

Date: April 26, 2001

To: T10 Committee (SCSI)

From: Jim Hafner (IBM) (hafner@almaden.ibm.com)

Subject: Long Identifiers in SPC-3, SAM-2, SBC-2 and other XOR issues

ABSTRACT:

As noted in T10/00-248r2 ("SBC-2 issues" by Robert Elliot), there are problems with the specification of logical units and targets in some of the XOR commands in SBC-2. Additionally, as noted in T10/00-279r0 ("Large SCSI Device Identifiers" by Ed Gardner), there are limitations in SAM-2 and SPC-3 that fix device identifiers to a maximum of 64bits and these are insufficient for both of the forthcoming iSCSI and SRP protocols. This proposal is aimed at resolving these problems. The next section details some of the existing problems and limitations. The subsequent sections provide an outline of the proposed solution and then the details of that solution.

The specific formats for specification of long names or addresses in parameter data is roughly sketched in this proposal, but is placed within a designed framework (see 3.6.1, especially Table xx5). Editorial and technical proposals in this area are welcome.

Revision 2 incorporates the following changes:

- a) removed the proposed changes to SAM-2 (these changes should be handled by other proposals dealing with the architecture - but we need/expect that in these proposals SCSI Devices and SCSI Ports will get "names")
- b) changed terminology: the term "designation" now refers to a data structure that contains name and optional address information of a particular SCSI device or port
- c) added a specification and discussion about what protocols can/should do when defining their formats for the alias parameter data; specifically, the role or roles that address information plays in this context
- d) updated references to spc2r19 and sam2r16
- e) added a general rule to the alias target descriptor specification that extends to other third-party commands an "atomicity" property for consulting the alias list
- f) changed the wording of the REPORT ALIASES and CHANGE ALIASES command clauses with respect to SCC-2 to match that of SPC-2 rev 19
- g) other editorial changes.

Revision 1 incorporates the following changes:

- a) added a requirement that each initiator shall have its own alias list; no shared lists
- b) added more reserved fields to the structures
- c) put the format for long identifiers (now called designations in Revision 2) into specific protocol documents

- d) proposes different changes to SAM-2 to accommodate SCSI devices having “names” and ports possibly having both names and addresses.
- e) allowed aliases to reference SCSI Device Identifiers (names for SCSI device) as well as SCSI Port Identifiers in the form of names or addresses.

1.0 Summary existing problems and limitations of 64bit identifiers

The following summarizes the current state of affairs:

- a) SBC-2 REBUILD and REGENERATE commands (both 16 and 32 byte versions) use only a 64bit identifier for a third party target device but make no explicit reference to a logical unit (e.g., by LUN) at that target.
- b) SBC-2 XDWRITE EXTENDED (both 16 and 32 byte versions) explicitly restrict their use to LUN0 and provide either a 1 byte third party address format or a pointer to an undefined table format to resolve the address; this has the dual problems of being unspecified (table) and restricting to only LUN0 which does not necessarily have a consistent meaning in the presence of LUN Mapping.
- c) SPC-2 EXTENDED COPY leaves only 16 bytes in a target descriptor for a target identifier; 16 bytes is insufficient for some long identifiers such as IPv6 address with IPport qualifier.
- d) SAM-2 specifies that device identifiers shall be at most 64bits and this is also insufficient to deal with IPv6 style addresses, for example.

2.0 Summary of proposed changes

The following summarizes the proposed changes:

- a) SBC-2: Add language to REBUILD (16) and REGENERATE (16) commands to specify that the LUN value should be zero (that is, add language similar to that in XDWRITE EXTENDED).
- b) SBC-2: Change the parameter data format for the REBUILD (32) and REGENERATE (32) to replace the 8 byte SERVICE DEVICE ADDRESS field with a 32 byte TARGET DESCRIPTOR field that is defined as in the EXTENDED COPY command of SPC-3.
- c) SBC-2: Convert TABLE ADDRESS bit in the XDWRITE EXTENDED (16) and (32) to reserved (it can't have been implemented as the table reference is explicitly undefined).
- d) SBC-2: Define a new XDWRITE EXTENDED (64) which contains a 32 byte TARGET DESCRIPTOR field that is defined as in the EXTENDED COPY command of SPC-3.
- e) SPC-3: Define a new pair of commands called REPORT ALIASES and CHANGE ALIASES that allow an initiator to define a mapping of an 8 byte alias to a longer formatted designation for a SCSI device or port. This proposal uses a new Service Action in each of the MAINTENANCE IN and MAINTENANCE OUT commands for this purpose.

