I/O Device Management

There are numerous devices that are connected to a PC, for example:

1) Storage Device (Disks)
2) Network devices (Ethernet)
3) Graphics
4) USB ports

In this class, we’ll study only about bullets 1 and 2, and this lecture focuses on 1 – Storage.

Memory map for IO devices
User program to Hardware

User space

Kernel space

Software

Hardware

read()

n = read()

sleep

wake()

Interrupt

Do something

Device Controller

CPU

(North Bridge)

Memory Controller

Memory

PCI Bus

Device Controller

SATA
IO request flow

1) What device to call?
2) Does it have a mapping?
3) Dispatches request to the device

User Space

Kernel Space

Hardware

Device Driver

DMA Descriptor

Command Register

Interrupt Handler

Return to user
Command request

It helps us use DMA thereby reducing PCI Bus traffic.

Device Driver

Easy to develop

Complex due to
1) Interrupts
2) Page faults
3) Dispatching corresponding response from the queue
Block Devices

- Any device that acts like a device is a block device
- It is something that you put a file system on top of

Disadvantages of Block devices

1) We can read the Nth byte in a sector directly
2) We have to read the whole block to retrieve a single byte
3) Similarly for writing
Disk Arrays and Fiber channel

RAID (Redundant Array of Inexpensive Independent disks) (refer paper by Garth Gibson, CMU)

**RAID 1 (mirroring)** -> for reliability

After an error, we repair the erroneous disk and copy the data from the other.
RAID 0 (Stripping) -> for speed

1) This can be extended to more than two disks
2) Increased speed for programs that have large continuous data
3) But it amplifies the chances for failure

RAID 0 (Parity)

1 0 1 1 0 1 1 -> Parity = 1

Using the parity bit, we can retrieve the lost bit

If this drive fails, the parity drive can replace it

The protection becomes thinner as we add more disks
For writing in RAID 0

1) Every write has to compare and update the parity

2) The parity drive is the bottleneck

**RAID 5 (Rotating Parity)**

```
+---+---+---+---+
|   |   |   | P |
+---+---+---+---+
| 1 | 2 | 3 | P |
| 4 | 5 | P | 6 |
| 7 | P | 8 | 9 |
| P | 10| 11| 12|
```

The performance increases here since the parity bottleneck is spread across all drives.

There are also mixed drives like RAID 1 0, etc where we combine two of the above RAIDs

**Logical Volume Management (LVM)**

```
Disk size N

writeAt(0)

writeAt(N/2)
```

1) Storage requirement Increases

2) But, we don't need to create new File System