SAFE SOFTWARE PRESENTS

HOW DO I DO THAT IN FME?

KNOWLEDGE.SAFE.COM

How Do I Do That in FME

Introduction

In 2004, a group of students at Cornell University created a guide titled: "How Do I Do That in ArcGIS/Manifold" which detailed how to complete a set of common GIS tasks using ArcGIS and Manifold tools. Their guide became increasingly popular in the GIS community and spawned many different iterations based on other GIS Packages, such as QGIS, PostGIS, ArcPy, and now FME. While FME is not a GIS, many of the operations described in these documents can be done as part of FME ETL workflows.

How Do I Do That in FME is a series of articles that focuses on the same five categories covered by its predecessors:

1. Database Management
2. Database Creation
3. Data Manipulation
4. Data Analysis
5. Data Display and Presentation.

The intention of this guide is to assist new users and GIS professionals who are in need of direction when performing a common task in FME. Additionally, this series of articles attempts to recreate the GIS tasks outlined in the "How do I do that in ArcGIS/Manifold" guide with a twist. Since FME is not a GIS, each exercise in this series aims to extract data from a source format, transform data, and load data into a destination format without data loss.

If you are a new FME user, please consider visiting the training page to learn more about the free online training provided by Safe Software or explore our FME tutorials.

Database Management

- Adding a Column to a Table
- Sorting Tabular or Graphical Data
- Calculating Values for New Fields
- Relating Data Files and Fields

Database Creation

- Assigning Topology
  - Creating a Polygon from Line Segments
  - Correcting Topological Errors
- Import and Export
  - Importing Database Tables, Raster Data, Vector Data

Data Manipulation

- Data Retrieval
- Select Data by Area Masks
- Data Restructuring
  - Convert from Raster to Vector

    - Convert from Vector to Raster
    - Modify Raster Cell Size by Resampling
    - Changing Raster Values by Geographic Feature
    - Reducing Unnecessary Coordinates – Weeding
- Smoothing Data to Recover Sinuosity
    - TIN from Point Data
    - Kriging from Point Data (RCaller: Interpolate Points to Raster Through Kriging)
    - Generate Contour Data from Points
    - Generate Contour Data from Raster
- Data Transformation
    - Mathematical Transformation of Raster Data (Simple Examples Using the RasterExpressionEvaluator Transformer)
    - Projection Definition and Coordinate Transformation
- Vector Overlay
    - Polygon in Polygon Overlay
    - Point in Polygon Overlay
    - Topological Intersection
    - Line in Polygon

Data Analysis

- Raster Processing
    - Mathematical Operations on One Raster (Simple Examples Using the RasterExpressionEvaluator Transformer)
    - Mathematical Operations on Two Rasters (Raster Calculations and Raster Palettes)
    - Raster Neighborhood Functions
- Statistical Function
    - Calculating Areas, Perimeters and Lengths
    - Cross Tabulation of Two Data Categories (Pivot Tables and FME)
- General
    - Specify Distance Buffers (Buffering features for Spatial Relationship Analysis)
    - Polygons Within Distance of Selected Features
    - Find Nearest Features
- 3D Analysis
    - Generating Slope and Aspect (Raster Slope Calculations)
    - Identifying Watersheds (Calculate Strahler Stream Order Numbers)
- Network Functions
    - Choosing the Optimal Path Through a Network (Shortest Route Calculations with the ShortestPathFinder)
    - Defining a Drive-Time Zone (Creating Time and Distance Isolines using the NetworkCostCalculator)
    - Geocoding Addresses

Data Display and Presentation

- Map and Map Feature Annotation

Data Attribution

Unless otherwise stated, the data used throughout this series originates from open data made available by the

Adding a Column to a Table

Overview

Adding new column(s) to an existing table can be accomplished by using either the AttributeCreator or AttributeManager and the FeatureHolder to prevent schema lock. In this tutorial, you will learn how to append a new column to an existing PostGIS table using the AttributeManager.

Downloads

add-column-to-table.fmwt

3531-downtown.csv

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this exercise, you will add a new column using the AttributeManager because you want to keep track of the last time your asset was reviewed. We will populate this new field by adding today's date using the DateTimeStamper. Alternatively, you can also calculate values, join tables, or leave the values as NULL.

Note: The scenario presented here is likely different than the intended real-world application. Because of the challenges of working with a public facing database, here you will read from the Downtown_PublicArt table and eventually write to a table named PublicArt_Date. You might notice PublicArt_Date already exists with the intended data model because other users would have also run the same workspace. In reality, you would likely read from Downtown_PublicArt (which has 2 columns) and then drop and create Downtown_PublicArt with the additional column.

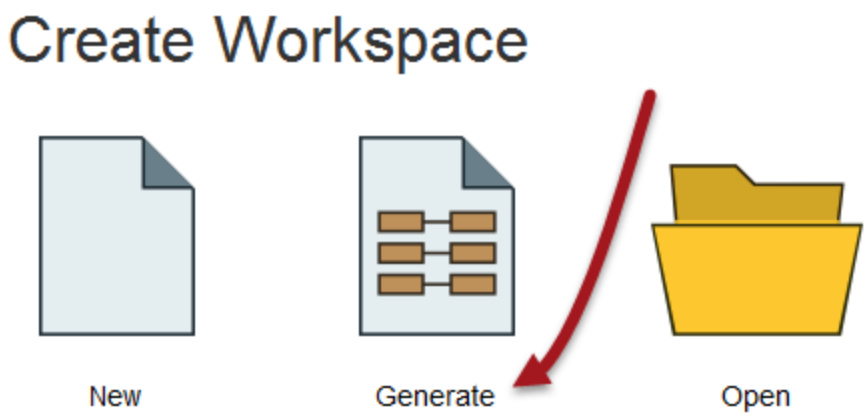
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

2. Generate Workspace

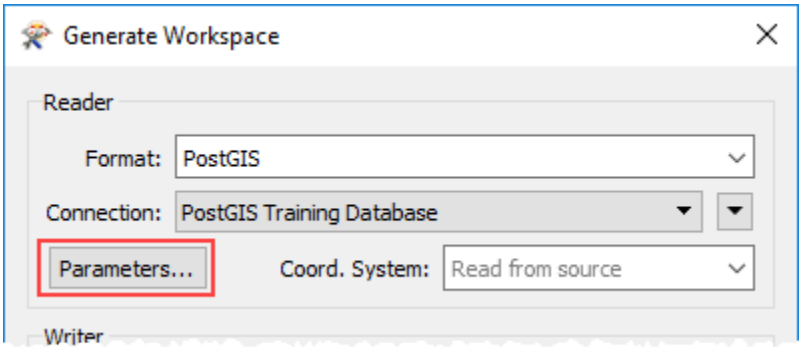
- In the Create Workspace part of the Start page select the option to Generate (Workspace). Alternatively, you can use the shortcut Ctrl+G.



For more information on Creating a Translation, see the [Desktop Basic Course Manual](#).

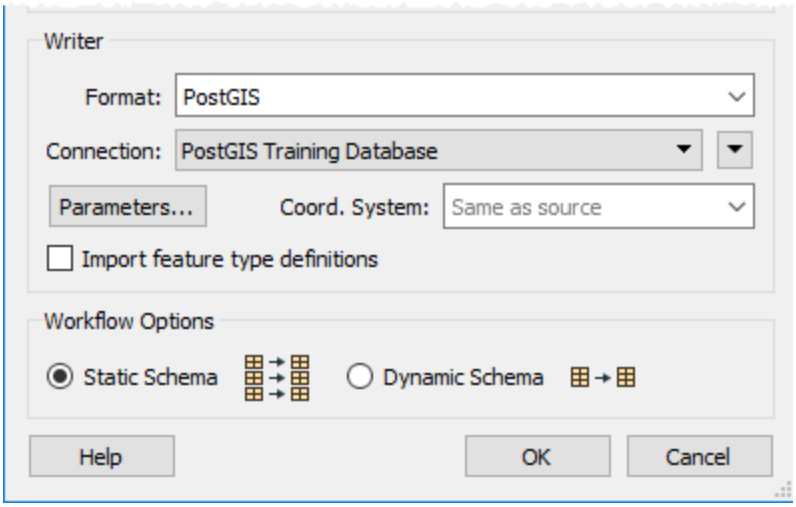
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Downtown_PublicArt table from the Table List.



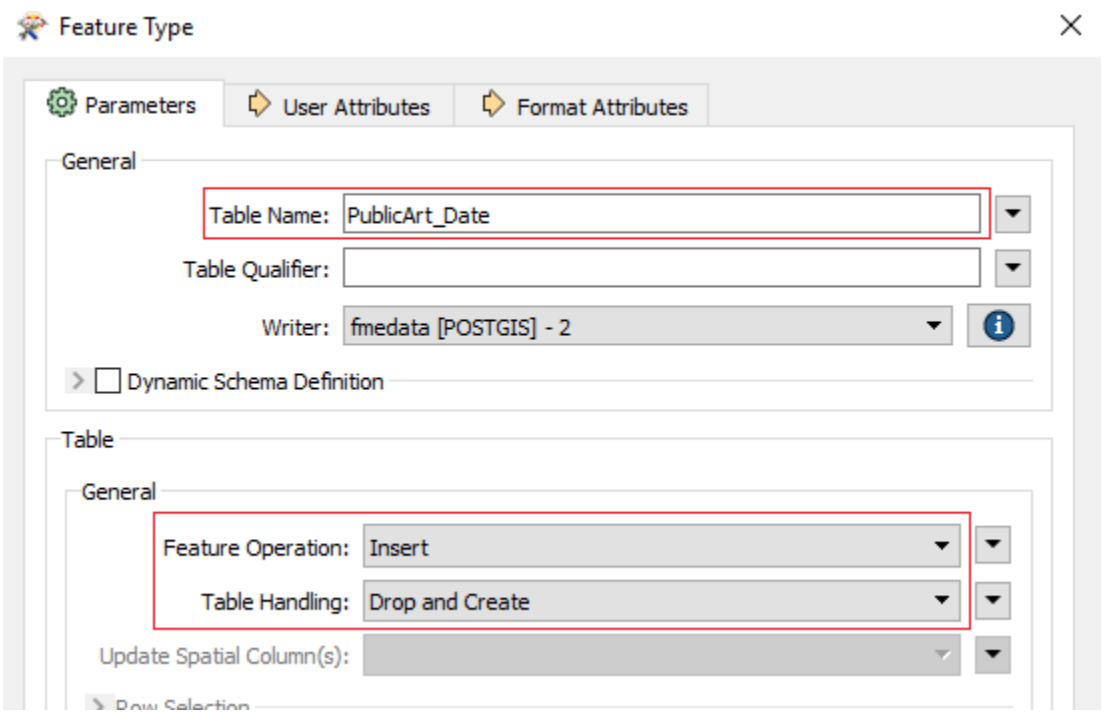
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to PublicArt_Date
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.

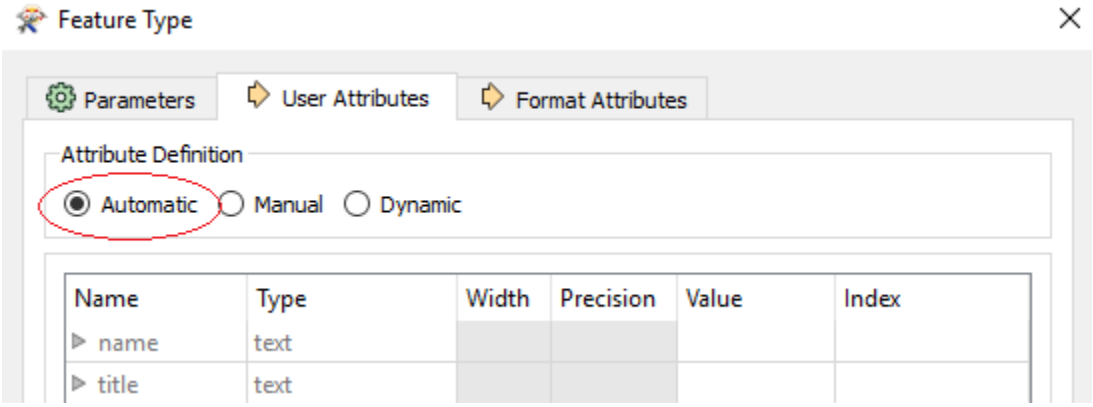


Note: Drop and Create is used when a table needs to be completely removed and recreated with updates to the database schema. For example, drop and create is used to remove the table structure, to possibly add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the Documentation.

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.

- Click OK to accept the changes and close the Writer Parameters dialog.



Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Documentation.

7. Close Source Table using a FeatureHolder

Prior to dropping the existing PublicArt table, the table needs to be closed by FME in order to prevent schema lock - which will cause an error in your translation. We'll use a FeatureHolder to read and store features from the PublicArt table. Once all of the features have been stored, the existing table is closed, allowing FME to drop the existing table and create the new table.

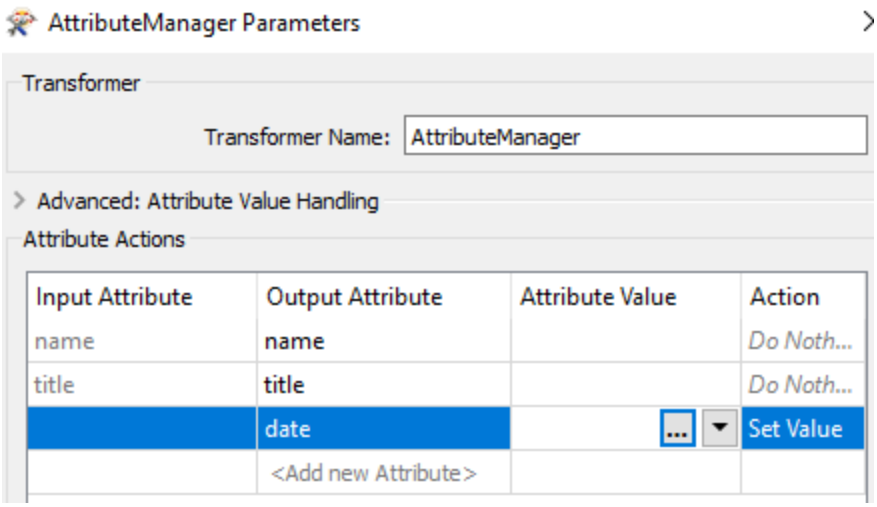
- Add a FeatureHolder to the canvas by typing “FeatureHolder” to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the Downtown_PublicArt Reader to the FeatureHolder.

8. Create a New Column with AttributeManager

- Place the AttributeManager after the FeatureHolder. The AttributeManager will be used to add a new column to the Downtown_PublicArt table.

9. Create a New Column

- Once the AttributeManager has been added, double-click the AttributeManager or click on the gear icon to open the transformer parameters dialog.
- Click the last row of the Output Attribute column <Add new Attribute> to create a new column.
- Set the new attribute name to “date”, this will automatically change the Action to “Set Value”.



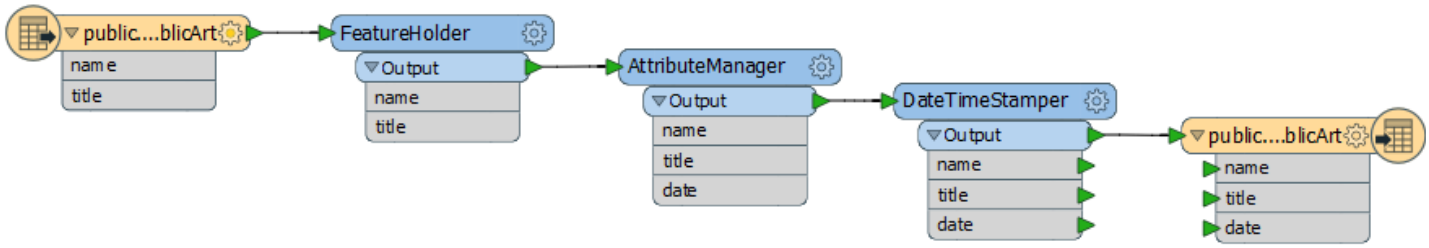
Note: The AttributeManager allows you to add/remove table columns, change attribute names, and set attribute values from existing FME Feature Attribute or using the Text Editor. For more information on Managing Attributes in the [Desktop Basic Course Manual](#).

10. Add Today's Date

- Add a DateTimeStamper transformer after the AttributeManager. After adding the transformer, connect it to the AttributeManager:Output port. Next, connect the DateTimeStamper:Output port to the PublicArt_Date writer.
- Once DateTimeStamper has been properly connected, open the transformer parameters.
- In the DateTimeStamper parameters dialog, set the Result Attribute to "date" - this will set the value of the "date" field to the current date in the column that you just created in the AttributeManager. Once you have assigned the Result Attribute, click OK.

11. Run Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, your table will have a date column with today's date. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected. Alternatively, you can attach an Inspector to the output of the final transformer in the workspace if you wish to automatically open the workspace output the FME Inspector.

Results

Input

Table View

fmedata [POSTGIS] - public.Downtown_PublicArt

	name	title
1	Harbour Centre...	The Belonging ...
2	Harbour Centre...	The Belonging ...
3	Chinese Cultur...	China Gate
4	Vancouver Inter...	Moving Pictures
5	Shanghai Alley	Suan Phan: Aba

in any column

Output

Table View

fmedata [POSTGIS] - PublicArt_Date

	name	title	date
1	Harbour Centre...	The Belonging ...	20180604
2	Harbour Centre...	The Belonging ...	20180604
3	Chinese Cultur...	China Gate	20180604
4	Vancouver Inter...	Moving Pictures	20180604
5	Shanghai Alley	Suan Phan: Aba	20180604

in any column

Transformers

- AttributeManager- Alters multiple attributes through adding, renaming, copying, deleting and re-ordering.
- DateTimeStamper- Adds a timestamp to a feature as a new attribute in the form of a date, time (with or without UTC offset), or datetime (with or without UTC offset), in local or UTC time.
- FeatureHolder- Stores incoming features until they have all arrived, and then releases them in their original order.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Sorting Tabular or Graphical Data

Overview

Sorting tabular data can be done multiple ways using FME. You can sort attributes using the FME Data Inspector to manually order columns or by using the Sorter Transformer in the Workbench. The Sorter functions similar to the SQL function: ORDER BY. In this tutorial, you will learn how to sort tabular data in the FME Data Inspector and using the Sorter Transformer.

Downloads

sort-tabular-data.fmwt

drinkingfountains.csv

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise 1 - *Sorting Data Using the FME Data Inspector*

In this scenario, you want to have the ability to sort your tabular data in ascending or descending order (either alphabetically or numerically) using the FME Data Inspector.

Instructions

1. Start FME Workbench

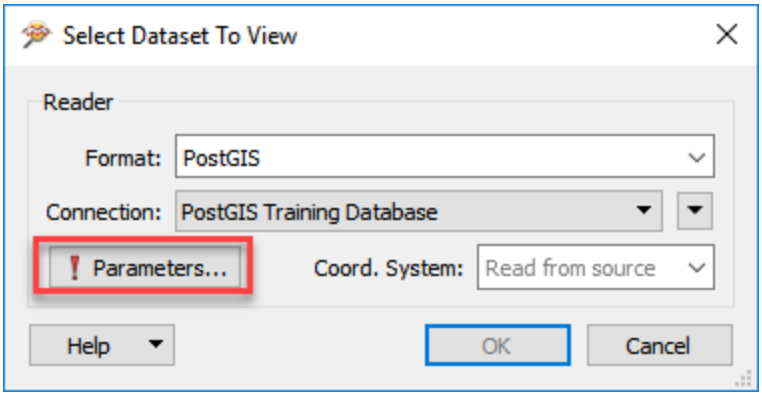
- If it isn't open already, launch FME Data Inspector.

2. Open a Dataset

- Using the shortcut CTRL + O, the open folder icon, or by navigating to File -> Open Dataset.

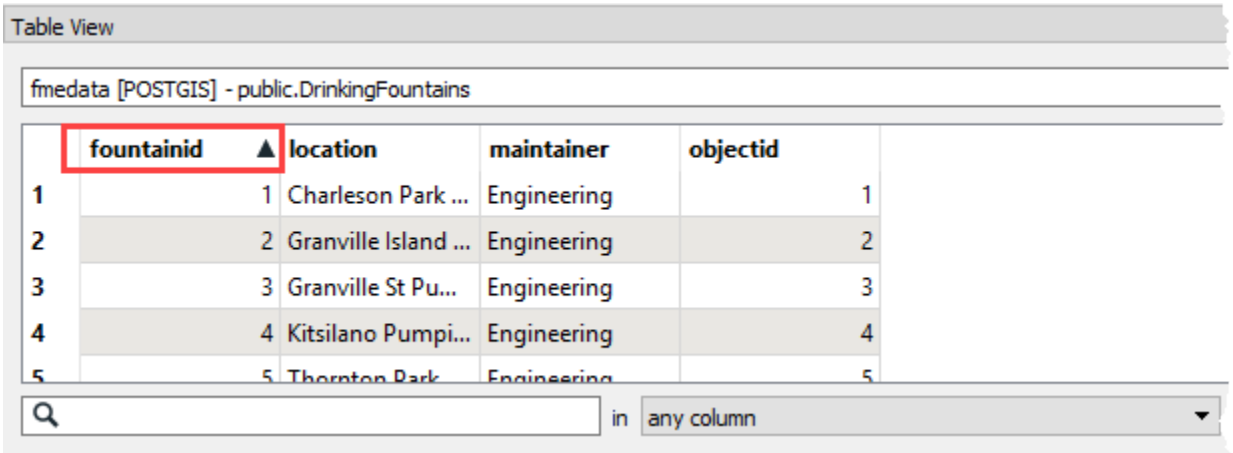
3. Select a Dataset to View

- Set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedataPassword: fmedata
- Next, open the Reader Parameters and select the DrinkingFountains table from the Table List.



4. Sort Columns (Ascending)

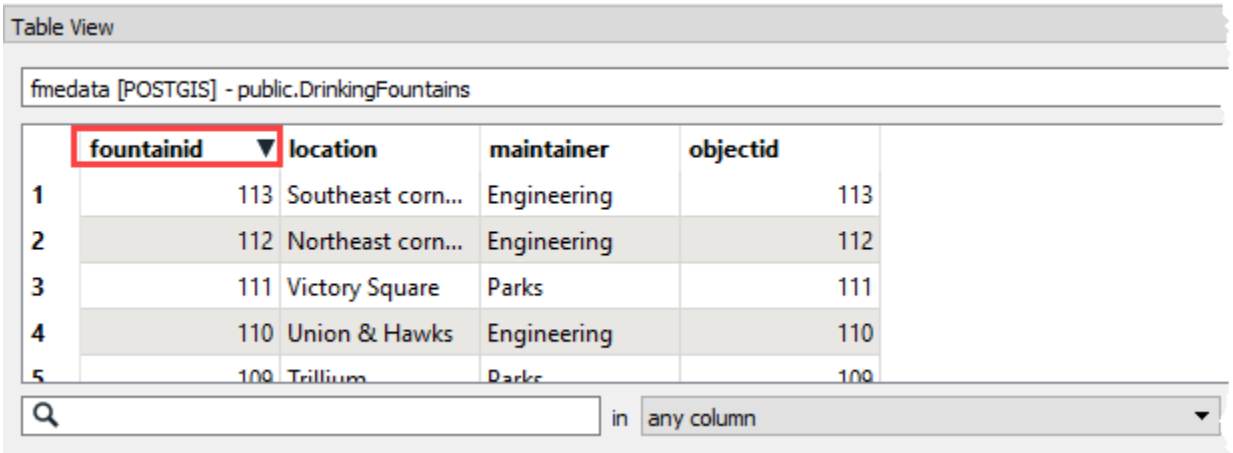
- In the Table View section of the Data Inspector, click the “fountainid” column header to sort by ascending order.



If the attributes aren’t sorting correctly you may be using the wrong sorting method. Right-click on the column header and select the Sort Numeric Ascending sorting method from the drop-down list.

5. Sort Columns (Descending)

- Click on the “fountainid” column header again to change the column sorting from ascending to descending.



6. Clear all Sorting

- Right-click on any of the column headers, then select “Clear All Sorting”. This will remove all sorting that has been applied to the table and return the dataset to its original order.

Exercise 2 - *Sorting Data Alphabetically using the Sorter*

In this scenario, you want to have the ability to sort your tabular data by multiple columns. First, you want your table sorted alphabetically by the maintainer field then numerically by the fountainid field in descending order.

Note: The scenario presented here is likely different than the intended real-world application. Because of the challenges of working with a public facing database, here you will read from the DrinkingFountains table and eventually write to a table named DrinkingFountainsSorted. You might notice DrinkingFountainsSorted already exists with the intended data model because other users would have also run the same workspace. In reality, you would likely read from DrinkingFountains, and use the Table Handling "Use Existing" since you are only reordering features in the table.

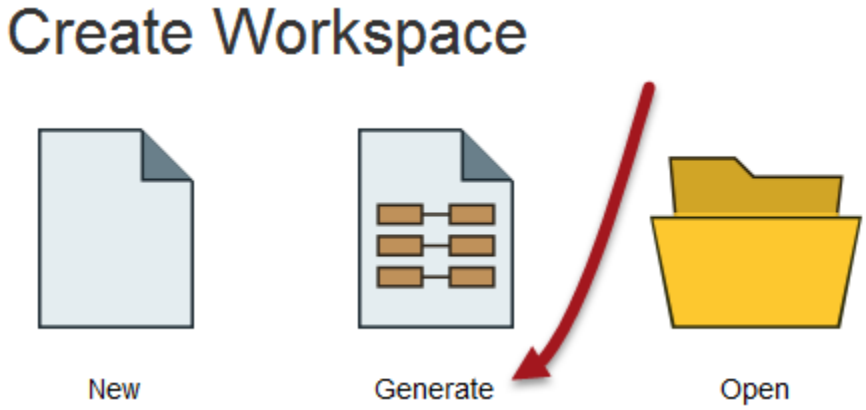
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

2. Select Generate Workspace

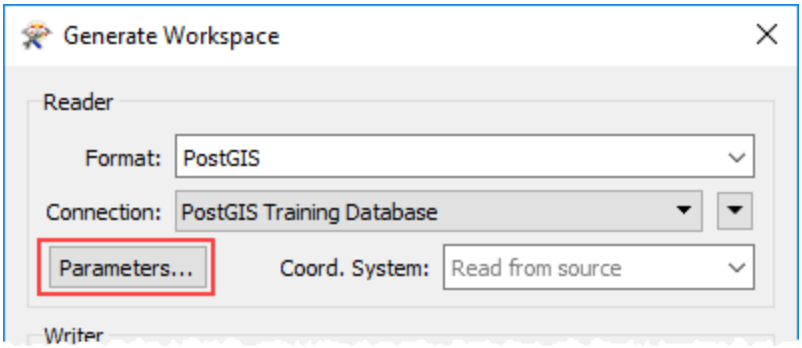
- In the Create Workspace part of the Start page, select the option to Generate a workspace.



Note: the shortcut to generate a workspace is Ctrl+G

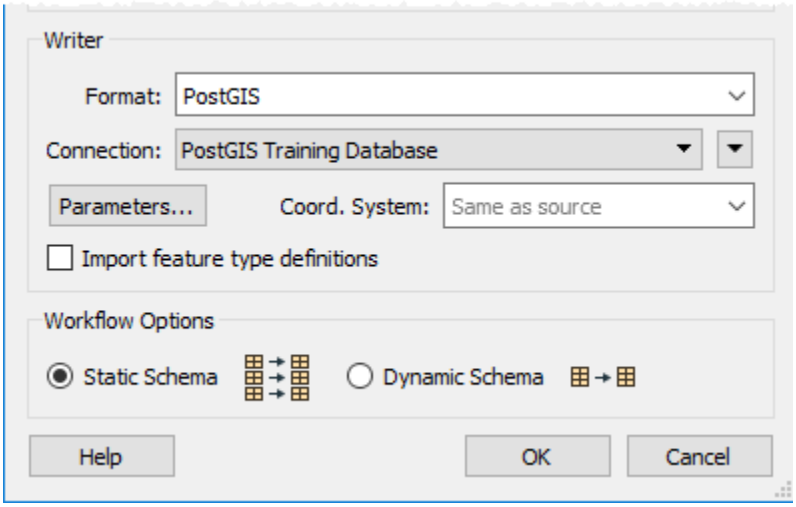
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the DrinkingFountains table from the Table List.



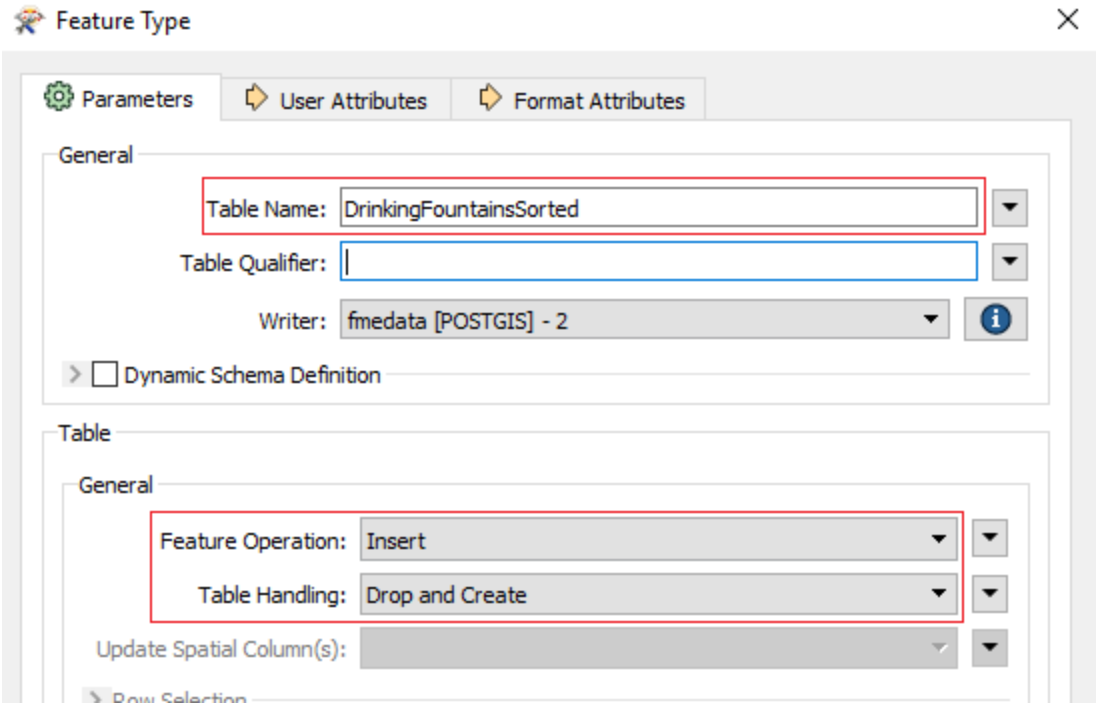
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

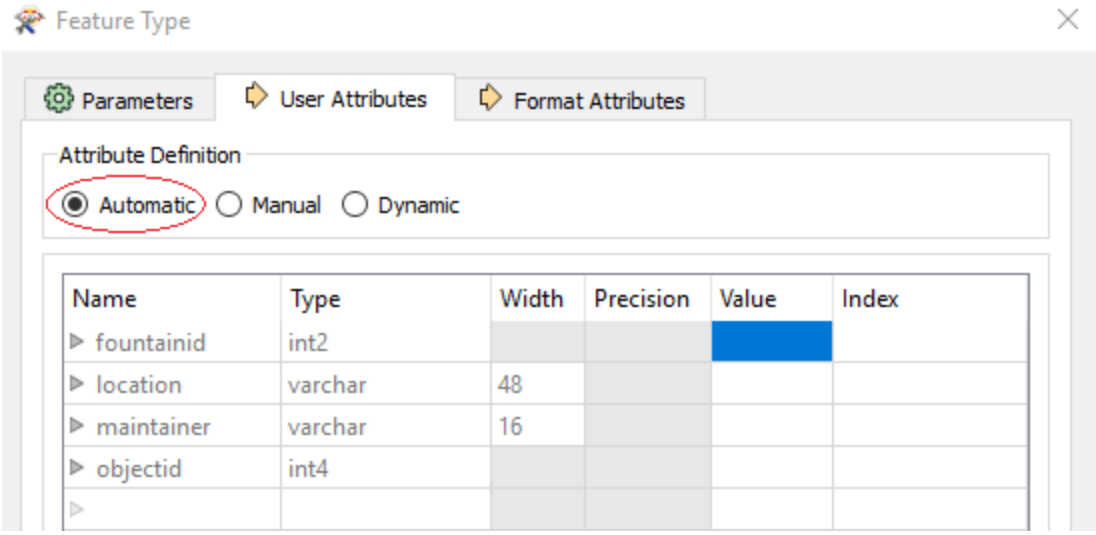
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to DrinkingFountainsSorted.
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the [Documentation](#).

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.
- Click OK to accept the changes and close the Writer Parameters dialog.



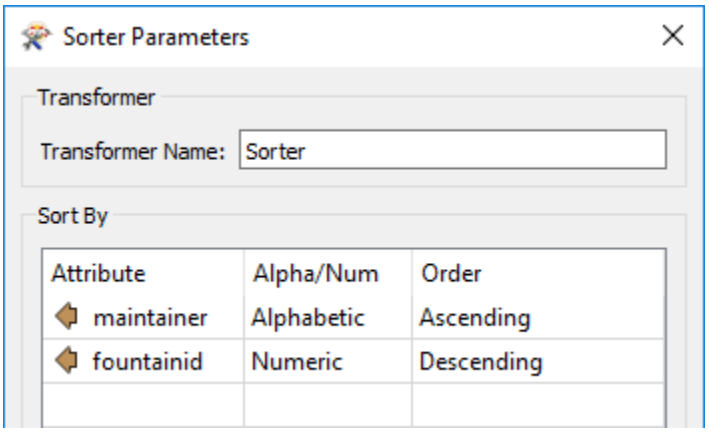
Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the [Documentation](#).

7. Add a Sorter

- Add a Sorter to the canvas by typing “Sorter” to bring up the list of FME Transformers in the Quick Add Search. Select the Sorter from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the DrinkingFountains Reader to the Sorter, then connect the Sorter:Sorted port is to the DrinkingFountainsSorted Writer.

8. Sort Alphabetically

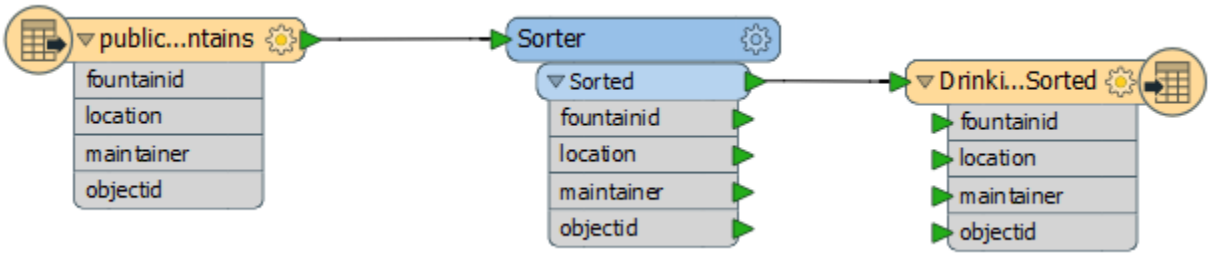
- Once the Sorter has been added, double-click the Sorter or click on the gear icon to open the transformer parameters dialog.
- In the Sort By section, click on the empty cell under the Attribute Column and select the “maintainer” attribute from the drop-down list.
- Next, ensure the Alpha/Num is set to Alphabetic and the Order is Ascending, again by selecting from the drop-down list.
- Similarly, set the second Sort By condition to sort the “fountainid” attribute in Numeric Descending order.



Note: the order of the Sort By operations will determine which column is sorted first (top of the list gets sorted first). You can adjust the order of each sort condition using the up and down arrows near the bottom of the dialog.

9. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the table, your table will be sorted by the ordering methods that were defined in the Sorter Parameters. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected.

Results

Input

Table View

fmedata [POSTGIS] - public.DrinkingFountains

	fountainid	location	maintainer	objectid
1	1	Charleson Park ...	Engineering	1
2	2	Granville Island ...	Engineering	2
3	3	Granville St Pu...	Engineering	3
4	4	Kitsilano Pumpi...	Engineering	4
5	5	Thornton Park	Engineering	5

in any column

Output

Table View

Table: fmedata [POSTGIS] - DrinkingFountainsSorted

	fountainid	location	maintainer	objectid
1	113	Southeast corn...	Engineering	113
2	112	Northeast corn...	Engineering	112
3	110	Union & Hawks	Engineering	110
4	72	Robson & Chilco	Engineering	72
5	71	Robson & Burra...	Engineering	71
6	66	Nicola & Beach	Engineering	66
7	65	Nelson & Brou...	Engineering	65

in any column

Transformers

- Sorter - Sorts features by a selected attribute's value.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Calculating Values for a New Field

Overview

In this tutorial, you learn how to calculate values using the ExpressionEvaluator and how to assign values using the AttributeValueMapper.

Downloads

calculating-field-values.fmwt

drinkingfountains.csv

neighborhoodsgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise 1 - *Calculating Values using the ExpressionEvaluator*

In the first exercise, you will calculate population change over a 10 years period. This requires a simple mathematical operation between two FME Feature Attributes that can be performed using the ExpressionEvaluator.

Note: The scenario presented here is likely different than the intended real-world application. Because of the challenges of working with a public facing database, here you will read from the Neighborhood table and eventually write to a table named Neighborhood_Pop. You might notice Neighborhood_Pop already exists with the intended data model because other users would have also run the same workspace. In reality, you would likely read from Neighborhood and then update the existing table using the Feature Type Handling Option "Use Existing".

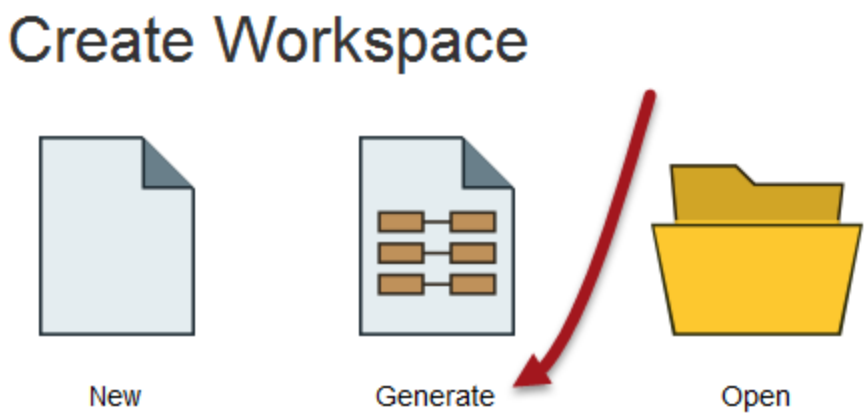
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

2. Select Generate Workspace

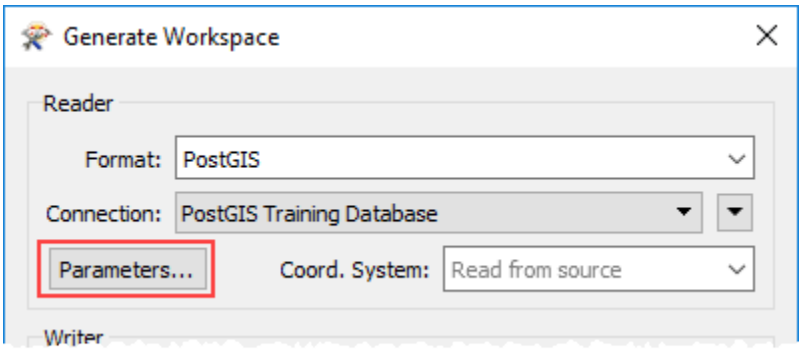
- In the Create Workspace part of the Start page select the option to Generate (Workspace). Alternatively, you can use the shortcut Ctrl+G.



- For more information on Creating a Translation, see the Desktop Basic Course Manual.

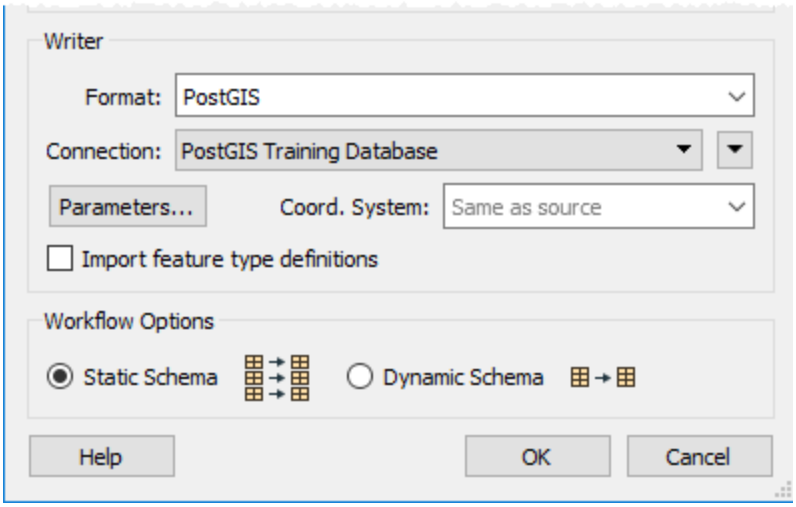
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list.
- If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Name: PostGIS Training Database
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Neighborhoods table from the Table List.



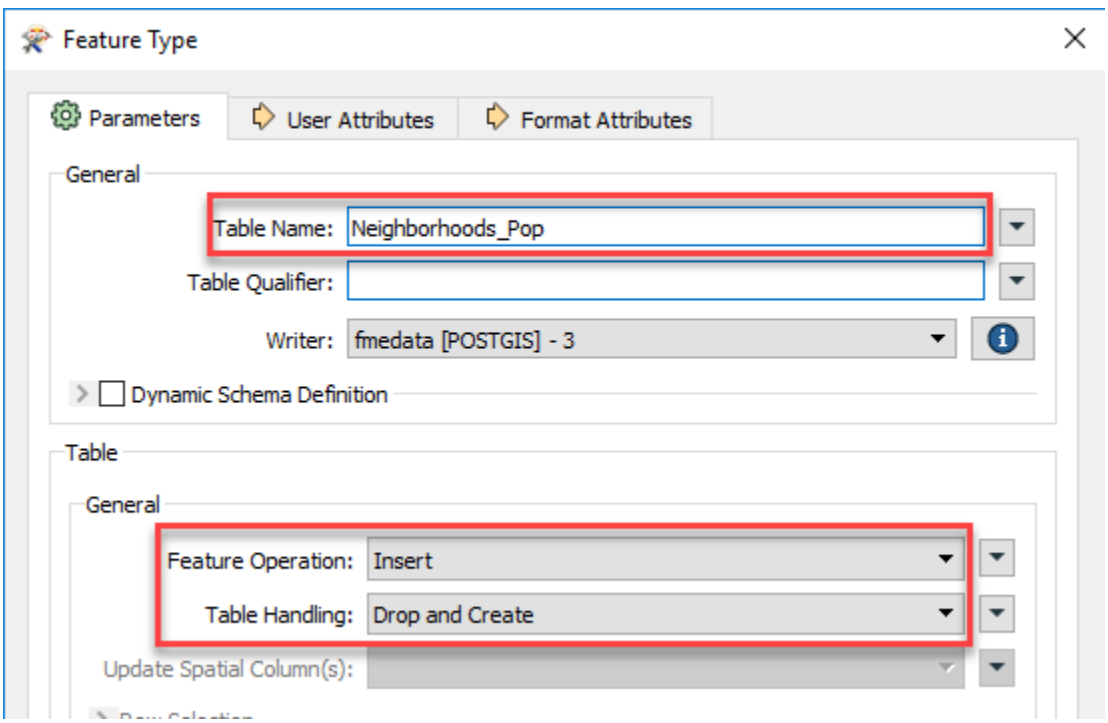
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace.



5. Set the Feature Operation and Table Handling

- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to Neighborhoods_Pop.
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.

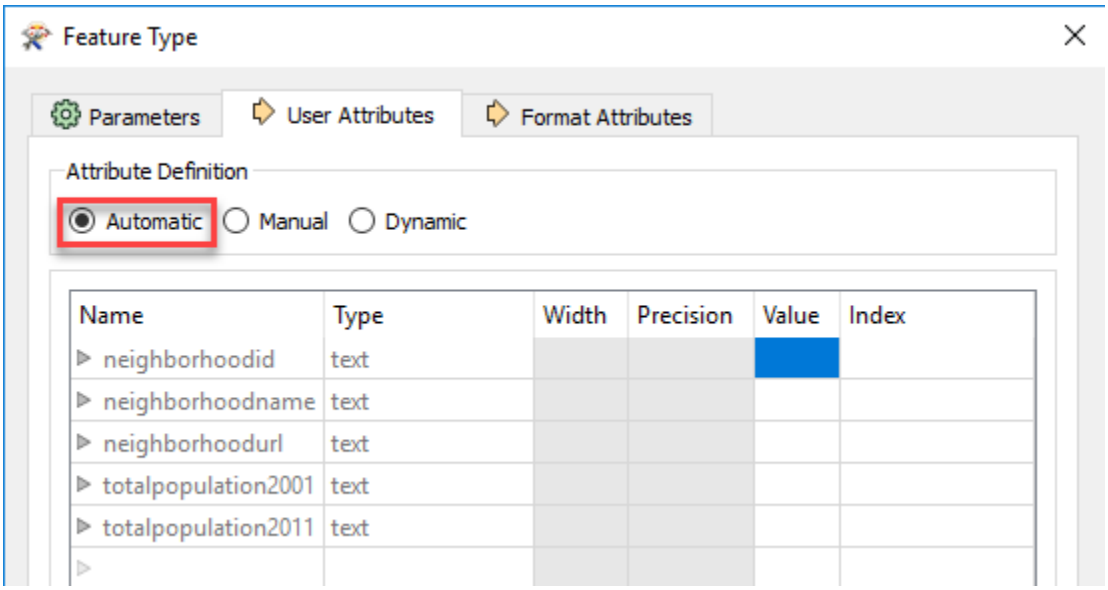


Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the [Documentation](#).

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.

- Click OK to accept the changes and close the Writer Parameter dialog.



Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Docum entation.

7. Add a FeatureHolder

- Add a FeatureHolder to the canvas by typing "FeatureHolder" to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the Neighborhoods Reader to the FeatureHolder.

The FeatureHolder reads and stores the features in the Neighborhoods table. Once all of the features have been stored, the existing table is closed and which allows FME to drop the existing table and create the new table.

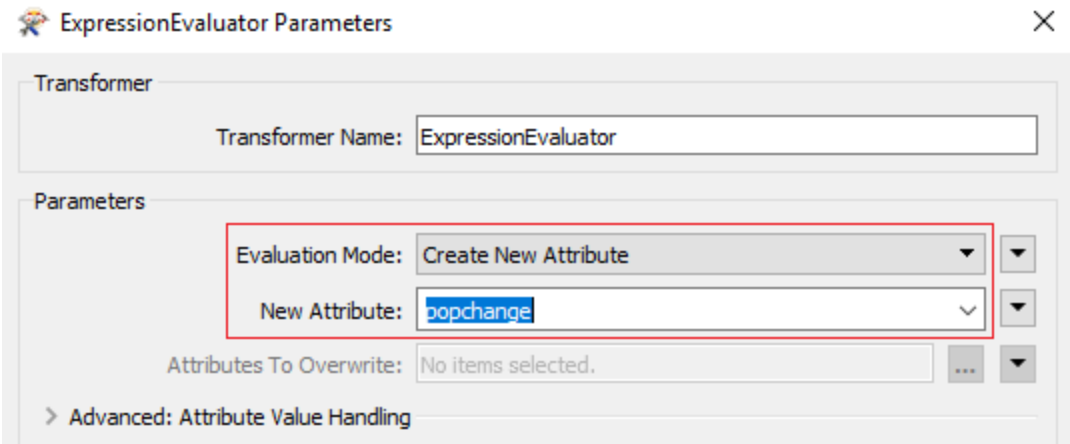
8. Add an ExpressionEvaluator

- Connect the FeatureHolder:Output port to the ExpressionEvaluator.
- Ensure that the ExpressionEvaluator:Output port is connected to the Neighborhoods_Pop Writer.

The ExpressionEvaluator performs a mathematical calculation on an expression that consists of FME Feature Functions, String Functions, Math Functions, and Math Operators. In this case, we will use it to determine the change in population from 2001 to 2011 for each neighborhood. For more information on Constructing Attributes, see the Desktop Basic Course Manual.

9. Set the ExpressionEvaluator Parameters

- Once the ExpressionEvaluator has been added, double-click the ExpressionEvaluator or click on the gear icon to open the transformer parameters dialog.
- Leave the evaluation mode as "Create New Attribute" and set the name of the New Attribute to "popchange".

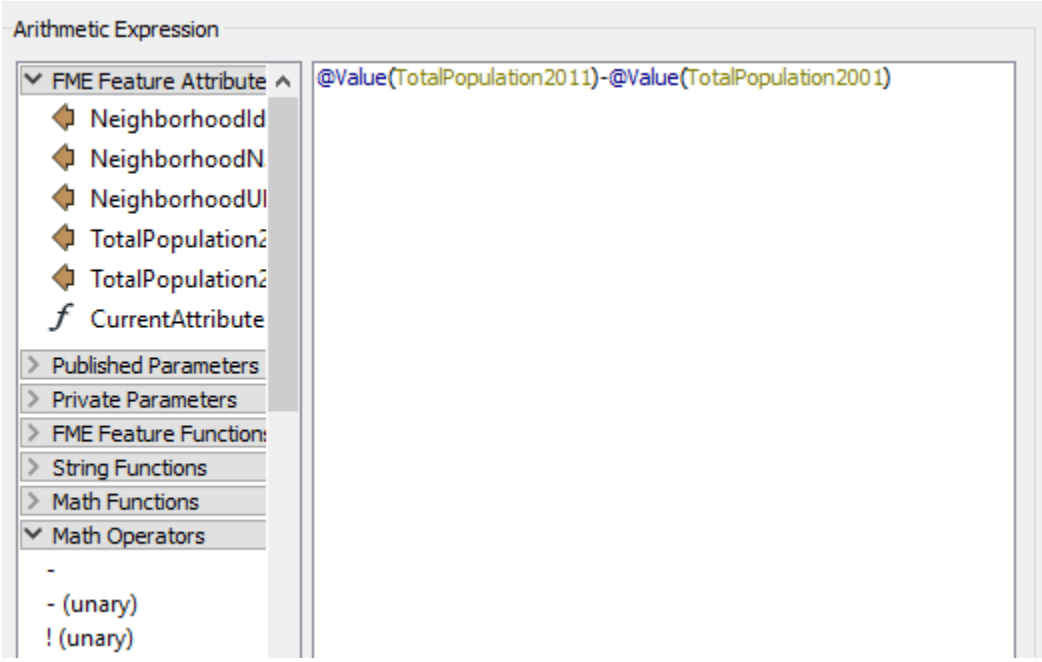


10. Calculate Population Change using an Arithmetic Expression

FME allows you to access attributes from any of the datasets that are connected to the ExpressionEvaluator via the FME Feature Attributes list. To calculate population change, subtract the 2001 population from the 2011 population - both of which are attributes stored in the table.

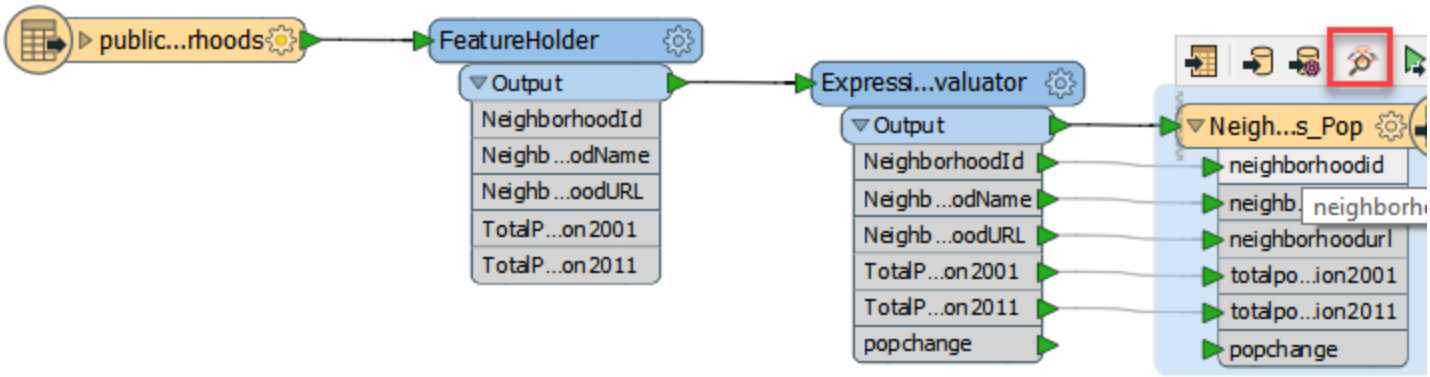
- Expand the FME Feature Attributes list in the Arithmetic Expression dialog and double-click “TotalPopulation2011” to add it to the expression.
- Next, expand the Math operators list and double-click the subtract operator (“-”) or hit the “-” key on the keyboard.
- Add the "TotalPopulation2001" value from the FME Feature Attributes list so that your final expression looks like this:

```
@Value(TotalPopulation2011)-@Value(TotalPopulation2001)
```



11. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After a successful translation, your table will have a new column with the population change for each neighborhood. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected.

Results

Input

Table View

fmedata [POSTGIS] - public.Neighborhoods

	NeighborhoodId	NeighborhoodName	NeighborhoodURL	TotalPopulation2001	TotalPopulation2011
1	1	Kitsilano	http://vancouver....	39620	41375
2	2	Fairview	http://vancouver....	28405	31445
3	3	Mount Pleasant	http://vancouver....	24535	26400
4	4	Downtown	http://vancouver....	27990	54690
5	5	West End	http://vancouver...	42120	44540

in any column

Output

Table View

fmedata [POSTGIS] - Neighborhoods_Pop

	neighborhoodid	neighborhoodname	neighborhoodurl	totalpopulation2001	totalpopulation2011	popchange
1	1	Kitsilano	http://vancouve...	39620	41375	1755
2	2	Fairview	http://vancouve...	28405	31445	3040
3	3	Mount Pleasant	http://vancouve...	24535	26400	1865
4	4	Downtown	http://vancouve...	27990	54690	26700
5	5	West End	http://vancouve...	42120	44540	2420

in any column

Exercise 2 - *Assign Values Based on Another Attribute*

In this exercise, you will assign values based on the value of the maintainer attribute using the AttributeValueMapper. The AttributeValueMapper will lookup the source value of the maintainer attribute and assign the destination value (department code) to a new field. This will allow you to join tables to the DrinkingFountains table using the department code as your primary key.

Note: The scenario presented here is likely different than the intended real-world application. Because of the challenges of working with a public facing database, here you will read from the DrinkingFountains table and eventually write to a table named DrinkingFountainsRemap. You might notice DrinkingFountainsRemap already exists with the intended data model because other users would have also run the same workspace. In reality, you would likely read from DrinkingFountains and then update the existing table using the Feature Type Handling Option "Use Existing".

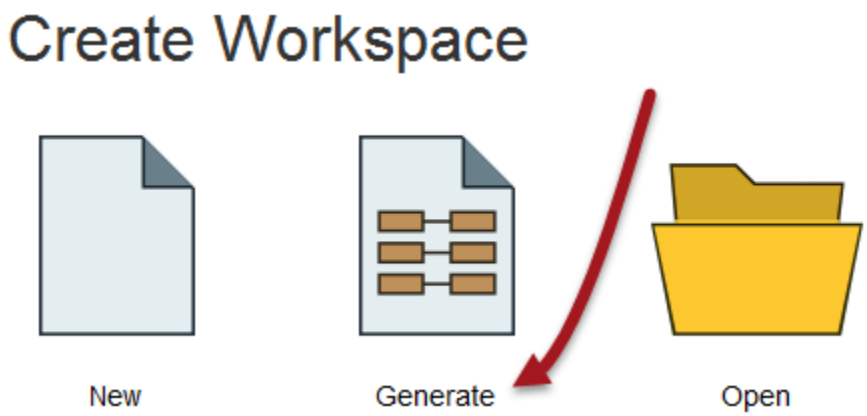
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Select Generate Workspace

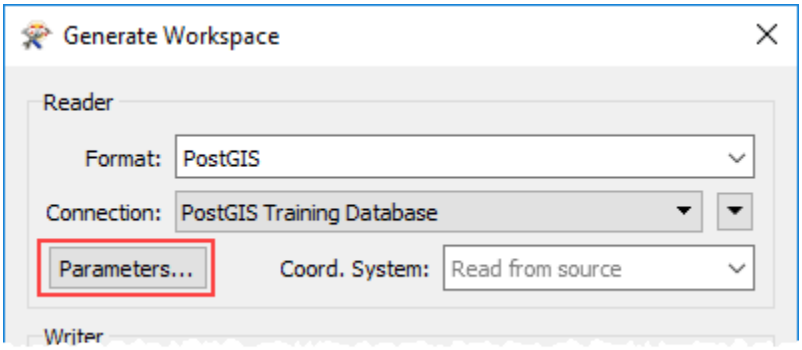
- In the Create Workspace part of the Start page select the option to Generate (Workspace). Alternatively, you can use the shortcut Ctrl+G.



For more information on Creating a Translation, see the Desktop Basic Course Manual.

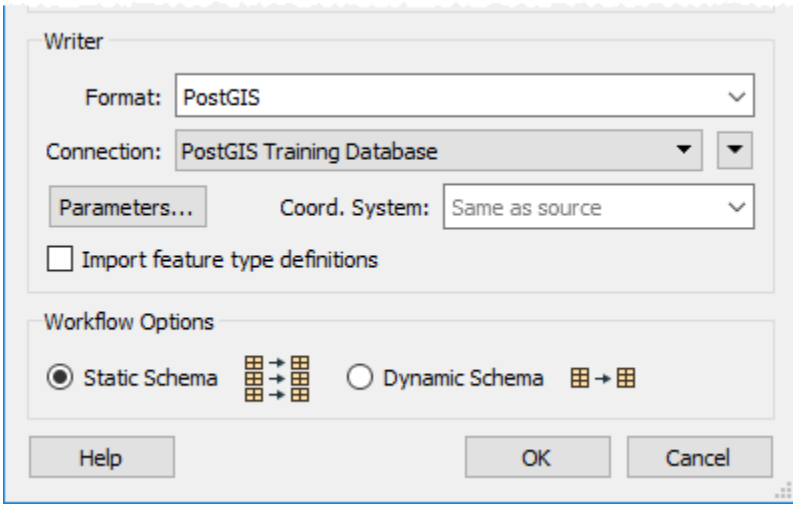
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list.
- If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the DrinkingFountains table from the Table List.



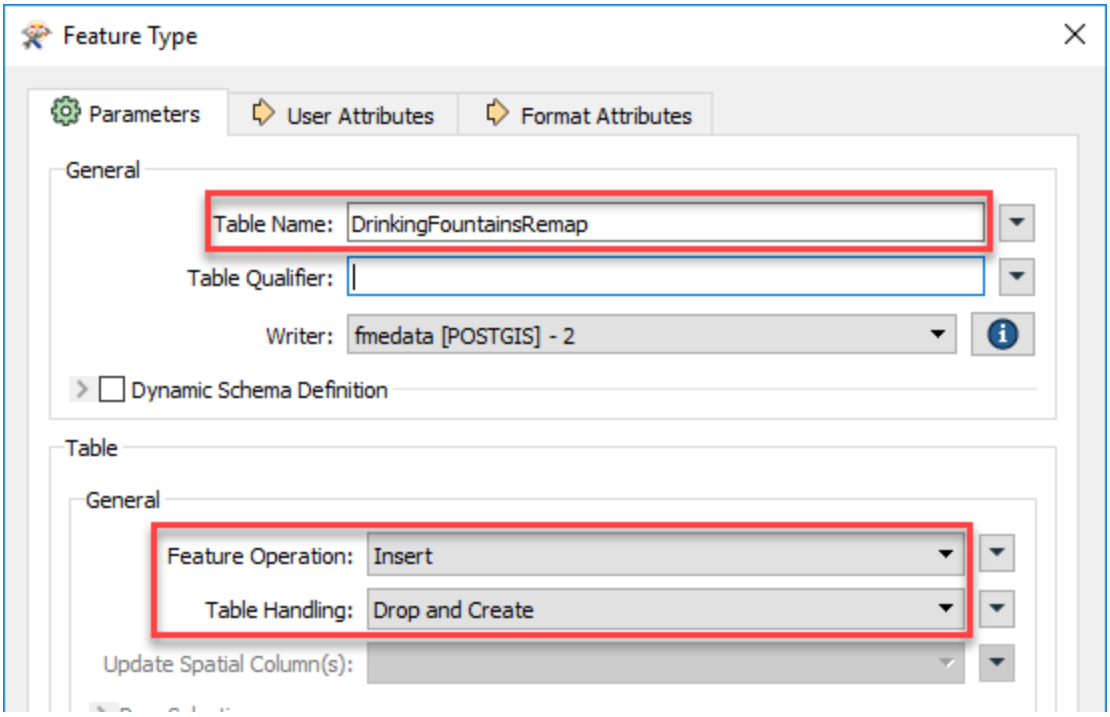
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace:



5. Set the Feature Operation and Table Handling

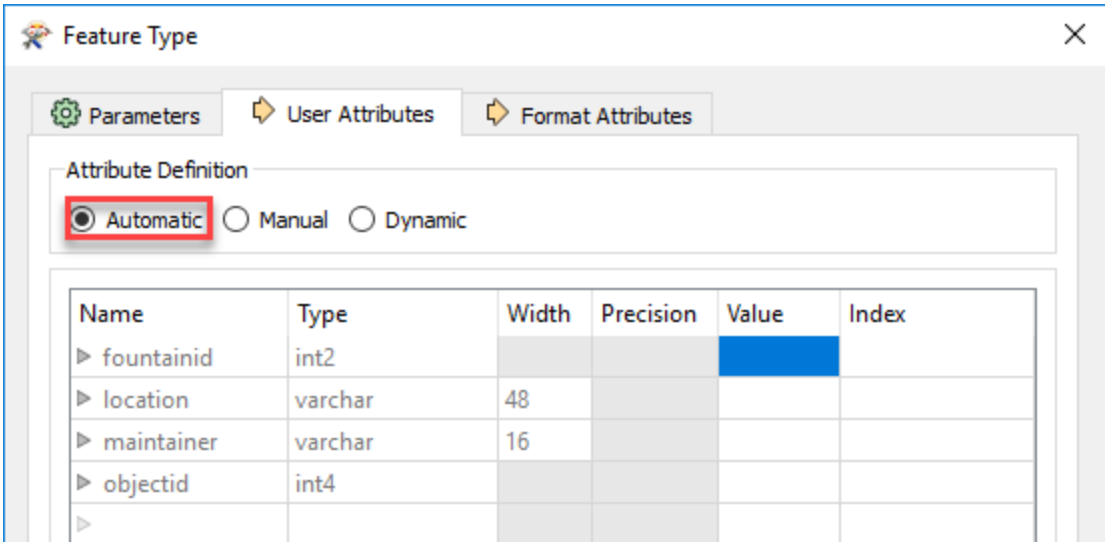
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to DrinkingFountainsRemap.
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the [Documentation](#).

6. Update Attributes

- In the User Attributes Tab of the Writer Feature Type, set the Attribute Definition to Automatic so any columns that are modified or created in the workspace will appear in the new table.
- Click OK to accept the changes and close the Writer Parameters dialog.



Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the [Documentation](#).

7. Add a FeatureHolder

- Add a FeatureHolder to the canvas by typing “FeatureHolder” to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the DrinkingFountains Reader to the FeatureHolder.

The FeatureHolder reads and stores the features in the DrinkingFountains table. Once all of the features have been stored, the existing table is closed and which allows FME to drop the existing table and create the new table.

8. Inspect Attributes

Before remapping attribute values, you first need to know the values of the attributes we are dealing with.
Note: the value of the maintainer attribute is either Engineering or Parks.

- Click on the Public.DrinkingFountains Reader to reveal the shortcut menu then click on the FME Data Inspector icon to preview the table.

In the Table View, notice that the value of the maintainer attribute is either Engineering or Parks.

After previewing the dataset, close the FME Data Inspector and return to the FME Workbench.

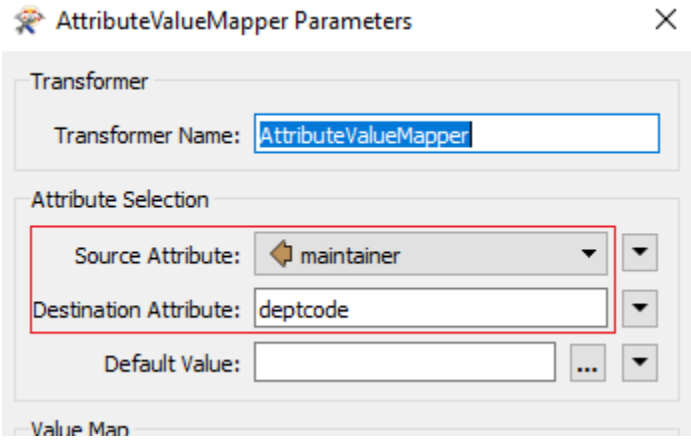
9. Add an AttributeValueMapper

- Place the AttributeValueMapper after the FeatureHolder. The AttributeValueMapper will be used to assign values to the new attribute.

10. Set the AttributeValueMapper Parameters

We want to create a new attribute called deptcode and assign the values based on the maintainer attribute:

- In the AttributeValueMapper transformer parameters, set the Source Attribute to the "maintainer" attribute and set the Destination attribute to "deptcode"

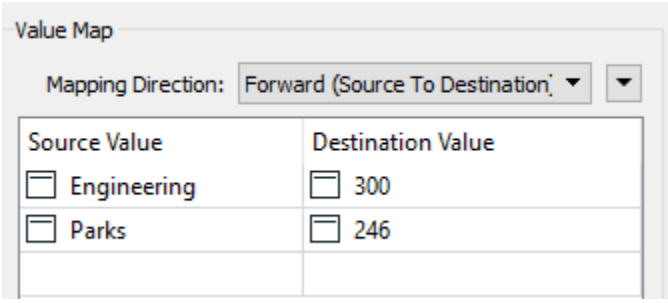


11. Set the Source and Destination Values

Since you know the only two attributes in the maintainer column are “Engineering” and “Parks”, you can simply type the values into the Source Value in the Value Map.

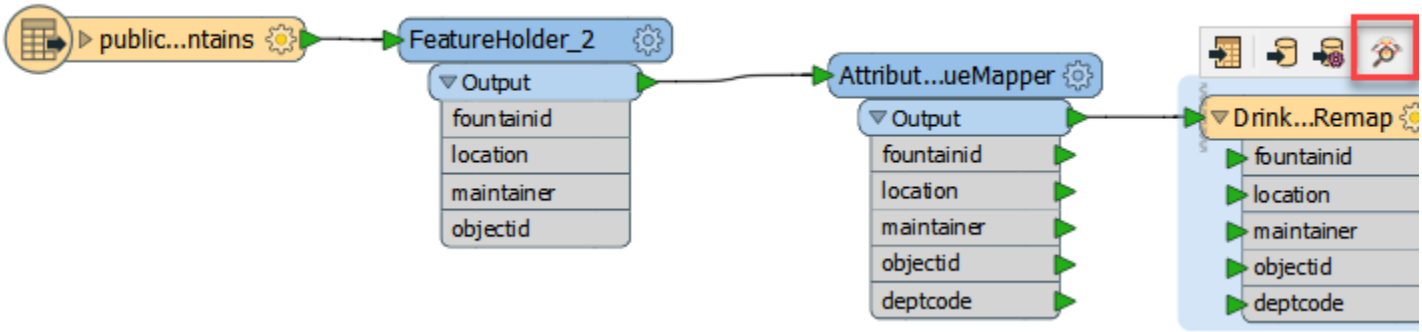
- If the Source value of the maintainer field is Engineering, set the Destination value to 300.
- Similarly, if the Source value of the maintainer field is Parks, set the Destination value to 246.

Ensure your Value Map is similar to the screenshot below, then click OK to accept the changes and close the AttributeValueMapper parameters



12. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, your table will have a new column with the department codes for the engineering and parks departments. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected.

Results

Input

Table View

fmedata [POSTGIS] - public.DrinkingFountains

	fountainid	location	maintainer	objectid
1	1	Charleson Park ...	Engineering	1
2	2	Granville Island ...	Engineering	2
3	3	Granville St Pu...	Engineering	3
4	4	Kitsilano Pumpi...	Engineering	4
5	5	Thornton Park	Engineering	5

in any column

Output

Table View

fmedata [POSTGIS] - DrinkingFountainsRemap

	fountainid	location	maintainer	objectid	deptcode
1	109	Trillium	Parks	109	246
2	108	Teaswamp	Parks	108	246
3	107	Sutcliffe	Parks	107	246
4	106	Sunset Beach	Parks	106	246
5	105	Strathcona	Parks	105	246

in any column

Transformers

- AttributeValueMapper - Looks up and assigns attribute values based on other attributes, and stores the looked-up value in a new attribute.
- ExpressionEvaluator - Performs a mathematical calculation on an expression that consists of FME Feature Functions, String Functions, Math Functions, and Math Operators.
- FeatureHolder - Stores incoming features until they have all arrived, and then releases them in their original order.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Relating Data Files and Fields

Overview

Relating database tables in FME 2018.0 is fast and easy using the new FeatureJoiner transformer. The FeatureJoiner combines the attributes of features based on common attribute values, similar to a SQL join operation. In this tutorial, you will learn how to join features from two database tables based on a primary key, but this method can be modified to combine features from multiple data types (for example, shapefile and database table).

Downloads

relate-fields.fmwt

publicart.csv

vanhomes.zip

addresspoints.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise 1 - *Relate Database Tables using the FeatureJoiner*

In this exercise, you will learn how to join two tables based on a primary key. In this instance, you are interested in joining two tables based on the Address ID because you want to have a dataset with complete mailing addresses.

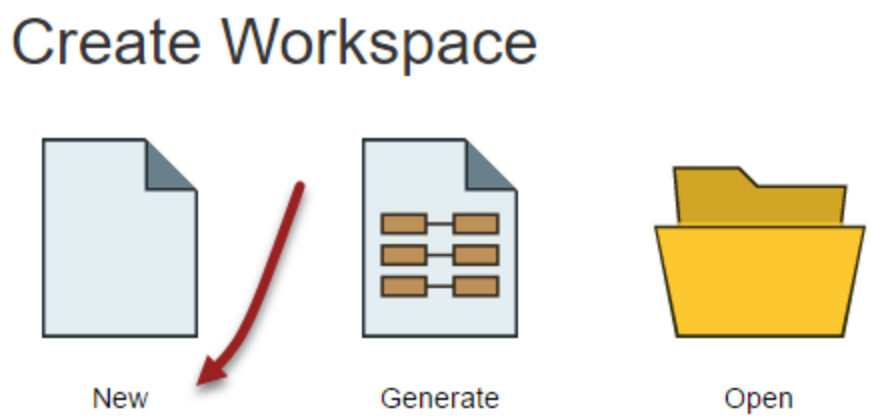
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

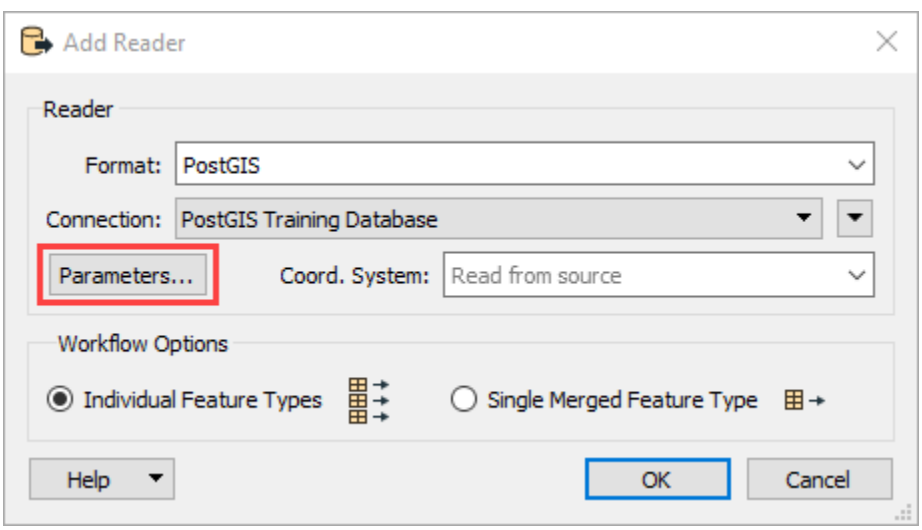
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



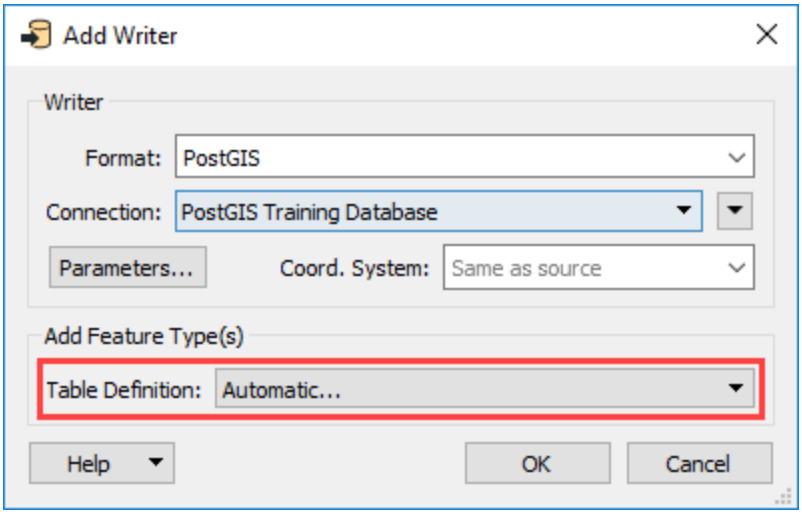
3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the workspace, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the AddressPoints and VanHomes tables from the Table List.



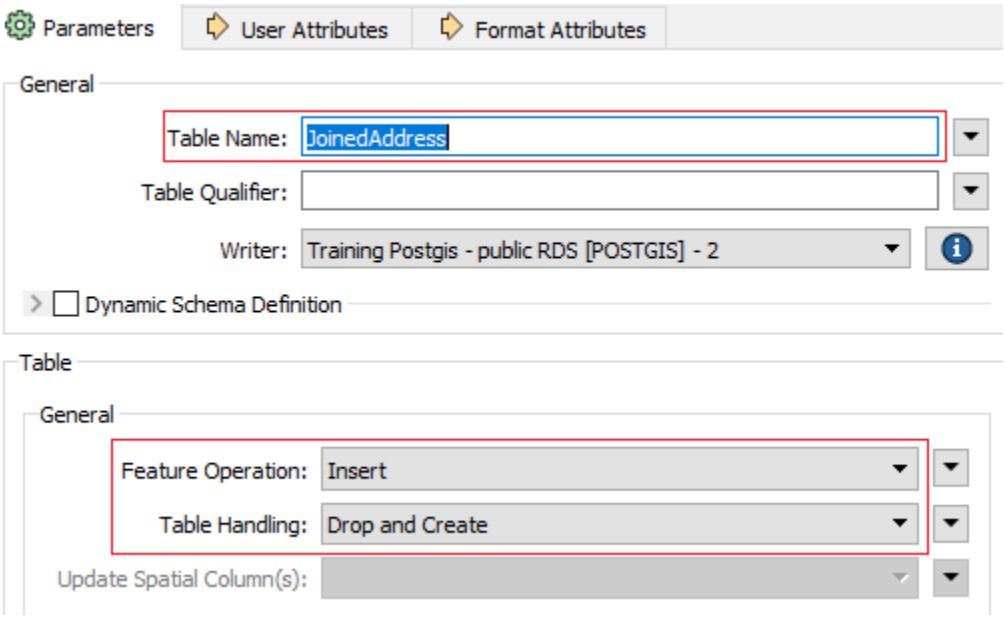
4. Add a PostGIS Writer

- Similar to adding a reader, type “PostGIS” then select the PostGIS format from the list of Writers.
- Set the writer format to PostGIS and connect the writer dataset to the same dataset as the reader (PostGIS Training Database). Set the Table Definition to Automatic and click OK.



5. Set the Table Name, Feature Operation, and Table Handling

- After the Writer Feature Type is added it will open the Writer parameters. Set the Table Name to "JoinedAddresses", by default the table will be placed in the "public" schema.
- Next, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



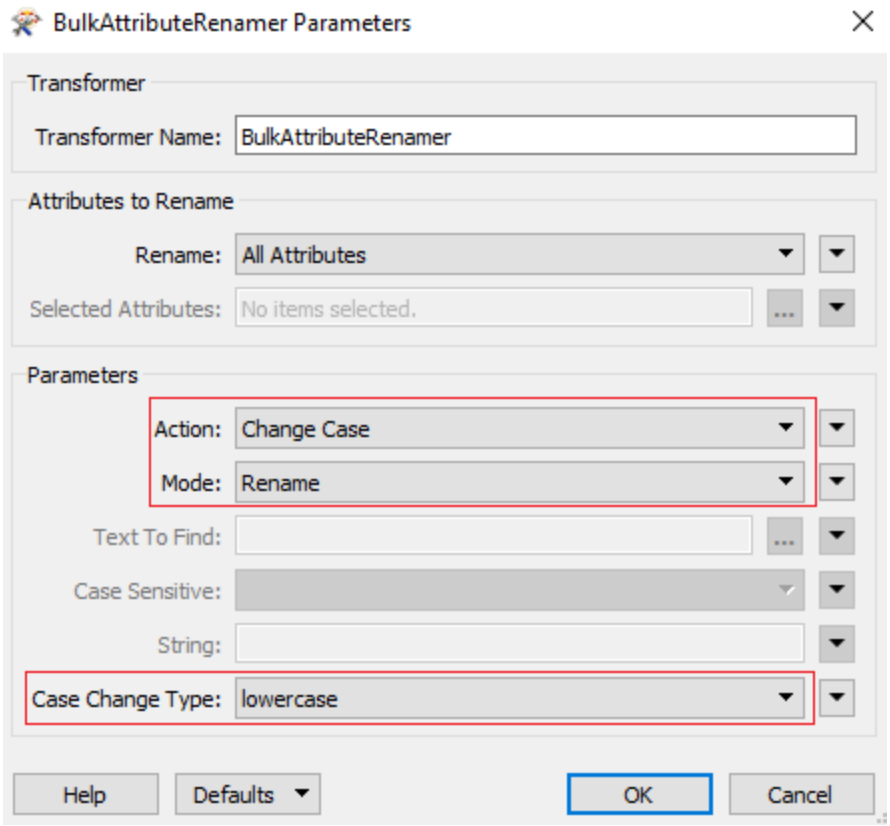
Note: if the table name matches an existing table the data will be overwritten.

6. Add a BulkAttributeRenamer

- Since you will be relating fields, you need to make sure the cases match on the joining fields. Add a BulkAttributeRenamer and connect it to the AddressPoints table.

7. Convert to Attributes to Lowercase

- Open the BulkAttributeRenamer Parameters and set the Action to Change Case and change the Case Change Type to lowercase. This will ensure the field names in the AddressPoints table match the field names in the VanHomes table.



Note: by clicking on the arrow on the reader, you will expose a list of attributes that have been read in. If you expand the attributes on both the AddressPoints and VanHomes reader you will notice that the AddressPoints table uses TitleCase whereas the VanHomes table uses lowercase.

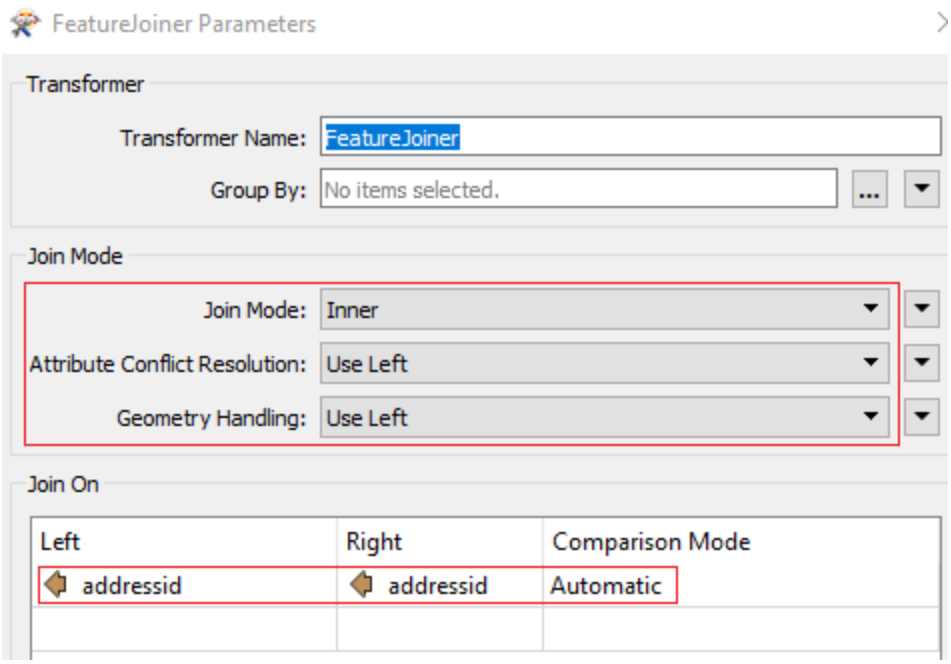
8. Add a FeatureJoiner

You will be performing an inner join so the position (left and right) of the inputs is not important since the output will only contain matched features; however, if you were performing a left join (matched features and all unmatched left features) the position is important otherwise you will have an unexpected result. For a more detailed explanation on how to use joining transformers, see the Working with Merging/Joining Transformers (How to choose the right one).

- Connect the output of the BulkAttributeRenamer to the Left parameter and connect the VanHomes table to the Right parameter of the FeatureJoiner. Once you have made the connections, open the FeatureJoiner Parameters.

9. Select the Attributes to Join On

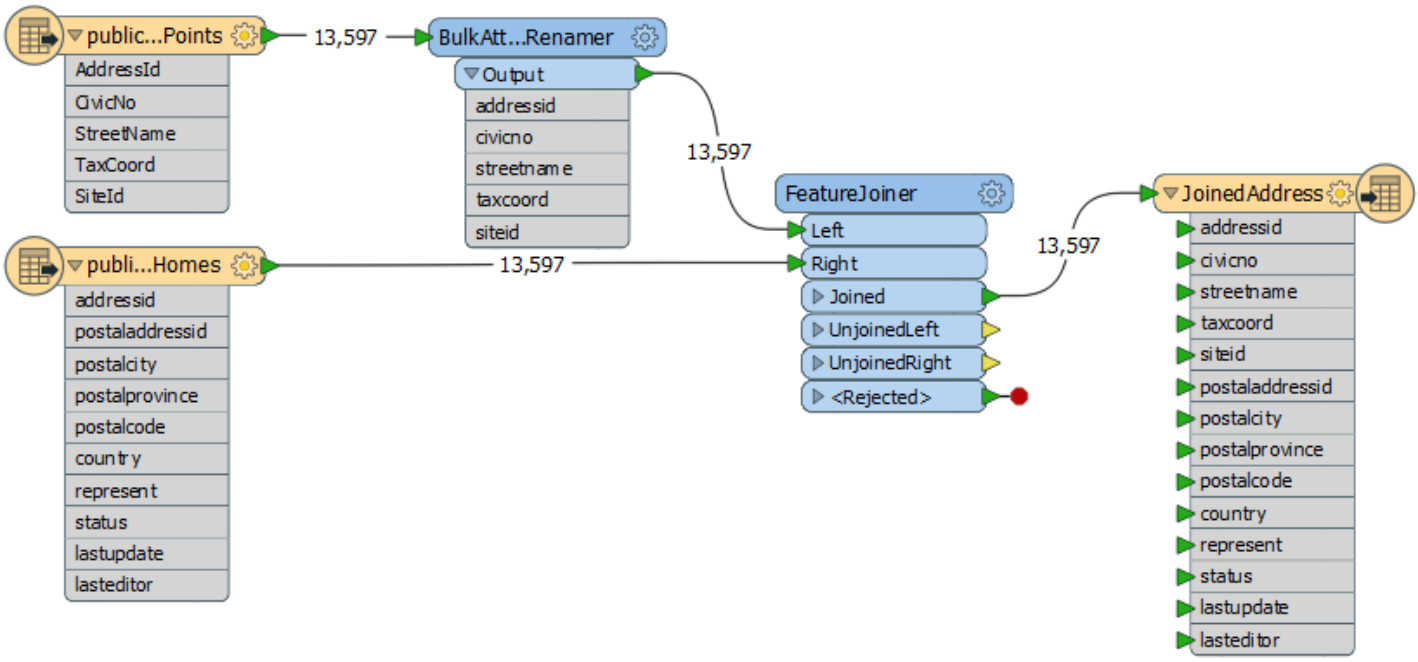
- Leave the default parameters for the Join Mode and set the Join On Left and Right to “addressid”. In this case, you are only interested in fields that are common in both tables. To learn more about database joins using the FeatureJoiner, see The FeatureJoiner Transformer article.



- Finally, connect the FeatureJoiner:Joined port to the PostGIS writer to complete the workspace.

10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the workspace, your database will have a new table (public.JoinedAddresses) with the joined fields. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected.

Results

Input

Table View

Training Postgis - public RDS [POSTGIS] - public.AddressPoints

	AddressId	CivicNo	StreetName	TaxCoord	SiteId
1	100161	1750	BEACH		
2	100162	1101	POWELL		
3	100178	1441	CREEKSIDE		

Table View

Training Postgis - public RDS [POSTGIS] - public.VanHomes

	addressid	postaladdressid	postalcity	postalprovince	postalcode	country	represent	status	lastupdate	lasteditor
1	193221	{1c55c207-5a3e-45...	Vancouver	BC	V6E 4V5	Canada	Owner	Other	20130707	GIS Technician
2	193213	{8afe5c37-e0a7-49f...	Vancouver	BC	V5Y 2Q7	Canada	Owner/Occupant	Retired	20130707	GIS Technician
3	193212	{6963b7db-653a-40...	Vancouver	BC	V6B 8E1	Canada	Other	Current	20130707	GIS Technician
4	193211	{67133025-d2f6-4c6...	Vancouver	BC	V5Y 4W5	Canada	Owner	Other	20130707	GIS Technician

Output

Table View

fmedata [POSTGIS] - JoinedAddress

	addressid	civicno	streetname	taxcoord	siteid	postaladdressid	postalcity	postalprovince	postalcode	country	represent	status	lastupdate	lasteditor
1	100161	1750	BEACH	<null>	<null>	{88e2ca07-300e-46...	Vancouver	BC	V6E 9T0	Canada	Other	Current	20130707	GIS Technician
2	100162	1101	POWELL	<null>	<null>	{69569d7b-7e44-42...	Vancouver	BC	V6A 0U9	Canada	Owner/Occupant	Current	20130707	GIS Technician
3	100178	1441	CREEKSIDE	<null>	<null>	{118181d5-355d-4b...	Vancouver	BC	V6J 7K2	Canada	Occupant	Current	20130707	GIS Technician
4	100179	3015	YUKON	<null>	<null>	{95215c5c-fc9d-403...	Vancouver	BC	V5Y 8H1	Canada	Other	Current	20130707	GIS Technician
5	100180	3025	YUKON	<null>	<null>	{3278d21a-7593-46...	Vancouver	BC	V5Y 0W9	Canada	Owner/Occupant	Current	20130707	GIS Technician

---

Exercise 2 - *Relate Data from a CSV File*

In exercise 2, you will be adding new columns to your table using the FeatureMerger. Rather than creating a column and assigning values, you will be appending longitude and latitude data from another file into your database table. You will be performing a join on fields that are common in both your database table and the CSV file that you are reading in (i.e. an inner join).

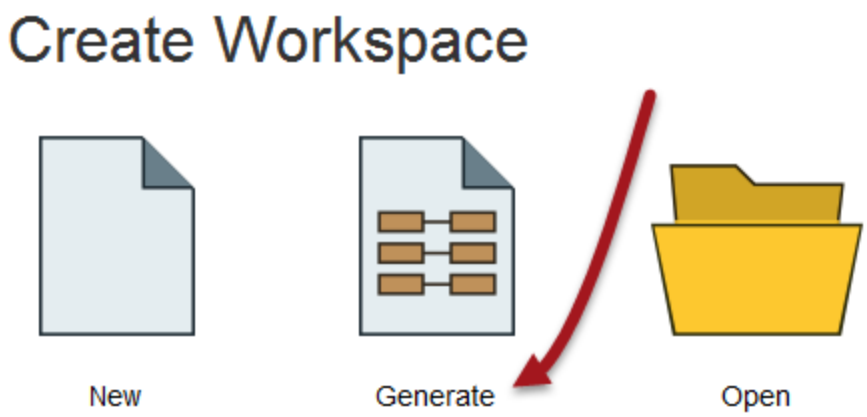
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Select Generate Workspace

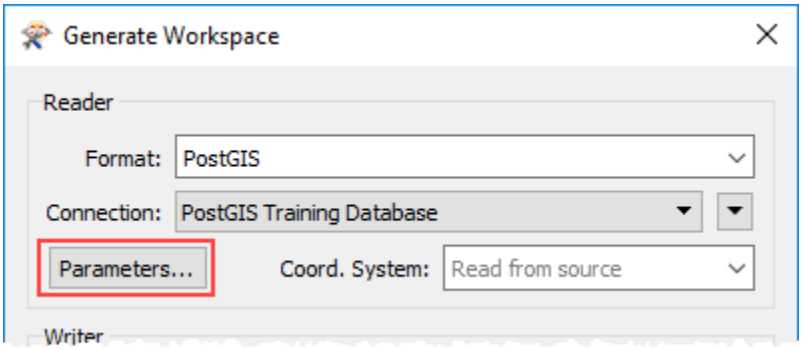
- In the Create Workspace part of the Start page select the option to Generate (Workspace). Alternatively, you can use the shortcut Ctrl+G.



For more information on Creating a Translation, see the [Desktop Basic Course Manual](#).

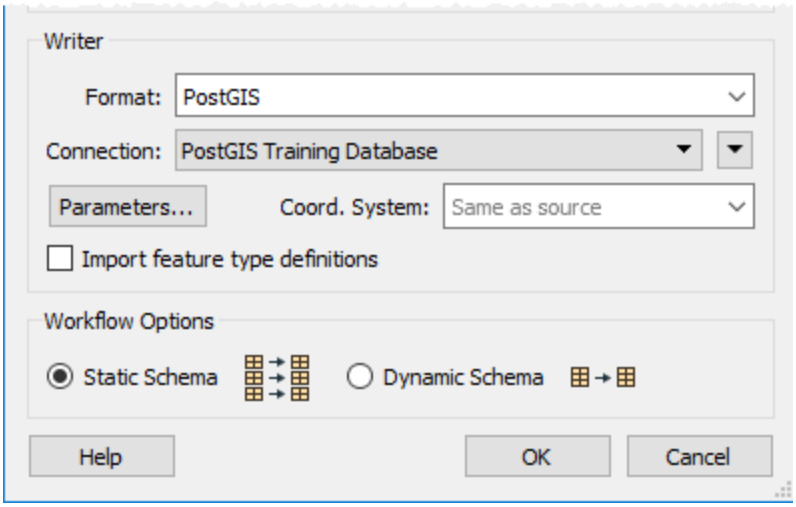
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Downtown_PublicArt table from the Table List.



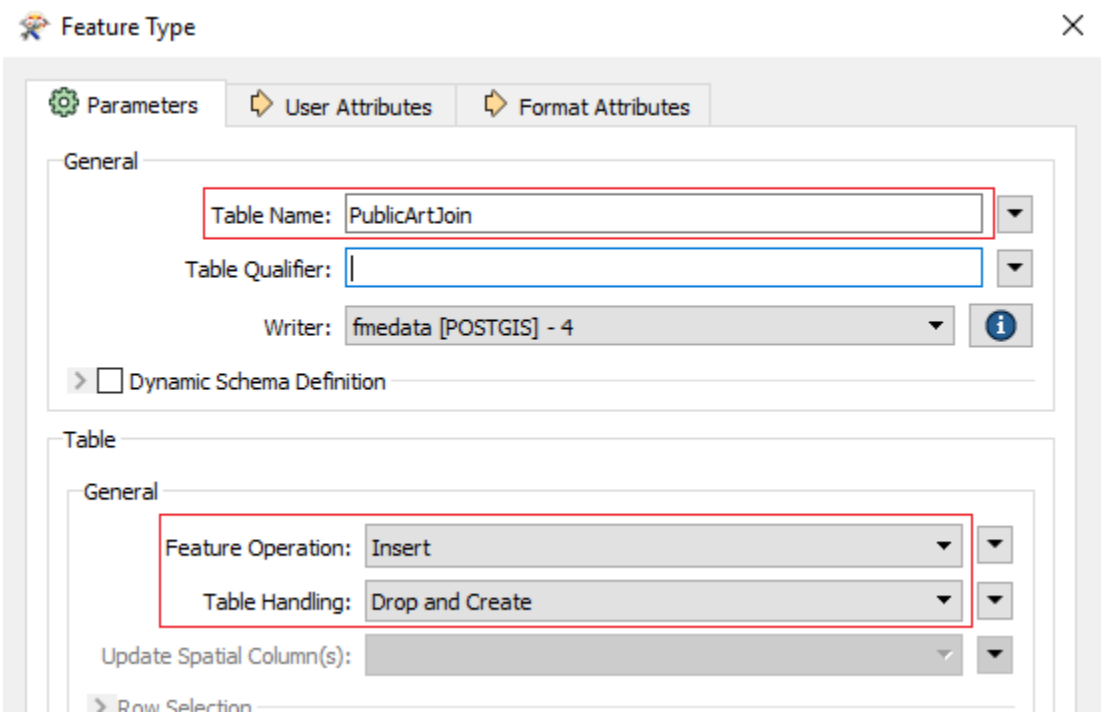
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

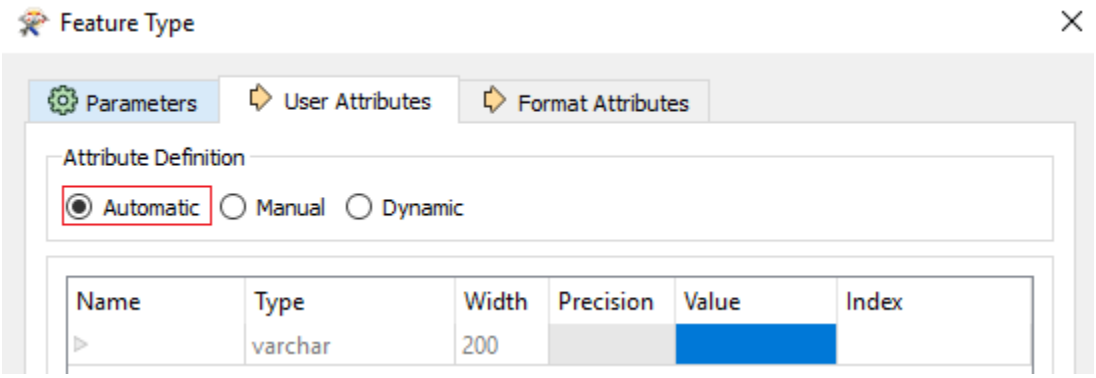
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to PublicArtJoin.
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when the table needs to be emptied and an update is made to the database schema. For example, this is used when you wish to update a table with new content and require a new column to be added to the table.

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.
- Click OK to accept the changes and close the Writer Parameters dialog.



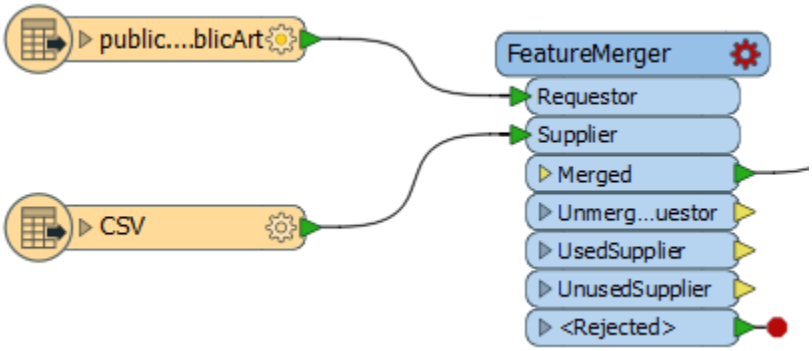
Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the [Documentation](#).

7. Add a CSV Reader

- Add a CSV reader to the canvas by typing "CSV" to bring up the list of FME Reader in the Quick Add Search. Select the CSV format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once selected, the Add Reader box will open. Click on the Dataset ellipsis and open the 3531-downtown CSV file from the sample data set then click OK.

8. Add a FeatureMerger

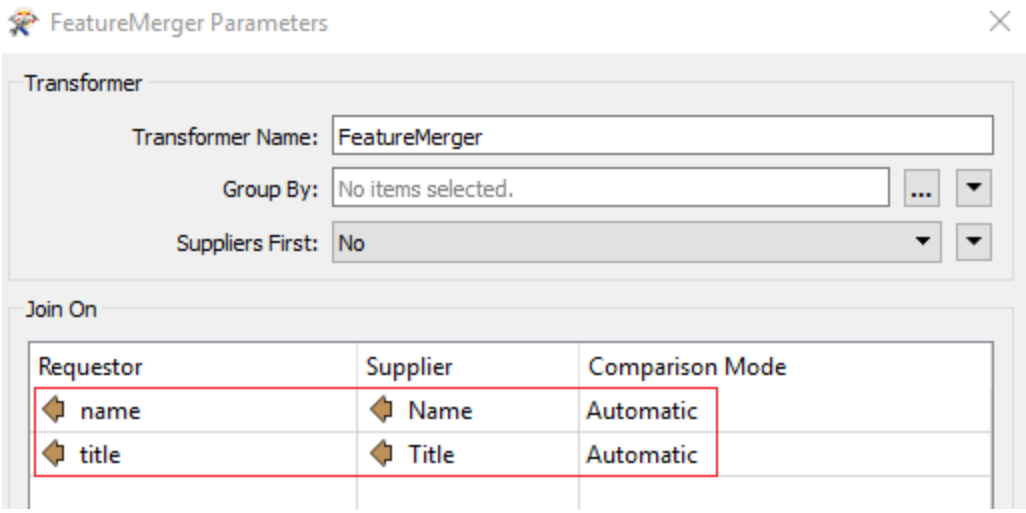
- Connect the PostGIS table to the Requestor port and the CSV to the Supplier port. The FeatureMerger is used to combine attributes and/or geometry from two different streams of features, based on a common key attribute value or expression.



To learn more about using Key-Based Transformers in FME, see the [Desktop Basic Course Manual](#).

9. Set the FeatureMerger Parameters

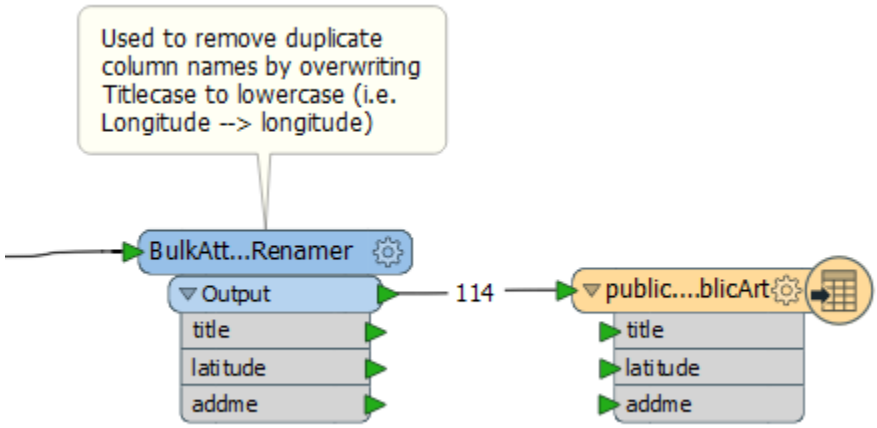
- Open the FeatureMerger Parameters dialog and set the Join On Requestor to "name" and "title" and the Supplier to "Name" and "Title". Leave all other parameters as default and click OK. This will match features based on both the name and title attributes.



10. Add a BulkAttributeRenamer

Only matched features pass the FeatureMerger which will result in duplicate rows (i.e. Name - from the CSV file, and name from the PostGIS table).

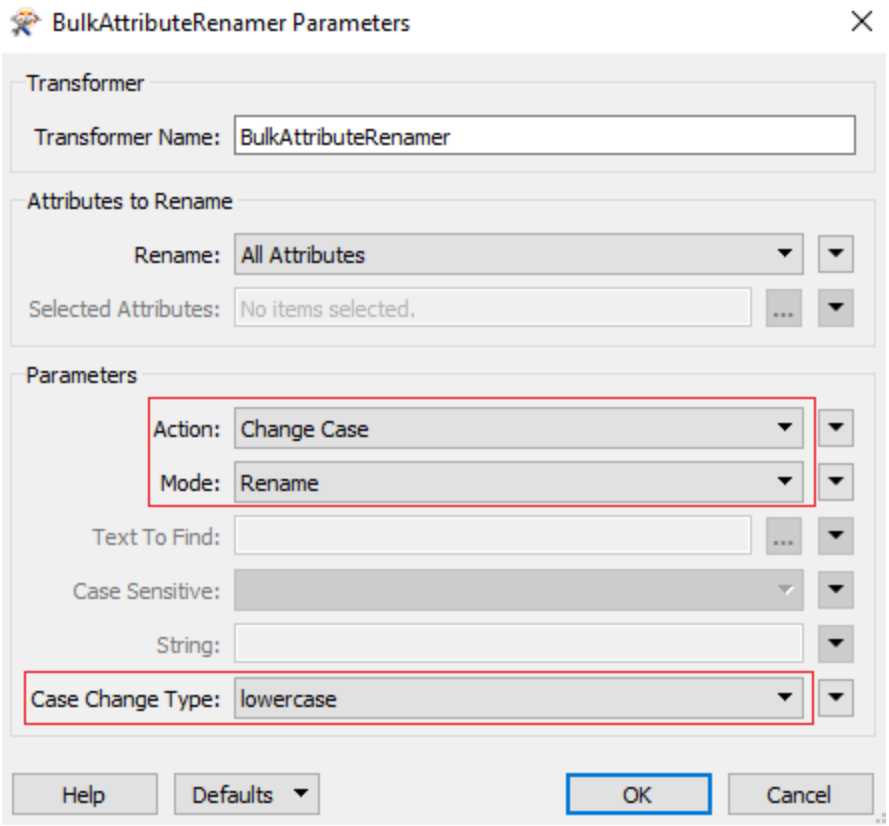
- Add a BulkAttributeRenamer to the canvas to automatically change the attributes to lowercase from Titlecase. This will also remove the duplicate column names (“Name” “name”).
- Connect the Merged port from the FeatureMerger to the BulkAttributeRenamer, then connect the output to the PublicArtJoin writer.



Note: you can also remove unwanted attributes using transformers such as the AttributeManager, AttributeRemover, or BulkAttributeRemover or by setting the Writer Attribute Definition to Manual and removing attributes. For more information on managing/removing attributes, see the [Desktop Basic Course Manual](#).

11. Set the BulkAttributeRenamer Parameters

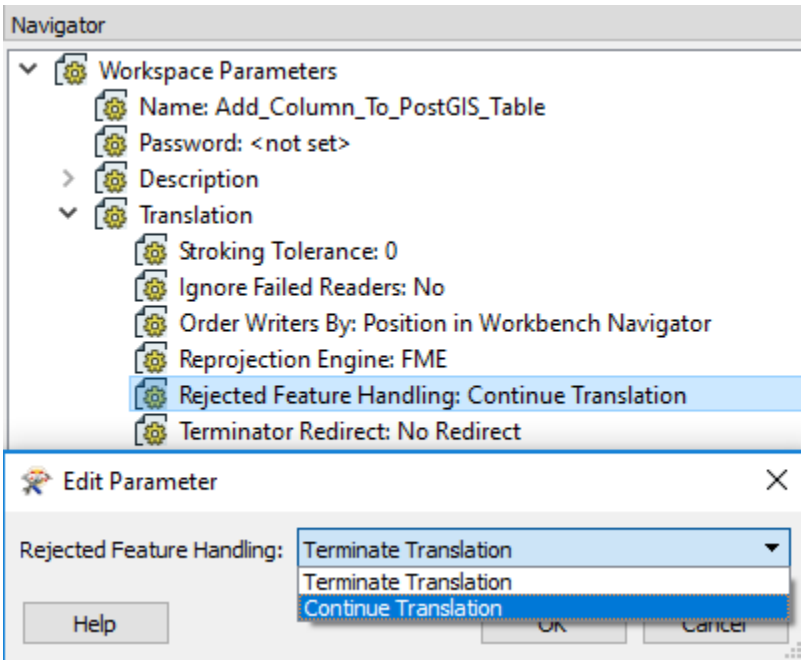
- Open the BulkAttributeRenamer Parameters dialog and set the action to Change Case and change the Case Change Type to lowercase.



12. Set the Workspace Parameters

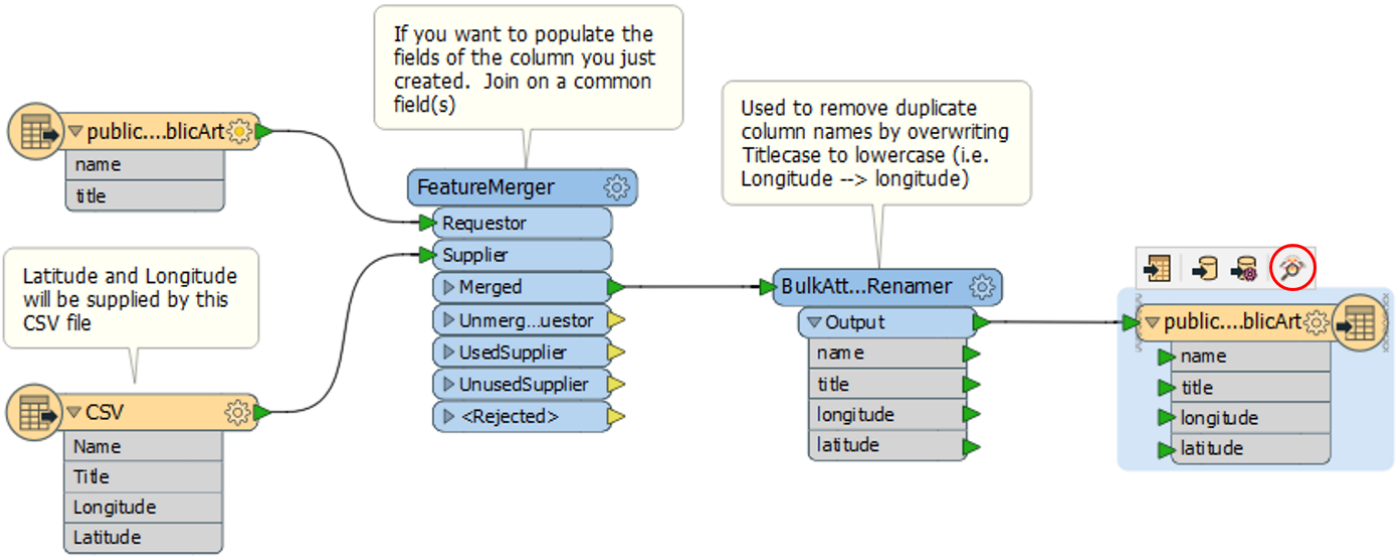
- Before you can run the workspace, you have to set the Workspace Parameters (found in the Navigator window) > Translation > Rejected Feature Handling to Continue Translation.

By default, the FME Workspace Parameter is set to Terminate Translation which will cause the Workspace to stop if any features are rejected by the FeatureMerger. You can learn more about rejected feature handling in the [Desktop Basic Course Manual](#).



13. Run the Translation

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, your table will have latitude and longitude columns. You can view the new table by selecting the writer feature type and clicking the Inspect button in the shortcut menu that appears when the writer is selected.

Results

Input

Table View

Table: Training Postgis - public RDS [POSTGIS] - public.Downtown_PublicArt

	name	title
1	Harbour Centre...	The Belonging ...
2	Harbour Centre...	The Belonging ...
3	Chinese Cultur...	China Gate
4	Vancouver Inter...	Moving Pictures
5	Shanghai Alley	Suan Phan: Aba...

Output

fmedata [POSTGIS] - PublicArtJoin

	name	title	longitude	latitude
1	Harbour Centre...	The Belonging ...	-123.110097741...	49.2837806793832
2	Harbour Centre...	The Belonging ...	-123.110097741...	49.2837806793832
3	Chinese Cultur...	China Gate	-123.103282272...	49.2797561341325
4	Vancouver Inter...	Moving Pictures	-123.125040050...	49.2770551706603

in any column

Transformers

- BulkAttributeRenamer - Renames attributes by adding or removing prefixes or suffixes, or replacing text using regular expressions or character strings.
- FeatureJoiner - Joins features by combining the attributes/geometry of features based on common attribute values, like a SQL join operation.
- FeatureMerger - Copies and merges the attributes/geometry from one feature (or multiple features) onto another feature (or multiple features).

