UPPSALA UNIVERSITET

Informationsteknologi

Today's class

- File management

UPPSALA UNIVERSITET

Informationsteknologi

File Management

- File management system consists of system utility programs that run as privileged applications
- Input to applications is by means of a file
- Output is saved in a file for long-term storage

File Operations

- Create
- Delete
- Open
- Close
- Read
- Write

Terms Used with Files

- Field
  - Basic element of data
  - Contains a single value
  - Characterized by its length and data type
- Record
  - Collection of related fields
  - Treated as a unit
    - Example: employee record

Terms Used with Files

- File
  - Collection of similar records
  - Treated as a single entity
  - Have file names
  - May restrict access
- Database
  - Collection of related data
  - Relationships exist among elements

Typical Operations on File Records

- Retrieve_All
- Retrieve_One
- Retrieve_Next
- Retrieve_Previous
- Insert_One
- Delete_One
- Update_One
- Retrieve_Few

File Management Systems

- The way a user of an application may access files
- Programmer does not need to develop file management software

Objectives for a File Management System

- Meet the data management needs and requirements of the user
- Guarantee that the data in the file are valid
- Optimize performance
- Provide I/O support for a variety of storage device types
- Minimize or eliminate the potential for lost or destroyed data
- Provide a standardized set of I/O interface routines
- Provide I/O support for multiple users

Minimal Set of Requirements

- Each user should be able to create, delete, read, write and modify files
- Each user may have controlled access to other users' files
- Each user may control what type of accesses are allowed to the users' files
- Each user should be able to restructure the user's files in a form appropriate to the problem

UPPSALA UNIVERSITET

Informationsteknologi

Minimal Set of Requirements

- Each user should be able to move data between files
- Each user should be able to back up and recover the user's files in case of damage
- Each user should be able to access the user's files by using symbolic names

