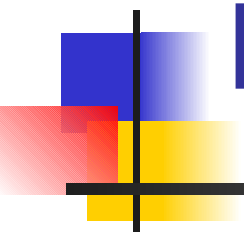
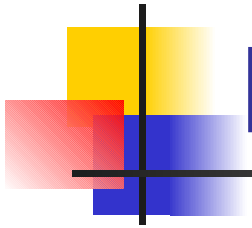


Hierarchical Replication Control

Jiaying Zhang
jiayingz@umich.edu

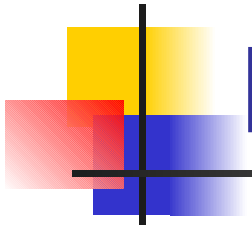
Peter Honeyman
honey@citi.umich.edu

Center for Information Technology Integration
University of Michigan



Motivations

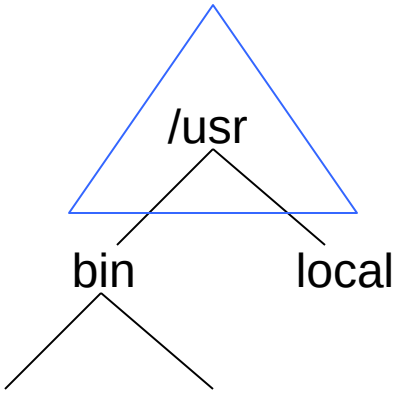
- Emerging global scientific collaborations
  - demand reliable, efficient, and convenient data access on widely distributed data
- Our solution
  - a mutable replicated file system that dynamically elects primary servers to coordinate concurrent writes



Hierarchical Locking

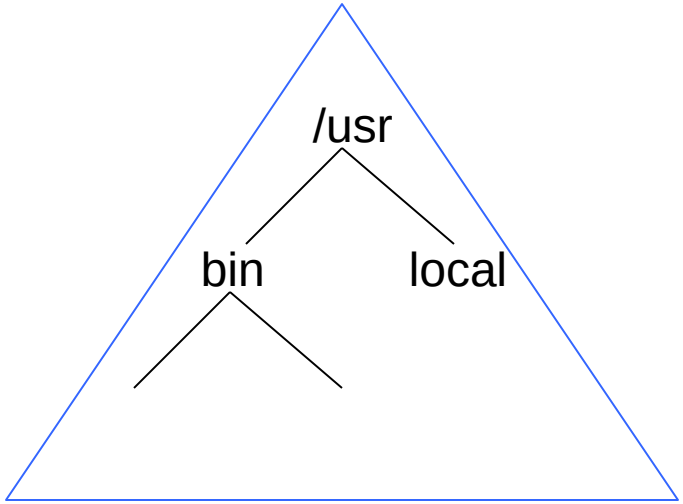
- Primary server election is costly over WAN
- Solution - hierarchical locking
  - a primary server can lock data with different granularities

