+

INFS4205/7205 Advanced Techniques for High Dimensional Data

An Introduction to Spatial Databases

Semester 1, 2021

The University of Queensland

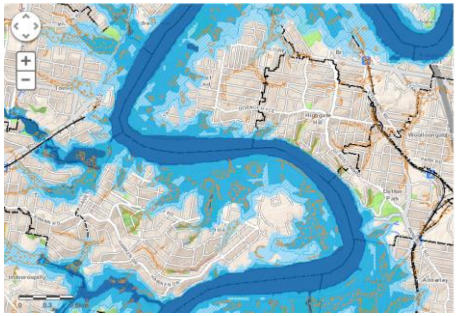
+ Advanced Techniques for High Dimensional Data

- Course Introduction
- Introduction to Spatial Databases
- Spatial Data Organization
- Spatial Query Processing
- Managing Spatiotemporal Data
- Managing High-dimensional Data
- Other High-dimensional Data Applications
- When High-dimensional Data Meets AI
- Route Planning
- Trends and Course Review

+ Why Starting with Spatial Databases?

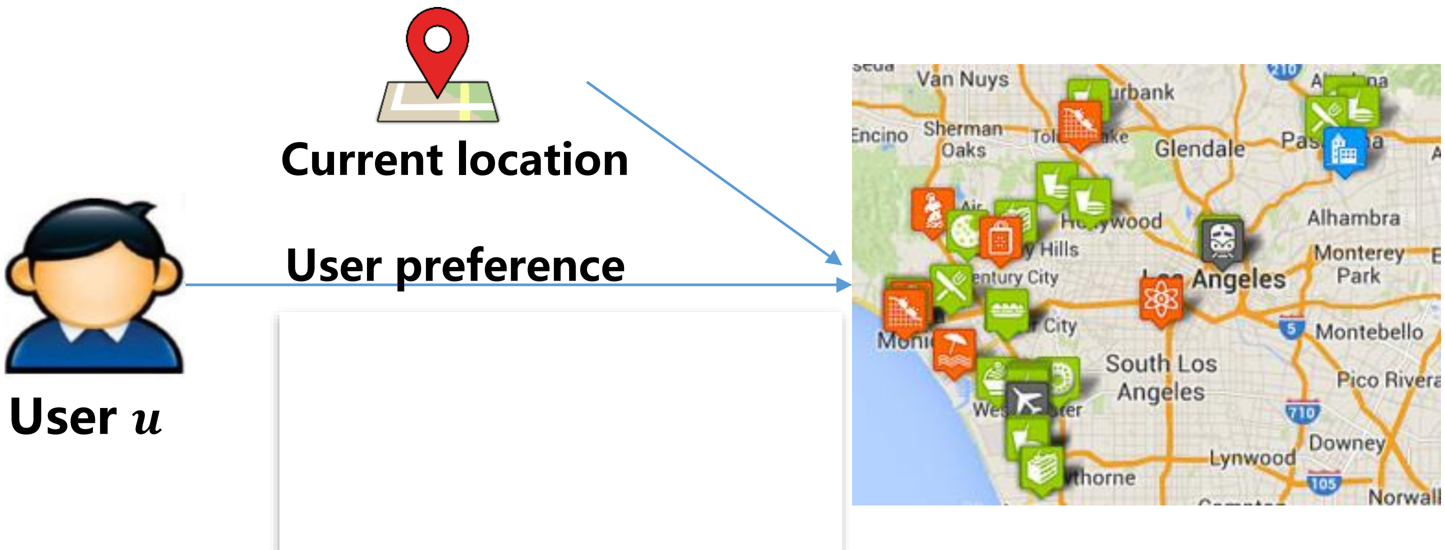
- One of the most widely used multidimensional databases
  - The most basic multidimensional data
  - Foundation of understanding other high-D data

- Many important application domains have **spatial data** and **queries**. Some Examples:
  - **Insurance Risk Manager**: Which homes are most likely to be affected in the next great flood on the Brisbane river?
  - **Location-based Recommendation**: A location-based recommendation is an information filtering service, which selectively returns spatial items (e.g., venues, events, travel routes) to a user with the consideration of relevant spatial information (e.g., current/historical locations) and the personal preferences

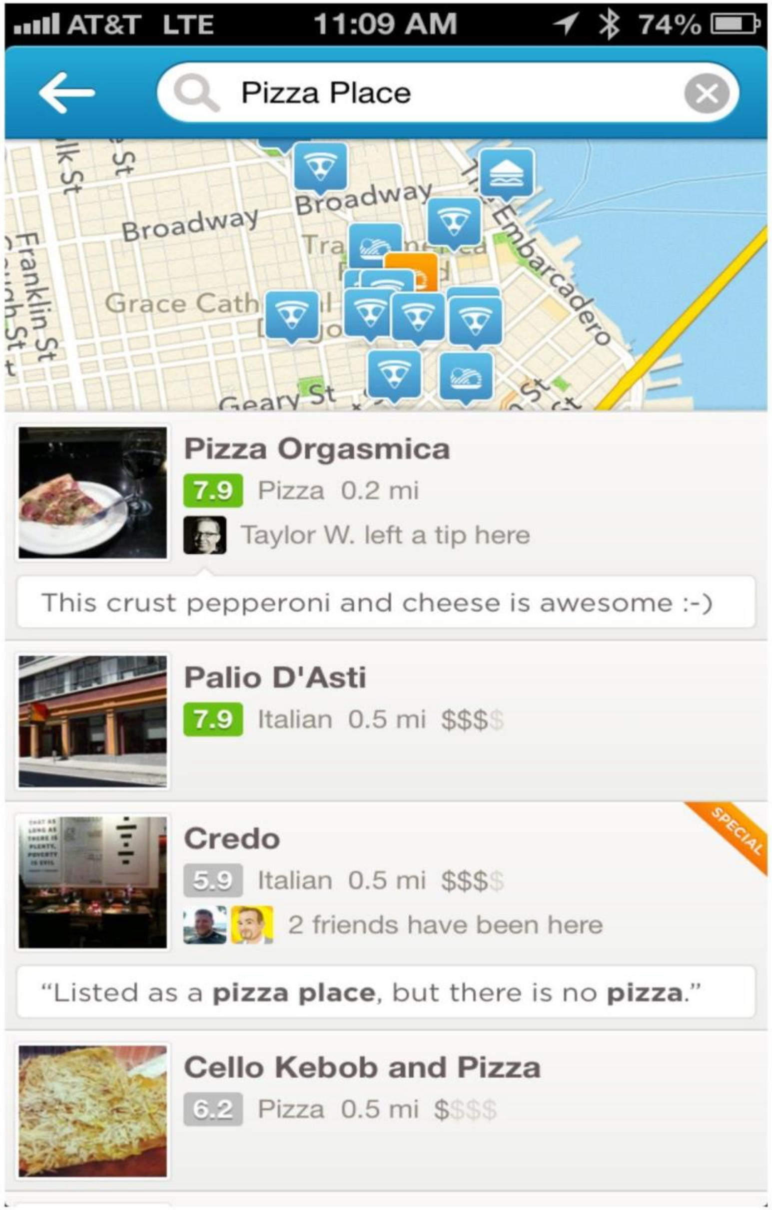


Location-based Recommendation

- **Given a user u with his/her current location l , recommend top-k spatial items that u would be interested in.**

