Why should you do an OS class?

IN THE AGE OF DATA CENTERS, HETEROGENEOUS HARDWARE AND TARGETED SECURITY ATTACKS

Anton Burtsev
www.cs.utah.edu/~aburtsev
anton.burtsev@utah.edu

Operating systems haven't changed for decades

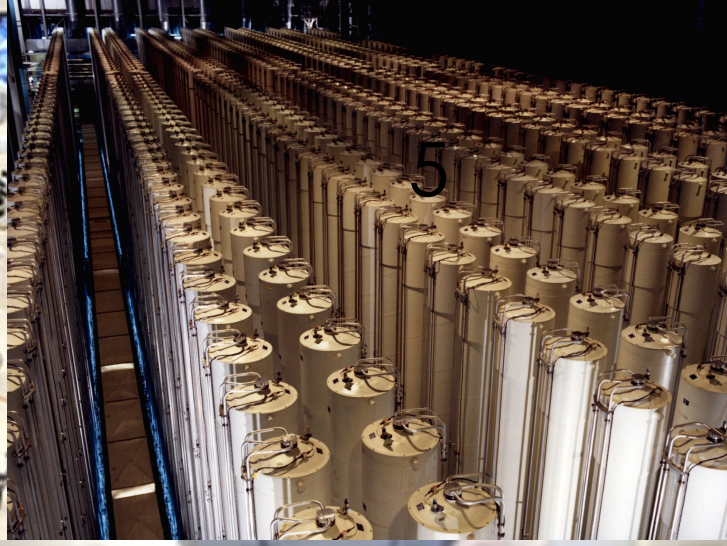
- 40 years old
  - Time-sharing
  - Expensive hardware
  - Overly general

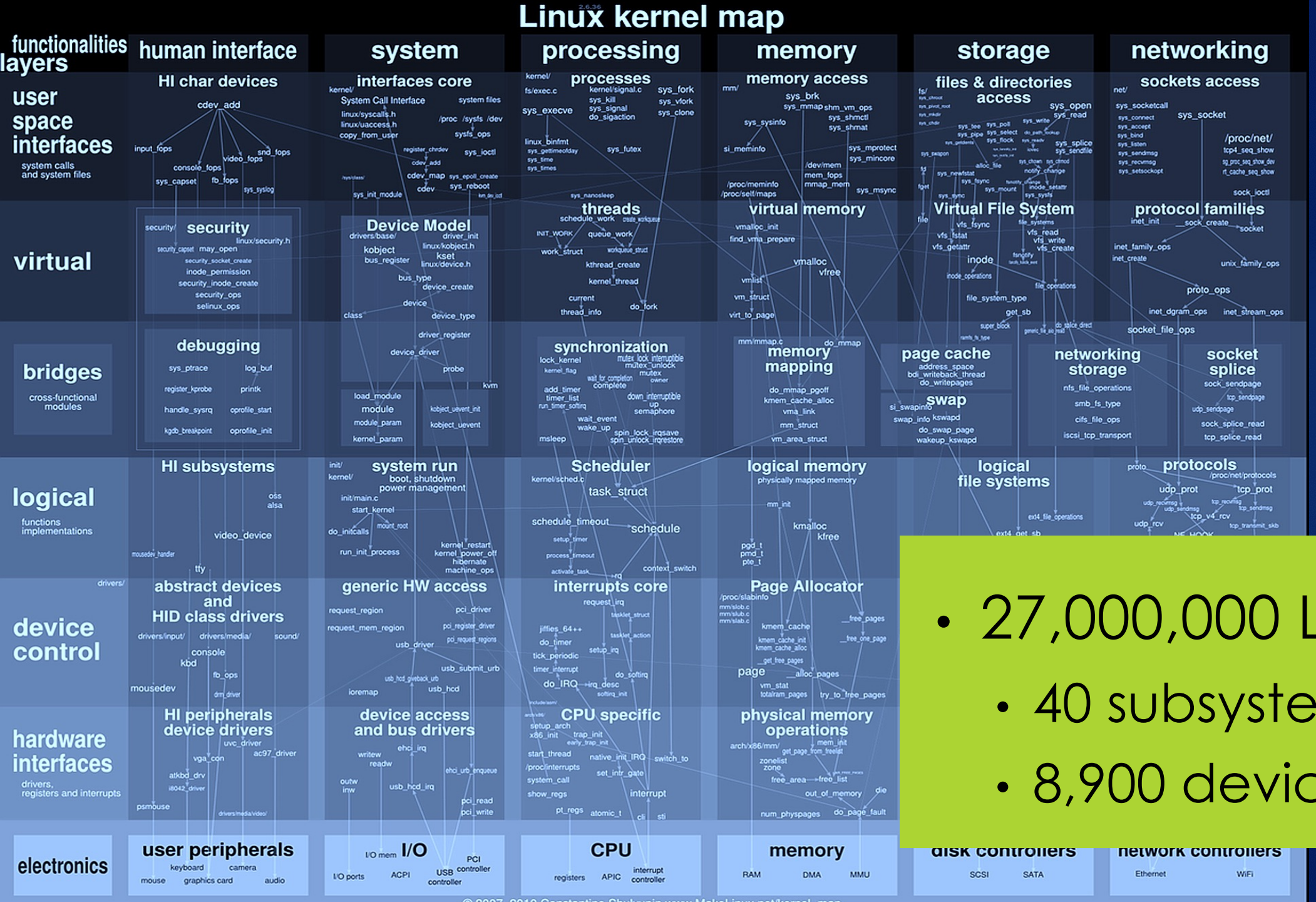
Ken Thompson (sitting) and Dennis Ritchie working together at a PDP-11 (1972)

cs5460/6460 teaches this system

- Xv6 is an x86 implementation of UNIX 6th edition
- All lectures are recorded
  - You're welcome to take a look

