Maps as Numbers

Getting Started with GIS
Chapter 3

Chapter 3: Maps as Numbers

- 3.1 Representing Maps as Numbers
- 3.2 Structuring Attributes
- 3.3 Structuring Maps
- 3.4 Why Topology Matters
- 3.5 Formats for GIS Data
- 3.6 Exchanging Data

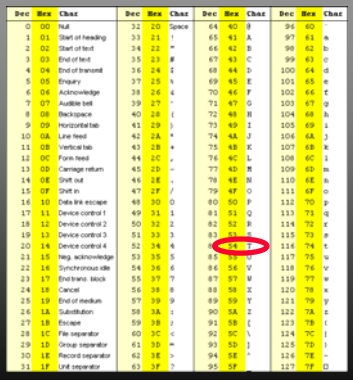
14SND5080090837

- A place given by coordinates
- A text string “14SND” followed by an easting 50800 and a northing 90837
- But how do we represent this in the computer?

Maps as Numbers

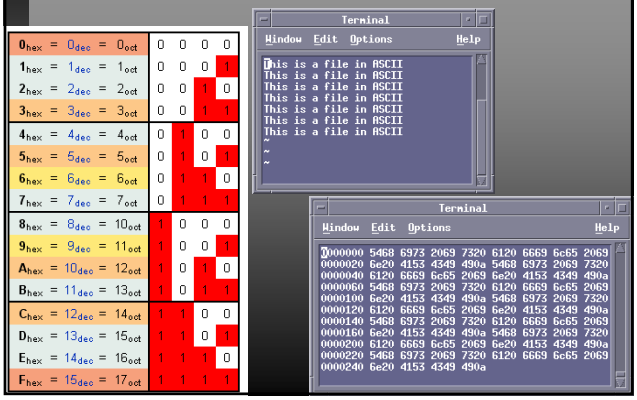
- GIS requires that both data and maps be represented as numbers
- The GIS places data into the computer’s memory in a physical data structure (i.e. files and directories)
- Files can be written in binary or as ASCII text
- Binary is faster to read and smaller, ASCII can be read by humans and edited but uses more space

ASCII Codes



Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	▯

Binary vs. HEX vs. ASCII



$0_{hex} = 0_{dec} = 0_{oct}$	0	0	0	0
$1_{hex} = 1_{dec} = 1_{oct}$	0	0	0	1
$2_{hex} = 2_{dec} = 2_{oct}$	0	0	1	0
$3_{hex} = 3_{dec} = 3_{oct}$	0	0	1	1
$4_{hex} = 4_{dec} = 4_{oct}$	0	1	0	0
$5_{hex} = 5_{dec} = 5_{oct}$	0	1	0	1
$6_{hex} = 6_{dec} = 6_{oct}$	0	1	1	0
$7_{hex} = 7_{dec} = 7_{oct}$	0	1	1	1
$8_{hex} = 8_{dec} = 10_{oct}$	1	0	0	0
$9_{hex} = 9_{dec} = 11_{oct}$	1	0	0	1
$A_{hex} = 10_{dec} = 12_{oct}$	1	0	1	0
$B_{hex} = 11_{dec} = 13_{oct}$	1	0	1	1
$C_{hex} = 12_{dec} = 14_{oct}$	1	1	0	0
$D_{hex} = 13_{dec} = 15_{oct}$	1	1	0	1
$E_{hex} = 14_{dec} = 16_{oct}$	1	1	1	0
$F_{hex} = 15_{dec} = 17_{oct}$	1	1	1	1

Terminal

Window Edit Options Help

This is a file in ASCII
This is a file in ASCII
This is a file in ASCII
This is a file in ASCII
This is a file in ASCII
This is a file in ASCII
This is a file in ASCII
~
~
~

