**FME With Eurostat – Examples**

Version 1.0.0 (12-04-2023)

Contents

[Contents 1](#_Toc132277683)

[1 Getting started with the ExamplesEurostat.fmw. 2](#_Toc132277684)

[1.1 Requirements 2](#_Toc132277685)

[1.1.1 FME Desktop Software 2](#_Toc132277686)

[1.1.2 FME Desktop Knowledge 2](#_Toc132277687)

[1.1.3 OS Support 2](#_Toc132277688)

[1.1.4 Eurostat Access 2](#_Toc132277689)

[2 Getting the Examples and preparing them. 3](#_Toc132277690)

[2.1.1 EurstatLoopedLookUp.fmx 3](#_Toc132277691)

[3 Examples in the workspace. 3](#_Toc132277692)

[3.1 1) Standard reading and enriching the data from Eurostat. 3](#_Toc132277693)

[3.1.1 Adding Geometries 4](#_Toc132277694)

[3.1.2 Enhancing the code list values 4](#_Toc132277695)

[3.1.3 Pivot the data. 5](#_Toc132277696)

[3.2 2) Reading with a FeatureReader. 6](#_Toc132277697)

[3.3 3) Using N observations filter. 6](#_Toc132277698)

[3.4 4) Filter on Time Period. 7](#_Toc132277699)

[3.5 Parent/Child workspaces. 8](#_Toc132277700)

# Getting started with the ExamplesEurostat.fmw.

This getting started package comes with a workspace called “ExamplesEurostat.fmw”. This workspace is made in FME 2022.2 and should be opened in this version or a later version.

This workspace is designed to give you some examples of reading and processing data that comes from the new Eurostat reader.

It will show examples of different parameters that can be set, reading with both the FeatureReader and a normal reader, using the extra custom transformers and how to call several datasets with a workspace runner.

## Requirements

### FME Desktop Software

To use the package you do need the software FME Desktop 2022.2 (build 22765) or a later version. If you do not already have access to FME it can be downloaded here: <https://www.safe.com/support/downloads/>

### FME Desktop Knowledge

This document is not a complete FME tutorial and using the functions requires basic FME knowledge. By visiting the FME Community (<https://community.safe.com/>) you can find training, webinars and more resources to learn the basics.

### OS Support

The workspace has been tested both on 64-bit Windows and macOS. It should also work on Linux-versions supported by FME.

### Eurostat Access

All users may consult or download data and publications free of charge from the Eurostat web site. For more information on Eurostat’s policies and copyrights rules please visit: <https://ec.europa.eu/eurostat/web/main/about-us/policies/copyright>

# Getting the Examples and preparing them.

### EurstatLoopedLookUp.fmx

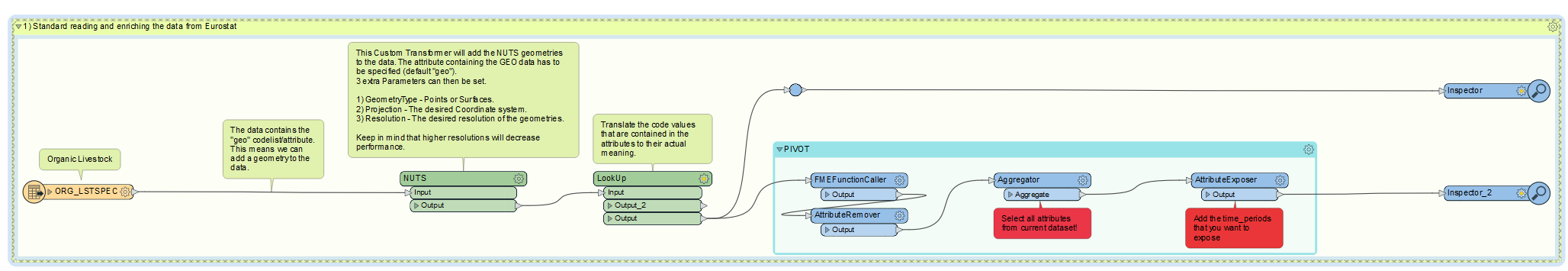
In order to run the ExamplesEurostat.fmw workspace, you first need to install the custom transformer “EurostatLoopedLookup.fmx”. This transformer is located in the /00) Install catalog. Right click it and choose “install with FME Workbench (Version)”. Make sure to choose a version of FME Desktop 2022.2 (build 22765) or higher.

Once this transformer is installed the “LookUp” custom transformer can be used in FME without missing dependencies.

