UNIT-II

DATA MANAGEMENT AND PROCESSING SYSTEMS

3.1 HARDWARE AND SOFTWARE TRENDS

Information and Communication Technology is an ever expanding field in both dimensions hardware and software. Advances in computer hardware seem to take place at an ever-increasing rate. Every now and then, a faster, more powerful processor generation replaces the previous one. Computers are also becoming increasingly portable, while offering this increased performance. The computing power that we have available in today's handheld computers is a multiple of the performance that the first PC had when it was introduced in the early 1980's. Current PCs have orders of magnitude more memory and storage capacity than the so-called minicomputers of 30 years ago. Computers are also becoming increasingly affordable. Hand-held computers are now commonplace in business and personal use, equipping field surveyors with powerful tools, complete with GPS capabilities for instantaneous georeferencing.

To support these hardware trends, software providers continue to produce application programs and operating systems that, while providing a lot more functionality, also consume significantly more memory. Software technology has developed somewhat slower and often cannot fully utilize the possibilities offered by the exponentially growing hardware capabilities. Existing software however performs better when run on faster computers.

Alongside these trends, there have also been significant developments in computer networks. Today almost any computer on Earth and outer space can be connect to network, and contact them virtually anywhere else, allowing fast and reliable exchange of data. Mobile phones are more and more frequently being used to connect to computers on the Internet. The UMTS protocol (Universal Mobile Telecommunications System), allows digital communication of text, audio, and video at a rate of approximately 2 Mbps. The new HSDPA protocol offers up to 10 times this speed. LTE/WiMax under 4G network offers speed ranging from 5 Mbps to 20 Mbps. 5G network is under development with a proposed speed of 500 Mbps.

Bluetooth version 2.0 /3.0 is a standard that offers up to 3 Mbps to 24Mbps over wireless connections, especially between palm- and laptop computers and their peripheral devices, such as a mobile phone, GPS or printer at short range. Wireless LANs (Local Area Networks), under the Wi-Fi standard, offer a bandwidth of up to 108 Mbps on a single connection point, to be shared between computers. They are more and more used for constructing a computer network in office buildings and in private homes.

When the medium of communication is not the air, but copper or fiber optic cables called wired networks, standard 'Dial-up' telephone modems allow rates up to 56 kbps. Digital telephone links (ISDN) support much higher rates: up to 1.5 Mbps. ADSL technology widely available through telephone companies on standard copper-wire networks supports transfer rates anywhere between 2 and 20 Mbps towards the customer (downstream), and between 1 and 8 Mbps towards the network (upstream) depending on the internet provider and quality of the network infrastructure.

Wide-area computer networks (national, continental, global) have a capacity of several Gbps. Fiber Optics-based Local Area Network (LAN), supports a transmission rate locally of 1 Gbps. Optical fiber however offers speed ranging from 5 Gbps to 20 Gbps. Also in 2012, Nippon Telegraph and Telephone successfully transferred 1 Penta bit per second over a single 50 km fiber.

3.2 GEOGRAPHIC INFORMATION SYSTEMS

A GIS is a computer-based system that provides the following four sets of capabilities to handle geo-referenced data:

1. **Data capture and preparation**
2. **Data management,**
3. **Data manipulation and analysis**
4. **Data presentation**

In early days manual maps in analogue form were used as data sources for GIS processing and paper maps were produced as a result. The use of modern techniques has led to an increased use of computers and digital information in all aspects of spatial data processing.

A GIS planning projects require data sources (spatial and non-spatial), from different institutes, like national mapping agencies (Survey of India), geological (USGS of United States, GSI in India), soil (SLUSI in India), and forest survey institutes (FSI in India), and national census bureau. The data sources obtained may be from different time periods, and the spatial data may be in different scales or projections. With the help of a GIS, the spatial data can be stored in digital form in world coordinates. This makes scale transformations unnecessary, and the conversion between map projections can be done easily with the software. With the spatial data thus prepared, spatial analysis functions of the GIS can then be applied to perform the planning tasks.

