

# Data Conversion and Management for The National Map Center of Excellence for Geospatial Information Science United States Geological Survey

## Introduction

This is a simple procedure that converts data for the National map which is stored on the Amazon S3 server. The original form of these datasets is in GDB (Geographic Data Base format) and the goal is to convert it to XML / JSON. The procedure consists of three basic steps:

- Step 1: Retrieving data from Amazon S3
- Step 2: Converting the retrieved GDB files to XML
- Step 3: Converting XML files to required format (mostly JSON)

The detailed description of each step of the conversion procedure is as follows:

### Step 1: Retrieving the data from Amazon S3 (Simple Storage Service - Cloud Storage)

- To retrieve the data files from the amazon server, go to the National map website nationalmap.gov

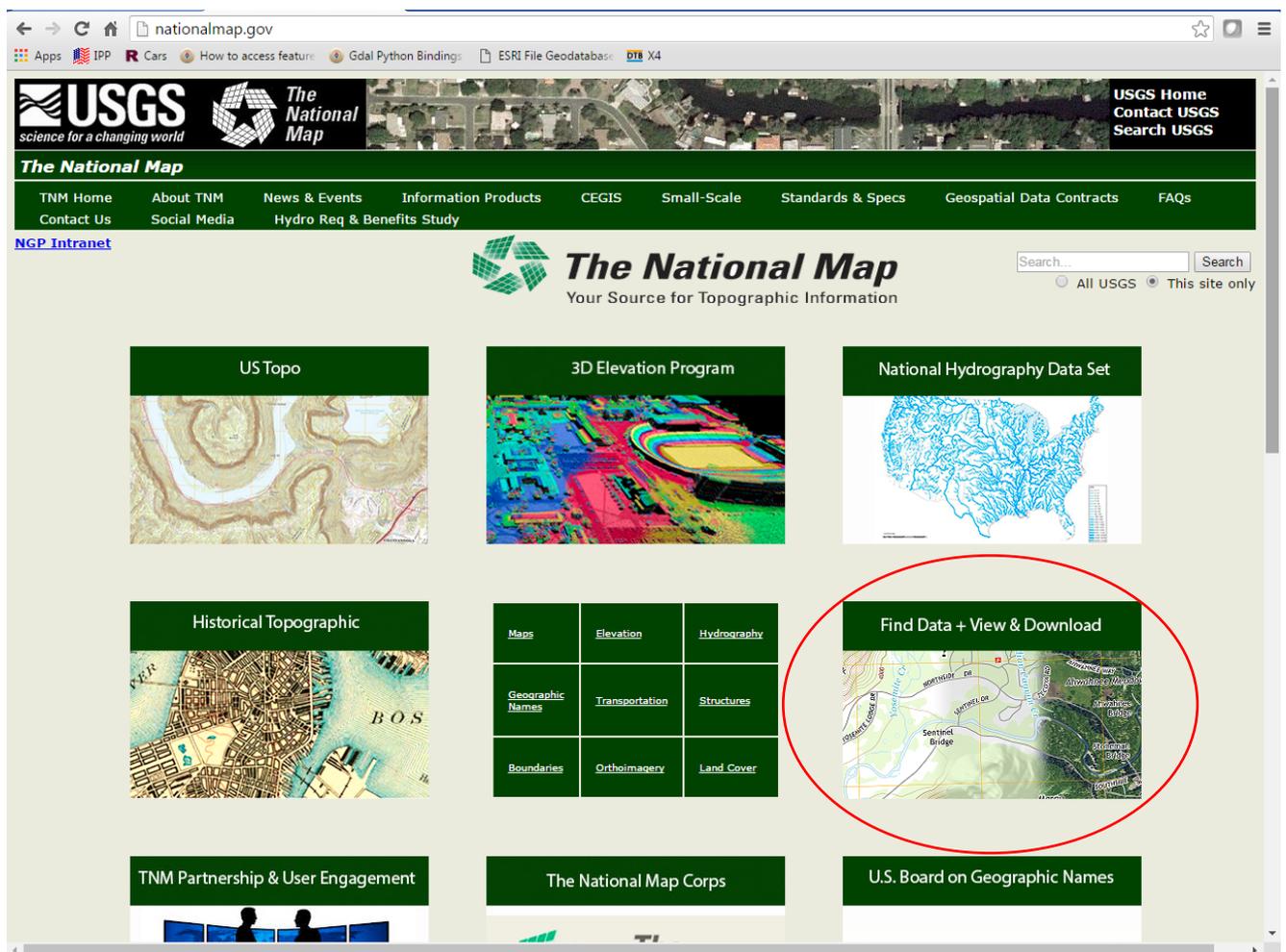


Figure 1 National Map website

- Download the Hydrography, Transportation or Boundaries data from these available data sets. The download viewer is shown in Figure 2.
- Select the option that you want to download datasets for
  - Select the GDB file format, as we want to download the GDB files.

