

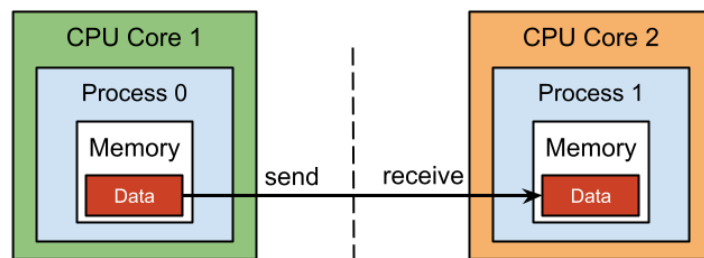
Universidad de Carabobo
Facultad Experimental de Ciencias y Tecnología
Departamento de Computación
Arquitectura del Computador

Paralelismo

Realizado por:
Leonardo Moreno - C.I.:28330952
Luis Romero – C.I.:26729561

MPI (Message Passing Interface)

MPI es una interfaz de programación diseñada para facilitar la programación paralela en sistemas distribuidos. Permite a los desarrolladores escribir programas que pueden ejecutarse simultáneamente en múltiples nodos de un clúster, facilitando así el procesamiento paralelo de grandes conjuntos de datos. La principal característica de MPI es la capacidad de intercambiar mensajes entre los nodos, lo que permite la sincronización y la comunicación entre los diferentes procesos en ejecución. Esto es esencial para aplicaciones científicas y de ingeniería que requieren una gran cantidad de cálculos, como simulaciones numéricas y modelos computacionales.

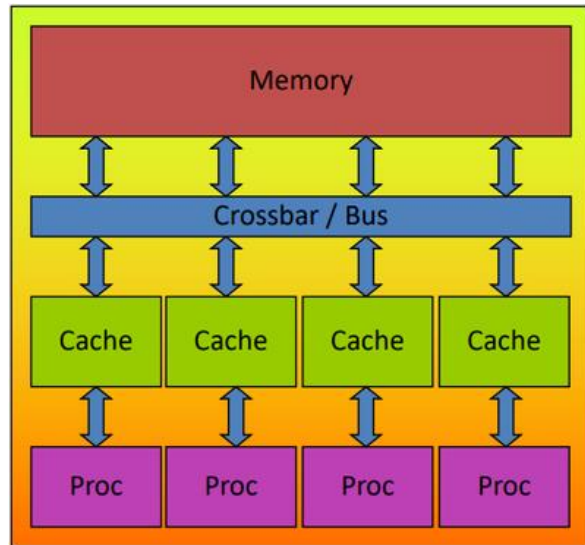


OpenMP (Open Multi-Processing)

es una API (Interfaz de Programación de Aplicaciones) de programación de memoria compartida utilizada para la programación paralela en sistemas de memoria compartida, como computadoras con múltiples núcleos de CPU. A diferencia de MPI, que se enfoca en la comunicación entre nodos en sistemas distribuidos, OpenMP se centra en la ejecución paralela en una sola máquina.

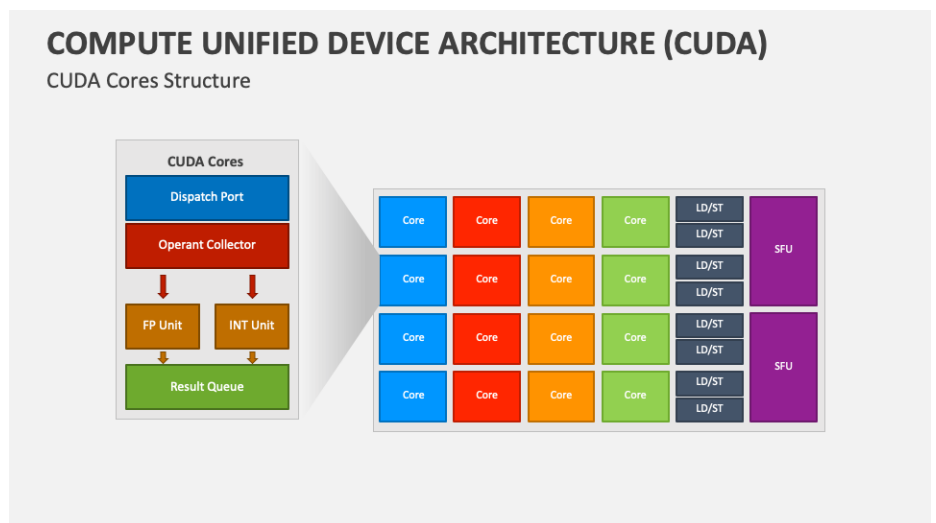
OpenMP simplifica la creación de programas paralelos al permitir a los desarrolladores agregar directivas de compilador a su código, indicando qué secciones de código pueden ejecutarse en paralelo. Estas directivas proporcionan instrucciones al compilador sobre cómo dividir el trabajo entre

múltiples hilos de ejecución, permitiendo que varias partes del programa se ejecuten simultáneamente en los diferentes núcleos de la CPU.



CUDA (Compute Unified Device Architecture)

CUDA es una plataforma desarrollada por NVIDIA que permite aprovechar la capacidad de procesamiento de las unidades de procesamiento gráfico (GPU) para realizar tareas de cómputo general. A diferencia de las CPU tradicionales, las GPU están diseñadas para manejar eficientemente operaciones en paralelo. CUDA proporciona un entorno de programación y un conjunto de herramientas que permiten a los desarrolladores escribir código para ejecutarse en las GPU de NVIDIA. Esto ha llevado a avances significativos en áreas como el aprendizaje profundo, la simulación científica y la visualización, donde la capacidad de procesamiento masivo de las GPU puede acelerar significativamente el rendimiento de las aplicaciones.



Diferencias y

Complementariedades

MPI: se centra en la programación paralela a nivel de clúster, permitiendo la ejecución de programas en múltiples nodos, mientras que CUDA se centra en el procesamiento paralelo en GPU, aprovechando la potencia de estas unidades de procesamiento.

En algunos casos, MPI y CUDA pueden combinarse, por ejemplo, para ejecutar simulaciones en paralelo distribuido entre nodos utilizando MPI, y aprovechar la potencia de las GPU para cálculos específicos dentro de cada nodo mediante CUDA. Esto permite una aproximación híbrida para obtener el mejor rendimiento en aplicaciones complejas y exigentes computacionalmente.

En resumen, MPI se centra en la comunicación entre nodos para ejecutar programas de manera distribuida, OpenMP se centra en la programación paralela en sistemas de memoria compartida y CUDA en el uso eficiente de las GPU para cálculos de alto rendimiento en una amplia gama de aplicaciones computacionales. Cada una tiene su enfoque particular, pero pueden combinarse para lograr resultados aún más potentes en áreas donde se superponen sus capacidades.