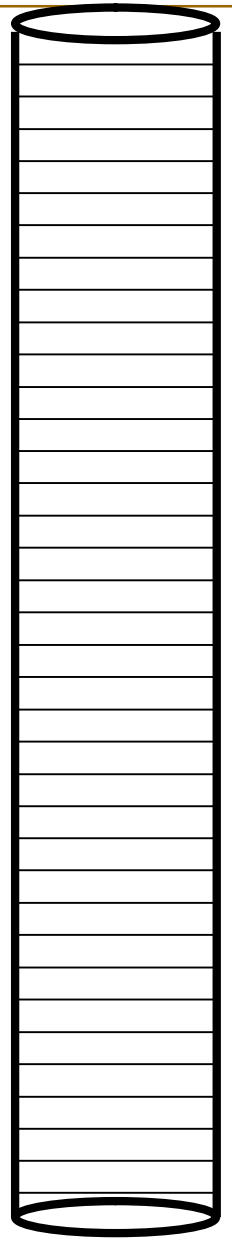
High Performance Computing
Lecture 35

Matthew Jacob

Indian Institute of Science

About Disks

- Our view of disk: linear address space of fixed size sectors/blocks numbered from 0 up



Operations on Files

- fd = open (name, operation)
- fd = creat (name, mode)
- status = close(fd)
- bytecount = read (fd, buffer, bytecount)
- bytecount = write (fd, buffer, bytecount)
- offset = lseek (fd, offset, whence)
- status = link (oldname, newname)
- status = unlink (name)
- status = stat (name, buffer)
- status = chown (name, owner, group)
- status = chmod (name, mode)

Common File Access Patterns

- Sequential access: bytes of file are read in order from start to finish
  - Reading a file from start to end
  - e.g., cat program.c
- Random access: bytes of file are read in some (random) order
  - e.g., Query to a database file

File System Design Issues

1. Disk management: efficient use of disk space
2. Name management: how users select files for use
3. Protection: of files from users

Disk Management

Issues

1. Disk Block Allocation: How are disk blocks associated with a file?
2. Arm scheduling: Which disk I/O request should be sent to disk next?

   FCFS, Shortest Seek Time First (SSTF)

Disk Block Allocation

- Question: How does the OS keep track of which disk blocks are associated with a given file?
- Consider a file that is 10 disk blocks in size
- File Descriptor: OS structure that describes which blocks on disk represent a file
- Issues:
  1. Take common file access patterns into account
  2. It must be possible for files to change in size

Disk Block Allocation: Contiguous

File is stored in contiguous blocks on disk

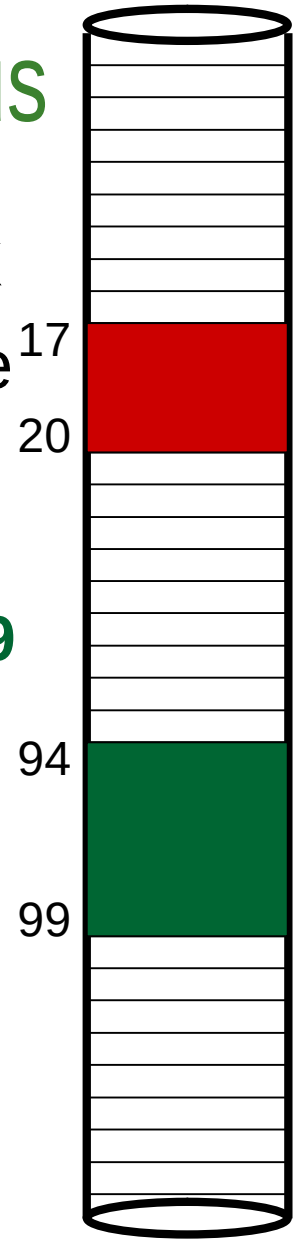
- File descriptor: first block address, file size

