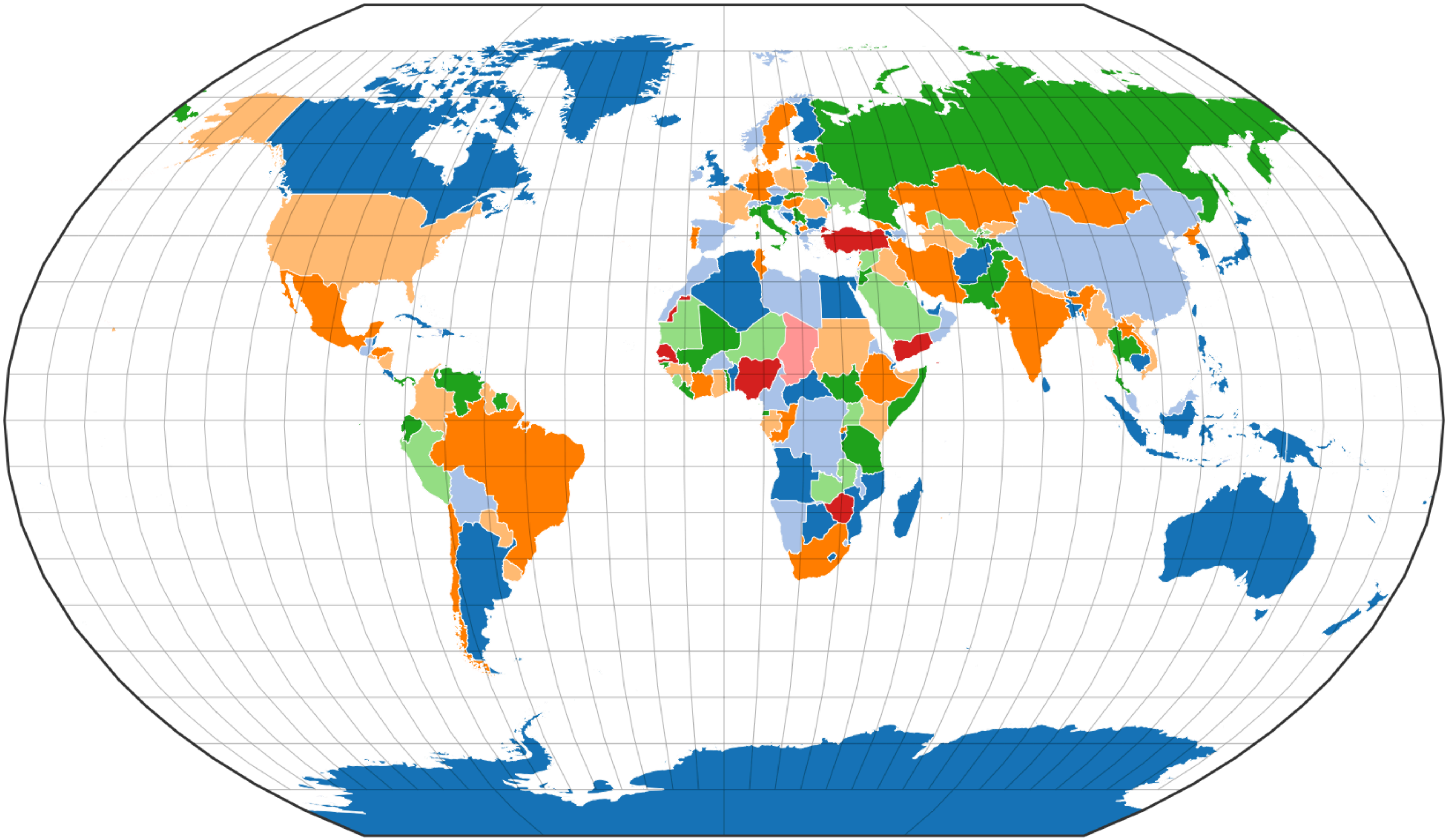
What might you need to know to make this map?



Based on Mike Bostock's World Map, modified to automatically colour countries such that no adjacent countries share the same colour.

http://bl.ocks.org/jasondavies/4188334

Geography 360
October 28, 2016

GIS data models

1. *Questions and Announcements*
   - More time for review on Monday.
   - More study guidance to be posted; watch for "Announcement."

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

3. *Your questions about material for Quiz I*

How GIS quantifies location (revisited)

When using GIS, there may be many different coordinate systems in play at the same time:

- Original data coordinate system stored on web or in files.
  (Sometimes the coordinate system isn't set or is set incorrectly and you have to [re-]set it yourself. Example: The .prj file in a Shapefile, created using 'Define Projection' *but* changed if it exists by the 'Project' tools in ArcMap)
- 'Data frame' coordinate system
  (GIS usually temporarily reproject all the coordinates of different data layers you have loaded into one single coordinate system so data can be compared! In ArcGIS Online, it is the coordinate system of the first data you loaded, but in most GIS, it can be set by you.)
- Analysis coordinate system
  (Make sure that the GIS has reprojected all the data into the same coordinate system before calculating anything…and make sure that the coordinate system preserves the properties that the calculation needs! Analysis tools often ask which coordinate system to use.)
- Visualization/graphical coordinate system
  (Make sure that you and your project's viewers are seeing data displayed in a coordinate system that is appropriate to the properties, phenomena, and message you want to convey!)

Data Models in a GIS

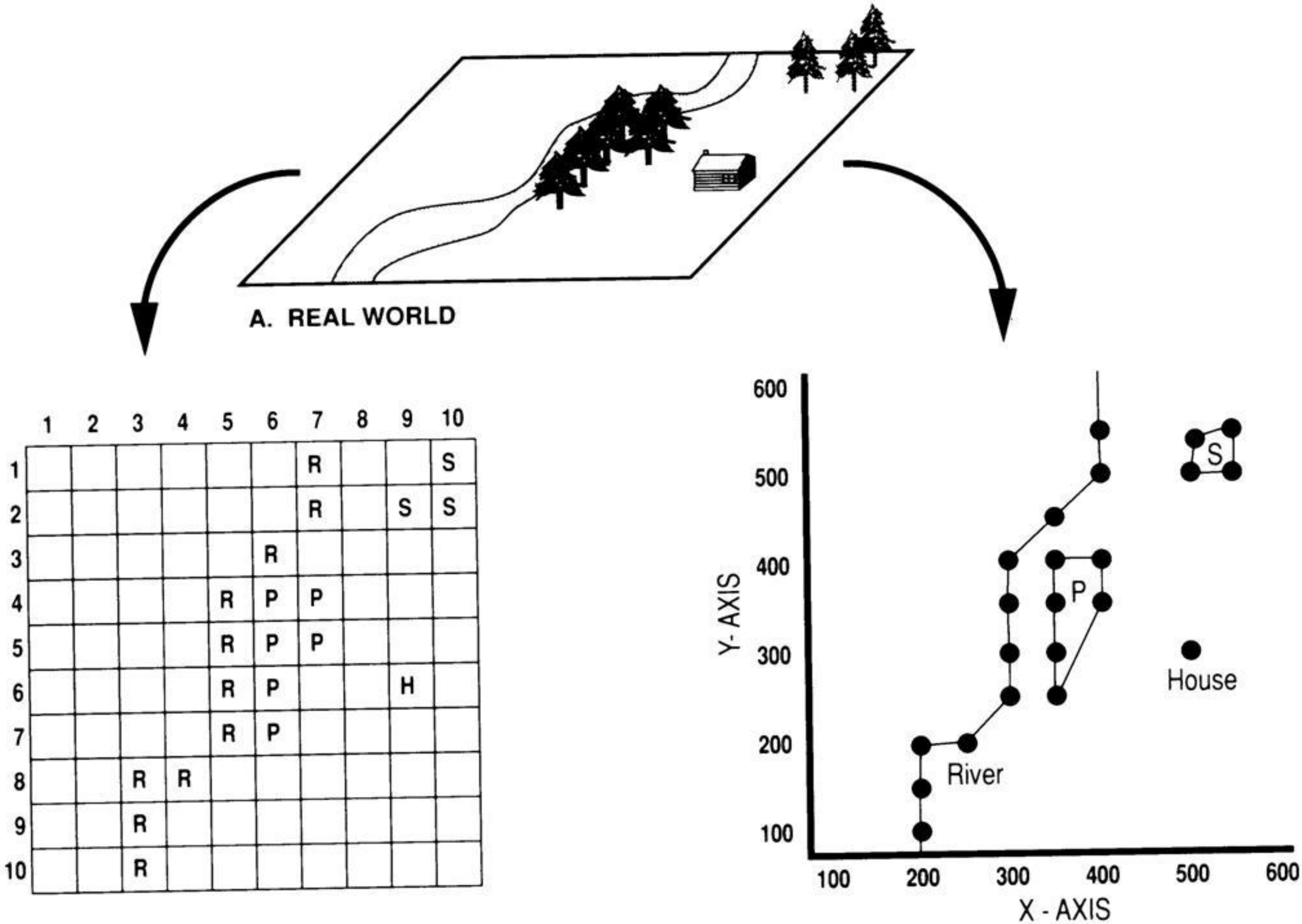
- **Data model:** the way that spatial relationships and attributes are conceptually defined and organized in a GIS.
  - Or: How we represent ‘real world’ objects and their characteristics in a computer.

