Spatial Concepts and Data Models

1

What is a Data Model?

•What is a model? (Dictionary meaning!)
- A set of plans (blueprint drawing)
- A miniature representation of a system to analyze properties of interest

•What is Data Model?
- Specify structure or schema of a data set
- Document description of data
- Facilitates early analysis of some properties, e.g. querying ability, redundancy, consistency, storage space requirements, etc.

• Examples:
- GIS organize spatial set as a set of layers
- Databases organize dataset as a collection of tables

•Data models facilitate
- Early analysis of properties, e.g. storage cost, querying ability, ...
- Reuse of shared data among multiple applications
- Exchange of data across organization
- Conversion of data to new software / environment

2

Types of Data Models

- Two Types of data models
  - Generic data models
    - Developed for business data processing
    - Support simple abstract data types (ADTs), e.g. numbers, strings, date
    - Not convenient for spatial ADTs, e.g. polygons
    - Need to extend with spatial concepts, e.g. ADTs
  - Application Domain specific, e.g. spatial models
    - Set of concepts developed in Geographic Info. Science
    - Common spatial ADTs across different GIS applications

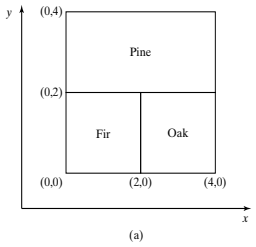
3

Models of Spatial Information

State-Park Spatial dB

- Forests – collections of forest stands
- Accessed by road
- Has Manager
- Fire-stations within State-Park
- Facilities: Camping ground, Offices
- Rivers passes through the park and supply water at different facilities

- Two common models
  - Field based
  - Object based
- Example: Forest stands
  - Fig.
    - (a) forest stand map
    - (b) Object view has 3 polygons
    - (c) Field view has a function



(a)

Object Viewpoint of Forest Stands

Area-ID	Dominant Tree Species	Area/Boundary
FS1	Pine	[(0,2),(4,2),(4,4),(0,4)]
FS2	Fir	[(0,0),(2,0),(2,2),(0,2)]
FS3	Oak	[(2,0),(4,0),(4,2),(2,2)]

(b)

Field Viewpoint of Forest Stands

$$f(x,y) = \begin{cases} \text{"Pine,"} & 2 \le x \le 4\,;\, 2 < y \le 4 \\ \text{"Fir,"} & 0 \le x \le 2;\, 0 \le y \le 2 \\ \text{"Oak,"} & 2 < x \le 4;\, 0 \le y \le 2 \end{cases}$$

(c)

4

Field based Model

- Three main concepts:
  - Spatial Framework is a partitioning of space
    - e.g., Grid imposed by Latitude and Longitude
  - Field Functions:
    f: Spatial Framework → Attribute Domain
  - Field Operations
    - Examples, addition(+) and composition(o).

$$f + g : x \rightarrow f(x) + g(x)$$
$$f \circ g : x \rightarrow f(g(x))$$

5

Types of Field Operations

- Local: value of the new field at a given location in the spatial framework depends only on the value of the input field at that location(e.g., Thresholding)
- Focal: value of the resulting field at a given location depends on the values that the input field assumes in a small neighborhood of the location(e.g., Gradient)
- Zonal: Zonal operations are naturally associated with aggregate operators or the integration function. An operation that calculates the average height of the trees for each species is a zonal operation.

- Classify following operations on elevation field
  - Identify peaks (points higher than its neighbors)
  - Identify mountain ranges (elevation over 2000 feet)
  - Determine average elevation of a set of river basins

6

Object Model

- Object model concepts
  - Objects: distinct identifiable things relevant to an application
  - Objects have attributes and operations
  - Attribute: a simple (e.g. numeric, string) property of an object
  - Operations: function maps object attributes to other objects
- Example from a roadmap
  - Objects: roads, landmarks, ...
  - Attributes of road objects:
    - spatial: location, e.g. polygon boundary of land-parcel
    - non-spatial: name (e.g. Route 66), type (e.g. interstate, residential street), number of lanes, speed limit, …
  - Operations on road objects: determine center line, determine length, determine intersection with other roads, ...

