

**Standard Operating Procedure for**

**Data Translation**

**Into GGDM 3.0**

**Version 2**

Prepared by:

Army Geospatial Center

7701 Telegraph Road

Alexandria, Virginia 22315-3864

Revision History

This record shall be maintained throughout the life of the document. Each published update shall be recorded. Revisions are a complete re-issue of entire document.

|  |  |  |
| --- | --- | --- |
| Version Number | Date | Change Description |
| 1.0 |  | Initial version |
| 2.0 |  | Formatting updates; GGDM to TDS translation clarification; metadata explanation; creating relationships option |
|  |  |  |

# Table of Contents

[Table of Contents 2](#_Toc500839764)

[Purpose 4](#_Toc500839765)

[Collect Sources 4](#_Toc500839766)

[Analyze Data 4](#_Toc500839767)

[Metadata Analysis 4](#_Toc500839768)

[GDB Attribute Evaluation 5](#_Toc500839769)

[Translation 7](#_Toc500839770)

[Post-Processing 7](#_Toc500839771)

[Metadata Tools 7](#_Toc500839772)

[GDB Checker 14](#_Toc500839773)

[GDB Attribute Evaluator 16](#_Toc500839774)

[Translation QC 16](#_Toc500839775)

[Gather Resources 16](#_Toc500839776)

[Complete QC 16](#_Toc500839777)

[Source Counts 16](#_Toc500839778)

[Attribute checks 16](#_Toc500839779)

[Appendix A GGDM to TDS to GGDM Python Script Translation 18](#_Toc500839780)

[Source Analysis 18](#_Toc500839781)

[Run Translation 18](#_Toc500839782)

[Review Output Logs 20](#_Toc500839783)

[Appendix B MGCP TRD3/TRD4 to GGDM 3.0 Translation 21](#_Toc500839784)

[Source Analysis 21](#_Toc500839785)

[Run Translation 22](#_Toc500839786)

[OPTIONAL Pre-Load Validate 26](#_Toc500839787)

[Appendix B1 Unmapped MGCP TRD3/4 to GGDM 3.0 Features 28](#_Toc500839788)

[Appendix C Create an Esri-formatted Correlation Mapping 29](#_Toc500839789)

[Esri Cross Reference Personal Geodatabase Overview 29](#_Toc500839790)

[Cross Reference Table Structure 29](#_Toc500839791)

[Gather Resources 30](#_Toc500839792)

[Cross Reference Mapping Table Setup for Editing 30](#_Toc500839793)

[Mapping table creation and updating 30](#_Toc500839794)

[Dataset Mapping: Identify source feature classes and the corresponding GGDM 3.0 Features 31](#_Toc500839795)

[Field Mapping: Match source fields with compatible GGDM fields 32](#_Toc500839796)

[Value Mapping: Match source field values with compatible GGDM field values 32](#_Toc500839797)

[Creating the Cross-reference database 33](#_Toc500839798)

[Appendix D Metadata Tools Listing 39](#_Toc500839799)

# Purpose

The purpose of this SOP is to detail the steps for translating feature datasets into the GGDM 3.0 schema, creating and populating required metadata features and attributes, and QC, as they apply to the SSGF Data Warehouse.

# Collect Sources

For each translation, collect the following items necessary to begin translation:

1. Source datasets – These are the datasets to be translated into GGDM.
2. Empty GGDM 3.0 schema – a new schema is necessary for any new translation. The GGDM\_Composite schema will be used in this SOP - GGDM\_Composite\_v3\_0\_20160121.gdb
3. The GGDM 3.0 Entity Catalog will be utilized when creating new translators: GGDM 3.0 Entity Catalog.xlsx
4. Translation Checklist – This checklist will provide an overview of steps for translation and should be filled out as each step is completed.

It is recommended to create a Translation Folder to be used during the translation process to keep the data organized and ensure each step is followed.

# Analyze Data

The source dataset should be reviewed to determine if an existing translator can be used, or if a new translator will need to be created.

1. If the source dataset is a known data schema with an existing translator [TDS 6.1 (Map of the World), TDS 4.0 (Project MAAX), GGDM 2.1], use the existing Python script translator. [Appendix A]
2. If the source dataset is MGCP TRD3/TRD4, use the existing Correlation database with the Esri ArcCatalog 10.1+ Production Mapping Extension Data Loader Tool. [Appendix B]
3. If the source dataset is an unknown data schema (such as a native source), it is considered to be a one-time translation. It is recommended to create a Correlation database and use the Esri ArcCatalog 10.1+ Production Mapping Extension Data Loader Tool. [Appendix C]

### Metadata Analysis

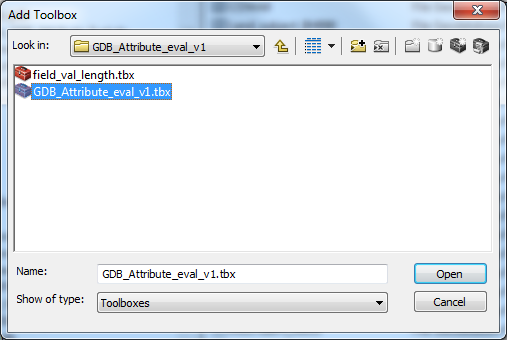
Using the data dictionary (if provided) or the source dataset, determine what metadata is available. There are required Metadata attributes in GGDM 3.0 that will need to be populated at the end of the translation. Some of this information includes:

* + 1. Metadata features or geometries
    2. Source creation / revision dates
    3. Content originator
    4. Source schema
    5. Source scale density
    6. Security classifications and dissemination controls

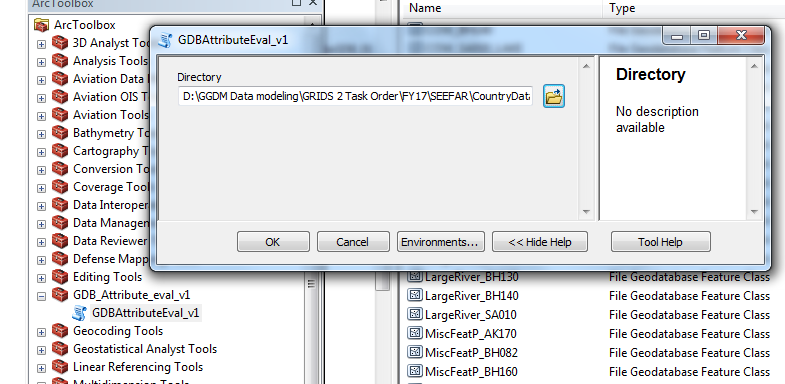
### GDB Attribute Evaluation

Use this tool to evaluate your source dataset in determining features, attributes, and values. The output report is a dataset, feature, attribute, and values spreadsheet. Use the spreadsheet to analyze your source dataset.

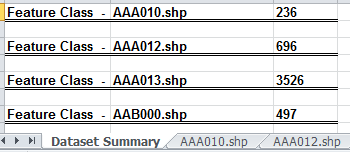
* 1. Open Esri ArcCatalog
  2. Open the Toolbox and add the GDBAttributeEval\_v1 toolbox



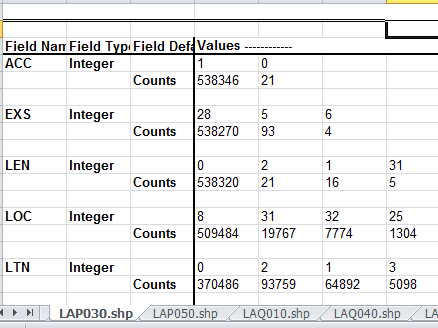
* 1. Double-click the GDBAttributeEval\_v1 toolbox to open it, then double-click the script to run.
  2. Input the folder containing the raw data to be analyzed and hit ‘OK’ for the tool to run.
     1. The tool will run on shapefiles or a file geodatabase (GDB). The shapefiles or GDB should be the only items in the folder provided for the tool to run on. If multiple datasets are within the input folder, they will all be included in the tool output.



* 1. Once the tool completes a spreadsheet called ‘*Filename*\_GDB\_Eval.xlsx’ will be added to the folder containing the data. The output will contain a report of the dataset contents, attribute and domain values for each feature type, and feature counts.
     1. The ‘Dataset Summary’ Worksheet contains a listing of all feature classes in the input folder (will notate either shapefiles or feature classes within a database), as well as the feature counts for each class.



* + 1. Each shapefile or feature class will have a Worksheet that lists all attributes, what type of field the attribute is, all values present in the dataset, and the counts associated with each attribute value.



# Translation

1. If the source dataset is a known data schema with an existing translator [TDS 6.1 (Map of the World), TDS 4.0 (Project MAAX), GGDM 2.1), use the existing Python script translator instructions in Appendix A.
2. If the source dataset is MGCP TRD3/TRD4, use the existing Correlation database with the Esri ArcCatalog 10.1+ Production Mapping Extension Data Loader Tool and follow the instructions in Appendix B.
3. If the source dataset is an unknown data schema (such as a native source), it is recommended to create a Correlation database and use the Esri ArcCatalog 10.1+ Production Mapping Extension Data Loader Tool and follow the instructions in Appendix C.

# Post-Processing

After the source dataset is translated into GGDM 3.0, you must post-process by adding required NMF Metadata and checking the translations in the resulting GGDM 3.0 file geodatabase.