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Creating a Polygon from Line Segments

Overview

Creating polygons in FME can be accomplished by using a polyline or line features, such as a street network, and the AreaBuilder transformer. Since a street network is a set of topologically connected linework, the AreaBuilder is able to create polygon features from lines that are connected by vertices/endpoints and form shapes (i.e. a closed polygon). However, areas can also be created from disconnected line features using the LineCloser.

Downloads

create-a-polygon-from-line-segments.fmwt

roadsgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this tutorial, you will create polygons for city blocks from a street network that consists of Arterial, Residential, Private, and Non-city streets using the AreaBuilder.

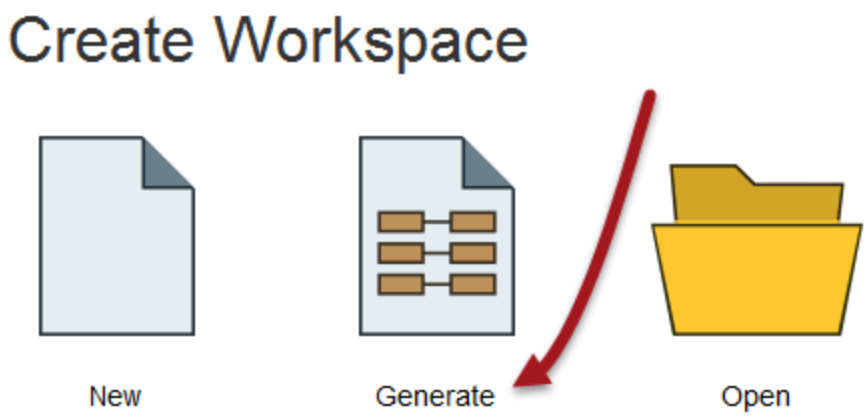
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Select Generate Workspace

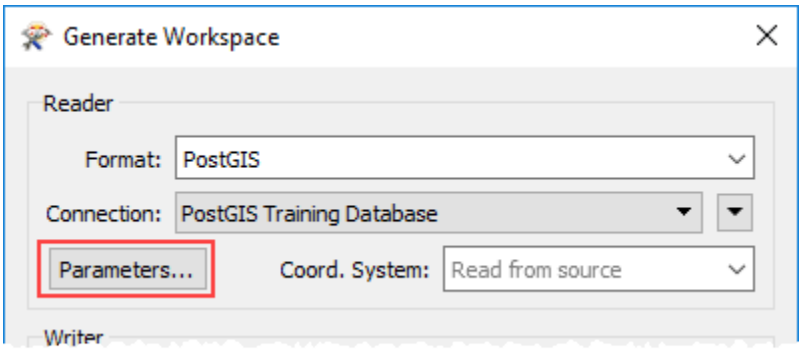
- In the Create Workspace part of the Start page select the option to Generate (Workspace). Alternatively, you can use the shortcut Ctrl+G.



For more information on Creating a Translation, see the [Desktop Basic Course Manual](#).

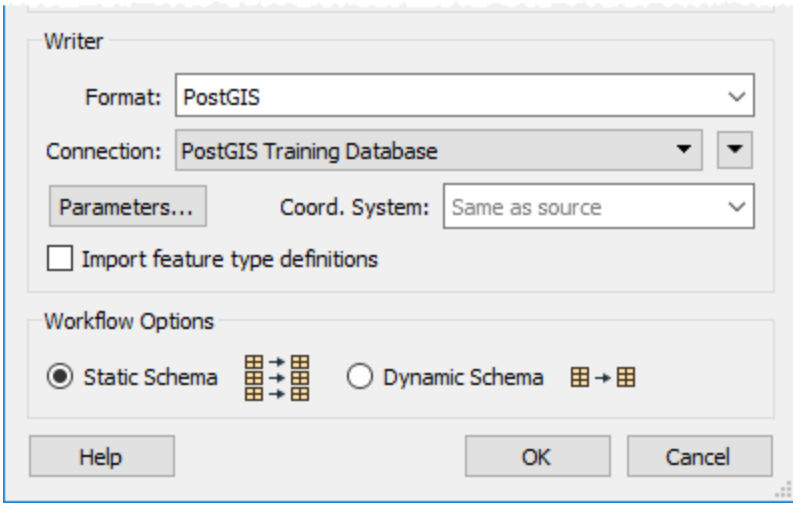
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the **Roads** table from the Table List.



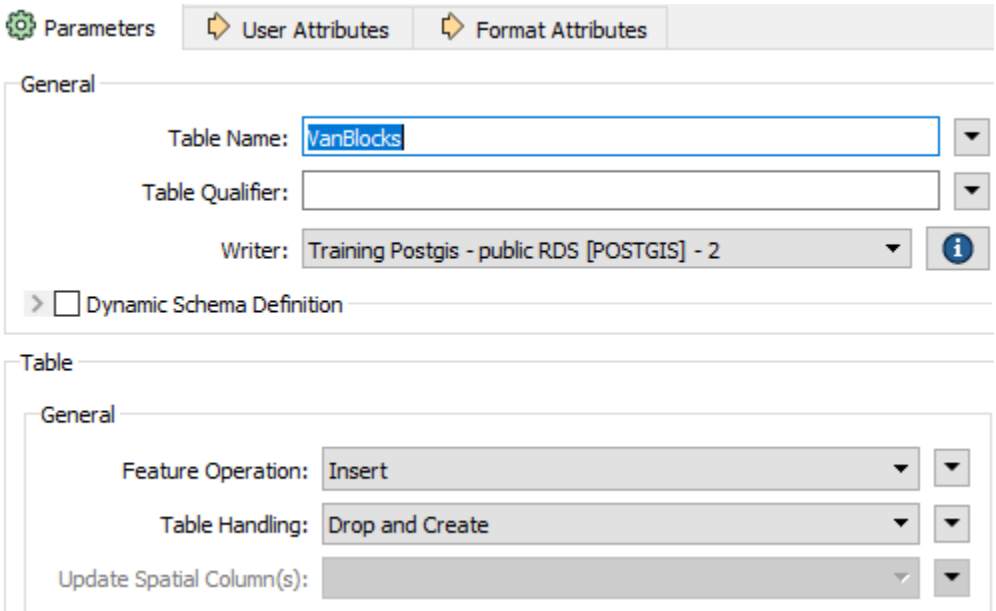
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

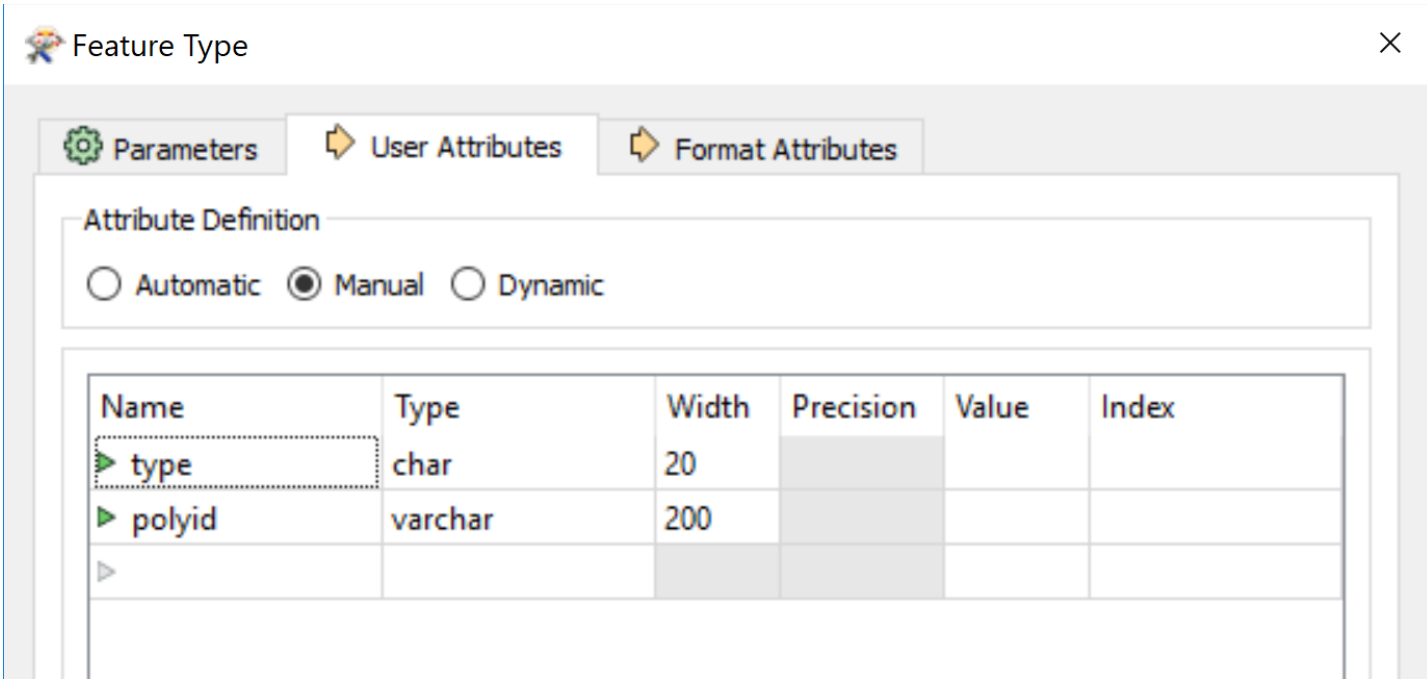
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to VanBlocks.
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the Documentation.

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Manual and remove all of the attributes except "**type**" and "**polyid**".
- Click OK to accept the changes and close the Writer Parameters dialog.



Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Documentation.

7. Add an Area Builder

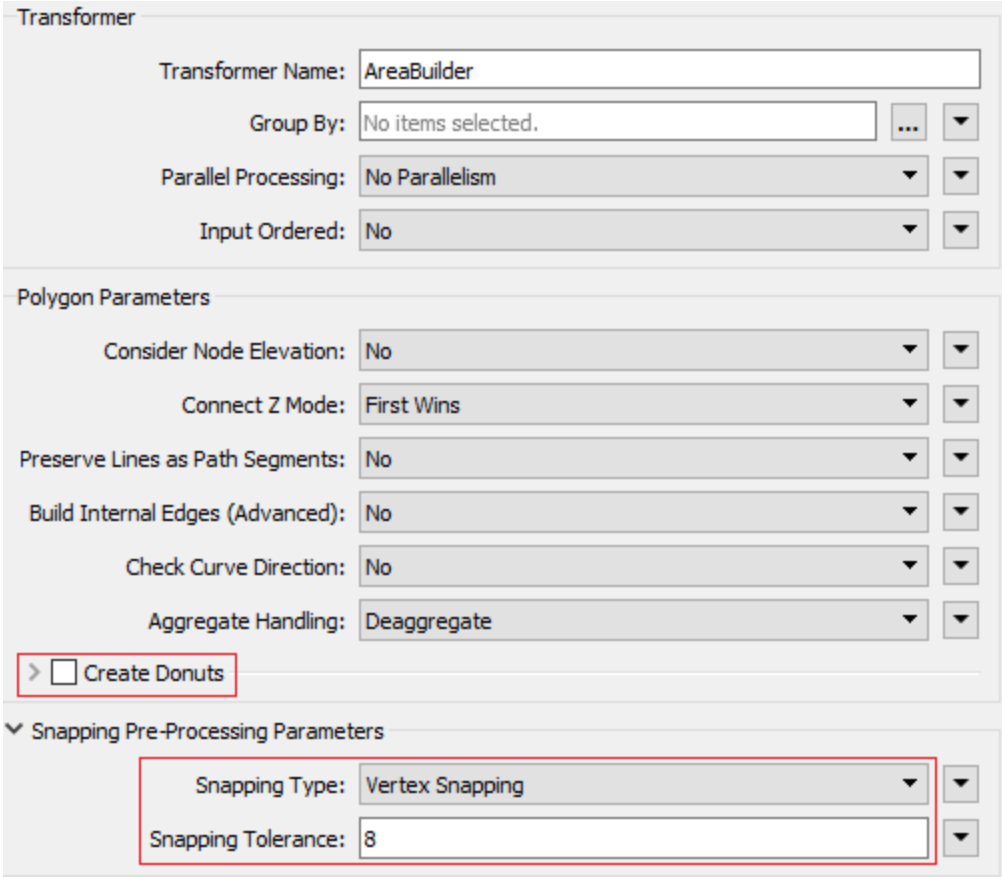
- Add an AreaBuilder to the canvas by selecting the connection between the reader and writer and typing "AreaBuilder" to bring up the list of FME Transformers in the Quick Add Search. Select the AreaBuilder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Note: by adding a transformer while the connection is selected, the ports will automatically be mapped to the first available input and output port.

8. Set the Snapping Tolerance

- Open the AreaBuilder Parameter, in the Polygon Parameters section, deselect Create Donuts.

Before closing the parameters, adjust the snapping pre-processing parameters to ensure the street network is topologically correct.

- Expand the Snapping Pre-Processing Parameters in the AreaBuilder Dialog to reveal additional parameters. Set the Snapping Type to Vertex Snapping and the Snapping Tolerance to 8. For more information on tolerance parameters, see the Desktop Basic Course Manual.
- Leave all other parameters as default, then click OK to accept the changes and close the AreaBuilder Parameters. The AreaBulider parameters should match the screenshot provided below:



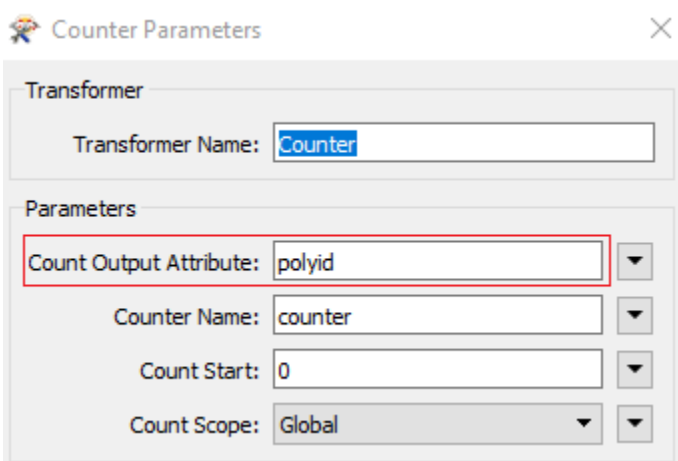
Note: snapping tolerance is measured in map units which in this case is meters.

9. Create PolyID

- Add a Counter after AreaBuilder and connect it to the AreaBuilder:Area port. This will give each of the newly created polygons a unique ID.

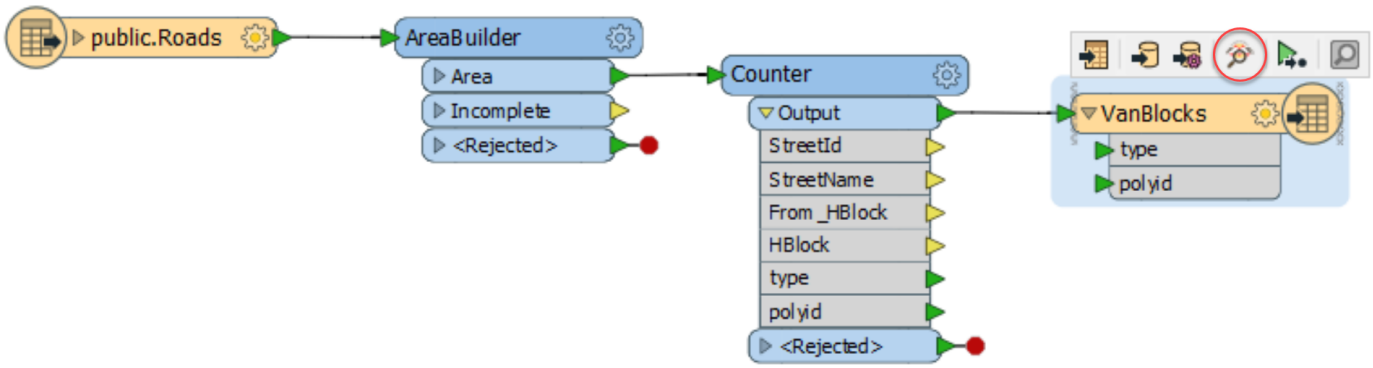
10. Set the Count Output Attribute

- Open the Counter parameters and change the Count Output Attribute to “polyid”. Leave all other parameters as default, then click OK to accept the changes and close the Counter Parameters.



11. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, you can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected.

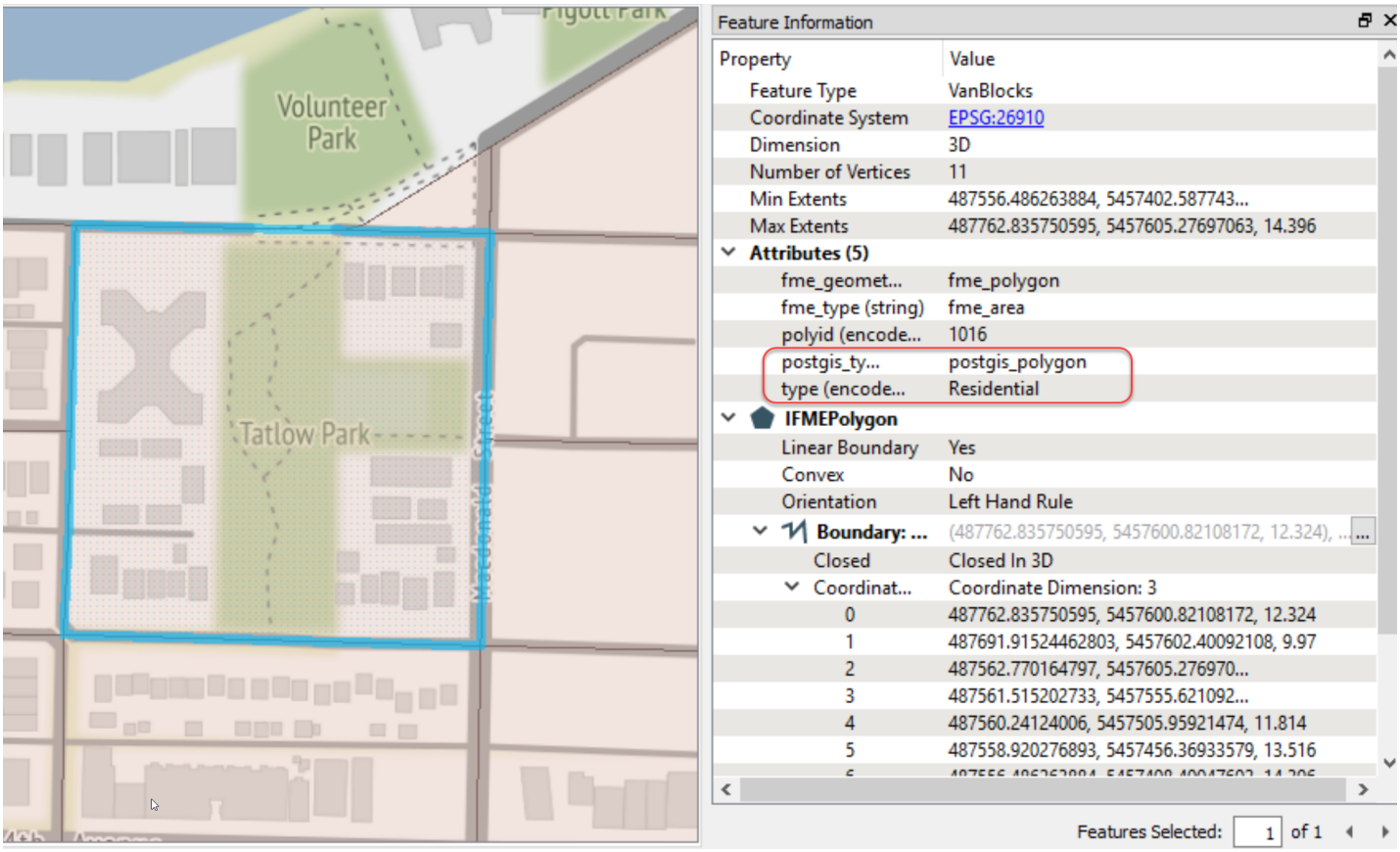
Results

Input

Feature Information

Property	Value
Feature Type	public.Roads
Coordinate System	EPSG:26910
Dimension	3D
Number of Vertices	2
Min Extents	487560.24124006, 5457505.959214...
Max Extents	487561.515202733, 5457555.621092...
Attributes (8)	
fme_geomet...	fme_line
fme_type (string)	fme_line
From_HBlock (...	1700
HBlock (encode...	1700 Bayswater St
postgis_ty...	postgis_linestring
StreetId (encode...	13578
StreetName (en...	Bayswater St
type (encode...	Residential
IFMELine (...	(487561.515202733, 5457555.62109.. ...
Closed	No
Coordinates (2)	Coordinate Dimension: 3
0	487561.515202733, 5457555.621092...
1	487560.24124006, 5457505.959214...

Output



Transformers

- AreaBuilder- Takes a set of topologically connected linework and creates topologically correct polygon features where the linework forms closed shapes.
- Counter - Adds a numeric attribute to a feature and assigns a value. Each subsequent feature passing through the transformer receives an incremented value – in other words, the Counter is counting the features.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Correcting Topological Errors

Overview

In order to correct topology errors, you should first test to which features break the appropriate topology rule for your dataset. In this article, you will learn how to use FME to identify topological errors using the SpatialFilter. To see more data validation scenarios, check out the Data Validation and QA with FME tutorial.

Downloads

correcting-topology-errors.fmwt

bikepathtopologygdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise 1a - *Testing for Topology Errors on Linear Features*

In this exercise, you will test for topological errors in a linear dataset. You are testing your dataset because you want to ensure the dataset was digitized correctly and bike path segments are snapped together.

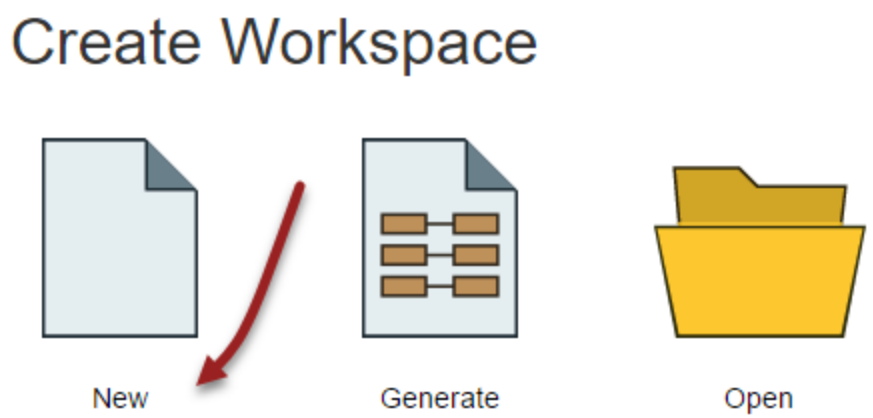
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

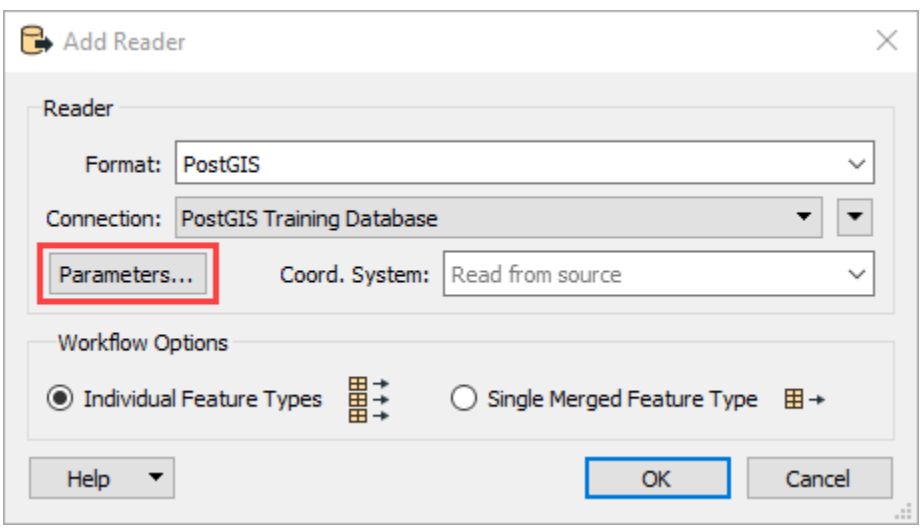
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to create a new workspace.



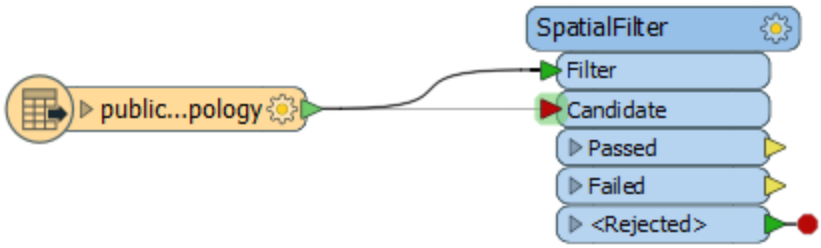
3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the workspace, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the BikePathTopology table from the Table List.



4. Add a SpatialFilter

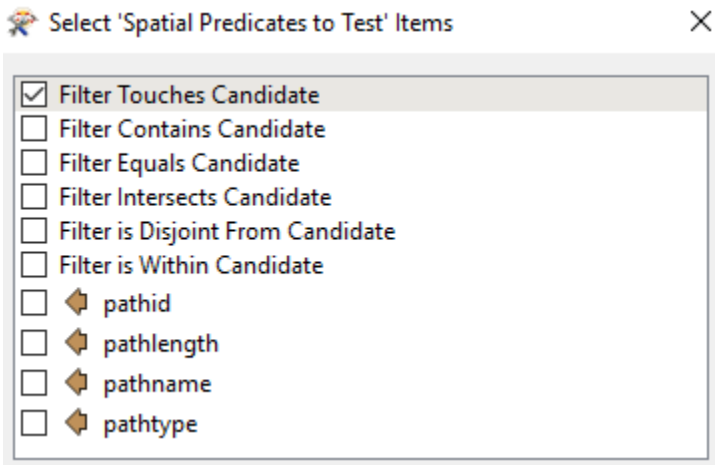
- Add a SpatialFilter to the canvas by typing “SpatialFilter” to bring up the list of FME Transformers in the Quick Add Search. Select the SpatialFilter from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the BikePathTopology Reader to both the SpatialFilter:Filter and Candidate ports.



5. Set the Spatial Predicate to Test

- Once the SpatialFilter has been added, double-click the SpatialFilter or click on the gear icon to open the transformer parameters dialog.
- Click the Spatial Predicates to Test ellipsis to open the test conditions and select the **Filter Touches Candidate** predicate then click OK. Please ensure you have unselected all other tests including Filter Intersects Candidate.

You will use the touches predicate to see which lines have been snapped together and which lines are disjoint.



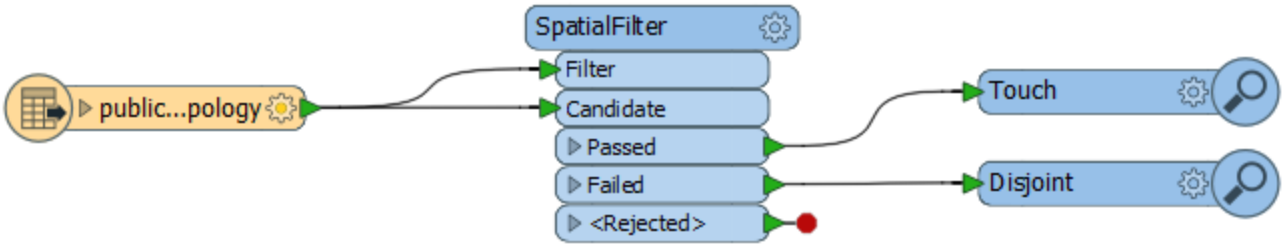
6. Add Two Inspectors

Next, you will add two Inspectors to the workspace.

- Add the first Inspector and open the transformer parameters.
  - Set the Transformer name to "Touch" then click OK to accept the changes
  - Connect the Touch Inspector to the SpatialFilter:Passed output port.
- Add a second Inspector, rename it "Disjoint" and connect to the SpatialFilter:Failed output port.

9. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the workspace, the output will automatically be displayed in the FME Data Inspector. Notice 12 line features are not topologically correct. You will learn how to correct these features in exercise 1b using the same workspace. Please save your workspace. Do not close this workspace as exercise 1b will continue where you left off.

Results

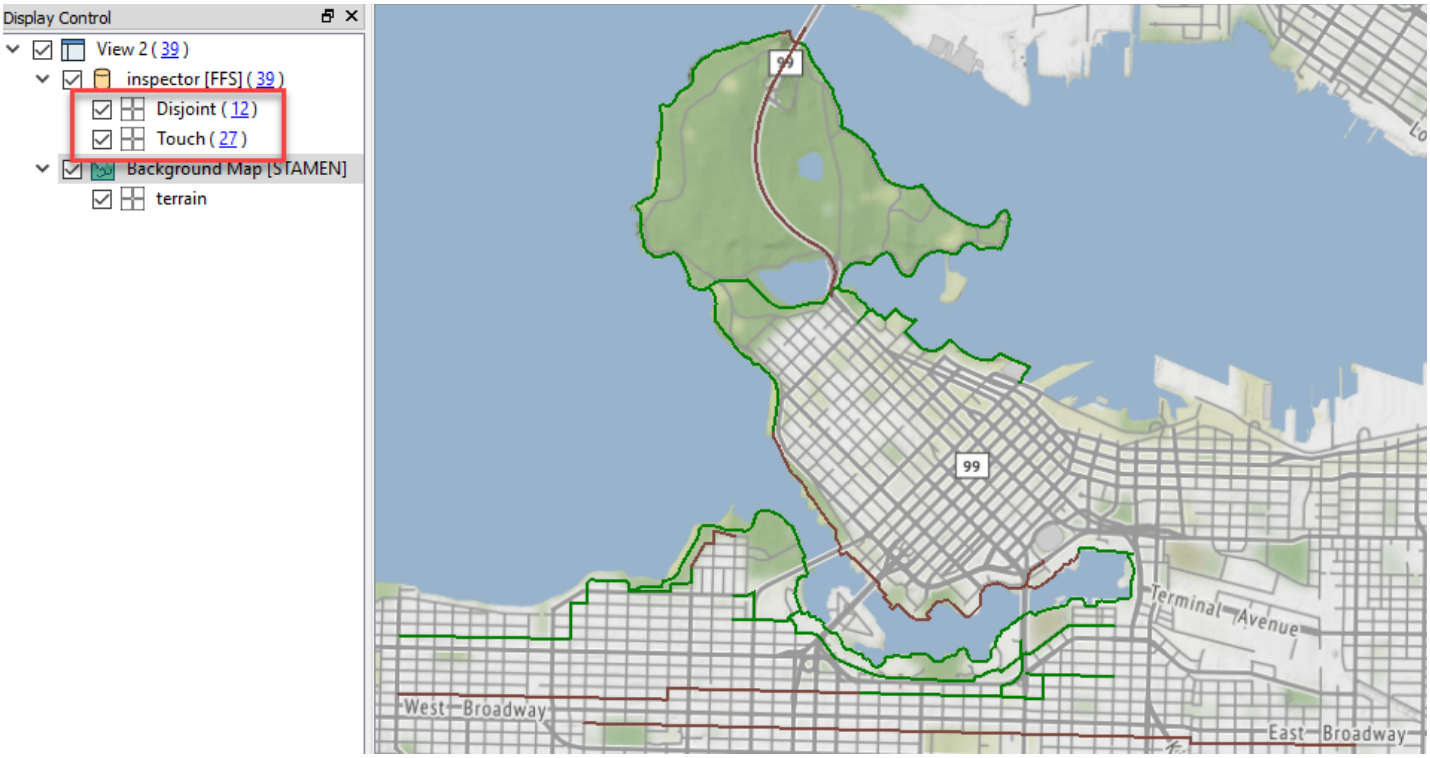
Input

Table View

Training Postgis - public RDS [POSTGIS] - public.BikePathTopology

	pathid	pathname	pathlength	pathtype
1	7	Seaside	31669	L
2	7	Seaside	31669	L
3	13	1st Ave	3998	L
4	13	1st Ave	3998	L

Output



Exercise 1b - *Correcting Topology Errors*

In exercise 1a, you tested for topological errors in a linear dataset by checking for disjoint lines. In this exercise, you will be correcting the disjoint features (identified by the <missing> predicate attribute) using the Snapper transformer. The goal of this exercise is to eliminate errors that were introduced by not snapping line segments to vertices/endpoints.

Instructions

1. Tidy up the Workspace

Since you are building off of the previous exercise, the workspace could start getting a little cluttered. To prevent this from happening, you will learn how to use bookmarks to keep the canvas organized.

- Click on the SpatialFilter and both the Touch and Disjoint Inspectors while pressing either the Shift or CTRL key, this will allow you to select multiple items on the workspace. Alternatively, you can click and drag your mouse to perform a multi-select.
- Once selected, right click on one of the selected items and choose Create Bookmark from the drop-down menu. Alternatively, you can use the Bookmark Shortcut (CTRL+B).
- Name the Bookmark "Topology Inspector"

Using bookmarks is a way to define areas of the workspace for easy access. Collapsible Bookmarks are available in FME versions 2018+. For more information on using collapsible bookmarks, see the Desktop Basic Course Manual.

2. Disable the Topology Inspector Bookmark

- Rather than selecting both of the transformers in the bookmark, simply right-click anywhere in the

bookmark and select Disable All Objects in the Bookmark from the drop-down menu.

3. Add a Snapper

Since the SpatialFilter is testing for touching bike path segments, you want to correct the segments that are failing this test.

- Connect the PostGIS reader to the Snapper. This transformer is used to snap bike path segments based on a snapping tolerance that you will define in the next step.

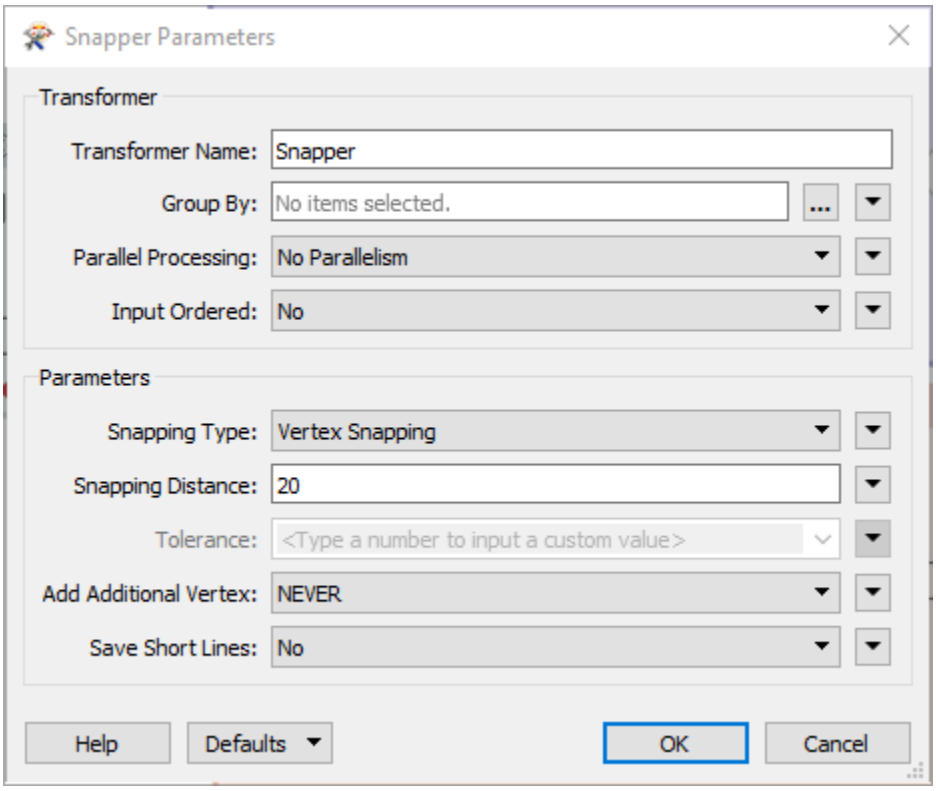
4. Set the Snap Tolerance

- Open the Snapper parameters and set the Snapping Type to Vertex Snapping.

Vertex snapping will snap line vertices together if their distances are within the specified tolerance. For more information on tolerance parameters, see the Desktop Basic Course Manual.

- Next, set the snapping distance to 20.

Note: The snapping distance is measured in map units which are meters in this case.



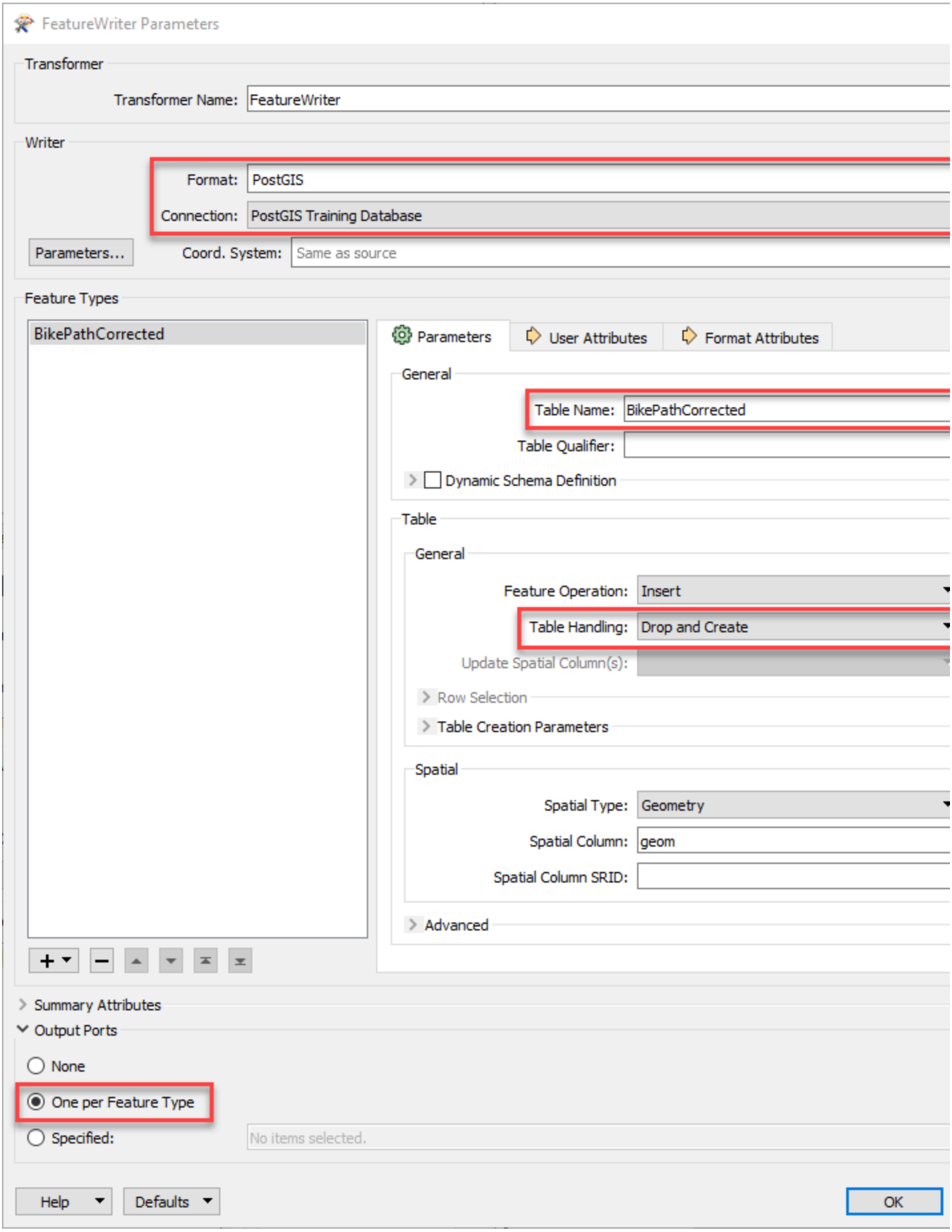
5. Add a FeatureWriter

The FeatureWriter can be used instead of a conventional FME writer. Using the FeatureWriter allows post-processing of writer results. In this case, we will use the FeatureWriter so we can re-test the written features using the same logic to confirm that our features have in fact been corrected.

- Add a FeatureWriter to the canvas and connect it to the Snapped and Untouched output ports from the Snapper.

6. Set the Table Name, Feature Operation, and Table Handling

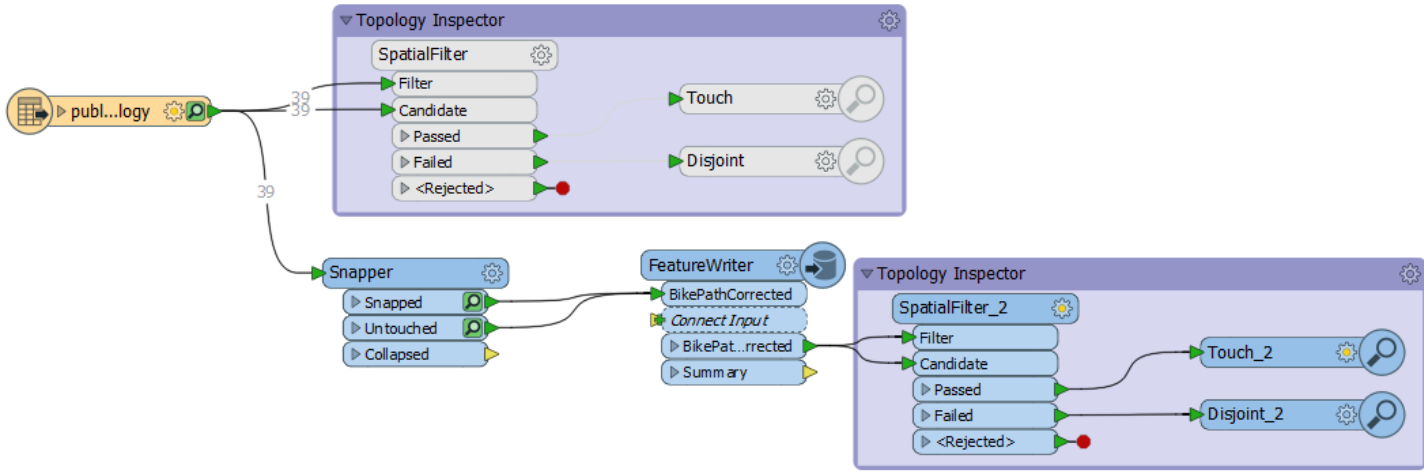
- In the FeatureWriter Parameters, set the Format to PostGIS and the connection to the same dataset as the reader (PostGIS Training Database).
- Next, in the General section, set the Table Name to "BikePathCorrected".
- Then, in the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.
- Switch to the User Attributes tab and ensure the Attribute Definition is set to Automatic.
- Lastly, expand the Output Ports section at the bottom of the parameters dialog and set the Output Port parameter to "One per Feature Type".



7. Re-Test the Line Features

Since we have already created a spatial test for our features, we can simply copy and paste the existing transformers to reuse them.

- Click and drag your mouse to select the Topology Inspector bookmark and all of its contents, then right click the top bar of the bookmark and select duplicate. Alternatively, you can use the shortcut CTRL+D while the bookmark and transformers are selected.
- One duplicated, place the new bookmark after the FeatureWriter.
- Similar to the first spatial test, connect BikePathCorrected output port from the FeatureWriter to both the Filter and Candidate ports on the SpatialFilter_2.
- Lastly, enable the objects in the new bookmark. Your final workspace should look similar to the screenshot below:



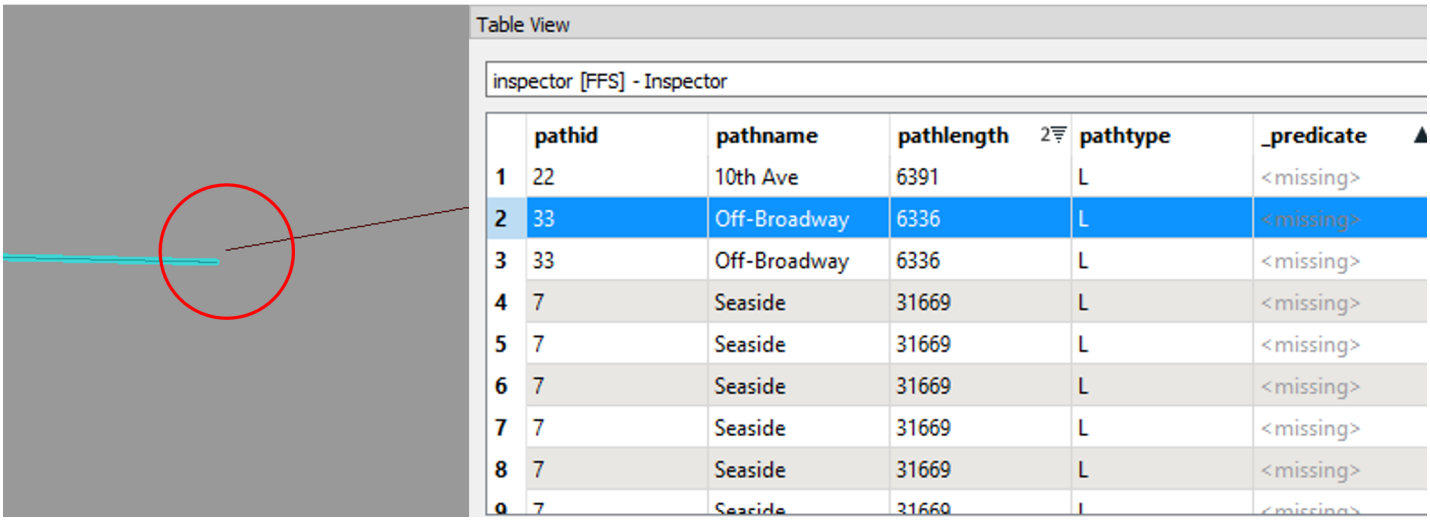
8. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. After the translation has successfully run, view the output in the FME Data Inspector by selecting the writer, then clicking the Inspect button on the shortcut menu that appears above the writer.

When you view the data in the FME Data Inspector you will notice that the number of disjoint features has been reduced from 12 to 1 by snapping line vertices.

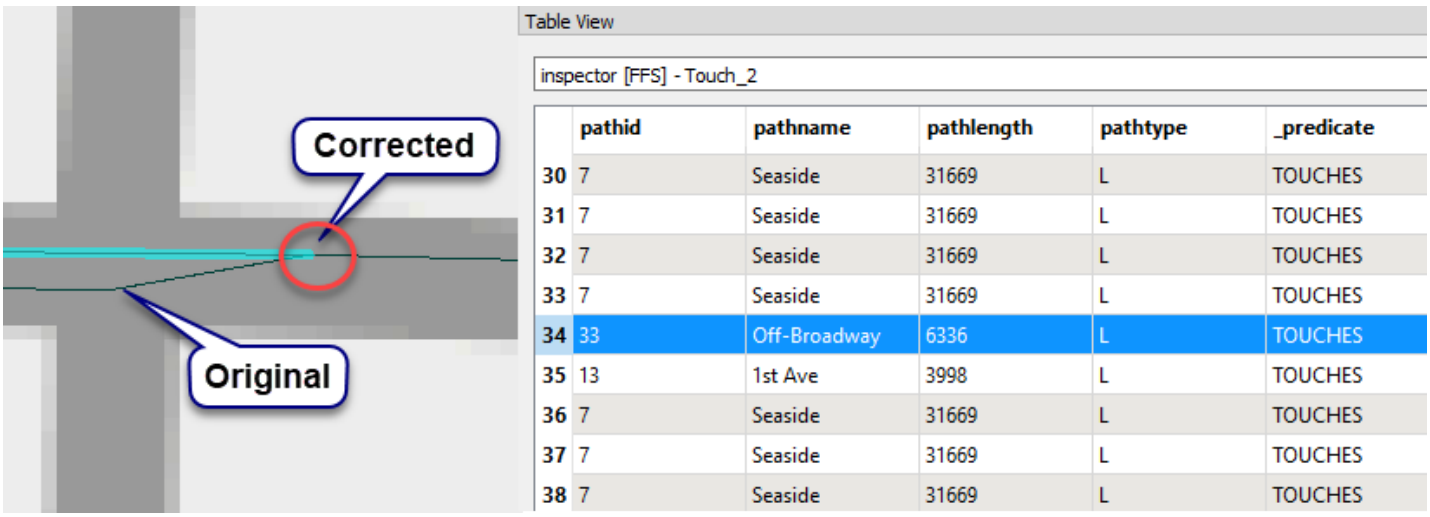
Results

Input



	pathid	pathname	pathlength	pathtype	_predicate
1	22	10th Ave	6391	L	<missing>
2	33	Off-Broadway	6336	L	<missing>
3	33	Off-Broadway	6336	L	<missing>
4	7	Seaside	31669	L	<missing>
5	7	Seaside	31669	L	<missing>
6	7	Seaside	31669	L	<missing>
7	7	Seaside	31669	L	<missing>
8	7	Seaside	31669	L	<missing>
9	7	Seaside	31669	L	<missing>

Output



	pathid	pathname	pathlength	pathtype	_predicate
30	7	Seaside	31669	L	TOUCHES
31	7	Seaside	31669	L	TOUCHES
32	7	Seaside	31669	L	TOUCHES
33	7	Seaside	31669	L	TOUCHES
34	33	Off-Broadway	6336	L	TOUCHES
35	13	1st Ave	3998	L	TOUCHES
36	7	Seaside	31669	L	TOUCHES
37	7	Seaside	31669	L	TOUCHES
38	7	Seaside	31669	L	TOUCHES

Transformers

- Inspector- Sends features to the FME Data Inspector for display.
- FeatureWriter - Writes features to any FME supported writer.
- Snapper - Brings lines, segments, endpoints or vertex points of features together if they are within a certain distance of each other and (optionally) if they have one or more attributes in common.
- Sorter - Sorts features by a selected attribute's value.
- SpatialFilter - Filters point, line, area, and text features based on spatial relationships.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Importing Database Tables, Raster Data, Vector Data

Overview

In this exercise, you will learn how to import data into your database and export data from your database using FME. It's a simple drag and drop process that supports over 400 data types; however, this tutorial will only provide examples of importing/exporting data from 3 categories: database tables, raster data, and vector data.

Downloads

import-export.fmwt

vancouver-blocks-polygon.zip

postaladdress.zip

06-07-lm.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise 1 - *Copying Tables*

In this scenario, you will learn how to make a copy of an existing database table.

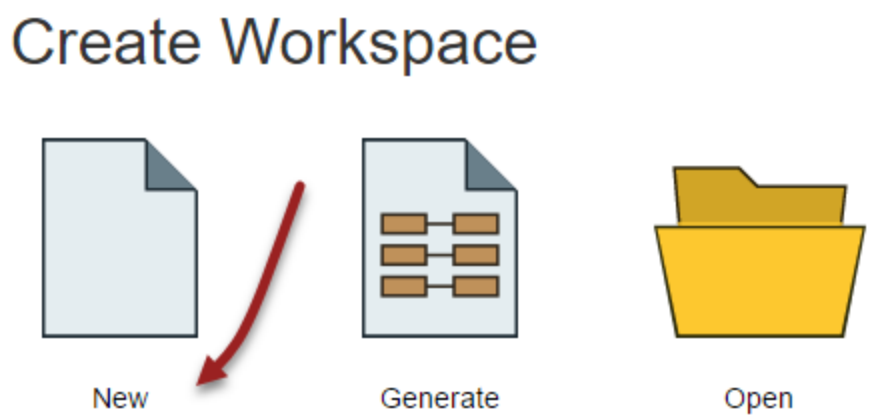
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

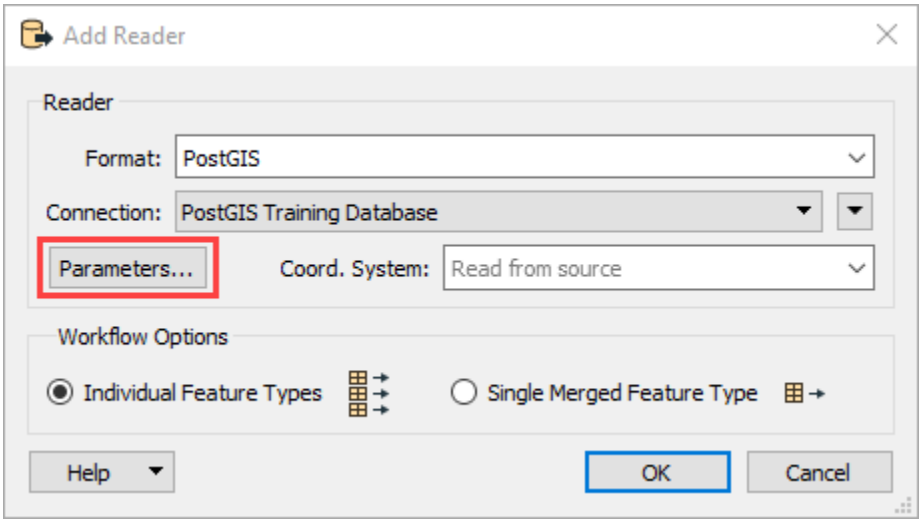
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



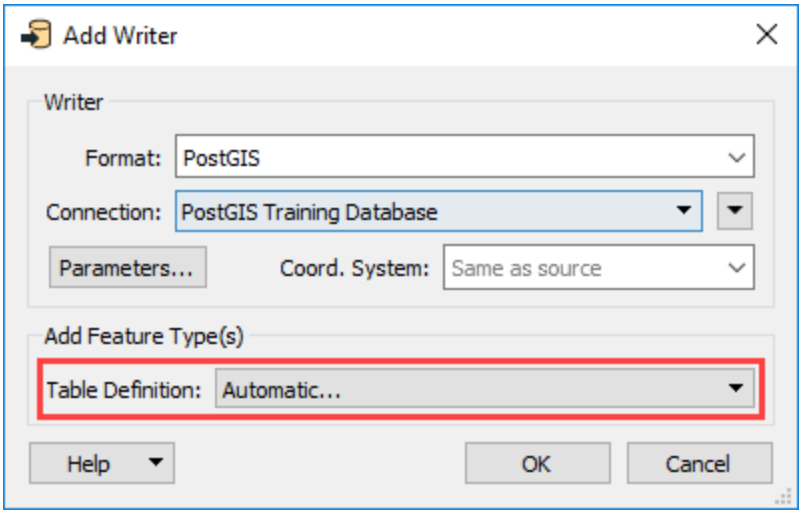
3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the PostalAddress table from the Table List.



4. Add a PostGIS Writer

- Similar to adding a reader, type “PostGIS” then select the PostGIS format from the list of Writers.
- Set the Writer Dataset to the PostGIS Training Database.
- Set the Table Definition to Automatic and click OK.

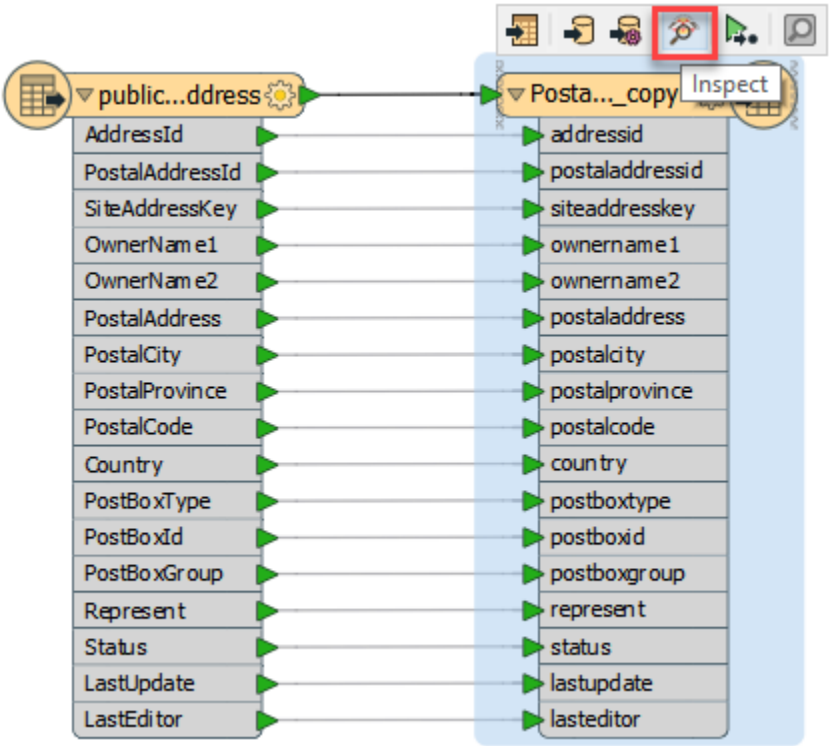


5. Set the Table Name

- After the Writer Feature Type is added it will open the Writer parameters. By default, the table will be placed in the "public" schema.
- Set the Table Name to "PostalAddress_copy"
- Leave all other Writer Feature Type Parameters as default and click OK.
- Lastly, connect the Reader Feature Type to the Writer Feature Type.

6. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.

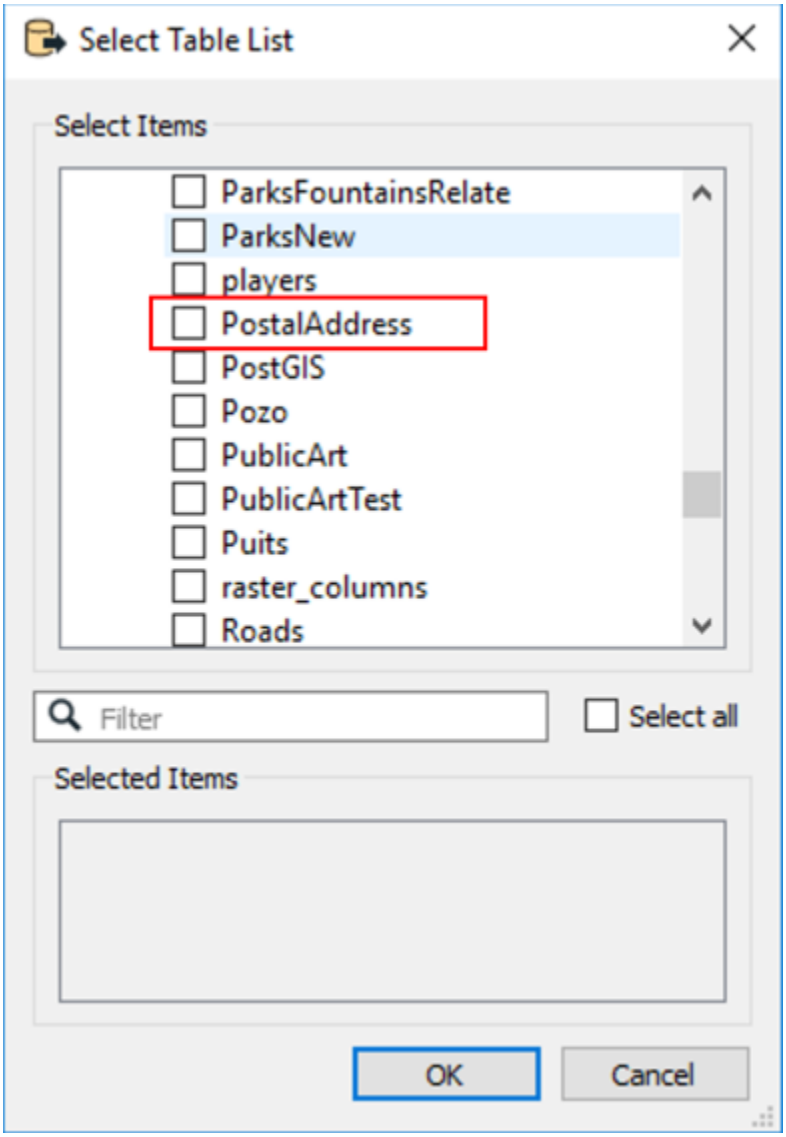


Alternatively, if you want to export the table as a CSV, swap out the PostGIS Writer for a CSV writer. This would result in your PostGIS table being exported to the CSV format. Similarly, if you have a CSV file that you want to be created as a table in your PostGIS database, replace the PostGIS Reader with a CSV Reader.

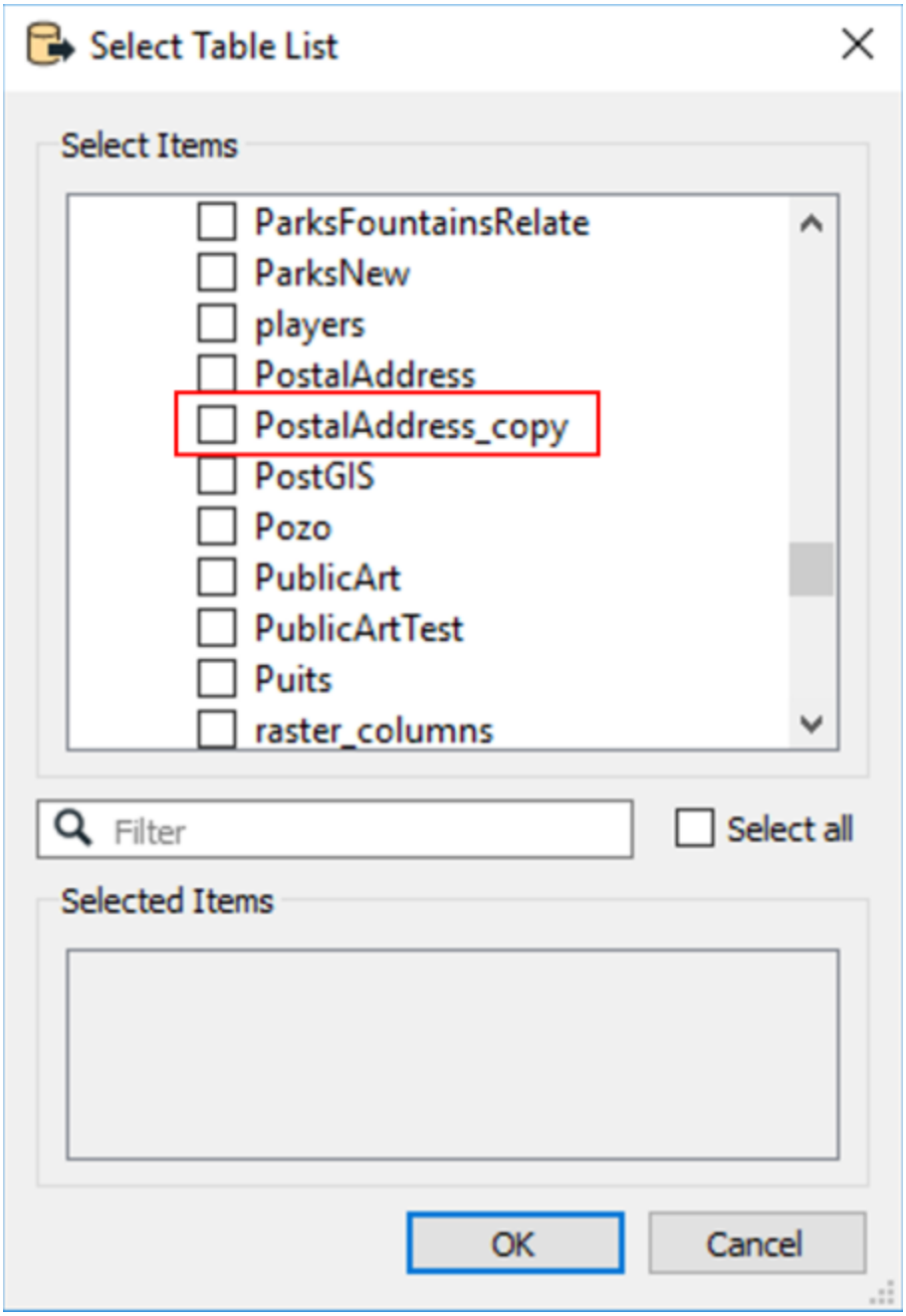
This would result in your CSV file being imported into PostGIS as a new table.

Results

Input



Output



Exercise 2 - *Exporting Raster Data*

In this scenario, you will be exporting a Raster image and the attribute data from your database to a GeoTIFF.

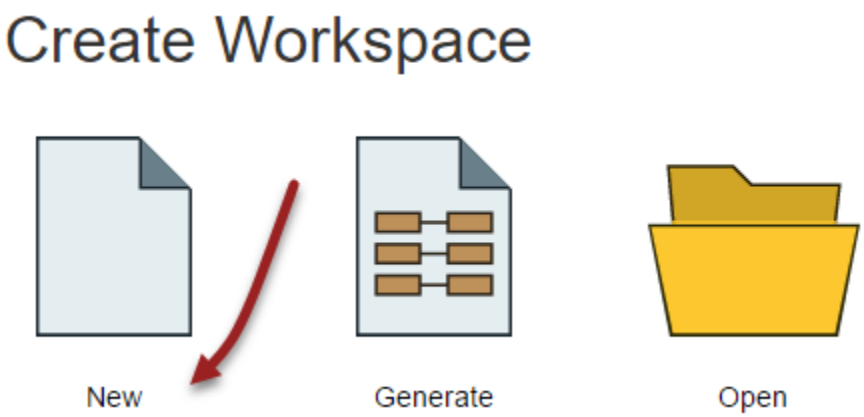
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

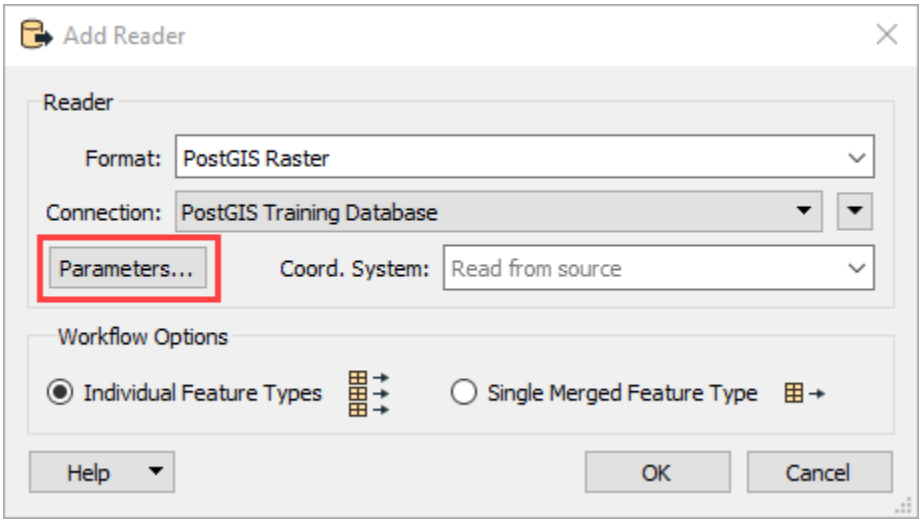
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



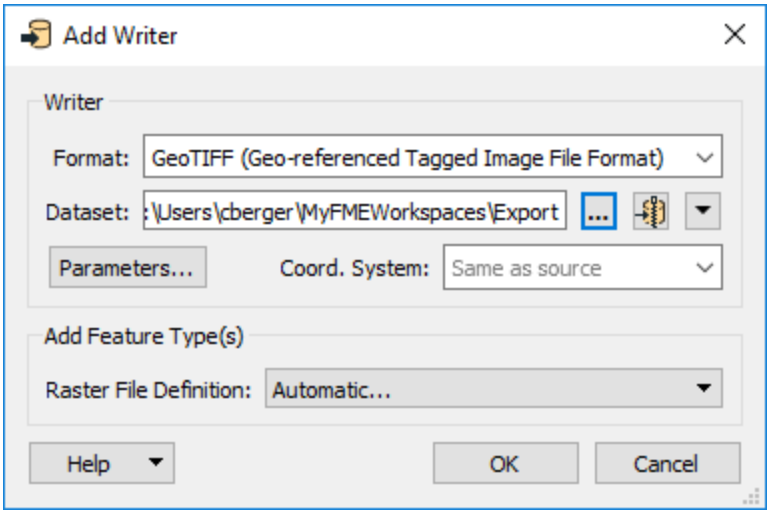
3. Add a PostGIS Raster Reader

- Start typing "**PostGIS Raster**" without anything selected on the canvas, then select the **PostGIS Raster** format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the 06-07-LM table from the Table List.



4. Add a GeoTIFF Writer

- Similar to adding a reader, type "GeoTIFF" then select the GeoTIFF format from the list of Writers.
- Click on the Dataset ellipsis and navigate to a folder where you want to save the GeoTIFF file such as C:\Users\Documents\MyFMEWorkspaces\Export.
- Set the Raster File Definition to Automatic to ensure the attributes are transferred from the database to the GeoTIFF. Click OK to add the writer to the canvas.



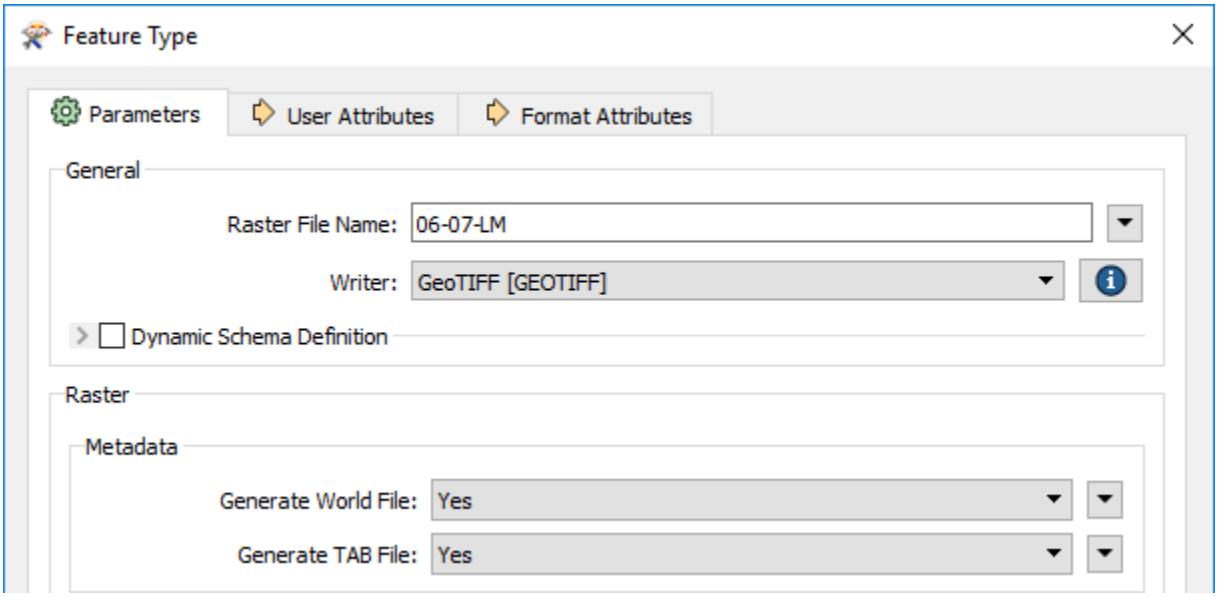
5. Specify a Filename

- After adding the Writer to the canvas, the writer feature type parameters will appear.
- Set the Raster File Name to “06-07-LM”.

6. Managing your Metadata

- Set the Generate World File and Generate Tab File to Yes from the drop-down list and click OK to accept the writer parameters.
- Lastly, connect the PostGIS Reader to the GeoTIFF Writer.

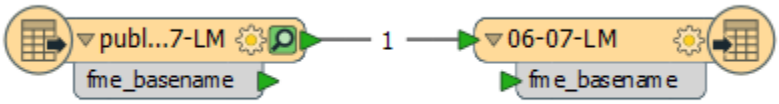
FME allows you to generate both World and TAB files when writing to the GeoTIFF format. World files (.tfw) are important because they store georeferencing information about the raster image (i.e. the origin, spacing, and rotation of the raster image). Additionally, TAB files (.tab) are important because they allow you to export the user-defined attributes of the raster image, meaning all of the attributes in the associated database table will be exported along with the GeoTIFF. For more technical information on using the GeoTIFF reader, see the Documentation.



7. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the

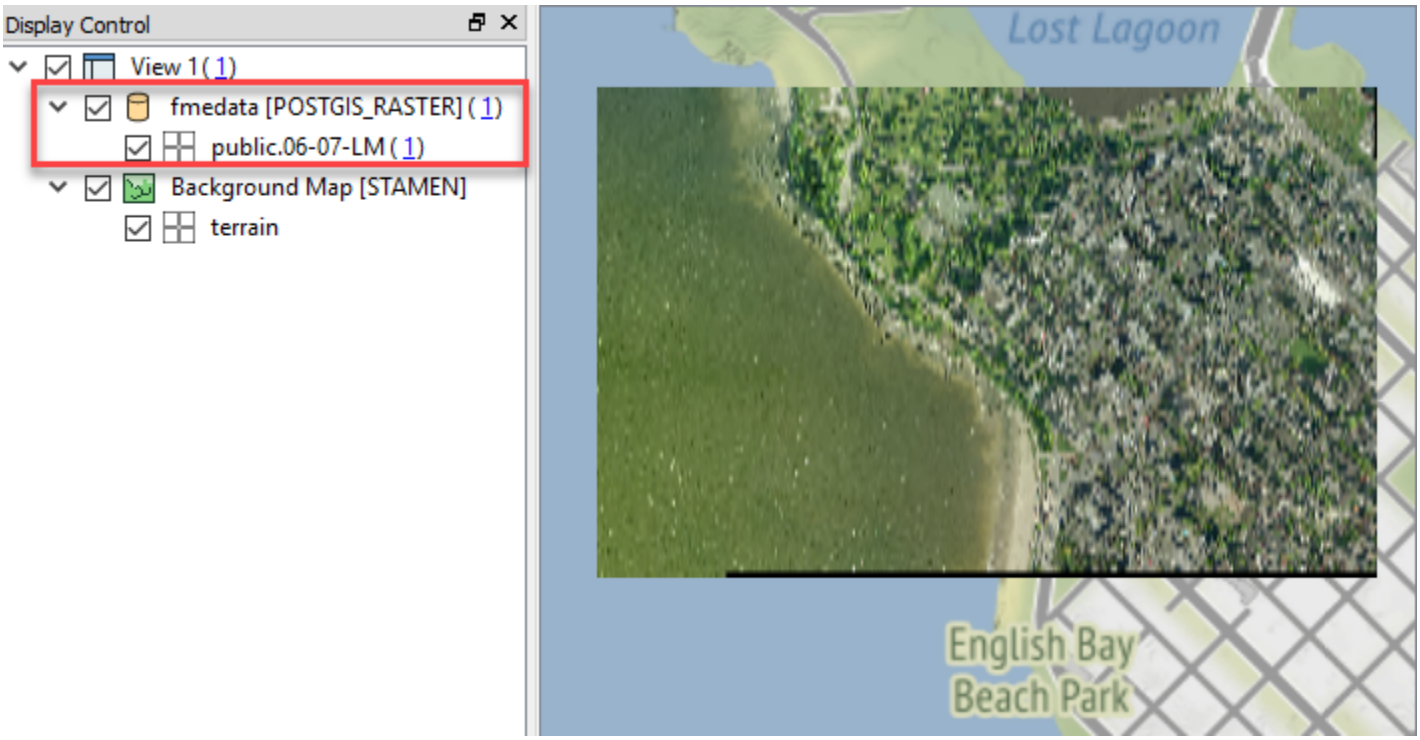
menu bar.



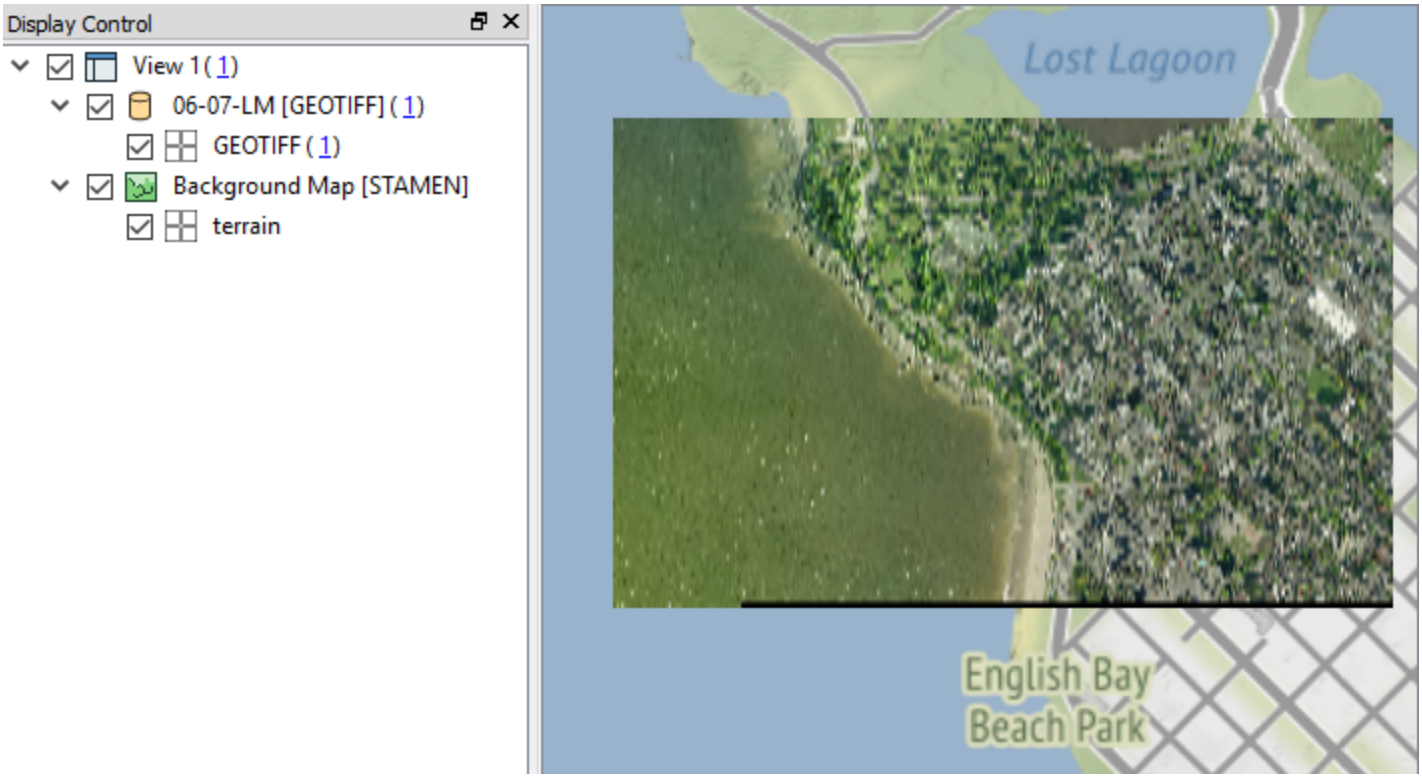
You have successfully exported Raster data from your database. You can view the output by selecting the writer feature type and clicking the FME Data Inspector logo in the shortcut menu that appears above it.

Results

Input



Output



Exercise 3 - *Importing Raster Data*

In this scenario, you will be importing a Raster image and its attributes into your database.

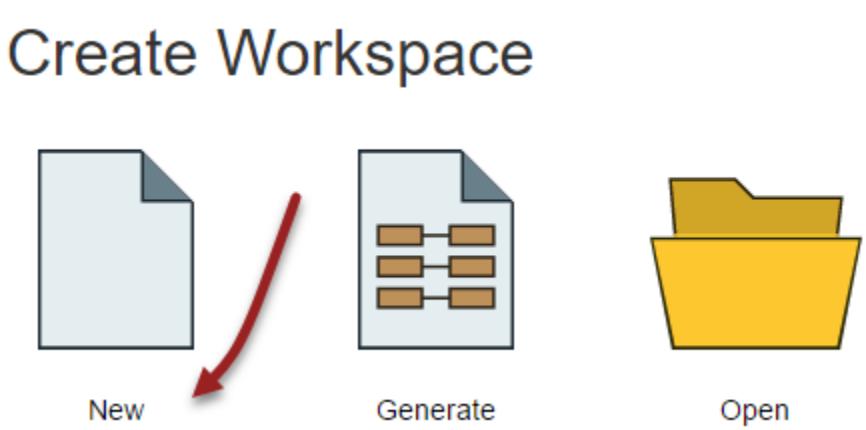
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

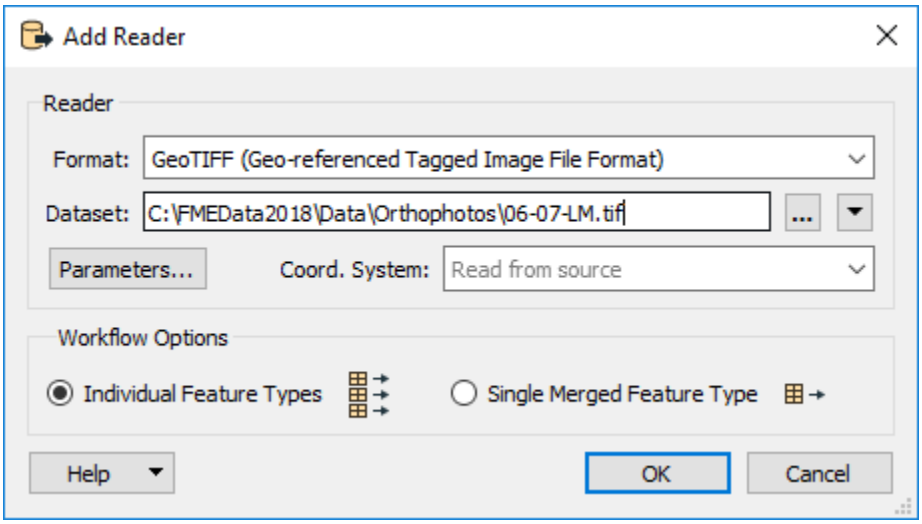
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



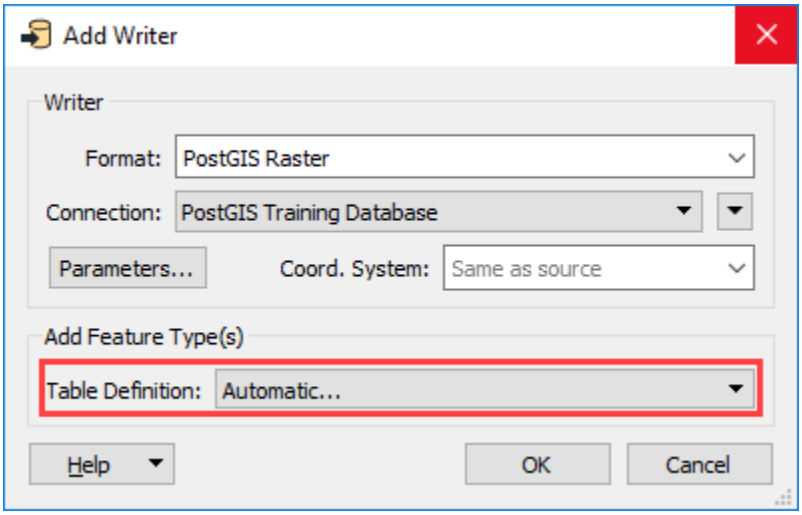
3. Add a GeoTIFF Reader

- Start typing “GeoTIFF” without anything selected on the canvas, then select the GeoTIFF format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Set the 06-07-LM GeoTIFF as the Dataset (available in the downloads section | 06-07-lm.zip).



4. Add a PostGIS Raster Writer

- Similar to adding a Reader, type “**PostGIS Raster**” then select the **PostGIS Raster** format from the list of Writers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Set the Table Definition to Automatic to ensure all of the attributes that are read into the workspace are written to the PostGIS Raster database.

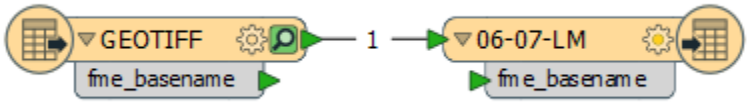


5. Set the Table Name

- After the PostGIS Raster Writer has been added, the Writer Feature Type parameters will automatically open. Set the Table Name to “06-07-LM-copy”. By default, the table will be placed in the “public” schema.
- Set the Table Handling to Drop and Create.
- Leave all other parameters as default and click OK to add the writer to the canvas.

6. Run the Workspace

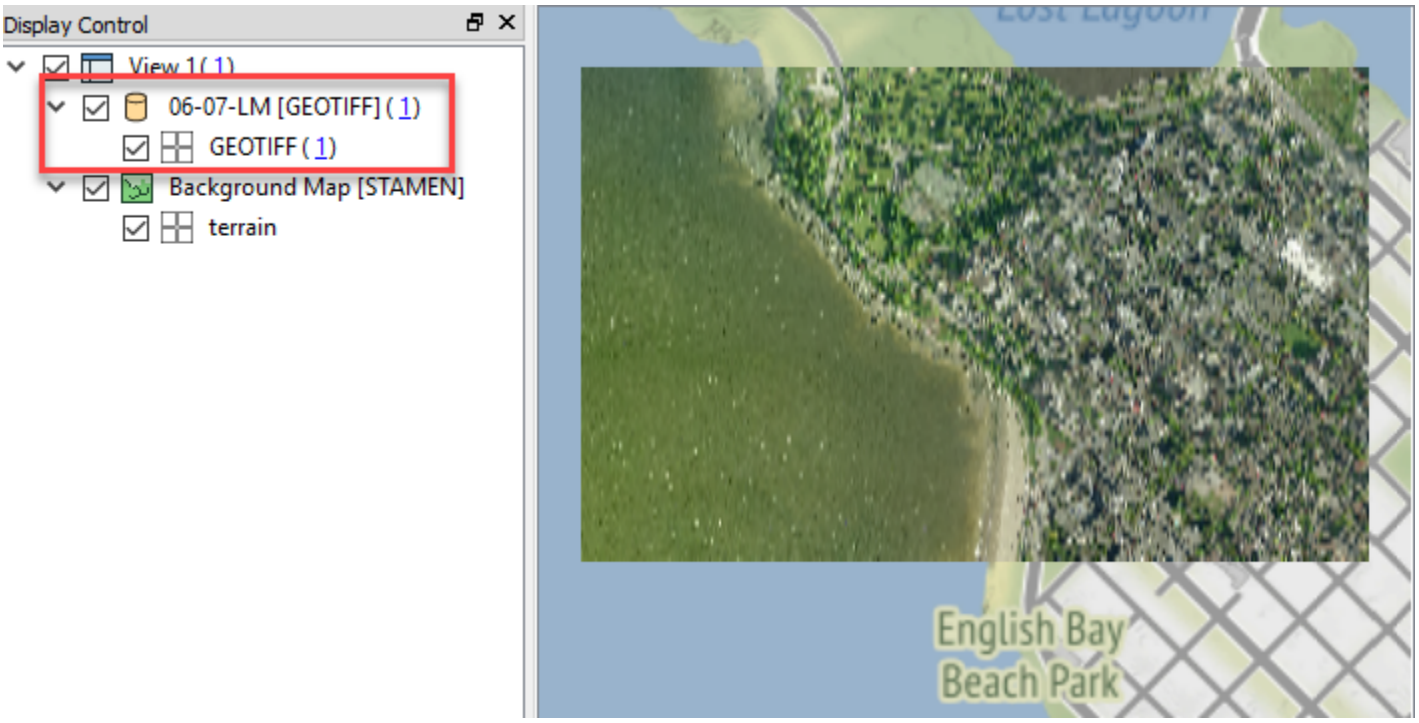
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



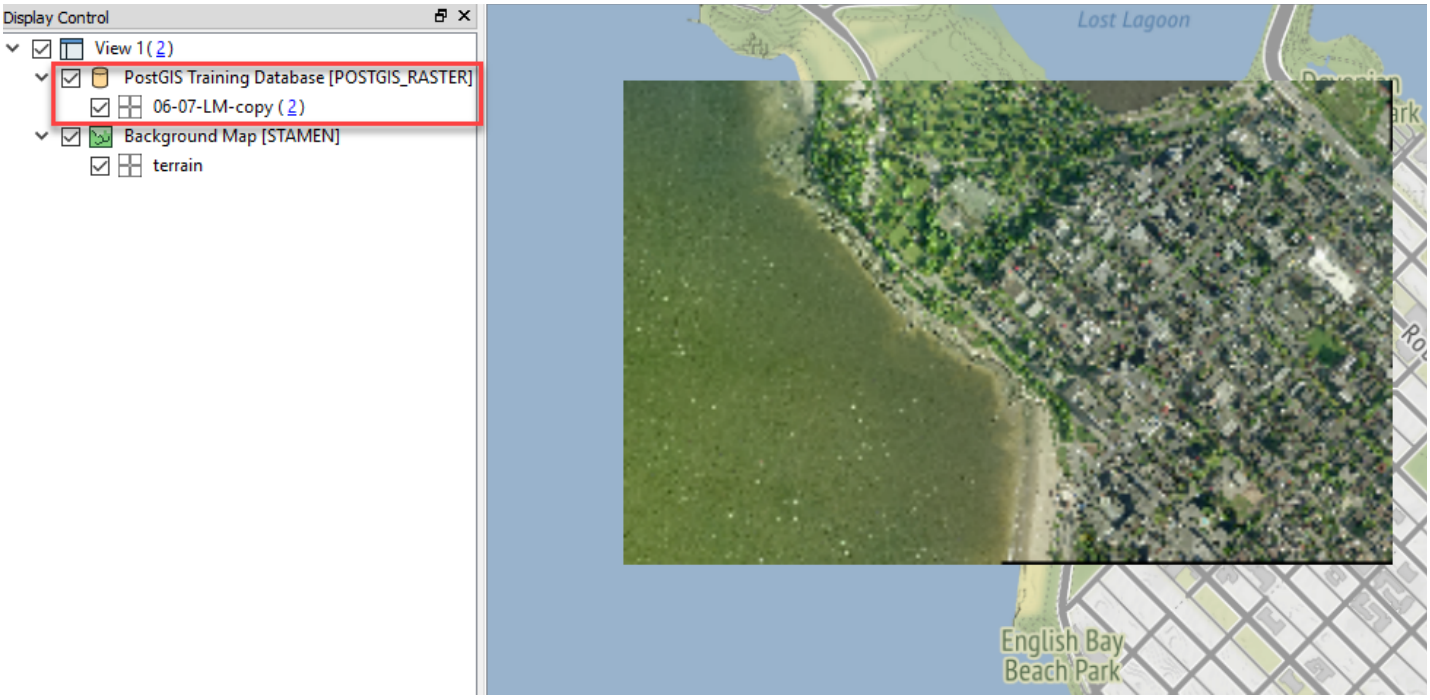
You have successfully imported a GeoTIFF into your database. You can view the output by selecting the writer and clicking the FME Data Inspector logo in the shortcut menu that appears.

Results

Input



Output



Exercise 4 - *Exporting Vector Data*

In this scenario, you want to export a copy of some vector data that is currently stored in your database.

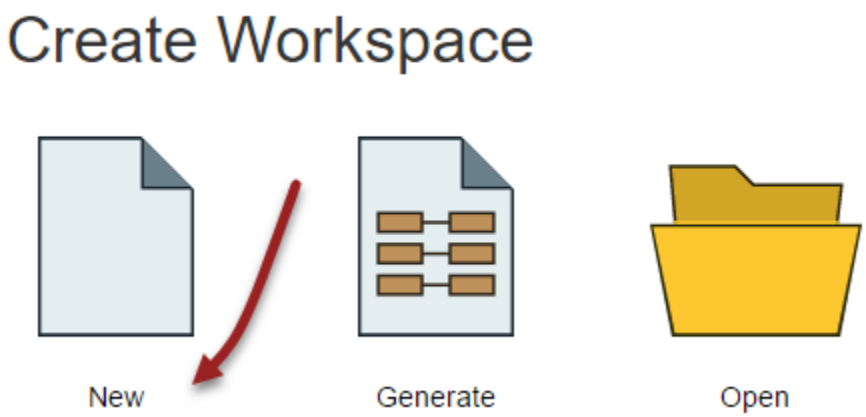
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

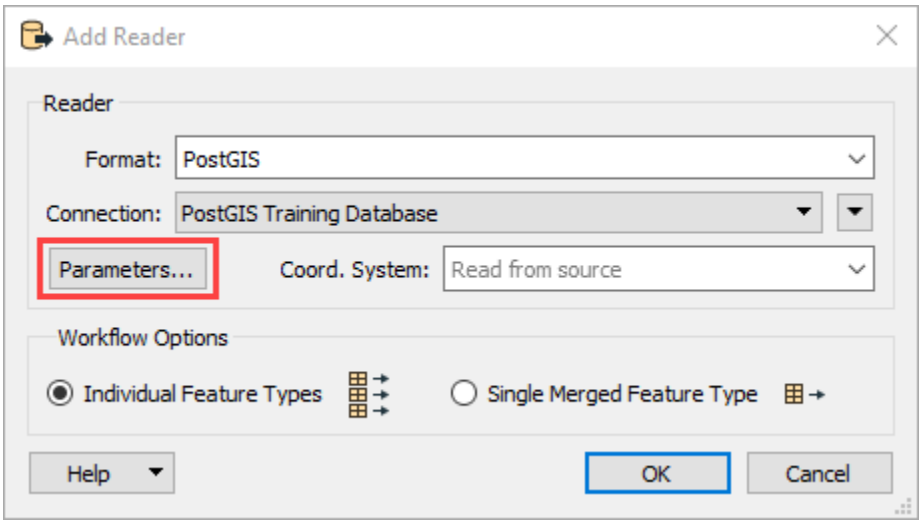
- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
    - Host: postgis.train.safe.com
    - Port: 5432
    - Database: fmedata
    - Username: fmedata
    - Password: fmedata

Next, open the Reader Parameters and select the VanBlocks table from the Table List.



4. Add a Shapefile Writer

- Similar to adding a reader, type “shapefile” then select the shapefile format from the list of Writers.
- Click on the Dataset ellipsis to specify a folder for the shapefile such as C:\Users\Documents\MyFMEWorkspaces\Export.
- Set the Shapefile Definition to Automatic to ensure the attributes are transferred from the database to the shapefile. Click OK to add the writer to the canvas.

5. Specify a Filename

- After adding the Writer to the canvas, the writer feature type parameters will appear.
- Set the Shapefile Name to something meaningful like “VanBlocks” and the Table Handling to Drop and Create.

6. Specify the Geometry and Output Dimension

Shapefile features consist of geometry and a set of user-defined attributes. In this case, you are reading 3D polygon data from a PostGIS database because a Z value (0) has been appended to the feature coordinates.

- Open the VanBlocks Shapefile parameters.
- If you want to make sure the shapefile has the polygon geometry, select the “shape_polygon” from the geometry drop-down list. This is an optional step since you already know the VanBlocks dataset that you read in only contains polygon features.

Note: since there is a Z coordinate specified in the coordinate dimensions the Shapefile Writer will assume

the output is intended to be 3D if you use the default output dimension "Dimension from the first feature".

- Set the Output Shapefile Dimension to 2D. This will ensure you are writing 2D data as it will remove the z coordinate.
- Click OK to accept the changes and close the Feature Type Parameters.
- Once the Writer has been added to the Canvas, connect the Reader Feature Type to the Writer Feature Type.

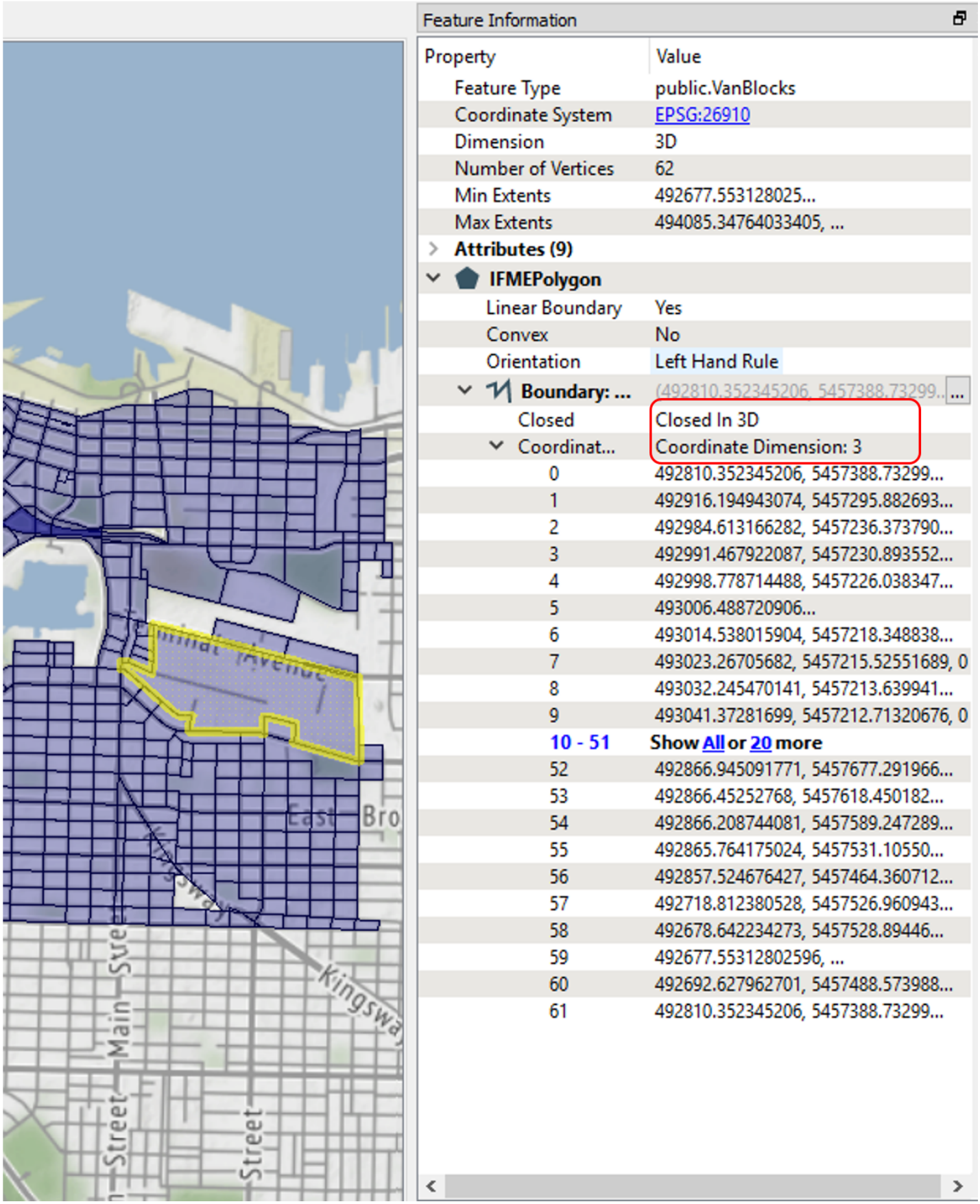
7. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.

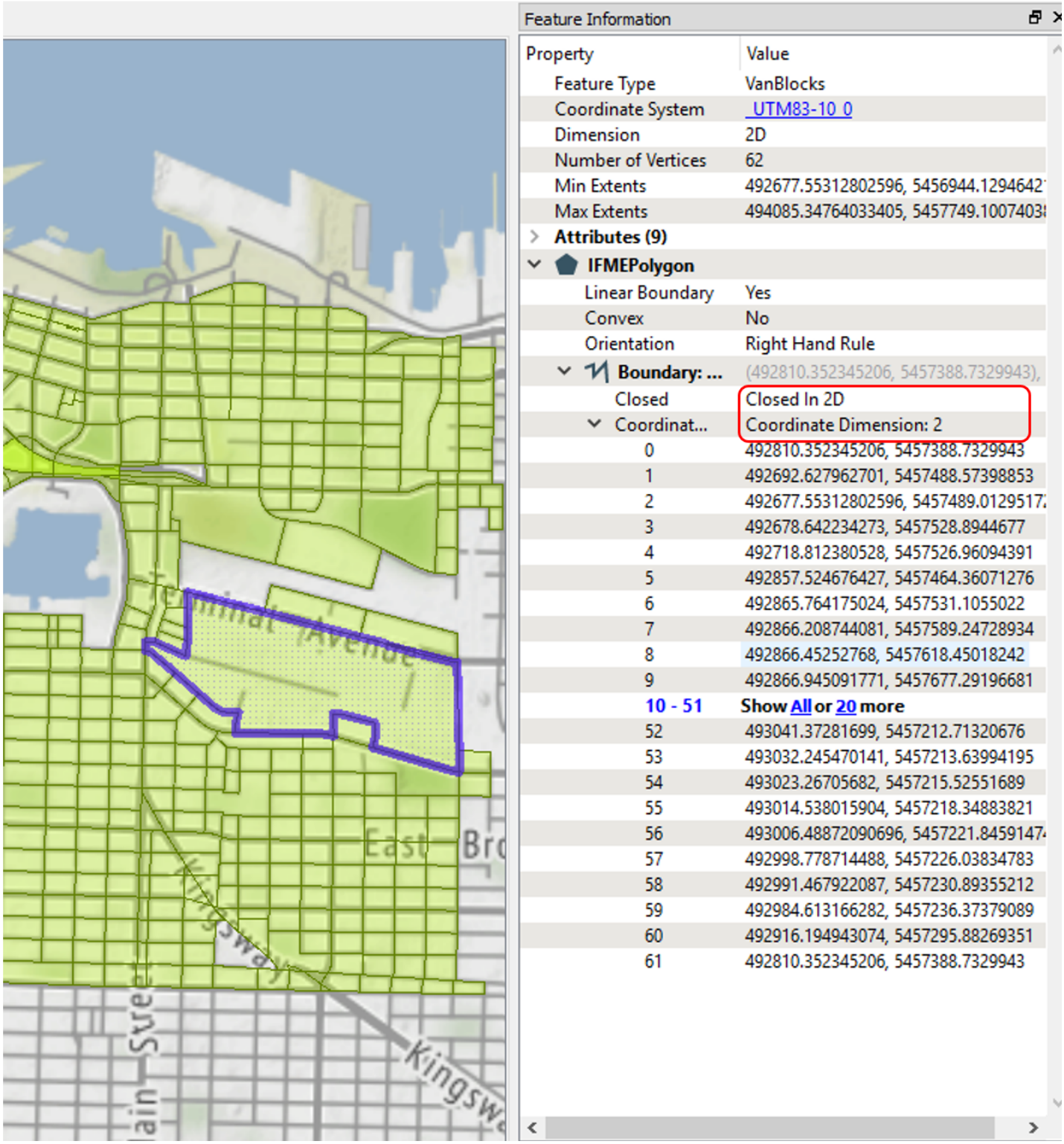
You have successfully exported polygon features from your database into the shapefile format. You can view the output by selecting the writer and clicking the FME Data Inspector logo in the shortcut menu that appears.

Results

Input


Feature Information
Property
Value
Feature Type
public.VanBlocks
Coordinate System
EPSG:26910
Dimension
3D
Number of Vertices
62
Min Extents
492677.553128025...
Max Extents
494085.34764033405, ...
Attributes (9)
IFMEPolygon
Linear Boundary
Yes
Convex
No
Orientation
Left Hand Rule
Boundary: ...
(492810.352345206, 5457388.73299...
Closed
Closed In 3D
Coordinat...
Coordinate Dimension: 3
0
492810.352345206, 5457388.73299...
1
492916.194943074, 5457295.882693...
2
492984.613166282, 5457236.373790...
3
492991.467922087, 5457230.893552...
4
492998.778714488, 5457226.038347...
5
493006.488720906...
6
493014.538015904, 5457218.348838...
7
493023.26705682, 5457215.52551689, 0
8
493032.245470141, 5457213.639941...
9
493041.37281699, 5457212.71320676, 0
10 - 51
Show All or 20 more
52
492866.945091771, 5457677.291966...
53
492866.45252768, 5457618.450182...
54
492866.208744081, 5457589.247289...
55
492865.764175024, 5457531.10550...
56
492857.524676427, 5457464.360712...
57
492718.812380528, 5457526.960943...
58
492678.642234273, 5457528.89446...
59
492677.55312802596, ...
60
492692.627962701, 5457488.573988...
61
492810.352345206, 5457388.73299...
Kingsway
Main Street
Street

Output



Exercise 5 - *Importing Vector Data*

In this scenario, you will be importing polygon features from a shapefile into your database.

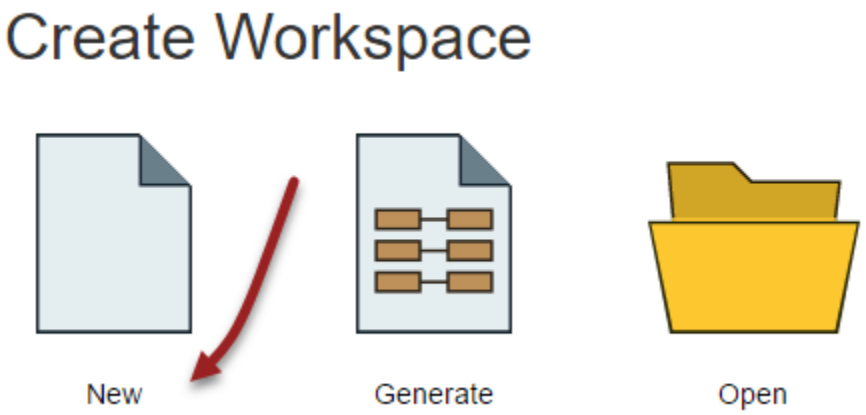
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

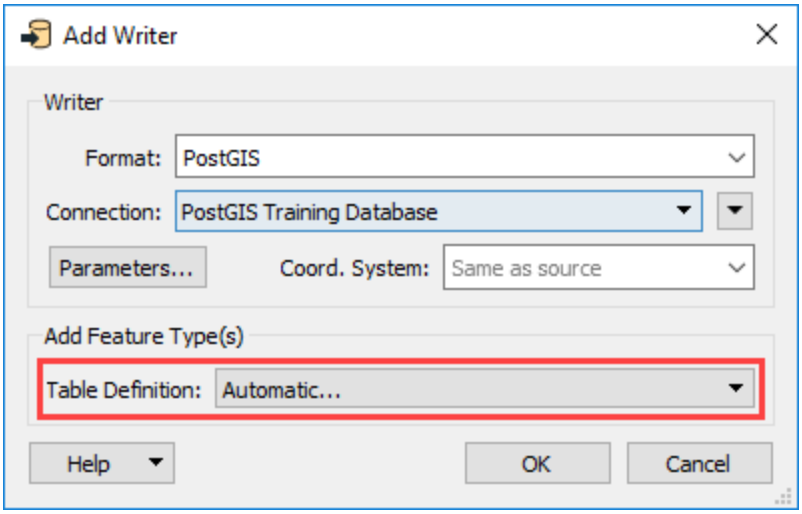


3. Add a Shapefile Reader

- Start typing "Shapefile" without anything selected on the canvas, then select the Shapefile format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis and navigate to the folder containing VanBlocks.shp that you just exported. Select the VanBlocks.shp and click Open, then OK to add the Shapefile Reader to the canvas.

4. Add a PostGIS Writer

- Type "PostGIS" then select the PostGIS format from the list of Writers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Set the Table Definition to Automatic to ensure all of the attributes that are read into the workspace are written to the PostGIS database.

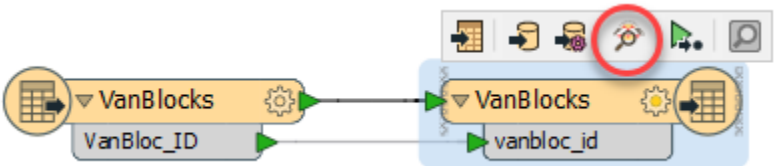


5. Set the Table Name

- After the workspace is generated, the Writer Feature Type parameters should automatically open. Set the Table Name to “VanBlocks_copy”. By default, the table will be placed in the “public” schema.
- Set the Table Handling to Drop and Create.
- Leave all other parameters as default and click OK to add the writer to the workspace.

