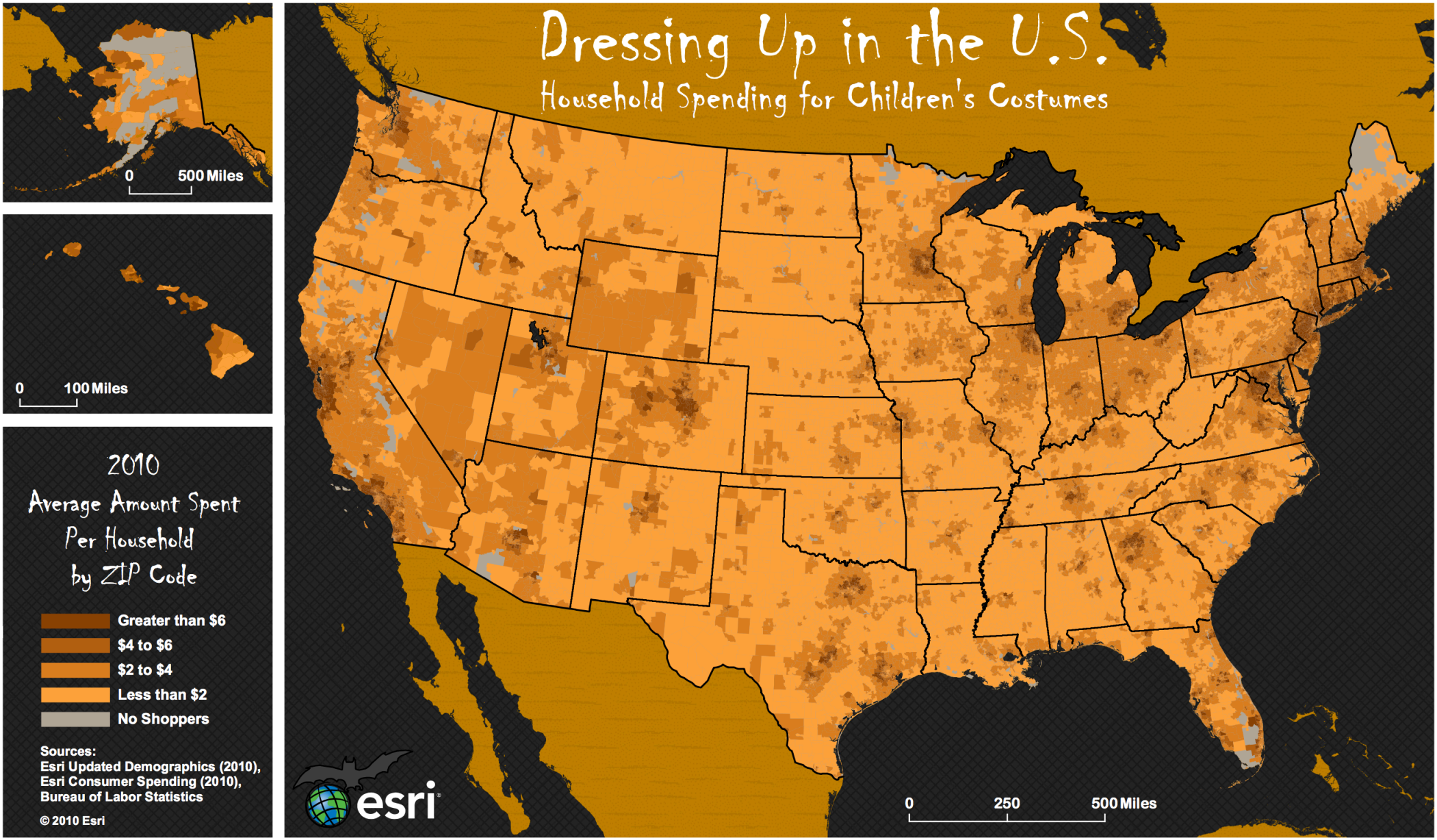
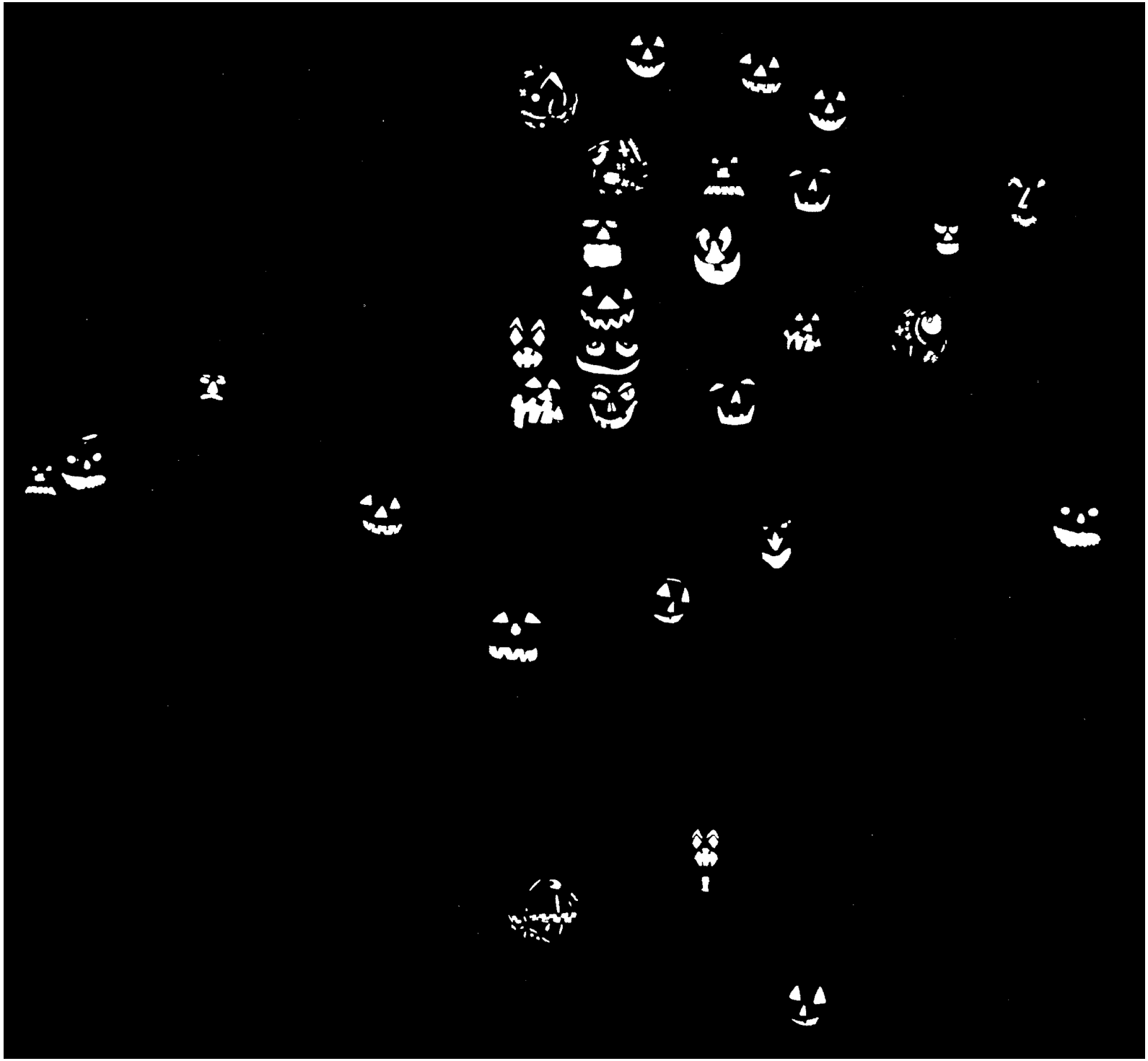

Dressing Up in the U.S.
Household Spending for Children's Costumes
0 500 Miles
0 100 Miles
2010
Average Amount Spent
Per Household
by ZIP Code
Greater than $6
$4 to $6
$2 to $4
Less than $2
No Shoppers
Sources:
Esri Updated Demographics (2010),
Esri Consumer Spending (2010),
Bureau of Labor Statistics
© 2010 Esri
esri
0 250 500 Miles



Happy Halloween!!

