Mutual exclusion in distributed system

Mutual exclusion is a concurrency control property which is introduced to prevent race conditions. It is the requirement that a process can not enter its critical section while another concurrent process is currently present or executing in its critical section i.e only one process is allowed to execute the critical section at any given instance of time.

Mutual exclusion in single computer system Vs. distributed system: In single computer system, memory and other resources are shared between different processes. The status of shared resources and the status of users is easily available in the shared memory so with the help of shared variable (For example: Semaphores) mutual exclusion problem can be easily solved. In Distributed systems, we neither have shared memory nor a common physical clock and there for we can not solve mutual exclusion problem using shared variables. To eliminate the mutual exclusion problem in distributed system approach based on message passing is used. A site in distributed system do not have complete information of state of the system due to lack of shared memory and a common physical clock.

Requirements of Mutual exclusion Algorithm:

- **No Deadlock:** Two or more site should not endlessly wait for any message that will never arrive.
- **No Starvation:** Every site who wants to execute critical section should get an opportunity to execute it in finite time. Any site should not wait indefinitely to execute critical section while other site are repeatedly executing critical section
- **Fairness:** Each site should get a fair chance to execute critical section. Any request to execute critical section must be executed in the order they are made i.e Critical section execution requests should be executed in the order of their arrival in the system.
- **Fault Tolerance:** In case of failure, it should be able to recognize it by itself in order to continue functioning without any disruption.

Some points are need to be taken in consideration to understand mutual exclusion fully :

1) It is an issue/problem which frequently arises when concurrent access to shared resources by several sites is involved. For example, directory management where updates and reads to a directory must be done atomically to ensure correctness.
2) It is a fundamental issue in the design of distributed systems.
3) Mutual exclusion for a single computer is not applicable for the shared resources since it involves resource distribution, transmission delays, and lack of global information.

Solution to distributed mutual exclusion: As we know shared variables or a local kernel can not be used to implement mutual exclusion in distributed systems. Message passing is a way to implement mutual exclusion. Below are the three approaches based on message passing to implement mutual exclusion in distributed systems:

1. Token Based Algorithm:

- A unique **token** is shared among all the sites.
- If a site possesses the unique token, it is allowed to enter its critical section
- This approach uses sequence number to order requests for the critical section.
- Each requests for critical section contains a sequence number. This sequence number is used to distinguish old and current requests.
- This approach insures Mutual exclusion as the token is unique

Example : Suzuki–Kasami Algorithm

2. Non-token based approach:

- A site communicates with other sites in order to determine which sites should execute critical section next. This requires exchange of two or more successive round of messages among sites.
- This approach use timestamps instead of sequence number to order requests for the critical section.
- When ever a site make request for critical section, it gets a timestamp. Timestamp is also used to resolve any conflict between critical section requests.
- All algorithm which follows non-token based approach maintains a logical clock. Logical clocks get updated according to Lamport's scheme

Example : Ricart–Agrawala Algorithm

3. Quorum based approach:

- Instead of requesting permission to execute the critical section from all other sites, Each site requests only a subset of sites which is called a **quorum**.
- Any two subsets of sites or Quorum contains a common site.
- This common site is responsible to ensure mutual exclusion

Example : Maekawa's Algorithm

FAQs:

How does the token-based algorithm handle the failure of a site that possesses the token?

What is a quorum in the quorum-based approach, and how is it determined?
A quorum is a subset of sites that a site requests permission from to enter the critical section. The quorum is determined based on the size and number of overlapping subsets among the sites.

How does the non-token-based approach ensure fairness among the sites?
The non-token-based approach uses a logical clock to order requests for the critical section. Each site maintains its own logical clock, which gets updated with each message it sends or receives. This ensures that requests are executed in the order they arrive in the system, and that no site is unfairly prioritized.