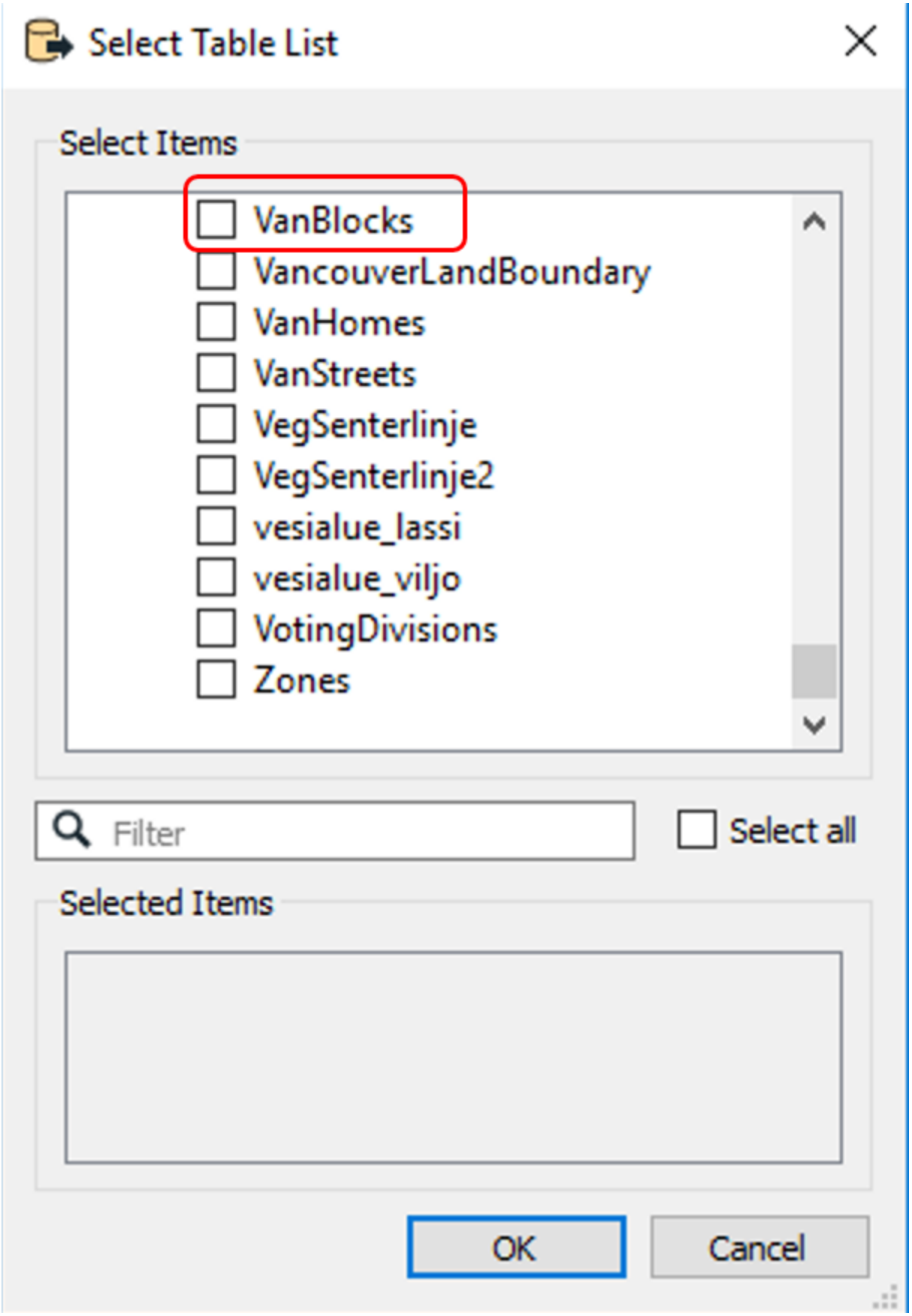
6. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.

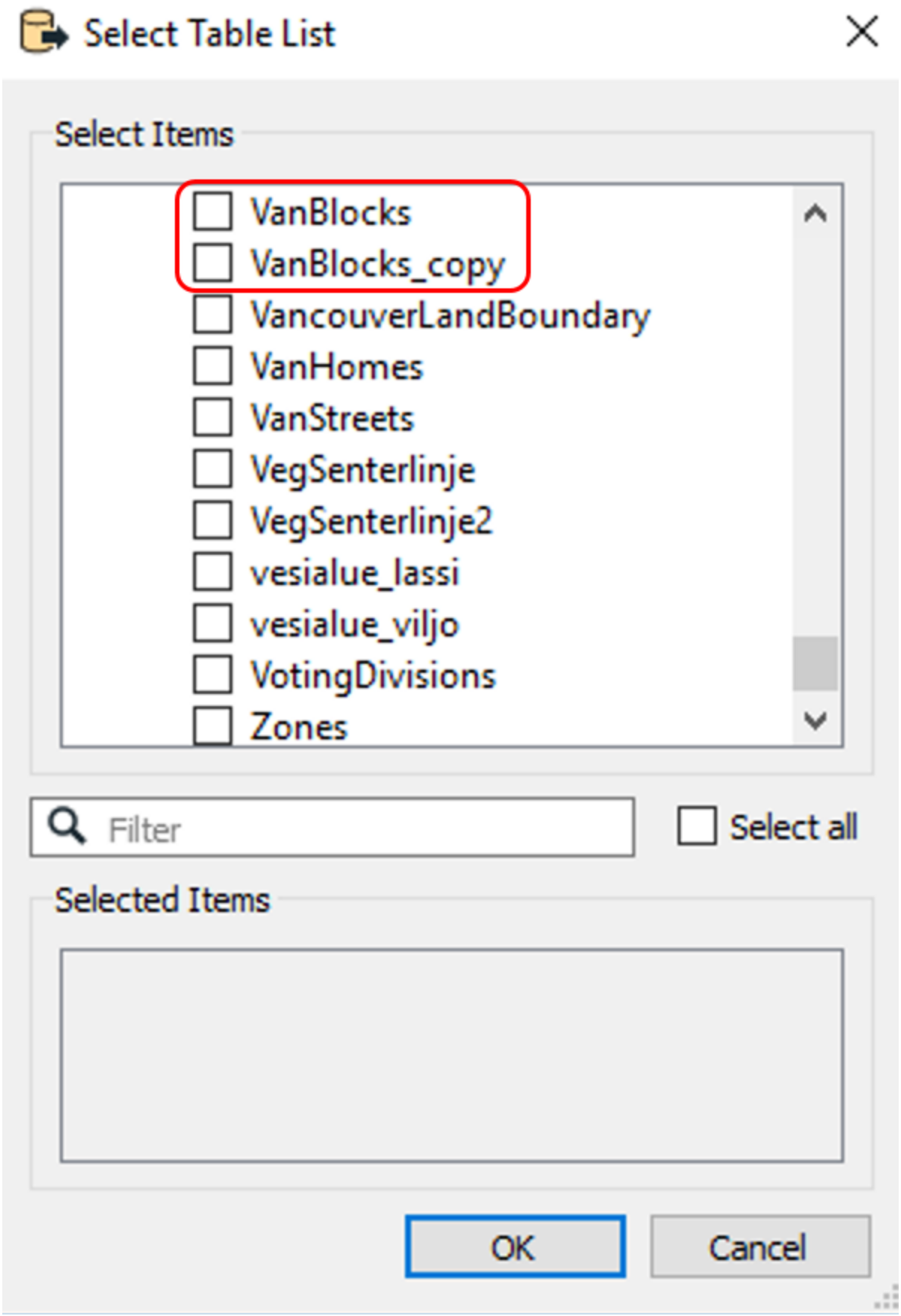


Results

Input


Select Table List
Select Items
VanBlocks
VancouverLandBoundary
VanHomes
VanStreets
VegSenterlinje
VegSenterlinje2
vesialue_lassi
vesialue_viljo
VotingDivisions
Zones
Filter
Select all
Selected Items
OK
Cancel

Output



Data Attribution

Data Retrieval

Overview

Data retrieval in FME is primarily done in the FME Data Inspector. In this tutorial, you will focus on retrieving data using the FME Data Inspector. For a more detailed tutorial on the FME Data Inspector, see the Desktop Basic Course Manual. Alternatively, if you are interested in using the FME Data Inspector for raster data, see the Viewing and Inspecting Rasters article.

Exercise

In this scenario, you will select features using the FME Data Inspector. First, you will retrieve data by selecting individual features on the screen, then you will learn how to select by querying attributes, and finally, you will learn how to select features using a search envelope. Once your data is selected, you will then export the features to a CSV file; however, you can just as easily export to another format (such as shapefile, file geodatabase, etc.).

Instructions

1. Start FME Workbench

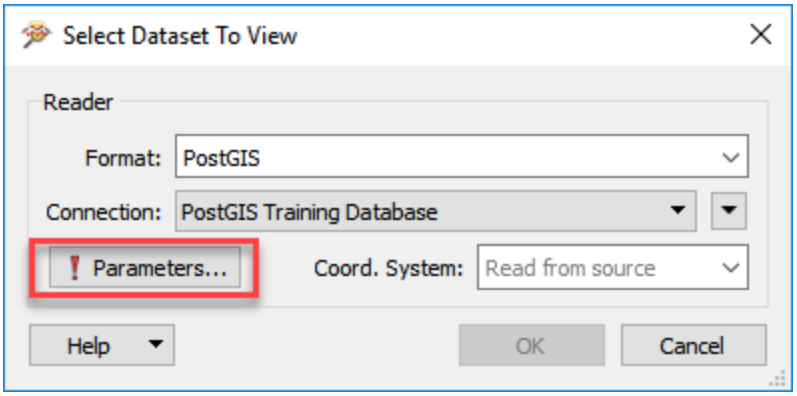
- If it isn't open already, launch the FME Data Inspector.

2. Open a Dataset

- Using the shortcut CTRL+O, the open folder icon, or by navigating to File -> Open Dataset.

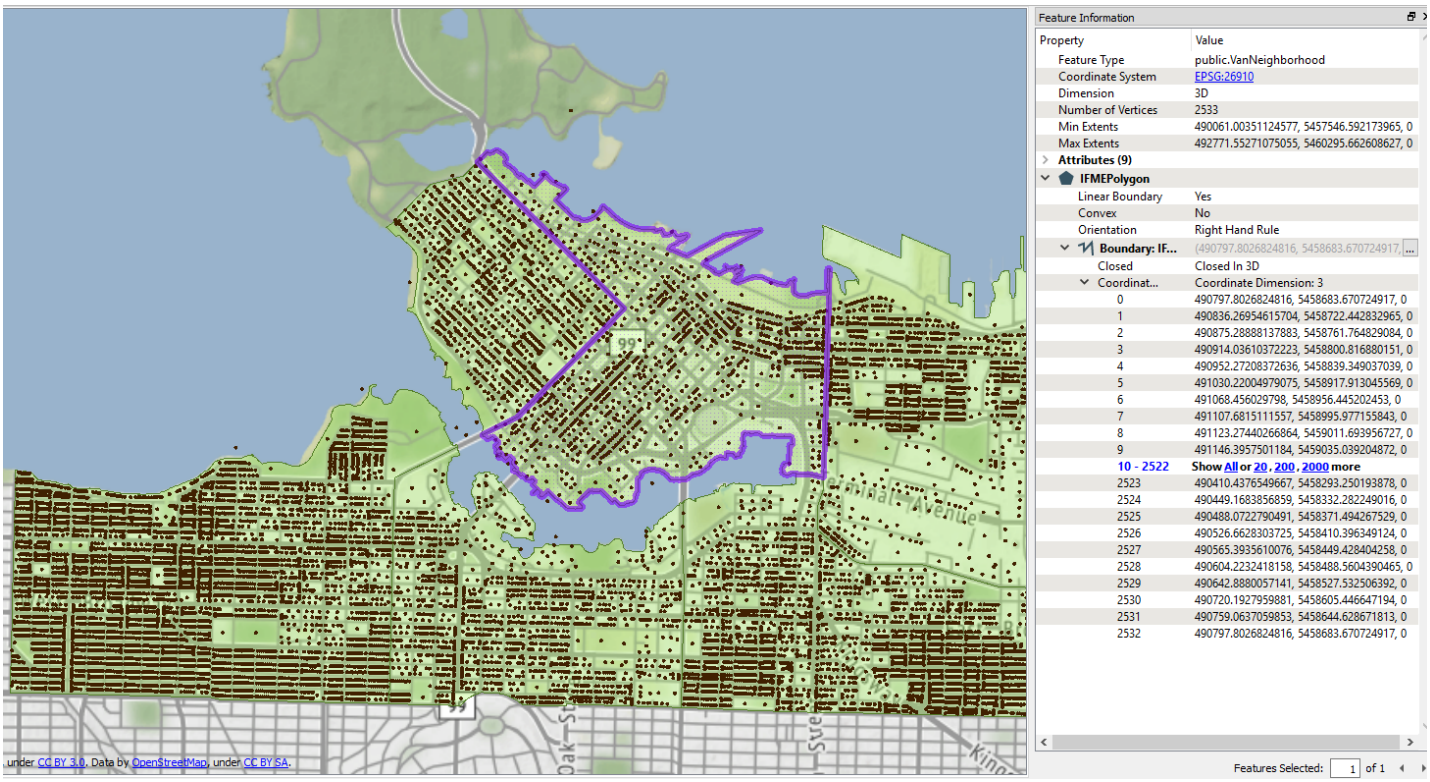
3. Select a Dataset to View

- Set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection, select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the PublicAddress and VanNeighborhood tables from the Table List, then click OK to open the dataset.

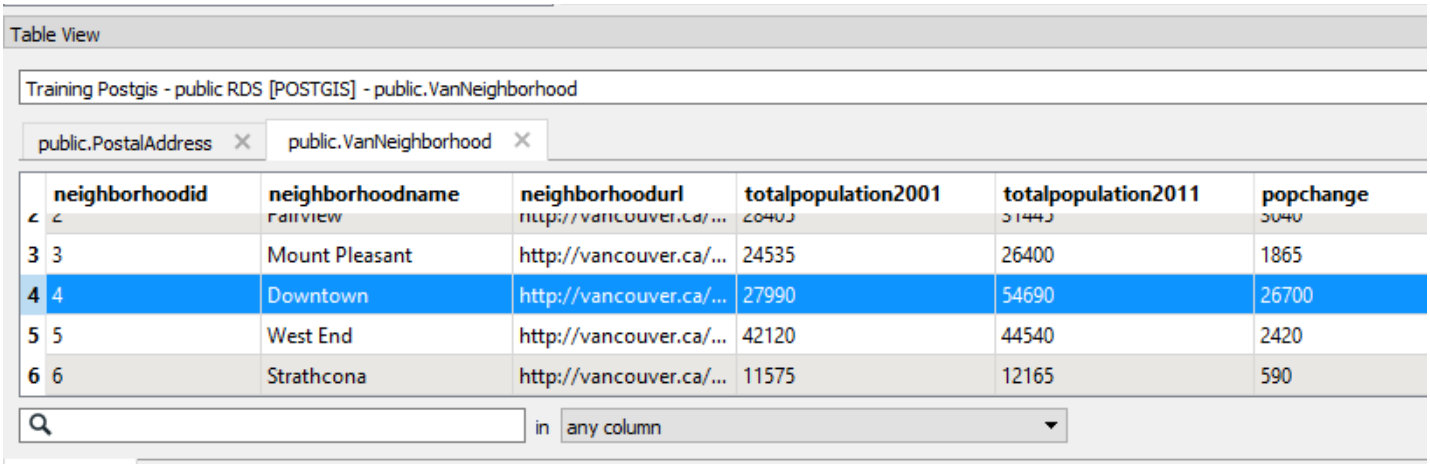


4. Select a Feature

- In the View Window, click on one of the VanNeighborhood polygons or PostalAddress points. Once selected, the Feature Information will appear in a docked window (i.e. attributes and metadata of the selected feature).

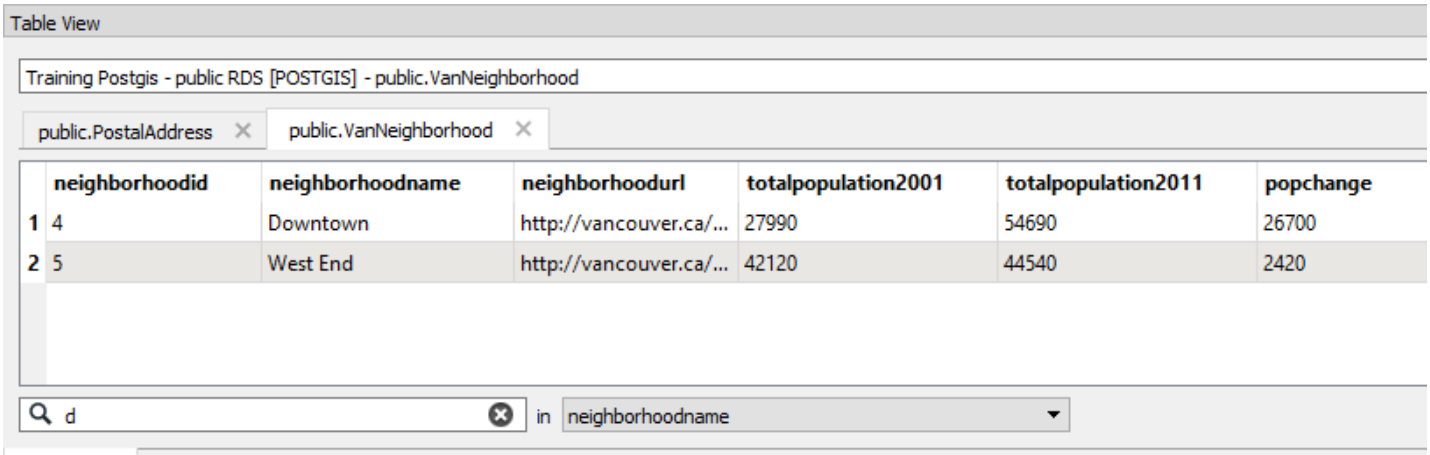


You will also notice that the selected feature is also highlighted in the Table View. Another method for selecting features is by clicking on them in the Table View. Here you can sort the columns by ascending/descending for both alphabetical and numerical data.

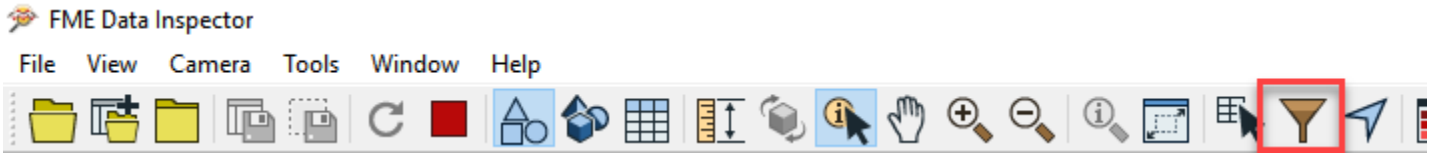


- Alternatively, you can narrow your selections by searching for values in specific columns using the search bar. Try typing in “D” in the neighborhoodname column.

This will narrow your selection down to 2 features, Downtown and West End

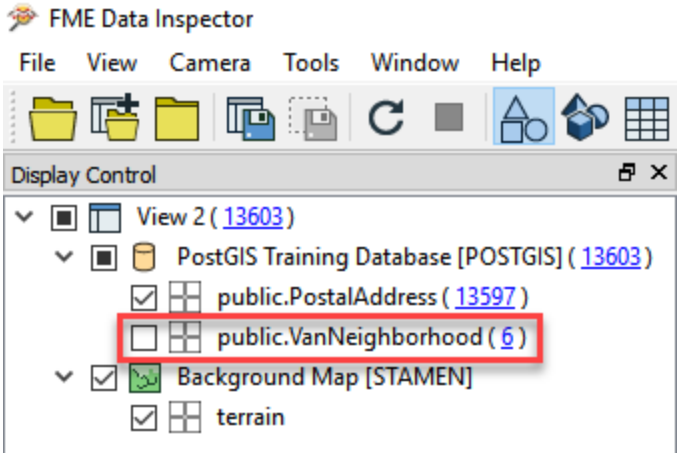


Another option is to use the Filter Features function to only display features that pass a test clause.



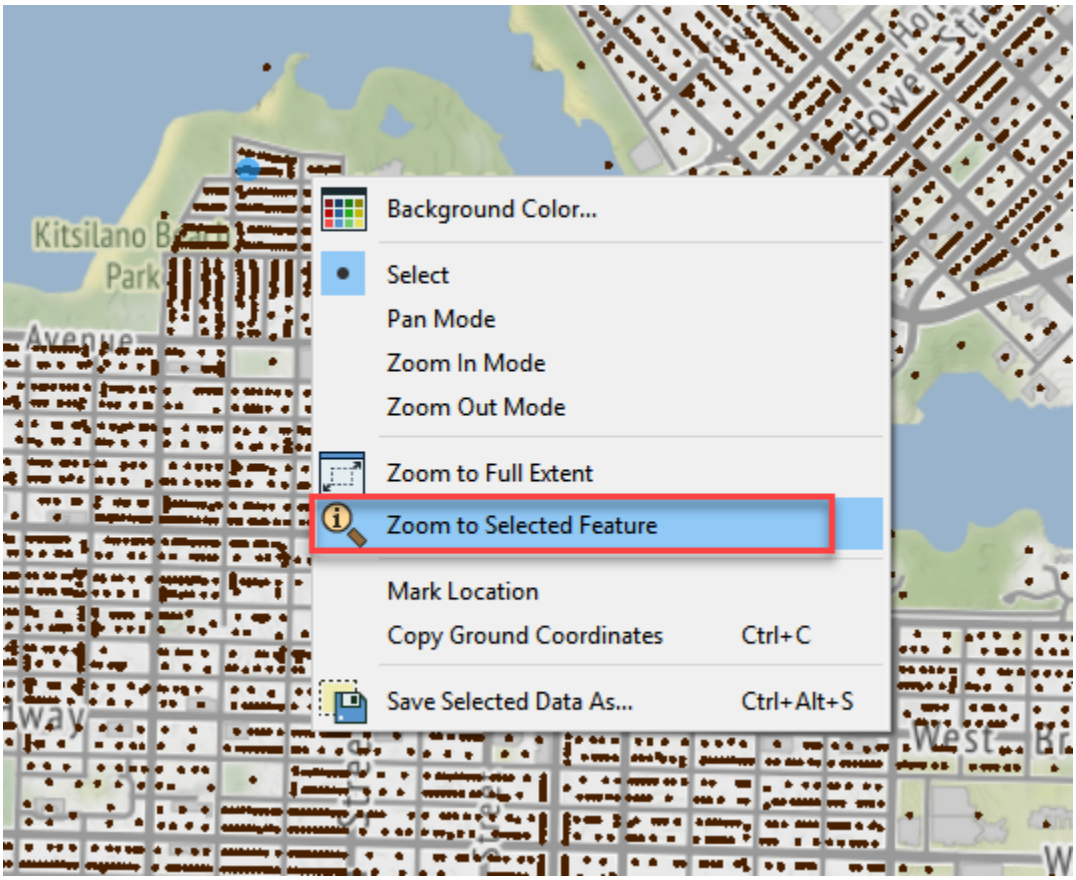
5. Turn off the VanNeighborhood Layer

- In the Display Control, under View 1, deselect the public.VanNeighborhood layer. This will leave you with only the Postal Address points layer selected.



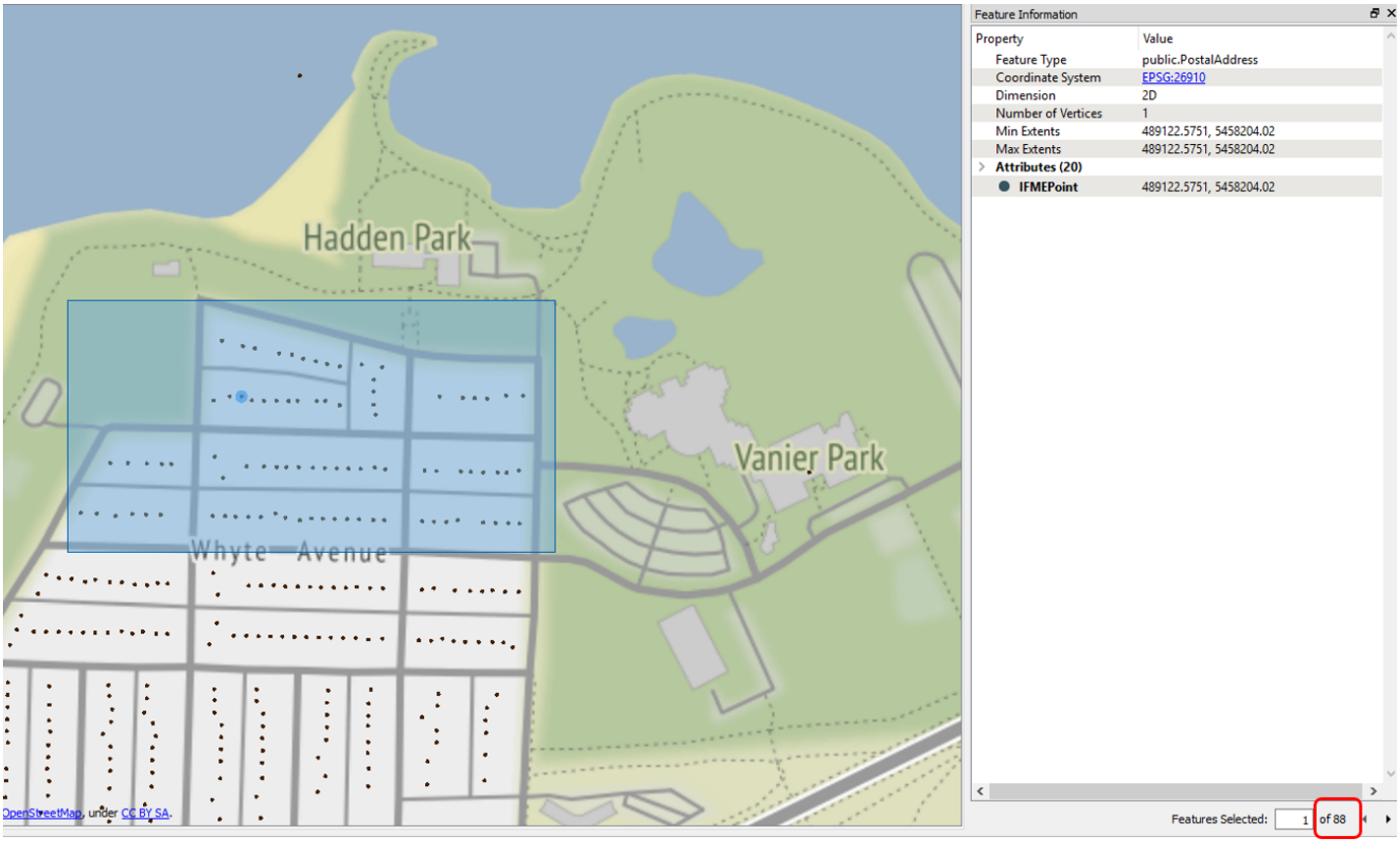
6. Zoom to Selected Features

- Select a point feature anywhere on the map.
- With a point selected, right-click anywhere in the View window to bring up the selection menu. Choose Zoom to Selected Feature to zoom into the feature.



7. Select by Rectangle

- Put your cursor over the selected point and zoom in a little more using the scroll wheel.
- Next, draw a rectangle over multiple points by clicking and dragging your mouse - this will act as a search envelope that will select features that are within or touch the selection box.



This selection method allows you to select multiple features which can be previewed using the arrows below the feature information window.

8. Export the Selected Data

- With the data still selected, go to File -> Save Selected Data As. Alternatively, you can use the shortcut CTRL+ALT+S to export the selected data.

9. Save the Selected Data

- Set the Format to CSV and set the dataset output folder (for example C:\Users\Documents\MyFMEWorkspaces\Export), then click OK.

10. Open the CSV File

- Navigate to the folder where you saved the dataset. Since you saved selected data from the public.PostalAddress dataset, the file will be called public.PostalAddress.csv. You can open the new dataset in the FME Data Inspector or in a text editor to view the output.

Results

Input

1 selected / 13597 row(s)

Output

public.PostalAddress.csv

Table View

Training Postgis - public RDS [POSTGIS] - public.VanNeighborhood

public.PostalAddress | public.VanNeighborhood

	neighborhoodid	neighborhoodname	neighborhoodurl	totalpopulation2001	totalpopulation2011	popchange
2	2	Fairview	http://vancouver.ca/...	28405	31445	3040
3	3	Mount Pleasant	http://vancouver.ca/...	24535	26400	1865
4	4	Downtown	http://vancouver.ca/...	27990	54690	26700
5	5	West End	http://vancouver.ca/...	42120	44540	2420
6	6	Strathcona	http://vancouver.ca/...	11575	12165	590

in any column

1 selected / 161 row(s)

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Select Data by Area Masks

Overview

Extracting data using area masks in FME requires the use of both a raster and a vector dataset and the Clipper transformer. In this tutorial, you will be selecting raster data inside of polygon features; however, this tutorial can easily be modified to select vector features within a search envelope.

Downloads

select-data-by-area-mask.fmwt

dem-full.zip

landboundarygdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you are interested in clipping the extent of an existing DEM to the boundary of your city based on a polygon feature. This exercise will use a municipal land boundary as a search envelope; however, you can also use multiple polygon features.

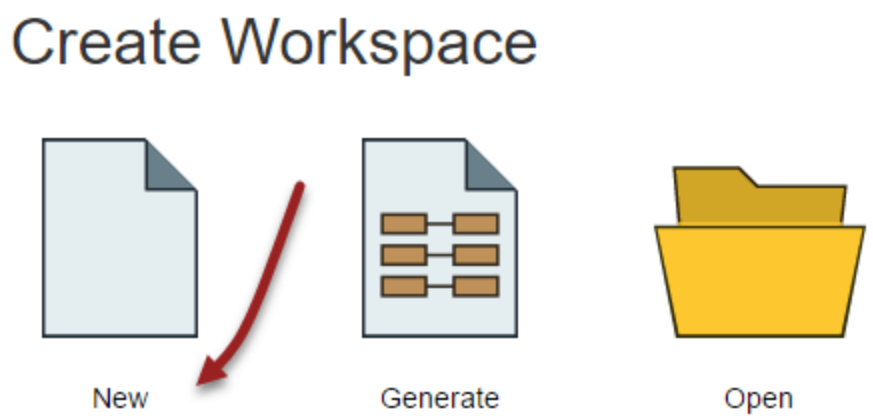
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

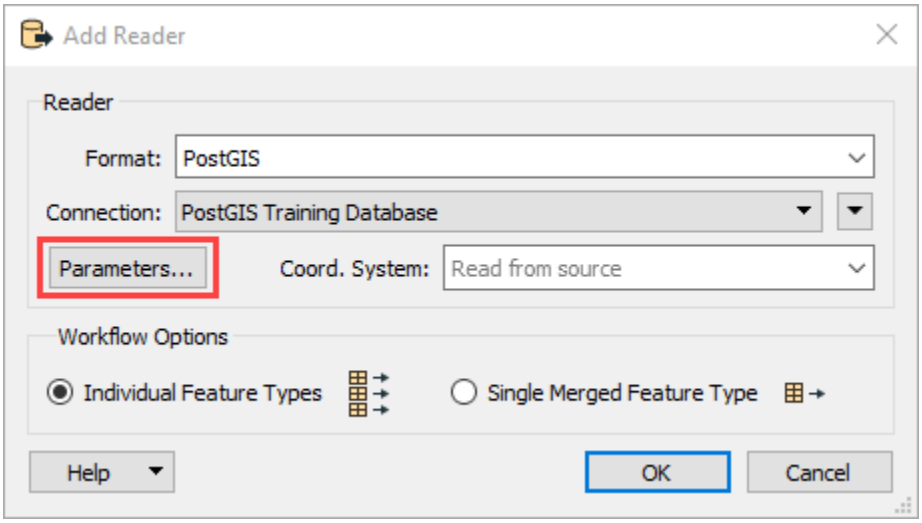
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection, select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the VancouverLandBoundary table from the Table List.



4. Add a Canadian Digital Elevation Data (CDED) Reader

- Start typing “CDED” without anything selected on the canvas to bring up the list of FME Transformers in the Quick Add Search. Then select the Canadian Digital Elevation Data format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis and navigate to the DEM-FULL.dem found in the ElevationModel folder of the provided sample data. Once selected, click Open, then click OK to add the reader to the canvas.

5. Reproject the Data

In this scenario, the provided datasets do not share the same coordinate system (one is currently in LL83 and the other is in UTM83-10). As a result, you cannot continue without reprojecting the DEM to a linear coordinate system. For more information on coordinate systems in FME, see the Using Coordinate Systems within FME article.

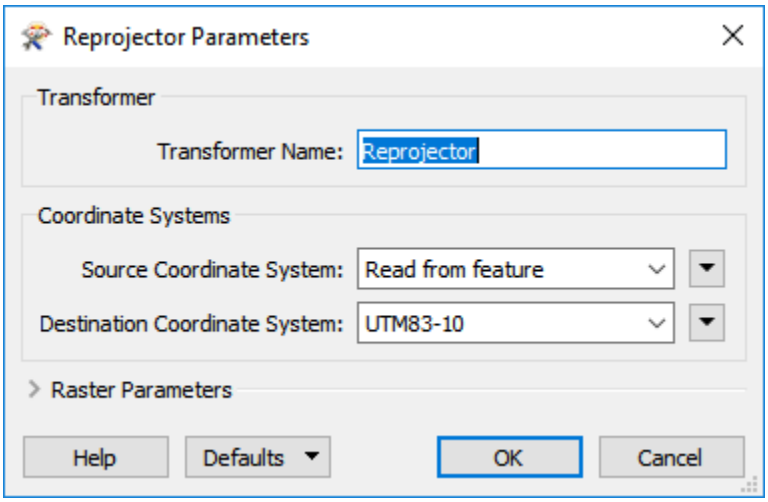
- Add a Reprojector to the canvas by typing “Reprojector” to bring up the list of FME Transformers in the Quick Add Search. Select the Reprojector from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the CDED Reader to the Reprojector.

6. Set the Destination Coordinate System

- Once the Reprojector connection has been made, double-click the Reprojector or click on the gear icon to open the transformer parameters dialog.

Notice the Source Coordinate System is automatically read from the feature. The only parameter that needs to be changed in this scenario is the Destination Coordinate System.

- Set the Destination Coordinate System to UTM83-10 by typing “UTM83-10” in the text box or by selecting it from the dropdown list and choosing “More Coord Systems”.



For more tips on Coordinate System Transformation, see the Desktop Basic Course Manual.

7. Add a Clipper

In FME, the Clipper performs a geometric clipping operation similar to a cookie cutter. With that in mind, you will clip the areas outside of the VancouverLandBoundary polygon because you are only interested in the DEM for the Vancouver area. For a more detailed description of how the Clipper works, see the Example Using the Clipper Transformer article.

- Connect the VancouverLandBoundary reader to the Clipper port.
- Next, Connect the Reprojector:Reprojected port to the Clipee port.

8. Add an Inspector

The Clipper:Inside port will output Clippee features that are completely within the Clipper, and Clippee features that intersect the Clipper which were broken into pieces. Conversely, the Clipper:Outside port will output Clippee features that are completely outside of the Clipper. You can learn more technical details, see

the Clipper Documentation.

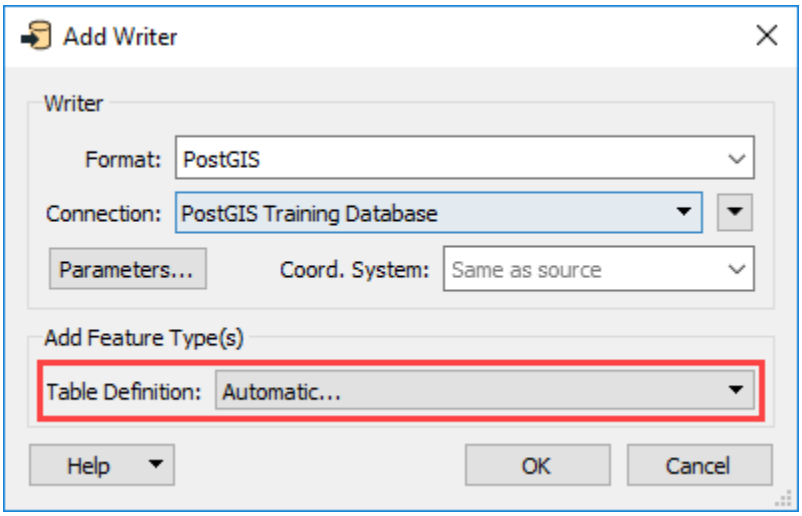
- Since you are only interested in the area within the clip boundary, add an Inspector and connect it to the Clipper:Inside port. This will also automatically open the output in the FME Data Inspector after running a successful translation

9. Add a PostGIS Raster Writer

- Similar to adding a reader, type “PostGIS Raster” then select the PostGIS Raster format from the list of Writers.
- Set the Writer Connection to the PostGIS Training Database (same as the connection listed in step 3) and set the Table Definition to Automatic, then click OK to add the writer to the canvas and the parameters.

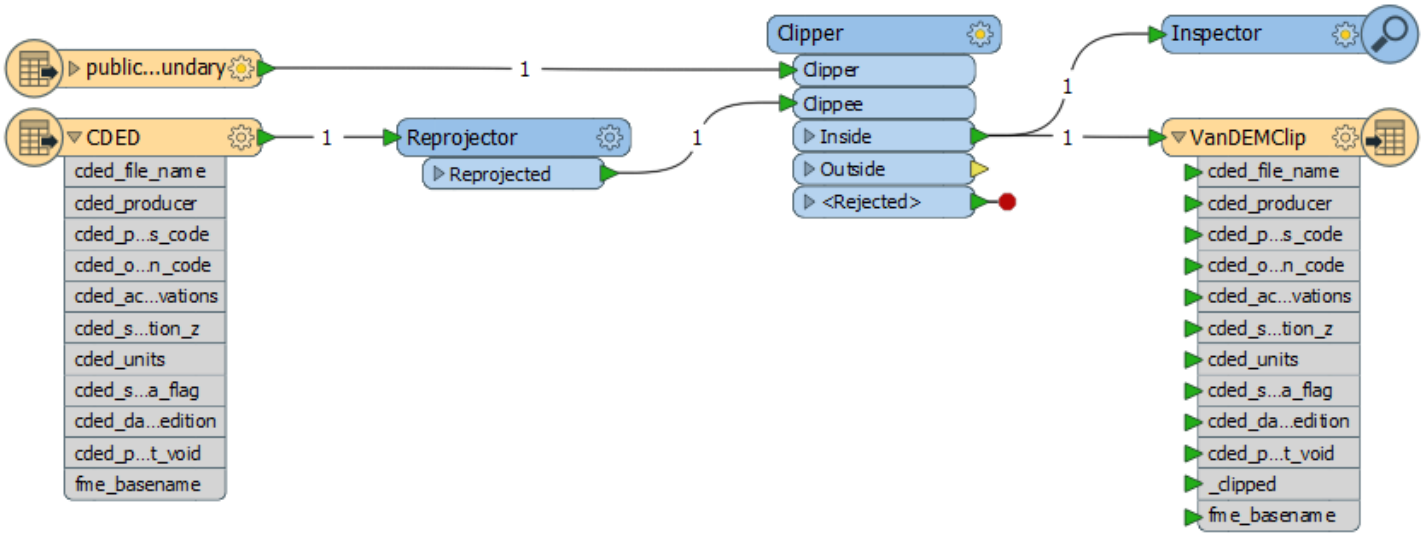
An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Documentation.

- After the Writer Feature Type is added the Writer parameters dialog will open. By default, the table will be placed in the “public” schema.
- In the General section, set the Table Name to VanDEMClip.
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create, then click OK.
- Once the PostGIS writer appears on the canvas, connect it to the Clipper:Inside port to complete the workspace.



10. Run the workspace

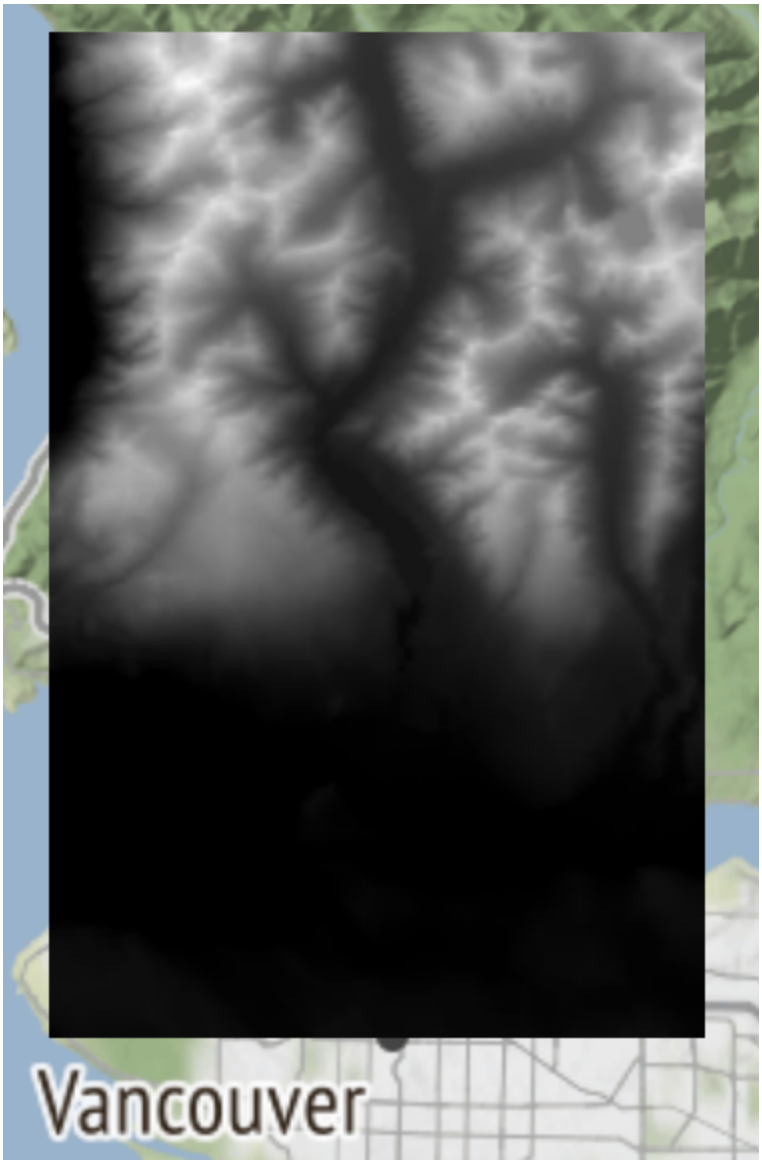
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.

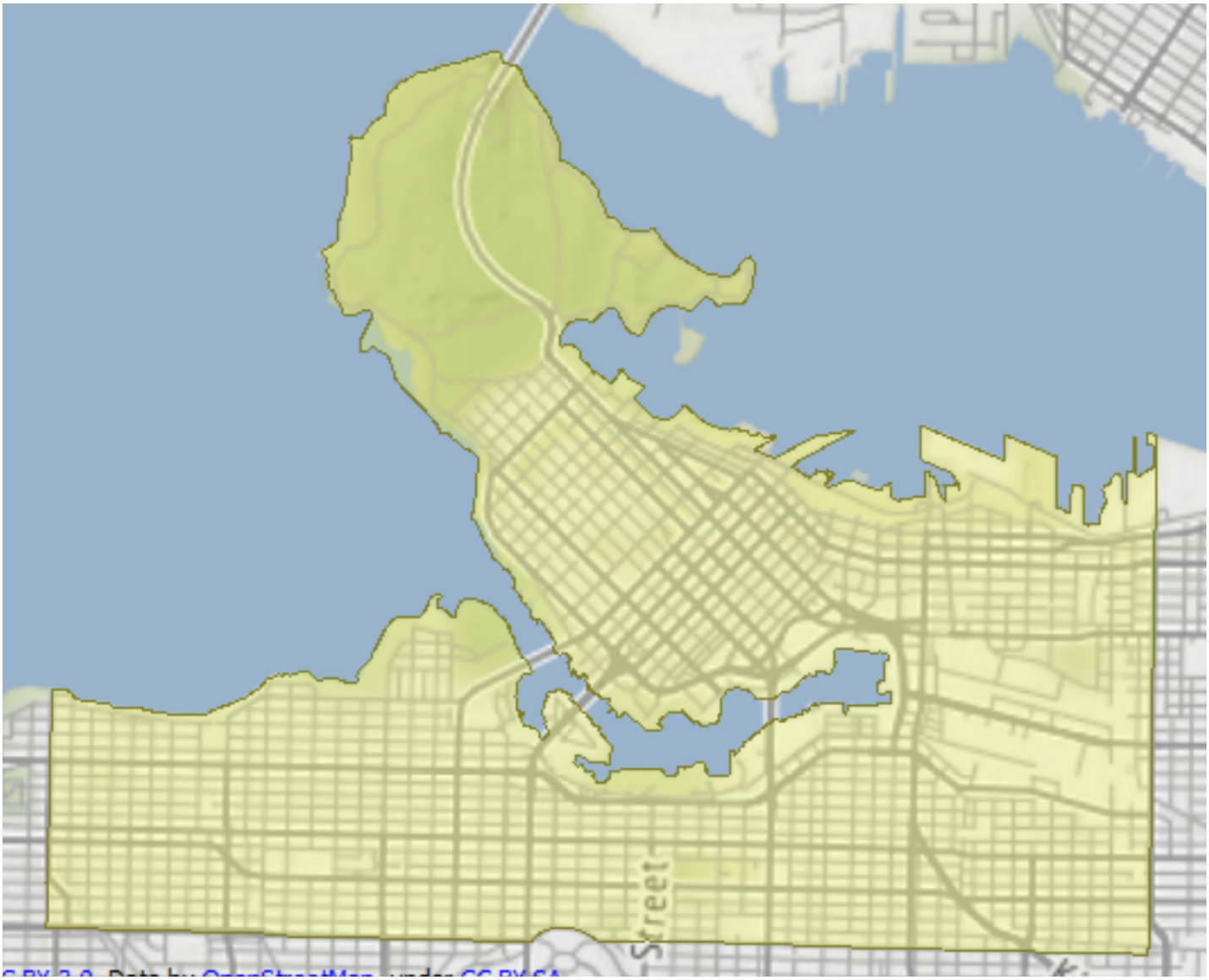


After running your workspace, the output will open in the FME Data Inspector - you will be viewing a digital elevation model of the Vancouver area.

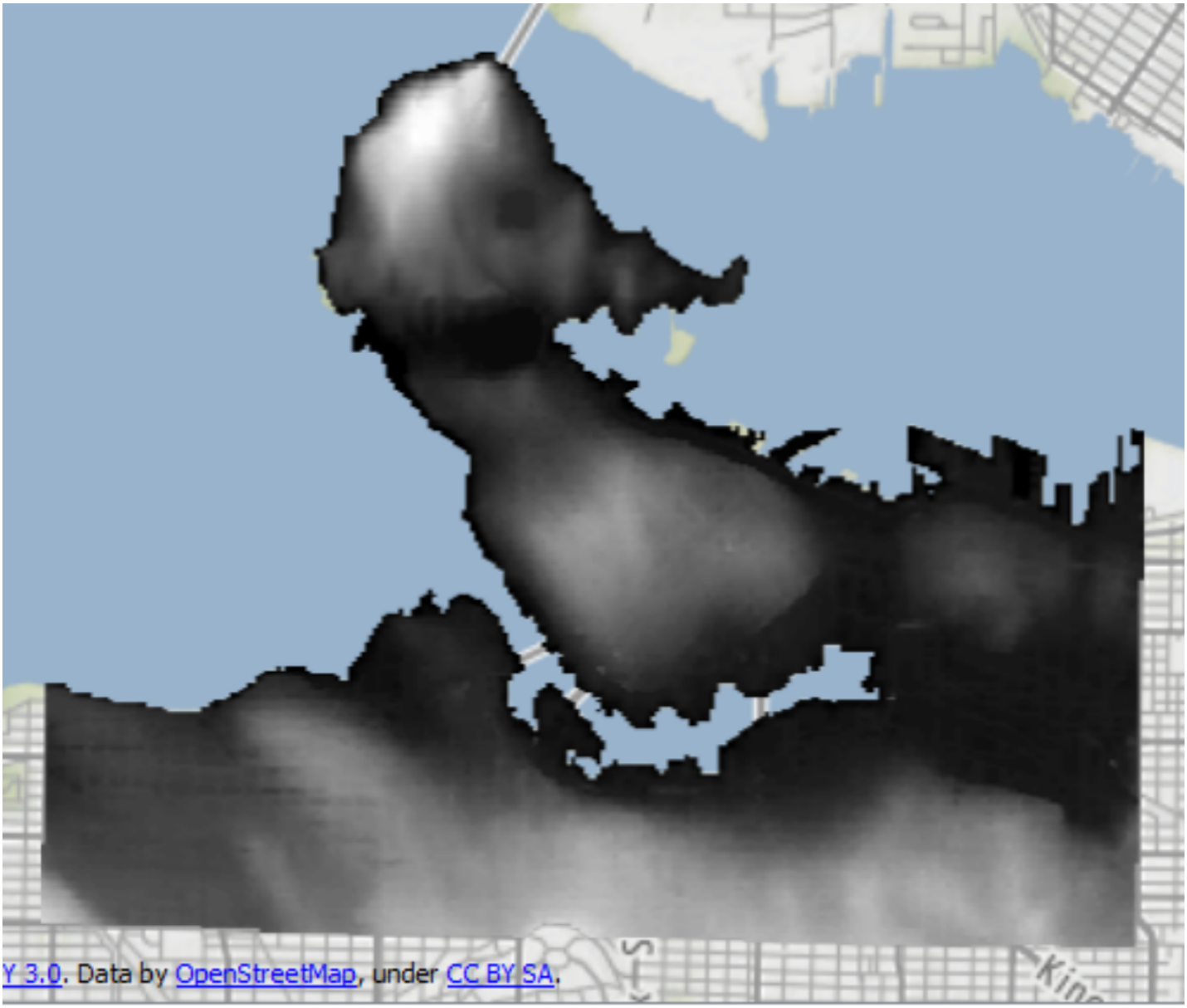
Results

Input


Vancouver



Output



Transformers

- Clipper - Performs a geometric clipping operation (sometimes called a cookie cutter).
- Inspector- Sends features to the FME Data Inspector for display.
- Reprojector - Reprojects feature coordinates from one coordinate system to another.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Convert from Raster to Vector

Overview

In this tutorial, you will learn how to convert raster data into polygons using the RasterToPolygonCoercer transformer.

Downloads

raster-to-vector.fmwt

vancouverparks.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you want to convert a 1m raster of urban and park areas into a city boundary polygon that will be stored in the KML format.

Instructions

1. Start FME Workbench

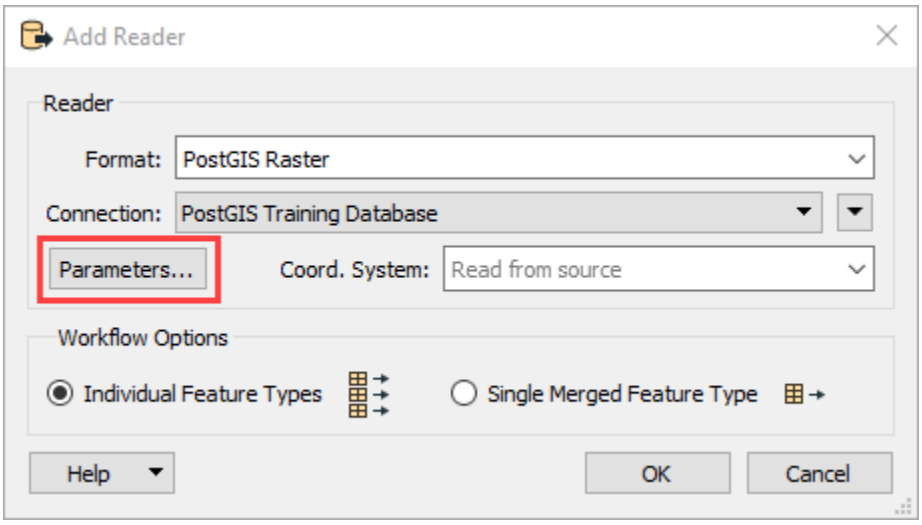
- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

3. Add a PostGIS Raster Reader

- Start typing “PostGIS Raster” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the VancouverParks table from the Table List.



4. Add a RasterToPolygonCoercer

The RasterToPolygonCoercer creates polygons from input raster features. One polygon is output for each contiguous area of pixels with the same value in the input raster.

- Select the VancouverParks Reader Feature Type, then start typing “RasterToPolygonCoercer” to bring up the list of FME Transformers in the Quick Add Search. Then select the RasterToPolygonCoercer from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas. This will automatically connect the VancouverParks reader to the RasterToPolygonCoercer.

5. Add an Inspector

- Next, add an Inspector after the RasterToPolygonCoercer and run the workspace. This will automatically open the vector dataset in the FME Data Inspector after the translation has run.

Table View

inspector [FFS] - Inspector_2

	fme_basename	_label
158	fmedata	0,0,0,255
159	fmedata	0,0,0,0
160	fmedata	0,0,0,255
161	fmedata	0,0,0,255
162	fmedata	0,0,0,255
163	fmedata	0,0,0,0

Notice the raster was properly converted to polygon features which are classified by the _label attribute. The selected area identifies the raster background (_label 0,0,0,0) which will be removed using a Tester transformer in the following step. The RasterToPolygonCoercer creates the _label attribute that represents the band values covered by that polygon (in this case R, G, B, A). In the original dataset, there are three classified cell values: urban, park, and background. Although the dataset appears to be clipped to the land boundary, the background cells still have values (0, 0, 0, 0) but have been made transparent. Since the background cells still have values, they made into polygons by the RasterToPolygonCoercer.

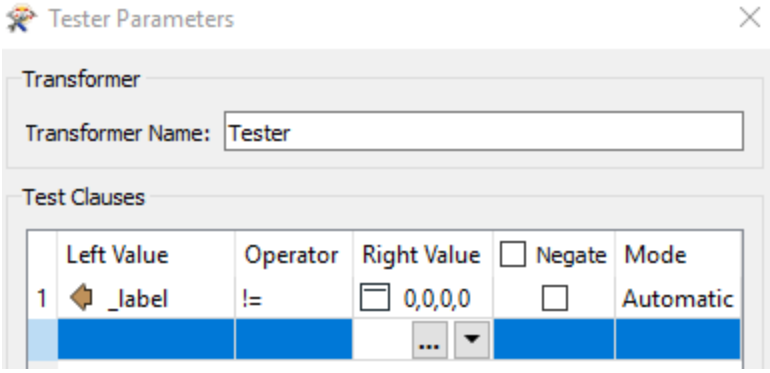
- Return to the FME Workbench and delete the Inspector that is connected to the RasterToPolygonCoercer as it is no longer needed.

6. Remove the Raster Background

- Add a Tester after the RasterToPolygonCoercer
- Once the Tester has been added, open the transformer parameters by double-clicking it or clicking on the red gear icon.

Recall from the previous step (as indicated in the screenshot above) that the raster background had a value of 0,0,0,0 for the _label attribute.

- Start by adding a Test Clause. Click on the empty row in the Left Value cell and click on the drop-down arrow to select an attribute from the workspace. Select the _label attribute from the Attribute Value list.
- Next, set the Operator to does not equal by selecting the != operator from the drop-down list.
- Lastly, click on the empty Right Value cell to enter in text. Enter the value by typing “0,0,0,0” into the Right Value to get rid of any _label attributes with the value 0,0,0,0 (i.e. the unwanted raster background), then click OK to accept the changes and close the Tester Parameters. The Tester Parameters should match the screenshot provided below.



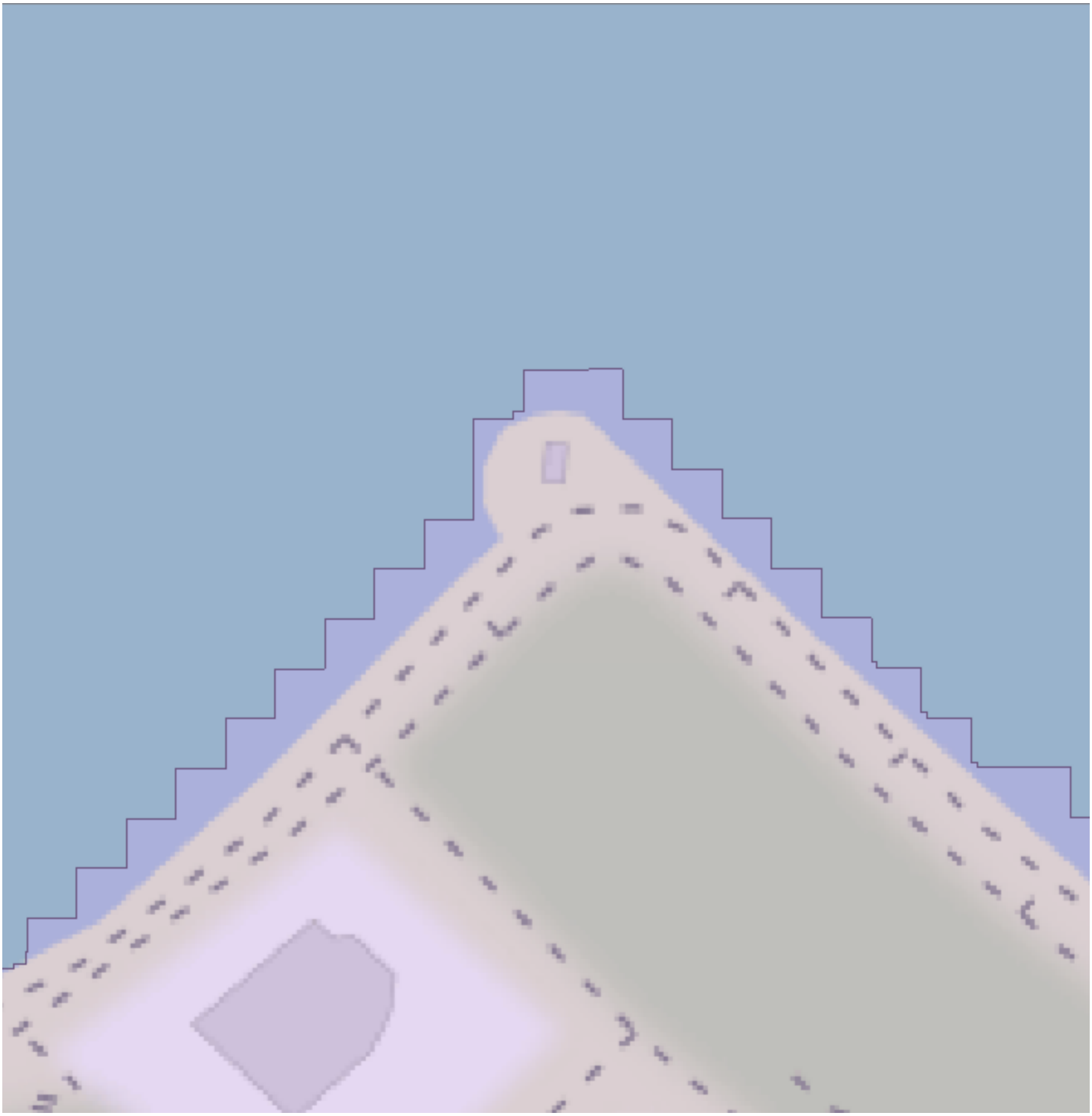
To learn more about the Tester and Test Clauses, see the Tester and TestFilter Transformers section in the FME Desktop Basic Training Manual.

7. Smooth the Polygon Edges

In this scenario, you only have to set the Generalization Tolerance. The larger the Generalization Tolerance, the smoother the line.

- Add a Generalizer after the Tester and open the transformer parameters.
- Set the value of the Generalization Tolerance to 10.

If you were to run the workspace and inspect the output in the FME Data Inspector at this stage, you would notice the polygon is very jagged as it follows the outline of the raster cells similar to the screenshot provided below:



To avoid this, you need to smooth the lines by reducing the density of coordinates by removing vertices. To learn more about smoothing lines using a generalizing algorithm, see the Generalizer Documentation.

Note: the unit of this value is controlled by the coordinate system of the feature(s) that were read in.

Generalization Tolerance 1:

Generalization Tolerance 10:

8. Add an Inspector

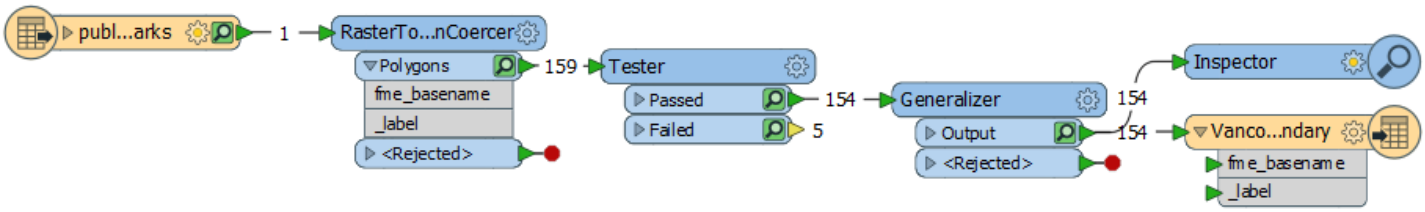
- Next, add an Inspector after the Generalizer. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

9. Add a Google KML Writer

- Similar to adding a reader, type “KML” then select the Google KML format from the list of Writers.
- Specify a file name and set the destination folder for the KML file by clicking on the Dataset ellipsis. After specifying a folder location and filename, set the Table Definition to Automatic and click OK.

10. Run the Workspace

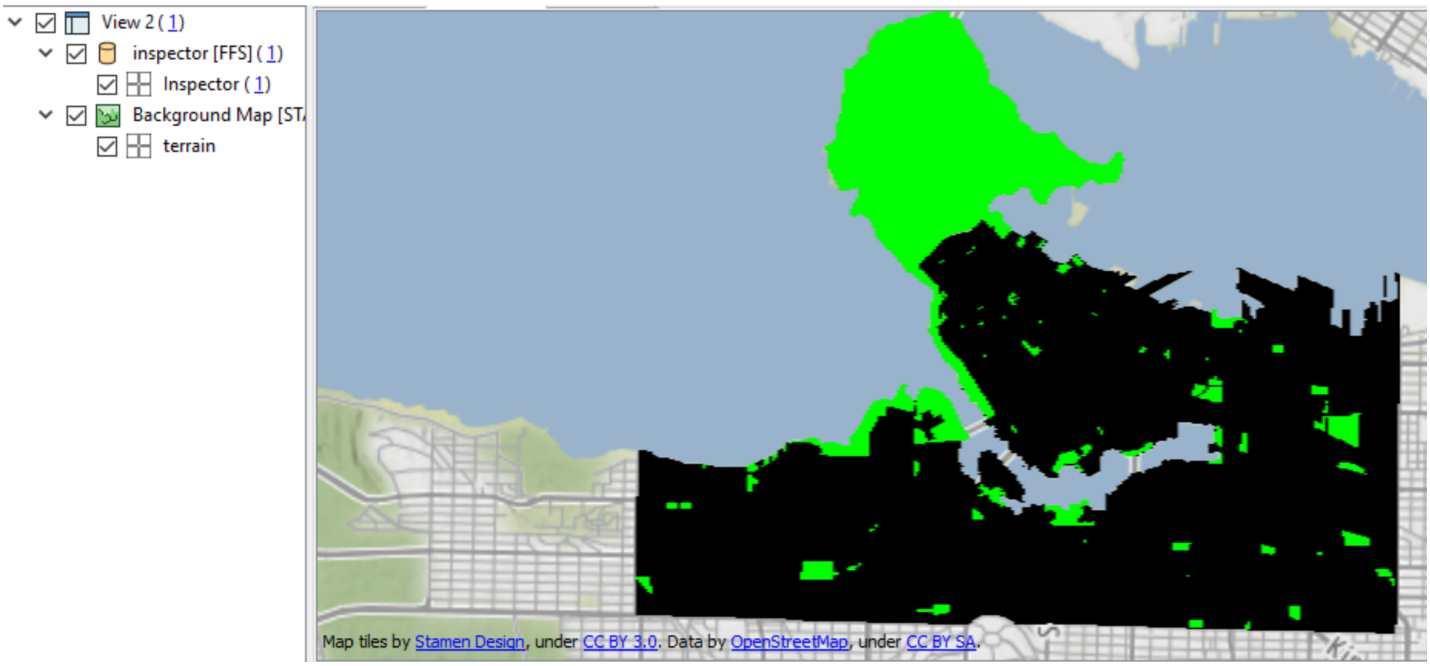
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



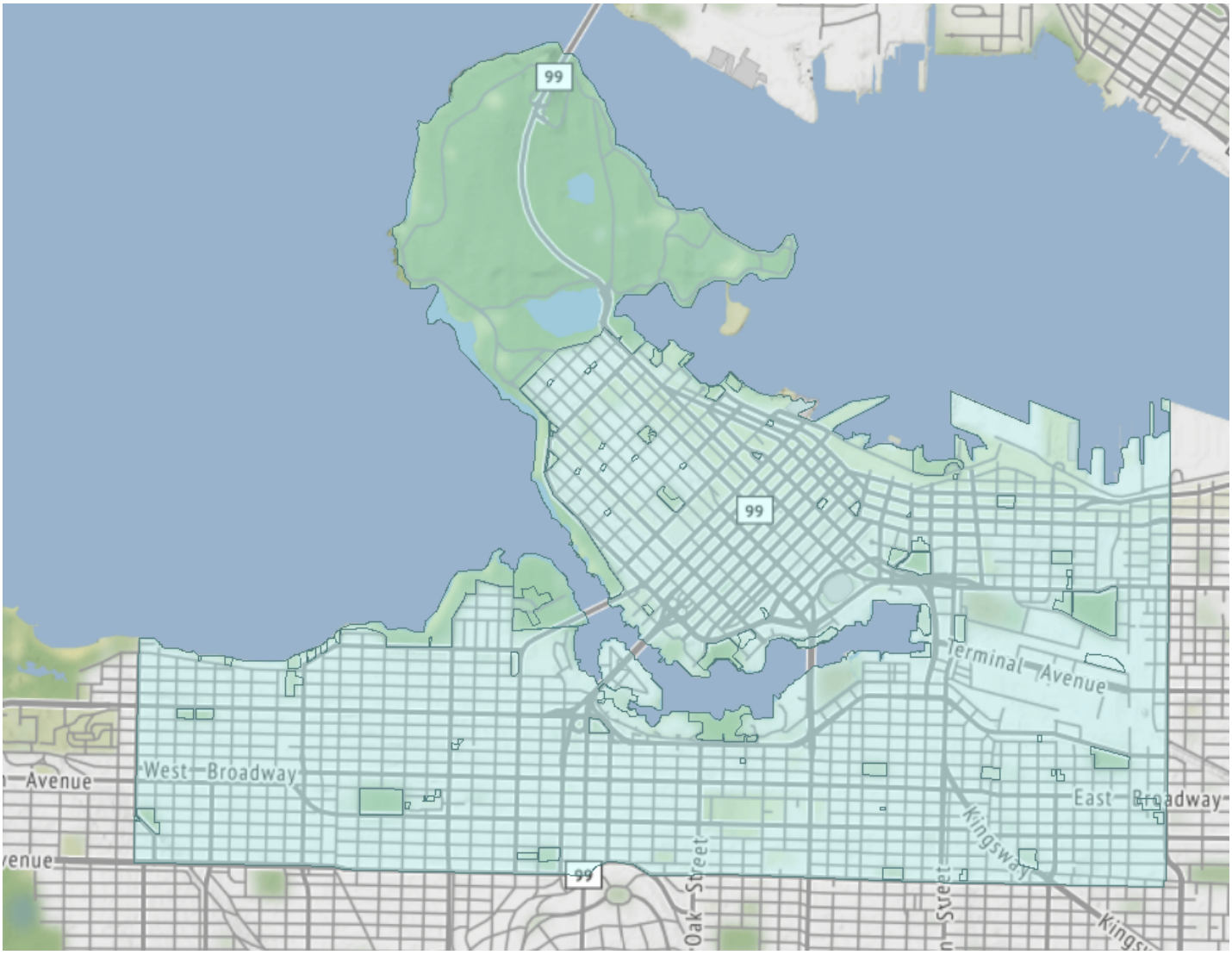
After running the translation, you will have a new Google KML file with a single polygon for the Vancouver land boundary. Since you attached an Inspector to the last transformer on the canvas, the output will automatically open in the FME Data Inspector.

Results

Input



Output



Transformers

- Generalizer- Transforms or measures geometry features based on a specified algorithm.
- Inspector- Sends features to the FME Data Inspector for display.
- RasterToPolygonCoercer - Creates polygons from input raster features.
- Tester - Evaluates one or more tests on a feature, and routes the feature according to the outcome of the test(s)

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Convert from Vector to Raster

Overview

In this tutorial, you will learn how to convert vector polygon features into raster data using the MapnikRasterizer, although this method also works for other geometries (such as point and lines). For a more detailed introduction on how to use the MapnikRasterizer transformer, see the Introduction to MapnikRasterizer article.

Downloads

vector-to-raster.fmwt

neighborhoodsgdb.zip

parksgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you want to create a raster map classified with land and park areas. Using Neighborhood and Park polygon features from the PostGIS training database, you will create a raster dataset with areas classified by RGB values (green for parks and black for urban).

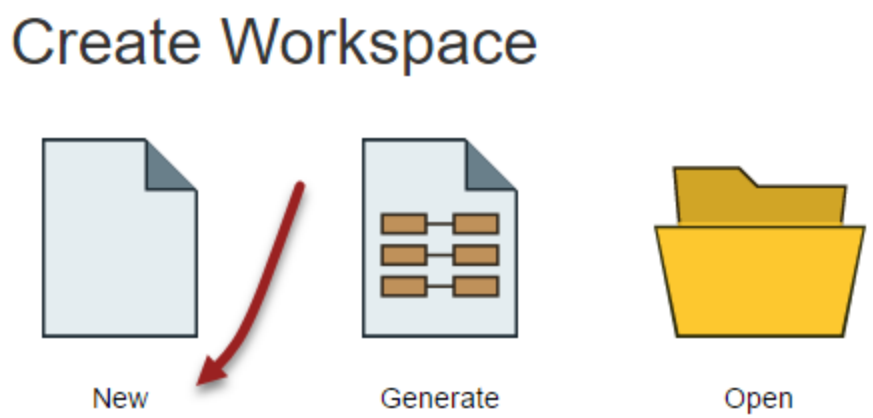
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

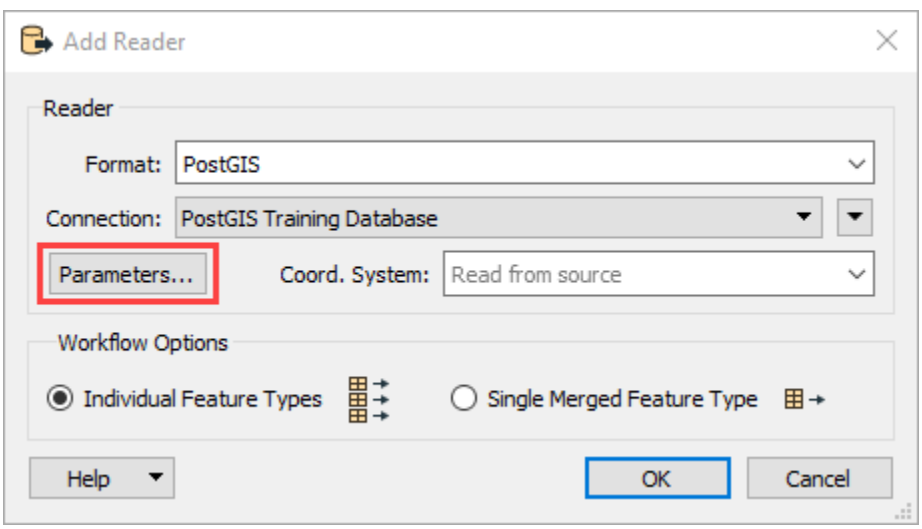
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

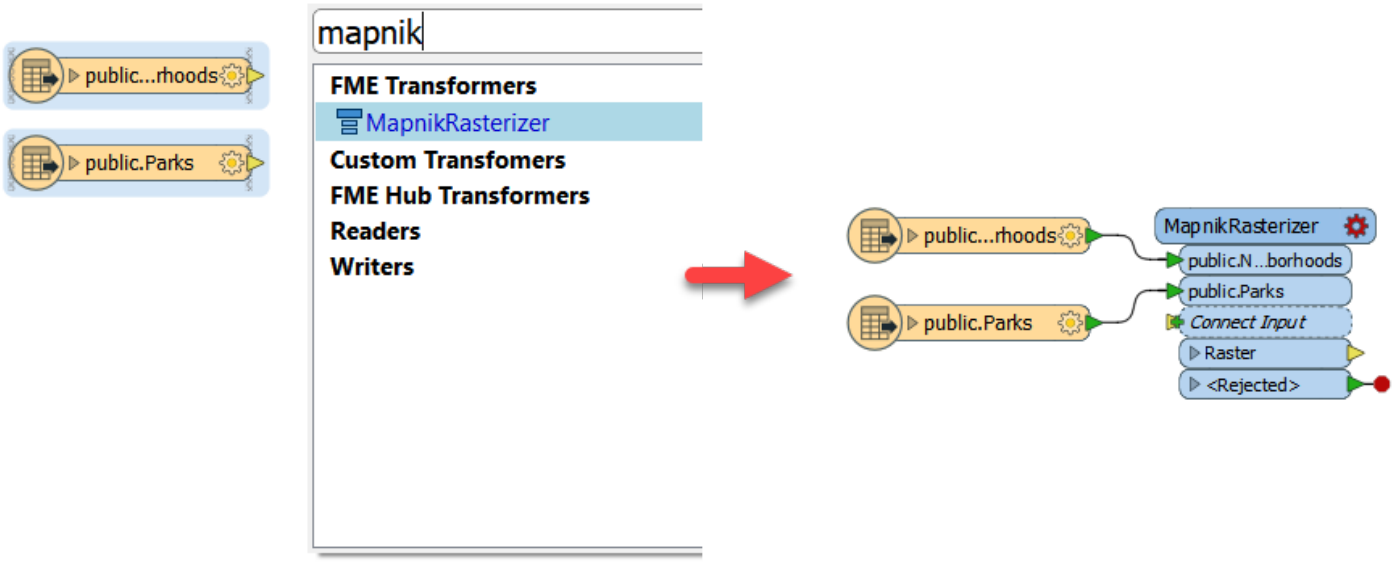
- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Parks and Neighborhoods tables from the Table List.



4. Add a MapnikRasterizer

The MapnikRasterizer allows you to convert point, line, and polygon features onto a new raster by dividing features into groups called layers. For more information see the MapnikRasterizer Documentation.

- Select the Neighborhoods and VancouverLandBoundary Reader’s then type “Mapnik” to bring up the list of FME Transformers in the Quick Add Search. Choose the MapnikRasterizer from the list of Transformers by double-clicking or using the arrow keys and the Enter key to add it to the workspace.

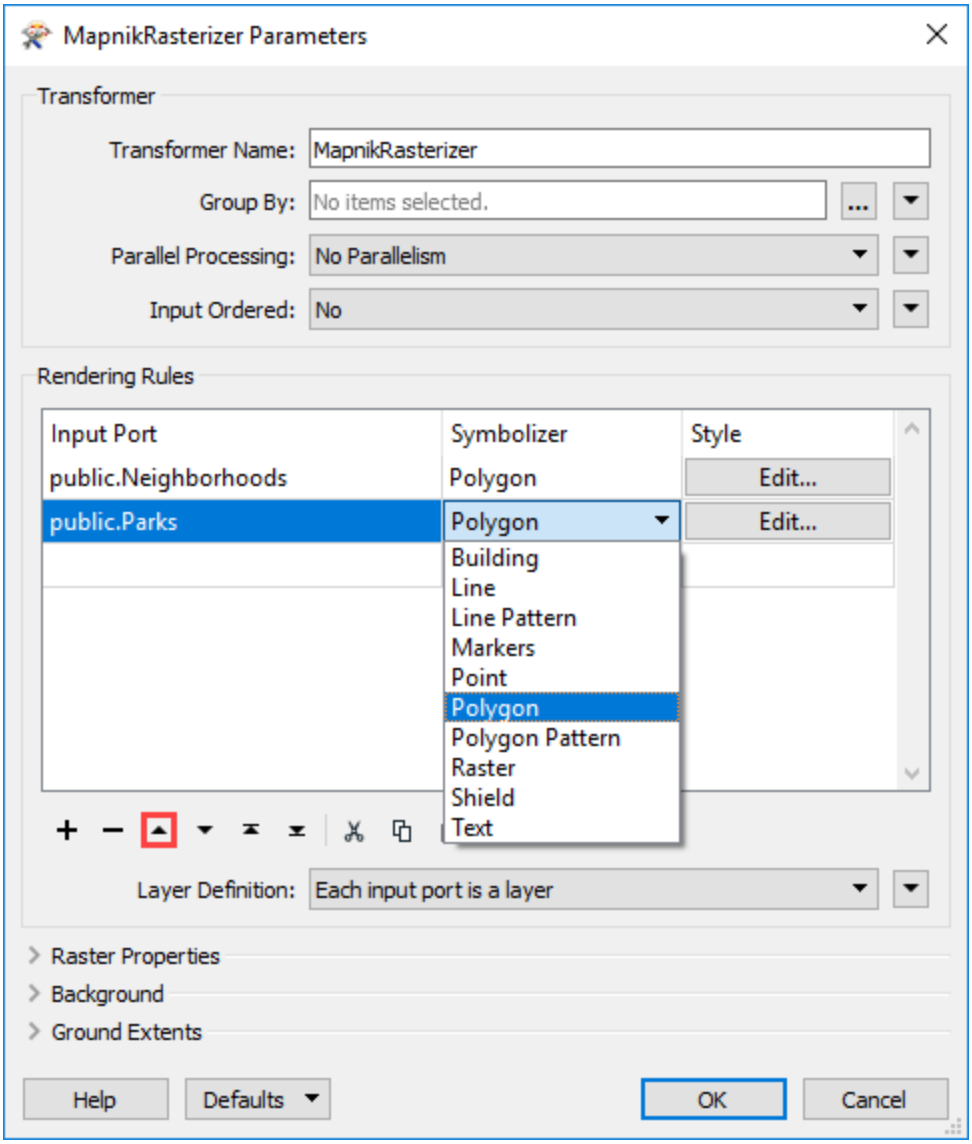


5. Set the MapnikRasterizer Parameters

- Once the transformer has been added, double-click the MapnikRasterizer or click on the red gear icon to open the transformer parameters.

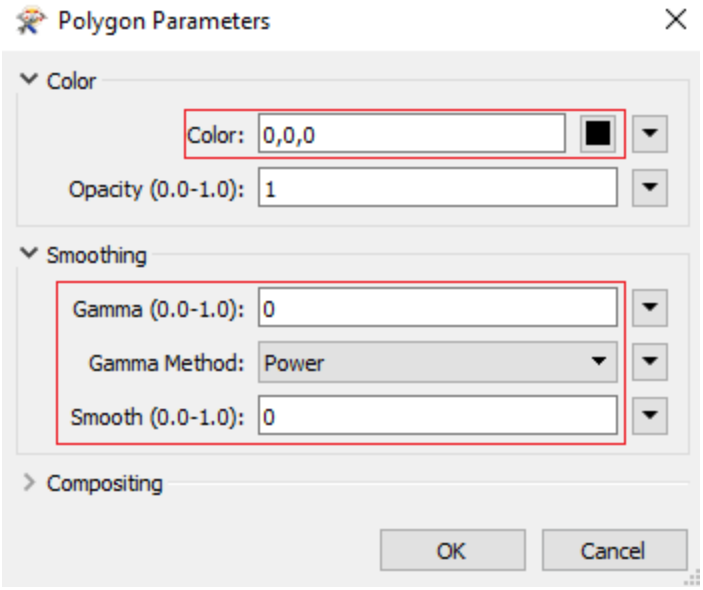
Notice the two connections are listed in the Rendering Rules; however, they are set as Line symbols.

- Click on “Line” to change both the Rendering Rules Symbolizer to Polygon.
- **Important**: ensure the rendering rules are ordered as they are in the screenshot above otherwise the urban cells will be created on top of the park cells. The urban cells need to be classified first, followed by the park cells so they appear on the output raster. You can adjust the ordering of the rendering rules using the move up/down arrows below the Rendering Rules box.



6. Classify Neighborhood Cells

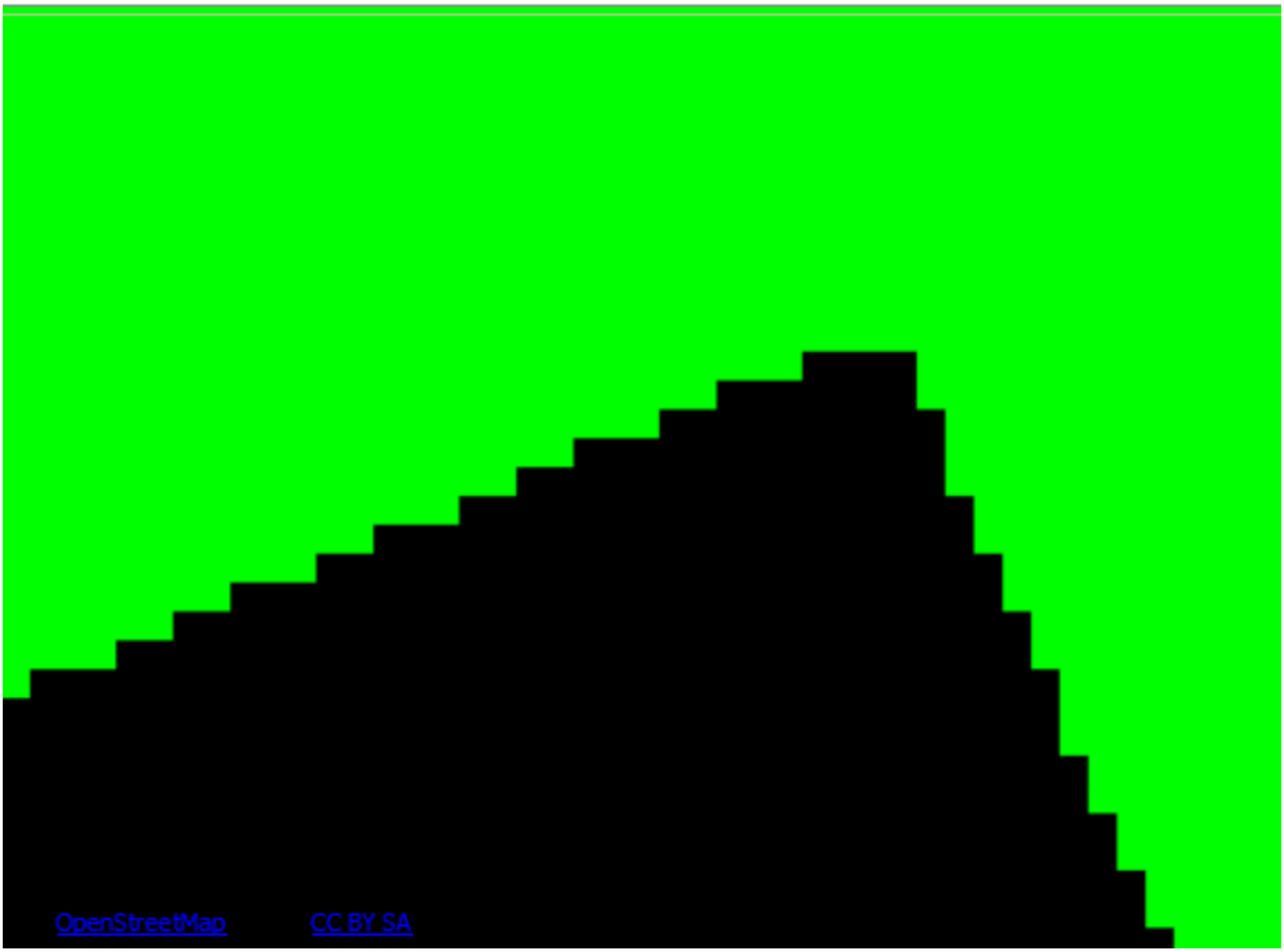
- Click Edit in the Style column of the Neighborhoods row to modify the polygon parameters. This is where you will set the color, smoothing, and compositing for the raster output. In this example, you want the urban cells from the neighborhood polygons to be black and the park cells from the park polygons to be green.
- Click on the color selector and choose black from the list of basic colors. Alternatively, you can set colors by using RGB, HSV, or HTML values. In this example, the RGB values are Red: 0, Green: 0, Blue: 0. Click OK once the color is set.
- Next, in the Polygon Parameters, expand the Smoothing section by clicking on the drop-down arrow. Smoothing is used to control whether the output will follow Boolean or Fuzzy logic.
- Set the smoothing Gamma to 0 then click OK to accept the Neighborhoods Polygon Parameters.



A Smoothing Gamma value of 1 uses Fuzzy logic which is useful for areas with movement or uncertainty such as a shoreline. Alternatively, a Smoothing Gamma value of 0 uses Boolean logic which is useful when you want to classify cells with 100% certainty. This exercise will use Boolean Logic so cells will be classified as either urban or park - this will prevent cells from being classified as 20% park and 80% urban.

Fuzzy Logic

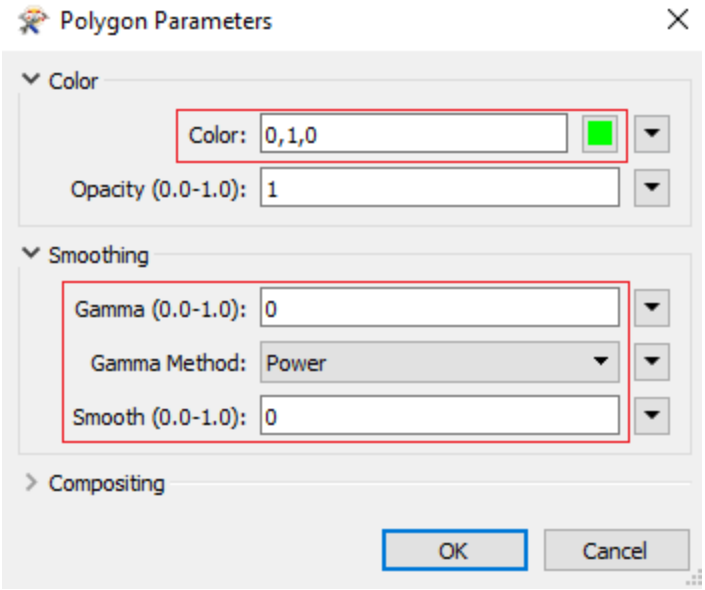
Boolean Logic



7. Classify Park Cells

Next, you will classify the park cells. Similar to classifying urban cells, modify the Polygon Parameters by clicking Edit in the Parks row.

- Set the color to green by clicking on the color selector. Alternatively, you can type the “0,1,0” into the color text box - this will give you the RGB value of Red: 0, Green 255, Blue: 0.
- Again, expand the Smoothing section and set the Smoothing Gamma to 0.



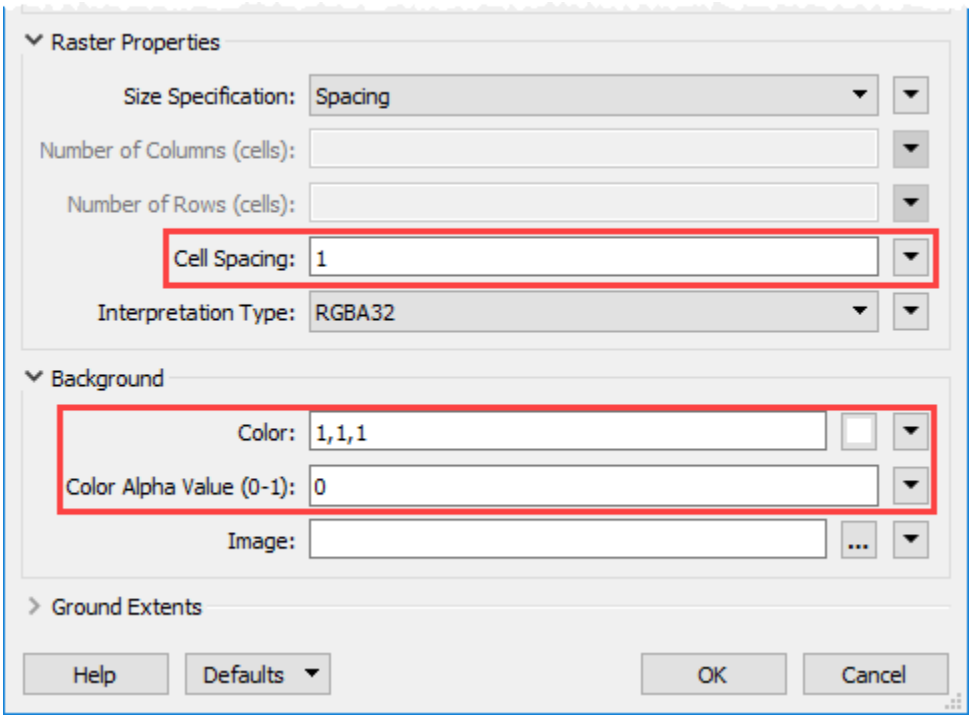
8. Set the Raster Properties

- Lastly, you have to define the cell size for the raster we will be producing. Set the Size Specification to Spacing from the drop-down list in the Raster Properties section.

The Cell Spacing parameter specifies the width of the output raster cells, measured in ground units. *Note: smaller cells provide greater resolution, however, they typically take longer to load when viewing large extents (i.e. a cell spacing of 1 will take longer to load than 5) and also have large file sizes.*

- Set the Cell Spacing to 1.
- Lastly, expand the Background section and set the Color Alpha Value to 0.

This will effectively remove the background cells by making them transparent since you are only interested in creating raster cells for areas that are either urban or park in this exercise. For more information see the [Setting NoData and Adding Alpha Bands to Remove Black Borders]() article.



9. Add an Inspector

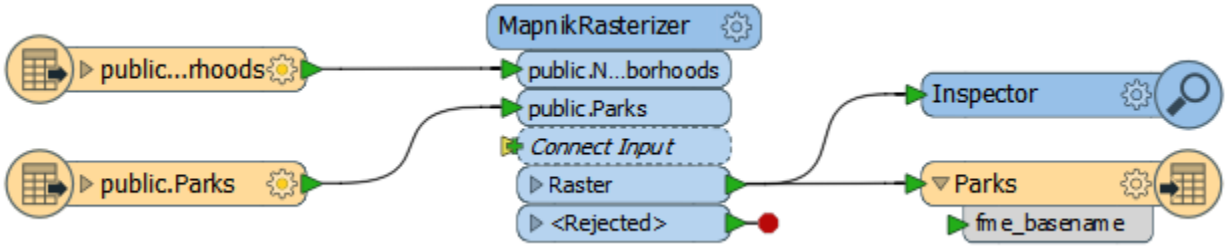
- Next, add an Inspector and connect it to the MapnikRasterizer:Raster port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

10. Add an ER Mapper ECW Writer

- Similar to adding a reader, type “ECW” then select the ER Mapper ECW format from the list of Writers.
- Specify a folder for the ECW file by clicking on the Dataset ellipsis. After specifying a folder location, set the Table Definition to Automatic then click OK.
- Next, set the Raster File Name in the Feature Type Parameters dialog then click OK to add the ECW writer to the canvas.
- Lastly, connect the ECW to the MapnikRasterizer:Raster port.

11. Run the Workspace

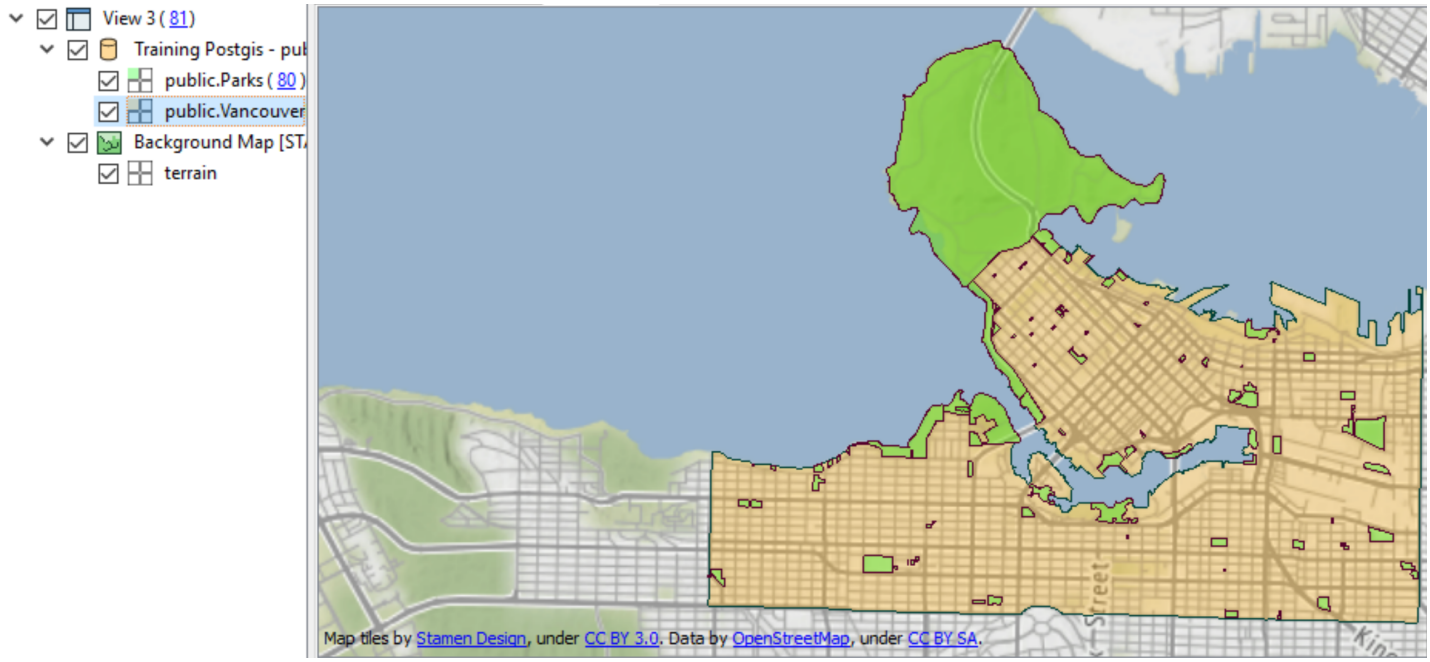
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



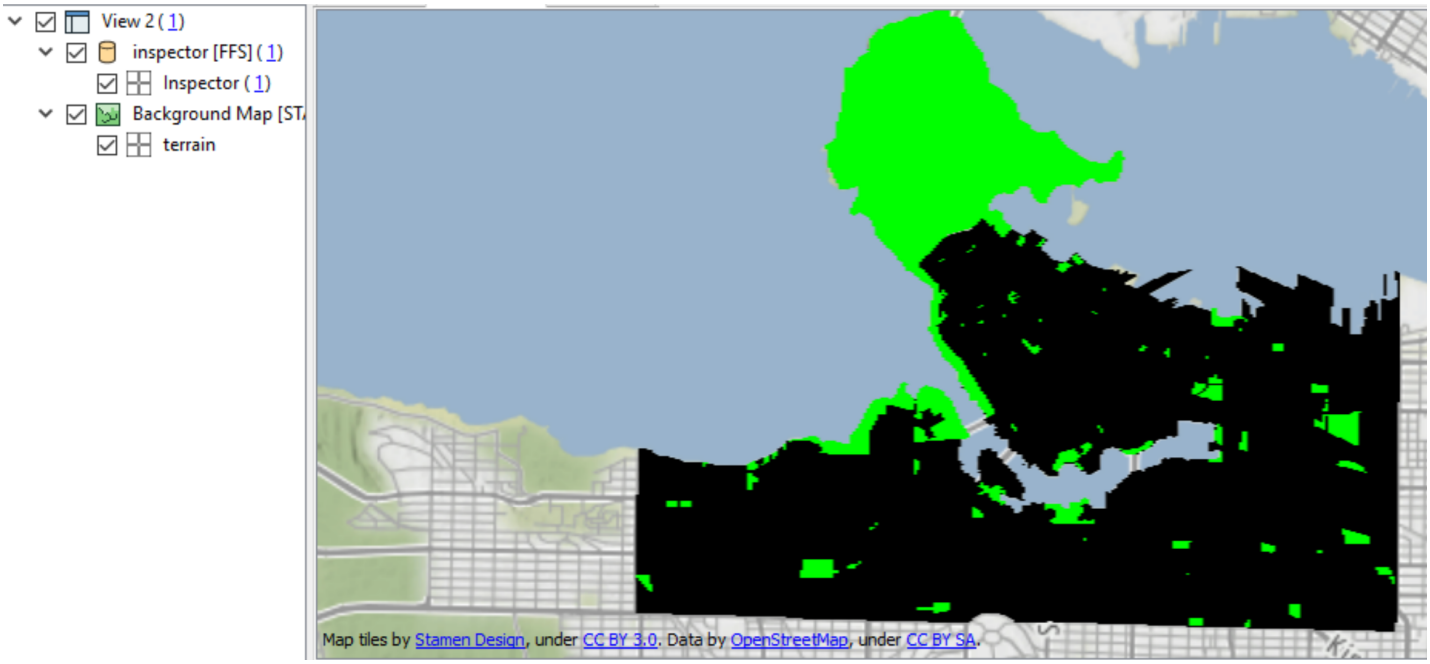
If you are getting unexpected results, you may need to reset the PostGIS table(s) that you read in. For more information see the [Resetting the PostGIS Training Database]() article.

Results

Input



Output



Transformers

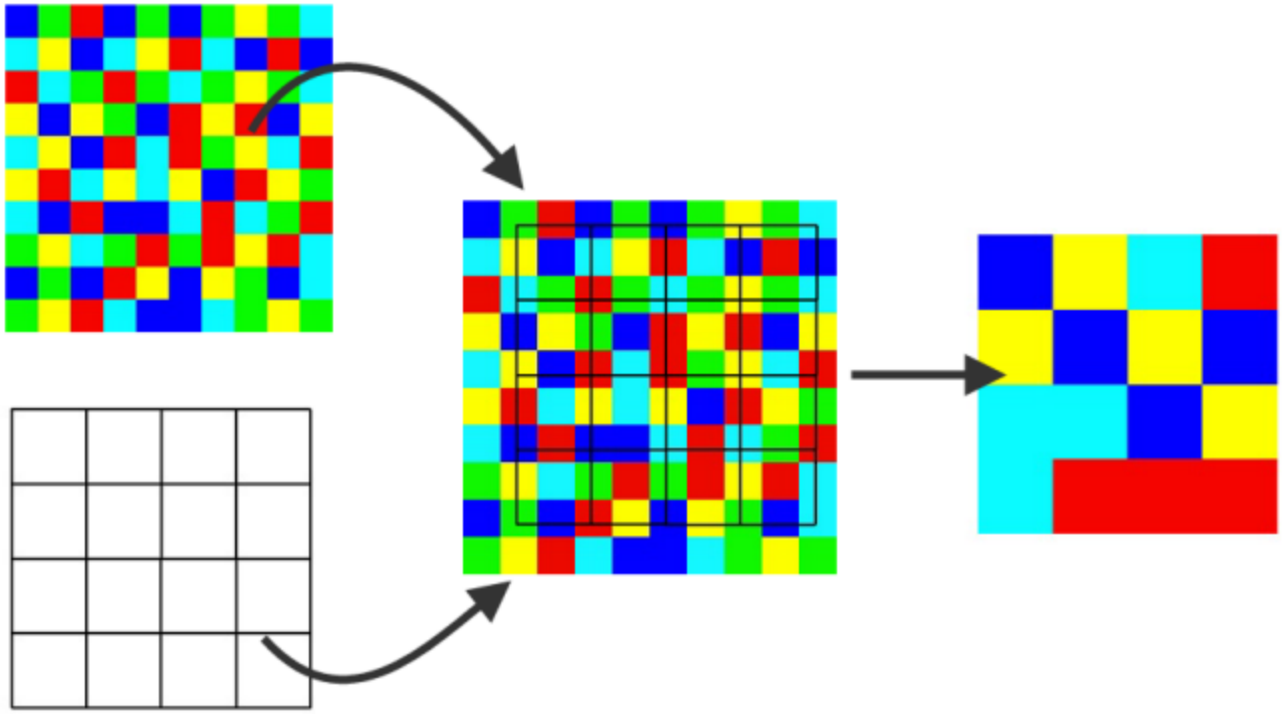
- Inspector- Sends features to the FME Data Inspector for display.
- MapnikRasterizer- Draws input point, line, polygon, and raster features onto a raster using the Mapnik toolkit.

Data Attribution

Modify Raster Cell Size by Resampling

Overview

In this tutorial, you will learn how to modify raster cell sizes using the FME RasterResampler transformer. The RasterResampler resamples an input raster based on user-specified parameters (raster dimension, cell size, or percentage of size) and an interpolation method (Nearest Neighbor, Bilinear, Bicubic, Average 4, or Average 16) to merge cells.



Downloads

raster-resampling.fmwt

dem-full.zip

Exercise

In this scenario, you want to change the spatial resolution of your raster dataset and set rules for aggregating or interpolating values across the new pixel sizes. This will allow you to change the cell size without changing the extent of the raster dataset. The objective of this exercise is to downsample the raster in order to reduce the file size and improve performance when displaying the raster.

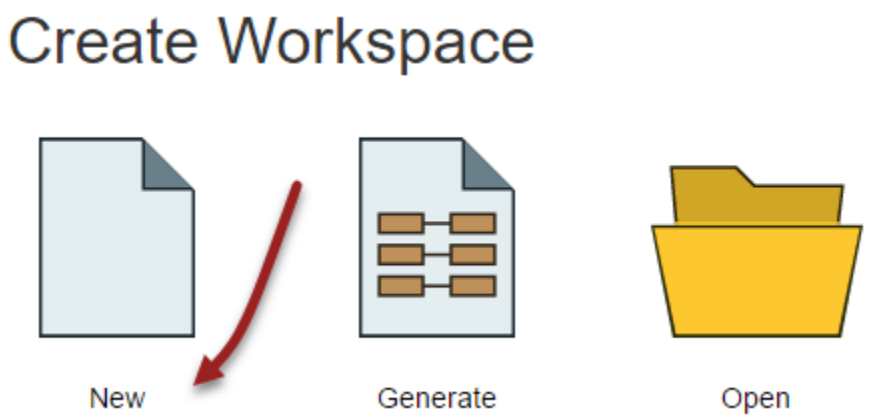
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a Canadian Digital Elevation Data (CDED) Reader

- Start typing "CDED" without anything selected on the canvas to bring up the list of FME Transformers in the Quick Add Search. Then select the Canadian Digital Elevation Data format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis and navigate to the DEM-FULL.dem found in the DEM-FULL.zip file, or you can read in the zipped file. Once selected, click Open, then click OK to add the reader to the canvas.

4. Add a RasterResampler

- Add a RasterResampler to the canvas by typing "RasterResampler" to bring up the list of FME Transformers in the Quick Add Search. Select the RasterResampler from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the CDED Reader to the RasterResampler.

5. Specify Cell Size and Interpolation Type

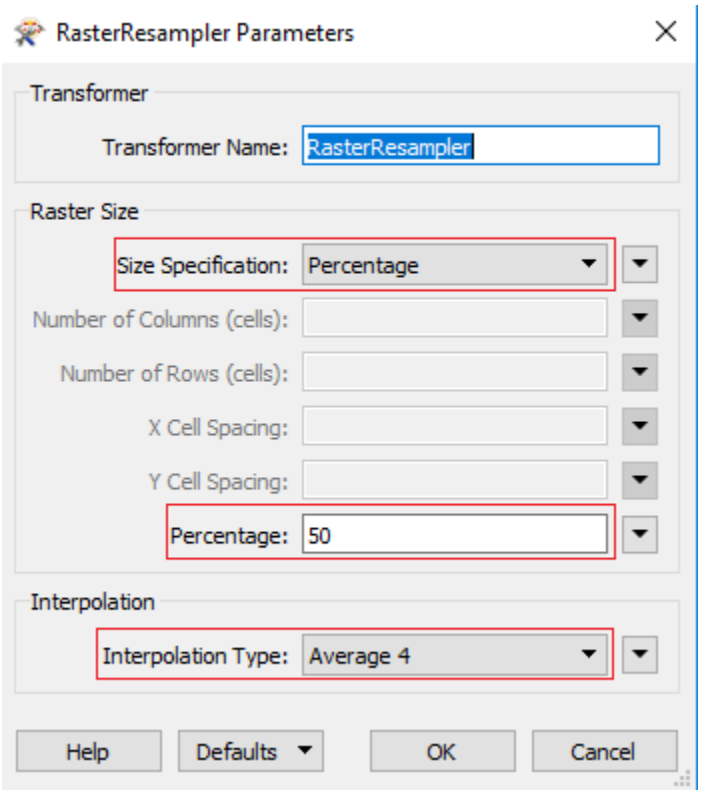
- Once the RasterResampler has been added, double-click the RasterResampler or click on the gear icon to open the transformer parameters dialog.
- In the Parameters dialog, set the Size Specification to Percentage by selecting it from the drop-down list. This will allow you to resize by a percentage of the input raster.

In this exercise, you want to reduce the number of cells in the raster by 50% because the original raster image isn't loading as quickly as you would like.

- Set the Percentage to 50 by typing it in the Percentage text box.
- Next, set the Interpolation Type to Average 4.

Average 4 and Average 16 are useful interpolation types for numeric rasters like DEMs and produce higher quality images than Bilinear and Nearest Neighbor. For more information see the RasterResampler.

- Ensure the RasterResampler dialog looks similar to the screenshot below, then click OK to accept the changes and close the dialog.



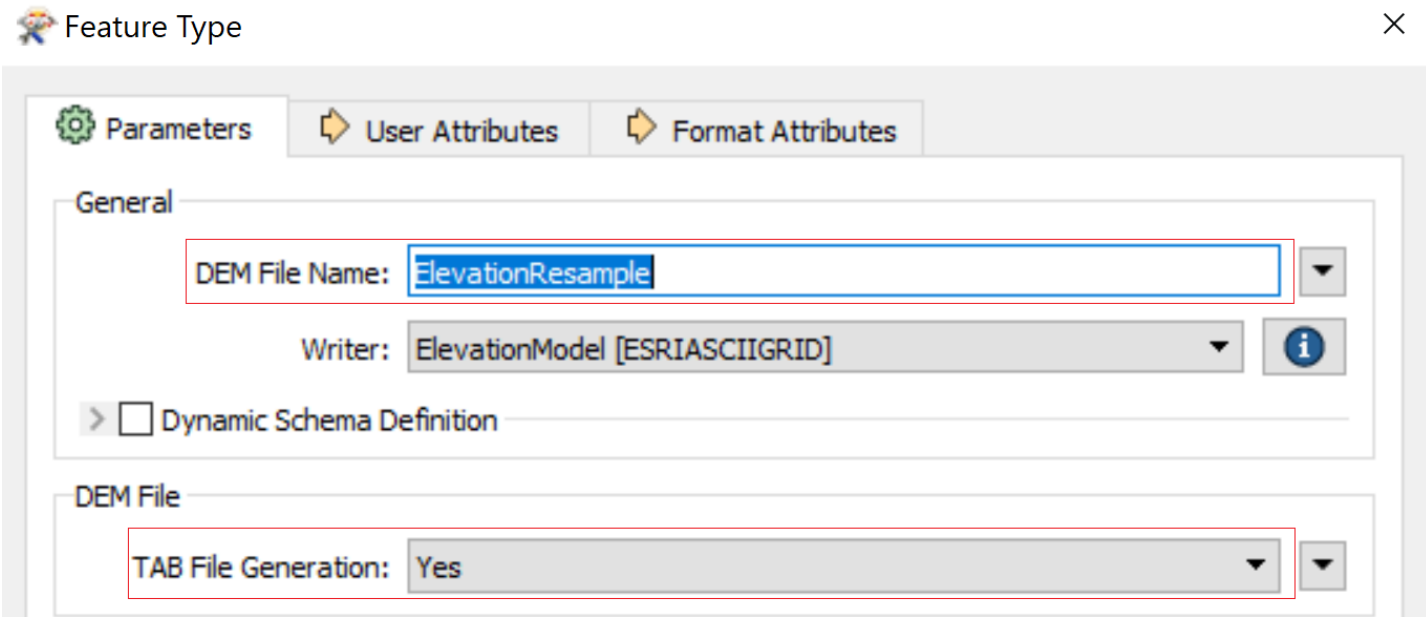
Note: Another common practice is to create raster pyramids to improve performance. To find out more about using the RasterPyramider in FME see the Raster Pyramiding Example article.

6. Add an Inspector

- Add an Inspector and connect it to the RasterResampler:Resampled port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

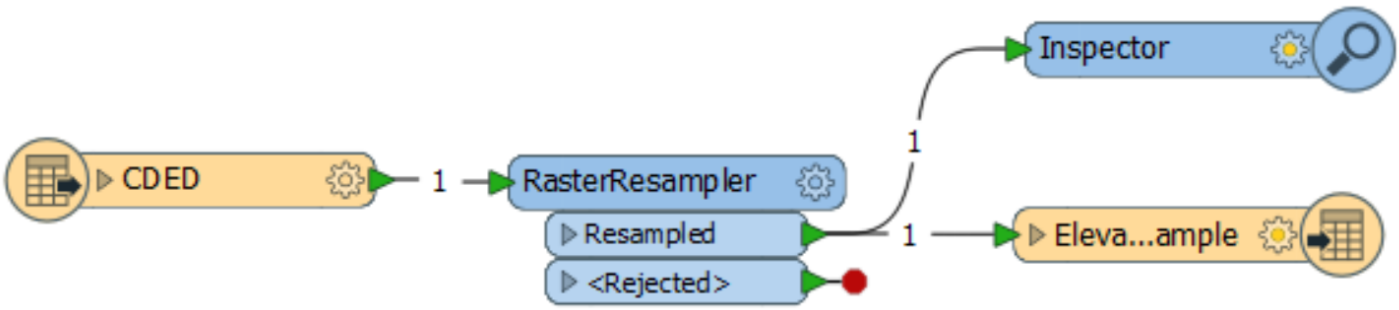
7. Add an Esri ASCII Grid Writer

- Similar to adding a reader, type “Esri ASCII Grid” then select the Esri ASCII Grid format from the list of Writers.
- Specify a folder for the Esri Grid file by clicking on the Dataset ellipsis. After specifying a folder location, set the Table Definition to Automatic then click OK.
- Next, set the Raster File Name in the Feature Type Parameters dialog. Additionally, in the DEM File section, set the TAB File Generation to Yes, then click OK to add the Grid writer to the canvas. TAB files are used to store control points, a coordinate system, and user attribute. To learn more about TAB files see the Raster Documentation.
- Lastly, connect the Grid writer to the RasterResampler:Resampled port.



