



---

Theses and Dissertations

---

2010-03-15

## An Enhanced Data Model and Tools for Analysis and Visualization of Levee Simulations

Thomas Richard Griffiths  
*Brigham Young University - Provo*

Follow this and additional works at: <https://scholarsarchive.byu.edu/etd>



Part of the [Civil and Environmental Engineering Commons](#)

---

### BYU ScholarsArchive Citation

Griffiths, Thomas Richard, "An Enhanced Data Model and Tools for Analysis and Visualization of Levee Simulations" (2010). *Theses and Dissertations*. 2061.

<https://scholarsarchive.byu.edu/etd/2061>

This Thesis is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact [scholarsarchive@byu.edu](mailto:scholarsarchive@byu.edu), [ellen\\_amatangelo@byu.edu](mailto:ellen_amatangelo@byu.edu).

An Enhanced Data Model and Tools for Analysis  
and Visualization of Levee Simulations

Thomas R. Griffiths

A thesis submitted to the faculty of  
Brigham Young University  
in partial fulfillment of the requirements for the degree of  
Master of Science

Norman L. Jones, Chair  
Alan K. Zundel  
E. James Nelson

Department of Civil and Environmental Engineering  
Brigham Young University

Thesis Completed  
April 2010

Copyright © 2010 Thomas R. Griffiths

All Rights Reserved



## ABSTRACT

### An Enhanced Data Model and Tools for Analysis and Visualization of Levee Simulations

Thomas R. Griffiths

Department of Civil and Environmental Engineering

Master of Science

The devastating levee failures associated with hurricanes Katrina and Rita, and the more recent Midwest flooding, placed a spotlight on the importance of levees and our dependence on them to protect life and property. In response to levee failures associated with the hurricanes, Congress passed the Water Resources Development Act of 2007 which established a National Committee on Levee Safety. The committee was charged with developing recommendations for a National Levee Safety Program. The Secretary of the Army was charged with the establishment and maintenance of a National Levee Database.

The National Levee Database is a critical tool in assessing and improving the safety of the nation's levees. However, the NLD data model, established in 2007, lacked a structure to store seepage and slope stability analyses – vital information for assessing the safety of a levee. In response, the Levee Analyst was developed in 2008 by Dr. Norm Jones and Jeffrey Handy. The Levee Analysis Data Model was designed to provide a central location, compatible with the National Levee Database, for storing large amounts of levee seepage and slope stability analytical data. The original Levee Analyst geoprocessing tools were created to assist users in populating, managing, and analyzing Levee Analyst geodatabase data.

In an effort to enhance the Levee Analyst and provide greater accessibility to levee data, this research expanded the Levee Analyst to include modifications to the data model and additional geoprocessing tools that archive GeoStudio SEEP/W and SLOPE/W simulations as well as export the entire Levee Analyst database to Google Earth. Case studies were performed to demonstrate the new geoprocessing tools' capabilities and the compatibility between the National Levee Database and the Levee Analyst database. A number of levee breaches were simulated to prototype the enhancement of the Levee Analyst to include additional feature classes, tables, and geoprocessing tools. This enhancement would allow Levee Analyst to manage, edit, and export two-dimensional levee breach scenarios.

Keywords: National Levee Database, Levee Analyst, GeoStudio, SEEP/W, SLOPE/W, GMS, WMS, GSSHA, levee breach, Google Earth, borehole log, seepage, slope stability





## ACKNOWLEDGMENTS

I am grateful for the research assistance I have received from many individuals. I wish to thank my advisor, Dr. Norman L. Jones, for his help and encouragement. I would also like to thank my committee members, Dr. E. James Nelson and Dr. Alan K. Zundel, for their assistance. Finally, I wish to thank my wife and parents for their continual support.



## TABLE OF CONTENTS

<b>LIST OF FIGURES .....</b>	<b>ix</b>
<b>1 Introduction.....</b>	<b>1</b>
1.1 National Levee Database .....	2
1.2 Levee Analyst .....	3
1.3 Literature Review .....	6
1.3.1 GIS-Based Levee Database Management Systems .....	6
1.3.2 Current Practice for Modeling and Assessment of Levee Breach Scenarios.....	7
1.4 Research Objectives.....	8
<b>2 Expanded Levee Analyst Tools.....</b>	<b>11</b>
2.1 Import GeoStudio Simulation.....	13
2.2 Export Raster From Attribute Field.....	17
2.3 Export Cross Section To KML .....	22
<b>3 Revised Levee Analyst Tools and Schema .....</b>	<b>25</b>
3.1 FileType Coded-Value Domain.....	25
3.2 Slope Stability Table.....	26
3.3 Simulations Query Geoprocessing Tool.....	27
3.4 Export Simulation Geoprocessing Tool.....	28
<b>4 Case Study: St. Louis National Levee Database .....</b>	<b>31</b>
4.1 Merging Levee Analyst Schema with the Existing National Levee Database .....	32
4.1.1 Adding the Seepage, Slope Stability, Simulation, and Executables Tables .....	32
4.1.2 Adding and Populating Fields in the levee_centerline and cross_section_line Feature Classes.....	32
4.2 Populating Seepage, Slope Stability, and Simulation Tables .....	35
4.2.1 Creating and Exporting SEEP2D and UTEXAS Simulations in GMS .....	35

4.2.2	Creating SEEP/W and SLOPE/W Simulations in GeoStudio 2004 and 2007.....	36
4.2.3	Archiving GeoStudio Simulations .....	40
4.3	Populating the Executables Table.....	41
4.4	Executing the Simulations Query Geoprocessing Tool.....	41
4.5	Executing the Export Simulation Geoprocessing Tool.....	42
4.6	Populating the borehole_point Feature Class .....	45
4.6.1	Adding Additional Fields to the borehole_point Feature Class.....	45
4.6.2	Populating the Additional Fields in the borehole_point Feature Class.....	46
4.7	Executing the Export Raster From Attribute Field Geoprocessing Tool.....	46
4.8	Executing the Layer To KML Geoprocessing Tool .....	49
4.9	Executing the Export Cross Section To KML Geoprocessing Tool.....	49
4.10	Exporting the Remaining Feature Classes .....	50
4.11	Viewing the KML and KMZ Files in Google Earth.....	50
<b>5</b>	<b>Case Study: Southern Texas Levee Data .....</b>	<b>57</b>
5.1	Creating the Levee Analyst Geodatabase .....	58
5.2	Digitizing the Levee Centerline and Levee Cross Sections.....	58
5.3	Archiving UTEXAS and SEEP2D Simulations .....	59
5.4	Archiving GeoStudio Simulations .....	61
5.5	Executing the Export Cross Section To KML Geoprocessing Tool.....	62
5.6	Populating and Exporting the Borehole Feature Class .....	62
5.6.1	Creating Additional Attribute Table Fields .....	62
5.6.2	Populating the Raster Field.....	63
5.6.3	Executing the Export Raster From Attribute Table Geoprocessing Tool.....	63
5.6.4	Executing the Layer To KML Geoprocessing Tool.....	63
5.7	Exporting the Remaining Feature Classes .....	64

5.8	Viewing the KMZ and KML Files in Google Earth.....	64
<b>6</b>	<b>Automated Levee Breach Simulator Prototype .....</b>	<b>67</b>
6.1	St. Louis Levee Breach Simulations Demonstration .....	68
6.2	Southern Texas Levee Breach Simulations Demonstration .....	73
6.3	Creation of GSSHA Flood Inundation Simulations .....	81
6.3.1	Flood Inundation Model Domain.....	81
6.3.2	Model Domain Discretization.....	82
6.3.3	Assigning Grid Elevation Values.....	82
6.3.4	Assigning Grid Overland Flow Roughness Values .....	82
6.3.5	Representation of Linear Features .....	83
6.4	Creation of GSSHA Flood Inundation Geoprocessing Tools.....	87
6.4.1	Creation of a Archive GSSHA Flood Inundation Simulation Geoprocessing Tool .....	88
6.4.2	Creation of an Update GSSHA Simulation Boundary Conditions Geoprocessing Tool .....	89
6.4.3	Creation of a Run GSSHA Simulation Geoprocessing Tool .....	92
6.4.4	The Creation of an Export GSSHA Simulation to KMZ Geoprocessing Tool.....	93
6.4.5	Viewing GSSHA Archived Flood Inundations in Google Earth .....	94
<b>7</b>	<b>Conclusion .....</b>	<b>97</b>
7.1	Research Accomplishments .....	97
7.2	Future Developments and Research .....	98
	<b>References.....</b>	<b>101</b>
	<b>Appendix A Updated Levee Analysis Data Model.....</b>	<b>103</b>



## LIST OF FIGURES

Figure 1-1: Original Levee Analysis Data Model.....	5
Figure 2-1: Original Levee Analyst toolbox seen within the ArcToolbox structure in ArcGIS.....	11
Figure 2-2: Updated Levee Analyst toolbox seen within the ArcToolbox structure in ArcGIS.....	13
Figure 2-3: <i>Import GeoStudio Simulation</i> geoprocessing tool.....	15
Figure 2-4: Workflow of the <i>Import GeoStudio Simulation</i> geoprocessing tool.....	16
Figure 2-5: Google Earth description balloon.....	18
Figure 2-6: <i>Export Raster From Attribute Field</i> geoprocessing tool.....	20
Figure 2-7: Workflow of the <i>Export Raster From Attribute Field</i> and <i>Layer To KML</i> geoprocessing tools.....	21
Figure 2-8: <i>Export Cross Section To KML</i> geoprocessing tool.....	23
Figure 3-1: <i>FileType</i> coded-value domain.....	26
Figure 3-2: Fields in the <i>Slope Stability</i> table.....	27
Figure 3-3: Fields in the <i>Simulations Query Output</i> table.....	28
Figure 3-4: Updated workflow of the <i>Export Simulation</i> geoprocessing tool.....	29
Figure 4-1: Adding the <i>Seepage</i> , <i>Slope Stability</i> , <i>Simulation</i> , and <i>Executables</i> tables to the NLD.....	33
Figure 4-2: Fields in the <i>LeveeCenterline</i> feature class.....	34
Figure 4-3: Fields in the <i>CrossSectionLine</i> feature class.....	34
Figure 4-4: Levee cross section created in GMS.....	36
Figure 4-5: Levee cross section SEEP/W analysis developed in GeoStudio 2004.....	37
Figure 4-6: Levee cross section SLOPE/W analysis developed in GeoStudio 2004.....	37
Figure 4-7: Levee cross section SEEP/W analysis developed in GeoStudio 2007.....	39
Figure 4-8: Levee cross section landside SLOPE/W analysis developed in GeoStudio 2007.....	39



Figure 4-9: Levee cross section riverside SLOPE/W analysis developed in GeoStudio 2007.....	40
Figure 4-10: <i>Executables</i> table with the GMS and GeoStudio program executables stored as table records.....	41
Figure 4-11: <i>Simulations Query</i> output table for a GeoStudio 2004 levee cross section. ....	42
Figure 4-12: Simulations Query output table for a GeoStudio 2007 levee cross section. ....	43
Figure 4-13: Simulation exported from ArcGIS to GeoStudio 2004 using the <i>Export Simulation</i> tool.....	44
Figure 4-14: Simulation exported from ArcGIS to GeoStudio 2007 using the <i>Export Simulation</i> tool.....	44
Figure 4-15: Loading an image in the raster field of the borehole_point feature class. ....	47
Figure 4-16: <i>Export Raster From Attribute Field</i> geoprocessing tool input. ....	48
Figure 4-17: <i>Export Raster From Attribute Field</i> results. ....	48
Figure 4-18: St. Louis NLD feature class KMZ and KML files displayed in Google Earth.....	51
Figure 4-19: Description balloon for a borehole_point KMZ feature. ....	52
Figure 4-20: Description balloon for a cross_section_line KML feature. ....	54
Figure 4-21: Levee cross section displayed in Google Earth web browser.....	55
Figure 5-1: Levee DEM in southern Texas.....	59
Figure 5-2: Southern Texas levee centerline polylines shown in ArcGIS.....	60
Figure 5-3: USACE UTEXAS simulation as seen in GMS.....	61
Figure 5-4: Texas Levee Analyst geodatabase feature class KMZ and KML files displayed in Google Earth.....	65
Figure 6-1: St. Louis NLD levee breach flood inundation simulations locations. ....	69
Figure 6-2: St. Louis Levee Breach Station: 248+13 flood inundation simulation summary. ....	70
Figure 6-3: St. Louis Levee Breach Station: 1018+79 flood inundation simulation summary. ....	70
Figure 6-4: St. Louis KMZ flood inundation simulation results for Station: 248+13 as seen in Google Earth.....	71
Figure 6-5: St. Louis KMZ flood inundation simulation results for Station: 1018+79 as seen in Google Earth.....	72

Figure 6-6: Southern Texas levee breach flood inundation simulations locations. ....	74
Figure 6-7: Southern Texas Levee Breach: 1 flood inundation simulation summary. ....	75
Figure 6-8: Southern Texas Levee Breach: 2 flood inundation simulation summary. ....	75
Figure 6-9: Southern Texas Levee Breach: 3 flood inundation simulation summary. ....	76
Figure 6-10: Southern Texas Levee Breach: 4 flood inundation simulation summary. ....	76
Figure 6-11: Texas KMZ flood inundation simulation results for Levee Breach: 1 as seen in Google Earth. ....	77
Figure 6-12: Texas KMZ flood inundation simulation results for Levee Breach: 2 as seen in Google Earth. ....	78
Figure 6-13: Texas KMZ flood inundation simulation results for Levee Breach: 3 as seen in Google Earth. ....	79
Figure 6-14: Texas KMZ flood inundation simulation results for Levee Breach: 4 as seen in Google Earth. ....	80
Figure 6-15: Model discretization of a NLD protected area polygon as seen in WMS. ....	84
Figure 6-16: Flood inundation simulation grid elevations as seen in WMS. ....	85
Figure 6-17: Flood inundation simulation grid roughness values as seen in WMS. ....	86
Figure 6-18: GSSHA overbank flow testing in WMS. ....	91
Figure 6-19: <i>Flood Inundation Simulation Summary</i> KML file as seen in Google Earth. ....	94
Figure A-1: Updated Levee Analysis Data Model summary. ....	103
Figure A-2: Updated Levee Analysis Data Model. ....	104



# 1 Introduction

Levees are earth embankments placed along rivers, canals, lakes, reservoirs, and shorelines to protect adjacent land from flooding. Floodwalls serve the same function as levees but are constructed of concrete and used in urban areas where the space required to construct levees is not available. These structures serve to reduce the risk of flooding to homes, businesses, and communities in the areas they protect. For decades these structures have prevented billions of dollars in damage to property and protected millions of people from the threat of floods (FEMA, 2006).

The devastating levee failures associated with hurricanes Katrina and Rita (2005), and the more recent Midwest flooding (2008), placed a spotlight on the importance of levees and our dependence on them to protect life and property. The Federal Emergency Management Agency (FEMA) estimates that approximately 22% of all U.S. counties contain levees. Further, FEMA estimates that 43% of the U.S. population lives in counties with levees (FEMA, 2006). More than 85% of the nation's estimated 100,000 miles of levees are locally owned and maintained. Many of these levees are more than 50 years old and were originally constructed to protect crops from flooding. As the levees have aged, and many of the agricultural areas they protect have been converted to homes and businesses, the risk associated with levee failures has increased (ASCE, 2009).

## 1.1 National Levee Database

In response to levee failures associated with hurricanes Katrina and Rita, Congress passed the Water Resources Development Act (WRDA) of 2007. The WRDA, also known as the National Levee Safety Act, established a *National Committee on Levee Safety* (NCLS). The NCLS was charged with developing recommendations for a National Levee Safety Program (NLSP). Congress also charged the Secretary of the U.S. Army, acting through the Chief of Engineers, with establishment and maintenance of the National Levee Database (NLD) (WRDA, 2007). The NLD is “an inventory of all federal levees, as well as those non-federal levees for which information is voluntarily provided by state and local government agencies. The inventory is intended to be a comprehensive, geospatial database that is shared between the U.S. Army Corps of Engineers, FEMA, the Department of Homeland Security (DHS), and the states” (ASCE, 2009).

The USACE performed an initial survey of all federal levees in 2006. In 2007 the USACE, FEMA, and others developed the National Levee Database model. That same year, in five pilot districts (3256 miles of levees), the USACE began to inspect and inventory all federally-owned and maintained levees, federally-built and locally-maintained levees, and levees constructed and maintained by local entities that requested the USACE to include their structures in its Rehabilitation and Inspection Program. By the end of 2008, the levees of the five pilot districts, along with 10 additional districts, were inventoried (9800 miles of levees). By the end of 2009, inventorying in all of the remaining districts was underway. In this initial inventory approximately 2,000 levees encompassing over 14,000 miles worth of levees will be included in the NLD (Pangburn, 2009).

In January of 2009 the NCLS submitted a report to Congress. Their report, entitled “Recommendations for a National Levee Safety Program,” contained 20 recommendations and a strategic plan for implementation of a National Levee Safety Program. Their second recommendation was that the NLD be expanded to include a one-time USACE inventory of all non-federal levees. If Congress approves this recommendation the USACE may add over 85,000 additional miles of levees to the NLD (NCLS, 2009).

## **1.2 Levee Analyst**

Currently the NLD represents a large investment of time and money. This investment will grow significantly if all non-federal levees are inventoried. The NLD will serve as an invaluable tool in assessing critical safety issues, evaluating the true costs of good stewardship, and assessing the state of individual levees. All this information can help establish priorities and supply data for risk-informed assessments and decision-making (NCLS, 2009).

In 2005, the USACE and FEMA separately began developing geospatially-supported levee database frameworks. FEMA became aware of the USACE database effort and joined with the USACE in defining the structure of the database. “Each of the agencies determined that agency-specific data would be developed, but the amount of common data merited a cooperative effort” (FEMA, 2006). As a result the NLD data structure was designed to meet the common needs of both agencies. The resulting design included a geodatabase with tables and feature classes to store most of the information relating to levees. However, the NLD does not include a structure for storing seepage and slope stability analyses associated with the cross sections of a levee. This data is critical in analyzing levee conditions, predicting future failure scenarios, and

improving the safety of levees. In response Dr. Norm Jones and Jeffrey Handy developed the Levee Analyst (Jones et al., 2008).

The Levee Analyst is a Geographic Information System (GIS) data model and set of geoprocessing tools. The Levee Analysis Data Model (LADM) provides a structure for storing levee centerlines, levee cross section lines, and model data associated with seepage and slope stability analyses. The data model schema includes feature classes, relationship classes, coded-value domains, and tables. This schema established a standard methodology for storing large amounts of levee-related information and created a standard set of field names necessary for using the geoprocessing tools in the Levee Analyst. The data model was created in a manner to allow future expansion and to be compatible with the NLD (Jones et al., 2008).

Figure 1-1 displays the original LADM, including all of the feature classes, tables, coded-value domains, and relationships. The feature classes included in the Levee Analysis data model include the *LeveeCenterline* and *CrossSectionLine* features classes, and the tables included in the data model are the *Executables*, *Seepage*, *Simulation*, and *Slope Stability* tables (Handy, 2008).

The *LeveeCenterline* feature class stores polylines representing the centerline of levees to be stored in the database. The *CrossSectionLine* feature class stores polylines, perpendicular to the levee centerline, that represent locations where seepage and/or slope stability analyses have been performed.

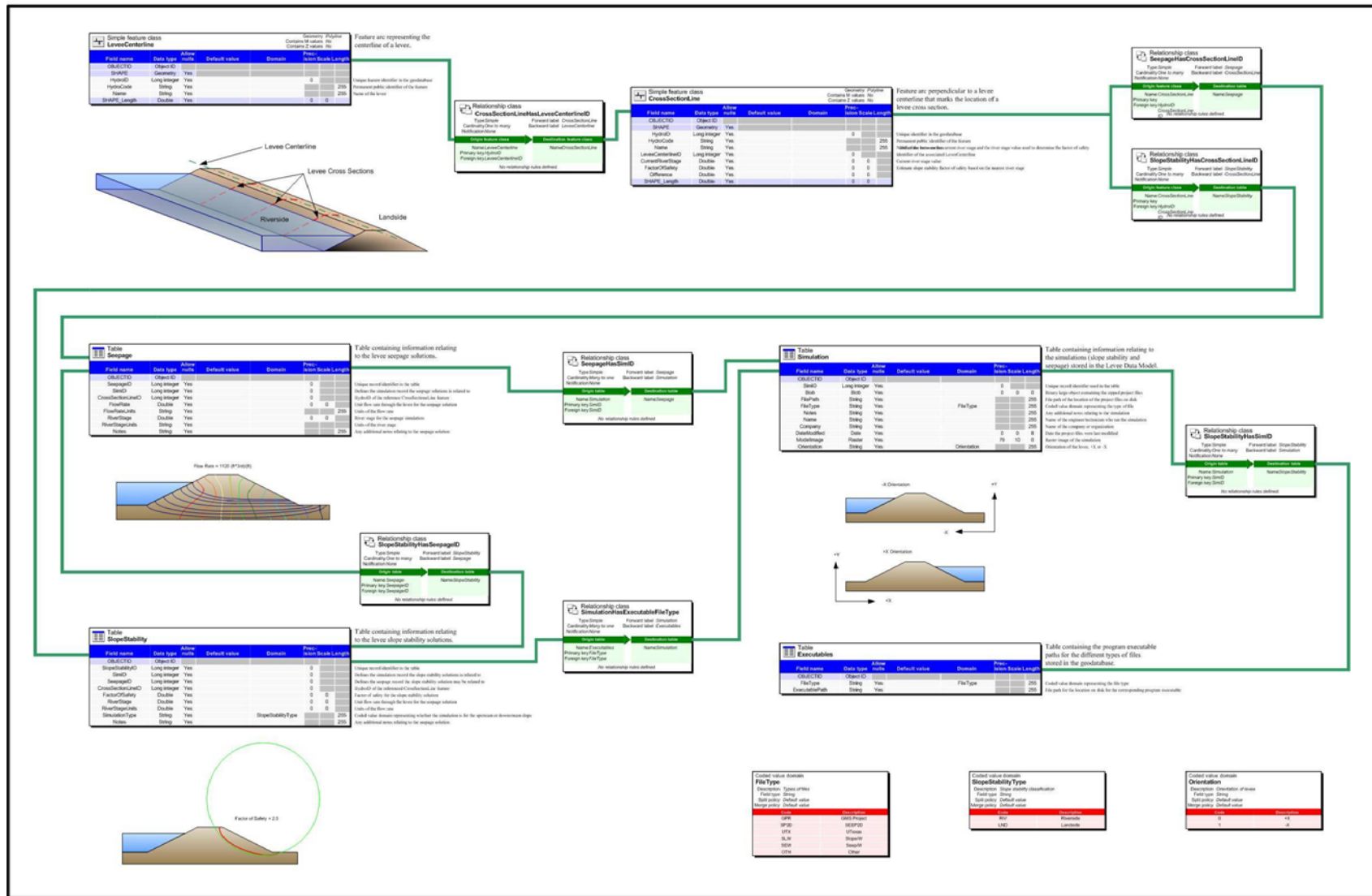


Figure 1-1: Original Levee Analysis Data Model.



The *Seepage* and *Slope Stability* tables store seepage and slope stability analysis information, respectively, for all the levee cross sections. These tables store scenario information and analytical results such as river stage, flow rate, factor of safety, and stability analysis procedure. The *Simulation* table stores summary metadata and the location of the seepage and slope stability simulation files. Lastly, the *Executables* table stores the location of the software program executable files used to perform the slope stability and/or seepage analysis.

### **1.3 Literature Review**

A literature review was performed to investigate the existence of other GIS-based levee database management systems and the current industry practice for modeling and assessment of levee breach scenarios.

#### **1.3.1 GIS-Based Levee Database Management Systems**

Very few organizations have published information about GIS-based levee database management systems. A recent survey by the Association of State Dam Safety Officials and the Association of State Floodplain Managers found that of the 23 states that have an agency with some responsibility for levee safety only 10 keep a listing of levees within their state (ASCE, 2009). As of 2006, only Wisconsin and North Dakota had geospatial inventories; while, Illinois, Mississippi, North Dakota, Ohio, and California had levee database efforts underway (FEMA, 2006).

To date, California's Department of Water Resources (DWR) has headed a massive effort to create and populate a GIS-based levee management system for the state's 13,000 miles of levees. The database is an Oracle/SDE geodatabase and it was designed to incorporate

elements from the USACE's Special Data for Facilities, Infrastructure, and Environment (SDSFIE) data standard; DWR's levee information system; and FEMA's Levee Inventory System (FLIS) (FEMA, 2006). No database structure has been published for public access.

On a federal level, aside from the NLD, FEMA appears to be the only organization that stores a substantial amount of levee GIS data. FEMA stores levee data in two databases: the FLIS database and the Digital Flood Insurance Rate Maps (DFIRM) database. The FLIS stores textual descriptions of the location of many levees throughout the nation. However, the FLIS has no GIS component. The DFIRM database has a GIS component; but, the database has no specific levee theme or layer. Levees are stored in the "General Structures" GIS theme. As a result, few levee attributes have been collected that would help evaluate levee failures or operate a levee maintenance program (FEMA, 2006).

Aside from the NLD and the Levee Analysis Data Model, no published GIS-based levee database schema could be located.

### **1.3.2 Current Practice for Modeling and Assessment of Levee Breach Scenarios**

A large number of numerical models exist that are used for modeling dam and levee breach scenarios. Altinakar et al. divided the existing models into the following general categories:

1. Simplified numerical models used to establish the envelope maximum water depths.
2. One-dimensional (1D) models that solve either fully-dynamic or simplified forms of the Saint-Venant equations.
3. Two-dimensional (2D) models that solve either fully-dynamic or simplified forms of 2D shallow water equations.

4. Coupled 1D-2D models that solve the 1D channel flow and 2D overland flow shallow water equations (2009).

In current engineering practice, the most popular modeling approach is the use of 1D models. Of the 1D models in use, the Hydrologic Engineering Center's River Analysis System (HEC-RAS) is the most common (Altinakar et al., 2009). However, the use of 1D models to simulate dam failures or levee breaches has a number of drawbacks, such as: interpolation of results from 1D to 2D is time consuming, involved, and requires engineering judgment calls; the resulting inaccuracies may be significant; and mass conservation is violated (Altinakar et al., 2008).

#### **1.4 Research Objectives**

The original Levee Analysis Data Model was designed to provide a central location, compatible with the NLD, for storing large amounts of levee seepage and slope stability analytical data. The original Levee Analyst geoprocessing tools were created to assist users in populating, managing, and analyzing Levee Analyst geodatabase data.

An objective of this research was to expand the universality of the LADM and geoprocessing tools by creating the capability of importing and exporting popular GeoStudio SEEP/W (Seepage, 2008) seepage and SLOPE/W (Stability, 2008) slope stability simulations.

Another objective was to make the data stored in a Levee Analyst database, or a NLD, more navigable to individuals with little or no ArcGIS experience by developing geoprocessing tools that would export the database to Google Earth.

The final objective of this research was to investigate the feasibility of developing ArcGIS geoprocessing tools to create, archive, edit, execute, and export flood inundation models

that simulate the breaching of levees due to river flooding, hurricane storm surges, or similar catastrophic events.



## 2 Expanded Levee Analyst Tools

Geoprocessing tools perform operations to manipulate GIS data. The Environmental Systems Research Institute's (ESRI) ArcGIS software includes hundreds of geoprocessing tools that are used to automate GIS tasks and to perform spatial analysis and modeling. The input to a geoprocessing tool consists of GIS data and parameters for manipulating that data. The geoprocessing tool performs its given operation using the supplied input - creating output GIS data that has been manipulated for a desired purpose.

The Levee Analyst includes a toolbox comprised of a set of ArcGIS geoprocessing tools. The Levee Analyst geoprocessing tools assist in importing, managing, analyzing, and exporting analytical data in a Levee Analyst database. The preliminary Levee Analyst toolbox, seen in Figure 2-1, contained five tools, including: *Add Blob to Table*, *Add Current River Stage*, *Export Simulation*, *Query Factor of Safety*, and *Simulations Query*. A brief summary of each original Levee Analyst geoprocessing tool is included to describe the functionality and uses of the tools.



Figure 2-1: Original Levee Analyst toolbox seen within the ArcToolbox structure in ArcGIS.

The *Add Blob to Table* geoprocessing tool is a general tool, created to assist users in storing any digital file in the *Blob* field of a table or feature class attribute table, as a BLOB (binary large object). Specifically, the tool was created to assist Levee Analyst users in storing zipped seepage and/or slope stability simulation files in the *Simulation* table as a BLOB. Additionally, the tool stores the path to the original zipped file.

The *Add Current River Stage* geoprocessing tool assists the user in specifying the river stage in the cross section feature class for a specified cross section.

The *Query Factor of Safety* geoprocessing tool presumes that for a levee cross section a series of stability analyses have been performed for small increments of river stage and archived in the *Slope Stability* table of the database. For a user-specified levee cross section, the tool uses the current river stage value stored in the *CrossSectionLine* feature class to query the *Slope Stability* table. The tool places the factor of safety in the *CrossSectionLine* feature class from the stability analysis with the river stage that is closest to the current river stage. Lastly, the tool takes the difference between the stability analysis river stage and the current river stage and places this value in the *Difference* field of the *CrossSectionLine* feature class.

The *Simulations Query* geoprocessing tool queries the *Simulation*, *Slope Stability*, and *Seepage* tables to find all of the simulation data associated with a user specified levee cross section. This data is copied to a new, user-named table.

The final original Levee Analyst geoprocessing tool, the *Export Simulation* geoprocessing tool, allows the user to specify a simulation from the output table produced by the *Simulations Query* geoprocessing tool. The *Export Simulation* geoprocessing tool then launches the external program executable associated with the simulation and opens the simulation in the external program.

The Levee Analyst data model and associated geoprocessing tools were designed in a manner to allow future expansion. An objective of this research was to make the data stored in a Levee Analyst database, or a NLD, more navigable to individuals with little or no ArcGIS experience by developing geoprocessing tools that would export the database to Google Earth. This was accomplished by developing the *Export Cross Section to KML* and *Export Raster From Attribute Field* geoprocessing tools. Another objective of this research was to expand the universality of the LADM and geoprocessing tools by including the capability of importing GeoStudio simulations. This was accomplished by developing the *Import GeoStudio Simulation* geoprocessing tool. Figure 2-2 shows the original Levee Analyst geoprocessing tools along with the tools added by this research (outlined in blue). A brief discussion is included to further describe the purpose and functionality of each new tool.