- After selecting appropriate choices, select on Find Products, and the results will display the products of all the states together.
- At the top of the results, there will be option to add all the products of that page all-together to the cart for downloading. Add products from all the available pages to the cart.
- Figure 3 displays the above procedure as shown by the National Map Viewer.

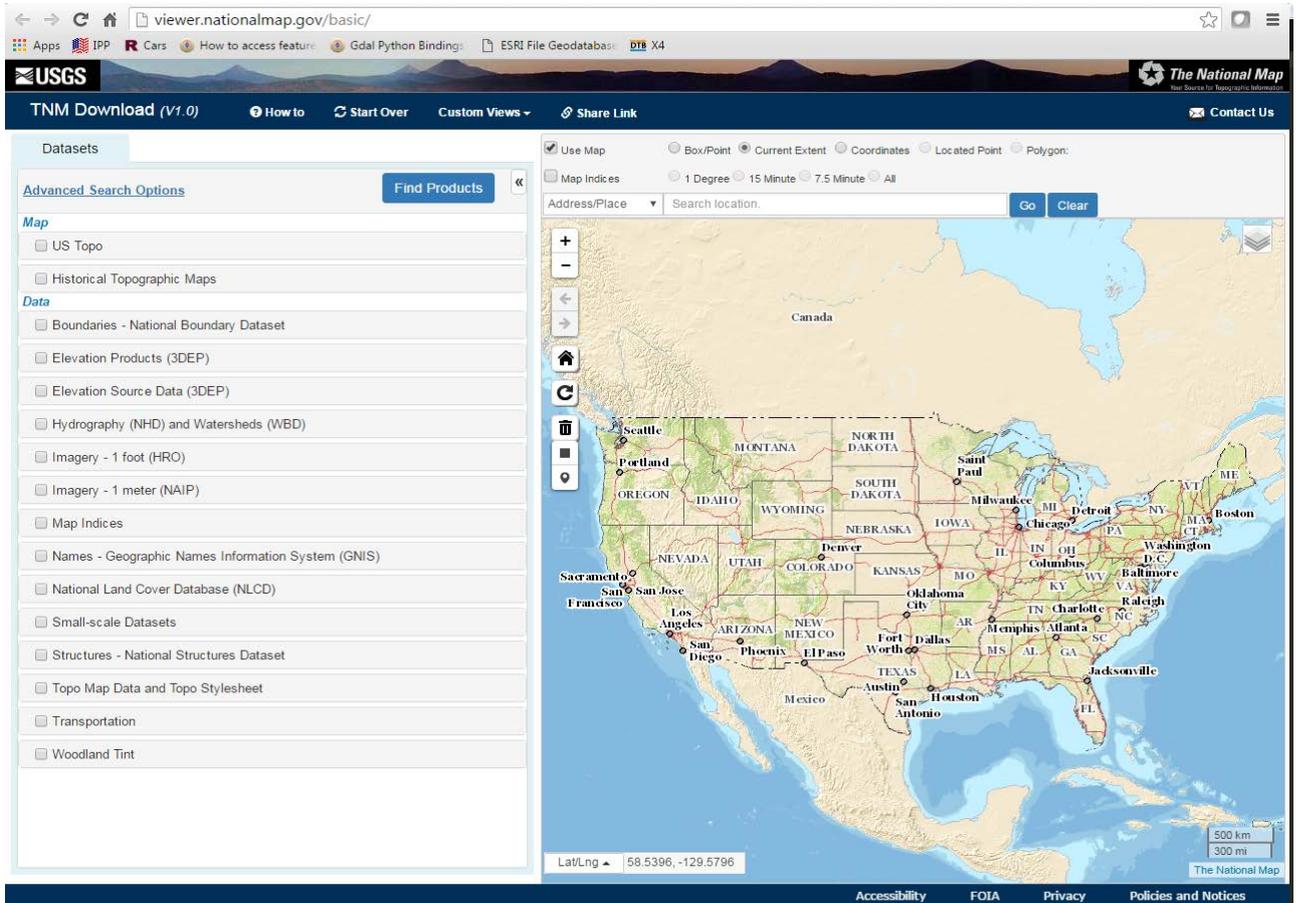


Figure 2 Download viewer

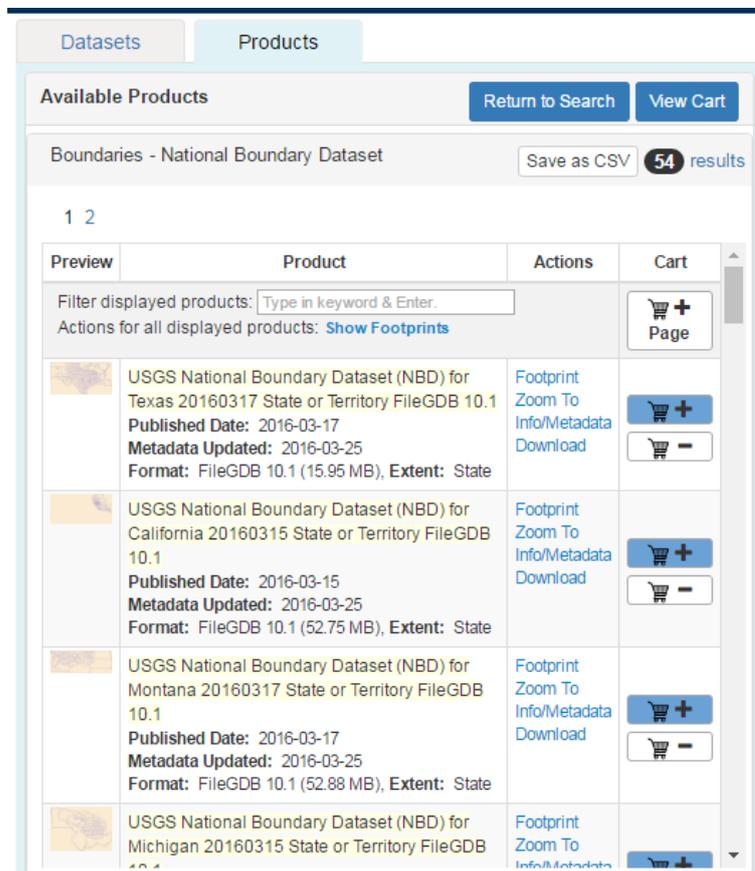


Figure 3 Adding Data sets to the Cart

- After selecting the required data, you need to go “View Cart” to see all the products added to your cart. From the cart there is an option to export the items in the cart. This will be a csv file listing your items.
- Select Download for each item if you have only a few, but if you have many, it will be helpful to download all the data files at one shot. The alternative way to download files is to use the Download Manager.
- The “TNM download manager” can be found at the URL shown in the Figure 4 below.

The **Download Manager** is a Java-based application that runs on the local desktop computer and enables download of multiple products without requiring the user to click each individual download link.

**TNM Download Client** users can export and save their Cart items as a Comma-Separated Value (CSV) file containing download links which can be easily imported into the Download Manager using the File-Import menu option.

The Download Manager can be downloaded in two ways:

- For Java-enabled browsers (like IE, Firefox and Safari) use the following Java Web Start Launch Button:
- For other cases (including Google Chrome users) use the [Direct Download Link](#) to download and save the latest .jar file and Open this file from your local computer.

*Please Note: A current version of Java must be installed on your local computer. See [Java.com](#) for instructions and links for downloading and installing an up-to-date Java Runtime Environment (JRE).*

### User's Guide

The *National Map* Download Manager [User's Guide](#) can be viewed in a PDF format by selecting the link below. The guide will be updated as new features or changes to existing features are made.

[User's Guide](#)

### Current Known Issues

The [Known Issues](#) document lists any issues that have been found by users or developers. It will be updated as issues are found and corrected. Before submitting an issue, please, check here to see if the problem has already been documented. For questions or to provide feedback on this product, please contact us at: [tnm\\_help@usgs.gov](mailto:tnm_help@usgs.gov)

### Version Archive

The following table contains all releases of the Download Manager. See the Notes column for changes and fixes that have been made since the previous release.