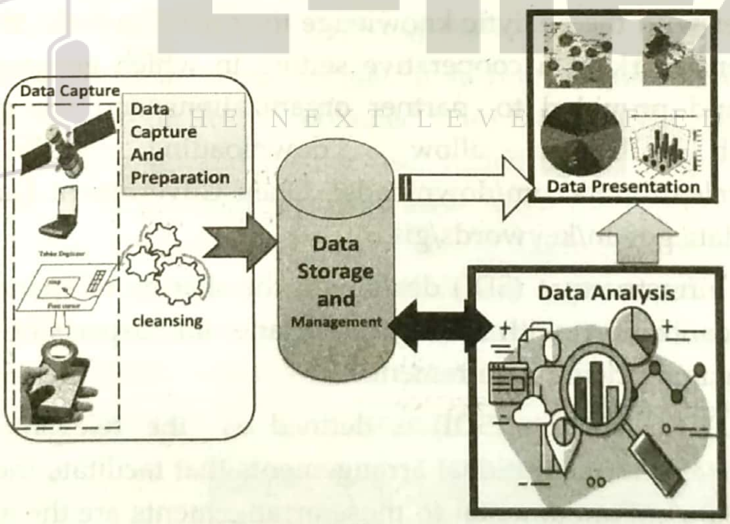
- **GIS Software**

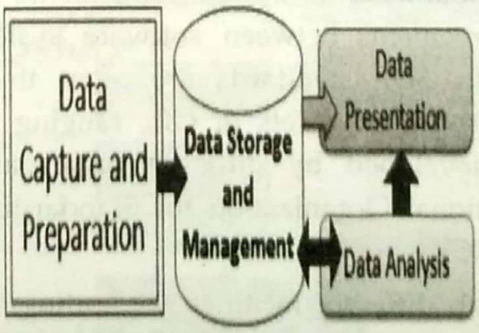
GIS is a huge data store that stores spatial data, a toolbox, a technology, an information source or a field of science. The main characteristics of a GIS software package are its powerful analytical functions that provide means for deriving new geoinformation from existing spatial and attribute data. The discipline of geographic information science is driven by the use of our GIS tools, and these are in turn improved by new insights and information gained through their application in various scientific fields. Spatial information theory is one such field, which focuses specifically on providing the background for the production of tools for the handling of spatial data.

All GIS packages available on the market have their own strengths and weaknesses, resulting from the development history and/or intended application domain(s) of the package. Some GISs have traditionally focused more on support for raster-based functionality, others more on vector-based spatial objects. Any package that provides support for only rasters or only objects, is not a complete GIS. Well-known, full-fledged GIS packages include ILWIS, Intergraph's GeoMedia, ESRI's ArcGIS, and MapInfo from Map-Info Corp. There is no particular GIS package which is necessarily 'better' than another one: this depends on factors such as the intended application, and the expertise of its user. ILWIS's traditional strengths are in raster processing and scientific spatial data analysis, especially in project-based GIS applications. Intergraph, ESRI and MapInfo products have been known better for their support of vector-based spatial data and their operations, user interface and map production.

- **GIS Architecture and Functionality**

Geographic information system consists of software, data, people, and an infrastructure. A GIS consists of several functional components—components which support key GIS functions. These are data capture and preparation, data storage, data analysis, and presentation of spatial data.





Data capture and preparation

Data can be collected through first hand observation called as primary source, or through individual, organization or published data called as secondary source. Capturing and acquisition is done through scanning, photogrammetric, remote sensing, digitization of analog map, field survey, GPS survey or manual data entry. This data is then prepared for an application under consideration by removing error, calibration, checking quality, rasterization, vectorization etc.

Data storage

The different types of information required for a GIS require storage which allows the information to be updated and queried for analysis by the user. Spatial data is usually stored as themes, layers, or coverage. Attribute data is the information about an object or feature.

Data analysis

A good system and/or software package allows the user to define and execute spatial and attribute procedures. This is commonly thought of as the heart of the GIS. Overlaying, buffering, modeling, and analysis are some of the methods used in building a coverage or project.

Presentation of spatial data

After the careful preparation on several mapping tools which are integrated with GIS, the maps are presented to users. The final maps are of high cartographic quality and are brought out using a wide range of devices.