7

Classifying Spatial objects

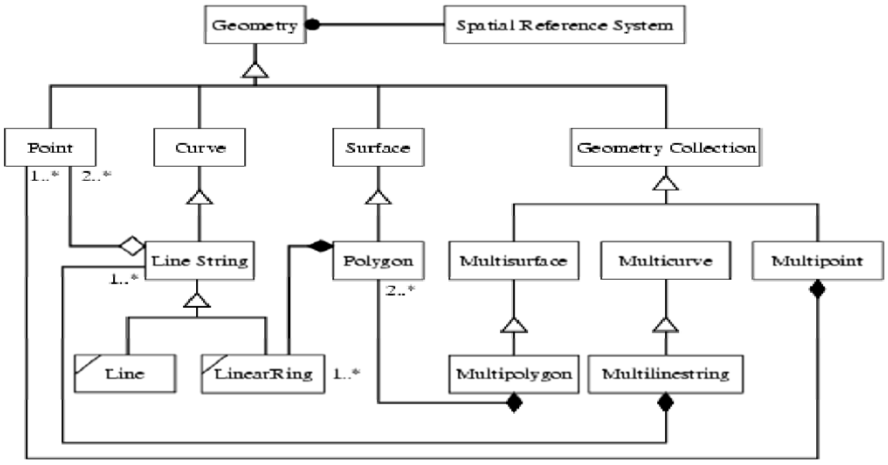
- Spatial objects are spatial attributes of general objects
- Spatial objects are of many types
  - Simple
    - 0- dimensional (points), 1 dimensional (curves), 2 dimensional (surfaces)
    - Example given at the bottom of this slide
  - Collections
    - Polygon collection (e.g. boundary of Japan or Hawaii), …

Spatial Object Types	Example Object	Dimension
Point	City	0
Curve	River	1
Surface	Country	2

8

Spatial Object Types in OGIS Data Model

Fig: Each rectangle shows a distinct spatial object type



9

Classifying Operations on spatial objects in Object Model

- Classifying operations
  - Set theory based: 2-dimensional spatial objects (e.g. polygons) are sets of points
    - a set operation (e.g. intersection) of 2 polygons produce another polygon
  - Topological operations: Boundary of USA touches boundary of Canada
  - Directional: New York city is to east of Chicago
  - Metric: Chicago is about 700 miles from New York city.

Set theory based	Union, Intersection, Containment,
Toplogical	Touches, Disjoint, Overlap, etc.
Directional	East,North-West, etc.
Metric	Distance

10

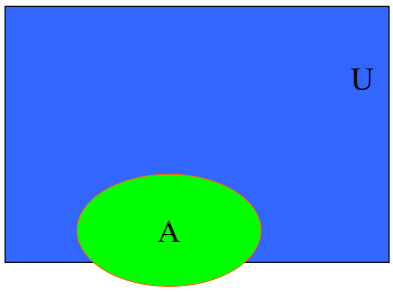
Topological Relationships

- Topological Relationships
  - Invariant under elastic deformation (without tear, merge).
  - Two countries which touch each other in a planar paper map will continue to do so in spherical globe maps.
- Topology is the study of topological relationships
- Example queries with topological operations
  - What is the topological relationship between two objects A and B ?
  - Find all objects which have a given topological relationship to object A ?

11

Topological Concepts

- Interior, boundary, exterior
  - Let A be an object in a "Universe" U.



Green is A interior (A^o)

Red is boundary of A (∂A)

Blue –(Green + Red) is A exterior (A^-)