If your source data did not contain ResourceSrf and MetadataSrf features, you will need to create these polygons for your translated data. The simplest way is to:

* Open Esri ArcMap
  + Add the empty ResourceSrf and MetadataSrf feature classes from GGDM 3.0 geodatabase you just translated to.
  + Add your original source data.
  + In an Edit session:
    - Draw a polygon around your data for ResourceSrf and one for MetadataSrf
    - Populate the required attributes [URI/UFI, MDE, EQS, RTL, and others as required]
    - Save your edits
  + Close Esri ArcMap

### Metadata Tools

The following procedures outline the steps required to populate feature level metadata, ResourceSrf and MetadataSrf.

There are 3 steps in the Metadata Tools Python script to populate any missing NMF-required metadata in a GGDM 3.0 file geodatabase. A listing of all python files contained within the overall script can be found in Appendix D.

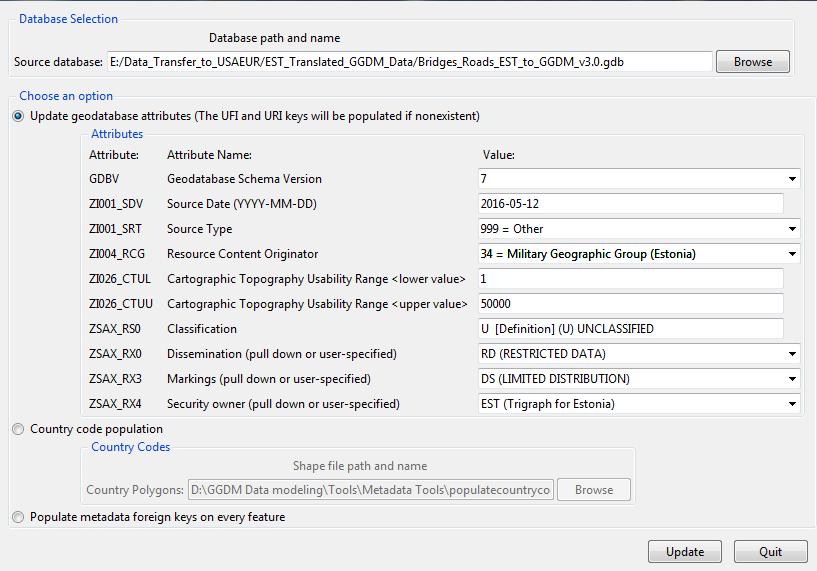
Step 1. The script will populate the primary key (UFI) on all features (URI on DATASET\_S) if the attributes are currently empty. The other feature-level attributes are:

* GDBV Geodatabase Schema Version
* ZI001\_SDV Source Date
* ZI001\_SRT Source Type
* ZI004\_RCG Resource Content Originator
* ZI026\_CTUL Cartographic Topography Usability Range <lower value>
* ZI026\_CTUU Cartographic Topography Usability Range <upper value>
* ZSAX\_RS0 Classification
* ZSAX\_RX0 Dissemination
* ZSAX\_RX3 Markings
* ZSAX\_RX4 Security owner

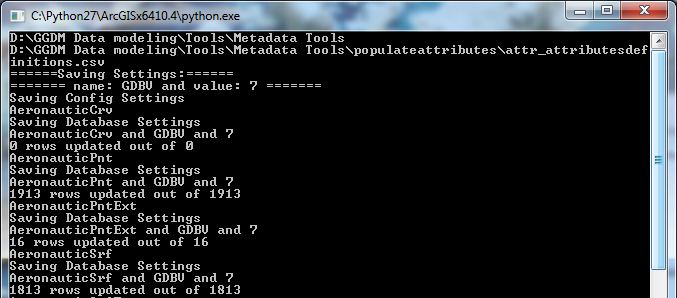
Step 2. The script will populate the Country Code attribute (ZI020\_GE4 Designation : GENC Short URN-based Identifier) on every feature per the authoritative world country boundaries.

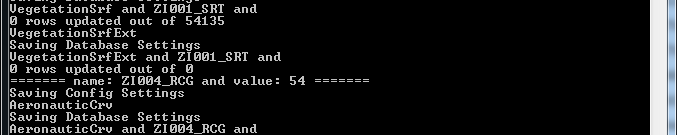
Step 3. The script will populate ZI039S\_UFI Entity Collection Metadata (Surface) : Unique Entity Identifier and the ZI031S\_URI Dataset (Surface) : Unique Resource Identifier (metadata foreign keys) on every feature using the centroid location in relation to the metadata surfaces.

1. Run the Metadata Tools.
   1. In Windows Explorer, navigate to the Metadata Tools folder.
   2. Double-click the ‘main.py’ Python script to start the translation.
   3. Two windows will appear:
      1. A cmd window c:\Python27\ArcGISx6410.3\python.exe – DO NOT close this window. Wait.
      2. The GGDM Metadata Utility window



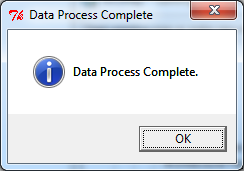
* + 1. In the Source database: Browse to the GGDM 3.0 file geodatabase containing metadata to populate [Note: the script will not overwrite existing metadata] and Select OK.
    2. Select the “Update geodatabase attributes” option [Step 1]
    3. Populate the 9 attributes with the necessary values.
    4. This Option will also populate any empty UFI/URI (primary identifier) attributes [required for metadata relationships]
    5. The ZSAX attributes have suggested drop-down values. If these are not sufficient, you can enter Structured Text.
    6. Select Update to run Option 1.
    7. Processing Results will appear in the cmd window C:\Python27\ArcGISx6410.4\python.exe
    8. The script will process each attribute, showing the actual value populating the attributes.





[Remember if the attribute already has a value, the script will not overwrite it]

* + 1. When the Option 1 script is finished, a message box for “Data Process Complete” will appear.
    2. Select OK.



* + 1. Select the “Country Geometry Population” option [Step 2]
       1. The authoritative world country boundaries are provided and you will need to create a subset of these countries for your specific dataset.
          1. The country boundaries shapefile is in the Metadata Tools folder \populatecountrycodes\world\_country\_file\_polygon\_av\world\_wide\_country\_polygons\_region24.shp. [Ensure this is the shapefile you use since it contains the 24-character Country Codes].
          2. Using Esri ArcMap:

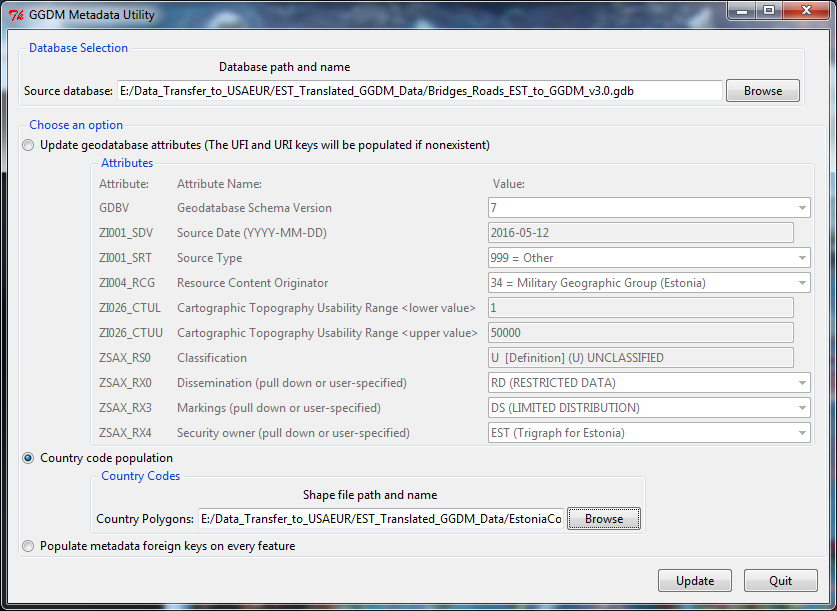
Add world\_wide\_country\_polygons\_region24.shp to your map.

Add ResourceSrf DATASET\_S to your map.

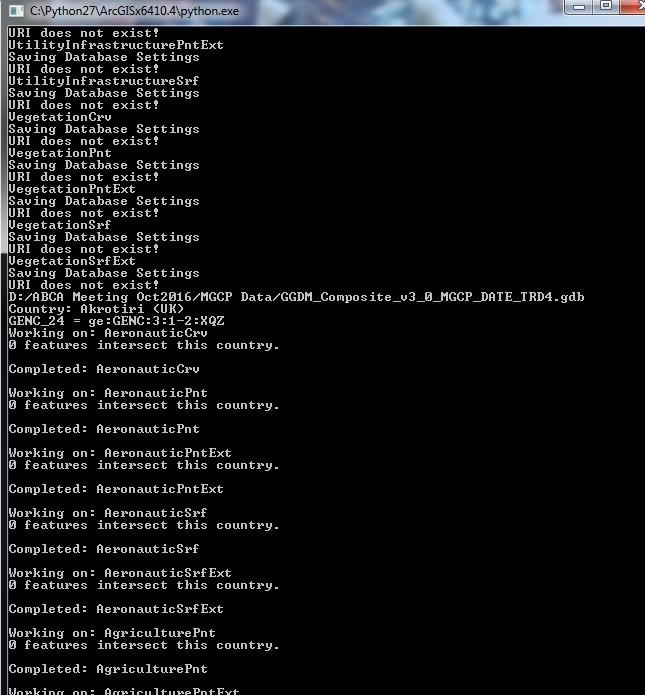
Select by Location of world\_wide\_country\_polygons\_region24.shp that intersect ResourceSrf. This will select all of the countries that are in your dataset.