# Examples in the workspace.

In this part of the document the different parts of the EaxmplesEurostat.fmw workspace will be explained. Why they are created and what they do with which settings.

## 1) Standard reading and enriching the data from Eurostat.



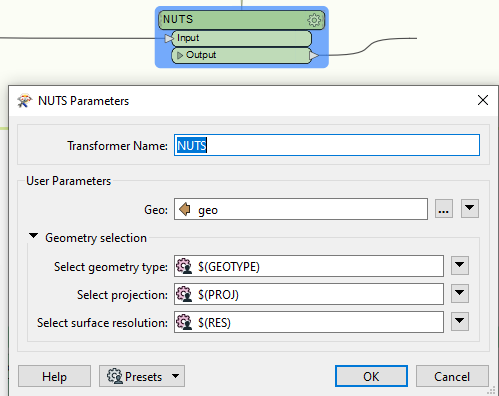
This example uses a standard reader to read the “ORGG\_LSTSPEC” dataset (Organic Livestock) with a standard reader. The webconnection used is linked to the ESTAT agency and all the parameters in the reader are left to their defaults.

This will result in the reader reading the full dataset with all its attributes. The FeatureType added has the same name as the dataset ID.

### Adding Geometries

The dataset has an attribute called “geo”. This attribute is often present in datasets and refers to the geographical NUTS areas. Based on this attribute in combination with the year of the time\_period attribute the correct geometry for that time can be added to the feature.

This is done by adding the “NUTS” custom transformer to the workspace. Simply connect the reader to it and specify in which column the GEO tag is located.



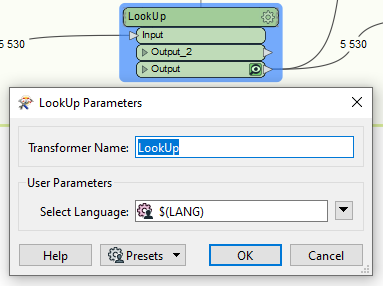
As extra parameters you can choose what type of geometry you want, points or surfaces. What projection you would like the data in and at which resolution. The resolution reduces performances at higher precision.

### Enhancing the code list values

Next up in this example, the “LookUp” custom transformer is used. In order to use this transformer, the previously mentioned: “EurostatLoopedLookup.fmx” transformer has to be installed on your version of FME.

This transformer convers the “code” attributes to their actual written value. This makes the values easier to interpret. For instance, it will change the value from the “freq” column from “A” to “Annual” and the “animals” column from “A1100” to “Live horses”.

This transformer is not fast since it must loop through all the attributes that are present to find the correct code list and value. Therefore, it is suggested to remove any attributes you don’t want to get the value from beforehand.



For this transformer you only set 1 parameter. Which concerns the language of the result. The output of the values can be either in English, German or French.

### Pivot the data.

The last part of this examples shows how to pivot the data.

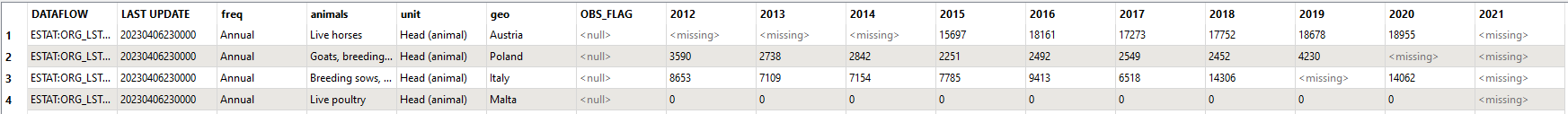
This is not done as a custom transformer since this part needs manual changes depending on the input data. When reading data from Eurostat, each observation will come in as a separate value in its own feature. However, often the only difference in the code values is the year of the observations.

Therefore, a pivot could be used. The FMEFunctionCaller will determine which columns we are going to get after the pivot and which value they are going to get. In most cases they can be left at “Time\_period” and “obs\_value”.

The Aggregator that comes after is deciding the group by attributes. For ease of use, all attributes can be selected. This will result in unique rows for each observation based on their attributes.

Lastly, the AttributeExposer is used to expose the new values. Here manual editing has to be done. Each year present in the dataset should be added to the AttributeExposer to get all the columns as an output. If no observation value existed for that year in the combination of all the attributes, the value will be set to <null>.

The result will look like:



## 2) Reading with a FeatureReader.

Reading data with the FeatureReader instead of a normal reader works the same.

