Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303-4900

May 19, 1998 T10/98-164, revision 0

To: NCITS Technical Committee T10

From: Bob Snively

Subject: Command behavior under reservations

This document reviews the presently defined behavior of commands encountering a legacy reservation and proposes the proper behavior for commands encountering the various classes of persistent reservation. The attached chart indicates those commands that may be executed in the presence of each type of reservation. If the entry in the chart is blank, a reservation conflict is the expected behavior. This work is based on a proposed chart by Ralph Weber, with additional inputs from the T11 SCSI working group meeting on May 6, 1998.

The chart assumes that extent reservations for both the RESERVE(n) and RELEASE(n) commands and the PERSISTENT RESERVE IN and PERSISTENT RESERVE OUT commands will be made obsolete. For those who are interested in how the obsolete reservations were handled, the shaded column contains the historical reservation conflict definitions. This column will be removed in the next revision of this document. Many of the descriptions are imprecise.

Making the extent reservations obsolete eliminates the following anomalous behavior. Violations of legacy logical unit reservations create a status of RESERVATION CONFLICT. Violations of legacy extent reservations create a status of CHECK CONDITION with a sense key of DATA PROTECT and “with the appropriate additional sense code for the condition if any part of the operation is prohibited by an extent reservation.” The choice of ASC/ASCQ for this case is somewhat unclear. It could be any of the following:

02h LOGICAL UNIT SOFTWARE WRITE PROTECTED

02h OPERATOR SELECTED WRITE PROTECT

00h WRITE PROTECTED

The chart assumes that the CHANGE DEFINITION command will be made obsolete. The row is shaded and will be removed in the next revision of this document. Its reserve behavior is not self-consistent.

For the persistent reservation definitions, logical unit and element reservation violations are treated identically and are expected to create a status of RESERVATION CONFLICT. This is possible because it is simple to identify what conflicting reservations may exist and which controlling host maintains the reservations.

The information in the left three columns of the chart is extracted from SPC-2 and SBC. Tape definitions will be provided in a later document. To fit the necessary information in the chart, some command names have been abbreviated, although those familiar with the SCSI command set should easily be able to identify the commands.

The following chart is a working document intended to provide input for later reservation definitions. Sample wording is provided in the appropriate note. This sample wording is expected to be added to SPC-2 and as an erratum, supplement, or future revision to the SBC document.

The following table lists the acronyms used in the reservation conflict chart.

Table 1: Key to acronyms

Acronym	Meaning
Reservation types and extents	
LU	Logical Unit
EX	Extent
Excl	Exclusive
RO	Registrants Only
RS	Reads Shared
WP	Writes Prohibited
WS	Write Shared
WX	Writes Exclusive
RX	Reads Exclusive
Reg	The initiator performing the operation is a registrant
Not Reg	The initiator performing the operating is not a registrant
Indication of conditions that permit command to be executed in presence of reservations	
A	Allowed
NE	Allowed if no effect on responses to reserving initiator
NL	Allowed if no overlap with reserved extents
ERD	Extent reservation dependent. Command may be allowed if extent reservation type does not conflict with operation to be performed
Grouping of similar command behaviors	
R	Command is treated like READ
W	Command is treated like WRITE
INQ	Command is treated like INQUIRY
AC	Always conflicting
SA	Shared Access
?	Indicates anomalous behavior in legacy reserve. Proposal included below table.

■

Table 2: Chart of commands that may execute in the presence of various reservations

