Information in Storage Devices
049063 – EE Department, Technion

LECTURE 2: HDD AND SSD ACCESS
A Tale of Two Media Stars

- Has been around forever
- Improves, but looks the same
- Predictable performance

- Fast to respond
- Heavily hyped
  - High media exposure
- You know can do wonders
  - But most encounters less exciting
A Tale of Two Media Stars

- Has been around forever
- Improves, but looks the same
- Predictable performance

- Fast to respond
- Heavily hyped
  - High media exposure
- You know can do wonders
  - But most encounters less exciting
Hard-Disk Drive (HDD)

• Revolving disks, magnetic media
• Invented 1956 (IBM)
  • Size: two refrigerators
  • # disks: 50
  • Capacity: 4MB
• Capacity today: 8TB
  – Scaling with media and head technologies
HDD Access

- 1D view

sector
512B or 4KB
HDD Access

- 2D view
HDD Access

• 3D view

PBA = (Cylinder, Head, Sector) – CHS address
Access Time

$T(\text{Read}) = T(\text{write}) = T(\text{cyl. switch}) + T(\text{head switch}) + T(\text{rot})$

(1) \hspace{2cm} (2) \hspace{2cm} (3)

↓

seek time
**Random Access**

**Define:**
Access request \( \text{Req}(L\_\text{addr}, \text{rw}) \)

**Definition:**
A device is called *random access* if any sequence of requests \( \text{Req}_1, \text{Req}_2, \ldots \) is allowed, and all such sequences exhibit a similar response behavior.

HDD Read/Write ordering
- HDD R/W switch time
Seek Times

\[ T(c \rightarrow c') = \tau \frac{|c - c'|}{\#cyls - 1} \]

Normalized cylinder addresses:

\[ \gamma = \frac{c}{\#cyls - 1} \quad \gamma' = \frac{c'}{\#cyls - 1} \]

\[ T(c \rightarrow c') = \tau |\gamma - \gamma'| \]

full-seek time
Seek-Time Distribution

- Max
  - all possible $\gamma'$  \( \max[T(\gamma)] = \max[\tau \gamma, \tau (1 - \gamma)] \)
  - all possible $\gamma, \gamma'$  \( \max[T] = \tau \)

- What is the expected seek time?
  \[ E[T] = ? \]
Expectation given origin $\gamma$

- Expectation given $\gamma$ (uniform $\gamma'$)

\[
E[T(\gamma)] = \tau \left[ \gamma^2 - \gamma + \frac{1}{2} \right]
\]
Overall Expectation

• Expectation (uniform $\gamma, \gamma'$)

\[
E[T] = \mathbb{E}\{\mathbb{E}[T(\gamma)]\} = \frac{\tau}{3}
\]
Access Time

\[ T(\text{Read}) = T(\text{write}) = T(\text{cyl. switch}) + T(\text{head switch}) + T(\text{rot}) \]

(1) \hspace{2cm} (2) \hspace{2cm} (3)

or

serpentine mode \hspace{2cm} cylinder mode
Access Time

\[ T(\text{Read}) = T(\text{write}) = T(\text{cyl. switch}) + T(\text{head switch}) + T(\text{rot}) \]
Rotational Latency

\[ T(S \rightarrow S') = T_{rev} \frac{S \div S'}{\#\text{sectors/rev}} \]

- Max rotational latency
  \[ \max[T] = T_{rev} \]
- Expectation
  \[ E[T] = \frac{T_{rev}}{2} \]
Command Queueing

- HDD manages command queues
- Allowed out-of-order execution

\[
S \rightarrow S_i : i = \arg \min_{j \in 1, \ldots, N} T[S \rightarrow S_j]
\]

- Optimal choice of next:

\[
E[T_N] = E\left[ \min_{j \in 1, \ldots, N} T[S \rightarrow S_j] \right] = ?
\]

- Expected latency with N-queue

\[
T[S \rightarrow S_j] \sim U[0, T_{rev}]
\]
Conventional Recording
Shingled Recording

Track 5

Track 4

Track 3

Track 2

Track 1
Shingled Recording – No Random Write

Write

Erase
Shingled Recording Magnetics

track layout for shingled-recording

head field contours

head motion
corner head
progressive scans
cross-track
downtrack

1 2 3 4
Shingled Drive Tradeoff

Excess Capacity

Heads / media

shingled drive

Non-shingled

Observed Performance
Performance with Shingling
Solid-State Drive (SSD)

- Silicon-based array of memory cells
  - with standard interface (HDD replacement)
- Invented 1995 (M-Systems, Israel)
  - Uses NAND flash for maximal density
- More expensive, but much faster
- Capacity scales by “Moore’s law”
Flash: No Random-Access Erase

- New: Erase unit

- Physical erase of cells: only full blocks
- Write → program/erase
No In-Place Updates

$W$  
$L_{write}: \quad L_{\text{unit}}$  
$P_{write}$

$E_{\text{unit}}$ used

$E_{\text{unit}}$ used

$L_{\text{write}}$: $LBA=m$  
$P_{write}$

$W$  
$\text{update}$
Option 1: RMEW

W
update

P_read

modify

E_erase

P_write
Option 2: Invalidation

W
update

E_unit
LBA=m
used, invalid
LBA=m
P_write
Flash State Diagram

- Free
- Used
- Dirty
- L_write
- E_erase
- L_update (L_write elsewhere)
Issues

Option 1: RMEW
- Time
- Wear

Option 2: Invalidation
- Over-provisioning
- Indirection

SSD
1) #PBA > #LBA
2) Mapping layer