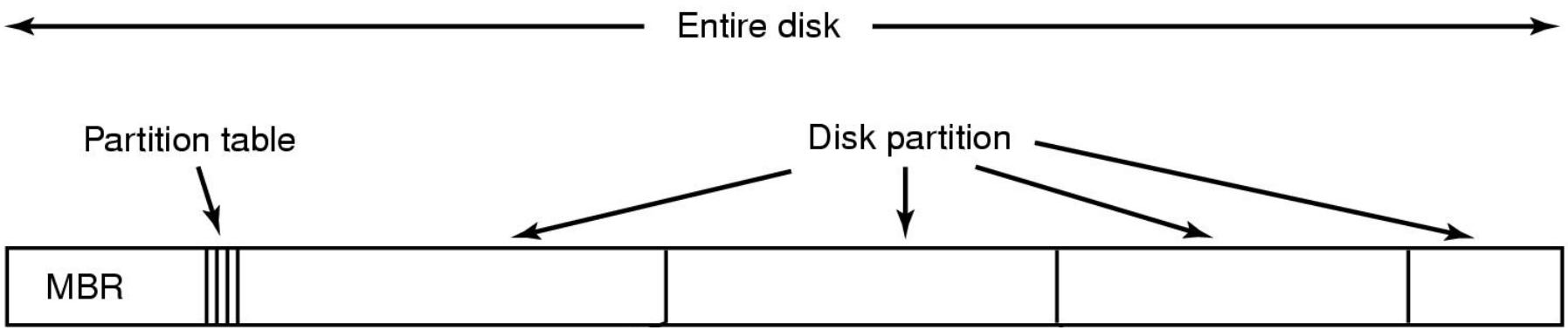
Operating Systems

Jinghui Zhong （钟竞辉）
Office： B3-515
Email： jinghuizhong@scut.edu.cn

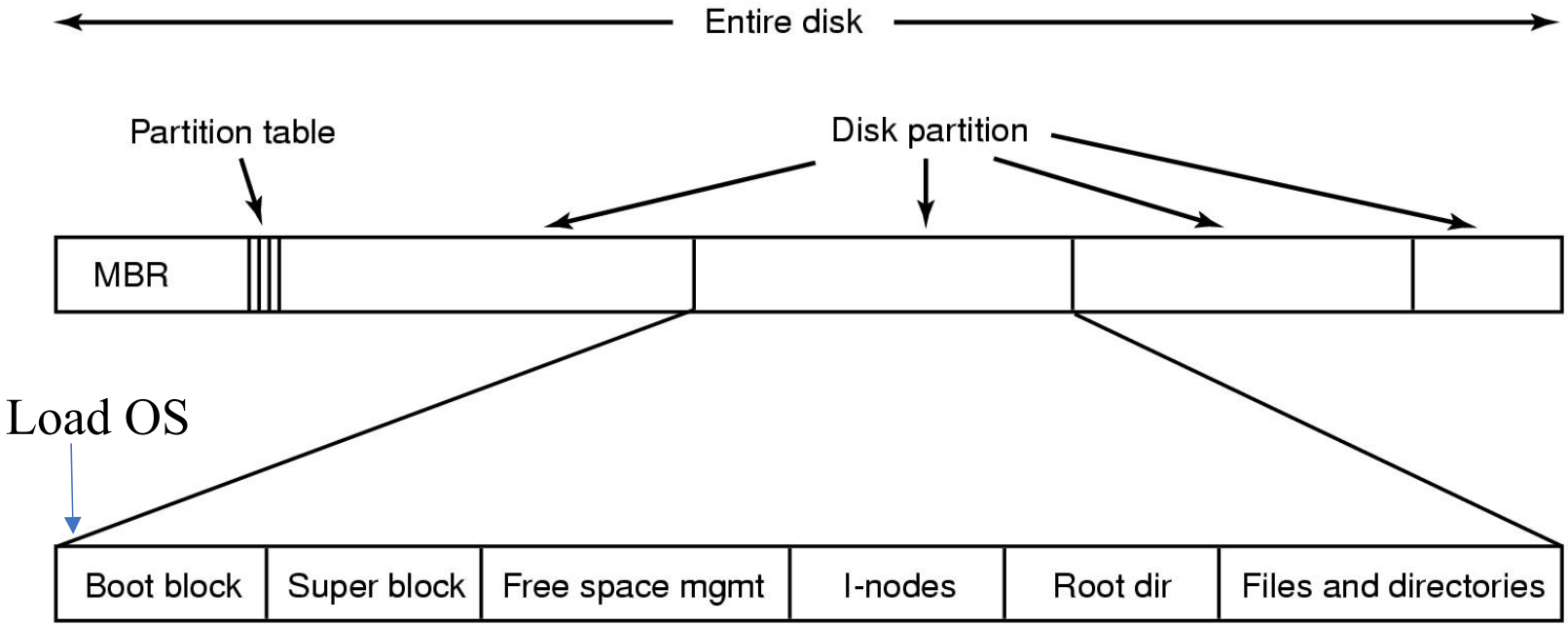
File System Layout

- **File system layout:**
  - MBR (Master Boot Record) is used to boot the computer.
  - The partition table gives the starting and ending addresses of each partition.



