Technical Forums (/s/forums) Knowledge Base (/s/knowledge-base) Ideas (/s/bridea/acideasULQ_bı Log in Q0(

Article

← FME Desktop (/S/Topic/0TO4Q000000QL9uWAG/Fme-…

Working with Geodatabase Domains: Writing A Coded Domain

Jun 9, 2022 • Knowledge

Product Type
FME Desktop

FME Version
2022.0

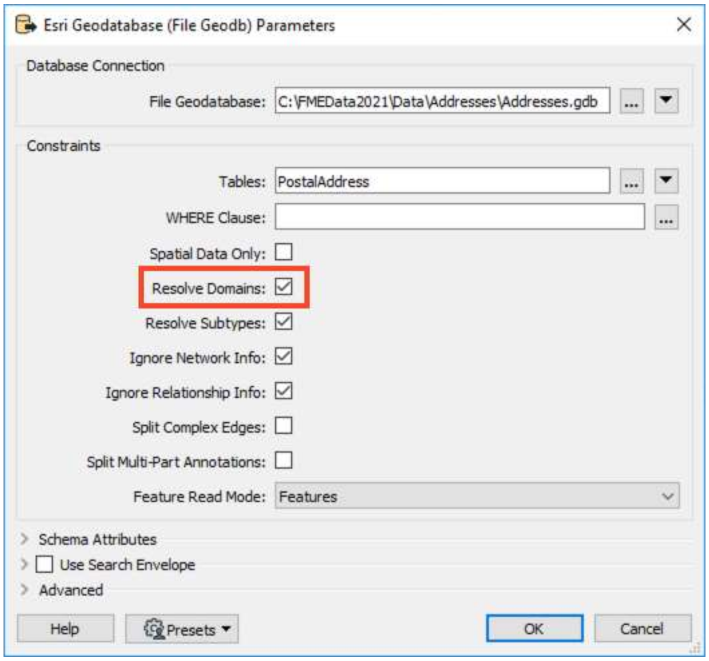
Tutorial: Tutorial: Geodatabase Transformations (https://community.safe.com/s/article/tutorial-getting-started-with-complex-geodatabase) | **Previous:** How to use an Esri Template (https://community.safe.com/s/article/how-to-use-an-esri-template) | **Next:** Working with Geodatabase Subtypes: Writing a Subtype (https://community.safe.com/s/article/working-with-geodatabase-subtypes)

Introduction

A domain is a set of rules that define permitted values for an attribute. They are used to constrain data values in order to ensure data integrity. A domain is defined in a geodatabase as a unique entity and can be applied to any attribute in any feature class within the geodatabase that contains the domain definition. There are two types of geodatabase domains (coded domains and range domains) and both are supported by FME. A coded domain is essentially a list of multiple valid values while a range domain is a single permitted range of numeric values.

Domain Reading

When reading a geodatabase, FME has an option to resolve domains.



Esri Geodatabase (File Geodb) Parameters

Database Connection

File Geodatabase: C:\FMEData2021\Data\Addresses\Addresses.gdb

Constraints

Tables: PostalAddress

WHERE Clause:

Spatial Data Only: □

Resolve Domains: ☑

Resolve Subtypes: ☑

Ignore Network Info: ☑

Ignore Relationship Info: ☑

Split Complex Edges: □

Split Multi-Part Annotations: □

Feature Read Mode: Features

When this option is checked an additional attribute will be added called _<attribute>_resolved_ which contains the textual description of the coded attribute value.

Domain Writing

Note: If you can, we would recommend using a geodatabase template as described in [How to Use an Esri Template (https://community.safe.com/s/article/how-to-use-an-esri-template)](https://community.safe.com/s/article/how-to-use-an-esri-template)**.** This is generally easier and more flexible than defining domains within FME feature types.

FME has the capability to write to existing or create new geodatabase tables. As such, when writing geodatabase domains, the workflow requirements will tend to fall under one of two scenarios:

- Write to an existing table using an existing domain.
- Write to a new table using an existing domain.

These scenarios will be controlled primarily by the following writer parameters:

- Data Type: coded_domain / range_domain
- Validate Features to Write parameter value: Yes / No

Despite the scenarios mentioned below, it should be noted that using an XML Workspace Document (containing the required domain definitions) or a file geodatabase as a template on a geodatabase writer, is the most efficient and highly recommended means of writing geodatabase domains. For more information on importing an XML document or a geodatabase Template with a schema, please see the Geodatabase Writer documentation.