12

Nine-Intersection Model of Topological Relationships

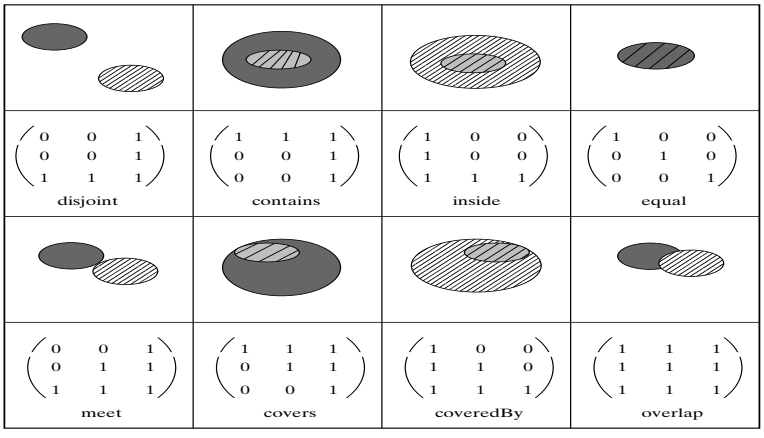
- Topological Relationship between A and B can be
  - specified using 9 intersection model
- Nine intersections
  - intersections between interior, boundary, exterior of A, B
  - A and B are spatial objects in a two dimensional plane.
  - Can be arranged as a 3 by 3 matrix
  - Matrix element take a value of 0 (false) or 1 (true).
- Q? Determine the number of many distinct 3 by 3 Boolean matrices

$$\Gamma_9(A,B) = \begin{pmatrix} A^\circ \cap B^\circ & A^\circ \cap \partial B & A^\circ \cap B^- \\ \partial A \cap B^\circ & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^\circ & A^- \cap \partial B & A^- \cap B^- \end{pmatrix}$$

13

Specifying topological operation in 9-Intersection Model

Nine (9) intersection matrices for a few topological operations



14

Using Object Model of Spatial Data

- Object model of spatial data
  - OGIS/OGC standard set of spatial data types and operations
  - Similar to the object model in computer software
  - Easily used with many computer software systems
  - Programming languages like Java, C++, Visual basic
  - Post-relational databases, e.g. OODBMS, ORDBMS

15

Three-Step Database Design

- Database applications are modeled using a three-step design process
  - Conceptual - datatypes, relationships and constraints(ER model)
  - Logical - mapping to a Relational model and associated query language(Relational Algebra)
  - Physical - file structures, indexing

16

Example Application Domain

- Database design is for a specific application domain
  - Often a requirements document is available
  - Designers discuss requirements with end-users as needed
  - We will use a simple spatial application domain
    - to illustrate concepts in conceptual and logical data models
    - to illustrate translation of conceptual DM to logical DM
- Spatial application domain
  - A *state-park* consists of *forests.*
  - A forest is a collection of *forest-stands* of different species
  - State-Park is accessed by *roads* and has a *manager*
  - State-Park has *faciltities*
  - *River* runs through state-park and supplies water to the facilities

17

Conceptual DM: The ER Model

- Three basic concepts
  - Entities have an independent conceptual or physical existence.
    - Examples: Forest, Road, Manager, ...
  - Entities are characterized by Attributes
    - Example: Forest has attributes of name, elevation, etc.
  - An Entity interacts with another Entity through relationships.
    - Road allow access to Forest interiors.
    - This relationship may be name "Accesses"
- Comparison with Object model of spatial information
  - Entities are collections of attributes are like objects
  - However ER model does not permit general user defined operations
  - Relationships are not directly supported in Object model
    - but may be simulated via operations

18

Relationship Types

- Relationships can be categorized by
  - cardinality constraints
  - other properties, e.g. number of participating entities
    - Binary relationship: two entities participate
- Types of Cardinality constraints for binary relationships
  - One-One: An instance of an entity relates to a unique instance of other entity.
  - Many-One: Many instances of an entity relate to an instance of an other.
  - Many-Many: Many instances of one entity relate to multiple instances of another.
- Identify type of cardinality constraint for following:
  - Many facilities belong to a forest. Each facility belong to one forest.
  - A manager manages 1 forest. Each forest has 1 manager.
  - A river supplies water to many facilities. A facility gets water from many rivers.

