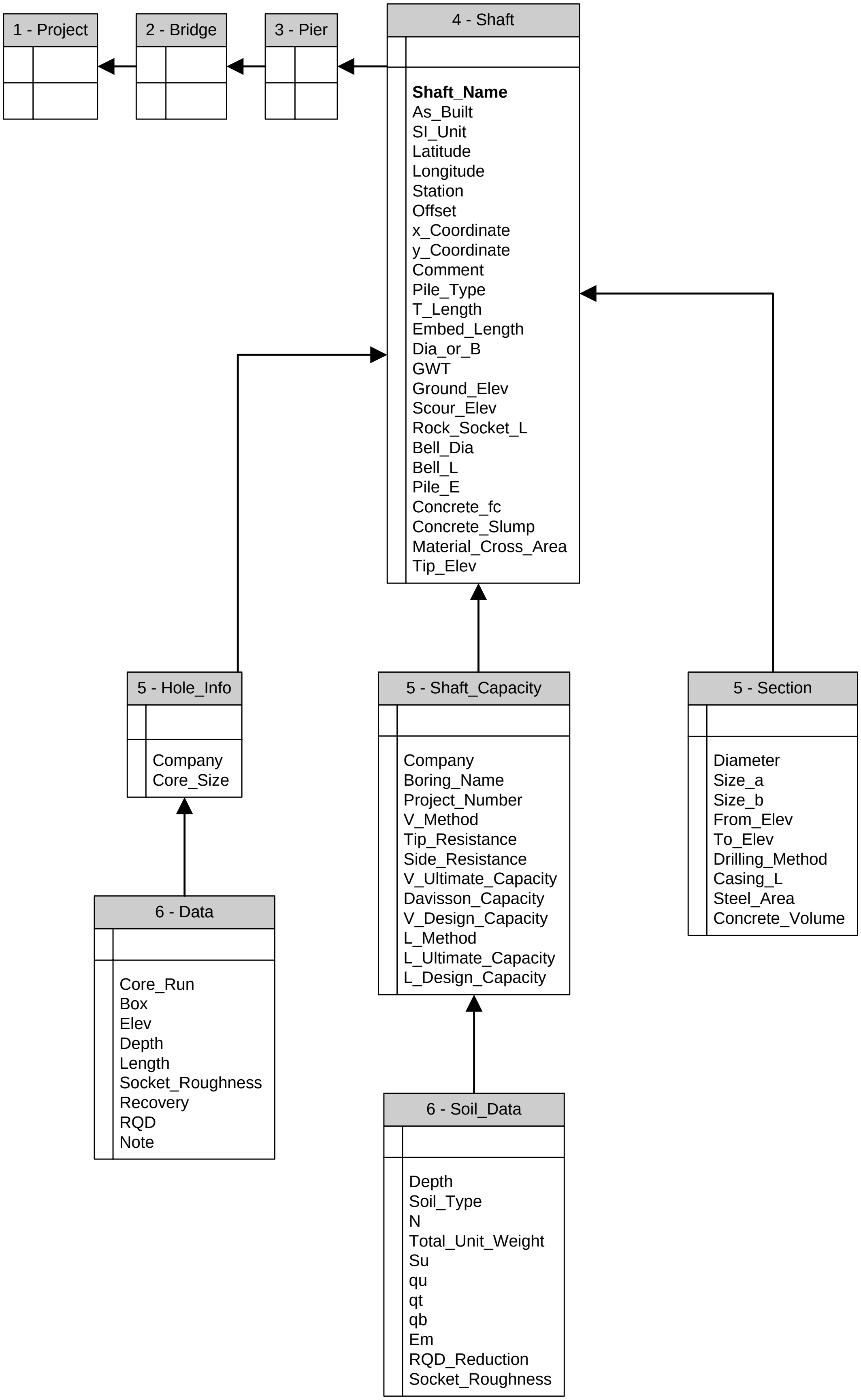

1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Driving_Info
Company
Contract_No
Inspector
Authorized_Length
Pile_Furnished
Pile_Driven
Manufact_By
Manufact_Work_Order
Manufact_Date_Cast
Manufact_Pile_No
Start_Time
Stop_Time
Weather
Temperature
Point_Protector
PDA
Extraction
Cut_Off
Code
Drive_Criteria
Date_Driving
Hammer_Type
Rated_Energy
Effective_Energy
Hammer_Weight
Pay_Item_No
Cut_Off_Elev
Min_Tip_Elev
Template_Elev
BM_No
BM_Elev
BM_Rod
HI_Elev
Pile_Top_Rod
Pile_Top_Elev
No_Redrive
Splices_Driven
Set_Check
Build_Up_Auth
Build_Up_Act
EDMS
TIMS
6 - Pile_Cushion
Material1
Thick1
Modulus1
Material2
Thick2
Modulus2
Coef_Restitution
6 - Log
Tip_Depth
L_Driven
N_Blows
Blows_Per_UnitL
Ram_Stroke
Hammer_Energy
Note
6 - Hammer_Cushion
Material1
Thick1
Modulus1
Material2
Thick2
Modulus2
Coef_Restitution

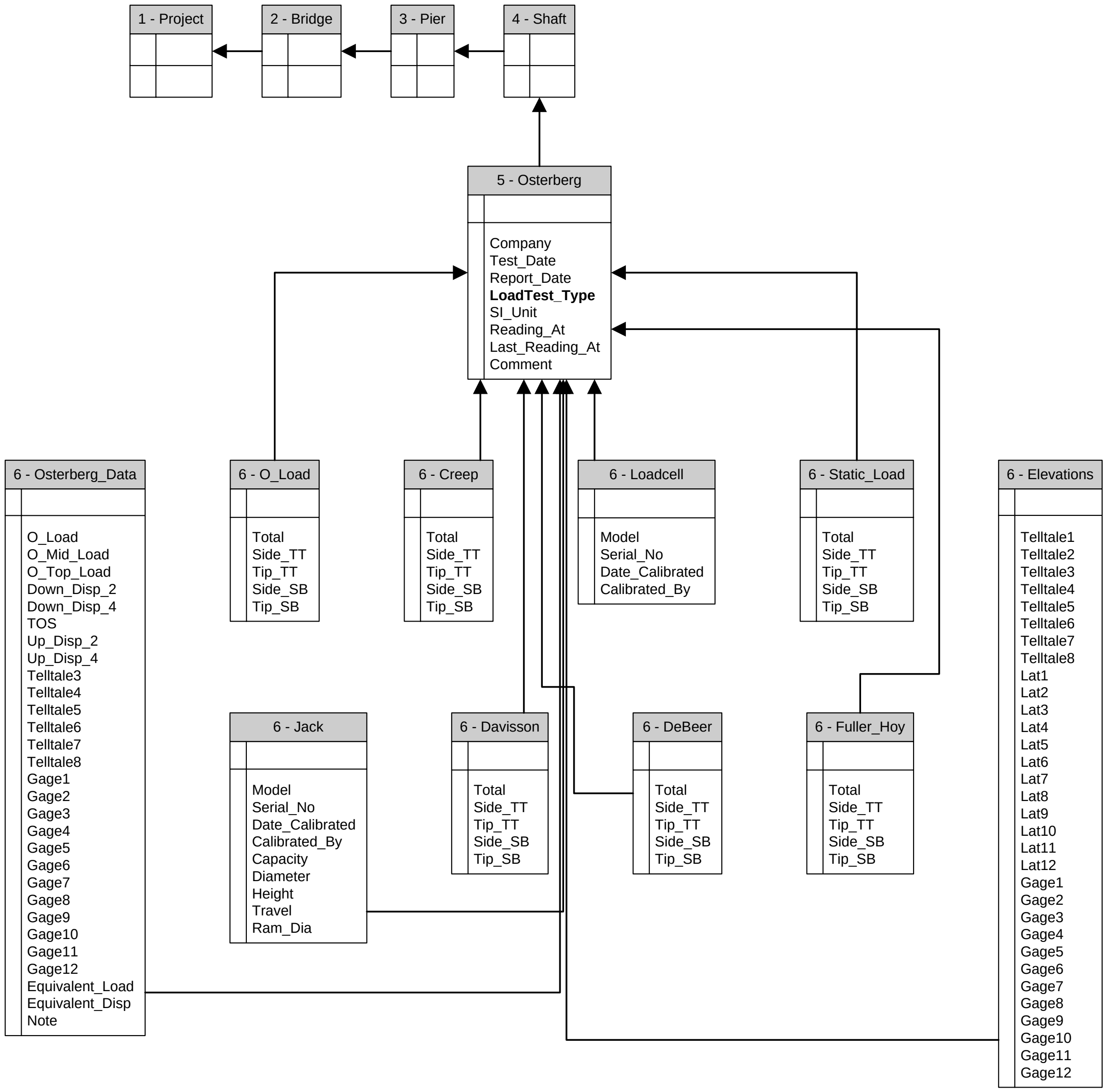

1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Driving_Analysis
SI_Unit
Table_Number
6 - Driving_Table
Start_Time
Wave_Speed
Hammer_Type
Sub_Table_Number
Comment
JC1
JC2
JC3
7 - Drv_Data
Blow_End
Depth
BLC
Type
CSX
CSI
TSX
EMX
BPM
BTA
RX1
RX2
RX3
RA2

