How to introduce « geography » into database ?

Jean-Paul Donnay
Unit of Geomatics - University of Liège
Institute of Physics, 17 Allée du 6 Août, Sart Tilman, 4000 Liège
http://www.geo.ulg.ac.be

Outline of the seminar

1. **Geographic or spatial data**
   - Form and content
2. **Geographic versus cartographic data:**
   - A simple semantic difference or two distinct realities?
3. **Mapping software solutions:**
   - CAD format + spreadsheet
4. **Desktop GIS solutions:**
   - FMS + DBMS
5. **Enterprise GIS solutions:**
   - FMS + Geo-relational
6. **First DBMS companies' solutions:**
   - Flat relational
7. **The good use of BLOB:**
   - "Spatial-enabled" database
8. **Present issues:**
   - Standards required !
   - GIS architecture of today

How to introduce « *geography* » into database? 2

Geographic data

JP Donnay – 2004

- For the man of the street, geographic data are :
  - **Names of places (« toponyms »)**
  - **Descriptive characteristics of places (figures…)**
    - Place : some part of the Earth !
    - Names : geographical identifier
    - Descriptive characteristics: geographical attributes
    - Attributes which could be attached to some identified part of the Earth !
  - **It's a part of the geographical information… but incomplete**
  - **Not suitable for geographical analysis, but for e.g. demographic or economic studies**

1	Monaco	16,428.2
2	Singapour	5,390.5
3	Malte	1,200.0
4	Maldives	973.9
5	Bahreïn	872.3
6	Bangladesh	864.5
7	La Barbade	602.4
8	Maurice	572.7
9	Nauru	495.3
10	Corée du Sud	467.6
11	Saint-Marin	411.5
12	Tuvalu	401.7
13	Pays-Bas	378.8
14	Îles Marshall	348.2
15	Liban	335.4
16	Belgique	333.3
17	Japon	333.3
18	Inde	310.8

How to introduce « *geography* » into database? 3

Geographic data

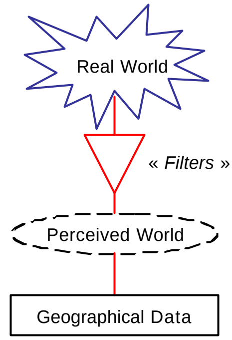
JP Donnay – 2004

- The specific part of the information required to perform geographical analysis is the « geometry » of the places
- Geometry :
  - **Geometrical description of the footprint of the geographical entities**
  - **Requiring :**
    - A surface of reference
      - Ellipsoid, sphere or plane
    - A coordinate system
      - 3D or 2D, rectangular or polar

    (jobs for geodesy and cartographic projections)

- Geographical data = Geometry + Identifier + Attributes

- How to obtain geographical data ?
  - **Several techniques working on the field – or a "snapshot" of the field (e.g. aerial photos, satellite image)**
    - Record of the "Real World"
    - No scale, but accuracy and precision
  - **Selection of the technique according to:**
    - Hardware and its precision
    - The kind of geographical entities to delineate and the required accuracy
  - **All techniques:**
    - Can always provide planimetric (X, Y) coordinates
    - Can often provide height (elevation), possibly with a different precision
    - Can only observe materialized entities (buildings, parcels, roads…)
    - Are limited by some physical restrictions (distance, clouds, shade…)
    - Provide information in digital format



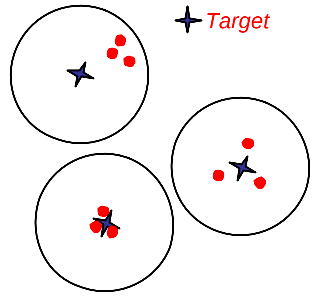
- **Topographic surveying**
  - Precise and accurate (< 1 cm)
  - Local coverage
- **Aerial photogrammetry**
  - Precise and accurate (~ 10 cm)
  - Regional to national coverage
- **Satellite remote sensing**
  - Various accuracies and precisions
    - Resolution from 60 cm to > 1 km
  - From local to worldwide coverage
- **All are supported for a decade by GNSS**
  - Global & Navigation Satellite Systems (e.g. Navstar GPS)

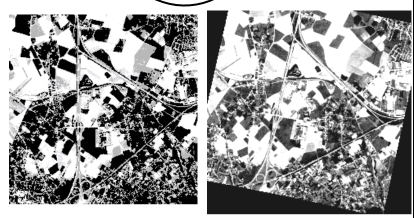
How to introduce « *geography* » into database? 6

Geographic data

JP Donnay – 2004

- Geometric precision & accuracy
  - **Precision:**
    - One measure: number of significant digits
    - Several measures of one quantity:
      - Dispersion index (standard deviation)
    - Largely due to the hardware and the measurement process
  - **Accuracy:**
    - Difference between the “real” position and the recorded position
    - Root mean square error (RMSE) in the various dimensions
    - Depends on the precision, the quality of the reference, the geometric correction process





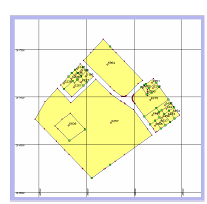
Same precision different accuracies

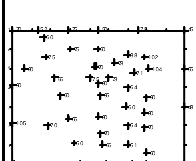
How to introduce « *geography* » into database? 7

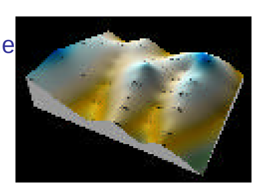
Geographic data

JP Donnay – 2004

- Geographic data types
  - **Materialized versus abstract geographic entities**
    - Abstract entities are not visible on the field or photo (ex. Boundaries)
      - Abstract entities must be located by computation / interpolation between materialized entities
      - Lower accuracy
  - **Discrete versus continuous geographic data**
    - Discrete entity:
      - Occupies a finite part of the territory
    - Continuous phenomenon (ex. Relief):
      - Occupies the whole territory
      - Infinity of points (ex. infinity of elevation value
      - Survey limited to a sample of points
      - Surface reconstituted by interpolation

