Before we begin…

Training Home | fme.ly/uctraining

Exercises | fme.ly/MergeDataTraining

C:\FMEData\Resources\FMEUC22\Merging and Joining Tabular Data

Workspaces

Start Exercises with
***Begin.fmw**

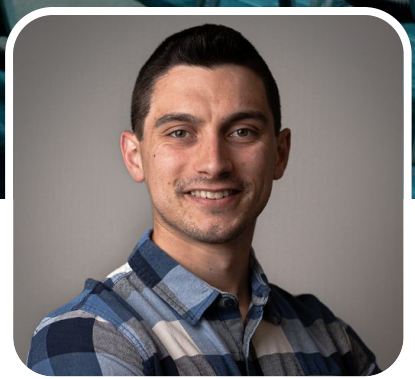
Source Data

JSON
PostgreSQL
CSV
Excel
SQLite
City of Surrey


THE PEAK OF DATA INTEGRATION
2022 UC
Merging & Joining Tabular Data
SAFE SOFTWARE

Presenters

Nampreet Singh
Support Specialist



Christian Berger
Support Specialist

Daragh Broderick
Support Specialist

Merging & Joining?

INTEGRATE

DISPARATE

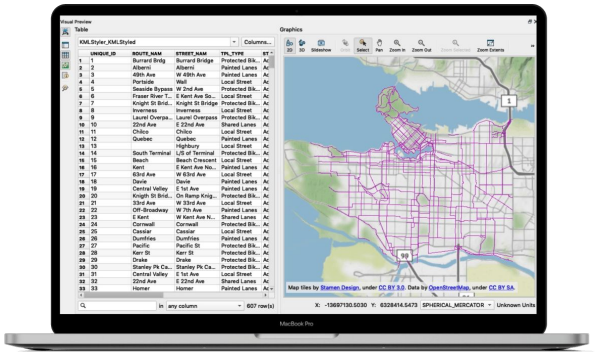
DATASETS

Data silos

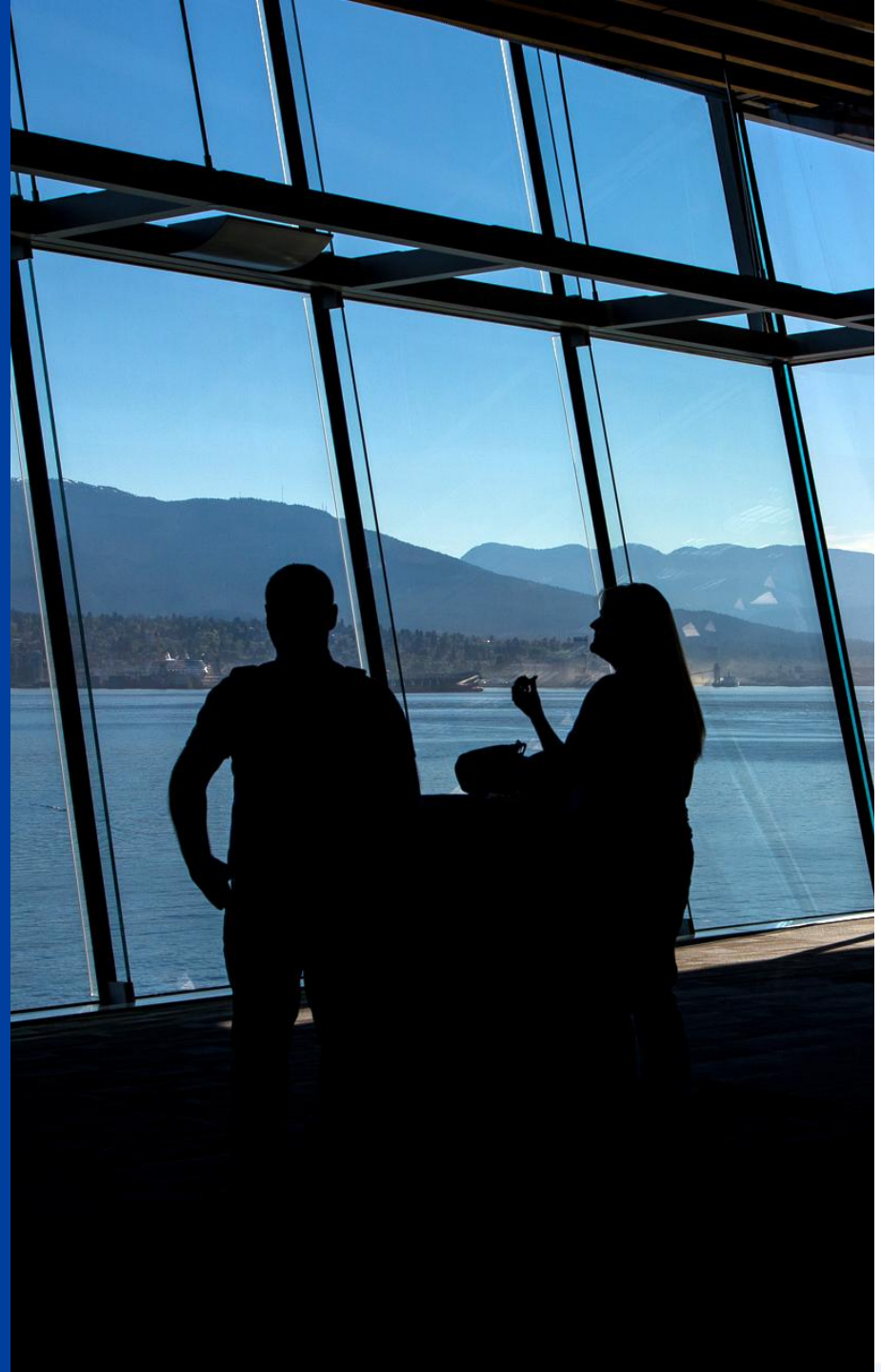
Just because you have disconnected and different kinds of data, doesn't mean they can't be brought together.

Merging & Joining with FME

The solution
(often includes FME)



Enhanced Data!



Agenda

- The Basics of Merging & Joining
- Transformer Categories
- 6 Exercises (FME Desktop)
  + 1 Optional (HTML Report & FME Server)
- Transformer Comparisons

The Basics of Merging & Joining

Append/Union

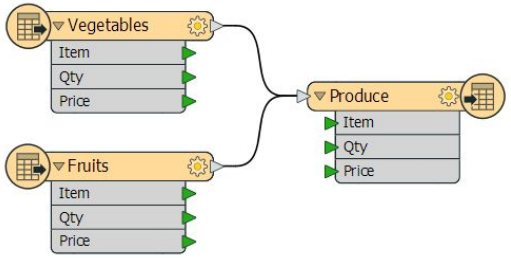
Item	Qty	Price
Tomatoes	354	$1.99
Corn	445	$0.31
Mushrooms	622	$1.98

Item	Qty	Price
Pineapples	1219	$0.84
Mangoes	8862	$3.87
Peaches	2154	$1.45

Append

Item	Qty	Price
Tomatoes	354	$1.99
Onions	445	$0.31
Mushrooms	622	$1.98
Pineapples	1219	$0.84
Mangoes	8862	$3.87
Peaches	2796	$1.17

Easy Peasy in FME!



Merge/Join

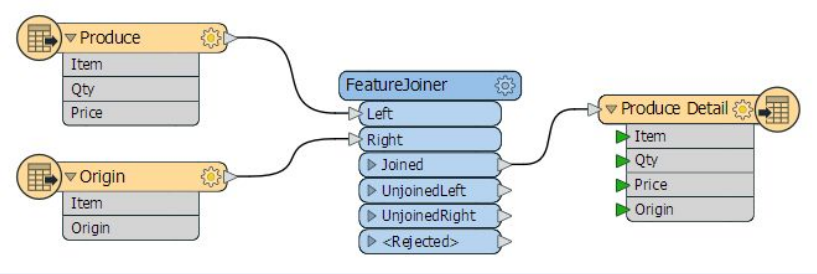
Item	Qty	Price
Tomatoes	354	$1.99
Mangoes	8862	$3.87
Peaches	2796	$1.17

Item	Origin
Tomatoes	California
Mangoes	Philippines
Peaches	British Columbia

Merge

Item	Qty	Price	Origin
Tomatoes	354	$1.99	California
Mangoes	8862	$3.87	Philippines
Peaches	2796	$1.17	British Columbia

Today's Focus… also Easy Peasy!



There are many paths you can take in FME.

Transformer Categories

SQL-Free Transformers

FeatureJoiner
FeatureMerger
DatabaseJoiner

Can be set up easily without any database knowledge

SQL-Based Transformers

InlineQuerier
SQLCreator
SQLExecutor

Requires Knowledge of SQL.