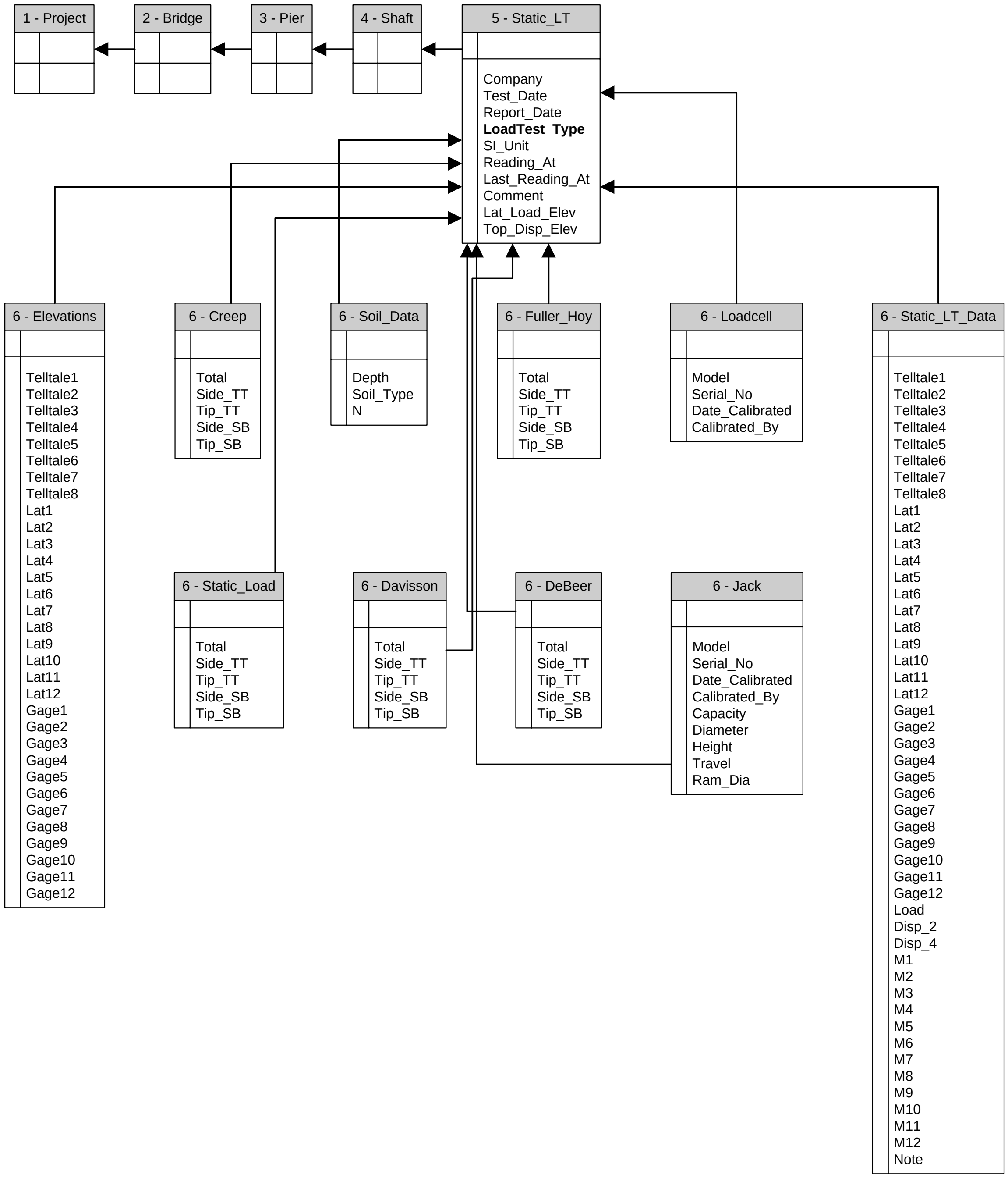

1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Osterberg
Company
Test_Date
Report_Date
LoadTest_Type
SI_Unit
Reading_At
Last_Reading_At
Comment
6 - Osterberg_Data
O_Load
O_Mid_Load
O_Top_Load
Down_Disp_2
Down_Disp_4
TOS
Up_Disp_2
Up_Disp_4
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
Equivalent_Load
Equivalent_Disp
Note
6 - DeBeer
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Fuller_Hoy
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Davisson
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Jack
Model
Serial_No
Date_Calibrated
Calibrated_By
Capacity
Diameter
Height
Travel
Ram_Dia
6 - Elevations
Telltale1
Telltale2
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
6 - Static_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - O_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Creep
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Loadcell
Model
Serial_No
Date_Calibrated
Calibrated_By

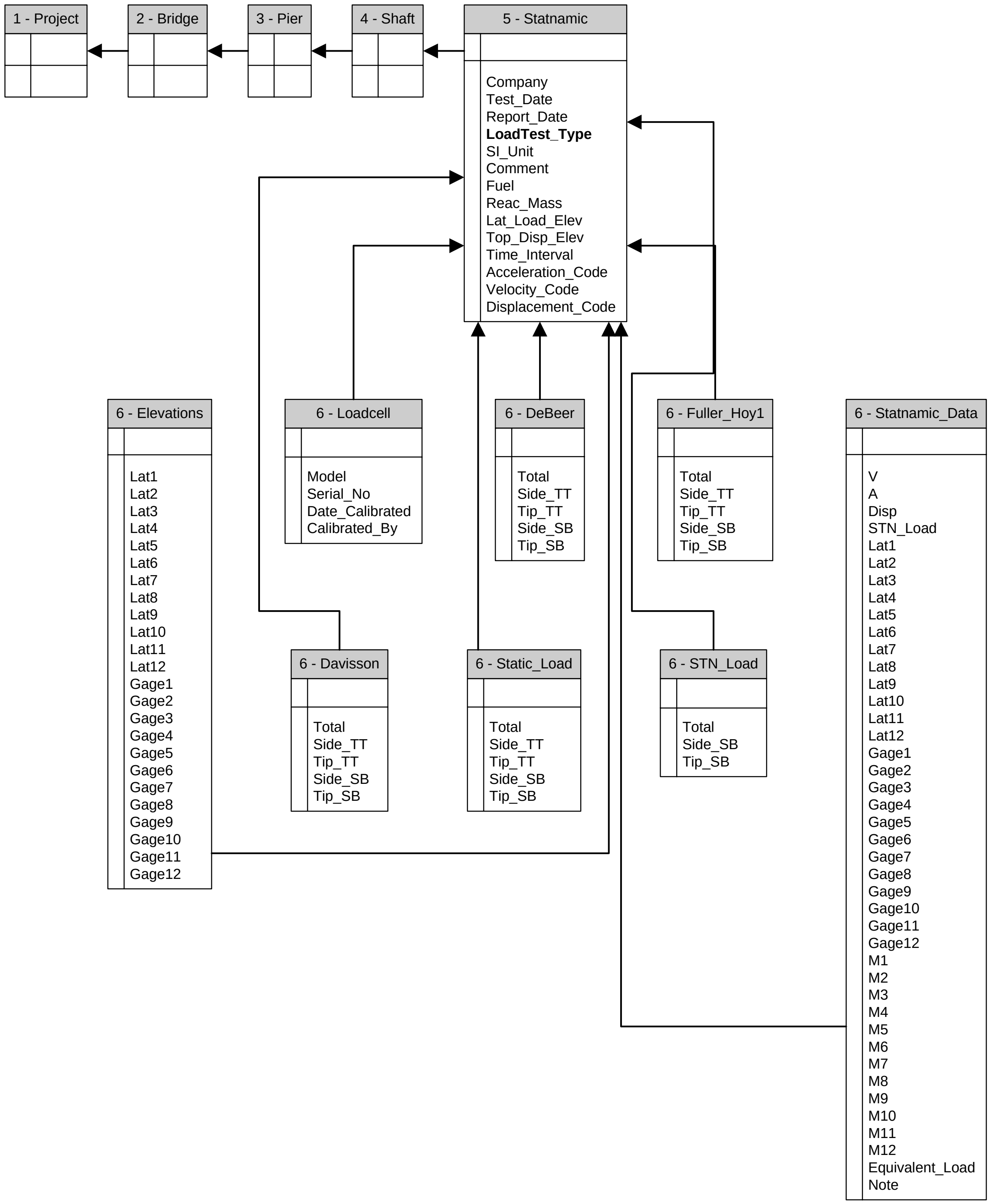

1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Pile_Capacity
Company
Hole_Name
Project_Number
V_Method
Tip_Resistance
Side_Resistance
V_Ultimate_Capacity
Davisson_Capacity
V_Design_Capacity
L_Method
L_Ultimate_Capacity
L_Design_Capacity
5 - Static_LT
Company
Test_Date
Report_Date
LoadTest_Type
SI_Unit
Reading_At
Last_Reading_At
Comment
Lat_Load_Elev
Top_Disp_Elev
6 - Elevations
Telltale1
Telltale2
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
6 - Static_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Davisson
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - DeBeer
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Creep
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Static_LT_Data
Telltale1
Telltale2
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
Load
Disp_2
Disp_4
M1
M2
M3
M4
M5
M6
M7
M8
M9
M10
M11
M12
Note
6 - Soil_Data
Depth
Soil_Type
N
6 - Fuller_Hoy
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Loadcell
Model
Serial_No
Date_Calibrated
Calibrated_By
6 - Jack
Model
Serial_No
Date_Calibrated
Calibrated_By
Capacity
Diameter
Height
Travel
Ram_Dia


1 - Project
2 - Bridge
3 - Pier
4 - Pile
5 - Statnamic
Company
Test_Date
Report_Date
LoadTest_Type
SI_Unit
Comment
Fuel
Reac_Mass
Lat_Load_Elev
Top_Disp_Elev
Time_Interval
Acceleration_Code
Velocity_Code
Displacement_Code
6 - Elevations
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
6 - Davisson
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Static_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - STN_Load
Total
Side_SB
Tip_SB
6 - Statnamic_Data
V
A
Disp
STN_Load
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
M1
M2
M3
M4
M5
M6
M7
M8
M9
M10
M11
M12
Equivalent_Load
Note
6 - Loadcell
Model
Serial_No
Date_Calibrated
Calibrated_By
6 - DeBeer
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Fuller_Hoy1
Total
Side_TT
Tip_TT
Side_SB
Tip_SB


1 - Project
2 - Bridge
3 - Pier
4 - Shaft
Shaft_Name
As_Built
SI_Unit
Latitude
Longitude
Station
Offset
x_Coordinate
y_Coordinate
Comment
Pile_Type
T_Length
Embed_Length
Dia_or_B
GWT
Ground_Elev
Scour_Elev
Rock_Socket_L
Bell_Dia
Bell_L
Pile_E
Concrete_fc
Concrete_Slump
Material_Cross_Area
Tip_Elev
5 - Hole_Info
Company
Core_Size
5 - Shaft_Capacity
Company
Boring_Name
Project_Number
V_Method
Tip_Resistance
Side_Resistance
V_Ultimate_Capacity
Davisson_Capacity
V_Design_Capacity
L_Method
L_Ultimate_Capacity
L_Design_Capacity
5 - Section
Diameter
Size_a
Size_b
From_Elev
To_Elev
Drilling_Method
Casing_L
Steel_Area
Concrete_Volume
6 - Data
Core_Run
Box
Elev
Depth
Length
Socket_Roughness
Recovery
RQD
Note
6 - Soil_Data
Depth
Soil_Type
N
Total_Unit_Weight
Su
qu
qt
qb
Em
RQD_Reduction
Socket_Roughness


