SGG 4653
Advance Database System

An Overview of Database System Architecture and Spatial Database

Outline

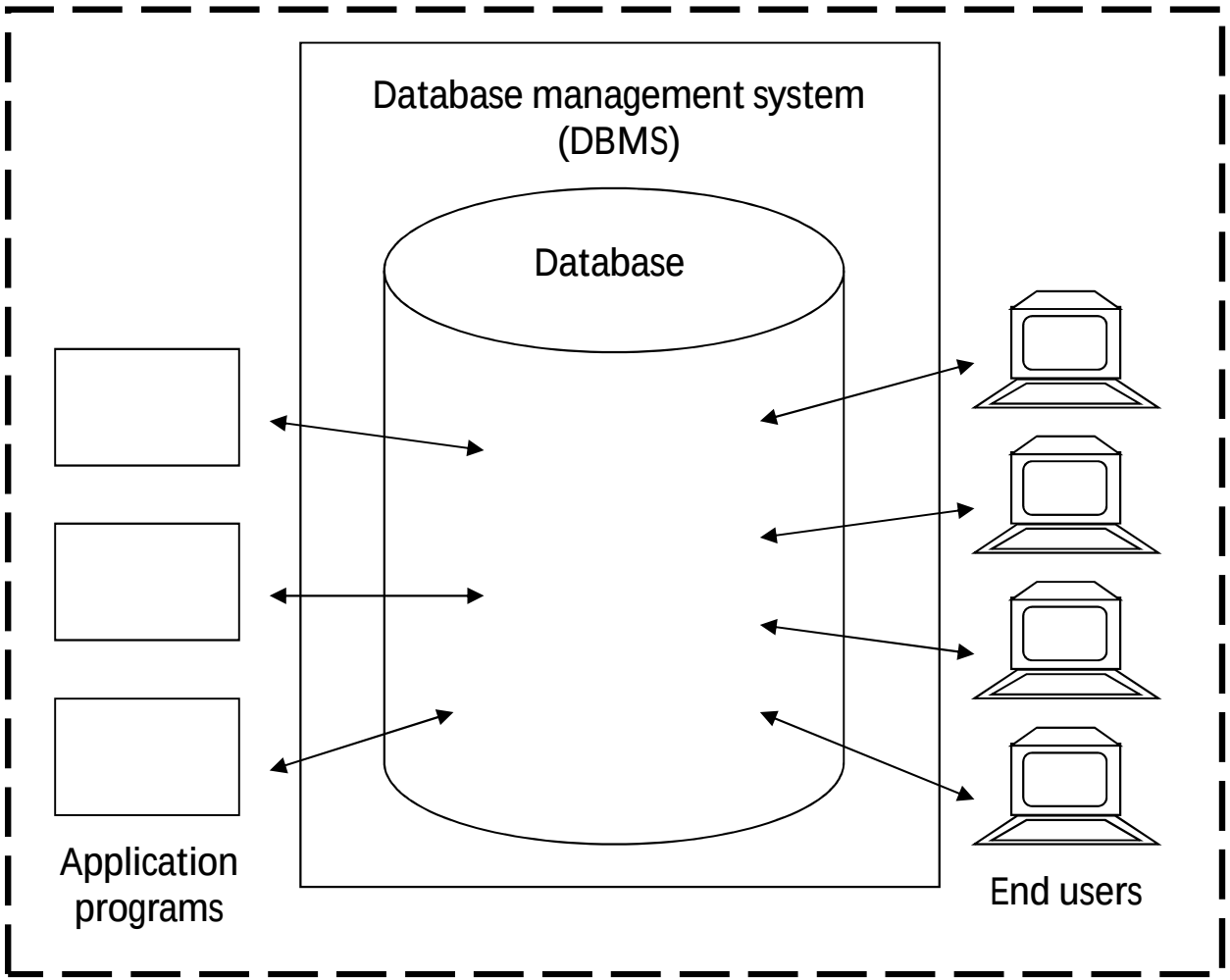
- Introduction
- The Database Management System (DBMS)
- Client Server Architecture
- Distributed DBMS
- Distributed Processing
- Spatial Database:
  - Introduction
  - Spatial Data model
  - Spatial Data Structure
  - Spatial Data Query Language

Introduction

- Database system - A computerized record keeping system
  - Adding new, empty files to the database
  - Inserting data into existing files
  - Retrieving data from existing
  - Changing data in existing files
  - Deleting data from existing files
  - Removing existing files from the database

- Database system involves four major components: data, hardware, software and users.

Introduction



Simplified picture of a database system

Introduction

- Database systems:
  - multi-user: many user can access the database simultaneously.
  - single-user: one user can access the database at any given time.

- DATA (the data in the database)
  - In the “large” environment **à** integrated & shared.
  - In “small” environment **à** data integration.

- HARDWARE
  - The secondary storage volume to stored data together with I/O devices, device controller, I/O channel.
  - The hardware processor and associated main memory that used to support the execution of the database system software.

Introduction

- SOFTWARE
  - DBMS
  - utilities
  - application development tools
  - design aids
  - report writers
  - the transaction manager

Introduction

- USERS
  - first, there are application programmers, responsible for writing database application programs in some programming language.
  - The second class of user, then, the end users, who interact with the system from on line workstation or terminals.
  - The third class of user is the database administrator or DBA.

