Representing the real world in a GIS: how geographic information is stored in the computer

Representing the real world in a GIS

- the world is infinitely complex
- the contents of a spatial database represent a limited view of reality
  -> the spatial database is a **model of reality**
- the user sees the real world through the medium of the database

Representing the real world

- a database may include

  - digital versions of real objects
  e.g., houses, roads, forests

  - digital versions of fictitious (i.e., invented) objects
  e.g., political boundaries

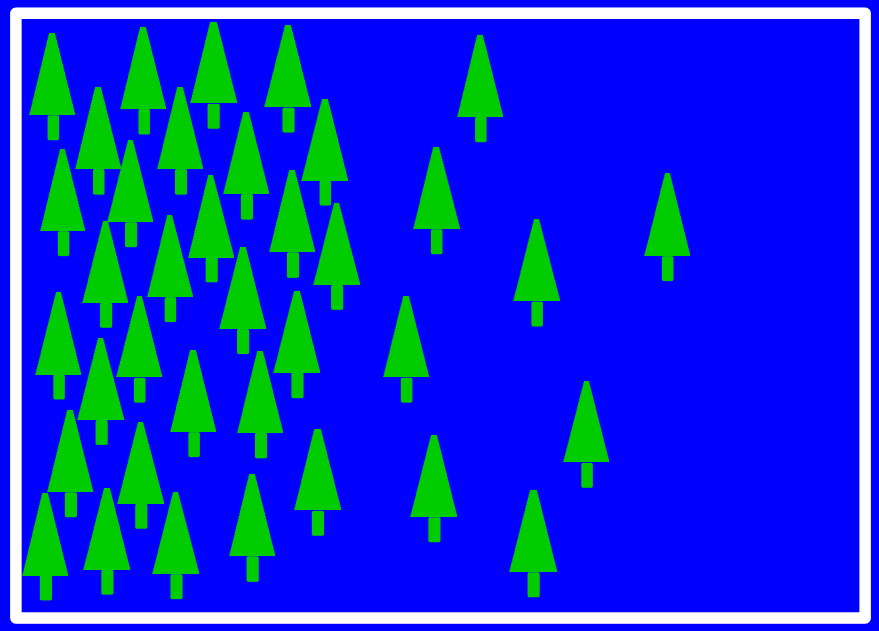
Representing the real world

- computers are good at storing discrete spatial data, but bad at storing continuous data

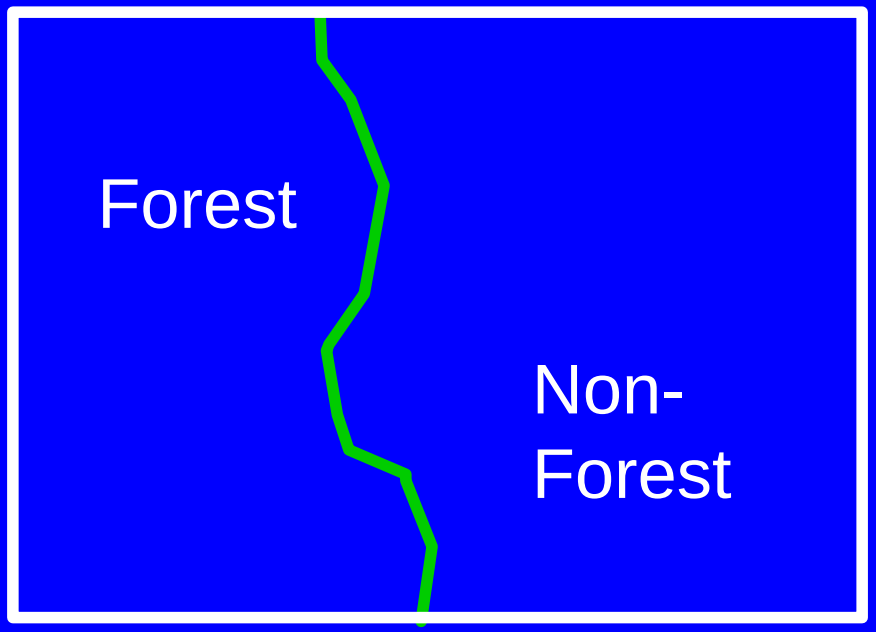
Representing the real world

- **some features are discrete, clearly defined entities (e.g., houses, districts)**
  -> discrete representation is no problem

- **other features exist everywhere and vary continuously (e.g., temperature)**
  -> variation needs to be approximated using discrete representations



reality



GIS representation

- **sometimes, the distinction between discrete and continuous is not very clear**