1 - Project
2 - Bridge
3 - Pier
4 - Shaft
5 - Osterberg
Company
Test_Date
Report_Date
LoadTest_Type
SI_Unit
Reading_At
Last_Reading_At
Comment
6 - Osterberg_Data
O_Load
O_Mid_Load
O_Top_Load
Down_Disp_2
Down_Disp_4
TOS
Up_Disp_2
Up_Disp_4
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
Equivalent_Load
Equivalent_Disp
Note
6 - O_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Creep
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Loadcell
Model
Serial_No
Date_Calibrated
Calibrated_By
6 - Static_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Elevations
Telltale1
Telltale2
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
6 - Jack
Model
Serial_No
Date_Calibrated
Calibrated_By
Capacity
Diameter
Height
Travel
Ram_Dia
6 - Davisson
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - DeBeer
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Fuller_Hoy
Total
Side_TT
Tip_TT
Side_SB
Tip_SB


1 - Project
2 - Bridge
3 - Pier
4 - Shaft
5 - Static_LT
Company
Test_Date
Report_Date
LoadTest_Type
SI_Unit
Reading_At
Last_Reading_At
Comment
Lat_Load_Elev
Top_Disp_Elev
6 - Elevations
Telltale1
Telltale2
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
6 - Creep
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Soil_Data
Depth
Soil_Type
N
6 - Fuller_Hoy
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Loadcell
Model
Serial_No
Date_Calibrated
Calibrated_By
6 - Static_LT_Data
Telltale1
Telltale2
Telltale3
Telltale4
Telltale5
Telltale6
Telltale7
Telltale8
Lat1
Lat2
Lat3
Lat4
Lat5
Lat6
Lat7
Lat8
Lat9
Lat10
Lat11
Lat12
Gage1
Gage2
Gage3
Gage4
Gage5
Gage6
Gage7
Gage8
Gage9
Gage10
Gage11
Gage12
Load
Disp_2
Disp_4
M1
M2
M3
M4
M5
M6
M7
M8
M9
M10
M11
M12
Note
6 - Static_Load
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Davisson
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - DeBeer
Total
Side_TT
Tip_TT
Side_SB
Tip_SB
6 - Jack
Model
Serial_No
Date_Calibrated
Calibrated_By
Capacity
Diameter
Height
Travel
Ram_Dia


1 - Project
2 - Bridge
3 - Pier
4 - Shaft
5 - Statnamic
Company
Test_Date
Report_Date
LoadTest_Type
SI_Unit
Comment
Fuel
Reac_Mass
Lat_Load_Elev
Top_Disp_Elev
Time_Interval
Acceleration_Code
Velocity_Code
Displacement_Code
6 - Elevations
Lat1 Lat2 Lat3 Lat4 Lat5 Lat6 Lat7 Lat8 Lat9 Lat10 Lat11 Lat12
Gage1 Gage2 Gage3 Gage4 Gage5 Gage6 Gage7 Gage8 Gage9 Gage10 Gage11 Gage12
6 - Loadcell
Model
Serial_No
Date_Calibrated
Calibrated_By
6 - DeBeer
Total Side_TT Tip_TT Side_SB Tip_SB
6 - Fuller_Hoy1
Total Side_TT Tip_TT Side_SB Tip_SB
6 - Statnamic_Data
V A Disp STN_Load
Lat1 Lat2 Lat3 Lat4 Lat5 Lat6 Lat7 Lat8 Lat9 Lat10 Lat11 Lat12
Gage1 Gage2 Gage3 Gage4 Gage5 Gage6 Gage7 Gage8 Gage9 Gage10 Gage11 Gage12
M1 M2 M3 M4 M5 M6 M7 M8 M9 M10 M11 M12
Equivalent_Load
Note
6 - Davisson
Total Side_TT Tip_TT Side_SB Tip_SB
6 - Static_Load
Total Side_TT Tip_TT Side_SB Tip_SB
6 - STN_Load
Total Side_SB Tip_SB

APPENDIX B

XML SCHEMA FDOT DATABASE

Table 1: Main levels of XML tags:

Level	Name	relationship to prev level	detail see page
0	**GML**	**1**	
1	**Project**	**1-∞**	**5**
2	**Subsurface**	**1-1**	**5**
3	**Hole**	**1-∞**	5
4	SPT	1-1	5-6
4	CPT	1-1	6
4	DMT	1-1	6-7
4	PMT	**1-∞**	7
4	VST	1-1	7
4	Lab Rock	1-1	7
5	Core	1-∞	
6	Rock Specimen	1-∞	
4	Lab Soil	1-1	8
5	Soil Specimen	1-∞	
2	**Bridge**	1-∞	11
3	**Pier**	**1-∞**	**11**
4	**Pile**	**1-∞**	**11-12**
5	Pile_Capacity	1-1	12
5	Static_LT	1-∞	13-15
5	Osterberg	1-∞	15-16
5	Statnamic	1-∞	16-17
5	Driving_Info	1-1	17-18
5	Driving_Analysis	1-1	
6	Results_From_1_Blow	1-∞	19-21
6	Driving_Table	1-∞	22
4	**Shaft**	**1-∞**	**22-23**
5	Shaft_Capacity	1-1	23-24
5	Static_LT (same as in Pile)	1-∞	24
5	Osterberg (same as in Pile)	1-∞	24
5	Statnamic (same as in Pile)	1-∞	24
5	Hole_Info	1-1	24
6	Results_From_1_Blow (same as in Pile)	1-∞	24

Table 2: All levels of XML tags:

Level	Name	relationship to prev level
0	**GML**	**1**
1	**Project**	**1-∞**
2	**Subsurface**	**1-1**
3	**Hole**	**1-∞**
4	SPT	1-1
5	SPT_Data	1-∞
4	CPT	1-1
5	CPT_Data	1-∞
4	DMT	1-1
5	Zero_Reading	1-1
6	Zero_Data	1-∞
5	DMT_Data	1-∞
4	PMT	1-∞
5	PMT_Data	1-∞
4	VST	1-1
5	VST_Data	1-∞
4	Lab_Rock	1-1
5	Core	1-∞
6	Rock_Specimen	1-∞
7	Erosion	1-1
8	Erosion_Data	1-∞
7	Qu_Qt_Test	1-1
8	Data	1-∞
4	Lab_Soil	1-1
5	Soil_Specimen	1-∞
6	Sieve_Analysis	1-1
7	Sieve_Size	1-1
7	Percentage	1-1
6	Oedometer	1-1
7	Oed_Data	1-∞
6	Consolidation	1-∞
7	Consolidation_Data	1-∞
6	Triaxial	1-1
7	Test	1-∞
8	Test_Data	1-∞

2	**Bridge**	1-∞
3	**Pier**	**1-∞**
4	**Pile**	**1-∞**
5	**Pile_Capacity**	**1-1**
6	Soil_Data	1-∞
5	**Static_LT**	**1-∞**
6	Loadcell	1-1
6	Jack	1-1
6	Elevations	1-1
6	Static_Load	1-1
6	Davisson	1-1
6	DeBeer	1-1
6	Fuller_Hoy	1-1
6	Creep	1-1
6	Static_LT_Data	1-∞
5	**Osterberg**	**1-∞**
6	Loadcell (see Static_LT)	1-1
6	Jack (see Static_LT)	1-1
6	Elevations (see Static_LT)	1-1
6	O_Load	1-1
6	Static_Load (see Static_LT)	1-1
6	Davisson (see Static_LT)	1-1
6	DeBeer (see Static_LT)	1-1
6	Fuller_Hoy (see Static_LT)	1-1
6	Creep (see Static_LT)	1-1
6	Osterberg_Data	1-∞
5	**Statnamic**	**1-∞**
6	Loadcell (see Static_LT)	1-1
6	Elevations (see Static_LT)	1-1
6	STN_Load	1-1
6	Static_Load (see Static_LT)	1-1
6	Davisson (see Static_LT)	1-1
6	DeBeer (see Static_LT)	1-1
6	Fuller_Hoy (see Static_LT)	1-1
6	Statnamic_Data	1-∞
5	**Driving_Info**	**1-1**
6	Hammer_Cushion	**1-1**
6	Pile_Cushion	**1-1**
6	Log	**1-∞**
5	**Driving_Analysis**	**1-1**
6	**Results_From_1_Blow**	**1-∞**
7	Summary	1-1
7	Trace	1-1

8	Trace_Data	1-∞
7	Parameters	1-1
8	Case_RS1	1-1
8	Case_RMX	1-1
8	Case_RSU	1-1
8	Pile_Profile_Model	1-1
9	Prof_Data	1-∞
7	CAPWAP	1-1
8	CAPWAP_Data	1-∞
7	Extrema_Table	1-1
8	Extrema_Data	1-∞
8	Absolute_Max	1-1
8	Absolute_Min	1-1
6	**Driving_Table**	**1-∞**
7	Drv_Data	1-∞
4	**Shaft**	**1-∞**
5	**Section**	**1-∞**
5	**Shaft_Capacity**	**1-1**
6	Soil_Data	1-∞
5	**Static_LT** (same as in Pile)	**1-∞**
5	**Osterberg** (same as in Pile)	**1-∞**
5	**Statnamic** (same as in Pile)	**1-∞**
5	**Hole_Info**	**1-1**
6	Data	1-∞

