COMP 3430

Operating Systems

June 24th, 2019

By the end of today's lecture, you should be able to:

Goal, not gaol… Pixabay License

- Explain how data integrity can be protected from external issues (chapter 42, 45)
- Describe how multiple disks can be abstracted as a single volume (chapter 38, in-class)
- List RAID levels (chapter 38)
- Choose an appropriate RAID level for a problem or system (chapter 38, in-class)
- List types of filesystems (in-class)

E-mail questions

Using the banking example we have a bad case where it's possible to deadlock with just regular locks. (same thing happening with obverse data) Now I'm wondering if semaphores would actually be able to solve this issue?

E-mail questions

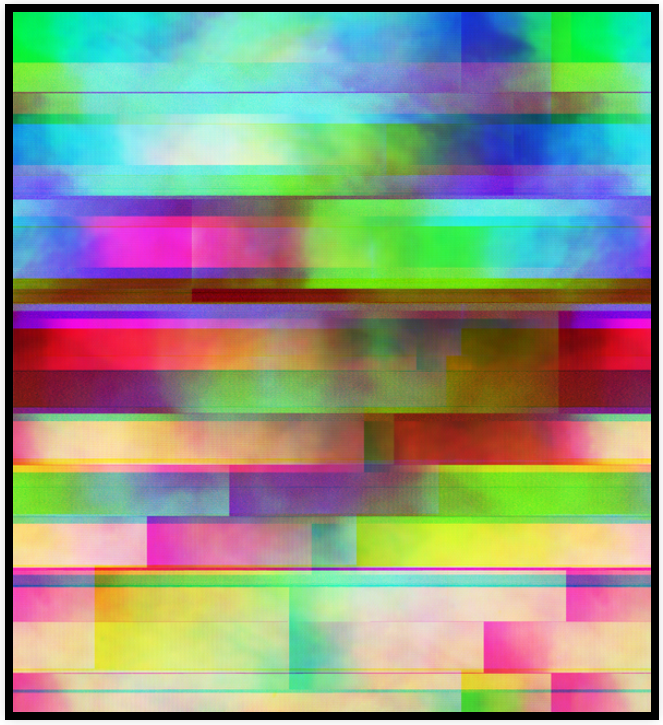
1. So first of all I just want to clarify that in part 1 we should be running the scheduling algorithms 10 times on the same file?
2. The average output (such as in the sample) should be the result from this 10 times, or each time should have its own output?
3. I just wanted to clarify that the shortest time remaining one should just be going through the job fully, no time slices, not round robin.
4. For the priority round robin then for example if there is only one process with priority 1 then it would that process for the timeslice, then when it gets added to the queue again since its the highest priority it just would go again?

E-mail questions

5. All the processes from the file should be loaded into the appropriate queue and then it should be started to go through them?
6. So for the IO, say my time slice is 10 and I do the random number thing and I get 5. Does that mean 5 of that timeslice I'll be doing IO and then 5 I'll be working on the process? Or does it only run for 5 and then stop. Also does the 5 that is doing IO count towards the process time or is just like an addition.

Digression: Melanie Willhide

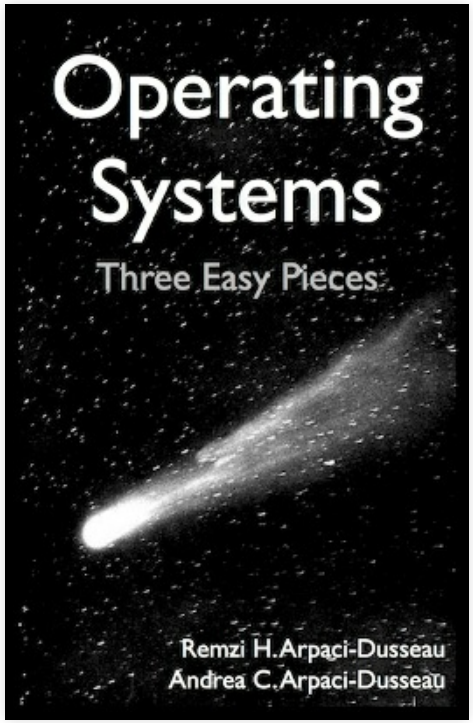
Related (tangentially) to the deletion time attribute on ext2 discussion we had last week.



