

TCCS 422: OPERATING SYSTEMS

Hard Disk Drives

Wes J. Lloyd
Institute of Technology
University of Washington - Tacoma



OBJECTIVES

- Chapter 37
 - HDD Internals
 - Seek time
 - Rotational latency
 - Transfer speed
 - Capacity
 - Scheduling algorithms

June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.2

HARD DISK DRIVE (HDD)

- Primary means of data storage (persistence) for decades
- Consists of a large number of data **sectors**
- Sector size is 512-bytes
- An n sector HDD can be addressed as an array of $0..n-1$ sectors

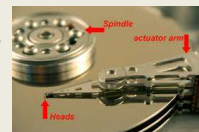
June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.3

HDD INTERFACE

- Writing disk sectors is atomic (512 bytes)
- Sector writes are completely successful, or fail
- Many file systems will read/write 4KB at a time
 - Linux ext3/4 default filesystem blocksize - 4096
- Same as typical memory page size



June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.4

BLOCK SIZE IN LINUX EXT4

- `mkefs.ext4 -i bytes-per-inode`

Specify the bytes/inode ratio. `mke2fs` creates an inode for every bytes-per-inode bytes of space on the disk. The larger the bytes-per-inode ratio, the fewer inodes will be created. This value generally shouldn't be smaller than the blocksize of the filesystem, since in that case more inodes would be made than can ever be used. Be warned that it is not possible to expand the number of inodes on a filesystem after it is created, so be careful deciding the correct value for this parameter.

June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.5

EXAMPLE: USDA SOIL EROSION MODEL WEB SERVICE (RUSLE2)

- Host ~2,000,000 files totaling 9.5 GB on a ~20GB filesystem on a cloud-based Virtual Machine
- With default inode ratio (4096 block size), only ~488,000 files will fit
- Drive less than half full, but files will not fit !
- HDDs support a minimum block size of 512 bytes
- OS filesystems such as ext3/ext4 can support "finer grained" management at the expense of a larger catalog size

June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.6

EXAMPLE: USDA SOIL EROSION MODEL
WEB SERVICE (RUSLE2) - 2

Free space in bytes (df)

Device	total size	bytes-used	bytes-free	usage
/dev/vda2	13315844	9556412	3049188	76% /mnt

Free inodes (df -i) @ 512 bytes / node

Device	total inodes	used	free	usage
/dev/vda2	3552528	1999823	1552705	57% /mnt

June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.7

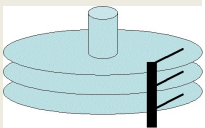
HDD INTERFACE - 2

Torn write

- When OS uses larger block size than HDD
- Block writes not **atomic** - they SPAN multiple HDD sectors
- Upon power failure only a portion of the OS block is written

HDD access

- Sequential reads of sectors is fastest
- Random sector reads are slow
- Disk head continuously must jump to different tracks



June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

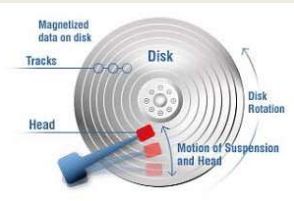
L18.8

HDD PLATTER

Made from aluminum coated with thin magnetic layer

HDD records on both sides of each platter

Data is stored by inducing magnetic changes



June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.9

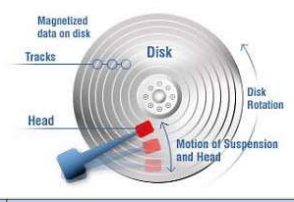
HDD SPINDLE

Connected to motor which spins the disk

Speed measures in RPM (rotations per minute)

Typical: 7200-15000 rpm

10000 rpm - 1 rotation in 6ms; 15k rpm 1 rotation in 4ms



June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.10

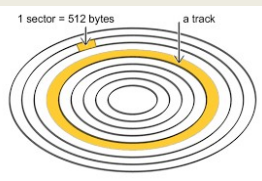
HDD TRACK

Concentric circle of sectors

Single side of platter contains 290 K tracks (2008)

Zones: groups of tracks with same # of sectors

Outer tracks have More sectors



June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

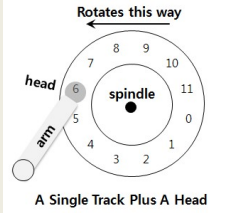
L18.11

EXAMPLE: SIMPLE DISK DRIVE

Single track disk

Head: one per surface of drive

Arm: moves heads across surface of platters

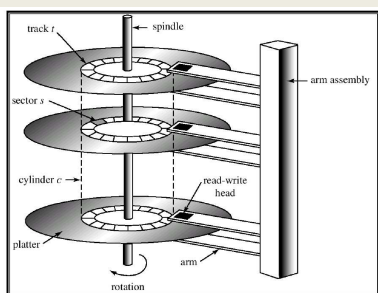


June 1, 2017

TCCS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.12

HARD DISK STRUCTURE



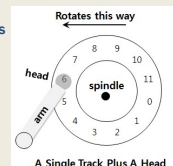
June 1, 2017

TCS5422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.13

SINGLE-TRACK LATENCY: THE ROTATIONAL DELAY

- Rotational latency (T_{rotation}): time to rotate to desired sector
- Average T_{rotation} is ~ half the time of a full rotation
- Calculate time for 1 rotation based on rpm
- 7200rpm = 8.33ms per rotation = ~4.166ms
- 10000rpm = 6ms per rotation = ~3ms
- 15000rpm = 4ms per rotation = ~2ms

