Distributed Systems
Workshop 4: Cordination

Semester II 2025
Sep 17, 2025
Prof. Francisco Hidrobo

Coordination

Groups

This workshop will be done in groups of two students.

Part 1. Coordination through UTC Server (only test)

Test a basic program to simulate coordination using a centralized UTC (Universal Time Coordinated) server to synchronize time across nodes.

1. **Client (client-UTC.py)**: Sends a request to the server to retrieve the current UTC time.
    1. Connect to the UTC server.
    2. Retrieve and print the synchronized UTC time.
2. **Server (server-UTC.py)**: Responds with the current system time in UTC format.
    1. Listen on a specific port for client requests.
    2. When a client connects, respond with the system's UTC time.

Part 2. Global Time without a UTC Server

Simulate coordination between nodes using their own system clocks without a centralized UTC server. Each process requests time from its peer nodes, averages the responses, and adjusts its local clock based on the mean time.

1. **Peer Nodes**:

- Each node will send its current system time to the others.
- The time will be adjusted based on the average time reported by all the peers.
- Run for each peer in parallel. Usage of threading and socket (zmq) to simulate distributed peers

Part 3. Vector Clocks

Implement vector clocks to handle causality in a distributed system.

- Every process has a vector clock to track causal relationships.
- Processes send and receive messages, and the vector clocks are updated accordingly.

Part 4. Mutual Exclusion

Implement a basic system where processes request access to a shared resource. Select a deployment option between ‘Central Resource Management Server’ or ‘Distributed with Token Ring approach"

Report: You must send a report with the code, results (screenshots) and discussions (Part 2, 3 and 4)