OS kernels are ubiquitous

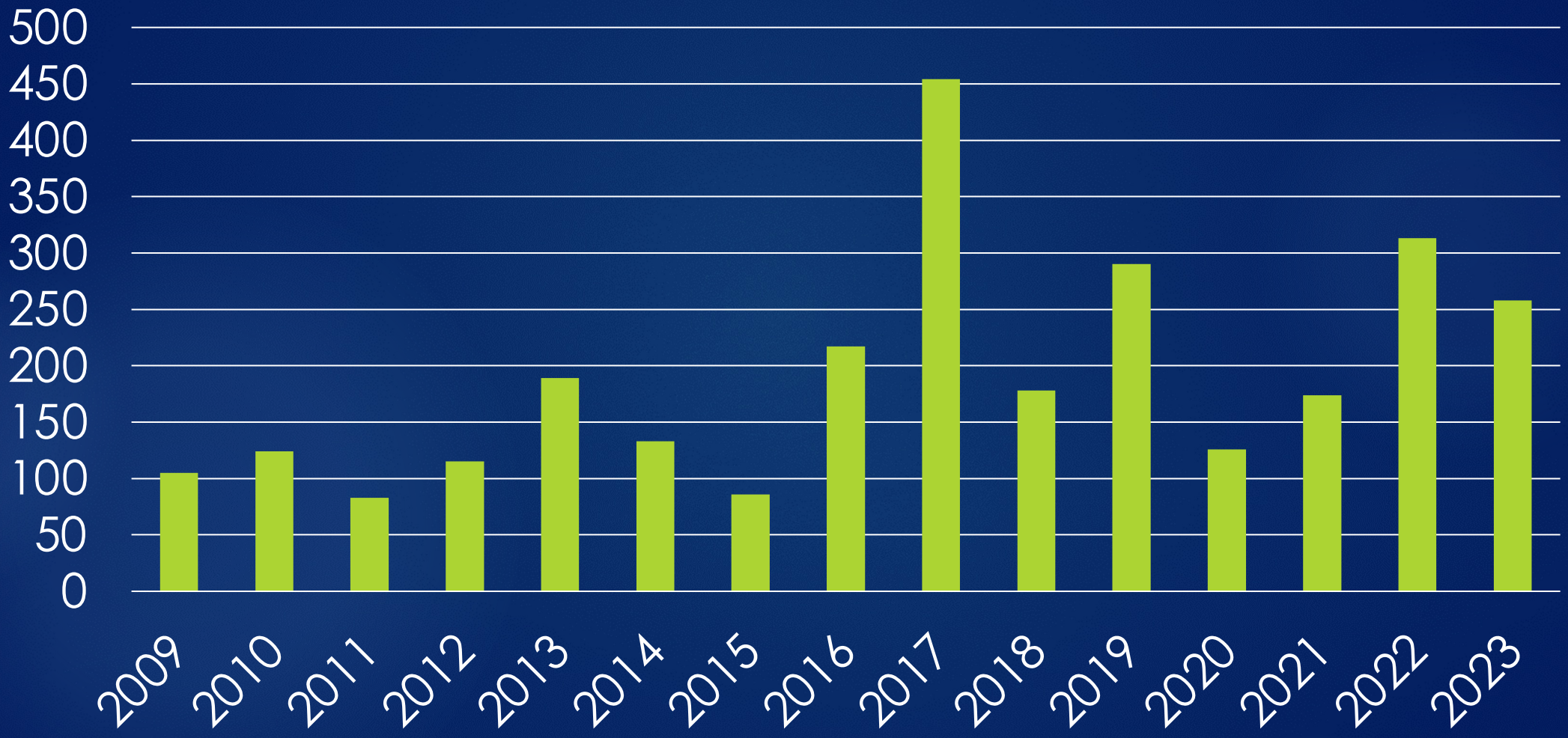




- 27,000,000 LoC
  - 40 subsystems
  - 8,900 device drivers

Problem #1: Security

Linux Kernel Vulnerabilities by Year



Example


```
static bool dccp_new (...) {
  struct dccp_header _dh, *dh;

- skb_header_pointer(skb, dataoff, sizeof(_dh), &dh);
+ skb_header_pointer(skb, dataoff, sizeof(_dh), &_dh);
};
```

Stack smash

Correct

- Remote exploit in Linux network firewall
  - Arbitrary code execution
  - Linux Kernel v 3.0 (June, 2011) – 3.13.6 (March, 2014)
  - CVE-2014-2523

In a modern system, an attacker is **one kernel vulnerability away** from gaining complete control of the entire machine

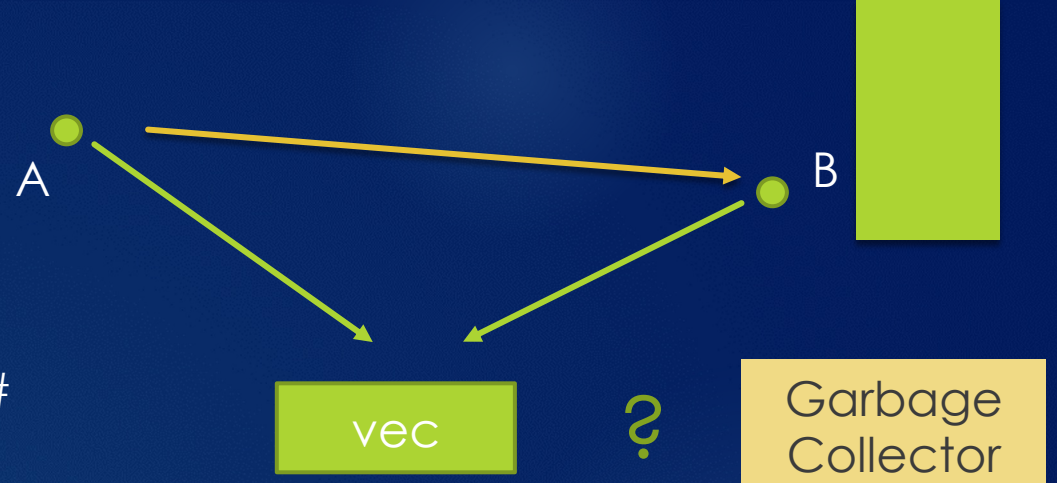
- Not going to change


Can we make these systems secure?
RedLeaf Operating System
Rust + Dafny-style verification

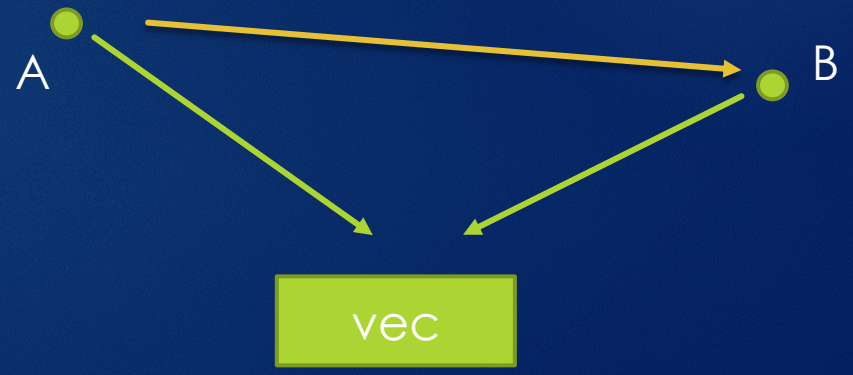
Rust

- Safe language build around idea of linear types
  - Normally, safety requires a garbage collector
    - Multiple pointers can point into an object
    - Even if one pointer is deallocated we don't know if there are other aliases
  - In Rust there are no aliases!
    - No need to walk the heap

Java, C#



Rust



Rust is the first safe alternative to C for low-level systems code

- Safe code remains fast
  - No garbage collection
  - Lightweight fine-grained software isolation
  - Zero-copy communication across isolated subsystems

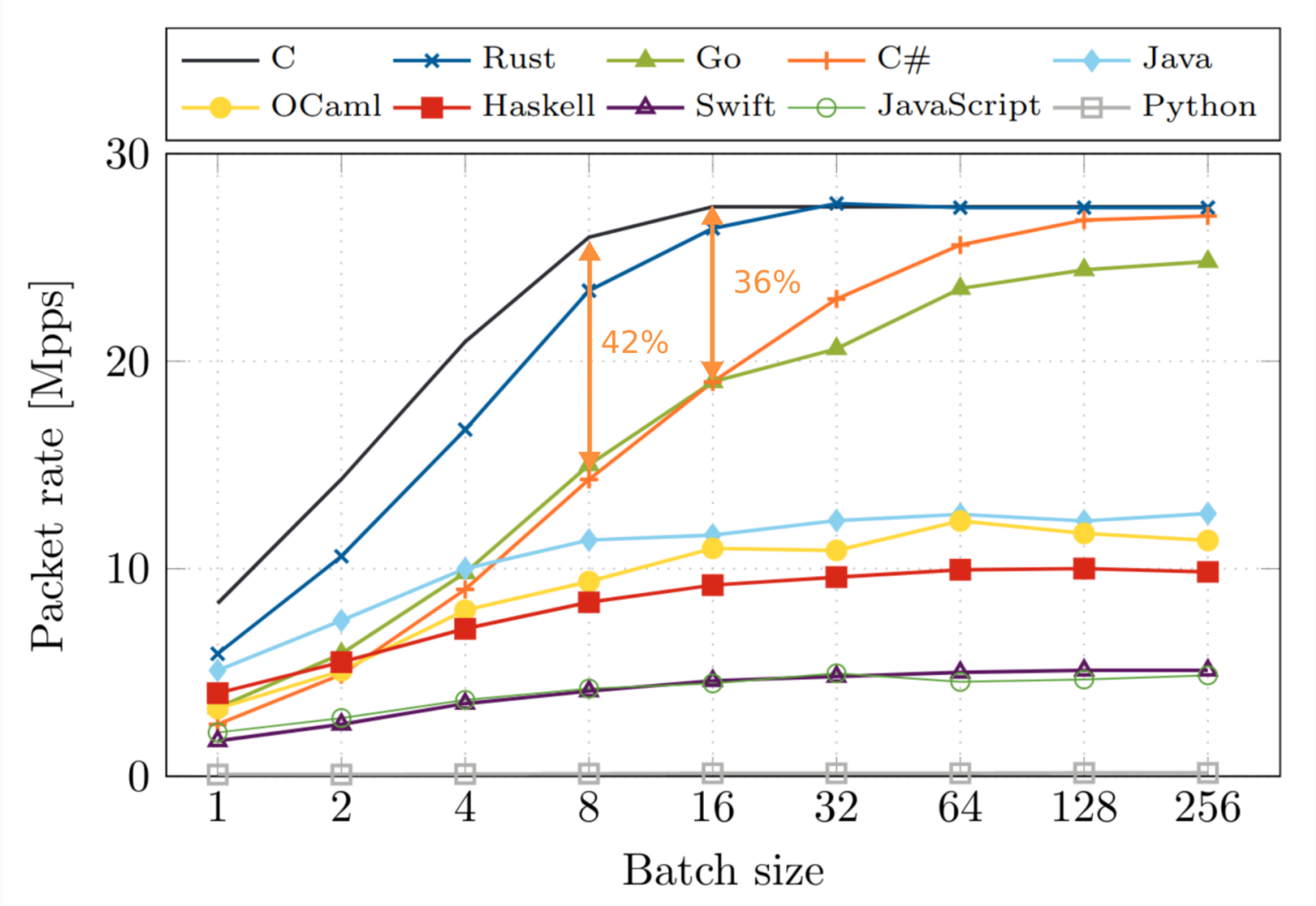


Figure 1: Forwarding rate for a minimal DPDK-like device driver implemented in 10 different languages. The driver uses one CPU core to forward packets on two 10 Gbit/s Intel X520 NICs.[2]