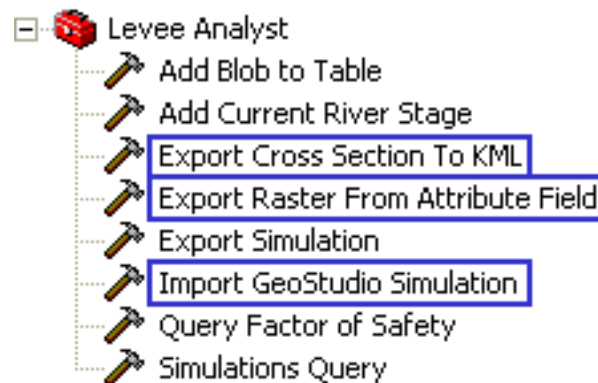


Figure 2-2: Updated Levee Analyst toolbox seen within the ArcToolbox structure in ArcGIS.

## 2.1 Import GeoStudio Simulation

Levee Analyst was designed to support different types of seepage and slope stability simulations. The preliminary tools were developed based on the SEEP2D and UTEXAS models



supported in the Groundwater Modeling System (GMS) (GMS, 2009). SEEP/W and SLOPE/W are additional, popular software programs designed to analyze groundwater seepage and slope stability, respectively. In an effort to expand the usefulness and benefit of the Levee Analysis Data Model and Levee Analyst geoprocessing tools the *Import GeoStudio Simulation* tool was developed. GeoStudio is a commercial suite of software applications for geotechnical and geo-environmental modeling. GeoStudio includes the software products SLOPE/W, SEEP/W, and a number of other analytical products. GeoStudio is the most widely used slope stability and seepage modeling software. The *Import GeoStudio Simulation* geoprocessing tool supports both the 2004 and 2007 versions of GeoStudio simulations.

Any number and combination of SLOPE/W and SEEP/W simulations can be stored in a single GeoStudio project file. The file extension for a GeoStudio project file is “.gsz.” Typically the user will use SEEP/W to determine the pore water pressure in and seepage flowrate through a levee at a particular cross section. The user then uses the calculated pore water pressures as input to a SLOPE/W analysis to determine the critical slip surface and corresponding minimum factor of safety. A completed seepage and/or slope stability analysis must then be saved to the user’s hard disk.

The *Import GeoStudio Simulation* geoprocessing tool, seen in Figure 2-3, requires the user to specify the location on disk of the saved GeoStudio analysis. The geoprocessing tool also requires the user to specify the output Levee Analyst database, the cross section feature class within the Levee Analyst database, the *CrossSectionLineID* of the levee cross section that was analyzed, the river stage used for the analysis, the engineer or technician’s name, the company name, the date the simulation was modified, any notes the user desires to enter, the orientation of

the levee cross section, and the name and location of a bitmap image of the levee cross section exported from within GeoStudio.

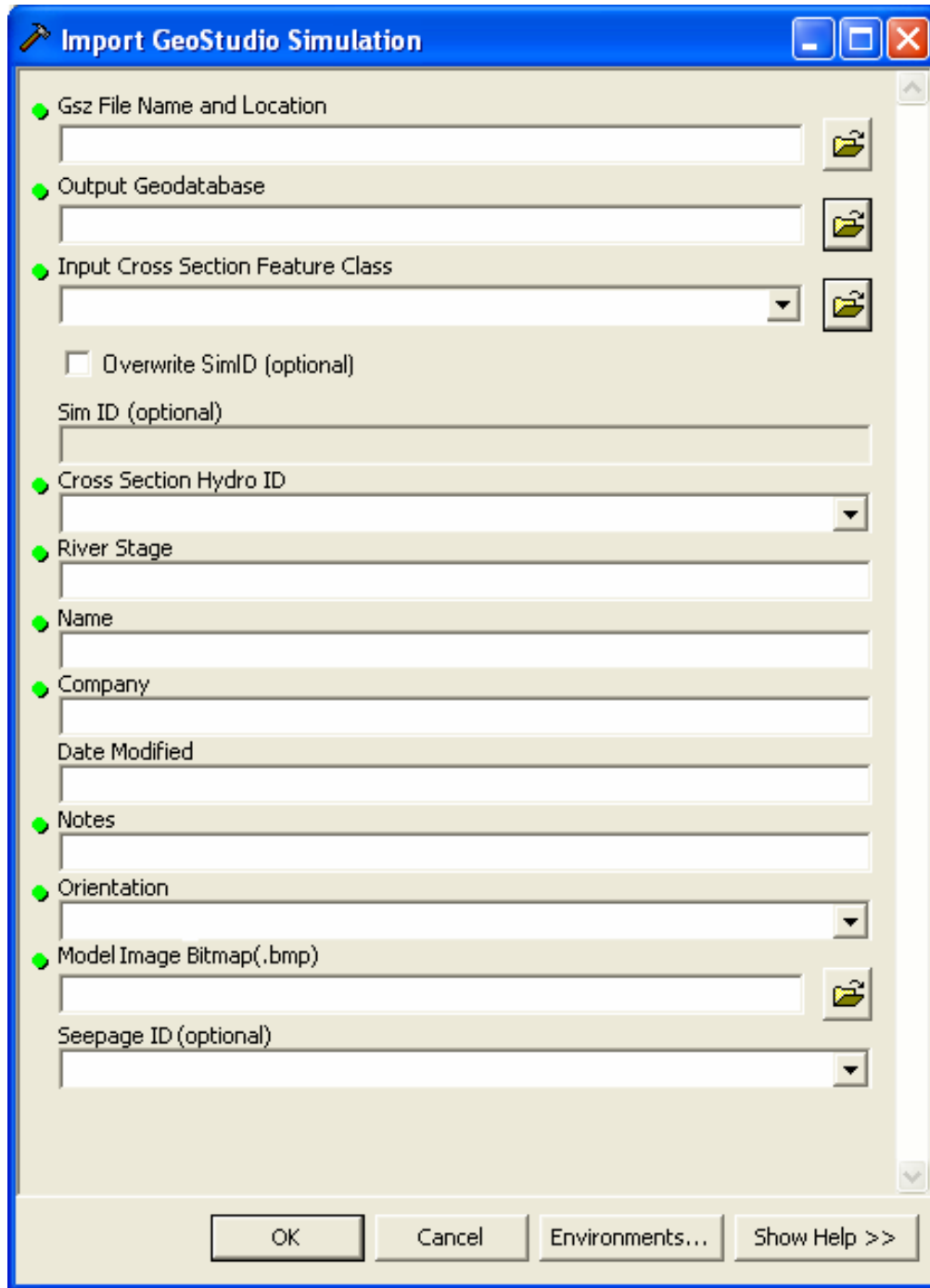


Figure 2-3: *Import GeoStudio Simulation* geoprocessing tool.

If the user has previously archived a simulation to a Levee Analyst database, and wishes to modify and re-archive the simulation, the *Overwrite SimID* box must be checked and the *SimID*, corresponding to the previously archived simulation, must be supplied.

In some situations the user may have previously archived the seepage analysis of a particular levee cross section and later the user may want to archive a slope stability analysis performed using the pore water pressures obtained in the previously archived seepage analysis. In this situation the user specifies the *SeepageID*, found in the *Seepage* table, of the previously archived seepage analysis.

Upon execution, the geoprocessing tool extracts the flowrate and flowrate units from each SEEP/W analysis and the factor of safety, analysis procedure, and simulation type from each SLOPE/W analysis present in the GeoStudio project. The GeoStudio project file is stored as a BLOB in the *Simulation* table and records in the *Simulation*, *Slope Stability*, and *Seepage* tables are created or updated (Figure 2-4).

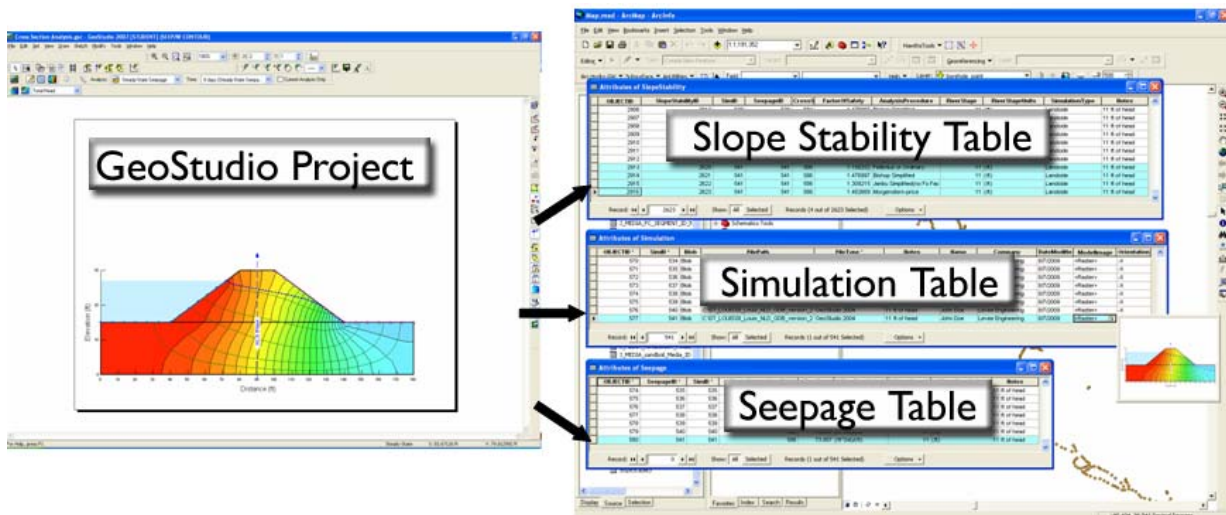


Figure 2-4: Workflow of the *Import GeoStudio Simulation* geoprocessing tool.

## 2.2 Export Raster From Attribute Field

A major objective of this research was to make the data stored in a Levee Analyst database more accessible and interactive to those unfamiliar with the ArcGIS software. Exporting data contained in a Levee Analyst database to a format that is easy to use, share, and navigate was an important step in obtaining that objective.

Keyhole Markup Language (KML) is a file format used to display geographic data in an Earth browser such as Google Earth, Google Maps, ArcGIS Explorer, and ArcGlobe. When viewed in Google Earth, KML files are easy to navigate and the symbologies of the points, lines, or polygons allow the viewer to quickly assess the condition and location of the feature being represented (i.e. borehole, cross section, levee centerline, etc.). Additional data pertaining to a point, line, or polygon displayed in Google Earth can be accessed by clicking on the feature. As a result of clicking on a feature a “description balloon” will be displayed, see Figure 2-5. The description balloon is a powerful feature of KML files. Hyper Text Markup Language (HTML) code may be placed in the description balloon; consequently, a vast amount of information can be displayed in the description balloons. Such information may include text, images, videos, hyperlinks, etc. KML files may be compressed using zip compression and given a “.kmz” extension. KML or KMZ files can be read by any Earth browser. KMZ files have small files sizes and are easy to share with interested parties (Google, 2010).

ArcGIS 9.2 and 9.3 contain the *Layer To KML* geoprocessing tool that converts an in-memory or file-based shapefile or feature class into a KML file containing a translation of ArcGIS geometries and symbology (ESRI, 2007). However, if a feature class contains an attribute field of type *raster* the *Layer To KML* geoprocessing tool ignores the field when creating the KML file. When the file is opened in Google Earth, and a particular point or arc is

selected, the description balloon window will display all of the attribute table information pertaining to that point or arc except the raster that was stored in the attribute table.

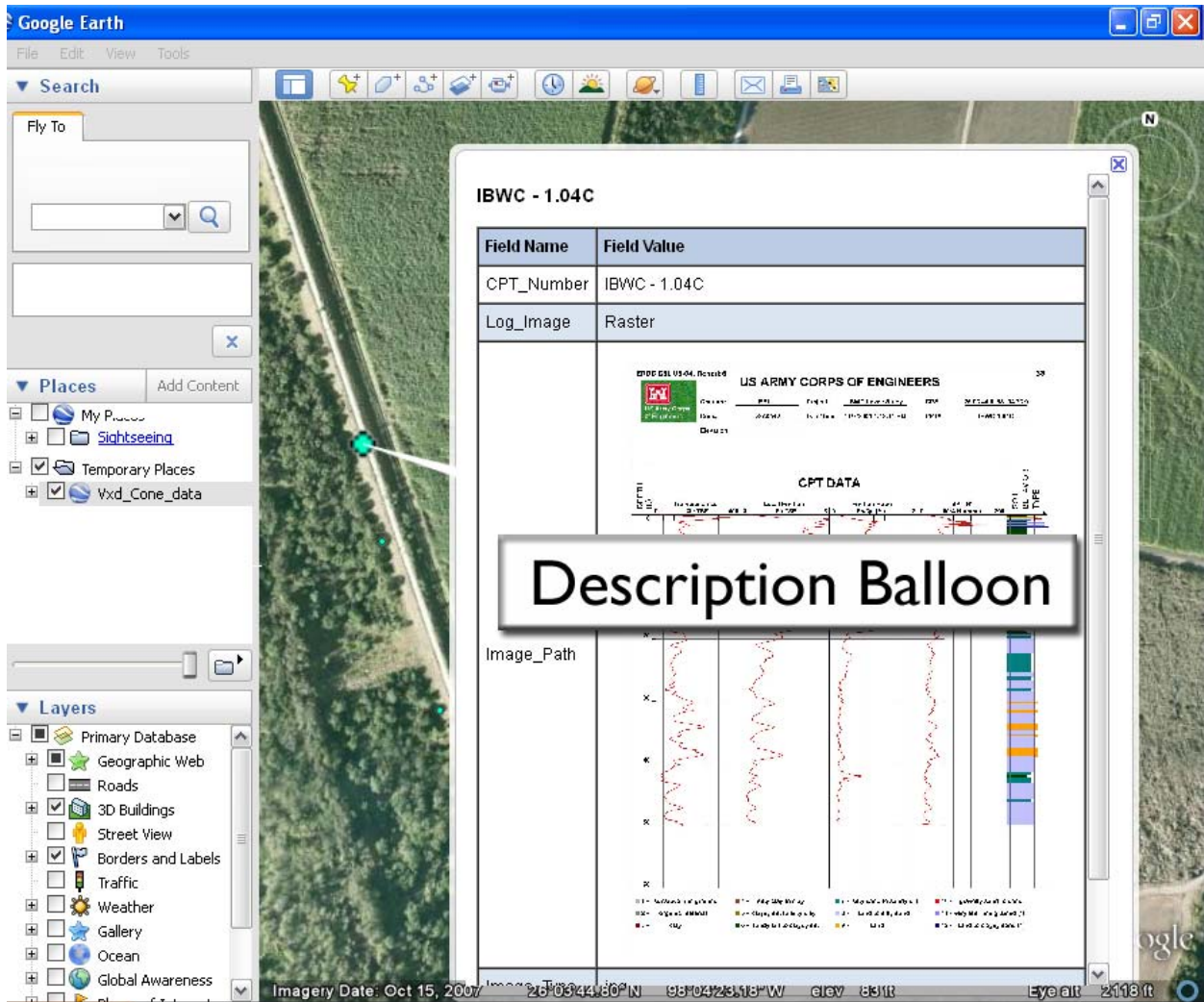


Figure 2-5: Google Earth description balloon.

For example, a user may store borehole log plot images in the attribute table of a borehole point feature class. The user would likely also store information in the attribute table such as drilling date, levee station, ground surface elevation, etc. Then, when the user exported the data to a KML file, using the *Layer To KML* geoprocessing tool, the description balloons for

the borehole points would include all of the information in the attribute table except the borehole log plot images.

The *Export Raster From Attribute Field* geoprocessing tool was designed and developed to overcome this shortcoming. The geoprocessing tool will work for any feature class that contains an attribute field of type *raster*, i.e. the geoprocessing tool is functional outside of the LADM.

The *Export Raster From Attribute Field* geoprocessing tool is shown in Figure 2-6. The first input requires the user to specify the feature class containing a raster attribute field. The second input requires the user to specify the raster attribute field within the attribute table of the feature class specified in the first input. KML files are not databases; consequently, the images stored in a Levee Analyst database must be exported to a local drive or server and a KML file created to reference these images. The third input for the geoprocessing tool is the attribute field that the tool will populate with a relative reference to the exported images. The fourth input is the attribute field the geoprocessing tool will use to name the images as they are exported. The fifth input is the windows directory where the user wants the exported images placed. The final input is the file extension of the original images that were imported into the feature class.

After the user executes the *Export Raster From Attribute Field* geoprocessing tool the user runs the ArcGIS *Layer To KML* geoprocessing tool to create a KMZ file. The resulting KMZ file will contain relative references to the images exported using the *Export Raster From Attribute Field* geoprocessing tool. Consequently, the user must place, or create, the KMZ file in the same directory as the exported images. The KMZ file can then be opened in Google Earth and the description balloon window associated with each point or arc will display the raster image exported from the feature class attribute table.

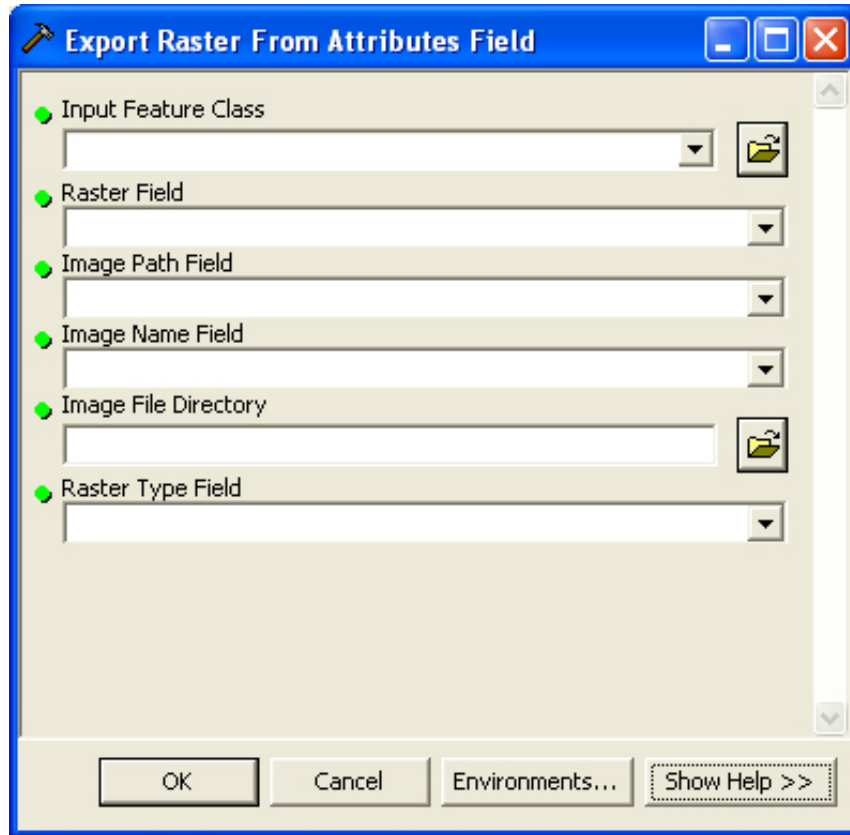


Figure 2-6: *Export Raster From Attribute Field* geoprocessing tool.

Figure 2-7 illustrates the results obtained from running the *Export Raster From Attribute Table* and *Layer To KML* geoprocessing tools in conjunction. The Figure illustrates the results for a single record and associated point in the *borehole\_point* feature class; the tools, however, export the raster and attribute table information for each feature in the input feature class.



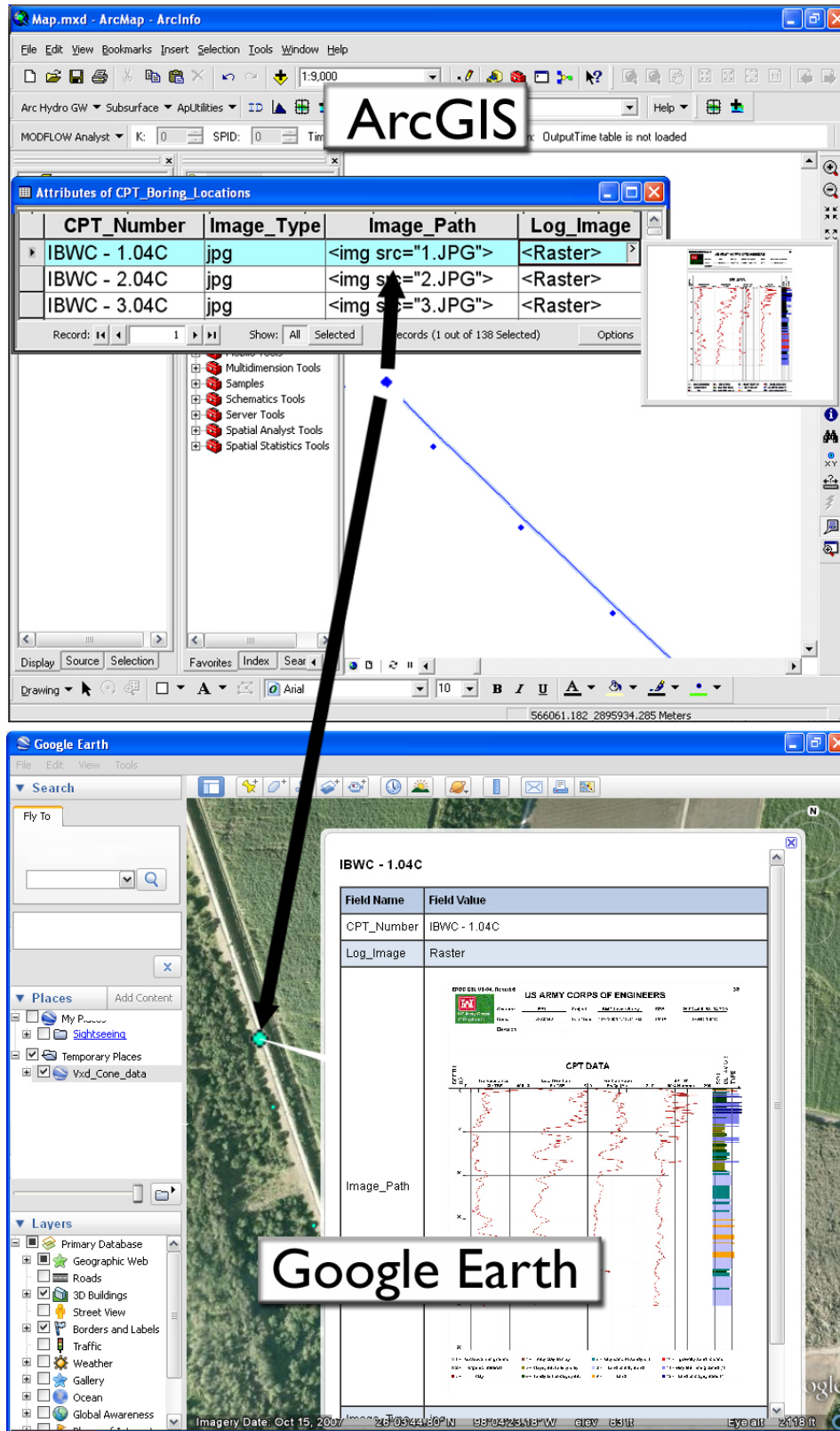


Figure 2-7: Workflow of the *Export Raster From Attribute Field and Layer To KML* geoprocessing tools.



### 2.3 Export Cross Section To KML

The final tool added to the Levee Analyst toolbox was the *Export Cross Section To KML* geoprocessing tool. This tool allows the user to export the cross section feature class, along with all of the associated seepage and slope stability simulation data stored in the *Seepage*, *Slope Stability*, and *Simulation* tables, to a KML file.

The *Layer To KML* geoprocessing tool, supplied with an ArcGIS license, is capable of exporting the cross section feature class geometries and attribute table to a KML file. The data fields in the attribute table associated with each feature exported from a feature class is converted to a KML description tag. The description tag is displayed as a description balloon when the feature is clicked in Google Earth. The *Layer To KML* geoprocessing tool does not, however, export the seepage and slope stability simulation information stored in the *Seepage*, *Slope Stability*, and *Simulation* tables to the description tag.

To append the seepage and slope stability simulation data to the cross section line feature description tags the *Export Cross Section To KML* geoprocessing tool uses the code found in the Levee Analyst *Simulations Query* geoprocessing tool. The *Export Cross Section To KML* geoprocessing tool first internally executes the *Layer To KML* geoprocessing tool to create a KML file containing the feature geometry and description tags. The description tags initially contain only the information from the cross section line attribute table. The *Export Cross Section To KML* geoprocessing tool then internally runs the *Simulations Query* geoprocessing tool code. This code sequentially creates and deletes a temporary table for each cross section line. The temporary table contains all of the information pertaining to the associated cross section found in the *Seepage*, *Slope Stability*, and *Simulation* tables. The *Export Cross Section To KML* geoprocessing tool takes the data from each temporary table and places it in the

corresponding description tag of the KML file. Finally, before deleting the temporary table, the tool exports the model image(s), found in the temporary table and obtained from the *Simulation* table, to the directory specified by the user.

The *Export Cross Section To KML* geoprocessing tool is shown in Figure 2-8. The user must specify the cross section feature class to be exported to KML. The location of the *Seepage*, *Slope Stability*, and *Simulation* tables, together with the geodatabase in which they are located, must also be identified. The user then must specify the location of the file directory to which the rasters and cross section KML file are to be exported. The final input is the name of the exported cross section KML file.

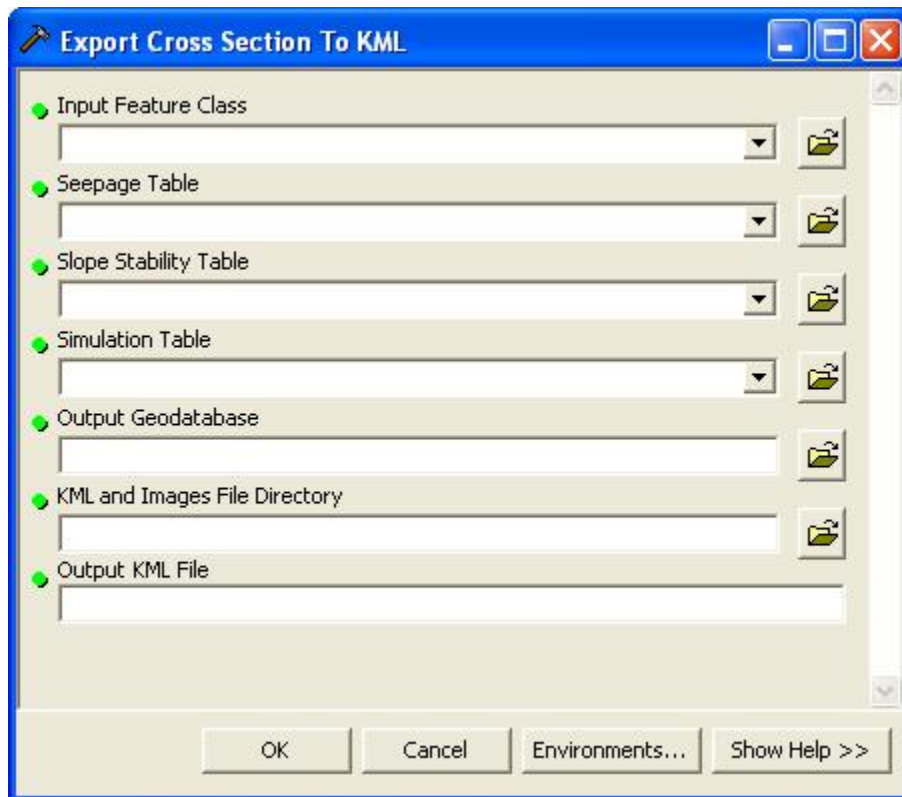


Figure 2-8: *Export Cross Section To KML* geoprocessing tool.

These additional Levee Analyst geoprocessing tools enable users to import GeoStudio 2004 and 2007 simulations, export a feature class and the images stored in its attribute table to Google Earth, and export a cross section line feature class and all the associated seepage and slope stability information stored in a Levee Analyst database to Google Earth. The toolbox could be further expanded to include tools that import additional analytical models, an *Import Levee Analysis Data Model* geoprocessing tool, a *Sync Levee Analyst and National Levee Databases* geoprocessing tool, and an *Export Levee Analyst Database to KML* geoprocessing tool.

### 3 Revised Levee Analyst Tools and Schema

As the additional geoprocessing tools were added to the Levee Analyst toolkit it became necessary to revise and update the Levee Analysis Data Model schema and certain existing Levee Analyst geoprocessing tools. A brief discussion is included to describe the purpose of each revision or update.

#### 3.1 FileType Coded-Value Domain

The *FileType* field of both the *Simulation* and *Executables* tables holds a coded-value representing the type of file stored. GeoStudio 2004 and 2007 project files have a “.gsz” file extension. To accommodate the *Import GeoStudio Simulation* geoprocessing tool the “.gsz” file extension was added to the *FileType* coded-value domain. Figure 3-1 shows the updated *FileType* coded-value domain. The list of file extensions in the coded-value domain is not exhaustive. Different seepage and slope stability simulation file extensions can be added to the coded-value domain as the user desires.

Coded value domain	
<b>FileType</b>	
Description	<i>Types of files</i>
Field type	<i>String</i>
Split policy	<i>Default value</i>
Merge policy	<i>Default value</i>
Code	Description
GPR	GMS Project
SP2D	SEEP2D
UTX	UTexas
GSZ7	GeoStudio 2007
GSZ6	GeoStudio 2004
SLW	Slope/W
SEW	Seep/W
OTH	Other

Figure 3-1: *FileType* coded-value domain.

### 3.2 Slope Stability Table

When performing a limit equilibrium method of slope stability analysis in GeoStudio, the Ordinary, Bishop, and Janbu methods are always used. The user can specify an additional method. The results for each method are stored in the project file. Typically the three or four different methods will produce different minimum values for the factor of safety. The *Import GeoStudio Simulation* geoprocessing tool was designed to import each of the methods used in the limit equilibrium slope stability analysis. The *AnalysisProcedure* field was added to the *Slope Stability* table to assist the user in identifying the method corresponding to a particular factor of safety. The updated *Slope Stability* table structure is shown in Figure 3-2, with the additional field highlighted in red.

Table SlopeStability							
Field name	Data type	Allow nulls	Default value	Domain	Prec- ision	Scale	Length
OBJECTID	Object ID						
SlopeStabilityID	Long integer	Yes			0		
SimID	Long integer	Yes			0		
SeepageID	Long integer	Yes			0		
CrossSectionLineID	Long integer	Yes			0		
FactorOfSafety	Double	Yes			0	0	
AnalysisProcedure	String	Yes					255
RiverStage	Double	Yes			0	0	
RiverStageUnits	Double	Yes			0	0	
SimulationType	String	Yes		SlopeStabilityType			255
Notes	String	Yes					255

Figure 3-2: Fields in the *Slope Stability* table.

