

INTRO TO GIS<sub>c</sub>

---

# SPATIAL JOINS

# AGENDA

1. Follow-up
2. GISc & Public Policy
3. Modifying Strings
4. Projecting Data
5. Spatial Joins
6. Table Joins

# 1 FOLLOW-UP

# 2 GIS<sub>c</sub> & PUBLIC POLICY

# 3 MODIFYING STRINGS

# MODIFYING STRINGS

In order to combine data sources, we sometimes need to modify identification variables or identify coordinate data that may not map correctly.

# POTENTIAL ISSUES

- ▶ ID variables are string and need to be numeric

ID	X	Y
01	-90.236560	38.637241
02	-90.236799	38.636550
03	-90.237290	38.636661
04	-90.238154	38.636829
05	-90.237682	38.636735
06	-90.238942	38.636996
07	-90.239924	38.637206
08	-90.241313	38.637499
09	-90.236564	38.636492
10	-90.236062	38.636354

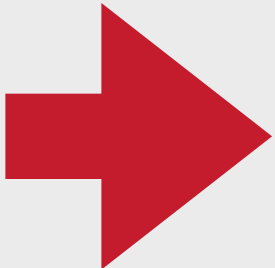
### 3. MODIFYING STRINGS

---

# POTENTIAL ISSUES

`destring` oldVar, generate(newVar)

- `destring` ID, generate(idNum)  
ID: all characters numeric;  
idNum generated as byte



ID	idNum
01	1
02	2
03	3
04	4
05	5
06	6
07	7
08	8
09	9
10	10



# POTENTIAL ISSUES

- ▶ ID variables are string and need to be numeric
- ▶ ID variables are numeric and need to be string

ID	X	Y
1	-90.236560	38.637241
2	-90.236799	38.636550
3	-90.237290	38.636661
4	-90.238154	38.636829
5	-90.237682	38.636735
6	-90.238942	38.636996
7	-90.239924	38.637206
8	-90.241313	38.637499
9	-90.236564	38.636492
10	-90.236062	38.636354

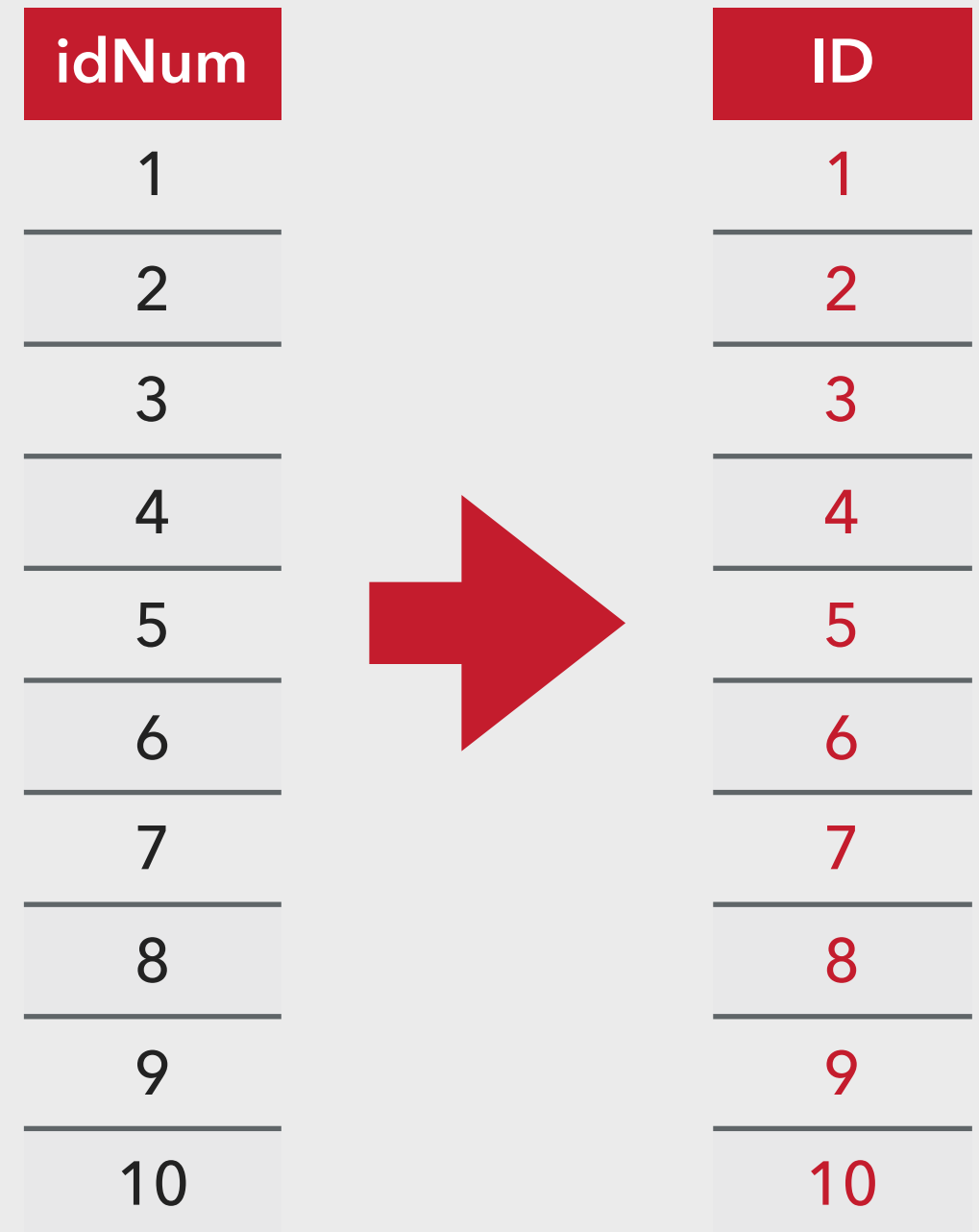
### 3. MODIFYING STRINGS

---

# POTENTIAL ISSUES

`generate` newVar = string(oldVar)

- `generate` ID = string(idNum)



### 3. MODIFYING STRINGS

---

# POTENTIAL ISSUES

`generate` newVar = string(oldVar, *fmt*)

- `generate` ID = string(idNum, "%02.0f")

idNum	ID
1	01
2	02
3	03
4	04
5	05
6	06
7	07
8	08
9	09
10	10

### 3. MODIFYING STRINGS

---

# POTENTIAL ISSUES

`generate` newVar = string(oldVar, *fmt*)

- `generate` ID = string(idNum, "%03.0f")

idNum		ID
1		001
2		002
3		003
4		004
5		005
6		006
7		007
8		008
9		009
10		010

# POTENTIAL ISSUES

- ▶ ID variables are string and need to be numeric
- ▶ ID variables are numeric and need to be string
- ▶ Coordinate variables are string and need to be numeric

ID	X	Y
1	-90.236560	38.637241
2	-90.236799	38.636550
3	-90.237290	38.636661
4	-90.238154	38.636829
5	-90.237682	38.636735
6	-90.238942	38.636996
7	-90.239924	38.637206
8	-90.241313	38.637499
9	-90.236564	38.636492
10	-90.236062	38.636354

# POTENTIAL ISSUES

- ▶ ID variables are string and need to be numeric
- ▶ ID variables are numeric and need to be string
- ▶ Coordinate variables are string and need to be numeric
- ▶ Incomplete coordinate data

ID	X	Y
1	-90.236560	38.637241
2	-90.236799	38.636550
3	-90.237290	38.636661
4	-90.238154	38.636829
5	-90.237682	38.636735
6	-90	38
7	-90.239924	38.637206
8	0	0
9	-90.236564	38.636492
10	-90.236062	38.636354

# POTENTIAL ISSUES

`generate newVar = strlen(oldVar)`

- `generate xLeng = strlen(X)`

X	xLeng
-90.236560	10
-90.236799	10
-90.237290	10
-90.238154	10
-90.237682	10
-90	3
-90.239924	10
0	1
-90.236564	10
-90.236062	10

# POTENTIAL ISSUES

generate newVar = strlen(oldVar)

- generate xLeng = strlen(X)
- replace X = "" if xLeng < 10

X	xLeng
-90.236560	-90.236560
-90.236799	-90.236799
-90.237290	-90.237290
-90.238154	-90.238154
-90.237682	-90.237682
-90	
-90.239924	-90.239924
0	
-90.236564	-90.236564
-90.236062	-90.236062



# POTENTIAL ISSUES

generate newVar = strlen(oldVar)

- generate xLeng = strlen(X)
- drop if xLeng < 10

X	xLeng
-90.236560	-90.236560
-90.236799	-90.236799
-90.237290	-90.237290
-90.238154	-90.238154
-90.237682	-90.237682
-90	-90.239924
-90.239924	-90.236564
0	-90.236062
-90.236564	
-90.236062	

# POTENTIAL ISSUES

- ▶ ID variables are string and need to be numeric
- ▶ ID variables are numeric and need to be string
- ▶ Coordinate variables are string and need to be numeric
- ▶ Incomplete coordinate data

# POTENTIAL ISSUES

PP	TRACTCE	GEOID	NAME	NAMESAD	MTFCC	FUNCTAT	ALAND	AWATER	INTPTLAT	INTPTLON	Shape_Leng	Shape_Area	sqMiles	sqMiles2
	116100	29510116100	1161	Census Tract 1161	G5020	S	1563757	0	+38.6235369	-090.2538987	0.059353	0.000162	0.6	0.603695
	117400	29510117400	1174	Census Tract 1174	G5020	S	1146068	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
	126700	29510126700	1267	Census Tract 1267	G5020	S	2248811	399279	+38.6628818	-090.1929647	0.088248	0.000274	1.02	1.021542
	119102	29510119102	1191.02	Census Tract 1191.02	G5020	S	581750	0	+38.8420554	-090.2528839	0.040858	0.00008	0.22	0.224588
	126800	29510126800	1268	Census Tract 1268	G5020	S	2792747	1464	+38.6145349	-090.3089427	0.094091	0.000269	1.06	1.078714
	126900	29510126900	1269	Census Tract 1269	G5020	S	4336761	0	+38.6855021	-090.2563370	0.124737	0.000449	1.67	1.674235
	106100	29510106100	1061	Census Tract 1061	G5020	S	3214656	0	+38.7072590	-090.2391665	0.087821	0.000333	1.24	1.241035
	127000	29510127000	1270	Census Tract 1270	G5020	S	1053044	4453892	+38.7056278	-090.2219202	0.272379	0.001552	5.76	5.784806
	127400	29510127400	1274	Census Tract 1274	G5020	S	3313718	217474	+38.6201595	-090.2080528	0.114014	0.000385	1.36	1.383238
	103700	29510103700	1037	Census Tract 1037	G5020	S	910953	0	+38.6080393	-090.2918885	0.045385	0.000094	0.35	0.351676

### 3. MODIFYING STRINGS

# POTENTIAL ISSUES

The screenshot displays a GIS application interface. On the left, a 'Table' menu is open, showing options like 'Find and Replace...', 'Select By Attributes...', 'Clear Selection', 'Switch Selection', 'Select All', 'Add Field...', 'Turn All Fields On', 'Show Field Aliases', 'Arrange Tables', 'Restore Default Column Widths', 'Restore Default Field Order', 'Joins and Relates', 'Related Tables', 'Create Graph...', 'Add Table to Layout', 'Reload Cache', 'Print...', 'Reports', 'Export...', and 'Appearance...'. The 'Add Field...' option is highlighted. In the background, a table of census tracts is visible, with columns: TRACTCE, GEOID, NAME, NAMELSAD, MTFCC, FUNCSTAT, ALAND, AWATER, INTPTLAT, INTPTLON, Shape\_Leng, Shape\_Area, sqMiles, and sqMiles2. The table contains 10 rows of data. Overlaid on the table is an 'Add Field' dialog box. The dialog has a title bar with a close button. It contains a 'Name' field with the text 'geoidNum', a 'Type' dropdown menu set to 'Long Integer', and a 'Field Properties' section with a 'Precision' field set to '0'. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

TRACTCE	GEOID	NAME	NAMELSAD	MTFCC	FUNCSTAT	ALAND	AWATER	INTPTLAT	INTPTLON	Shape_Leng	Shape_Area	sqMiles	sqMiles2
118100	29510118100	1181	Census Tract 1181	G5020	S	1563757	0	+38.6235569	-090.2538987	0.059353	0.000162	0.6	0.603695
117400	29510117400	1174	Census Tract 1174	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
128700	29510128700	1287	Census Tract 1287	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
119102	29510119102	1191.02	Census Tract 1191.02	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
126800	29510126800	1268	Census Tract 1268	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
126900	29510126900	1269	Census Tract 1269	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
108100	29510108100	1081	Census Tract 1081	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
127000	29510127000	1270	Census Tract 1270	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
127400	29510127400	1274	Census Tract 1274	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455
103700	29510103700	1037	Census Tract 1037	G5020	S	1146088	0	+38.6095244	-090.2373976	0.04886	0.000119	0.44	0.442455

10 of 105 Selected)

**Add Field**

Name:

Type:

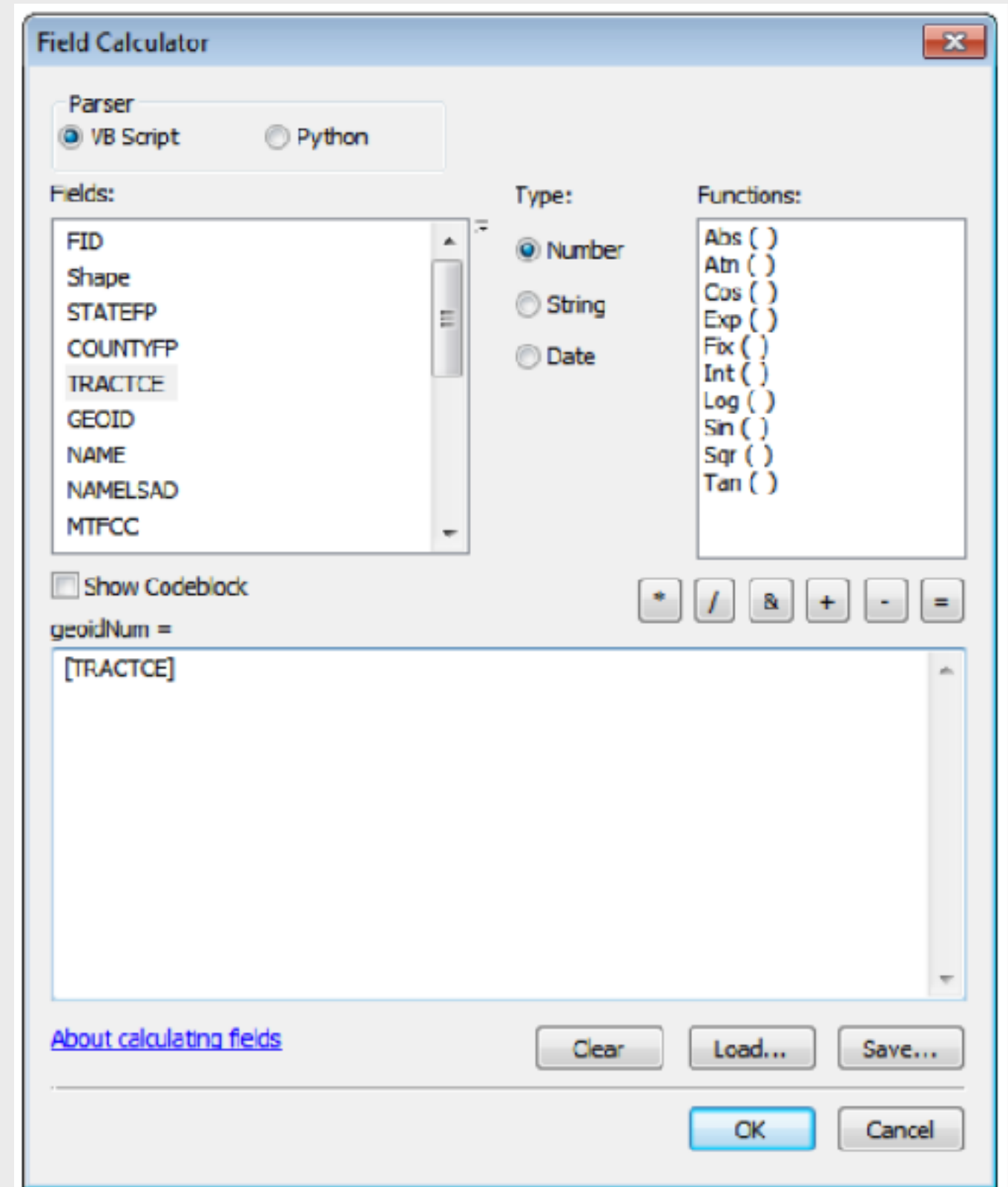
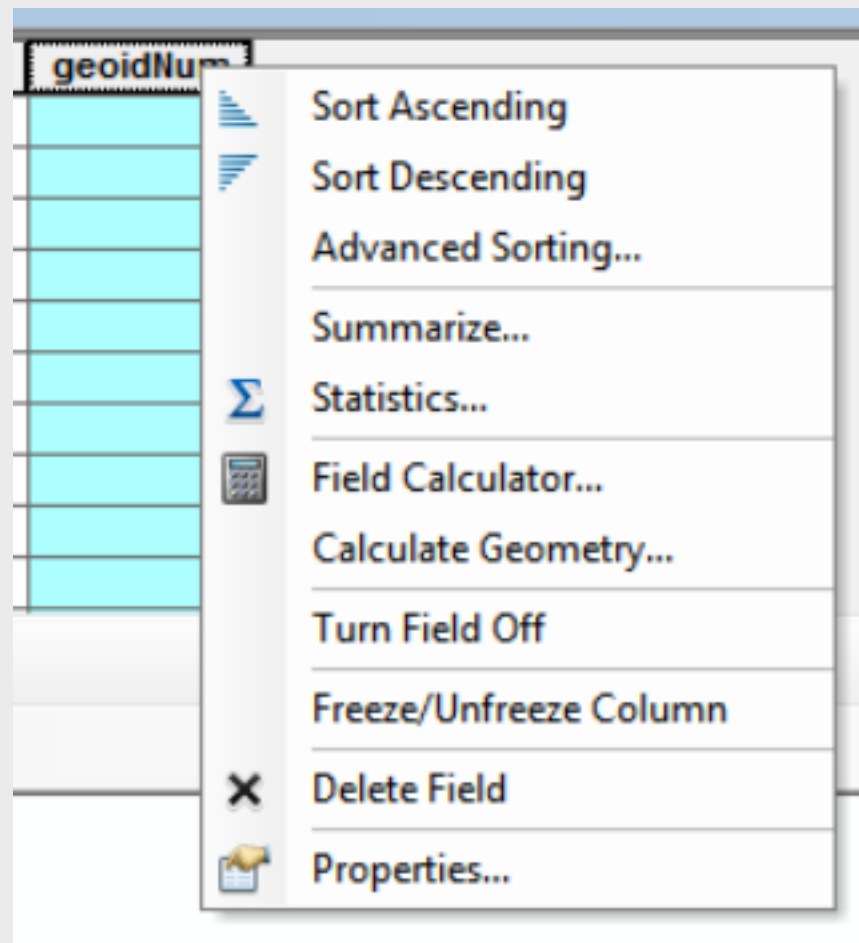
Field Properties