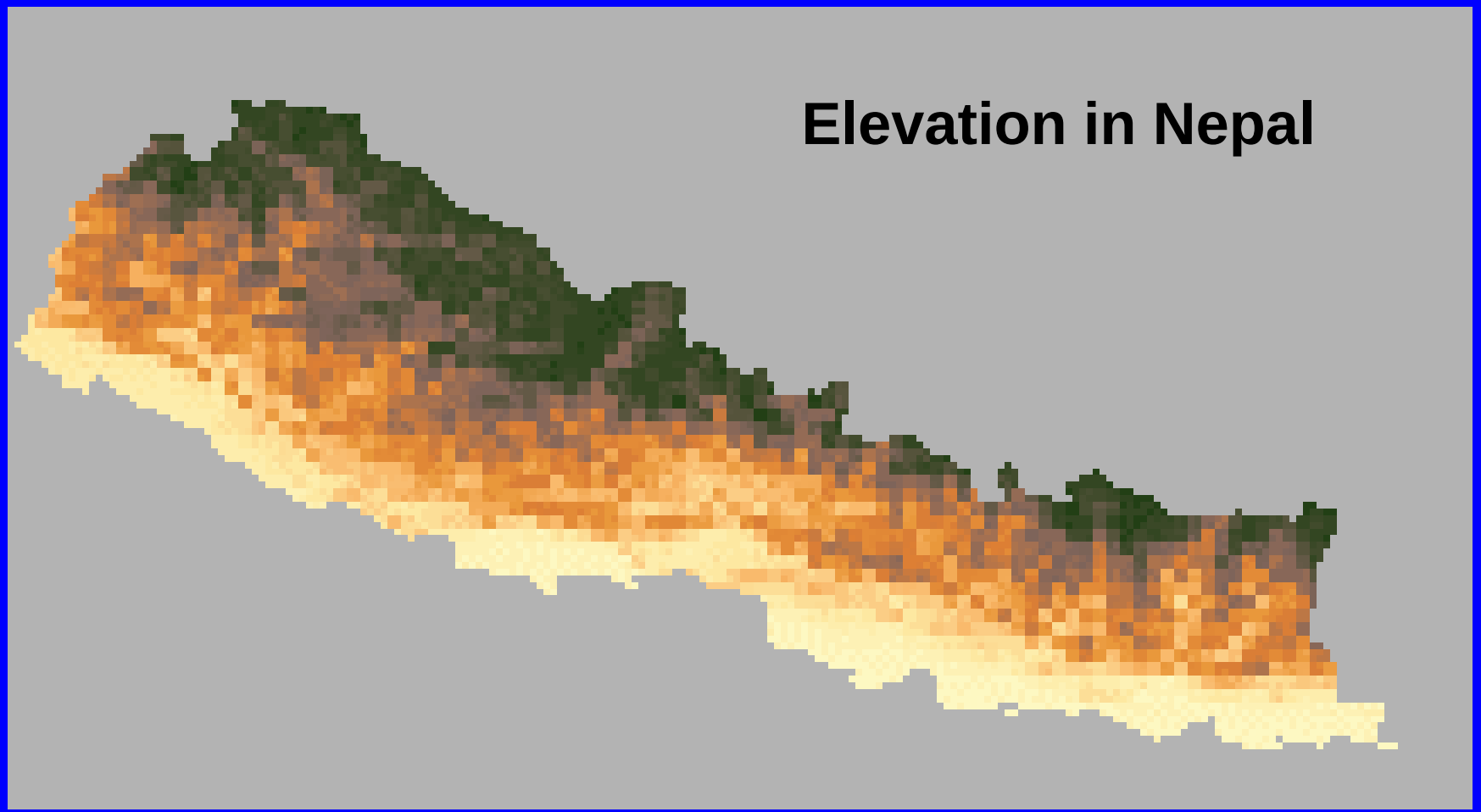
Objects versus Fields

- **Object view**
  “empty space littered with objects”
  (points, lines or areas)

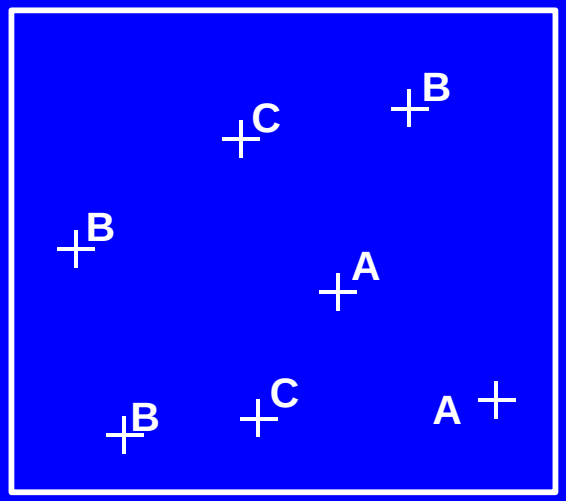
- **Field view**
  value is defined for every location

