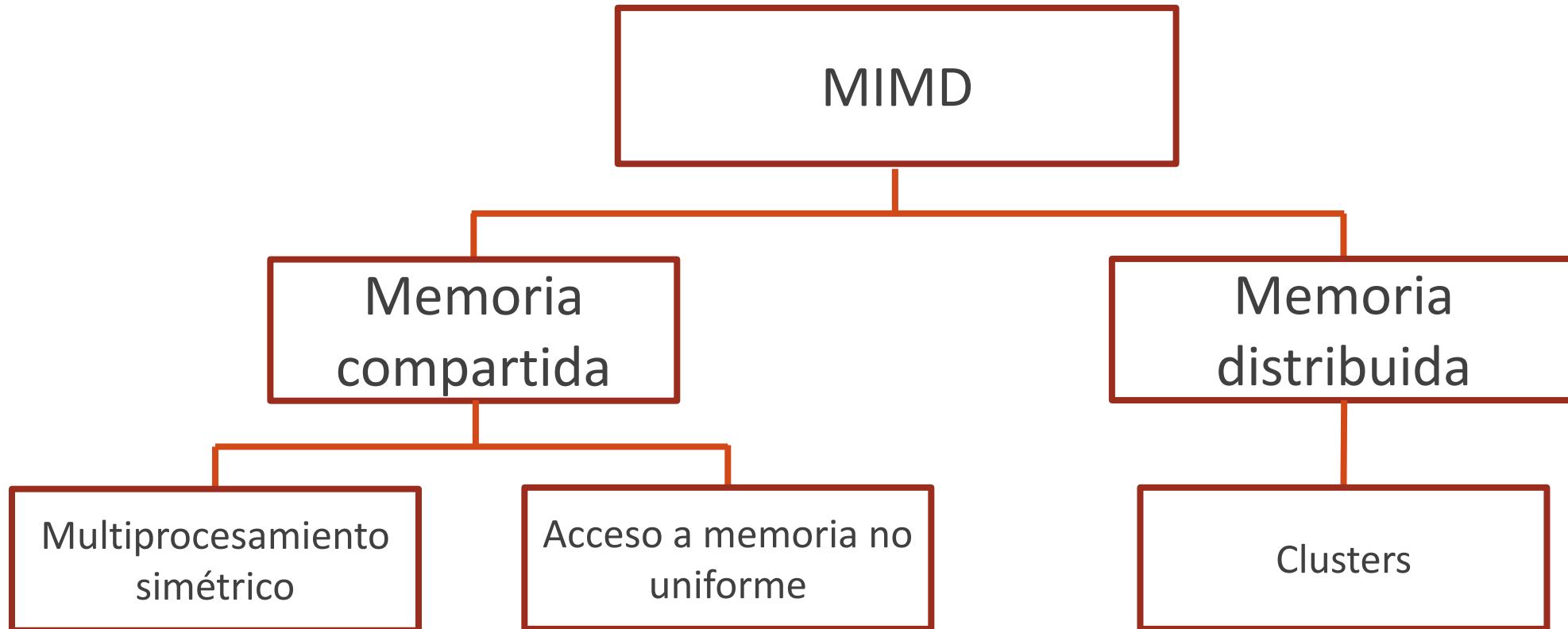


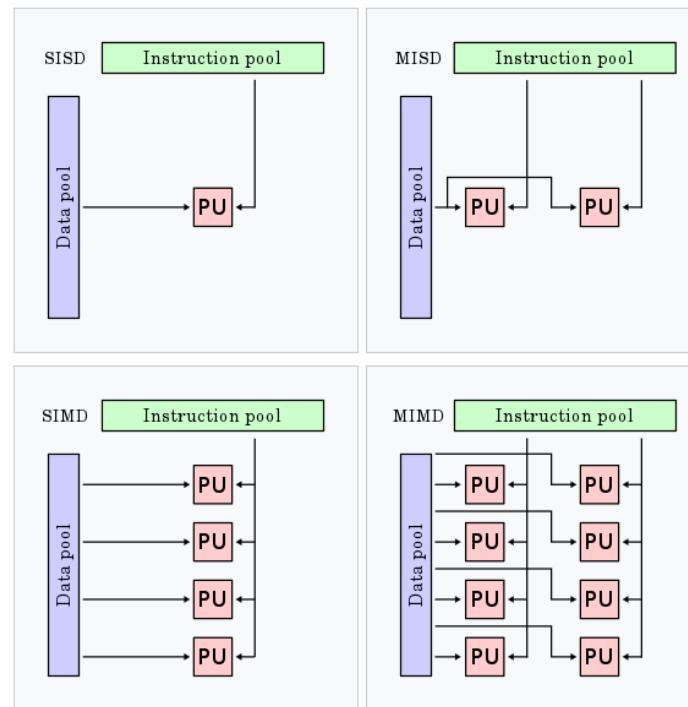
# Taxonomía de procesadores - MIMD

- Un conjunto de procesadores ejecuta simultáneamente secuencias diferentes de instrucciones sobre conjuntos diferentes de datos.



# Taxonomía de procesadores

---



[https://en.wikipedia.org/wiki/Flynn%27s\\_taxonomy](https://en.wikipedia.org/wiki/Flynn%27s_taxonomy)

# Paralelismo a nivel de instrucciones - SISD

---

- Se realizaba una ejecución “simultánea” de las instrucciones.
- Pipeline, Superpipeline.
- Superescalares.
  - Ejecución fuera de orden.
  - Ejecución dinámica (Tomasulo, Marcador).
- Calendarización estática:
  - VLIW.

# Paralelismo a nivel de instrucciones - SISD

---

Surgen limitaciones que son inevitables:

- Dependencias
- Riesgos

# MIMD

---

Los sistemas multiprocesador deben establecer las reglas para distribuir las tareas según las siguientes categorías:

- **Simetría:** simétricos o asimétricos.
- **Tiempo de acceso:** UMA o NUMA.
- **Organización de memoria:** centralizada o distribuida.
- **Direccionamiento:** unificado o privado.

# MIMD

---

Surgen limitaciones que son inevitables:

- Necesitamos más hardware.
  - Mayor área y potencia.
- Sincronización
  - Tenemos más recursos compartidos.
- Memoria
  - Se realizan accesos simultáneos a memoria.

# Taxonomías

---

## Flynn's taxonomy

**SISD** – Single Instruction, Single Data stream

Defines serial computers

**MISD** – Multiple Instruction, Single Data stream

Multiple processors applying different instructions to a single data stream.

Possible but impractical.

**SIMD** – Single Instruction, Multiple Data streams

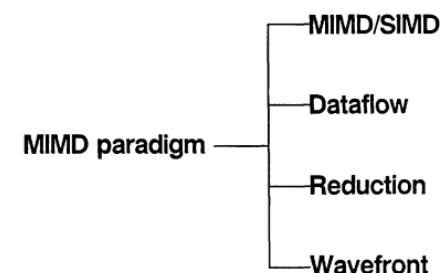
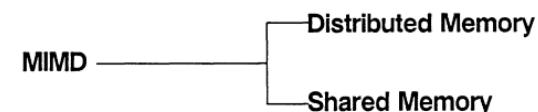
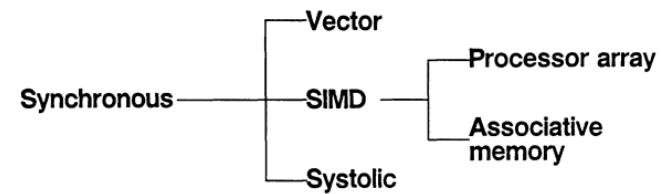
Multiple processors simultaneously executing the same instruction on different data.

