Date: June 20, 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 Elliott), there are problems with the specification of logical units and targets in some of the XOR commands in SBC-2. This proposal is aimed at resolving these problems. (The problem of the 64bit limitation on identifiers mentioned in previous versions of this proposal has been removed from SAM-2, so this is no longer an issue.) 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.

Open issues:

a) Is OPERATION IN PROGRESS the best choice for ASC/ASCQ when a (substantive) CHANGE ALIASES is processed while another task is using the alias list? Other choices are INVALID FIELD IN PARAMETER DATA (vague?) or to define a new ASC/ASCQ such as ALIAS LIST IN USE.

b) Is the language of the proposed SPC-3 clause 7.xx.3 (see 3.6.1) for “content” and “validity” requirements for designation formats acceptable (see also 2.1 for a discussion of the issues)?

c) Is there any need for any other protocol independent designation formats besides the Null Designation? See the proposed SPC-3 clause 7.xx.4 (in 3.6.1 of this proposal).

Revision 4 incorporates the following changes (all suggested by Ralph Weber):

a) minor editorial changes;

b) addition of three proposed glossary entries;

c) selection of OPERATION IN PROGRESS for the additional sense code response to a CHANGE ALIASES command while another task is using the alias list.

Revision 3 incorporates the following changes:

a) major editorial changes (as suggested by Ralph Weber)

b) removal of “atomicity” requirement and addition of error condition if a CHANGE ALIASES is processed at a time when the device server has enabled tasks that reference the alias list

c) change to reservation conflict rules; previous drafts had “Allowed” for both CHANGE and REPORT ALIASES in all cases; this draft has “Allowed” in all cases for REPORT ALIASES and “Conflict” for CHANGE ALIASES except in the specific

case of the “Registered initiator (RO all types)” where it is “Allowed”. Previous revisions allowed CHANGE ALIASES in all cases. This was changed because “initiator-specific” alias lists avoid collisions. However, since this command can consume resources, it seemed more reasonable to change this to “Conflict”.

d) change of a field name from TYPE CODE to the name FORMAT CODE in the alias entries because “format” is a better term

e) change the definition of the NUMBER OF ALIASES field in CHANGE ALIASES parameter data to reserved (it’s redundant with the rest of the parameter data in this case).

f) addition of a place for protocol independent formats and specification of the Null Designation under this scope

g) more detailed and explicit proposals for designation formats for specific protocols including FCP and iSCSI. SRP is being handled by the SRP working group.

Revision 2 incorporates the following changes:

h) 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”)

i) 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

j) 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

k) updated references to spc2r19 and sam2r16

l) 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

m) 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

n) 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 only16 bytes in a target descriptor for a target identifier; 16bytes is insufficient for some long identifiers such as IPv6 address with TCPport qualifier.

d) SAM-2 specifies that port identifiers shall be at most 64bits and this is also insufficient to deal with SRP and iSCSI protocols. [This has already been addressed in the multiport model changes approved for SAM-2.]

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 eight 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 CHANGE ALIASES and REPORT ALIASES that allow an initiator to define an association of an eight 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 eight 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) Definition of alias designations for FCP and iSCSI, as well as one Null Designation that is protocol independent.

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 REPORT ALIASES is never blocked by a reservation. The CHANGE ALIASES is only allowed in the presence of a reservation by the appropriately authorized registsrants. This is different from Revision 2.

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 alias list can be variable length, up to a maximum of 65532 bytes. The specific format is defined by a protocol code and format code. The protocol code specifies the transport protocol and the format 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.] This proposal also defines a format code space for protocol independent designations (such as the Null Designation).

f) The REPORT ALIASES command will always report the entire alias list. There is no defined mechanism for requesting any single entry in the alias 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 port that is accessible to the device server (e.g., when it needs to find the indicated device or port in an EXTENDED COPY operation). In particular, the proposal

(see 3.6.1 of this proposal and proposed SPC-3 clause 7.xx.3) specifies that the following are legitimate 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 (but we leave this as beyond the scope of the standard).

Along with specifying what belongs in a designation, this proposal (see 3.6.1 of this proposal and the proposed clause SPC-3 7.xx.3) defines the term “valid” for designations and when the check for validity should occur. There are cases where such a designation may not be “valid” from the point of view of the device server managing the alias list (say, 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. In this case the designation is considered “invalid” and the target unreachable. 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 addresses, 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.