Version	Notes	Download
Download Manager V1.6	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.6.jar</a>
Download Manager V1.5	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.5.jar</a>
Download Manager V1.4	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.4.jar</a>
Download Manager V1.3	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.3.jar</a>
Download Manager V1.2	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.2.jar</a>
Download Manager V1.1	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.1.jar</a>
Download Manager V1.0	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V1.0.jar</a>
Download Manager V0.9.1	<a href="#">Release Notes</a>	<a href="#">TNMDownloadManager_V0.9.1.jar</a>

Figure 4 TNM Download Manager by the USGS

- There are 2 versions, of this application. One is the browser supported version and the second is a stand-alone application version.
  - For the Browser supported version to work, your browser must have java plug-in installed.
  - If the browser does not have java installed, download the stand-alone application.
    - This version requires your PC to have Java installed. If java is not installed you can find it at the following Download links:
      - Java Development Kit
        - <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
      - Java Runtime Environment
        - <http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>
- Once you have everything installed and TNM running on your PC, you can import the excel file downloaded from the National Map Viewer.
  - Click on File -> Import, and select the csv file and this will add all the products of the cart to the downloading queue.
  - Once you can see all the results in the queue, click on Start All.
  - This will download all your files in GDB format.

Note: You can change path for downloading files from File -> Options.