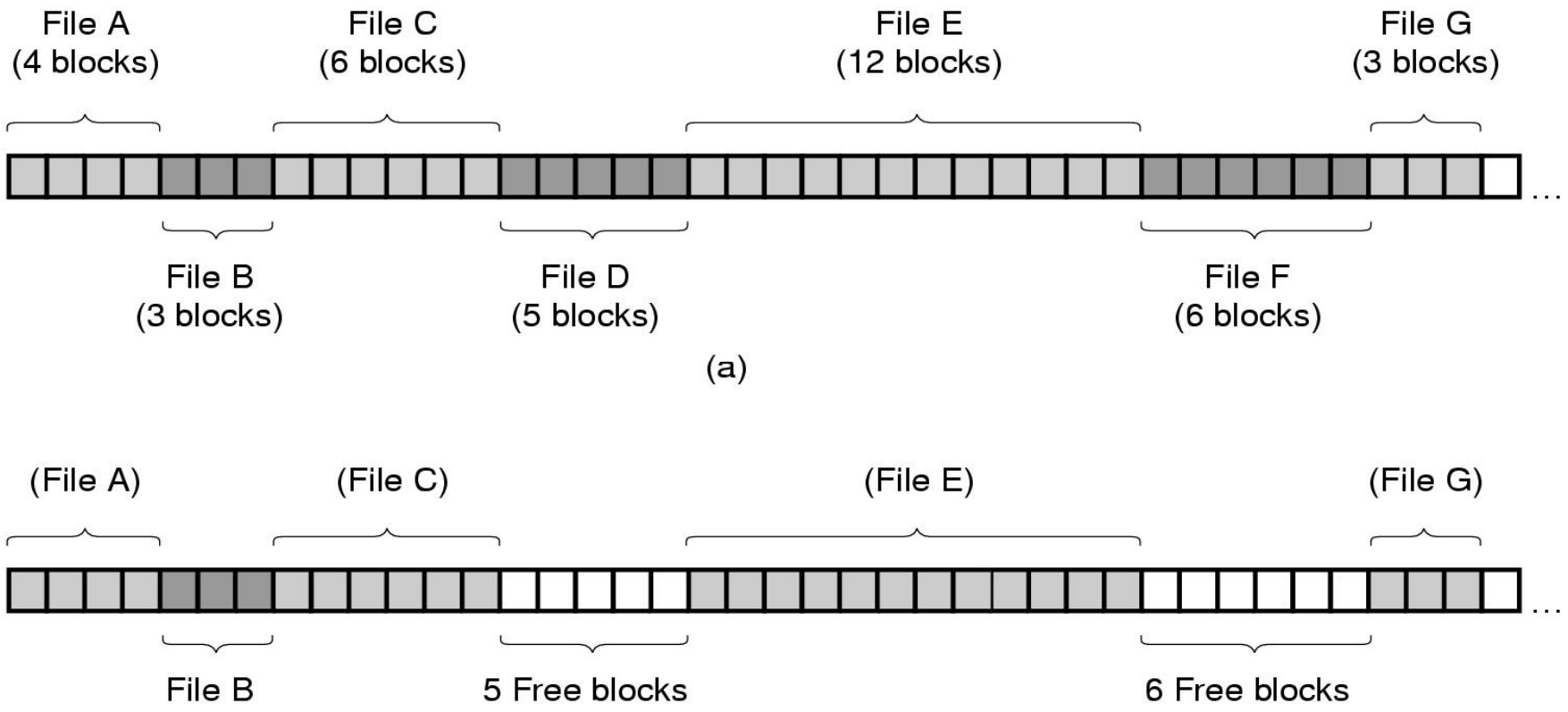
File System Layout

① **Boot block:** read in by the MBR program when the system is booted.
② **Superblock:** contains the key parameters about the file system.
③ Free blocks information
④ I-nodes tells all about the file.
⑤ Root directory
⑥ Directories and files



File Allocation

- **Contiguous Allocation:**
  store each file as contiguous block of data.



File Allocation

- **Contiguous Allocation**
  - **Advantages:**

    Simple to implement;

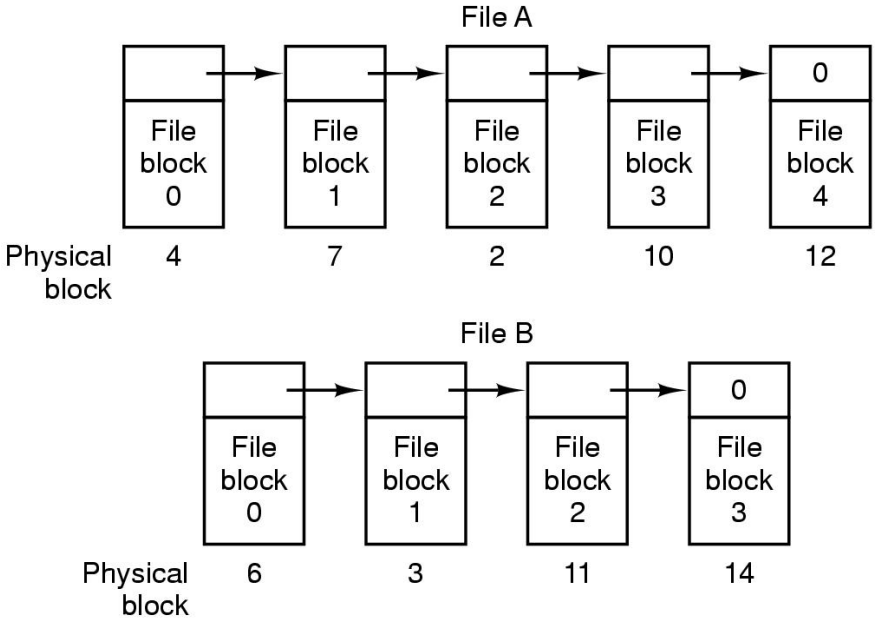
    Read performance is excellent.