- **Spatial Data Infrastructure (SDI)**

Government, industry leaders, academics, and nongovernmental organizations connect each other with the analytic knowledge they need to make the critical decisions. Many organizations work in a cooperative setting in which geographic information is obtained from, and provided to, partner organizations and the general public. E.g. Natural Earth Data allow downloading GIS data from http://www.naturalearthdata.com/downloads/, Open Government Data (OGD) Platform in India - https://data.gov.in/keywords/gis.

Spatial Data Infrastructure (SDI) deals with the sharing of spatial data between the GIS in various organizations with the key importance and aspects of data dissemination, security, copyright and pricing requirement.

Spatial Data Infrastructure (SDI) is defined as "the relevant base collection of technologies, policies and institutional arrangements that facilitate the availability of and access to spatial data". Fundamental to those arrangements are the agreements between organizations and the agreements between software systems on how to share the geographic information. In SDI, standards are often the starting point for those agreements. Standards exist for all facets of GIS, ranging from data capture to data presentation. They are developed by different organizations, of which the most prominent are the International Organization for Standardization (ISO) and the Open Geospatial Consortium (OGC).

SDI provides users with different facilities for finding, viewing, downloading and processing data. Because the organizations in an SDI are normally widely distributed over space, computer networks are used as the means of communication. With the development of the internet, the functional components of GIS have been gradually become available as web-based applications. Much of the functionality is provided by so called geo-webservices, software programs that act as an intermediate between geographic data and the users of the web. Geo-webservices can vary from a simple map display service to a service which involves complex spatial calculations. For their spatial data handling, these services commonly use standardized raster and vector representations.

3.3 STAGES OF SPATIAL DATA HANDLING

1. Spatial data capture and preparation
2. Spatial data storage and maintenance
3. Spatial query and analysis
4. Spatial data presentation

1. Spatial Data Capture and Preparation

Data can be collected through first hand observation called as primary source, or through individual, organization or published data called as secondary source. Capturing and acquisition is done through scanning, photogrammetric, remote sensing, digitization of analog map, field survey, GPS survey or manual data entry. Remote sensing, in particular, is the field that provides photographs and images as the raw base data from which spatial data sets are derived. Surveys of the study area often need to be conducted for data that cannot be obtained with remote sensing techniques, or to validate data thus obtained. Traditional techniques for obtaining spatial data, typically from paper sources, included manual digitizing and scanning. In recent years there has been a significant increase in the availability and sharing of digital (geospatial) data.

The data, once obtained in digital format, may not be quite ready for use in the system. This may be because the format obtained from the capturing process may not fit the format required for storage and further use, which means that some type of data conversion is required. This problem may also arise when the captured data represents only raw base data, out of which the real data objects of interest to the system still need to be constructed. For example, semi-automatic digitizing may produce line segments, while the application's requirements are that non-overlapping polygons are needed. A build-and-verification phase would then be needed to obtain these from the captured lines.