Before we begin…

Training Home | fme.ly/uctraining

Exercises | fme.ly/MergeDataTraining

C:\FMEData\Resources\FMEUC22\Merging and Joining Tabular Data

Workspaces

Start Exercises with
***Begin.fmw**

Source Data

JSON
PostgreSQL
CSV
Excel
SQLite
City of Surrey

Today's Exercises

Exercises
fme.ly/MergeDataTraining

C:\FMEData\Resources\FMEUC22\
Merging and Joining Tabular Data

Original Water Meter Dataset | WaterMeters-SURREY.json

Table

water_meters

	FACILITYID	ACCOUNT_NO	FOLIO	STATUS	GPS	IMAGE	LOTLINK	METER_CODE	PID	json_ogc_wkt_crs	json_geometry.type
1	1000947091	443723	2340-01027-5	In Service	Y	http://cosmos.surre...	69713	12	018363318	PROJCS["NAD83 / U...	Point
2	1001945997	501989	2340-98108-6	In Service	N		<null>	10	001637070	PROJCS["NAD83 / U...	Point
3	1001182234	469871	2340-04023-1	In Service	Y	http://cosmos.surre...	118164	12	028162960	PROJCS["NAD83 / U...	Point
4	1001166028	466959	2270-00034-1	In Service	Y	http://cosmos.surre...	36487	13	010311041	PROJCS["NAD83 / U...	Point
5	1000920360	315897	2270-01034-6	In Service	Y	http://cosmos.surre...	36488	13	001825771	PROJCS["NAD83 / U...	Point
6	1000949186	311836	2270-04016-8	In Service	Y	http://cosmos.surre...	36518	14	009755055	PROJCS["NAD83 / U...	Point

Create an enhanced Water Meter Dataset | Excel Workbook

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	FACILITYID	FOLIO	IMAGE	METER CODE	LEGAL FREEFORM	OWNER CLASS	ZONING	HOUSE	STREET	POSTAL_CODE	Label	Address	MSLINK	PLAN_NUMBER	PLAN_TYPE	PLAN_YEAR
2	1000947091	2340-01027-5	http://cosmos.surrey.ca	12	LOT 2 SECTION 34 RANGE 2 PLAN LMP11416	PRIVATE	Single Family Residential Zone	9688	132 St	V3T 3S4	ID: 1000947091 Meter Code: 12	9688 132 St, Surrey, BC, V3T 3S4	69713	LMP11416	Subdivision	1993
3	1001945997	2340-98107-4,2340-9810		10	LOT 1 SECTION 34 RANGE 2 PLAN NWS1050	PRIVATE	Duplex Residential Zone	13235	97 Ave	V3T 1A4	ID: 1001945997 Meter Code: 10	13235 97 Ave, Surrey, BC, V3T 1A4	35983	NWS1050	Strata	1978
4	1001182234	2340-04023-1	http://cosmos.surrey.ca	12	LOT 5 BLOCK 5N SECTION 34 RANGE 2W PLA	PRIVATE	Semi Detached Residential Zone	9904	132 St	V3T 3S8	ID: 1001182234 Meter Code: 12	9904 132 St, Surrey, BC, V3T 3S8	118164	BCP43901	Subdivision	2010
5	1001166028	2270-00034-1	http://cosmos.surrey.ca	13	LOT 1 BLOCK 5N SECTION 27 RANGE 2W PLA	PRIVATE	Single Family Residential Zone	10218	132 St	V3T 3T7	ID: 1001166028 Meter Code: 13	10218 132 St, Surrey, BC, V3T 3T7	36487	NWP17054	Subdivision	1956
6	1000920360	2270-02022-4	http://cosmos.surrey.ca	13	LOT 3 SECTION 27 RANGE 2 PLAN NWP17054	PRIVATE	Single Family Residential Zone	10198	132 St	V3T 3T7	ID: 1000920360 Meter Code: 13	10198 132 St, Surrey, BC, V3T 3T7	36489	NWP17054	Subdivision	1956
7	1000949186	2270-04016-8	http://cosmos.surrey.ca	14	LOT 5 SECTION 27 RANGE 2 PLAN NWP12933	PRIVATE	Single Family Residential Zone	10304	132 St	V3T 3T9	ID: 1000949186 Meter Code: 14	10304 132 St, Surrey, BC, V3T 3T9	36518	NWP12933	Subdivision	1952
8	1001212640	2220-76902-1	http://cosmos.surrey.ca	14	LOT 120 BLOCK 5N SECTION 22 RANGE 2W PL	PRIVATE	Single Family Residential Zone	10690	132 St	V3T 3W1	ID: 1001212640 Meter Code: 14	10690 132 St, Surrey, BC, V3T 3W1	41043	NWP12661	Subdivision	1952
9	1001158231	2220-76102-2	http://cosmos.surrey.ca	11	LOT 112 SECTION 22 RANGE 2 PLAN NWP126	PRIVATE	Single Family Residential Zone	10728	132 St	V3T 3W3	ID: 1001158231 Meter Code: 11	10728 132 St, Surrey, BC, V3T 3W3	41035	NWP12661	Subdivision	1952

V3T.Water Meters | V3V.Water Meters | V3R.Water Meters | Missing Addresses

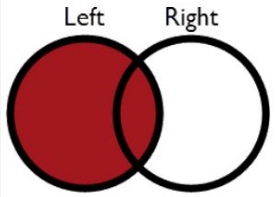
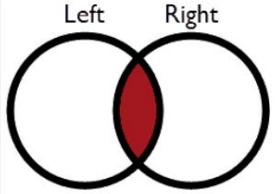
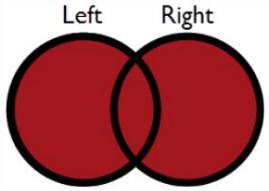
Course Content

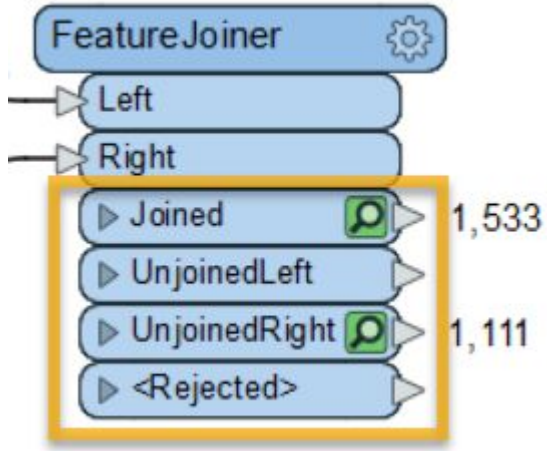
fme.ly/MergeDataTraining

C:\FMEData\Resources\FMEUC22\
Merging and Joining Tabular Data

Exercise 1: **FeatureJoiner** | *Add Zone Codes*

FeatureJoiner Review

Mode	Venn Diagram	Joined	UnjoinedLeft	UnjoinedRight
Left		All Matches and Unmatched Left Features	Nothing	Unmatched Right Features
Inner		All Matches **ONLY**	Unmatched Left Features	Unmatched Right Features
Full		Everything	Nothing	Nothing



Documentation | FeatureJoiner
Tutorial | The FeatureJoiner Transformer

FeatureJoiner | Cardinality

Cardinality	Description	Output (assuming 1 key value)
1:1	**One to One:** If each Left feature has a single match among the Right features (for example a single point feature is mapped to an address table via a unique address ID key), this is a 1:1 match and produces a single Joined feature.	**1** Left matches **1** Right: **1** Joined Feature output
1:M	**One to Many:** If each Left feature has multiple matches among the Right features (for example a single address record is mapped to a list of planning applications for that address), this is a 1:M (one-to-many) match and produces a Joined feature for every match that occurs.	**1** Left matches **10** Right: **10** Joined Features output
M:1	**Many to One:** If multiple Left features match a single Right feature record (for example a number of addresses match to the same census data via a postal code field) this is a M:1 (many-to-one) match and produces a Joined feature for every match that occurs	**10** Left match **1** Right: **10** Joined Features output
M:N	**Many to Many:** If multiple Left features match multiple Right features (for example a number of addresses match to a number of records for electrical power outages) this is a M:N (many-to-many) match and produces a Joined feature for every match that occurs.	**10** Left match **10** Right: **100** Joined Features output* *When all features have identical key values - all Left match all Right.

