Prioritized Scalable Distributed Concurrency Services

Goals

Supporting prioritized mutual exclusion on a distributed system by way of implementing the extensions to known protocol for prioritized mutual exclusion for distributed systems. The extensions are meant for intent locking support coupled with read, write and upgrade lock modes.

Current

Changes to the current implementation for keeping the concurrency level constant are done. Experiments are being run for concurrency level C = 5. The effect has to be studied carefully to understand the request latency behavior.

Work Done (During Last Month)

- Protocol implementation on MPI was done. The problem of MP_THREAD_MULTIPLE was solved by not calling mpi_thread_init() (undocumented fact).
- The ratio of inter arrival request time to the critical section length was found useful to study the behavior under changing concurrency level. This is different from the current work mentioned above. Behavior is superlinear at higher lower ratio.
- Experiments on LINUX machines were carried out and comparison with Naimi's fine grain as well as coarse grain was made. Use of network latency simulator was made to have a random network latency with mean value of 150 msec. Previously the coarse grain option wasn't studied. Our protocol has better message overhead behavior. Here ratio=10.
- Submission to IPDPS 2003 with the results obtained by MPI implementation on IBM SP. Experiment was repeated for various ratio values.
- Survey of various approaches to fault tolerant distributed systems and algorithms. Idea was to study the problem in the context of our protocol.
- Poster was prepared for OOPSLA student research and poster sessions. It will be presented at seattle from November 3 – 8. Print should be received by today.
- Problem with the current request latency measurements was the effect of increasing concurrency. Actually the element of interest is the distribution among nodes. The solution might be to keep the concurrency level constant.
- $$\text{Concurrency level } C = \frac{\text{Critical section length}}{\text{Inter arrival time of requests}}$$

  experiments are being run for C = 5 and then measuring the request latencies.

Future Work

- Solution to measuring the latency at constant concurrency level should be found.
- The high level direction now is to run the simulations with different levels of priorities. Priority inversions should be avoided or bounded. The implementation

for prioritized protocol is already done. The results should reflect the preference for prioritized requests.

- Thesis committee should be formed and formalities taken care of.
- Requirements for this semester regarding thesis preparation depend on the advisor.

Nirmit Desai
nvdesai@unity.ncsu.edu
Date: 10-31-2002