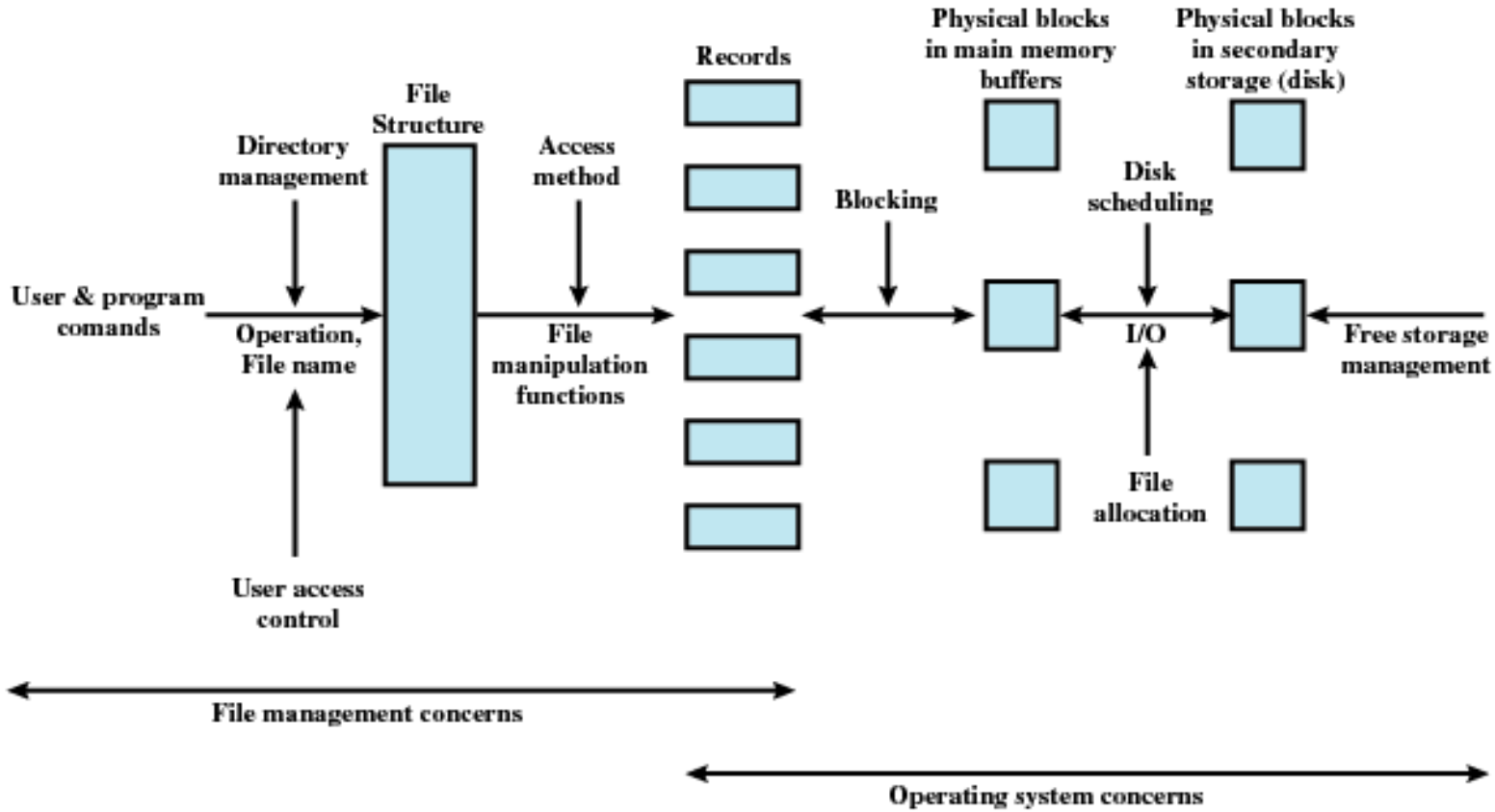


Figure 12.2 Elements of File Management

UPPSALA UNIVERSITET

Informationsteknologi

File Management Functions

- Identify and locate a selected file
- Use a directory to describe the location of all files plus their attributes
- On a shared system describe user access control
- Blocking for access to files
- Allocate files to free blocks
- Manage free storage for available blocks

File Organization

- The logical structuring of the records in a file as determined by the way in which they are accessed

Criteria for File Organization

- Short access time
  - Needed when accessing a single record
- Ease of update
- Economy of storage
- Simple maintenance
- Reliability

Five Fundamental File Organizations

- Pile
- Sequential
- Indexed sequential
- Indexed
- Direct (hashed)

The Pile

- Least complicated form of file organization
- Data are collected in the order they arrive
- Purpose is to accumulate a mass of data and save it
- Records may have different fields
- No structure
- Record access is by exhaustive search

The Sequential File

- The most common form of file structure
- Fixed format used for records
- Records are the same length
- All fields the same (order and length)
- Field names and lengths are attributes of the file
- One field is the key field
  - Uniquely identifies the record
  - Records are stored in key sequence
- New records are placed in a log file or transaction file
- Batch update is performed to merge the log file with the master file

UPPSALA UNIVERSITET

Informationsteknologi

The Indexed Sequential File

- Index provides a lookup capability to quickly reach the vicinity of the desired record
  - Contains key field and a pointer to the main file
  - Indexed is searched to find highest key value that is equal to or precedes the desired key value
  - Search continues in the main file at the location indicated by the pointer
- The index can be as simple as a sequential file itself

Comparison of Sequential and Indexed Sequential Files

- Suppose a file contains 1 million records
- On average 500,000 accesses are needed to find a record in a sequential file
- If an index contains 1000 entries, it will take on average 500 accesses to find the key, followed by 500 accesses in the main file. Now on average it is 1,000 accesses

Multiple levels of indexing

- Can have two levels of indexing
- Let the lower level index file contain 10,000 entries indexing into the main file
- Let the higher level index file contain 100 entries indexing into the lower level index
- Search begins at the higher level index
- Now have on average 50 accesses at the higher level index, 50 accesses at the lower level index, and 50 accesses in the main file, for a total of 150 accesses

New Records

- New records are added to an overflow file
- Record in main file that precedes it is updated to contain a pointer to the new record
- The overflow is merged with the main file during a batch update

The Indexed File

- Uses multiple indexes for different key fields
- May contain an exhaustive index that contains one entry for every record in the main file
- May contain a partial index

The Direct or Hashed File

- Directly access a block at a known address
- Key field required for each record

File Directories

- Contains information about files
  - Attributes
  - Location
  - Ownership
- Directory itself is a file owned by the operating system
- Provides mapping between file names and the files themselves

UPPSALA UNIVERSITET

Informationsteknologi

Simple Structure for a Directory

- List of entries, one for each file
- Sequential file with the name of the file serving as the key
- Provides no help in organizing the files
- Forces user to be careful not to use the same name for two different files

UPPSALA UNIVERSITET

Informationsteknologi

Two-level Scheme for a Directory

- One directory for each user and a master directory
- Master directory contains entry for each user
  - Provides address and access control information
- Each user directory is a simple list of files for that user
- Still provides no help in structuring collections of files

Hierarchical, or Tree-Structured Directory

- Master directory with user directories underneath it
- Each user directory may have subdirectories and files as entries
- Files can be located by following a path from the root, or master, directory down various branches
  - This is the pathname for the file
- Can have several files with the same file name as long as they have unique path names