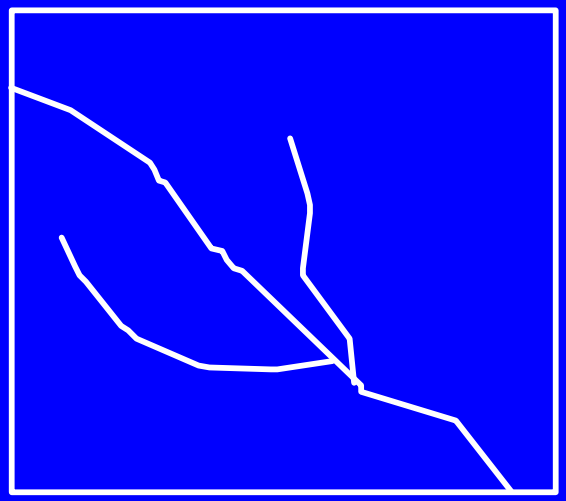
Example of a field

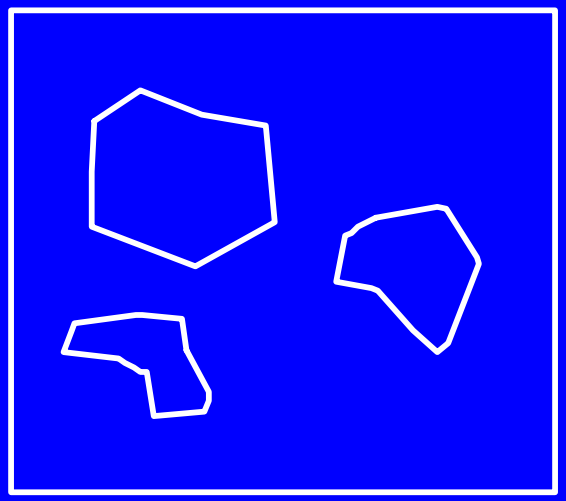
- digital elevation models (DEMs)