"Jack-O'-Lanterns" by Wood (2010)

Geography 360
October 31, 2016

# GIS data models & Review

1. *Questions and Announcements*
   - Happy Halloween!
   - Great job rating Seattles for their recognizability!
     3000+ votes already…

2. *GIS Data Models*
   - Topological vector versus spaghetti vector
   - Representing attributes

3. *Your questions about material for Quiz I*

# Quiz I: Wednesday

- Please be *on time* (or early) so you don't disturb your classmates.
- Worth 12.5% of your class grade.
- In lecture, so no computers are involved.
- No blue books. Can't interact with electronic devices. Bring only pencil/pen.
- You have the whole lecture period …but you may well not need all of it.
- Study Guide: https://canvas.uw.edu/courses/1066006/pages/quiz-i-study-guide

Format:

- Multiple choice questions
- Choice of short response questions.

- Short response questions:
  - Won't only ask you to recall things, but will also you to apply them to a situation you haven't quite seen before. May ask you to think of examples. May ask you to sketch. May ask you to explain the advantages and disadvantages of one approach compared to another.
  - None of the short response questions should require more than several well-chosen sentences if your responses are concise and effective. Clarity and reason are valued more than the unexplained presence of 'keywords'.

Scope:

- Everything assigned is fair game, up to and including lecture and readings for today. That said, focus first on understanding (and thinking about how to creatively apply and evaluate) the material mentioned in lecture and in the labs. Assigned readings (and even supplemental readings) help you do so, especially in those parts of the course that are more distant from your past experiences.

# Vector Models

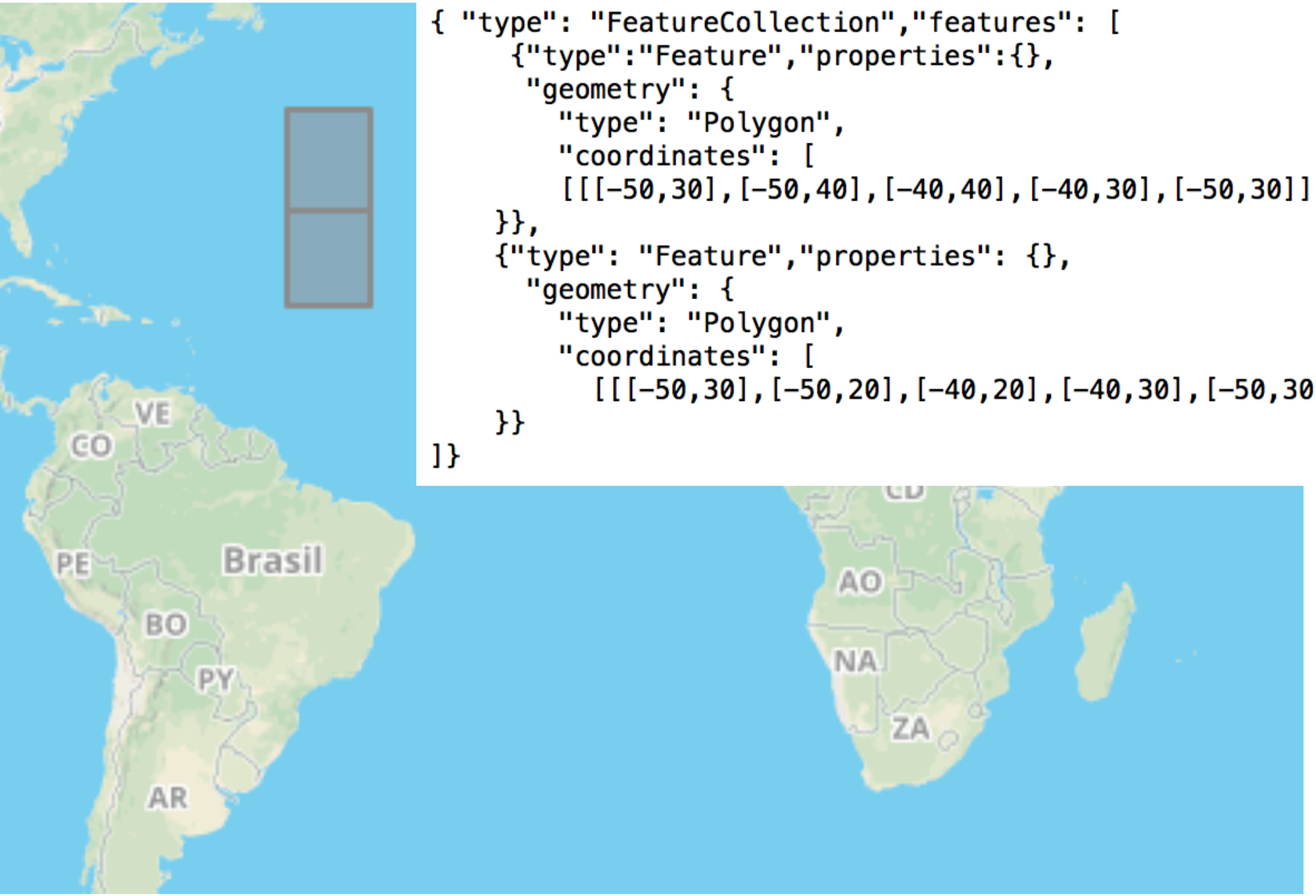
There are different approaches to **representing feature geometries** within a vector model.

One major differentiation is between vector data structures that are:

- ‘Spaghetti’ versus
- Topological

# Example of ‘Spaghetti’ model in practice

GeoJSON




```
{ "type": "FeatureCollection","features": [
     {"type":"Feature","properties":{},
      "geometry": {
        "type": "Polygon",
        "coordinates": [
        [[[-50,30],[-50,40],[-40,40],[-40,30],[-50,30]]]
    }},
    {"type": "Feature","properties": {},
      "geometry": {
        "type": "Polygon",
        "coordinates": [
          [[[-50,30],[-50,20],[-40,20],[-40,30],[-50,30]]]
    }}
]}
```

