Operating system

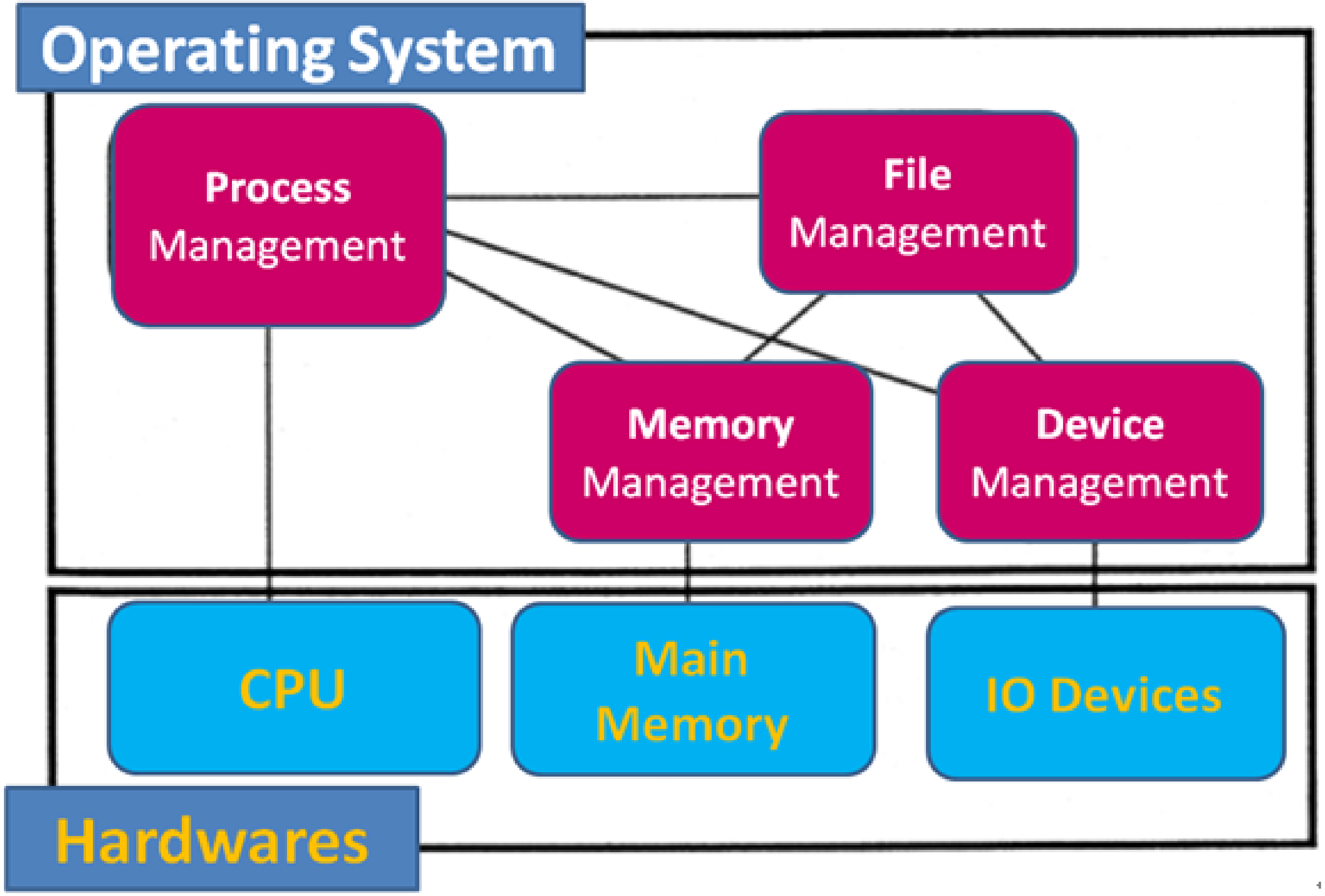
Part XI: IO System (Other)

Goals

- Know the advanced services provided by most Oss
  - Scheduling algorithms for disk I/O requests
  - SPOOLING [虚拟脱机技术]
  - RAID
    - Redundant backup for the safe storage
  - USB
    - Universal interface for diverse devices
  - NAS, SAN, ...
    - Scattered storage

Review

- We also have mentioned the four components of OS



Adv anc ed I O

- Scheduling algorithms for disk I/O req uests
- SPOOLing
- RAID
  - Redundant backup for the safe storage
- USB
  - Universal interface for diverse devices
- NAS, SAN, ...
  - Scattered storage

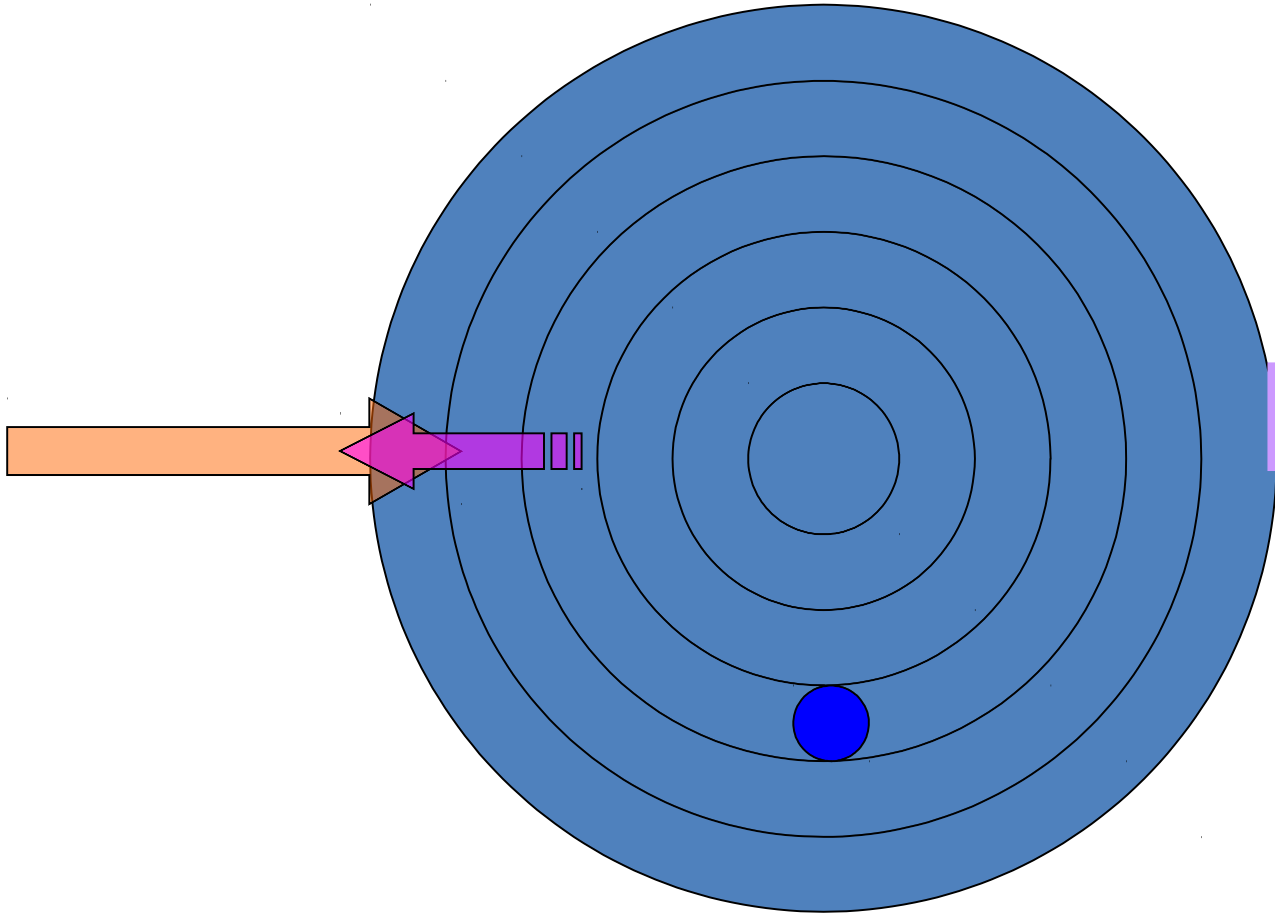
OS is responsible for using hardware efficiently

- For disk drives ☾ fast access time and disk bandwidth

- Access time has two major components
  - Seek time is the time for the disk to move the heads to the cylinder containing the desired sector
    - Seek time ≈ seek distance
    - Minimize seek time
  - Rotational latency [旋转延迟] the additional time waiting for the disk to rotate the desired sector to the disk head
    - Difficult for OS
- Disk bandwidth is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer

Disk Optimizations

- **Disk latency** time: Rotational delay waiting f or proper sector to rotate under R/W head
- **Disk seek** time: Delay while R/W head moves to the destination track/cylinder
- **Transfer** Time: Time to copy bits from disk s urface to memory
- **Access Time** = seek + latency + transfer



Seek time

Rotational delay

Transfer time

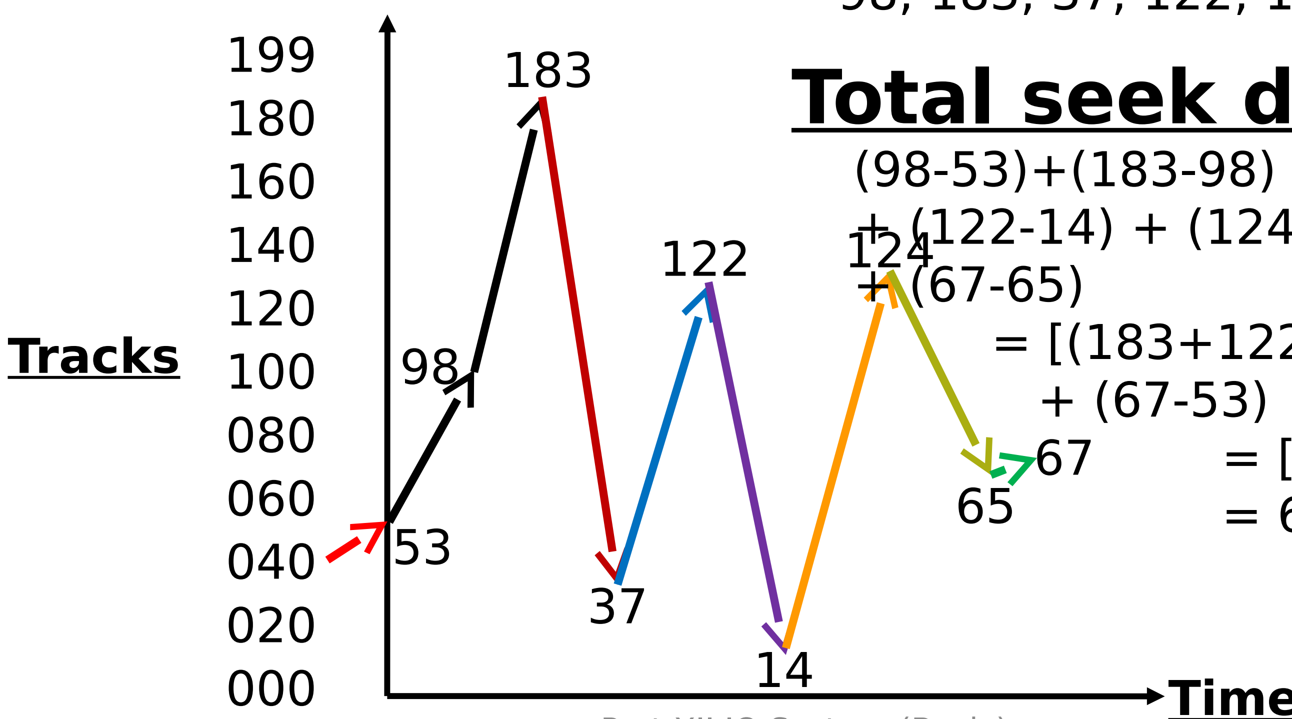
Several algorithms exist to schedule the disk I/O requests

- We illustrate them with a request queue (0-199).
  - 98, 183, 37, 122, 14, 124, 65, 67
  - After visiting 40, current Head pointer is at 53

