Article

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

GML Writing

Aug 3, 2022 • Knowledge

Product Type
FME Desktop

FME Version
2022.0

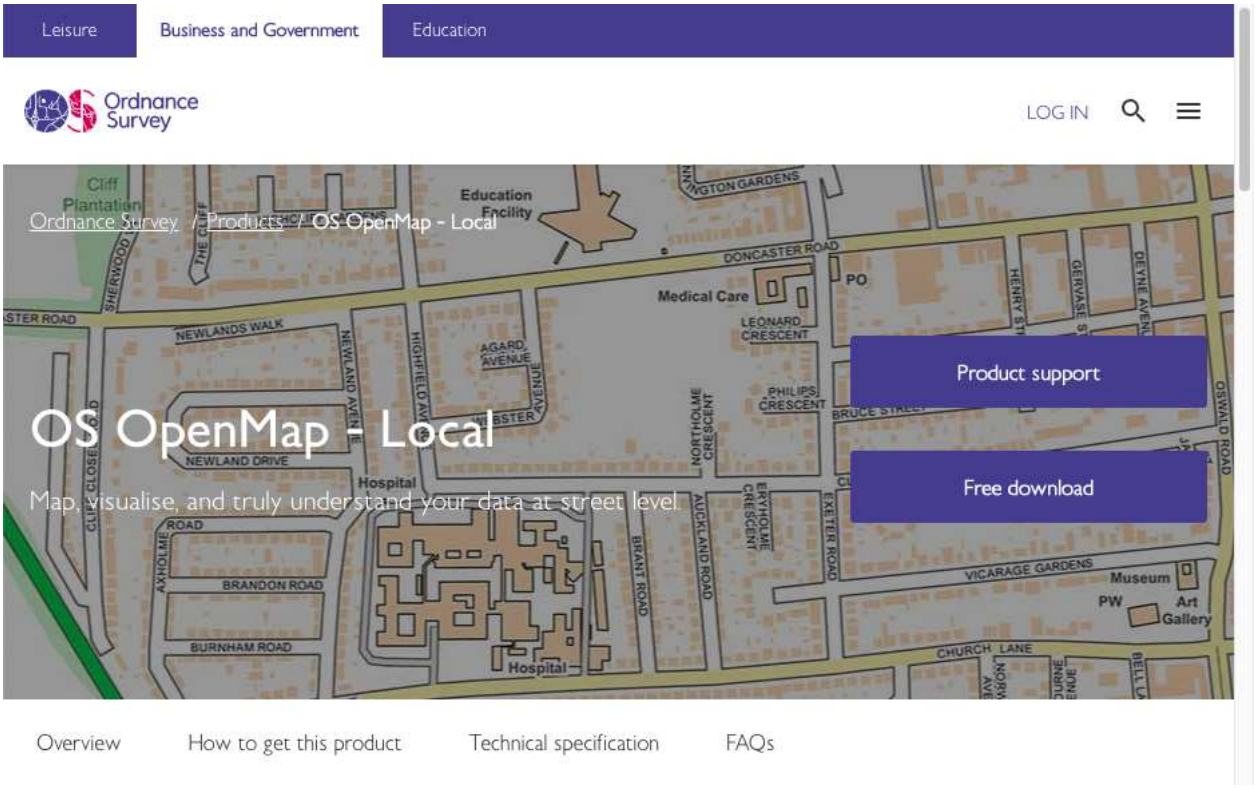
Tutorial: Tutorial: Getting Started with XML (/s/article/tutorial-getting-started-with-xml) | **Previous:** GML Reading (https://community.safe.com/s/article/reading-xmlgml)

Introduction

The overall objective of this GML tutorial is to demonstrate a complete example workflow for reading from GML and loading it into a database, which we did in the previous tutorial (https://community.safe.com/s/article/reading-xmlgml). Now in this tutorial, we'll read the database and then write back to GML, with full validation. This is a common workflow required for those using GML as an exchange format. In some cases, GML downloaded from a geoportal needs to be loaded into a local database, such as the basemap data used in this exercise. In other cases, tables from a local database need to be exported to GML to be shared with other users or uploaded to a geoportal. For these exercises, the database employed is OGC Geopackage.

In general, it's often beneficial to use a standardized GML profile described by a well-defined application schema. This allows the data to be validated at each step to ensure quality and completeness. The data and schemas employed for this exercise are a comprehensive base map dataset for the Gloucester area from UK Ordnance Survey OpenMap Local GML, downloaded from the UK OS geoportal. This dataset and others like it for other areas of the UK can be downloaded

by tile in GML from the UK OS Open Map Geoportal (https://www.ordnancesurvey.co.uk/business-government/products/open-map-local).



Screenshot from the UK OS Open Map Geoportal (https://www.ordnancesurvey.co.uk/business-government/products/open-map-local) home page.

Requirements

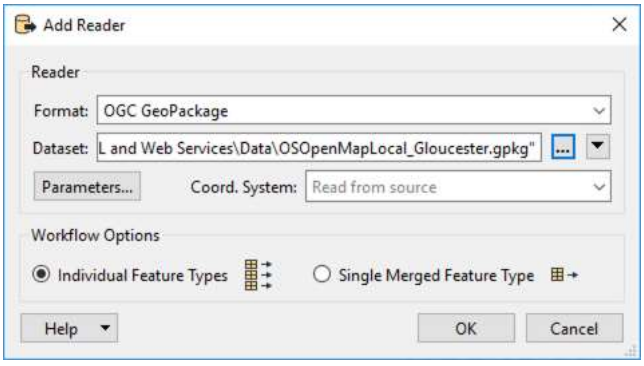
- XML Editor - Such as Notepad++ (https://notepad-plus-plus.org/downloads/) with the XML Tools plug-in (https://community.notepad-plus-plus.org/topic/16983/xml-tools) .
  - This allows the inspection of the actual source XML / GML documents and FME writers' xml/gml outputs. This can reveal problems in the data that may not be visible in FME Data Inspector or other XML/GML viewing software.

Step-by-Step Instructions

In the following exercise, we will see how to write GML using an Application Schema with data coming from a Geopackage.