[2] The Case for Writing Network Drivers in High-Level Programming Languages, ANCS 2019

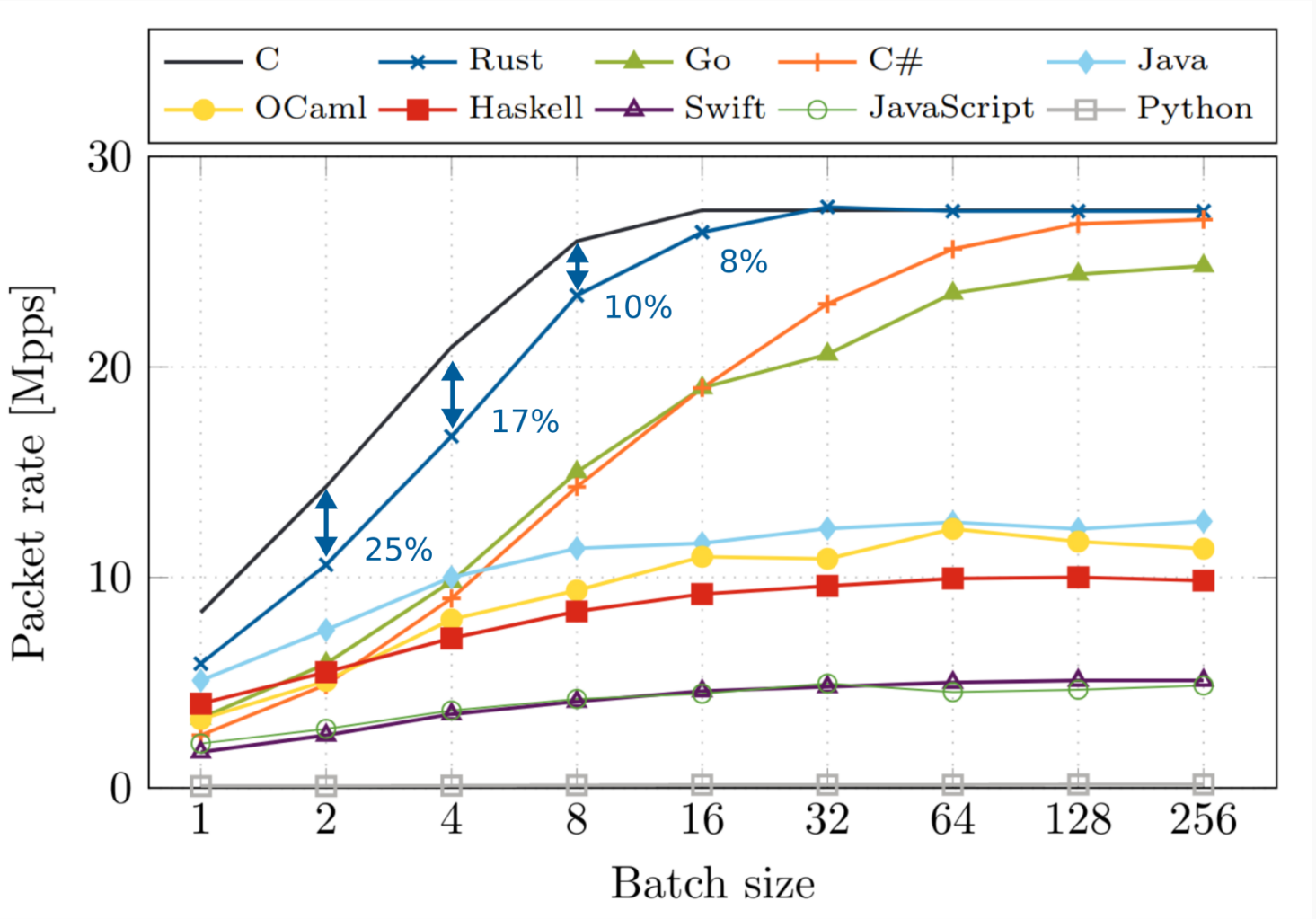


Figure 1: Forwarding rate for a minimal DPDK-like device driver implemented in 10 different languages. The driver uses one CPU core to forward packets on two 10 Gbit/s Intel X520 NICs.[2]

[2] The Case for Writing Network Drivers in High-Level Programming Languages, ANCS 2019