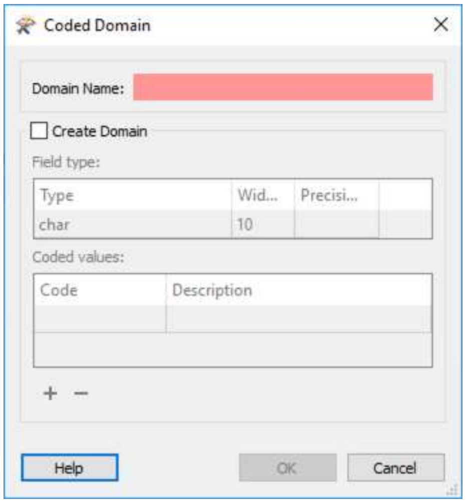
Scenario 1: Writing to an existing table using an existing domain

Any data written to an existing domain field is, by default, inserted as normal. As the table already exists, its attribute(s) will already be associated with the required domain and there is no need to set any parameter to define this connection.

However, if you wish to validate incoming data – for instance, compare it to a domain definition to ensure it has valid attribute values – then you must set the writer parameter “Validate Features to Write” to Yes. If the validation parameter is set to No, the data would pass into the Geodatabase without error, despite the fact that some data values may not meet domain rules.

Scenario 2: Writing to a new table using an existing domain

When creating a new table that is to use an existing domain, the attribute that needs to be associated with a domain should be given the data type coded_domain or range_domain (depending on its type) in the schema definition. One would then click the Edit button in the attribute width field and enter the name of the existing domain in the "Domain Name" section.



Domain Name:

☐ Create Domain

Field type:

Type	Wid...	Precisi...
char	10	

Coded values:

Code	Description

Limitations

There are a few limitations in regard to domain writing.

Firstly, it is not possible to write to an existing table and to either create an association with an existing domain or create an entirely new domain. That is because this association is wrapped up in the table definition, and an existing table definition cannot be changed by FME. You would need to drop the existing table and re-create it entirely in order to be able to do this.

Secondly, for the same reason stated above, creating a domain is a one-off translation. You would set the data type to coded_domain/range_domain for the initial process, but subsequent loads of the data should be done with the data type changed back the actual type of data (char, integer, etc.)

Finally, it is not possible to create a domain dynamically (i.e. the domain definition should be manually defined prior to execution and not defined during the workspace process).

The following example demonstrates Scenario 2: Writing to a new table and using an existing domain through the use of a template.

