Technical Forums (/s/forums) Knowledge Base (/s/knowledge-base) Ideas (/s/bridea/acideasULD_bı Log in Q0(

Article

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

Dynamic Workflows: Dealing with Schema Drift

Aug 4, 2022 • Knowledge

Product Type
FME Desktop

FME Version
2022.0

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

Introduction

In many workplaces, data sources and their schemas can change without notice. Attributes and data types can be added, removed, and changed depending on who is managing and uploading the data.

Imagine a scenario where a new hire doesn't know the set schemas for current datasets and tries to update a dataset with some newly created data. Now, you either have a dataset with unwanted additional columns that are just a repetition of the old columns, or you're missing data in your existing dataset because the schemas didn't match. This can be a huge headache for whoever is managing the data in the long term, and nobody likes going back and correcting datasets.

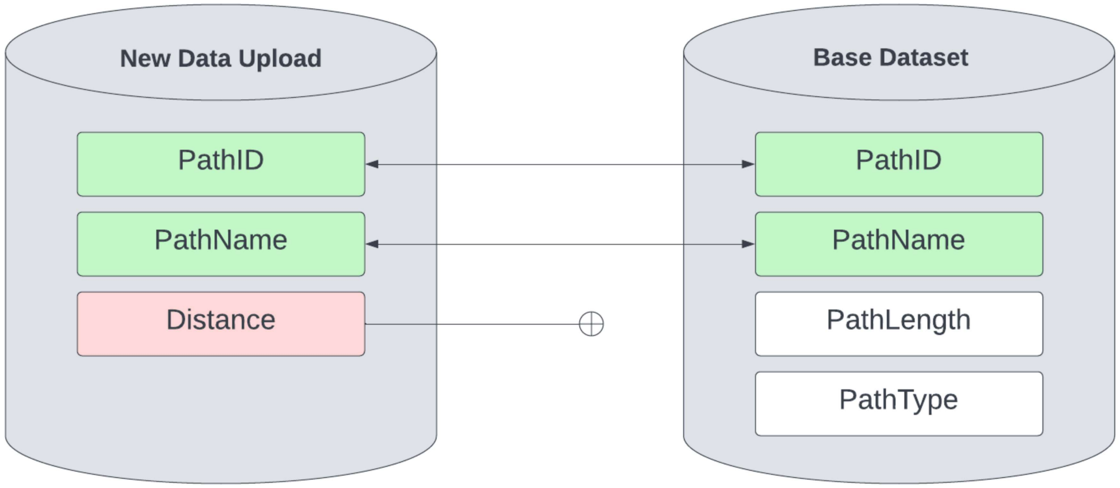
That's why with the SchemaScanner (https://docs.safe.com/fme/2021.1/html/FME_Desktop_Documentation/FME_Transformers/Transformers/schemascanner.htm) transformer, we can be proactive by detecting and reacting to schema drift in our datasets. These two essential steps make it easy to adapt to changing schemas with FME.

For example, we have a dataset in our database with a table containing the following attributes:

- PathID, PathName, PathLength, PathType

Every few days, an employee uploads a new set of recordings to add on top of this data. The employee may not be familiar with the original dataset and has made some assumptions in naming the attributes. Their dataset contains the following attributes:

- PathId, PathName, Distance



In this case, the schema has drifted, with the PathType attribute missing and the PathLength attribute having the incorrect name: Distance.
Before the user is able to upload the data to the existing dataset, we want to be able to detect changes in the schema so we can notify the user to change their input data schema.

This is where the SchemaScanner comes into use. It allows us to extract the schema of the incoming dataset and compare it against the known, existing, dataset before allowing the data to be uploaded. If the schemas match, then all is good to go and we can allow the data to flow freely into the base dataset. However, if the data's schema doesn't match the base dataset, then we need to react to what is going on here. In our case, we're going to create a Schema Change Report that we can then feed back to the user whether that be through an email or through an FME Server App.

Content Overview

Step-by-step Instructions

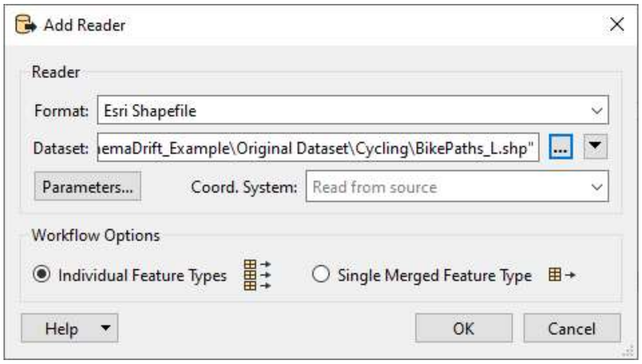
Part 1: Detecting Schema Drift

The first step in our example is to detect the schema drift. We will do this using the SchemaScanner and the ChangeDetector transformer to detect changes between the original schema and the schema of the data being uploaded.

Before beginning, download the SchemaDrift_Datasets.zip file located in the Files section of this article.

1. Open FME Workbench
First, open up a blank workspace in FME Workbench 2022.0 or newer.

2. Add a Shapefile Reader
Add a reader to the workspace and set the Format to Esri Shapefile. Select the Dataset as BikePaths_L.shp, which was included in the downloaded file. Press OK to add the reader to the canvas.