8. Run the Workspace

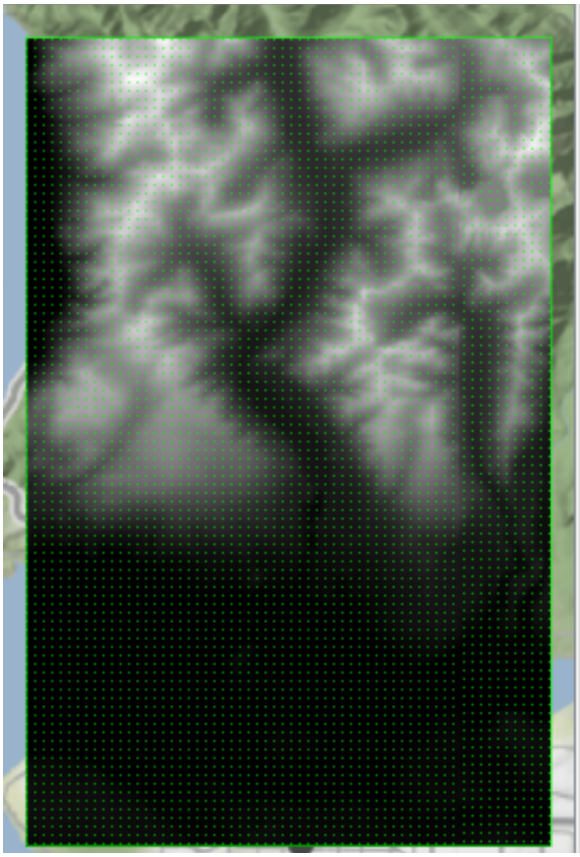
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



- After the translation has run, the output will be displayed in the FME Data Inspector. Notice the resolution of the raster has been reduced to 601 x 601px from the original 1201 x 1201px.

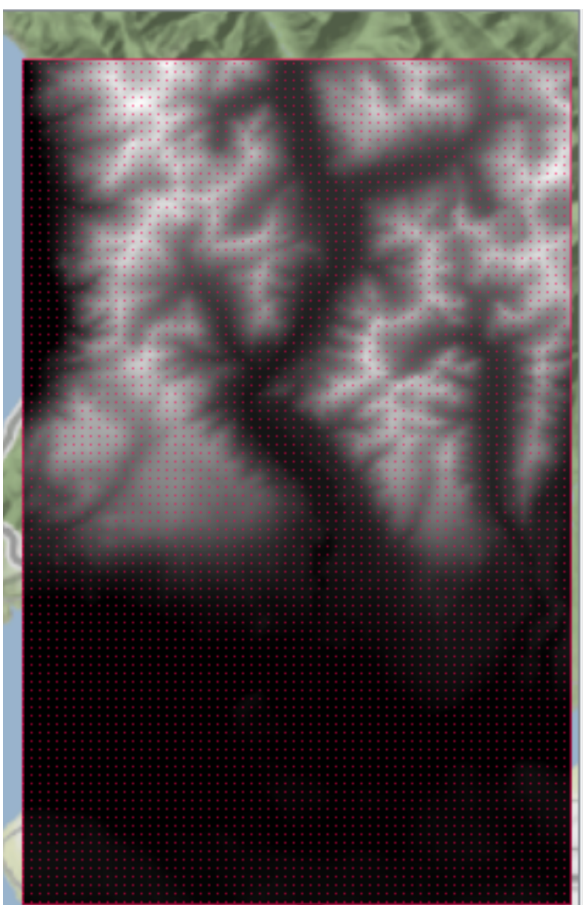
Results

Input



Property	Value
Feature Type	CDED
Coordinate System	LL83-SECONDS
Dimension	2D
Number of Vertices	5
Min Extents	-443700.375, 177299.625
Max Extents	-442799.625, 178200.375
Attributes (14)	
IFMERaster	
Min Extents	-443700.375, 177299.625
Max Extents	-442799.625, 178200.375
Resolution (Column...	1201 x 1201 Pixels
Origin	-443700.375, 178200.375
Spacing	0.75, 0.75
Rotation in Radian CCW	0, 0
Cell Origin	0.5, 0.5
Affine Transform	0.75, 0, -443700.375, 0, -0.75, 178200.375
Number Of Bands	1 Band
Band 0 (INT32)	
Number Of Ro...	1201
Number Of Co...	1
Nodata Value	-32767

Output



Feature Information

Property	Value
Feature Type	Inspector
Coordinate System	LL83-SECONDS
Dimension	2D
Number of Vertices	5
Min Extents	-443700.375, 177299.625
Max Extents	-442799.625, 178200.375
Attributes (18)	
IFMERaster	
Min Extents	-443700.375, 177299.625
Max Extents	-442799.625, 178200.375
Resolution (Column...	601 x 601 Pixels
Origin	-443700.375, 178200.375
Spacing	1.4987520798668885, 1.4987520798668885
Rotation in Radi...	0, 0
Cell Origin	0.5, 0.5
Affine Transform	1.4987520798668885, 0, -443700.375,...
Number Of Bands	1 Band
Band 0 (INT32)	

Transformers

- RasterResampler - Resamples an input raster using the desired dimensions, the desired cell size in ground units, or a percentage of the size.

Data Attribution

Data used in this tutorial originates from open data made available by the Government of Canada. It contains information licensed under the Open Government Licence - Canada.

Changing Raster Values by Geographic Feature

Overview

In this tutorial, you will learn how to select an area and change the values of raster cells by overlaying a polygon onto a raster dataset. This can be easily accomplished in FME using the MapnikRasterizer, an existing raster dataset, and point, line, or polygon features.

Downloads

changing-raster-values-by-geographic-feature.fmwt

vancouverland.zip

parksgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you will reclassify raster cells based on areas where park polygons overlay urban cells. This will change the band values from [Band 0: 0], [Band 1: 0], [Band 2: 0], [Band 3: 255] to [Band 0: 0], [Band 1: 255], [Band 2: 0], [Band 3: 255] where park polygons overlay raster cells in the VancouverLand dataset.

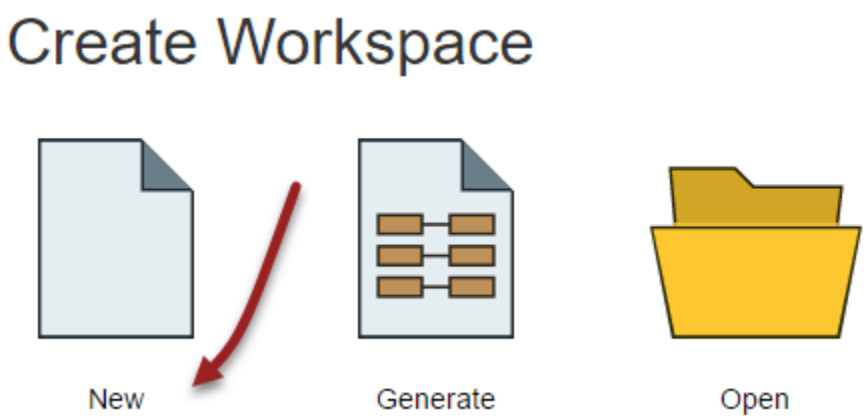
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

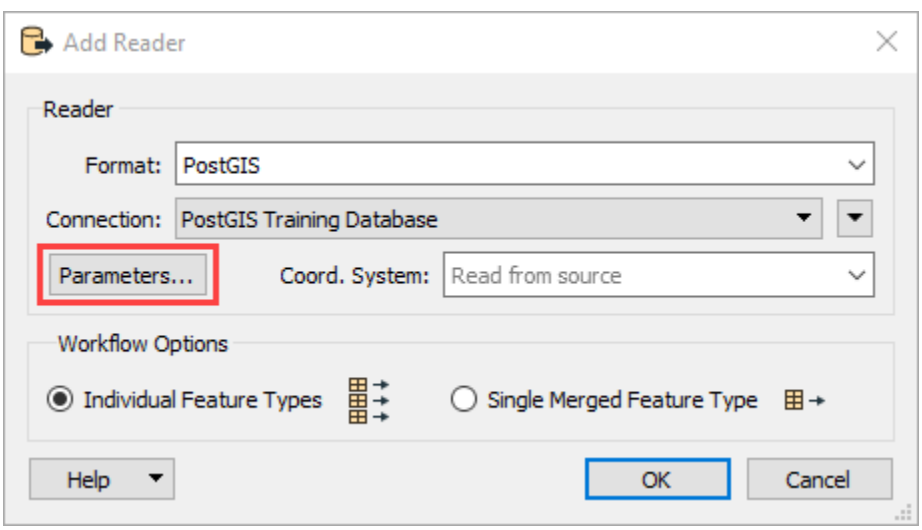
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Parks table from the Table List.



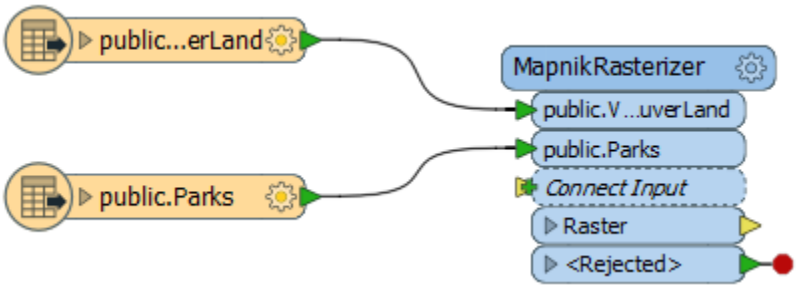
4. Add a PostGIS Raster Reader

- Similarly, add a PostGIS Raster
- Select the PostGIS Training Database from the Reader Connection list.
- Next, open the Reader Parameters and select the VancouverLand table from the Table List.

5. Add a MapnikRasterizer

The MapnikRasterizer allows you to convert point, line, and/ polygon features onto a new or existing raster by dividing features into groups called layers. For more information see the MapnikRasterizer Documentation.

- Select the VancouverLand and Parks Reader Feature Types, then type “MapnikRasterizer” to bring up the list of FME Transformers in the Quick Add Search. Then select the MapnikRasterizer from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas. This will automatically connect the VancouverLand and Parks readers to the MapnikRasterizer.



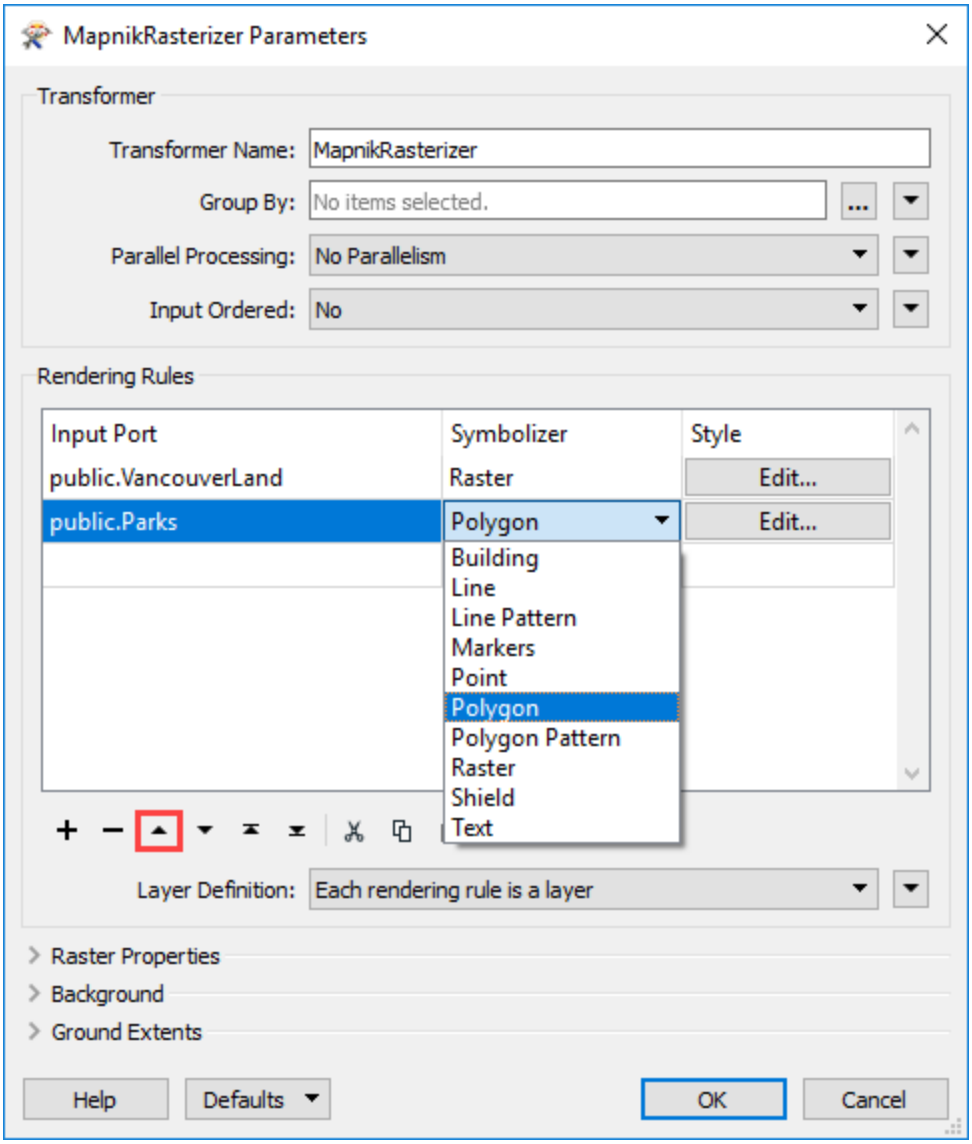
6. Set the MapnikRasterizer Parameters

- Once the MapnikRasterizer has been added, double-click the MapnikRasterizer or click on the red gear icon to open the transformer parameters dialog.

Notice how the two connections are listed in the Rendering Rules; however, they are set as Line symbols. Since you are classifying raster values by park polygons, you need to change the symbolizers for both of the Rendering Rules.

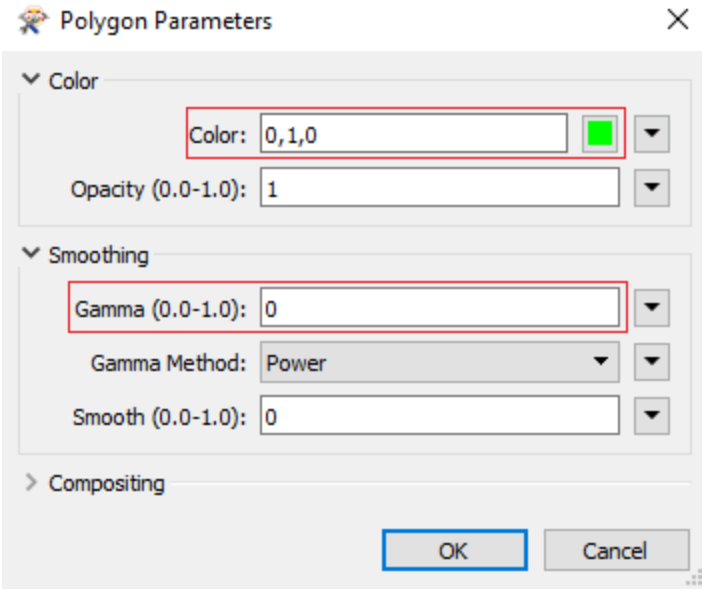
- Change the Symbolizer of the VancouverLand Rendering Rule to Raster and Parks to Polygon. Typically you want the Symbolizer to match the geometry of the features that are ported into the MapnikRasterizer.

Important: ensure the rendering rules are ordered as they are in the screenshot above otherwise the urban cells will be created on top of the park cells. The urban cells need to be classified first, followed by the park cells so they appear on the output raster. You can adjust the ordering of the rendering rules using the move up/down arrows below the Rendering Rules box.



7. Edit the Park Cell Style

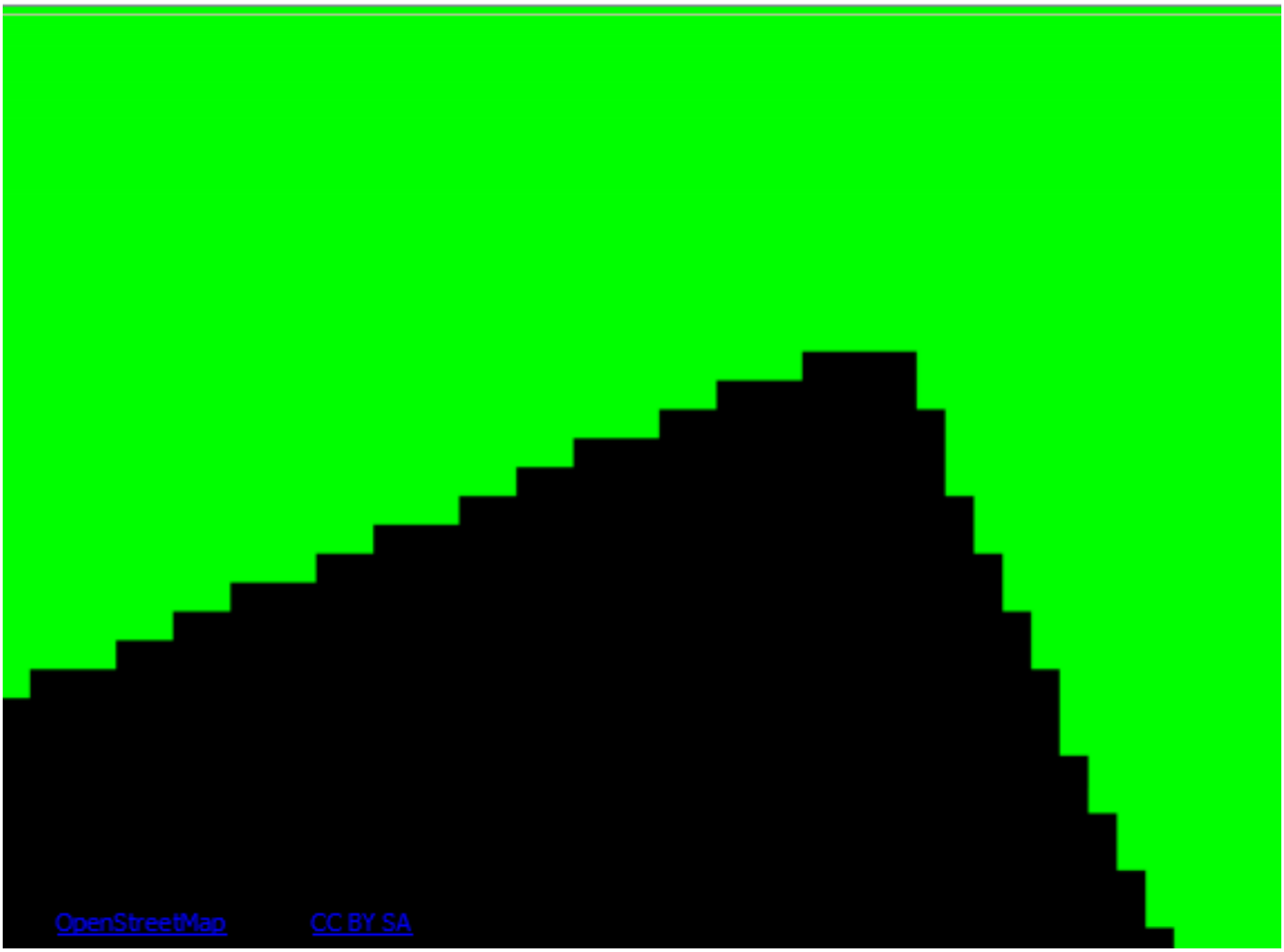
- Click Edit in the Style column of the Parks row to modify the polygon parameters. This is where you will set color, smoothing, and compositing for the raster output. In this example, you want the park cells to be green.
- Click on the color selector and choose Green from the list of basic colors. Alternatively, you can set colors by using RGB, HSV, or HTML values. In this example, the RGB values are Red: 0, Green: 255, Blue: 0. Click OK once you the color is set.
- Next, in the Polygon Parameters, expand the Smoothing section by clicking on the drop-down arrow.
- Set the smoothing Gamma to 0 then click OK to accept the parks Neighborhoods Polygon Parameters.



Smoothing is used to control whether the output will have to follow Boolean or Fuzzy logic. A Smoothing Gamma value of 1 uses Fuzzy logic which is useful for areas with movement or uncertainty such as a shoreline. Alternatively, a Smoothing Gamma value of 0 uses Boolean logic which is useful when you want to classify cells with 100% certainty. This exercise will use Boolean Logic so cells will be classified as either urban or park - this will prevent cells from being classified as 20% park and 80% urban.

Fuzzy Logic

Boolean Logic

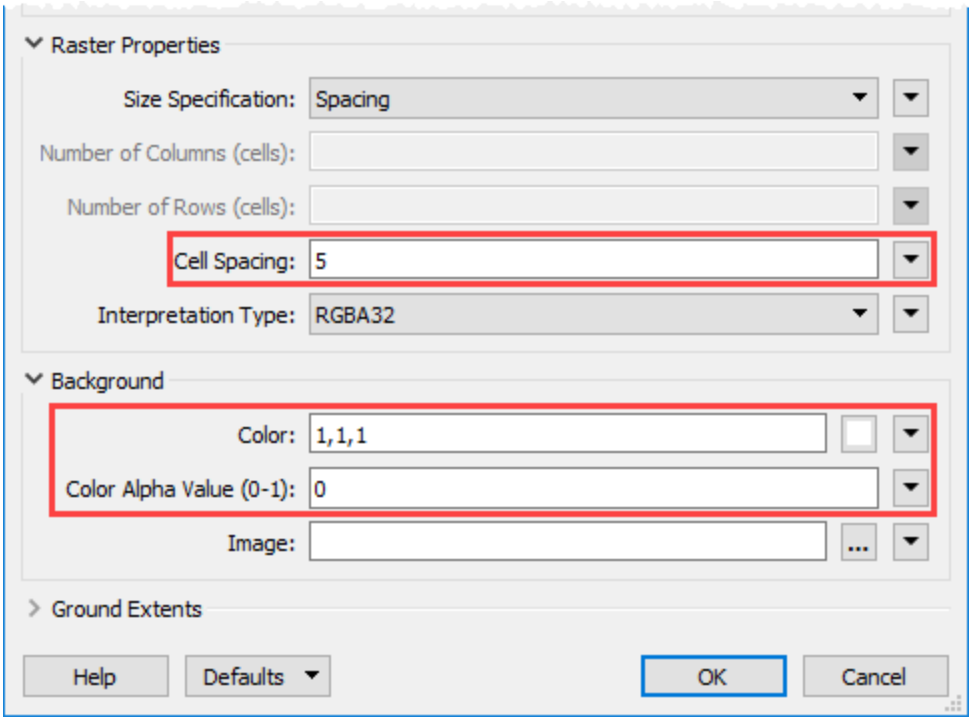


8. Set the Raster Properties

- Lastly, you have to define the cell size for the raster we will be producing. Set the Size Specification to Spacing from the drop-down list in the Raster Properties section.

The Cell Spacing parameter specifies the width of the output raster cells, measured in ground units. Note: smaller cells provide greater resolution, however, they typically take longer to load when viewing large extents (i.e. a cell spacing of 1 will take longer to load than 5) and also have large file sizes.

- Set the Cell Spacing to 1.
- Lastly, expand the Background section and set the Color Alpha Value to 0. This will effectively remove the background cells by making them transparent since you are only interested in creating raster cells for areas that are either urban or park in this exercise. For more information on adding alpha bands to remove cells, see the Setting NoData and Adding Alpha Bands to Remove Black Borders article.



9. Add an Inspector

- Next, add an Inspector and connect it to the MapnikRasterizer:Raster output port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

10. Add a GeoTIFF Writer

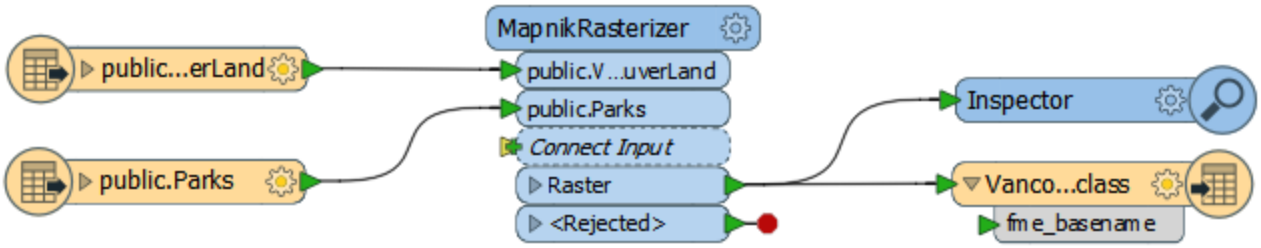
- Similar to adding a reader, type “GeoTIFF” then select the GeoTIFF format from the list of Writers.
- Click on the Dataset ellipsis and navigate to a folder where you want to save the GeoTIFF file such as C:\Users\Documents\MyFMEWorkspaces\Export.
- Set the Raster File Definition to Automatic to ensure the attributes are transferred from the database to the GeoTIFF. Click OK to open the writer parameters dialog.

An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected, for more information see the About Writer Feature Types: User Attributes documentation.

- In the Writer parameters dialog, set the Raster File Name to VancouverReclass, then click OK to add the reader to the canvas.
- Lastly, connect the GeoTIFF writer to the MapnikRasterizer:Raster port.

11. Run the Workspace

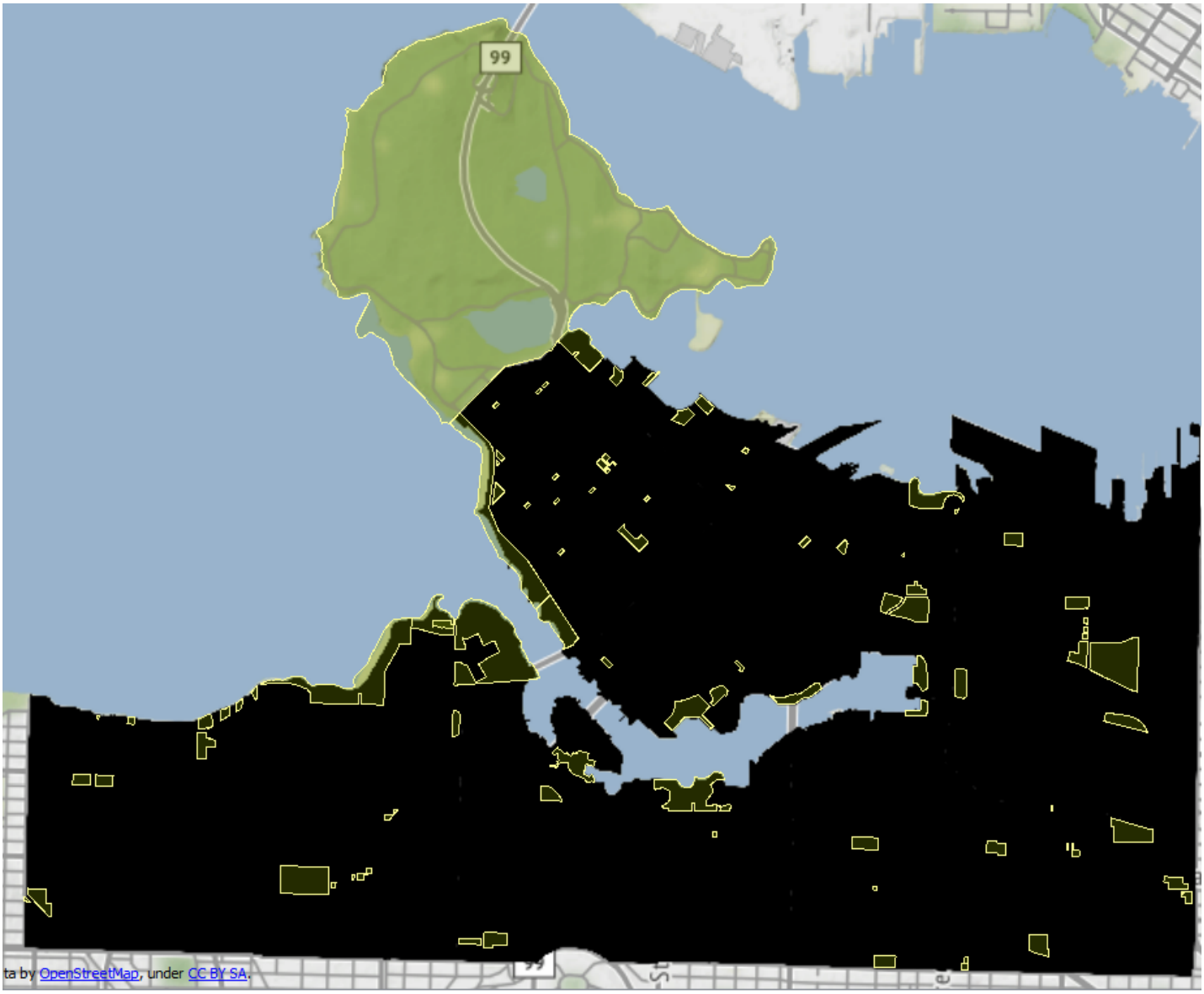
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



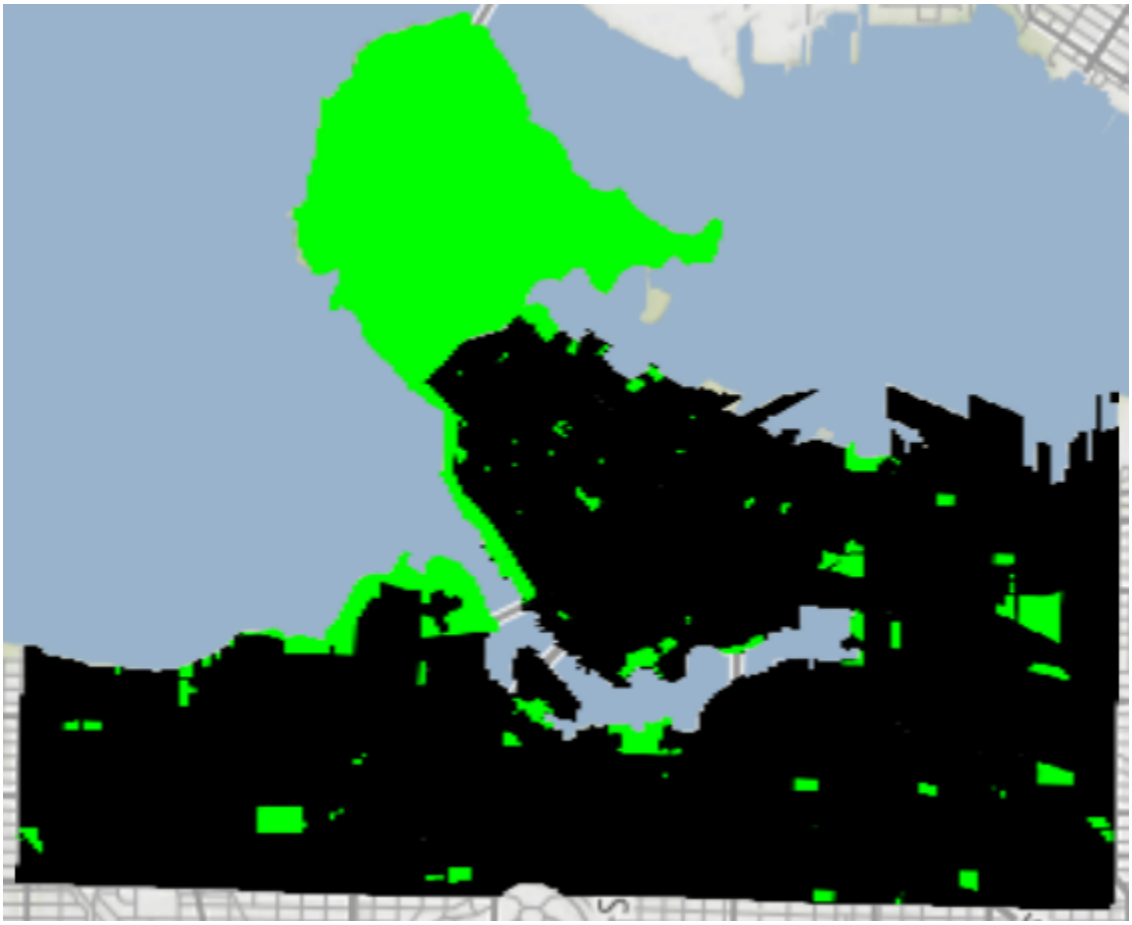
After running the translation, the output will be displayed in the FME Data Inspector. Notice the park cells have been classified using the Raster Bands (Band 0-3) and urban areas are black.

Results

Input



Output



Transformers

- Inspector- Sends features to the FME Data Inspector for display.
- MapnikRasterizer- Draws input point, line, polygon, and raster features onto a raster using the Mapnik toolkit.

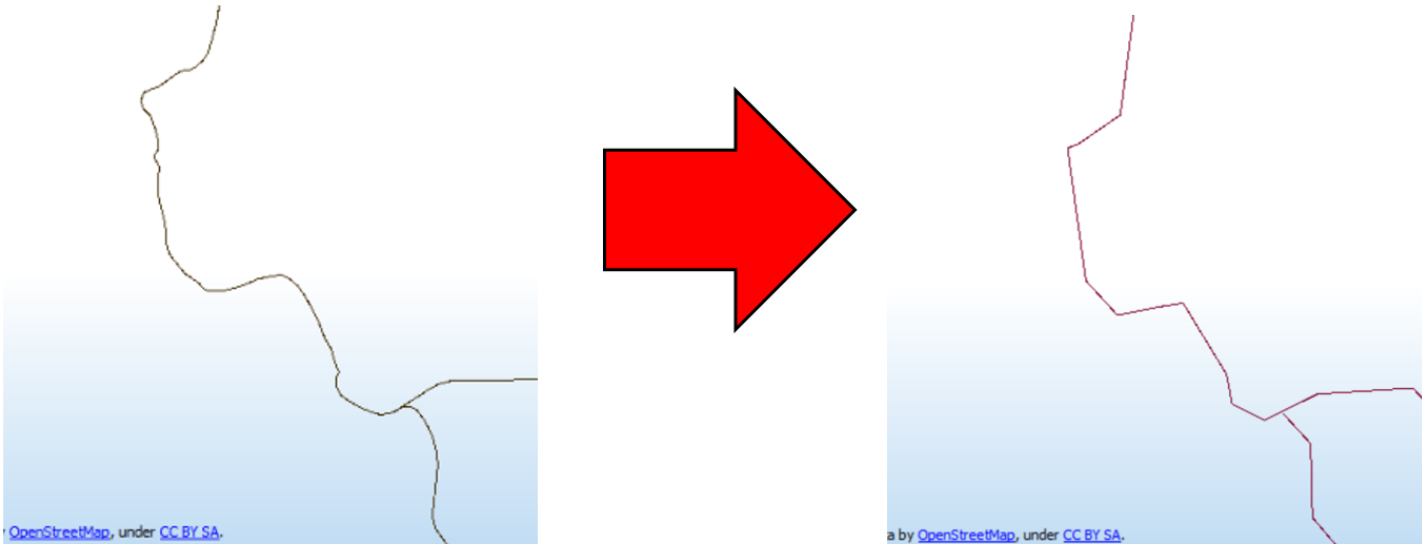
Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Reducing Unnecessary Coordinates

Overview

In this exercise, you will learn how to simplify linear features by removing unnecessary coordinates using the GeneralizerTransformer in FME - this operation is commonly referred to as weeding.



Downloads

reducing-unnecessary-coordinates.fmwt

bike-paths.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you want to simplify a bike path network to show the general path rather than every single turn. We will use the Douglas (Generalize) algorithm to reduce the density of coordinates by removing vertices, thus simplifying the line.

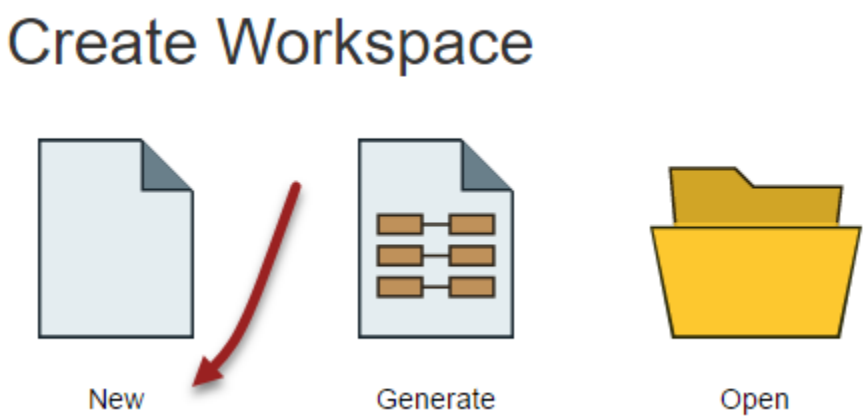
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

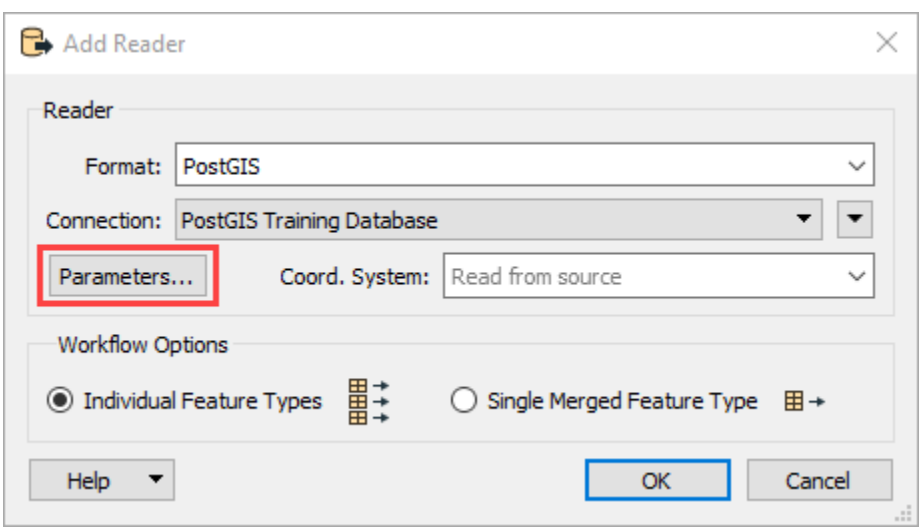
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing "PostGIS" without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters.
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the BikePaths table from the Table List.



4. Add a Generalizer

The Generalizer transforms or measures geometry features based on a specified algorithm. There are 4 types of algorithms: Generalizing, Smoothing, Measuring, and Fitting. In this exercise, you will use a Generalizing

algorithm because you want to reduce the density of coordinates by removing vertices.

- Add a Generalizer to the canvas by typing “Generalizer” to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the BikePaths Reader to the Generalizer.

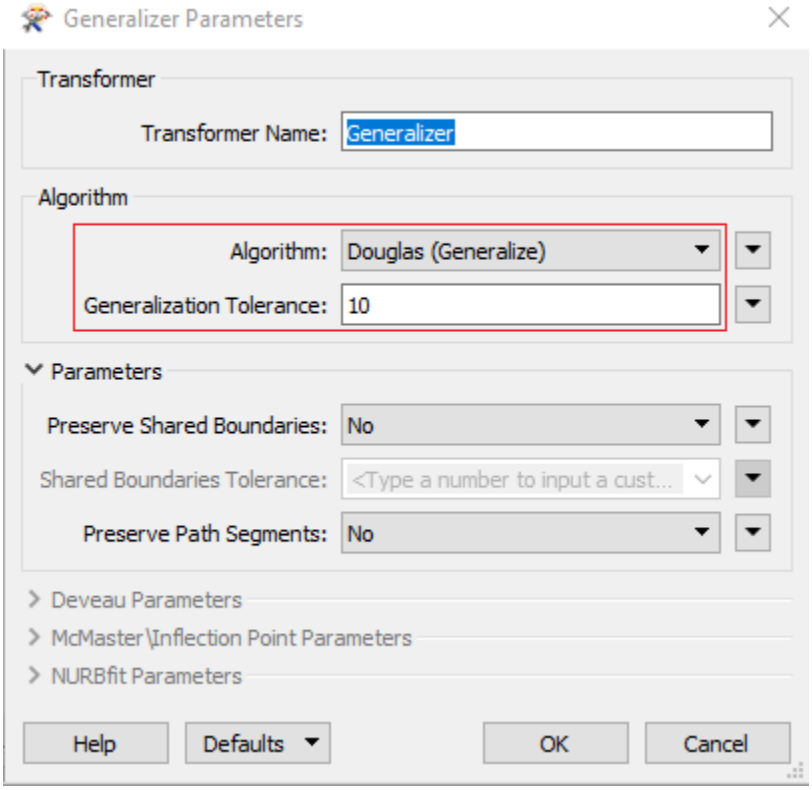
5. Specify a Generalization Tolerance

Next, specify the Generalization Tolerance. The Generalization Tolerance is a parameter that is used by the generalizing algorithm that is measured in the same units as the feature coordinates. For more technical details, see the Generalizer Documentation.

- Once the Generalizer has been added, double-click the Generalizer or click on the gear icon to open the transformer parameters dialog.

You will be using the default algorithm, Douglas (Generalize) because it will remove vertices which causes a deviation of less than the Generalization Tolerance, but the location of remaining vertices are not altered. This algorithm is good at reducing the number of points in a line; however, it is not very good at preserving the shape or the spatial relationship of the line relative to other entities.

- Set the value of the Generalization Tolerance to 10 to simplify the BikePaths line features to 10 map units (meters in this case). Then click OK to accept the Generalizer Parameters.



6. Add an Inspector

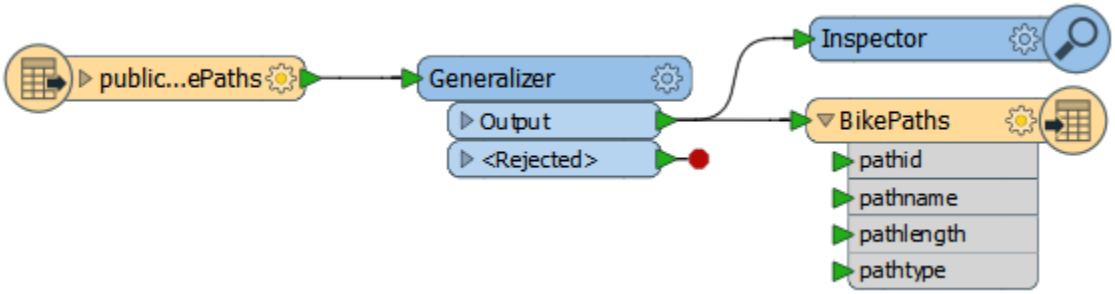
- Next, add an Inspector and connect it to the Generalizer:Output port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

7. Add a Google KML Writer

- Similar to adding a reader, type “KML” then select the Google KML format from the list of Writers.
- Specify a file name and set the destination folder for the KML file by clicking on the Dataset ellipsis. After specifying a folder location and filename, set the Table Definition to Automatic and click OK.
- Once the writer is added to the canvas, connect it to the Generalizer:Output port.

8. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the final transformer, the output dataset will automatically be opened in the FME Data Inspector.

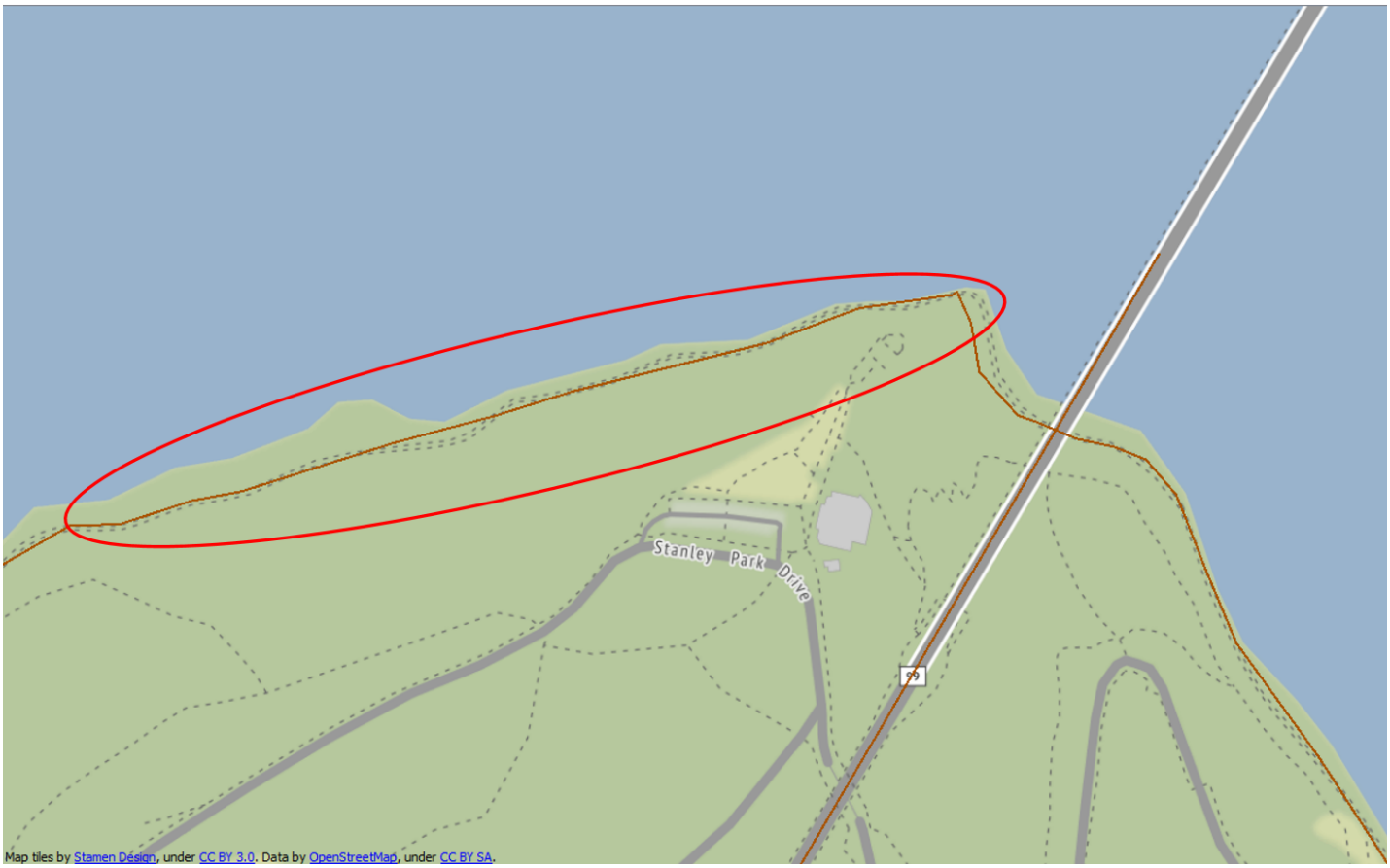


Results

Input


Stanley Park Drive
99

Output


Stanley Park Drive
99
Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under CC BY SA.

Transformers

- Generalizer - Transforms or measures geometry features based on a specified algorithm.
- Inspector- Sends features to the FME Data Inspector for display.

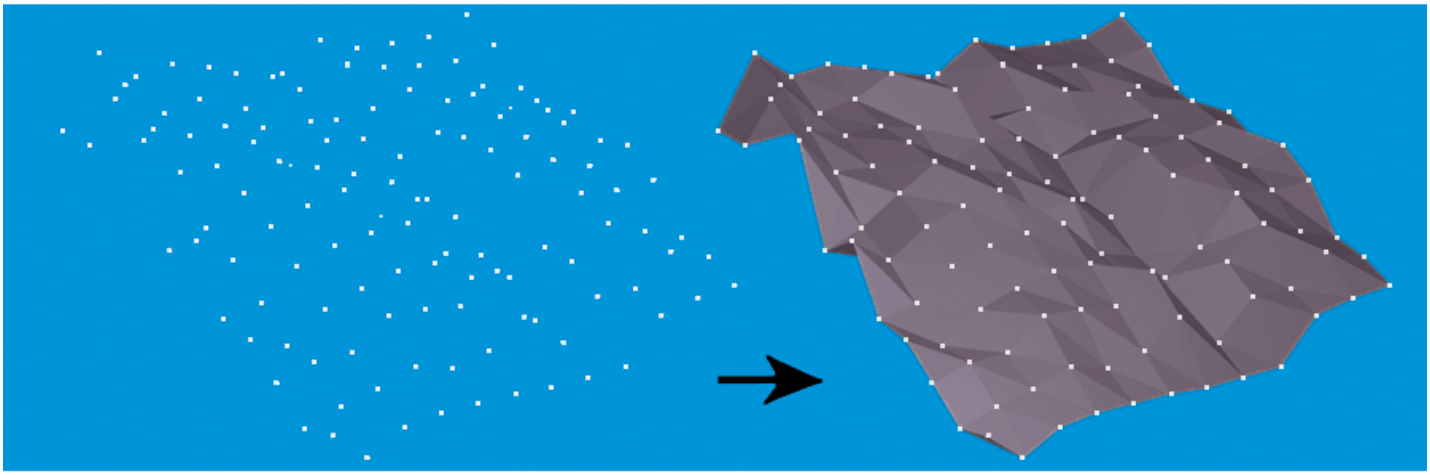
Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

TIN from Point Data

Overview

In this tutorial, you will learn how to create a Triangulated Irregular Network (TIN) from a 3D point dataset using the TINGenerator in FME. The TINGenerator in FME is able to create TIN's from Points/Lines or Breaklines whether the features are 2D or 3D.



Downloads

generate-tin.fmwt

elevationpoints.zip

Exercise

In this scenario, you want to create a TIN because you are interested in draping 2D features on a 3D TIN. This tutorial will only cover the TIN creation but if you are interested in learning how to drape imagery textures on terrain surfaces, read the Draping Imagery Textures on Terrain Surfaces article.

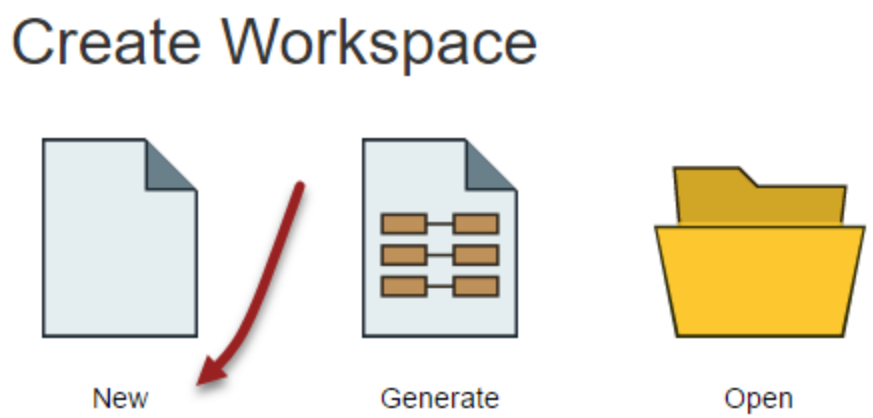
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

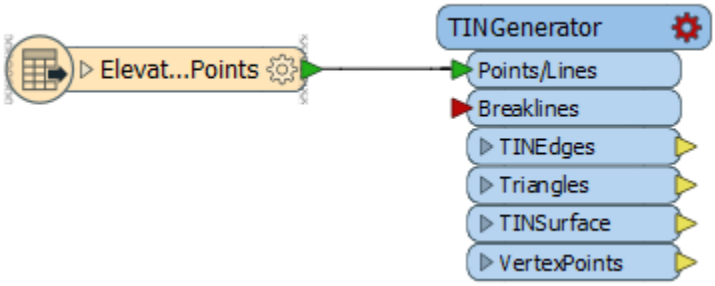


3. Add an ASPRS Lidar Data Exchange Format (LAS) Reader

- Start typing “LAS” without anything selected on the canvas, then select the ASPRS Lidar Data Exchange Format (LAS) format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis then navigate to the sample data folder and select the ElevationPoints LAS file (for example, C:\Users\Documents\FME\ElevationModel\ElevationPoints.las).

4. Add a TINGenerator

- Add a TINGenerator to the canvas by typing “TINGenerator” to bring up the list of FME Transformers in the Quick Add Search. Select the TINGenerator from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the ElevationPoints Reader to the TINGenerator - it should be connected to TINGenerator:Points/Lines port.

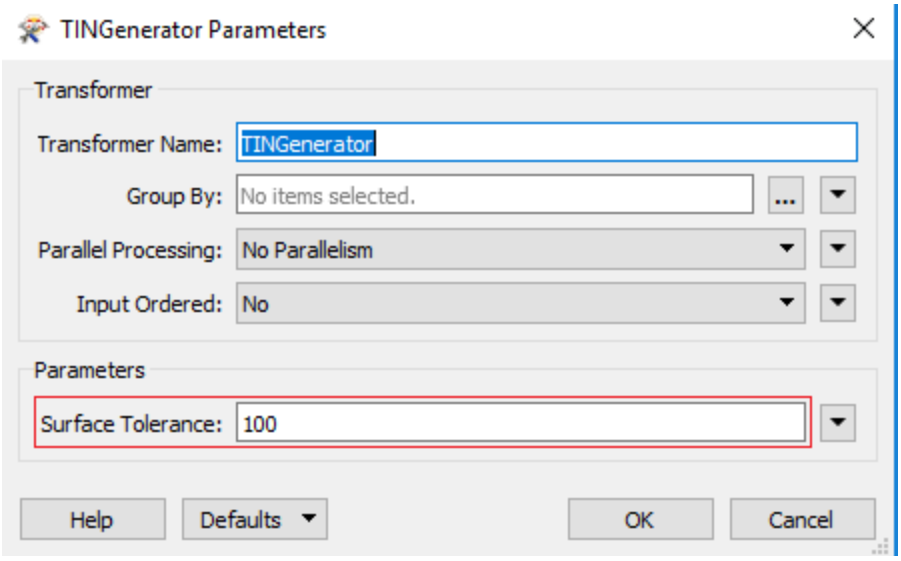


5. Set the TIN Surface Tolerance

- Once the TINGenerator has been added, double-click the TINGenerator or click on the gear icon to open the transformer parameters dialog.

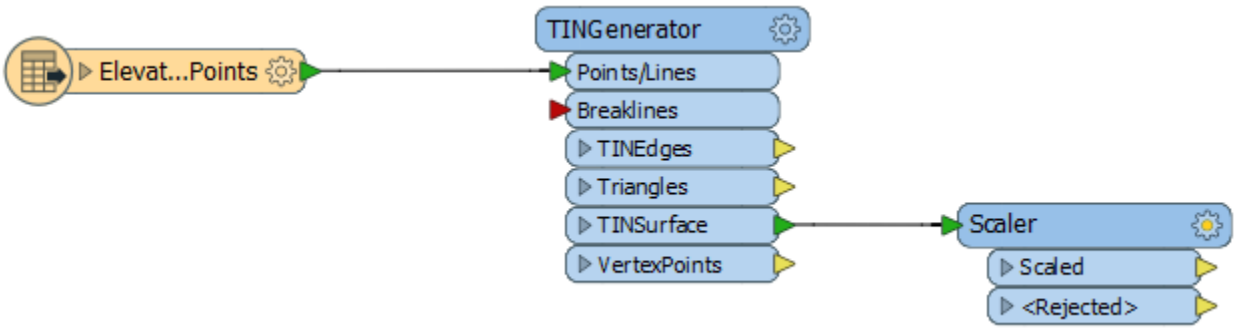
The ElevationPoints dataset you are working with contains approximately 1.4 million elevation points. Without filtering out points, the TIN would be very jagged and require more processing power. In order to reduce the number of points used to create the TIN, you need to specify a surface tolerance. A larger surface tolerance value will speed up the surface model construction and simplify the TIN - the larger the value, the more input points will be filtered out. To learn more about Surface Tolerance values, see the [TINGenerator Documentation].

- Set the surface tolerance value to 100, then click OK to accept the TINGenerator Parameters.



6. Scale the Z coordinate

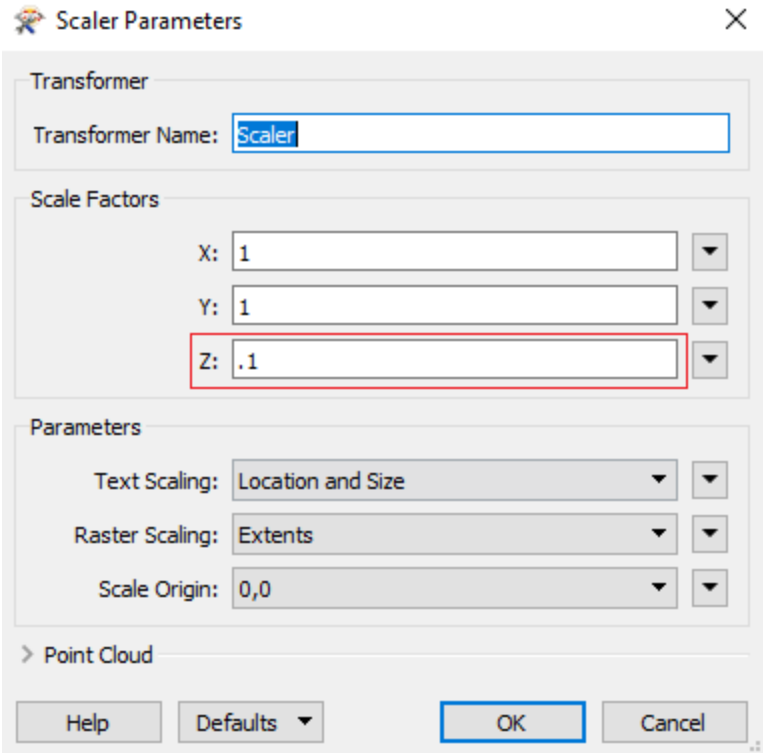
- Add a Scaler Transformer to the canvas and connect the TINGenerator:TINSurface port to the Scaler transformer. Your workspace should look like the screenshot below:



7. Set the Z Scale Factor

The only parameter that needs to be changed is the Z value in the Scale Factor section. You need to change the Z scale otherwise the 3D features in the TIN will be too tall and look unrealistic.

- Open the Scaler Parameters.
- Set the value of the Z Scale Factor to 0.1 to control how high features will be extruded from the ground in the output.



8. Add an Inspector

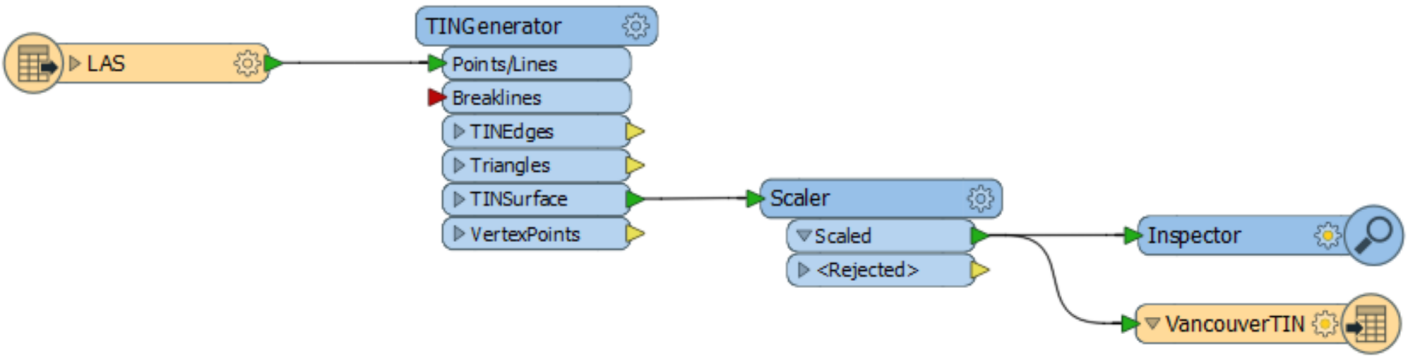
- Next, add an Inspector after the Scaler and connect it to the Scaler:Scaled port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

9. Add a Shapefile Writer

- Similar to adding a reader, type “Shapefile” then select the Shapefile format from the list of Writers.
- Specify a file name and set the destination folder for the Shapefile by clicking on the Dataset ellipsis. After specifying a folder location and filename, set the Table Definition to Automatic and click OK.
- Set the Shapefile Name in the Writer Parameters dialog that appears, then click OK to add the Shapefile Writer to the Canvas.
- Once the Shapefile Writer has been added, connect it to the Scaler:Scaled port.

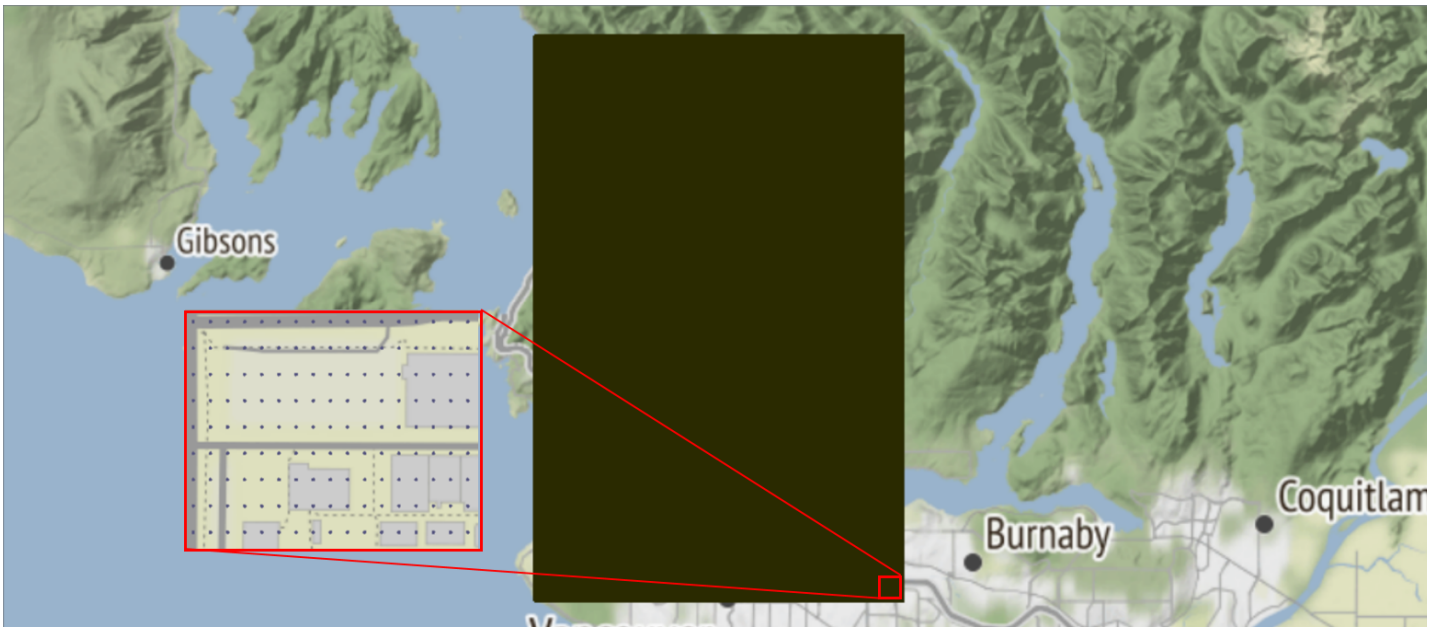
10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the final transformer, the output dataset will automatically be opened in the FME Data Inspector.

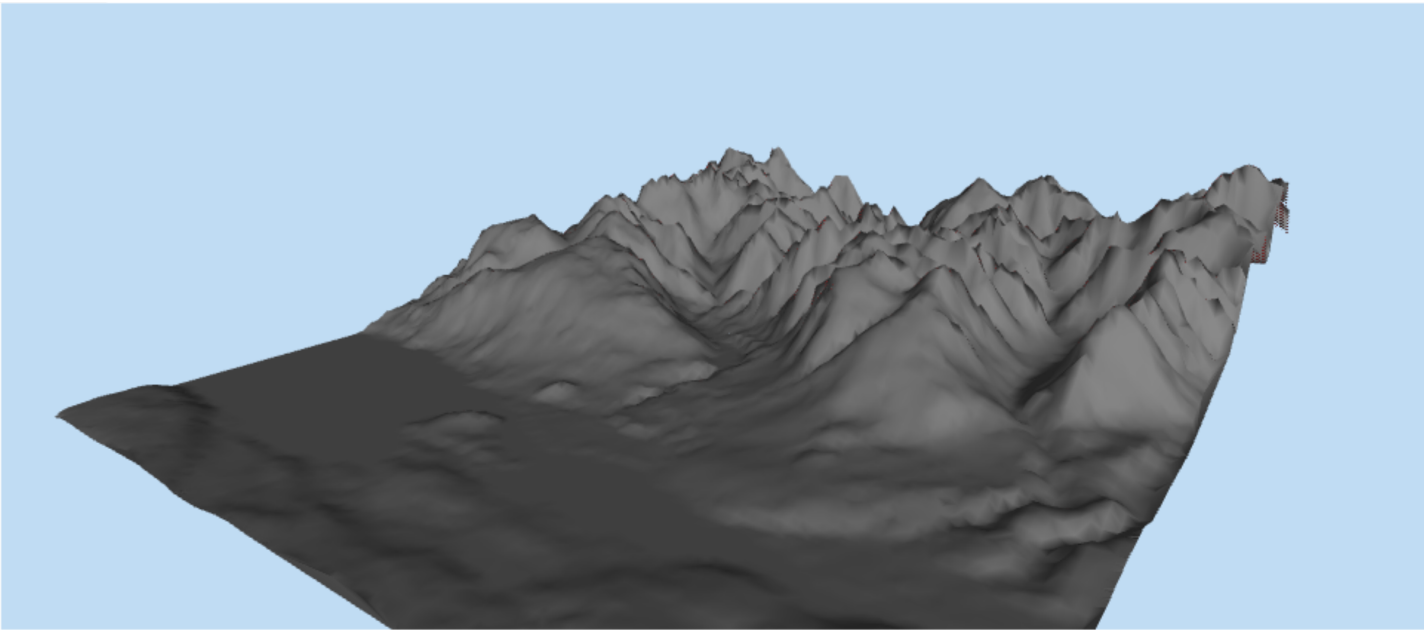


Results

Input



Output



Transformers

- Inspector - Sends features to the FME Data Inspector for display.
- Scaler- The Scaler scales objects to make them bigger or smaller.
- TINGenerator - Constructs a Delaunay triangulation based on input points and breaklines.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

RCaller: Interpolate Points to Raster Through Kriging

Intro: Getting Started with the RCaller | **Previous: RCaller: Is Tree Height and Tree Width Correlated?**

Introduction

The RCaller transformer allows for much more statistical analysis than the StatisticsCalculator transformer. This example shows how the RCaller can be used to create a raster from points. Kriging, an interpolation method, is used to create an estimated surface from cell signal point data.

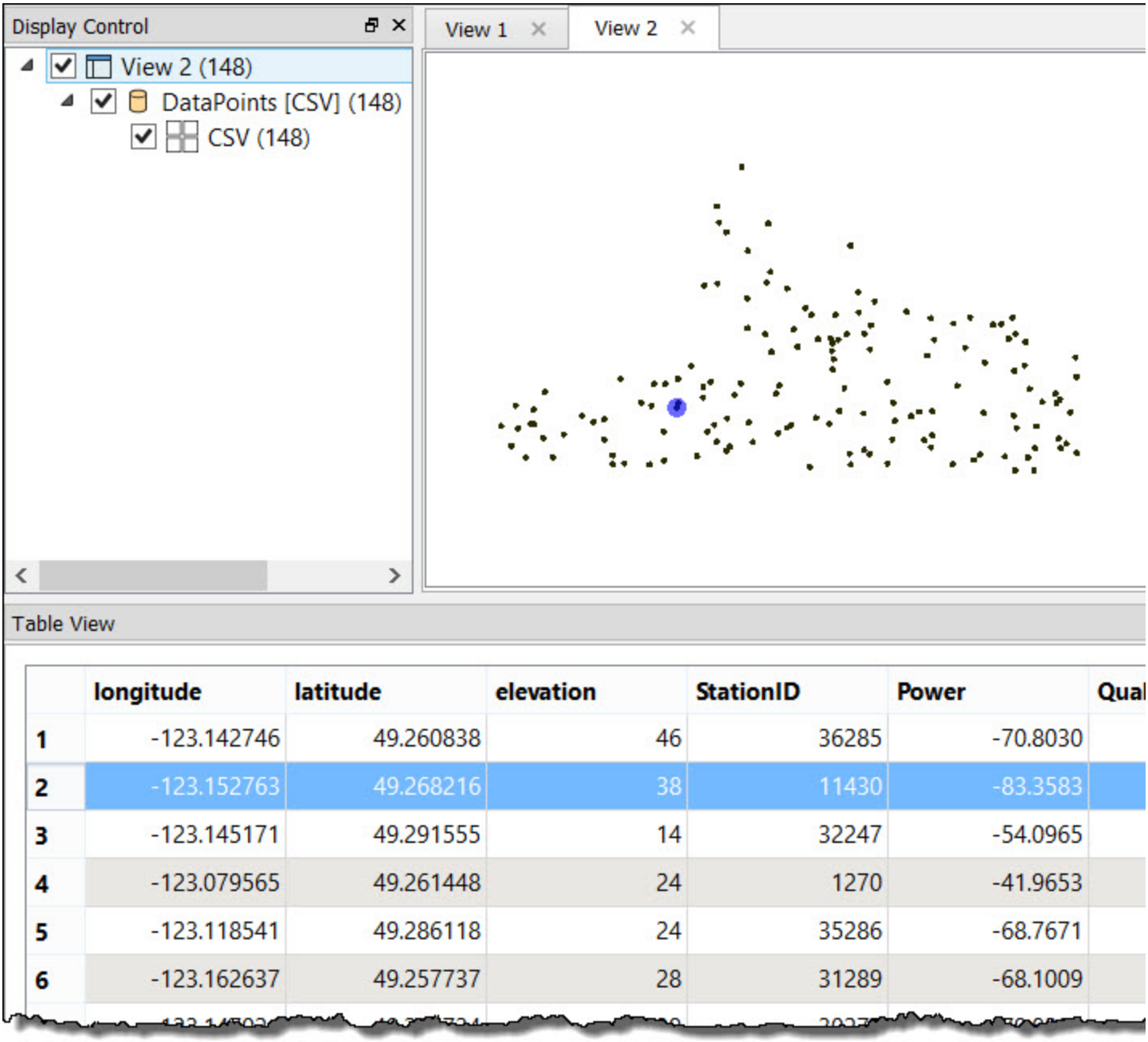
Downloads

datapoints.csv

kriging-fme20161.fmwt

Source Data

Cell signal point data in CSV format.



	longitude	latitude	elevation	StationID	Power	Qual
1	-123.142746	49.260838	46	36285	-70.8030	
2	-123.152763	49.268216	38	11430	-83.3583	
3	-123.145171	49.291555	14	32247	-54.0965	
4	-123.079565	49.261448	24	1270	-41.9653	
5	-123.118541	49.286118	24	35286	-68.7671	
6	-123.162637	49.257737	28	31289	-68.1009	

Source CSV data viewed with the Data Inspector

Instructions

The download *kriging_FME2016.1.fmwt* is the complete workspace. If you would like to create the workspace yourself, please download *DataPoints.csv*, and follow the steps below to create it.

Please note that the workspace requires users to install the 'geoR', and 'sp' R modules.

1. Read Source Data

Read the DataPoints.csv using a CSV Reader.

2. Reproject

Reproject the data from lat/long (degree) to UTM83-10 (meter) using the Reprojector

3. Extract coordinate values

The CoordinateExtractor extracts the x and y coordinate values of the points and adds them as attributes named easting (x) and northing (y). Easting and northing will be used to create a map and for kriging in the RCaller.

4. Create a temporary file for the raster output

One feature is used to generate a temporary file with the TempPathnameCreator. The raster image from the RCaller will be temporarily saved to this location.

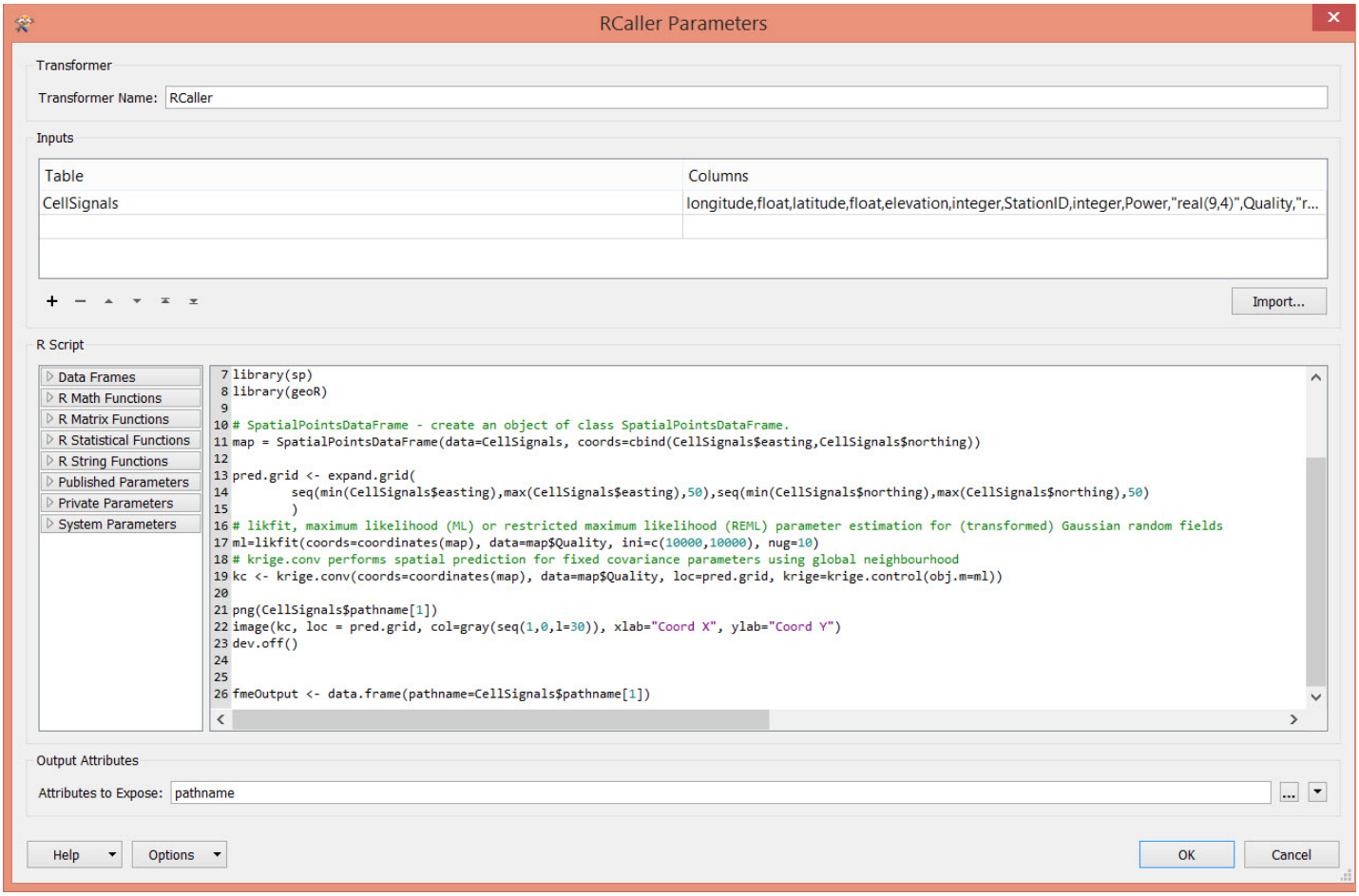
5. Add a RCaller

The R script interpolates points into a raster and creates a raster png at the temporary file location. For the R script, please specify the following:

```
library(sp)
library(geoR)

map = SpatialPointsDataFrame(data=CellSignals, coords=cbind(CellSignals$easting,CellSignals$
pred.grid <- expand.grid(
seq(min(CellSignals$easting),max(CellSignals$easting),50),seq(min(CellSignals$northing),max(
)
ml=likfit(coords=coordinates(map), data=map$Quality, ini=c(10000,10000), nug=10)
kc <- krige.conv(coords=coordinates(map), data=map$Quality, loc=pred.grid, krige=krige.contr
png(CellSignals$pathname[1])
image(kc, loc = pred.grid, col=gray(seq(1,0,l=30)), xlab="Coord X", ylab="Coord Y")
dev.off()

fmeOutput <- data.frame(pathname=CellSignals$pathname[1])
```



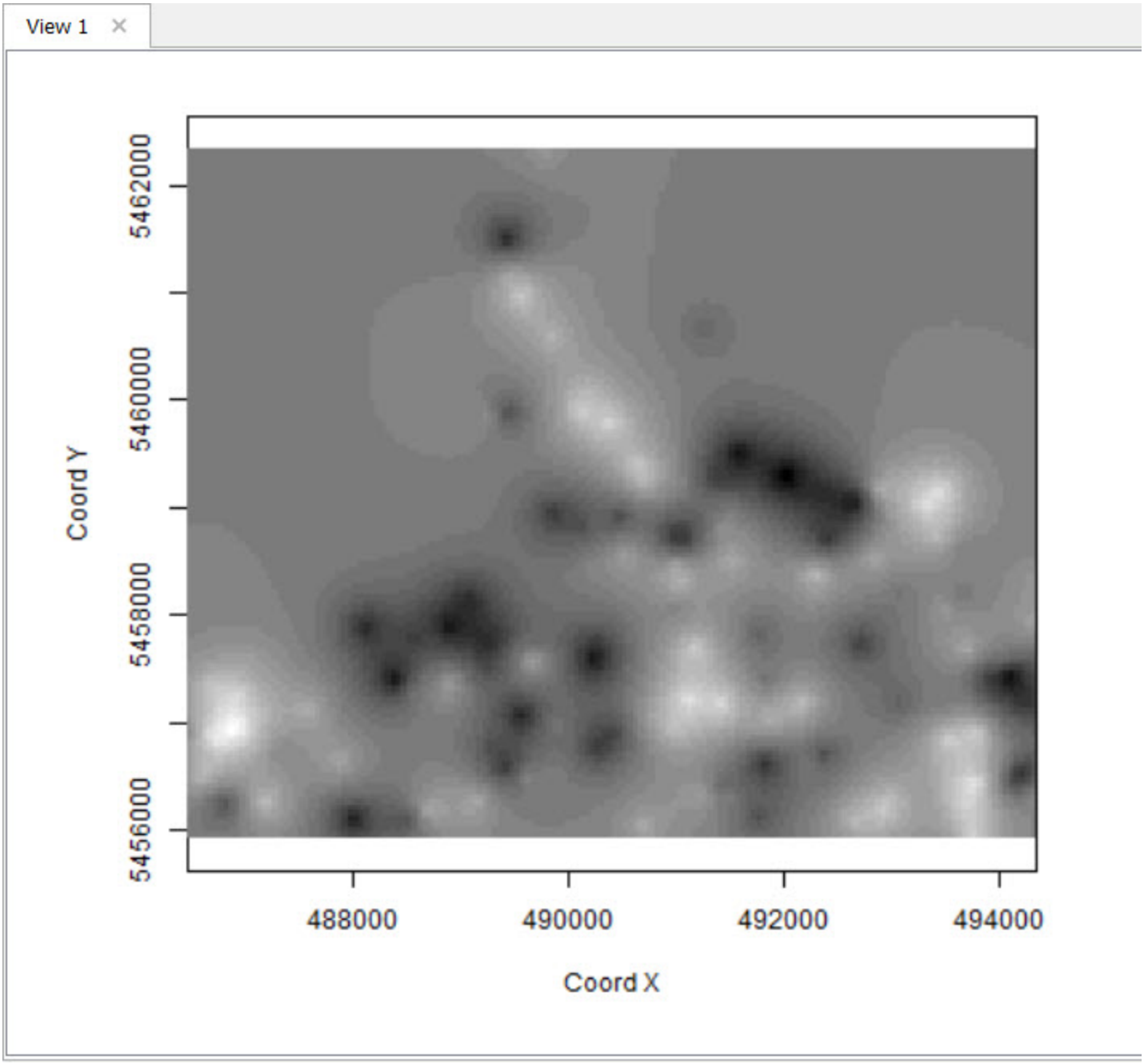
RCaller parameters

6. Read raster png

The FeatureReader reads the raster png created within the RCaller, which allows us to view the output data.

7. View the result

Inspect the output raster in Data Inspector. The result demonstrates that the RCaller can be used to generate a raster from point data. Note that this example may not be statistically accurate.

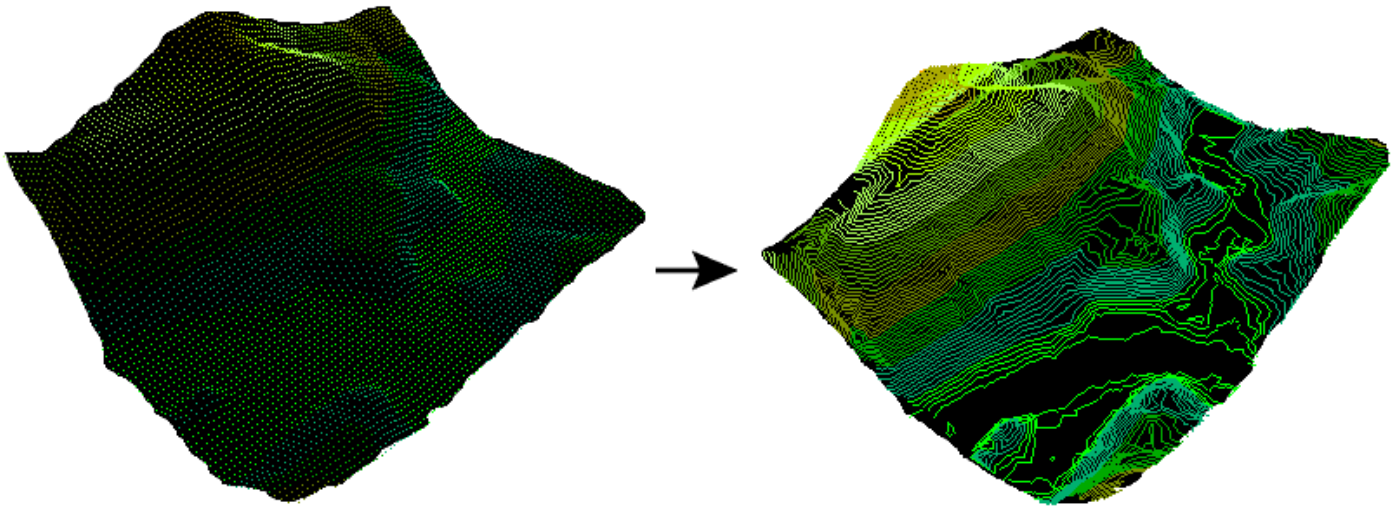


Kriging output displayed in Data Inspector

Generate Contour Data from Points

Overview

In this tutorial, you will learn how to create contour lines using the ContourGenerator and a point dataset. The ContourGenerator in FME constructs a Delaunay triangulation based on the input 3D point features which are then used to generate contour lines.



Downloads

generate-contours-from-points.fmwt

elevationpoints.zip

Exercise

In this scenario, you will create contour lines from an elevation point dataset because you want to create a 50m contour dataset for later use.

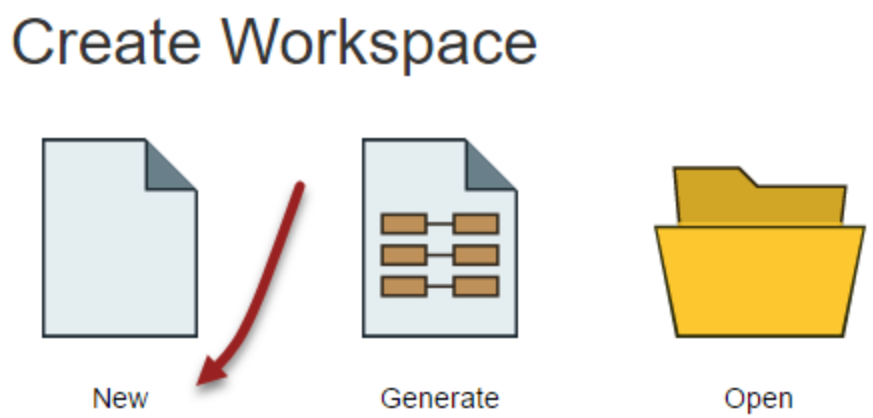
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



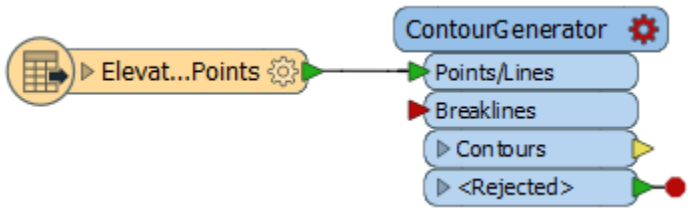
3. Add a Shapefile Reader

- Start typing “Shapefile” without anything selected on the workspace, then select the Shapefile format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the Schools shapefile (for example, C:\Users\Documents\FME\ElevationModel\ElevationPoints.shp).

4. Add a ContourGenerator

- Add a ContourGenerator to the canvas by typing “ContourGenerator” to bring up the list of FME Transformers in the Quick Add Search. Select the ContourGenerator from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.

The ElevationPoints Reader Feature Type should be connected to the ContourGenerator:Points/Lines port.



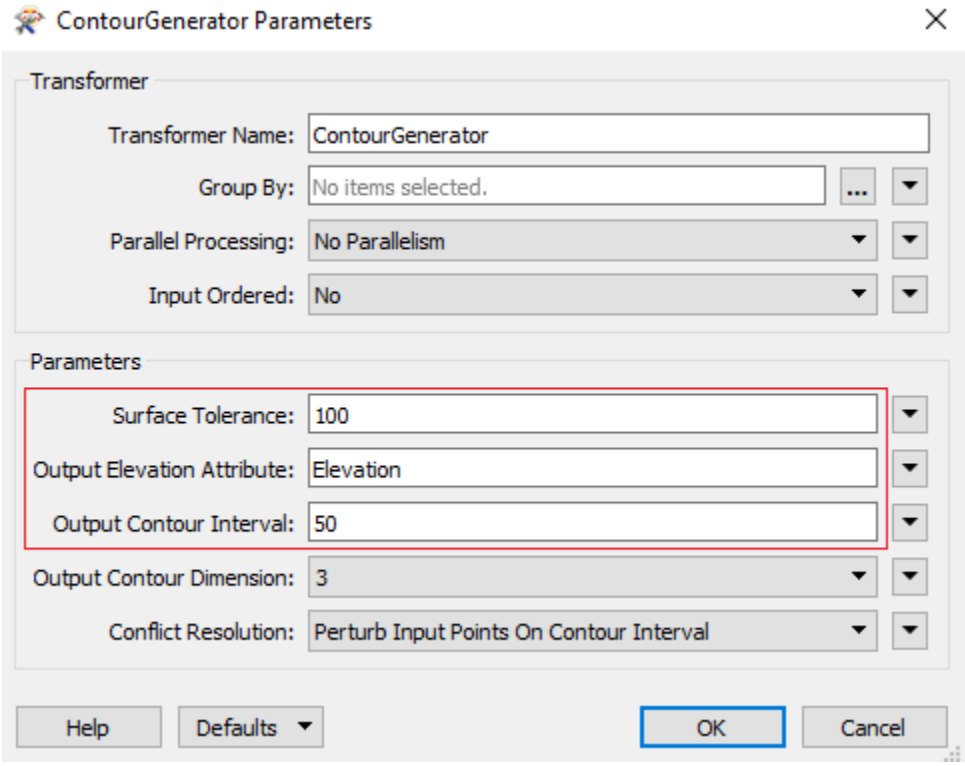
5. Set the Output Contour Interval

- Once the ContorGenerator has been added, open the transformer parameters by double-clicking it or clicking on the red gear icon.

The ElevationPoints dataset you are working with contains approximately 1.4 million elevation points. Without filtering out points, the TIN would be very jagged and require more processing power. In order to reduce the number of points used to create the TIN, you need to specify a surface tolerance. A larger surface tolerance value will speed up the surface model construction and simplify the TIN - the larger the value, the more input points will be filtered out. To learn more about Surface Tolerance values see the ContourGenerator Documentation.

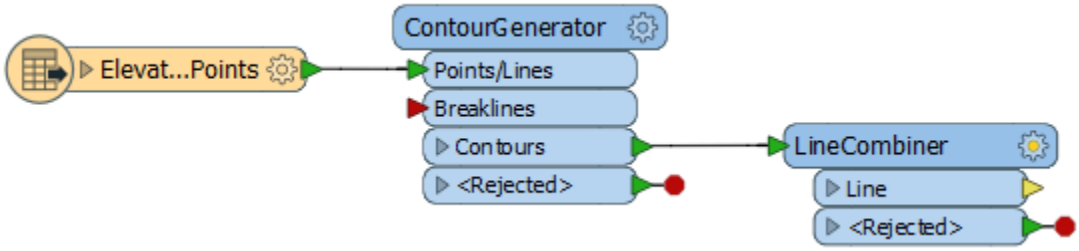
- Set the surface tolerance value to 100.
- Next, change the name of the Output Elevation Attribute from “_elevation” to “elevation”.
- Lastly, set the Output Contour Interval to 50 - this will be applied using the input feature units.
- Leave all other parameters as default, then click OK to accept the changes and close the parameters

dialog. The ContourGenerator Parameters dialog should look similar to the one in the screen capture provided below:



6. Combine Contour Line Segments

- Add a LineCombiner Transformer to the canvas and it to the connect the ContourGenerator:Contours port. So far, your workspace should look like the screenshot below:

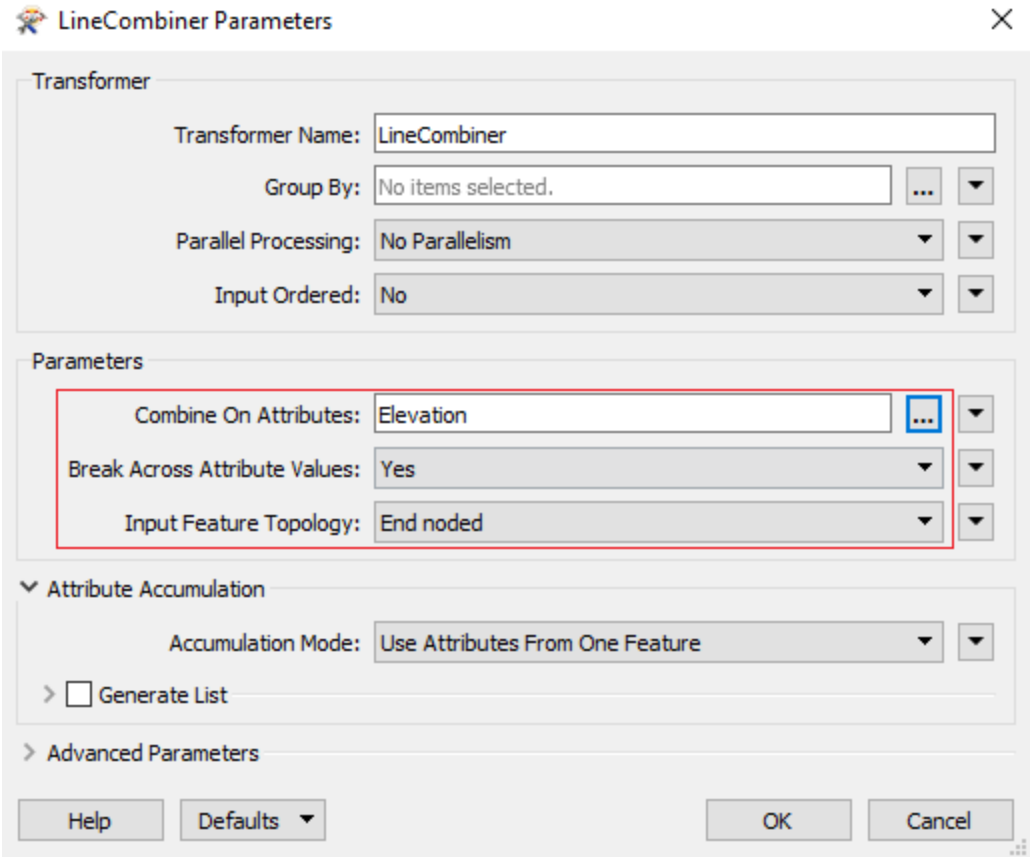


7. Set the Combine On Attribute

The purpose of this transformer is to combine paths to have one record per pathid rather than having segmented contour lines.

- Open the LineCombiner Parameters.
- The only parameter needs to be changed is the Combine On Attributes. Click on the Combine On Attributes ellipsis to reveal a list of attributes to combine on. Select the Elevation attribute from the list, then click OK to return to the LineCombiner Parameters.

Note: In order for lines to be combined, they must meet at the exact same start/end point. Additionally, lines will remain broken at points where three or more lines converge. For more information see the [LineCombiner Documentation](#).



- Leave all other parameters as default, then click OK to accept the changes to the LineCombiner Parameters.

8. Add an Inspector

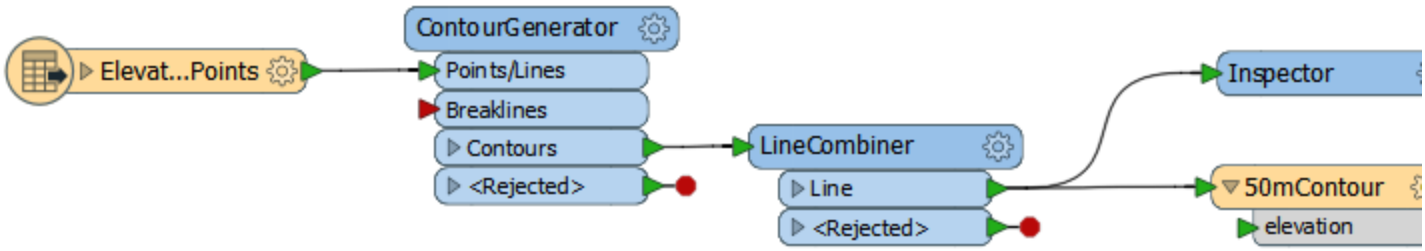
- Next, add an Inspector after the LineCombiner and connect it to the LineCombiner:Line port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

9. Add an Autodesk AutoCAD DWG/DXF Writer

- Similar to adding a reader, type “DWG” then select the Autodesk AutoCAD DWG/DXF format from the list of Writers.
- Click on the Dataset ellipsis to specify where to save your data and name the file: 50mContour (for example, C:\Users\Documents\FME\Contours\50mContour.dwg)
- Set the Layer Definition to Automatic to ensure all attributes that are created or renamed will be written to the dwg file.
- Similar to the Inspector, connect the 50mContour Writer to the LineCombiner:Line port.

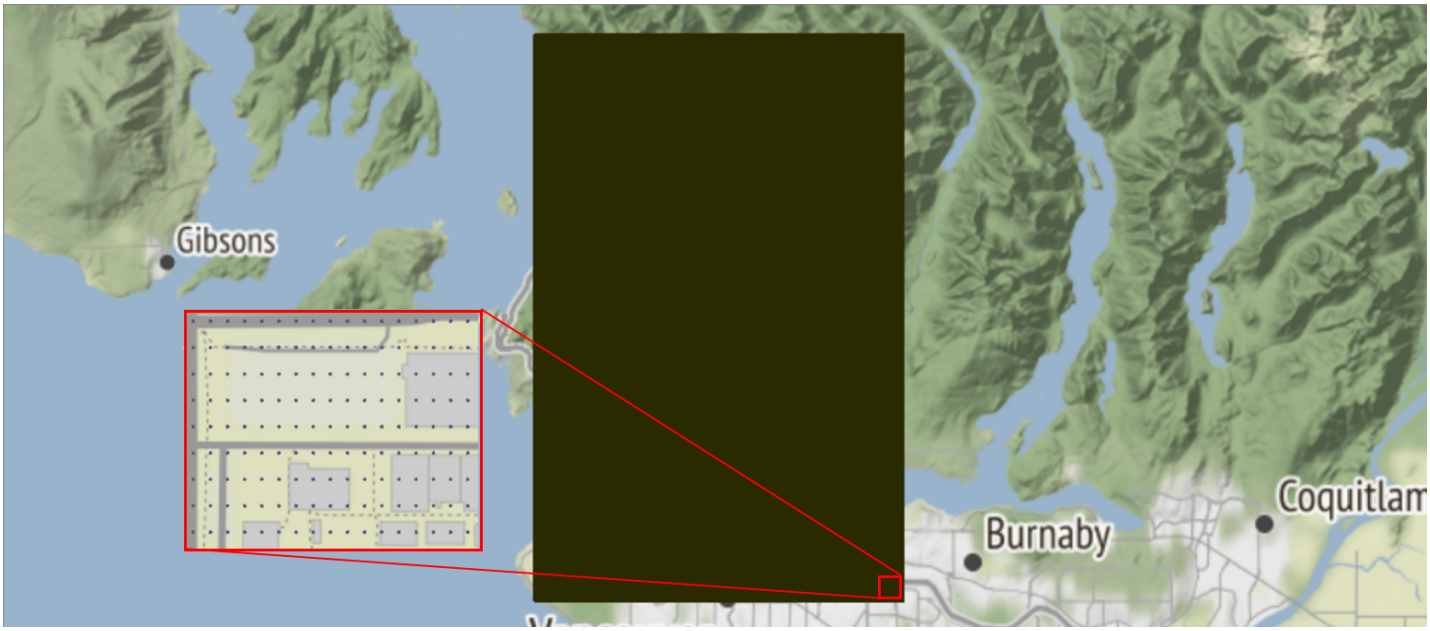
10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the final transformer, the output dataset will automatically be opened in the FME Data Inspector.

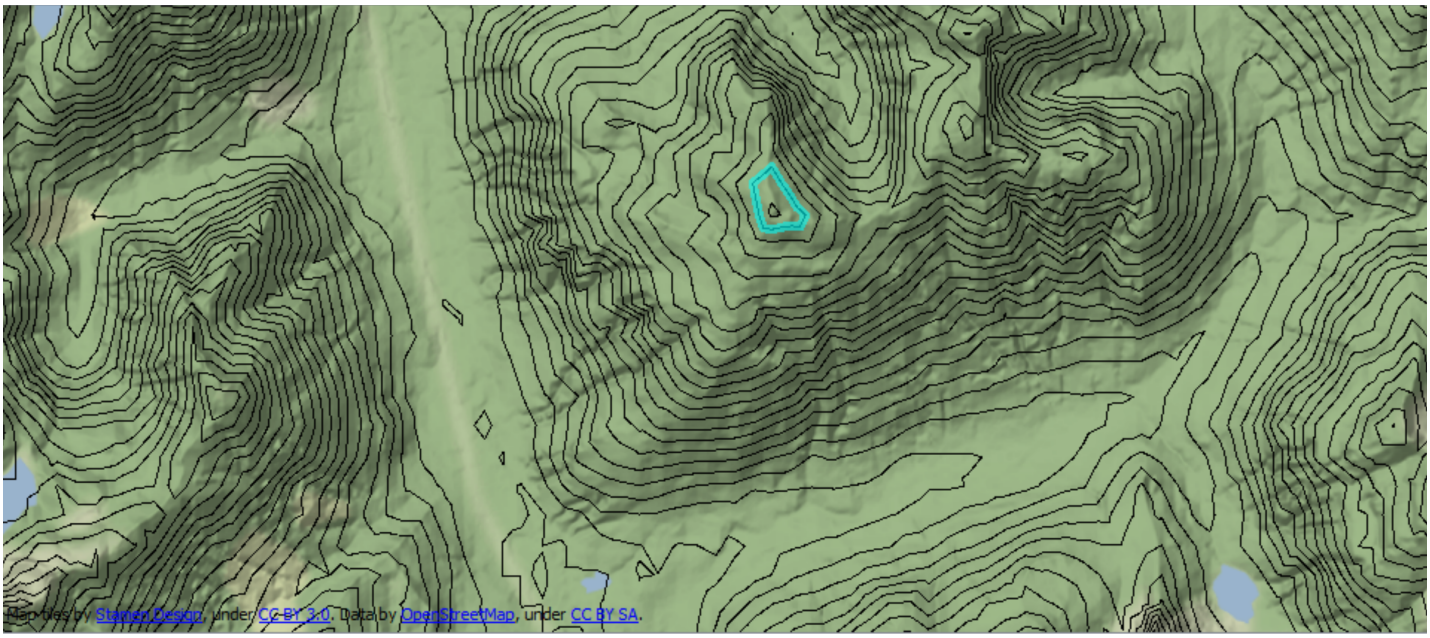


Results

Input



Output



Transformers

- ContourGenerator - Constructs a Delaunay triangulation based on input points and breaklines.
- Inspector - Sends features to the FME Data Inspector for display.
- LineCombiner - Takes lines and connects them to form longer lines.

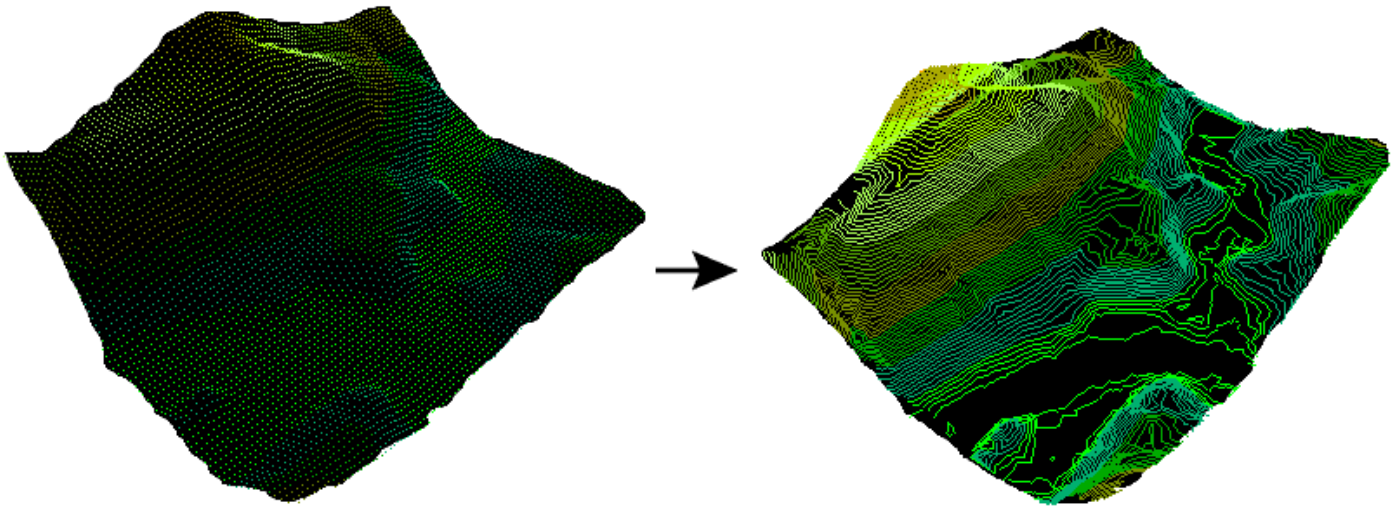
Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Generate Contour Data from Raster

Overview

In this tutorial, you will learn how to create contour lines using the ContourGenerator and a Digital Elevation Model (DEM). The ContourGenerator in FME constructs a Delaunay triangulation based on the input value of the raster cells which are then used to generate contour lines.



Downloads

generate-contours-from-raster.fmwt

dem-full.zip

Exercise

In this scenario, you will create contour lines from DEM because you want to create a 50m contour dataset for later use.

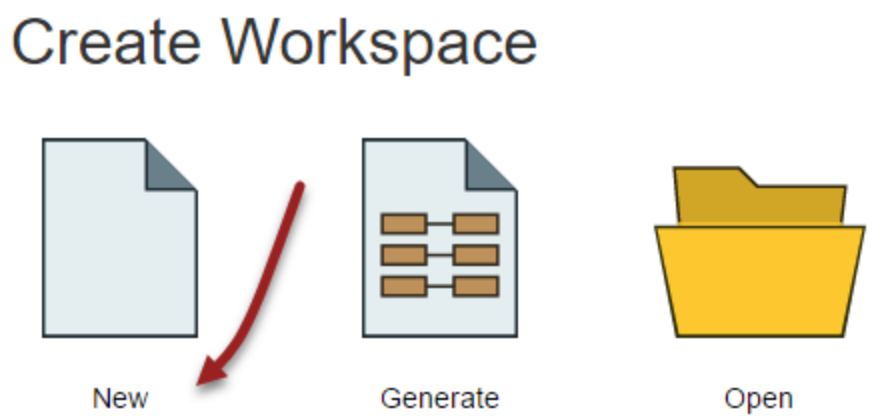
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

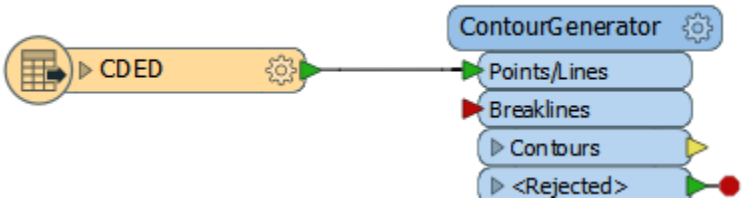


3. Add a Canadian Digital Elevation Data (CDED) Reader

- Start typing “CDED” without anything selected on the canvas, then select the CDED format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the DEM-Full.dem file (for example, C:\Users\Documents\FME\ElevationModel\DEM-FULL.dem).

4. Add a ContourGenerator

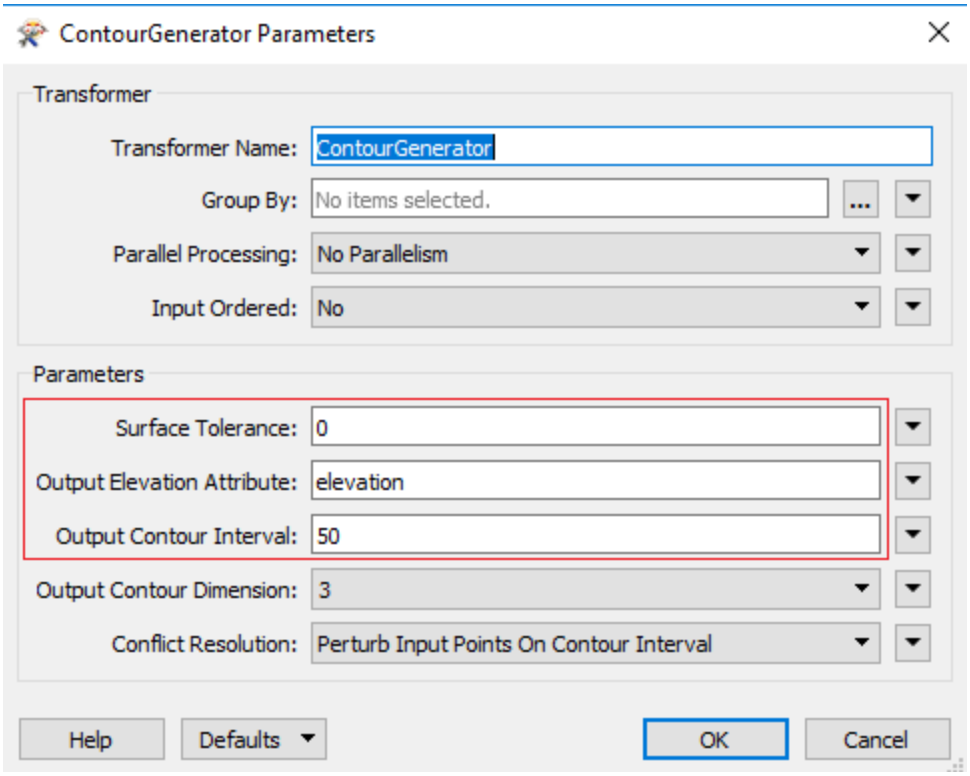
- Select the CDED reader and type “ContourGenerator”, then select the ContourGenerator from the list of Transformers in the Quick Add Search. This will automatically make the connection from the Reprojector to the ContourGenerator.



5. Set the Output Contour Interval

- Open the ContourGenerator transformer parameters.
- Set the value of the Surface Tolerance parameter to 0. Surface Tolerance is a parameter used to filter vertices; however, since you are working with a raster dataset you can turn this feature off by specifying a value of 0.
- Next, change the name of the Output Elevation Attribute from “_elevation” to “elevation”.
- Lastly, set the Output Contour Interval to 50 - this will be applied using the input feature units.

