COMPUTER ORGANIZATION

Dr. Samantha Foley

CS270

Spring 2018

What is a computer?

A **computer** is a device that can be instructed to carry out arbitrary sequences of arithmetic or logical operations automatically.

- What kinds of computers are there?
- What makes them different?
- Why do they need to be different?

Classes of computers

- Personal computer
- Servers
- Supercomputers
- Embedded computers
- Personal mobile devices
- Cloud computing systems
- Internet of Things

Differ based on how they are used. This usage also drives the size, interfaces, and cost.

Classes of computers: Personal computers

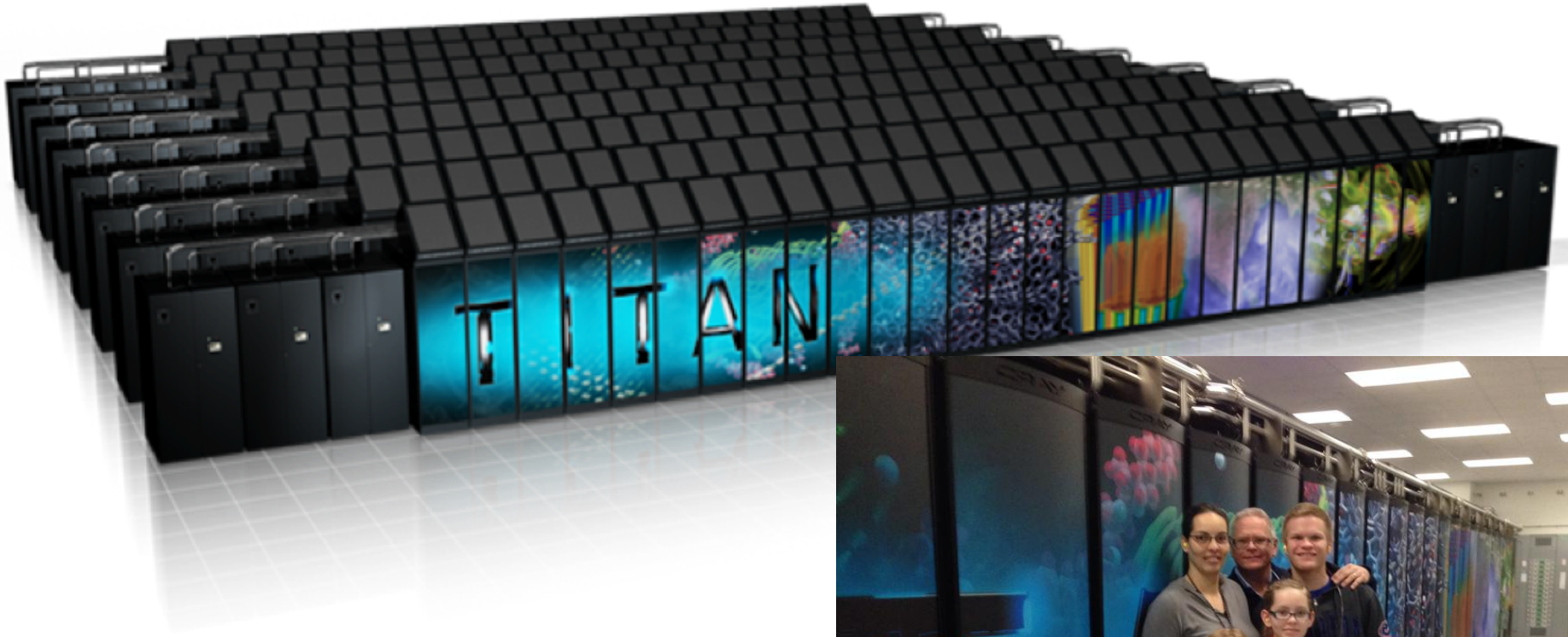
- A **personal computer** is a computer designed for use by an individual, usually incorporating a graphics display, a keyboard, and a mouse.
- Design requirements
  - General purpose computing for a wide range of users
  - Single user, or possibly a small number of users
  - Wide variety of software
  - User interface design and availability of accessories desirable
  - Relatively low cost, but performance is important
- Examples:
  - Desktop, laptop, tablet, netbook

Classes of computers: Servers and supercomputers

- A **server** is a computer used for running larger programs for multiple users, often simultaneously, and typically accessed only via a network.
  - Design requirements:
    - High capacity, performance and reliability
  - Range in size from desktop computers to server farms that take up buildings
- A **supercomputer** is a high-end server designed to achieve the highest performance (often also at the highest cost and with first generation of the hardware and software).
  - Design requirements:
    - High speed and high capacity – sometimes at the expense of reliability
  - Used for high-end scientific and engineering simulation and modeling

