Some Examples

(for DATA REPRESENTATION)

NOTE: Examples are not lectured,
but it is highly recommended that you go through them to check your understanding.

Example: Disk Organization – I

- ✓ Consider a (very old) disk with the following characteristics:

  block size B=512 bytes

  gap size G=128 bytes

  20 sectors per track

  400 tracks per surface

  15 double-sided platters

  disk platters rotates at a speed of 2400 RPM

  average seek time 30 ms

  (all numbers are given as a factor of 2, i.e., KB = 2^{10}, MB = 2^{20})

- ✓ What is the total capacity of a track

⇨ Total = 20 * (512+128) = 12800 bytes = 12.5 KB

- ✓ What is its useful capacity of a track (excluding interblock gaps)

⇨ Useful = 20 * 512 = 10240 bytes = 10 KB

Example: Disk Organization – II

- ✓ How many cylinders are there?
- ⇨ Number of cylinders = number of tracks = <u>400</u>
- ✓ What is the total capacity of a cylinder?
- ⇨ Total = 15 * 2 * 20 * (512+128) = 384000 B = <u>375 KB</u>
- ✓ What is the useful capacity of a cylinder?
- ⇨ Useful = 15 * 2 * 20 * 512 = 307200 B = <u>300 KB</u>
- ✓ What is the total capacity of the disk?
- ⇨ Total = 15 * 2 * 400 * 20 * (512+128) = 153600000 B = <u>146.5 MB</u>
- ✓ What is the useful capacity of a disk?
- ⇨ Useful = 15 * 2 * 400 * 20 * 512 = <u>117.18 MB</u>
- ✓ What is the average rotational delay rd?
- ⇨ rd = (time for one disk revolution) / 2 = 25 / 2 = <u>12.5 ms</u>

Example: Disk Organization – III

- ✓ What is the total transfer rate (ttr)?
- ⇨ ttr = (total track size in bytes) / (time for one disk revolution)
  = 12800 B / (60 / 2400) s = 12800 B / 25 ms = <u>500 KB/s</u>

- ✓ What is the efficient (formatted) transfer rate (etr)?
- ⇨ etr = (useful capacity of a track) / (time for one disk revolution)
  = 10240 B / (60 / 2400) s = 10240 B / 25 ms = <u>400 KB/s</u>

- ✓ What is the sector transfer time (stt)?
- ⇨ 512 / (512 + 128) * 360 / 20 = 14.4 degrees per sector
  stt = 14.4 / 360 * 25 ms = <u>1 ms</u>

Example: Disk Organization – IV

- ✓ What is the time to transfer 4 KB (sectors in same track)?
- ⇨ 7 gaps and 8 sectors must pass under disk head
  14.4 degrees per sector
  128 / (512 + 128) * 360 / 20 = 3.6 degrees per gap
  btt = ((8 * 14.4) + (7 * 3.6)) / 360 * (60 / 2400) s
  = 140.4 / 360 * 25 ms = <u>9.75 ms</u>
- ⇨ an approximate value may be calculated using the efficient transfer time or *bulk transfer rate* (btr)
  btr = (B/(B + G)) * ttr = 0.8 * 500 = 400 KB/s
  time to transfer 4 KB ≈ 4096 / 400 KB/s = <u>10 ms</u>

Example: Disk Organization – V

- ✓ How much time does it take (on average) to locate and transfer a single sector given its address?
- ⇨ average time = s + rd + stt = 30 + 12.5 + 1 = 43.5 ms

- ✓ Calculate the average time to transfer 20 random sectors
- ⇨ time = 20 * (s + rd + stt) = 20 * 43.5 = 870 ms

- ✓ How much time is saved if the 20 sectors are stored contiguously (on same track)?
- ⇨ 20 sectors and 19 gaps = 356.6 degrees
  time = 30 + 12.5 + (356.6/360 * 25) = 67.25 ms
  you save = 870 – 67.25 = 802.75 ms (92.2 %)

Example: Disk Organization – VI

✓ Assume a process uses 100 ms to process the data in a 4 KB block. How much time is saved using double buffering compared to single buffering if we shall process 10 blocks?

⇨ single = 10 * (retrieval time + processing time)
= 10 * (9.75 + 100) = <u>1097.5 ms</u>

double = retrieval time + 10 * processing time
= 9.75 + 10 * 100 = <u>1009.75 ms</u>

savings = 1097.5 – 1009.75 = <u>87.75 ms</u> (8 %)

Example: Disk Organization – VI

- ✓ Assume we want to read an MPEG movie in DVD quality (average 3.5 Mbps). How many disks do we need in parallel to achieve requested bandwidth assuming
  (i) random placement
- ⇨ 3.5 Mbit/s ≈ 458.8 KB/s
  per-disk-transfer-rate = 512 byte / (30+12.5+1) ms ≈ 11.5 KB/s
  number of needed disks = ceiling(458.8 / 11.5) = <u>40 disks</u>

- ✓ (i) random placement, but increasing block size to 4 KB
- ⇨ per-disk-transfer-rate = 4096 byte / (30+12.5+9.75) ms ≈ 214 KB/s
  number of needed disks = ceiling(458.8 / 214) = <u>3 disks</u>

