

Date: May 08, 2001

To: NCITS

From: George Penokie (Tivoli)

Subject: IBM Public Review Comments on NCITS 351 (SPC-2)

### **1 Clause 5.5.3.2**

The following paragraph:

The capability of preserving persistent reservations and registration keys across power cycles requires the use of a nonvolatile memory within the SCSI device. Any SCSI device that supports the persist through power loss capability of persistent reservation and has nonvolatile memory that is not ready shall allow the following commands into the task set:

Should be changed to:

The capability of preserving persistent reservations and ~~registration~~ reservation keys across power cycles requires the use of a nonvolatile memory within the SCSI device. Any SCSI device that supports the persist through power loss capability of persistent reservation and has nonvolatile memory that is not ready shall allow the following commands into the task set:

This change is required because there is no such thing as a 'registration key' defined in SPC-2. The correct term is 'reservation key'.

### **2 Clause 5.5.3.4**

The following paragraph:

If a PERSISTENT RESERVE OUT with a REGISTER AND IGNORE EXISTING KEY service action is sent when an established registration key exists, the registration shall be superseded with the specified service action reservation key. If a PERSISTENT RESERVE OUT with a REGISTER AND IGNORE EXISTING KEY service action is sent when there is no established registration key, a new registration shall be established.

Should be changed to:

If a PERSISTENT RESERVE OUT with a REGISTER AND IGNORE EXISTING KEY service action is sent when an established ~~registration~~ reservation key exists, the registration shall be superseded with the specified service action reservation key. If a PERSISTENT RESERVE OUT with a REGISTER AND IGNORE EXISTING KEY service action is sent when there is no established ~~registration~~ reservation key, a new registration shall be established.

This change is required because there is no such thing as a 'registration key' defined in SPC-2. The correct term is 'reservation key'.

### **3 Clause 7.21.4**

The following paragraph:

Superseding reservations is mandatory if the RELEASE(10) command is implemented. An application client that holds a current logical unit reservation may modify that reservation by issuing another RESERVE command to the same logical unit. The superseding RESERVE command shall release the

previous reservation state when the new reservation request is granted. The current reservation shall not be modified if the superseding reservation request cannot be granted. If the superseding reservation cannot be granted because of conflicts with a previous reservation, other than the reservation being superseded, then the device server shall return RESERVATION CONFLICT status.

Should be changed to:

Superseding reservations is mandatory if the RELEASE(10) command is implemented. An application client that holds a current logical unit reservation may modify that reservation by issuing another RESERVE command to the same logical unit. The superseding RESERVE command shall release the previous reservation state when the new reservation request is granted. The current reservation shall not be modified if the superseding reservation request ~~cannot be~~ is not granted. If the superseding reservation cannot be granted because of conflicts with a previous reservation, other than the reservation being superseded, then the device server shall return RESERVATION CONFLICT status.

This change is required because the term 'cannot' is not in the keyword list and therefore has to be removed.

#### **4 Clause 8.4.1**

The following paragraph:

This subclause describes the vital product data page structure and the vital product data pages (see table 167) that are applicable to all SCSI devices. These pages are optionally returned by the INQUIRY command (see 7.3) and contain vendor specific product information about a target or logical unit. The vital product data may include vendor identification, product identification, unit serial numbers, device operating definitions, manufacturing data, field replaceable unit information, and other vendor specific information. This standard defines the structure of the vital product data, but not the contents.

Should be changed to:

This subclause describes the vital product data page structure and the vital product data pages (see table 167) that are applicable to all SCSI devices. These pages are optionally returned by the INQUIRY command (see 7.3). ~~and contain vendor specific product information about a target or logical unit. The vital product data may include vendor identification, product identification, unit serial numbers, device operating definitions, manufacturing data, field replaceable unit information, and other vendor specific information. This standard defines the structure of the vital product data, but not the contents.~~

This change is required because VPD data pages are now defined that go beyond what is currently defined. By deleting the indicated text the currently defined VPD pages become legal.

#### **5 Clause 7.3.1**

The following paragraph:

If the standard INQUIRY data changes for any reason, the device server shall generate a unit attention condition for all initiators (see SAM-2). The device server shall set the additional sense code to INQUIRY DATA HAS CHANGED.

Should be changed to:

If the standard INQUIRY data changes for any reason, the device server shall generate a unit attention condition for all initiators (see SAM-2). The device server shall set the additional sense code to INQUIRY DATA HAS CHANGED. If INQUIRY VPD data changes for any reason, the device server may generate a unit attention condition for all initiators (see SAM-2). The device server shall set the additional sense code to INQUIRY VPD DATA HAS CHANGED.

This change is required to allow a VPD data change to generate a unit attention.

## 6 Clause 7.13.6

The following paragraph:

The READ BUFFER command shall return the same number of bytes of data as received in the prior echo buffer mode WRITE BUFFER command from the same initiator. If a prior echo buffer mode WRITE BUFFER command was not successfully completed the echo buffer mode READ BUFFER command shall terminate with a CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST and the additional sense code to COMMAND SEQUENCE ERROR. If the data in the echo buffer has been overwritten by another initiator the target shall terminate the echo buffer mode READ BUFFER command with a CHECK CONDITION status, the sense key shall be set to ABORTED COMMAND and the additional sense code to ECHO BUFFER OVERWRITTEN.

Should be changed to:

The READ BUFFER command shall return the same number of bytes of data as received in the prior echo buffer mode WRITE BUFFER command from the same initiator. If the allocation length is insufficient to accommodate the number of bytes of data as received in the prior echo buffer mode WRITE BUFFER command, the data returned shall be truncated as described in 4.3.4.6, and this shall not be considered an error. If a prior echo buffer mode WRITE BUFFER command was not successfully completed the echo buffer mode READ BUFFER command shall terminate with a CHECK CONDITION status, the sense key shall be set to ILLEGAL REQUEST and the additional sense code to COMMAND SEQUENCE ERROR. If the data in the echo buffer has been overwritten by another initiator the target shall terminate the echo buffer mode READ BUFFER command with a CHECK CONDITION status, the sense key shall be set to ABORTED COMMAND and the additional sense code to ECHO BUFFER OVERWRITTEN.

The current text for Read Data from echo buffer (1010b) as written implies the number of bytes returned is the same as the number of bytes received in the previous Write echo buffer command and makes no mention of allocation length. A strict reading could interpret this to mean the allocation length is ignored for this mode. The above change clears this up.