Assume that algorithms may be a deadlock, that is, in a series of nodes, because they are waiting for one or more reply, resulting in a loop waiting so that no one node can enter the critical section. In the wait loop, there must be a node Si, its request has the highest priority, and the other waiting nodes priority is lower than it. In the queue, the lower priority nodes are always ranked higher priority nodes behind, they cannot get a reply before the Si, this algorithm does not deadlock.

5. Performance Analysis

Currently distributed mutual exclusion algorithms generally use Maekawa algorithm, the message complexity of the algorithm is O($\sqrt{N}$).In this algorithm message complexity depends on the number of control nodes in each cluster, the message complexity is O(3K)-O(7K)。And in a large distributed system, the k value has nothing to do with the entire system, it depends on the stability of each node in the cluster, the nodes are more stable, k is smaller, else larger. The probability of error in a single node is small, so k is usually a single digit number, does not increase with the increase of the system. Therefore this algorithm message complexity is smaller than fully distributed algorithm one or more orders of magnitude.

The synchronization delay is still 2T, this algorithm does not improve.

In this algorithm, right is determined by control nodes of the node to access a critical section, and the coordinator has all of the information of the system, and the coordinator itself is controlled node, so even if only one right node in the system does not affect the algorithm running. Therefore, the maximum node fault tolerance of the proposed algorithm is N-1.

6. Conclusion

Comprehensive centralized algorithm message complexity is low, and high fault tolerance of distributed algorithms, the paper proposes a distributed mutual exclusion algorithm has higher fault tolerance and smaller communication cost. The algorithm in synchronization delay is not improved, but the message complexity of the algorithm can be reduced by several orders of magnitude, which has an extremely important value to establish large-scale distributed systems based on the Internet. At the same time, the algorithm increase node fault tolerance to N-1. These results show that the algorithm significantly improves the efficiency of distributed mutual exclusion algorithm, but also improve the efficiency of distributed systems, and has a high value in use.

7. References:

[1] Liu Peng. Cloud computing (second edition)[M] Beijing: Publishing House of electronics industry,2011.5

[2] MAEKAWA M. A $\sqrt{N}$ algorithm for mutual exclusion in decentralized systems[J]. ACM Trans Computer Systems，1985，3(2):145-159.

[3] HARADA T, YAMASHRIA M. Transversal merge operation: a non-dominated coterie construction method for distributed mutual exclusion[J]. Parallel and Distributed Systems, IEEE transactions on, 2005, 2(2): 183-192.

[4] LI Mei-an, LIU Xin-song, WANG Zheng. High performance distributed mutual exclusion algorithm based 0n quorum and dynamic token[J], Journal on Communications, 2006(4): 124-130

[5] LAMPORT L. Time clocks and ordering of events in distributed systems[J]. Comm ACM, 1978, 2l(7): 558-565.