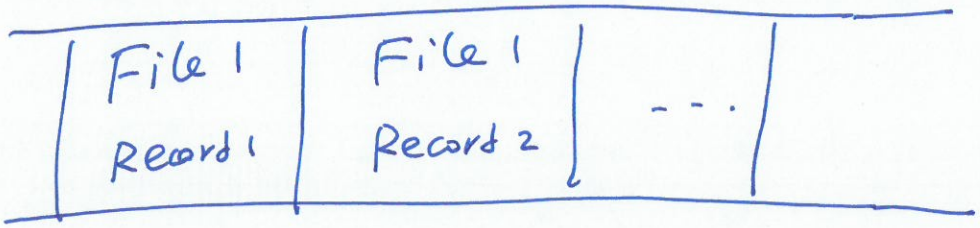
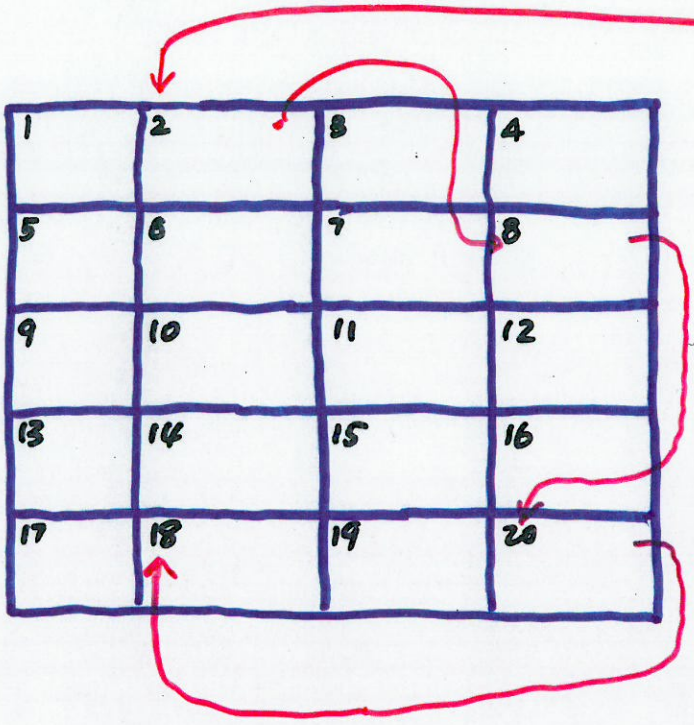
3. Physical Storage Allocation

- Contiguous Storage: records are stored one by one.



- Advantage: simple to implement & manage. Disadvantage: file cannot be expanded.

- Noncontiguous Storage: allows files to use any storage space available on the disk.