### 3.3 Simulations Query Geoprocessing Tool

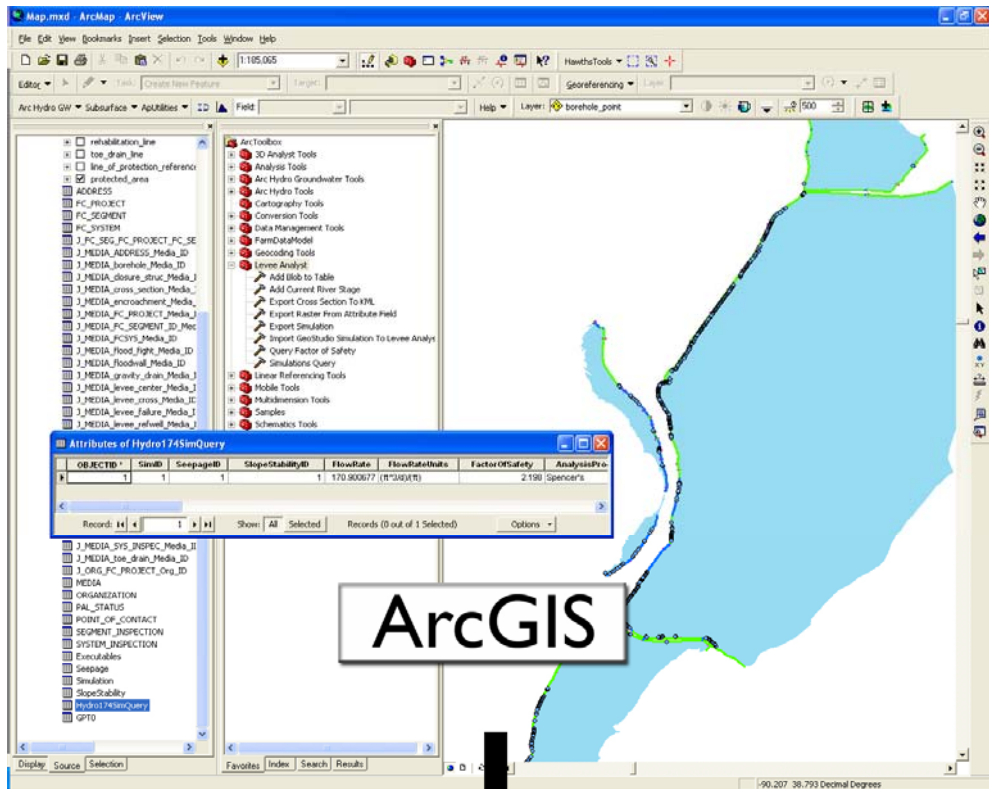
The *Simulations Query* geoprocessing tool is used to query the *Seepage*, *Slope Stability*, and *Simulation* tables. The tool creates an output table containing all of the records from these tables corresponding to a user-specified cross section line. The addition of the *AnalysisProcedure* field to the *Slope Stability* table created the need to update the *Simulations Query* geoprocessing tool to include the *AnalysisProcedure* field and value(s) in the resulting table. The updated table structure for the output table created using the *Simulations Query* geoprocessing tool is shown in Figure 3-3, with the additional field highlighted in red.

Table Simulations Query Output							
Field name	Data type	Allow nulls	Default value	Domain	Prec- ision	Scale	Length
OBJECTID	Object ID						
SimID	Long integer	Yes			0		
SeepageID	Long integer	Yes			0		
SlopeStabilityID	Long integer	Yes			0		
FlowRate	Double	Yes			0	0	
FlowRateUnits	String	Yes					255
FactorOfSafety	Double	Yes			0	0	
AnalysisProcedure	String	Yes					255
RiverStage	Double	Yes			0	0	
RiverStageUnits	String	Yes					255
FilePath	String	Yes					255
Blob	Blob	Yes			0	0	0
FileType	String	Yes		FileType			255
ModellImage	Raster	Yes			79	10	255

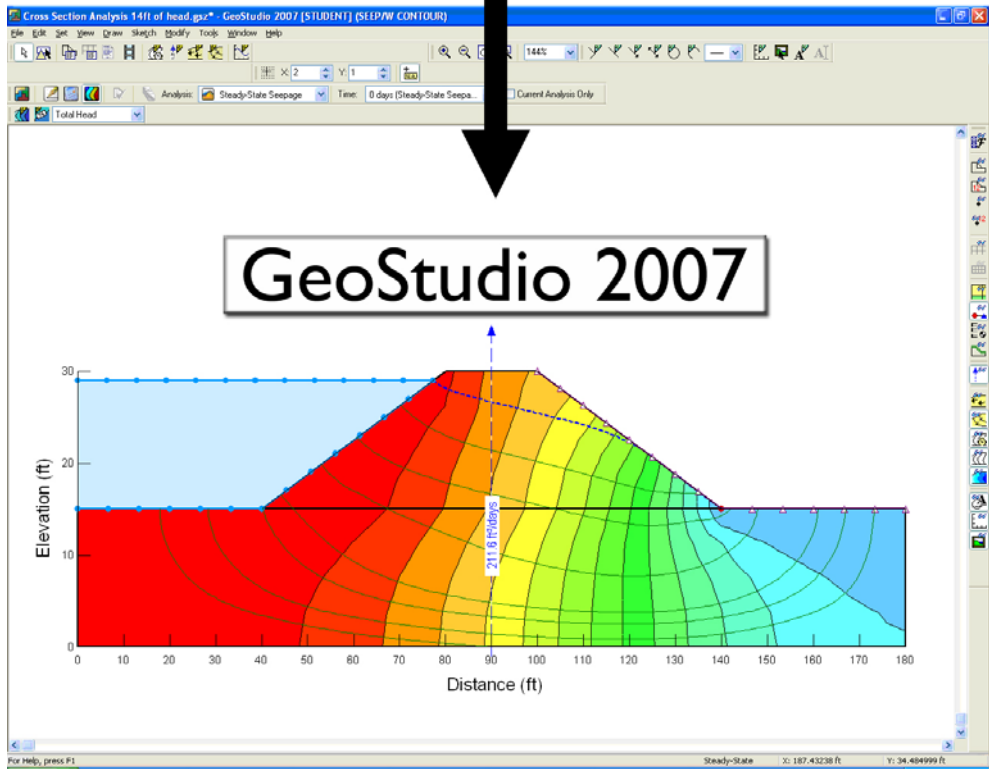
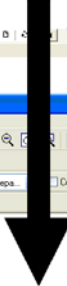
Figure 3-3: Fields in the *Simulations Query Output* table.

### 3.4 Export Simulation Geoprocessing Tool

The *Export Simulation* geoprocessing tool allows the user to specify a simulation from an output table produced by the *Simulations Query* geoprocessing tool. For the specified simulation, the tool uses the simulation file type, stored in the output table, to query the *Executables* table to find the file path of the external program executable. Then the geoprocessing tool locates the simulation file on the user's hard disk using the *FilePath* field of the output table. The *Export Simulation* geoprocessing tool then executes the associated program executable and opens the simulation file. The original version of the geoprocessing tool was incapable of launching the GeoStudio 2004 or 2007 executable and opening an associated project file. The tool was updated to facilitate this additional functionality. Figure 3-4 shows the results from executing the tool on a GeoStudio 2007 simulation stored in a Levee Analyst database.



ArcGIS



GeoStudio 2007

Figure 3-4: Updated workflow of the *Export Simulation* geoprocessing tool.





## 4 Case Study: St. Louis National Levee Database

The USACE provided a National Levee Database to this project so that we could demonstrate the compatibility of the Levee Analysis Data Model and the NLD and for testing the functionality of Levee Analyst geoprocessing tools. As provided, the NLD was populated with data from levees in the St. Louis area along the Mississippi, Missouri, and Illinois rivers.

National Levee Databases are not structured to store levee seepage or slope stability simulations. To enable such functionality the NLD must be merged with the LADM. The USACE did not supply any levee seepage or slope stability simulations; consequently, fictitious SEEP2D, UTEXAS, SEEP/W, and SLOPE/W simulations were created and imported into the merged NLD.

The NLD database contains a *borehole\_point* feature class. The NLD database supplied by the USACE did not contain any information in the *borehole\_point* feature class so fictitious borehole points and borelog images were created and imported into the NLD in order to illustrate the functionality of the Levee Analyst geoprocessing tools on a merged NLD.

Once the LADM and the St. Louis NLD were merged, seepage and slope stability simulations created and archived, and the *borehole\_point* feature class populated, the entire St. Louis NLD was exported to KML files. The exported KML files were then opened and browsed in Google Earth.

This case study successfully demonstrated the compatibility of the NLD and the LADM; the capabilities of the original, revised, and additional Levee Analyst geoprocessing tools; and the functionality of the revised *Export to Levee Analyst Database Wizard* tool found in GMS.

#### **4.1 Merging Levee Analyst Schema with the Existing National Levee Database**

The first task in the case study was to merge the NLD personal geodatabase with the Levee Analysis Data Model schema. Merging the NLD geodatabase and LADM schema provided the feature classes and attribute fields required to run the Levee Analyst geoprocessing tools.

##### **4.1.1 Adding the Seepage, Slope Stability, Simulation, and Executables Tables**

The NLD database lacks the capability to store seepage and slope stability simulations. Adding the *Seepage*, *Slope Stability*, and *Simulation* tables provides the desired simulation storage capability. The *Executables* table provides the path to the executable associated with archived simulations – allowing archived simulations to be exported and launched from within ArcGIS. In ArcCatalog the *Seepage*, *Slope Stability*, *Simulation*, and *Executables* tables were copied into the St. Louis NLD from an empty Levee Analyst database (See Figure 4-1).

##### **4.1.2 Adding and Populating Fields in the levee\_centerline and cross\_section\_line Feature Classes**

Figure 4-2 and Figure 4-3 depict the fields and field data types in the *LeveeCenterline* and *CrossSectionLine* feature classes, respectively. The *HydroID*, *HydroCode*, and *Name* fields were added to the *levee\_centerline* feature class in the St. Louis NLD with the data types shown in Figure 4-2. The *HydroID*, *HydroCode*, *Name*, *LeveeCenterlineID*, *CurrentRiverStage*,

*FactorOfSafety*, and *Difference* fields were added to the *cross\_section\_line* feature class in the St. Louis NLD with the data types shown in Figure 4-3.

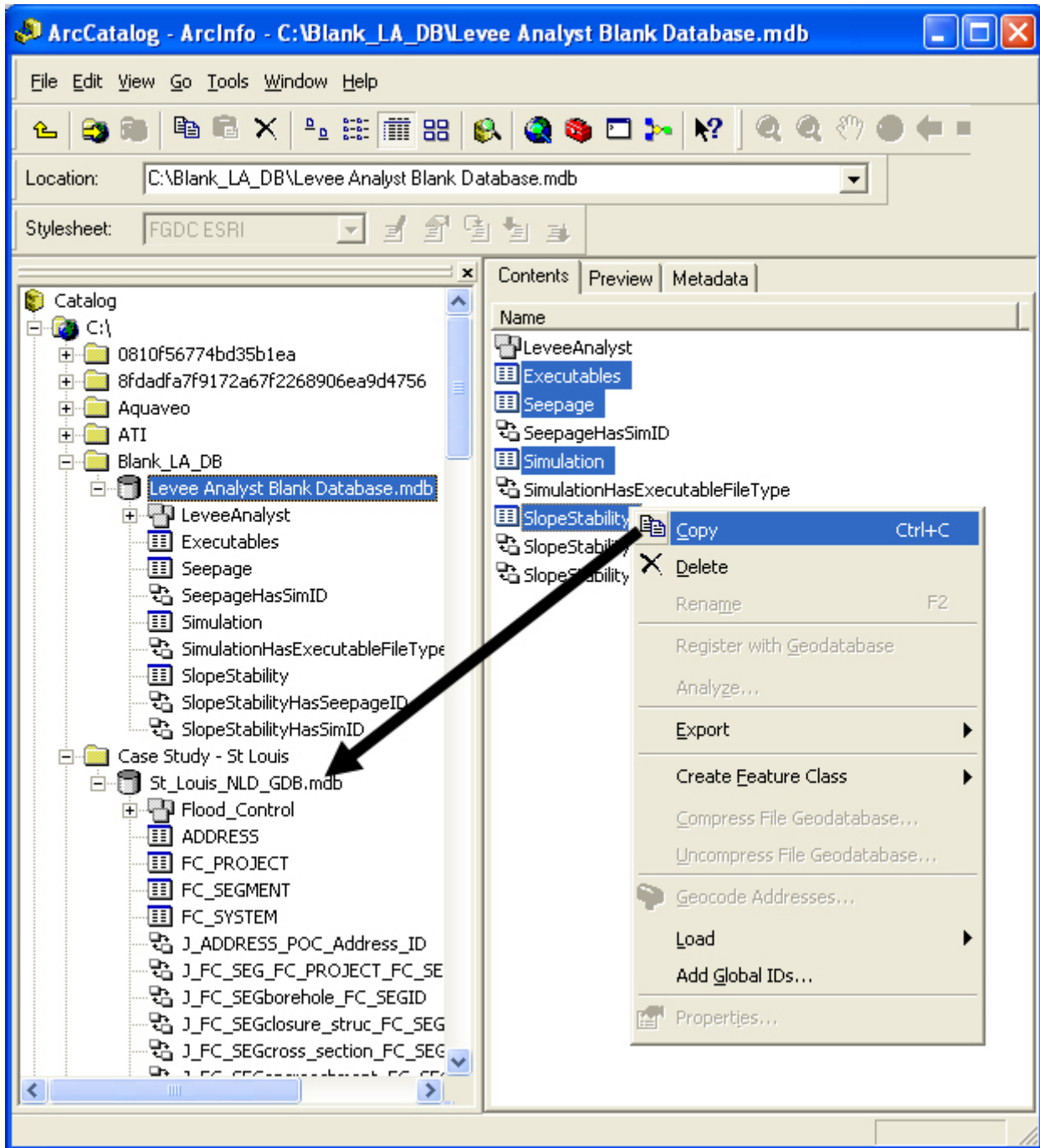


Figure 4-1: Adding the *Seepage*, *Slope Stability*, *Simulation*, and *Executables* tables to the NLD.

Simple feature class				Geometry <i>Polyline</i>			
<b>LeveeCenterline</b>				Contains M values <i>No</i>			
				Contains Z values <i>No</i>			
Field name	Data type	Allow nulls	Default value	Domain	Prec-ision	Scale	Length
OBJECTID	Object ID						
SHAPE	Geometry	Yes					
HydroID	Long integer	Yes			0		
HydroCode	String	Yes					255
Name	String	Yes					255
SHAPE_Length	Double	Yes			0	0	

Figure 4-2: Fields in the *LeveeCenterline* feature class.

Simple feature class				Geometry <i>Polyline</i>			
<b>CrossSectionLine</b>				Contains M values <i>No</i>			
				Contains Z values <i>No</i>			
Field name	Data type	Allow nulls	Default value	Domain	Prec-ision	Scale	Length
OBJECTID	Object ID						
SHAPE	Geometry	Yes					
HydroID	Long integer	Yes			0		
HydroCode	String	Yes					255
Name	String	Yes					255
LeveeCenterlineID	Long integer	Yes			0		
CurrentRiverStage	Double	Yes			0	0	
FactorOfSafety	Double	Yes			0	0	
Difference	Double	Yes			0	0	
SHAPE_Length	Double	Yes			0	0	

Figure 4-3: Fields in the *CrossSectionLine* feature class.

The *HydroID* fields in both feature classes were populated using the *Assign HydroID* geoprocessing tool available in the *Arc Hydro Tools* toolbox. The *HydroCode* fields in both feature classes were populated using the *Field Calculator* by setting the *HydroCode* fields equal to the *Primary Key Identifier* fields. The *Name* field of the *levee\_centerline* field was populated using the *Field Calculator* by setting the *Name* field equal to the *Feature Name* field. The *Name* field of the *cross\_section\_line* field was populated using the *Field Calculator* by setting the

*Name* field equal to the *Levee Stationing Code* field. The *LeveeCenterlineID* field of the *cross\_section\_line* feature class was populated using the *Spatial Join* geoprocessing tool available in the *Analysis Tools* toolbox. The intersect option, available in the *Spatial Join* geoprocessing tool, was used to generate a *LeveeCenterlineID* corresponding to the *HydroID* of the levee the cross section line intersected. The *CurrentRiverStage*, *FactorOfSafety*, and *Difference* fields in the *cross\_section\_line* feature class were left empty as they are populated by Levee Analyst geoprocessing tools using seepage and slope stability simulation data.

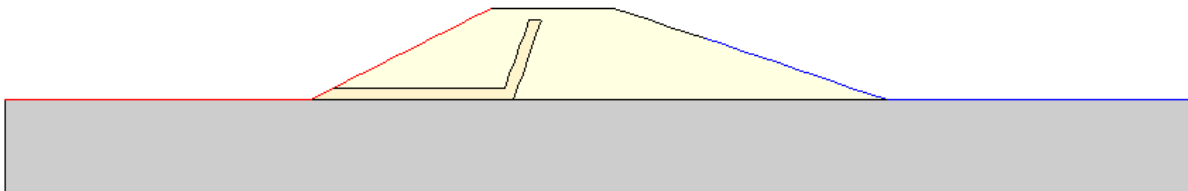
## **4.2 Populating Seepage, Slope Stability, and Simulation Tables**

Once the Levee Analyst database was merged with the St. Louis NLD the latter was ready for the importation of seepage and slope stability simulations.

### **4.2.1 Creating and Exporting SEEP2D and UTEXAS Simulations in GMS**

The next step in the case study was to create a series of SEEP2D and UTEXAS simulations in GMS. The simulations were then archived to the *Seepage*, *Slope Stability*, and *Simulation* tables in the St. Louis NLD using the *Export to Levee Analyst Database Wizard*. To begin, a 15-foot generic levee cross section was first created (See Figure 4-4) and then SEEP2D seepage and UTEXAS slope stability (for both the landside and riverside of the levee) simulations were generated at one foot intervals for every vertical foot along the cross section, creating a set of 15 seepage and 30 slope stability simulations. The seepage and landside slope stability simulations corresponding to 15 feet of head were copied to new folders for 21 of the 498 cross sections in the St. Louis NLD. The seepage and land side slope stability simulations corresponding to 10 feet of head were copied to new folders for 151 of the cross sections. The

original set of 15 seepage and 30 slope stability simulations was copied to new folder for one of the cross sections. In reality each cross section would have unique seepage and slope stability simulations; however, for simplification and time-saving purposes, the same seepage and slope stability simulations were assigned to many of the levee cross sections.



**Figure 4-4: Levee cross section created in GMS.**

The *Export to Levee Analyst Database Wizard* was then used to export the SEEP2D seepage and UTEXAS slope stability simulations, for each of the 173 cross sections folders, to the corresponding levee cross section in the St. Louis NLD.

#### **4.2.2 Creating SEEP/W and SLOPE/W Simulations in GeoStudio 2004 and 2007**

After the SEEP2D and UTEXAS simulations were created in GMS and exported to the St. Louis NLD, a number of SEEP/W and SLOPE/W simulations were created in GeoStudio 2004 and 2007 and then imported into the St. Louis NLD using the *Import GeoStudio Simulation* geoprocessing tool. As in GMS, a generic 15-foot levee cross section was first created in GeoStudio 2004. A single SEEP/W simulation was then created with 11 feet of head (See Figure 4-5). The material properties, geometry, and pore water pressure from the SEEP/W simulation were used to create a GeoStudio 2004 SLOPE/W landside simulation (See Figure 4-6).

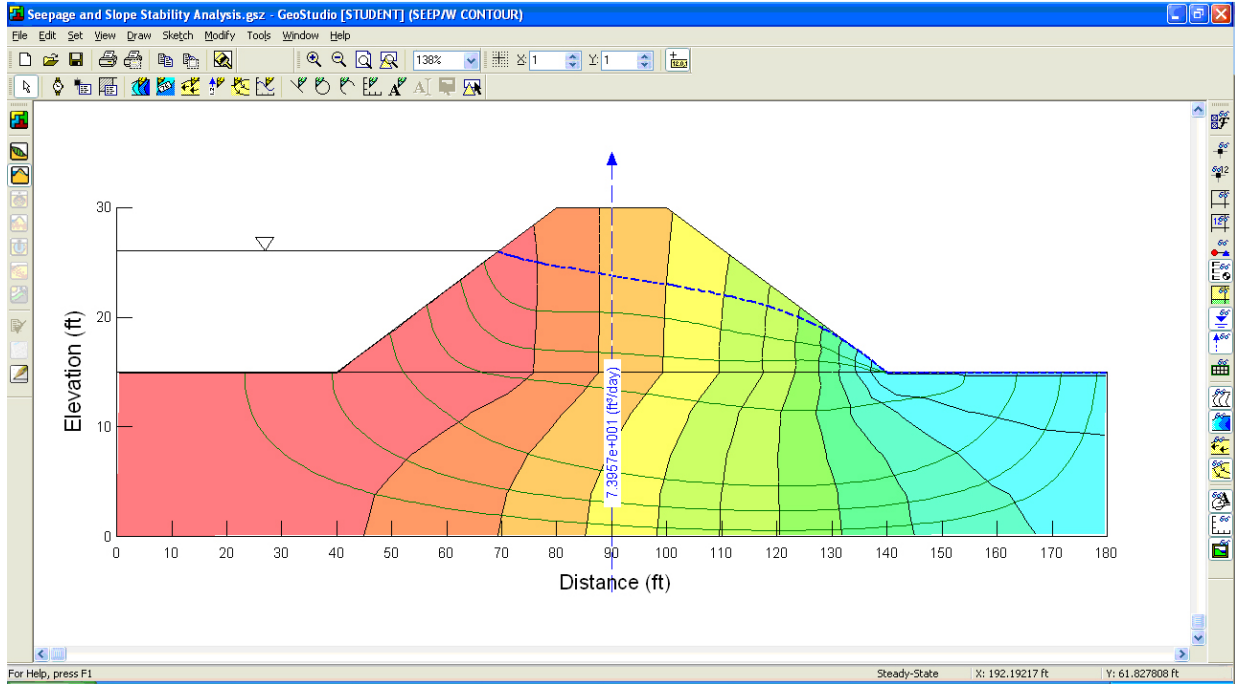


Figure 4-5: Levee cross section SEEP/W analysis developed in GeoStudio 2004.

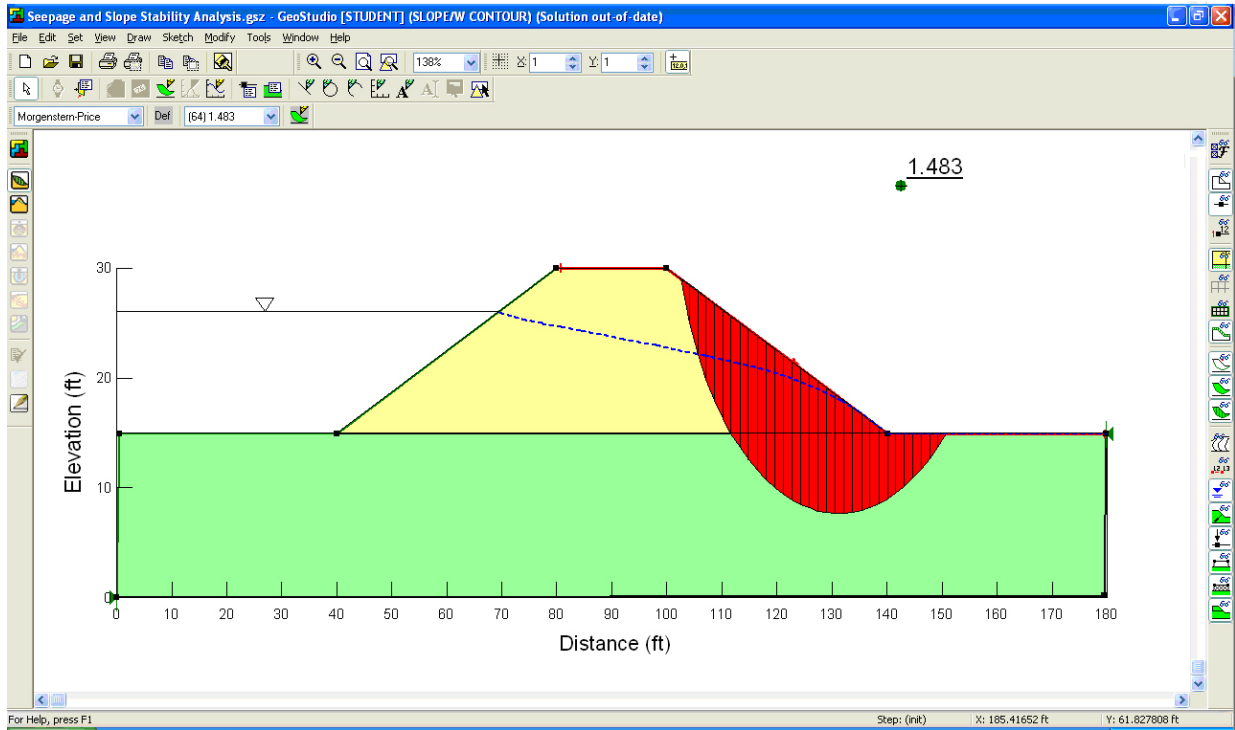


Figure 4-6: Levee cross section SLOPE/W analysis developed in GeoStudio 2004.



In addition to the mandatory Ordinary, Bishop, and Janbu limit equilibrium methods, the Morgenstern-Price method was selected. The SEEP/W and SLOPE/W simulations were saved as a single GeoStudio 2004 project file. The project file was then copied to new folders for 126 of the cross sections in the St. Louis NLD. Each folder was given the name of the corresponding levee cross section HydroID.

After the SEEP/W and SLOPE/W simulations were created in GeoStudio, a series of SEEP/W and SLOPE/W simulations were created in GeoStudio 2007. As in GMS and GeoStudio 2004, a generic 15-foot levee cross section was first created. A series of SEEP/W simulations was then created at one-foot intervals of head. The material properties, geometry, and pore water pressure from the SEEP/W simulations were used to create a corresponding series of GeoStudio 2007 SLOPE/W landside and riverside simulations. Figure 4-7 shows the SEEP/W simulation that was created for 14 feet of head, Figure 4-8 shows the corresponding landside SLOPE/W simulation, and Figure 4-9 shows the corresponding riverside SLOPE/W simulation. In addition to the mandatory Ordinary, Bishop, and Janbu limit equilibrium methods, the Morgenstern-Price method was selected for the landside simulations and the Spencer method was selected for the riverside simulations. The riverside simulations also had the “Optimize critical slip surface location” option enabled.

For each one-foot increment, the SEEP/W, landside SLOPE/W, and riverside SLOPE/W simulations were saved as a single GeoStudio 2007 project file. The GeoStudio 2007 project file for 12-feet of head was copied to new folders for 198 of the cross sections in the St. Louis NLD. All 15 project files were copied into a new folder for one of the cross sections. Each folder was given the name of the corresponding levee cross section HydroID.

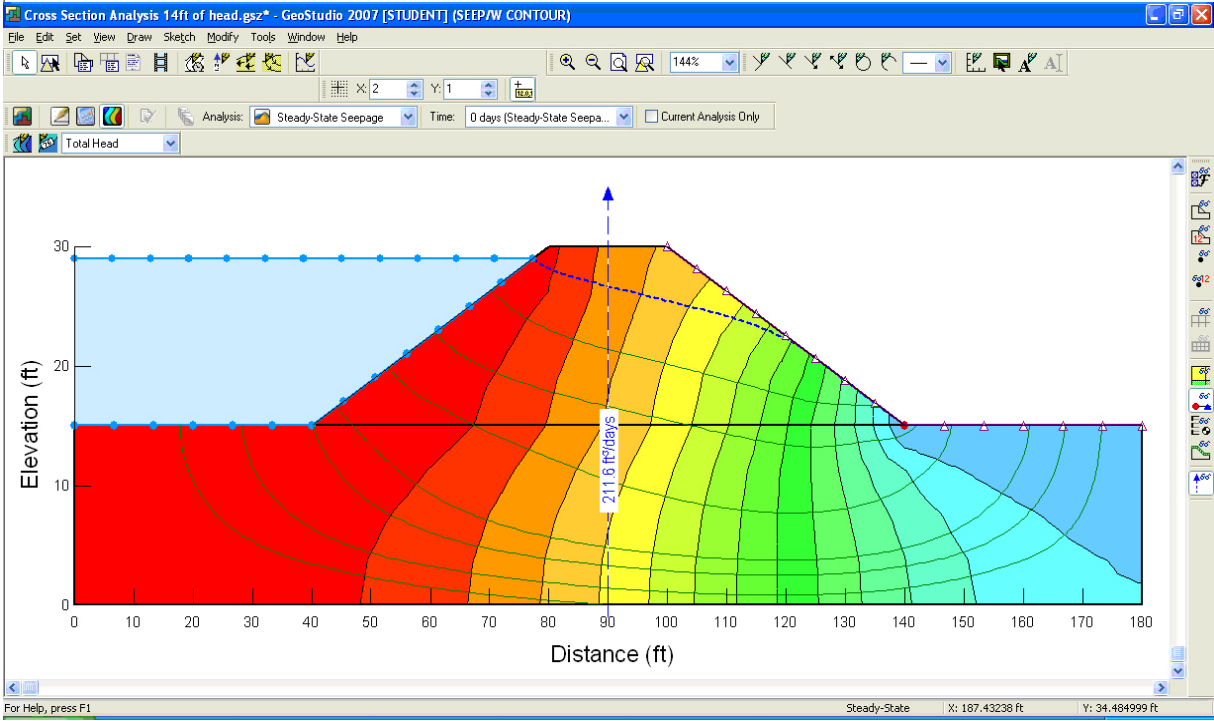


Figure 4-7: Levee cross section SEEP/W analysis developed in GeoStudio 2007.

