ENDL
TEXAS

Date: 12 July 2006
To: T10 Technical Committee
From: Mark Evans, Gerry Houlder, Ralph O. Weber
Subject: SAM-4, et al.: making linked commands obsolete

Introduction

The reasons for linked commands have been overtaken by time and events. Making linked commands obsolete would greatly simplify the SCSI standards. This proposal details the changes required in the current T10 standards projects to make linked commands obsolete. The following revisions of the draft standards in development by T10 were checked:

a) SDI revision 00 (0 instances);
b) SAM-4 revision 05 (17 instances);
c) SPC-4 revision 04 (9 instances);
d) SBC-3 revision 03 (1 instance);
e) SSC-3 revision 02 (0 instances);
f) SMC-3 revision 01 (3 instances);
g) MMC-5 revision 03 (1 instance);
h) SES-2 revision 14 (1 instance);
i) OSD-2 revision 00 (1 instance);
j) BCC revision 00 (0 instances);
k) ADC-2 revision 04 (1 instance);
l) SAS-2 revision 02 (4 instances);
m) FCP-3 revision 04 (8 instances);
n) ADT-2 revision 01 (4 instances); and
o) SAT revision 08 (1 instance).

These instances requiring change were discovered by searching on:

a) linked; and
b) intermediate

Revision 0 of this proposal adds inclusion of linked commands in the list of items that are obsolete in SAM-4, SPC-4, and SBC-4.

Revision 1 contains the changes requested by the July CAP working group.

Change #1, SAM-4r05

1.1 [Scope] Introduction: add “linked commands” to the list of concepts made obsolete by this standard.

Change #2, SAM-4r05

3.1.17 completed command: A command that has ended by returning a status and service response of TASK COMPLETE~~ or LINKED COMMAND COMPLETE~~.

Change #3, SAM-4r05

3.1.31 enabled task state: When in this state a task may complete at any time~~ or is waiting to receive the next command in a series of linked commands~~.

Change #4, SAM-4r05

3.1.49 I/O operation: An operation defined by a~~n unlinked~~ command~~, a series of linked commands~~ or a task management function.

Change #5, SAM-4r05

~~**3.1.59 linked command:** One in a series of commands processed by a single task that collectively make up a discrete I/O operation. In such a series, each command is represented by the same I_T_L_Q nexus, and all, except the last, have the LINK bit in the CDB CONTROL byte set to one.~~

Change #6, SAM-4r05

3.1.63 implicit head of queue: An optional processing model for specified commands wherein the ~~first~~ command in task may be treated as if it had been received with HEAD OF QUEUE task attribute. See 8.2 .

Change #7, SAM-4r05

3.1.122 task: An object within the logical unit representing the work associated with a command~~ or a group of linked commands~~. See 4.11.

Change #8, SAM-4r05

~~**3.1.134 unlinked command:** A command having the LINK bit set to zero in the CDB CONTROL byte that is not the last command in a series of linked commands (see 3.1.59).~~

Change #9, SAM-4r05

4.3 The SCSI client-server model

As shown in figure 6, each SCSI target device contains one or more logical units and provides services performed by device servers and task management functions performed by task managers. A logical unit is an object that implements one of the device functional models described in the SCSI command standards and processes commands (e.g., reading from or writing to the media). Each pending command ~~or series of linked commands~~ defines a unit of work to be performed by the logical unit. Each unit of work is represented within the SCSI target device by a task that may be externally referenced and controlled through requests issued to the task manager.

All requests originate from application clients residing within a SCSI initiator device. An application client is independent of the interconnect and SCSI transport protocol (e.g., an application client may correspond to the device driver and any other code within the operating system that is capable of managing I/O requests without requiring knowledge of the interconnect or SCSI transport protocol). An application client creates one or more application client tasks each of which issues a ~~single~~ command~~, a series of linked commands,~~ or a task management function. Application client tasks are part of their parent application client. An application client task ceases to exist once the command~~, series of linked commands,~~ or task management function ends.

