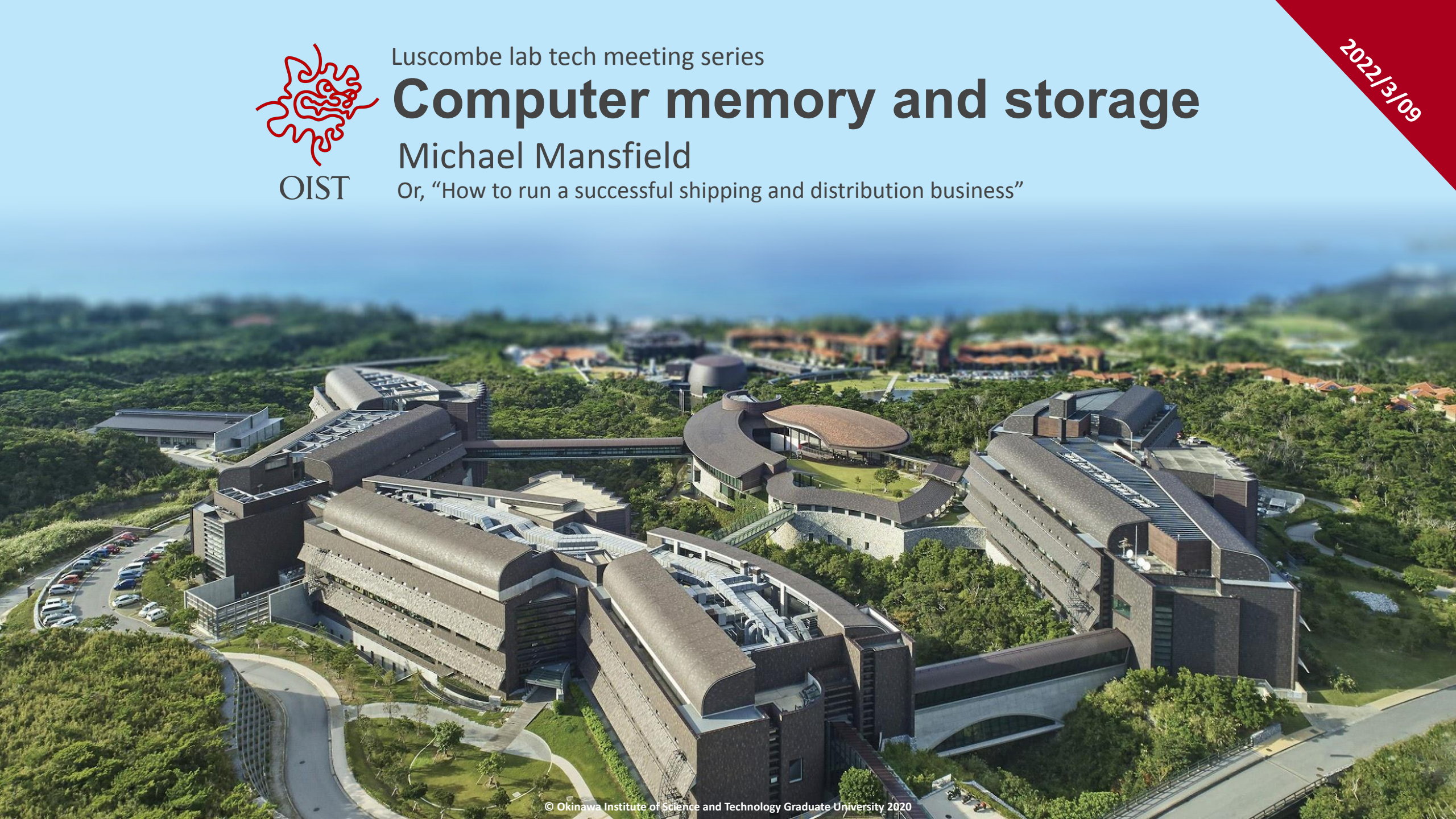
Luscombe lab tech meeting series

Computer memory and storage

Michael Mansfield

Or, “How to run a successful shipping and distribution business”

2022/3/09



Preamble to the world's most contrived analogy

You write a really, really good grant application only to have it rejected. Time to give a **career in industry** a shot!

You find a bargain, and take over a shipping and distribution business from the mysterious **Mr. Bean**.

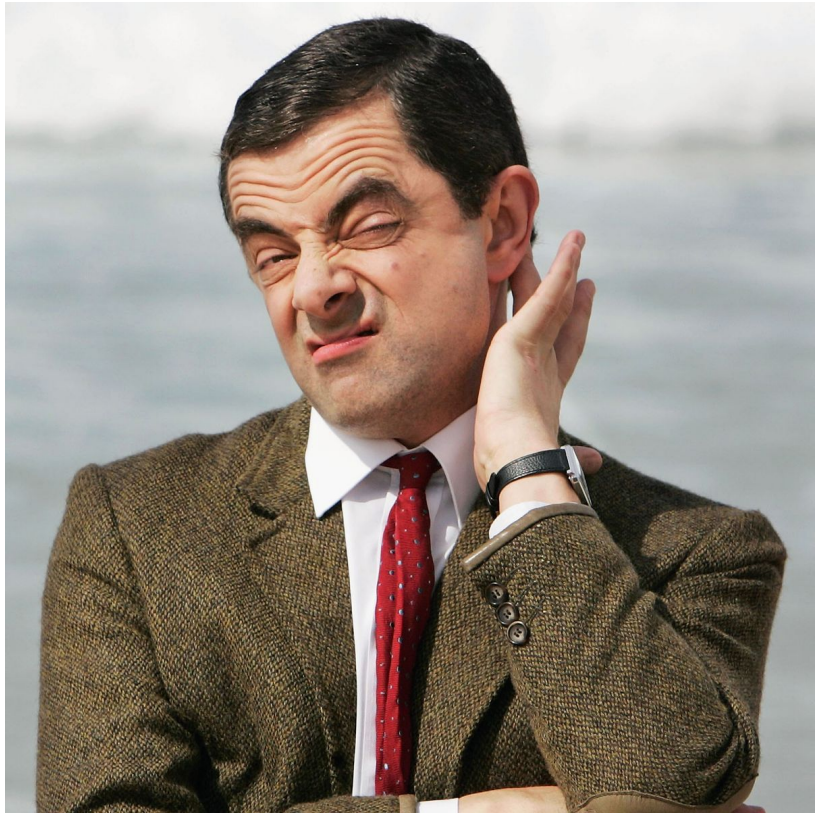
You are shocked - due to **egregious mismanagement**, the business is in danger of going under!

You order an audit of your business practices. The auditor give you a **long list of practices** that have contributed to your current woes.

Your new business

Company problems: Overview

1. The memos
2. The forklifts
3. The hallways
4. The space



Company problem #1: The memos

1. Every instruction to every employee must be explicitly written on paper memos and stored forever.

Company problem #1: The memos
Solution: Don't store everything forever!

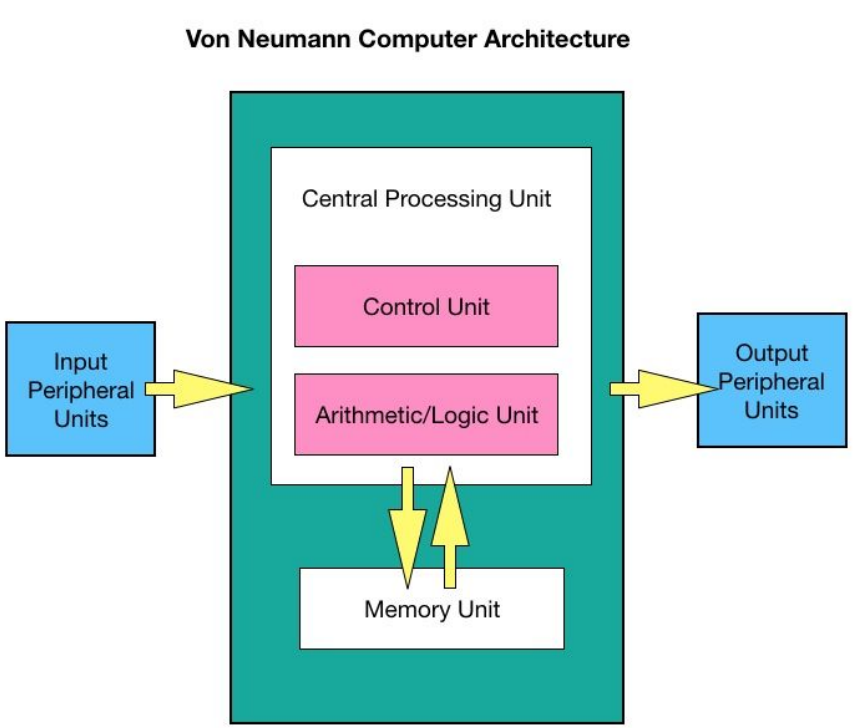
- What distinguishes memory and storage?
  - *Volatility, persistence*. Memory usually requires power to persist, storage does not
  - *Speed, capacity*. Memory is usually smaller and faster than storage

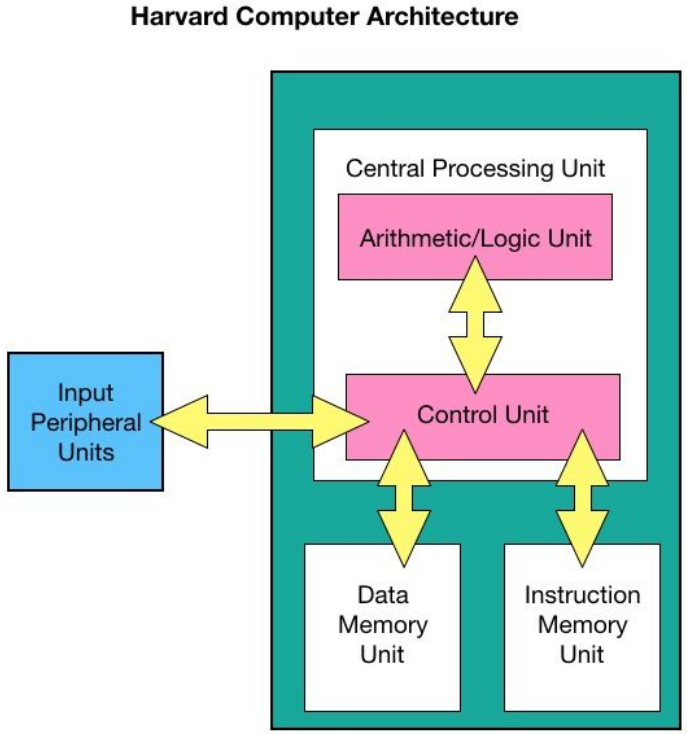
Small (16 GB)
Fast (~20 GB/s)
Volatile

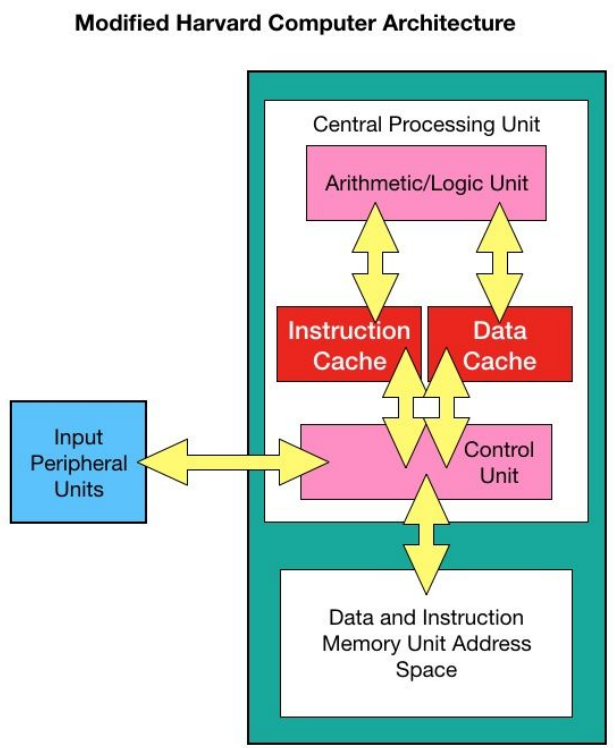
Big (>1,000 GB)
Slow (~0.2 GB/s)
Persistent