Servers

Supercomputer: Titan



Supercomputers: Titan and Summit

TITAN VS SUMMIT

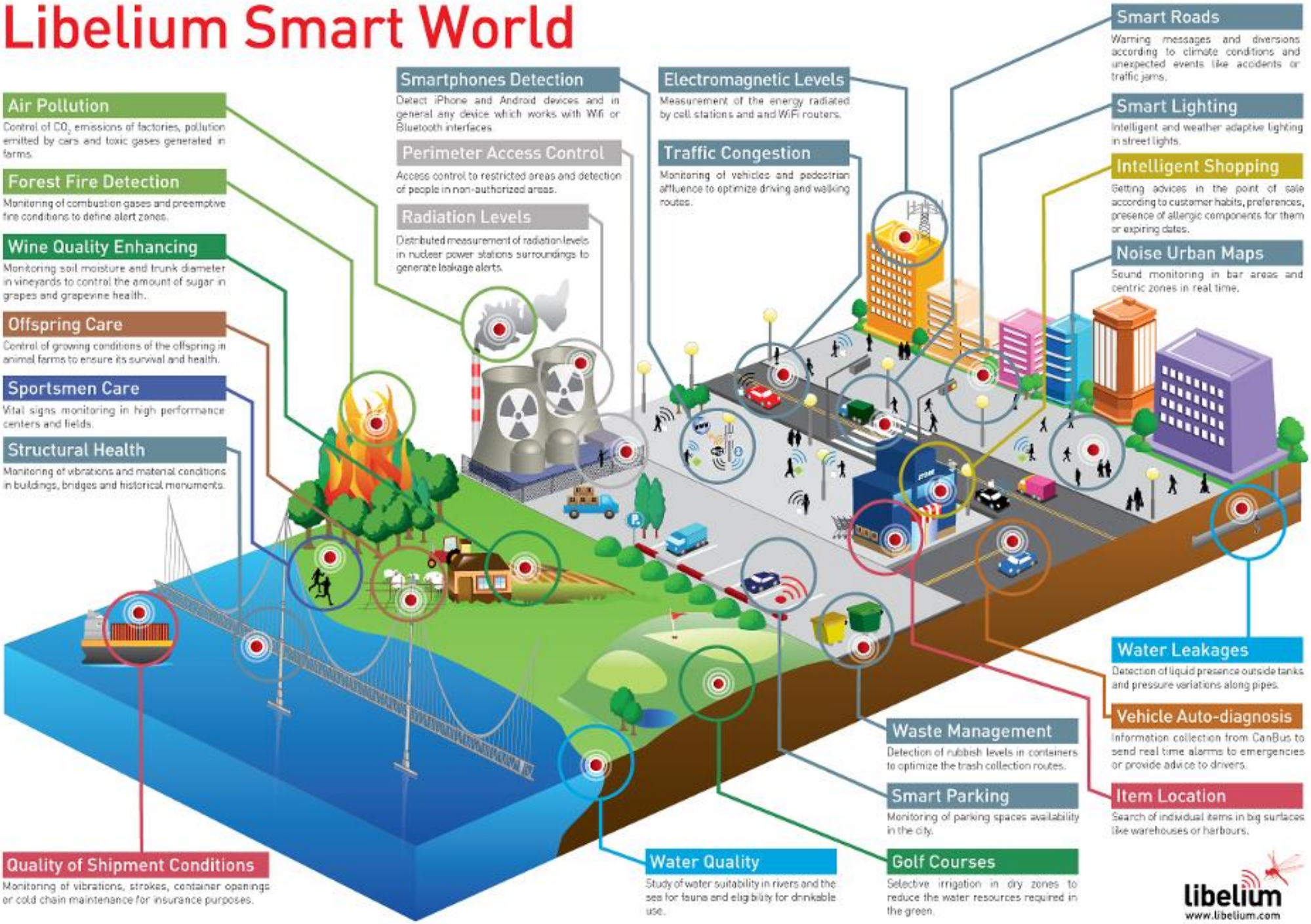
Compute System Comparison

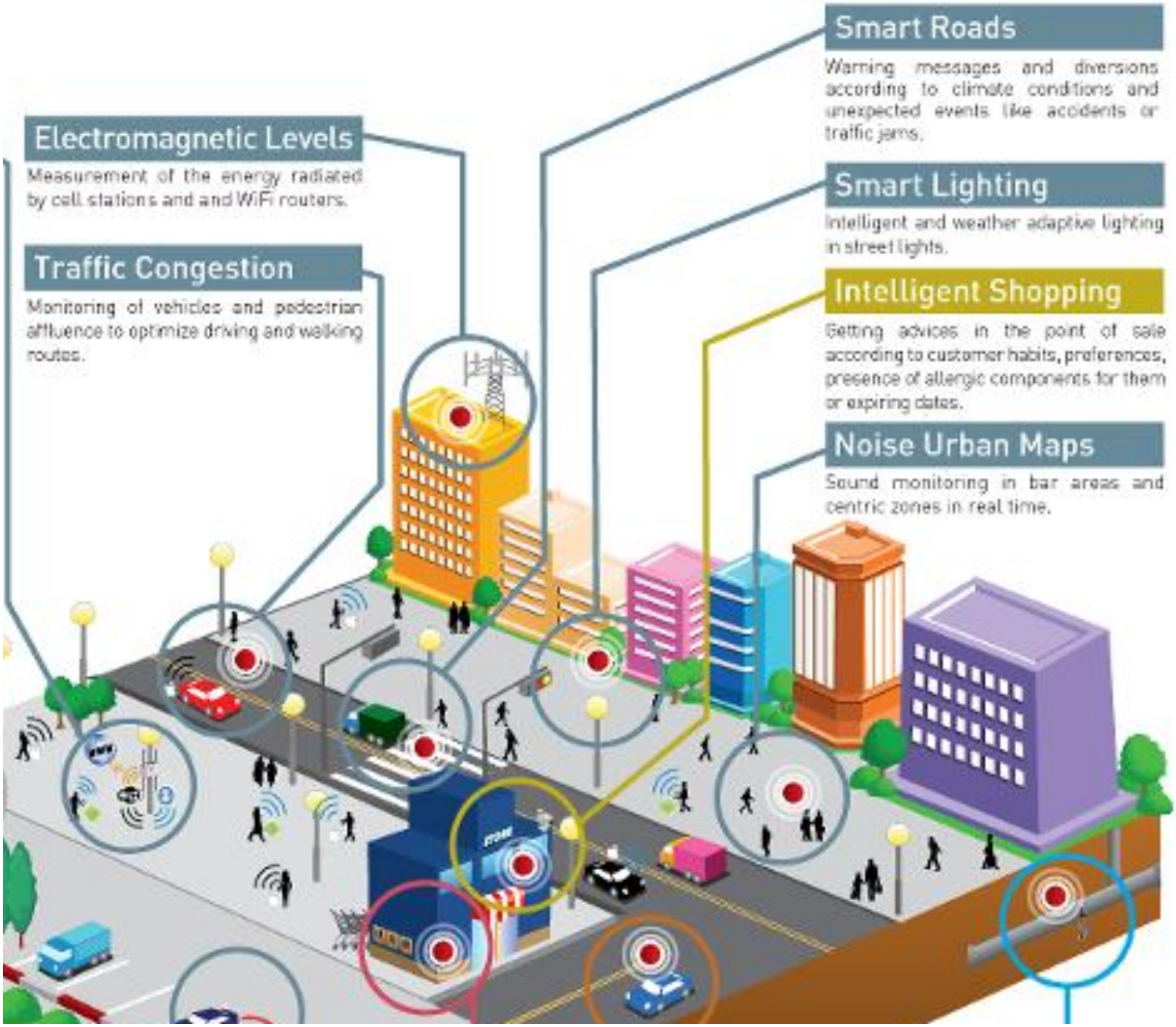
ATTRIBUTE	TITAN	SUMMIT
Compute Nodes	18,688	~3,400
Processor	(1) 16-core AMD Opteron per node	(Multiple) IBM POWER 9s per node
Accelerator	(1) NVIDIA Kepler K20x per node	(Multiple) NVIDIA Volta GPUs per node
Memory per node	32GB (DDR3)	>512GB (HBM+DDR4)
CPU-GPU Interconnect	PCI Gen2	NVLINK (5-12x PCIe3)
System Interconnect	Gemini	Dual Rail EDR-IB (23 GB/s)
Peak Power Consumption	9 MW	10 MW

Each POWER9 has 24 cores, and there are 8 per node for 192 threads of execution.

Classes of computers: Embedded computers and IoT

- An **embedded computer** (embedded system) is a computer that is included inside another device, typically used to run one predetermined application or collection of software.
- Design requirements:
  - Integrated as components of larger systems
  - Tight power, performance, and cost constraints
- Examples:
  - Routers, modems, GPS, cars, televisions, Bluetooth speakers
- The **Internet of Things (IoT)** is the ecosystem of internet enabled embedded computers that can be used to interact with the real world through these embedded systems.
- Design Requirements:
  - Internet enabled, able to communicate information across this interface and interact with a real world device or the environment.
- Examples:
  - Smart locks, smart thermostats, smart outlets, CCTV devices


Libelium Smart World
Air Pollution
Control of CO_2 emissions of factories, pollution emitted by cars and toxic gases generated in farms.
Forest Fire Detection
Monitoring of combustion gases and preemptive fire conditions to define alert zones.
Wine Quality Enhancing
Monitoring soil moisture and trunk diameter in vineyards to control the amount of sugar in grapes and grapevine health.
Offspring Care
Control of growing conditions of the offspring in animal farms to ensure its survival and health.
Sportsmen Care
Vital signs monitoring in high performance centers and fields.
Structural Health
Monitoring of vibrations and material conditions in buildings, bridges and historical monuments.
Quality of Shipment Conditions
Monitoring of vibrations, strokes, container openings or cold chain maintenance for insurance purposes.
Smartphones Detection
Detect iPhone and Android devices and in general any device which works with Wifi or Bluetooth interfaces.
Perimeter Access Control
Access control to restricted areas and detection of people in non-authorized areas.
Radiation Levels
Distributed measurement of radiation levels in nuclear power stations surroundings to generate leakage alerts.
Electromagnetic Levels
Measurement of the energy radiated by cell stations and and WiFi routers.
Traffic Congestion
Monitoring of vehicles and pedestrian affluence to optimize driving and walking routes.
Water Quality
Study of water suitability in rivers and the sea for fauna and eligibility for drinkable use.
Smart Roads
Warning messages and diversions according to climate conditions and unexpected events like accidents or traffic jams.
Smart Lighting
Intelligent and weather adaptive lighting in street lights.
Intelligent Shopping
Getting advices in the point of sale according to customer habits, preferences, presence of allergic components for them or expiring dates.
Noise Urban Maps
Sound monitoring in bar areas and centric zones in real time.
Water Leakages
Detection of liquid presence outside tanks and pressure variations along pipes.
Vehicle Auto-diagnosis
Information collection from CanBus to send real time alarms to emergencies or provide advice to drivers.
Item Location
Search of individual items in big surfaces like warehouses or harbours.
Waste Management
Detection of rubbish levels in containers to optimize the trash collection routes.
Smart Parking
Monitoring of parking spaces availability in the city.
Golf Courses
Selective irrigation in dry zones to reduce the water resources required in the green.
libelium
www.libelium.com


Smart Roads
Warning messages and diversions according to climate conditions and unexpected events like accidents or traffic jams.
Smart Lighting
Intelligent and weather adaptive lighting in street lights.
Intelligent Shopping
Getting advices in the point of sale according to customer habits, preferences, presence of allergic components for them or expiring dates.
Noise Urban Maps
Sound monitoring in bar areas and centric zones in real time.
Electromagnetic Levels
Measurement of the energy radiated by cell stations and and WiFi routers.
Traffic Congestion
Monitoring of vehicles and pedestrian affluence to optimize driving and walking routes.