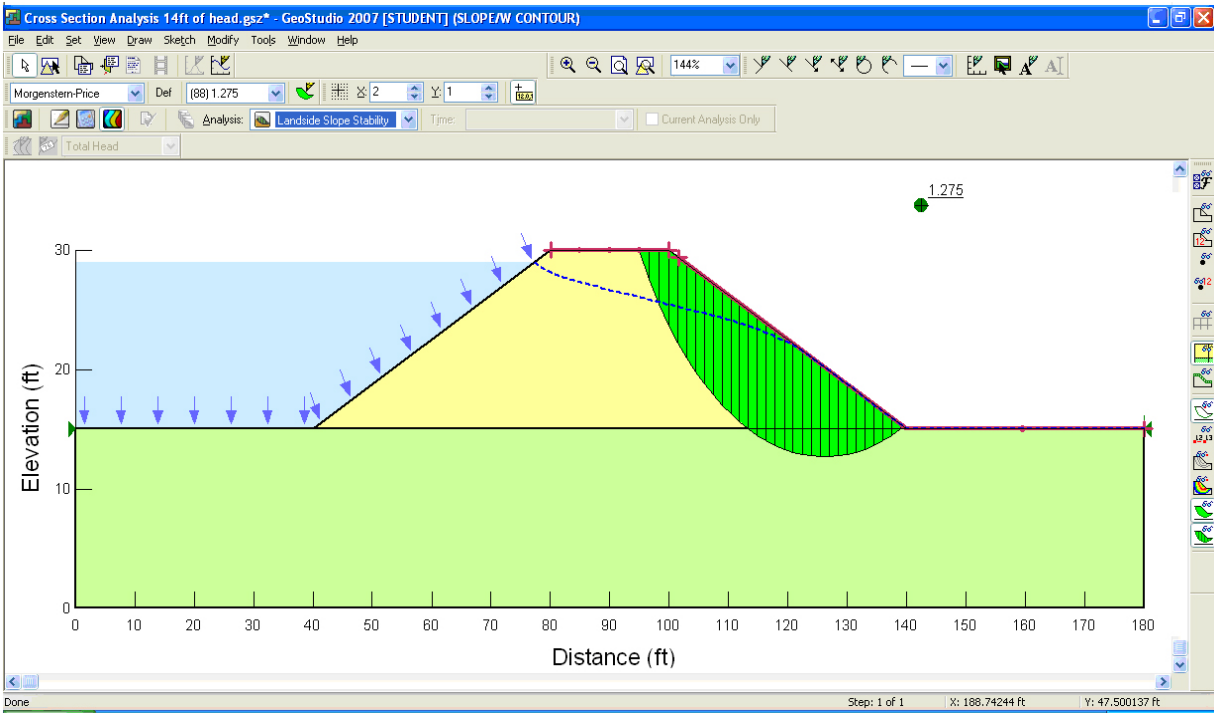


Figure 4-8: Levee cross section landside SLOPE/W analysis developed in GeoStudio 2007.

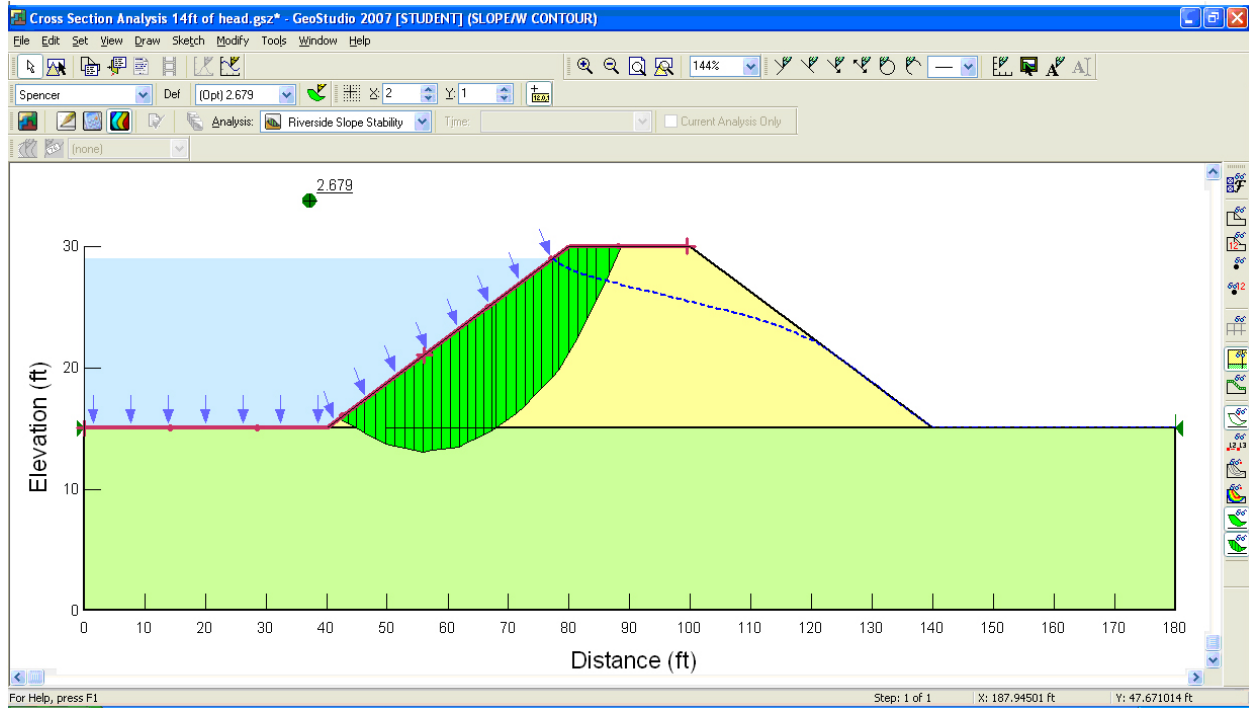


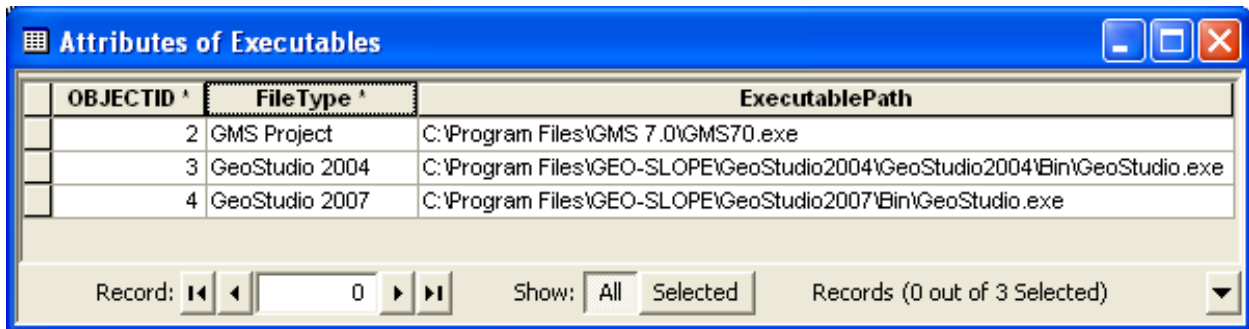
Figure 4-9: Levee cross section riverside SLOPE/W analysis developed in GeoStudio 2007.

#### 4.2.3 Archiving GeoStudio Simulations

The *Import GeoStudio Simulation* geoprocessing tool was then used to import the SEEP/W and SLOPE/W simulations from each folder to the corresponding levee cross section within the St. Louis NLD. The *Import GeoStudio Simulation* geoprocessing tool was executed 339 times, resulting in the importation of seepage and slope stability simulations for 325 levee cross sections. In total, 339 records were created in both the *Simulation* and *Seepage* tables and 2421 records were created in the *Slope Stability* table.

### 4.3 Populating the Executables Table

The *Executables* table stores the file location of the external program executable files for the programs in which the user created the seepage and slope stability simulations. This table must be properly populated for the *Export Simulation* geoprocessing tool to work correctly. For this case study, GMS was used to create, display, and edit the SEEP2D seepage and UTEXAS slope stability simulations while GeoStudio 2004 and 2007 were used to create, display, and edit the SEEP/W seepage and SLOPE/W slope stability simulations. Therefore, the *GMS Project*, *GeoStudio 2004*, and *GeoStudio 2007* coded-value domain values in conjunction with the path for the GMS, GeoStudio 2004, and GeoStudio 2007 executables were stored in the *Executables* table as seen in Figure 4-10.



OBJECTID ^	FileType ^	ExecutablePath
2	GMS Project	C:\Program Files\GMS 7.0\GMS70.exe
3	GeoStudio 2004	C:\Program Files\GEO-SLOPE\GeoStudio2004\GeoStudio2004\Bin\GeoStudio.exe
4	GeoStudio 2007	C:\Program Files\GEO-SLOPE\GeoStudio2007\Bin\GeoStudio.exe

Figure 4-10: *Executables* table with the GMS and GeoStudio program executables stored as table records.

### 4.4 Executing the Simulations Query Geoprocessing Tool

The *Simulations Query* geoprocessing tool queries the *Simulation*, *Slope Stability*, and *Seepage* tables to find all of the seepage and slope stability simulation data associated with a user specified levee cross section. This data is copied to a new, user-named table. To demonstrate

the functionality of the *Simulations Query* geoprocessing tool with SEEP/W and SLOPE/W simulations, the *Simulations Query* tool was executed twice. First, the HydroID of a levee cross section line with archived GeoStudio 2004 simulations was used to query the St. Louis NLD to find all of the corresponding seepage and slope stability simulations. The tool was then executed for a levee cross section line with archived GeoStudio 2007 simulations. The output tables created by running the *Simulations Query* geoprocessing tool for levee cross section lines with GeoStudio 2004 and GeoStudio 2007 simulations are seen in Figure 4-11 and Figure 4-12, respectively. These figures demonstrate that the update of the *Simulations Query* geoprocessing tool was successful as the *Analysis Procedure* fields were properly populated.

SimID	Seepage	SlopeStabilityID	FlowRate	FlowRateU	FactorOfSafety	AnalysisProcedure	RiverStage	Riv	FilePath
416	416	2120	73.957	(ft <sup>3</sup> /d)/(ft)	1.158252	Fellenius or Ordinary	11	(ft)	C:\ST_LOUIS\HydroID196\Seepag
416	416	2121	73.957	(ft <sup>3</sup> /d)/(ft)	1.476997	Bishop Simplified	11	(ft)	C:\ST_LOUIS\HydroID196\Seepag
416	416	2122	73.957	(ft <sup>3</sup> /d)/(ft)	1.309215	Janbu Simplified(no Fo Factor)	11	(ft)	C:\ST_LOUIS\HydroID196\Seepag
416	416	2123	73.957	(ft <sup>3</sup> /d)/(ft)	1.482669	Morgenstern-price	11	(ft)	C:\ST_LOUIS\HydroID196\Seepag

Figure 4-11: *Simulations Query* output table for a GeoStudio 2004 levee cross section.

#### 4.5 Executing the Export Simulation Geoprocessing Tool

After running the *Simulations Query* geoprocessing tool, the user can then use the *Export Simulation* geoprocessing tool to export a seepage or slope stability simulation to the program in which it was created. The *Export Simulation* geoprocessing tool was used to export the first record in the output tables created by the *Simulations Query* geoprocessing tool, as discussed in

Section 4.4. After providing the necessary information in the *Export Simulation* geoprocessing tool, the GeoStudio 2004 project file associated with the first record in the output table shown in Figure 4-11 was launched in GeoStudio 2004 and the GeoStudio 2007 project file associated with the first record in the output table shown in Figure 4-12 was launched in GeoStudio 2007. The results for the GeoStudio 2004 and 2007 simulations are shown in Figure 4-13 and Figure 4-14, respectively.

SimID	Seepage	Slope	FlowRate	FlowRate	FactorOfSafety	AnalysisProcedure	RiverStage	Rive	FilePath
401	401	1985	7.135338	(ft^3/d)/(ft)	1.361634	Fellenius or Ordinary	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1986	7.135338	(ft^3/d)/(ft)	1.640252	Bishop Simplified	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1987	7.135338	(ft^3/d)/(ft)	1.440664	Janbu Simplified(no Fo Factor)	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1988	7.135338	(ft^3/d)/(ft)	1.636827	Morgenstern-price	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1989	7.135338	(ft^3/d)/(ft)	1.367623	Fellenius or Ordinary	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1990	7.135338	(ft^3/d)/(ft)	1.631416	Bishop Simplified	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1991	7.135338	(ft^3/d)/(ft)	1.414376	Janbu Simplified(no Fo Factor)	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1992	7.135338	(ft^3/d)/(ft)	1.623626	Spencer	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
401	401	1993	7.135338	(ft^3/d)/(ft)	1.568653	Spencer_opt	1	(ft)	C:\ST_LOUIS\HydroID630\1 ft
402	402	1994	15.328482	(ft^3/d)/(ft)	1.344448	Fellenius or Ordinary	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	1995	15.328482	(ft^3/d)/(ft)	1.622904	Bishop Simplified	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	1996	15.328482	(ft^3/d)/(ft)	1.424598	Janbu Simplified(no Fo Factor)	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	1997	15.328482	(ft^3/d)/(ft)	1.619598	Morgenstern-price	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	1998	15.328482	(ft^3/d)/(ft)	1.375056	Fellenius or Ordinary	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	1999	15.328482	(ft^3/d)/(ft)	1.646803	Bishop Simplified	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	2000	15.328482	(ft^3/d)/(ft)	1.432302	Janbu Simplified(no Fo Factor)	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	2001	15.328482	(ft^3/d)/(ft)	1.639157	Spencer	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
402	402	2002	15.328482	(ft^3/d)/(ft)	1.572371	Spencer_opt	2	(ft)	C:\ST_LOUIS\HydroID630\2ft
403	403	2003	24.254965	(ft^3/d)/(ft)	1.323971	Fellenius or Ordinary	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2004	24.254965	(ft^3/d)/(ft)	1.602091	Bishop Simplified	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2005	24.254965	(ft^3/d)/(ft)	1.405931	Janbu Simplified(no Fo Factor)	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2006	24.254965	(ft^3/d)/(ft)	1.598907	Morgenstern-price	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2007	24.254965	(ft^3/d)/(ft)	1.389843	Fellenius or Ordinary	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2008	24.254965	(ft^3/d)/(ft)	1.664784	Bishop Simplified	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2009	24.254965	(ft^3/d)/(ft)	1.458685	Janbu Simplified(no Fo Factor)	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2010	24.254965	(ft^3/d)/(ft)	1.654685	Spencer	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
403	403	2011	24.254965	(ft^3/d)/(ft)	1.589351	Spencer_opt	3	(ft)	C:\ST_LOUIS\HydroID630\3ft
404	404	2012	33.862267	(ft^3/d)/(ft)	1.30138	Fellenius or Ordinary	4	(ft)	C:\ST_LOUIS\HydroID630\4ft
404	404	2013	33.862267	(ft^3/d)/(ft)	1.579718	Bishop Simplified	4	(ft)	C:\ST_LOUIS\HydroID630\4ft

Figure 4-12: Simulations Query output table for a GeoStudio 2007 levee cross section.

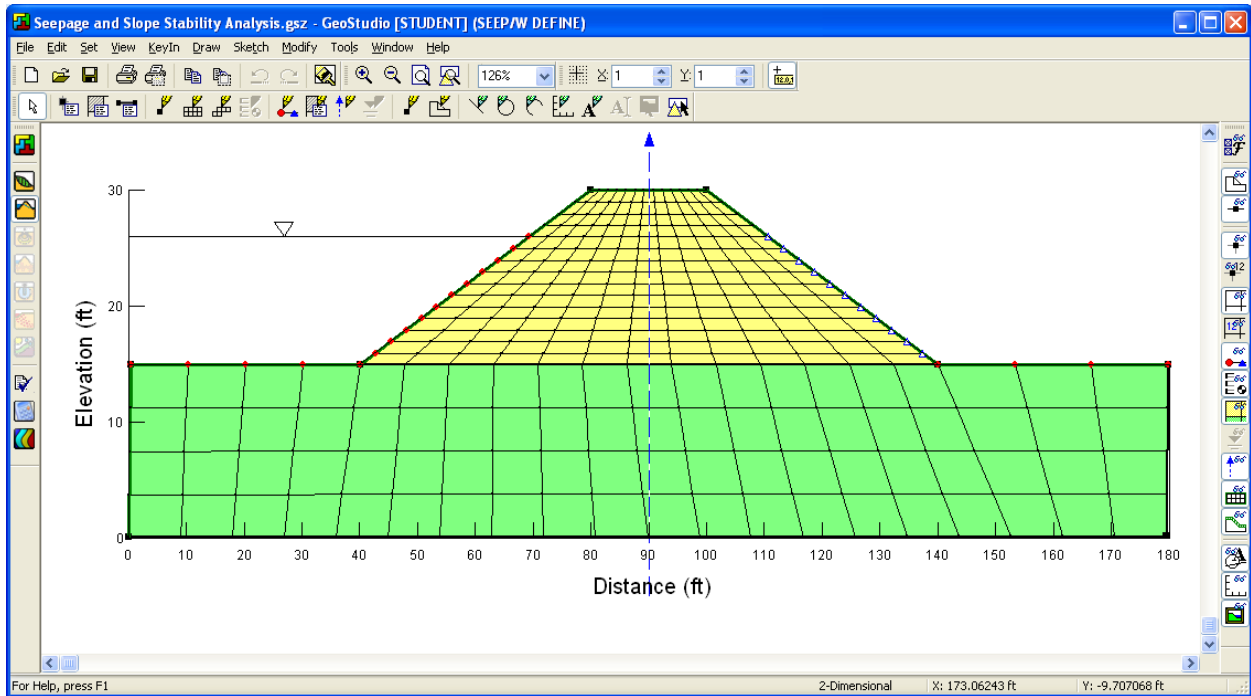


Figure 4-13: Simulation exported from ArcGIS to GeoStudio 2004 using the *Export Simulation* tool.

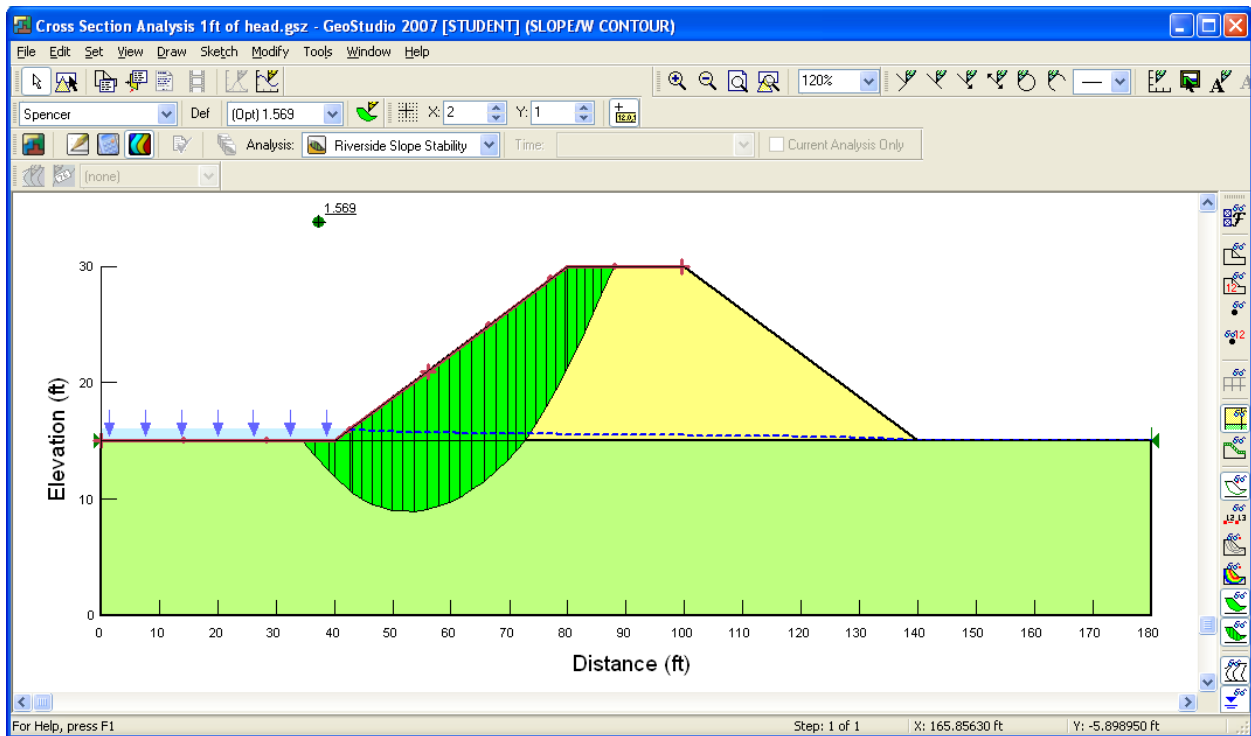


Figure 4-14: Simulation exported from ArcGIS to GeoStudio 2007 using the *Export Simulation* tool.



## 4.6 Populating the borehole\_point Feature Class

The NLD Data Model contains a *borehole\_point* feature class. The *borehole\_point* feature class contains a number of fields for storing information such as drilling start date, drilling completion date, reference elevation, ground surface elevation, total depth, feature description, and boring method. Often levee management agencies will have a digital copy of a borehole log associated with each borehole point in the feature class. Currently, the NLD Data Model does not have a raster field in the *borehole\_point* feature class attribute table for storing the borehole log images. A user must store the images in a separate location on disk. This case study illustrates the potential and convenience of storing images in a raster field of a feature class attribute table and exporting them, along with the feature class geometry and remaining attribute table fields, to a KMZ file – viewable in Google Earth.

### 4.6.1 Adding Additional Fields to the borehole\_point Feature Class

It was necessary to add three fields to the *borehole\_points* feature class. A field entitled *LogImage* was added with the field type of *raster* to hold the borehole log images. A field entitled *ImageType* was added to store the extension of the original images (i.e. “.jpg,” “.png,” “.bmp,” “.gif,” etc.). The file type extension field must be added to enable the *Export Raster From Attribute Field* geoprocessing tool to export and correctly name the images. The final field that was added was entitled *ImagePath*. The *ImagePath* field is used by the *Export Raster From Attribute Field* geoprocessing tool to store an HTML image tag.



#### 4.6.2 Populating the Additional Fields in the *borehole\_point* Feature Class

The St. Louis NLD supplied by the USACE did not contain any borehole points. To populate the *borehole\_point* feature class, 522 borehole points were placed on or near the levee. Borehole log images from another location were used to populate each record in the *LogImage* field of the *borehole\_point* feature class attribute table. From within an editing session, a borehole log image was added to each of the records in the *borehole\_point* feature class attribute table by clicking the value for the raster property (the value is originally <Null> and changes to *Raster* after the image is added), clicking the *Show Raster* button (the small square to the right of the value for the raster property), right-clicking in the new window and clicking *Load*, navigating to the location of the raster dataset, selecting the raster dataset, and clicking *Add* (ESRI, 2008) (See Figure 4-15).

All of the borehole log images were in the Portable Network Graphics (PNG) file format; consequently, the *ImageType* field was populated, using the *Field Calculator*, with “png.”

The *ImagePath* field was not populated as it is populated by the *Export Raster From Attribute Field* geoprocessing tool.

#### 4.7 Executing the Export Raster From Attribute Field Geoprocessing Tool

Once the *borehole\_point* feature class was populated with borehole points and the appropriate fields added and populated in the attribute table, the *Export Raster From Attribute Field* geoprocessing tool was ready to run. The input used for the tool is shown in Figure 4-16. For each borehole point and associated record in the attribute table the tool exported the image stored in the attribute table to the user specified directory (C:\ST\_LOUIS), named the exported image using the entry in the *Borehole\_ID* field, added the file extension found in the *ImageType*





Figure 4-16: *Export Raster From Attribute Field* geoprocessing tool input.

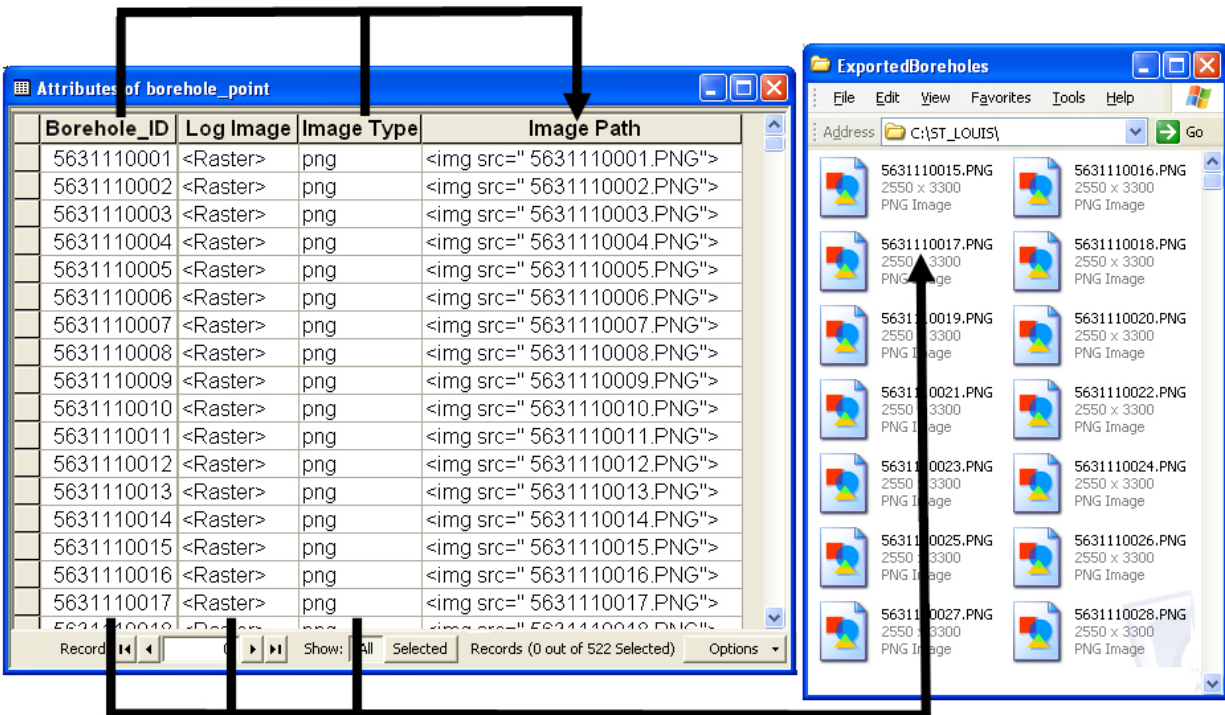


Figure 4-17: *Export Raster From Attribute Field* results.

#### 4.8 Executing the Layer To KML Geoprocessing Tool

The final step in exporting data from a feature class, containing a raster attribute field, to a KMZ file is to execute the *Layer To KML* geoprocessing tool. As illustrated in Section 4.8, the *Export Raster From Attribute Field* placed an HTML image tag in a user specified attribute table field. The *Layer To KML* geoprocessing tool copies each value from the fields in the attribute table associated with each record to the *<description>* tag of each record in the KMZ file. When the KMZ file is opened in Google Earth, the information stored in the *<description>* tag is displayed in the description balloon of a feature when it is clicked. The *Layer To KML* geoprocessing tool requires the user to specify the name and location of the output KMZ file. The file was placed in the same location as the images exported using the *Export Raster From Attribute Field* geoprocessing tool.

#### 4.9 Executing the Export Cross Section To KML Geoprocessing Tool

As described previously, the *Export Cross Section To KML* geoprocessing tool allows the user to export the geometry, attributes, and associated seepage and slope stability analyses (found in the *Seepage*, *Slope Stability*, and *Simulation* tables) for each levee cross section to a KML file. The rasters stored in the *Simulation* table, illustrating the seepage and slope stability analyses, are also exported to the folder containing the KML file.

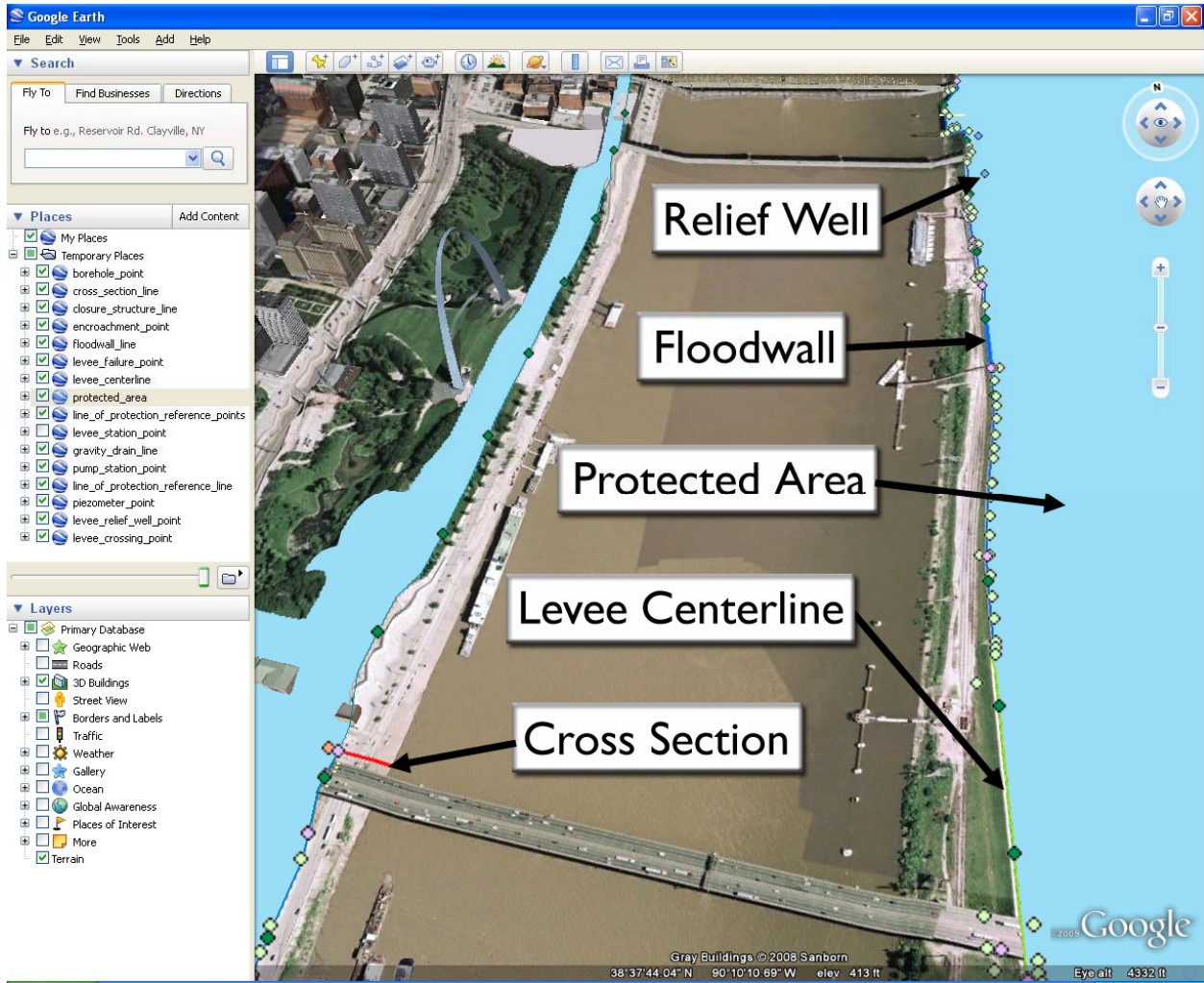
For the St. Louis NLD, the *Export Cross Section To KML* geoprocessing tool was executed. The tool created a KML file with the geometry and attributes for all 498 levee cross sections. The tool also queried the *Seepage*, *Slope Stability*, and *Simulation* tables for records associated with each cross section and appended these records to the KML description balloon tags. Finally, the tool exported 2,623 rasters illustrating the seepage and slope stability analyses.