FCFS [先来先服务算法]

- ***First come, first serve (FCFS):*** requests are served in the order of arrival
  - \+ Fair among requesters
  - \- Poor for accesses to random disk blocks

98, 183, 37, 122, 14, 124, 65, 67



<u>Total seek distance</u>:

(98-53)+(183-98) + (183-37) + (122-

\+ (122-14) + (124-14) + (124-65)

\+ (67-65)

= [(183+122+124) – (37+14+6

\+ (67-53)

= [429 - 116]*2 + 14

= 640

SSTF [最短寻道时间优先]

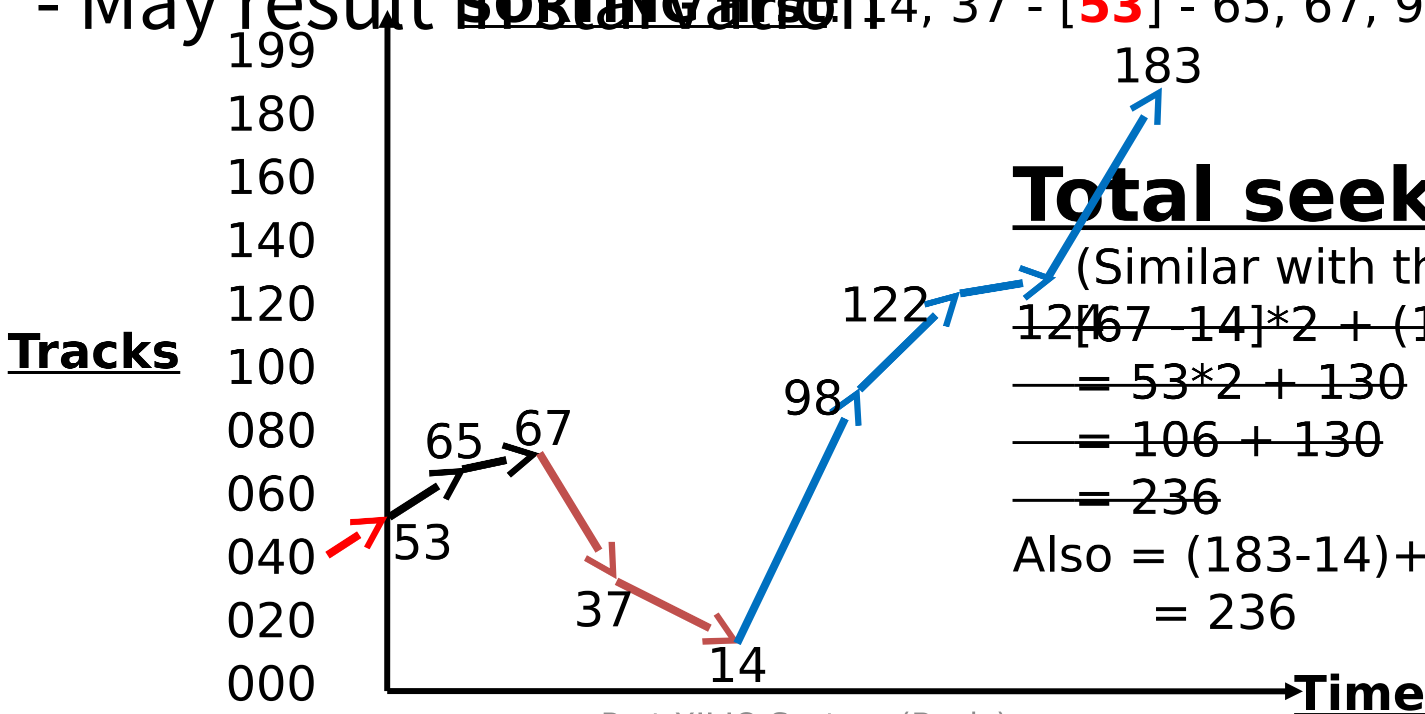
- ***Shortest seek time first (SSTF):*** picks the request that is closest to the current disk arm position

  \+ Good at reducing seeks

  \- May result in starvation

98, 183, 37, 122, 14, 124, 65, 67

SORTING first: 14, 37 - [**53**] - 65, 67, 98, 122, 124, 1



Total seek distanc

(Similar with the **FCFS**)

~~[67-14]*2 + (183-53)~~

~~= 53*2 + 130~~

~~= 106 + 130~~

~~= 236~~

Also = (183-14)+(67-14)+(67-

= 236

SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, serving requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues.
  - Sometimes called **the elevator algorithm [电梯算法]**.

Elevator Algorithms

PPTs.2012\PPTs from others\u.cs.biu.ac.il_~ariel_download_os381_ppts\os10-3_dsk.ppt

- Algorithms based on the common elevator principle.
- Four combinations of Elevator algorithms:
  - Service in both directions or in only one direction.
  - Go until last cylinder or until last I/O request.

Go until / Direction	Go until the last cylinder	Go until the last request
Service both directions	**Scan**	**Look**
Service in only one direction	**C-Scan**	**C-Look**

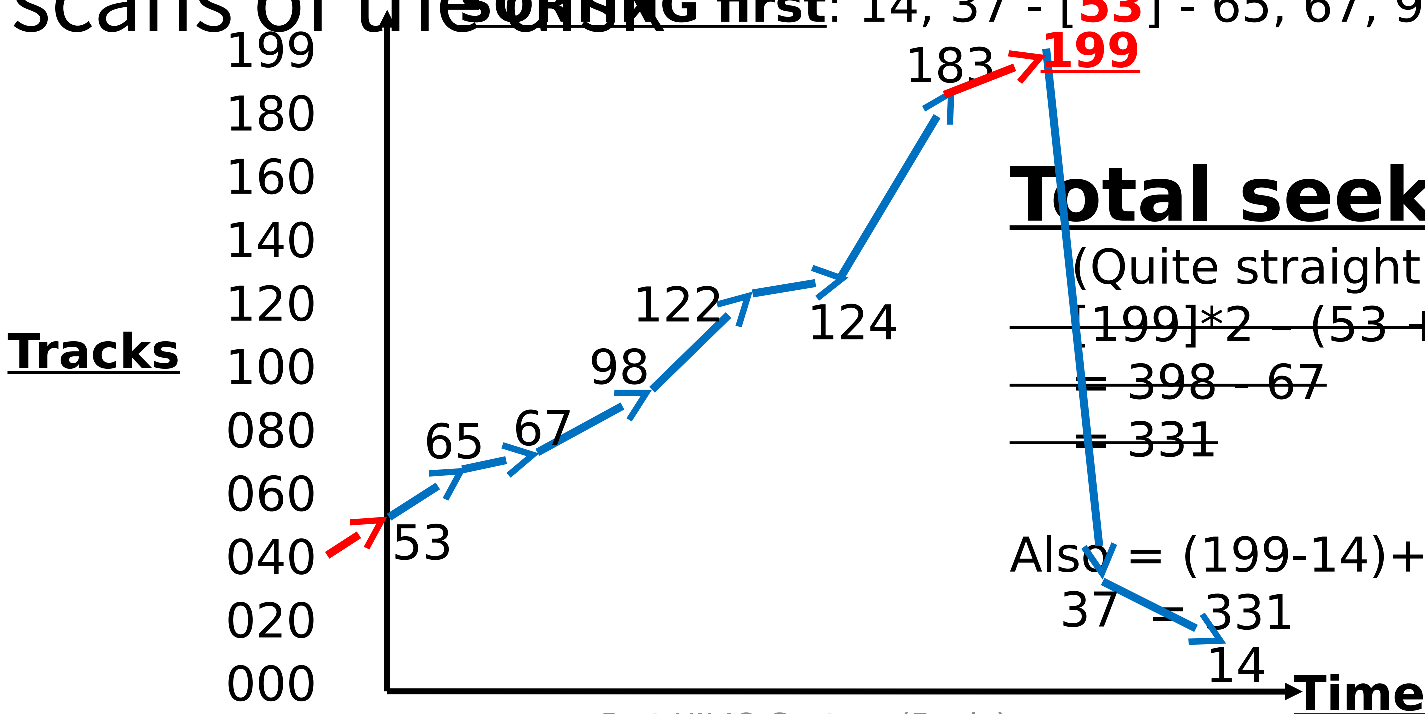
SCAN

- ***SCAN:*** takes the closest request in the direction of travel (an example of elevator algorithm)

  - a new request can wait for almost two full scans of the disk

98, 183, 37, 122, 14, 124, 65, 67

SORTING first: 14, 37 - [**53**] - 65, 67, 98, 122, 124, 1



Total seek distanc

(Quite straight)

~~[199]*2 – (53 + 14)~~

~~= 398 - 67~~

~~= 331~~

Also = (199-14)+(199-53)

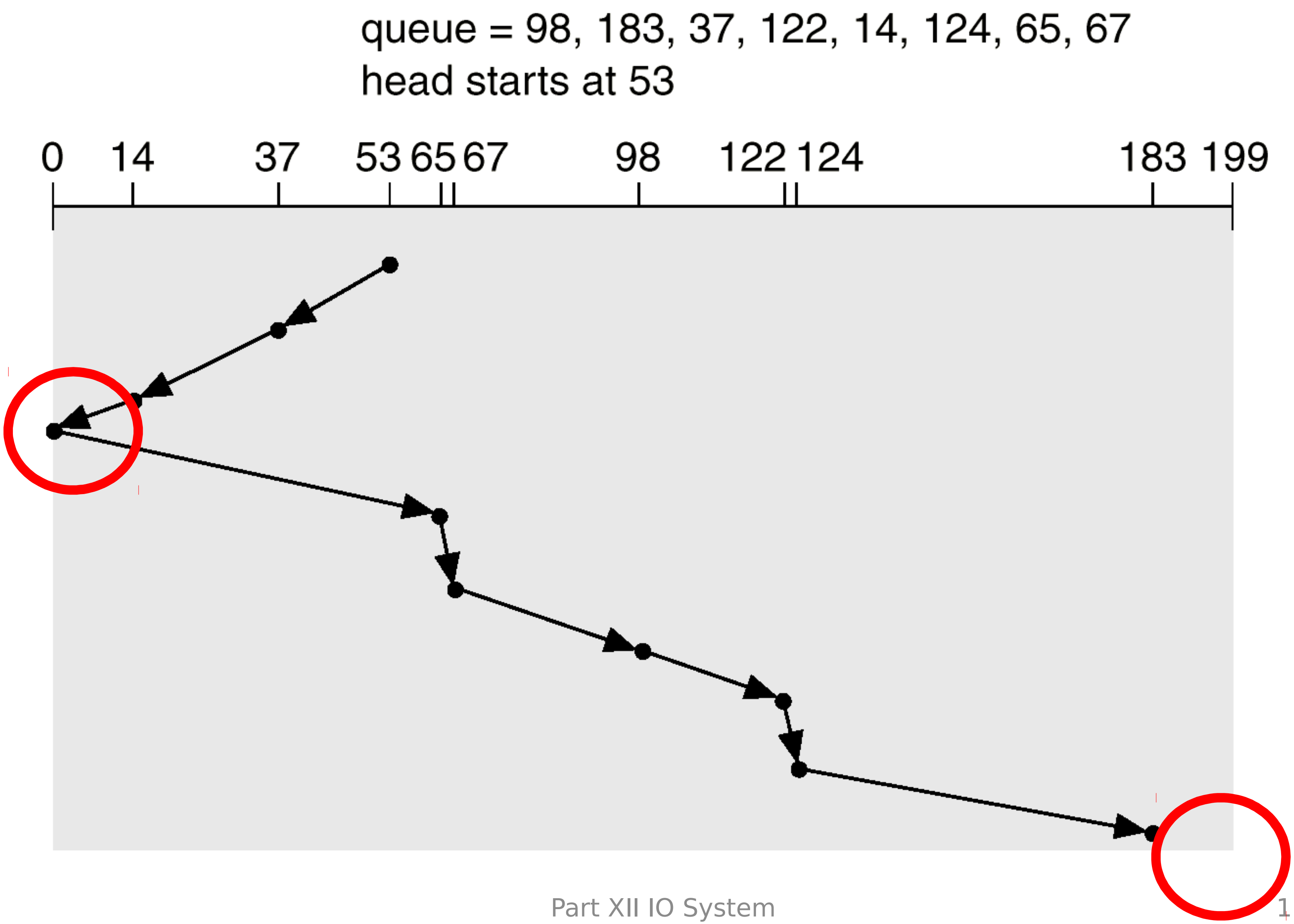
= 331

Get the **end** of each side [摸边]

PPTs from others\www.cs.mtu.edu_~shene_OS-slides\chap13-IO-Sys.pdf

- This algorithm requires one more piece of information:
  - the disk head movement direction, inward or outward.
- The disk head starts at one **end**, and move toward the other in the current direction.
- At the other **end**, the direction is reversed and service continues.
  - Some authors refer the SCAN algorithm as the elevator algorithm.
  - However, to some others the elevator algorithm means the LOOK algorithm.