- There are multiple/different data models.
  - They have different rules & conventions.
  - They approach representation differently.

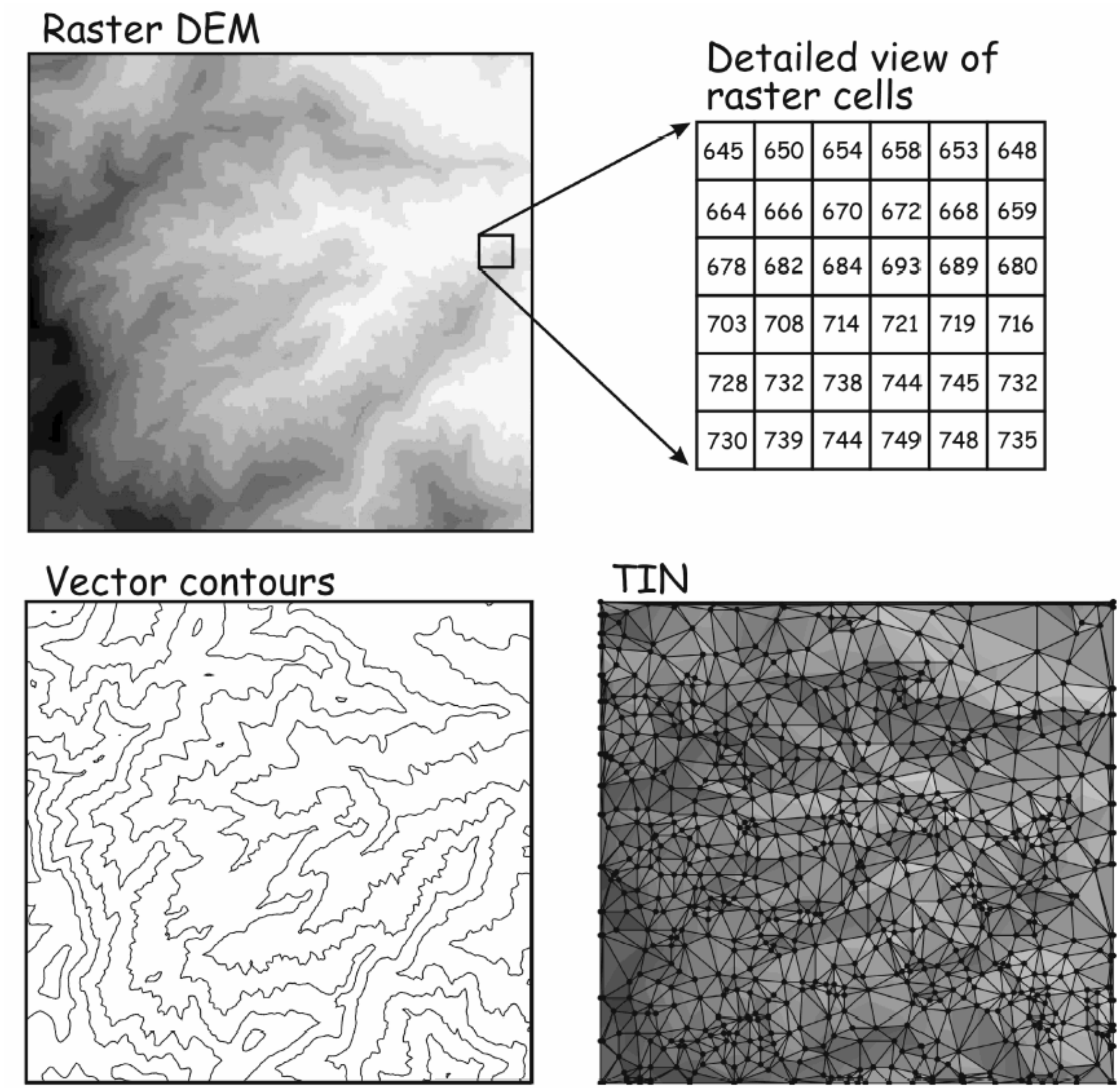
- Common examples: Raster, Vector, TIN

(Why should you care?)

Using two different data models:



[Visualizing the results of using] Different data models:



Vector Models

- Basic geometries: Points, Lines, and Polygons.
- Points stored by spatial coordinates.
- Lines and polygons built from sequences of points.
- Attributes then associated with those geometric figures.
- Data you have been using in ArcGIS Online, geojson.io, and OpenStreetMap are stored using three variants of a vector data model.

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

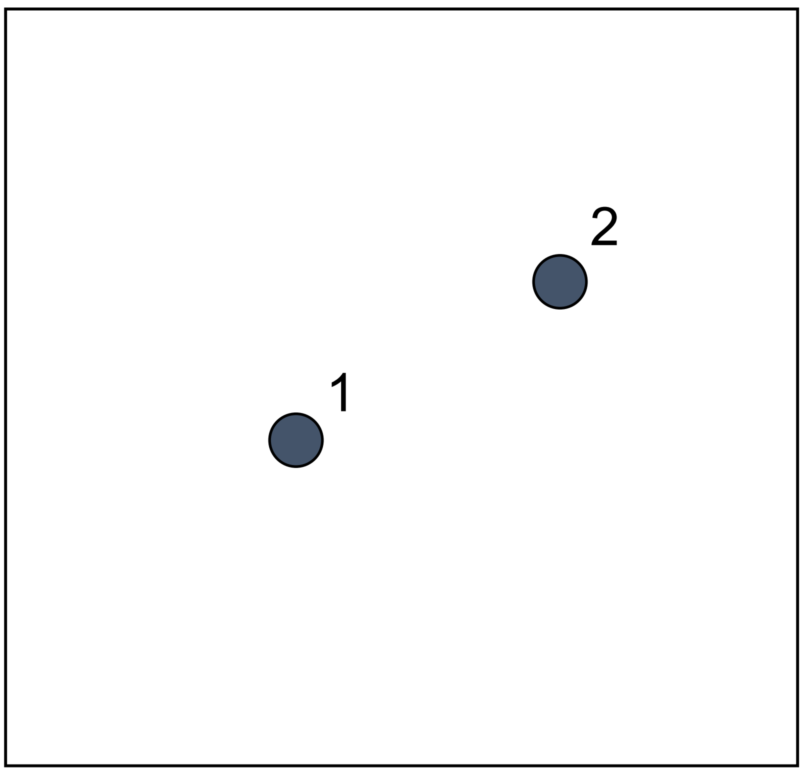
Implementing the vector data model: Spaghetti approaches

Each point, line, or polygon is stored as a record providing:

1) an ID for the object, and

2) the coordinates that define that object

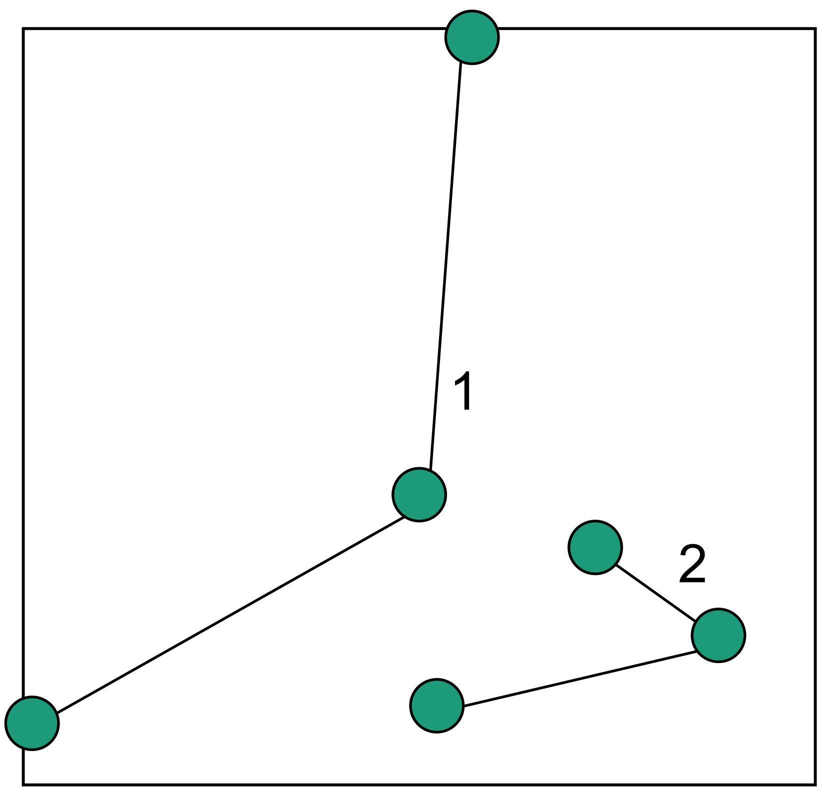
e.g. points:



ID	Coordinates
1	3,4
2	5,5

Spaghetti vector models, continued.

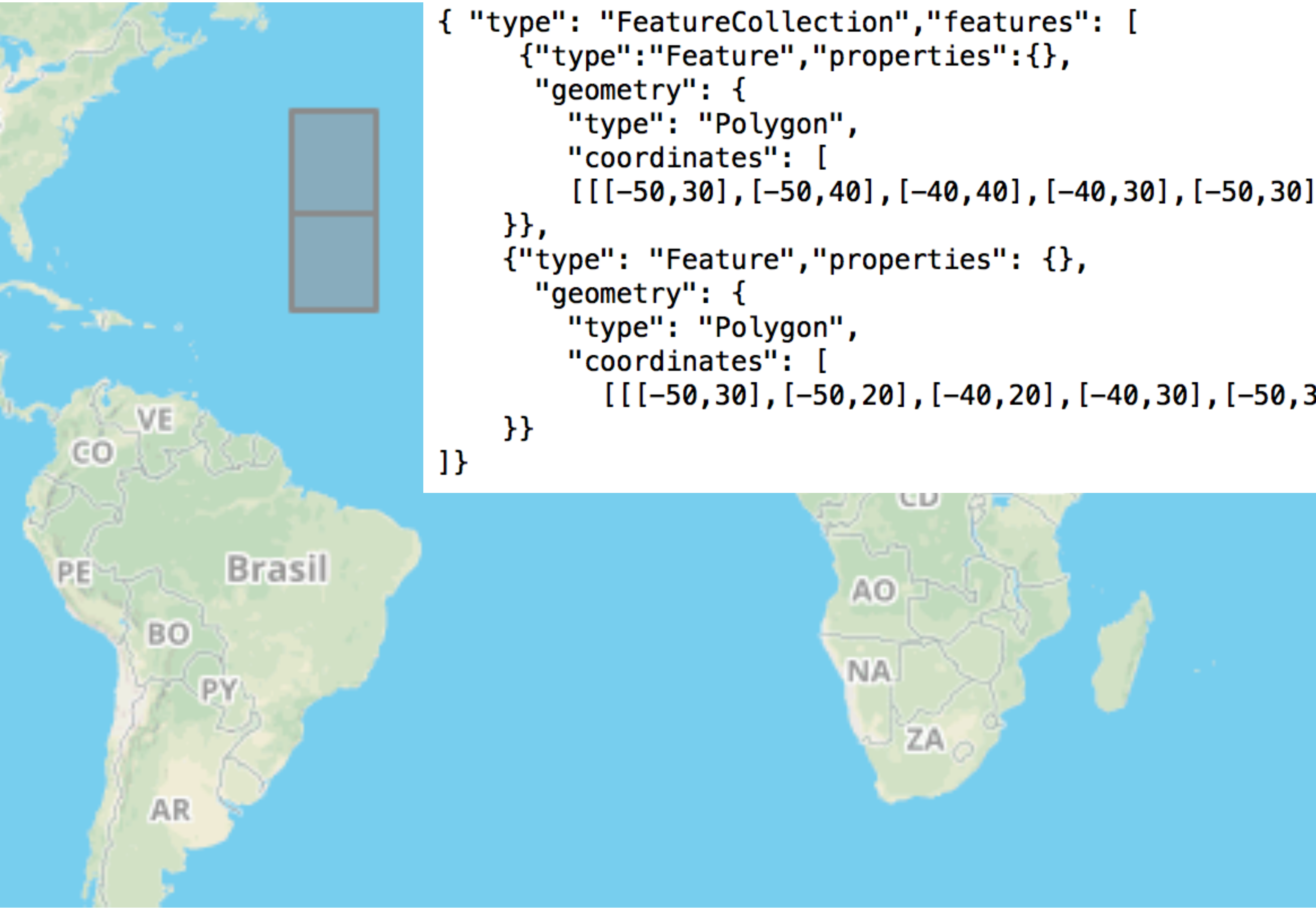
Lines would be defined like this:



ID	Coordinates
1	(0,1), (3,4), (5,6)
2	(3,1), (5,2), (4,3)

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/

Spaghetti vector models

Advantages

- Simple and robust
- Thus tend to be easy to use to exchange data

Disadvantages

- Doesn’t tell us anything about the relationships between different features…
  (do two lines overlap? Are two polygons adjacent to each other?)
    - Less efficient for many types of spatial analysis
    - Can be too unstructured / Allows for ‘errors’

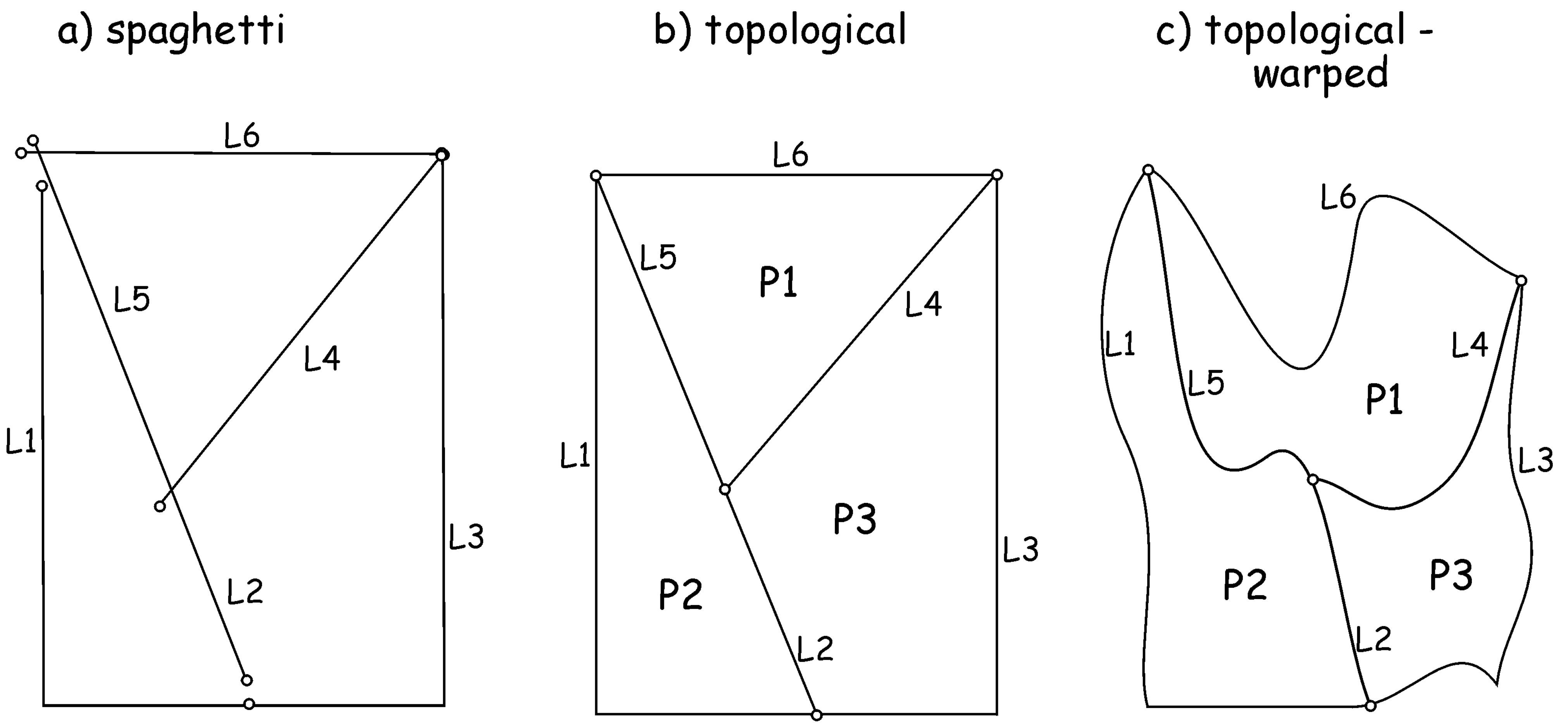
Implementing the vector data model: Topological vector structures