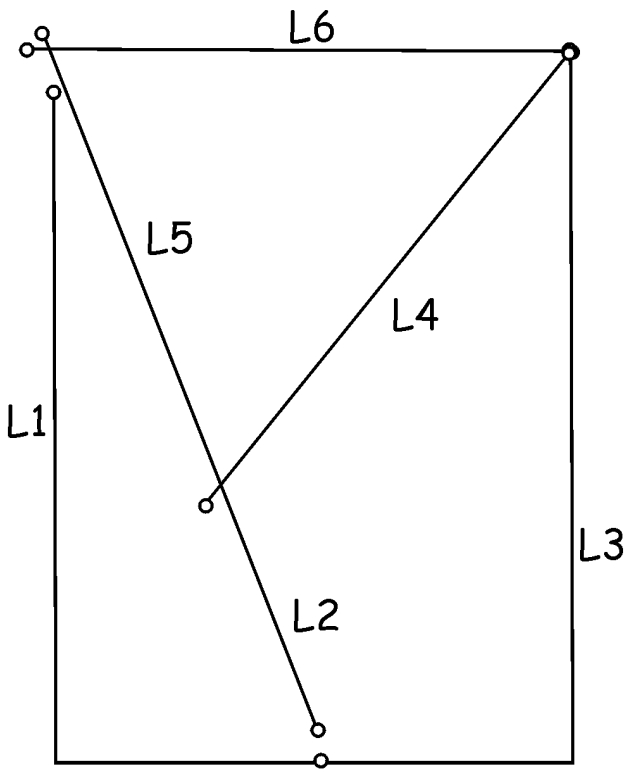
To do this yourself: [http://geojson.io/](http://geojson.io/)

# Implementing the vector data model: Topological vector structures

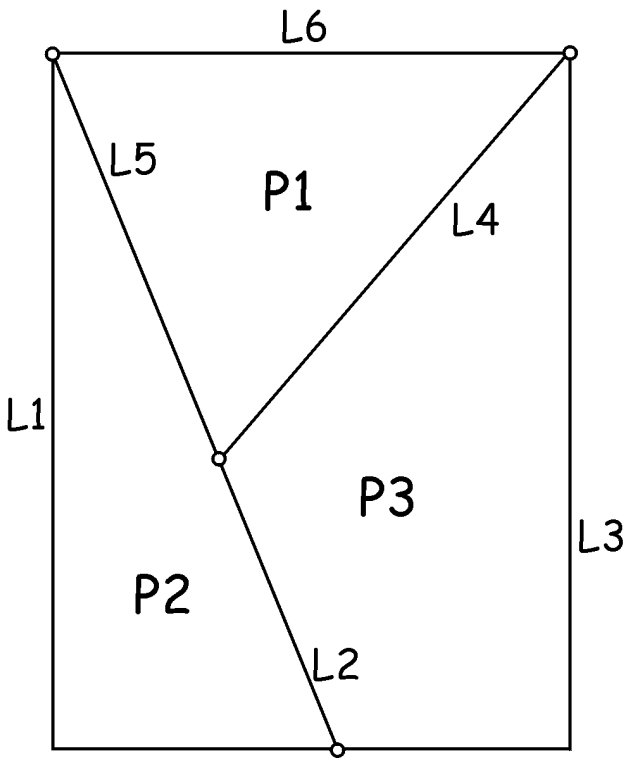
- Like the spaghetti approach, use points, lines, and polygons to define features
- But…in addition to coordinate locations, this approach records “topological relationships” among geometric features.
- A data structure that is aware of topology (in the sense this term is used in GIS) is one that stores relationships between different spatial elements.

  (e.g., the data structure has stored the fact that two polygons share the same line in their borders.)

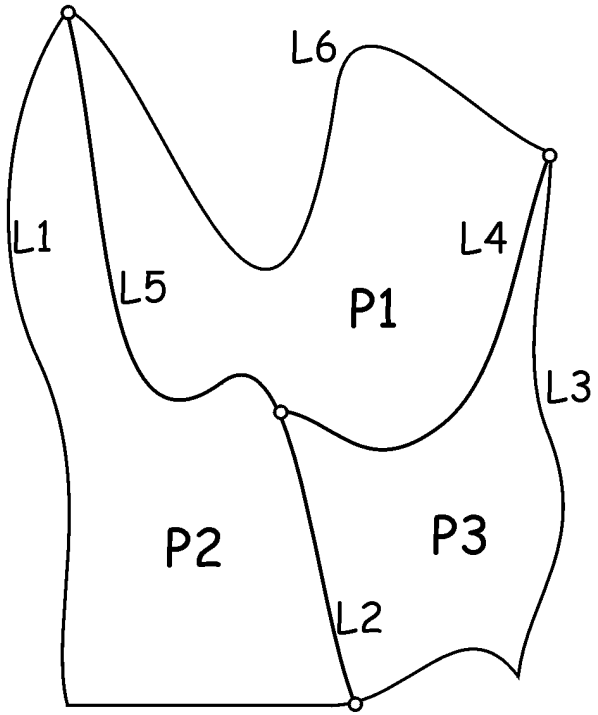
a) spaghetti




b) topological



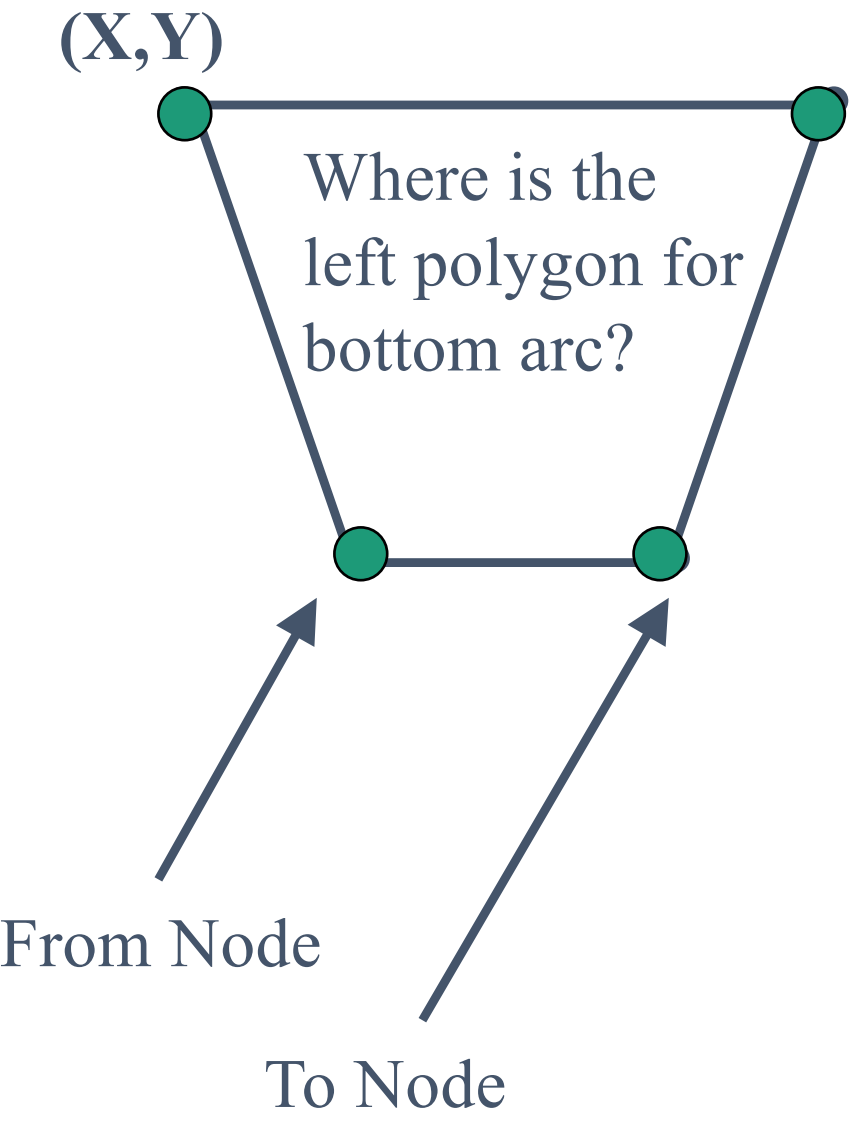

c) topological - warped




In a GIS sense, Figures b) and c) have the same topology (are topologically equivalent): their features have the same connectivity and adjacency.

