ENDL
TEXAS

Date: 25 June 2002
To: T10 Technical Committee
From: Ralph O. Weber
Subject: Making Application Clients and Device Servers True Peers in SAM-2

Background

SAM-2 letter ballot comment Cisco 4 proposes that application clients should be true peers of device servers. This is in contrast to the current description and usage of application client that makes the application client more exactly the peer of a task, to whit:

> {4.3} … an application client is created to issue a single SCSI command or task management function.

However, this view of application clients is not universal in SAM-2, as exhibited by the glossary entry for application client:

> 3.1.4 application client: An object that is the source of SCSI commands.

The difficulty is that most uses of application client in SAM-2 and elsewhere in the SCSI family of standards do not depend on or outright ignore the "… issue a single SCSI command or task management function" aspect of the application client model. Nearly everyone who reads "application client" sees the words as meaning a functional peer of device server, wherein state is maintained from one command to the next and where persistent reservations information is remembered even across logical unit resets.

In a surprisingly small number of SAM-2 uses, the application client is simply not a peer of the device server. This proposal intends to remedy that.

Nature of the Proposal

This proposal follows two principles:

1) Where "application client" is used to describe a peer of the device server, no changes are made; and
2) Where "application client" is used to describe a peer of a task, "application client thread" is inserted in place of "application client".

It may be useful to note that the proposed changes affect only seven clauses:

4.3 The SCSI client-server model
4.7.1 SCSI initiator device
4.7.3 SCSI target/initiator device
5.5 Task and command lifetimes
5.7.1 Unlinked command example
5.7.2 Linked command example
6.10 Task management function example

It is understood that "application client thread" is seen as an undesirable term, however, its usage (as we shall see) is minimal. The term "application client thread" appears in only five of the modified clauses. The changes 4.7.1 and 4.7.3 only remove a note that there is one application client per command.

18484 Preston Road, Suite 102 PMB 178, Dallas, TX 75252 214-912-1373 972-596-2775 Fx ENDL_TX@computer.org

Specific Proposed Changes

All references in this proposed changes are to sam2r23.pdf.

In 4.3 (The SCSI client-server model) on PDF page 42, modify the first paragraph after figure 6 as follows:

> All requests originate from application clients residing within a SCSI initiator device. An application client ~~represents a thread of processing whose functionality~~ is independent of the interconnect and SCSI protocol. In an implementation, ~~that thread~~ the application client could correspond to the device driver and any other code within the operating system that is capable of managing I/O requests without requiring knowledge of the interconnect or SCSI protocol. In the architecture model, an application client ~~is created to issue~~ creates one or more application client threads each of which issues a single SCSI command or task management function. Application client threads are considered to be part of their parent application client. An application client thread ceases to exist once the command or task management function ends. Consequently, there is one application client thread for each pending command or task management request. ~~Within the initiator, one or more controlling entities, whose definition is outside the scope of the architecture model, oversee the creation of and interaction among application clients.~~

In 4.7.1 (SCSI initiator device) on PDF page 47 make the following changes.

Change "zero or more application clients" to "one ore more application clients" and modify figure 11 accordingly. Since an application client is a peer of a device server and since targets are required to have at least one device server, it makes sense that initiators should be required to have at least one application client.

Modify the forth paragraph after figure 11 (the last paragraph on the page) as follows:

> An application client is the source of commands and task management functions. ~~This model assumes that a SCSI initiator device contains one application client for each pending command or task management function.~~

In 4.7.3 (SCSI target/initiator device) on PDF page 48, change "zero or more application clients" to "one ore more application clients". On PDF page 49, modify figure 13 to reflect the change in the minimum number of application clients.

In 4.7.3 (SCSI target/initiator device) on PDF page 49, modify the fifth paragraph after figure 13 (the last in the clause) as follows:

> When the SCSI target/initiator device is operating as a SCSI initiator device an application client is the source of commands and task management functions. ~~This model assumes that there is one application client for each pending command or task management function.~~

In 5.5 (Task and command lifetimes) on PDF page 81, modify the paragraph after the first a,b list as follows:

> The application client assumes that the task exists and maintains an application client thread to interact with the task from the time the **Send SCSI Command** SCSI protocol service request is invoked until it receives one of the following target responses:

In 5.7.1 (Unlinked command example), make the following changes.

In figure 29, change "Application Client" to "Application Client Thread".

Change item 1 in the list describing figure 29 as follows:

> The application client thread performs an **Execute Command** remote procedure call by invoking the **Send SCSI Command** SCSI protocol service to send the CDB and other input parameters to the logical unit.

Change item 4 in the list describing figure 29 as follows:

A confirmation of **Command Complete Received** is passed to the ~~ULP~~ application client thread by the ~~initiator's service delivery subsystem~~ SCSI initiator port.

Note: Some of these changes do relate to the mulit-port SCSI device changes, not to this principal issue of this proposal.

In 5.7.2 (Linked command example), make the following changes.

In figure 30, change two instances of "Application Client" to "Application Client Thread".

Change the first sentence of item 1 in the list describing figure 30 as follows:

The application client thread performs an **Execute Command** remote procedure call by invoking the **Send SCSI Command** SCSI protocol service to send the CDB and other input parameters to the logical unit.

Change item 4 in the list describing figure 30 as follows:

The ~~LLP~~ SCSI initiator port returns the status and service response to the ~~ULP~~ application client thread by means of a **Command Complete Received** confirmation.

Change the first sentence of item 5 in the list describing figure 30 as follows:

The application client start a new application client thread to perform ~~performs~~ an **Execute Command** remote procedure call by means of the **Send SCSI Command** SCSI protocol service as described in 1).

Change item 8 in the list describing figure 30 as follows:

The ~~LLP~~ SCSI initiator port delivers an **Command Complete Received** confirmation containing the service response and status to the application client thread ~~that contains the service response and status~~.

In 6.10 (Task management function example), made the following changes.

In figure 31, change "Application Client" to "Application Client Thread".

Change item 1 in the list describing figure 31 as follows:

The application client thread issues a task management request by invoking the **Send Task Management Request** SCSI protocol service.

Change item 4 in the list describing figure 31 as follows:

A **Received Function-Executed** confirmation is received by the application client thread.