Operating System Design

2019711610 박찬열

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

This paper introduces read-log-update which is the extension of the read-copy-update synchronization mechanism that supports scalability of concurrent code by allowing unsynchronized sequences of reads to execute concurrently with updates. RCU mechanism provides scalability by allowing read-only traversals to proceed concurrently with updates by creating a copy of the data structure being modified. However, the copy of updates must be installed while there is no readers that reference the copy, it is expensive work on modern multicore systems. RLU replace the copy creation and installation with a clock-based logging mechanism. RLU maintain an object-level write-log per thread and record all modifications to objects in this log which make RLU performs as well as RCU in simple structure and outperforms in complex structure. As the multicore system becomes more important by limitation of increasing single CPU clock, the study that research about synchronization mechanism which has scalability in multi-readers/-writers case, are continuously increasing. This paper benefits in the case of many writers and made researchers handle multi core system more comfortably.