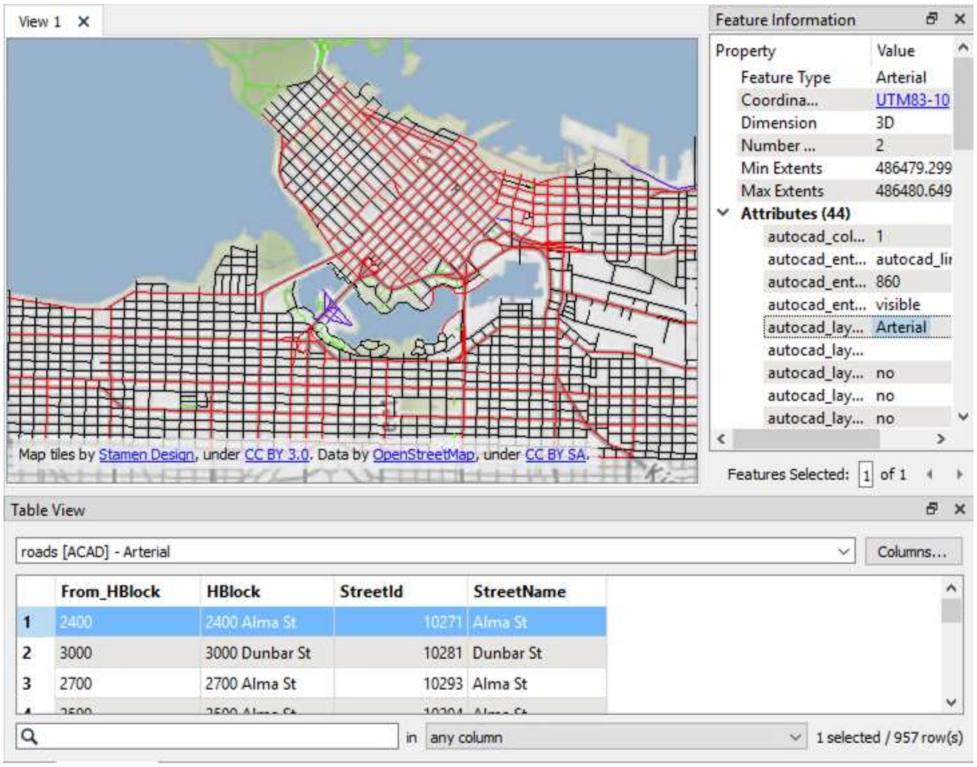
Requirements

The Esri Geodatabase (File Geodb) reader/writer used in the following example requires that a licensed version of ArcGIS be available to the user. For more information on required ArcGIS license levels, please see Required ArcGIS License Types for FME Geodatabase Formats (https://community.safe.com/s/article/required-arcgis-license-types-for-fme-geodatabase).

Video

Source Data

Roads (Autodesk AutoCAD DWG/DXF)



View 1
Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under CC BY SA.
Feature Information

Property	Value
Feature Type	Arterial
Coordina...	UTM83-10
Dimension	3D
Number ...	2
Min Extents	486479.299
Max Extents	486480.649
Attributes (44)	
autocad_col...	1
autocad_ent...	autocad_li
autocad_ent...	860
autocad_ent...	visible
autocad_lay...	Arterial
autocad_lay...	
autocad_lay...	no
autocad_lay...	no
autocad_lay...	no

Features Selected: 1 of 1

Table View

roads [ACAD] - Arterial

	From_HBlock	HBlock	StreetId	StreetName
1	2400	2400 Alma St	10271	Alma St
2	3000	3000 Dunbar St	10281	Dunbar St
3	2700	2700 Alma St	10293	Alma St
4	2500	2500 Alma St	10304	Alma St

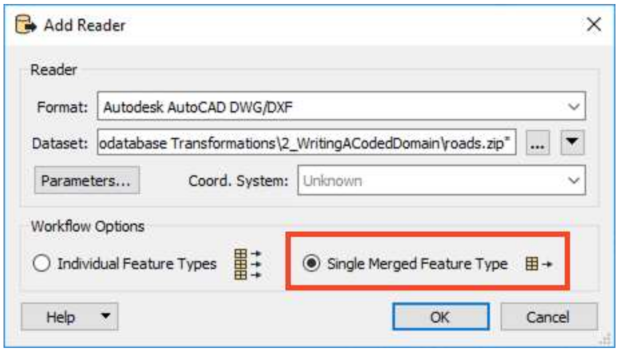
in any column 1 selected / 957 row(s)

In the above image, we see a visualization of the source Roads AutoCAD DWG. The values that will be subject to the destination geodatabase domain definition can be seen in the autocad_layer format attribute.

Step-by-step Instructions

1. Read Source Data

In a blank workspace, add an Autodesk AutoCAD DWG/DXF writer to the canvas, browse to the Road.zip dataset that is provided in the Files section of this tutorial, you can leave the file zipped. Change the Workflow Options to Single Merged Feature Type. Then open the parameters.



Add Reader

Reader

Format: Autodesk AutoCAD DWG/DXF

Dataset: odatabase Transformations\2_WritingACodedDomain\roads.zip"

