Article

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

Dynamic Workflows with the SchemaScanner

Aug 4, 2022 • Knowledge

Product Type
FME Desktop

FME Version
2022.0

Previous: Dynamic Workflows: Components (https://community.safe.com/s/article/Dynamic-Workflow-Components) | **Next:** Dynamic Workflows with a Multi-Dataset SchemaScanner (https://community.safe.com/s/article/Dynamic-Workflows-with-a-Multi-Dataset-SchemaScanner)

Introduction

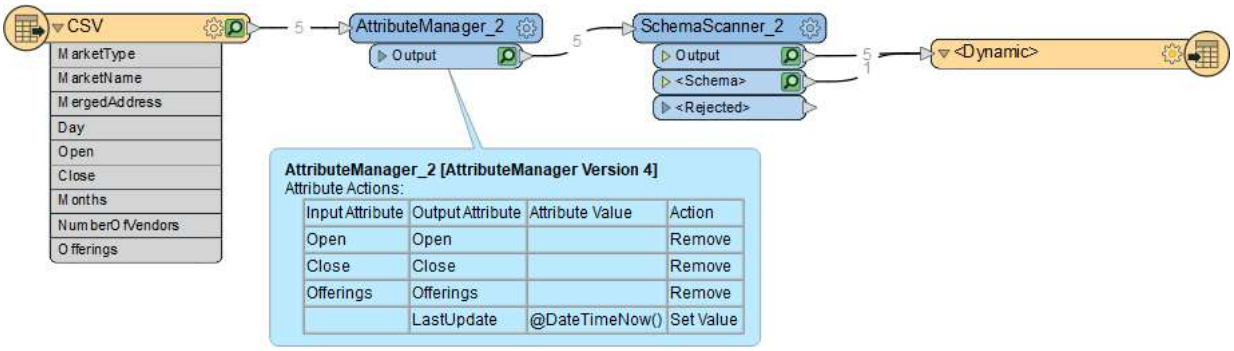
As we've seen, dynamic workflows can obtain their schema from multiple locations.

One of those locations is in the workspace itself, and the *SchemaScanner (https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Transformers/Transformers/schemascanner.htm)* transformer is a key tool in making that happen.

The SchemaScanner - as the name suggests - scans incoming features and produces a schema from them. That schema can then be used in a dynamic writer. The schema is stored in a specific FME feature.

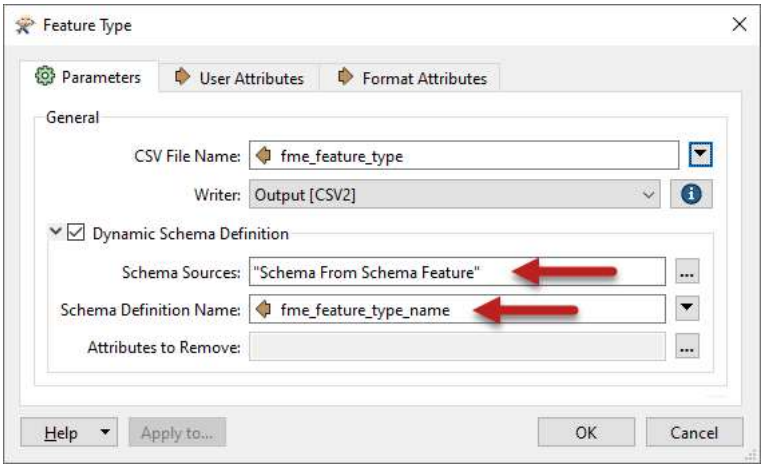
The schema produced by this transformer may be different from the reader schema, as a result of processes in the workspace; such as attribute renaming, removal, or addition.

In this workspace, attributes are being added and removed from the reader schema



The SchemaScanner generates a new schema for the output, assigns it a name, and passes it to a dynamic writer. Notice how the schema feature passes from the SchemaScanner:Schema output port to the same writer input port as incoming data.

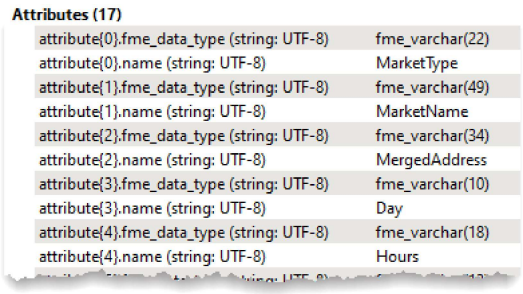
The dynamic writer is set up to recognize incoming schema features and will make use of them:



Notice how the Schema Source is set to “Schema From Schema Features” to inform the writer from where the schema is to be obtained. Also, notice the parameter that defines the name of the schema. This handles the situation where the same writer is fed multiple schema features.

