Module 4: Spatial & Multimedia Databases

Spatial Databases

- Also known as a “geospatial database” — is built to capture and store the points, lines, and areas of cartographic information that we refer to as spatial data.
  - Geographic coordinates: Two-dimensional (2D) coordinates
  - Geometric shapes: Lines or polygons
  - GPS data collected using Global Positioning System (GPS) receivers, Drones, and Wireless sensors, etc.
- Most spatial databases allow the representation of simple geometric objects such as points, lines and polygons.

Value of SDBMS – Spatial Data Examples

- Examples of non-spatial data
  - Names, phone numbers, email addresses of people
- Examples of Spatial data
  - Census Data
  - NASA satellites imagery - terabytes of data per day
  - Weather and Climate Data
  - Rivers, Farms, ecological impact
  - Medical Imaging

Value of SDBMS – Users, Application Domains

- Many important application domains have spatial data and queries. Some Examples follow:
  - **Army Field Commander**: Has there been any significant enemy troop movement since last night?
  - **Insurance Risk Manager**: Which homes are most likely to be affected in the next great flood on the Mississippi?
  - **Medical Doctor**: Based on this patient's MRI, have we treated somebody with a similar condition ?
  - **Molecular Biologist**:Is the topology of the amino acid biosynthesis gene in the genome found in any other sequence feature map in the database ?
  - **Astronomer**:Find all blue galaxies within 2 arcmin of quasars.

Queries

- Non-spatial queries:
  - List the names of all bookstore with more than ten thousand titles.
  - List the names of ten customers, in terms of sales, in the year 2001
- Spatial Queries:
  - List the names of all bookstores within ten miles of VIT
  - List all customers who live in Chennai and its adjoining states

What is a SDBMS ?

- A SDBMS is a software module that
  - can work with an underlying DBMS
  - supports spatial data models, spatial abstract data types (ADTs) and a query language from which these ADTs are callable
  - supports spatial indexing, efficient algorithms for processing spatial operations, and domain specific rules for query optimization
- Example: Oracle Spatial data cartridge, ESRI SDE
  - can work with Oracle 8i DBMS
  - Has spatial data types (e.g. polygon), operations (e.g. overlap) callable from SQL3 query language
  - Has spatial indices, e.g. R-trees

SDBMS Example

- Consider a spatial dataset with:
  - County boundary (dashed white line)
  - Census block - name, area, population, boundary (dark line)
  - Water bodies (dark polygons)
  - Satellite Imagery (gray scale pixels)

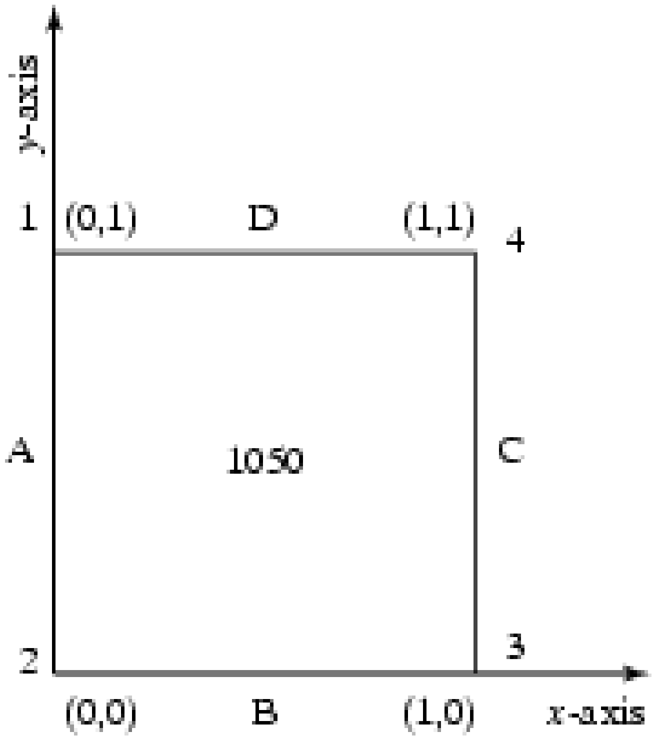
- Storage in a SDBMS table:

create table **census_blocks** (

name	string,
area	float,
population	number,
boundary	polygon);

Modeling Spatial Data in Traditional DBMS

- A row in the table census_blocks
- Question: Is Polyline datatype supported in DBMS?