Documentation | FeatureJoiner

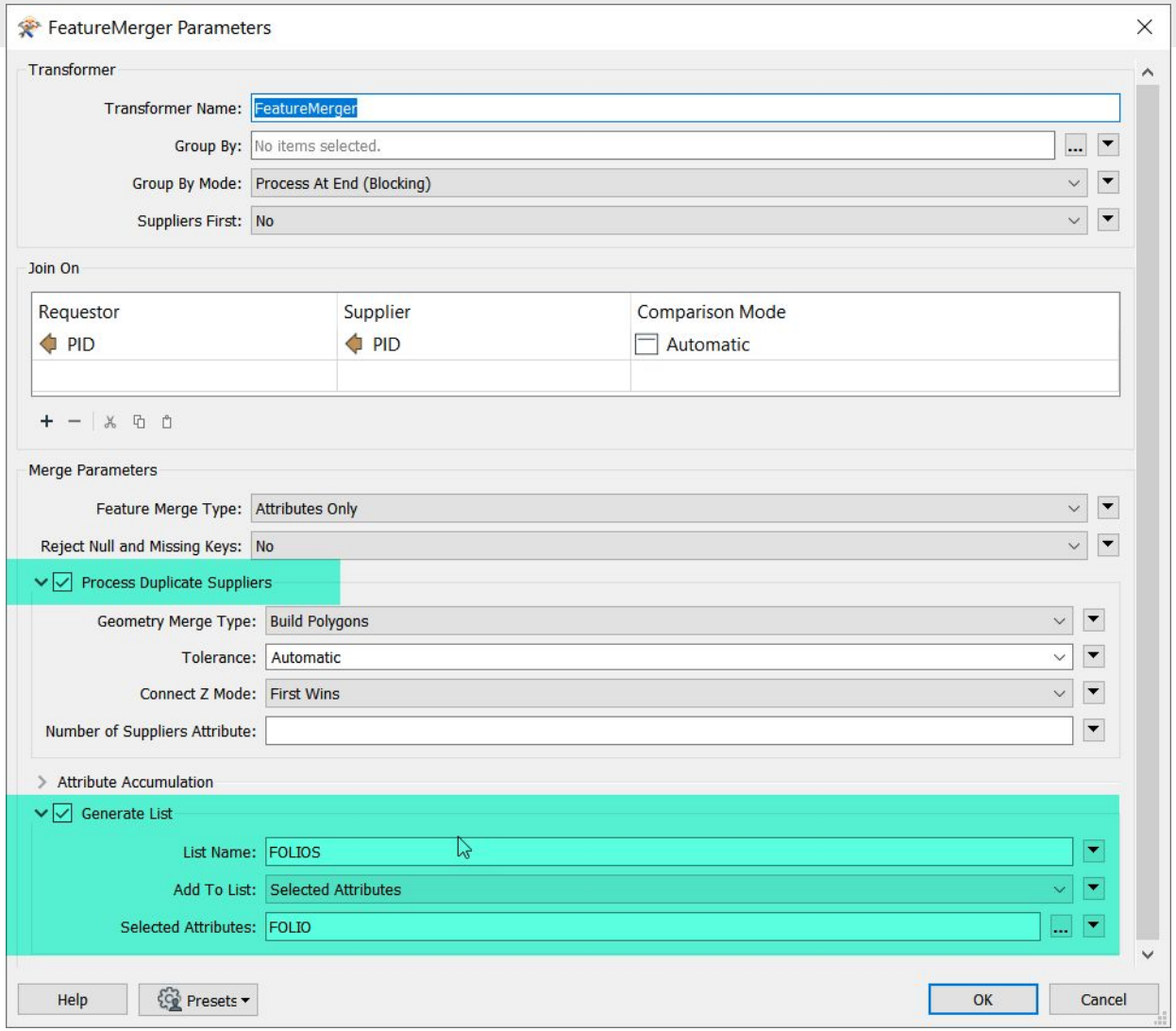
Exercise 2: **FeatureMerger** | *Add FOLIO Numbers* | *1:M Join*

FeatureMerger Summary

1:M join results in a single match by default

- Generate a list of matches for each Requestor by enabling:
  - **Process Duplicate Suppliers**
  - **Generate List**
    Tutorial: Working with List Attributes

Documentation | FeatureMerger
Tutorial | The FeatureMerger Transformer



FeatureJoiner

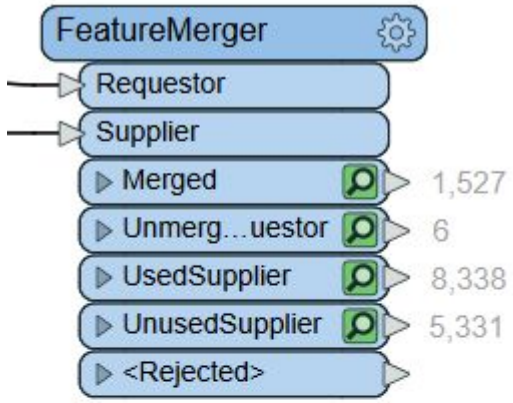
- **Simpler**, and **performs faster** than FeatureMerger
- SQL-like terminology
- Handles cardinalities easier than the FeatureMerger (ie. 1:M, M:N and M:1)
- Supports multiple matches per source (by default, 1:M results in a record for each match)
- Join Modes are more intuitive.

FeatureMerger

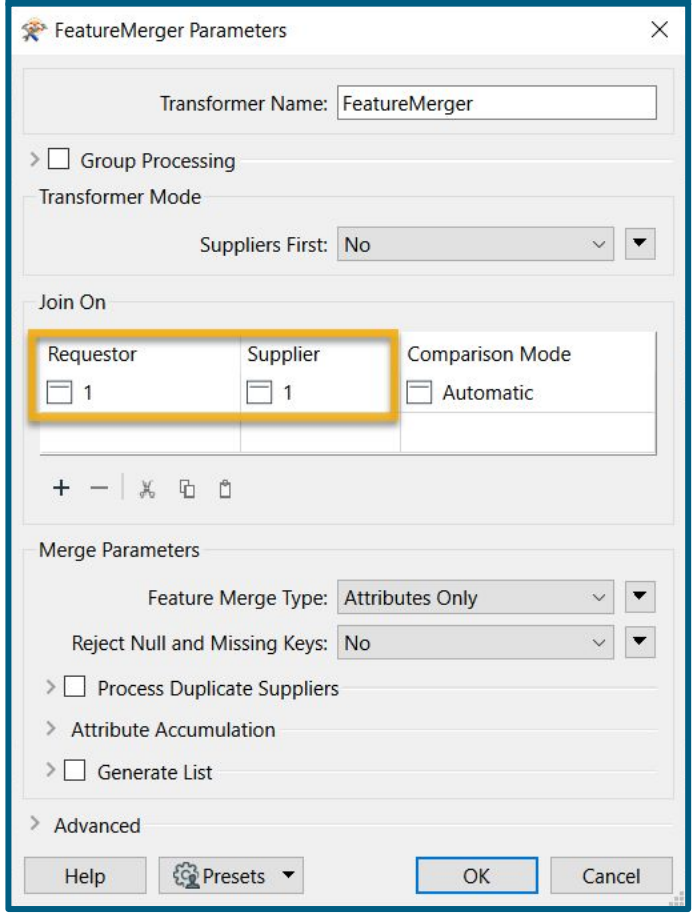
- Handling 1:M joins is a bit more work
- Default: Single match is output for 1:M relationship
- Generate lists of matches
- Many Suppliers can be merged on to a single Requestor.
- Drag & Drop Join Modes (connecting multiple ports)
- Can create geometries!

Extra Tips!

1:Many Joins
FeatureJoiner vs. FeatureMerger



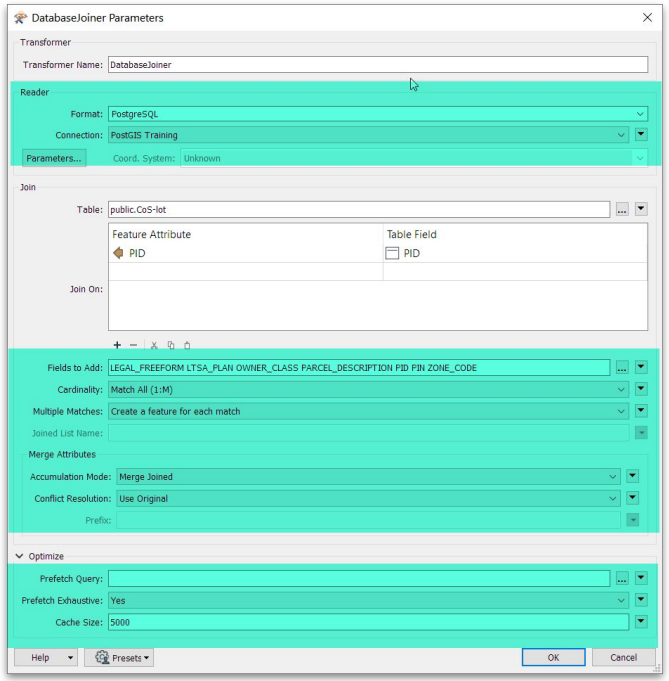
Join without
common attribute



Exercise 3: **DatabaseJoiner** | *Add Zone Descriptions*

Midstream Join with DatabaseJoiner

Provides the ability to form a join against a database (incl. CSV & Excel) with an existing dataset in the workspace.



