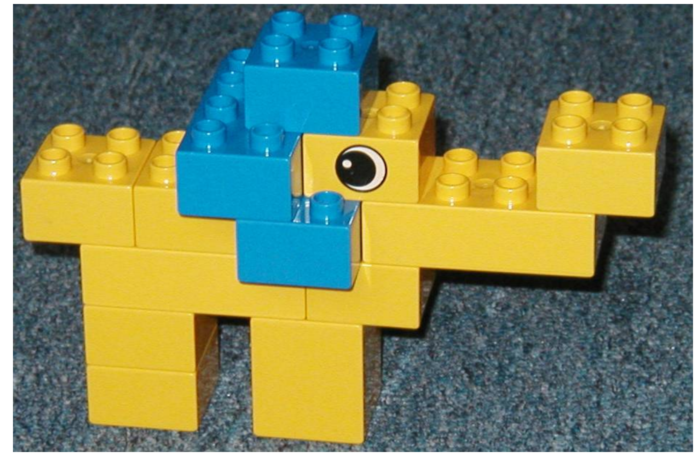
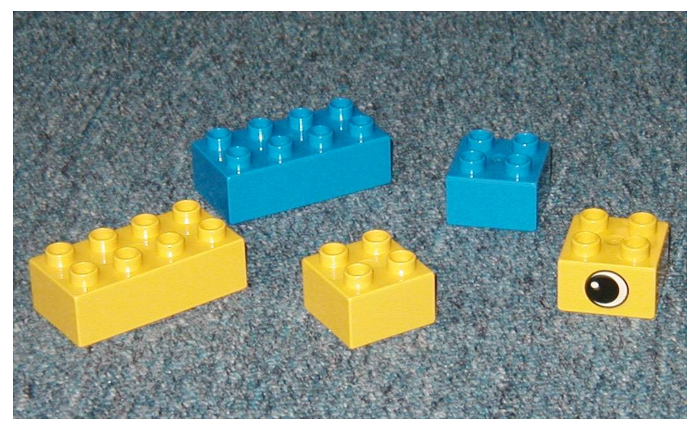
GIS Data Models

GIS Data Modeling

- The process of designing and populating a GIS database using its building blocks
  - What geographic features, surfaces, and networks are needed in a GIS database?
  - What kinds of positional details and attributes are needed?
  - What analyses can be performed?
  - What kinds of modeling elements are available from the GIS system?
  - What modeling elements should be used to represent the features, surfaces, and networks?
- Both the problem domain and the GIS modeling elements have to be well understood



model

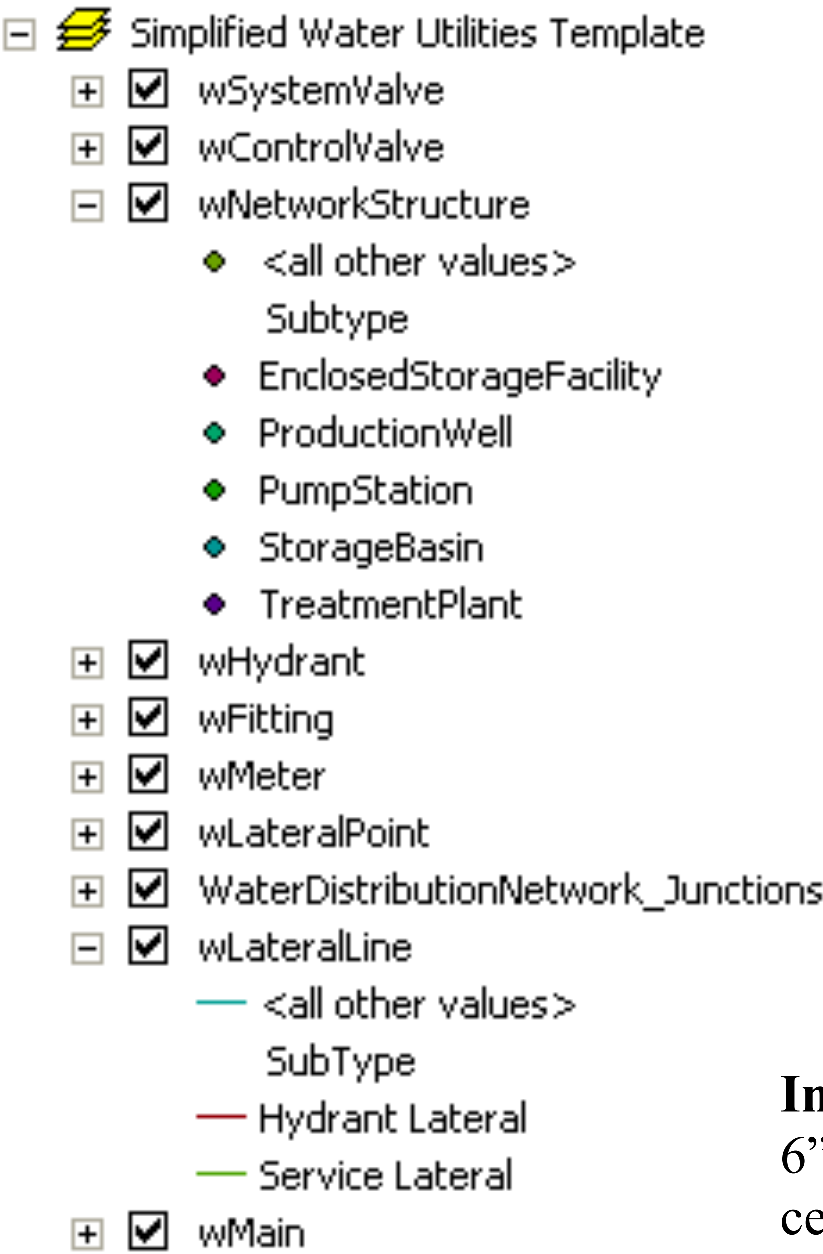


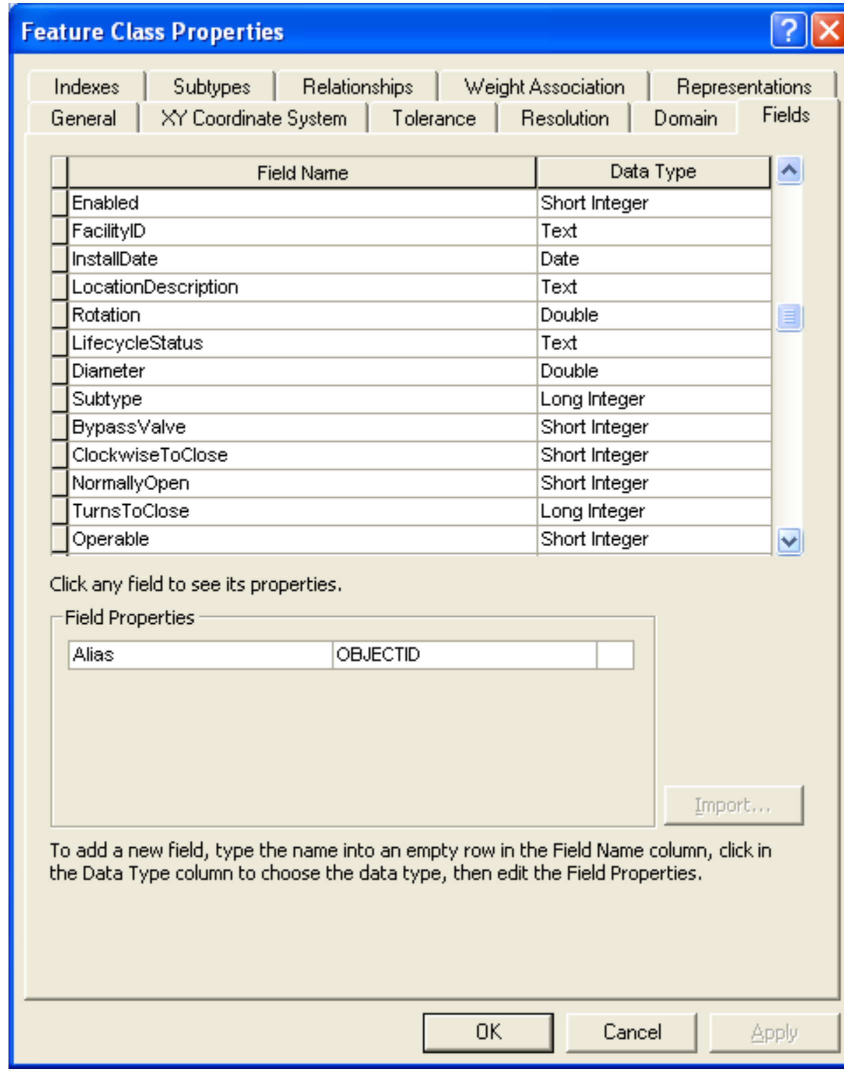
blocks

GIS Database Templates (Domain Data Models)

- Different application domains have different features, surfaces, and relationships to be represented
- Data modeling can be a complex process
  - Understanding the objects, surfaces, relationships, and analysis needed in the domains
  - Select appropriate "building blocks" from a GIS
- Templates help data modeling
  - Organizations in the same industry use similar database structures
  - Templates serve as a starting point for creating GIS databases and save database design time and cost
  - Serve as standards to facilitate data exchange and interoperability

Water Utility Data Model





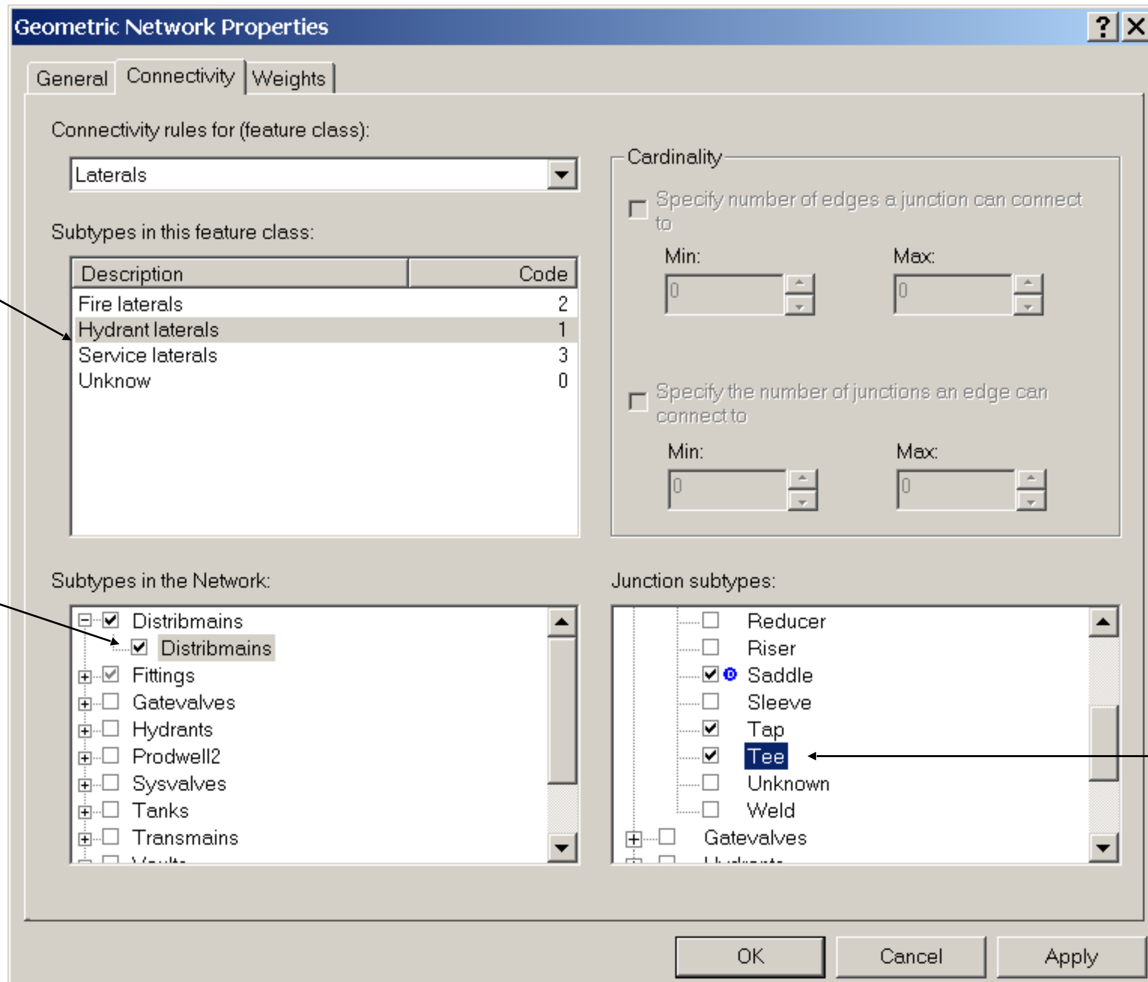
Implements business rules:

