+

CSIT6000P Spatial and Multimedia Databases

2022 Spring

An Introduction to Spatial Databases

Prof Xiaofang Zhou

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

+ Learning Objectives

- What we will cover
  - Spatial data types and modelling
  - Spatial relationships, operations and queries
  - SDBMS architectures
- Goals
  - Understand how spatial data is different from the relational data
  - Understand how these differences affect those relational techniques we learned before
  - Understand what spatial DBMS is

+ Readings

- R. Güting, An Introduction to Spatial Database Systems, *The VLDB Journal*, 3:4, 1994
- Hanan Samet and Walid G. Aref, Spatial Data Models and Query Processing, in W. Kim (Ed), *Modern Database Systems: The Object Model, Interoperability, and Beyond*, 1995

+ Why Spatial DBMS?

- Huge amounts of spatial data, extensive and comprehensive
- Increasing needs to store, search and use spatial data, together with other data, efficiently, enterprise-wide
- Alternatives?
  - File system-based solutions?
  - Application-based solutions?

… why DBMS?

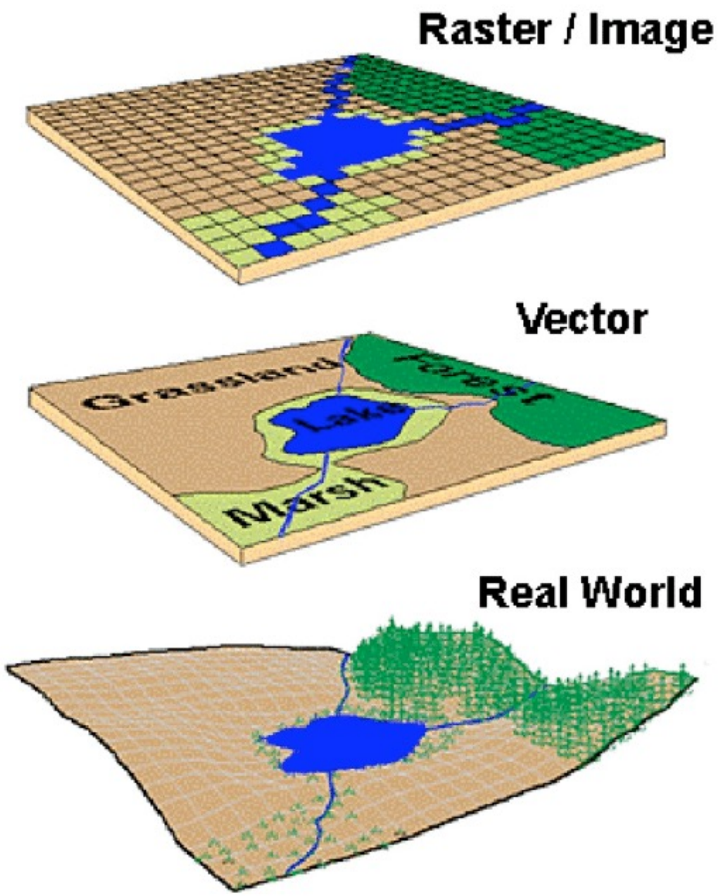
+ Spatial Data

- A location is a place or position in a space
- Spatial data is any data with a location component
  - 2D space
    - Geographical space: GIS, urban planning
    - Graphics: CAD, VLSI design
  - 3D space
    - The universe, brain model, molecule structure
- Two types of spatial data
  - Those data about the space (e.g., road networks, maps)
  - Those data about objects in a space (e.g., location of shops, location of cars)

+ More Spatial Data

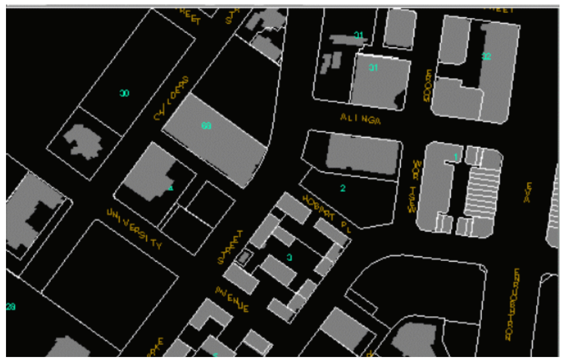
- Natural area data
  - Soil types, land use (industrial, agriculture, residential etc), vegetation, water (rivers, ponds etc)
- Manmade area data
  - Political and administrative boundaries, school districts, emergency service areas, land records data (lot boundaries, zoning, easements)
- Network data
  - Utilities (sewers, water pipes, powerlines etc)
  - Roads (centrelines, curb lines, intersections etc)

+ Modelling Spatial Data



+ Vector Data

- A spatial object is described by a sequence of points
  - A point can be specified using latitude and longitude coordinate values
- Advantages
  - Suitable for processing & manipulation
  - More compact, better rendering quality
  - Query by spatial relationships (e.g., find all shops within 300 meters)

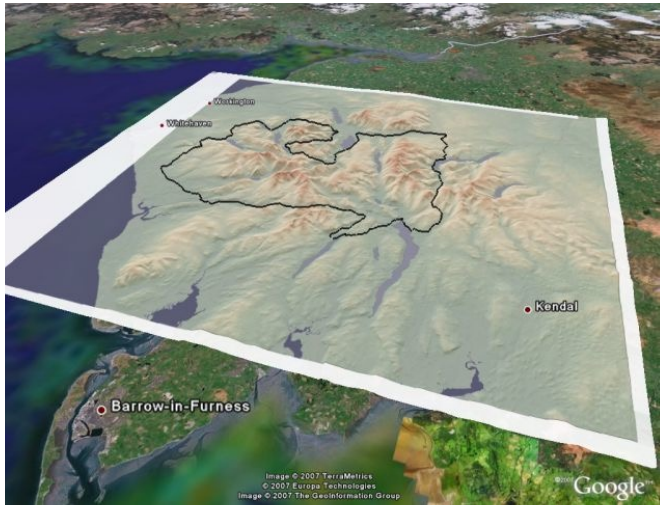


+ Raster Data

- A spatial object is described by a set of pixels
  - E.g., satellite imagery
- Advantages
  - Suitable for display
  - Query by colour, texture, etc.
  - Efficient for some type of processing, such as monitoring desertization using remote sensing images over time

+ Vector vs Raster Data

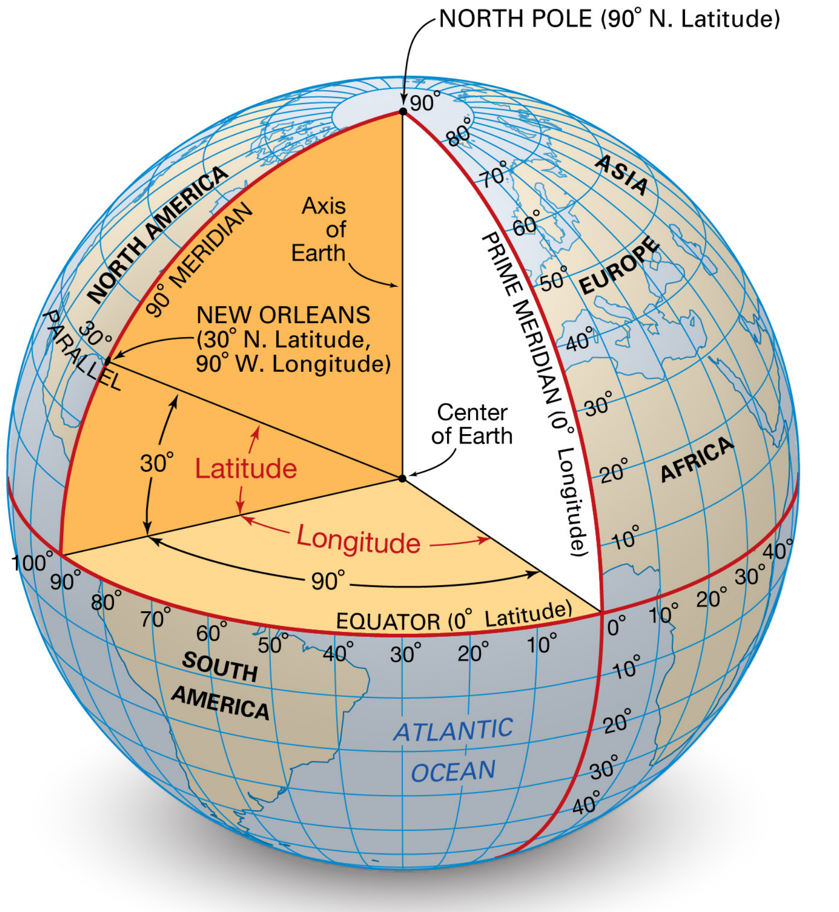
- Different representation of same data
- Suitability application-dependent
- Mutually convertible
  - Rasterization
  - Vectorization



- Often used together
  - E.g., overlay satellite image, road network, elevation data etc.
  - Hybrid mode for Google Earth/Google Map
- We deal with vector data in spatial databases

+ Geographical Coordinate Systems

- Location reference systems for spatial features on the earth surface
  - Different models can be selected to approximate the shape and size of the Earth
  - Points are described using longitude and latitude angles measured in degrees (called geographical coordinates)



+ Map Projection

- In cartography, a map projection is a way to flatten a globe's surface into a plane in order to make a map
  - Systematic transformation of the latitudes and longitudes of locations from the surface of the globe into locations on a plane