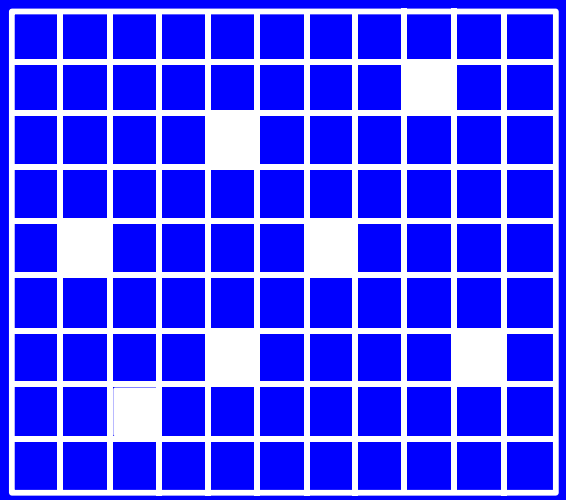
Objects



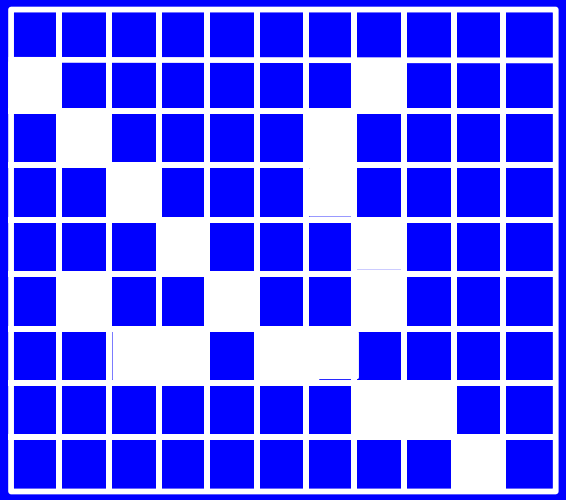




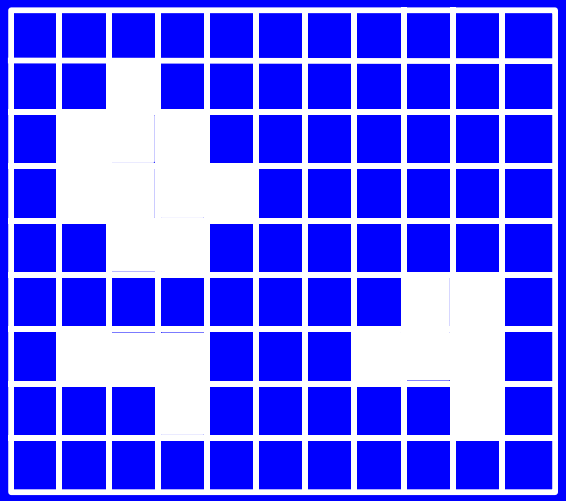
Points



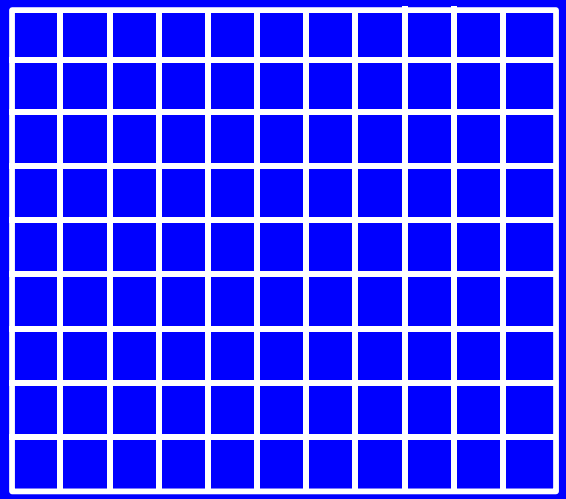
Lines



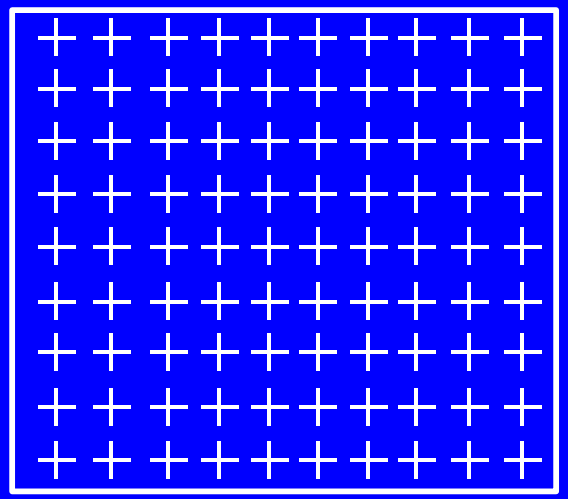
Polygons



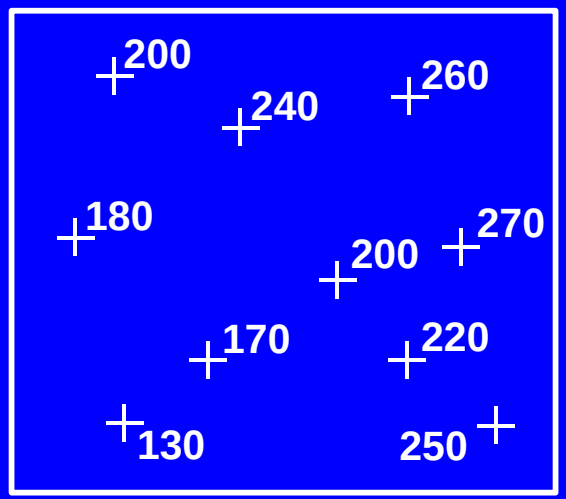
Fields



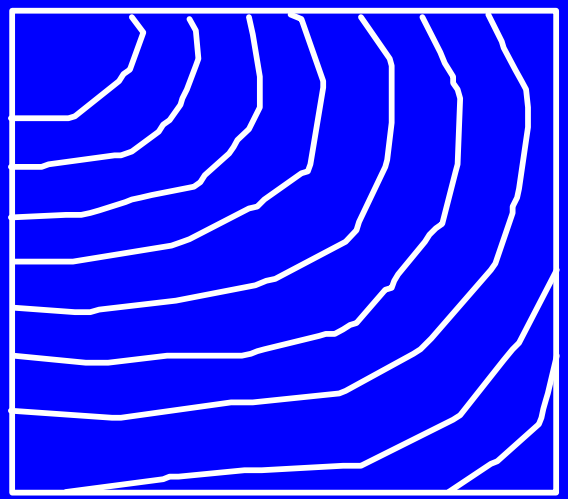
Raster grid



Regular point grid



Irregular points



Contour lines

Data model implementation: the vector data model

The data model

- rules to convert real geographic variation into discrete representations

GIS data models

- two major types:

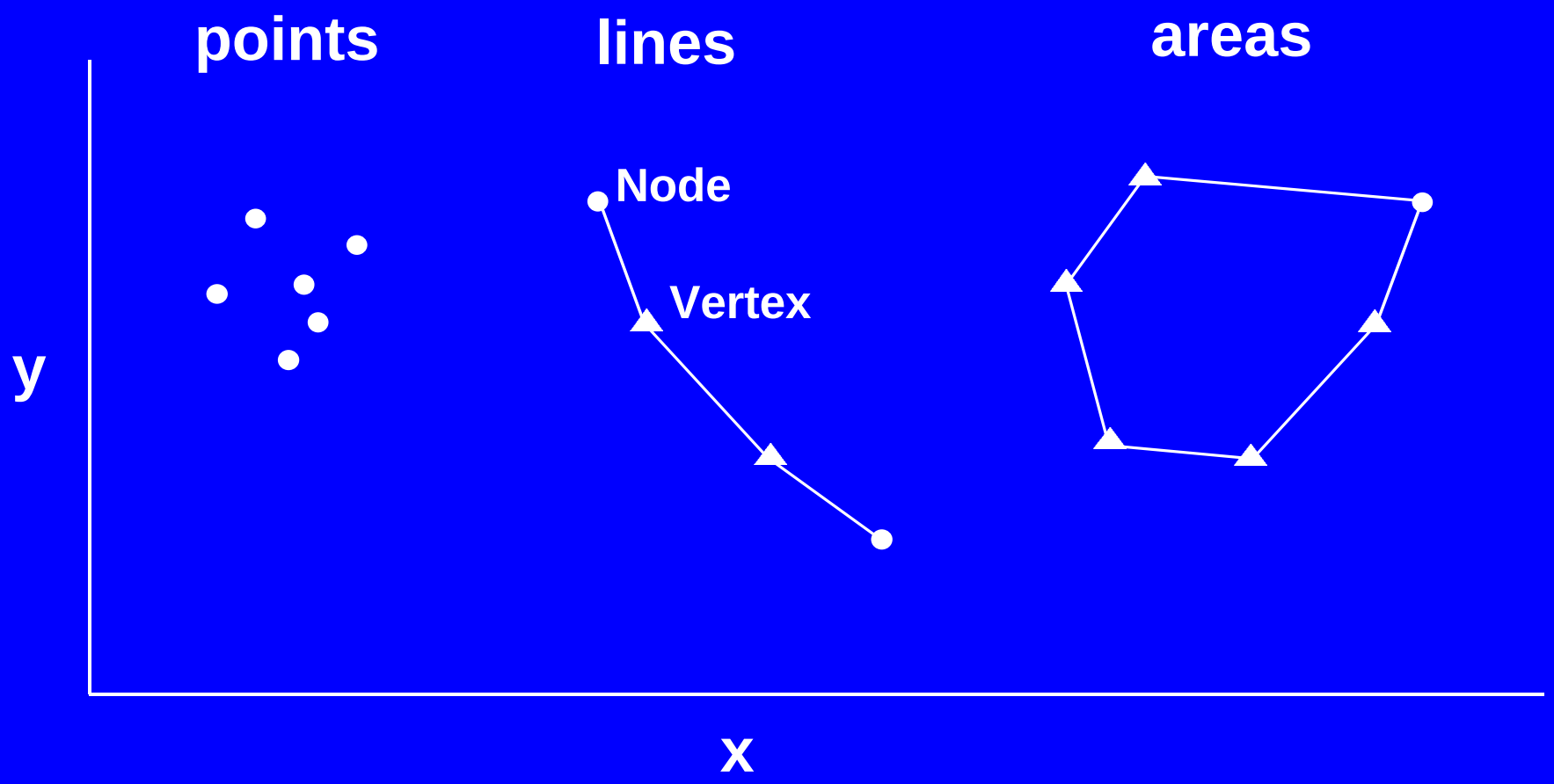
  - raster data model
  - vector data model

- raster data model will be discussed later

The vector data model

- real world objects are represented as points, lines and areas
- points identify locations
- lines connect points
- areas (polygons) consist of connected line segments

The vector data model



The vector data model

- objects are defined by their x/y coordinates in the planar (Cartesian) coordinate system
- precision of coordinates virtually infinite (only machine-dependent)
- but: accuracy most often limited!

