Time, date and user id : This information may be kept for creation, last modification and last use. These data can be useful for protection, security and usage monitoring

* File Organization:

File Organization is refers to the logical structuring of the records as determined by the way in which they are accessed. The physical organization of the file on secondary storage depends on the blocking strategy and the file allocation strategy.

In choosing a file organization, several criteria are important:

i) Rapid access
ii) Ease of update
iii) Economy of storage
iv) Simple maintenance
v) Reliability

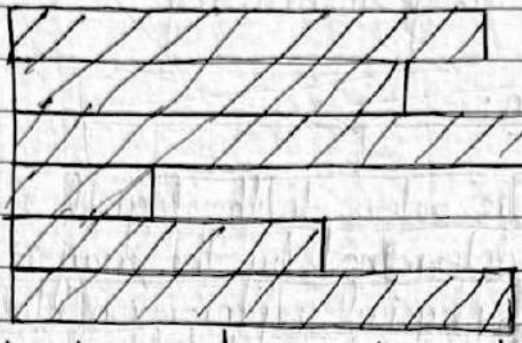
The relative priority of these criteria will depend on the applications that will use the file. The five organizations are as follows:

i) Pile
ii) The sequential file
iii) The indexed sequential file
iv) The indexed file
v) The direct or hashed file.

i) Pile:

The least-complicated form of file organization may be termed the pile. Data are collected in the order in which they arrive. Each record consists of one burst of data. The purpose of the pile is simply to accumulate the mass of data and save it. Because there is no structure to the pile file, record access is by exhaustive search. Pile files are encountered when data are collected and stored prior

to processing or when data are not easy to organize. This type of file uses space well when the stored data vary in size & structure.



Variable length records, Variable set of fields & Chronological order

fig: File File.

ii) Sequential File:

In this type of file, a fixed format is used for records. All records are of the same length, consisting of the same number of fixed length fields in a particular order. One particular field, usually the first field in each record, is referred to as the key field. The key field uniquely identifies the record thus key values for different records are always different. Access requires the sequential search of the file for a key match.

Fixed length records, Fixed set of fields in fixed order & sequential order based on key field.

fig: Sequential file.

iii) Indexed Sequential file

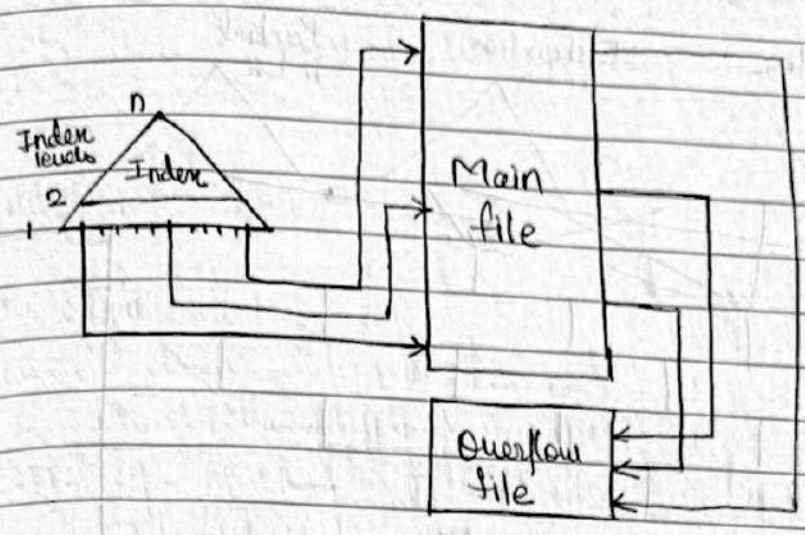


fig: Indexed sequential file.

The indexed sequential file maintains the key characteristics of the sequential file: Records are organized in sequence based on a key field.

Two features are added: an index to the file to support random access & an overflow file.

The index provides a lookup capability to reach quickly the vicinity of a desired record. The overflow file is similar to the log file used with a sequential file but is integrated so that records in the overflow file are located by following a pointer from their predecessor record.

In the simplest indexed sequential structure, a single level of indexing is used. The index in this case is a simple sequential file. Each record in the index file consists of two fields: a key field, which is the same as the key field in the main file, and a pointer into the main file. To find a specific field, the index is searched to find the highest key value that is equal to or precedes the desired key value. The search continues in the main file at the location indicated by the pointer.

The indexed sequential file greatly reduces the time required to access a single record. To provide even greater efficiency in access, multiple levels of indexing can be used.

iv) Indexed file:

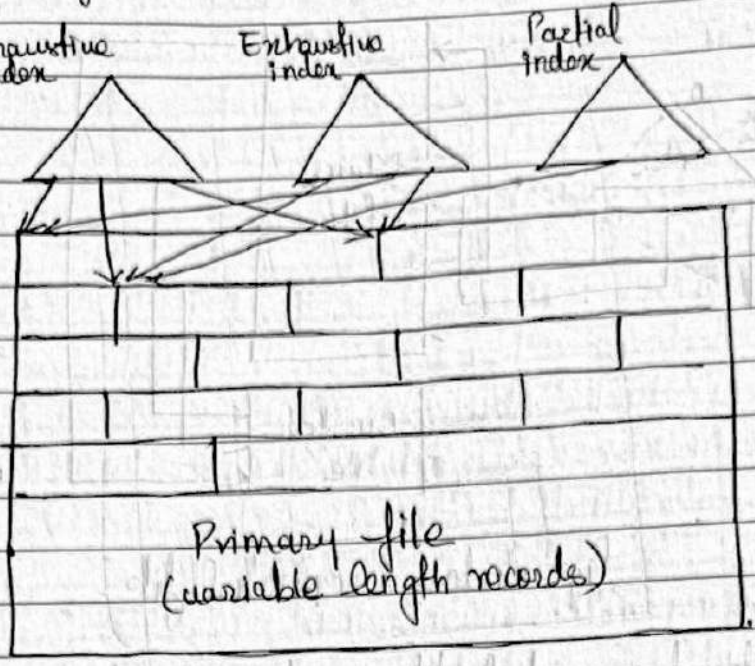


fig: Indexed file.

The index sequential file retains one limitation of the sequential file: Effective processing is limited to that which is based on a single field of the file. When it is necessary to search for a record on the basis of some other attribute than the key field, both formats of sequential file are inadequate. For some applications, this flexibility is desireable.

To achieve this flexibility, a structure is needed that employs multiple indexes, one for each type of field that may be the subject of a search. Two types of indexes are used. An exhaustive index contains one entry for every record in the main file.

The index itself is organized as a sequential file for ease of searching. A partial index contains entries to records where the field of interest exists.

v) Direct or Hashed file:

The direct or hashed file exploits the capability found on disks to access directly any blocks of a known address. There is no concept of sequential ordering here.