Terminal

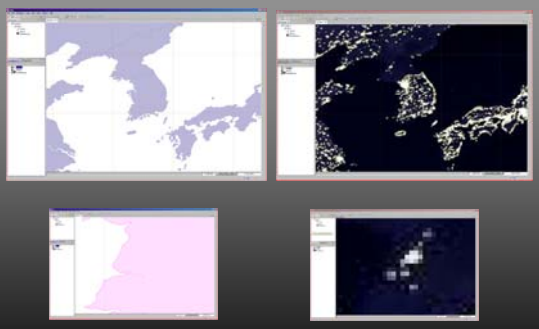
Window Edit Options Help

0000000 5468 6973 2069 7320 6120 6669 6c65 2069
0000020 6e20 4153 4349 490a 5468 6973 2069 7320
0000040 6120 6669 6c65 2069 6e20 4153 4349 490a
0000060 5468 6973 2069 7320 6120 6669 6c65 2069
0000100 6e20 4153 4349 490a 5468 6973 2069 7320
0000120 6120 6669 6c65 2069 6e20 4153 4349 490a
0000140 5468 6973 2069 7320 6120 6669 6c65 2069
0000160 6e20 4153 4349 490a 5468 6973 2069 7320
0000200 6120 6669 6c65 2069 6e20 4153 4349 490a
0000220 5468 6973 2069 7320 6120 6669 6c65 2069
0000240 6e20 4153 4349 490a

An ASCII joke

- “123” != 123

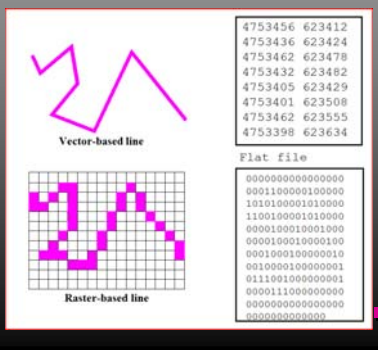
Features vs. Fields



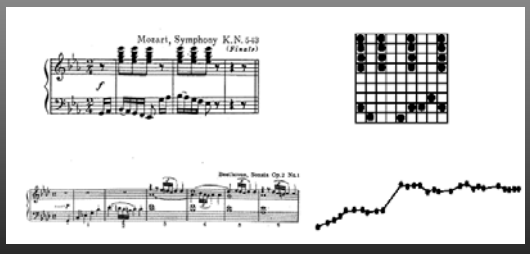
The Data Model

- A logical data model is how data are organized for use by the GIS
- GISs have traditionally used either
  - raster
  - vector

Rasters and vectors can be flat files … if they are simple



Beethoven is vector … Mozart is raster!



Spot the data structure

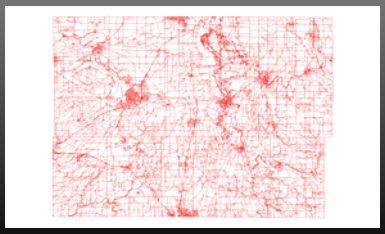
- A. RASTER
- B. VECTOR

Spot the data structure

- A. RASTER
- B. VECTOR

Spot the data structure

- A. RASTER
- B. VECTOR



Spot the data structure

- A. RASTER
- B. VECTOR

Spot the data structure

- A. RASTER
- B. VECTOR

Spot the data structure

- A. RASTER
- B. VECTOR

Raster:
Digital Orthophotoquad

Features and Maps

- A GIS map is a scaled-down digital representation of point, line, area, and volume features
- While most GIS systems can handle raster and vector, only one is used for the internal organization of spatial data
- Only one can be used in combined operations across layers

Attribute data

- Attribute data are stored logically in flat files
- A flat file is a matrix of numbers and values stored in rows and columns, like a spreadsheet
- Both logical and physical data models have evolved over time
- DBMSs use many different methods to store and manage flat files in physical files