#### 4.10 Exporting the Remaining Feature Classes

The remaining feature classes found in the St. Louis NLD did not contain rasters in the attribute tables. However, some of the remaining feature classes did contain information in exterior tables in addition to the information stored in the feature class attribute table. To export the remaining feature classes to KMZ files it is necessary for the user to join the desired exterior tables to the feature classes and then to run the *Layer To KML* geoprocessing tool. For the St. Louis NLD, the *Layer To KML* geoprocessing tool was executed for 14 feature classes.

#### 4.11 Viewing the KML and KMZ Files in Google Earth

Once the St. Louis NLD feature classes were exported to KMZ or KML files, they were opened in Google Earth 5.1. Figure 4-18 shows the points, lines, and polygons associated with the various feature classes. In Google Earth the KMZ or KML feature information is displayed in two ways. First, the points, lines, and polygons are visible in the Google Earth 3D Viewer window. A user can navigate the 3D Viewer window by zooming, panning, rotating, and tilting. In this way the user can navigate to a particular feature or browse the terrain to view the spatial location of the various KMZ or KML features. The user then clicks on the point, line, or polygon representing a feature in the 3D Viewer window and the description balloon will be displayed in the 3D Viewer window. Second, the KMZ or KML files are listed in the Places panel. Each file can be expanded to list the features contained in that file. By clicking on a particular feature in the Places panel the feature's description balloon will be displayed in the 3D Viewer window. If a user double-clicks on a feature listed in the Places panel the 3D Viewer window automatically navigates to the feature and displays the feature's description balloon.



**Figure 4-18: St. Louis NLD feature class KMZ and KML files displayed in Google Earth.**

As discussed in Sections 4.8 and 4.9, the St. Louis NLD *borehole\_point* feature class contained rasters in the feature class attribute table. The *Export Raster From Attribute Field* and *Layer To KML* geoprocessing tools were used to export the images and KMZ file to the same directory. The *borehole\_point* KMZ file was among the KMZ files opened in Google Earth and displayed in Figure 4-18. Figure 4-19 shows the description balloon of a *borehole\_point* KMZ feature.



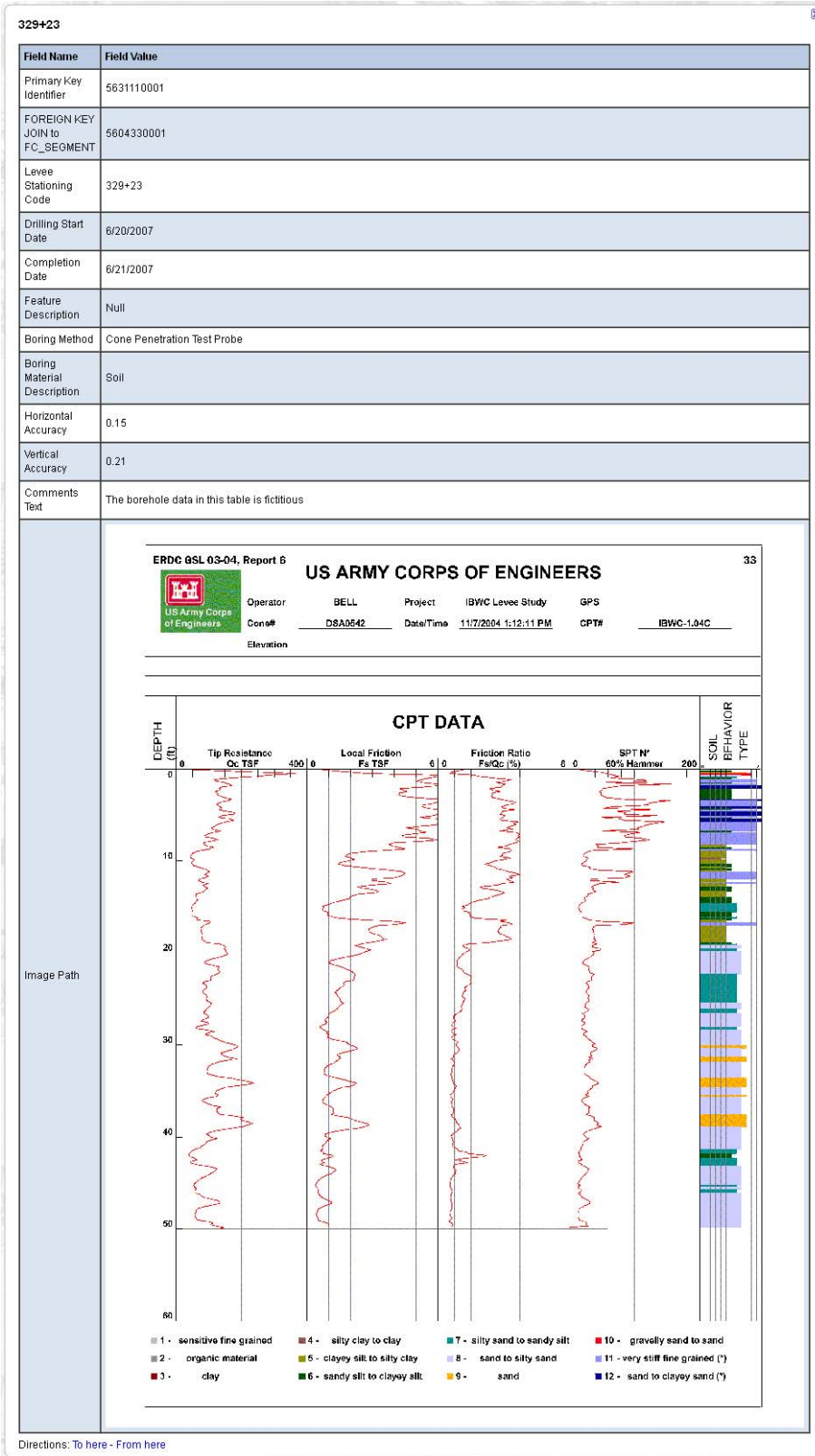


Figure 4-19: Description balloon for a borehole\_point KMZ feature.

As discussed in Section 4.10, the *Export Cross Section To KML* geoprocessing tool was used to export the St. Louis NLD *cross\_section\_line* feature class and associated summaries and images of the seepage and slope stability analyses to a windows directory as a single KML file and associated images. The *cross\_section\_line* KML file was among the files opened in Google Earth and displayed in Figure 4-18. Figure 4-20 shows the description balloon of a *cross\_section\_line* KML feature.

The first table shown in Figure 4-20 contains the information found in the *cross\_section\_line* feature class attribute table for this particular levee cross section. The second table shown in Figure 4-20 contains the seepage and slope stability information found in the *Seepage*, *Slope Stability*, and *Simulation* tables pertaining to this particular levee cross section. In order to reduce the size of the description balloon, the images of the seepage and slope stability analyses are displayed as thumbnails in the description balloon. The user may click a thumbnail to display a larger image in an external browser, external picture editor or viewer, or an internal Google Earth web browser (See Figure 4-21).

In the St. Louis NLD case study, the NLD was merged with the Levee Analyst schema; the *Seepage*, *Slope Stability*, and *Simulation* tables were populated with fictitious seepage and slope stability simulations; SEEP/W and SLOPE/W simulations were exported to GeoStudio, edited, and re-archived to the NLD; the *borehole\_point* feature class attribute table was populated with rasters and then exported to a KMZ file along with the rasters; the *cross\_section\_line* feature class was exported to a KML file along with all of the associated records and rasters from the *Seepage*, *Slope Stability*, and *Simulation* tables; the remaining NLD feature classes were exported to KMZ files; and the KML and KMZ files were opened and browsed in Google Earth. The St. Louis NLD case study illustrated the ability to import seepage



and slope stability simulations and rasters into a NLD and then export that information to Google Earth using the Levee Analyst tools.

Field Name	Field Value
Cross_Sec_ID	5618050003
FC_Segment_ID	5604050001
Levee_Station_Code	478+62
River_Mile	39
Survey_Date	5/31/2007 12:00:00 AM
Coordinate_Method_d	Surveyed using RTK GPS
Horiz_Accuracy	0.13
Vert_Accuracy	0.17
Comments	OUTSIDE ENDS AT EDGE OF WATER INSIDE ENDS AT EDGE OF PAVEMENT
SHAPE_Length	0.00061917148912491
HydroID	187
HydroCode	5618050003
Name	478+62
LeveeCenterlineID	4
CurrentRiverStage	
FactorOfSafety	
Difference	

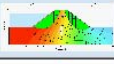
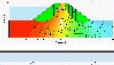
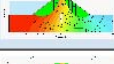
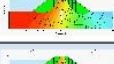

SimID	SeepageID	SlopeStabilityID	FlowRate	FlowRateUnits	FactorOfSafety	AnalysisProcedure	RiverStage	RiverStageUnits	FilePath	File Type	ModelImage
205	205	221	162.34556	(ft <sup>3</sup> /d)/(ft)	1.1034043	Fellenius or Ordinary	12	(ft)	C:\SLM187.gsz	GeoStudio 2007	
205	205	222	162.34556	(ft <sup>3</sup> /d)/(ft)	1.3542727	Bishop Simplified	12	(ft)	C:\SLM187.gsz	GeoStudio 2007	
205	205	223	162.34556	(ft <sup>3</sup> /d)/(ft)	1.1870297	Janbu Simplified (no Fo Factor)	12	(ft)	C:\SLM187.gsz	GeoStudio 2007	
205	205	224	162.34556	(ft <sup>3</sup> /d)/(ft)	1.3418958	Morgenstern-price	12	(ft)	C:\SLM187.gsz	GeoStudio 2007	
205	205	225	162.34556	(ft <sup>3</sup> /d)/(ft)	1.8291679	Fellenius or Ordinary	12	(ft)	C:\SLM187.gsz	GeoStudio 2007	

Figure 4-20: Description balloon for a cross\_section\_line KML feature.

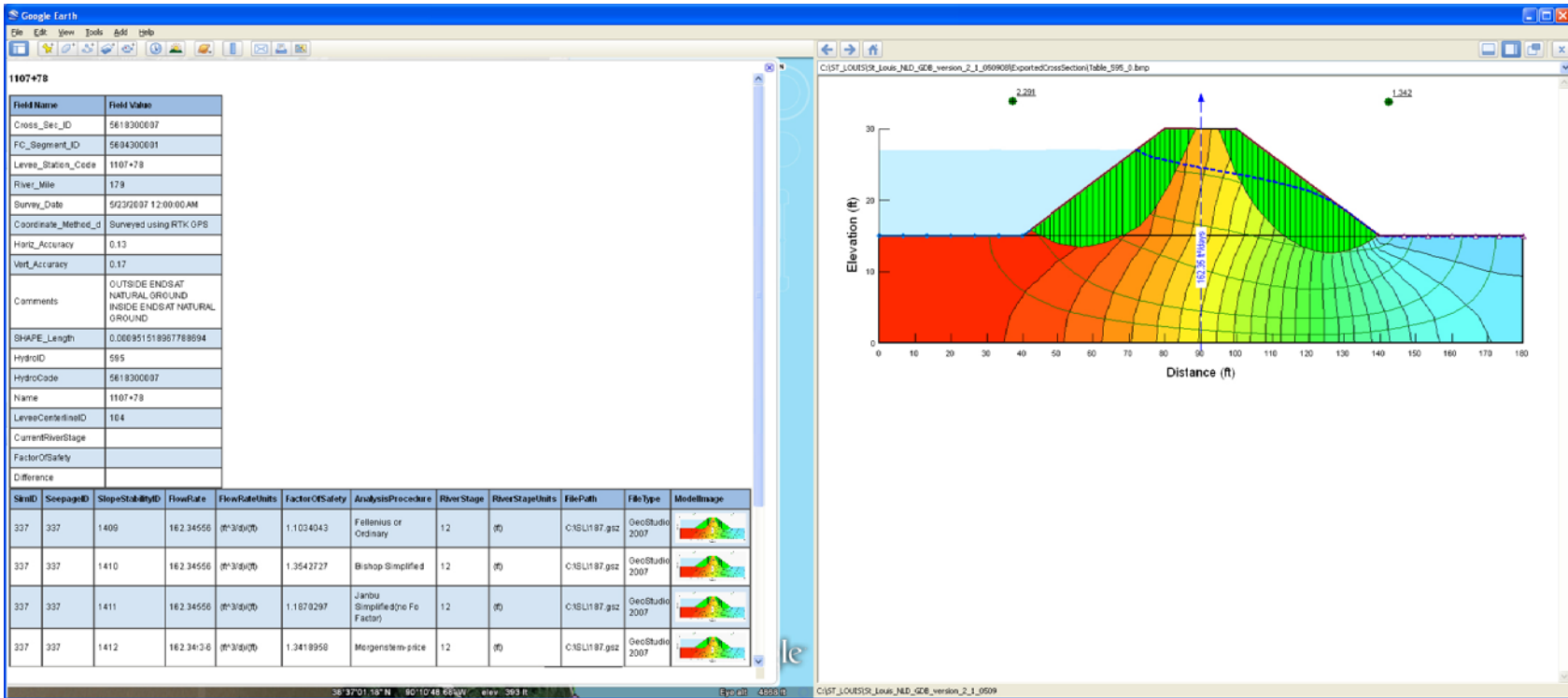


Figure 4-21: Levee cross section displayed in Google Earth web browser.



## 5 Case Study: Southern Texas Levee Data

In addition to the St. Louis data, the USACE provided levee data from a region in southern Texas along the Rio Grande River. Unlike the St. Louis NLD, the provided southern Texas data did not follow any particular schema; therefore, a blank Levee Analysts geodatabase was created and used to import the Texas data. Additionally, the Texas data included actual UTEXAS simulations for four levee cross sections. These four UTEXAS simulations were imported into the Levee Analyst geodatabase along with fictitious SEEP2D, UTEXAS, SEEP/W, and SLOPE/W simulations from the St. Louis case study. The Texas data also included images of 138 cone penetration tests (CPT) along with a shapefile containing a point for each CPT. The CPT shapefile was imported into the Levee Analyst geodatabase. The CPT images were imported to the *CPT* feature class attribute table. The data did not contain ArcGIS files of the levee centerline or levee cross sections. The levee centerline and levee cross section lines were digitized to the *LeveeCenterline* and *CrossSectionLine* feature classes, respectively. Finally, the feature classes in the Levee Analyst geodatabase were exported to Google Earth.

This case study was successful in demonstrating the capabilities of the revised LADM; the original, revised, and additional Levee Analyst geoprocessing tools; and the revised GMS *Export to Levee Analyst Database Wizard* tool.

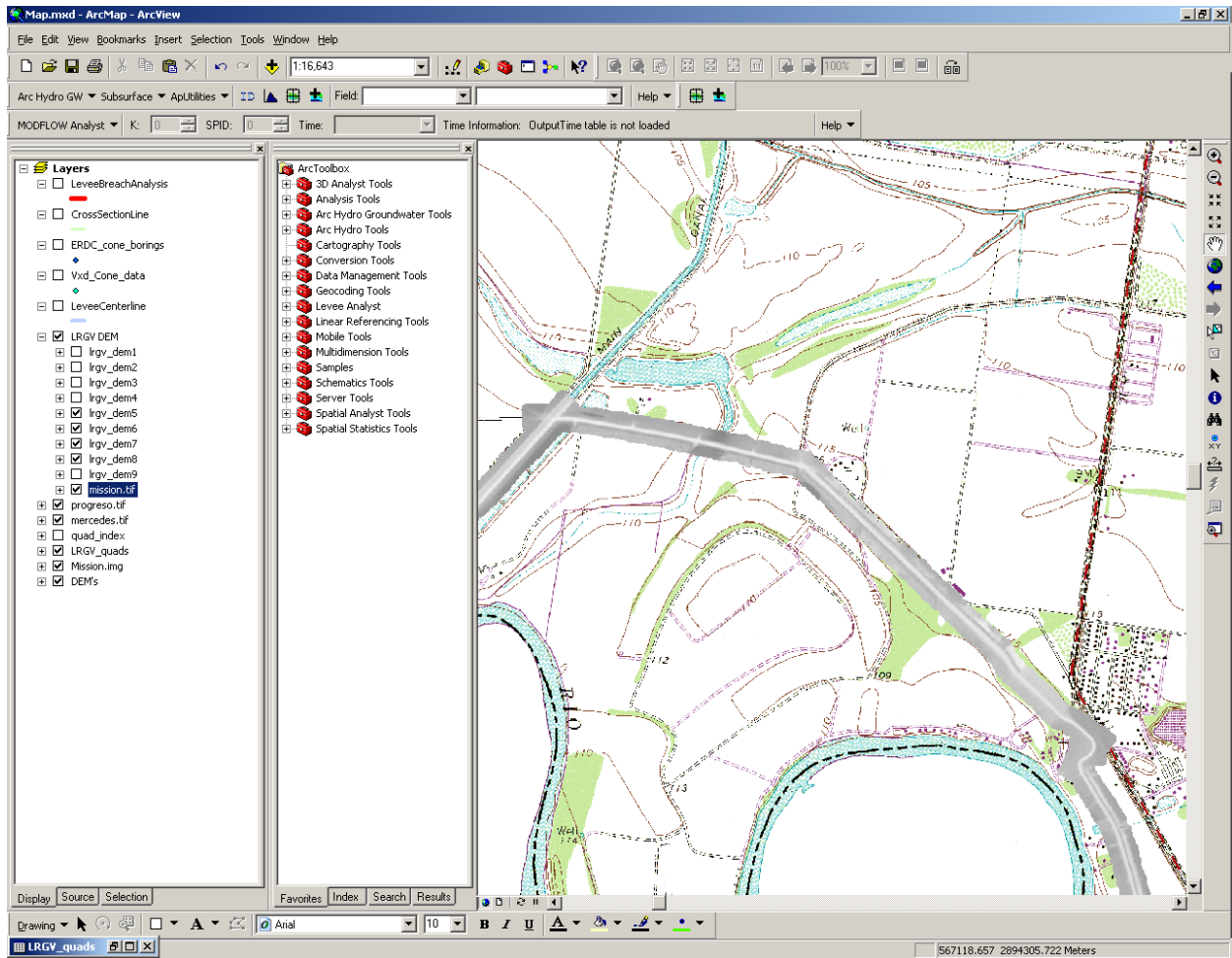
## 5.1 Creating the Levee Analyst Geodatabase

The first step in this case study was to create a new personal geodatabase and import the Levee Analysis Data Model schema, stored in an XML document, into the new personal geodatabase. Importing the schema creates all of the necessary Levee Analyst feature classes and tables.

## 5.2 Digitizing the Levee Centerline and Levee Cross Sections

The Texas levee data contained a raster catalog of digital elevation model (DEM) rasters. These DEM rasters were cropped to only include the levees in the region (See Figure 5-1). In ArcGIS, the DEM rasters were used to create polylines representing the levee centerlines in the region. A total of 167 levee centerline polylines were created and added to the *LeveeCenterline* feature class. Each levee centerline polyline was assigned a unique *HydroID* and *HydroCode* in the appropriate attribute table field of the *LeveeCenterline* feature class. The *Name* field of the attribute table was populated using the name of the United States Geological Survey (USGS) quad in which each levee centerline was located. The southern Texas levee centerline polylines, along with the associated USGS quads, are shown in Figure 5-2.

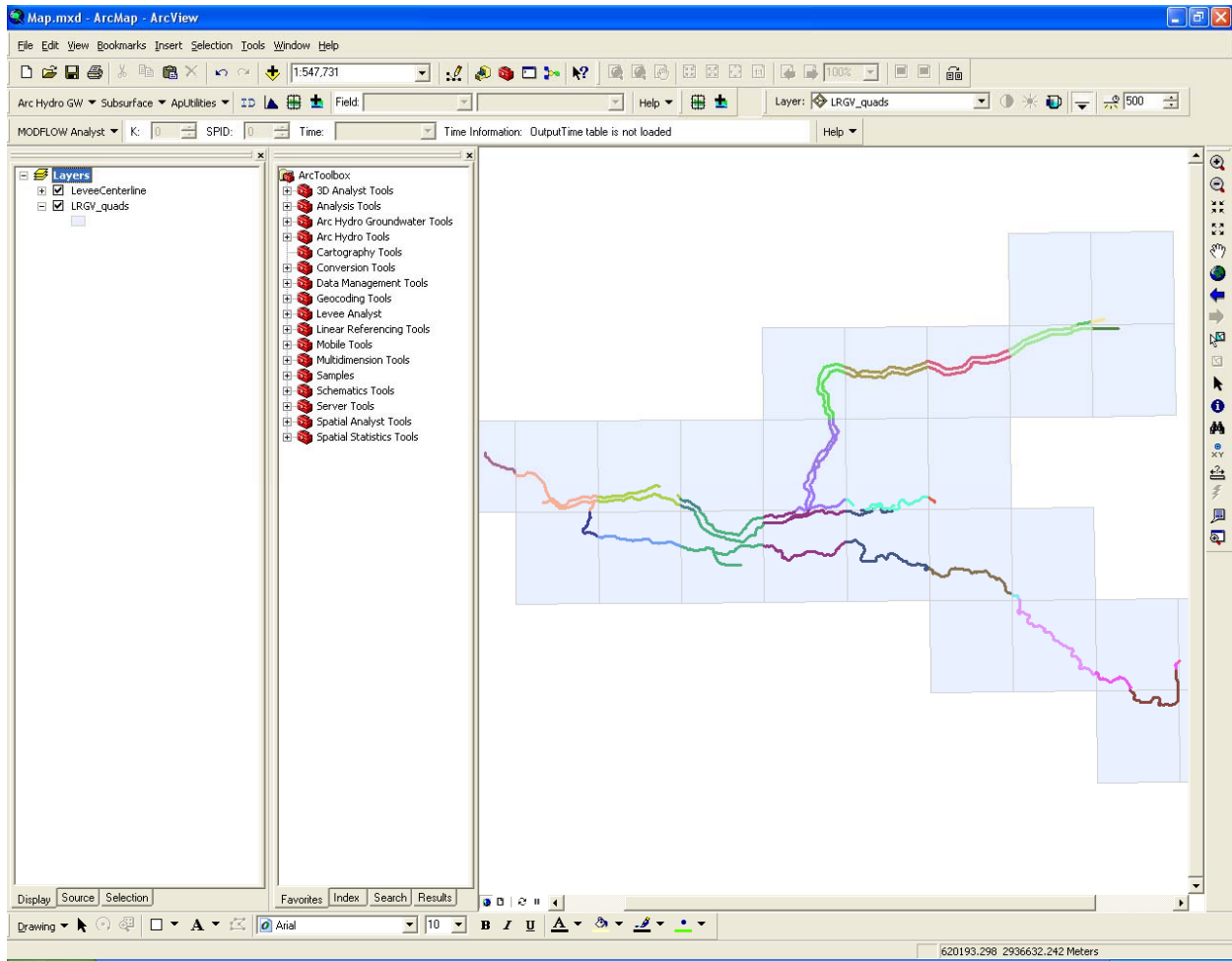
Once the levee centerline was digitized, 61 cross sections polylines were created in the *CrossSectionLine* feature class. Of these, 57 were placed arbitrarily throughout the region and four were drawn to coincide with the location of levee cross sections for which the USACE had performed UTEXAS slope stability simulations. The *HydroID*, *HydroCode*, *Name*, and *LeveeCenterlineID* fields in the *CrossSectionLine* feature class were populated for each cross section



**Figure 5-1: Levee DEM in southern Texas.**

### 5.3 Archiving UTEXAS and SEEP2D Simulations

The USACE supplied UTEXAS slope stability simulations for four levee cross sections. UTEXAS simulations were supplied for the riverside of the levee at one cross section, the landside of the levee for two cross sections, and both the riverside and landside of the levee for the final cross section. The USACE UTEXAS files were first opened in GMS (See Figure 5-3) and then exported to the Texas Levee Analyst geodatabase using the *Export to Levee Analyst Database Wizard*.



**Figure 5-2: Southern Texas levee centerline polylines shown in ArcGIS.**

To further demonstrate the capabilities of the revised LADM and the revised GMS *Export to Levee Analyst Database Wizard* tool, 61 fictitious SEEP2D seepage and 61 fictitious UTEXAS slope stability simulations, corresponding to 32 levee cross sections, were exported to the Texas Levee Analyst geodatabase using the GMS *Export to Levee Analyst Database Wizard* tool. These fictitious simulations were among those used in the St. Louis case study.

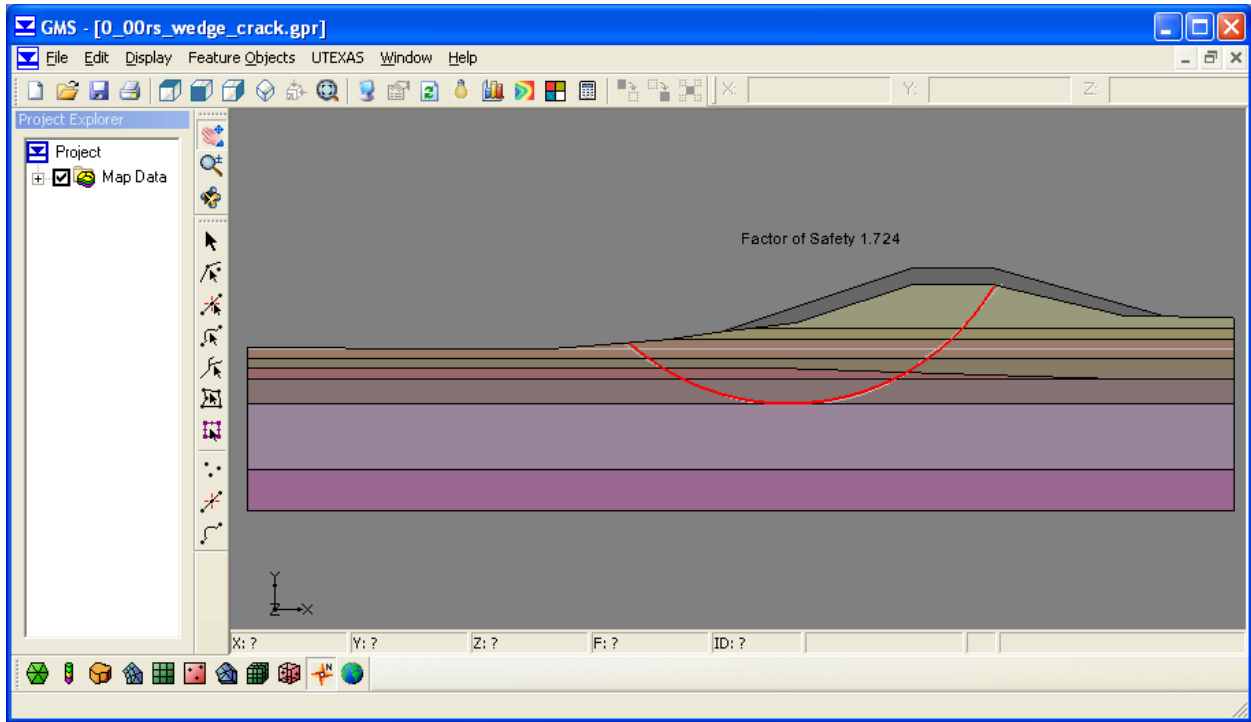


Figure 5-3: USACE UTEXAS simulation as seen in GMS.

#### 5.4 Archiving GeoStudio Simulations

To further demonstrate the capabilities of the revised LADM and the new *Import GeoStudio Simulation* geoprocessing tool, 32 fictitious 2007 SEEP/W seepage, 7 fictitious 2004 SEEP/W seepage, 288 fictitious 2007 SLOPE/W slope stability, and 288 fictitious 2007 SLOPE/W slope stability simulations, corresponding to 39 levee cross sections, were imported into the Texas Levee Analyst geodatabase using the *Import GeoStudio Simulation* geoprocessing tool. These fictitious simulations were also among those used in the St. Louis case study.



## 5.5 Executing the Export Cross Section To KML Geoprocessing Tool

Once the *CrossSectionLine* feature class was created and populated and the *Seepage*, *Slope Stability*, and *Simulation* tables populated, the *Export Cross Section To KML* geoprocessing tool was executed. The tool created a KML file with the geometry and attributes for all 61 levee cross sections. The tool also queried the *Seepage*, *Slope Stability*, and *Simulation* tables for records associated with each cross section and appended these records to the KML description balloon tags. Finally, the tool exported 382 rasters illustrating the seepage and slope stability analyses.

## 5.6 Populating and Exporting the Borehole Feature Class

The USACE supplied a shapefile representing the location of 138 CPT test locations and an image of the CPT log for each location. A raster field cannot be added to a shapefile; consequently, it was necessary to import the shapefile into the Texas Levee Analyst geodatabase. This was accomplished using the *Feature Class to Feature Class* geoprocessing tool found in the ArcGIS Conversion toolbox.

### 5.6.1 Creating Additional Attribute Table Fields

With the CPT shapefile imported into the Texas Levee Analyst Geodatabase it was then possible to add a field to the *CPT* feature class attribute table entitled *Log\_Image*, with the field type of *raster*, to hold the borehole log images. A field entitled *Image\_Type* was added to store the extension of the original image. Finally, a field entitled *Image\_Path* was added to store an HTML image tag.

## 5.6.2 Populating the Raster Field

The 138 CPT boring log images, each corresponding to one of the 138 CPT locations, were imported into the raster field of the *CPT* feature class attribute table as demonstrated in Section 4.6.2.

## 5.6.3 Executing the Export Raster From Attribute Table Geoprocessing Tool

Once the appropriate fields were added and populated in the *CPT* feature class attribute table, the *Export Raster From Attribute Field* geoprocessing tool was ready to run. The tool was executed as demonstrated in Section 4.7. For each CPT point and associated record in the attribute table the tool exported the image stored in the attribute table to the user specified directory (C:\TEXAS), named the exported image using the entry in the *BORING* field, added the file extension found in the *Image\_Type* field to the exported image, and placed an HTML image tag in the *Image\_Path* field.