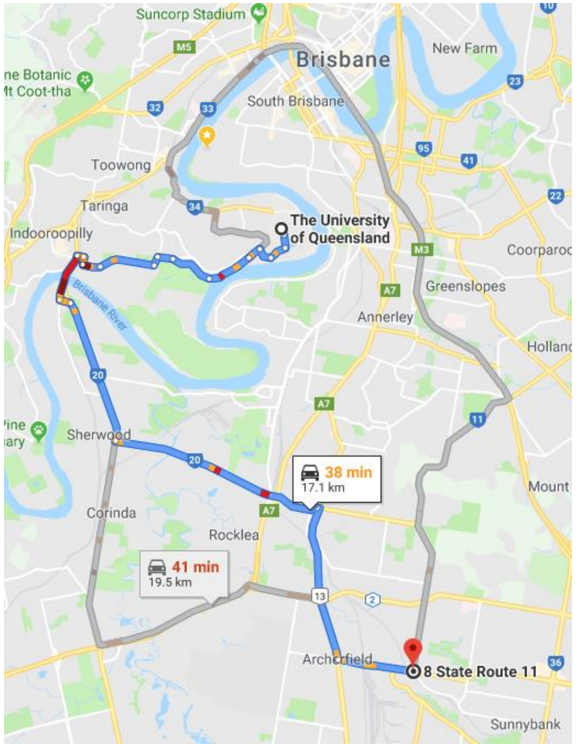


+ Location-based Search



+ Route Planning in Road Network

1. Shortest path from UQ to Coopers Plain
2. Fastest path from UQ to Coopers Plain when I depart at 5pm
3. Safest path from UQ to Coopers Plain when I depart at 5pm
4. Fastest path from West End to Airport with toll fee under $6 (Go Between? M7?)
5. I want to spend at least 3 hours at UQ, 1 hour at Garden City, 1 hour at Southbank Parkland and arrive home by 9pm, what's the schedule to spend the least time on transportation?



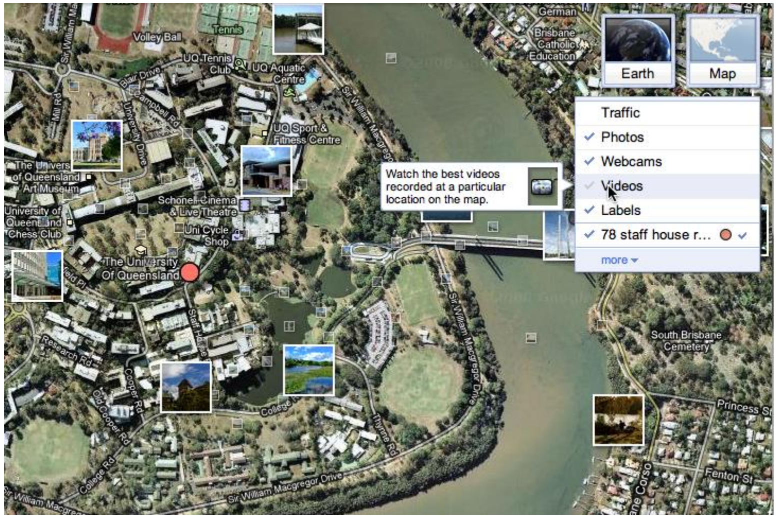
+ How About Relational DBMS?

- Queries
  - Where is the tutorial room of INFS4205?
  - Which courses are delivered in GP Building?

Non-spatial

  - Which buildings are adjacent to *the UQ* lake?
  - Which buildings are adjacent to *a* lake?

spatial



+ What’s Special about Spatial Queries?

- Retrieval & update of spatial data is based on the **spatial location** of a spatial object
  - *(vs. alphanumeric attributes in RDBMS)*
- Fast execution of **geometric operations like** the intersection of spatial objects
  - *(vs. simple comparison (=, >, <) in RDBMS)*
- Fast execution of **complex spatial queries**
  - Distance and knn queries
- RDBMS Obvious limitations
  - Limited data types – no support for multidimensional data!
  - Limited query types

+ 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 a SDBMS is

+ Why Spatial DBMS?

- Huge amounts of spatial data, extensive and comprehensive
  - Google Map: 20 Petabytes of data in 2012
    - Nowadays: every phone that enables traffic condition sharing…
  - Uber/DiDi: Terabytes of taxi trajectory per day
  - OpenStreetMap: Planet.osm 1166 GB for the static Earth
    - NearMap
  - NASA satellites image: TB per day
  - Twitter / Weibo with geo-tags
  - Weather / Climate Data: windy.com, every 6 hours
  - Foursquare check-in: 3 billion visits / month in 2017
  - Cellular signalling data from Vodafone / Telstra…
  - …

+ Why Spatial DBMS?

- Increasing needs to store, search and use spatial data, together with other data, efficiently, enterprise-wide
  - **File system-based** solutions are not good enough
    - Hard to manage, search, share,…
  - **Application-based** solutions are not good enough either
    - There are many common spatial data and operations across these different applications
    - Data is hard to share
    - Lots of repetitive and redundant work will be done in each app
  - **RDMBS**
    - Limited data types – no support for spatial data!
    - Limited query types – no support for geometric operations and spatial queries
    - Limited indexing structures – no support for multidimensional data

+ Why Spatial DBMS?

- RDBMS cannot support spatial data directly
  - Store the spatial objects into tables?
  - How about spatial operations?
    - Find the fuel prices of all petrol stations with 5km of my current location (e.g., GasBuddy)
    - List of the suburbs along Brisbane river
    - What is the average speed of M3 from Kangaroo Point to Milton during the last 10 minutes?
  - How to retrieve the spatial data efficiently?
    - Hash?
    - B-Tree?