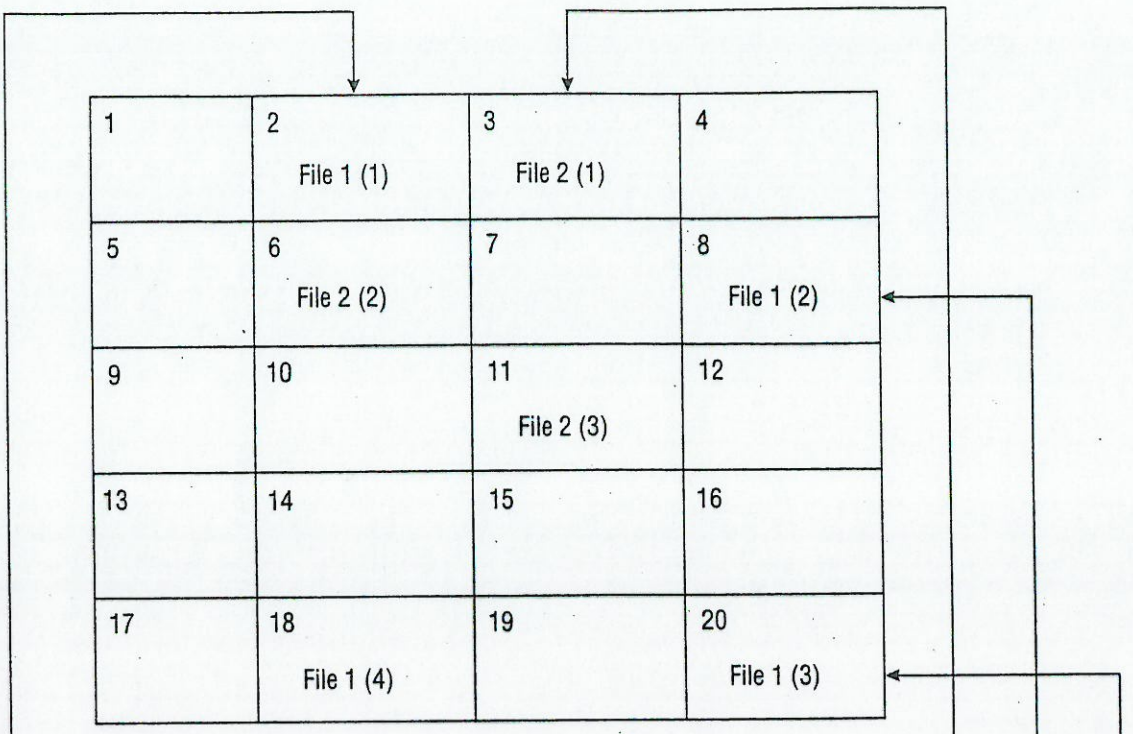


Directory

File No.	Start	End	No. extents
1	2	18	3
2	3	11	2

File 1 starts in address 2 & continues in 8, 20, 18.

Linking takes place at the storage level.