- Commonly used map projections
  - UTM (Universal Transverse Mercator)
    - Developed by Gerhardus Mercator, a Flemish Cartographer, in 1569
    - Badly distorts Greenland, Alaska, and Australia.
  - AMG (Australian Map Grid)
- Need spatial data (and metadata) standards to record the mapping parameters used
  - Spatial data must be aligned before they can be used together

+ Spatial Data Acquisition

- There are many publically available spatial datasets
  - Government sources, and other map providers such as Google
- Generating new data
  - Remote sensing data
  - Field data collection using GPS devices
  - Crowd-source data, such as OpenStreetMap

OpenStreetMap

openstreetmap.org

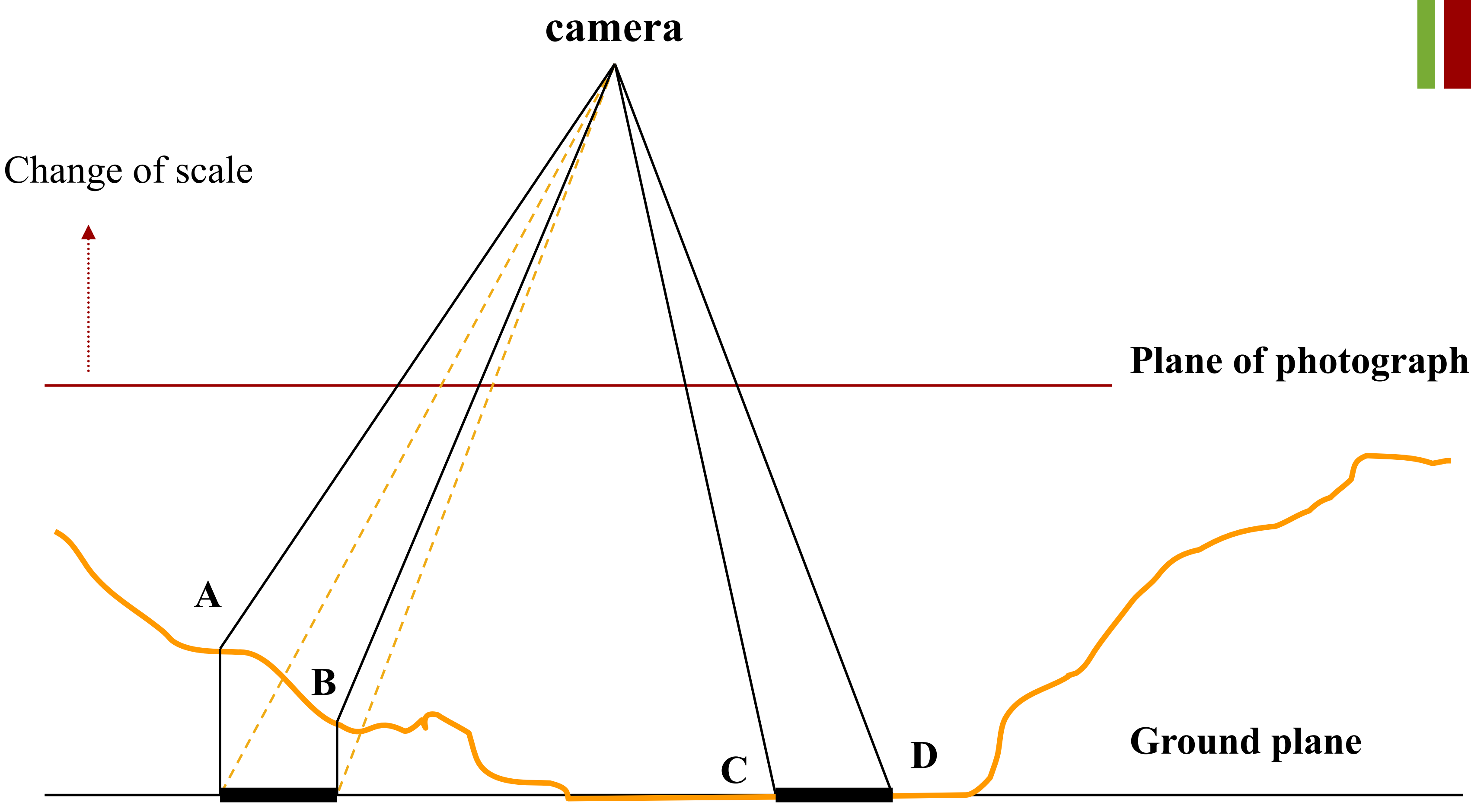
OpenStreetMap is a collaborative project to create a free editable geographic database of the world. The geodata underlying the maps is considered the primary output of the project. Wikipedia

Owner: Community-owned; supported by OpenStreetMap Foundation

Created by: Steve Coast (Page in OSM wiki)

Users: 8,104,497

+ Remote Sensing



+ Spatial Data Accuracy

- Accuracy: how close the recorded location of a spatial feature is to its ground location?
- Scale: the ratio between distance on a map and the corresponding distance on the earth (e.g., 1:100,000)
- Resolution: the size of the smallest feature that can be represented in a surface
- Precision: how exactly the location is recorded (i.e., # of digits)

… errors can be introduced from many sources, such as data capturing devices and environments, data processing operations

+ Spatial DBMS

- A spatial DBMS is a DBMS
- It offers spatial data types in its data model and the query language
- It supports spatial data types in its implementation, providing at least spatial indexing and efficient spatial query processing algorithms

+ Alternative Names

- AM/FM System
  - Automated mapping and facilities management
- GIS (Geographical Information System)
- Land Information Systems
- Natural Resources Information Systems
- Spatial Data Management (or Handling) System
- Object-Relational Database Systems
  - Oracle has many types of data cartridges, including one for spatial data

+ GIS and Spatial Databases

- GIS applications
  - Data capture, editing, conversion, conflation
  - Map generation
  - Image processing
  - Data analysis (in application areas)
- Spatial DBMS
  - Integrated management of spatial and non-spatial data
  - Database support expected in a RDBMS, independent from applications
  - RDBMS-comparable performance

+ SDBMS is a Multidiscipline Area

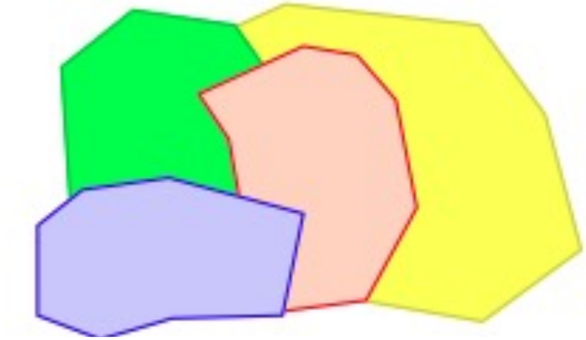
- Cartography
  - Display of visual information (or, you can call it *visualization*)
- Computer Science
  - Databases, computer graphics, image processing, machine learning
- Geography
  - Spatial analysis
- Mathematics
  - Geometry, graph theory
- Statistics
  - Models, analysis of error
- Photogrammetry, remote sensing, surveying…

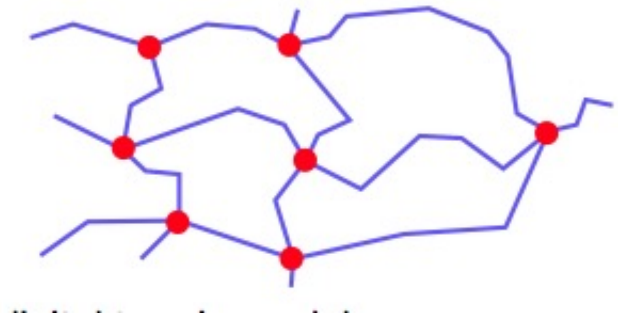
+ History

- Canada: Geographic Information System (the 60’s)
- Harvard: ODYSSEY project (the 70’s)
  - ESRI: Arc/Info
- American Bureau of Census
  - 1970 census (only urban areas)
  - TIGER (entire country, from 1990)
- Australia
  - From CSIRO: SIRO-DBMS (SDM) (in the late 80’s)
  - Many small GIS companies
- Main-stream DB vendors (late 1990’s)
  - Oracle, Informix, IBM (DB2), Microsoft (MapPoint)
- And now, Google and many Internet companies…

+ Spatial Data Models

- Objects in space
  - Single objects
    - Point (city)
    - Line/polyline (river, cable, road)
    - Polygon or Region (forest, lake, city)
  - Spatially related collections of objects
    - Partition (land use, districts, land ownership)
    - Network (roads, rivers, electricity, phone)



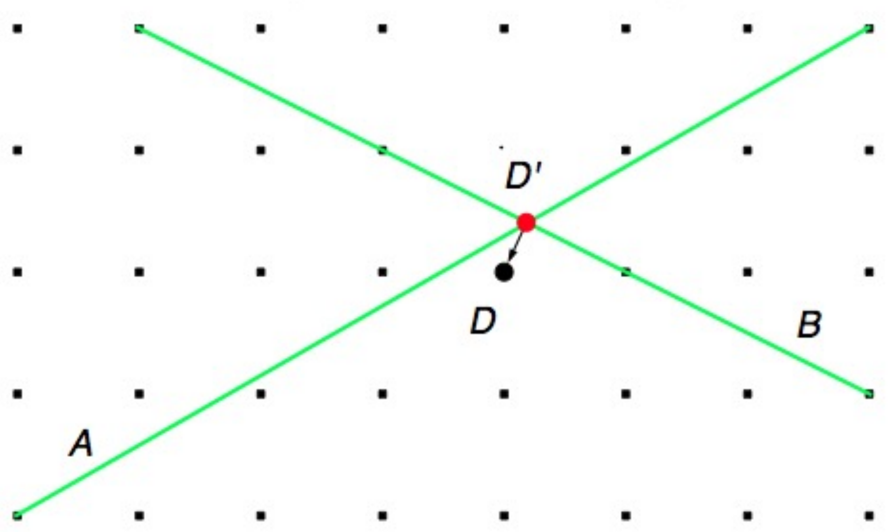


- Space itself
  - The spatial extent, Euclidean or other types of spaces

+ The Underlying Space

- Euclidean space is continuous, computer numbers are finite and discrete

- Problems: numerical rounding errors and topological inconsistency and degeneracy



Is D on A?