Introduction

- The advantages of a database system over traditional, paper-based methods of record-keeping are:
  - Compactness.
  - Speed.
  - Less hardworking.
  - Currentness.
- Benefits of using Database:
  - The data can be shared.
  - Redundancy can be reduced.
  - Inconsistency can be avoided, thus integrity can be maintained.
  - Transaction support can be provided.
  - Standards & security can be enforced.

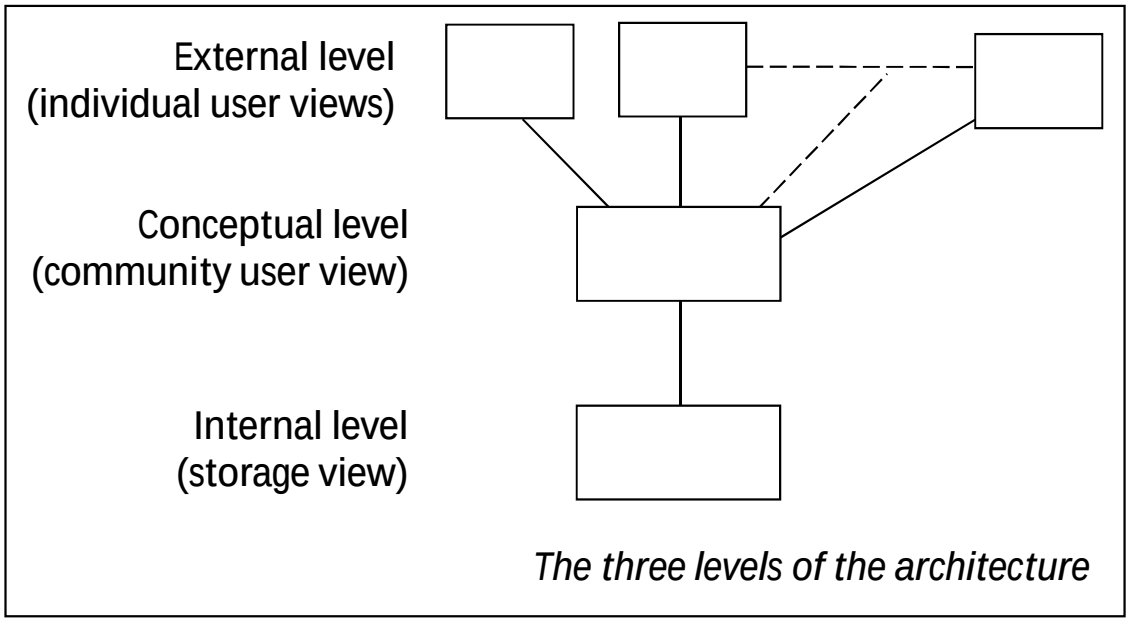
The Database Management System (DBMS)

- DBMS is the software that handles all access the database.

- Those functions will include support for at least all of the following:
  - *Data definition*
  - *Data manipulation*
  - *Optimization and execution*
  - *Data security and integrity*
  - *Data recovery and concurrency*
  - *Data dictionary*
  - *Performance*

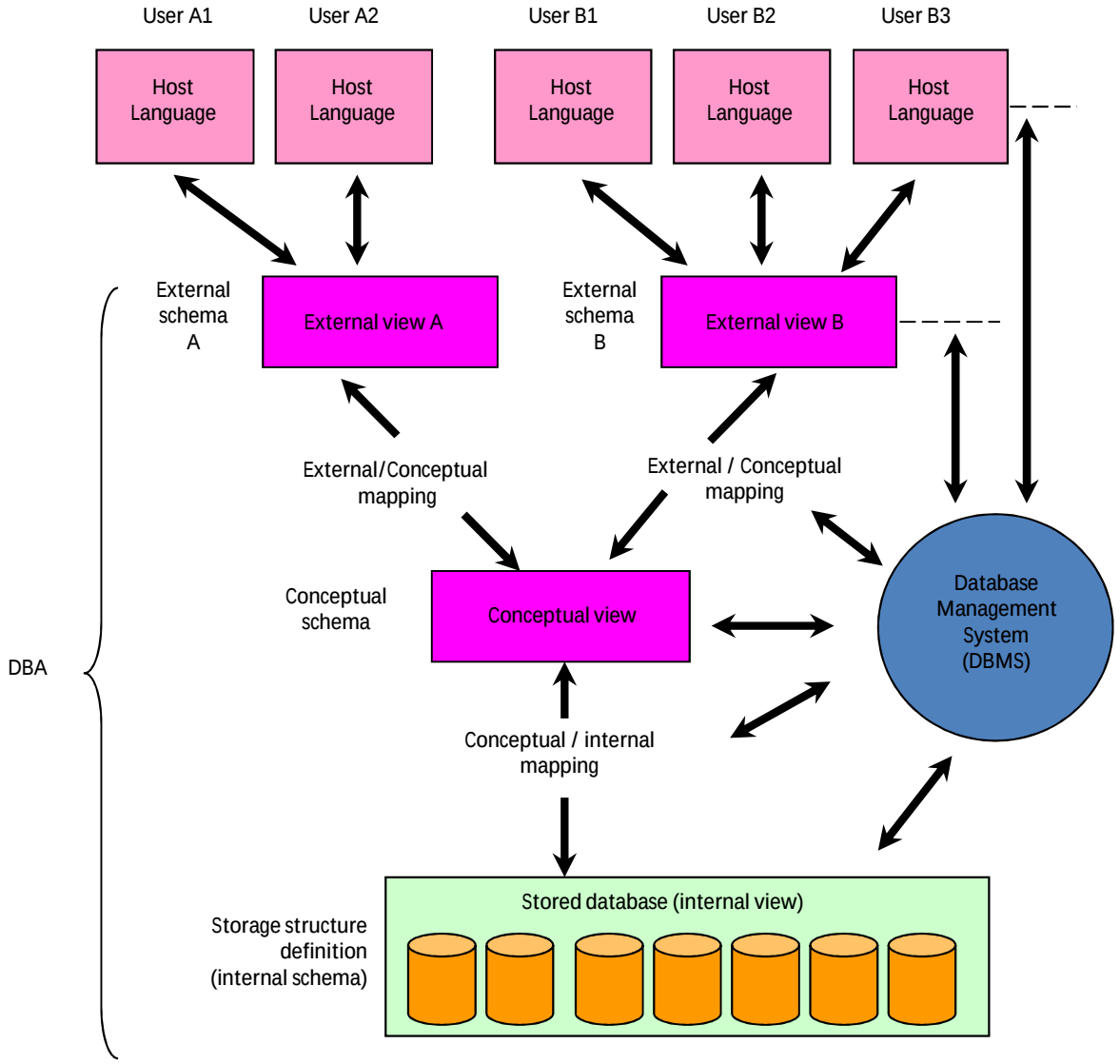
The Database Management System (DBMS)