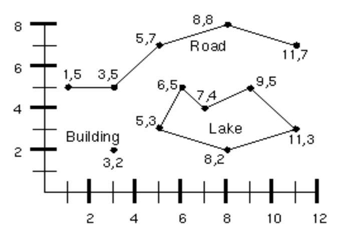




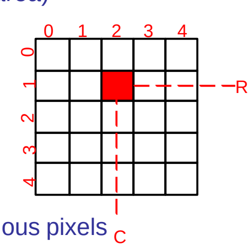


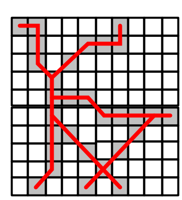
- Geometric footprint of geographic data
  - **All materialized entity has an area, delineated by a polygon**
    - Ex. Parcel, building, river, electricity pole, etc.
    - Footprint defined by a 2-D geometric primitive
  - **According to the required accuracy, the footprint of the entities can be simplified:**
    - A river or a road can be simplified to its central axis
      - Definition and record of a linear geographic entity
      - Footprint defined by a 1-D geometric primitive
    - A pole or a manhole can be simplified to its center
      - Definition and record of a punctual geographic entity
      - Footprint defined by a 0-D geometric primitive
    - This reduction of dimensionality – so called "conceptual generalization" – modifies the qualities of data and the kind of process which are supposed to be applied on these data

- Digital format for the geometries
  - **Vector format**
    - Primitive = point (x,y)
    - Coordinates: floating numbers
    - Precision: possibly high
    - Geometries:
      - Point
      - Polyline (connected segments)
      - Polygon (closed polyline + area)



  - **Raster format**
    - Primitive = pixel (area)
    - Coordinates: integer (image row & col indices)
    - Precision = resolution
    - Geometries: extracted from specific arrangements of contiguous pixels





Exrtaction of a linear network

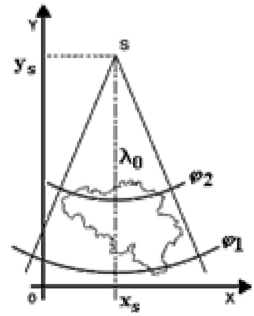
How to introduce « *geography* » into database?

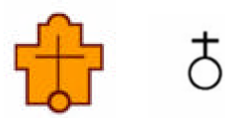
Cartographic data

- Mapping = a representation process
  - **Plane :**
    - Cartographic projection
    - From geodetic coordinates to rectangular (x, y) coordinates in km
  - **At scale :**
    - From (x, y) in km to (x, y) in mm
    - Not possible to draw all geographic entities at true scale
      - Limits of visual capacities
  - **Cartographic processes include :**
    - Cartographic generalization
    - Cartographic symbolization

  Cartographic data ? Geographic data !

  « *the map is not the territory* **»**

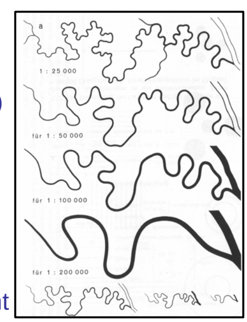




How to introduce « *geography* » into database?

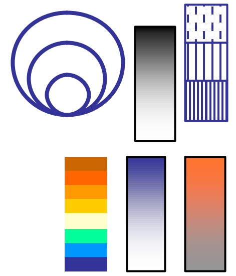
Cartographic data

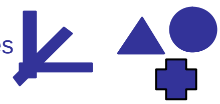
- Cartographic generalization
  - **Selection**
    - Point data : kept or withdrawn (functionally based)
    - Line / polygon data : kept if larger than minimum length / area + functional aspect
  - **Simplification (various algorithms)**
    - Deletion of small curves and small details along the lines and boundaries
    - Enlargement of small details functionally important
    - Change of dimensionality (e.g. polygon – axis)
  - **Moving**
    - Required to keep consistency in the location of objects after selection and simplification
  - **Caricature, merging, fusion, classification, etc.**
  - **Mainly depending on the final scale**

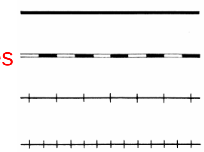


Cartographic data

- Cartographic symbolization
  - **Graphical variables & rules (semiology)**
    - Selection of the graphic variable matching the characteristics of the portrayed attribute
      - Nominal scale
      - Ordinal scale
      - Quantitative scale
    - Enlargement of the symbols in order to perceive the variations of the graphics (e.g. colour)
    - Use of pictograms instead of footprints of the features
      - Conventional pictograms imposed for some categories of objects
  - **Visualization device**
    - Graphical parameters depending on device capabilities
      - Resolution (display, printer, image setter…)
      - Colour rendering (screen, inks…)

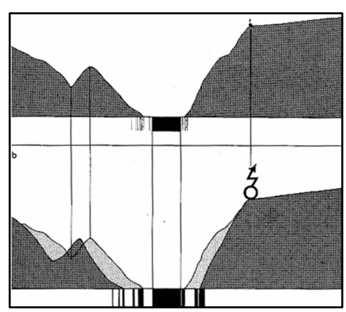


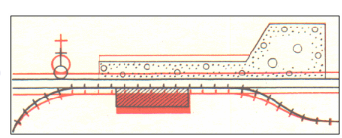




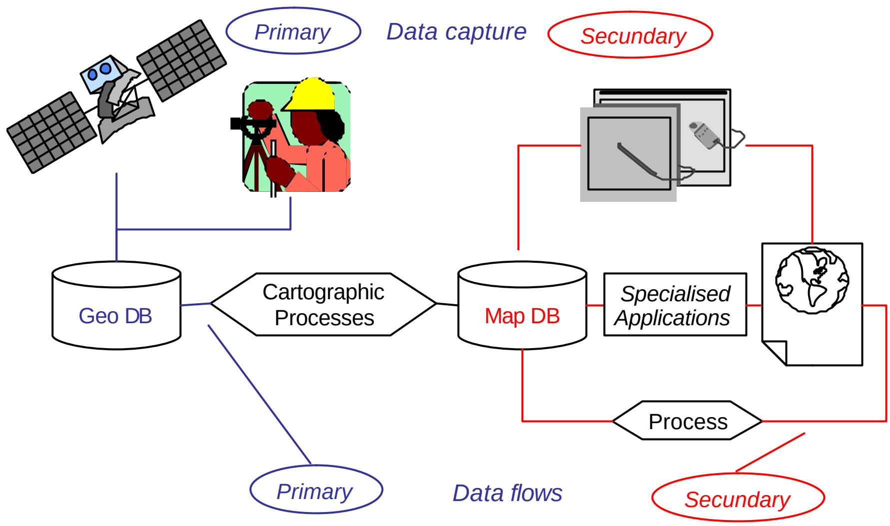
Cartographic data

- Quality of cartographic data
  - **Generalization + symbolization**
    - Change size of features
    - Change shape of features
    - Move position of features
    - Change dimensionality of features
    - Add graphical attributes to features
    - Change semantic of features
  - **Alterations cannot be propagated through the whole map**
    - Must be minimized on most important features (rivers, road axis & junctions…)
    - Must be get over on least important features (crops & forest areas, inside city blocks, etc.)
  - **Cartographic data**
    - Have positional & semantic qualities different from geographic data