Command	Current		Addressed LU has this type of reservation held by another initiator									Class of Action	Notes
			Read Shared	Write Excl	Read Excl	Excl Access	Shared Access	Reg Write Excl RO	Not Reg Write Excl RO	Reg Excl Access RO	Not Reg Excl Access RO		
	LU	EX	RS WP	RS WX	RX WS	RX WX	RS WS	RS WX	RS WX	RX WX	RX WX		
CHANGE DEF	NE**?**	NE	NE	NE	NE		NE	NE	NE	NE			
COMPARE		ERD	A	A			A	A	A	A		R	
COPY		ERD			A		A	A		A		W	
COPY & VERIFY		ERD			A		A	A		A		W	
INQUIRY	A	A	A	A	A	A	A	A	A	A	A	INQ	3
LOG SELECT		A					A	A		A		SA	4
LOG SENSE	**A**	A	A	A	A	A	A	A	A	A	A	INQ	6
MODE SELECT		NE										AC	5
MODE SENSE		A					A	A		A		SA	4
PREVNT/ALLOW (prevent equal 0)	A		A	A	A	A	A	A	A	A	A	INQ	3, 7
PREVNT/ALLOW (prevent not equal 0)												AC	5, 7
READ BUFFER		A										AC	5
RECEIVE DIAG.		NE					A	A		A		SA	4
REPORT LUNS	A	A	A	A	A	A	A	A	A	A	A	INQ	
REQUEST SENSE	A	A	A	A	A	A	A	A	A	A	A	INQ	
SEND DIAG PC = 1 and page code not equal 0		NE					A	A		A		SA	4, 8
SEND DIAG PC = 0 or page code equal 0		NE										AC	5, 8
TEST UNIT RDY		A										AC	5
WRITE BUFFER												AC	5
FORMAT UNIT							A	A		A		SA	4
LOCK/UNL CACHE		ERD					A	A		A		SA	4

■

Table 2: Chart of commands that may execute in the presence of various reservations

Command	Current		Addressed LU has this type of reservation held by another initiator									Class of Action	Notes
			Read Shared	Write Excl	Read Excl	Excl Access	Shared Access	Reg Write Excl RO	Not Reg Write Excl RO	Reg Excl Access RO	Not Reg Excl Access RO		
	LU	EX	RS WP	RS WX	RX WS	RX WX	RS WS	RS WX	RS WX	RX WX	RX WX		
PRE-FETCH		ERD	A	A			A	A	A	A		R	
READ		ERD	A	A			A	A	A	A		R	1
READ CAPACITY		A	A	A			A	A	A	A		R	
READ DEFCT DATA		A	A	A			A	A	A	A		R	
READ LONG		ERD	A	A			A	A	A	A		R	
REASSIGN BLKS		NL										AC	5, 9
REBUILD	**A**											AC	5, 10
REGENERATE	**A**											AC	5, 10
SEEK		ERD	A	A			A	A	A	A		R	
SET LIMITS		NL	A	A	A	A	A	A	A	A	A	INQ	11
START/STOP UNIT START = 1 and power condition = 0			A	A	A	A	A	A	A	A	A	INQ	12
START/STOP UNIT START = 0 or power condition not 0							A	A		A		SA	4, 12
SYNCH CACHE		ERD			A		A	A		A		W	
VERIFY		ERD	A	A			A	A	A	A		R	
WRITE		ERD			A		A	A		A		W	
WRITE & VERIFY		ERD			A		A	A		A		W	
WRITE LONG		ERD			A		A	A		A		W	
WRITE SAME		ERD			A		A	A		A		W	
XDREAD		ERD	A	A			A	A	A	A		R	
XDWRITE		ERD			A		A	A		A		W	
XDWRITE EXT		ERD			A		A	A		A		W	
XPWRITE		ERD			A		A	A		A		W	

■

Notes:

1. This is the defining entry for all READ type reservation commands. The following sample wording is suggested for all READ type commands.

   If the logical unit is reserved by a RESERVE(6) or a RESERVE(10) command, a reservation conflict shall occur when a [insert read type command name here] command is received from an initiator other than the one holding a logical unit reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Read Exclusive or Exclusive Access, a reservation conflict shall occur when a [insert read type command name here] command is received from an initiator other than the one holding the reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Exclusive Access Registrants Only, a reservation conflict shall occur when a [insert read type command name here] command is received from an initiator that is not registered. If a reservation conflict occurs, the command shall be rejected with RESERVATION CONFLICT status, no data transfer shall take place, and no change in the storage media's state or content shall take place.

2. This is the defining line for all WRITE type reservation commands. The following sample wording is suggested for all WRITE type commands.