As described in 4.2, each request takes the form of a procedure call with arguments and a status to be returned. An application client may request processing of a command through a request directed to the device server within a logical unit. Each device service request contains a CDB defining the operation to be performed along with a list of command specific inputs and other parameters specifying how the command is to be processed.~~ If supported by a logical unit, a series of linked commands may be used to define an extended I/O operation.~~

A task is an object within the logical unit representing the work associated with a command~~ or series of linked commands~~. A new command ~~or the first in a series of linked commands~~ causes the creation of a task. The task persists until a task complete response is sent or until the task is ended by a task management function or exception condition. For an example of the processing for a ~~single~~ command see 5.7.1.~~ For an example of linked command processing see 5.7.2.~~

Change #10, SAM-4r06

4.5.9 SCSI initiator device class …

[in figure 19 remove the Linked task class]

Change #11, SAM-4r06

4.5.11 Application client task class

An application client task class (see figure 19) shall be substituted with:

a) task class (see 4.5.12); or
b) task management function class (see 4.5.13). ~~; or~~
~~c) linked task class (see 4.5.14).~~

An application client task class is the source for single command, series of linked commands, or single task management function.

Change #12, SAM-4r06

~~4.5.14 Linked task class ...~~

[remove all of subclause 4.5.14]

Change #13, SAM-4r05

4.7.2 SCSI initiator device

.....

Application clients are the sources of application client tasks~~commands and task management functions~~.

An application client task is the source for a ~~single~~ command~~, series of linked commands,~~ or a ~~single~~ task management function.

Change #14, SAM-4r05

4.11 Tasks and task tags

A task is represented by an I_T_L_Q nexus (see 4.12) and is composed of:

a) A definition of the work to be performed by the logical unit in the form of a command~~or a group of linked commands~~;

Change #15, SAM-4r05

5.1 The Execute Command procedure call

.....

Task Attribute: A value specifying one of the task attributes defined in 8.6. SCSI transport protocols may or may not provide the ability to specify a different task attribute for each task (see 8.6.1).~~For a task that processes linked commands, the Task Attribute shall be the value specified for the first command in a series of linked commands. The Task Attribute specified for the second and subsequent commands shall be ignored.~~

.....

Data-In Buffer: A buffer to contain command specific information returned by the logical unit by the time of command completion. The **Execute Command** procedure call shall not return a status of GOOD~~,~~ or CONDITION MET~~, INTERMEDIATE, or INTERMEDIATE-CONDITION MET~~ unless the buffer contents are valid. The application client shall treat the buffer contents as invalid unless the command completes with a status of GOOD~~,~~ or CONDITION MET~~, INTERMEDIATE, or INTERMEDIATE-CONDITION MET~~. While some valid data may be present for other values of status, the application client should rely on additional information from the logical unit (e.g., sense data) to determine the state of the buffer contents. If the command ends with a service response of SERVICE

DELIVERY OR TARGET FAILURE, the application client shall consider this argument to be undefined.

.....

TASK COMPLETE: A logical unit response indicating that the task has ended. The Status argument shall have one of the values specified in 5.3~~ other than INTERMEDIATE or INTERMEDIATE-CONDITION MET~~.

~~**LINKED COMMAND COMPLETE:** Logical unit responses indicating that the task has not ended and that a linked command has completed successfully. As specified in 5.3, the Status argument shall have a value of INTERMEDIATE or INTERMEDIATE-CONDITION MET.~~

.....

The SCSI transport protocol events corresponding to a response of TASK COMPLETE~~, LINKED COMMAND COMPLETE~~ or SERVICE DELIVERY OR TARGET FAILURE shall be specified in each SCSI transport protocol standard.

~~An application client requests processing of a linked command by setting the LINK bit to one in the CDB CONTROL byte as specified in 5.2. The task attribute and task priority are determined by the Task Attribute argument and the Task Priority argument, respectively, specified for the first command in the series of linked commands. Upon receiving a response of LINKED COMMAND COMPLETE, an application client may issue the next command in the series through an **Execute Command** procedure call having the same I_T_L_Q nexus and omitting the Task Attribute argument. If the logical unit receives the next command in a series of linked commands before completing the current command in that linked command series, the overlapped command condition described in 5.8.3 shall result.~~

Change #16, SAM-4r05

5.2 Command descriptor block (CDB)

.....

Table 23 – CONTROL **byte:** ~~LINK~~Obsolete

.....

All SCSI transport protocol standards shall define as mandatory the functionality needed for a logical unit to implement the NACA bit~~ and LINK bit~~.

.....