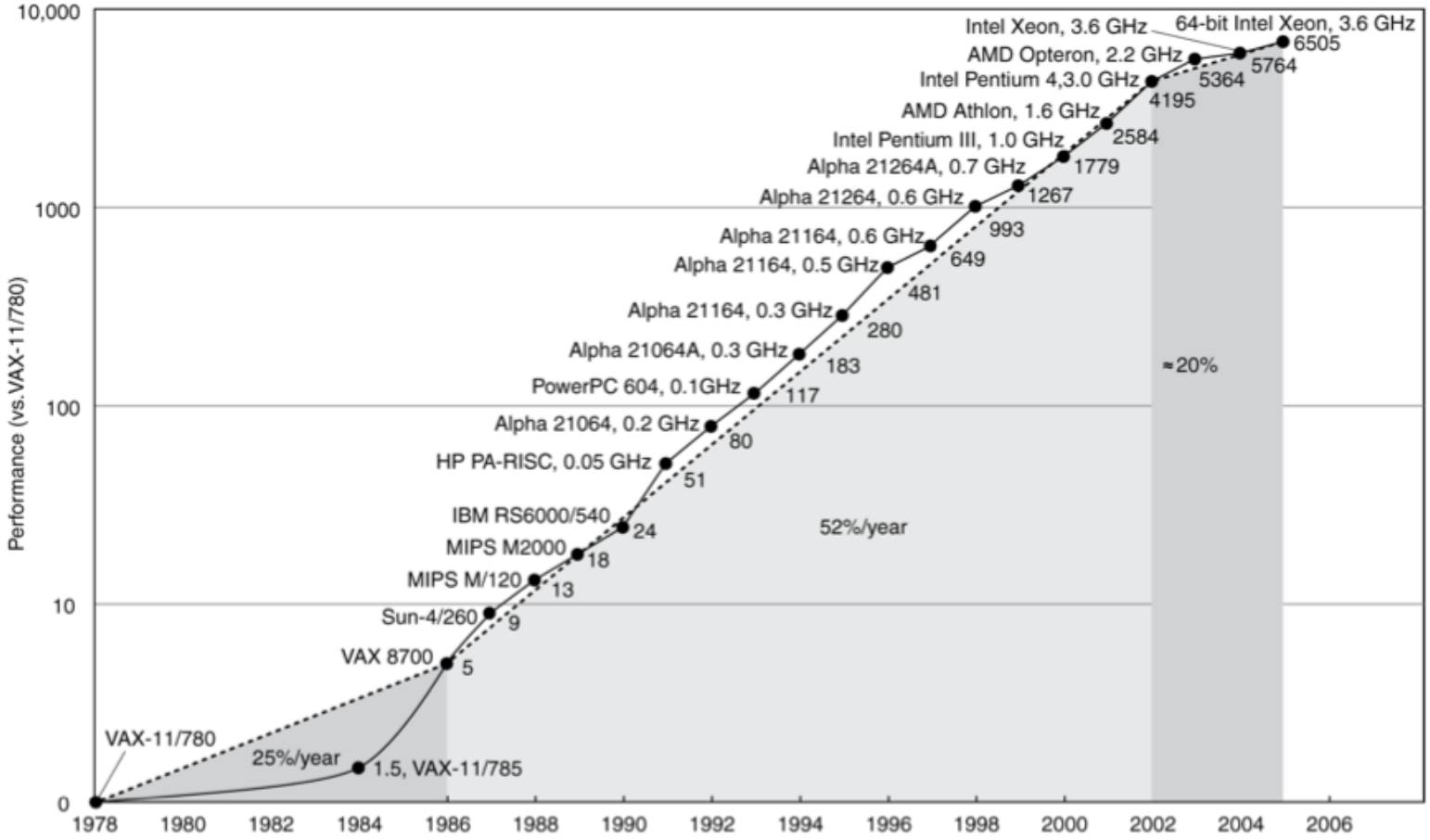
Classes of computers: Personal mobile devices and cloud computing systems

- A **personal mobile device** is a personal computing system that has a smaller form factor than a traditional personal computer but generally serves the same purpose as the larger device.
- Design requirements:
  - Small form factor, intuitive and compact user interface
  - Convenient security features, good battery life
  - Relatively cheap but also good performance
- Examples: smartphone, tablet, smart watch
- **Cloud computing** is a computing model in which processing power and computer resources are provided (on-demand) across the network, often in the form of services.
- This has largely enabled the rise of personal mobile devices since much of the data and computation does not need to happen on the device itself. It happens on the service provider or a third-party's systems.

How did we get here?

- The field of computing is constantly evolving and changing due to rapid progress in computer technology, best represented by Moore's Law:
- ***The complexity for minimum component costs has increased at a rate of roughly a factor of two per year. Certainly over the short term this rate can be expected to continue, if not to increase. Over the longer term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years. — Gordon Moore, 1965***
- Often paraphrased as "computing speed doubling every 18 months"

Uniprocessor Performance



- Basically follows Moore's law until the early 2000s. Power, instruction-level parallelism and memory latency reduced the "free" speedup from smaller transistors

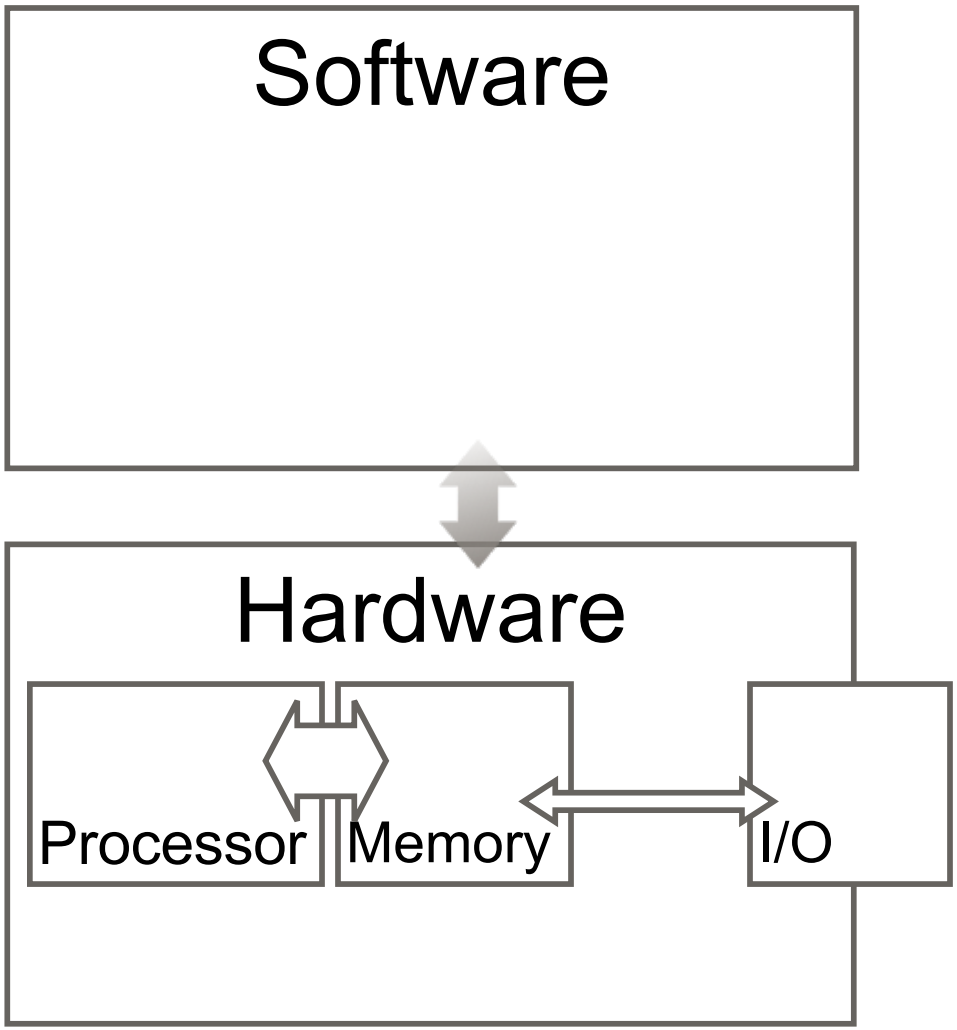
Smaller, faster processors

- Smaller, faster processors has led to an explosion in the number of available processors around the world, and thus computers. It has fueled the revolution of small devices for everyone to carry around and for computers to be embedded into all sorts of other things.
- Additionally, the internet allowed us to connect these devices so that they can all talk to each other.

- These two ideas have lead to the multitude of connected devices and the ability to off-load much of the computing to the cloud.

IF THESE ARE ALL COMPUTERS, WHAT ARE THE COMPONENTS OF A COMPUTER?