Table 3: Detail of XML tags and attributes:

level	name	type		tag	attr	description	unit
0	**GML**		**1**	**x**			
1	**Project**		**1-∞**	**x**			
	Name	str			x	project name	
(id)	***Project_Number***	***str***			***x***	***project number***	
	Financial_Number	str			x		
	Comment	str			x		
	Base_Latitude	real			x	GPS coordinations of the beginning of project	
	Base_Longitude	real			x		
	Base_Station	str			x	Station & offset of the beginning of project;	
	Base_Offset	real			x	offset: - is to the left; + is to the right	
2	**Subsurface**		**1-1**	**x**			
3	**Hole**		**1-∞**	**x**			
	SI_Unit	boolean			x	either "true" or "false"	
	Hole_Name	***str***			***x***	***boring name (or #)***	
	Latitude	***real***			***x***	***GPS coordination of the hole***	
(id)	***Longitude***	***real***			***x***		
	Station	***str***			***x***	***Station and offset of the hole.***	
	Offset	***real***			***x***	***Offset: + if to the right, - if to the left***	
	GWT	real			x	ground water table elevation	ft, m
	Ground_Elev	real			x	ground elevation	ft, m
	x_Coordinate	real			x	user input coordination of the hole, in referenced to	ft, m
	y_Coordinate	real			x	begin of the project	ft, m
4	SPT		1-1	x		SPT table	
	Company	str			x	company that run SPT test	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
	SPT_Type	str			x	SPT equiment type	
	Eft_Energy_Ratio	real			x	effective energy ratio (e.g. 0.60)	
	Spoon_Inside	real			x	inside diameter of the SPT spoon	in, mm
	Spoon_Outside	real			x	outside diameter of the SPT spoon	in, mm
	Hammer_drop	real			x	the drop height of the SPT hammer	in, mm
	Hammer_weight	real			x	the weight of the SPT hammer	kip, kN
	Comment	str			x	any comment/ note on the SPT	
5	SPT_Data		1-∞	x		detail SPT log (sub-table)	
	Elev	real			x	elevation	ft, m
	Depth	real			x	depth	ft, m
	Sample	str			x	Sample type and number	
	N	real			x	N value	
	Interval	real			x	interval (usually 12 inch, however, when refusal, it may be less than 12 inch)	in, mm
	Color	str			x	the visual color of soil (e.g. brown)	

	Soil	str			x	the name of soil (e.g. Sand)	
	USCS	str			x	USCS classification (e.g. SP)	
	AASHTO	str			x	AASHTO classification (e.g. A-1-b)	
	Core_Length	real			x	core length, if rock is encountered	in, mm
	Note	str			x	any note at this depth	
4	CPT		1-1	x		CPT table	
	Company	str			x	company that run CPT test	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
	Ratio_a	real			x	pore pressure area ratio a ($q_t = q_c + u_2 * (1-a)$)	
	Ratio_b	real			x	pore pressure area ratio b ($f_t = f_s - u_2 * b$)	
	Cone_angle	real			x	cone angle of the equiment, usually 60^0	
	Rod_dia	real			x	cone rod diameter, usually 37.5 mm	in, mm
	Cone_dia	real			x	cone diameter, usually 37.5 mm	in, mm
	Comment	str			x	any comment/ note on the CPT	
5	CPT_Data		1-∞	x		detail CPT log (sub-table)	
	Elev	real			x	elevation	ft, m
	Depth	real			x	depth	ft, m
	qc	real			x	tip pressure	**tsf, MPa**
	q2	real			x	second cone tip pressure (in DCP)	**tsf, MPa**
	fs	real			x	sleeve friction	**tsf,** kPa
	u2	real			x	pore pressure	**tsf,** kPa
	u0	real			x	steady state pore pressure	**tsf, MPa**
	Note	str			x	any note at this depth	
4	DMT		1-1	x		DMT table	
	Company	str			x	company that run DMT test	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
	Comment	str			x	any comment/ note on the DMT	
5	Zero_Reading		1-1	x		zero reading calibration	
6	Zero_Data		1-∞	x			
	Depth	real			x	depth that the calibration is taken. If Depth=0, the calibration is taken before inserting the blade to ground	ft, m
	DA	real			x	ΔA	ksf, kPa
	DB	real			x	ΔB	ksf, kPa
	Zm	real			x	Z_m	ksf, kPa
5	DMT_Data		1-∞	x		detail DMT log (sub-table)	
	Elev	real			x	elevation	ft, m
	Depth	real			x	depth	ft, m
	Thrust	real			x	thrust	lbs, kN
	A	real			x	A reading	ksf, kPa
	B	real			x	B reading	ksf, kPa
	C	real			x	C reading	ksf, kPa

	Note	str			x	any note at this depth	
4	PMT		1-∞	x		PMT table	
	Company	str			x	company that run PMT test	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
	PMT_Type	str			x	type of PMT equiment (e.g. TEXAM)	
(id)	***Test_Depth***	***real***			***x***	***depth of the test point***	***ft, m***
	Comment	str			x	any comment/ note on the PMT	
	Description	str			x	Results interpreted from the PMT test:	
	Modulus	real			x	Modulus, at rest pressure p_0, limit pressure p_L,	ksf, kPa
	At_Rest_po	real			x	net limit pressure p^*_L	ksf, kPa
	Limit_pl	real			x		ksf, kPa
	Net_Limit_pl	real			x		ksf, kPa
5	PMT_Data		1-∞	x		detail DMT raw data	
	Volume	real			x	corrected volume injected	in^3, mm^3
	Pressure	real			x	corrected pressure	ksf, kPa
	Note	str			x	any note at this level	
4	VST		1-1	x		VST table	
	Company	str			x	company that run VST test	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
	VST_Type	str			x	type of VST equiment	
	Comment	str			x	any comment/ note on the VST	
5	VST_Data		1-∞	x		detail VST log (sub-table)	
	Elev	real			x	elevation	ft, m
	Depth	real			x	depth	ft, m
	Su	real			x	undrained shear strength (uncorrected)	ksf, kPa
	Sr	real			x	remolded shear strength (uncorrected)	ksf, kPa
	Muy	real			x	Brejum correction factor μ	
	Note	str			x	any note at this depth	

4	**Lab_Rock**		**1-1**	**x**			
	Company	str			x	company that compile rock table	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the rock is tested	
	Report_Date	date			x	date the test result is reported	
	Core_Size	real			x	core size	in, mm
5	Core		1-∞	x			
(id)	***Core_Run***	***str***			***x***	***Core run***	
	Length	real			x	Length of core	ft, m
	Recovery	real			x	recovery, number in % (eg. 65%, not 0.65)	
	RQD	real			x	RQD , number in %	

6	Rock_Specimen		1-∞	x			
(id)	***Sample***	***str***			***x***	***sample #***	
	Elev	sr			x	elevation from which the sample is taken. It may be in a range, such as “1.0 to 1.4”	ft, m
	Depth	str			x	depth (referenced to ground elevation) from which the sample is taken. It may be in a range, such as “1.0 to 1.4”	ft, m
	Dry_Unit_Weight	real			x	dry unit weight	pcf, kN/m3
	Description	str			x	rock description	
	qu	real			x	unconfined compressive strength	ksf, kPa
	qt	real			x	tensile strength	ksf, kPa
	E50	real			x	intact modulus, at ε_{50}	ksf, kPa
	E100	real			x	intact modulus, at ε_{100}	ksf, kPa
	E_Mass	real			x	mass modulus	ksf, kPa
	Epsilon100	real			x	strain ε_{100} , in decimal point (0.02) (not 2%)	
	Epsilon50	real			x	strain ε_{50}, in decimal point (0.02) (not 2%)	
7	Erosion		1-1	x			
	SI_Unit	boolean			x	either "true" or "false"	
	Company	str			x	company that run erosion test	
8	Erosion_Data		1-∞	x			
	Pressure	real			x	pressure in the erosion test	ksf, kPa
	Erosion_Rate	real			x	erosion rate in the erosion test	in/h, mm/h
7	Qu_Qt_Test		1-1	x			
	Test_Type	str			x	either "qu" or "qt"	
	SI_Unit	boolean			x	either "true" or "false"	
	Company	str			x	company that run q_u or q_t test	
	Start_Strain	real			x	the “zero” strain, before which the test data is not good	
8	Data		1-∞	x			
	Stress	real			x	stress	ksf, kPa
	Strain	real			x	strain (e.g. 0.02, not in % like 2%)	
4	**Lab_Soil**		**1-1**	**x**			
	SI_Unit	boolean			x	either "true" or "false"	
	Company	str			x	company that compile soil tables	
	Test_Date	date			x	date the soil is tested	
	Report_Date	date			x	date the test result is reported	
	Comment	str			x	note (remarks)	
5	**Soil_Specimen**		**1-∞**	x			
	Elev	sr			x	elevation from which the sample is taken. It may be in a range, such as “1.0 to 1.4”	ft, m
	Depth	str			x	depth (referenced to ground elevation) from which the sample is taken. It may be in a range, such as “1.0 to 1.4”	ft, m
(id)	***Sample***	***str***			***x***	***Sample #***	
	Color	str			x	visual color of the soil (e.g. Brown)	
	Soil	str			x	description of the soil (e.g. Sand)	
	USCS	str			x	USCS classification of soil	
	AASHTO	str			x	AASHTO classification of soil	

	Organic_Content	real			x	organic content, number in % (e.g. 6%, not 0.06)	
	fi	real			x	effective internal friction angle	
	c	real			x	effective cohesion	ksf, kPa
	Su	real			x	undrained shear strength (for cohesive soil)	ksf, kPa
	OCR	real			x	Over Consolidation Ratio	
	Total_Unit_Weight	real			x	total unit weight	pcf, kN/m3
	Dry_Unit_Weight	real			x	dry unit weight	pcf, kN/m3
	Permeability_k	real			x	coeficient of permeability	ft/s, m/s
	Void_Ratio	real			x	void ratio e	
	Moisture_Content	real			x	moisture content w, number in % (e.g 18.3%, not 0.183)	
	LL	real			x	Atterberg limits, number in % (e.g. 25.3)	
	PL	real			x		
	PI	real			x		
	Dr	real			x	Relative density, number in% (e.g. 67.5, not 0.675)	
	Optimum_Moisture_Content	real			x	optimum moisture content (from Proctor test), number in % (e.g 18.3)	
	Max_Dry_Unit_Weight	real			x	maximum dry unit weight (from Proctor test)	pcf, kN/m3
	Cc	real			x	Compression index C_c (result from the test)	
	Cr	real			x	Recompression index C_r (result from the test)	
	Pre_Consolidation_Stress	real			x	Pre consolidation stress σ'_p (result from the test)	ksf, kPa
	Constraint_Modulus	real			x	Constraint modulus, from Oedometer test	ksf, kPa
	E50	real			x	Triaxial Modulus E_{50} at strain of ε_{50}	ksf, kPa
	E100	real			x	Triaxial Modulus E_{100} at strain of ε_{100}	ksf, kPa
	Epsilon50	real			x	strain ε_{50}, in decimal point (0.02) (not 2%)	
	Epsilon100	real			x	strain ε_{100} , in decimal point (0.02) (not 2%)	
6	**Sieve_Analysis**		**1-1**	**x**			
	Company	str			x	company that run sieve analysis	
	Test_Date	date			x	date the sieve analysis is taken	
	Description	str			x	Visual description of the soil (e.g. yellow silty sand)	
7	Sieve_Size		1-1	x			
	Size1	real			x	Size of different sieve (Size1 is biggest) eg. Size7="0.074" (means #200) Size8="0" (means pan) Size9="" (nothing)	always in mm
	Size2	real			x		
	...						
	...						
	Size11	real			x		
	Size12	real			x		
7	Percentage		1-1	x			
	On_Sieve1	real			x	Percent of mass retain on each sieve, number in % (e.g. 17.5%, not 0.175)	
	On_Sieve2	real			x		
	....						
	.....						
	On_Sieve11	real			x		
	On_Sieve12	real			x		