1. Open FME Workbench and Add an OGC GeoPackage Reader

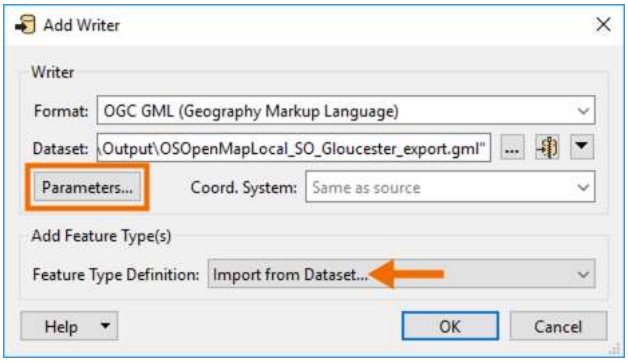
Open FME Workbench and create a blank workspace. Add an OGC GeoPackage reader to the canvas and browse to the OSOpenMapLocal_Gloucester.gpkg file located in the Data folder, or you can use the output dataset from the previous tutorial.



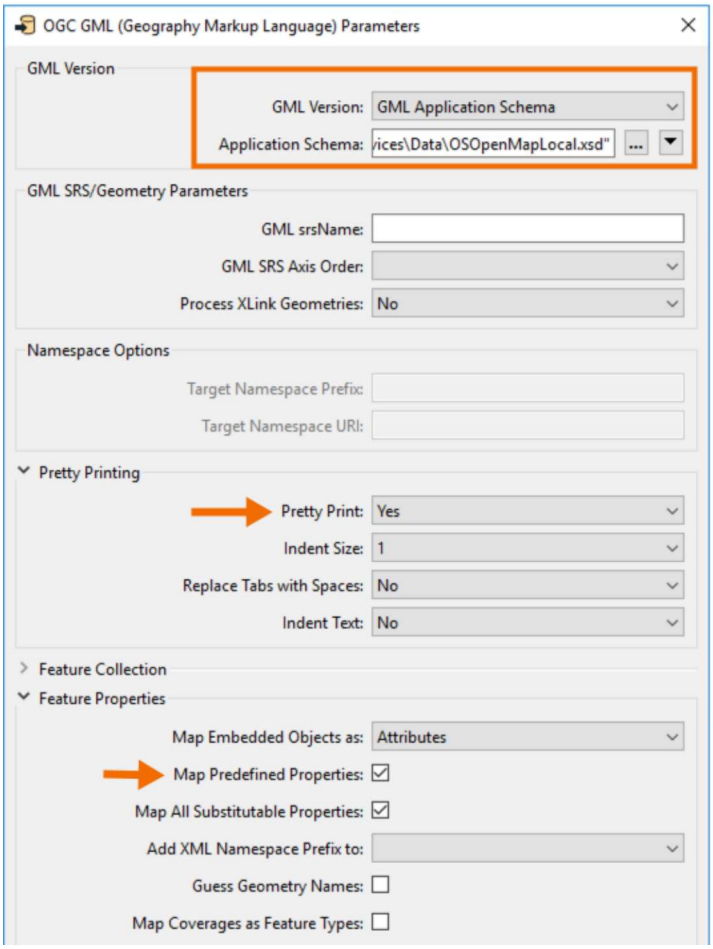
Click OK to add the reader, and in the Select Feature Types dialog, select all.

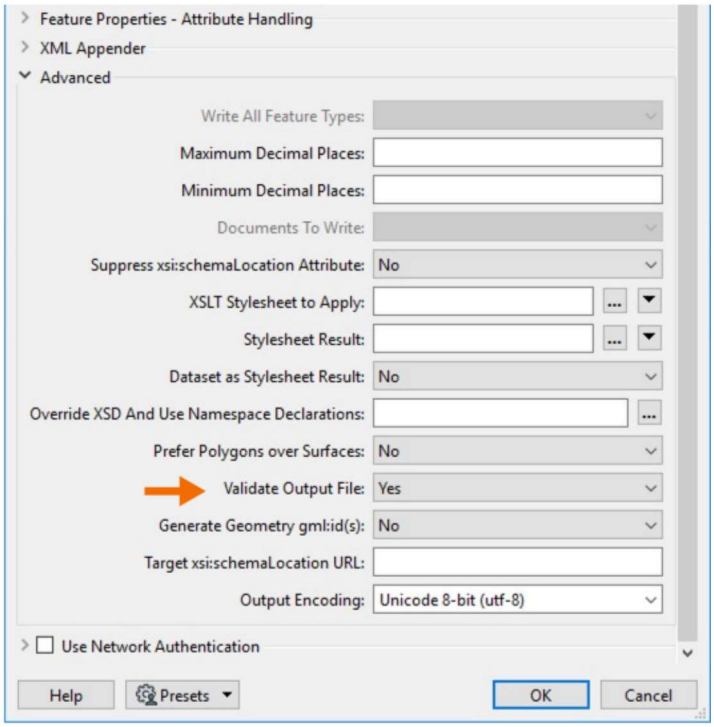
2. Add a GML Writer

Add a GML Writer. Browse to an Output folder and name the file OSOpenMapLocal_SO_Gloucester_export.gml. Change the Feature Type Definition to Import from Dataset, then open the Parameters.

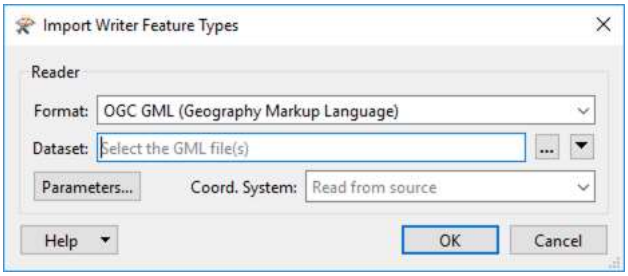


In the GML writer parameters, set the GML Version to GML Application Schema, then set the Application Schema to the OSOpenMapLocal.xsd dataset. Next, expand Pretty Printing and set Pretty Print to Yes. Then expand Feature Properties and enable Map Predefined Properties. Finally, expand Advanced, and set Validate Output File to Yes. Click OK twice.