~~The LINK bit is used to continue the task across multiple commands. Support for the LINK bit is optional. The application client sets the LINK bit to one to specify a request for continuation of the task across two or more commands. If the LINK bit is set to one and the command completes successfully, a logical unit that supports the LINK bit shall continue the task and return a status of INTERMEDIATE or INTERMEDIATE-CONDITION MET and a service response of LINKED COMMAND COMPLETE (see 5.3). If the LINK bit is set to one and the logical unit does not support linked commands, the command shall be terminated with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.~~

Change #17, SAM-4r05

5.3.1 Status codes

The status codes are specified in table 24. Status shall be sent from the device server to the application client whenever a command ends with a service response of TASK COMPLETE~~ or LINKED COMMAND COMPLETE~~.

.....

Table 24 – Status codes

10h ~~INTERMEDIATE~~Obsolete ~~No LINKED COMMAND COMPLETE~~

14h ~~INTERMEDIATE-CONDITION MET~~Obsolete ~~No LINKED COMMAND COMPLETE~~

.....

~~**INTERMEDIATE.** This status or INTERMEDIATE-CONDITION MET shall be returned for each successfully completed command in a series of linked commands (except the last command), unless the command is terminated with CHECK CONDITION, RESERVATION CONFLICT, TASK SET FULL, or BUSY status. If INTERMEDIATE or INTERMEDIATE-CONDITION MET status is not returned, the series of linked commands is terminated and the task is ended. This status is the equivalent of GOOD status for linked commands.~~

~~**INTERMEDIATE-CONDITION MET.** The use of this status is limited to commands for which it is specified (see the PRE-FETCH commands in the SBC-2 standard), unless the command is terminated with CHECK CONDITION, RESERVATION CONFLICT, TASK SET FULL, or BUSY status. If INTERMEDIATE or INTERMEDIATE-CONDITION MET status is not returned, the series of linked commands is terminated and the task is ended.~~

Change #18, SAM-4r05

5.3.2 Retry delay timer codes

.....

Table 25 – Retry delay timer

~~INTERMEDIATE 0000h - FFFFh Reserved~~

~~NTERMEDIATECONDITION MET 0000h - FFFFh Reserved~~

Change #19, SAM-4r05

5.3.3 Status precedence

.....

~~D) INTERMEDIATE;~~

~~E) INTERMEDIATE-CONDITION MET;~~

Change #20, SAM-4r05

5.5 Task and command lifetimes

This subclause specifies the events delimiting the beginning and end (i.e., lifetime) of a task or pending command from the viewpoint of the device server and application client.

The device server shall create a task upon receiving a **SCSI Command Received** indication~~ unless the command represents a continuation of a linked command as described in 5.1~~.

.....

To the application client, the command is pending from the time it calls the **Send SCSI Command** SCSI transport protocol service until one of the responses described in this subclause~~ or a service response of LINKED COMMAND COMPLETE~~ is received.

....

Unless a command completes with a GOOD~~,~~ or CONDITION MET~~, INTERMEDIATE, or INTERMEDIATE-CONDITION MET~~ status the degree to which the required command processing has been completed is vendor specific.

Change #21, SAM-4r05

5.7 Command processing example~~s~~

~~**5.7.1 Unlinked command example**~~

A~~n unlinked~~ command is used to show the events associated with the processing of a single device service request (see figure 33). This example does not include error or exception conditions.

.....

~~**5.7.2 Linked command example**~~

[delete this clause including figure 34.]

Change #22, SAM-4r05

8.7 Task priority

.....

~~For a task that processes linked commands, the task priority shall be that specified for the first command in the series of linked commands. The task priority specified for the second and subsequent commands shall be ignored.~~

Change #23, SPC-4r04

1 Scope: add “linked commands” to the list of items made obsolete by this standard.

Change #24, SPC-4r04

~~**3.1.54 linked command:** One in a series of SCSI commands processed by a single task that collectively make up a discrete I/O operation. A detailed definition of a linked command may be found in SAM-3.~~

Change #25, SPC-4r04

3.1.117 task: An object within a logical unit that represents the work associated with a command~~ or a group of linked commands~~. A detailed definition of a task may be found in SAM-3.

Change #26, SPC-4r04

3.1.25 deferred error: A CHECK CONDITION status and sense data that is returned as the result of an error or exception condition that occurred during processing of a previous command for which GOOD~~,~~ or CONDITION MET~~, INTERMEDIATE, and INTERMEDIATE-CONDITION MET~~ status has already been returned. See 4.5.5.

