File System

File Attributes:

- Name – only information kept in human-readable form
- Identifier – unique tag (number) identifies file within file system
- Type – needed for systems that support different types
- Location – pointer to file location on device
- Size – current file size
- Protection – controls who can do reading, writing, executing
- Time, date, and user identification – data for protection, security,

and usage monitoring

File Operations

- File is an abstract data type
- Create
- Write – at write pointer location
- Read – at read pointer location
- Reposition within file - seek
- Delete
- Truncate
- Open(Fi) – search the directory structure on disk for entry Fi , and move the content of entry to memory
- Close (Fi) – move the content of entry Fi in memory to directory structure on disk
- Open File Locking
- Shared lock similar to reader lock – several processes can acquire

concurrently

- Exclusive lock similar to writer lock

Access Methods

- Sequential Access

read next

write next

reset

no read after last write

(rewrite)

- Direct Access – file is fixed length logical records

read n

write n

position to n

    read next

    write next

rewrite n

Operations Performed on Directory

- Search for a file
- Create a file
- Delete a file
- List a directory
- Rename a file
- Traverse the file system

Single-Level Directory

- All files are contained in the same directory, which is easy to support and understand
- Naming problem
- Grouping problem

Two-Level Directory

- Separate directory for each user
- Path name
- Can have the same file name for different user
- Efficient searching

- Three classes of users on Unix / Linux

RWX

a) owner access 7 1 1 1

RWX

b) group access 6 1 1 0

RWX

c) public access 1 0 0 1

- No grouping capability

File Allocation Methods

- Three major methods of allocating disk space are
- Contiguous
- Linked
- Indexed

Contiguous Allocation

- Each file occupy a set of contiguous blocks on the disk
- Disk addresses define a linear ordering on the disk

Linked Allocation

- Each file is a linked list of disk blocks
- The disk blocks may be scattered anywhere on the disk

- The directory contains a pointer to the first and last blocks of the file

Magnetic disks

- Magnetic disks provide bulk of secondary storage of modern

Computers

- Transfer rate is rate at which data flow between drive and computer
- Positioning time (random-access time) is time to move disk arm to desired cylinder (seek time) and time for desired sector to rotate under the disk head (rotational latency)
- Head crash results from disk head making contact with the disk surface --

Magnetic Tape

- Relatively permanent and holds large quantities of data
- Access time slow
- Random access ~1000 times slower than disk

Disk Scheduling

FCFS

SSTF

- Shortest Seek Time First selects the request with the minimum seek time from the current head position
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some

Requests

SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues

C-SCAN

- Circular SCAN
- When the head reaches the other end, however, it immediately returns to the beginning of the disk without servicing any requests on the return trip

LOOK

- LOOK a version of SCAN, C-LOOK a version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk

I/O Hardware

- Port – connection point for device
- Bus - daisy chain or shared direct access

□PCI bus common in PCs and servers, PCI Express (PCIe)

□expansion bus connects relatively slow devices

- Controller (host adapter) – electronics that operate port, bus, device
- Interrupt handler receives interrupts
- Interrupt vector to dispatch interrupt to correct handler

Direct Memory Access

- Used to avoid programmed I/O (one byte at a time) for large data movement
- Requires DMA controller
- Unix ioctl() call to send arbitrary bits to a device control register and data to device data register
- Character devices include keyboards, mice, serial ports

- Commands include get(), put()
- Libraries layered on top allow line editing
- ioctl() (on UNIX) covers odd aspects of I/O such as clocks and timers

Nonblocking and Asynchronous I/O

- Blocking - process suspended until I/O completed
- Easy to use and understand
- Insufficient for some needs
- Nonblocking - I/O call returns as much as available
- User interface, data copy (buffered I/O)
- Implemented via multi-threading
- Returns quickly with count of bytes read or written
- select() to find if data ready then read() or write() to transfer
- Asynchronous - process runs while I/O executes
- Difficult to use
- I/O subsystem signals process when I/O completed

- Vectored I/O allows one system call to perform multiple I/O operations
- For example, Unix readve() accepts a vector of multiple buffers to read into or write from
- Buffering - store data in memory while transferring between devices
- To cope with device speed mismatch
- To cope with device transfer size mismatch
- Double buffering – two copies of the data

□Kernel and user

□Varying sizes

- Caching - faster device holding copy of data
- Always just a copy
- Key to performance
- Sometimes combined with buffering
- Spooling - hold output for a device
- If device can serve only one request at a time
- i.e., Printing
- Device reservation - provides exclusive access to a device
- System calls for allocation and de-allocation
- Watch out for deadlock