# FME Tutorial

FME is a GIS software, which includes features to transform and modify geographical data. With FME software it's possible to do format and coordinate system conversions to geographical data and modify attributes and geometry.

## Contents

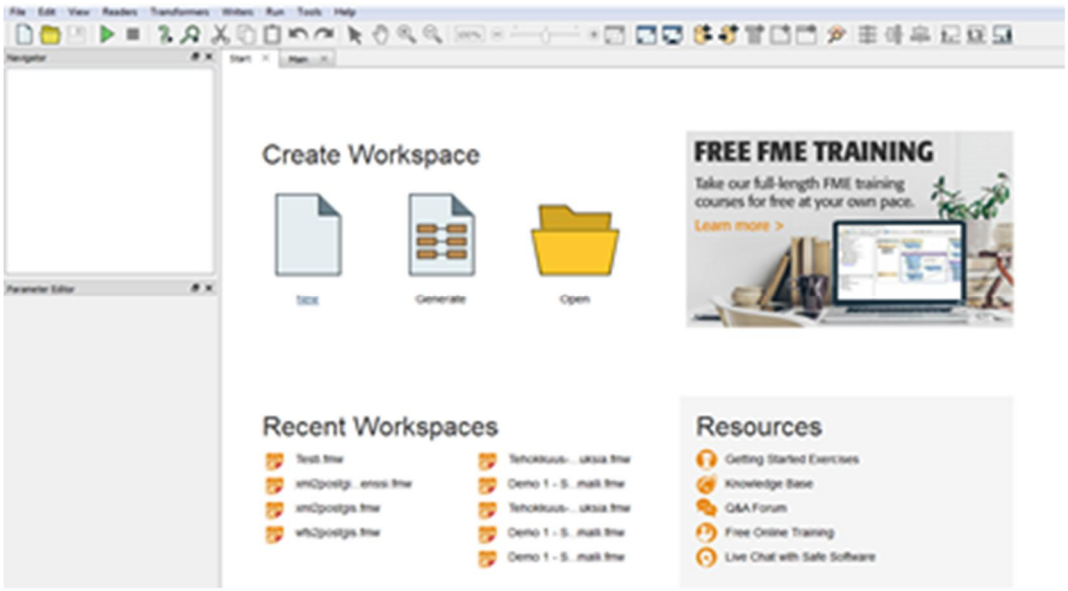
# 1 Installing FME

FME Desktop software can be installed from https://www.safe.com/support/support-resources/fme-downloads/. Choose the right version for your OS (for example FME 64-bit Windows Desktop -version for Windows operating system). To complete the installation the license information must be filled. The licence is purchased from the FME software vendor.

# 2 Getting started and usage

In this tutorial we will use the FME Workbench 2017.0 -software. The following view will show up when the software starts up.

Choose *New Workspace* option.

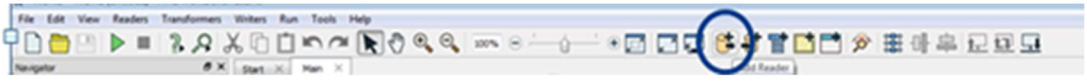



**Picture 1. Creating a new workspace.**

## 2.1 Using important features in FME

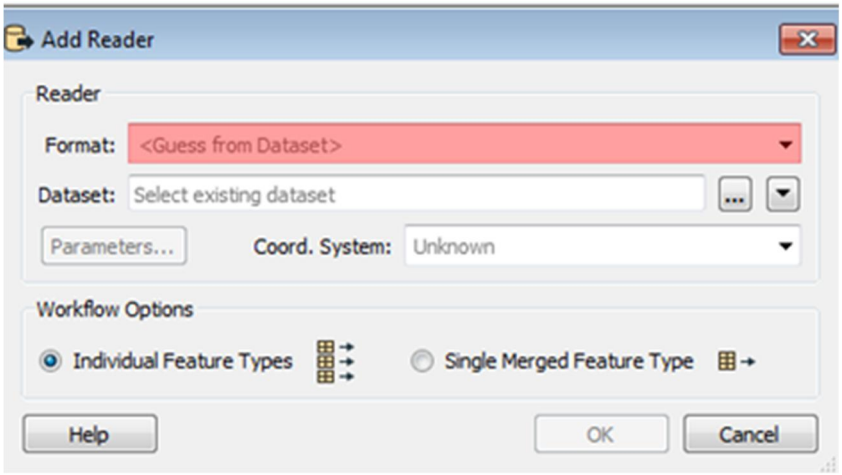
### 2.1.1 Reading from a data source (Add reader)

Data source is selected from the upper bar by pressing *Add Reader* (Picture 2).



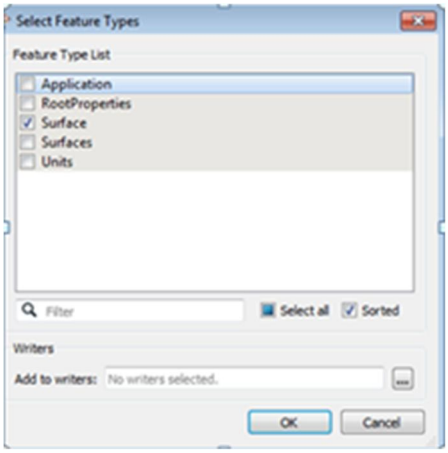
**Picture 2. Adding data source to the workspace.**

The file shape (format) and dataset are selected. The other parameters that can be added are coordinate system and the feature types added as individuals or single merged feature type (Picture 3). Format, dataset and coordinate system are selected from dropdown lists. The dataset can be selected either from your local computer or from an external data source.




**Picture 3. Data format, dataset and coordinate system of the data source.**

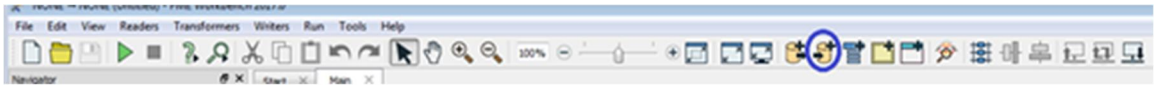
Choose features (Picture 4).




**Picture 4. Choosing the feature types.**

### 2.1.2 Writing to a data target (Add writer)

Data target is selected from the upper menu bar by pressing *Add writer* (Picture 5).




**Picture 5. Adding the destination data to the workspace**

Data format, dataset and coordinate system are selected for the target data (Picture 6). If the format of target data is PostGIS, the data connection must be created by

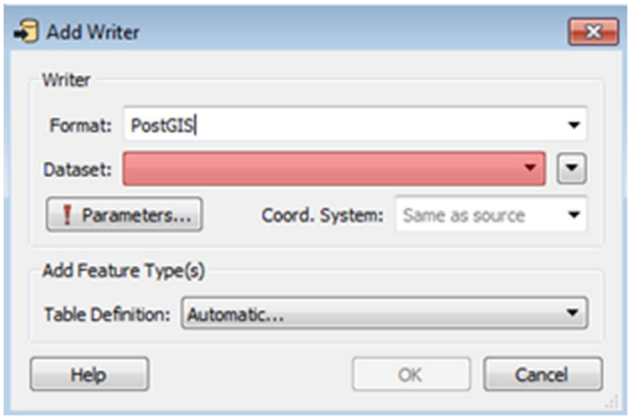
choosing *Add Database Connection* from the Dataset downshift option. Fill in database connection parameters (Picture 7).