SCAN

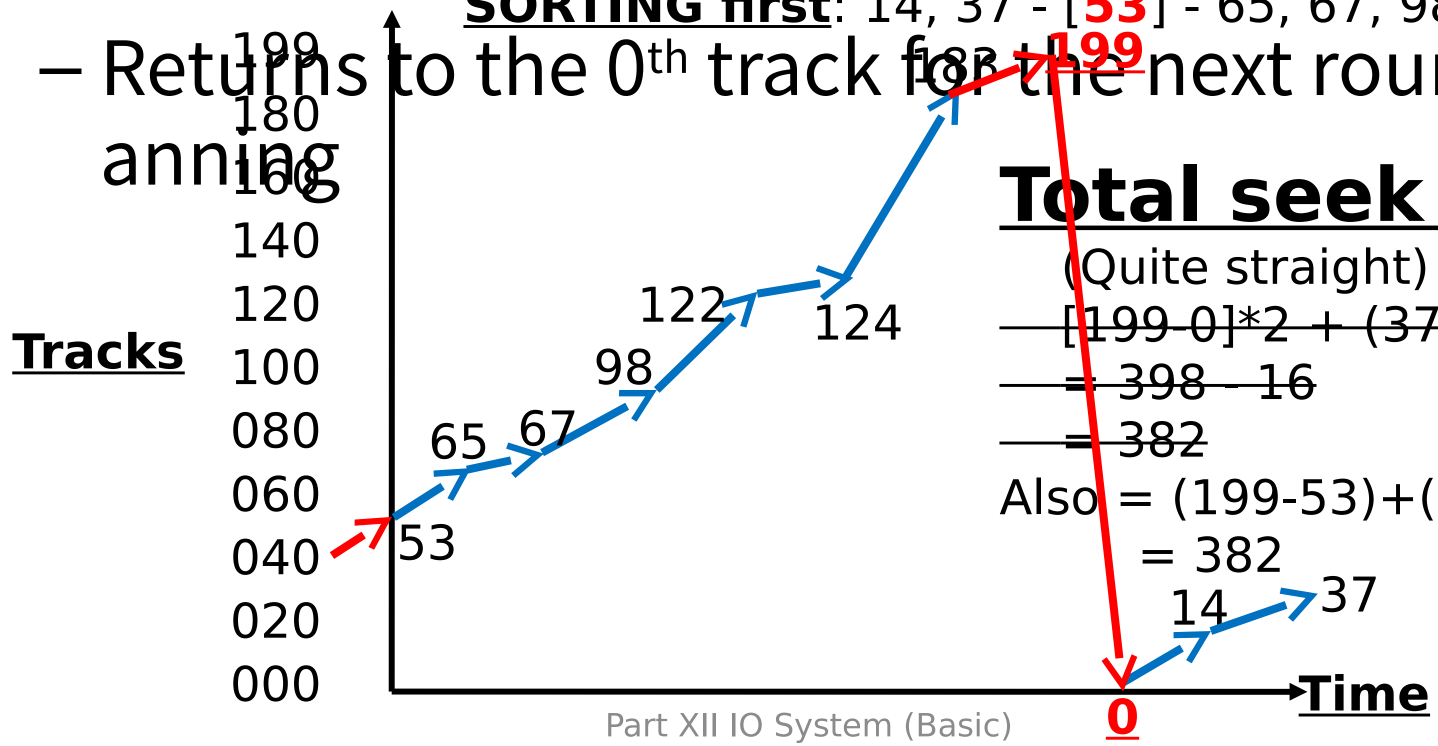

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
0 14 37 53 65 67 98 122 124 183 199

C-SCAN

- ***Circular SCAN (C-SCAN):*** disk arm always serves requests by scanning in one directio n.
  - Once the arm finishes scanning for one directio n
  - Returns to the 0th track for the next round of sc anning

98, 183, 37, 122, 14, 124, 65, 67

SORTING first: 14, 37 - [**53**] - 65, 67, 98, 122, 124, 1



Total seek distanc

(Quite straight)

~~[199-0]*2 + (37-53)~~

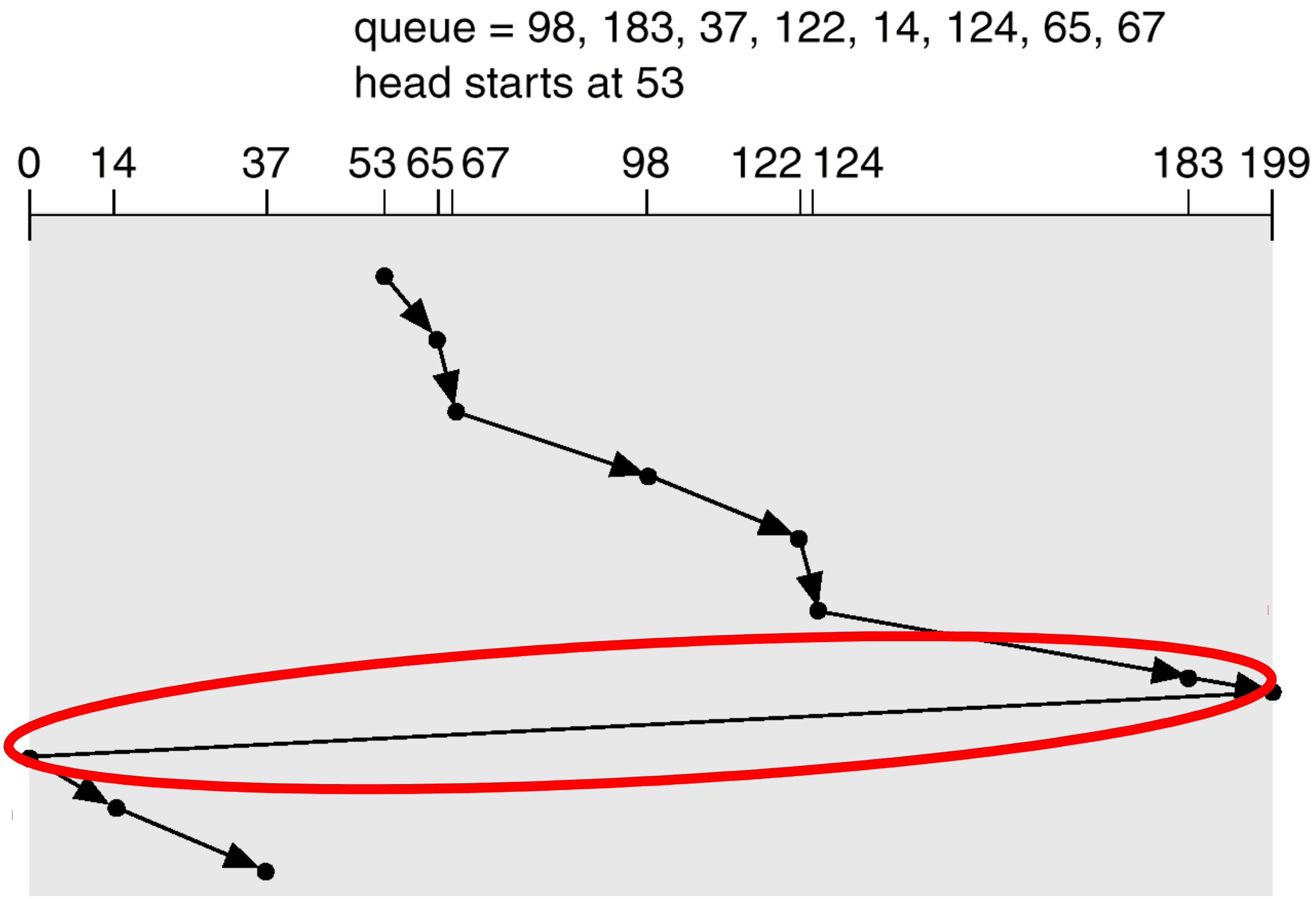
~~= 398 - 16~~

~~= 382~~

Also = (199-53)+(199-0)+(37-0

= 382

C-SCAN


queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
0
14
37
53
65
67
98
122
124
183
199

LOOK scheduling

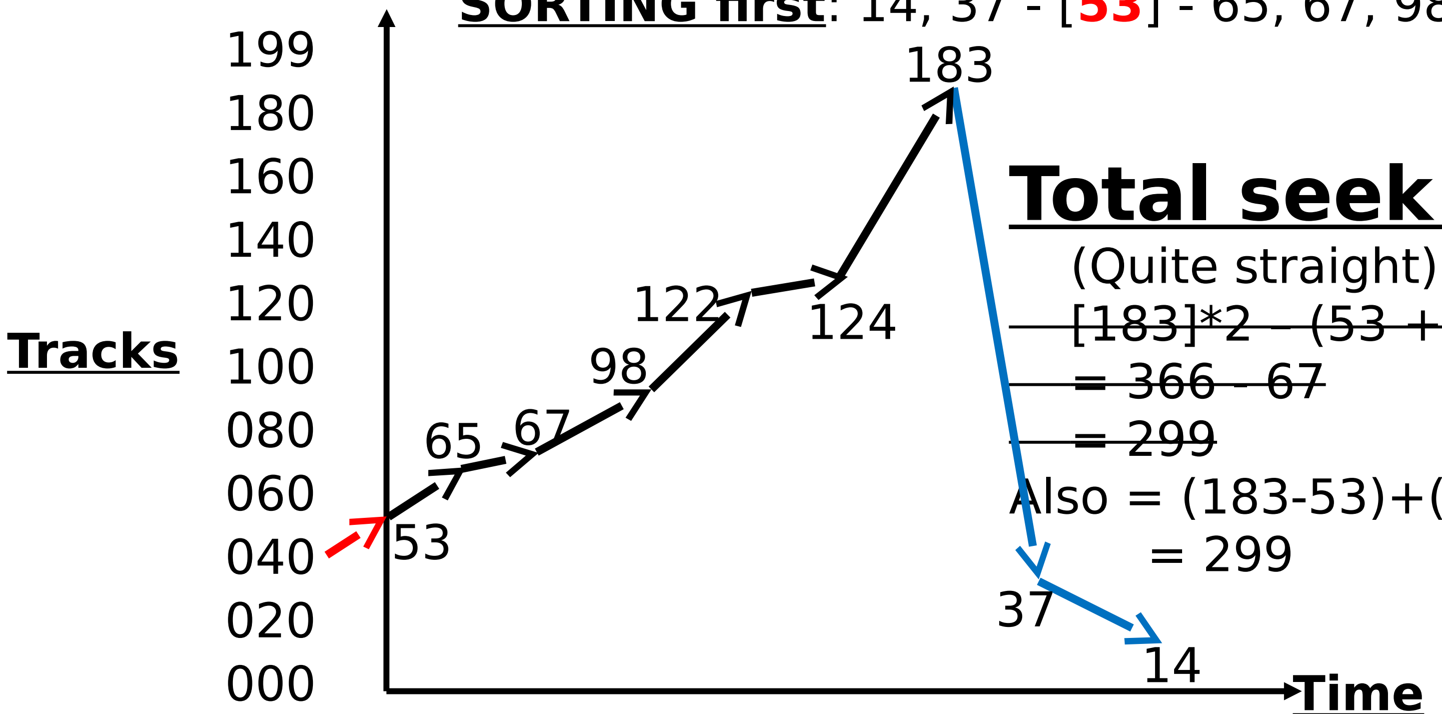
- With SCAN and C-SCAN, the disk head moves across the full width of the disk.
- This is very time consuming. In practice, SCAN and C-SCAN are not implemented this way.
- LOOK:
  - It is a variation of SCAN. The disk head goes as far as the last request and reverses its direction.
- C-LOOK:
  - It is similar to C-SCAN. The disk head also goes as far as the last request and reverses its direction.

