



# Maintaining 9-1-1 Data Integrity Using FME

May 17, 2022

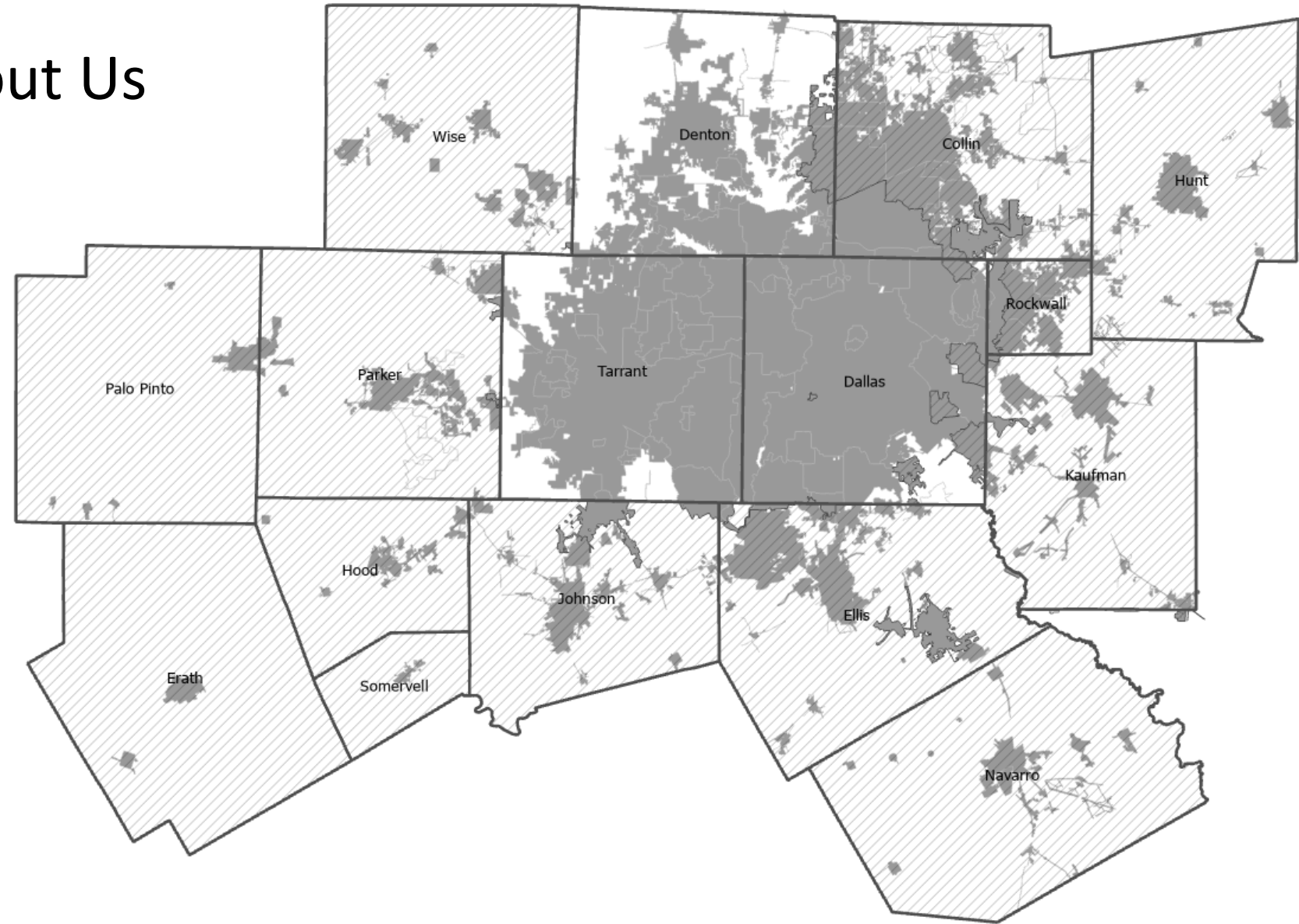
Regional GIS Meeting  
May 2022

# Agenda:

- Introduction
- What are FME/ETL
- Why an ETL
- ETL @ NCT9-1-1
  - Field, Value, and Default Mapping
  - Conditional mapping
- Beyond an ETL
- Resources

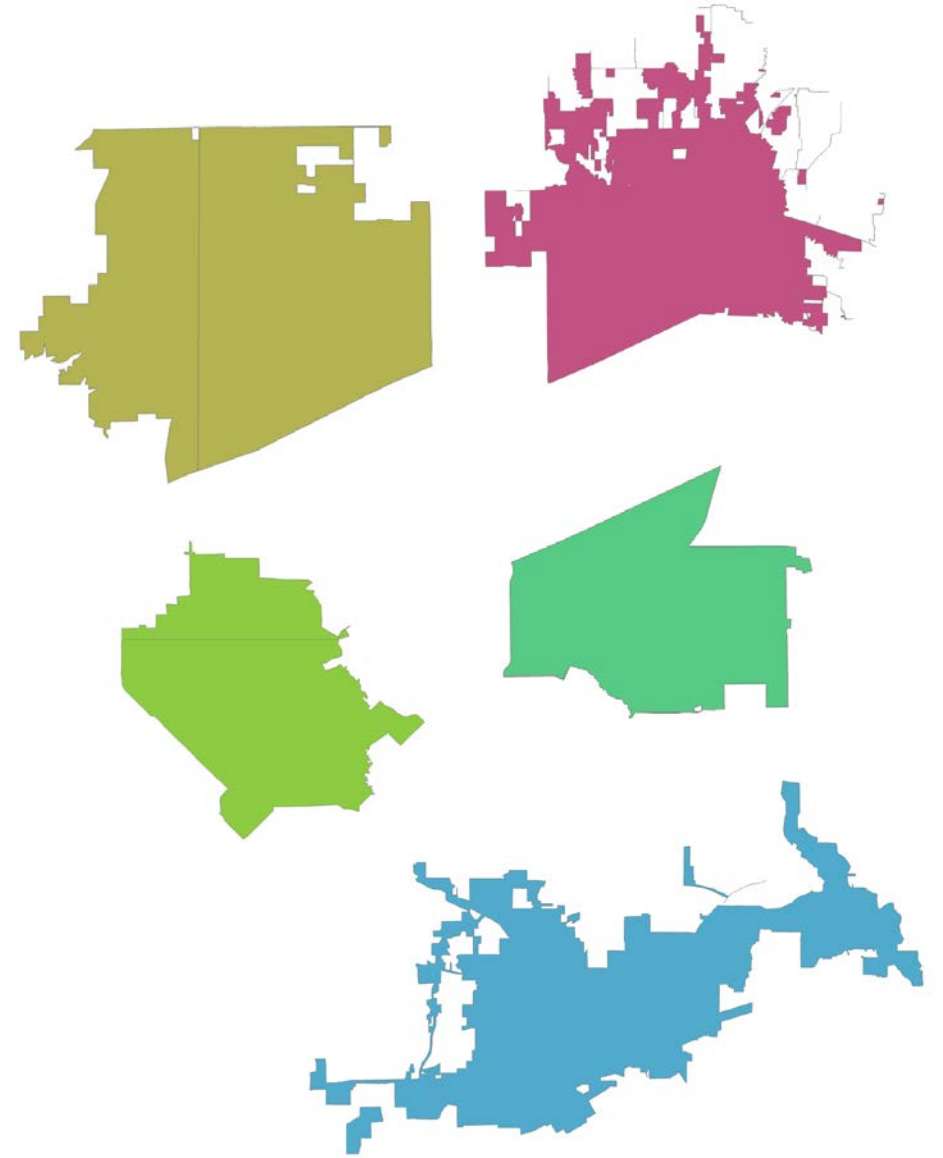
## About Us

- 13 counties in the DFW metroplex. Excluding Dallas\*, Tarrant, and Denton
- 40 **E**mergency **C**ommunication **C**enters (ECC)
- ~2 million citizens served



\*The cities of Sachse, Sunnyvale, Balch Springs, Seagoville, Combine, Wilmer, and Cockrell Hill are part of NCT9-1-1's service area as well