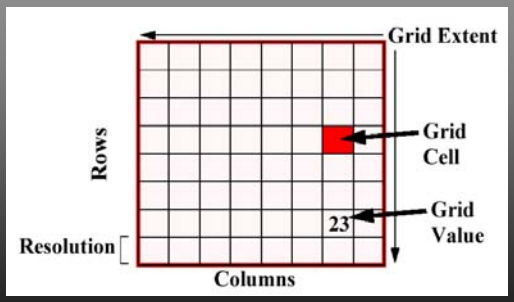
A geographical flat file

"City_fips","City_name","State_fips","State_name","State_city","Type","Capital","Elevation","Popl
990","Households","Males","Females","White","Black","Ameri_es","Asian_pi","Other","Hispanic","Age
_under5","Age_5_17","Age_18_64","Age_65_up","Nevermarry","Married","Separated","Widowed","Divorce
d","Hsehld_1_m","Hsehld_1_f","Marhh_chd","Marhh_no_c","Mhh_child","Fhh_child","Hse_units","Vacant
","Owner_occ","Renter_occ","Median_val","Medianrent","Units_1det","Units_1att","Units2","Units3_9
","Units10_49","Units50_up","Mobilehome"
05280,Bellingham,53,Washington,5305280,city,N,99,52179,21189,24838,27341,48923,411,943,1453,449,1
286,2903,7101,34814,7361,16389,18960,663,3186,4576,2758,3937,3766,6237,300,1381,22114,928,10793,1
0396,89100,371,12808,368,1198,2247,3317,1229,73235060,Havre,30,Montana,3035060,city,N,2494,10201,
4027,4955,5246,3313,15,790,65,18,116,787,2017,5949,1448,1835,4401,116,714,728,497,702,1039,1076,6
8,347,4346,319,2362,1665,54000,242,2576,57,278,651,303,86,334
01990,Anacortes,53,Washington,5301990,city,N,99,11451,4669,5506,5945,10945,62,192,154,98,233,725,
1981,6276,2449,1420,9818,161,933,916,437,711,1032,1777,76,295,4992,323,3191,1489,85300,342,3724,1
21,134,380,353,0,200
47540,MountVernon,53,Washington,5347560,city,N,99,17647,6885,8459,9188,15809,78,200,245,1916,1921
,1626,3949,10322,2460,3149,7294,291,1026,1657,802,1182,1496,1772,136,589,7167,282,3914,2971,78900
,359,4139,194,249,791,1014,171,592
50360,OakHarbor,53,Washington,5350360,city,N,99,17176,5971,8532,8444,14562,757,153,1455,249,916,2
196,3582,10164,1234,1971,8481,178,538,845,421,577,2493,1580,85,388,6173,202,2379,3592,86500,411,3
815,301,177,1004,872,0,405
53380,Minot,38,NorthDakota,3853380,city,N,1880,34644,13965,16467,18077,33098,380,724,241,81,268,2
467,6276,20983,4818,7537,14938,289,2226,2215,1783,2545,3511,3738,182,1006,15040,1078,8404,5589,56
200,279,8308,400,893,1920,1901,276,1215

A raster data model uses a grid.

- One grid cell is one unit or holds one attribute
- Every cell has a value, even if it is "missing"
- A cell can hold a number or an index value standing for an attribute
- A cell has a resolution, given as the cell size in ground units
- Often create a "mask" to cover part of rectangle not in AOI

Generic structure for a grid



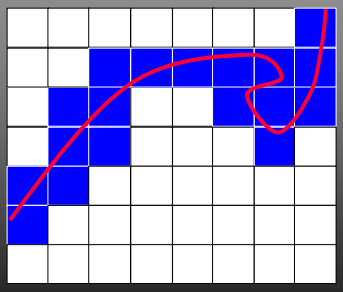
The mixed pixel problem

Grids and missing data

Rasters are faster...

- Points and lines in raster format have to move to a cell center
- Lines can become fat
- Areas may need separately coded edges
- Each cell can be owned by only one feature
- As data, all cells must be able to hold the maximum cell value
- Rasters are easy to understand, easy to read and write, and easy to draw on the screen

