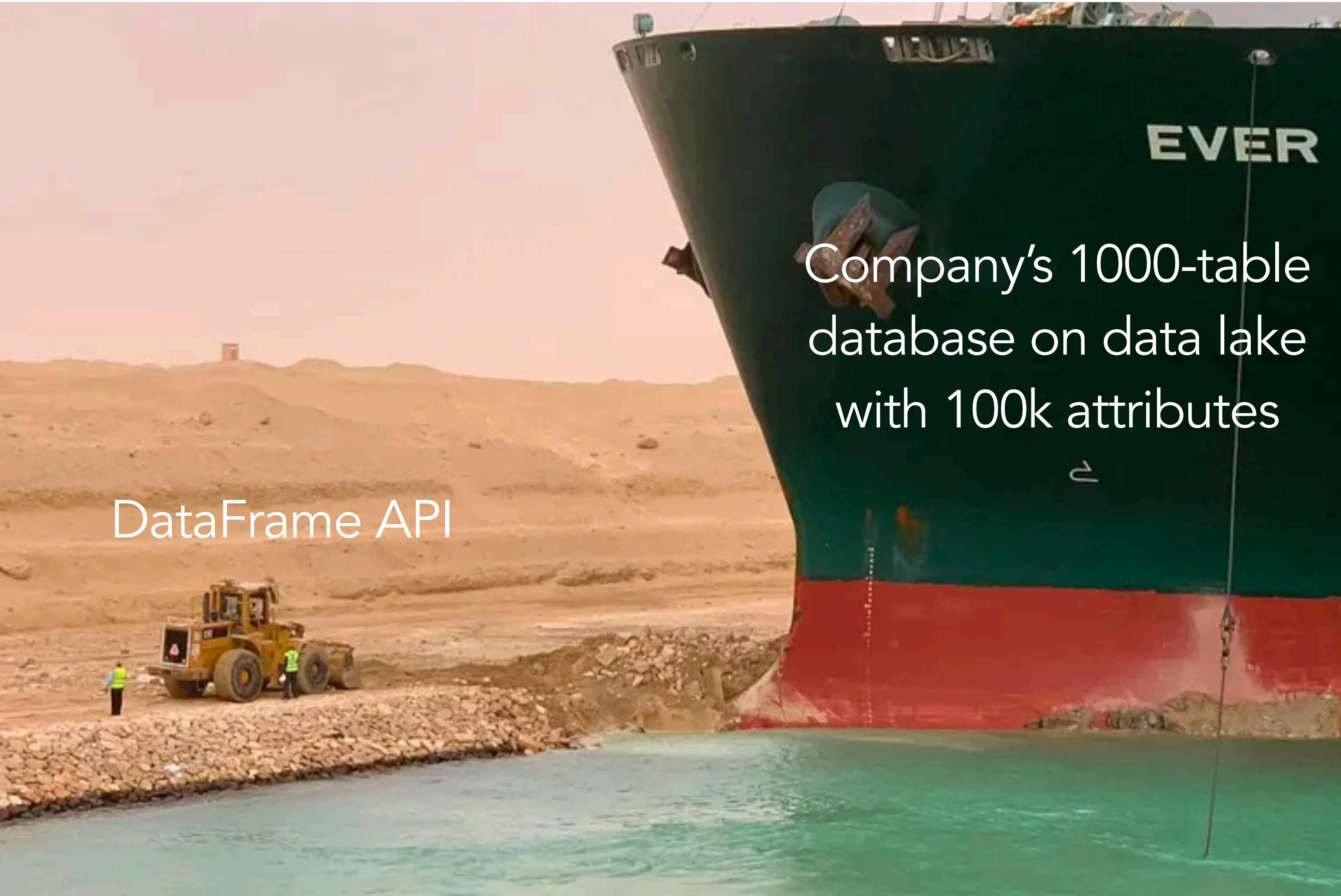


DSC 204a

Scalable Data Systems

- Haojian Jin



Logistics

- Give me feedback!
 - Diverse background.
 - So I can adjust the speed.
 - Wrong feedback loop.
 - Less is more!
- We will release the first assignment next Monday.