## 5.6.4 Executing the Layer To KML Geoprocessing Tool

The final step in exporting data from a feature class, containing a raster attribute field, to a KMZ file is to execute the *Layer To KML* geoprocessing tool. The *Layer To KML* geoprocessing tool was executed as illustrated in Section 4.8. The KMZ file was placed in the same location as the CPT images exported using the *Export Raster From Attribute Field* geoprocessing tool.

## 5.7 Exporting the Remaining Feature Classes

The remaining feature classes found in the Texas Levee Analyst geodatabase did not contain rasters in the attribute tables. Additionally, none of the remaining feature classes contained information in exterior tables in addition to the information stored in the feature class attribute table. To export the remaining feature classes to KMZ files it was only necessary to run the *Layer To KML* geoprocessing tool for each feature class. For the Texas Levee Analyst geodatabase, the *Layer To KML* geoprocessing tool was executed for two feature classes.

## 5.8 Viewing the KMZ and KML Files in Google Earth

After the Texas Levee Analyst geodatabase feature classes were exported to KMZ or KML files, they were opened in Google Earth 5.1. Figure 5-4 shows both the points and lines associated with the various feature classes and the description balloon for a levee cross section for which the USACE provided a UTEXAS simulation.

In the southern Texas case study, existing levee data was imported into the Levee Analyst geodatabase; the *Seepage*, *Slope Stability*, and *Simulation* tables were populated with actual and fictitious seepage and slope stability simulations; the *CPT* feature class attribute table was populated with rasters and then exported to a KMZ file along with the rasters; the *CrossSectionLine* feature class was exported to a KML file along with all of the associated records and rasters from the *Seepage*, *Slope Stability*, and *Simulation* tables; the remaining Levee Analyst feature classes were exported to KMZ files; and the KML and KMZ files were opened and browsed in Google Earth. The southern Texas case study illustrated the ability to import actual slope stability simulations and rasters into a Levee Analyst geodatabase and then export that information to Google Earth using the Levee Analyst tools.

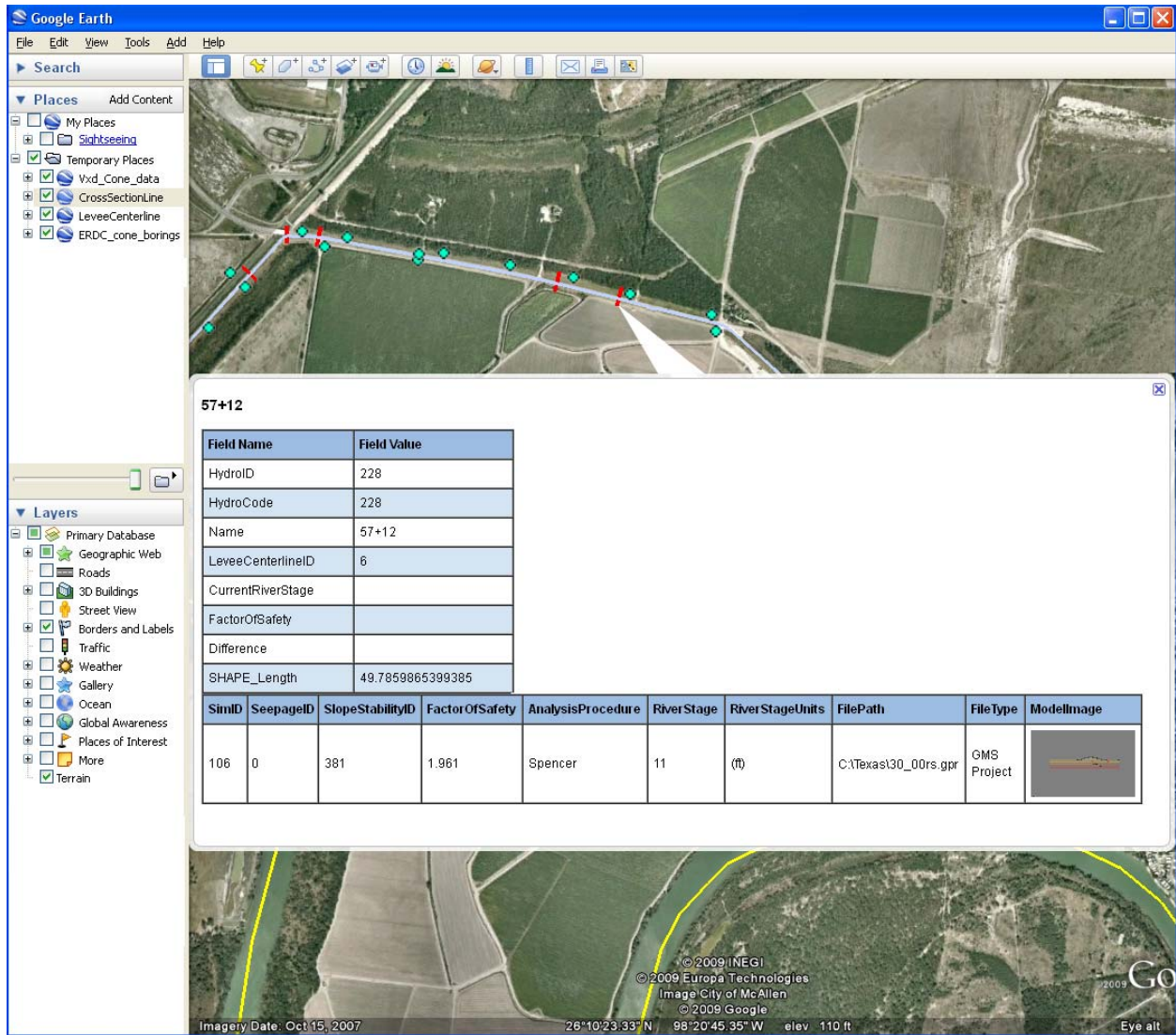


Figure 5-4: Texas Levee Analyst geodatabase feature class KMZ and KML files displayed in Google Earth.



## 6 Automated Levee Breach Simulator Prototype

The final objective of this research was to investigate the feasibility of developing ArcGIS geoprocessing tools to create, archive, edit, execute, and export flood inundation models that simulate the breaching of levees due to river flooding, hurricane storm surges, or similar catastrophic events. A number of numerical models exist that could be used to create flood inundation simulations. This research investigates the development of ArcGIS geoprocessing tools based upon the Gridded Surface/Subsurface Hydrologic Analysis (GSSHA) numeric model. GSSHA is the USACE, Engineer Research and Development Center (ERDC) grid-based, two-dimensional hydrologic model. GSSHA simulates 2D overland flow, 1D stream flow, 1D infiltration, 2D groundwater, and full coupling between groundwater and surface water (Downer, 2010).

For any given location in a land area adjacent to a levee system, flood inundation simulations developed with GSSHA would allow users to estimate the timing and depth of inundating flood waters, the duration of static flood waters, and the timing of flood water recession. Such information would be a powerful tool for evacuation planners, levee maintenance directors, and emergency response directors.

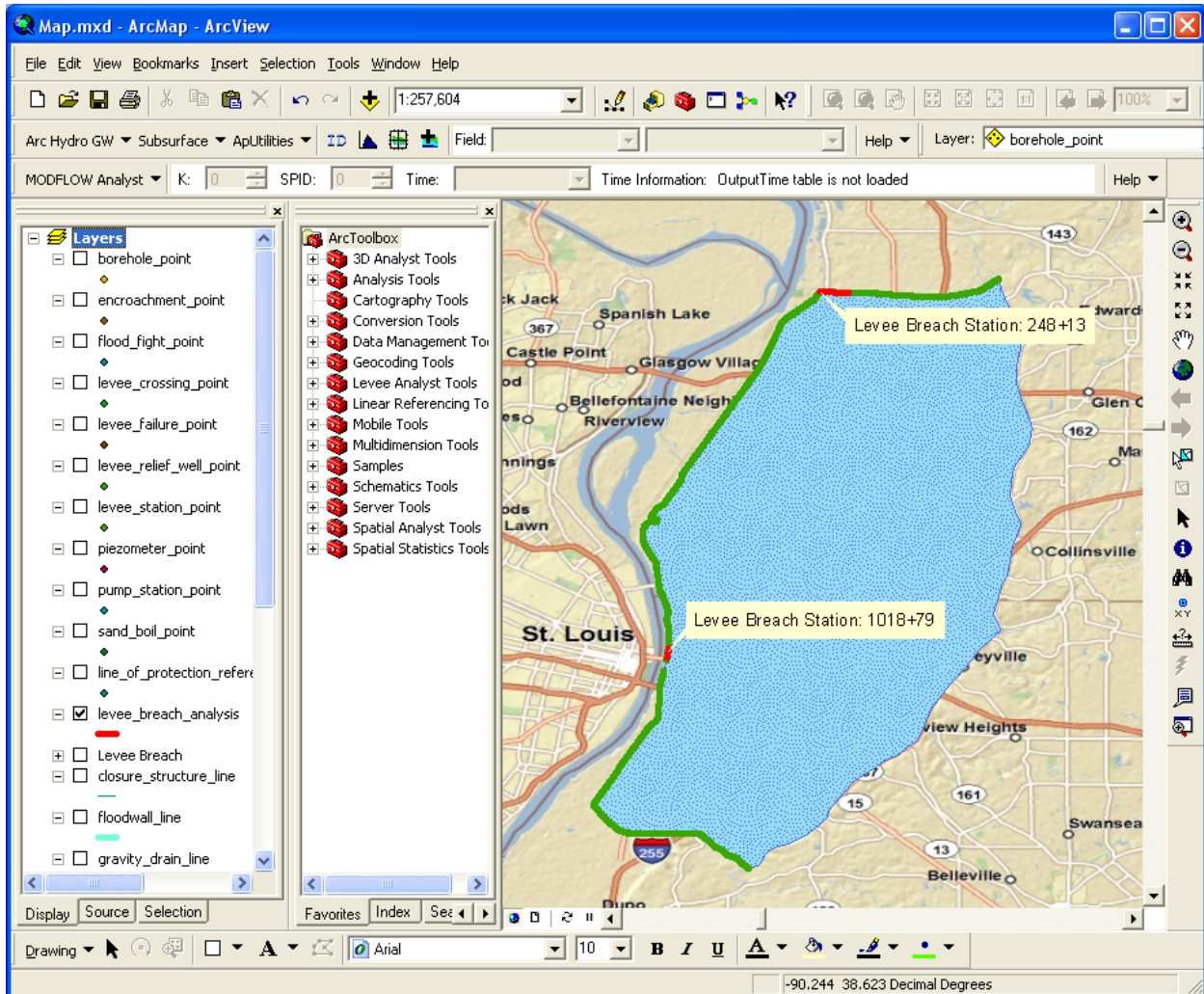
To demonstrate the function and usefulness of flood inundation geoprocessing tools several GSSHA models were manually developed for the St. Louis and southern Texas areas. For each GSSHA flood inundation simulation a number of tasks were performed. First, the

required inputs for the creation of a GSSHA model adjacent to a levee system were determined; second, the GSSHA model was constructed using Watershed Modeling System (WMS) software (WMS, 2009); third, the process of determining, establishing, and modifying boundary conditions was investigated; and fourth, the GSSHA model was executed and the results exported to KMZ files – viewable in Google Earth.

### 6.1 St. Louis Levee Breach Simulations Demonstration

The NLD schema includes a *protected\_area* polygon feature class. The *protected\_area* feature class polygons represent the area that could be inundated should an adjacent levee fail or experience overtopping. In the St. Louis NLD, the *protected\_area* polygon covering the East St. Louis area (See the blue polygon in Figure 6-1) was selected for two GSSHA flood inundation simulations. In Figure 6-1, the levee system protecting the East St. Louis area is shown in green and the locations of the simulated levee breaches are shown in red. Because both levee breach locations coincide with the same *protected\_area* polygon it was only necessary to create one GSSHA model. Once the GSSHA model was created, the boundary conditions were modified, once for both flood inundation simulations, to reflect the simulated levee breach conditions. The boundary conditions were arbitrary and do not necessarily represent a realistic supply of flood water. For both simulations, the GSSHA model was executed and the results exported to a KMZ file.





**Figure 6-1: St. Louis NLD levee breach flood inundation simulations locations.**

In prototyping the GSSHA flood inundation simulation geoprocessing tools, a feature class was created that illustrated the location of the two levee breaches. The feature class was then exported to a KML file. Associated with each levee breach in the KML file was a description balloon. The description balloons provided basic flood inundation simulation summaries. The description balloons for Levee Breach Station: 248+13 and 1018+79 are shown in Figure 6-2 and Figure 6-3, respectively.



**248+13**

Field Name	Field Value
Cross Section	248+13
Normal River Stage	402 ft
River Flood Stage	436 feet
Levee Crest	452 feet
Simulation Duration	2000 minutes
Maximum Flood Depth	63.86 feet
Breach Length	2015 feet
<b>Flood Simulations</b>	
<a href="#">GSSHA: 2000 Minutes - 248+13 - 8/03/2009</a>	

Figure 6-2: St. Louis Levee Breach Station: 248+13 flood inundation simulation summary.

**1018+79**

Field Name	Field Value
Cross Section	1018+79
Normal River Stage	392 ft
River Flood Stage	418 feet
Levee Crest	420 feet
Simulation Duration	2000 minutes
Maximum Flood Depth	28.86 feet
Breach Length	1200 feet
<b>Flood Simulations</b>	
<a href="#">GSSHA: 2000 Minutes - 1018+79 - 8/03/2009</a>	

Figure 6-3: St. Louis Levee Breach Station: 1018+79 flood inundation simulation summary.

Both simulations simulated a levee breach with a 2000 minute (33.3 hours) duration. The results were exported to KMZ files and were opened in Google Earth. Google Earth users can open KMZ animations and watch the results at various speeds, altitudes, and angles. Screenshots of both simulations were obtained from Google Earth showing the KMZ animations at a point 24 hours from the beginning of the flood inundation simulation (See Figure 6-4 and Figure 6-5).

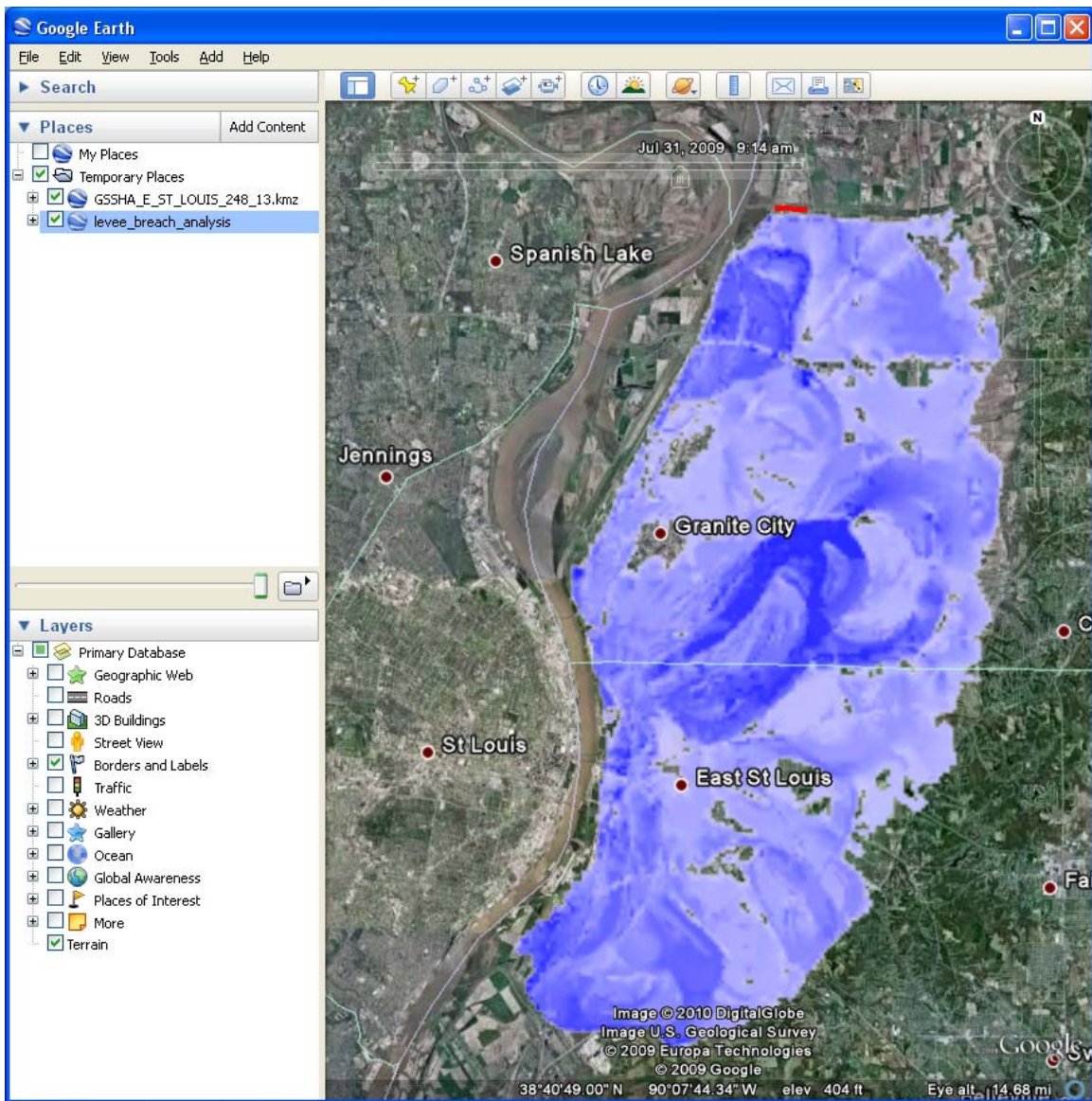


Figure 6-4: St. Louis KMZ flood inundation simulation results for Station: 248+13 as seen in Google Earth.



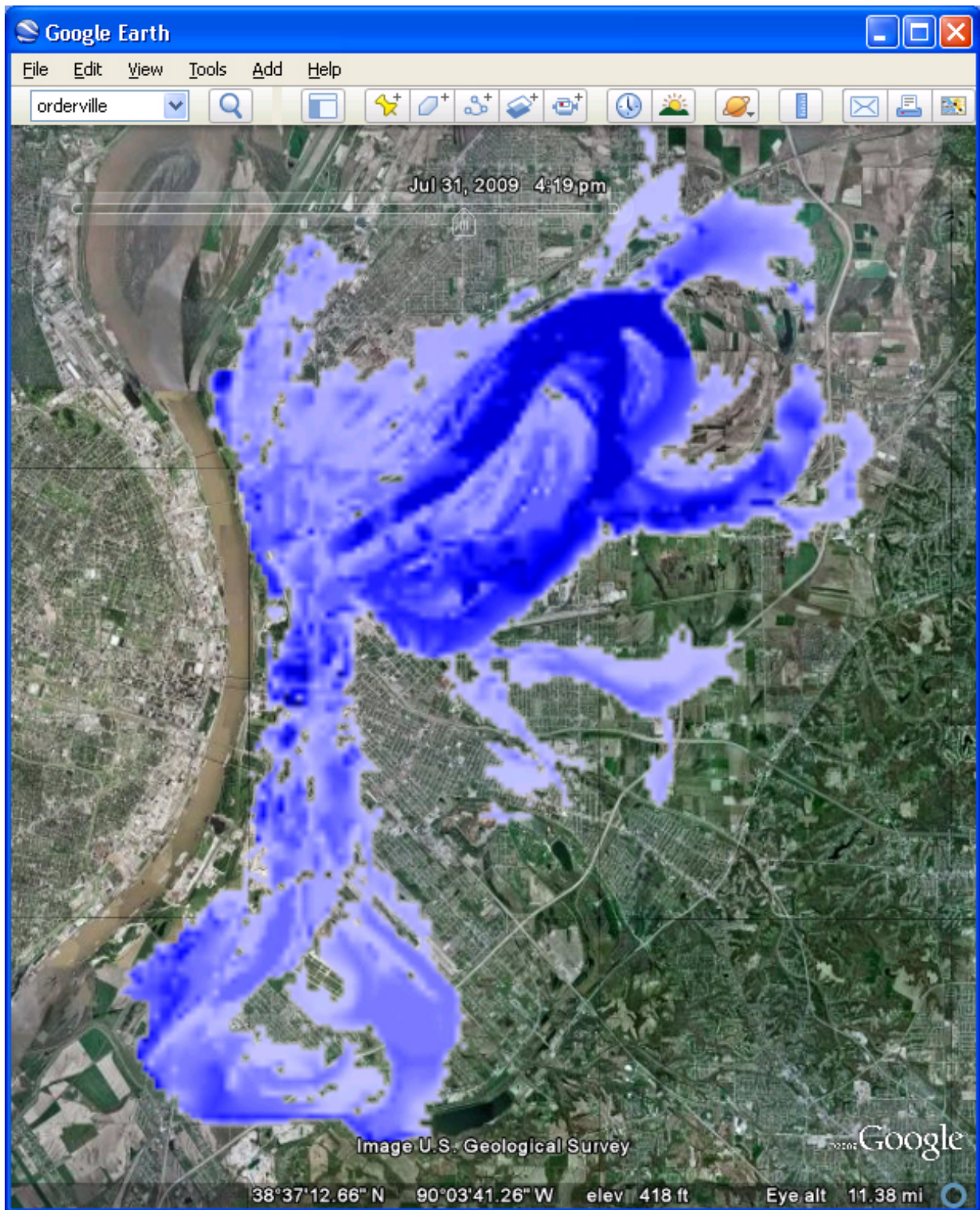


Figure 6-5: St. Louis KMZ flood inundation simulation results for Station: 1018+79 as seen in Google Earth.

## 6.2 Southern Texas Levee Breach Simulations Demonstration

The levee data provided for the southern Texas area did not follow the NLD schema; consequently, a *protected\_area* polygon was not available to use as a model domain. Elevation contours were used to develop a suitable shapefile representing the model domain for the area of interest adjacent to the levee system in the Lower Rio Grande Valley. Four locations along the levee system were selected as levee breach locations for flood inundation simulations. As in the St. Louis flood inundation simulations, a single GSSHA model was developed and the boundary conditions modified to simulate each of the levee breach scenarios. As in the St. Louis case study, the boundary conditions were arbitrary and do not necessarily represent a realistic supply of flood water. Figure 6-6 shows the locations of the southern Texas flood inundation simulations.

As in the St. Louis flood inundation simulations, a feature class was created that illustrated the location of the levee breaches and stored levee simulation metadata. Again, the feature class was exported to a KML file. Associated with each levee breach in the KML file was a description balloon. The description balloons provided basic flood inundation simulation information. The description balloons for Levee Breach: 1, 2, 3, and 4 are shown in Figure 6-7, Figure 6-8, Figure 6-9, and Figure 6-10, respectively.

The duration for the southern Texas flood inundation simulations varied from 6000 to 7000 minutes. The results were exported to KMZ files and were opened in Google Earth. Screenshots of each simulation were obtained from Google Earth showing the KMZ animations at a point 24 hours from the beginning of the flood inundation (See Figure 6-11, Figure 6-12, Figure 6-13, and Figure 6-14).

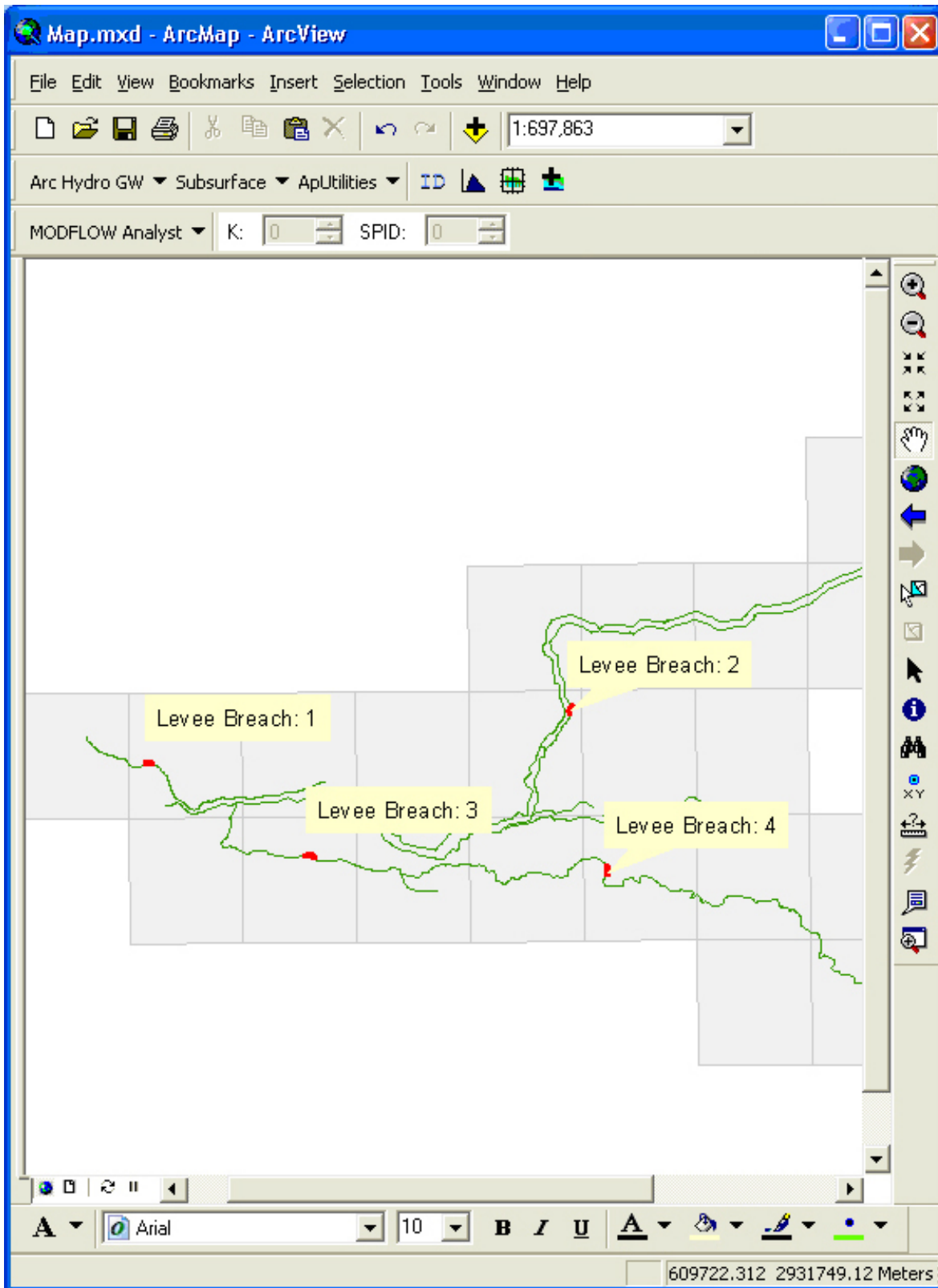


Figure 6-6: Southern Texas levee breach flood inundation simulations locations.

Levee Breach: 1	
Field Name	Field Value
USGS_QD_ID	26098-B4
Breach_Len	782 feet
Norm_Elevation	105 feet
Flood_Stage	126 feet
Levee_Crest	130 feet
Simulation_Duration	7000 minutes
Max_Depth	43 feet
Flood Simulations	
<a href="#">GSSHA: 7000 Minutes - 47+85 - 8/03/2009</a>	

Figure 6-7: Southern Texas Levee Breach: 1 flood inundation simulation summary.

Levee Breach: 2	
Field Name	Field Value
USGS_QD_ID	26097-B8
Breach_Len	855 feet
Norm_Elevation	50 feet
Flood_Stage	68 feet
Levee_Crest	72 feet
Simulation_Duration	6000 minutes
Max_Depth	37 feet
Flood Simulations	
<a href="#">GSSHA: 6000 Minutes - 2782+34 - 8/03/2009</a>	

Figure 6-8: Southern Texas Levee Breach: 2 flood inundation simulation summary.

**Levee Breach: 3**

Field Name	Field Value
USGS_QD_ID	26098-A2
Breach_Len	1578 feet
Norm_Elevation	79 feet
Flood_Stage	100 feet
Levee_Crest	105 feet
Simulation_Duration	6500 minutes
Max_Depth	42 feet
<b>Flood Simulations</b>	
<a href="#">GSSHA: 6500 Minutes - 354+27 - 8/03/2009</a>	

Figure 6-9: Southern Texas Levee Breach: 3 flood inundation simulation summary.

**Levee Breach: 4**

Field Name	Field Value
USGS_QD_ID	26097-A7
Breach_Len	1223 feet
Norm_Elevation	63 feet
Flood_Stage	76 feet
Levee_Crest	77 feet
Simulation_Duration	6000 minutes
Max_Depth	41 feet
<b>Flood Simulations</b>	
<a href="#">GSSHA: 6000 Minutes - 773+21 - 8/03/2009</a>	

Figure 6-10: Southern Texas Levee Breach: 4 flood inundation simulation summary.