File 1: Size 4 blocks; Blocks 17, 18, 19, 20

File 2: Size 6 blocks; Blocks 94, 95, 96, 97, 98, 99

File 1: Start 17 Size 4

File 2: Start 94 Size 6



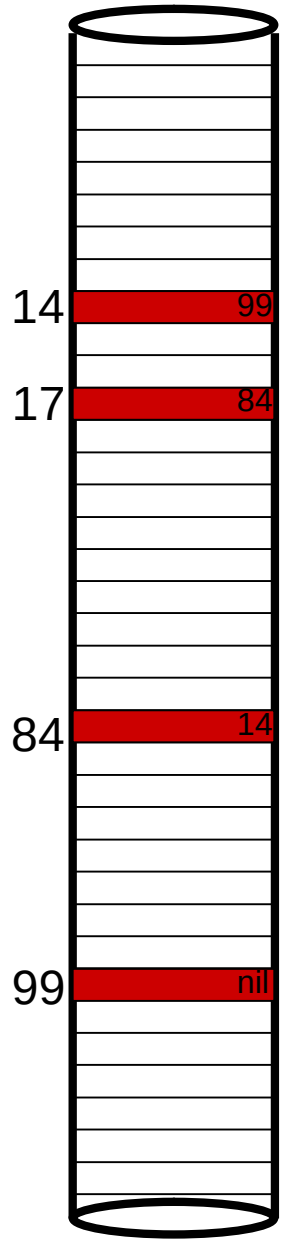
Disk Block Allocation: Linked

Each file block contains the disk address of the next file block

- File descriptor: first disk block address

File 1: Size 4 blocks; Blocks 17, 84, 14, 99

File 1: Start 17



Disk Block Allocation: Indexed

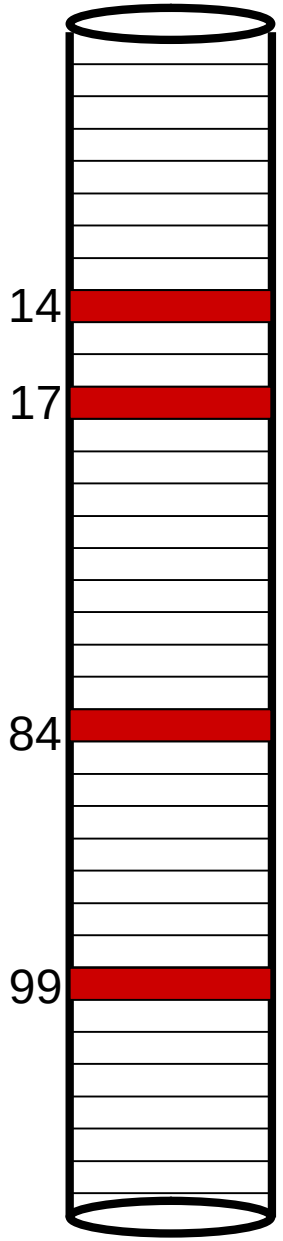
File Index is an array containing addresses of 1st, 2nd, etc block of file

- File descriptor: index

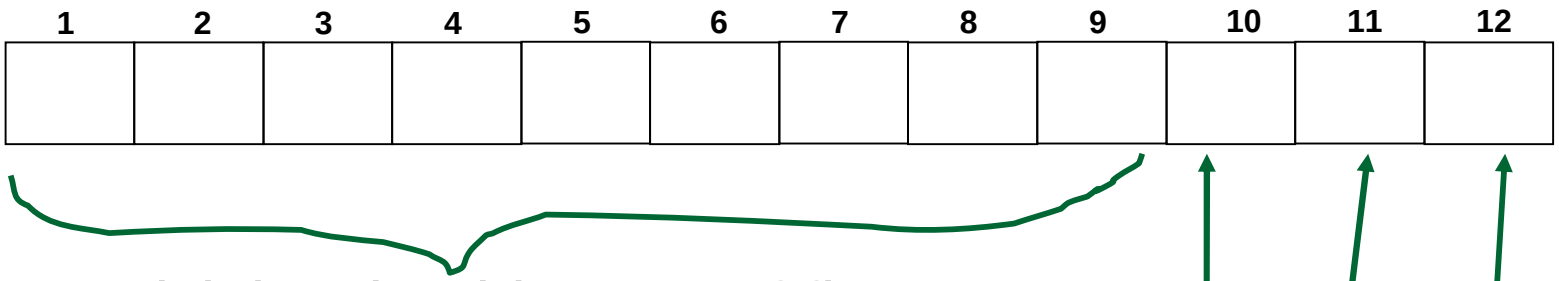
File 1: Size 4 blocks; Blocks 17, 84, 14, 99

	1	2	3	4
INDEX	17	84	14	99

Problem: size of the index?



UNIX Version of Indexed Allocation



Assume disk block size: 1KB,

disk block address size: 4B

Indirect disk block address
- address of disk block containing more disk block addresses of the file

Maximum file size:
9KB + 256KB
+ 256*256 KB
+256*256*256 KB

Doubly indirect disk block address

Triply indirect disk block address