6” main line can join another 6” main line with a certain type of fittings

Defining Relationships
(not necessary spatial)

This edge

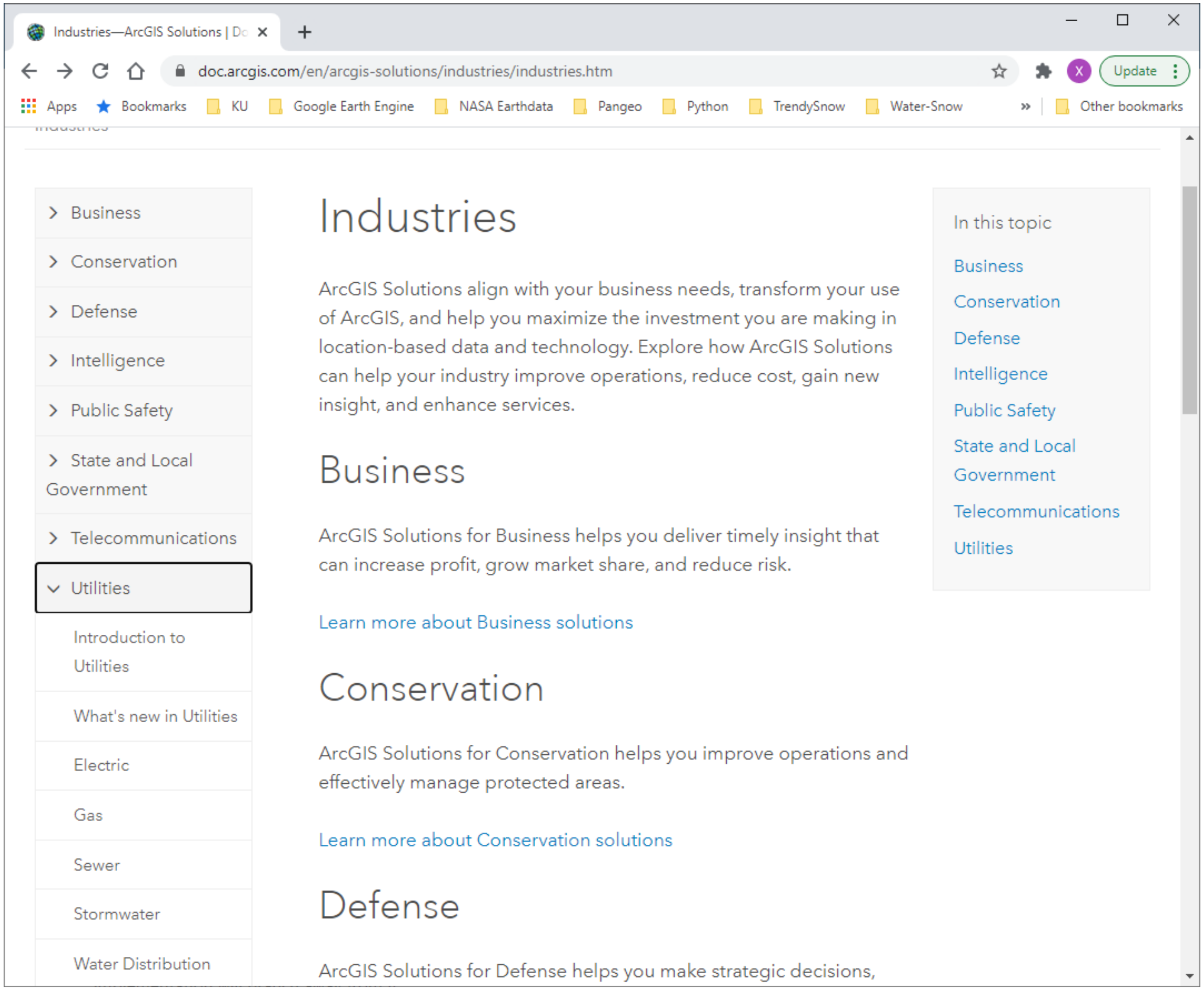
Connects to this edge



Through this junction

Setting Edge-to-Edge connection rules

ESRI Domain Data Models (ArcGIS Solutions)



Stormwater Utility Network Foundation

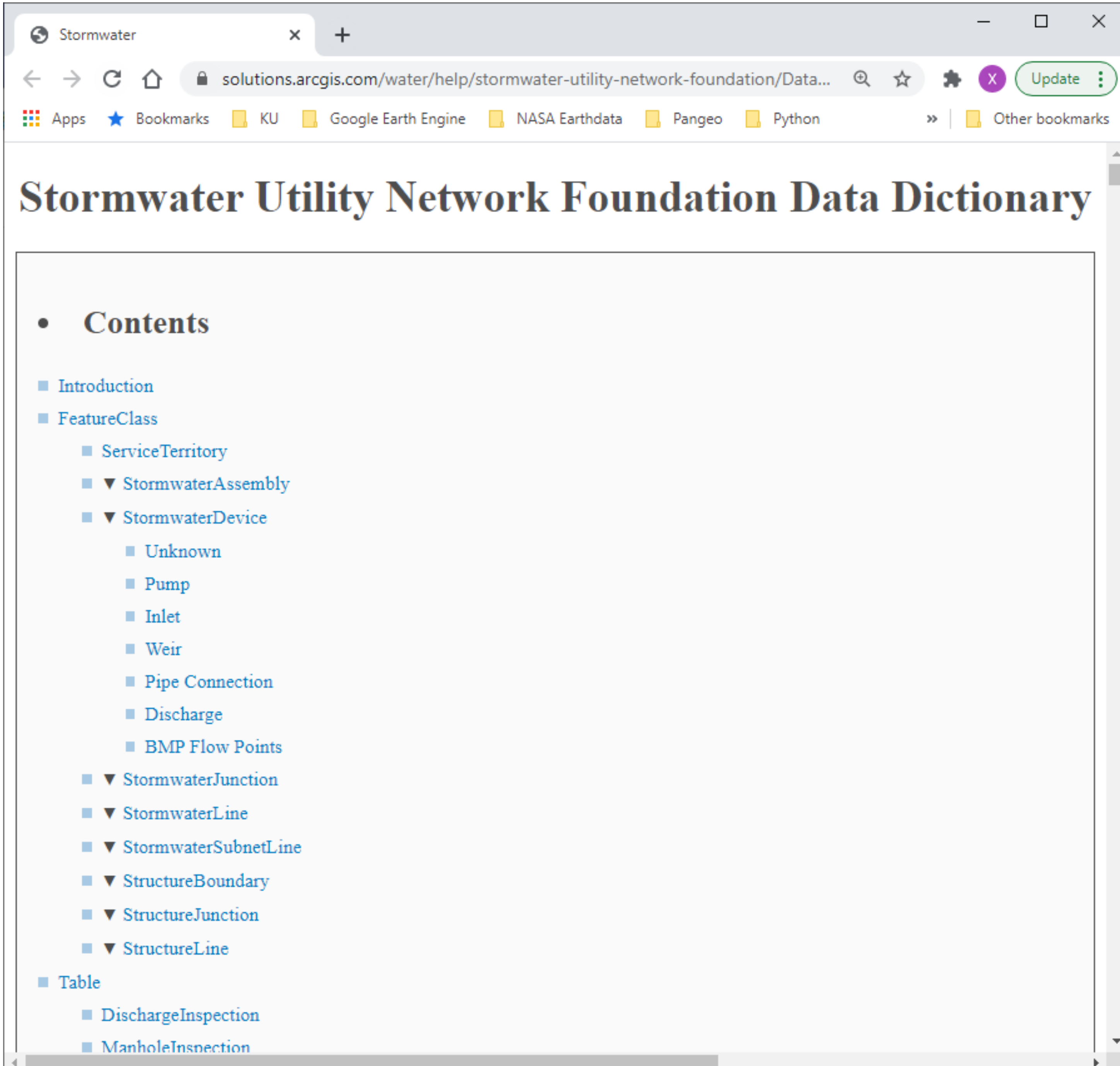
What is the stormwater data model

The stormwater data model is a configuration of the utility network that includes all the feature classes, asset groups and asset types, rules, and associations needed to get started with the Stormwater Utility Network Foundation.

To help you understand the elements in a stormwater utility network, the data dictionary provides an overview of the fields and asset groups included in each feature class and outlines how the same field can be reused for different purposes. There are two data dictionaries, a classic version which is a static document and an enhanced version which dynamically reads a published utility network service.

View the classic stormwater data dictionary.

Tip: Try out the enhanced data dictionary. Review the requirements to ensure browser compatibility.