Peer Instruction Activity



bit.ly/dsc204aapr7

Peer Instruction Activity (About 1 min per 1 pt)

1. [3 x 4pts] What is the result of these calculations in final decimal representation?

- A. $10_2 + 10_{10} + 10_{16}$
- B. $0xABC - 0xBC + 0xC$
- C. $101_3 * 0x10$

- A. 28
- B. 2572
- C. 160

Peer Instruction Activity (About 1 min per 1 pt)

2. [3 x 3pts] What is the hexadecimal representation of these numbers in the given bases?

A. 161 in base 10

B. 32 in base 4

C. 64 in base 8

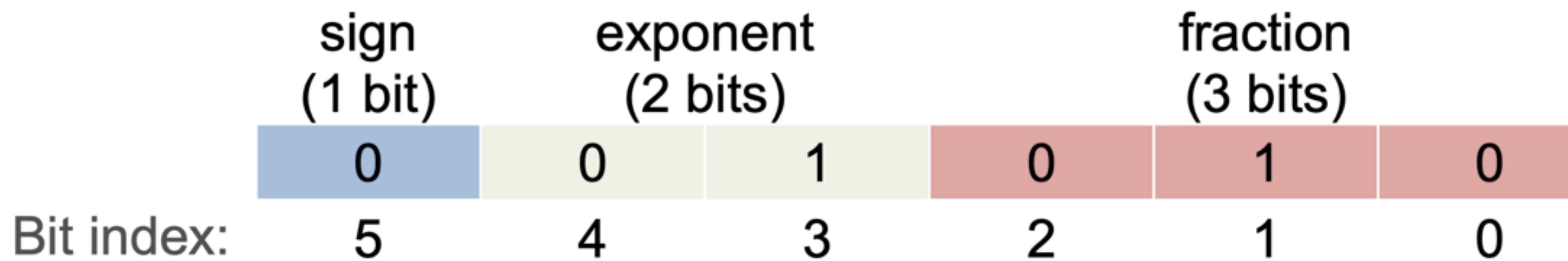
A. A1₁₆ (aka 0xA1)

B. 0xE

C. 0x34

Peer Instruction Activity (About 1 min per 1 pt)

Here is a custom float6 representation and interpretation.



$$(-1)^{sign} \times 2^{exponent-2} \times \left(1 + \sum_{i=1}^3 b_{3-i} 2^{-i}\right)$$

3. [3 x 3pts] What is the decimal real number representation of these float6 bit sequences:

A. 001010

A. $5/8$, i.e., 0.625

B. 111101

B. $-13/4$, i.e., -3.25

Review Questions

- Why do computers use binary digits?
- How many integers can you represent with 5 bits?
- How many bits do you need to represent 5 integers?
- What is the hexadecimal for 20_{10} ?
- Why do we need a float standard?
- Why should a data scientist know about float formats?
- What does “lower precision” mean for a float weight in DL?

Where are we in the class?

Foundations of Data Systems

- Digital representation of Data → **Basics of Computer Organization** → Basics Operating Systems
- Array data representation → Storage Memory Hierarchy → Cache
- Data encoding and evolution