**MIMD** – Multiple Instruction, Multiple Data streams

Multiple processors autonomously executing diverse instructions on diverse data.

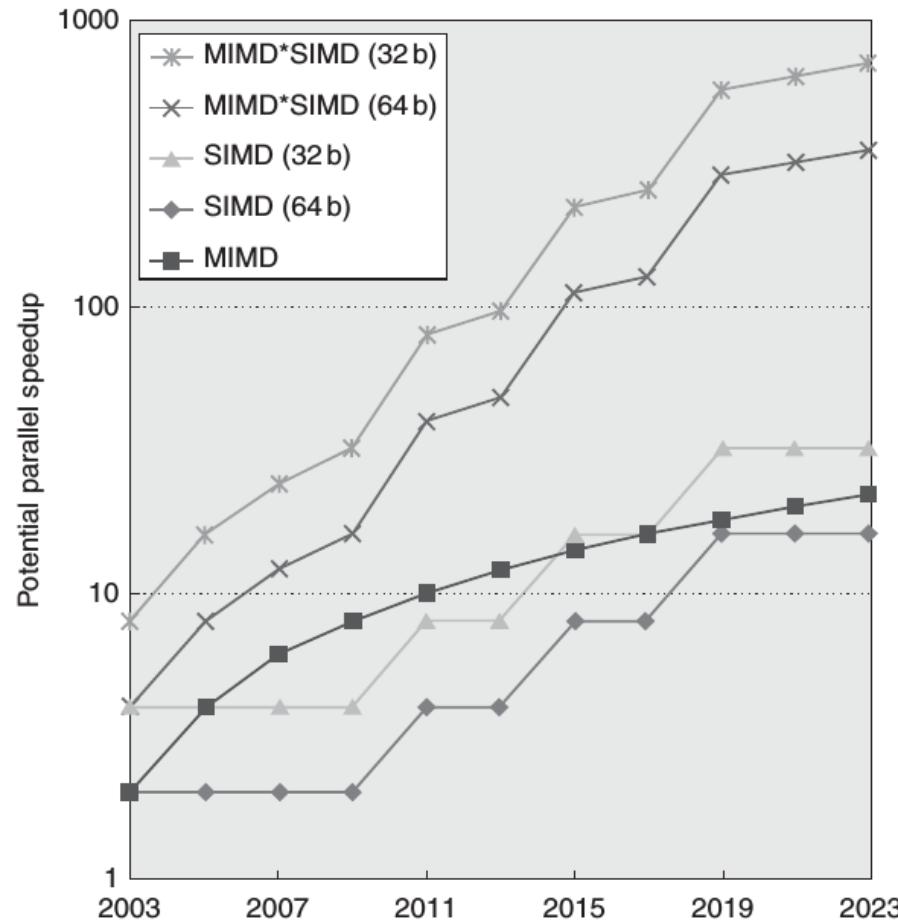
<https://cds.cern.ch/record/319566/files/AT00000317.pdf>

## Duncan's taxonomy



# Mejora potencial en computadoras x86

---



# SIMD

---

- Los procesadores ejecutan las instrucciones en diferentes conjuntos de datos de forma paralela.
- Se utiliza en operaciones vectoriales.
- Toma un conjunto de datos, los coloca de manera secuencial, opera sobre esos registros y luego envía los datos a memoria.
- Al tratarse de vectores, sólo se hace una penalización de memoria por vector y no por cada elemento.

# SIMD

---

- Sumar 64 números con una sola instrucción.
- Hardware envía 64 datos a 64 ALUs para obtener 64 resultados en el mismo ciclo de reloj.
- Máxima eficiencia en programas con alto paralelismo de datos.
  - Por ejemplo: un arreglo en un *for*.
- Arm Neon: extensión para procesadores Arm para instrucciones SIMD.
  - Mejorar multimedia, gráficos 2D/3D y videojuegos.