(When you switch projections, those are relationships that generally remain constant.)

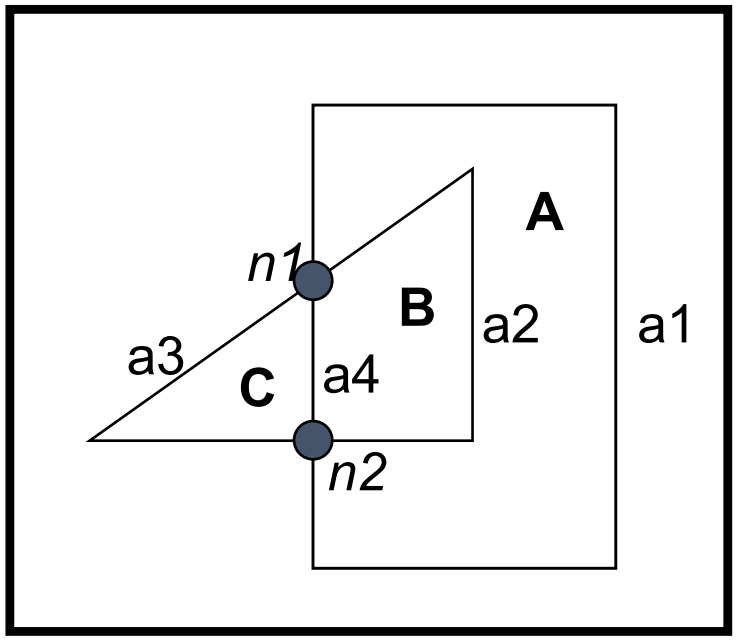
# With topological data models, the data encode relationships:




Each element of this image (node, arc, polygon) is stored with information about:

- X,Y locations for all points
- From/To nodes for each arc
- Arcs in each polygon
- Left/Right polygons for each arc

# Topological vector model: How might data be stored?




Note: Coordinate values and most vertices not shown

**Arc Coordinate Data**

| Arc | StartXY | IntermediateXY | EndXY |
|---|---|---|---|
| a1 | 4,5 | (4,8), (8,8), (8,1), (4,1) | 4,3 |
| a2 | 4,5 | (6,7), (6,3) | 4,3 |
| a3 | 4,5 | (1,3) | 4,3 |
| a4 | 4,3 | | 4,5 |

**Arc Topology**

| Arc | Start | End | Left | Right |
|---|---|---|---|---|
| a1 | n1 | n2 | | A |
| a2 | n1 | n2 | A | B |
| a3 | n1 | n2 | C | |
| a4 | n2 | n1 | C | B |

**Node Topology**

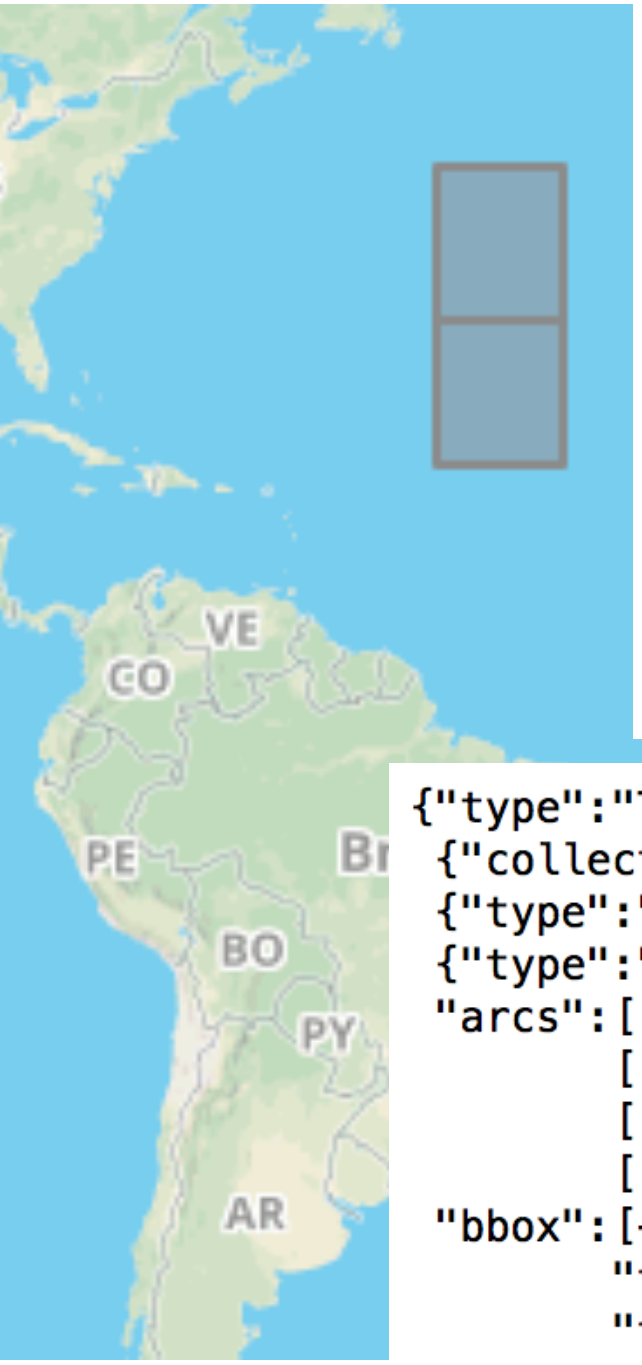
| Node | Arcs |
|---|---|
| *n1* | a4, a2, a1, a3 |
| *n2* | a2, a4, a3, a1 |

**Polygon Topology**

| ID | Arcs |
|---|---|
| A | a1, a2 |
| B | a2, a4 |
| C | a3, a4 |

# ‘Spaghetti’ versus Topological

## GeoJSON: Spaghetti




```
{ "type": "FeatureCollection","features": [
    {"type":"Feature","properties":{},
     "geometry": {
       "type": "Polygon",
       "coordinates": [
       [[-50,30],[-50,40],[-40,40],[-40,30],[-50,30]]]
   }},
   {"type": "Feature","properties": {},
     "geometry": {
       "type": "Polygon",
       "coordinates": [
         [[-50,30],[-50,20],[-40,20],[-40,30],[-50,30]]]
   }}
]}
```

```
{"type":"Topology","objects":
 {"collection":{"type":"GeometryCollection","geometries":[
 {"type":"Polygon","arcs":[[0,1]]},
 {"type":"Polygon","arcs":[[2,1]]}]}},
 "arcs":[
        [[0,5000],[0,4999],[9999,0],[0,-4999]],
        [[9999,5000],[-9999,0]],
        [[0,5000],[0,-5000],[9999,0],[0,5000]]],
 "bbox":[-50,20,-40,40],
        "transform":{"scale":[0.001000100010001,0.002000200020002],
        "translate":[-50,20]}}
```

To do this yourself: https://jeffpaine.github.io/geojson-topojson/ and http://geojson.io/

## TopoJSON: Topological

# Some reasons you can benefit from storing topological relationships in your data

(or reasons to at least benefit from having the GIS consider topological relationships when doing your analyses.)