Change #27, SPC-4r04

4.5.5 Deferred errors

Response codes 71h and 73h (deferred error) indicate that the sense data returned is the result of an error or exception condition that occurred during processing of a previous command for which GOOD~~,~~ or CONDITION MET~~, INTERMEDIATE, and INTERMEDIATE-CONDITION MET~~ status has already been returned. Such commands are associated with the use of the immediate bit and with some forms of caching. Device servers that implement these features shall implement deferred error reporting.

Change #28, SPC-4r04

Table 269 – Method of reporting informational exceptions (MRIE) field

Conditionally generate recovered error: The device server shall report informational exception conditions, if the reporting of recovered errors is allowed, by returning a CHECK CONDITION status. If the TEST bit is set to zero, the status may be returned after the informational exception condition occurs on any command for which GOOD status ~~or INTERMEDIATE status~~ would have been returned. If the TEST bit is set to one, the status shall be returned on the next command received on any I_T nexus that is normally capable of returning an informational exception condition when the test bit is set to zero. The sense key shall be set to RECOVERED ERROR and the additional sense code shall indicate the cause of the informational exception condition.

Unconditionally generate recovered error: The device server shall report informational exception conditions, regardless of whether the reporting of recovered errors is allowed, by returning a CHECK CONDITION status. If the TEST bit is set to zero, the status may be returned after the informational exception condition occurs on any command for which GOOD status ~~or INTERMEDIATE status~~ would have been returned. If the TEST bit is set to one, the status shall be returned on the next command received on any I_T nexus that is normally capable of returning an informational exception condition when the TEST bit is set to zero. The sense key shall be set to RECOVERED ERROR and the additional sense code shall indicate the cause of the informational exception condition.

Generate no sense: The device server shall report informational exception conditions by returning a CHECK CONDITION status. If the TEST bit is set to zero, the status may be returned after the informational exception condition occurs on any command for which GOOD status ~~or INTERMEDIATE status~~ would have been returned. If the TEST bit is set to one, the status shall be returned on the next command received on any I_T nexus that is normally capable of returning an informational exception condition when the TEST bit is set to zero. The sense key shall be set to NO SENSE and the additional sense code shall indicate the cause of the informational exception condition.

Change #29, SPC-4r04

5.6.1 Persistent Reservations overview

.....

A~~n unlinked~~ command shall be checked for reservation conflicts before the task containing that command enters the enabled task state. ~~The reservation state as it exists when the first command in a group of linked commands enters the enabled task state shall be used in checking for reservation conflicts for all the commands in the task.~~ Once a task has entered the enabled task state, the command ~~or commands~~ comprising that task shall not be terminated with a RESERVATION CONFLICT due to a subsequent reservation. ~~Any command in a group of linked commands that changes the reservation state shall be the last command in the group.~~

Change #30, SPC-4r04

6.4.2 Standard INQUIRY data

.....

Table 81 – Standard INQUIRY data format: ~~LINKED~~Obsolete

.....

~~A linked command (LINKED) bit set to one indicates that the device server supports linked commands (see SAM-3). A LINKED bit set to zero indicates the device server does not support linked commands.~~

Change #31, SPC-4r04

NOTE 25 - In a system employing multiple application clients, a buffer may be altered between the WRITE BUFFER and READ BUFFER commands by another application client. Buffer testing applications should ensure that only a single application client is active. Use of reservations to all logical units on the device ~~or linked commands~~ may be helpful in avoiding buffer alteration between these two commands.

Change #32, SPC-4r04

8.3.1.7 Verifying access rights

.....

~~All the linked commands in a single task shall be processed based on the ACL that is in effect when the task first enters the task enabled state.~~ Relationships between access controls and tasks in a task set are described in 8.3.1.11.1.

Change #33, SBC-3r03

1 Scope: add “linked commands” to the list of concepts made obsolete by this standard.

Change #34, SBC-3r03

5.3 PRE-FETCH (10) command

.....

If the IMMED bit is set to zero and the specified logical blocks were successfully transferred to the cache, then the device server shall return~~: a)~~ CONDITION MET status~~ if the LINK bit is set to zero in the CONTROL byte (see SPC-4); or b) INTERMEDIATE-CONDITION MET status if the LINK bit is set to one~~.