- f) SPC-3: Define a new EXTENDED COPY target descriptor called an Alias Target Descriptor. In this descriptor, the target device is identified by an 8 byte value that should be found in the alias list of the target (as defined by CHANGE ALIASES commands); the logical unit is identified as is done in (all but one) existing target descriptors as either a LUN value or a Proxy Token.
- g) Protocol documents should add specification for the format of the alias designations; some suggestions are given in this proposal.

This would result in the following:

- a) clear up the ambiguity in existing REBUILD and REGENERATE commands;
- b) leverage the extensible and defined third party target descriptors of EXTENDED COPY to the other third party commands (REBUILD, REGENERATE, and XDWRITE EXTENDED);
- c) provide a generic and extensible method to map long name and/or addresses to shorter identifiers that can fit in 32 byte target descriptors (without requiring longer or variable length target descriptors).

We propose the following properties of the alias list and the REPORT ALIASES and CHANGE ALIASES commands.

- a) The alias lists are by logical unit (that is each device server maintains its own set of alias lists).
- b) The device server maintains a separate alias list for each initiator (analogous in this respect to initiator-specific mode pages; however, in this case, this is a requirement). This is different from Revision 0.
- c) The CHANGE ALIASES and REPORT ALIASES commands are not blocked by reservations. This is different from Revision 0. There is no longer a requirement for this since each initiator has its own alias list.
- d) The alias list is volatile and is cleared under any event that resets the logical unit.
- e) A designation for a SCSI device or port in the list can be variable length, up to a maximum of 65532 bytes. The specific format is defined by a protocol code and type code. The protocol code specifies the transport protocol and the type code specifies a format specific to that protocol. [This allows each protocol to have a private name space where they can define additional designation formats without changes to SPC-x.]
- f) The REPORT ALIASES command will always report the requesting initiator's entire alias list. There is no defined mechanism for requesting any single entry in the list (this just cluttered the design with no major functional gain).
- g) For opcodes, we propose new service actions in MAINTENANCE IN and MAINTENANCE OUT (of SCC-2); we also propose that these be documented in SPC-3 in a manner analogous to SET DEVICE IDENTIFIER and REPORT DEVICE IDENTIFIER).

2.1 Validity checking on Device/Port Designations

The intent of this proposal is that a device/port designation be used to specify a unique SCSI Device or SCSI Port. We include the following identifying data in a designation:

- a) one SCSI Device Name or one SCSI Port Name (required),
- b) one or more optional SCSI Port Addresses or transport-specific addresses.

When only Name information is provided, it is assumed that the device server has access to some sort of nameserver function to resolve names to addresses.

There are cases where such a designation may not be “valid” from the point of view of the device server managing the alias list (for example, on behalf of a copy manager). For example, it may be that the device server has no port into the SCSI domain of the particular device. Additionally, it may be that optional addressing information does not correspond to the named entity (that is, the named entity is not addressable using one or any of the addresses given). In this case, we allow the protocol standard to define the conditions under which such designations are valid or invalid. In one case, a protocol may choose to view the addressing information as the “mandatory” path to find the named entity (e.g., this is rule for the FC N_Port with World Wide Name checking target descriptor format). In another case, a protocol may choose to view the addressing information as “hint” to assist the device server in finding a named entity without having to use services (such as nameservers) to resolve names to address, but to use such services if the “hint” turns out to be incorrect.

This proposal specifies that validity checking is only done when the alias list is consulted for the purposes of resolving alias values (for example, when the copy manager needs to resolve an alias in a target descriptor). It is not checked when the REPORT ALIASES or CHANGE ALIASES commands are processed.

3.0 Detailed description of proposed changes

3.1 Changes for SBC-2 (sbc2r02) REBUILD (16) and REGENERATE (16)

Add the underlined words to the indicated paragraph of clause 5.1.13 that deals with the specification of the source device (though this is only the clause for REBUILD (16), it covers the clause for REGENERATE (16) by reference):

The source device address field specifies an ANSI X3.270 SAM compliant target identifier (of no more than 64 bits) of a device that is a data source. The implied LUN at the target shall be zero.

