

File Organization

Dr. Manzoor Ahmad Chahoo
Scientist D
Department of Computer Science

University of Kashmir

27-04-2020

Outline

- 1 Introduction
 - Factors
- 2 File Organizations
- 3 Relative File Organization
- 4 Indexed Sequential File Organization
- 5 Multi Key File Organization

File Organization

- File Organization refers to the way records are physically arranged on a storage device.
- There may be a single or a multiple key associated with a file .

Objective

Files are classified based on the Physical Storage and the keys used to access records.

Factors that influence the choice of a particular file organization

- Economy of Storage
- Convenience of updates
- Ease of retrieval
- Reliability
- Security
- Integrity

Sequential File Organization

The records are written one after the other in the order when the file is created and can be accessed only in that order in which they are written when the file is used for input.

In some cases, records of a sequential file are sorted by value of some field in each record.

The field whose value is used to sort the records is known as sort key. A file can be sorted in ascending or descending order by a sort key comprised of one or more fields.

Sequential File Organization - contd..

Processing requirement determine the order for sorting the records. Systems based on Sequential files tend to replicate records across files that are in different sort orders. Sequential files are most commonly used in batch processing than in interactive processing.

Sequential File Organization - Advantages

- Easy to handle
- Involves no overheads
- can be stored on tapes as well as disks
- well suited for batch oriented applications
- records can be of varying length

Sequential File Organization - DisAdvantages

- Records can be accessed only in a particular sequence.
- Doesnot support updation operation in place
- Deosnot support interactive applications.

Relative File Organization

- Effective way to access individual records directly.
- There is a predictable relation between the key and the records location in the file, However the logical ordering of records bear no relationship to their physical sequencing.
- The records donot necessarily appear physically in sorted order of their keys.
- A relationship which is basically amapping function (R) between the keyvalue and the physical address is designated.

Relative File Organization

- When a record is to be written to a relative file, the mapping function (R) is used to translate the records key to an address which indicates where the record is to be stored.
- When the record for a particular key is to be retrieved, the mapping function (R) is applied to that key value translating it to address where the record can be found.
- A relative file can also be accessed sequentially but in that case the keys may not be in logical sequence.

Mapping functions

- Directory Lookup.
Directory of {keyvalue : address} pairs.
In order to find a record in a relative file , one locates its key value in the directory and then the indicated address is used to find the record on storage device.
- Address Calculation Technique
Perform hashing on the key value such that the result is a relative address. The idea is to apply a function that translates a relatively large domain of possible key values to a relatively small range of relative address space.

Advantages

- Records can be accessed out of sequence.
- Well suited for interactive online applications.
- supports updation operation in place.

Dis-Advantages

- Can only be stored on disks.
- Involves more overhead in the form of maintenance of indices.
- Complex in handling.
- Records can be of fixed length.

Background

- Uses combinations of access types supported by sequential and a relative file
- Useful when there is a need both to access sequentially by some key value and also to access the records individually by the same key
- used frequently to support applications that require both batch and interactive processing.

Structure

- Index with pointers in a sequential file
- Index has been structured as a binary search tree, The index is used to service a request for access to a particular record and the sequential data file alone is used to support sequential access to the entire collection of records.

Advantages

- Records can be accessed sequentially as well as randomly.
- supports interactive as well as batch processing applications.
- supports updation operation in place.

Dis-Advantages

- Can only be stored on disks.
- involves more overhead in the form of maintenance of indices.
- Records are of fixed length.

Background

- Some Applications require support for multi-key files.
- A situation where many users need to access the same data in different ways.
- One solution is to use several files, each one organized to serve one type of request for all these type of access but this will lead to wastage of storage space as such is not desirable.
- The alternative is multi key file organizaiton.

Definition

A multi key file organizaiton has only one data file and set of indices to provide different access paths to data records of the file.

Inverted File Organization

- An approach to provide linkage between index and a file is called inversion.
- A keys inversion index contains all the values that the key presently has in the records of the file..
- Each key-value entry in the inversion index points to all data records that have the corresponding value.
- The file is said to be inverted on that key.

Inverted File Organization

- A completely inverted file has an inverted index for every data field.
- If a file is not completely inverted but has at least one inversion index. it is called partially inverted file
- Used in many commercially available DBMS such IBM DB2, Oracle and INGRES.

Background

- Another approach to provide linkage between index and a file
- A multi list file maintains an index for each secondary key.
- There is one entry in the secondary key index for each value that the secondary key presently has in the file.

Difference

- Multi-List file organization differs from inversion in the sense that entry in the inversion index for a key value has a pointer to each data record with the key value
- The entry in the multi list index for a key has just one pointer to the first record with that key value which in turn points to the next record with that key value and so forth.
- Thus there is a linked list of records for each value of the secondary key.
- Multi list chains usually are bidirectional and occasionally are circular to improve update performance.

Difference

- Multi-List file organization differs from inversion in the sense that entry in the inversion index for a key value has a pointer to each data record with the key value
- The entry in the multi list index for a key has just one pointer to the first record with that key value which in turn points to the next record with that key value and so forth.
- Thus there is a linked list of records for each value of the secondary key.
- Multi list chains usually are bidirectional and occasionally are circular to improve update performance.