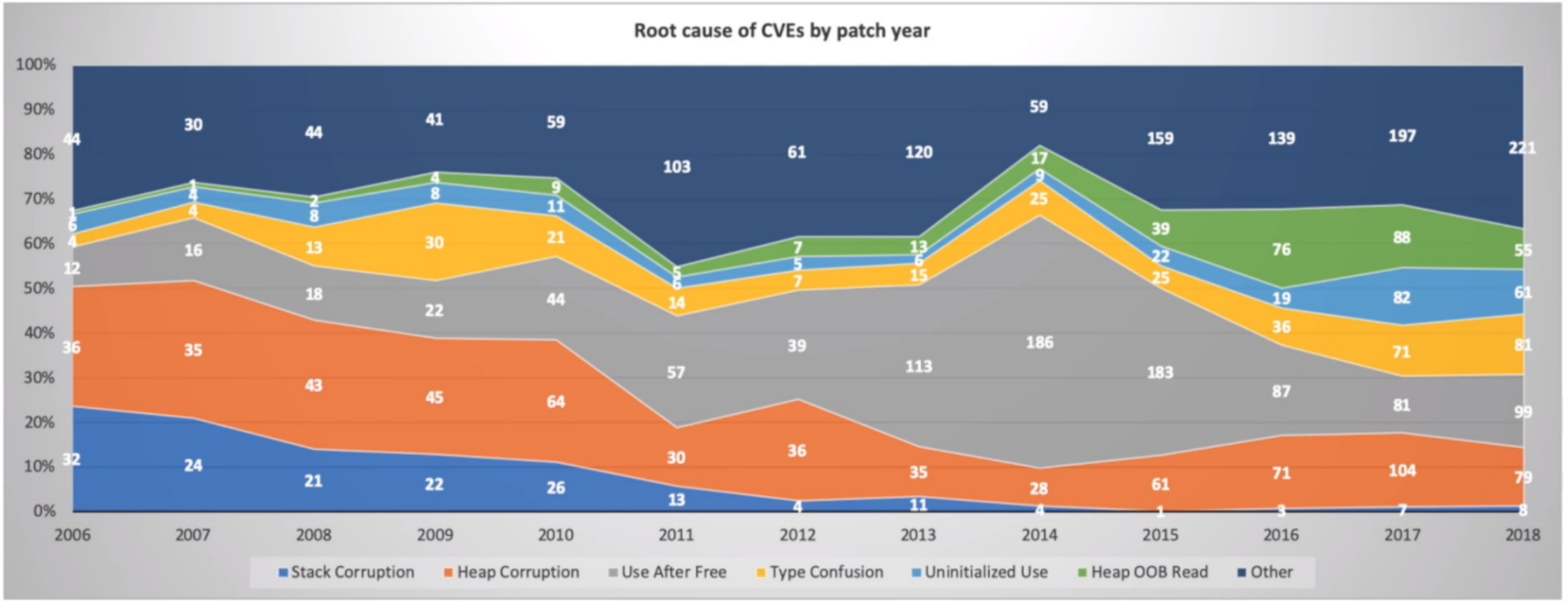
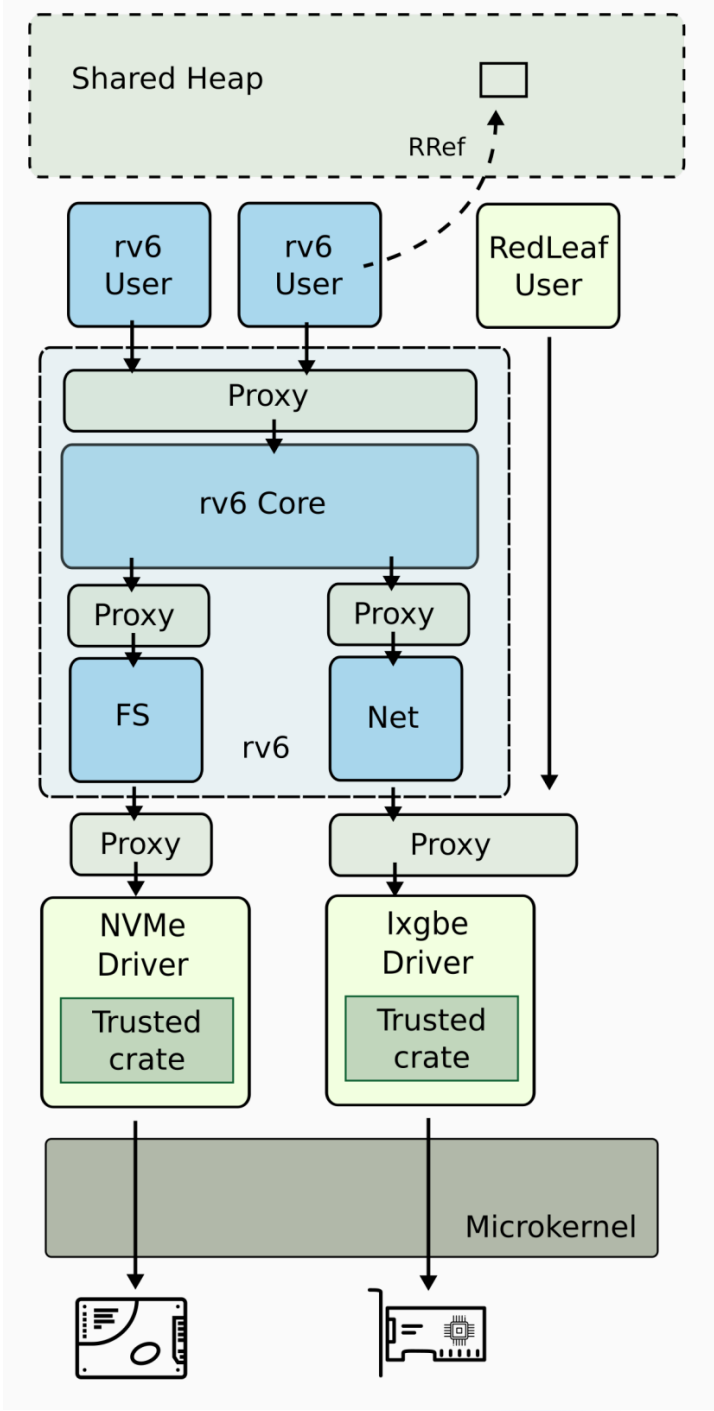


Figure 2: Breakdown of root causes for CVEs by year in Microsoft products.[2]Only 221 out 604 are not safety related.

[2] Digital Security by Design: Security and Legacy at Microsoft. https://vimeo.com/ 376180843, 2019.

- Device drivers
- Rv6, a POSIX-like operating system
  - A collection of domains
  - File system, network stack, and system calls
  - And user processes
- Device pass through
- Shared heap