Precision

OK Cancel

### 3. MODIFYING STRINGS

# POTENTIAL ISSUES



# 4 PROJECTING DATA

# PROJECTING DATA

When your **tabular data** have **x,y coordinates** (decimal degrees or UTM coordinates), they can be **projected** into your map document.

# PROJECTING DATA

A word of caution - ArcGIS struggles with **csv** files when importing data in ArcMap. You will likely need to save csv files as **xlsx** files using Microsoft Excel. This applies to table joins as well.

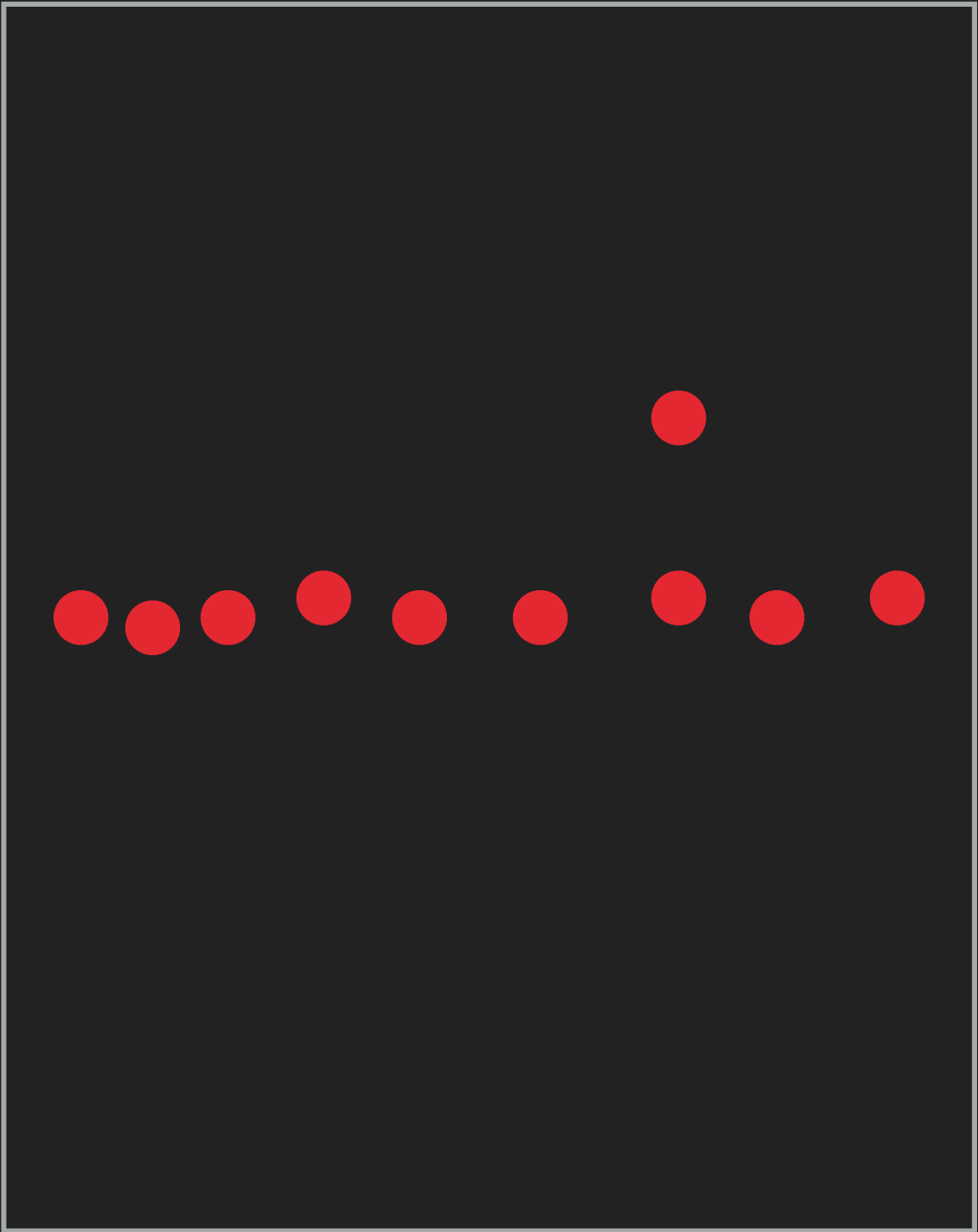


# PROJECTING DATA

Remember that the x coordinate is longitude and the y coordinate is latitude if your data are in decimal degrees.

# TRANSFORMING TABULAR DATA TO SHAPEFILES

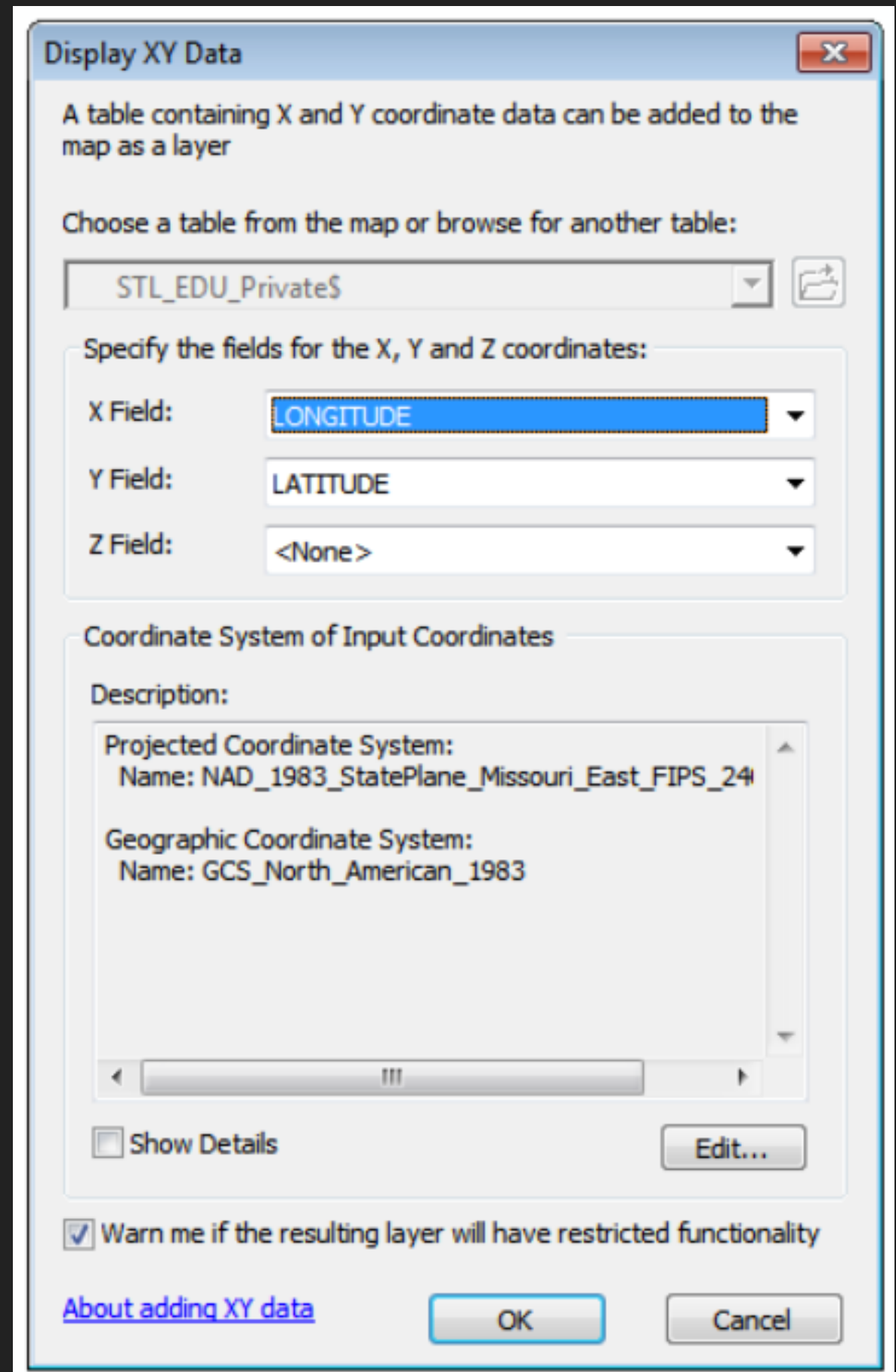
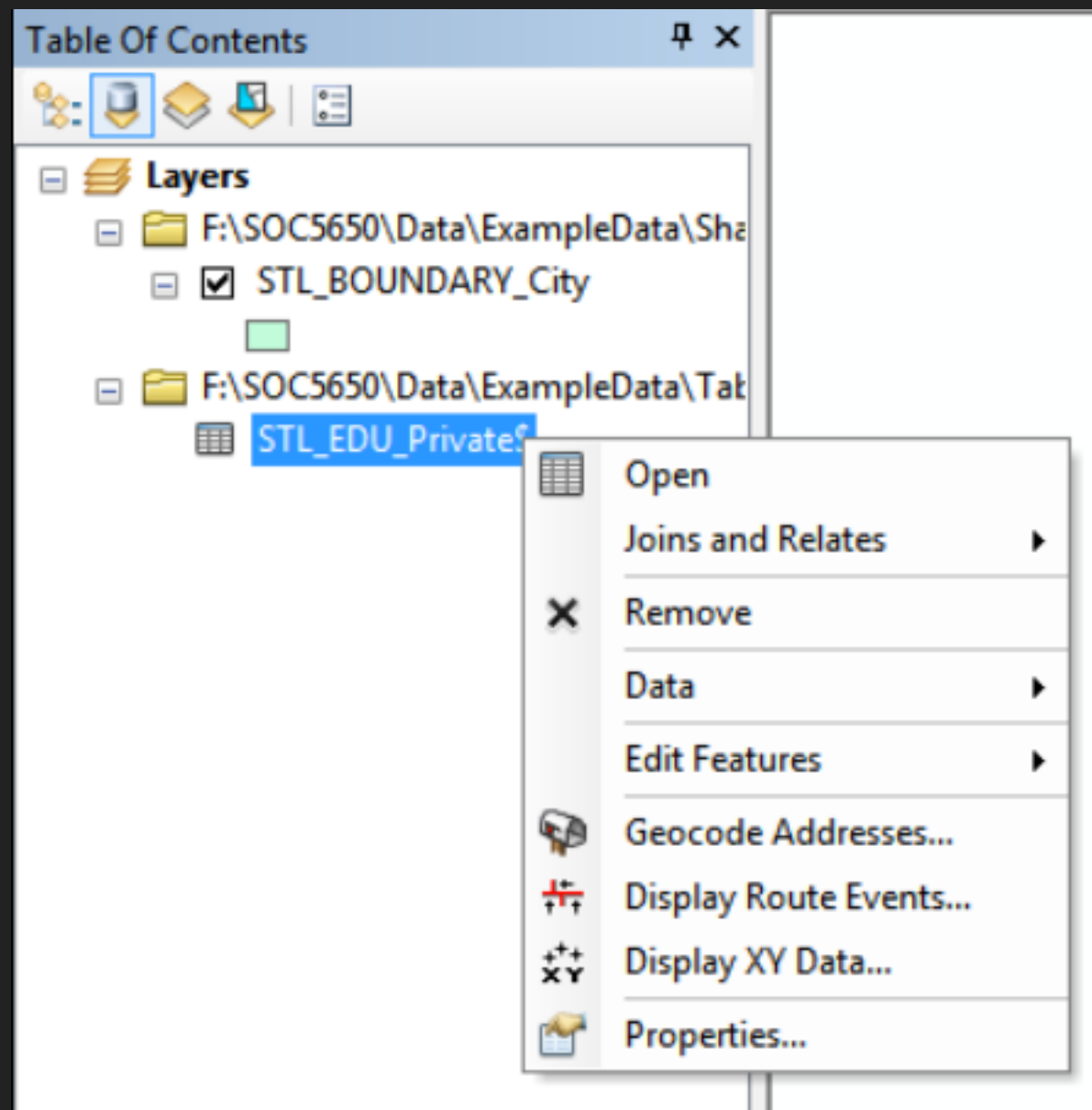
ID	X	Y
1	-90.236560	38.637241
2	-90.236799	38.636550
3	-90.237290	38.636661
4	-90.238154	38.636829
5	-90.237682	38.636735
6	-90.238942	38.636996
7	-90.239924	38.637206
8	-90.241313	38.637499
9	-90.236564	38.636492
10	-90.236062	38.636354