PPTs from others\www.cs.mtu.edu_~shene_OS-slides\chap13-IO-Sys.pdf

- LOOK

98, 183, 37, 122, 14, 124, 65, 67

SORTING first: 14, 37 - [**53**] - 65, 67, 98, 122, 124, 183



Total seek distance

(Quite straight)

~~[183]*2 – (53 + 14)~~

~~= 366 - 67~~

~~= 299~~

Also = (183-53)+(183-14)

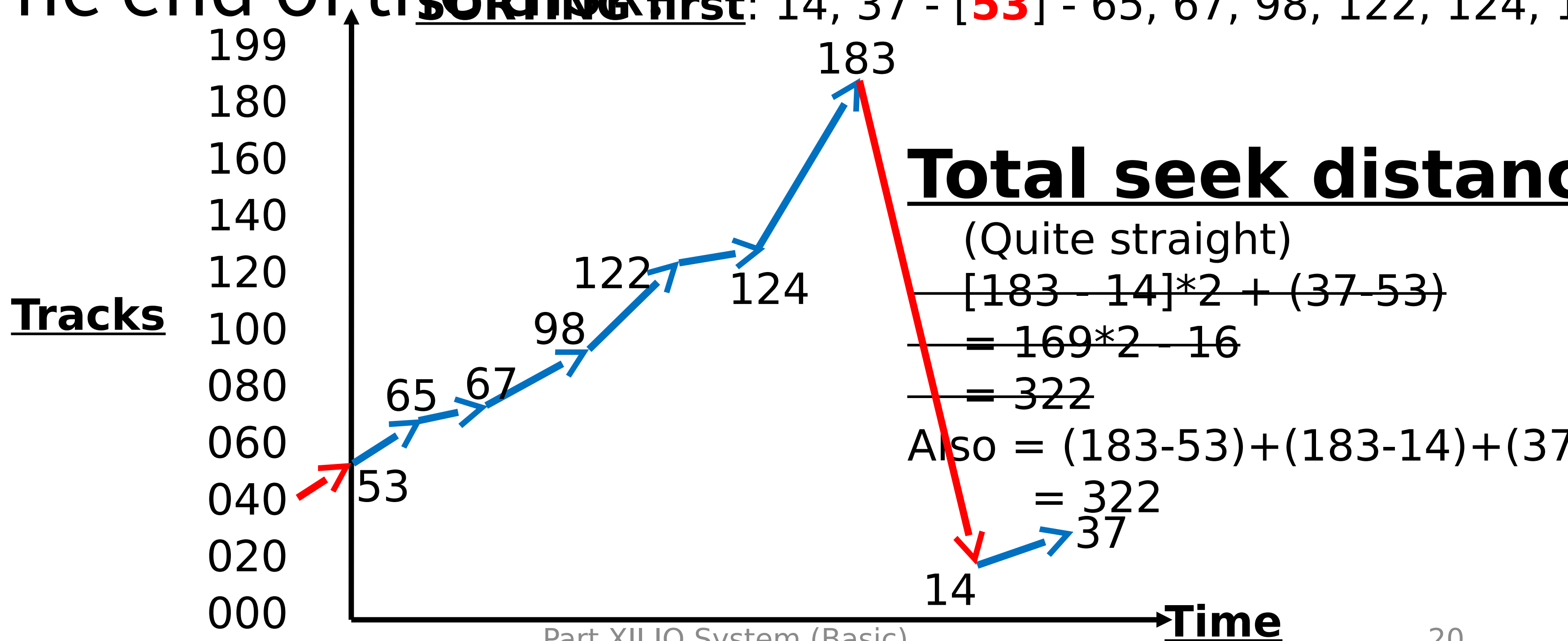
= 299

C-LOOK

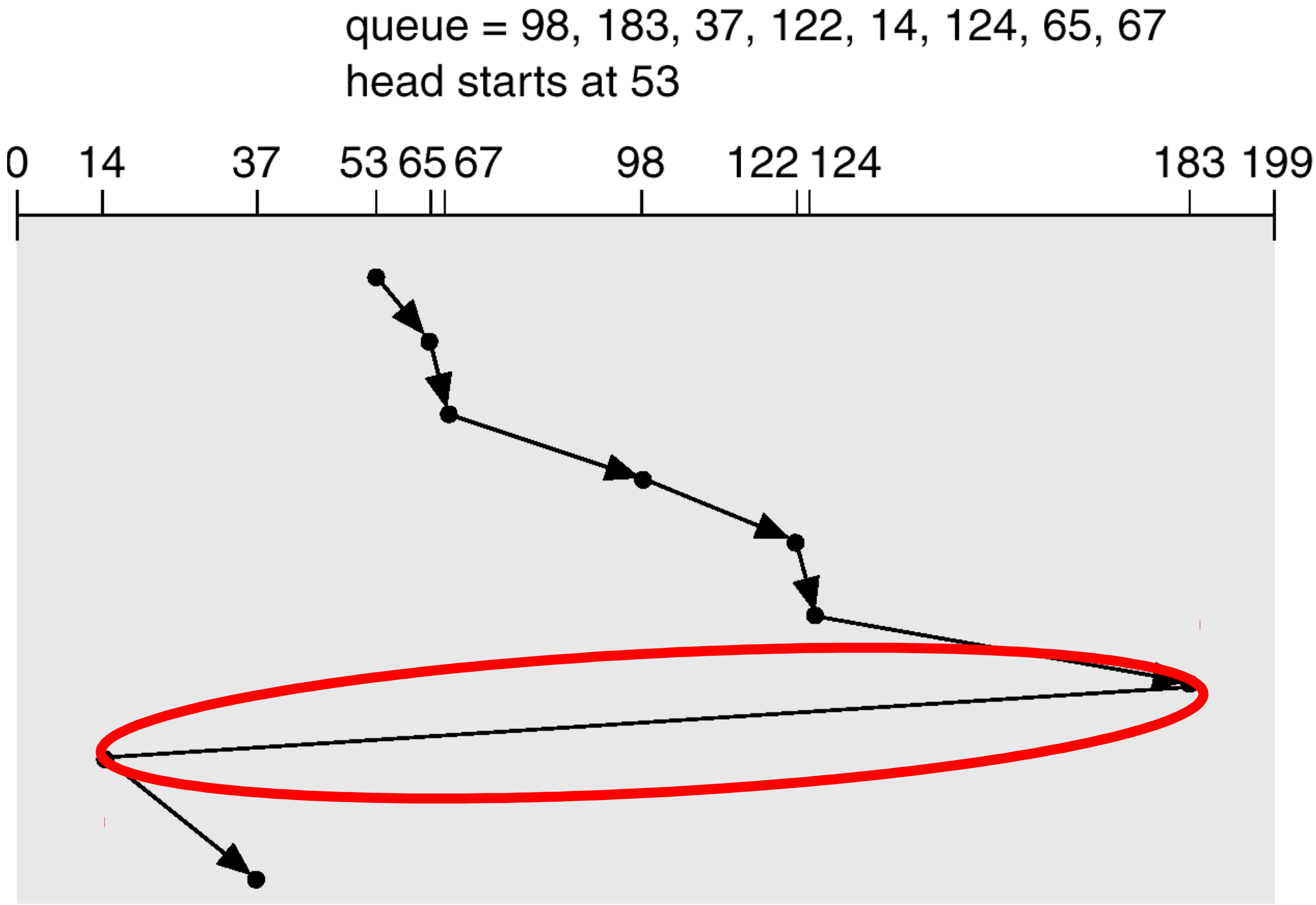
- Variation of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk.

98, 183, 37, 122, 14, 124, 65, 67

SORTING first: 14, 37 - [**53**] - 65, 67, 98, 122, 124, 1



C-LOOK


queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
0 14 37 53 65 67 98 122 124 183 199

Selecting a Disk-Scheduling Algorithm

- **SSTF** is common and has a natural appeal
- **SCAN** and **C-SCAN** perform better for systems th at place a heavy load on the disk
- Performance relies on the number and types of requests
  - Requests for disk service can be influenced by the fil e-allocation method (Contiguous? Linked? Indexed?)

Selecting a Disk-Scheduling Algorithm

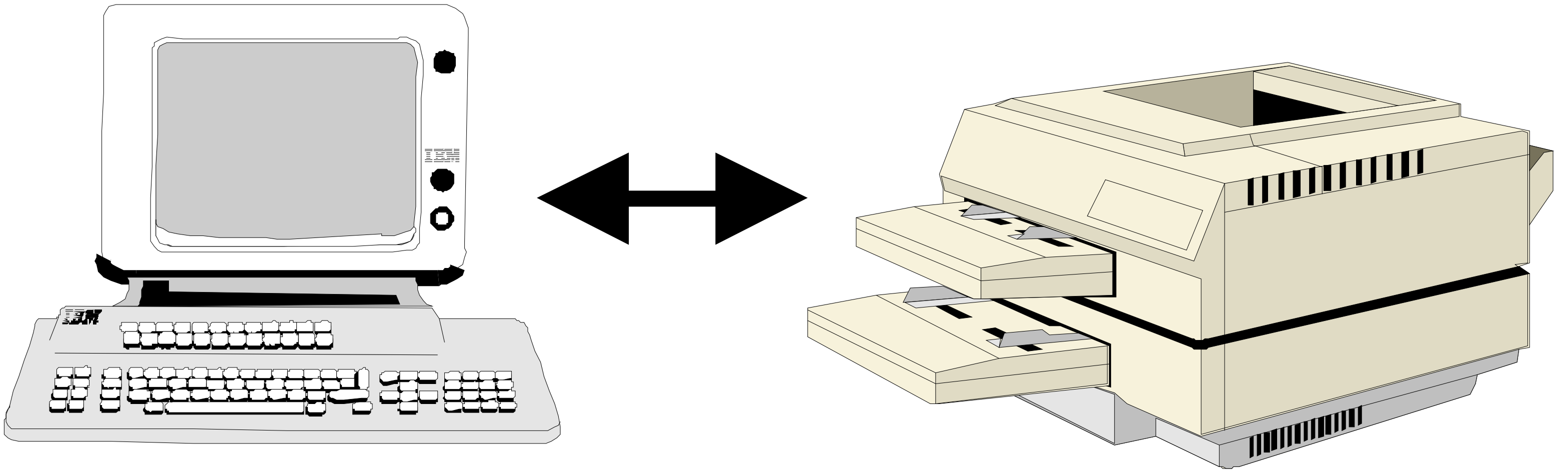
- The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary
  - Either SSTF or LOOK is a reasonable choice for the default algorithm
  - Scheduled by OS or by disk controller?