If the IMMED bit is set to zero and the cache does not have sufficient capacity to accept all of the specified logical blocks, then the device server shall transfer to the cache as many of the specified logical blocks that fit. If these logical blocks are transferred successfully, then the device server~~it~~ shall return~~: a)~~ GOOD status~~ if the LINK bit is set to zero in the CONTROL byte (see SPC-4); or b) INTERMEDIATE status if the LINK bit is set to one~~.

If the IMMED bit is set to one and the cache has sufficient capacity to accept all of the specified logical blocks, then the device server shall return~~: a)~~ CONDITION MET status~~ if the LINK bit is set to zero in the CONTROL byte (see SPC-4); or b) INTERMEDIATE-CONDITION MET status if the LINK bit is set to one~~.

If the IMMED bit is set to one and the cache does not have sufficient capacity to accept all of the specified logical blocks, then the device server shall return~~: a)~~ GOOD status~~ if the LINK bit is set to zero in the CONTROL byte (see SPC-4); or b) INTERMEDIATE status if the LINK bit is set to one~~.

Change #35, SMC-3r01

~~3.1.16 linked command: One in a series of SCSI commands executed by a single task. A detailed definition of a linked command may be found in SAM-2.~~

Change #36, SMC-3r01

3.1.36 task: An object within a logical unit that represents the work associated with a command~~ or a group of linked commands~~. A detailed definition of a task may be found in SAM-2.

Change #37, SMC-3r01

6.3 Commands allowed in the presence of various reservations

.....

A~~n unlinked~~ command shall be checked for reservation conflicts before the task containing that command enters the enabled task state. ~~The reservation state as it exists when the first command in a group of linked commands enters the enabled task state shall be used in checking for reservation conflicts for all commands in the task.~~ Once a task has entered the enabled task state, the command ~~or commands~~ comprising that task shall not be terminated with a RESERVATION CONFLICT due to a subsequent reservation.~~ Any command in a group of linked commands that changes the reservation state shall be the last command in the group.~~

Change #38, MMC-6r00

Table 319 – INQUIRY Data for ATAPI and USB Drives

.....

~~LINKED 0b~~ [remove this entire row of the table]

Change #39, SES-2r14

3.1.30 task: An object within the logical unit representing the work associated with a command~~ or group of linked commands~~. A task consists of one initial connection and zero or more physical or logical reconnections, all pertaining to the task.

Change #40, OSD-2r00

~~4.15 Linked commands~~

~~OSD device servers shall not support linked commands.~~

Change #41, ADC-2r04

3.1.39 task: An object within the logical unit representing the work associated with a command or group of linked commands (see SAM-3). A task consists of one initial connection and zero or more physical or logical reconnections, all pertaining to the task.

Change #42, SAS-2r02

3.1.241 task: An object within the logical unit representing the work associated with a command~~ or group of linked commands~~.

Change #43, SAS-2r02

9.2.6.3.2.3 Processing transport protocol service requests and responses

.....

If this state machine receives a Send Command Complete transport protocol service response from the SCSI application layer with the Service Response argument set to TASK COMPLETE~~ or LINKED COMMAND COMPLETE~~, then this state machine shall send a Request (Send Application Response) message to the ST_TTS state machine specified by the tag. The message shall include the following Application Response arguments:

Change #44, SAS-2r02

Table 153 – Send Command Complete transport protocol service arguments

,,,,,

Specifies the DATAPRES field and STATUS field in the RESPONSE frame:

a) TASK COMPLETE: The DATAPRES field is set to NO_DATA or SENSE_DATA and the STATUS field is set to a Status Code value (see SAM-4)~~other than INTERMEDIATE or INTERMEDIATE-CONDITION MET; b) LINKED COMMAND COMPLETE: The DATAPRES field is set to NO_DATA or SENSE_DATA and the STATUS field is set to INTERMEDIATE or INTERMEDIATE-CONDITION MET~~; or

b) SERVICE DELIVERY OR TARGET FAILURE: The DATAPRES field is set to RESPONSE_DATA and the RESPONSE CODE field is set to INVALID FRAME or OVERLAPPED TAG ATTEMPTED.

Change #45, SAS-2r02

Table 154 – Command Complete Received transport protocol service arguments

.....