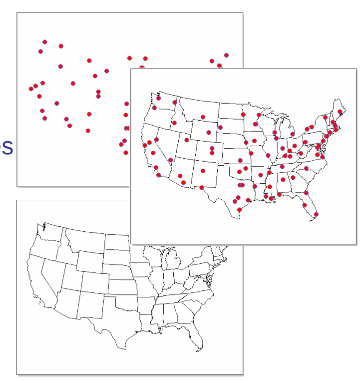


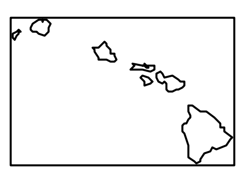
How to introduce « *geography* » into database? 14
Cartographic data
JP Donnay – 2004



How to introduce « *geography* » into database? 15
Mapping software
JP Donnay – 2004

- Hybrid / Dual Data Model
  - **Geometric data :**
    - Position :
      - Geodetic or rectangular coordinates
      - CAD (vector) format (ASCII)
      - Sequential organization
    - Separate layers :
      - Different geometries
      - Map layers + Map layout
    - Logical relationship :
      - Composition
      - No geometric relationship
  - **Attribute data :**
    - ASCII files or spreadsheet files
    - One file or one sheet / One geometric layer
    - Sequential organization





How to introduce « *geography* » into database? 16

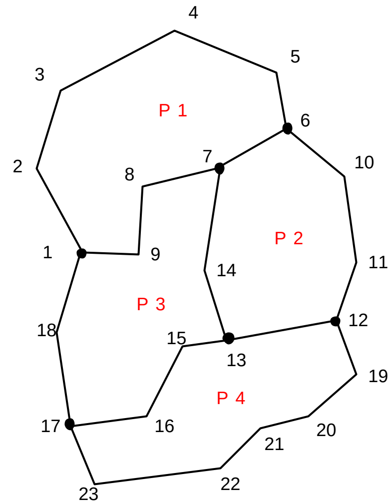
Mapping software

JP Donnay – 2004

P1
x_{1}, y_{1}
x_{2}, y_{2}
x_{3}, y_{3}
x_{4}, y_{4}
x_{5}, y_{5}
x_{6}, y_{6}
x_{7}, y_{7}
x_{8}, y_{8}
x_{9}, y_{9}
x_{1}, y_{1}
Separator
P2
x_{6}, y_{6}
x_{10}, y_{10}
x_{11}, y_{11}
x_{12}, y_{12}
x_{13}, y_{13}
x_{14}, y_{14}
x_{7}, y_{7}
x_{6}, y_{6}
Separator
P3
x_{7}, y_{7}
x_{14}, y_{14}
x_{13}, y_{13}
x_{15}, y_{15}
x_{16}, y_{16}
x_{17}, y_{17}
x_{18}, y_{18}
x_{1}, y_{1}
x_{9}, y_{9}
x_{8}, y_{8}
x_{7}, y_{7}
Separator
P4
x_{12}, y_{12}
x_{19}, y_{19}
x_{20}, y_{20}
x_{21}, y_{21}
x_{22}, y_{22}
x_{23}, y_{23}
x_{17}, y_{17}
x_{16}, y_{16}
x_{15}, y_{15}
x_{13}, y_{13}
x_{12}, y_{12}
Separator

< End of File >

Simple sequential structure

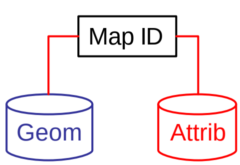


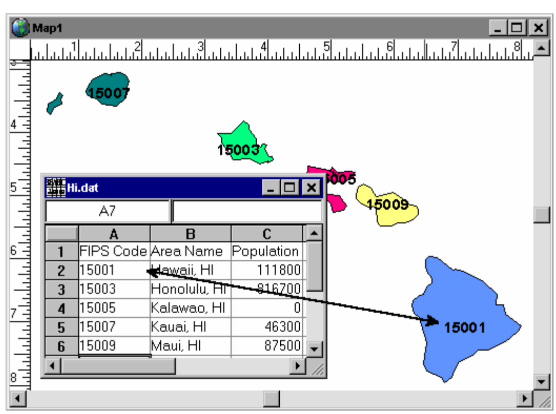
How to introduce « *geography* » into database? 17

Mapping software

JP Donnay – 2004

- **Cartographic ID :**
  - Separate access to geometry vs. attributes
  - 1 to 1 relationship between geometry and attribute thanks to the cartographic ID (number)



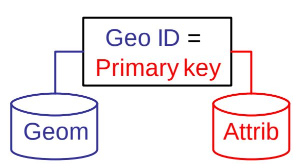


GIS

- GIS or GI-S or G-IS ?
  - **Ambiguity about Geographical Information Systems**
    - Vendors and laymen viewpoint : GI Software (+ hardware + data)
    - US academic viewpoint : GI Science (geographic processes & analysis)
    - Computer science viewpoint (French-speaking literature) : specialized Information System (cf. multimedia Information Systems)
  - **Information System**
    - Connected to the theory of organization modeling
      - Processing System – Information System – Decision System
    - Connected to database theory
      - System made of (one or) several databases managed by one or several DBMS, using similar or distinct models and languages
  - **Question at issue : how to put geometry into the database ?**
    - In vector format ? In raster format ? With positions and geometric relationships ? Keeping connection with attributes ?
    - No operational solution before the mid of the '90 !