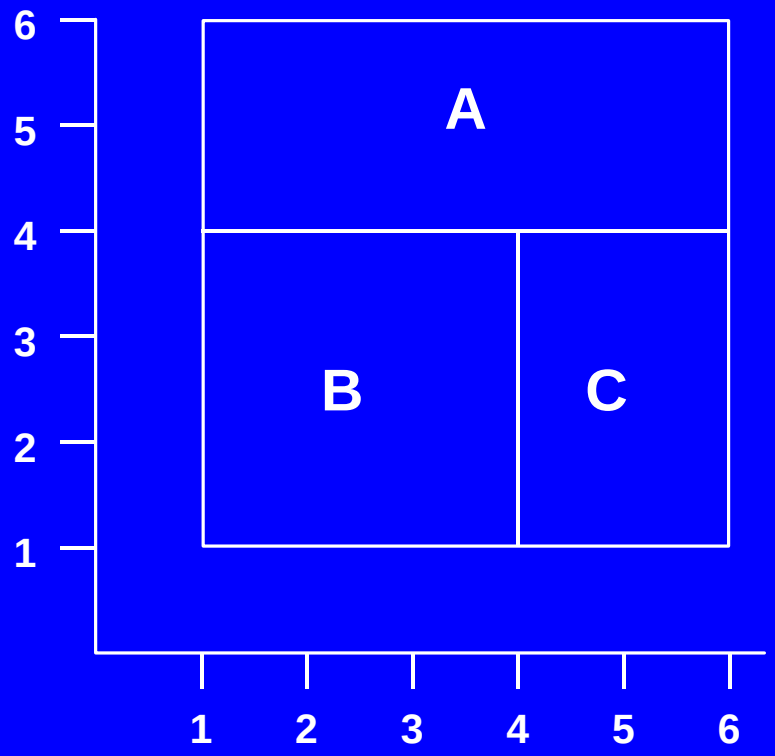
Precision versus Accuracy

- **precision** is the ability to distinguish between small quantities or distances in measurement
- **accuracy** is freedom from error

“Spaghetti” data model

- **point** is recorded as x,y coordinate pair
- **line** is a series of x,y coordinates
- **area** is a series of x,y coordinates, with the first and last coordinate being identical (e.g., “closed-loop polygons”)

Area	Coordinates
A	(1,4), (1,6), (6,6), (6,4), (4,4), (1,4)
B	(1,4), (4,4), (4,1), (1,1), (1,4)
C	(4,4), (6,4), (6,1), (4,1), (4,4)



“Spaghetti” Data Model

Points and lines would be encoded in a similar way; note: there is no relationship between points, lines and areas

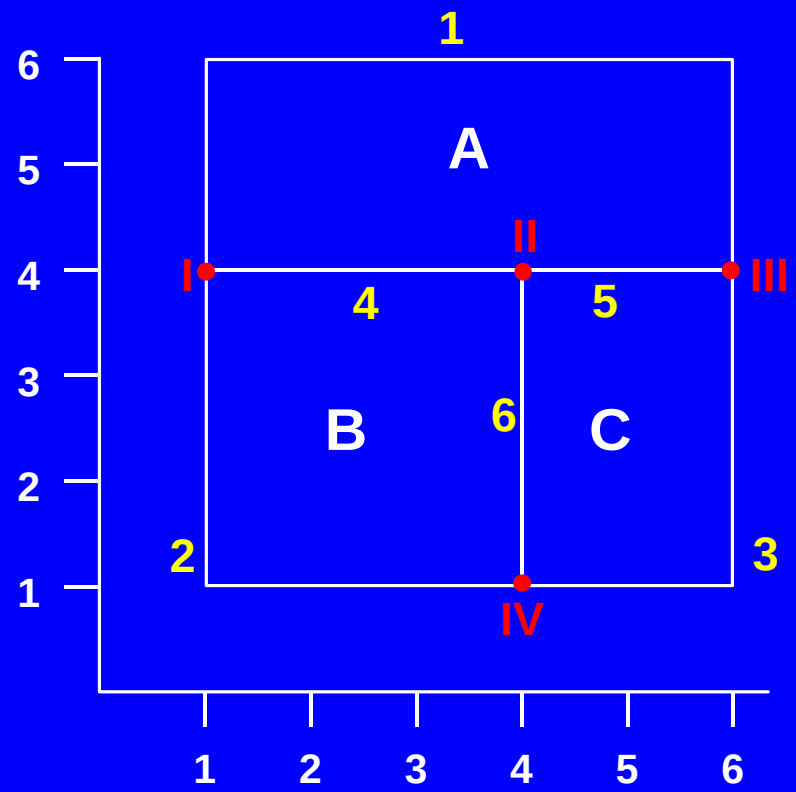
Topological data model

- records x/y coordinates of spatial features *and* encodes spatial relationships
- also called “arc-node” data model
- arc = line
- node = end-point of a line, or a point where two or more lines connect