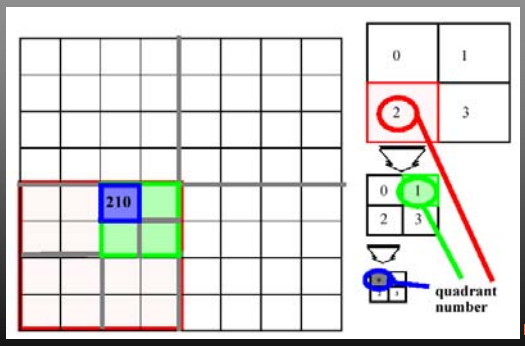
Fat lines



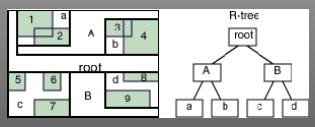
RASTER

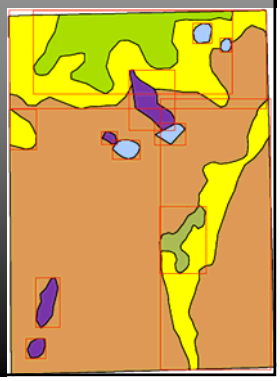
- A grid or raster maps directly onto a programming computer memory structure called an array
- Grids are poor at representing points, lines and areas, but good at surfaces
- Grids are good only at very localized topology, and weak otherwise
- Grids are a natural for scanned or remotely sensed data
- Grids suffer from the mixed pixel problem
- Grids must often include redundant or missing data
- Grid compression techniques used in GIS are run-length encoding, R-trees and quad trees

The quad-tree structure



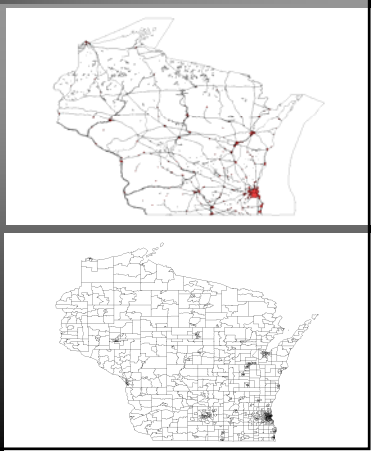
Range (R-) Trees





Vectors

Wisconsin
Top: transportation and urban places from VMAP0
Bottom: Census tracts



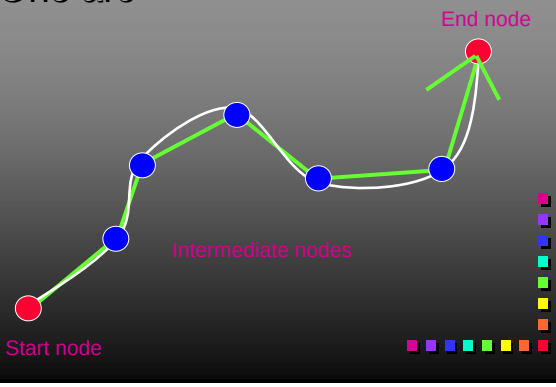
The Vector Model

- A vector data model uses points stored by their real (earth) coordinates
- Lines and areas are built from sequences of points in order
- Lines have a direction to the ordering of the points.
- Polygons can be built from points or lines
- Vectors can store information about topology

VECTOR

- At first, GISs used vector data and cartographic spaghetti structures
- Vector data evolved the arc/node model in the 1960s
- In the arc/node model, an area consist of lines and a line consists of points
- Points, lines, and areas can each be stored in their own files, with links between them
- The topological vector model uses the line (arc) as a basic unit. Areas (polygons) are built up from arcs
- The endpoint of a line (arc) is called a node. Arc junctions are only at nodes
- Stored with the arc is the topology (i.e. the connecting arcs and left and right polygons)

One arc



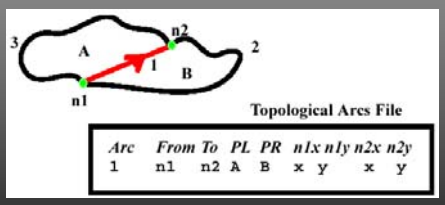
Vectors just seemed more correcter

- TIN must be used to represent volumes
- Vector can represent point, line, and area features very accurately
- Vectors are far more efficient than grids
- Vectors work well with pen and light-plotting devices and tablet digitizers
- Vectors are not good at continuous coverages or plotters that fill areas