Elements of a Computer



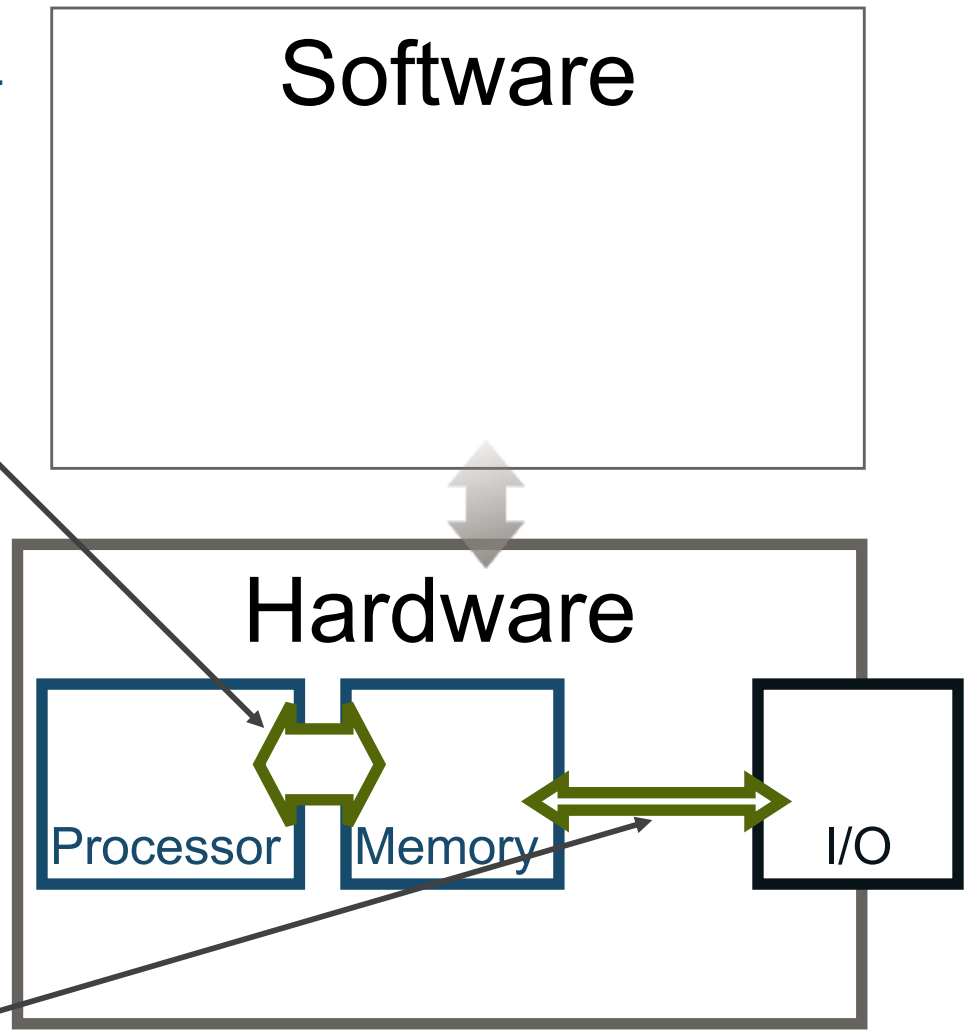
Elements of a Computer: Hardware

Processor (FASTEST):
Executes instructions to perform actions on data.

Fast connection between processor and memory

Memory (FAST):
Stores the data for currently programs.

Slower connection to I/O devices because they can be slow and accessed less frequently



I/O (SLOW):
Devices that communicate data to and/or from the computer system.

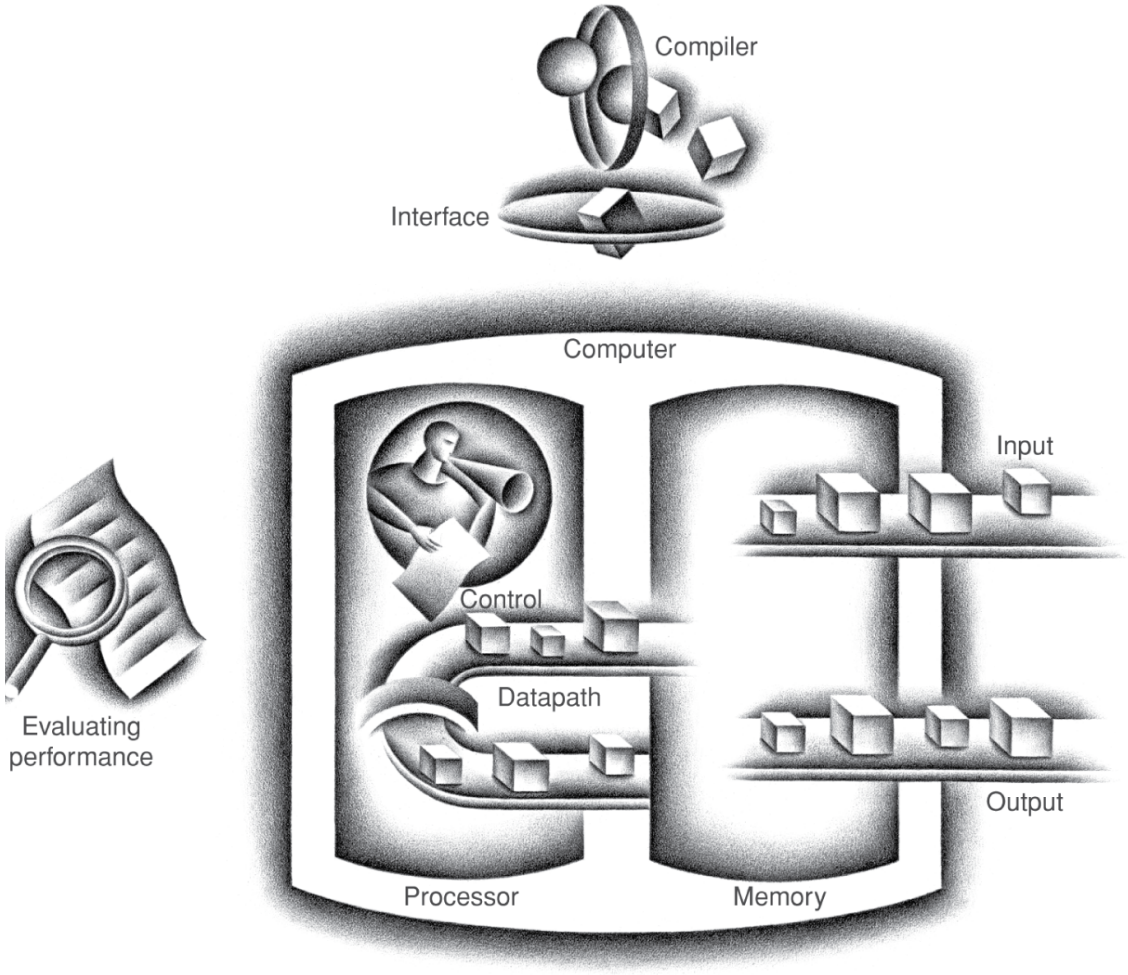
Storage:
Stores all data for the computer system.

User-interface:
Devices that are designed to communicate data to or from humans.

Network:
Devices that are designed to communicate data to and from other computers or devices.