Two Lock Types – Shallow Lock



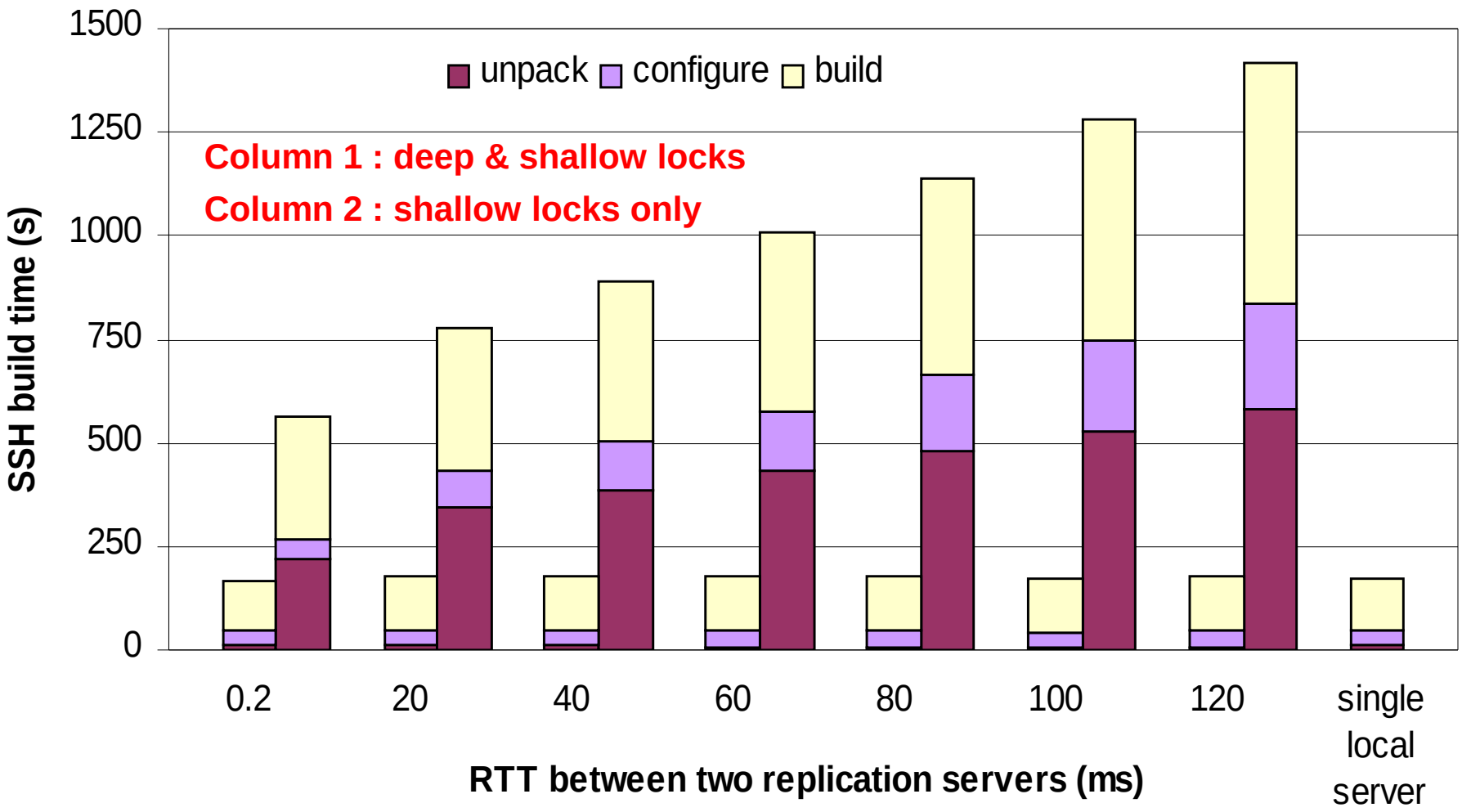
A server with a shallow lock on a file or directory is the primary server for that single object

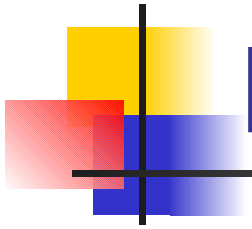
Two Lock Types – Deep Lock



A server with a deep lock on a directory is the primary server for everything in the subtree rooted at that directory