6	**Oedometer**		**1-1**	**x**			
	Company	str			x	company that run oedometer & consolidation test	
	SI_Unit	boolean			x	either "true" or "false"	
	Test_Date	date			x	date the oedometer test is taken	
	Report_Date	date			x	date the result is reported	
	Operator	str			x	operator	
	Sample_Height	real			x	sample height	in, mm
	Solid_Height	real			x	solid height	in, mm
	Insitu_Stress	real			x	Insitu effective stress σ'_{vo}	ksf, kPa
7	Oed_Data		1-∞	x			
	Pressure	real			x	pressure	ksf, kPa
	Disp	real			x	displacement	in, mm
6	**Consolidation**		**1-∞**	**x**			
	Test_Date	date			x	date the consolidation test is taken	
	Report_Date	date			x	date the result is reported	
	Operator	str			x	operator	
(id)	***Pressure***	***real***			***x***	***pressure at which the consolidation test is carried***	***ksf, kPa***
	Taylor_Cv	real			x	coef. of consolidation Cv, interpreted by Taylor method	ft^2/day,
	Casagrande_Cv	real			x	coef. of cons. Cv, interpreted by Casagrande method	m^2/day
7	**Consolidation_Data**		**1-∞**	**x**			
	Time	real			x	time	minutes
	Dial_Reading	real			x	dial reading	
6	**Triaxial**		**1-1**	**x**		**Triaxial table**	
	SI_Unit	boolean			x	either "true" or "false"	
	Company	str			x	company that runs triaxial test	
	Test_Type	str			x	UU, CU or CD	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the result is reported	
	Operator	str			x	operator	
	c_Total_Stress	real			x	cohesion (total stress parameters)	ksf, kPa
	fi _Total_Stress	real			x	friction angle (total stress parameters)	
7	Test		1-∞	x		Detail of each triaxial test parameters & results	
(id)	***Test_Number***	***integer***			***x***	***Test_Number***	
	Strain_Rate	real			x	Strain_Rate	
	Back_Pressure	real			x	Back_Pressure	ksf, kPa
	B_Value	real			x	B_Value	
	Diameter	real			x	initial diameter of sample	in, mm
	Ac	real			x	area after consolidation. If UU, Ac is initial area	in2, mm2
	Hc	real			x	height after consolidation. If UU, Ac is initial height	in, mm
	Uplift_Force	real			x	uplift force due to the chamber pressure to the piston	lb, kN
	Load_Cell	real			x	vertical load when the sample fails	lb, kN
	Disp	real			x	vertical displacement when the sample fails	in, mm
	Pore_Pressure	real			x	pore pressure when the sample fails	ksf, kPa
	V_Strain	real			x	volumetric strain when the sample fails. V_Strain="0" for UU and CU tests	

	Pressure_3	real			x	σ_3 when the sample fails	ksf, kPa
	Effective_Load	real			x	corrected vertical load when the sample fails Effective_Load=Load_Cell - Uplift_Force	lb, kN
	Strain	real			x	Axial strain when the sample fails (e.g. 0.003, not 0.3%)	
	Corrected_Area	real			x	cross area of the sample when the it fails	in2, mm2
	Devia_Stress	real			x	deviator stress σ'_d when the sample fails	ksf, kPa
8	Test_Data		1-∞	x		Sub table detailing the test data points	
	Devia_Stress	real			x	deviator stress σ'_d	ksf, kPa
	Strain	real			x	axial strain (e.g. 0.003, not 0.3%)	
2	**Bridge**		1-∞	x			
	Bridge_Name	str			x	bridge name	
	Financial_Number	str			x	financial number	
(id)	***Bridge_Number***	***str***			***x***	***bridge number***	
	County	str			x	county	
	State	str			x	state	
	Location	str			x	specific location	
	B_Latitude	real			x	GPS coordinations of the beginning of bridge	
	B_Longitude	real			x		
	B_Station	str			x	Station & offset of the beginning of bridge;	
	B_Offset	real			x	offset: - is to the left; + is to the right	
	Comment	str			x	any note generally about the bridge	
	Client	str			x		
	Consultant	str			x	the consultant	
	Contractor	str			x	the contractor building the bridge	
	Geo_Subcontractor	str			x	the sub-contractor doing geotechnical engineering	
	Pile_Test_Company	str			x	the sub-contractor doing pile testing	
3	**Pier**		**1-∞**	**x**		Pier, or Bent, or Wall, or Pile-Group	
(id)	***Pier_Name***	***str***			***x***	***name of Pier, Bent, Wall or Pile-Group***	
4	**Pile**		**1-∞**	**x**			
(id)	***Pile_Name***	***str***			***x***	***name of pile (or pile number)***	
	As_Built	boolean			x	either "true" or "false"	
	SI_Unit	boolean			x	either "true" or "false"	
	Pile_Type	integer			x	1 - concrete; 2 - composite; 3 - H steel; 4 - pipe steel	
	Description	str	for concrete or composite: "square" or "circular" for H: 14x117, 14x102, 14x89, 14x73, 13x100, 13x87, 13x73, 13x60, 12x84, 12x74, 12x63, 12x53, 10x57, 10x42, 8x36 If SI_Unit="true" then: 360x174, 360x152, 360x132, 360x108, 330x149, 330x129, 330x110, 330x89, 310x125, 310x110, 310x93, 310x79, 250x85, 250x62, 200x53 for Pipe: "opened end" or "closed end"				
	Latitude	real			x	GPS coordination of the pile	
	Longitude	real			x		
	Station	str			x	Station and offset of the pile.	
	Offset	real			x	Offset: + if to the right, - if to the left	
	x_Coordinate	real			x	user input coordination of the pile, in referenced to	ft, m

	y_Coordinate	real			x	begin of the bridge	
	Comment	str				any note about the pile	ft, m
	T_Length	real			x	Total pile length	ft, m
	Embed_Length	real			x	embedded pile length (also known as "Penetration pile length" when referred to the pile in driving process	ft, m
	GWT	real			x	ground water table	ft, m
	Ground_Elev	real			x	ground elevation	ft, m
	Scour_Elev	real			x	scour elevation	ft, m
	Excavation_Elev	real			x	excavation elevation	ft, m
	Driving_Elev_Code	integer			x	code to refer to the ground when driving: 1 – ground elevation; 2 – scour; 3 - excavation	
	Dia_or_B	real			x	Diameter of pile (if circular), or size (if square)	in, mm
	Void	real			x	void diameter of the pile	in, mm
	Thick	real			x	thickness of steel pipe pile	in, mm
	N_Slices	integer			x	number of slices	
	Prebored_Depth	real			x	depth of the prebored (predrilled) hole	ft, m
	Auger_Dia	real			x	diameter of the auger in case of prebored	in, mm
	Jetting_Depth	real			x	Jetting depth	ft, m
	Batter_Ratio	real			x	batter ratio	
	Concrete_str	real			x	concrete strength f'_c	ksi, MPa
	Pile_E	real			x	Pile modulus	ksi, MPa
	Pile_Cross_Area	real			x	Cross area, supposed the pile is pluged (H or pipe)	in^2, mm^2
	Material_Cross_Area	real			x	Cross area of the pile material (plug is not accounted)	in^2, mm^2
	Material_Weight	real			x	the weight of the pile material	kip, kN
	Tip_Elev	real			x	Tip elevation	ft, m
	N_Strands	integer				Number of bar/ strands of steel in concrete pile	
	Bar_Area	real			x	area of 1 bar/ 1 strand of steel in concrete pile	in^2 , mm^2
	Prestressed	real			x	prestress after losses	ksi, MPa
5	**Pile_Capacity**		**1-1**	**x**			
	Company	str			x	Company that does pile capacity prediction & pile design	
	Hole_Name	str			x	The name (#) of the boring that is used to predict pile capacity	
	Project_Number	str			x	the project number that this boring is taken from. This value can be blank. In that case, this boring is taken from the current project	
	V_Method	str			x	method of vertical capacity prediction (e.g. Schmertmann)	
	Tip_Resistance	real			x	tip ultimate resistance	kip, kN
	Side_Resistance	real			x	side ultimate resistance	kip, kN
	V_Ultimate_Capacity	real			x	vertical ultimate capacity	kip, kN
	Davisson_Capacity	real			x	Davisson capacity	kip, kN
	V_Design_Capacity	real			x	Vertical design capacity	kip, kN
	L_Method	str			x	method of lateral capacity prediction (e.g. FB-Pier)	
	L_Ultimate_Capacity	real			x	lateral ultimate capacity	kip, kN
	L_Design_Capacity	real			x	lateral design capacity	kip, kN
6	Soil_Data		1-∞	x			
	Depth	real			x	depth	ft, m
	Soil_Type	integer			x	1 - Clay; 2 - Silt, or Mix; 3 - Sand; 4 - Limestone; 5 – Void	