Components of a Computer

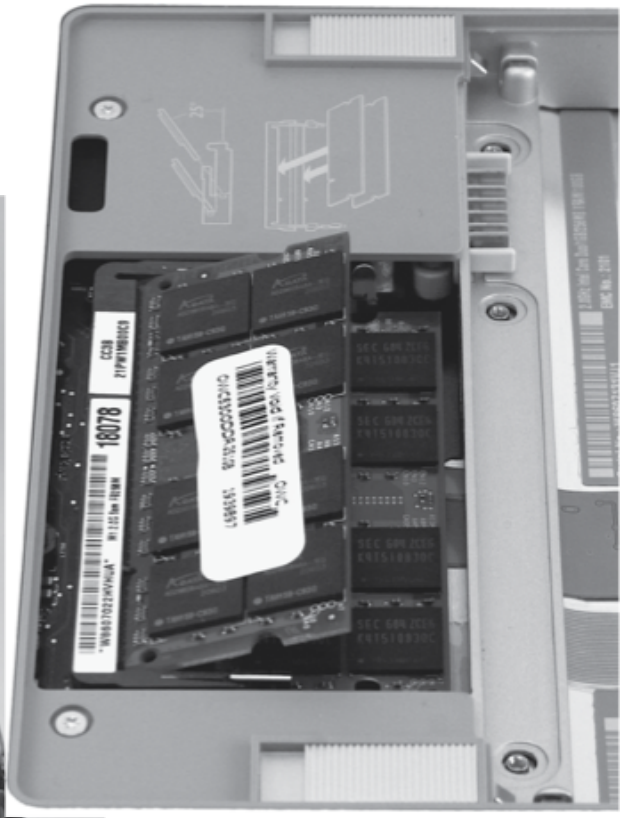
The BIG Picture

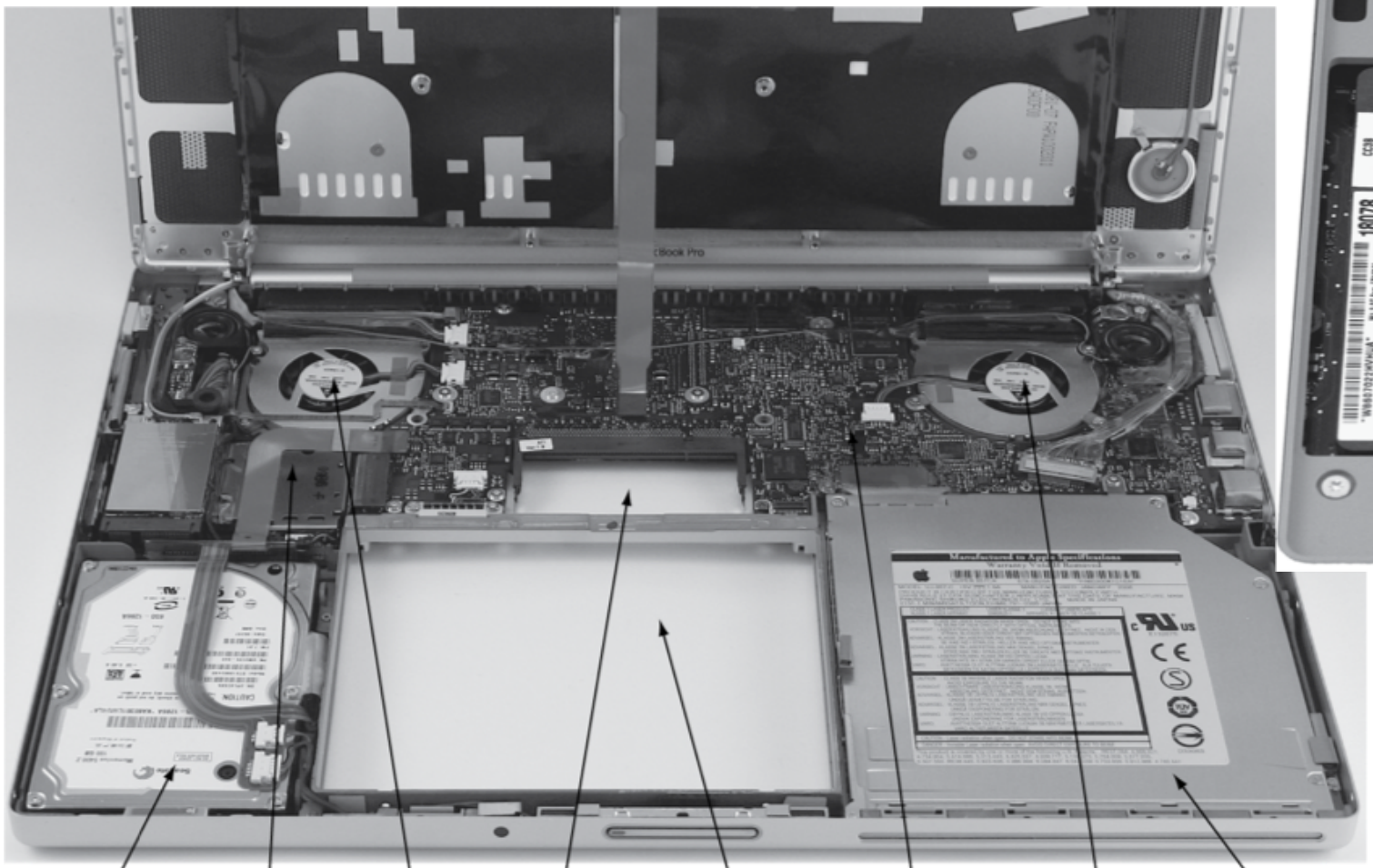


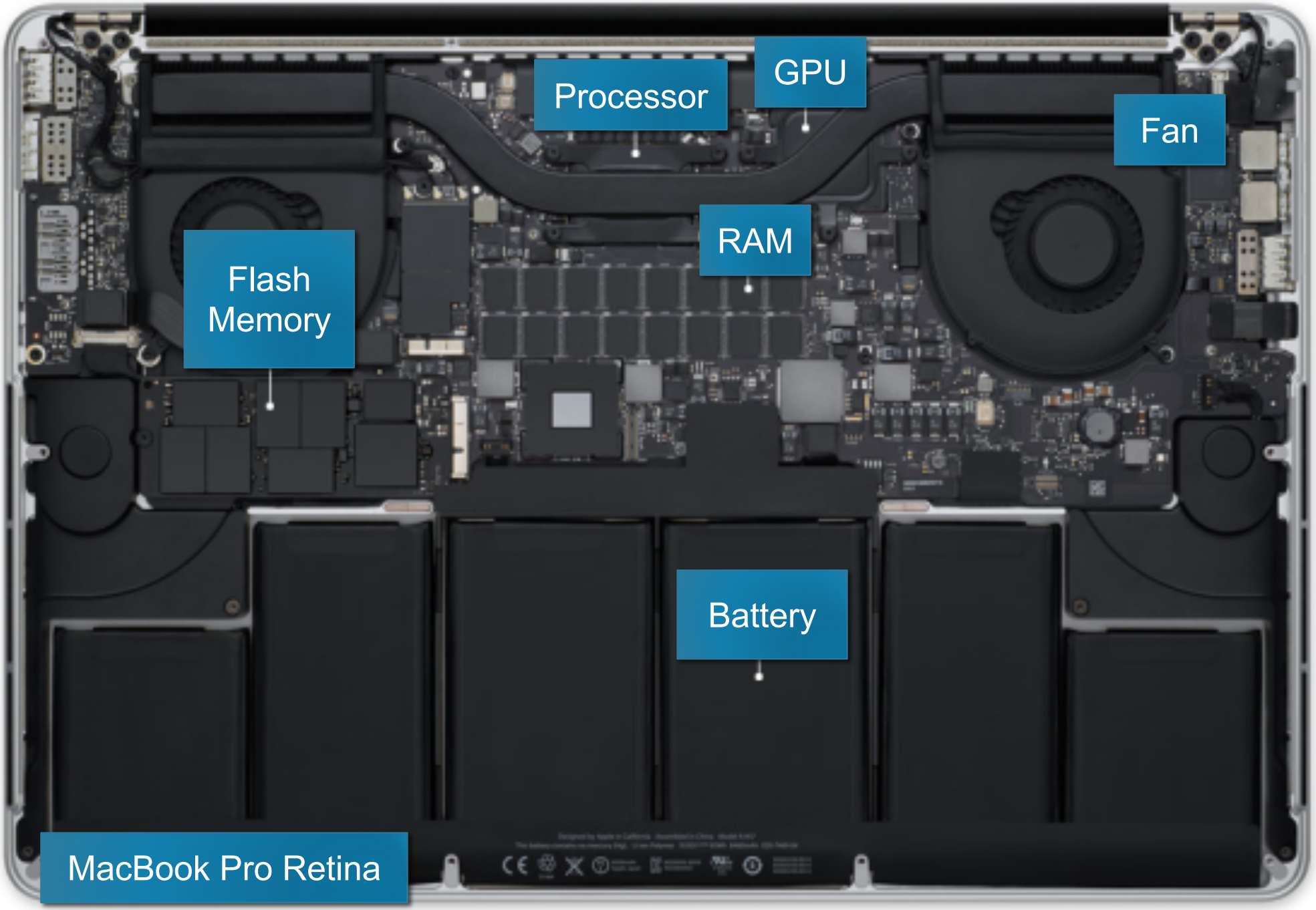
- Same components for all kinds of computer
  - Desktop, server, embedded
- Input/output includes
  - User-interface devices
    - Display, keyboard, mouse
  - Storage devices
    - Hard disk, CD/DVD, flash
  - Network adapters
    - For communicating with other computers

Opening the Box

DRAM (main memory)

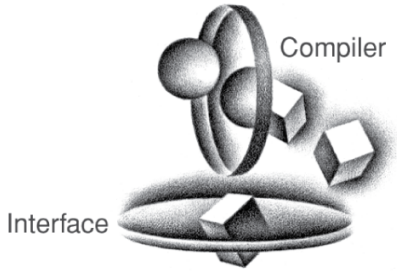


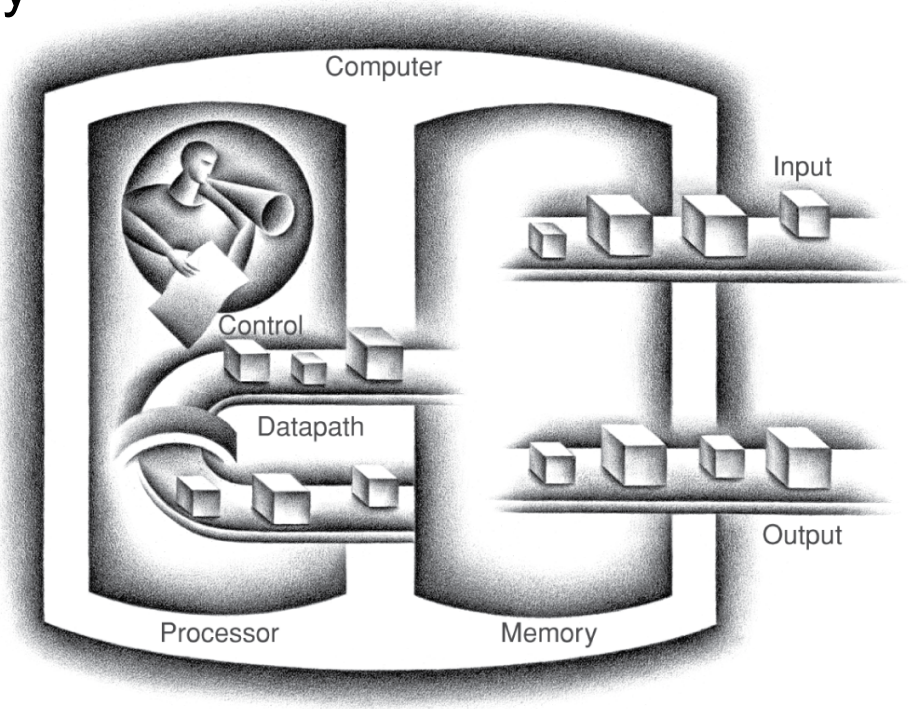



Processor
GPU
Fan
RAM
Flash Memory
Battery
MacBook Pro Retina

Inside the Processor (CPU)