The question was raised whether these requirements (content and validity) belong in the standard (in particular in SPC-3). For the “content” issue, the author is attempting to specify what really belongs in such a designation (namely, that it really should have name information in it, and that optional addressing information is also allowed.) This is intended to both give the reader a clear understanding of what to look for in a designation and is instructional to future authors of new designation formats. Is there an alternative mechanism other than the wording proposed here to accomplish at least the first of these goals? Would a NOTE be sufficient? In the first paragraph of the proposed SPC-3 clause 7.xx.1 we have the following definition:

> A designation contains a name and optional addressing information that identifies a SCSI device or port.

Is this definition and wording sufficient?

For the “validity” issue, the author is attempting to provide a protocol independent vocabulary or framework for checking whether the contents of a designation has any meaning (validity) to the device server that has to use it. That vocabulary is used, for example, in the alias target descriptor (see 3.7 of this proposal) and could be used in a protocol standard that defines a format. Examples of similar notions are: (a) MODE SELECT says that, in the protocol specific LUN page and protocol specific port page, use of a PROTOCOL IDENTIFIER that doesn’t match the device server’s service delivery subsystem causes CHECK CONDITION and (b) the language in the “FC N_Port with World Wide Name checking target descriptor format” (SPC-2, rev 19, clause 7.2.6.4) that defines a rule (check) on the contents of the target descriptor. The language in this proposal defines the vocabulary for this, but leaves the rules to the specific protocols or format specifications. The requirements and notes in the proposed SPC-3 clause 7.xx.2 are intended to mirror the two examples cited here. Is there an alternative wording or mechanism to accomplish this goal? Also, if the standard can’t specify “content” as name+optional address, then it becomes harder for any language to speak about validity.

The author welcomes alternative suggestions on how to deal with these two issues.

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-3 for a description of the fields in this command.

with

> See the REBUILD (16) command (5.1.13) and SPC-3 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

	Bit							
Byte	**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 43	SOURCE DESCRIPTOR (first)							
40*x*-36 40*x*+3	SOURCE DESCRIPTOR (last)							
40*x*+4 40*x*+*y*+3	PAD, if any (length *y*)							
40*x*+*y*+4	MSB							
40*x*+*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 eight Byte LBA version).

	Bit							
Byte	**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-3.

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-3 for a description of the fields in this command.

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.

[AUTHOR'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 t~~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 6	Reserved							
7	ADDITIONAL CDB LENGTH (38h)							
8	MSB							
9	SERVICE ACTION (0007h)							LSB
10	Reserved			DPO	FUA	DISABLE WRITE	PORT CONTROL	
11	Reserved							
12	MSB							
43	SECONDARY ADDRESS DESCRIPTOR (32 bytes)							LSB
44	MSB							
51	LOGICAL BLOCK ADDRESS (8 bytes)							LSB
52	MSB							
59	SECONDARY LOGICAL BLOCK ADDRESS (8 bytes)							LSB
60	MSB							
63	TRANSFER LENGTH (4 bytes)							LSB

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-3.

See the XDWRITE EXTENDED (16) command (5.1.36) and SPC-3 for a description of all other fields in this command.

3.5 Addition to SPC-3 for new glossary entries

The following should be added to SPC-3's glossary.

3.1.aa alias list: a list of alias values and their associated designations maintained by the device server and managed by the CHANGE ALIASES (see 7.xx) and REPORT ALIASES command (see 7.yy).

3.1.bb alias value: an eight byte value associated to a designation in the alias list and used in command or parameter data to reference a SCSI device or port (see 7.xx.1).

3.1.cc designation: a name and optional addressing information that identifies a SCSI device or port (see 7.xx.1).

3.6 Addition to SPC-3 of CHANGE ALIASES and REPORT 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.]

3.6.1 A clause for CHANGE ALIASES command for SPC-3

The following clause should be added to SPC-3.

7.xx CHANGE ALIASES command

7.xx.1 CHANGE ALIASES command introduction

The CHANGE ALIASES command (see Table xx1) requests that the device server maintain and make changes to a list of associations between eight byte alias values and SCSI device or port designations. A designation contains a name and optional addressing information that identifies a SCSI device or port (see 7.xx.3). The alias list may be queried by the initiator via the REPORT ALIASES command (7.yy). If the REPORT ALIASES command is supported then the CHANGE ALIASES command shall also be supported.