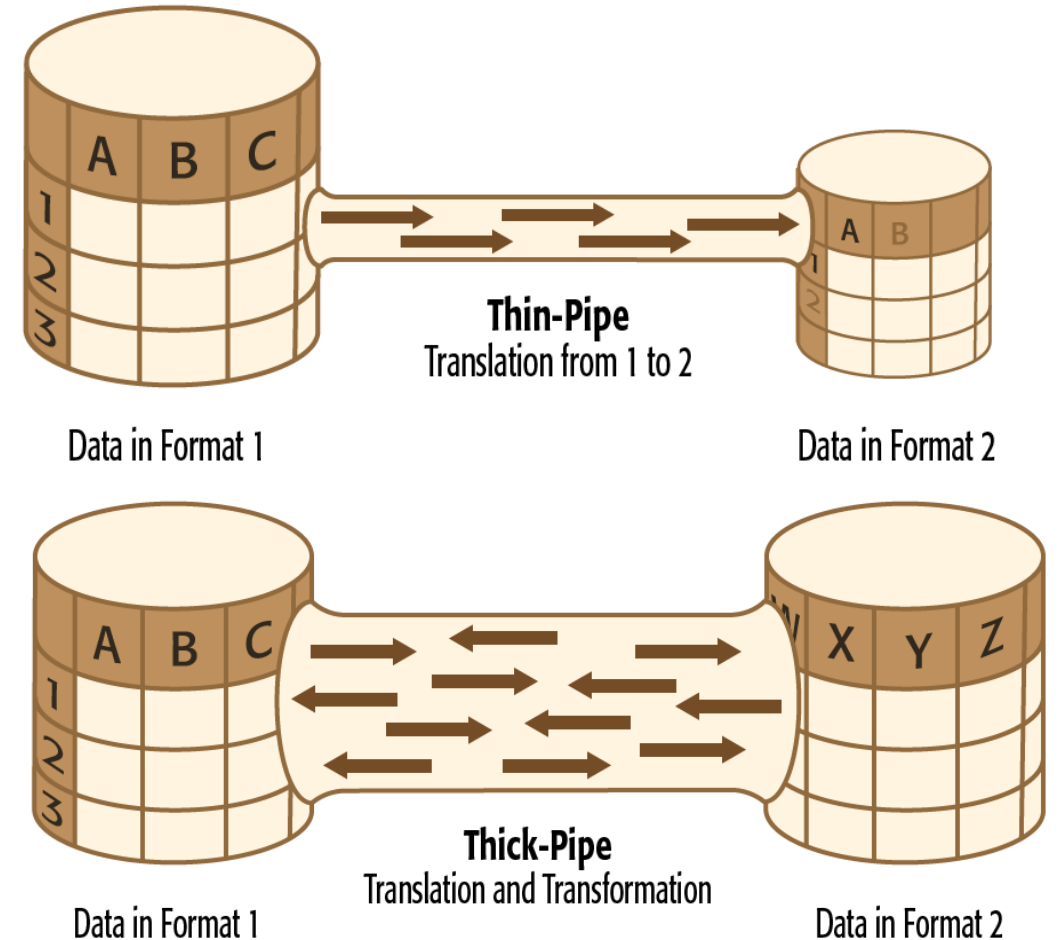
- Dozens of entities using database replication and 5 cities\* submitting data via an ETL
- Biweekly updates to public safety systems
  - ECRF & LVF
  - 9-1-1 Call taking map
  - Locators
  - Supplemental apps
  - Download site



\*The cities are Frisco, McKinney, Allen, Sachse, and Weatherford

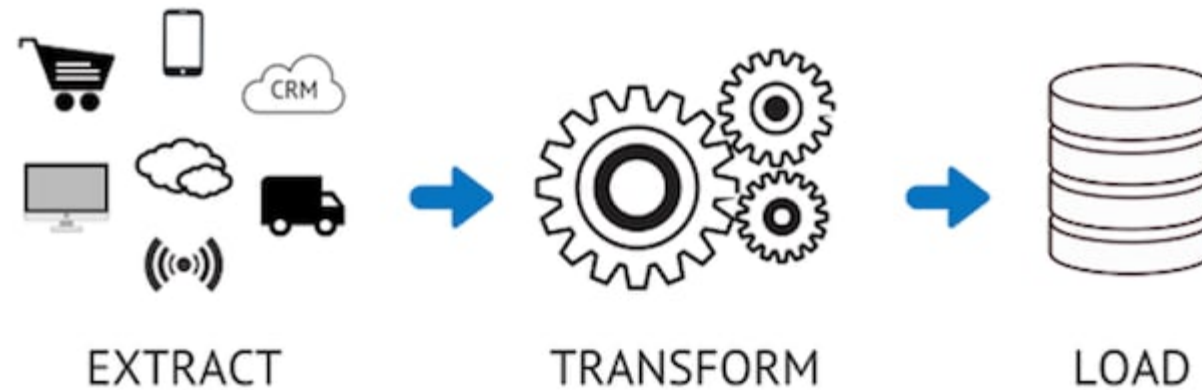
# What is FME

- **Feature Manipulation Engine.** Enables data transformations via its rich data model which covers all possible geometry and attribute types.
- FME was the first tool designed to be a spatial ETL application. Today, FME's ETL capabilities cover many different kinds of data, both spatial and non-spatial.



# What is an ETL

**Extract, Transform, Load.** Process that extracts data from one data source/format, transforms it, and loads in a destination database and format.



Graphics from <https://www.talend.com/resources/what-is-etl/>

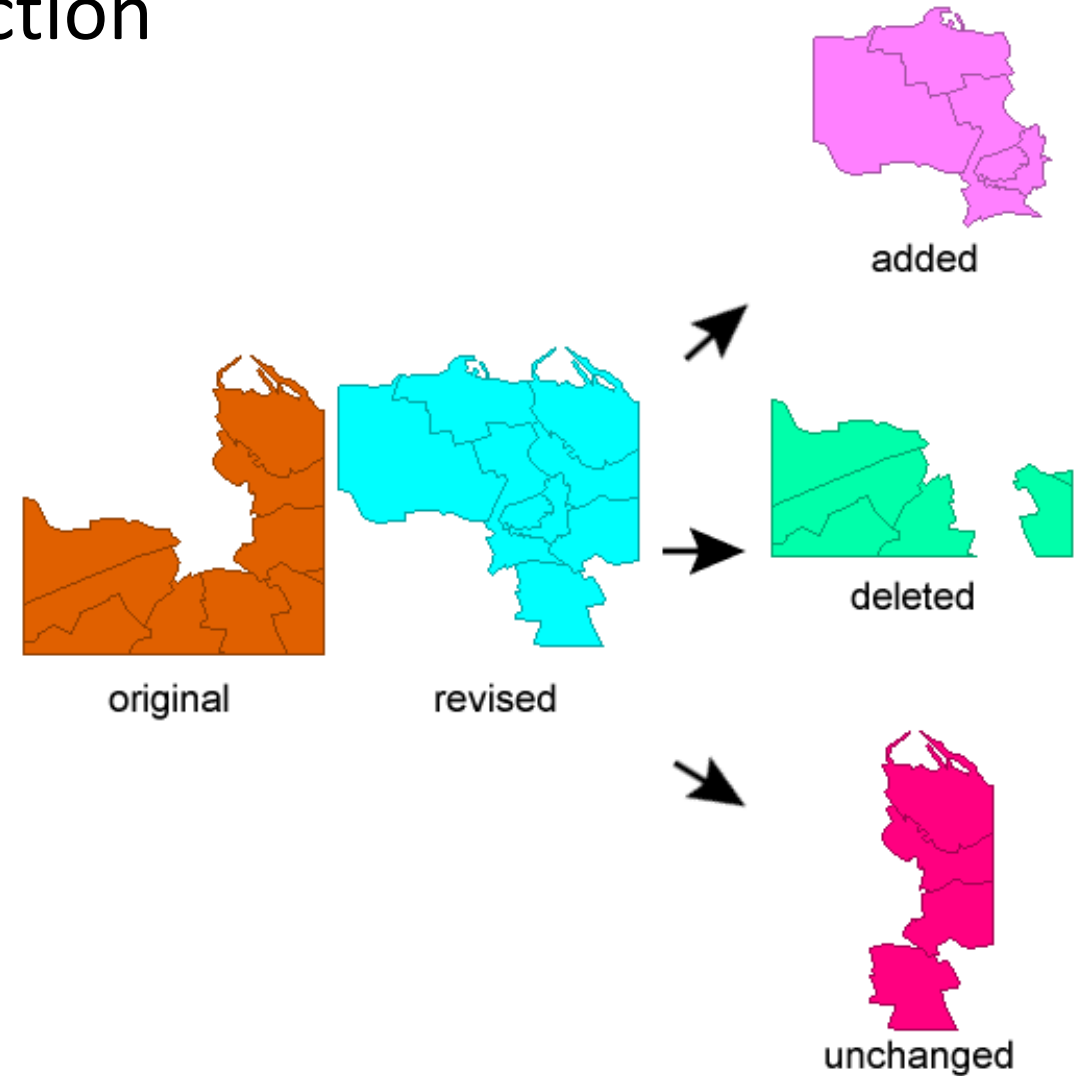
## Why an ETL

- Removes the requirement of everyone having a unified schema
- QAQC can be largely automated via on-the-fly fixes and reporting
- Automation potential from beginning to end
- FME is “data agnostic”
- Change Detection