Right-click the world\_wide\_country\_polygons\_region24.shp on the Layers TOC and select Data🡪 Export Data with the option to export selected features. Choose a folder location and name for the new shapefile with the country selection set. This will be the Countries shapefile used as the input for the Country Polygons.

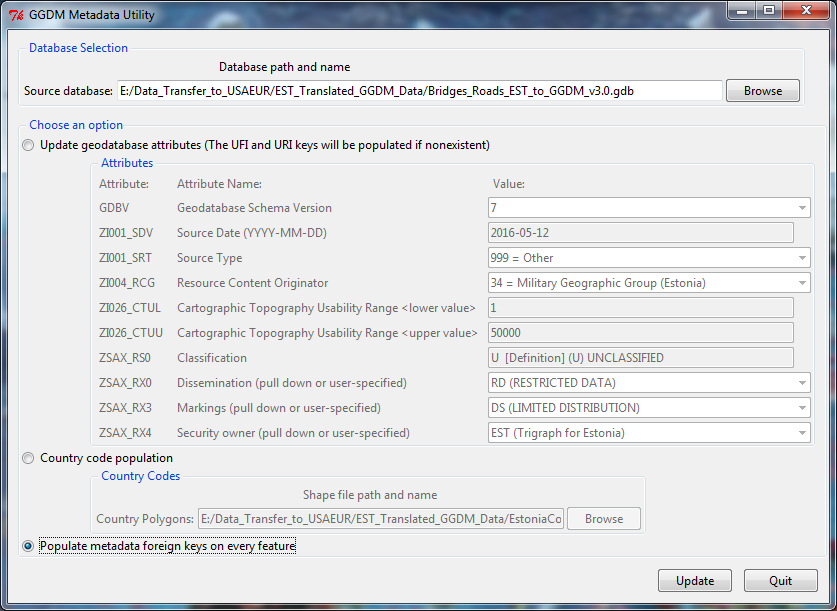
* + - 1. The script will populate the ZI020\_GE4 attribute on every feature utilizing the centroid with a Location match (intersect) to the Country polygon. This requires a parsing of every country so be patient.
      2. Select Update to run Option 2.



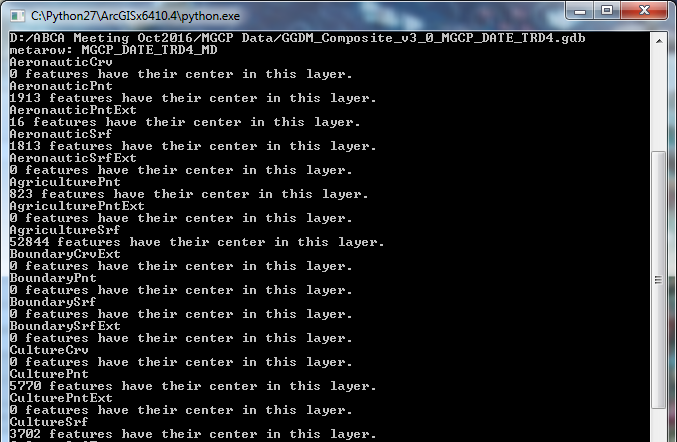
Processing Results will appear in the cmd window C:\Python27\ArcGISx6410.4\python.exe



* + 1. When the Option 2 script is finished, a message box for “Data Process Complete” will appear.
    2. Select OK.
  1. Select the “Populate metadata foreign keys on every feature” option [Step 3]
     1. Select Update to run Option 3



* + 1. Processing Results will appear in the cmd window C:\Python27\ArcGISx6410.4\python.exe



* + 1. When the Option 3 script is finished, a message box for “Data Process Complete” will appear.
    2. Select OK.

An optional step is to use the GGDM Tool for Relationships Builder to complete the relationships and allow users to see the related features and metadata in Esri ArcMap.

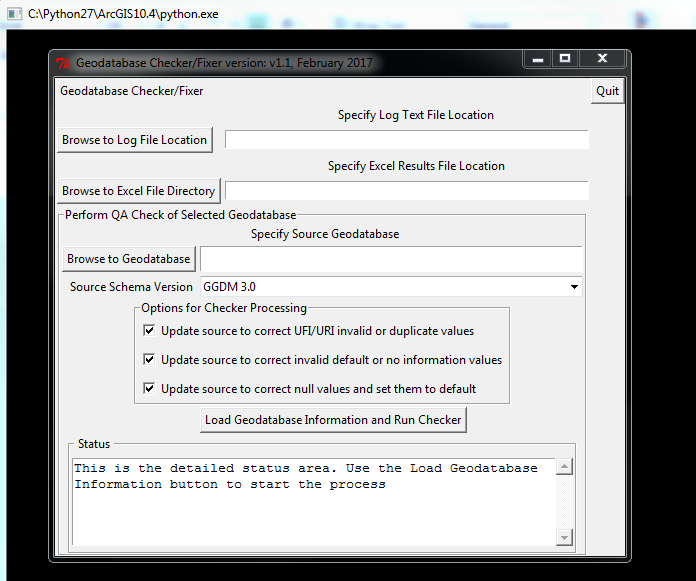
### GDB Checker

After metadata population is complete, run the GDB Checker python script. The Geodatabase Checker/Fixer tool checks the GGDM 3.0 file geodatabase for common issues and fixes them if possible. Specifically, the GDB Checker will identify and replace illegal default or null values and populate Unique Feature Identifiers (UFIs) if they are missing.

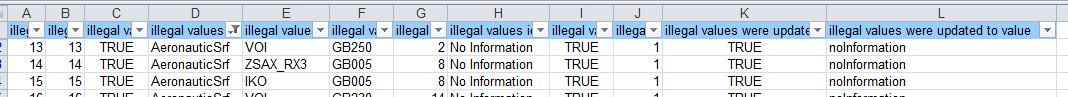
1. Make a copy of the GGDM 3.0 file geodatabase you have just created.

NOTE: it is important to make a copy of the geodatabase, as the tool will make changes to the database and it may be impossible to recover the original version if something goes wrong.

1. Run the Checker.
   1. In Windows Explorer, navigate to the Py\_GDB\_Checker folder.
   2. Double-click the ‘\_Go.py’ Python script to start the translation.
   3. Two windows will appear:
      1. A cmd window c:\Python27\ArcGISx6410.3\python.exe – DO NOT close this window. Wait.
      2. The Geodatabase Checker/Fixer Utility window



1. Click “Browse to Log File Location” and navigate to the folder containing the GGDM 3.0 file geodatabase. A log file will be created in this location with a name based off of the current date and time.
2. Click “Browse to Excel File Directory” and navigate to the folder containing the GGDM 3.0 file geodatabase. An Excel spreadsheet will be created in this location with a name based off of the current date and time.
3. Click “Browse to Geodatabase” and navigate to the GGDM 3.0 file geodatabase you created.
4. Ensure that the “Source Schema Version” is set to GGDM 3.0.
5. Check all three boxes under “Options for Checker Processing”.
6. Click “Load Geodatabase Information and Run Checker” to run the tool. The processing status will update in both the GUI and the command window. The tool may take several hours to complete depending on the size and complexity of the geodatabase.
7. When the tool has finished running, navigate to the folder containing the database and output log. Open the ‘Checker\_Results\_2017xxxxxxx.xlsx’.
   1. The ‘Content’ Worksheet contains a summary of the inputs.
   2. The ‘values’ Worksheet contains a list of feature classes, attributes, and any illegal values encountered by the tool. In the example below, several values of ‘No Information’ were corrected to the appropriate ‘noInformation’ value.



* 1. If Column K contains the word TRUE, then the tool was able to fix the error. If the value is FALSE, then the tool encountered an illegal value and was not able to automatically fix it. All of these issues will have to be manually reviewed and corrected. In the example below, the values of WDAC (column E) were not fixed and require manual corrections (using Esri ArcGIS).



### GDB Attribute Evaluator

After the GDB Checker tool and review is complete, the GDB Attribute Evaluator should be run on the final GGDM 3.0 file geodatabase. The spreadsheet output report should be used to compare the source dataset with the GGDM 3.0 resulting data to ensure no data loss occurred during translation.

# Translation QC

The translated GGDM 3.0 file geodatabase is now ready for QC.

Translation QC focuses on accounting for all source features in the translated GGDM 3.0 file geodatabase and reviewing the transfer of the feature attributes.

## Gather Resources

Ensure the translation folder contains:

1. Source dataset and Eval spreadsheet
2. Translation log files or mapping databases
3. GGDM 3.0 file geodatabase and Eval spreadsheet

## Complete QC

Any errors, anomalies, or anything unusual, interesting, or just odd, should be documented in a ‘QC Summary’ spreadsheet.

### Source Counts

Verification of source counts against translated features is necessary to ensure all features have been successfully translated. Each source feature needs to be accounted for in the translated database or its absence documented. Some examples of verification include:

1. Do the source and translated dataset total feature counts match?
   * 1. If no, can the difference be attributed to non-standard mapping?
        1. If yes, the dataset was translated completely.
        2. If no, check each feature class and applicable mapping.
   1. Do the Feature class / subtype count compare?
      1. If no, can the difference be attributed to non-standard mapping?
         1. If yes, the dataset was translated completely.
         2. If no, check each feature class and applicable mapping.