Read in database formats, Excel or CSV

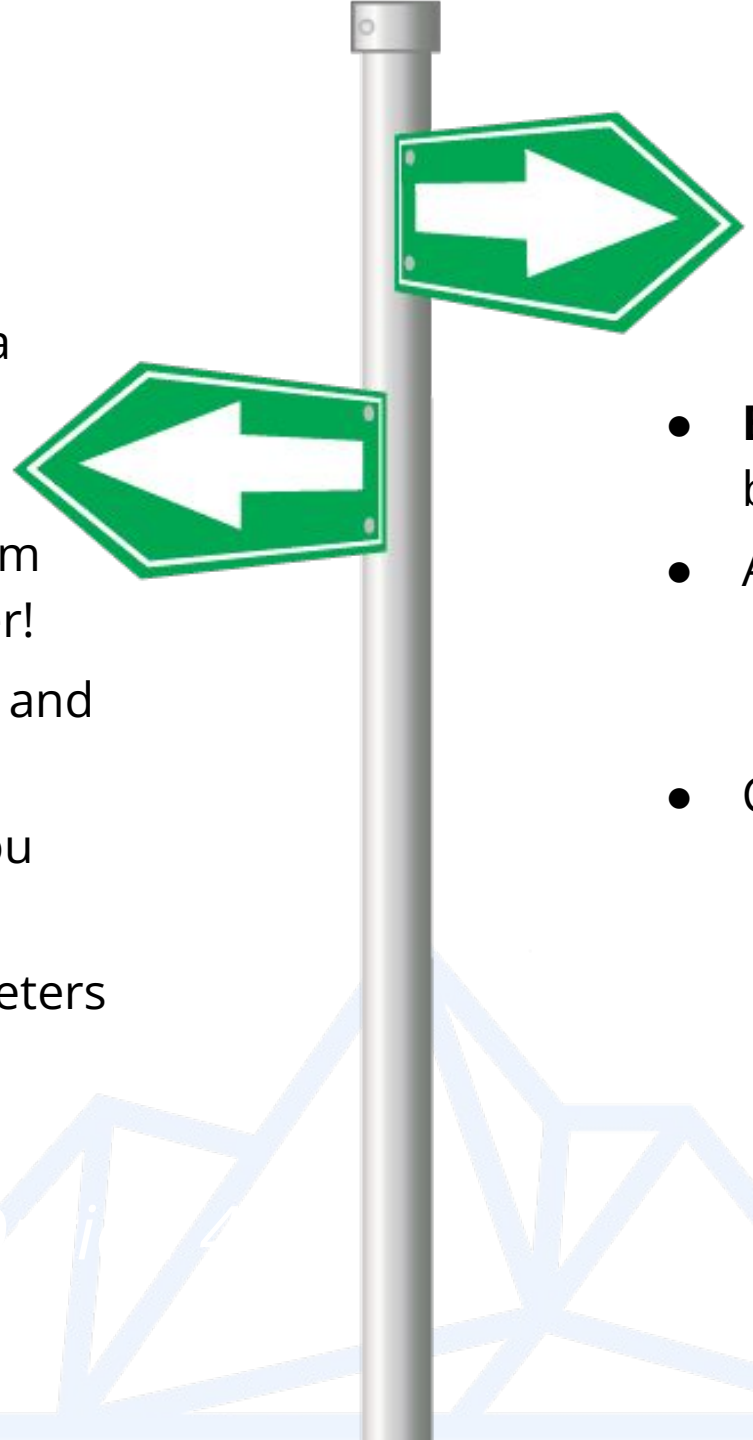
Matching and accumulation options

Let the database do the work by pre-filtering a subset of the table you're reading into the workspace

Documentation | DatabaseJoiner
Tutorial | The DatabaseJoiner Transformer

DatabaseJoiner

- **Non-Blocking** - Joins data as each row flows by
- Connect to and join to database tables midstream with one easy transformer!
- Can also connect to Excel and CSV
- Select which attributes you want to join
- Explicit cardinality parameters
- Good for quick lookups - caches first 5000 records

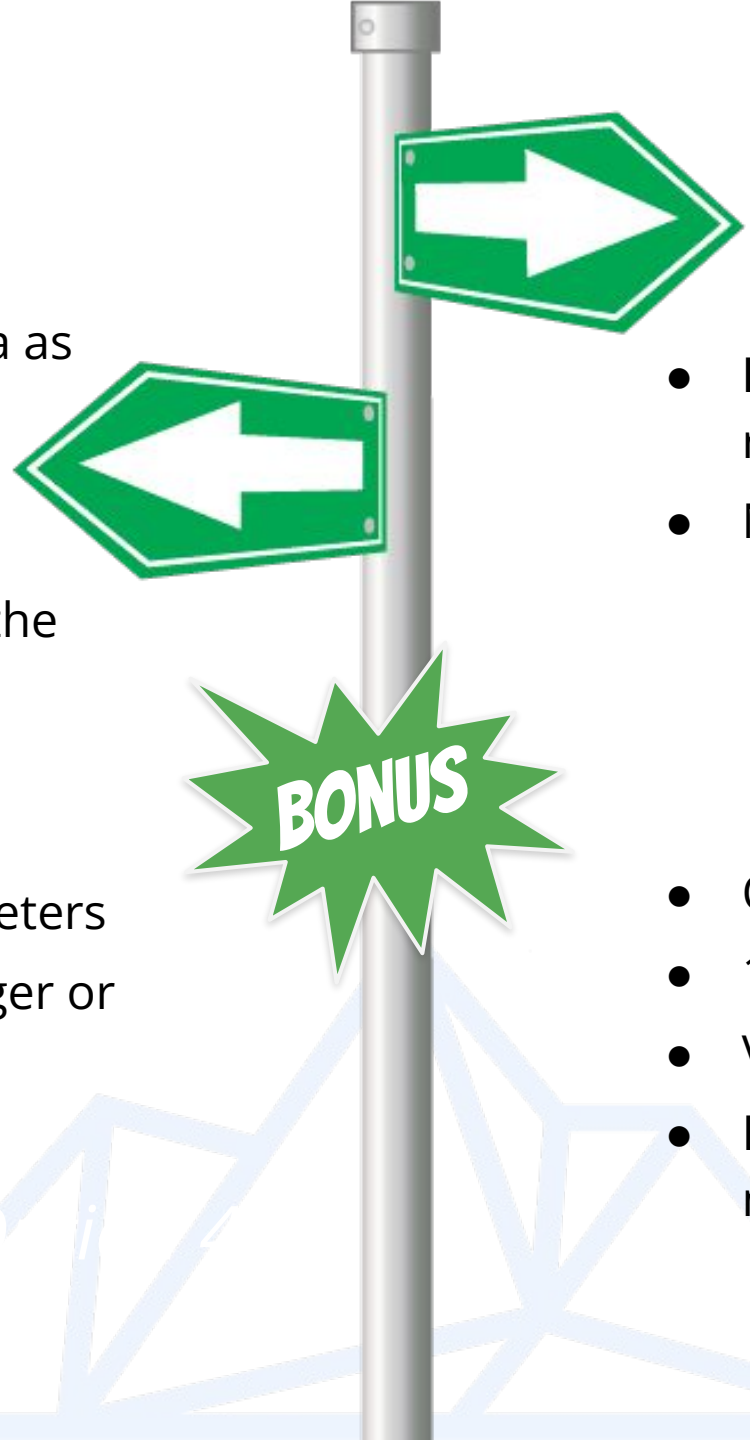


FeatureMerger/ FeatureJoiner

- **Blocking** - Wait until all data has been read before doing processing
- All FME readers can be used
  - No need for anything to be in a database
- Cardinality options vary

DatabaseJoiner

- **Non-Blocking** - Joins data as each row flows by
- Data is external to the workspace and can be changed without editing the workspace
- Can merge in multiple attributes
- Explicit cardinality parameters
- Best if the join data is larger or changing



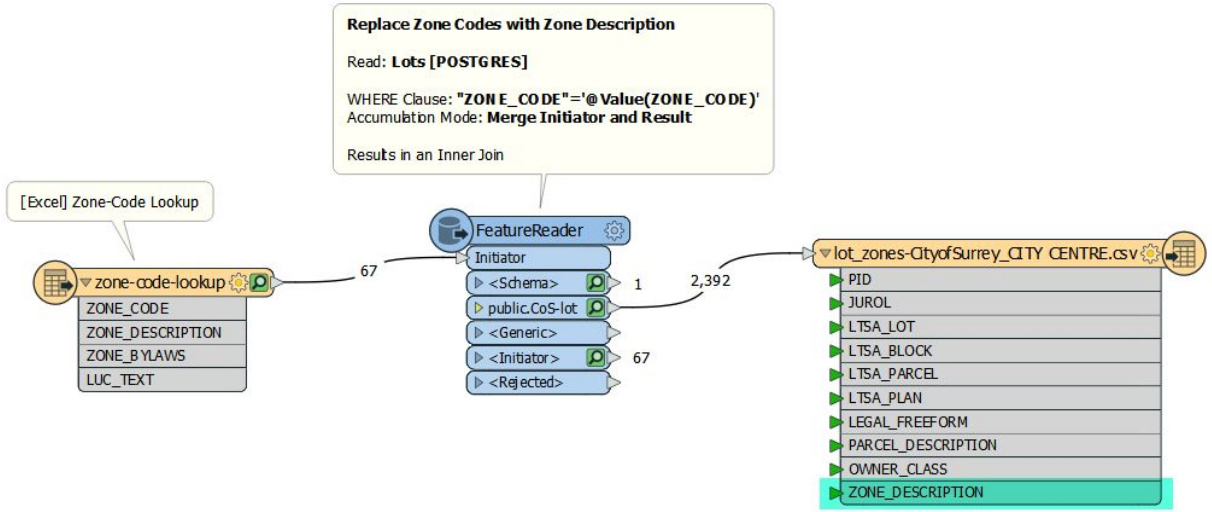
AttributeValueMapper

- **Non-Blocking** - Joins data as each row flows by
- No external dependencies
  - Data to lookup is stored in the workspace
  - Can be imported at design time from an external source
- Only a single attribute can be added
- 1:1 lookups only
- Very fast
- Best for small, simple, stable mappings