# Change Detection

- Change detection eliminates the need for a wipe & replace
  - Simple to use & reliable
- Example: New data (**revised**) is compared to existing data in destination databases (**original**)
  - If present in revised but missing in original, **add** to original
  - If it no longer exists in revised, but is present in original, **delete** from original
  - If it exists in both, but is slightly different in each, delete original version and add revised version (“edits” or **delete** + **add**)
  - If identical, leave them **unchanged**





# Field, Value Mapping

- Field mapping allows for *Field A = Field B* translations
- Value mapping allows for *Value 1 of Field A = Value 3 of Field B* translations
- Default mapping allows for *Field B = Uniform Value*

## Field & Default Mapping

GC_Exception	GC_Exception
Source	ALLEN ETL (Default Mapping)
FromAddr_L	LeftFromAddress
ToAddr_L	LeftToAddress
FromAddr_R	RightFromAddress
ToAddr_R	RightToAddress

## Value Mapping

**Attribute Selection**

Source Attribute: COA\_Classification

Destination Attribute: NCT\_Type

Default Value: Problem in NCT\_Type transformer

---

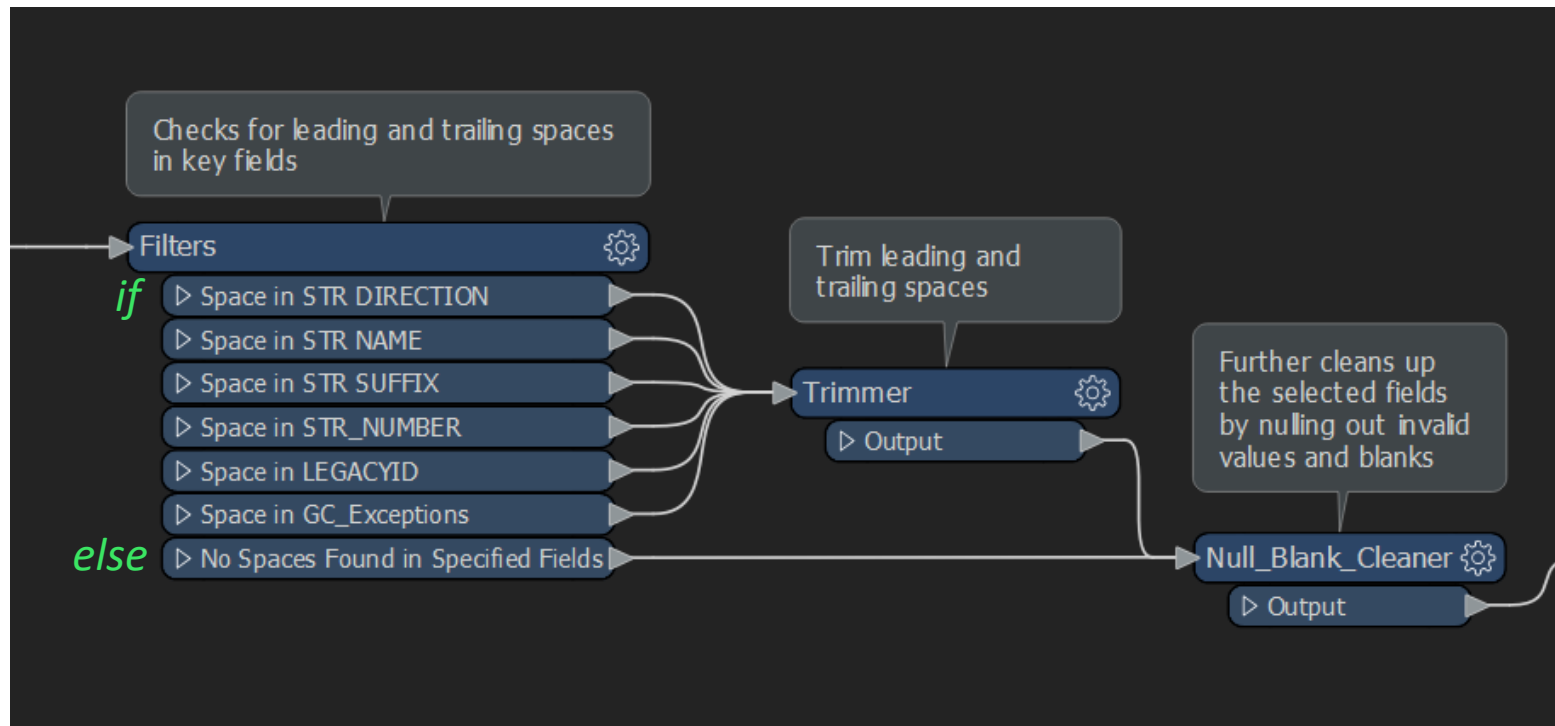
**Value Map**

Mapping Direction: Forward (Source To Destination)

Source Value	Destination Value
<input type="checkbox"/> CONVENIENCE STORE	<input type="checkbox"/> Shop
<input type="checkbox"/> GARAGE_PUBLIC PARKING	<input type="checkbox"/> Garage
<input type="checkbox"/> 2_3_4PLEX	<input type="checkbox"/> Duplex
<input type="checkbox"/> AMUSEMENT_COMM_IN	<input type="checkbox"/> Other
<input type="checkbox"/> ANTIQUE_SHOP	<input type="checkbox"/> Shop
<input type="checkbox"/> APT	<input type="checkbox"/> Apartment
<input type="checkbox"/> AUTO_PAINT_BODY_SHP	<input type="checkbox"/> Shop
<input type="checkbox"/> AUTO_PARTS_SALES	<input type="checkbox"/> Shop
<input type="checkbox"/> AUTO_REPAIR_MAJOR... (MultiLine)	<input type="checkbox"/> Shop
<input type="checkbox"/> AUTO_REPAIR_MAJOR... (MultiLine)	<input type="checkbox"/> Shop
<input type="checkbox"/> AUTO_REPAIR_MINOR	<input type="checkbox"/> Shop

# Conditional Mapping

- “Conditional mapping” refers to any relational “translation” from source to destination features that requires more than a simple 1 = a, Apt = Apartment, etc.
- Similar idea to if/else statements



# Conditional Mapping – Examples

Goal: Create and populate the field “Low Range” from the values of “LeftFromAddress” and “RightFromAddress.” This field stores the lowest of all “from” ranges.

Calculate LowRange:

If LeftFromAddress = 0, <Null>

Then, Low Range = RightFromAddress

Elif RightFromAddress = 0, <Null>

Then, LowRange = LeftFromAddress

Else, take the minimum of LeftFromAddress and RightFromAddress

Test Condition	Attribute Value
If LeftFromAddress ATTRIBUTE_VALUE_NULL OR LeftFromAddress ATTRIBUTE_IS_EMPTY OR LeftFromAddress ATTRIBUTE_IS_MISSING OR @Value(LeftFromAddress) = 0	 RightFromAddress
Else If RightFromAddress ATTRIBUTE_VALUE_NULL OR RightFromAddress ATTRIBUTE_IS_EMPTY OR RightFromAddress ATTRIBUTE_IS_MISSING OR @Value(RightFromAddress) = 0	 LeftFromAddress
Else If	
Else <All Other Conditions>	 @Evaluate(@min(@Value(LeftFromAddress),@Value(RightFromAddress)))

# Conditional Mapping - Examples

Goal: Create and calculate the field “Parity\_L” from the values of “sFromAddr\_L” and “sToAddr\_L.” This field indicates whether the left ranges for a given road are **Even**, **Odd**, **Zero**, or **Both**\*

Calculate Parity\_L:

If sFromAddr\_L ENDS\_WITH with 0,2,4,6,8  
 AND sFromAddr\_L ≠ 0\*\*  
 AND sToAddr\_L ENDS\_WITH with 0,2,4,6,8  
 AND sToAddr\_L ≠ 0\*\*  
 Then **Parity\_L = Even**