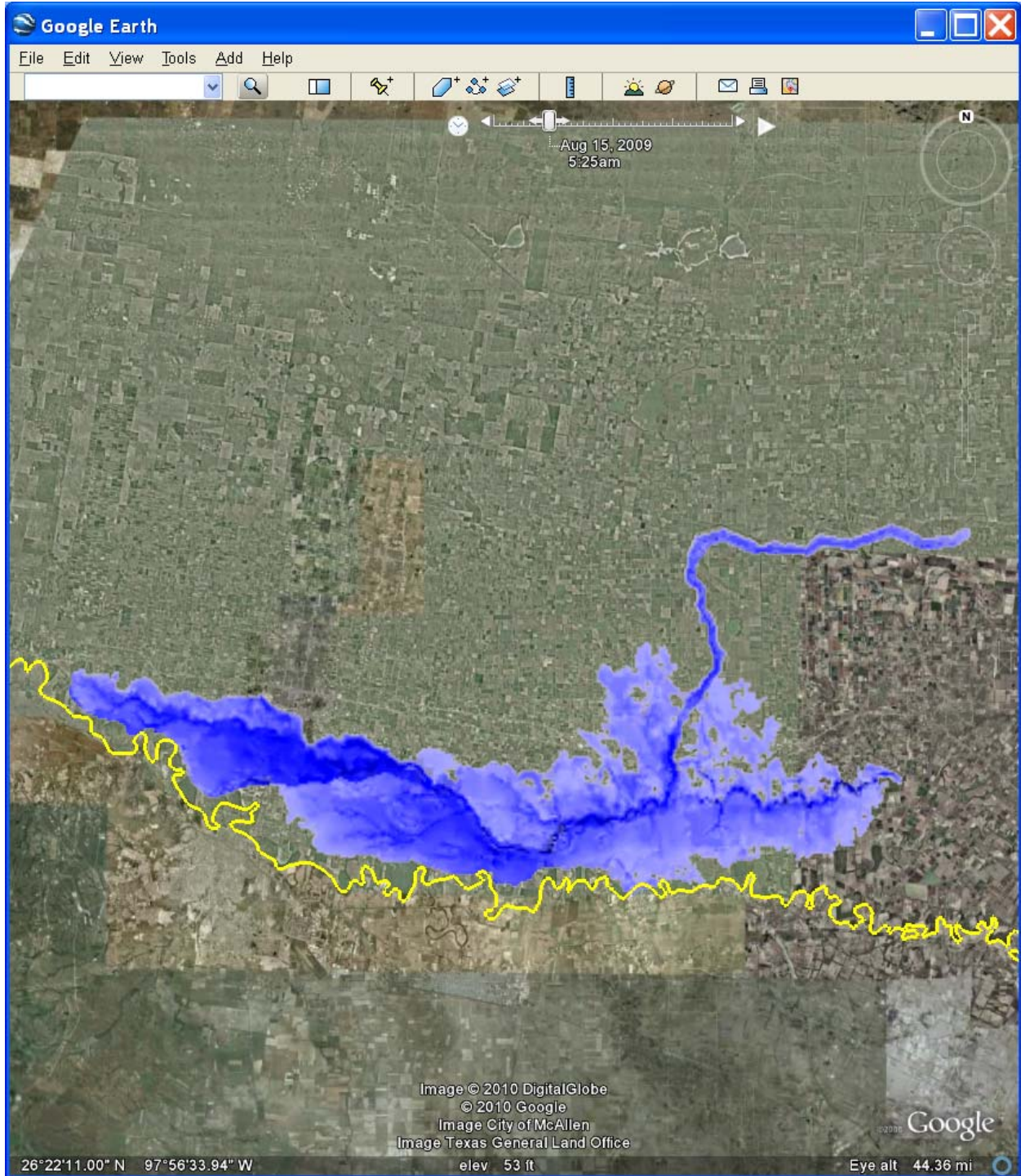


Figure 6-11: Texas KMZ flood inundation simulation results for Levee Breach: 1 as seen in Google Earth.



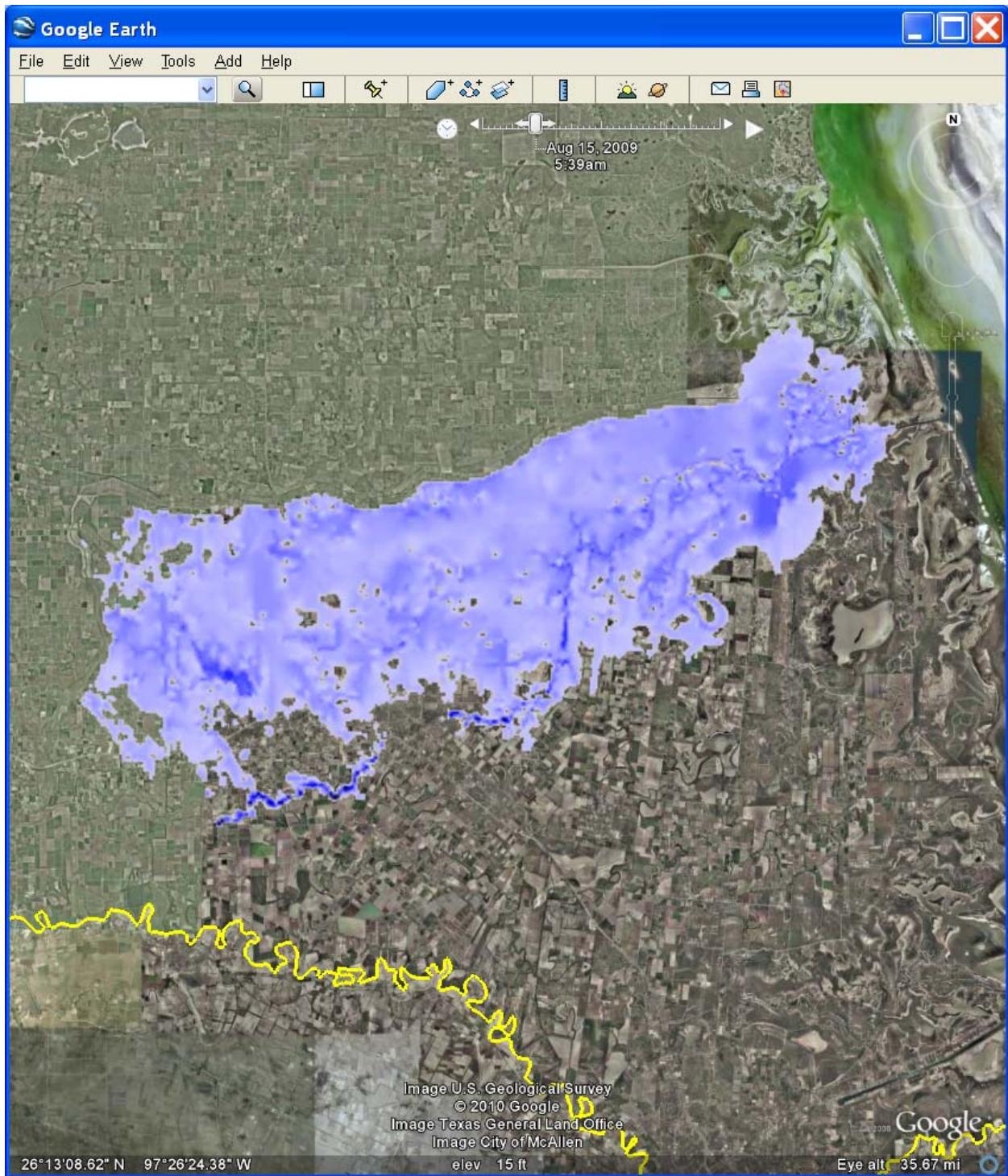


Figure 6-12: Texas KMZ flood inundation simulation results for Levee Breach: 2 as seen in Google Earth.



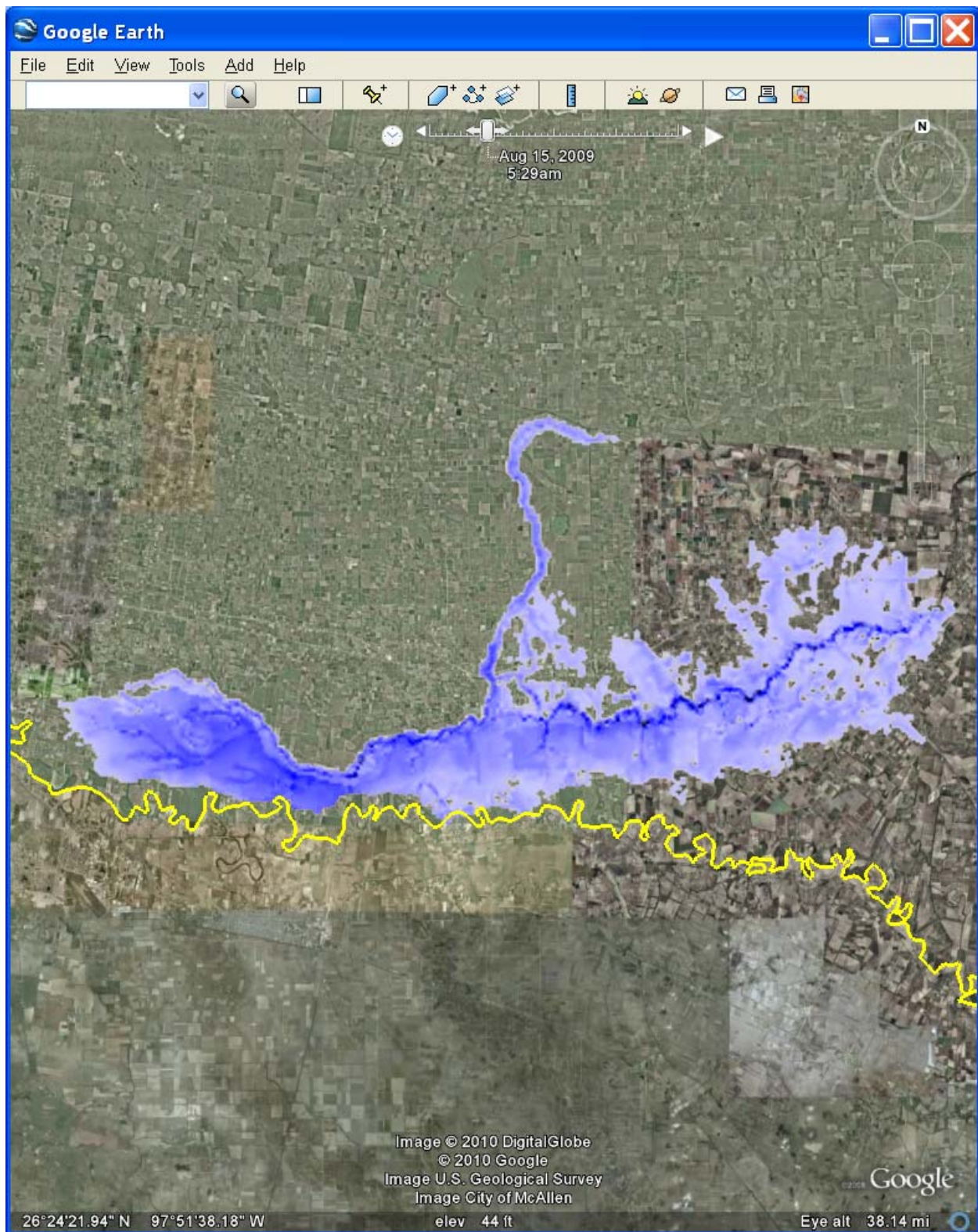


Figure 6-13: Texas KMZ flood inundation simulation results for Levee Breach: 3 as seen in Google Earth.



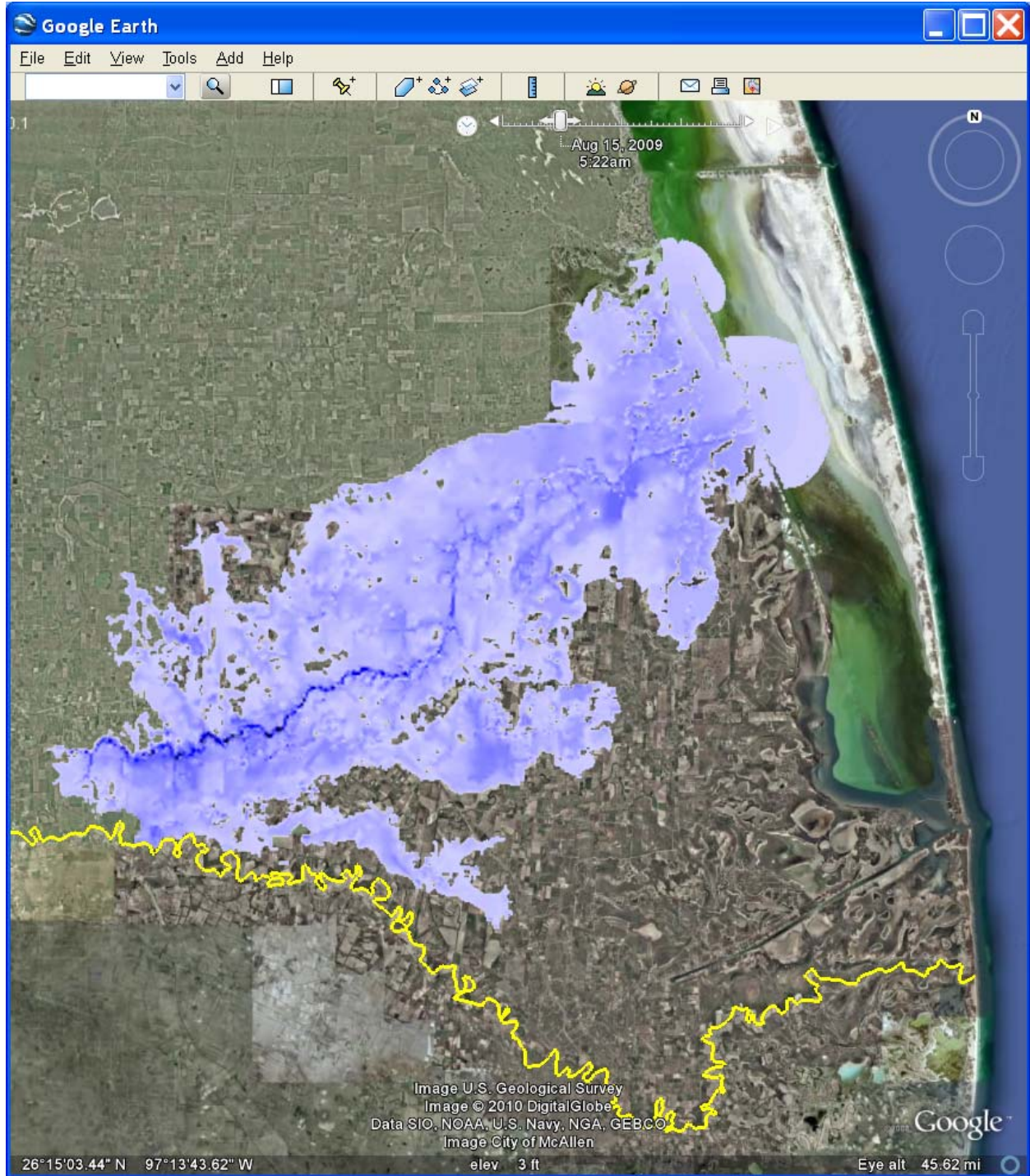


Figure 6-14: Texas KMZ flood inundation simulation results for Levee Breach: 4 as seen in Google Earth.

### 6.3 Creation of GSSHA Flood Inundation Simulations

The St. Louis and southern Texas flood inundation simulation prototypes established the necessary inputs and the potential outputs for GSSHA flood inundation simulations. Any number and combination of geoprocessing tools could be developed to create, edit, archive, and export GSSHA flood inundation simulations. However, most of the processes necessary to create these simulations currently exist in WMS and the duplication of efficient functionality would be pointless. Yet, the creation of geoprocessing tools that aid in archiving, updating boundary conditions, and exporting of GSSHA flood inundation simulations would likely be advantageous. The steps in WMS necessary to create a GSSHA flood inundation are described below.

The first step in creating a GSSHA simulation is to establish the model domain for the flood inundation simulation. The model domain must then be discretized to a grid composed of square, constant sized grid cells. Elevation and roughness values must then be assigned to each grid cell. Finally, features such as levees, railroads, roads, etc. are represented as embankment arcs along grid cell edges or elevated grid cells. The GSSHA model can then be archived for later retrieval, editing, executing, and exporting to Google Earth.

#### 6.3.1 Flood Inundation Model Domain

If the levee data is stored in a NLD, establishing the model domain is simplified because each polygon in the NLD *protected\_area* feature class represents the model domain for the simulation of a levee breach of the levees adjacent to that polygon.

If the levee data is stored in a format devoid of polygons representing protected areas, an alternative method of establishing the model domain must be employed. A DEM could be used

to create a model domain polygon by contouring the necessary elevation. The contouring could be accomplished using the levee centerline polyline and maximum possible water level elevation along the levee centerline polyline.

### **6.3.2 Model Domain Discretization**

With the model domain established, discretization of the domain is simply a matter of creating a rectangle encompassing the domain, dividing the rectangle into squares of constant size, and establishing which cells are contained within the model domain. In WMS the user simply specifies the desired grid cell size, or number of grid cells, and all of these steps are performed automatically. Figure 6-15 displays the discretization of a model domain polygon in WMS. The blue polygon represents the area protected by a levee in St. Louis. The red grid cells represent inactive cells and the black grid cells represent active cells.

### **6.3.3 Assigning Grid Elevation Values**

The next step in creating a GSSHA flood inundation simulation is to assign an elevation value to each of the active grid cells. This is typically performed in WMS by mapping the values in a DEM to the grid. Figure 6-16 shows the active grid cells of a flood inundation simulation colored according to assigned elevation values.

### **6.3.4 Assigning Grid Overland Flow Roughness Values**

In addition to elevation values, a number of parameters may be assigned to each grid cell in a GSSHA model. These parameters facilitate the calculation of overland flow, infiltration, evapo-transpiration, interception, etc. These parameters can be supplied to GSSHA as “a

specified uniform value, as a GRASS ASCII map with a different value for every cell, or as tabled values referenced to index maps” (Downer, 2008). The Geographic Resources Analysis Support System (GRASS) American Standard Code for Information Interchange (ASCII) is a simple ASCII format for describing gridded data (PRISM, 2005). Overland flow roughness parameters are typically derived from:

- A land use (land cover or vegetation cover) coverage
- A soil type coverage
- A combination of land use and soil type coverages

To assign grid roughness values in WMS, the user typically specifies the type and location of land use and/or soil type coverages and the location of a table relating land use/soil type to Manning’s roughness values. Figure 6-17 demonstrates the grid cells of a flood inundation colored according to assigned grid roughness values.

### **6.3.5 Representation of Linear Features**

Typically, the width of linear features such as levees, roads, railroad, etc. is less than the width of a single GSSHA grid cell. The coarseness of digital elevation data often results in GSSHA model grid elevations that fail to represent these linear features. To account for this, embankment arcs can be placed along cell edges to represent linear features. The NLD *levee\_centerline* polyline feature class geometry includes z-values. Users with levee data in a format without levee centerline feature class geometry containing z-values would need to create such feature classes. For all users, polyline feature classes with geometry including z-values would need to be created to represent all other important linear features in the model domain.

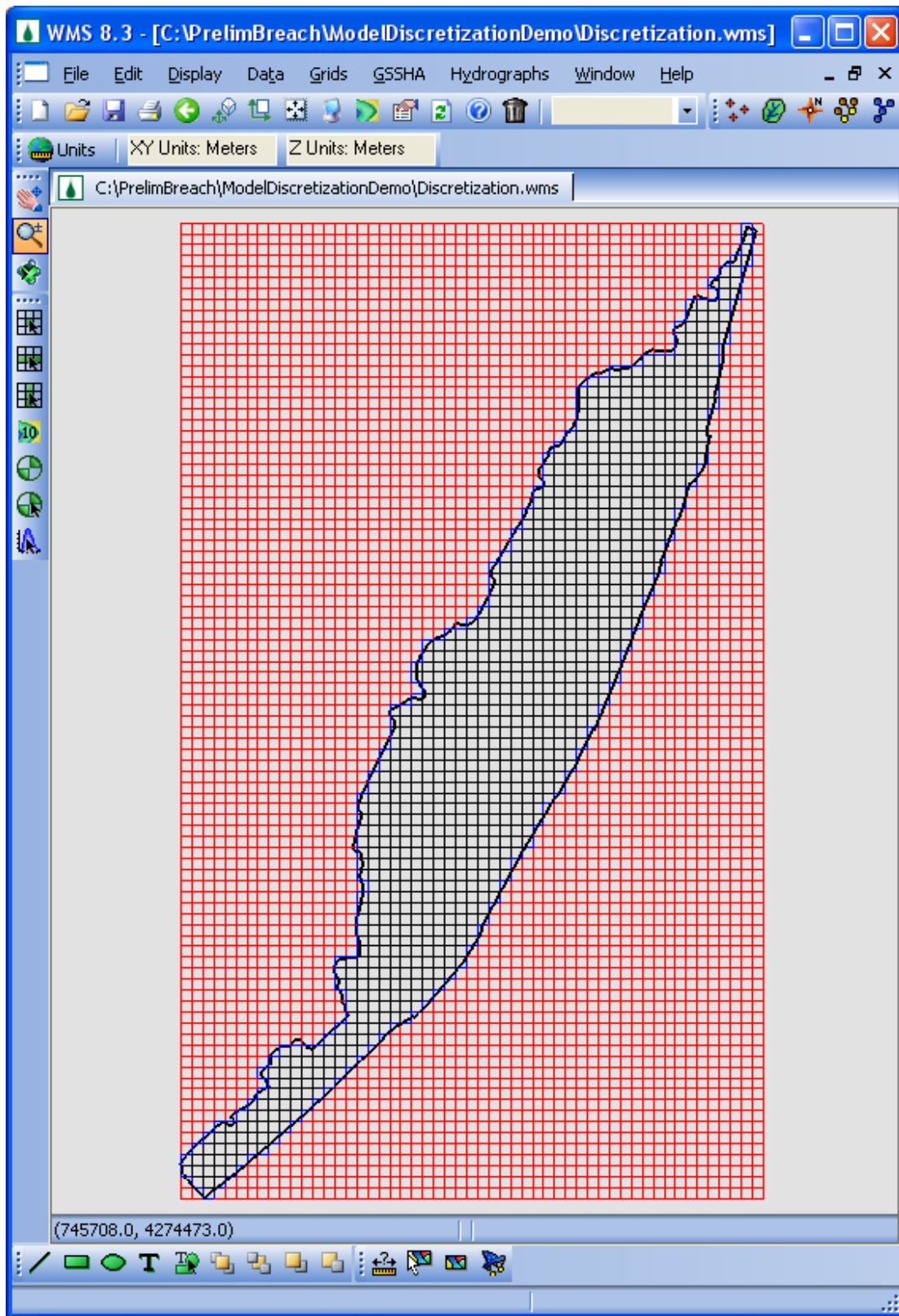


Figure 6-15: Model discretization of a NLD protected area polygon as seen in WMS.



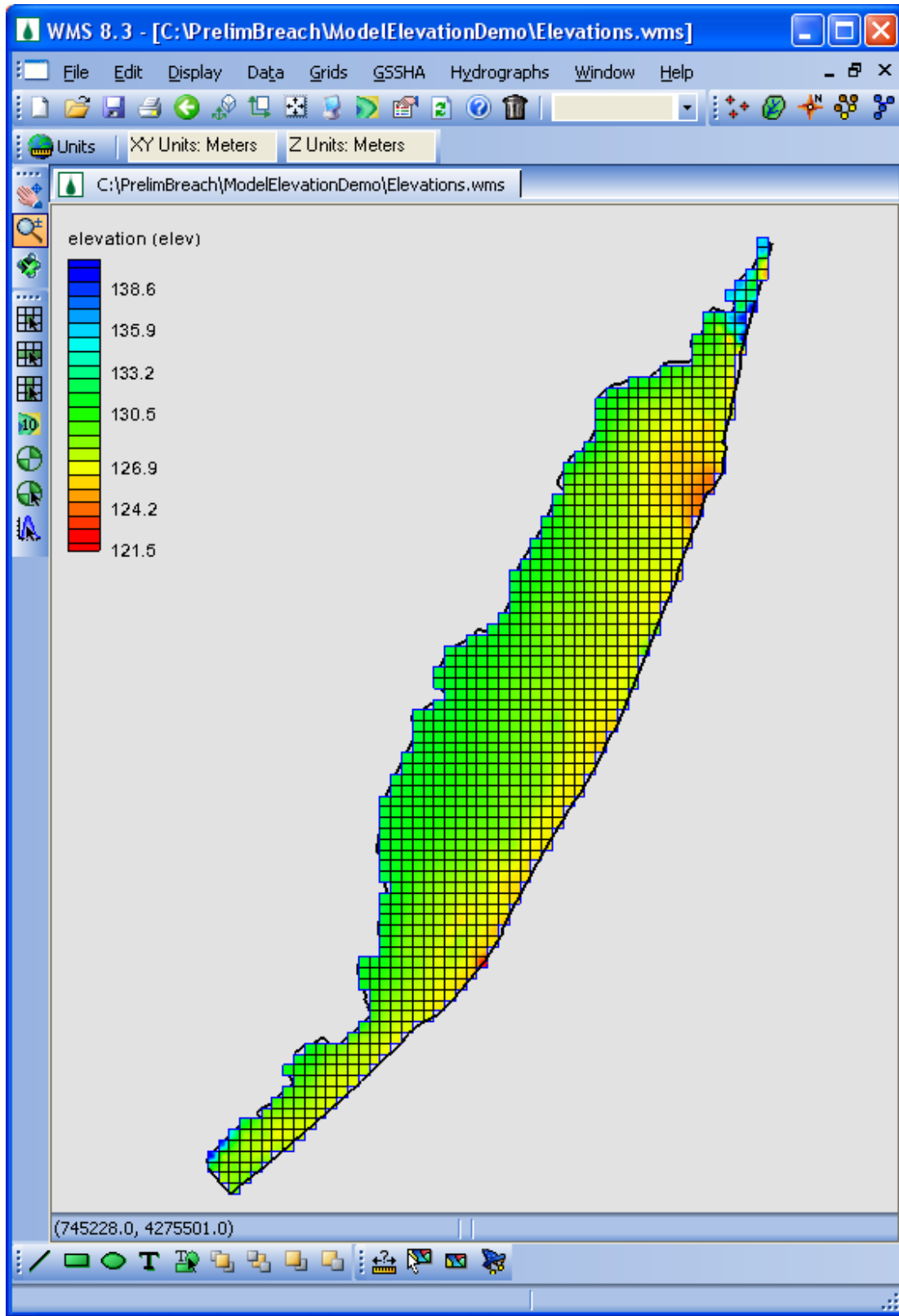


Figure 6-16: Flood inundation simulation grid elevations as seen in WMS.



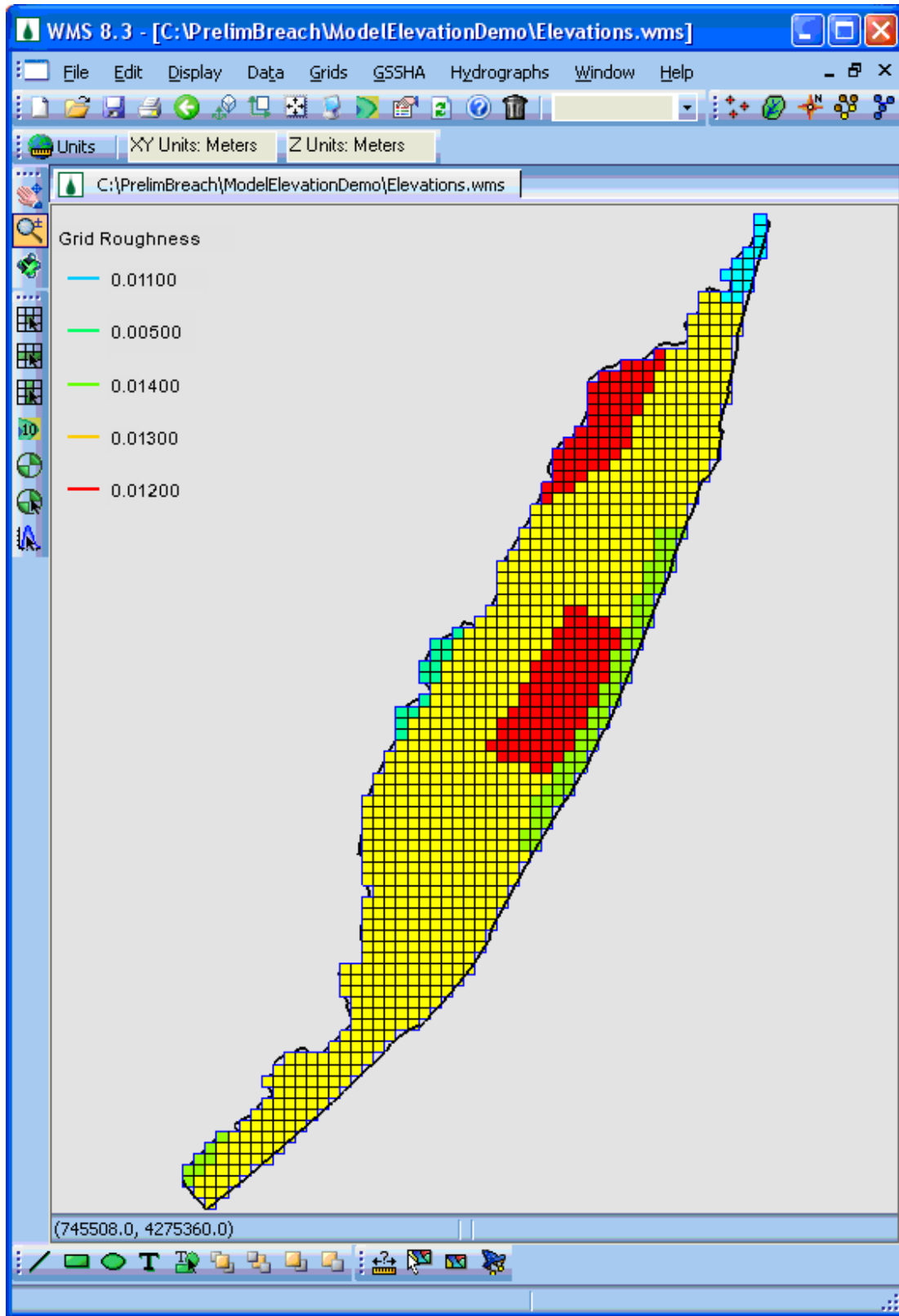


Figure 6-17: Flood inundation simulation grid roughness values as seen in WMS.

To create embankment arcs in WMS representing levees, roads, railroads, etc. the user would simply import the appropriate feature class, map the feature class to a feature object, and change the feature object's attribute type from general to embankment arc. Then, WMS would automatically assign the embankment arc along the appropriate cell edges with the proper elevations.

#### **6.4 Creation of GSSHA Flood Inundation Geoprocessing Tools**

As levee management personnel complete the creation of GSSHA flood inundation simulations for protected areas along a levee system, a method to automate simulation archiving and to establish the linkage between the simulations and the Levee Analyst database would be helpful. The creation of an *Archive GSSHA Flood Inundation Simulation* geoprocessing tool could assist Levee Analyst users in archiving simulations and establishing the necessary linkage. After inundation simulations have been archived, it would also be helpful to have a tool that assisted users in quickly locating a simulation and updating the simulation's boundary conditions based upon a levee breach scenario. The creation of an *Update GSSHA Simulation Boundary Conditions* geoprocessing tool would facilitate the updating of simulation boundary conditions. With the boundary conditions updated, the users would then find it useful to have a *Run GSSHA Simulation* geoprocessing tool that would launch GSSHA and run the simulation. Lastly, an *Export GSSHA Simulation to KMZ* geoprocessing tool would allow Levee Analyst users to view the GSSHA flood inundation simulation in Google Earth. A brief outline of each proposed geoprocessing tools is provided below.

#### 6.4.1 Creation of a Archive GSSHA Flood Inundation Simulation Geoprocessing Tool

Two apparent options exist for archiving GSSHA flood inundation simulations. First, the simulation files could be stored in a windows directory. Second, the simulation files could be compressed into a single file and stored as a BLOB in a geodatabase table.

