Project Proposal

Concurrency Manager: Two-Phase Locking Protocol with Deadlock Detection

Group Name - ACHA

Group Members:

Harshit Singh (22CS10031)

Arpit Sodhani (22CS30012)

Chandransh Singh (22CS30017)

Aryan Sanghi (22CS30013)

Submission Date: April 15, 2025

Abstract

Concurrency control is a critical component of database systems to ensure data consistency and isolation in multi-transaction environments. This project aims to implement a Concurrency Manager using the Two-Phase Locking (2PL) protocol to manage concurrent transactions while maintaining serializability. A Resource Allocation Graph (RAG) will be employed to detect and handle deadlocks dynamically. Additionally, for bonus credit, the Tree Locking Protocol will be implemented to optimize locking mechanisms in hierarchical data structures. During the demo, detailed logs will be generated for all lock and unlock operations to validate correctness and transparency.

Weekly Work Plan

1. **Week 1 (March 17–March 24):**
   - Set up the development environment and finalize tools and frameworks.
   - Research Two-Phase Locking (2PL) protocol and Resource Allocation Graphs.
   - Divide tasks among group members.
2. **Week 2 (March 25–March 31):**
   - Implement the basic Two-Phase Locking (2PL) protocol, including shared and exclusive locks.
   - Develop a logging mechanism to track lock and unlock operations.
   - Test basic functionality with sample transactions.
3. **Week 3 (April 1–April 7):**
   - Implement the Resource Allocation Graph (RAG) for deadlock detection.
   - Integrate RAG with the Two-Phase Locking protocol.
   - Test deadlock scenarios and verify detection accuracy.
4. **Week 4 (April 8 - April 14):**
   - Implement the Tree Locking Protocol for hierarchical data structures.

- Conduct final testing and debugging to ensure functionality before submission.
- Prepare documentation and demo materials for submission.

Conclusion

This project will provide hands-on experience with concurrency control mechanisms in database systems, focusing on implementing robust protocols like Two-Phase Locking and Tree Locking. The use of a Resource Allocation Graph will enhance our understanding of deadlock detection and resolution techniques. The detailed logging mechanism will ensure transparency in lock management during the demo.

Group Members:

Harshit Singh (22CS10031), Arpit Sodhani (22CS30012), Chandransh Singh (22CS30017), Aryan Sanghi (22CS30013)