All code runs in **Ring 0**

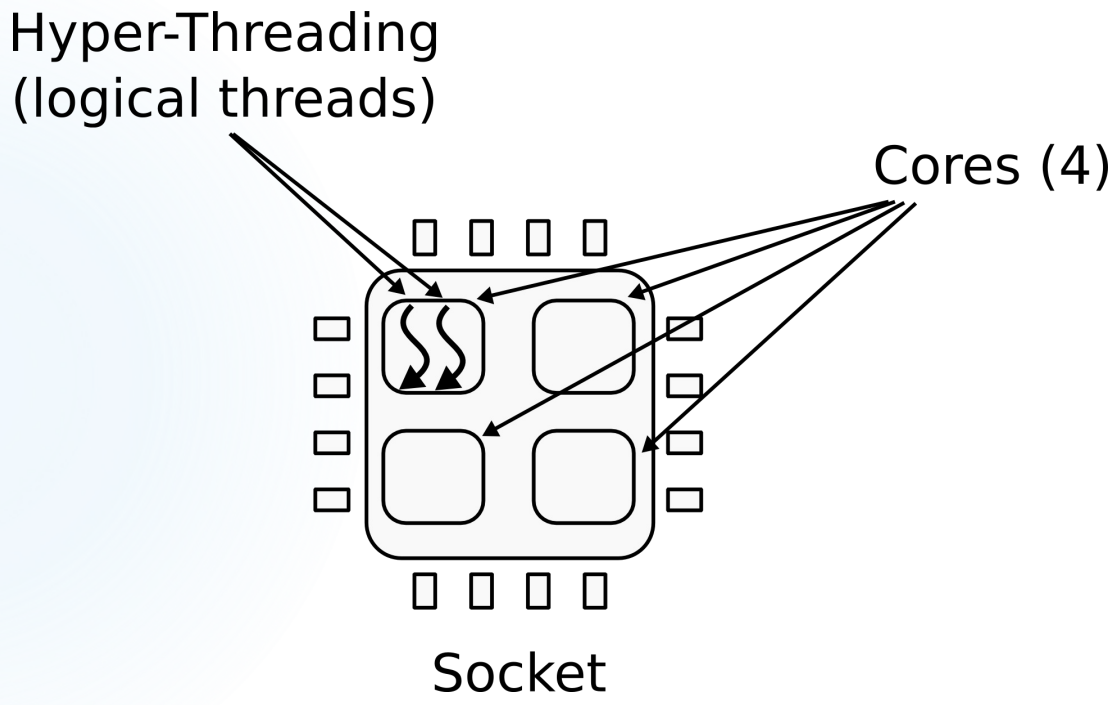


RedLeaf

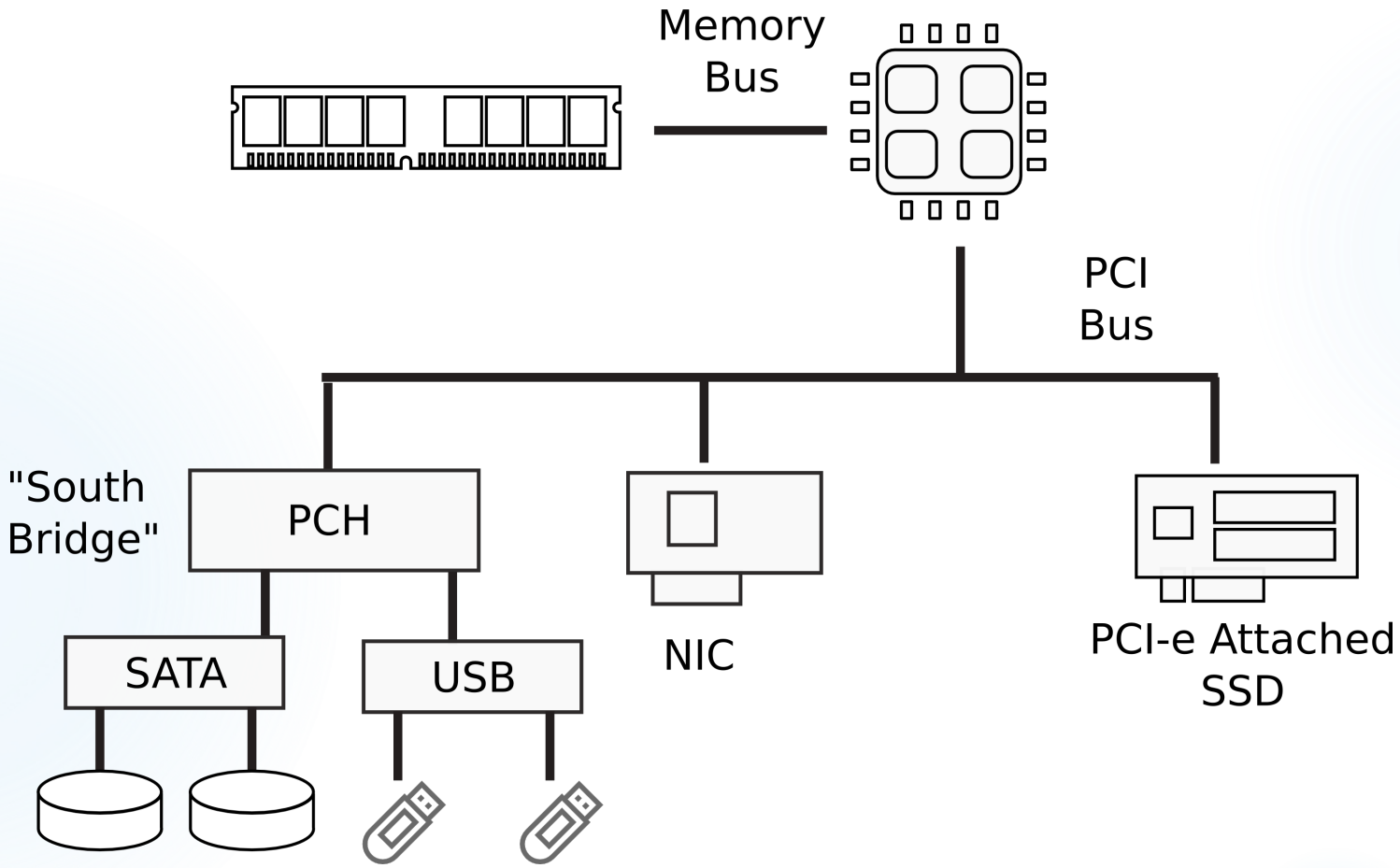
Problem #2: Performance
What does the new hardware look like?

CPU

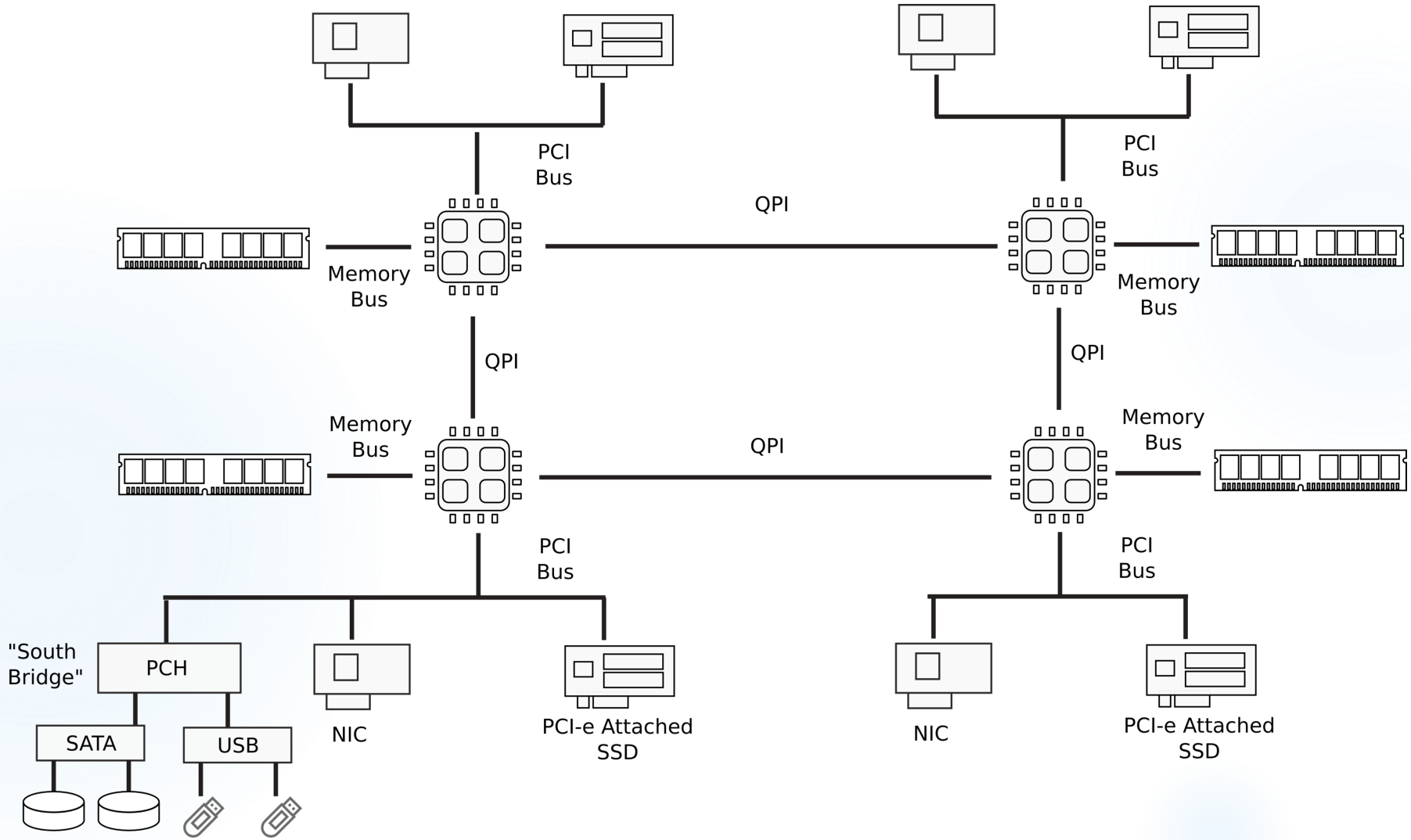
- 1 CPU socket
  - 4 cores
  - 2 logical (HT) threads each