  - **Disadvantages:**
    - ✓Disk fragmentation
    - ✓The maximum file size must be known when file is created.
  - **Example:** CD-ROMs, DVDs

File Allocation

- **Linked List Allocation:**

  keep linked list of disk blocks



- **Disadvantages ?**

  ① Slow random access speed

  ② The amount of data in a block is not a power of 2

Linked List Allocation using an index

- Take table pointer word from each block and put them in an index table, **FAT (File Allocation Table)** in memory.

  - **Advantages?**

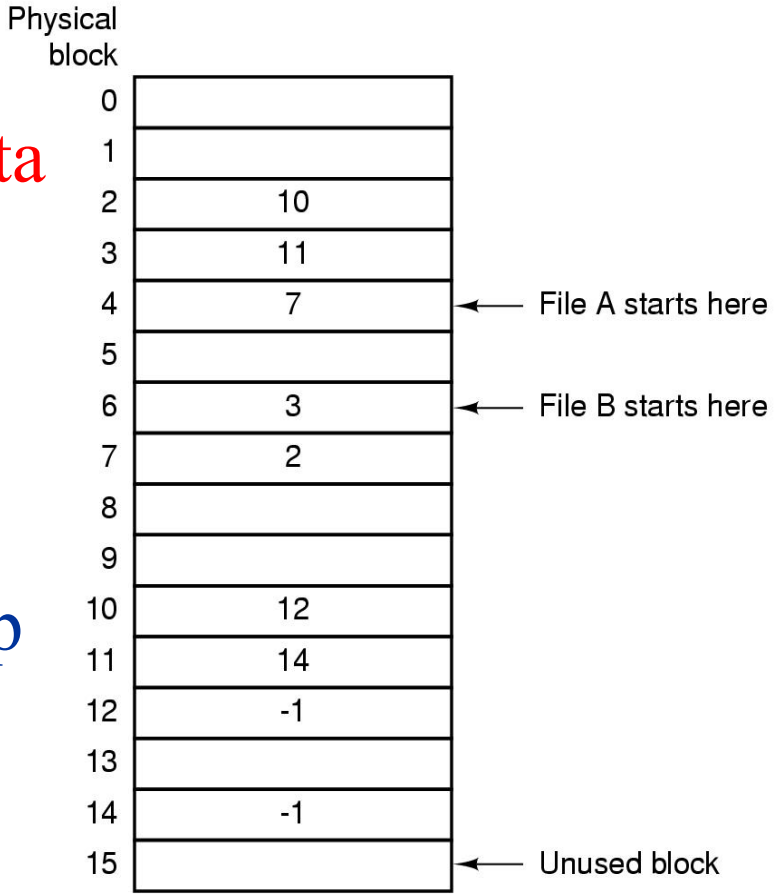
  ① The entire block is available for data

  ② Stored in memory, fast

  - **Drawbacks?**

  Occupies a large amount of memory.

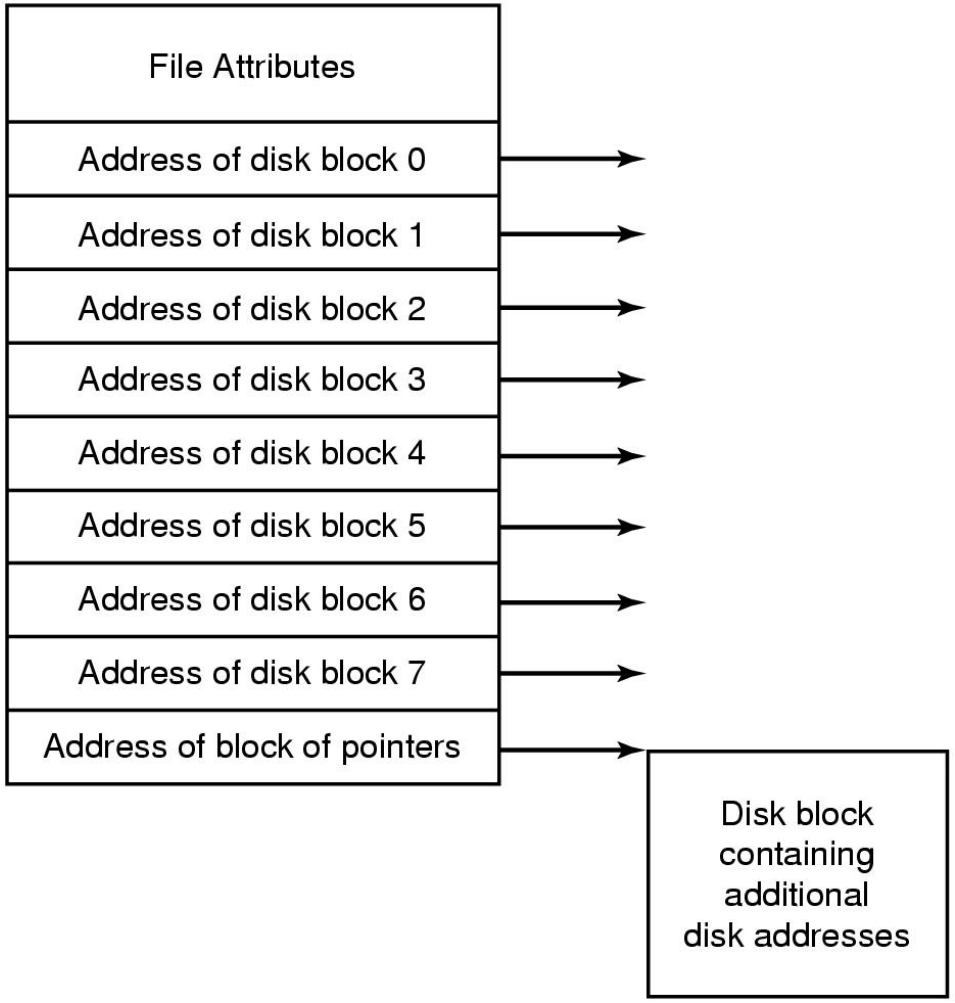
  For 200-GB disk, the table will take up 600M or 800 M memory.



