

## Multiprocessor Definitions

**multiprocessor** A computer system with at least two processors. This is in contrast to a **uniprocessor**, which has one.

### **multicore microprocessor**

A microprocessor containing multiple processors (“**cores**”) in a single integrated circuit.

**job-level parallelism** or **process-level parallelism** Utilizing multiple processors by running independent programs simultaneously.

**parallel processing program** A single program that runs on multiple processors simultaneously.

**shared memory multiprocessor (SMP)** is a multiprocessor system that offers the programmer a *single physical address space* across all processors. Note that such systems can still run independent jobs/programs in their own virtual address spaces (virtual memory), even if they all share a physical address space (physical memory). Processors communicate through shared variables in memory, with all processors capable of accessing any memory location via loads and stores.

**cluster** A set of computers connected over a local area network (LAN) that functions as a single large multiprocessor system. These processors communicate over standard network switches via message-passing. They do not share a single physical address space.

### Types of multiprocessors based on Instruction and Data parallelism

**instruction-level parallelism** The parallelism among instructions.

**data-level parallelism** Parallelism achieved by operating on independent data.

A conventional uniprocessor has a single instruction stream and single data stream, and a conventional multiprocessor has multiple instruction streams and multiple data streams.

**SISD**. Single Instruction stream, single Data stream. A uniprocessor.

**MIMD**. Multiple Instruction streams, multiple data streams. A multiprocessor.

**SIMD.** Single Instruction stream, multiple data streams.

A multiprocessor. The same instruction is applied to many data streams, as in a vector processor or array processor.

**SPMD.** Single Program, multiple Data streams. The conventional MIMD programming model, where a single program runs across all processors.

Reference:

Chapter 7, fourth edition of Computer Organization and Design by Patterson and Hennessy.