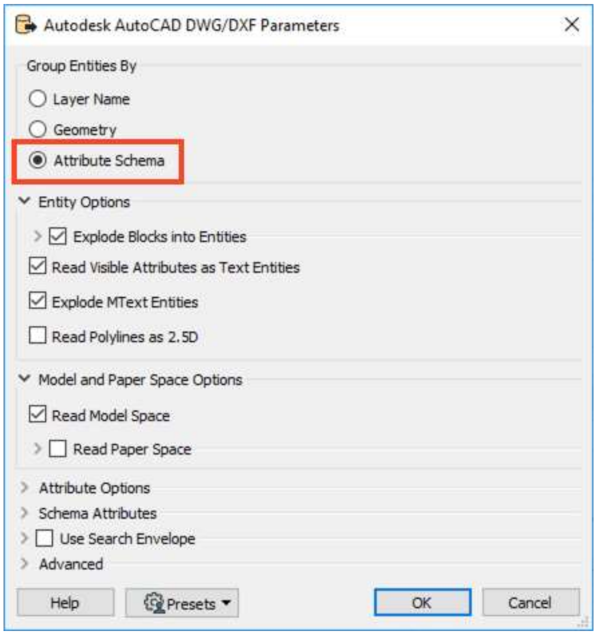
Coord. System: Unknown

Workflow Options

○ Individual Feature Types

● Single Merged Feature Type

In the parameters, set the Group Entities By to Attribute Schema, then click OK twice to add the reader.



Autodesk AutoCAD DWG/DXF Parameters

Group Entities By

○ Layer Name
○ Geometry
● Attribute Schema

Entity Options

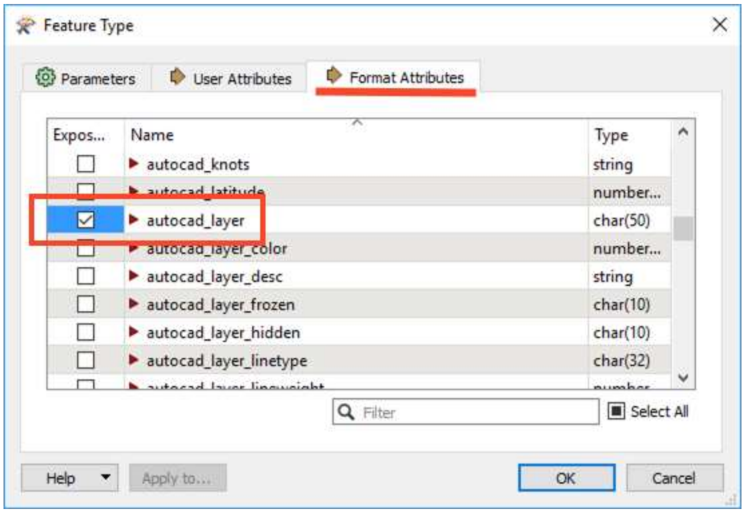
☑ Explode Blocks into Entities
☑ Read Visible Attributes as Text Entities
☑ Explode MText Entities
□ Read Polylines as 2.5D

Model and Paper Space Options

☑ Read Model Space
□ Read Paper Space

Attribute Options
Schema Attributes
□ Use Search Envelope
Advanced

Once the reader has been added to the workspace canvas, double-click on the reader feature type to open the parameters. Switch to the Format Attributes tab and enable the autocad_layer, then click OK.



Feature Type

Expos...	Name	Type
□	autocad_knots	string
□	autocad_latitude	number...
☑	autocad_layer	char(50)
□	autocad_layer_color	number...
□	autocad_layer_desc	string
□	autocad_layer_frozen	char(10)
□	autocad_layer_hidden	char(10)
□	autocad_layer_linetype	char(32)
□	autocad_layer_lineweight	number...

2. Map Attributes

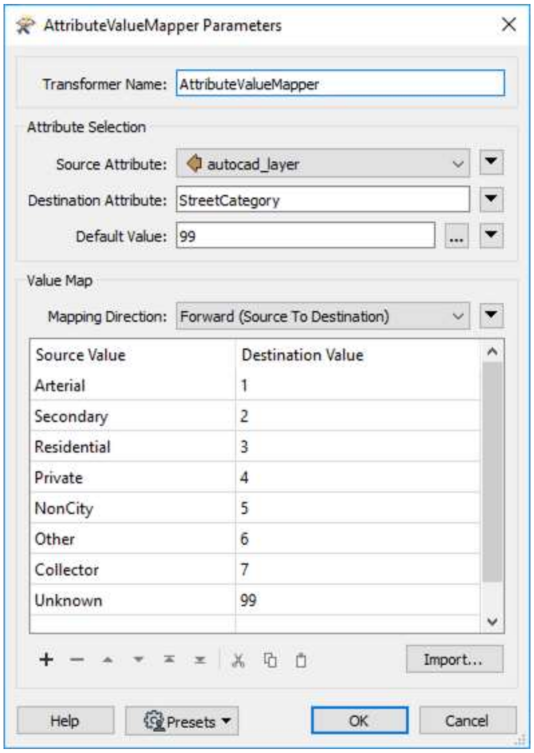
There are two different ways to map attributes to domains, either by domain code or by resolved domain values. If following along with the demo, only use one method.