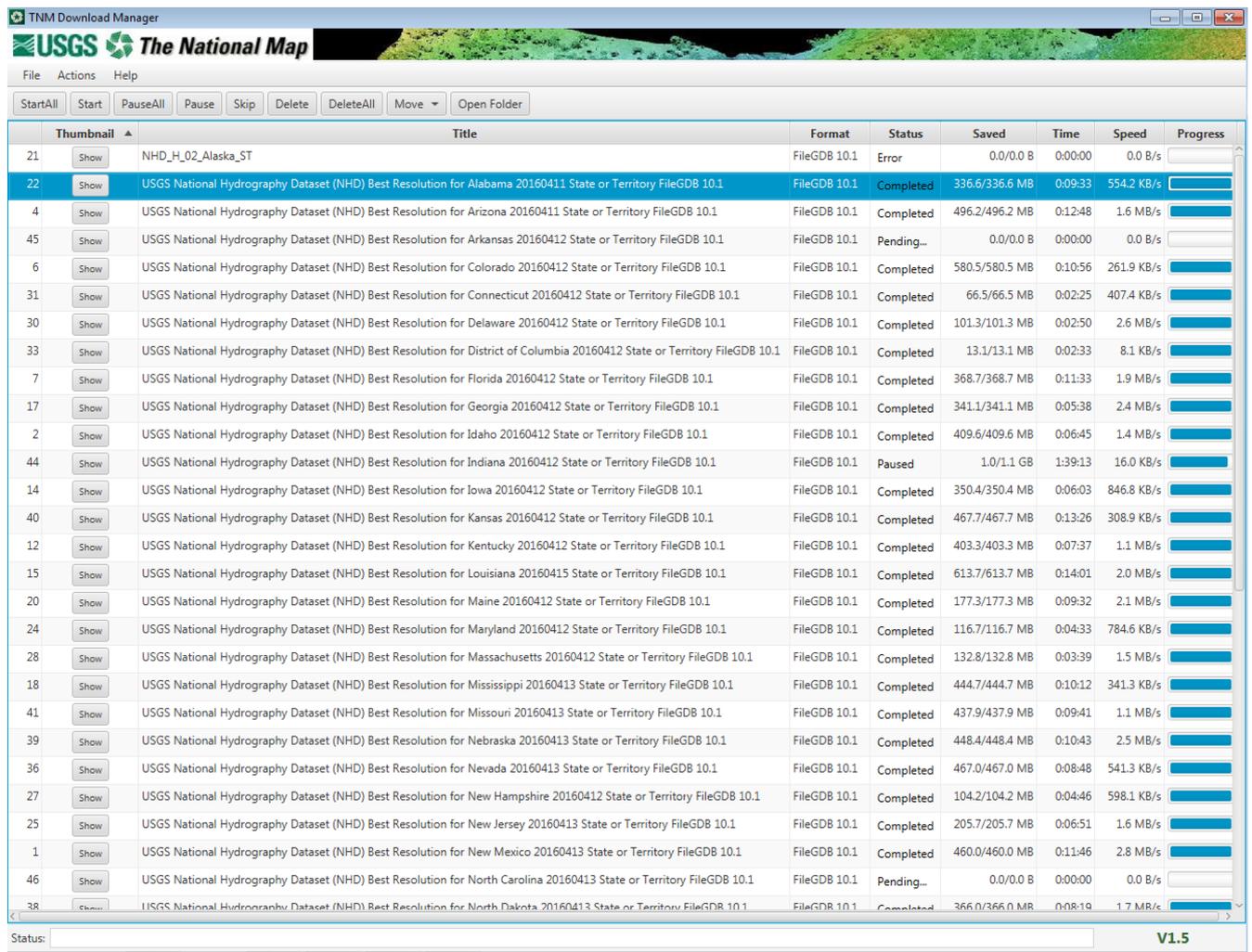


Figure 5 USGS TNM Downloader

- Once all the files are downloaded, they will be by default in ZIP format. Extract all the files and you'll find the required datasets which are in GDB format. The GDB files are up-to-date with the data on Amazon S3 and will serve as an input for the conversion process.

## Step 2: Converting these files from GDB to XML

- Load these files into ArcMap to individually extract all these files into XML format.
  - The ArcMap Catalog will show all the files for the connected folder.
  - Right Click on each individual GDB file and Select Export -> XML Workspace Document. If the file already exists and is included, the export option will be grey.
  - To export, select the path where you want to extract the XML files.
    - While exporting you'll be asked "What do you want to export?" In that response, select the radio button for "Data"
    - Also, select the "Binary" version for the storage of geometry.
    - And select the check-box for exporting the meta-data.
  - Once this is done, hit "Next", and select all the Geo-Features that you want to export. Ideally it should be all the available features.
  - Hit "Finish", and the software will give you a finished version of XML file which is converted from GDB.
  - Figure 6 shows a part of this process:

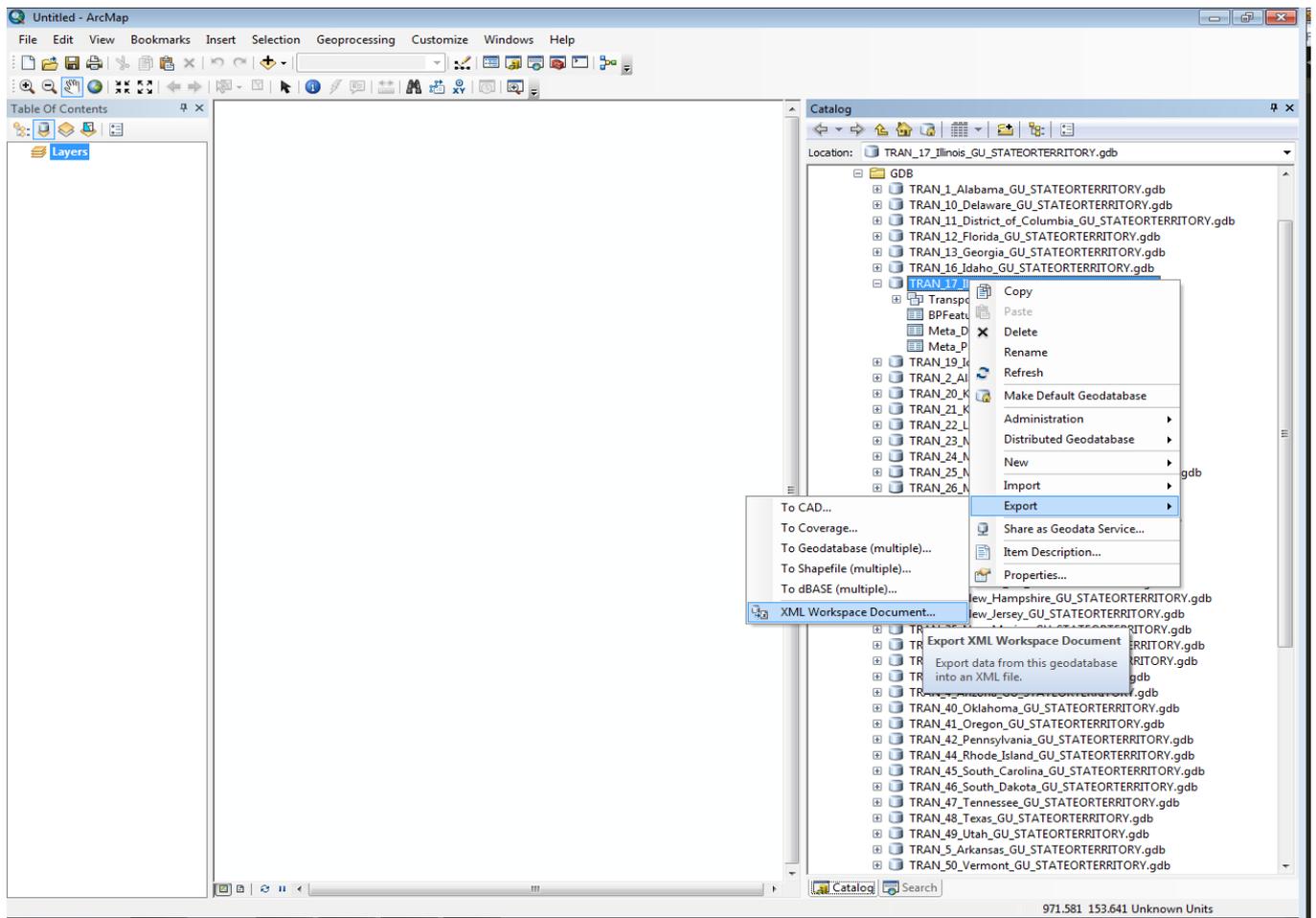


Figure 6 GDB to XML via ArcMap

### Step 3: Converting these files from XML to JSON

- Download Google App Engine
  - <https://cloud.google.com/appengine/downloads>  
Be sure you have the Python version 27 that supports the Google App Engine Platform.
  - After installing the Google App Engine open the “Google App Engine Launcher.” Download the application [https://github.com/rushirajenuji/N3\\_to\\_JSON](https://github.com/rushirajenuji/N3_to_JSON) and import using File –Add Existing Application. Browse to the root directory of the app and select it.
  - Set up the port when asked and start the application. Figure 7 describes the Google App Engine Launcher that is running the Current Application.
  - This can be accessed from the localhost on your browser: <http://localhost:8080/> (where 8080 is the port). (Results may not appear with graphic style as shown in Figure 8.)
- Figure 8 also indicates that the code RDF Translator uses the Python library RDFLIB found at:
  - <https://github.com/RDFLib/rdfliib>
  - RDFLib is a Python library for working with RDF.