Add Reader

Reader

Format: Esri Shapefile

Dataset: ıemaDrift_Example\Original Dataset\Cycling\BikePaths_L.shp"

Parameters... Coord. System: Read from source

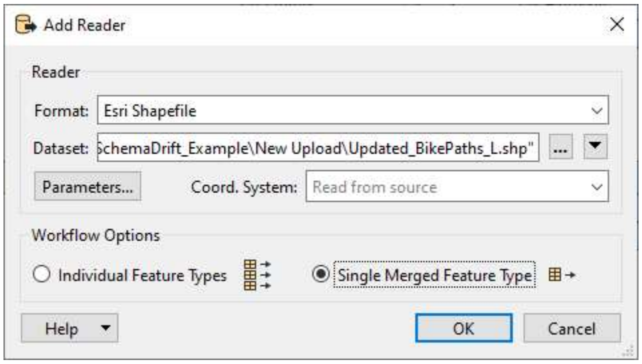
Workflow Options

● Individual Feature Types ○ Single Merged Feature Type

Help OK Cancel

3. Add Another Shapefile Reader

Now that we have the original dataset in our workspace, we want to add the updated data to the workspace to compare against it. Again, add a reader to the canvas and select Esri Shapefile as the Format. For the dataset, select Updated_BikePaths_L.shp. Because the dataset may change at any time, we want to make sure we're able to read in the data dynamically. To do so, change Workflow Options to Single Merged Feature Type. Press OK to add the reader to the canvas.



Add Reader

Reader

Format: Esri Shapefile

Dataset: SchemaDrift_Example\New Upload\Updated_BikePaths_L.shp"

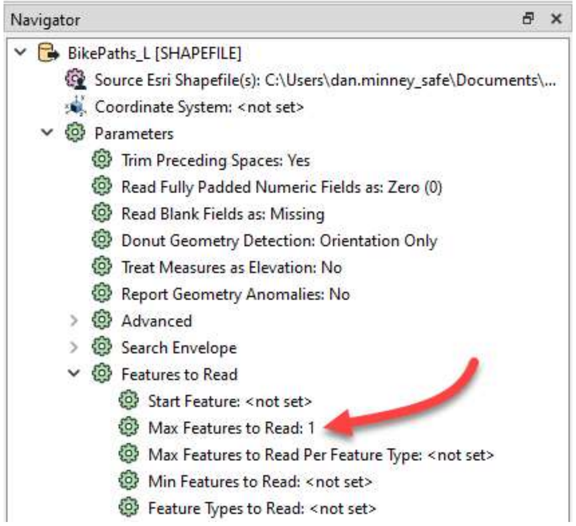
Parameters... Coord. System: Read from source

Workflow Options

○ Individual Feature Types ● Single Merged Feature Type

Help OK Cancel

Improving workspace performance tip: Since we only need to receive one feature from the reader to extract the dataset's schema, we can limit the reader to reading in only 1 feature. To do this, go to the Navigator Window, expand the reader's parameters, click on Parameters > Features to Read > and set Max Features to Read to 1. You can do this for both the shapefile readers.



Navigator

- BikePaths_L [SHAPEFILE]
  - Source Esri Shapefile(s): C:\Users\dan.minney_safe\Documents\...
  - Coordinate System: <not set>
  - Parameters
    - Trim Preceding Spaces: Yes
    - Read Fully Padded Numeric Fields as: Zero (0)
    - Read Blank Fields as: Missing
    - Donut Geometry Detection: Orientation Only
    - Treat Measures as Elevation: No
    - Report Geometry Anomalies: No
    - Advanced
    - Search Envelope
    - Features to Read
      - Start Feature: <not set>
      - Max Features to Read: 1
      - Max Features to Read Per Feature Type: <not set>
      - Min Features to Read: <not set>
      - Feature Types to Read: <not set>

4. Add a SchemaScanner Transformer

Next, we need to extract the schema from the source base dataset and prepare it for comparison against the newly uploaded dataset. To do this, we can add a SchemaScanner to our workspace and connect the BikePaths_L shapefile feature type to it.

5. Setting the SchemaScanner Parameters

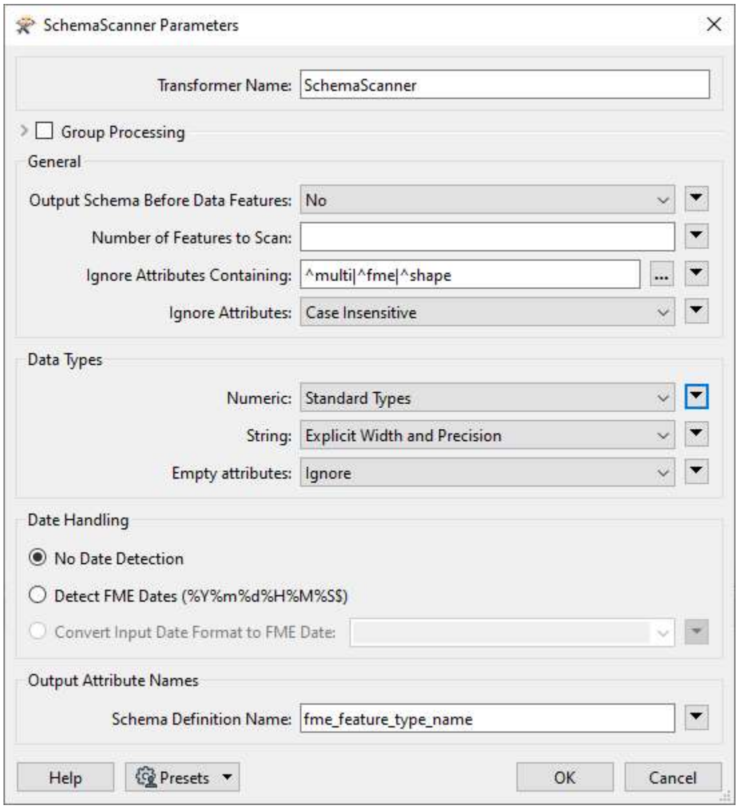
The SchemaScanner on its own won't work with our dataset, because it will produce schema features that we don't want. We want to keep only the original exposed attributes and remove any additional attributes within the schema related to FME or the format itself.