Option 1: Map Attributes to Domain Codes

Next, we will use an AttributevalueMapper to map the incoming values from the source DWG to the domain codes which we will apply to the domain definition. Add an AttributeValueMapper to the canvas and connect it to the <ALL> reader feature type. In the parameters, set the Source Attribute to autocad_layer and enter StreetCategory as the Destination Attribute. Enter 99 as the Default Value as this will be the domain code we will use for Unknown street types.
In the Value Mapping part of the parameters dialog, enter the values as seen in the table.

Source Value	Destination Value
Arterial	1
Secondary	2

Source Value	Destination Value
Residential	3
Private	4
NonCity	5
Other	6
Collector	7
Unknown	99



Transformer Name: AttributeValueMapper

Source Attribute: autocad_layer

Destination Attribute: StreetCategory

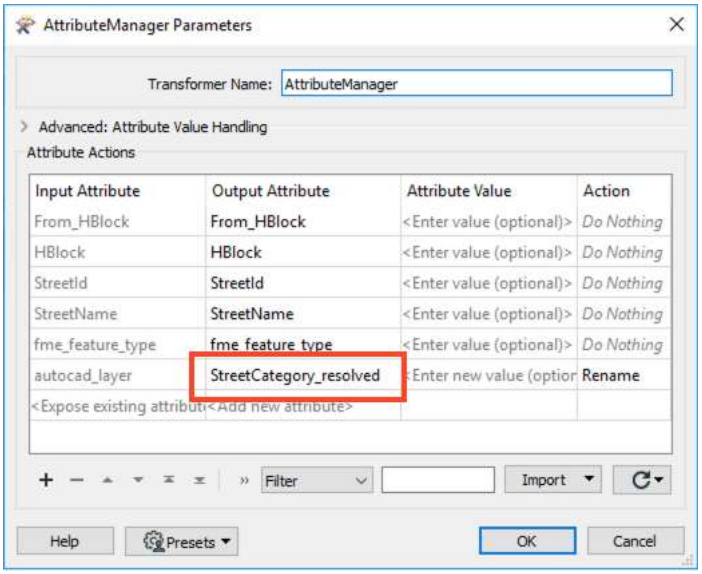
Default Value: 99

Mapping Direction: Forward (Source To Destination)

Source Value	Destination Value
Arterial	1
Secondary	2
Residential	3
Private	4
NonCity	5
Other	6
Collector	7
Unknown	99

Option 2: Map Attributes to Resolved Domain Values

As an alternative to the step described above, one could instead use an AttributeManager, to take the value from the source DWG file and map it to the domain definition based on the domain definitions resolved value. To do so, add an AttributeManager to the workspace and open the parameters dialog. Rename autocad_layer to StreetCategory_resolved - the resolved version of the domain.