From the DATAPRES field and STATUS field in the RESPONSE frame, or from a NAK on the COMMAND frame:

a) TASK COMPLETE: The RESPONSE frame contains a DATAPRES field set to NO_DATA or SENSE_DATA and a STATUS field set to a Status Code value (see SAM-4)~~other than INTERMEDIATE or INTERMEDIATE-CONDITION MET; b) LINKED COMMAND COMPLETE: The RESPONSE frame contains a DATAPRES field set to NO_DATA or SENSE_DATA and a STATUS field set to INTERMEDIATE or INTERMEDIATE-CONDITION MET~~; or

b) SERVICE DELIVERY OR TARGET FAILURE: The RESPONSE frame contains a DATAPRES field set to RESPONSE_DATA and a RESPONSE CODE field set to INVALID FRAME or OVERLAPPED TAG ATTEMPTED, or a NAK was received for the COMMAND frame, or the length of the RESPONSE frame is incorrect.

Change #46, FCP-4r??

3.1.55 SCSI I/O operation: An operation defined by a SCSI command~~, a series of linked SCSI commands,~~ or a task management function. See SAM-3.

Change #47, FCP-4r??

3.1.62 task: An object within the logical unit representing the work associated with a command~~ or group of linked commands~~. See SAM-3.

Change #48, FCP-4r??

4.2 FCP I/O operations

An application client begins an FCP I/O operation when it invokes a Send SCSI Command SCSI transport protocol service request or a Send Task Management Request SCSI transport protocol service request (see SAM-3). The Send SCSI Command SCSI transport protocol service request conveys a single request ~~or a list of linked requests~~ from the application client to the FCP service delivery subsystem.

.....

After all the data has been transferred, the device server transmits the Send Command Complete protocol service response (described in SAM-3) by requesting the transmission of an IU containing the FCP_RSP IU payload. That payload contains the SCSI status and, if the SCSI status is CHECK CONDITION, the autosense data describing the condition. The FCP_RSP IU indicates completion of the SCSI command. If no ~~command linking, ~~error recovery~~,~~ or confirmed completion is requested, the FCP_RSP IU is the final sequence of the Exchange.~~ The device server determines whether additional linked commands are to be performed in the FCP I/O operation. If this is the last or only command processed in the FCP I/O operation, the FCP I/O operation and the Exchange are terminated.~~ If an FCP protocol error occurred during processing of the command, the FCP_RSP IU payload carries the FCP Response information instead of the SCSI status and autosense data.

.....

~~If the command is linked to another command, the FCP_RSP IU payload shall contain the proper status (i.e., INTERMEDIATE or INTERMEDIATE-CONDITION MET) indicating that another command shall be processed. The target FCP_Port shall present the FCP_RSP using the IU that allows command linking, I5 (see 9.1). The initiator FCP_Port shall continue the same Exchange with an FCP_CMND IU, beginning the next SCSI command. All SCSI commands linked in the FCP I/O operation except the last are processed in the manner described above. SAM-3 defines the cases that interrupt and terminate a series of linked commands. In those cases, the FCP_RSP IU of the last command in the set of linked commands shall be transmitted using the IU that does not allow command linking, I4 (see 9.1). See 4.5.~~

Change #49, FCP-4r??

4.5 Confirmed completion of FCP I/O operations

.....

~~If command linking is being performed, the target FCP_Port shall not request confirmed completion for an FCP_RSP IU containing INTERMEDIATE or INTERMEDIATE-CONDITION MET status. The target FCP_Port may request confirmed completion:~~

~~a) when providing the FCP_RSP IU for the last command of the set of linked commands; or~~
~~b) when providing the FCP_RSP IU for a command that terminates linking because of an error or CHECK CONDITION status.~~

Change #50, FCP-4r??

Table 16 - FCP Information Units (IUs) sent to target FCP_Ports

.....

~~T3 Command request (Linked) 6 FCP_CMND M T O~~

~~T4 Command request (Linked) 6 FCP_CMND M H O~~

.....

Notes:

T3, T4, T5, T7, T8, T9, T10, and T11 are obsolete.

T2 ~~and T4 are~~is only permitted when transfer ready IUs are disabled (see table 7).

~~T3 and T4 are only permitted for linked SCSI commands.~~

T2 ~~and T4~~ allows optional sequence streaming during write operations.

Change #51, FCP-4r??

Table 17 - FCP Information Units (IUs) sent to initiator FCP_Ports

.....

