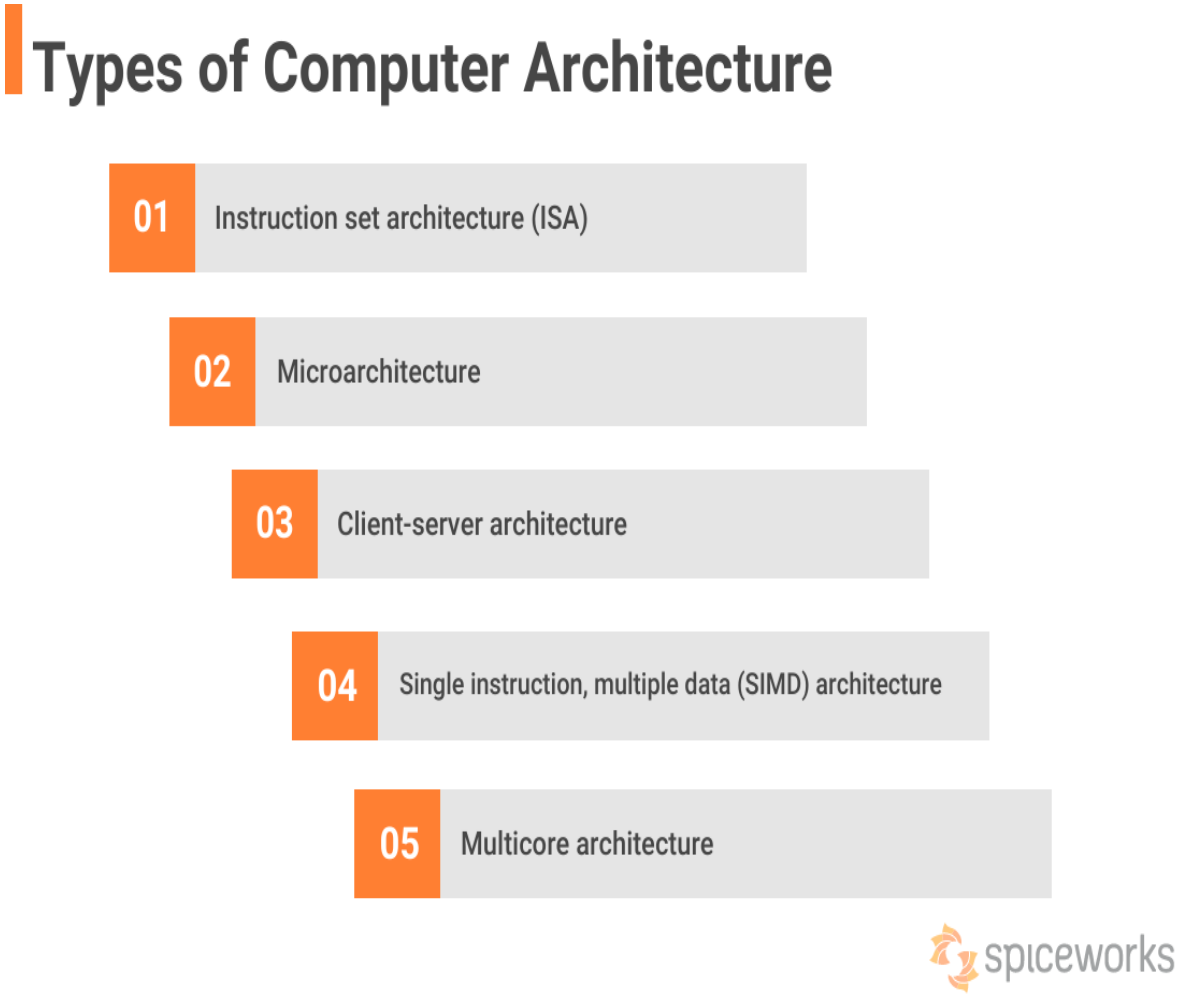
Types of Computer Architecture

It is possible to set up and configure the above architectural components in numerous ways. This gives rise to the different types of computer architecture. The most notable ones include:



Types of Computer Architecture

1. Instruction set architecture (ISA)

Instruction set architecture (ISA) is a bridge between the software and hardware of a computer. It functions as a programmer's viewpoint on a machine. Computers can only comprehend binary language (0 and 1), but humans can comprehend high-level language (if-else, while, conditions, and

the like). Consequently, ISA plays a crucial role in user-computer communications by translating high-level language into binary language.