Elif sFromAddr\_L ENDS\_WITH with 1,3,5,7,9  
sToAddr\_L ENDS\_WITH with 1,3,5,7,9  
 Then **Parity\_L = Odd**

Elif sFromAddr\_L = 0  
 AND sToAddr\_L = 0  
 Then **Parity\_L = Zero**

Else **Parity = Both**

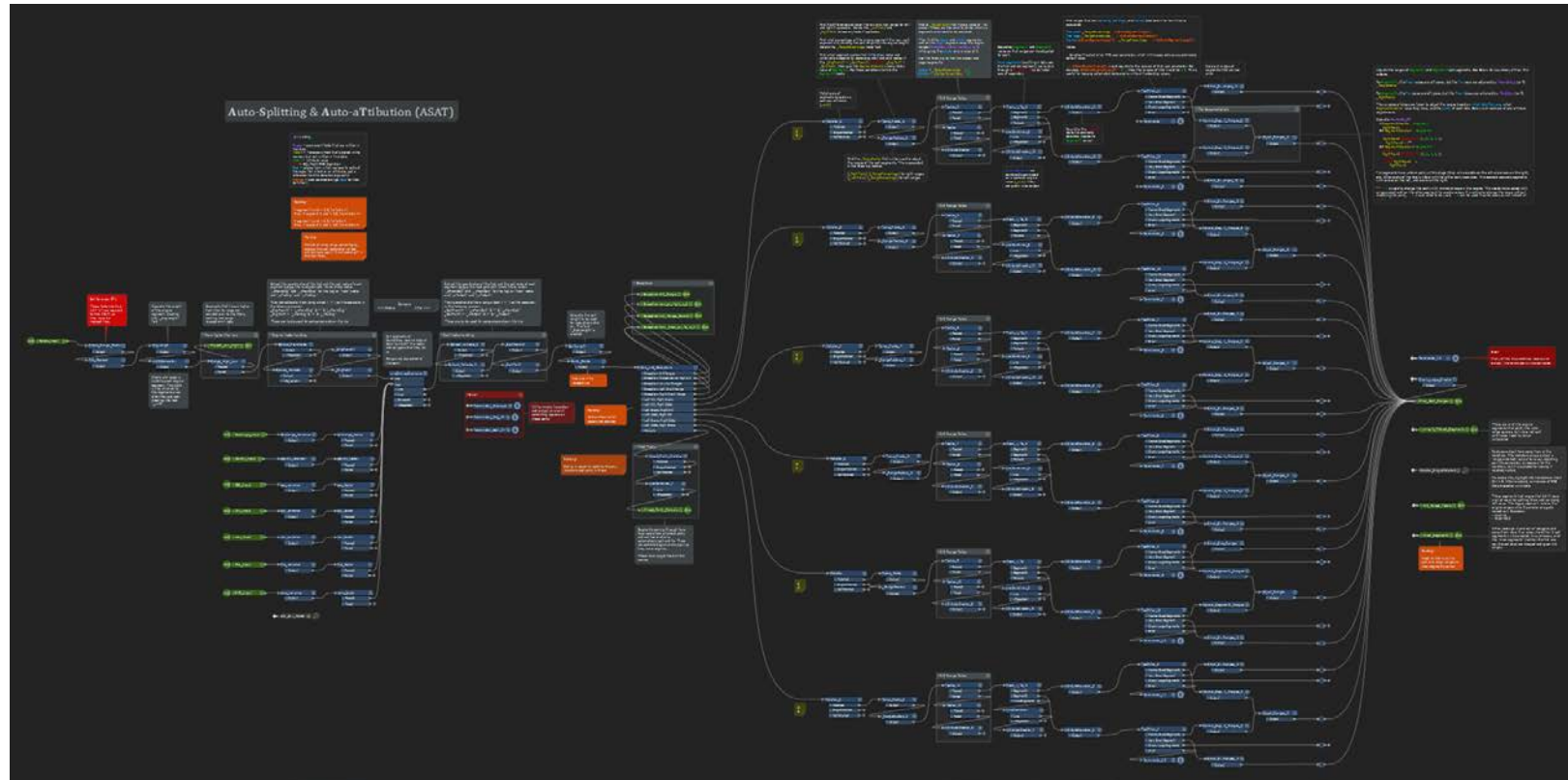
Test Condition	Attribute Value
If @Value(sFromAddr_L) ENDS_WITH 0 @Value(sFromAddr_L) ENDS_WITH 2 @Value(sFromAddr_L) ENDS_WITH 4 @Value(sFromAddr_L) ENDS_WITH 6 @Value(sFromAddr_L) ENDS_WITH 8 @Value(sFromAddr_L) NOT = 0 @Value(sToAddr_L) ENDS_WITH 0 @Value(sToAddr_L) ENDS_WITH 2 <4 more test(s)> ... Composite Test: ( 1 OR 2 OR 3 OR 4 OR 5 AND 6 ) AND ( 7 OR 8 OR 9 OR 10 O...	<input type="checkbox"/> E
Else If @Value(sFromAddr_L) ENDS_WITH 1 @Value(sFromAddr_L) ENDS_WITH 3 @Value(sFromAddr_L) ENDS_WITH 5 @Value(sFromAddr_L) ENDS_WITH 7 @Value(sFromAddr_L) ENDS_WITH 9 @Value(sToAddr_L) ENDS_WITH 1 @Value(sToAddr_L) ENDS_WITH 3 @Value(sToAddr_L) ENDS_WITH 5 <2 more test(s)> ... Composite Test: ( 1 OR 2 OR 3 OR 4 OR 5 ) AND ( 6 OR 7 OR 8 OR 9 OR 10 )	<input type="checkbox"/> O
Else If @Value(sFromAddr_L) = 0 AND @Value(sToAddr_L) = 0	<input type="checkbox"/> Z
Else If <All Other Conditions>	<input type="checkbox"/> B

\*These special scenarios are dealt with at a later step

\*\* This accounts for numbers that end in 0 but do not equal 0. E.g.: 10, 20, etc.