- Divided into three levels
  - The internal level,
  - The external level,
  - The conceptual level.



The three levels of the architecture

The Database Management System (DBMS)



Detailed system architecture.

The Database Management System (DBMS)

- The external level (also known as the user logical level)
  - The way the data is seen by individual users or the content of the database as seen by some particular user.
  - user can be either an application programmer or an end user of any degree of sophistication
    - DSL (data sublanguage) **à** subset of the total language that is concerned specifically with database objects
    - Host language – providing various non-database facilities, local variable, computational operation, branching logic.

The Database Management System (DBMS)

- The conceptual level (also known as the community logical level):
  - Is a representation of the entire information content of the database.
  - is intended to be a view of the data “as it is”.
  - is defined by means of the conceptual schema.

The Database Management System (DBMS)

- The Internal Level (also known as the physical level)
  - The internal view is a low-level representation of the entire database.
  - Is described by means of the internal schema.
  - It is the way the data is physically stored.
  - Defines the various stored record, what indexes exit, how stored fields are represented, what physical sequence the stored records are in.

The Database Management System (DBMS)

- The data administrator (DA) is the person who makes the strategic and policy decisions regarding the data of the enterprise.

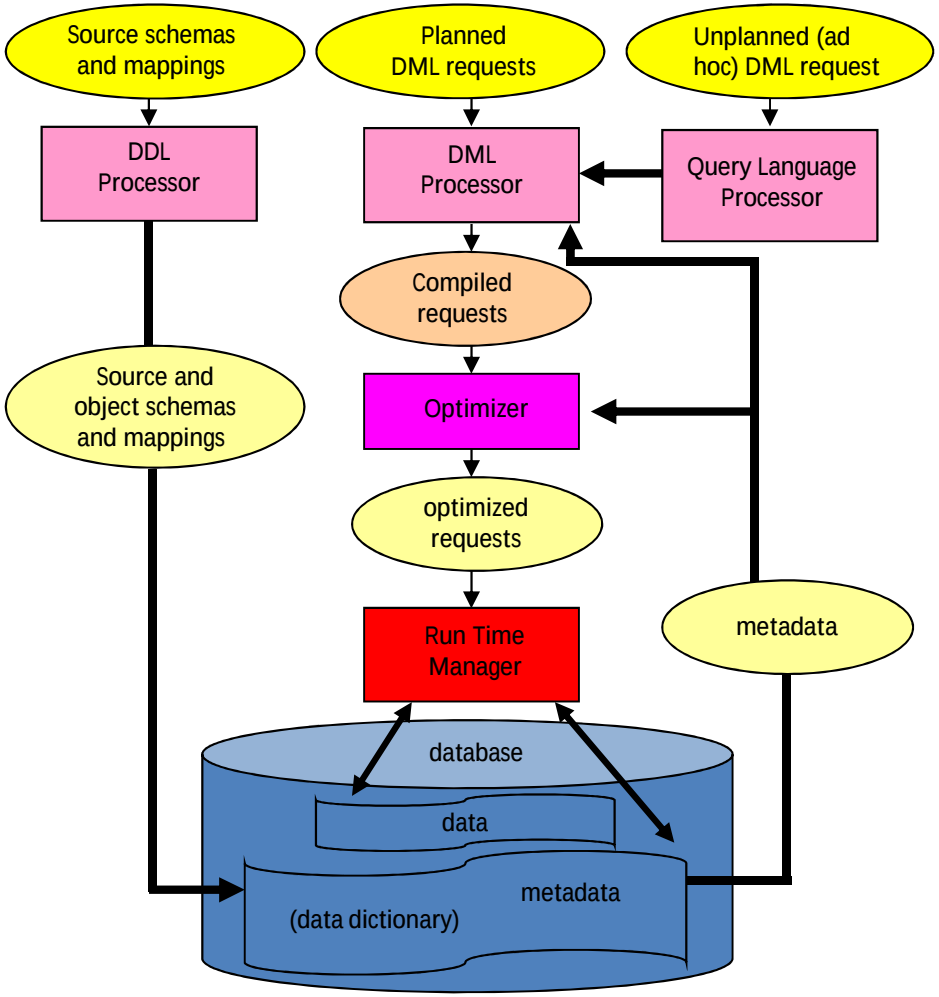
- The database administrator (DBA) is the person who provides the necessary technical support for implementing those decisions.