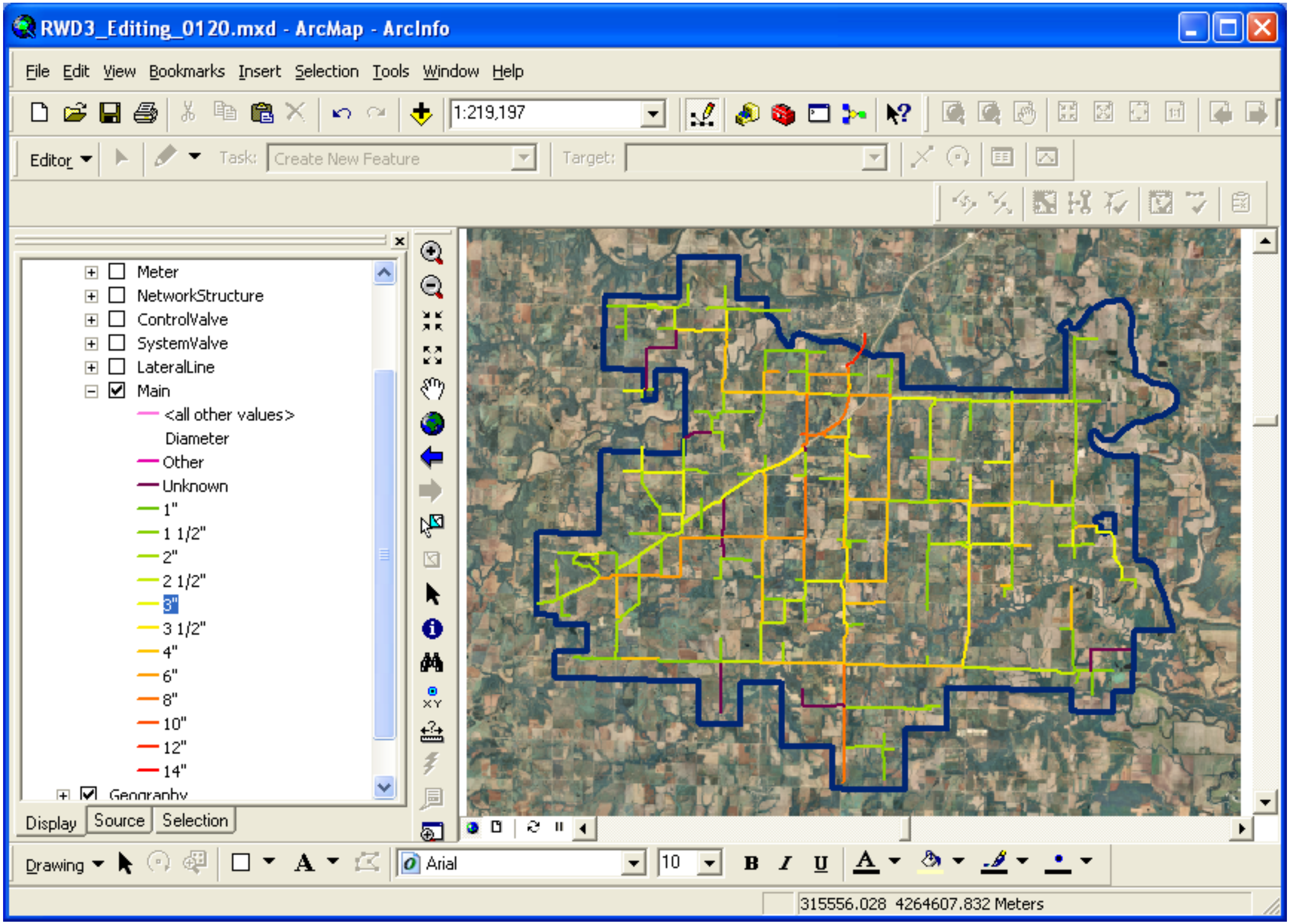

Stormwater
solutions.arcgis.com/water/help/stormwater-utility-network-foundation/Data...
Update
Apps
Bookmarks
KU
Google Earth Engine
NASA Earthdata
Pangeo
Python
Other bookmarks
Stormwater Utility Network Foundation Data Dictionary
Contents
Introduction
FeatureClass
ServiceTerritory
StormwaterAssembly
StormwaterDevice
Unknown
Pump
Inlet
Weir
Pipe Connection
Discharge
BMP Flow Points
StormwaterJunction
StormwaterLine
StormwaterSubnetLine
StructureBoundary
StructureJunction
StructureLine
Table
DischargeInspection
ManholeInspection

Subtype: Inlet

The Inlet asset group in the StormwaterAssembly feature class represents features that allow for the stormwater runoff to enter the conveyance system. These features can be complex and include additional devices and lines, such as filters and pipes.

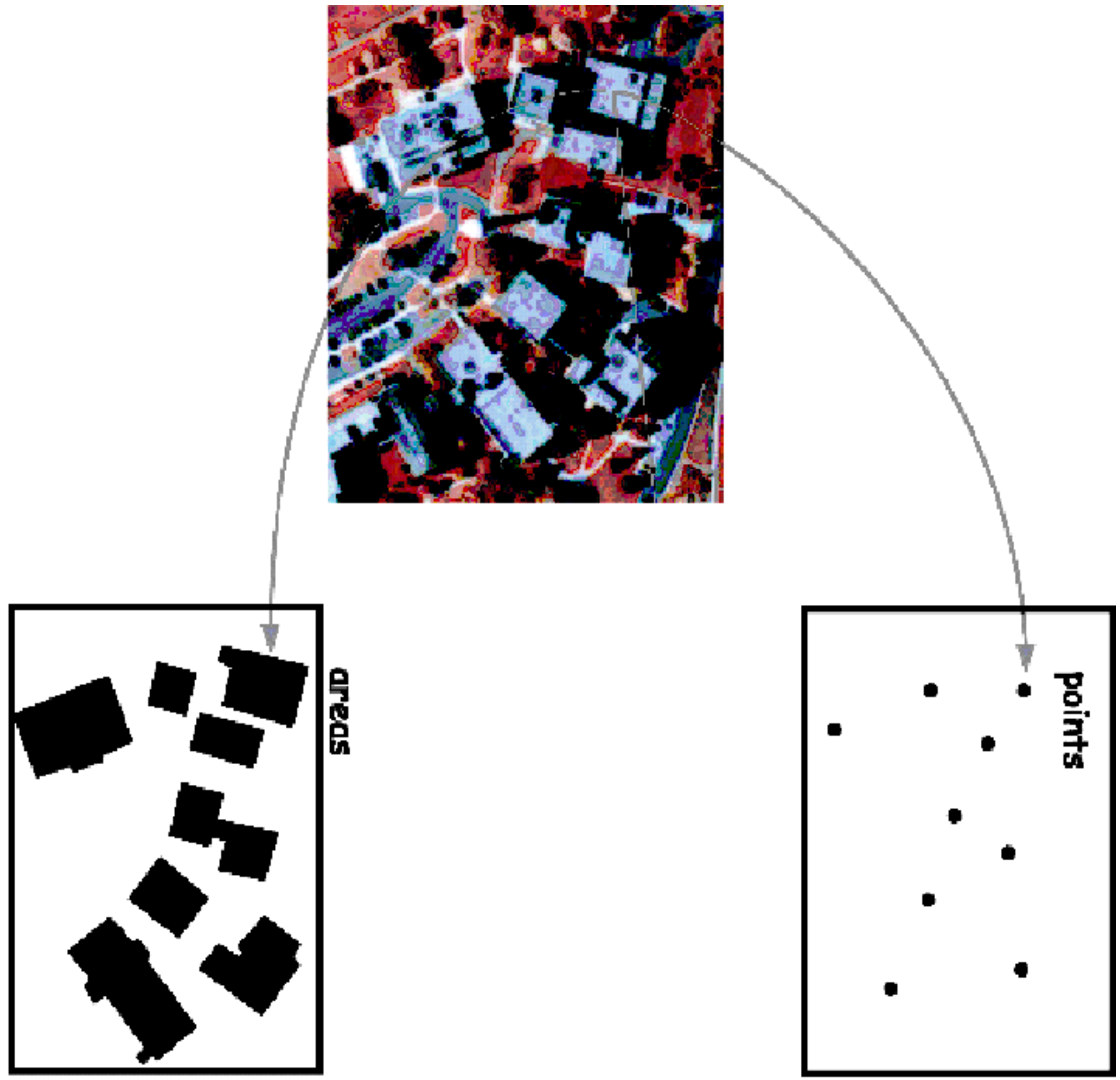
Name	Alias	Type	Default Value	Domain	Nullable	Description
objectid	Object ID	OID			False	
shape	Shape	GEOMETRY			False	
assetgroup	Asset group	LONG			False	
assettype	Asset type	SHORT	0	Asset_Type_Stormwater_Device_Inlet	False	
associationstatus	Association status	SHORT	0	UtilityNetwork_5_AssociationStatus	False	
issubnetworkcontroller	Is subnetwork controller	SHORT	0	UtilityNetwork_5_IsSubnetworkController	False	
isconnected	Is connected	SHORT	2	UtilityNetwork_5_IsConnected	False	
subnetworkcontrollername	Subnetwork controller name	TEXT (2000)	Unknown		False	
tiername	tiername: Tier name	LONG	0	UtilityNetwork_5_TierName	False	
tierrank	Tier rank	LONG	0		False	

A GIS Database is a Realization (Instance) of the (Water Utility) Data Model



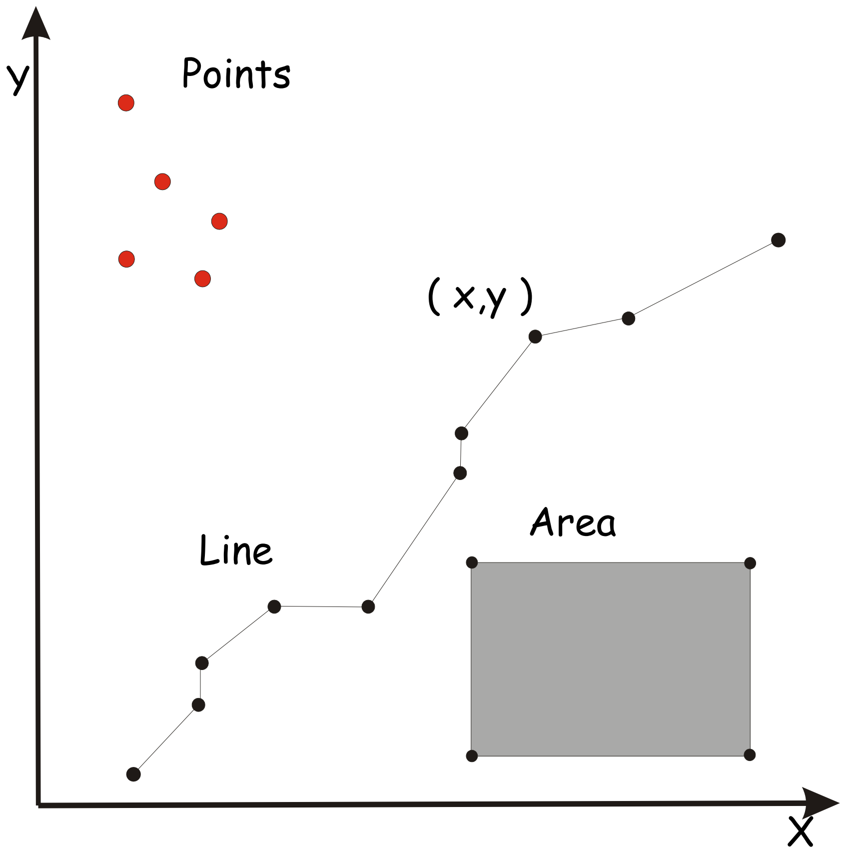
Representations Depend on Application Needs

There are usually several possible ways of representing the same geographic phenomena