Advanced IO

- Scheduling algorithms for disk I/O requests
- SPOOLING

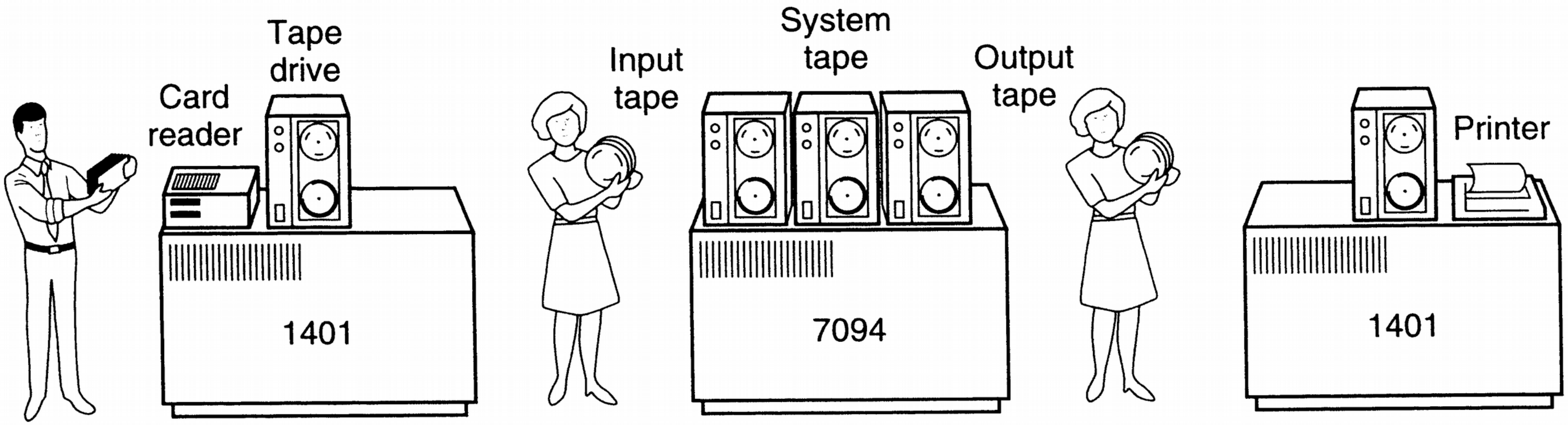
  Simultaneous Peripheral Operations On-line: 同时的外围设备联机操作（假脱机技术）
- RAID
  - Redundant backup for the safe storage
- USB
  - Universal interface for diverse devices
- NAS, SAN, …
  - Scattered storage

Basic Printer Operation



- A Printer is a peripheral device, usually attached to a host computer
- The host computer transfer print files t o the printer over the communication channel

Traditionally



1. There is an input machine to read cards into tapes
2. Then a person carries that full tape to the processor
3. After the processor output the result into an output tap e, another person carries the output tape to an output machine
4. The output machine prints the result out

Motivation

- Background – Multiprogramming
  - The value of multiprogramming is that more programs can be run in the same amount of time.
  - If the **turnover rate** [**周转率**] of those programs can be incr eased, even greater efficiencies can be realized.
- For example
  - Imagine a given system executes five concurrent programs, and that each program occupies memory for 10 seconds.
  - As soon as a program finishes executing, another one repla ces it in memory. Thus, the computer can run thirty progra ms a minute.
  - If we could reduce each program’s run time to five second s, we could run sixty programs in that same minute

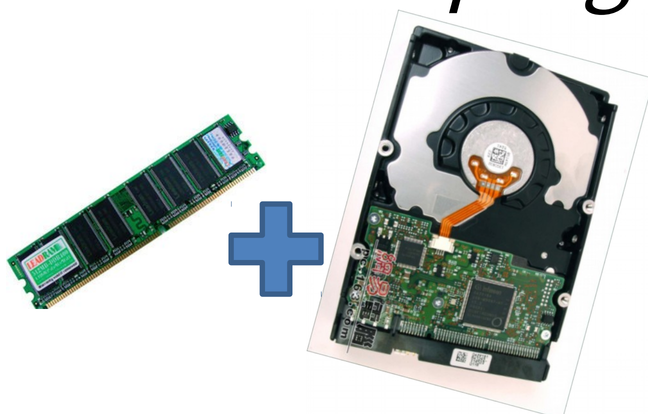
- On output, data are **spooled** to disk and later dumped to the printer.
- Because the application program deals only with high-speed I/O, it finishes processing much more quickly, thus freeing space for another program

- **Spooling** is a way of dealing with **dedicated I/O devices in a multiprogramming system**.
- A **spooling directory** is used for storing the spooling jobs

Spooler now

http://en.wikipedia.org/wiki/Spooling

- In computer science, **spool** refers to the process of *placing data in a temporary working area for another program to process*.



Spooling

SPOOLing 技术（Simultaneous Peripheral Operation On Line）

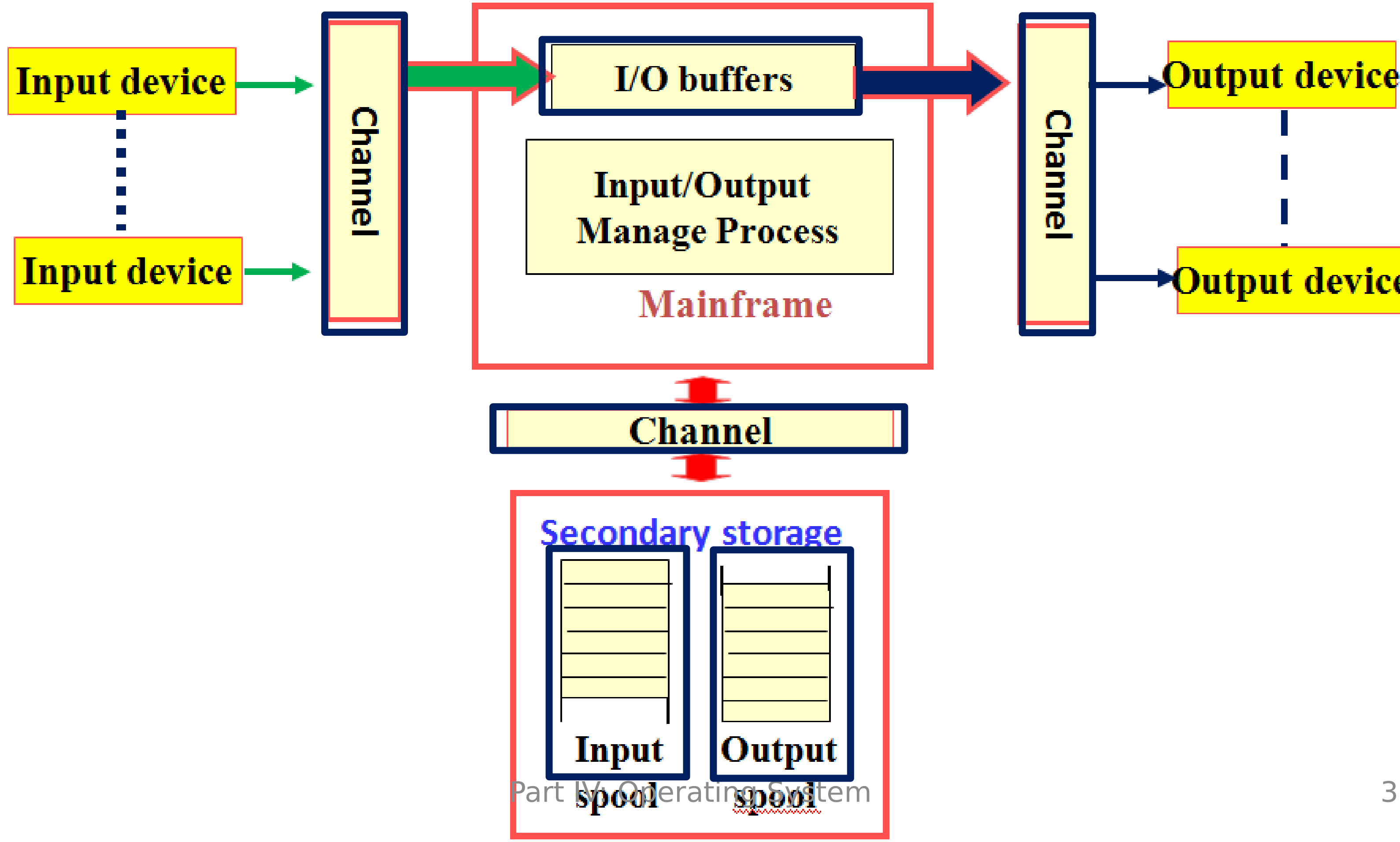
- SPOOLING 技术 [**联机同步外设操作**] is a way (buffer-based tech) of dealing with dedicated I/O devices in a multiprogramming system

, I/O buffering

Channels

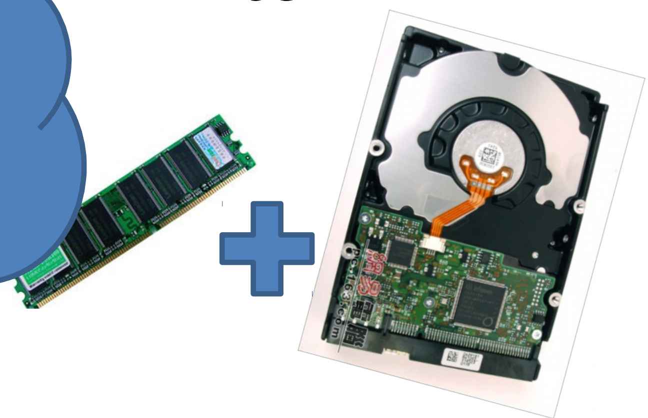
, Input/output spooling directories [输入 / 输出**井**]

—A spooling daemon process



- Two Separate Functions
  - Program-to-Queue.
  - to-Printer

Program-to-Queue



Queue-to-Printer

A

B

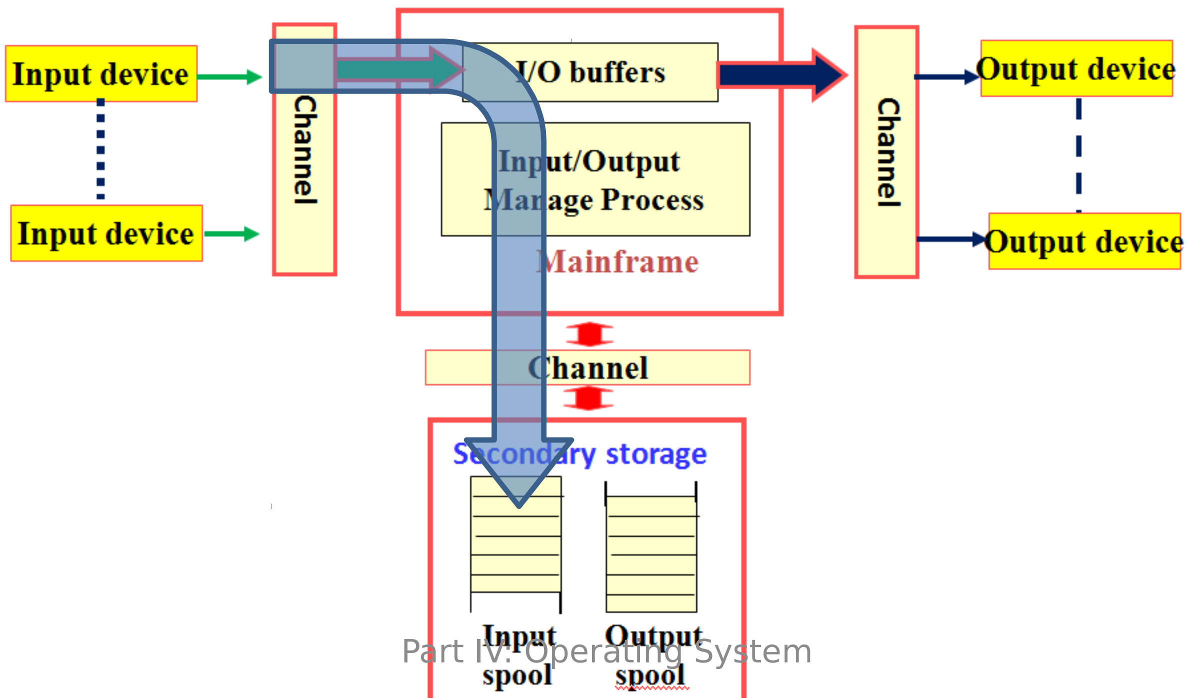
C

Spooling

SPOOLing 技术

(Simultaneous Peripheral Operation On Line)