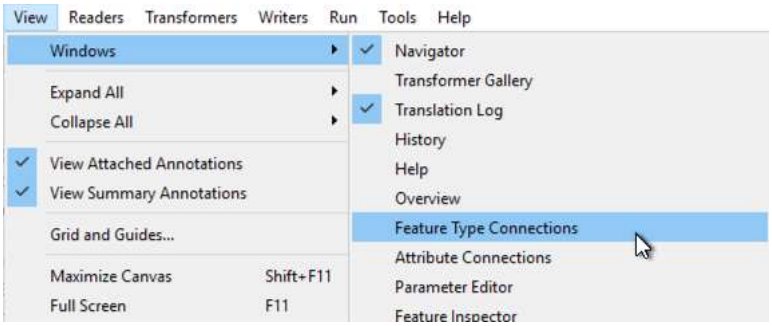
In the Import Writer Feature Types dialog, keep the Format set to OGC GML, but delete the Dataset path; this should now be blank. Open the parameters and confirm that the Application Schema and Map Predefined Properties are the same as what was set in the writer. Click OK twice to finish importing the schema.



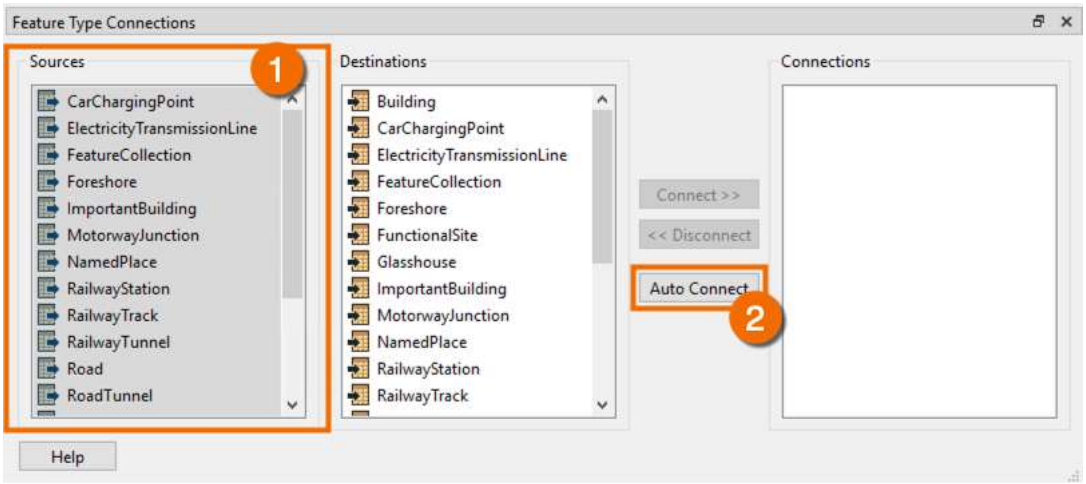
Finally, in the Select Feature Types dialog, Select All.

3. Connect Reader Feature Types

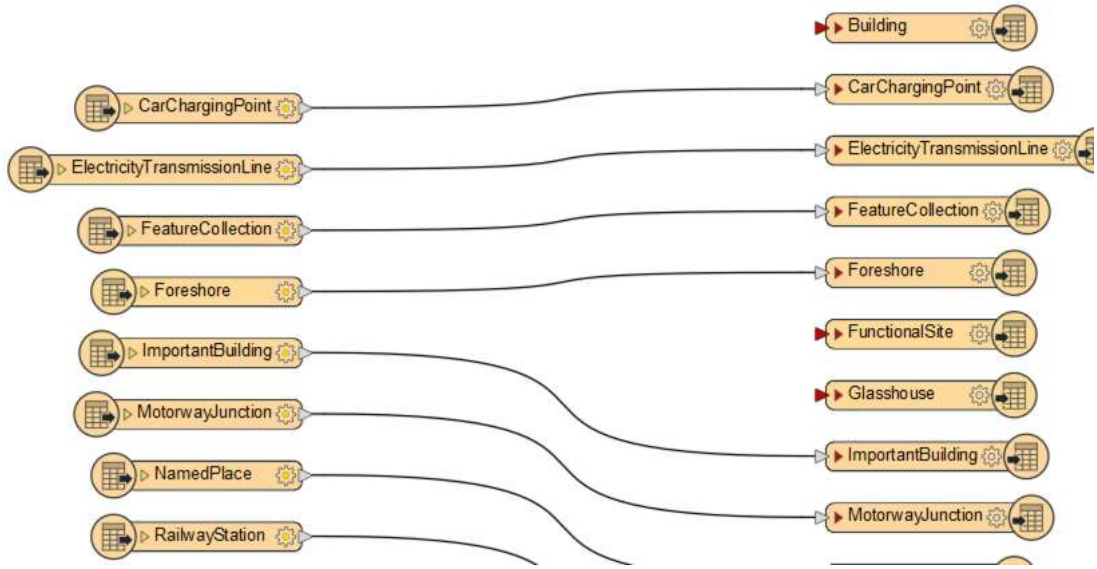
Now we will use the Feature Connection Window to connect all of the reader feature types to their matching writer feature types. You can find the Feature Connection Window by going to View > Windows on the top menu bar.



In the Feature Connection Window, click and drag all of the Sources to select them all then click Auto Connect.

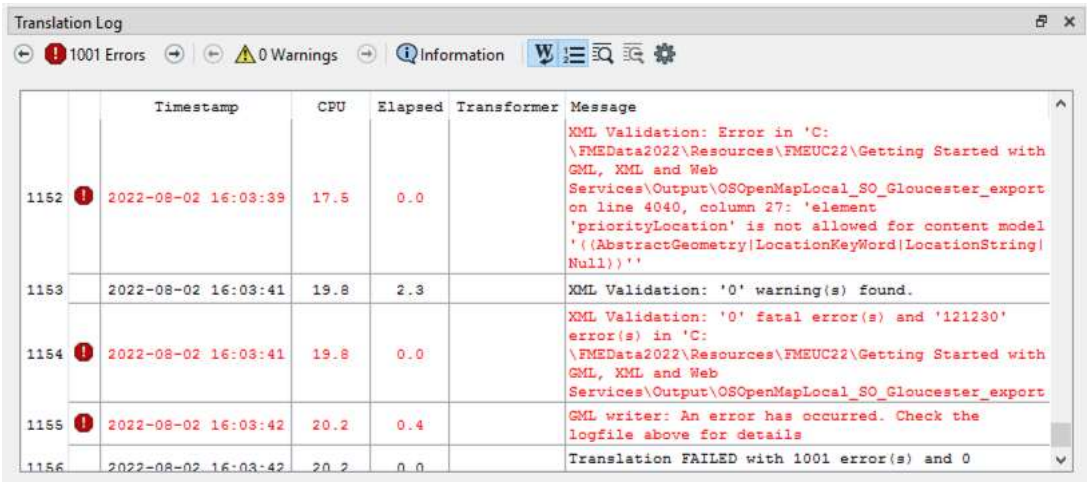


There will be four writer feature types without a connection; Building, FunctionalSite, Glasshouse, and Transaction. Move them to the side to clean up the workspace or remove them.