Change the title of Table 37 from “REBUILD and REGENERATE parameter data” to “REBUILD (16) and REGENERATE (16) parameter data”.

3.2 Changes for SBC-2 (sbc2r02) REBUILD (32) and REGENERATE (32)

3.2.1 REBUILD (32), clause 5.1.14

Replace the paragraph immediately following the current Table 39 which reads

See the REBUILD (16) command (5.1.13), Table 40, and SPC-2 for a description of the fields in this command.

with

See the REBUILD (16) command (5.1.13) and SPC-2 for a description of the fields in this command. Tables 40 and 41 define the parameter data format for the REBUILD (32) and REGENERATE (32) commands.

and replace Table 40 with the following two tables and additional text.

[EDITOR'S NOTE: Table 41 as specified in the above paragraph is a new table, so the appropriate renumbering of subsequent tables will have to be performed.

]

Table 40. REBUILD (32) and REGENERATE (32) parameter data

Byte	Bit							
	7	6	5	4	3	2	1	0
0	NUMBER OF SOURCE DESCRIPTORS (x)							
1	RESERVED							
2	MSB							
3	SOURCE DESCRIPTOR /PAD LENGTH							
	LSB							
SOURCE DESCRIPTOR(S) (if any)								
4	SOURCE DESCRIPTOR (first)							
43								
40x-36	SOURCE DESCRIPTOR (last)							
40x+3								
40x+4	PAD, if any (length y)							
40x+y+3								
40x+y+4	MSB							
40x+y+z+3	INTERMEDIATE DATA, if any (length z)							
	LSB							

The SOURCE DESCRIPTOR format is specified in Table 41. All other fields in the parameter data are as defined in 5.1.13.

Table 41. REBUILD (32) and REGENERATE (32) source descriptor format
(32 Byte source device address and 8 Byte LBA version).

Byte	Bit							
	7	6	5	4	3	2	1	0
0	MSB							
31	SOURCE DEVICE ADDRESS (32 bytes)							LSB
32	MSB							
40	SOURCE STARTING LOGICAL BLOCK ADDRESS (8 bytes)							LSB

The SOURCE DEVICE ADDRESS specifies a third party logical unit as a data source. The format of this conforms to one of the target descriptor formats of the EXTENDED COPY command as specified in SPC-2.

[EDITOR'S NOTE: should that be SPC-3, both here and in the paragraph above?]

[EDITOR'S NOTE: Should there be additional language here to describe the rules for when target descriptors cannot be resolved by the target?]

3.2.2 REGENERATE (32), clause 5.1.16

Add the underlined words to the indicated paragraph at the end of 5.1.16.

See the REGENERATE (16) command (5.1.15), Table 40 and Table 41, and SPC-2 for a description of the fields in this command.

[EDITOR'S NOTE: as above, should this be SPC-2 or SPC-3?]

3.3 Changes for SBC-2 (sbc2r02) for XPWRITE EXTENDED (16) and (32)

Convert the TABLE ADDRESS bit in the current Table 63 and Table 64 to Reserved. This bit was used to indicate that the target should consult some table to resolve the SECONDARY ADDRESS field.

[EDITOR'S NOTE: The current text specifically says "The lookup table is reserved for future definition" so changing to Reserved should not be a problem. If that is not the opinion of the committee, then converting to Obsolete would be acceptable.]

Change the paragraph immediately following Table 63 as indicated:

See 4.2.1.9 for reservation requirements for this command. ~~A TABLE ADDRESS bit of zero indicates that~~ The SECONDARY ADDRESS field contains the target identifier of the target that will receive the XOR data transfer. The implied LUN of the secondary target shall be zero. If the transport protocol requires more than one byte for the target identifier ~~and the Table Address bit is zero~~, the SECONDARY ADDRESS field specifies the least significant byte of the secondary target identifier. The

upper bytes of the secondary target identifier shall be equal to the upper bytes of the target identifier of the XDWRITE EXTENDED target.

Delete the paragraph following in that clause that specifies the meaning of the TABLE ADDRESS bit of one.