Open up the SchemaScanner parameters. For Ignore Attributes Containing, set the value to:

```
^multi|^fme|^shape
```

This will remove any attributes that contain "multi", "fme", or "shape" from being output as part of the scanned schema.

Press OK to accept the new parameters.



SchemaScanner Parameters

Transformer Name: SchemaScanner

☐ Group Processing

General

Output Schema Before Data Features: No

Number of Features to Scan:

Ignore Attributes Containing: ^multi|^fme|^shape

Ignore Attributes: Case Insensitive

Data Types

Numeric: Standard Types

String: Explicit Width and Precision

Empty attributes: Ignore

Date Handling

● No Date Detection

○ Detect FME Dates (%Y%m%d%H%M%S$)

○ Convert Input Date Format to FME Date:

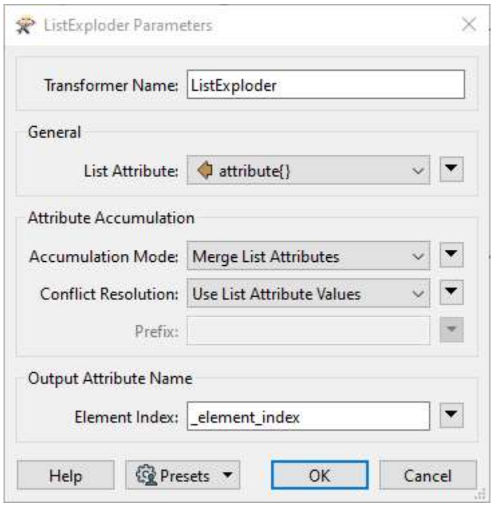
Output Attribute Names

Schema Definition Name: fme_feature_type_name

6. Add a ListExploder

At this point, if we were to run the workspace, the output schemas from the SchemaScanner would not be visible in the Visual Preview table as they remain as part of the attributes{} list; they would only be visible in the Feature Information window. Add a ListExploder (https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Transformers/Transformers/listexploder.htm) transformer to your workspace and connect the SchemaScanner <Schema> output port to it.

In the parameters, set the List Attribute to attribute{}.



ListExploder Parameters

Transformer Name: ListExploder

General

List Attribute: attribute{}

Attribute Accumulation

Accumulation Mode: Merge List Attributes

Conflict Resolution: Use List Attribute Values

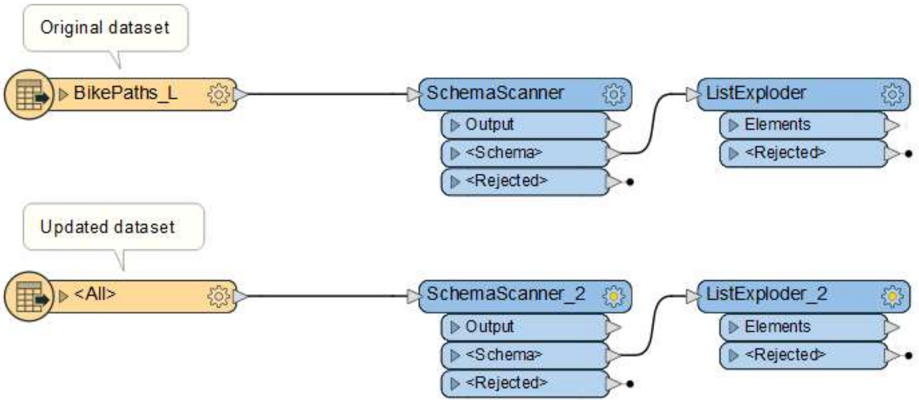
Prefix:

Output Attribute Name

Element Index: _element_index

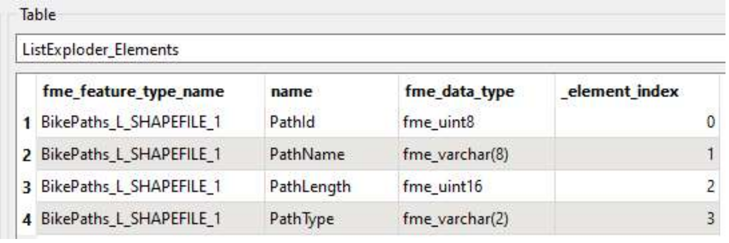
7. Duplicate the SchemaScanner & ListExploder

We also want to extract the schema from the Updated dataset for comparison against the original dataset. Hold the CTRL key and select both the SchemaScanner and the ListExploder. Then, duplicate (CTRL+D or right-click > duplicate) the transformers. Connect the Updated_BikePaths_L dataset to SchemaScanner_2. Ensure that the SchemaScanner_2 is connected to the ListExploder_2 transformer. Your workspace should now look like this:



8. Inspect Output

At this point, we can run the workspace with Feature Caching Enabled to see what our output schema looks like. What we're expecting to see is a list of the attribute names and their data type. Below you can see an example of the original dataset's ListExploder output.