Example: Disk Organization – VI

- ✓ The disk we have used is OLD. What would the total transfer time (using bulk/efficient transfer time) be for a 4 KB blocks if use specifications from Seagate X15 to change
  (i) only data density on the platter to 617 sectors per track (20)
- ⇨ the new track can hold 315904 B data (10240 B)
  etr = (useful capacity of a track) / (time for one disk revolution)
  = (617 * 512) B / (60 / 2400) s = 12.05 MB/s (400 KB/s)
  total = 30 + 12.5 + (4 / 12340) = 42.5003 ms (52.5 ms)
- ✓ (ii) only seek time to 3.6 ms (30 ms)
- ⇨ total = 3.6 + 12.5 + 10 = 26.1 ms (52.5 ms)
- ✓ (iii) only rotational speed to 15.000 RPM (2400 RPM)
- ⇨ etr = 10240 / (60 / 15000) = 10240 / 0.004 = 2.44 MB/s (400 KB/s)
  total = 30 + 2 + (4 / 2500) = 32.502 ms (52.5 ms)
- ✓ (iv) all the above
- ⇨ etr = 315904 / 0.004 = 75.31 MB/s (400 KB/s)
  total = 3.6 + 2 + (4 / 77125) = 5.60005 ms (52.5 ms)

Example: File Organization – I

- A file has r=20000 fixed-length STUDENT records. Each record has the following fields:
  - NAME (30 bytes),
  - SSN (9 bytes),
  - ADDRESS (40 bytes),
  - PHONE (9 bytes),
  - BIRTHDATE (8 bytes),
  - SEX (1 byte),
  - MAJORDEPTCODE (4 bytes),
  - MINORDEPTCODE (4 bytes),
  - CLASSCODE (4 bytes),
  - DEGREEPROGRAM (3 bytes)

  The file is stored on the disk used in the previous example.

Example: File Organization – II

- ✓ If the record header have 5 fields of 4 bytes for pointers to schema, record length, etc. Calculate the record size R in bytes

⇨ R = record header + record fields
= (4+4+4+4+4) + (30+9+40+9+8+1+4+4+4+3)
= <u>132 B</u>

- ✓ Calculate the blocking factor bfr

⇨ bfr = floor(B / R) = floor(512 / 132) = <u>3 records per block</u>

- ✓ Calculate the number of file blocks b needed assuming
  (i) an unspanned organization.

⇨ b = ceiling(r / bfr) = ceiling(20000 / 3) = <u>6667 blocks</u>

(ii) a spanned organization (assume 3 * 2 B extra for record header)

⇨ b = 138 bytes per record * 20000 records / 512 B = <u>5391 blocks</u>

Example: File Organization – III

- ✓ If the 32-bit processor requires 4-byte alignment. What is the new record size?
- ⇨ For each field: size = ceiling(old_size/4) * 4
  R = <u>144 B</u>
- ✓ If the block header have 3 fields of 4 bytes for ID, pointer to first record, and pointer to overflow. What is the new blocking factor?
- ⇨ bfr = floor((512 – 12) / 144) = <u>3 records per block</u>
- ✓ What is the new number of file blocks b needed assuming
  (i) an unspanned organization.
- ⇨ b = ceiling(r / bfr) = ceiling(20000 / 3) = <u>6667 blocks</u>
  (ii) a spanned organization (assume 3 * 2 B extra for record header)
- ⇨ b = 156 bytes per record * 20000 records / 500 B = <u>6240 blocks</u>

Example: File Organization – IV

- ✓ What is the time to retrieve the whole file assuming random placement on disk for
  (i) unspanned records?
- ⇨ Average time to retrieve a block is 43.5 ms
  Total time = 6667 * 43.5 ms = <u>290 s</u>

  (i) spanned records?
- ⇨ Total time = 6240 * 43.5 ms = <u>271 s</u>
- ✓ How much time is saved using larger block sizes (4096 B) when using unspanned records?
- ⇨ bfr = floor((4096 – 12) / 144) = 28 records per block
  b = ceiling(20000 / 28) = 715 blocks
  time to transfer one block is = 30 + 12.5 + 9.75 = 52.25 ms
  total time = 52.25 * 715 = 37 s
  savings = 290 – 37 = <u>253 s</u> (87 %)

Example: File Organization – V

- ✓ Assume we want to change the order of the fields of all records, i.e., update all fields. The update processing operation takes 1 µs (10^{-6}) per record. What is the time to update the database using contiguous placement and
  (i) single buffering, 512 B blocks, unspanned records
- ⇨ 3 records per block → 6667 blocks,
  update operation: read, update, write, verify
  total time = 6667 * (43.5 + ($3*10^{-3}$) + 43.5 + 25) = <u>747 s</u>
- ✓ (ii) single buffering, 4 KB blocks, unspanned records
- ⇨ 28 records → 715 blocks
  total time = 715 * (52.25 + ($28*10^{-3}$) + 52.25 + 25) = <u>93 s</u>

- ⇨ <u>NOTE</u>: processing time consumes about 0.003 % and 0.022 % of the total time, respectively → disk I/O consumes VERY much time