You cannot use the spatial filter setting of the FeatureReader to limit the amount of data you read. By default, the datasets come without a geometry and are therefore not prepared for this functionality.

Regarding the output ports you do have a choice to use the generic output port vs a port per FeatureType. The only difference this will make is regarding the exposed attributes and the reading of data on a dynamic way (see example 5). The reader is designed to only read 1 FeatureType per reader. Therefore, you will at max get 1 output port per FeatureReader if you read the data on a normal way.

Only when using calling datasets in sequence you will benefit from using the generic port (and are required to).

## 3) Using N observations filter.

In this example the same dataset as before is read. This time the parameter for “First N Observations” and “Last N Observations“ is set to 1. This means that we should only get records for the first recorded time\_period and the last recorded time\_period in combination with their unique code list values.

It is very important to know that this does not equal features to read. In this example we get over 800 features as a result. This is because an observation is a unique combination of the codelists + time\_period. Using this parameter, the reader will also add the null values for time\_periods and obs\_values for those combinations that were missing them.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **DATAFLOW** | **LAST UPDATE** | **freq** | **animals** | **unit** | **geo** | **TIME\_PERIOD** | **OBS\_VALUE** |
| ESTAT:ORG\_LSTSPEC(1.0) | 20230406230000 | A | A6900 | HD | AT | <null> | <null> |
| ESTAT:ORG\_LSTSPEC(1.0) | 20230406230000 | A | A6710R | NR | AT | <null> | <null> |
| ESTAT:ORG\_LSTSPEC(1.0) | 20230406230000 | A | A6110 | HD | AT | 2012 | 3463 |
| ESTAT:ORG\_LSTSPEC(1.0) | 20230406230000 | A | A5140 | HD | AT | 2012 | 339089 |
| ESTAT:ORG\_LSTSPEC(1.0) | 20230406230000 | A | A5110OH | HD | AT | 2021 | 1052706 |

Looking at the example above, we can see 4 first observations and 1 last observations (this is not a representation of the full dataset). The first 2 rows both have a <null> value for time\_period and obs\_value. This means that initially they did not exist in the dataset, but they are now present to complete the time series. The reason they are added twice is because the Animals column and the Unit column are different.

Looking at row 3 and 4 we can see that both observations are for Austria in 2012. Here we see that they both count as a “first observation” because they are observations for different kind of animals.

So, when working with First N and Last N observations, it is important to keep in mind that an observation is the unique combination of the codelists for a certain time\_period.

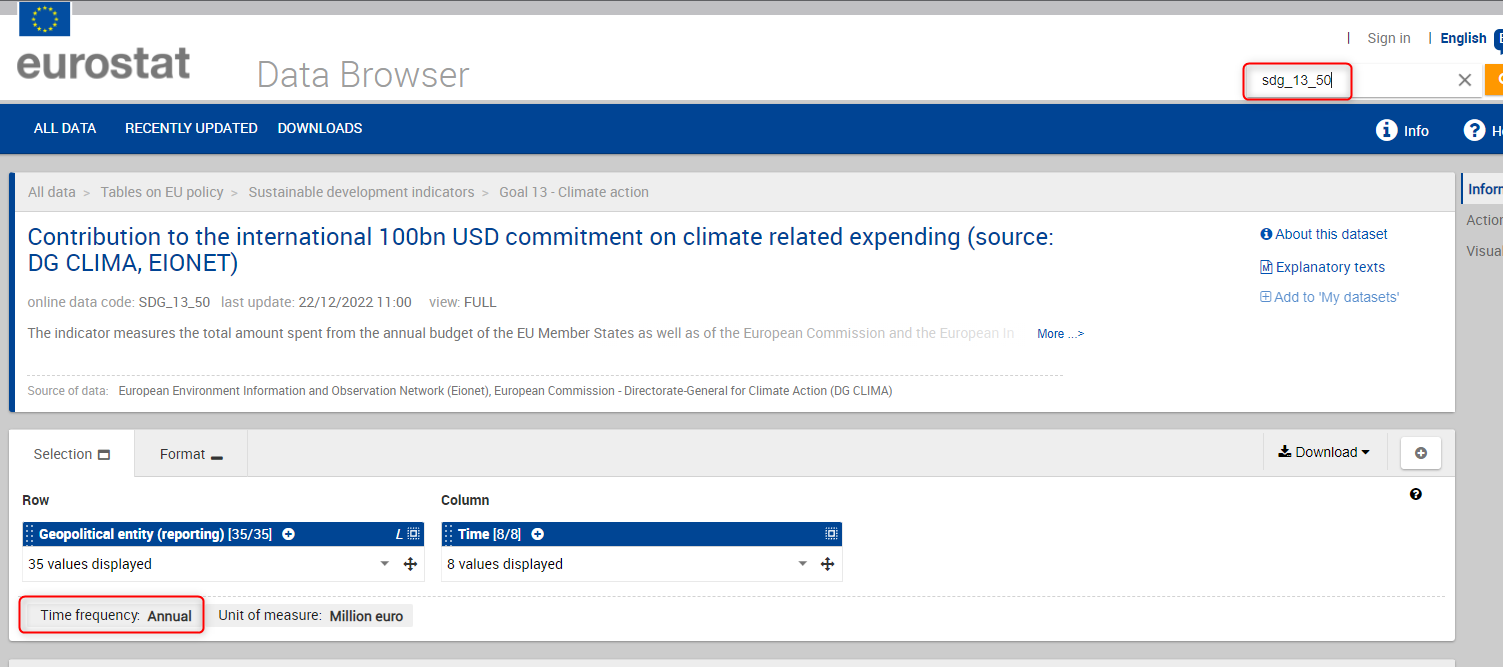
## 4) Filter on Time Period.