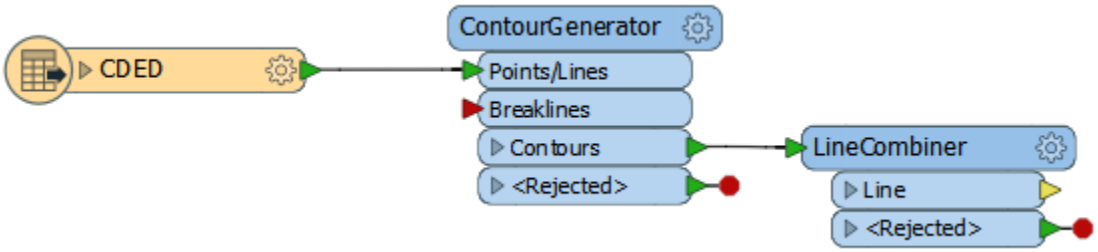
Leave all other parameters as default, then click OK to accept the changes to the ContourGenerator Parameters. The ContourGenerator Parameters dialog should look similar to the one in the screen capture provided below:



Note: the default dimension of the output of the ContourGenerator is 3D.

6. Combine Contour Line Segments

- Add a LineCombiner Transformer to the canvas and connect the ContourGenerator:Contours port. Your workspace should look like the screenshot below:

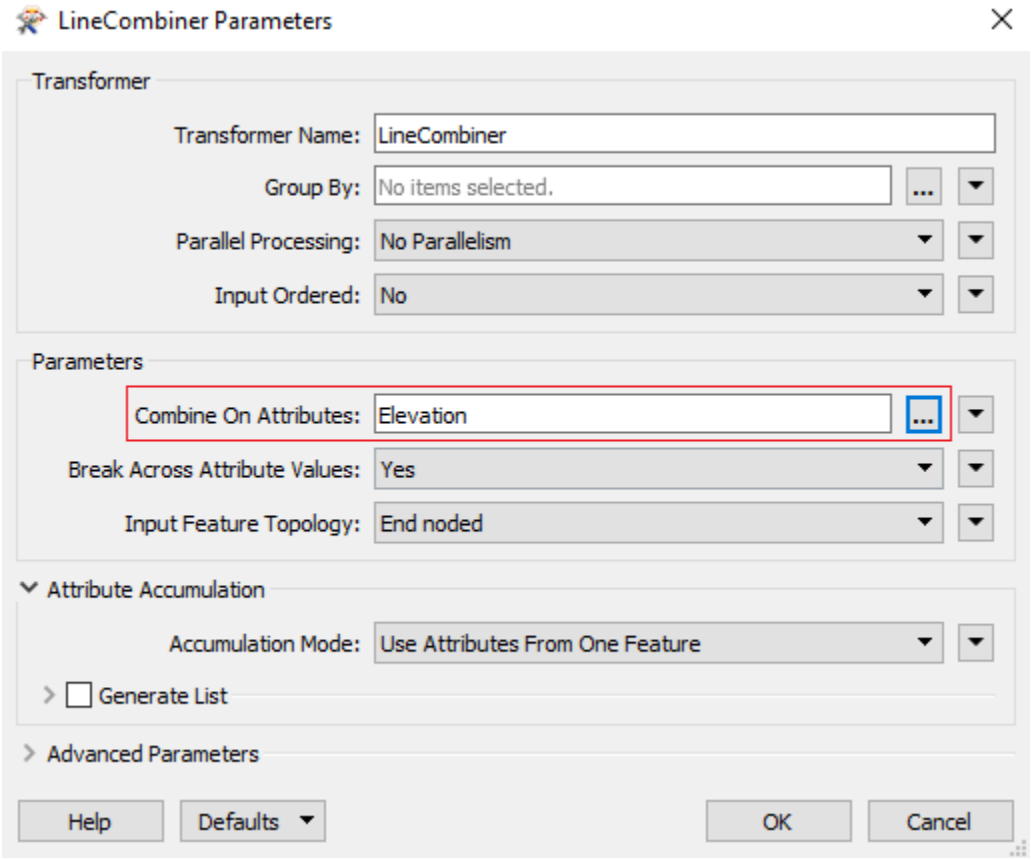


The purpose of this transformer is to combine paths to have one record per Path ID rather than having segmented contour lines.

- Open the LineCombiner Parameters.
- The only parameter needs to be changed is the Combine On Attributes. Click on the Combine On Attributes ellipsis to reveal a list of attributes to combine on. Select the Elevation attribute from the list, then click OK to return to the LineCombiner Parameters.
- Click OK to accept the changes and close the LineCombiner Parameters.

Note: In order for lines to be combined, they must meet at the exact same start/end point. Additionally, lines will remain broken at points where three or more lines converge. For more information see the LineCombiner Documentation.

- Leave all other parameters as default, then click OK to accept the changes to the LineCombiner Parameters.



8. Add an Inspector

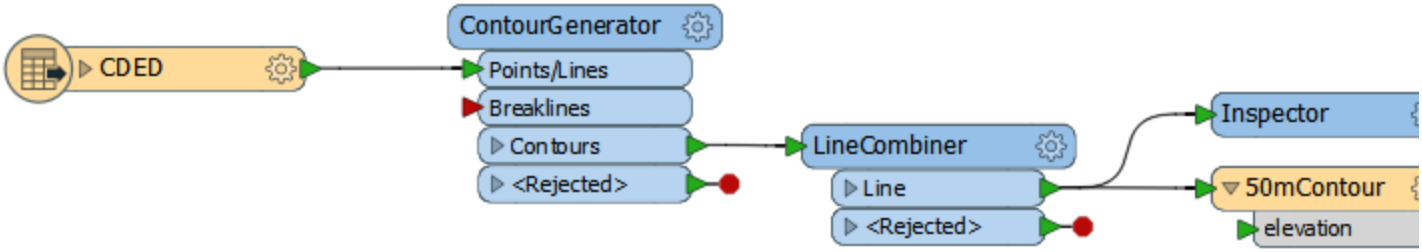
- Next, add an Inspector after the LineCombiner and connect it to the LineCombiner:Line port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

9. Add an Industry Foundation Class STEP Files (IFC) Writer

- Similar to adding a reader, type “IFC” then select the Industry Foundation Class STEP Files (IFC) format from the list of Writers.
- Click on the Dataset ellipsis to specify where to save your data and name the file: 50mContour (for example, C:\Users\Documents\FME\Contours\50mContour.ifc)
- Set the Layer Definition to Automatic to ensure all attributes that are created or renamed will be written to the IFC file.

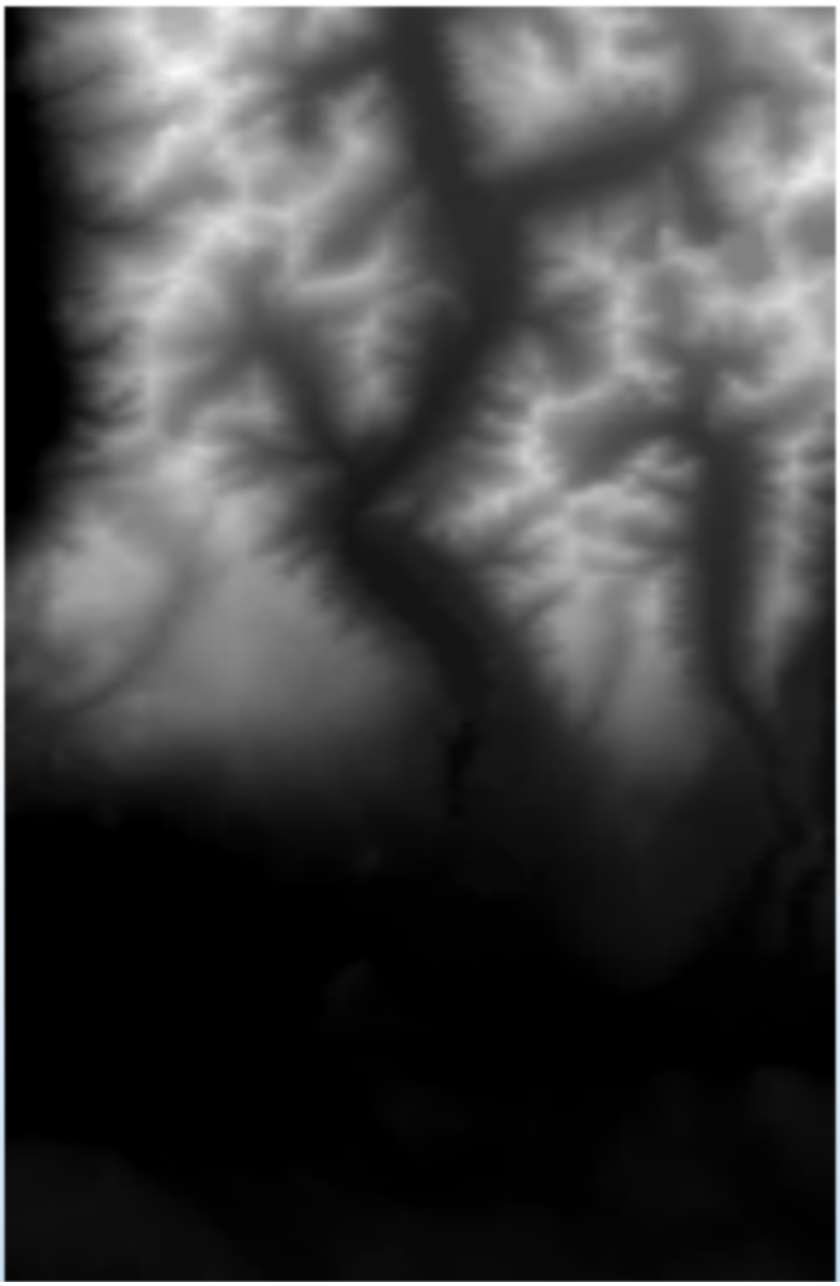
10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the final transformer, the output dataset will automatically be opened in the FME Data Inspector.

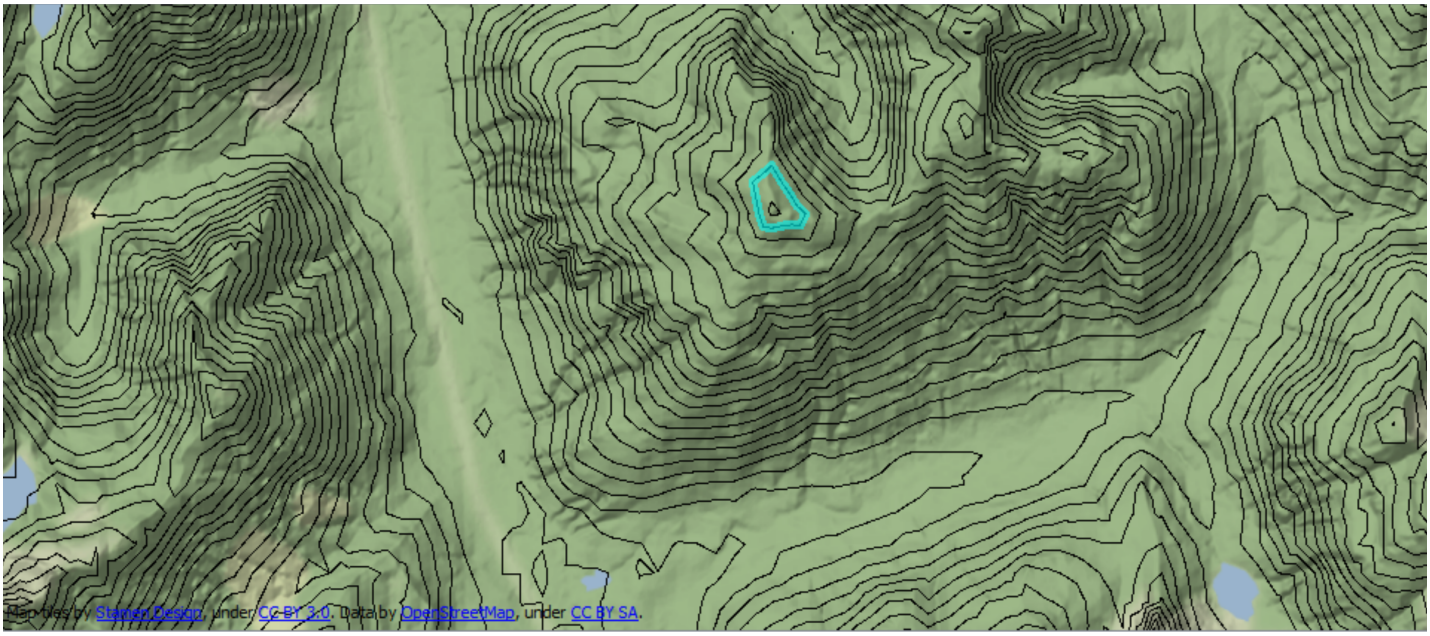


Results

Input



Output



Transformers

- ContourGenerator- Constructs a Delaunay triangulation based on input points and breaklines.
- Inspector- Sends features to the FME Data Inspector for display.
- LineCombiner - Takes lines and connects them to form longer lines.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Simple Examples Using the RasterExpressionEvaluator Transformer

Introduction

The RasterExpressionEvaluator is a transformer that evaluates expressions on each cell in a raster, such as algebraic operations or conditional statements. This article will demonstrate commonly used raster transformations using the RasterExpressionEvaluator.

Downloads

RasterExpressionEvaluator-Workspaces.zip *(All of the workspaces)*

Instructions

Example 1: Inverting Colors

RGBColorInverter.fmwt

This example demonstrates how to invert images with the following band interpretation list and expression:

Interpretation	Expression
Red8	255-A[0]
Green8	255-A[1]
Blue8	255-A[2]

The images show Heritage Mountain Elementary School and the surrounding area in Google Earth

Example 2: Brightness Correction

BrightnessCorrection.fmwt

If an image is too dark, we can boost brightness with the following simple expression:

Interpretation	Expression
Red8	A[0]*1.5
Green8	A[1]*1.5
Blue8	A[2]*1.5

Port Moody, Greater Vancouver

Note that this method should be used cautiously - if there are bright areas in the image, they will reach or exceed the maximum value (255 in my example), which will make them purely white (255, 255, 255 in case of RGB24), and the information in those areas will be lost.

Example 3: Color Correction

ColorCorrection.fmwt

If one of the colors on an image seems to dominate over the others, we can change only one band (or apply different coefficients to different bands):

Interpretation	Expression
Red8	A[0]
Green8	A[1]/1.2
Blue8	A[2]

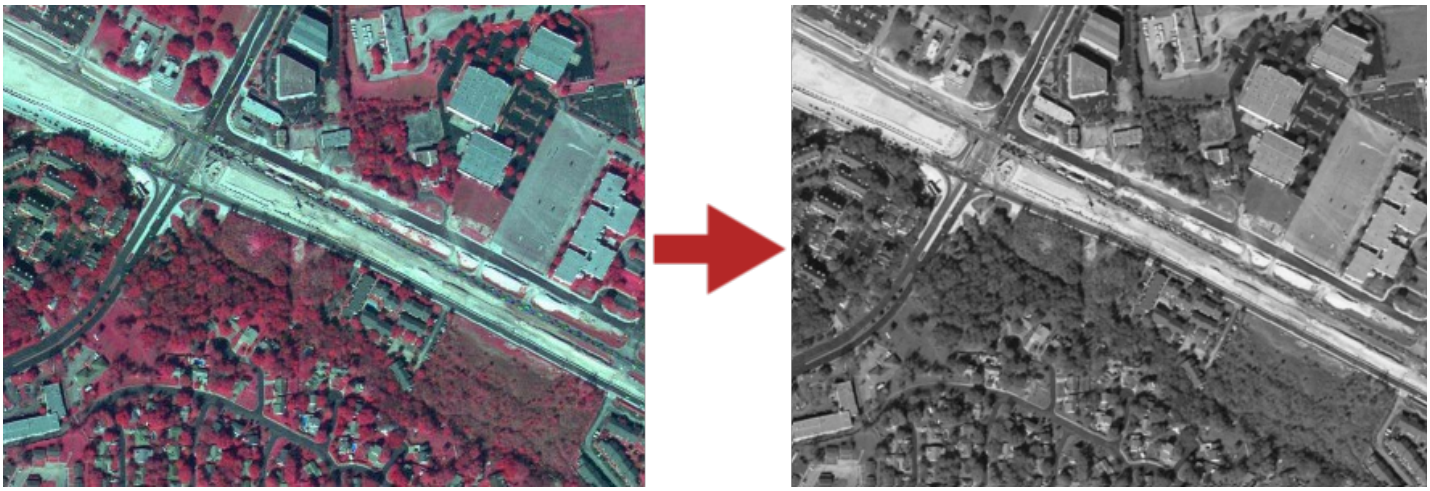
Port Moody, Greater Vancouver

Example 4: Color to Grayscale Conversion

RGB2Grayscale.fmwt

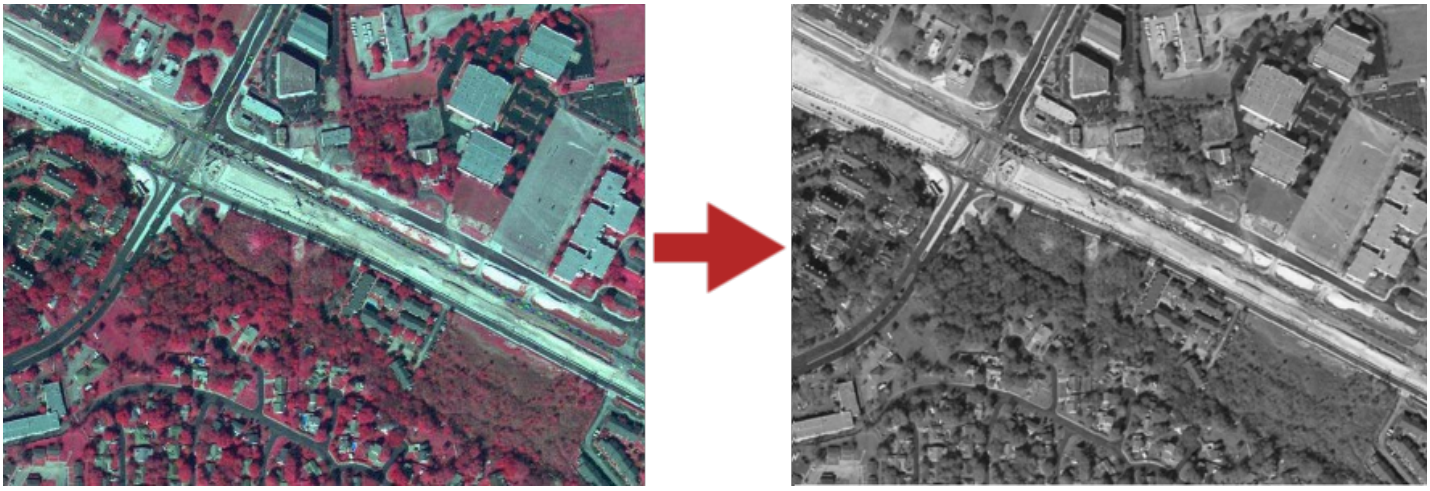
Most sources suggest the following formula for a color to grayscale conversion (and many sources also mention that depending on the image it may vary). There are two commonly used formulas the first formula gives better contrast:

Interpretation	Expression
Grey8	0.2989*A[0] + 0.5870*A[1] + 0.1140*A[2]



Vancouver, British Columbia

Interpretation	Expression
Grey8	0.3333*A[0] + 0.3333*A[1] + 0.3333*A[2]



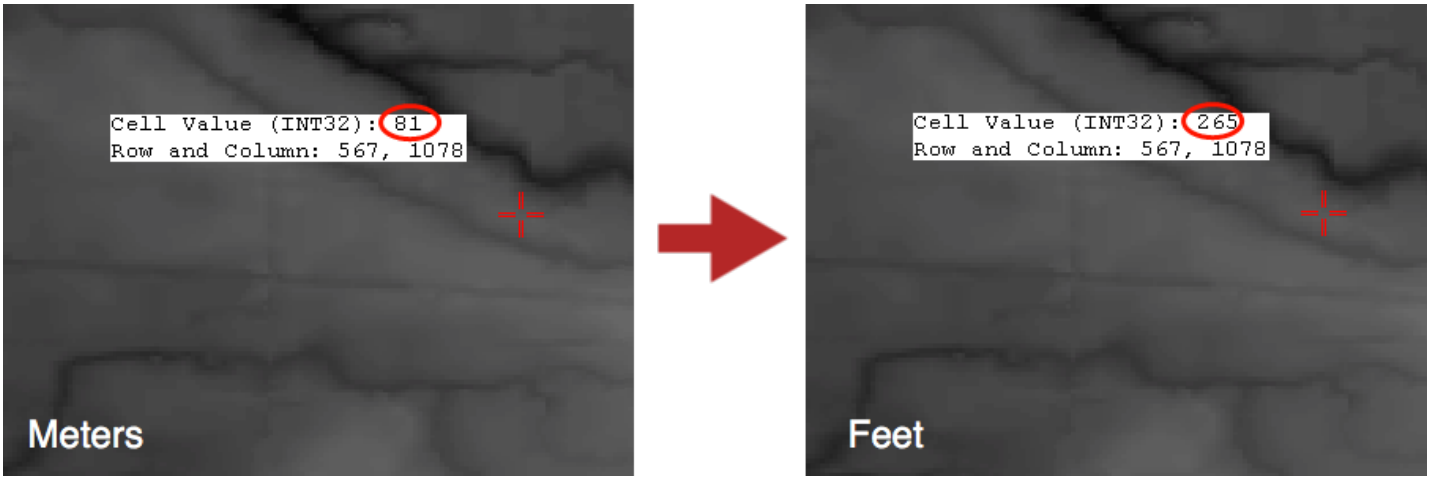
Vancouver, British Columbia

Example 5: Unit Conversion

UnitConversion.fmwt

Conversion between all kinds of units is a very natural operation for RasterExpressionEvaluator. In this example we take a numeric raster representing a DEM in meters and convert the units into feet:

Interpretation	Expression
Int32	A[0]/0.3048

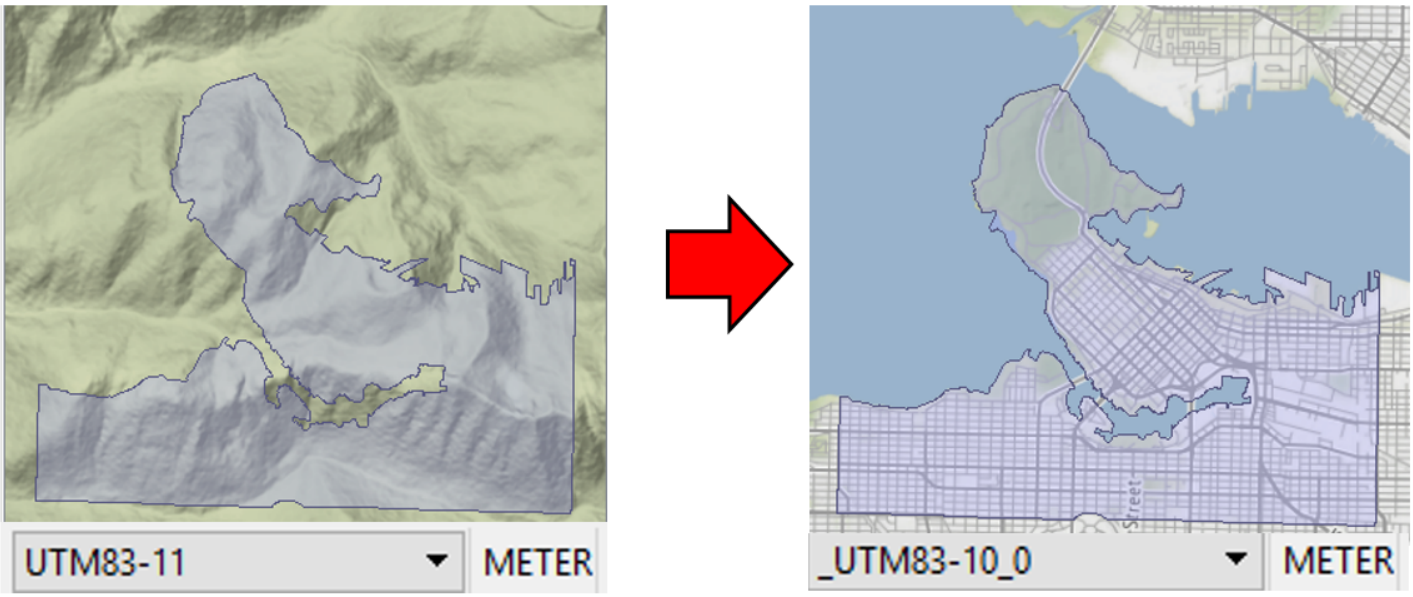


North Vancouver, Greater Vancouver

Projection Definition and Coordinate Transformation

Overview

In this tutorial, you will learn how to define a coordinate system in FME Workbench and how to reproject data to another coordinate system using the Reprojector Transformer.



Downloads

projection-definition-and-coordinate-transformation.fmwt

contoursgdb.zip

landboundarygdb.zip

Exercise 1 - *Defining Projections When Adding a Reader*

In this scenario, you want to define the projection of your dataset to match the coordinate system information that is stated in a metadata file. The polygon boundary you will use was inadvertently projected to the wrong coordinate system (UTM83-11) so you want to define the correct projection (UTM83-10) before performing your analysis.

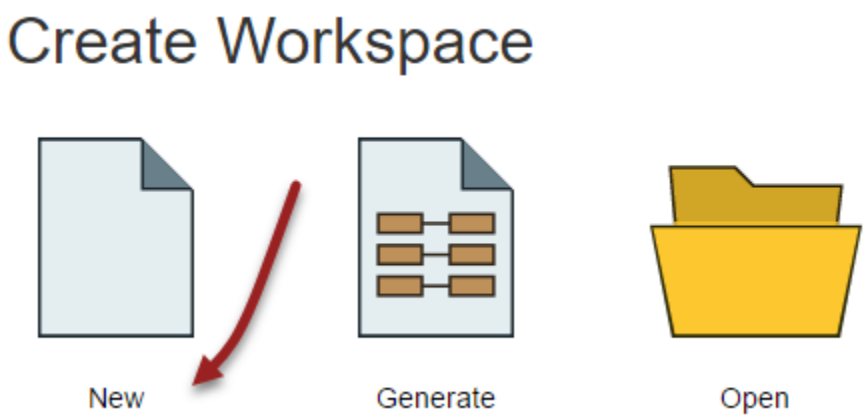
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

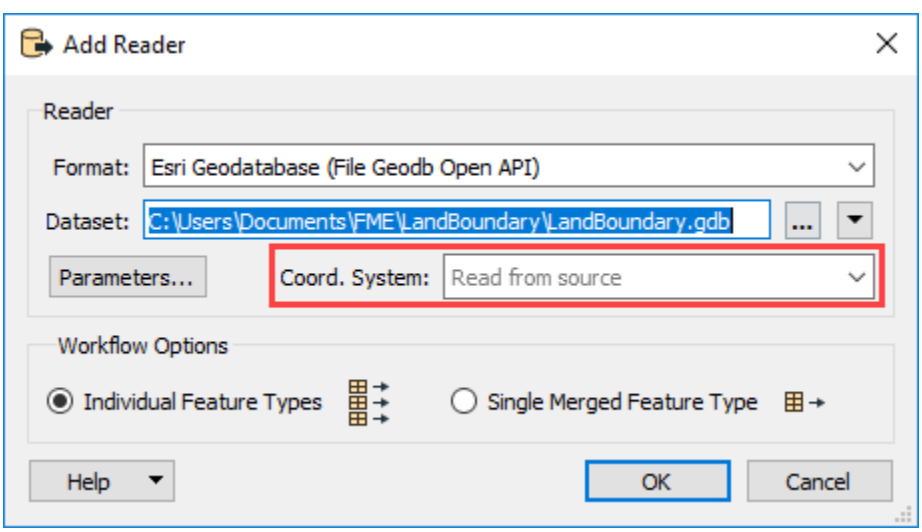


3. Add an Esri Geodatabase (File Geodb Open API) Reader

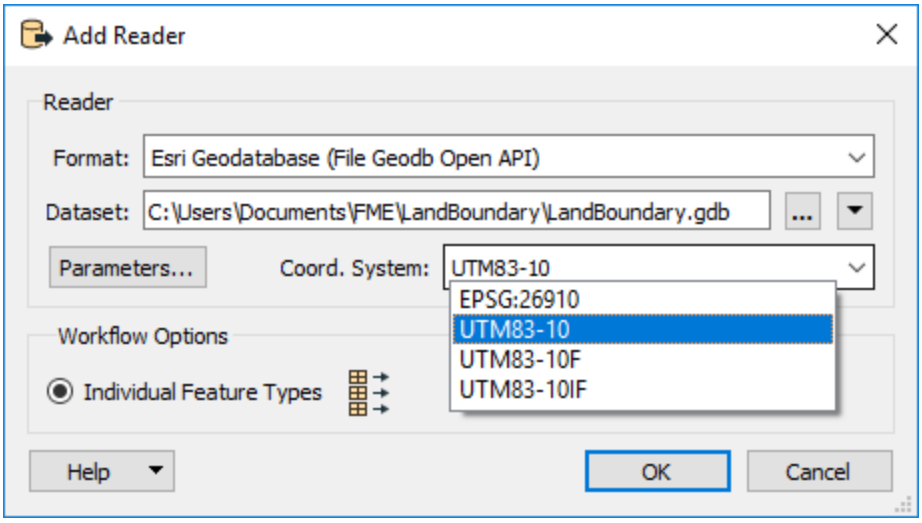
- Start typing "Geodatabase" without anything selected on the canvas, then select the Esri Geodatabase (File Geodb Open API) format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the LandBoundary Geodatabase (for example, C:\Users\Documents\FME\Landboundary\Landboundary.gdb).

4. Define the Coordinate System

The easiest way to define a coordinate system in FME is in the Add Reader dialog.



- To define a coordinate system, you can either select a coordinate system from the drop-down list or type in the name of the coordinate system (i.e. "UTM83-10") - the list will filter coordinate systems as you type and will only return coordinate systems that contain what is being typed in.



- Select the UTM83-10 from the list of Coordinate System or type in “UTM83-10” to define the coordinate system.

Alternatively, you can set the reader/writer’s Coordinate System in the Navigator window. See the Using Coordinate Systems within FME article for more information.

5. All Set!

- Click OK to add the Reader to the canvas with the correctly defined coordinate system. For more information on Coordinate System Transformations in FME see the Desktop Basic Course Manual.

Exercise 2 - *Reprojecting Data*

In Exercise 2, you will reproject data because it is currently using the LL83 Coordinate system (measured in degrees) and you need it to be in a coordinate system that uses linear units. In this scenario, you will reproject data from lat/long LL83 (degrees) to UTM83-10 (meters).

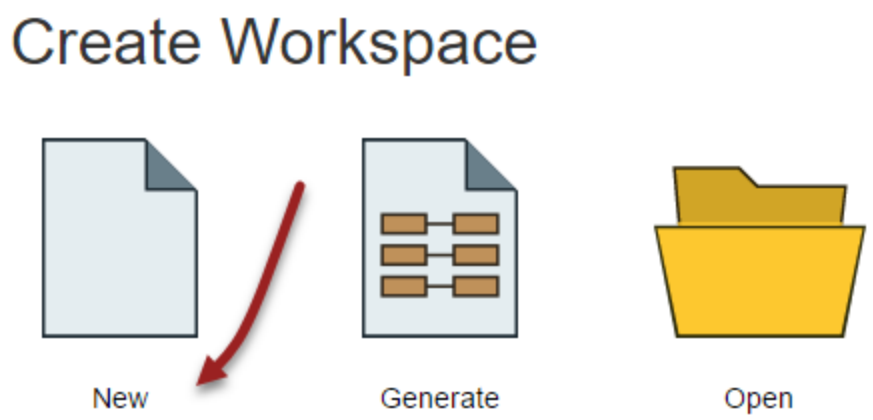
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

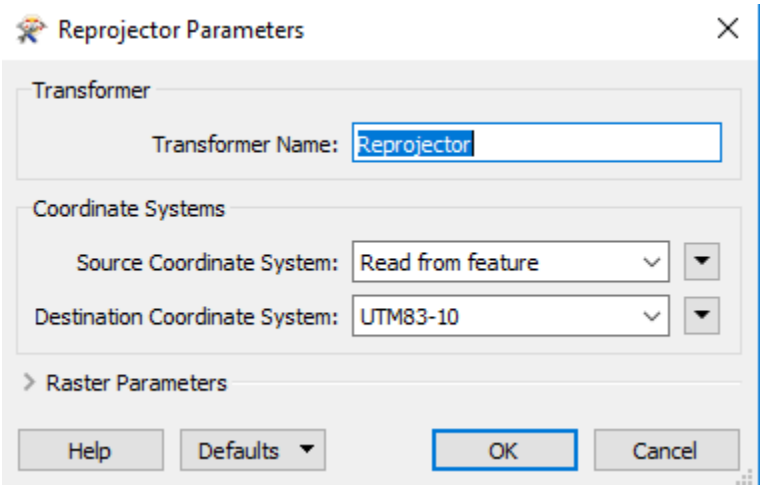


3. Add an Esri Geodatabase (File Geodb Open API) Reader

- Start typing “Geodatabase” without anything selected on the canvas, then select the Esri Geodatabase (File Geodb Open API) format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the Contours Geodatabase (for example, C:\Users\Documents\FME\ElevationModel\Contours.gdb).

4. Change the Coordinate System

- Add a Reprojector to the canvas by typing “Reprojector” to bring up the list of FME Transformers in the Quick Add Search. Select the Reprojector from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the Contours Reader to the Reprojector.
- Next, set the Destination Coordinate System to UTM83-10. This can be done by typing “UTM83-10” in the Destination Coordinate System text box (autofill will provide suggestions) or selecting the coordinate system from the drop-down list.



5. Add an Inspector

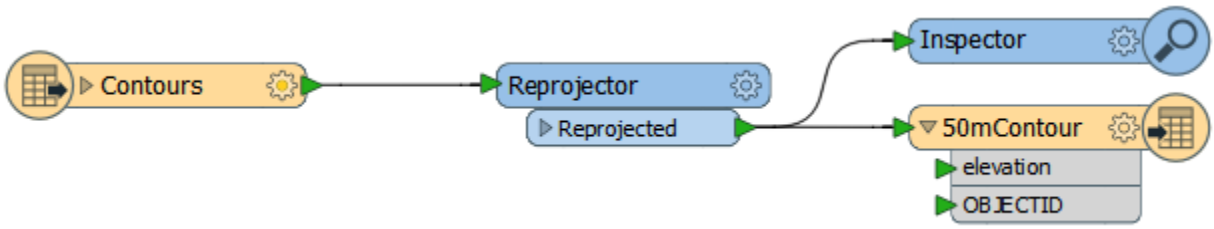
- Next, add an Inspector and connect it to the Reprojector:Reprojected port. This will automatically open the raster dataset in the FME Data Inspector after the translation has run.

6. Add a GML (Geography Markup Language) Writer

- Similar to adding a reader, type “GML” then select the GML (Geography Markup Language) format from the list of Writers.
- Click on the dataset ellipsis to specify where to save your data and name the file: 50mContour (for example, C:\Users\Documents\FME\Contours\50mContour.gml)
- Set the Layer Definition is set to Automatic to ensure all attributes that are created or renamed will be written to the GML file.

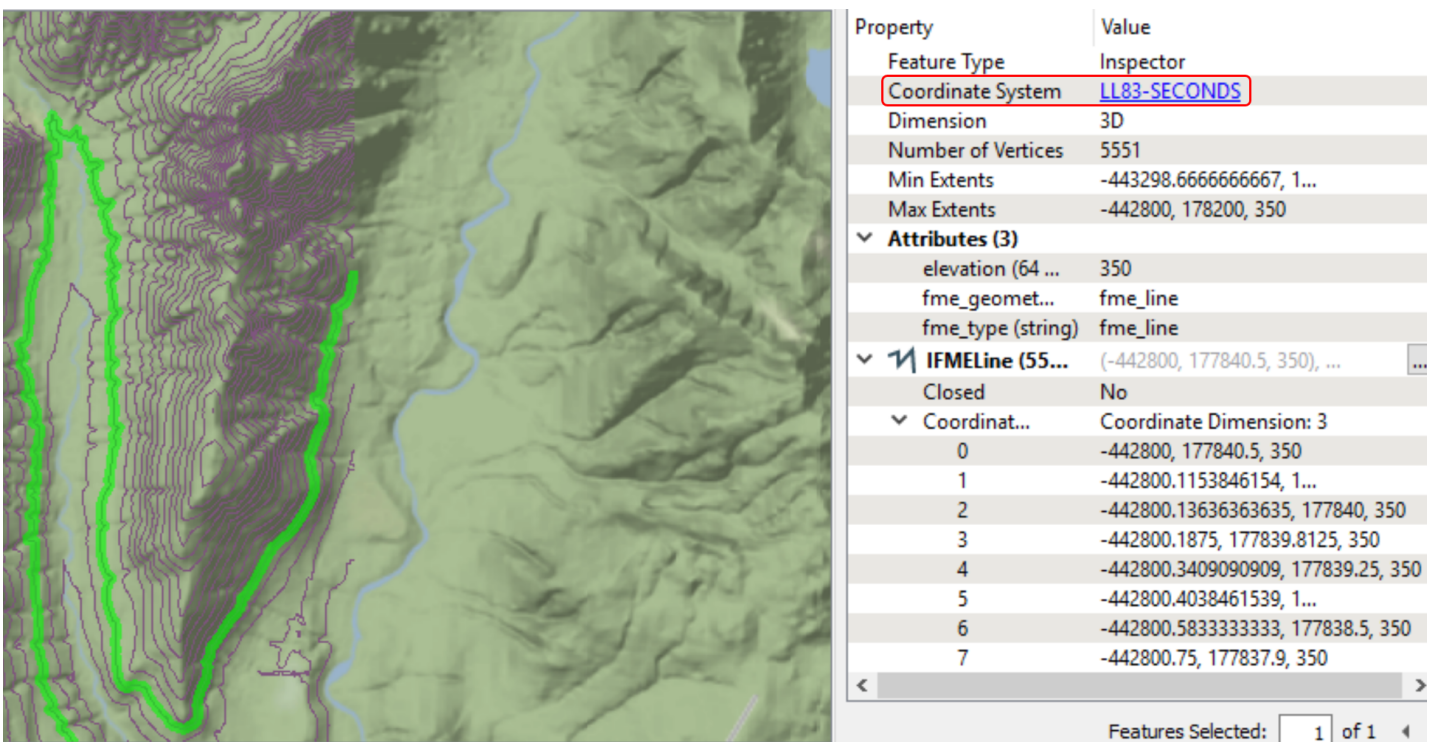
7. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the final transformer, the output dataset will automatically be opened in the FME Data Inspector.

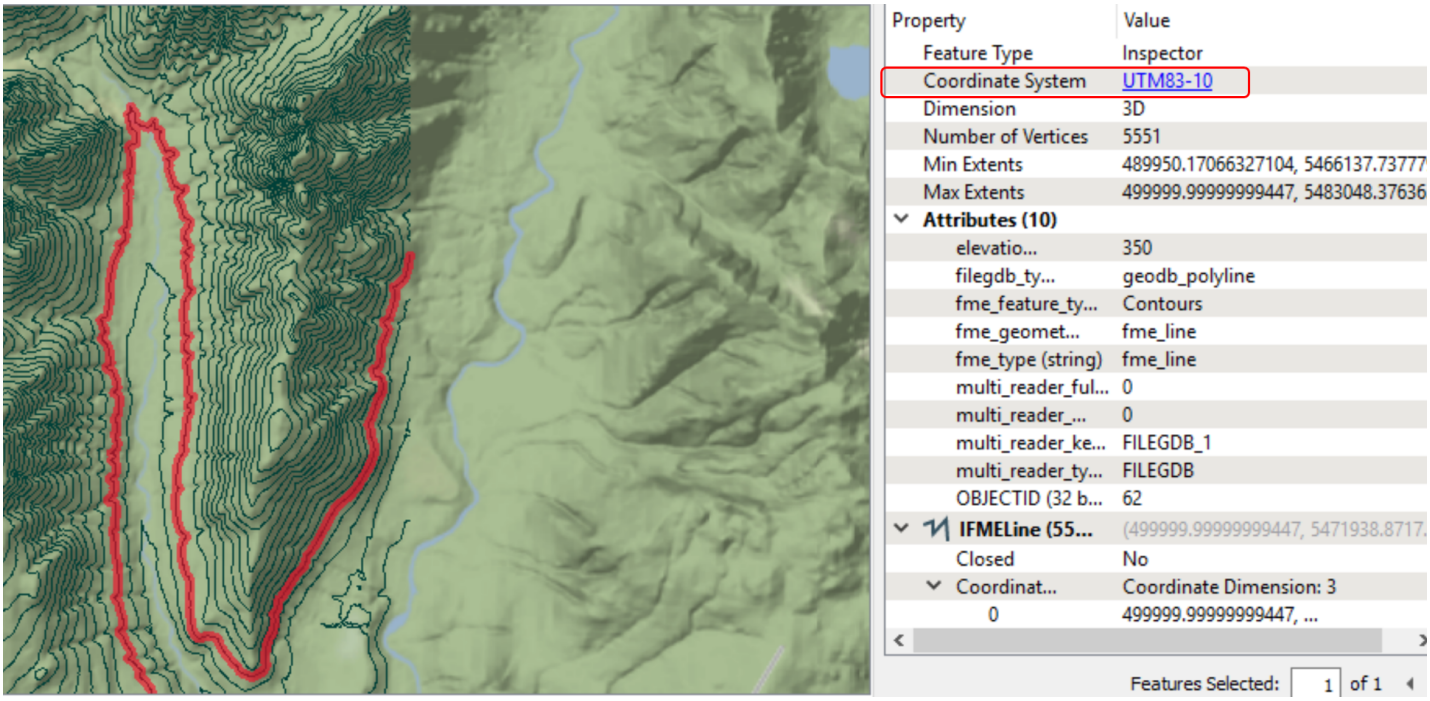


Results

Input



Output



Transformers

- Inspector- Sends features to the FME Data Inspector for display.
- Reprojector - Reprojects feature coordinates from one coordinate system to another.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Polygon in Polygon Overlay

Overview

In this exercise, you will learn how to perform a union between overlapping polygon features using the AreaOnAreaOverlayer. The AreaOnAreaOverlayer is typically used for three main operations: (1) deconstruct overlapping polygons to produce the intersections and differences, (2) compare multiple datasets for area overlaps, and (3) area calculations based on overlapping area data.

Downloads

polygon-in-polygon-overlay.fmwt

historiczones.zip

neighborhoodsgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you will be performing area calculations based on overlapping historic zones and neighborhood polygons. This will allow you to identify which historic areas belong to each neighborhood and the total area of each historic zone.

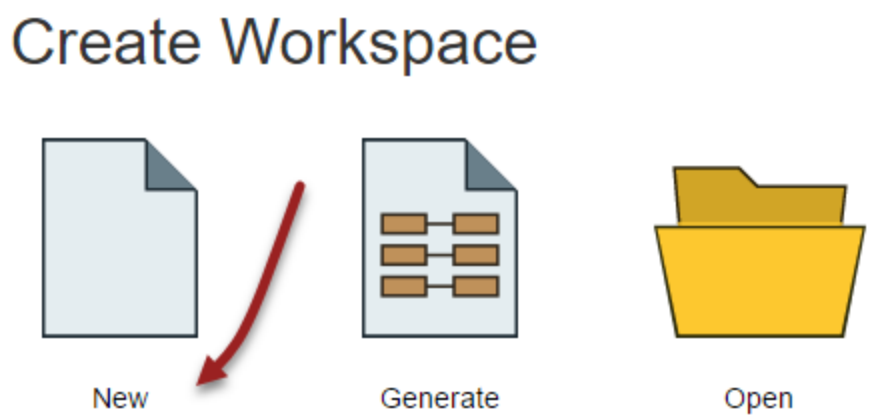
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

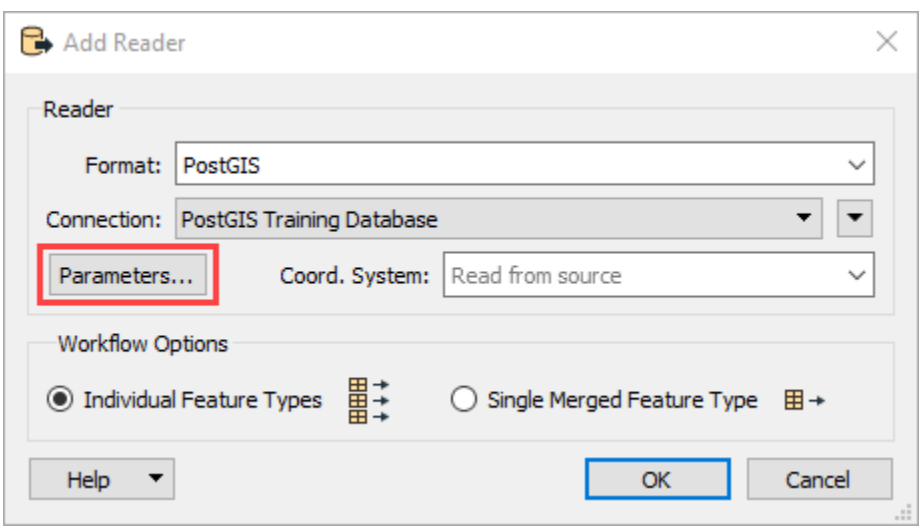
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



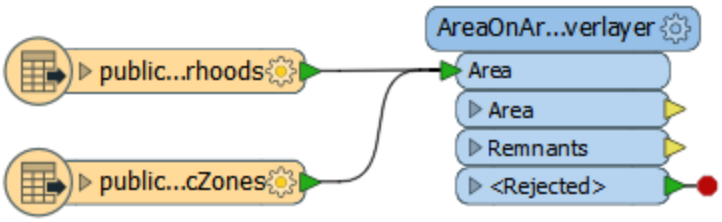
3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Neighborhoods and HistoricZones tables from the Table List.



4. Add an AreaOnAreaOverlayer Transformer

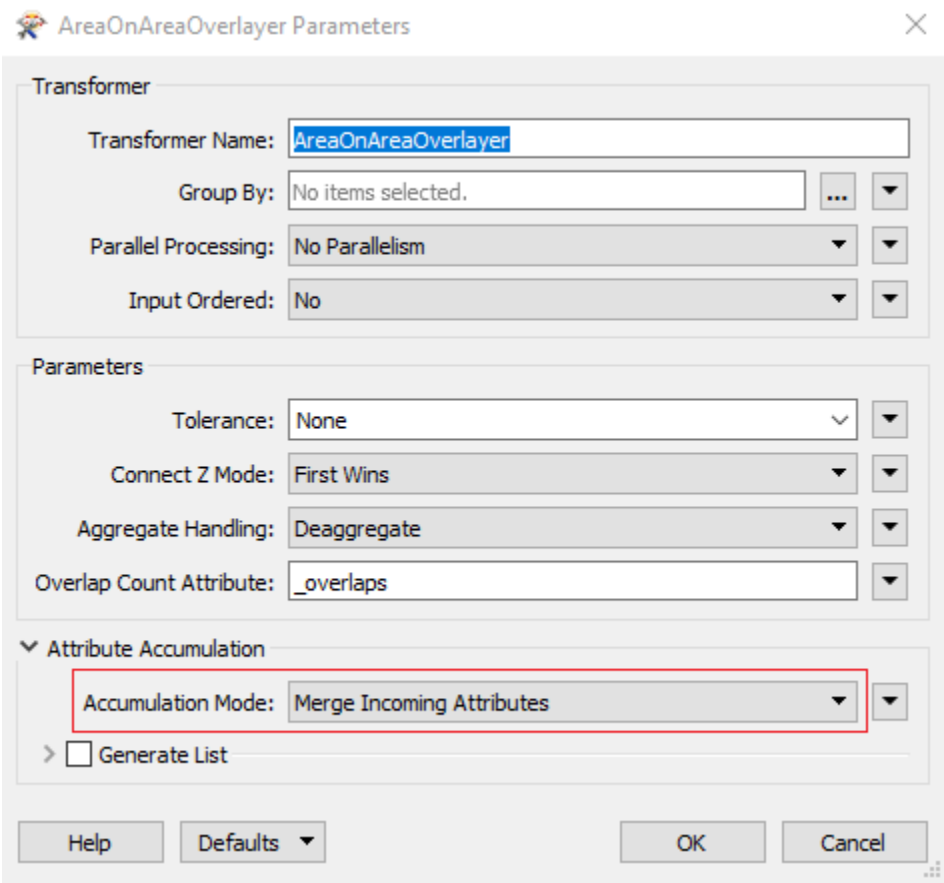
- Select the Neighborhoods and HistoricZones Reader Feature Types and type “AreaOnAreaOverlayer” then select the AreaOnAreaOverlayer from the list of Transformers in the Quick Add Search. This will automatically make the connection from the Neighborhood and HistoricZones Readers to the AreaOnAreaOverlayer. The Neighborhoods and HistoricZones Reader Feature Types should be connected to the AreaOnAreaOverlayer:Area port.



5. Set the Attribute Accumulation Mode

The AreaOnAreaOverlayer transformer will perform an area-on-area overlay for intersecting polygons - all polygons are considered against each other, and where they overlap, new polygons are created that represent both the overlapping area and the original areas with the overlap removed. For more technical details about the AreaOnAreaOverlayer. For more technical details about the AreaOnAreaOverlayer, see the Documentation.

- Once the AreaOnAreaOverlayer has been added, double-click the AreaOnAreaOverlayer or click on the gear icon to open the transformer parameters.
- Set the Attribute Accumulation Mode to Merge Incoming Attributes. This will allow you to join the attributes from both the HistoricZones and Neighborhood polygons where features overlap.



6. Calculate the Area of Each Polygon

- Add an AreaCalculator after the AreaOnAreaOverlayer - this will automatically calculate the area of each polygon that is created by the AreaOnAreaOverlayer.
- Connect the AreaCalculator to the AreaOnAreaOverlayer:Area port.

Note: area is calculated in square map units. For a more in-depth tutorial on area calculations, see the [Area Measurements with the AreaCalculator](#) article.

7. Add an Inspector

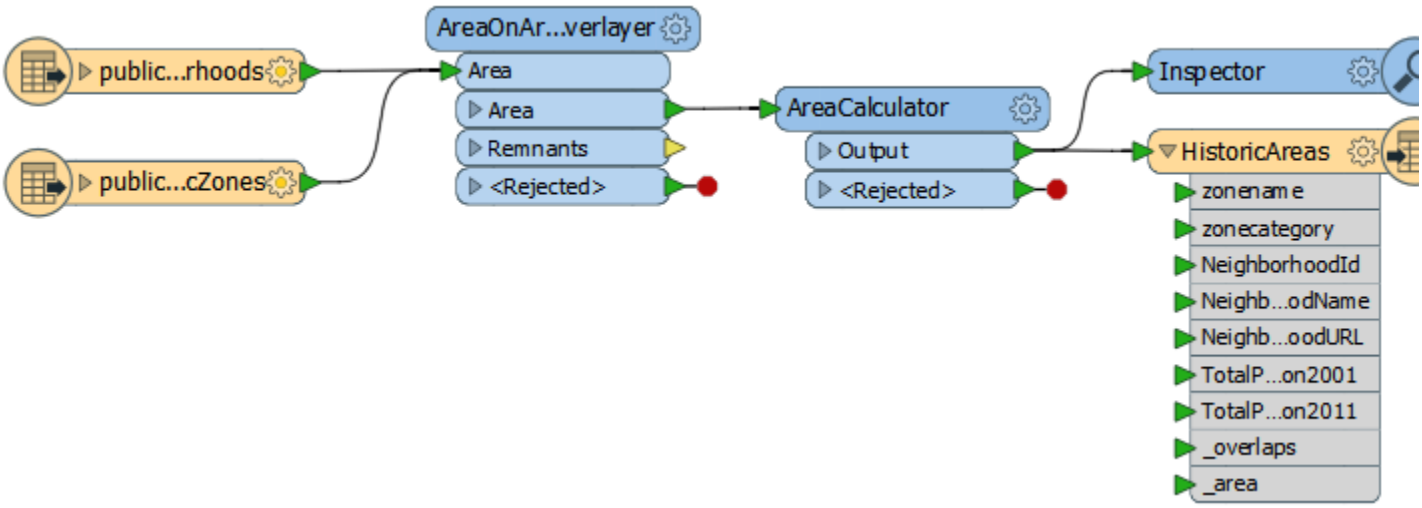
- Next, add an Inspector and connect it to the AreaOnAreaOverlayer:Output port.

8. Add a GeoJSON (Geographic JavaScript Object Notation) Writer

- Similar to adding a reader, type “GeoJSON” then select the GeoJSON format from the list of Writers.
- Click on the Dataset ellipsis to specify where to save your data and name the file: HistoricAreas (for example, C:\Users\Documents\FME\HistoricAreas\HistoricAreas.json)
- Set the Layer Definition to Automatic to ensure all attributes that are created or renamed will be written to the GeoJSON file.
- Connect the GeoJSON writer to the AreaOnAreaOverlayer:Output port.

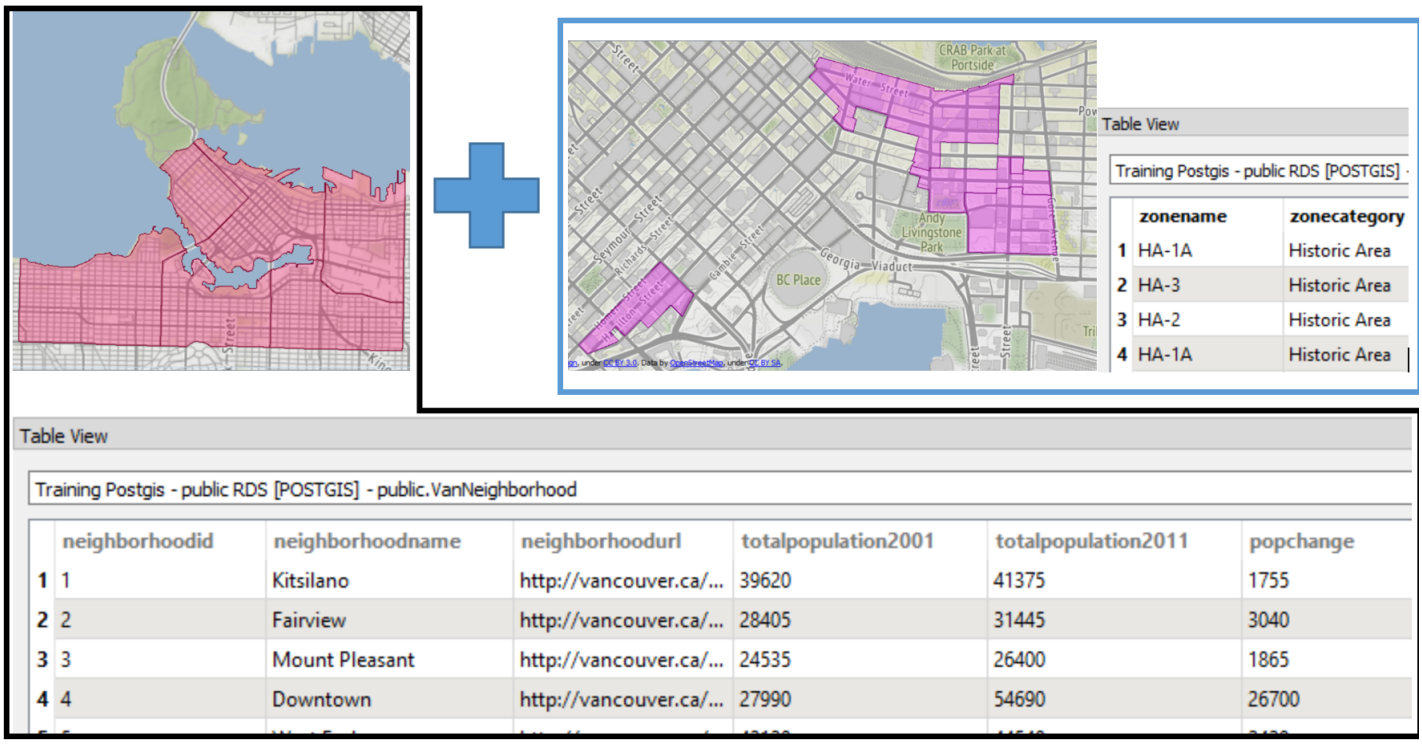
9. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the final transformer, the output dataset will automatically be opened in the FME Data Inspector.



Results

Input


CRAB Park at Portside
Water Street
Andy Livingstone Park
BC Place
Georgia Viaduct
Table View
Training Postgis - public RDS [POSTGIS]
zonename
zonecategory
1 HA-1A Historic Area
2 HA-3 Historic Area
3 HA-2 Historic Area
4 HA-1A Historic Area
Table View
Training Postgis - public RDS [POSTGIS] - public.VanNeighborhood
neighborhoodid
neighborhoodname
neighborhoodurl
totalpopulation2001
totalpopulation2011
popchange
1 1 Kitsilano http://vancouver.ca/... 39620 41375 1755
2 2 Fairview http://vancouver.ca/... 28405 31445 3040
3 3 Mount Pleasant http://vancouver.ca/... 24535 26400 1865
4 4 Downtown http://vancouver.ca/... 27990 54690 26700

Output

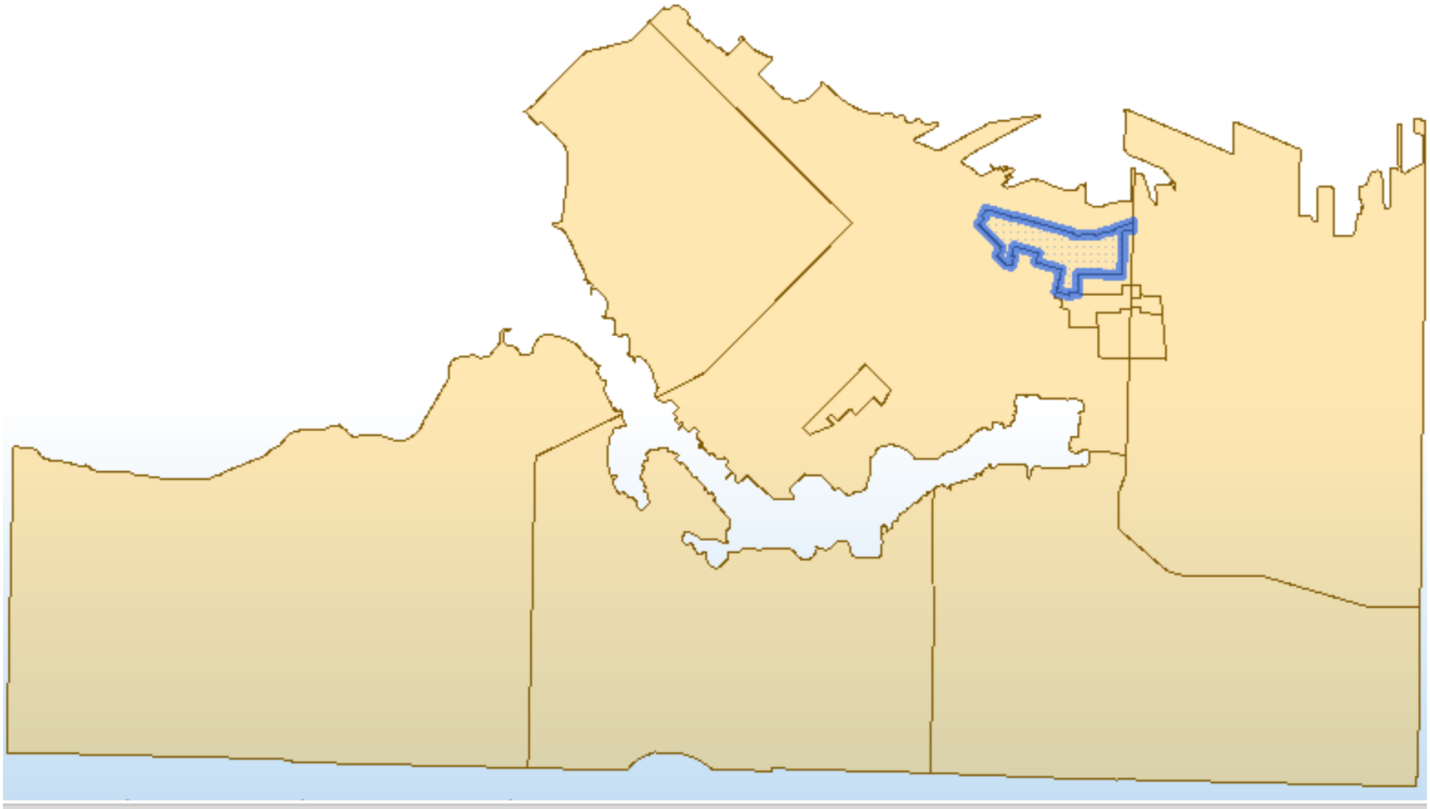


Table View

inspector [FFS] - Inspector

	neighborhoodid	neighborhoodname	neighborhoodur	totalpopulation2001	totalpopulation2011	popchange	zonename	zonecategory	_overlaps	_area
1	1	Kitsilano	http://vancouv...	39620	41375	1755	<missing>	<missing>	1	5475379.337504...
2	2	Fairview	http://vancouv...	28405	31445	3040	<missing>	<missing>	1	3274417.636603...
3	3	Mount Pleasant	http://vancouv...	24535	26400	1865	<missing>	<missing>	1	3663949.285246...
4	4	Downtown	http://vancouv...	27990	54690	26700	HA-2	Historic Area	3	179658.8718754...

Transformers

- AreaCalculator - Calculates the area of a polygonal object and stores the value in an attribute.
- AreaOnAreaOverlayer - Performs an area-on-area overlay (intersection of polygons) so that all input areas are intersected against each other and resultant area features are created and output.
- Inspector - Sends features to the FME Data Inspector for display.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Point in Polygon Overlay

Overview

In this exercise, you will learn how to perform a spatial join between point features that are contained within polygons using the SpatialFilter. The SpatialFilter is typically used for three main operations:

- Determining whether a spatial relationship between features exists based on a spatial predicate test
- Performing quality control on a dataset by checking for expected spatial relationships with another dataset
- Performing a spatial join to transfer attributes from one feature to another based on their spatial relationship

---

Downloads

point-in-polygon-overlay.fmwt

mountpleasantparcels.zip

mountpleasantpostaladdress.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the

PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you will learn how to spatially join address point features with parcel polygon features then output the joined dataset to the Esri Geodatabase format. If you want to learn more about Transformers that perform Spatial Joins, see the Desktop Basic Course Manual.

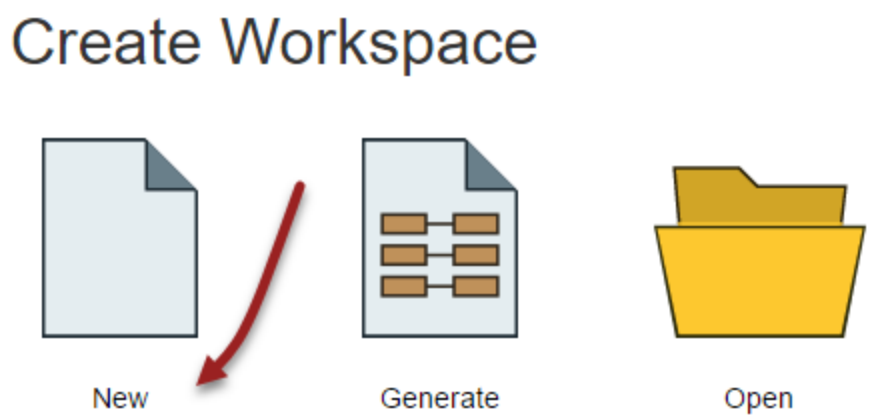
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

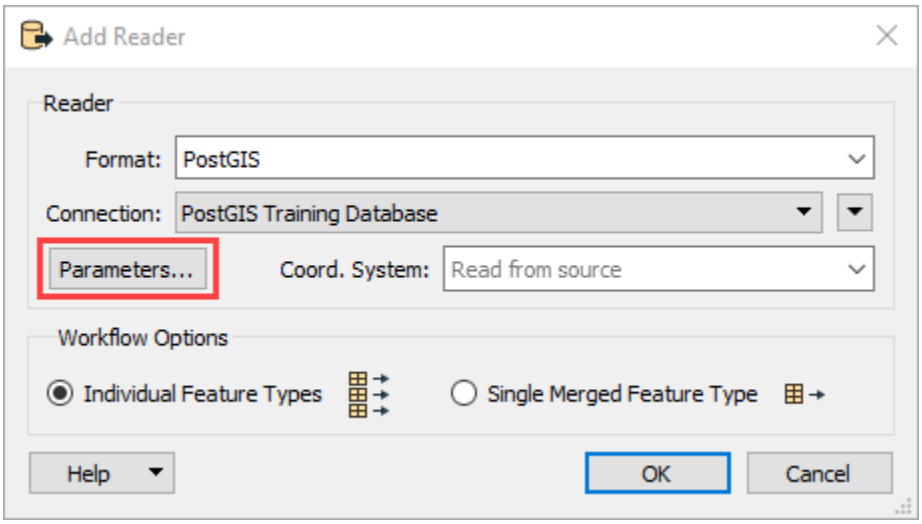
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the MountPleasantParcels and MountPleasantPostalAddress tables from the Table List.



4. Add a SpatialFilter Transformer

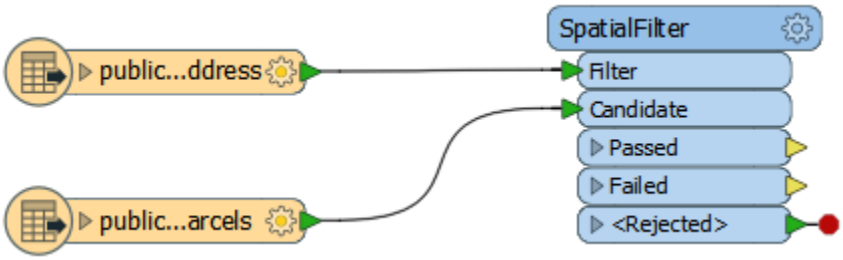
The SpatialFilter compares two sets of features to see if their spatial relationships meet the selected test condition(s).

- Similar to adding a reader, type “SpatialFilter” to bring up the list of FME transformers. Select the SpatialFilter from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.

5. Connect the Transformers Inputs

- Click and drag the MountPleasantPostalAddress to the SpatialFilter:Filter port. Similarly, connect the MountPleasantParcels to the SpatialFilter:Candidate port.

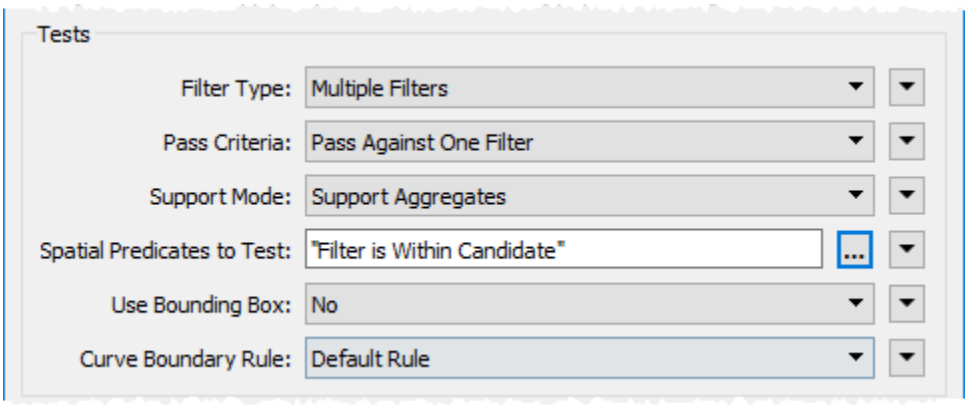
The connections to the Filter and Candidate ports on the SpatialFilter should match the screenshot provided below:



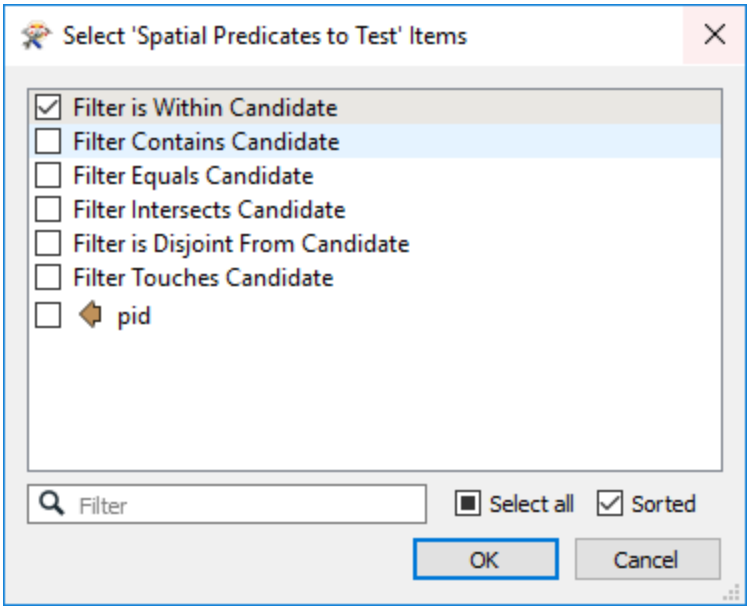
Note: the SpatialFilter outputs will not output Filter features, only Candidate features. For more technical details, see the [SpatialFilter Documentation](#).

6. Set the Spatial Predicates to Test

- Once the connections to the SpatialFilter have been made, double-click the SpatialFilter or click on the gear icon to open the transformer parameters.
- In the Test section, click on the Spatial Predicates to Test ellipsis to reveal a set of test conditions.



- Deselect the default spatial predicate, Filter Intersects Candidate, and select Filter is Within Candidate. Your dialog should look like the screenshot provided below:



Once the selection has been made, click OK in the open dialogs to accept the changes and return to the workspace.

7. Add an Inspector

- Next, add an Inspector after the SpatialFilter and connect it to the SpatialFilter:Passed port.

This will automatically open the joined dataset in the FME Data Inspector after the translation has successfully run.

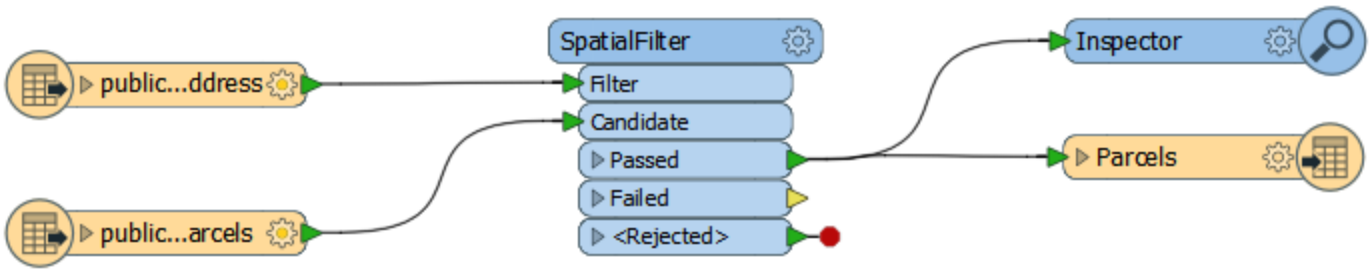
8. Add an Esri Geodatabase (File Geodb Open API) Writer

- Similar to adding a reader, type “Geodatabase” then select the Esri Geodatabase (File Geodb Open API) format from the list of Writers.
- Specify the File Geodatabase Folder by clicking on the Dataset ellipsis, creating a new folder or using an existing folder with the file extension .gdb, then selecting the .gdb folder (i.e. C:\Users\Documents\Parcels\Parcels.gdb). After specifying the file geodatabase folder location, set the Table Definition to Automatic and click OK. For more detailed information on working with Esri Geodatabases in FME, see the Getting Started with Geodatabases tutorial.
- Lastly, set the Feature Class Name in the Writer Parameters and Geometry in the dialog that appears, then click OK to add the Geodatabase Writer to the Canvas. In this case, the SpatialFilter will output polygon features, so you should set the geometry type to: geodb_polygon.
- Once the File Geodatabase Writer has been added, connect it to the SpatialFilter:Passed port.

Note: FME supports a number of readers/writers for SDE and Geodatabase; however, this depends on the platform you are running on. For more information, see the Comparison of FME readers and writers for SDE and Geodatabase article.

9. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector to the SpatialFilter:Passed port, the polygon dataset will automatically be opened in the FME Data Inspector.



Results

You have now successfully joined the PostalAddress point dataset to the MountPleasantParcels polygon dataset through a spatial relationship.

Input

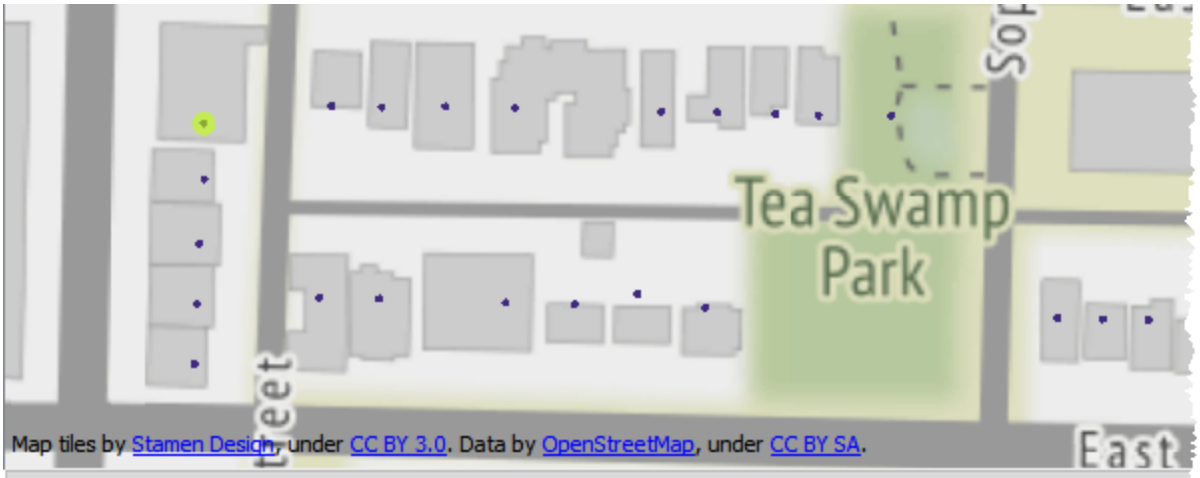


Table View

PostGIS Training Database [POSTGIS] - public.MountPleasantPostalAddress

public.MountPleasantParcels | public.MountPleasantPostalAddress

	addressid	postaladdr	siteaddres	ownername1	ownername2	postalad00	postalcity	postalpr
1534	181434	{9046f04b...				3102 Main St	Vancouver	BC
1535	181433	{efe161ec-...				210 E 15th ...	Vancouver	BC

in any column

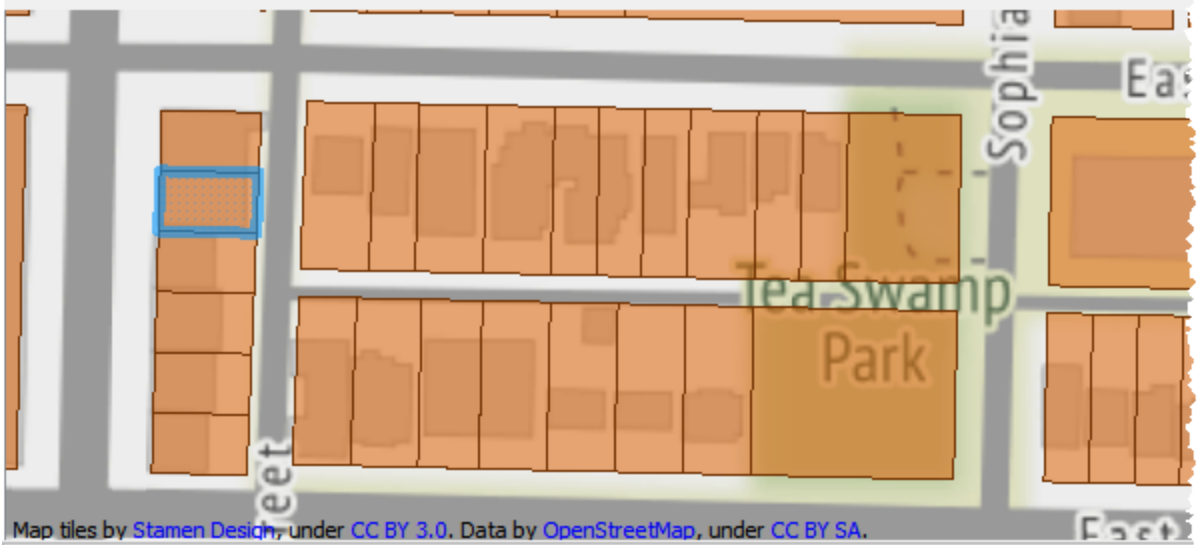


Table View

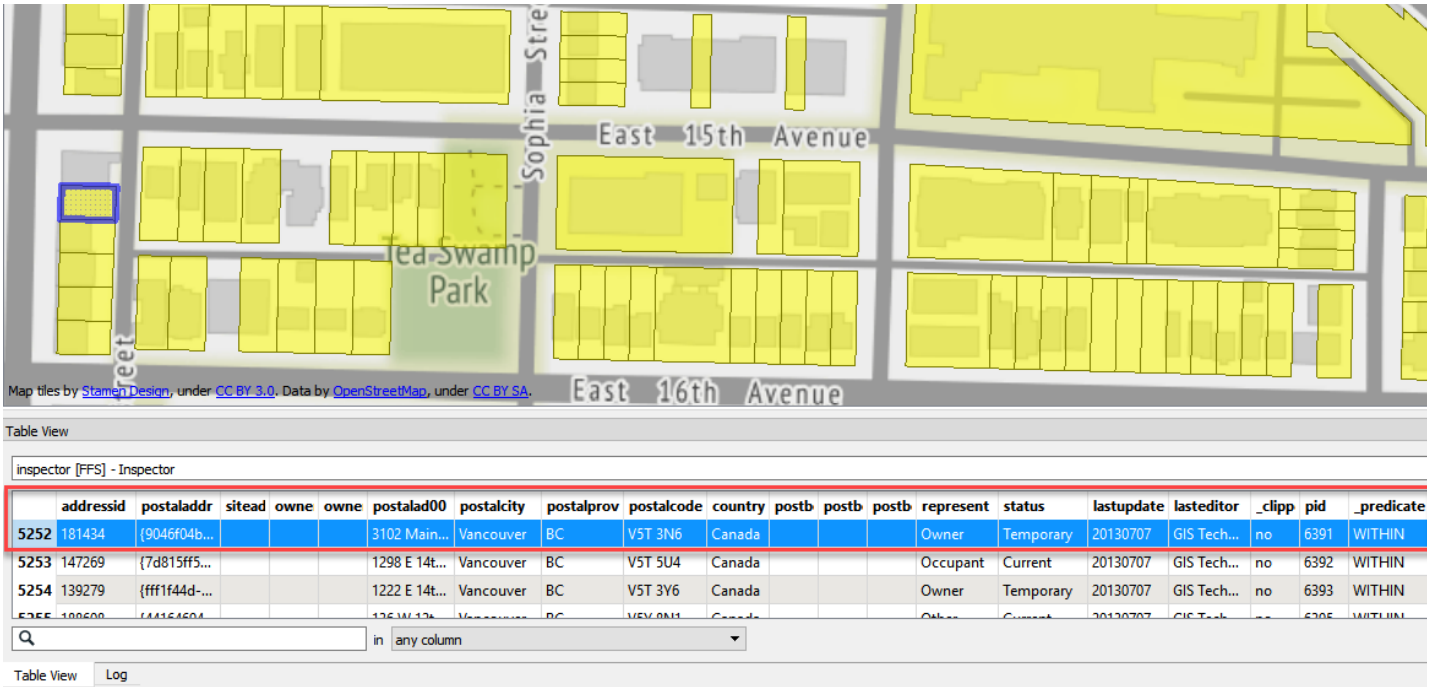
PostGIS Training Database [POSTGIS] - public.MountPleasantParcels

public.MountPleasantParcels | public.MountPleasantPostalAddress

	pid
6393	6391
6394	6392

in any column

Output



Transformers

- Inspector - Sends features to the FME Data Inspector for display.
- SpatialFilter - Filters point, line, area, and text features based on spatial relationships

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Topological Intersection

Overview

Topological intersection is the process of intersecting multiple polygon layers. In this exercise, you will use the SpatialFilter to set a spatial predicate that will determine which polygon features are merged together. The SpatialFilter compares two sets of features to see if their spatial relationships meet selected test conditions. The features being tested (Candidate features) are identified as having Passed or Failed the test. For example, if you have a roads dataset (lines), and wanted to extract all the roads that passed through parks (polygons), you would direct the roads into the Candidate input port, and the parks into the Filter input port.

Downloads

topological-intersection.fmwt

parksgdb.zip

clipperdata.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you want to merge the MountPleasantClipBoundary and Parks polygon features and their attributes based on the spatial predicate: “Filter Intersect Candidate”. In this case, the clip boundary is the filter because you only want to extract the park polygons that intersect the MountPleasantClipBoundary - this will also join the attributes of both features.

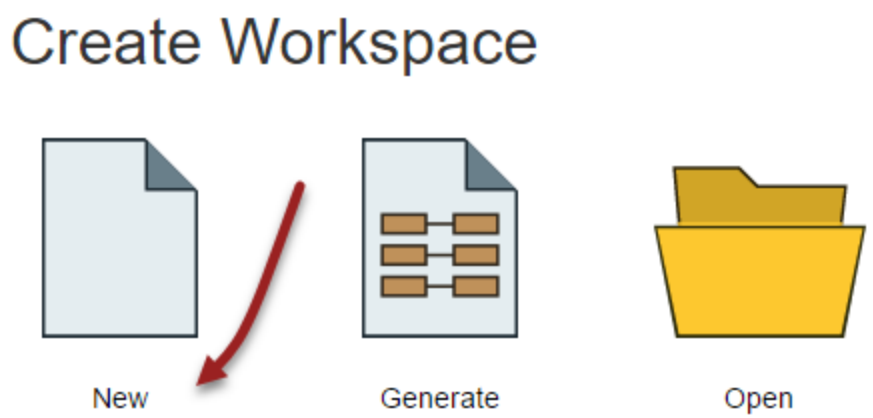
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

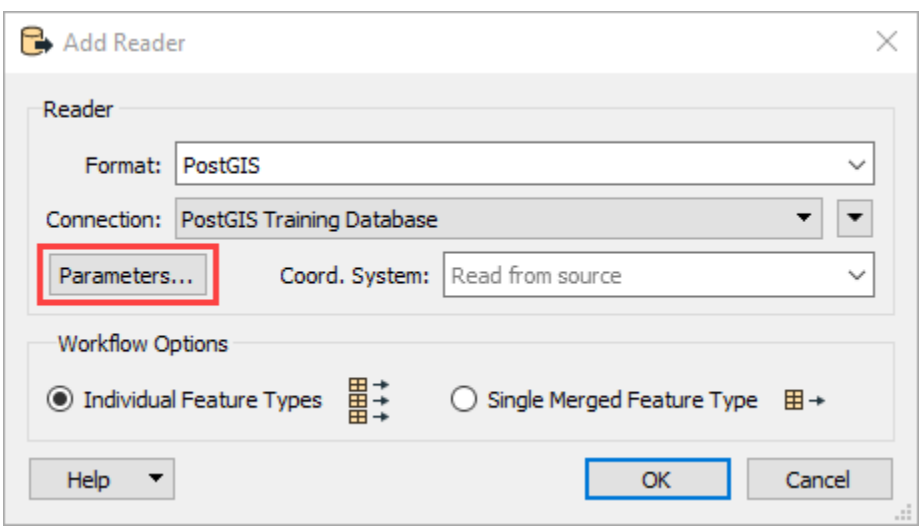
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the MountPleasantClipBoundary and Parks tables from the Table List.



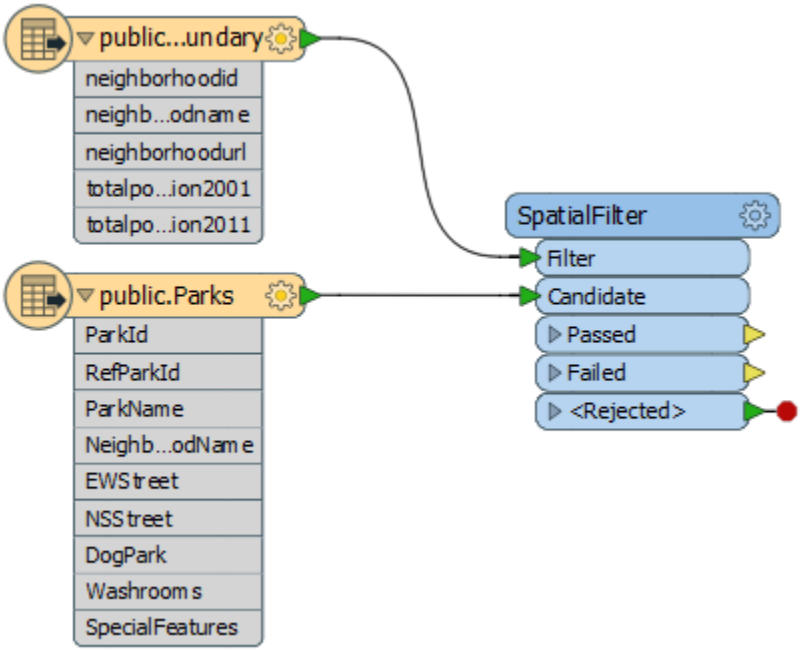
4. Add a Spatial Filter

- Similar to adding a reader, type “SpatialFilter” to bring up the list of FME transformers. Select the SpatialFilter from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.

5. Connect the Transformers Inputs

- Click and drag the MountPleasantClipBoundary Reader to the SpatialFilter:Filter port. Similarly,

connect the Parks Reader SpatialFilter:Candidate port. Your connections should look like the screenshot below:



In this case, you do not have to modify any of the SpatialFilter's parameters as the default is spatial predicate is "Filter Intersects Candidate" and Merge Attributes is selected. For more information on Spatially Based Join Transformers, see the Desktop Basic Course Manual. Additionally, you can learn more about each of the spatial predicates in the Defining Spatial Relationships *as well as technical details about the* AreaOnAreaOverlayer *in the Documentation.*

6. Add an Inspector

- Next, add an Inspector after the SpatialFilter and connect it to the SpatialFilter:Passed port.

This will automatically open the joined dataset in the FME Data Inspector after the translation has successfully run.

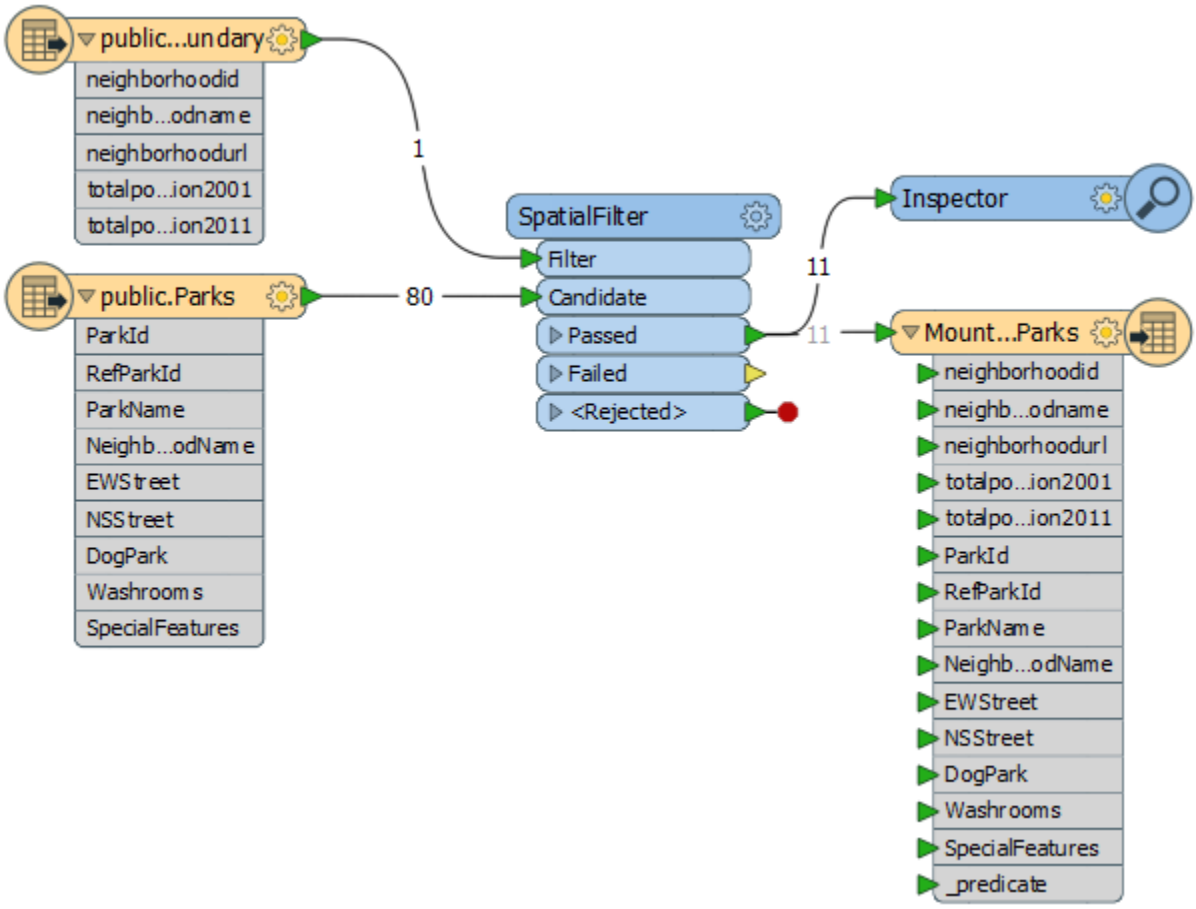
7. Add an Esri Geodatabase (File Geodb Open API) Writer

- Similar to adding a reader, type "Geodatabase" then select the Esri Geodatabase (File Geodb Open API) format from the list of Writers.
- Specify the File Geodatabase Folder by clicking on the Dataset ellipsis, creating a new folder or using an existing folder with the file extension .gdb, then selecting the .gdb folder (i.e. C:\Users\Documents\Parks\MountPleasantParks.gdb). After specifying the file geodatabase folder location, set the Table Definition to Automatic and click OK. For more detailed information on working with Esri Geodatabases in FME, see the Getting Started with Geodatabase article.
- Lastly, set the Feature Class Name in the Writer Parameters and Geometry in the dialog that appears, then click OK to add the Geodatabase Writer to the Canvas. In this case, the SpatialFilter will output polygon features, so you should set the geometry type to: geodb_polygon.
- Once the File Geodatabase Writer has been added, connect it to the SpatialFilter:Passed port.

Note: FME supports a number of readers/writers for SDE and Geodatabase; however, this depends on the platform you are running on. For more information, see the Comparison of FME readers and writers for SDE and Geodatabase article.

8. Run the Workspace

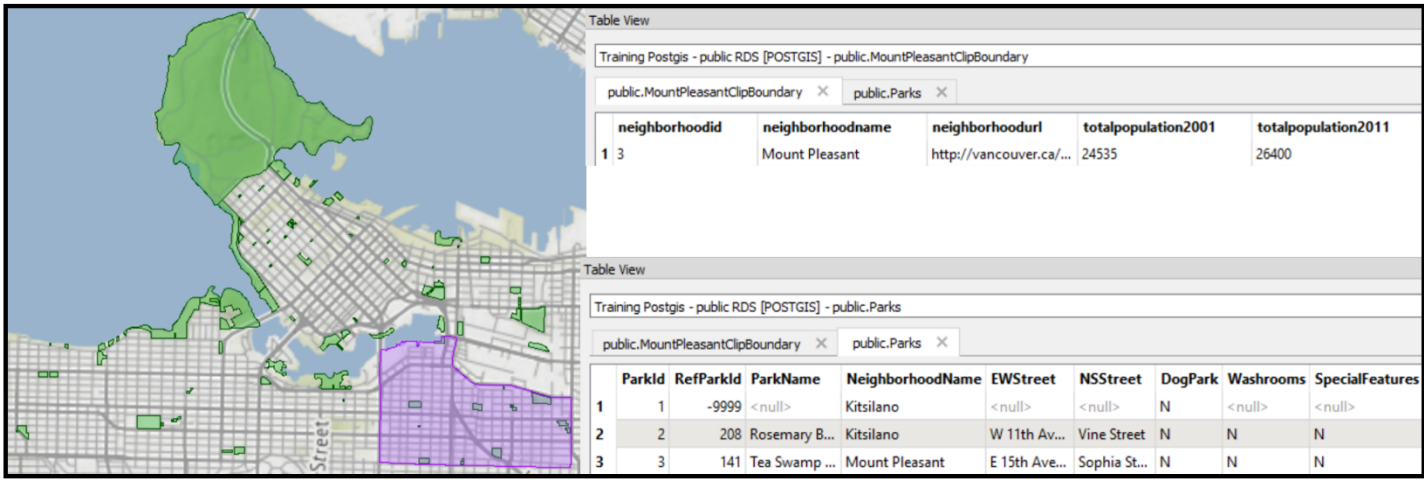
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector, the output will automatically appear in the FME Data Inspector.



Results

You have now successfully merged the Parks polygon dataset with the MountPleasantClipBoundary polygon dataset using the spatial predicate "Filter Intersects Candidate".

Input


Street
Table View
Training Postgis - public RDS [POSTGIS] - public.MountPleasantClipBoundary
public.MountPleasantClipBoundary
public.Parks
neighborhoodid | neighborhoodname | neighborhoodurl | totalpopulation2001 | totalpopulation2011
1 | 3 | Mount Pleasant | http://vancouver.ca/... | 24535 | 26400
Table View
Training Postgis - public RDS [POSTGIS] - public.Parks
public.MountPleasantClipBoundary
public.Parks
ParkId | RefParkId | ParkName | NeighborhoodName | EWStreet | NSStreet | DogPark | Washrooms | SpecialFeatures
1 | 1 | -9999 | <null> | Kitsilano | <null> | <null> | N | <null> | <null>
2 | 2 | 208 | Rosemary B... | Kitsilano | W 11th Av... | Vine Street | N | N | N
3 | 3 | 141 | Tea Swamp ... | Mount Pleasant | E 15th Ave... | Sophia St... | N | N | N

Output

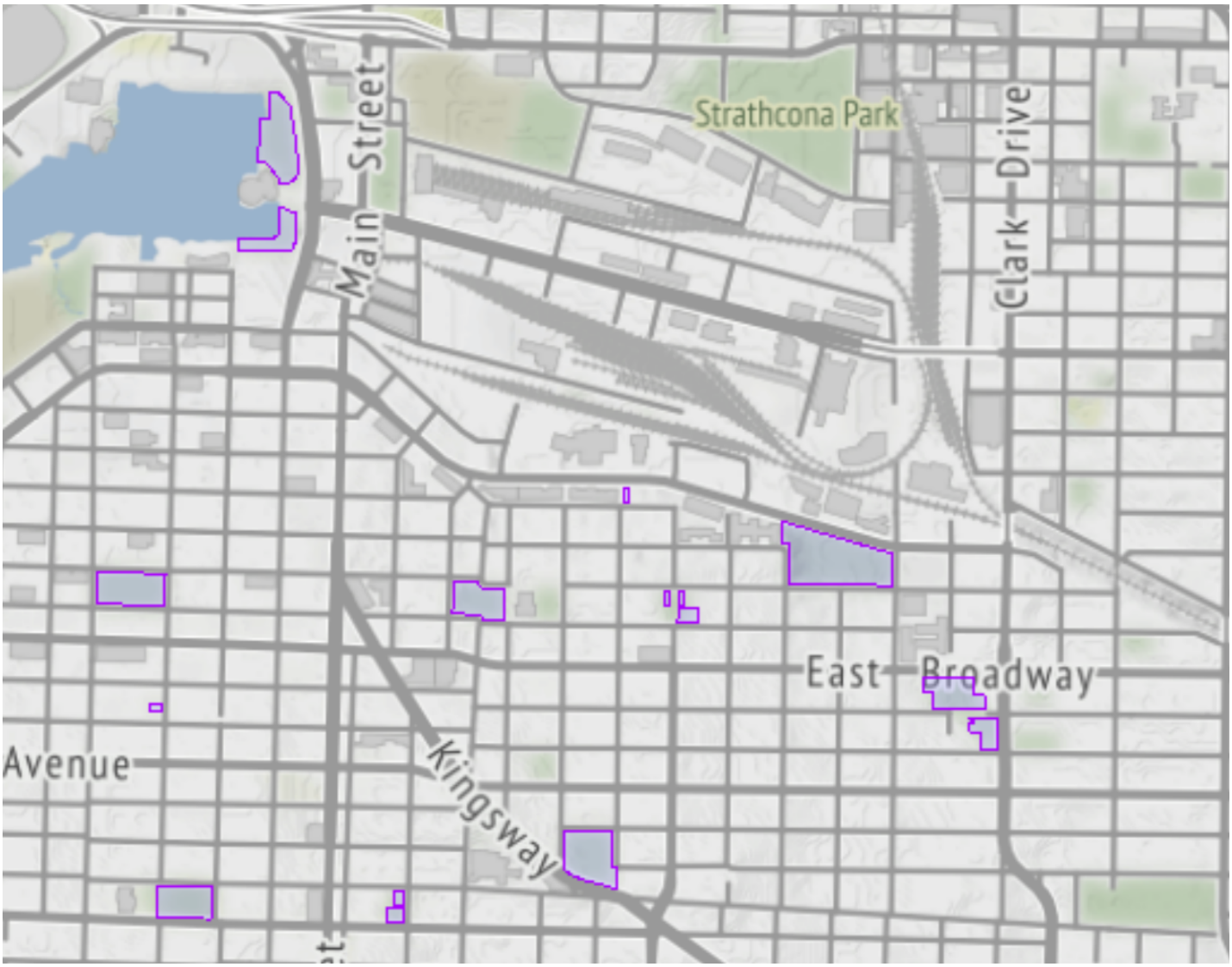

Strathcona Park
Main Street
Clark Drive
East Broadway
Avenue
Kingsway

Table View

inspector [FFS] - Inspector

	neigh	neighborhoodn	neighborhoodurl	totalpopu	totalpopula	ParkId	RefParl	ParkName	NeighborhoodName	EWStreet	NSStreet	DogPark	Washrooms	Speciall	_predicate
1	3	Mount Pleasant	http://vancouv...	24535	26400	3	141	Tea Swamp Park	Mount Pleasant	E 15th Avenue	Sophia Street	N	N	N	INTERSECTS
2	3	Mount Pleasant	http://vancouv...	24535	26400	8	-9999	<null>	Mount Pleasant	<null>	<null>	N	<null>	<null>	INTERSECTS
3	3	Mount Pleasant	http://vancouv...	24535	26400	9	15	Creekside Park	Mount Pleasant	Terminal Avenue	Quebec Street	N	N	Y	INTERSECTS
4	3	Mount Pleasant	http://vancouv...	24535	26400	10	134	China Creek So...	Mount Pleasant	E 10th Avenue	Clark Drive	N	N	N	INTERSECTS

Transformers

- Inspector- Sends features to the FME Data Inspector for display.
- SpatialFilter- Filters point, line, area, and text features based on spatial relationships.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Line in Polygon

Overview

In this exercise, you will use the SpatialFilter to set a spatial predicate that will determine which line features intersect polygon features. The SpatialFilter compares two sets of features to see if their spatial relationships meet selected test conditions. The features being tested (Candidate features) are identified as having Passed or Failed the test. For example, if you have a roads dataset (lines), and wanted to extract all the roads that passed through parks (polygons), you would direct the roads into the Candidate input port, and the parks into the Filter input port.

Downloads

line-in-polygon.fmwt

mountpleasantclipboundary.zip

streets.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you will merge MountPleasantClipBoundary polygon features with linear Street features and join attributes based on the spatial predicate, “Filter Intersect Candidate”. In this case, the Neighborhoods dataset is the filter because you want to extract all of the Streets that intersect the MountPleasantClipBoundary and merge the attributes of both features.

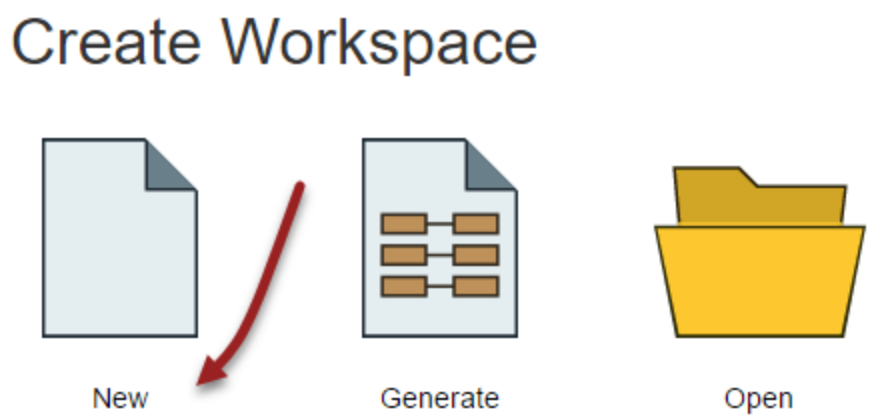
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

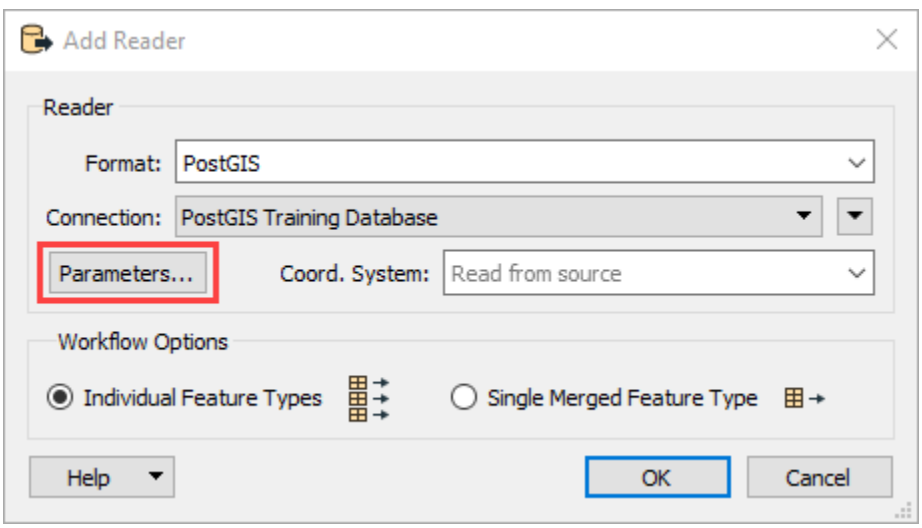
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection, select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the MountPleasantClipBoundary and Streets tables from the Table List.



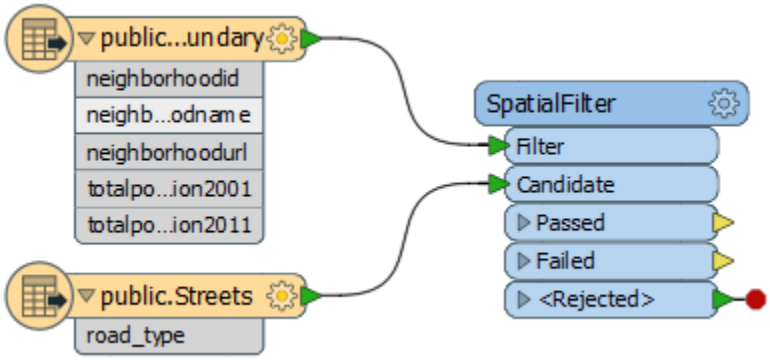
4. Add a Spatial Filter

- Similar to adding a reader, type “SpatialFilter” to bring up the list of FME Transformers. Select the SpatialFilter from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.

5. Connect the Transformers Inputs

- Connect the MountPleasantClipBoundary Reader to the SpatialFilter:Filter port. Similarly, connect the

Streets Reader to the SpatialFilter:Candidate port. Your connections should look like the screenshot below:



In this case, you do not have to modify any of the SpatialFilter's parameters as the default is spatial predicate is "Filter Intersects Candidate" and Merge Attributes is already selected. For more information on the how the SpatialFilter works, please refer to the SpatialFilter Documentation.

6. Add an Inspector

- Next, add an Inspector and connect it to the SpatialFilter:Passed port.

This will display the Streets that intersect the MountPleasantClipBoundary polygons in the FME Data Inspector after the translation has run.

7. Add an Esri Geodatabase (File Geodb Open API) Writer

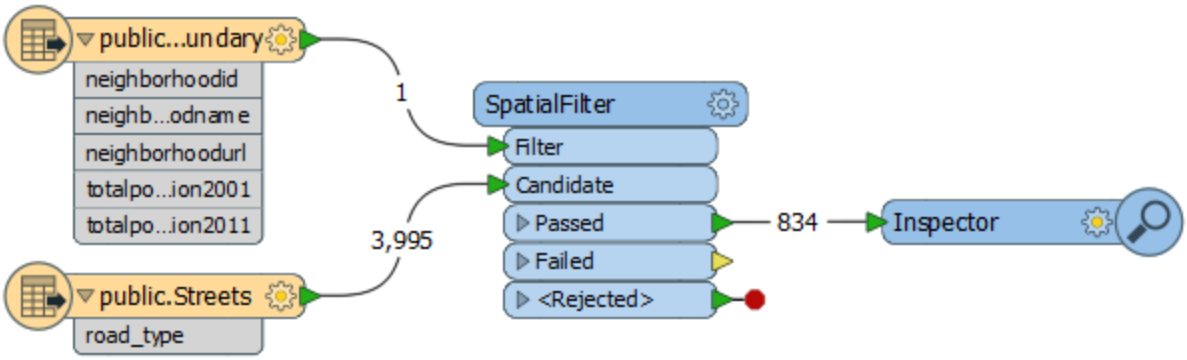
- Similar to adding a reader, type "Geodatabase" then select the Esri Geodatabase (File Geodb Open API) format from the list of Writers.
- Specify the File Geodatabase Folder by clicking on the Dataset ellipsis, creating a new folder or using an existing folder with the file extension .gdb, then selecting the .gdb folder (i.e. C:\Users\Documents\MountPleasant\Streets.gdb). After specifying the file geodatabase folder location, set the Table Definition to Automatic and click OK. To find out more about working with Esri Geodatabases see the Getting Started with Geodatabases article.

An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected, to learn more about Attribute Definitions, see the About Writer Feature Types Documentation.

- Lastly, set the Feature Class Name in the Writer Parameters and Geometry in the dialog that appears. In this case, the SpatialFilter will output line features, so you should set the geometry type to: geodb_polyline
- After setting the Feature Class Name and specifying the geometry, click OK to add the Esri Geodatabase writer to the canvas.
- Once the writer has been added, connect it to SpatialFilter:Passed port.

8. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar. Since you attached an Inspector, the output will automatically appear in the FME Data Inspector.