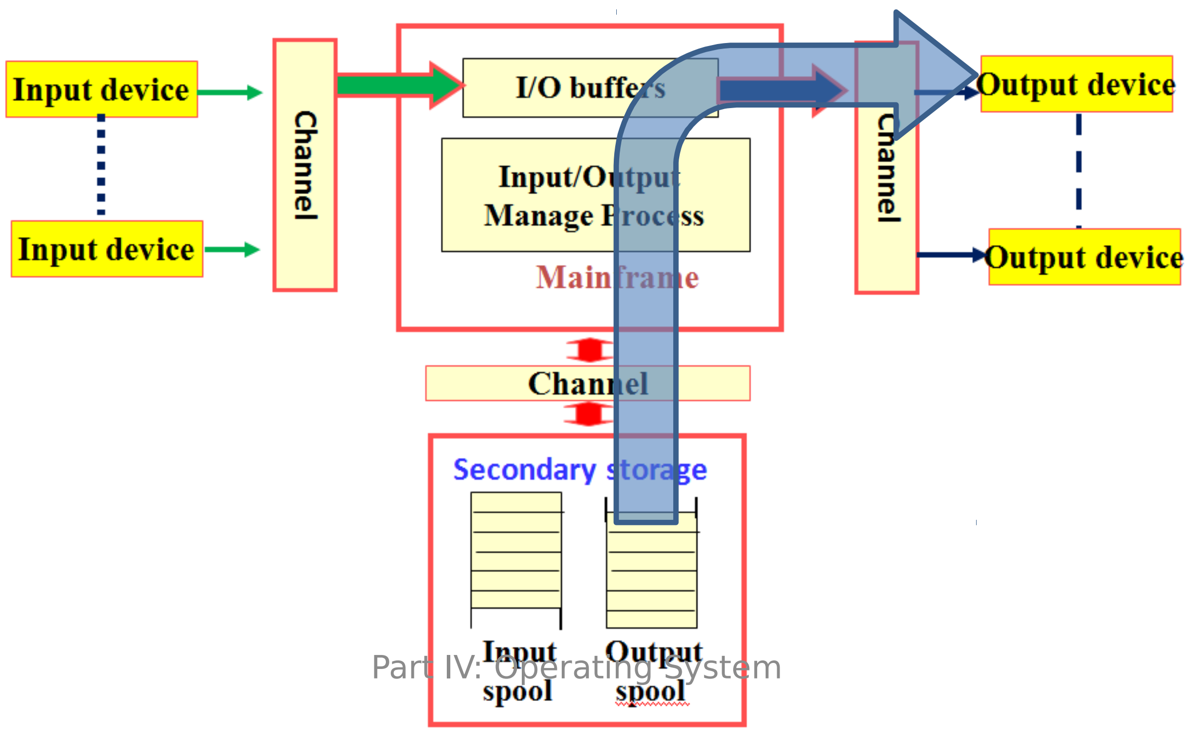
- When there are processes need input tasks (reading punched cards), they tells the SPLOOLING daemon, then daemon
  - calls the input management process to put the input data into the **inpu t-spooling directory** [输入井]
  - when finished, the daemon notifies the corresponding input process
- No matter how many input tasks, all the processes feel like th at each of them owns the input device by itself.



SPOOLing 技术

(Simultaneous Peripheral Operation On Line)

- When there are processes need output tasks (reading punched cards), they just
  - put their output data into **output-spooling directory** [输出井]
  - then the daemon finishes the output task
- No matter how many output tasks, all the output processes fee l like that each of them owns the output device (printer) by itse lf.



SPOOLing can also be used to

- Send emails
  - When user sends a mail, the system places a copy in it s private storage (spool area).
  - The system then initiates transfer to remote machine as a background activity.
  - The background mail transfer process becomes client.
  - If it succeeds, the transfer process passes a copy of th e message to remote server.
  - If it fails, the transfer process records the time delivery was attempted and terminates.
  - The background transfer process sweeps through the spool area periodically (typically every 30 mins).

Cont’

- Whenever it finds a message or whenever user deposits new outgoing mail, it attempts delivery.
- If mail message cannot be delivered after an extended time, the mail software returns the mail message to sender

Adv
anc
ed I
O

- Scheduling algorithms for disk I/O req
uests
- SPOOLING

Simultaneous Peripheral Operations On-line:
同时的外围设备联机操作(假脱机技术)

- RAID
  – Redundant backup for the safe storage
- USB
  – Universal interface for diverse devices
- NAS, SAN, ...
  – Scattered storage

RAID

- RAID is originally defined as **Redundant Array of Inexpensive Disk**(**廉价磁盘冗余阵列**) ， but the industry redefined I as Independent ， not Inexpensive.
  - On the contrary, we have **SLED, that is, Single Large Expensive Disk** (**单个大而贵的磁盘**) 。
- The basic idea of RAID is
  - To store the **backup** (copy or parity bits) of your data among several disks
  - By using RAID controller, we can use that disk set as one disk
    - Namely, RAID is just like ... ce and better reliability

RAID controller

We've learn the so-called controller

PPTs from others\蒋塔 . 操作系统\CH05.p

Motivation of RAID – for better Reliability by redundancy

- You all have this experience
  - After long time consideration of your paper, your computer crashes, and you have not done any backup!
  - So you MUST have learned that you should make some backup of your important documents every time!
  - I used to backup my materials in 3 places: **computer**, a **portable hard disk**, and **network storage space** (like “微盘” of BaiDU)

RAID Levels

Mirror the data, or store parity bits

Category	Level	Description	(Read/Write)	(Read/Write)	Typical Application
Striping	0	Nonredundant	Large strips: Excellent	Small strips: Excellent	Applications requiring high performance for noncritical data
Mirroring	1	Mirrored	Good/Fair	Fair/Fair	System drives; critical files
Parallel access	2	Redundant via Hamming code	Poor	Excellent	
	3	Bit-interleaved parity	Poor	Excellent	Large I/O request size applications, such as imaging, CAD
Independent access	4	Block-interleaved parity	Excellent/Fair	Fair/Poor	
	5	Block-interleaved distributed parity	Excellent/Fair	Fair/Poor	High request rate, read-intensive, data lookup
	6	Block-interleaved dual distributed parity	Excellent/Poor	Fair/Poor	Applications requiring extremely high availablity

From Ariel J. Frank\OS381\raid.p

Hamming Code [哈明码]

- Designed to correct single bit errors
- Family of (n, k) block error-correcting codes with parameters:
  - Block length: $n = 2^m - 1$
  - Number of data bits: $k = 2^m - m - 1$
  - Number of check bits: $n - k = m$
  - Minimum distance: $d_{min} = 3$
- Single-error-correcting (SEC) code
  - SEC double-error-detecting (SEC-DED) code

(7,4) Hamming code

- As an example, we are sending the string "01 10", where m = 4, hence, we need 3 bits for pa rity check.
- The message to be sent is: $m_7m_6m_5$**P_4**m_3**P_2P_1** where m_7=0, m_6=1, m_5=1, and m_3=0.
- Compute the value of the parity bits by:
  - $P_1 = m_7 + m_5 + m_3 = 1$
  - $P_2 = m_7 + m_6 + m_3 = 1$
  - $P_4 = m_7 + m_6 + m_5 = 0$
- Hence, the message to be sent is "0110011".

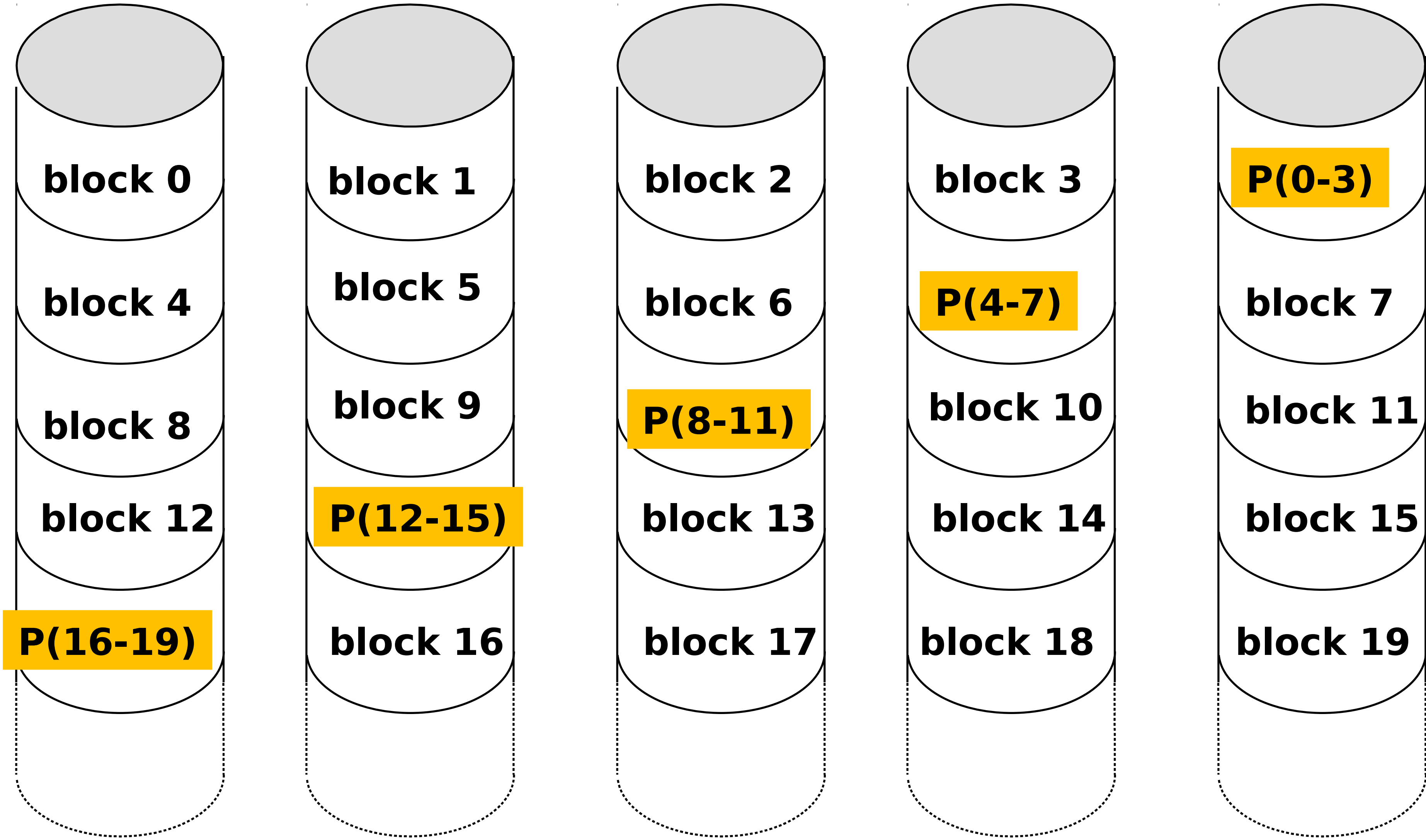
PPTs from others\www.comp.hkbu.edu.hk_~jng\2006Chapter11.ppt

- Say for example, if during the transmission, an error has occurred at position 6 from the right, the receiving message will now beco me "0010011".
- To detect and correct the error, compute th e followings:
- For P_1, compute $m_7 + m_5 + m_3 + P_1 = 0$
  - 0+1+0+1 = 0
- For P_2, compute $m_7 + m_6 + m_3 + P_2 = 1$
  - 0+0+0+1 = 1
- For P_4, compute $m_7 + m_6 + m_5 + P_4 = 1$
  - 0+0+1+0 = 1

- If ($P_4P_2P_1 = 0$) then there is no error
- else $P_4P_2P_1$ will indicate the position of error.
- With $P_4P_2P_1 = 110$, we know that position 6 is in error.
- To correct the error, we change the bit at the 6th position from the right from ‘0’ to ‘1’.
  - That is the string is changed from “0010011” to “0110011” and g… message “0110” fro… m_5m_3.

How about is the parity bit broken? – 0111011?