Desktop GIS

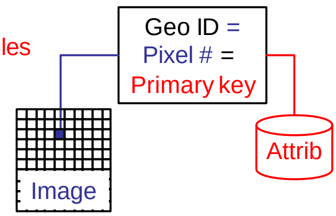
- Hybrid / Dual Data Model
  - **Very similar to mapping software solutions**
  - **The main difference : the management of the attributes**
    - Attribute values are managed by a DBMS (desktop : Access, Dbase…)
      - One main table (relation) where the primary key = geographic ID
      - Possibility to join other tables with SQL jointure



    - Transactions and requests on attributes are performed thanks to SQL interface :
      - If the result of the SQL select instruction contains geographic IDs :
        - » The corresponding geometries are displayed in a map
      - Extended SQL statements often implemented to support simple spatial requests
        - » Distance, buffer, area, etc.

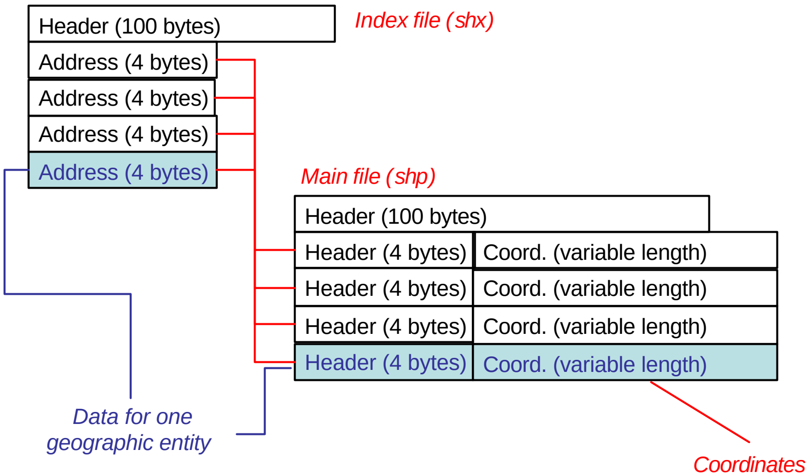
How to introduce « *geography* » into database? 20
Desktop GIS
JP Donnay – 2004

– **Geometries in vector or raster format**
- Vector based GIS
  – Geometries managed by a File Management System (FMS)
    » CAD format or ASCII files
    » Generally proprietary binary format to improve efficiency
  – Position : geodetic or rectangular coordinates
  – Logical relationships :
    » Composition
    » No geometric relationship
- Raster based GIS
  – Spatial information stored in image files
    » Spatial requests performed by image processing facilities
  – Attributes managed by DBMS
    » Attribute requests performed by SQL statements



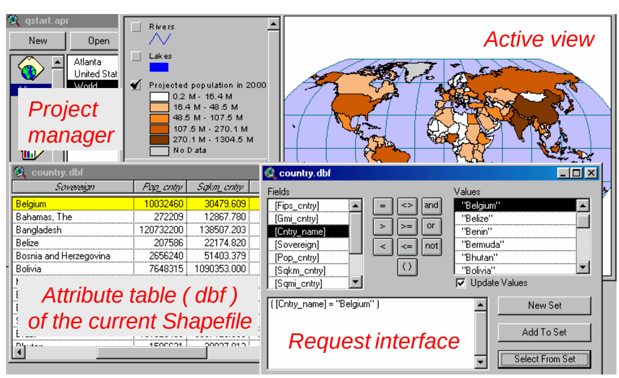
How to introduce « *geography* » into database? 21
Desktop GIS
JP Donnay – 2004

- ArcView : an example of desktop vector based GIS software
  – **Shapefile :**
  - Proprietary format of data files used by ArcView
  - One *shapefile* is made of 3 files : 2 for geometries – 1 for attributes
    – *Main file* (suffix shp) :
      » Collection of coordinates in 2 (x, y), 3 ($x, y, z/m$) or 4-D (x, y, z, m)
      » Sequential presentation of the geographic entities
      » One type of geometry / file (but many types are available)
    – *Index file* (suffix shx) : addresses of the beginning of each geographic entity in the main file (pointers for direct access)
    – dBase file (suffix dbf) : main table of attribute values
      » One tuple = one geographic entity
      » Sequential number = geographic ID (= sequential number of geometries in the Main file)
      » Transactions and requests : interactive SQL interfaces



Format used by ArcView to manage geometries of geographic entities

- **Structure of the application**
  - Project :
    - Collections of *themes* (one theme = one Shapefile) selected by the user
  - View (in the sense given by database theory) : view of the data contained in one theme of the project :
    - Attribute table
    - Map of the geographic entities
    - Attached layout (legend)
- **Interfaces**
  - Project manager
  - Interactive request interface



How to introduce « *geography* » into database?

Desktop GIS

- Idrisi : an example of desktop raster based GIS software
  - **Reference image :**
    - Image of the geometries of the geographic entities
    - Any geographic entity has an area (min = 1 pixel)
    - Mandatory for all operations (cannot be altered)
    - Pixel value = geographic ID
      - Integer (>= 0)
      - Same pixel value = same geographic entity
        - » True for neighbor or not neighbor pixels
      - Number of distinctive pixel values = number of geographic entities
      - Pixel value = primary key of the attribute table

Reference image
Pixel # = Geographic ID

2	2	3	3
2	2	3	5
1	2	2	5
6	6	1	5

Id.	Fréquence	Attribut 1	...	Attribut n
1	2	...	...	...
2	6	...	...	...
3	3	...	...	...
4	0	...	...	...
5	3	...	...	...
6	2	...	...	...

Attribute table connected to the reference image

How to introduce « *geography* » into database?