Table

ListExploder_Elements

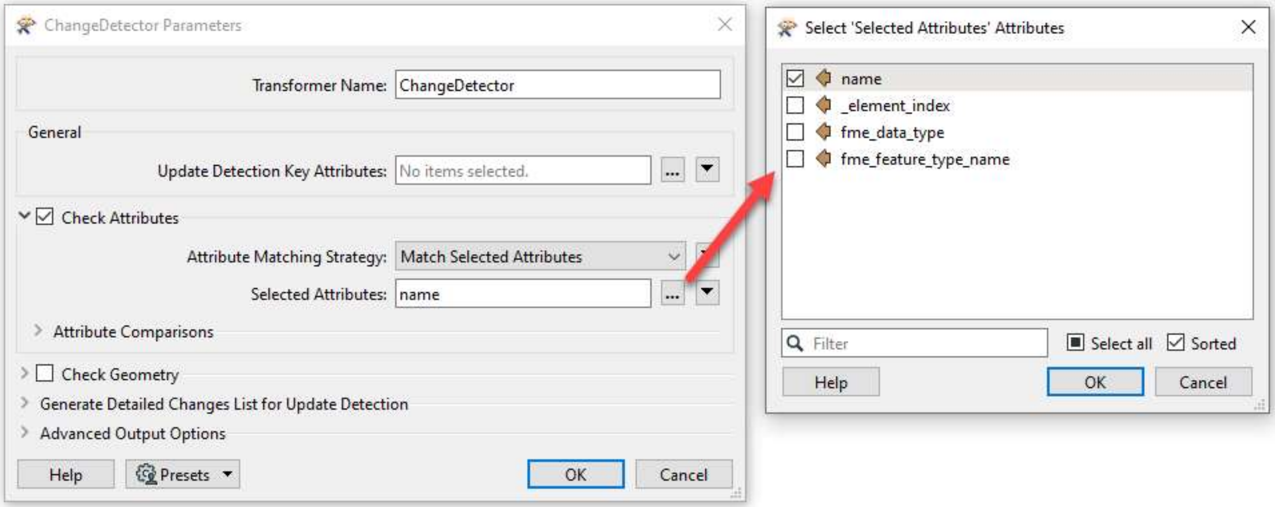
	fme_feature_type_name	name	fme_data_type	_element_index
1	BikePaths_L_SHAPEFILE_1	PathId	fme_uint8	0
2	BikePaths_L_SHAPEFILE_1	PathName	fme_varchar(8)	1
3	BikePaths_L_SHAPEFILE_1	PathLength	fme_uint16	2
4	BikePaths_L_SHAPEFILE_1	PathType	fme_varchar(2)	3

Hint: "fme_feature_type_name" denotes which dataset the schema belongs to. "name" is the name of the dataset's attributes. "fme_data_type" is the associated data type for each attribute.

9. Detecting Changes in the Schema

Now that we've been able to extract the schemas from our datasets into a set of attributes, we can now use the ChangeDetector (https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Transformers/Transformers/changedetector.htm) transformer to compare our dataset schemas against each other.

Add a ChangeDetector transformer to your workspace. Connect the ListExploder for the original dataset to the Original input port. Connect the ListExploder_2 for the updated dataset to the Revised input port. Open the ChangeDetector parameters. Under Check Attributes, click the ellipsis next to Selected Attributes and select name.



ChangeDetector Parameters

Transformer Name: ChangeDetector

General

Update Detection Key Attributes: No items selected.

☑ Check Attributes

Attribute Matching Strategy: Match Selected Attributes

Selected Attributes: name

Attribute Comparisons

☐ Check Geometry

Generate Detailed Changes List for Update Detection

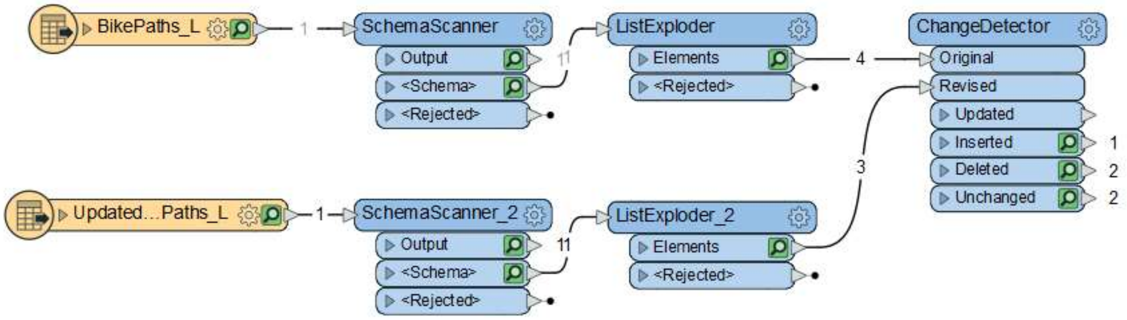
Advanced Output Options

Select 'Selected Attributes' Attributes

☑ name
☐ _element_index
☐ fme_data_type
☐ fme_feature_type_name

Press OK twice to accept the transformer parameters.

Now if we run the workspace, we can see that the schema has changed. One new attribute has been Inserted, two attributes have been Deleted, and two remain Unchanged.



Part 2: Creating a Schema Change Report

So far in our workflow, we've exposed the schemas of our datasets and compared them against each other to reveal which attributes have changed. In the next part of this exercise, we look at how to create a report on the changes in the schema.