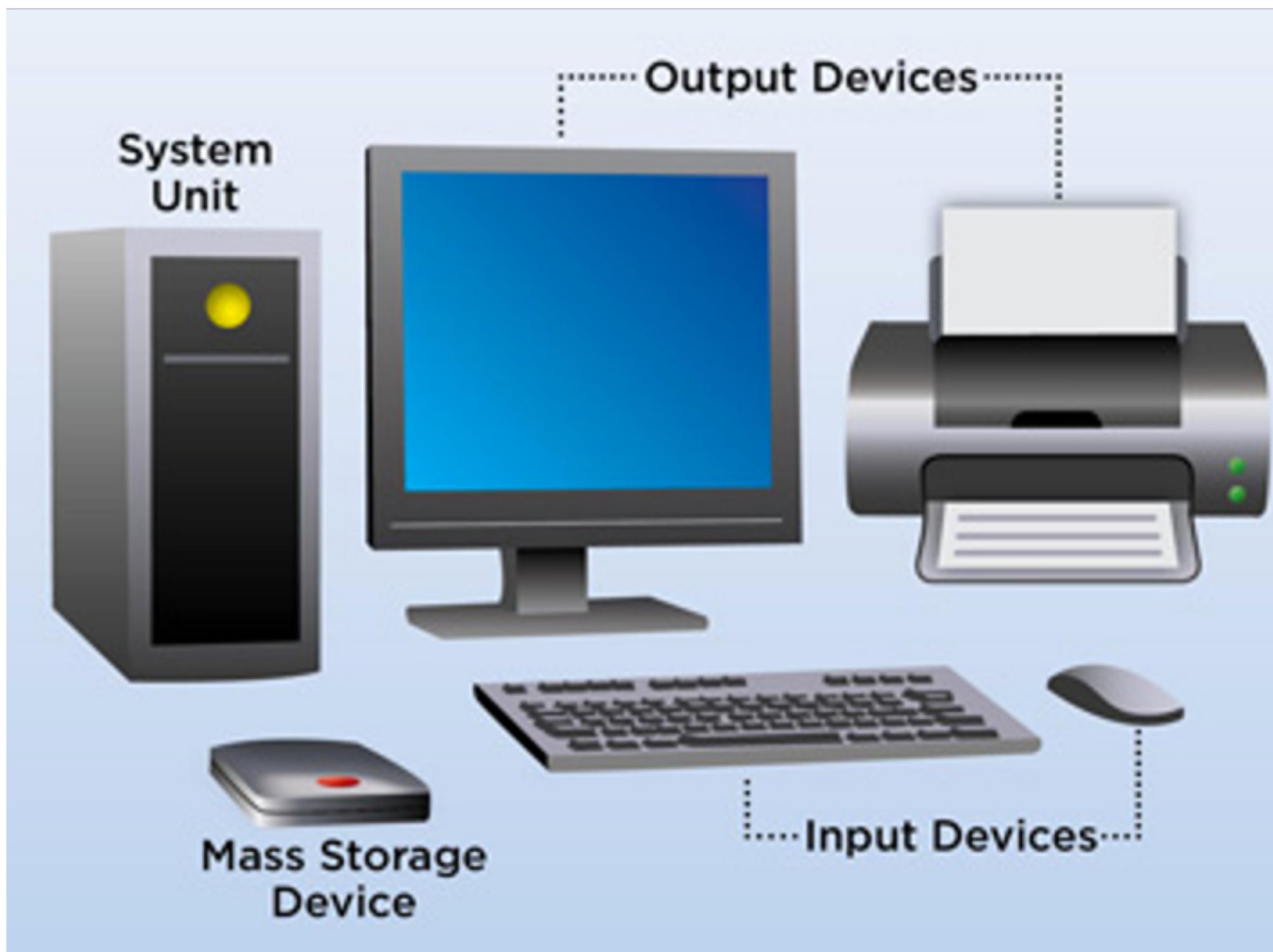
Scaling Distributed Systems

Data Processing and Programming model

What is a computer?

A **programmable** electronic device that can
store, retrieve, and process digital data.