- Datapath: performs operations on data
- Control: sequences datapath, memory, ...
- Cache memory
  - Small fast bank(s) of SRAM memory

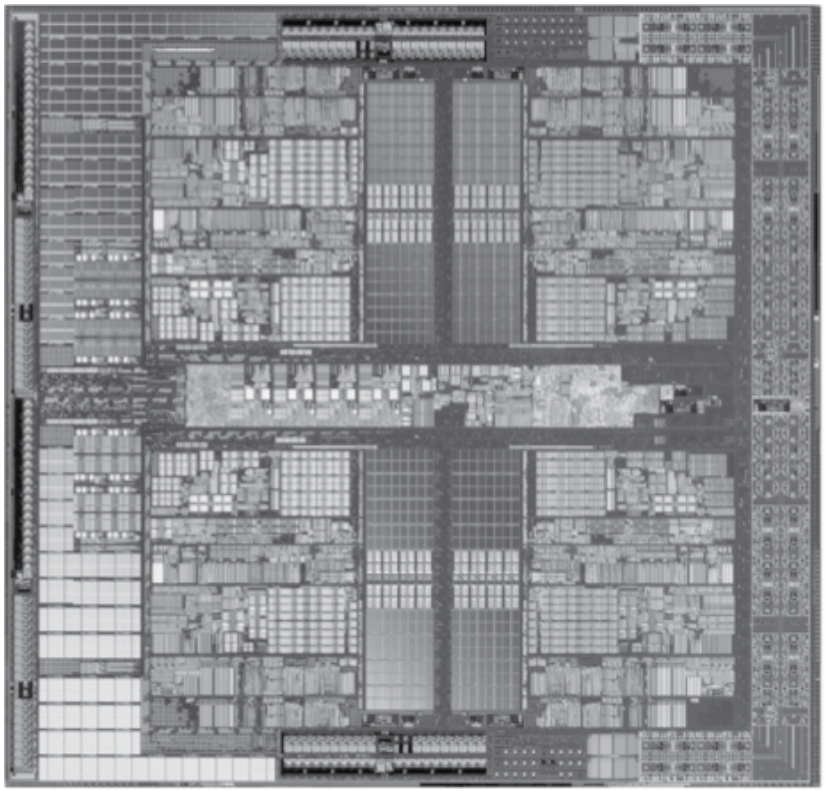


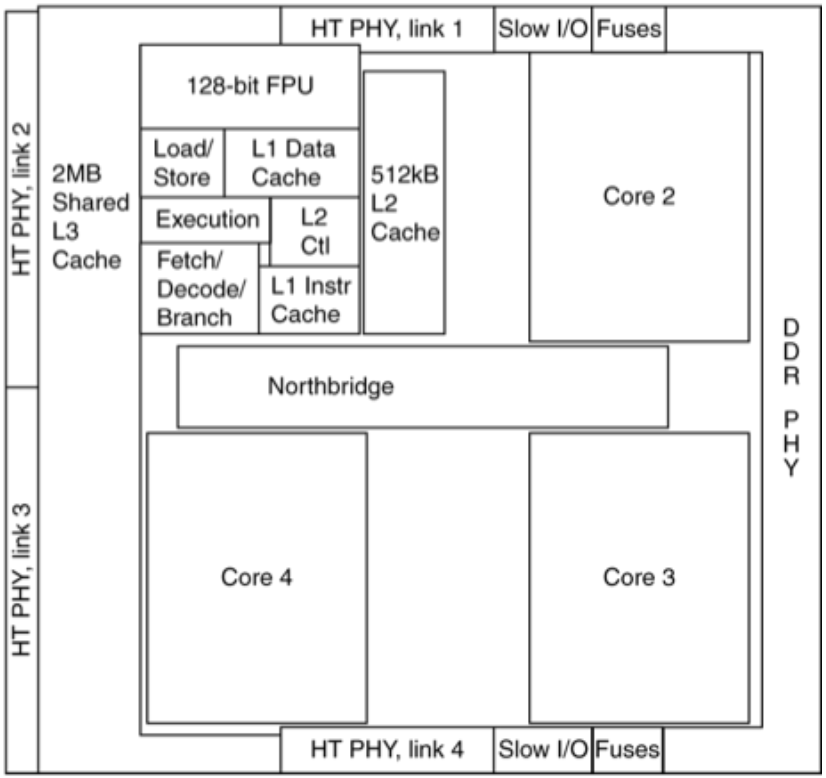


Performs operations – the faster the processor the more operations it can perform per second → faster computer

