Lecture 1  Disks & Filesystems

Lecture 2  RAID

Lecture 3  Storage Systems and Virtualization
Lecture 1  Disks & Filesystems
  ▶  Revisions
  ▶  Performance
  ▶  Limitations and solutions

Lecture 2  RAID

Lecture 3  Storage Systems and Virtualization
Lecture 1  Disks & Filesystems
  ▶  Revisions
  ▶  Performance
  ▶  Limitations and solutions

Lecture 2  RAID
  ▶  build server filestore from (inexpensive) PC parts

Lecture 3  Storage Systems and Virtualization
Lecture 1  Disks & Filesystems
  ▶ Revisions
  ▶ Performance
  ▶ Limitations and solutions

Lecture 2  RAID
  ▶ build server filestore from (inexpensive) PC parts

Lecture 3  Storage Systems and Virtualization
  ▶ Logical Volume Management
  ▶ Storage Area Networks
Lecture 1  Disks & Filesystems
  ▶ Revisions
  ▶ Performance
  ▶ Limitations and solutions

Lecture 2  RAID
  ▶ build server filestore from (inexpensive) PC parts

Lecture 3  Storage Systems and Virtualization
  ▶ Logical Volume Management
  ▶ Storage Area Networks
  ▶ Solid State Disks
Learning Objectives - Storage 1

- Review disk and file system characteristics
- Understand the operational limitations of conventional disk usage
- Introduce simple solutions using multiple disks
Characterisation

- Write Once, Read Many (times) – \textit{WORM}

- Write Many, Read Many
Characterisation

- Write Once, Read Many (times) – WORM
  - CD-ROM, DVD, Blu-ray Disc

- Write Many, Read Many
  - Hard disk drive, tape drive
Characterisation

- Write Once, Read Many (times) – *WORM*
  - CD-ROM, DVD, Blu-ray Disc

- Write Many, Read Many
  - Hard disk drive, tape drive

- Write (not too) Many, Read Many
  - CD/DVD±RW (100s to 1000s)
  - Flash Memory (1000s to ...)
Characterisation

- Write Once, Read Many (times) – *WORM*
  - CD-ROM, DVD, Blu-ray Disc
  - Irreversible writes

- Write Many, Read Many
  - Hard disk drive, tape drive
  - Fully reversible writes (almost)

- Write (not too) Many, Read Many
  - CD/DVD±RW (100s to 1000s)
  - Flash Memory (1000s to ...)
  - Mostly reversible writes – “wear”
HDD Internals – tinyurl.com/disk-video

Source: http://systemspro.blogspot.co.uk/2011/09/hard-disk-drive.html
Hard Disk Drive Storage Structure

- **Capacity**
  - 8TB HDD (Seagate)
  - 6TB HDD (Seagate TBC - SSD!)

From above

- Track
- Sector
- Read/write head
- Cylinder
- Platter
- Surfaces
Hard Disk Drive Storage Structure

- Capacity
  - 2TB platter (2012/13)
  - 8TB HDD (Seagate 2014)
  - 10TB (WD HGST 2015)
  - 12TB (WD HGST 2017)
Hard Disk Drive Storage Structure

- **Capacity**
  - 2TB platter (2012/13)
  - 8TB HDD (Seagate 2014)
  - 10TB (WD HGST 2015)
  - 12TB (WD HGST 2017)
  - 60TB (Seagate TBC - SSD!)
Price trends

![Graph showing price trends for different types of storage devices over time. The graph illustrates the cost per gigabyte for storage SSD, server SSD, mission critical HDD, and business critical HDD. The data source is Gartner's Market Trends: Evolving HDD and SSD Storage Landscapes (October 2013).](Image)
Price trends

SSD Cost per Gigabyte over Time

USD/GB

Price trends

Backblaze Average Cost per Drive Size

By Quarter: Q1 2009 - Q2 2017
Hard Disk Drive Storage Structure

- **Capacity**
  - 2TB platter (2012/13)
  - 8TB HDD (Seagate 2014)
  - 10TB (WD HGST 2015)
  - 12TB (WD HGST 2017)
  - 60TB (Seagate TBC - SSD!)

- **Power consumption**
  - Spinning platters
  - Moving the heads (seek)
  - Reading/Writing
  - Controllers
  - Data transfer (I/O)

Hard Disk Drive Storage Structure

► Capacity
  ► 2TB platter (2012/13)
  ► 8TB HDD (Seagate 2014)
  ► 10TB (WD HGST 2015)
  ► 12TB (WD HGST 2017)
  ► 60TB (Seagate TBC - SSD!)