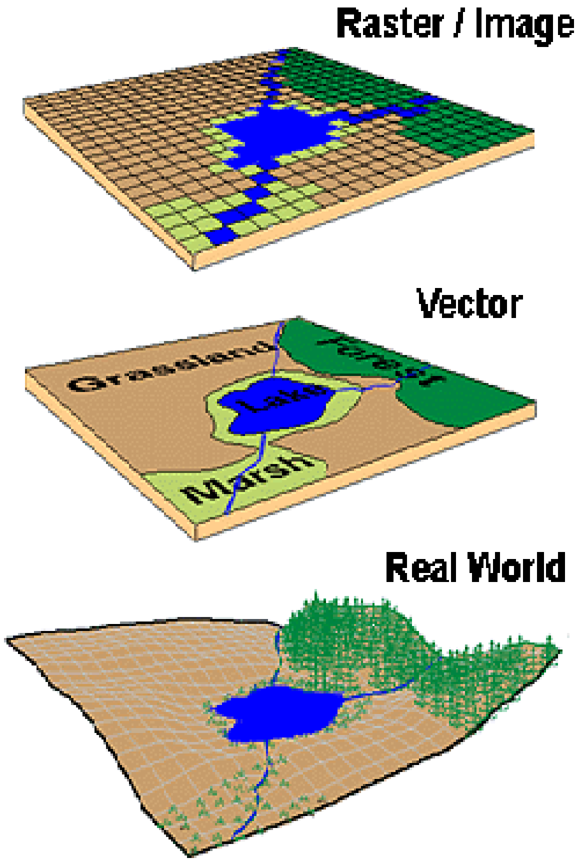
+ Spatial Data

- Any data with a location component
  - 2D Space
    - Geographical space: GIS, Urban Planning
  - 3D Space
    - (x, y, z): The universe, brain model, molecule structure…
    - (x, y, t): Trajectory
- Two types of spatial data
  - Those data about the space (e.g., road networks, maps)
  - Those data about objects (e.g., location of shops, location about cars)

+ More Spatial Data

- Natural Area Data
  - Soil types, land use (industrial, agriculture, zoning 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 (phones, sewers, water, electricity etc)
  - Roads (centrelines, curb lines, intersections, lights)

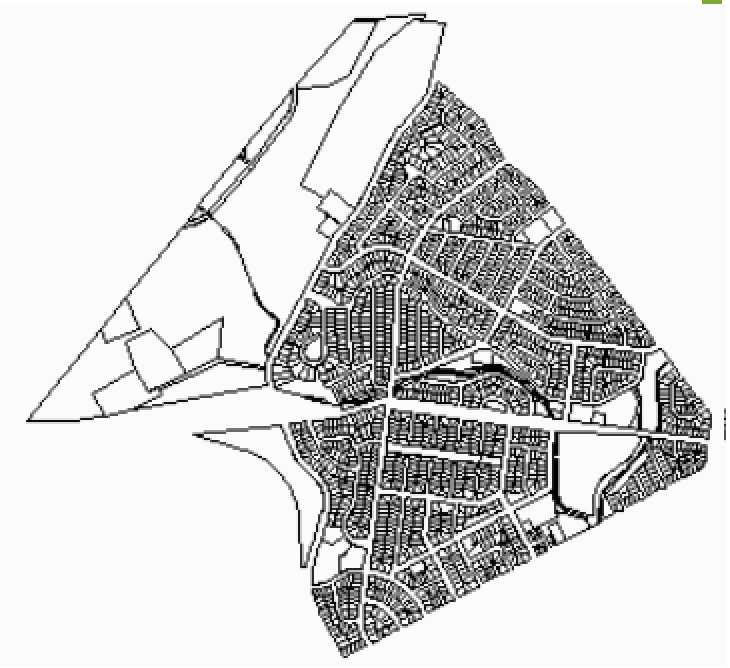
+ Modelling Spatial Data

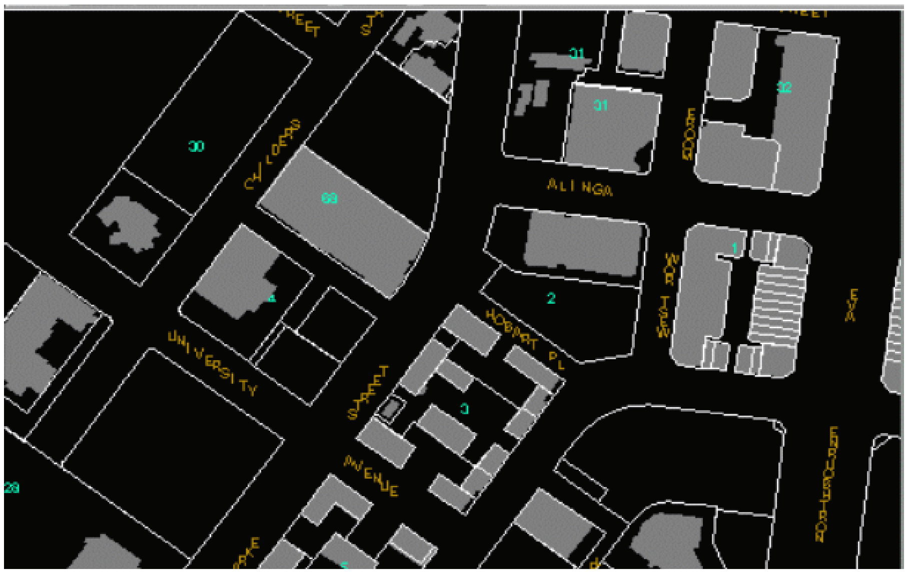


+ Vector Data

- A spatial object is described by a sequence of points
  - Defined using geometric data types
    - Points, Lines and Polygons / Region
  - Work with coordinates
    - Each point is denoted by a coordinate

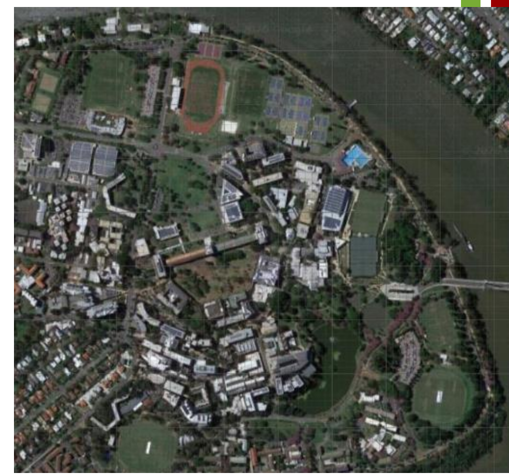
- Advantages
  - Suitable for processing & manipulation
  - More compact
  - Support of queries by spatial relationships
    - List all buildings adjacent to UQ lake;
    - Find rivers that intersect with M1



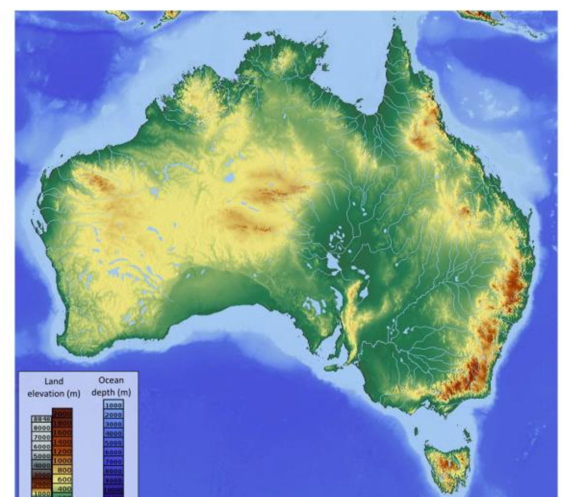


+ Raster Data

- A spatial object is described by a set of pixels
  - E.g., satellite imagery, remote sensing
  - Work with pixels