The Database Management System (DBMS)

- DBA is responsible for the overall control of the system at a technical level. Some of the task:
  - *Defining the internal & conceptual schema*
  - *Liaising with users*
  - *Defining security and integrity constraints*
  - *Defining dump and reload policies*
  - *Monitoring performance and responding to changing requirements*

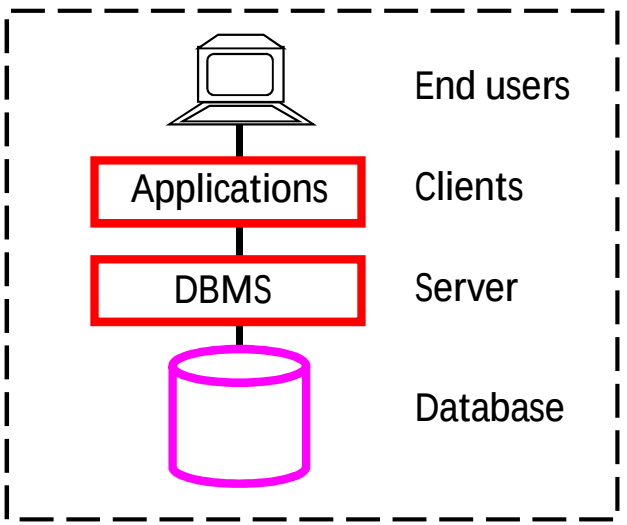
The Database Management System (DBMS)

- Major DBMS Functions and Components:



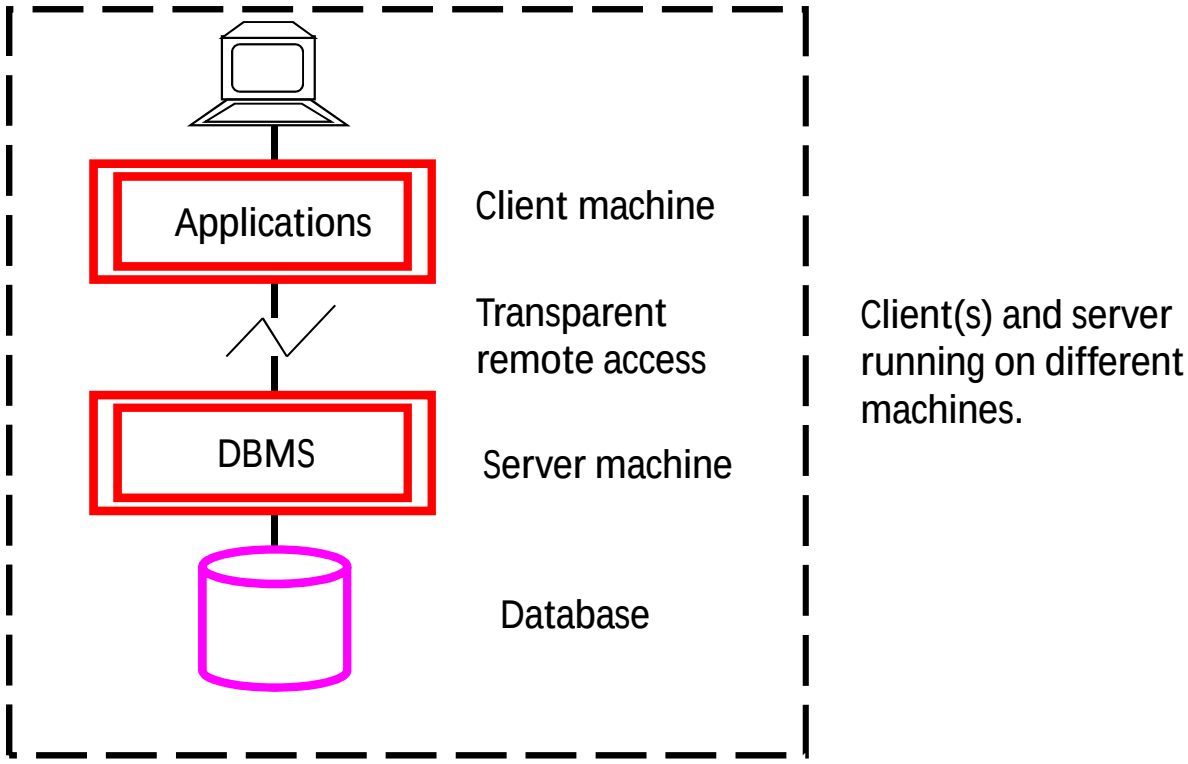
Client Server Architecture

- A database system can be regarded as having a very simple two-part structure:
  - consisting of a server (also called the backend) and set of clients (also called as frontends).
    - The server is just the DBMS itself. It supports all of the basic DBMS functions
    - The clients are the various applications that run on top of the DBMS



