**Lab Report**

Title: APIs: Conceptual Models and Application

Notice: Dr. Bryan Runck

Author: Jake Ford

Date: 09-03-2022

**Project Repository:**https://github.com/ThisFord/GIS5571-arc1.git

**Google Drive Link:** *<if applicable with data, notebooks, etc.>*

Time Spent: 6 so far

**Abstract**

*<Delete this text in light grey throughout>*

*250 words max. Clearly summarize the following major sections. Each gets one or two sentences.*

**Problem Statement**

Application Programming Interfaces are useful tools for programmers, the interfaces make it possible to automate tasks and queries programmatically, they allow all the actions of a website interface and more. Understanding how APIs work and developing experience in their use is essential for many GIS tasks. (*The CKAN API — CKAN Documentation 2.1.5 Documentation*, n.d.; *What Is an Application Programming Interface (API)?*, 2022) This lab strengthens understanding by comparing and contrasting three APIs, using informal conceptual models and custom built Extract Transform and Load (ETL) routines derived from user interface; and deepens experience by building an ETL pipeline with Open Source Tools in Jupyter Notebooks in Arc Pro and Arc Online platforms, fetching data and and performing a spatial join.

*Diagram

Description automatically generated.*

*Table 1. API comparison and use requirements*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Requirement** | **Defined As** | **(Spatial) Data** | **Attribute Data** | **Dataset** | **Preparation** |
| 1 | Internet connection | Ability to access information on the web |  |  |  |  |
| 2 | Target databases | the information you’re trying to access programatically |  |  |  |  |
| 3 | Jupyter Notebook in ArcPro | Python programming interface in esri’s arcpro software |  |  |  |  |
| 4 | Jupyter Notebook in Arc Online | Python programming interface in esri’s online version of ArcGIS |  |  |  |  |
| 5 | Visual model builder | A graphic representation of your informal models |  |  |  |  |

**Input Data**

The input data is largely user generated, as experience and usability is key to the compare and contrast method of analysis. The target data we’re trying to access is relatively arbitrary, as the experience is derived from mapping the APIs and using the ETL pipeline. In the case of this study three APIs are used, the Minnesota Geospatial Commons, Google Places, and NDAWN. The data gathered differs for each API, the data listed in table 2 are the datasets used in the fetch and join operations.

Table 2. data collected from the APIs

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Title** | **Purpose in Analysis** | **Link to Source** |
| 1 | TBD | Demonstrate ETL capability |  |
| 2 |  | Demonstrate ETL capability |  |
| 3 |  |  |  |
|  |  |  |  |

**Methods**

1. Build Informal Conceptual Models
2. Test model comprehension with custom ETL
3. Build ETL pipelines with Jupyter Notebooks
   1. downloads two data sets
   2. transform both datasets to the same [coordinate reference system](https://pro.arcgis.com/en/pro-app/latest/help/mapping/properties/coordinate-systems-and-projections.htm) (geographic and projected)
   3. spatial join them
   4. print to screen the head of the table showing the merged attributes
   5. save the integrated dataset to a geodatabase.

*Include a data flow diagram or screenshot from model builder. Do references in line (Rammankutty, 2033). Document any and all steps that you did to the input data in the data flow diagram. Provide natural language description of the most important steps, giving a narrative arc and provide well formatting screenshots with a boarder and centered throughout.*

*Resources on Data Flow Diagrams:*

* [*https://www.visual-paradigm.com/tutorials/data-flow-diagram-dfd.jsp*](https://www.visual-paradigm.com/tutorials/data-flow-diagram-dfd.jsp)
* [*https://www.lucidchart.com/pages/data-flow-diagram/how-to-make-a-dfd*](https://www.lucidchart.com/pages/data-flow-diagram/how-to-make-a-dfd)

*Figure 1. Data flow diagram.*

*If appropriate, add in pseudo-code describing model algorithms and/or objects. If using mathematical equations, create a clear mapping between the reference equation, pseudo-code, and actual implementation in a programming language.*

**Results**

*Show the results in figures and maps. Describe how they address the problem statement.*

*Diagram

Description automatically generated*

*Diagram

Description automatically generated*

*Follow best practice for map design, coloring, etc.*

**Results Verification**

The results from the compare and contrast are qualitative and somewhat subjective, accuracy of the assessments is tested against a survey of the experiences of the class and is reviewed by expert instructors. Results for the ETL implementation are a clear succeed/fail outcome, as each ETL must perform each step properly to output an integrated dataset to the geodatabase. Successful implementation of the ETL results in a working output to the geodatabase.

**Discussion and Conclusion**

*What did you learn? How does it relate to the main problem?*

**References**

*Use a common format*

**Self-score**

*Fill out this rubric for yourself and include it in your lab report. The same rubric will be used to generate a grade in proportion to the points assigned in the syllabus to the assignment.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Description** | **Points Possible** | **Score** |
| **Structural Elements** | All elements of a lab report are included **(2 points each)**:  Title, Notice: Dr. Bryan Runck, Author, Project Repository, Date, Abstract, Problem Statement, Input Data w/ tables, Methods w/ Data, Flow Diagrams, Results, Results Verification, Discussion and Conclusion, References in common format, Self-score | 28 | **28** |
| **Clarity of Content** | Each element above is executed at a professional level so that someone can understand the goal, data, methods, results, and their validity and implications in a 5 minute reading at a cursory-level, and in a 30 minute meeting at a deep level **(12 points)**. There is a clear connection from data to results to discussion and conclusion **(12 points)**. | 24 | **24** |
| **Reproducibility** | Results are completely reproducible by someone with basic GIS training. There is no ambiguity in data flow or rationale for data operations. Every step is documented and justified. | 28 | **28** |
| **Verification** | Results are correct in that they have been verified in comparison to some standard. The standard is clearly stated **(10 points)**, the method of comparison is clearly stated **(5 points)**, and the result of verification is clearly stated **(5 points)**. | 20 | **20** |
|  |  | 100 | **100** |

**Diagram

Description automatically generated**

**https://www.geeksforgeeks.org/get-post-requests-using-python/**