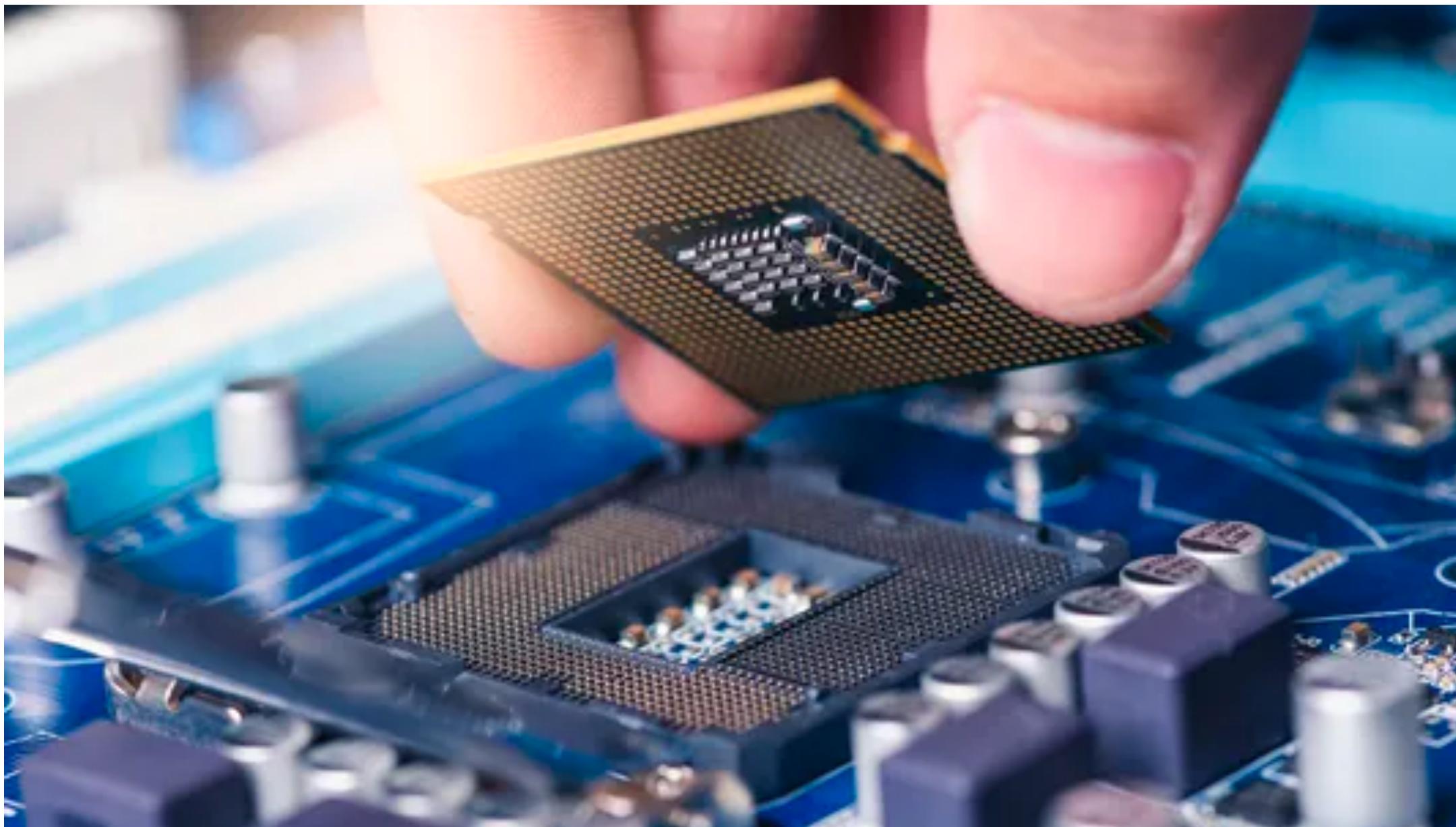
Basics of Computer Organization



- **Hardware:** The electronic machinery (wires, circuits, transistors, capacitors, devices, etc.)
- **Software:** Programs (instructions) and data