3.4 Addition to SBC-2 (sbc2r02) of new XDWRITE EXTENDED (64)

In the Service action code assignments Table 1, add an entry for the XDWRITE EXTENDED (64) service action code (0007h) and a reference to a new clause 5.1.38, and fix the reference to existing clause 5.1.39 in the XPWRITE (32) command of Table 1. Insert the following clause as a new clause 5.1.38.

5.1.38 XDWRITE EXTENDED (64) command

The XDWRITE EXTENDED (64) command (see Table xx) requests that the target XOR the data transferred with the data on the medium. The resulting XOR data may be subsequently sent to a secondary device using an XPWRITE (32) command.

Table xx. XDWRITE EXTENDED (64) command.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	OPERATION CODE (7F)							
1	CONTROL							
2	RESERVED							
6								
7	ADDITIONAL CDB LENGTH (38h)							
8	SERVICE ACTION (0007h)							
9								
10	RESERVED		DPO	FUA	DISABLE WRITE	PORT CONTROL		
11	RESERVED							
12	SECONDARY ADDRESS DESCRIPTOR (32 bytes)							
43								
44	LOGICAL BLOCK ADDRESS (8 bytes)							
51								
52	SECONDARY LOGICAL BLOCK ADDRESS (8 bytes)							
59								
60	TRANSFER LENGTH (4 bytes)							
63								

The SECONDARY ADDRESS DESCRIPTOR field contains the logical unit identifier of the logical unit that will receive the XOR data transfer. The format of this field conforms to one of the target descriptor formats of the EXTENDED COPY command as specified in SPC-2.

[EDITOR'S NOTE: as always, should this be SPC-3 both above and below?]

See the XDWRITE EXTENDED (16) command (5.1.36) and SPC-2 for a description of all other fields in this command.

3.5 Addition to SPC-3 of REPORT ALIASES and CHANGE ALIASES commands

[EDITOR'S NOTE: all clause references to SPC-3 are actually from the SPC-2 rev 19 (spc2r19) version as no copy of SPC-3 was available at the time of this writing.]

3.5.1 A clause for REPORT ALIASES command for SPC-3

The following clause should be added to SPC-3.

7.xx REPORT ALIASES command

The REPORT ALIASES command (see Table xx1) requests that the device server send a report of the requesting initiator's alias list. The alias list itself is managed by the initiator via the CHANGE ALIASES command (7.yy). This command is optional for all device types, however, if the CHANGE ALIASES command is supported then the REPORT ALIASES command shall also be supported. As defined in the SCC-2 standard, the REPORT ALIASES command is the REPORT ALIASES service action of the MAINTENANCE IN command. Additional MAINTENANCE IN and MAINTENANCE OUT service actions are defined in SCC-2 and in this standard.

The MAINTENANCE IN service actions defined only in SCC-2 shall apply only to SCSI devices that return a device type of 0Ch or the sccs bit equal to one in their standard INQUIRY data. When a SCSI device returns a device type of 0Ch or the sccs bit equal to one in its standard INQUIRY data, the implementation requirements for the SCC-2 MAINTENANCE IN service actions shall be as specified in SCC-2. Otherwise the MAINTENANCE IN service action definitions and implementation requirements stated in this standard shall apply.

Table xx1. REPORT ALIASES command.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	RESERVED			SERVICE ACTION (0Bh)				
2	RESERVED							
3								
4	SEE SCC-2							
5								
6	MSB							
9	ALLOCATION LENGTH						LSB	
10	RESERVED						see SCC-2	RSRVD
11	CONTROL							

SCC-2 defines specific usage for bytes 4 and 5 and bit 1 in byte 10, however, these fields are reserved for the REPORT ALIASES command as defined in this standard.

The ALLOCATION LENGTH field indicates how much space has been allocated for the returned parameter data. If the length is not sufficient to contain all the parameter data, the first portion of the data shall be returned. This shall not be considered an error. The actual length of the parameter data can be computed from the ADDITIONAL LENGTH field in the parameter data. If the remainder of the parameter data is required, the application client should send a new REPORT ALIASES command with an ALLOCATION LENGTH field large enough to contain all the data.

The format of the returned parameter data is specified in Table xx2. It contains a header that indicates the amount of additional data, and the number of entries in the alias list.