	N	real			x	N Value	
5	**Static_LT**		**1-∞**	**x**		**Conventional load test Table**	
	Company	str			x	Company that run conventional load test	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
(id)	***LoadTest_Type***	***integer***			***x***	1 - compression cycle 1; 2- cycle 2; 3 - cycle 3 4 - tension cycle 1; 5- cycle 2; 6 - cycle 3 7 - lateral cycle 1; 8- cycle 2; 9 - cycle 3	
	SI_Unit	boolean			x	either "true" or "false"	
	Reading_At	real			x	The times at which results are recorded. Example: for Quick Load test, Reading_At="2" Last_Reading_At="4"	minutes
	Last_Reading_At	real			x		
	Comment	str			x	Note, or remarks on loadtest	
	Lat_Load_Elev	real			x	Elevation of lateral load (lateral load test)	ft, m
	Top_Disp_Elev	real			x	Elevation at the point that top displacement is measured	ft, m
6	Loadcell		1-1	x		**Sub-table about the type of load cell (used to measure load)**	
	Model	str			x	model of load cell	
	Serial_No	str			x	serial number(s) of load cell(s)	
	Date_Calibrated	date			x	date load cell is calibrated	
	Calibrated_By	str			x	person who do the calibration	
6	Jack		1-1	x		**Sub-table about the type of load test jack used**	
	Model	str			x	model	
	Serial_No	str			x	serial number(s) of jack(s)	
	Date_Calibrated	date			x	date calibrated	
	Calibrated_By	str			x	person who do the calibration	
	Capacity	real			x	capacity of the jack	kip, KN
	Diameter	real			x	diameter of the jack	in, mm
	Height	real			x	height of the jack	in, mm
	Travel	real			x	maximum travel distance	in, mm
	Ram_Dia	real			x	ram diameter of the jack	in, mm
6	Elevations		1-1	x		**Sub-table about elevation of telltales and strain gages**	
	Telltale1	real			x	elevations of telltales, level 1 is closest to tip of pile	ft, m
	Telltale2	real			x		
	...	real			x		
	Telltale8	real			x		
	Lat1	real			x	elevations of instrumentations (may be accelerometers, or inclinometer sensors, etc.) to get lateral displacements in lateral load tests.	ft, m
	Lat2	real			x		
	...	real			x		
	...	real			x		
	...	real			x		
	Lat9	real			x		
	Lat10	real			x		
	Lat11	real			x		
	Lat12	real			x		
	Gage1	real			x		

	Gage2	real			x	elevations of strain gages (sister bars) level 1 is closest to tip of pile;	ft, m
	...	real			x		
	...	real			x		
	...	real			x		
	Gage11	real			x		
	Gage12	real			x		
6	Static_Load		1-1	x		**Sub-table about maximum static load**	
	Total	real			x	max total static load	kip, KN
	Side_TT	real			x	max side resistance, reduced by using telltales data	
	Tip_TT	real			x	max tip resistance, reduced by using telltales data	
	Side_SB	real			x	max side resistance, reduced by using sister bar data	
	Tip_SB	real			x	max tip resistance, reduced by using sister bar data	
6	Davisson		1-1	x		**Sub-table about interpreted Davisson capacity**	
	Total	real			x	total Davission capacity	kip, KN
	Side_TT	real			x	side capacity, reduced by using telltales data	
	Tip_TT	real			x	tip capacity, reduced by using telltales data	
	Side_SB	real			x	side capacity, reduced by using sister bar data	
	Tip_SB	real			x	tip capacity, reduced by using sister bar data	
6	DeBeer		1-1	x		**Sub-table about interpreted DeBeer capacity**	
	Total	real			x	total DeBeer capacity	kip, KN
	Side_TT	real			x	side capacity, reduced by using telltales data	
	Tip_TT	real			x	tip capacity, reduced by using telltales data	
	Side_SB	real			x	side capacity, reduced by using sister bar data	
	Tip_SB	real			x	tip capacity, reduced by using sister bar data	
6	Fuller_Hoy		1-1	x		**Sub-table about interpreted Fuller_Hoy capacity**	
	Total	real			x	total Fuller_Hoy capacity	kip, KN
	Side_TT	real			x	side capacity, reduced by using telltales data	
	Tip_TT	real			x	tip capacity, reduced by using telltales data	
	Side_SB	real			x	side capacity, reduced by using sister bar data	
	Tip_SB	real			x	tip capacity, reduced by using sister bar data	
6	Creep		1-1	x		**Sub-table about creep load**	
	Total	real			x	total creep load	kip, KN
	Side_TT	real			x	side creep, reduced by using telltales data	
	Tip_TT	real			x	tip creep, reduced by using telltales data	
	Side_SB	real			x	side creep, reduced by using sister bar data	
	Tip_SB	real			x	tip creep, reduced by using sister bar data	
6	Static_LT_Data		1-∞	x		**Sub-table detailing the load test data**	
	Load	real			x	Load (imposed from the jack)	kip, KN
	Disp_2	real			x	displacements at the last 2 readings (e.g. at 2 minutes and 4 minutes in Quick Load Tests)	in, mm
	Disp_4	real			x		
	Telltale1	real			x	displacements recorded at the telltale levels (telltales readings) (these displacements have to be corrected for *Top of Shaft* movement, if the telltale readings are refered to the top of shaft	in, mm
	Telltale2	real			x		
	....	real			x		
	Telltale8	real			x		
	Lat1	real			x		

	Lat2	real			x		
	Lat3	real			x	Lateral displacements got (or interpreted) from inclinometers or accelerometers, etc.	in, mm
	Lat4	real			x		
	...	real			x		
	...	real			x		
	...	real			x		
	Lat11	real			x		
	Lat12	real			x		
	Gage1	real			x		
	Gage2	real			x		
	Gage3	real			x		
	...	real			x	In vertical load test: these are interpreted vertical loads transfered at strain gages level	kip, kN
	...	real			x		
	...	real			x		
	Gage10	real			x		
	Gage11	real			x		
	Gage12	real			x		
	M1	real			x		
	M2	real			x		
	...	real			x	In lateral load test: these are interpreted bending moments in piles/shafts.	kipft, kNm
	...	real			x		
	M11	real			x		
	M12	real			x		
	Note	str			x	Note at this level of load	
5	**Osterberg**		**1-∞**	**x**		**Osterberg load test Table**	
	Company	str			x	Company that run Osterberg load test	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
(id)	***LoadTest_Type***	***integer***			***x***	**This can be cycle number, or stage number (in case of multiple levels test). But this load test ID must be unique: In 1 pile, there can't be 2 same values for LoadTest_Type.** **If you provide duplicate LoadTest_Type for 1 pile (or 1 shaft), the whole load test data will be overwritten.**	
	SI_Unit	boolean			x	either "true" or "false"	
	Reading_At	real			x	The times at which results are recorded. E.g. for Quick Load test, Reading_At="2" Last_Reading_At="4"	minutes
	Last_Reading_At	real			x		
	Comment	str			x	Note, or remarks on loadtest	
6	Loadcell		1-1	x		Same as in Static_LT	
6	Jack		1-1	x			
6	Elevations		1-1	x		Same as in Static_LT (without Lat elevations)	
6	O_Load		1-1	x			
	Total	real			x	max total O-cell load	
	Side_TT	real			x	max O-cell load on side, reduced by using telltales data	
	Tip_TT	real			x	max O-cell load on tip, reduced by using telltales data	kip, KN
	Side_SB	real			x	max O-cell load on side, reduced by using sister bars data	
	Tip_SB	real			x	max O-cell load on tip, reduced by using sister bars data	

6	Static_Load		1-1	x		Same as in Static_LT, only that this Static load is the derived static load (or equivalent top-down load)	
6	Davisson		1-1	x		Same as in Static_LT	
6	DeBeer		1-1	x			
6	Fuller_Hoy		1-1	x			
6	Creep		1-1	x			
6	Osterberg_Data		1-∞	x		**Sub-table detailing the load test data**	
	O_Load	real			x	Load imposed from the bottom O-cell(s) or jack(s)	kip, KN
	O_Mid_Load	real			x	Load imposed from the middle O-cell(s) or jack(s)	kip, KN
	O_Top_Load	real			x	Load imposed from the top O-cell(s) or jack(s)	kip, KN
	Down_Disp_2	real			x	downward displacements at bottom of O-cell level, recorded at 2 & 4 minutes; *(If reading refered to top of shaft, then it is corrected as: Down_Disp = BP - TOS BP: reading recorded at telltale1 (bottom of O-cell) TOS: top of shaft movement)*	in, mm
	Down_Disp_4	real			x		
	TOS	real			x	Top of shaft movement (referd to reference beam)	in, mm
	Up_Disp_2	real			x	upward displacements at top of last O-cell level, recorded at 2 & 4 minutes; *(If reading refered to top of shaft, then it is corrected as: Up_Disp=COMP + TOS COMP: reading recorded at telltale2 (top of last O-cell))*	in, mm
	Up_Disp_4	real			x		
	Telltale3	real			x	Absolute displacements recorded at the telltale levels. If telltales readings refered to top of shaft, then these numbers have to be corrected as: Telltalei = readingi ± TOS	in, mm
	...	real			x		
	...	real			x		
	Telltale8	real			x		
Note:	Down_Disp_4 is Telltale1 Up_Disp_4 is Telltale2. All readings, except Down_Disp_2 and Up_Disp_2, are at last time step (e.g. 4 minutes)						
	Gage1	real			x	Load transfer at strain gages level	kip, KN
	Gage2	real			x		
	Gage3	real			x		
	...	real			x		
	...	real			x		
	...	real			x		
	Gage10	real			x		
	Gage11	real			x		
	Gage12	real			x		
	Equivalent_Load	real			x	Derived static load as in top-down conventional test	kip, KN
	Equivalent_Disp	real			x	derived displacement as in top-down test	in, mm
	Note	str			x	Note at this level of load	
5	**Statnamic**		**1-∞**	**x**		**Statnamic load test Table**	
	Company	str			x	Company that run Statnamic load test	
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
(id)	***LoadTest_Type***	***integer***			***x***	1 - compression cycle 1; 2- cycle 2; 3 - cycle 3 4 - tension cycle 1; 5- cycle 2; 6 - cycle 3 7 - lateral cycle 1; 8- cycle 2; 9 - cycle 3	
	SI_Unit	boolean			x	either "true" or "false"	