Distributed Processing

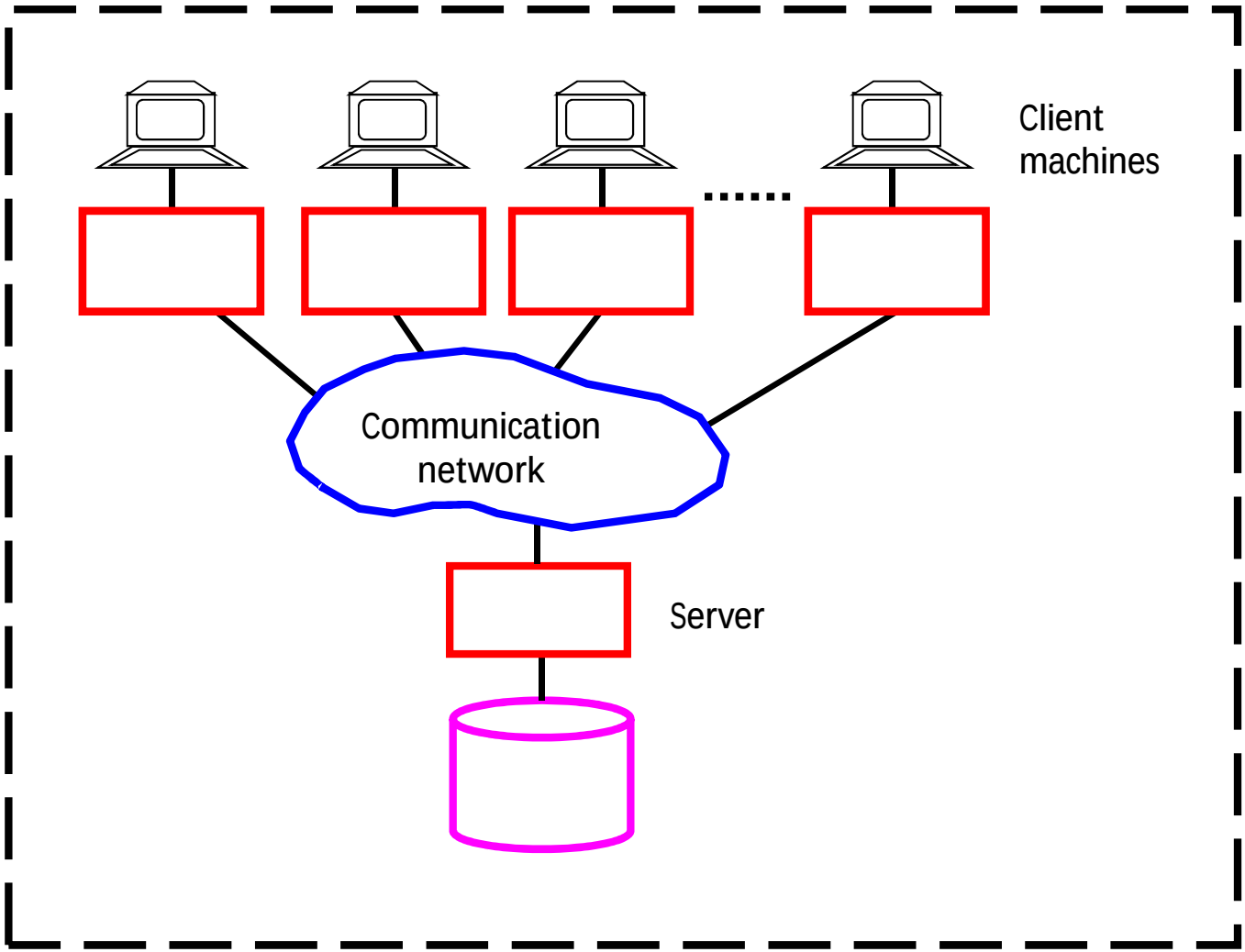
- Distributed processing - the machines can be connected together into a communications network such as the internet, a single data-processing task can span several machines in the network.



Distributed Processing

- Several different machine might be able to access the same server machine.
- A single database might be shared across several distinct client systems.
- The client(s) and server running on separate machines.

