To: T10 Technical Committee
From: Rob Elliott, HP (elliott@hp.com)
Date: 8 November 2007
Subject: 07-481r1 SBC-3 Mention DIF equals protection information

Revision history

Revision 0 (2 November 2007) First revision

Revision 1 (8 November 2007) Incorporated comments from November 2007 CAP WG - eliminate mention of "end-to-end data protection"

Related documents

sbc3r11 - SCSI Block Commands - 3 (SBC-3) revision 11

03-111 SBC-2 End-to-End Data Protection Proposal (Gerry Houlder, Seagate)

07-373 SSC-3 Tape end-to-end data protection (Kevin Butt, IBM)

Overview

The industry often refers to SBC-3's "protection information" feature by the term "Data Integrity Field (DIF)". A recent example:

> EMULEX, LSI, ORACLE AND SEAGATE COLLABORATE TO REDUCE SYSTEM DOWNTIME WITH GROUNDBREAKING DATA INTEGRITY INITIATIVE
> Companies to extend T10 Data Integrity Field (DIF) standard to enable full end-to-end data integrity for enterprise storage systems
>
> SAN DIEGO, Calif., Storage Networking World - April 18, 2007- Emulex (NYSE:ELX), LSI (NYSE:LSI), Oracle (NASDAQ GS: ORCL) and Seagate (NYSE: STX) today announced the Data Integrity Initiative (DII), an unprecedented technology collaboration that leverages and extends the T10 DIF standard to enable complete end-to-end data integrity for enterprise storage systems.

This term and acronym should appear inside the standard so the correct feature can be found.

The term "end-to-end data protection" should also appear. That was the name of the original proposal for the feature by Seagate, and is the name being used by IBM's current proposal to add a similar feature to SSC-3.

Suggested changes

3.1 Definitions

3.1.39 protection information: Fields appended to each logical block that contain a cyclic redundancy check (CRC), an application tag, and a reference tag. See 4.17.

3.1.xx data integrity field (DIF): Another term for protection information (see 3.1.39).

3.1.45 user data: Data contained in logical blocks that is not protection information.

3.2 Symbols and abbreviations

See table 1 for abbreviations of standards bodies (e.g., ISO). Additional symbols and abbreviations used in this standard include:

Abbreviation Meaning

CDB command descriptor block (see 3.1.7)

CRC cyclic redundancy check (see 3.1.8)

CLIST logical unit certification list (see 3.1.28)

DIF data integrity field (see 3.1.xx)

DLIST data defect list (see 3.1.9)

...

4 Direct-access block device type model

4.1 Direct-access block device type model overview

SCSI devices that conform to this standard are referred to as direct-access block devices. This includes the category of logical units commonly referred to as rigid disks and removable rigid disks. MMC-4 is typically used by CD-ROM devices.

This standard is intended to be used in conjunction with SAM-4, SPC-4, SCC-2, SES-2, and SMC-2.

Direct-access block devices store data for later retrieval in logical blocks. Logical blocks contain user data, may contain protection information accessible to the application client, and may contain additional information not normally accessible to the application client (e.g., an ECC). The number of bytes of user data contained in each logical block is the logical block length. The logical block length is greater than or equal to one byte and should be even. Most direct-access block devices support a logical block length of 512 bytes and some support additional logical block lengths (e.g., 520 or 4096 bytes). The logical block length does not include the length of protection information and additional information, if any, that are associated with the logical block.

The logical block length is the same for all logical blocks on the medium.

Each logical block is stored at a unique LBA, which is either four bytes (i.e., a short LBA) or eight bytes (i.e., a long LBA) in length. The LBAs on a logical unit shall begin with zero and shall be contiguous up to the last logical block on the logical unit. An application client uses commands performing write operations to store logical blocks and commands performing read operations to retrieve logical blocks. A write operation causes one or more logical blocks to be written to the medium. A read operation causes one or more logical blocks to be read from the medium. A verify operation confirms that one or more logical blocks were correctly written and are able to be read without error from the medium.

Logical blocks are stored by a process that causes localized changes or transitions within a medium. The changes made to the medium to store the logical blocks may be volatile (i.e., not retained through power cycles) or non-volatile (i.e., retained through power cycles). The medium may contain vendor-specific

information that is not addressable through an LBA. Such data may include defect management data and other device management information.

4.17 Protection information model

4.17.1 Protection information overview

The protection information model provides for protection of user data while it is being transferred between a sender and a receiver. Protection information is generated at the application layer and may be checked by any object associated with the I_T_L nexus. Once received, protection information is retained (e.g., written to medium, stored in non-volatile memory, or recalculated on read back) by the device server until overwritten. Power loss, hard reset, logical unit reset, and I_T nexus loss shall have no effect on the retention of protection information.

Support for protection information shall be indicated in the PROTECT bit in the standard INQUIRY data (see SPC-4).

If the logical unit is formatted with protection information and the EMDP bit is set to one in the Disconnect-Reconnect mode page (see SPC-4), then checking of the logical block reference tag within a service delivery subsystem without accounting for modified data pointers and data alignments may cause false errors when logical blocks are transmitted out of order.

Protection information is also referred to as the data integrity field (DIF).