- 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.

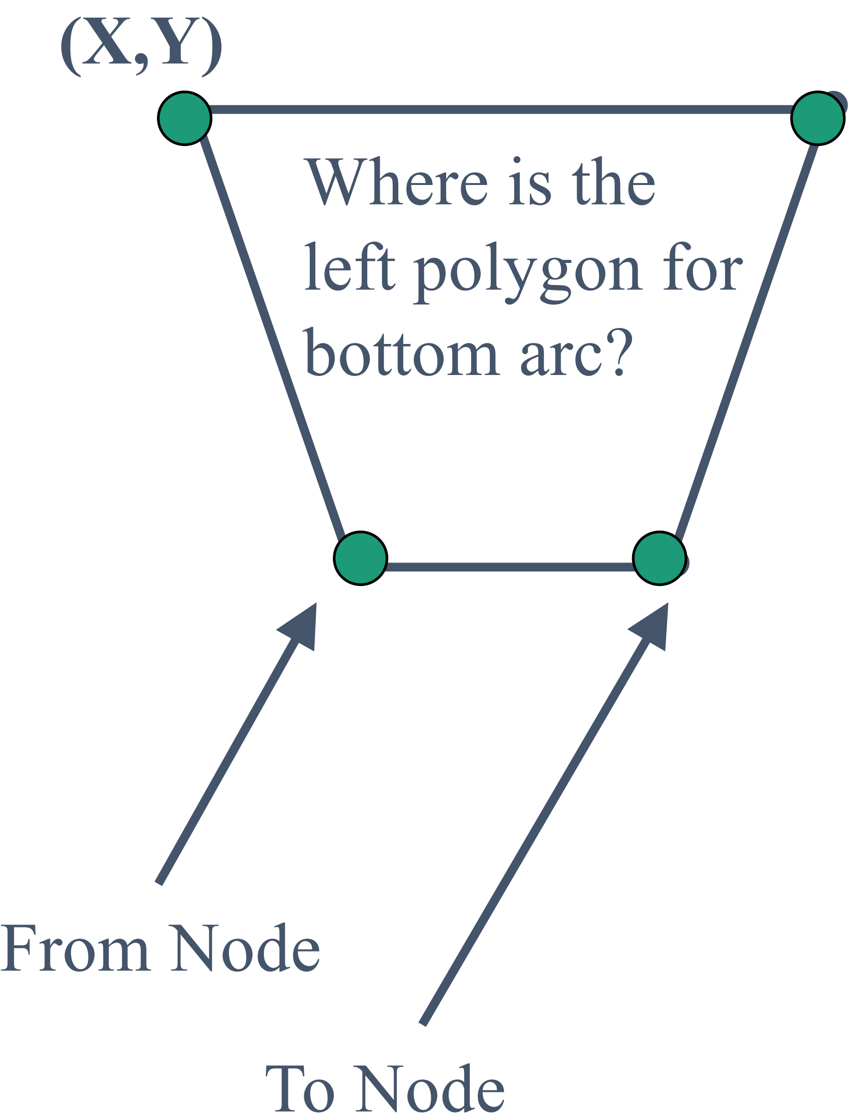
  (e.g., the data structure has stored the fact that two polygons share the same line in their borders.)



- Figures b) and c) are topologically equivalent, in a GIS sense: they have the same connectivity and adjacency.

- (When you switch projections, those relationships remain constant.)

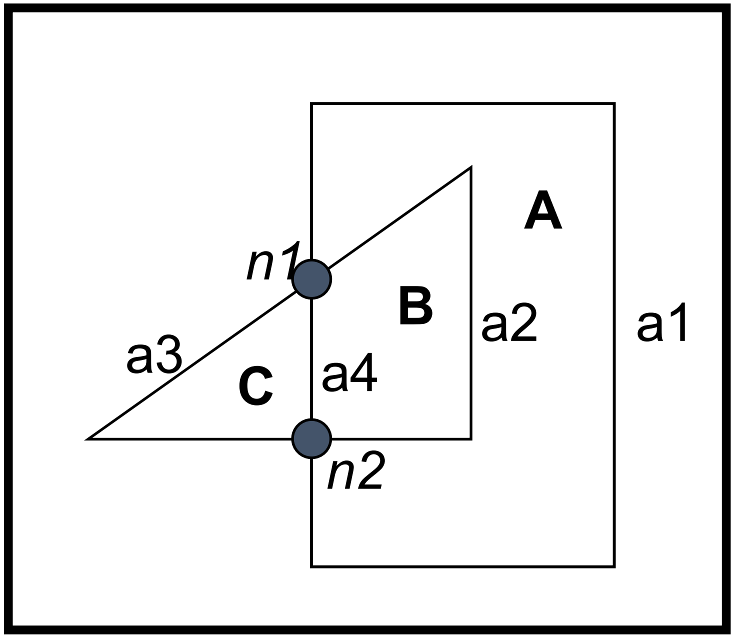
With topological data models, the data encode relationships:



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

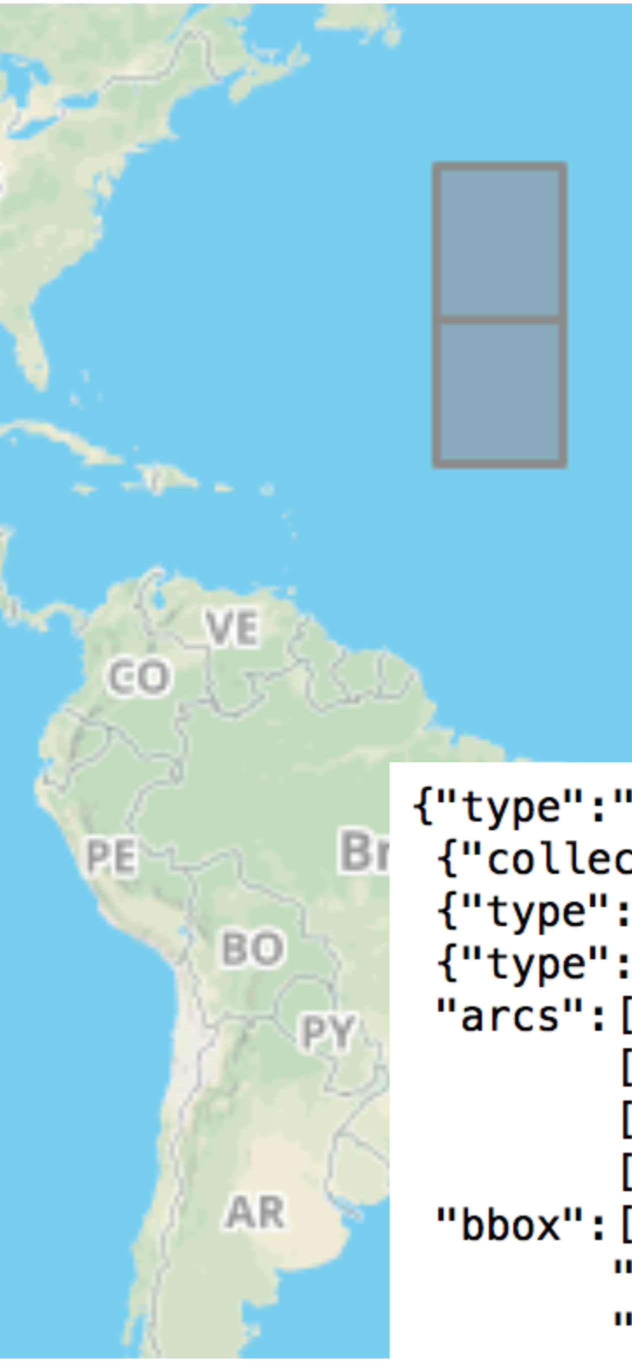
Node	Arcs
n1	a4, a2, a1, a3
n2	a2, a4, a3, a1