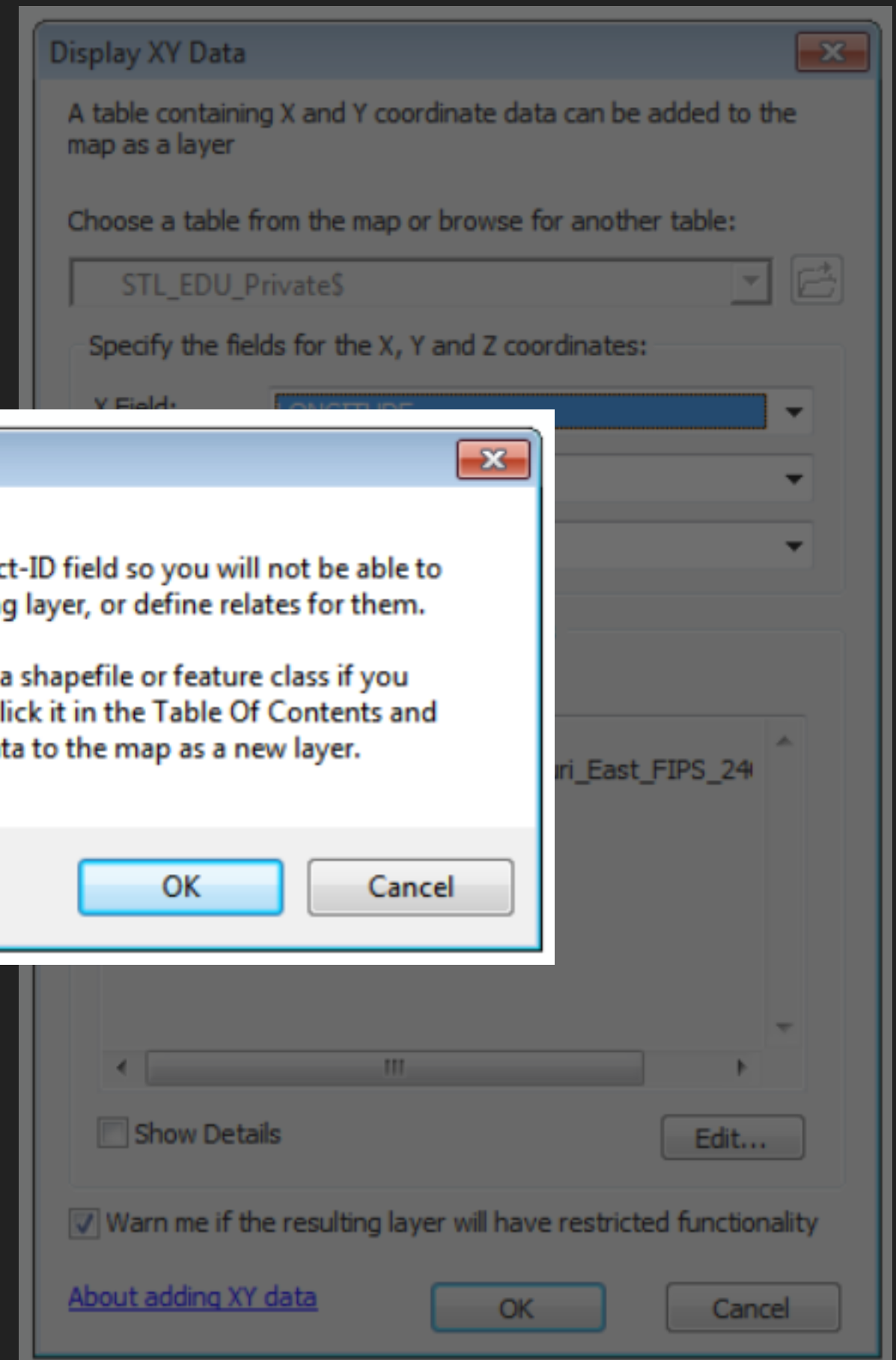
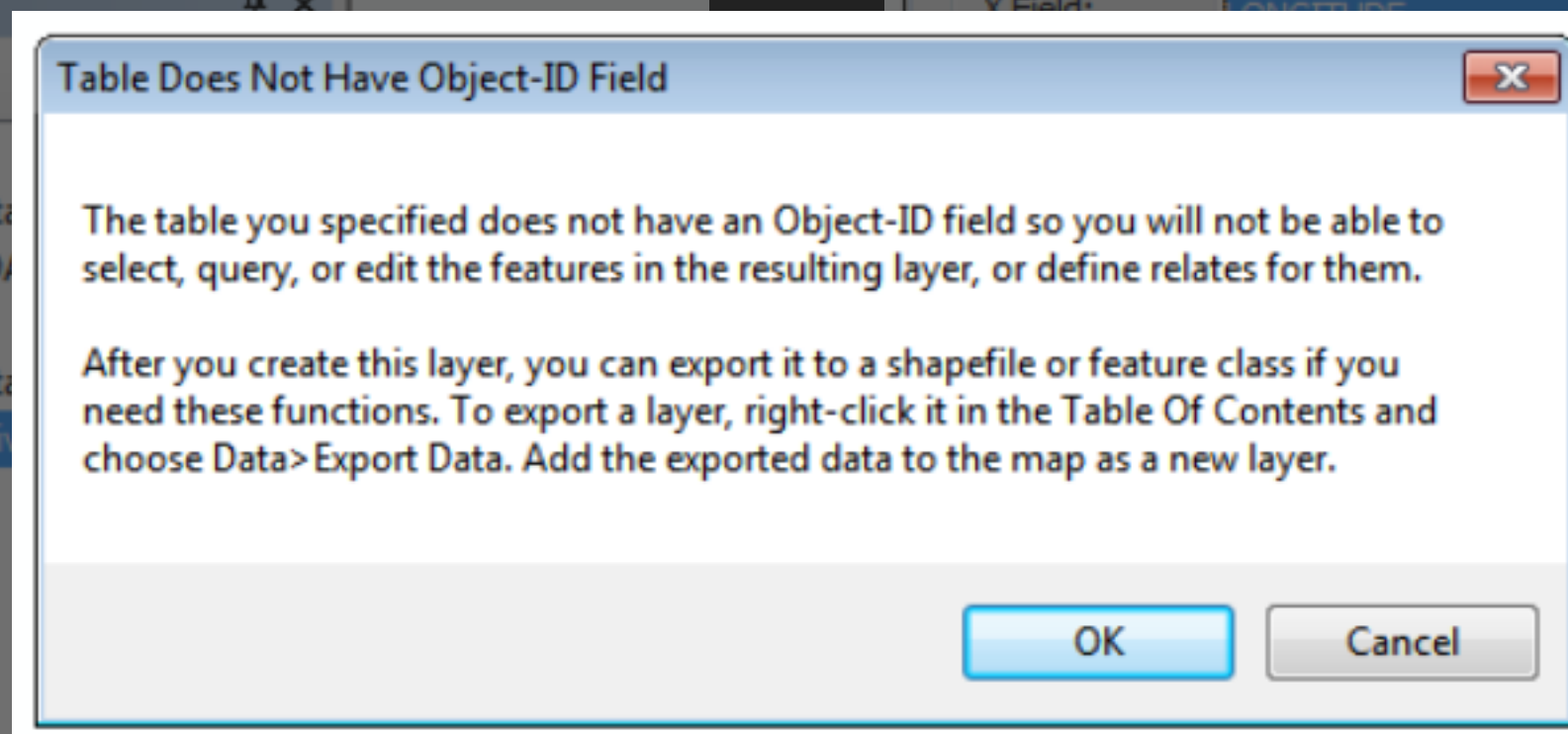
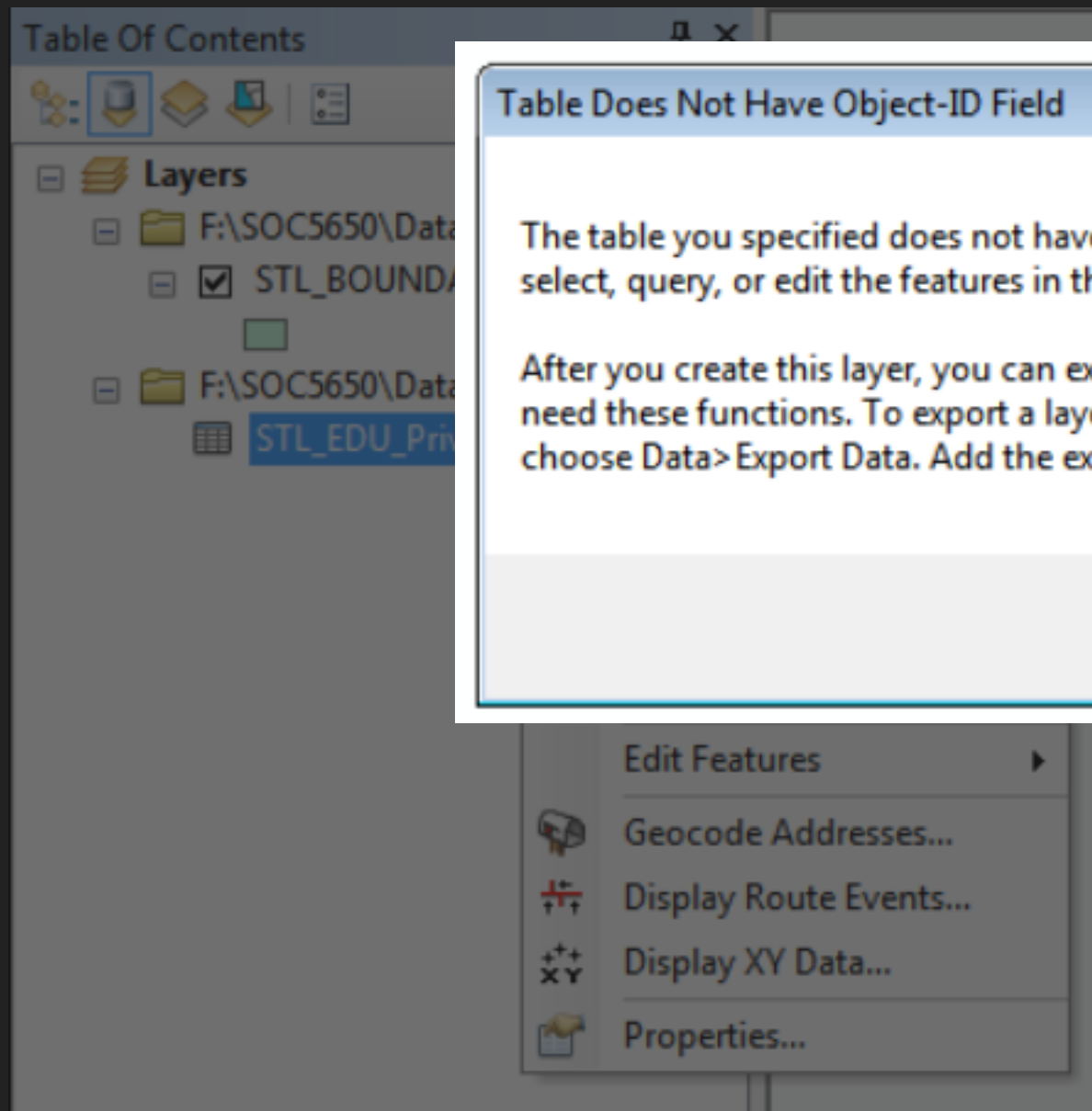
# PROJECTING DATA

We have the locations of all private schools stored in a `csv` file with `latitude` and `longitude` coordinates. We want to `convert` them from tabular to geometric data.

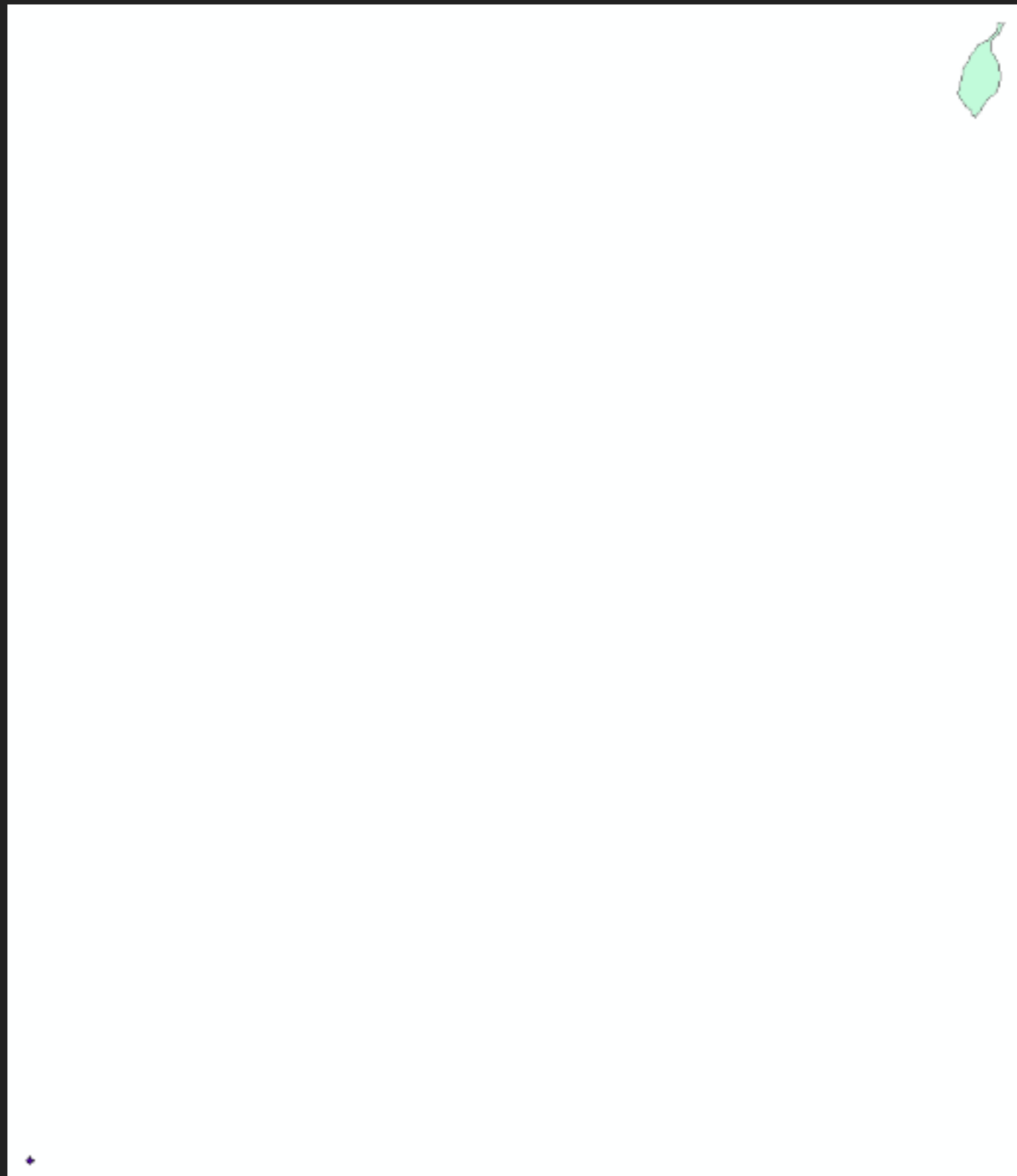
# DISPLAY DATA



# DISPLAY DATA



# CHECKING DATA



**Display XY Data** [X]

A table containing X and Y coordinate data can be added to the map as a layer

Choose a table from the map or browse for another table:

STL\_EDU\_Private\$ [Browse]

Specify the fields for the X, Y and Z coordinates:

X Field: LONGITUDE [v]

Y Field: LATITUDE [v]

Z Field: <None> [v]

**Coordinate System of Input Coordinates**

Description:

Projected Coordinate System:  
Name: NAD\_1983\_StatePlane\_Missouri\_East\_FIPS\_241

Geographic Coordinate System:  
Name: GCS\_North\_American\_1983

[Show Details] [Edit...]

