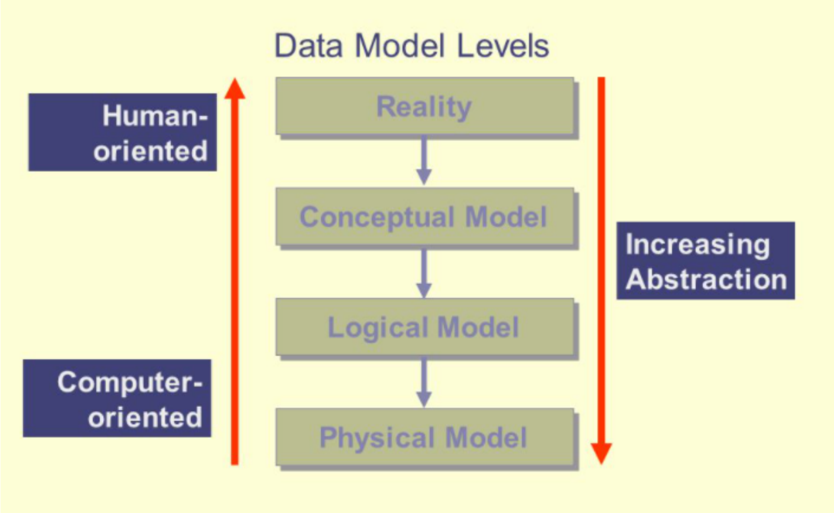
Ekonomická fakulta
Faculty of Economics
Jihočeská univerzita v Českých Budějovicích
University of South Bohemia in České Budějovice

Geographic Information Systems 1
Lecture 3: vector data

Renata Klufová
University of South Bohemia in České Budějovice, Faculty of Economics

March 2021


Data Model Levels
Human-oriented
Computer-oriented
Reality
Conceptual Model
Logical Model
Physical Model
Increasing Abstraction

... first used in CAD systems - formats: DXF, DWG (Autodesk), DGN (Intergraph), IGES (Initial Graphics Exchange Specification), HPGL (Hewlett Packard Graphic Language)

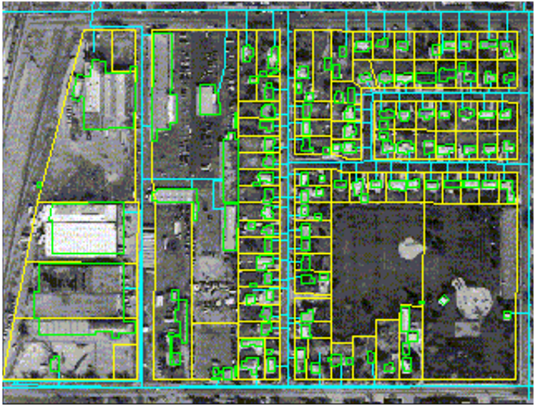
The essence of vector data is the expression of the geometric part of the description of the geo-element using linear characteristics. All geographical phenomena can be represented in two-dimensional space by three types of formations:

- **points**,
- **lines**,
- **areas** (polygons),

extended concept of areas and lines:

- **networks** (interconnected lines along which the flow of information takes place),
- **surfaces** (contiguous entities for which there is an indication or value at each point, which may be of a quantitative or qualitative nature).

Try to create a simple map of the real area from this aerial image using points, lines and areas.



What types of networks can be identified in the image?

basic relations between elements - **graph theory**:

- **nodes**,
- **edges** - each edge is a connection of two nodes, if the order of the nodes is important when defining the edges, it means that the edge has direction - **oriented** graphs.

The direction of the edges and the subsequent orientation of the graph is important for several reasons - **topological relations**:

- oriented edges are a means of defining **orientation**,
- direction of line objects,
- for the description of **adjacency** (joints) of areas on the border - to the left and to the right of the edge,
- the direction needed to describe the areas bounded by the lines of the planar graph.

Two edges are **adjacent** if they have a common node.

- If the node exists in m edges, then the degree - **order** of this **node** is $D(n) = m$.
- Therefore, if the order of the node $D(n) = 0$, the node does not exist on any connector, it is **isolated node - point**.
- **chain** denotes a sequence of adjacent edges if it meets the following conditions:
  - each edge occurs only once in a certain chain,
  - there are at most two nodes that occur on only one edge of the string. Their order is then $D(n) = 1$ and it is the start and end node of the string.
  - the other nodes in the chain occur exactly in two edges, their order $D(n) = 2$.

The closed string is called **polygon**, while there are no nodes in the polygon with the order $D(n) = 1$, all have the order $D(n) = 2$.

Two nodes in a graph are connected if there is a string in which both occur. A graph is connected if all possible nodel pairs are connected. By combining multiple polygons, we can build a **polygon network**. When building it, we must pay attention to:

- non-duplicate designation of vertices and edges,
- inclusion of all necessary points and lines,
- distinguishing right and left areas for oriented maps,
- taking into account the ”outdoor” area.

