SETDAT Diagnostic

Data stream definition

Wed, Jul 27, 2005

There is a diagnostic that records RETDAT requests that are received and initialized, but there is not yet one that records SETDAT messages. This note explores the definition of a data stream called SETDLOG that is somewhat analogous to the RETDLOG data stream.

Consider the following 16 byte record to be written by the code in the ACReq task that processes SETDAT messages.

Field	*Size*	*Meaning*
SDNode	2	Source node sending SETDAT message
SDMsgId	2	Source node message id from Acnet header
SDNDev	2	#devices in SETDAT message
SDExTime	2	Execution time to process SETDAT message, in μs
SDFlags	1	Miscellaneous flags
SDTime	7	Time of completion of setting, as mo-da-hr-mn-sc-cy-ms

The SDFlags byte uses the following bit assignments:

7	=1
6	=1
5	=1
4	=1
3	=1 if processed as a server request
2	=1 if message was multicast by source node
1	=1 if message is REQ, =0 if message is USM
0	=1 if dbflag set in SETDAT header

Viewed in hexadecimal, this byte will have the value Fx, obviously not a BCD year. A likely value would be F3, meaning a nonserver request was processed that had the dbflag set that asks for new setting values to be echoed to CDBS via the "DBM..." protocol.

In order to make use of this diagnostic in a node, it is necessary to have a SETDLOG data stream defined in the node. Consider an approach that can do this semi-automatically. At initialization, let the code look for a defined data stream by that name and save its index number. In the case that none exists, check a fixed entry in the DSTRM table to see if it is in use. If it is empty, then define it according to fixed constant values for the name, the base address, and the size. The next time the node is rebooted, it will find the defined entry and use it. What this approach does is eventually define the data stream for all nodes, save those that already had a DSTRM entry in the chosen place. If one wants to define the data stream ahead of time manually, it can be in place (at any DSTRM entry) and ready to go the first time the new system code is booted.

As for the SDNode field, it should be a friendly node number if possible, and certainly not a (dynamically assigned) pseudo node number. If a native node number is available, use that, but at least use the Acnet node number, which must exist since SETDAT is an Acnet protocol.

Usually no cancel is used for a SETDAT message, but if the client times out awaiting an acknowledgment, it will send one. Such occurrences could be recognized by a SDNDev value of zero.