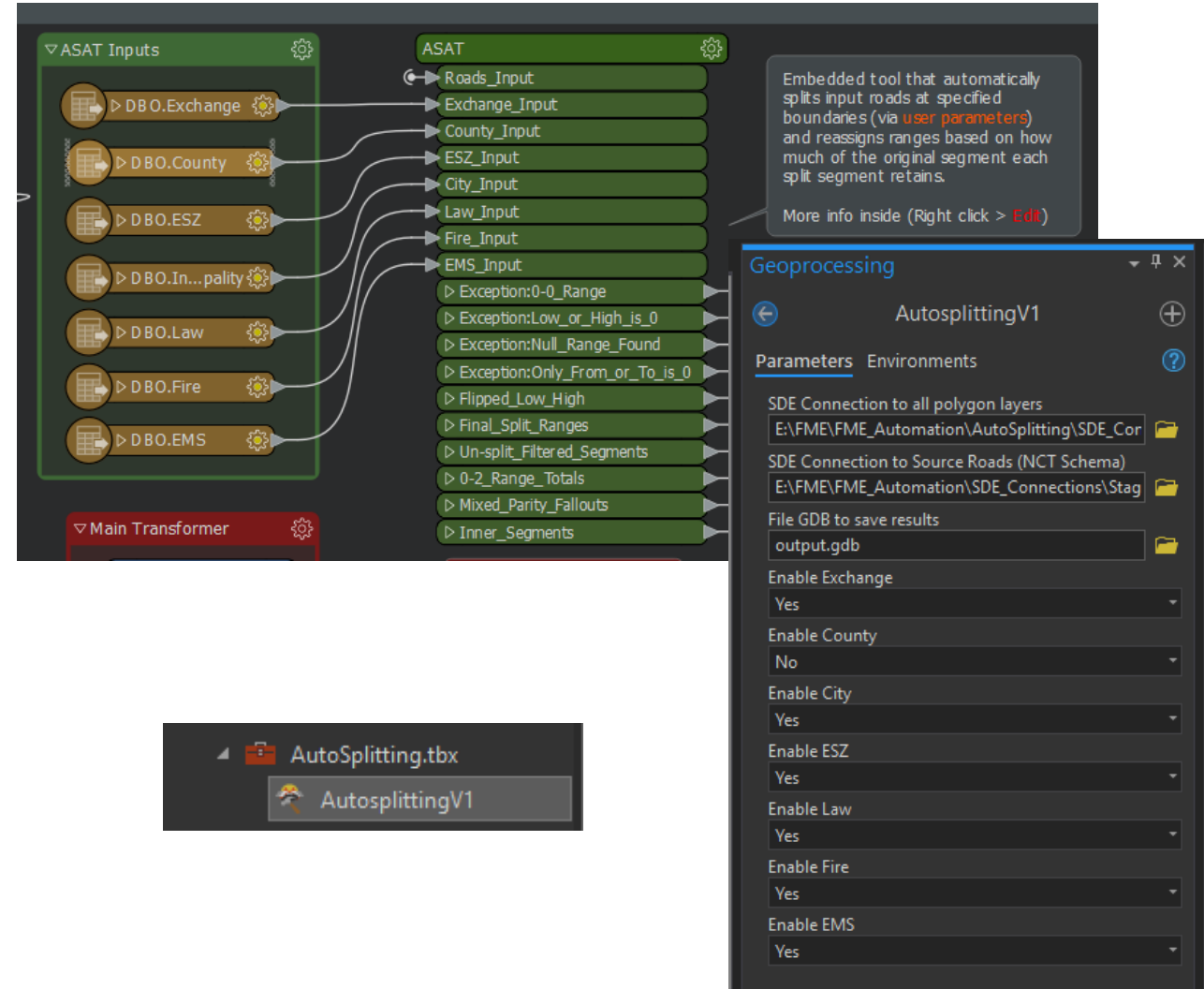
# Conditional Mapping – Examples

Goal: Split incoming road segments at specified boundaries (city limits, county boundary, etc.), assign the corresponding left & right values (County\_L, County\_R, etc.), and rearrange the road ranges based on how much of the original road each segment retained.



# Beyond an ETL

- Custom tools can be created within an ETL or as a standalone tool. These tools can help with:
  - QAQC
  - Data quality
  - Reporting
  - General automation
- Can integrate existing scripts & automation
- [FME server](#) allows for further automation, scheduling, etc.



Embedded tool that automatically splits input roads at specified boundaries (via **user parameters**) and reassigns ranges based on how much of the original segment each split segment retains.

More info inside (Right click > **Edit**)

**Geoprocessing**

AutosplittingV1

Parameters Environments

SDE Connection to all polygon layers  
E:\FME\FME\_Automation\AutoSplitting\SDE\_Cor

SDE Connection to Source Roads (NCT Schema)  
E:\FME\FME\_Automation\SDE\_Connections\Stag

File GDB to save results  
output.gdb

Enable Exchange  
Yes

Enable County  
No

Enable City  
Yes

Enable ESZ  
Yes

Enable Law  
Yes

Enable Fire  
Yes

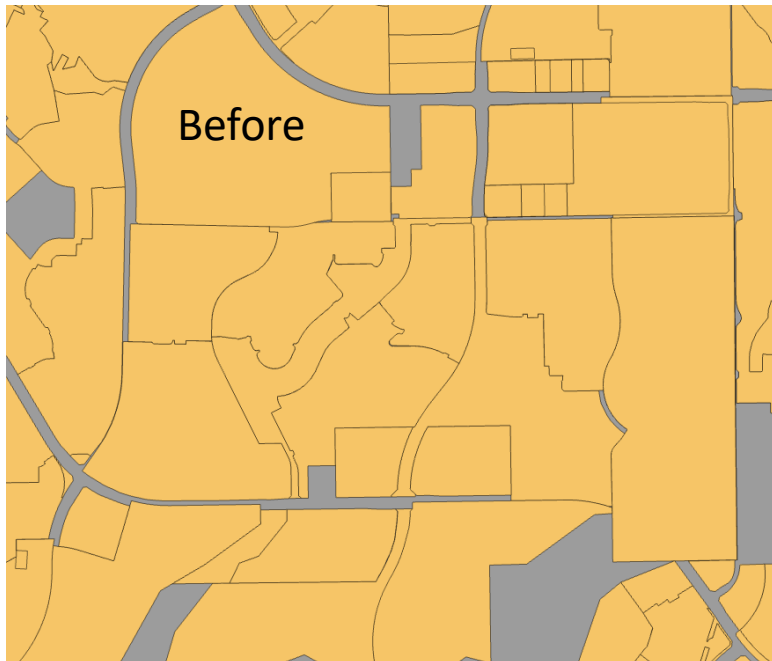
Enable EMS  
Yes

AutoSplitting.tbx

AutosplittingV1

## Beyond an ETL - Examples

- “Neighborhoods” tool
  - Tool that downloads latest subdivision layer from relevant appraisal districts, cleans up **geometries** & labels, and loads finalized product in destination databases



# Beyond an ETL - Examples

- “Neighborhoods” tool
  - Tool that downloads latest subdivision layer from relevant appraisal districts, cleans up geometries & **labels**, and loads finalized product in destination databases

*Before*



*After*



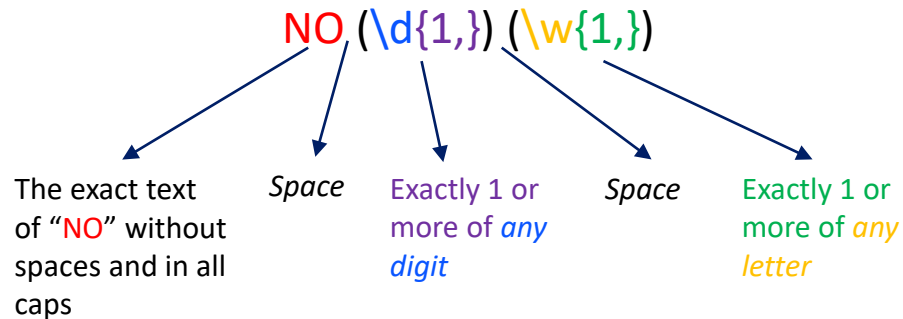


# Beyond an ETL - Examples

- “Neighborhoods” tool
  - Can manipulate strings via SQL or regular expressions (regex)

Look for...

Logic	Left Value	Operator	Right Value
	← Nbrhd_Comm	Contains Regex	<input type="checkbox"/> NO (\d{1,}) (\w{1,})



Quick Reference

[abc]	A single character of: a, b, or c	\d	Any digit
[^abc]	Any single character except: a, b, or c	(a b)	a or b
[a-z]	Any single character in the range a-z	a?	Zero or one of a
[a-zA-Z]	Any single character in the range a-z or A-Z	a*	Zero or more of a
^	Start of line	a+	One or more of a
\$	End of line	a{3}	Exactly 3 of a
.	Any single character	a{3,}	3 or more of a
\s	Any whitespace character	a{3,6}	Between 3 and 6 of a
\w	Any word character		

Regular Expression (case sensitive)

NO (\d{1,}) (\w{1,})

Test String

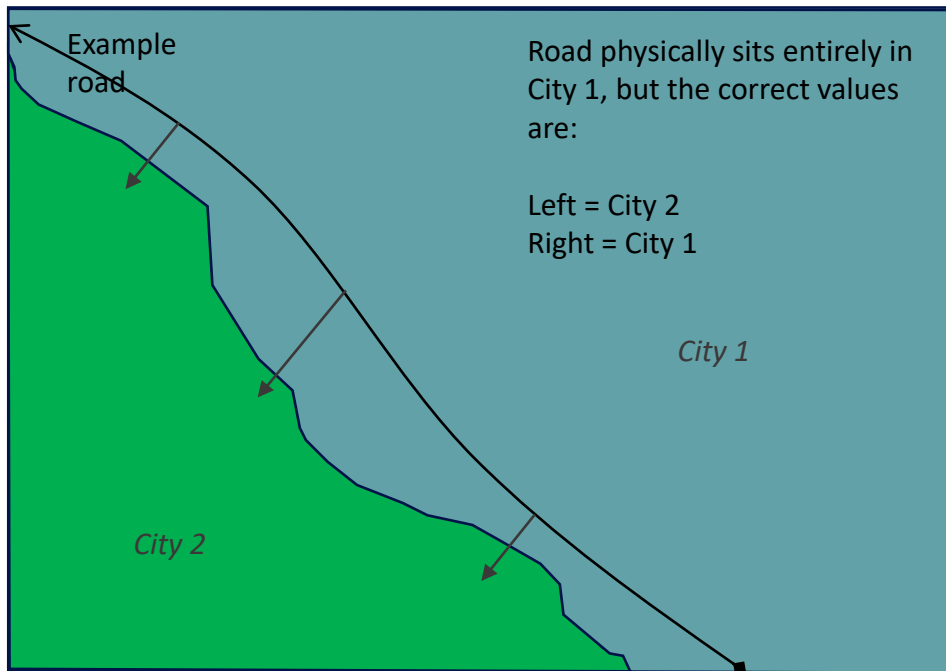
THIS IS AN EXAMPLE NEIGHBORHOOD LABEL. NO A 1 DOES NOT MATCH BUT NO 1 A DOES! NO 189765 G WILL WORK TOO, BUT NOT NO G 124!