File System Implementation

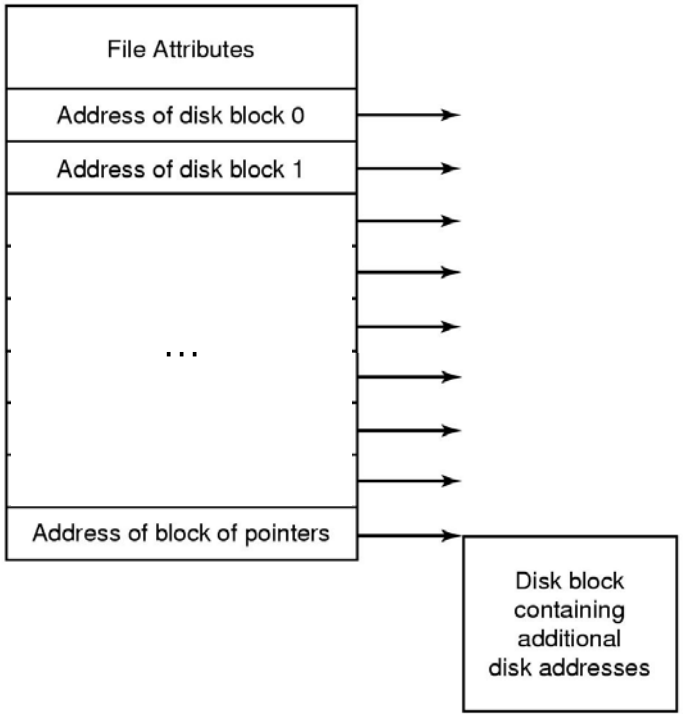
- I-node (index-node):

  lists the attributes and disk addresses of the file's blocks.



Problem

Consider an i-node, which contains 10 direct addresses and these were 8 bytes each and all disk blocks were 1024KB, what would the largest possible file be?

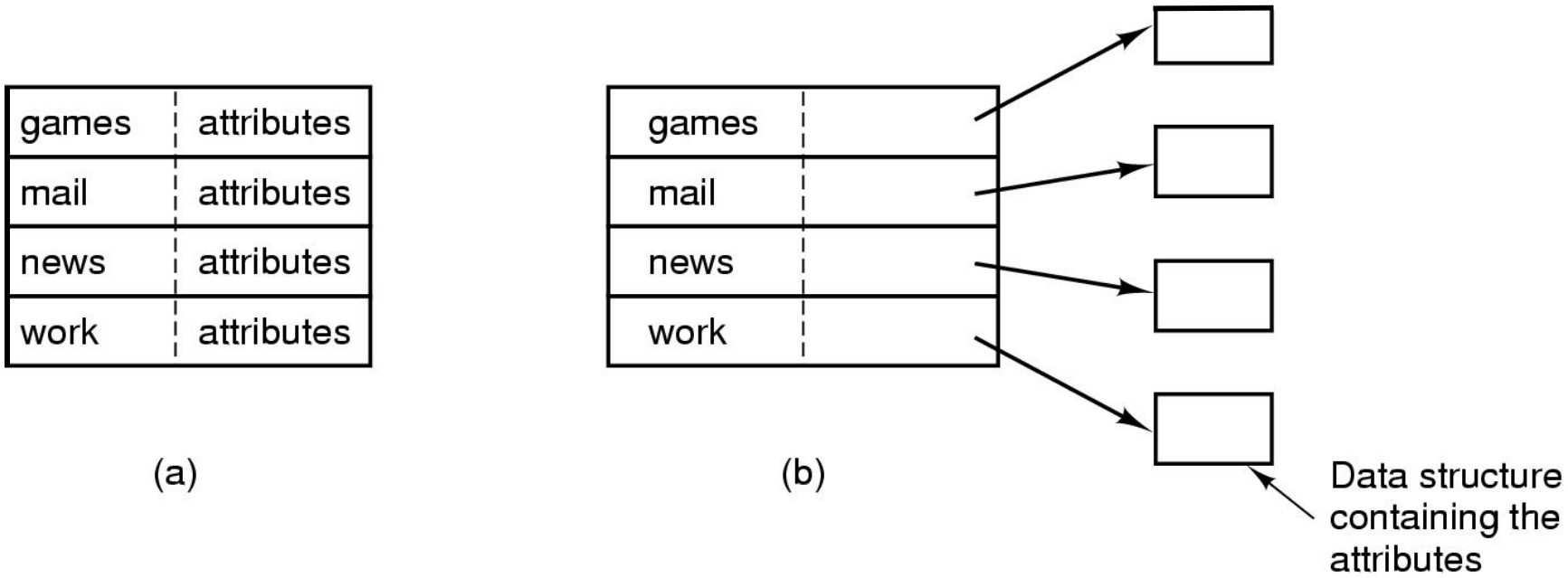


Directories

- When a file is opened, the file system uses the path name to locate the **directory entry**.
- **Directory:** provides information needed to find the disk blocks.
  - ① disk address of the entire file (contiguous blocks)
  - ② the number of first block (linked list)
  - ③ the number of i-node (i-node)
- Where to store attributes? In directory or i-node?