1. Topological rules can be used to create consistently structured data, even to detect and correct 'errors'.

2. Efficiency in spatial analysis.

   *Simple example:* Testing whether there are two neighboring countries on a map that share the same color becomes easy.

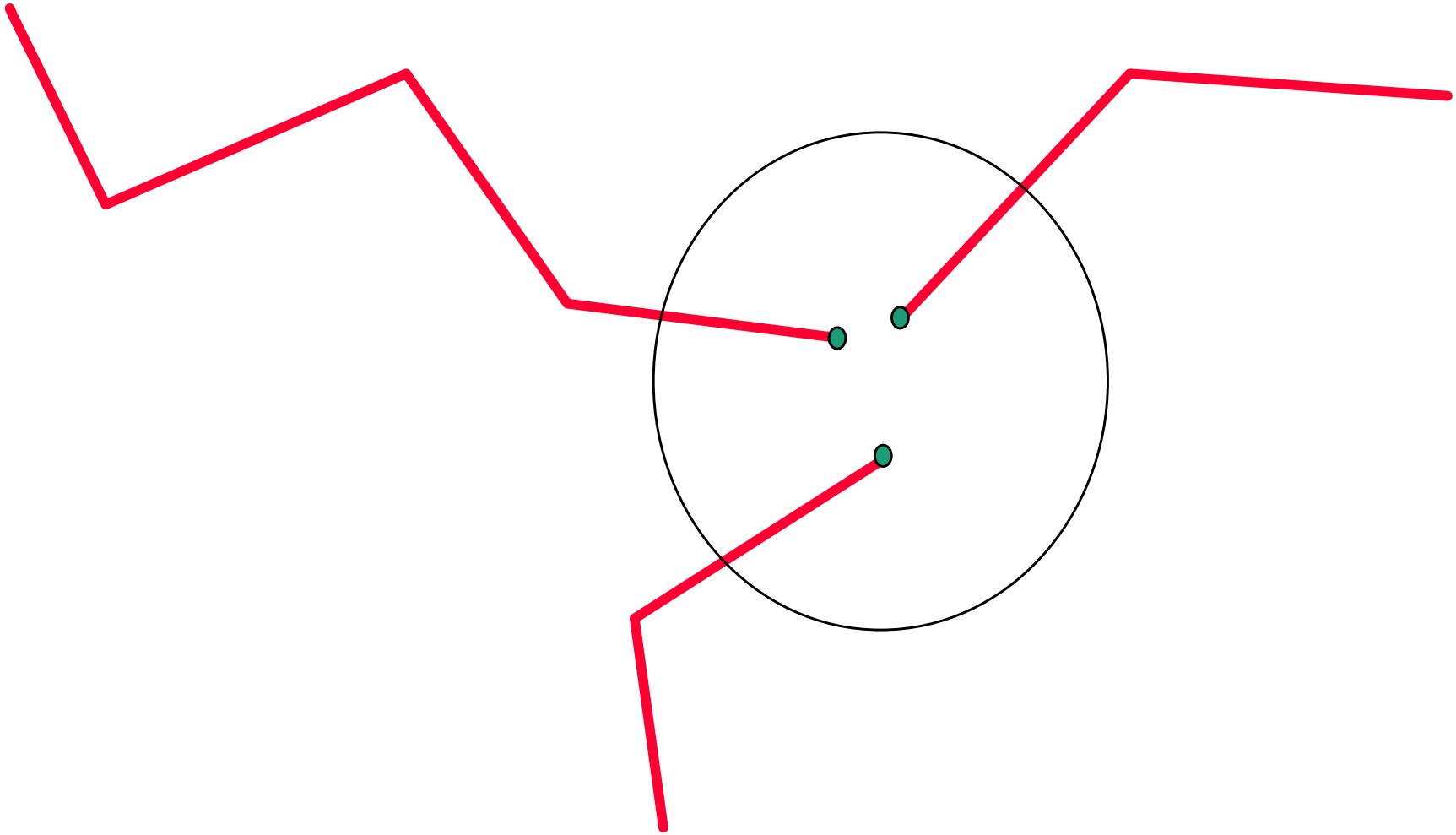
   (Can you think of what procedure you would tell the computer to follow in order to figure that out, otherwise?)

# Examples of topological rules

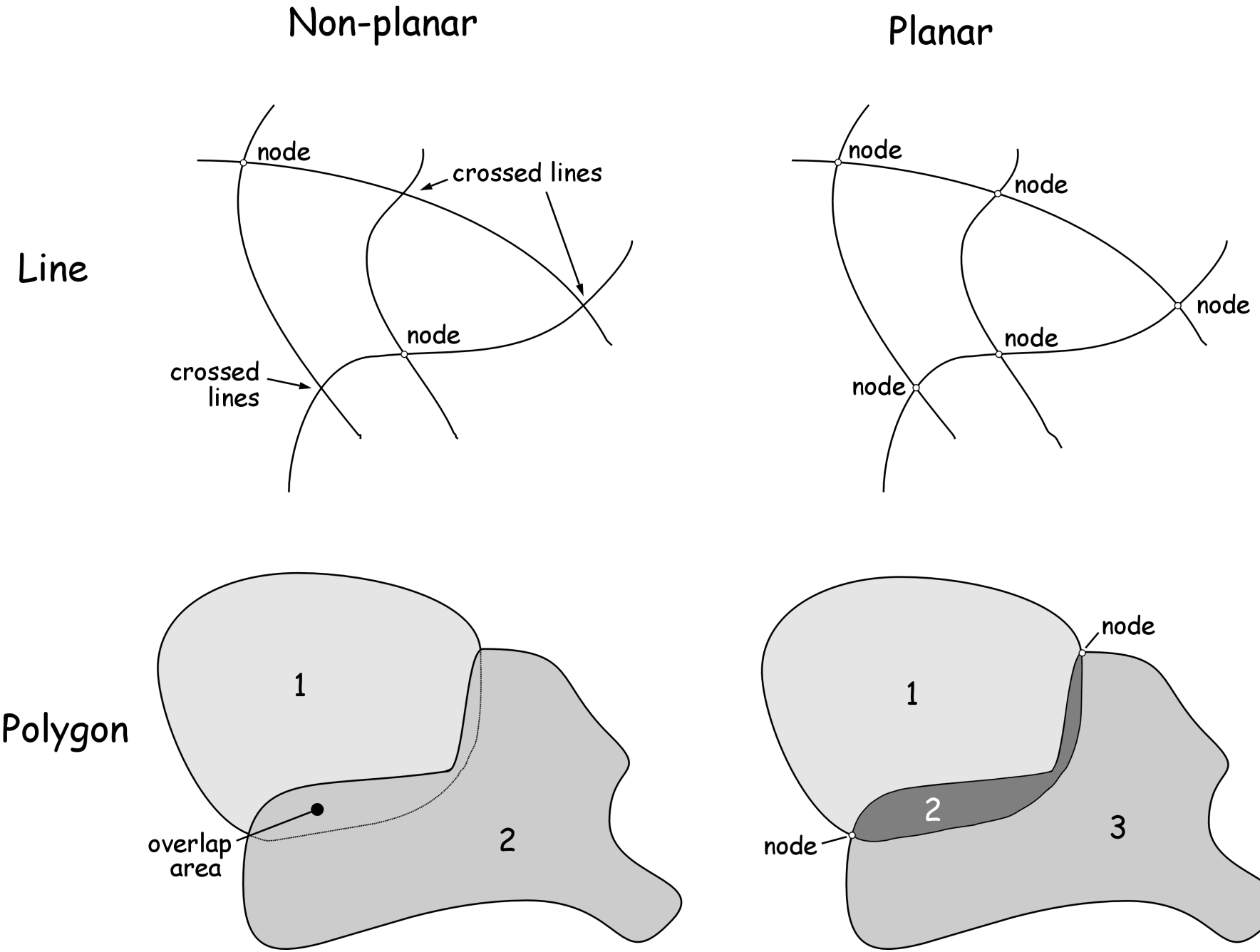
- No two individual features can overlap within a layer.

