

COPPERBELT UNIVERSITY

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

CS 120 – Introduction to Computer Systems

Unit 2: The File Concept

Introduction

Computers can store information on several different storage media, such as magnetic disks, magnetic tapes and optical disks. These storage devices are usually non-volatile, so the contents are persistent through power failures. A computer file can be thought of much like a traditional file that one would find in an office's file cabinet.

What is a Computer File?

A file is an organized collection of data. A file could also be defined as a named collection of related information that is recorded on secondary storage. Alternatively, files are named collections of stored data. In computer terminology, an employee file, for example, consists of a collection of data about all employees; a stock file consists of a collection of data about all stock items, and so on.

A file can also be defined as an object on a computer that stores [data](#), [information](#), settings, or commands used with a computer [program](#). In a [GUI](#), such as [Microsoft Windows](#), files display as [icons](#) that relate to the program that opens the file. A file can also be defined as a collection of data stored in one unit, identified by a [filename](#). It can be a document, picture, audio or video stream, data library, [application](#), or other collection of data. A file therefore consists of a number of **records**.

A **record** is the collection of data pertaining to one item or individual. It consists of a number of fields, which each hold one piece of data such as a name, date of birth etc. Some fields will hold **characters**, and some fields will be **numeric** and used in calculations, such as price or quantity of stock.

Examples of Files

- Documents include text files, such as a Word documents, RTF (Rich Text Format) documents, [PDFs](#), Web pages, and others.
- Pictures include [JPEGs](#), [GIFs](#), [BMPs](#), and layered image files, such as Photoshop documents (PSDs).
- Audio files include [MP3s](#), AACs, WAVs, [AIFs](#), and several others.
- Video files can be encoded in [MPEG](#), MOV, WMV, or DV formats, just to name a few.

Primary and Secondary Keys

Each record usually has some way of identifying it uniquely, with one of the fields within the record acting as a **primary key field**. Thus, each record in a file must have a different primary key in order to distinguish it from all the other records. Sometimes there is no one field which is unique to each

record and a second field is used as part of the primary key, which is then called a **composite primary key**.

In addition to having one primary key, records may have one more **secondary keys**. These are fields which are not unique to each record, but which may be put in an index and used to retrieve records. For example, in a stock file, the primary key might be a unique stock number, and secondary keys could be supplier and type of item.

File Organisation and Access

However data is stored on the computer, it must be capable of being retrieved quickly for inspection or processing. This is achieved through the use of files. A file may be described in terms of its “structure” and in terms of its “organization”. Its **structure** is determined by which data items are included in records and how the data items are grouped within records. Its **organization** is determined by how the records are arranged within the file.

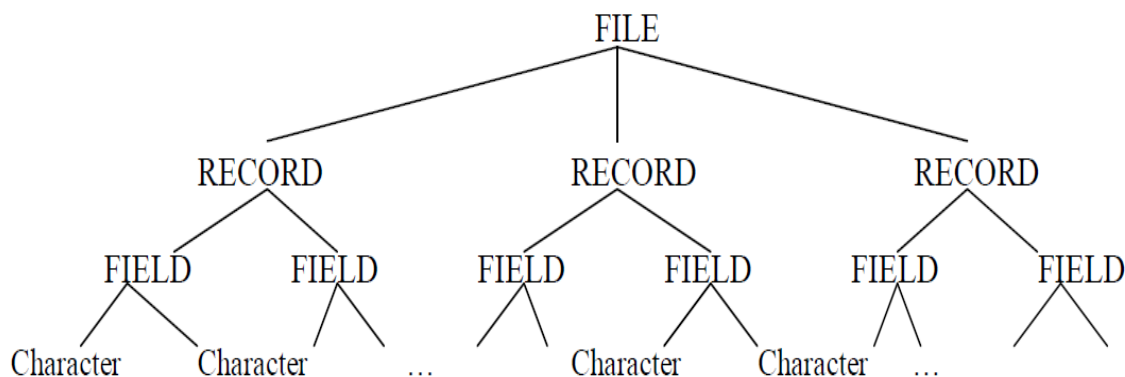
There are two common ways of viewing files:

- *Logical files*. A “logical file” is a file viewed in terms of *what* data items its records contain and *what* processing operations may be performed upon the file. The user of the file will normally adopt such view.
- *Physical files*. A “physical file” is a file viewed in terms of *how* the data is stored on a storage device such as a magnetic disk and *how* the processing operations are made possible.

