Chapter 1: Introduction to Spatial Databases

Today’s Goals
- Overview of spatial databases
- Answer any course related questions
Learning Objectives

Learning Objectives (LO)

- LO1: Understand the value of SDBMS
  - Application domains
  - Users
  - How is different from a DBMS?
- LO2: Understand the concept of spatial databases
- LO3: Learn about the Components of SDBMS

Mapping Sections to learning objectives

- LO1 - 1.1, 1.2, 1.4
- LO2 - 1.3, 1.5
- LO3 - 1.6

Value of SDBMS

Traditional (non-spatial) database management systems provide:
- Persistence across failures
- Allows concurrent access to data
- Scalability to search queries on very large datasets which do not fit inside main memory of computers
- Efficient for non-spatial queries, but not for spatial queries

Non-spatial queries:
- List the names of all bookstores with more than ten thousand titles.
- List the names of ten customers, in terms of sales, in the year 2015

Spatial Queries:
- List the names of all bookstores within ten miles of Auburn.
- List all customers who live in Alabama and its adjoining states.
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 satellite images - terabytes of data per day
  - Weather and Climate Data
  - Rivers, Farms, ecological impact
  - Medical Imaging
- Exercise: Identify spatial and non-spatial data items in
  - A phone book (yellow pages)
  - A cookbook with recipes

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?
  - Corporate supply manger: Given trends about our future customer profile, which are the best places to build retail stores?
  - Sports: Which seats in a baseball stadium provide best view of pitcher and hitter? Where should TV camera be mounted?
  - Mobile phone user: Where is the nearest gas station?
- Exercise: List two ways you have used spatial data. Which software did you use to manipulate spatial data?
Learning Objectives

Learning Objectives (LO)

LO1: Understand the value of SDBMS
LO2: Understand the concept of spatial databases
  • What is a SDBMS?
  • How is it different from a GIS?
LO3: Learn about the Components of SDBMS

Sections for LO2

Section 1.5 provides an example SDBMS
Section 1.1 and 1.3 compare SDBMS with DBMS and GIS

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 10g 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 (Figure 1.3)
- Question: Is Polyline datatype supported in DBMS?

<table>
<thead>
<tr>
<th>Name</th>
<th>Area</th>
<th>Population</th>
<th>Boundary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1050</td>
<td>1</td>
<td>1839</td>
<td>Polyline((0.0),(0.1),(1.0))</td>
</tr>
</tbody>
</table>
**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 1.4 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 square represented as 16 rows across 3 tables
  - Simple spatial operators, e.g. area(), require joining tables
  - Tedious and computationally inefficient

---

**Mapping “census table” into a Relational Database**

<table>
<thead>
<tr>
<th>Census_blocks</th>
<th>Polygon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Area</td>
</tr>
<tr>
<td>340</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>edge-name</td>
</tr>
<tr>
<td>A</td>
</tr>
<tr>
<td>A</td>
</tr>
<tr>
<td>B</td>
</tr>
<tr>
<td>B</td>
</tr>
<tr>
<td>C</td>
</tr>
<tr>
<td>C</td>
</tr>
<tr>
<td>D</td>
</tr>
<tr>
<td>D</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Point</th>
</tr>
</thead>
<tbody>
<tr>
<td>endpoint</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
</tbody>
</table>

Fig 1.4
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
  - **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 focuses on:
  - Efficient storage, querying, sharing of large spatial datasets
  - 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 SDBMS to answer the following:
  - How many neighboring countries does USA have?
  - Which country has the highest number of neighbors?
Three meanings of the acronym GIS

- Geographic Information Services
  - Web-sites and service centers for casual users, e.g. travelers
  - Example: Service (e.g. AAA, Google maps) for route planning

- Geographic Information Systems
  - Software for professional users, e.g. cartographers
  - Example: ESRI Arc/View software

- Geographic Information Science
  - Concepts, frameworks, theories to formalize use and development of geographic information systems and services
  - Example: design spatial data types and operations for querying

Learning Objectives

- Learning Objectives (LO)
  - LO1: Understand the value of SDBMS
  - LO2: Understand the concept of spatial databases
  - LO3: Learn about the Components of SDBMS
    - Architecture choices
    - SDBMS components:
      - data model, query languages,
      - query processing and optimization
      - File organization and indices
      - Data Mining

- Chapter Sections
  - 1.5 second half
  - 1.6 – entire section
**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.
- Figure 1.6 shows these components
- We discuss each component briefly in chapter 1.6 and in more detail in later chapters.