Sources: https://en.wikipedia.org/wiki/DDR4_SDRAM, https://en.wikipedia.org/wiki/Hard_disk_drive

Company problem #1: The memos
Aside: what constitutes storage?

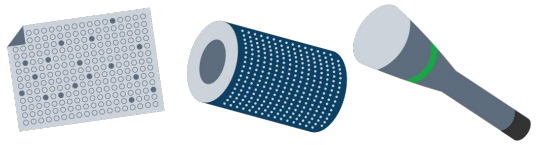






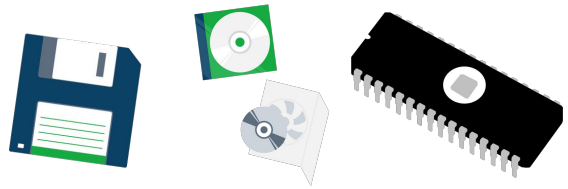
Sources: https://blocksandfiles.com/2019/08/20/back-to-the-future-with-persistent-memory/

Company problem #1: The memos
Aside: what constitutes storage?



1. Punch cards, magnetic drum, Williams-Kilburn tube (memory)
   a. *1890-1947.* 0.08 KB - 48 KB

2. Magnetic tape, hard disk drive, magnetic core (memory)
   a. *1951-1956*. 2 KB - 3 MB



3. Floppy disk, compact disk, digital video disk, semiconductor memory (memory and storage)
   a. *1967-1995*. 80 KB - 1.5 GB

4. Secure Digital card, USB flash drive, Blu-ray optical disk
   a. *1999-2003*. 64 MB - 25 GB

Sources: https://www.frontierinternet.com/gateway/data-storage-timeline/

Company problem #1: The memos
Aside: what constitutes storage?

- Punch cards
  - Physical, manual programming (note: thimbles!)
  - Write once, read many
  - Brushes/sensors read punched cards, store the program in memory, then executes code
  - Output can be saved to… more cards… :(

Sources: https://www.computerhope.com/jargon/p/punccard.htm

Company problem #1: The memos
Aside: what constitutes storage?

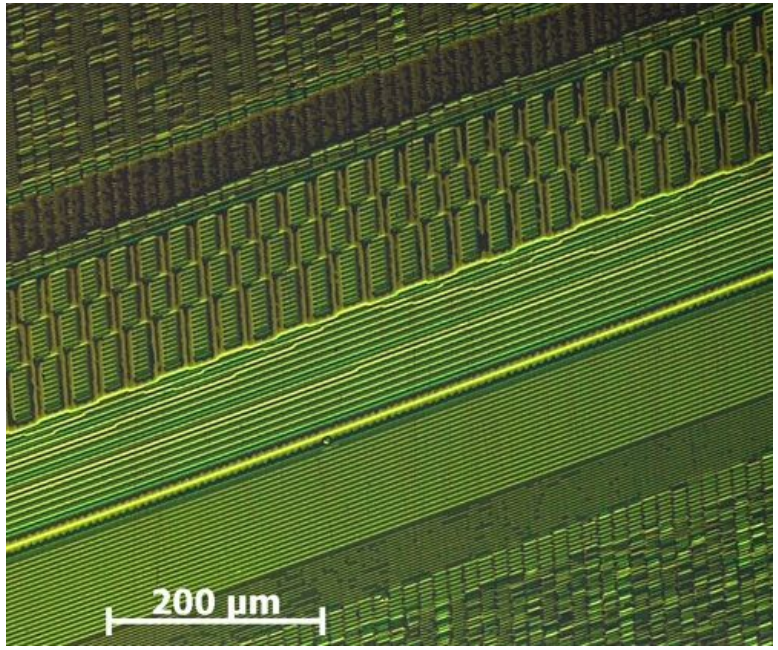
- Right: a program consisting of ~62,000 punch cards (~5 MB)

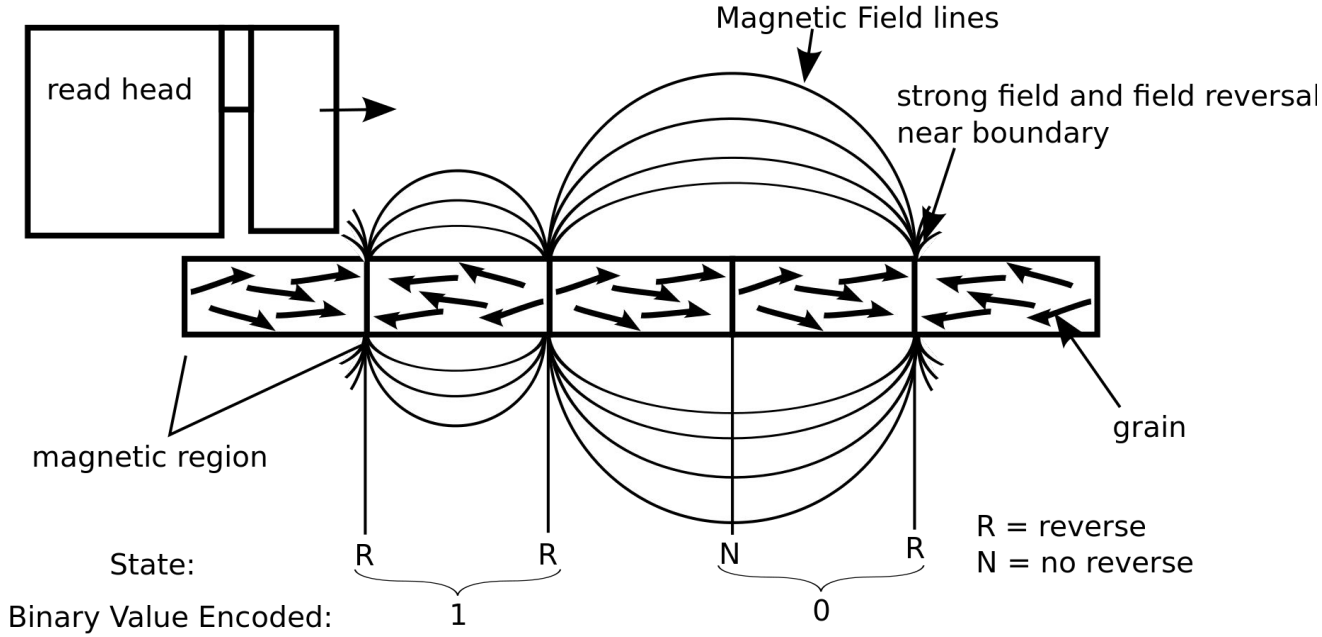
- **Bonus Q**: what happens when Mr. Bean knocks this over…?

Sources: https://www.lanl.gov/discover/publications/national-security-science/2020-winter/computing-history.shtml

Company problem #1: The memos
Aside: what constitutes storage?

- Punch cards, magnetic tape, hard disk drives are all similar: instead of holes in paper, they use magnetic cells to store a state

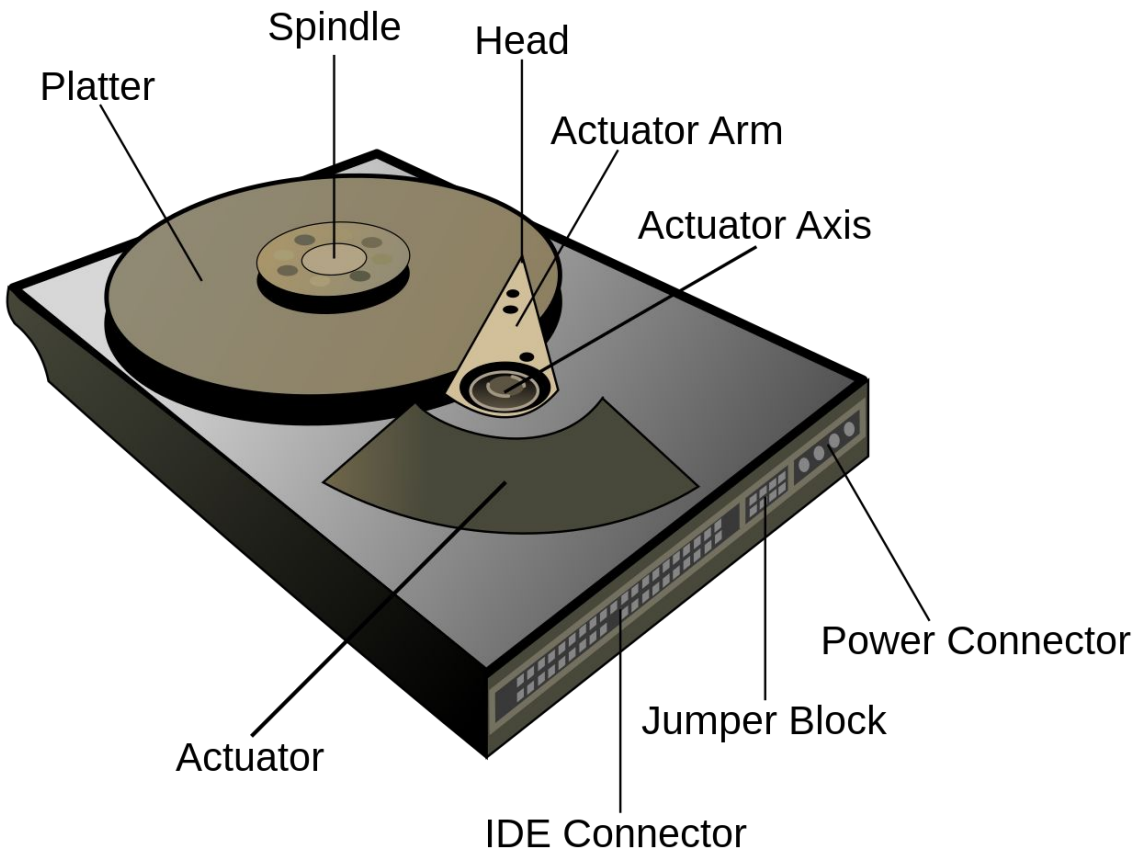




Sources: https://en.wikipedia.org/wiki/Hard_disk_drive

Company problem #1: The memos
Aside: what constitutes storage?