Is D contained in the area below A & B?

A practical solution: for two points **a** and **b**, never ask if **a**=**b**; instead, test if *distance*(**a**, **b**) < ε

+ Representing Spatial Data

- How can we use a relational database system to store spatial data?

```
Point(pID: INT, x: number, y: number)

Shop(sID: INT, loc: INT …) // loc is an FK to point ID
River(rID: INT, pointID: INT, order: INT …)
Lanuse(landID: INT, pointID: INT, order: INT …)

Landuse(landID: INT, area: BLOB …)
```

… do you see any problems for such representations?

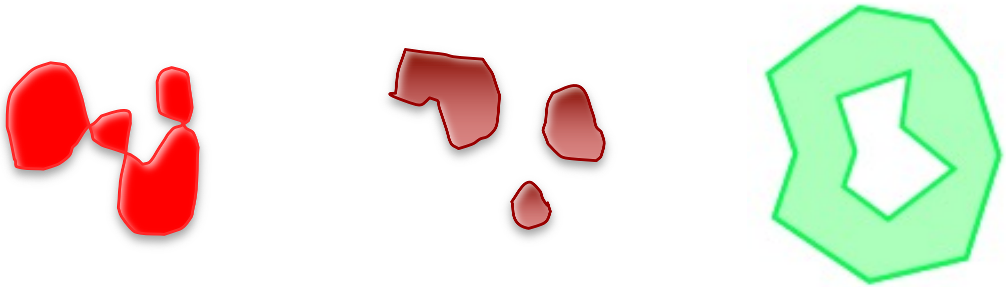
+ A Better Way

- Spatial data model
  - `Shop(sID: INT, loc: POINT …)`
  - `River(rID: INT, route: POLYLINE …)`
  - `landuse(landID: INT, area: POLYGON …)`

... what do you expect from these data types?

+ Data Model in Oracle (I)

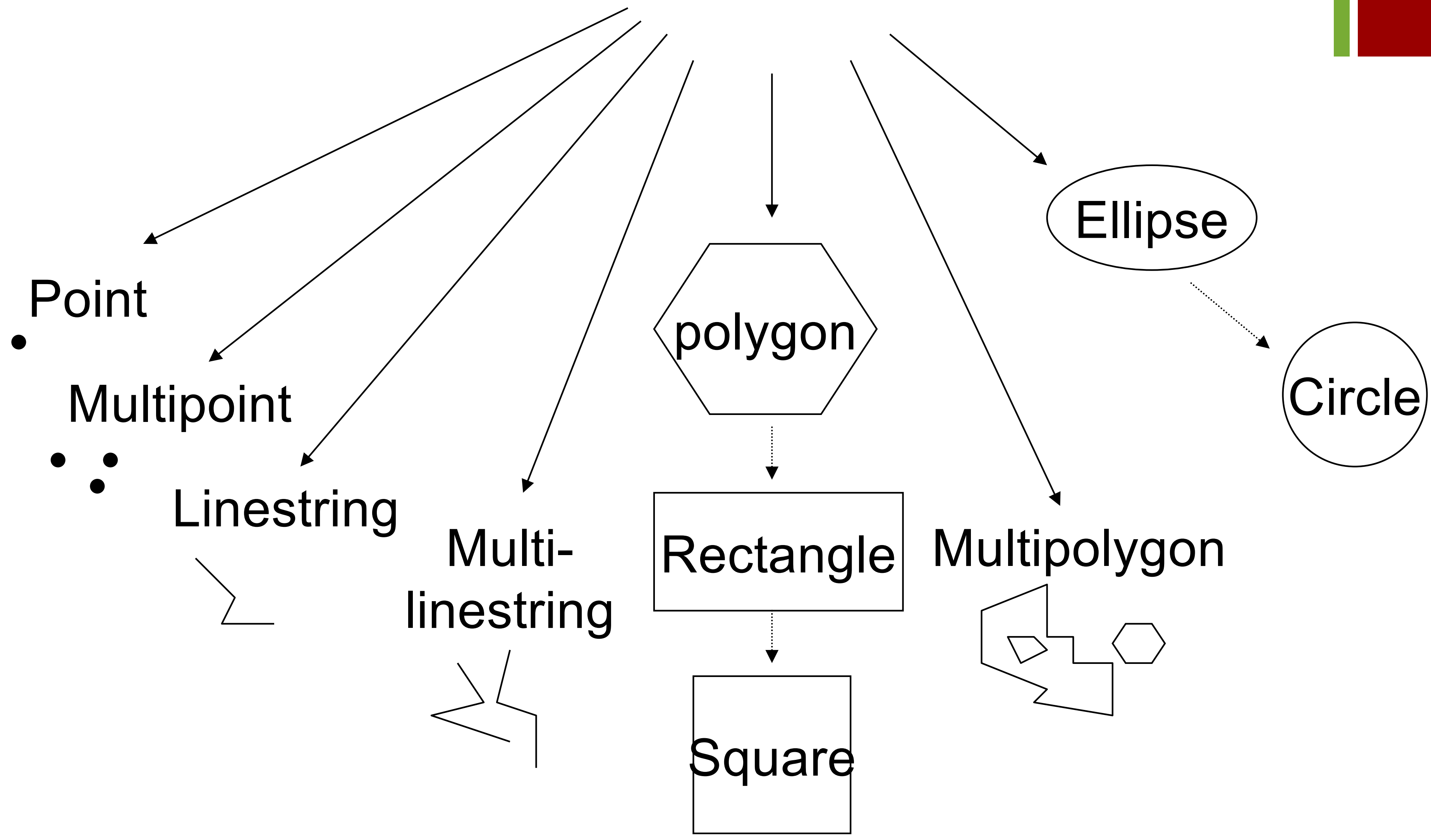
- Element, Geometry and Layer
- Element: the basic building block of a geometric feature
  - Point data: one coordinate, stored as an (x, y) pair
  - Line data: two coordinates representing the start and the end of a line segment
  - Polygon data: a sequence of coordinates, one vertex pair for each line segment of the polygon
    - Both boundary and the interior
    - Complex polygons: self-intersecting boundary, multiple connected components



+ Data Model in Oracle (II)

- Geometry: representation of a user's spatial feature, modelled as an ordered set of elements
  - Each geometry has a unique id, and can be associated with a set of attributes
  - Example: a geometry might describe a lake, represented as a polygon with nested polygons for islands, with attributes such as lake name, water capacity etc
- Layer: a collection of geometries having the same attribute set
  - Examples: soil types, road network, political boundaries, population density, crops

+ Data Model in DB2

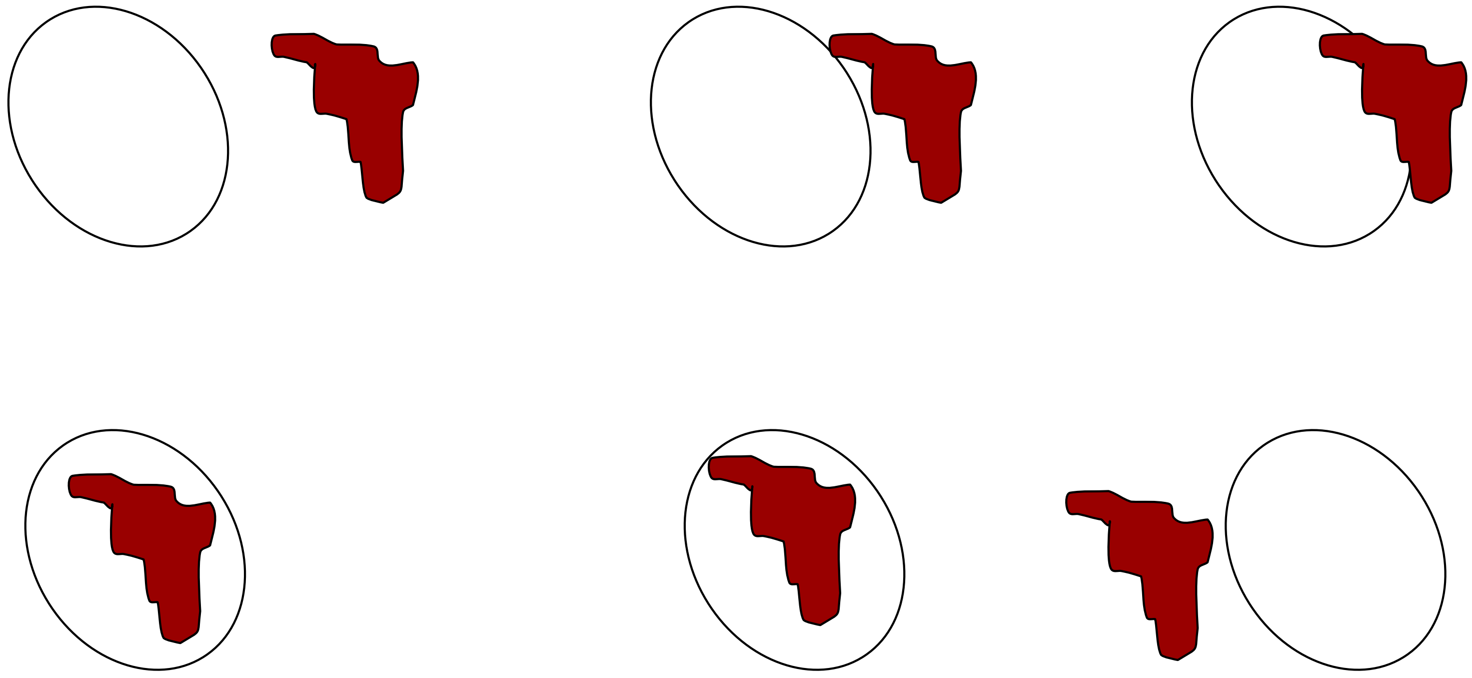


+ Spatial Relationships

- Topological relationship
  - E.g., *inside*, *intersect*, *adjacent*
  - Invariant under translation such as rotation and scaling
- Directional relationship
  - E.g., *above*, *left*
  - May change with rotation
- Metric relationship
  - E.g., *distance*
  - May change with scaling