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]}}
```

TopoJSON: Topological

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

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

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.

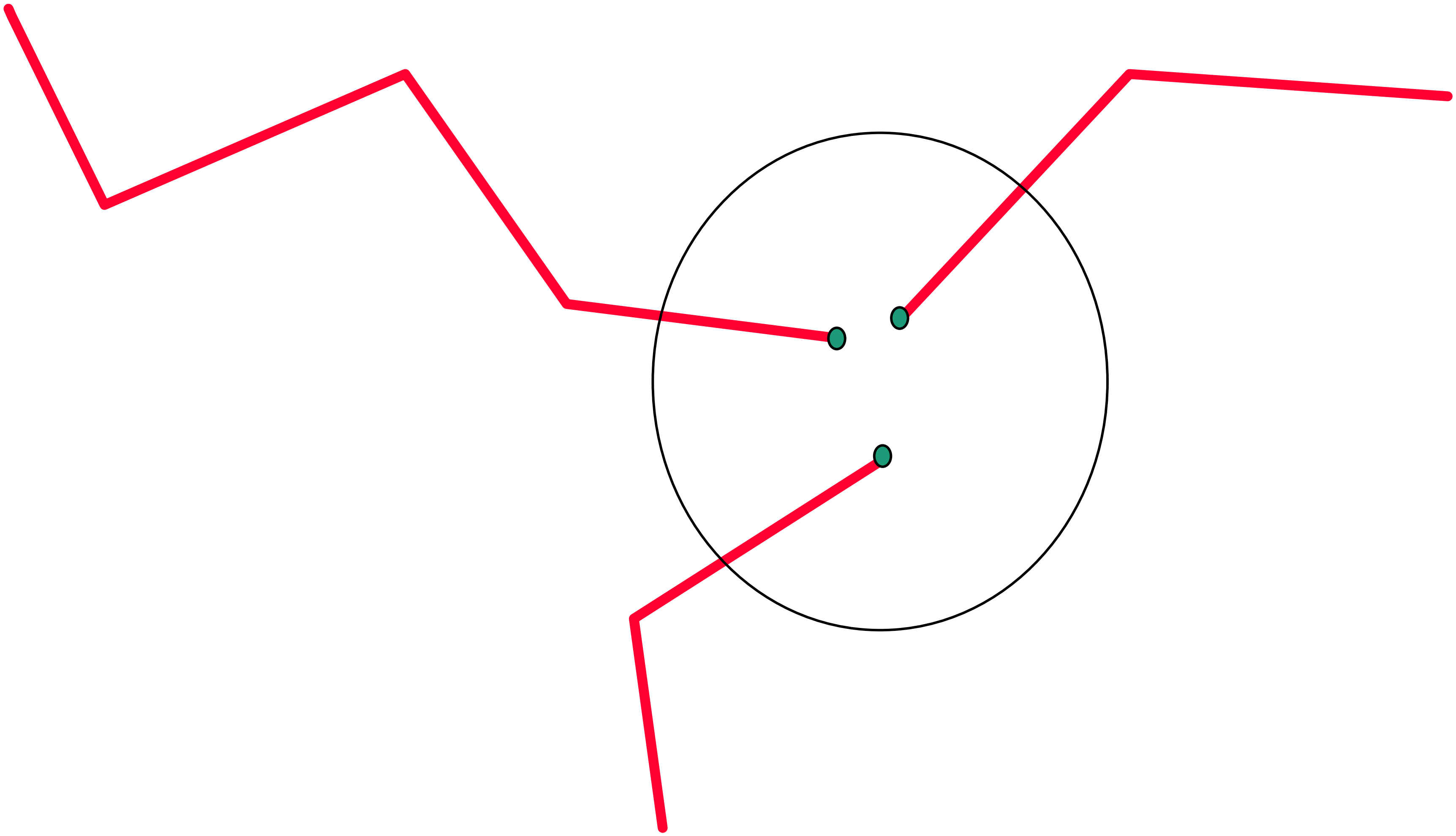
   (Can you think of what procedure you would tell the computer to follow in order to figure that out, otherwise?)

Examples of topological rules

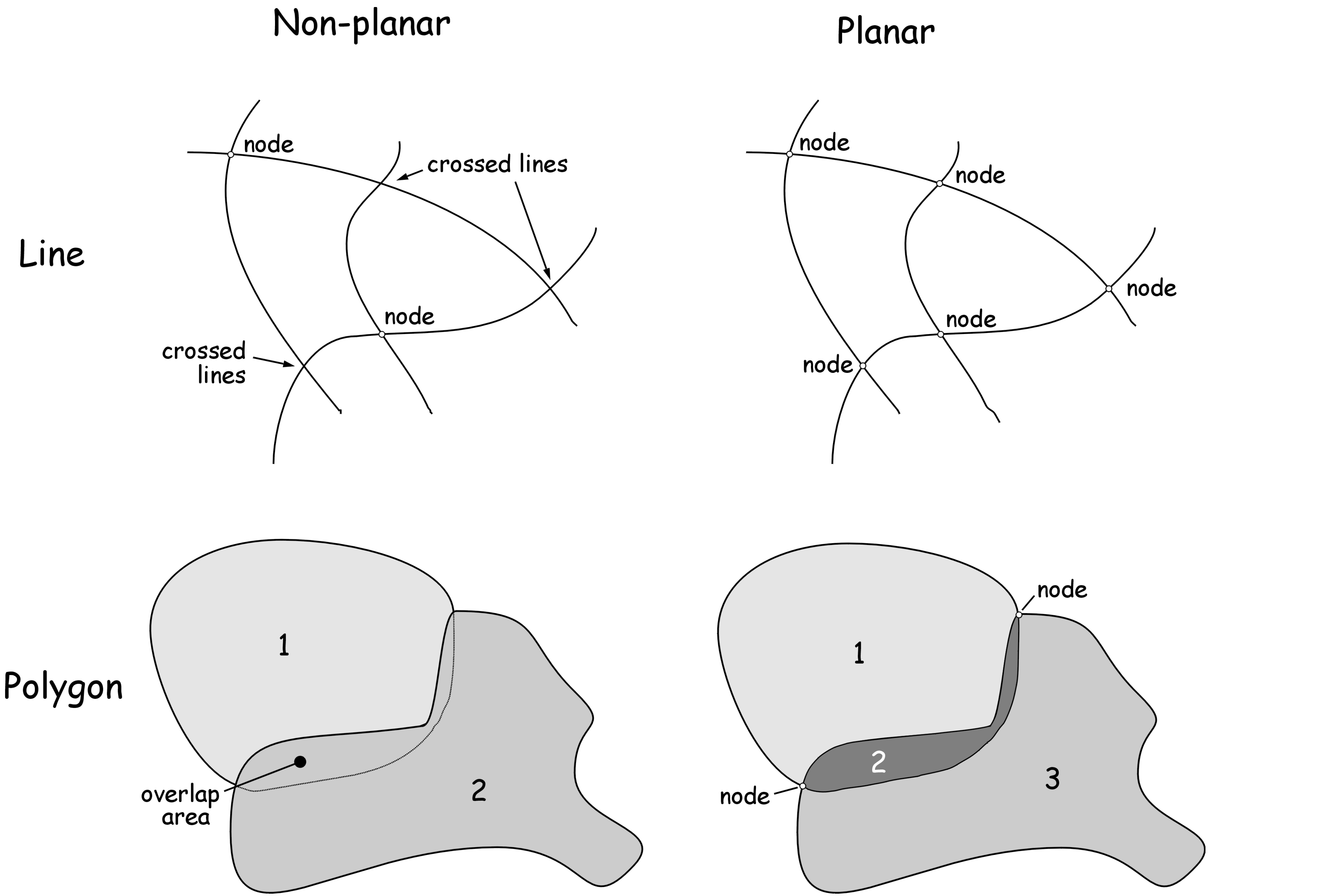
- No two individual features can overlap within a layer.