RAID 5 (block-level distributed parity) as instance



From Ariel J. Frank\OS381\raid.pp

Adv anc ed I O

- Scheduling algorithms for disk I/O req uests
- SPOOLING

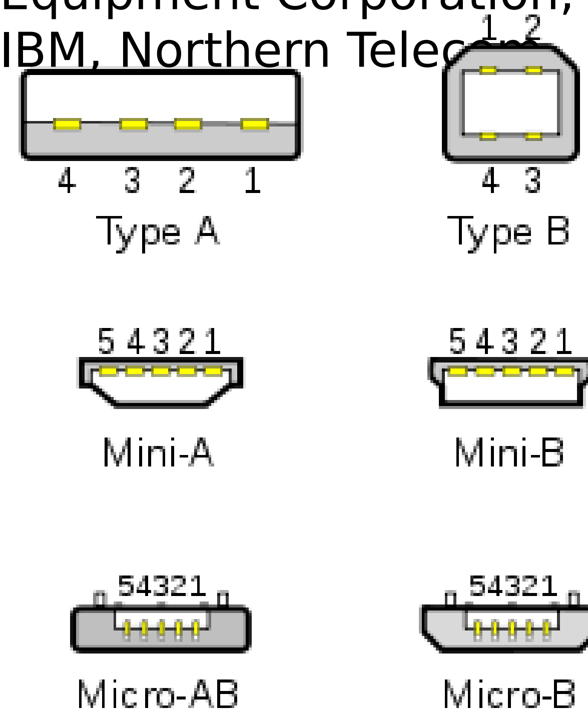
Simultaneous Peripheral Operations On-line: 同时的外围设备联机操作（假脱机技术）

- RAID
  - Redundant backup for the safe storage
- USB
  - Universal interface for diverse devices
- NAS, SAN, ...
  - Scattered storage

USB: Universal Serial Bus

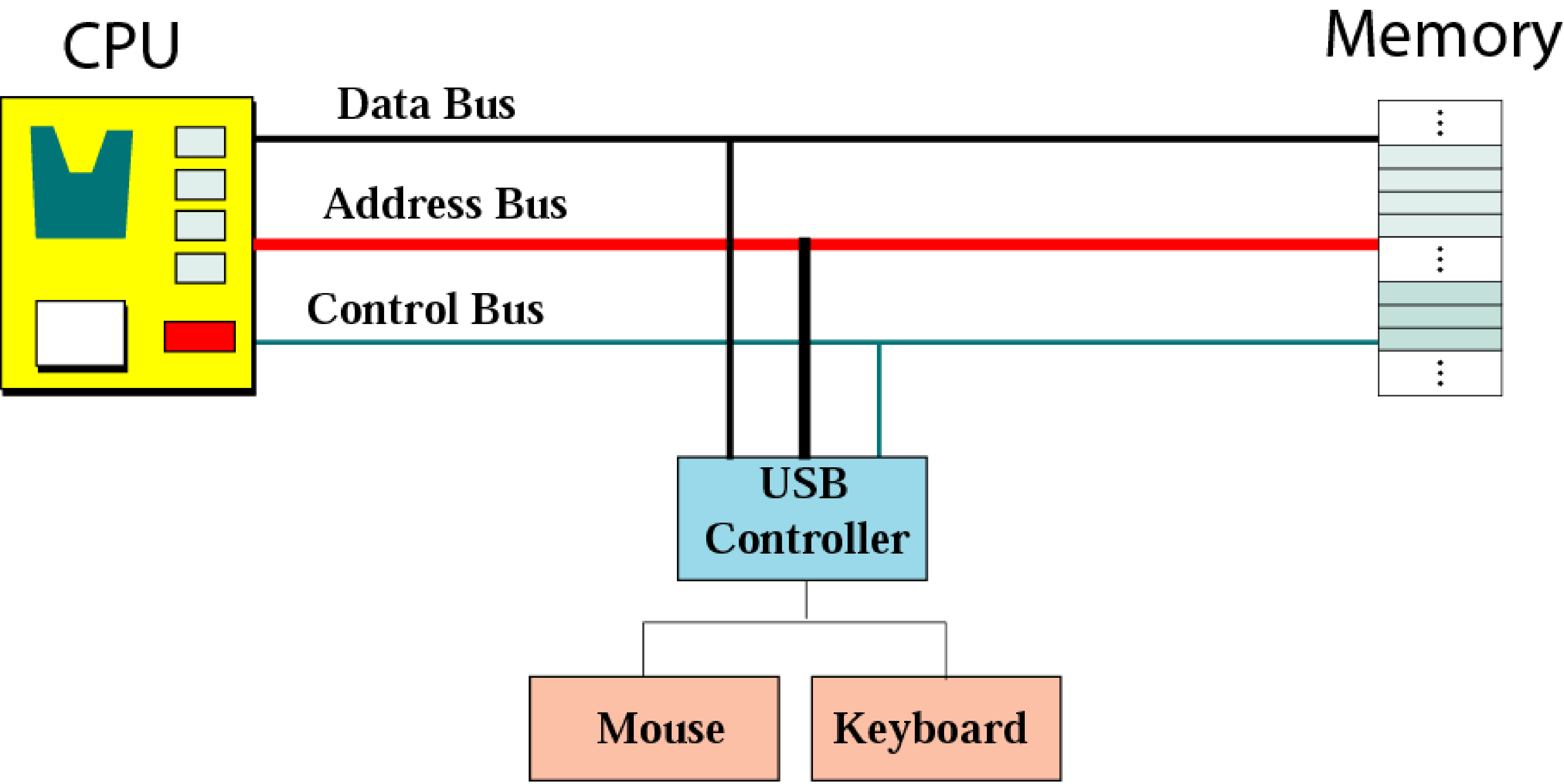
- USB (Universal Serial Bus) is a way of setting up communicati on between a computer and pe ripheral devices.
  - USB can connect computer perip herals such as **mice**, **keyboards**, **PDAs**, **gamepads** and **joysticks**, **s canners**, **digital cameras**, **printe rs**, **personal media players**, **flas h drives**, and **external hard driv es**.
  - For many of those devices, USB h as become the standard connecti on method.

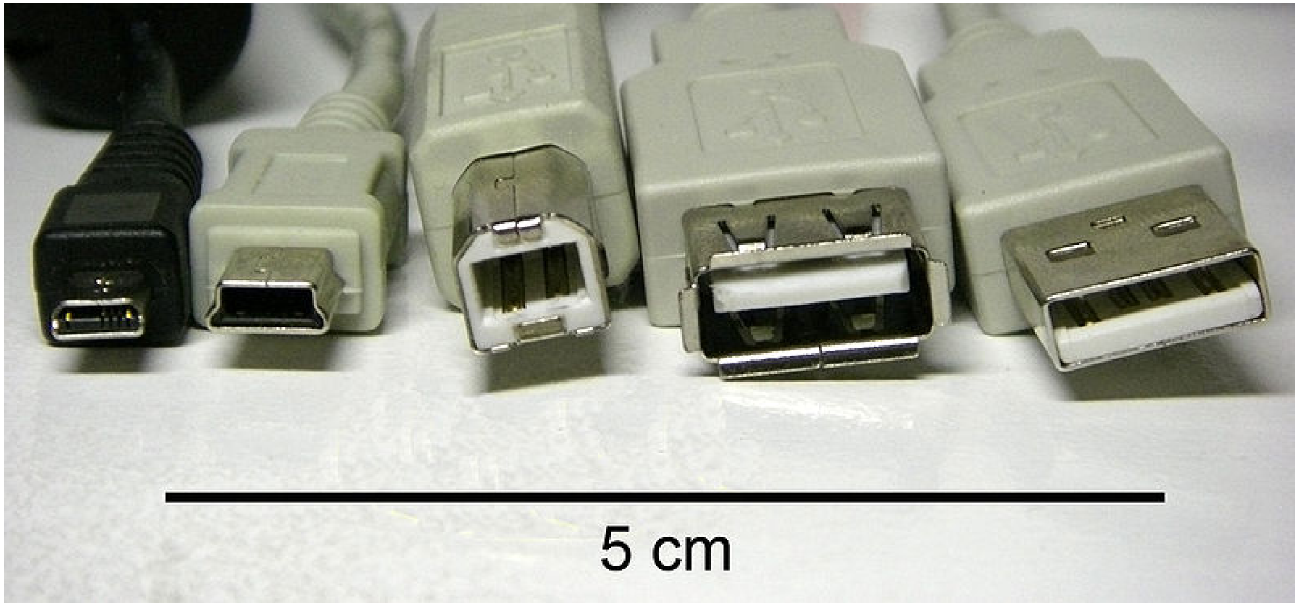
Year created: January 1996
Created by: Intel, Compaq, Microsoft, Digital Equipment Corporation, IBM, Northern Telec



http://en.wikipedia.org/wiki/Universal_Serial_

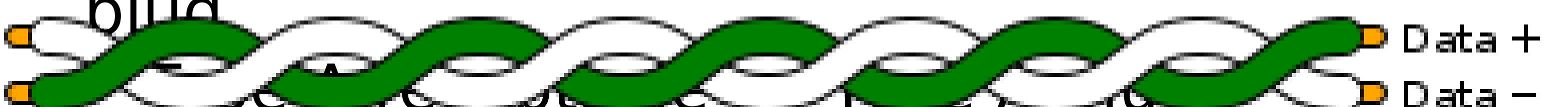
USB controller





Different types of USB connectors (from left to right)

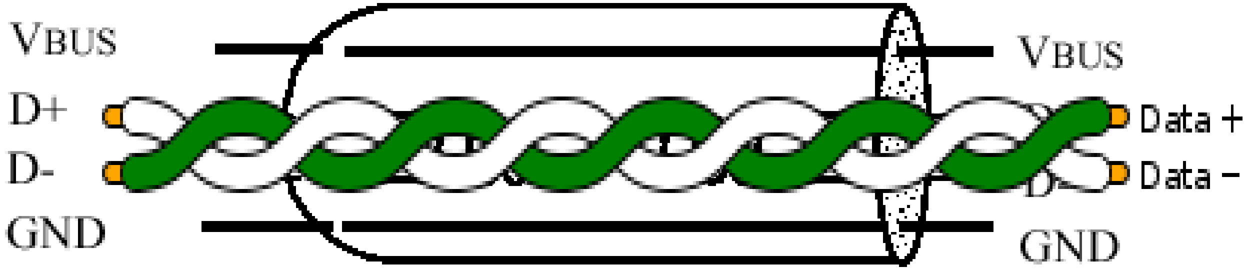
• 8-pin AGOX • Mini-B plug • Type B plug



The data cables for USB 1.x and USB 2.x use a twisted pair to reduce noise and **crosstalk**.

Physical Interface

pin	Name	Descryption
1	**Vcc**	**+5 Vdc**
2	**D-**	**Data-**
3	**D+**	**Data+**
4	**GND**	**Ground**



USB communication takes the form of packets

- Cutting the huge target into fixed size packets
  - Just as we do in **Netwo rking**
  - Each packet has unique ID
- Merge all the packets a gain to rebuild the hug e target
  - After all packets ... coll ected

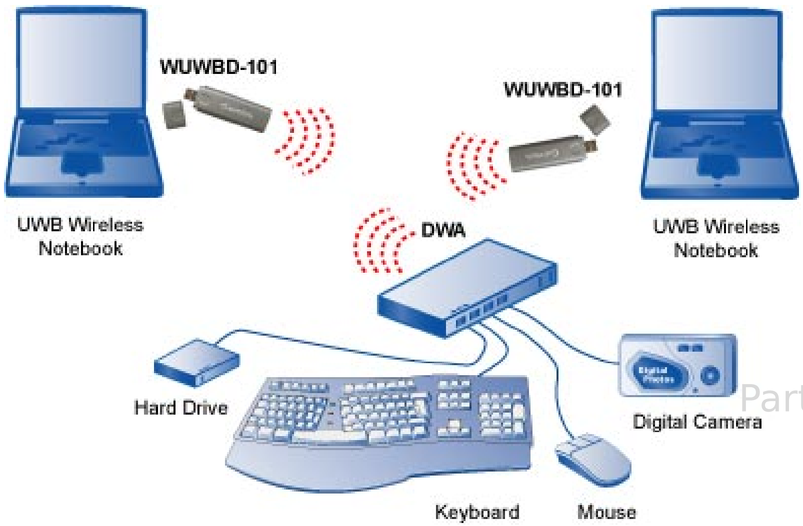
How to move this huge Buddha to ...er place ?

just as the data is transferred through network!

