**Spatial databases**: specialized type of db designed to manage and analyze data that is related to spatial or geographic information. They are equipped with features tailored to handle geometric objects and enable efficient searching and analysis of spatial data.

**Components of Spatial Databases:-**

*1. Spatial Data Storage:*

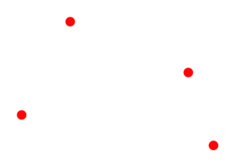
* Store various geometric objects such as points, lines, polygons, and other shapes specified by the Open Geospatial Consortium (OGC).
* They can handle complex stuff like 3D objects which is done through specialized indexing methods and data structures tailored for spatial information.s
* Coordinate systems enable accurate representation & manipulation of geo-locations.
* Traditional databases have been enhanced to handle spatial data types effectively, often through the inclusion of geometries or feature data types.
* Most relational & ORDBMS use spatial extensions to support spatial data handling.

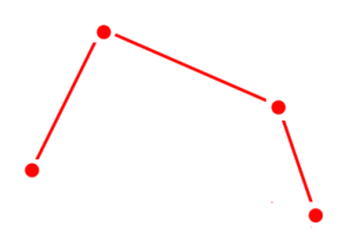
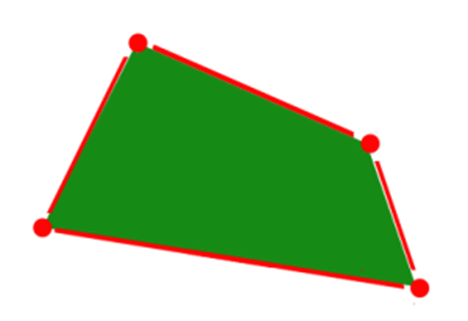
*2. Spatial Objects:*

* Include points, lines, polygons, geometry collections, curves, and surfaces.
* Points represent individual locations in space defined by coordinates.
* Lines are sequences of connected points, forming paths or routes.
* Polygons define enclosed areas through connected points, creating closed shapes.
* Geometry collections are containers for multiple spatial objects of different types.
* Some databases support more complex shapes like curves and surfaces.

*3. Spatial Dimensions:*

* Spatial databases can handle both 2D and 3D spatial data, providing depth and breadth to spatial representation.
* Some databases include time as a dimension, allowing them to store and analyze spatiotemporal data, they can track changes over time + spatial information, enhancing the depth of spatial analysis by considering temporal aspects.

In the **vector data model**, information is stored as individual spatial elements, typically geometric primitives like points, lines, and polygons.

1. *Vector Points:* Represented by XY coordinates, often in latitude and longitude. Used for small features that may be too minute to be shown as polygons, such as city locations on a global scale.
2. *Vector Lines:* Connect vertices in a set order, forming linear features like rivers, roads, and pipelines on maps. The thickness of the lines may vary to indicate the significance of features, such as busier highways having thicker lines.
3. *Vector Networks:* Different from regular lines, networks are topologically connected elements with junctions and turns, often representing interconnected systems like road networks. They follow set rules, restricting turns and movement on one-way streets when determining optimal routes.
4. *Vector Polygons:* They represent areas and have measurable properties such as square footage for buildings and acreage for agricultural fields.

**Raster data** consists of pixels or grid cells, each possessing its own value or class. These pixels are typically arranged regularly and can be square or differently shaped. Raster images often appear pixelated due to the discrete nature of pixel values. For instance, in satellite images, each pixel has red, green, and blue values, while in elevation maps, each pixel represents a specific height.

1. *Discrete Rasters:* These have distinct values representing themes/categories, like land cover classes/soil types. They are commonly used in network analysis & proximity operations.
2. *Continuous Rasters:* These contain grid cells with gradually changing data, such as elevation or temperature. These surfaces can be derived from fixed registration points, like sea level for digital elevation models. Phenomena can gradually vary along a continuous raster, for example, depicting the diffusion of an oil spill from a source with higher conc to lower conc over dist.

Raster data models, especially in a grid format, are commonly employed for satellite and remote sensing data. The cell size in the raster grid corresponds to the spatial resolution, providing a simple understanding of the data's positional accuracy.

**Spatial relations** play a crucial role in spatial db & geographic information systems (GIS). They enable the analysis & understanding of the relationships b/w different spatial objects.

* *Distance: M*easures spatial separation b/n two or more spatial objects. Helps determine how far apart objects are from each other and is fundamental for various spatial analyses like proximity analysis and network analysis.
* *Intersection:* Intersection checks whether two spatial objects share a common area or point. It identifies where the boundaries of two or more objects meet or overlap, providing insights into spatial overlaps and intersections.
* *Containment:* Containment examines whether one spatial object is entirely within another. It helps determine if one object fully encompasses or contains another object, which is essential for tasks like spatial containment queries and spatial joins.
* *Proximity:* Proximity identifies spatial objects that are close to each other. It helps find neighboring objects within a certain distance threshold and is crucial for applications such as location-based services and spatial clustering.
* *Topological relations:* Describe boundaries, interiors, and exteriors of spatial objects. These relations define how spatial objects are connected and organized in space, providing a foundation for more advanced spatial analyses and services. Applications - spatial network analysis, route planning, and geometric operations in GIS services.

These spatial relations facilitate various spatial analyses, decision-making processes, and applications in fields like urban planning, environmental management, and transportation.