Name: The name of your connection

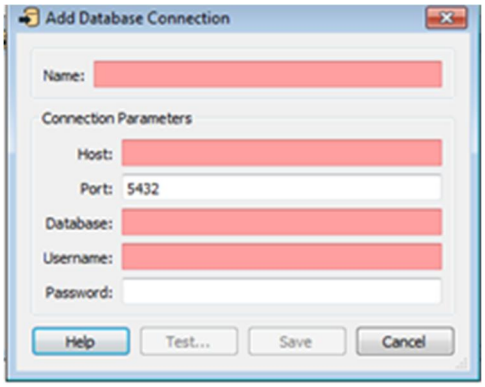
Host: IP -address or Hostname of your DB-server (if your DBMS is installed on your local machine this is “localhost”)

Port: The port of your DB (PostgreSQL default s 5432)

Database: The actual database name (PostgreSQL default is postgres)




**Picture 6. Format, dataset and coordinate system.**

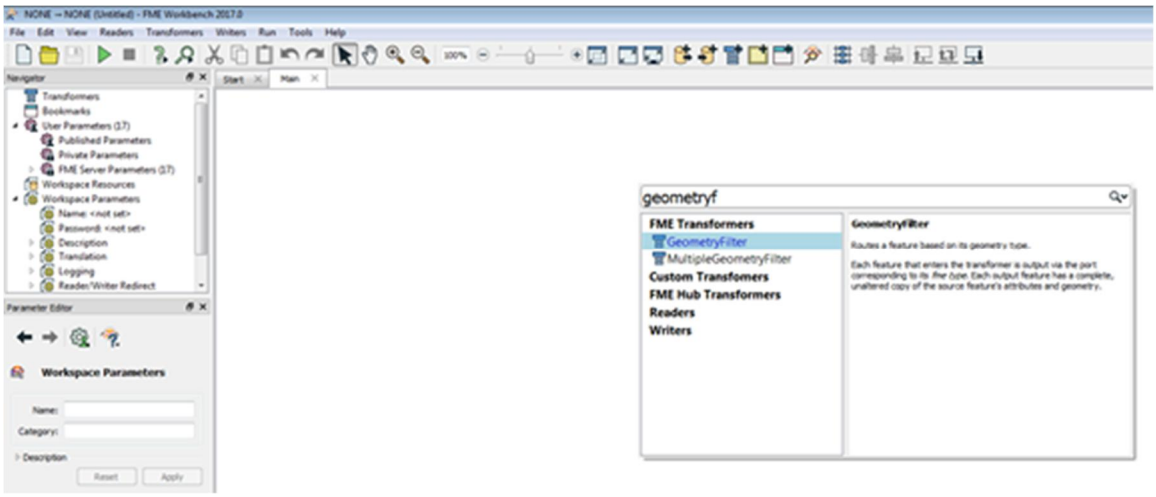



**Picture 7. Adding the database connection.**

### 2.1.3 Adding the transformers (Add transformer)

The transformer tools are used to make various mathematical data transformations. The transformer can be added either from upper bar by selecting *Add Transformer* (Picture 8) or by starting to type the name of the transformer on the workspace (Picture 9).


**Picture 8. Adding transformer to workspace.**



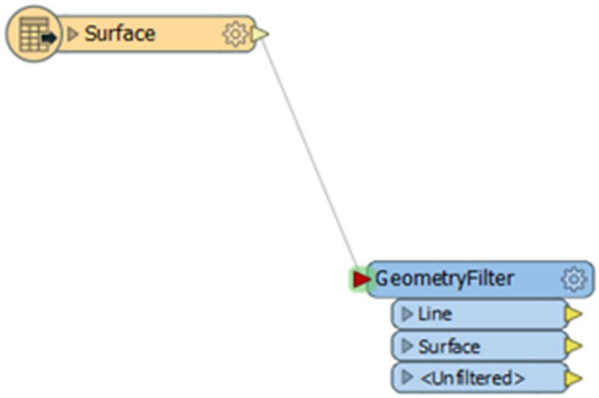

**Picture 9. Transformer can be added to workspace also by typing its name**.

When the transformer is added to workspace, its parameters can be adjusted by double-clicking the transformer on the workspace; then the parameters will appear on the screen.

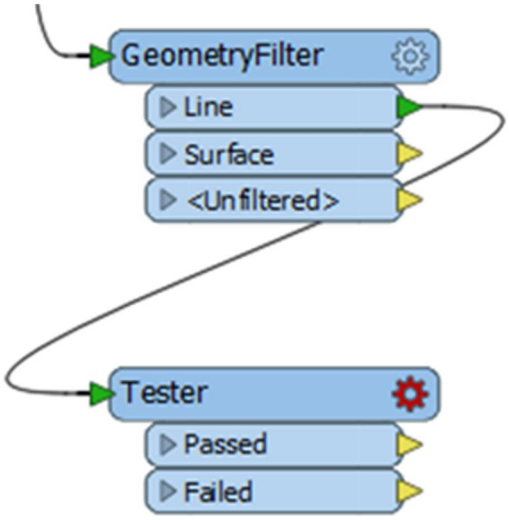
The parameters of readers and writers can be adjusted on the same way.

### 2.1.4 Creating the connections between data and transformers

The data is transferred from dataset/transformer to another by drawing a line between them (Picture 10). In Picture 11 we can see how the Line-geometry is transferred to *Tester* from *GeometryFilter*.



