# UNIT 6 NON-SPATIAL DATA STRUCTURE

## Structure



## 6.1 INTRODUCTION

You have read about GIS concepts, spatial data, data models and data structures in previous two units. A typical GIS involves both spatial and non-spatial data. Spatial data provides the location information of the features whereas non-spatial data describes characteristics of the features. Non-spatial data is also known as attribute data. A combination of both data is known as geospatial data. It means that both (spatial and non-spatial) data are essential for successful operation of a GIS.

Therefore, the first principle you will learn here is that no GIS will work in the absence of non-spatial data. In this unit, you will be introduced to non-spatial data structures and database models. In the end of the unit, we will discuss the non-spatial data query such as structured query language.

### Objectives

After studying this unit, you should be able to:

- define non-spatial data;
- differentiate between data and information;
- describe types and models of databases;

- discuss geographic data and geographic information;
- enumerate the information organisation process; and
- explain non-spatial data query.

## 6.2 DATA AND INFORMATION

Data not properly collected and organised are a burden rather than an asset to an information user.

Let us first discuss the basic concepts of data and information. Generally, we use the terms *data* and *information* interchangeably but these two terms actually convey very distinct concepts. ***Data*** is defined as a body of facts or figures which have been gathered systematically for one or more specific purposes. Data is a plural and in a broad sense it can include things such as pictures (binary images), programmes and rules. Informally, data are the things we want to store in a database. It can exist in the following forms:

- linguistic expressions (e.g., name, age, address, date, ownership)
- symbolic expressions (e.g., traffic signs)
- mathematical expressions (e.g., $E = mc^2$) and
- signals (e.g., electromagnetic waves).

You have been introduced to the Data and Information in Unit 4, in MGY-001.

***Information*** is defined as data which have been processed into a form that is meaningful to a recipient and is of perceived value in current or prospective decision-making. It is data that make information useful for one person and same information may not be useful to another person. Information is only useful to its recipients when it is:

- relevant (i.e. has intended purposes and an appropriate level of required detail)
- reliable, accurate and verifiable (by independent means)
- up-to-date and timely (depending on purposes)
- complete (in terms of attribute, spatial and temporal coverage)
- intelligible (i.e. comprehensible by its recipients)
- consistent (with other sources of information)
- convenient and easy to handle, and
- adequately protected.

The primary function of an information system (e.g., GIS) is to convert *data* into *information* using the following processes:

**Conversion:** It is the process of transforming data from one format to another. The transformation, for example, may be from one unit of measurement to another likes km to cm or from one feature classification to another.

**Organisation**: It involves the process of organising or re-organising data according to database management rules and procedures and can be accessed cost-effectively.

**Structuring:** It represents the formatting or re-formatting of data, so that it can be acceptable to a particular software application or information system.

**Modelling**: It includes the statistical analysis and visualisation of data that improves user's knowledge base and intelligence in decision-making.

## 6.3 DATABASE

Everyday human activities produce data and most of these data are managed in databases. Conventionally, databases store only non-spatial entities but in GIS these non-spatial data can be converted as spatial data.

Before discussing the non-spatial data structure, it is required that you should know about the database. Let us discuss the concept of database in GIS. The concept of database is the approach to information organisation in computer-based data processing. A ***database*** is defined as an automated, formally defined and centrally controlled collection of persistent data used and shared by different users in an enterprise. The term '*centrally controlled*' means that databases tend to be physically distributed in different computer systems in the same time at different locations. A database is set-up to serve the information needs of an organisation. The sharing of data is the key to the concept of a database. Data in a database are described as 'permanent' in the sense that they are different from 'transient' data such as input to and output from an information system. The data usually remain in the database for a considerable length of time, although the actual content of the data can change very frequently. The use of database does not mean the demise of data files; data in a database are still organised and stored as *data files*. The use of database represents a change in the perception of data, mode of data processing and purposes of using data rather than physical storage of the data.

### 6.3.1 Creating a Database

The design and implementation of a database is guided by the relationships between data that to be stored in the database. Database design process involves identification and expression of these relationships and implementation of the database by setting-up of a new structure for these relationships within the chosen database software. Following steps are involved for the creation of database:

**Step 1:** This step involves data investigation. It is the 'fact finding' stage of database creation. Here the task is to consider the type, quantity and qualities of data to be included in the database. The nature of entities and attributes is decided.

**Step 2:** It comprises of data modelling. It is the process of forming a conceptual model of data by examining the relationships between entities and characteristics of entities and attributes. This stage, like data investigation stage, can be carried out independently of the software to be used.

**Step 3:** It consists of database design. Database design is the process of creating a practical design for database. The design will depend on the database software being used and its data model. This is the process of translating the logical design for the database (produced during data modelling stage) into a design for the chosen DBMS. Field names, type and structure are decided. In practice, design will be a compromise to fit the database design model with chosen DBMS.