- How hard disk drives work
  - Spinning metal magnetic plates (platters), rotated very quickly
  - An actuator connected to a motor seeks locations on platters to read/write from
- How fast?
  - Commonly: 5400, 7200, 15000 RPM
    - ~60km/h, ~81 km/h, >>100 km/h!
  - Read/write: about 300 MB/s
  - Latency: ~4 **m**s



Sources: https://en.wikipedia.org/wiki/Hard_disk_drive

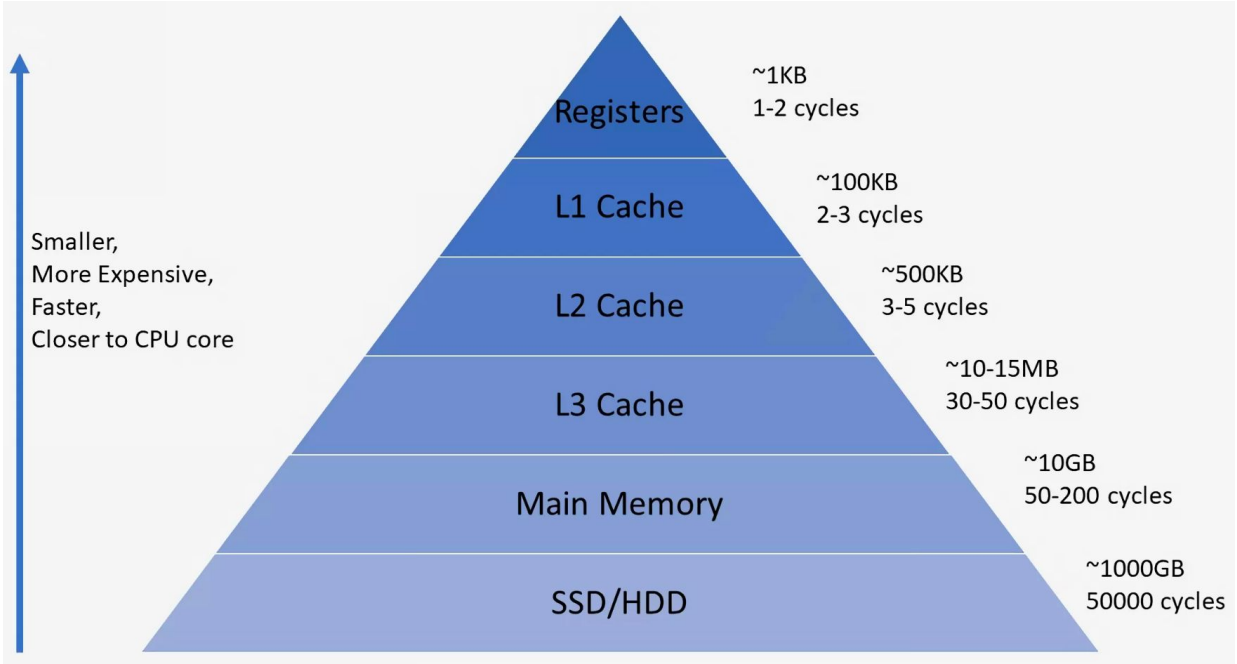
Company problem #2: The forklifts

2. All memos must be passed from location to location, in order, via forklifts that travel at 0.5 km/h.

Company problem #2: The forklifts
Solution: Trade off speed, capacity, and persistence

- Memory vs. storage
  - *Sequential vs. random access*
  - *Speed*
  - *Capacity*
  - *Re-usability*
  - *Persistence*

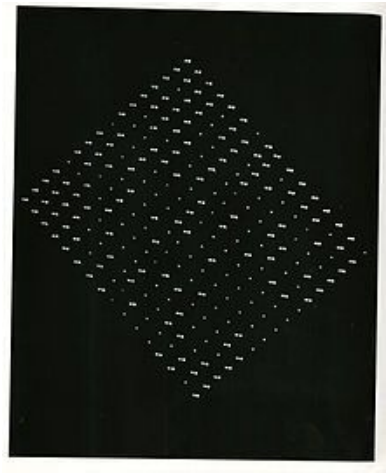
- "Memory" is one kind of storage
  - Fast, small, volatile, randomly accessed, and (usually) re-writable storage!



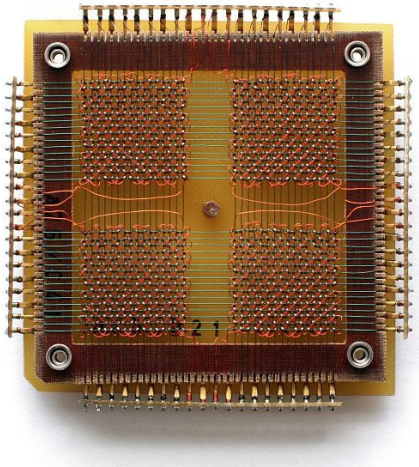
Sources: Marc's presentation

Company problem #2: The forklifts
Aside: Memory vs. storage

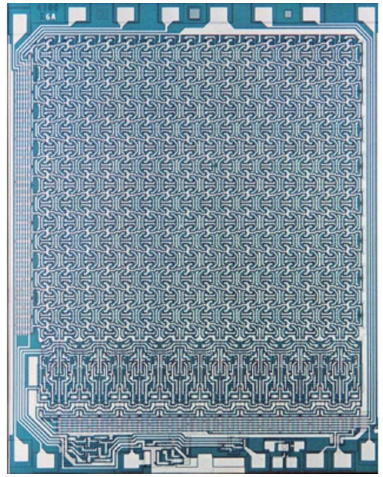
- *Random access memory (RAM):* Data read and written in any order
  - Don’t waste time waiting to read/write; need fast access to fast-changing data
  - Intermediates do not need to stored - storing a value takes time!



Williams-Kilburn tube
~512-2048 bits



Magnetic core memory
32x32 (1024) bits

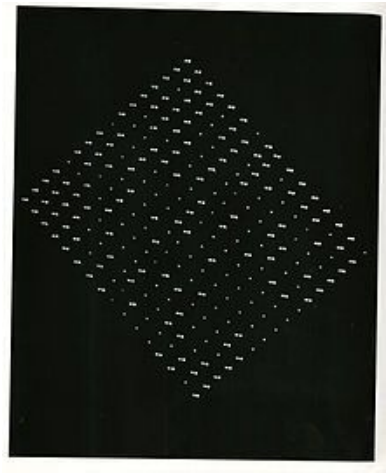


Semiconductor memory
256 bits

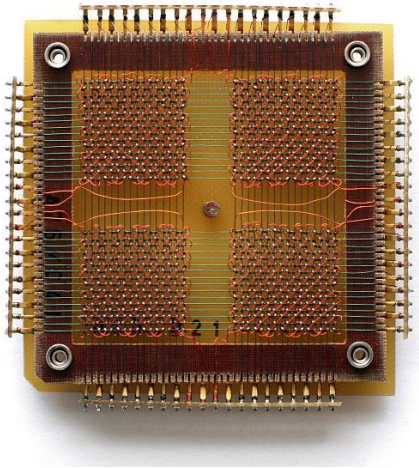
Sources: https://en.wikipedia.org/wiki/Memory_cell_(computing), https://www.computerhistory.org/siliconengine/semiconductor-rams-serve-high-speed-storage-needs/

Company problem #2: The forklifts
Aside: Memory vs. storage

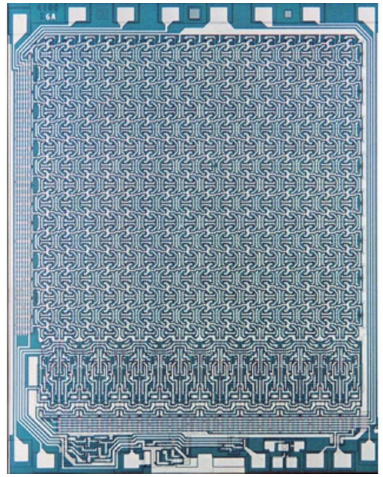
- Like disks, memory *physically* stores bits of information
- How bits are stored relates to material properties
  - E.g., semiconductor memory cannot retain its state without power; thus, *volatility*



Williams-Kilburn tube
~512-2048 bits



Magnetic core memory
32x32 (1024) bits

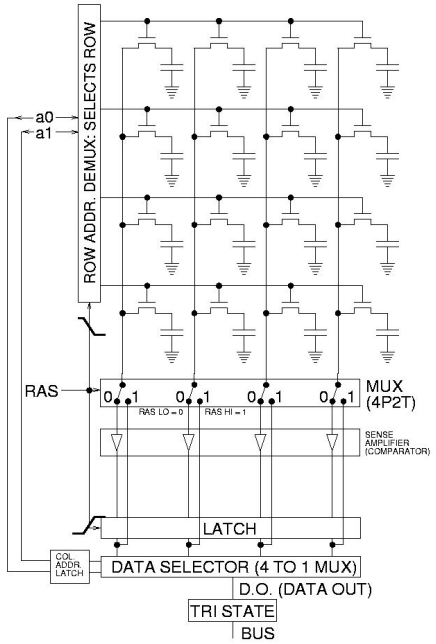


Semiconductor memory
256 bits

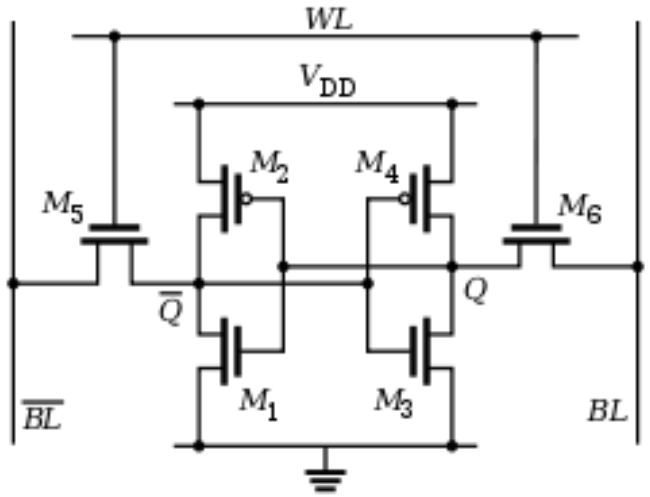
Sources: https://en.wikipedia.org/wiki/Memory_cell_(computing), https://www.computerhistory.org/siliconengine/semiconductor-rams-serve-high-speed-storage-needs/