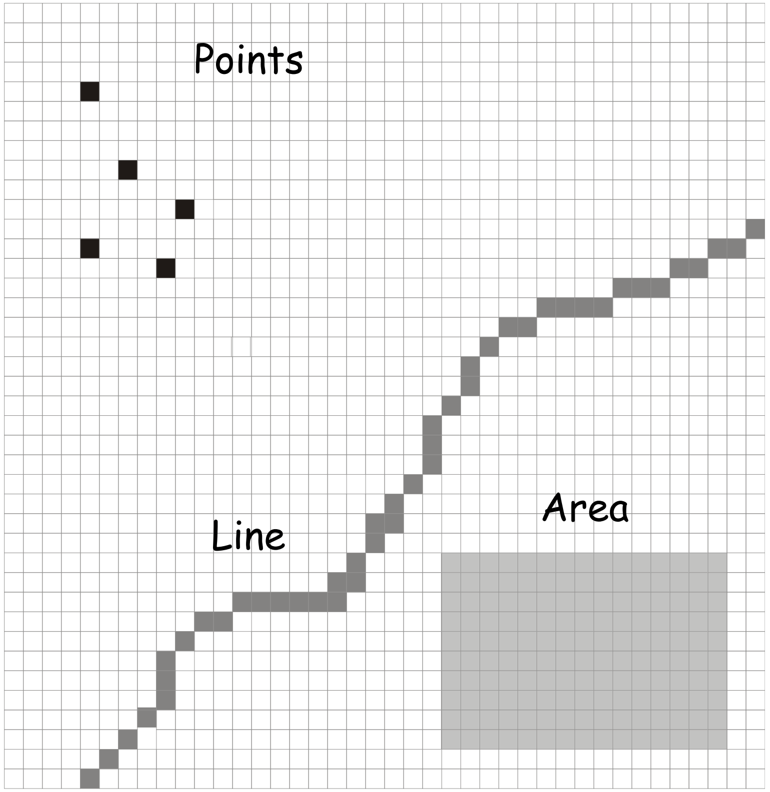


Multiple Representations of Discrete Features

Vector

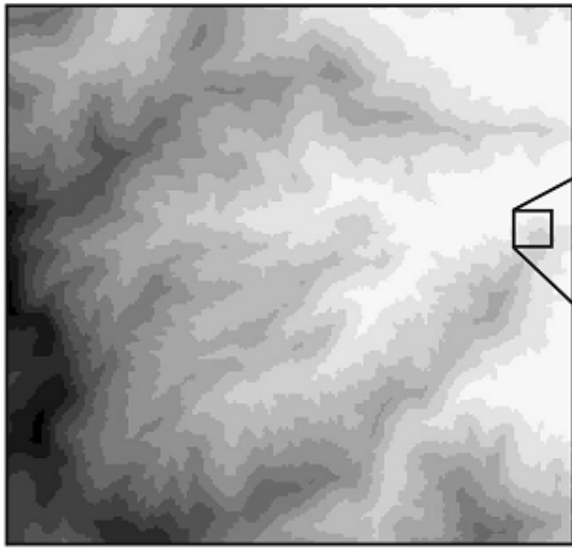


Raster



Multiple Representations of Fields

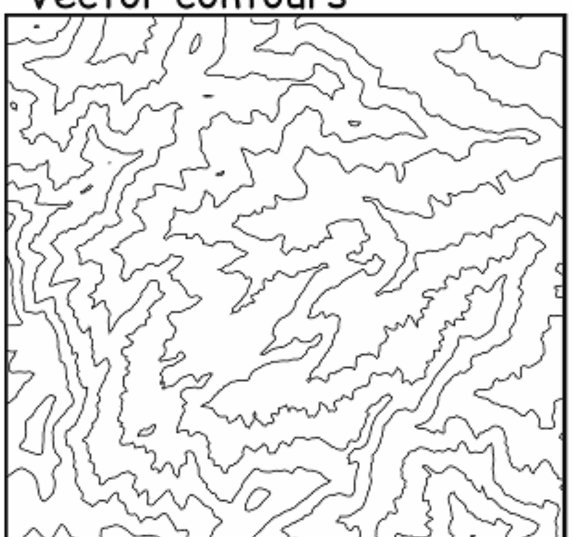
Raster DEM



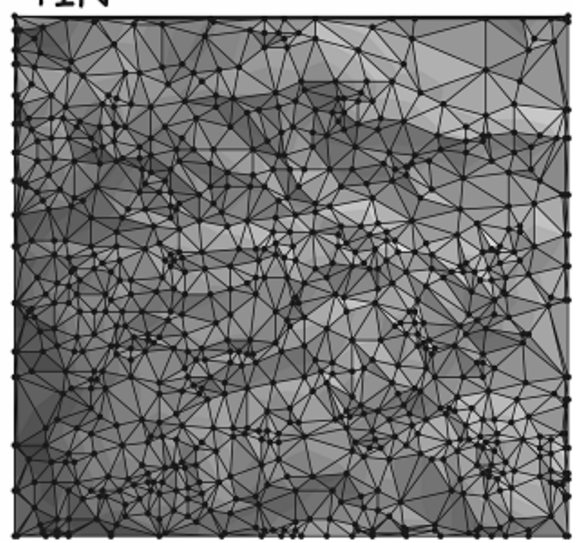
Detailed view of raster cells

645	650	654	658	653	648
664	666	670	672	668	659
678	682	684	693	689	680
703	708	714	721	719	716
728	732	738	744	745	732
730	739	744	749	748	735

Vector contours



TIN



Triangulated Irregular Networks

Vector-Raster Conversion

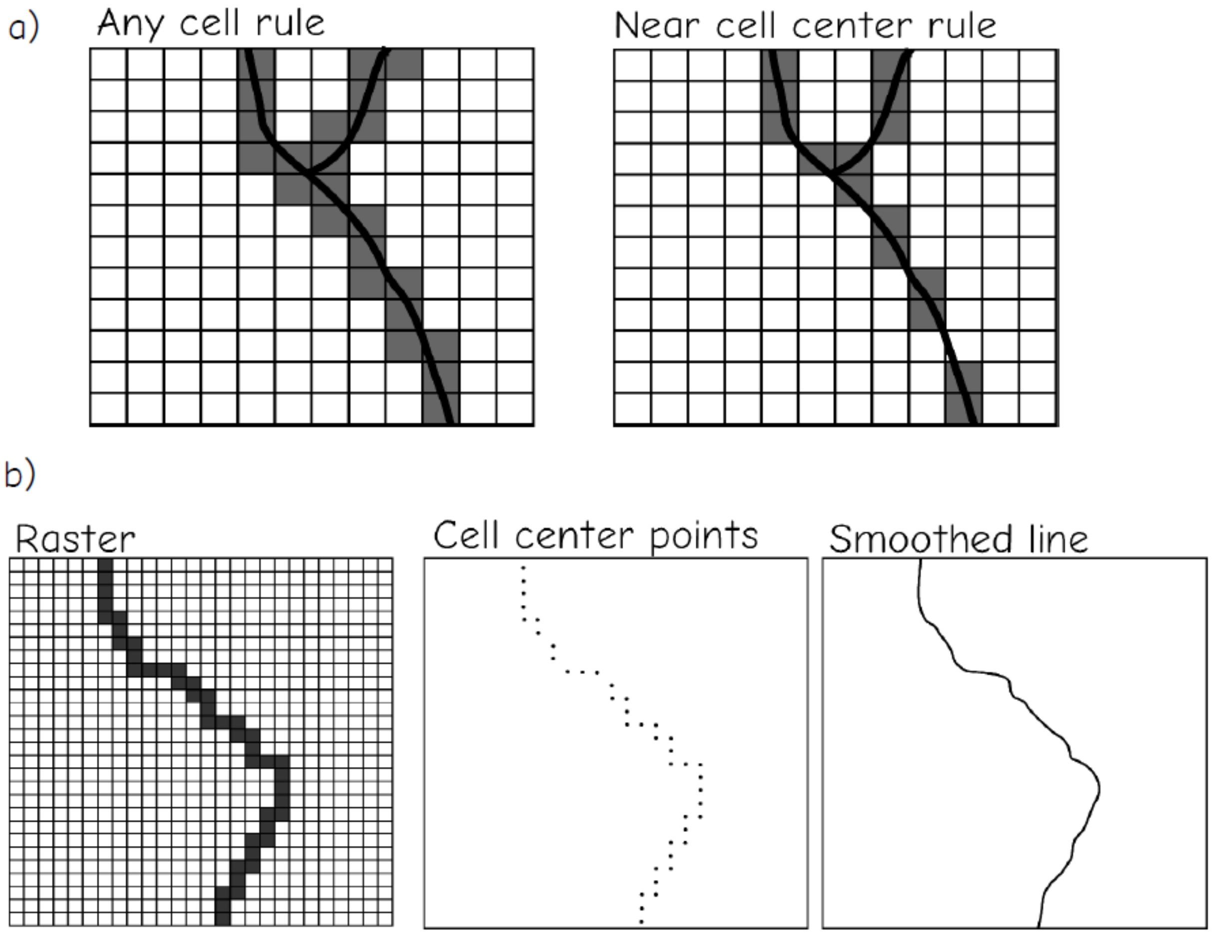


Figure 2-36: Vector-to-raster conversion (a) and raster-to-vector conversion (b). In a, cells are assigned in a raster if they intersect with a converted vector. The left and right panels show how two assignment rules result in different raster coding near lines. Panels in b show how raster data may be converted to vector formats, and may involve line smoothing or other operations to remove the "stair-step" effect.