**Step 4:** It involves database implementation which is the procedure of populating the database with attribute data and this is always followed by monitoring and upkeep, including fine tuning, modification and updating.

### 6.3.2 Relationship Perspective of Database

Relationships represent an important concept in database management. It describes the logical association between entities. Relationships can be *categorical* or *spatial*, depending on whether they describe location or other characteristics.

**Categorical Relationships:** These relationships describe the association among individual features in a classification system. The classification of data is based on the concept of *scale of measurement.* There are four scales of measurement: nominal, ordinal, interval and ratio.

- ***Nominal*** is a qualitative, non-numerical and non-ranking scale that classifies features on intrinsic characteristics. For example, in a land use classification scheme, polygons can be classified as industrial, commercial, residential, agricultural, public and institutional.

- ***Ordinal*** is a nominal scale with ranking which differentiates features according to a particular order. For example, in a land use classification scheme, residential land can be denoted as low density, medium density and high density.

- ***Interval*** is an ordinal scale with ranking based on numerical values that are recorded with reference to an arbitrary datum. For example, temperature readings in degrees centigrade are measured with reference to an arbitrary zero (i.e. zero degree temperature does not mean no temperature).

- ***Ratio*** is an interval scale with ranking based on numerical values that are measured with reference to an absolute datum, for example, rainfall data are recorded in mm with reference to an absolute zero (i.e. zero mm rainfall mean no rainfall).

Categorical relationships based on ranking are hierarchical or taxonomic in nature which means that data are classified into progressively different levels of detail. Data in the top level are represented by limited broad basic categories. Data in each basic category are then classified into different sub-categories. Sub-categories can be further classified into another level if required. Classification of descriptive data is typically based on categorical relationships.

**Spatial Relationships:** These relationships describe the association among different features in space. Spatial relationships are visually displayed when data are presented in the graphical form. There are numerous types of spatial relationships possible among features. Recording spatial relationships implicitly demands considerable storage space. Computing spatial relationships on-the-fly slows down data processing particularly, if relationship information is required frequently. There are two types of spatial relationships such as topological and proximal:

- ***Topological relationship*** which describes the property of adjacency, connectivity and containment of contiguous features about which you will be reading more in Unit 8 of MGY-003.

- ***Proximal relationship*** shows the property of closeness of non-contiguous features.

Spatial relationships are very important in geographical data processing and modelling. The objective of information organisation and data structure is to find a way that will handle spatial relationships with the minimum storage and computation requirements.

### 6.3.3 Types of Database

Database is primarily of two types:

- spatial, and
- non-spatial data.

**Spatial Data:** It includes location, shape, size and orientation information of features or objects. For example, a particular square in which its center (the intersection of its diagonals) specifies its location; its shape is a square; length of one of its sides specifies its size and angle its diagonals e.g., the *x*-axis specifies its orientation. Spatial data includes spatial relationships, for example, the arrangement of three stumps in a cricket ground.

**Non-spatial Data:** It is also known *attribute* or *characteristic* data. It consists of the characteristics of spatial features which are independent of all geometric considerations. Let us illustrate this with the help of an example. The non-spatial data of town comprise of name of the town, its population, settlement type, means of transportation and communication, administration set-up, education institutions, occupations and facilities. It is important to note that all the above mentioned data of town are not dependent on their location identities. Hence, non-spatial data is independent from location information. The fundamental difference between spatial and non-spatial data is given in Table 6.1.

**Table 6.1: Basic difference between spatial and non-spatial data**

| Spatial data | Non-spatial data |
|---|---|
| It has multi-dimensional nature and autocorrelated | It has one-dimensional nature and independent |

These above distinctions put spatial and non-spatial data into different philosophical camps with far-reaching implications for conceptual, processing and storage issues. For example, sorting is, perhaps, the most common and important non-spatial data processing function that is performed. It is not obvious how to even sort locational data such that all points end up nearby their nearest neighbours. These distinctions justify a separate consideration of spatial and non-spatial data models. In GIS, georelational data model stores spatial and non-spatial (attribute) data separately and also links them on the basis of identity of features. Further, georelational data model arranges these two data-sets in such a way that they can simultaneously be queried, analysed and displayed.

### 6.3.4 Non-Spatial Data in GIS

Non-spatial data are stored in GIS as tables. Such tables are known as non-spatial (attribute) tables. A non-spatial table is represented by rows and columns in which each row shows a spatial feature and each column represents a characteristic. The intersection of a row and a column gives the value of a specific characteristic for a particular feature as shown in Table 6.2. A row is also known as a record or a tuple and a column is known as a field or item.

**Table 6.2: Arrangement of rows and columns of a non-spatial data**

| Label-1A | Rivers | Length | Dams |
|---|---|---|---|
| 1 | 2 | 700 | 4 |
| 2 | 3 | 500 | 6 |
| 3 | 5 | 300 | 2 |

**Check Your Progress I**

*Spend 5 mins*

1) What is non-spatial data?