- A rule involving two layers:
  Each feature in a given layer must lie within a single feature in a second layer (e.g., counties within states).

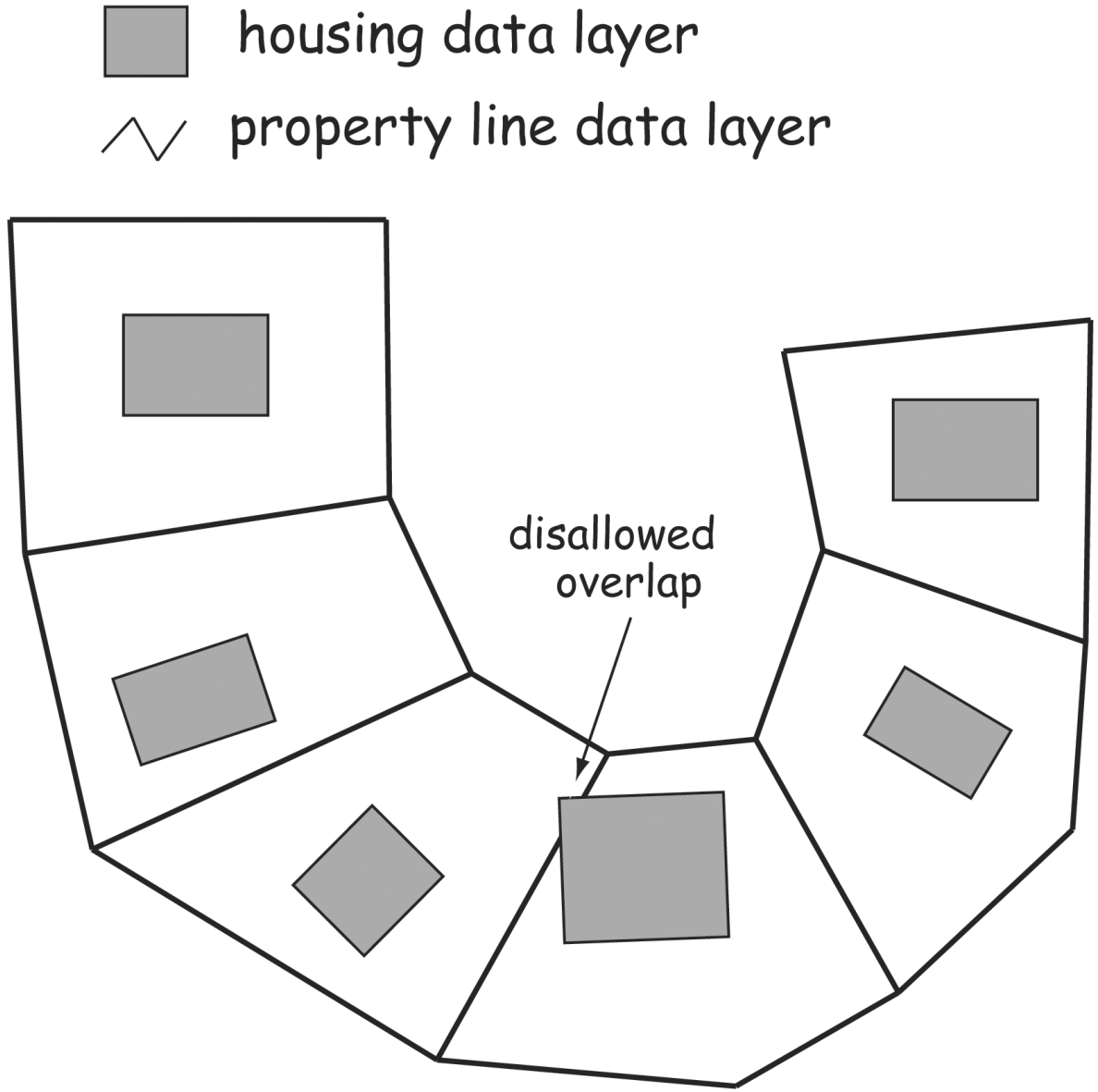
Topological rules can be chosen to exclude: *unsnapped nodes*