+ Defining Spatial Relationship

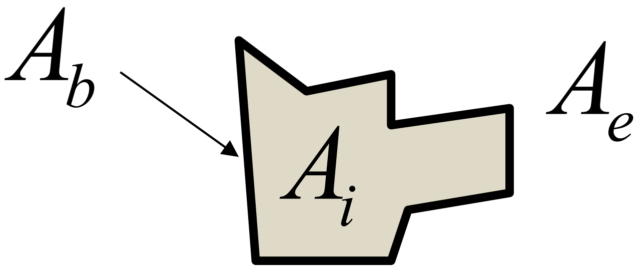
- Q: how to define a spatial relationship precisely?



+ Formal Definitions

- Better understanding of the complex semantics of spatial objects and operations at the designer's level
- Clarity and consistency at the user’s level
- A step towards standardization

+ The 9-Intersection Matrix



A_b: boundary

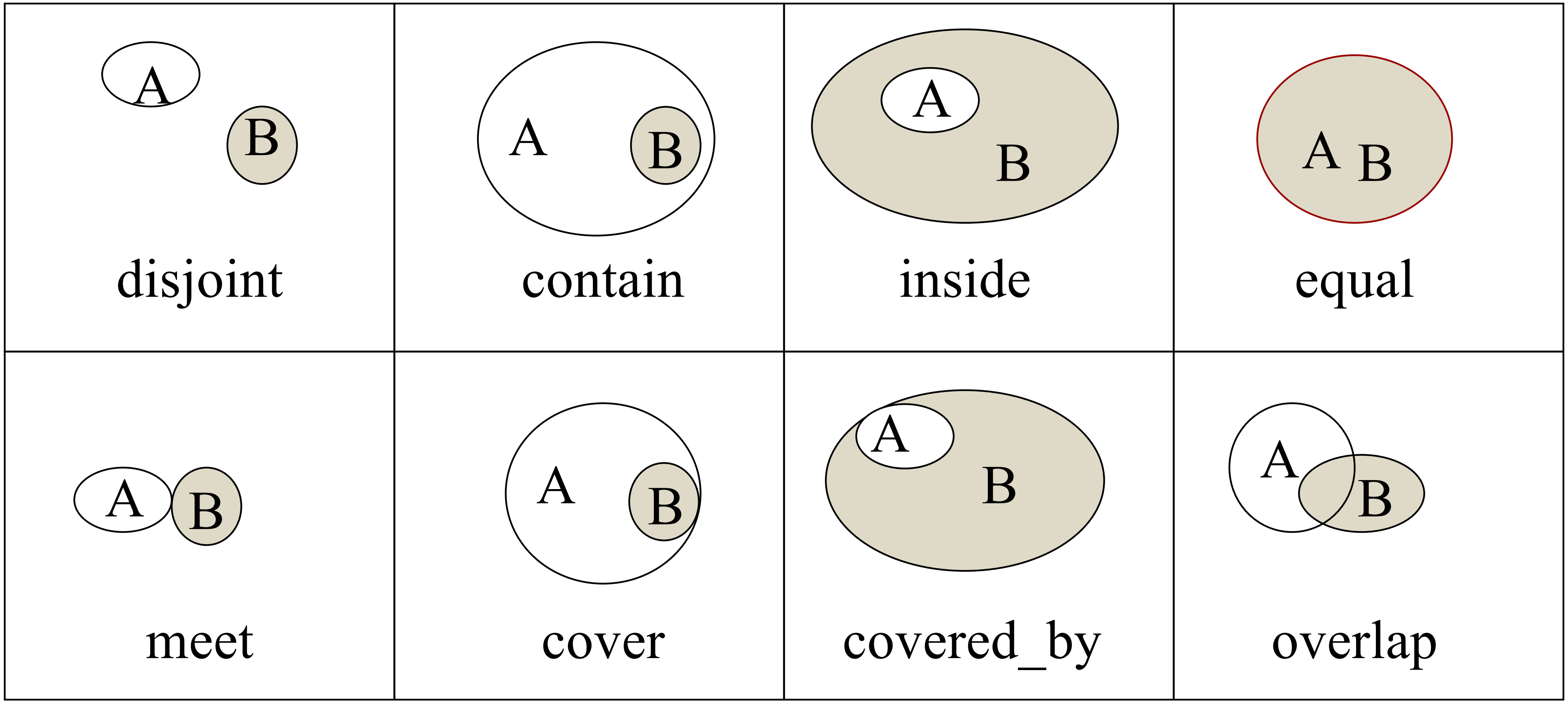
A_i: interior

A_e: exterior

$$\begin{pmatrix} A_b \cap B_b & A_b \cap B_i & A_b \cap B_e \\ A_i \cap B_b & A_i \cap B_i & A_i \cap B_e \\ A_e \cap B_b & A_e \cap B_i & A_e \cap B_e \end{pmatrix}$$

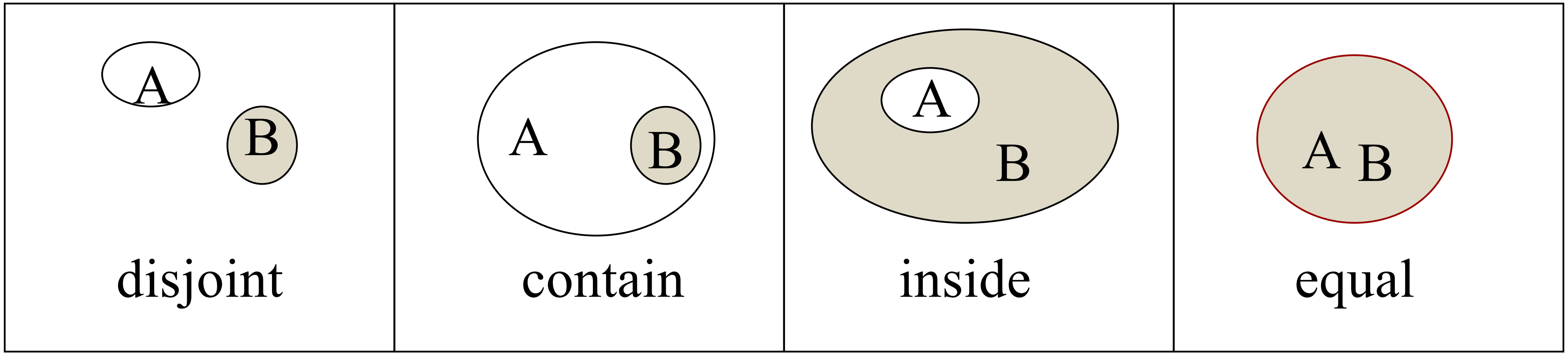
The 8 Spatial Relationships (I)

- Complete, and mutually exclusive.



The 8 Spatial Relationships (I)

$$\begin{pmatrix} A_b \cap B_b & A_b \cap B_i & A_b \cap B_e \\ A_i \cap B_b & A_i \cap B_i & A_i \cap B_e \\ A_e \cap B_b & A_e \cap B_i & A_e \cap B_e \end{pmatrix}$$