Results

Input

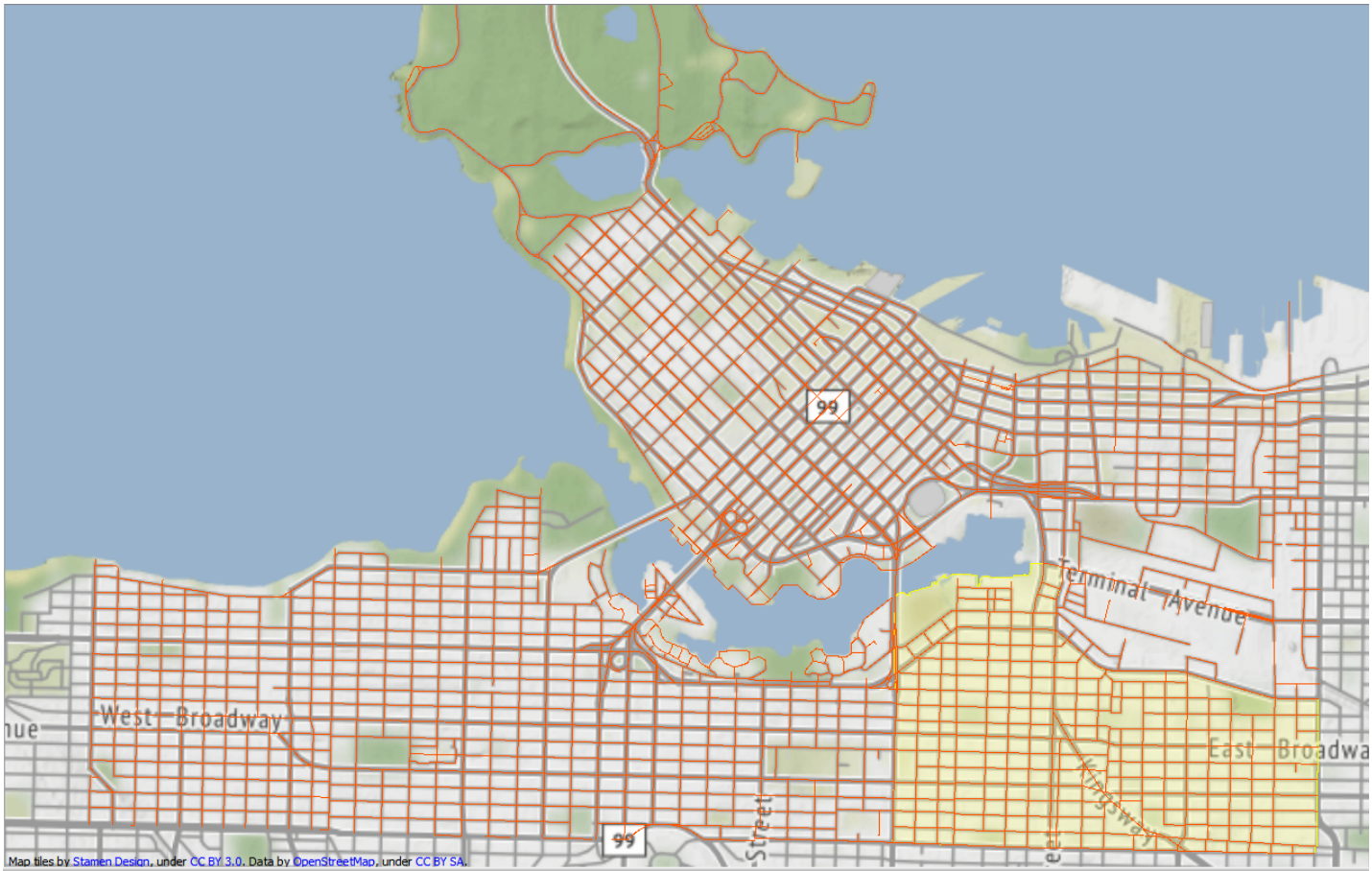


Table View

fmedata [POSTGIS] - public.MountPleasantClipBoundary

public.MountPleasantClipBoundary | public.Streets

	neighborho	neighbor00	neighbor01	totalpopul	totalpop00
1	3	Mount Pleasant	http://vancouv...	24535	26400

Table View

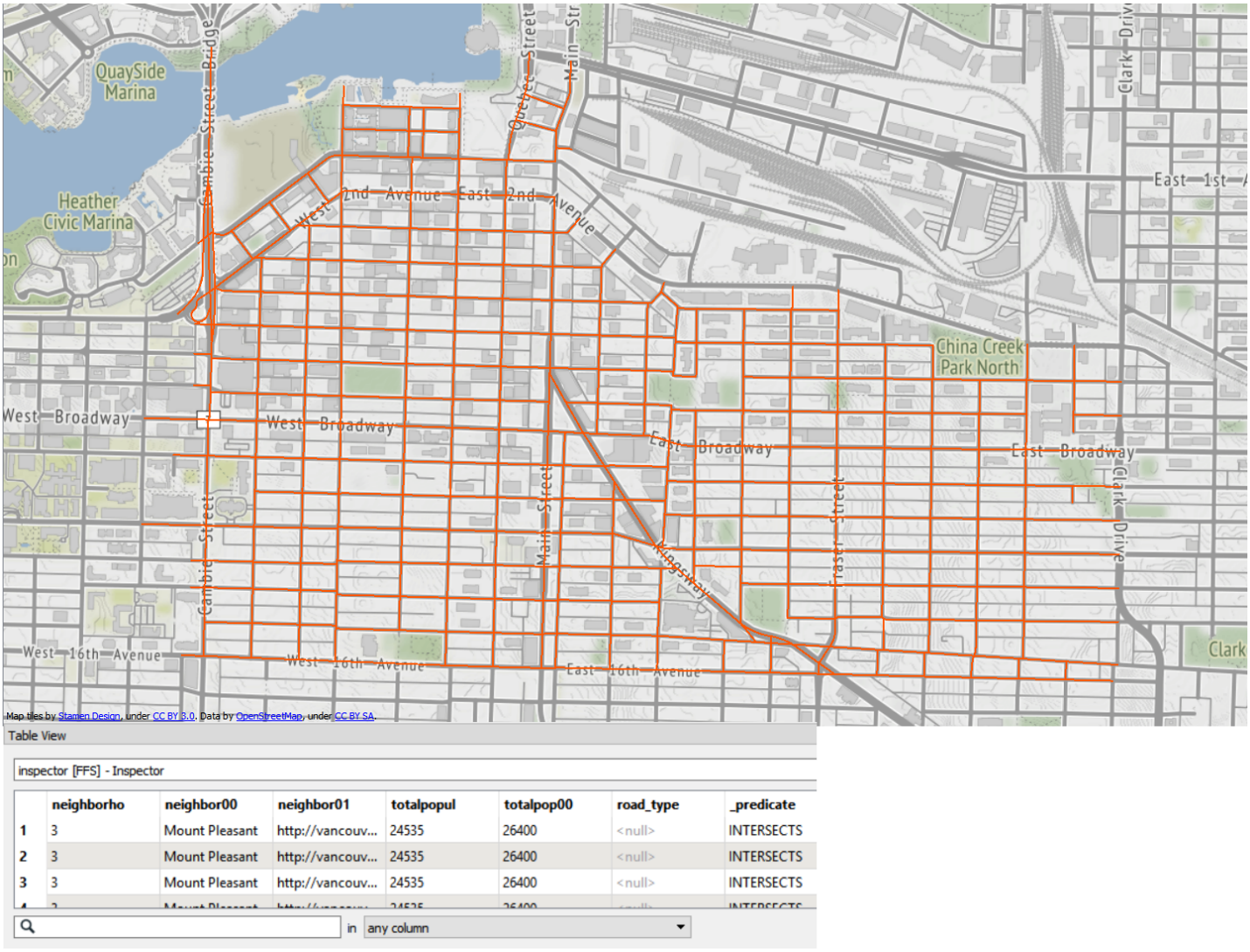
PostGIS Training Database [POSTGIS] - public.Streets

public.MountPleasantClipBoundary | public.Streets

	road_type
1	<null>
2	<null>

in any column

Output



	neighborho	neighbor00	neighbor01	totalpopul	totalpop00	road_type	_predicate
1	3	Mount Pleasant	http://vancouv...	24535	26400	<null>	INTERSECTS
2	3	Mount Pleasant	http://vancouv...	24535	26400	<null>	INTERSECTS
3	3	Mount Pleasant	http://vancouv...	24535	26400	<null>	INTERSECTS

Transformers

- Inspector- Sends features to the FME Data Inspector for display.
- SpatialFilter- Filters point, line, area, and text features based on spatial relationships.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Simple Examples Using the RasterExpressionEvaluator Transformer

Introduction

The RasterExpressionEvaluator is a transformer that evaluates expressions on each cell in a raster, such as algebraic operations or conditional statements. This article will demonstrate commonly used raster transformations using the RasterExpressionEvaluator.

Downloads

RasterExpressionEvaluator-Workspaces.zip *(All of the workspaces)*

Instructions

Example 1: Inverting Colors

RGBColorInverter.fmwt

This example demonstrates how to invert images with the following band interpretation list and expression:

Interpretation	Expression
Red8	255-A[0]
Green8	255-A[1]
Blue8	255-A[2]

The images show Heritage Mountain Elementary School and the surrounding area in Google Earth

Example 2: Brightness Correction

BrightnessCorrection.fmwt

If an image is too dark, we can boost brightness with the following simple expression:

Interpretation	Expression
Red8	A[0]*1.5
Green8	A[1]*1.5
Blue8	A[2]*1.5

Port Moody, Greater Vancouver

Note that this method should be used cautiously - if there are bright areas in the image, they will reach or exceed the maximum value (255 in my example), which will make them purely white (255, 255, 255 in case of RGB24), and the information in those areas will be lost.

Example 3: Color Correction

ColorCorrection.fmwt

If one of the colors on an image seems to dominate over the others, we can change only one band (or apply different coefficients to different bands):

Interpretation	Expression
Red8	A[0]
Green8	A[1]/1.2
Blue8	A[2]

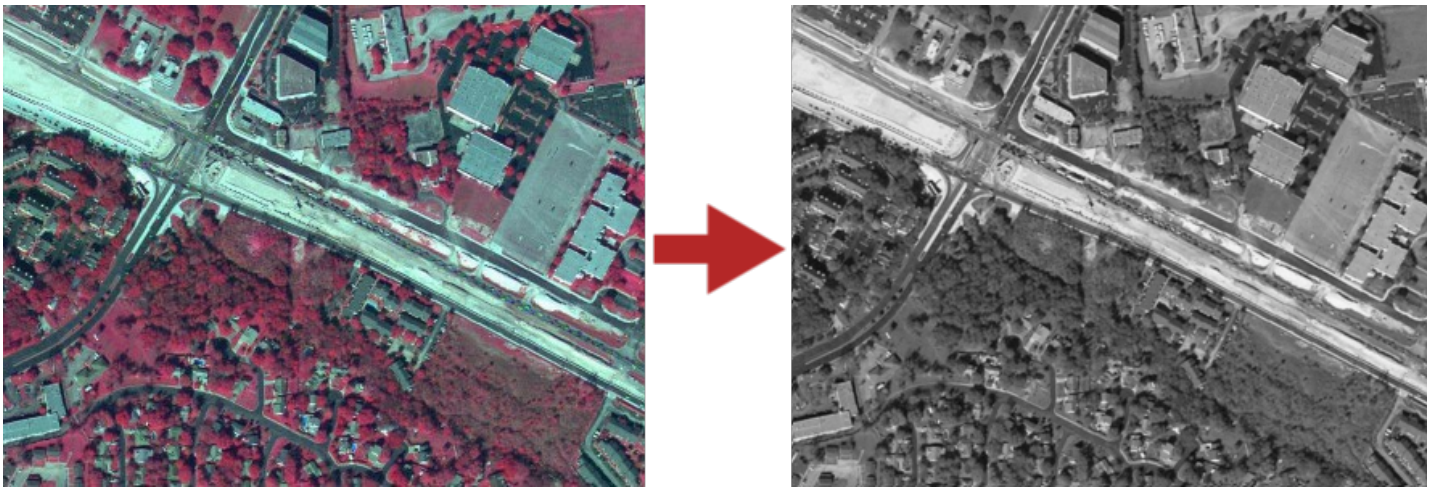
Port Moody, Greater Vancouver

Example 4: Color to Grayscale Conversion

RGB2Grayscale.fmwt

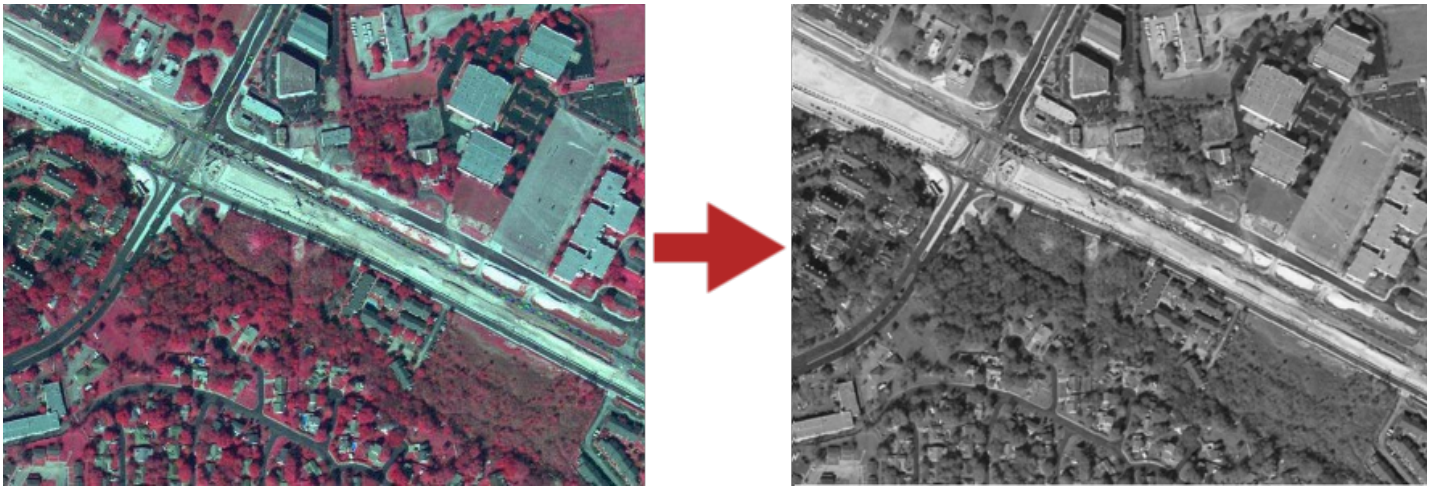
Most sources suggest the following formula for a color to grayscale conversion (and many sources also mention that depending on the image it may vary). There are two commonly used formulas the first formula gives better contrast:

Interpretation	Expression
Grey8	0.2989*A[0] + 0.5870*A[1] + 0.1140*A[2]



Vancouver, British Columbia

Interpretation	Expression
Grey8	0.3333*A[0] + 0.3333*A[1] + 0.3333*A[2]



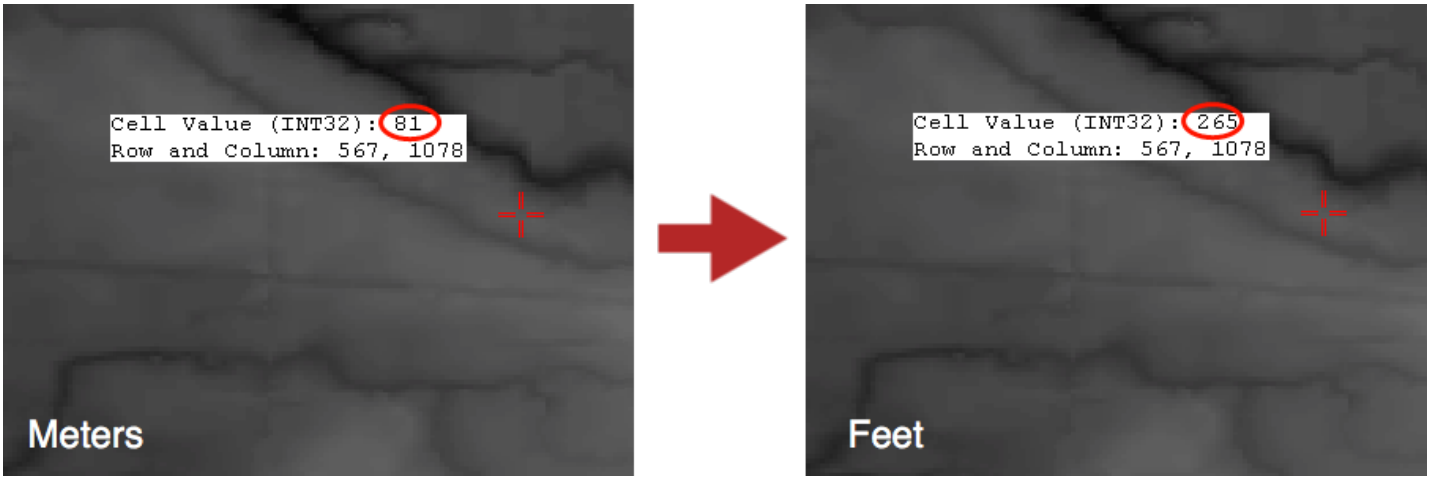
Vancouver, British Columbia

Example 5: Unit Conversion

UnitConversion.fmwt

Conversion between all kinds of units is a very natural operation for RasterExpressionEvaluator. In this example we take a numeric raster representing a DEM in meters and convert the units into feet:

Interpretation	Expression
Int32	A[0]/0.3048



North Vancouver, Greater Vancouver

Raster Calculations and Raster Palettes

Start: Tutorial: Reading and Transforming Satellite Imagery in FME | Next: Raster bit Masks: Making Sense of the Numbers

Introduction

This article is part of the Tutorial: Reading and Transforming Satellite Imagery in FME

In this article we will be calculating the Normalized Difference Vegetation Index (NDVI). In the process, we will learn how to build conditions in the RasterExpressionEvaluator as well as how to create rasters with palettes. NDVI is based on the ability of plants to consume visible light radiation for photosynthesis (mostly in blue and red parts of the spectrum, this is why we see our plants mostly green, this color is mostly reflected). Near-infrared light, on the other hand, cannot cause the proper reaction in cells, and hence, is being reflected (otherwise it will overheat the plants). The more infrared light is reflected and the more red light is consumed, the healthier the plant is.

This article assumes you have completed the previous article in the tutorial: Making RGB images with Sentinel data. If you have not, please complete the exercise in the first part of the article to download the appropriate images.

Downloads

Completed workspace (without source data): create-palettes.fmw

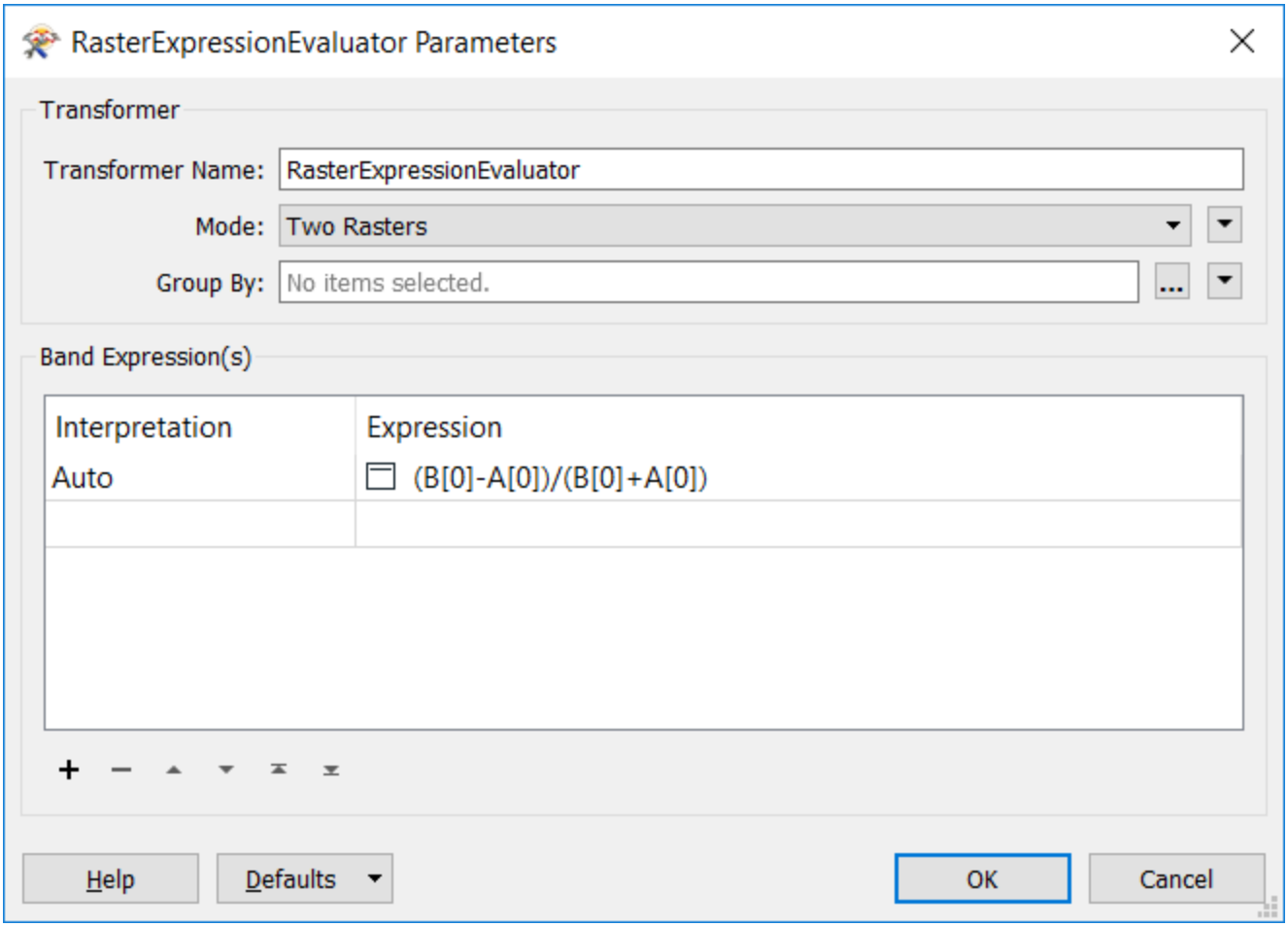
NDVI Calculator

1) Create a new workspace and call it NDVICalculator.

2) Add a JPEG 2000 reader and select the preview images of the red and NIR bands (B04 and B08). Under Parameters, be sure the Feature Type Name is set to 'From File Name'

3) First add in a RasterExpressionEvaluator, set Mode to 'Two Rasters'. The Red band should go into the 'A' port and the NIR should go into the 'B' port. Use Auto for interpretation and for expression, use:

```
(B[0]-A[0])/(B[0]+A[0])
```

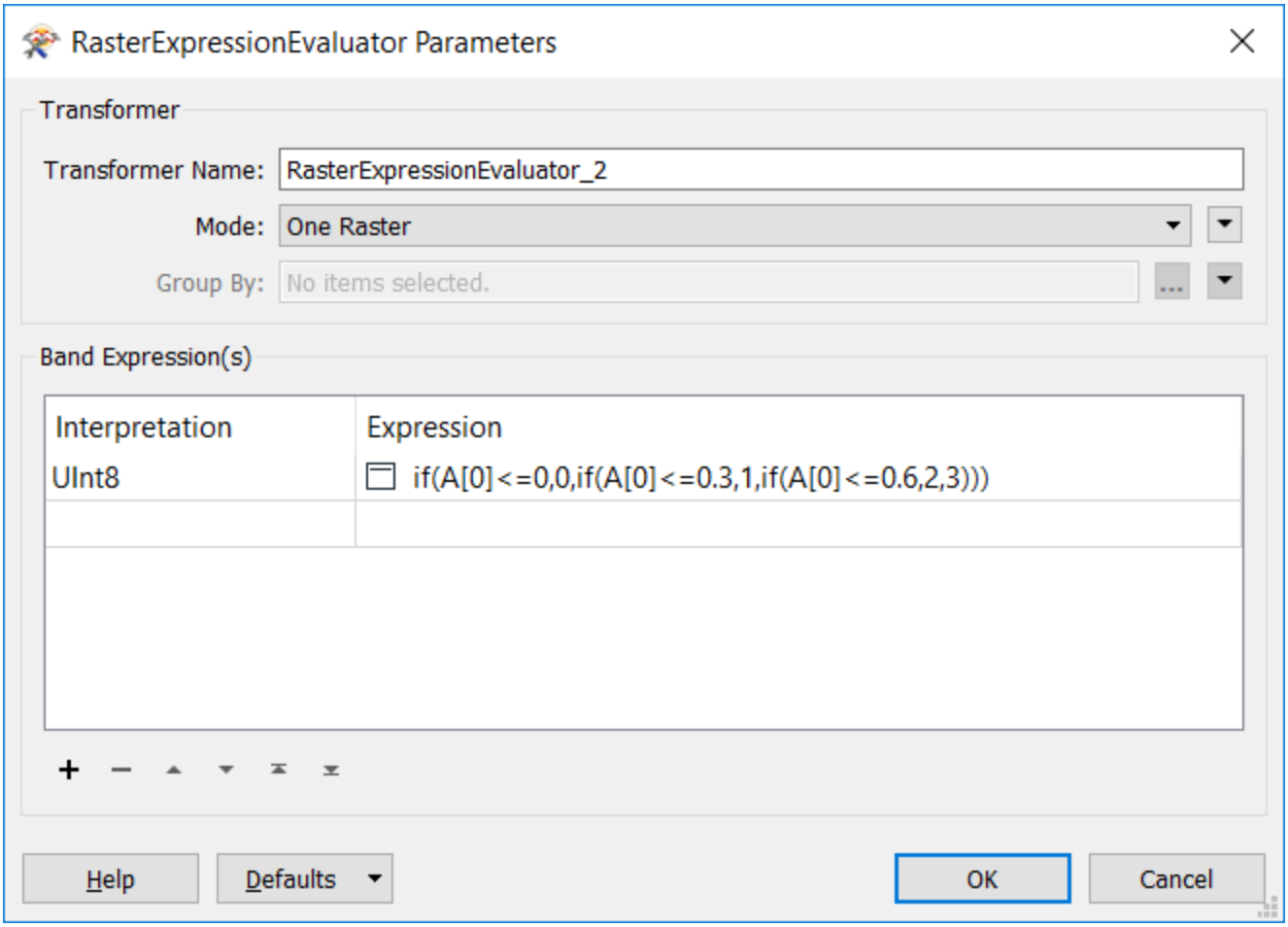


RasterExpressionEvaluator: *This expression calculates the NDVI which is a ratio of reflected NIR to reflected Red light.*

4) Run, inspect results, notice now that the band ranges of the resulting pixels falls between -1 and 1 and our interpretation has changed to REAL64.

5) Add another RasterExpressionEvaluator. In the evaluator set the interpretation to UINT8 and copy and paste the following for the expression:

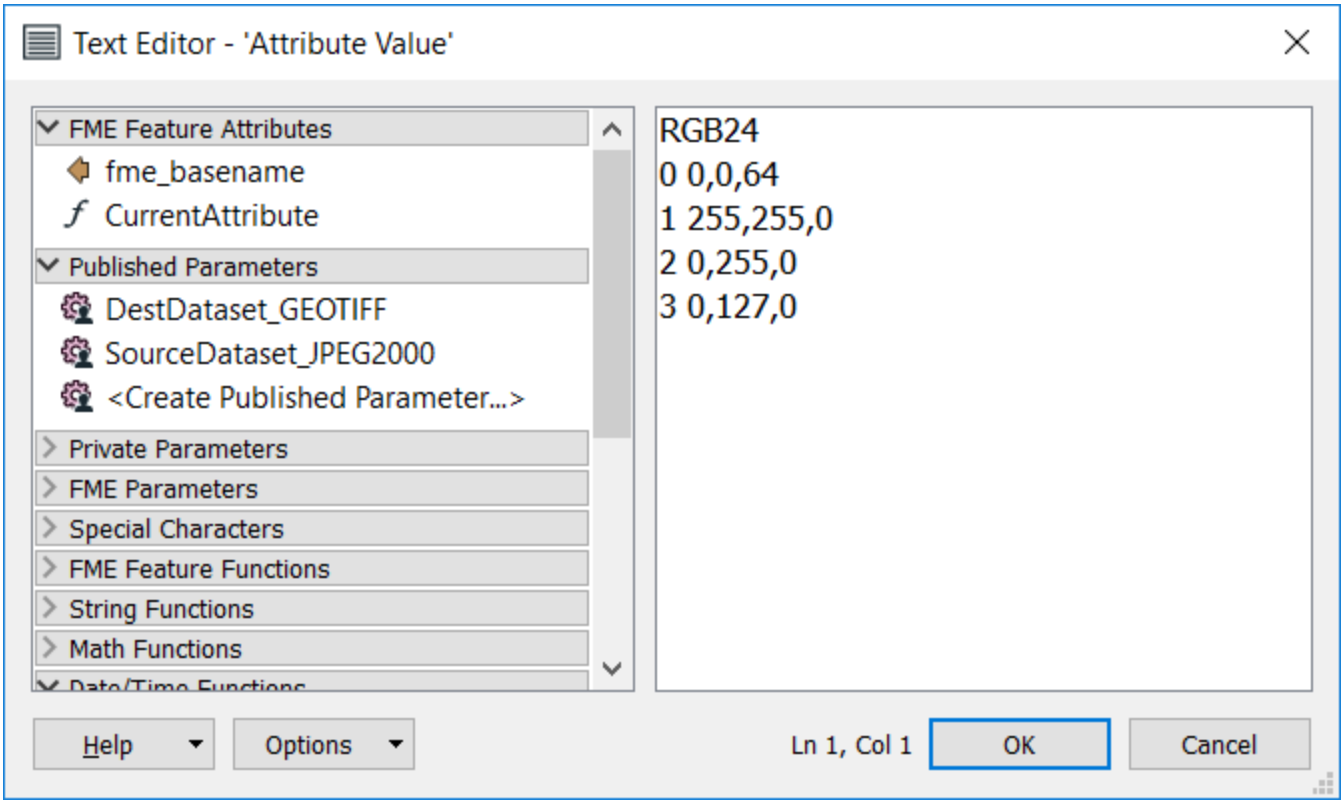
```
if(A[0]<=0,0,if(A[0]<=0.3,1,if(A[0]<=0.6,2,3)))
```



RasterExpressionEvaluator_2: *here we use a series of nested 'if' functions to build a conditional expression to begin building our palette based on our calculated NDVI. This will keep only four raster values - 0,1,2 and 3*

6) Add an AttributeCreator and use the text editor to create a _palette attribute as is shown below. The colors below are, navy blue, yellow, bright green and dark green but you can set the RGB triads to what you think is appropriate:

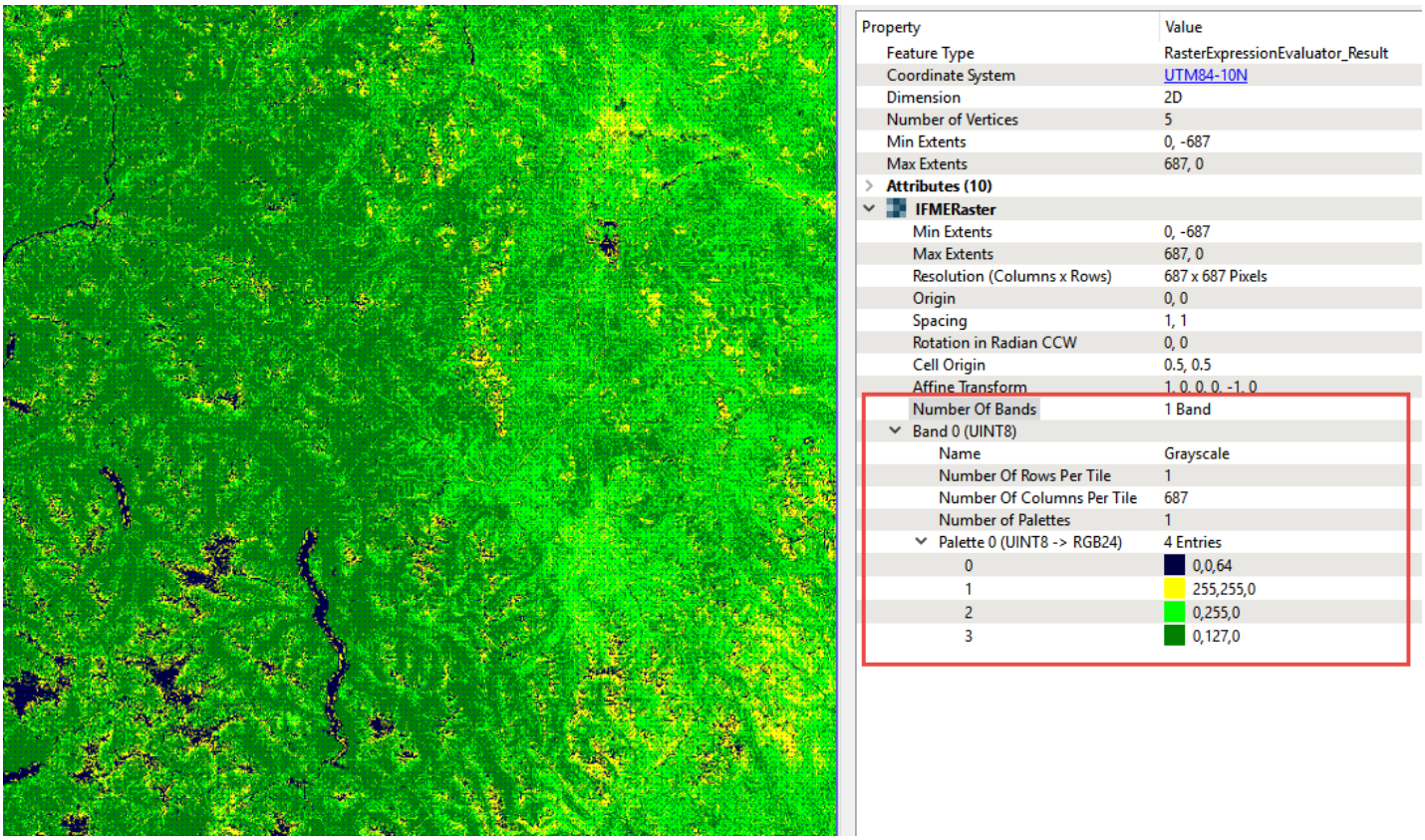
```
RGB24
0 0,0,64
1 255,255,0
2 0,255,0
3 0,127,0
```



_palette Attribute: Here we are defining the interpretation of the palette as RGB24 and are assigning each of the integers from 0 - 3 a RGB color.

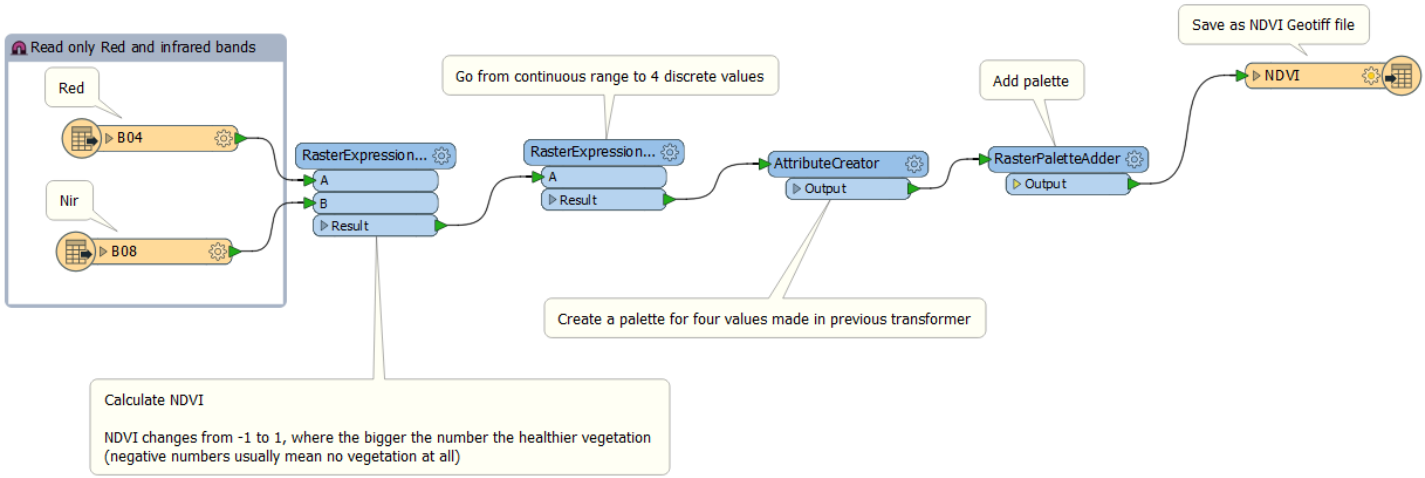
7) Add RasterPaletteAdder and add _palette to the raster

8) Run and inspect results, it should look something like this:



Final Image: *You will likely have a different image to what is seen here, but make sure you take note of the palette information displayed in the Feature Information window*

The final workspace should look like this:



Final Workspace: *Do you have something different? Download the final workspace at the top of this article to compare*

Raster Neighborhood Functions

Overview

The highly anticipated RasterConvolver is finally here! New to FME 2018.1, the RasterConvolver is a transformer that applies a convolution filter to raster data. Convolution filters are applied using a kernel which is specified by a matrix of weights for the neighboring values. Filtering can be used to smooth, sharpen, enhance edges and other raster image manipulation operations.

Downloads

neighborhood-functions.fmwt

bcvanc15-o7.zip

Exercise

In this scenario, you are interested in applying a convolution filter in order to sharpen a GeoTIFF image. For more information on Raster processing using the RasterConvolver, check out the Edge Detection with the RasterConvolver article or the Raster Slope Calculations as a RasterConvolver Example article.

Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.

3. Add a GeoTIFF Reader

- Start typing “GeoTIFF” without anything selected on the canvas, then select the GeoTIFF format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis and navigate to the folder containing BCVANC15_O7.tif. Select the BCVANC15_O7.tif file and click Open then OK to add the GeoTIFF reader to the canvas.

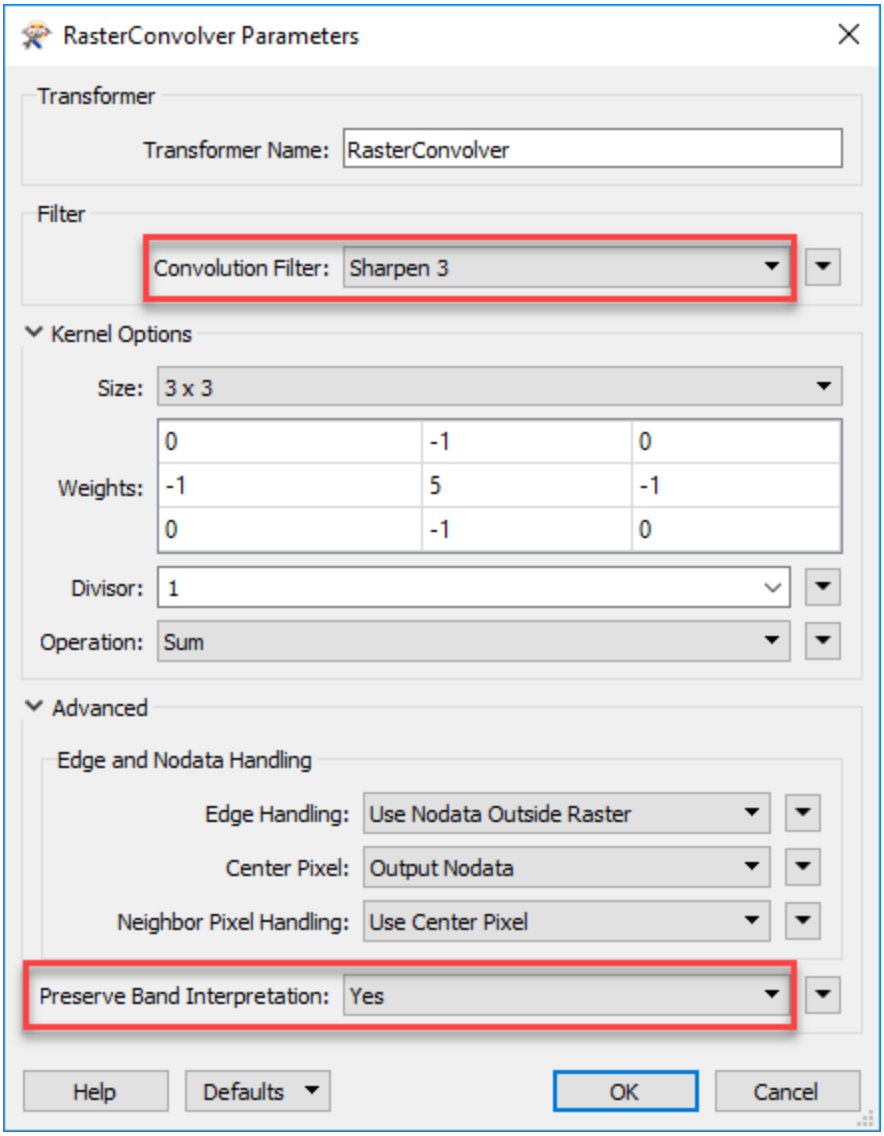
4. Add a RasterConvolver

The RasterConvolver accepts input features containing a raster geometry and outputs the features after applying a convolution filter to each band.

- Add a RasterConvolver to the canvas by typing “RasterConvolver” to bring up the list of FME Transformers in the Quick Add Search. Select the RasterConvolver from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the transformer has been added, connect the GeoTIFF Reader to the ConvolutionFilter.

5. Set the Convolution Filter Type

- Once the RasterConvolver has been added, double-click the RasterConvolver or click on the gear icon to open the transformer parameters dialog.
- In the Parameters dialog, change the Convolution Filter to Sharpen 3.
- Expand the Advanced section and set the Preserve Band Interpretation to Yes.



Note: By default, the preserve band interpretation is set to No - leaving this as No would convert the Red, Green, and Blue bands of the GeoTIFF to Real64 (i.e. convert to a grayscale image).

6. Add an Inspector

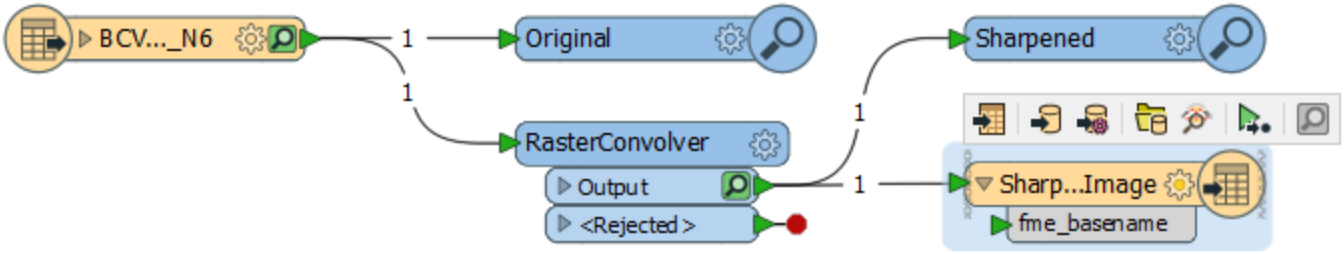
- Add an Inspector to the canvas and connect it to the RasterConvolver:Output port. Once you have made the connection, open the Inspector parameters and name it Sharpened.
- Similarly, attach an inspector to the GeoTIFF reader and name it Original. This will automatically open both the original and sharpened raster dataset in the FME Data Inspector after the translation has run. Inspecting both datasets will allow you to toggle between the original image and the sharpened image to see how the RasterConvolver's Sharpen 3 convolution filter processed the image.

7. Add an ECW Writer

- Similar to adding a reader, type "ECW" then select the ECW format from the list of Writers.
- Specify a folder for the ECW file by clicking on the Dataset ellipsis. After specifying a folder location, set the Raster File Definition to Automatic then click OK.
- Next, set the Raster File Name in the Feature Type Parameters dialog then click OK to add the ECW writer to the canvas. Here, you can also specify the Raster Compression by percentage and have the option to generate a tab file. To learn more about TAB files see the Raster Documentation.
- Lastly, connect the ECW Writer to the RasterConvolver:Output port.

8. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.

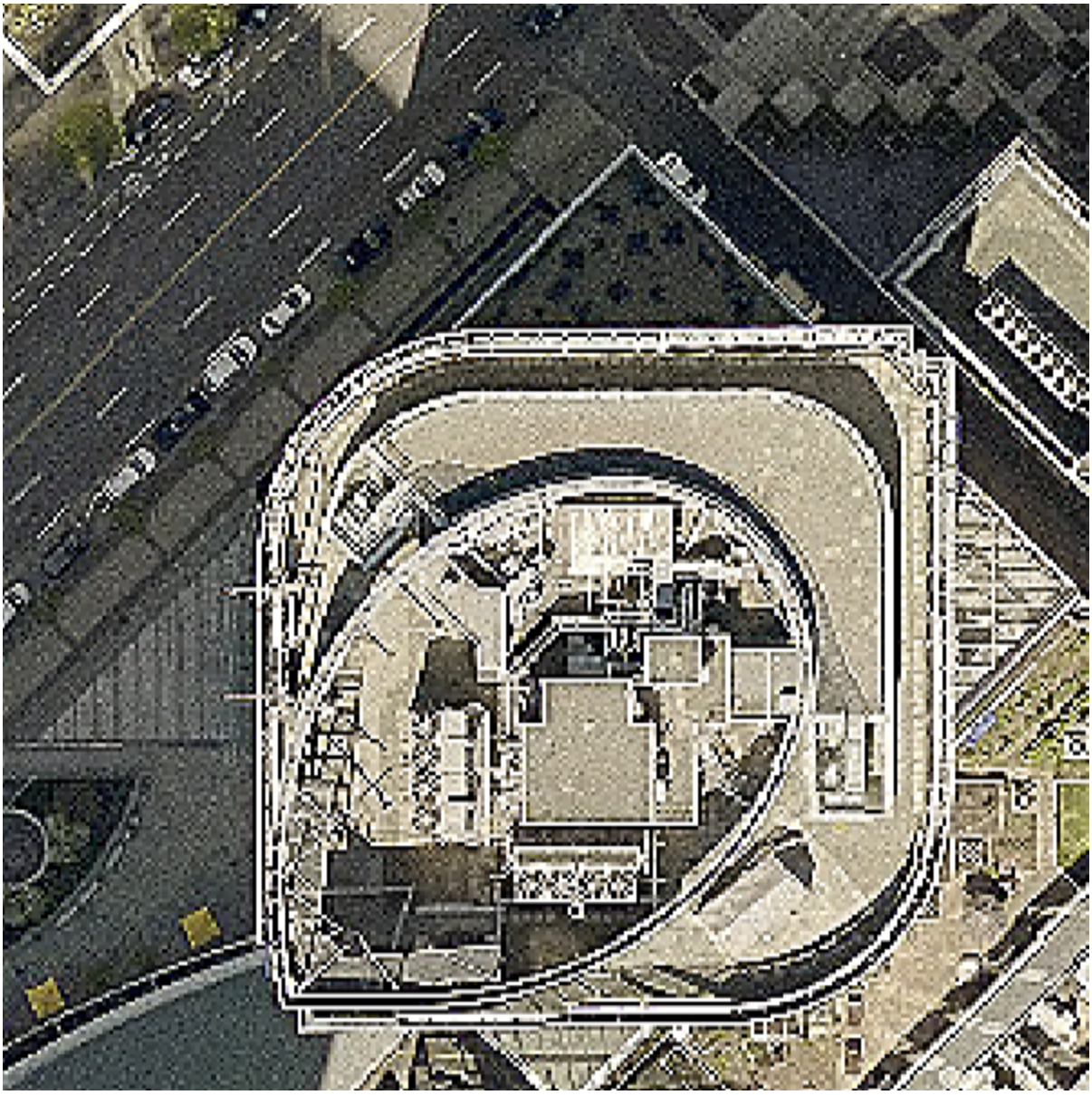


- After the translation has run, toggle between the Sharpened and Original image to see what the Sharpen 3 convolution filter did. Try picking another filter from the RasterConvolver's predefined convolution filters or enter your own values by setting the Convolution Filter to User Defined and entering Kernel Weights.

Results

Input

Output



Transformers

- Inspector - Sends features to the FME Data Inspector for display.
- RasterConvolver - Accepts input features containing a raster geometry and outputs the features after applying a convolution filter to all the bands.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Calculating Area, Perimeter, and Lengths

Overview

Area, perimeter, and length can be calculated for features using transformers such as the AreaCalculator and LengthCalculator in FME. In this tutorial, you will learn how to calculate the area and perimeter of polygon features that are stored in the PostGIS Parks table, calculate polygon perimeters, and calculate the length of bike paths in the Vancouver area.

Downloads

area-of-polygon.fmwt

perimeter-of-polygon.fmwt

length-of-line.fmwt

bike-paths.zip

parksgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise 1 - *Calculating the Area of Polygon Features*

In this exercise, you will learn how to calculate the area of park polygons using the AreaCalculator. Determining the area of each park will help you understand the distribution of green space throughout the city. For more information on calculating statistics using polygon areas, see the Desktop Basic Course Manual.

Instructions

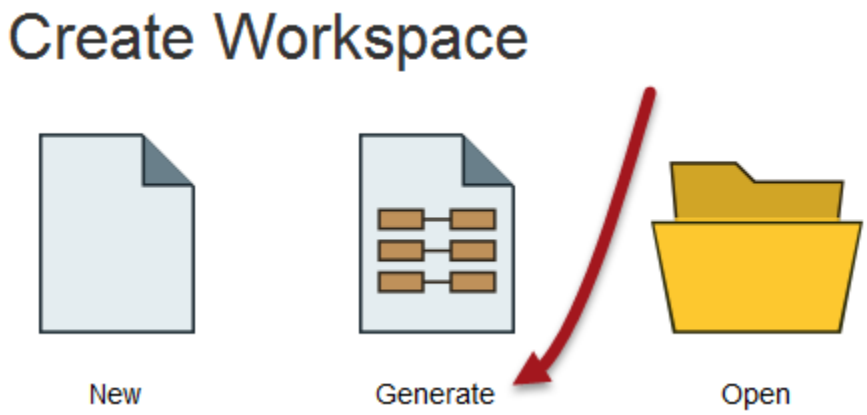
1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Select Generate Workspace

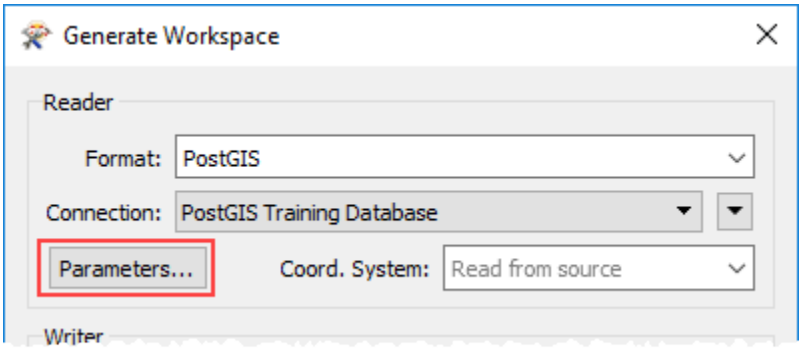
- In the Create Workspace part of the Start page, select the option to Generate a workspace.

Note: the shortcut to generate a workspace is Ctrl+G



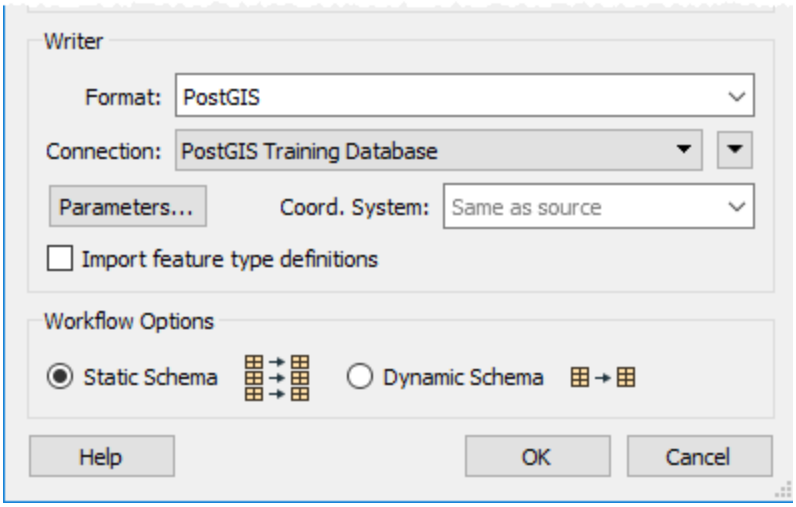
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection, select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Parks table from the Table List.



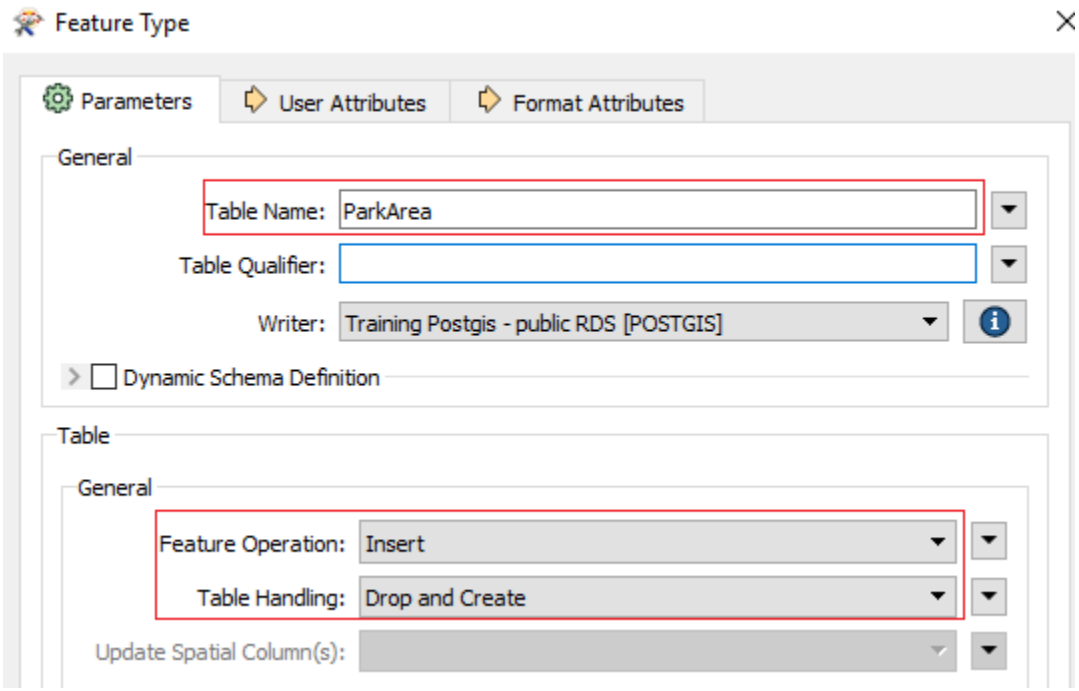
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

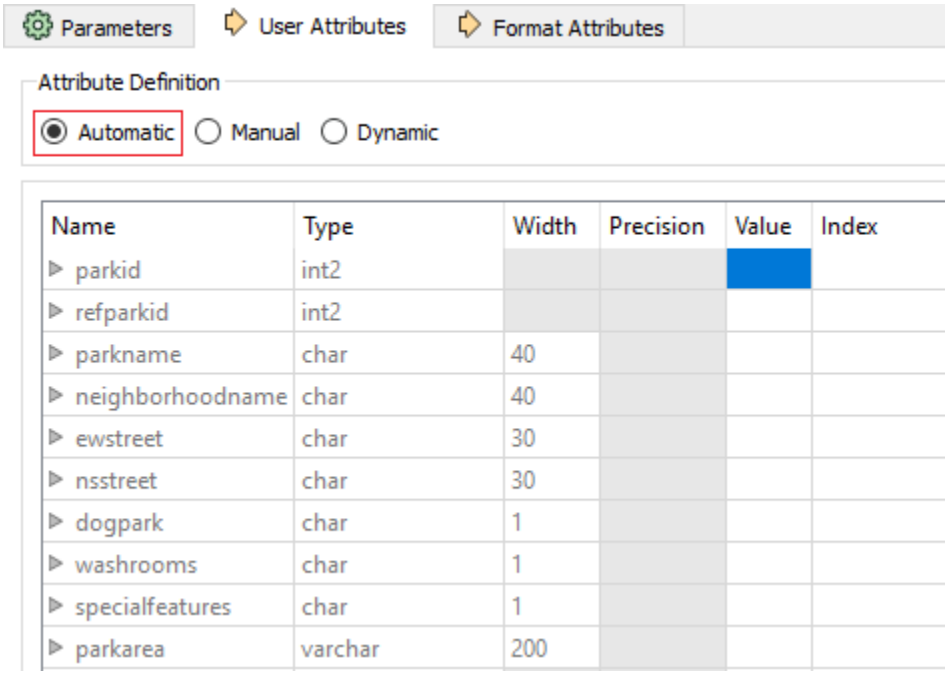
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to: ParkArea
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the [Writer Feature Type Parameters](#) Documentation.

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.
- Click OK to accept the changes and close the Writer Parameters dialog.



Name	Type	Width	Precision	Value	Index
parkid	int2				
refparkid	int2				
parkname	char	40			
neighborhoodname	char	40			
ewstreet	char	30			
nsstreet	char	30			
dogpark	char	1			
washrooms	char	1			
specialfeatures	char	1			
parkarea	varchar	200			

Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Documentation.

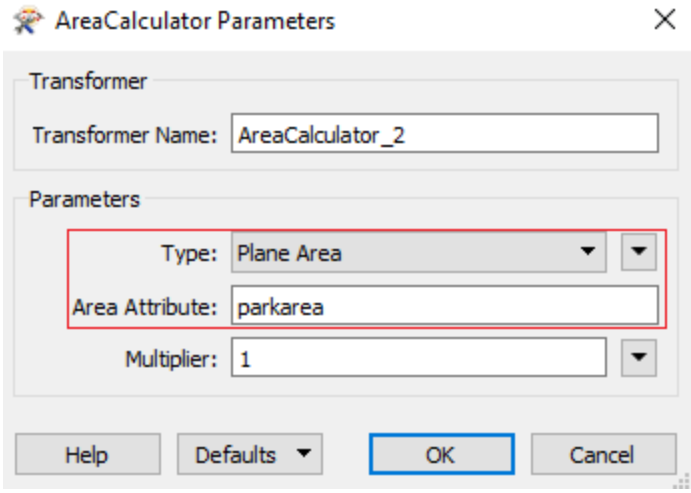
7. Add a FeatureHolder

- Add a FeatureHolder to the canvas by typing “FeatureHolder” to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the Parks Reader to the FeatureHolder.

The FeatureHolder reads and stores the features in the Parks table. Once all of the features have been stored, the existing table is closed and which allows FME to drop the existing table and create the new table.

8. Calculate Area

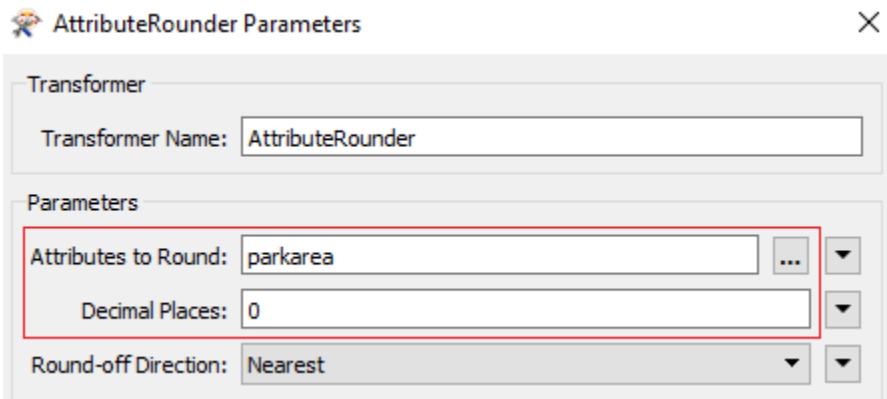
- Add an AreaCalculator after the FeatureHolder and open the transformer parameters. Set the Area Attribute to “parkarea” - this will add a new column to the table and calculate the area of each polygon feature in the dataset.
- Leave all other parameters as default and click OK to accept the changes and close the AreaCalculator parameters.



Note: the area is calculated in square map units.

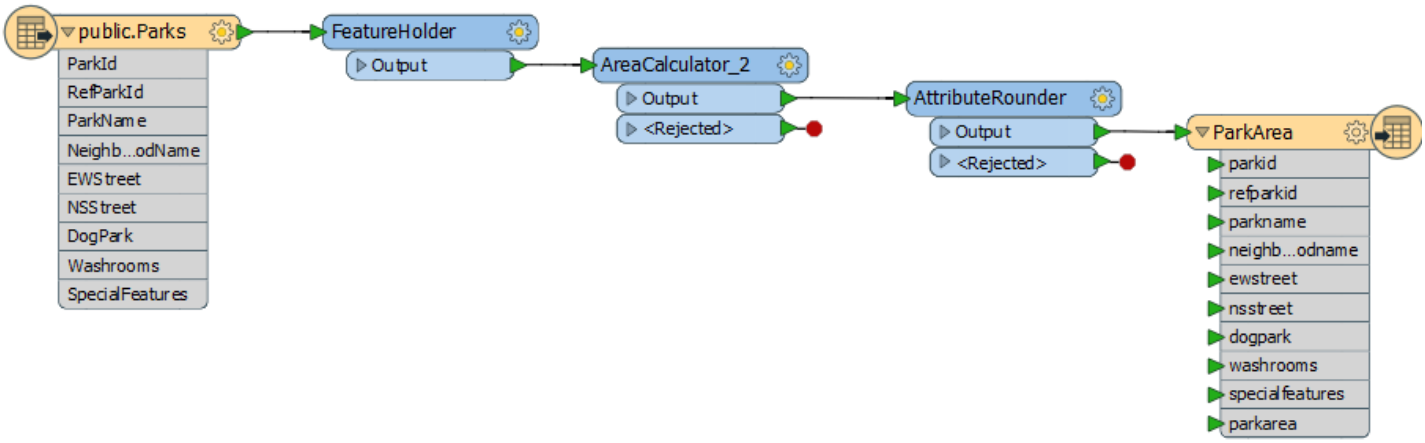
9. Round Area Values

- Next, add an AttributeRounder as you are only interested in finding the area of each park to the nearest map unit (which is meters in this case).
- Open the AttributeRounder parameters and set the Attributes to Round to “parkarea” and the Decimal Places to 0.



10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, your table will have a parkarea column with the area for each polygon feature. You can view the new table by selecting the writer and clicking the Inspector button in the shortcut menu that appears when the writer is selected. Alternatively, you can attach an Inspector to the output of the final transformer in the workspace if you wish to automatically open the workspace output the FME Inspector.

Results

Input

Table View

Table: Training Postgis - public RDS - 1 [POSTGIS] - public.Parks

	ParkId	RefParkId	ParkName	NeighborhoodName	EWStreet	NSStreet	DogPark	Washrooms	SpecialFeatures
1	1	-9999	<null>	Kitsilano	<null>	<null>	N	<null>	<null>
2	2	208	Rosemary Brow...	Kitsilano	W 11th Avenue	Vine Street	N	N	N
3	3	141	Tea Swamp Park	Mount Pleasant	E 15th Avenue	Sophia Street	N	N	N
4	4	-9999	<null>	Strathcona	<null>	<null>	N	<null>	<null>

Output

Table View

Training Postgis - public RDS [POSTGIS] - ParkArea

	parkid	refparkid	parkname	neighborhoodname	ewstreet	nsstreet	dogpark	washrooms	specialfeatures	parkarea
1	1	-9999	<null>	Kitsilano	<null>	<null>	N	<null>	<null>	448
2	2	208	Rosemary Brow...	Kitsilano	W 11th Avenue	Vine Street	N	N	N	1035
3	3	141	Tea Swamp Park	Mount Pleasant	E 15th Avenue	Sophia Street	N	N	N	2631
4	4	-9999	<null>	Strathcona	<null>	<null>	N	<null>	<null>	1985

Exercise 2 - *Calculating the Perimeter of Polygon Features*

In this exercise, you will learn how to calculate the perimeter of park polygons using the LengthCalculator.

Instructions

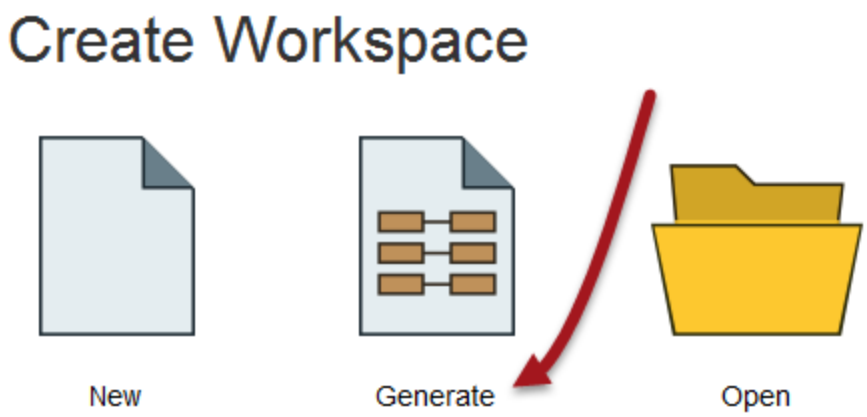
1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Select Generate Workspace

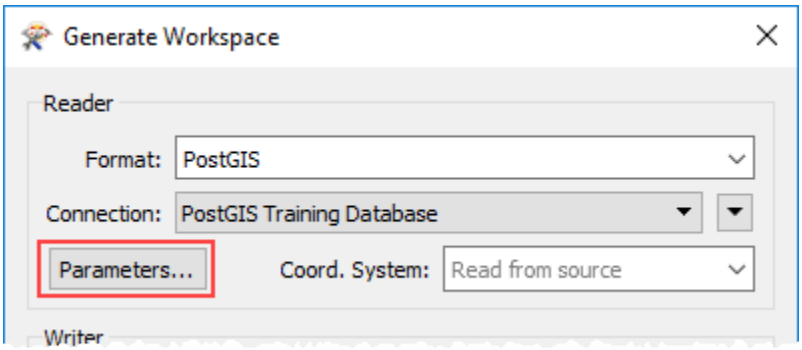
- In the Create Workspace part of the Start page, select the option to Generate a workspace.

Note: the shortcut to generate a workspace is Ctrl+G



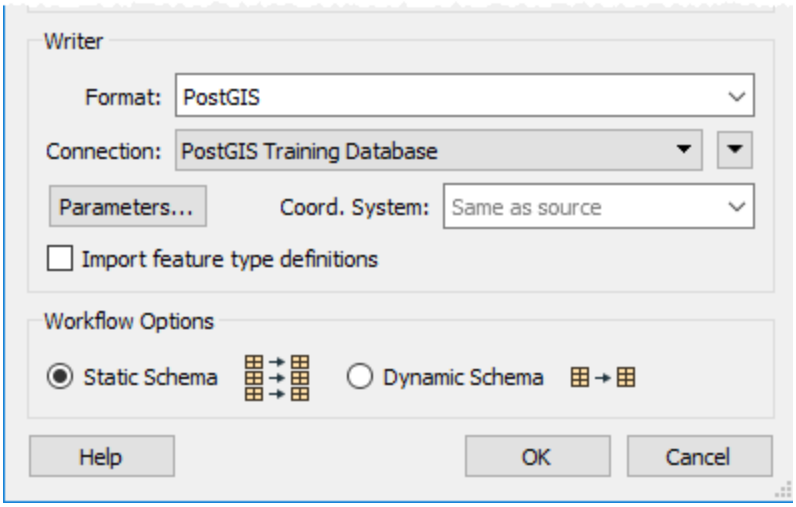
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection, select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Parks table from the Table List.



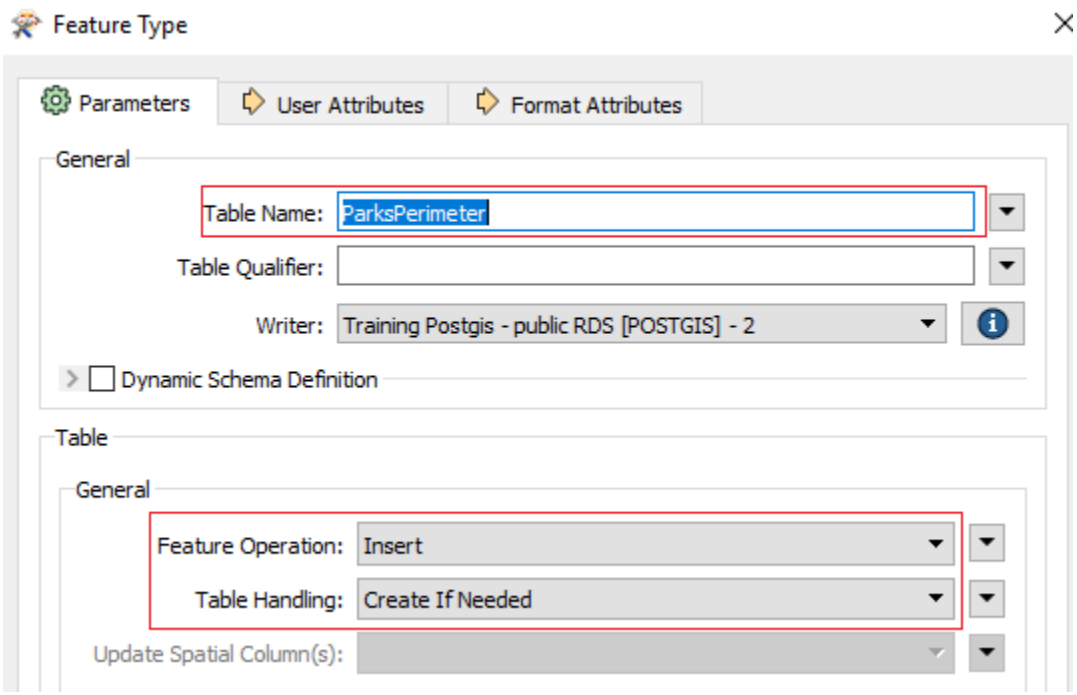
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

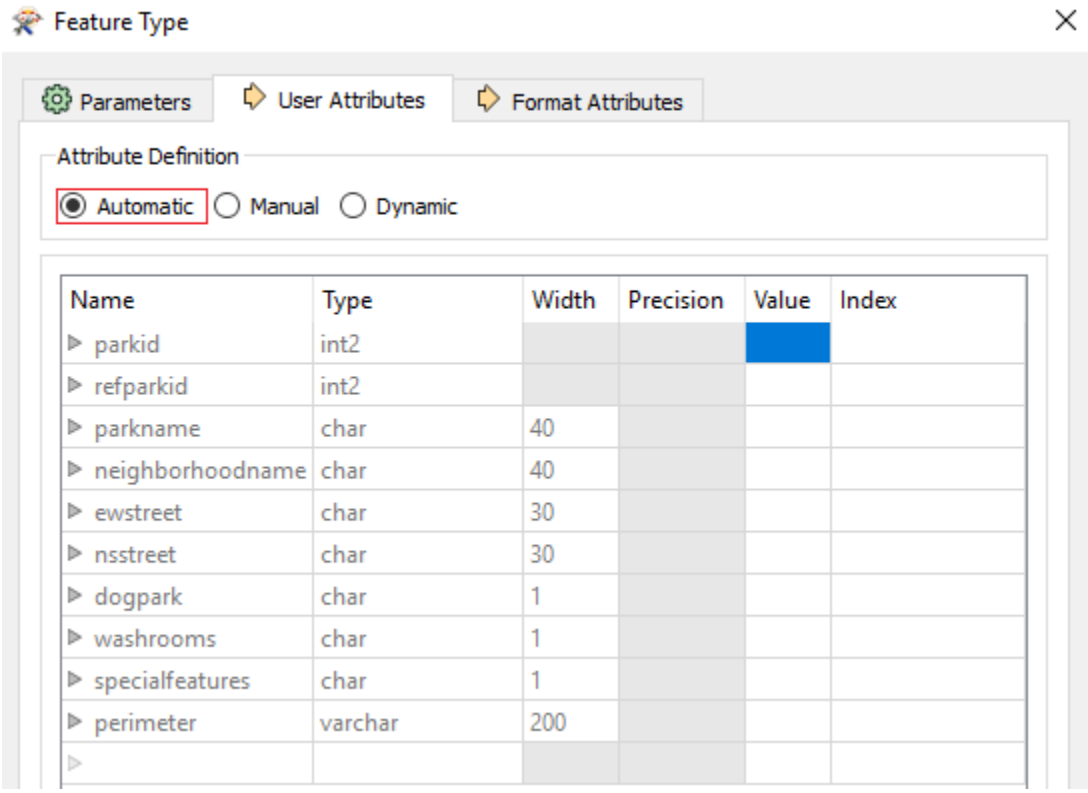
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to: ParksPerimeter
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the [Documentation](#).

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.
- Click OK to accept the changes and close the Writer Parameters dialog.



Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Documentation.

7. Add a FeatureHolder

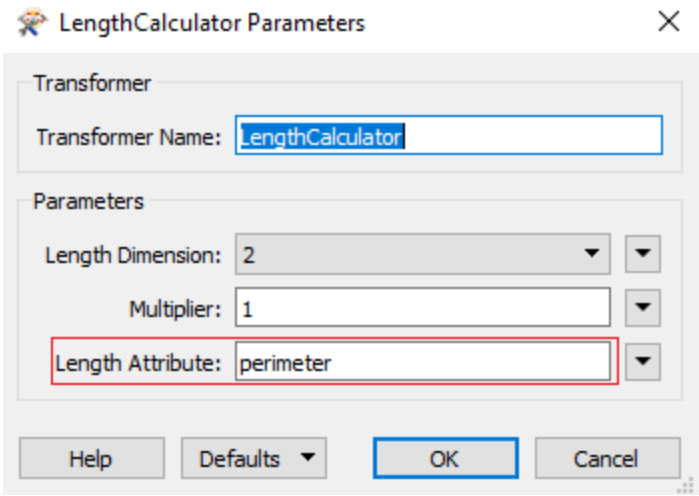
- Add a FeatureHolder to the canvas by typing “FeatureHolder” to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the Parks Reader to the FeatureHolder.

The FeatureHolder reads and stores the features in the Parks table. Once all of the features have been stored, the existing table is closed and which allows FME to drop the existing table and create the new table.

8. Calculate Perimeter

- Add a LengthCalculator and connect it to the FeatureHolder. Once the connection has been made, open the transformer parameters and set the Length Attribute to “perimeter”

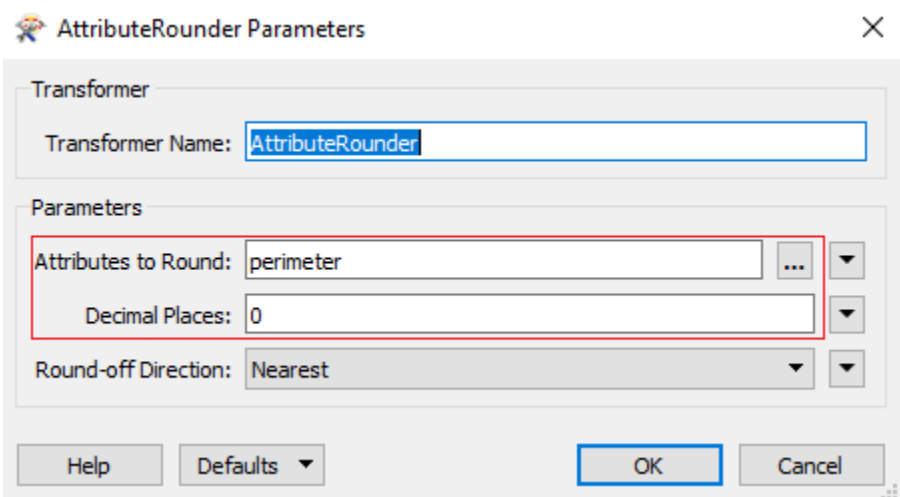
This will add a new column to the table and calculate the perimeter of each polygon feature in the dataset. Leave all other parameters as default and click OK to accept the changes and close the LengthCalculator parameters.



Note: the perimeter is calculated in map units.

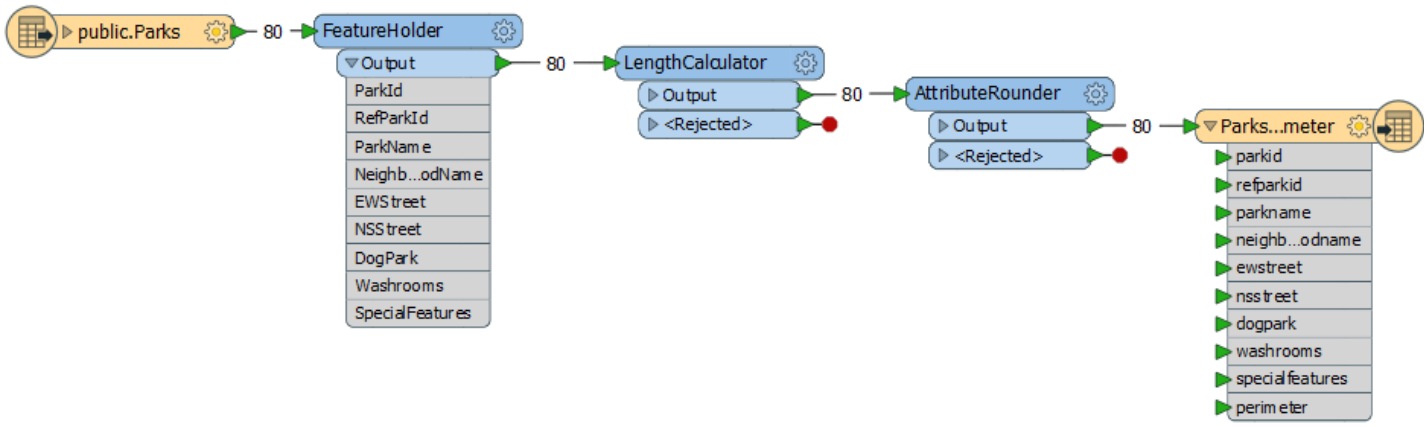
9. Round Area Values

- Next, add an AttributeRounder after and connect it to the LengthCalculator:Output port.
- Set the Attributes to Round to “perimeter” and set the Decimal Places to 0, then click OK.



9. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, your table will have a perimeter column with the perimeter for each polygon feature. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected. Alternatively, you can attach an Inspector to the output of the final transformer in the workspace if you wish to automatically open the workspace output the FME Inspector.

Results

Input

Table View

Table: Training Postgis - public RDS - 1 [POSTGIS] - public.Parks

	ParkId	RefParkId	ParkName	NeighborhoodName	EWStreet	NSStreet	DogPark	Washrooms	SpecialFeatures
1	1	-9999	<null>	Kitsilano	<null>	<null>	N	<null>	<null>
2	2	208	Rosemary Brow...	Kitsilano	W 11th Avenue	Vine Street	N	N	N
3	3	141	Tea Swamp Park	Mount Pleasant	E 15th Avenue	Sophia Street	N	N	N
4	4	-9999	<null>	Strathcona	<null>	<null>	N	<null>	<null>

Output

Table View

Training Postgis - public RDS [POSTGIS] - ParksPerimeter

	parkid	refparkid	parkname	neighborhoodname	ewstreet	nsstreet	dogpark	washrooms	specialfeatures	perimeter
1	1	-9999	<null>	Kitsilano	<null>	<null>	N	<null>	<null>	101
2	2	208	Rosemary Brow...	Kitsilano	W 11th Avenue	Vine Street	N	N	N	129
3	3	141	Tea Swamp Park	Mount Pleasant	E 15th Avenue	Sophia Street	N	N	N	290
4	4	-9999	<null>	Strathcona	<null>	<null>	N	<null>	<null>	256

Exercise 3 -*Calculating Length of a Line*

In this exercise, you will learn how to calculate the Length of bike paths using the LengthCalculator. To learn more about calculating length, see the [Distance Measurements with the LengthCalculator]() article.

Instructions

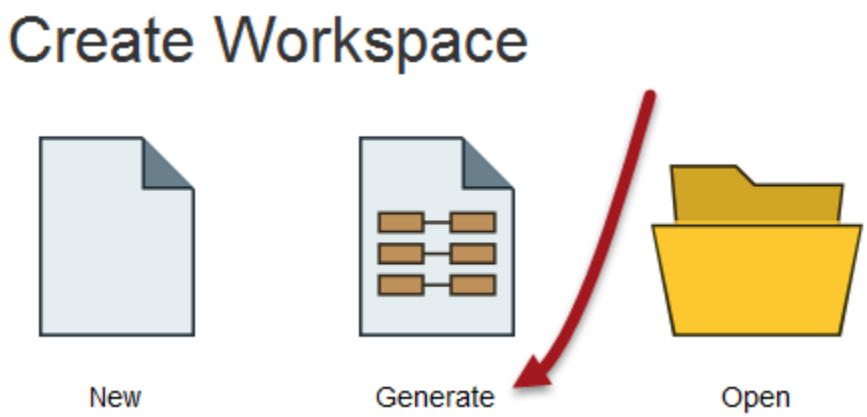
1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Select Generate Workspace

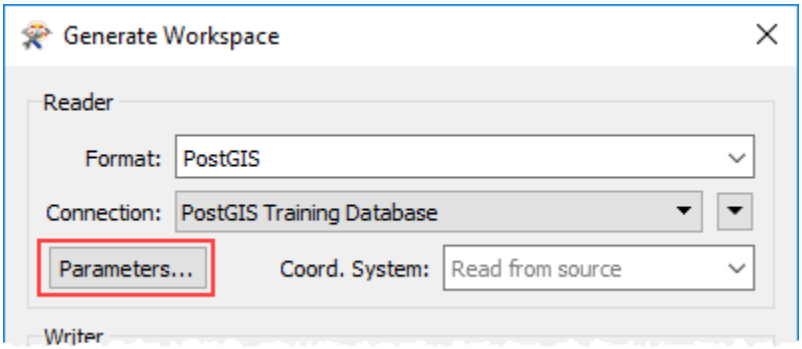
- In the Create Workspace part of the Start page, select the option to Generate a workspace.

Note: the shortcut to generate a workspace is Ctrl+G



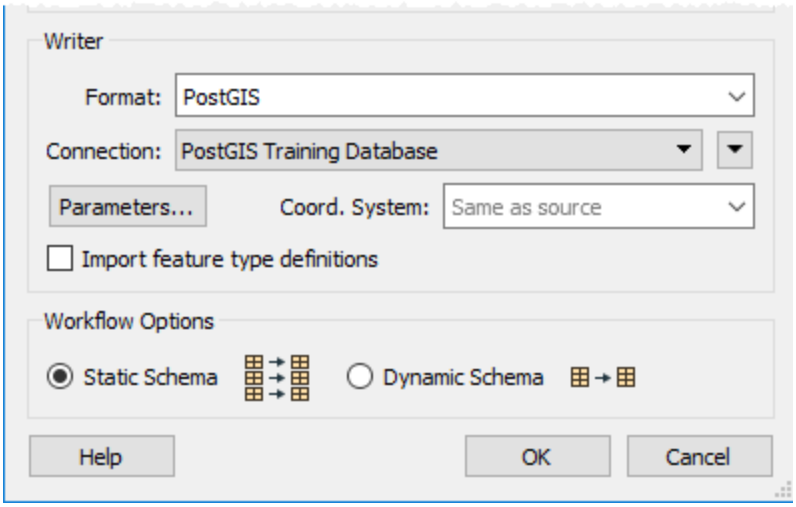
3. Set the Reader Format and Connect to the PostGIS Training Database

- In the Generate Workspace dialog, set the Reader Format to PostGIS.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection, select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the BikePaths table from the Table List.



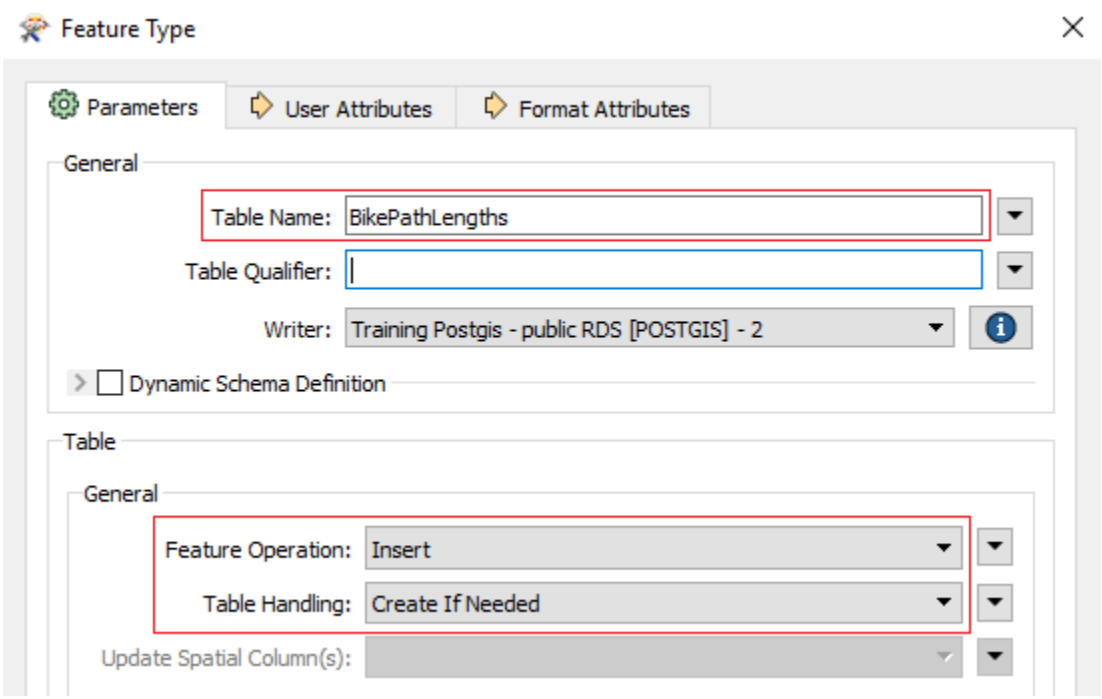
4. Set the Writer Format and Connect to the PostGIS Training Database

- Set the Writer Format to PostGIS and the Writer Connection to the same connection as the reader (PostGIS Training Database) and click OK to generate the workspace. The Generate Workspace dialog should look like the screenshot provided below:



5. Set the Feature Operation and Table Handling

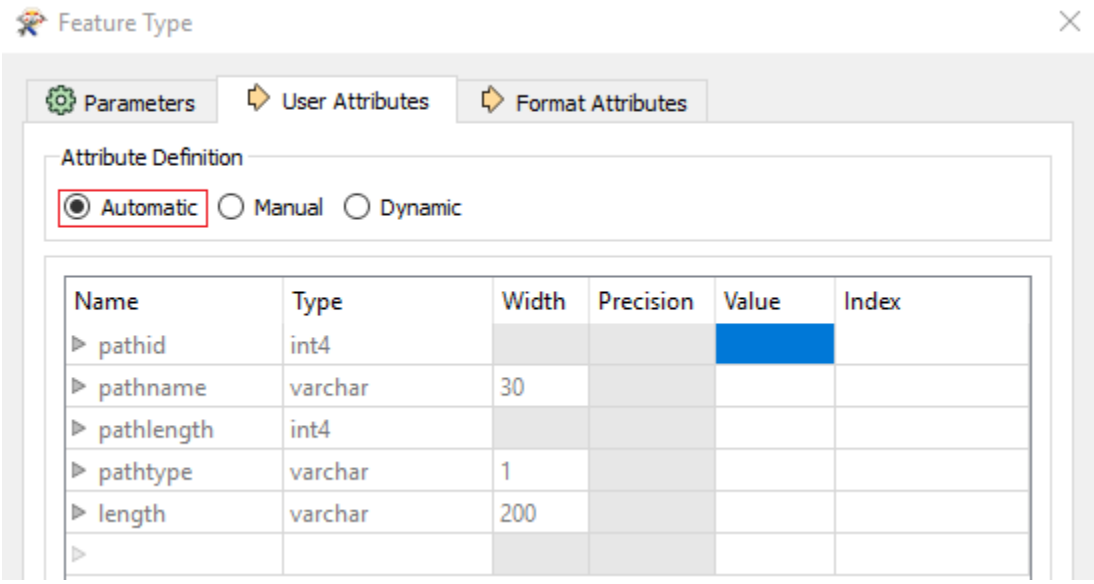
- After the workspace is generated, double-click the Writer Feature Type to open the Writer Feature Type Parameters dialog.
- In the General section, set the Table Name to: BikePathLengths
- In the Table section, ensure the Feature Operation is set to Insert and set the Table Handling to Drop and Create.



Note: Drop and Create is used when a table needs to be emptied and an update is made to the database schema. For example, drop and create is used to remove the table structure, possibly to add or remove columns, change data types, etc. For more information on Writer Feature Type Parameters such as Table Handling, see the [Documentation](#).

6. Update Attributes

- Switch to the User Attributes Tab in the Writer Parameters dialog, set the Attribute Definition to Automatic.
- Click OK to accept the changes and close the Writer Parameters dialog.



Note: An Automatic Attribute Definition is when Workbench automatically defines the list of attributes, depending on which Reader feature types are connected. Further, the list of attributes on the Writer feature type will update automatically whenever attributes are changed (i.e. renamed, removed, etc.) in the workspace. For more information on User Attributes and Attribute Definitions, see the Documentation.

7. Add a FeatureHolder

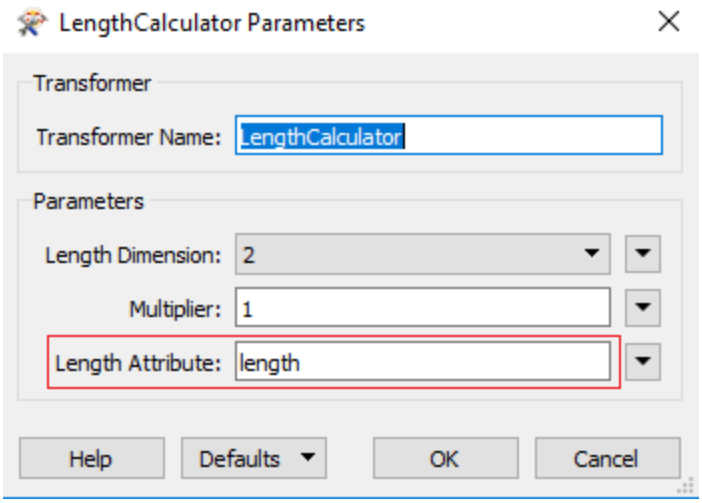
- Add a FeatureHolder to the canvas by typing “FeatureHolder” to bring up the list of FME Transformers in the Quick Add Search. Select the FeatureHolder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Once the transformer has been added, connect the BikePaths Reader to the FeatureHolder.

The FeatureHolder reads and stores the features in the BikePaths table. Once all of the features have been stored, the existing table is closed and which allows FME to drop the existing table and create the new table.

8. Calculate Length

- Add a LengthCalculator after the FeatureHolder and open the transformer parameters.
- Set the Length Attribute to “length” - this will add a new column to the table and calculate the length of line features.

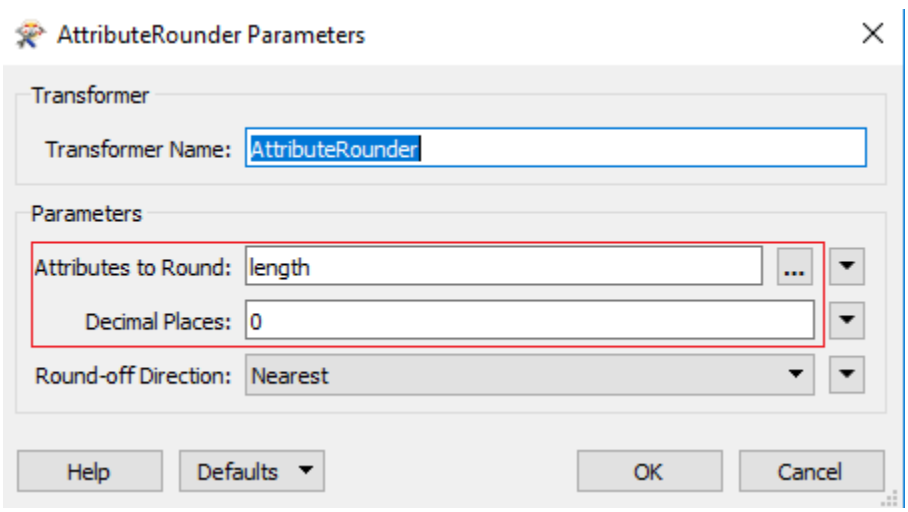
Leave all other parameters as default and click OK to accept the changes and close the LengthCalculator parameters.



Note: length is calculated in map units.

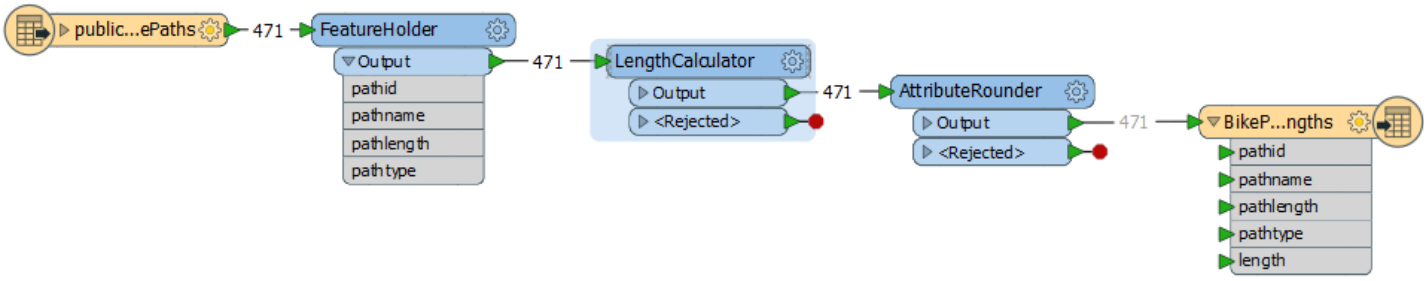
9. Round Area Values

- Next, add an AttributeRounder after and connect it to the LengthCalculator:Output port.
- Set the Attributes to Round to “length” and set the Decimal Places to 0, then click OK.



10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



After running the translation, your table will have a length attribute with the length of each line feature. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that

appears when the writer is selected. Alternatively, you can attach an Inspector to the output of the final transformer in the workspace if you wish to automatically open the workspace output the FME Inspector.

Results

Input

Table View

Training Postgis - public RDS [POSTGIS] - public.BikePaths

	pathid	pathname	pathlength	pathtype
1	7	Seaside	31669	L
2	7	Seaside	31669	L
3	7	Seaside	31669	L
4	7	Seaside	31669	L

Output

Table View

Training Postgis - public RDS [POSTGIS] - BikePathLengths

	pathid	pathname	pathlength	pathtype	length
1	7	Seaside	31669	L	12
2	7	Seaside	31669	L	40
3	7	Seaside	31669	L	33
4	7	Seaside	31669	L	23

Transformers

- AreaCalculator- Calculates the area of a polygonal object and stores the value in an attribute.
- AttributeRounder- Rounds off attributes to the specified number of decimal places.
- FeatureHolder- Stores incoming features until they have all arrived, and then releases them in their original order.
- LengthCalculator - Calculates the length of a feature and adds it as a new attribute. Using this transformer on a polygon will produce the polygon's perimeter.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Pivot Tables and FME

Introduction

A pivot table, also known as a cross-tab, allows a user to change the structure of a spreadsheet to perform data summarization. An example of a Microsoft Excel pivot table is found below which demonstrates summarization of ‘observed’ data, by the ‘region’, and ‘potential’ attributes. This article demonstrates the FME transformers that can be used to construct a pivot-table.

	A	B	C	D	E	F	G	H	I	J
1	city	region	potential	observed		Row Labels	potential	Sum of observed	Average of observed	Count of observed
2	Surrey	Vancouver	high	3.4		Cariboo	medium	7.4	7.40	1
3	Vancouver	Vancouver	high	1.2		Kootenay	high	9.9	4.95	2
4	Prince George	Cariboo	medium	7.4			medium	0.9	0.90	1
5	Richmond	Vancouver	medium	4.8		Northwest	low	9.3	9.30	1
6	Nelson	Kootenay	high	1.4			medium	1.1	1.10	1
7	Ferrie	Kootenay	high	8.5		Vancouver	high	4.6	2.30	2
8	Terrace	Northwest	low	9.3			medium	4.8	4.80	1
9	Prince Rupert	Northwest	medium	1.1		Grand Total		38.0	4.22	9
10	Golden	Kootenay	medium	0.9						

Initial table (orange) and the pivot table created in Microsoft Excel (blue)

Video

Note that this video was created using FME 2016 the interface might look different but the content and workflow is the same.

Downloads

AttributePivoterExample.fmwt

StatisticsCalculatorPivoterExample.fmwt

Pivot-Source.csv

Instructions

Example 1: Using the AttributePivoter Transformer

If you are only analyzing one attribute, you can use the AttributePivoter transformer to create a pivot table or crosstab in FME. This transformer allows you to create pivot tables that are similar to Excel, and this

approach has distinct advantages over other approaches, including:

- Adding dynamic attributes to the results, which makes it easier to write data to the pivot table. This is useful because it is sometimes difficult to predict the names used in the output data schema after the pivot has been performed.
- Summary statistics are created automatically by the AttributePivoter.
- Row order is preserved when using the AttributePivoter transformer.

1) Read in CSV data

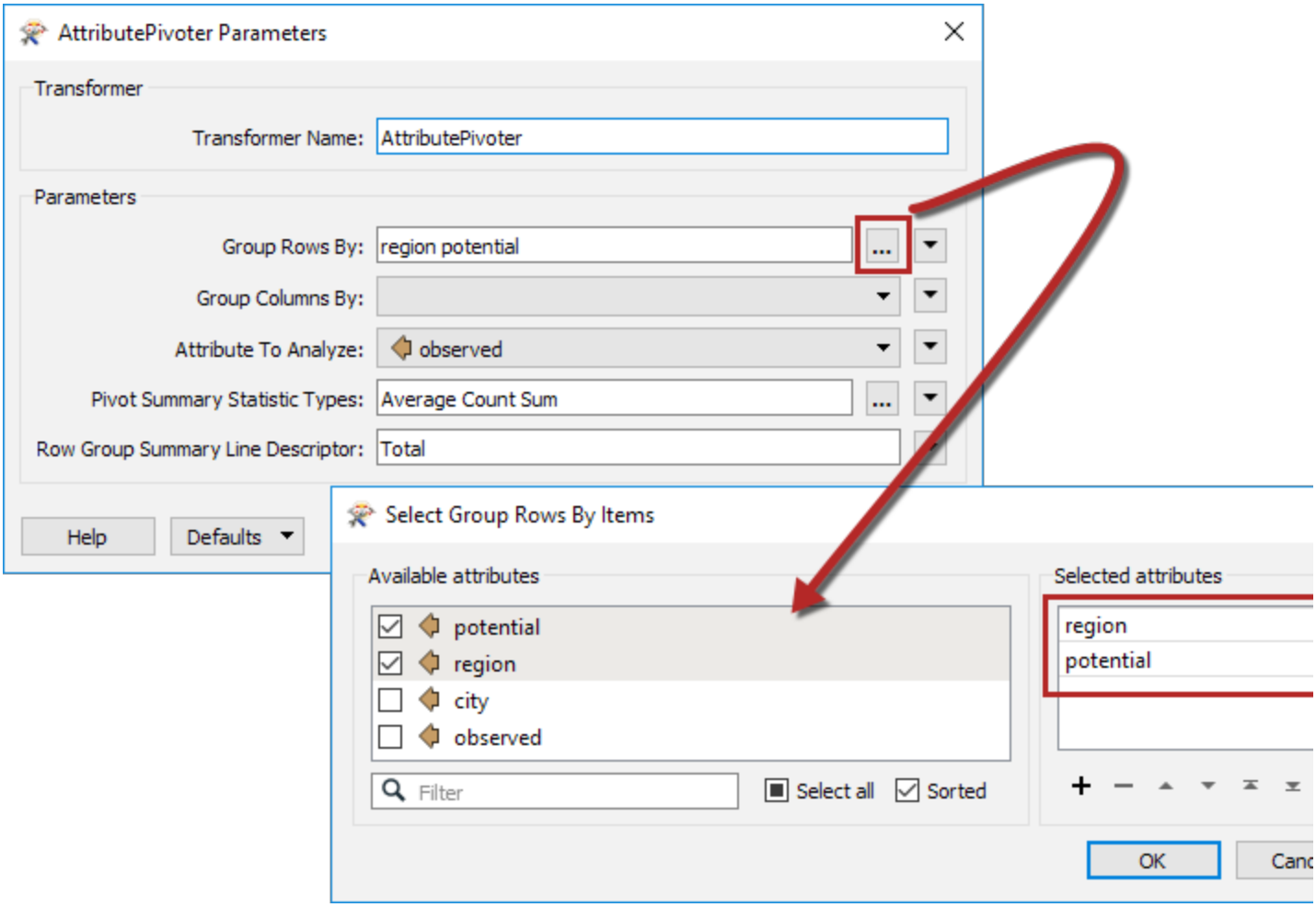
In a blank workspace, add a CSV (Comma Separated Value) Reader to the canvas and select the pivot-source.csv file as the Dataset. If you inspect the data, you will see that we have a small table containing city names, which region they are in, their potential and an observed number. We will pivot based on the observed attribute.

	city	region	potential	observed
1	Surrey	Vancouver	high	3.4
2	Vancouver	Vancouver	high	1.2
3	Prince George	Cariboo	medium	7.4
4	Richmond	Vancouver	medium	4.8
5	Nelson	Kootenay	high	1.4
6	Ferrie	Kootenay	high	8.5
7	Terrace	Northwest	low	9.3
8	Prince Rupert	Northwest	medium	1.1
9	Golden	Kootenay	medium	0.9

Initial data viewed in the FME Data Inspector

2) Pivot the table

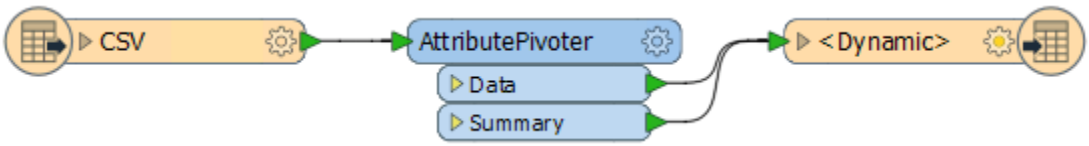
Add and connect the AttributePivoter transformer. In the parameters, set the Group By Rows to both potential and region, switch the order of the Selected Attributes so that region is first. Next, set the Attribute to Analyze to observed, then for the Pivot Summary Statistics Types select Average, Count, and Sum. Click OK.



AttributePivoter parameters; ensure that region is first in the Group By Rows order

3) Write to Microsoft Excel

Add a Microsoft Excel Writer and set the Sheet Definition to 'Dynamic (Advanced)' mode. Connect both the Data and Summary AttributePivoter output ports to the Feature Type. See the documentation for more information about Dynamic Workspace.



Final workspace, connect both the output ports on the AttributePivoter to the Dynamic Writer

4) Run the workspace and inspect the output.

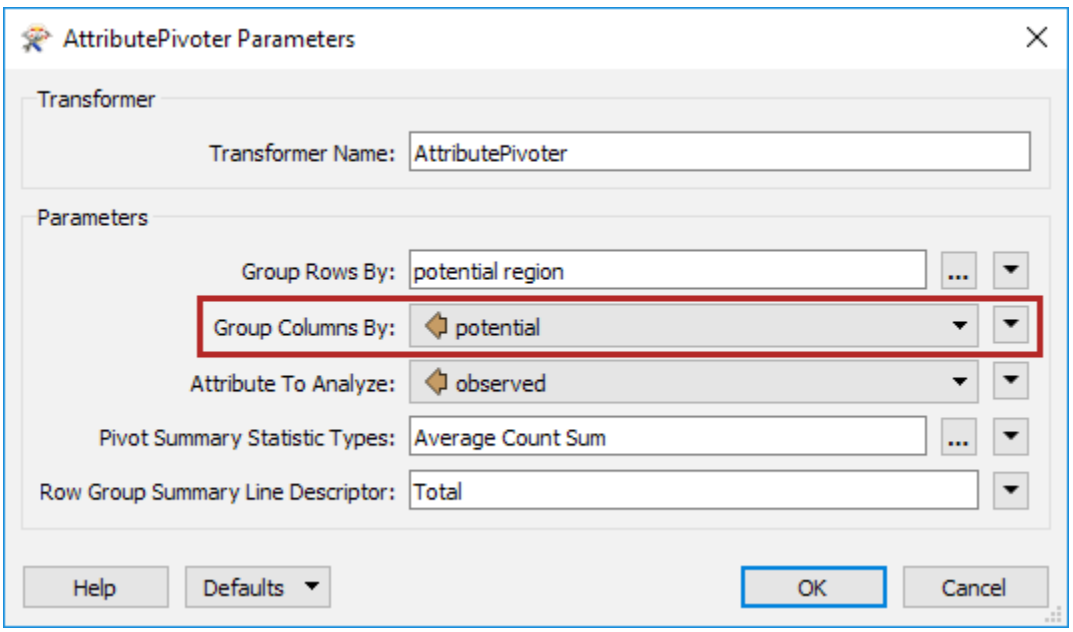
Run the workspace and then inspect the output either in Microsoft Excel or in the FME Data Inspector.

region	potential	observed Average	observed Count	observed Sum	Grand Total Average	Grand Total Count	Grand Total Sum
Cariboo	medium	7.4	1	7.4	7.4	1	7.4
Cariboo	potential Total	7.4	1	7.4	7.4	1	7.4
Kootenay	high	4.95	2	9.9	4.95	2	9.9
Kootenay	medium	0.9	1	0.9	0.9	1	0.9
Kootenay	potential Total	3.6	3	10.8	3.6	3	10.8
Northwest	low	9.3	1	9.3	9.3	1	9.3
Northwest	medium	1.1	1	1.1	1.1	1	1.1
Northwest	potential Total	5.2	2	10.4	5.2	2	10.4
Vancouver	high	2.3	2	4.6	2.3	2	4.6
Vancouver	medium	4.8	1	4.8	4.8	1	4.8
Vancouver	potential Total	3.133333333	3	9.4	3.133333333	3	9.4
region Total		**4.222222222**	**9**	**38**	**4.222222222**	**9**	**38**

Output data viewed and manually stylized in Microsoft Excel

5) Create more pivot columns

Back in FME Workbench, open the AttributePivoter parameters, change the Group Columns By potential to further pivot the table. Rerun the translation and inspect the results.



Change the Group Columns By to potential in the AttributePivoter

region	potential	high Average	high Count	high Sum	low Average	low Count	low Sum	medium Average	medium Count	medium Sum	Grand Total Average	Grand Total Count	Grand Total Sum
Cariboo	medium							7.4	1	7.4	7.4	1	7.4
Cariboo	potential Total							7.4	1	7.4	7.4	1	7.4
Kootenay	high	4.95	2	9.9							4.95	2	9.9
Kootenay	medium							0.9	1	0.9	0.9	1	0.9
Kootenay	potential Total	4.95	2	9.9				0.9	1	0.9	3.6	3	10.8
Northwest	low				9.3	1	9.3				9.3	1	9.3
Northwest	medium							1.1	1	1.1	1.1	1	1.1
Northwest	potential Total				9.3	1	9.3	1.1	1	1.1	5.2	2	10.4
Vancouver	high	2.3	2	4.6							2.3	2	4.6
Vancouver	medium							4.8	1	4.8	4.8	1	4.8
Vancouver	potential Total	2.3	2	4.6				4.8	1	4.8	3.133333333	3	9.4
region Total		**3.625**	**4**	**14.5**	**9.3**	**1**	**9.3**	**3.55**	**4**	**14.2**	**4.222222222**	**9**	**38**

Additional columns added when using the Group Columns By parameter

Example 2: Using the StatisticsCalculator Transformer

If you have multiple attributes to analyze or you want more control over which statistics are calculated using the StatisticsCalculator transformer to create a pivot table. The StatisticsCalculator can generate statistics for groups (or categories) of features.