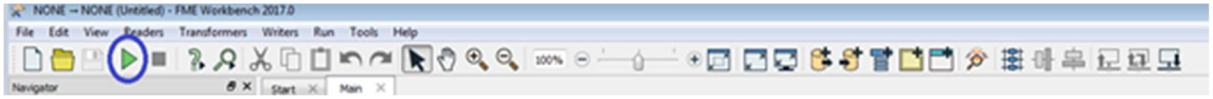

**Picture 10. Creating connection between source data and transformer.**




**Picture 11. Transferring Line-geometry from *GeometryFilter* to *Tester.***

### 2.1.5 Executing the translation

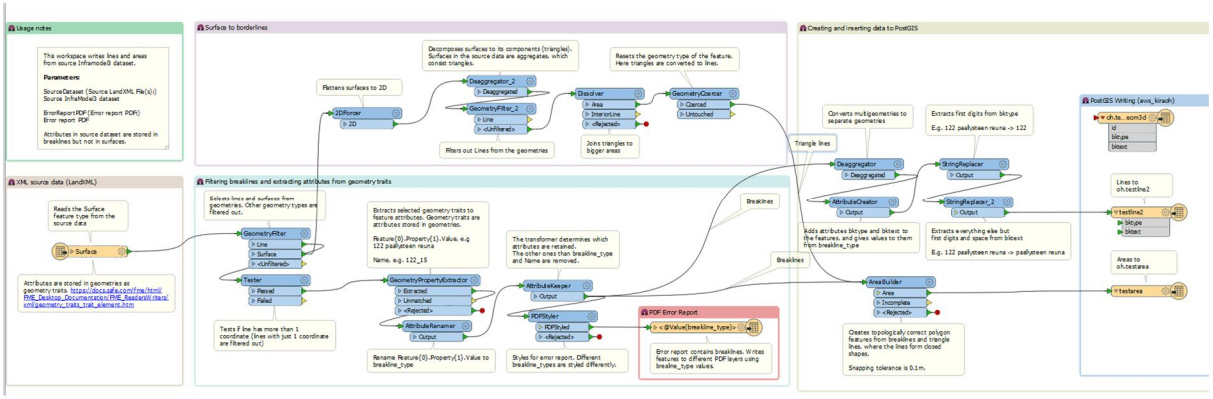
When all necessary translations and connections are created, the transforms can be executed by choosing *Run Translation* from the upper bar (Picture 12). If all the parts of the process are valid, the transformed source data is transferred to destination data.




**Picture 12. Executing the translation.**

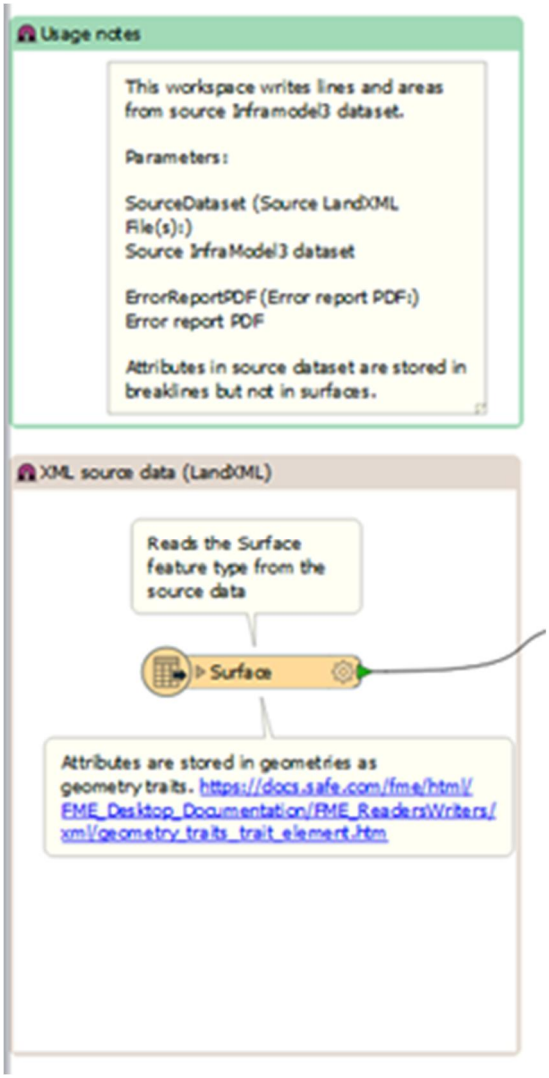
# 3 Source data transformation and exportation to destination data

In the tutorial below the Inframodel 3 -source material will be transformed to two-dimensional format. Single lines will be transformed to surfaces by the triangulation and the surface will be transferred to PostGis-database. The process is represented below in picture 13.

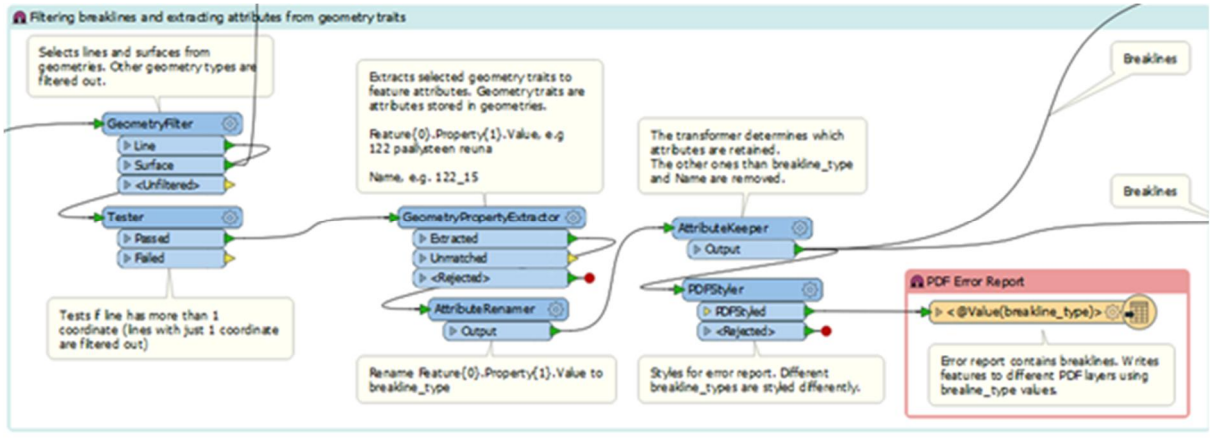



**Picture 13. Workspace of the whole process.**

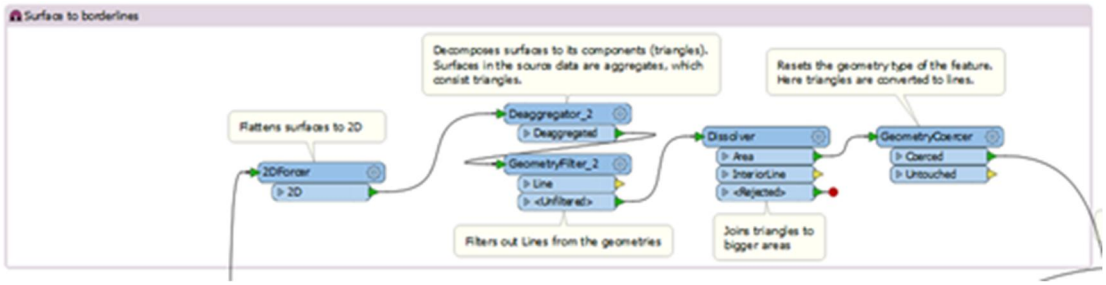
Next three pictures (14, 15 and 16) represents parts of the workspace so that the details can be observed.