   If the logical unit is reserved by a RESERVE(6) or a RESERVE(10) command, a reservation conflict shall occur when a [insert write type command name here] command is received from an initiator other than the one holding a logical unit reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Read Shared, Write Exclusive, or Exclusive Access, a reservation conflict shall occur when a [insert write type command name here] command is received from an initiator other than the one holding the reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Exclusive Access Registrants Only or Write Exclusive Registrants Only, a reservation conflict shall occur when a [insert write type command name here] command is received from an initiator that is not registered. If a reservation conflict occurs, the command shall be rejected with RESERVATION CONFLICT status, no data transfer shall take place, and no change in the storage media's state or content shall take place.

3. This is the defining command line for all INQUIRY type commands. The following sample wording is suggested for all INQUIRY type commands.

   The [insert name of inquiry type command here] command shall not be affected by reservations or persistent reservations.

4. These commands are considered shared access type commands. The following sample wording is suggested for all shared access type commands.

   If the logical unit is reserved by a RESERVE(6) or a RESERVE(10) command, a reservation conflict shall occur when a [shared access type command name here] command is received from an initiator other than the one holding a logical unit reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Read Shared, Write Exclusive, Read Exclusive, or Exclusive Access, a reservation conflict shall occur when a [insert shared access type command name here] command is received from an initiator

■

other than the one holding the reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Exclusive Access Registrants Only or Write Exclusive Registrants Only, a reservation conflict shall occur when a [insert shared access type command name here] command is received from an initiator that is not registered. If a reservation conflict occurs, the command shall be rejected with RESERVATION CONFLICT status, no data transfer shall take place, and no change in the storage media's state or content shall take place.

5. These commands always will cause a reservation conflict if any reservation exists. The following sample wording is suggested for all always conflicting type commands.

   The [insert name of always conflicting command here] command shall always cause a reservation conflict if any reservation or persistent reservation be affected by reservations or persistent reservations. If a reservation conflict occurs, the command shall be rejected with RESERVATION CONFLICT status, no data transfer shall take place, and no change in the storage media's state or content shall take place.

6. The reservation behavior of the LOG SENSE command is presently defined by the following text:

   If reservations are active, they shall affect the execution of the LOG SENSE command as follows. A reservation conflict shall occur when a LOG SENSE command is received from an initiator other than the one holding a logical unit reservation. The LOG SENSE command shall not be affected by extent or element reservations.

   This text would be changed to read as follows:

   The LOG SENSE command shall not be affected by reservations or persistent reservations.

7. The PREVENT ALLOW MEDIA REMOVAL behaves in such a manner that any initiator can allow permission to remove media, but only the initiator holding a reservation can prevent the removal of media. The actual removal process requires all initiators to agree that media can be removed before media removal can take place. The sample wording is obviously the proper combination of note 5 and note 3.

   Note that an interesting additional mechanism for resetting the prevent state might be desirable, as shown in proposal 1.

8. The SEND DIAGNOSTIC command has been separated into two behaviors. The committee requested that the reservation behavior depend upon the page code values included in the data field. If the page codes are unused or if the page code value is 0, the SEND DIAGNOSTIC command shall be treated as always conflicting. If the page codes are used and the page code value is nonzero, the SEND DIAGNOSTIC command shall be treated as shared access, since this allows enclosure services access and standard diagnostic access to the device. This is still a preliminary proposal.

■

9. The REASSIGN BLOCKS command had previously been allowed if no extent reservations overlapped with the blocks to be reassigned. In the absence of extent reservations, I have assumed that this command should operate like a MODE SELECT command, which is always conflicting.

10. The REBUILD and REGENERATE commands had previously been allowed for shared access reservations. I have assumed that these commands should operate like a MODE SELECT command, which is always conflicting. The behavior of these commands with a legacy reservation present is not defined. Reservation conflicts should occur in that case, as specified in proposal 2.