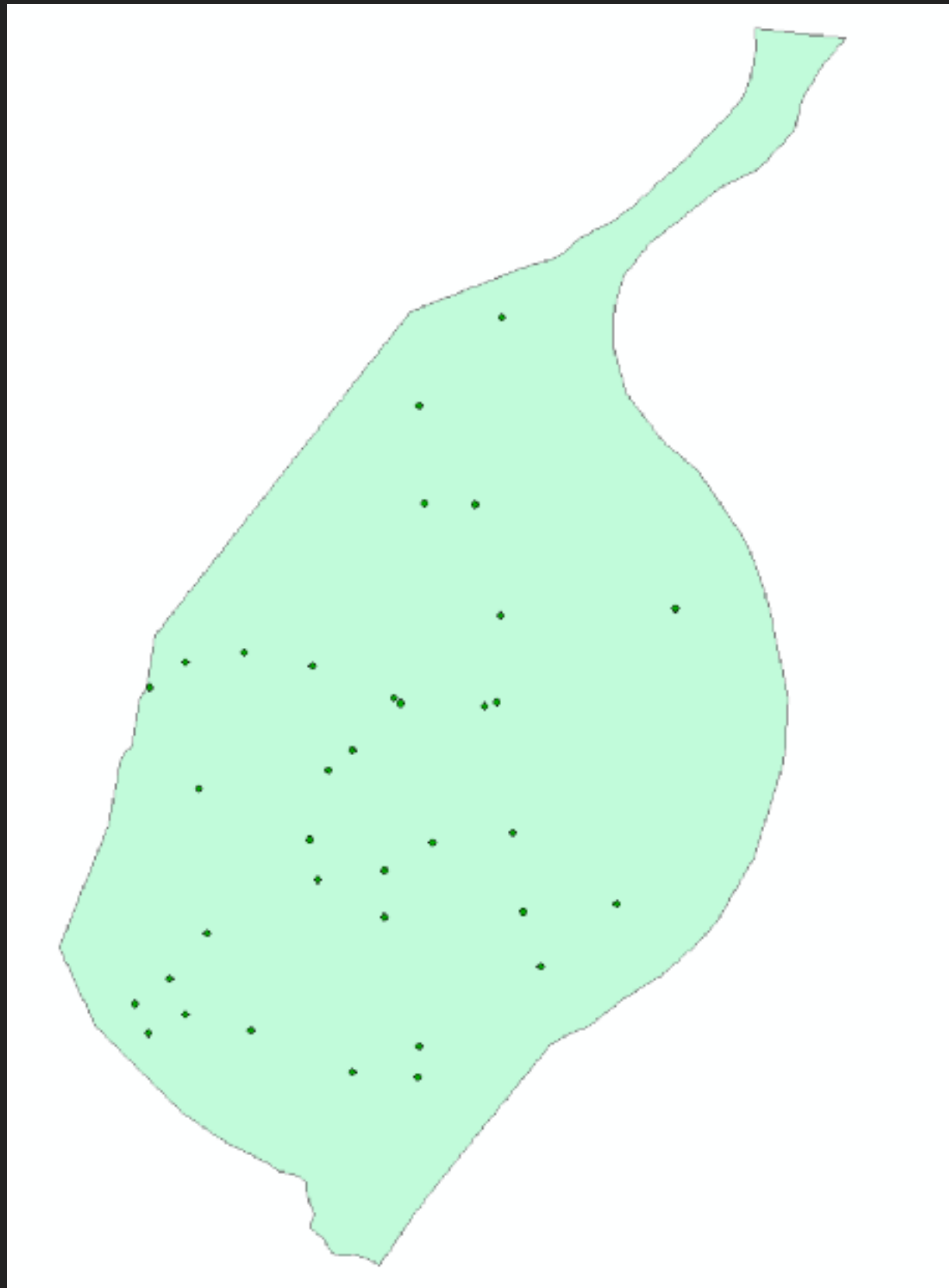
☒ Warn me if the resulting layer will have restricted functionality

[About adding XY data](#) [OK] [Cancel]

# CHECK DATA



# CHECKING DATA



Display XY Data

A table containing X and Y coordinate data can be added to the map as a layer

Choose a table from the map or browse for another table:

STL\_EDU\_Private\$

Specify the fields for the X, Y and Z coordinates:

X Field: LONGITUDE

Y Field: LATITUDE

Z Field: <None>

Coordinate System of Input Coordinates

Description:

Geographic Coordinate System:  
Name: GCS\_North\_American\_1983

☐ Show Details

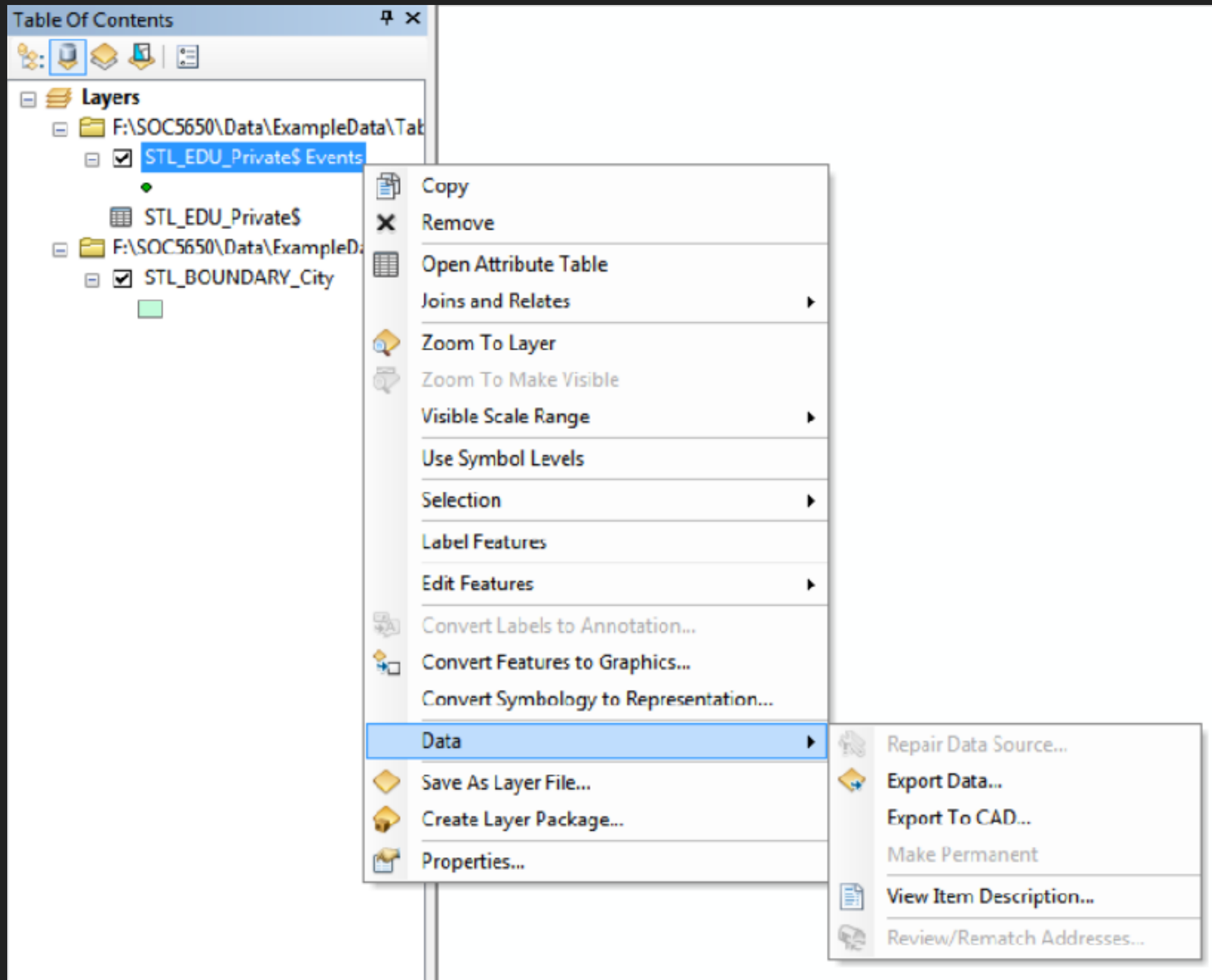
☒ Warn me if the resulting layer will have restricted functionality

[About adding XY data](#)

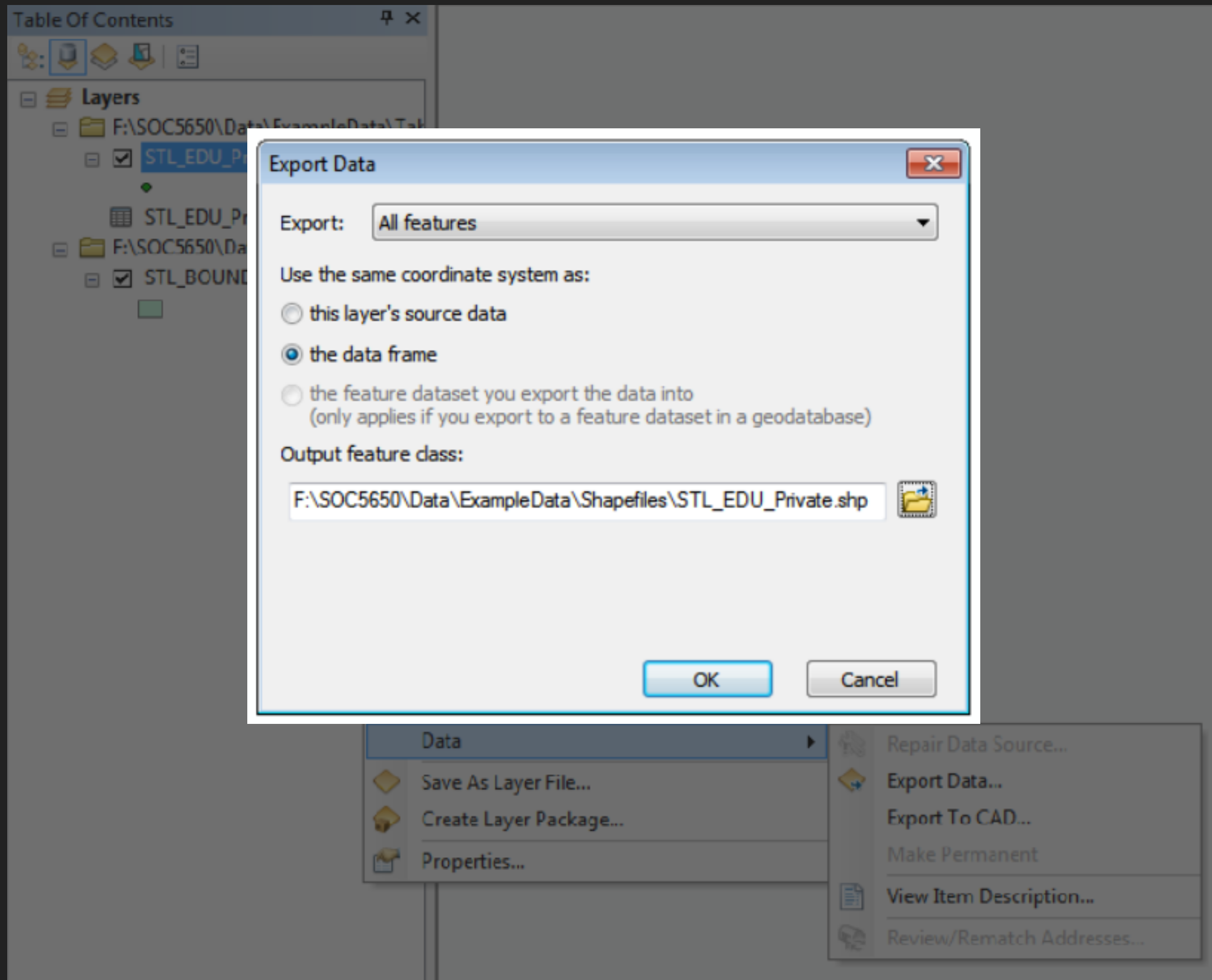
OK Cancel



# EXPORTING DATA



# EXPORTING DATA



# 5 SPATIAL JOINS

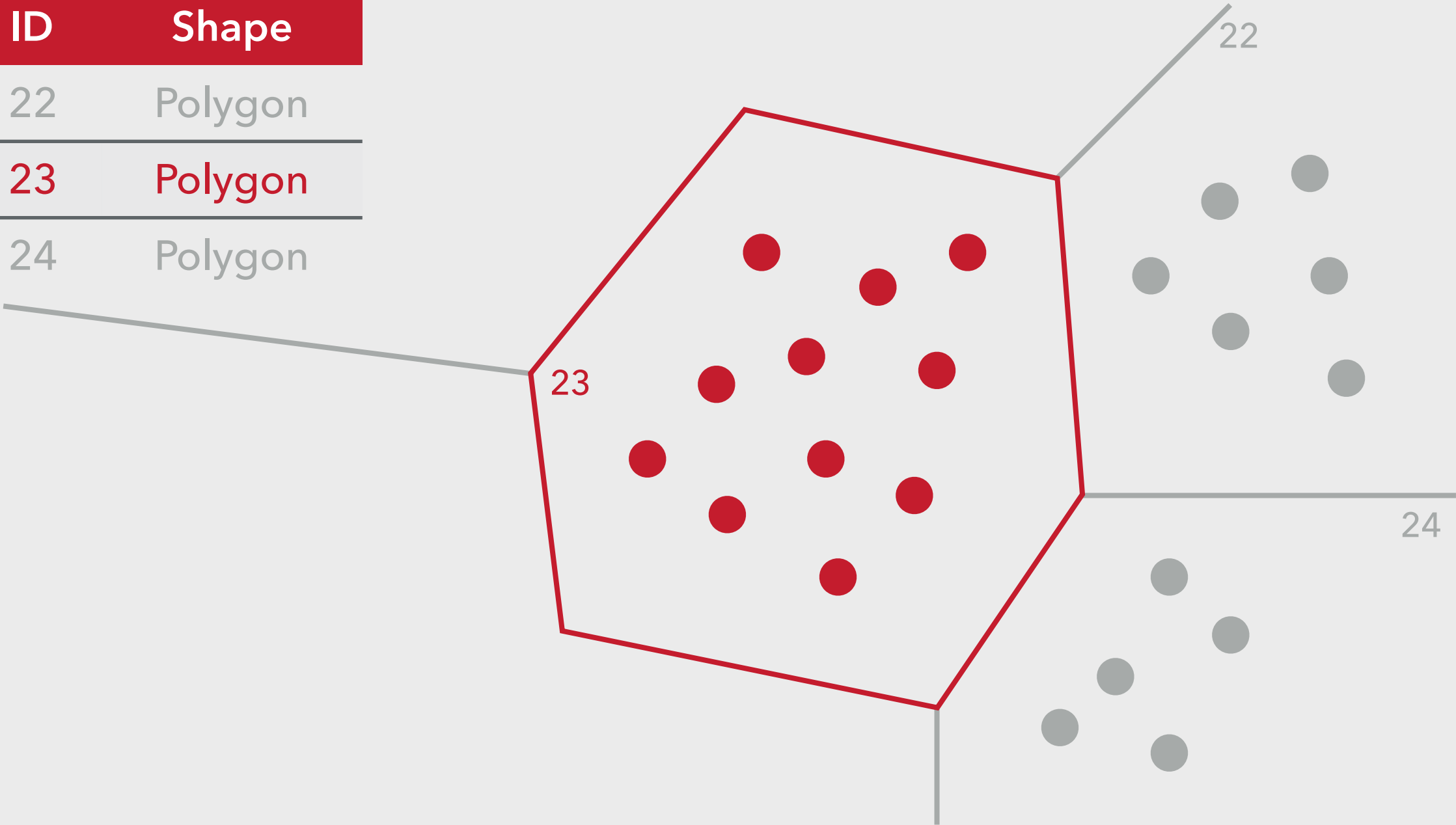
# POINTS TO POLYGONS

When the **target** layer consists of **polygons** and the **reference** layer consists of **point** data, you can **get a count** or average of point values within each polygon.

# POINTS TO POLYGONS

Target Layer

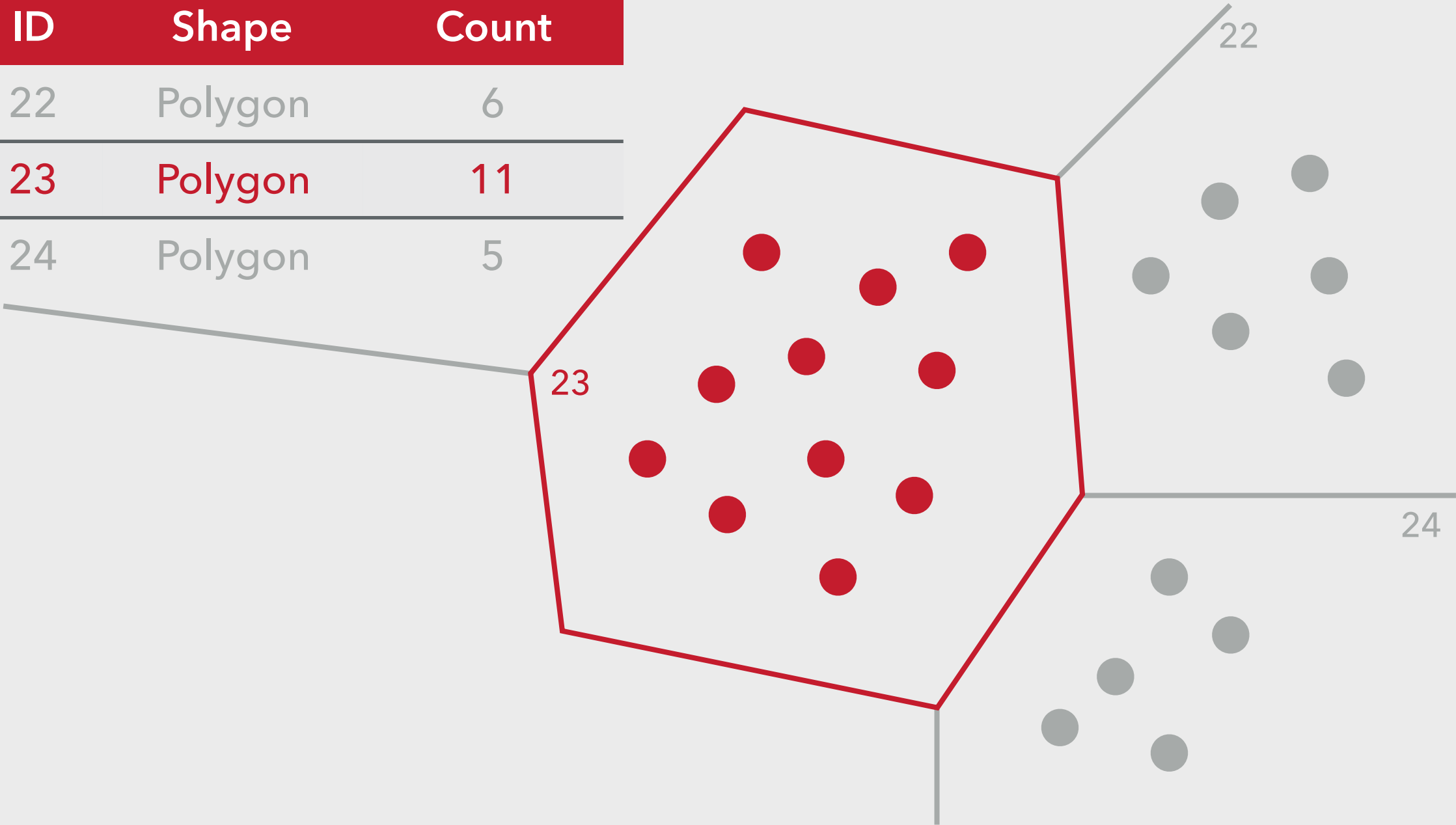
ID	Shape
22	Polygon
23	Polygon
24	Polygon