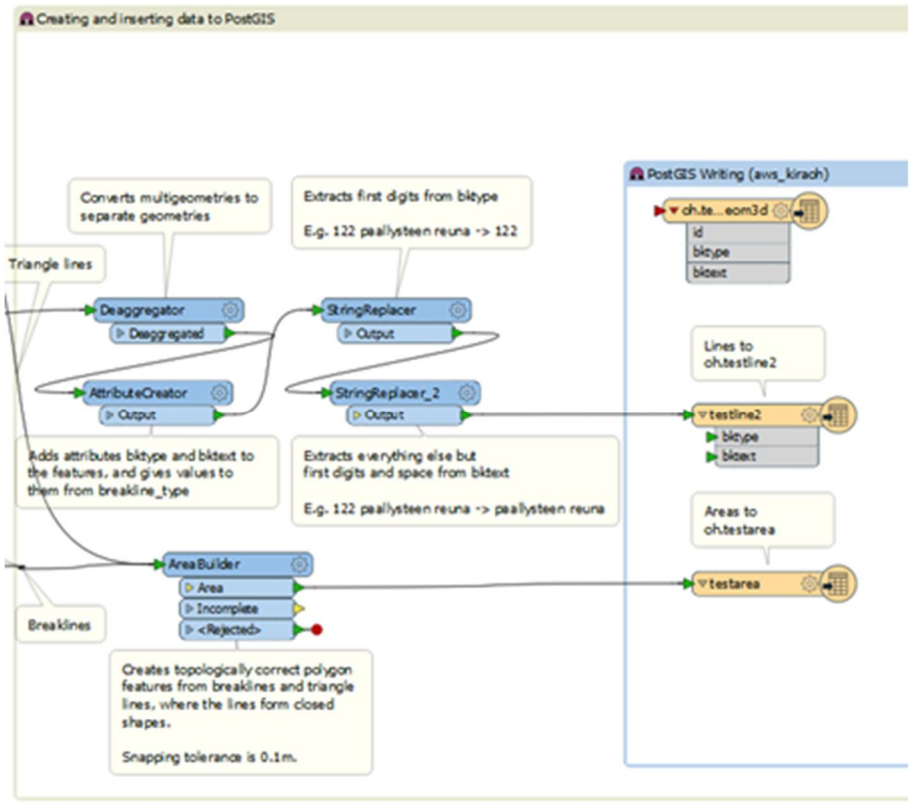

**Picture 14. Notes of the workspace and source data.**




**Picture 15. First part of the workspace.**

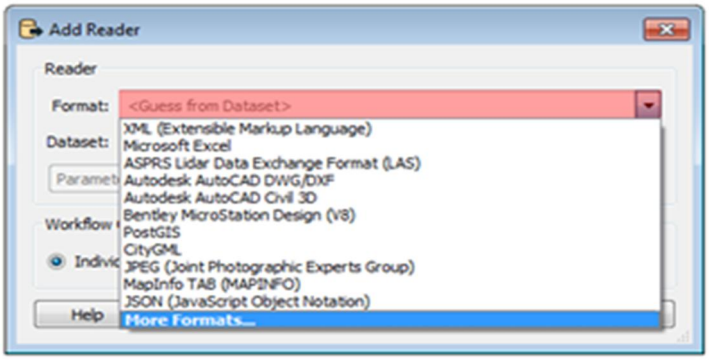



**Picture 16. Second part of the workspace.**




**Picture 17. Last part of the workspace.**

At the beginning of the process the source material must be chosen from the toolbar *Add Reader*. After that we will choose the format for the source material which is in this case LandXML. The format can be found from the downshift option *More Formats* (Picture 18). IM3-shaped source material is selected from the computer files (dataset).

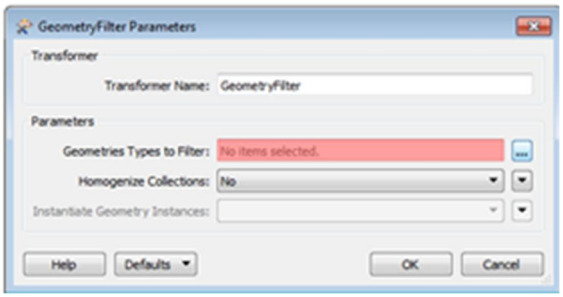



**Picture 18. LandXML is selected as format. The LandXML-format can be found from *More Formats* -option.**

The software will ask which features will be chosen from the source data. In this case, we will choose only *Surface* feature. After the selection, the source data (Reader) will emerge to the workspace.

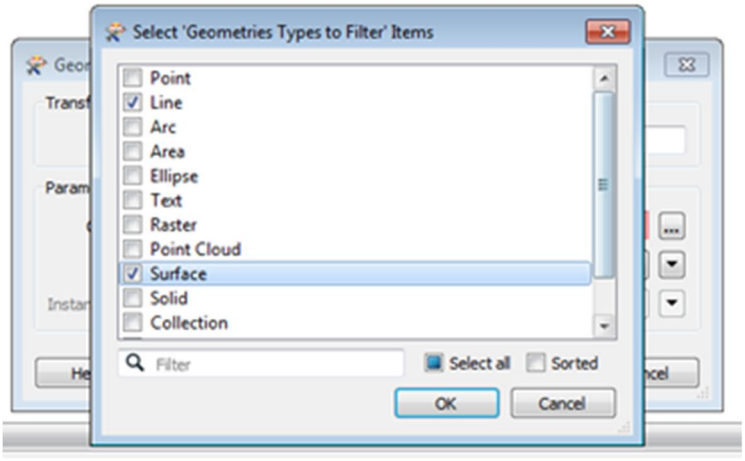
## 3.1 Phase 1: Filtering breaklines and choosing attributes

1. Several transformations must be done to the source data before it is ready to be transferred to PostGIS-database. For the transformations, we will use several tools (FME Transformers). First, we will treat data with *GeometryFilter* (*Add transformer -> GeometryFilter*). *GeometryFilter* is a tool which routes the features based on their geometry type.



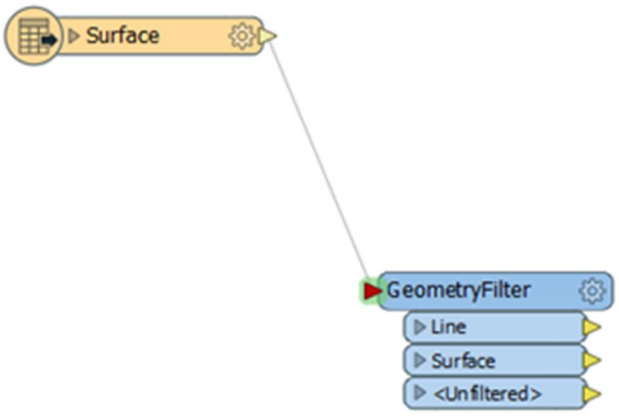

**Picture 19. Parameters of the *GeometryFilter* -tool.**

2. *GeometryFilter* -tool choose those geometry types which will be separated from the data. In this case, we will choose *Line*- and *Surface*- geometries (single lines and closing areas) and the transformer will delete other types of geometries.