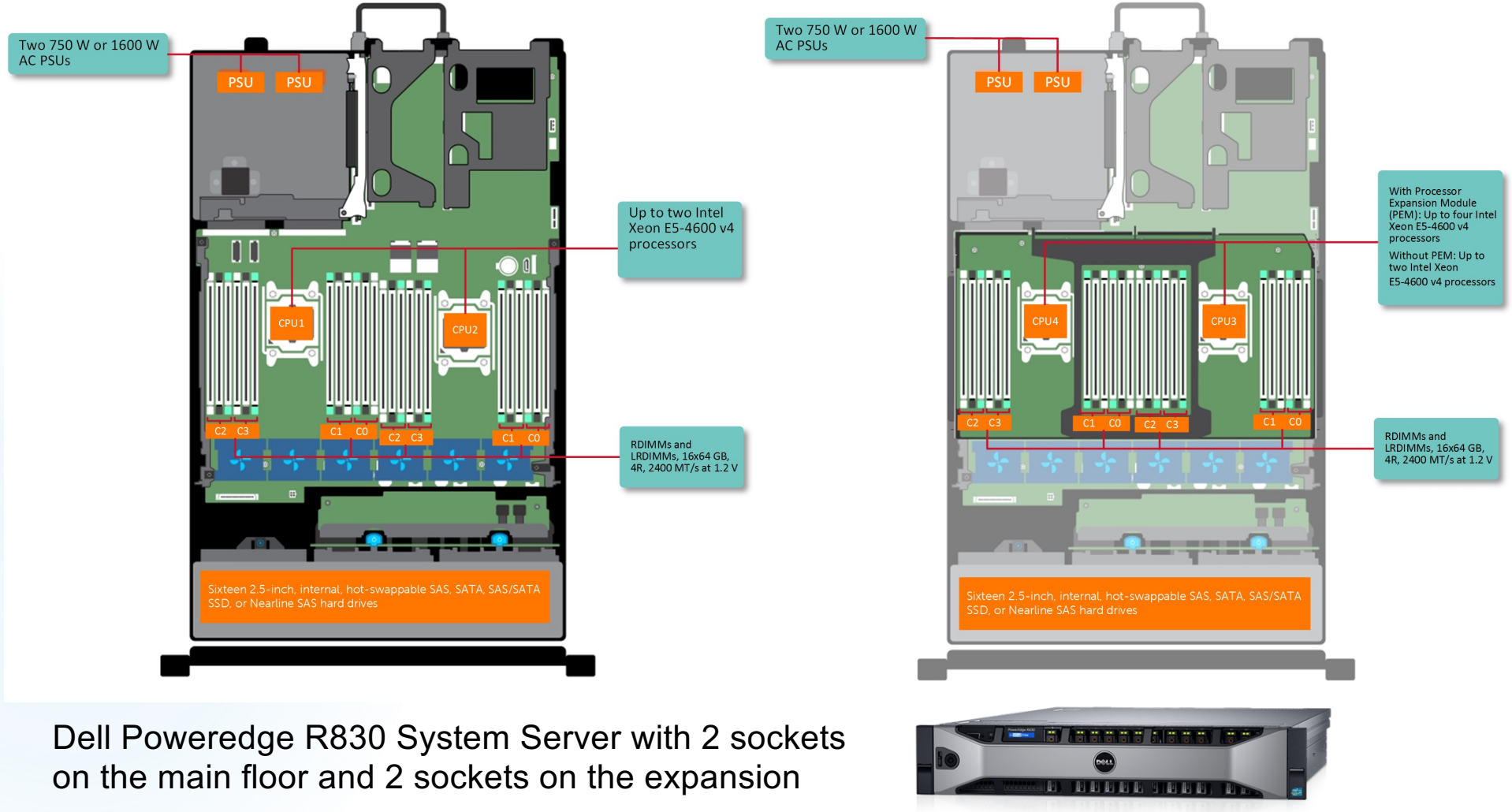
I/O Devices


Memory
Bus
PCI
Bus
"South
Bridge"
PCH
SATA
USB
NIC
PCI-e Attached
SSD

Multi-socket machines

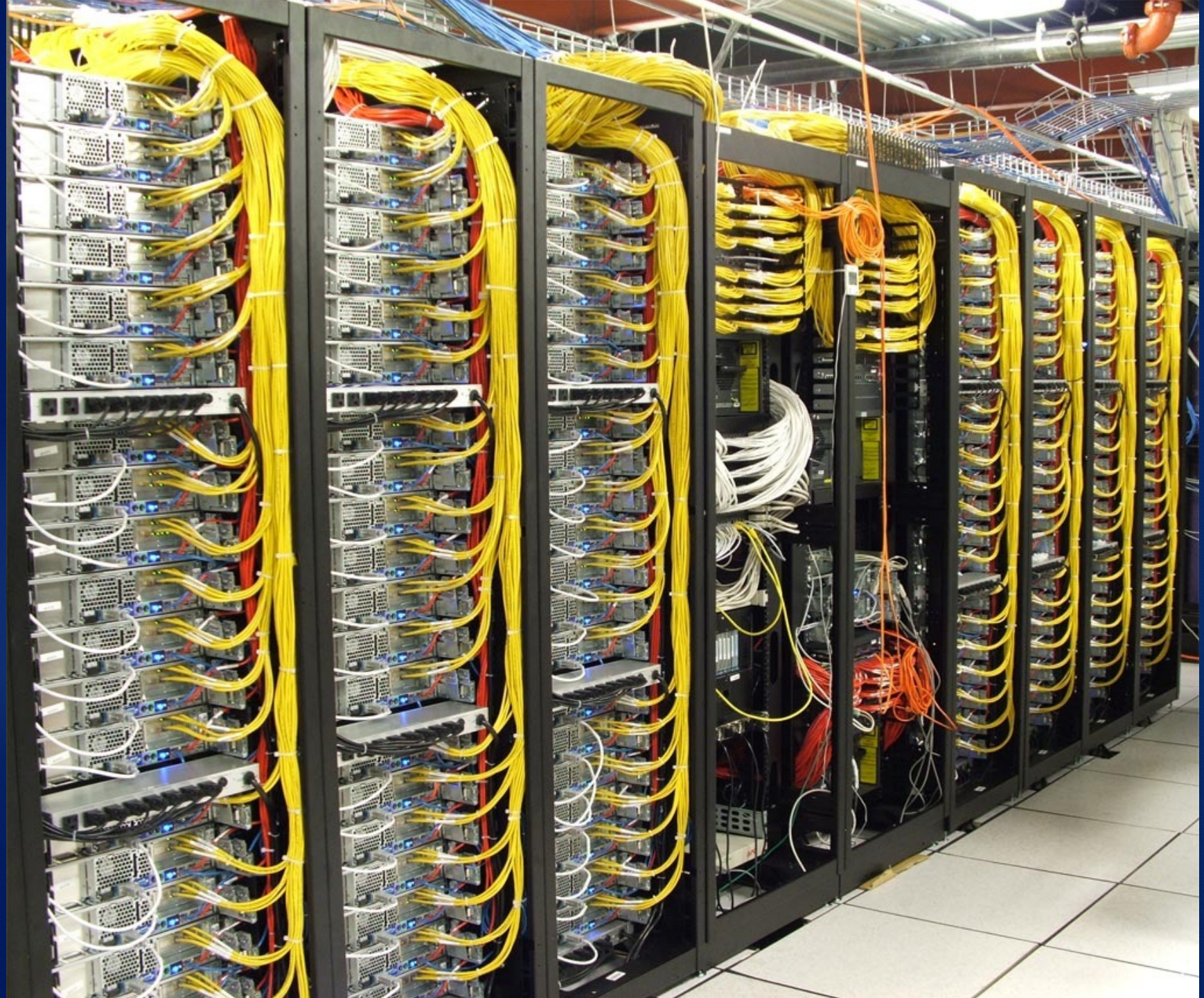


Dell R830 4-socket server



Dell Poweredge R830 System Server with 2 sockets on the main floor and 2 sockets on the expansion

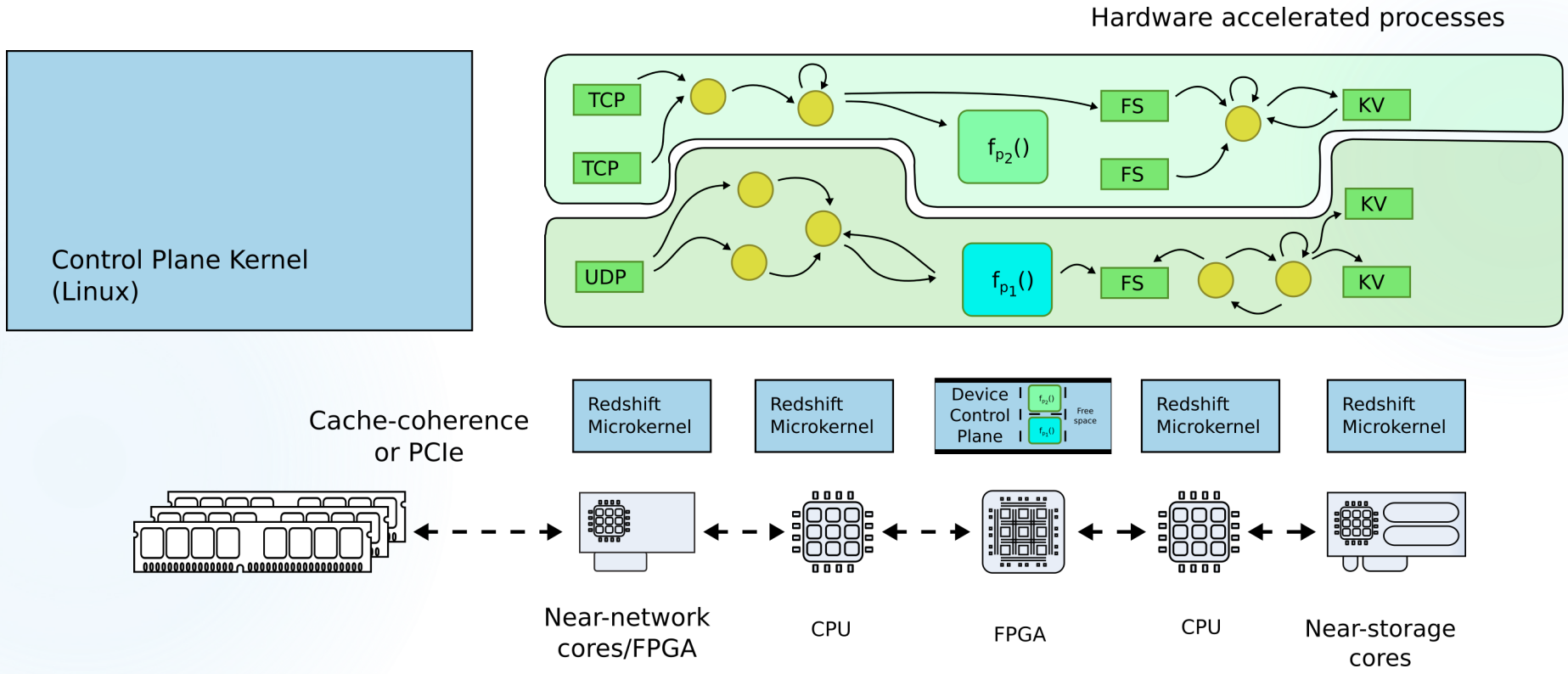
http://www.dell.com/support/manuals/us/en/19/poweredge-r830/r830_om/supported-configurations-for-the-poweredge-r830-system?guid=guid-01303b2b-f884-4435-b4e2-57bec2ce225a&lang=en-us



But what will it look like in 5-10 years?