Deep Locks vs. Shallow Locks Only

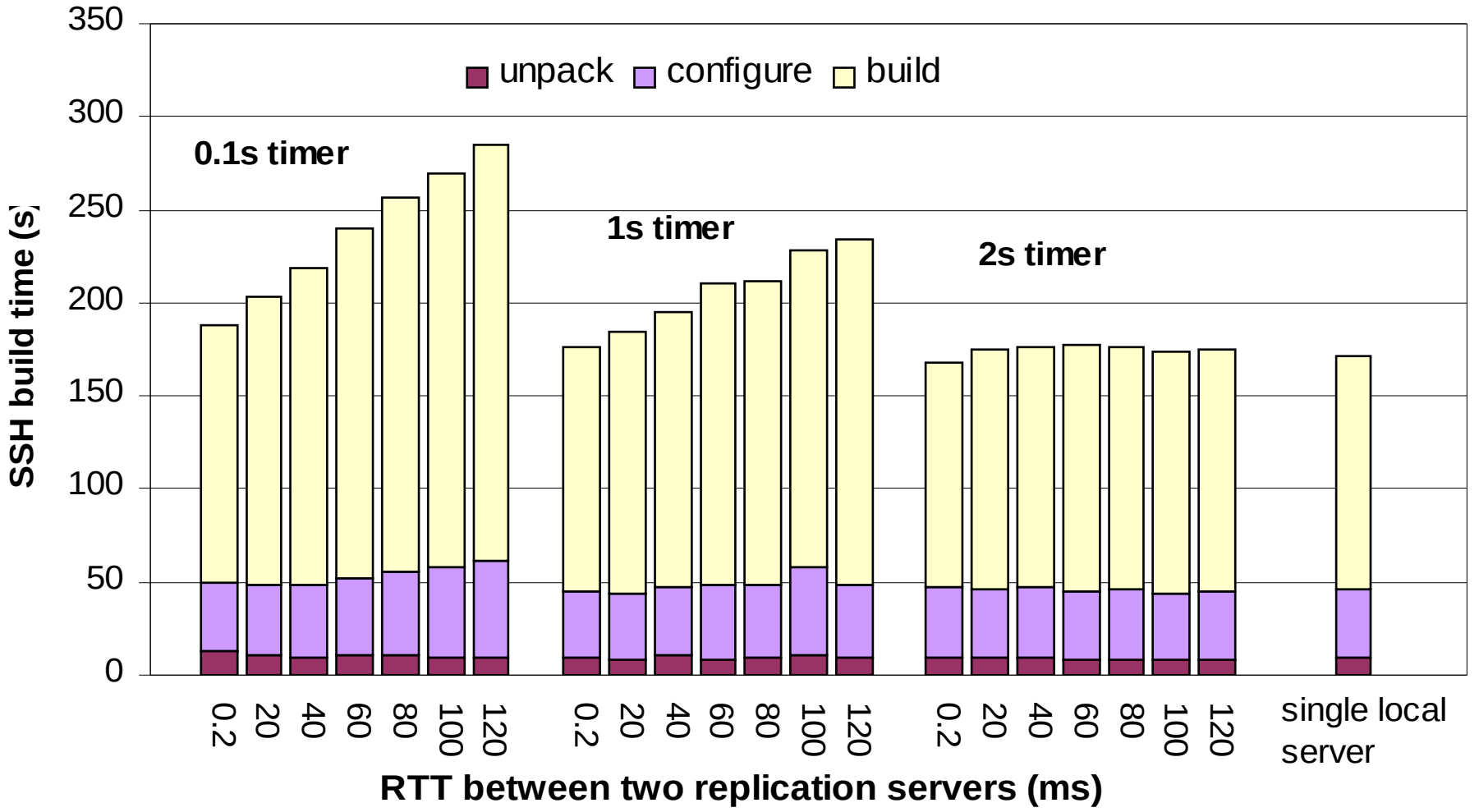


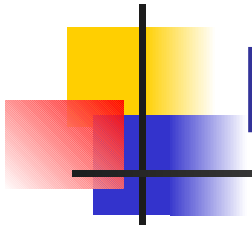


Duration to Hold a Lock

- The primary server associates a timer with a lock to control the duration to hold the lock
  - Restart the timer if a subsequent client update arrives before timeout
  - Relinquish the lock when the timer expires

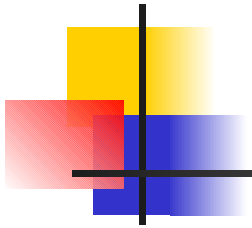
Deep Lock Timer





Future Work

- Optimize the prototype implementation
- Investigate the impact of hierarchical locking on currency and availability
- **Evaluate our design with real distributed applications**



http://www.citi.umich.edu/techreports/reports/citi-tr-06-3.pdf