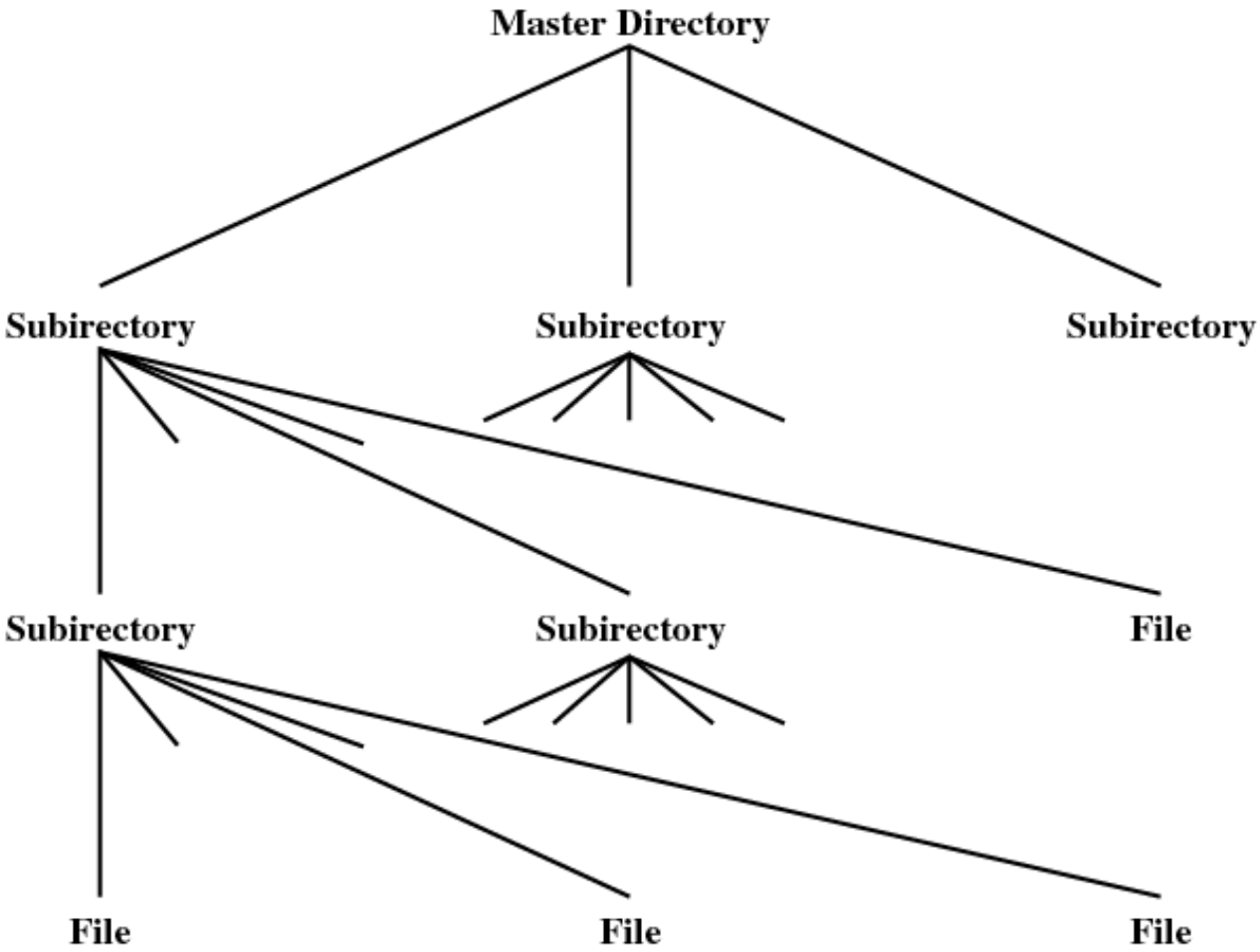


Figure 12.4 Tree-Structured Directory

UPPSALA UNIVERSITET

Informationsteknologi

File Sharing

- In multiuser system, allow files to be shared among users
- Two issues
  - Access rights
  - Management of simultaneous access

Access Rights

- None
  - User may not know of the existence of the file
  - User is not allowed to read the user directory that includes the file
- Knowledge
  - User can only determine that the file exists and who its owner is
- Execution
  - The user can load and execute a program but cannot copy it

Access Rights

- Reading
  - The user can read the file for any purpose, including copying and execution
- Appending
  - The user can add data to the file but cannot modify or delete any of the file's contents
- Updating
  - The user can modify, delete, and add to the file's data. This includes creating the file, rewriting it, and removing all or part of the data.

UPPSALA UNIVERSITET

Informationsteknologi

Access Rights

- Changing protection
  - User can change access rights granted to other users
- Deletion
  - User can delete the file
- Owners
  - Have all rights previously listed
  - May grant rights to others using the following classes of users
    - Specific user
    - User groups
    - All for public files