Company problem #2: The forklifts
Aside: Memory vs. storage

- How data are actually stored in memory: *memory cells*
  - D-(dynamic)-RAM: Transistor+capacitor; used in memory modules
  - S-(static)-RAM: 6-transistor flip-flop circuits; used in CPU caches



4x4 DRAM array, each with transistor+capacitor



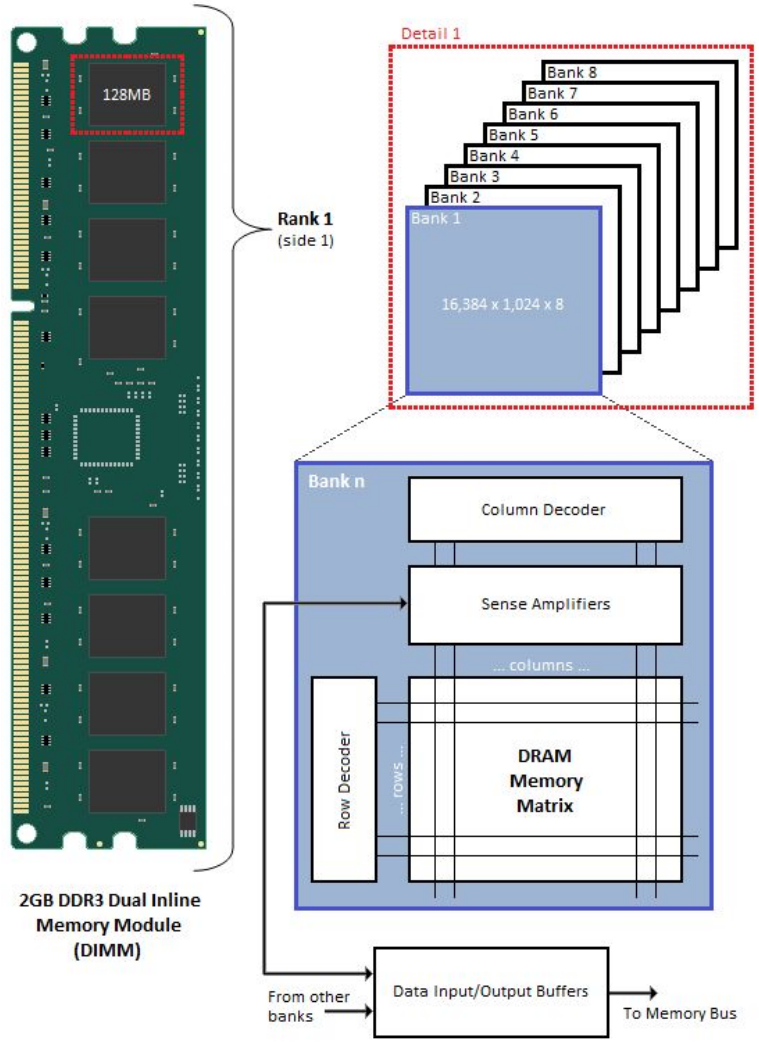
6-transistor SRAM cell

Sources: https://en.wikipedia.org/wiki/Static_random-access_memory , https://en.wikipedia.org/wiki/Dynamic_random-access_memory

Company problem #2: The forklifts
Aside: Memory vs. storage

- Modern semiconductor DRAM
  a. Smallest unit is the *memory cell*
  b. Cells arranged into rows (*word lines*) and columns (*bit lines)*
  c. Grids of bit lines and word lines arranged in *banks*
  d. Voltage states are detected using *sense amplifiers*
  e. Decoders translate bank coordinates into *addresses*
  f. Addresses are passed off to I/O buffers, which are passed off to the *memory controller*

- Round trip time: > 50ns
  a. Basically **ages** compared to on-chip cache (<1 ns)!
  b. Like CPUs, modern memory has a *clock* that governs synchronicity



Sources: https://www.anandtech.com/show/3the memory bus851/everything-you-always-wanted-to-know-about-sdram-memory-but-were-afraid-to-ask/2

Company problem #2: The forklifts
Aside: Memory vs. storage

1. Set the address (of the memory location) on the **address bus**.

2. Set the *read/write* wire of the **control bus** high (i.e. request a read operation).

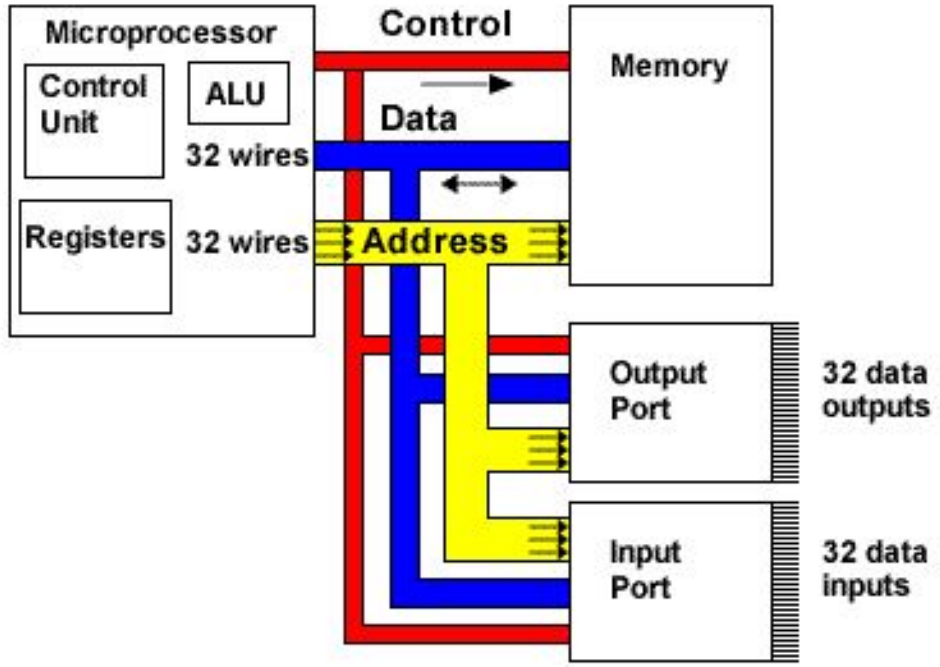
3. Set the *address valid* control wire high.

4. The *address valid* signal, together with the value on the **address bus** will activate the *chip select* wire on the appropriate memory chip.

5. The contents of the memory location will now be placed on the **data bus**.

6. Read the value from the **data bus** - usually into a *register* in the microprocessor.

7. The *read/write*, *address valid* and *chip select* wires can now all be set low.

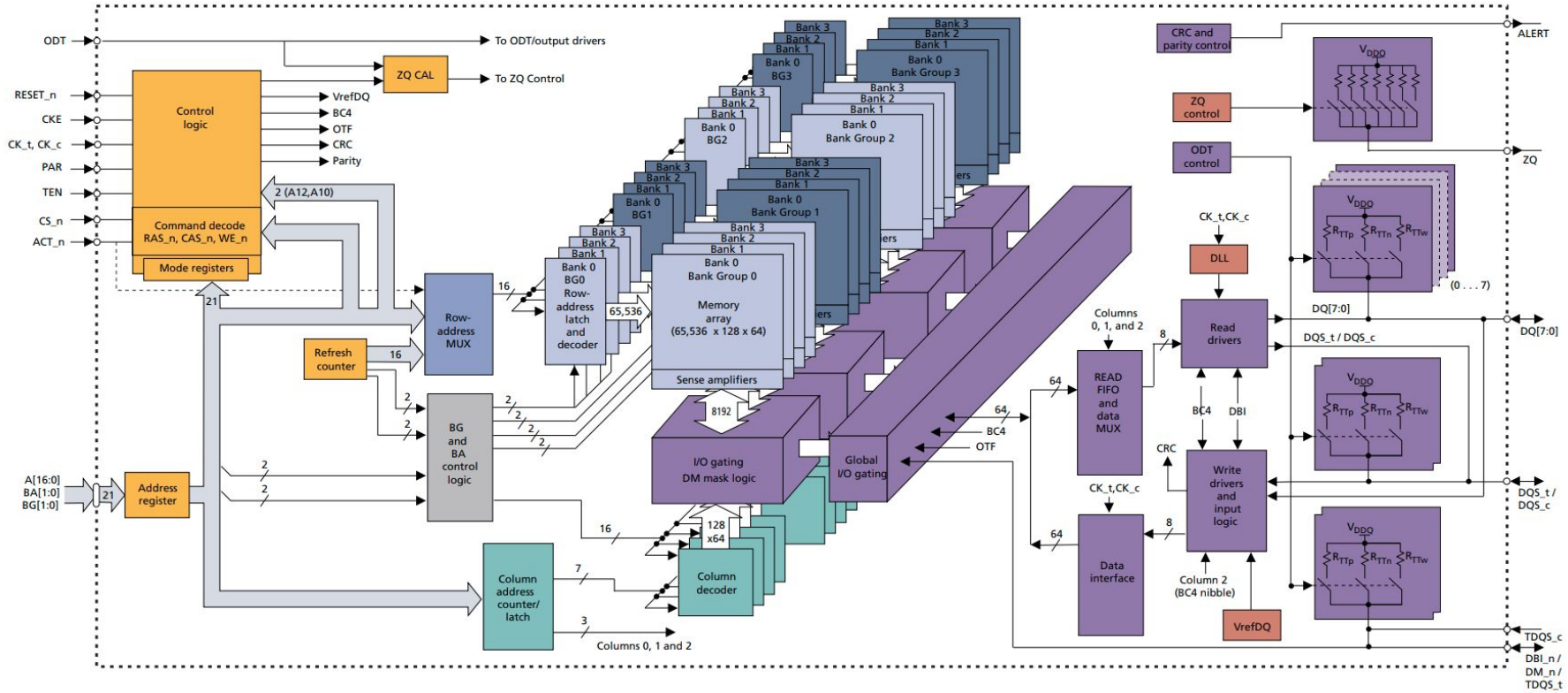


Sources: http://www-mdp.eng.cam.ac.uk/web/library/enginfo/mdp_micro/lecture1/lecture1-3-2.html

Company problem #2: The forklifts
Aside: Memory vs. storage

- DDR4 memory: basically, **a lot** of memory cells (plus some spice)