# Enforcing a rule that the topology must be *planar*

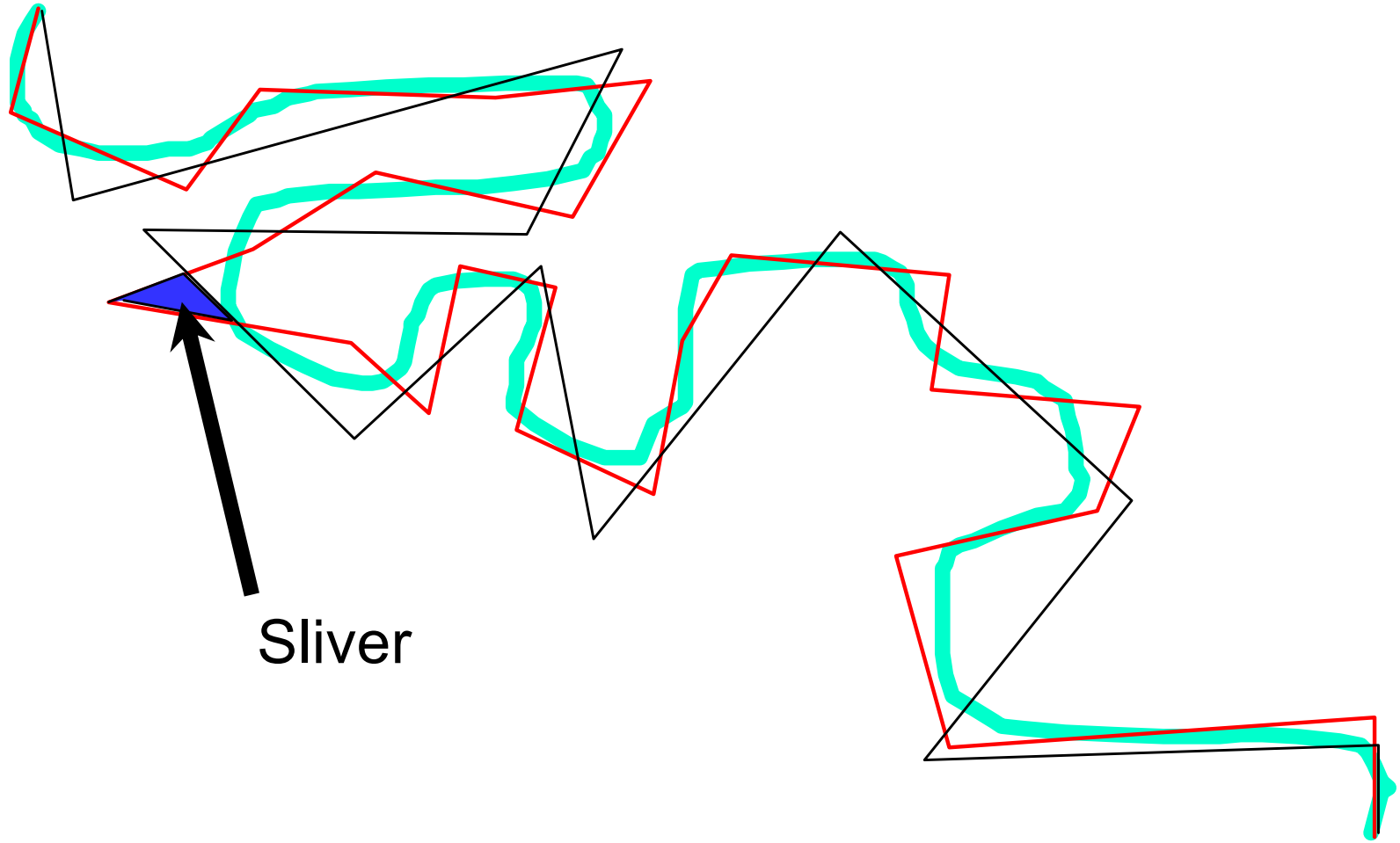


# Enforcing topological rules across data layers



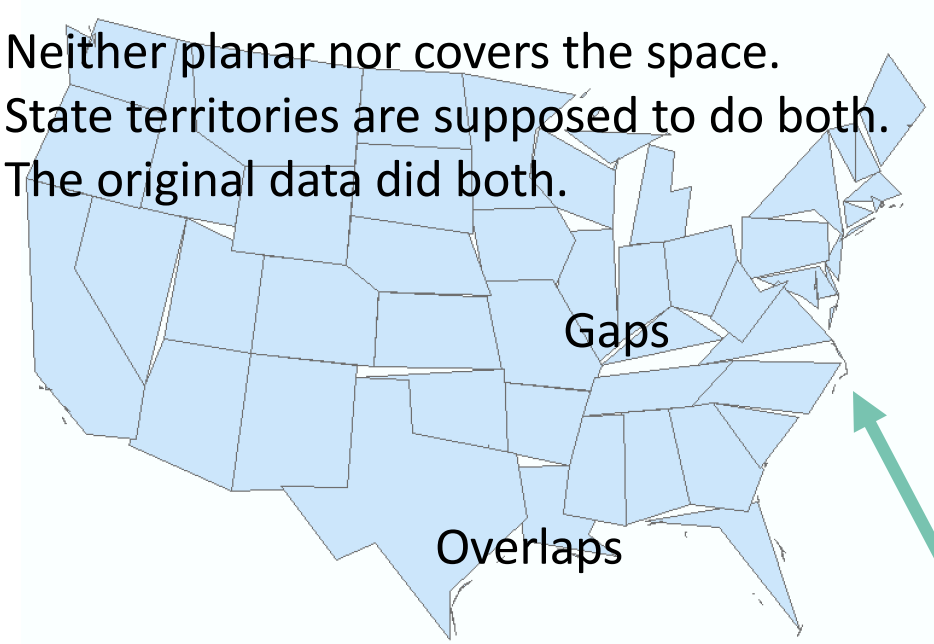
# Topological rules can be used to identify (and perhaps avoid or exclude) *slivers*.

Slivers can be introduced by your analyses, often by combining data from multiple sources.

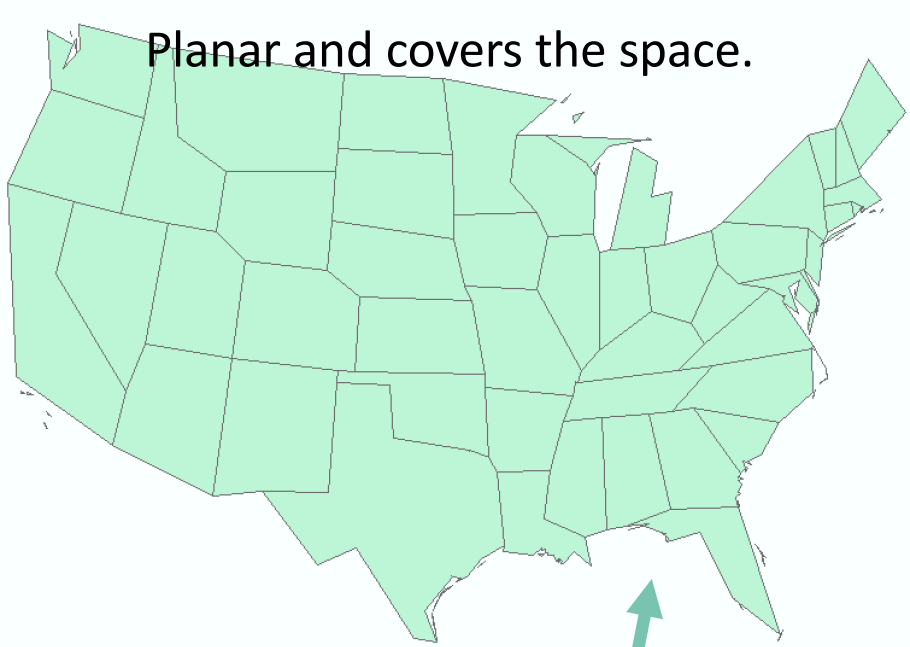


# When the GIS gives you an option, have topological awareness work in your favor…

Neither planar nor covers the space.
State territories are supposed to do both.
The original data did both.