Table xx2. REPORT ALIASES and CHANGE ALIASES parameter list.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	MSB							
3	ADDITIONAL LENGTH (n-4)							LSB
4	RESERVED							
5	RESERVED							
6	MSB							
7	NUMBER OF ALIASES							LSB
8	ALIAS LIST ENTRIES							
n								

The ADDITIONAL LENGTH field specifies the number of bytes following this field.

The NUMBER OF ALIASES field indicates the number of ALIAS LIST ENTRIES. If there are no alias entries in the requesting initiator's alias list, then the ADDITIONAL LENGTH field shall be set to four and the NUMBER OF ALIASES field shall be set to zero.

The ALIAS LIST ENTRIES contains a list of alias entries as specified in Table xx3. Each alias entry contains an 8 byte ALIAS value that the device server associates to the value in the DEVICE/PORT DESIGNATION field. The structure of the DEVICE/PORT DESIGNATION field is described in Table xx4. The number of bytes in each

alias entry can be determined from the length field in the DEVICE/PORT DESIGNATION field.

Table xx3. REPORT ALIASES and CHANGE ALIASES alias entry.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	MSB							
7	ALIAS							LSB
8	MSB							
n	DEVICE/PORT DESIGNATION							LSB

Table xx4. DEVICE/PORT DESIGNATION field format.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	PROTOCOL IDENTIFIER							
1	RESERVED							
2	RESERVED							
3	TYPE CODE							
4	RESERVED							
5	RESERVED							
6	MSB							
7	DESIGNATION LENGTH (n-7)							LSB
8	MSB							
n	DESIGNATION							LSB

The PROTOCOL IDENTIFIER field defines the transport protocol to which the IDENTIFIER applies; the defined values of the PROTOCOL IDENTIFIER field are defined in Table 165 (of SPC-2 rev 19). The TYPE CODE defines the format of the DESIGNATION field with respect to the specific transport protocol as defined in the appropriate protocol standard. The DESIGNATION LENGTH specifies the number of bytes of the DESIGNATION field. This value shall be a multiple of 4. A DESIGNATION value that is not a multiple of 4 bytes long shall be padded in the least significant bytes by zeros to the next largest multiple of 4 bytes long.

The DESIGNATION field should designate a unique SCSI Device or SCSI Port using the following identifying data:

- a) one SCSI Device Name or one SCSI Port Name (required),
- b) one or more optional SCSI Port Addresses or transport-specific addresses.

If a designation identifies a unique SCSI Device or SCSI Port that is within a SCSI domain of the device server, the designation is considered valid.

A designation that can not be resolved according to the rules of the relevant protocol standard to a unique SCSI Device or Port within the SCSI domains of the device server, is considered invalid.

NOTE1: This may happen, for example, if the device server has no ports on the SCSI domain of the named device or port.

A designation with both name and address data may be inconsistent if the named device or port can not be addressed by one or more of the given addresses. In such cases, the designation shall be declared valid or invalid according to the rules specified in the relevant protocol standard.

NOTE2: For example, in FCP both an N_Port and World Wide Name for a SCSI Port may be given in a designation. The protocol may require that the N_Port be that of the named port. In that case, the designation would be invalid. Alternatively, the protocol may view the N_Port as an addressing “hint” for the named FC Port accessible to the device server through a different D_ID. In that case, the designation would be valid and designate the named FC Port.

NOTE3: When only name information is provided in a designation, it is assumed that the device server has access to some service like a nameserver to resolve names to addresses. Access to such a service is protocol and implementation specific.

The device server shall not validate any designation at the time of processing either the REPORT ALIASES or CHANGE ALIASES command. Such validation shall occur only when the device server consults the alias list to resolve an alias to a designation in the context of third-party commands (e.g., EXTENDED COPY).

Table xx5 provides specific suggestions for formats of different designations for some existing protocols. Each protocol document needs to adopt its own specification for these designations.

Table xx5: PROTOCOL IDENTIFIER, TYPE CODE values and DESIGNATION formats.