- Raster line length >= vector line length

Vector-Raster Conversion

- Appropriate cell size
- Missing points or polygons

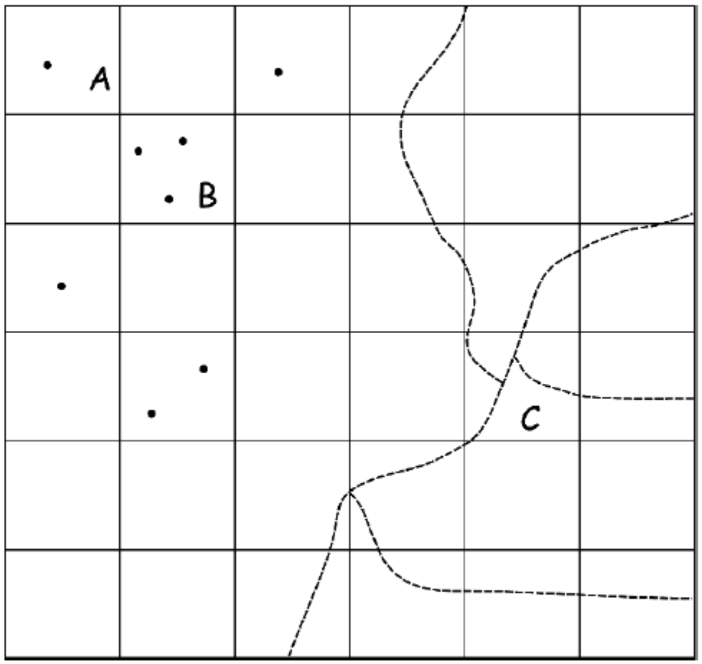
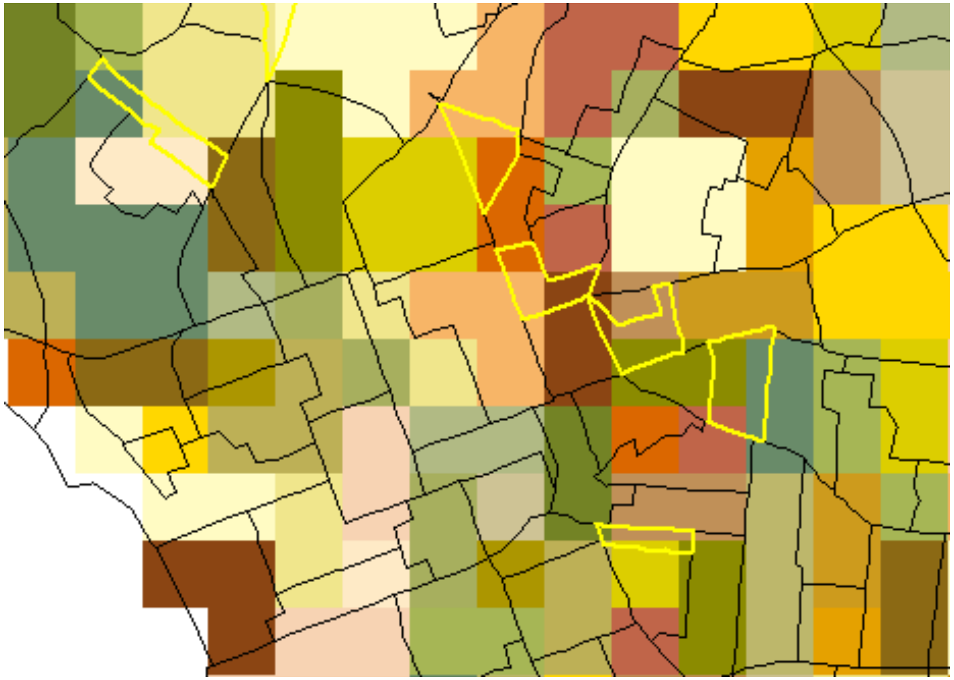
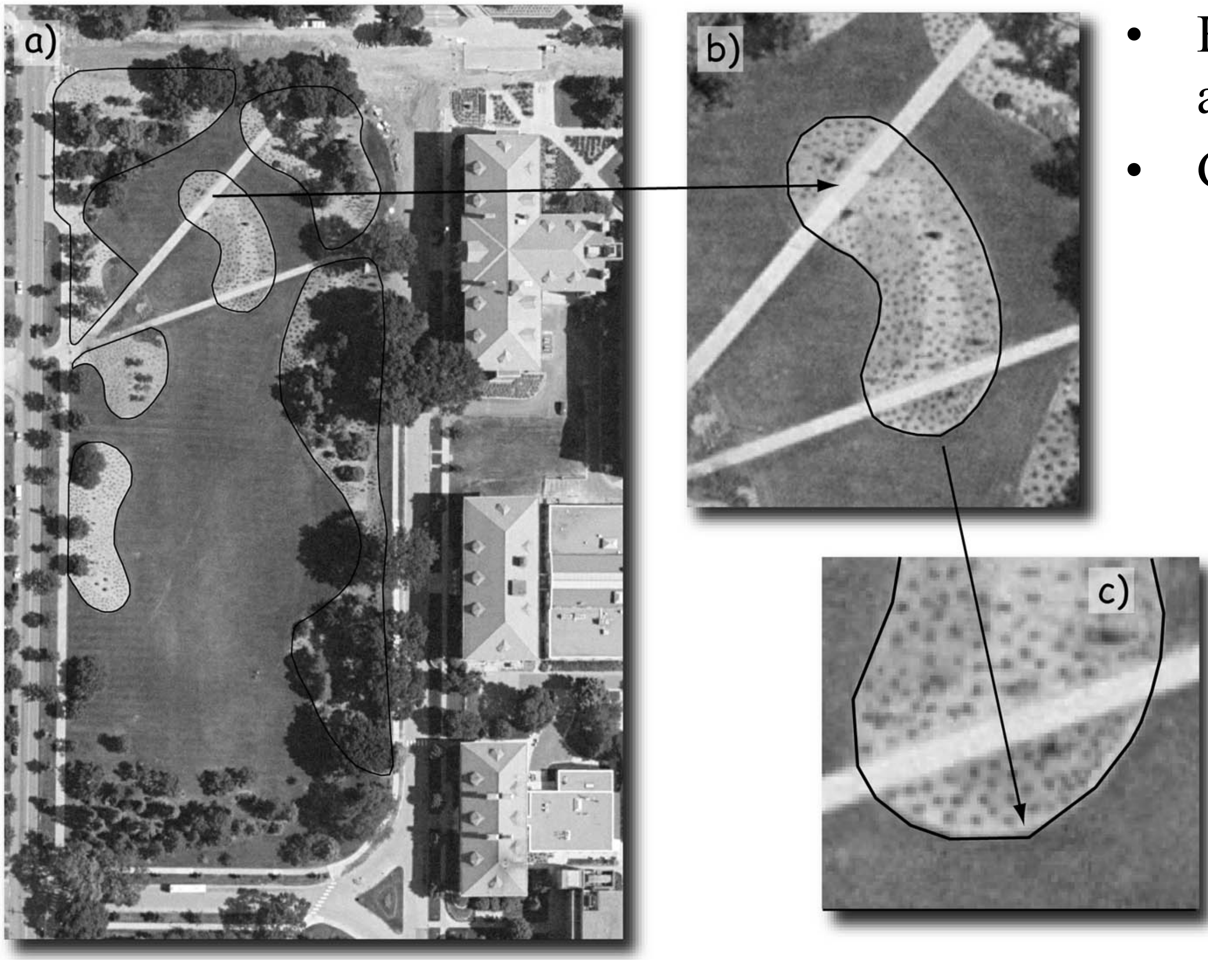


Figure 2-33: Raster cell assignment requires decisions when multiple objects occur in the same cell.

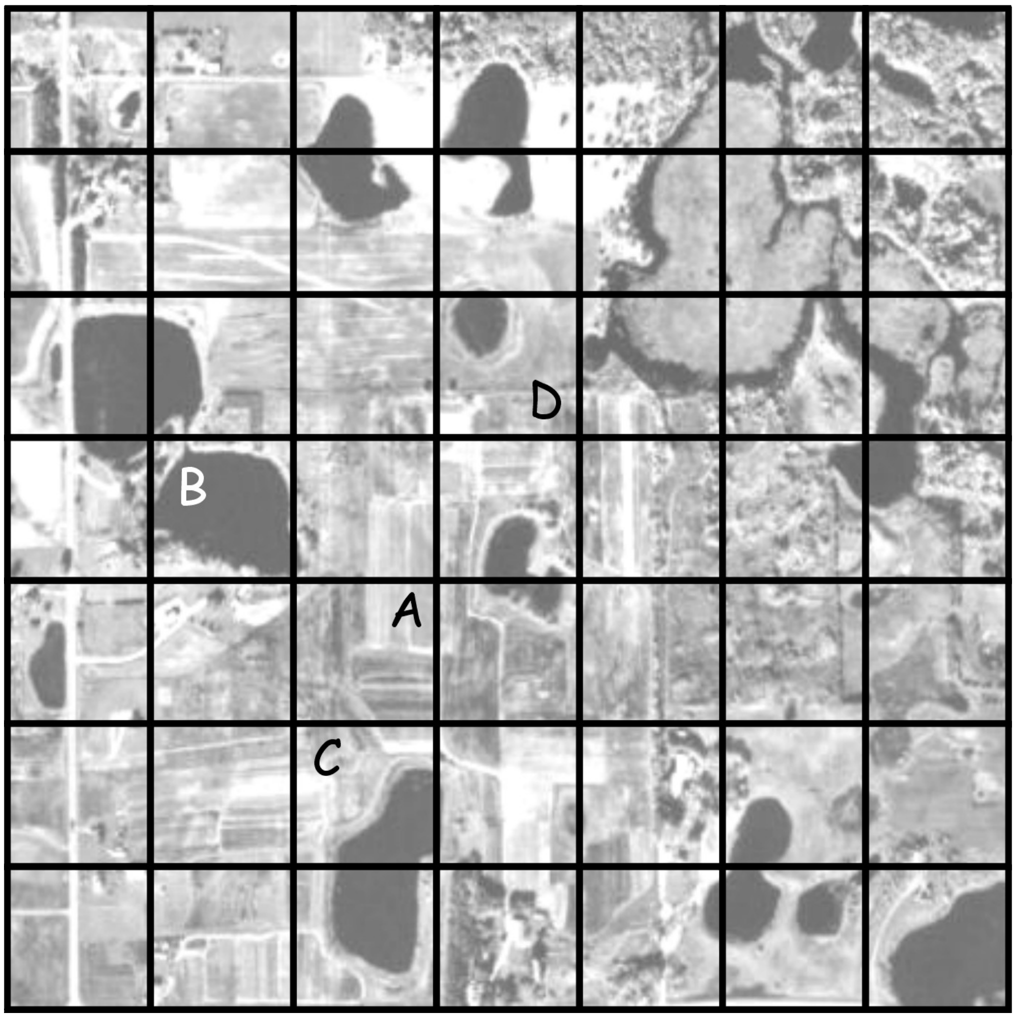


Representations Are Approximations!



- How much detail (location & attribute) is necessary?
- Generalization?

Representations Are Approximations!



- What cell size?
- How to deal with mixed cells

Summary

- Vector and raster data models
  - Building blocks used to represent geographic phenomena in GIS
  - The representation power of a GIS
- Database templates for specific application domains
  - data modeling (database design)
  - Database blueprints
- GIS databases (models of reality)
  - Digital representations of geographic entities and phenomena, where their locations, attributes, and relationships are stored
  - A realization of a database template

GIS Data Structures

Topics

- Binary number
- Vector data structures
- Raster data structures

GIS Data Structures

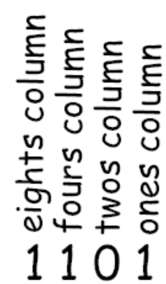
- Ways data stored in computer (memory and hard disk)
  - Involved more in computer science
- Data in GIS
  - Vector data model (points, lines, and polygons)
  - raster data model (rasters)
  - Attributes, location, and relationships (spatial and non-spatial)
- The same data can be stored in different ways
- Why data structures matter
  - Reduce data size
  - Speed up access, update and analysis
- Results in different (usually proprietary) file formats
  - Different GIS vendors
  - Same vendor over time

The Binary Number System

- Everything is represented by 0 & 1
- Binary numbers use a base of 2
  - Using bits are 0 or 1
  - Each successive column of a number represents a power of 2
- Integers and real numbers can be represented by the binary system
  - Not every real number can be precisely represented
- Coordinates and attributes are stored as binary numbers
- Data size
  - 8-bit is called a **byte**
  - KB (KiloByte) = 1024 bytes
  - MB (MegaByte) = 1000 KB
  - GB (GigaByte), TB (TeraByte), PB (PetaByet), EB (ExaByte)

Decimal number system:
$301 = 3*10^2+0*10^1+1$

Binary Columns



8 + 4 + 0 + 1 = 13

$1*2^3 + 1*2^2 + 0*2^1 + 1 = 13$

binary	decimal
00000001	1
00000010	2
00000011	3
00000100	4
00000101	5
00000110	6
00000111	7
00001000	8
00001001	9
00001010	10
00001011	11
....	....

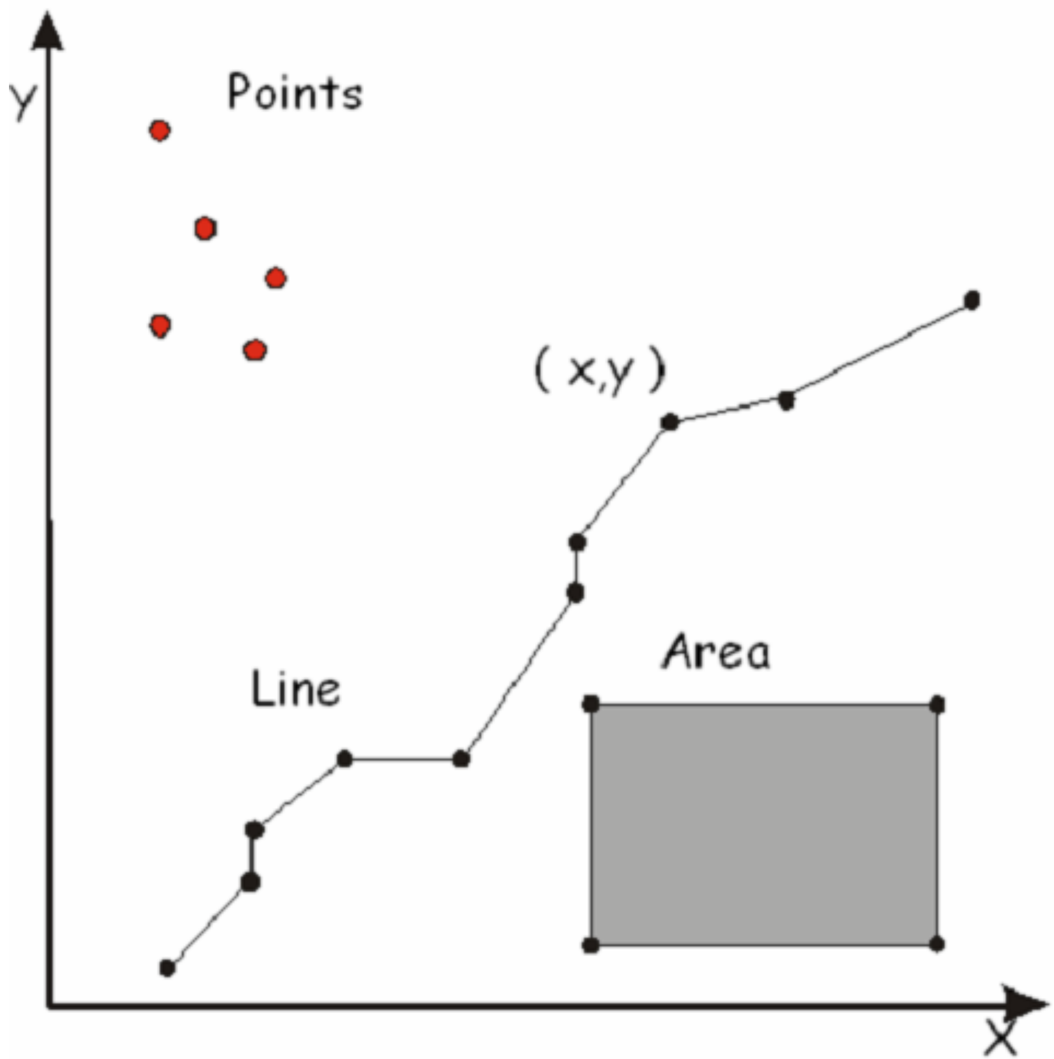
How about string or text?