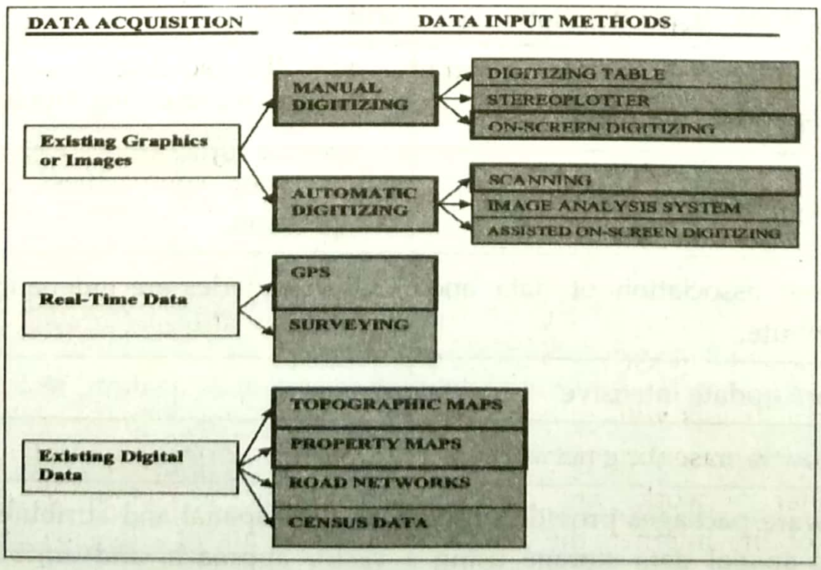


Figure : Spatial Input methods and devices

2. Spatial Data Storage and Maintenance

Data storage and management plays a key role in the processing and the analyzing geospatial data. Spatial data is organized in layers by theme and/or scale, like in thematic categories, such as land use, topography and administrative subdivisions, or according to map scale. This spatial data must be a representation of the real world that has to be designed to reflect phenomena and their relationships as naturally as possible. In a GIS, features are represented with their (geometric and non-geometric) attributes and relationships. The geometry of features is represented with primitives of the respective dimension: a windmill probably as a point, an agricultural field as a polygon. The primitives follow either the vector, as in the example, or the raster approach.

Vector data types describe an object through its boundary, thus dividing the space into parts that are occupied by the respective objects. The raster approach subdivides space into (regular) cells, mostly as a square tessellation of dimension two or three. These cells are called cells or pixels in 2D, and voxels in 3D. The data indicates for every cell which real world feature it covers, in case it represents a discrete field. In case of a continuous field, the cell holds a representative value for that field. The storage of a raster is, straightforward. It is stored in a file as a long list of values, one for each cell, preceded by a small list of extra data (called as 'file header') that informs how to interpret the long list. The order of the cell values in the list can be—but need not be—left-to-right, top-to-bottom. This simple encoding scheme is known as row ordering. The header of the raster file will typically inform how many rows and columns the raster has, which encoding scheme is used, and what sort of values are stored for each cell. Raster files can be quite big data sets. For computational reasons, it is wise to organize the long list of cell values in such a way that spatially nearby cells are also near to each other in the list.

Sr. No	Vector	Raster
1	Complex Data structure representation.	Simple data structure representation.
2	Overlay more difficult of implement.	Overlay implementation in simple way.
3	Inefficient in image processing as it works using function.	Efficient for image processing as it works with pixel.
4	Topologies represented efficiently.	Difficult is representing topologies.
5	Adapts well to scale changes.	Allow poor compactness of data while compression.
6	Allow association of data and attribute.	Cell boundaries are independent of feature boundary.
7	More update intensive	----------
8	Allow representing network.	----------

GIS software packages provide support for both spatial and attribute data, i.e. they accommodate spatial data storage using a vector approach, and attribute data using tables. Substantial GIS applications have been able to link to an external database to store attribute data and make use of its superior data management functions. All major GIS packages provide facilities to link with a DBMS and exchange attribute data with it. Spatial (vector) and attribute data are still some times stored in separate structures, although they can now be stored directly in a spatial database.

Maintenance of Spatial data is defined as the combined activities to keep the data set up-to-date and as supportive as possible to the application under consideration. It deals with obtaining new data, and entering them into the system, possibly replacing outdated data. The purpose is to have an up-to-date stored data set available. After a major earthquake, for instance, we may have to update our road network data to reflect that roads have been washed away, or have otherwise become impassable and irrelevant. The need for updating spatial data stems from the requirements that the data users impose, as well as the fact that many aspects of the real world change continuously. These data updates can take different forms. It may be that a complete, new survey has been carried out, from which an entirely new data set is derived that will replace the current set. It may also be that local (ground) surveys have revealed local changes, for instance, new constructions, or changes in land use or ownership. In such cases, local change to the large spatial data set is more typically required. Such local changes should respect matters of data consistency, i.e. they should leave other spatial data within the same layer intact and correct.

3. Spatial Query and Analysis

A spatial query is a special type of database query supported by geodatabases and spatial databases. Spatial analysis uses operators that use spatial data to derive new geoinformation. Spatial queries and process models play an important role in this functionality. One of the key use of GIS has been to support spatial decisions. Spatial decision support systems (SDSS) are a category of information systems composed of a database, GIS software, models, and a knowledge engine which allow users to deal specifically with location based problems.

In a GIS, data are usually grouped into layers (or themes). Usually, several themes are part of a project. The analysis functions of a GIS use the spatial and non-spatial attributes of the data in a spatial database to provide answers to user questions. GIS functions are used for maintenance of the data, and for Spatial analyzing the data in order to infer information from it. Analysis of spatial data can be defined as computing new information that provides new insight from the existing, stored spatial data. The exact nature of the analysis will depend on the application requirements, but computations and analytical functions operate on both spatial and non-spatial data.