$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

disjoint contain inside equal

The 8 Spatial Relationships (I)

$$\begin{pmatrix} A_b \cap B_b & A_b \cap B_i & A_b \cap B_e \\ A_i \cap B_b & A_i \cap B_i & A_i \cap B_e \\ A_e \cap B_b & A_e \cap B_i & A_e \cap B_e \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

meet

$$\begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

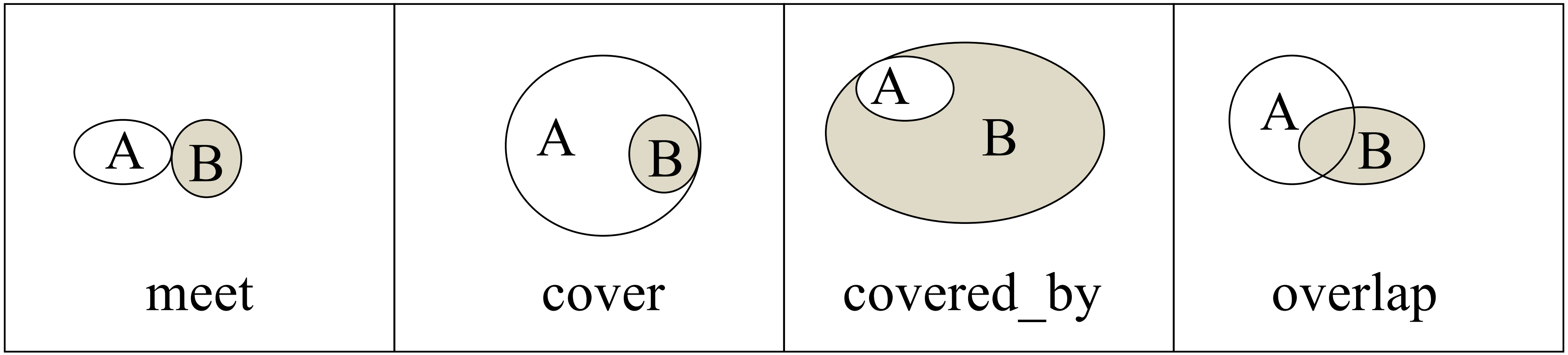
cover

$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

covered_by

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

overlap



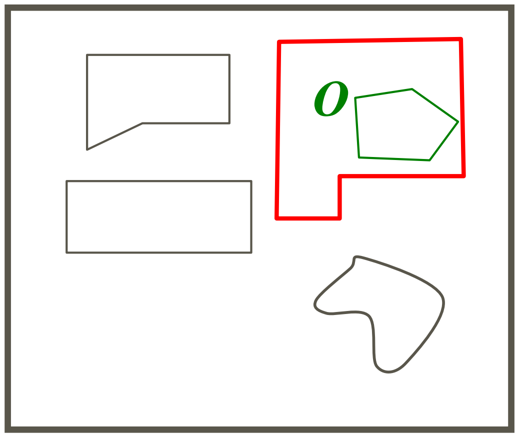
+ Spatial Operations

- Spatial relationships
  - Topological and directional predicates
  - Advanced ones: e.g., nearest neighbors, within-distance
- Spatial functions
  - Length (of lines), center, perimeter and area (of polygons)
  - Distance between two spatial objects (e.g., point-to-point, point-to-line, polygon-to-polygon)
- Spatial operations
  - Intersection, union, overlay, clipping
- Spatial queries
  - Similar to SQL, such as selection, projection, join, complex predicates, group-by and sub-queries

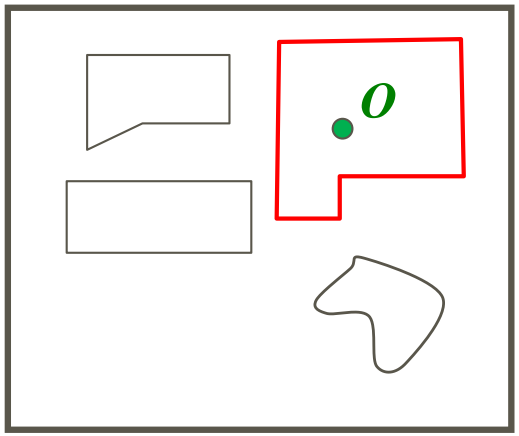
... spatial data types should come with these operation support

+ Containment Query

- Given a spatial object O, find all objects in the database that completely contain O
- When O is a point, the query is called Point Query



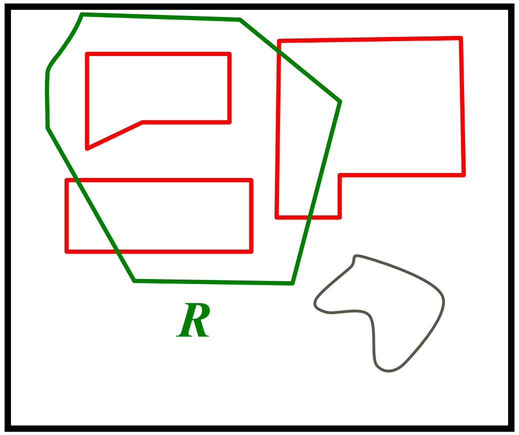
O is a polygon (O is the green one) and the red one is the query result.

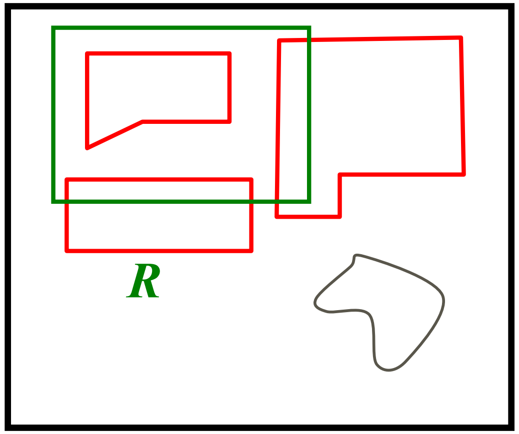


O is a point (the green dot).

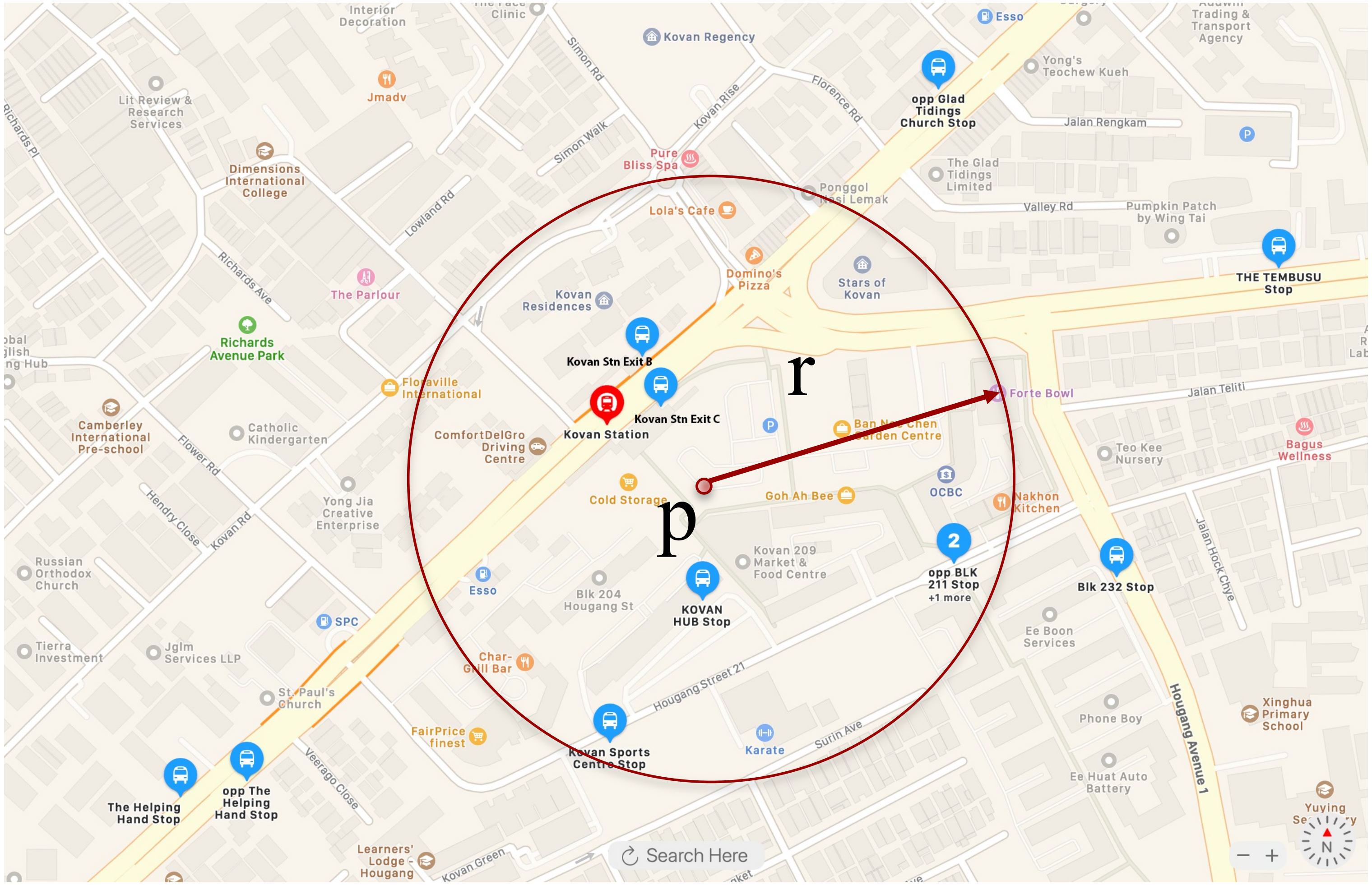
+ Region Query

- Given a region R, find all objects in the map that intersect R
- When R is a rectangle, the query is called Window Query