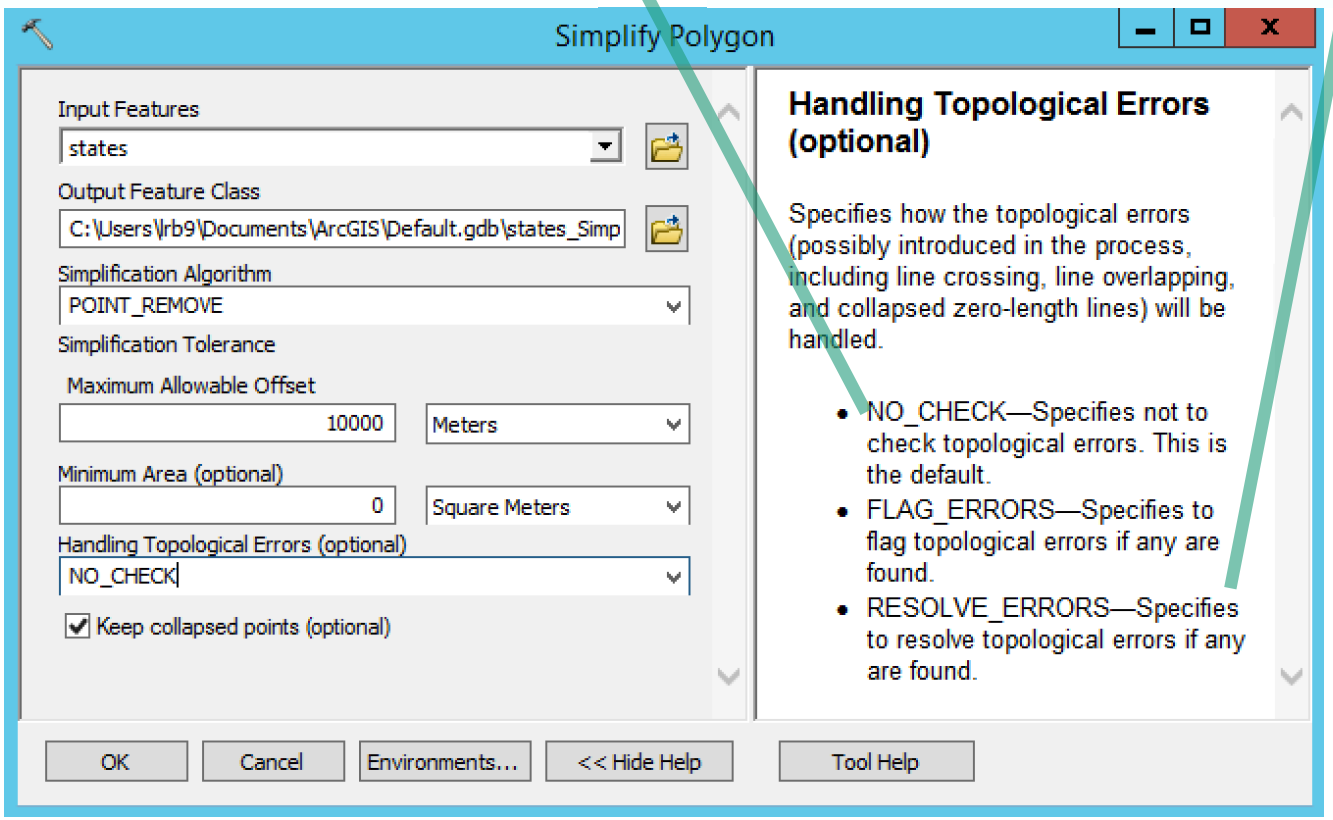



Planar and covers the space.



On the right side simplification, topology has not been stored in the data structure… **but** it is being calculated and checked for whether it conforms to an ideal (and if not, resolving the 'errors').

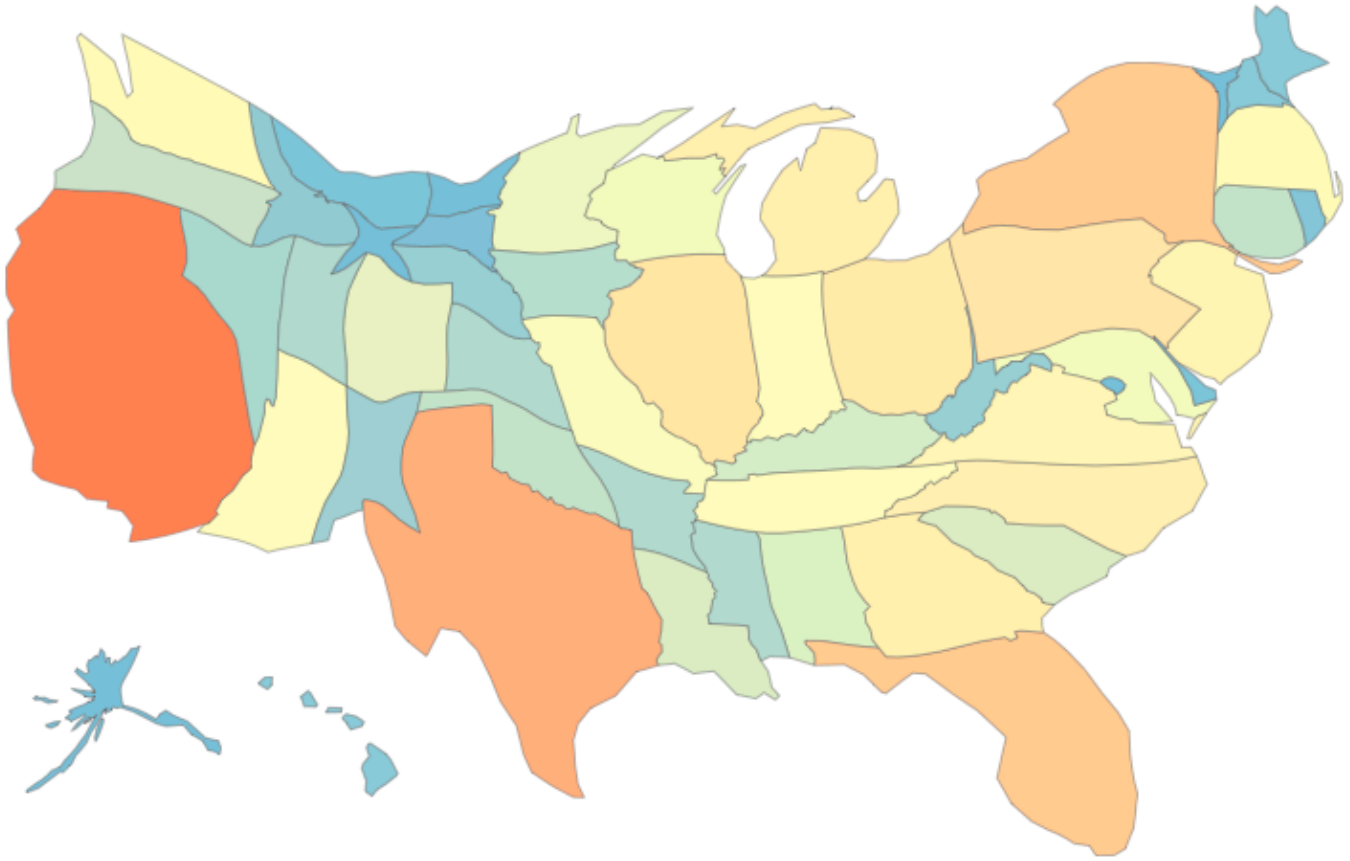
Less efficient, but still very useful.



# Using topological data can prevent problems like slivers, too…

## Cartograms with d3 & TopoJSON

Scale by Population Estimate in 2011 calculated in 0.1 seconds



### About

cartogram.js is a JavaScript implementation of an algoritm to construct continuous area cartograms, by James A. Dougenik, Nicholas R. Chrisman and Duane R. Niemeyer, ©1985 by the Association of American Geographers. This example combines TopoJSON-encoded boundaries of the United States from Natural Earth with 2011 US Census population estimates to size each state proportionally.

http://prag.ma/code/d3-cartogram/?segmentized#popest/2011

# Can you figure out which of the following applications would NOT be helped by having topological information?

A. Finding which toxic chemical storage sites are within a mile of a given house.
[Would not be particularly helped as written because the geometrical relationships you need to determine are not meaningfully stored within any of the data layers I mentioned here—the exception might be if you stored information about the proximity of other points to a point with each such point, but this is not likely. I won't ask you a question in this ambiguous format on the quiz.]

B. Finding the fastest road route from one city to another.
[Would be helped because topological information would let you immediately know how the roads were connected, instead of having to search this out of a set of unconnected coordinates.]

C. Finding places where the rain is greater than 30 inches/year and the elevation is less than 3000 feet.
[This is a query where responding involves knowledge of attribute data first and foremost, not geometrical relationships, so topological information (in the GIS sense) is not particularly useful here.]

D. They would all be helped!