Figure 1: 8Gb, x8 DDR4 SDRAM Functional Block Diagram

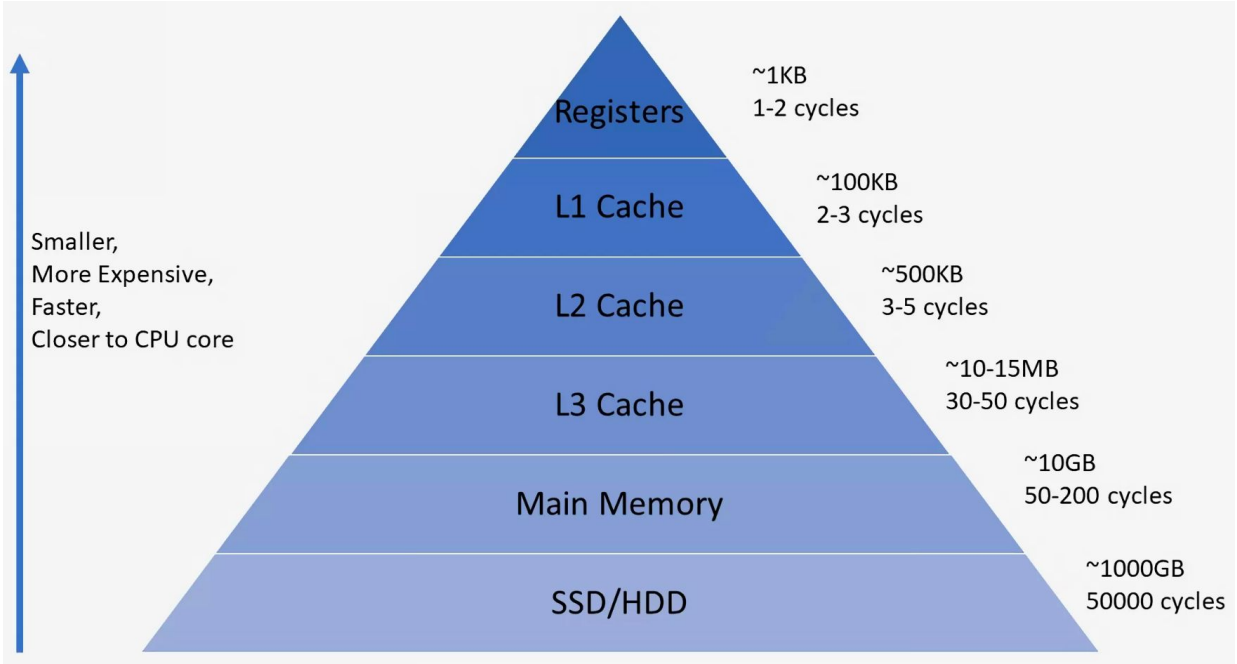


Sources: https://www.micron.com/-/media/client/global/documents/products/technical-note/dram/tn4007_ddr4_power_calculation.pdf

Company problem #2: The forklifts
Solution: Trade off speed, capacity, and persistence

- *Recall*: Memos must be passed, in order, via slow forklifts.

- Solutions:
  - Allow simultaneous read/writes
    - *random vs. sequential* memory
  - Use race cars instead of forklifts; send emails instead of memos
    - Don't use hard disks as L1 cache
  - Increase forklift speed
    - Memory clock too slow? Run it faster!



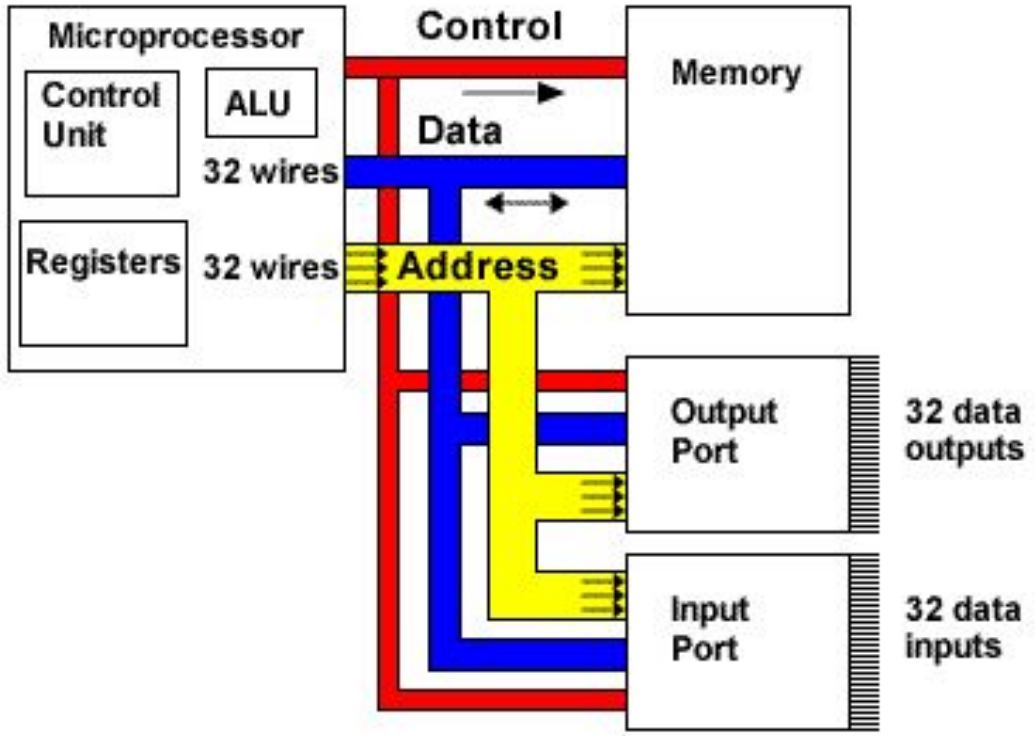
Sources: Marc's presentation

Company problem #3: The hallways

3. All forklifts travel through a single hallway to deliver their memos or their goods, creating a traffic jam.

Company problem #3: The hallways
Solution: Make bigger hallways

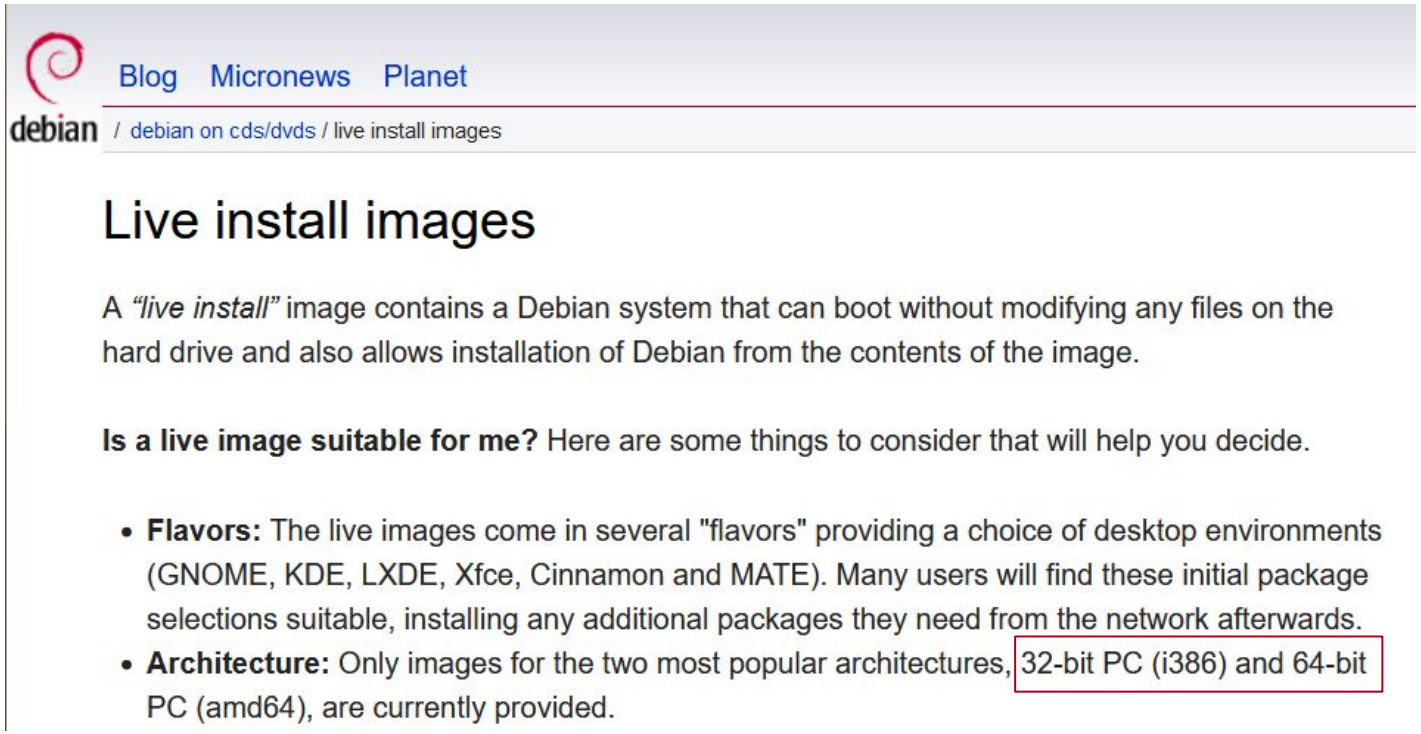
- Memory ⟶ memory controller ⟶ memory bus ⟶ CPU
  - Locations in memory must be *addressable*
  - Address space is related to the width of the address bus
    - 16-bit = 64 KB or 512,000 bits
    - 32-bit = ~4 GB or 2^{32} bits
    - 64-bit = 2^{64} bits
  - Does it matter how bits and bytes are addressed?
    - No, it is arbitrary: *endianness*



Sources: http://www-mdp.eng.cam.ac.uk/web/library/enginfo/mdp_micro/lecture1/lecture1-2.html

Company problem #3: The hallways
Solution: Make bigger hallways

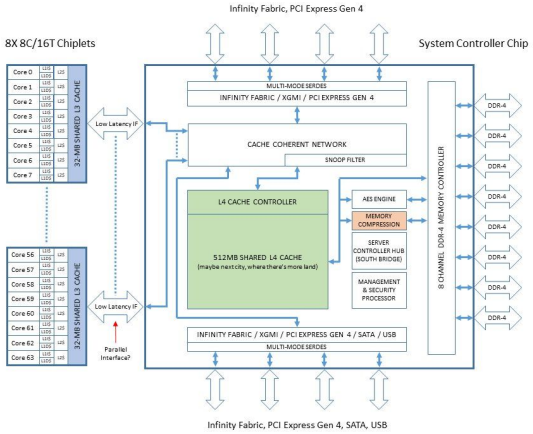
- Memory address spaces: practicality
  - Bigger address spaces = more addressable memory
    - e.g. 32-bit OSes cannot use >4 GB RAM
  - Binaries built for one space may not work in another

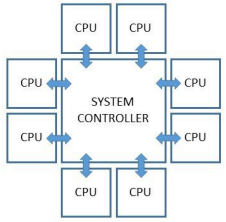


64-bit, before it was cool
(or really necessary)

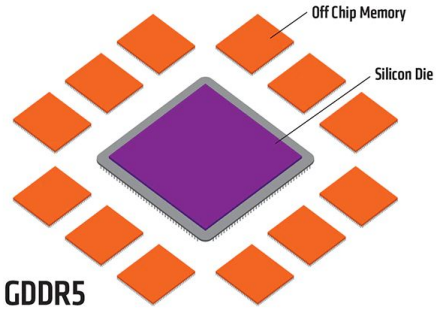
Company problem #3: The hallways
Solution: Make bigger hallways

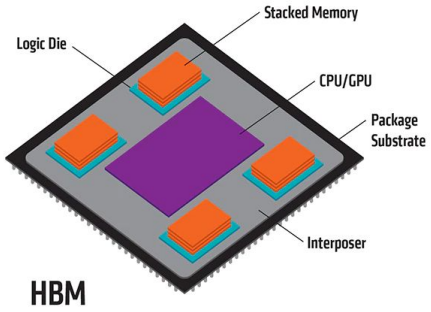
- So: what constitutes a “bigger hallway”?