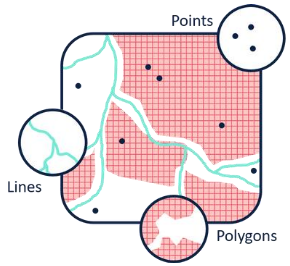


- Advantages
  - Suitable for display
  - Query by colour, texture etc.
  - Containing more rich information
    - Continuous information like
      - Temperature / Elevation / Humidity
      - Census / Health / Employment

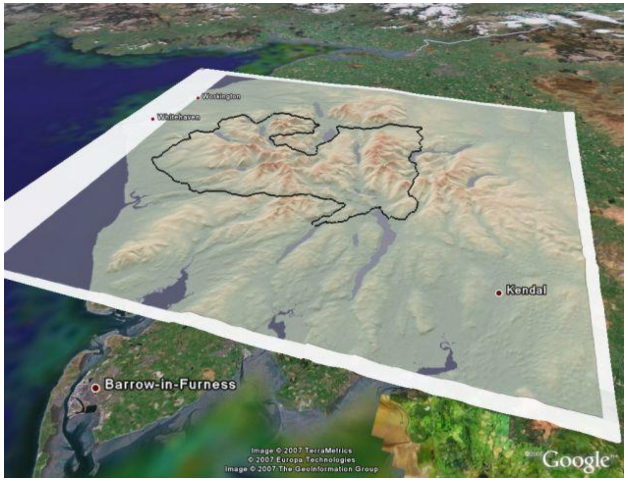


- Disadvantages
  - Pixelated look
  - Hard to perform topology rules
  - Large data size

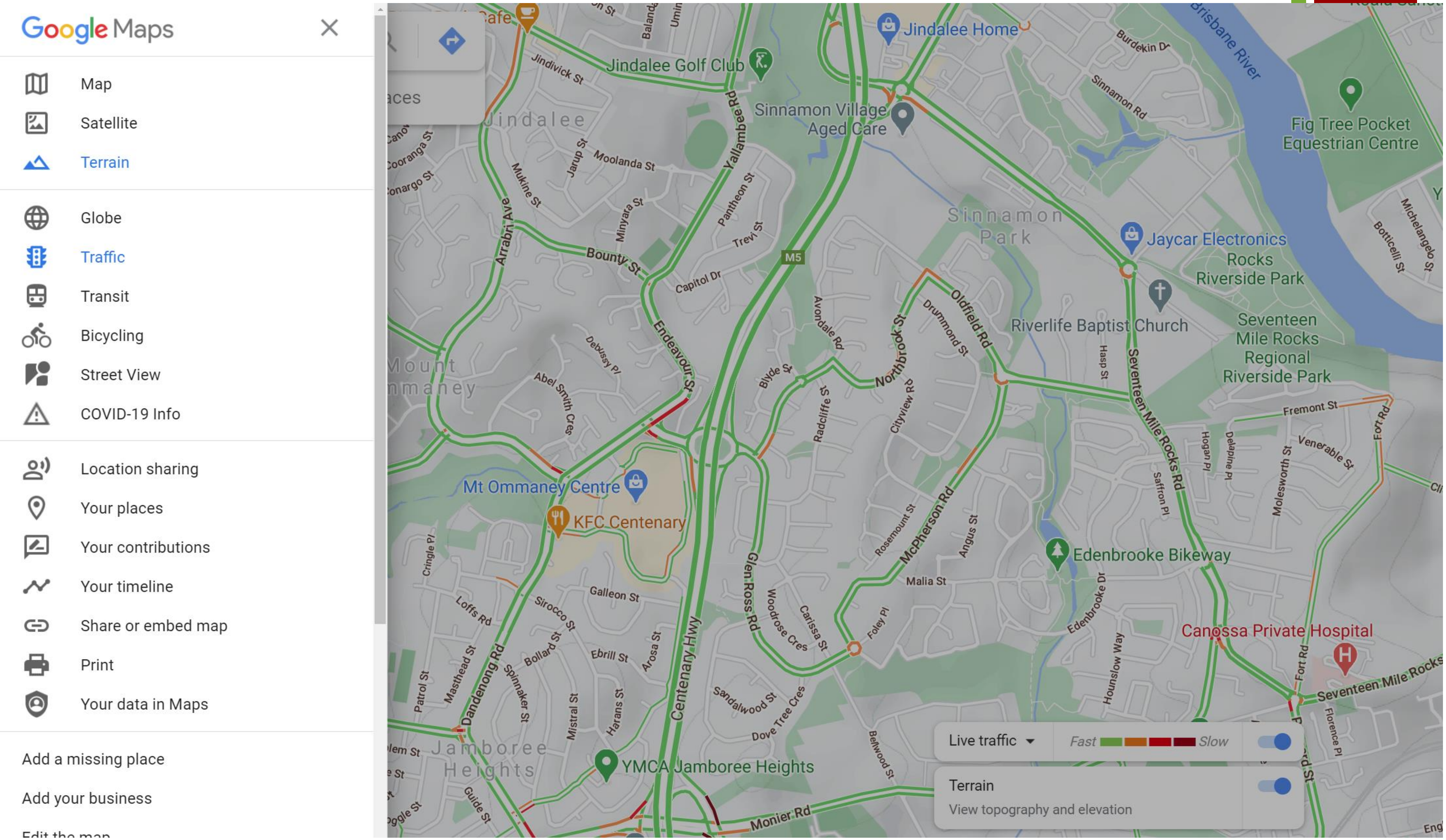
+ Vector vs Raster Data



- 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 the vector data in spatial databases