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 6.4 GEOGRAPHIC DATA AND GEOGRAPHIC INFORMATION

Geographic data are a special type of data because of their geographic nature. The geographic data are geographically referenced wherein we can locate and identify features by a spherical coordinate system (latitude and longitude). They are made up of *descriptive elements* (which tell what they are) and *graphical elements* (which tell what they look like, where they are found and how they are spatially related to one another). The descriptive elements are commonly referred to as *non-spatial data* while the graphical elements are commonly referred to as *spatial data*. The geographic data are always pertinent to features and resources of the Earth as well as human activities based on or associated with these features and resources. The primary purpose of collection of geographical data and their utilisation is for problem solving and decision-making associated with geography, i.e. location, distribution and spatial relationships within a particular geographical framework.

On the other hand, processing of geographic data give the geographic information. The geographic information are useful to improve the user's knowledge about the geography of the Earth's features and resources as well as human activities associated with these features and resources. It enables the user's to develop spatial intelligence for problem solving and decision-making concerning the occurrence, utilisation and conservation of the Earth's features and resources as well as impacts and consequences of human activities associated with them.

Spatial nature and characteristics of geographic data, generic concepts of information organisation and data structure cannot be applied directly to them. Geographic data have three dimensions which are given below:

a) temporal (e.g., $26^{th}$ December, 2004)

b) thematic (e.g., occurrence of tsunami in Indian ocean), and

c) spatial (e.g., affected area including southeastern coast of India).

GIS emphasises the use of spatial dimension for turning data into information which assist our understanding of geographic phenomena.

An operating system is a software programme that acts like an interface between the user and the hardware. It performs basic tasks like processing, CPU scheduling, memory management, file system, networking, access and security.

## 6.5 INFORMATION ORGANISATION

Information organisation can be studied from four perspectives as given below:

- data
- relationship
- operating system, and
- application architecture perspectives

Let us discuss about data perspective as we are more concerned about data in this unit first. Information organisation of geographic data is considered in terms of their descriptive and graphical elements. These two types of data elements have distinct characteristics, different storage requirements and separate processing requirements.

In the information organisation of descriptive data, each data item is served as the most basic element of information organisation. A *data item* represents an *occurrence* or *instance* of a particular characteristic pertaining to an entity (which can be a person, thing, event or phenomenon). It is the smallest unit of stored data in a database and commonly referred to as an *attribute*. In database terminology, an attribute is also referred to as a *stored field*. The value of an attribute can be in the form of a number (integer or floating - point), a character string, a date or a logical expression (e.g., *T* for 'true' or 'present' and *F* for 'false' or 'absent'). Some attributes have a definite set of values known as *permissible values* or *domain of values* (e.g., age of people from 1 to 70, categories in a land use classification scheme and the academic departments in a university).

A group of related data items form a *record* (Fig. 6.1). Related data items mean that items are occurrences of different characteristics pertaining to the same person, thing, event or phenomenon (e.g., in a land resource inventory, a record may contain related data items such as identification number, owner, size of land holding and use of land). A record may contain a combination of data items having different types of values. In the above example, a record has two character strings representing the identification number and dominant use of land, an integer representing the average size of land holding rounded to the nearest meter and a floating-point number representing identification. In database terminology, a record is always formally referred to as a *stored record* while in relational database management systems, records are called *tuples*.

A se[illegible]ated records constitute a *data file* (Fig. 6.1). Data file differs from [illegible]ta[illegible]ased on certain characteristics as given in Table 6.3. Related record [illegible]ea[illegible] records represent different occurrences of same type or class of people, [illegible]ing[illegible]nts or phenomena. A data file made up of a single record type with single [illegible]alu[illegible] items is called a *flat file* (Table 6.4). A data file made up of a single [illegible]eco[illegible] with nested repeating groups of items forming a multi-level organisation [illegible] ca[illegible] *hierarchical file* (Table 6.5).