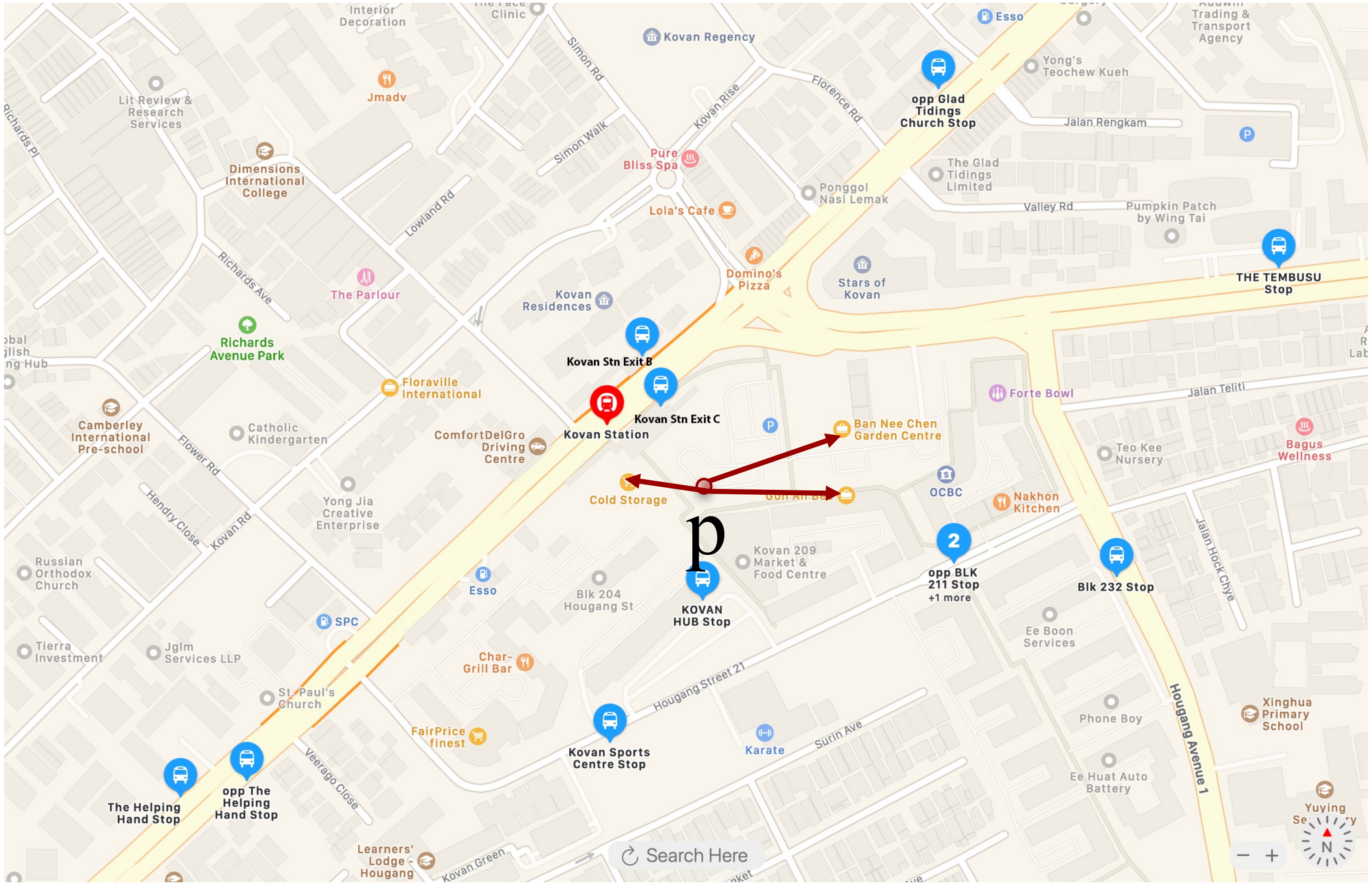




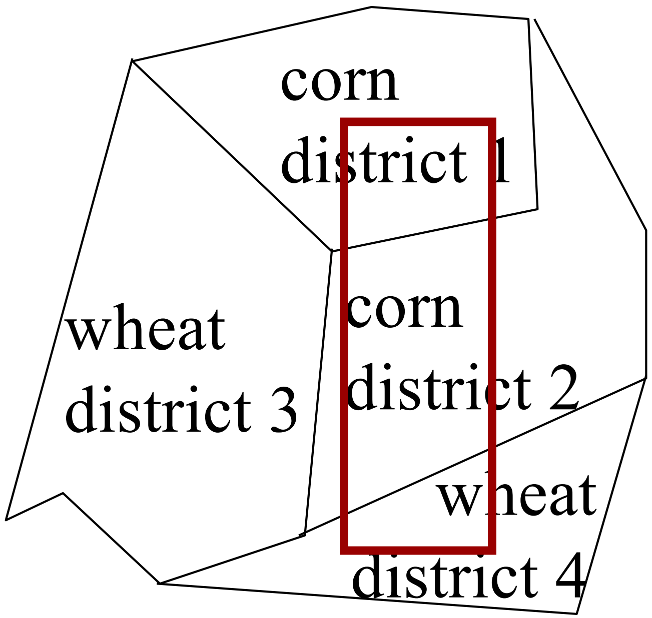
+ Within-Distance Query

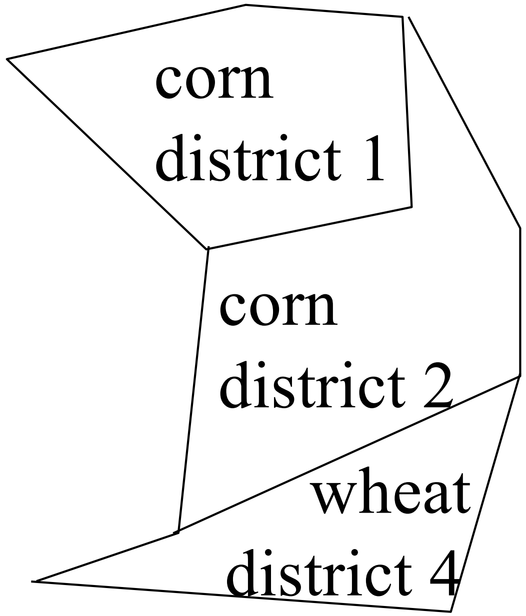


+ K Nearest Neighbor (kNN) Query

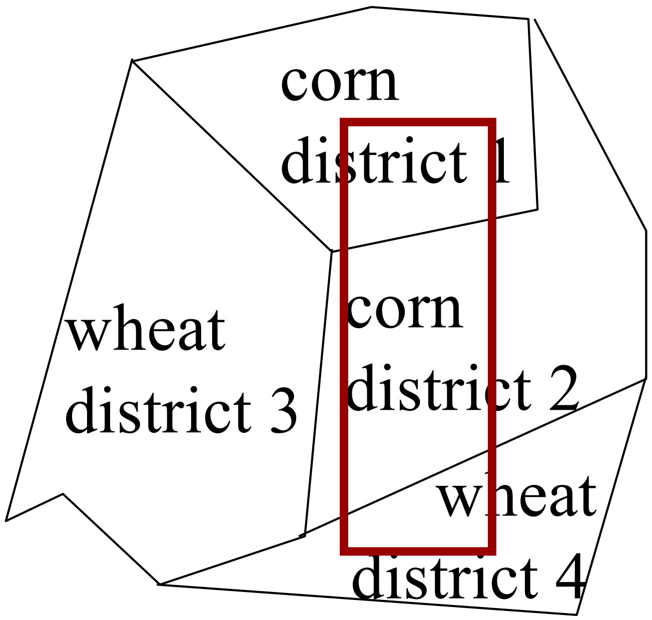


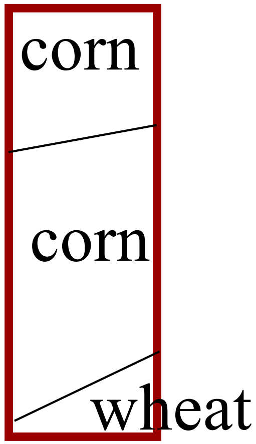
+ Overlapping Query



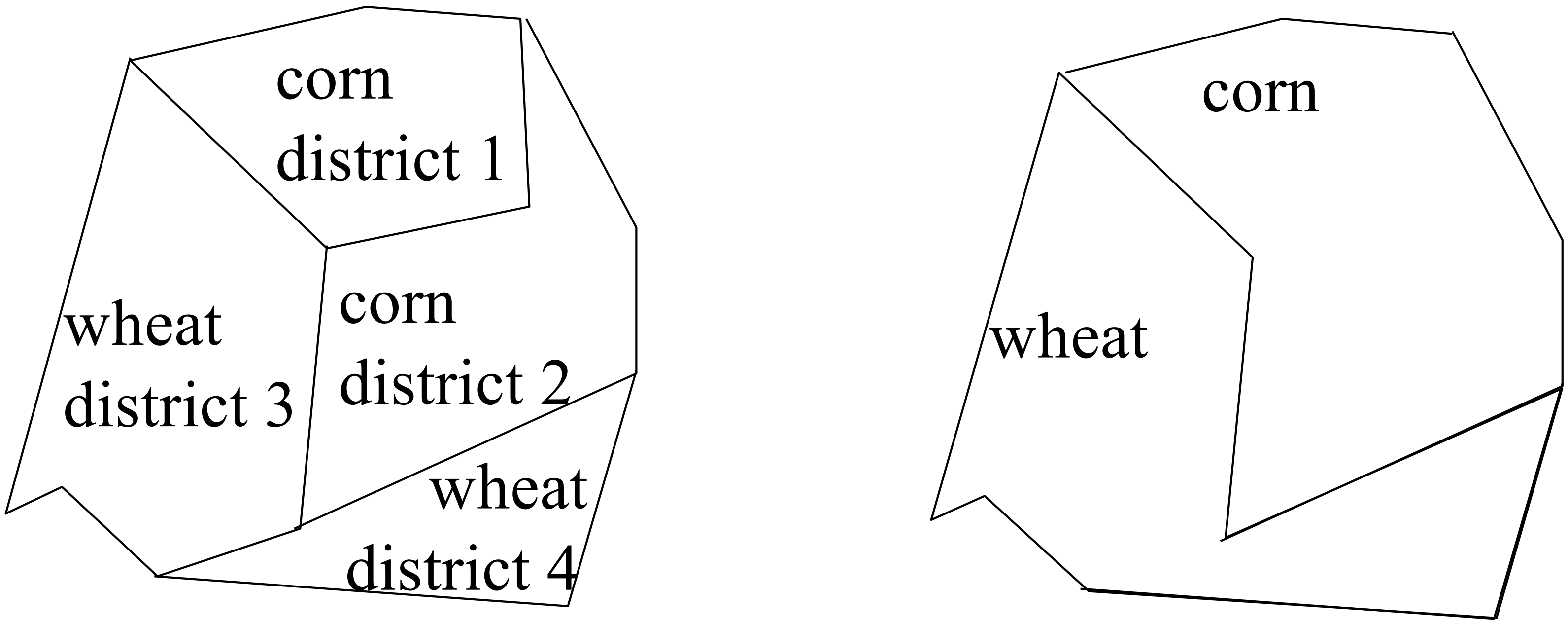


+ Clipping Query



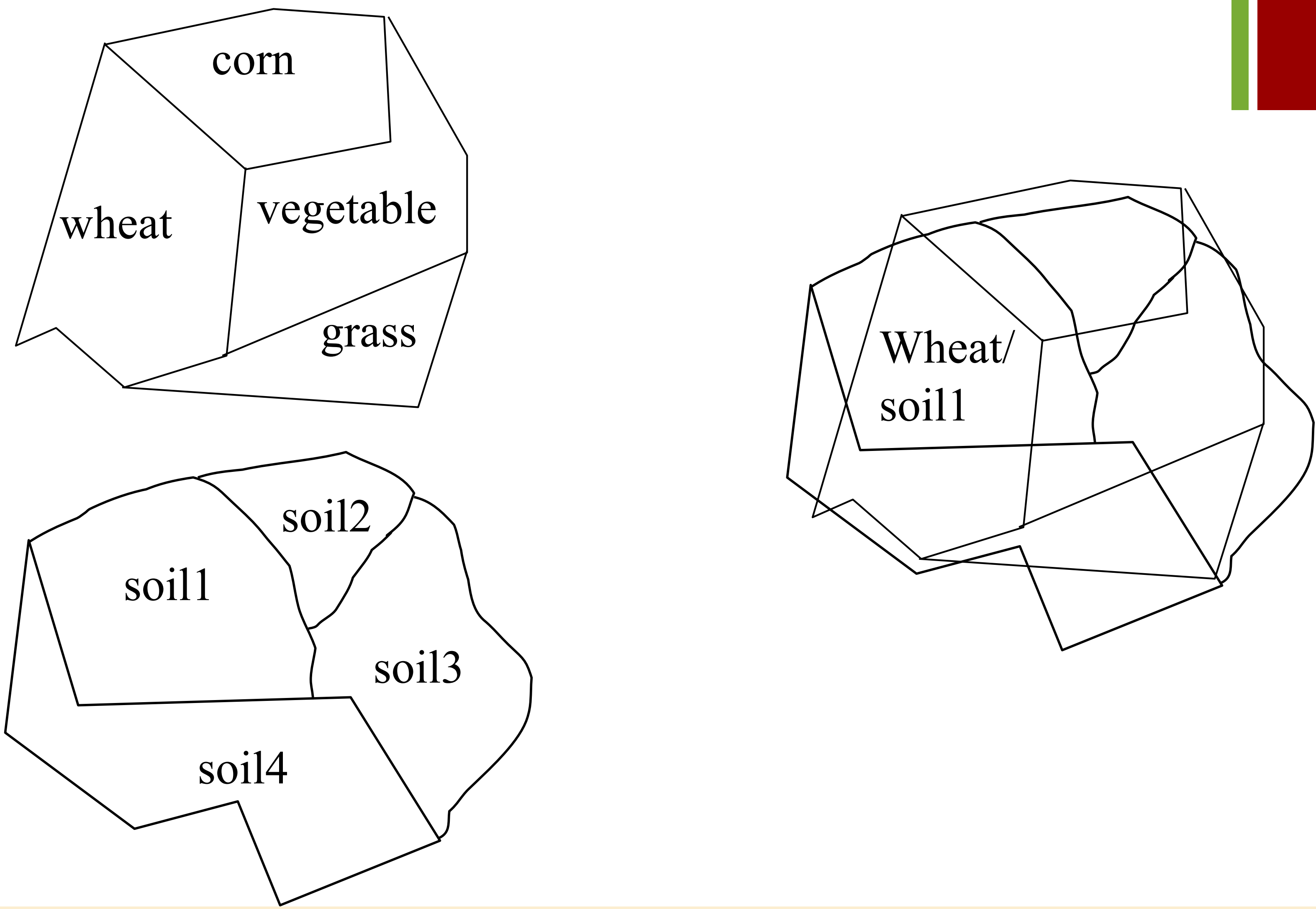


+ Amalgamation Query



…important applications in spatial OLAP and spatial data mining

+ Overlay Query



… also important applications in spatial OLAP and spatial data mining

+ Denoting SDT Values Input

- Problem: “Hong Kong” vs. the **geometry** of Hong Kong

1. Define named atomic data type values

```
DEFINE Hong Kong
ELEMENT  SELECT  boundary
         FROM    city  // city(name, boundary, ...)
         WHERE   name = “Hong Kong”;
```

2. Use spatial constants

```
WHERE  boundary inside POLYGON(100.2, 102.3; 1.5, 107; ....);
```

+ Fundamental DB Operations (I)

- Selection

```
SELECT  *
FROM    river // river(name, route, …)
WHERE   route intersect ClearWaterBayRoad;
```

- Join

```
SELECT  river.name, road.name
FROM    river, road // road(name, route, …)
WHERE   river.route intersect road.route;
```

+ Fundamental DB Operations (II)

- Function applications

```
SELECT  river.name, road.name,
         intersection(river.route, road.route)
FROM    river, road
WHERE   river.route intersect road.route;

SELECT  river.name,