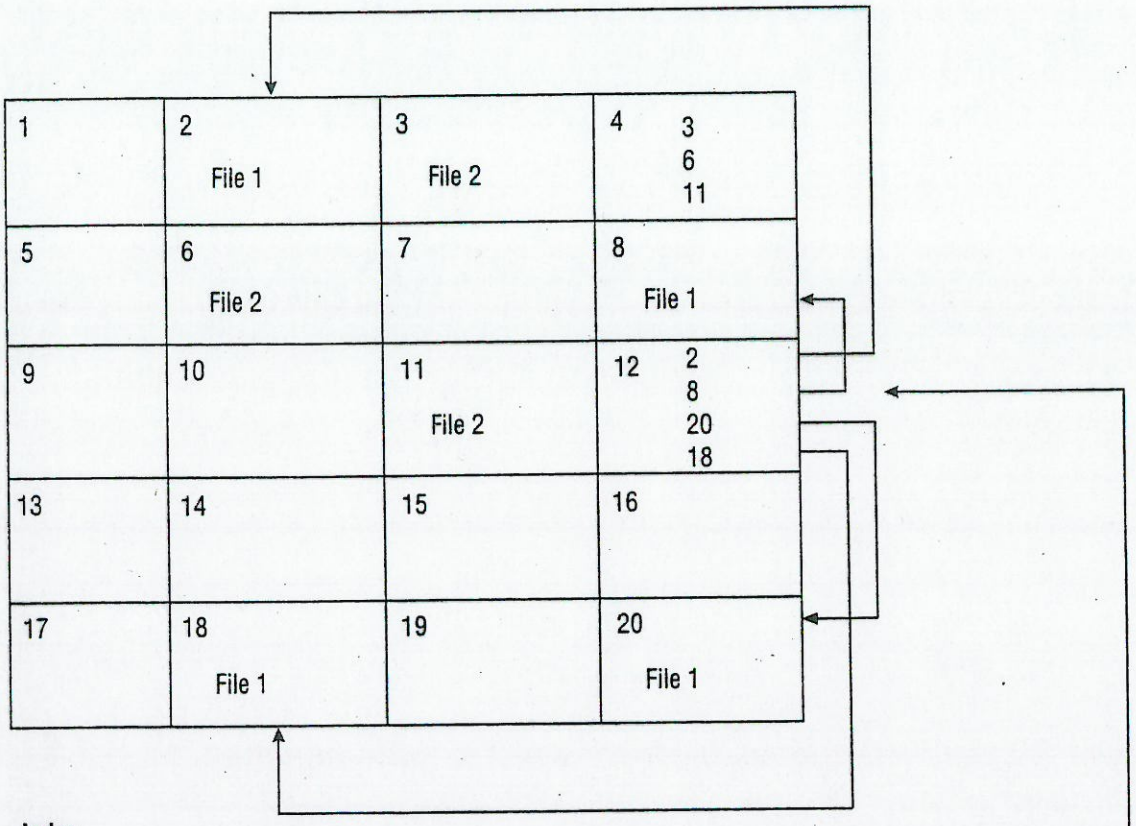


Directory

File	Address	Size	Next
	1	512	
File 1 (1)	2	512	8
File 2 (1)	3	512	6
	⋮	⋮	
File 2 (2)	6	512	11
	7	512	
File 1 (2)	8	512	20
	⋮	⋮	
File 2 (3)	11	512	-
	⋮	⋮	
File 1 (4)	18	512	-
	19	512	
File 1 (3)	20	512	18

Linking takes place at the directory level.

- Indexed Storage: allows direct record access by bringing together into an index block, the pointers linking every content of that file.



Index

File	Index Block
File 1	12
File 2	4

Indexed storage allocation

4. Data Compression

- Records with repeated characters can be abbreviated.

  Adambbbb ⟶ Adam4b

  3 00,000,000 ⟶ 3#8

- Repeated terms can also be compressed.

  Grade Point Average – GPA

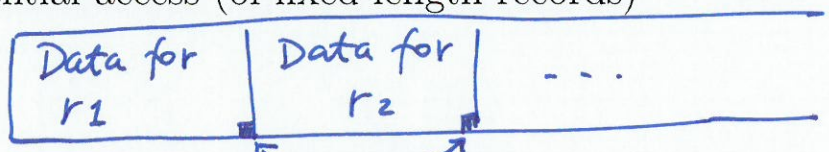
  Bachelor of Arts – BA

- Front-end compression, which is usually used in database for index compression.

Original List	Compressed List
Jack, Betty	Jack, Betty
Jackson, John	4son, John
Jackson, peter	8 peter
Jacksonn, peter	7n, peter

5. Access Methods

- Access methods are determined by a file's organization:
  - 1. Indexed sequential files are the most flexible.
  - 2. Sequential files are the least flexible.
- For sequential records, File Manager uses the **current byte address (CBA)** — address of the last byte read to access the next sequential record.
- Sequential access (of fixed-length records)



Current CBA = previous CBA + (record length)

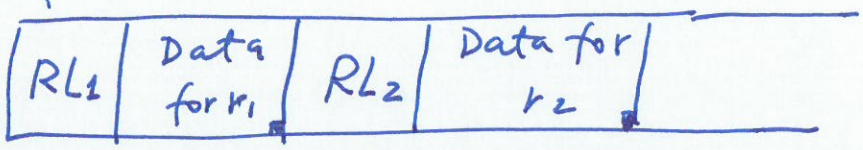
RL

- Sequential access (of variable-length records)

$$CBA_K = CBA_{K-1} + RL_K + N$$

space used for storing the length.

store length of r_1



- Direct access (with fixed-length records)

$$CBA = (\text{Record Number} - 1) \cdot RL$$

- Direct access (with variable-length records)

  - Hard. sequential search is a trivial solution.
  - Augmented list (skip list)

6. Access Control

- In 1950's a copy of FORTRAN compiler can only serve one user at one time. So at that time, we have no problem with access control.
- Five possible actions on a file
  - 1. Read only.
  - 2. Write only.
  - 3. Execute only.
  - 4. Delete only.
  - 5. Combinations.
- How to do access control?
  - 1. Access control matrix.
  - 2. Access control list.
  - 3. Capacity lists.
  - 4. Lockword.

	user1	user2	...
File1	RWED	R-E-	...
File2	RW--	--E-	...
File3	R--D	RW-D	...

Each file is entered in the list and contain names of the users who are allowed to access (with the proper access right)

File	Access
File1	user1(RWED), user2(R-E-)...
File2	user1(RW--), ...

almose like password.

List every user and the files they can access.

user	Access
user1	File1(RWED), File2(RW--), ...