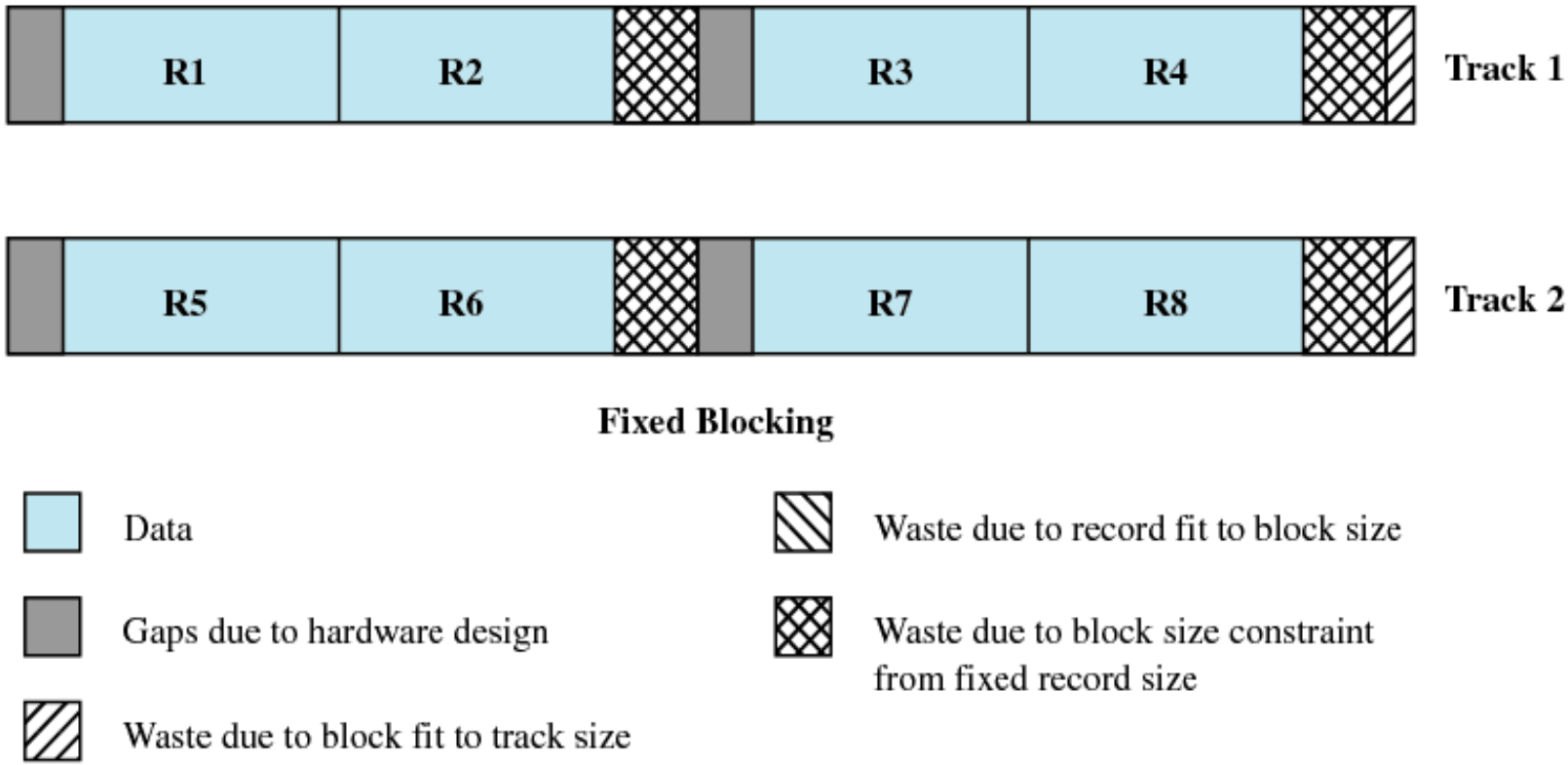
Simultaneous Access

- User may lock entire file when it is to be updated
- User may lock the individual records during the update
- Mutual exclusion and deadlock are issues for shared access