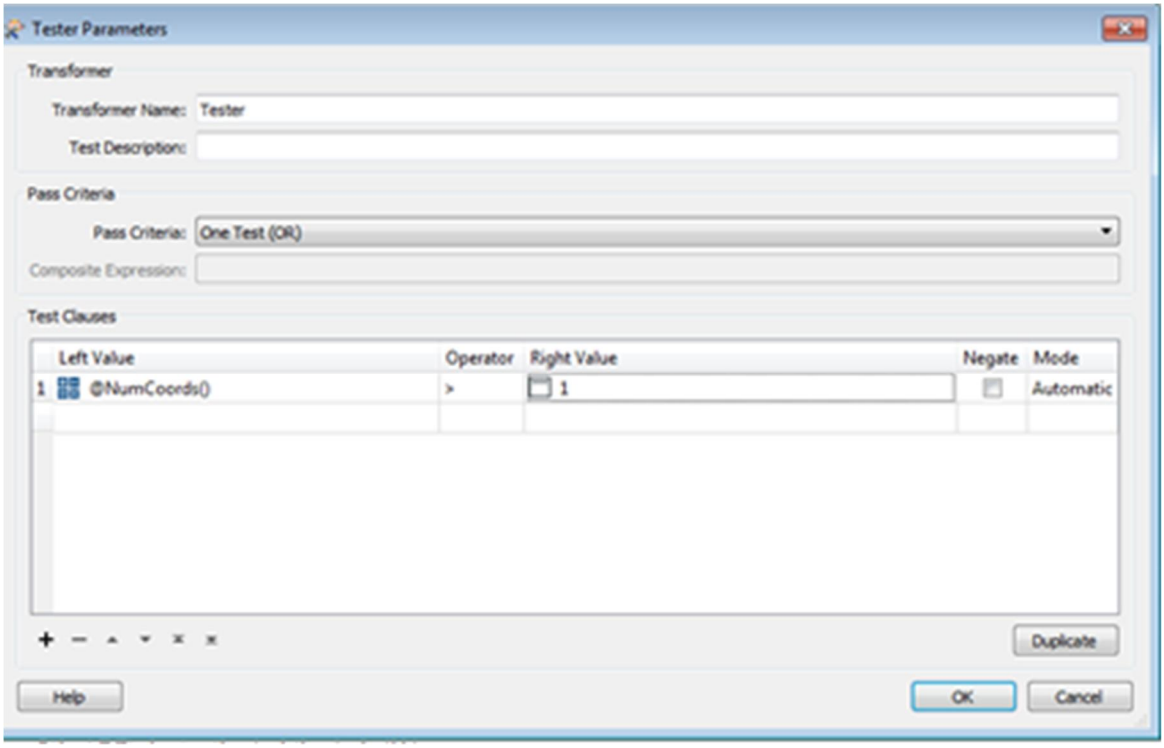

**Picture 20. Line and surface geometry types will be chosen from the option.**

3. When the user has accepted the options for *GeometryFilter*, the transformer will emerge to workspace. In this point, we will create connection between the source data and the transformer tool. The connection will be created by drawing a line between those two actors as in the picture 21.



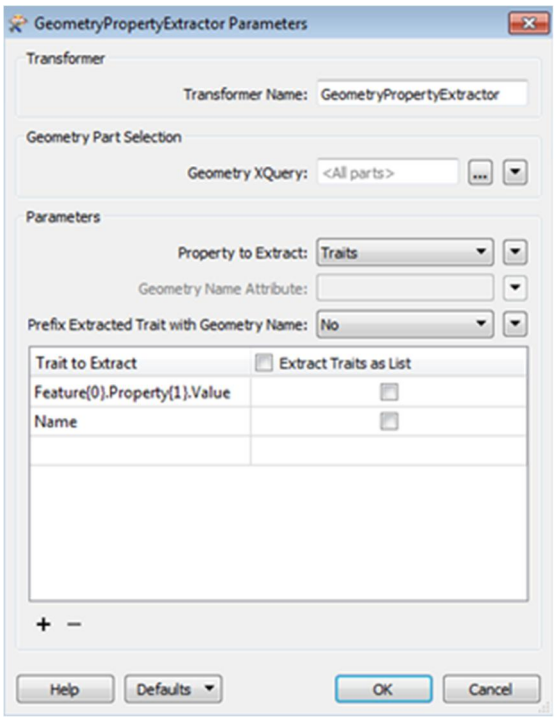

**Picture 21. Connection is created between the source material and *GeometryFilter* -tool.**

4. Next, we add *Tester*-tool, which evaluates tests on the target data and routes the features based on the test outcome. In this case, we will test if the *Line*-geometry has more than one coordinates. *Tester* will discard all the geometries which have only one coordinate.
   Picture 22 shows how the test is done by using *tester*-tool. Tested parameters can be determined by double-clicking *Left Value-, Operator-* and *Right Value*-options.



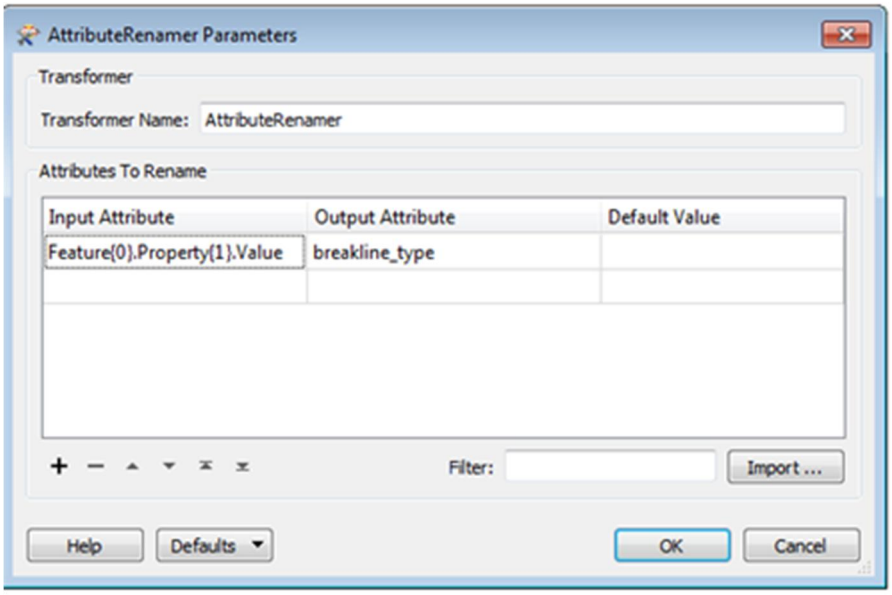

**Picture 22. Fill the options of *Tester*-tool.**

5. Next the data will be treated with the *GeometryPropertyExtractor*-tool. Tool modifies the metadata of the geometry (geometry traits) to normal attributes which can be used in the workspace. Picture 23 shows the parameters that should be chosen in *GeometryPropertyExtractor*.