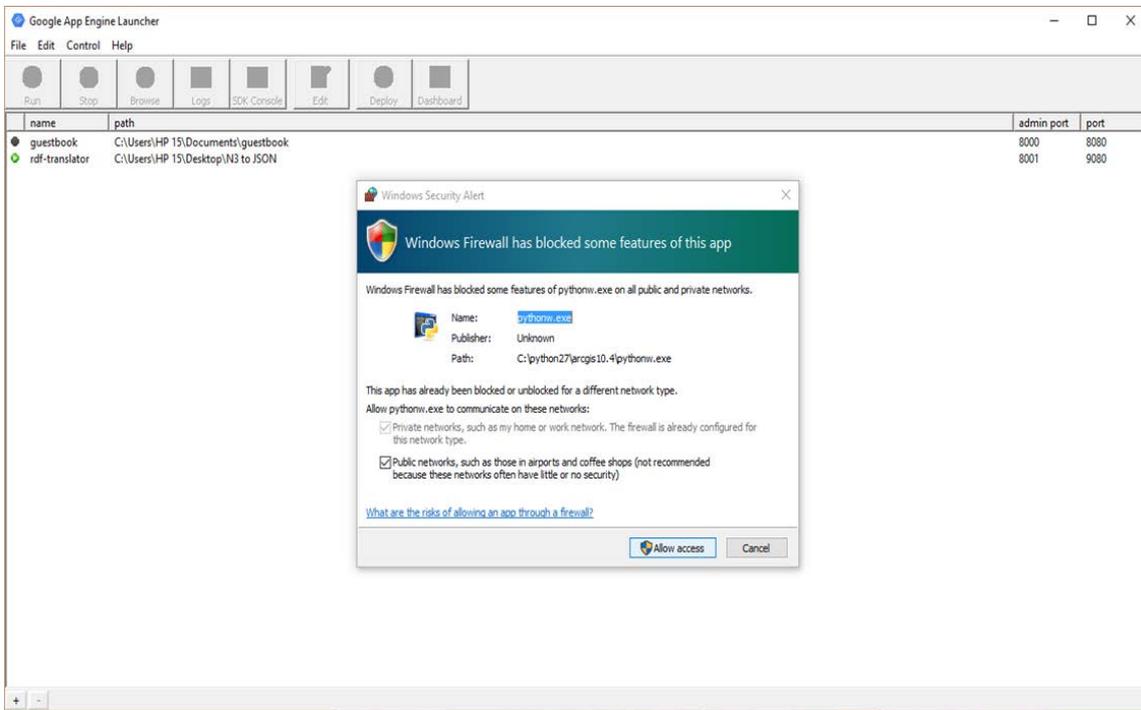


Figure 7 Running Google App Engine Launcher

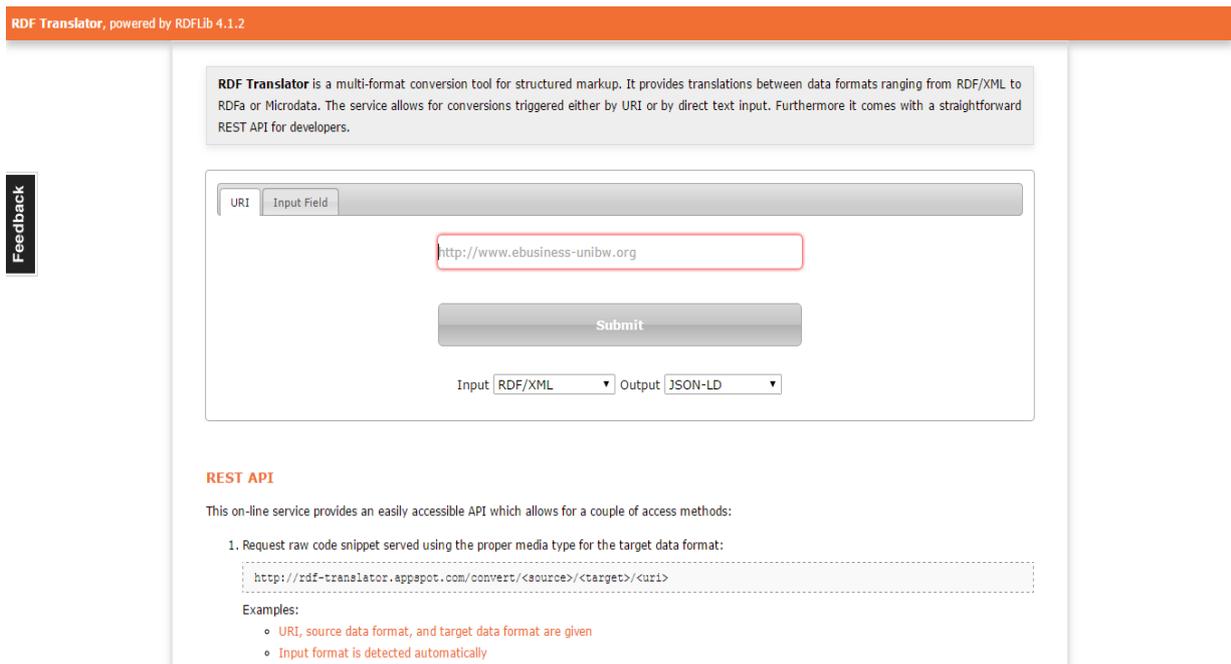


Figure 8: XML to JSON

Setting up GIT-HUB if not already installed.

- Download is available from <https://git-scm.com/download/win>
- Open Git-Bash (preferably in ADMIN mode) and navigate to the location where the XML files are saved in the Step 2 of the conversion process.
- Once you navigate to the location of the data files, run the code given below:

```
curl --data-urlencode content@file_to_convert_name.xml http://localhost:9080/convert/n3/json-ld/content > converted_filename.json
```