         length(intersection(river.route, NewTerritories))
FROM    river
WHERE   river.route intersect NewTerritories;
```

+ Fundamental DB Operations (III)

- Other set operations

```
SELECT      amalgamation(area), round(amount/100)*100
FROM        rainfall //rainfall(area, date, amount)
WHERE       date = '08/02/2021'
GROUP BY round(amount/100);
```

… man spatial operations are not standard, so you may need to define it. For example, amalgamation takes a set of polygons and return the boundary of all polygons combined

+ Spatial Data Generalization

- Reduce level of details (LoDs) without reducing key characteristics of a spatial dataset
  - Improve processing efficiency
  - Essential to support operations on heterogeneous data
  - Adapt data to suit target device resolution
- Generalization of vector data is difficult
  - Non-algorithmic
  - Multiple criteria: metric, semantic, topological, gestalt

+ SDBMS Architectures

- Requirements
  - Representations for spatial data types
  - Procedures for spatial operations, including spatial join algorithms
  - Spatial index structures, and access operations
  - Spatial query optimizer with cost functions & statistics
  - Spatial query language, incorporating spatial data types, operations and graphical I/O
- Three generations
  - Using file systems; using standard RDBMS; integrated systems

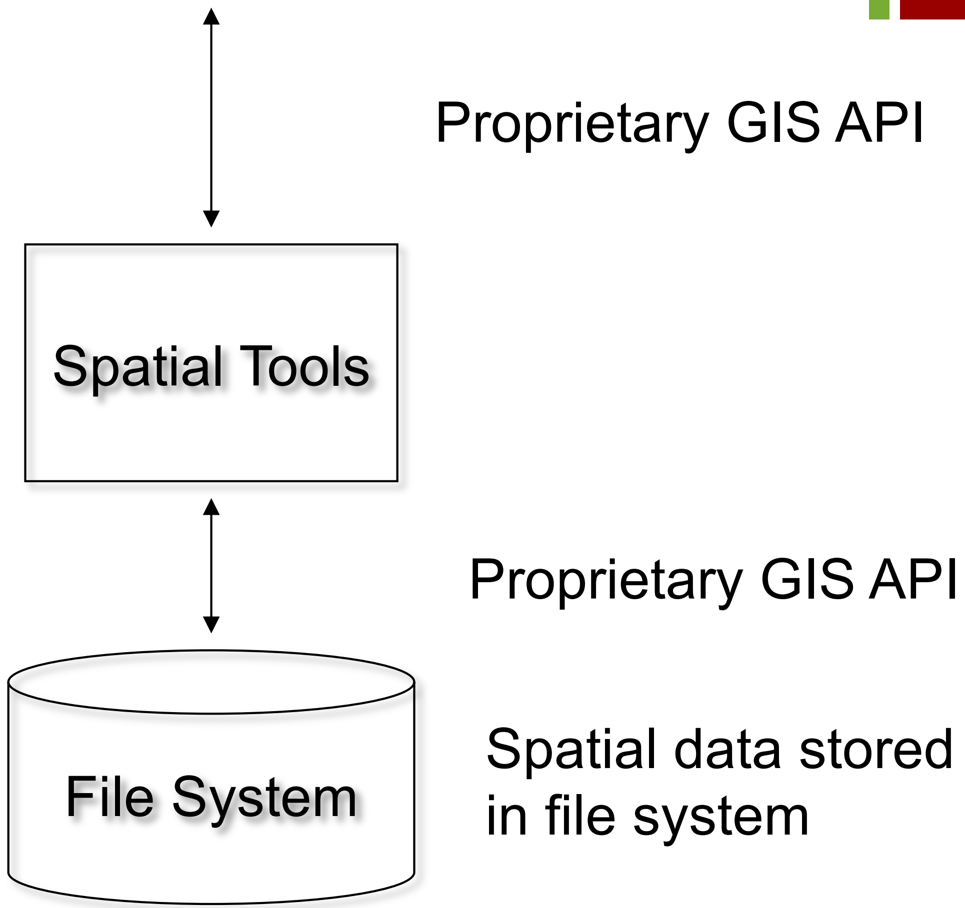
+ 1st Generation: Using File System

Problems:

•no high level data definition

•no flexible querying

•no DBMS support (eg, TM)



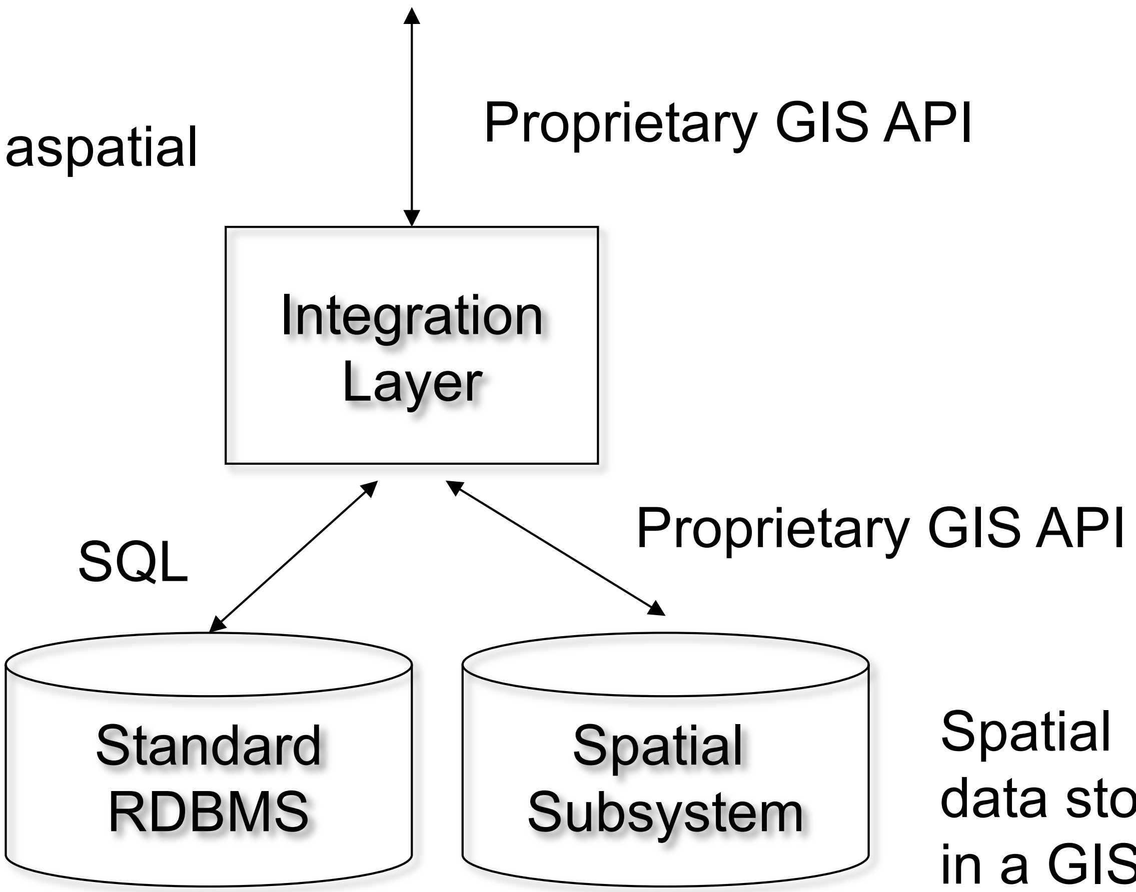
+ 2nd Generation: Using RDBMS (I)

Dual architecture:

•easy implementation

•efficient for spatial or aspatial queries

•but not both!



Aspatial data stored in a RDBMS

Spatial data stored in a GIS system

+ 2nd Generation: Using RDBMS (II)

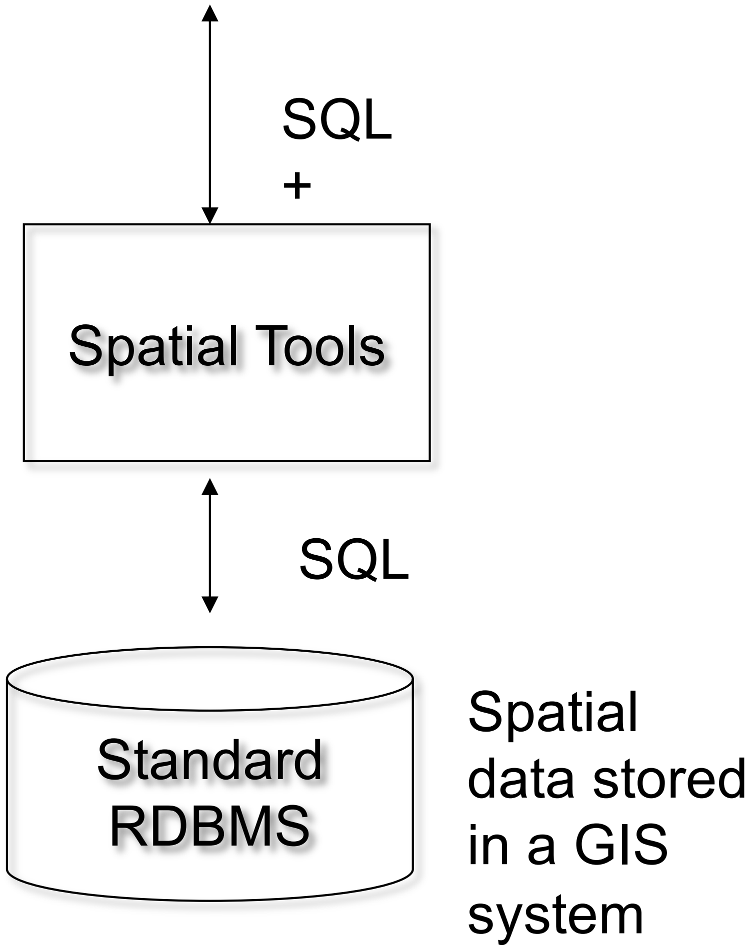
Layered architecture:

•representing SDT as a **set of tuples**, one tuple per point or line segment; or as **long fields** (ie, uninterpreted byte strings).

•mapping SDT into numbers and using B-Tree as index.

•right step towards integrated system

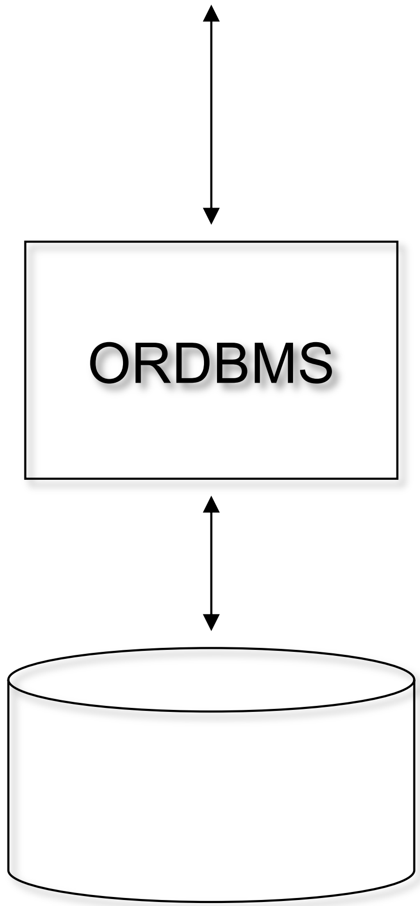
•poor spatial performance.



+ 3rd Generation: Part of ORDBMS

Integrated architecture:

- Future direction
- Possibility for users to extend the system
- Now a mature technology
- Performance still a problem for for complex queries
- Transaction Management is hard (for long running transactions)



+ SDBMS Implementation

- Continues space v discrete computer numbers
- High level query language (eg, SQL-like)
- Spatial query processing (at least spatial join)
- Spatial data access methods (indexing)
- Other DBMS issues
  - Query optimization
  - Transaction management
  - Integrity and consistency
  - Parallel processing
  - Spatial data warehousing and data mining

+ Spatial DB and Graph DB

- A graph database (GDB) is a database that uses graph structures for semantic queries with nodes, edges, and properties to represent and store data
  - For example: what's the different between a social network graph and a road network?
- GDB is related to SDB, but with quiet different focus
  - Oracle 12 combines spatial and graph into one data cartridge

+ Products and Systems

- Most GIS products don not handle databases aspects anymore

- Most famous one
  - Oracle Spatial (since 1994)
  - PostgresSQL (PostGIS since 2001)
  - Micorsoft SQL (since 2008)
  - Many NoSQL systems such as MonetDB, Redis, CouchDB
  - Many modern Hadoop-based systems such as GeoMeda

- Many spatial datasets are publicly available

… you can install a system, such as Oracle or PostGIS, and download some data to get hands-on experiences

+ Summary

- Spatial DBMS is still a DBMS
  - Data types and query languages may look different, but are still the same in essence
  - Just need to think data types in its intrinsic meaning
- If your job is to develop a database system to manage spatial data, this lecture gives you a good example
  - What about images? Videos? Graphs?
- You need to understand the concepts introduced
  - Spatial data, data types, operations, queries