To: X3T10 Membership X3T10/94-247r0
Subject: XOR Command meeting minutes 11/7/94
Date: 94-11-22 02:22:22 EST
From: Jay_Elrod@notes.seagate.com (Jay Elrod)

Attendees:

Gerry Houlder	Seagate
Jay Elrod	Seagate
Brian Davis	Seagate
Jeff Stai	Western Digital I/O Products
John Baudrexl	Fujitsu/Intellistor
Colin Schaffer	Fujitsu/Intellistor
Bob Snively	Sun Microsystems
Paul Hodges	IBM
Stephen Fuld	StorageTek
Bill Hutchison	Hewlett Packard
Charles Monia	Digital Equipment Corp.

Gerry Houlder acted as chairman for the meeting. The issues discussed are summarized below.

1) Redundancy Group Addressing Mode Page - A description of the new mode page to do redundancy group mapping was added to the RAID 5 document and was formally presented at the meeting, along with an alternative more complex mode page configuration. The more complex version allows each drive in a RAID group to have a unique logical block address range for that group, rather than a common range which applies to all drives in the group. Bob Snively (Sun Microsystems) thought the added flexibility of this configuration may bedesirable, based on the way their (Sun's) RAID controllers are designed. Bob also recommended that we restrict use of the Address mode page to those interfaces which require more than 3 bytes for a physical device address (the primary purpose of the table is to provide a greater-than-3-byte device address mechanism).

2) New mode page effect on operation of XDWRITE, REBUILD, and REGENERATE commands - the implications of an address mode page as it relates to thesecommands were discussed, including the concept of requiring the logical block address to be a relative address when using the more complex table configuration described in item 1 above. The various forms of parameters for the REBUILD and REGENERATE commands were also discussed briefly.

3) Possible data corruption problems - Paul Hodges (IBM) presented a scenario where a Regenerate and XDWRITE command issued to separate data drives in a RAID group, addressing the same LBA range, could result in improper data regeneration by the Regenerate command. This would happen if the Regenerate drive issued a Read to the parity drive prior to the XDWRITE target issuing the XPWRITE to that same parity drive, but issued the Read to the XDWRITE target after the XDWRITE was complete. The result is that the Regenerate command would be using old parity (from the parity drive) and new data (from the data drive).

This would result in bad regenerate data. The concensus was that the controller needs to ensure that this situation does not happen. This could be accomplished by preventing an XDWRITE from being issued any time a Regenerate or Rebuild is outstanding for an area which overlaps with the extent of the XDWRITE command (and vice versa). It was pretty much agreed that this is what is already done in existing RAID controllers to prevent the same problem (stripe locking). It was agreed that an implementor's note should be added to the RAID document which advises the controller designer of the need to guard against this possibility of data corruption. Jay Elrod will add this note.

4) 3rd party recovery procedure - It was pointed out prior to the meeting that the 3rd party reservation procedure agreed to last meeting can still result in a problem with previously queued commands starting execution before the error on the parity drive is resolved. Everything from transferring of ACA to not doing anything was discussed. There is some feeling that it may not be necessary to do any 3rd party reserves, transferring of ACA, etc. If this is the case, this issue once again becomes a non-issue. It seems that one question at hand is "Exactly how much is necessary at the host level when it comes to error recovery of an XPWRITE?" For the time being, the concensus is that some sort of REBUILD would probably be appropriate. This still needs thought and discussion.

5) Steve Fuld (STK) pointed out that the NDisk bit in the XDWRITE command may be redundant, due to the addition of the secondary control field value of 11, which specifies that the drive save the data in the buffer for the host. This will be further investigated by Jay Elrod, and the NDisk bit will be removed if Steve's observation is found to be correct.

6) Write caching was discussed, and it was recommended that the RAID document define at what point status is returned for the XPWRITE command when write caching is enabled. It was casually agreed that, with write caching enabled, ending status should probably not be returned for the XPWRITE command until the xor of new and old data has completed, and the result is in the buffer ready for writing to the media. Jay Elrod will add this to the document.

7) The question was raised by Steve Fuld as to whether there needs to be entries for both the Max XPWRITE and Max XDWRITE lengths in the RAID Control mode page. There was no good case made during the meeting for leaving both of these fields in the mode page. Jay Elrod will investigate this further and consolidate the two fields into one field if no reason can be found to keep them both.