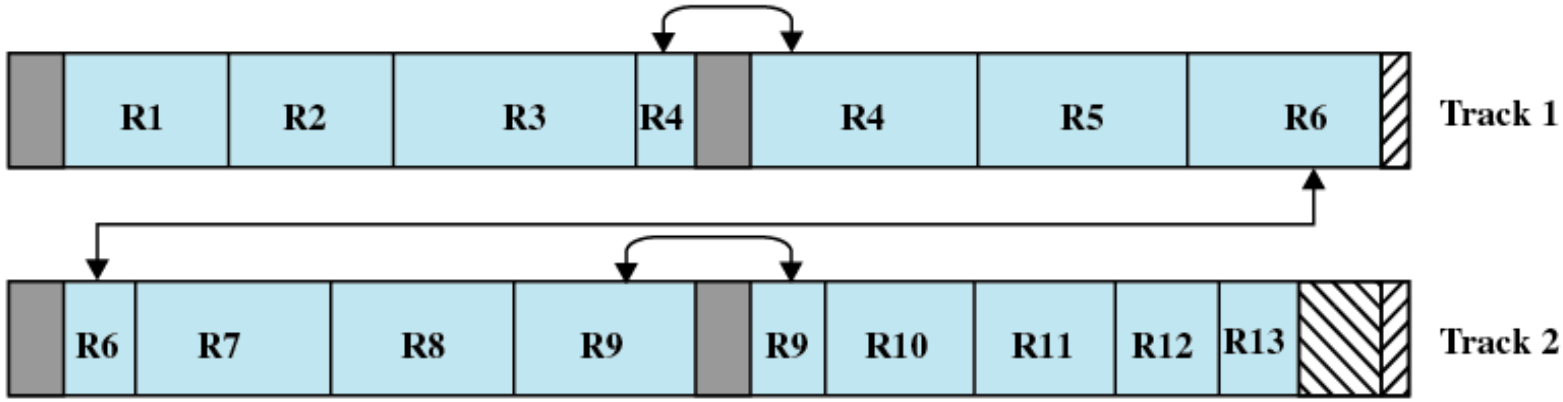
Record Blocking

- File records are organized into blocks for I/O
- Some issues
  - Fixed length or variable length
  - Block size compared to record size
- Three methods of blocking
  - Fixed blocking
  - Variable-length spanned blocking
  - Variable-length unspanned blocking

Fixed Blocking



Variable Blocking: Spanned



Variable Blocking: Spanned

Data

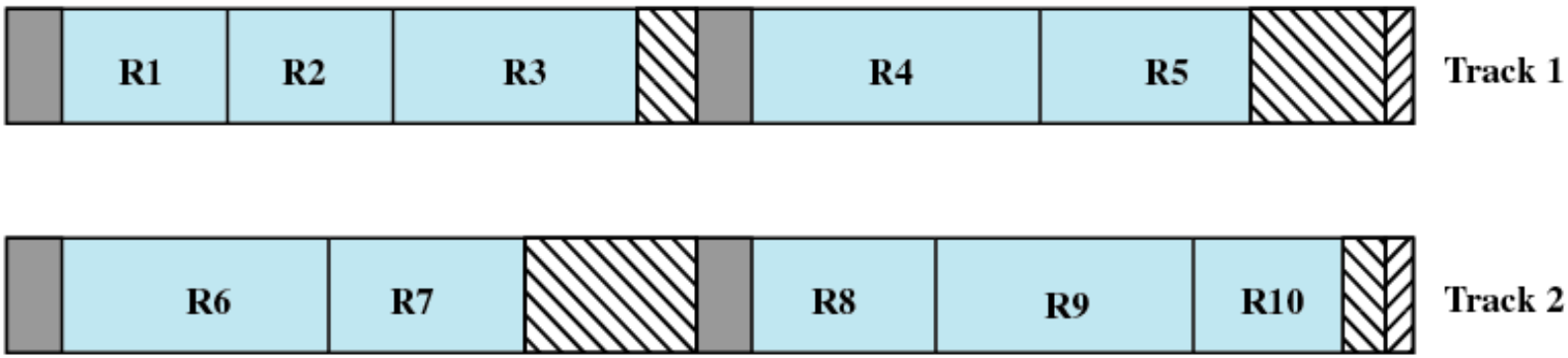
Gaps due to hardware design

Waste due to block fit to track size

Waste due to record fit to block size

Waste due to block size constraint from fixed record size

Variable Blocking Unspanned



Variable Blocking: Unspanned

- Data
- Gaps due to hardware design
- Waste due to block fit to track size
- Waste due to record fit to block size
- Waste due to block size constraint from fixed record size

Secondary Storage Management

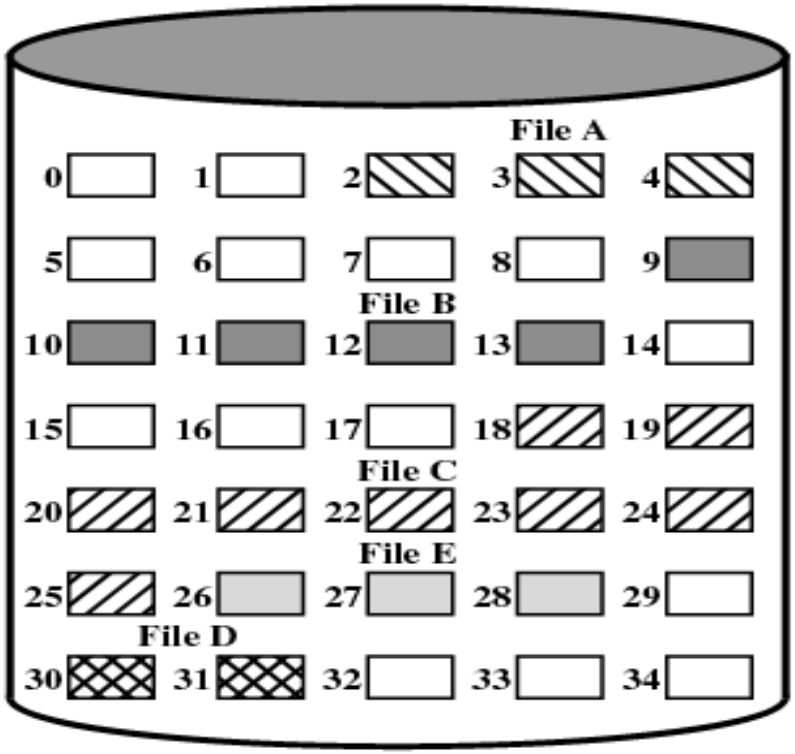
- Space must be allocated to files
- Must keep track of the space available for allocation