The 4 small examples in this part of the workspace show how you can use filters on start\_period and end\_period. An example is given for, Year, month, and a day filter.

The filter used on time period is very strict and can be hard to set properly since the dataflow itself doesn’t specify what it accepts in the name itself. A dataset that is called “Daily cigarette usage” doesn’t per say accept a daily filter.

To see which format of filter is allowed, it is suggested to look at the [Databrowser](https://ec.europa.eu/eurostat/databrowser) of Eurostat.

Here you can search for the corresponding dataflow either by name or code. The databrowser will then refer to the “Time Frequency” to help you choose the correct format for filtering.



**Filter on Year:**

For this example, Daily smokers of cigarettes by sex, age and degree of urbanization is read. Even though the name contains “daily” the time frequency that allows filter is “annual”. Therefore a filter based on year is used.

In this example the start period and the end period are both set to 2014. This means the data that is read is either equal to or bigger than 2014 and equal to or smaller than 2014. Resulting in data from only 2014. To set this, you can simply write the year in for form of YYYY in both the filter settings.

**Filter on Month:**

For this example, Euro yield curves – Monthly data is read. In this case the data is using a time frequency of Month. Therefore, the month filter can be used on this dataset.

When using the month filter, you must also define the year. There are two ways to write the filter:

1. 2016-04
2. 2016-M04

Both will result in the same data.

**Filter on Day:**

For this example, Euro yield curves – Daily Data is read. This dataset is based on daily data and can therefore use a daily filter. For the Start and End Period you can simply write the day(s) you want by writing out the full format as: yyyy-mm-dd, for instance: 2023-03-28.

## Parent/Child workspaces.

There are cases where you would like to read data from Eurostat dynamically in the form of a Parent/Child workspace structure. This is possible but requires some extra work.

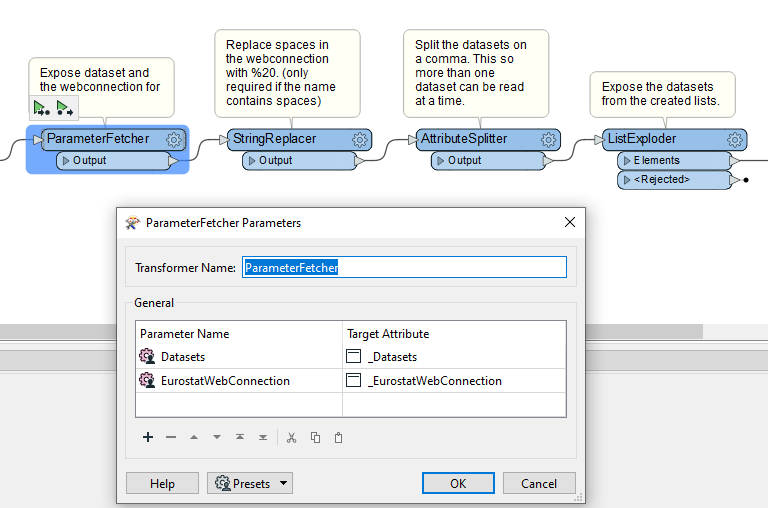
First design the child workspace:

The child-workspace will require certain user parameters depending on how dynamic the workspace must be.