### Attribute checks

Attribute checks ensure all attributes were properly mapped to the translated GGDM 3.0 file geodatabase. Perform the following checks on all features in each populated feature class using Esri ArcGIS 10.1+.

1. Run ‘Validate Features’
   1. ‘Validate Features’ is found in the ‘Editor’ dropdown on the ‘Editor’ toolbar.
      1. You must be in an edit session and the tool runs only on a selection.
      2. A high feature count may necessitate running the tool on smaller selections.
   2. ‘Validate Features’ will select features with invalid domain values. These should be recorded per subtype in the ‘QC Summary’ tab.
2. Verify UFI is populated for each feature.
   1. This can be checked by opening the ‘Select by Attributes’ dialogue and generating a list of unique values for the UFI field. If null or ‘no information’ values are present, record counts and subtypes of the features in the ‘QC Summary’ tab.
3. Check that all attributed values with a ‘999’ (‘Other’) value have a corresponding OTH entry.
   1. Feature classes and fields with ‘999’ values can be identified in the attribute evaluation output.
4. Check all attributes in populated features that may have an entry in the ‘OTH overwrite tab’.
   1. If multiple fields have ‘999’ populated, each should have a separate OTH entry in the following format: (<field name>: <attribute value>)
5. Verify any Un-mapped source attribute values
   1. As many attributes as possible will be retained in the translated database but there will be a few that do not have an equivalent GGDM field to be translated into and will be dropped.
   2. Ensure any attribute values that are not mapped, and should remain un-mapped, are recorded with an explanation.
6. Check all Coded value domains
   1. Invalid coded domain values will appear without a description.

# Appendix A GGDM to TDS to GGDM Python Script Translation

There are 26 AGC-provided Python scripts provided for the translation which requires access to an Esri ArcGIS 10.1+ license. Functions within the translation are:

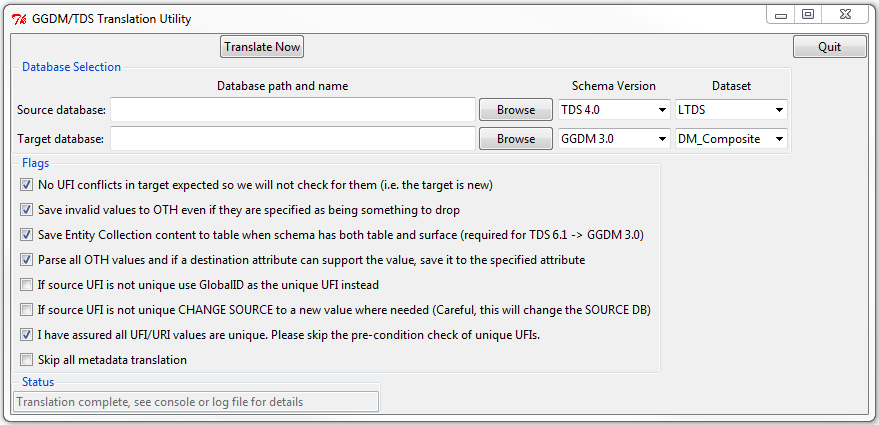
* 1. Transform functions - operate at the level of an attribute translation. These functions can look up other attribute values to derive content, or can use pre-defined mappings to translate attribute content. Many of the transform functions will be specific to a particular translation.
  2. Metadata translation functions - operate on the metadata within the schema. For some translations, this will be common but other translations will require specific metadata handling.
  3. General translation functions - operate on standard inputs specifying feature classes, tables, and attributes that translate.

## Source Analysis

1. Analyze your source file geodatabase and determine the name of the feature dataset.
2. Copy the GGDM 3.0 Composite empty file geodatabase into an appropriate name.
3. When working with TDS 4.0 datasets [i.e. Project MAAX (Magellan, Apollo, Alexander, Xerxes)]:
   1. Ensure the relationship classes found within the feature dataset are deleted. These features lock the database for editing and will cause the translation to fail.
   2. Ensure the translator option ‘If source UFI is not unique, CHANGE SOURCE to a new value where needed’ is selected. The MAAX data has ‘No Information’ populated for all UFI values.
   3. If required, make a copy of the source dataset since the translation will update the source data.
4. When working with TDS 6.1 datasets [i.e. Map of the World (MoW)]:
   1. Ensure the translator option ‘If UFI is not unique use GlobalID as the unique UFI instead’ is selected.
   2. Not all of the MoW data has unique UFI attributes.

## Run Translation

1. Run the Python script to translate.
   1. In Windows Explorer, navigate to the Translate\_GGDM\_TDS\_20170516 folder.
   2. Double-click the ‘\_Go.py’ Python script to start the translation.
   3. Two windows will appear:
      1. A cmd window c:\Python27\ArcGISx6410.3\python.exe – DO NOT close this window. Wait.
      2. The GGDM/TDS Translation Utility window



1. Configure the GGDM/TDS Translation Utility:
   1. In the Source database field, Browse to the database to be translated and select the appropriate Schema Version and Dataset.
   2. In the Target database field, Browse to the database and select the appropriate Schema Version and Dataset.
   3. Several flag options within the translation interface are available depending on the contents of the source dataset.
      1. ‘No UFI conflicts in target expected…’ – Check this option when loading data into a blank TARGET database. If you are loading (adding) into TARGET geodatabase that has data, this should be checked off to ensure no duplicate features are loaded.
      2. ‘Save invalid values to OTH…’ – This will allow values no longer acceptable within TDS/GGDM to map to the OTH field if the user wants to retain that information.
      3. ‘Save Entity Collection content…’ – Per the description, a requirement for TDS 6.1 to GGDM 3.0 due to the portrayal of the metadata in TDS 6.1.
      4. ‘Parse all OTH values…’ – This option will compare the value in the OTH field of the SOURCE to the codes available in the TARGET. If the value is present in the TARGET it will be populated as such instead of ‘Other’ with the OTH field.
      5. ‘If source UFI is not unique…’ – These options should be checked if the UFI field in the SOURCE is either blank OR is not truly unique
      6. I have assured…’ – This option will skip the unique UFIs requirement.
      7. ‘Skip all metadata…’ – This option will skip all metadata translation.
   4. The first 4 check box options AND THE “I have assured…” option are the defaults.
   5. After all options are configured, select the ‘Translate Now’ button at the top of the GGDM/TDS Translation Utility window.
      1. The cmd prompt window will show the status of the translation. There are five steps in the process labeled as Steps 0-4:
         1. Preconditions
         2. Translate RESTRICTION\_INFO\_T
         3. Translate Metadata surfaces
         4. Translate Metadata tables
         5. Translate feature classes
   6. When the translation is complete:
      1. The Status window message ‘Translation complete, see console or log file for details’ will appear.
      2. The cmd window will report: All steps complete.

## Review Output Logs

The output log from the Python tool is saved in the folder containing the scripts. The log file name is shown at the end of the command prompt output generated during processing. It lists the settings used for translation, metadata translation results, and the number of records processed for each feature. If you want to save the output log report, move it to the folder with the translated database. Scan through the log and search for error or warning messages. Any issues encountered should be evaluated and fixed if required.

# Appendix B MGCP TRD3/TRD4 to GGDM 3.0 Translation

To translate MGCP TRD3 or TRD4 shapefiles to GGDM 3.0 Composite file geodatabase use AGC-provided products and the Esri ArcGIS 10.1+ Production Mapping Extension Data Loader tool.

There are four AGC-provided products provided for the translation:

1. A cross-reference spreadsheet in Esri required format for Production Mapping tools - MGCPToGGDM30Final.xlsx
2. A cross-reference database - MGCPCrosstoGGDM30.mdb.
3. An empty GGDM 3.0 Composite file geodatabase - GGDM\_Composite\_v3\_0\_20160121.gdb
4. The GGDM Toolbox with the MGCP Translator Python script/tool to assist in translation [optional].

You will use two of these for the translation:

1. A cross-reference database - MGCPCrosstoGGDM30.mdb.
2. An empty GGDM 3.0 Composite file geodatabase - GGDM\_Composite\_v3\_0\_20160121.gdb

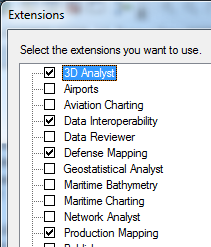
## Source Analysis

Before beginning the translation:

1. Analyze your MGCP shapefile folders to ensure no duplicate subfolders exist (these will cause duplicate features to be translated).
2. Ensure any unmapped MGCP features due to geometry constraints are known. (Appendix B1) Change the geometry of these MGCP features if required.
3. Determine if any of the unmapped MGCP attributes are required. If so, these will need to be mapped separately.
4. Ensure the two AGC-provided products are available.
5. Optional: Validate the AGC-provided cross-reference personal geodatabase against the MGCP source shapefiles and GGDM 3.0 Composite target file geodatabase using Preload Validate Tool. [This can help determine MGCP TRD3 and TRD4 schema shapefiles if this information is required]. The procedure for this is Step 4 Pre-Load Validate.

## Run Translation

1. Open Esri ArcCatalog
   1. Enable the Production Mapping Extension by clicking on Customize -> Extensions and checking Production Mapping. Select Close.