I5 Response (~~Linked or c~~Confirm request) 7 FCP_RSP M T O

.....

Notes:

I2, I6, and I7 are obsolete.

I5 is permitted ~~for linked SCSI commands or~~ to request the confirm completion protocol.

I3 allows optional sequence streaming to I3, I4, or I5.

Change #52, FCP-4r??

9.5.1 Overview and format of FCP_RSP IU

.....

~~If command linking is being performed, an FCP_RSP IU is provided for each command. For linked commands, INTERMEDIATE status or INTERMEDIATE - CONDITION MET status indicates successful completion of a command with no other information valid if all other fields are zero. If command linking is requested, the use of the INTERMEDIATE or INTERMEDIATE - CONDITION MET status indicates that linking shall be performed. The LINKED COMMAND COMPLETE or LINKED COMMAND COMPLETE (WITH FLAG) Service Response defined by SAM-3 is implicit in the presentation of INTERMEDIATE or INTERMEDIATE - CONDITION MET status in the FCP_RSP IU.~~

Change #53, FCP-4r??

B.1.11 SCSI linked commands

[Delete this clause.]

Change #54, ADT-2r01

3.1.44 task: An object within the logical unit representing the work associated with a command~~ or group of linked commands~~.

Change #55, ADT-2r01

Table 31 – Send Command Complete transport layer protocol service arguments

.....

Used to set the RESPONSE CODE and STATUS fields:

a) TASK COMPLETE: the RESPONSE CODE field is set to COMMAND OR TASK MANAGEMENT FUNCTION COMPLETE or COMMAND COMPLETE WITH UNIT ATTENTION and the STATUS field is set to a Status Code value (see SAM-4)~~other than INTERMEDIATE or INTERMEDIATE-CONDITION MET; b) LINKED COMMAND COMPLETE: The RESPONSE CODE field is set to COMMAND OR TASK MANAGEMENT FUNCTION COMPLETE or COMMAND COMPLETE WITH UNIT ATTENTION and the STATUS field is set to INTERMEDIATE or INTERMEDIATE-CONDITION MET~~; or

b) SERVICE DELIVERY OR TARGET FAILURE: The RESPONSE CODE field is set to SERVICE DELIVERY FAILURE.

Change #56, ADT-2r01

7.1.4 SCSI Response information unit

.....

The response code value of COMMAND COMPLETE WITH UNIT ATTENTION shall be sent by the remote SMC device server if bridging is enabled and a command completes with a SCSI status of GOOD that results in the generation of a unit attention to initiator ports other than the one that initiated the command. In this case, the SCSI STATUS field shall contain GOOD~~, INTERMEDIATE, or INTERMEDIATE-CONDITION MET~~ and the SCSI AUTOSENSE DATA field shall contain the sense data to be reported to those other initiator ports. Additionally, any data cached by the local SMC device server shall be invalidated (see ADC-2).

Change #57, ADT-2r01

Table 32 – Command Complete Received transport layer protocol service arguments (Sheet 2 of 2)

.....

From the SCSI Response IU RESPONSE CODE and STATUS field, or from a NAK on the SCSI Command IU:

a) TASK COMPLETE: The RESPONSE CODE field is set to COMMAND OR TASK MANAGEMENT FUNCTION COMPLETE or COMMAND COMPLETE WITH UNIT ATTENTION and the STATUS field is set to a Status Code value (see SAM-4)~~other than INTERMEDIATE or INTERMEDIATE-CONDITION MET; b) LINKED COMMAND COMPLETE: The RESPONSE CODE field is set to COMMAND OR TASK MANAGEMENT FUNCTION COMPLETE or COMMAND COMPLETE WITH UNIT ATTENTION and the STATUS field is set to INTERMEDIATE or INTERMEDIATE-CONDITION MET~~; or

b) SERVICE DELIVERY OR TARGET FAILURE: RESPONSE CODE field is set to Service delivery failure.

Change #58, SATr08

3.1.79 task: An object within the logical unit representing the work associated with a command~~ or group of linked commands~~. A task consists of one initial connection and zero or more physical or logical reconnections, all pertaining to the task.

Change #59, SATr08

Table 8 – Standard INQUIRY data fields

.....

~~LINKED The SATL shall set this field to 0 to indicate that this peripheral device does not support linked commands.~~