19

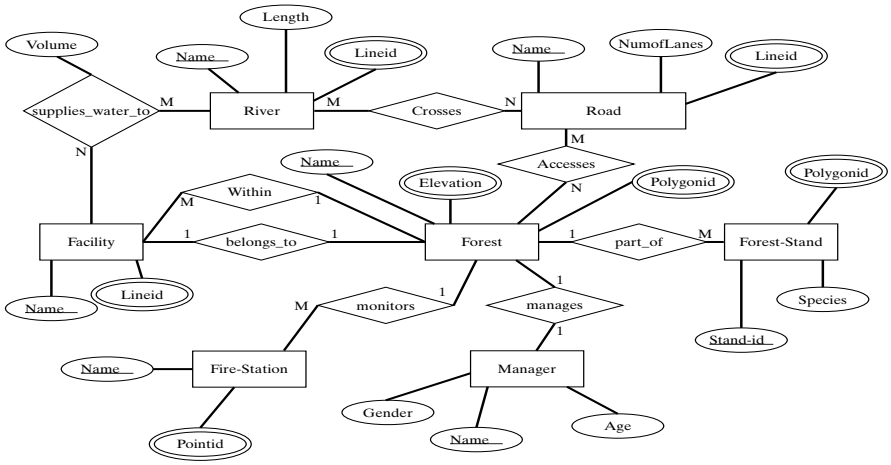
ER Diagrams Graphical Notation

- ER Diagrams are graphic representation of ER models
  - Several different graphic notation are used
- Q? Compare and contrast "Attributes" and "Multi-valued attributes"

Concept	***Symbol***
Entities	
Attributes	
Multi-valued Attributes	
Relationships	
Cardinality of Relationship	1:1, M:1, M:N

20

ER Diagram for "State-Park"



- Examples:
  - List the entities, attributes, relationships in this ER diagram
  - Identify cardinality constraint for each relationship.
  - How many roads "Accesses" a "Forest_stand"? (one or many)

21

Logical Data Model: The Relational Model

- Relational model is based on set theory
- Main concepts
  - Domain: a set of values for a simple attribute
  - Relation: cross-product of a set of domains
    - Represents a table, i.e. homogeneous collection of rows (tuples)
    - Set of columns (i.e. attributes) are same for each row
- Comparison to concepts in conceptual data model
  - Relations are similar to but not identical to entities
  - Domains are similar to attributes
  - Translation rules establishing exact correspondence

22

Relational Schema

- Schema of a Relation
  - Enumerates columns, identifies primary key and foreign keys.
  - Primary Key :
    - one or more attributes uniquely identify each row within a table
  - Foreign keys
    - R's attributes which form primary key of another relation S
    - Value of a foreign key in any tuple of R match values in some row of S
- Relational schema of a database
  - Collection of schemas of all relations in the database
  - A Blue print summary drawing of the database table structures
  - Allows analysis of storage costs, data redundancy, querying capabilities
  - Some databases were designed as relational schema in 1980s

23

Relational Schema Example

- Identify relations with
  - primary keys
  - foreign keys
  - other attributes
- Compare with ER diagram

Forest-Stand

Stand-id	Species	Forest-name
(Integer)	(varchar)	(varchar)

Fstand-Geom

Stand-id	Polygonid
(Integer)	(Integer)

River

Name	Length
(varchar)	(Real)

River-Geom

Name	Lineid
(Integer)	(Integer)

Road

Name	NumofLanes
(varchar)	(Integer)

Road-Geom

Rname	Lineid
(varchar)	(Integer)

Facility

Name	Forest-name	Forest-name-2
(varchar)	(varchar)	(varchar)

Facility-Geom

Name	Pointid
(varchar)	(Integer)

Forest

Name
(varchar)

Forest-Geom

Name	Polygonid
(varchar)	(Integer)

Fire-Station

Name	ForName
(varchar)	(varchar)

Fstation-Geom

Name	Pointid
(varchar)	(Integer)

Supplies_Water_To

FacName	RivName	Volume
(varchar)	(varchar)	(Real)

Road-Access-Forest

RoadName	ForName
(varchar)	(varchar)

Manager

Name	Age	Gender	ForName
(varchar)	(Integer)	(varchar)	(varchar)

24

Relational Schema for "Point", "Line", "Polygon" and "Elevation"

- Relational model restricts attribute domains
  - Simple atomic values, e.g. a number
  - Disallows complex values (e.g. polygons) for columns
  - Complex values need to be decomposed into simpler domains
  - A polygon may be decomposed into edges and vertices

Polygon

Polygonid (Integer)	Seq-no (Integer)	Pointid (Integer)

Line

Lineid (Integer)	Seq-no (Integer)	Pointid (Integer)

Point

Pointid (Integer)	Latitude (Real)	Longitude (Real)