- Massively heterogeneous
  - Not just many-cores
  - GPUs, AI accelerators, near-storage and near-network cores
- But also
  - Fine-grained hardware ASICs accelerators
  - Programmable hardware (FPGA)

Redshift: Operating system for heterogeneous hardware



- Execution will no longer stay on the CPU

Problem #3: Can we own our data in the cloud?

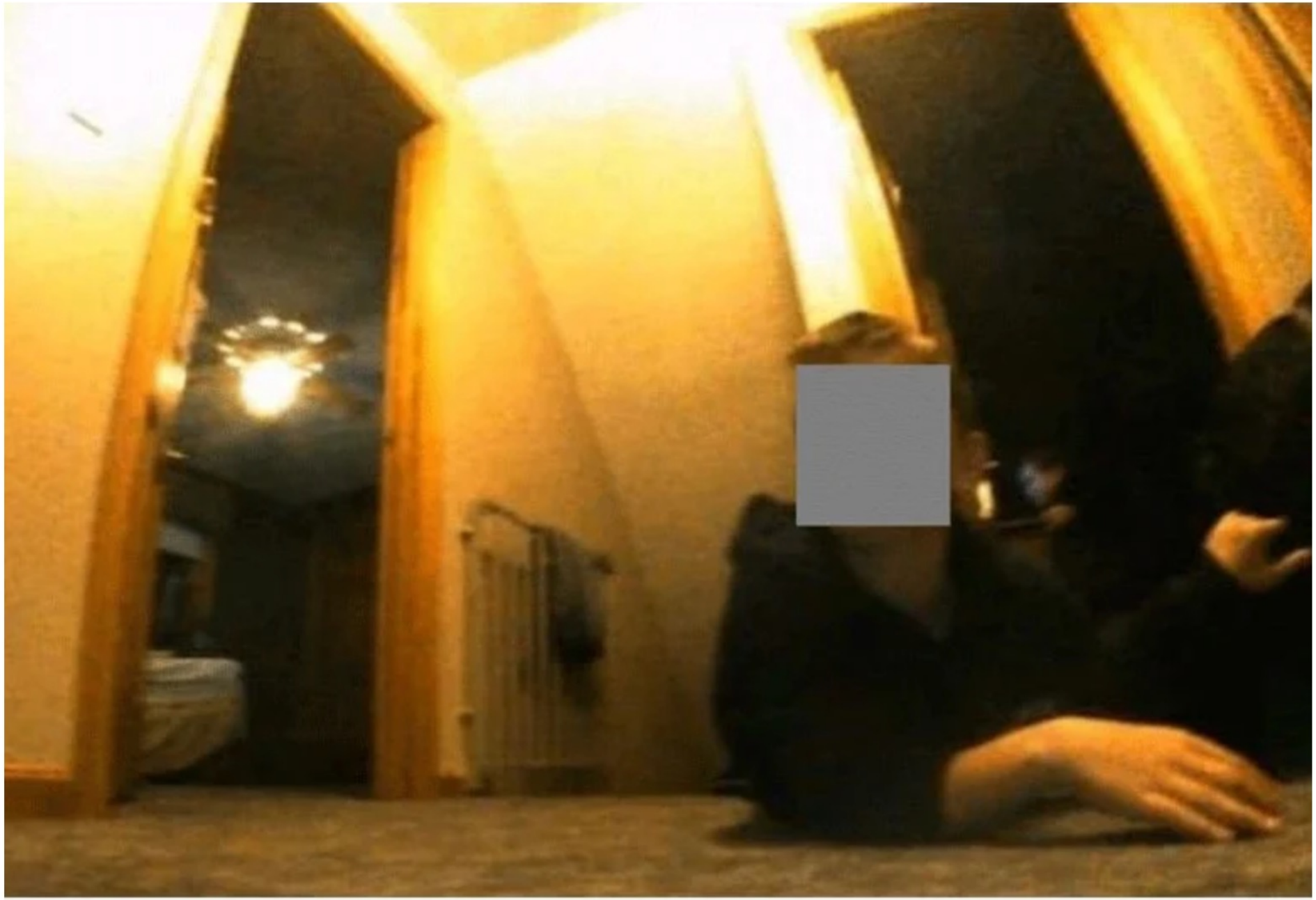
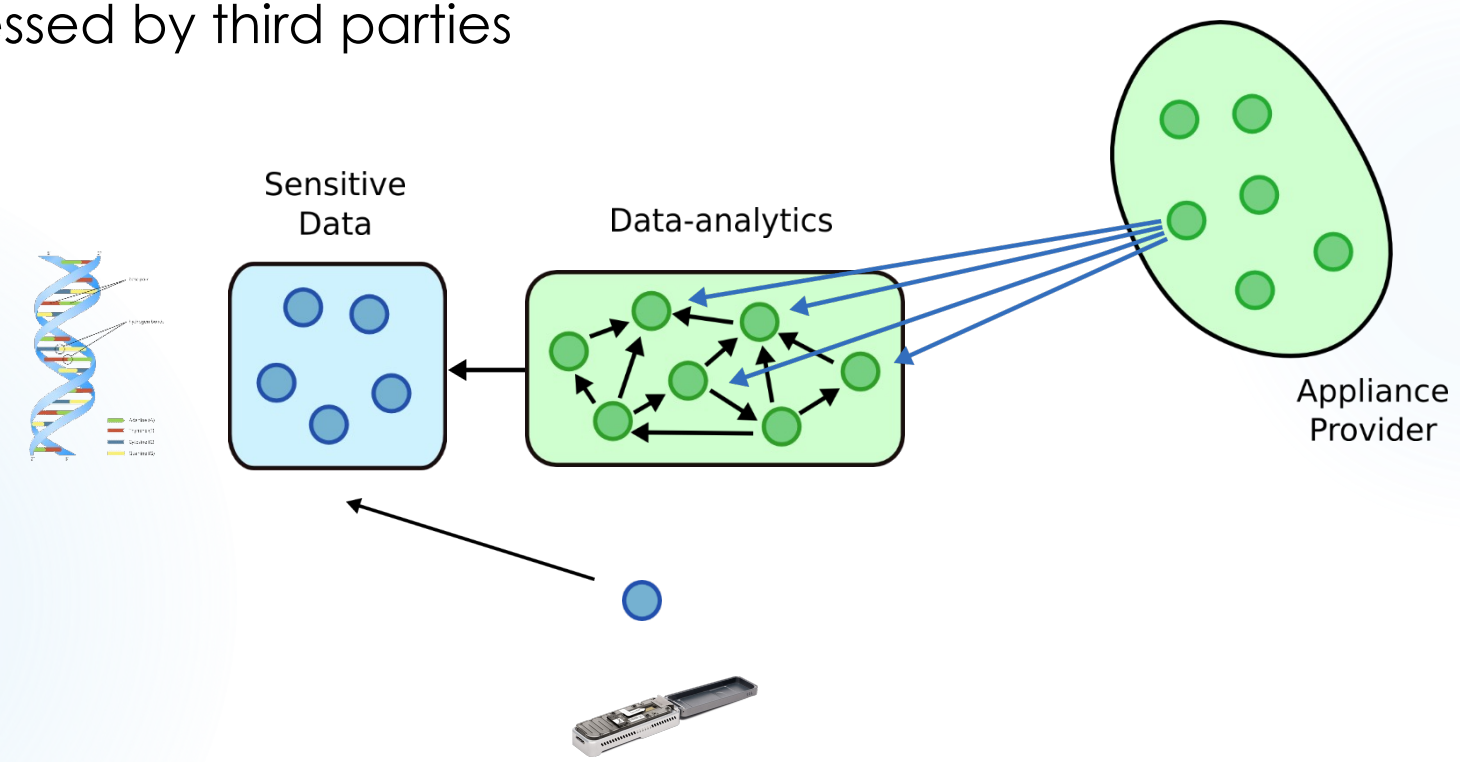