11. The SET LIMITS command had previously been allowed if no extent reservations overlapped with the blocks to be reassigned. Since the SET LIMITS command provides protection in links of commands only, and since the command for which protection is to be forced is not defined, I have assumed that the SET LIMITS command can always be executed without conflict. If the subsequent commands in the link conflict with a reservation, reservation conflict will be presented to that subsequent command only. If the subsequent commands in the link conflict with the limits established by the SET LIMITS command, a CHECK CONDITION with a sense key of DATA PROTECT will be presented. SET LIMITS and command linking are optional and not commonly implemented. See proposal 3.

12. The START STOP UNIT command was reviewed by the committee. The committee proposed that any unit be able to start the device, but that only those units sharing a persistent reservation should be able to stop the device. I have followed this proposal, but added power condition commands to those restricted by reservations. The following text should replace the second paragraph of 6.1.14 in SBC the next time SBC is updated or an annex is published:

    If the logical unit is reserved by a RESERVE command, a reservation conflict shall occur when a START STOP UNIT command is received from an initiator other than the one holding a logical unit reservation. If a persistent reservation is present from any initiator and if a START STOP UNIT command with a POWER CONDITIONS field equal to zero and a START bit equal to one, indicating the device is to be started, no reservation conflict shall occur. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Read Shared, Write Exclusive, Read Exclusive, or Exclusive Access, a reservation conflict shall occur when a START STOP UNIT command with the POWER CONDITIONS field not equal to zero or the START bit equal to zero is received from an initiator other than the one holding the reservation. If the logical unit is successfully reserved by a PERSISTENT RESERVE OUT service action of Exclusive Access Registrants Only or Write Exclusive Registrants Only, a reservation conflict shall occur when a START STOP UNIT command with the POWER CONDITIONS field not equal to zero or the START bit equal to zero is received from an initiator that is not registered. The command shall be rejected with RESERVATION CONFLICT status if a reservation conflict occurs and no power state change shall occur.

■

Proposals:

1. PREVENT ALLOW MEDIA REMOVAL clearing mechanism

   At present, there are only two mechanisms for allowing media removal once one initiator has set the machine to prevent media removal. Either all initiators can release the prevent media removal condition or a hard reset can be performed. In the presence of multiple initiators and persistent reservations, an initiator that is the target of a Preempt service action or a Preempt and Clear service action no longer be considered in the media removal prevention algorithm. One mechanism to perform this would be to insert the following sample paragraph after the second paragraph of section 7.14 in SPC-2.

   The execution of a PERSISTENT RESERVE OUT command with a Preempt service action or a Preempt and Clear service action shall implicitly set the Prevent value to 00b for the initiator or initiators targeted by the service action.

2. Reservation conflict for REBUILD and REGENERATE

   At present, the reservation behavior for the REBUILD and REGENERATE commands is not specified. With the additional text included for persistent reservation, the following text is proposed for a new paragraph for the REBUILD command immediately following table 23, section 6.1.10 of SBC:

   The REBUILD command shall always cause a reservation conflict if any reservation or persistent reservation be affected by reservations or persistent reservations. If a reservation conflict occurs, the command shall be rejected with RESERVATION CONFLICT status, no data transfer shall take place, and no change in the storage media's state or content shall take place.

   With the additional text included for persistent reservation, the following text is proposed for a new paragraph for the REGENERATE command immediately following table 27, section 6.1.11 of SBC:

   The REGENERATE command shall always cause a reservation conflict if any reservation or persistent reservation be affected by reservations or persistent reservations. If a reservation conflict occurs, the command shall be rejected with RESERVATION CONFLICT status, no data transfer shall take place, and no change in the storage media's state or content shall take place.

3. No reservation conflict for SET LIMITS

   At present, the SET LIMITS command always receives RESERVATION CONFLICT status if a legacy reserve is present. I have proposed that the command never conflict for persistent reservations, since it is only the subsequent commands in the link that are meaningfully influenced by the SET LIMITS command or reservations. Assuming we do not take the wiser course of making the SET LIMITS command obsolete, the following wording is proposed to replace the second paragraph of section 6.1.13 of SBC.

   The SET LIMITS command shall not be affected by reservations or persistent reservations.

■