	Comment	str			x	Note, or remarks on loadtest	
	Fuel	real			x	fuel	lb, kg
	Reac_Mass	real			x	reaction mass	lb, kg
	Lat_Load_Elev	real			x	Elevation of lateral load (lateral load test)	ft, m
	Top_Disp_Elev	real			x	Elevation at the point that top displacement is measured	ft, m
	Time_Interval	real			x	time interval between readings	ms
	Acceleration_Code	integer			x	1 - measured data will be used for acceleration 2 - differentiated	
	Velocity_Code	integer			x	1 - integrated; 2 - differentiated	
	Displacement_Code	integer			x	1 - measured; 2 - integrated	
6	Loadcell		1-1	x		Same as in Static_LT	
6	Elevations		1-1	x		Similar as in Static_LT (without Telltales elevation)	
6	STN_Load		1-1	x		**Sub-table about maximum STN load imposed**	
	Total	real			x	max total STN load	kip, kN
	Side_SB	real			x	max STN load on side, reduced by using sister bars data	
	Tip_SB	real			x	max STN load on tip, reduced by using sister bars data	
6	Static_Load		1-1	x		Same as in Static_LT, only that this Static load is the derived static load (or equivalent top-down load)	
6	Davisson		1-1	x		Same as in Static_LT	
6	DeBeer		1-1	x			
6	Fuller_Hoy		1-1	x			
6	Statnamic_Data		1-∞	x		**Sub-table detailing the load test data**	
	V	real			x	velocity	ft/s, m/s
	A	real			x	acceleration	ft^2/s, m^2/s
	Disp	real			x	displacement	in, mm
	STN_Load	real			x	Statnamic Load	kip, KN
	Lat1	real			x	Lateral displacement got (or interpreted) from inclinometers or accelerometers, etc.	in, mm
	Lat2	real			x		
	...	real			x		
	...	real			x		
	Lat12	real			x		
	Gage1	real			x	In vertical load test: these are interpreted vertical loads transfered at strain gage levels	kip, kN
	Gage2	real			x		
	...	real			x		
	...	real			x		
	...	real			x		
	Gage11	real			x		
	Gage12	real			x		
	M1	real			x	In lateral load test: these are interpreted bending moment in piles/shafts at strain gage levels	kipft, kNm
	M2	real			x		
	...	real			x		
	...	real			x		
	M12	real			x		
	Equivalent_Load	real			x	Derived static load as in conventional test	kip, KN
	Note	str			x	Note at this level of load	

5	**Driving_Info**		**1-1**	**x**			
	Company	str			x	Company that do pile driving	
	Contract_No	str			x	contract number	
	Inspector	str			x	Inspector	
	Authorized_Length	real			x	Authorized Length	ft, m
	Pile_Furnished	real			x	Length of Pile Furnished	ft, m
	Pile_Driven	real			x	Length of Pile Driven	ft, m
	Manufact_By	str			x	Manufacturer Company	
	Manufact_Work_Order	str			x	Manufacturer Work Order	
	Manufact_Date_Cast	date			x	Manufacturer Date Cast	
	Manufact_Pile_No	str			x	Manufacturer Pile Number	
	Start_Time	time			x	Start_Time	
	Stop_Time	time			x	StopTime	
	Weather	str			x	Weather	
	Temperature	real			x	Temperature	
	Point_Protector	boolean			x	either "true" or "false"	
	PDA	boolean			x	either "true" or "false"	
	Extraction	boolean			x	either "true" or "false"	
	Cut_Off_Code	boolean			x	either "true" or "false"	
	Drive_Criteria	str			x	driving criteria	
	Date_Driving	date			x	date driving	
	Hammer_Type	str			x	hammer type	
	Rated_Energy	real			x	Rated Energy	kip.ft,
	Effective_Energy	real			x	Effective Energy	kN.m
	Hammer_Weight	real			x	Hammer Weight	kip, kN
	Pay_Item_No	str			x	Pay Item Number	
	Cut_Off_Elev	real			x	Cut Off Elevation	ft, m
	Min_Tip_Elev	real			x	Minimum Tip Elevation	ft, m
	Template_Elev	real			x	Template Elevation	ft, m
	BM_No	str			x	BM Number	
	BM_Elev	real			x	BM Elevation	ft, m
	BM_Rod	real			x	BM Rod	
	HI_Elev	real			x	HI Elevation	ft, m
	Pile_Top_Rod	real			x	Pile Top Rod read	
	Pile_Top_Elev	real			x	Pile Top Elevation	ft, m
	No_Redrive	integer			x	Number of Redrives	
	Splices_Driven	integer			x	Splices that are driven	
	Set_Check	integer			x	Set check paid for	
	Build_Up_Auth	real			x	Build-Up Authorized	ft, m
	Build_Up_Act	real			x	Build-Up Actual	ft, m
	EDMS	str			x	reference to FDOT EDMS database	
	TIMS	str			x	reference to FDOT TIMS database	
6	**Hammer_Cushion**		**1-1**	**x**			
	Material1	str			x	the name of material 1 for hammer cushion	
	Thick1	real			x	thickness	in, mm

	Modulus1	real			x	modulus	ksi, MPa
	Material2	str			x	if hammer cushion is composite, then	
	Thick2	real			x	these lines are for material 2	in, mm
	Modulus2	real			x		ksi, MPa
	Coef_Restitution	real			x	coef. of restitution	
6	**Pile_Cushion**		**1-1**	**x**			
	Material1	str			x	the name of material 1 for pile cushion	
	Thick1	real			x	thickness	in, mm
	Modulus1	real			x	modulus	ksi, MPa
	Material2	str			x	if pile cushion is composite, then	
	Thick2	real			x	these lines are for material 2	in, mm
	Modulus2	real			x		ksi, MPa
	Coef_Restitution	real			x	coef. of restitution	
6	**Log**		**1-∞**	**x**		**driving log sub-table**	
	Tip_Depth	real			x	depth	ft, m
	L_Driven	real			x	length driven	ft, m
	N_Blows	real			x	number of blows for that driven length	
	Blows_Per_UnitL	real			x	blows per unit length (per ft if Engish, per meter if SI)	
	Ram_Stroke	real			x	Ram stroke	ft, m
	Hammer_Energy	real			x	hammer energy	kip.ft, kN.m
	Note	str			x	note at this depth	
5	**Driving_Analysis**		**1-1**	**x**			
	SI_Unit	boolean			x	either "true" or "false"	
	Table_Number	integer			x	numbering system for statistical purpose; e.g. all piles in Table_Number="1" will be listed in Table 1	
6	**Results_From_1_Blow**		**1-∞**	**x**			
(id)	***Tip_Elev***	***real***			***x***	***tip elevation***	***ft, m***
7	Summary		1-1	x			
	Take_Average	boolean			x	either "true" or "false", if "true" then the capacities prediction of this pile will be used to calculate the average, standard deviation, etc. in its Table_Number	
	Blow_Type	str			x	EOD or BOR	
	Hammer_Type	str			x	Hammer type	
	Blow_Number	real			x	blow number	
	Data_Set	str			x	data set	
	Msd_Blow_Count	real			x	measured blow count	
	Cal_Blow_Count	real			x	calculated blow count	
	Last2_Side	real			x	sum side resistance of the last two soil segments	kip, kN
	Total_Side	real			x	total side resistance	kip, kN
	Tip	real			x	tip resistance	kip, kN
	Total	real			x	total capacity	kip, kN
	Side_Smith_Damping	real			x	Side Smith Damping	s/ft, s/m
	Tip_Smith_Damping	real			x	Tip Smith Damping	
	Side_Quake	real			x	Side Quake	in, mm

	Tip_Quake	real			x	Tip Quake	
	Gap	real			x	Gap	in, mm
	Side_Case_Damping	real			x	Side Case Damping	
	Tip_Case_Damping	real			x	Tip Case Damping	
	Pile_Damp	real			x	Pile Damp, number in % (e.g. 1%, not 0.01)	%
	Side_Quake_Ratio	real			x	Side Quake Ratio	
	Tip_Quake_Ratio	real			x	Tip Quake Ratio	
	Side_Unload_Level	real			x	Side Unload Level	
	Max_Comp_Force	real			x	Max Compression Force	kip, kN
	Max_Energy	real			x	Max Energy	kip.ft, kN.m
	MQN	real			x	MQN	
	FCLP	real			x	FCLP	
	AA12	real			x	AA12	ft/s^2, m/s^2
	ACAS	real			x	ACAS	
7	Trace		1-1	x			
8	Trace_Data		1-∞	x			
	Force	real			x	force	kip, kN
	V	real			x	velocity	ft/s, m/s
	WU	real			x	wave up	kip, kN
	D	real			x	displacement	in, mm
7	Parameters		1-1	x			
	Distance_Gage_To_Grade	real			x	distance from gage to grade (ground)	ft, m
	Test_Date	date			x	date the test is taken	
	Report_Date	date			x	date the test is reported	
	Operator	str			x	operator	
	Side_Case_Damp	real			x	Side Case Damping factor	
	Side_Unload_Quake	real			x	Side Unload Quake (% of loading quake)	
	Side_Reload_Level	real			x	Side Reloading Level (% of Ru)	
	Side_Unload_Level	real			x	Side Unloading Level (% of Ru)	
	Side_Radiation_Damp	real			x	Side Radiation Damping	
	Side_Soil_Plug	real			x	Side Soil Plug weight	kip, kN
	Tip_Case_Damp	real			x	Tip Case Damping factor	
	Tip_Unload_Quake	real			x	Tip Unload Quake (% of loading quake)	
	Tip_Reload_Level	real			x	Tip Reloading Level (% of Ru)	
	Tip_Resistance_Gap	real			x	Tip Resistance Gap (included in tip quake)	in, mm
	Tip_Radiation_Damp	real			x	Tip Radiation Damping	
	Tip_Soil_Plug	real			x	Tip Soil Plug weight	kip, kN
	Match_Quality	real			x	Match Quality MQ	
	Observed_Blow_Count	real			x	Observed Blow Count	blows/ft,
	Computed_Blow_Count	real			x	Computed Blow Count	blows/m
	Observed_Final_Set	real			x	Observed Final Set	in, mm
	Computed_Final_Set	real			x	Computed Final Set	
	RAU	real			x	RAU	kip, kN
	RA2	real			x	RA2	kip, kN

