KUPE for Parallel System

이철
박소은

Index

- Why Parallel Computing?
- Challenges of Parallel Computing
- Solutions
- Applying to KUPE
- Q&A

Why Parallel Computing?

- Parallel Computing
  - A type of computation in which many calculations or the execution of processes are carried out simultaneously
  - Larger problems can often be divided into smaller ones

- Why parallel computing?
  - Due to the physical constraints preventing frequency scaling
  - Hard to improve performance above the power wall

- Using parallel computing
  - Climate modelling
  - Drug discovery
  - Data analysis

[Reference: An Introduction to Parallel Programming by Peter Pacheco]

Challenges of Parallel Computing

- Synchronization
  - More than two threads read or write the shared resource at the same time

- Communication
  - More than two processes or threads use to share the data that are not in the same memory space of each process or thread
  - For collaboration
  - Message Passing Interface(MPI)

- Load balancing
  - Important that each thread computes the balanced amount of tasks
  - For performance

Solutions

- Synchronization
  - Atomic execution
  - Lock
- Communication
  - Pipe
  - Blocking queue
- Load balancing
  - Task scheduling

Applying to KUPE

- Synchronization
  - Define requirements
  - Define Use case
  - Define synchronization sequence diagram
  - Define operation contracts
  - Synchronization testing

Applying to KUPE

- Communication
  - Define Requirements
  - Additional diagram to identify the communication between threads
  - Integration testing
  - System testing

- Load balancing
  - Define requirements
  - Add an information to the report for analyzing the load to each thread
  - Performance testing

Q&A