Elements of a Computer File

A file consists of a number of *records*. A record is the collection of data pertaining to one item or individual. Each record is made up of a number of *fields*, which each hold one piece of data such as name, date of birth etc. Each field consists of a number of *characters*. A character is the smallest element in a file and can be alphabetic, numeric or special. An item of data within a record is called a field – it is made up of a number of characters, for example, a name, a date, or an amount. A record is made up of a number of related fields, for example, a customer record or an employee payroll record.

The data structure of a file forms part of a data hierarchy as follows:



TYPES OF FILES

Master file. Master files are fairly permanent files kept up to date by applying the transactions that occur during operation of the business. They contain generally two basic types of data:

- Data of a more or less permanent nature such as on a payroll file, name address, rate of pay etc.
- Data which will change every time transactions are applied to the file – for example, gross pay to date, tax paid to date, etc.

Transaction or Movement file. Transaction files contain details of all transactions that have occurred in the last period. A period may be the time that has elapsed since business started that day, or it may be a day, a week, a month or more. The transactions are usually created from source documents. For example, a sales transaction file may contain details of all sales made that day. Once the data has been processed it can be discarded (although backup copies may be kept for a while). This file will be used to update the master file.

Reference file. A reference file has a reasonable amount of permanency. Examples of data used for reference purposes are price lists, tables of rates of pay, names and addresses.

Once having created files of data in magnetic form one must have the means of access to particular records within those files. In general terms this is usually done by giving each record a “key” field and the record will be recognized or identified by that particular key. Examples of key fields are:

- Customer number in a customer ledger record.
- Stock code number in a stock record.
- Employee number in a payroll record

Not only does the key field assist in accessing records but the records themselves can, if required, be sorted into the sequence indicated by the key.

Characteristics of Files

Hit rate: This is the term used to describe the rate of processing of master files in terms of active records.

- For example if 1,000 transactions are processed each day against a master file of 10,000 records, then the hit rate is said to be 10% (that is $1,000/10,000 \times 100$). Or if on a payroll run if 190 out of 200 employees were to be paid the hit rate would be 95% (that is $190/200 \times 100$).
- Hit rate is a measure of the “activity” of the file.

Volatility: This is the frequency with which records are added to the file or deleted from it.

- Sequential organization is suitable here because the whole file is copied during processing, and new records are automatically incorporated and deleted records left out during copying.
- If the frequency is high the file is said to be volatile. A file which is not altered is “static”.
- If the frequency is low the file is said to be “semi-static”.

Size: This is the amount of data stored in the file. It may be expressed in terms of the number of characters or number of records.

Growth: Files often grow steadily in size as new records are added. Growth must be allowed for when planning how to store a file.

Types of File Organization

The available methods include:

- ❖ Serial
- ❖ Sequential
- ❖ indexed sequential
- ❖ random

1. SERIAL FILE ORGANISATION AND ACCESS

Records are “written” in the first place from main storage onto the tape by the processor. Each record is written onto tape in response to a “write instruction”. This process will be repeated until all the required records are written onto the tape, that is, until the file is complete. In diagrammatic form the tape will look as illustrated below:

Cust. 1	Inter Record Gap	Cust. 2	Inter Record Gap	Cust. 3	Inter Record Gap	Cust. 4	...
------------	------------------------	------------	------------------------	------------	------------------------	------------	-----