Inside the Processor

- AMD Barcelona: 4 processor cores

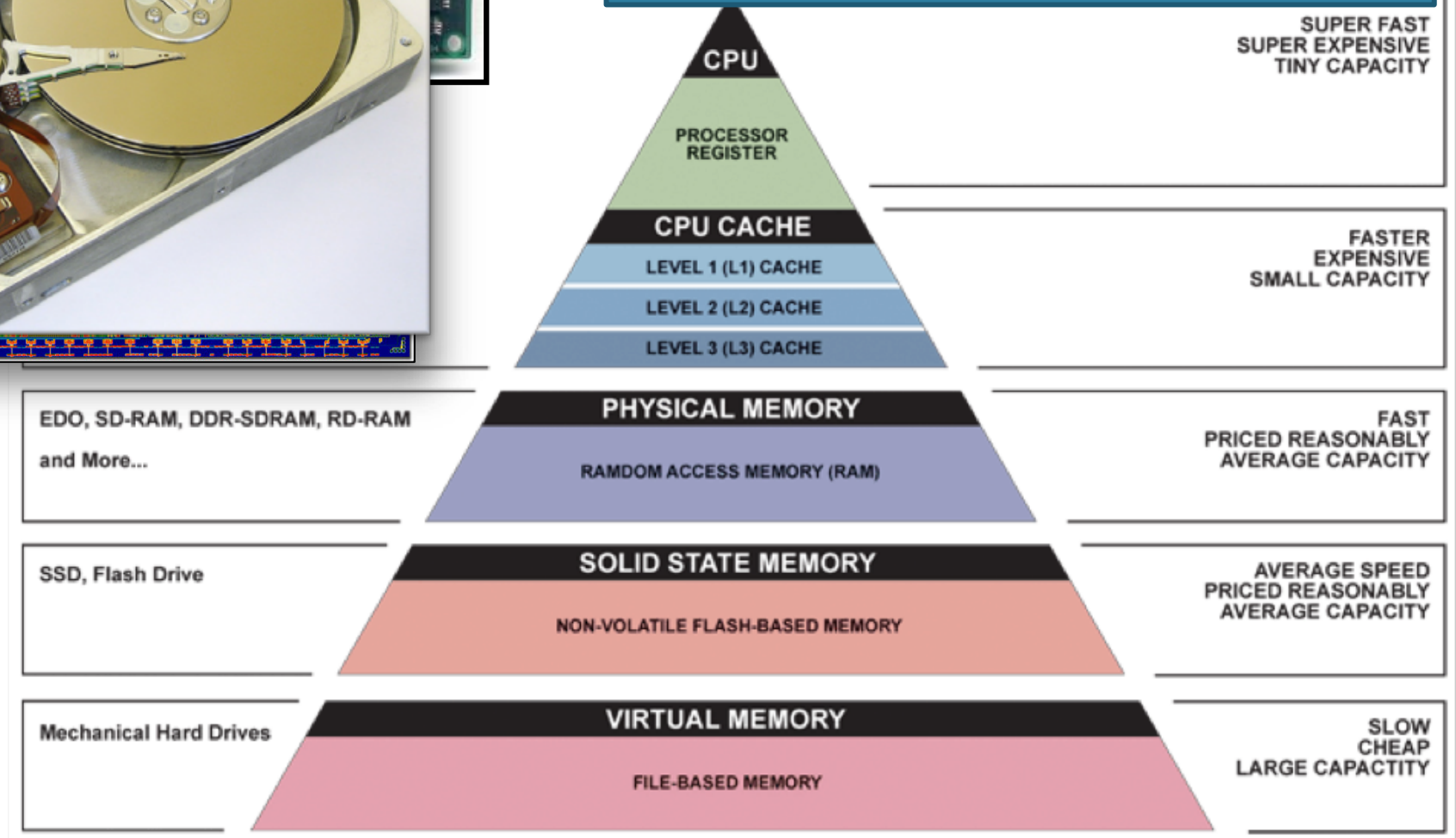




Multiple cores allow multiple instructions to execute at the same time

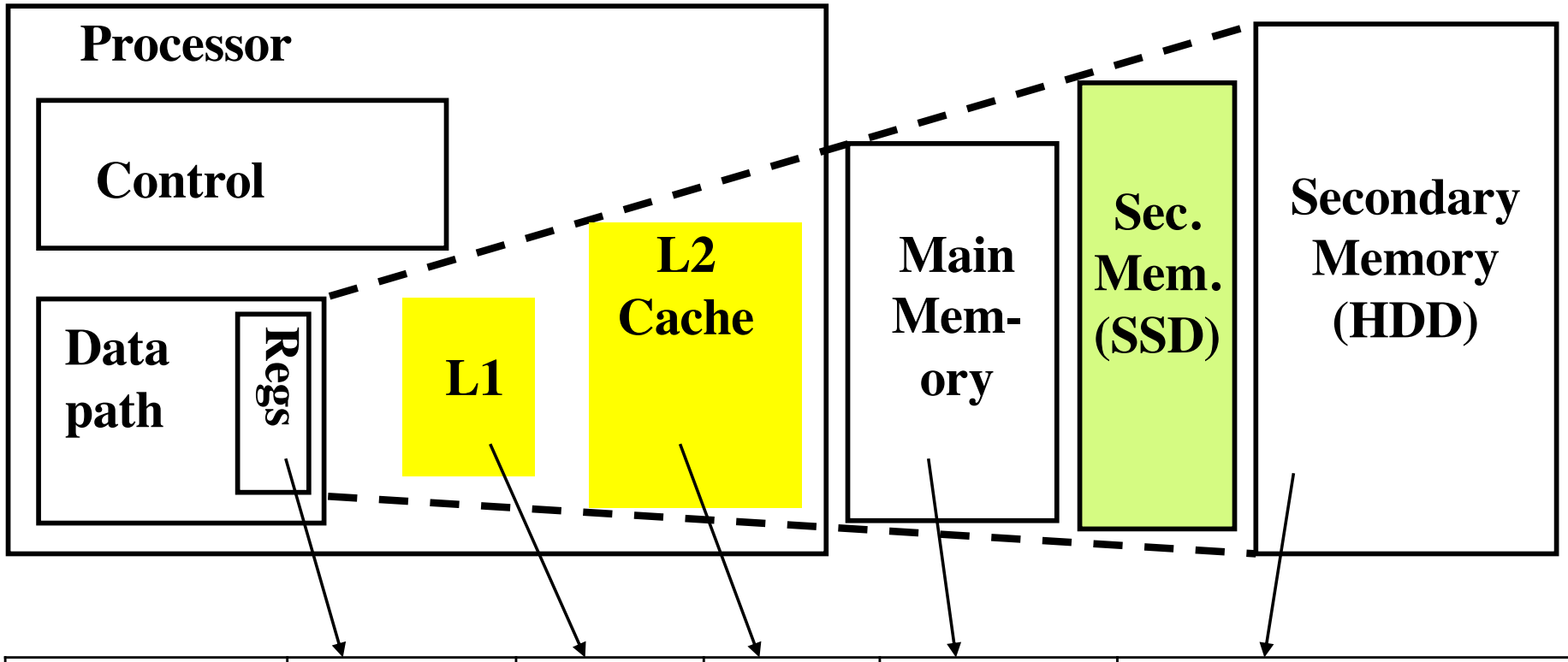
Memory Hierarchy

Large amounts of fast memory increase the perceived speed of a computer. Large amounts of slow memory increase the capacity of the machine to work on large datasets.



▲ Simplified Computer Memory Hierarchy
Illustration: Ryan J. Leng

Computer System: Memory Hierarchy



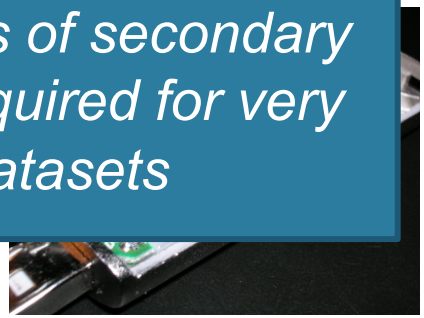
Speed (ns)	0.5ns	2ns	6ns	100ns	10,000,000ns
Size (MB)	0.0005	0.05	1-4	100-1000	100,000
Cost ($/MB)	---	$100	$30	$1	$0.05
Technology	flip-flops	SRAM	SRAM	DRAM	Disk

A Safe Place for Data

- Volatile main memory (RAM)
  - Loses instructions and data when power off
- Non-volatile secondary memory
  - Magnetic disk
  - Flash memory
  - Optical disk (CDROM, DVD)

Large amounts of main memory → more applications can run at once, or memory intensive applications run faster

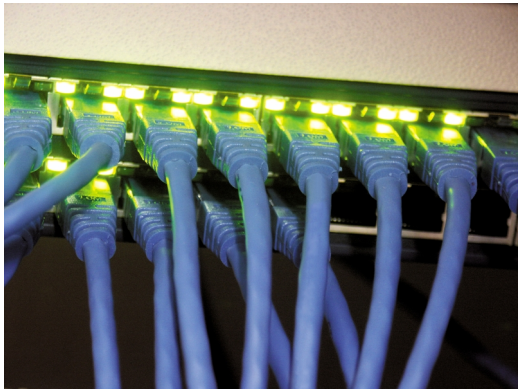
Large amounts of secondary memory → required for very large datasets