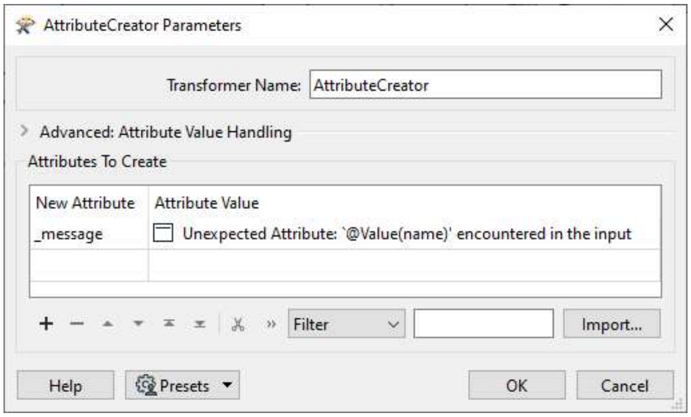
Perhaps we're looking towards a more manual method where the user is given an HTML report that tells them which attributes have changed and how the schema differs from the original dataset.

1. Add an AttributeCreator

We want to create a message for each attribute that has been deleted since the original schema. To do this, add an AttributeCreator (https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Transformers/Transformers/attributecreator.htm?Highlight=attributecreator) transformer to your workspace and connect the ChangeDetector Inserted output port to it.

Open the AttributeCreator parameters and create a New Attribute called _message. For Attribute Value, enter the following:

```
Unexpected Attribute: `@Value(name)' encountered in the input dataset.
```


AttributeCreator Parameters

Transformer Name: AttributeCreator

Advanced: Attribute Value Handling

Attributes To Create

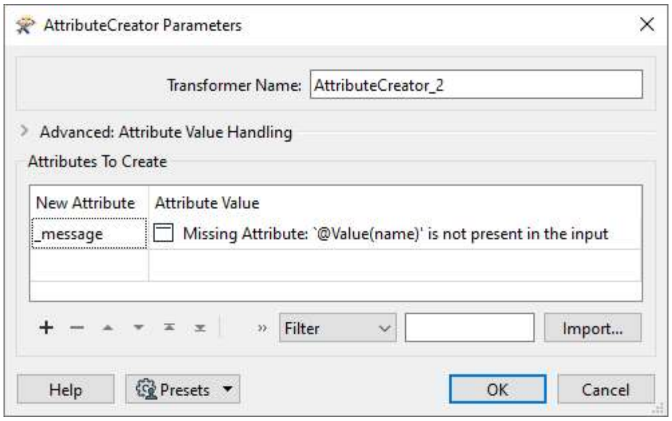
New Attribute	Attribute Value
_message	Unexpected Attribute: `@Value(name)' encountered in the input

Press OK to accept the parameters and close the transformer.

2. Add a Second AttributeCreator

We also want to create a message for each attribute that has been deleted since the original schema. To do so, we can add another AttributeCreator and connect the ChangeDetector's Deleted output port to it. Open the AttributeCreator_2 parameters and create a New Attribute called _message. For Attribute Value, enter the following:

```
Missing Attribute: `@Value(name)' is not present in the input dataset.
```



AttributeCreator Parameters

Transformer Name: AttributeCreator_2

Advanced: Attribute Value Handling

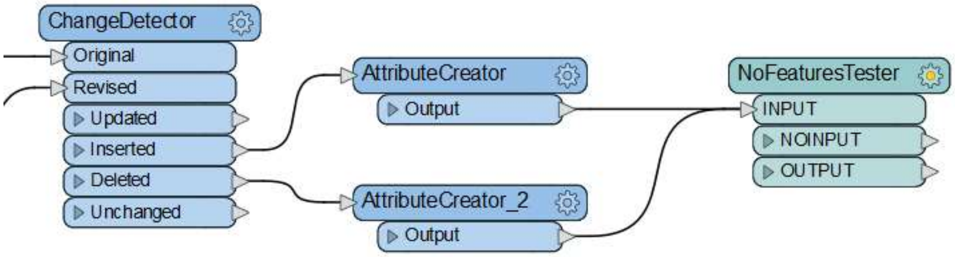
Attributes To Create

New Attribute	Attribute Value
_message	Missing Attribute: `@Value(name)' is not present in the input

3. Filtering for a Valid Schema

We need a way to test if there are any features coming from the ChangeDetector. If no features are output from the two AttributeCreators then that tells us the schema is valid and no error messages were produced.

Add a NoFeaturesTester (https://hub.safe.com/publishers/safe-lab/transformers/nofeaturestester) custom transformer to the workspace (FME will automatically download the transformer from the FME Hub). Connect both the AttributeCreators to it.



4. Create an HTML Report for Failed Schemas

Now that we've bundled our messages into a list that will be presentable in an HTML document, we can go ahead and create the report.

Add a HTMLReportGenerator (https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Transformers/Transformers/htmlreportgenerator.htm) to your workspace and connect both the AttributeCreators to it.

Open up the HTMLReportGenerator parameters. Under Page Contents, change the existing Chart(Bar) page to Header, by clicking on it and selecting Header. Click anywhere on the right side of the dialog to confirm the page change; the parameters on the right side will change to the Header parameters. Set the Header Content Settings as follows:

- **Text:** Schema Check Failed
- **Header Level:** H4
- **Text Alignment:** Left
- **Color:** 0,0,0

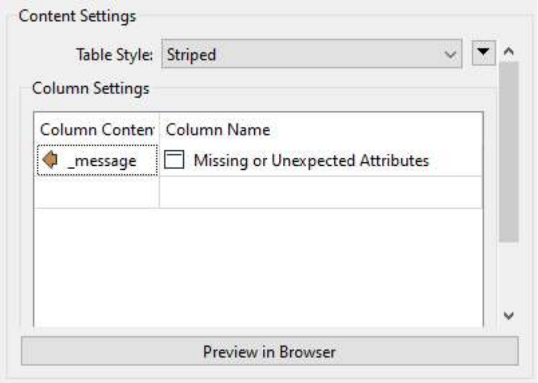
Now that we've set up a simple header, we can add our list of messages.