Distributed Processing



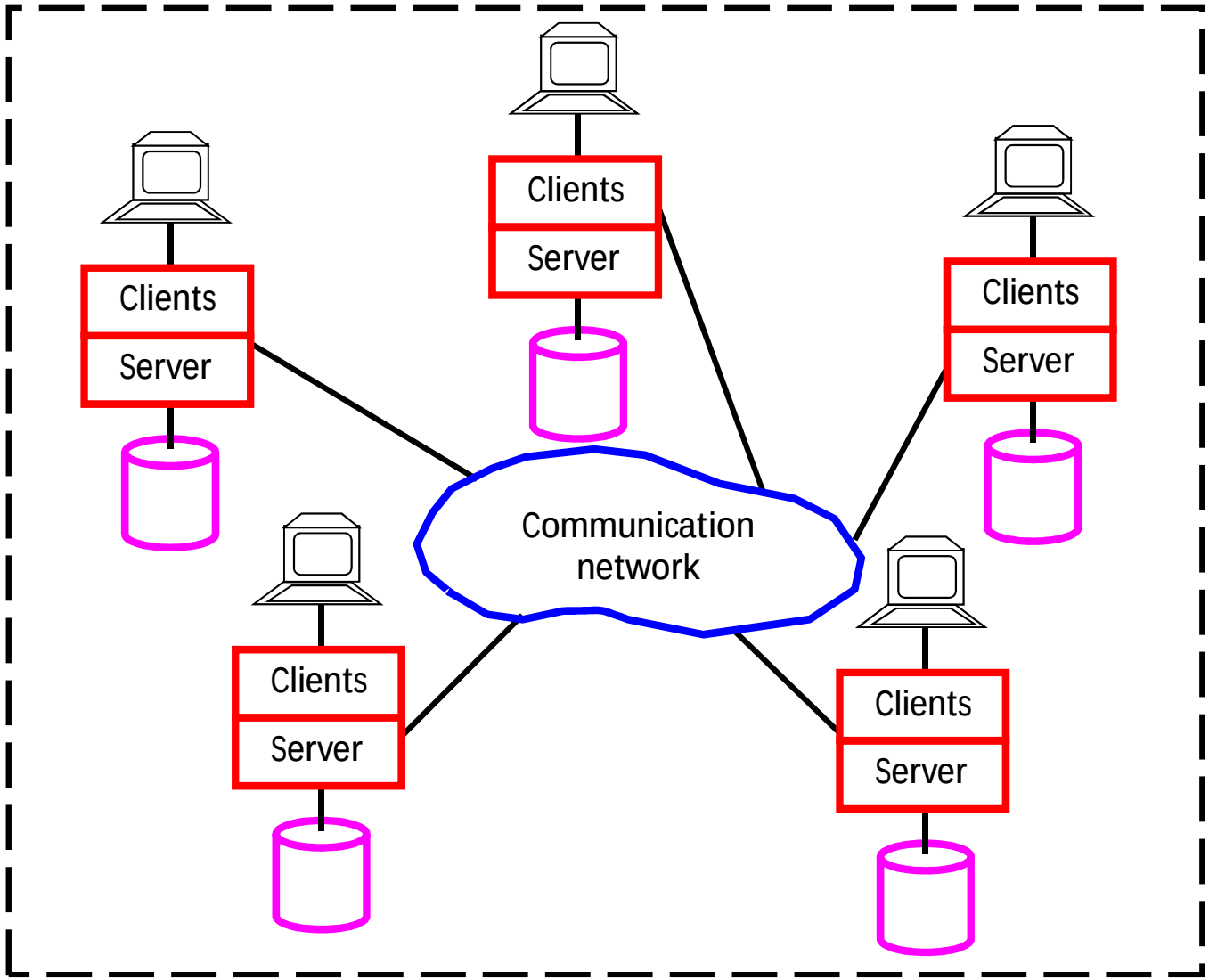
One server machine, many client machines.

Distributed Processing

- For a single enterprise – (for example: a bank) the data for one portion of the enterprise is stored on the computer and the data for another portion of the enterprise is stored on another (user at one branch office).

- The client machine might have stored data of their own and the server machine might have applications of its own.

Distributed Processing



Each machine runs both client(s) and server.

Spatial Database

Spatial (geometric, geographic) database system: handling objects in space that have identity and well-defined extents, locations, topology and relationships.

Problems of modeling spatial data in relational DBMS:

- Implementation of spatial data types are quite complex.
- Data cannot be naturally mapped onto a relational database.
- Need a database to handle spatial information in order to reduce semantic gap between user's view of spatial data and the database implementation.

Spatial Database

Solutions:

- Relational database which incorporates abstract data models and other principles of object oriented design.
- Object relational DBMS.
- Design a specialized object relational DBMS to meet requirements of spatial data.

Spatial Database

How is a Spatial database 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, classification
  - Location analysis: Buffer, overlay, intersect
  - 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

Spatial Database

How is a Spatial database different from a GIS?

- GIS uses Spatial database:
  - to store, search, query, share large spatial data sets.

Spatial Database

- Spatial DBMS focuses 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.