Results

Contains 2 Matches

# Beyond an ETL - Examples

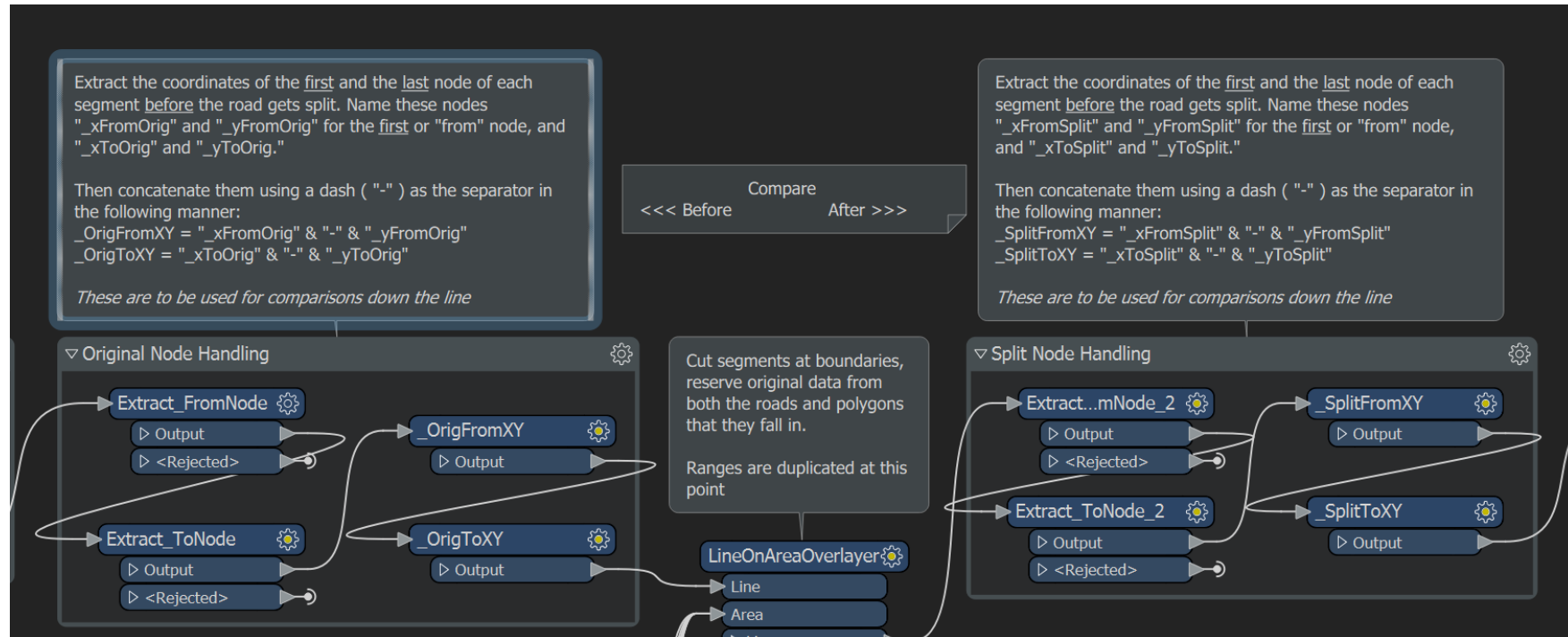
- Road Centerline “Bufferer”\*
  - Tool that grabs attributes from nearby polygons and not where the road sits in



\*Still just a prototype

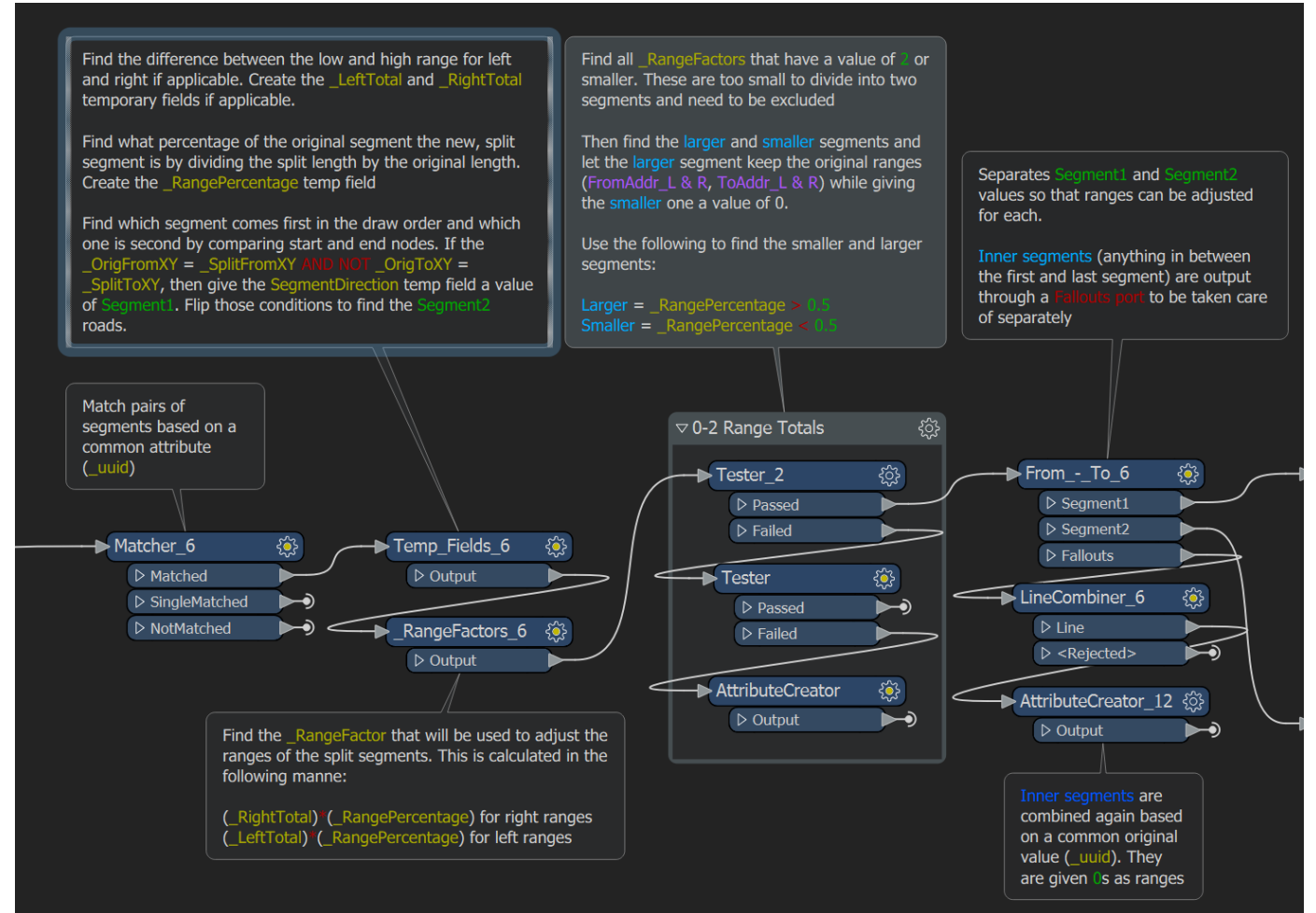
# Beyond an ETL – Examples (ASAT)

Logical step 1: extract xy coordinates before and after splitting (used later for determining the order of splits)