► Power consumption
  ► Spinning platters
  ► Moving the heads (seek)
  ► Reading/Writing
  ► Controllers
  ► Data transfer (I/O)

► Rotation speed
  ► 5400/7200/10000/15000

Hard Disk Attributes – *Performance*

Seek time

Search time

Transfer rate

Note: all values are average as they depend on many factors.
Hard Disk Attributes – Performance

Seek time  Time for the head to reach the target track.

Search time

Transfer rate

Note: all values are average as they depend on many factors.
Hard Disk Attributes – *Performance*

Seek time  Time for the *head* to reach the target *track*.

Search time  Time for the target *sector* to arrive under the *head*. Also called *rotational latency*.

Transfer rate

Note: all values are average as they depend on many factors.
Hard Disk Attributes – *Performance*

Seek time  Time for the **head** to reach the target **track**.

Search time  Time for the target **sector** to arrive under the **head**. Also called *rotational latency*.

Transfer rate  Amount of data that can be read / written per unit of time. Dependent on access patterns.

Note: all values are average as they depend on many factors.
Hard Disk Attributes – Performance

Seek time  Time for the head to reach the target track.

Search time  Time for the target sector to arrive under the head. Also called rotational latency.

Transfer rate  Amount of data that can be read / written per unit of time. Dependent on access patterns.

Aka. “sustained transfer rate” in contrast to “interface transfer rate”

Note: all values are average as they depend on many factors.
Hard Disk Attributes – *Performance*

**Seek time**  
Time for the **head** to reach the target **track**.

**Search time**  
Time for the target **sector** to arrive under the **head**. Also called *rotational latency*.

**Transfer rate**  
Amount of data that can be read / written per unit of time. Dependent on access patterns.

Aka. “sustained transfer rate” in contrast to “interface transfer rate”

**Disk access time** = seek time + search time + transfer time

Note: all values are average as they depend on many factors.
Host initiates read
sends a list of blocks to read
Disk access example

- Host initiates read
  sends a list of blocks to read
Disk access example

- Host initiates read
  sends a list of blocks to read
- Block schedule requested...

Rotation
Disk access example

- Host initiates read
  sends a list of blocks to read

- Block schedule requested...
  ... may not be optimal
Host initiates read
sends a list of blocks to read

Block schedule requested...
... may not be optimal
and leads to extra revolutions
Disk access example

- Host initiates read
  sends a list of blocks to read

- Block schedule requested...
  ... may not be optimal
  and leads to extra revolutions

- HDD internal processor
  optimizes the schedule

- No direct mapping from
  block numbers to the
  sector/track/cylinder position
  (high-level interfaces like ATA / SCSI)
Example HDD specs

HGST Western Digital He6 HUS726060ALA640

- Capacity 6TB
- Power consumption: 7.3/5.3/3.7 W
- Rotational speed: 7200 RPM
- Seek time: 8.5 ms
- Sustained transfer rate: 177 MB/sec
- Interface transfer rate: 600 MB/sec (SATA)
- Data buffer: 64 MB
- MTBF: 2,500,000 hours
- Price: £250 to £400 (Q1 2015)

Similar spec. Q1 2018: £155

Seagate IronWolf 6 TB ST6000VN0033 – SATA 6Gb/s – 7,200 rpm
Example: disk access time (1)

How long would it take on average to read / write a 512 byte sector on this disk?
Example: disk access time (1)

How long would it take on average to read / write a 512 byte sector on this disk?

Disk access time = seek time + search time + transfer time
Example: disk access time (1)

How long would it take **on average** to read / write a 512 byte sector on this disk?

\[
\text{Disk access time} = \text{seek time} + \text{search time} + \text{transfer time}
\]

**seek time:** 8.5 ms
Example: disk access time (1)

How long would it take **on average** to read / write a 512 byte sector on this disk?

Disk access time = seek time + search time + transfer time

seek time: 8.5 ms

search time: the disk must, on average, complete a half rotation

\[ 7200 \text{ RPM} \rightarrow \frac{0.5 \text{ rotations} \cdot 60 \text{ sec}}{7200 \text{ RPM}} = 4.16 \text{ ms} \]
How long would it take on average to read / write a 512 byte sector on this disk?

**Disk access time = seek time + search time + transfer time**

**seek time:** 8.5 ms

**search time:** the disk must, on average, complete a half rotation

\[
7200 \text{ RPM} \implies \frac{0.5 \text{ rotations} \cdot 60 \text{ sec}}{7200 \text{ RPM}} = 4.16 \text{ ms}
\]

**transfer time:**

\[
\frac{512 \text{ B}}{177 \cdot 10^6 \text{ B/sec}} = 2.89 \mu\text{s}
\]
Example: disk access time (1)

How long would it take on average to read / write a 512 byte sector on this disk?

