

CS 423 MP4: Report

Chengyin Liu (liu189), Lin-Ming Hsu (lhsu7), Rohan Sharma (sharma27)

December 2, 2011

Our object structure mirrors the components given in the flowchart, except for that we directly use the hardware monitor for worker thread running. For our workload, we decided to choose matrix-matrix addition, with each job being addition of two integer elements. While this might not be the most efficient design decision, note that load-balancing, and not the workload itself, was our main focus. In fact, the workload is abstracted away by means of generic superclasses.

For networking, we use TCP/IP for convenience and efficiency, and we adopt a client/server architecture for remote and local nodes; also, we have separate runners for each, though the overall design for both are similar.

For load-balancing, we've just considered the CPU utilization and throttling.

We simply use a thread-safe data structure for our job queue.