Elevation

Forest-name (varchar)	Pointid (F.K.) (Integer)	Elevation (Real)

25

More on Relational Model

- Integrity Constraints
  - Key: Every relation has a primary key.
  - Entity Integrity: Value of primary key in a row is never undefined
  - Referential Integrity: Value of an attribute of a Foreign Key must appear as a value in the primary key of another relationship or must be null.
- Normal Forms (NF) for Relational schema
  - Reduce data redundancy and facilitate querying
  - 1st NF: Each column in a relation contains an atomic value.
  - 2nd and 3rd NF: Values of non-key attributes are fully determined by the values of the primary key, only the primary key, and nothing but the primary key.
  - Other normal forms exists but are seldom used
  - Translating a well-designed ER model yields a relational schema in 3rd NF
    - satisfying definition of 1st, 2nd and 3rd normal forms

26

Mapping ER to Relational

- Highlights of transactional rules
  - Entity becomes Relation
  - Attributes become columns in the relation
  - Multi-valued attributes become a new relation
    - includes foreign key to link to relation for the entity
  - Relationships (1:1, 1:N) become foreign keys
  - M:N Relationships become a relation
    - containing foreign keys or relations from participating entities

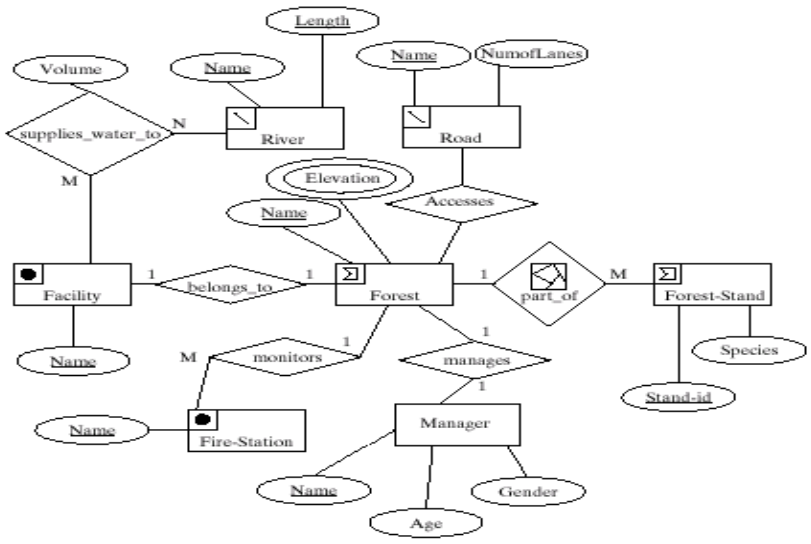
27

Extending ER with Spatial Concepts

- Motivation
  - ER Model is based on discrete sets with no implicit relationships
  - Spatial data comes from a continuous set with implicit relationships
  - Any pair of spatial entities has relationships like distance, direction, …
- Explicitly drawing all spatial relationship
  - Clutters ER diagram
  - Generates additional tables in relational schema
  - Misses implicit constraints in spatial relationships (e.g. partition)
- Pictograms
  - Label spatial entities along with their spatial data types
  - Allows inference of spatial relationships and constraints
  - Reduces clutter in ER diagram and relational schema

28

ER Diagram with Pictograms: An Example



29

Specifying Pictograms

- Grammar based approach
  - Rewrite rule
  - like English syntax diagrams
- Classes of pictograms
  - Entity pictograms
    - basic: point, line, polygon
    - collection of basic
    - ...
  - Relationship pictograms
    - partition, network

⟨ Pictogram ⟩ → ⟨ Shape ⟩
→ *
→ !