Directories

- **Two methods to store attributes**

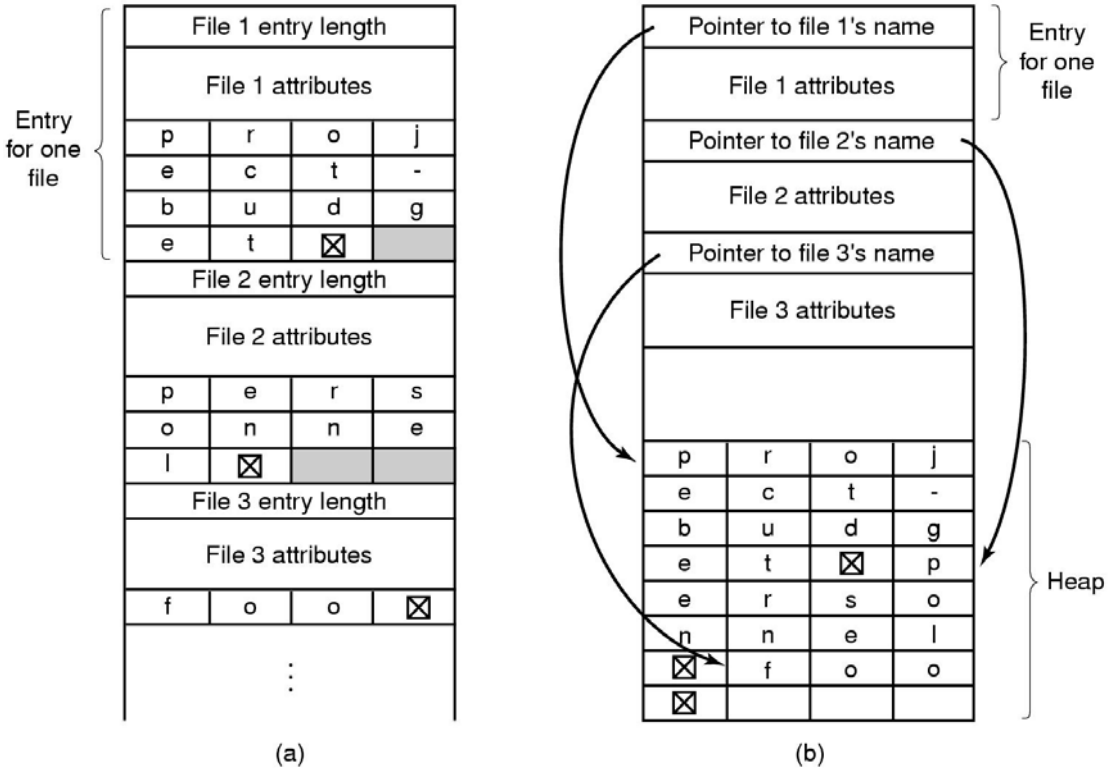


① Fixed size entries, disk addresses and attributes in directory entry (MS-DOS/Windows)

② Directory in which each entry just refers to an i-node (UNIX)

Implementation directories

- **Handling long file names in a directory:**
  - ① Fixed-length names (Waste space)
  - ② In-line (When a file is removed, a variable-sized gap is introduced., see(a))
  - ③ Heap (The heap management needs extra effort, see (b))



Implementation directories

- **How to search files in each directory?**

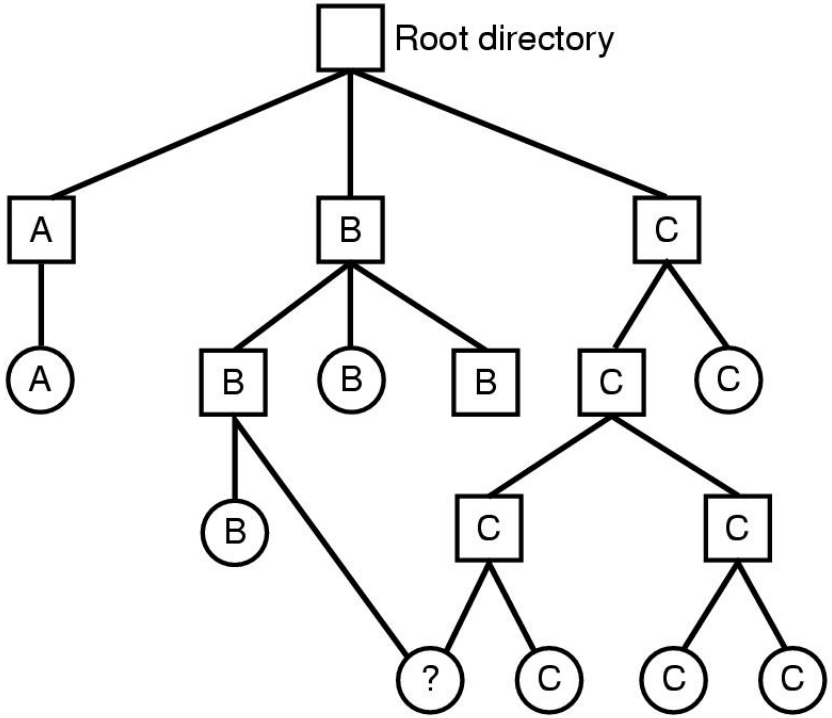
  ① Linearly (slow)