If the network of polygons consistently covers the entire examined area = **area partitioning** - checking the completeness and uniqueness of the partition - so-called **Euler's equations** - they monitor the relationship between the number of nodes, edges and polygons.

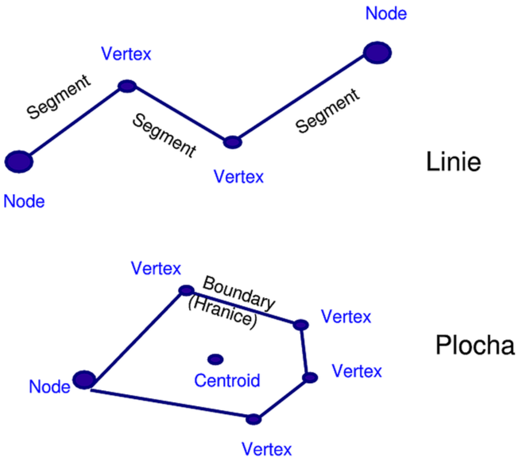
Euler's rule: there is a fixed relationship between the number of facets f, the number of nodes n and the number of edges e in the graph:

$$f + n - e = 2$$

The rule in this basic form applies to connected graphs with surfaces without holes. For unlinked graphs it is necessary to modify it:

$$f + n - e = c + 1,$$

where c is the number of discontinuous subgraphs.

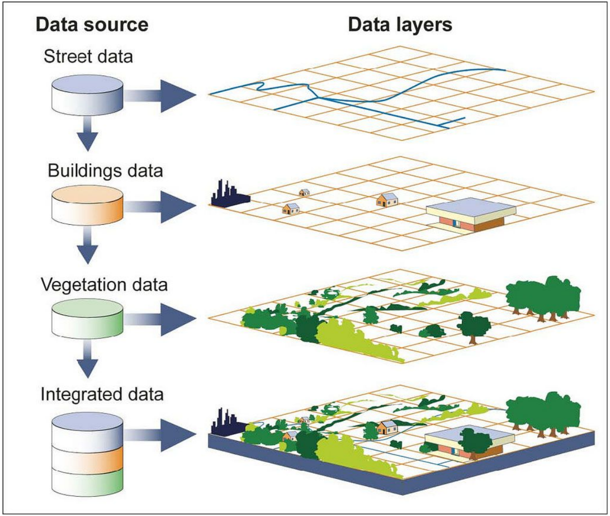

Node
Vertex
Segment
Segment
Segment
Vertex
Node
Linie
Vertex
Boundary
(Hranice)
Vertex
Centroid
Node
Vertex
Vertex
Plocha

- If the represented line objects are connected, we must define their continuity at intersections - nodes. This relationship is called **connectivity**.

For represented planar objects we have to define three types of topological relations:

- continuity of edges forming the outline of a given surface in nodes - **connectivity**,
- the affiliation of the edge to the given area, ie the line-polygon relationship or to define areas **(area definition)**,
- neighborhood of faces - edges have a given direction, so we can specify the face to the right and left of the given edge, ie neighborhood of faces - **contiguity**.

layer approach



object approach - newer - basic characteristics:

- each object has its own geometry, topology, theme and behavior,
- objects can be grouped into classes,
- it is possible to create hierarchical relationships between objects,
- attributes and methods are inherited by deriving for a subclass from an existing class.

(A) **dynamic nature of the real world** - the real world is not static (cities grow, the appearance of agricultural land changes during the year):

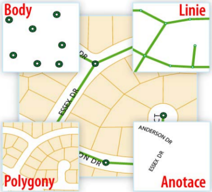
1. **entity selection**, which is the most appropriate representation of a modeled geoelement or phenomenon - is it better to display a forest as a set of points that represent the position of individual trees or as an area (area covered by a forest)?
2. **changes in** a certain **time** - the forest represented by a polygon can disintegrate so that in fact only a scattered group of trees remains, which would be more appropriate to represent using point objects

(B) **scale** - at a scale of 1: 1 000 000 it is suitable to display the city of České Budějovice as a point object, at a scale of 1: 50 000 it will be more appropriate to display them flat;

(C) **identification of discontinuous (discrete) phenomena** - example: if the forest is represented as an area object, where should we place the edge of this forest? Do forests have edges?

Vector stuctures:

- non-topological,
- topological.