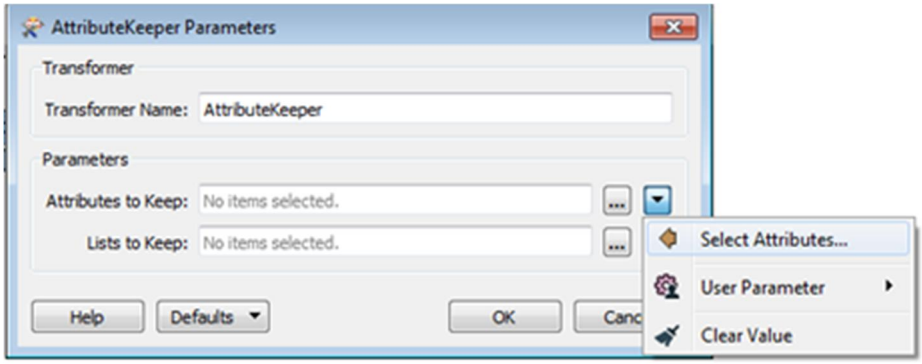

**Picture 23. Parameters in GeometryPropertyExtractor**

6. To be more clear, it is preferred to rename the attributes. Next, we transfer the data to the *AttributeRenamer.* The original attribute is placed to input column and in the output column we rename it (picture 24). The extracted attribute will be renamed with the *GeometryPropertyExtrractor* as breakline_type.

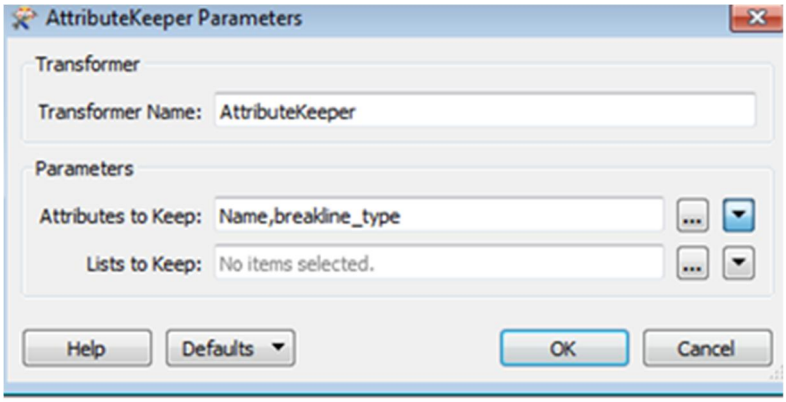



**Picture 24. Renaming the attribute.**

7. In *AttributeKeeper* we choose which attributes should be saved. In this case, we save attributes *Name* and *Breakline_type*. Transformer deletes rest attributes from use.