# POINTS TO POLYGONS

Target Layer

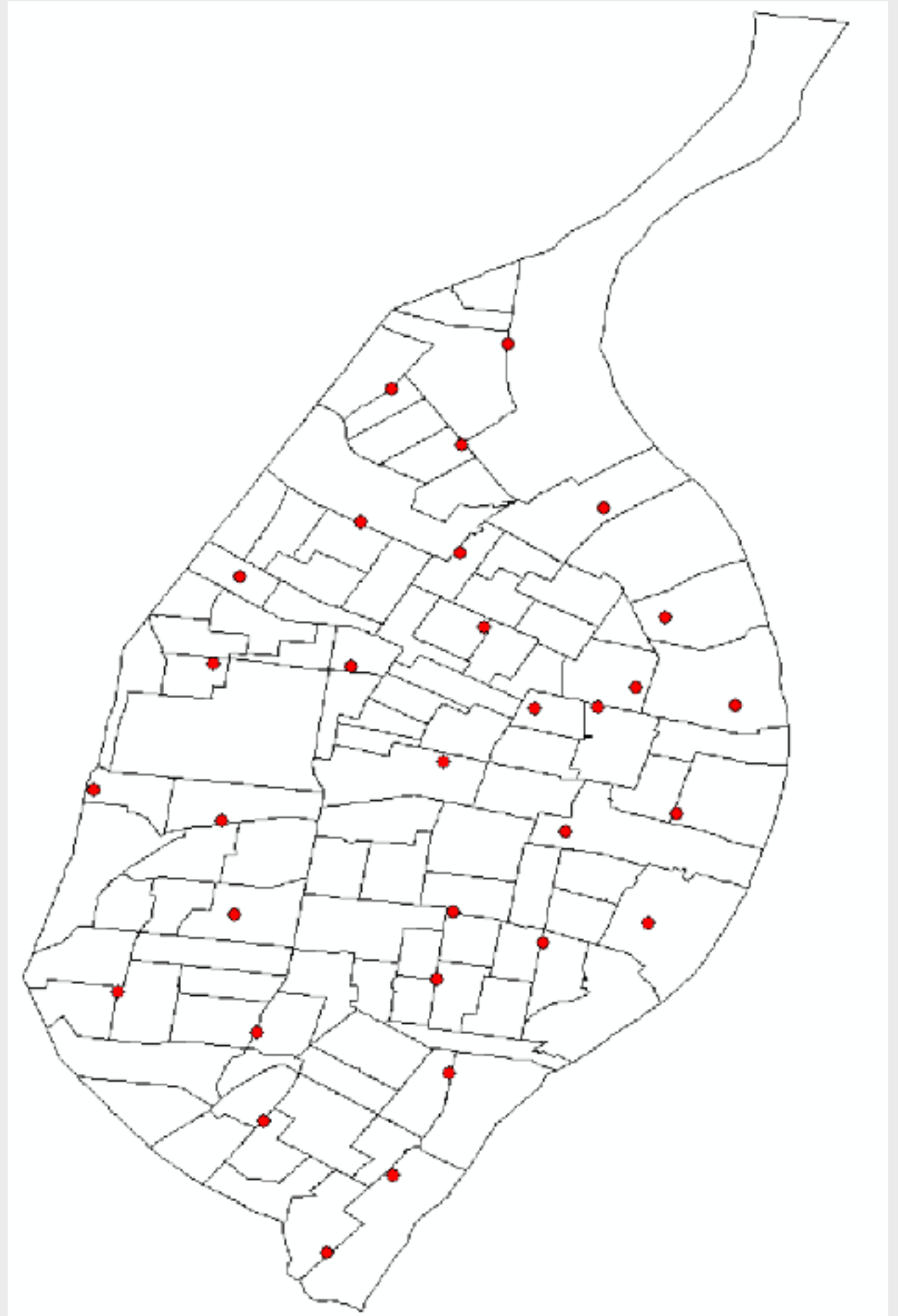
ID	Shape	Count
22	Polygon	6
23	Polygon	11
24	Polygon	5



# POINTS TO POLYGONS

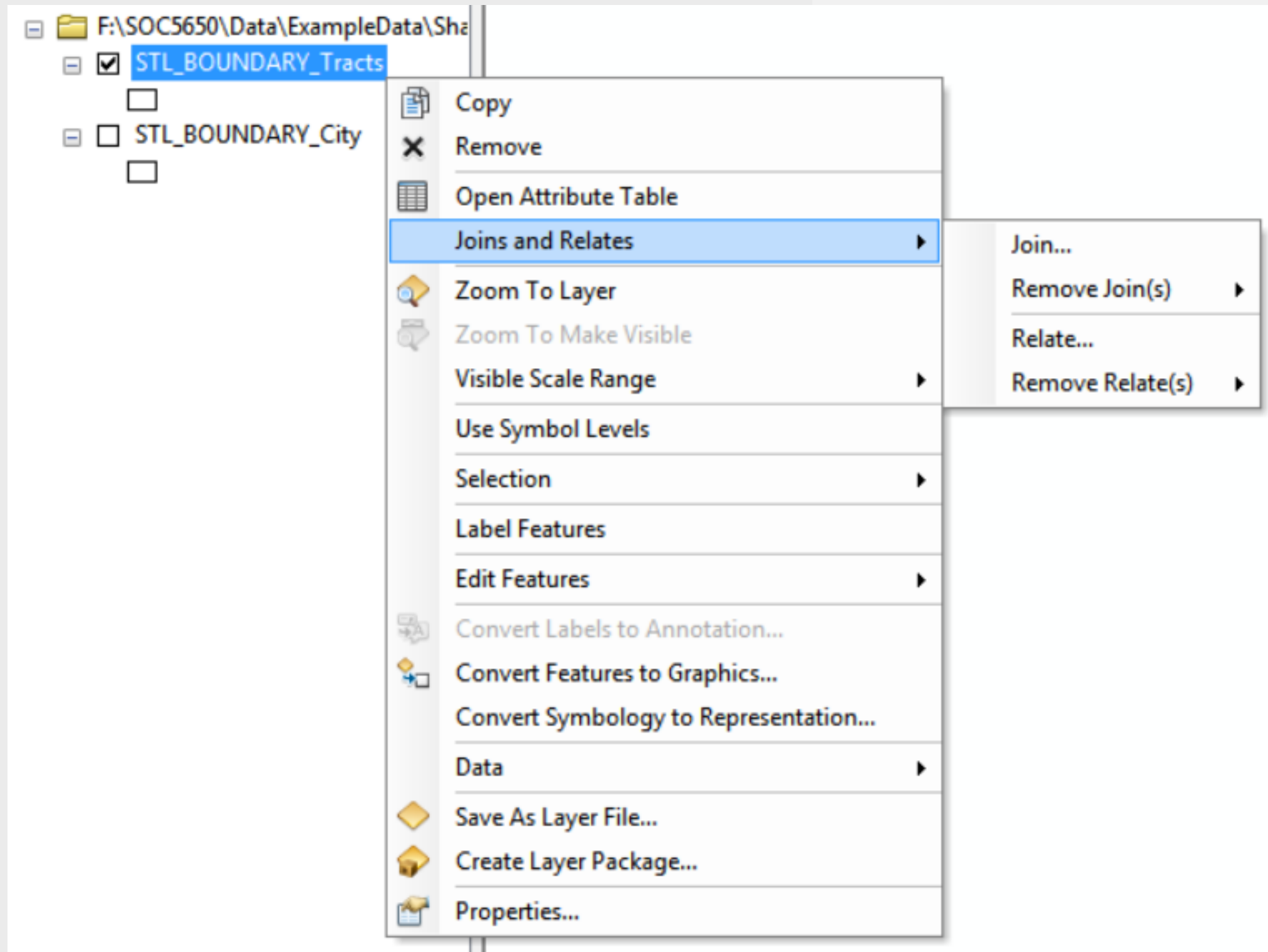
How many fire stations in St. Louis lie within each Census Tract?

# POINTS TO POLYGONS

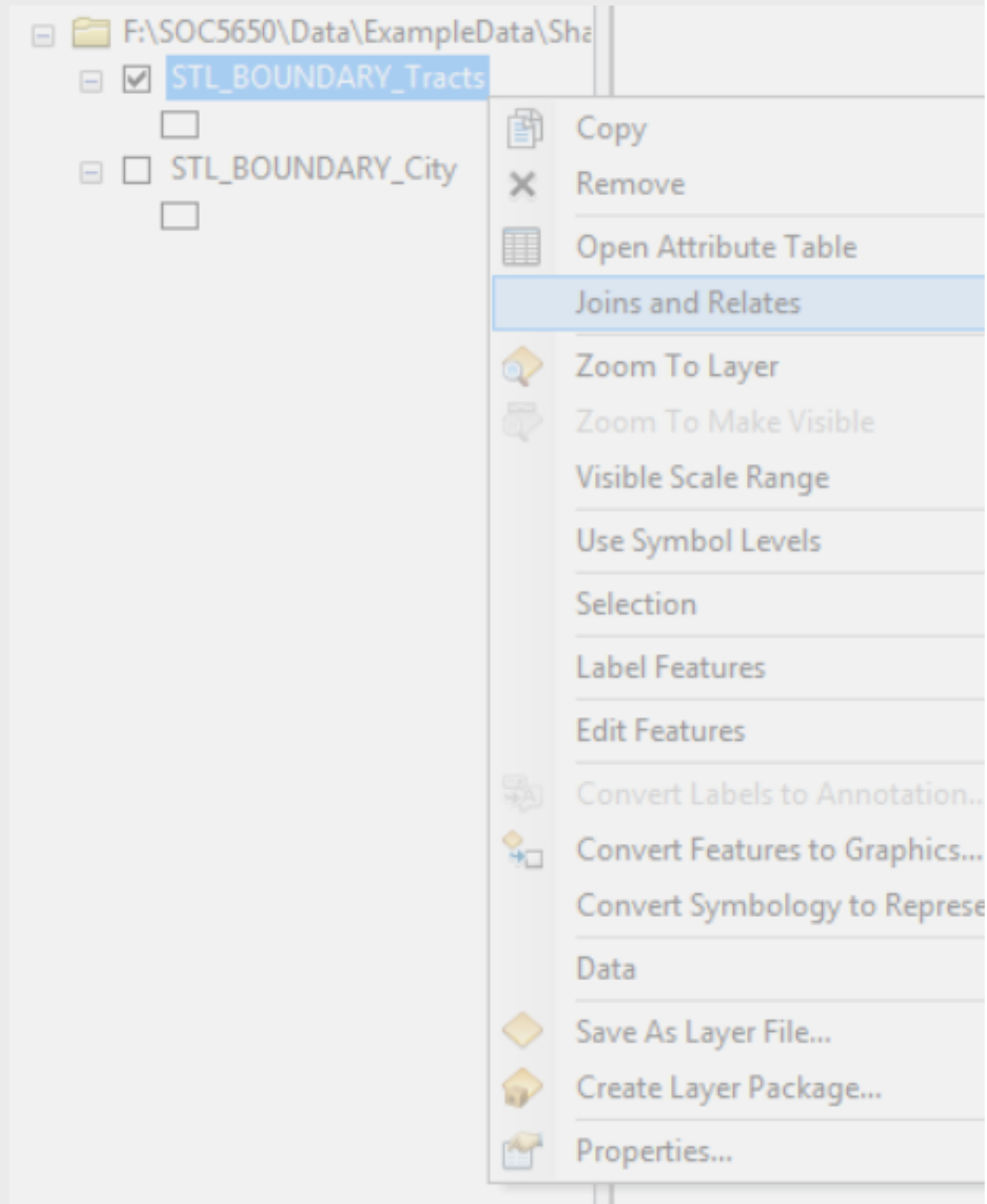




# POINTS TO POLYGONS



# POINTS TO POLYGONS




**Join Data**

Join lets you append additional data to this layer's attribute table so you can, for example, symbolize the layer's features using this data.

What do you want to join to this layer?

Join data from another layer based on spatial location

1. Choose the layer to join to this layer, or load spatial data from disk:

 STL\_PUBLICSAFETY\_FireStations

2. You are joining: Points to Polygons

Select a join feature class above. You will be given different options based on geometry types of the source feature class and the join feature class.

☒ Each polygon will be given a summary of the numeric attributes of the points that fall inside it, and a count field showing how many points fall inside it.

How do you want the attributes to be summarized?

☐ Average ☐ Minimum ☐ Standard Deviation

☒ Sum ☐ Maximum ☐ Variance

☐ Each polygon will be given all the attributes of the point that is closest to its boundary, and a distance field showing how close the point is (in the units of the target layer).

Note: A point falling inside a polygon is treated as being closest to the polygon, (i.e. a distance of 0).

3. The result of the join will be saved into a new layer.

Specify output shapefile or feature class for this new layer:

F:\SOC5650\Data\ExampleData\PublicSafety.gdb\STL\_JOIN

[About joining data](#)

OK Cancel

# POLYGONS TO POINTS

When the **target** layer consists of **points** and the **reference** layer consists of **polygons** data, you can **assign the attributes** of the polygons to the points that lie within them.