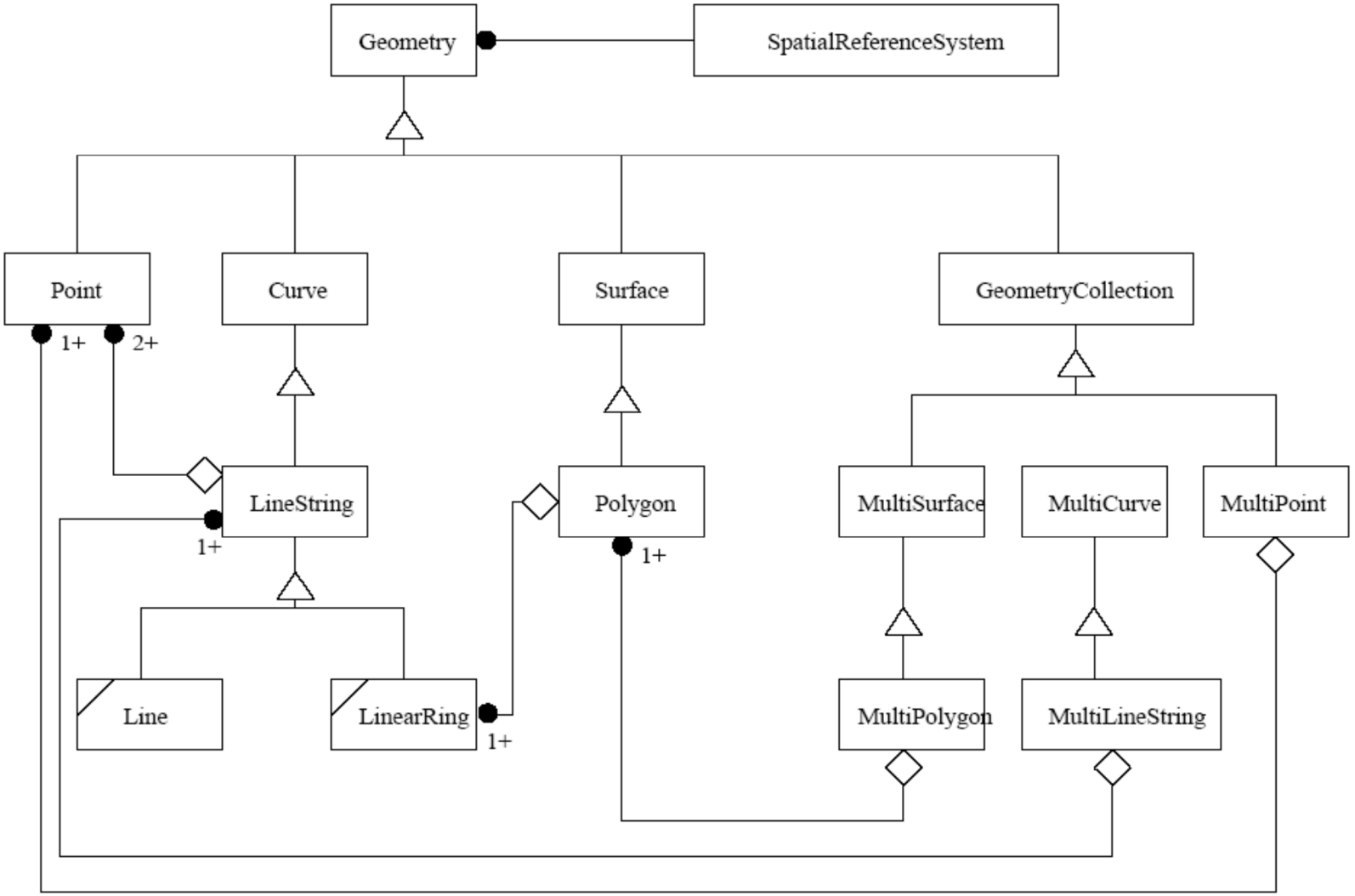
- Spatial DBMS may be used by applications other than GIS
  - Astronomy, Multimedia information systems, CAD, etc.

Spatial Database

- Spatial DBMS is a software module that
  - can work with an underlying DBMS (ORDMBS, OODBMS)
  - supports spatial data models, spatial abstract data types (ADT) and a query language to call these ADT
  - supports spatial indexing, algorithms for processing spatial operations, and domain specific rules for query optimization

- Components
  - spatial data model
  - query language, query processing, & query optimization
  - file organization and indices
  - etc.

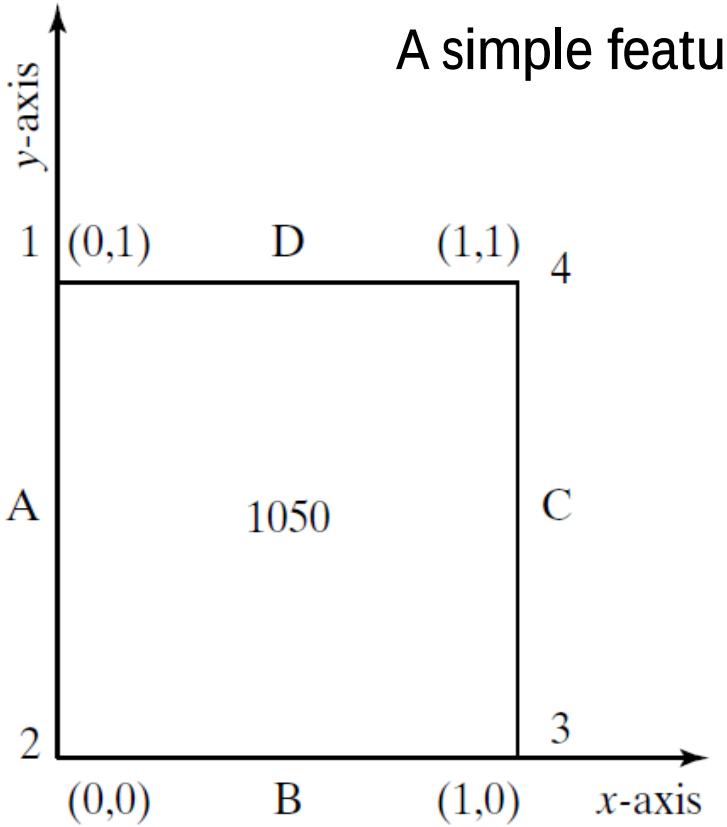
Spatial Database



Source: OGC Simple Feature specification

Spatial Database

A simple feature (census blocks) with ID = 1050



Census_blocks

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

Spatial 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

Four tables required in a relational database with overlapping attributes to accommodates the polyline data type.