Preallocation

- Need the maximum size for the file at the time of creation
- Difficult to reliably estimate the maximum potential size of the file
- Tend to overestimate file size so as not to run out of space

Methods of File Allocation

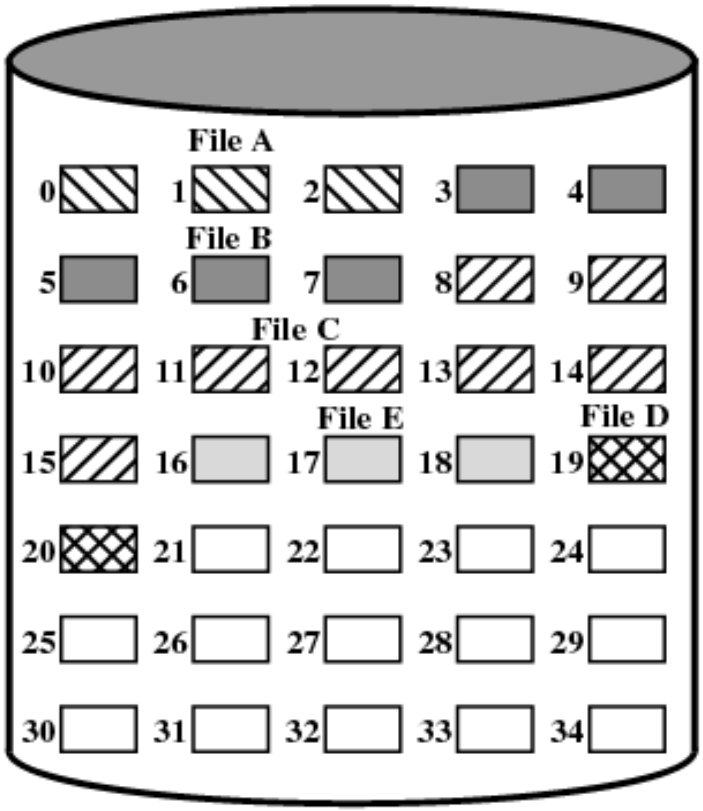
- Contiguous allocation
  - Single set of blocks is allocated to a file at the time of creation
  - Only a single entry in the file allocation table
    - Starting block and length of the file
- External fragmentation will occur
  - Need to perform compaction



File Allocation Table

File Name	Start Block	Length
File A	2	3
File B	9	5
File C	18	8
File D	30	2
File E	26	3

Figure 12.7 Contiguous File Allocation



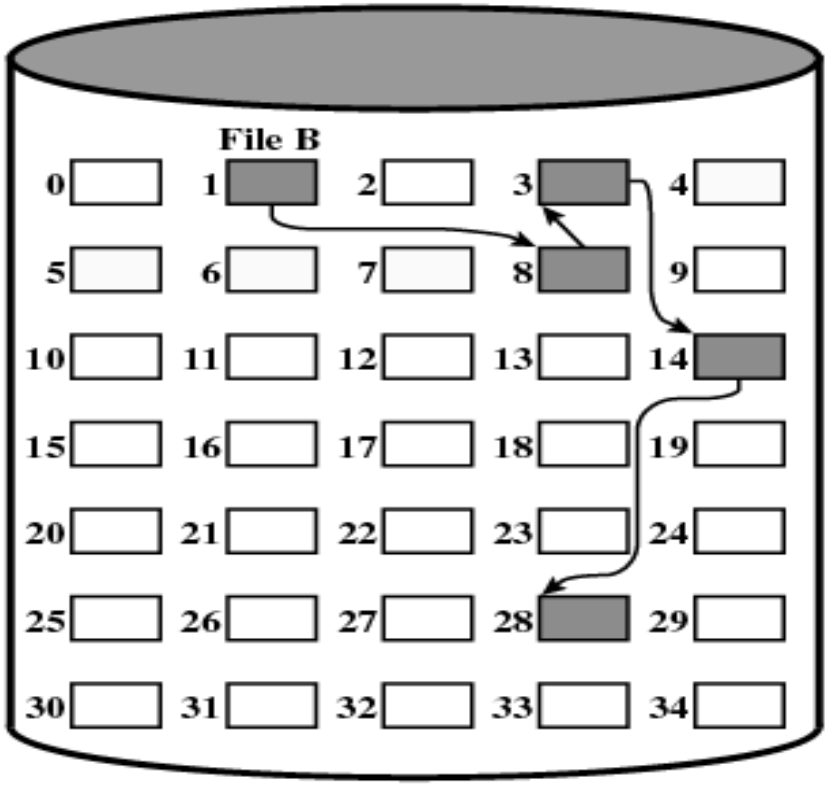
File Allocation Table

File Name	Start Block	Length
File A	0	3
File B	3	5
File C	8	8
File D	19	2
File E	16	3