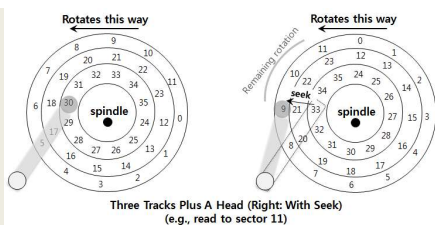


June 1, 2017

TCS5422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.14

SEEK TIME



- Seek time (T_{seek}): time to move disk arm to proper track
- Most time consuming HDD operation

June 1, 2017

TCS5422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.15

FOUR PHASES OF SEEK

- Acceleration → coasting → deceleration → settling
- Acceleration:** the arm gets moving
- Coasting:** arm moving at full speed
- Deceleration:** arm slow down
- Settling:** Head is carefully positioned over track
 - Settling time is often high, from .5 to 2ms

June 1, 2017

TCS5422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.16

HDD I/O

- Data transfer
 - Final phase of I/O: time to read or write to disk surface
- Complete I/O cycle:
 1. Seek (accelerate, coast, decelerate, settle)
 2. Wait on rotational latency
 3. Data transfer

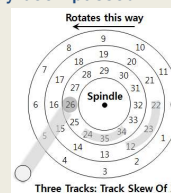
June 1, 2017

TCS5422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.17

TRACK SKEW

- Sectors are offset across tracks to allow time for head to reposition for sequential reads
- Without track skew, when head is repositioned sector would have already been passed

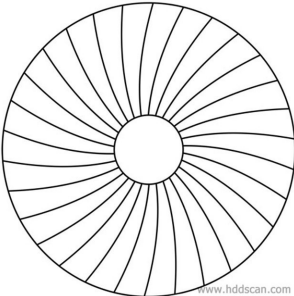


June 1, 2017

TCS5422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.18

TRACK SKEW - 2



June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.19

HDD CACHE

- Buffer to support caching reads and writes
- Improves drive response time
- Up to 128 MB, slowly have been growing
- Two styles
 - Writeback cache
 - Report write complete immediately when data is transferred to HDD cache
 - Dangerous
 - Writethrough cache
 - Reports write complete only when write is physically completed on disk

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.20

TRANSFER SPEED

- I/O Time $T_{i/o} = T_{seek} + T_{rotation} + T_{transfer}$
- The rate of I/O $R_{i/o} = \frac{Size_{transfer}}{T_{i/o}}$

	Cheetah 15K.5	Barracuda
Capacity	300 GB	1 TB
RPM	15,000	7,200
Average Seek	4 ms	9 ms
Max Transfer	125 MB/s	105 MB/s
Platters	4	4
Cache	16 MB	16/32 MB
Connects Via	SCSI	SATA

Disk Drive Specs: SCSI Versus SATA

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.21

I/O SPEED

- Random workload: 4KB random read on HDD
- Sequential workload: read 100MB contiguous sectors

		Cheetah 15K.5	Barracuda
Random	T_{seek}	4 ms	9 ms
	$T_{rotation}$	2 ms	4.2 ms
	$T_{transfer}$	30 microsecs	38 microsecs
	$T_{i/o}$	6 ms	13.2 ms
Sequential	$R_{i/o}$	0.66 MB/s	0.31 MB/s
	$T_{transfer}$	800 ms	950 ms
	$T_{i/o}$	806 ms	963.2 ms
	$R_{i/o}$	125 MB/s	105 MB/s

Disk Drive Performance: SCSI Versus SATA

There is a huge gap in drive throughput between random and sequential workloads

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.22

MODERN HDD SPECS

- See sample HDD configurations here:
- <https://www.hgst.com/products/hard-drives>

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.23

DISK SCHEDULING

- Disk scheduler: determine how to order I/O requests
- Multiple levels - OS and HW
- OS: provides ordering
- HW: further optimizes using intricate details of physical HDD implementation and state

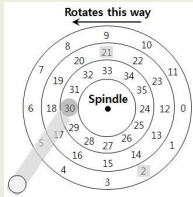
June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.24

SSTF – SHORTEST SEEK TIME FIRST

- Disk scheduling – which I/O request to schedule next
- Shortest Seek Time First (SSTF)
- Order queue of I/O requests by nearest track



SSTF: Scheduling Request 21 and 2
Issue the request to 21 → issue the request to 2

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.25

SSTF ISSUES

- Problem 1: HDD abstraction
- Drive geometry not available to OS. Nearest-block-first is a comparable alternate algorithm.
- Problem 2: Starvation
- Steady stream of requests for local tracks may prevent arm from traversing to other side of platter

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.26

DISK SCHEDULING ALGORITHMS

- **SWEEP**
- Single repeated passes across disk
- Issue: if request arrives for a recently visited track it will not be revisited until a full cycle completes
- **F-SCAN**
- Freeze request queue during sweep
- Cache arriving requests until later
- **Elevator (C-SCAN)** – circular scan
- Sweep from outer to inner track and reverse, inner to outer track, etc.

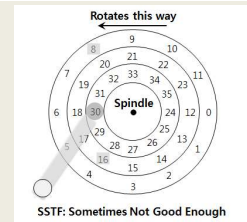
June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.27

SHORTEST TIME POSITIONING FIRST

- Determine next sector to read?
- On which track?
- On which sector?



SSTF: Sometimes Not Good Enough

On modern drives, both seek and rotation are roughly equivalent:
Thus, SPTF (Shortest Positioning Time First) is useful.

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.28

I/O MERGING

- Group temporary adjacent requests
- Reduce overhead
- Read (memory blocks): 33 8 34
- How long we should wait for I/O ?
- When do we know we have waited too long?

June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.29

QUESTIONS



June 1, 2017

TCSS422: Operating Systems [Spring 2017]
Institute of Technology, University of Washington - Tacoma

L18.30