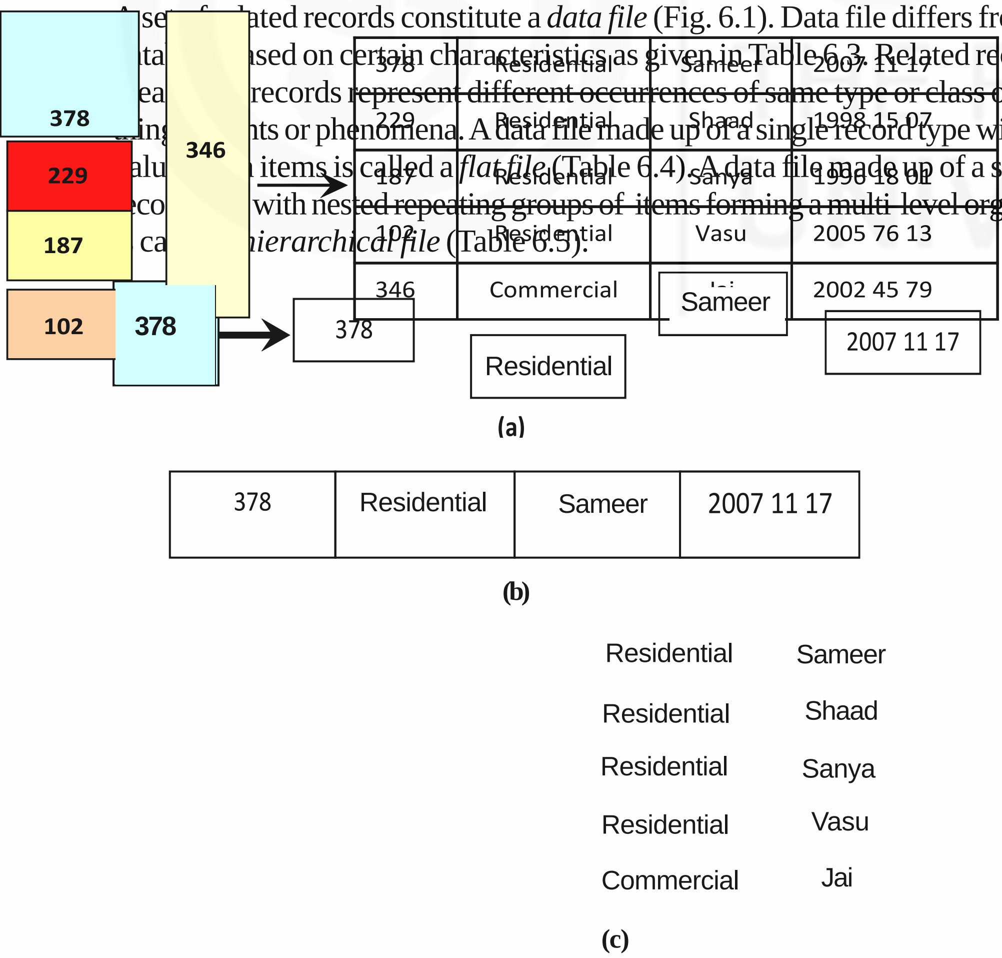


**Fig. 6.1: Arrangement of data item, record and data file in GIS; (a) data item pertaining to a land parcel, (b) a record of data items and (c) a table of records**

**Table 6.3: Distinction between a data file and a database**

| Characteristics of a data file | Characteristics of a database |
|---|---|
| A collection of records usually of the same data type and format description | A collection of interrelated records, organised in one or more data files, that may have different data types and format descriptions |
| Data file processing is usually associated with computer programming that aims at solving a particular problem, i.e. it stops when an answer is obtained | Database processing is always associated with database management systems that aim at solving the operation or production needs of an organisation, i.e. it involves routine, largely repetitive applications executed over and over again |
| Mainly used in support of the information need of an *ad hoc* application | Mainly used in support of the day to day operation of business (transaction processing) but increasingly used in decision support (management decision-making) |

**Table 6.4: Flat file**

| Ward No. | Population | No. of households | Average monthly income |
|---|---|---|---|
| 14 | 2431 | 645 | Rs. 10,500 |
| 21 | 1740 | 389 | Rs. 15,000 |
| 56 | 1985 | 557 | Rs. 12,000 |

**Table 6.5: Hierarchical file**

| Ward No. | Population | | No. of households | | Average monthly income | |
|---|---|---|---|---|---|---|
| | 1991 | 2001 | 1991 | 2001 | 1991 | 2001 |
| 14 | 1434 | 431 | 568 | 654 | Rs. 8,000 | Rs. 10,500 |
| 21 | 1047 | 1740 | 307 | 389 | Rs. 13,500 | Rs. 15,000 |
| 56 | 1286 | 1985 | 489 | 557 | Rs.9,000 | Rs. 12,000 |

A data file is individually identified by a *file name.* A data file may contain records having different types of data values or having a single type of data value. A data file containing records made up of character strings is called a *text file* or *ASCII file.* A data file containing records made up of numerical values in binary format is called a *binary file.* Data processing literature, collections of data items or records are sometimes referred to by other terms other than 'data file' according to their characteristics and functions. An *array* is a collection of data items of the same size and type (although they may have different values). There are two types of array:

- a one-dimensional array is called a *vector,* and
- a two-dimensional array is called a *matrix.*

A *table* is a data file with data items arranged in rows and columns. Data files in relational databases are organised as tables. Such tables are also called *relations* in relational database terminology. A *list* is a finite, ordered sequence of data items (known as *elements*). Here 'ordered' means that each element has a position in the list. An ordered list has elements positioned in ascending order of values; while an unordered list has no permanent relation between element values and position. Each element has a data type, in the simple list implementation, all elements must have same data type but there is no conceptual objection to lists whose elements have different data types.