Census_blocks

Name	Area	Population	Boundary
1050	1	1839	Polyline((0,0),(0,1),(1,1),(1,0))

Spatial Data Types and Traditional Databases

- Traditional relational DBMS
  - Support simple data types, e.g. number, strings, date
  - Modeling Spatial data types is tedious
- Example: Figure X shows modeling of polygon using numbers
  - Three new tables: polygon, edge, points
    - Note: Polygon is a polyline where last point and first point are same
  - A simple unit sqaure represented as 16 rows across 3 tables
  - Simple spatial operators, e.g. area(), require joining tables
  - Tedious and computationally inefficient

- Question. Name post-relational database management systems which facilitate modeling of spatial data types, e.g. polygon.

Mapping “census_table” into a Relational Database

Census_blocks

Name	Area	Population	boundary-ID
340	1	1839	1050

Polygon

boundary-ID	edge-name
1050	A
1050	B
1050	C
1050	D

Edge

edge-name	endpoint
A	1
A	2
B	2
B	3
C	3
C	4
D	4
D	1

Point

endpoint	x-coor	y-coor
1	0	1
2	0	0
3	1	0
4	1	1

Fig X

How is a SDBMS different from a GIS ?

- GIS is a software to visualize and analyze spatial data using spatial analysis functions such as
  - **Search** Thematic search, search by region, (re-)classification
  - **Location analysis** Buffer, corridor, overlay
  - **Terrain analysis** Slope/aspect, catchment, drainage network
  - **Flow analysis** Connectivity, shortest path
  - **Distribution** Change detection, proximity, nearest neighbor
  - **Spatial analysis/Statistics** Pattern, centrality, autocorrelation, indices of similarity, topology: hole description
  - **Measurements** Distance, perimeter, shape, adjacency, direction
- GIS uses SDBMS
  - to store, search, query, share large spatial data sets

How is a SDBMS different from a GIS ?

- SDBMS focusses on
  - Efficient storage, querying, sharing of large spatial datasets
  - Provides simpler set based query operations
  - Example operations: search by region, overlay, nearest neighbor, distance, adjacency, perimeter etc.
  - Uses spatial indices and query optimization to speedup queries over large spatial datasets.
- SDBMS may be used by applications other than GIS
  - Astronomy, Genomics, Multimedia information systems, ...
- Will one use a GIS or a SDBM to answer the following:
  - How many neighboring countries does USA have?
  - Which country has highest number of neighbors?

Components of a SDBMS

- Recall: a SDBMS is a software module that
  - can work with an underlying DBMS
  - supports spatial data models, spatial ADTs and a query language from which these ADTs are callable
  - supports spatial indexing, algorithms for processing spatial operations, and domain specific rules for query optimization
- Components include
  - spatial data model, query language, query processing, file organization and indices, query optimization, etc.

1.6.1 Spatial Taxonomy, Data Models

- Spatial Taxonomy:
  - multitude of descriptions available to organize space.
  - Topology models homeomorphic relationships, e.g. overlap
  - Euclidean space models distance and direction in a plane
  - Graphs models connectivity, Shortest-Path
- Spatial data models
  - rules to identify identifiable objects and properties of space
  - Object model help manage identifiable things, e.g. mountains, cities, land-parcels etc.
  - Field model help manage continuous and amorphous phenomenon, e.g. wetlands, satellite imagery, snowfall etc.

Spatial Query Language

- Spatial query language
  - Spatial data types, e.g. point, linestring, polygon, …
  - Spatial operations, e.g. overlap, distance, nearest neighbor, …
  - Callable from a query language (e.g. SQL3) of underlying DBMS

```
SELECT S.name
FROM   Senator S
WHERE S.district.Area() > 300
```

- Standards
  - SQL3 (a.k.a. SQL 1999) is a standard for query languages
  - OGIS is a standard for spatial data types and operators
  - Both standards enjoy wide support in industry

Example

CREATE TABLE Parks (park_id SERIAL PRIMARY KEY, name VARCHAR(255) NOT NULL, boundary POLYGON NOT NULL -- Storing the park's boundary as a polygon);

Example Record:

- `park_id`: 1
- `name`: "Central Park"
- `boundary`: `POLYGON((40.7681 -73.9817, 40.7681 -73.9580, 40.8006 -73.9580, 40.8006 -73.9817, 40.7681 -73.9817))`

This `boundary` field stores a polygon representing the shape of the park as a series of coordinates that form a closed loop.