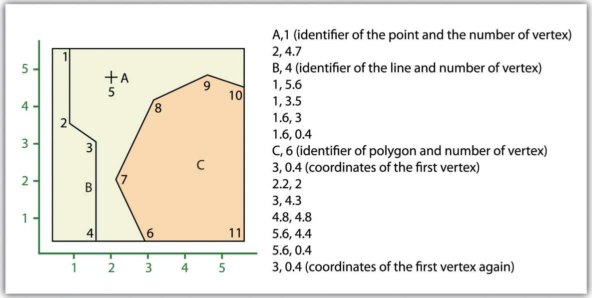


Most usable vector data structures can essentially be classified into one of the following "classical types":

- **spaghetti** model,
- **topological** model,
- **hierarchical** model.

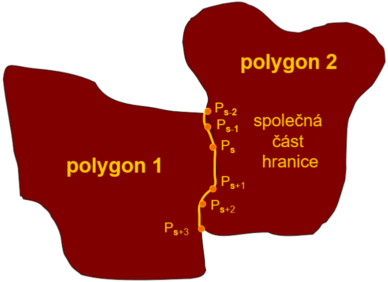
Vector data
spaghetti model

= the simplest (non-topological) type - each geo-feature on the map is coded separately, without creating relationships with surrounding geo-features.



In essence, it is a direct transcription of the classic map line by line into digital form.

problems that may occur:

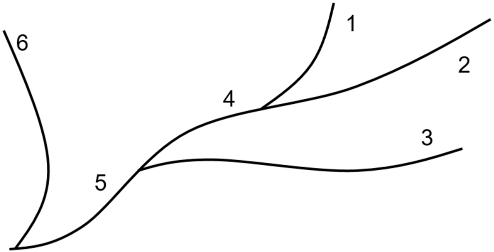


polygon 1: $P_{11}(x_{11}, y_{11})$, $P_{12}(x_{12}, y_{12})$,....., $P_{s-2}(x_{s-2}, y_{s-2})$, $P_{s-1}(x_{s-1}, y_{s-1})$,
$P_s(x_s, y_s)$, $P_{s+1}(x_{s+1}, y_{s+1})$, $P_{s+2}(x_{s+2}, y_{s+2})$, $P_{s+3}(x_{s+3}, y_{s+3})$, ..., $P_{11}(x_{11}, y_{11})$,

polygon 2: $P_{21}(x_{21}, y_{21})$, $P_{22}(x_{22}, y_{22})$,....., $P_{s-2}(x_{s-2}, y_{s-2})$, $P_{s-1}(x_{s-1}, y_{s-1})$, $P_s(x_s, y_s)$,
$P_{s+1}(x_{s+1}, y_{s+1})$, $P_{s+2}(x_{s+2}, y_{s+2})$, $P_{s+3}(x_{s+3}, y_{s+3})$, ..., $P_{21}(x_{21}, y_{21})$,

Vector data
spaghetti model

problems that may occur:

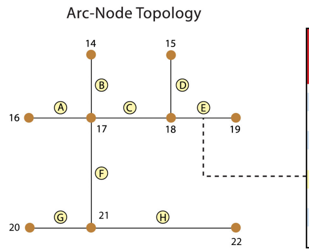


příklad uložení sítě do paměti:
řádek 1: x_1,y_1; x_2,y_2; x_3,y_3; .....
řádek 2: x_4,y_4; x_5,y_5; x_3,y_3; .....
řádek 3: x_3,y_3; x_6,y_6; x_7,y_7; .....
řádek 4: x_8,y_8; x_9,y_9; x_7,y_7; .....
řádek 5: x_7,y_7; x_{10},y_{10}; x_{11},y_{11};

počítač bude schopen reprodukovat obraz, ale nastává problém v okamžiku, kdy chceme řešit problémy, týkající se sítě

The basis of the topological model is the recording of the lines forming the map in the form of a planar graph.

Sometimes such an arrangement is also referred to as a **NAA (Node-Arc-Area)** -node-edge-surface representation (e.g., Worboys, 1995). The basic components of the model are: oriented arc - edge - arc, intersection - node - node and area - polygon - area.
Arc - node topology:

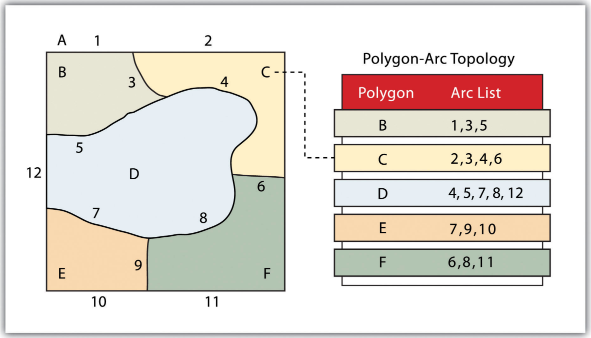


Arc-Node List

Arc	From-Node	To-Node
A	16	17
B	14	17
C	17	18
D	15	18
E	18	19
F	17	21
G	20	21
H	21	22