- Each feature in a given layer must lie within a single feature in a second layer (e.g., counties within states).

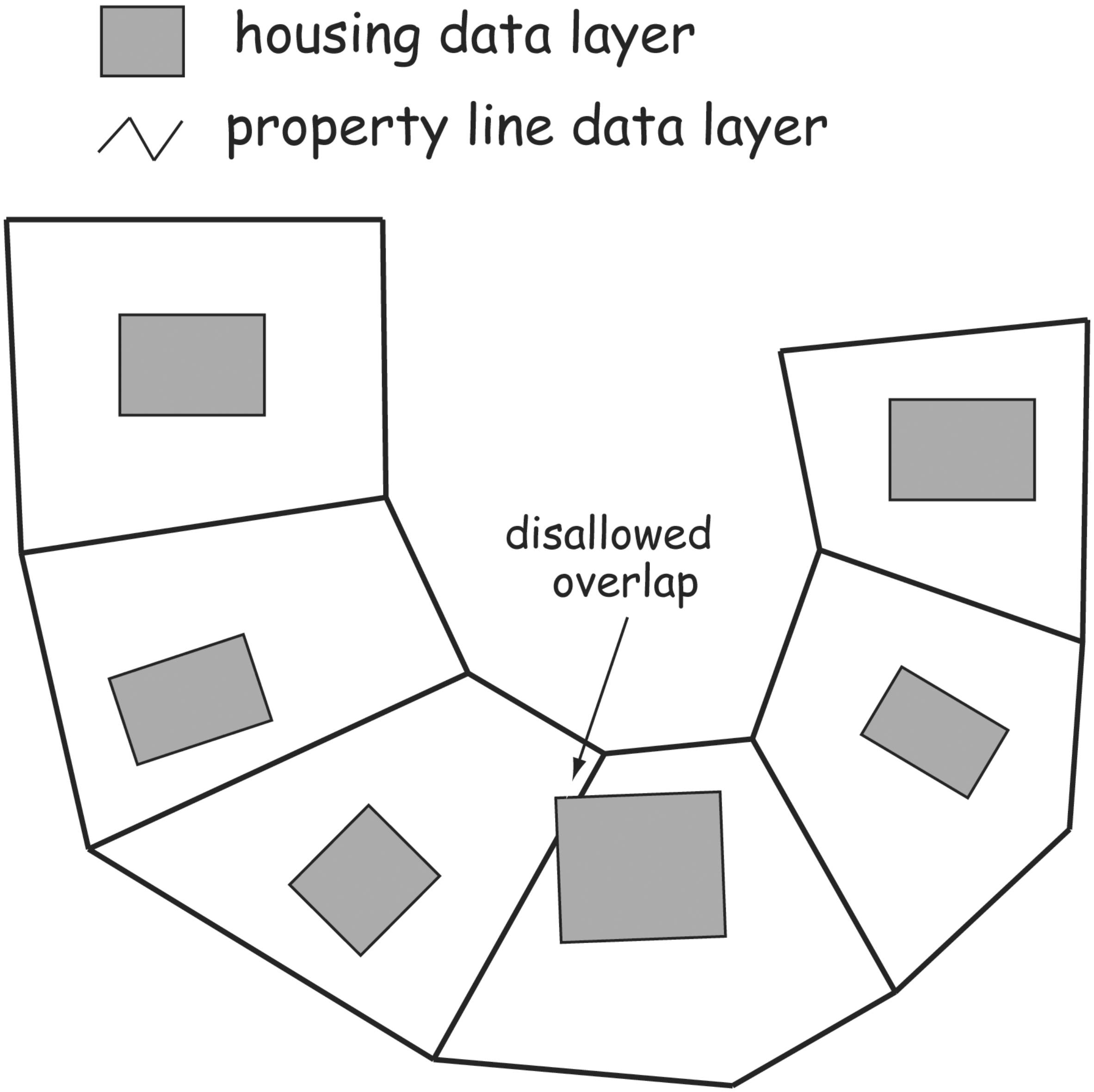
Topological rules can be chosen to exclude: unsnapped nodes



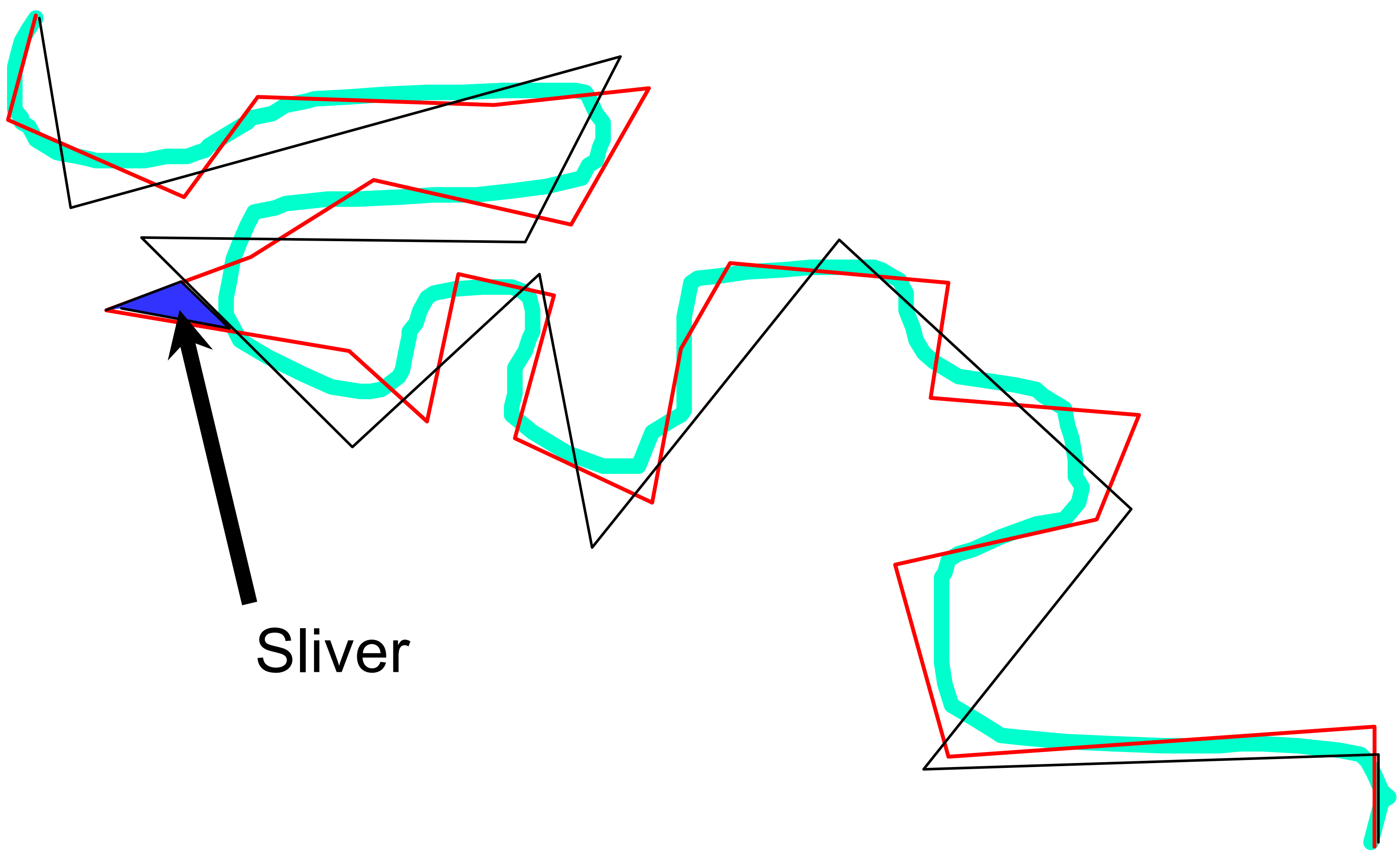
Enforcing a rule that the topology must be 'planar'



Enforcing topological rules across data layers

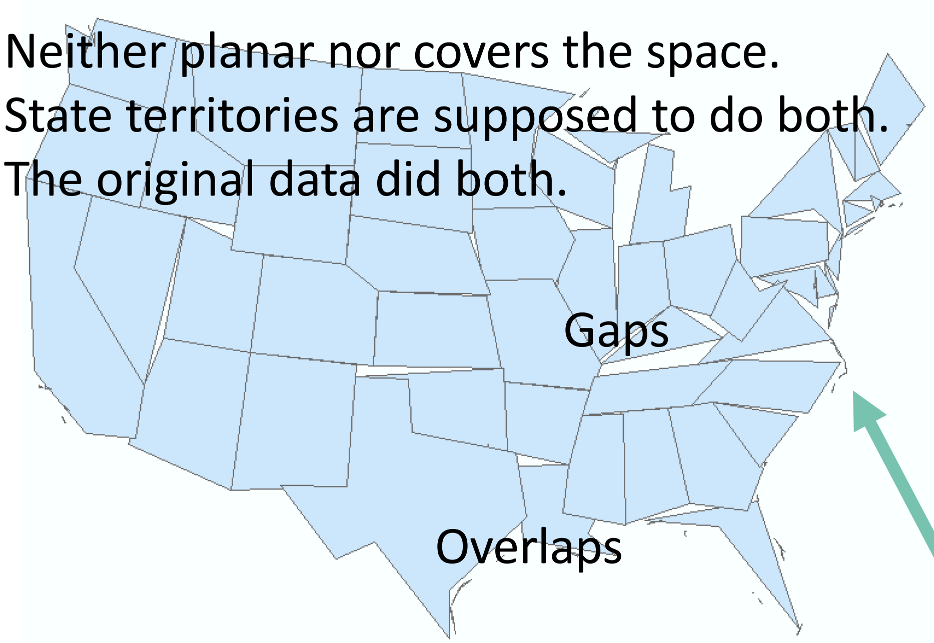


Topological rules can thus be used to exclude 'slivers.' Slivers can be introduced by your analyses, often by combining data from multiple sources.

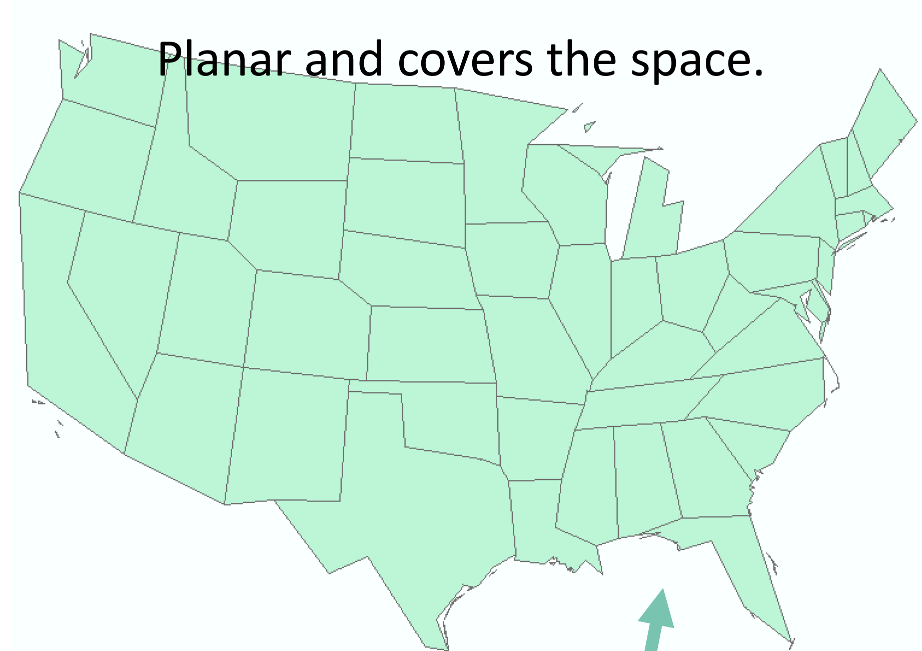