CREATE TABLE Hospitals (hospital_id SERIAL PRIMARY KEY, name VARCHAR(255) NOT NULL, location POINT NOT NULL -- Storing hospital location as a point (latitude, longitude));

Example Record:

- `hospital_id`: 1
- `name`: "City General Hospital"
- `location`: `POINT(40.712776 -74.005974)`

This `location` field stores the coordinates of the hospital as a point.

CREATE TABLE Roads (road_id SERIAL PRIMARY KEY, name VARCHAR(255) NOT NULL, path LINESTRING NOT NULL -- Storing the road path as a linestring);

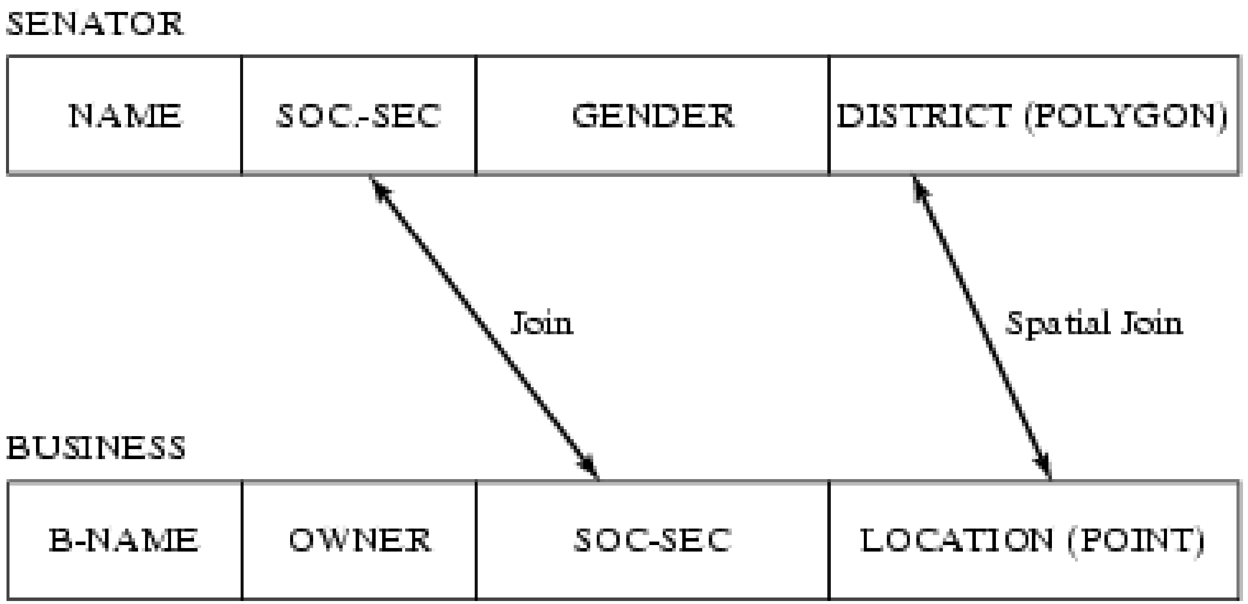
Example Record:

- `road_id`: 1
- `name`: "Main Street"
- `path`: `LINESTRING(40.712776 -74.005974, 40.715776 -74.002974, 40.718776 -73.998974)`

This `path` field stores the geometry of the road as a series of points connected by lines.

Multi-scan Query Example

- Spatial join example
  SELECT S.name FROM Senator S, Business B
  WHERE S.district.Area() > 300 AND Within(B.location, S.district)
- Non-Spatial Join example
  SELECT S.name FROM Senator S, Business B
  WHERE S.soc-sec = B.soc-sec AND S.gender = ‘Female’



SQL Query Example (assuming a PostGIS-enabled database):

Consider a spatial database of a city's infrastructure with tables for roads, parks, and buildings. The Park table contains data about each park, including its name, area (stored as polygons), and location.

