SAM Revision 12 Review Comments for Working Group Discussion
January 19, 1994 X3T10/94-029R0

From: Charles Monia
SAM Technical Editor

To: Members of X3T10

Subject: SAM Revision 12 Review Comments for Working Group Consideration

This document contains all proposed responses referred for working group consideration during the January 1994 meeting of X3T10.
The following issues for working group discussion are extracted from the review comments on SAM, revision 12 submitted by Jeff Williams of Hewlett-Packard.

Begin working group issues:
#010 (E) Page 20, Section 3.5, Para 2

Shouldn't the "[input-1,input-2,...]" be "[input]" indicating zero or more input parameters. This is consistent with the notation in the previous section.

> The suggested notation ("{...}"") is used to defined a composite object. I acknowledge the potential for confusion and will consider modifying the notation accordingly. I am reluctant, however, to extend object annotation to procedure calls.

#014 (T) Page 31, Section 4.6, Para 1

You cannot require that the order be preserved in all cases for a given pair of devices. For example, if I run in fibre channel over a fabric and send two command frames, the order is only guaranteed if I request in-order transmission over the fabric. I may not want to do this for performance reasons. I think that you need to say that order may be "imposable" in the cases where ordered tasks are sent or some other ordering is required, but you cannot require it in all cases.

> I believe that, although some physical transports, such as Fibre Channel, may reorder data in transit, the ordering specified by the sender is restored before the transaction is presented to the consumer.

> In any event, ordering is implicit in SCSI-2 today. Therefore placing a new ordering burden on the application client (e.g., the device driver) may lead to implementations that break existing code that would otherwise be portable.

> I suggest this issue be left open for discussion at the next working group.
TASK SET FULL states that it is required if tagged tasks are supported. Tagged tasks support is required, therefore remove the statement about it being optional (the first sentence).

> As I understand it, device support for tagged tasks is still optional.
> If it is not optional, then devices such as tape drives will have to be modified for SCSI-3 compliance. i.e. As a minimum, even if such devices are limited to one task per logical unit, they will have to understand queue tag messages and the like.
> I recommend discussing the matter at the next working group.
>
The following issues for working group discussion were extracted from the comments submitted by John Lohmeyer of NCR

Begin comments

6. In 3.5, Notation for Procedures and Functions, the usage of () and [] differ from their usage in 3.4.1, Object Notation. Readers could become confused. I suggest aligning these two notations by always using [] for optional items and always using () for item lists.

> While I agree there's potential for confusion, I don't see an easy way around this problem since both notations are based on well-established conventions.

From: US3RMC::"John.Lohmeyer@FtCollinsCO.NCR.COM" "John Lohmeyer"
4-JAN-1994 03:55:29.14
To: Charles Monia <starch::monia>
CC:
Subj: Re: Proposed responses to your review comments on SAM Rev 12.

6. In 3.5, Notation for Procedures and Functions, the usage of () and [] differ from their usage in 3.4.1, Object Notation. Readers could become confused. I suggest aligning these two notations by always using [] for optional items and always using () for item lists.

> While I agree there's potential for confusion, I don't see an easy way around this problem since both notations are based on well-established conventions.

I would agree that both notations have been used, but I don't think both are commonly used in the SAME document. That is the point of my comment. Why does SAM have to use BOTH conventions? Are we being compatible with the notation that exists in two external documents? If so, we should reference these documents and state why our notations are consistent with the external documents rather than being self-consistent.

>> In my opinion, the way around the problem is to ensure that the context in which each
>> is used is clear to the reader so that it is possible to distinguish which notation is in effect.
>> In that light, I'd appreciate specific instances where such confusion has been encountered.
SAM Revision 12 - Panasonic Comments for Working Group Discussion
January 19, 1994 X3T10/94-029R0

The following are comments received from Steve Heil of Panasonic for consideration by the X3T10 Working Group during the January 1994 meeting.

Begin summary of Panasonic Working Group issues
====================================================================
FR1. Though the SAM document has been through several major revisions there is still significant work needed for the document valuable to the industry.
This is particularly the case with Annexes A and C which represent a significant amount of committee effort but are not consistent with the remainder of the document. Concepts like queuing and terminology such as "execute", "task", "response", "confirmation" are not consistent. The document will confuse readers in its present state.

> SAM must, of course, be in full technical agreement with the queuing model passed by the committee. Any discrepancies or inconsistencies between the body of the document and the queuing model will be corrected. In that regard, please cite specific instances where the draft is either unclear or at variance with the queuing model.
>
> It is my understanding that once such corrections are made annex C will be deleted. The committees' intent regarding the alternate task set descriptions in annex A, however, is not clear to me. I propose that the issue of whether or not to retain annex A be resolved in the working group."

