

What to do if ACKs or NACKs get lost?

(Issue from New Orleans meeting)

May 25, 1999

Akihiro Shimura

CANON INC.

DISCONNECT_CONFIRMATION

- **At New Orleans, it was identified that the "DISCONNECT_CONFIRMATION" control function solves a problem with missing ACK on status block write for the "final" ORB.**
- **ANSI NCITS 325-1998(SBP-2) leaves recovery mechanism to higher-level when the target detects a missing acknowledgement after a write to an initiator's status FIFO (See Annex E).**

Usage of ack for the status block

- The acknowledge for the status block should solely be used as a one of events to discard history log of ORB execution that is maintained internally for the recovery in the target because it may not be delivered.
 - I2T queue will work fine because delivery is initiated by signaling ORB and confirmed by status block.
 - Disconnect will work fine by having “DISCONNECT_CONFIRMATION” control function.

Delivery confirmation for T2I queue

- For T2I queue, delivery is initiated by storing status block and confirmed by **receiving ack!**
- Without succeeding ORB for the queue, the delivery will not be confirmed until shutting down the queue.

Confirmation after Missing ack

- **Missing ack will be detected target at first, and then discovered by the initiator by looking into AGENT_STATE.**
 - **Initiator should take appropriate action to the fetch agent.**
- ⇒ **To delivery get confirmed, initiator need to signal succeeding (NOP) ORB to the queue before activating fetch agent if the queue does not have succeeding (pending) ORB.**

Control Information

- **“DISCONNECT_CONFIRMATION” is still valid and necessary because succeeding (NOP) ORB cannot be signaled to the disconnecting queue itself.**
- **Control queue (mix) also need signaling succeeding (NOP) ORB if queue is empty because it contains T2I flow.**
- **(Control Information is never delivered more than once to the management service because it is guaranteed by signature in the same way as other queues have.)**