Text and ASCII Code

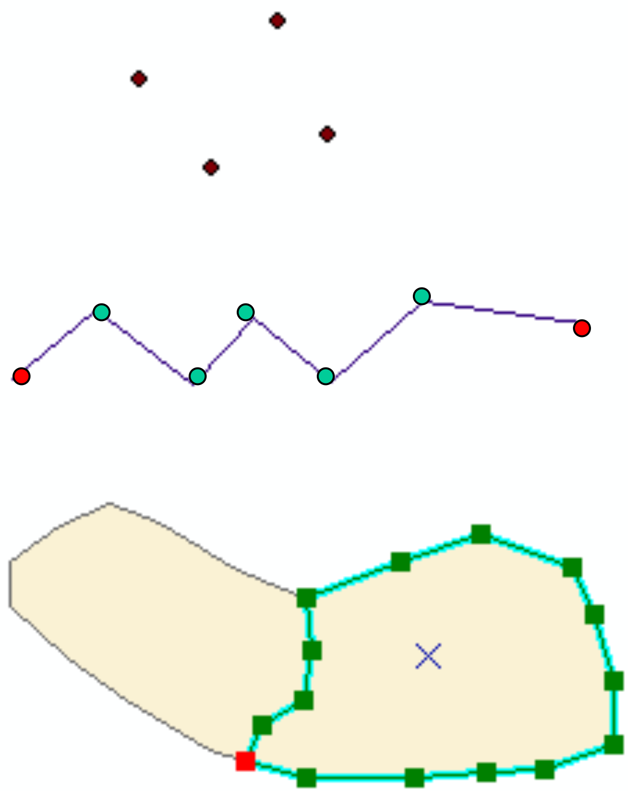
Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.asciitable.com

Vector Data Model and Data Structures



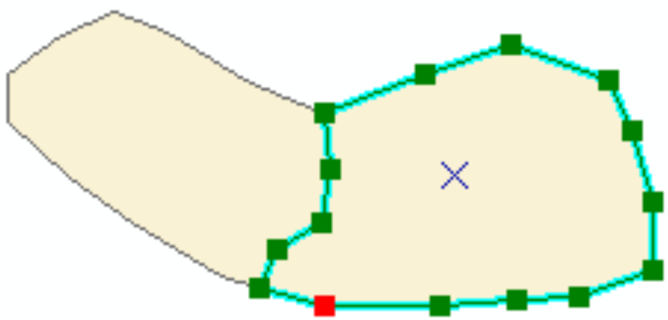
Spaghetti (Simple) Data Structures

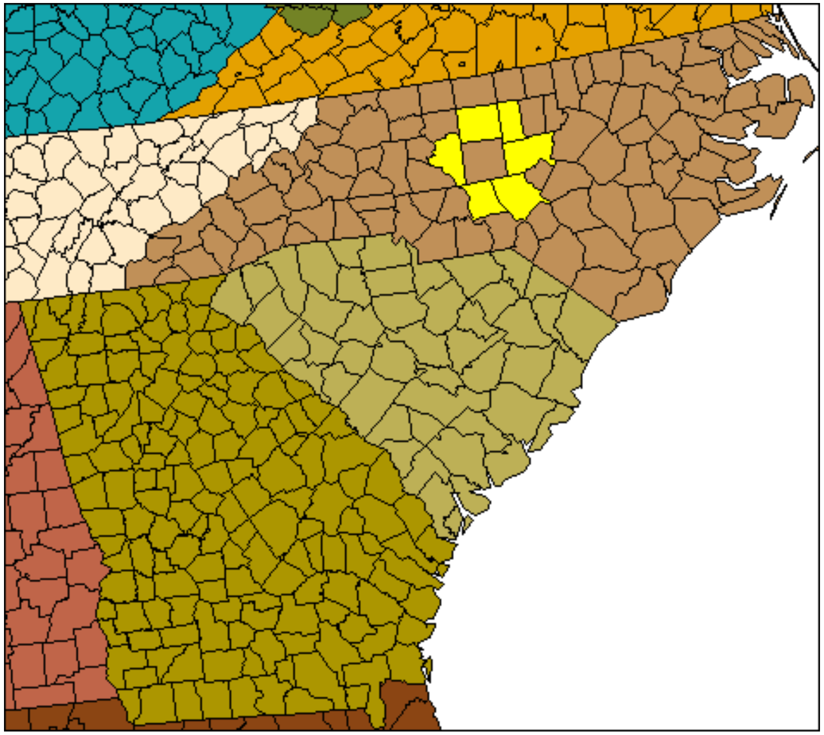


- Point—a pair of x, y coordinate.
- Line—a string of x,y pairs.
- Polygon—closed loop of x, y coordinate pairs
- Straight forward storage
- Features are independent of each other

Issues With a Spaghetti Data Structure

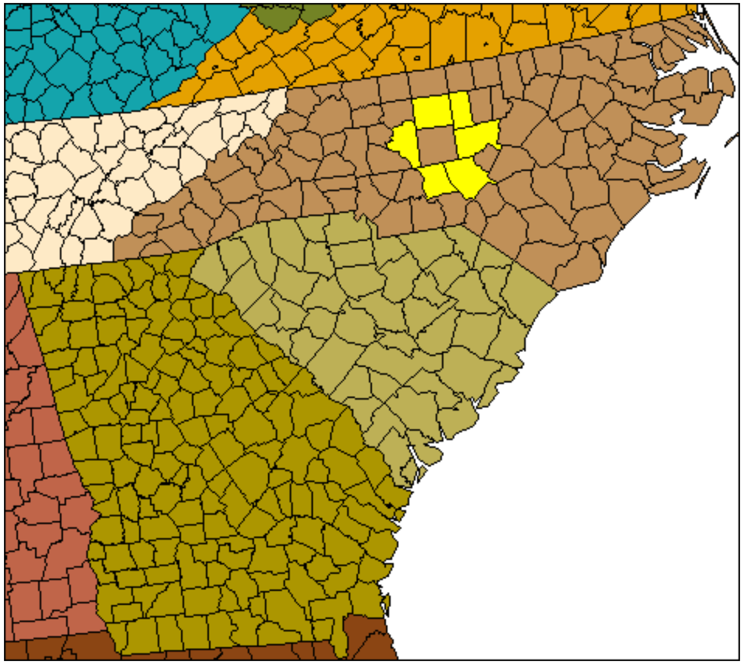
- How many times are shared polygon boundaries stored?
- What are the adjacent counties?





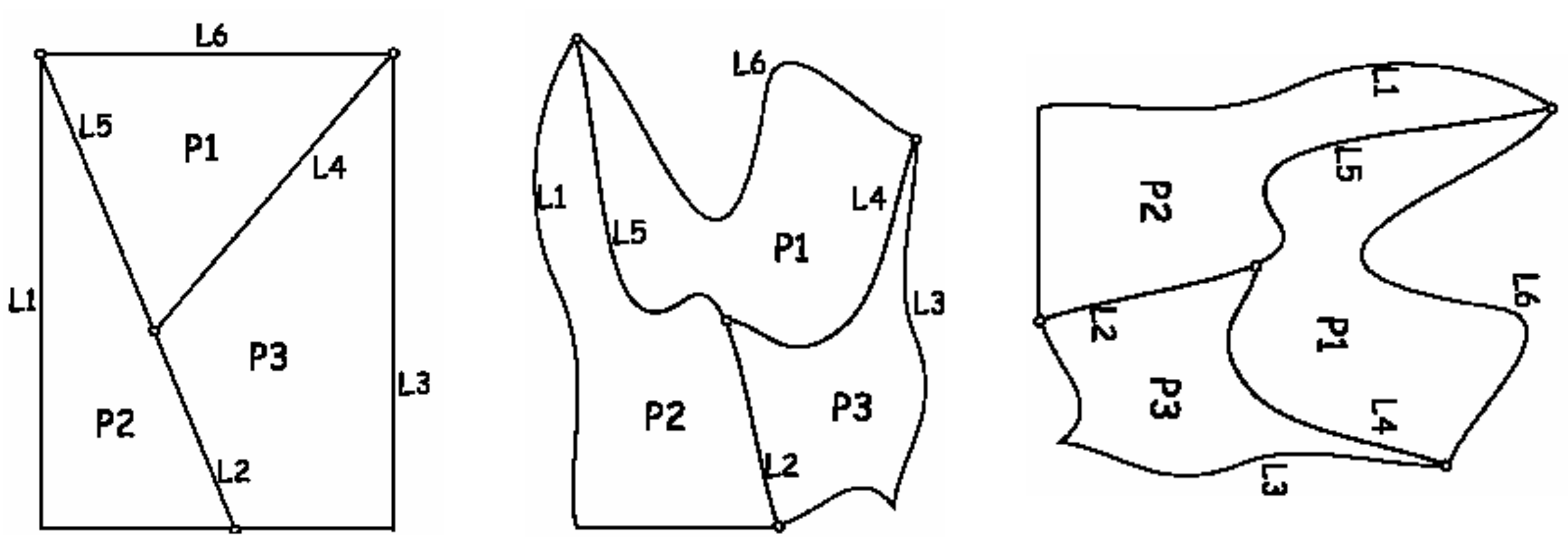
Issues with a Spaghetti Data Structure

- Shared boundaries are stored twice
- Polygon adjacency not stored explicitly
  - Has to be computed from coordinates whenever needed
  - Calculation repeated if the relationship is frequently queried



Topology (***not Topography***)

- Topology are the spatial relationships among geographic features
- Can be derived from coordinates
- But independent of coordinate systems and transformation (such as rotation and stretching)