To store the simulation files in a windows directory the user would specify the simulation's directory as an input to the *Archive GSSHA Flood Inundation Simulation* geoprocessing tool. Then the *Archive GSSHA Flood Inundation Simulation* geoprocessing tool would place the directory path, along with the ID of the associated database protected area polygon in a geodatabase table. Later, the information stored in the geodatabase table would be used by the *Update GSSHA Simulation Boundary Conditions* geoprocessing tool to locate the correct folder and open the GSSHA simulation files for editing. The advantage of this method for archiving the simulations is that writing and reading the simulation files would be faster than the alternative method. The disadvantage is that sharing the geodatabase and associated simulation files would require the recipient to maintain the correct file structure or manipulate the geodatabase table to reflect an altered file structure.

To store the simulation files as a BLOB in a geodatabase table the user would specify the simulation file location, geodatabase table used to store the simulation, and table BLOB field as input for the *Archive GSSHA Flood Inundation Simulation* geoprocessing tool. The *Archive GSSHA Flood Inundation Simulation* geoprocessing tool would then zip up the files and place the single zipped file, along with the ID of the associated protected area polygon, in the geodatabase table. Later, the simulation stored in the geodatabase table must first be extracted before it could be used by the *Update GSSHA Simulation Boundary Conditions* geoprocessing tool to open the GSSHA simulation files for editing. The advantage of this method for archiving

the simulations is that all of the data corresponding to a levee system can be contained in a single geodatabase. A central repository of all levee data would simplify sharing the data and eliminate the file structure issues that arise with externally linked data. The disadvantage of this method is the extra time it would require to zip up and archive the files as a BLOB and the time required to extract and unzip the BLOB before it could be edited, updated, or exported.

#### **6.4.2 Creation of an Update GSSHA Simulation Boundary Conditions Geoprocessing Tool**

The unique aspects of modeling a levee breach lead to a number of issues that warrant further study and investigation. The main issues arise in establishing boundary conditions that adequately simulate the complex interaction between stream flow and overland flow resulting from the breaching or overtopping of a levee. GSSHA currently has the capability to simulate this interaction in three different ways.

The first way is for the user to specify the water stage (elevation) on the grid cell(s), adjacent to the levee, due to a levee breach. This can be accomplished in GSSHA using a variable stage boundary condition. This option would require the user to use an external water modeling program to simulate the river flow, levee breach, and resulting spill onto the floodplain.

The second method available for simulating a levee breach is to place a grid cell hydrograph on the cell(s) representing the location of the levee breach. This option would also require the user to use an external water modeling program to simulate the river flow, levee breach, and resulting spill onto the floodplain.

The third option available in GSSHA for simulating the interaction between stream flow and breach flow is the use of GSSHA's overbank flow capabilities. In this scenario, the levee

should be simulated as raised grid cells as apposed to an embankment arc. The stream properties (cross section geometry, roughness, thalweg, node distribution, etc) along a stretch of the stream bordering the levee must be provided. A hydrograph must be obtained and assigned to the appropriate stream node. Finally, the grid cell(s) corresponding to the location of the levee breach must be lowered to the appropriate elevation(s). This research briefly investigated this levee breach modeling approach. A simple GSSHA model was constructed to simulate a levee breach (See). Various GSSHA model parameters were adjusted to determine the sensitivity of overbank flow to those parameters. As expected, varying stream cross section, stream roughness, and stream slope affected the amount of overbank flow. Interestingly, varying stream vertex distribution, even in stream stretches several grid cells away from the simulated levee breach, affected the amount of overland flow from the levee breach. Also notably, varying grid cell size, while maintaining a constant levee breach length and depth, affected the amount of overbank flow. The magnitude of the sensitivity of the overbank flow calculation to stream vertex distribution and grid cell size was unclear. Additional research may validate this approach and establish guidelines for selecting stream node distribution and grid cell size. However, additional research may indicate that the overbank flow calculations inadequately simulate the complex nature of levee breach flow.

The final option presented in this research for simulating the interaction between stream flow and levee breach flow is a combination of the preceding options. This option proposes that GSSHA be enhanced to allow the assignment of a lateral weir boundary condition to a GSSHA stream node. Again, the stream properties (cross section geometry, roughness, thalweg, node distribution, etc) along a stretch of the stream bordering the levee must be provided. A hydrograph must be obtained and assigned to the appropriate stream node. The lateral weir

boundary condition would then allow GSSHA to calculate the spill onto the floodplain based upon a specified levee breach length, discharge coefficient, and low point elevation.

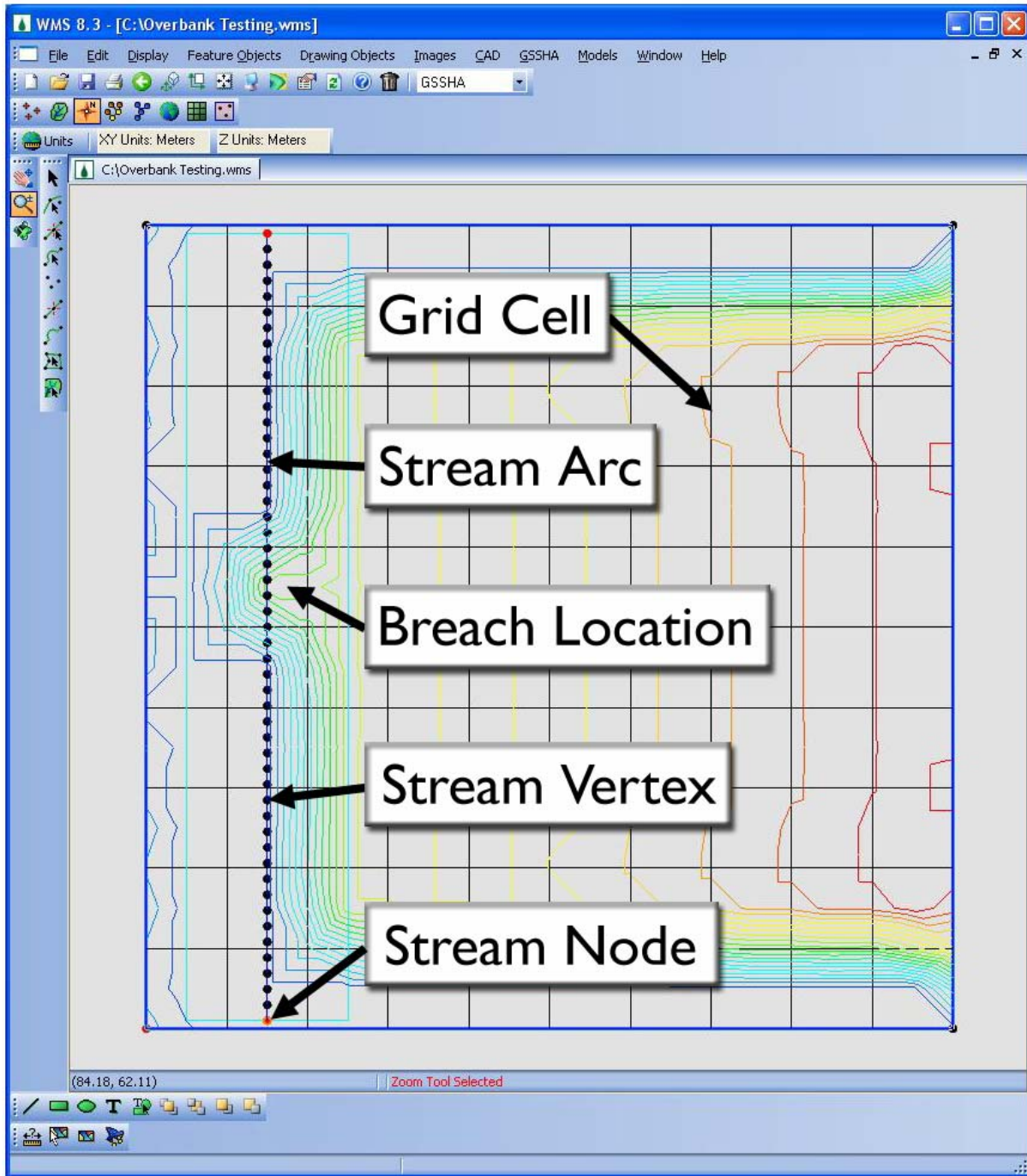


Figure 6-18: GSSHA overbank flow testing in WMS.

As input to the *Update GSSHA Simulation Boundary Conditions* geoprocessing tool the user could specify the distance along the levee at which the levee breach is to be simulated, the length of the levee breach, and the low point elevation of the breach. Using the dynamic segmentation linear referencing capabilities of ArcGIS the geoprocessing tool could then create a levee breach polyline from the levee centerline feature class (ESRI, 2009). This polyline, the associated protected area polygon ID, and a simulation duration could be combined with whatever other input is necessary (based upon the option selected to simulate stream flow and breach flow interaction) to extract the appropriate GSSHA model, update the appropriate boundary conditions, and save the model. The *Update GSSHA Simulation Boundary Conditions* geoprocessing tool could then place the polyline representing the levee breach, a simulation summary, and a simulation ID in a proposed *Levee Analyst Flood Inundation Simulation Summary* feature class designed to store levee breach simulation locations and metadata.

### **6.4.3 Creation of a Run GSSHA Simulation Geoprocessing Tool**

The creation of a GSSHA simulation results in the creation of a number of simulation files. Among these, the “.prj” file stores the name and location of all of the necessary simulation files. The input for a *Run GSSHA Simulation* geoprocessing tool could simply be the location of a GSSHA simulation “.prj” file or a simulation ID from the proposed *Flood Inundation Simulation Summary* feature class. In the latter case, the tool would obtain the location of the GSSHA simulation “.prj” file from the feature class attribute table. Then the tool would launch the GSSHA executable (perhaps the location of the GSSHA executable could be stored in the *Executables* table) and pass the “.prj” file location to the executable. GSSHA would then run the

simulation and place a “.dep” file in the folder containing the “.prj” file. The “.dep” file contains the depth of water on each grid cell, for each time-step of the simulation.

#### **6.4.4 The Creation of an Export GSSHA Simulation to KMZ Geoprocessing Tool**

An input required for the *Export GSSHA Simulation to KMZ* geoprocessing tool could be the levee breach simulation ID of a simulation stored in the *Flood Inundation Simulation Summary* feature class. The tool could then locate the “.dep” file using the simulation location stored in the *Flood Inundation Simulation Summary* feature class attribute table. Located in the same folder would be the simulation “.pro” file. This file describes the projection of the GSSHA simulation which would be necessary for the geoprocessing tool to create a KMZ file. Additional input for the tool could be depth contouring options. The geoprocessing tool would then take the “.dep” and “.pro” files, along with the contouring specifications, and use WMS functionality to create a KMZ file.

The geoprocessing tool could determine depth statistics such as maximum depth, average time of inundation, average time of recession, etc. These statistics, along with a hyperlink to the KMZ file, could then be placed in the *Flood Inundation Simulation Summary* feature class attribute table using the simulation ID specified as tool input.

Upon creation of the KMZ file, the *Export GSSHA Simulation to KMZ* geoprocessing tool could then execute Google Earth (the executable path could be stored in the *Executables* table) and open the KMZ file for viewing.



### 6.4.5 Viewing GSSHA Archived Flood Inundations in Google Earth

One of the utilities of populating a *Flood Inundation Simulation Summary* feature class is that such a feature class would greatly enhance the ease of viewing archived GSSHA flood inundation KMZ files in Google Earth. The KML of an exported, prototyped *Flood Inundation Simulation Summary* feature class is shown in Figure 6-19

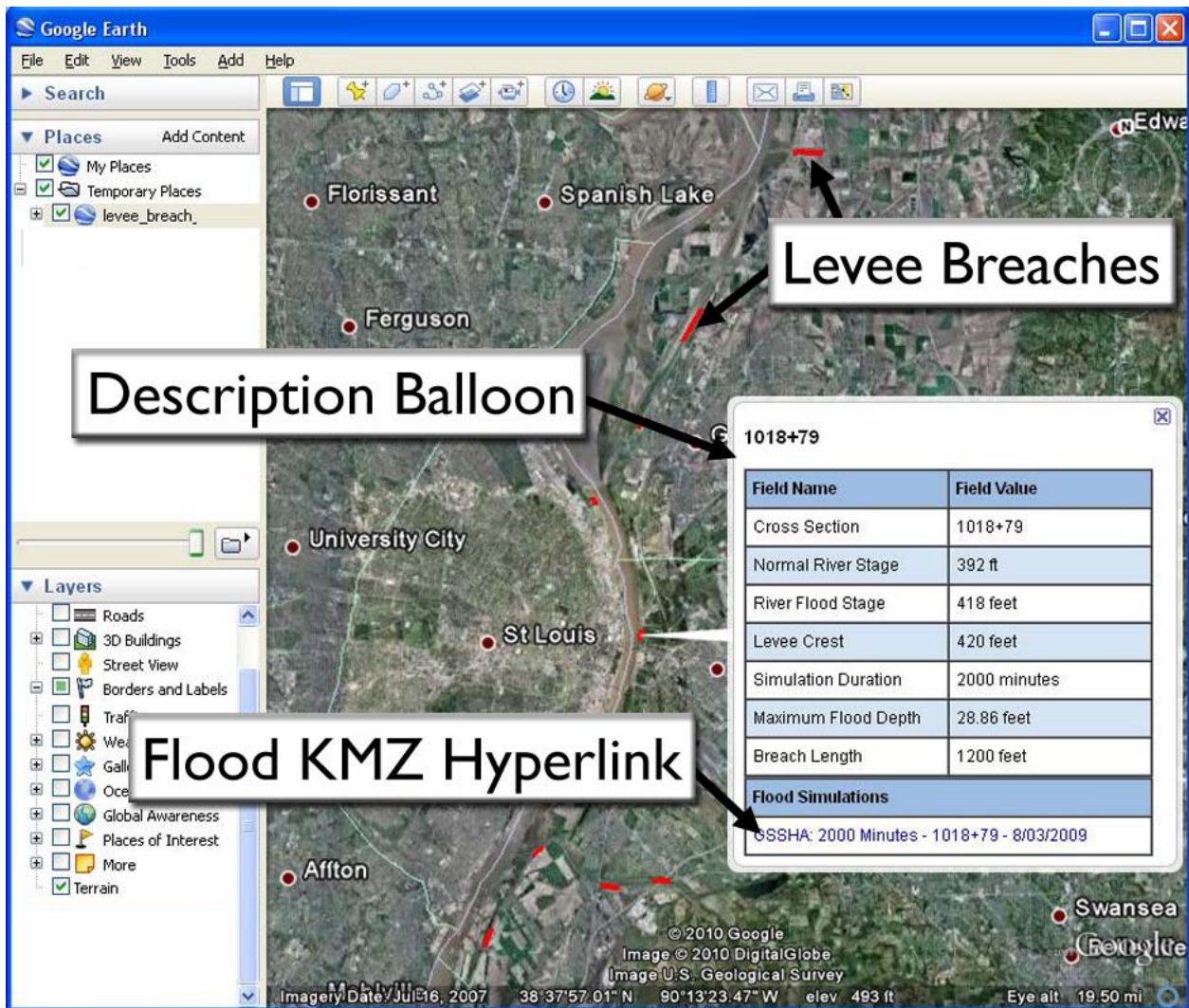


Figure 6-19: *Flood Inundation Simulation Summary* KML file as seen in Google Earth.

To view GSSHA archived flood inundations in Google Earth, a user must first export the *Flood Inundation Simulation Summary* feature class to a KML file using the ArcGIS *Layer To KML* geoprocessing tool. Upon opening the resulting KML file in Google Earth, the user could view a polyline showing the physical location of each archived levee breach scenario. The user could click on each polyline to display a description balloon. The description balloon would contain all of the summary information stored in the *Flood Inundation Simulation Summary* feature class attribute table for that simulation – including a hyperlink to the simulation KMZ file. The user could click this hyperlink and the KMZ file would be loaded into Google Earth and begin playback of the levee breach scenario.



## 7 Conclusion

The original Levee Analysis Data Model was designed to provide a central location, compatible with the NLD, for storing large amounts of levee seepage and slope stability analytical data. The original Levee Analyst geoprocessing tools were created to assist users in populating, managing, and analyzing Levee Analyst geodatabase data. This research sought to enhance the original data model and tools by adding additional options for analyzing and visualizing levee simulations.

### 7.1 Research Accomplishments

This research expanded the universality of the LADM and geoprocessing tools by introducing the capability of importing and exporting popular GeoStudio 2004 and 2007 SEEP/W seepage and SLOPE/W slope stability simulations. This was accomplished by updating the database schema and creation of the *Import GeoStudio Simulation* geoprocessing tool.

Further, this research made the data stored in a Levee Analyst database more accessible to individuals with little or no ArcGIS experience by developing the *Export Raster From Attribute Field* and *Export Cross Section To KML* geoprocessing tools that export the database to Google Earth.

An additional accomplishment of this research was the demonstration of the compatibility of the National Levee Database and the Levee Analysis Data Model. This was

accomplished using the National Levee Database populated with levee data in the St. Louis area -supplied by the USACE.

The final accomplishment of this research was prototyping the creation of GSSHA flood inundation simulations using levee database information and proposing levee analyst flood inundation geoprocessing tools. These simulations also established the need for additional Levee Analyst tables and feature classes for storing simulation data and results.

## 7.2 Future Developments and Research

Through the course of this research a number of potential future developments were identified and are described below.

The Levee Analyst toolbox could be further expanded to include geoprocessing tools that import additional seepage and limit equilibrium slope stability analytical models. An *Import Levee Analysis Data Model* geoprocessing tool could be developed to automatically create the additional fields and tables in a NLD that are necessary to store levee seepage and slope stability data and execute Levee Analyst geoprocessing tools. Future development may create a *Sync Levee Analyst and National Levee Databases* geoprocessing tool that would facilitate the continual expansion of levee data stored in the NLD by merging an expanded NLD with an older NLD containing seepage and slope analyses. Finally, the Levee Analyst toolbox could be expanded to include an *Export Levee Analyst Database to KML* geoprocessing tool that would export a Levee Analyst to Google Earth without the need to run several geoprocessing tools.

Additional research could be performed to determine the most efficient and accurate method for simulating levee breach boundary conditions in GSSHA. Research may validate the use of overbank flow in conjunction with raised grid cells representing levee embankments.

Guidelines for selecting stream vertex distribution and grid cell size may be established. However, additional research may indicate that the overbank flow calculations inadequately simulate the complex nature of levee breach flow and warrant the creation of a GSSHA lateral stream weir boundary condition. A lateral stream weir boundary condition could offer greater refinement in the coupling of 1D and 2D models in GSSHA.

Finally, the *Archive GSSHA Flood Inundation Simulation*, *Update GSSHA Simulation Boundary Conditions*, *Run GSSHA Simulations*, and *Export GSSHA Simulation to KMZ* geoprocessing tools could be further researched and developed. To accommodate these geoprocessing tools the Levee Analysis Data Model could be expanded to include tables and feature classes that store flood inundation simulations and results.



## References

- Altinakar, M. S., McGrath, M. Z., Ozeren, Y., and Omari, H. "Modeling and Risk Analysis for Floods due to Failure of Water Control Infrastructures." International Symposium on Uncertainties in Hydraulic Modeling. Montréal, Canada. 15-17 October 2008.
- Altinakar, M. S., Matheu, E. E., and McGrath, M. Z. "New Generation Modeling and Decision Support Tools for Studying Impacts of Dam Failures." Association of Dam Safety Officials Annual Conference. Hollywood, FL. 27 September – 1 October 2009.
- American Society of Civil Engineers (ASCE). Report Card for America's Infrastructure. 2009. 16 February, 2010. < <http://www.infrastructurereportcard.org/fact-sheet/levees>>
- Downer, C. W. GSSHA Primer. 2008. 17 February 2010. < [http://www.gsshawiki.com/gssha/GSSHA\\_Primer#References](http://www.gsshawiki.com/gssha/GSSHA_Primer#References)>
- Downer, C. W. Overview: Overview. 2010. 17 February 2010. <[http://www.gsshawiki.com/gssha/GSSHA\\_Primer#References](http://www.gsshawiki.com/gssha/GSSHA_Primer#References)>
- Environmental Systems Research Institute (ESRI). An Overview of Linear Referencing. 2009. 22 January 2010. <[http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=An\\_overview\\_of\\_linear\\_referencing](http://webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=An_overview_of_linear_referencing)>
- Environmental Systems Research Institute (ESRI). Map to KML (3D Analyst). 2007. 22 January 2010. <[http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Map%20to%20KML%20\(3D%20Analyst\)](http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Map%20to%20KML%20(3D%20Analyst))>
- Environmental Systems Research Institute (ESRI). Raster Datasets as Attributes. 2008. 22 January 2010. < [http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Raster\\_datasets\\_as\\_attributes](http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Raster_datasets_as_attributes)>
- Groundwater Modeling System (GMS). Computer Software. Aquaveo, LLC. Provo, Utah. 2009.
- Google KML Development Group. KML Tutorial. 2010. 25 January 2010. <[http://code.google.com/apis/kml/documentation/kml\\_tut.html](http://code.google.com/apis/kml/documentation/kml_tut.html)>
- Handy, J. "A GIS-Based Strategy for Managing Levee Simulation Data." MA thesis. Brigham Young University. 2008



Jones, N. L., Handy, J. R., and Wallace, R. M. "Levee Analyst: A GIS-Based Levee Modeling and Management System." Association of Dam Safety Officials Annual Conference, Indian Wells, CA. 7-11 September 2008.

National Committee on Levee Safety (NCLS). Recommendations for a National Levee Safety Program. 2009. 16 February 2010. < [http://www.iwr.usace.army.mil/ncls/docs/NCLS-Recommendation-Report\\_012009\\_DRAFT.pdf](http://www.iwr.usace.army.mil/ncls/docs/NCLS-Recommendation-Report_012009_DRAFT.pdf)>

Pangburn, T. "National Levee Database Infrastructure Conference." National Levee Database Infrastructure Conference. Cleveland, OH, 23 July 2009.

Parameter-elevation Regressions on Independent Slopes Model (PRISM) Climate Group. GRASS: ASCII Grid File Format Specification. 2005. 17 February 2010. < [www.prism.oregonstate.edu/pub/prism/docs/manual/Grass\\_Format.doc](http://www.prism.oregonstate.edu/pub/prism/docs/manual/Grass_Format.doc)>

Seepage Modeling with SEEP/W 2007 Version: An Engineering Methodology. Canada, Alberta: GEO-SLOPE International, Ltd., 2008.

Stability Modeling with SLOPE/W 2007 Version: An Engineering Methodology. Canada, Alberta: GEO-SLOPE International, Ltd., 2008.

United States. Federal Emergency Management Agency (FEMA). Interagency Levee Policy Review Committee. The National Levee Challenge: Levees and the FEMA Flood Map Modernization Initiative. Washington, D.C.: FEMA, 2006.

Water Resources Development Act of 2007. Pub. L. 110-114. 9 November 2007. Stat. 1041.

Watershed Modeling System (WMS). Computer Software. Aquaveo, LLC. Provo, Utah. 2009.

# Appendix A Updated Levee Analysis Data Model

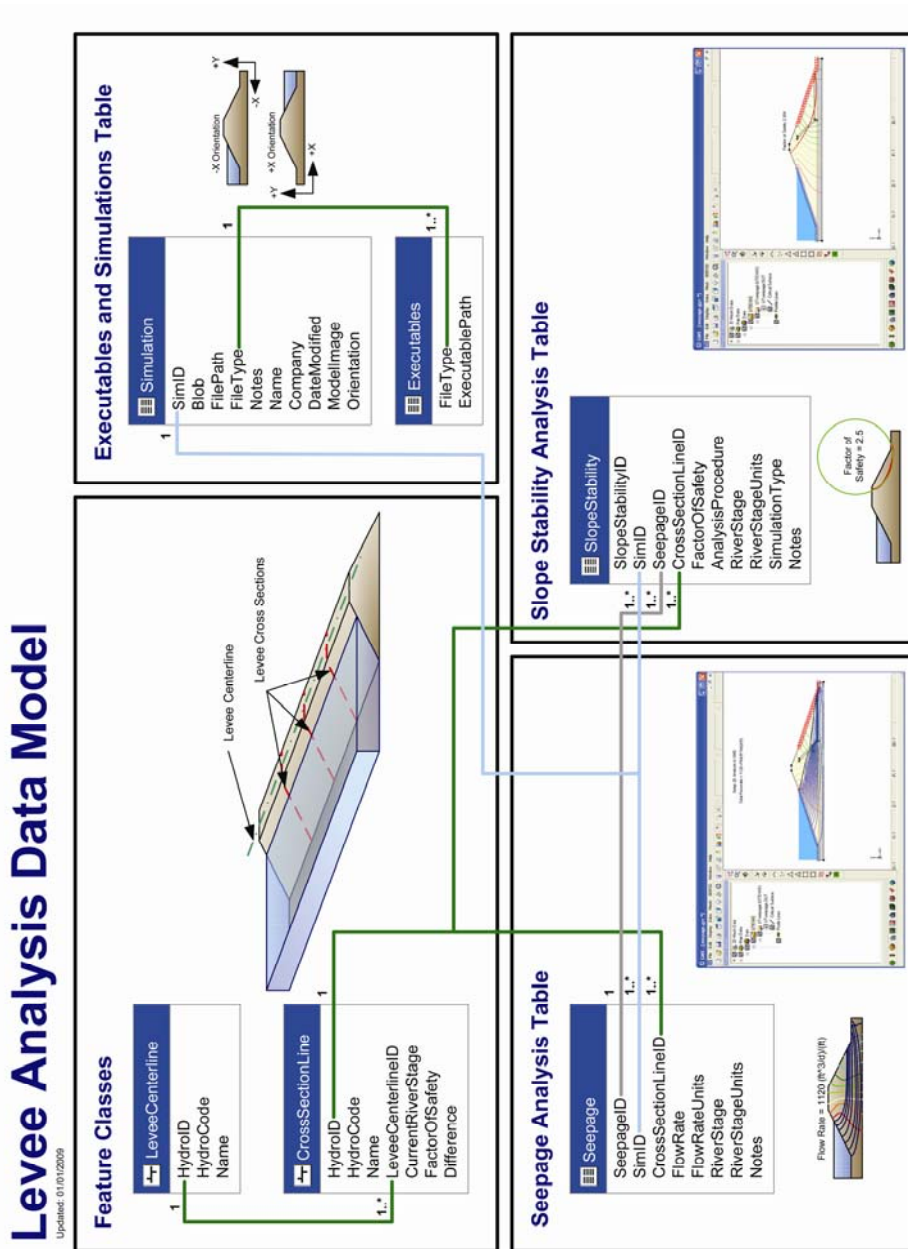


Figure A-1: Updated Levee Analysis Data Model summary.

# Levee Analysis Data Model

Updated: December 1, 2009

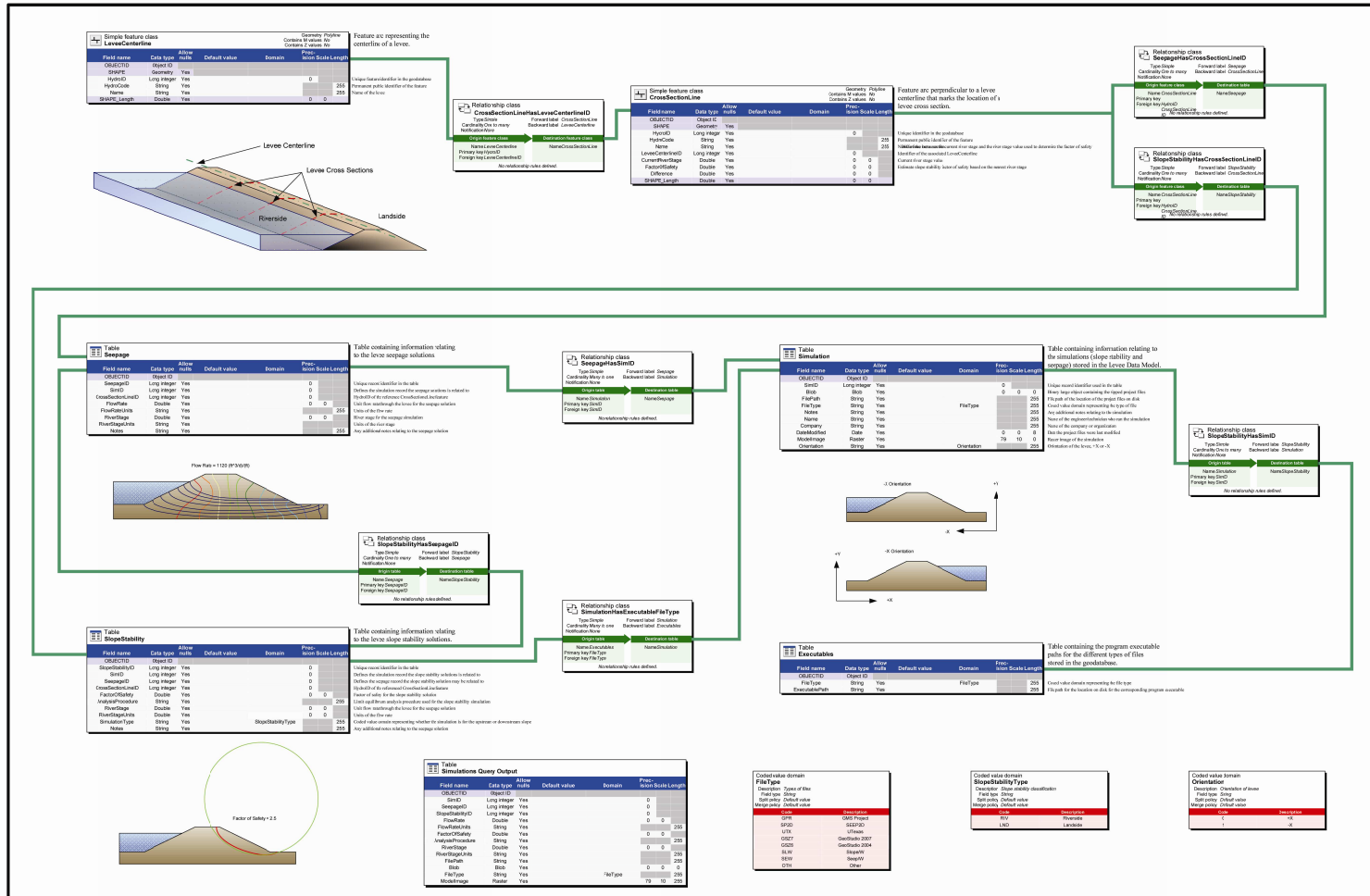


Figure A-2: Updated Levee Analysis Data Model.