Disk access time = seek time + search time + transfer time

seek time: 8.5 ms

search time: the disk must, on average, complete a half rotation

\[
7200 \text{ RPM} \quad \frac{0.5 \text{ rotations} \cdot 60 \text{ sec}}{7200 \text{ RPM}} = 4.16 \text{ ms}
\]

transfer time: \[
\frac{512 \text{ B}}{177 \cdot 10^6 \text{ B/sec}} = 2.89 \mu\text{s}
\]

access time = 8.5 + 4.16 + 2.89 \cdot 10^{-3} = 12.66 ms
Example: disk access time (2)

How long would it take on average to read / write 512 MB on this disk? (assuming sectors are “contiguous”)

Disk access time = seek time + search time + transfer time

seek time: 8 /five.pnum ms

search time: the disk must, on average, complete a half rotation /seven.pnum/two.pnum/zero.pnum/zero.pnum RPM = zero.pnum/five.pnum rotations · /seven.pnum/two.pnum/zero.pnum/zero.pnum RPM = four.pnum/one.pnum6 ms

transfer time: 
five.pnum/one.pnum/two.pnum · /one.pnum/zero.pnum 6 B /seven.pnum/seven.pnum · /one.pnum/zero.pnum 6 B /sec = two.pnum/nine.pnum s

access time = 8 /five.pnum · /one.pnum/zero.pnum − three.pnum + four.pnum/one.pnum6 · /one.pnum/zero.pnum − three.pnum + two.pnum/nine.pnum = two.pnum/nine.pnum s
Example: disk access time (2)

How long would it take on average to read / write 512 MB on this disk? (assuming sectors are “contiguous”)

Disk access time = seek time + search time + transfer time
Example: disk access time (2)

How long would it take **on average** to read / write 512 MB on this disk? (assuming sectors are “contiguous”)

\[
\text{Disk access time} = \text{seek time} + \text{search time} + \text{transfer time}
\]

seek time: 8.5 ms
Example: disk access time (2)

How long would it take **on average** to read / write 512 MB on this disk? (assuming sectors are “contiguous”)

**Disk access time = seek time + search time + transfer time**

**seek time:** 8.5 ms

**search time:** the disk must, on average, complete a half rotation

\[
7200 \text{ RPM} \quad \Rightarrow \quad \frac{0.5 \text{ rotations} \cdot 60 \text{ sec}}{7200 \text{ RPM}} = 4.16 \text{ ms}
\]
Example: disk access time (2)

How long would it take on average to read / write 512 MB on this disk? (assuming sectors are “contiguous”)

Disk access time = seek time + search time + transfer time

seek time: 8.5 ms

search time: the disk must, on average, complete a half rotation

\[
7200 \text{ RPM} \implies \frac{0.5 \text{ rotations} \cdot 60 \text{ sec}}{7200 \text{ RPM}} = 4.16 \text{ ms}
\]

transfer time: \[
\frac{512 \cdot 10^6 \text{ B}}{177 \cdot 10^6 \text{ B/sec}} = 2.89 \text{ s}
\]
Example: disk access time (2)

How long would it take on average to read / write 512 MB on this disk? (assuming sectors are “contiguous”)

Disk access time = seek time + search time + transfer time

seek time: 8.5 ms

search time: the disk must, on average, complete a half rotation

\[
7200 \text{ RPM} \quad \Rightarrow \quad \frac{0.5 \text{ rotations} \cdot 60 \text{ sec}}{7200 \text{ RPM}} = 4.16 \text{ ms}
\]

transfer time: \[
\frac{512 \cdot 10^6 \text{ B}}{177 \cdot 10^6 \text{ B/sec}} = 2.89 \text{ s}
\]

access time = \( 8.5 \cdot 10^{-3} + 4.16 \cdot 10^{-3} + 2.89 = 2.9 \text{ s} \)
File System Review

- Naming service
  - files
  - directories
  - links
- Storage service
  - “vector of bytes”
  - owners, permissions...
- Data and metadata
- Space allocation
  - contiguous
  - linked
  - indexed
- Recovery
  - chkdsk, fsck
Problems with disks
Problems with disks

Small
Problems with disks

Small  Slow
Problems with disks

Small  Slow  Unreliable
Disks are (were?) too small
Disks are (were?) too small

1956 first HDD IBM 350

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
Disks are (were?) too small

1956 first HDD IBM 350: \( \sim 3.5 \text{ MB} \)

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
Disks are (were?) too small

1956 first HDD IBM 350: $\sim$ 3.5 MB (enough to store one selfie!)