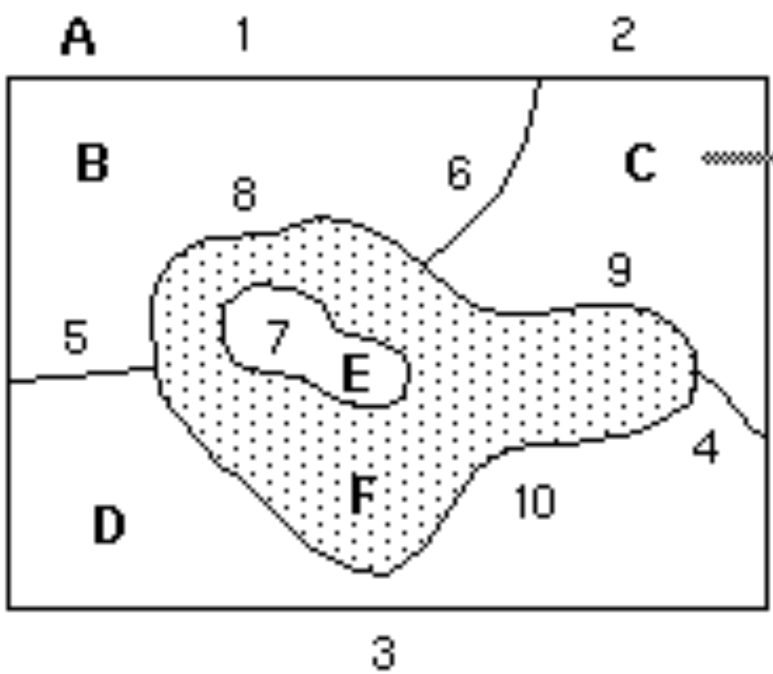


Topological Data Structures

- A topological data structure stores
  - Coordinates
  - Attributes
  - **Spatial relationships**
- Different geometric types have different topology
  - Lines (connectivity)
  - Polygons (Polygon-Line composition and adjacency)
- Different GIS systems may store different kinds of topology
- Topology for polygons as an example

Polygon-Arc (Line) Topology

- Store lines first
- Represent polygons using lines
  - Defines what lines (arcs) make a polygon
  - Avoid storing shared lines twice

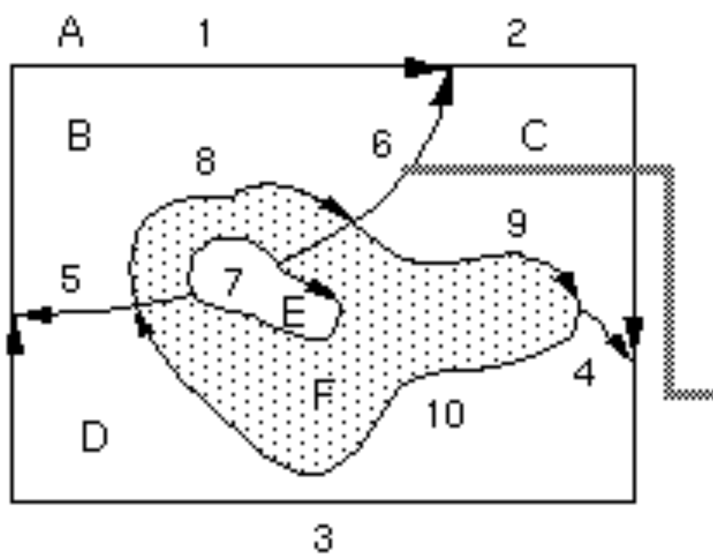


Polygon-Arc Topology

Polygon	Arc List
B	1, 6, 8, 5
C	2, 4, 9, 6
D	3, 5, 10, 4
E	7
F	8, 9, 10, 0, 7

Left-Right (Adjacency) Topology

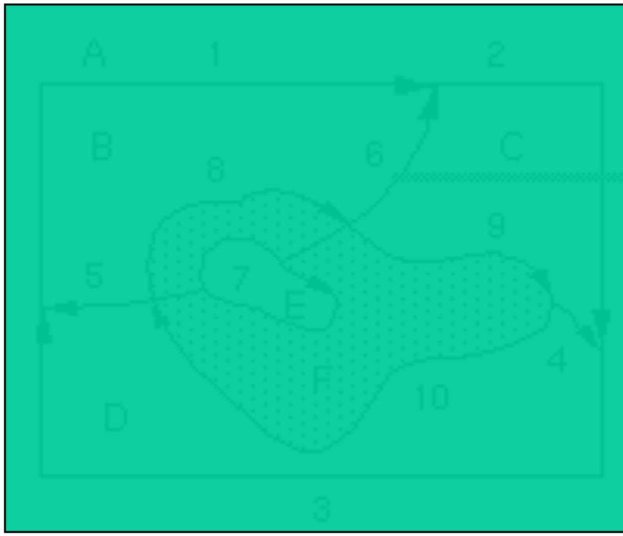
- Store the adjacency between polygons
- Useful in adjacency analysis



Left-Right Topology

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

Find Adjacent Polygon with Topological Information



Left-Right Topology

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

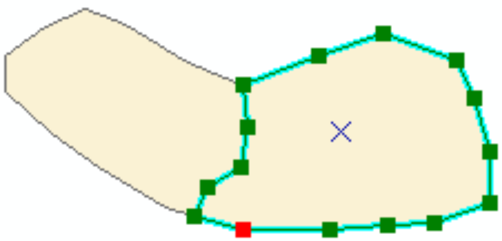
Polygon-Arc Topology

Polygon	Arc List
B	1, 6, 8, 5
C	2, 4, 9, 6
D	3, 5, 10, 4
E	7
F	8, 9, 10, 0, 7

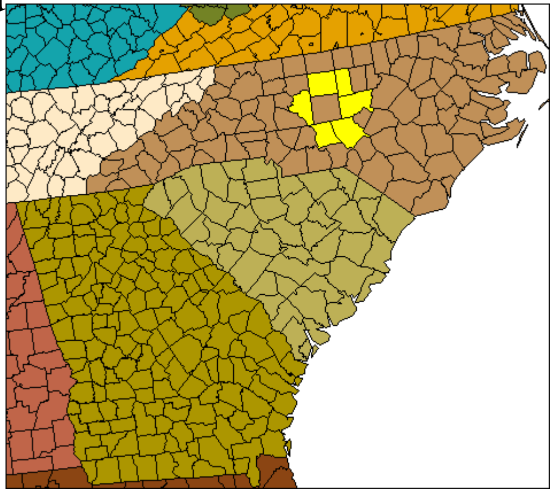
Find polygons which are adjacent to polygon “C” (lines 2, 4, 9, 6)

Topological Data Structures

- Advantages
  - Consistent data—building topology helps find errors
  - Reduce disk space--shared boundaries are stored only once
  - Speed up certain analyses--adjacency analysis

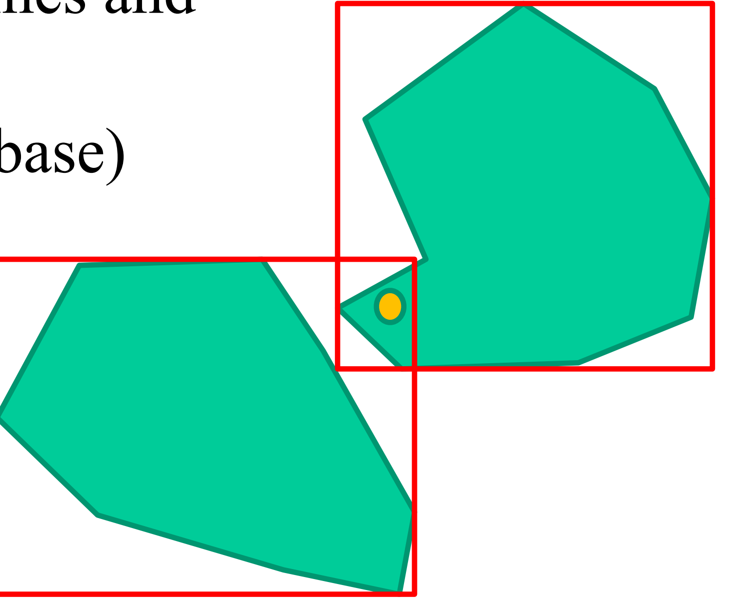


- Disadvantages
  - Have to maintain the topology whenever data is updated
  - May need more disk space (line layers)
- Space vs. time
  - Save time at the expense of space
  - Save space at the expense of time



Spatial Indexing

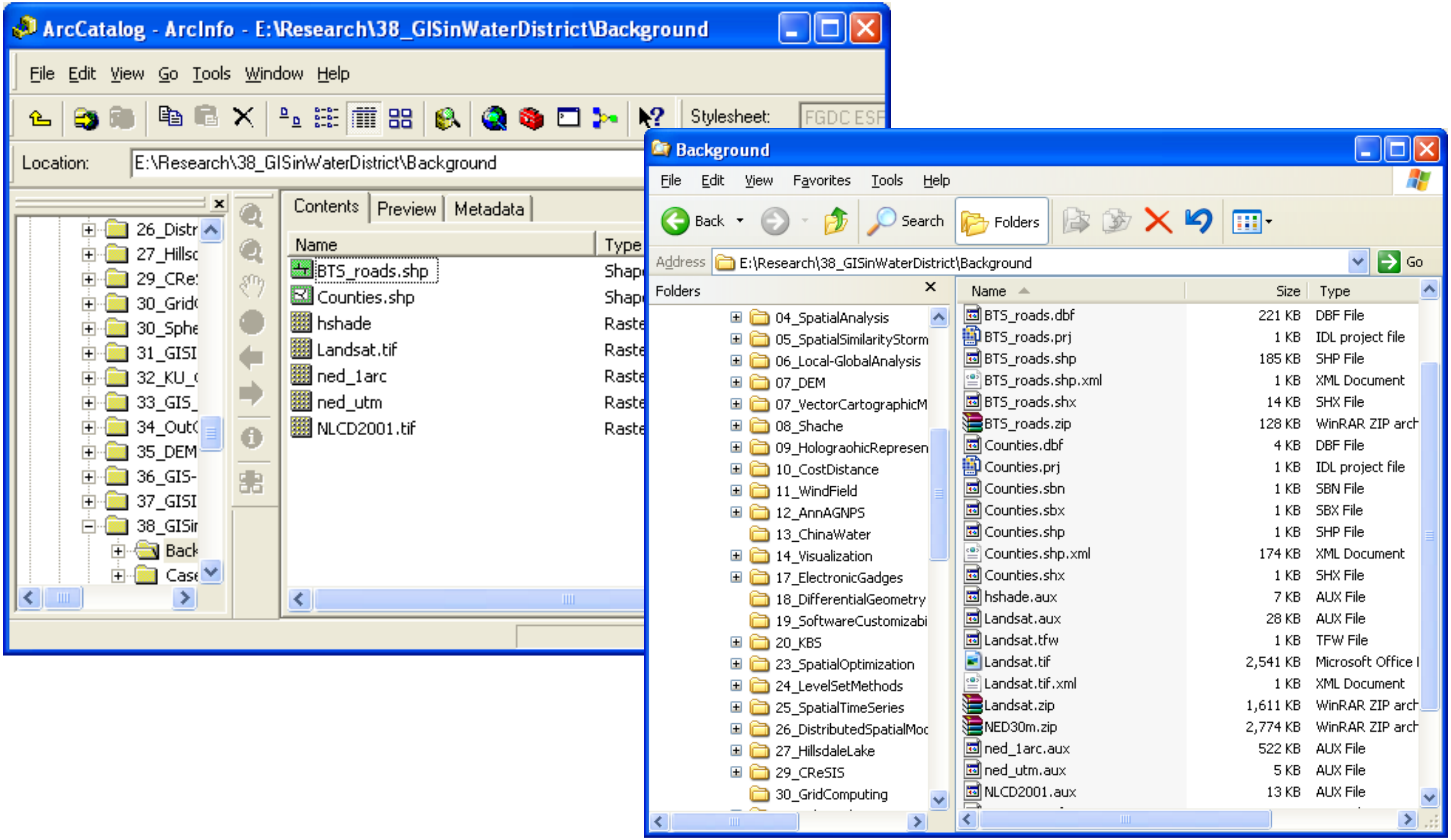
- Data used to speed up data access
  - Table of contents (chapters and their start pages) and Index at the end of a book (find terms quickly)
  - Avoid sequential access of the data
- Spatial indices are used to speed up the access to geographical features by their locations
- Minimum bounding rectangles for lines and polygons
- More from GEOG528 (Spatial Database)



