Lecture 9 – Random Access Devices

Karl R. Wilcox
Karl@cs.rhul.ac.uk

Royal Holloway University of London

Objectives

- In this class we will discuss:
  - What are random access devices
  - Their characteristics and performance
  - How to manage such devices

Random Access Devices

- **Hard Disks**
- **Floppy Disks**
- **CD-Drives**
- **DVDs**
- **RAM**
- **Optical Disks**

These are all the same from the point of view of a user of the Operating System.
They have different properties from the point of view of the designer.

Royal Holloway University of London

Properties of "Disks"

- **Read-only**
- **Removable**
- **Write once**
- **Fast**
- **Slow**
- **Volatile**

Royal Holloway University of London

Disk Performance Parameters

- **To read or write, the disk head must be positioned at the desired track and at the beginning of the desired sector**
- **Seek time**
  - **time it takes to position the head at the desired track**
- **Rotational delay or rotational latency**
  - **time it takes for the beginning of the sector to reach the head**

Timing of a Disk I/O Transfer

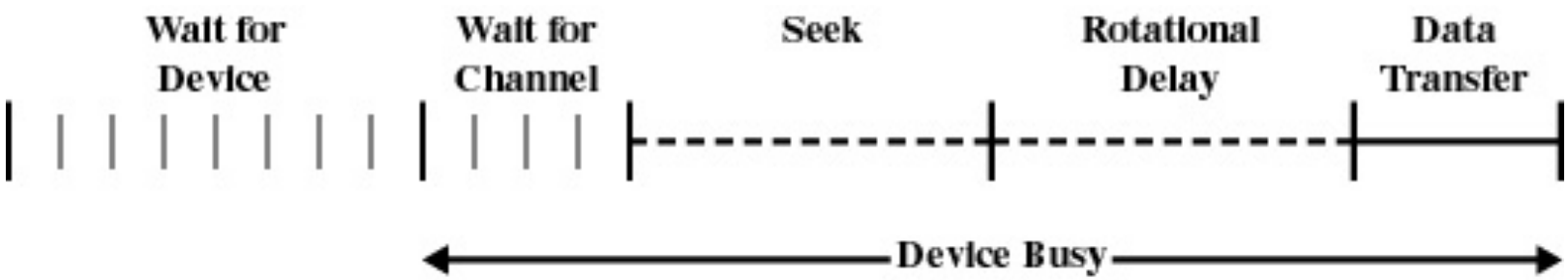

Figure 11.7 Timing of a Disk I/O Transfer

Disk Performance Parameters

- **Access time**
  - **Sum of seek time and rotational delay**
  - **The time it takes to get in position to read or write**
- **Data transfer occurs as the sector moves under the head**

Royal Holloway University of London

Disk Scheduling Policies

- **Seek time is the reason for differences in performance**
- **For a single disk there will be a number of I/O requests**
- **If requests are selected randomly, we will get the worst possible performance**

Disk Scheduling Policies

- **First-in, first-out (FIFO)**
  - **Process request sequentially**
  - **Fair to all processes**
  - **Approaches random scheduling in performance if there are many processes**

Royal Holloway University of London

Disk Scheduling Policies

- **Priority**
  - **Goal is not to optimize disk use but to meet other objectives**
  - **Short batch jobs may have higher priority**
  - **Provide good interactive response time**

Disk Scheduling Policies

- **Last-in, first-out**
  - **Good for transaction processing systems**
    - **The device is given to the most recent user so there should be little arm movement**
  - **Possibility of starvation since a job may never regain the head of the line**

Royal Holloway University of London

Disk Scheduling Policies

- **Shortest Service Time First**
  - **Select the disk I/O request that requires the least movement of the disk arm from its current position**
  - **Always choose the minimum Seek time**

Disk Scheduling Policies

- **SCAN**
  - **Arm moves in one direction only, satisfying all outstanding requests until it reaches the last track in that direction**
  - **Direction is reversed**

Disk Scheduling Policies

- **C-SCAN**
  - **Restricts scanning to one direction only**
  - **When the last track has been visited in one direction, the arm is returned to the opposite end of the disk and the scan begins again**

Disk Scheduling Policies

- **N-step-SCAN**
  - **Segments the disk request queue into subqueues of length N**
  - **Subqueues are process one at a time, using SCAN**
  - **New requests added to other queue when queue is processed**
- **FSCAN**
  - **Two queues**
  - **One queue is empty for new request**

Royal Holloway University of London

Disk Scheduling Algorithms

Name	Description	Remarks
	Selection according to requestor	
RSS	Random scheduling	For analysis and simulation
FIFO	First in first out	Fairest of them all
PRI	Priority by process	Control outside of disk queue management
LIFO	Last in first out	Maximize locality and resource utilization
	Selection according to requested item:	
SSTF	Shortest service time first	High utilization, small queues
SCAN	Back and forth over disk	Better service distribution
C-SCAN	One way with fast return	Lower service variability
N-step-SCAN	SCAN of N records at a time	Service guarantee
FSCAN	N-step-SCAN with N = queue size at beginning of SCAN cycle	Load-sensitive

Royal Holloway University of London

Disk Cache

- **Buffer in main memory for disk sectors**
- **Contains a copy of some of the sectors on the disk**

Royal Holloway University of London

Least Recently Used

- The block that has been in the cache the longest with no reference to it is replaced
- The cache consists of a stack of blocks
- Most recently referenced block is on the top of the stack
- When a block is referenced or brought into the cache, it is placed on the top of the stack

Royal Holloway University of London

Least Recently Used

- The block on the bottom of the stack is removed when a new block is brought in
- Blocks don't actually move around in main memory
- A stack of pointers is used

Royal Holloway University of London

Least Frequently Used

- The block that has experienced the fewest references is replaced
- A counter is associated with each block
- Counter is incremented each time block accessed
- Block with smallest count is selected for replacement
- Some blocks may be referenced many times in a short period of time and then not needed any more

Royal Holloway University of London

Summary

- **We have covered**
  - **The characteristics of random access devices**
  - **Managing "disk" like devices**
  - **Scheduling algorithms**
  - **Caching policies**

Royal Holloway University of London

Next Lecture

- **We will begin discussing file systems**
- **Lecture Notes: http://www.cs.rhul.ac.uk/~karl**