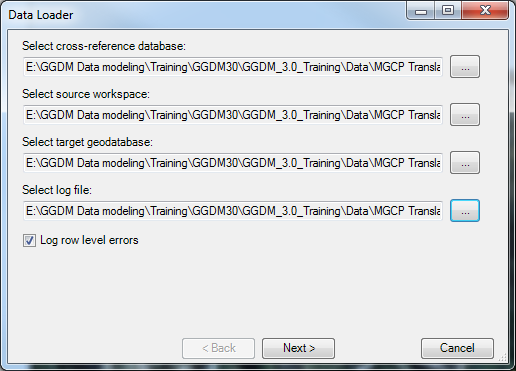


* 1. Add the Production Mapping toolbar by clicking on Customize -> Toolbars -> Production Mapping.
  2. Click on the Data Loader tool on the Production Mapping toolbar (third one):



2. Translation of MGCP shapefiles:

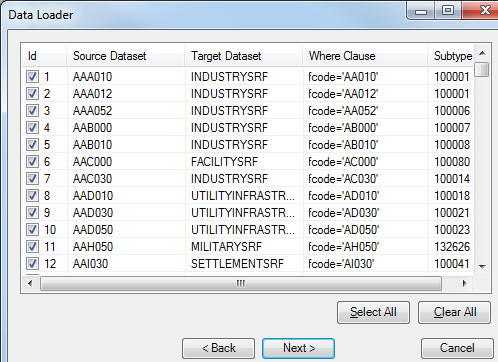
* 1. Click the ellipsis (…) next to the Select cross-reference database field. The Select cross-reference database dialog box appears. Select the cross-reference personal geodatabase provided - MGCPCrosstoGGDM30.mdb. Click Open.
  2. Click the ellipsis (…) next to the Select source workspace field. The Select source workspace dialog box appears. Navigate to your source MGCP shapefiles. Click Open.
  3. Click the ellipsis (…) next to the Select target geodatabase field. The Select target workspace dialog box appears. Navigate to an empty GGDM 3.0 Composite file geodatabase. Click Open.
  4. Click the ellipsis (…) next to the Select log file field. The Specify log file dialog box appears. Navigate to your translation folder and name the log file. Click Save.



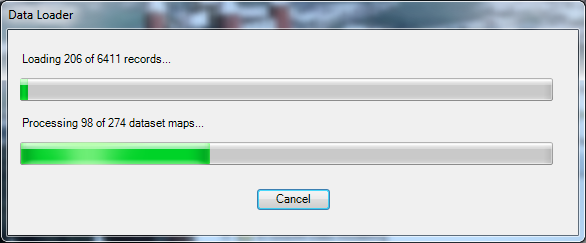
* 1. Click Next >.
  2. The Data Loader box appears with the names of the MGCP Source Dataset shapefiles and the GGDM 3.0 Target Dataset feature classes and the Where Clause for each subtype or feature.

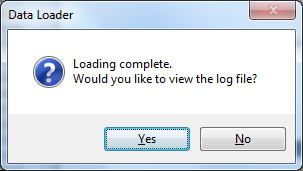
Tip: The Subtype number is in the GGDM 3.0 Entity Catalog.xlsx on the Feature Index worksheet.

If necessary, click Select All or Clear All to check or uncheck all the check boxes next to the feature class names.

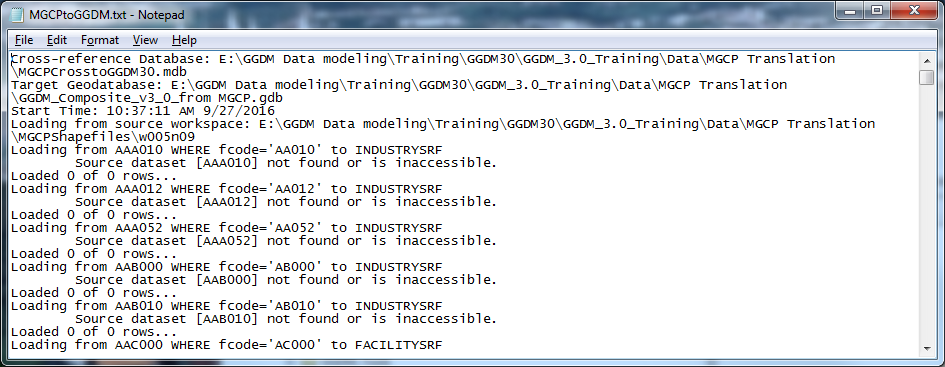


* 1. Click Next >.
  2. The Data Loader Summary appears and shows the cross-reference database, the MGCP shapefiles folder (source), the GGDM 3.0 Composite file geodatabase (target), and log file path. Check that this information is correct.
  3. Click Finish.
  4. The Data Loader window appears with a progress bar that displays the features and subtypes as they are loaded. When the loading process is finished, a message appears to notify you that loading is complete (approximately 2 minutes).
  5. Since this is a schema-to-schema translation, the Data Loader window will always display 274 dataset maps even though your shapefiles may be less than this.





* 1. Click Yes to view the log file.

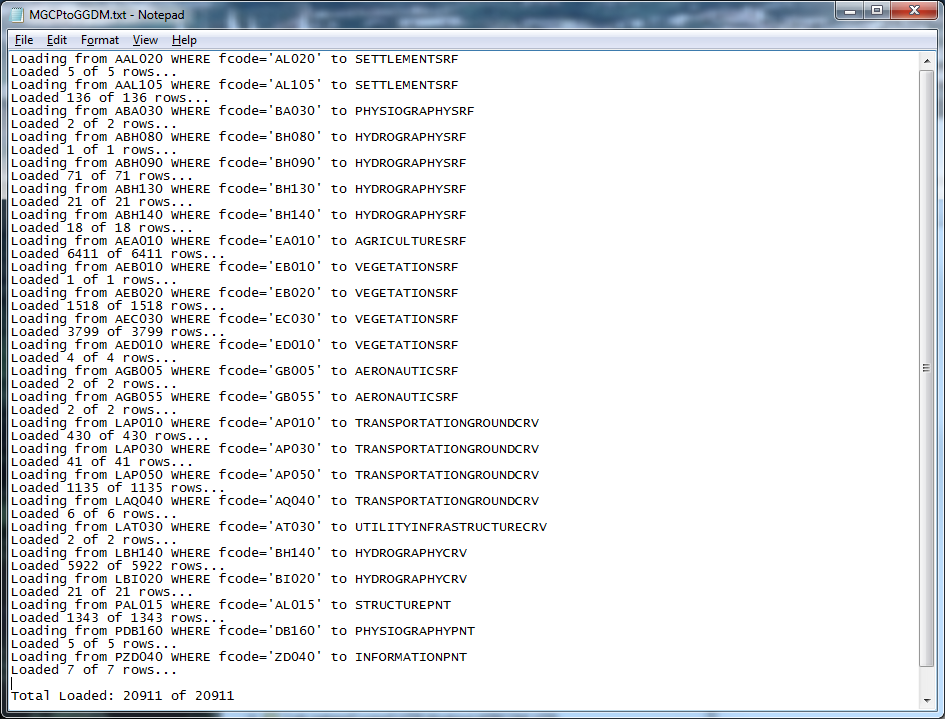


* 1. The cross-reference database contains correlations for ALL of the MGCP features (see Appendix B1 for the complete list) so you will see “Loaded 0 of 0 rows…” if the MGCP shapefile is not in your source workspace.
  2. The log file will contain “Loaded # of # rows…” for the MGCP shapefiles that are in your source workspace.

1. Checking the translation.

You will need to compare the log file results against the now populated target GGDM 3.0 file geodatabase to ensure all data was converted correctly.

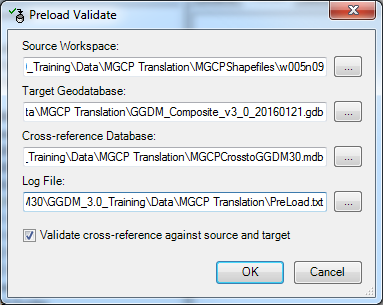
* 1. Compare the Log file Loaded # of # rows with the populated .dbf files names from step 3 – this will ensure you all of the shapefiles were translated.



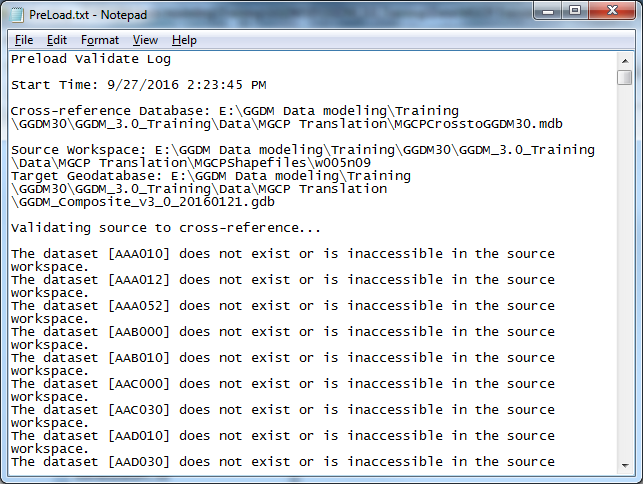
* 1. Using Esri ArcGIS ArcCatalog, compare the Log file Loaded # of # rows with the MGCP shapefiles features – this will ensure you all of the shapefiles features (or records) were translated.