R2. The document requires the use of "Per Logical Unit Task Set Boundaries" but discusses and provides for other implementations. This is very confusing. My experience is that these options in a standard will become requirements in the marketplace and therefore should be better documented in the standard. If the intent is to provide extensibility through these options it should be clearly stated.
>
> I believe the discussion of other alternatives in annex A was in accordance with the committees' wishes. Please see the previous response.
>
R3. I am confused by the requirement in clause 4.6 that all transactions be received in the order they were sent. My understanding was that some of the SCSI-3 transports do not maintain order (i.e. P 1394, Fibre Channel and possibly SSA).

> I believe that, although some physical transports, such as Fibre Channel, may reorder data in transit, the ordering specified by the sender is restored before the transaction is presented to the consumer.
> In any event, ordering is implicit in SCSI-2 today. Therefore
> placing a new ordering burden on the application client (e.g., the
> device driver) may lead to implementations that break existing
> code that would otherwise be portable.
> >
> > I suggest this issue be left open for discussion
> > at the next working group.
SAM Revision 12 - Seagate Comments for Working Group Discussion
January 19, 1994 X3T10/94-029R0

The following are comments received from Gene Milligan of Seagate for consideration by the X3T10 Working Group during the January 1994 meeting.

3) I think CAM should be added to Figure Anonymous in the scope. In listing (10) the description should be updated and the (?) deleted. In listing (12) it should be corrected to "Common" rather than "Command".

> It has been suggested to me that this matter should be discussed with the folks in the CAM Working Group. I recommend raising this issue during the upcoming working group.

52) Should the Group 6 and 7 vendor specific commands now be limited to 16 or less bytes?

> I believe that issue should be discussed in the working group. Assuming the committee concurs with removing the CDB format definitions from SAM, I assume this would then become an issue to be addressed by the command standards.

55) I remain concerned that there may be a conflict with the installed base which had a presumption of bit significance and the use of the BUSY bit in the Task Set Full Status code.

> Please feel free to raise the issue in the working group.

65) Why should a protocol standard not require overlapped commands (in the SCSI sense) to be detected? (6.5.3).

> The rationale is that, in some protocol implementations, checking for command overlap has too much overhead due to the large tag address space.

> I suggest discussing this matter during the working group.
The following are comments received from Bob Snively of Sun for consideration by the X3T10 Working Group during the January 1994 meeting.

Begin Sun Comments
==============================================
T 007 Page 31, Section 4.6

The last paragraph of section 4.6 indicates that the service delivery transactions are received in the order in which they are sent for a given pair of source and destination devices. Fibre Channel and some other channels allow the proper operation of SCSI with out of order delivery of command information. This restriction should be modified to allow the out of order delivery of commands if operating system or channel conventions can guarantee the proper behavior of the scsi targets. As an example, ordering of groups of commands can be enforced by the host adapter function or by management of individual commands with respect to the acknowledgment processing of commands requiring ordering.

> I had assumed that, while data may arrive out of order, the receiver would restore ordering before presenting the data to the consumer.
> I am concerned that relaxing the ordering requirement in the manner suggested will lead to implementations that break existing host code which depends on the implicit ordering provided by SIP/SPI.
> I would like to reopen this issue at the next working group.

E 008 Page 33, Section 4.7.1

Shouldn't the Initiator equation be:

\[ \text{Initiator} = 0(\text{Application Client}) + 1(\text{Initiator Identifier})1 \]

If not, a considerable amount of additional information is required to indicate what the rules are for the execution of tasks across multiple independent ports.

> Multiple identifiers should be considered as nothing more than an alias for the same physical entity.
> I would like input from the working group on this issue. According
> to past feedback, multiple identifiers for the same entity were
> considered acceptable.
>

E 009 Page 34, Section 4.7.2

Shouldn't the Target equation be:

\[ \text{Target} = 1\{\text{Logical Unit}\} + 1\{\text{Target Identifier}\} \]

If not, a considerable amount of additional information is required to indicate what the rules are for execution of tasks to a LUN having multiple target ports within a single task.

> See reply to comment 008.
>

T 014 Page 39, Section 6 and page 74, Section 9.1

The Autosense Data, as an output argument, is not defined elsewhere. In particular, it is missing from the definition of section 9.1. the Confirmation returned to the Application Client. The referenced clause should be defined.

> The autosense return flag indicates whether or not sense data was returned to the autosense buffer. Input from the working group is needed to determine the conditions under which autosense data is to be returned. i.e. Can autosense data be returned for any command or only those which complete with a status of CHECK CONDITION or COMMAND TERMINATED?
>
> If the latter, then this flag is unnecessary.
>

T 027 Page 52, Section 6.5.7

In the last paragraph of the page, the text indicates that AEN should be reported only once per occurrence of the causing event. In fact, for errors that are generic and may influence the operation of any attached initiators, the AEN should be presented to all attached initiators. The text should be modified to clearly indicate that AEN should be offered only
once to the initiator related to the command causing the failure, but is allowed to be offered to every initiator if it is unclear to the target which initiator will be affected by the failure.