Use more memory channels
(8-channel vs. 2-channel)





Use higher-throughput memory
(DDR4: ~70 GB/s; HBM2: 256 GB/s)

Use bigger bus width
(Top: AMD 6500XT, 64-bit bus
Bottom: AMD 6800XT, 256-bit bus)

Sources: https://en.wikichip.org/wiki/amd/microarchitectures/zen, https://www.amd.com/ja/technologies/hbm

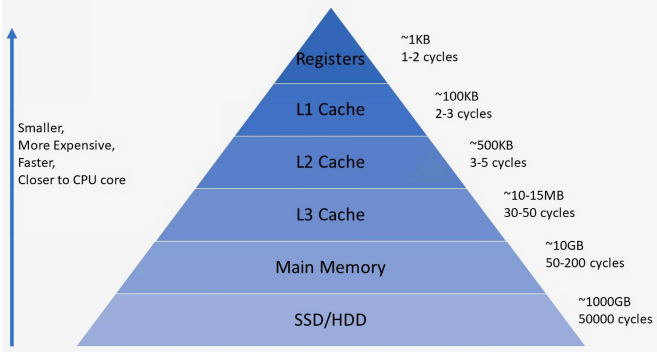
Company problem #4: The space

4. Products sent to your business are immediately returned if there is insufficient shelf space.

Company problem #4: The space
Solution: Use other storage media

- Insufficient memory
  - Worst case: complete system failure
  - Best case: use slower memory to make up the difference
    - “Swapping out to disk”
    - Recall that HDDs are ~50,000X slower than L1 cache!

- Capacity used to be a huge limiting factor
  - Programmers were *forced* to be efficient: Apollo landed on the moon with 4 kB of RAM and 72 kB of ROM!



Company problem #4: The space
Aside: Limits of memory size

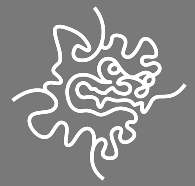
- What are the limits of memory capacity?
  - Today? ~4 TB
  - Memory cells can be as small as transistors+capacitors allow
    - “There’s plenty of room at the bottom” - Richard Feynman

One DIMM DDR4, 128 GB =
137,438,953,472 transistors

Sources: https://semiconductor.samsung.com/dram/module/

Company problems and solutions

Problem	Solution
1. The memos	1. Don’t store everything: random vs. sequential
2. The forklifts	2. Trade offs: speed, capacity, and longevity
3. The hallways	3. Increase bandwidth: bigger hallways
4. The space	4. Increase capacity: get a bigger factory



Your business is saved!

…thanks to an interest in semiconductors and computing history.

From here, please vote on topics you'd like discussed.

1. You decide to **replace all forklifts with Ferraris**. How do they work? (solid-state drives)

2. Your forklifts keep breaking! Have you tried **taping a bunch of them together**? (RAID)

3. Because your instructions are encoded in plain English, your enemy - the mountebank Mr. Fawlty - has been **stealing company secrets**… (security)

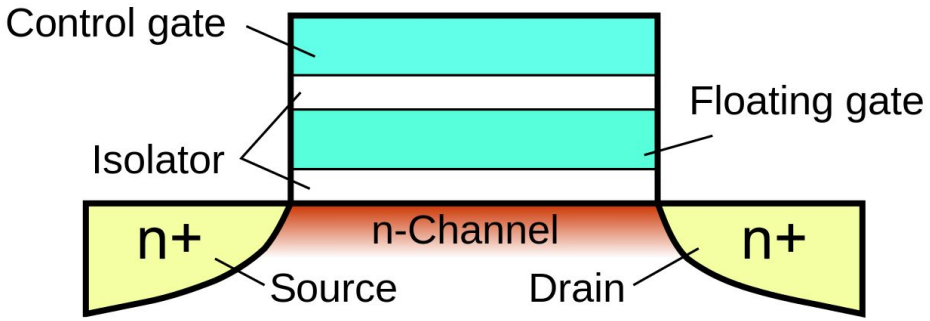
4. You grow your facility to the **maximum legally allowable** size. Now what? (compute clusters)

5. There are so many Ferraris, Segways™, hovercrafts, skateboards, and rockets, **you have difficulty deciding**… (basically, Mike's advice on memory and storage)

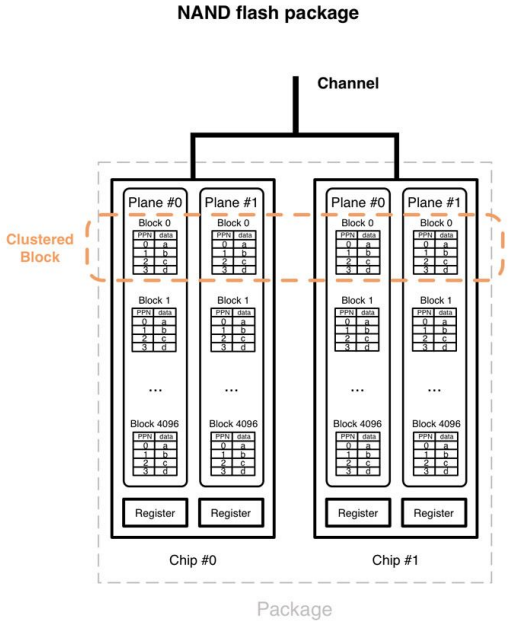
1. The Ferraris of storage: solid state drives

- Special floating-gate MOSFET transistors retain state without power
  - Hold their charge for ~10 years
  - Unlike DRAM, reading/writing to flash cells wears them out

- Like DRAM, NAND flash organized into planes, blocks, etc.



Flash memory cell: floating-gate MOSFET



1. The Ferraris of storage: solid state drives

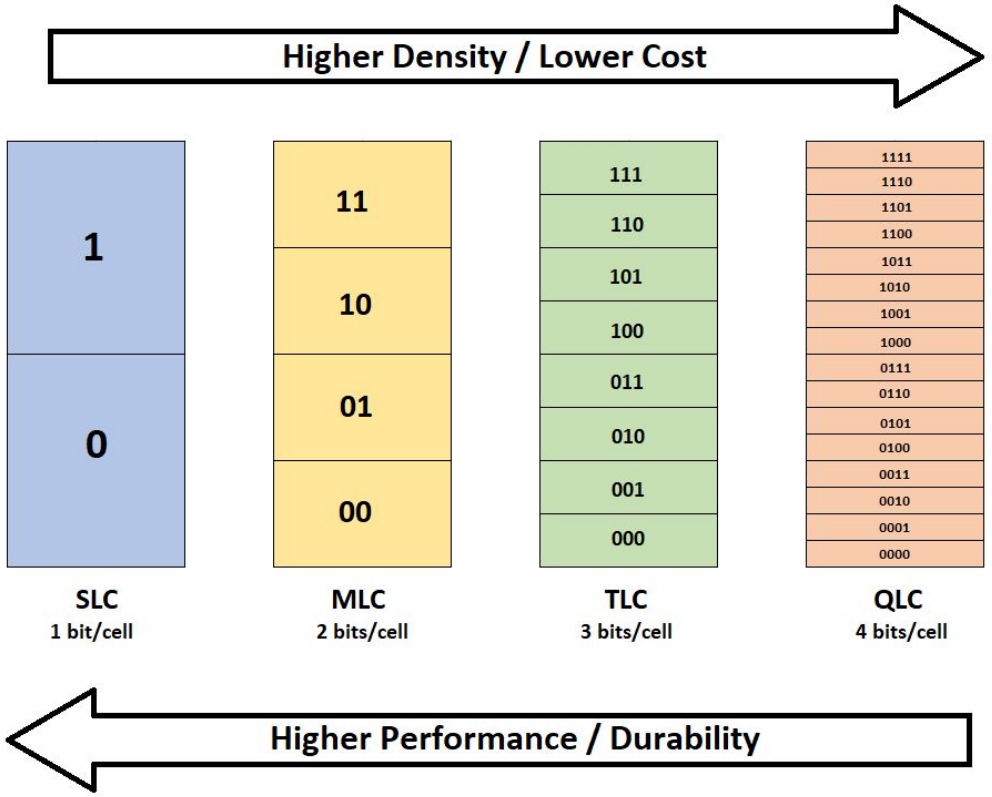
- Caveat emptor: not all SSDs are equal
  - 2.5” SATA (slow), M.2 SATA (slow), and M.2 NVMe (fast) drives all exist
    - There are also 3 NVMe revisions- 5.0 is double speed of 4.0, which is double 3.0

Left to right: M.2 NVMe, M.2 SATA (notice connector), 2.5” SATA

1. The Ferraris of storage: solid state drives

- Quality of *NAND flash* matters a lot
  - Not always reported
  - Manufacturers sometimes change it without telling anyone!