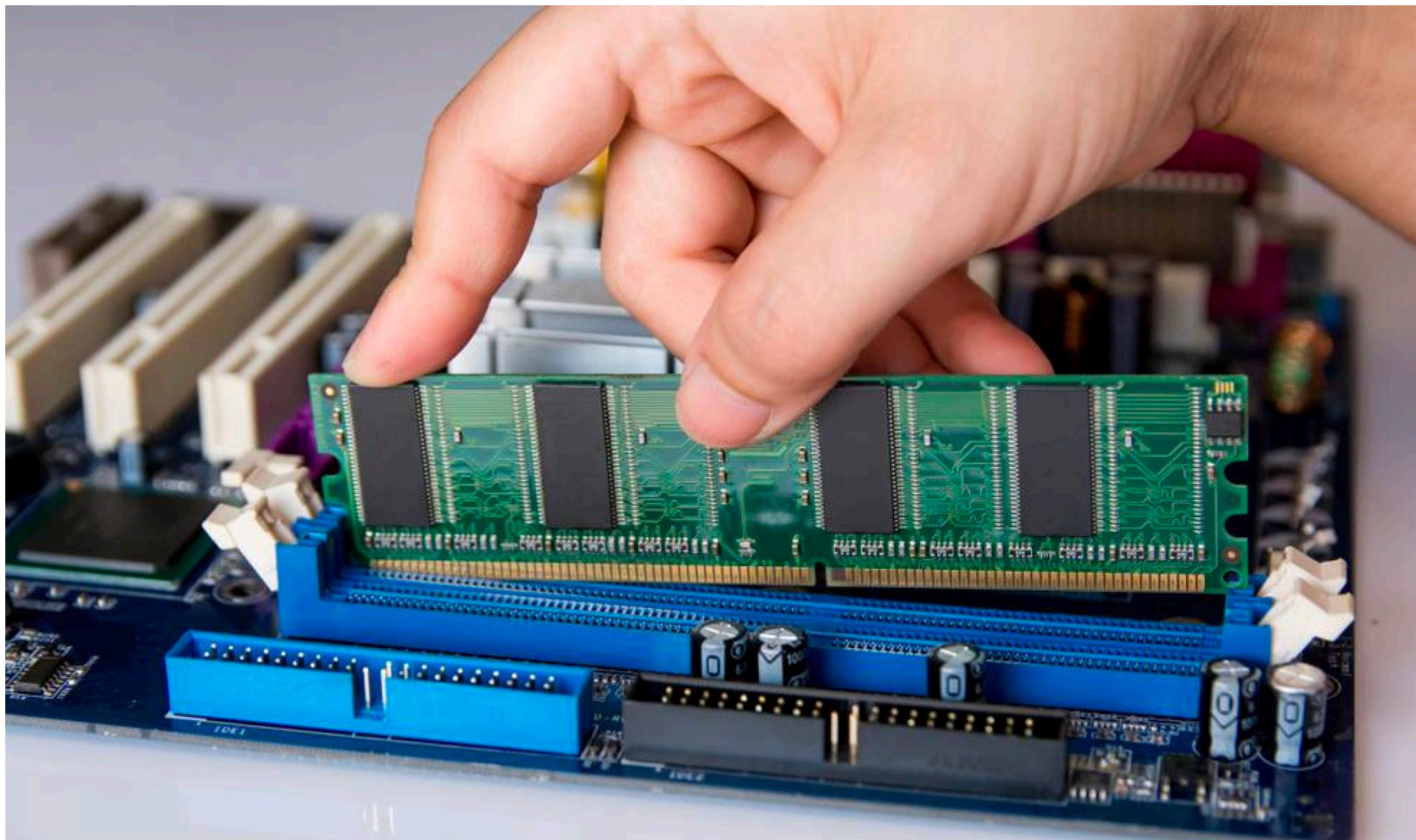
Key Parts of Computer Hardware

- Processor (CPU, GPU, etc.)
 - Hardware to orchestrate and execute instructions to manipulate data as specified by a program



Key Parts of Computer Hardware

- Main Memory (aka Dynamic Random Access Memory)
 - Hardware to store data and programs that allows very fast location/retrieval; byte-level addressing scheme



Key Parts of Computer Hardware

- Disk (aka secondary/persistent storage)
 - Similar to memory but persistent, slower, and higher capacity / cost ratio; various addressing schemes