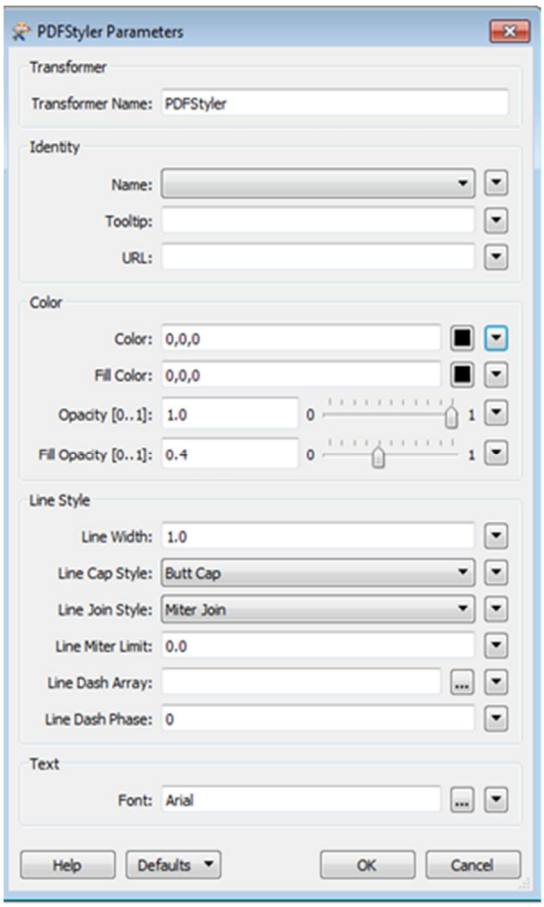



**Picture 25. The saved attributes are chosen in *AttributeKeeper "Select Attributes".***

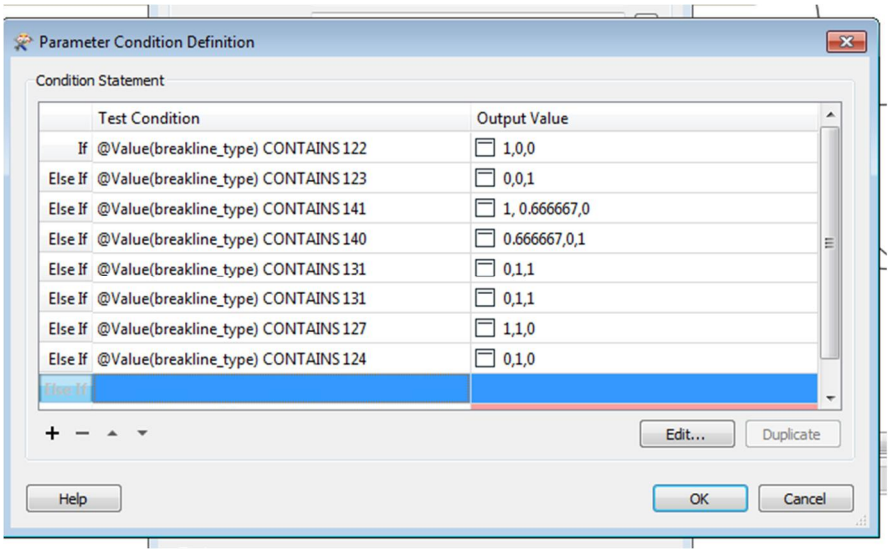



**Picture 26. We choose Name and breakline_type attributes.**

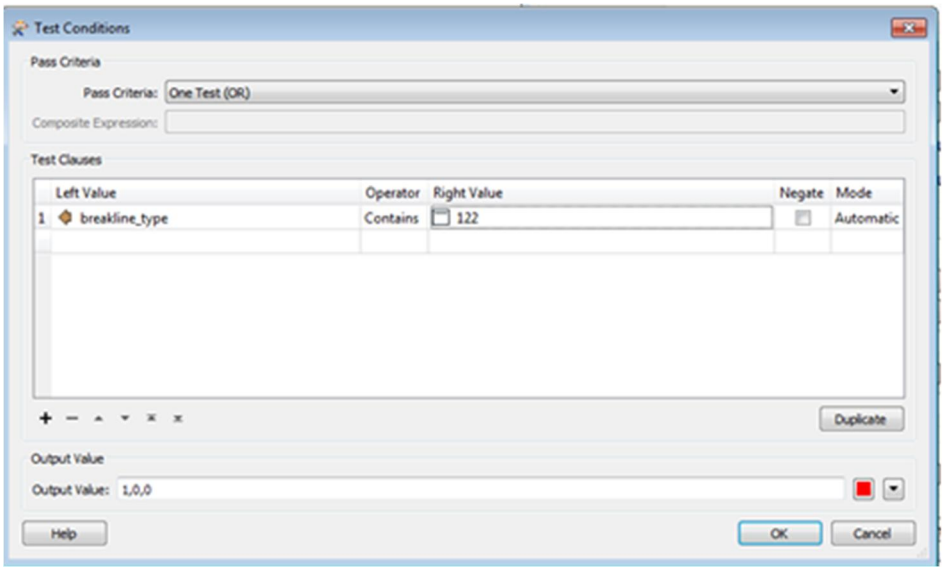
8. In the end of stage 1, we create error report, which illustrates data's breakline_type geometry. To create error report, we first use *PDFStyler* to make different styles for geometries. We give different colors to geometries per *breakline_type* -attribute. In PDFStyler Parameters -option we choose downshift button besides the color button and choose conditional value. Here we can choose color to each value as shown in pictures 28 and 29. By plus and minus buttons in the left down edge we can add more options to styles.




**Picture 27. PDFStyler parameters.**

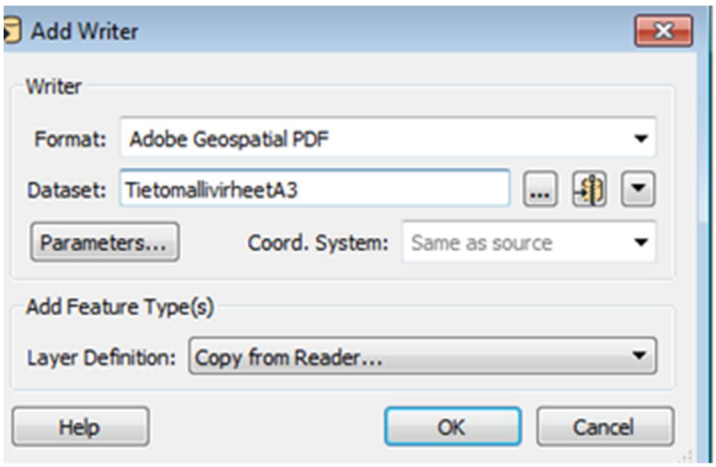



**Picture 28. Choose *Color -> Conditional Value* to determinate PDF-styles conditional expressions. Conditional expressions can be edited in *Edit* button.**




**Picture 29. Determining conditional expressions.**

9. For the error report, we add target material to the workspace (*Add writer*). File shape in this case is Adobe Geospatial PDF. We write the name of the target material to *Dataset*-field (picture 30).



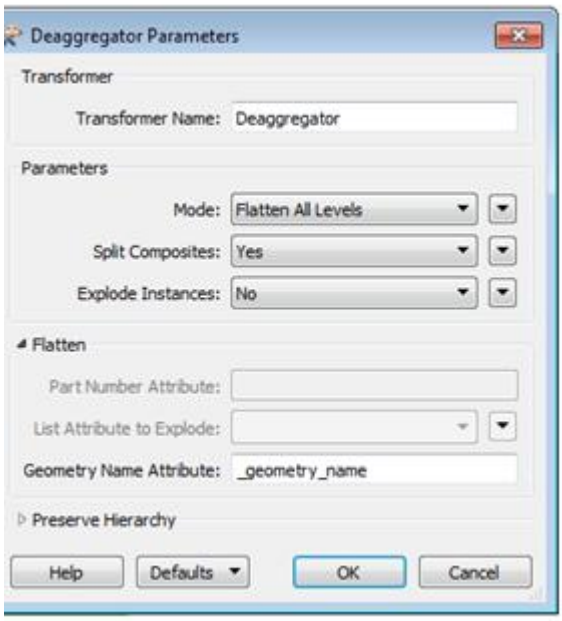

**Picture 30. Options to target material.**

We can consider the formed error report as PDF file.

## 3.2 Phase 2: Surface to borderlines

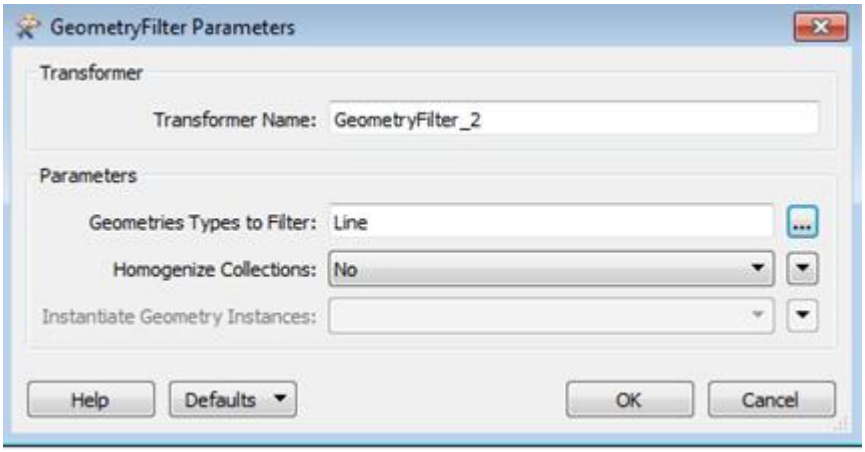
In this phase we transfer Surface-geometry of the source data to two-dimensional shape.

10. We choose *2DForcer* -tool and connect that tool to *GeometryFilters* surface-targets (which we created in phase 1). *2DForcer* flattens the data by deleting the z-coordinate.
11. The formed two-dimensional geometries are triangulated with *Deaggregator*-transformer. *Deaggregator* breaks geometry features to separate components (triangles). We chose parameters to transformer as in picture 31.