Vector Data Models From ESRI

- ESRI provides several slightly different vector data models to represent discrete objects
  - The geometric elements (building blocks) in those vector data models are slightly different
- Data structures are also evolving
  - More efficient ways are invented
  - Development of database management
- Data models + data structures create new ways of storing geographic data → new file formats
  - Coverages, shapefiles, geodatabases
  - Conversions can be made between different vector datasets
  - Coverages → shapefiles → Geodatabases

Shapefile Example



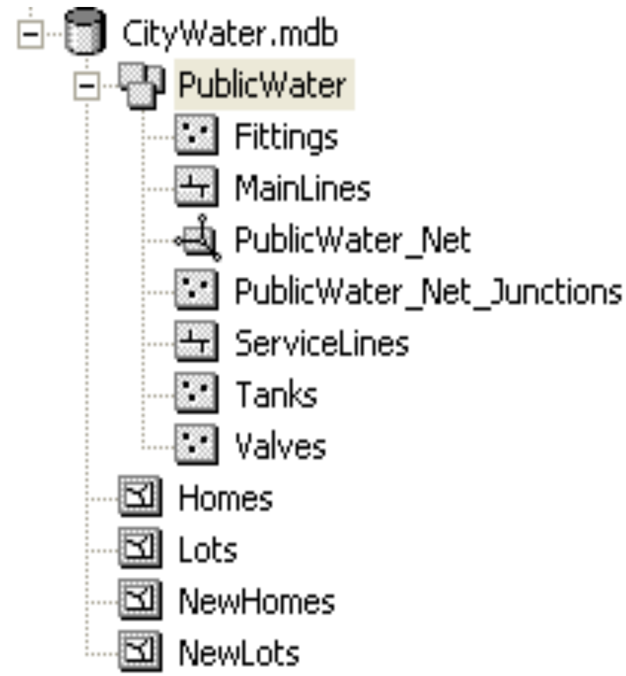
Shapefile Extensions

- Required files:
  1. .shp—the file that stores feature geometry.
  2. .shx—the file that stores the index of feature geometry.
  3. .dbf--the dBASE file that stores the attribute information of features.
- Optional files:
  1. .prj—file containing coordinate system information.
  2. .sbn and .sbx--additional spatial index of features.
  3. .ain and .aih--the files that store the attribute index.

Shapefiles

- A shapefile is a set of files under a folder
- All the files of a shapefile have the same major name but with different extensions
- Spatial data (location) and attributes are stored in the separate files
- No topological data is stored
- A shapefile has only one geometric type (i.e., point, line or polygon).
- Geo-relational database

Geodatabases (Object-Oriented Databases)



- A Geodatabase stores a set of tables
- Coordinates, attributes, and topology are all stored in tables (thanks for the database technology!)
- Vector data in a Geodatabase are organized as *feature classes* and *feature datasets*
  - A feature class corresponds to a GIS layer and has the same geometric type (point, line or polygon)
  - A feature dataset have many related feature classes for an application.
  - All feature classes in a feature dataset have the same coordinate system
- Difference from shapefiles
  - Relationships among features within a feature class - topology
  - Relationship among feature classes (between layers)
    - Main pipe, lateral line, tee — business rules
  - Attribute domains (constrains on what the values the attributes can have)

Raster Data Structures

- Run length coding
  - Reduce the size of raster
  - Compression ratio depending on the redundancy along the rows
  - Guaranteed compression?
- Quad-tree
  - 2D run length coding

Raster

9	9	6	6	6	6	6	7
6	6	6	6	6	6	6	6
9	9	6	6	6	6	7	7
9	8	9	6	6	7	7	5

Run-length codes

2:9, 5:6, 1:7

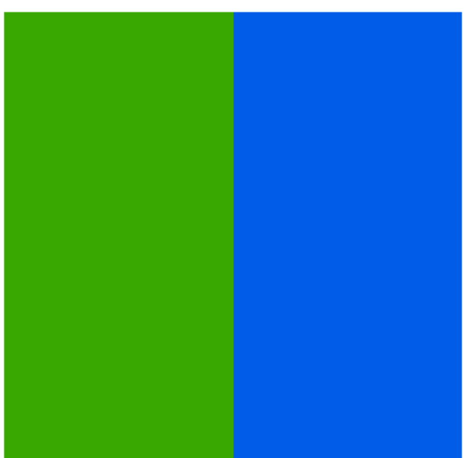
8:6

2:9, 4:6, 2:7

1:9, 1:8, 1:9, 2:6, 2:7, 1:5

Raster Attribute Table
(Figure 2-35 in textbook is not a good explanation)

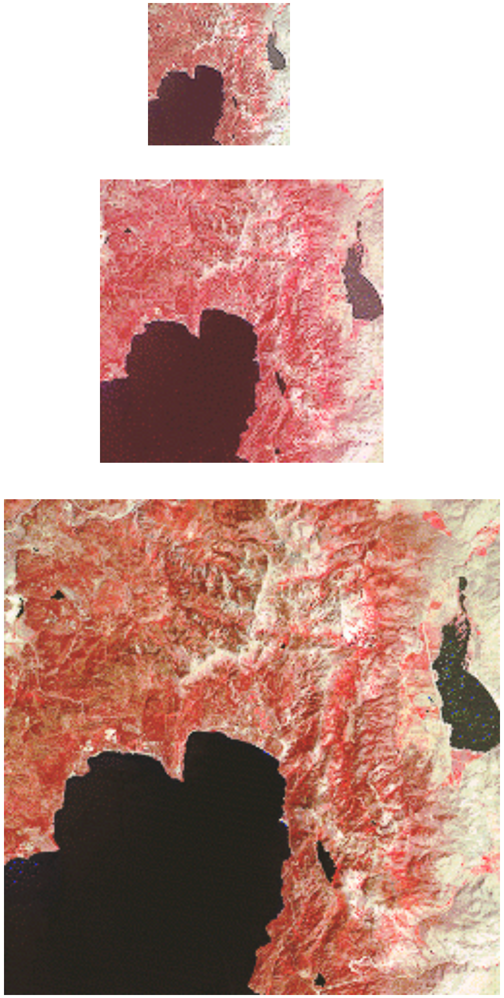
- Zone raster
  - Raster with many cells having the same value
  - Cells with the same value form a zone
  - Each unique cell value forms a zone
- Typically has an attribute table
  - Histogram of the values
  - May save storage space by storing zone IDs instead of zone value at each cell



Zone ID	Count	Value
0	50	200
1	50	325.6

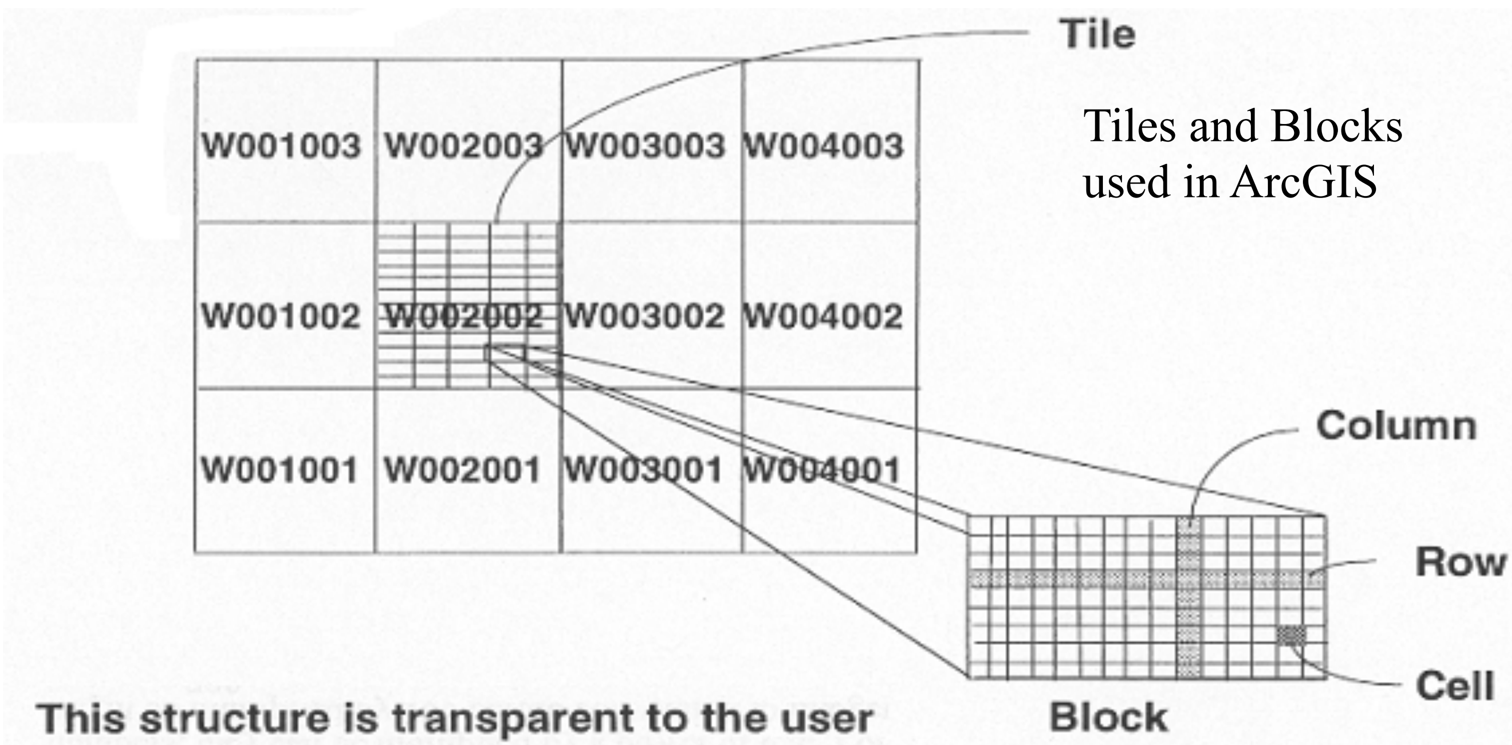
Raster Pyramids

- Raster with multi-resolution (cell size) rasters
- Fast display and analysis
  - Progressive transmission over the Internet
- Several ways to create pyramids



Handling Very Large Rasters

- Disk-based analysis--one of the advantages ArcGIS has over MATLAB
- Parallelizing analysis--Cloud-based geospatial analysis systems (Google Earth Engine)



Tiles and Blocks used in ArcGIS

Vector-Raster Comparison

Table 2-2: A comparison of raster and vector data models.

Characteristic	Raster	Vector
data structure	usually simple	usually complex
storage requirements	larger for most data sets without compression	smaller for most data sets
coordinate conversion	may be slow due to data volumes, and require resampling	simple
analysis	easy for continuous data, simple for many layer combinations	preferred for network analyses, many other spatial operations more complex
spatial precision	floor set by cell size	limited only by positional measurements
accessibility	easy to modify or program, due to simple data structure	often complex
display and output	good for images, but discrete features may show "stairstep" edges	maplike, with continuous curves, poor for images