# POLYGONS TO POINTS

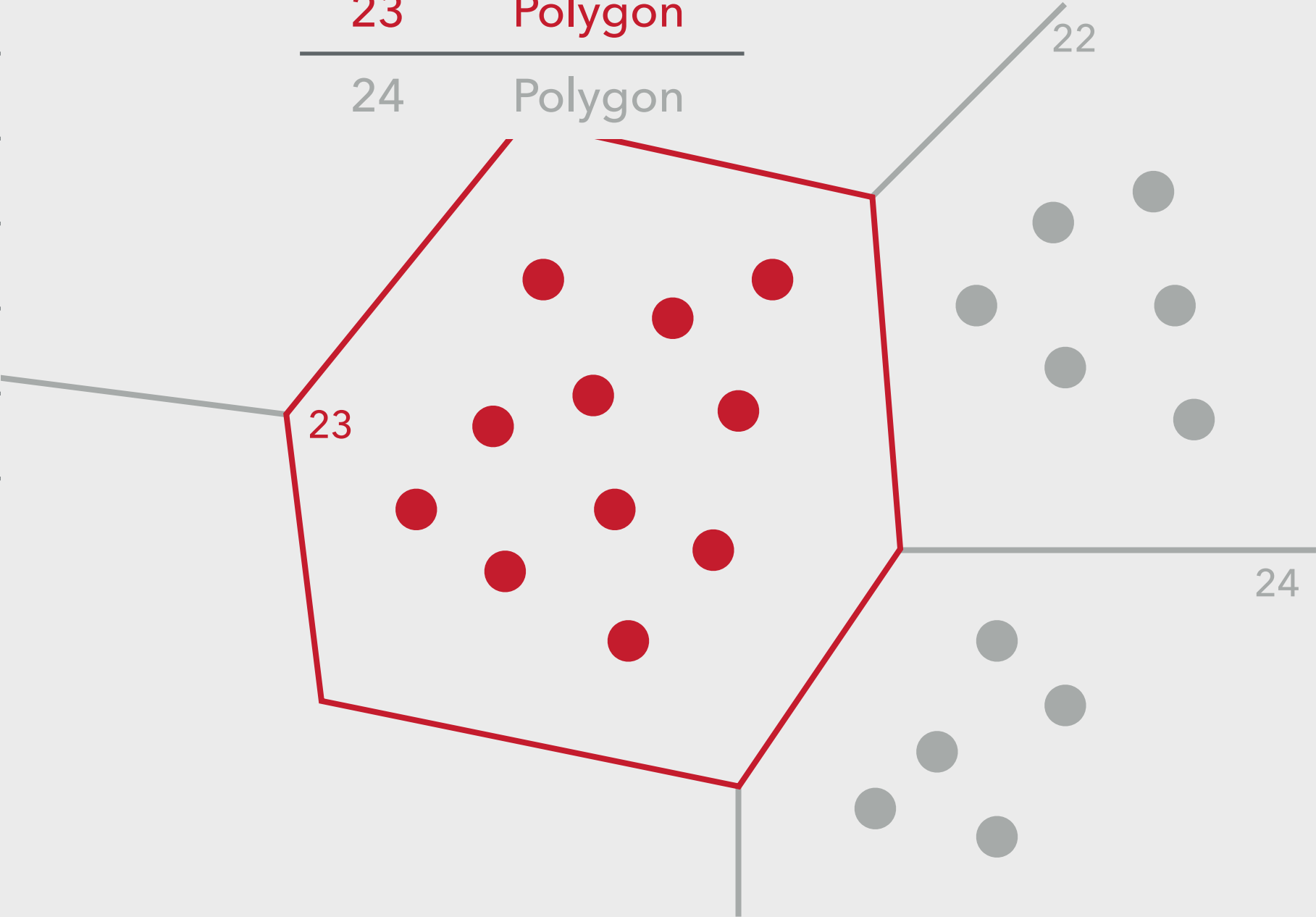
Target Layer

ID	Shape
104	Point
105	Point
106	Point
105	Point
106	Point
107	Point
108	Point

109	Point
-----	-------

Reference Layer

ID	Shape
22	Polygon
23	Polygon
24	Polygon



# POLYGONS TO POINTS

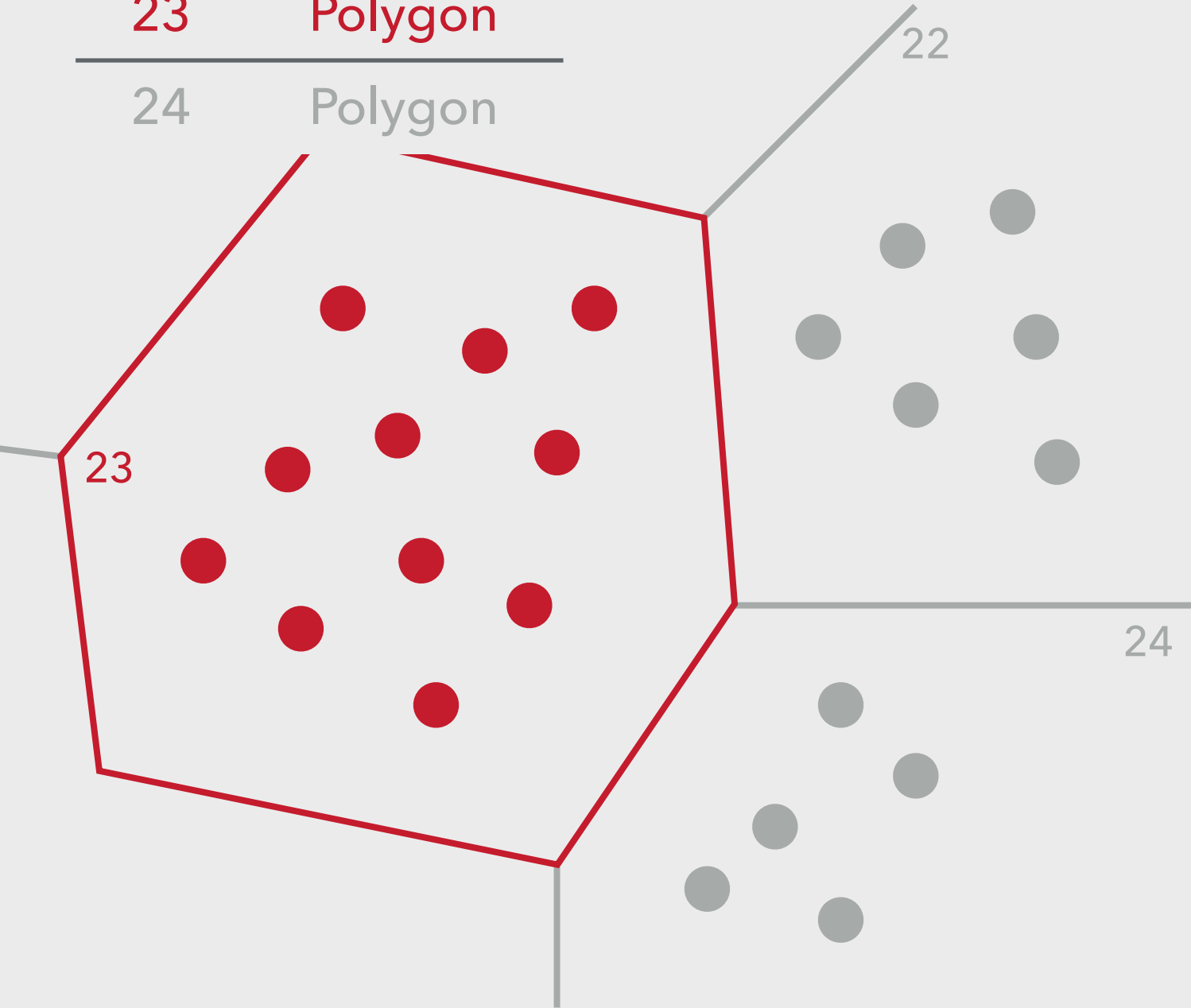
Target Layer

ID	Shape	polyID
104	Point	22
105	Point	23
106	Point	24
105	Point	23
106	Point	23
107	Point	23
108	Point	23

109	Point	24
-----	-------	----

Reference Layer

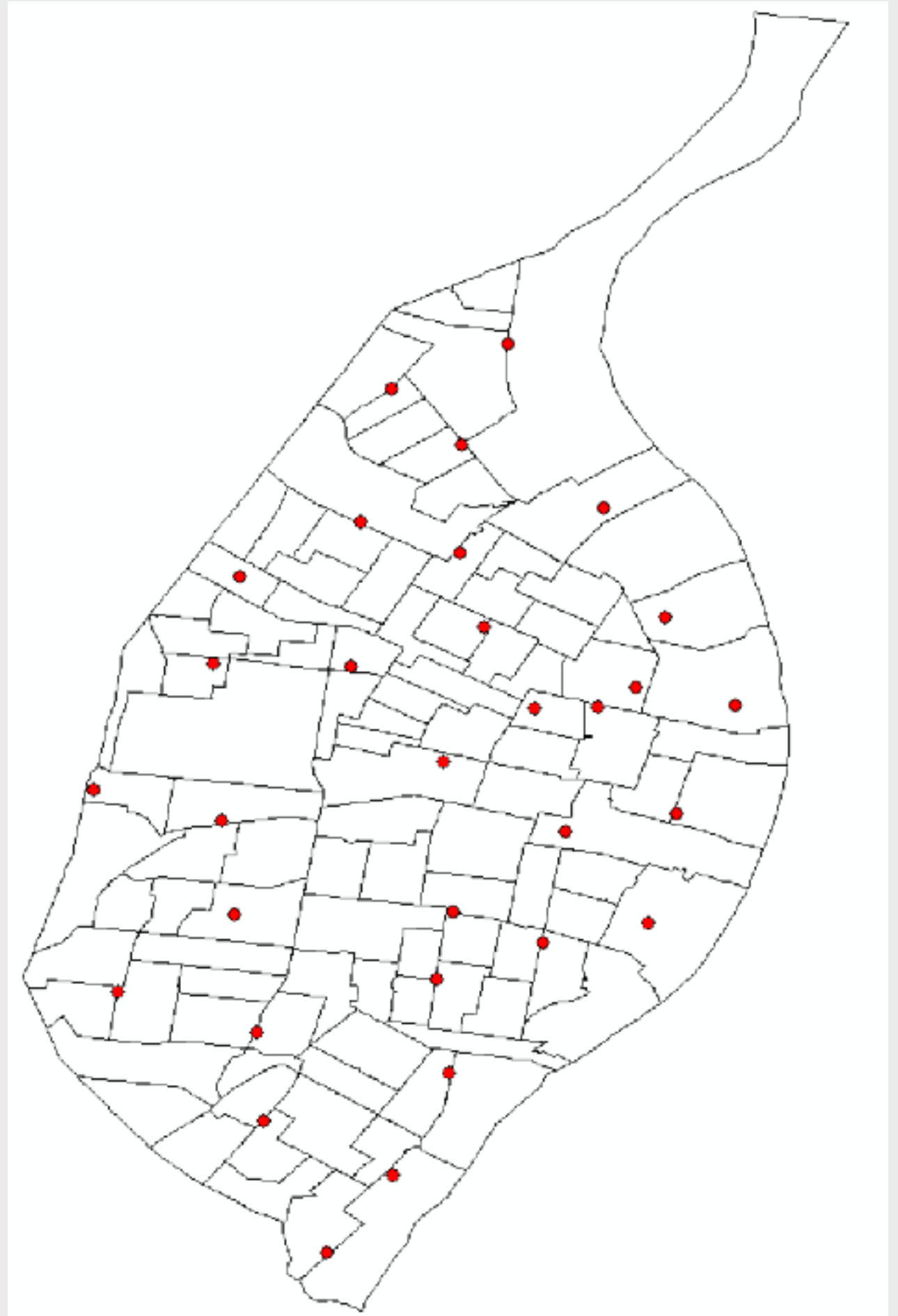
ID	Shape
22	Polygon
23	Polygon
24	Polygon



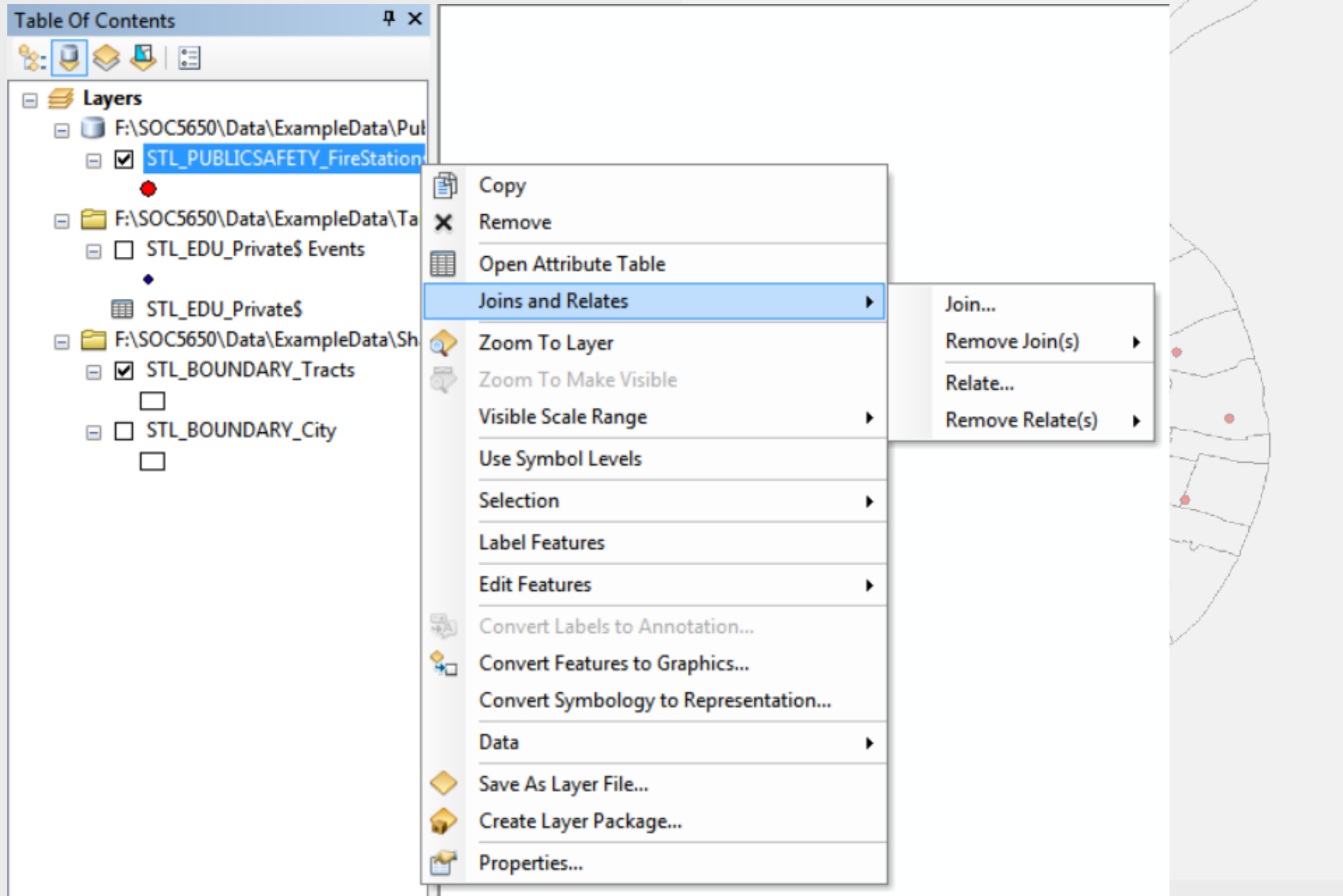
# POLYGONS TO POINTS

What Census Tract does each fire station lie within?