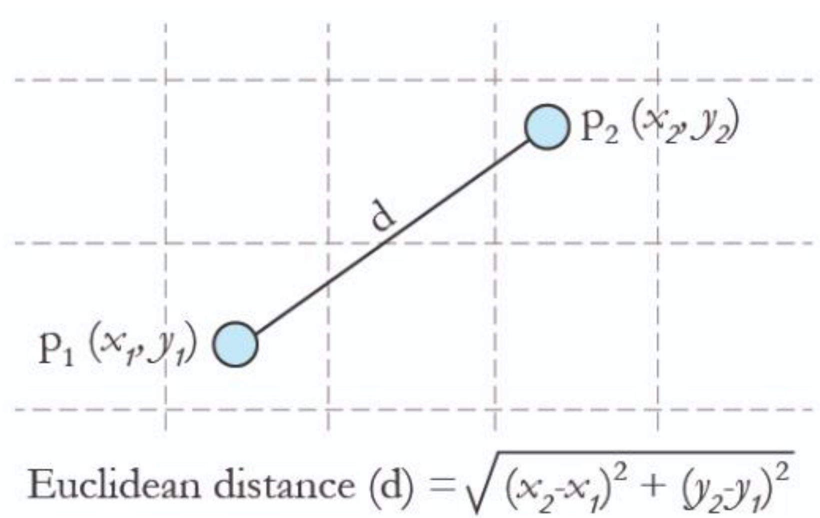


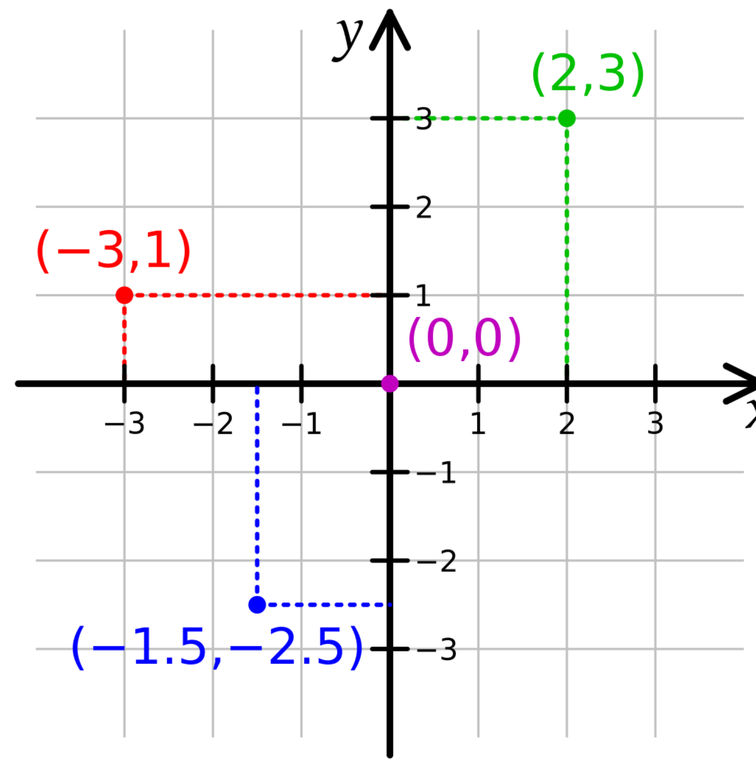
+ Google Map


Google Maps
Map
Satellite
Terrain
Globe
Traffic
Transit
Bicycling
Street View
COVID-19 Info
Location sharing
Your places
Your contributions
Your timeline
Share or embed map
Print
Your data in Maps
Add a missing place
Add your business
Jindalee Golf Club
Jindalee Home
Sinnamon Village Aged Care
Jindalee
Sinnamon Park
Brisbane River
Fig Tree Pocket Equestrian Centre
Jaycar Electronics
Rocks Riverside Park
Riverlife Baptist Church
Seventeen Mile Rocks Regional Riverside Park
Mt Ommaney Centre
KFC Centenary
Edenbrooke Bikeway
Canossa Private Hospital
YMCA Jamboree Heights
Jamboree Heights
Centenary Hwy
Monier Rd
Seventeen Mile Rocks Rd
Dandenong Rd
McPherson Rd
Glen Ross Rd
Endeavour St
Bounty St
Oldfield Rd
Northbrook St
M5
Live traffic
Fast
Slow
Terrain
View topography and elevation

+ Coordinate Systems

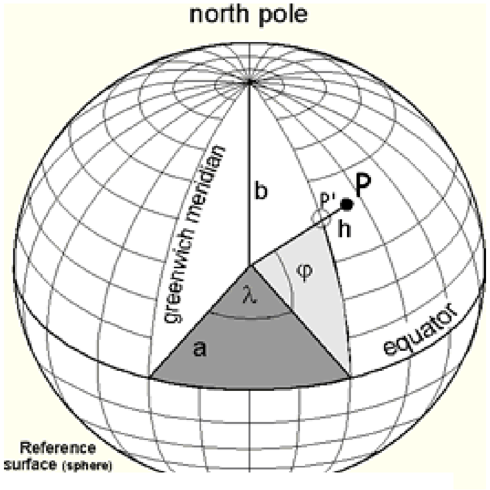
- Assigning coordinates to a location and establishing relationships between sets of such coordinates
- Cartesian Coordinates
  - (x, y)





+ Geographical Coordinate Systems

- Lat/long system measures angles on spherical surfaces
  - Lat/long values are NOT Cartesian (X, Y) coordinates
    - 1° of longitude at the equator ≠ 1° of longitude near the poles
    - 1° $longnitude \approx 111 \times \cos(latitude)$



- Euclidean approximation of the distance between two geolocations:

$$\partial y = 12430 \frac{|lat_1 - lat_2|}{180}$$

$$\partial x = 24901 \frac{|lng_1 - lng_2|}{360} \cos\left(\frac{lat_1 + lat_2}{2}\right)$$

$$d = \sqrt{\partial x^2 + \partial y^2}$$

- Network Distance (e.g., road network)

+ Spatial Data Acquisition

- There are many publically available spatial datasets
  - Government sources, and other providers such as Google
    - http://www.spatial-data.brisbane.qld.gov.au
  - Open Street Map / Near Map
  - Geography Mashups
- Generating new data
  - Remote sensing
  - Field data
    - Survey data
    - GPS devices
  - Crowd Sourcing (e.g., GasBuddy)

+ Spatial Data Accuracy

- Accuracy – how accurate is the data (i.e., 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
    - Scale is 1:10,000, meaning that 1 unit of measurement on the map represents 10,000 of the same units on the earth's surface
  - Resolution: the size of the smallest feature that can be represented in a surface
    - When scale decreases, resolution decreases too
  - Precision: how exactly the location is recorded (i.e., # of digits, 3.14 vs. 3.1415926)
- Errors can be introduced from many sources, from data capturing sensors, data editing and pre-processing operations

+ Spatial DBMS

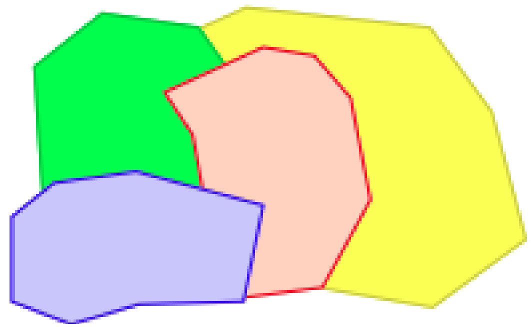
- A spatial database system is a database system
- It offers spatial data types in its data model and the query language
- It supports spatial data types in its implementation, by providing spatial indexing and efficient algorithms for spatial join queries and other types of spatial queries
  - Retrieve the spatial data without scanning the whole set