> I believe this requires discussion by the working group at large.

T 040 Page 59, Section 7.3

The second sentence of the last paragraph may be overly general. The CLEAR ACA should only abort tasks if the QErr bit is set to one.

> This should be discussed at the next working group. The description in the specification reflects inputs from others in the working group.

TT 046 Page 70, sections 8.2 - 8.7

(Modification required to change vote to affirmative).

It is clear from the document that each protocol shall be required to provide a mechanism to perform each of the task management functions. In addition to this, it must be made absolutely clear which of the task management functions are optional for a SCSI device to implement and which ones are required. Some task management functions may only be required if certain other optional capabilities are allowed.

Terminate Task is an example of a function that is always optional.

Clear Task Set is an example of a function that is only required if the task set elects the definition:

Task Set = \{0(\text{Tagged Task}) + 0(\text{UnTagged Task})\}

Clear Auto Contingent Allegiance is an example of a function that is only required if the ACA bit is allowed to be set to one.

These are probably best placed in a new paragraph under each Task Management Function entitled "Service Response"
SAM Revision 12 - Sun Comments for Working Group Discussion
January 19, 1994 X3T10/94-029R0

A typical case would be for Terminate Task:

Service Response:

Function Complete: Indicates Terminate Task Function was accepted and will be attempted by Device Server
Function Rejected: Indicates Terminate Task Function not implemented by Device Server
Failure: Indicates Terminate Task Function could not be delivered to Device Server

A contrasting case would be for Clear Task Set:

Service Response:

Function Complete: Indicates Clear Task Set Function was accepted and will be attempted by Device Server.
Function Rejected: This response is only allowed for Device Servers that reject all Tagged Tasks.
Failure: Indicates Clear Task Set Function could not be delivered to Device Server.

Table of desired optionality:

<table>
<thead>
<tr>
<th>Task Management Function</th>
<th>Opt/Rqrd</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abort Task</td>
<td>Rqrd</td>
<td></td>
</tr>
<tr>
<td>Abort Task Set</td>
<td>Opt</td>
<td>Rqrd if Tagged Tasks supported</td>
</tr>
<tr>
<td>Clear ACA</td>
<td>Opt</td>
<td>Rqrd if ACA bit = 1 supported</td>
</tr>
<tr>
<td>Clear Task Set</td>
<td>Opt</td>
<td>Rqrd if Tagged Tasks supported</td>
</tr>
<tr>
<td>Target Reset</td>
<td>Rqrd</td>
<td></td>
</tr>
<tr>
<td>Terminate Task</td>
<td>Opt</td>
<td></td>
</tr>
</tbody>
</table>

> This issue needs to be discussed at the next working group.
> Others are of the impression that support for tagged queuing
> is mandatory for all devices.
SAM Revision 12 - Western Digital Comments for Working Group Discussion
January 19, 1994 X3T10/94-029R0

The following are comments received from Jeff Stai of Western Digital for consideration by the X3T10 Working Group during the January 1994 meeting.

Begin Western Digital Comments
===============================================
......Also (more importantly) all of the tools and services provided by SAM seem to allow for multiple Initiator and Target Identifiers only on the most simplified level: you can have more than one, but you can't relate one to the another within the same device. Given this, why don't we say:

Initiator = 1{Appl Client} + 1{Initiator Identifier}1

Likewise for target... The alternative is adding considerable complexity to make multiple IDs fully functional.

>
> Multiple I/D's are supposed to be aliases representing the same physical device.
>
> Since this was added by request, I'd like some feedback from the working group before I modify the definition. In my opinion, the issue of multiple target identifiers is irrelevant to SAM.
> Although the behavioral model requires one unique identifier,
> a system could implement more than one without violating any architectural requirements.
>
>
> pg 35, 4.7.3.1: "should" should be "shall", shouldn't it?

>
> The present wording was added by request to eliminate what was thought to be an unnecessary behavioral restriction.
>
> I suggest reopening the issue at the upcoming meeting of X3T10.

===============================================

pg 50, 6.5.3, 2nd pgf.: I don't understand the first sentence. Why not always detect overlaps? Give a reason in a note or delete the protocol specific provision and make it global.

>
> The reason for making it a protocol-specified requirement is due to
The large tag space for protocols like FCP [I should have
added "and hence the large perceived overhead to search for duplicate
identifiers". In any event, I believe this is another item for further
discussion].