"Which parks are within 2 kilometers of 'Main Street'?“

```
SELECT p.name FROM Parks p, Roads r WHERE r.name = 'Main Street' AND ST_DWithin(p.location, r.location, 2000);
```

- `ST_DWithin` is a spatial function that checks whether two geometries are within a specified distance of each other (in this case, 2000 meters or 2 kilometers).
- The query finds all parks (`p`) that are within 2 kilometers of 'Main Street' (`r`).

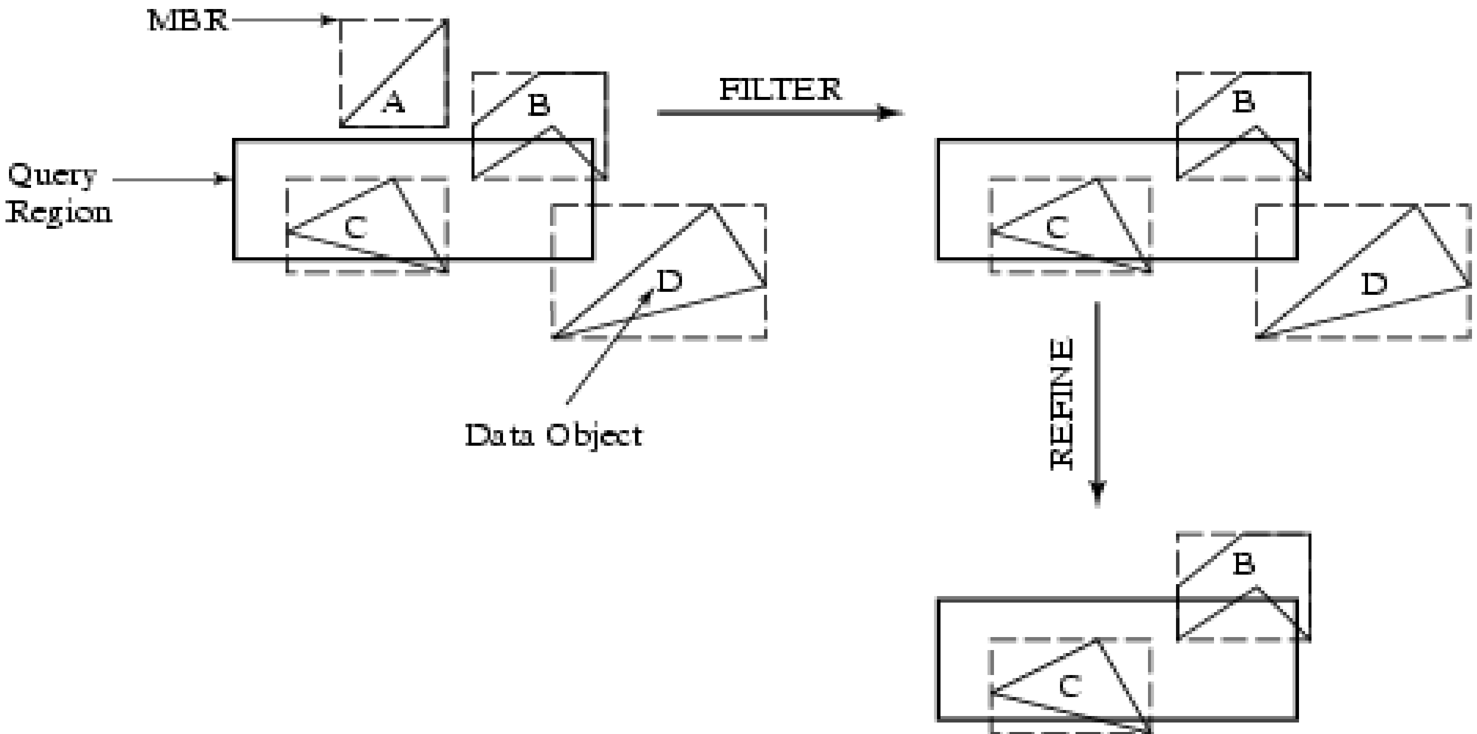
which schools are within 3 kilometers of Central Park for a school safety program.