A *tree* is a data file in which each data item is attached to one or more data items directly beneath it (Fig. 6.2). The connections between data items are called *branches.* Trees are often called *inverted trees* because they are normally drawn with the root at the top. The data items at the very bottom of an inverted tree are called *leaves*; other data items are called *nodes.* A *binary tree* is a special type of inverted tree in which each element has only two branches below it. A *heap* is a special type of binary tree in which the value of each node is greater than the values of its leaves. Heap files are created for sorting data in computer processing – the *heap sort algorithm* works by first organising a list of data into a heap.

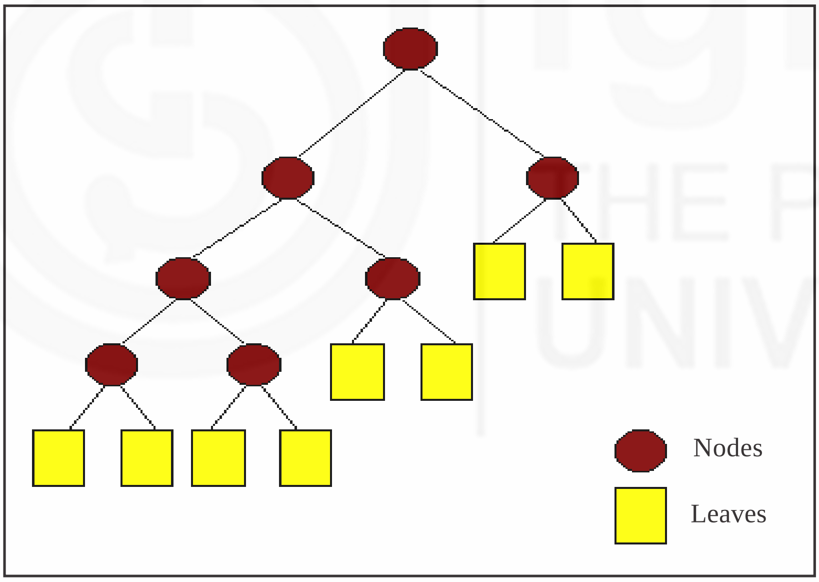


**Fig. 6.2: Tree data structure**

Information organisation is concerned with the internal organisation of data. It represents the user's view of data, i.e. conceptualisation of the real world. It is the lowest level of data abstraction, which can be done with or without any intent for computer implementation and it is expressed in terms of *data models.*

**Check Your Progress II**

***Spend 5 mins***

1) List two characteristics of a database.

.................................................................................................

.................................................................................................

.................................................................................................

## 6.6 DATABASE MODELS

A database model is the theoretical foundation of a database and fundamentally determines the manner in which a data can be stored, organised and manipulated in a database system. It thereby defines the infrastructure offered by a particular database system.

Databases can be organised in different ways known as *database models.* The database models are: *relational*, *network*, *hierarchical* and *object-oriented* (Fig. 6.3).

- ***hierarchical data*** are organised by records on a parent-child one-to-many relations
- ***network data*** are organised by records which are classified into record types within pointers linking associated records
- ***relational data*** are organised by records in relations which resemble a table, and
- ***object-oriented data*** are uniquely identified as individual objects that are classified into object types or classes according to the characteristics (attributes and operations) of the object.