Spatial Query Language

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

```
SELECT D.name
FROM District D
WHERE D.Area() > 300
```

Spatial Query Language

- Querying via SQL: “Return the POIs at a specified radius (15km)”

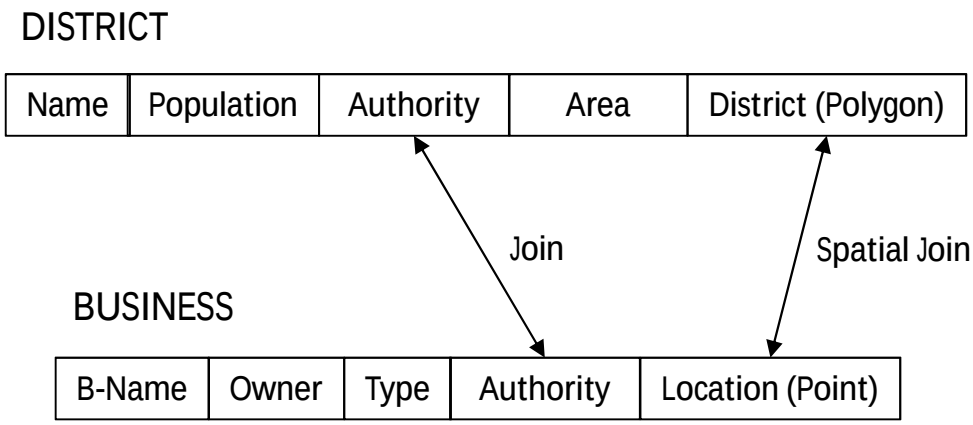
```
SELECT address, name, lat, lng, (6371 * acos (cos (radians('15'))
        * cos(radians( lat )) * cos(radians(lng) - radians('15')) +
        sin(radians('15')) * sin(radians(lat)))) AS distance
FROM POI_markers
HAVING distance < '15'
ORDER BY distance
LIMIT 0,20
```

- The query involves retrieving set of coordinates from table POI_markers (POIs data) that are within 15km radius to a specific user define coordinates (lat, lng).

Spatial Query Language

- Spatial Join example:

```
SELECT D.name
FROM District D, Business B
WHERE D.Area() > 300
AND Within (B.location, D.district)
```



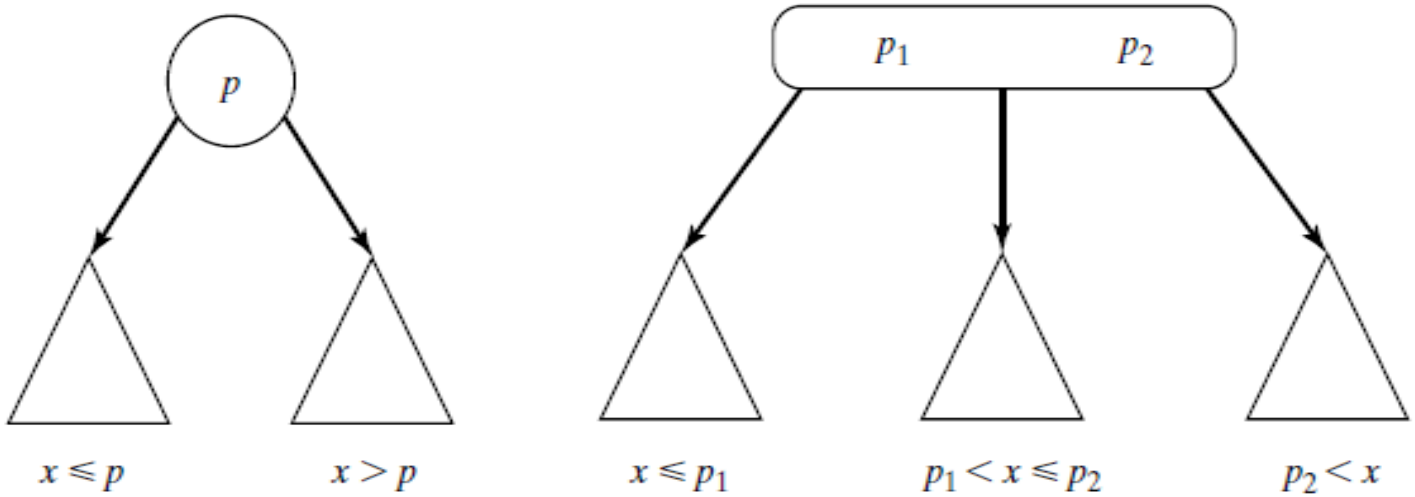
- Non-Spatial Join example:

```
SELECT D.name
FROM District D, Business B
WHERE D.authority=B.authority
AND B.type = 'Restaurant'
```

Spatial Indexing

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.



Spatial Indexing

Choice for spatial indexing:

- R-tree provides better search performance yet!
- R-tree is a hierarchical collection of rectangles.