Protocol Identifier	Protocol Standard	Type Code	Type Description	Format/length
00h	FCP-n	00h	WWPortName	8 bytes
00h	FCP-n	01h	WWNodeName	8 bytes
01h	SPI-n	00h	SCSI Address	decimal address (binary, 2 bytes) + Rsvd (2 bytes)
02h	SSA-SnP	00h	Reserved	
03h	SBP-n	00h	Reserved	
04h	SRP-n	00h	Reserved	
05h	iSCSI-n	00h	iSCSI Name (SCSI Device Name)	Name in UTF-8 format (m bytes + pad)
05h	iSCSI-n	01h	iSCSI Name + IPv4 address + tcpport	Name length=m (binary, 1byte)+ Rsvd (1 byte) + Name in UTF-8 format (m bytes + pad) + Rsvr (2 bytes) + IPv4 address (binary, 4 bytes) + Rsvd (2 bytes) + tcpport (binary, 2 bytes) + Rsvd (2 bytes) + Internt Protocol Number (binary, 2 bytes)
05h	iSCSI-n	02h	iSCSI Name + IPv6 address + tcpport	Name length=m (binary, 1byte) + Rsvd (1byte) + Name in UTF-8 format (m bytes + pad) + Rsvr (2 bytes) + IPv6 address (binary, 16 bytes) + Rsvd (2 bytes) + tcpport (binary, 2 bytes) + Rsvd (2 bytes) + Internt Protocol Number (binary, 2 bytes)
06h-FFh	Reserved	n/a		n/a

[EDITOR'S NOTE: By using a PROTOCOL IDENTIFIER and TYPE CODES, we are allowing the independent protocol documents to extend the naming conventions used in the DESIGNATION field without resorting to changes to SPC-3.]

[EDITOR'S NOTE: Table xx5 above is admittedly not well done. The consensus seems to be that these formats should be defined within each protocol document. The above is just the outline of possible formats.]

[EDITOR'S NOTE: Do we want to define in SPC-3 any protocol-independent formats, e.g., a pure IPv4 format?]

3.5.2 A clause for CHANGE ALIASES command for SPC-3

The following clause should be added to SPC-3.

7.yy CHANGE ALIASES command

The CHANGE ALIASES command (see Table yy1) requests that the device server maintain and make changes to a table of associations between short (8 byte) alias identifiers and (possibly) longer SCSI device or port designations. The alias list can be queried by the initiator via the REPORT ALIASES command

(7.xx). This command is optional for all device types, however, if the REPORT ALIASES command is supported then the CHANGE ALIASES command shall also be supported. As defined in the SCC-2 standard, the CHANGE ALIASES command is the CHANGE ALIASES service action of the MAINTENANCE OUT command. Additional MAINTENANCE IN and MAINTENANCE OUT service actions are defined in SCC-2 and in this standard.

The MAINTENANCE OUT service actions defined only in SCC-2 shall apply only to SCSI devices that return a device type of 0Ch or the SCCS bit equal to one in their standard INQUIRY data. When a SCSI device returns a device type of 0Ch or the SCCS bit equal to one in its standard INQUIRY data, the implementation requirements for the SCC-2 MAINTENANCE OUT service actions shall be as specified in SCC-2. Otherwise the MAINTENANCE OUT service action definitions and implementation requirements stated in this standard shall apply

On successful completion of a CHANGE ALIASES command, the device server shall maintain a mapping or association of each assigned 8 byte alias value to the paired possibly longer SCSI device or port designation. These associations shall be cleared under any event that includes a logical unit reset. Other protocol-specific events may also clear these associations. The device server shall maintain a separate list for each initiator.

A CHANGE ALIASES command may add, change or remove entries from the requesting initiator's alias list. Changes to the alias list indicated by a CHANGE ALIASES command affect only those entries that are specified in the parameter data for the command. Alias list entries not specifically referenced in the parameter data remain unchanged.

NOTE1: An application client may use alias values to reference SCSI devices or ports in third party commands (e.g., the EXTENDED COPY command). The alias list provides a mechanism for eight byte third party identifier fields to reference or indicate a third party device or port whose name or addressing information is longer than eight bytes. For example, an application may use the CHANGE ALIASES command to establish an association between an alias value and a SCSI target device or port designation. Then it may send an EXTENDED COPY command that use the alias target descriptor format (see 7.5.6.z) that includes this alias value. At the completion of the EXTENDED COPY command the application should clear this entry from the device server's alias list by sending a subsequent CHANGE ALIASES command that requests association of the alias value to a null designation as described below.

