T10/02-128r0

Hewlett Packard Corporation
Filton Road, Stoke Gifford
Stoke Gifford
Bristol BS34 8QZ
United Kingdom

Phone +44 117 979 9910
Web www.hp.com

SSC-2 Proposal

Explicit Address Write Sequence Identifier

SSC-2 Proposal

Explicit Address Write Sequence Identifier

Introduction

Background

SSC-2 includes explicit addressing commands. From HP's point of view, these commands justify their existence by providing a greater level of data integrity assurance to backup applications and users. The transition of streaming serial devices from direct connect to SAN and long-haul configurations expands the set of possible order and timing combinations for the delivery of commands and data.

HP believes that the existing model for the explicit write commands (i.e., Erase(16), Write(16), and Write Filemarks(16)) does not guarantee data integrity under a number of conditions.

Objectives

With this proposal, HP seeks to modify the operation of explicit Erase, Write, and Write Filemarks commands, such that the explicit address and other CDB fields (including FCS and LCS) guarantee data integrity in the presence of duplicate, missing, and out of order delivery of tagged write commands by packetized transport layers.

Problems with Current Draft Technical Standard

The draft standard does not address ordering implications for target devices supporting the basic task management model (SAM-2, 7.2):

> b) The device server may reorder the actual processing sequence of tasks in any manner. Any data integrity exposures related to task sequence order shall be explicitly handled by the application client using the appropriate commands.

This omission severely reduces the utility of the explicit command set.

Note: the draft standard also does not address ordering implications for target devices configured for unrestricted reordering (SPC-3, 8.4.6). However, since the Queue Algorithm Modifier field defaults to restricted reordering, this data integrity exposure will only occur after specific intervention by an application client that modifies the Queue Algorithm Modifier value. Since an application client must specifically ask the device to allow the data integrity exposure, it appears reasonable to expect the application clients to manage the exposure through their selection and timing of commands.

The draft standard does not adequately address the implications of multiple application clients that send data from different sequences to a streaming serial device server in an interleaved order. The FCS and LCS bits provide a degree of protection, however they fail in two important cases:

1. The sequences, though different, share overlapping LBAs.

2. If the transport layer loses a command with FCS equal to one and LCS equal to zero or if the transport layer delivers such a command late and out of order, the device server cannot always distinguish that a problem has occurred.

In general these cases will result in data corruption.

Change Concept

Link tagged write sequences and task management such that, for a target device supporting tagged queuing, a simple tagged write command task for LBA *n* shall not transition from dormant to enabled state until a tagged write command task for LBA *n-1* has transitioned from enabled state to ended state. Hence the LBA becomes a task management ordering mechanism. It should act in this way only as an inferior ordering criteria for simple tagged write command tasks under the basic task management model or the more general tagged management model with restricted reordering.

Add a sequence identifier to the command descriptor blocks utilized in tagged write sequences. This sequence identifier allows the device server to unambiguously associate each Erase(16), Write(16), and Write Filemarks(16) command with a specific tagged write sequence. When staring a tagged write sequence, the application client sets the sequence identifier field to zero and FCS equal to one. The device server generates an x_T_L unique sequence identifier for every tagged write command it receives with FCS equal to one. It returns this sequence identifier to the application client for those tagged write commands with FCS equal to one and LCS equal to zero. Subsequent application clients sending tagged write commands for inclusion in this sequence set the sequence identifier field to the value provided by the device server.

Detailed Changes to Draft Technical Standard

Awaiting description.