Key Parts of Computer Hardware

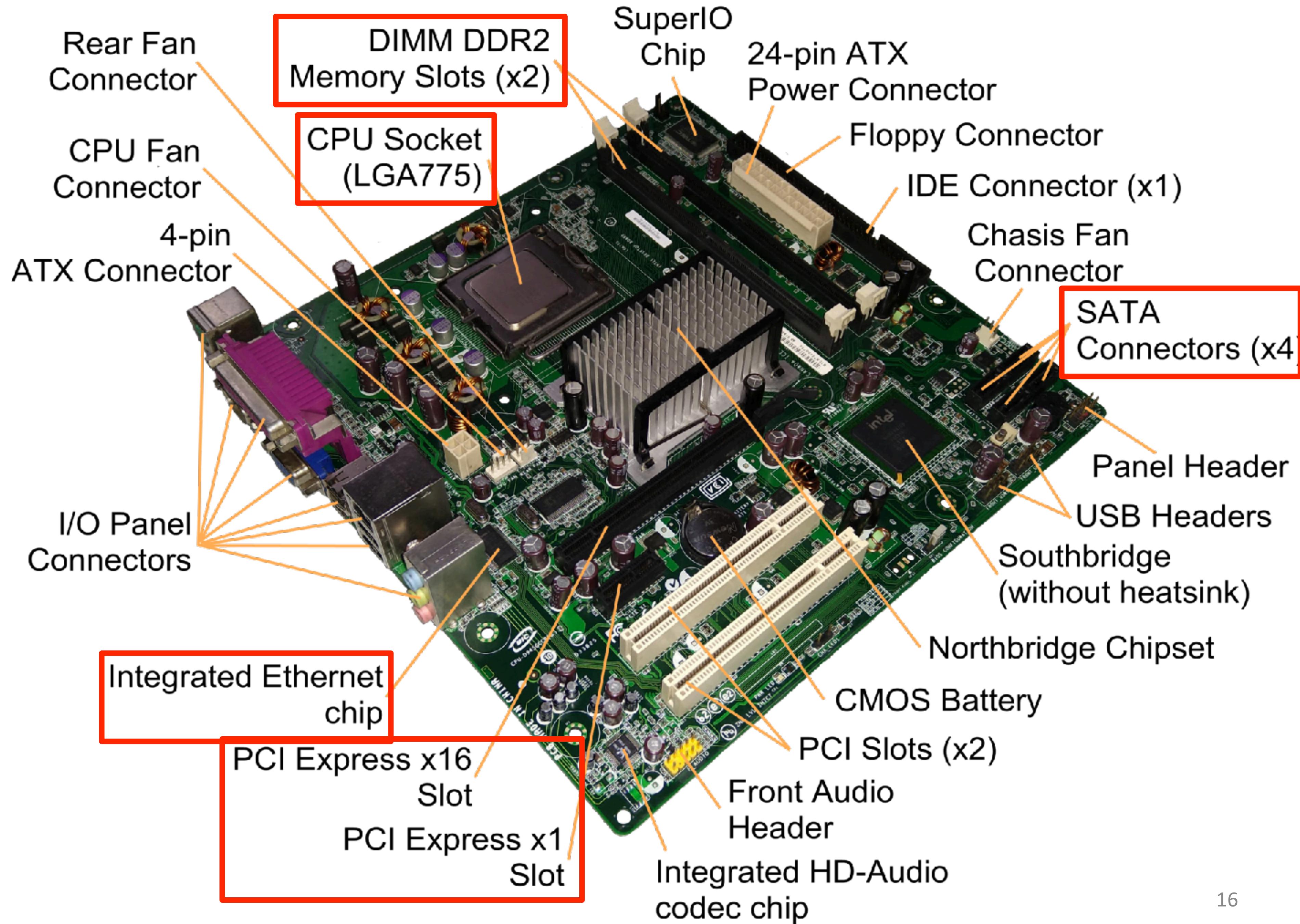
- Network interface controller (NIC)
 - Hardware to send data to / retrieve data over network of interconnected computers/devices