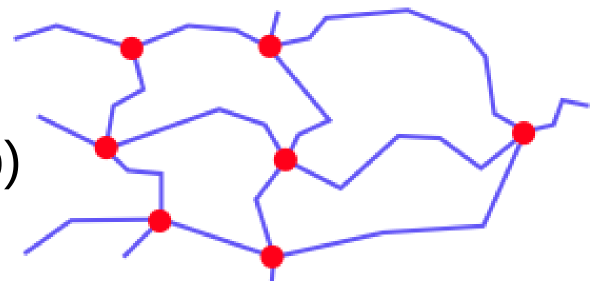
+ GIS and Spatial Databases

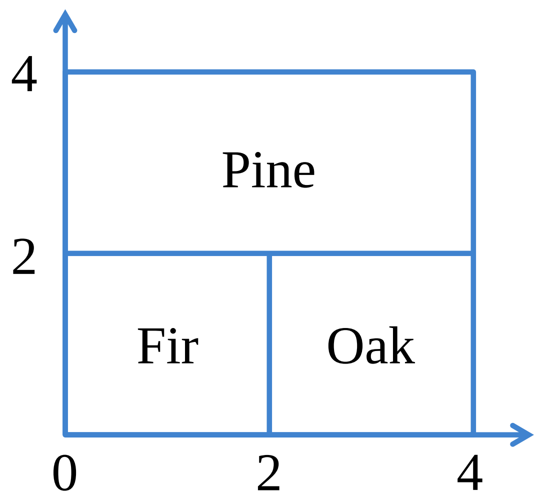
- GIS Applications
  - Front End
  - Data capture, editing, conversion, conflation
  - Map generation
  - Image processing
  - Data analysis (in application areas)
  - Use both raster and vector
- Spatial DBMS
  - Back End
  - Integrated management of spatial and non-spatial data
  - Independent from applications
  - RDBMS-comparable performance

+ Spatial Data Models

- **Objects** in space (Object-based)
  - Single objects
    - Point (location, POI)
    - 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)







Area-ID	Dominant Tree	Area
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)]

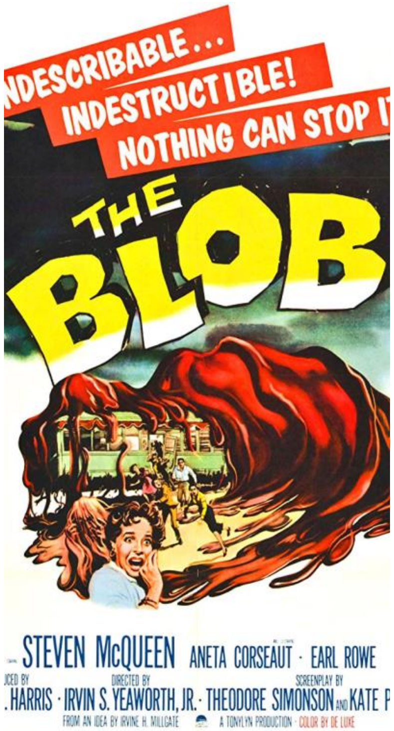
+ Spatial Data Models

- **Space** itself
  - The spatial extent, Euclidian space or other types of spaces (frame/location references, dimension#, distance measures)
- Space is **continuous real number**
  - Computer numbers are finite and discrete, with limited approximation
- Problems: numerical **rounding errors**

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

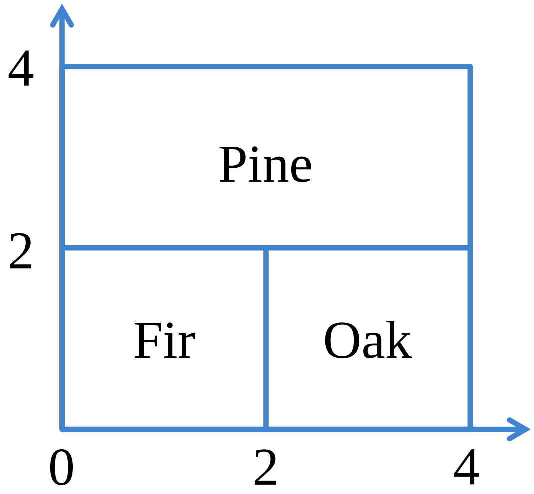
+ Geometry in Data Model

- How to represent real-world spatial objects using database objects (based on the data model)
- Relational data model
  - BLOB
    - Binary Large OBject
      - “The thing that ate Cincinnati, Cleveland, or whatever”
    - Create a table to store the land use information
    - `landuse(area: BLOB, type: string)`
      - Not searchable, Cannot be indexed, No operation…



+ Geometry in Data Model

- How to represent real-world spatial objects using database objects (based on the data model)
- Relational data model
  - Table
    - `landuse(areaID: number, type: string);`
    - `polygon(areaID: number, pntID: number, x: number, y: number)`



Area-ID	Dominant Tree
FS1	Pine
FS2	Fir
FS3	Oak

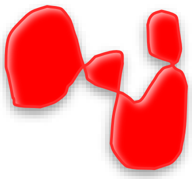
Area-ID	pntID	x	y
FS1	0	0	2
FS1	1	4	2
FS1	2	4	4
FS1	3	0	4
…	…	…	…

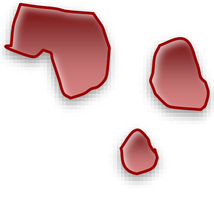
+ Geometry in Data Model

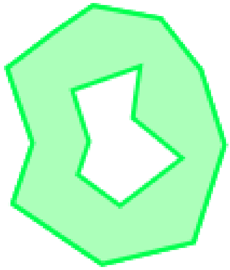
- How to represent real-world spatial objects using database objects (based on the data model)
- Spatial Data Model
  - `landuse(area: polygon, type: string)`
  - Data Type
    1. Valid Values
    2. Supported Operations

+ Spatial Data Model in Oracle

- Element, Geometry and Layer
- **Element:** the basic building block of a geometric feature
  - **Point data:** One point, stored as an (x, y) pair
  - **Line data:** Two points 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
    - Simple Polygon: Both boundary and the interior
    - Complex polygons (Geometry)
      - Self-intersecting boundary
      - Multiple dis-connected components





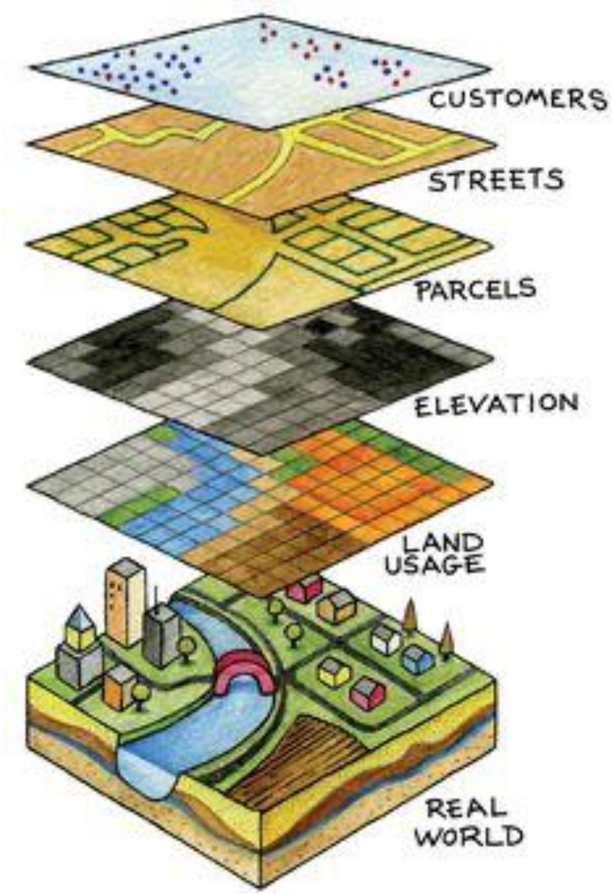


Spatial Data Model in Oracle

- **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
  - A geometry might describe a lake
    - A polygon with nested polygons for islands
    - Attributes such as lake name, water capacity, fauna, flora, …

