Ex6 from GHOSH book chap.9

Consider the algorithm Chandy Misra and Haas with the WFG, Started by node 1

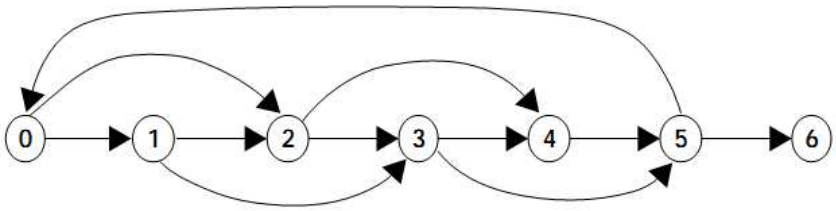

FIGURE 9.6 A wait for graph using the OR-model.

Step 1

P1 send P(1,1,2) , P (1,1,3) to node 2 and 3

Step 2

Node 2 receives P(1,1,2) : set parent = 1 and sends P(1,2,3) and P(1,2,4)

Node 3 receives P(1,1,3) : set parent = 1 and send P(1,3,4) and P(1,3,5)

Step 3 :

Node 3 receives P(1,2,3) -> send ACK to Node 2 (because this is not first probe)

Node 4 receives P(1,2,4) -> set parent = 2 and sents P(1,4,5) to node 5

Node 4 receives P(1,3,4) -> send ACK to Node 3 (because this is not first probe)

Node 5 receives P(1,3,5) -> set parent = 3 and sents P(1,5,0) to node 0 and P(1,5,6) to node 6

Step 4 :

Node 5 receives P(1,4,5) -> sends ACK to node 4

Node 6 receives P(1,4,5) -> ignore this msg because it is computing.

Node 0 receives P(1,5,0) -> set parent = 5 and send P(1,0,1) to node 1 and P(1,0,2) to node 2

Step 5 :

Node 4 receive ACK from node 5 -> send ACK to parent (node 2)

Node 1 receive P(1,0,1) -> send ACK to node 0

Node 2 receive P(1,0,2) -> send ACK to node 0

Step 6 :

Node 0 receives ACK from all successor (1,2)-> send ACK to node 5

Node 2 receives ACK from all successor (4,3)-> send ACK to node 1

Terminate

Conclusion :

node 1 have NOT received ACK from node 3 -> Node 1 will not detect the communication deadlock.

Compare 2 phase commit and 3 phase commit

Reference : http://www.hypergurl.com/blog/databases/two-phase-protocol.html

This article was written by Tanya Puntti for Database Application Development - Central Queensland University.

2 Phase commit

In the two-phase commit the coordinator sends a prepare message to all participants (nodes) and waits for their answers. The coordinator then sends their answers to all other sites. Every participant waits for these answers from the coordinator before committing to or aborting the transaction. If committing, the coordinator records this into a log and sends a commit message to all participants. If for any reason a participant aborts the process, the coordinator sends a rollback message and the transaction is undone using the log file created earlier. The advantages of this are all participants reach a decision consistently, yet independently .

However, the two-phase commit protocol also has limitations in that it is a blocking protocol . For example, participants will block resource processes while waiting for a message from the coordinator. If for any reason this fails, the participant will continue to wait and may never resolve its transaction. Therefore the resource could be blocked indefinitely. On the other hand, a coordinator will also block resources while waiting for replies from participants. In this case, a coordinator can also block indefinitely if no acknowledgement is received from the participant.

3-phase commit protocol

As with the two-phase commit, the three-phase also has a coordinator who initiates and coordinates the transaction. However, the three-phase protocol introduces a third phase called the pre-commit. The aim of this is to 'remove the uncertainty period for participants that have

committed and are waiting for the global abort or commit message from the coordinator' .When receiving a pre-commit message, participants know that all others have voted to commit. If a pre-commit message has not been received the participant will abort and release any blocked resources.