When the GIS gives you an option, have topological awareness work in your favor…

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

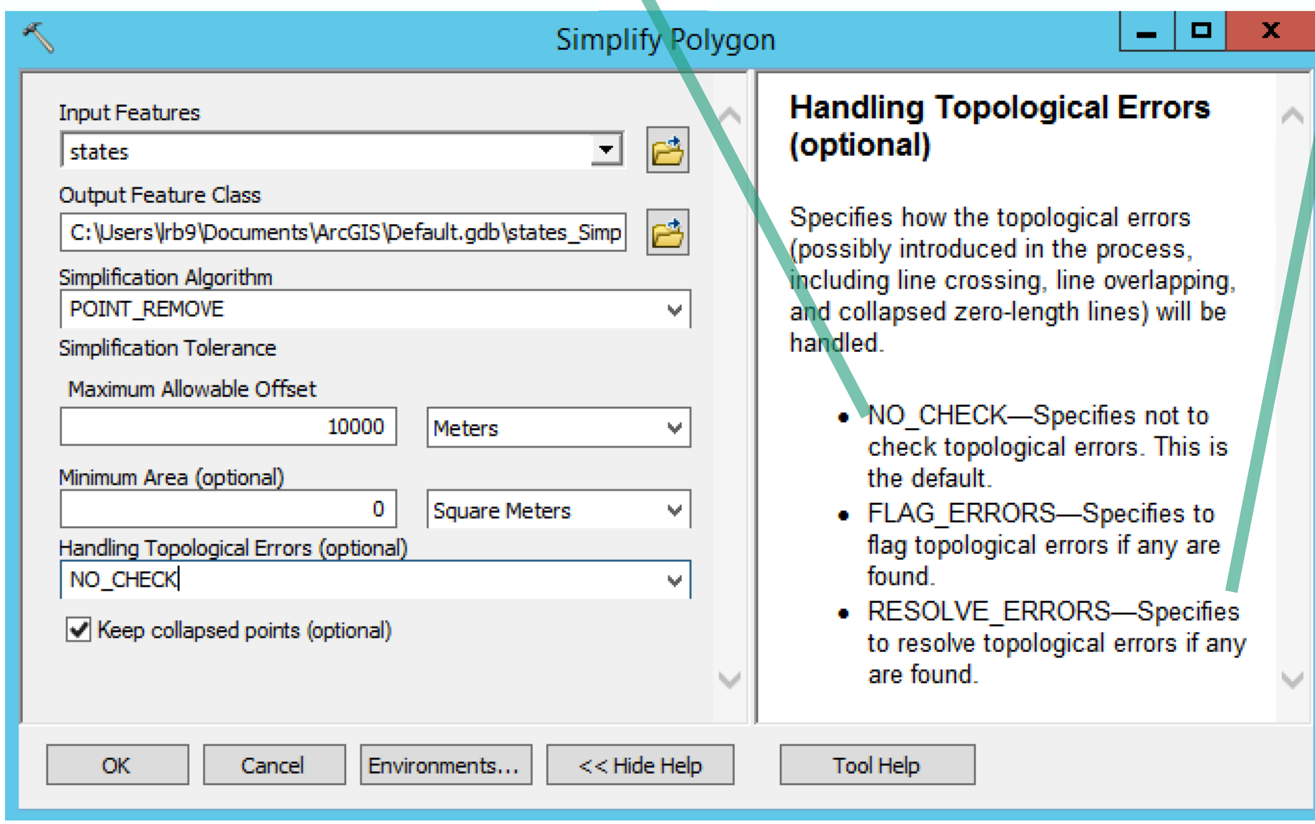


Planar and covers the space.

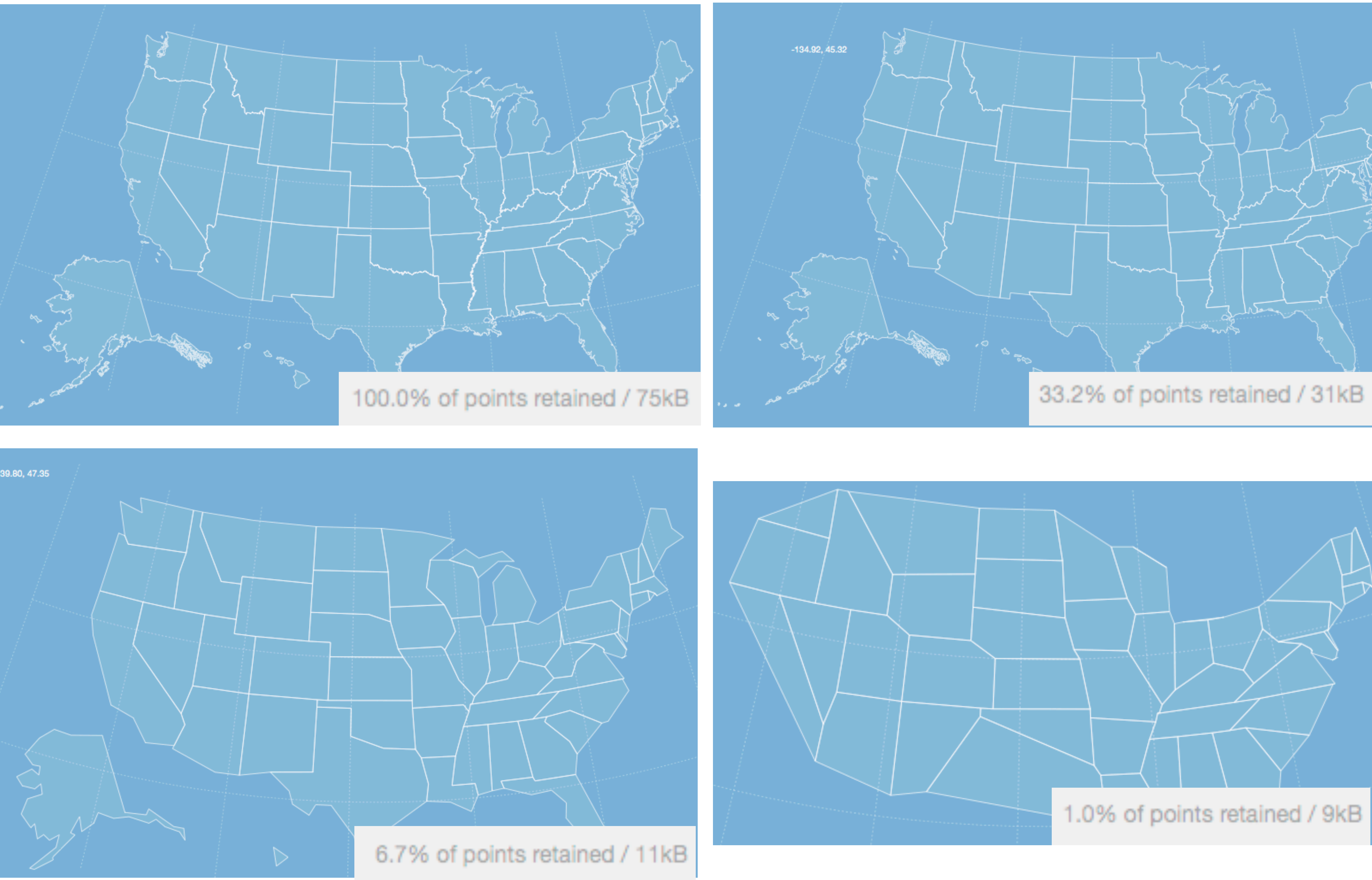


On the right side simplification, topology has not been stored in the data structure… **but** it is being calculated and checked for whether it conforms to an ideal (and if not, resolving the 'errors').

Less efficient, but still very useful.



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