	J_Rs	real			x	J(Rs)	
	J_Rx	real			x	J(Rx)	
	VMAX	real			x	VMAX	ft/s, m/s
	VFIN	real			x	VFIN	ft/s, m/s
	VT1Z	real			x	VT_1*Z	kip, kN
	FT1	real			x	FT1	kip, kN
	FMAX	real			x	FMAX	kip, kN
	DMAX	real			x	DMAX	in, mm
	DFIN	real			x	DFIN	in, mm
	EMAX	real			x	EMAX	kip.ft, kN.m
	RLT	real			x	RLT	kip, kN
	Note	str			x	Note	
	Tip_Area	real			x	Tip Area	ft^2, m^2
	Top_Impedance	real			x	Top Impedance	kip/ft/s, kN/m/s
	Time_Inc	real			x	Time Increment	ms
	Top_Seg_Length	real			x	Top Segment Length	ft, m
	Pile_Damp	real			x	Pile Damping	%
	Wave_Speed	real			x	Wave Speed	ft/s, m/s
8	Case_RS1		1-1	x			
	At_J0	real			x	Case method: RS1 at different J=0 to J=0.9	
	.	real			x		
	.	real			x		
	At_J9	real			x		
8	Case_RMX		1-1	x		Case method: RMX at different J=0 to J=0.9	
	At_J0	real			x		
	.	real			x		
	.	real			x		
	At_J9	real			x		
8	Case_RSU		1-1	x		Case method: RSU at different J=0 to J=0.9	
	At_J0	real			x		
	.	real			x		
	.	real			x		
	At_J9	real			x		
8	Pile_Profile_Model		1-1	x			
9	Prof_Data		1-∞	x		profile data and pile model	
	Depth	real			x	depth	ft, m
	Area	real			x	cross sectional area	in^2, mm^2
	Modulus	real			x	dynamic modulus of pile material	ksi, MPa
	Sp_Weight	real			x	specific weight of pile material	pcf, kN/m^3
	Circumference	real			x	circumference	ft, m
7	CAPWAP		1-1	x			
8	CAPWAP_Data		1-∞	x			
	Gage_Distance	**str**			x	Distance below gages. Note that the variable type is string, because it also has values like "Average Side" and "Tip"	ft, m

	Side_Ru	real			x	side resistance (Ru)	kip, kN
	Pile_Tip_Force	real			x	Force in pile tip	kip, kN
	Smith_Damp	real			x	smith damping factor	s/ft, s/m
	Quake	real			x	quake	in, mm
7	Extrema_Table		1-1	x			
8	Extrema_Data		1-∞	x			
	Gage_Distance	real			x	Distance below gages	ft, m
	Max_Force	real			x	Max Force	kip, kN
	Min_Force	real			x	Min Force	kip, kN
	Max_Trf_Energy	real			x	Max Transfered Energy	kip.ft, kN.m
	Max_Vel	real			x	Max Velocity	ft/s, m/s
	Max_Disp	real			x	Max Displacement	in, mm
8	Absolute_Max		1-1	x			
	Gage_Distance	real			x	Distance below gages at which the compression stress is max	ft, m
	Cmp_Stres	real			x	maximum compression stress	ksi, MPa
	T	real			x	time occured	ms
8	Absolute_Min		1-1	x			
	Gage_Distance	real			x	Distance below gages at which the tension stress is max	ft, m
	Tens_Stress	real			x	maximum tension stress	ksi, MPa
	T	real			x	time occured	ms
6	**Driving_Table**		**1-∞**	**x**			
	Start_Time	date & time			x	Start Time	
	Wave_Speed	real			x	Wave Speed	ft/s, m/s
	Hammer_Type	str				Hammer type	
(id)	***Sub_Table_Number***	***integer***			***x***	***1 pile may have many driving table (depending on the Drv_Data "Type" is "AV/10" or similar***	
	Comment	str			x	Comment	
	JC1	real			x	JC value used for RX1	
	JC2	real			x	JC value used for RX2	
	JC3	real			x	JC value used for RX3	
7	Drv_Data		1-∞	x			
	Blow_End	real			x	Blow number	
	Depth	real			x	Depth	ft, m
	BLC	real			x	Blow count	
	Type	str			x	Type, e.g. AV/10	
	CSX	real			x	Max measured compression stress	ksi, MPa
	CSI	real			x	Max F1 or F2 compression stress	ksi, MPa
	TSX	real			x	Max tension stress	ksi, MPa
	EMX	real			x	Max transferred energy	kip.ft, kN.m
	BPM	real			x	blow per minutes	
	BTA	real			x	beta (β) integrity factor	

	RX1	real			x	RX corresponding to JC1	kip, kN
	RX2	real			x	RX corresponding to JC2	kip, kN
	RX3	real			x	RX corresponding to JC1	kip, kN
	RA2	real			x	RA2	kip, kN
4	**Shaft**		**1-∞**	x			
(id)	***Shaft_Name***	***str***			***x***		
	As_Built	boolean			x	either "true" or "false"	
	SI_Unit	boolean			x	either "true" or "false"	
	Latitude	real			x	GPS coordination of the shaft	
	Longitude	real			x		
	Station	str			x	Station and offset of the shaft.	
	Offset	real			x	Offset: + if to the right, - if to the left	
	x_Coordinate	real			x	user input coordination of the shaft, in referenced to	ft, m
	y_Coordinate	real			x	begin of the bridge	
	Comment	str				any note about the shaft	ft, m
	Pile_Type	integer			x	6 - drilled shaft; 7 - auger cast	
	T_Length	real			x	Total shaft length	ft, m
	Embed_Length	real			x	embedded shaft length	ft, m
	Dia_or_B	real			x	equivalent diameter of shaft (to interpret Davission capacity from load test)	in, mm
	GWT	real			x	ground water table	ft, m
	Ground_Elev	real			x	ground elevation	ft, m
	Scour_Elev	real			x	scour elevation	ft, m
	Rock_Socket_L	real			x	rock socket length	ft, m
	Bell_Dia	real			x	bell diameter	in, mm
	Bell_L	real			x	bell length	ft, m
	Pile_E	real			x	modulus of the material	ksi, MPa
	Concrete_fc	real			x	concrete strength f 'c	ksi, MPa
	Concrete_Slump	real			x	concrete slump	in, mm
	Material_Cross_Area	real			x	Cross area of the shaft material	in^2, mm^2
	Tip_Elev	real			x	Tip elevation	ft, m
5	Section		1-∞	x		Shaft may have multiple sections. Each section is stored in this sub-table	
	Diameter	real			x	diameter the section (if circular)	in, mm
	Size_a	real			x	size of each side if the section is rectangle	in, mm
	Size_b	real			x		in, mm
	From_Elev	real			x	the elevation of the section	ft, m
	To_Elev	real			x		ft, m
	Drilling_Method	integer			x	code: 1 – dry; 2 – slurry and/ or casing; 3 – casing only	
	Casing_L	real			x	casing length	ft, m
	Steel_Area	real			x	reinforced steel area	in, mm
	Concrete_Volume	real			x	concrete volume	ft^3, m^3
5	**Shaft_Capacity**		**1-1**	x			
	Company	str			x	Company that does pile capacity prediction & design pile	

	Boring_Name	str			x	The name (#) of the boring that is used to predict shaft capacity	
	Project_Number	str			x	the project number that this boring is taken from. This value can be blank. In that case, this boring is taken from the current project	
	V_Method	str			x	method of vetical capacity prediction (e.g. FHWA 1999)	
	Tip_Resistance	real			x	tip ultimate resistance	kip, kN
	Side_Resistance	real			x	side ultimate resistance	kip, kN
	V_Ultimate_Capacity	real			x	vetical ultimate capacity	kip, kN
	Davisson_Capacity	real			x	Davisson capacity	kip, kN
	V_Design_Capacity	real			x	vetical design capacity	kip, kN
	L_Method	str			x	method of lateral capacity prediction (e.g. FB-Pier)	
	L_Ultimate_Capacity	real			x	lateral ultimate capacity	kip, kN
	L_Design_Capacity	real			x	lateral design capacity	kip, kN
6	Soil_Data		1-∞	x			
	Depth	real			x		ft, m
	Soil_Type	integer			x	1 - Clay; 2 - Silt, or Mix; 3 - Sand; 4 - Limestone; 5 - Void	
	N	real			x	N value	
	Total_Unit_Weight	real			x	total unit weight	pcf, kN/m3
	Su	real			x	undrained shear strength (for cohesive soil)	ksf, kPa
	qu	real			x	unconfined compressive strength	ksf, kPa
	qt	real			x	tensile strength	ksf, kPa
	qb	real			x	bearing	ksf, kPa
	Em	real			x	mass modulus	ksf, kPa
	RQD_Reduction	real			x	RQD reduction modification	
	Socket_Roughness	integer			x	0 - “smooth” or 1 - “rough"	
5	**Static_LT**		**1-∞**			same as in Pile	
5	**Osterberg**		**1-∞**				
5	**Statnamic**		**1-∞**				
5	**Hole_Info**		**1-1**	**x**			
	Company	str			x	the company that collect the hole information when drilling/ taking the core from the hole	
	Core_Size	real			x	core size	in, mm
6	Data		**1-∞**	x			
	Core_Run	str			x	Core run	
	Box	str			x	box	
	Elev	real			x	elevation	ft, m
	Depth	real			x	depth	ft, m
	Length	real			x	length of core run	ft, m
	Socket_Roughness	integer			x	0 - “smooth” or 1 - “rough”, which describes the roughness of the wall surface (taken into account when predict shaft capacities)	
	Recovery	real			x	recovery in %	
	RQD	real			x	RQD in %	
	Note	str			x	any note at this depth	

Default unit:

Length (or depth):	ft, in, m, mm
(in and mm are used for small measurements, such as displacement, diameter, etc.)	
Force:	lb, kip, KN
Stress, modulus, strength:	ksf and kPa are used for soil & rock
	ksi and MPA are used for concrete, steel, cushion, etc.
	MPa is also used for q_c and u_0 in CPT test if SI unit is used
Unit weight	pcf, kN/m3

All other unit combinations will use ft, m, kip and kN (e.g. kip.ft, kN.m, m/s, ft/s. m^2/s, ft^2/s, etc.)

Exception: CPT data (qc, fs, etc.), tsf instead of default ksf (if English unit is used)

XML prototype:

XML tags represent the Table (or Sub-Table) names
XML attributes represent data.
example:

```
<GML>
    <Project
        Name="Example project"
        Project_Number="791803514"
        Financial_Number="123A45">
        <Bridge
            Bridge_Name="West bound">
            <Pier
                Pier_Name="11">
                <Pile
                    Pile_Name="1"
                    T_Length="90"
                    Tip_Elev="-84"  />
            </Pier>
        </Bridge>
    </Project>
</GML>
```

In this example, "Project" is a table, that contains "Bridge" sub-table, and then nested inside this sub-table are other sub-tables: "Pier", then "Pile", etc. All of these are represented by XML tags. Project_Number, Financial_Number, Bridge_Name, Pier_Name, Pile_Name, T_Length, Tip_Elev, etc. contains data, therefore they are represented by XML attributes