OSTEP Q 'n A

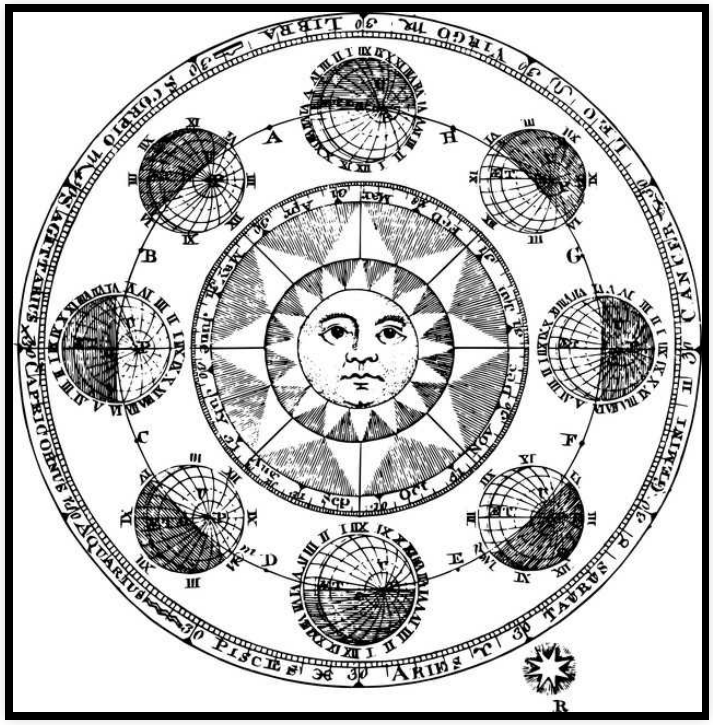
Chapters for this week:

- Chapter 38
- Chapter 42
- Chapter 45
- FAT32 whitepaper (for next class)

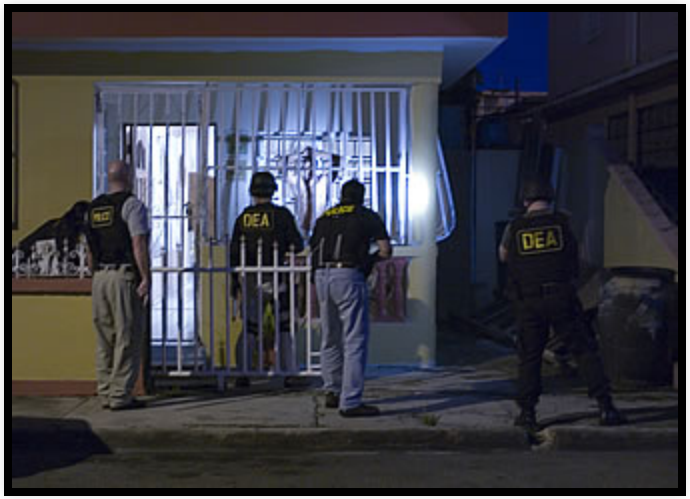


I/O Scheduling

- I/O scheduling and process scheduling are *remarkably* similar.
  - Time remaining ↔ # sectors/blocks to write
- … but there's *one* **fundamental** difference.
- **Think about it** : What is *fundamentally* different between I/O scheduling and process scheduling? (hint: why did we *abandon* algorithms like shortest remaining time first?)



RAID



Public Domain

Controllers

Let's, uh, look at a RAID controller.

Note: This is an *ancient* RAID controller.

Public Domain

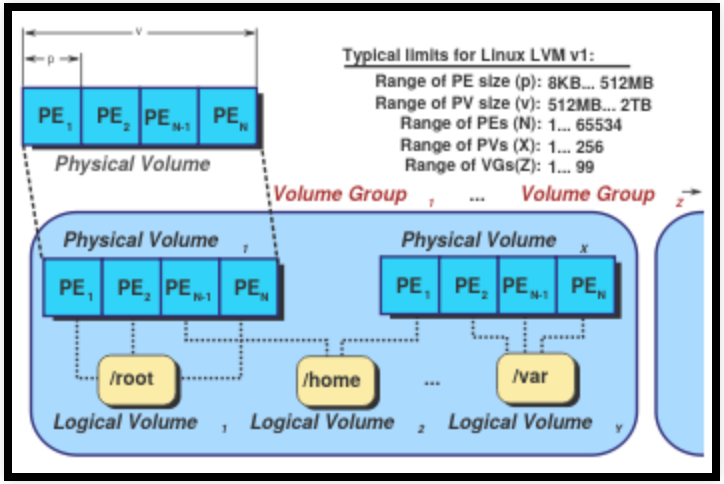
Software RAID

- Hardware RAID cards are…
  - … expensive
  - … hard to configure
  - … require your hardware to know about it
- Idea: So let's implement RAID, but in *software*.

Soft controllers.

LVM

- Let's take a look at the Linux Kernel's implementation of LVM
  - We'll try to come up with a basic idea of the *structure* and *responsibilities* the kernel has when mapping logical I/O to physical I/O.
- Let's start by looking at `dm.h` and `dm.c`



What kind of RAID do I want?

Let's look at some scenarios and decide what RAID level would be appropriate for the problem, situation, or person.