**Three Layer Architecture**
1.6.1 Spatial Taxonomy, Data Models

- **Spatial Taxonomy:**
  - Topology models homomorphic relationships, e.g. overlap
  - Euclidean space models distance and direction in a plane
  - Graphs models connectivity, Shortest-Path

<table>
<thead>
<tr>
<th>Set theory based</th>
<th>Union, Intersection, Containment,</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topological</td>
<td>Touches, Disjoint, Overlap, etc.</td>
</tr>
<tr>
<td>Directional</td>
<td>East, North-West, etc.</td>
</tr>
<tr>
<td>Metric</td>
<td>Distance</td>
</tr>
</tbody>
</table>

- **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.

- **Spatial data models**
  - rules to identify objects and properties of space
  - Object models help manage identifiable things, e.g. mountains, cities, land-parcels, etc.
  - Field models help manage continuous and amorphous phenomenon, e.g. wetlands, snowfall, etc.
**Model Spatial Information**

- Example: Forest stands
  - (a) forest stand map
  - (b) Object view has 3 polygons
  - (c) Field view has a function

### Object Viewpoint of Forest Stands

<table>
<thead>
<tr>
<th>Area-ID</th>
<th>Dominant Tree Species</th>
<th>Area/Boundary</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS1</td>
<td>Pine</td>
<td>[(0,2),(4,4),(4,0)]</td>
</tr>
<tr>
<td>FS2</td>
<td>Fir</td>
<td>[(0,0),(2,2),(2,0)]</td>
</tr>
<tr>
<td>FS3</td>
<td>Oak</td>
<td>[(2,0),(4,4),(2,2)]</td>
</tr>
</tbody>
</table>

### Field Viewpoint of Forest Stands

**f(x,y) =**

- "Pine": $0 \leq x \leq 4 \land 0 \leq y \leq 4$
- "Fir": $0 \leq x \leq 2 \land 0 \leq y \leq 2$
- "Oak": $2 \leq x \leq 4 \land 0 \leq y \leq 2$

---

**1.6.2 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. SQL 2011) of underlying DBMS

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

- Standards
  - SQL 2011 is a standard for query languages
  - OGIS (Open GIS) is a standard for spatial data types and operators
  - Both standards enjoy wide support in industry
  - More details in chapters 2 and 3
**Multi-scan Query Example**

- Non-Spatial Join example
  “Find the name of all female senators who own a business”

- Spatial join example
  “Find all senators who serve a district of area greater than 300 square miles and who own a business within the district”

**1.6.3 Query Processing**

- Efficient algorithms to answer spatial queries
- Common Strategy - filter and refine
  - **Filter Step**: Query Region overlaps with MBRs of B, C, and D
  - **Refine Step**: Query Region overlaps with B and C
**Query Processing of Spatial 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

![Diagram of sweep line and rectangles]

**Query Processing of Join Queries (Cont.)**

1. Move a sweep line, for example, a line perpendicular to the x-axis, from left to right, and stop at the first entry in $R \cup S$. This is the rectangle $T$ with the smallest $T.xl$ value. In our example (see Figure 1.9a) this is the rectangle $R4$.

2. Search through the sorted rectangles of $S$ until arriving at the first rectangle $S_f$ such that $S_f.xl > T.xu$. Clearly, the relation $[T.xl, T.xu] \cap [S_j.xl, S_j.xu]$ holds (nonempty) for all $1 \leq j < f$. In our example $S_f$ is $S_1$. Thus $S_2$ is a candidate rectangle that may overlap $R_4$. This will be confirmed in the next step.

3. If the relation $[T.yl, T.yu] \cap [S_j.yl, S_j.yu]$ holds for any $1 \leq j \leq f$, then the rectangle $S_j$ intersects $T$. Thus this step confirms that $R_4$ and $S_2$ indeed overlap and $<R_4, S_2>$ are part of the join result. Record all of this information and remove the rectangle $T$ from the set $R \cup S$. $R_4$ is removed from the set $R \cup S$ because it cannot participate in any more pairs in the resulting set.

4. Move the sweep line across the set $R \cup S$ until it reaches the next rectangle entry. This is rectangle $S_2$ in our example. Now proceed as in steps 2 and 3.
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](image)

Organizing spatial data with space filling curves

- Issue:
  - Sorting is not naturally defined on spatial data
  - Many efficient search methods are based on sorted 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](image)
**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

![B-tree and R-tree diagrams](image)

### 1.6.5 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
    - selecting efficient strategy for each operation based on the details of a given dataset

- Example Query:
  "Find the name of all female senators who own a business"

  ```sql
  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 )
1.7 Summary

- SDBMS is valuable to many important applications
- SDBMS is a software module
  - works with an underlying DBMS
  - provides spatial ADTs callable from a query language
  - provides methods for efficient processing of spatial queries
- Components of SDBMS include
  - spatial data model, spatial data types and operators,
  - spatial query language, processing and optimization
  - spatial data mining
- SDBMS is used to store, query and share spatial data for GIS as well as other applications