DINH DUY KHA

2019712308

Read-Log-Update: A Lightweight Synchronization Mechanism for Concurrent Programming - Paper Critique

Summary

This paper introduces read-log-update (RLU), an extension to read-copy-update (RCU), a commonly used synchronization mechanism. RLU improves on RCU by increasing parallelism, allowing multiple of reads with multiple writers. Evaluations on RLU show that it achieves similar or better performance to RCU with simpler code and is capable of performing tasks RCU could not do.

Strengths

The authors are able to improve on an existing synchronization design, making it simpler for programmers and increase its performances and scalability.

An important optimization is made to RLU by deferring synchronize calls and aggregating write-logs and locks. This helps reduce overall synchronize calls, thus improving performance.

RLU is backward-compatible with RCU, which makes upgrading to this new mechanism simpler.

Weaknesses

RLU could have considerably worse performance than RCU in some cases e.g. hash table test and update-only test, which keeping it from being a perfect replacement for RCU.

Improvement suggestions

RLU only keep two versions of an object. Extending it to keep multiple versions might improve overall write performance.