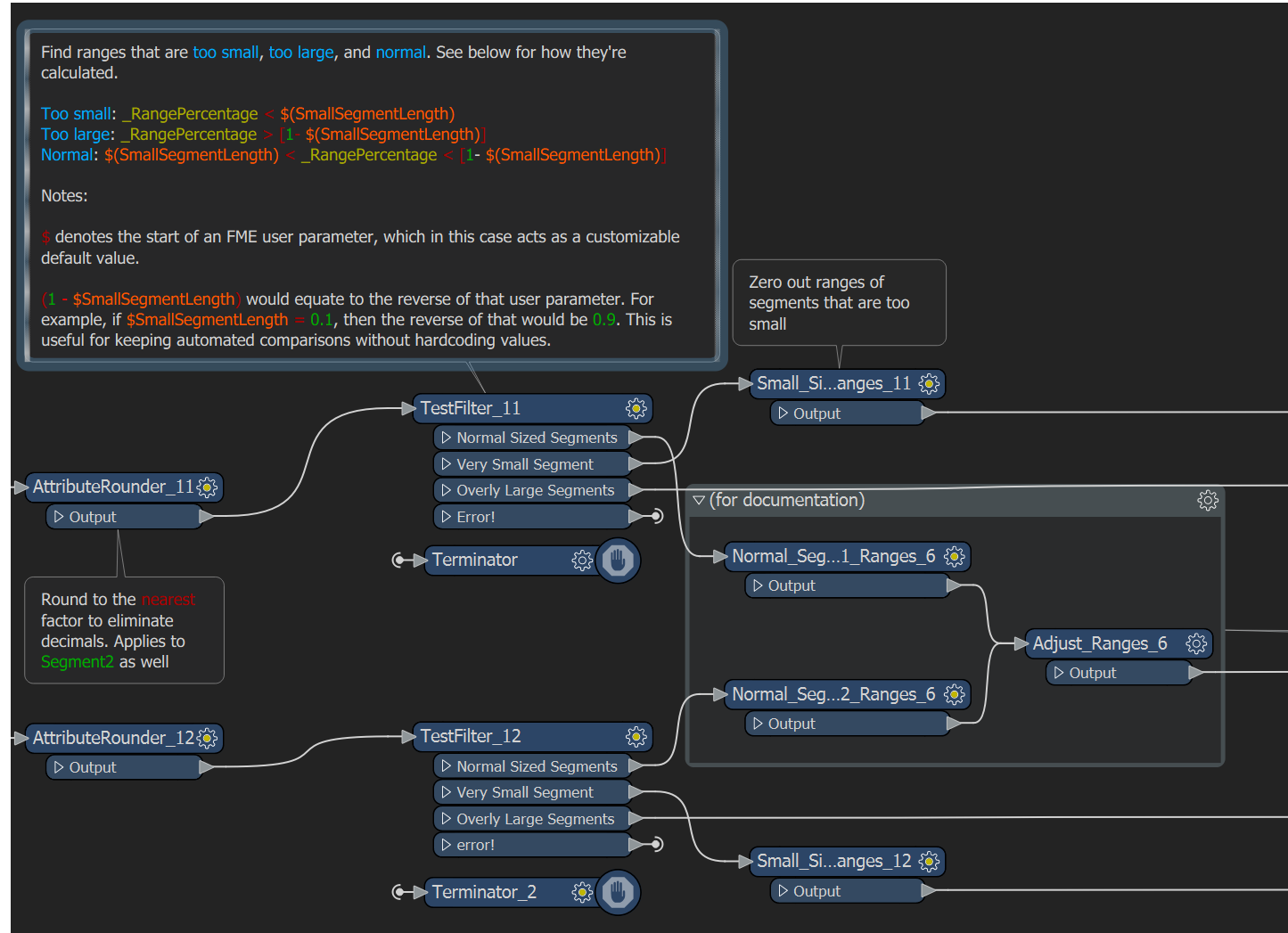
# Beyond an ETL – Examples (ASAT)

Logical step 2: Determine order of splits and how much of the original segment was retained after splits. Create temporary fields to be used to calculate the new ranges later on



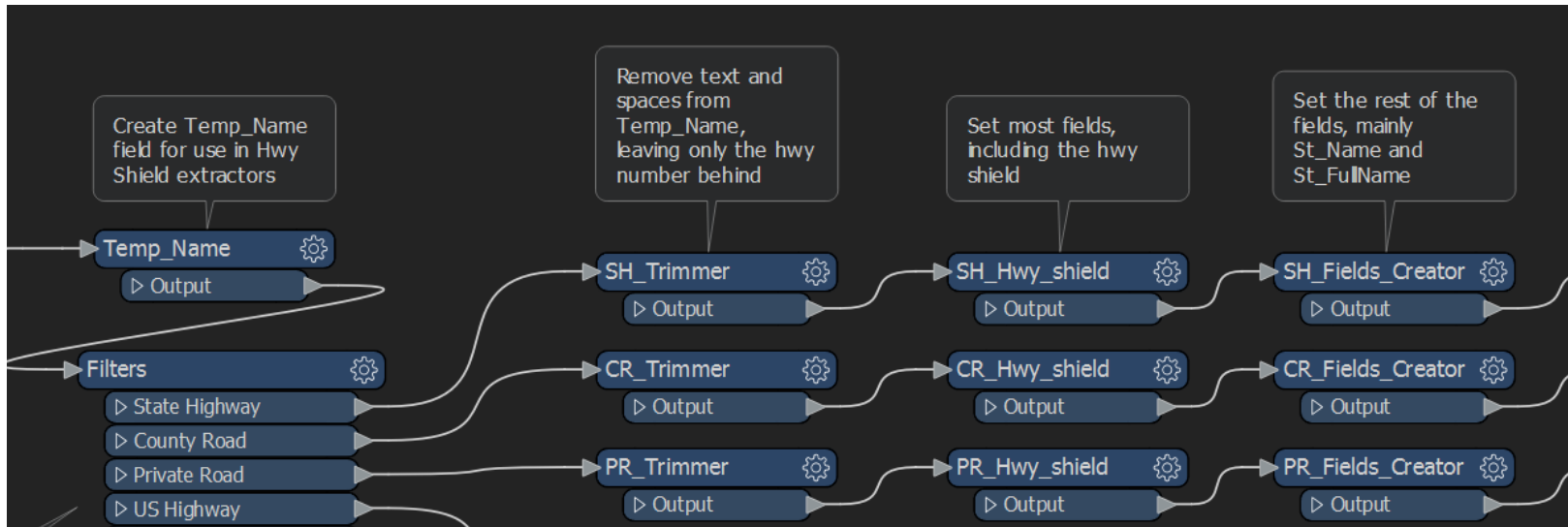
# Beyond an ETL – Examples (ASAT)

Logical step 2: using temporary fields, calculate the left and right ranges of both split segments. Add exceptions for segments that are too short or too long



