CSCI 251: Concepts of Parallel and Distributed Systems

Alvin Lin

October 30th, 2017

Coordination and Agreement

Topics:

- Mutual Exclusion
- Election
- Multicast

Mutual Exclusion

Requirements:

- Safety: at any time, only one process has access to a critical section
- Livencess: a process making a request for access to a critical section, will eventually get access
- HB order (happened before order): entry to a critical section is in "happened before" order. A process P_i that makes a request before process P_j should get access before process P_j

Performance:

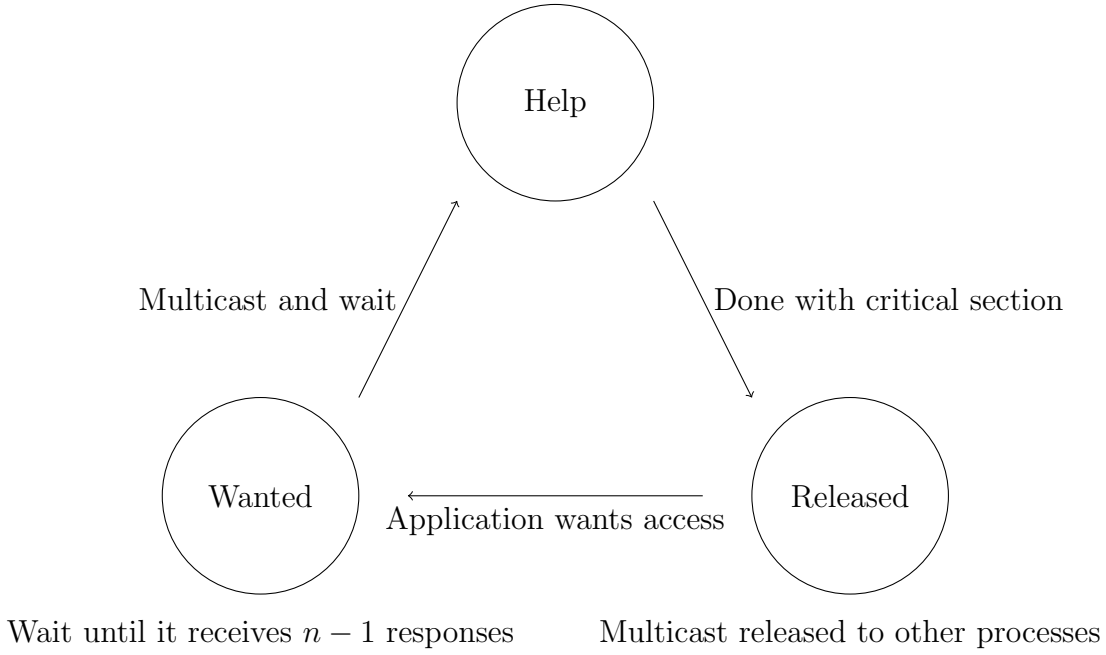
- Bandwidth Consumption: number of messages sent on communication channels
- Client Delay: Amount of time a process waits after making a request
- Throughput: Number of requests granted per unit time

Client Server Mechanisms

A server that receives requests from processes grants access to them one at a time. The server maintains a queue to give access (first in first out).

Multicast

A process can be in one of three states:



Election Algorithm

In distributed systems, it is necessary to select a leader among a set of processes to assign tasks to processes, to make decisions, and decide on variables. Any process in the system can decide whether a leader needs to be elected. This can happen by agreement when process p_i initiates the election process designating p_j as the leader and sends a multicast to all other processes. Other processes can agree or disagree, in which case the processes suggests another leader process and multicasts.

Reminders

Check MyCourses for details on Project 2.
Professor Mohan Kumar:
`mjkvcs@rit.edu`
`https://cs.rit.edu/~mjk`

Rahul Dashora (TA):
`rd5476@mail.rit.edu`

You can find all my notes at `http://omgimanerd.tech/notes`. If you have any questions, comments, or concerns, please contact me at alvin@omgimanerd.tech