# POLYGONS TO POINTS

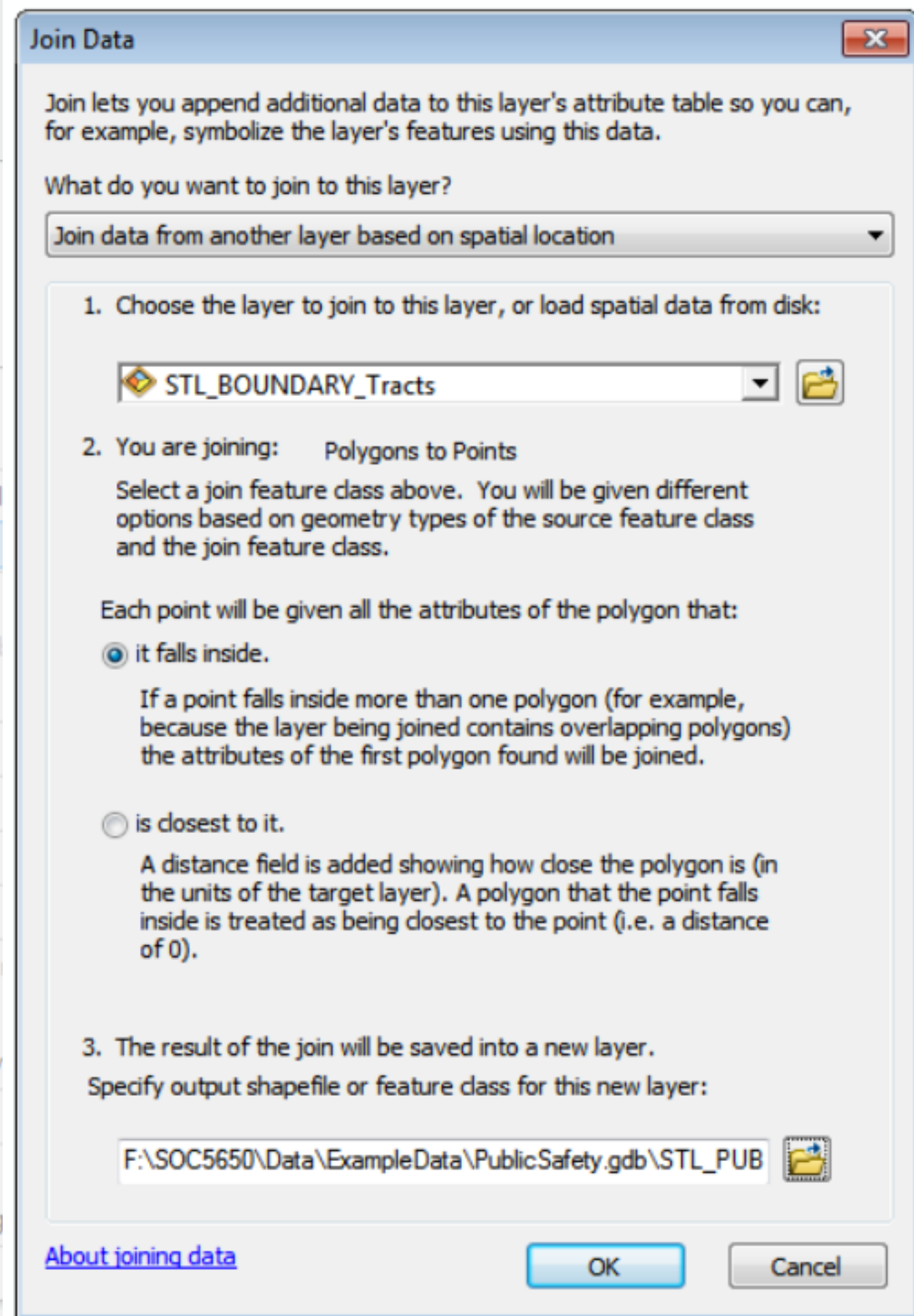
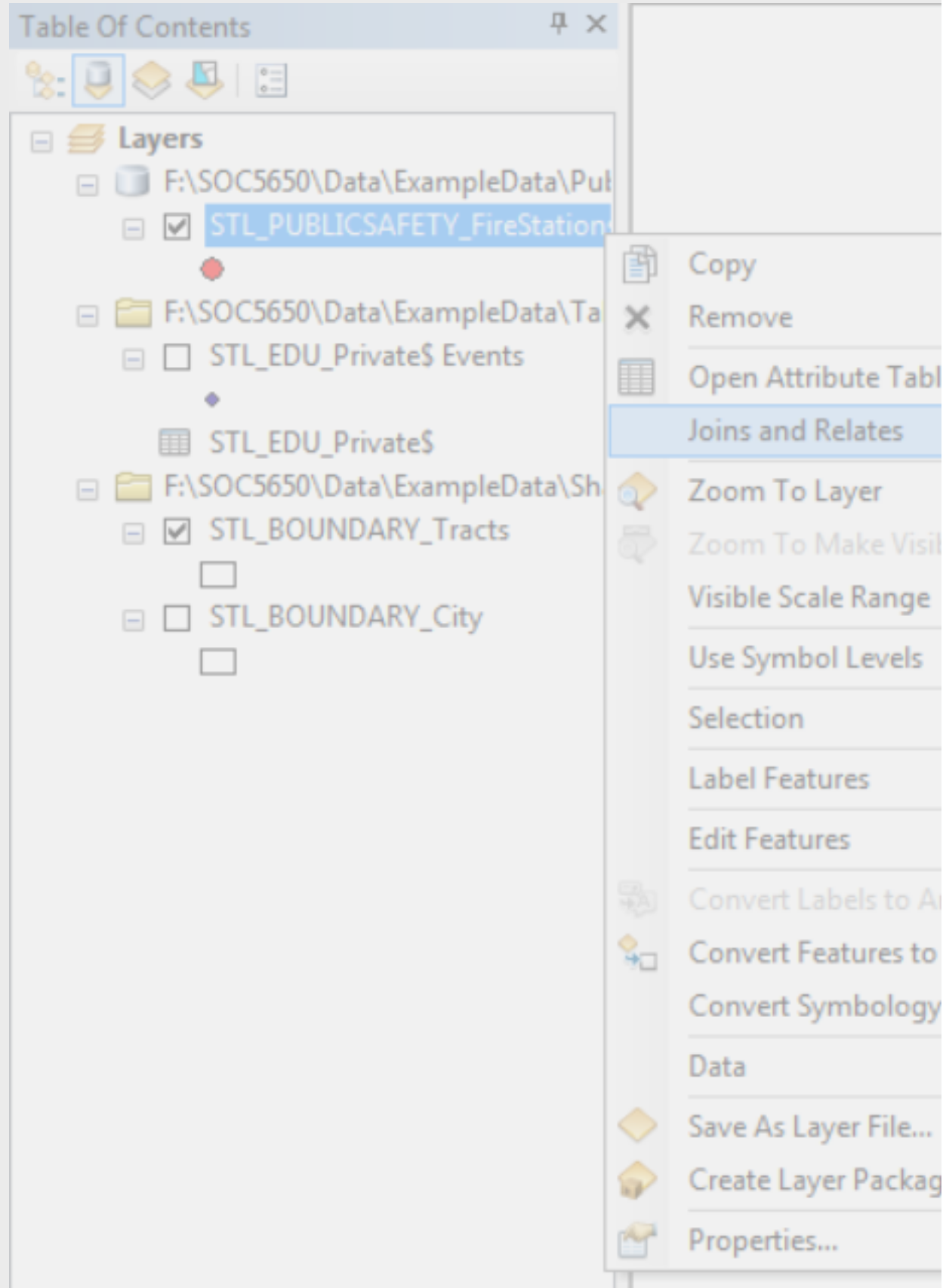


# POLYGONS TO POINTS





# POLYGONS TO POINTS



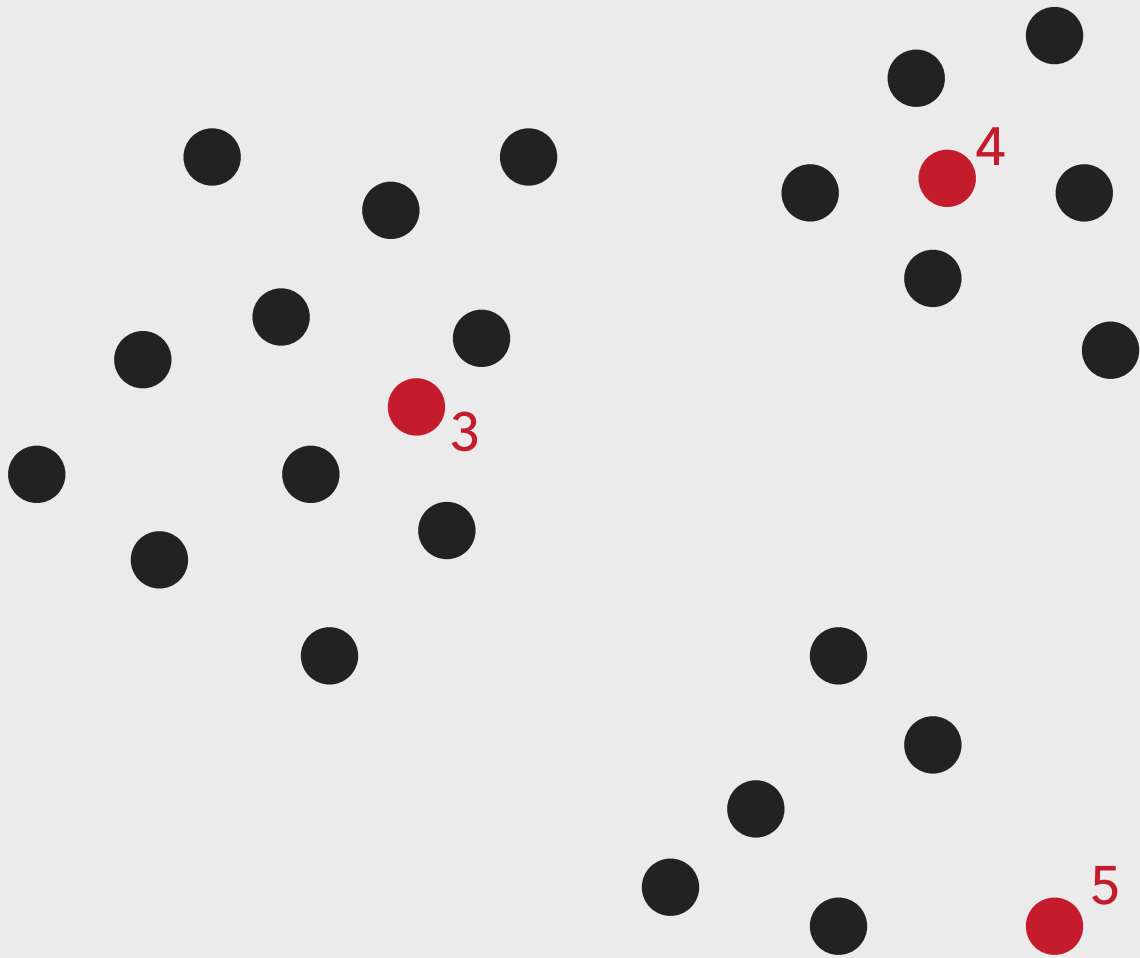
# POINTS TO POINTS

When the **both** layers consist of **points**,  
you can **assign the attributes** of one point  
to another based on proximity.

# POLYGONS TO POINTS

Target Layer	
ID	Shape
104	Point
105	Point
106	Point
105	Point
106	Point
107	Point
108	Point
109	Point

Reference Layer	
ID	Shape
3	Point
4	Point
5	Point



# POLYGONS TO POINTS

Target Layer

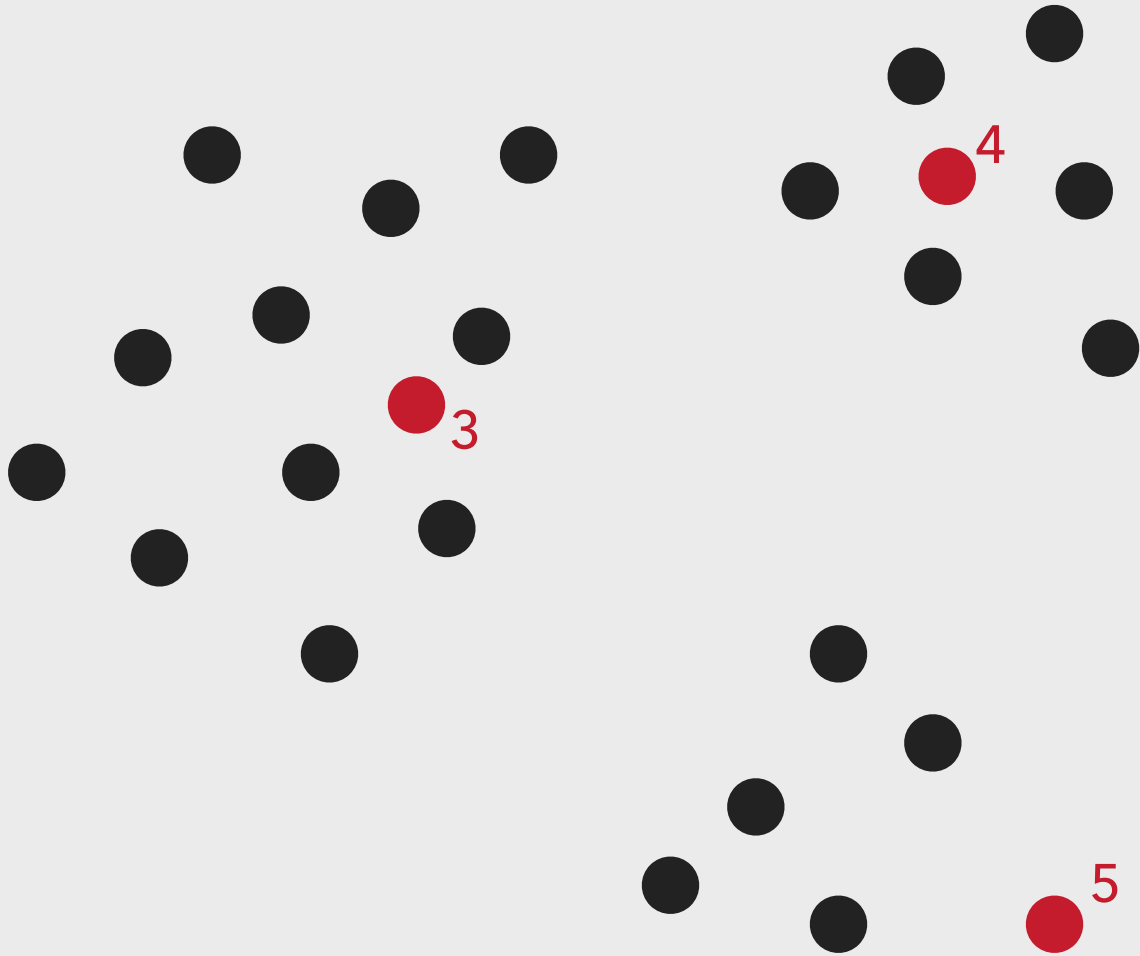
ID	Shape	sID
104	Point	3
105	Point	3
106	Point	4
105	Point	3
106	Point	3
107	Point	3
108	Point	3