Last Updated : 25 May, 2023

32

Previous

Event Ordering in Distributed System

Next

Performance Metrics For Mutual Exclusion Algorithm

Similar Reads

Maekawa’s Algorithm for Mutual Exclusion in Distributed System

Ricart–Agrawala Algorithm in Mutual Exclusion in Distributed System

Lamport's Algorithm for Mutual Exclusion in Distributed System

Suzuki–Kasami Algorithm for Mutual Exclusion in Distributed System

Performance Metrics For Mutual Exclusion Algorithm

Peterson's Algorithm for Mutual Exclusion | Set 1 (Basic C implementation)

Peterson's Algorithm for Mutual Exclusion | Set 2 (CPU Cycles and Memory Fence)

Mutual Exclusion in Synchronization

Distributed System - Thrashing in Distributed Shared Memory

Operating System - Difference Between Distributed System and Parallel System

Article Contributed By :

ihritik
ihritik

Current difficulty : Easy

Easy Normal Medium Hard Expert

Improved By : krishna_97, marcus007

Article Tags : Distributed System , GATE CS , Operating Systems

Improve Article Report Issue

GeeksforGeeks
Sanchhaya Education Private Limited
A-143, 9th Floor, Sovereign Corporate Tower, Sector-136, Noida, Uttar Pradesh - 201305

Company

About Us
Legal
Careers
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program
Apply for Mentor

Explore

Job-A-Thon Hiring Challenge
Hack-A-Thon
GfG Weekly Contest
Offline Classes (Delhi/NCR)
DSA in JAVA/C++
Master System Design
Master CP
GeeksforGeeks Videos

Languages

Python
Java
C++
PHP

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems

SQL

R Language

Android Tutorial

Data Science & ML

Data Science With Python

Data Science For Beginner

Machine Learning Tutorial

ML Maths

Data Visualisation Tutorial

Pandas Tutorial

NumPy Tutorial

NLP Tutorial

Deep Learning Tutorial

Python

Python Programming Examples

Django Tutorial

Python Projects

Python Tkinter

Web Scraping

OpenCV Python Tutorial

Python Interview Question

DevOps

Git

AWS

Docker

Kubernetes

Azure

GCP

DevOps Roadmap

System Design

What is System Design

Monolithic and Distributed SD

High Level Design or HLD

Low Level Design or LLD

Crack System Design Round

System Design Interview Questions

Grokking Modern System Design

NCERT Solutions

Class 12

Top 100 DSA Interview Problems

DSA Roadmap by Sandeep Jain

All Cheat Sheets

HTML & CSS

HTML

CSS

Bootstrap

Tailwind CSS

SASS

LESS

Web Design

Computer Science

GATE CS Notes

Operating Systems

Computer Network

Database Management System

Software Engineering

Digital Logic Design

Engineering Maths

Competitive Programming

Top DS or Algo for CP

Top 50 Tree

Top 50 Graph

Top 50 Array

Top 50 String

Top 50 DP

Top 15 Websites for CP

JavaScript

TypeScript

ReactJS

NextJS

AngularJS

NodeJS

Express.js

Lodash

Web Browser

School Subjects

Mathematics

Class 10

Class 9

Class 8

Complete Study Material

Chemistry

Biology

Social Science

English Grammar

Commerce

Accountancy

Business Studies

Indian Economics

Macroeconomics

Microeconimics

Statistics for Economics

Management & Finance

Management

HR Managament

Income Tax

Finance

Economics

UPSC Study Material

Polity Notes

Geography Notes

History Notes

Science and Technology Notes

Economy Notes

Ethics Notes

Previous Year Papers

SSC/ BANKING

SSC CGL Syllabus

SBI PO Syllabus

SBI Clerk Syllabus

IBPS PO Syllabus

IBPS Clerk Syllabus

SSC CGL Practice Papers

Colleges

Indian Colleges Admission & Campus Experiences

Top Engineering Colleges

Top BCA Colleges

Top MBA Colleges

Top Architecture College

Choose College For Graduation

Companies

IT Companies

Software Development Companies

Artificial Intelligence(AI) Companies

CyberSecurity Companies

Service Based Companies

Product Based Companies

PSUs for CS Engineers

Preparation Corner

Company Wise Preparation

Preparation for SDE

Experienced Interviews

Internship Interviews

Competitive Programming

Aptitude Preparation

Puzzles

Exams

JEE Mains

JEE Advanced

GATE CS

NEET

UGC NET

More Tutorials

Software Development

Software Testing

Product Management

SAP

Write & Earn

Write an Article

Improve an Article

Pick Topics to Write

Share your Experiences