Where: file\_to\_convert\_name.xml is changed to the name of the file to convert

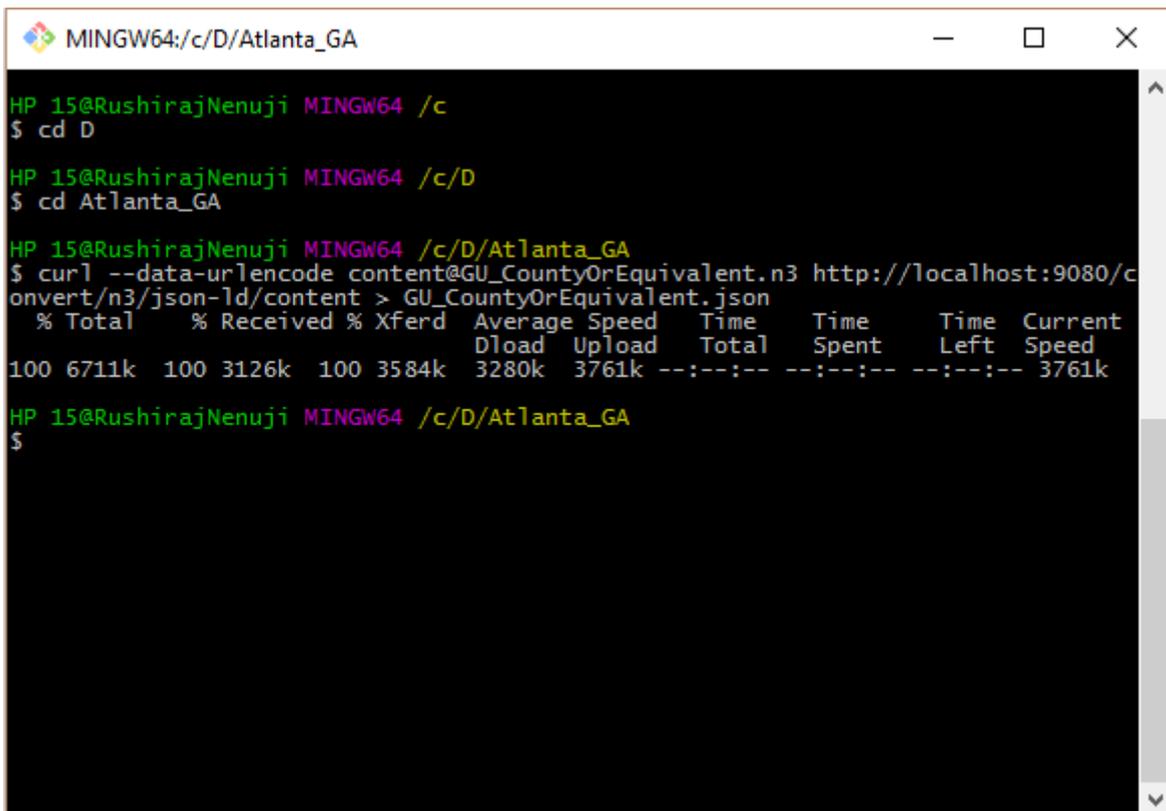
Localhost:9080 is changed to the port number

Converted\_filename.json is changed to the new file name

Example:

```
curl --data-urlencode content@NHD_H_1029_GDB.xml http://localhost:8080/convert/n3/json-ld/content > NHD_H_1029_GDB.json
```

Figure 9 shows how the Git Bash will respond to the above command.



```
HP 15@RushirajNenuji MINGW64 /c
$ cd D
HP 15@RushirajNenuji MINGW64 /c/D
$ cd Atlanta_GA
HP 15@RushirajNenuji MINGW64 /c/D/Atlanta_GA
$ curl --data-urlencode content@GU_CountyOrEquivalent.n3 http://localhost:9080/convert/n3/json-ld/content > GU_CountyOrEquivalent.json
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left     Speed
100 6711k  100 3126k  100 3584k   3280k  3761k  --:--:--  --:--:--  --:--:--  3761k
HP 15@RushirajNenuji MINGW64 /c/D/Atlanta_GA
$
```

Figure 7 Running the command to Convert XML to JSON

Note: This code was developed by Alex Stolz, Location: Neubiberg, Germany as a part of his PHD project.

Implemented version of his work can be found at <http://rdf-translator.appspot.com/> location

The code can be found at:

URL: <https://bitbucket.org/alexstolz/rdf-translator>

URL: [https://github.com/rushirajnenuji/N3\\_to\\_JSON](https://github.com/rushirajnenuji/N3_to_JSON)

- **Advantages:**
  - Data Privacy is maintained as this code runs on local-machine.
  - Supports multiple formats such as XML / JSON / N3 / Turtle / etc.
- **Dis-Advantages:**
  - Currently the localhost has maximum file size limit of 32 MB for uploads .
  - It is little bit slow as the data set is very large.
- **Possible Solutions:**
  - To extend the size from 32 MB to 1-2 GB, Google has provided an extension for Python libraries called “Blobstore” that can support larger data files. More details can be found at:
    - <https://cloud.google.com/appengine/docs/python/blobstore/>

By: Rushiraj Nenuji,  
Developer Intern,  
CEGIS,  
USGS.

Guided By: Dalia Varanka,  
Project Manager,  
CEGIS,  
USGS.