Node	X	Y	Lines
I	1	4	1,2,4
II	4	4	4,5,6
III	6	4	1,3,5
IV	4	1	2,3,6

Line	From	To	Left	Right
1	I	III	O	A
2	I	IV	B	O
3	III	IV	O	C
4	I	II	A	B
5	II	III	A	C
6	II	IV	C	B

O = “outside” polygon



Poly	Lines
A	1,4,5
B	2,4,6
C	3,5,6

Topological Data Structure

Topological data model

can quickly answer these questions:

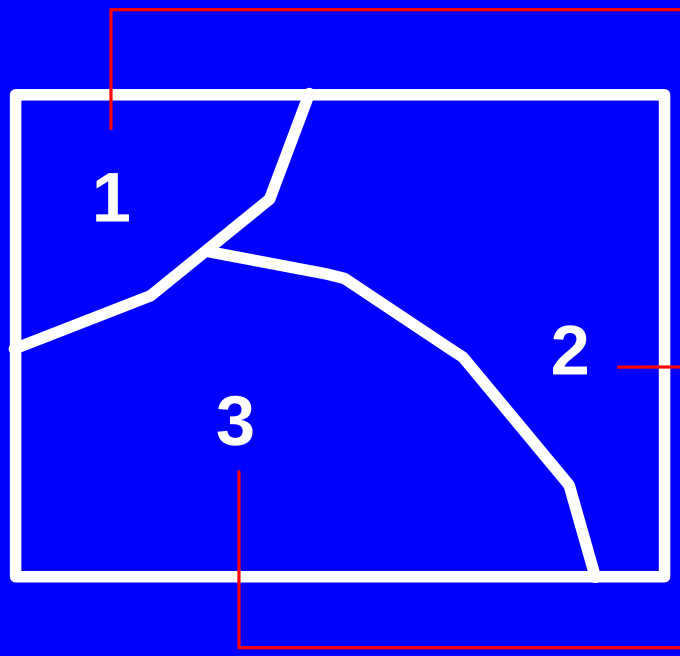
- **which roads connect to the central square?**
- **which roads do I take to get from here to the hospital?**
- **what are the fertility rates in the neighboring districts?**

Storing attribute data

- attribute data are stored separately from the coordinate data

- feature identifier points to an attribute table:
  - point attribute table
  - line or arc attribute table
  - polygon attribute table