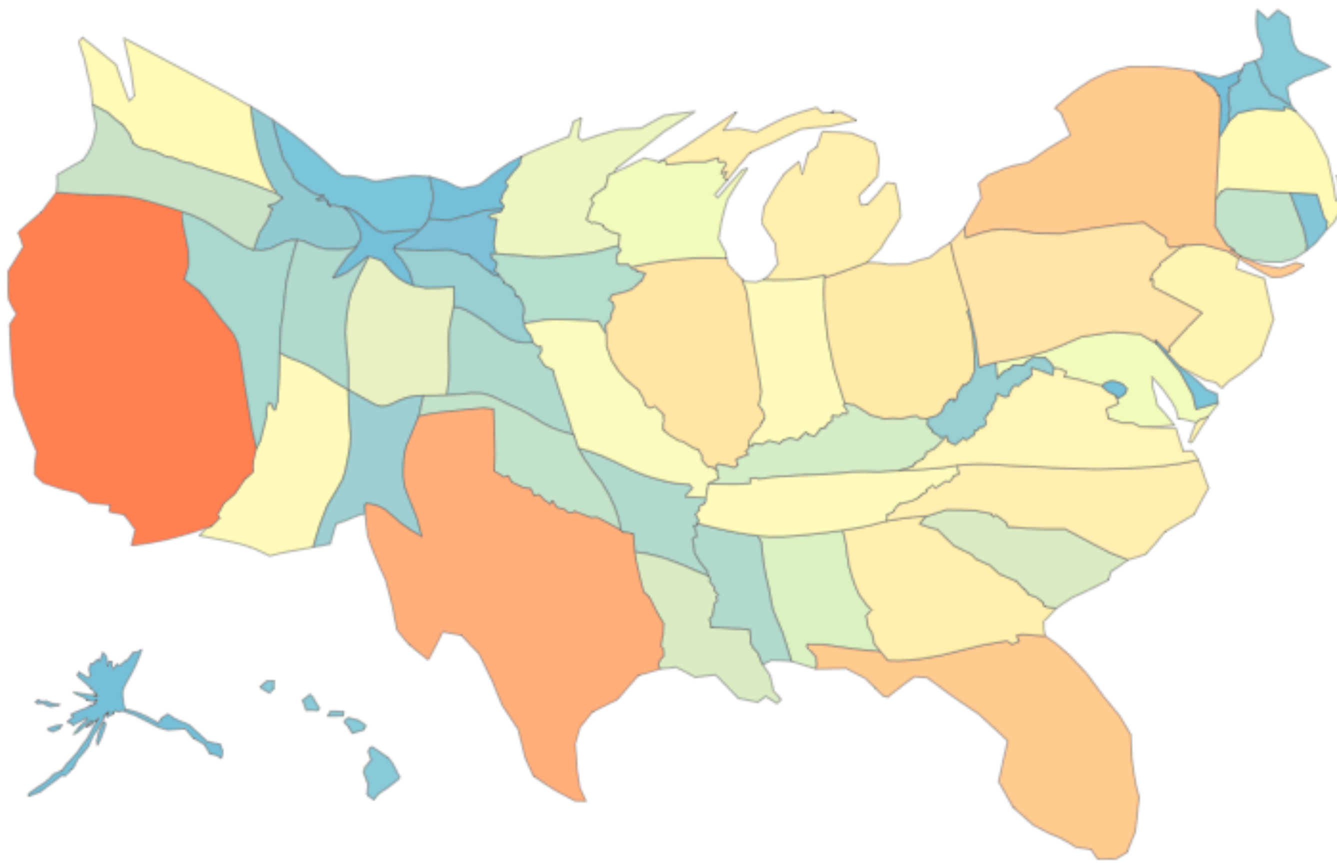


https://shancarter.github.io/distillery/

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 guess 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.

B. Finding the fastest road route from one city to another.

C. Finding places where the rain is greater than 30 inches/year and the elevation is less than 3000 feet.

D. They would all be helped!

Quiz I: Next 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.

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 Monday. 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.