AttributeManager Parameters

Transformer Name: AttributeManager

Advanced: Attribute Value Handling

Attribute Actions

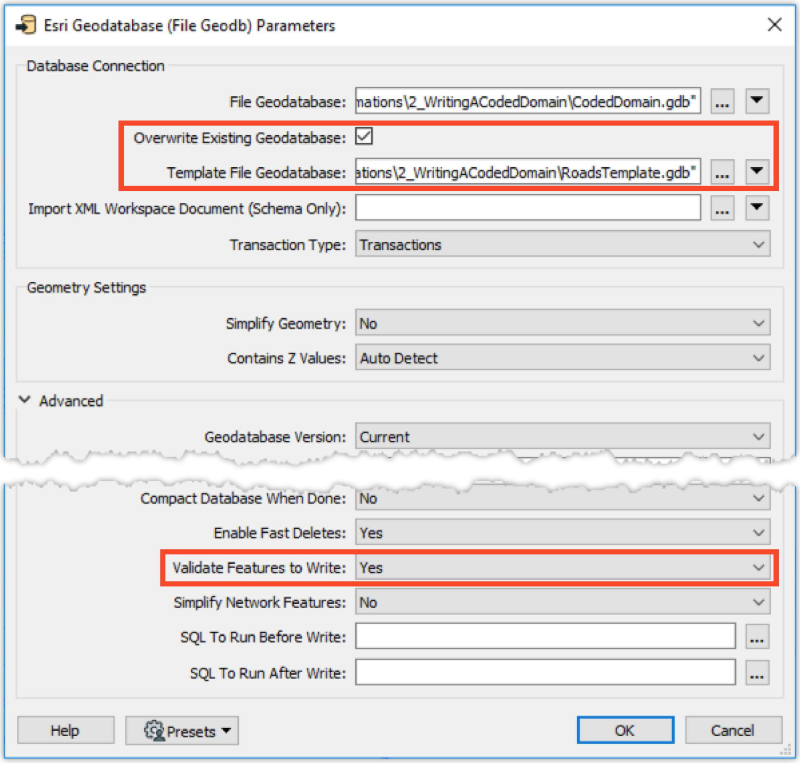
Input Attribute	Output Attribute	Attribute Value	Action
From_HBlock	From_HBlock	<Enter value (optional)>	Do Nothing
HBlock	HBlock	<Enter value (optional)>	Do Nothing
StreetId	StreetId	<Enter value (optional)>	Do Nothing
StreetName	StreetName	<Enter value (optional)>	Do Nothing
fme_feature_type	fme_feature_type	<Enter value (optional)>	Do Nothing
autocad_layer	StreetCategory_resolved	<Enter new value (optior	Rename
<Expose existing attribut	<Add new attribute>		

The above tends to be most applicable when your workflow falls under Scenario #1 (writing to an existing table using an existing domain definition) or Scenario #2 (writing to a new table in the geodatabase using an existing domain). In both scenarios, it would be assumed that one does not know the actual domain codes contained in the domain definition and instead know the resolved value.

3. Write Features and Create Domain

Add an Esri Geodatabase (File Geodb) writer to the canvas, browse to a location to save the geodatabase. Leave the Feature Class or Table Definition set to Copy from Reader then open the parameters.

In the parameters, enable Overwrite Existing Geodatabase, then browse to the RoadsTemplate.gdb for the Template File Geodatabase. Next, expand the Advanced section and set Validate Features to Write to Yes. This setting will ensure that the incoming source data adheres to our domain definition. Then click OK twice to add the writer.



Esri Geodatabase (File Geodb) Parameters

Database Connection

File Geodatabase: nations\2_WritingACodedDomain\CodedDomain.gdb"

Overwrite Existing Geodatabase: ☑

Template File Geodatabase: ations\2_WritingACodedDomain\RoadsTemplate.gdb"

Import XML Workspace Document (Schema Only):

Transaction Type: Transactions

Geometry Settings

Simplify Geometry: No

Contains Z Values: Auto Detect

Advanced

Geodatabase Version: Current

Compact Database When Done: No

Enable Fast Deletes: Yes

Validate Features to Write: Yes

Simplify Network Features: No

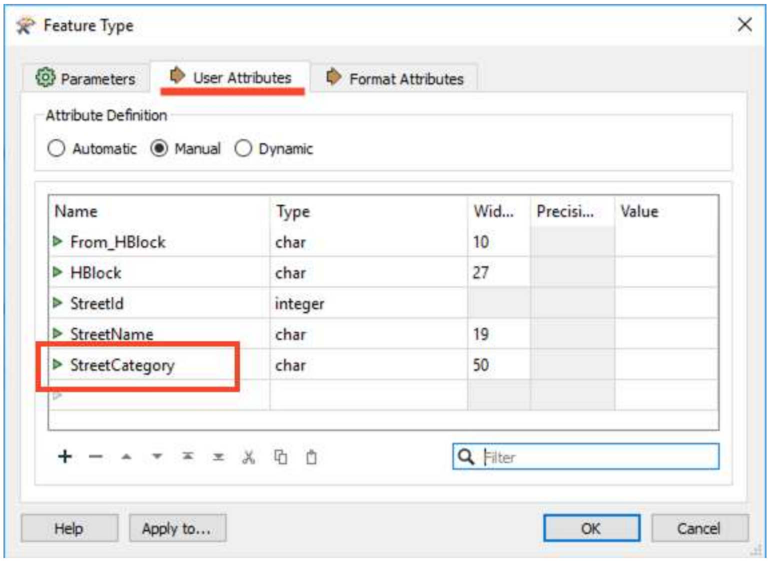
SQL To Run Before Write:

SQL To Run After Write:

Connect the writer to either the AttributeValueMapper or the AttributeManager, then open the writer feature type parameters.

