

T10/01-267r1

Date: October 18, 2001

To: T10 Committee (SCSI)

From: Julian Satran (IBM) (julian\_satran@il.ibm.com), Jim Hafner (IBM) (hafner@almaden.ibm.com)

Subject: Interlocking for exceptions - BUSY, RESERVATION CONFLICT and TASK SET FULL

## 1. Introduction

This proposal picks up where T10/01-219r0 (proposed by Ed Gardner) left off. Revision0 of this proposal amended the (pending?) proposal T10/00-359r2 (also proposed by Ed Gardner). Revision1 (and beyond if necessary) adds additional function built on those of T10/00-359r6 (the last revision) but is not strictly an amendment to that proposal. The "solution" proposed in 00-219 introduced some additional problems and also didn't fully address the stated problem:

The various serial protocols (FCP,SBP, SRP, iSCSI) share a possible problem regarding BUSY, RESERVATION CONFLICT and TASK SET FULL. If a system environment where both command queuing and command ordering are important, there is at present no satisfactory interlock between a command being rejected with one of these status codes and subsequent commands.

This is particularly a concern for iSCSI, which contemplates operating in environments with larger communication latencies than traditional SCSI environments.

[To be fair, the author of that proposal was not 100% supportive, but was performing a public service by bringing the issue and the proposal to T10.]

What complicates the issue is the fact that while we would like all the commands in flight after a command is rejected, to be rejected as well (until explicit repair action is performed at the initiator), we would like to see all the commands already queued make as much progress as possible during this type of interlock.

This proposal defines the following model as a solution to this problem.

After the logical unit encounters one of these conditions (busy, task set full or a reservation is established), all commands already in the task set are handled by the current rules. Note that in the reservation state, new commands may enter the task set; for the other two states, no new commands can enter the task set until the condition changes. After the state changes, if any command has been rejected with the rejecting status, a Unit Attention condition is established for that initiator. If no command has been rejected, no other changes are effected. For RESERVATION CONFLICT the end of the condition is when any reservation is released. For BUSY and TASK SET FULL, the end of the condition is when the device server's resources become available.

[AUTHOR'S NOTE: the text of the proposal has been modified for Revision1 to more clearly coincide with the model of the preceding paragraph. Revision0 was somewhat lax in the wording in this regard (in particular, it didn't address the "change from the exception condition" as the point in time when the unit attention is generated.)]

In short, as viewed by an initiator the condition BUSY, RESERVATION CONFLICT or TASK SET FULL, when cleared, is replaced by a Unit Attention only if the condition caused at least one command from that initiator to terminate.

This new behavior only has value when the unit attention interlock is enabled (see T10/00-359r6). So, this proposal requires T10/00-359r6 as a prerequisite. It adds additional function on top of

those in that proposal; these new functions are optionally enabled only when the function of T10/00-359 are enabled. Revision0 of this proposal had this additional function as mandatory behavior.

[Aside: strictly speaking, the device server may set up a Unit Attention condition as specified here without this proposal, because the device server has the "right" to set up UAs when it feels a need to. However, we feel that it is important to make this behavior standard and predictable.]

The Unit Attention condition is cleared by the mechanism defined in 00-359r6, namely, by a REQUEST SENSE.

## 2. Proposed changes to SAM-2 and SPC-3

[AUTHOR'S NOTE: The proposed changes here add an additional bit to the Control Mode Page next to the UAINLCK bit proposed in T10/00-359r6 and allows this bit to be one only if the UAINLCK bit is one. An alternative would be to join the two bits into a field (perhaps called the UAINLCK field) with values 00b for current behavior, 10b for behavior specified in T10/00-359r6, 11b for 00-359 behavior plus that specified here and 01b as reserved. We consider this only editorial and leave that to the editor.]

### 2.1 In SPC-3, Clause 8.3.6, Control mode page:

Add a bit field called UAEXCPT next to the UAINLCK bit (we suggest UAEXCPT in bit 4 of byte 4 and UAINLCK in bit 5 of byte 4).

Add the following additional text after the paragraph on pg 203 that will be added for UAINLCK bit (which should probably follow the paragraph on the RAC bit).

A unit attention on exception condition (UAEXCPT) bit of one specifies that, after an exception condition is cleared, a unit attention condition shall be generated for an initiator when a command from that initiator was terminated with a status indicating the exception condition, as defined in SAM-2. A UAEXCPT bit of zero specifies that the device server should not generate such a unit attention condition.

If the UAINLCK bit is zero, the UAEXCPT bit shall be zero.

### 2.2 In SAM-2 add a clause 5.3.x as follows:

#### 5.3.x UA generated when an exception condition is cleared

An exception condition occurs when the device server enters a busy state or task set full state or when a reservation is established. The exception condition is cleared when the device server transitions out of the busy or task set full state or when the reservation is released.

When the UAEXCPT bit in the control mode page (see SPC-3) is set to one and the device server clears an exception condition, then the device server shall generate a unit attention condition for each initiator that sent a command that was terminated with a status indicating the exception condition, that is, with a status of BUSY, TASK SET FULL or RESERVATION CONFLICT.

The unit attention condition is generated only once regardless to the number of commands terminated with a status of BUSY, TASK SET FULL or RESERVATION CONFLICT.

The UAEXCPT bit may be set to one only if the UAINTLCK bit is one (see SPC-3). Consequently, this unit attention condition is cleared only by a REQUEST SENSE command that reports this unit attention condition (see 5.8.5). The additional sense code in the REQUEST SENSE parameter data shall be BUSY CONDITION CLEARED, TASK SET FULL CONDITION CLEARED, or RESERVATION CONFLICT CLEARED.

When the UAEXCPT bit in the control mode page is set to zero, the device server should not generate the unit attention condition.

**2.3 In SAM-2 (clause 5.8.5)**

Insert a new item in the list of events that cause Unit Attention between the current h) and i) to read:

x) when the UAEXCPT bit in the control mode page (see SPC3) is set to one and at least one command was terminated with a status of BUSY, RESERVATION CONFLICT or TASK SET FULL (see 5.3.x); or

~~In SPC-3 modify clause 8.3.6 as proposed in 00-359r2 in the definition of the UAINTLCK in the Control Mode Page as follows:~~

~~A unit attention interlock (uaintlck) bit of zero specifies that the logical unit shall clear any unit attention condition reported with autosense or asynchronous event reporting (see SAM-2).—~~

~~A unit attention interlock (uaintlck) bit of one specifies that the logical unit shall not clear any unit attention condition reported with autosense or asynchronous event reporting. In addition, if any command is terminated with a BUSY, RESERVATION CONFLICT or TASK SET FULL status, then a unit attention condition shall be established for the initiator receiving that status (see SAM-2) and the additional sense data shall be set to BUSY CONDITION CLEARED, RESERVATIONS RELEASED, or TASK SET FULL CONDITION CLEARED, respectively.~~

**2.4 In SPC-3, Appendix C, Table C.1**

Add three entries for new ASC/ASCQ values (as suggested by the SPC-3 editor):

2C 07	D T L P W R S O M C A E B K	BUSY CONDITION CLEARED
2C 08	D T L P W R S O M C A E B K	TASK SET FULL CONDITION CLEARED
2C 09	D T L P W R S O M C A E B K	RESERVATION CONFLICT CLEARED

~~[Editor's note: RESERVATIONS RELEASED is already defined (2A/04). The specific values above are just a suggestion, and are subject to the SPC-3 editor's approval or revision.]~~

~~[Editor's note: is there a need to add anything more to the reservation clause concerning this behavior?]~~