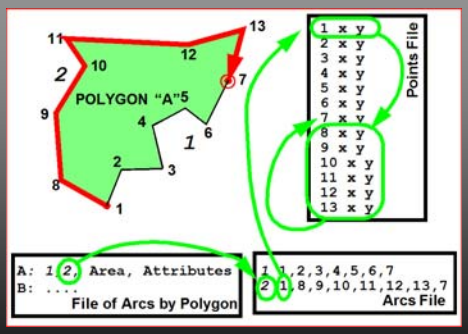
TOPOLOGY

- Topological data structures dominate GIS software
- Topology allows automated error detection and elimination
- Rarely are maps topologically clean when digitized or imported
- A GIS has to be able to build topology from unconnected arcs
- Nodes that are close together are snapped
- Slivers due to double digitizing and overlay are eliminated

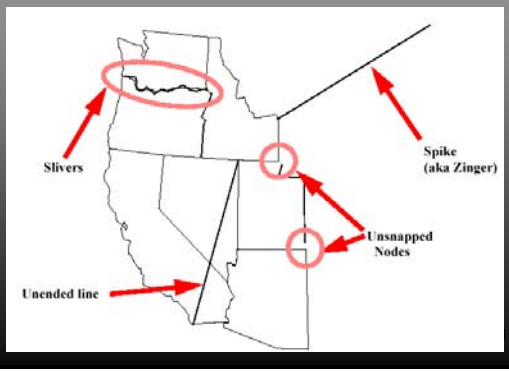
Basic arc topology



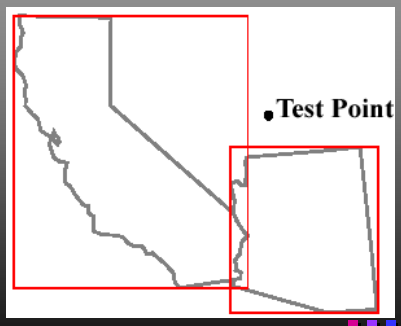
Arc/node map data structure with files



Topological errors



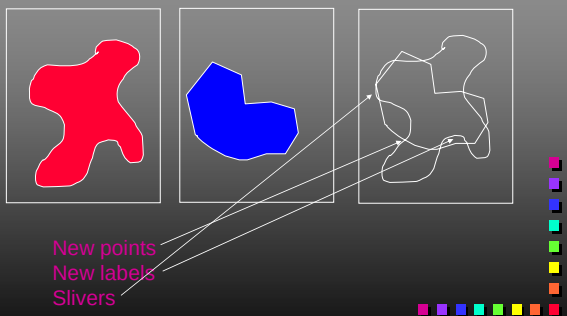
The bounding rectangle



Topology Matters

- The tolerances controlling snapping, elimination, and merging must be considered carefully, because they can move features
- Complete topology makes map overlay feasible
- Topology allows many GIS operations to be done without accessing the point files

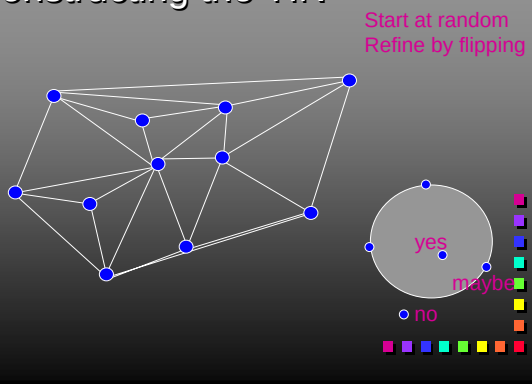
Vector overlay



Vectors and 3D

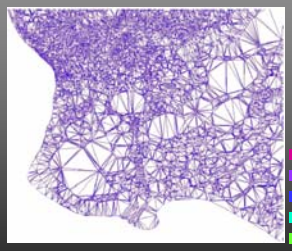
- Volumes (surfaces) are structured with the TIN model, including edge or triangle topology
- TINs use an optimal Delaunay triangulation of a set of irregularly distributed points
- TINs are popular in CAD and surveying packages