1. Eurostat webconnection – The Eurostat reader uses a webconnection. This one can be hardcoded.
2. Dataset – You will have to specify what dataset to read. It is recommended to use a text value parameter here and use a clear description. For instance, that the user must separate the datasets with a “,” if they want to read several datasets. Keep in mind that the Dataset ID is required.
3. First N observations – if required.
4. Last N Observations – if required.
5. Start Period – if required.
6. End Period – if required.

In the child workspace, before reading the data, make sure the parameters are set correct. You will need the Dataset as an attribute if more than one dataset is used.

The webconnection also must be checked to make sure it does not contain spaces.



In the example above, the datasets and webconnection are exposed. The webconnection contains a space and should be replace with %20. Hence the StringReplacer.

In this example, 2 datasets are read. They come in via the Datasets user parameter and are separated by a “,”. The attribute splitter is used to create a list and they are then exposed by using a ListExploder. This results in 2 features, 1 per dataset.

After you’ve controlled the user parameters and exposed the correct settings you can use the FeatureReader to read the data dynamically.

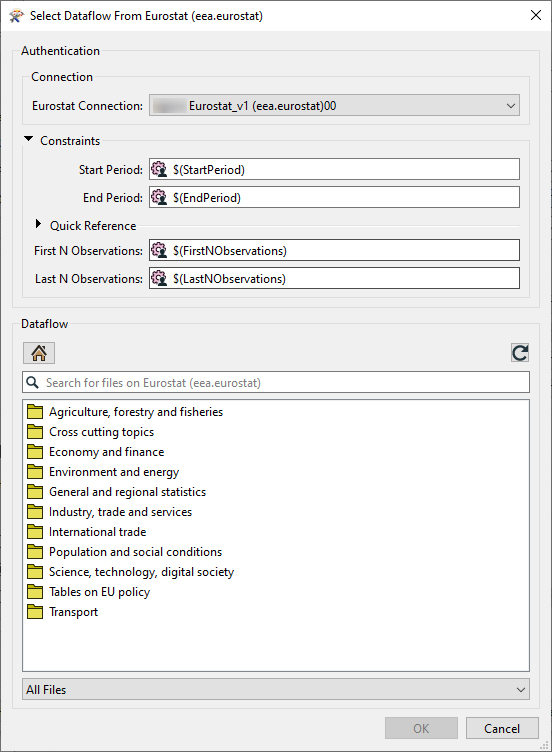
The easiest way to achieve this is by following these steps:

1. Choose the Eurostat Format
2. Select File from Web (Eurostat) like you would normally do for this reader.
3. For now, choose the Webconnection from the list you have available.
4. If you have to set constraints make the following steps, you can skip these if you don’t use constraints:
   1. For start period, click in the field and write: $(UserParameterName)
   2. For End period, click in the field and write: $(UserParameterName)
   3. For First N Observations, click in the field and write: $(UserParameterName)
   4. For Last N Observations, click in the field and write: $(UserParameterName)

By wring the syntax like this, FME will automatically understand that you want to use an user parameter in this field.

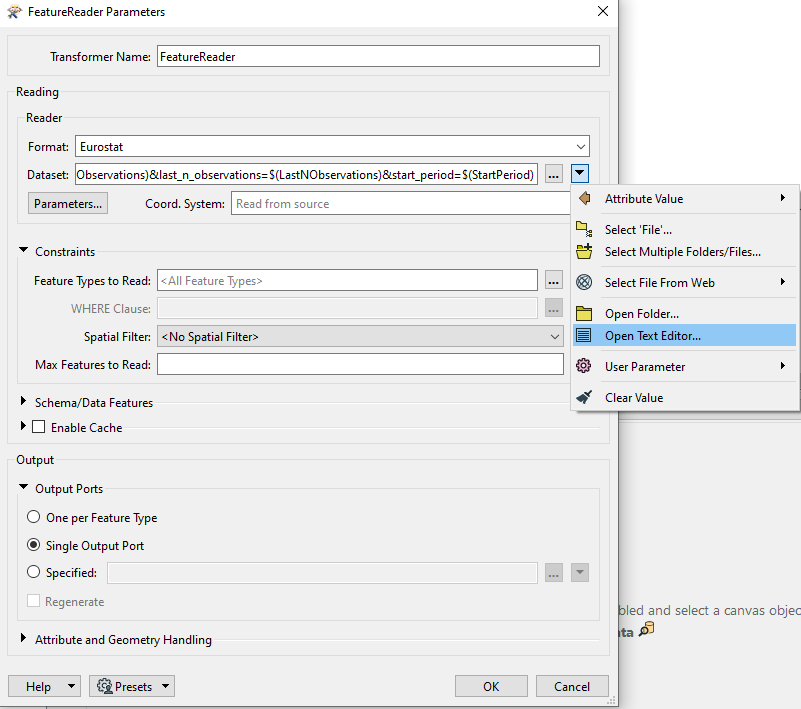
1. For now, choose any random dataset.
2. Click on “OK”

Your settings should now look something like:



Once you’ve selected a random dataset and clicked on OK you will be navigated back to the FeatureReader settings screen.