+ Spatial Data Model in Oracle

- **Layer:** a **collection of geometries** having the same attribute set
  - Examples: soil types, road network, political boundaries, population density, crops, weather conditions,…

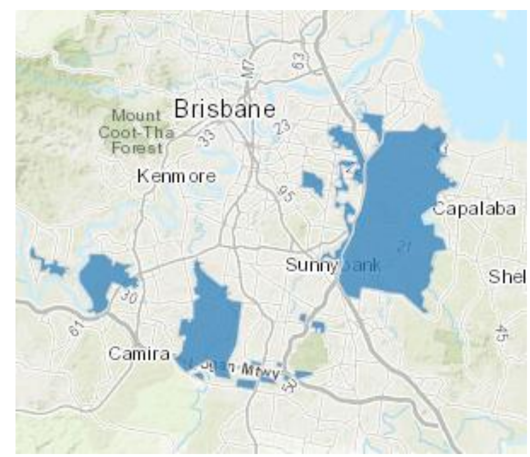


+ Spatial Data Model in Oracle

- **Layer:** a **collection of geometries** having the same attribute set
  - Examples: soil types, road network, political boundaries, population density, crops, weather conditions,…

Magpie Swooping

Transport Noise Corridor Overlay



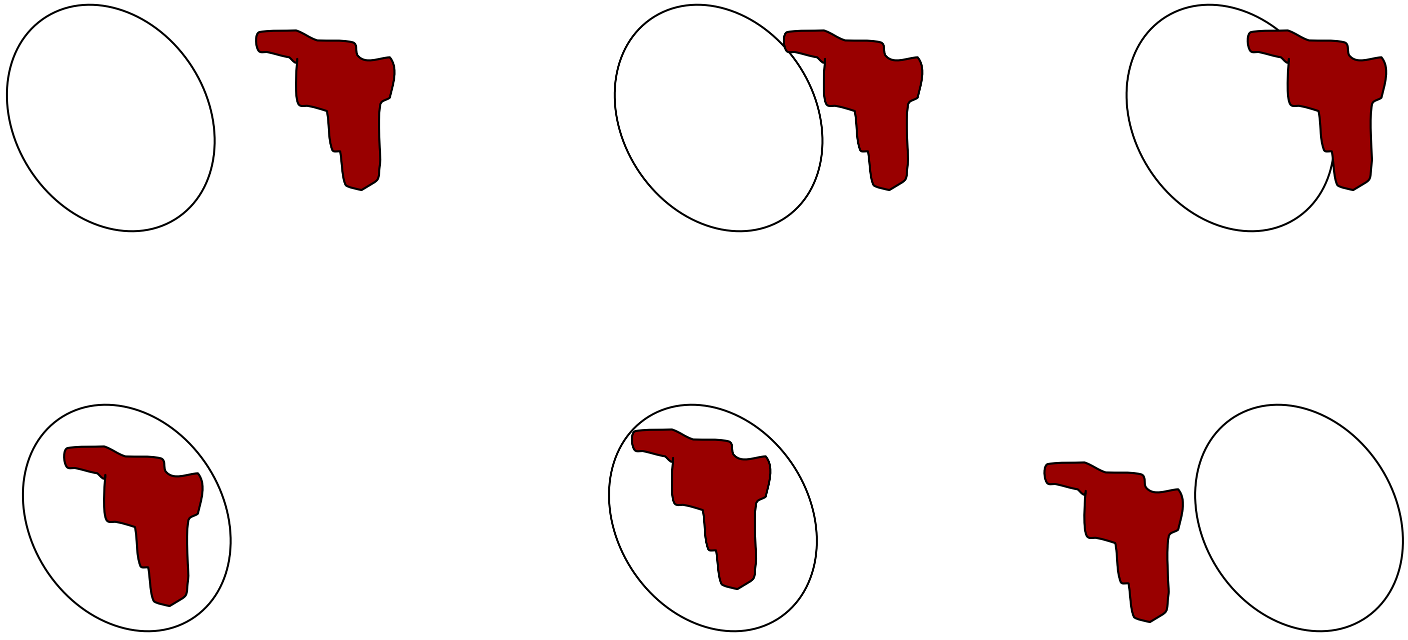
Koala Habitat Areas

+ Spatial Relationships

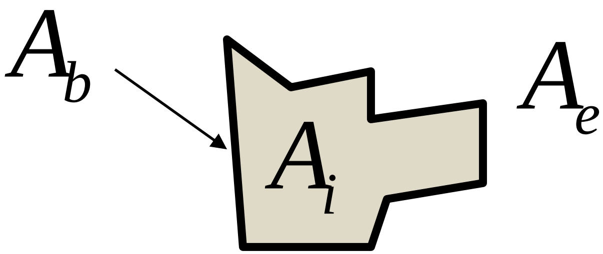
1. **Topological**
   - E.g., *inside*, *intersect*, *adjacent*
   - Invariant under translation such as rotation and scaling
2. **Directional**
   - E.g., *above*, *left, north of*
   - May change with rotation
3. **Metric**
   - E.g., *distance*
   - May change with scaling

+ Defining Spatial Topological Relationship

- Q: How to define a spatial relationship precisely?