[AUTHOR'S NOTE: I've added a definition of "designation" to the above paragraph. It is consistent with the "content" issue raised in 2.1 of this proposal.]

The CHANGE ALIASES command is a service action of the MAINTENANCE OUT command. Additional MAINTENANCE OUT service actions are defined in SCC-2 and in this standard. The MAINTENANCE OUT service actions defined in SCC-2 apply only to SCSI devices that return a device type of 0Ch or the SCCS bit equal to one in their standard INQUIRY data.

On successful completion of a CHANGE ALIASES command, the device server shall maintain an association of each assigned eight byte alias value to the SCSI device or port designation. These associations shall be cleared under any event that resets the logical unit and events designated by the transport protocol. The device server shall maintain a separate alias list for each initiator.

A CHANGE ALIASES command may add, change or remove entries from the alias list. Alias list entries not referenced in the CHANGE ALIASES parameter data shall not be changed.

NOTE1: An application client may use alias values to reference SCSI devices or ports in third party commands (e.g., EXTENDED COPY). The alias list provides a mechanism for eight byte third party identifier fields to reference 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 containing in the parameter data an alias tar-

get descriptor (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 (see 7.xx.4).

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 terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and additional sense code shall be set to INSUFFICIENT RESOURCES.

Table xx1. CHANGE ALIASES command.

	Bit							
Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	Reserved			SERVICE ACTION (0Bh)				
2								
5	Reserved							
6	MSB							
9	PARAMETER LIST LENGTH							LSB
10	Reserved							
11	CONTROL							

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 value of zero indicates that no data shall be transferred and no changes shall be made in the alias list.

If the parameter list length results in the truncation of the header or any alias entry, then the device server shall make no changes to the alias list and terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to PARAMETER LIST LENGTH ERROR.

If the device server processes a CHANGE ALIASES command that contains at least one alias entry (see Table xx2) while there exists any other enabled task that references an alias entry in the alias list, the device server shall terminate the CHANGE ALIASES command with CHECK CONDITION status and sense key set to ILLEGAL REQUEST. The additional sense code shall be set to OPERATION IN PROGRESS.

[AUTHOR'S NOTE: The above ASC/ASCQ is only slightly more informative than the generic INVALID FIELD IN PARAMETER DATA. The proposed value of OPERATION IN PROGRESS (00h 16h) is used in extended copy for other purposes. Another alternative is to define an new ASC/ASCQ such as ALIAS LIST IN USE (perhaps 00h 17h).]

The format of the parameter data for the CHANGE ALIASES command is specified in 7.xx.2.

7.xx.2 CHANGE ALIASES and REPORT ALIASES parameter data

The format of the parameter data for the CHANGE ALIASES and REPORT ALIASES 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. CHANGE ALIASES and REPORT ALIASES parameter list.

	Bit							
Byte	7	6	5	4	3	2	1	0
0	MSB							
3	ADDITIONAL LENGTH (n-3)							LSB
4								
5	Reserved							
6	MSB							
7	NUMBER OF ALIASES							LSB
8								
n	Alias Entry/Entries							

The ADDITIONAL LENGTH field specifies the number of bytes in the remaining parameter data. For the REPORT ALIASES command, the value of this field shall contain the actual number of bytes available in the parameter data and shall not be changed if the CDB contains insufficient allocation length.

For the CHANGE ALIASES command, the NUMBER OF ALIASES field is reserved. For the REPORT ALIASES command, the NUMBER OF ALIASES field indicates the number of alias entries in the alias list and shall not be changed if the CDB contains insufficient allocation length.

For the REPORT ALIASES command, if there are no alias entries in the alias list, then the ADDITIONAL LENGTH field shall be set to four and the NUMBER OF ALIASES field shall be set to zero.

7.xx.3 Alias entry formats

The format for an alias entry is specified in Table xx3.

Table xx3. CHANGE ALIASES and REPORT ALIASES alias entry format.