4. Spatial Data Presentation

The presentation of spatial data, whether in print or on-screen, in maps or in tabular displays, or as 'raw data', is closely related to the disciplines of cartography, printing and publishing. The presentation may either be an end-product, for example as a printed atlas, or an intermediate product, as in spatial data made available. Cartography and scientific visualization make use of these methods and devices to produce their products. After the careful preparation on several mapping tools which are integrated with GIS, the maps are presented to users. The final maps are of high cartographic quality and are brought out using a wide range of devices.

Method	Devices
Hard copy	Printer
	Plotter
	Film writer
Soft copy	Computer screen (CRT), flat panel
Output of digital data sets	Magnetic tape
	CD-ROM
	computer network

3.4 DATABASE MANAGEMENT SYSTEM

A database is a large, computerized collection of structured data. A database management system (DBMS) is a software package that allows the user to set up, use and maintain a database.

Designing a database is not an easy task. Firstly, one has to consider carefully what the database purpose is, and who its users will be. Secondly, one needs to identify the available data sources and define the format in which the data will be organized within the database. This format is usually called the database structure. Lastly, data can be entered into the database. It is important to keep the data up-to-date, and it is therefore wise to set up the processes for this, and make someone responsible for regular maintenance of the database. Documentation of the database design and set-up is crucial for an extended database life. DBMS offers generic functionality for database organization and data handling. DBMS package offers a useful set of functions, and the capacity to store terabytes of information.

- **Reasons for using a DBMS**

There are various reasons why one would want to use a DBMS for data storage and processing.

1. **A DBMS supports the storage and manipulation of very large data sets.**

Some data sets are so big that storing them in text files or spreadsheet files becomes too awkward for use in practice. The result may be that finding simple facts takes minutes, and performing simple calculations perhaps even hours. A DBMS is specifically designed for this purpose.

2. **A DBMS can be instructed to guard over data correctness.**

For instance, an important aspect of data correctness is data entry checking: ensuring that the data that is entered into the database does not contain obvious errors. Rule check in DBMS is known as integrity constraints that can be defined in and automatically checked by a DBMS. More complex integrity constraints are certainly possible, and their definition is part of the design of a database.

3. **A DBMS supports the concurrent use of the same data set by many users.**

Large data sets are built up over time, which means that substantial investments are required to create and maintain them, and that probably many people are involved in the data collection, maintenance and processing. These data sets are often considered to be of a high strategic value for the owner(s), which is why many may want to make use of them within an organization. Moreover, for different users of the database, different views on the data can be defined. In this way, users will be under the impression that they operate on their personal database, and not on one shared by many people. They may all be using the database at the same time, without affecting each other's activities. This DBMS function is called concurrency control.

4. **A DBMS provides a high-level, declarative query language**

A query is a computer program that extracts data from the database that meet the conditions indicated in the query

5. **A DBMS supports the use of a data model**

A data model is a language with which one can define a database structure and manipulate the data storedin it. The commonly used data model is the relational data model. Its primitives are tuples (also known as records, or rows) with attribute values, and relations, being sets of similarly formed tuples.

6. **A DBMS includes data backup and recovery functions to ensure data availability at all times.**

As potentially many users rely on the availability of the data, the data must be safeguarded against possible calamities. Regular back-ups of the data set and automatic recovery schemes provide an insurance against loss of data.

7. **A DBMS allows the control of data redundancy.**

A well-designed database takes care of storing single facts only once. Storing a fact multiple times—a phenomenon known as data redundancy—can lead to situations in which stored facts may contradict each other, causing reduced usefulness of the data. Redundancy, however, is not necessarily always problematic, as long as we specify where it occurs so that it can be controlled for.

- **Alternatives For Data Management**

The decision whether or not to use a DBMS will depend, among other things, on how much data there is or will be, what type of use will be made of it, and how many users might be involved.

When the data set is small, its use relatively simple, and with just one user—we might use simple text files, and a text processor. Think of a personal address book as an example, or a small set of simple field observations. Text files offer no support for data analysis whatsoever, except perhaps in alphabetical sorting.