4. Run and Review Translation Log

Once the source feature types are connected to their destinations, click Run > Run workspace. Review the Translation Log and observe any errors. Pay special attention to comments about what is missing and what is not allowed.



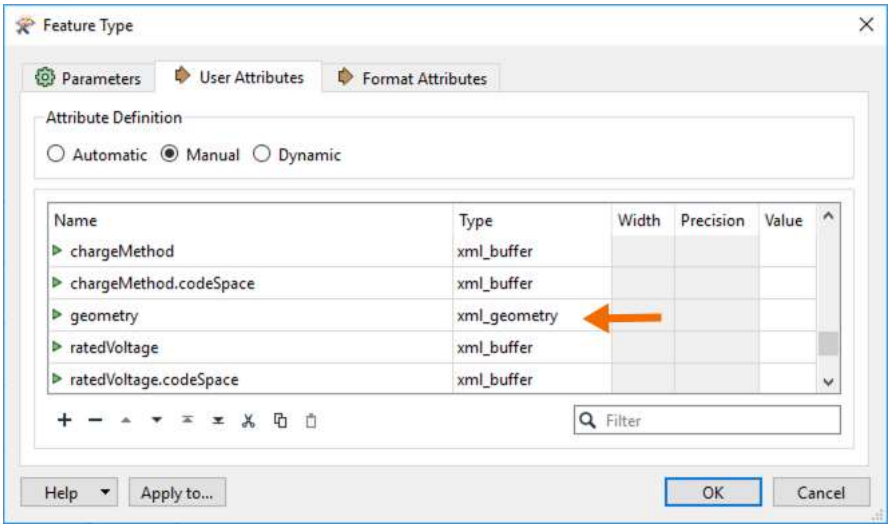
5. Review Results

Open the Output in an XML Editor like Notepad++ and note what is present or missing. In particular notice the lack of geometry in the GML Output:

```
<os:metadata xlink:href="http://www.os.uk/xml/products/OML.xml"/>
<os:featureMember>
    <oml:CarChargingPoint gml:id="id9DD975F5-27BC-4E44-94DB-CE8FBF2EF87C">
        <gml:description/>
        <gml:identifier/>
        <gml:boundedBy/>
        <gml:priorityLocation/>
        <oml:featureCode>15624</oml:featureCode>
        <oml:chargeMethod codeSpace="http://www.os.uk/xml/codelists/map/ChargeMethodOML.xml">Three Phase AC</oml:chargeMethod>
        <oml:geometry/>
        <oml:ratedVoltage codeSpace="http://www.os.uk/xml/codelists/map/RatedVoltageOML.xml">400</oml:ratedVoltage>
        <oml:type codeSpace="http://www.os.uk/xml/codelists/map/CarChargingTypeOML.xml">Type 2 Mennekes (IEC 62196)</oml:type>
    </oml:CarChargingPoint>
</os:featureMember>
```

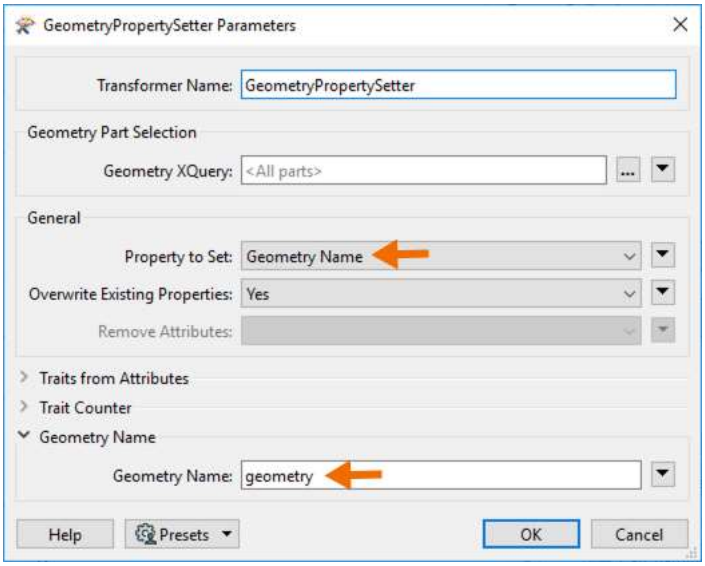
6. Setting the Geometry

The reason the geometries are not appearing is that we have to set the geometry name according to what is required by the destination schema. Our main clue as to what to name the geometry shows up in the destination feature type as the properties with the xml_geometry data type.