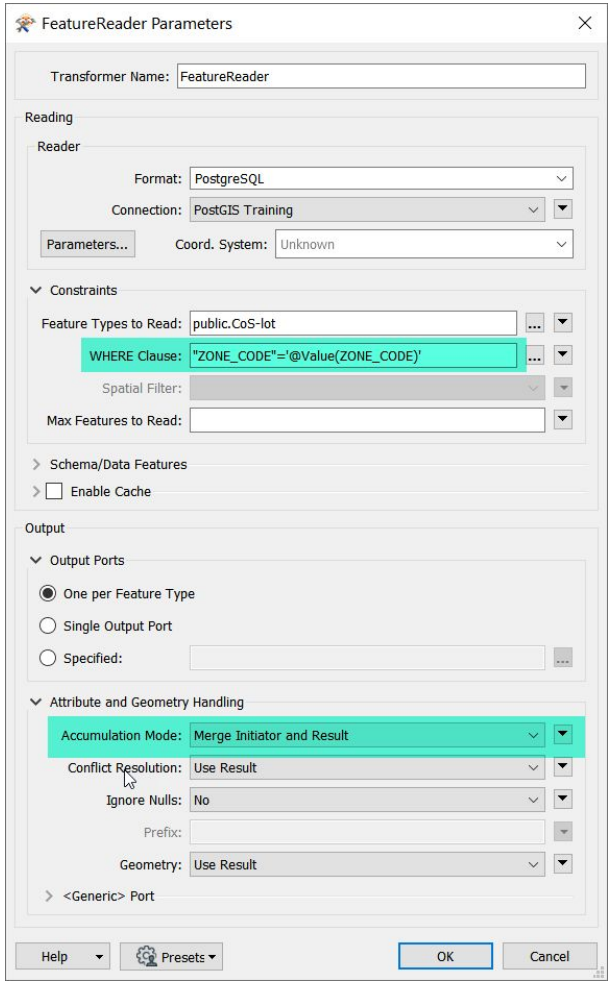
Exercise 4: Obtain Addresses & Generate Report

Midstream Join with FeatureReader

- Simpler midstream option
- Join against any database or spatial format
- Typically used for spatial joins
- Also used to perform tabular joins with database tables



Tips for FeatureReader

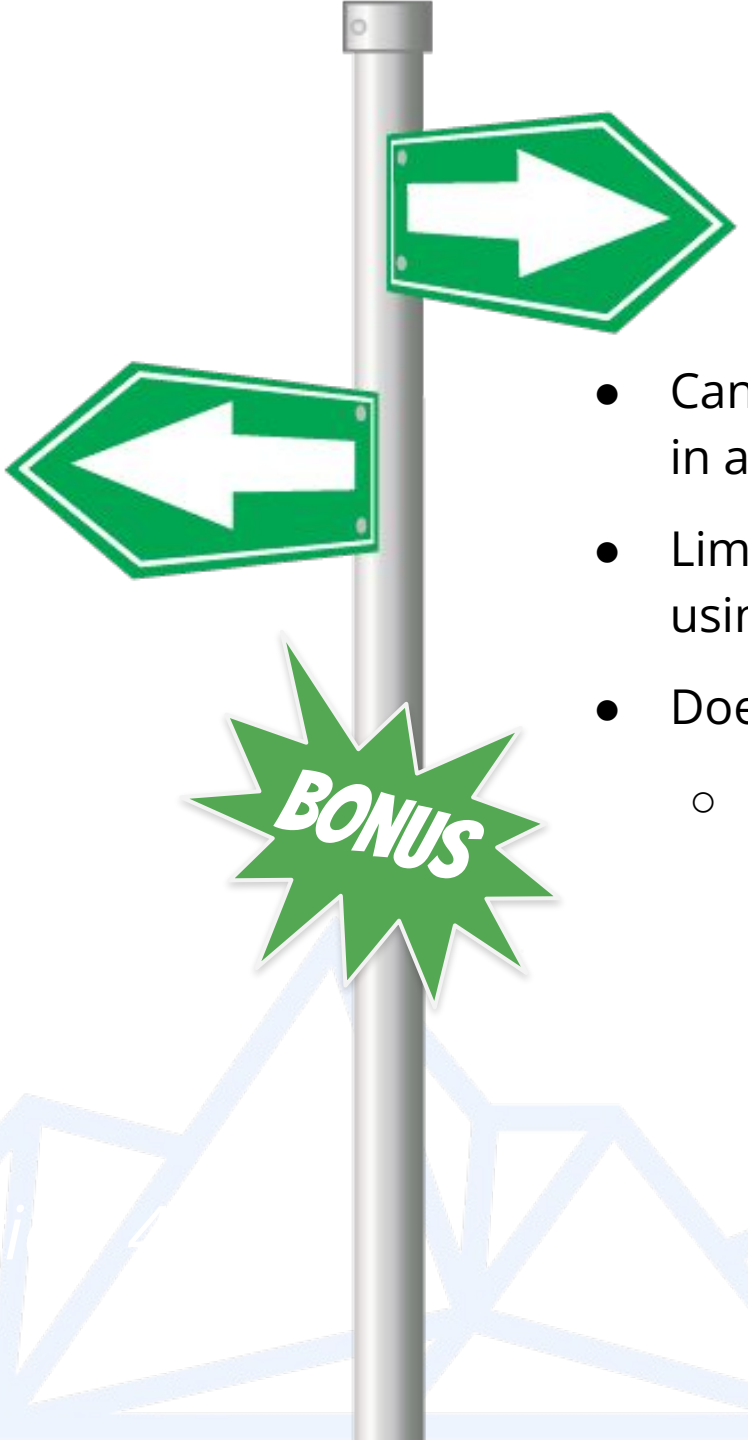


1. Set the WHERE Clause to:
   "<Incoming Attribute>" = '@Value(<Initiator Attribute>)'

2. Set **Accumulation Mode** to **Merge Initiator and Result**

DatabaseJoiner

- Connect to and join to database tables midstream with one easy transformer!
- Can also connect to Excel and CSV
- Select which attributes you want to join
- Explicit cardinality parameters
- Very efficient, especially when using a primed cache
    - Automatically done for File-based joins



FeatureReader

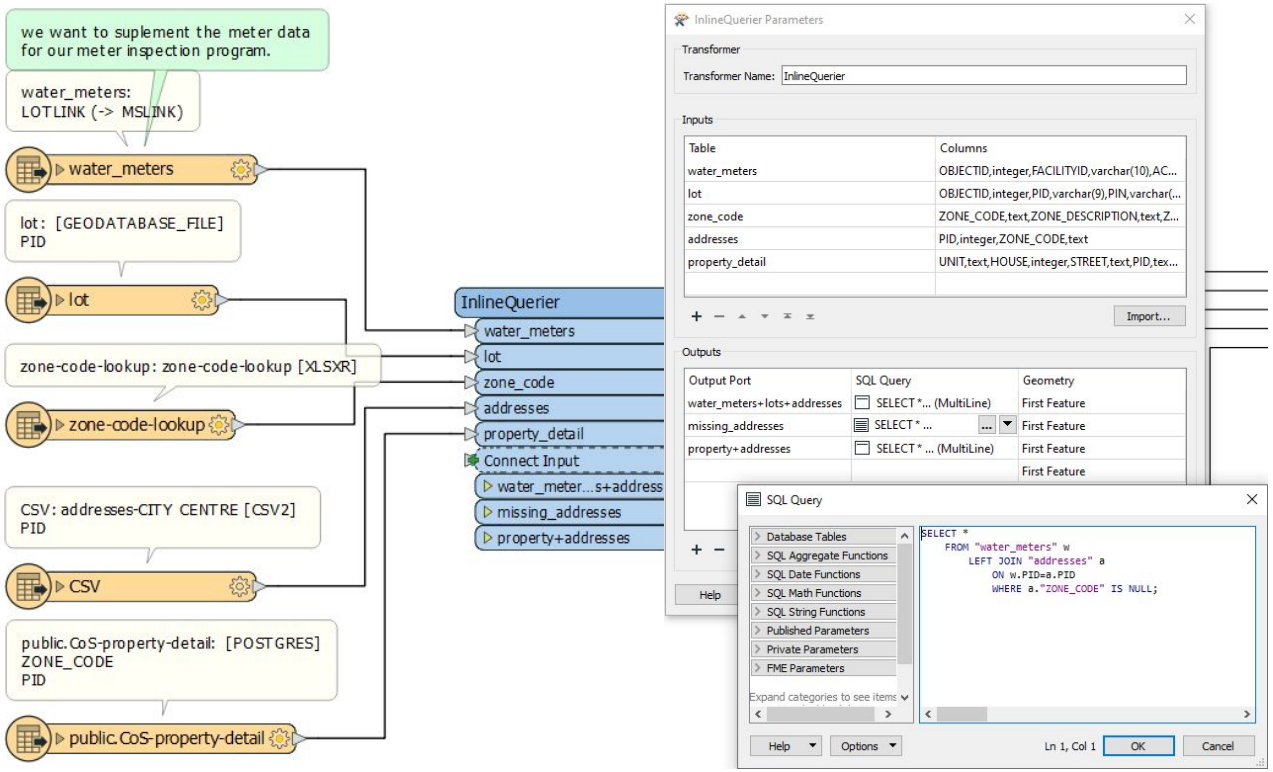
- Can be done midstream which aids in adding and joining data quickly
- Limited options: Inner Joins only using WHERE clause
- Does lots of work per input feature
    - Only efficient when 1 input results in large numbers of output