1) Read in CSV data

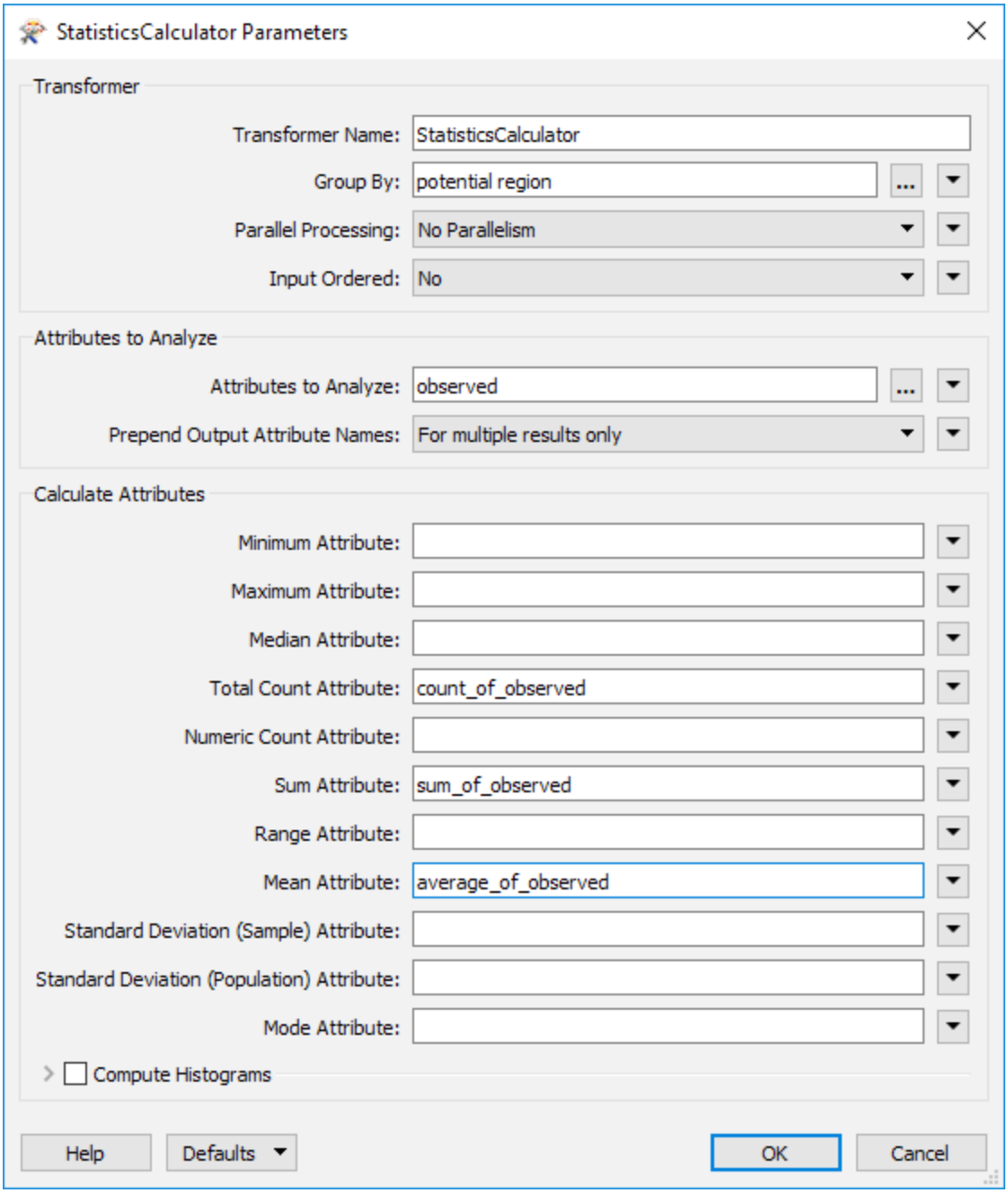
In a blank workspace, add a CSV (Comma Separated Value) Reader to the canvas and select the pivot-source.csv file as the Dataset. If you inspect the data, you will see that we have a small table containing city names, which region they are in, their potential and an observed number. We will pivot based on the observed attribute.

	city	region	potential	observed
1	Surrey	Vancouver	high	3.4
2	Vancouver	Vancouver	high	1.2
3	Prince George	Cariboo	medium	7.4
4	Richmond	Vancouver	medium	4.8
5	Nelson	Kootenay	high	1.4
6	Ferrie	Kootenay	high	8.5
7	Terrace	Northwest	low	9.3
8	Prince Rupert	Northwest	medium	1.1
9	Golden	Kootenay	medium	0.9

Initial data viewed in the FME Data Inspector

2) Calculate statistics

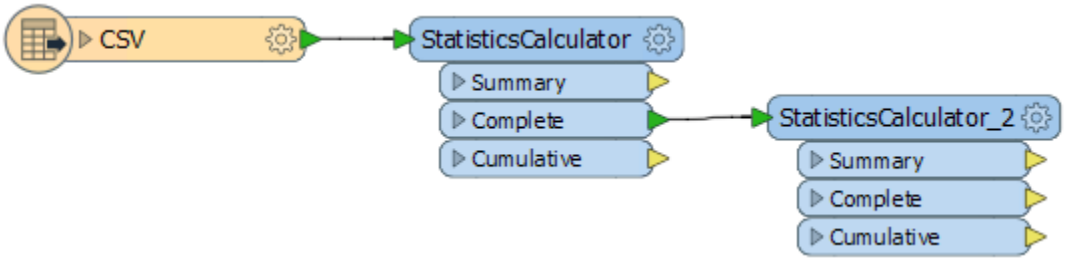
Add a StatisticsCalculator transformer to the canvas, in the parameters, set Group By to region and potential. Then under Attributes to Analyze select observed. Under Calculate Attributes, delete all the values. Then enter count_of_observed, sum_of_observed and average_of_observed into Total Count Attribute, Sum Attribute, and Mean Attribute, respectively.



StatisticsCalculator parameters; set the Group By, Attributes to Analyze and Calculate Attribute parameters

3) Calculate totals

Duplicate the StatisticsCalculator by right-clicking on it and selecting Duplicate. Connect the duplicated StatisticsCalculator to the Complete output port on the first StatisticsCalculator. In the parameters, remove the Group By attributes, the remaining parameters are the same as the first StatisticsCalculator.



Connect the duplicated StatisticsCalculator to the Complete output port on the first StatisticsCalculator

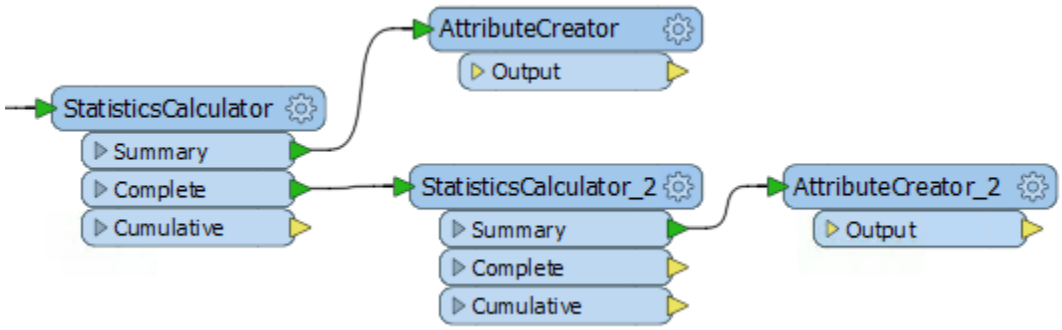
4) Order features

Next, we will order the features coming from the StatisticsCalculators. Add an AttributeCreator transformer to the canvas and connect it to the Summary output port on the first StatisticsCalculator. Create a new attribute called _FeatureOrder and set the value to 0

Add a second AttributeCreator and connect it to the Summary output port on the second StatisticsCalculator. Create three new attributes:

Attribute Name	Attribute Value
region	Grand Total
potential	<leave blank>
_FeatureOrder	1

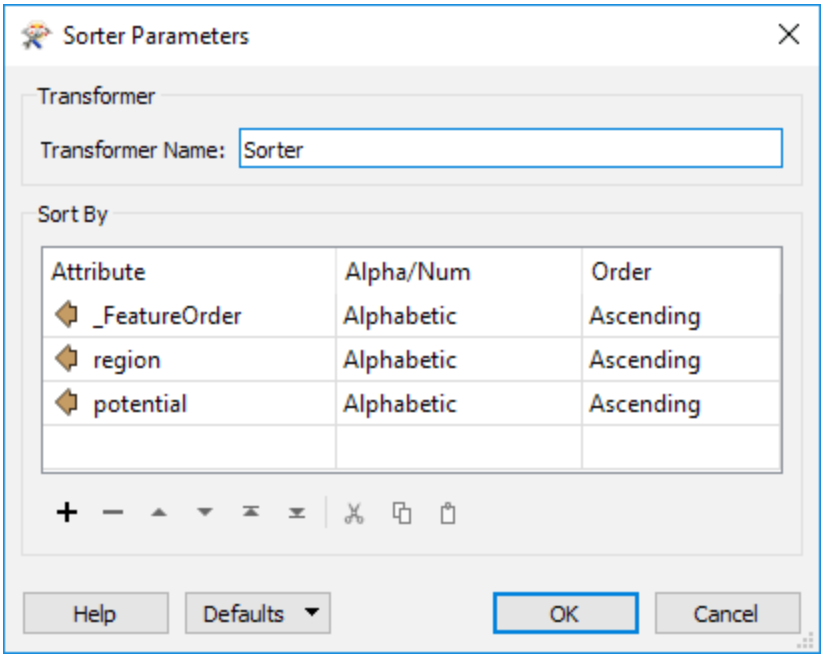
Note that region and potential are in lowercase to match the original attributes.



Connect the AttributeCreators to the Summary output ports on the StatisticsCalculators

5) Sort features

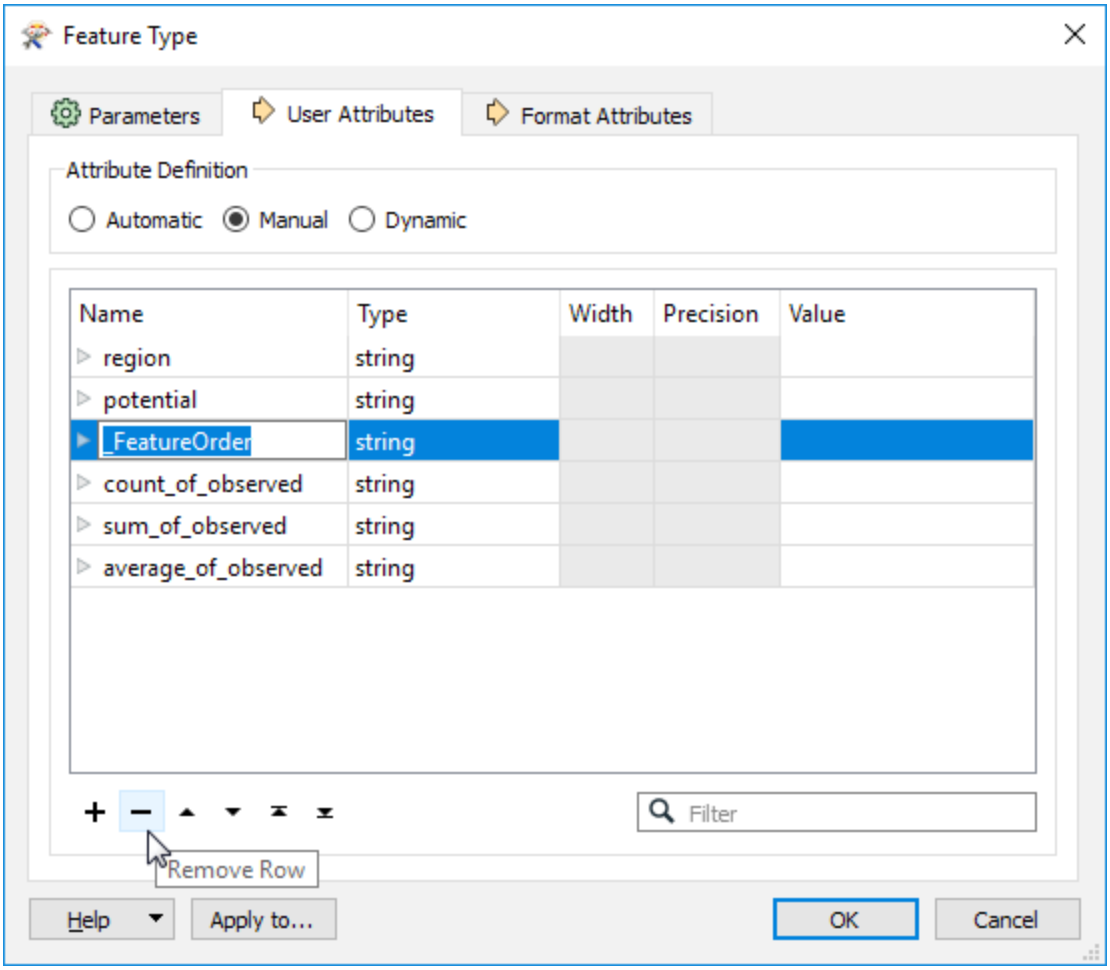
Add a Sorter transformer to the canvas. Connect it to both AttributeCreator Output ports. Then sort by the attribute _FeatureOrder by Numeric Ascending order. Then sort both region and potential by ascending alphabetical order.



Sorter parameters; sort _FeatureOrder, region, and potential by an Alphabetic Ascending order

6) Add a CSV Writer

Add a CSV (Comma Separated Value) writer to the canvas and browse to a location to save your output file. Set the File Definition to Automatic and click OK. Connect the writer to the Sorted output port on the Sorter. In the writer parameters, click on the User Attributes tab and set the Attribute Definition to Manual and remove _FeatureOrder.



In the User Attribute tab, switch to Manual Attribute Definition and remove _FeatureOrder

7) Run the workspace and inspect the output.

Run the workspace and then inspect the output either in Microsoft Excel or the FME Data Inspector.

region	potential	count_of_observed	sum_of_observed	average_of_observed
Cariboo	medium	1	7.4	7.4
Kootenay	high	2	9.9	4.95
Kootenay	medium	1	0.9	0.9
Northwest	low	1	9.3	9.3
Northwest	medium	1	1.1	1.1
Vancouver	high	2	4.6	2.3
Vancouver	medium	1	4.8	4.8
Grand Total		**9**	**38**	**4.222222222**

Output data viewed and manually stylized in Microsoft Excel

Additional Resources

[Tutorial: Getting Started with Excel]

ExcelStyler Documentation

Buffering features for Spatial Relationship Analysis

Buffering Features for Spatial Relationship Analysis

When assessing spatial relationships a common scenario is finding out whether one feature is within X distance of another feature; for example is there a fire station/hall within 5km of a planned fireworks factory!

FME has several transformers that assess spatial relationships when features physically overlap (inside, outside, intersect, match, etc) but no specific transformer to find relationships within X distance.

Therefore the solution is to create a buffer of X distance around one of the features and then carry out a standard spatial overlay of the two datasets.

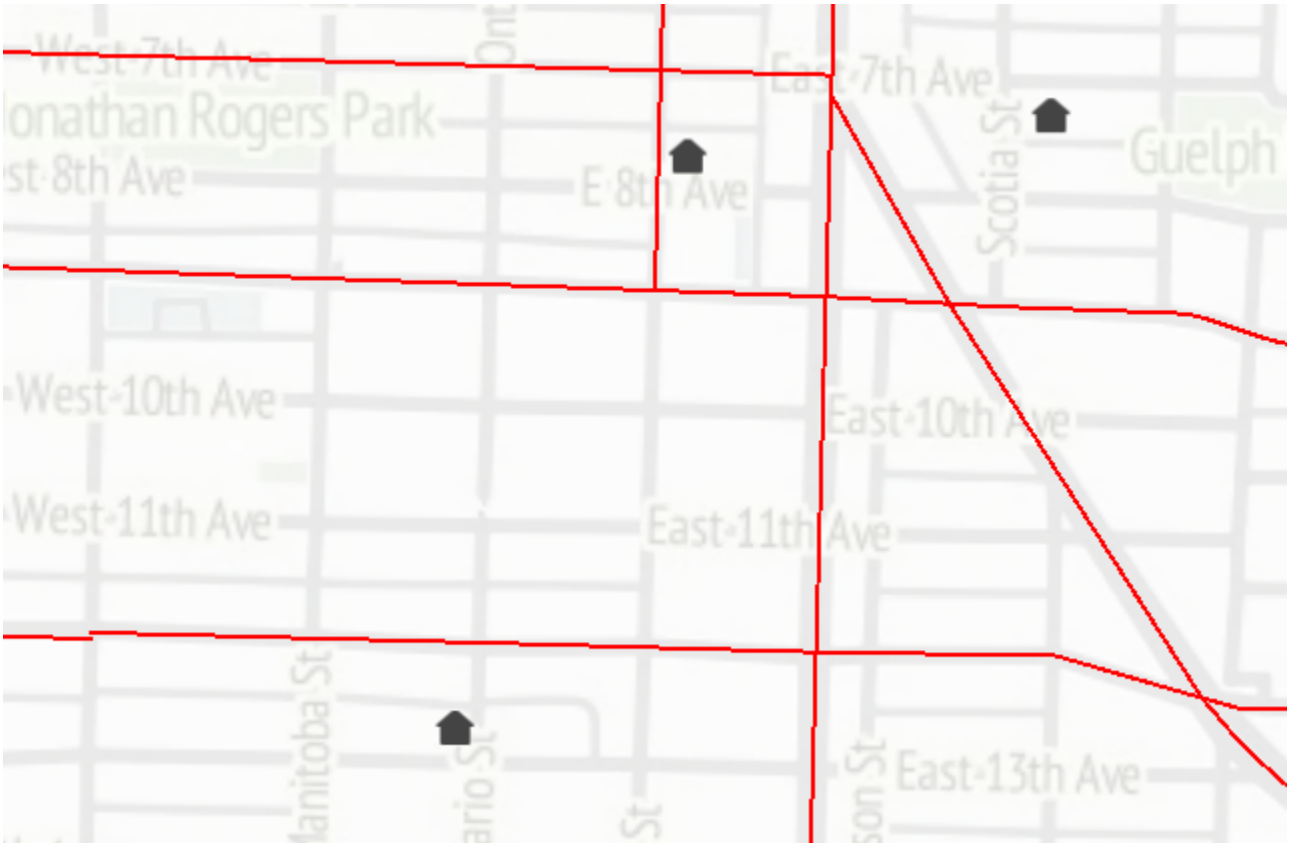
Video

Source Data

The first source dataset for this example is a set of lines (in a SpatiaLite database) representing roads that are snow removal routes; i.e. roads that have parking restrictions during winter so that snow can be cleared more efficiently.

The second dataset (in GML format) is a set of point features that represent addresses.

The datasets look like this in the FME Data Inspector:



The scenario here is that these properties are for sale and, as a member of the municipal GIS team, we have to tell prospective buyers if a property is within 25 metres of a snow removal route.

Step-By-Step Instructions

1. Start FME Workbench and begin with an empty canvas.

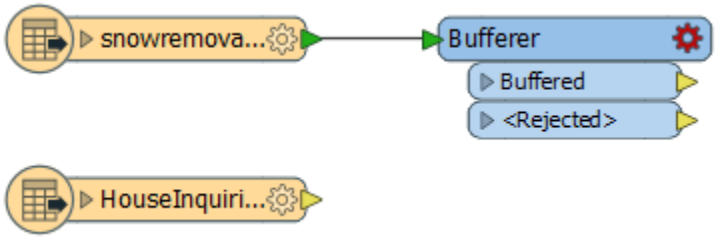
Select Readers > Add Reader from the menubar.

Set the data format to SpatiaLite. Select the attached sl3 file as the source dataset.

2. Again, select Readers > Add Reader from the menubar.

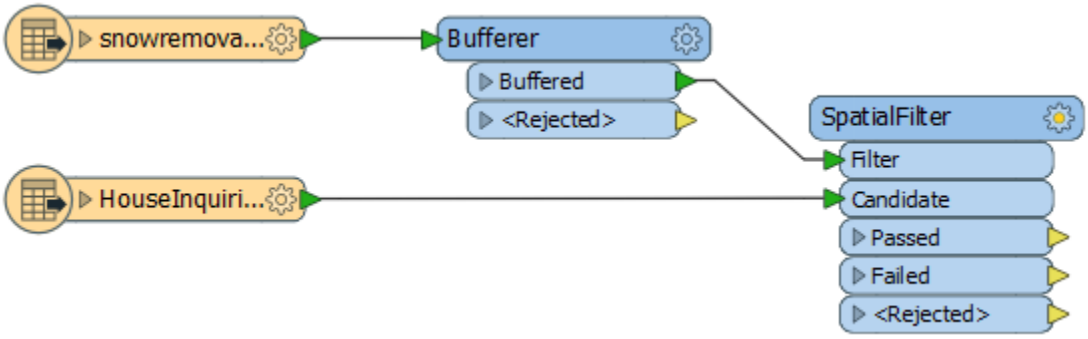
This time set the data format to GML (Geography Markup Language). Select the attached GML dataset as the source.

3. To find if one of these features is within 25 metres of another requires the creation of buffers around one set of features. So, add a Bufferer transformer. Connect it to the snowremovalroutes feature type:



Open the parameters dialog. Set the buffer amount to 25. The units do not need to be specified; they are taken from the units of the current coordinate system.

4. Add a SpatialFilter transformer. Connect the Bufferer:Buffered port to the SpatialFilter:Filter port. Connect the HouseInquiries feature type to the SpatialFilter:Candidate port:



Open the parameters dialog. Set the "Tests to Perform" parameter to carry out the test *Contains*.

This will now assess whether each candidate (property address) is contained by the buffered area of a snow removal route. Features that pass the test must fall within such a buffer and so are within 25m of a snow removal route.

5. Add some Inspector transformers and run the workspace. By inspecting (and setting the symbology of) the output, you can see that the procedure has successfully picked out which addresses are (and which are not) within 25 metres of a snow removal route:

Downloads

Source Datasets

Completed Workspace as a Template

Polygon Within a Distance of Selected Features

Overview

Typically select by location is done using a graphical user interface like ArcMap or QGIS. However, in FME you can perform the same task by querying your data with a Tester to “select” the desired feature(s), then select the features that pass the query using the NeighborFinder.

Downloads

polygon-within-distance-of-selected-features.fmwt

parcelsgdb.zip

schoolsgdb.zip

Note: Since you are working on a public facing database, tables can occasionally be overwritten and/or modified. If your translation produces unexpected results, see the Resetting the PostGIS Training Database article. Alternatively, you can download the provided file(s) and replace the PostGIS reader with a reader of the appropriate format.

Exercise

In this scenario, you will learn how to query the school's dataset in order to select parcel polygons within 200m of King George Secondary and Henry Hudson Elementary.

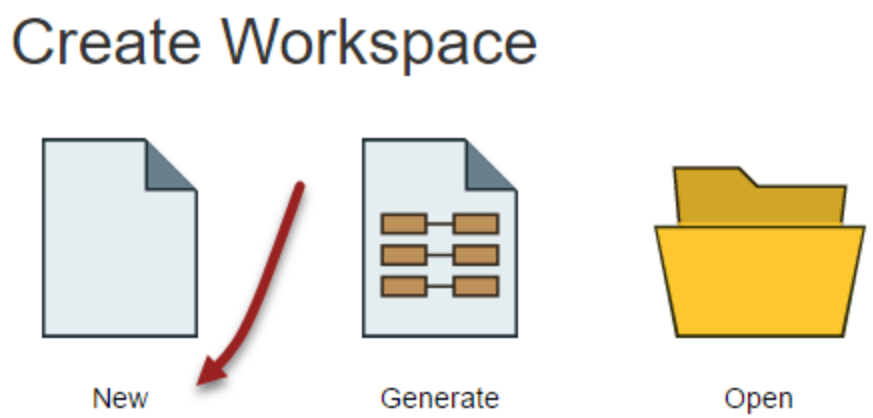
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

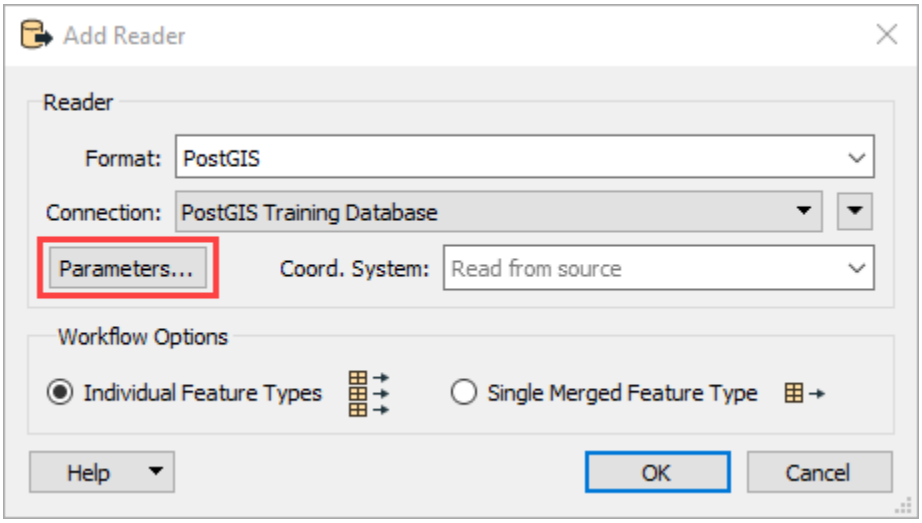
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Schools table from the Table List.

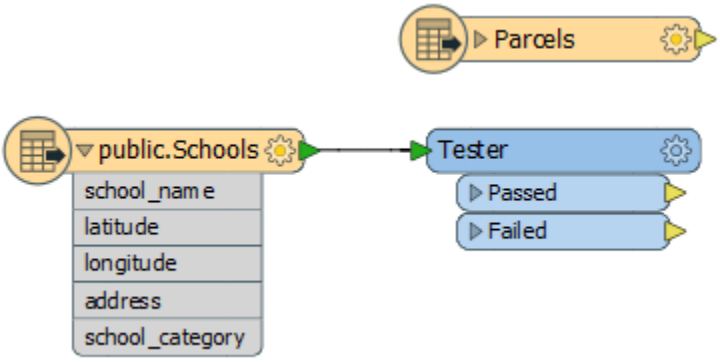


4. Add an Esri Geodatabase (File Geodb Open API) Reader

- Start typing “Geodatabase” without anything selected on the canvas, then select the Esri Geodatabase (File Geodb Open API) format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the Parcels geodatabase (for example, C:\Users\Documents\FME\Parcels\Parcels.gdb).

5. Add a Tester

- Select the Schools Reader Feature Type and start typing “Tester”, then select the Tester from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas. This will automatically connect the Schools reader to the Tester that you just added.



6. Open the Tester Parameters

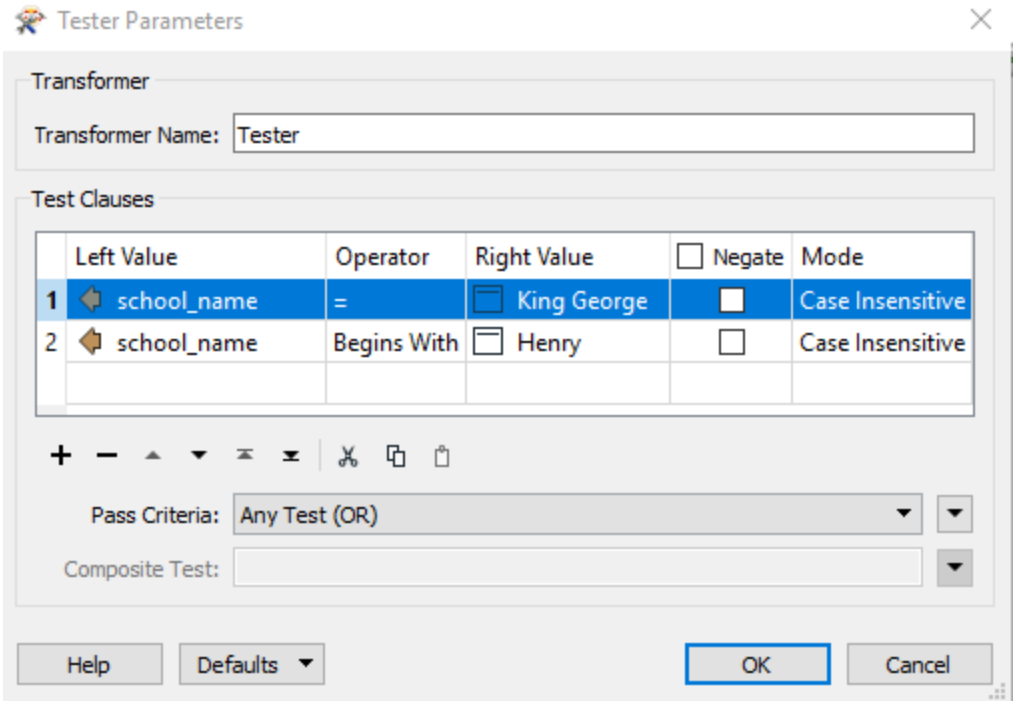
The Tester will be used to “select” schools from the dataset. You will set two test conditions that will use the Pass Criteria “Any Test (Or)” - meaning if any test passes, the input feature will be classified as passed. For more information on the Tester and Pass Criteria, see the Tester Documentation.

- To create a new test condition, click on the empty row in the Left Value column of the Test Clauses section (this will reveal two buttons, an ellipsis and a drop-down list).
- Click on the drop-down list, then select “school_name” from the list of Attribute Values.
- Next, click the Right Value cell in the same row and type in “King George Secondary”. Leave the Operator as the default (equals) for this clause as it will query the schools dataset, school_name column and only return the feature named King George Secondary.
- Similarly, add a second test clause by clicking on the next row in the Left Operator column.

The two test clauses should be entered as follows:

Left Value	Operator	Right Value	Negate	Mode
school_name	=	King George Secondary		Case Insensitive
school_name	Begins With	Henry		Case Insensitive

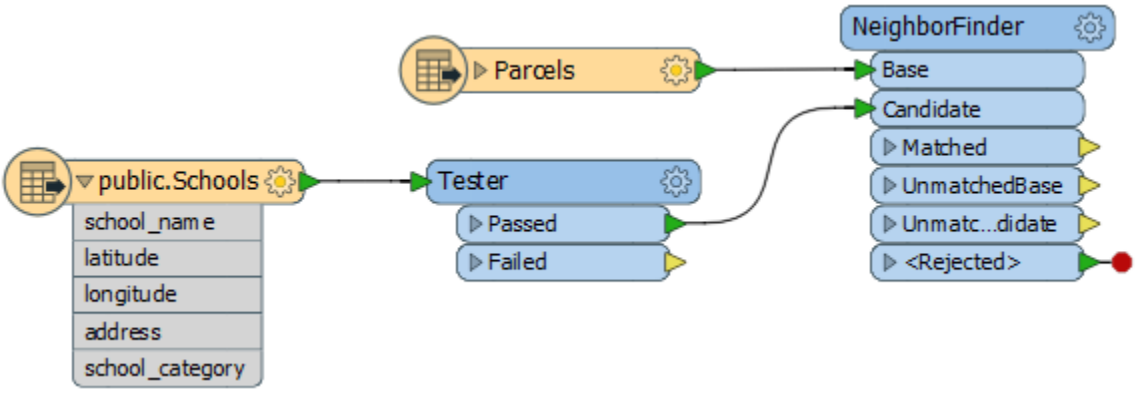
In this case, you will use the default pass criteria, Any Test (OR), your Tester Parameters dialog should match the screenshot provided below:



7. Add a NeighborFinder

- Connect the Parcels Reader to the NeighborFinder:Base port and the Tester:Passed port to the NeighborFinder:Candidate port.

So far, your workspace should look like the screenshot below:



8. Set the Search Distance

The NeighborFinder takes in two sets of features - Base and Candidate. For each Base feature, the transformer checks the Candidates for matches, based on proximity and parameter selections. In this case, it will check all Candidates that fall within a specified distance of the Base feature. For more technical information on the NeighborFinder see the Documentation. Alternatively, to learn more about Spatially Based Join Transformers see the Desktop Basic Course Manual.

- Open the NeighborFinder Parameters.
- In the Parameters section, set the Maximum Distance to 200.

The Maximum Distance is measured in the units of coordinates of the input features - meters in this case.

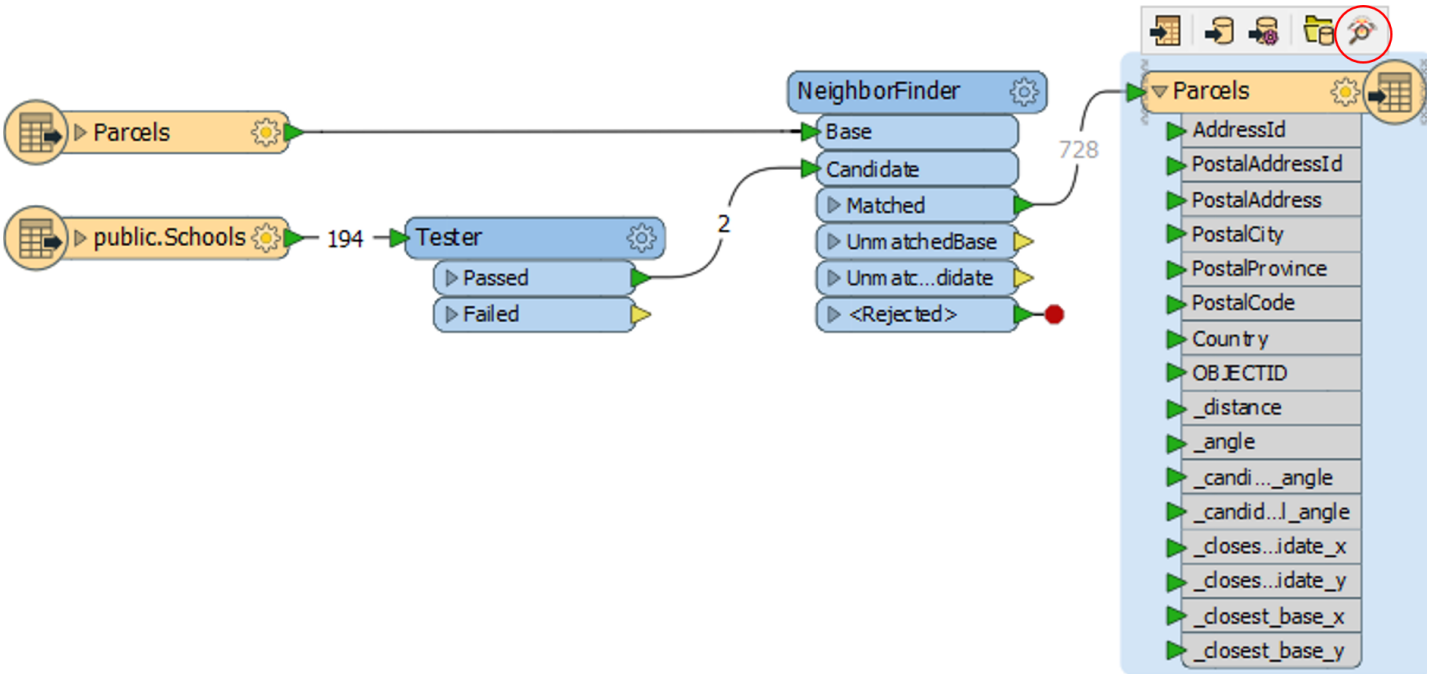
Note: To correctly analyze spatial relationships, all features should be in the same coordinate system. The [Reprojector]() may be useful for reprojecting features within the workspace. However, in this exercise, the features are already in the same coordinate system.

9. Add an Autodesk AutoCAD DWG/DXF Writer

- Similar to adding a reader, type “DWG” then select the Autodesk AutoCAD DWG/DXF format from the list of Writers.
- Click on the dataset ellipsis to specify where to save your data and name the file: ParcelsNearSchools (for example, C:\Users\Documents\FME\Parcels\ParcelsNearSchools.dwg)
- Set the Layer Definition to Automatic to ensure all attributes that are created or renamed will be written to the dwg file.
- Connect the ParcelsNearSchools Writer to the NeighborFinder:Matched port.

10. Run the Workspace

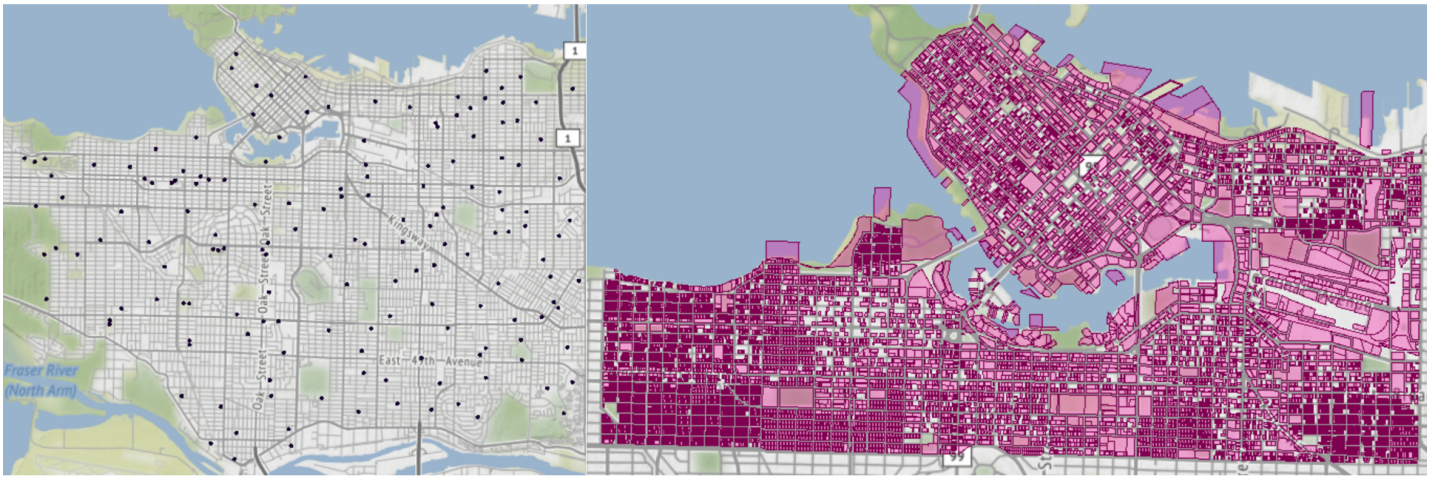
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



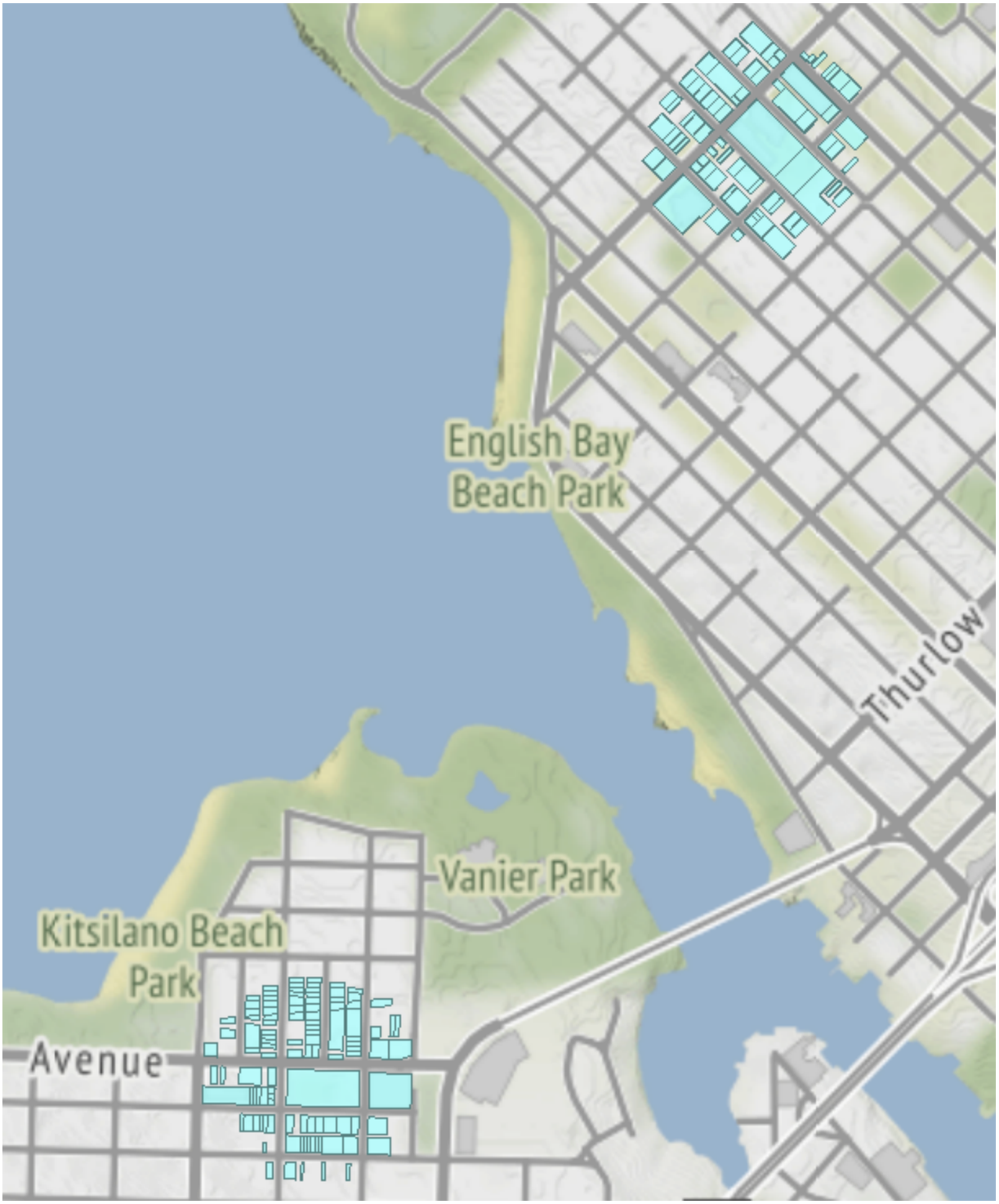
After running the workspace, you will have a new dataset of the parcels within 200m of the King George Secondary and Henry Hudson Elementary in the DWG format. You can view the new table by selecting the writer and clicking the Inspect button in the shortcut menu that appears when the writer is selected. Alternatively, you can attach an Inspector to the NeighborFinder:Matched port to automatically open the output in the FME Data Inspector after the translation has successfully run.

Results

Input


Fraser River
(North Arm)
Oak Street
Oak Street
Oak Street
Kingsway
East 49th Avenue

Output



Transformers

- NeighborFinder - Finds the nearest Candidate feature(s) to each Base feature and merges their attributes onto the Base feature.
- Tester - Evaluates one or more tests on a feature, and routes the feature according to the outcome of the test(s).

Data Attribution

Find Nearest Features

Overview

In this tutorial, you will learn how to find the nearest features using point and polygon datasets. The purpose of this operation is to find the nearest feature from an object of interest based on the straight line distance between features (i.e. starting from a point of interest, such as a school in Vancouver, find the nearest park polygon feature). In order to accomplish this, you need to use the NeighborFinder transformer. The NeighborFinder is typically used to identify nearest features, features within a specified distance, or finding the closest feature in a certain direction.

Downloads

find-nearest-features.fmwt

parksgdb.zip

schoolsgdb.zip

Exercise

In this scenario, you are interested in determining where the nearest park with washroom facilities is from the starting point, Alexander Academy. In this exercise, you will learn how to query your datasets based on attribute values and perform a simple spatial analysis using a point and polygon dataset. For an in-depth tutorial on finding nearest neighbors and calculating both the distance and bearing for multiple features, see the Determining Nearest Neighbors article.

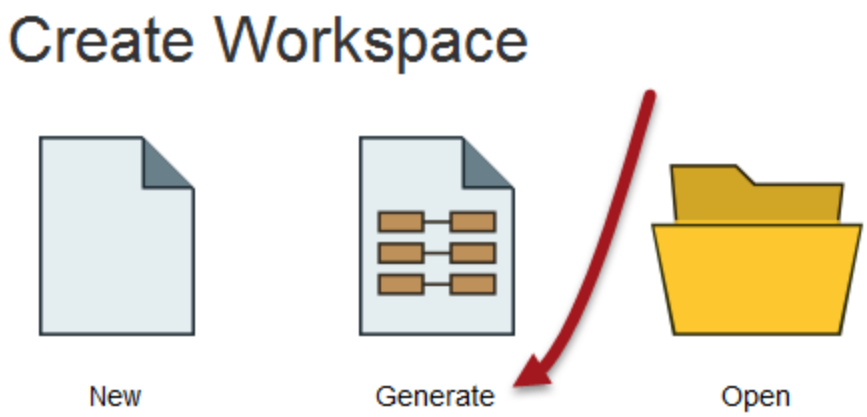
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

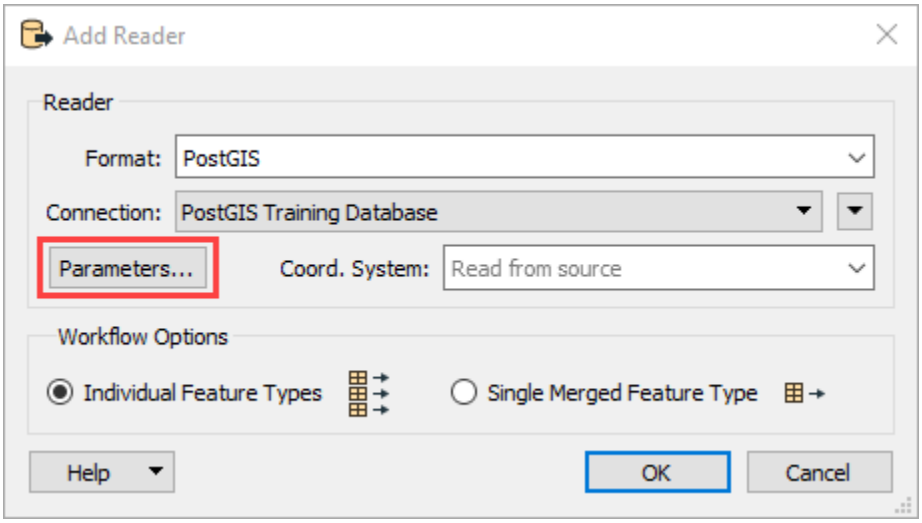
2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a PostGIS Reader

- Start typing “PostGIS” without anything selected on the canvas, then select the PostGIS format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Next, select the PostGIS Training Database from the Reader Connection list. If you have not already set up the database connection select Add Database Connection and enter the following parameters:
  - Host: postgis.train.safe.com
  - Port: 5432
  - Database: fmedata
  - Username: fmedata
  - Password: fmedata
- Next, open the Reader Parameters and select the Schools table from the Table List.



4. Add an Esri Geodatabase (File Geodb Open API) Reader

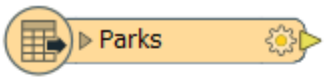
- Start typing “Geodatabase” without anything selected on the canvas, then select the Esri Geodatabase (File Geodb Open API) format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Click on the Dataset ellipsis button, then navigate to the sample data folder and select the Parks geodatabase (for example, C:\Users\Documents\FME\Parks\Parks.gdb).

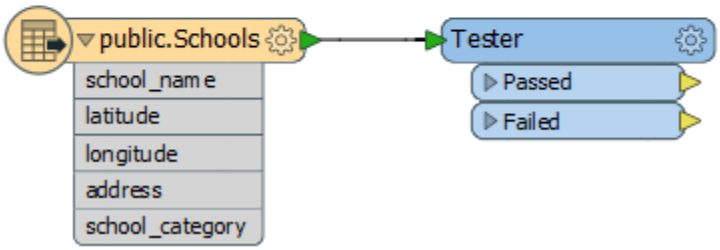
5. Add a Tester

- Add a Tester to the canvas by typing “Tester” to bring up the list of FME Transformers in the Quick

Add Search. Select the Tester from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.

- Once the transformer has been added to the canvas, connect the Schools Reader Feature Type to the Tester.





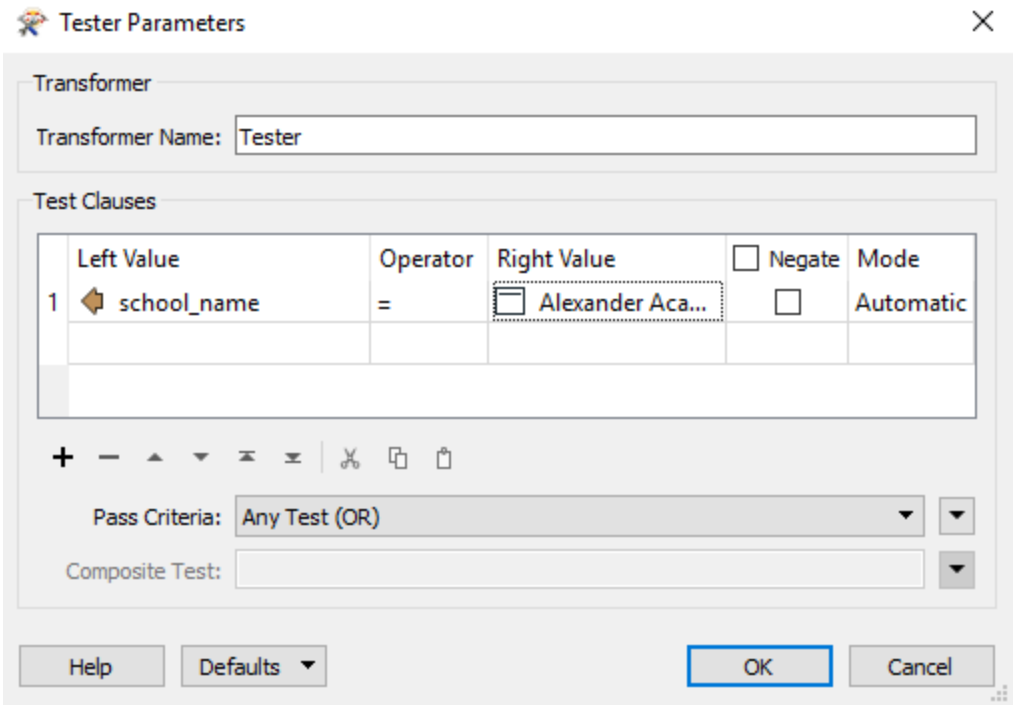
6. Select Alexander Academy

The Tester will be used to “select” a school from the dataset.

- Once the Tester has been added and the connection to the Schools reader has been made, double-click the Tester or click on the gear icon to open the transformer parameters.
- To create a new test condition, click on the blank row in the Left Value column in the Test Clauses section.

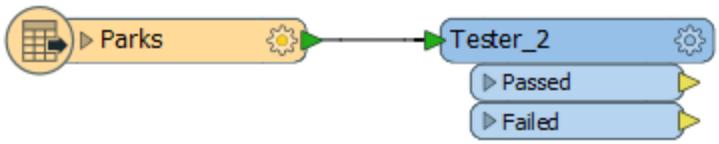
This will reveal two buttons, an ellipsis button and a drop-down list.

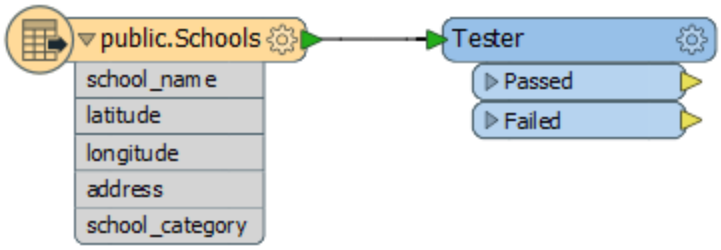
- Click on the drop-down list, then select “school_name” from the list of Attribute Values.
- Next, click the Right Value cell in the same row and type in “Alexander Academy”.
- Leave the Operator as the default (equals) for this clause as it will query the schools dataset, school_name column and only return the feature named “Alexander Academy”.



7. Add another Tester

- Add another Tester to the canvas and connect it to the Parks Reader.





To find out more about duplicated transformers, see the [Desktop Basic Course Manual](#).

Note: FME allows you to duplicate transformers that are currently on the canvas by right-clicking the transformer and selecting duplicate from the menu. Alternatively, you can select the transformer and use the shortcut CTRL+D to duplicate the selected transformer(s). So far, your workspace should look like the screenshot provided below:

8. Open the Tester_2 Parameters

The second Tester will be used to query the Parks dataset so only features with the specified attributes can be returned. You will set two test conditions that will use the Pass Criteria “All Test (AND)” - all tests must pass for the input feature to be classified as Passed. For more information on the Tester and Pass Criteria, see the

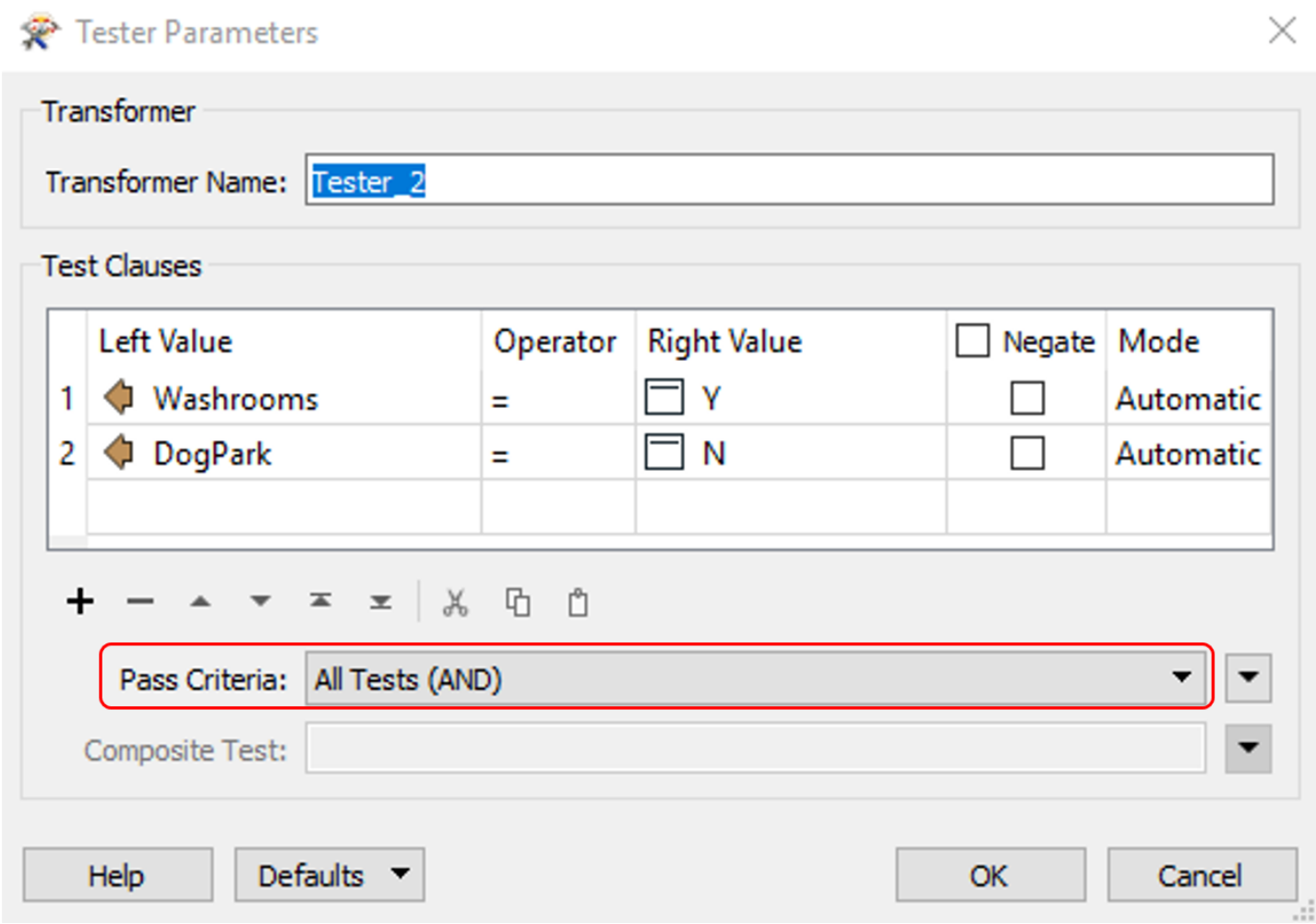
Tester Documentation.

- Using the same steps for adding a new test clause in step 6, create two test clauses in order to find parks that have washrooms and are not dog parks. The two clauses are:

Left Value	Operator	Right Value
Washrooms	=	Y
DogPark	=	N

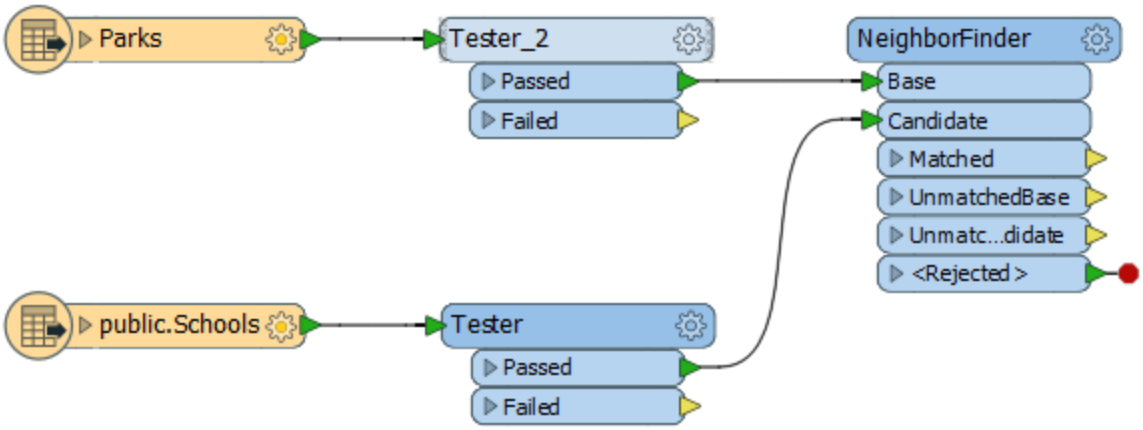
- Lastly, change the Pass Criteria to “All Tests (AND)’ - this will ensure only parks that meet both test clauses are returned.

Ensure your Tester Parameters dialog matches the screenshot below then click OK to accept the changes and close the dialog.



9. Add a NeighborFinder

- Connect the Tester_2:Passed port (Parks Tester) to the NeighborFinder:Base port and the Tester:Passed port (Schools Tester) to the NeighborFinder:Candidate input port. Your connections should match the screenshot below:



In this case, the default parameters for the NeighborFinder work because it will find only find 1 neighbor.

Note: To correctly analyze spatial relationships, all features should be in the same coordinate system. The Reprojector may be useful for reprojecting features within the workspace. However, in this case, both features are already in the same coordinate system.

10. Add a Sorter

- Connect the Sorter to the NeighborhoodFinder:Matched port. You have to sort the matched features because the NeighborFinder will calculate the distance to each park feature.

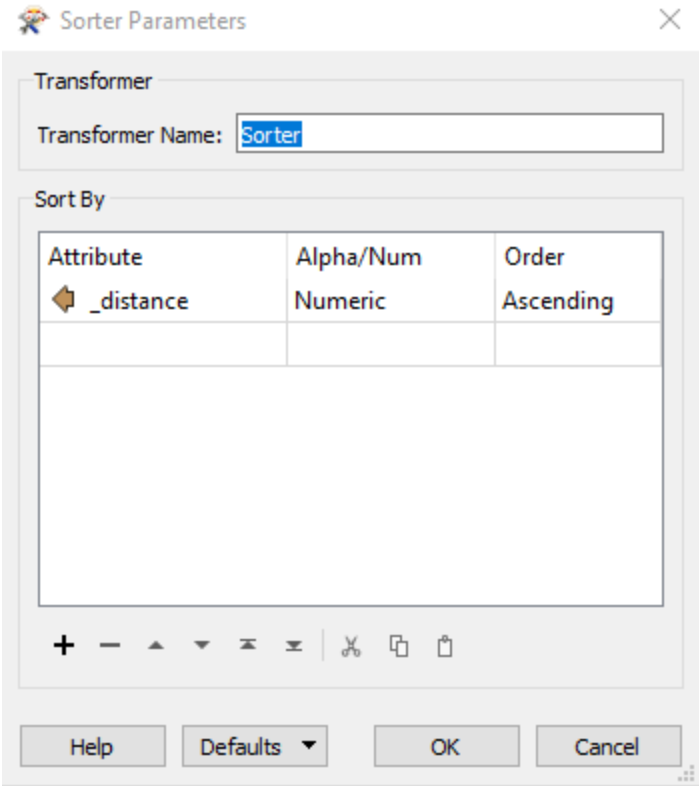
By sorting numerically in ascending order, the nearest features will be placed at the top of the list in the attribute table.

11. Sort By Distance

Since you are interested in finding the nearest feature you will sort based on the _distance attribute in Numerical Ascending order.

- Open the Sorter parameters and set the Sort By condition to:

Attribute	Alpha/Num	Order
_distance	Numeric	Ascending

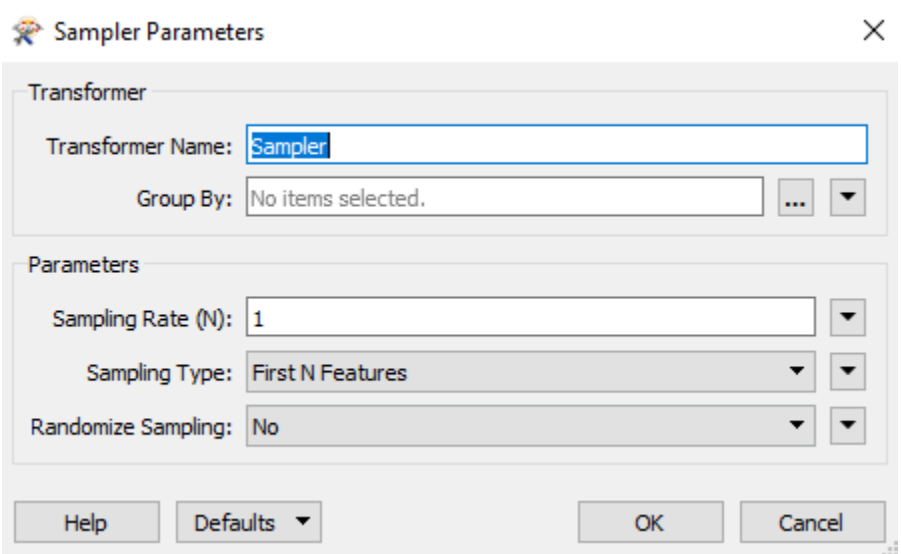


- Once you have set the Sort By parameter, click OK to accept the changes and close the Sorter parameters dialog.

12. Add a Sampler

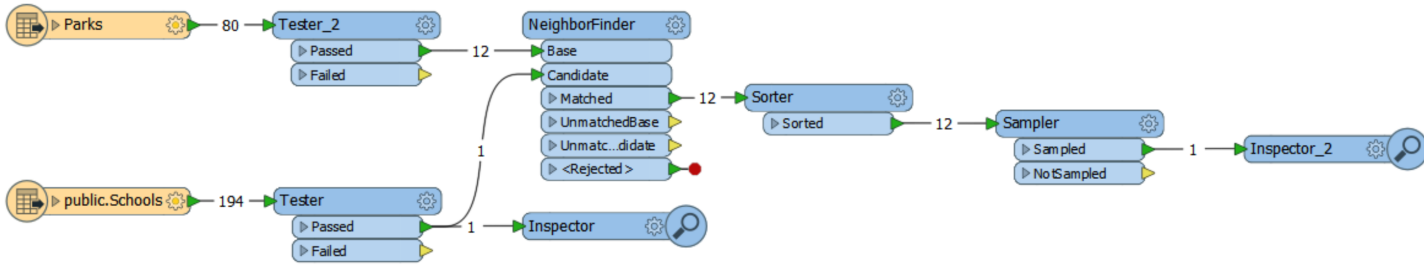
In this case, the Sampler transformer is used to select the first record in the attribute table (i.e. the nearest feature since we just sorted by ascending order on the _distance attribute).

- Connect the Sampler to the Sorter:Sorted port.
- Once the connection has been made, open the Sampler parameters and change the Sampling Type to: First N Features.



13. Add an Inspector

- Connect one of the Inspectors to the Tester:Passed port (Schools Tester) to display the point for Alexander Academy in the FME Data Inspector.
- Connect the second Inspector to Sampler:Sampled port to display the nearest park to Alexander Academy that has washrooms and is not a dog park. Your final workspace should look like the screenshot below:

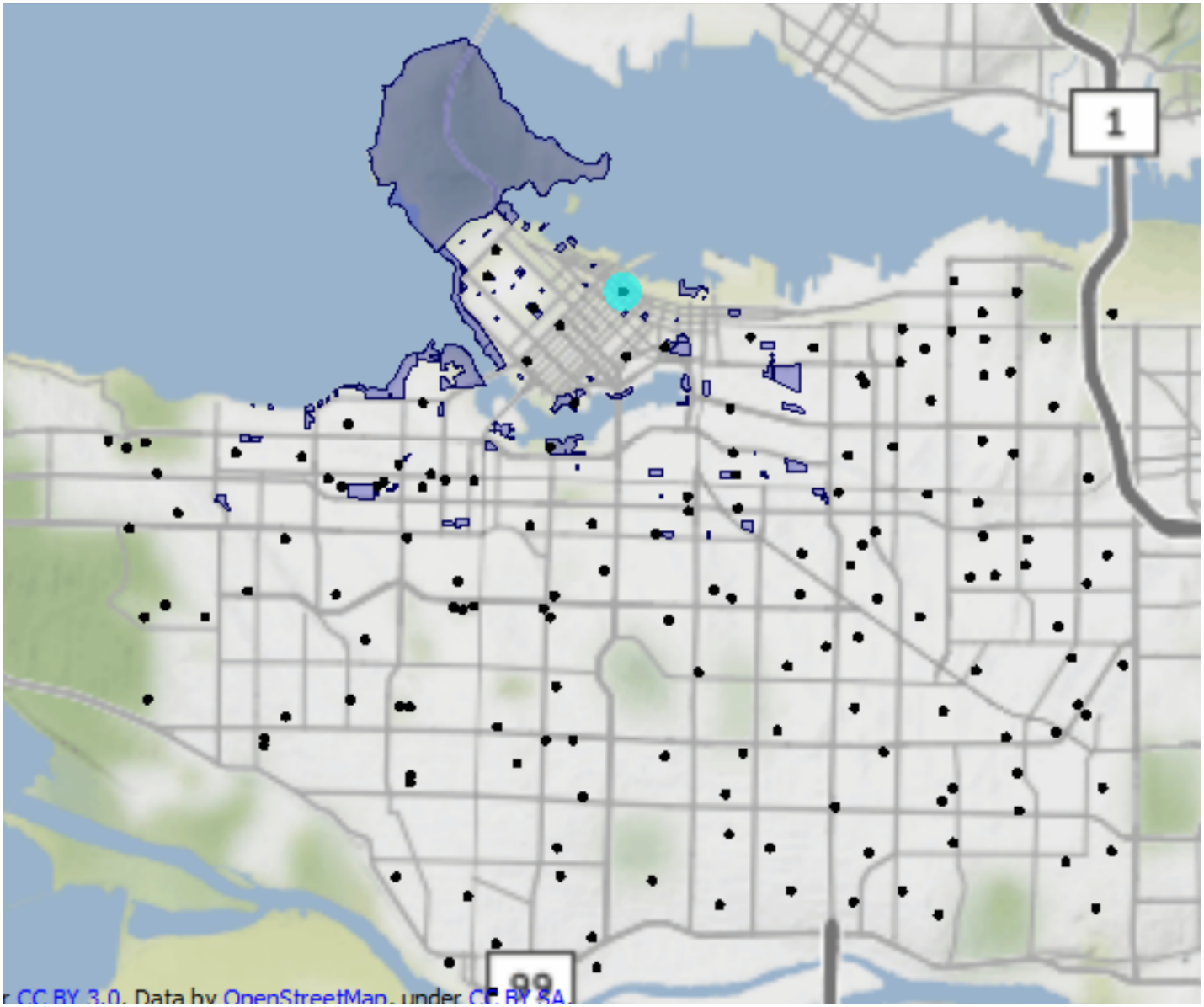


14. Run the Workspace

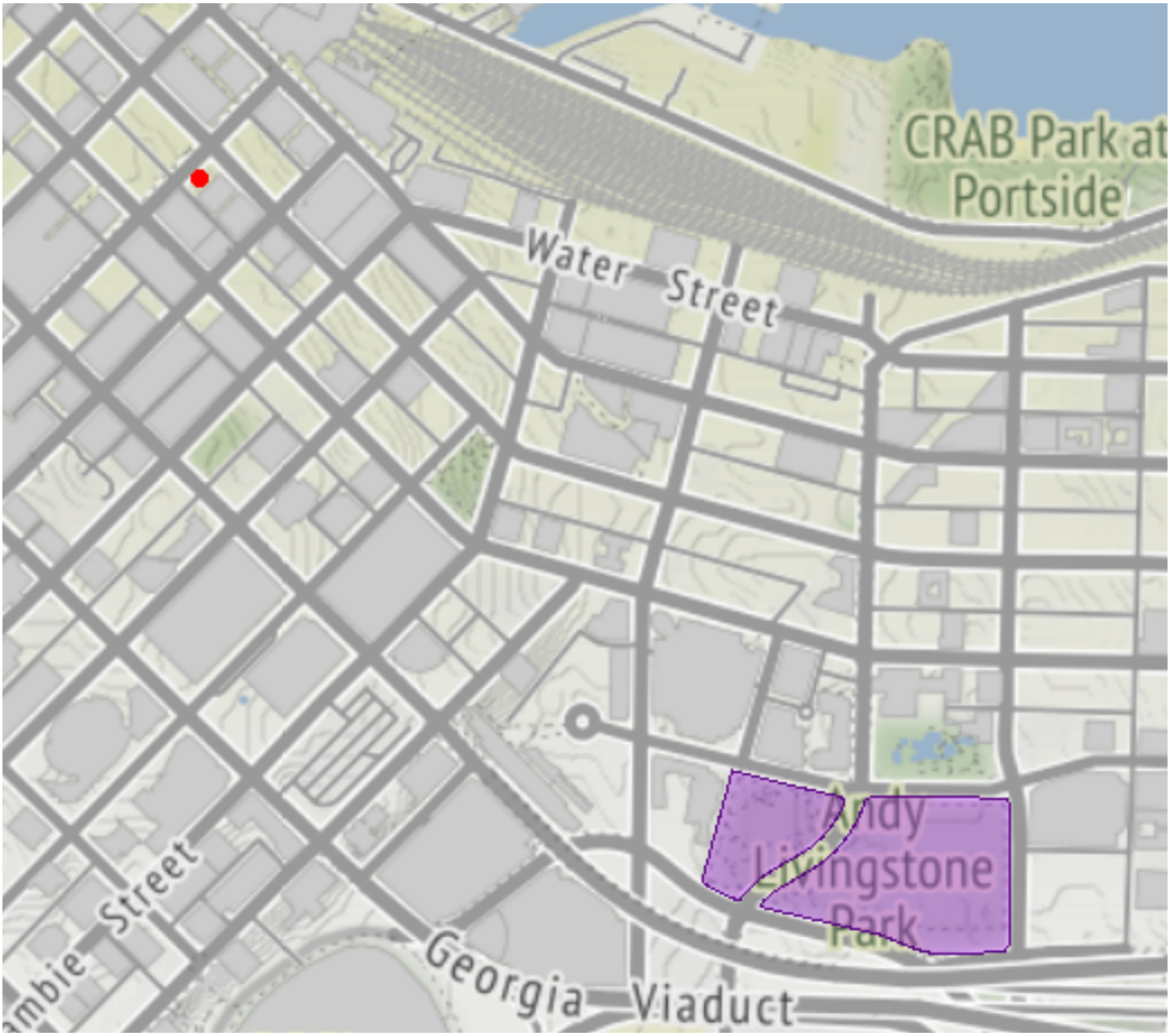
After running the workspace, the output of the nearest park will be displayed in the FME Data Inspector with a point on the map for Alexander Academy for reference. If you wish to save the output, you can either connect a writer to the Sampler:Sampled port or by using the Save Selected Data as in the FME Data Inspector and selecting the format that you want to save the data as (i.e. PostGIS, File Geodatabase, DWG, etc.). For more information on saving data in the FME Data Inspector, see the Documentation.

Results

Input



Output



Transformers

- Inspector- Sends features to the FME Data Inspector for display.
- NeighborFinder- Finds the nearest Candidate feature(s) to each Base feature and merges their attributes onto the Base feature.
- Sampler - Preserves either a total number of features or a sampling of features, depending on the Sampling Type selection
- Sorter- Sorts features by a selected attribute's value.
- Tester- Evaluates one or more tests on a feature, and routes the feature according to the outcome of the test(s).

Data Attribution

Raster Slope Calculations

Raster Slope Calculations

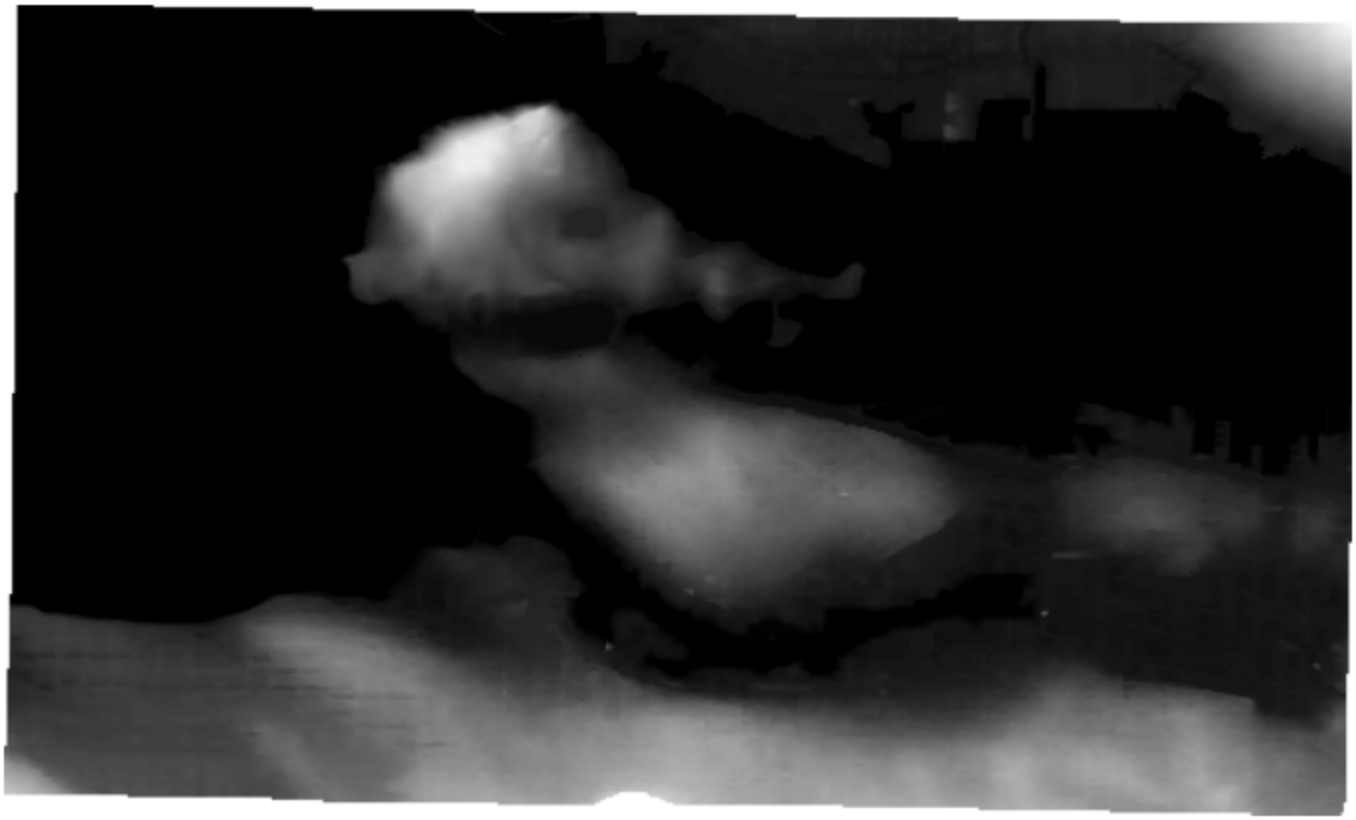
Raster slopes can be measured in two ways: the slope angle and the slope aspect (direction). In FME slope angle is measured with the RasterSlopeCalculator transformer and slope aspect with the RasterAspectCalculator.

In this tutorial we will try each of these transformers to see what they do, and then use the RasterSlopeCalculator in a workspace that assesses land-use capabilities.

Video

Source Data

The source data for these examples is a Raster DEM dataset in CDED format. In the FME Data Inspector it looks like this:



The lighter the color, the higher the elevation. Darker areas represent low-lying areas.

Step-By-Step Instructions (Exercise 1)

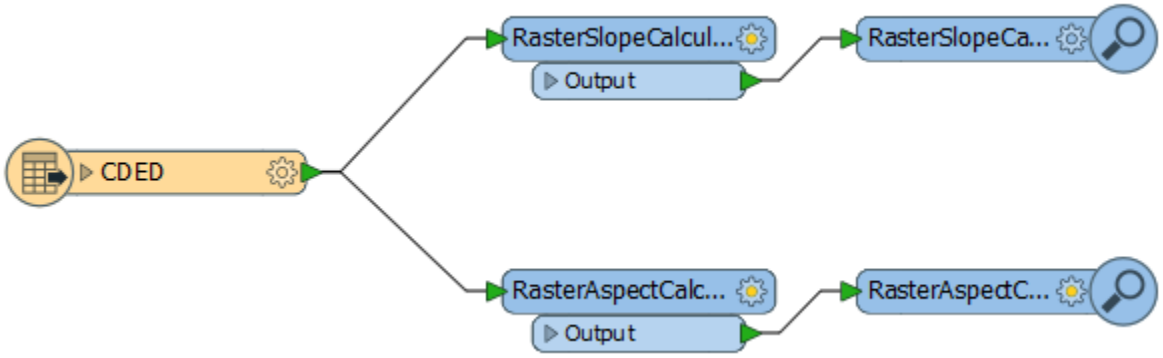
1. Start FME Workbench and begin with an empty canvas. Select Readers > Add Reader from the menubar.

When prompted set the data format to Canadian Digital Elevation Data (CDED) (or U.S. Geological Survey

Digital Elevation Model (USGSDEM) - the two are different names for the same format).

Select the attached data as the source dataset.

2. Place a RasterSlopeCalculator transformer and a RasterAspectCalculator on the canvas. Draw a connection from the reader feature type to each of the transformers, then connect the output from each of the transformers to an Inspector transformer:



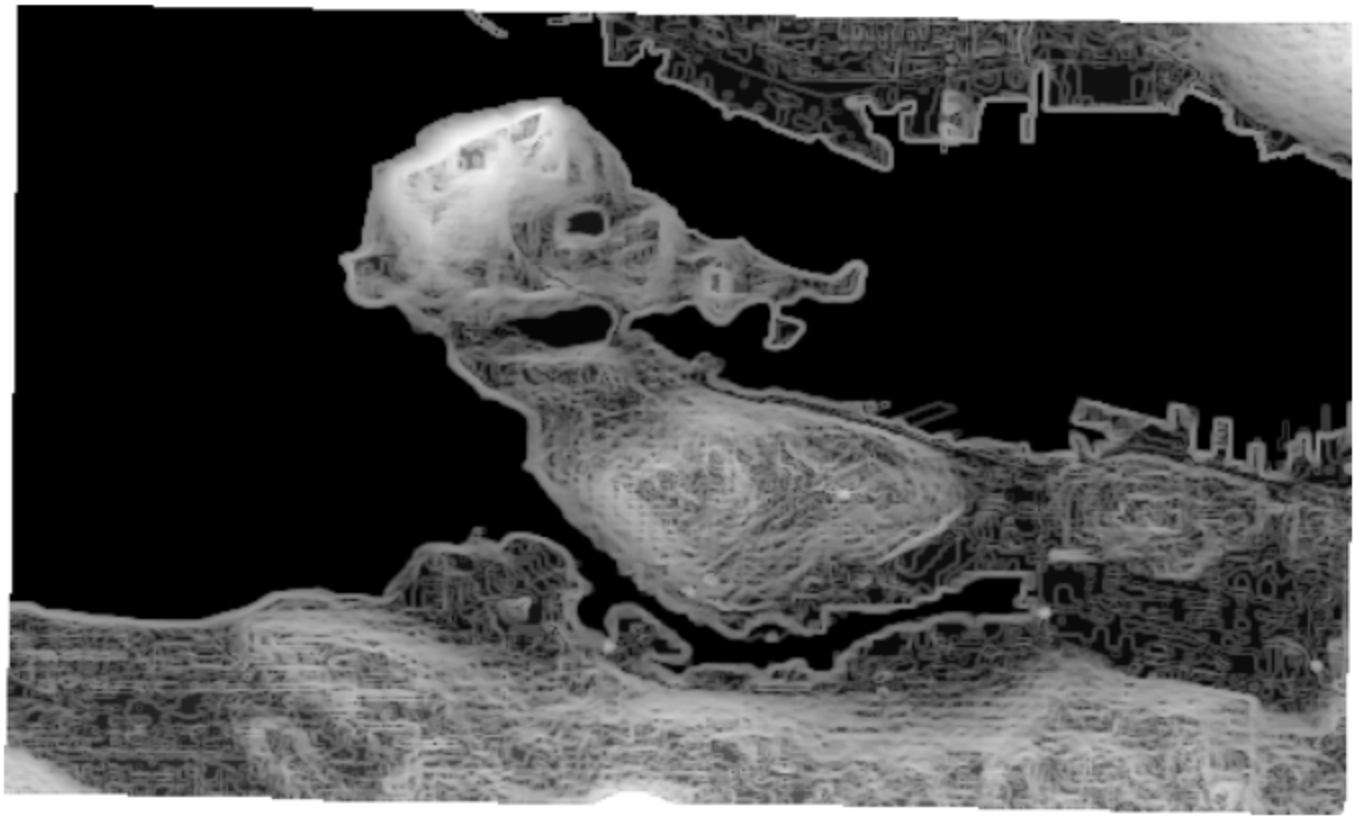
Check the parameters dialogs for each of the raster calculators to see what options are available, but generally leave the parameters as they are.

3. Run the workspace. The result will be two raster features that appear in the FME Data Inspector.

Turn off background maps, if you have them turned on, as they will make it harder to query the data.

Now hide each raster in turn to enable you to inspect the other.

The slope feature looks like this:



Query some of the raster cells by clicking on them. Each cell now represents the slope at that location, rather than elevation. You will be able to pick out bright spots in the centre and right of the image. These are features that - although low-lying - have a steep slope.

The aspect feature looks like this:

Each cell now represents the direction of the slope at that location from 0-360 degrees. FME doesn't visualize this very well, because it interprets the values from zero (very dark) to 360 (very light) but you can identify each cell's value by clicking it.

4. We can (optionally) adjust the slope aspect for a better display using the RasterExpressionEvaluator transformer.

Place a RasterExpressionEvaluator into the workspace between the RasterAspectCalculator and the Inspector it is connected to. Open the parameters dialog. Set interpretation to "Preserve". Double-click the Expression field and use the drop-down arrow to open the arithmetic calculator.

Inside the calculator enter the following expression:

```
if (A[0]>180,@abs(A[0]-360),A[0])
```

This will cause the aspect to be calculated from 0 to 180 both clockwise and anti-clockwise. The result will look like this:

Notice how the coloring is now more consistent, making it easier to see the dark areas that face north.

Step-By-Step Instructions (Exercise 2)

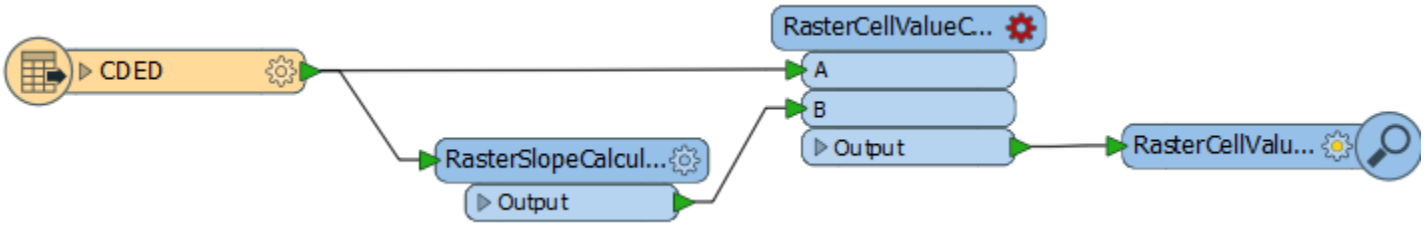
1. This exercise is going to use raster data and the RasterSlopeCalculator to calculate the suitability of land for development. Building developments usually prefer land that is both flat and low-lying.

Start FME Workbench and begin with an empty canvas. Select Readers > Add Reader from the menubar.

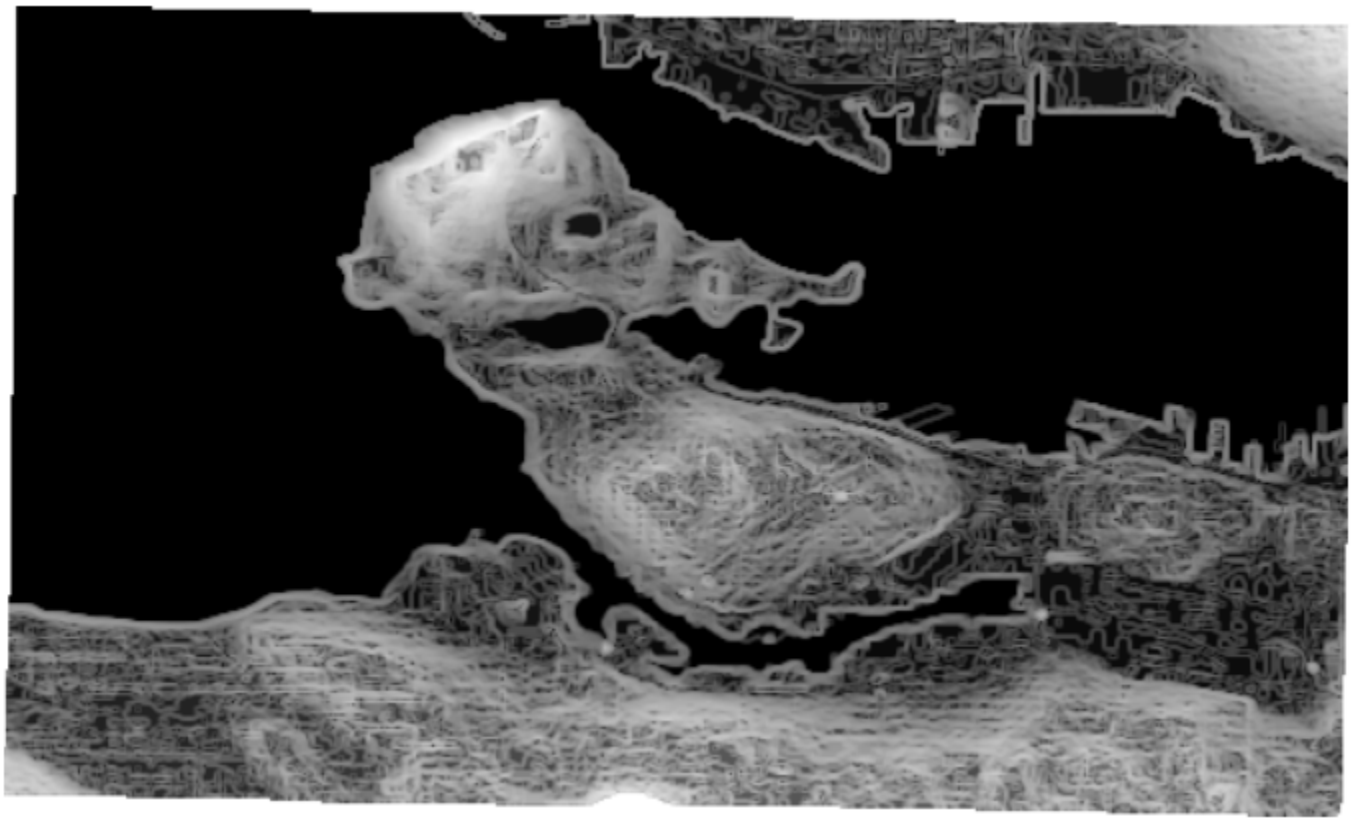
When prompted set the data format to Canadian Digital Elevation Data (CDED) (or U.S. Geological Survey Digital Elevation Model (USGSDEM)) and select the attached data as the source dataset.

2. Place a RasterSlopeCalculator transformer and draw a connection from the reader feature type to the transformer.

3. Place a RasterCellValueCalculator transformer. Draw a second connection from the reader feature type to the RasterCellValueCalculator:A input port. Draw a connection from the RasterSlopeCalculator:Output port to the RasterCellValueCalculator:B input port:



4. Open the parameters dialog for the RasterCellValueCalculator. Simply change the Operation parameter to the + option; i.e. this will add the original data (representing elevation) to the transformed data (representing slope). The result will represent the suitability of each cell for building development:



The lighter colors represent cells that have both a steep slope and a higher elevation. Darker colors represent cells that have both a shallow slope and a low elevation.

Downloads

Source Dataset

Example 1: Completed workspace as a template

Example 2: Completed workspace as a template

Calculate Strahler Stream Order Numbers

Introduction

Stream order can be used to identify and classify stream types based on the number of tributaries connected to it. This tutorial will cover how to prepare your data before using the StreamOrderCalculator transformer to determine stream order based on the Strahler Algorithm. The Strahler algorithm classifies streams without a tributary as order 1.

Downloads

StreamOrderCalculator.fmwt

StreamOrderCalculator-Data.zip

Instructions

Using a Digital Elevation Model of the area of interest allows us to determine the direction of flow of the streams using elevation.

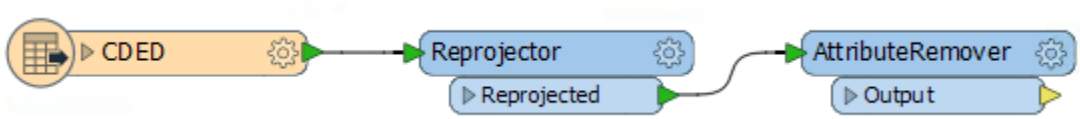
1) Add DEM and reproject

In a blank workspace in FME Workbench add a new Canadian Digital Elevation Data (CDED) Reader to the canvas and browse to the RasterDEM-250K.dem dataset.

Then add a Reprojector transformer to the canvas, set the Destination Coordinate System to TX83-CF and click OK.

2) Clean up data

Next, we will need to remove all of the extra attributes we won't need to help speed up the translation. Add an AttributeRemover transformer and in the parameters click the ellipsis next to Attributes to Remove then click Select All.



Workspace with CDED Reader, Reprojector, and an AttributeRemover

3) Add Streams data

Add another Reader to the canvas, this time set the Format to MapInfo MIF/MID and browse to the HydrographyLine.mif dataset.

4) Clean up lakes

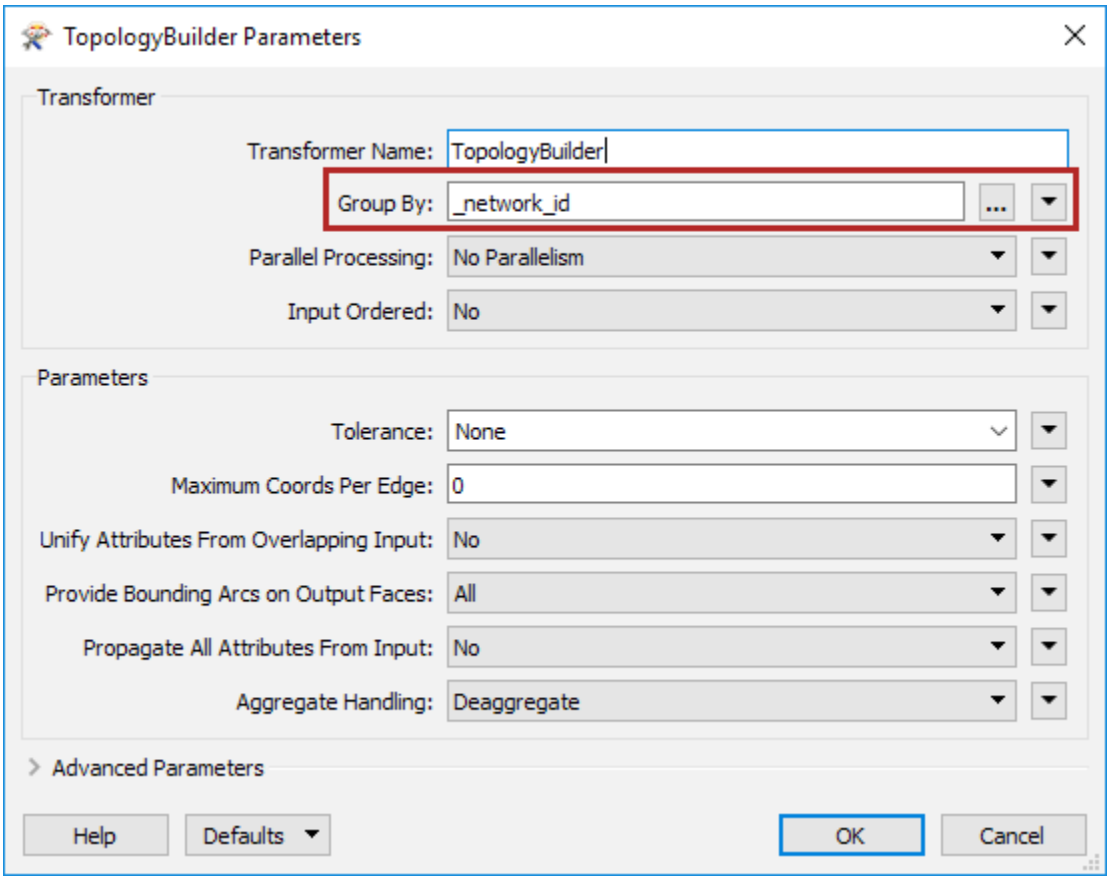
Since we are only interested in streams, we will need to remove the line data that creates lakes or other bodies of water. Add an AreaBuilder transformer to the canvas and connect it to the HydrographyLines feature type. In the parameters, set the Accumulation Mode which is under Attribute Accumulation to Drop Incoming Features.

5) Generate network IDs

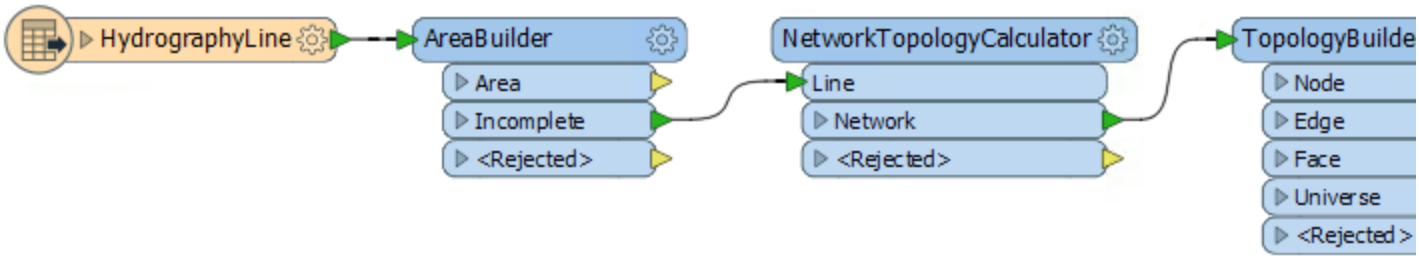
We will need to create a unique ID for each of the streams, to do this we will use the NetworkTopologyCalculator transformer which will assign a _network_id attribute to all of the lines which participate in a network. Add the NetworkTopologyCalculator to the canvas and connect it to the Incomplete output port on the AreaBuilder.

6) Create topology

Now the TopologyBuilder transformer can be used to build topology on the lines within a particular network. Here we are interested in the nodes so that we can determine the lowest node in a network to find the direction of flow of the stream. Add a TopologyBuilder to the canvas and set the Group By parameter to _network_id.



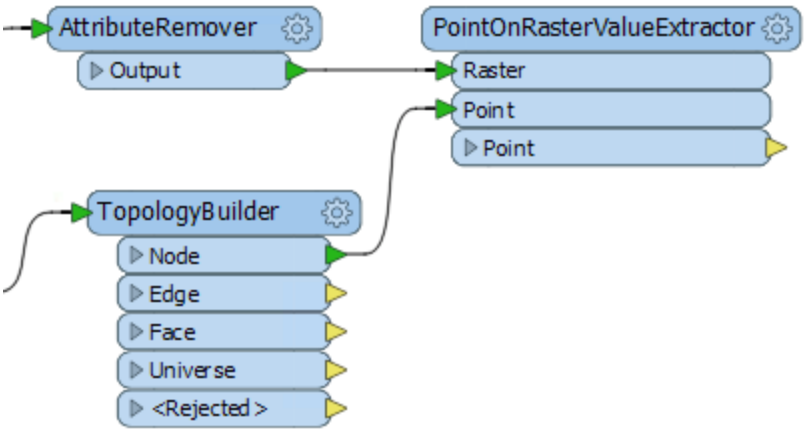
TopologyBuilder parameters set the Group By to _network_id



MIF Reader with an AreaBuilder, NetworkTopologyCalculator and a TopologyBuilder

7) Find elevation values

Now that we have joined all our lines based on _network_id using the TopologyBuilder we can determine the elevation for each node within the line. Add a PointOnRasterValueExtractor transformer to the canvas. Connect the Output port on the AttributeRemover to the Raster input port on the PointOnRasterValueExtractor. Then connect the Nodes output port on the TopologyBuilder to the Point input port. This will overlay the stream data on the raster dem and then extract the elevation which will be the attribute _band{0}.value, which is the elevation.

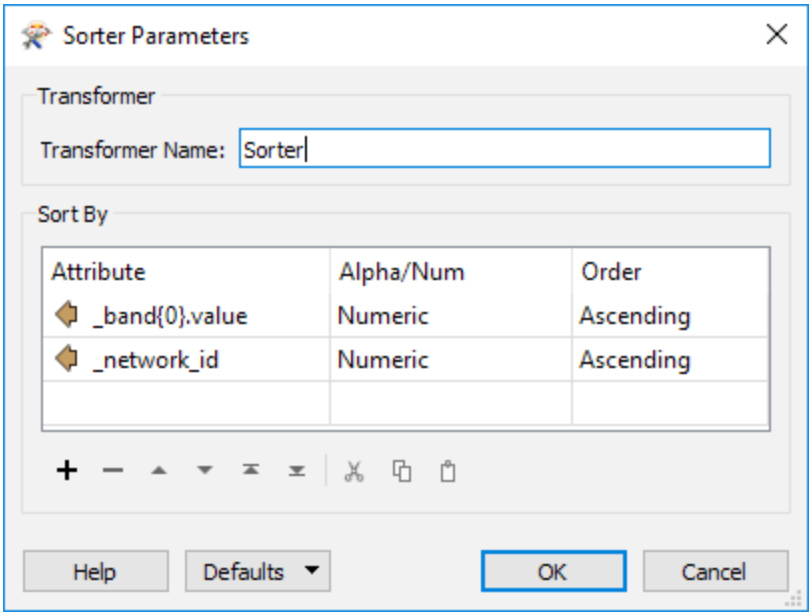


Connect the AttributeRemover to the Raster input port and the TopologyBuilder to the Point input port

8) Sort attribute values

Add a Sorter transformer to the canvas and connect it to the Point output port on the PointOnRasterValueExtractor. In the parameters set the following parameters:

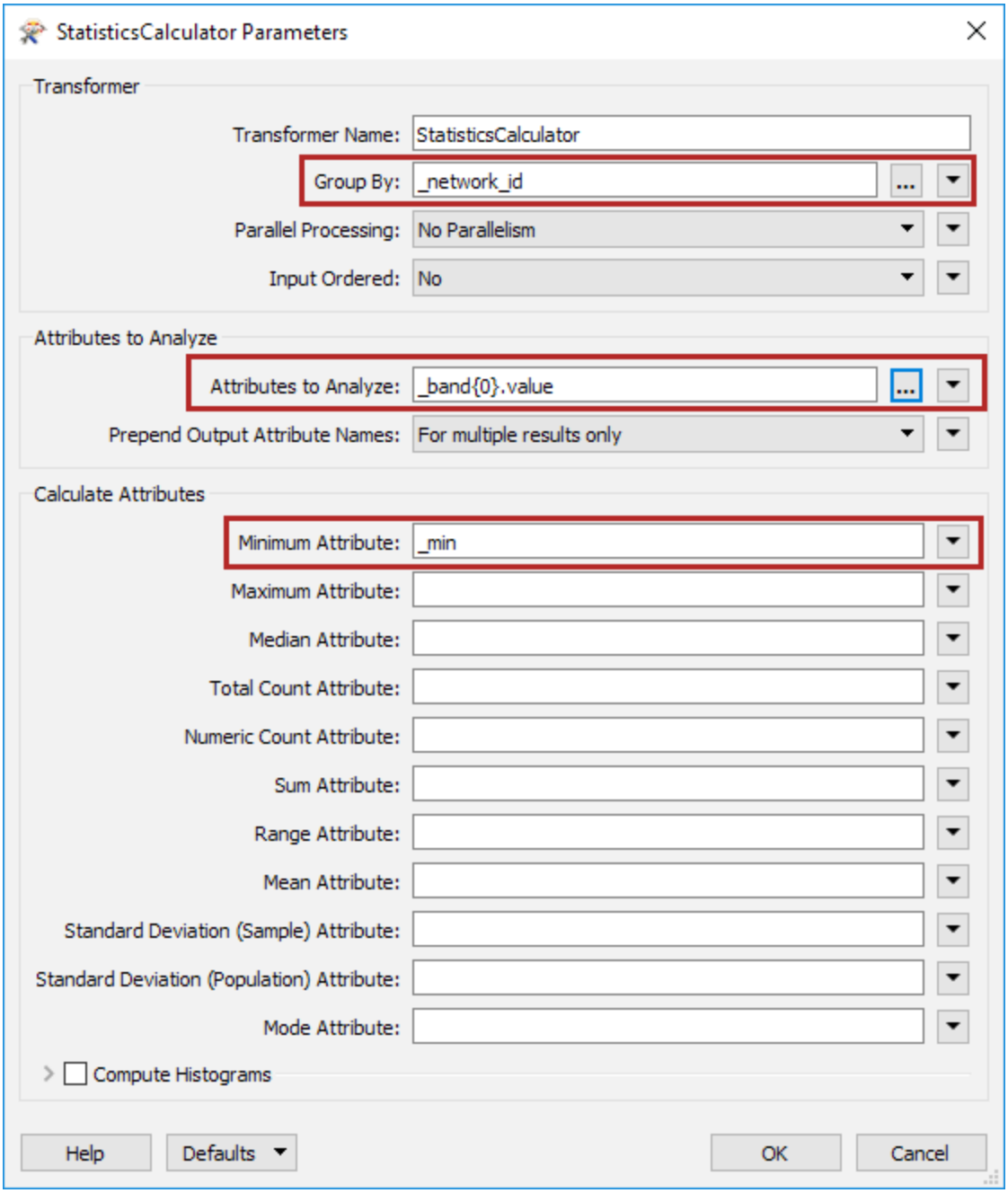
Attribute	Alpha/Num	Order
_band{0}.value	Numeric	Ascending
_network_id	Numeric	Ascending



Sorter parameters, sort both _band{0}.value and _network_id by ascending

9) Calculate minimum and standard deviation

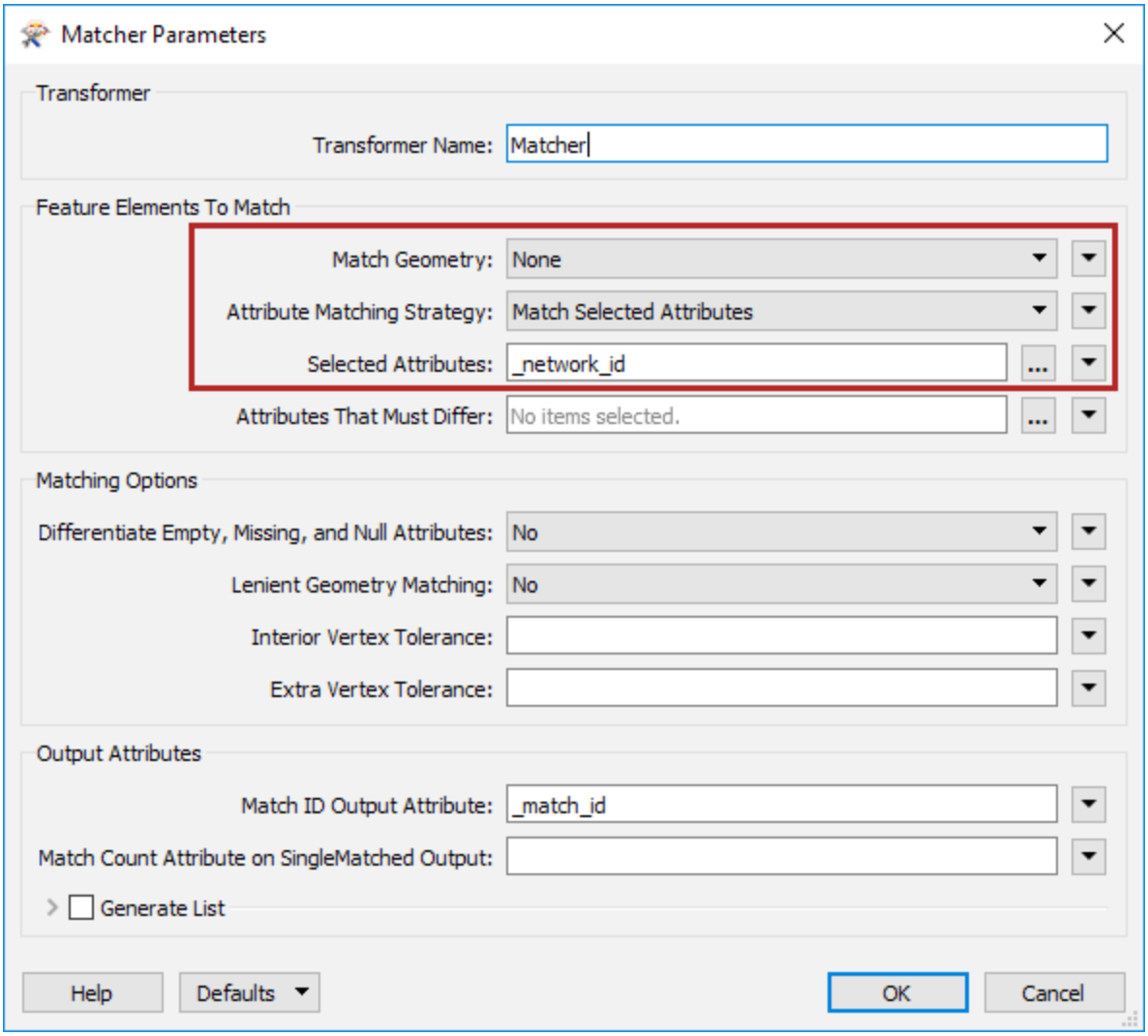
With the attribute sorted we can calculate the minimum and standard deviation for the elevation. Add a StatisticsCalculator transformer to the canvas and connect it to the Sorter. In the parameters, set the Group By to _network_id and the Attributes to Analyze to _band{0}.value. Next, remove all of the attribute names for the Calculate Attributes except _min.



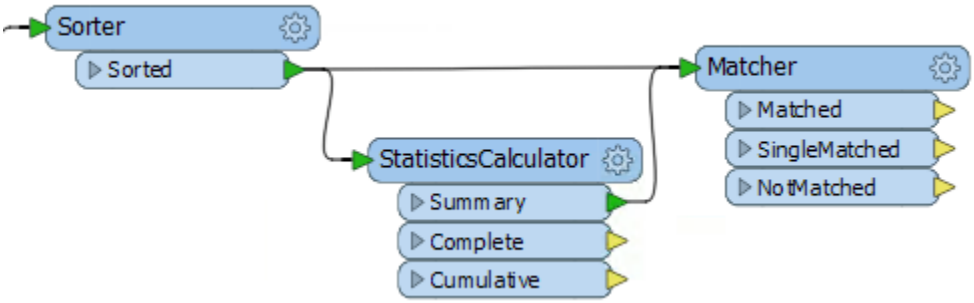
StatisticsCalculator parameters. Set the Group By to _network_id and the Attributes to Analyze to _band{0}.value

10) Find the node with the lowest elevation

To find the node with the lowest elevation, we will now use the Matcher transformer to match the attributes from the Sorter and the StatisticsCalculator based on _network_id. Add a Matcher to the canvas and connect it to both the Sorter and the Summary output port on the StatisticsCalculator. In the parameters, set the Match Geometry to None and the Selected Attributes to _network_id.



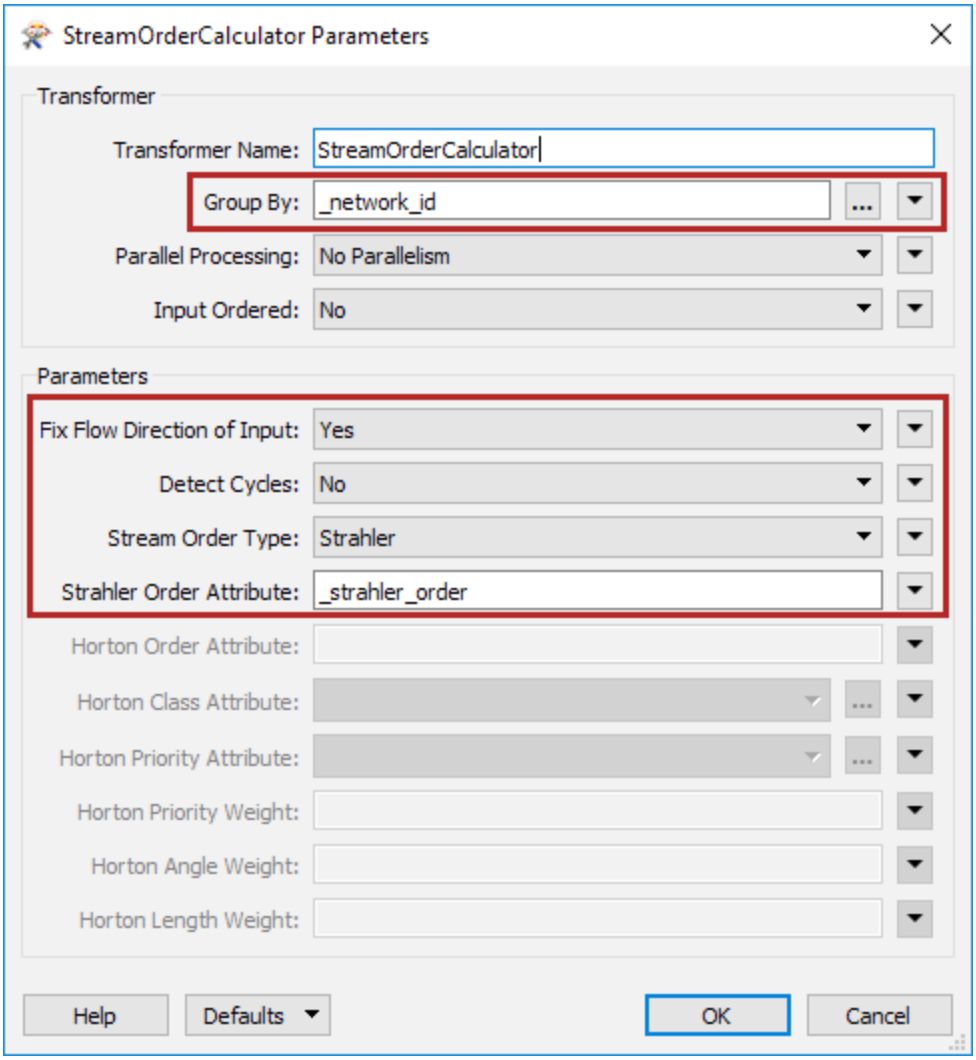
Matcher parameters; set the Match Geometry to None and the Selected Attributes to _network_id



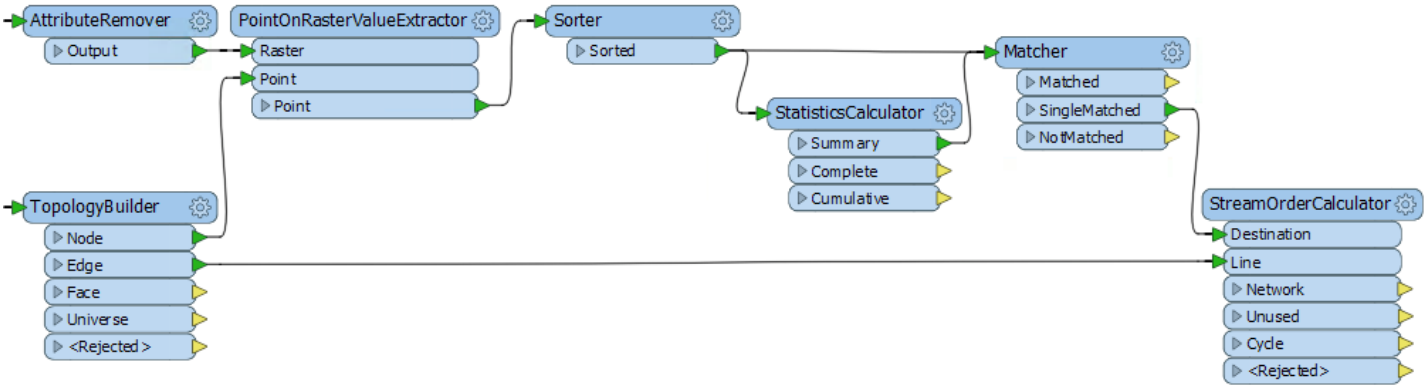
Connect the Matcher to both the Sorter and the Summary output port on the StatisticsCalculator

11) Calculate Stream Order

Using the StreamOrderCalculator transformer, we will calculate the stream order based on the Strahler algorithm. Add a StreamOrderCalculator to the canvas and connect the Line input port to the Edge output port on the TopologyBuilder and then connect the Destination input port to the SingleMatched output port on the Matcher. In the parameters, set the Group By to _network_id then set the Fix Flow Direction of Input to Yes. Finally, set the Stream Order Type to Strahler.