  ② Hash table (faster lookup but more complex administration)

  ③ Cache the results of searches

Shared files

- A shared file is used to allow a file to appear in several directories.
- The connection between a directory and the shared file is called a **link**. The file system is a **Directed Acyclic Graph (DAG)**.



Shared file

Shared files

- **Problem:**

If directories contain disk address, a copy of the disk address will have to be made in directory B. What if A or B append the file, the new blocks will only appear in one directory.

- **Solution:**

① Do not list disk block addresses in directories but in a little data structure. e.g., i-nodes

② Create a new file of type link which contains the path name of the file to which it is linked ➔ **symbolic linking**

Virtual File Systems

- **Definition:** the Virtual File System (or the **Virtual Filesystem Switch**) is the software layer in the kernel that provides the filesystem interface to user space programs.
  - From "the" filesystem to many filesystem types
  - Examples – Ext2, UFS(Solaris), NFS, Ext3, Veritas, ReiserFS, XFS, ISO9660 (CD), UDF (DVD) etc.

Virtual File Systems

- **Same API for different types of file systems**

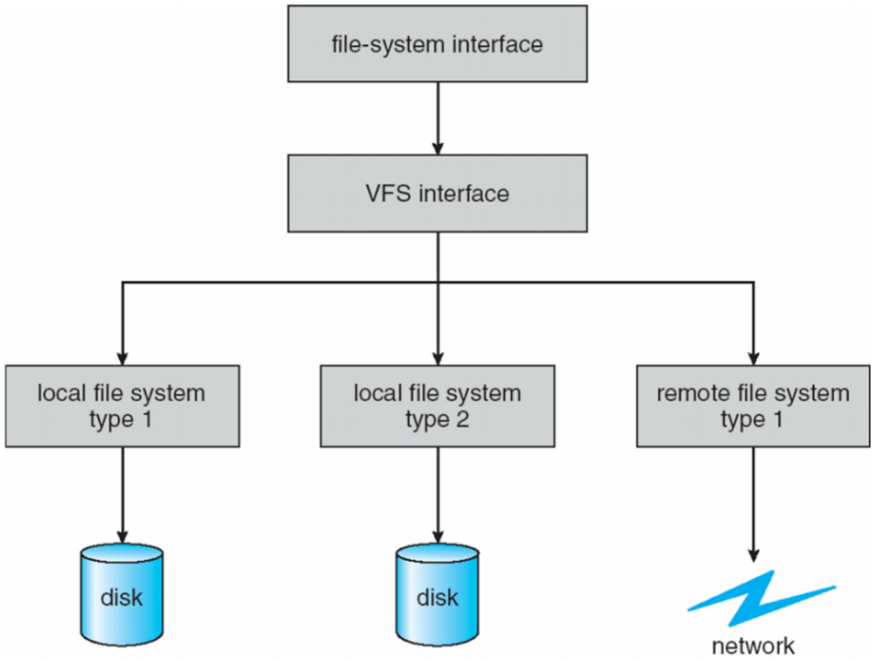
① Separates file-system generic operations from implementation details

② Syscalls program to VFS API rather than specific FS interface

- **Very flexible use cases:**

① User files remote and system files local?

② Boot from USB? Network?



Disk space management

- **Strategies for storing an *n* byte file:**

① Allocate n consecutive bytes of disk space

If the file grows it will have to be moved on the disk, it is an expensive operation and causes external fragmentation.

② Allocate a number [n/k] blocks of size k bytes each

Blocks do not need to be adjacent.

How to determine block size?