Figure 12.8 Contiguous File Allocation (After Compaction)

Methods of File Allocation

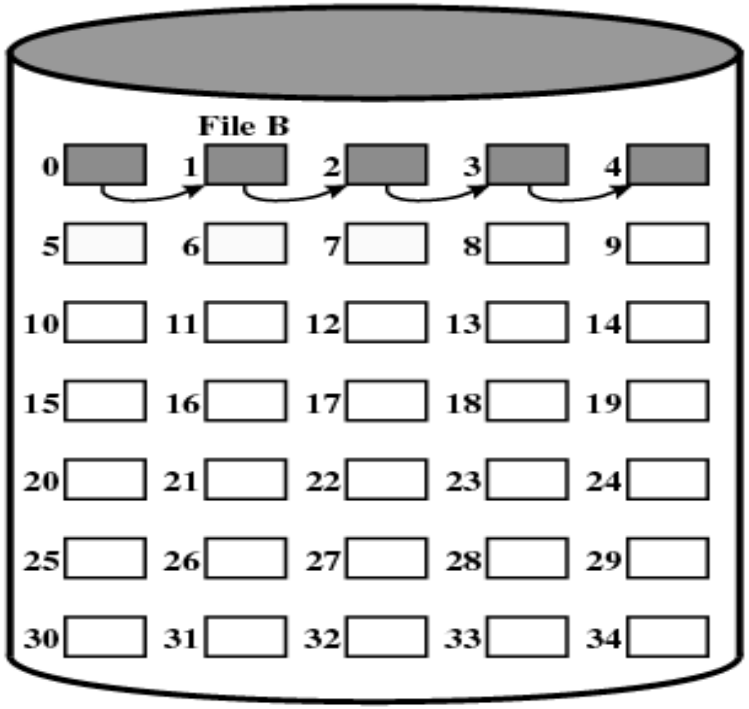
- Chained allocation
  - Allocation on basis of individual block
  - Each block contains a pointer to the next block in the chain
  - Only single entry in the file allocation table
    - Starting block and length of file
- No external fragmentation
- Best for sequential files
- No accommodation of the principle of locality



File Allocation Table

File Name	Start Block	Length
...	...	...
File B	1	5
...	...	...

Figure 12.9 Chained Allocation



File Allocation Table

File Name	Start Block	Length
. . .	. . .	. . .
File B	0	5
. . .	. . .	. . .

Figure 12.10 Chained Allocation (After Consolidation)

Methods of File Allocation

- Indexed allocation
  - File allocation table contains a separate one-level index for each file
  - The index has one entry for each portion allocated to the file
  - The file allocation table contains the block number for the index