Desktop GIS

- **Attribute management**
  - Attribute values managed by DBMS Access
  - One main table containing :
    - One tuple = one geographic entity
      - » At least as many tuples as pixel values in the reference image
    - One field = geographic ID (integer)
  - SQL interface
- **Thematic image**
  - The result of a SQL request is visualized in a thematic image in 2 steps:
    - Creation of a 2 columns Look Up Table by the projection of
      - » The ID attribute
      - » The thematic attribute converted in number
    - Assignation of the LUT to the reference image to carry out the thematic image

Attribute table

Id.	Fréq.	Population	...
1	2	140	
2	6	120	
3	3	60	
4	0	45	
5	3	90	
6	2	180	

Reference image

2	2	3	3
2	2	3	5
1	2	2	5
6	6	1	5

Population / Frequency

SQL instruction

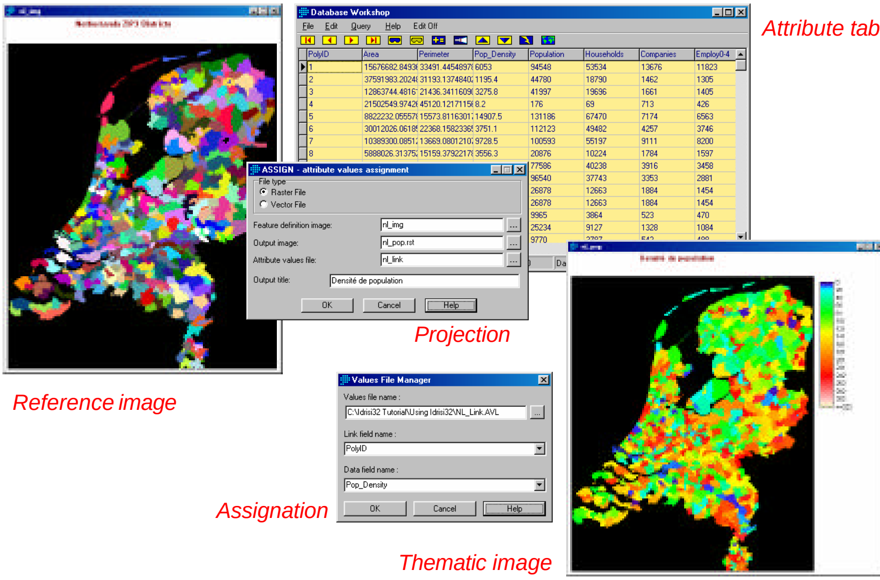
LUT

Id.	Population
1	70
2	20
3	20
5	30
6	90

Thematic image

20	20	20	20
20	20	20	30
70	20	20	30
90	90	70	30

Desktop GIS



Attribute table

Reference image

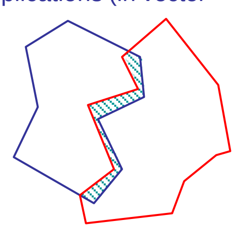
Projection

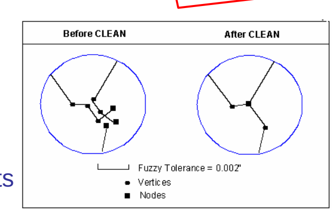
Assignation

Thematic image

Topology

- Extended spatial requests
  - **Until now, geometries are limited to position and logical relationships (coordinates, composition)**
    - These elements limit spatial requests to simple applications (in vector format)
  - **Extended spatial requests require more information about geometry : the topology**
    - Neighbourhood
    - Superposition
  - **Topology of a map can be extracted automatically provided that:**
    - Layer contains only one kind of geometry (polylines or polygons)
    - Positions are cleaned to avoid dangling nodes and sliver polygons
    - Geometric primitives are seen as elements of the graph theory (node, arc, path)





How to introduce « *geography* » into database? 28
Topology
JP Donnay – 2004

Points Dictionary

1: x_1, y_1	11: x_{11}, y_{11}
2: x_2, y_2	12: x_{12}, y_{12}
3: x_3, y_3	13: x_{13}, y_{13}
4: x_4, y_4	14: x_{14}, y_{14}
5: x_5, y_5	15: x_{15}, y_{15}
6: x_6, y_6	16: x_{16}, y_{16}
7: x_7, y_7	17: x_{17}, y_{17}
8: x_8, y_8	
9: x_9, y_9	
10: x_{10}, y_{10}	

List of points / arc
A1: 1, 2, 3, 4
A3: 5, 6
A4: 7, 8
A6: 9
A7: 10, 11
A8: 12
A9: 13, 14, 15, 16, 17

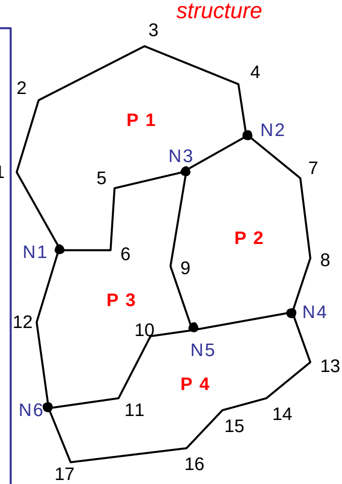
Nodes Dictionary
N1: x_{N1}, y_{N1}
N2: x_{N2}, y_{N2}
N3: x_{N3}, y_{N3}
N4: x_{N4}, y_{N4}
N5: x_{N5}, y_{N5}
N6: x_{N6}, y_{N6}

List of arcs / polygon
P1: A1, A2, A3
P2: A4, A5, A6, A2
P3: A6, A7, A8, A3
P4: A9, A7, A5

File of arcs with topology

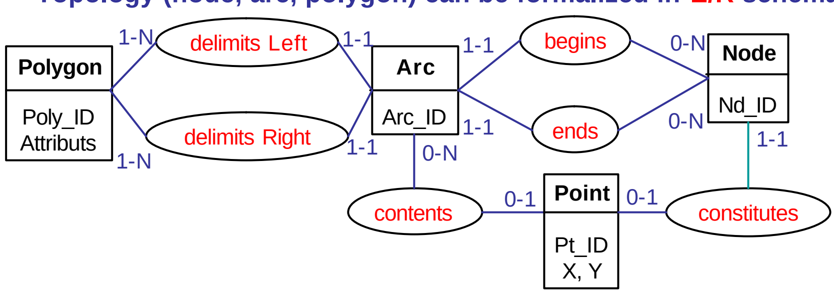
					Nbr Pts
A1	N1	N2	0	P1	4
A2	N2	N3	P2	P1	0
A3	N3	N1	P3	P1	2
A4	N2	N4	0	P2	2
A5	N4	N5	P4	P2	0
A6	N5	N3	P3	P2	1
A7	N5	N6	P4	P3	2
A8	N6	N1	0	P3	1
A9	N4	N6	0	P4	5

POLYVRT structure



How to introduce « *geography* » into database? 29
Geo-relational
JP Donnay – 2004

- Relational schema
  - Topology (node, arc, polygon) can be formalized in E/R schema

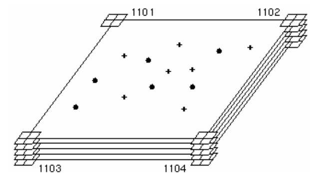


  - This schema can be implemented in a Relational DB
    - However the positional data (nodes and points coordinates) are only retrieved through a huge number of jointures !
  - The Geo-Relational solutions
    - Only conserve topologic information in a relational DB
    - Let positional information in binary files (direct access, proprietary format)

How to introduce « *geography* » into database?