Here, click on the arrow next to the dataset and click on “Open Text Editor”

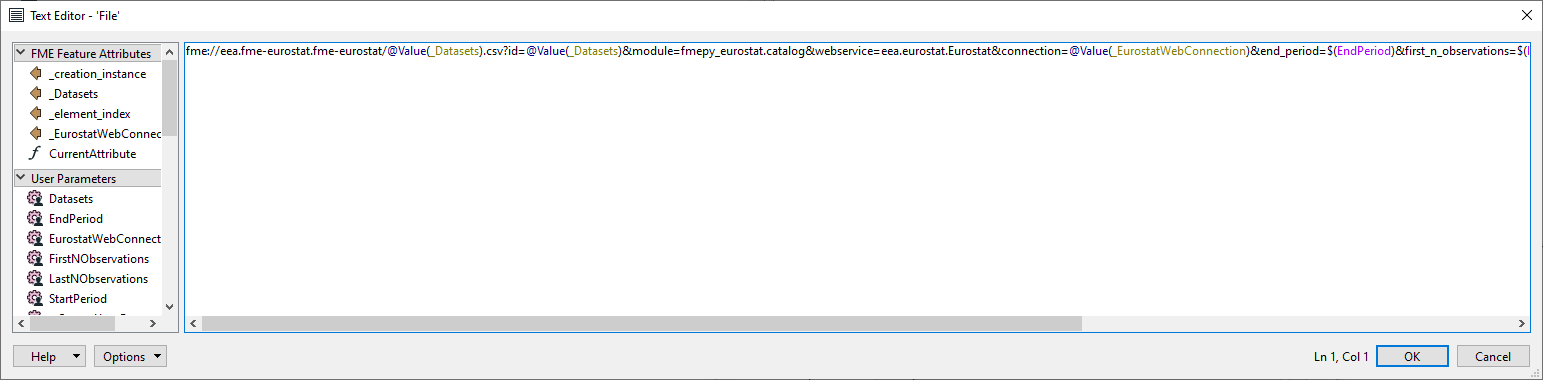


In this text editor, replace the dataset id that comes after “fme://eea.fme-eurostat.fme-eurostat/” and before “.csv” with the attribute that contains your dataset id.

Do the same for the dataset ID that comes after “?id=”.

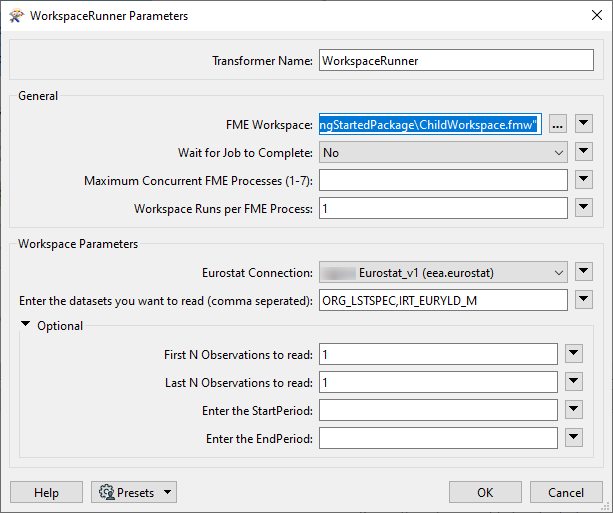
You also must change the webconnection if you don’t use it hardcoded by changing the value after “connection=” with the attribute containing your webconnection name.

The result you get should look something like this:



The last setting you must set is to enable “Single Output Port” under “output” settings in the FeatureReader. This since you don’t know what datasets will be read. Therefore, we cannot specify one feature per output port.

Now that the required basics of the child workspace are set, you can create the Parent workspace. As shown in the “02) ExamplesEurostat.fmw”.



In this example, the child workspace will read 2 datasets: ORG\_LSTSPEC,IRT\_EURYLD\_M. For these datasets it will only read the first and last observations.