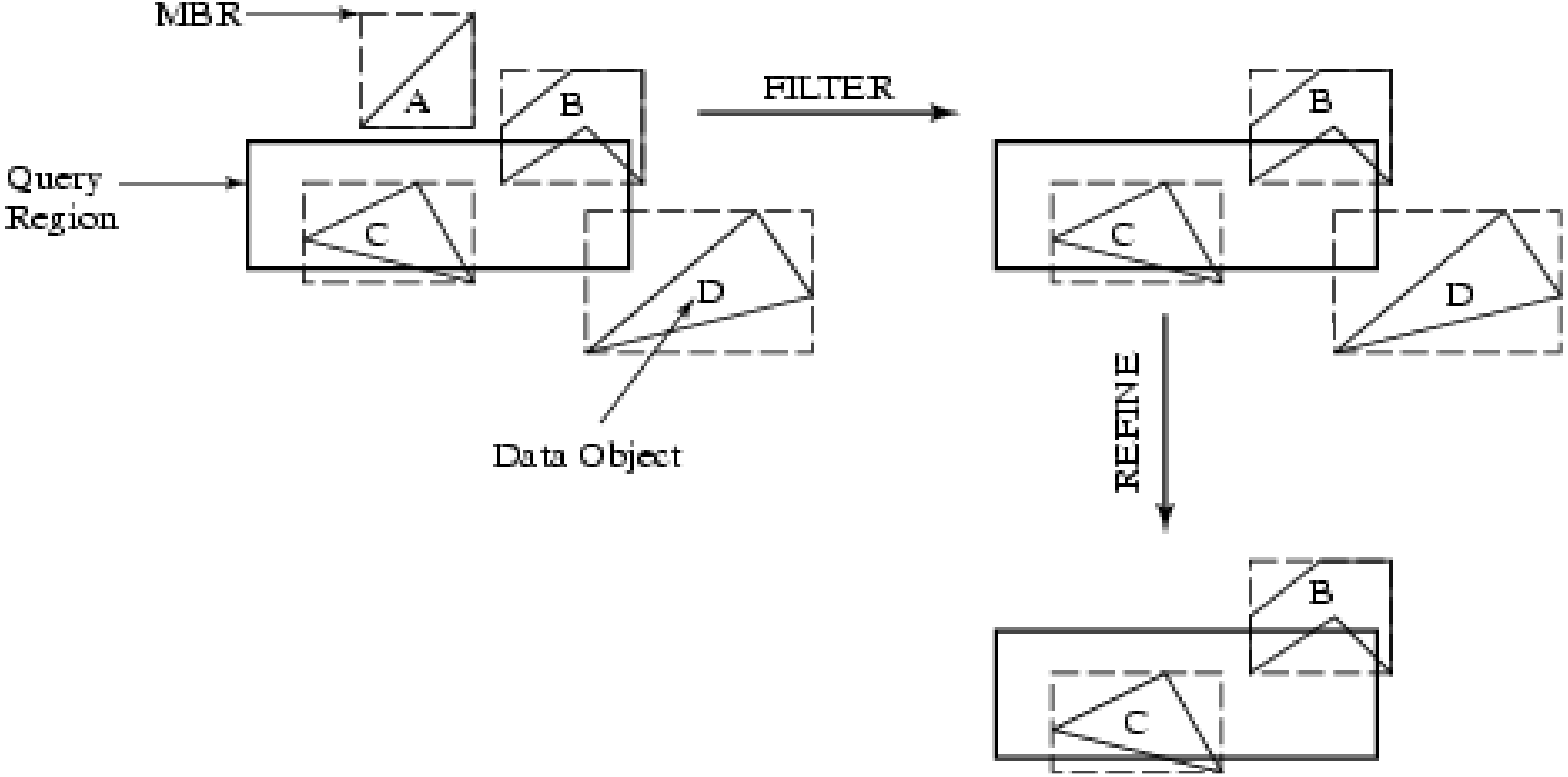
SELECT s.name FROM Schools s, Parks p WHERE p.name = 'Central Park' AND ST_DWithin(s.location, p.boundary, 3000);

- `ST_DWithin(s.location, p.boundary, 3000)` checks whether the schools (`s.location`) are within 3000 meters (3 kilometers) of Central Park's boundary (`p.boundary`).
- This query returns all schools within the specified distance of Central Park.