Exercise 5: **InlineQuerier** | *One transformer to rule them all?*

InlineQuerier

Move over FeatureJoiner:

- Any data source can be used.
- Utilizes the power of SQL
- Multiple queries
- Joins data in a temporary SQLite database
- Includes spatial

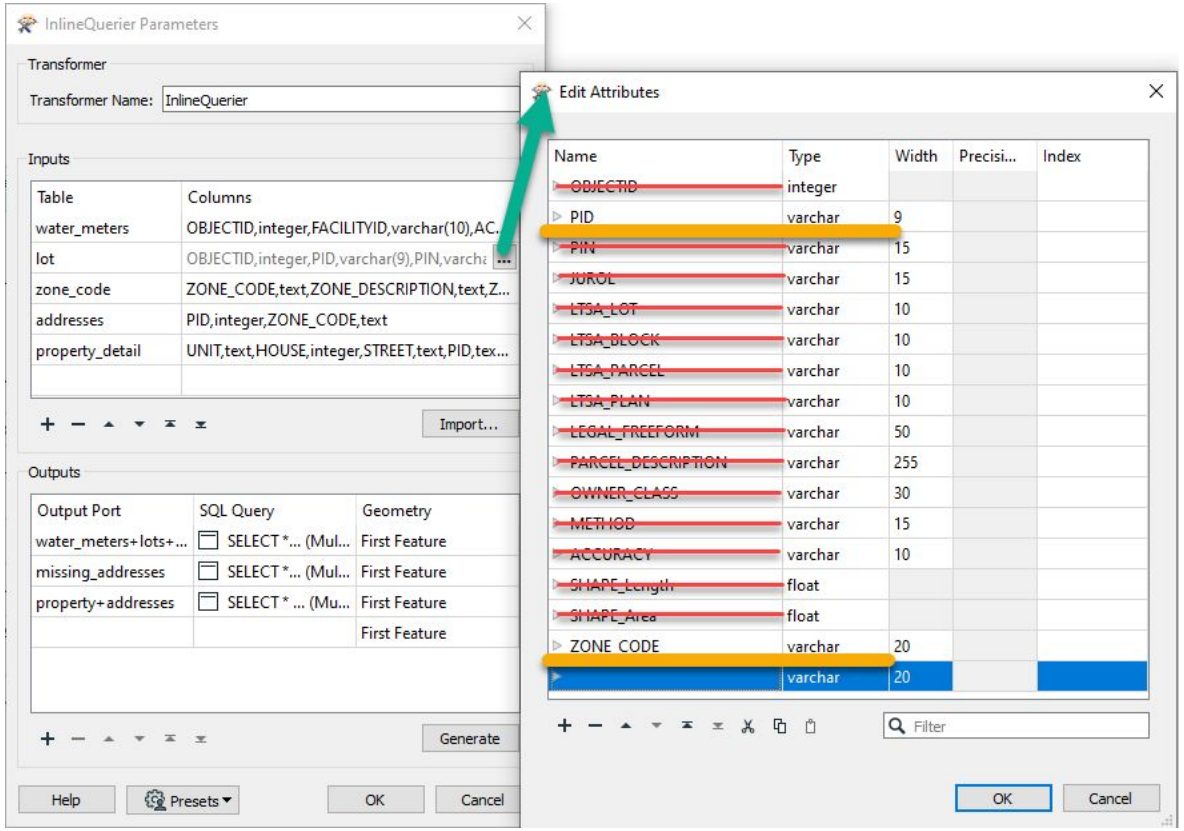


[SQLite Tutorial - An Easy Way to Master SQLite Fast]

Tips for InlineQuerier

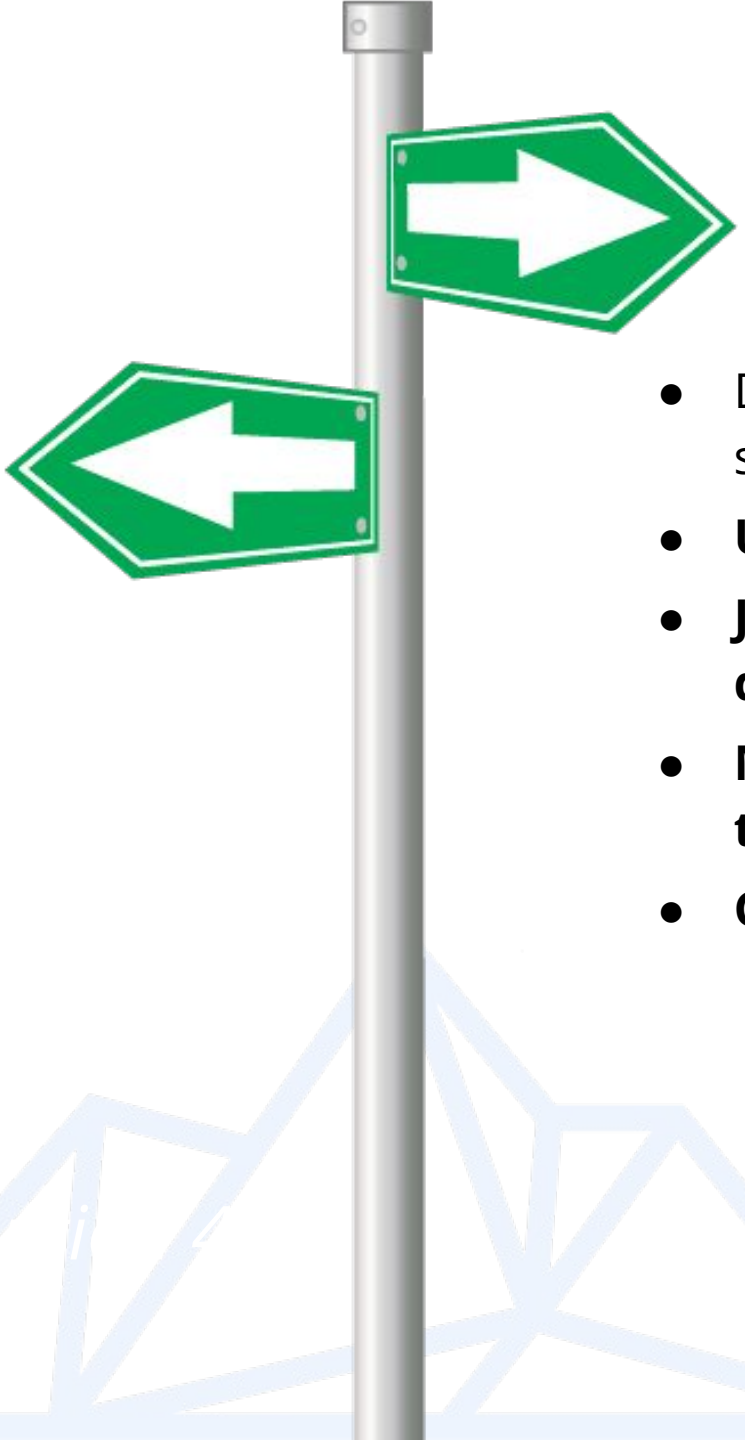
- Loads a temporary SQLite database - so consolidate all your queries if you can
  - Takes some time to load
  - Fast once it's loaded
- Don't use it for simple tasks, i.e. Filtering

- Use smart configuration to improve performance
  - Only define input columns needed in the queries



FeatureJoiner

- Data already loaded, any
- data source can be used
- Uses SQL-Free joins
- Very easy to set-up
- **Great performance**
- **No prior SQL knowledge needed**