Byte	Bit 7	6	5	4	3	2	1	0
0	MSB							
7	ALIAS VALUE							LSB
8	PROTOCOL IDENTIFIER							
9								
10	Reserved							
11	FORMAT CODE							
12								
13	Reserved							
14	MSB							
15	DESIGNATION LENGTH (n-15)							LSB
16	MSB							
n	DESIGNATION							LSB

Each alias entry contains an eight byte ALIAS VALUE that the device server associates, in the alias list, with the SCSI device or port specified by the values of the PROTOCOL IDENTIFIER, FORMAT CODE and DESIGNATION fields.

The PROTOCOL IDENTIFIER field indicates the transport protocol to which the FORMAT CODE and DESIGNATION format applies. Values of the PROTOCOL IDENTIFIER field between 0h and Fh inclusive are specified in Table 165 (of SPC-2 rev 19). The value 80h designates the protocol independent formats as specified in 7.xx.4. All other values of the PROTOCOL IDENTIFIER field are reserved.

The FORMAT CODE defines the format of the DESIGNATION field with respect to the specific transport protocol indicated by the PROTOCOL IDENTIFIER value. The format is defined in the appropriate protocol standard or in this standard.

The DESIGNATION LENGTH specifies the number of bytes of the DESIGNATION field. This value shall be a multiple of four.

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

[AUTHOR’S NOTE: for the following, see the comments in 2.1 of this proposal.]

The DESIGNATION field should designate a unique SCSI device or port using the following identifying data:

a) a SCSI device name or a SCSI port name (required),

b) one or more optional SCSI port addresses or transport-specific addresses.

A DESIGNATION value that is not a multiple of four bytes long shall be padded in the least significant bytes by zeros.

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) or any other command that requires reference to the alias list.

[AUTHOR'S NOTE: does the above paragraph fit better here or somewhere later?]

If a designation identifies a unique SCSI device or port that is within a SCSI domain accessible to the device server, the designation is considered valid.

Based on the protocol specific rules for a given format, a designation that does not identify a unique SCSI device or port within the SCSI domain(s) accessible to 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 device server is not able to address the named device or port through 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 or this 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 a mechanism for resolving names to addresses. Access to such a service is protocol and implementation specific.

7.xx.4 Protocol independent alias entry formats

Table xx4 specifies the Format Code values defined for a protocol independent alias entry.

Table xx4: Protocol independent FORMAT CODE values and DESIGNATION formats.

Format Code	Designation Description	Designation length	Designation content
00h	Null Designation	0 bytes	empty
01h-FFh	reserved	n/a	n/a

The Null Designation should be used in the parameter data of the CHANGE ALIASES command to remove an alias entry from the alias list. The Null Designation shall not appear in REPORT ALIASES parameter data.

[AUTHOR'S NOTE: The only currently defined protocol independent format is the Null Designation. This is used to delete alias list entries. Are there any other protocol independent formats we need to define? If not, should we add a NOTE here or other words that make this clear?]

[AUTHOR'S NOTE: Rob Elliott's is developing a proposal for SRP that contains specifications for designation formats for that protocol. Clause 3.6.4 of this proposal specifies two formats for FCP that presumably would go in an annex of SPC-3 until FCP-3 gets started. These cover the two FC specific target descriptors in EXTENDED COPY that include WWNs. Clause 3.6.5 of this proposal specifies formats for iSCSI. Since the iSCSI protocol document is under the auspices of IETF, this clause is only informative with respect to this proposal. The author will work with IETF to complete the specification of these formats for iSCSI.]

3.6.2 A clause for REPORT ALIASES command for SPC-3

The following clause should be added to SPC-3.

7.yy REPORT ALIASES command

The REPORT ALIASES command (see Table yy1) requests that the device server return the alias list. The alias list itself is managed by the initiator via the CHANGE ALIASES command (7.yy). If the CHANGE ALIASES command is supported then the REPORT ALIASES command shall also be supported.

The REPORT ALIASES command is a service action of the MAINTENANCE IN command. Additional MAINTENANCE IN service actions are defined in SCC-2 and in this standard. The MAINTENANCE IN service actions defined in SCC-2

apply only to SCSI devices that return a device type of 0Ch or the SCCS bit equal to one in their standard INQUIRY data.

Table yy1. REPORT ALIASES command.