Grammar (for Pictogram)


Part_of(Network) Part_of(Partition)

Pictograms for Relationships

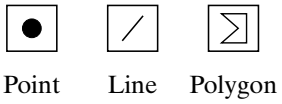
⟨ Shape ⟩ → ⟨ Basic Shape ⟩
→ ⟨ Multi-Shape ⟩
→ ⟨ Derived Shape ⟩
→ ⟨ Alternate Shape ⟩

Grammar (for Shape)

30

Entity Pictograms: Basic shapes, Collections

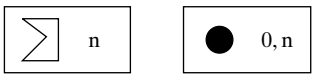
⟨ Basic Shape ⟩ → ●
→ ⁄
→ ∑



Grammar (for Basic Shape)

Pictograms for Basic Shapes

⟨ Cardinality ⟩ → 0, 1
→ 1
→ 1, n
→ 0, n
→ n



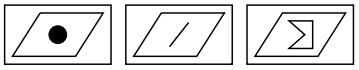
Grammar (for Cardinality)

Pictograms Multishapes (using cardinality)

Entity Pictograms: Derived and Alternate Shapes

- Derived shape example is city center point from boundary polygon
- Alternate shape example: A road is represented as a polygon for construction, or, as a line for navigation

⟨ Derived Shape ⟩ → ⟨ Basic Shape ⟩

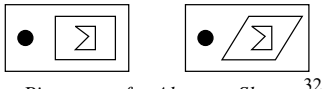


Grammar (for Derived Shape)

Pictograms for Derived Shapes

⟨ Alternate Shape ⟩ → ⟨ Basic Shape ⟩ ⟨ Derived Shape ⟩
→ ⟨ Basic Shape ⟩ ⟨ Basic Shape ⟩

Grammar (for Alternate Shape)



Pictograms for Alternate Shapes

Conceptual Data Modeling with UML

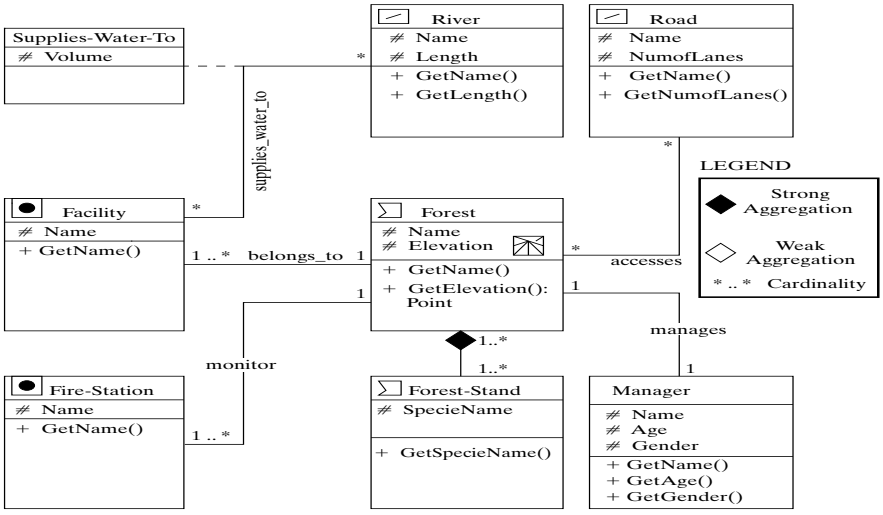
- Motivation
  - ER Model does not allow user defined operations
  - Object oriented software development uses UML
  - UML stands for Unified Modeling Language
  - It is a standard consisting of several diagrams
    - class diagrams are most relevant for data modeling
- UML class diagrams concepts
  - Attributes are simple or composite properties
  - Methods represent operations, functions and procedures
  - Class is a collection of attributes and methods
  - Relationship relate classes

33

UML Class Diagram with Pictograms: Example

•Exercise: Identify classes, attributes, methods, relationships in the Figure.

•Compare it with corresponding ER diagram



34

Comparing UML Class Diagrams to ER Diagrams

- Concepts in UML class diagram vs. those in ER diagrams
  - Class without methods is an Entity
  - Attributes are common in both models
  - UML does not have key attributes and integrity constraints
  - ERD does not have methods
  - Relationships properties are richer in ERDs
  - Entities in ER diagram relate to datasets, but UML class diagram can contain classes which have little to do with data

35

Summary

- Spatial Information modeling can be classed into Field based and Object based
- Field based for modeling smoothly varying entities, like rainfall
- Object based for modeling discrete entities, like country

36

Summary

- A data model is a high level description of the data
  - it can help in early analysis of storage cost, data quality
- There are two popular models of spatial information
  - Field based and Object based
- Database are designed in 3-steps
  - Conceptual, Logical and Physical
- Pictograms can simplify Conceptual data models

37