- SLC is highest-performing, highest-endurance
  - No consumer SSDs use SLC directly anymore
  - SLC is actually used to *cache* TLC and QLC - all in one SSD



1. The Ferraris of storage: solid state drives

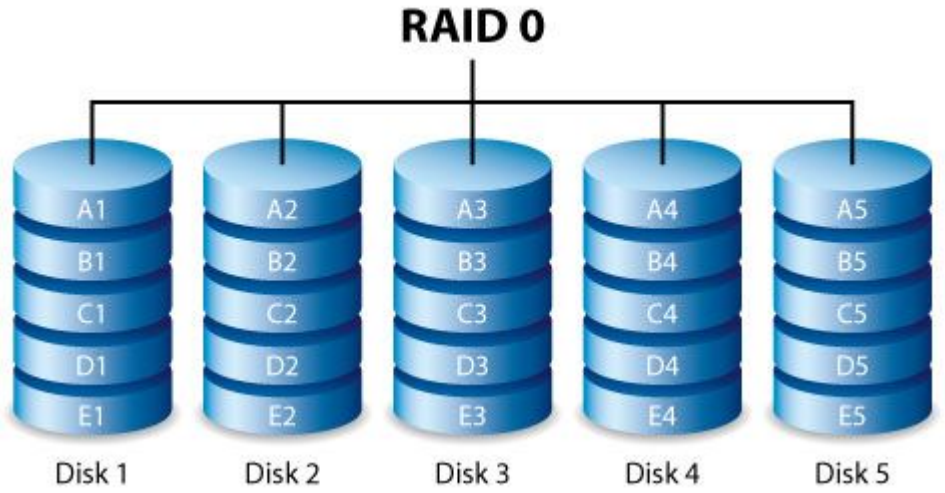
- The *true* king of SSDs: Intel Optane
  - Not for consumers (multiple thousands of dollars)
  - Extremely low latency (~50µs)
  - Extremely high endurance (100 DWPD)

- Almost no one on earth needs this. Don’t buy it.

2. RAID: or, taping forklifts together

- Hard disk drives are *fragile* - they fail more than any other component

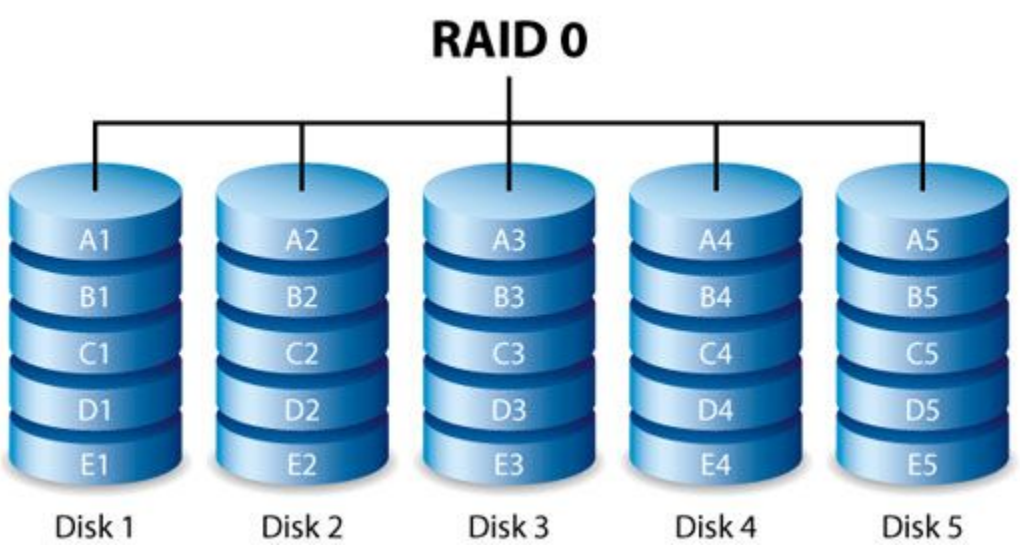
- Like memory interleaving, read/write operations can be split between multiple drives
  - Redundant Array of Independent Disks

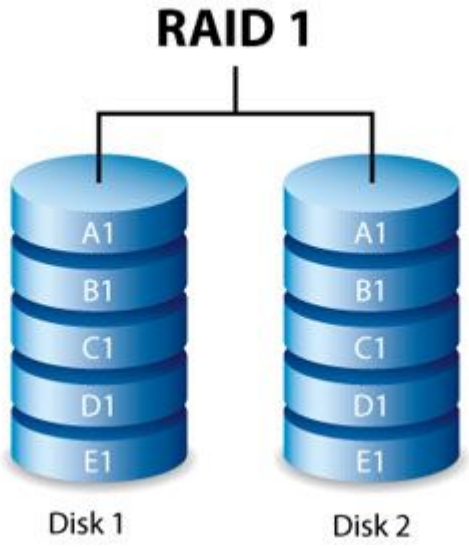


Sources: https://www.seagate.com/jp/ja/manuals/network-storage/business-storage-nas-os-4/raid-modes/
Check out BackBlaze's statistics on drive failures - https://www.backblaze.com/b2/hard-drive-test-data.html

2. Taping forklifts together to get work done

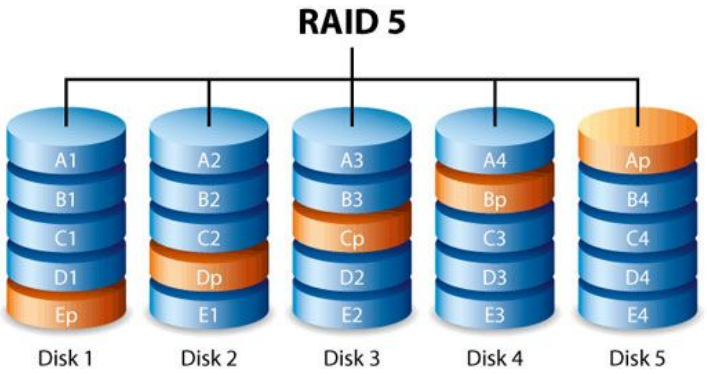
- RAID has different “levels”
  - RAID0, aka “striping” - split read/writes between multiple disks
    - Speeds up I/O: no single disk is limiting
    - If one drive dies, the data is kaput
  - RAID1, aka “mirroring” - make a copy of one disk on another
    - Data are now redundant. But not independent!
    - “RAID is not a backup.” “RAID is not a backup.” “RAID is not a backup.”

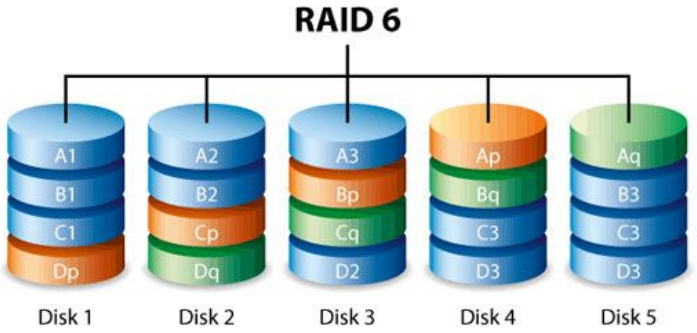


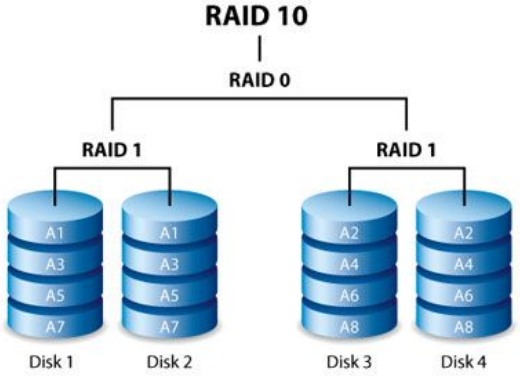


2. Taping forklifts together to get work done

- RAID has different “levels”
  - RAID5, RAID6, etc.
    - Combines mirroring and striping together, giving redundancy and I/O improvement
  - There are more…

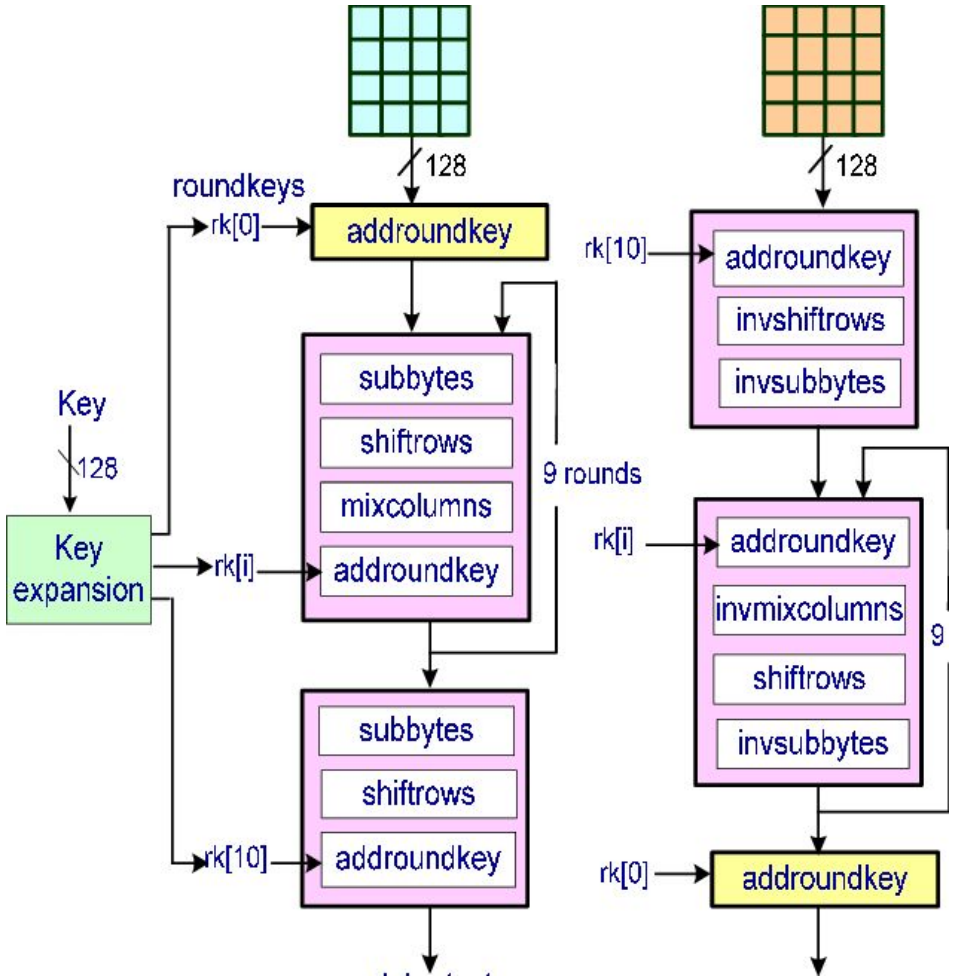






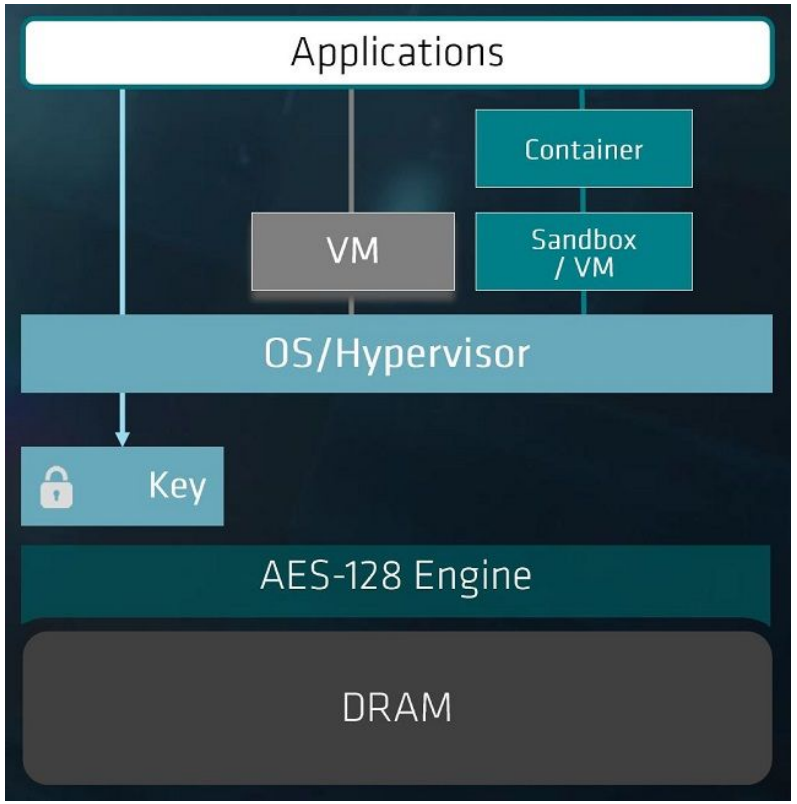
3. Dealing with the dastardly Mr. Fawlty