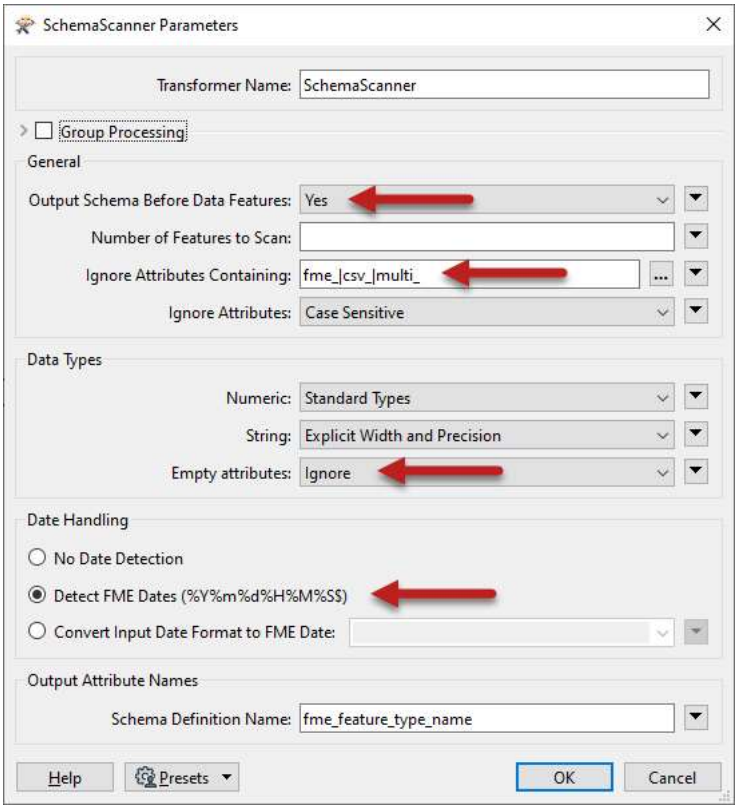
Now the data will be written to the output CSV dataset with the schema as modified by the AttributeManager.

Definition: A Schema Feature stores information about a schema that can be passed to a writer. The information is held as a series of list attributes called *attribute{}.name* and *attribute{}.fme_data_type*



Schema features are generated by the SchemaScanner transformer, but also by the FeatureReader transformer, Schema (Any Format) reader, and even the AttributePivoter transformer!

Going back to the SchemaScanner, let’s look at the parameters:



The important things to note here are:

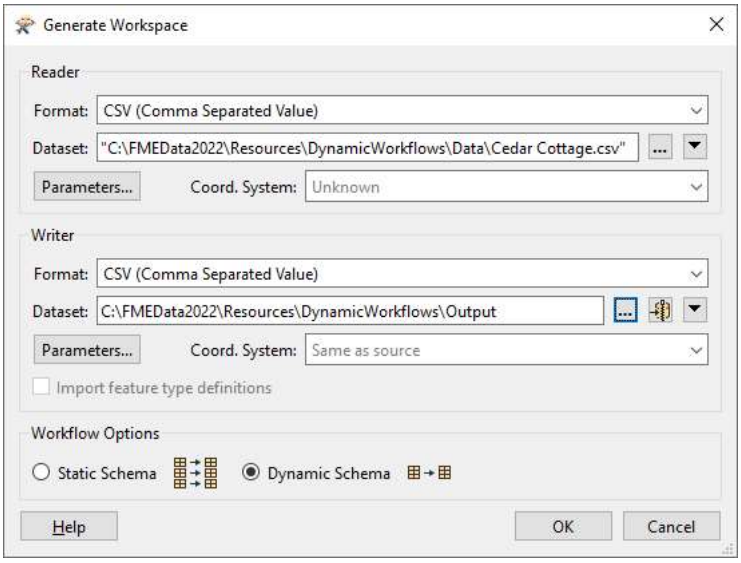
- The transformer is set up to output the schema feature before any data features. This is vital. The writer must receive the schema feature before any data features that make use of it.
- The transformer is set up to exclude from the schema any attributes that contain fme_, csv_, or multi_. These are format attributes that we don't really need in the output.
- Attributes with empty (or null) values will be ignored; i.e. excluded from the output schema, just as if they'd been removed in the AttributeManager
- The transformer has been instructed to detect dates in the incoming data. If this had not been set, values such as 20220812 would be interpreted as a number, not a date.

Step-by-step Instructions

The goal here is to transform some CSV data, but with a dynamic schema to take account of changes made to the data within the workspace itself.

1. Generate Workspace

Generate a workspace to translate from CSV to CSV. Set the Reader Dataset to Cedar Cottage.csv which can be found in the downloaded Files. Set the Writer Dataset to an output location. Finally, set the Workflow Options to Dynamic Schema.