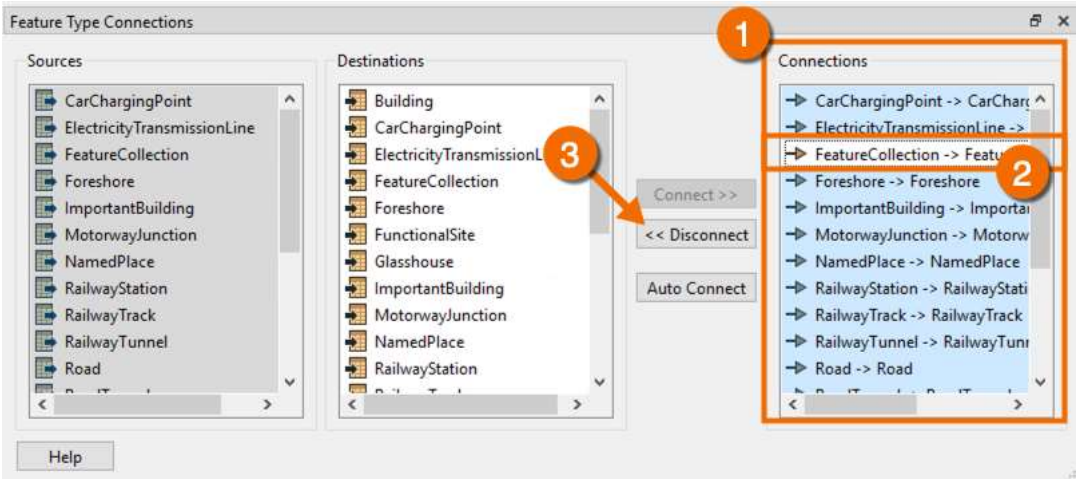


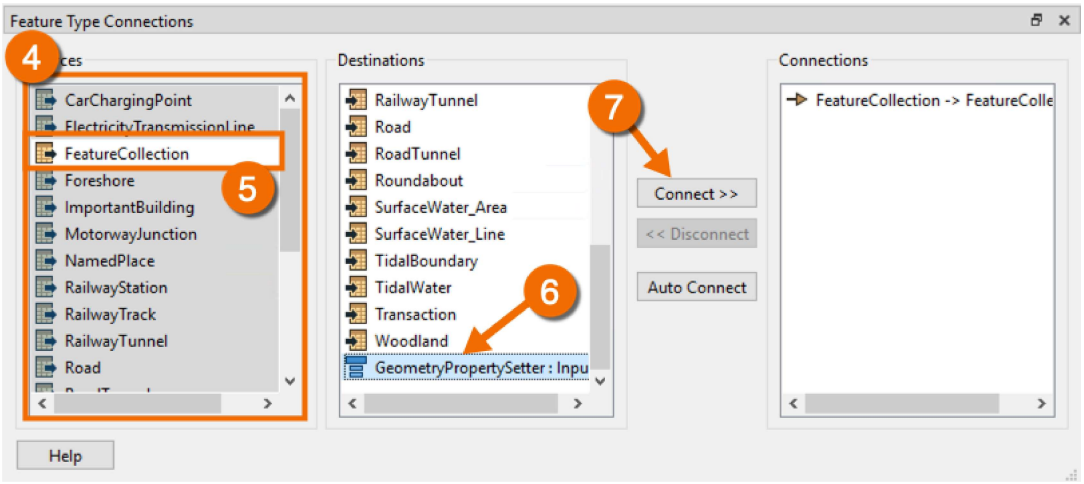
For example, most of the destination feature types store their geometry in the 'geometry' property, which shown above, has data type = xml_geometry.

Add a GeometryPropertySetter to the workspace. Set the Property to Set to Geometry Name, then set the Geometry Name to geometry. Be sure to set up the GeometryPropertySetter prior to connecting it, otherwise, it will pick up the geometry attribute name from the source. The name needs to be geometry and not the geometry value.

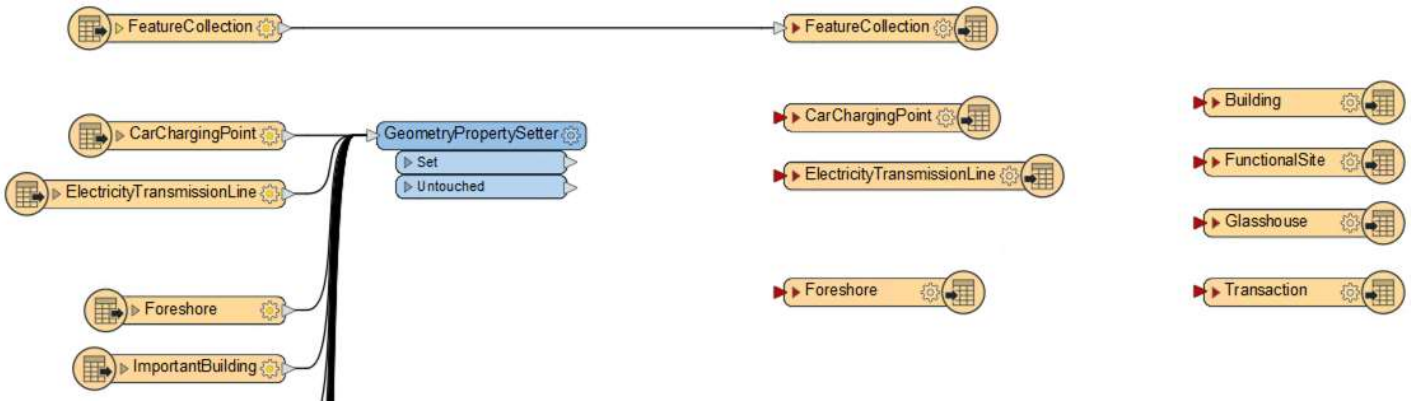


Using the Feature Type Connections window, disconnect all of the connections except for FeatureCollection. You can de-select FeatureCollection by holding the ctrl/cmd key and clicking on it. Then reconnect them (except FeatureCollection) to the GeometryPropertySetter, which can be found at the bottom of the Destinations list.



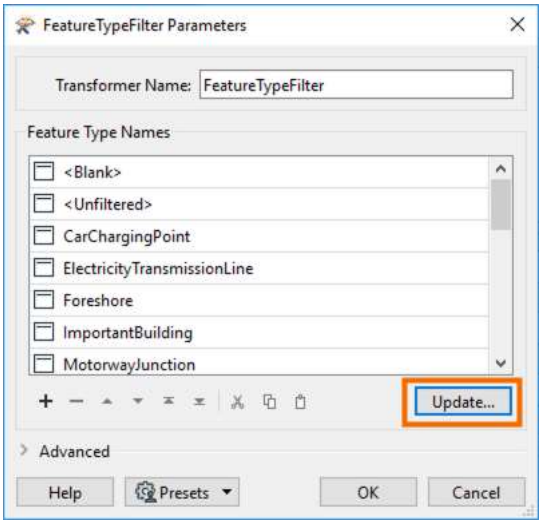


After connecting the GeometryPropertySetter, move the FeatureCollection reader/writer feature type connection to the top of the workspace.