Storing attribute data

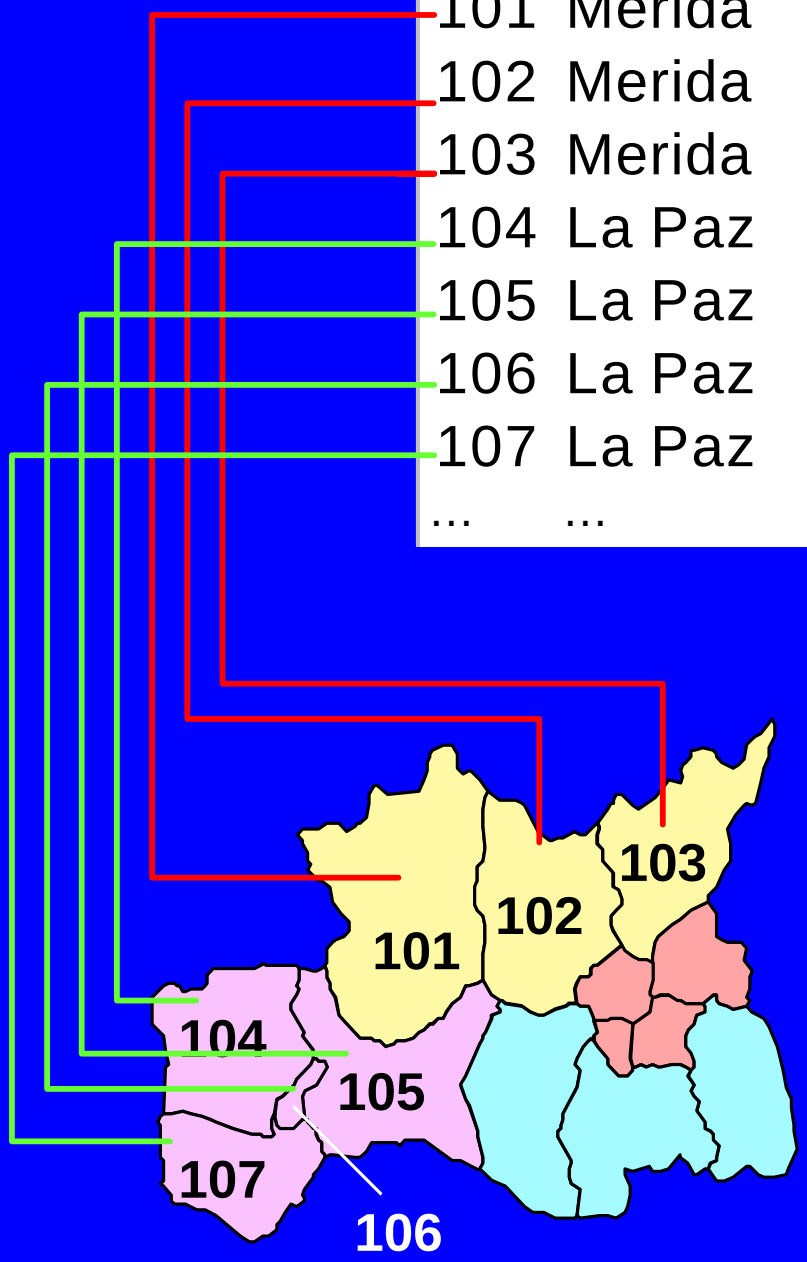


polygon attribute table

poly id	area	pop	TFR
1	297	4583	2.7
2	607	3927	3.1
3	806	9271	3.8

similarly we can define point or line attribute tables if the spatial features are, for example, villages and roads

Id	Province	District	P_Pop	P_TFR	D_Pop	D_TFR
101	Merida	Palma	214084	3.2	89763	3.4
102	Merida	S. Maria	214084	3.2	45938	2.9
103	Merida	Veralo	214084	3.2	78383	3.2
104	La Paz	Bolo	397881	3.7	98302	3.9
105	La Paz	Jose	397881	3.7	67352	4.2
106	La Paz	Malabo	397881	3.7	102839	3.7
107	La Paz	Chilabo	397881	3.7	129388	2.8
...	...	...	...	...	...	...

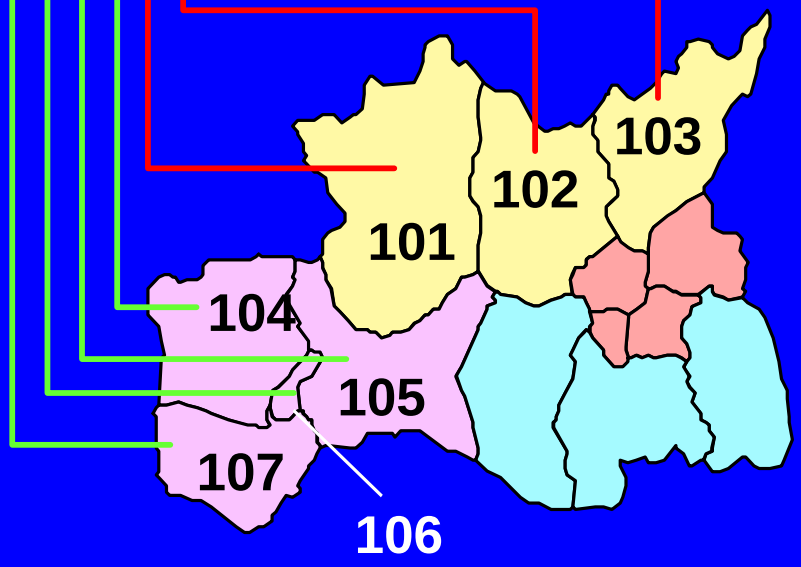


storing the province and district data in the same table is inefficient, because province data need to be repeated for each district

Storing attribute data

- instead we can produce a more efficient database that does not include as much redundancy
- relational database
- process to separate variables into several files is called normalization

Id	District	D_Pop	D_TFR	Province
101	Palma	89763	3.4	Merida
102	S. Maria	45938	2.9	Merida
103	Veralo	78383	3.2	Merida
104	Bolo	98302	3.9	La Paz
105	Jose	67352	4.2	La Paz
106	Malabo	102839	3.7	La Paz
107	Chilabo	129388	2.8	La Paz
...	...	...	...	...



P_Pop	P_TFR	Province
397881	3.7	La Paz
214084	3.2	Merida
...	...	...

a relational database design provides better storage efficiency

Storing attribute data

- good organization of the attribute data is very important
- in socioeconomic GIS applications, the attribute data component is often much larger than the database component; e.g., few provinces, but hundreds of variables