If our data set is still small and numeric by nature, and we have a single type of use in mind, a spreadsheet program will suffice. This might be the case if we have a number of field observations with measurements that we want to prepare for statistical analysis, for example. However, if we carry out region- or nation- wide censuses, with many observation stations and/or field observers and all sorts of different measurements, one quickly needs a database to keep track of all the data. It should also be noted that spreadsheets do not accommodate concurrent use of the data set well, although they do support some data analysis, especially when it comes to calculations over a single table, like averages, sums, minimum and maximum values.

All such computations are usually restricted to just a single table of data. When one wants to relate the values in the table with values of another nature in some other table, some expertise and significant amounts of time are usually required to make this happen.

- **The Relational Data Model**

A data model is a language that allows the definition of:

❖ The structures that will be used to store the base data,

❖ The integrity constraints that the stored data has to obey at all moments in time, and

❖ The computer programs used to manipulate the data

❖ For the relational data model, the structures used to define the database are attributes, tuples and relations. Computer programs either perform data extraction from the database without altering it, in which case we call them queries, or they change the database contents, and we speak of updates or transactions.

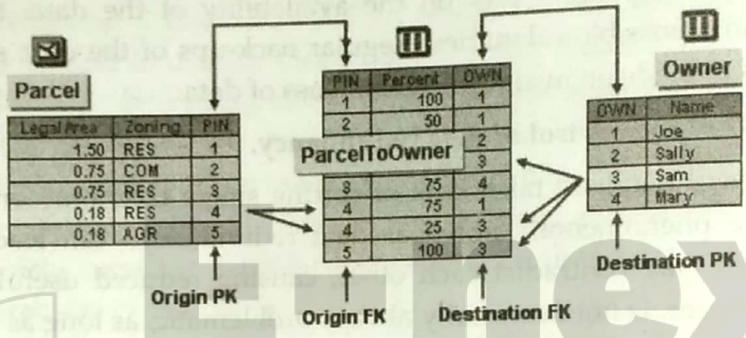


Figure : Relational Model Example

The above database consists of three tables, one for storing parcle's details, one for mapping parcel to its owner and a third one for storing owner details.

Relations, tuples and attributes

In the relational data model, a database is viewed as a collection of relations, commonly also known as tables. A table or relation is itself a collection of tuples (or records). In fact, each table is a collection of tuples that are similarly shaped. A tuple has a fixed number of named fields, also known as attributes. All tuples in the same relation have the same named fields. An attribute is a named field of a tuple, with which each tuple associates a value, the tuple's attribute value.

When a relation is created,

- Must provide a name for the relation,
- Must indicate which attributes it will have, and
- Must set the domain of each attribute.

A relation definition obtained in this way is known as the relation schema of that relation. The definition of relation schemas is an important part of database design. The relation schemas together make up the database schema. Relation schemas are stable, and will rarely change over time. This is not true of the tuples stored in tables: they, typically, are often changing; either because new tuples are added, others are removed, or yet others will see changes in their attribute values. The set of tuples in a relation at some point in time is called the relation instance at that moment. This tuple set is always finite: It is possible to count how many tuples there are.

Finding tuples and building links between them

Database systems are particularly good at storing large quantities of data. The DBMS must support quick searches amongst many tuples. This is why the relational data model uses the notion of a key.

A key of a relation comprises one or more attributes. A value for these attributes uniquely identifies a tuple. If we have a value for each of the key attributes we are guaranteed to find no more than one tuple in the table with that combination of values, such that there is no tuple for the given combination. Every relation has a key.

A tuple can refer to another tuple by storing that other tuple's key value. This attribute is called a foreign key because it refers to the primary key of another relation. Two tuples of the same relation instance can have identical foreign key values.

- **Querying a Relational Database**

A query is a computer program that extracts data from the database that meet the conditions indicated in the query.

The **first query operator** is called tuple selection; Tuple selection works like a filter: it allows tuples that meet the selection condition to pass, and disallows tuples that do not meet the condition.

The operator is given some input relation, as well as a selection condition about tuples in the input relation. A selection condition is a truth statement about a tuple's attribute values such as: Distance <1000.