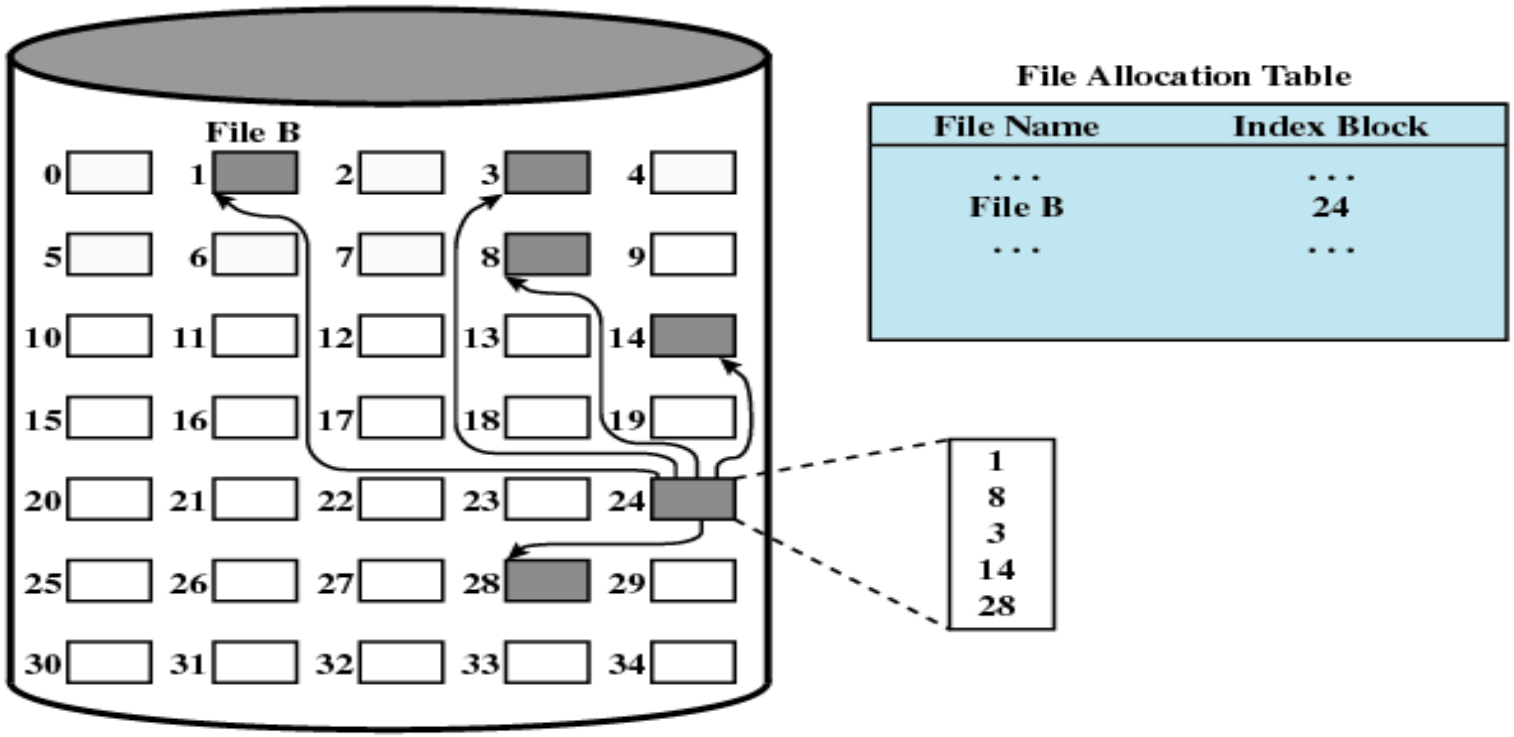


Figure 12.11 Indexed Allocation with Block Portions

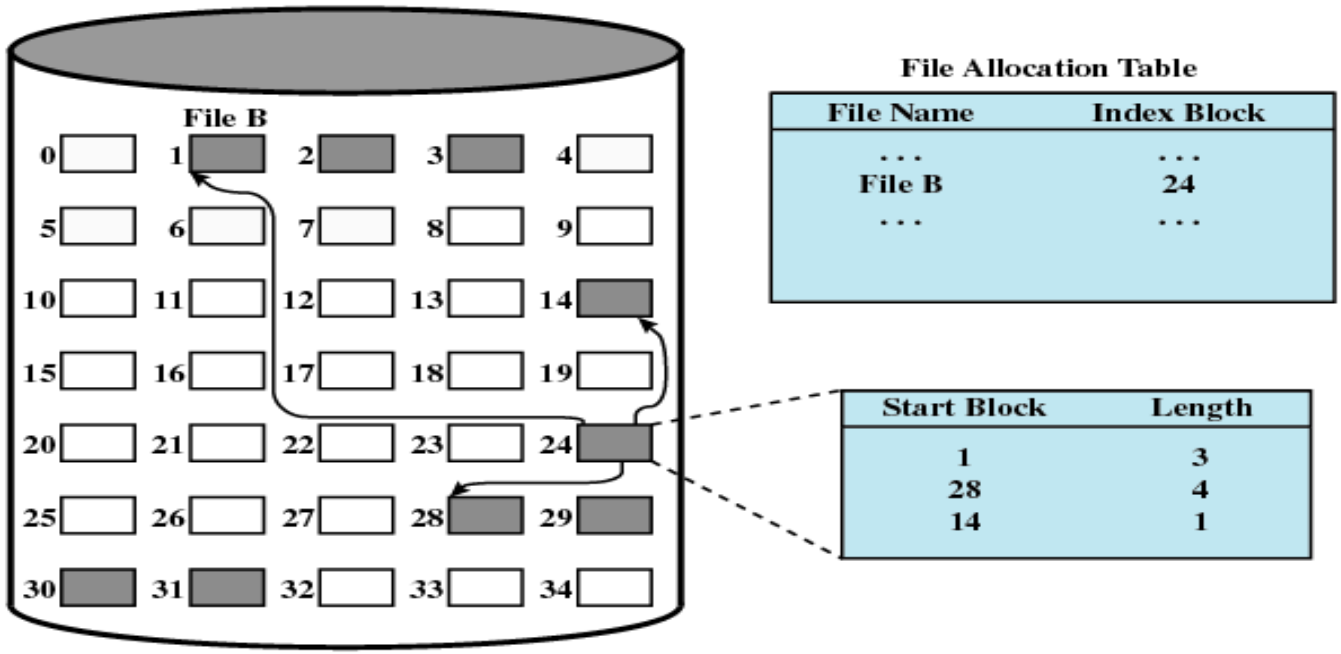


Figure 12.12 Indexed Allocation with Variable-Length Portions