## OPTIONAL Pre-Load Validate

1. Pre-Load Validate - validate the cross-reference personal geodatabase against the MGCP source shapefiles and GGDM 3.0 target file geodatabases
   1. Click the ‘Preload Validate’ icon (second one) on the Production Mapping toolbar. The Preload Validate dialog box appears.
   2. Click the ellipsis button (…) next to the Source Workspace text box. The Select source workspace dialog box appears. Browse to your source MGCP shapefiles. Click Select.
   3. The Preload Validate dialog box appears.
   4. Click the ellipsis button next to the Target Geodatabase text box. The Select target geodatabase dialog box appears. Browse to the empty GGDM 3.0 Composite file geodatabase. Click Select.
   5. The Preload Validate dialog box appears.
   6. Click the ellipsis button next to the Cross-reference Database text box. The Select cross-reference database dialog box appears. Browse to the cross-reference personal geodatabase provided - MGCPCrosstoGGDM30.mdb. Click Add.
   7. The Preload Validate dialog box appears.
   8. Click the ellipsis button next to the Log File text box. The Specify log file dialog box appears. Navigate to the Translation folder and name the log file PreLoad.txt. Click Save.
   9. The Preload Validate dialog box appears.
   10. Check the Validate cross-reference against source and target box.



* 1. Click OK.
  2. The MGCP shapefiles and GGDM 3.0 file geodatabase are compared to the cross-reference database, and any differences are reported in the log file. Once the validation process is complete, a message appears asking if you want to see the log file.
  3. Click Yes to view the log file.
  4. The log file opens in a text editor.



* 1. In the Validating source to cross-reference…
     1. Since this translation is for MGCP TRD3 and TRD4, you will see fields that do not exist on the source dataset. In MGCP TRD3, the field is CCN. In MGCP TRD4, the field is CPYRT\_NOTE. This is a quick way to determine the MGCP schema release of your source shapefiles.
     2. MGCP is a subset of GGDM so there will be fields in the target dataset (GGDM) that are not in the cross-reference database.

## Appendix B1 Unmapped MGCP TRD3/4 to GGDM 3.0 Features

There are MGCP TRD3/4 features with geometries not available in GGDM 3.0 (NAS 7.0).

* Two Point [Features](#_TGD_Features_that) [Area features are mapped]
* Two Point [Features](#_TGD_Features_that) [Line features are mapped]
* One Line Feature [Point and Area features are mapped]

In order to translate these five MGCP features, they will need to be converted to the appropriate geometries.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | MGCP |  | GGDM | | |  |
| Shapefile | Feature Name | Geom | | GGDM Code | Feature Name | Geom |
| PAC030 | SETTLING POND | Point | | AC030 | SETTLING\_POND\_S | Area |
| PBA050 | BEACH | Point | | BA050 | BEACH\_S | Area |
| PBH010 | AQUEDUCT | Point | | BH010 | AQUEDUCT\_C | Line |
| PGB050 | AIRCRAFT REVETMENT | Point | | GB050 | DEFENSIVE\_REVETMENT\_C | Line |
| LDB160 | ROCK FORMATION | Line | | DB160 | ROCK\_FORMATION\_P/\_S | Point or Area |

# Appendix C Create an Esri-formatted Correlation Mapping

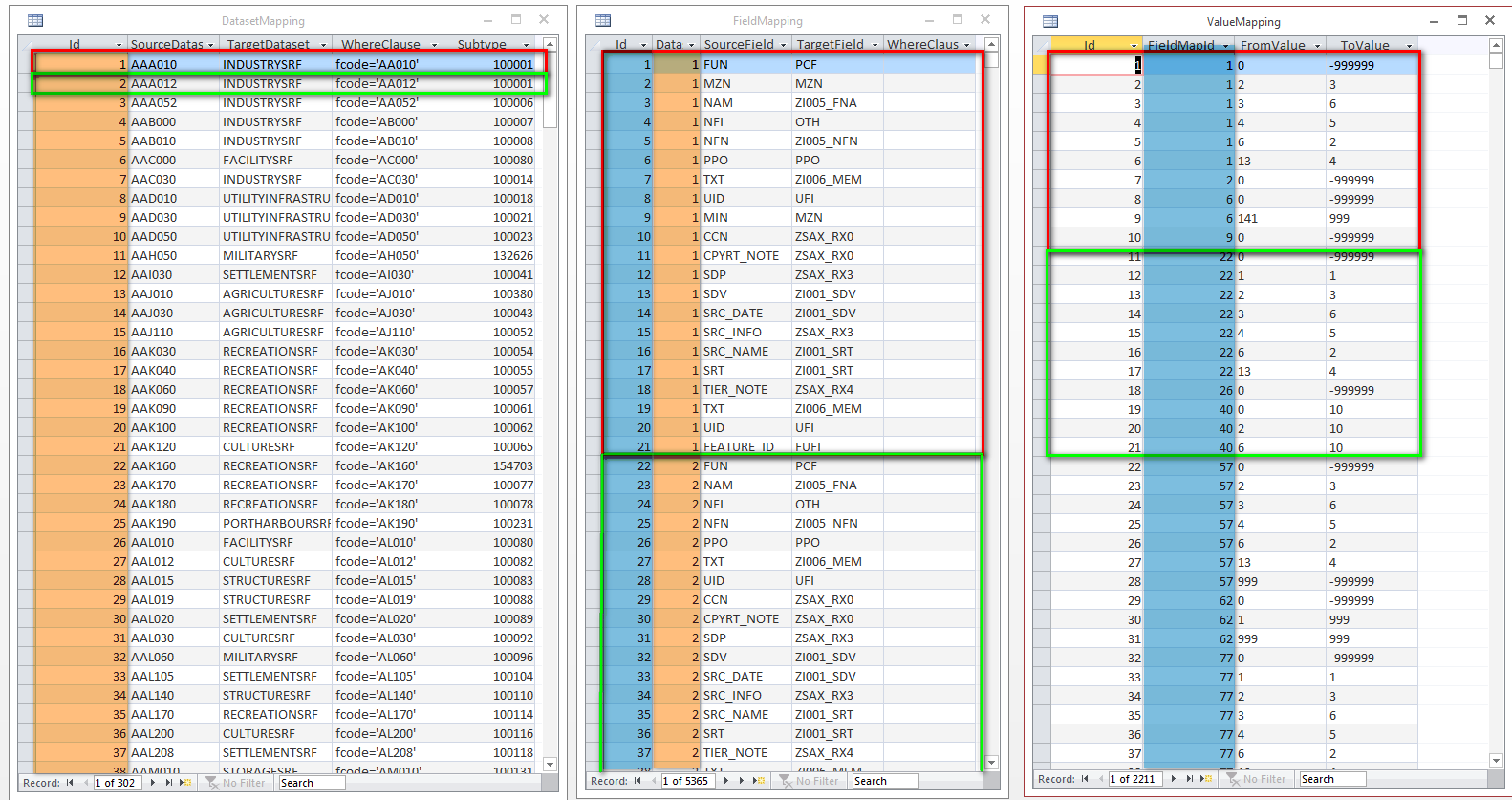
## Esri Cross Reference Personal Geodatabase Overview

The cross reference mappings designate the relationships between feature classes, subtypes, and attributes (fields and values) in a source schema and a target schema. We are translating into the GGDM 3.0 schema, using the GGDM 3.0 Entity Catalog spreadsheet for guidance. The cross reference mappings can be developed either as a:

* Microsoft Excel spreadsheets and then converted to a personal geodatabase using the Esri ArcGIS Production mapping Extension Data Loader Arc Create Cross-Reference tool or
* Esri personal geodatabase (using Microsoft Access)

## Cross Reference Table Structure

The cross reference mapping spreadsheet or database contains three essential Worksheets or Tables: Dataset Mapping, Field Mapping, and Value Mapping.



* 1. Dataset Mapping - contains the relationships between features in the source dataset and corresponding features in GGDM. There are five columns in the Dataset Mapping Worksheet:

1. ID: Each row will have a unique identifier. This is named DatasetMapId in worksheets or tables. [NOTE: This is an AutoNumber in Access so do not generate – let the system do it]
2. SourceDataset: Name of the source dataset feature. [NOTE: If this is a shapefile, do not include the .shp in the name]
3. TargetDataset: Name of the GGDM Feature Class containing the feature subtype the source feature is being mapped to.
4. WhereClause: An optional SQL statement used to select subsets of the source dataset based on an attribute value.
5. Subtype: Numeric ID of the GGDM feature subtype. [NOTE: These are found in the GGDM Entity Catalog]
   1. Field Mapping - contains the relationships between attributes or fields for each source feature and corresponding attributes or fields in the GGDM feature. There are five columns in the FieldMapping Worksheet:
6. ID: Each row will have a unique identifier. This is named FieldMapId in worksheets or tables. [NOTE: This is an AutoNumber in Access so do not generate – let the system do it]
7. DatasetMapID: The corresponding ID from DatasetMapping Worksheet.
8. SourceField: Field name from the source feature.
9. TargetField: Corresponding field name from target GGDM feature.
10. WhereClause: SQL statement used to select subsets of the source field based on attribute values for mapping to different target fields.
    1. Value Mapping - contains mappings for specific values in source and target fields. These include coded integer values (that have a corresponding definition). There are four columns in the ValueMapping Worksheet:
11. ID: Each row will have a unique identifier. [NOTE: This is an AutoNumber in Access so do not generate – let the system do it]
12. FieldMapID: The corresponding ID from FieldMapping Worksheet.
13. From Value: The source field value.
14. TargetField: The target (GGDM 3.0) field value

## Gather Resources

You will need:

* GGDM 3.0 Entity Catalog spreadsheet
* GDB\_Eval spreadsheet (from source dataset)
* SourceToGGDM30Mappings spreadsheet to be created

## Cross Reference Mapping Table Setup for Editing

Editing and updating the cross-reference databases can be performed in Excel or Access. It is easier to create the initial mappings in Excel and then load them into the Esri Personal geodatabase (using Microsoft Access). This allows you to read and understand the mappings before Esri assigns Ids and uses only numeric values. Use the SourceToGGDM30Mappings.xlsx for a template to start, as it contains all three Worksheets.