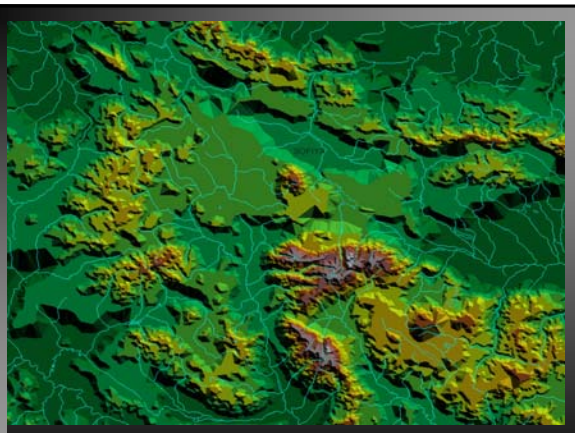
Constructing the TIN



TIN: Triangulated Irregular Network

- Way to handle field data with the vector data structure
- Common in some GISs and most AM/FM packages
- More efficient than a grid
- Surface is interpolated through the triangles





FORMATS

- Most GIS systems can import different data formats, or use utility programs to convert them
- Data formats can be industry standard, commonly accepted or standard
- Formats are part way from logical to physical data structures

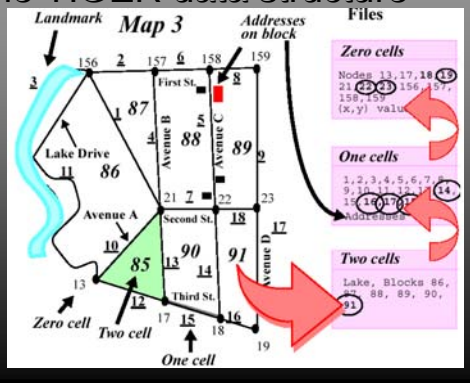
Vector Data Formats

- Vector formats are either page definition languages or preserve ground coordinates.
- Page languages are HPGL, PostScript, and Autocad DXF
- GeoPDF gaining acceptance
- Script languages like GML, SVG, KML
- True vector GIS data formats are DLG and TIGER, which has topology

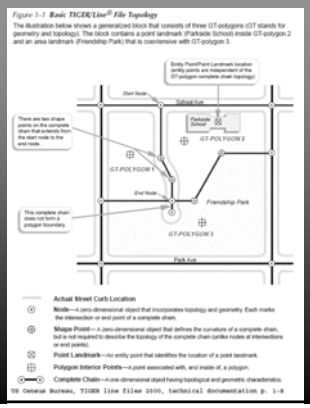
KML sample

```
<Placemark> <name>Untitled Path</name> <LineString>
    <tessellate>1</tessellate>
    <altitudeMode>relativeToGround</altitudeMode>
 <coordinates>
-134.148103,37.752967 -128.917074,38.803008
 -125.166954,39.583592 -122.137625,39.656880
-120.421783,40.036311 -118.298157,40.235316
-114.348386,40.631532 -112.670431,40.761033
-111.916045,40.681939 -110.177711,40.653055
-109.544331,40.619327 -107.155697,40.642007
-105.410526,40.421505 -103.192299,40.430138
-102.853712,40.427904 -98.168302,40.363524
-97.093391,40.308754 -94.831304,40.479175
-93.760070,40.395392 -84.913828,39.466651
-84.414888,39.387332 -81.380660,39.188551
 -80.276261,38.977744 -77.811560,38.872542
-75.062267,38.521146 -72.006956,38.101733
-66.67819,37.664687 </coordinates>
</LineString> </Placemark>
```

The TIGER data structure



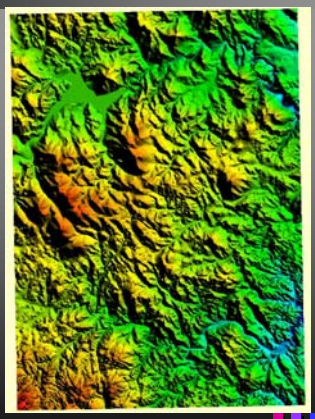
Another view



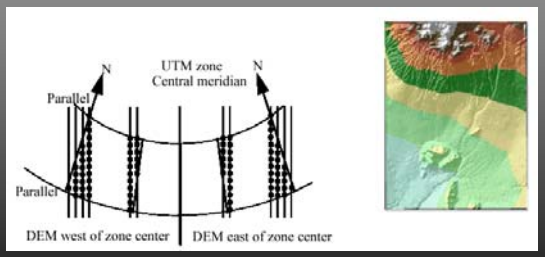
Raster Data Formats

- Most raster formats are digital image formats
- Most GISs accept TIF, GIF, JPEG or encapsulated PostScript, which are not georeferenced
- DEMs are true raster data formats
- GeoTIF now common, tif plus wrl