Image captured by iRobot Roomba Vacuum shows young child on the floor

MinION Nanopore DNA Sequencing

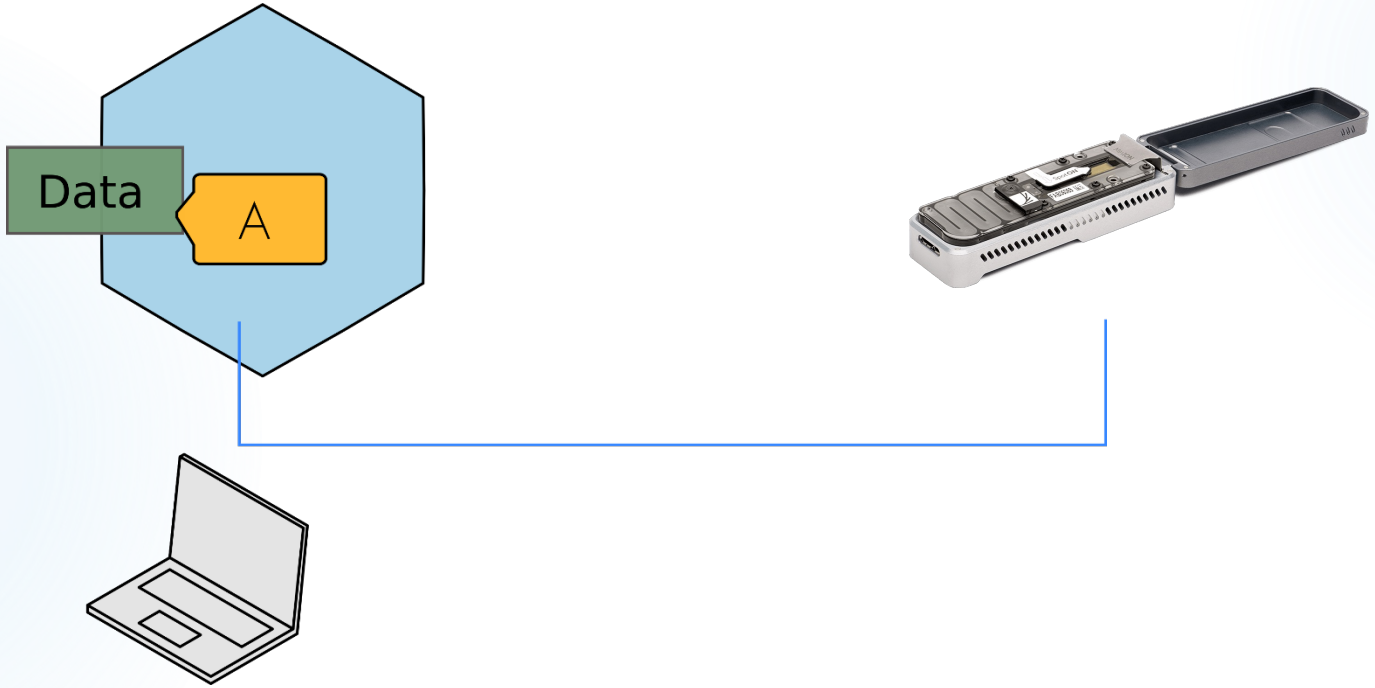
Deeper problem: cloud is inherently collaborative

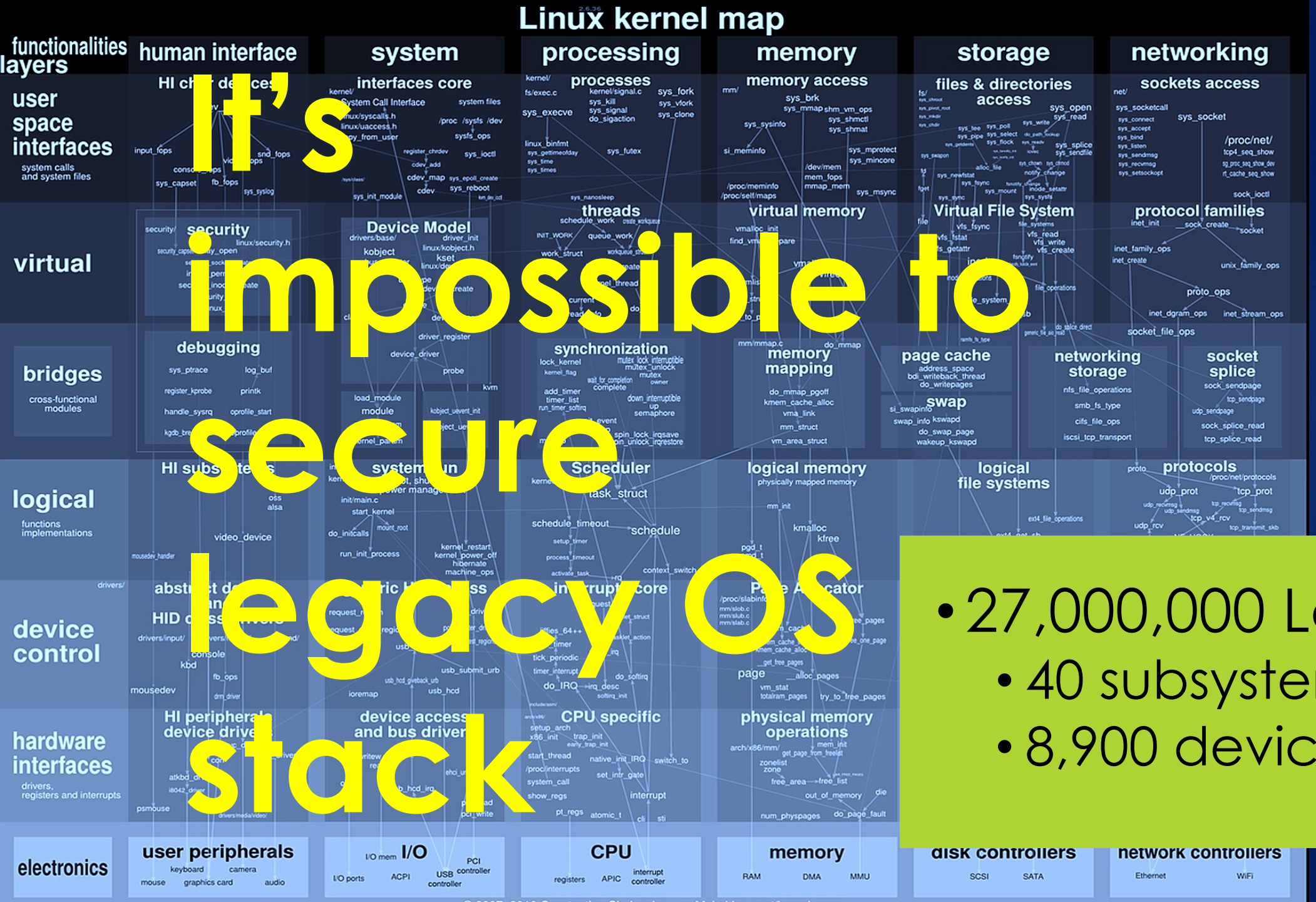
- Data is processed by third parties



What needs to be done?

- Goal #1





It's impossible to secure legacy Os stack

- 27,000,000 LoC
  - 40 subsystems
  - 8,900 device drivers

Thank you!

Anton Burtsev
http://www.cs.utah.edu/~aburtsev
anton.burtsev@utah.edu