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
Disks are (were?) too small

1956 first HDD IBM 350: \(\sim 3.5 \text{ MB}\) (enough to store one selfie!)
2015 first 10 TB disk: 1000s of times smaller, \(3 \times 10^6\times\text{ capacity}\)

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
Disks are (were?) too small

1956 first HDD IBM 350: $\sim 3.5$ MB (enough to store one selfie!)
2015 first 10 TB disk: 1000s of times smaller, $3 \cdot 10^6 \times$ capacity

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
Disks are (were?) too small

1956 first HDD IBM 350: ~ 3.5 MB (enough to store one selfie!)
2015 first 10 TB disk: 1000s of times smaller, \(3 \cdot 10^6 \times \) capacity

\(10^{10}\) higher storage density in 60 years

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
Disks are (were?) too small

1956 first HDD IBM 350: \(\sim 3.5\) MB (enough to store one selfie!)
2015 first 10 TB disk: 1000s of times smaller, \(3 \cdot 10^6 \times\) capacity

\(10^{10}\) higher storage density in 60 years: is this enough?

Source: https://www-03.ibm.com/ibm/history/exhibits/storage/storage_350.html
If one disk is not enough ...

Use multiple disks
If one disk is not enough ...

Use multiple disks

- Independent disks
If one disk is not enough ...

Use multiple disks

- Independent disks
- Can we have a single volume with the combined capacity?
If one disk is not enough ...

Use multiple disks

- Independent disks
- Can we have a single volume with the combined capacity?
- Storage virtualization
If one disk is not enough ...

Use multiple disks

- Independent disks
- Can we have a single volume with the combined capacity?
- Storage virtualization

**Redundant Array of Independent Disks**
Disks are too slow

Slow because of:

- High seek time
- Reduce the number of times the head must move
- Multiple platters ⇒ more tracks/sectors per cylinder
- High search time (aka. rotational latency)
- Increase the rotation speed (e.g., server disks up to /one.pnum/five.pnum/zero.pnum/zero.pnum RPM)
- Low sustained transfer rate
- Increase rotation speed (physical limitations)
- Increase the recording density (physical limitations)
- Apply cache and prefetch principles
- "Stripe" /uniFB01le system across multiple disks
Disks are too slow

Slow because of:
  ▶ High seek time
Disks are too slow

Slow because of:

- High seek time

- High search time (aka. rotational latency)
Disks are too slow

Slow because of:
- High seek time
- High search time (aka. rotational latency)
- Low sustained transfer rate

Reduce the number of times the head must move
- Multiple platters = more tracks/sectors per cylinder

Increase the rotation speed (e.g., server disks up to /one.pnum/five.pnum/zero.pnum/zero.pnum RPM)

Increase rotation speed (physical limitations)
Increase the recording density (physical limitations)

Apply cache and prefetch principles
- “Stripe” /uniFB01le system across multiple disks
Disks are too slow

Slow because of:

- High seek time
  - Reduce the number of times the head must move

- High search time (aka. rotational latency)

- Low sustained transfer rate
Disks are too slow

Slow because of:

- High seek time
  - Reduce the number of times the head must move
  - Multiple platters ➔ more tracks/sectors per cylinder
- High search time (aka. rotational latency)
- Low sustained transfer rate
Disks are too slow

Slow because of:

- High seek time
  - Reduce the number of times the head must move
  - Multiple platters → more tracks/sectors per cylinder
- High search time (aka. rotational latency)
  - Increase the rotation speed (e.g., server disks up to 15000 RPM)
- Low sustained transfer rate
Disks are too slow

Slow because of:

- High seek time
  - Reduce the number of times the head must move
  - Multiple platters $\rightarrow$ more tracks/sectors per cylinder

- High search time (aka. rotational latency)
  - Increase the rotation speed (e.g., server disks up to 15000 RPM)

- Low sustained transfer rate
  - Increase rotation speed (physical limitations)
  - Increase the recording density (physical limitations)
Disks are too slow

Slow because of:

- High seek time
  - Reduce the number of times the head must move
  - Multiple platters $\Rightarrow$ more tracks/sectors per cylinder
- High search time (aka. rotational latency)
  - Increase the rotation speed (e.g., server disks up to 15000 RPM)
- Low sustained transfer rate
  - Increase rotation speed (physical limitations)
  - Increase the recording density (physical limitations)
  - Apply cache and prefetch principles
Disks are too slow

Slow because of:

- High seek time
  - Reduce the number of times the head must move
  - Multiple platters $\rightarrow$ more tracks/sectors per cylinder
- High search time (aka. rotational latency)
  - Increase the rotation speed (e.g., server disks up to 15000 RPM)
- Low sustained transfer rate
  - Increase rotation speed (physical limitations)
  - Increase the recording density (physical limitations)
  - Apply cache and prefetch principles
  - “Stripe” file system across multiple disks
Solution: Disk Striping (RAID 0)

- Split data evenly across multiple disks
Solution: Disk Striping (RAID 0)

- Split data evenly across multiple disks
- Distribute fixed-size “stripes” of a virtual volume
Solution: Disk Striping (RAID 0)

- Split data evenly across multiple disks
- Distribute fixed-size “stripes” of a virtual volume
- Illusion of faster and larger disk
Solution: Disk Striping (RAID 0)

- Split data evenly across multiple disks
- Distribute fixed-size “stripes” of a virtual volume
- Illusion of **faster** and **larger** disk

**BUT lower reliability!**
Disks are unreliable

- Mechanical components subject to wear
- Partial failure: sectors go bad
- Total failure: no data recoverable

If reliability cannot be improved:
- Tolerate failures
- Fault-tolerance through redundancy
- Disk "mirror"
Disks are unreliable

- Mechanical components subject to wear
Disks are unreliable

- Mechanical components subject to wear
- Partial failure: sectors go bad
- Total failure: no data recoverable
Disks are unreliable

- Mechanical components subject to wear
- Partial failure: sectors go bad
- Total failure: no data recoverable

- If reliability cannot be improved: **tolerate failures**
  - Fault-tolerance through redundancy
  - Disk “mirror”
Solution: Disk Mirroring (RAID 1)

- Use two (or more) redundant disks
Solution: Disk Mirroring (RAID 1)

- Use two (or more) redundant disks
- Write to each (same, replicated data)
Solution: Disk Mirroring (RAID 1)

- Use two (or more) redundant disks
- Write to each (same, replicated data)
- Read from either (possibly choose “nearest” for performance)
Solution: Disk Mirroring (RAID 1)

- Use two (or more) redundant disks
- Write to each (same, replicated data)
- Read from either (possibly choose “nearest” for performance)
- If one fails: use the other and re-create a new copy (slowly)
Nested RAID: RAID 1+0 (aka. RAID 10)

Operation continues in case of disk failure
Can tolerate failures as long as no mirror loses all drives
Nested RAID: RAID 1+0 (aka. RAID 10)

Operation continues in case of disk failure
Nested RAID: RAID 1+0 (aka. RAID 10)

- Operation continues in case of disk failure
- Can tolerate failures as long as no mirror loses all drives
Summary: Problems and (simple) Solutions

- Disks are too small
- Disks are too slow
- Disks are unreliable
Summary: Problems and (simple) Solutions

- Disks are too small
  - Fixed: use multiple disks (possibly striped)
- Disks are too slow
- Disks are unreliable

What happens when we migrate a Virtual Machine?
Summary: Problems and (simple) Solutions

- Disks are too small
  - Fixed: use multiple disks (possibly striped)
- Disks are too slow
  - Fixed: disk striping (RAID 0)
- Disks are unreliable
Summary: Problems and (simple) Solutions

- Disks are too small
  - Fixed: use multiple disks (possibly striped)
- Disks are too slow
  - Fixed: disk striping (RAID 0)
- Disks are unreliable
  - Fixed: disk mirroring (RAID 1)
Summary: Problems and (simple) Solutions

- Disks are too small
  - Fixed: use multiple disks (possibly striped)
- Disks are too slow
  - Fixed: disk striping (RAID 0)
- Disks are unreliable
  - Fixed: disk mirroring (RAID 1)
- Disks may be in the wrong place!
Summary: Problems and (simple) Solutions

- Disks are too small
  - Fixed: use multiple disks (possibly striped)
- Disks are too slow
  - Fixed: disk striping (RAID 0)
- Disks are unreliable
  - Fixed: disk mirroring (RAID 1)
- Disks may be in the wrong place!
  - What happens when we migrate a Virtual Machine?
Summary: Problems and (simple) Solutions

▶ Disks are too small
  ▶ Fixed: use multiple disks (possibly striped)
▶ Disks are too slow
  ▶ Fixed: disk striping (RAID 0)
▶ Disks are unreliable
  ▶ Fixed: disk mirroring (RAID 1)
▶ Disks may be in the wrong place!
  ▶ What happens when we migrate a Virtual Machine?

Better solutions on Monday