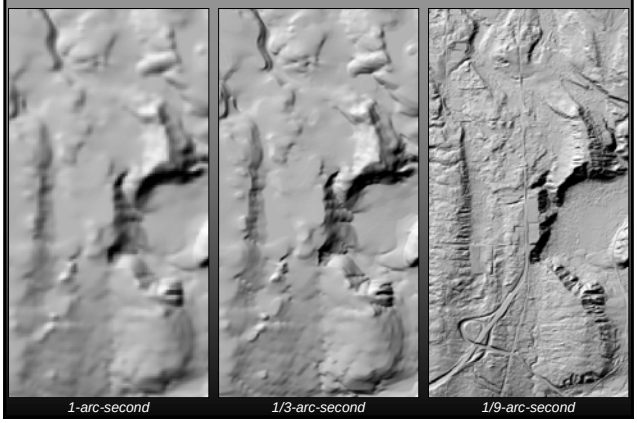
A DEM



DEMs and UTM (7.5 minute 30m)



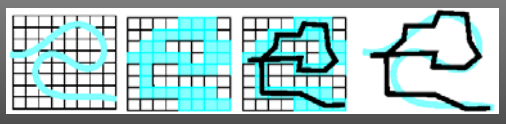
Multi-resolution NED: Puget Sound



EXCHANGE

- Most GISs use many formats and one data structure
- If a GIS supports many data structures, changing structures becomes the user's responsibility
- Changing vector to raster is easy; raster to vector is hard
- Data also are often exchanged or transferred between different GIS packages and computer systems
- The history of GIS data exchange is chaotic and has been wasteful

Vector to raster exchange errors



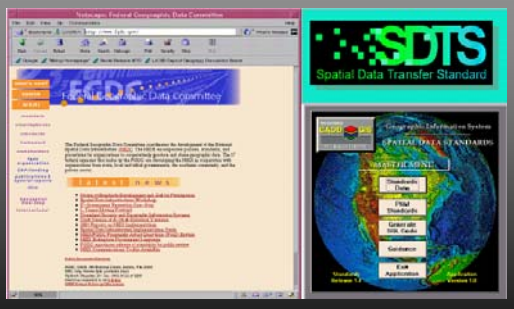
GIS Data Exchange: SDTS

- Data exchange by translation (export and import) can lead to significant errors in attributes and in geometry
- In the United States, the SDTS was evolved to facilitate data transfer
- SDTS became a federal standard (FIPS 173) in 1992
- SDTS contains a terminology, a set of references, a list of features, a transfer mechanism, and an accuracy standard

GIS data exchange: Standards

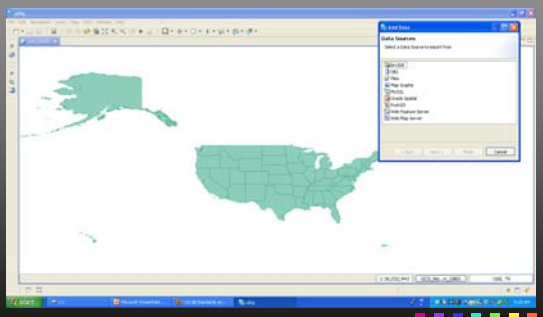
- FGDC has published metadata standards
- Both DLG and TIGER data are available in SDTS format
- Other standards efforts are DIGEST, DX-90, the Tri-Service Spatial Data Standards, and many other international standards
- OpenGIS Consortium has pioneered open standards and interoperability
- Format conversion still an issue, but much better!
- Efficient data exchange is important for the future of GIS

Transfer Standards



Interoperability: OpenGIS (www.ogc.org)

uDig: Import: SDE, Oracle, WFS, WMS, PostGIS, etc



Coming up next:

Getting the map into the computer