The **second operator** is called attribute projection. Besides an input relation, this operator requires a list of attributes, all of which should be attributes of the schema of the input relation. Attribute projection works like a tuple formatter: it passes through all tuples of the input, but reshapes each of them in the same way.

The output relation of this operator has as its schema only the list of attributes given, and we say that the operator projects onto these attributes.

The most common way of defining queries in a relational database is through the SQL language. SQL stands for Structured Query Language.

SELECT queries do not create stored tables in the database. This is why the result tables have no name: they are virtual tables. The result of a query is a table that is shown to the user who executed the query. Whenever the user closes her/his view on the query result, that result is lost. The SQL code for the query can be stored for future use. The user can re-execute the query again to obtain a view on the result once more.

Our **third query operator** differs from the two above in that it requires two input relations. The operator is called the join. It takes two input relations and produces one output relation, by gluing two tuples together, if they meet a specified condition

The number of attributes therefore increases.

INPUT

OBJECT ID#	Landuse Code
1	2
2	0
3	1

← Join Fields →

+

Landuse Code	Landuse Type
0	Unclassified
1	shrub
2	water

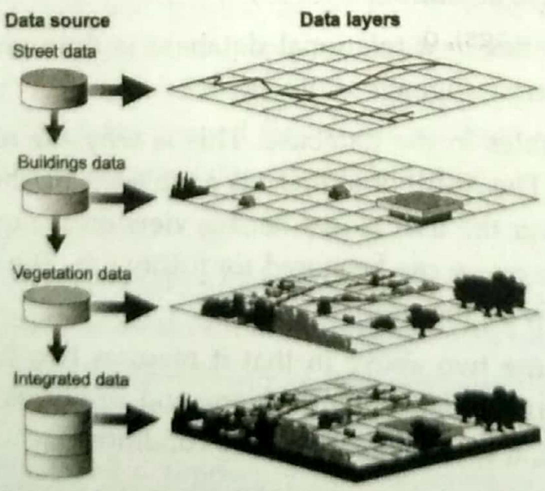
OUTPUT

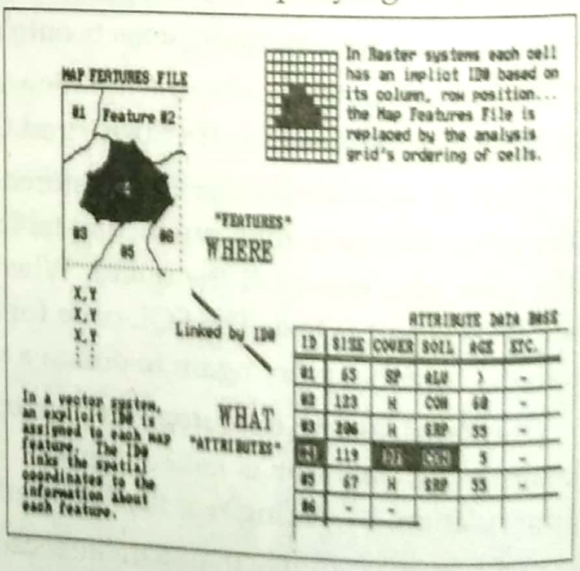
OBJECT ID#	Landuse Code	Join Table Landuse Code	Join Table Landuse Type
1	2	2	water
2	0	0	Unclassified
3	1	1	shrub

3.5 GIS AND SPATIAL DATABASES

- **Linking GIS and DBMS**

GIS software provides support for spatial data and thematic or attribute data. GIS stores spatial data and attribute data separately. This required the GIS to provide a link between the spatial data (represented with rasters or vectors), and their non-spatial attribute data. The strength of GIS technology lies in its built-in 'understanding' of geographic space and all functions that derive from this, for purposes such as storage, analysis, and map production. GIS packages themselves can store tabular data; however, they do not always provide a full-fledged query language to operate on the tables. DBMSs have a long tradition in handling attribute (i.e. administrative, non-spatial, tabular, thematic) data in a secure way, for multiple users at the same time. DBMS offer much better table functionality, since they are specifically designed for this purpose. A lot of the data in GIS applications is attribute data, so it made sense to use a DBMS for it. For this reason, many GIS applications have made use of external DBMS for data support. In this role, the DBMS serves as a centralized data repository for all users, while each user runs her/his own GIS software that obtains its data from the DBMS. This meant that a GIS had to link the spatial data represented with rasters or vectors, and the attribute data stored in an external DBMS. With raster representations, each raster cell stores a characteristic value. This value can be used to look up attribute data in an accompanying database table.