Query Processing

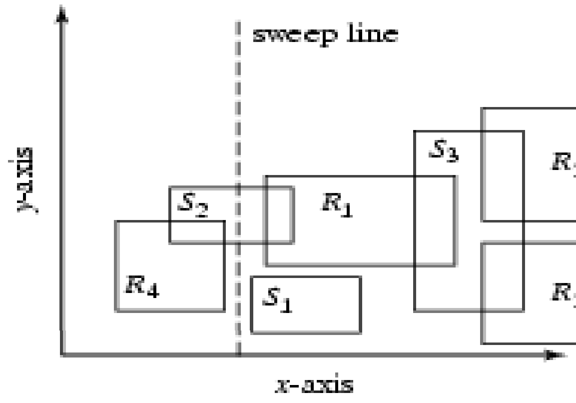
- Efficient algorithms to answer spatial queries
- Common Strategy - filter and refine
  - Filter Step:Query Region overlaps with minimum bounding rectangle (MBR) of B,C and D
  - Refine Step: Query Region overlaps with B and C



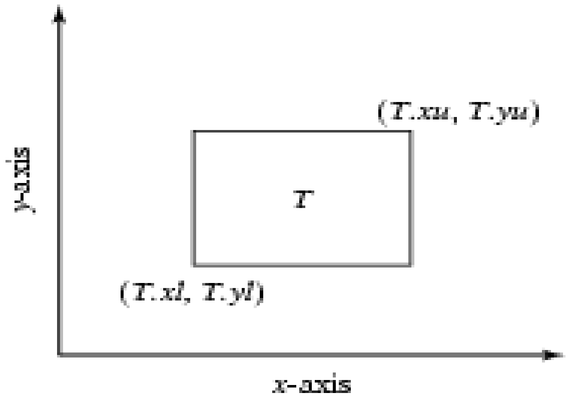

MBR
A
B
FILTER
Query
Region
C
D
Data Object
B
C
D
REFINE
B
C

Query Processing of Join Queries

- Example - Determining pairs of intersecting rectangles
  - (a):Two sets R and S of rectangles, (b): A rectangle with 2 opposite corners marked, (c): Rectangles sorted by smallest X coordinate value
  - Plane sweep filter identifies 5 pairs out of 12 for refinement step
  - Details of plane sweep algorithm on page 15



(a)



(b)

R_4	S_2	S_1	R_1	S_3	R_2	R_3

(c)

1.6.4 File Organization and Indices

- A difference between GIS and SDBMS assumptions
  - GIS algorithms: dataset is loaded in main memory (Fig. 1.10(a))
  - SDBMS: dataset is on secondary storage e.g disk (Fig. 1.10(b))
  - SDBMS uses space filling curves and spatial indices
    - to efficiently search disk resident large spatial datasets