	Bit							
Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	Reserved				SERVICE ACTION (0Bh)			
2 5	Reserved							
6	MSB							
9	ALLOCATION LENGTH							LSB
10	Reserved							
11	CONTROL							

The ALLOCATION LENGTH field indicates the number of bytes that have 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 may be determined 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 parameter data is specified in 7.xx.2. The header indicates the amount of additional data and the number of entries in the alias list. This is followed by a list of alias entries.

3.6.3 Additional changes to Reservation clause of SPC-3.

In addition to the above clauses, two new lines 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 and this line should match that for SET DEVICE IDENTIFIER (that is, “Conflict” in all cases except “From registered initiator (RO all types)” where it should be “Allowed”). An additional line should be added for 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 could 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.]

3.6.4 Normative annex to SPC-3 for FCP alias entry designation formats

Add the following as a normative annex to SPC-3 to define alias entry designation formats for FCP in order to bridge the gap until FCP-3 gets going.

Annex X. FCP protocol dependent alias entry formats

X.1 FCP protocol specific format codes

This annex defines the FCP protocol specific alias entry formats and codes used in the CHANGE ALIASES and REPORT ALIASES commands (see SPC-3) to designate SCSI devices or ports on an FCP service delivery subsystem. The contents of this annex will be placed in FCP-3.

[AUTHOR'S NOTE: Is the above language the correct way to indicate where this really belongs?]

For an FCP protocol specific alias entry, the PROTOCOL IDENTIFIER shall be set to 00h (as defined in Table 165 of SPC-2 rev 19) and the FORMAT CODE values are defined in Table X1.

Table X1: FCP protocol specific FORMAT CODE values and DESIGNATION formats.

Format Code	Designation Description	Designation length	Designation content
00h	WWPortname	8 bytes	FC port World Wide Name (see X.2.)
01h	WWPortname with N_Port checking	12 bytes	FC port World Wide Name with N_Port checking (see X.3)
02h-FFh	reserved	n/a	n/a

X.2 FCP WWPortname designation format

Table X2 describes the FCP WWPortname designation format.

Table X2. FCP WWPortname designation format.

	Bit							
Byte	7	6	5	4	3	2	1	0
0	MSB							
7	FC PORT WORLD WIDE NAME							LSB

The FC PORT WORLD WIDE NAME field shall contain the port World Wide Name defined by the Physical Log In (PLOGI) extended link service, defined in FC-FS.

An FCP WWPortname designation is valid if the device server has access to a SCSI domain containing a Fibre Channel network and there exists a Fibre Channel port on that network with the specified port World Wide Name.

X.3 FCP WWPortname with N_Port checking designation format

Table X3 describes the FCP WWPortname with N_Port checking designation format.

Table X3. FCP WWPortname with N_Port checking designation format.

Byte	Bit 7	6	5	4	3	2	1	0
0	MSB							
7	FC PORT WORLD WIDE NAME							LSB
8	Reserved							
9	MSB							
11	N_PORT							LSB

The FC PORT WORLD WIDE NAME field shall contain the port World Wide Name defined by the Physical Log In (PLOGI) extended link service, defined in FC-FS.

The N_PORT field shall contain the FC_FS port D_ID to be used to transport frames including PLOGI and FCP-2 related frames.

This designation is valid if all of the following conditions are true:

a) device server has access to a SCSI domain formed by a Fibre Channel fabric,

b) the fabric contains a port with the specified port World Wide Name, and

c) the value in the N_PORT field is the N_PortID of a Fibre Channel port whose port World Wide Name matches that in the FC PORT WORLD WIDE NAME field.

3.6.5 Informative clause for iSCSI standard's alias entry designation formats

This clause specifies alias entry designation formats for iSCSI. As that protocol standard is under the auspices of IETF and its IPS working group, this is only informative with respect to T10. The author will work with that working group to finalize this specification.

Clause Y. iSCSI protocol specific alias entry formats

Y.1 iSCSI protocol specific format codes

This clause defines the iSCSI protocol specific alias entry formats and codes used in the CHANGE ALIASES and REPORT ALIASES commands (see SPC-3) to designate SCSI devices or ports on an iSCSI service delivery subsystem.

For an iSCSI protocol specific alias entry, the PROTOCOL IDENTIFIER shall be set to 05h (as defined in Table 165 of SPC-2 rev 19) and the FORMAT CODE values are defined in Table Y1.

