

Arquitectura Paralela. HPC y Tipos.

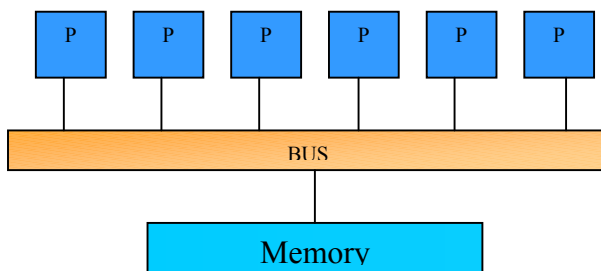
¿Que es HPC? (Computación de Alto Rendimiento)

Puede significar rapidez de ejecución, con sistemas mas poderosos o que usa múltiples sistemas simultáneamente. Otros lo definen como varios procesadores trabajando sobre un mismo problema (Computación Paralela) o varios procesadores trabajando sobre problemas relacionados (Computación Distribuida).

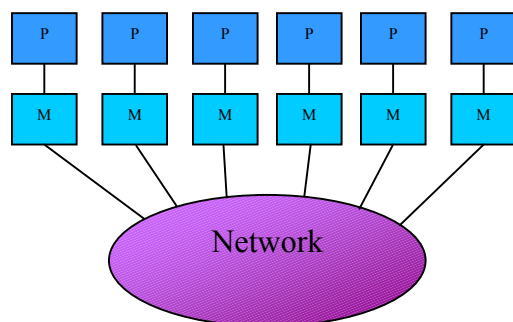
Pero en si, la HPC no es mas que *cualquier* técnica computacional que solucione un problema grande de forma más rápida que usando posiblemente sistemas simples. Esto se logra con procesadores de alto rendimiento (Ej. Cray), Computación Paralela, Computación Distribuida, etc.

La HPC ha tenido gran impacto sobre todas las áreas de las ciencias computacionales e ingeniería a nivel académico, gobierno e industria. Muchos problemas han sido solucionados con técnicas de HPC que eran imposibles de solucionar con estaciones de trabajo individuales o computadores personales. (Ej. Predicción del clima, Aerodinámica, Turbulencia en aviones, Combustión, etc)

Arquitectura Compartida vs. Distribuida. (Hardware del Equipo, cantidad de procesadores, distribución de la memoria, no tiene que ver con la distribución de los datos)

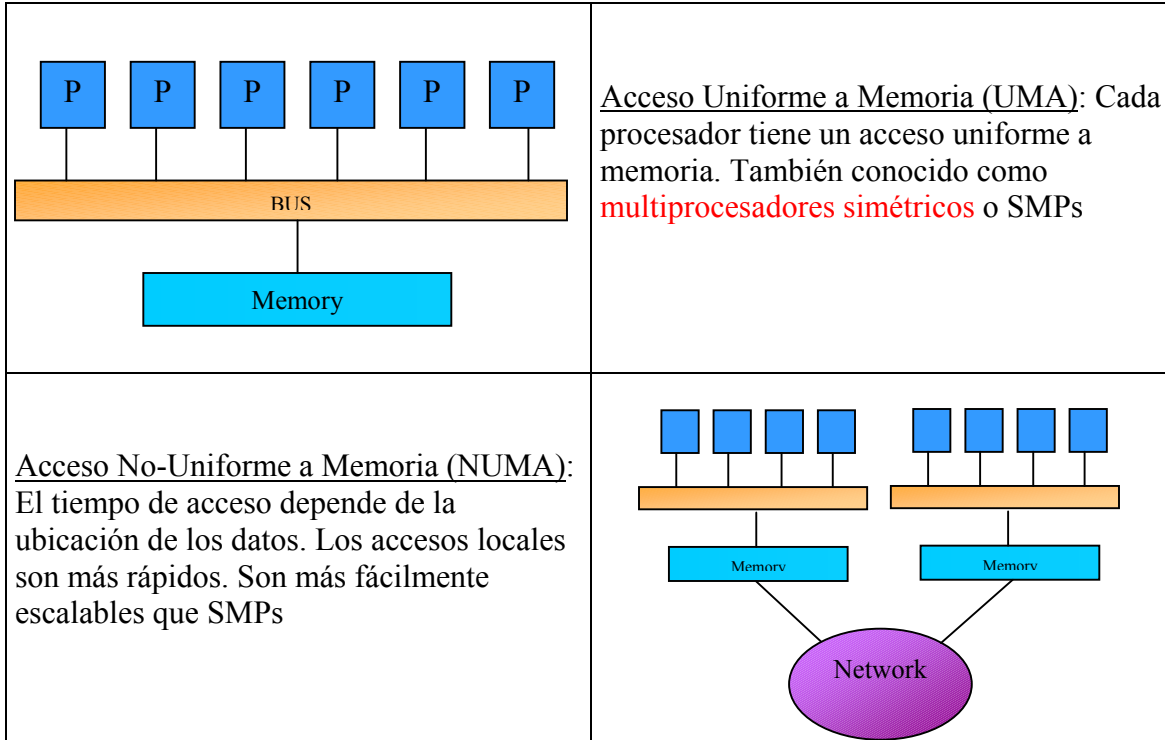


Memoria Compartida – Un espacio simple de direccionamiento. Todos los procesadores tienen acceso a un *pool* de memoria compartida



Memoria Distribuida – cada procesador tiene su propia memoria local. Se debe hacer paso de mensaje para intercambiar datos entre procesadores

Memoria Compartida: UMA vs NUMA



Tipos de Paralelismo. (Programación y Manipulación de los Datos, no tiene que ver con la distribución del hardware del equipo)

A nivel de manipulación de datos (es diferente a la **Memoria Compartida** y **Distribuida** porque el paralelismo tiene que ver con la cantidad de procesadores y la distribución del hardware en la maquina paralela) el paralelismo se puede ver como:

- **Paralelismo de datos**: cada procesador ejecuta la misma tarea sobre diferentes conjuntos o subregiones de datos
- **Paralelismo de tareas**: cada procesador ejecuta una diferente tarea
- La mayoría de las aplicaciones están entre estos dos extremos

Una común forma de usar paralelismo en el desarrollo de aplicaciones desde hace años fue el paralelismo Maestro-Eslavo:

- Un simple procesador es responsable de distribuir los datos y recolectar los resultados (paralelismo de tarea)
- Todos los otros procesadores ejecutan la misma tarea sobre su porción de datos (paralelismo de datos)