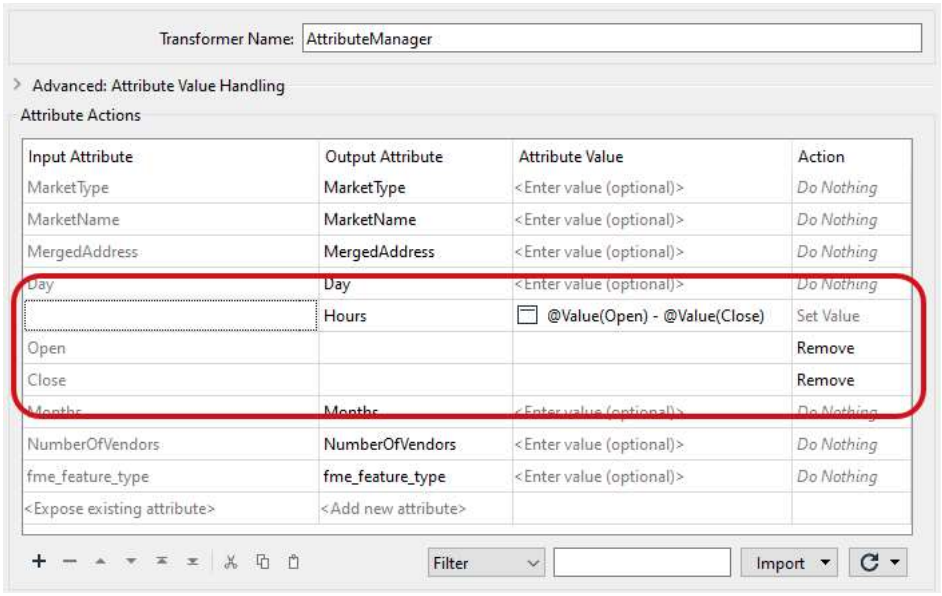
Run the workspace to load the data and inspect it to see what we have.

2. Create New Attributes

Add an AttributeManager transformer and connect it between the reader and writer feature types. We'll use it to create a new attribute called Hours, which is a combination of the Open and Close attributes. Open the Text Editor for the Hours Value and set it to:

```
@Value(Open) - @Value(Closed)
```

Delete the Open and Close attributes, since we no longer need them.



Be sure to create “Hours” above “Open” and “Close”; otherwise the deletion occurs first and Hours has no content. You can use the Up/Down arrows to move attributes.

You can run the translation again at this point, but the output will still have the same schema. That’s because it’s still coming from the reader, and proves why we need the SchemaScanner.

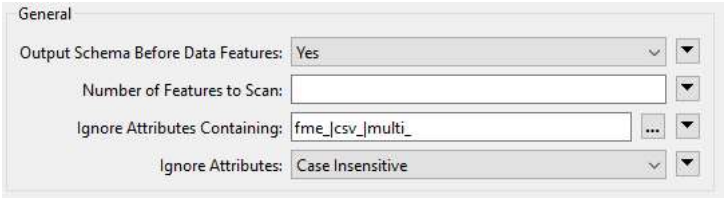
3. Add a SchemaScanner Transformer

Place a SchemaScanner transformer. Ensure it is after the AttributeManager and connect both the Output and <Schema> ports to the writer feature type.

Check the transformer parameters. Ensure that Output Schema Before Data Features is set to Yes.

For Ignore Attributes Containing, enter:

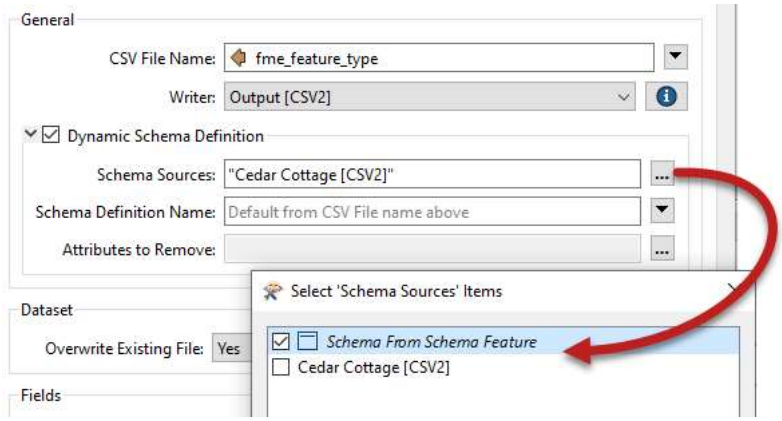
```
fme_|csv_|multi_
```



4. Modify Writer Parameters

Now open the parameters for the <dynamic> CSV writer feature type

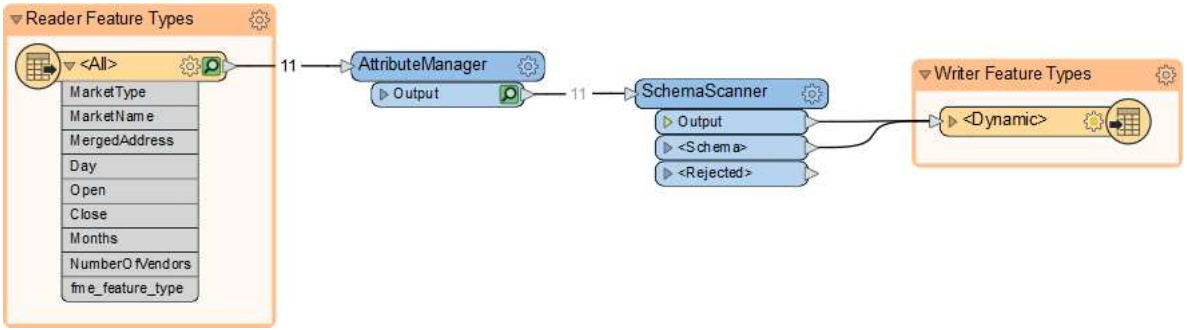
Click the ellipsis [...] button to the right of the Schema Sources parameter. In the dialog, uncheck the box for Cedar Cottage and check the box for Schema From Schema Feature:



Now set the Schema Definition Name parameter to the attribute created by the SchemaScanner; fme_feature_type_name

5. Run Workspace

Your workspace should now look like this:



Run the workspace and inspect the output. It will look like this:

	MarketType	MarketName	MergedAddress	Day	Hours	Months	NumberOfVendors
1	Community Food Market	Alternates between 4 locations	<null>	Saturday	10:00am - 12:00pm	<null>	<null>
2	Community Food Market	South Vancouver Neighbourhood House	49th and Victoria	Sunday	9:00am - 11:00am	<null>	<null>
3	Community Food Market	Brant Villa	2290 E 2th Av, Vanc...	Saturday	10:30am - 12:30pm	Ongoing	2-3 Tables
4	Community Food Market	Cedar Cottage Neighbourhood House	4065 Victoria Dr, Va...	Saturday	10:00am - 12:00pm	Ongoing	2-3 Tables
5	Community Food Market	Culloden Court	1375 E 47th Av, Van...	Saturday	2:00pm - 4:00pm	Ongoing	2-3 Tables
6	Community Food Market	Orchard Park Mobile Market and Tasting Kitchen	5988 Nanaimo St, V...	Saturday	2:00pm - 4:00pm	May-December	1
7	Community Food Market	Cedar Cottage Neighbourhood House Mobile Market	4065 Victoria Drive, ...	Saturday	10:00am - 12:00pm	May-December	1
8	Community Food Market	Kensington Community Centre	5175 Dumfries Street	Sunday	9:00am - 11:45am	<null>	2-3 Tables
9	Community Food Market	Orchard Park	5988 Nanaimo St, V...	Wednesday	2:00pm - 4:00pm	Ongoing	<null>
10	Community Food Market	Culloden Court Mobile Market and Tasting Kitchen	1375 E 47th Av, Van...	Wednesday	2:00pm - 4:00pm	May-December	1
11	Community Food Market	Brant Villa Mobile Market	2290 E 25th Av, Van...	Saturday	10:00am - 12:00pm	May-December	1

Note that the Open and Close attributes are not there, but there is an attribute called Hours.

6. Save Workspace

Save the workspace. We'll use it as the starting point for the next exercise (https://community.safe.com/s/article/Dynamic-Workflows-with-a-Multi-data-SchemaScanner).

Continue to the next article: Dynamic Workflows with a Multi-Dataset SchemaScanner (https://community.safe.com/s/article/Dynamic-Workflows-with-a-Multi-Dataset-SchemaScanner)

Data Attribution

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

Log In to Comment

Follow

Files (1) (/s/relatedlist/ka14Q000001DWyxQAG/AttachedContentDocuments)

3-WorkflowSchemaScanner
Aug 4, 2022 • 29KB • zip

View All
(/s/relatedlist/ka14Q000001DWyxQAG/AttachedContentDocuments)

Related Articles

Tutorial: Dynamic Workflows (/s/article/dynamic-workflow-tutorial-introduction)

Dynamic Workflows: Advanced Example - Modifying the Schema Feature (/s/article/dynamic-workflows-advanced-example-modifying-the-s)

Dynamic Workflows: Destination Schema Derived from Multiple Sources (/s/article/dynamic-workflows-destination-schema-derived-from)

Dynamic Workflows: Basics of Dynamics (/s/article/dynamic-workflow-tutorial-basics-of-dynamics)

Dynamic Workflows: Destination Schema as a Mirror Image of the Source Dataset (/s/article/dynamic-workflow-tutorial-destination-schema-as-a)

Land Acknowledgement —

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