InlineQuerier

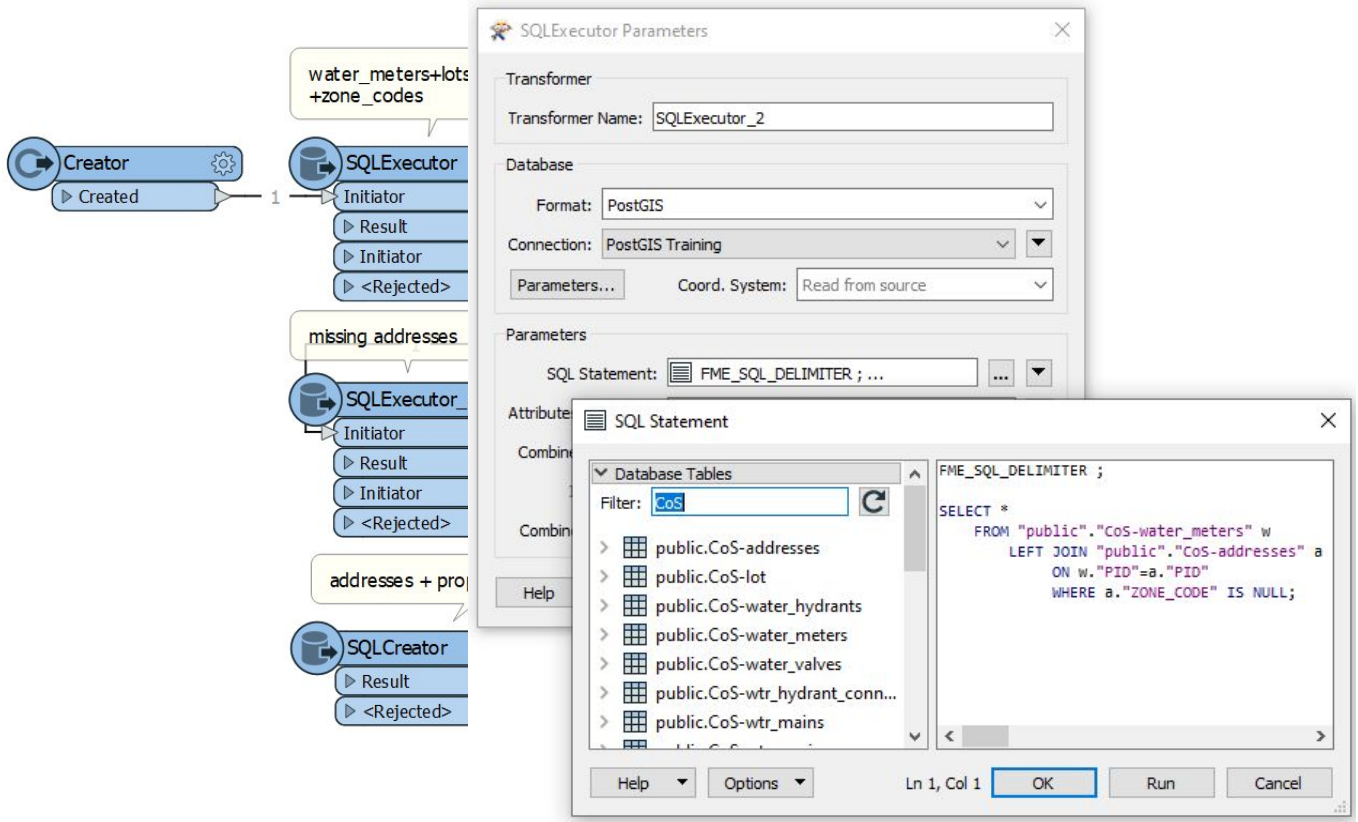
- Data already loaded, any data source can be used
- **Utilizes the power of SQL**
- **Join data in a temporary SQLite database**
- **Multiple queries in a single transformer**
- **Great if you 💕SQL**

Exercise 6: **SQLCreator** & **SQLExecutor** | *Let the database do the work*

SQLExecutor & SQLCreator

- Needs a database
- Utilizes the power of SQL
- You can use ***any*** SQL -
  - Select data
  - Create indices
  - Drop tables

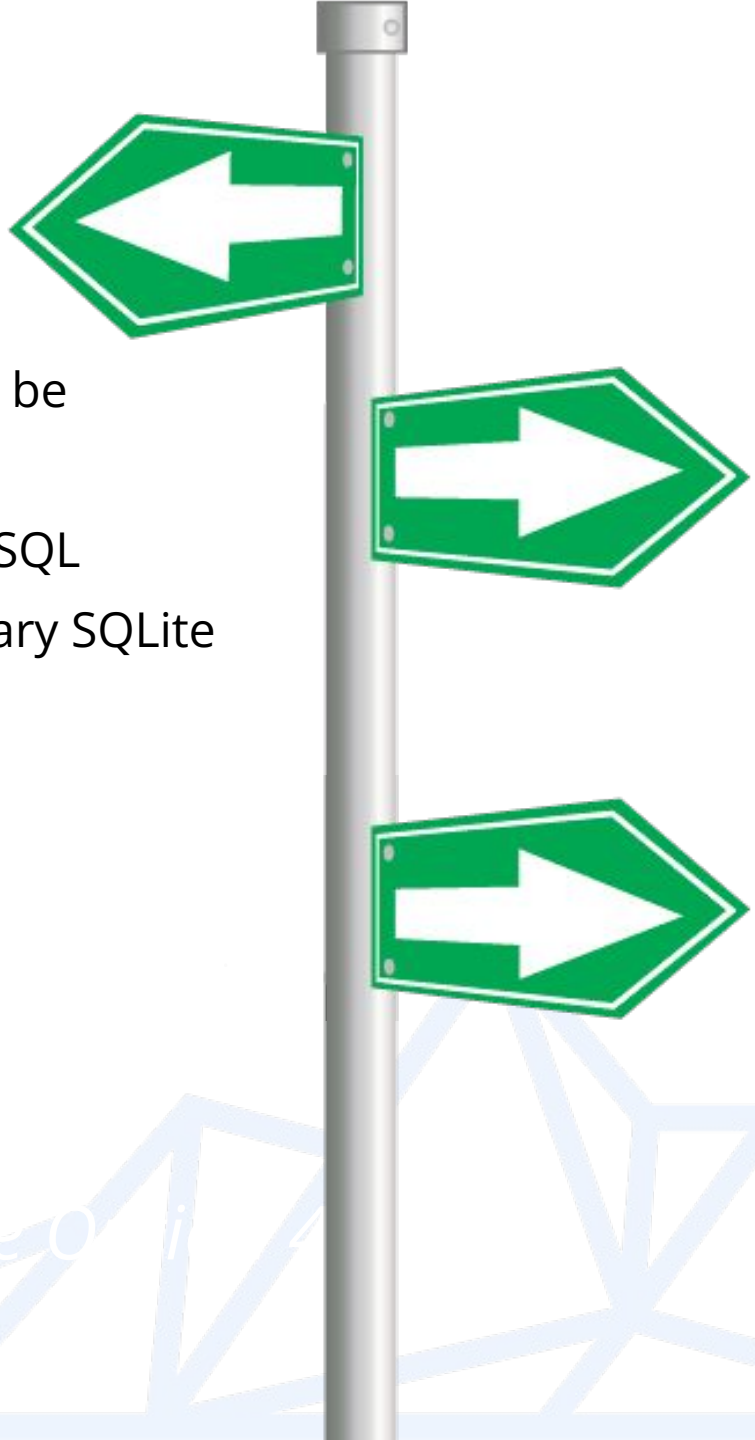
You don't have so just read data



Let Your Database Do The Work

InlineQuerier

- **Any data source** can be used
- Utilizes the power of SQL
- Join data in a temporary SQLite database
- Multiple queries



SQLCreator

- Used to execute SQL against a **database**
- No incoming feature required

SQLExecutor

- Used to execute SQL against a database
- Requires an incoming feature to trigger the SQL statement