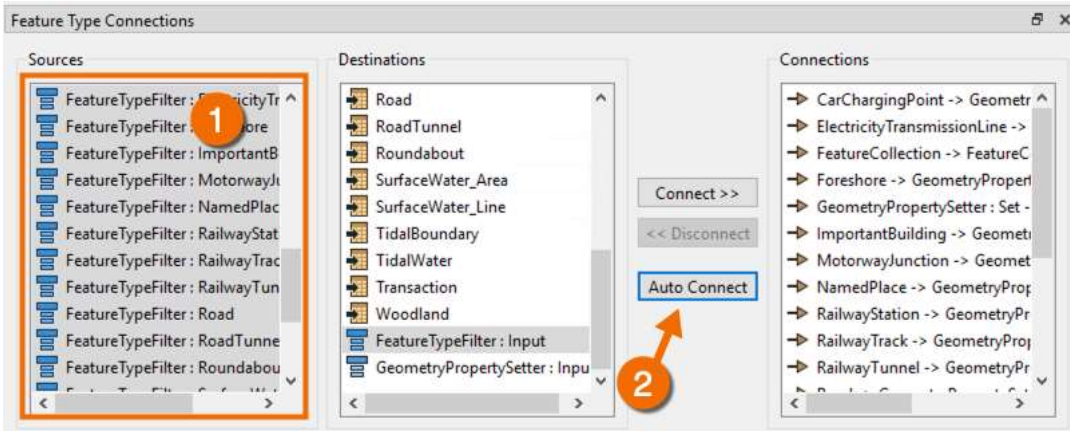
7. Filter Feature Types

Add a FeatureTypeFilter to the canvas and connect it to the Set output port on the GeometryPropertySetter. In the parameters, select Update to import the Feature Type Names.



8. Connect FeatureTypeFilter Outputs

Using the Feature Type Connections window, select all of the FeatureTypeFilter sources, then click Auto Connect. This will automatically connect the FeatureTypeFilter output ports to their corresponding writer feature types.



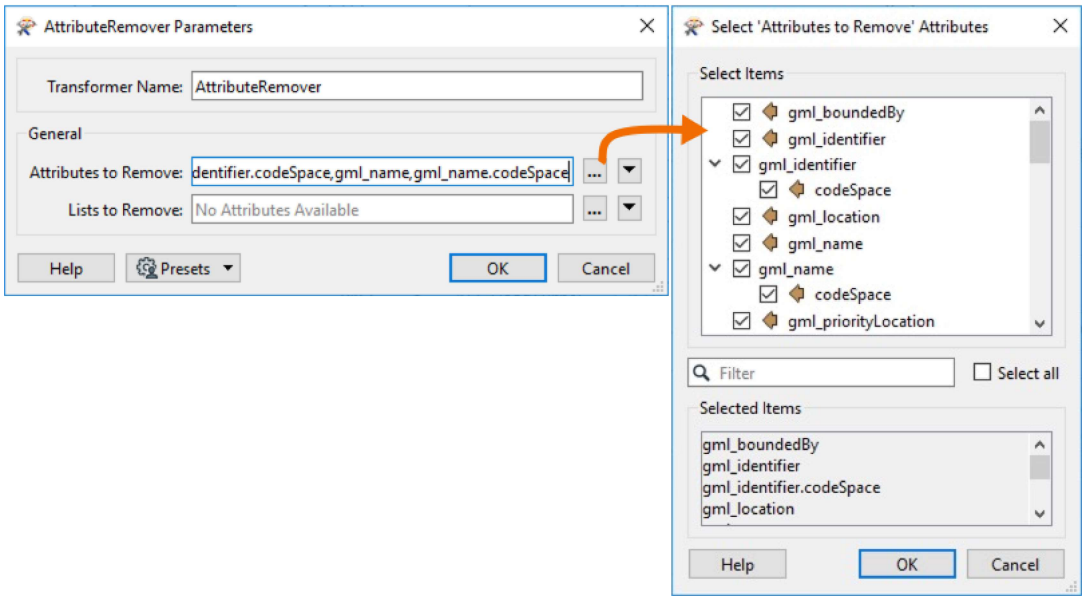
9. Run Workspace

Rerun the workspace, review any errors in the log file, and re-check the data using an XML editor like Notepad++ again.

10. Cleanup Attributes

Note that while we now should be seeing geometry in the output, we are still getting a number of errors due to unexpected properties. To correct this, add an AttributeRemover between the GeometryPropertySetter and the FeatureTypeFilter. Copy and paste the following comma-delimited list into the Attributes to Remove field:

```
gml_boundedBy,gml_priorityLocation,gml_location,gml_identifier,gml_identifier.codeSpace,gml_name,gml_name.codeSpace
```

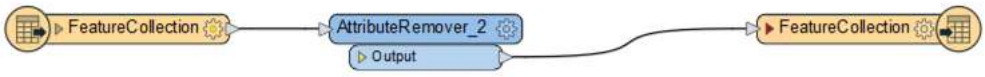


11. Run the Workspace

Run Workspace again, review the log for any errors, and review the output in the Text Editor. Try reading the output GML file into FME Data inspector with the Application Schema set to OSOpenMapLocal.xsd.

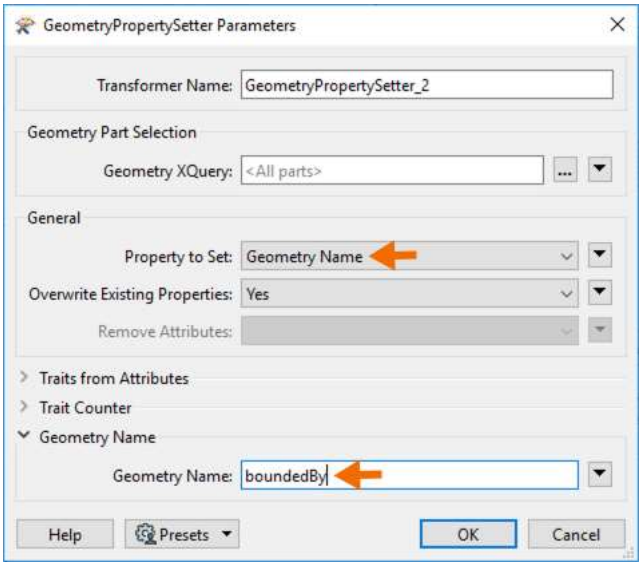
12. Feature Collection Attributes

Copy the AttributeRemover used between the GeometryPropertySetter and the FeatureTypeFilter. Paste this and connect the FeatureCollection Reader. Uncheck the gml_boundedBy attribute since we need that for the dataset Bounding Box.



