Master Well Database

Introduction

The goal of this outline is to discuss the creation and implementation of a database containing all of the given information on oil and gas wells within the area of interest. The ending database will contain a tool that is able to import data from various sources, determine the coordinate system being used, use aerial images to determine location accuracy and save all information to a geodatabase. External links are provided as examples and tutorials for each of the following topics. Oftentimes well location can be difficult to decide on as various datasets may not match. The tools and software discussed will be able to take data from multiple sources and to provide a more clear location of oil and gas wells.

Software Development

All of the code provided will be created using Visual Studio Code and stored in a Github repository. The data, text files, code and work notes can be shared and saved in the repository for ease of access, updates and editing. Examples of creating a Github repository and its benefits can be accessed below.

Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab%201

Loading Datasets

With the use of python coding you can open, read, and edit dataset files from various sources. These datasets can be assigned specific variables to make editing and adding information easier. Examples of importing datasets and manipulating data are provided below.

Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab3

Location Accuracy

Accuracy of well locations can be increased by comparing the location coordinates from each dataset. Examples of comparing data and lists are given in the link below.
Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab2

Geodatabases

Creating and saving all information to a geodatabase will increase the ability to save, share and edit documents. The code in the example shows that you can specify the name, location and output of the geodatabase.
Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab4

Coordinate Systems

The ability to determine if the data is saved in the same coordinate system by using the input coordinate system to specify a preferred coordinate system, and the project management settings to change and view the coordinate systems. Finally creating a buffer around each well location to see if data is projected in the correct system.
Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab5

Choropleth Maps

The code provided shows how to create and customize a color map to help decide on the correct well location. With this choropleth map the darker color is used to show areas with higher points of agreement on well location.
Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab6

Aerial Imagery

After obtaining images for the research area the code provided can apply the images to a map of the area with current well location information. Examples of data types, where to download the images and then apply them to your map are given below.
Examples: https://github.com/tessahrichards/GEOG676/blob/main/Lab8/Lab7

Discussion Points

Return on Investment

With the use of open source images and data when available can decrease project costs. Visual Studio Code is free to use and works as a text editor with the ability to write the needed python code and import the necessary packages. The increased accuracy of well locations will help reduce unnecessary costs.

User Interface

The software can be run on various machines and data can be uploaded into commonly used programs including all ESRI products. The interface will also be easy to use and remain consistent across all departments and platforms.

Gather Requirements

Most major data types may be used in the various applications discussed and all data should have proper documentation and licensing if applicable. Again the use of open source data and images will aid in the speed and cost of the project production.

Software and Code

The software used will include ESRI platforms and VSCode as described above. There will also be extensive use of Python packages including NumPy, PySAL, and ArcPy.

- NumPy-package for scientific computing in Python, performs complex mathematical operations.
- PySAL- a Python package that contains several sub-packages, each one geared towards a specific area of spatial analysis and statistics.
- ArcPy is a Python site package that provides a useful and productive way to perform geographic data analysis, data conversion, data management, and map automation with Python.

Testing, Status Updates, and Reporting

All aspects of the software updates and programming will be performed by a specialized GIS team. Frequent updates and reports will be provided as needed and the team will be open for discussions, problems, or updates. GIS management and maintenance will be included with purchase of software.