+ The 9-Intersection Matrix



A_b: boundary

A_i: interior

A_e: exterior

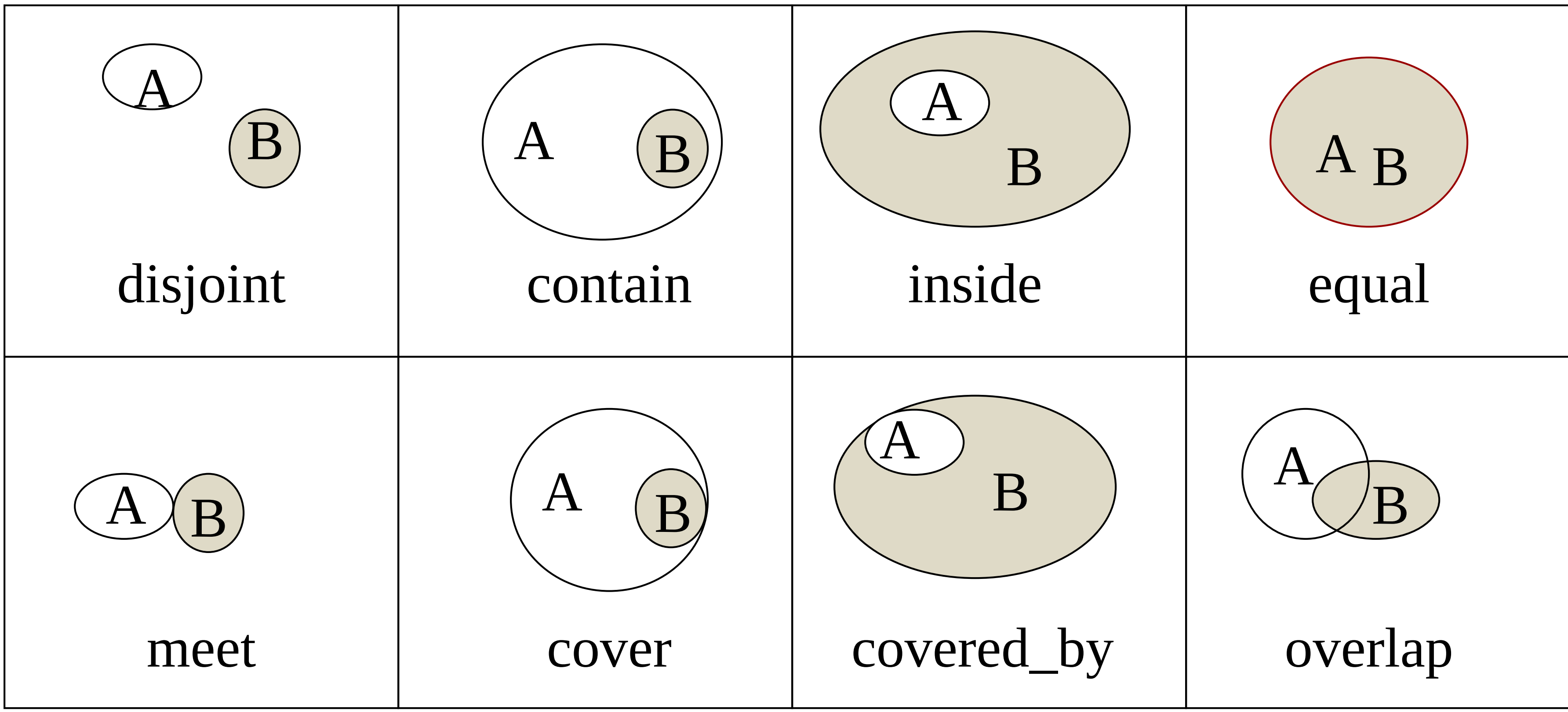
$$\begin{pmatrix} A_b \cap B_b & A_b \cap B_i & A_b \cap B_e \\ A_i \cap B_b & A_i \cap B_i & A_i \cap B_e \\ A_e \cap B_b & A_e \cap B_i & A_e \cap B_e \end{pmatrix}$$

Each element is either 1 or 0.

Egenhofer, M.J.; Herring, J.R. (1990). "A Mathematical Framework for the Definition of Topological Relationships"

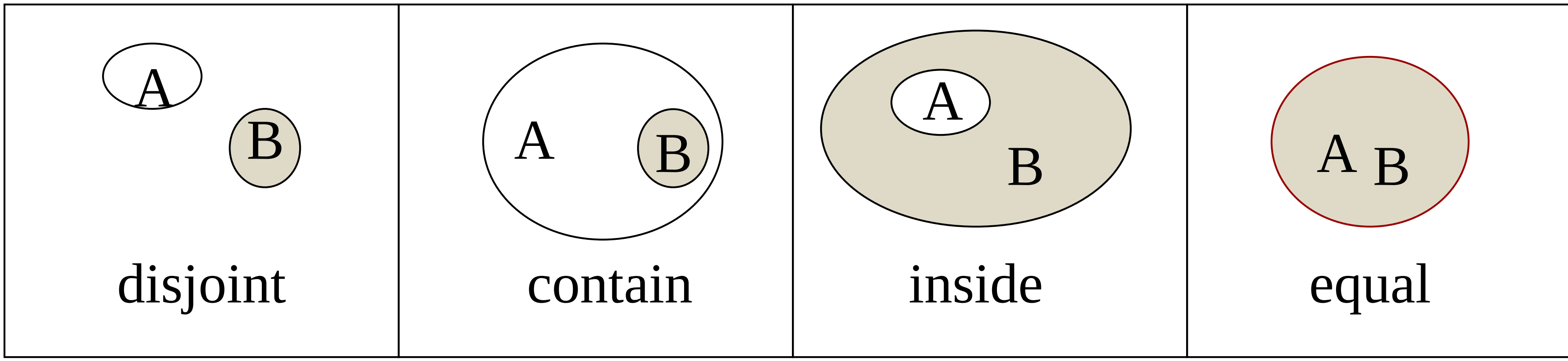
The 8 Spatial Relationships (I)

- Complete, and Mutually Exclusive.



The 8 Spatial Relationships (I)

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



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

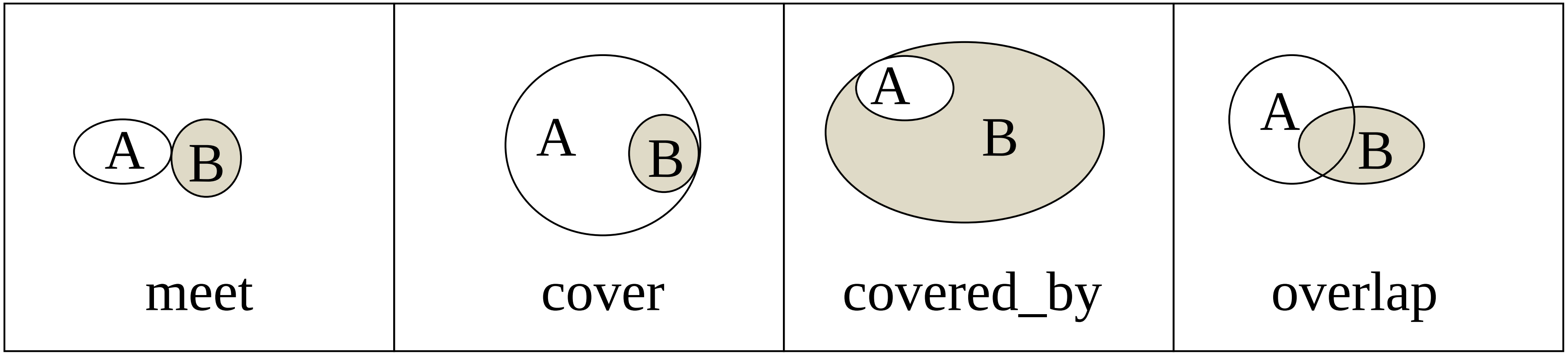
disjoint | contain | inside | equal

The 8 Spatial Relationships (I)

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

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

meet cover covered_by overlap



+ 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

...all papers are available at the course website