## Mapping table creation and updating

Creating mapping tables consists of matching features, fields, and then values. A simplified process is:

1. Identify source features and the corresponding GGDM 3.0 Feature Classes and Features
2. Match source fields with compatible GGDM fields [NOTE: Capture any value mappings at this time]
3. Create the required Field Mappings and generate a cross-walk between source and target values

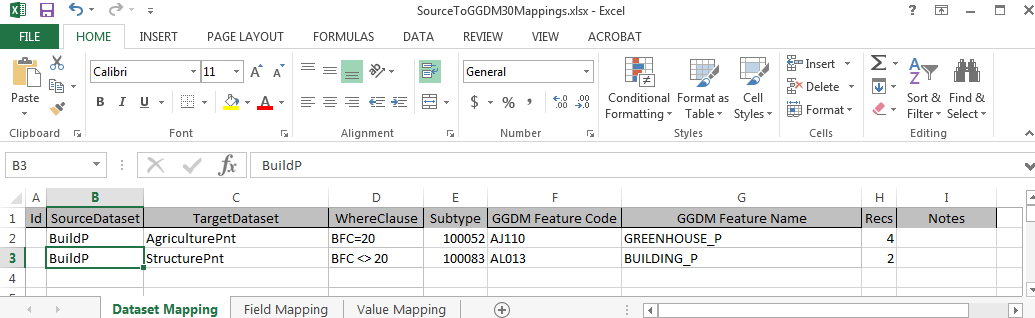
### Dataset Mapping: Identify source feature classes and the corresponding GGDM 3.0 Features

1. Use the GDB\_Eval Dataset Summary worksheet to copy all of the source features into the Dataset Mapping Worksheet.
   1. For example, the source dataset contains a feature named BuildP
   2. Note the geometry type (ex. Point)
   3. Determine any “helpful” attributes, such as feature codes.
2. Identify the target dataset and subtype(s).
   1. Determine the GGDM 3.0 Feature Class and Feature for the source dataset features.
   2. In the case of BuildP, this is probably mapped to GGDM 3.0 Building\_P. Using the GGDM 3.0 Entity Catalog, you can determine the Feature Class and Subtype NumericID for Building\_P:

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature Name** | **Feature Code** | **Composite** | **NumericID** |
| [BUILDING\_P](file:///C:\Projects\SEEFAR\GGDM_v3.0_schema_and_EC\%5b#]Feature_Schema_1!A8504:F8504) | AL013 | StructurePnt | 100083 |

* 1. So the TargetDataset = ‘StructurePnt’ and the Subtype = 100083.
  2. [NOTE: Source features may map to multiple GGDM features based on attribution]

1. Add WhereClause if applicable.
   1. Only include a WhereClause when needed
   2. In the case of BuildP, there was a BFC (Building Function Code)=20 (Greenhouse) so the WhereClause would be:



1. Continue mapping all of the source dataset features to GGDM 3.0 features until complete.

### Field Mapping: Match source fields with compatible GGDM fields

1. Use the GDB\_Eval Dataset Summary worksheet to identify all of the source feature attributes or fields and copy into the Field Mapping Worksheet. [NOTE: If an attribute is empty, skip it]
   1. For example, the source dataset feature BuildP contains a field called EXS (Existence Status). The GGDM 3.0 target field is PCF (Physical Condition). This is a common field mapping.
   2. Value Mappings are required on our example since PCF is a coded domain, so use the Notes column to capture this.
2. Add WhereClause if applicable.
3. The completed Field Mapping is:

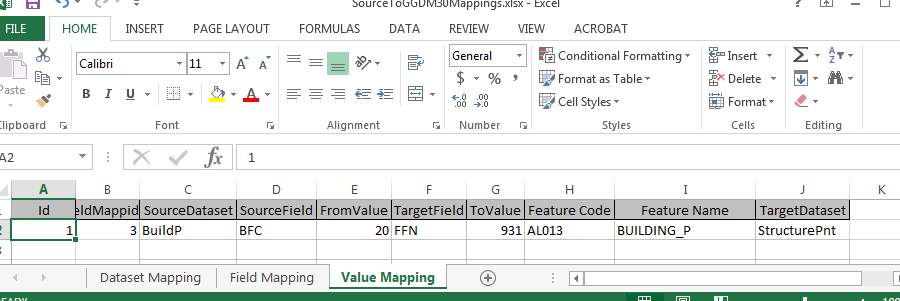
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| WhereClause | SourceDataset | SourceField | TargetField | GGDM Feature |
| (BFC <> 20) | BuildP | exs | PCF | StructurePnt |

|  |  |
| --- | --- |
| Subtype | Subtype Code |
| Building | 100083 |

1. Continue the field mappings for all source features and fields,

### Value Mapping: Match source field values with compatible GGDM field values

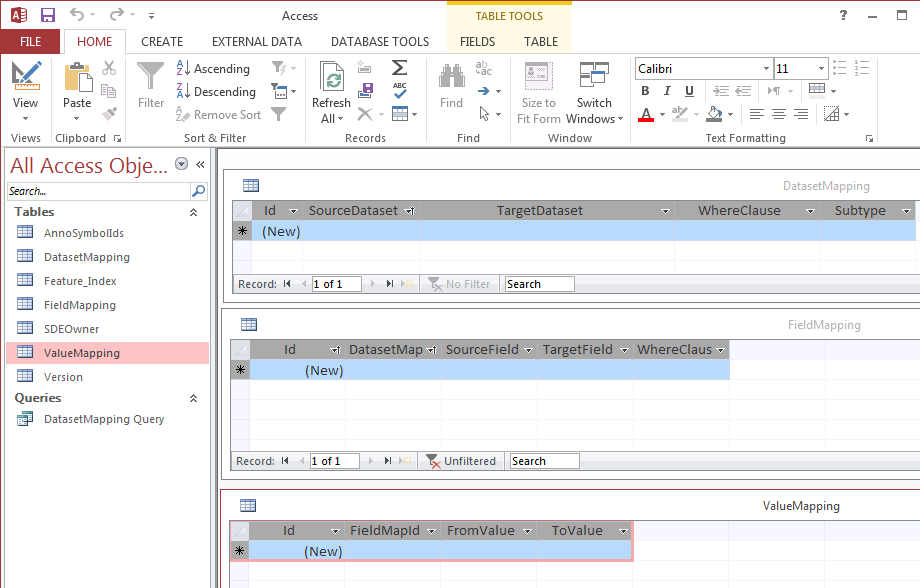
1. Use the GDB\_Eval Dataset Summary worksheet to identify all of the source feature field values for the Value Mapping Worksheet. [NOTE: only include values where they are mapped to a GGDM field that requires a different value for the same result – such as a name field containing strings or a width field containing real numbers]
2. During the Field Mapping, you captured all of the required Value Mappings in the Notes column. Filter this to exclude Blanks and copy the resulting rows into the Value Mapping Worksheet.
3. In the example for BFC, place of worship=2, and the GGDM FFN field=931. So the row will be:



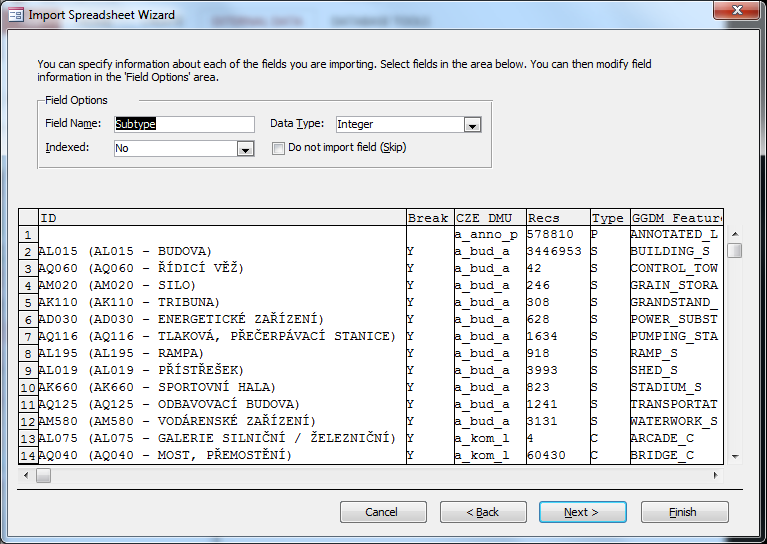
1. Continue adding all of the value mappings.

