INTERFACE SYSTEM AND METHOD FOR ASYNCHRONOUSLY UPDATING A SHARED RESOURCE

ABSTRACT

Tasks make updates requested by calling tasks to a shared resource serially in a first come first served manner, atomically, but not necessarily synchronously, such that a current task holding an exclusive lock (100) on the shared resource makes the updates on behalf of one or more calling tasks queued on the lock (100). Updates waiting in a queue (102) on the lock (100) to the shared resource may be made while the lock (100) is held, and others deferred for post processing after the lock (100) is released. Some update requests may also, at the calling application's option, be executed synchronously. Provision is made for nested asynchronous locking. Data structures (wait-elements) describing update requests may be queued in a wait queue (102) for update requests awaiting execution by a current task, other than the calling task, currently holding an exclusive lock on the shared resource.

FIG 36