Table yy1. CHANGE ALIASES command.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	RESERVED			SERVICE ACTION (0Bh)				
2	RESERVED							
3								
4	SEE SCC-2							
5								
6	MSB							
9	PARAMETER LIST LENGTH						LSB	
10	RESERVED						see SCC-2	RSRVD
11	CONTROL							

SCC-2 defines specific usage for bytes 4 and 5 and bit 1 in byte 10, however, these fields are reserved for the CHANGE ALIASES command as defined in this standard.

The PARAMETER LIST LENGTH field specifies the length in bytes of the parameter data that shall be transferred from the application client to the device server. A parameter list length of zero indicates that no data shall be transferred and no changes are requested in the initiator's alias list.

The format of the parameter data is specified in Table xx2. It contains a header that indicates the amount of additional data, the number of alias entries in the parameter data and the list of new, changed or removed entries.

Besides the DEVICE/PORT DESIGNATION formats specified in the applicable protocol standards, an additional null designation format may be used in the parameter data of the CHANGE ALIASES command. This format shall consist of the DEVICE/PORT DESIGNATION header only with DESIGNATION LENGTH set to zero and an empty DESIGNATION field. The PROTOCOL TYPE and TYPE CODE shall be ignored. The application client may use this format to remove an alias entry from its initiator's alias list.

If the device server has insufficient resources to make all requested changes to the alias list, the device server shall make no changes to the alias list and shall return CHECK CONDITION status, sense key of ILLEGAL REQUEST and additional sense code of INSUFFICIENT RESOURCES,

The device server shall not attempt to verify or validate any SCSI device or port designation mapped to an alias value at the time of processing the CHANGE ALIASES command.

3.5.3 Additional changes to Reservation clause of SPC-3.

In addition to the above clauses, one new line should be added to Table 10 (of SPC-2 rev 19) concerning the commands allowed in the presence of reservations. One line should be added for CHANGE ALIASES/REPORT ALIASES and this line should match that for REPORT DEVICE IDENTIFIER (that is, “Allowed” in all cases).

[EDITOR’S NOTE: with all the “special” commands which are just service actions for MAINTENANCE IN/OUT defined in SPC-3, perhaps we can look for a cleaner approach to all these, with respect to proliferating clauses for each service action and with respect to reservation clause. I don’t have a good suggestion, however.]

3.6 Addition to SPC-3 of a new EXTENDED COPY target descriptor

Add the following clause to the EXTENDED COPY clause on target descriptors (7.5.6).

[EDITOR’S NOTE: it has been suggested that ALL the target descriptors be moved to the model clause of EXTENDED COPY for cleaner reference by the XOR/SBC-x commands. I agree with this sentiment, but have no specific proposal to that effect.]

7.5.6.zz Alias target descriptor format

Targets may be identified indirectly using the alias target descriptor format shown in Table zz1.

Table zz1. Alias target descriptor format.

Byte	Bit							
	7	6	5	4	3	2	1	0
0	DESCRIPTOR TYPE CODE (E6h)							
1	RESERVED		NUL	PERIPHERAL DEVICE TYPE				
2	RESERVED							
3	RESERVED						LU ID TYPE	
4	MSB							
11	LU IDENTIFIER							LSB
12	MSB							
19	TARGET DEVICE/PORT ALIAS							LSB
20	RESERVED							
27	RESERVED							
28	RESERVED							
31	Device type specific parameters							

The DESCRIPTOR TYPE CODE, PERIPHERAL DEVICE TYPE and NUL fields and the device type specific parameters are described in 7.5.6.1.

The LU ID TYPE field and LU IDENTIFIER field are described in 7.5.6.1.

The TARGET DEVICE/PORT ALIAS field indicates an alias value in the requesting initiator's alias list as managed by the CHANGE ALIASES command (see 7.yy) and maintained by the device server. If the value of the TARGET DEVICE/PORT ALIAS field is not in the requesting initiator's alias list, the device server shall return CHECK CONDITION status, with sense key set to COPY ABORTED and additional sense code set to COPY TARGET DEVICE NOT REACHABLE.

The copy manager shall resolve the TARGET DEVICE/PORT ALIAS values in all alias target descriptors of a single EXTENDED COPY command as an atomic operation with respect to the alias list. This rule shall apply to any device server function or manager that consults the alias list in the context of any third-party command, unless otherwise specified in specific command subclause.