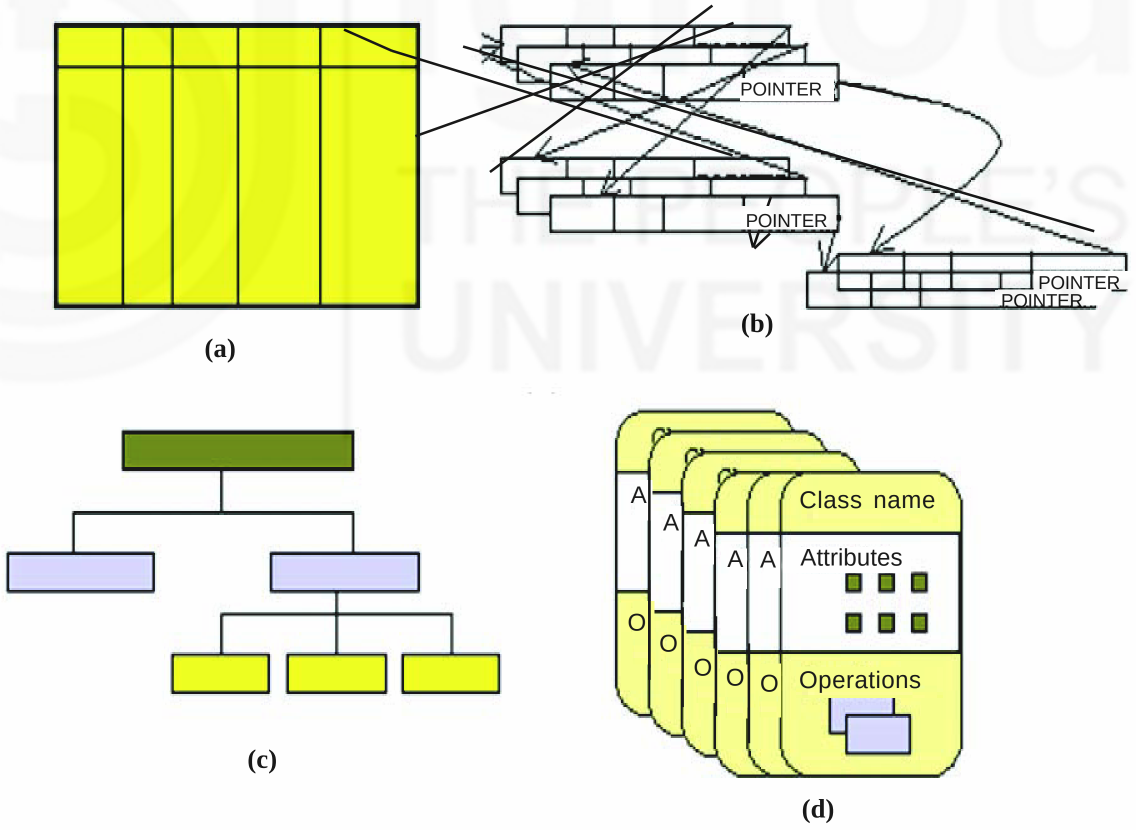


**Fig. 6.3: Database models; (a) relational, (b) network, (c) hierarchical and (d) object-oriented database models**

Now let us discuss all these database models one by one.

### 6.6.1 Hierarchical Model

It organises data in a tree structure. There is a hierarchy of parent and child data segments. This structure implies that a record can have repeating information generally in the child data segments. Data in a series of records have a set of field values attached to it. It collects all the instances of a specific record together as a record type. These record types are the equivalent of tables in the relational model and with the individual records being the equivalent of rows. To create links between these record types, the hierarchical model uses parent child relationships. These are a 1: N mapping between record types. This is done by using trees, like set theory used in the relational model, borrowed from mathematics. For example, an organisation might store information about population in a city such as ward name, locality name, street number, house number, residents name, etc. The organisation might also store information about resident's children such as name and date of birth. The resident and children data forms a hierarchy where the resident data represents parent segment and children data represents child segment. If a resident has four children then there would be four child segments associated with one resident segment. In a hierarchical database, the parent-child relationship is one-to-many. This restricts a child segment to having only one parent segment. In the hierarchical model, the links established by the pointers are permanent and cannot be modified. This makes the hierarchical model more rigid and inflexible causing difficulties in expansion or modification of databases. Hierarchical Database Management Systems (DBMSs) were popular from the late 1960s with the introduction of IBM's Information Management System (IMS) DBMS through the 1970s.

### 6.6.2 Network Model

The popularity of the network data model coincided with the popularity of the hierarchical data model. Some data were more naturally modelled with more than one parent per child. But the network model permitted the modelling of many-to-many relationships in data. The basic data modelling construct in the network model is the set construct. A set consists of an owner record type, a set name and a member record type. A member record type can have that role in more than one set; hence the multi-parent concept is supported. An owner record type can also be a member or owner in another set. The data model is a simple network, and link and intersection record types may exist as well as sets between them. Thus, the complete network of relationships is represented by several pair wise sets; in each set, some (one) record type is owner (at the tail of the network arrow) and one or more record types are members (at the head of the relationship arrow). Network model becomes complex with the increase in size of database. This model also suffers from inflexibility but the degree of inflexibility is lower compared to the hierarchical model because it provides multi-parent relationship.

### 6.6.3 Relational Model

Relational model based on the concept proposed by Codd (1970) and is popular among GIS users. A relational database allows the definition of data structures, storage and retrieval operations and integrity constraints. In such a database, data and relations between them are organised in tables. A table is a collection of records and each record in a table contains the same fields. The properties of relational tables are:

- values are atomic
- each row is unique
- sequence of rows is insignificant
- each column has a unique name

- column values are of the same kind, and
- sequence of columns is insignificant.

Certain fields may be designated as keys, which mean that searches for specific values of that field will use indexing to speed them up. Fields in two different tables take values from the same set; a join operation can be performed to select related records in the two tables by matching values in those fields. Often, but not always, the fields will have the same name in both tables. For example, an organisation table might contain (employee-ID, department-code) pairs and a department table might contain (department name-code, number of employee in the department) pairs to identify an organisation's salary expenses. We could sum the salaries of employees in the department by joining on the employee-code and department-code fields of the two tables. This can be extended to joining multiple tables on multiple fields. Because these relationships are only specified at retrieval time, the relational databases are classed as dynamic database management system. The Relational Database Management System (RDBMS) is database based on relational model.