- Data on disks sometimes needs additional security
- Instead of storing data directly, apply a cipher to transform the data: *encryption*
- Many ciphers, implementations exist
  - e.g. AES, LUKS



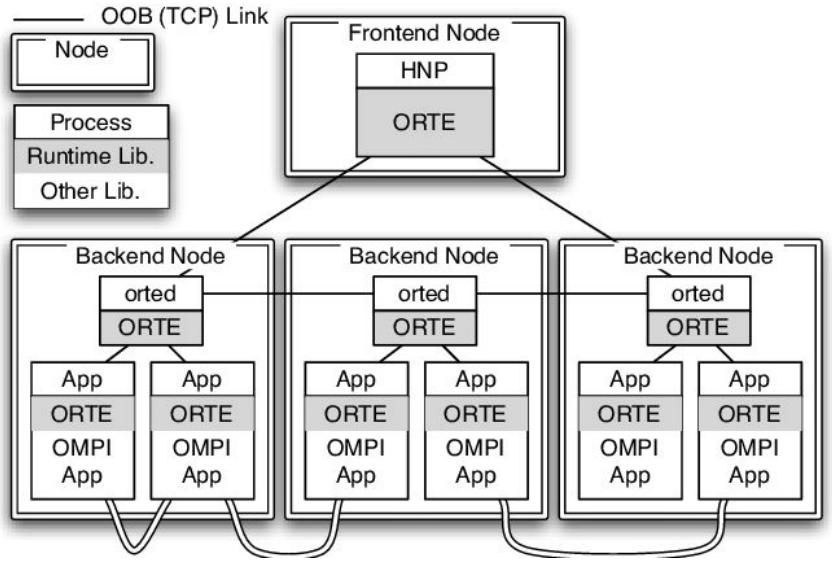
3. Dealing with the dastardly Mr. Fawlty

- Note that *data in memory* is also vulnerable
  - Spectre/Meltdown: security vulnerabilities that exploit speculative execution and CPU cache memory
- The future: fully-encrypted memory

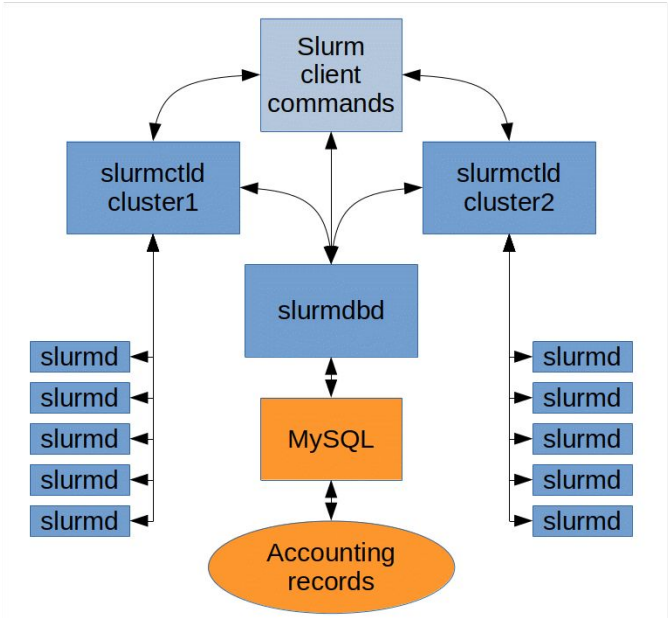


4. Maximum size servers

- A single computer can only have access to so much system memory
- How can you scale after maxing out memory? Use multiple computers



OpenMPI

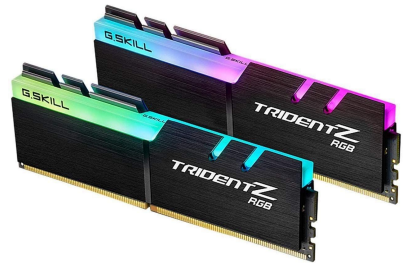


Job schedulers, e.g. SLURM

Coming soon: CXL
PCIe 5.0-enabled coherent memory
I.e. memory-sharing between nodes

5. Mike's opinions on buying RAM and storage

- What RAM to buy?
  - *Capacity*.
    - I wouldn't buy less than **16 GB**
    - I try keep my devices for at least 5 years. 16 GB is *just* suitable today. It probably won't be tomorrow.



  - *Clock speed and latency*.
    - If you play games: get **fast RAM with good timings** (e.g. DDR4 3600Mhz CAS 16; DDR5 no idea, still too expensive). For games, memory latency matters - thus, *cache size* matters
    - If you don't play games: you probably won't notice a difference with increased memory speed

  - *Bandwidth*.
    - Make sure both memory channels are populated (**even laptops**)
    - You should already know if you need 4+ channel memory if you are considering it ($$$)
      - If you play games, web browsing, whatever: No way
      - If you crunch numbers of pi or do fluid dynamics simulations every day: **essential**

5. Mike's opinions on buying RAM and storage

- Storage recommendations
  - *Hard disk drives*
    - Most people would be better off buying cloud storage and forgetting about HDDs
    - *Acceptable use cases*. A good way to **back up large volumes of data**, e.g., a PhD thesis. Look into network-attached storage solutions from QNAP, Synology, …
    - Send Mike a message before buying if you're on the fence

5. Mike's opinions on buying RAM and storage

- Storage recommendations
  - *Solid state drives - Internal devices (SATA, M.2, etc.)*.
    - M.2 for laptop: **SK Hynix P31**. Best on the market for power efficiency.
    - M.2 for desktop: depends on CPU. The Samsung EVO/PRO lines are quite good. Many decent products in here; many traps though! **Avoid QLC** unless you know what you're doing.
    - Budget-friendly: In everyday use, SATA SSDs are as good as NVMe. A good SATA drive, e.g., **Crucial MX500** can save money to put towards something that will actually affect your enjoyment of the system (e.g., a nice monitor).

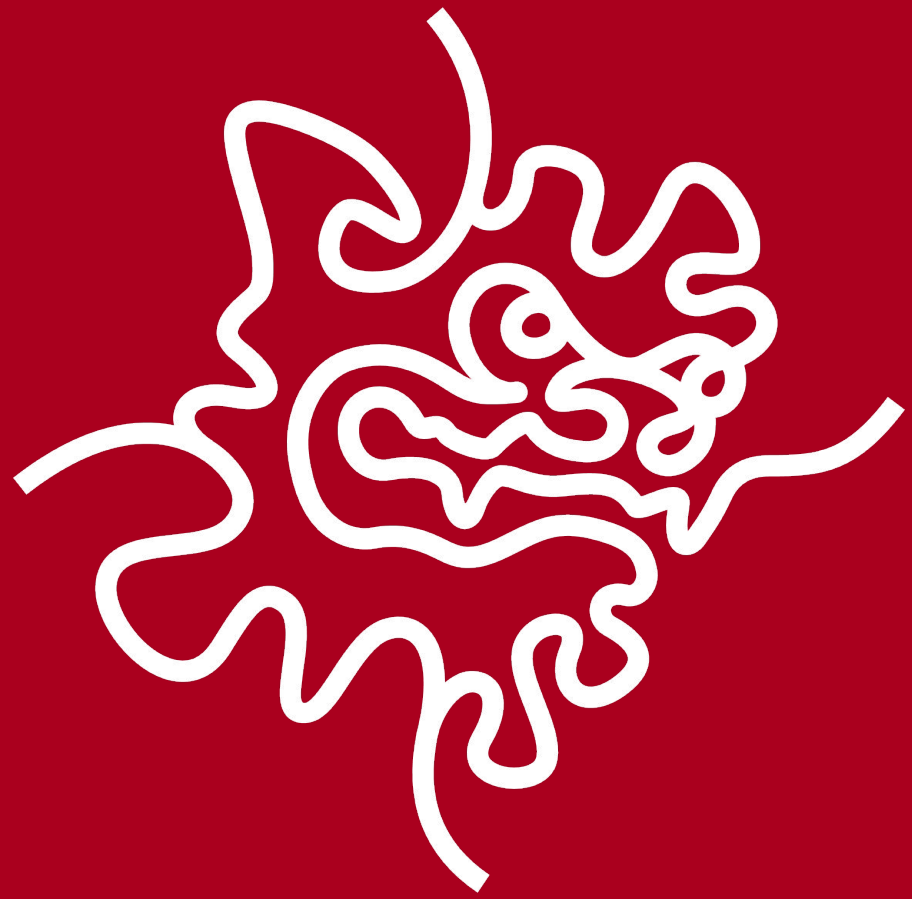
5. Mike's opinions on buying RAM and storage

- Storage recommendations
  - *Solid state drives - External devices*
    - SD card: I have Samsung brand loyalty, never failed me. Kingston should be fine.
    - External storage: I have a rando-brand M.2 enclosure with a **WD Blue M.2** drive and it's fine. I wouldn't spend a lot on either an external drive or its enclosure (>$200 = :/)
    - USB: I don't care really. Note that USB naming is a disaster and all of the following standards are completely different:
      - *2.0*
      - *3.0*
      - *3.1*
      - *"3.2"* aka *3.2 Gen 1*
      - *3.2 Gen 2*
      - *3.2 Gen 2x2*
      - *4.0*
      - Also none of these are the same as Thunderbolt (USB 4.0 is close)

5. Mike's opinions on buying RAM and storage

If you learn nothing else, learn this!

- 3, 2, 1 Rule:
  - 3 backups
  - 2 on different local media (i.e., different disks)
  - 1 must be offsite (i.e., cloud)



Thank you!