Add another page by clicking on the small plus (+) symbol in the bottom left, this time setting it to Custom HTML. Under Content Settings, set the Custom HTML as follows:

```
The upload was halted because the source schema has been changed! Please check the daily input and log for more details. <br>
Below you can find more details on missing or unexpected attributes.<br> <br>
```

Next, add another page. Set it as a Table and change the Table Style to Striped. Under Column Settings, set Column Contents to the _message attribute, and set the Column Name to:

```
Missing or Unexpected Attributes
```



Content Settings

Table Style: Striped

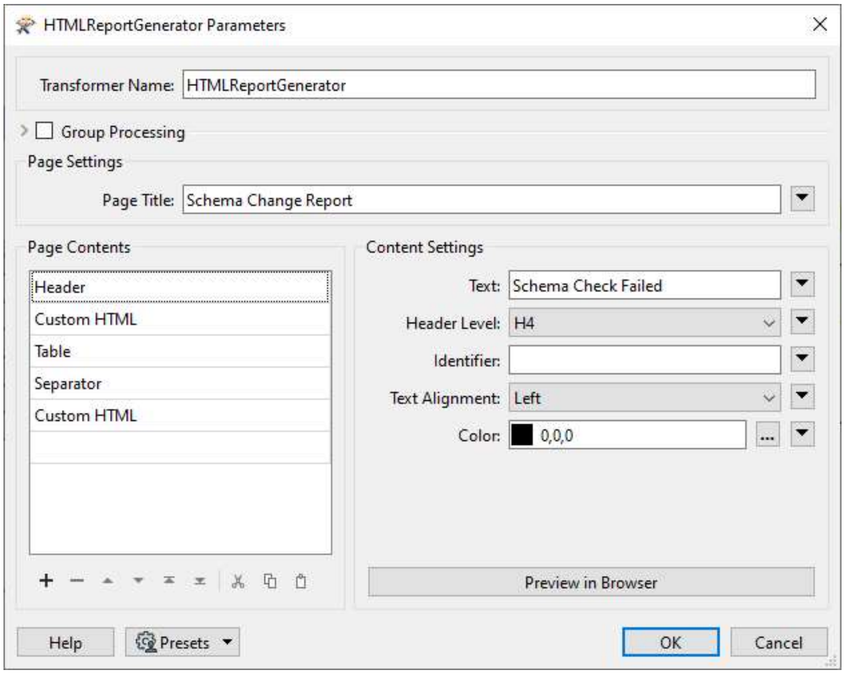
Column Settings

Column Conten	Column Name
_message	□ Missing or Unexpected Attributes

Below this, add another page and set it to a Separator. Set the thickness to 2.

Add one last page and set it to Custom HTML: Set the Custom HTML as follows:

```
<br>Thanks for your cooperation.
```



HTMLReportGenerator Parameters

Transformer Name: HTMLReportGenerator

□ Group Processing

Page Settings

Page Title: Schema Change Report

Page Contents

- Header
- Custom HTML
- Table
- Separator
- Custom HTML

Content Settings

Text: Schema Check Failed

Header Level: H4

Identifier:

Text Alignment: Left

Color: 0,0,0

Press OK to accept the parameters and close the transformer.

5. Cleaning up the HTML Layout

Before sending out the HTML to the HTML writer, we want to clean up and organize the HTML layout. Add an HTMLLayouter (https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Transformers/Transformers/htmllayouter.htm?Highlight=htmllayouter) to the canvas and connect the HTMLReportGenerator to it. We can use the default parameters. The HTMLLayouter is commonly paired with the HTMLReportGenerator to create a bootstrap grid or vertical layouts. In this case, the default parameters will provide us with a vertical layout.

6. Create an HTML Report for Valid Schemas

We also want our users to be notified if their schema is valid, not just if their schema has failed. The message needs to go both ways to make the user understand what actions they need to take next. Add another HTMLReportGenerator to the canvas and connect the NoFeaturesTester NOINPUT output port to it.

Open up the HTMLReportGenerator_2 parameters. Under Page Contents, change the existing Chart(Bar) page to Header, by clicking on it and selecting Header. Click anywhere on the right side of the dialog to confirm the page change; the parameters on the right side will change to the Header parameters. Set the Header Content Settings as follows:

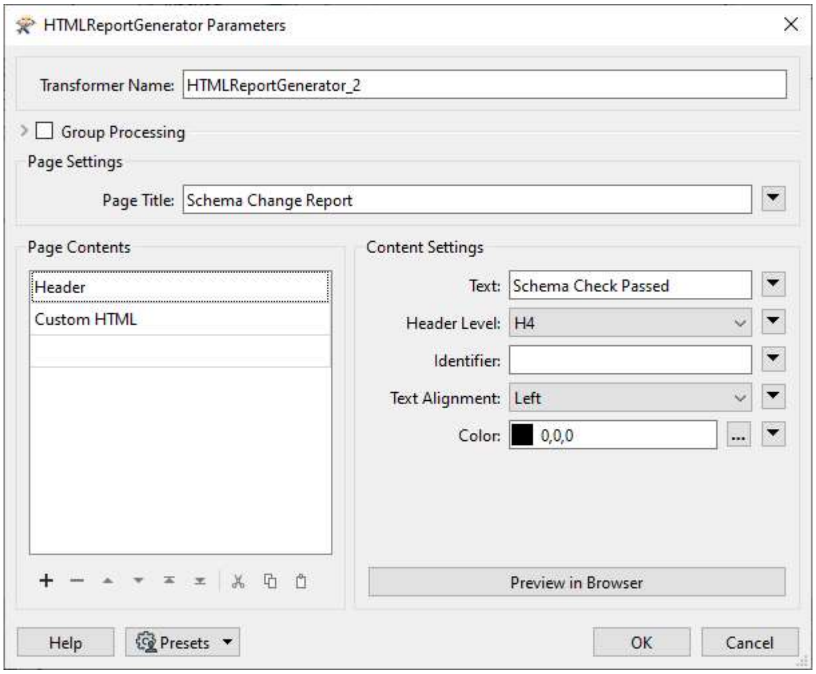
- **Text:** Schema Check Passed
- **Header Level:** H4
- **Text Alignment:** Left
- **Color:** 0,0,0

Add another page by clicking on the small plus (+) symbol in the bottom left, this time setting it to Custom HTML. Under Content Settings, set the Custom HTML as follows:

```
Hello, <br>

Your data has passed the schema validation test and is ready to be uploaded. <br>

Thanks for your cooperation.
```



HTMLReportGenerator Parameters

Transformer Name: HTMLReportGenerator_2

□ Group Processing

Page Settings

Page Title: Schema Change Report

Page Contents

- Header
- Custom HTML

Content Settings

Text: Schema Check Passed

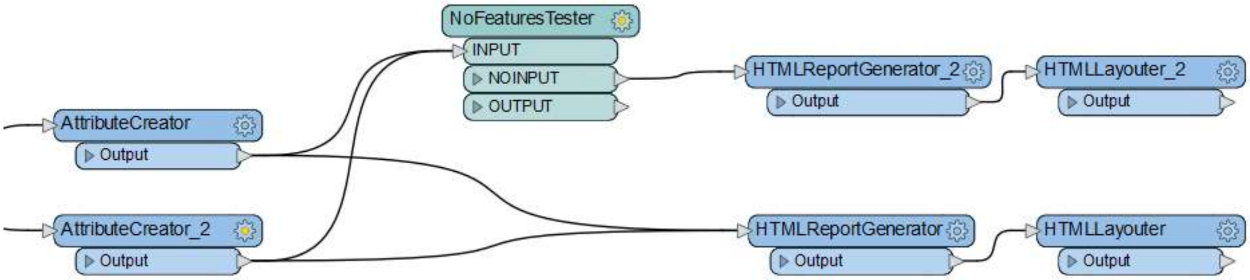
Header Level: H4

Identifier:

Text Alignment: Left

Color: 0,0,0

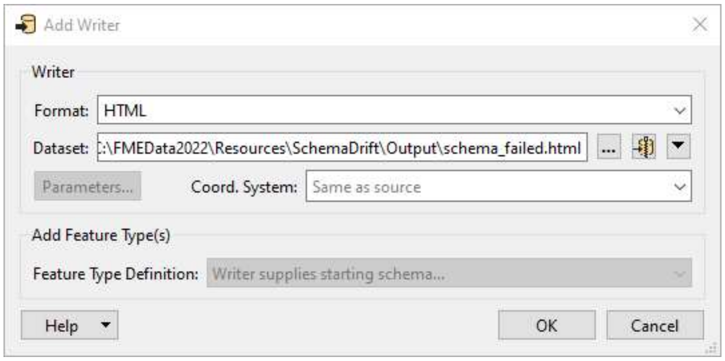
Press OK to close the HTMLReportGenerator_2 and accept the new parameters. Add another HTMLLayouter to the canvas and connect the HTMLReportGenerator_2 to it. Again, we can use the default parameters to create a vertical bootstrap layout.



7. Output the HTML Report Back to the User

Whether you plan to use this workspace in a workspace app or just create a simple HTML document, we need to add a writer to output our final schema report!

Add a writer to your workspace and set the Format to HTML. Browse to an output file location and then name the file schema_failed.html. Press OK to add the writer to your workspace. Now, let's connect the HTMLLayouter to the HTML writer feature type.



Add Writer

Writer

Format: HTML

Dataset: :\FMEData2022\Resources\SchemaDrift\Output\schema_failed.html

Coord. System: Same as source

Add Feature Type(s)

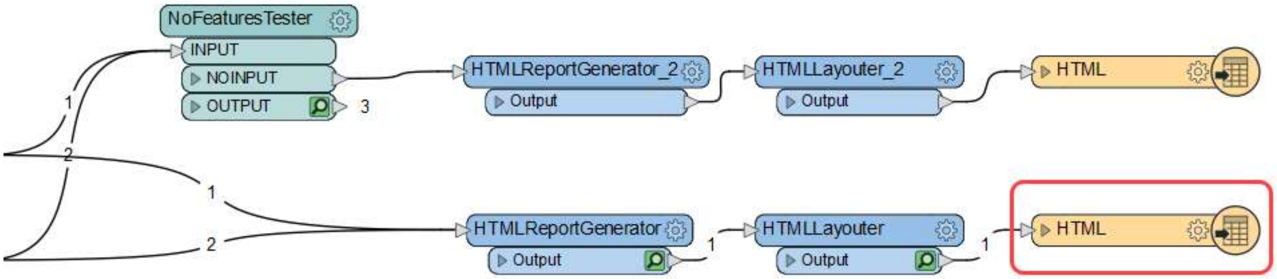
Feature Type Definition: Writer supplies starting schema...