Table Y1: iSCSI protocol specific FORMAT CODE values and DESIGNATION formats.

Format Code	Designation Description	Designation length (maximum)	Designation content
00h	iSCSI Name	256 bytes	Name in UTF-8 format (null terminated), with pad (see Y.2)
01h	iSCSI Name with binary IPv4 address	268 bytes	Name in UTF-8 format (null terminated), binary IPv4 address, binary TCP port, binary Internet Protocol Number, with pad (see Y.3)
02h	iSCSI Name with IPName	520 bytes	Name in UTF-8 format (null terminated), IPName (null terminated), binary TCP port, binary Internet Protocol Number, with pad (see Y.4)
03h	iSCSI Name with binary IPv6 address	280 bytes	Name in UTF-8 format (null terminated), binary IPv6 address, binary TCP port, binary Internet Protocol Number, with pad (se Y.5)
04h-FFh	reserved	n/a	n/a

NOTE: a designation that contains no IP addressing information or contains IP addressing information that does not address the named SCSI device may require a device server to have access to a name server or to other discovery protocols to resolve the given iSCSI Name to an IP address through which the device server may establish iSCSI Login. Access to such a service is protocol and implementation specific.

Y.2 iSCSI Name designation format

Table Y2 describes the iSCSI Name designation format.

Table Y2. iSCSI Name designation format.

	Bit							
Byte	7	6	5	4	3	2	1	0
0 n	ISCSI NAME							
n+1	NULL (00h)							
n+2 4m-1	PAD (if needed)							

The ISCSI NAME field shall contain the iSCSI Name of an iSCSI Node. Refer to RFC XXXX for a description of iSCSI Names. The ISCSI NAME field shall not include a byte set to 00h.

A NULL (00h) byte shall terminate the iSCSI Name.

Zero to three bytes set to zero shall be appended in the PAD field so that the total length of the designation is a multiple of four. The PAD field shall be ignored.

An iSCSI Name designation is valid if the device server has access to a SCSI domain containing an IP network and there exists an iSCSI Node on that network with the specified iSCSI Name.

Y.3 iSCSI Name with binary IPv4 address designation format

Table Y3 describes the iSCSI Name with binary IPv4 address designation format.

Table Y3. iSCSI Name with binary IPv4 address designation format.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0 n	ISCSI NAME							
n+1	NULL (00h)							
n+2 4m-1	PAD (if needed)							
4m	MSB							
4m+3	IPV4 ADDRESS							LSB
4m+4 4m+5	Reserved							
4m+6	MSB							
4m+7	PORT NUMBER							LSB
4m+8 4m+9	Reserved							
4m+10	MSB							
4m+11	INTERNET PROTOCOL NUMBER							LSB

The ISCSI NAME field shall contain the iSCSI Name of an iSCSI Node. Refer to RFC XXXX for a description of iSCSI Names. The ISCSI NAME field shall not include a byte set to 00h.

A NULL (00h) byte shall terminate the iSCSI Name.

Zero to three bytes set to zero shall be inserted in the PAD field so that the total length of the designation is a multiple of four. The PAD field shall be ignored.

The IPV4 ADDRESS field shall contain an IPv4 address. Refer to RFC 791 for a description of IPv4 addresses.

The PORT NUMBER field shall contain a port number. Refer to RFC 790 for a description of port numbers.

The INTERNET PROTOCOL NUMBER field shall contain an Internet protocol number. Refer to RFC 790 for a description of Internet protocol numbers.

An iSCSI Name with binary IPv4 address designation is valid if the device server has access to a SCSI domain containing an IP network and there exists an iSCSI Node on that network with the specified iSCSI Name. The IPv4 address, port number and internet protocol number provided in the designation may be used by a device server for addressing to discover and establish communication with the named iSCSI Node. Alternatively, the device server may use other protocol specific or vendor specific methods to discover and establish communication with the named iSCSI Node.

Y.4 iSCSI Name with IPname designation format

Table Y4 describes the iSCSI Name with IPname designation format.

Table Y4. iSCSI Name with IPname designation format.