- **When block size increase, disk space utilization decrease**

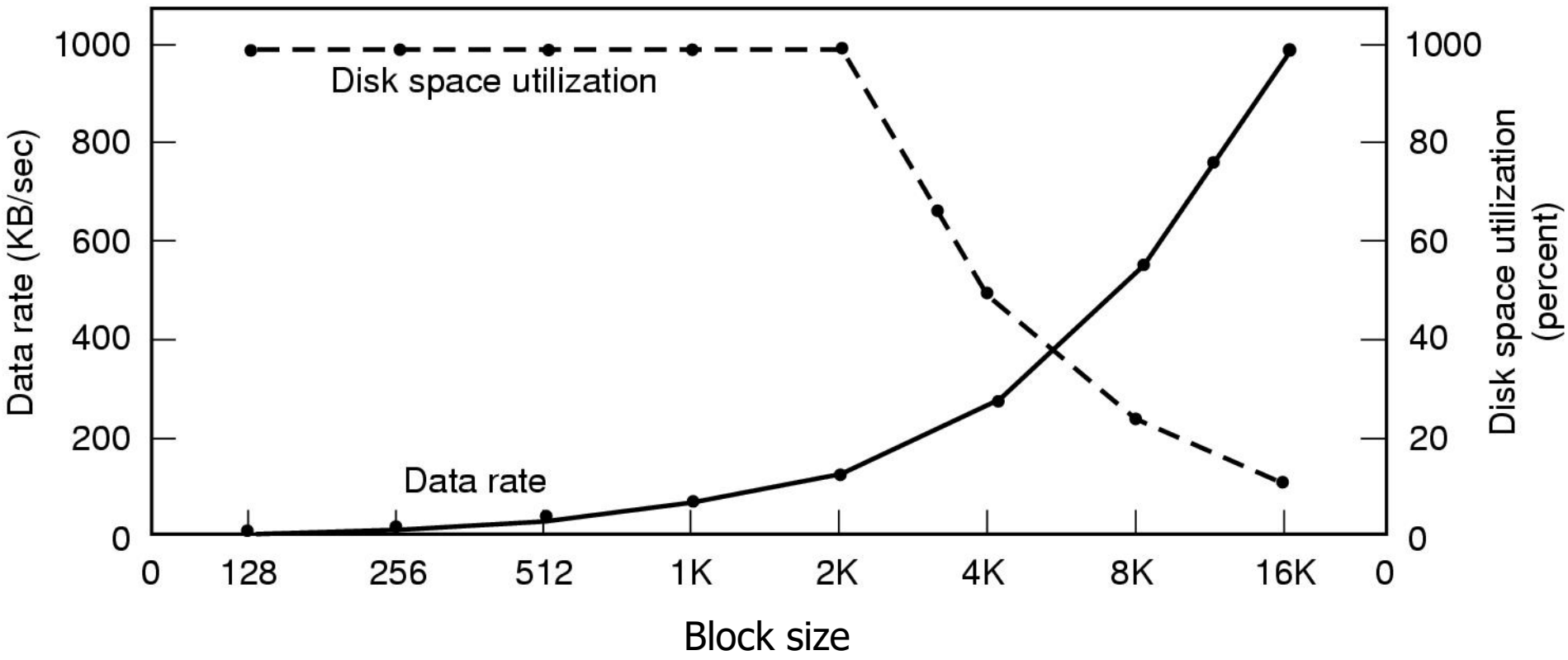
  Internal fragmentation, space efficiency decrease