Geo-relational

- Arc-Info (until vers. 7) : an example of enterprise GIS using a geo-relational implementation
  - **Geometries :**
    - Topologic primitives : isolated nodes, arcs & polygons
      - Identified by internal numbers
      - Contain positional data (nodes & arcs) & topologic information (arcs)
      - Stored in dedicated files (proprietary format)
    - User's geometries = *feature classes*
      - Nodes, arcs, polygons + composed features (regions…)
      - Receive User IDs
      - IDs (internal + user), topologic info (if any) & computed geometric attributes of feature classes are stored in tables (proprietary format)
    - Coverage :
      - Portion of space, referenced by tic points and MBR coordinates
      - All geometries (collection of files) concerning one theme are gathered in one coverage

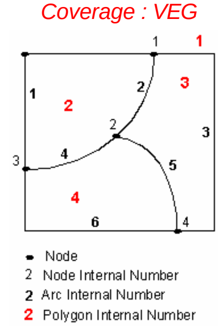


How to introduce « *geography* » into database?

Geo-relational

- **Attributes & topology :**
  - Info : R-DBMS managing tables of feature classes
  - Feature class table : automatically generated table
    - View of the corresponding table contained in the coverage
      - » IDs, topologic info, geometric attributes
    - Generation of one table for every feature class contained in every coverage
  - User's attributes can be added to any feature class table

Coverage : VEG



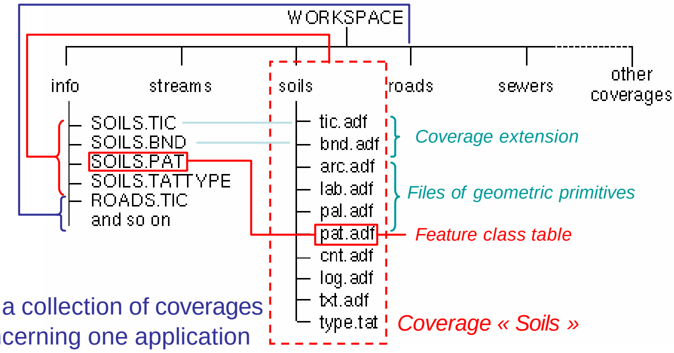
FNODE	TNODE	LPOLY	RPOLY	LENGTH	VEG#	VEG-ID
1	3	2	1	5.0	1	1
1	2	3	2	2.0	2	2
1	4	1	3	10.0	3	3
2	3	4	2	1.9	4	4
4	2	4	3	3.9	5	5
3	4	4	1	5.1	6	6

Feature class table (arc) available in « Info »
All these attributes are generated by the system

How to introduce « *geography* » into database?

Geo-relational

– **Structure of the GIS :**

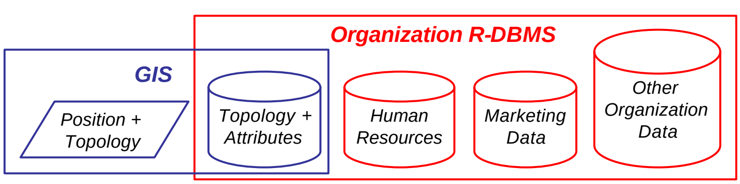


- Workspace : a collection of coverages (themes) concerning one application constitutes a workspace
- Directories : every coverage is a directory in a workspace
  - The views of all tables are gathered in one directory Info, the only directory “open” to the user for SQL transactions
- Application modules : Arc-Info offers a complete collection of software modules able to exploit both coverage and info directories, in order to perform extended spatial requests (notably based on topology relationship)

How to introduce « *geography* » into database?

Integrated GIS model

- Hybrid GIS
  - **e.g. Geo-relational**
    - FMS (position + topology) + R-DBMS (topology + attributes)
  - **Problems for the organization :**
    - A part of data are out of the DBMS
      - No guarantee about : security, daily data management, archive & back-up, etc.
    - Depending on a dedicated software
      - Hardware & software compatibility
      - Organization data are stored in proprietary formats
      - No guarantee about extended maintenance and viability



Integrated GIS model

- Integrated model
  - **Required by organizations**
  - **R-DBMS : position + topology + attributes ?**
    - (see E/R schema p. 29)

Polygon (Poly_ID, Attribute_1, …, Attribute_n)
TablePolArc (Poly_ID, Arc_ID, N°_Order_Arc)
Arc (Arc_ID, #Nd_Begin, #Nd_End, #Poly_Left, #Poly_Right)
TableArcPt (Arc_ID, Pt_ID, N°_Order_Pt)
Node (Nd_ID, #Pt_ID)
Point (Pt_ID, X, Y)

    - Drawback : number of jointures to obtain the coordinates of any point
      - Due to the topology !
    - Topology requested for extended spatial analysis (a few % of requests !)
      - Topology can be automatically extracted from clean geo-data
  - **Flat relational : R-DBMS : position + attributes**
    - Topology on the fly if requested !

Flat relational

- Oracle Spatial Cartridge
  - **« *Spatial Data Option* » (SDO) of Oracle R-DBMS (mid of '90s)**
  - **Spatial cartridge concept covers :**
    - A strictly relational implementation (*normalized schema*)
    - Vector data model : simple & composed features, without topology
      - Compatible with the future OGC standard (see below)
    - A method for spatial indexing (based on quadtrees)
    - An extension of SQL statements, in order to allow spatial requests and geometric processes
  - **Data model (3 levels)**
    - Element (primitive) : point data (1), line data (2) & polygon data (3)
    - Geometry (user feature) : made of one or several elements, assigned to only one user ID (main key for the jointure with attribute tables)
    - Layer : geo-referenced support of one or several geometries, supporting the spatial index
    - 4 tables generated by specific SQL instructions
      - Spatial indexation is automatically achieved by the system

Flat relational

- **Structure of the tables**
  - SDO-LAYER

OrdCnt	Level	NumTiles	CoordSys
number	number	number	varchar

- **OrdCnt** : nbr of columns for coordinate data in the table SDO-GEOM
- **Level** : quadtree tessellation level
- **NumTiles** : option for variable size tessellation
- **CoordSys** : name of the reference coordinate system