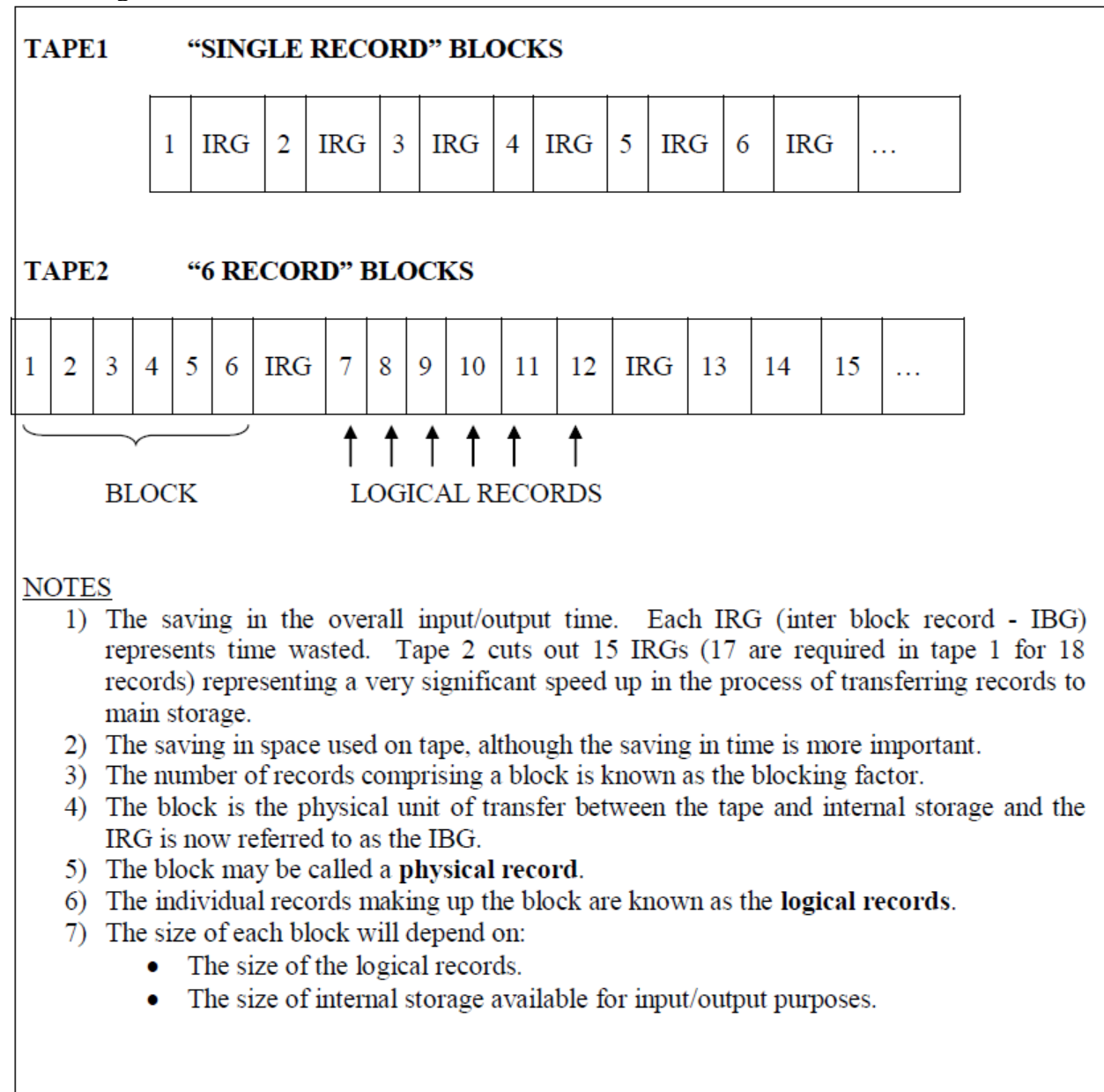
Note that the inter-record gap (IRG) which is caused by the tape slowing down at the conclusion of one ‘write’ and accelerating at the beginning of the next.

The significant feature with regard to the usage of magnetic tape as a storage medium is the fact that it is a serial access medium. The records on a serial file are not in any particular sequence, and so this type of organization would not be used for a master file as there would be no way to find a particular record except by reading through the whole file, starting at the beginning, until the right record was located. Serial files are used as temporary files to store transaction data.

BLOCKING RECORDS

The gaps created between each record on tape represent wasted space but more importantly they represent unproductive time spent by the tape unit in slowing down and accelerating in between each write and read operation. In order to reduce the number of IRGs and thus speed up the total time to process a tape file the technique of blocking records is adopted. Thus a single “read” or “write” instruction will cause a number of records to be read or written.

See the figure below:



ACCESS

The only way to access a serial file on tape is serially. This simply means to say that each record is read from the tape into main storage one after the other in the order they occur on the tape.

ADDING AND DELETING RECORDS ON A SERIAL FILE

If one or more new records have to be added to a serial file, there is no problem; the new records can simply be appended to the end of the file. Deleting a record is more complex. To find the record to be deleted, the computer has to read the tape from the beginning; but once it has found it, it cannot back up and ‘wipe’ just that portion of the tape occupied by the record, leaving a blank space. The technique therefore, is to create a brand-new tape, copying over all the records up to

the one to be deleted, leaving that one off the new tape, and then copying over all the rest of the records.

USE OF SERIAL FILES

Serial files are normally only used as transaction files, recording data in the order in which events take place; for example, sales in a shop, customers taking cash from a cash machine (ATM), orders arriving at a mail order company.