StreamOrderCalculator parameters; set the Group By to _network_id and the Stream Order Type to Strahler

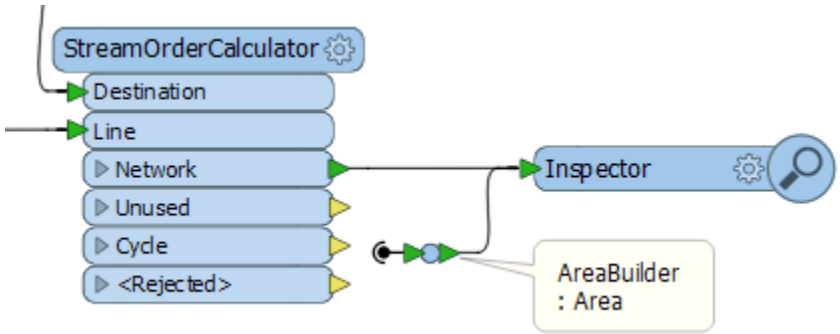


Connect the StreamOrderCalculator to the Edge output port on the TopologyBuilder and the SingleMatched output port on the Matcher

12) Inspect the output

Add an Inspector transformer to the canvas and connect it to the Network output port on the StreamOrderCalculator. Also, connect the same Inspector to the Area output port on the AreaBuilder, this

will add the lakes back in when we inspect the data. You can right click on the connection to the AreaBuilder and select Create a Tunnel to clean up the canvas.

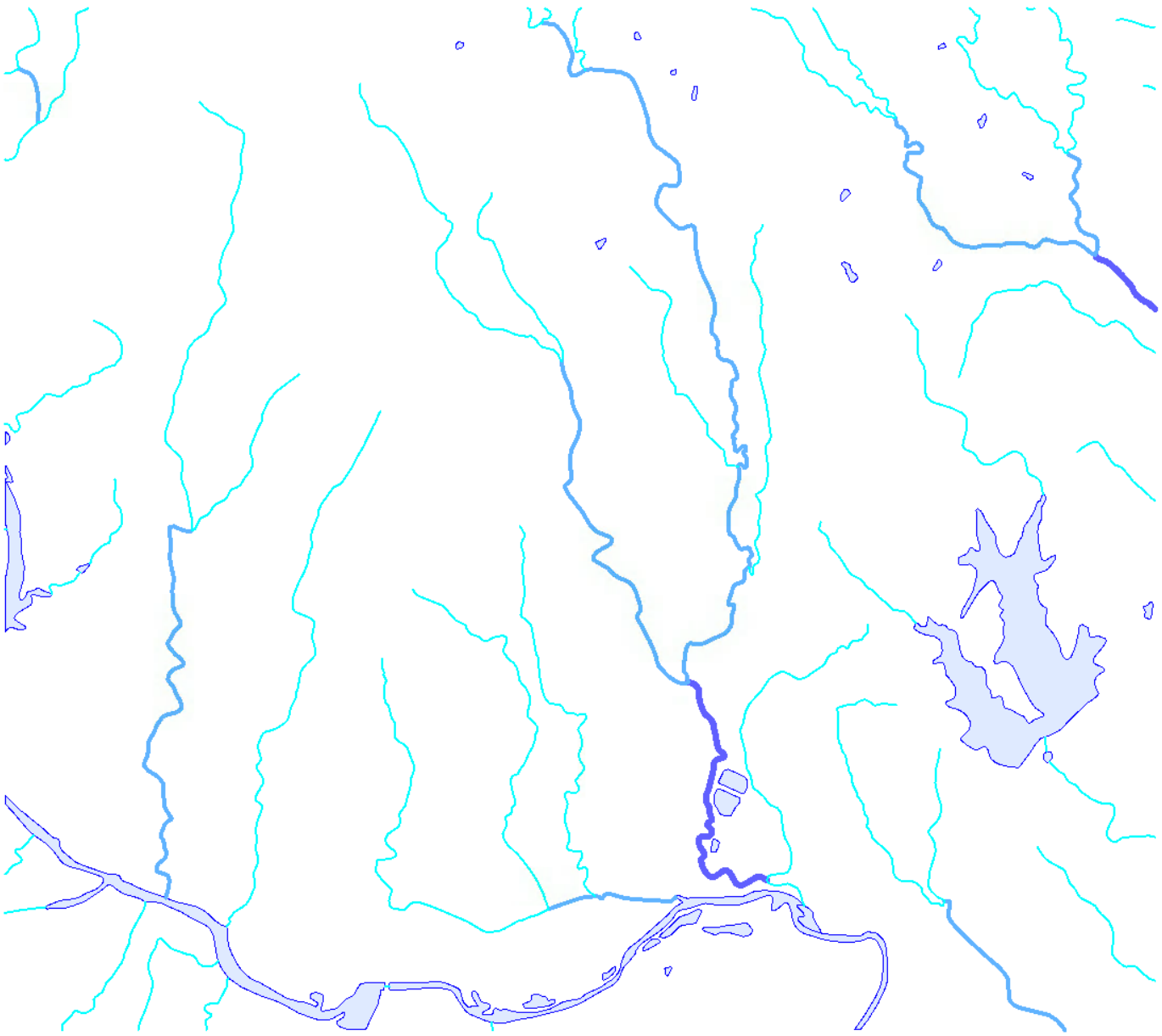


Connect the Inspector to the Area output port on the AreaBuilder and the Network output port on the StreamOrderCalculator

In the parameters of the Inspector set the Group By to _strahler_order, this is the stream order that was calculated.

Run the translation. In the Data Inspector, we will need to change the colors and weights of the lines to view our data. In the Display Control panel, set the Inspector_(29) to a blue fill color with the Fill Opacity set to 0.2, then move it to the bottom of the list, this is the lakes.

Next set the Inspector_1(46) to a light blue color and a Width of 2. Then set the Inspector_2(12) to a blue color and a Width of 4. Finally, set the Inspector_3(2) to a dark blue color with a Width of 6. Now you can see the order of the streams.



Final output classified and colored by strahler_order

One remaining issue is that if the stream segment is relatively flat so that the elevation of each end of the stream is the same, then the workspace may not correctly identify the low point of the stream. This happens to the small network in the upper left-hand corner of this image. Two low points (nodes) are passed into the StreamOrderCalculator, and it can only make use of one of them. It may not be the correct one, so additional logic may be required to resolve this issue.

Data Attribution

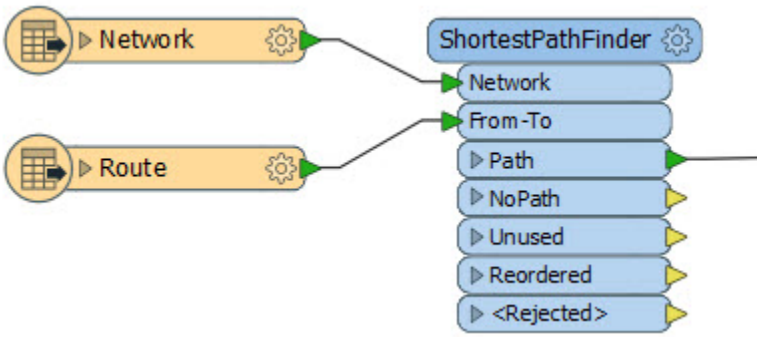
Data provided by the City of Austin and Travis County, Texas

Shortest Route Calculations with the ShortestPathFinder

Shortest Route Calculations

Shortest route calculations are done in FME using the ShortestPathFinder transformer. This transformer calculates the shortest path from a source node to a destination node on a given network.

Here, for example, a user is finding the shortest path between the start/end points of a given route, using a network of lines:



A from-to line defines the start and end of the route. Any number of from-to lines can be passed into the ShortestPathFinder for multiple calculations on the same network.

There are some simple variations and issues to be aware of.

Variations

- A path can be weighted by an attribute other than length; for example each segment of the network can be given a "cost" related to (for example) speed limits or vehicle weight limits.
- The from-to line defines the start and end of the route, but can also contain intermediate points that the route must pass through.
- Implemented as the engine to a web service, the from-to points could be defined by a user in a web mapping tool.

Cautions

- The From-To input port accepts line features, not point features.
- The points on a from-to line should match a coordinate on the network. If they do not, then a parameter is available to set a permitted tolerance.

Example 1: Routes by Length

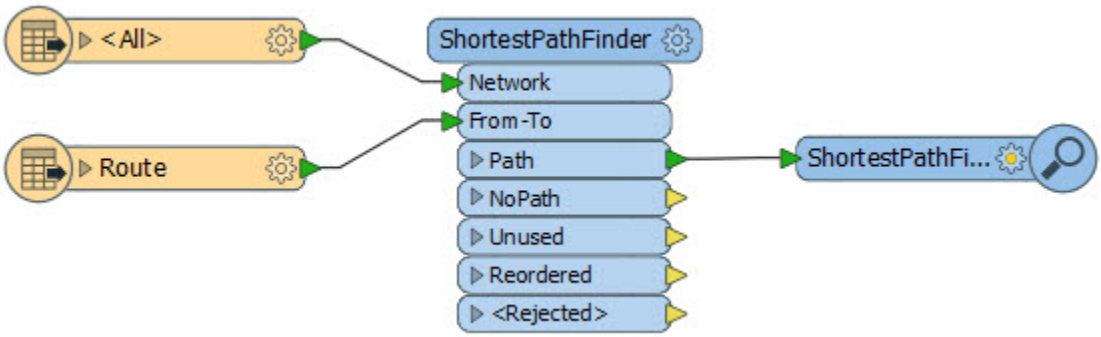
Follow these steps as an example of how to calculate the shortest route in a network.

1. Start FME Workbench. Add a reader (Readers > Add Reader on the menubar) to read the attached Shape

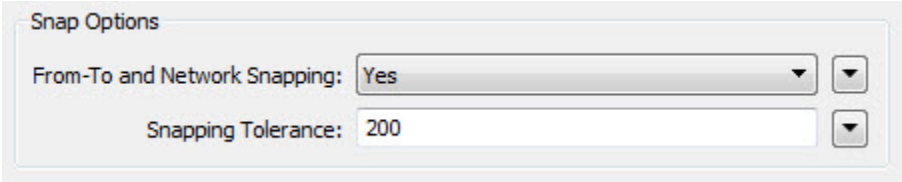
dataset of the from-to line.

2. Add a second reader to read the attached AutoCAD DWG dataset containing a road network. In the Add Reader dialog click the option for "Single Merged Feature Type". This will ensure all data is added as a single layer:

3. Add a ShortestPathFinder transformer and connect the road network and from-to line to the correct input ports. Add an Inspector transformer to visualize the output:



4. Open the parameters dialog for the ShortestPathFinder. Because the from-to end points may not sit exactly on the network, change the parameter From-To and Network Snapping to Yes. Set a tolerance of 200:



5. Run the workspace. The output will look like this:

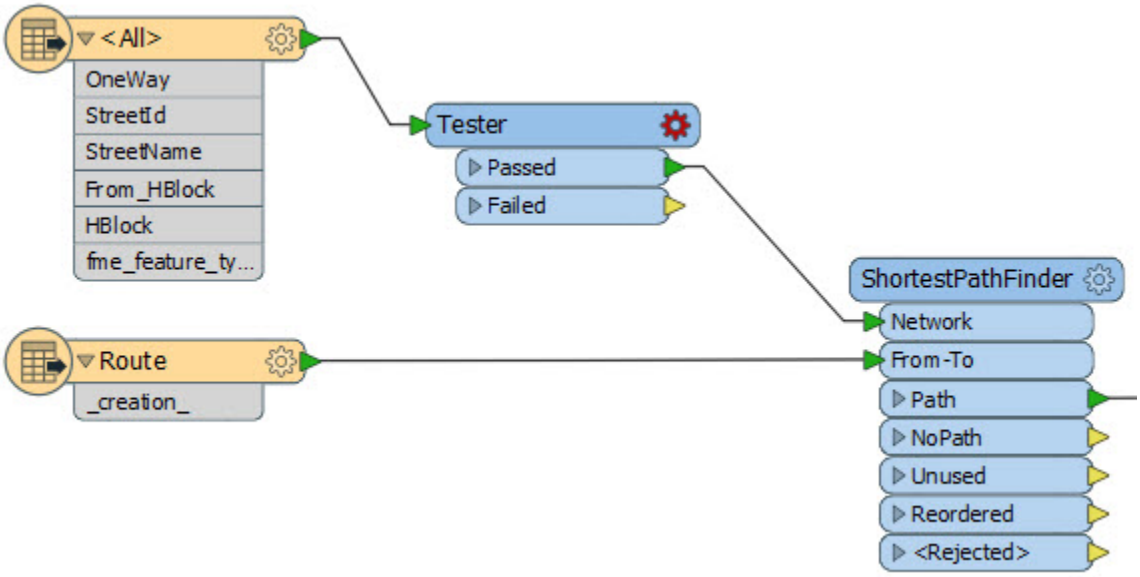
Example 2: Routes by Cost

Travelling the wrong way along a one-way street is avoided by using costs instead of distance. It is necessary for the one-way streets to be tagged (usually with an attribute) and for their direction (from the first coordinate to the last) to match the permitted direction of travel.

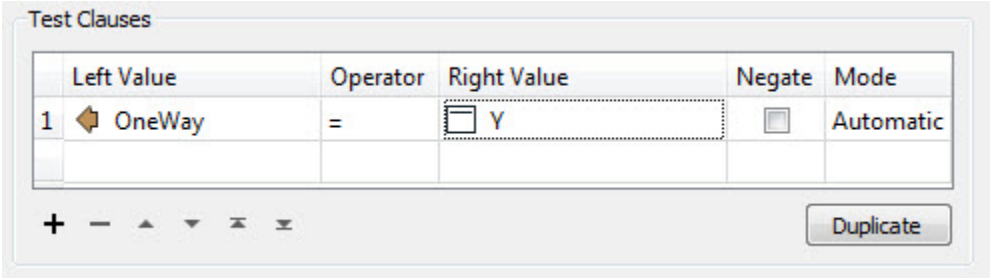
Follow these steps as an example of how to calculate the shortest route in a network and avoid travelling the wrong way along a one-way street.

1. Start FME Workbench. Open the attached workspace (or continue from example 1).

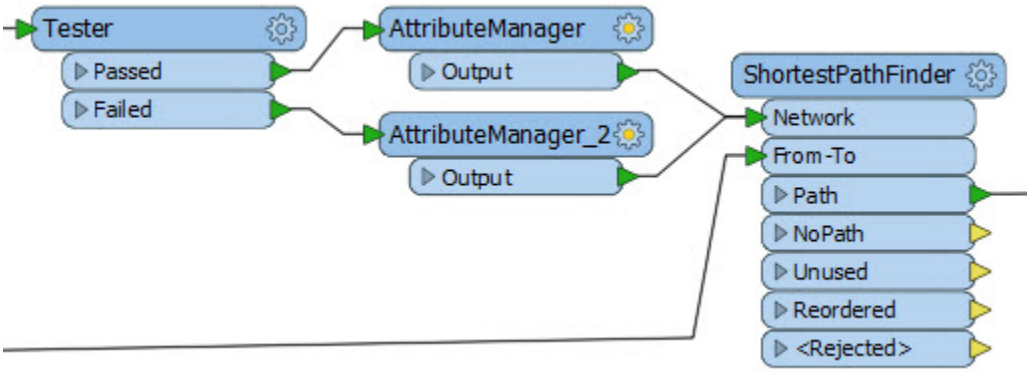
2. Add a Tester transformer to the workspace, between the streets feature type and the ShortestPathFinder transformer:



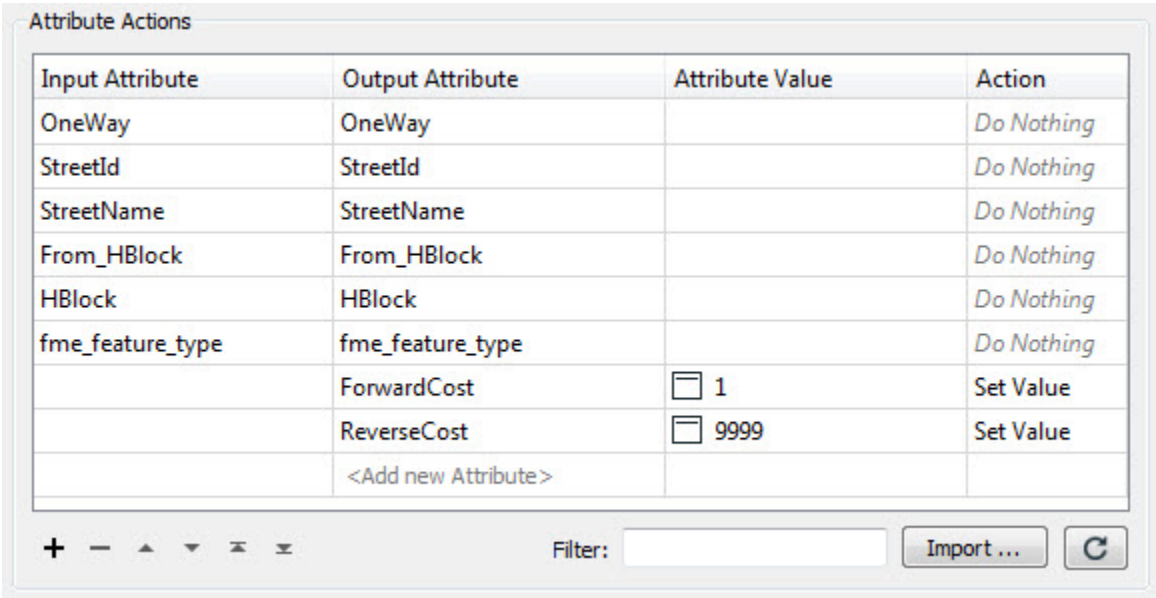
3. Open the Tester parameters dialog. Set up a test for where the attribute OneWay equals Y



4. Add two AttributeManager transformers, one connected to each of the Tester output ports. These will be used to set the travel cost:



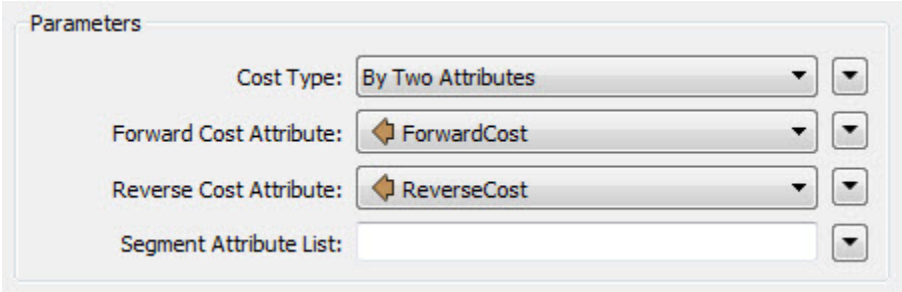
5. Open the parameters dialog for the first AttributeManager (the one connected to the Tester:Passed port). Add two new attributes, ForwardCost and ReverseCost. Set ForwardCost to a value of 1 (representing the correct direction on a one-way street) and the ReverseCost to a value of 9999 (representing the incorrect direction on a one-way street)



6. Open the parameters dialog for the second AttributeManager (the one connected to the Tester:Failed port). Add two new attributes, ForwardCost and ReverseCost. Set both attributes to a value of 1. These are two-way streets where the cost is equal in both directions.

7. Open the parameters dialog for the ShortestPathFinder transformer. Change the Cost Type parameter from

"By Length" to "By Two Attributes". Set ForwardCost as the Forward Cost Attribute and ReverseCost as the Reverse Cost Attribute



8. Run the workspace. The output will look like this:

Notice the difference between this output and example 1. Example 1 was using one-way streets incorrectly, whereas example 2 avoids doing so.

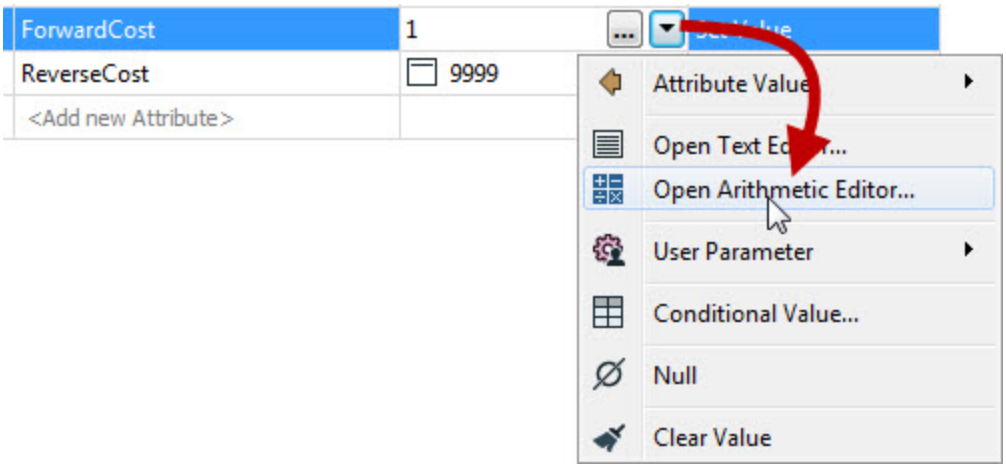
Example 3: Routes by Length AND Cost

Using only cost attributes to determine a route means that distance is no longer considered. For instance, in example 2 there is no difference between a road of 100 metres and 100 kilometres, provided it is not a one-way street. To solve that requires use of both length AND cost.

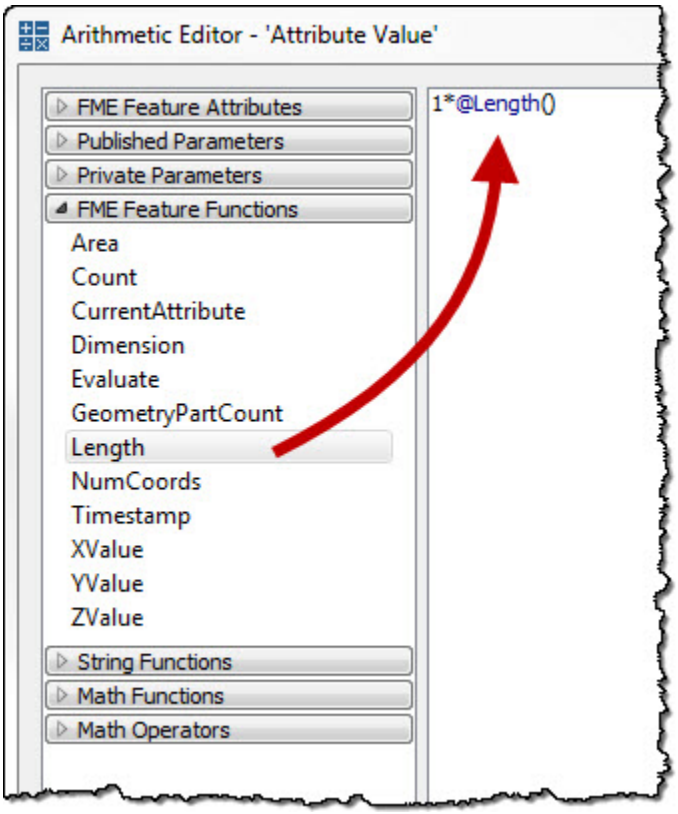
Follow these steps as an example of how to calculate the shortest route in a network, applying both costs and length.

1. Start FME Workbench. Open the attached workspace (or continue from example 2).

2. Open the parameters dialog for the first AttributeManager transformer. Next to the value for the ForwardCost attribute, click the drop-down arrow and select Open Arithmetic Editor



3. In the arithmetic editor, add a multiplication symbol and then drag in the FME Feature Function called Length



Repeat this step for the ReverseCost attribute, and repeat again for both cost attributes in the second AttributeManager.

4. Run the workspace. The output will look like this:

Notice again the difference in result from examples 1 and 2.

NB: *The method of multiplying cost by distance is only one example of what might be used. You could, for instance, weight the costs higher by multiplying cost by half the distance (cost * (distance/2))*

Downloads

Example 1 - Workspace as Template

Example 2 (Begin) - Workspace as Template

Example 2 (Complete) - Workspace as Template

Example 3 (Begin) - Workspace as Template.fmwt

Example 3 (Complete) - Workspace as Template

Creating Time and Distance Isolines using the NetworkCostCalculator

Time and Distance Isolines

Isolines are lines depicting constant values; for example a contour is an isoline that depicts a line of constant elevation.

Isodistances and **isochrones** are a form of isoline. Isodistances depict a line of equal distance from an origin point, and isochrones depict a line of equal time. In many cases they are used to depict travel distances and travel times.

FME is capable of creating isodistances and isochrones using the NetworkCostCalculator transformer combined with the ContourGenerator transformer.

Video

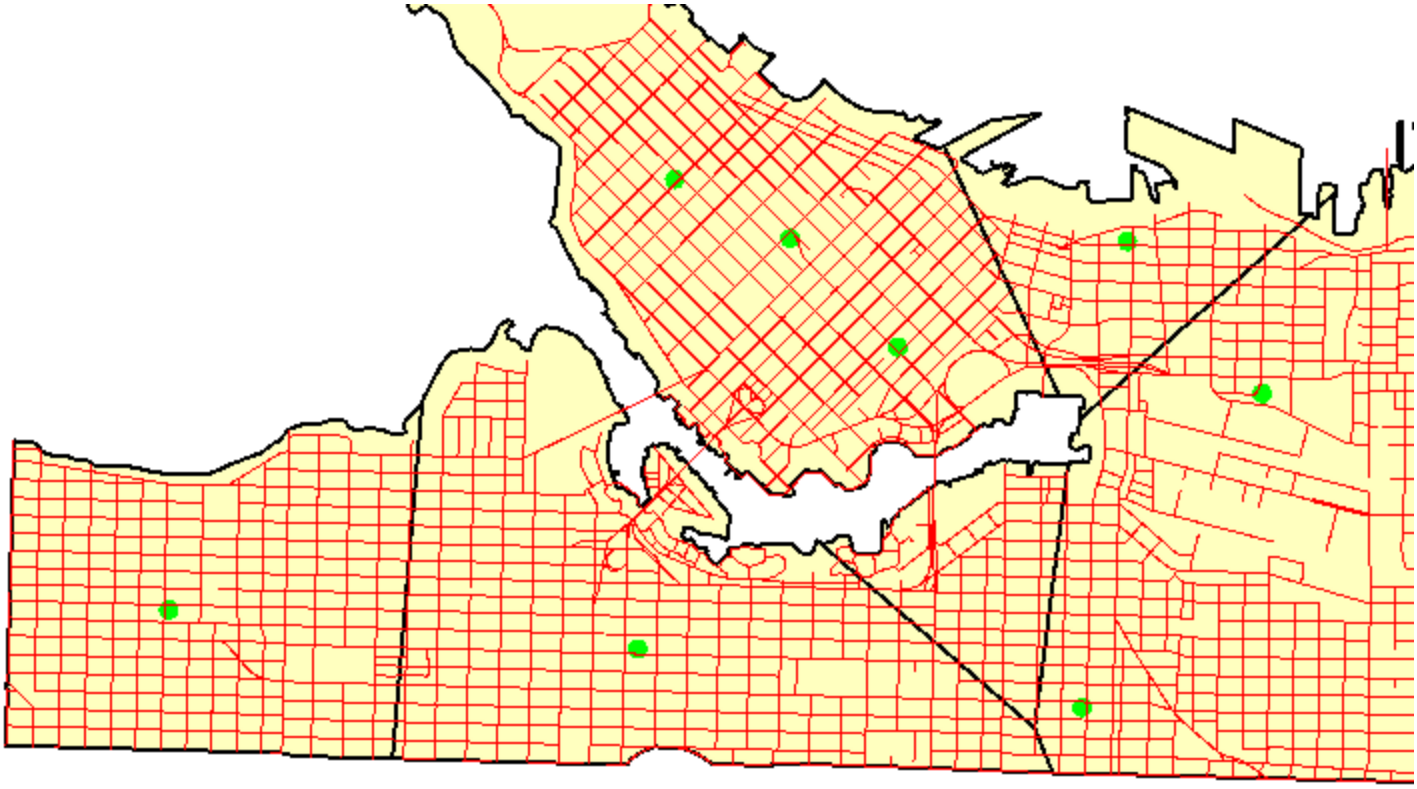
This video demonstrates the example as it was in FME2016. The article has since been updated with a newer dataset and using FME2018.

Source Data

The first source dataset for this example is a set of road features (in an AutoCAD DWG dataset) representing the road network available for travel.

The second dataset (in GML format) is a collection of firehalls and their area of responsibility.

The datasets look like this in the FME Data Inspector:



The scenario here is that we wish to calculate travel times for a particular firehall, to ensure that all properties within the area of responsibility can be provided with emergency medical coverage within a 4 minute time period.

As with most FME translations, only part of the authoring process involves calculating the travel times, with the remainder of the process involving setting up the source datasets correctly.

Step-By-Step Instructions

1. Start FME Workbench and begin with an empty canvas.

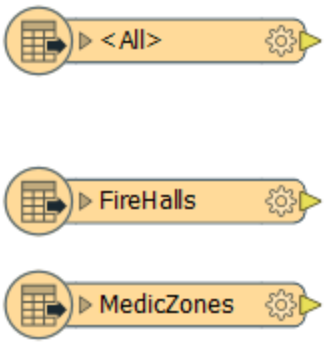
Select Readers > Add Reader from the menubar.

Set the data format to Autodesk AutoCAD DWG/DXF. Select the attached dwg file as the source dataset. Set the Workflow Options parameter to "Single Merged Feature Type".

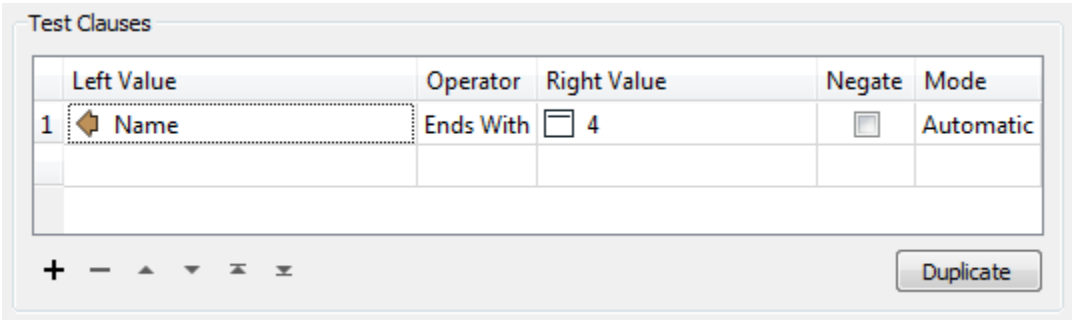
2. Again, select Readers > Add Reader from the menubar.

This time set the data format to GML (Geography Markup Language). Select the attached GML dataset as the source. When prompted select only the FireHall feature type and the MedicZones feature type. Make sure Workflow Options is set back to "Individual Feature Types".

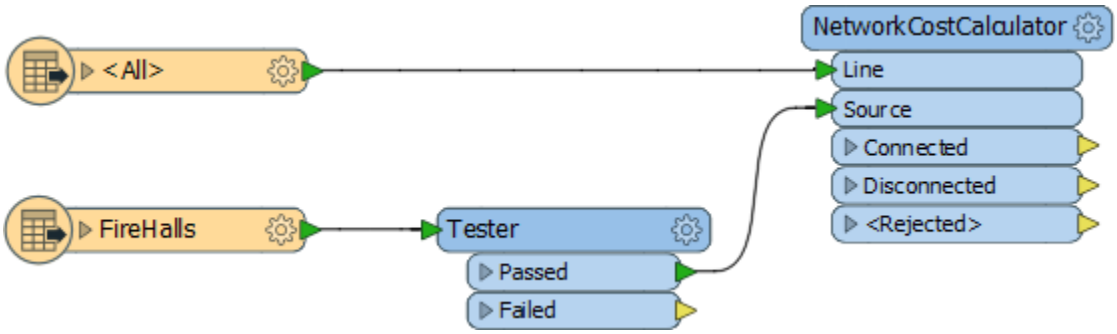
The workspace will now look like this:



3.Add a Tester transformer connected to the FireHalls feature type. Open the parameters dialog and set it up to keep only the #4 firehall. The best solution is to test where the hall name ends in the number 4:



4. Add a NetworkCostCalculator transformer. Connect the Tester:Passed port to the NetworkCostCalculator:Source input port, and the AutoCAD Roads feature type (labelled <All>) to the NetworkCostCalculator:Line input port:

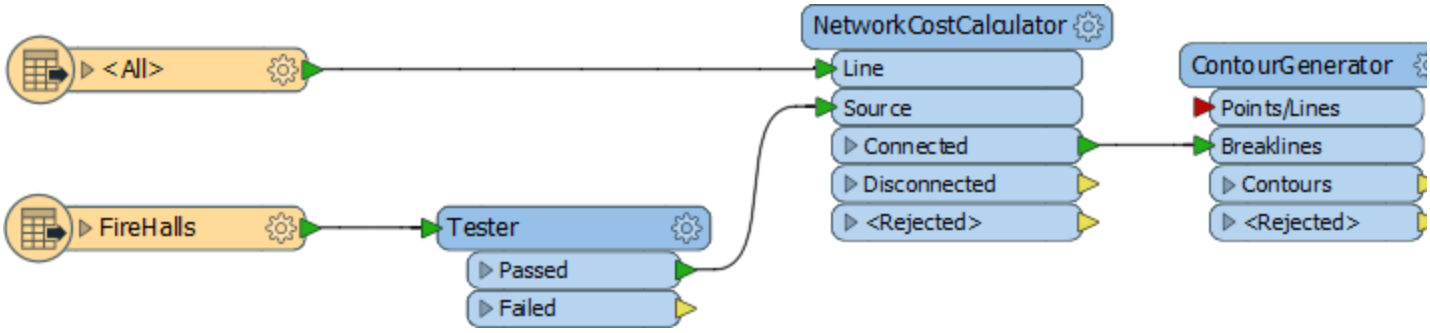


5. Open the NetworkCostCalculator parameters dialog. Set Output Optimal Cost As to "Z-Values". Set Snap Source Points to "Yes" and set a Snapping Tolerance of 100.

These parameters will output the costs as Z values on each network line, and will snap the source firehall to the network if it doesn't sit exactly upon it.

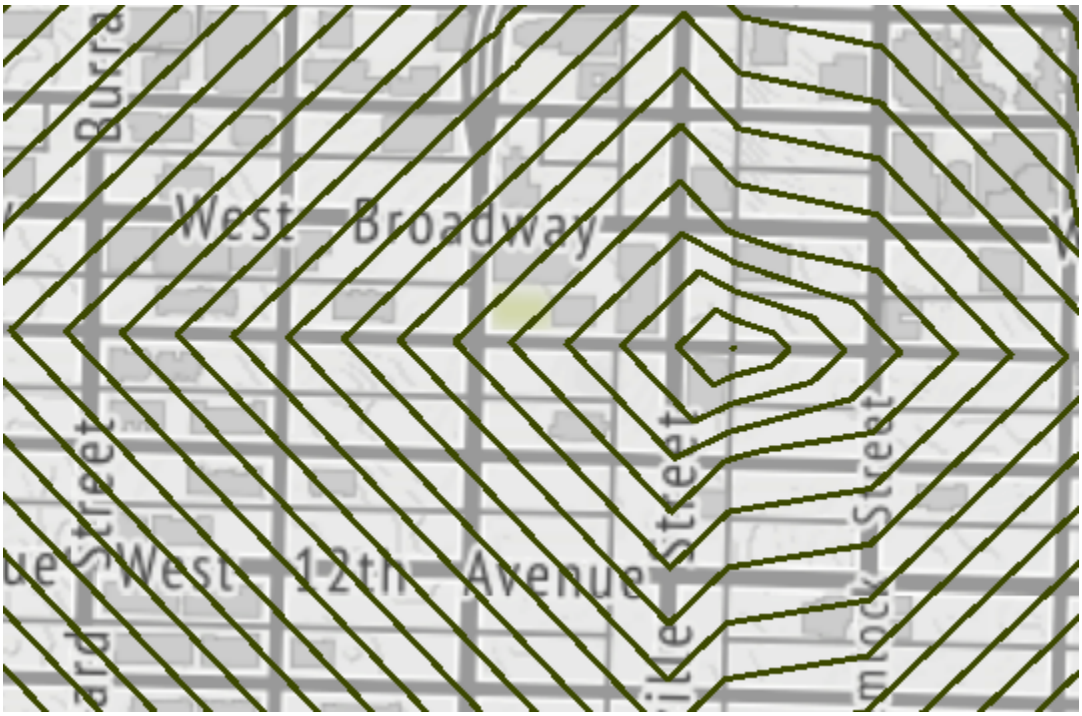
If you wish you can run the workspace now and inspect the output from the transformer. What you will see is the road network where the Z (elevation) of each node is set as the distance from the firehall.

6. To visualize these results, add a ContourGenerator transformer connected to the NetworkCostCalculator:Connected output port. The ContourGenerator port to connect is Breaklines:



7. Open the ContourGenerator parameters dialog. Set the Surface Tolerance to 10 and the Output Contour Interval to 50.

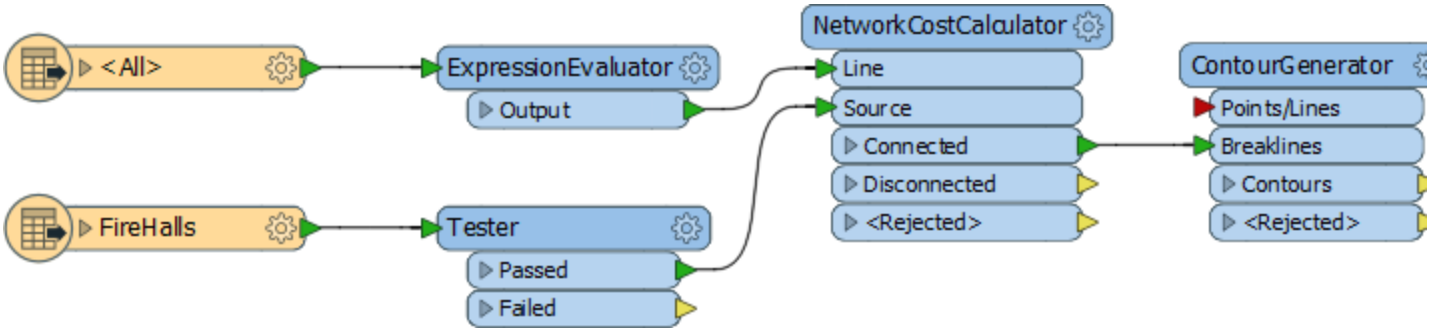
Connect an Inspector transformer to the Contours output port and run the workspace. You will get a set of Isodistance lines showing the distance (by road) from the chosen firehall:



Map tiles by Stamen Design, under CC-BY-3.0. Data by OpenStreetMap, under CC-BY-SA.

8. To create Isochrones (lines of equal travel time) we need to specify the speed at which a vehicle can travel. Obviously this varies depending on the type of road, terrain, weather, and intersection crossings; but a quick search shows that the average speed of a fire truck in a city is 55kph (35mph).

So, add an ExpressionEvaluator transformer to the workspace, between the AutoCAD Roads feature type and the NetworkCostCalculator transformer:

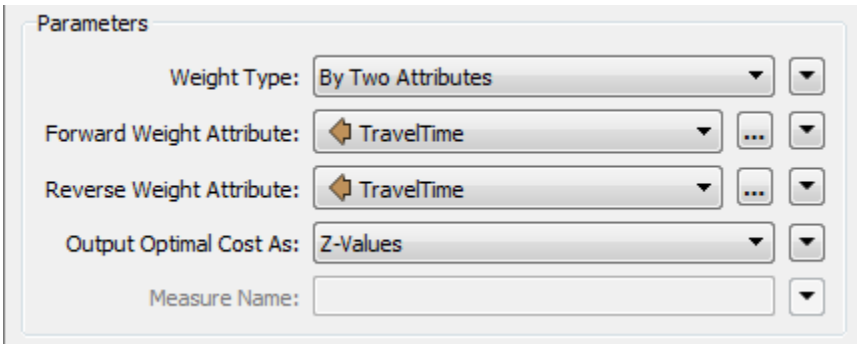


9. Open the parameters dialog for the ExpressionEvaluator. Set up an expression to calculate an attribute called TravelTime. The expression should be:

```
@Length()/916.666666
```

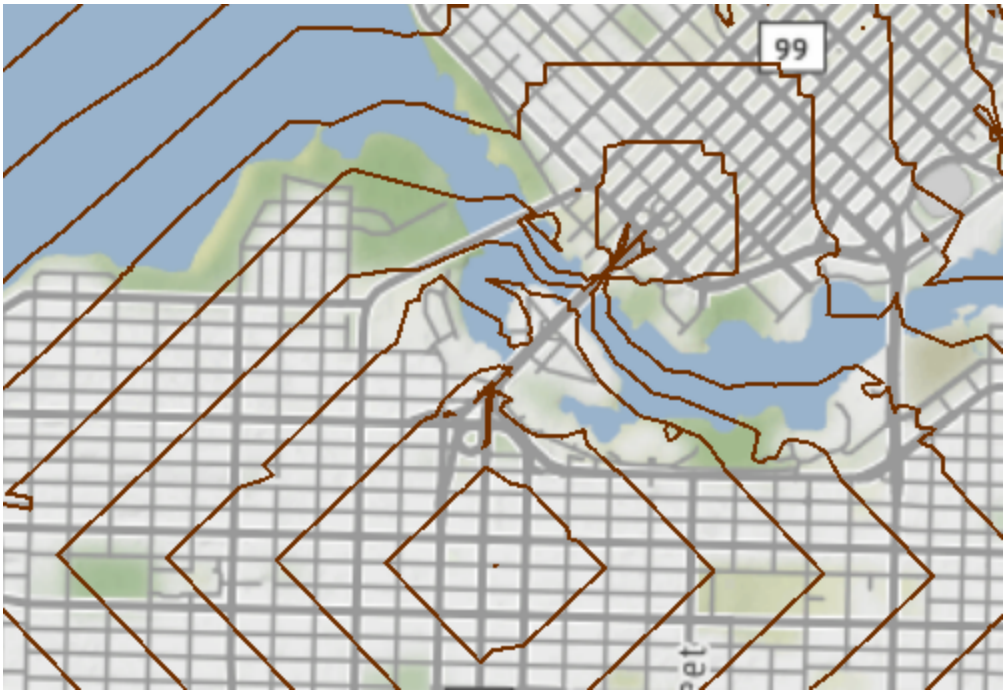
@Length() is an FME function that measures the length of each road link. 916.66666 comes from (speed * (metres-per-km / seconds-per-minute)) which is (55*(1000/60))

10. Now open the parameters dialog for the NetworkCostCalculator. Change the Weight Type parameter from "By Length" to "By Two Attributes". Select TravelTime as the attribute for both the forward and reverse weight:



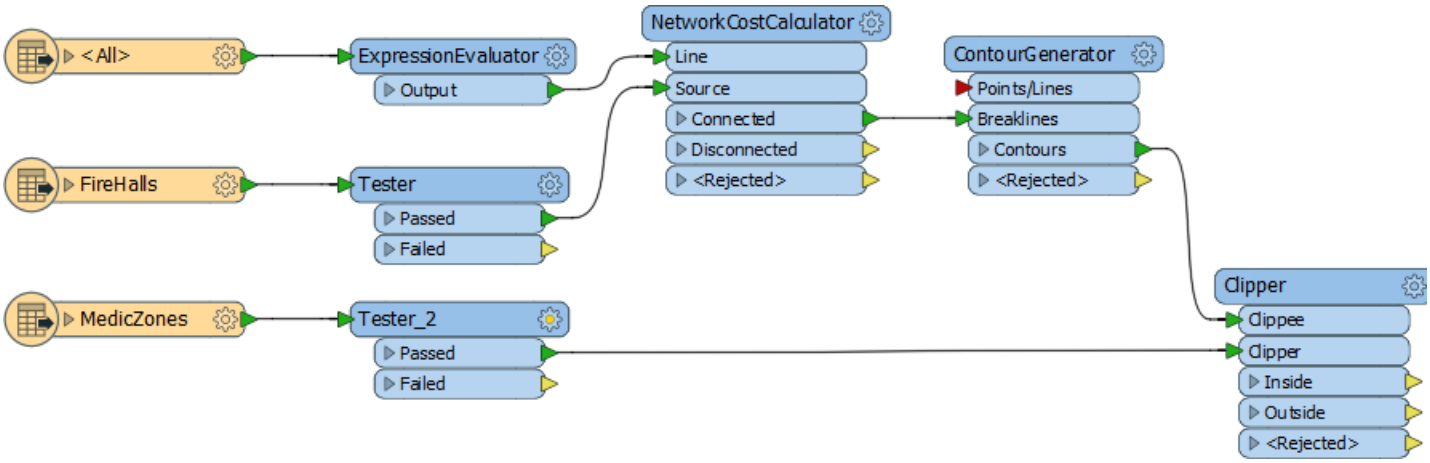
11. Open the parameters dialog for the ContourGenerator. Change the Surface Tolerance to 1 and the Output Contour Interval to 0.5 (i.e. each contour will represent 0.5 minutes or 30 seconds of travel time).

Run the workspace. The result will be a series of isochrones representing travel times for emergency vehicles in the city of Vancouver:

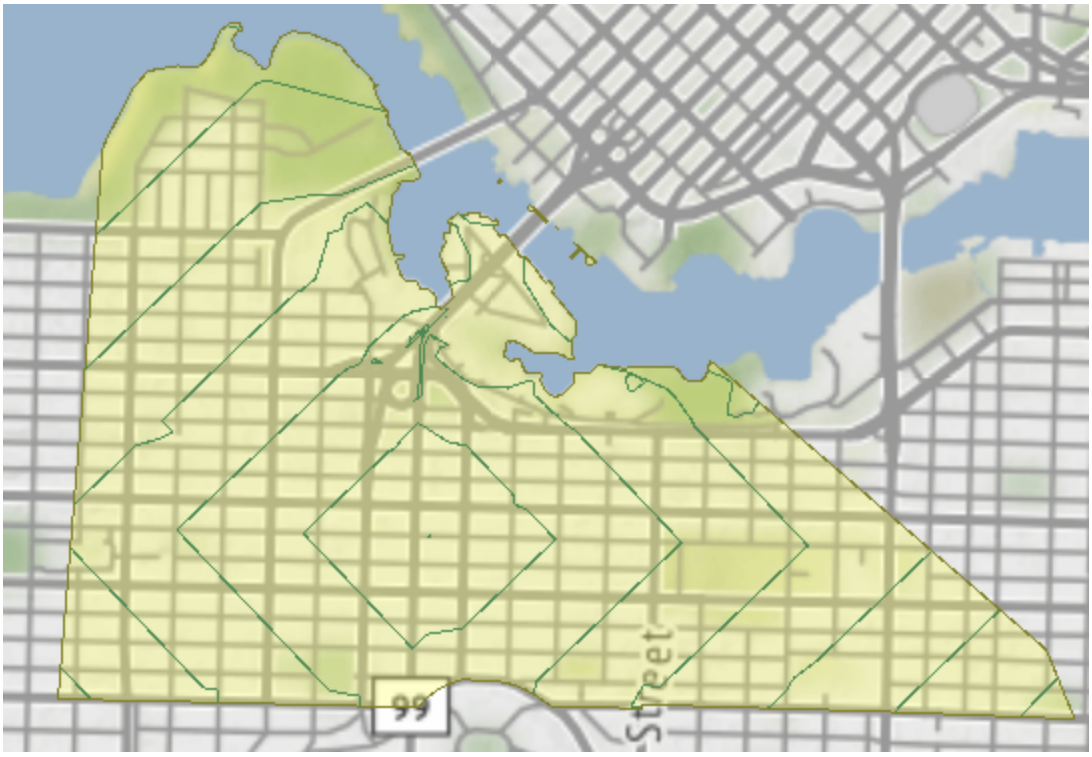


12. To determine whether all parts of this firehall's area of responsibility are covered by the four-minute limit, we'll need to clip the results to that boundary. So add a Clipper transformer to the workspace. Connect the contours to the Clippee input port and the MedicZone feature type to the Clipper port:

13. We only need the MedicZone that relates to the chosen firehall and should filter the rest out with a Tester. The simplest solution is to duplicate the existing Tester (select it and press Ctrl+D) and drop the duplicate Tester into the MedicZones-Clipper connection:



Now run the workspace again. With this result we can see that there are no parts of this hall's area of responsibility that fall outside of the four-minute travel limit:



Notes

Further updates to this project, should you wish to try them, would be to:

- Set a different average travel speed for each road type; for example arterial roads could be 60kph, whereas residential roads are 40kph. See attached template for an example of this. Notice how it changes the results so that some areas do fall outside the four-minute limit.
- Do as in this example, where each block of properties is color-coded depending on the amount of time required to travel to it.
- Write the processed road network (the output from the NetworkCostCalculator) to a dataset using a fanout so that each layer of data represents the road extents that could be travelled with an additional minute of travel time.

Downloads

Source Datasets (2016)

Source Datasets (2018)

Completed Workspace as a Template (2016)

Created Workspace as a Template (2018)

Completed Workspace (Note 1) (2016)

Geocoding Addresses

Overview

In this tutorial, you will learn how to geocode addresses using the Forward and Reverse Geocoding Modes of the Geocoder transformer. In Exercise 1, you will use the Forward Mode by reading in a spreadsheet that contains a list of schools with the address attributes in a single column. Similarly, in Exercise 2, you will use the Reverse Mode by reading in a spreadsheet that contains a list of schools with the corresponding latitude and longitude coordinates in order to derive the address information.

Downloads

geocoding-addresses.fmwt

schooladdress.xlsx

schoolcoords.json

Exercise 1 - *Forward Geocoding*

In this scenario, you will be geocoding using the address attribute from an Excel file for deriving X, Y coordinates in the LL84 coordinate system. This tutorial will utilize the OpenStreetMaps geocoding service in order to obtain the lat/long coordinates for schools in Vancouver.

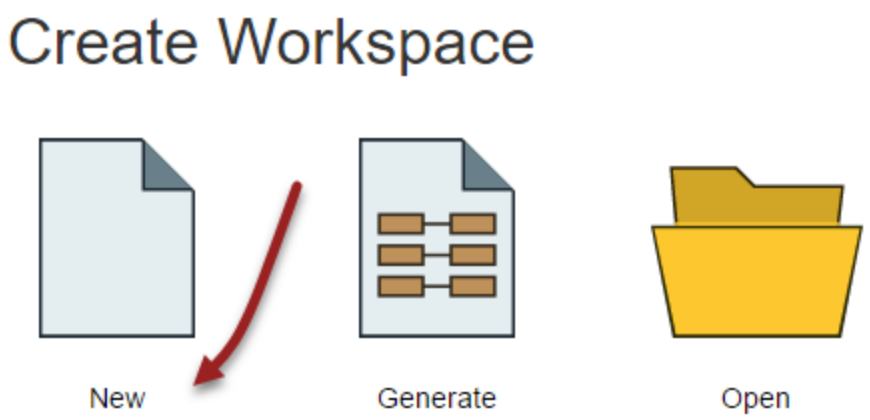
Instructions

1. Start FME Workbench

- If it isn’t open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a CSV Reader

- Start typing “CSV” without anything selected on the canvas, then select the CSV format from the reader list by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once the CSV reader has been added, the reader dialog will open, click on the Dataset ellipsis and open the SchoolAddress excel file from the sample data set then click OK (for example C:\Users\Documents\FME\Schools\SchoolAddress.xlsx).

The Excel Workbook contains 3 attribute columns: School Name, School Category, and Address. You will use the Address column to geocode the schools and obtain their coordinates in the LL84 coordinate system.

4. Add a Geocoder

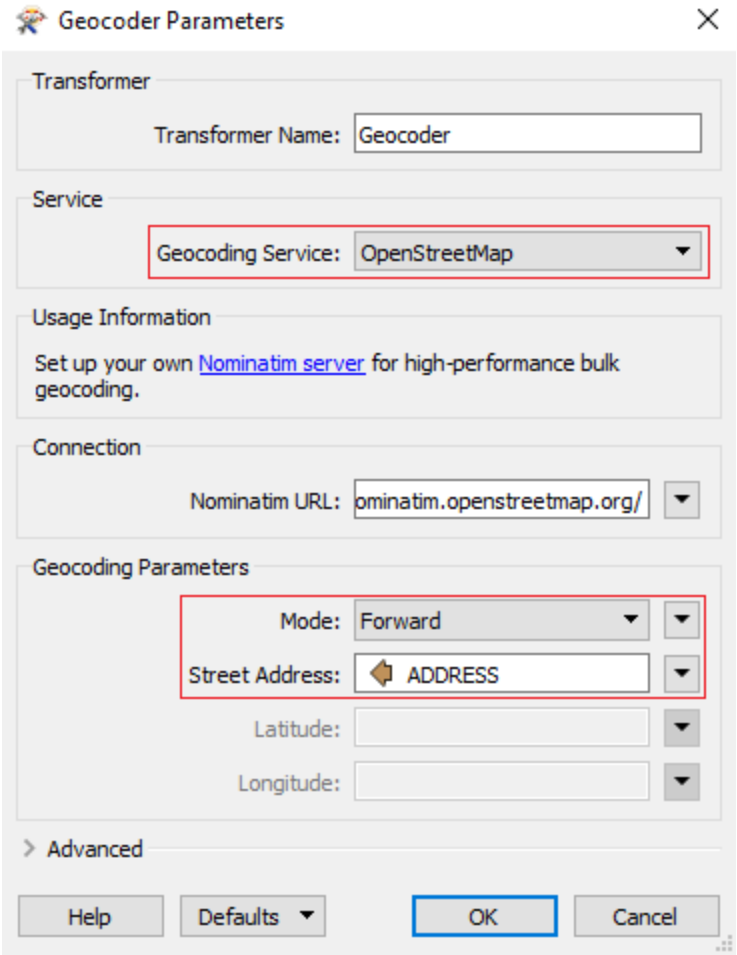
- Similar to adding a reader, type “Geocoder” to bring up the list of FME Transformers. Select the Geocoder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Connect the Geocoder to the SchoolAddress reader.

5. Specify the Geocoding Service

- Once the Geocoder has been added, double-click the Geocoder or click on the gear icon to open the transformer parameters.
- In the Service Section, set the Geocoding Service to OpenStreetMap by selecting it from the drop-down list.

For more information on the geocoding services available in the Geocoder transformer, see the Geocoder Documentation. Additionally, many services have restrictions on how their results may be used or displayed and may produce different address formats. Check with your chosen service for terms and conditions.

- In the Geocoding Parameters section, ensure the Mode is set to Forward and set the Geocoding Parameters Street Address to ADDRESS by selecting it from the drop-down list.
- Once the parameters are set, click OK to accept the changes and close the Geocoder Parameters dialog.



6. Add an AttributeManager

In this exercise, the AttributeManager will be used to both remove unwanted attributes that are created by the Geocoder and rename the latitude and longitude attributes in order to match the naming convention of the input file.

- Connect the AttributeManager to the Geocoder:Output port.

7. Open the AttributeManager Parameters

The Geocoder transformer will create a number of attributes; however, you are only interested in obtaining the address information in this case. In the Attribute Actions section, remove the unwanted attributes either by selelcting “Remove” from the drop-down list in the Actions column OR selecting the row and using the remove row button (-) at the bottom of the Attribute Actions section. For more information on removing attributes, see the Desktop Basic Course Manual.

- All attributes should be removed other than the 5 attributes listed below:

Input Attribute	Output Attribute	Attribute Value	Action
SCHOOL_NAME	SCHOOL_NAME		Do Nothing
SCHOOL_CATEGORY	SCHOOL_CATEGORY		Do Nothing
_latitude	**LATITUDE**		**Rename**
_longitude	**LONGITUDE**		**Rename**
ADDRESS	ADDRESS		Do Nothing

- To rename an attribute, simply click the Output Attribute cell of the row of the Input Attribute you wish to change. Once the name of the attribute is changed, the Action will automatically be set to “Rename”.

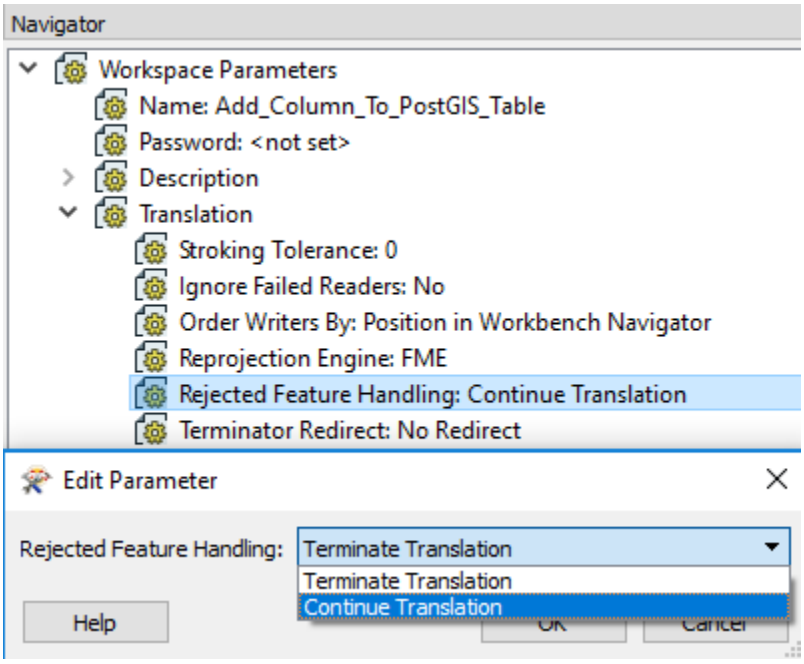
8. Add an Inspector

- Next, add an Inspector after the AttributeManager. This will automatically open the point feature dataset in the FME Data Inspector after the translation has run.

9. Set the Workspace Parameters

- Before you can run the workspace, you have to set the Workspace Parameters (found in the Navigator window) > Translation > Rejected Feature Handling to Continue Translation.

By default, the FME Workspace Parameter is set to Terminate Translation which will cause the Workspace to stop if any features are rejected by the Geocoder. You can learn more about rejected feature handling in the Desktop Basic Course Manual.



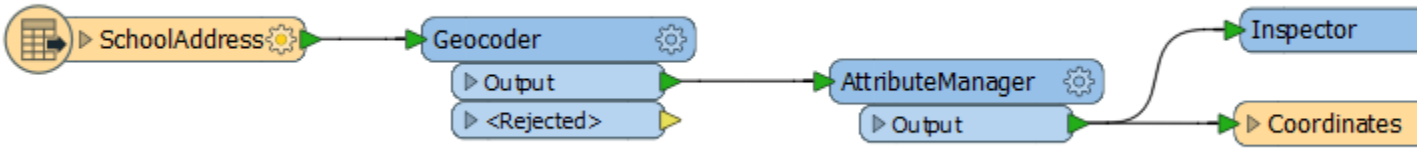
In this case, we set the parameter to continue because some addresses will be rejected if the address information were not entered incorrectly (i.e. 1234 West 49th vs. West 49, 1234). Again, the expected input address format varies between services. See the documentation provided by your chosen service for details. After running the workspace you will notice that 35 addresses were rejected and 159 were correctly geocoded, without changing this parameter, the translation would terminate after the first feature is rejected.

10. Add a Google KML Writer

- Similar to adding a reader, type “KML” then select the Google KML format from the list of Writers.
- Specify a file name and set the destination folder for the KML file by clicking on the Dataset ellipsis. After specifying a folder location and filename, set the Table Definition to Automatic and click OK.
- Once the writer is added to the canvas, connect it to the AttributeManager:Output port.

11. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



If you wish to save the output, you can either connect a writer to the AttributeManager:Output port or by using the Save Selected Data as in the FME Data Inspector and selecting the format that you wish to save the data as (i.e. PostGIS, File Geodatabase, DWG, etc.). For more instructions on saving data in the FME Data Inspector, see the Saving Data Documentation.

Results

Input

Table View

SchoolAddress [XLSXR] - SchoolAddress

	SCHOOL_NAME	SCHOOL_CATEGORY	ADDRESS
1	Admiral Seym...	Public School	Admiral Seymour Elementary School, Keefer Street, Strathcona, Vancouver, Metro Vancouver, British Columbia, V6A, Canada
2	Admiral Seym...	StrongStart BC	Admiral Seymour Elementary School, Keefer Street, Strathcona, Vancouver, Metro Vancouver, British Columbia, V6A, Canada
3	Alexander Aca...	Independent School	688, West Hastings Street, Davie Village, Downtown, Vancouver, Metro Vancouver, British Columbia, V7X, Canada
4	Anchor Point ...	Independent School	Anchor Point 3, 1333, Hornby Street, Davie Village, West End, Vancouver, Metro Vancouver, British Columbia, V6Z, Canada
5	BC Children's	Public School	Eric Hamber Secondary School, Willow Street, Cambie Village, South Cambie, Vancouver, Metro Vancouver, British Columbia

in any column

Output

Table View

inspector [FFS] - Inspector_2

	SCHOOL_NAME	LATITUDE	LONGITUDE	SCHOOL_CATEGORY	ADDRESS
1	Admiral Seym...	49.27859427	-123.0803143	Public School	Admiral Seymour Elementary School, Keefer Street, Strath
2	Admiral Seym...	49.27859427	-123.0803143	StrongStart BC	Admiral Seymour Elementary School, Keefer Street, Strath
3	Alexander Aca...	49.285000...	-123.114009...	Independent School	688, West Hastings Street, Davie Village, Downtown, Vanc
4	Anchor Point ...	49.277061...	-123.130918...	Independent School	Anchor Point 3, 1333, Hornby Street, Davie Village, West E
5	BC Children's	49.239630	-123.12579006	Public School	Eric Hamber Secondary School, West 33rd Avenue, Camb

in any column

Exercise 2 - *Reverse Geocoding*

In this scenario, you will be geocoding using the latitude and longitude attributes from an Excel file in order to derive postal address information. This tutorial will utilize the OpenStreetMaps geocoding service in order to obtain school addresses in the Vancouver area.

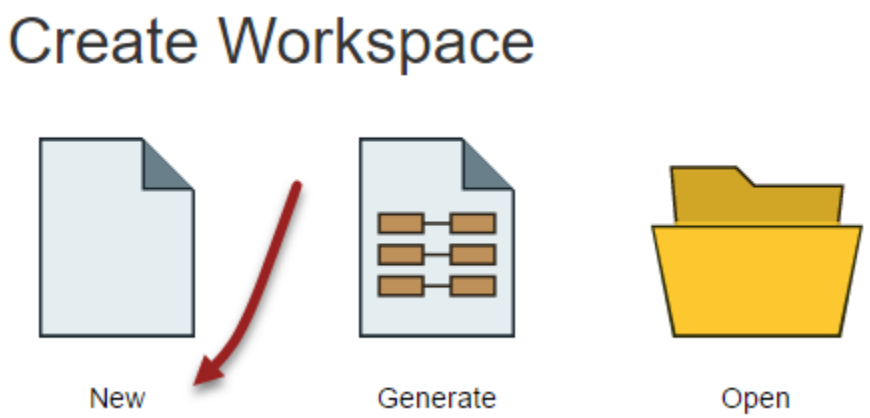
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a JSON Reader

- Start typing "JSON" without anything selected on the canvas, then select the JSON format from the reader list by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Once added, the reader box will open, click on the Dataset ellipsis and open the SchoolCoords excel file from the sample data set then click OK (for example C:\Users\Documents\FME\Schools\SchoolCoords.xlsx).

The JSON file contains 4 attributes: School Name, School Category, Latitude, and Longitude. You will be using the Latitude and Longitude columns to geocode the schools and obtain each X, Y coordinates address.

4. Add a Geocoder

- Similar to adding a reader, type "Geocoder" to bring up the list of FME Transformers. Select the Geocoder from the list of Transformers by double-clicking or by using the arrow keys and the Enter key to add it to the canvas.
- Connect the Geocoder to the SchoolCoords reader.

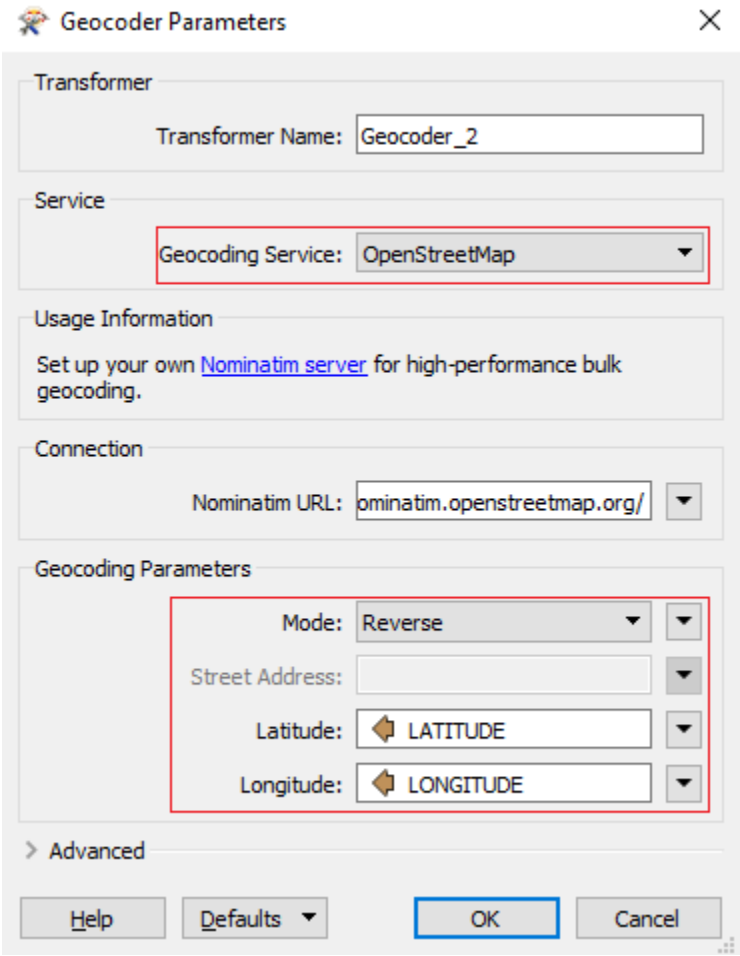
5. Specify the Geocoding Service

- Once the Geocoder has been added, double-click the Geocoder or click on the gear icon to open the transformer parameters.
- In the Service Section, set the Geocoding Service to OpenStreetMap by selecting it from the drop-down list.

For more information on the geocoding services available in the Geocoder transformer, see the Geocoder Documentation. Additionally, many services have restrictions on how their results may be used or displayed and may produce different address formats. Check with your chosen service for terms and conditions.

- In the Geocoding Parameters section, ensure the Mode is set to Reverse.

- Next, set the Latitude parameter to the LATITUDE attribute by selecting it from the drop-down list. Similarly, set the longitude parameter to the LONGITUDE attribute.
- Once the Latitude and Longitude parameters are set, click OK to accept the changes and close the Geocoder Parameters dialog.



6. Add an AttributeManager

In this exercise, the AttributeManager will be used to both remove unwanted attributes that are created by the Geocoder and rename the latitude and longitude attributes in order to match the naming convention of the input file.

- Connect the AttributeManager to the Geocoder:Output port.

7. Open the AttributeManager Parameters

The Geocoder transformer will create a number of attributes; however, you are only interested in obtaining the address information in this case. In the Attribute Actions section, remove the unwanted attributes by either selecting “Remove” from the drop-down list in the Actions column OR selecting the row and using the remove row button (-) at the bottom of the Attribute Actions section. For more information on removing attributes, see the Desktop Basic Course Manual.

- All attributes should be removed other than the 5 attributes listed in the table below:

Input Attribute	Output Attribute	Attribute Value Action
SCHOOL_NAME	SCHOOL_NAME	Do Nothing

SCHOOL_CATEGORY	SCHOOL_CATEGORY	Do Nothing
LATITUDE	LATITUDE	Do Nothing
LONGITUDE	LONGITUDE	Do Nothing
_address	**ADDRESS**	**Rename**

- To rename an attribute, simply click the Output Attribute cell of the row of the Input Attribute you wish to change. Once the name of the attribute is changed, the Action will automatically be set to “Rename”. Once the parameters are set, click OK to accept the changes and close the AttributeManager Parameters dialog.

8. Add an Inspector

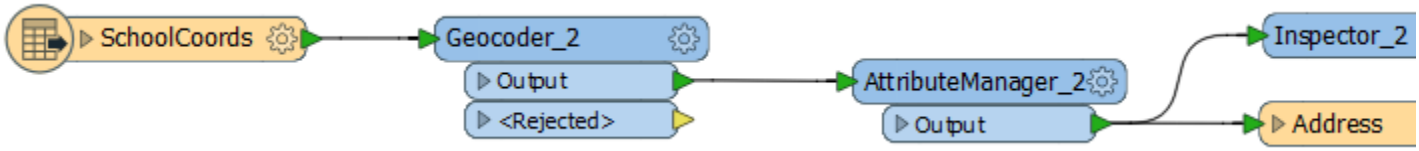
- Next, add an Inspector after the AttributeManager. This will automatically open the point feature dataset in the FME Data Inspector after the translation has run.

9. Add a Google KML Writer

- Similar to adding a reader, type “KML” then select the Google KML format from the list of Writers.
- Specify a file name and set the destination folder for the KML file by clicking on the Dataset ellipsis. After specifying a folder location and filename, set the Table Definition to Automatic and click OK.
- Once the writer is added to the canvas, connect it to the AttributeManager:Output port.

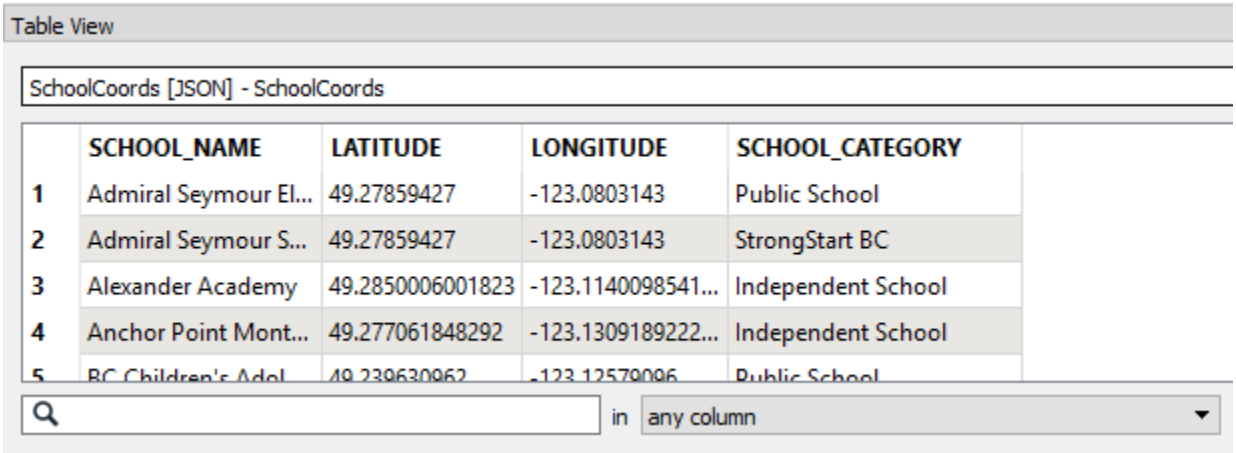
10. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



Results

Input



	SCHOOL_NAME	LATITUDE	LONGITUDE	SCHOOL_CATEGORY
1	Admiral Seymour El...	49.27859427	-123.0803143	Public School
2	Admiral Seymour S...	49.27859427	-123.0803143	StrongStart BC
3	Alexander Academy	49.2850006001823	-123.1140098541...	Independent School
4	Anchor Point Mont...	49.277061848292	-123.1309189222...	Independent School
5	BC Children's Adol...	49.239630962	-123.12579096	Public School

Output

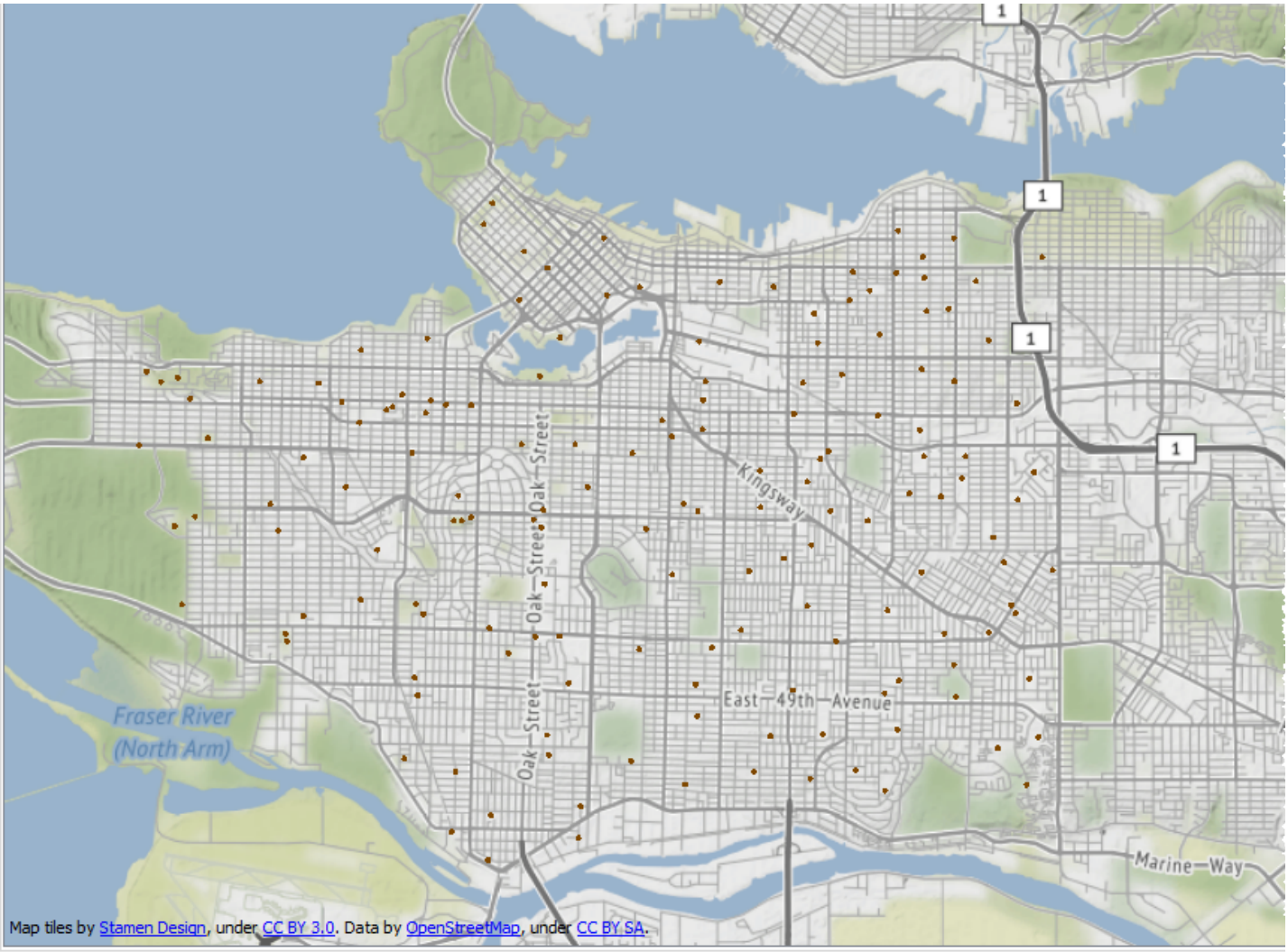


Table View

inspector [FFS] - Inspector_2

	SCHOOL_NAME	LATITUDE	LONGITUDE	SCHOOL_CATEGORY	ADDRESS
1	Admiral Seymour El...	49.27859427	-123.0803143	Public School	Admiral Seymour Elementary School, Keefer Street, Strathcona, ...
2	Admiral Seymour S...	49.27859427	-123.0803143	StrongStart BC	Admiral Seymour Elementary School, Keefer Street, Strathcona, ...
3	Alexander Academy	49.2850006001823	-123.1140098541...	Independent School	688, West Hastings Street, Davie Village, Downtown, Vancouver, ...
4	Anchor Point Mont...	49.277061848292	-123.1309189222...	Independent School	Anchor Point 3, 1333, Hornby Street, Davie Village, West End, Va...
5	BC Children's Adol...	49.239630962	-123.12579096	Public School	Eric Hamber Secondary School, West 33rd Avenue, Cambie Villa...

in any column

Transformers

- AttributeManager - Alters multiple attributes through adding, renaming, copying, deleting and re-ordering.
- Geocoder - Uses various external web services to convert addresses to latitude/longitude coordinates, or to find the closest addresses to latitude/longitude coordinates (reverse geocode).
- Inspector - Sends features to the FME Data Inspector for display

Data Attribution

Map and Map Feature Annotation

Overview

Although FME is capable of creating cartographic elements, such as labels, it is not intended to be used for cartographic purposes. In this tutorial, you will learn how to create labels for point features and how to create a center point in polygons which can then be converted to a label point. For a detailed list of categories and transformers related to cartography see the Transformer Gallery and filter by Cartography and Reports. Additionally, if you want to learn more about creating labels for features, see the Labelling Features exercise in the Desktop Basic Training Manual.

Downloads

map-and-map-feature-annotation.fmwt

schools.zip

neighborhoodsgdb.zip

Exercise 1 - *Labelling Point Features*

In this scenario, you are interested in creating labels for school point features that will be exported to an AutoCAD DWG file for later use.

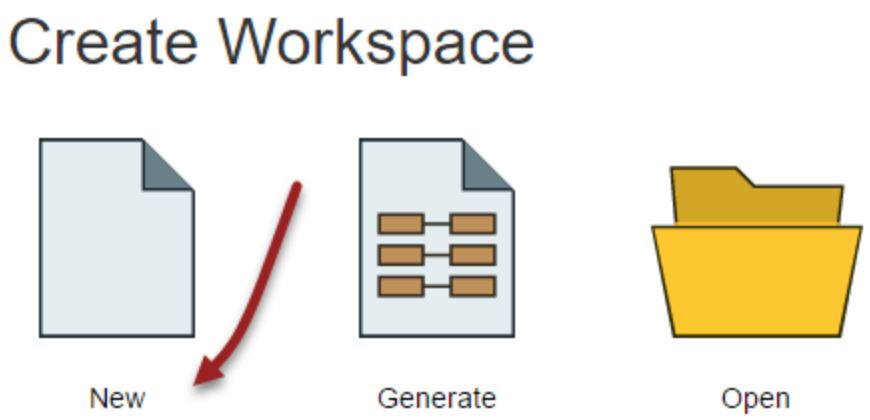
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add a Shapefile Reader

- Start typing “Shapefile” without anything selected on the workspace, then select the Shapefile format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the Schools shapefile (for example, C:\Users\Documents\FME\Schools\Schools.shp).

4. Add a LabelPointReplacer

As the name suggests, the LabelPointReplacer will replace point features with a label point.

- Select the Schools Reader Feature Type then type “LabelPointReplacer” to search via the Quick Add.
- Select the LabelPointReplacer from the list of Transformers that appears by double-clicking or using the arrow keys and the Enter key to add it to the workspace.

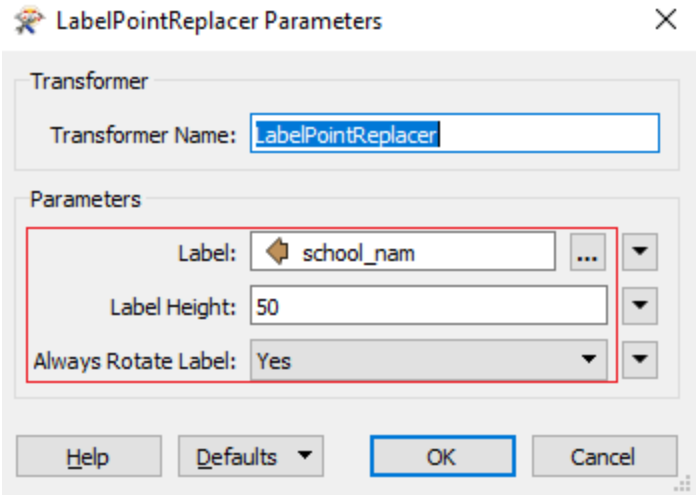
This will automatically make the connection from the Schools reader to the LablePointReplacer.

5. Set LabelPointReplacer’s Label Parameter and Label Height

- Once the transformer has been added, double-click the LabelPointReplacer or click on the gear icon to open the transformer parameters.

The first parameter you have to set is the Label - this will retrieve the value of an attribute and convert it into a label.

- Set the Label Parameter to school_nam to create a label for each school’s name.
- Next, set the Label Height to 50.



Note: Label height is measured in ground units (meters in this case) and may either be entered as a number or can be taken from the value of a feature attribute by selecting the attribute name from the drop-down list. If the label height is too tall, you can either adjust the label height accordingly in the LabelPointReplacer parameters or edit the annotation size in AutoCAD as you will be writing to a DWG file in this exercise.

6. Add an Inspector

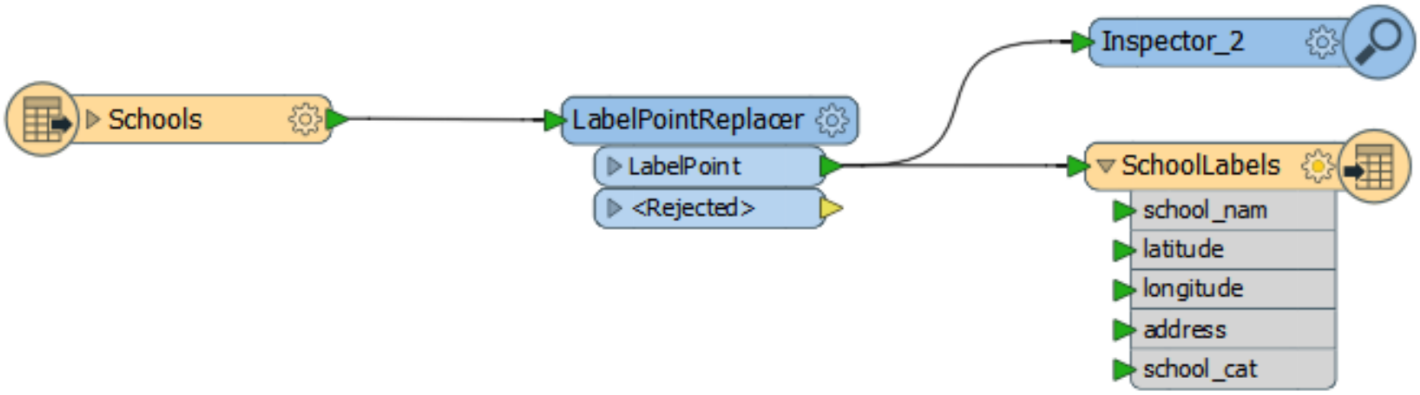
- Next, add an Inspector after the LabelPointReplacer. This will automatically open the output in the FME Data Inspector after the translation has successfully run.

7. Add an Autodesk AutoCAD DWG/DXF Writer

- Similar to adding a reader, type “DWG” then select the Autodesk AutoCAD DWG/DXF format from the list of Writers.
- Click on the Dataset ellipsis to specify where to save your data and name the file: SchoolLabels (for example, C:\Users\Documents\FME\Schools\SchoolLabels.dwg)
- Set the Layer Definition to Automatic to ensure all attributes that are created or renamed will be written to the dwg file.
- Similar to the Inspector, connect the SchoolLabels Writer to the LabelPointReplacer:LabelPoint port.

8. Run the Workspace

- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.

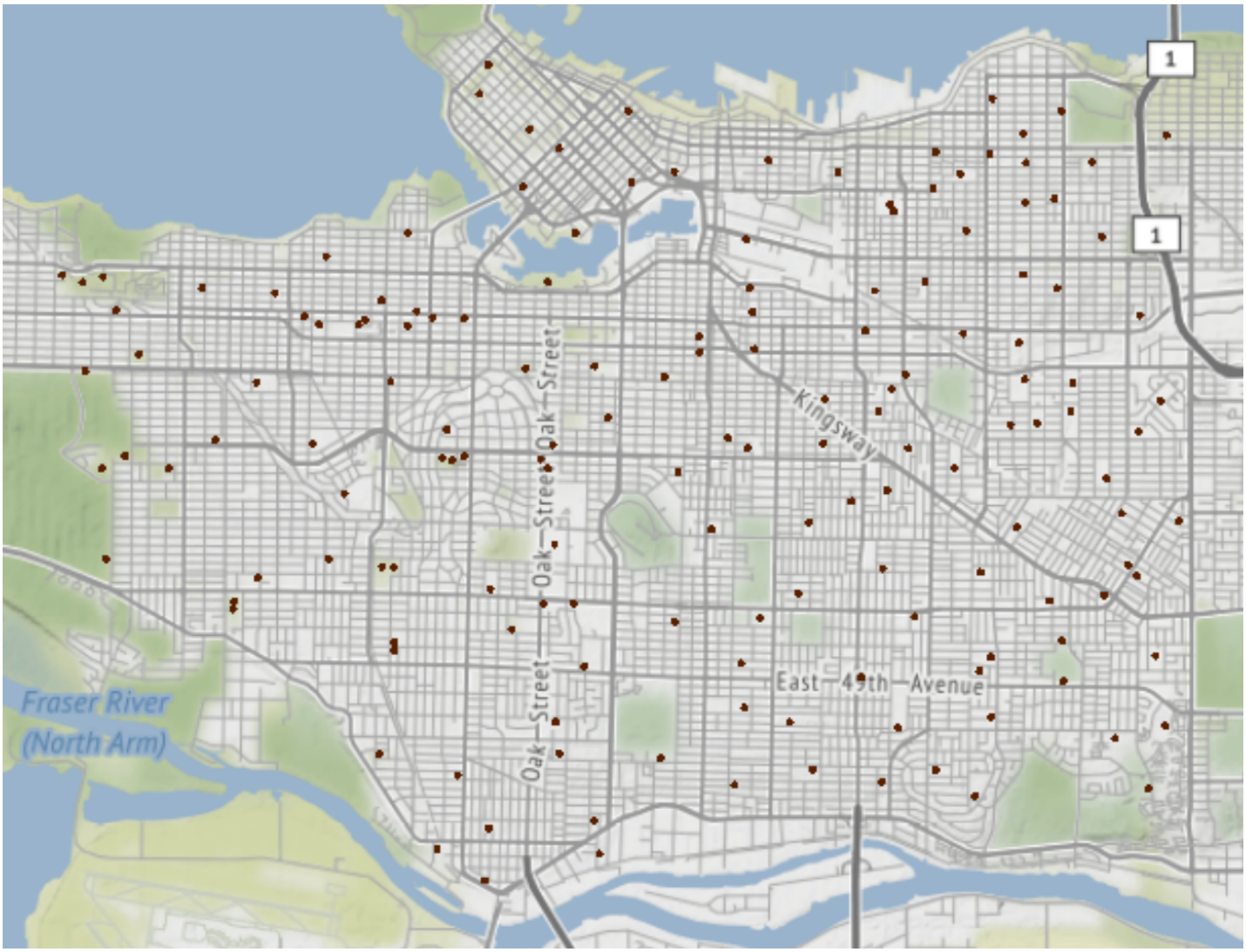


After running the workspace, you will have a new dataset with school labels in the DWG format. You can

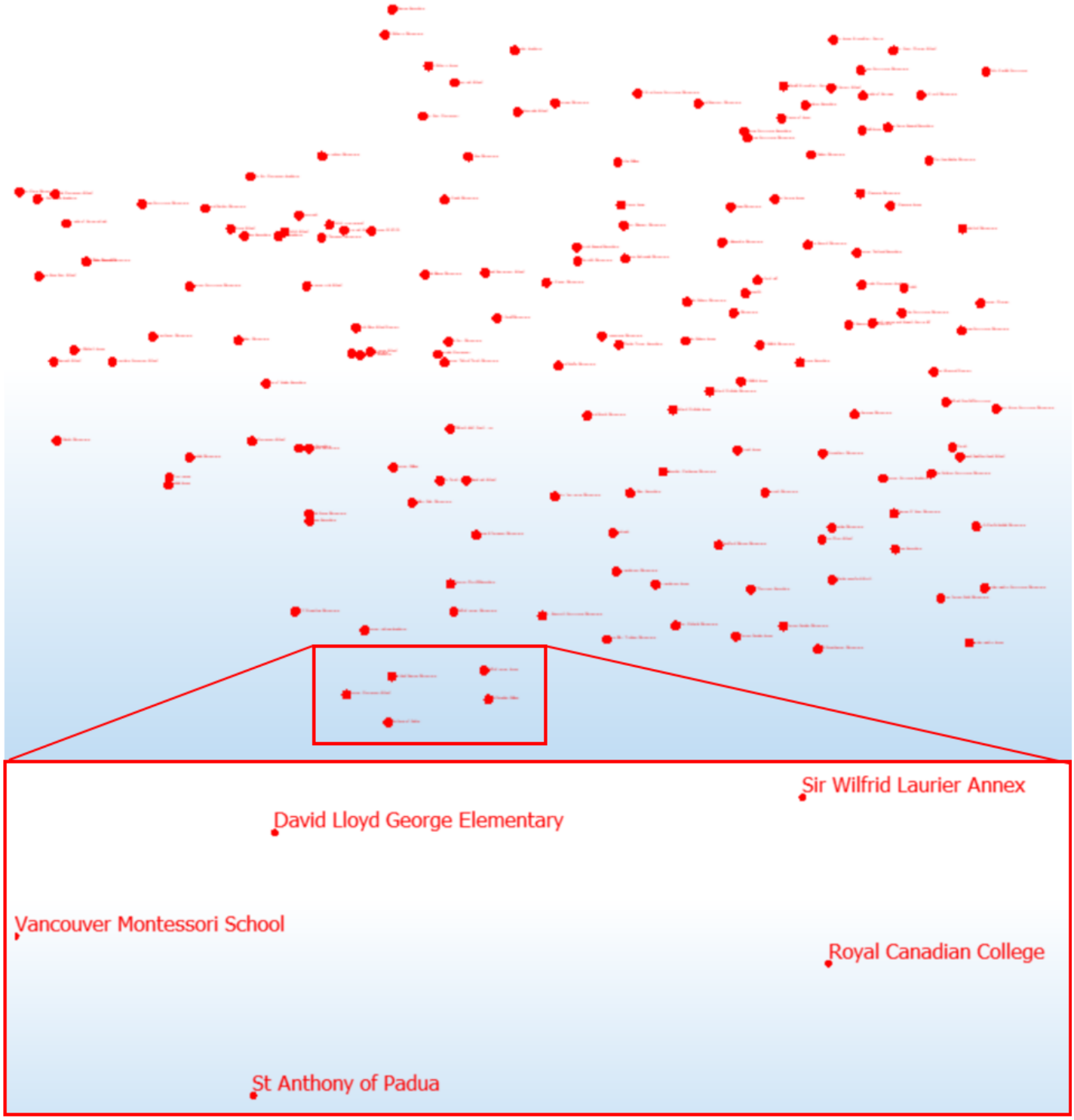
view the school labels by selecting the writer and clicking the Inspector button in the shortcut menu that appears when the writer is selected or by opening the SchoolLabels DWG file in AutoCAD.

Results

Input



Output



Exercise 2 - *Labelling Polygon Features*

In this scenario, you are interested in creating labels for neighborhood polygon features that will be exported to an AutoCAD file for later use.

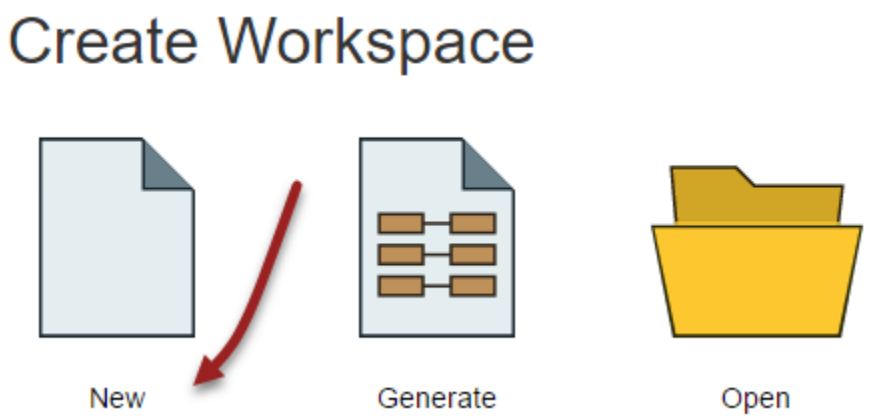
Instructions

1. Start FME Workbench

- If it isn't open already, launch FME Workbench.

2. Create a New Workspace

- In the Create Workspace part of the Start page, select the option to Create a New Workspace.



3. Add an Esri Geodatabase (File Geodb Open API) Reader

- Start typing "Geodatabase" without anything selected on the workspace, then select the Esri Geodatabase (File Geodb Open API) format from the list of Readers by double-clicking or by using the arrow keys and the Enter key to add it to the workspace.
- Click on the Dataset ellipsis, then navigate to the sample data folder and select the Neighborhoods geodatabase (for example, C:\Users\Documents\FME\Neighborhoods\Neighborhoods.gdb).

4. CenterPointExtractor

- Select the Neighborhoods Reader Feature Type then type "CenterPointExtractor".
- Select the CenterPointExtractor from the list of Transformers that appears by double-clicking or using the arrow keys and the Enter key to add it to the workspace.

This will automatically connect the Neighborhoods Reader to the CenterPointExtractor. The CenterPointExtractor will replace the geometry of the features with a point in the center of the features bounding box. For more information on Point Extraction Modes, see the CenterPointReplacer documentation.

5. Add a LabelPointReplacer

As the name suggests, the LabelPointReplacer will replace point features with a label point.

- Add a LabelPointReplacer after the CenterPointExtractor - the LabelPointReplacer should be connected to the CenterPointExtractor:Output port.

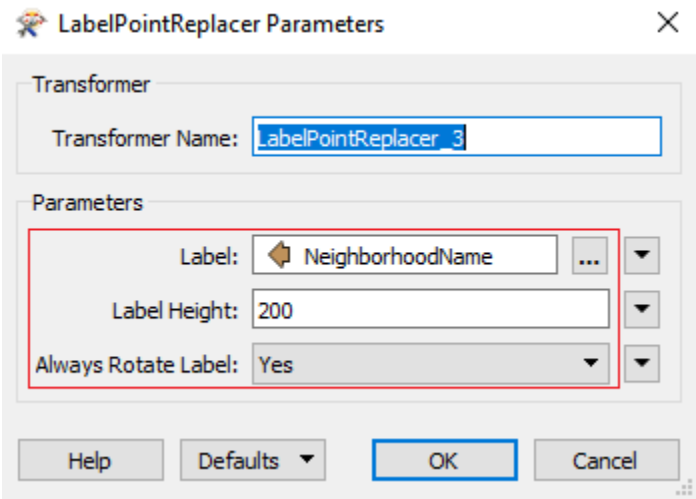
6. Set LabelPointReplacer's Label Parameter and Label Height

- Once the transformer has been added, double-click the LabelPointReplacer or click on the gear icon to open the transformer parameters.

The first parameter you have to set is the Label - in this case, it will retrieve the value of an attribute and

convert it into a label.

- Set the Label Parameter to NeighborhoodName to create a label for each Neighborhood's name.
- Next, set the Label Height to 200.



Note: Label height is measured in ground units (which is meters in this case) and may either be entered as a number or can be taken from the value of a feature attribute by selecting the attribute name from the pull-down list. If the label height is too tall, you can either adjust the size accordingly or edit the size of the text in AutoCAD later.

6. Add an Inspector

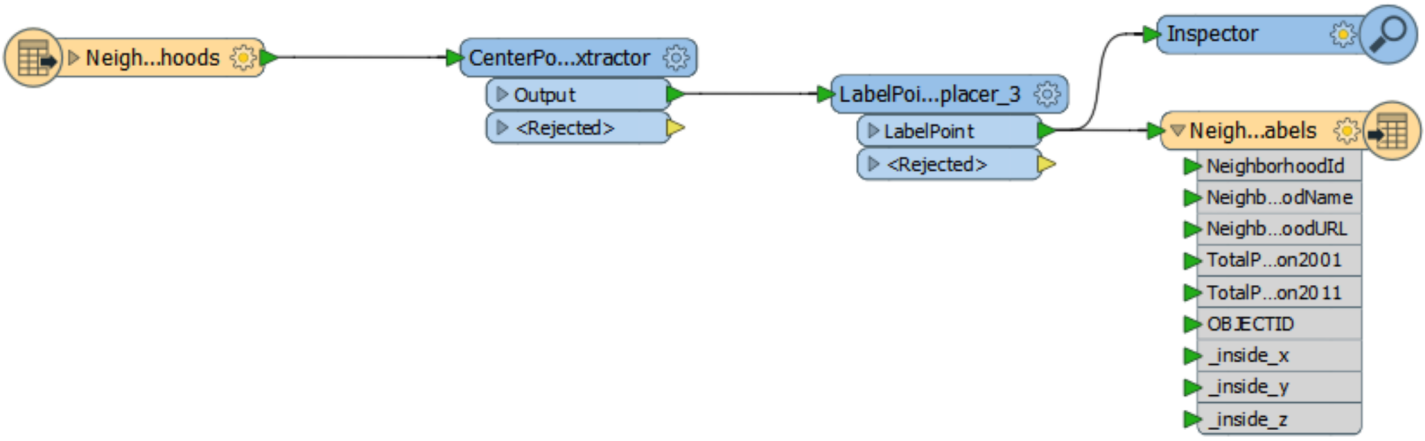
- Next, add an Inspector after the LabelPointReplacer. This will automatically open the output in the FME Data Inspector after the translation has successfully run.

7. Add an Autodesk AutoCAD DWG/DXF Writer

- Similar to adding a reader, type "DWG" then select the Autodesk AutoCAD DWG/DXF format from the list of Writers.
- Click on the Dataset ellipsis to specify where to save your data and name the file: NeighborhoodLabels (for example, C:\Users\Documents\FME\Neighborhoods\NeighborhoodLabels.dwg)
- Set the Layer Definition to Automatic to ensure all attributes that are created or renamed will be written to the DWG file.
- Similar to the Inspector, connect the NeighborhoodLabels Writer to the LabelPointReplacer:LabelPoint port.

8. Run the Workspace

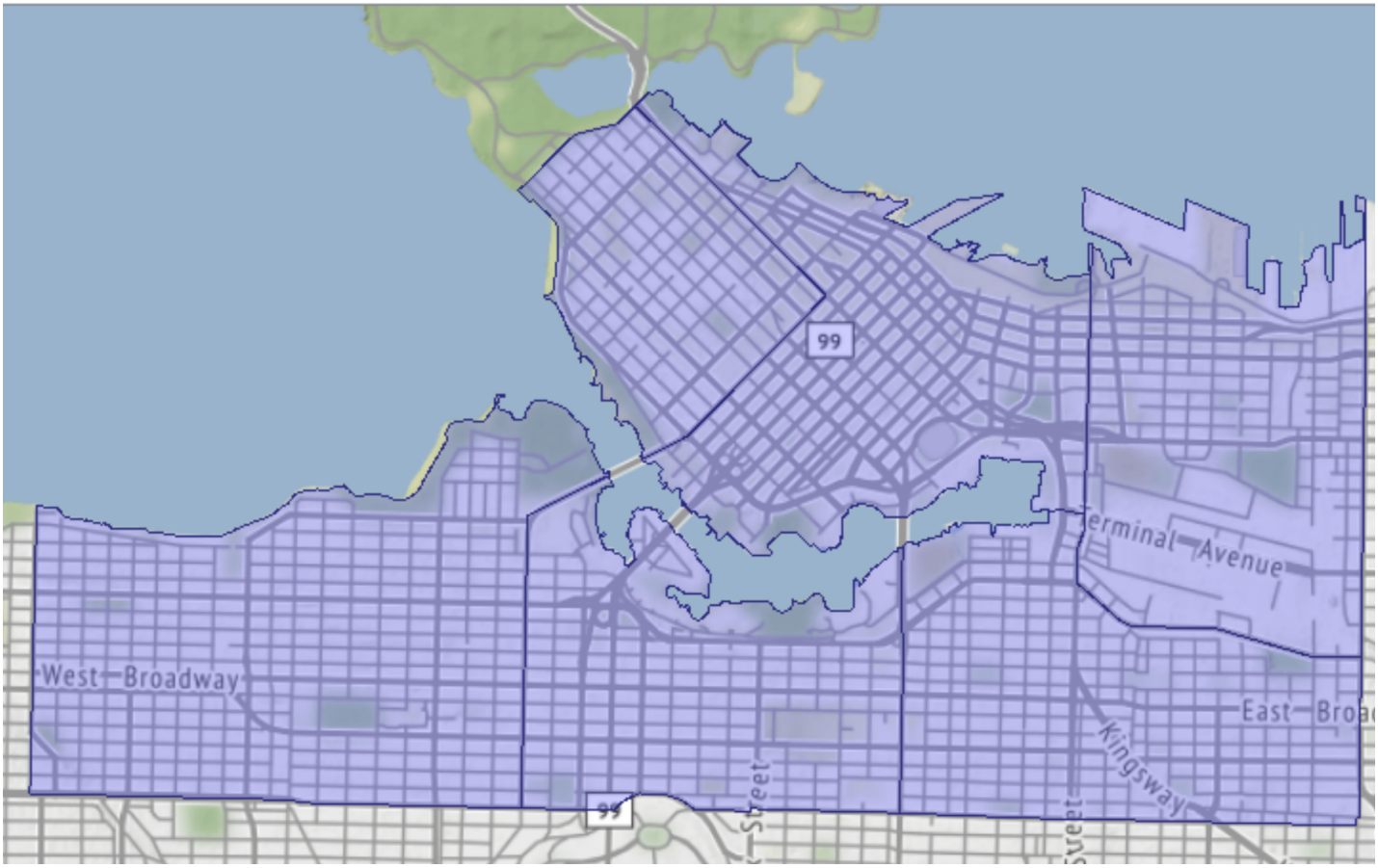
- Run the workspace by clicking the run button on the toolbar, or by using Run > Run Translation on the menu bar.



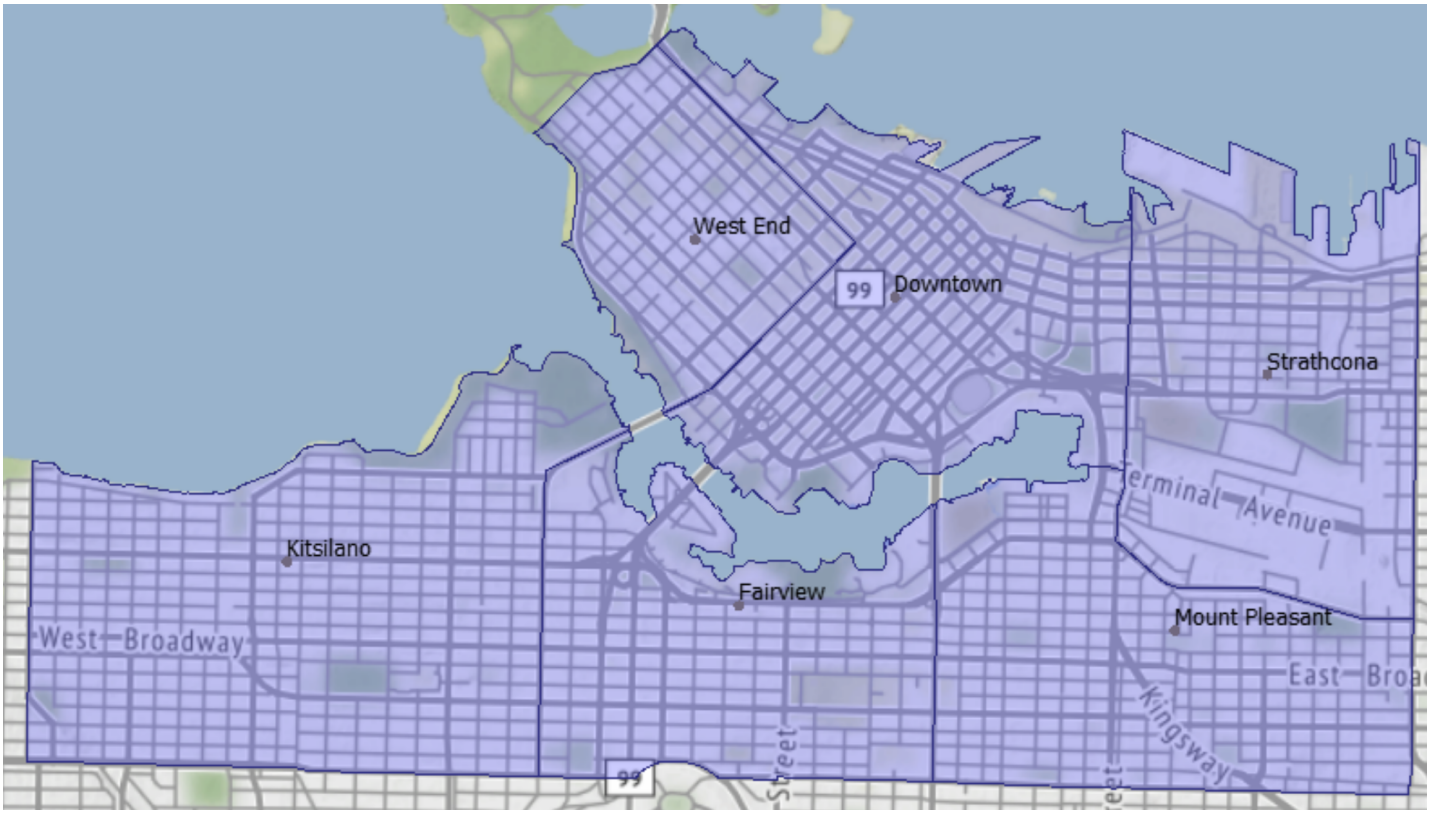
After running the workspace, you will have a new dataset with school labels in the DWG format. You can view the school labels by selecting the writer and clicking the Inspector button in the shortcut menu that appears when the writer is selected or by opening the NeighborhoodLabels DWG file in AutoCAD.

Results

Input



Output



Transformers

- CenterPointExtractor- Replaces the geometry of the feature with a point that is either in the center of the features bounding box, somewhere inside the features bounding box or replaces the geometry of the feature with a point at the center of mass of the feature.
- Inspector- Sends features to the FME Data Inspector for display.
- LabelPointReplacer- Replaces the geometry of the feature with a label point.

Data Attribution

Data used in this tutorial originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.