In addition, ISA outlines the architecture of a computer in terms of the fundamental activities it must support. It's not involved with implementation-specific computer features. Instruction set architecture dictates that the computer must assist:

- **Arithmetic/logic instructions:** These instructions execute various mathematical or logical processing elements solely on a single or maybe more operands (data inputs).
- **Data transfer instructions:** These instructions move commands from the memory or into the processor registers, or vice versa.
- **Branch and jump instructions:** These instructions are essential to interrupt the logical sequence of instructions and jump to other destinations.

2. Microarchitecture

Microarchitecture, unlike ISA, focuses on the implementation of how instructions will be executed at a lower level. This is influenced by the microprocessor's structural design.

Microarchitecture is a technique in which the instruction set architecture incorporates a processor. Engineering specialists and hardware scientists execute ISA with various microarchitectures that vary according to the development of new technologies. Therefore, processors may be physically designed to execute a certain instruction set without modifying the ISA.

Simply put, microarchitecture is the purpose-built logical arrangement of the microprocessor's electrical components and data pathways. It facilitates the optimum execution of instructions.

3. Client-server architecture

Multiple clients (remote processors) may request and get services from a single, centralized server in a client-server system (host computer). Client computers allow users to request services from the server and receive the server's reply. Servers receive and react to client inquiries.

A server should provide clients with a standardized, transparent interface so that they are unaware of the system's features (software and hardware components) that are used to provide the service.

Clients are often located on desktops or laptops, while servers are typically located somewhere else on the network, on more powerful hardware. This computer architecture is most efficient when the clients and the servers frequently perform pre-specified responsibilities.

4. Single instruction, multiple data (SIMD) architecture

Single instruction, multiple data (SIMD) computer systems can process multiple data points concurrently. This cleared the path for supercomputers and other devices with incredible performance capabilities. In this form of design, all processors receive an identical command from the control unit yet operate on distinct data packets. The shared memory unit requires numerous modules to interact with all CPUs concurrently.

5. Multicore architecture

Multicore is a framework wherein a single physical processor has the logic of multiple processors. A multicore architecture integrates numerous processing cores onto only one integrated circuit. The goal is to develop a system capable of doing more tasks concurrently, improving overall system performance.

Examples of Computer Architecture

Two notable examples of computer architecture have paved the way for recent advancements in computing. These are 'Von Neumann architecture' and 'Harvard architecture.' Most other architectural designs are proprietary and are therefore not revealed in the public domain beyond a basic abstraction.

Here's a description of what these two examples of computer architecture are all about.

1. Von Neumann architecture

The von Neumann architecture, often referred to as the Princeton architecture, is a computer architecture that was established in a 1945 presentation by John von Neumann and his collaborators in the First Draft of a Report on the EDVAC (electronic discrete variable automatic computer). This example of computer architecture proposes five components:

- A processor with connected registers
- A control unit capable of storing instructions
- Memory capable of storing information as well as instructions and communicating via buses
- Additional or external storage
- Device input as well as output mechanisms

2. Harvard architecture

The Harvard architecture refers to a computer architecture with distinct data and instruction storage and signal pathways. In contrast to the von Neumann architecture, in which program instructions and data use the very same memory and pathways, this design separates the two. In practice, a customized Harvard architecture with two distinct caches is employed (for data and instruction); X86 and Advanced RISC Machine (ARM) systems frequently employ this instruction.