### Creating the Cross-reference database

1. In your translation folder, use Esri ArcCatalog and create a blank personal geodatabase or copy an existing correlation personal geodatabase, which you will erase all of the records from. [NOTE: Since Esri is so particular about formatting, this is easier]
2. Exit ArcCatalog
3. Use Microsoft Access to open the personal geodatabase you just created
4. Delete any records in the 3 tables (if you copied an existing geodatabase)



1. Use EXTERNAL DATA 🡪 Excel to Import in the Dataset Mapping worksheet of the spreadsheet you just completed.
   1. Select Import the source data into a new table in the current database.
   2. Click OK
   3. Ensure the Dataset Mapping worksheet is highlighted and Select Next >
   4. Select First Row Contains Column Headings and Select Next >
   5. Ensure the Subtype Field Name is a Data Type : Long Integer
   6. Do not import field (Skip) if there is a column you do not need



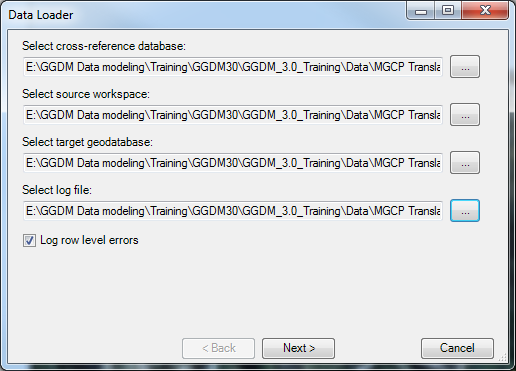
* 1. Select Next >
  2. Select No primary key and Select Next >
  3. Ensure the table is NOT named Dataset Mapping and Select Finish
  4. Select Close
  5. Ensure no errors occurred and open the Dataset mapping table you just created.

2. Translation of data:

1. Click on the Data Loader tool on the Production Mapping toolbar (third one):



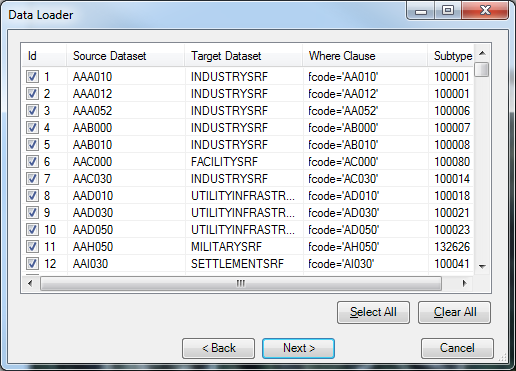
* 1. Click the ellipsis (…) next to the Select cross-reference database field. The Select cross-reference database dialog box appears. Select the cross-reference personal geodatabase you just created. Click Open.
  2. Click the ellipsis (…) next to the Select source workspace field. The Select source workspace dialog box appears. Navigate to your source data. Click Open.
  3. Click the ellipsis (…) next to the Select target geodatabase field. The Select target workspace dialog box appears. Navigate to an empty GGDM 3.0 Composite file geodatabase. Click Open.
  4. Click the ellipsis (…) next to the Select log file field. The Specify log file dialog box appears. Navigate to your translation folder and name the log file. Click Save.



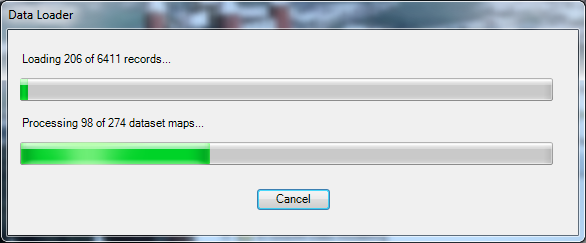
* 1. Click Next >.
  2. The Data Loader box appears with the names of the Source Dataset and the GGDM 3.0 Target Dataset feature classes and the Where Clause for each subtype or feature.

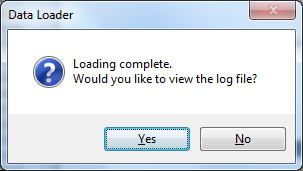
Tip: The Subtype number is in the GGDM 3.0 Entity Catalog.xlsx on the Feature Index worksheet.

If necessary, click Select All or Clear All to check or uncheck all the check boxes next to the feature class names.

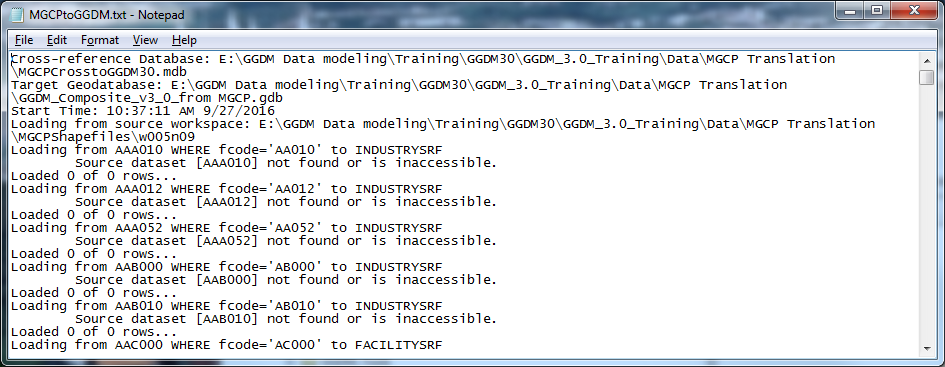


* 1. Click Next >.
  2. The Data Loader Summary appears and shows the cross-reference database, the source, the GGDM 3.0 Composite file geodatabase (target), and log file path. Check that this information is correct.
  3. Click Finish.
  4. The Data Loader window appears with a progress bar that displays the features and subtypes as they are loaded. When the loading process is finished, a message appears to notify you that loading is complete (approximately 2 minutes).





* 1. Click Yes to view the log file.

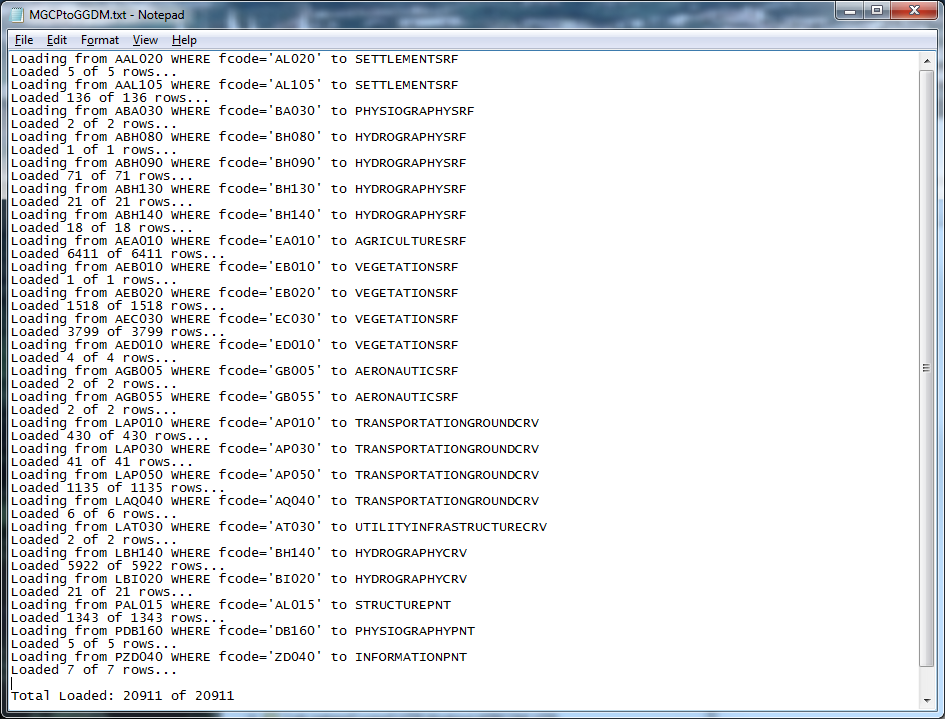


* 1. The cross-reference database contains correlations for ALL of the source features.
  2. The log file will contain “Loaded # of # rows…” for the source data that are in your source workspace.

1. Checking the translation.

You will need to compare the log file results against the now populated target GGDM 3.0 file geodatabase to ensure all data was converted correctly.

* 1. Compare the Log file Loaded # of # rows with your source data – this will ensure you all of the data were translated.



# Appendix D Metadata Tools Listing

Files in the Python utility developed to populate required NSG Metadata Foundation (NMF) required metadata for GGDM 3.0 file geodatabases:

Files:

* main.py – Main starting point, initiates the GUI.
* helperfunctions.py - Functions that are universally used throughout the Metadata Tools.
* \Metadata Tools\developerfiles\xmlparser.py – XML Parser used for developer purposes only. It reads the suggested XML files and outputs the values.
* \Metadata Tools\populateattributes\classes.py – Python script to generate the attributes for update.
* \Metadata Tools\populateattributes\criteria.py – Python script to check if the user provided values fit the criteria before data processing.
* \Metadata Tools\populateattributes\dataprocessing.py – Python script to update the metadata default attributes in a geodatabase and save the settings in the configuration file.
* \Metadata Tools\populatecountrycodes\dataprocessing.py – Python script to match country polygons to geodatabase features and populate the country code attribute.
* \Metadata Tools\populateforeignkeys\dataprocessing.py – Python script to populate the 2 metadata foreign keys on every geodatabase feature.

Configuration Files:

* \Metadata Tools\Config.txt – generated for each unique user instance.
* \Metadata Tools\ populateattributes \attr\_attributesdefinitions.csv – configuration file for Option 1 to populate metadata attributes.
* \Metadata Tools\ populateattributes \attr\_picklists.csv – configuration file for the values of the drop-down lists for Option 1.
* Metadata Tools\populatecountrycodes\world\_country\_file\_polygon\_av – Shapefile for world country polygons.