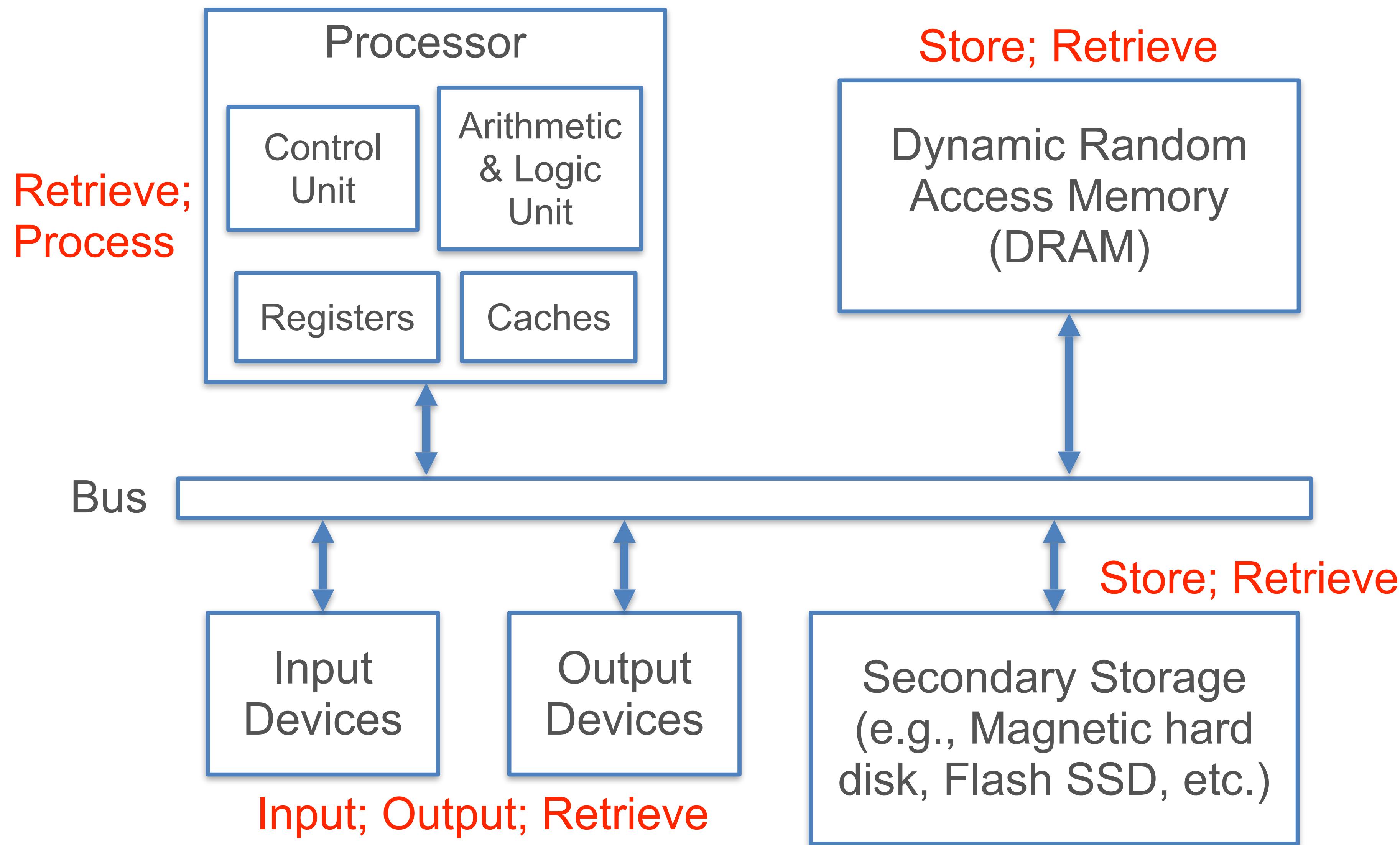
Bonus: Debugging your network



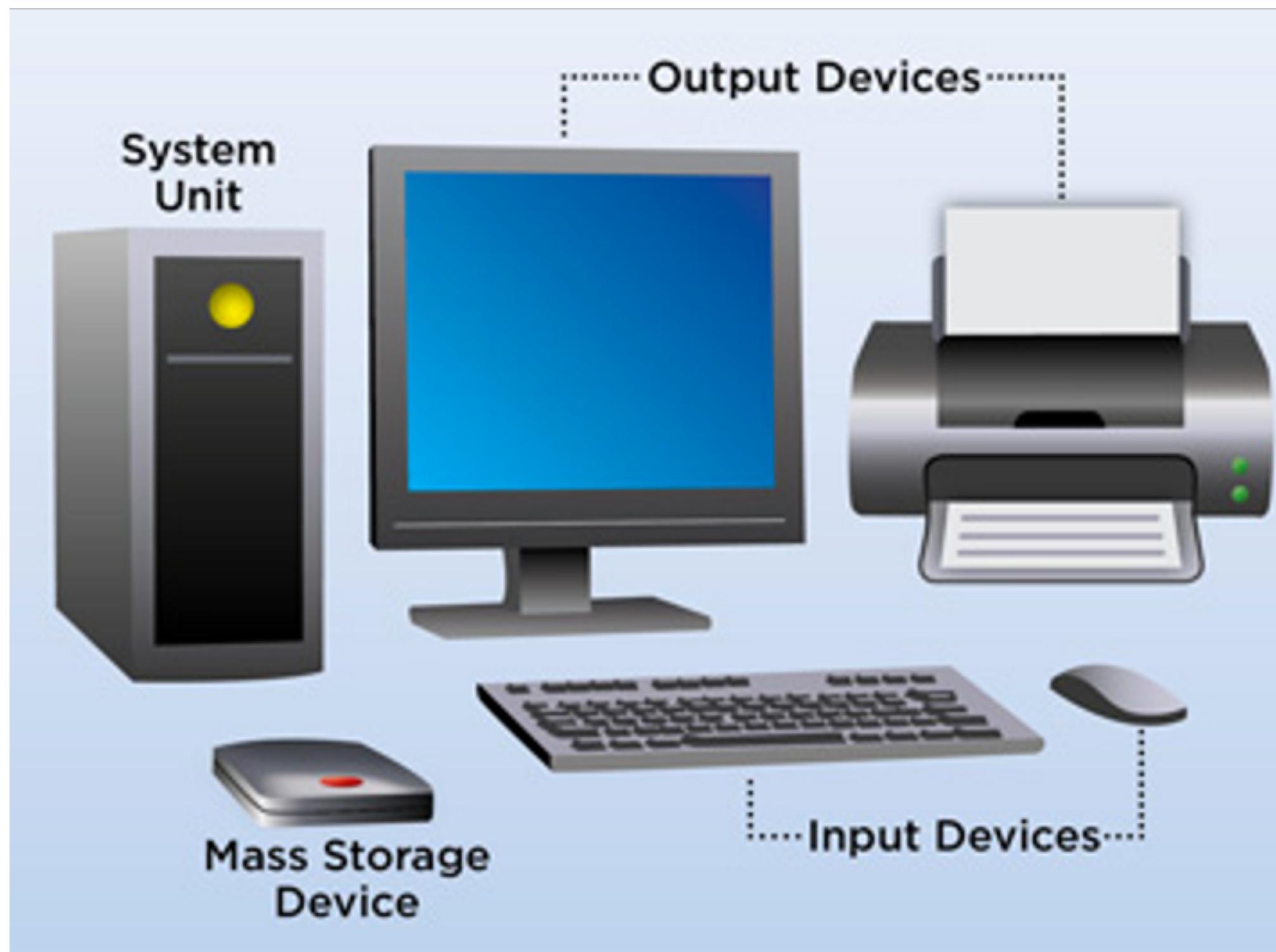
- **Green**
 - on: 100 Mbps (fast)
 - off: 10 Mbps (slow)
- **Orange**
 - on: Connected. No data
 - off: Disconnected
 - blinking: being transferred.



Abstract Computer Parts and Data



Parts of a Computer



- **Hardware:** The electronic machinery (wires, circuits, transistors, capacitors, devices, etc.)
- **Software:** Programs (instructions) and data

Key Aspects of Software

- Instruction
 - A command understood by hardware; finite vocabulary for a processor: Instruction Set Architecture (ISA); bridge between hardware and software
- Program (aka code)
 - A collection of instructions for hardware to execute

Key Aspects of Software

- Programming Language (PL)
 - A human-readable formal language to write programs; at a much higher level of abstraction than ISA
- Application Programming Interface (API)
 - A set of functions (“interface”) exposed by a program/set of programs for use by humans/other programs
- Data
 - Digital representation of information that is stored, processed, displayed, retrieved, or sent by a program

Main kinds of Software

- **Firmware**
 - Read-only programs “baked into” a device to offer basic hardware control functionalities
- **Operating System (OS)**
 - Collection of interrelated programs that work as an intermediary platform/service to enable application software to use hardware more effectively/easily
 - Examples: Linux, Windows, MacOS, etc.

Main kinds of Software

- Application Software
 - A program or a collection of interrelated programs to manipulate data, typically designed for human use
 - Examples: Excel, Chrome, PostgreSQL, etc.