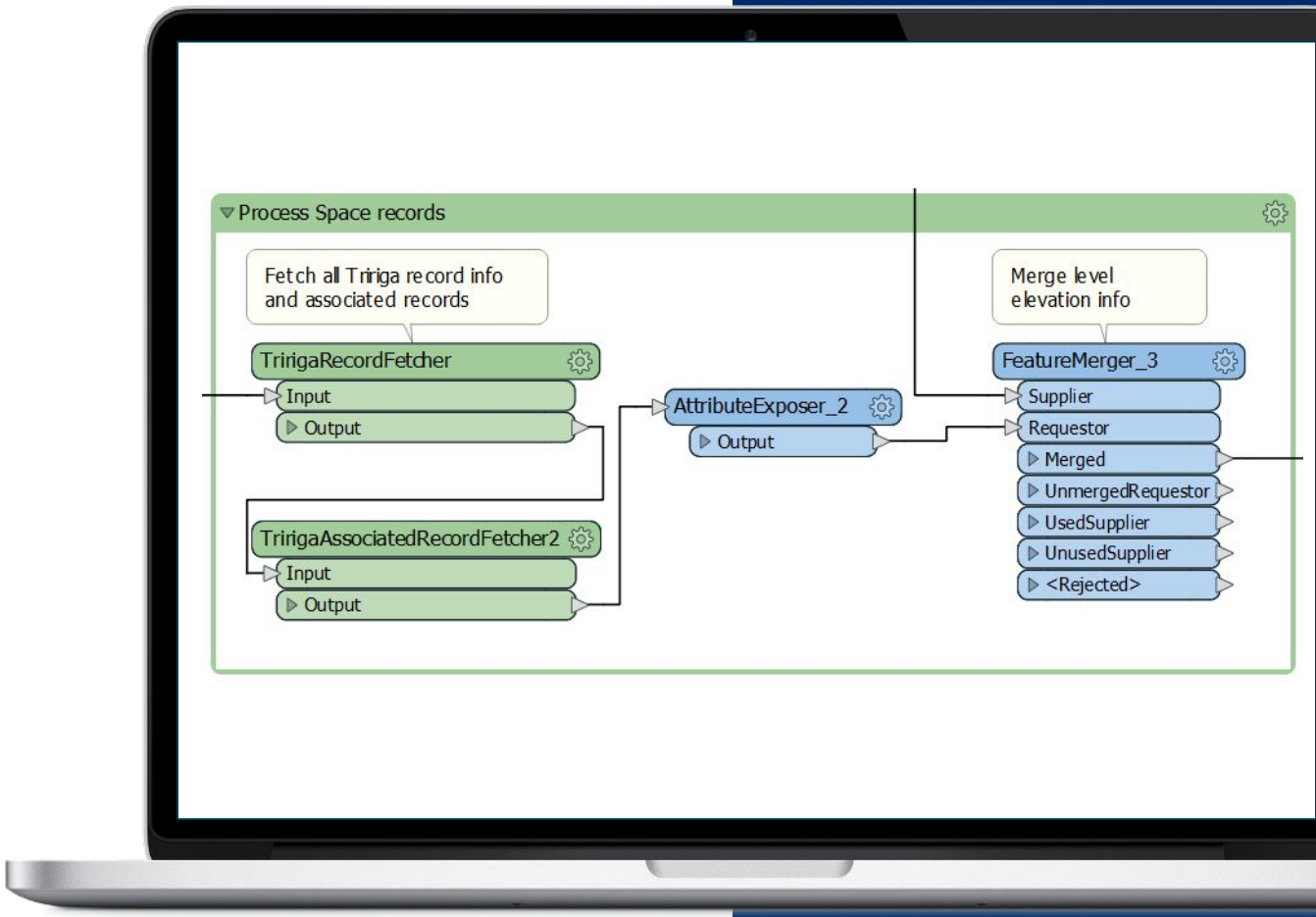
Other Join Ideas

Database views

with SQLExecutor or SQLCreator

API Joins

`<ws:getAllAssociatedRecords>`



Optional Exercise: HTML Report & Workspace App (FME Server)

Wrap Up…

Merging & Joining

CAN BE EASY!

FME TRANSFORMERS CAN DO THE WORK FOR YOU.

Choose the right Transformer to merge & join data more efficiently.

Follow the Flow-Chart

Inspect your data and ask yourself:

1) Is all of my data already inside the workspace?

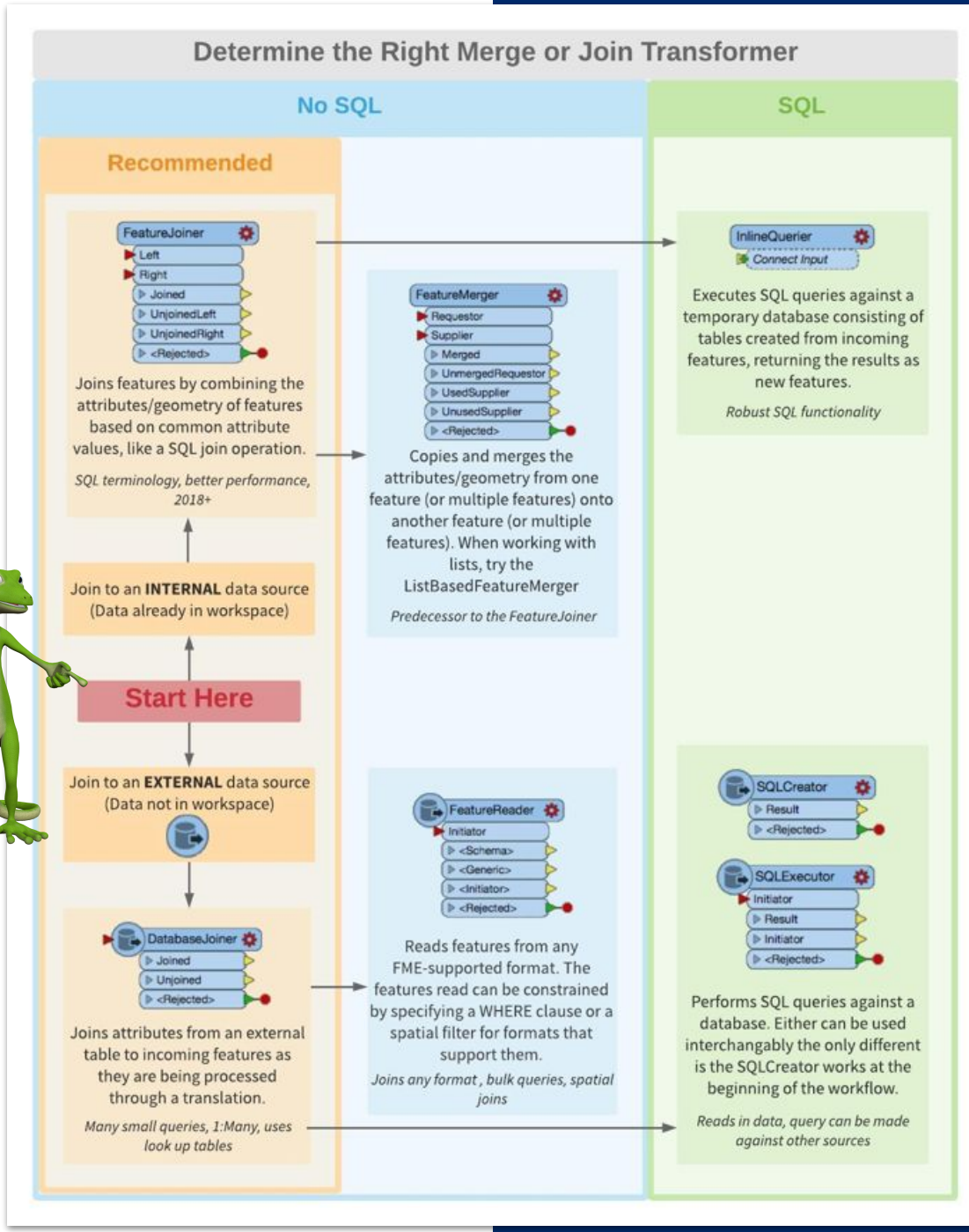
- If yes, follow the INTERNAL path
- If no, follow the EXTERNAL path

2) Does the FeatureJoiner for internal or DatabaseJoiner for external work for my data?

- If yes, Great! End here.
- If no, continue to question 3

3) Do I know or want to use SQL?

- If yes, see the transformers in the green SQL box
- If no, see the transformers in the blue box



Knowledge Base | [Merging or Joining Spreadsheet or Database Data]

Integrate disparate data with FME to breakdown data silos

Resources

Exercises | How to Merge and Join Tabular Data

Knowledge Base | Merging or Joining Spreadsheet or Database Data

Knowledge Base | Merging or Joining Spatial Data

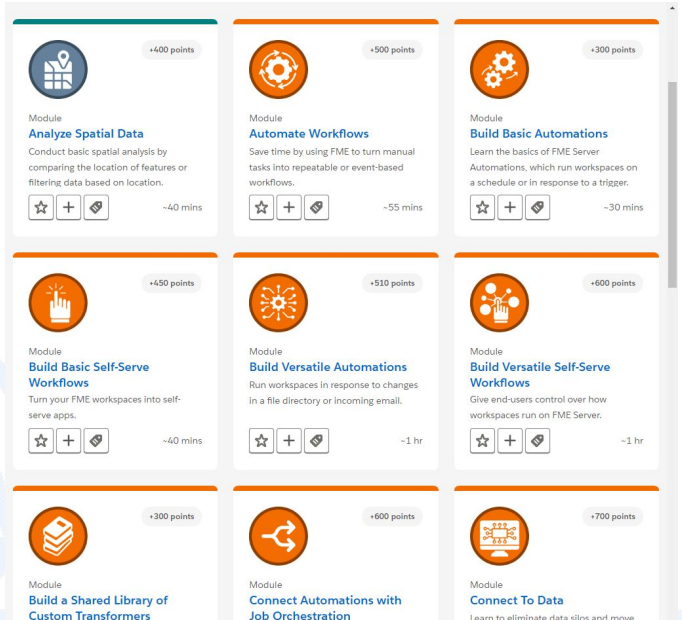
Next steps…

Get involved with the [FME Community]

- Knowledge Base
- Q&A Forums

[FME Academy]

Self directed training modules




THE PEAK OF DATA INTEGRATION
2022 UC
Thank You!
nampreet.singh@safe.com
SAFE SOFTWARE