The main disadvantage is the terminology of relational database which can be confusing because of the use of different terminologies by different users. Although the relational model is flexible than hierarchical model and network model but still suffers from data redundancy and can be slow and difficult to implement. Its efficiency is reduced with handling of complex data formats of GIS because of limited range of data types.

### 6.6.4 Object-Oriented Model

Object-Oriented Database (OODB) paradigm is the combination of Object-Oriented Programming Language (OOPL) systems and persistent systems. The power of OODB comes from the seamless treatment of both persistent data as found in databases and transient data as found in executing programmes. Object DBMSs add database functionality to object programming languages. They bring much more than persistent storage of programming language objects and provide full-featured database programming capability. A major benefit of this approach is the unification of the application and database development into a seamless data model and language environment. As a result, applications require less code, use more natural data modelling and code bases are easier to maintain.

A programme is software, with sequence of written stored instructions which can be executed to perform a special task with a computer.

In contrast to a relational DBMS, where a complex data structure must be flattened out to fit into tables or joined together from those tables to form the in-memory structure, object DBMSs have no performance overhead to store or retrieve a web or hierarchy of interrelated objects. This one-to-one mapping of object programming language objects to database objects has two benefits over other storage approaches. Firstly, it provides higher performance management of objects and secondly, it enables better management of the complex interrelationships between objects. This makes object DBMSs better suited to support applications such as risk analysis systems, telecommunications service applications, World Wide Web (WWW) document structures, design and manufacturing systems which have complex relationships between data. Main problem of object-oriented model is the implicit uncertainty of geographical ideas; therefore, it is difficult to represent them in rigidly bounded datasets. There is also no theoretical base or standard query language for object oriented model.

## 6.7 NON-SPATIAL DATA QUERY

Non-spatial data query retrieves a data subset by operating with non-spatial (attribute) data. During the non-spatial data query, the selected data subset can be studied in the table, displayed in figures, maps or charts and linked to the certain highlighted features in the map. We can also save the selected data subset in the system for further processing. For non-spatial data query, we have to make the use of expressions which should be interpretable by a GIS or DBMS. One of the widely used expressions is structured query language. Let us discuss this expression in detail.

### 6.7.1 Structured Query Language Model

Structured Query Language (SQL) is a data query and manipulation language with designed for relational databases (Chang, 2010). SQL was originally developed by IBM in the 1970s and later many commercial DBMSs like Microsoft, Oracle, Access, Informix and other have adopted SQL. SQL is a standard sublanguage for querying tables. It involves the use of tables which should be linked before the query. Language becomes much more powerful when tables are linked by using common keys and as a result, the processing of much more complex queries that involve multiple tables are possible. More complex methods of table interrogation include the ability to average the values of an attribute across selected records and to create new attributes through arithmetic operations on existing ones (e.g., create a new attribute equal to the ratio of two selected attributes).

Sublanguage is a part of computer language basically used to define or manipulate the structure and contents of a RDBMS.

To use SQL to access a database, we should follow the structure (also known as syntax) of the query language. The fundamental syntax of SQL is given below:

*select*<attribute list>

*from*<relation>

*where*<condition>

Note that the keywords are in italic.

The keyword '*select*' chooses field or fields and the keyword '*from*' chooses table or tables from the database whereas the keyword '*where*' specifies the condition for data query.

SQL model is applicable to query local and external databases. SQL model only works in a GIS package who has already prepared the keywords *select*, *from* and *where*.

SQL programming is relatively complex and requires skills for programming. SQL statements can be divided into two categories:

- Data Definition Language (DDL) which is used for creating tables, relationships and other structures and
- Data Manipulation Language (DML) which is used for queries and data modification.

**Check Your Progress III**

***Spend 5 mins***

1) What are conventional database models?

.....................................................................................................

.....................................................................................................

.....................................................................................................

## 6.8 ACTIVITY

1) Check the hard disk of your computer and see how files are stored in different folders and subfolders.

2) Visit any drug shop and observe how they arrange the medicines in different stacks or shelf and relate this data in a GIS.

## 6.9 SUMMARY

In this unit, you have learnt the following:

- There are two types of databases: spatial and non-spatial. Spatial data carries spatial multi-dimensional information and non-spatial data carries one-dimensional attribute information. Information organises on the basis of data, relationship, operating system and application architecture perspectives.
- There are four main database models which are hierarchical, network, relational and object-oriented. The relational data are organised by records in relations which resemble a table, network data are organised by records which are classified into record types, hierarchical data are organised by records on a parent-child one-to-many relations and object-oriented data are uniquely identified as individual objects that are classified into object types or classes.
- Structured query language model is one of best expressions used for query in non-spatial data.

*Spend 30 mins*

## 6.10 UNIT END QUESTIONS

1) What is data? How it is different from information?

2) What is database? Explain its types and characteristics.

3) What do you mean by geographic data?

4) Explain how the database is organised?

5) Differentiate between relational and hierarchical database models.