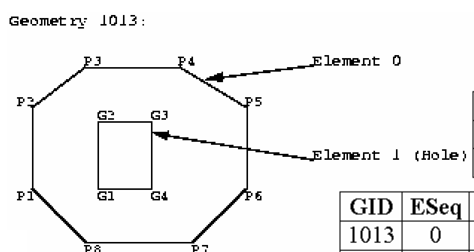
  - SDO-GEOM

GID	ESeq	EType	Seq	X1	Y1	…	Xn	Yn
number	number	number	number	number	number	…	number	number

- **GID :** Geometry ID
- **ESeq** : order n° of the current element in the geometry (0,…)
- **EType** : element type (1, 2 or 3)
- **Seq** : order n° of the current tuple in the element (0,…)

  - SDO-DIM
    - Table documenting the dimensions used to define the layer
  - SDO-INDEX
    - Table concerning the quadtree spatial index

Flat relational



OrdCnt	Level	NumTiles	CoordSys
4	1	NULL	

DimNum	LB	UB	Tolerance	DimName
1	0	100	0.05	X axis
2	0	100	0.05	Y axis

GID	ESeq	EType	Seq	X1	Y1	X2	Y2
1013	0	3	0	P1(X)	P1(Y)	P2(X)	P2(Y)
1013	0	3	1	P2(X)	P2(Y)	P3(X)	P3(Y)
1013	0	3	2	P3(X)	P3(Y)	P4(X)	P4(Y)
1013	0	3	3	P4(X)	P4(Y)	P5(X)	P5(Y)
1013	0	3	4	P5(X)	P5(Y)	P6(X)	P6(Y)
1013	0	3	5	P6(X)	P6(Y)	P7(X)	P7(Y)
1013	0	3	6	P7(X)	P7(Y)	P8(X)	P8(Y)
1013	0	3	7	P8(X)	P8(Y)	P1(X)	P1(Y)
1013	1	3	0	G1(X)	G1(Y)	G2(X)	G2(Y)
1013	1	3	1	G2(X)	G2(Y)	G3(X)	G3(Y)
1013	1	3	2	G3(X)	G3(Y)	G4(X)	G4(Y)
1013	1	3	3	G4(X)	G4(Y)	G1(X)	G1(Y)

Record of a composed polygon
(2 elements)
in Spatial Cartridge

Tables :
SDO-LAYER
SDO-DIM
SDO-GEOM

How to introduce « *geography* » into database?

“Spatial enabled” databases

- Success of Oracle SDO
  - **Wishes of the organizations**
    - To keep their “costly” data in standard and reliable DBMS
    - To have access to transparent schemas
  - **Limits of SDO**
    - Large redundancy (coordinates)
    - Limited features types (versus e.g. geo-relational capabilities)
    - Poor efficiency of spatial indexing
    - Very limited spatial analysis and geometric processing
- Evolution of the relational model and R-DBMS
  - **BLOB (*Binary Long OBject*)**
  - **Object-relational extensions**
  - **SQL 3 / MM**
  - **Stored procedures**

How to introduce « *geography* » into database?

“Spatial enabled” databases

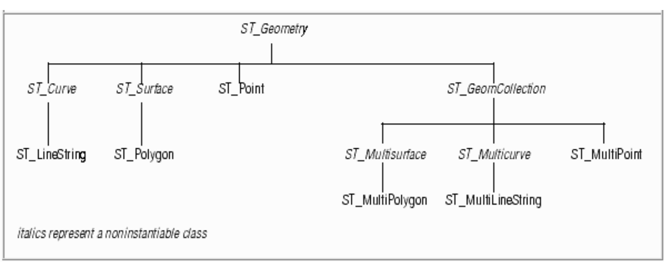
- Examples of object-relational statements
  - **Abstract types & reference sharing :**
    - (TYPE) **Point** (Pt-ID Int, X Float, Y Float)
    - (TYPE) **Node** (Nd-ID Int, Position REF(Point))
    - (TYPE) **Polyline** (Pll-ID Int, List (Point), Attribute-1 …)
  - **Methods :**
    - (TYPE) **Segment** (Seg-ID Int, Pt1 REF (Point), Pt2 REF (Point),
      FUNCTION Compute_Length (S Segment) RETURNS (Float)
      …END FUNCTION
    - Example of request :
      SELECT Compute_Length (S)
      FROM TableSegment S
  - **Heritage & specialization :**
    - (TYPE) **Polyline** (PLL-ID Int, List (Point))
    - (TYPE) **RoadAxis** UNDER Polylign (Nom String)

How to introduce « *geography* » into database? 40
"Spatial enabled" databases
JP Donnay – 2004

- Advantages for geographic data
  - **Multiplication of geographic features**
    - Definition of any kind of feature thanks to abstract types
      - So called "extended data types"
    - O-O design of the database (UML + dedicated spatial extensions)
      - Geographic features (class & objects) close to the users' concerns
  - **Storage of complex features in BLOB**
      - Including Images !
  - **Number of jointures dramatically reduced**
    - Thanks to the sharing by reference
  - **Definition of spatial and geometric procedures**
    - Written in dedicated languages based on SQL (or C or Java)
    - Stored on the server
- Emergence of "spatial enabled" databases
  - **From the leader companies of DBMS : Oracle, Informix, IBM…**
  - **Repositioning of the standard GIS companies : ESRI, MapInfo…**

How to introduce « *geography* » into database? 41
"Spatial enabled" databases
JP Donnay – 2004

- **Informix Spatial DataBlade Module**
  - **Extension to Informix DBMS (IDS or IIF) providing extended spatial data types and associated functions available via SQL 3**



Data types class diagram in Spatial Datablade

  - **Spatialization of the database is performed thanks to :**
    - **A "spatial column" in the feature table**
    - **A couple of metadata tables giving :**
      - **Information on the type of feature in the spatial column**
      - **Information on the geo-reference system**

How to introduce « *geography* » into database? 42
“Spatial enabled” databases
JP Donnay – 2004

- **The “spatial column”**
  - Corresponds to one field of the table containing the feature ID and possible feature attributes (= “spatial table”)
  - Can only refer to one type of spatial data
    - Different tables must be used to store the various spatial data types
  - Contains the geometry (position data) of the feature or (generally) a pointer to a BLOB (large geometries)