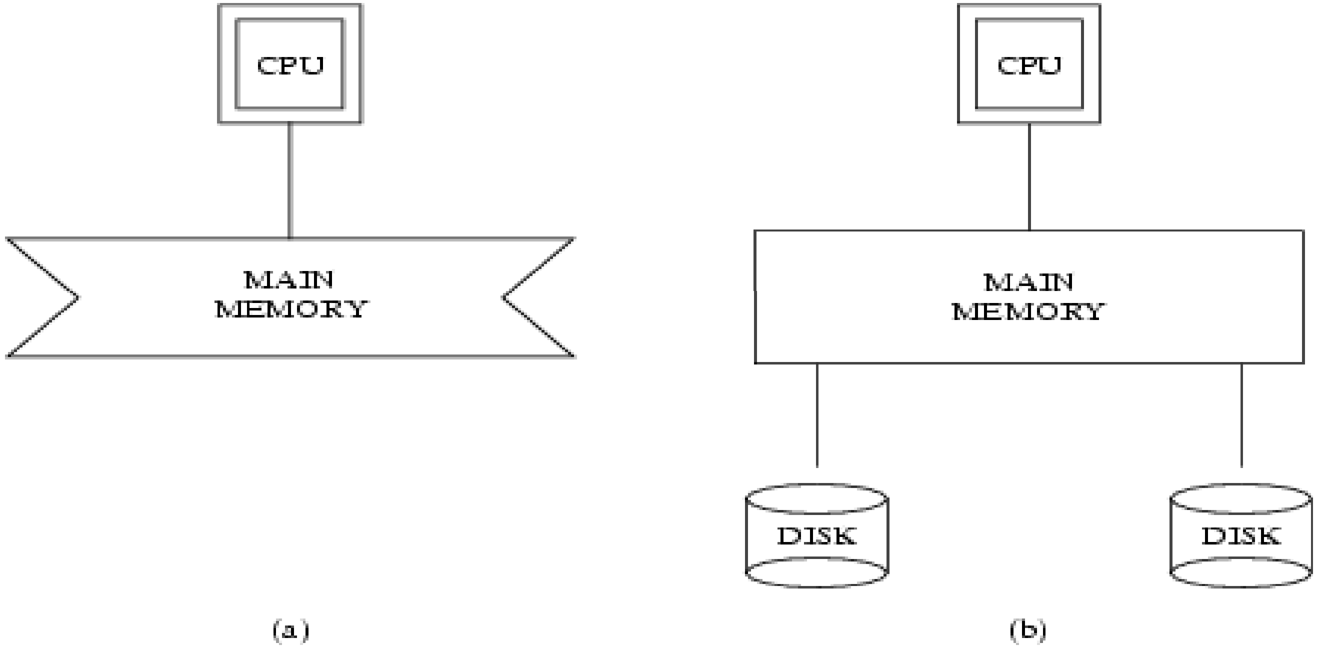


Fig 1.10

Organizing spatial data with space filling curves

- Issue:
  - Sorting is not naturally defined on spatial data
  - Many efficient search methods are based on sorting datasets
- Space filling curves
  - Impose an ordering on the locations in a multi-dimensional space
  - Examples: row-order (Fig. 1.11(a), z-order (Fig 1.11(b))
  - Allow use of traditional efficient search methods on spatial data

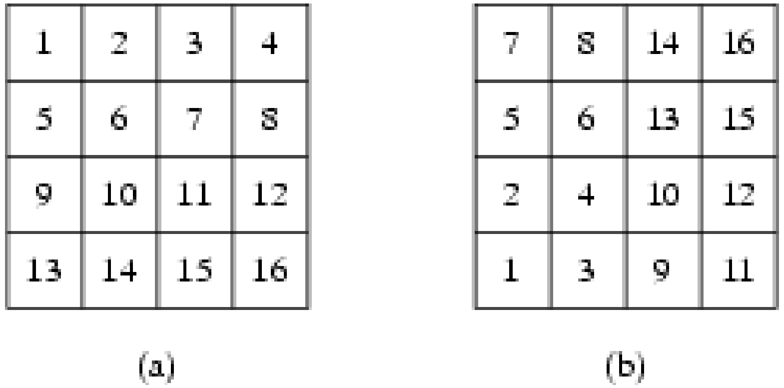


Fig 1.11

Spatial Indexing: Search Data-Structures

- Choice for spatial indexing:
  - B-tree is a hierarchical collection of ranges of linear keys, e.g. numbers
  - B-tree index is used for efficient search of traditional data
  - B-tree can be used with space filling curve on spatial data
  - R-tree provides better search performance yet!
  - R-tree is a hierarchical collection of rectangles
  - More details in chapter 4