	Bit							
Byte	7	6	5	4	3	2	1	0
0 n	ISCSI NAME							
n+1	NULL1 (00h)							
n+2 n+l+2	IPNAME							
n+l+3	NULL2 (00h)							
n+l+4 4m-1	PAD (if needed)							
4m 4m+1	Reserved							
4m+2	MSB							
4m+3	PORT NUMBER							LSB
4m+4 4m+5	Reserved							
4m+6	MSB							
4m+7	INTERNET PROTOCOL NUMBER							LSB

The ISCSI NAME field shall contain the iSCSI Name of an iSCSI Node. Refer to RFC XXXX for a description of iSCSI Names. The ISCSI NAME field shall not include a byte set to 00h.

A NULL1 (00h) byte shall terminate the iSCSI Name.

The IPNAME field shall contain a Internet protocol domain name. Refer to RFC 1035 for a description of domain names.

A NULL2 (00h) byte shall terminate the Internet protocol domain name.

Zero to three bytes set to zero shall be inserted in the PAD field so that the total length of the designation is a multiple of four. The PAD field shall be ignored.

An iSCSI Name with IPname designation is valid if the device server has access to a SCSI domain containing an IP network and there exists an iSCSI Node on that network with the specified iSCSI Name. The domain name, port number and internet protocol number provided in the designation may be used by a device server for addressing to discover and establish communication with the named iSCSI Node. Alternatively, the device server may use other protocol specific or vendor specific methods to discover and establish communication with the named iSCSI Node.

Y.5 iSCSI Name with binary IPv6 address designation format

Table Y5 describes the iSCSI Name with binary IPv6 address designation format.

Table Y5. iSCSI Name with binary IPv6 address designation format.

	Bit							
Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0 n	ISCSI NAME							
n+1	NULL (00h)							
n+2 4m-1	PAD (if needed)							
4m	MSB							
4m+15	IPV6 ADDRESS							LSB
4m+16 4m+17	Reserved							
4m+18	MSB							
4m+19	PORT NUMBER							LSB
4m+20 4m+21	Reserved							
4m+22	MSB							
4m+23	INTERNET PROTOCOL NUMBER							LSB

The ISCSI NAME field shall contain the iSCSI Name of an iSCSI Node. Refer to RFC XXXX for a description of iSCSI Names. The ISCSI NAME field shall not include a byte set to 00h.

A NULL (00h) byte shall terminate the iSCSI Name.

Zero to three bytes set to zero shall be inserted in the PAD field so that the total length of the designation is a multiple of four. The PAD field shall be ignored.

The IPV6 ADDRESS field shall contain an IPv6 address. Refer to RFC 2373 for a description of the IPv6 address.

The PORT NUMBER field shall contain a port number. Refer to RFC 790 for a description of port numbers.

The INTERNET PROTOCOL NUMBER field shall contain an Internet protocol number. Refer to RFC 790 for a description of Internet protocol numbers.

An iSCSI Name with binary IPv6 address designation is valid if the device server has access to a SCSI domain containing an IP network and there exists an iSCSI Node on that network with the specified iSCSI Name. The IPv6 address, port number and internet protocol number provided in the designation may be used by a device server for addressing to discover and establish communication with the named iSCSI Node. Alternatively, the device server may use other protocol specific or vendor specific methods to discover and establish communication with the named iSCSI Node.

3.7 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. The author agrees with this sentiment, but has no specific proposal to that effect.]

7.5.6.zz Alias target descriptor format

Target devices or ports 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	ALIAS VALUE							LSB
20								
27	Reserved							
28								
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 ALIAS VALUE field indicates an alias value in the alias list as managed by the CHANGE ALIASES command (see 7.xx) and maintained by the device server.

When the device server first processes a target descriptor of this format, it shall check the value of the ALIAS VALUE field for a corresponding entry in the alias list. If the value is not in the alias list or the device server is unable to validate the designation associated with the alias value, the command shall be terminated because the target is unavailable (see 7.2.3). An application client generating third party commands that include alias target descriptors in the CDB (e.g., XDWRITE EXTENDED (64)) or in parameter data (e.g., EXTENDED COPY) is responsible for providing a valid entry in the alias list using the CHANGE ALIASES command (see 7.xx) prior to sending such third party commands.

[AUTHOR'S NOTE: do we need a reference to the definition of "valid", that is, to clause 7.xx.2?]