Pixabay License

Scenario 1: the data hoarder

Scenario 2: the gamer

Scenario 3: the scientist

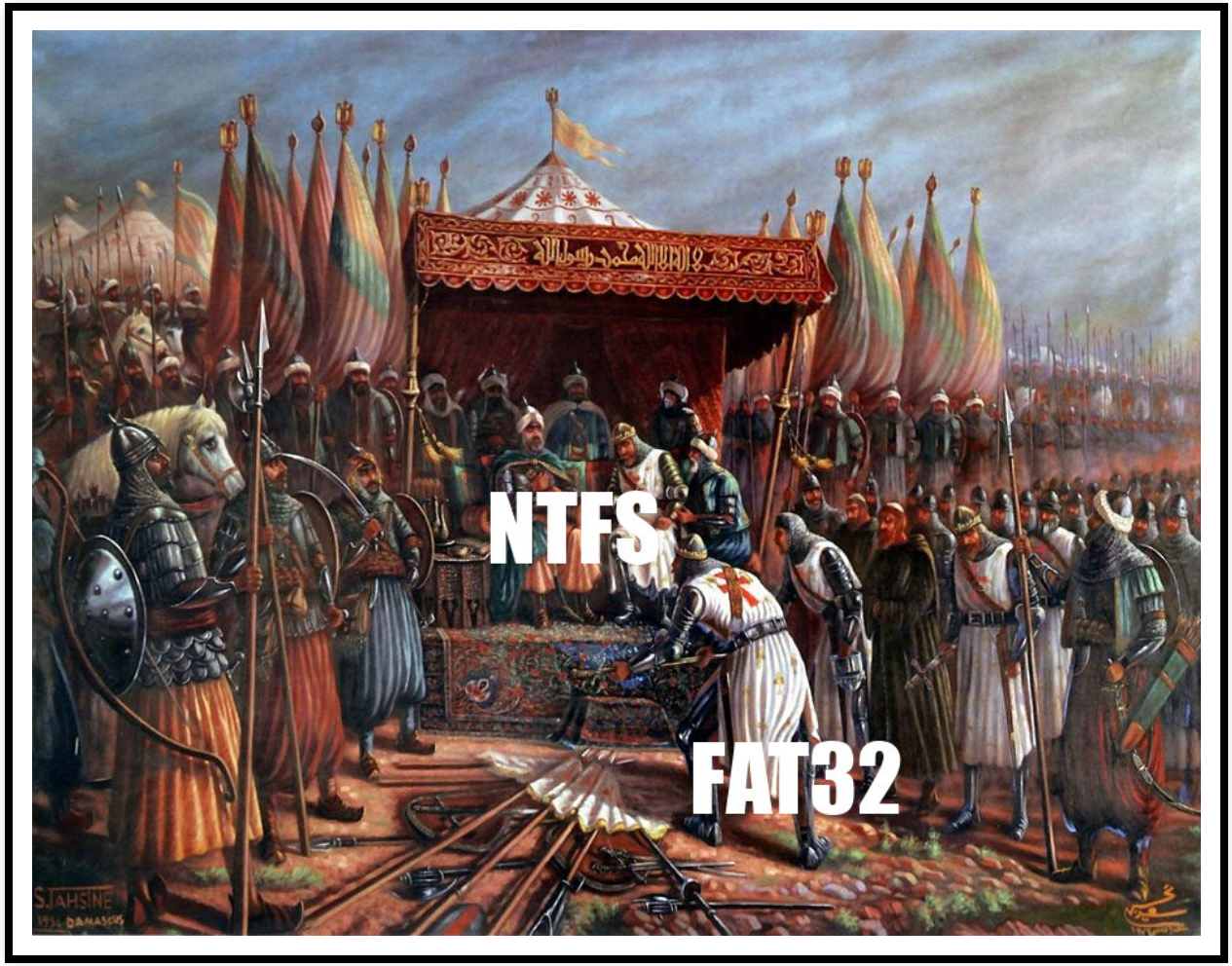
Filesystems

- Many different kinds of fileystems.

```
Disk /dev/nvme0n1: 477 GiB, 512110190592 bytes, 1000215216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 90400728-DEEA-4CB7-B7AD-8D1CDDC26D2E

Device           Start        End   Sectors   Size Type
/dev/nvme0n1p1    2048    1050623   1048576   512M EFI System
/dev/nvme0n1p2 1050624    2549759   1499136   732M Linux filesystem
/dev/nvme0n1p3 2549760 1000214527 997664768 475.7G Linux filesystem
```

`/dev/nvme0n1` on my system.



Public Domain

Filesystems

- Many different kinds of filesystems.
  - ext`{2,3,4}`
  - FAT`{12,16,32}`
  - NTFS
  - APFS
  - ZFS
  - Btrfs
- **Think about it** : *Why* might there be so many different kinds of filesystems?

You should be able to:

Goal, not gaol… Pixabay License

- Explain how data integrity can be protected from external issues (chapter 42, 45)
- Describe how multiple disks can be abstracted as a single volume (chapter 38, in-class)
- List RAID levels (chapter 38)
- Choose an appropriate RAID level for a problem or system (chapter 38, in-class)
- List types of filesystems (in-class)

frawgs

how much scarier would a frog be if it ran instead of hopped .. u just hear plat plat plat plat coming towards u and u look down and it's a frog going at full speed

spaceepigeon