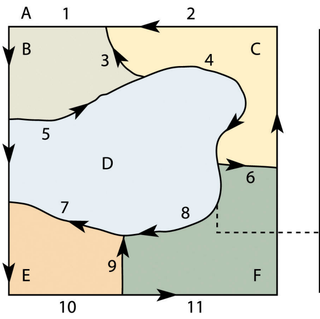
polygon definition

Arc - polygon topology:



contiguity

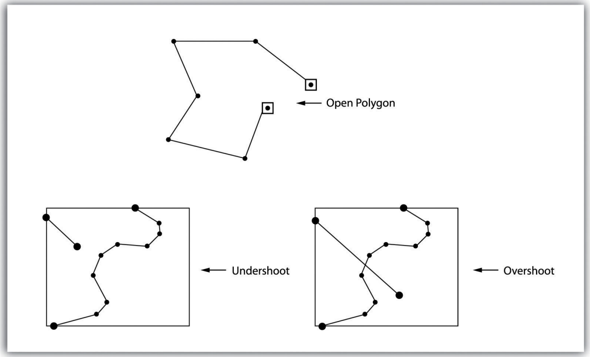
Polygon topology:



Left-Right Topology

Arc	Left Polygon	Right Polygon
1	A	B
2	C	A
3	B	C
4	C	D
5	B	D
6	C	F
7	E	D
8	F	D
9	E	F
10	E	A

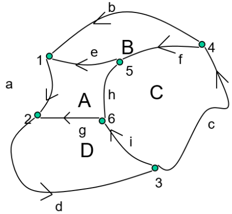
topology errors:



NAA representation - relational database - rules:

- each oriented arc has exactly one start and one end node,
- each node must be the start or end point of at least one oriented arc,
- each area is bounded by one or more oriented arcs,
- oriented arcs can intersect only in their nodes,
- each oriented arc has exactly one area on its left and on its right side,
- each area must be the left or right side of at least one oriented arc.

DCEL (Double Connected Edge List) - complete representation of the topology of an interconnected planar graph - improves the ability to search the structure by specifying the preceding and following edges for each edge described.

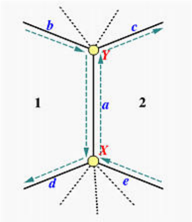


Arc_ID	poč_nod	konc_nod	levý_pol	pravý_pol	předch.	násl.
a	1	2	A	X	e	d
b	4	1	B	X	f	a
c	3	4	C	X	i	b
d	2	3	D	X	g	c
e	5	1	A	B	h	b
f	4	5	C	B	c	e
g	6	2	D	A	i	a
h	5	6	C	A	f	g
i	3	6	D	C	d	h

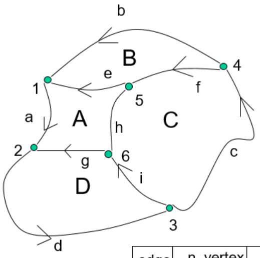
NAA

Windged edge - all possible information about connections between nodes, edges and surfaces is described:

- basic spatial object = edge,
- Each edge is associated with its two respective surfaces: p-face, n-face and two vertices: p-vertex, n-vertex.
- There are also four other edges associated with each edge: nc-edge, pc-edge, na-edge, and pa-edge edge preceding counterclockwise).

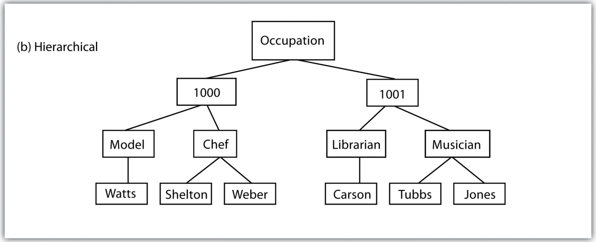


Winged edge



edge	n_vertex	p_vertex	n_face	p_face	nc_edge	na_edge	pc_edge	pa_edge
a	1	2	A	X	g	e	b	d
b	4	1	B	X	e	f	c	a
c	3	4	C	X	f	i	d	b
d	2	3	D	X	i	g	a	c
e	5	1	A	B	a	h	f	b
f	4	5	C	B	h	c	b	e
g	6	2	D	A	d	i	h	a
h	5	6	C	A	i	f	e	g
i	3	6	D	C	g	d	c	h

... separately stores data on points, nodes, lines - edges and surfaces - polygons in a logical hierarchical structure.



advantages	disadvantages
it is possible to work with individual objects as separate units	computational complexity (problems in analyisis)
less memory requirements	complexity of data structure
good representation of the phenomenal structure of data	more complex answers to locational questions
high geometric accuracy	difficult overlay creation of vector overlays
quality graphics, accurate drawing, representation close to maps	problems in modeling and simulation of phenomena
simple search, editing, generalization of objects and their attributes	

Thank you for your attention.