- **When block size decrease, data transfer rate decrease**

  Time efficiency decrease

  usual size k = 512bytes, 1k (UNIX), or 2k

Disk Space Management



Dark line gives data rate of a disk

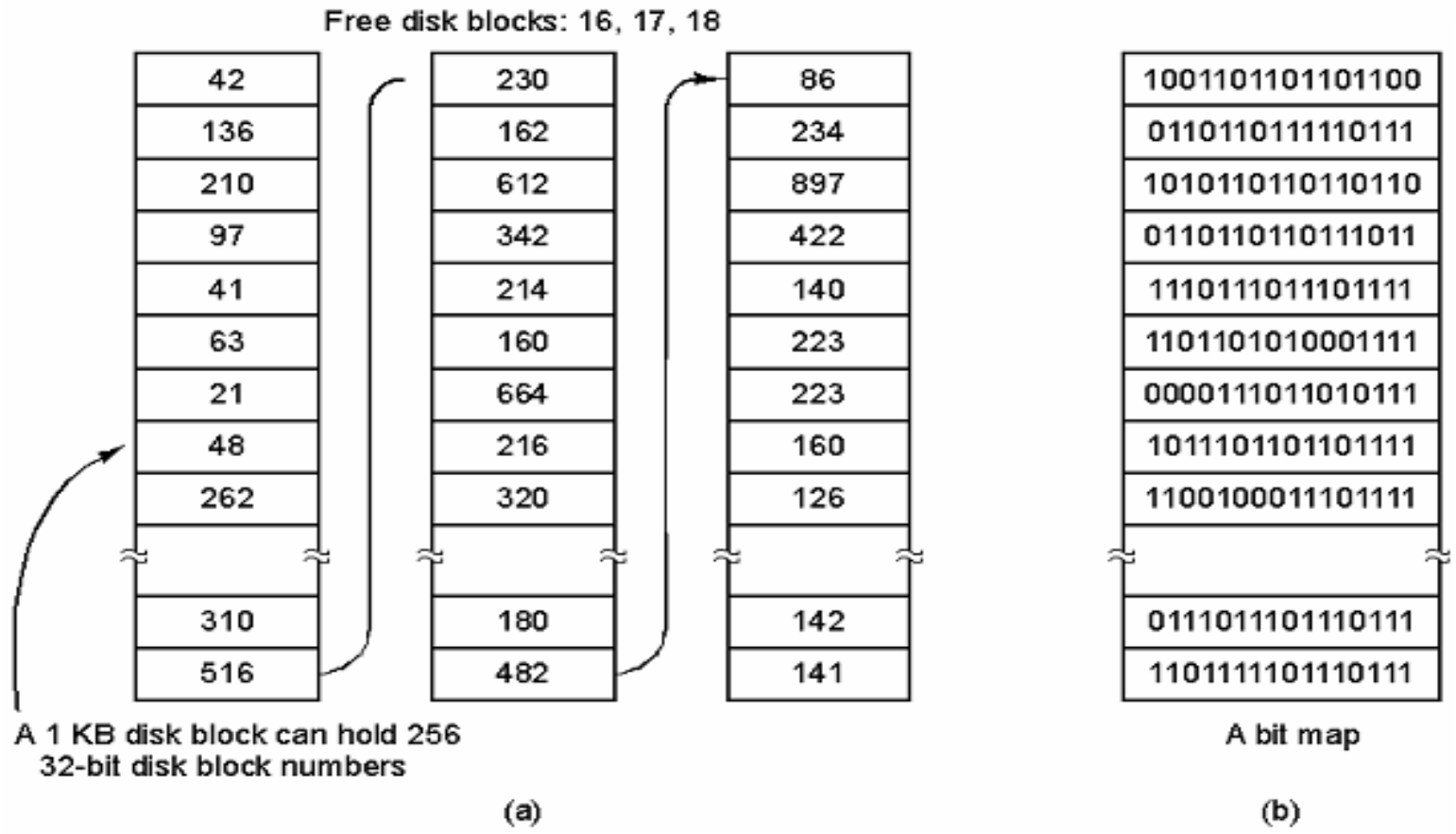
Dotted line gives disk space efficiency

Keeping Track of Free Blocks

- **Use linked list of disk blocks:**

With 1 KB block and 32-bit disk block number.

- **Use bit-map:** Free blocks ->1, Allocated blocks -> 0



Keeping Track of Free Blocks

- **Use linked list of disk blocks**: Each block holds as many free disk block numbers as will fit.

  With 1 KB block and 32-bit disk block number ➔ 1024 * 8/32 = 256 disk block numbers ➔ 255 free blocks (and) 1 next block pointer.

- **Use bit map**: A disk with (n) blocks requires a bitmap with (n) bits

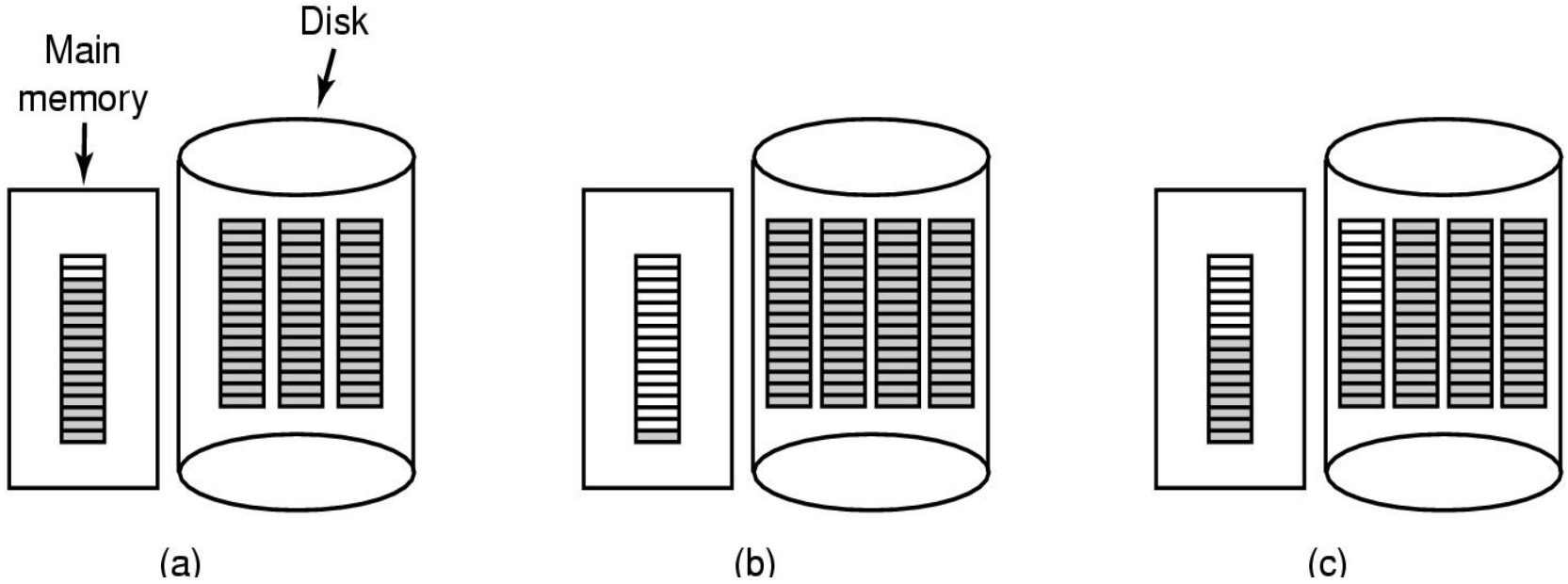
  ✓Free blocks are represented by 1's

  ✓Allocated blocks represented by 0's

  ✓16GB disk has 2^{24} 1-KB and requires 2^{24} bits ➔ 2048 blocks

  ✓Using a linked list = $2^{24}/255$ = 65793 blocks.

Disk Space Management



(a) Almost-full block of pointers to free disk blocks in RAM
- three blocks of pointers on disk

(b) Result of freeing a 3-block file

(c) Alternative strategy for handling 3 free blocks
- Keep the one in memory about half full.

Check Points

① Please describe the advantages and disadvantages of contiguous allocation.

② Please describe the advantages and disadvantages of linked listed allocation.

③ Please describe the advantages and disadvantages of FAT.

④ Please describe the two methods for keeping track of free blocks

Presentation & Poster

① Evaluation:

Correctness, Clarity, time, Group work; Interestingness;

② Score: A, B, C; A, B, C