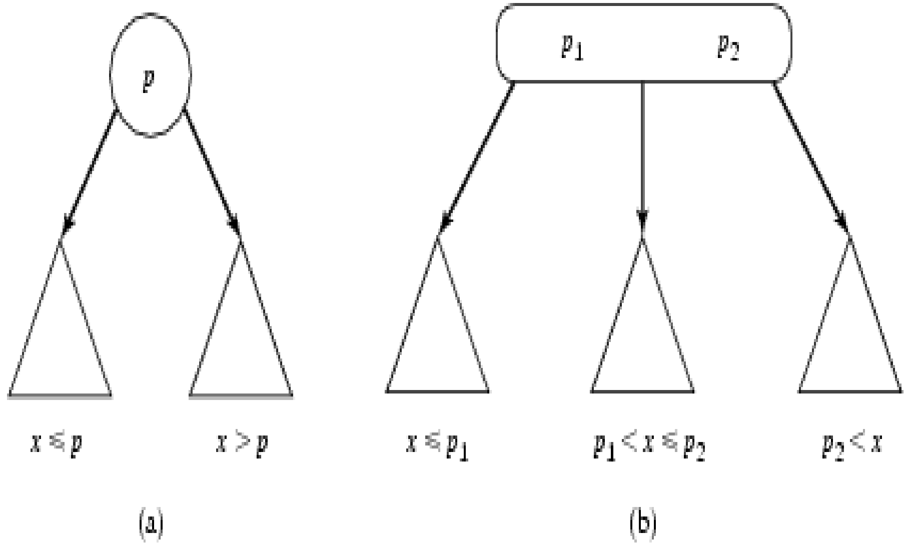


Fig 1.12: B-tree

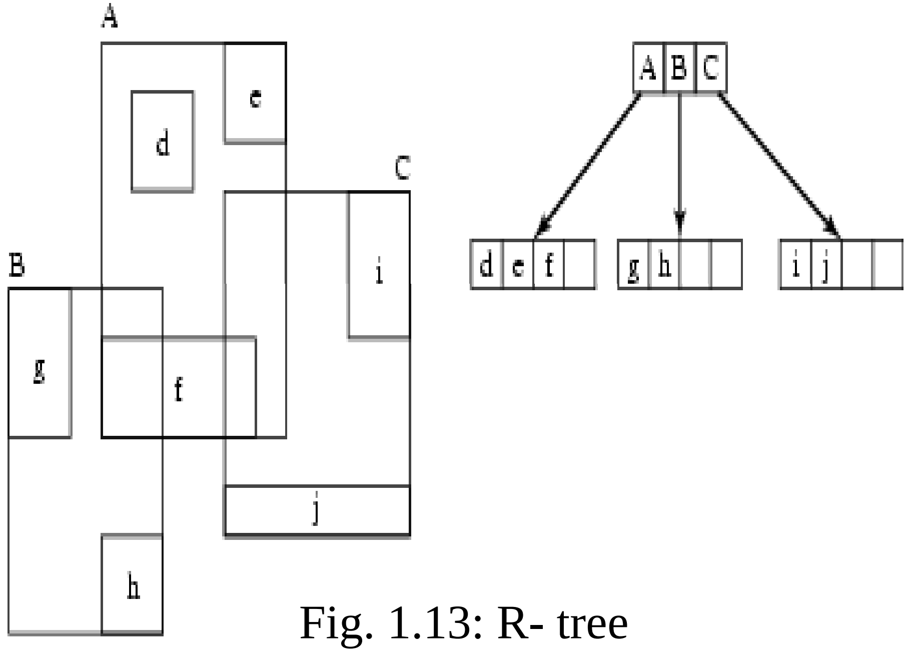


Fig. 1.13: R- tree

Query Optimization

- Query Optimization
  - A spatial operation can be processed using different strategies
  - Computation cost of each strategy depends on many parameters
  - Query optimization is the process of
    - ordering operations in a query and
    - selecting efficient strategy for each operation
    - based on the details of a given dataset
- Example Query:

  SELECT S.name FROM Senator S, Business B
  WHERE S.soc-sec = B.soc-sec AND S.gender = 'Female'
- Optimization decision examples
  - Process (S.gender = 'Female') before (S.soc-sec = B.soc-sec)
  - Do not use index for processing (S.gender = 'Female')

Data Mining

- Analysis of spatial data is of many types
    - Deductive Querying, e.g. searching, sorting, overlays
    - Inductive Mining, e.g. statistics, correlation, clustering,classification, …
- Data mining is a systematic and semi-automated search for interesting non-trivial patterns in large spatial databases

- Example applications include
    - Infer land-use classification from satellite imagery
    - Identify cancer clusters and geographic factors with high correlation
    - Identify crime hotspots to assign police patrols and social workers

Multimedia Database Concepts

- Multimedia databases allow users to store and query images, video, audio, and documents
- Content-based retrieval
  - Automatic analysis
  - Manual identification
  - Color often used in content-based image retrieval
  - Texture and shape
- Object recognition
  - Scale-invariant feature transform (SIFT) approach

Multimedia Database Concepts (cont’d.)

- Semantic tagging of images
  - User-supplied tags
  - Automated generation of image tags
  - Web Ontology Language (OWL) provides concept hierarchy
- Analysis of audio data sources
  - Text-based indexing
  - Content-based indexing