In the feature type parameters, change the Feature Class or Table Name to something more meaningful such as Streets.

Next, switch to the Attributes tab and rename autocad_layer to StreetCategory and click OK.



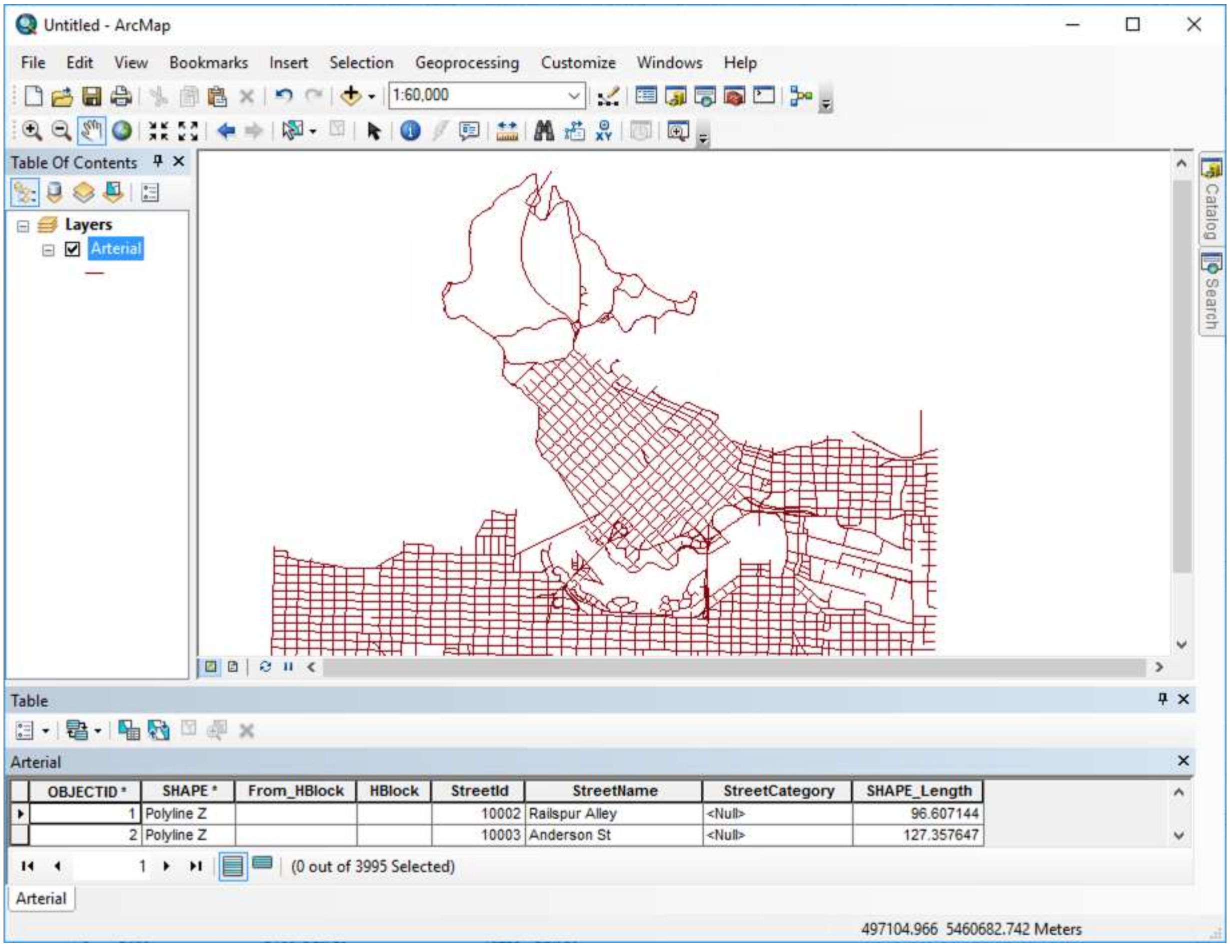
Attribute Definition

○ Automatic ● Manual ○ Dynamic

Name	Type	Wid...	Precisi...	Value
From_HBlock	char	10		
HBlock	char	27		
StreetId	integer			
StreetName	char	19		
StreetCategory	char	50		

4. Save and Run the Workspace

Save the workspace and then run it. View your output in ArcMap.