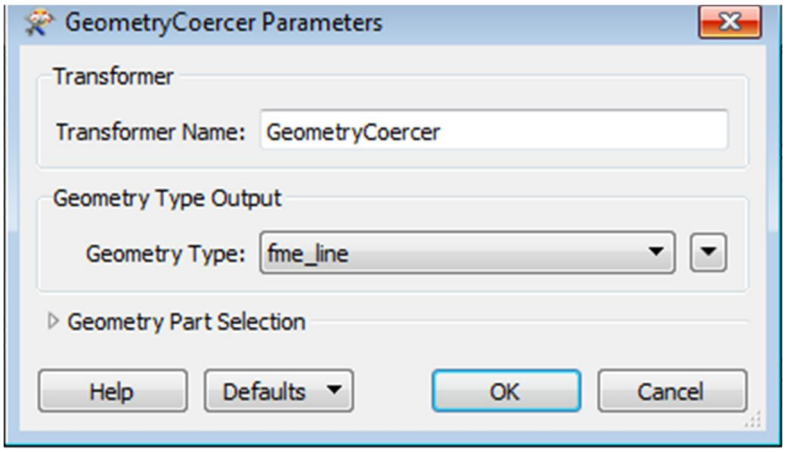

**Picture 31. Deaggregator parameters.**

12. Next, we filtrate Line-geometries from data with *GeometryFilter* (picture 32).




**Picture 32. Filtrate Line-geometries by using *GeometryFilter.***

13. All the other target (except Line-geometries) will be transferred to *Dissolver*, which connects the triangles to solid surface. These surfaces will be transferred to *GeometryCoercer*, which returns geometry type from triangles to lines (picture 33).




**Picture 33. In *GeometryCoercer* the Geometry type is fme_line. As a result, the geometry type of the data is changed to lines.**

## 3.3 Phase 3: Write data to PostGIS-database

In the last phase, the modified data will be transferred to PostGIS-database. In this case, we transfer the surface and line datas separated to PostGIS-database.

### 3.3.1 Surface data to PostGIS-database

14. First, we create topologically solid polygons with the *AreaBuilder*, where lines forms closed geometry shapes. Data will be transferred to *AreaBuilder* from

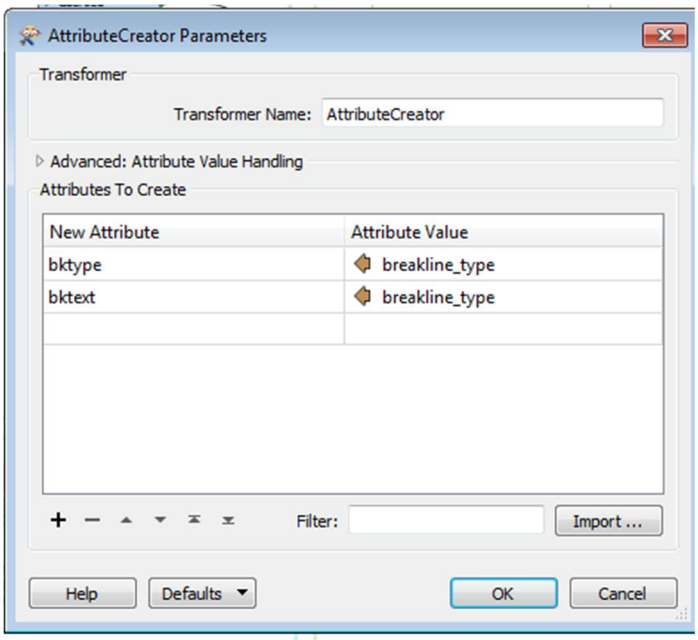
*GeometryCoercer* (trianglelines) and *AttributeKeeper* (breaklines) created before. Parameters in *AreaBuilder* will be determined as in picture 34.


**Picture 34. Parameters in *Areabuilder*.**

15. Formed areas will be transferred to PostGIS-database (Add writer → choose PostGIS as file shape and create database connection).

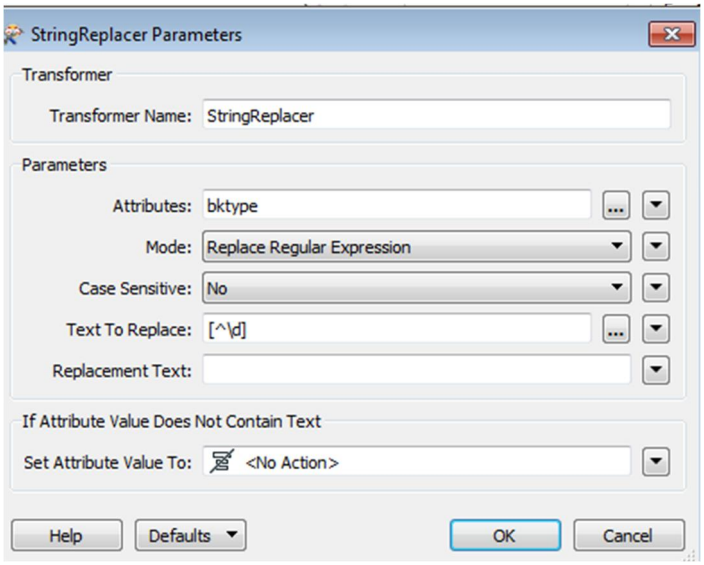
### 3.3.2 Line data to PostGIS

16. First, we transfer breakline-geometries from *AttributeKeeper* to components with *Deaggregator*. Transformer transforms multigeometries to separate geometries. Next, we add attributes to the data with *AttributeCreator*. We add bktype and bktext -attributes to the data by choosing parameters as in picture 35.

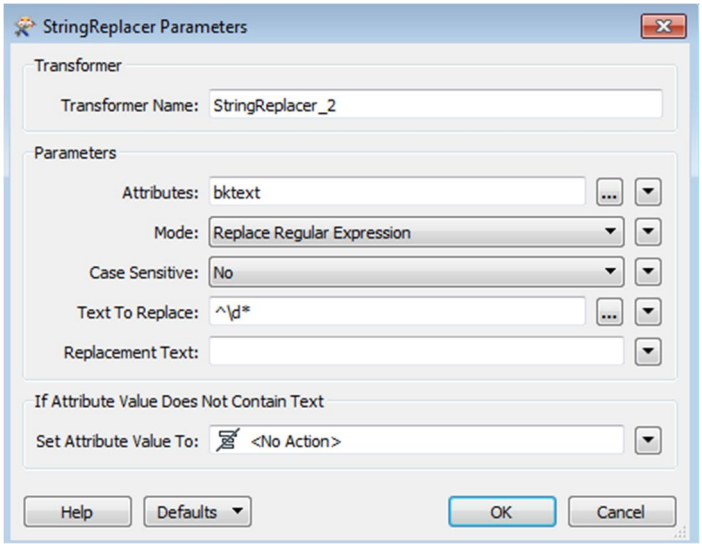



**Picture 35. Adding attributes with *AttributeCreator*.**

17. Before writing the data to PostGIS-database, bktype- and bktext -attributes will be modified so that the first numbers will be extracted from the bktype-attribute (for example 122 paving edge → 122). From bktext-attribute all features except numbers will be extracted (for example 122 paving edge → paving edge). After this data editing is ready.




**Picture 36. Extracting first numbers from bktype-attribute by using *StringReplacer*.**




**Picture 37. All other features will be extracted from the bktext-attribute expect numbers.**

18. Now data (bktext- and bktype-attributes) can be added to PostGIS-data by choosing first *Add Writer* and then PostGIS to target datas file shape.