109	Point	5
-----	-------	---

Reference Layer

sID	Shape
3	Point
4	Point
5	Point

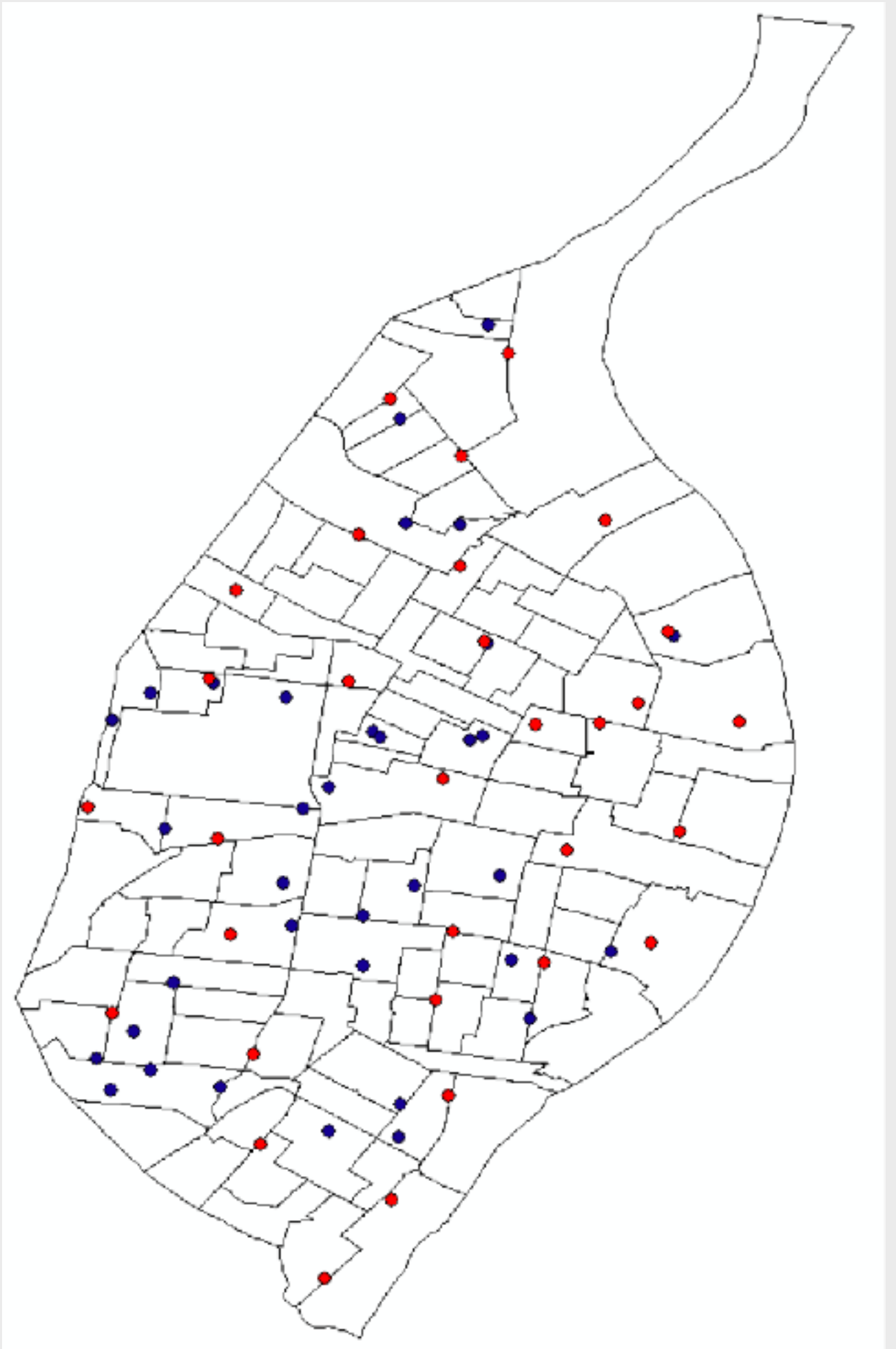
--	--



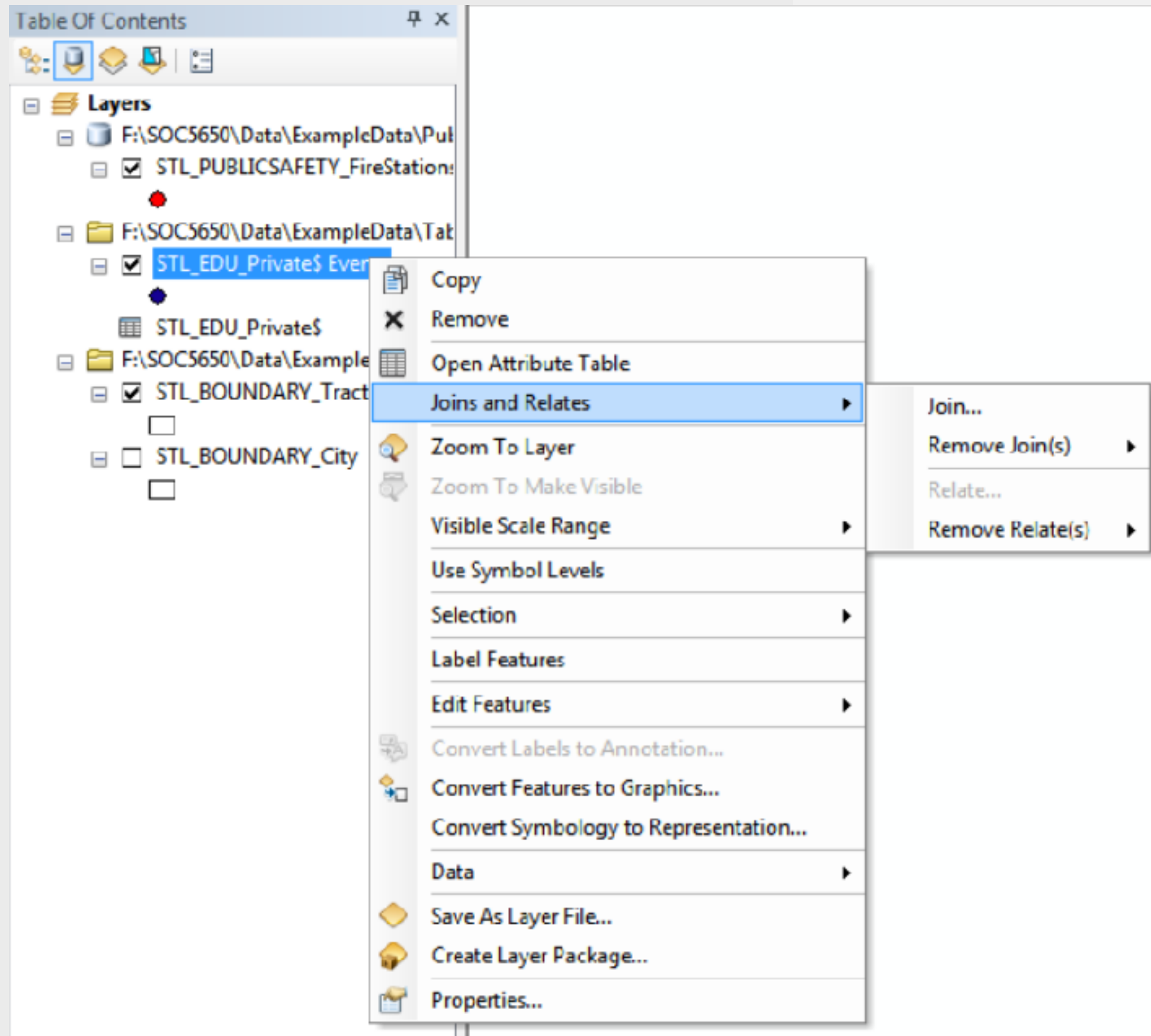
# POINTS TO POINTS

What is the nearest fire station to each private school?

# POINTS TO POINTS

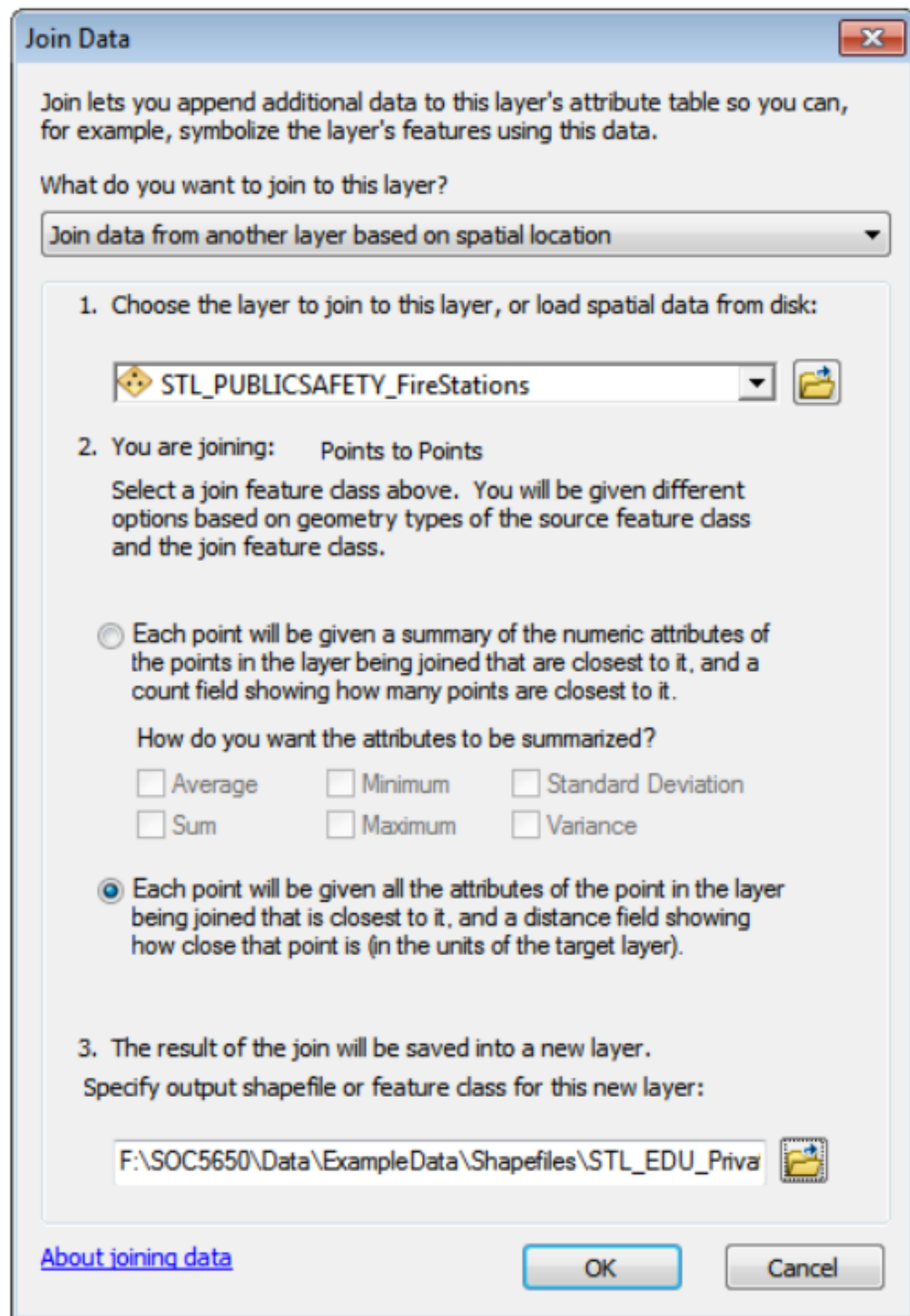
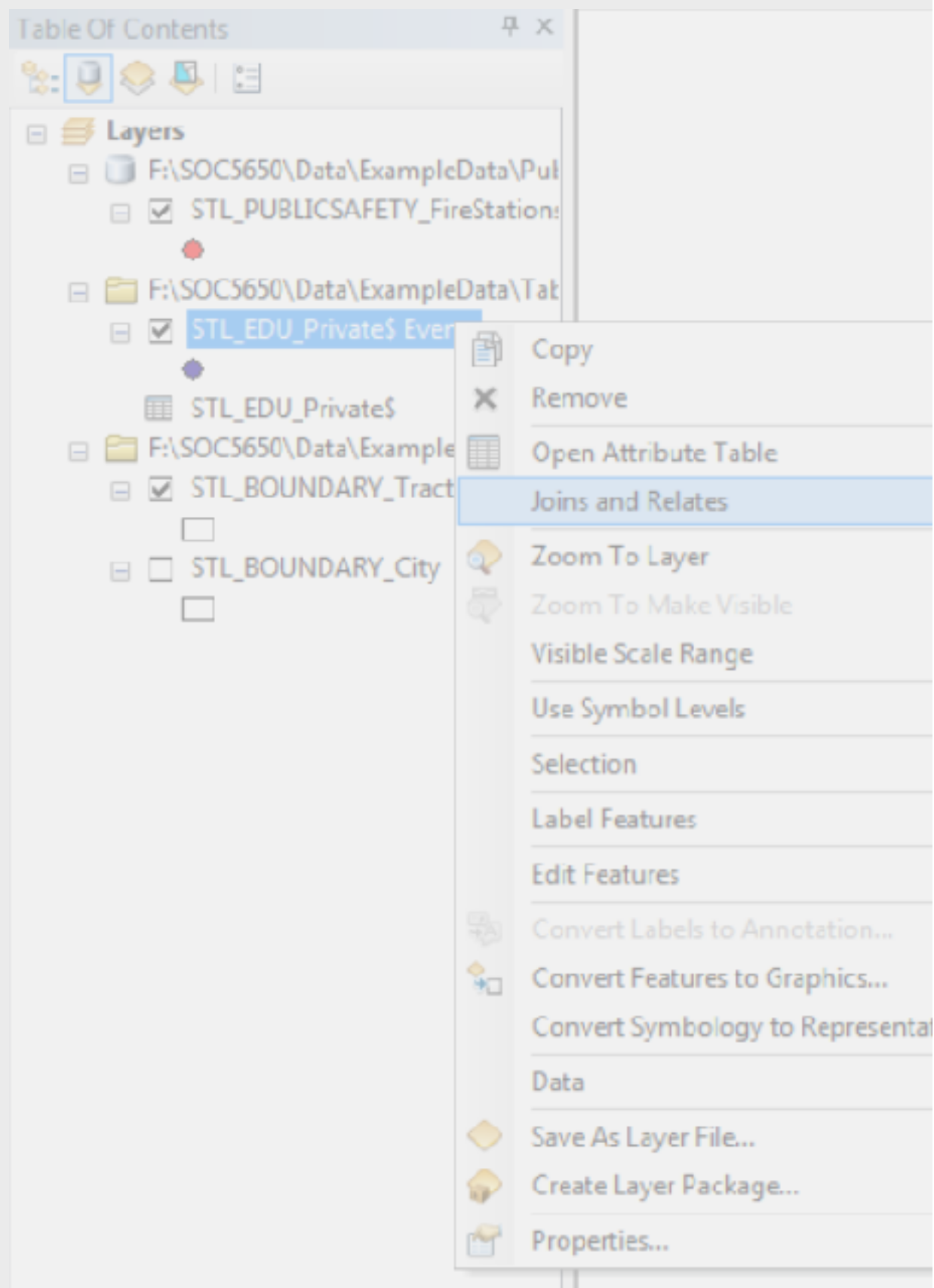


# POINTS TO POINTS





# POINTS TO POINTS





# 6 TABLE JOINS

# TABLE JOINS

When your **tabular data** have **ID numbers** that correspond to **ID numbers** in a **shapefile** or feature class, they can be **joined** to those spatial data.

# COMBINING TABULAR DATA

Target Layer

ID	X	Y
1	-90.236560	38.637241
2	-90.236799	38.636550
3	-90.237290	38.636661
4	-90.238154	38.636829
5	-90.237682	38.636735
6	-90.238942	38.636996
7	-90.239924	38.637206
8	-90.241313	38.637499
9	-90.236564	38.636492
10	-90.236062	38.636354



Tabular Data

ID	NAME
1	Morrissey Hall
2	Duborg Hall
3	Des Peres Hall
4	Beracha Hall
5	Xavier Hall
6	Xavier Annex
7	Notre Dame Hall
8	McGannon Hall
9	Adorjan Hall
10	Wuller Hall

# COMBINING TABULAR DATA

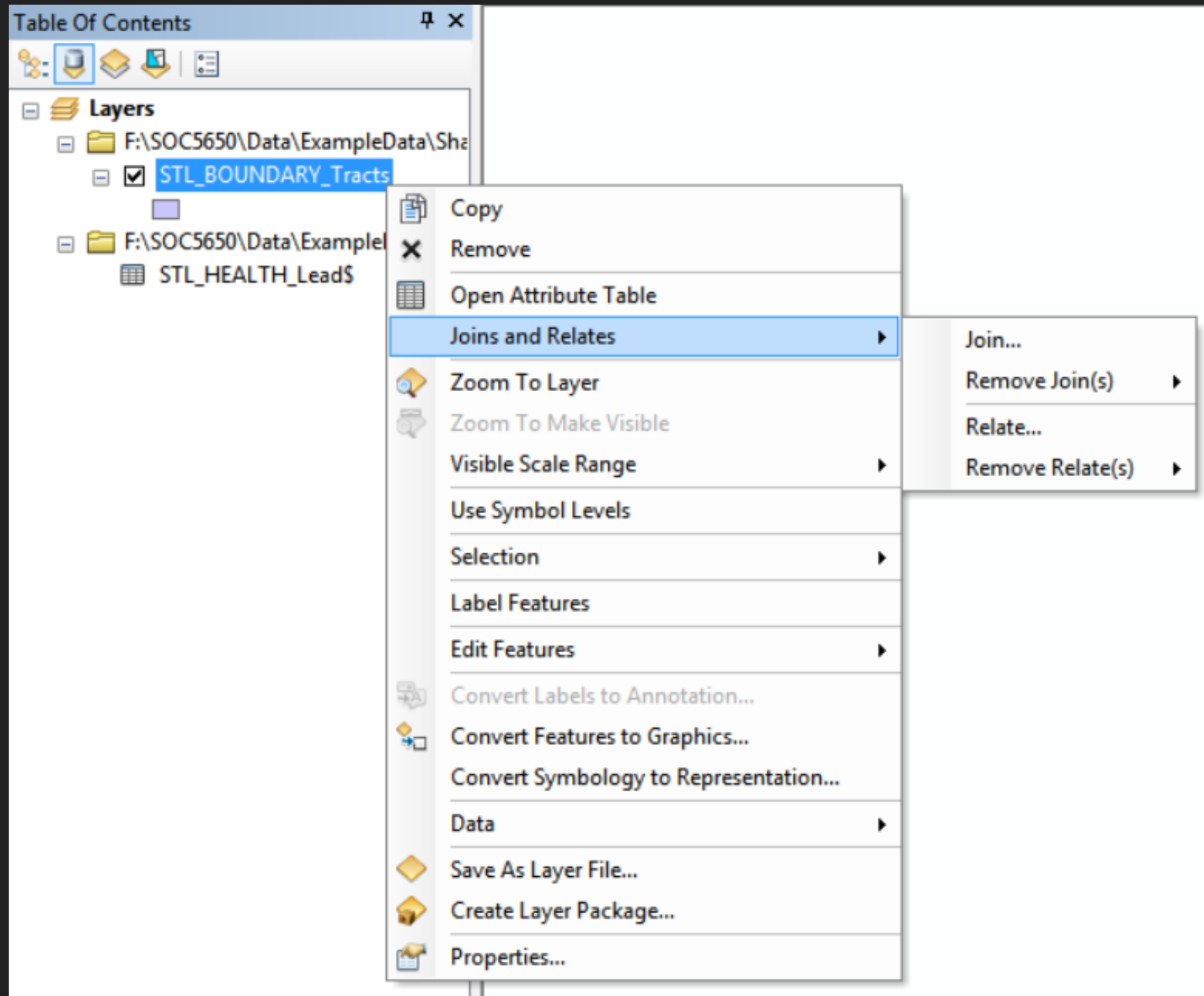
Joined Data

ID	X	Y	NAME
1	-90.236560	38.637241	Morrissey Hall
2	-90.236799	38.636550	Duborg Hall
3	-90.237290	38.636661	Des Peres Hall
4	-90.238154	38.636829	Beracha Hall
5	-90.237682	38.636735	Xavier Hall
6	-90.238942	38.636996	Xavier Annex
7	-90.239924	38.637206	Notre Dame Hall
8	-90.241313	38.637499	McGannon Hall
9	-90.236564	38.636492	Adorjan Hall
10	-90.236062	38.636354	Wuller Hall

# TABLE JOINS

We have data on blood lead level testing results for each Census Tract in a csv file as well as a shapefile of Census Tract boundaries.

# COMBINING TABULAR DATA IN ARCGIS



# COMBINING TABULAR DATA IN ARCGIS

**Join Data**

Join lets you append additional data to this layer's attribute table so you can, for example, symbolize the layer's features using this data.

What do you want to join to this layer?

Join attributes from a table

1. Choose the field in this layer that the join will be based on:  
geoidNum
2. Choose the table to join to this layer, or load the table from disk:  
STL\_HEALTH\_Lead\$  
☒ Show the attribute tables of layers in this list
3. Choose the field in the table to base the join on:  
TRACTCE

Join Options

☒ Keep all records  
All records in the target table are shown in the resulting table. Unmatched records will contain null values for all fields being appended into the target table from the join table.

☐ Keep only matching records  
If a record in the target table doesn't have a match in the join table, that record is removed from the resulting target table.

Validate Join

[About joining data](#)

OK Cancel