We can repeat the process above to create an HTML writer for the valid schema report. Add another HTML writer to your workspace. Set the Dataset parameter location to the same output location as above, then name the file schema_passed.html.

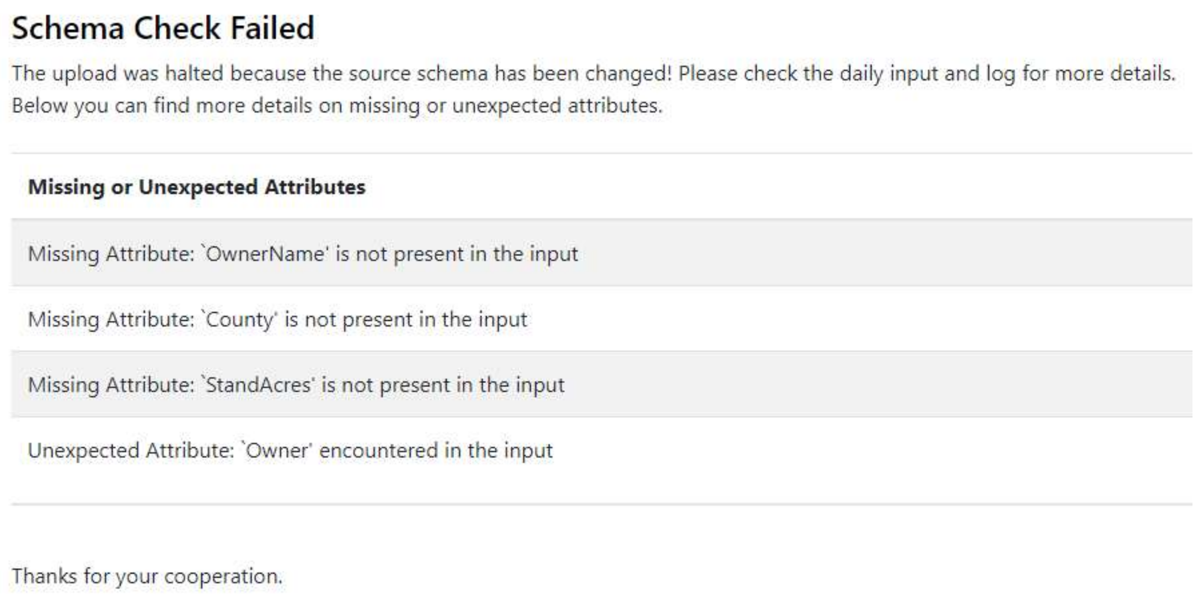
At this point, everything in our workspace is complete and we can output our data to produce the report. Connect the schema_passed writer feature type to the HTMLLayouter_2.

8. Run the Workspace and Inspect the Results

Now that we've set up our workspace to both detect changes in the schema, and produce a report on the condition of the schema, we can try running the workspace to see our results! Try running the workspace, click Open Containing Folder on the HTML Writer that receives features from its HTMLLayouter.



Open up the resulting HTML file in your web browser by double-clicking on it. You should see a Schema Check Failed report.



Schema Check Failed

The upload was halted because the source schema has been changed! Please check the daily input and log for more details. Below you can find more details on missing or unexpected attributes.

Missing or Unexpected Attributes
Missing Attribute: `OwnerName' is not present in the input
Missing Attribute: `County' is not present in the input
Missing Attribute: `StandAcres' is not present in the input
Unexpected Attribute: `Owner' encountered in the input

Thanks for your cooperation.

Conclusion

Throughout this exercise, we learned how to utilize the SchemaScanner alongside the ChangeDetector to perform a change detection process on two dataset schemas. The result? We're now able to create a report on whether or not the schemas match. With this new information, users will feel more confident when updating datasets and inserting new records. This also means time saved by not having to go back and fix the dataset after a user has uploaded features with a mismatched schema!

How else can we use this workspace? Instead of writing out our data to an HTML file, we could email a user the results of the Schema Change Report in the form of the HTML content produced through the HTMLReportGenerator and the HTMLLayouter. Alternatively, if you're familiar with FME Server Apps, you could create an app that allows the user to upload a file to the server app themselves and instantly receive a copy of the Schema Change Report in their browser upon submission of the file.

First Published Date
7/11/2022, 8:31 PM

Last Published Date
8/4/2022, 3:09 PM

Authoring (/s/topic/0TO4Q000000...   FME Desktop (/s/topic/0TO4Q000000...

Sort by:
Latest Posts

LizAtSafe (/s/profile/0050c00000CeGV0AAN) (Employee) published a new version of this Knowledge.
August 4, 2022 at 3:09 PM (/s/feed/0D54Q00009hQx0FSAS)

Like   Comment

Log In to Comment

danminneyatsafe (/s/profile/0054Q00000F9IqdQAF) (Employee) published this new Knowledge.
July 11, 2022 at 8:31 PM (/s/feed/0D54Q00009e0OytSAE)

Like Comment

Log In to Comment

Follow

Files (1) (/s/relatedlist/ka14Q000001DWyrQAG/AttachedContentDocuments)

ZIP 5-DealingWithSchemaDrift
Aug 4, 2022 • 171KB • zip

View All
(/s/relatedlist/ka14Q000001DWyrQAG/AttachedContentDocuments)

Related Articles

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

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

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

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

Dynamic Workflows: Destination Schema is Derived from an External Dataset (/s/article/dynamic-workflow-tutorial-destination-schema-is-de)

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.