Summary on Serial File Organization

Serial files are stored in chronological order, that is as each record is received it is stored in the next available storage position. In general, it is only used on a serial medium such as magnetic tape. This type of file organisation means that the records are in no particular order and therefore to retrieve a single record the whole file needs to be read from the beginning to end. Serial organisation is usually the method used for creating Transaction files (unsorted), Work and Dump files.

2. SEQUENTIAL FILE ORGANISATION AND ACCESS

Sequential files are serial files whose records are sorted and stored in an ascending or descending on a particular key field. A sequential file is a serial file that has been given a logical order. The order of records in a sequential file is usually by key field. Sequential files are used as master files for high hit rate applications such as payroll.

ADDING AND DELETING RECORDS ON A SEQUENTIAL FILE

With a sequential file, all the records on the tape (or disk) are in order, perhaps of employee number, so just adding a new record on the end is not good at all. Of course, the records could then be sorted but sorting is a very time-consuming process. The best and 'correct' way is to make a new copy of the file, copying over all records till the new one can be written in its proper place, and then copying over the rest of the records. It's exactly as if you had just made a list on a nice clean piece of paper of all the students in the class in order of surname, and then discovered you had left out Carter, A. N. The only way to end up with a perfect list is to copy it out again, remembering to include Carter this time.

Deleting a record is exactly the same as for serial organization

Summary on Sequential Files

- ❖ Records are stored in key sequence
- ❖ Adding/Deleting requires making a new file
- ❖ Used as master file

3. INDEXED SEQUENTIAL FILE ORGANISATION AND ACCESS

Indexed Sequential file organisation is logically the same as sequential organisation, but an index is built indicating the block containing the record with a given value for the Key field. This method combines the advantages of a sequential file with the possibility of direct access using the Primary Key (the primary Key is the field that is used to control the sequence of the records). These days manufacturers providing Indexed Sequential Software allow for the building of indexes using fields other than the primary Key. These additional fields on which indexes are built are called Secondary Keys. Accessing of records is very fast, if the index is properly organized.

There are three major types of indexes used:

Basic Index: This provides a location for each record (key) that exists in the system.

Implicit Index: This type of index gives a location of all possible records (keys) whether they exist or not.

Limit Index: This index groups the records (keys) and only provides the location of the highest key in the group. Generally, they form a hierarchical index.

Data records are blocked before being written to disk. An index may consist of the highest key in each block, (or on each track).

Index	Data	
1 A0025	A0012	Block 1
2 A0053	A0017	
3 A0075	A0025	
	A0037	Block 2
	A0038	
	A0053	
	A0064	Block 3
	A0073	
	A0075	

The Block Index

In the above example, data records are shown as being 3 to a block. The index, then, holds the key of the highest record in each block. (An essential element of the index, which has been omitted from the diagram for simplicity, is the physical address of the block of data records). Should we wish to access record 5, whose key is A0038, we can quickly determine from the index that the record is held in block 2, since this key is greater than the highest key in block 1, A0025, and less than the highest key in block 2, A0053. By way of the index we can go directly to the record we wish to retrieve, hence the term "direct access".

Disadvantage: Extra storage space is required for the index.

Summary on Indexed Sequential File Organisation

- ❖ Each record of a file has a key which uniquely identifies that record
- ❖ An index consists of keys and addresses.
- ❖ An indexed sequential file is a sequential file (i.e. sorted in order of key) which has an index.
- ❖ A full index to a file is one which has an entry for every record.

4. RANDOM OR RELATIVE FILE ORGANISATION AND ACCESS

This device, that is the magnetic disk (e.g. hard disk), is a direct access medium and therefore has a distinct advantage over tape in this respect. Direct access is also called Random access because it means access is in no set order.

A random file (also called a hash file, direct or relative file) has records that are stored and retrieved according to either their disk address or their relative position within the file. This means that the program which stores and retrieves the records has first to specify the address of the record in the file.

This is done by means of an algorithm or formula which transforms the record key into an address at which the record is stored. In the simplest case, record number 1 will be stored in block 1, record number 2 in block 2 and so on. This is called relative file addressing, because each record is stored at a location given by its key, relative to the start of the file.

Summary

A randomly organized file contains records arranged physically without regard to the sequence of the primary key. Records are loaded to disk by establishing a direct relationship between the Key of the record and its address on the file, normally by use of a formula (or algorithm) that converts the primary Key to a physical disk address. This relationship is also us