In vector representation, a spatial object like points, lines or polygon, are automatically given a unique identifier by the system. This identifier is usually just called the object ID or feature ID and is used to link the spatial object (as represented in vectors) with its attribute data in an attribute table. The principle applied here is similar to that in raster settings, but in this case each object has its own identifier. The ID in the vector system functions as a key, and any reference to an ID value in the attribute database is a foreign key reference to the vector system.

- **Spatial Database Functionality**

DBMS vendors have recognized the need for storing more complex data, like spatial data. The main problem was that there is additional functionality needed by DBMS in order to process and manage spatial data. Object-oriented and object-relational data models were developed for just this purpose. These extend standard relational models with support for objects, including 'spatial' objects.

GIS software packages are able to store spatial data using a range of commercial and open source DBMSs such as Oracle, Informix, IBM DB2, Sybase, and PostgreSQL, with the help of spatial extensions. Some GIS software have integrated database 'engines', and therefore do not need these extensions. ESRI's ArcGIS and QGIS for example, have data base software built-in. This means that the designer of a GIS application can choose whether to store the application data in the GIS or in the DBMS. Spatial databases, also known as geodatabases, are implemented directly on existing DBMS, using extension software to allow them to handle spatial objects.

A spatial database allows users to store query and manipulate collections of spatial data

There are several advantages in doing this, spatial data can be stored in a special database column, known as the geometry column, (or feature or shape, depending on the specific software package),. This means GISs can rely fully on DBMS support for spatial data, making use of a DBMS for data query and storage (and multi-user support), and GIS for spatial functionality. Small-scale GIS applications may not require a multi-user capability, and can be supported by spatial data support from a personal database.

A geodatabase allows a wide variety of users to access large data sets (both geographic and alphanumeric), and the management of their relations, guaranteeing their integrity. The Open Geospatial Consortium (OGC) has released a series of standards relating to geodatabases that (amongst other things), define :

- Which tables must be present in a spatial database (i.e. geometry columns table and spatial reference system table)
- The data formats, called 'Simple Features' (i.e. point, line, polygon, etc.)
- A set of SQL-like instructions for geographic analysis.

The architecture of a spatial database differs from a standard RDBMS not only because it can handle geometry data and manage projections, but also for a larger set of commands that extend standard SQL language (e.g. distance calculations, buffers, overlay, conversion between coordinate systems, etc.). Spatial databases support the storage of image data, the capabilities of spatial databases will continue to evolve over time. Currently, ESRI's ArcGIS geodatabase can store topological relationships directly in the database, providing support for different kinds of features (objects) and their behavior (relations with other objects), as well as ways to validate these relations and behaviors.

Querying a spatial database

A Spatial DBMS provides support for geographic co-ordinate systems and transformations. It also provides storage of the relationships between features, including the creation and storage of topological relationships. As a result one is able to use functions for 'spatial query' (exploring spatial relationships). To illustrate, a spatial query using SQL to find all the Metro City within 20 km of a River GANGA would look like this:

```
SELECT C. Name
FROM River AS R,
City as C
WHERE C. Type = "METRO" AND
R. name = "GANGA" AND
ST_Intersects (C. Geometry, CT_Buffer(R. Geometry, 20000)
```

In this case the WHERE clause uses the ST_Intersects function to perform a spatial join between a 20000 m buffer of the selected River and the selected subset of Cities. The Geometry column carries the spatial data.

QUESTIONS

1. Discuss the latest hardware and software trends in hardware and software.
2. Explain GIS architecture and functionality using suitable diagram.
3. Write a note on Spatial Data Infrastructure (SDI).
4. Briefly explain the four stages of spatial data handling.
5. List any five differences between Vector data and Raster data.
6. What is DBMS? Explain any five reasons for using DBMS is GIS.
7. Write a short note on Relational Model.
8. Explain the linking of GIS with Database.
9. Write a note on spatial database functionality.
10. Explain spatial database Querying using suitable example.