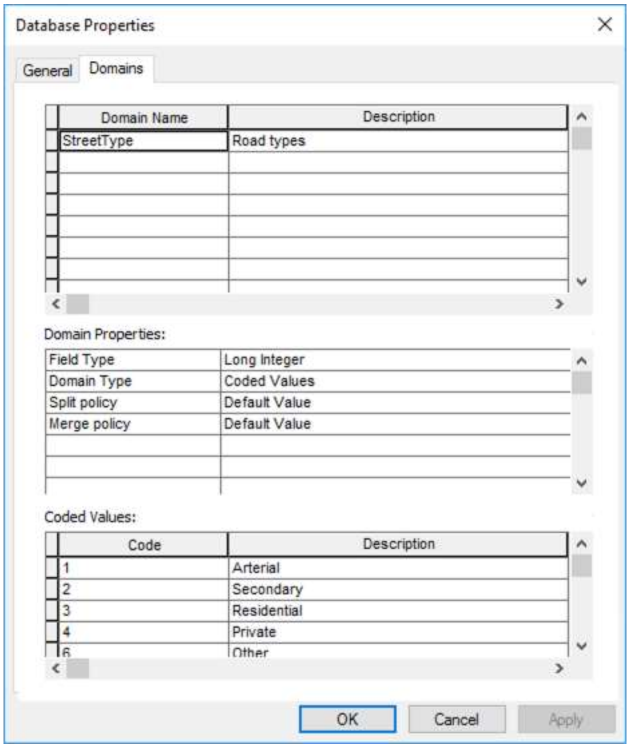


Arterial

OBJECTID *	SHAPE *	From_HBlock	HBlock	StreetId	StreetName	StreetCategory	SHAPE_Length
1	Polyline Z			10002	Railspur Alley	<Null>	96.607144
2	Polyline Z			10003	Anderson St	<Null>	127.357647

(0 out of 3995 Selected)

497104.966 5460682.742 Meters



Domain Name	Description
StreetType	Road types

Domain Properties:

Field Type	Long Integer
Domain Type	Coded Values
Split policy	Default Value
Merge policy	Default Value

Coded Values:

Code	Description
1	Arterial
2	Secondary
3	Residential
4	Private
6	Other

Data Attribution

The data used here originates from data made available by the City of Vancouver, British Columbia (https://opendata.vancouver.ca/pages/licence/). It contains information licensed under the Open Government License - Vancouver.

Continue to the Next Article: Working with Geodatabase Subtypes: Writing a Subtype (https://community.safe.com/s/article/working-with-geodatabase-subtypes)

First Published Date
7/29/2020, 12:15 AM

Last Published Date
6/9/2022, 10:24 PM

Transformation (/s/topic/0TO4Q000000...

FME Desktop (/s/topic/0TO4Q000000...

Sort by:

Latest Posts

Nothing here yet?

Log in to post to this feed.

Follow

Files (2) (/s/relatedlist/ka14Q000000wn4yQAA/AttachedContentDocuments)

ZIP WritingACodedDomain_2022
Jun 9, 2022 • 1.1MB • zip

ZIP WritingACodedDomain_2021.1
Jun 9, 2022 • 1MB • zip

View All (/s/relatedlist/ka14Q000000wn4yQAA/AttachedContentDocuments)

Related Articles

Extract Geodatabase Coded Domain Definitions (/s/article/extract-geodatabase-coded-domain-definitions)

Creating and Associating Geodatabase Domains Dynamically Using A Feature Class and GDB Table (/s/article/creating-and-associating-geodatabase-domains-dynam)

Working with Geodatabase Subtypes: Writing A Subtype (/s/article/working-with-geodatabase-subtypes)

Working with Geodatabase Subtypes and Domains (/s/article/working-with-geodatabase-subtypes-and-domains)

Working with Geodatabase Metadata: Writing/Updating Metadata (/s/article/working-with-geodatabase-metadata-writing-to-xml)

Getting Started Ideas Feedback

Land Acknowledgement —

Safe Software respectfully acknowledges that we live, learn and work on the traditional and unceded territories of the Kwantlen, Katzie, and Semiahmoo First Nations.