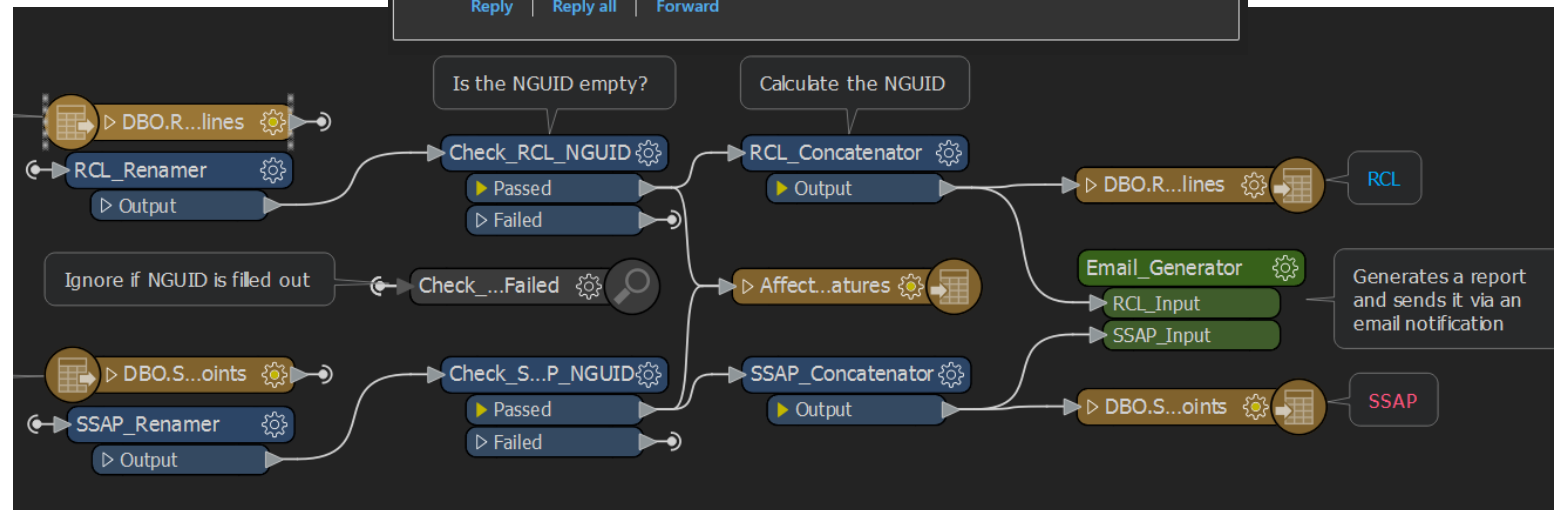
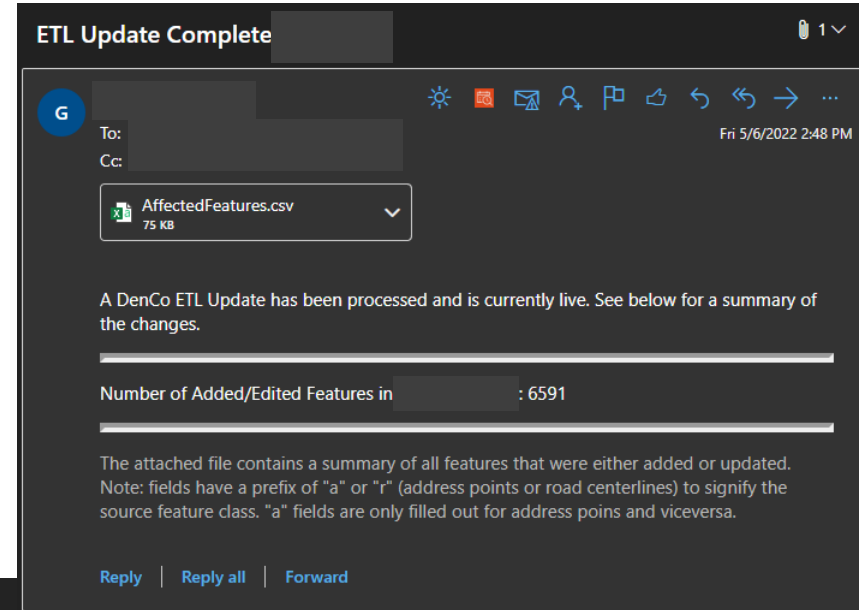
# Beyond an ETL - Examples

- Extracting highway shields from street names
  - Populate highway shield (or temporary field) with street name, then remove text and spaces
  - [Attribute Trimmer](#), [String Replacer](#)



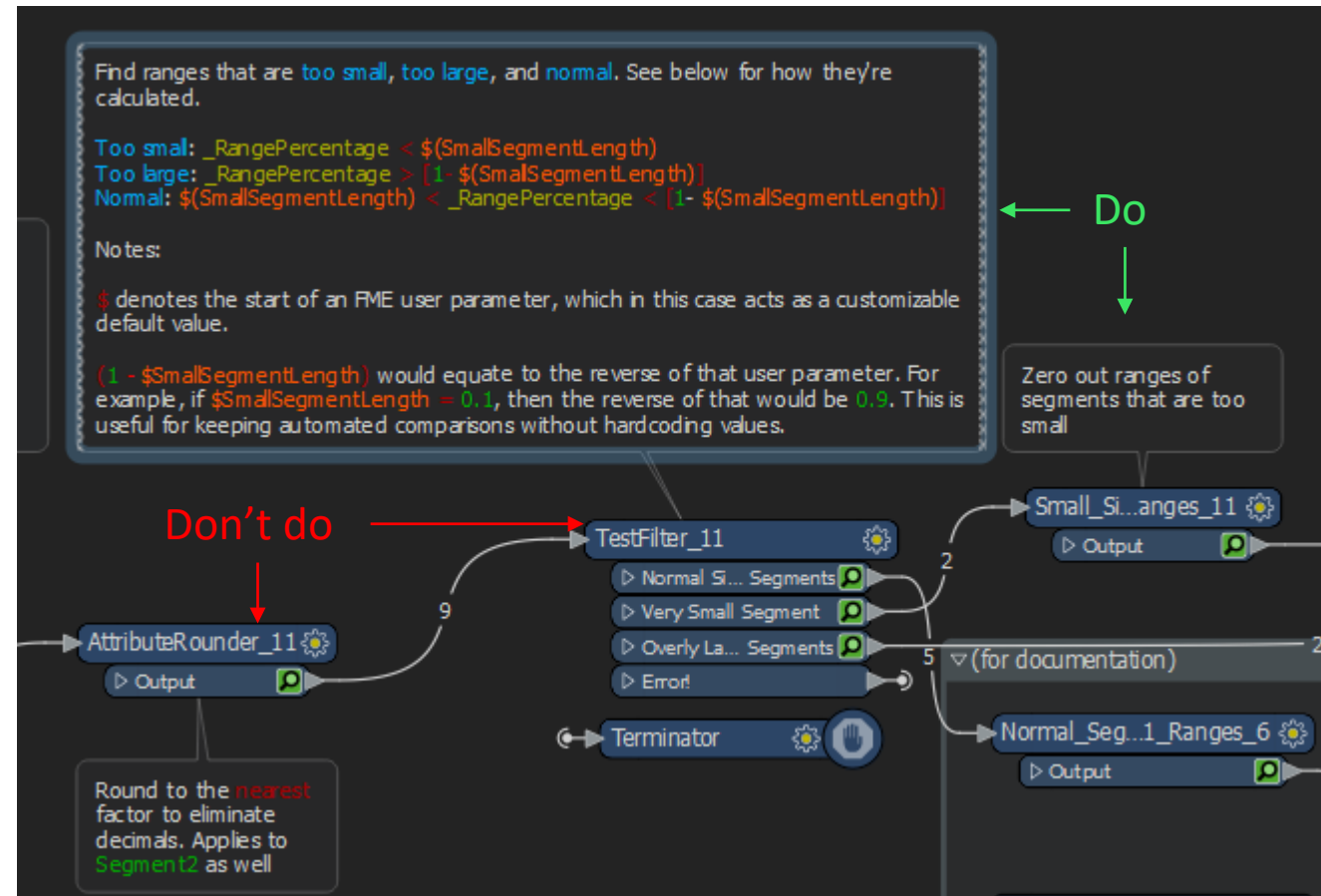
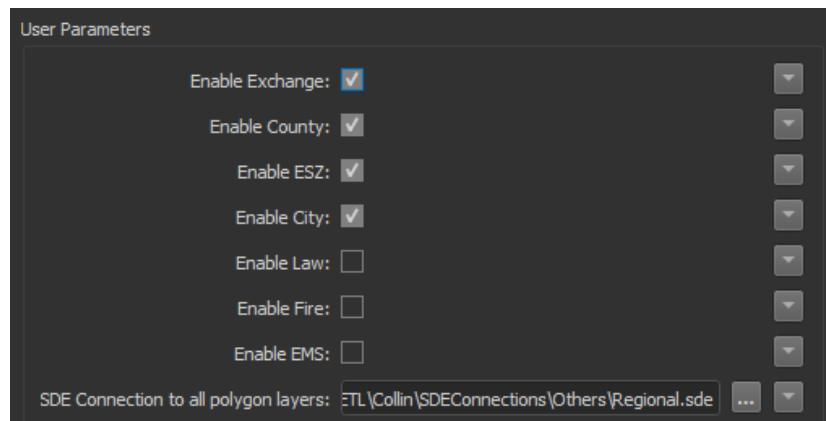
# Beyond an ETL - Examples

- Calculating NGUIDs (globally unique identifiers) & Email Notifications
  - Check for NGUIDs and calculate them if necessary
  - Use HTML Formatter and Emailer to send email notifications



# Quick tips

- Document your tools
- Give unique names to [transformers](#)
- Avoid hardcoded values
  - Use [User Parameters](#) as much as possible





## Resources

- [Safe.com](#)
  - Tutorials & Training
  - [FME Community](#)
  - Official [documentation](#)
  - Links to [FME partners](#)
  - [FME Hub](#)
- Regular Expressions/Regex Editor:
  - [Regix101](#)
  - [Regular-Expression.info](#)
- Webinars & Conferences
  - Look for FME/ETL contents!



# Questions or Comments?

## Contact Info

Bruno Blanco  
9-1-1 GIS Specialist  
[BBlanco@nct911.org](mailto:BBlanco@nct911.org) or  
[911GISTeam@nct911.org](mailto:911GISTeam@nct911.org)  
817-980-2586