## 6.11 REFERENCES


- Chang, K.- t. (2010), *Geographic Information System*, Tata McGraw-Hill, New Delhi.
- Codd, E. (1970), A relational model for large shared data banks. *Communications of the Association for Computing Machinery*, vol 13, no 6, pp 377-87.


## 6.12 FURTHER/SUGGESTED READING


- Fazal, S. (2008), *GIS Basics*, New Age International Publishers, New Delhi.
- Longley, P. A., Goodchild, M. F., Maguire, D. J. and Rhind, D. W. (2001), *Geographical Information Systems and Science*, Wiley Publication, Chichester.

## 6.13 ANSWERS

### Check Your Progress I

1) Non-spatial data is the information which is independent of all geometric considerations. It is unidimensional and independent when compared with spatial data.

### Check Your Progress II

1) Database processing is associated with database management system which is used in solving operation/production needs of an organisation. It helps in day-to-day operations of business and also in decision making.

### Check Your Progress III

1) The conventional database models are of four types such as relational, network, hierarchical and object-oriented.

### Unit End Questions

1) Refer to section 6.2
2) Refer to section 6.3
3) Refer to section 6.4
4) Refer to section 6.5
5) Refer to section 6.6

## GLOSSARY

**Area :** A closed two-dimensional shape defined by its boundary.

**Coverage :** A topological vector data format.

**Data :** Any collection of related facts arranged in a particular format; often, the basic elements of information that are produced, stored, or processed by a computer.

**Database :** One or more structured sets of persistent data, managed and stored as a unit and generally associated with software to update and query the data. A simple database might be a single file with many records, each of which references the same set of fields.

**Data structure :** Is the format of the data as stored and manipulated on the computer.

**DBMS (Database Management System) :** Software designed to access and structure a database.

**Line :** A shape having length and direction but no area, connecting at least two x, y coordinates. Lines represent geographic features too narrow to be displayed as an area at a given scale, such as contours, street centerlines, or streams, or linear feature, with no area, such as state and district boundary lines.

**Models :** Are the abstract representation of the real world in various forms like pictorial/graphical/ sculpture.

**Network database :** A database that is based on the built-in connections across tables.

**Object-oriented data model :** A data model that uses objects to organise spatial data. An object is an entity such as a land parcel that has a set of properties and can perform operations upon requests.

**Point :** A single x, y coordinate that represents a geographic feature too small to be displayed as a line or area at that scale.

**Raster data model :** It is a computer based representation of spatial or geographical data which are saved as grid cells evenly spaced as rows and columns.

**Spaghetti data :** It is a non-topological vector data model, where spatial relations are not possible.

**Spatial data :** Information about the locations and shapes of geographic features, and the relationships between them; usually stored as coordinates and topology.

**SQL (Structured Query Language) :** A standard language interface to relational database management systems.

**Topology :** It can be defined as the organisation of spatial relationships between features in a GIS.

**Vector data model :** It is a computer based representation of spatial or geographical data which are saved as coordinates where every feature is a discrete object.

# ABBREVIATIONS

| | | |
|---|---|---|
| **ASCII** | : | American Standard Code for Information Interchange |
| **CAD** | : | Computer Aided Design |
| **CGIS** | : | Canadian Geographic Information Systems |
| **CPU** | : | Central Processing Unit |
| **DDL** | : | Data Definition Language |
| **DEM** | : | Digital Elevation Model |
| **DBMS** | : | Database Management System |
| **DGPS** | : | Differential Global Positioning System |
| **DIME** | : | Dual Independent Map Encoding |
| **DML** | : | Data Manipulation Language |
| **ESRI** | : | Environmental Systems Research Institute |
| **GBF** | : | Geographic Base File |
| **GIRAS** | : | Geographical Information Retrieval and Analysis System |
| **GIS** | : | Geographical Information System |
| **GPS** | : | Global Positioning System |
| **GRASS** | : | Geographic Resources Analysis Support System |
| **GUI** | : | Graphical User Interface |
| **IGIS** | : | Integrated GIS and Image Processing Software |
| **IMS** | : | Information Management System |
| **ISRO** | : | Indian Space Research Organisation |
| **IIRS** | : | Indian Institute of Remote Sensing |
| **IIT** | : | Indian Institute of Technology |
| **JIS** | : | Joint Information System |
| **LAMIS** | : | Local Authority Management Information System |
| **NRSC** | : | National Remote Sensing Centre |
| **OGC** | : | Open Geospatial Consortium |
| **OODB** | : | Object-Oriented Database |
| **OOPL** | : | Object-Oriented Programming Language |
| **PC** | : | Personal Computer |
| **RLE** | : | Run-Length Encoding |
| **SAC** | : | Space Applications Centre |
| **SLDB** | : | Swedish Land Data Bank |
| **SQL** | : | Structured Query Language |
| **TIGER** | : | Topologically Integrated Geographic Encoding and Referencing |
| **WWW** | : | World Wide Web |