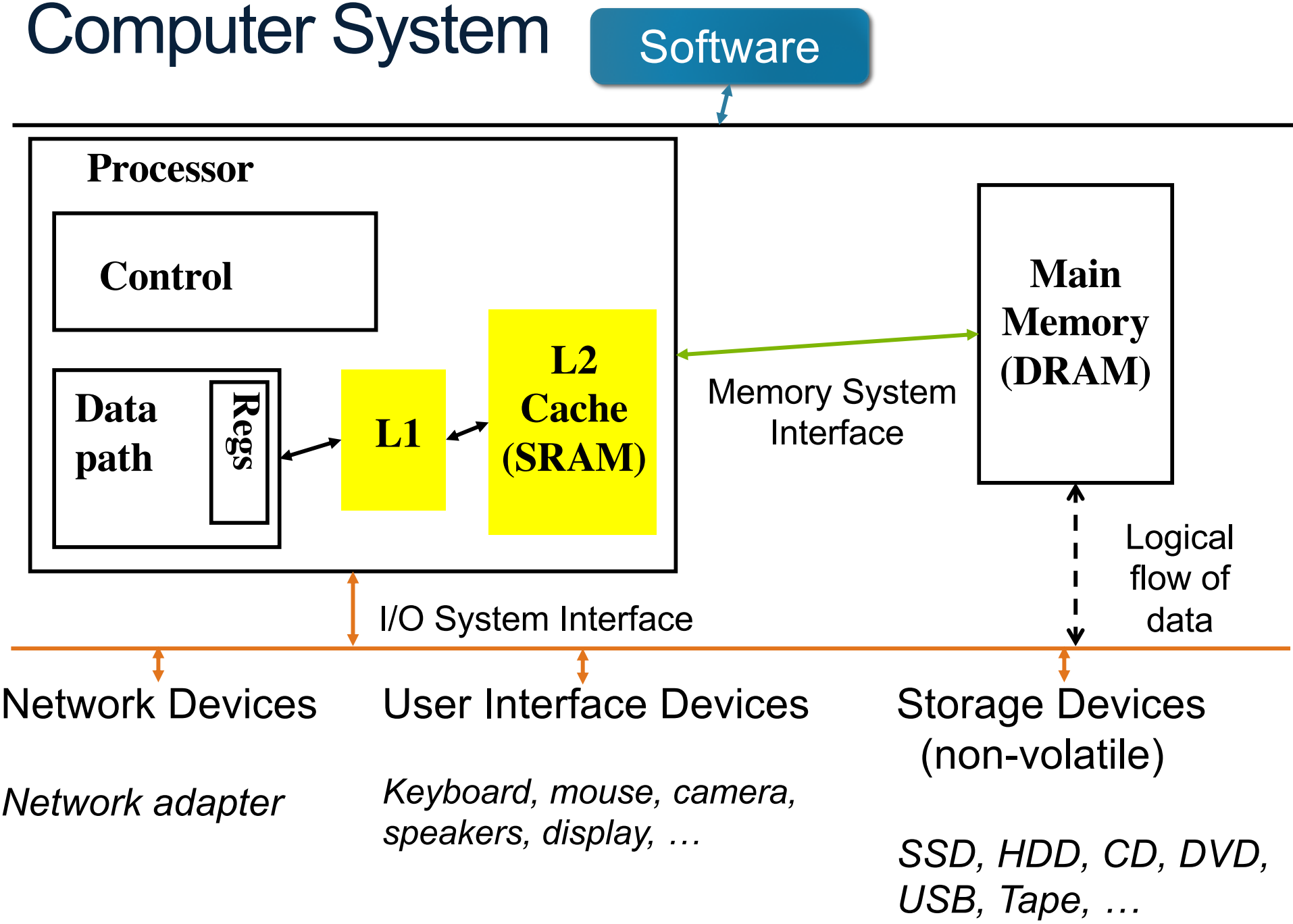
Networks

- Communication and resource sharing
- Network technologies, physical media and locality determine performance
  - Local area network (LAN): Ethernet
    - Within a building
  - Wide area network (WAN: the Internet)
  - Wireless network: WiFi, Bluetooth

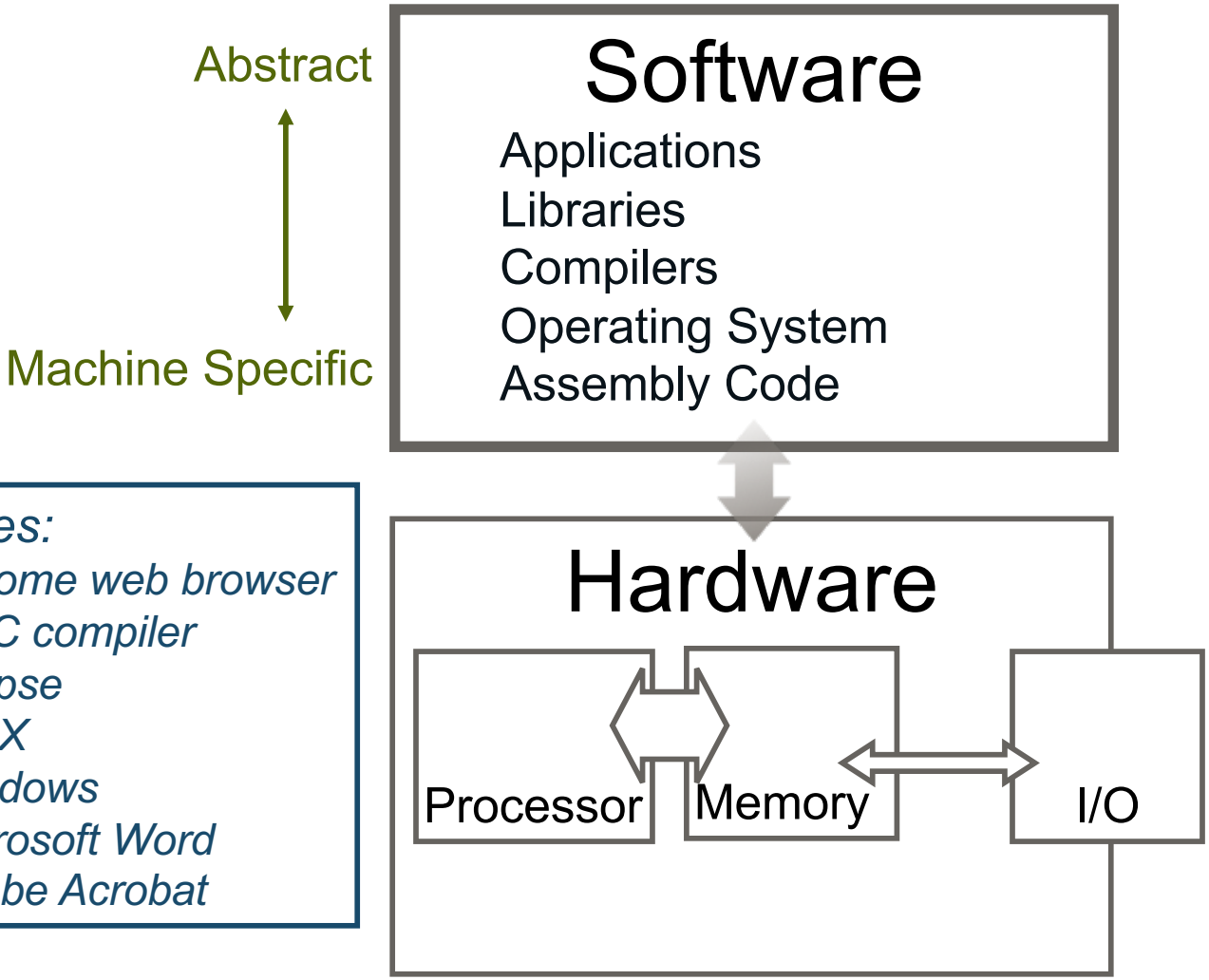
Network connectivity and speed determine latency and bandwidth to other systems on the network.



Computer System



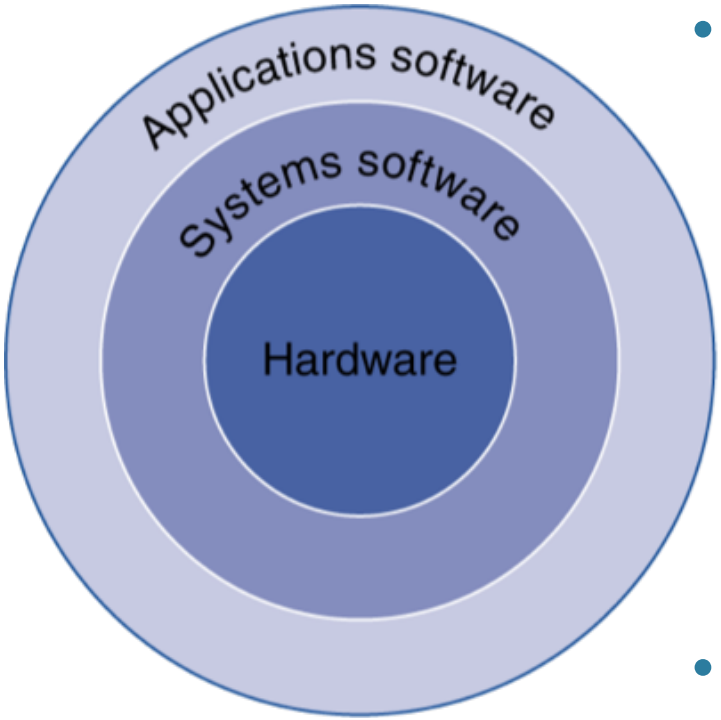
Elements of a Computer: Software



Examples:
- *Chrome web browser*
- *GCC compiler*
- *Eclipse*
- *OS X*
- *Windows*
- *Microsoft Word*
- *Adobe Acrobat*

Software:
The data and instructions that control what happens in the hardware.
Software written in high level languages and abstractions must be translated into low level representations that can be used by the hardware.

Below Your Program



- Application software
  - Written in high-level language
- System software
  - Compiler: translates HLL code to machine code
  - Operating System: service code
    - Handling input/output
    - Managing memory and storage
    - Scheduling tasks & sharing resources
- Hardware
  - Processor, memory, I/O controllers

Levels of Program Code

- High-level language
  - Level of abstraction closer to problem domain
  - Provides for productivity and portability
  - 5-10 assembly instructions per line of code
- Assembly language
  - Textual representation of instructions
- Hardware representation
  - Binary digits (bits)
  - Encoded instructions and data

```
swap(int v[], int k) {
    int temp;
    temp = v[k];
    v[k] = v[k+1];
    v[k+1] = temp;
}
```

Compiler

```
swap: muli $2, $5, 4
      add $2, $4, $2
      lw $15, 0($2)
      lw $16, 4($2)
      sw $16, 0($2)
      sw $15, 4($2)
      jr $31
```

Assembler

```
00000000101000000000000011000
00000000001100000000011000100001
10001100011000100000000000000000
10001100111100100000000000000100
10101100111100100000000000000000
10101100011000100000000000000100
00000011111000000000000000001000
```