...s.2009\Part II Computer

USB Applications now

- Wireless video display
- Home and office
- MP3s
- General data transfer
- And More



Advanced IO

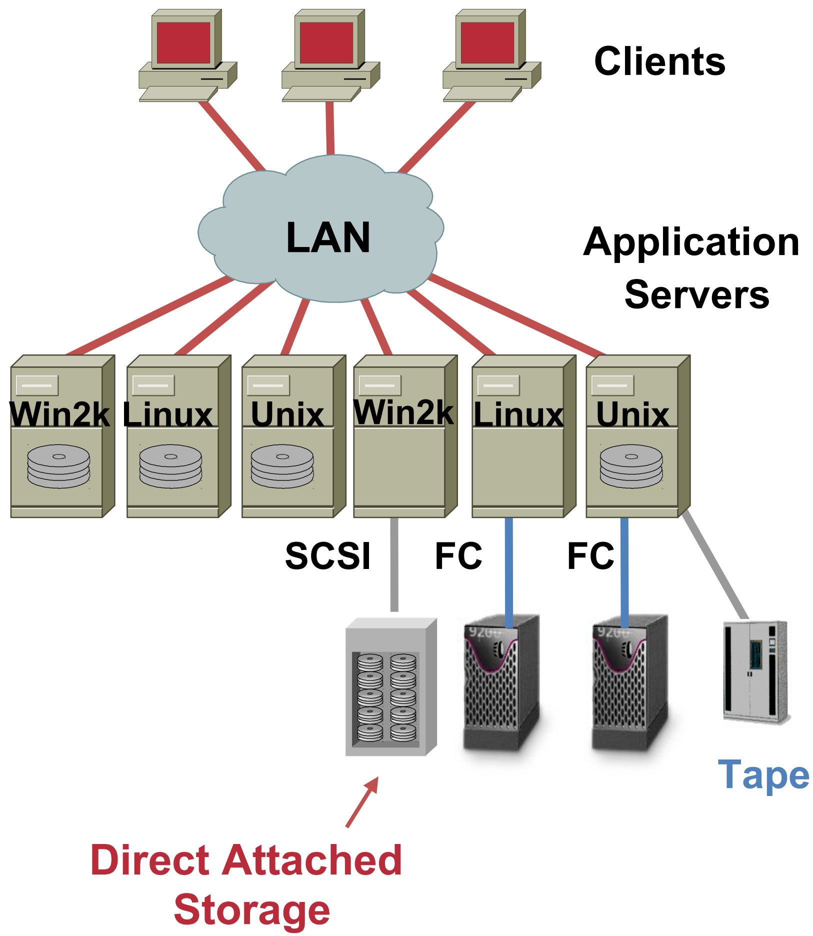
- Scheduling algorithms for disk I/O requests
- SPOOLING

Simultaneous Peripheral Operations On-line: 同时的外围设备联机操作（假脱机技术）

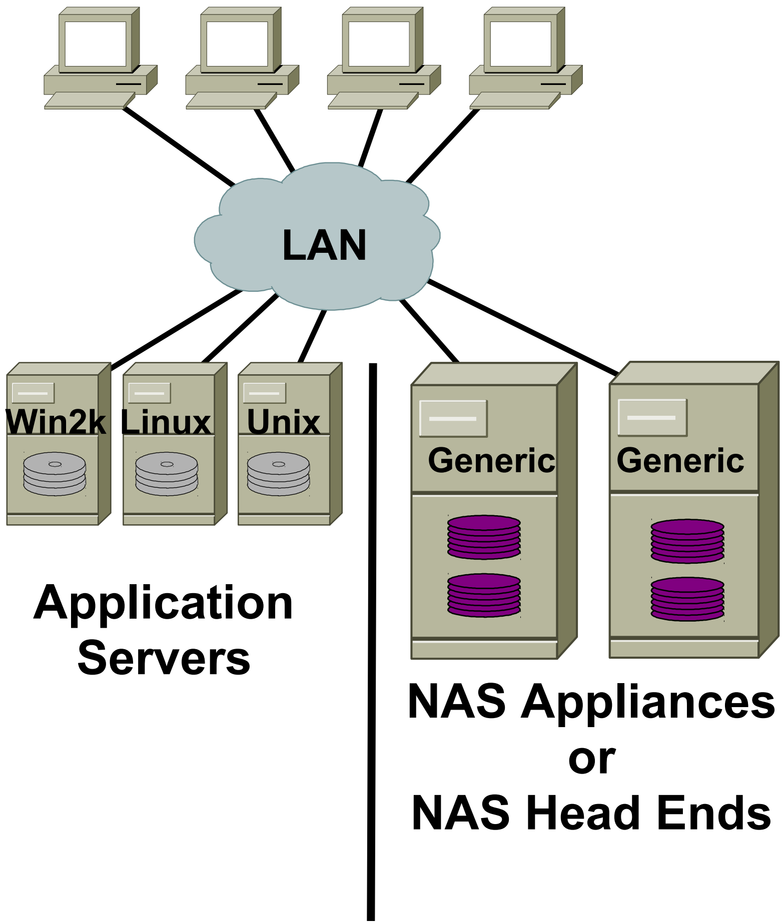
- RAID
  - Redundant backup for the safe storage
- USB
  - Universal interface for diverse devices
- NAS, SAN, ...
  - Scattered storage

DAS: Direct Attached Storage

- We are familiar with this.



NAS: Network Attached Storage [网络附加存储]



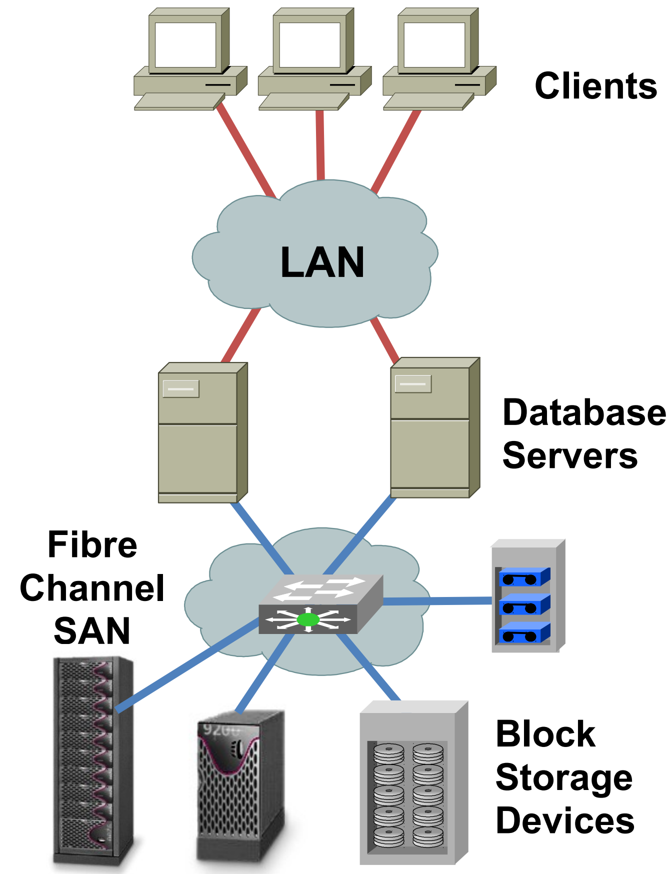
- Each Device Connected Directly to network, with own IP Address
- Various Devices (CD towers, **Tape Towers**, SCSI Towers, Specialty Servers)
  - If a Server crashes, the data on a NAS device **may be** still accessible, depending on what device is used

- **StorageTek Automated Cartridge System (ACS), with 6,000 tape cartridges storing a total of 300 terabytes**
  - **Location:** Laurel, Maryland **Date:** June 2004
  - **Camera:** Canon EOS 10D **ID Number:** #88015409

http://www.mccullagh.org/image/10d-15/storagetek-automated-cartridge-system.html

SAN: Storage Area Network [存储(区)域网(络)]

- Storage is accessed at bl ock level not at file level
- Very high performances
- Storage is shared
- Good management tools
- Interoperability issues



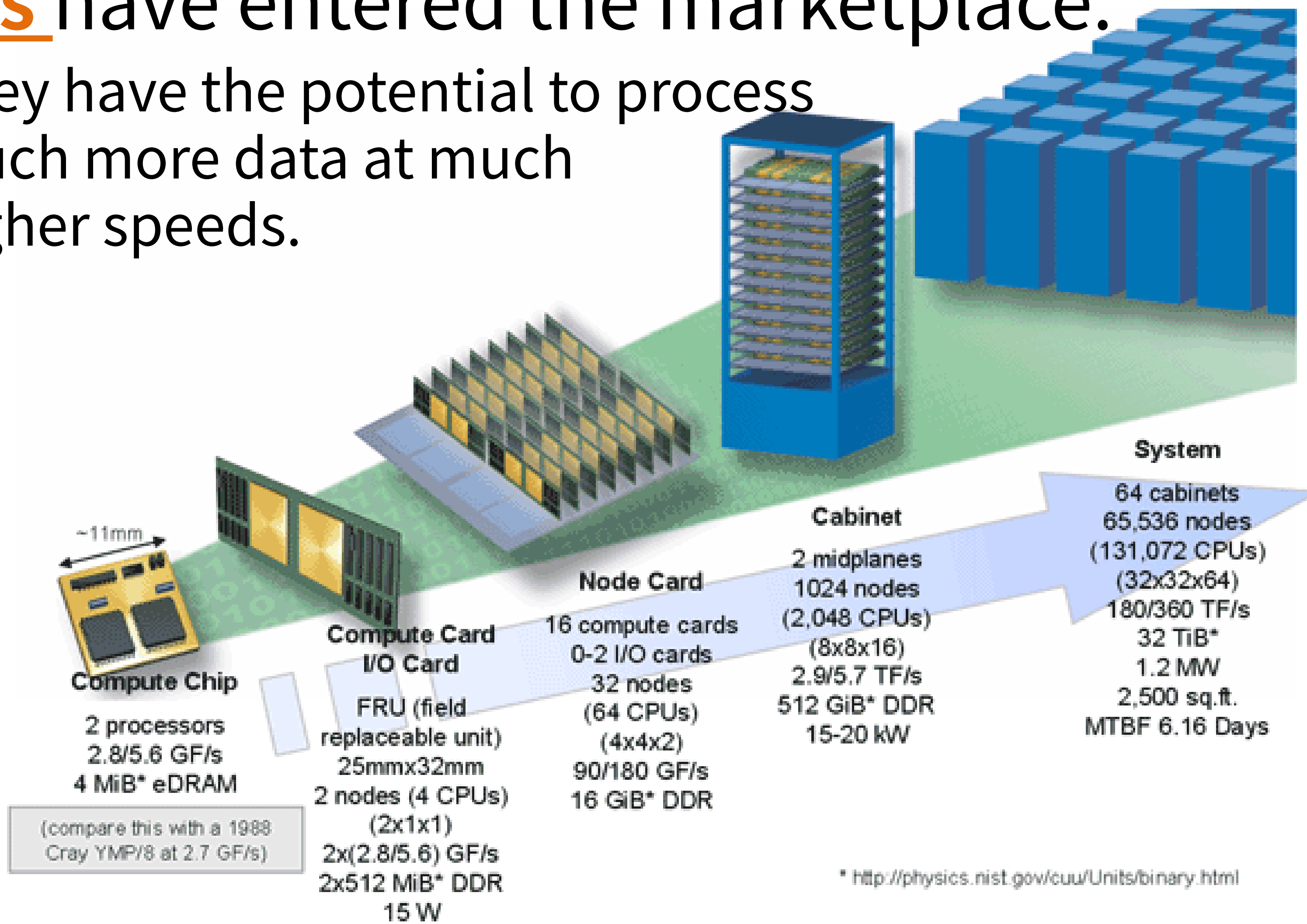
Storage Area Network (SAN)

- Google datacenter

One is not
adequate of
course

Non-von Neumann Architectures are also popular

- Since 1990, alternative **parallel-processing systems** have entered the marketplace.
  - They have the potential to process much more data at much higher speeds.



Blue Gene/L

Distributed System!

- Many nodes