**Spatial SQL queries:-**

* *SELECT Statements:* Retrieve spatial data from the db using SELECT statements.
* *Spatial Operators:* Use operators like ST\_Within or ST\_Intersects for operations.
* *Index Utilization:* Leverage spatial indexing to optimize query performance.
* *Aggregation Functions:* Calculate spatial statistics or aggregate spatial data.
* *Join Operations:* Combine spatial data from multiple tables based on spatial relationships.
* *Spatial Analysis Functions:* Perform spatial analyses such as buffering, overlay, and nearest neighbor search.
* *Spatial Databases:* PostgreSQL with PostGIS, Oracle Spatial, or Microsoft SQL Server with Spatial Extensions are popular choices.

**A geodatabase**, or **Geographic Database**, is a specialized spatial database used to store and manage georeferenced data related to specific locations on Earth. They facilitate the integration and management of spatial data across different domains, such as urban planning, environmental management, and public safety services. For example, a city might use it to manage datasets related to wastewater management, land registry, transportation networks, and emergency response services.

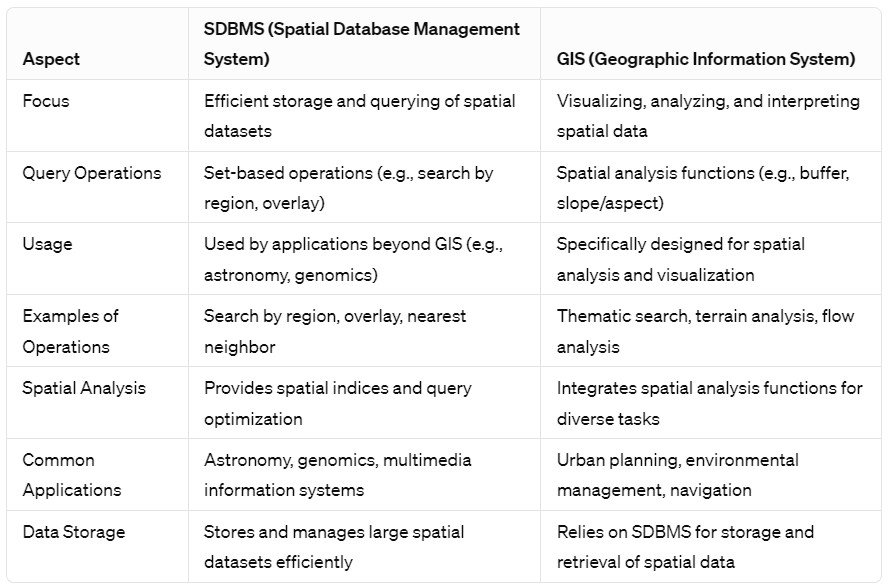
**Value of SDBMS:**

* Like traditional databases, SDBMS ensure data persistence (durability) even in the event of system failures or crashes.
* SDBMS allows multiple users to access and manipulate spatial data simultaneously, ensuring concurrency (handle multiple transactions) and data integrity.
* Capable of handling large spatial datasets that may exceed the memory capacities of computers, providing scalability to accommodate vast amounts of spatial information.
* Unlike non-spatial databases, SDBMS are optimized for spatial queries, enabling efficient retrieval and analysis of spatial data.

1. *Examples of Non-Spatial Queries:* Listing bookstore names with more than ten thousand titles. Listing top customers based on sales in a specific year.
2. *Examples of Spatial Queries:* Listing bookstore names within ten miles of Minneapolis. Listing customers residing in Tennessee and neighboring states.
3. *Examples of Non-Spatial Data:* Names, phone numbers, & emails add of people.
4. *Examples of Spatial Data:* Census data. Satellite imagery from NASA. Weather and climate data. Ecological features like rivers and farms. Medical imaging data.

**Application Domains for Spatial Data and Queries:**

* *Military:* Tracking enemy troop movements.
* *Insurance:* Identifying homes at risk of flooding.
* *Healthcare:* Analyzing MRI scans for medical diagnoses.
* *Biology:* Studying gene topology and sequence features.
* *Astronomy:* Identifying celestial objects like galaxies.



**Spatial Query Language:**

* *Spatial Data Types:* Includes point, linestring, polygon, and other geometric shapes.
* *Spatial Operations:* Encompass operations like overlap, dist, & nearest neighbor.
* *Callable from Query Language:* SQL3, used by the underlying DBMS.

**Example Spatial Query:**

-- Select senators whose district area is greater than 300

SELECT S.name

FROM Senator S

WHERE S.district.Area() > 300;

**Multi-scan Query Example:**

1. *Spatial Join Example:*

-- Select senators whose district area is greater than 300 and who have businesses within their district

SELECT S.name

FROM Senator S, Business B

WHERE S.district.Area() > 300 AND Within(B.location, S.district);

1. *Non-Spatial Join Example:*

-- Select female senators with matching social security numbers from businesses

SELECT S.name

FROM Senator S, Business B

WHERE S.soc\_sec = B.soc\_sec AND S.gender = 'Female';

**Query Optimization:**

1. *Spatial Operation Strategies:* Different strategies for processing spatial operations.
2. *Computation Cost:* Depends on various parameters.
3. *Query Optimization Process:* Orders operations and selects efficient strategies based on dataset details.
4. *Example Query Optimization:*

SELECT S.name

FROM Senator S, Business B

WHERE S.soc\_sec = B.soc\_sec AND S.gender = 'Female';

Process (S.gender = 'Female') before (S.soc\_sec = B.soc\_sec)

Do not use an index for processing (S.gender = 'Female')

Query optimization involves arranging operations in a query and selecting efficient strategies, considering dataset details. In the example query, processing the condition (S.gender = 'Female') before (S.soc\_sec = B.soc\_sec) and choosing not to use an index for (S.gender = 'Female') are examples of optimization decisions.