Spatial table (ex. polygone)					
se-row-id	Area-id	Name	Size	Type	Zone
					<ST_Polygon>

Spatial table (ex. point)			
se-row-id	Site-id	Name	Location
			<ST_Point>

- **The “Geometry table”**
  - Specify the name and the content of the spatial columns in all s patial tables of the database
- **The “Reference table”**
  - Specify the spatial reference for all spatial tables of the data base

How to introduce « *geography* » into database? 43
“Spatial enabled” databases
JP Donnay – 2004

Spatial table (ex. polygone)					
se-row-id	Area-id	Name	Size	Type	Zone
					<ST_Polygon>

Geometry columns table					
DB-name	User-id	Table-name	Col-Spat-name	Col-type	SR-id

Spatial Reference Table											
SR-id	Descript.	Ref	Ref-id	FOX	FOY	XYUnit	FOZ	ZUnit	FOM	MUnit	SRText
											<WKT>

False origin & units in x, y, z, m

System of reference in Well-Known Text format (OGC standard)

Spatialization of a table & tables of metadata according to the Spatial DataBlade

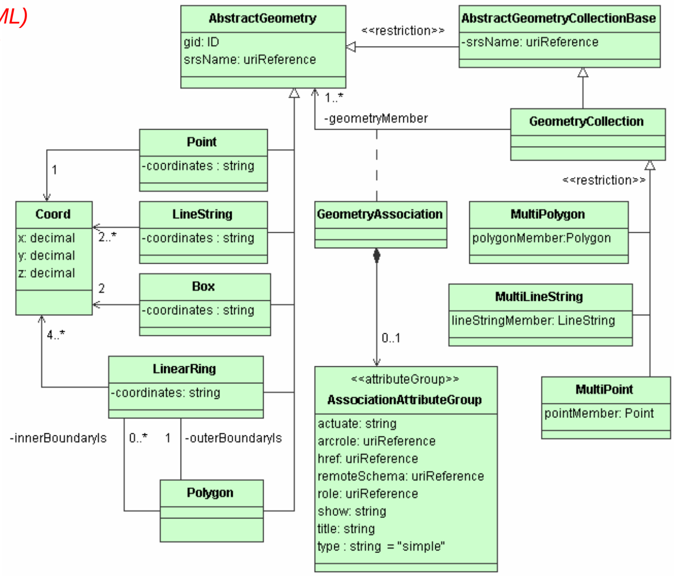
Standards required !

- Pitfalls of object-relational solutions
  - **Diversity of spatialization approaches**
    - Extended data types : any kind of spatial feature could be created, according to user's domain
    - BLOB : sophisticated data structures could be stored in a "black box"
    - Spatial procedures : emergence of un-standard SQL statements
    - Etc.
- The increasing role of spatial data (economic & strategic)
  - **Clear distinction between data providers (specialized agencies) and data users**
    - Importance of data quality and metadata
    - Requirements for data exchange and data remote access
      - Internet plays a key role in distributed databases
  - **Projects of spatial data clearinghouse, spatial data gateways, spatial data infrastructures**
- Everything calls for standards !

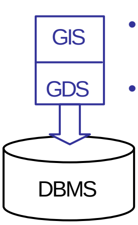
Standards required !

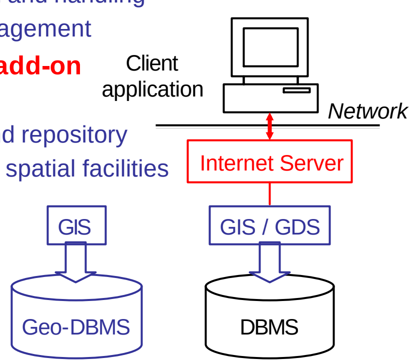
- An international concern for about 20 years
  - **ISO TC 211, CEN TC 287, national and sectoral committees**
    - Antagonisms between NMA in the public sector, and between many middle-size private companies
- Significant advances since the end of the '90s
  - **FGDC (Federal Geographic Data Council) : metadata standards**
  - **OGC (Open GIS Consortium) :**
    - Member driven, "non profit" industry consortium (258 members)
    - Quite slow advances until the end of the '90s : US public agencies + international "GIS" companies + universities
    - Since the end of the '90s : new members (Oracle, IBM, Microsoft, etc.) impose the standards, on the basis of :
      - Object definition of spatial features
      - Spatial enabled O-R databases
      - Internet interoperability
- Norms ISO CD 191xx
  - **Published from 2001, and still in progress**

Object schema (UML) of OGC geometries



- GIS architecture of today
  - **Spatial enabled database is the rule**
    - Oracle solutions (Spatial Object - 8i, 9i, [10g]) = 90 % implementations !
    - DBMS progressively perform all tasks related to spatial data
      - Long transactions, spatial indexation
      - Topologic features extraction and handling
      - Geographic image data management
  - **Specialized GIS software is an add-on covering DBMS**
    - Database design, data loading and repository
    - Spatialization of R-DBMS without spatial facilities (GDS : GeoData Server)
    - Specialized spatial analysis and mapping





Standards required !

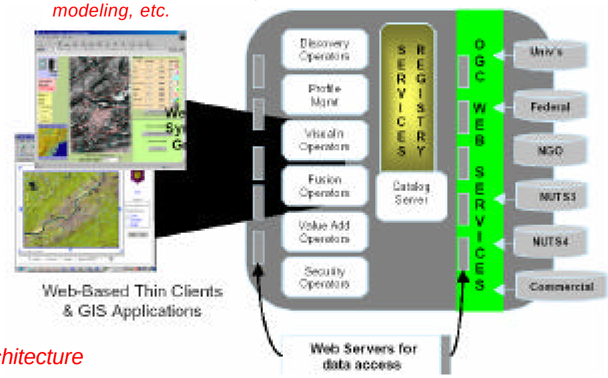
- **Standardized metadata and normalized schemas are published on Internet to set up future distributed GIS**
  - Emergence of services allowing remote access to various objects :
    - Spatial data selected by the producers
    - Interoperable components processing standardized data

Clients for visualization, Analysis, modeling, etc.

Examples of interoperable components

Services & Data metadata

Multiple format Data



Example of geo-data services architecture

Thank you for your attention …