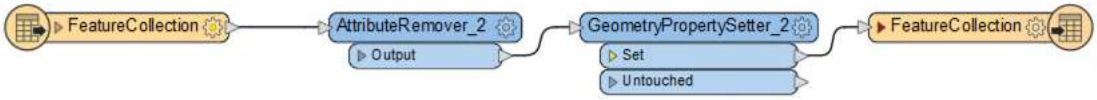
13. Feature Collection Geometry

Add a GeometryPropertySetter to the canvas but don't connect it yet. In the parameters, set the Property to Set to Geometry Name, followed by the Geometry Name to boundedBy. Again make sure this value is literal and not pointing to the attribute value.



Once the parameters are set, connect the GeometryPropertySetter_2 between the AttributeKeeper_2 and the FeatureCollection writer feature type. The writer

feature type should be connected to the Set output port.

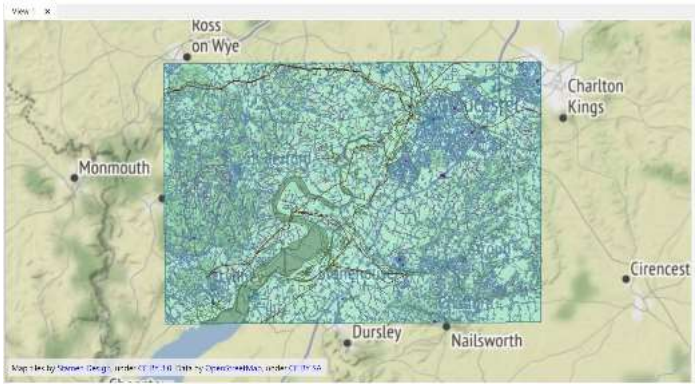


14. Run Workspace

Rerun and observe any errors, or warnings.

15. View Output

View output in Notepad++ and in FME Data Inspector. Compare the views in FME Data Inspector depending on whether or not you enabled map predefined properties.



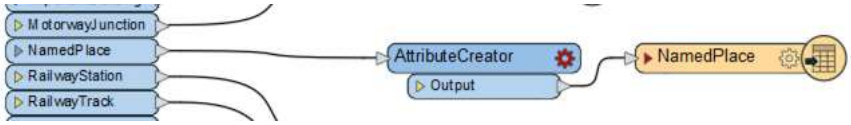
Map tiles by Stamen Design (http://stamen.com/), under CC-BY-3.0 (http://creativecommons.org/licenses/by/3.0). Data by OpenStreetMap (http://openstreetmap.org/), under CC-BY-SA (http://creativecommons.org/licenses/by-sa/3.0).

Congratulations, you have now successfully exported the Gloucester basemap data from your local GeoPackage database to a GML file in the UK OS OpenMap Local profile / format. This completes the GML tutorial in which we completed the round trip from reading the GML, importing it into a database, and then exporting it from the database back out to GML again. You can also compare your GML output to the original GML input from the GML Reading exercise to see if there are any substantive differences.

Optional

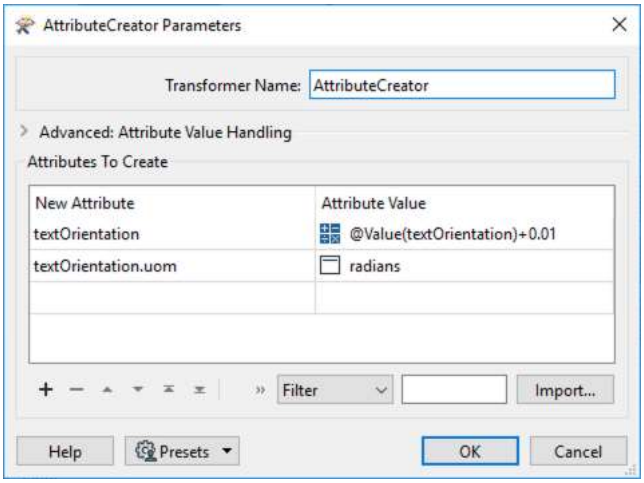
16. Optional Step: Modify GML Properties for NamedPlace

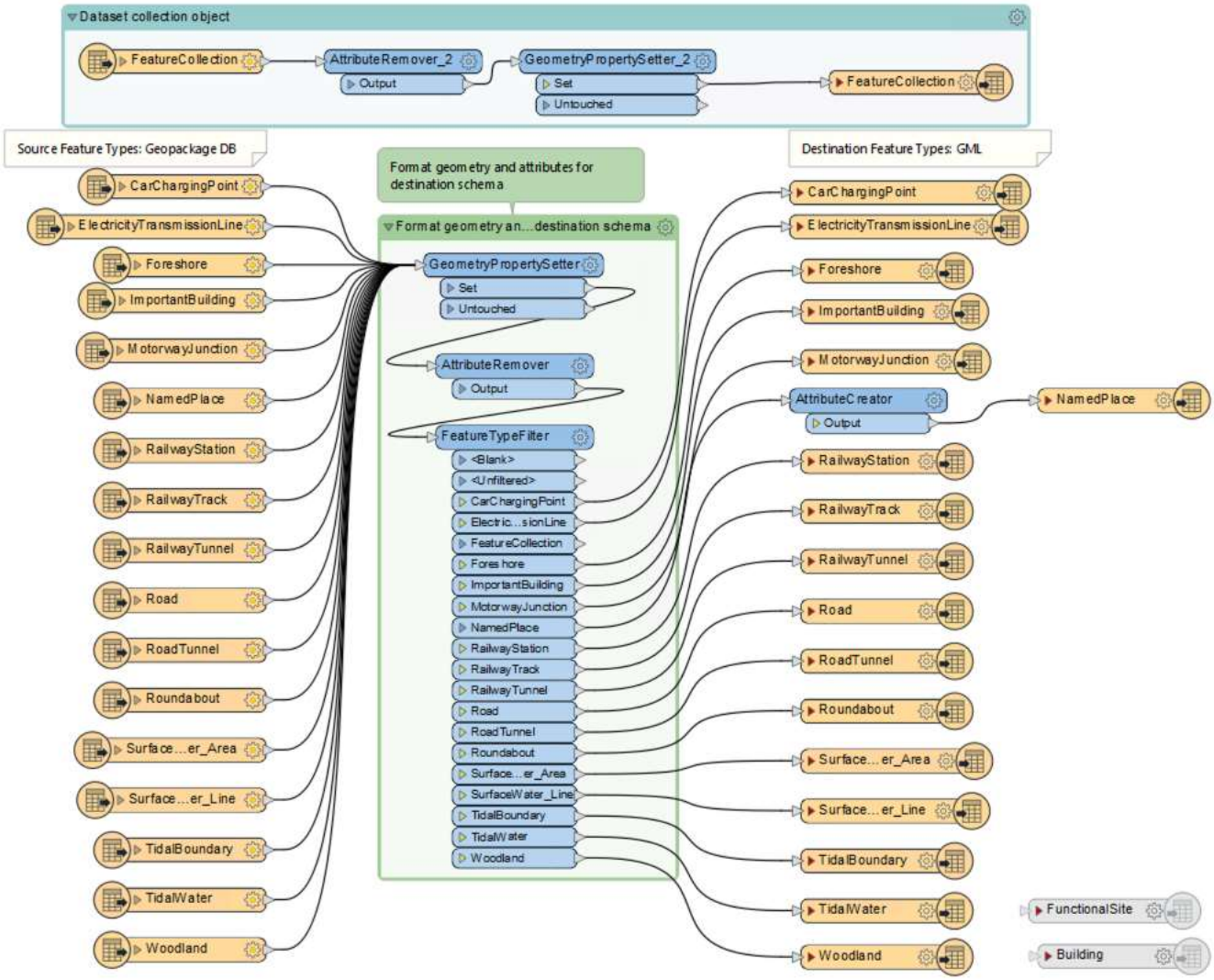
Connect an AttributeCreator between the FeatureTypeFilter NamedPlace output port and the NamedPlace writer feature type.



In the parameters, create the following attributes:

New Attribute	Attribute Value
textOrientation	@Evaluate(@Value(textOrientation)+0.01)
textOrientation.uom	radians


AttributeCreator Parameters
Transformer Name: AttributeCreator
Advanced: Attribute Value Handling
Attributes To Create
New Attribute
Attribute Value
textOrientation
@Value(textOrientation)+0.01
textOrientation.uom
radians
Filter
Import...
Help
Presets
OK
Cancel


Dataset collection object
FeatureCollection
AttributeRemover_2
Output
GeometryPropertySetter_2
Set
Untouched
FeatureCollection
Source Feature Types: Geopackage DB
Format geometry and attributes for destination schema
Destination Feature Types: GML
CarChargingPoint
ElectricityTransmissionLine
Foreshore
ImportantBuilding
MotorwayJunction
NamedPlace
RailwayStation
RailwayTrack
RailwayTunnel
Road
RoadTunnel
Roundabout
Surface...er_Area
Surface...er_Line
TidalBoundary
TidalWater
Woodland
Format geometry an...destination schema
GeometryPropertySetter
Set
Untouched
AttributeRemover
Output
FeatureTypeFilter
<Blank>
<Unfiltered>
CarChargingPoint
Electric...sionLine
FeatureCollection
Foreshore
ImportantBuilding
MotorwayJunction
NamedPlace
RailwayStation
RailwayTrack
RailwayTunnel
Road
RoadTunnel
Roundabout
Surface...er_Area
SurfaceWater_Line
TidalBoundary
TidalWater
Woodland
AttributeCreator
Output
FunctionalSite
Building

Rerun workspace and inspect output gml searching for NamedPlace elements in both your XML editor and in FME Data Inspector. Note how the textOrientation and nested uom properties have been updated.

```
<os:featureMember>
    <oml:NamedPlace gml:id="id2F045AB3-0E68-42EF-BF4E-8BAE2D318707">
        <gml:description/>
        <oml:distinctiveName>Shurdington</oml:distinctiveName>
        <oml:classification codeSpace="http://www.os.uk/xml/codelists/map/NamedPlaceClassificationOML.xml">Populated Place</oml:classification>
        <oml:fontHeight codeSpace="http://www.os.uk/xml/codelists/map/FontHeightClassificationOML.xml">Medium</oml:fontHeight>
        <oml:textOrientation uom="radians">0.01</oml:textOrientation>
        <oml:geometry>
            <gml:Point srsName="EPSG:27700" srsDimension="2">
                <gml:pos>392025 218557</gml:pos>
            </gml:Point>
        </oml:geometry>
        <oml:featureCode>15801</oml:featureCode>
    </oml:NamedPlace>
</os:featureMember>
```

Return to the tutorial start page: Tutorial: Getting Started with XML (/s/article/tutorial-getting-started-with-xml)

Data Attribution

The data here originates from the Ordnance Survey (OS) OpenMap Portal (https://www.ordnancesurvey.co.uk/business-government/products/open-map-local) and contains Contains OS data © Crown copyright and database right 2022. It contains information licensed under the Open Government License (http://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/) - United Kingdom.

First Published Date
7/29/2020, 12:09 AM

Last Published Date
8/3/2022, 9:50 PM

Formats and Applicatio... (/s/topic/0TO4Q000000...) FME Desktop (/s/topic/0TO4Q000000...)

Sort by:

Latest Posts

LizAtSafe (/s/profile/0050c00000CeGV0AAN) (Employee) published a new version of this Knowledge.
August 3, 2022 at 9:50 PM (/s/feed/0D54Q00009ggOnaSAE)

Like Comment

Log In to Comment

Follow

Files (1) (/s/relatedlist/ka14Q000001DWxZQAW/AttachedContentDocuments)

3b. Writing GML
Aug 3, 2022 • 23.1MB • zip

View All

(/s/relatedlist/ka14Q000001DWxZQAW/AttachedContentDocuments)

Related Articles

INSPIRE GML Basic Writing and Validation Example (/s/article/inspire-gml-basic-writing-example)

INSPIRE GML Reading Writing and Validation Basics (/s/article/inspire-gml-reading-writing-and-validation-basics)

Writing JSON with the JSONTemplater (/s/article/json-writing-with-jsontemplater)

Advanced XML / GML writing with Stylesheets (FME 2010) (/s/article/writing-xmlgml-with-xslt-style-sheets)

Writing XML/GML (/s/article/writing-xmlgml)

© Safe Software Inc | Legal (https://www.safe.com/legal/)

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.