From: Cris Simpson *cris.simpson@intel.com*
Date: March 21, 2002
Title: LOGOUT signals: Model concerns

References

T10/02-121r0	Draft minutes, SRP WG March 14, 2002
SAM-2	SCSI Architecture Model-2, revision 23
SBC-2	SCSI Block Commands-2, revision 05a
SPC-3	SCSI Primary Commands-3, revision 05
SRP	SCSI RDMA Protocol, revision 12, T10/01-328r4

1 Introduction

Recent discussions within the SRP WG have suggested the need for a “logout signal”, allowing certain resources to be released by targets and logical units. Defining this signal requires a model for what it does and does not do. Clause 2 introduces new concepts to support the signal, and highlights some apparent problems with the current direction. Clause 3 includes some material from SAM-2, with comments.

2 Model and Issues

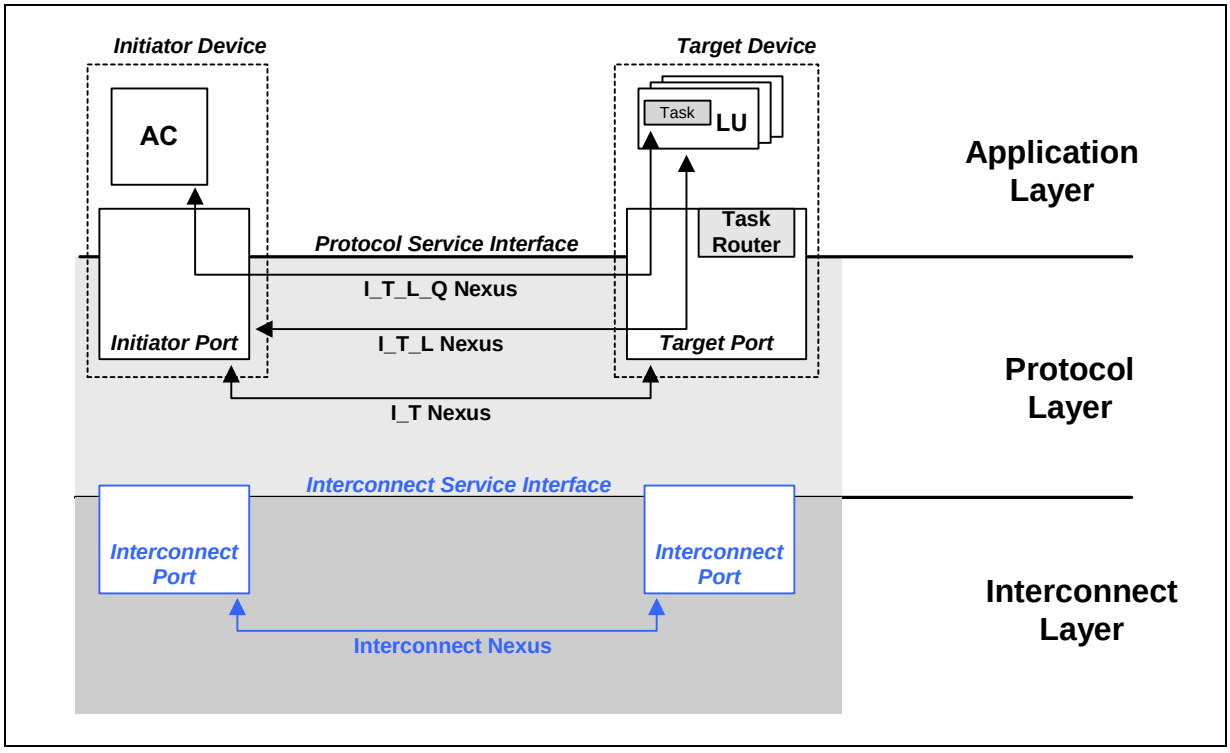


Figure 1: Proposed Interconnect Port Model

A complete ‘logout signal’ mechanism must consider the interconnect layer, not currently discussed in SAM-2. The figure introduces *interconnect ports*, which serve as the connection between the protocol and interconnect layers. The association of two interconnect ports is called an *interconnect nexus*, paralleling

the SCSI I_T nexus at the protocol layer. In SRP, such an interconnect nexus (IN) is called an *RDMA Channel*.

SRP supports a multi-channel mode in which multiple INs form a single I_T nexus, but when one of the INs is disestablished, all tasks delivered to the target over that IN are to be aborted. This suggests the need for another signal, INTERCONNECT NEXUS LOST. This signal would be sent from the interconnect port to its associated SCSI (initiator or target) port. This is distinct from the proposed LOGOUT signal, which presumably would be generated when the last IN of an I_T nexus was disestablished. (This could be confusing for SRP because SRP_LOGIN_REQ and SRP_X_LOGOUT_REQ, in multi-channel mode, add/remove INs from the I_T nexus.)

Several issues discussed at the March 14 SRP WG were resolved by tying them to the LOGOUT signal, with the intention that the signal concept would be introduced into other relevant specifications(e.g., SAM, SPC, SBC). While valuable, LOGOUT does not solve the problems completely.

(Section numbers based on T10/02-121r0):

4.3.1.3 Existing tasks at logout

The WG decided to make logout an SRP protocol-specific event that caused all the initiator's tasks to be aborted. Tasks are typically aborted by an application client issuing an ABORT TASK or ABORT TASK SET task management function request to a logical unit's task manager. The whole point of this discussion being that the application client cannot send a task management request, it must be generated elsehere, namely within the target port.

There appears to be no support in SAM-2 for a target port generating task management requests.

The requirement that a target port abort all tasks issued by a particular initiator creates an implicit requirement that target ports supporting multiple initiators maintain a list of the pending tasks issued by each initiator, henceforth called a *per-initiator task list* (PITL). In addition, SRP's multiple channel support appears to require a subset of the per-initiator task list, based on the interconnect nexus, a *per-initiator nexus task list*[1] (PINTL). A PITL is the union of its subsidiary PINTLs.

The SRP spec does not define the order in which tasks are to be aborted by the target port.

4.3.1.6 Buffered data for EXTENDED COPY, 4.3.1.7 Buffered data for XOR commands

At the 3/14 meeting, the recommendation was that SPC3 and SBC2 would add a "protocol-specific behavior" statement, and that SRP's behavior was to clear the buffered data at logout. It is unclear whether this action occurs upon LOGOUT or upon INTERCONNECT NEXUS LOST.

Ignoring that question for now, how *does* the target port effect the clearing of that data? For example, SPC3r05 says the following for EXTENDED COPY (7.16.2):

> The copy manager shall discard the COPY STATUS data when:
>
> a) A RECIEVE COPY RESULTS command with COPY STATUS service action is received from the same initiator with a matching list identifier;
>
> b) When another EXTENDED COPY command is received from the same initiator and the list identifier matches the list identifier associated with the data preserved for the COPY STATUS service action;
>
> c) When the copy manager detects a hard reset condition; or
>
> d) When the copy manager requires the resources used to preserve the data.

1. Suggestions for better names are solicited.

Which method should a target port use to clear the data? Option *c)* is too dramatic, option *d)* is too non-deterministic, leaving *a)* and *b)* (or a new action TBD). Both options appear to require that the PINTL entry contain much more than just a 'pending' flag.

SCBC2r05a says:

> **4.2.3.7 XOR data retention requirements**
>
> The target shall retain XOR data while awaiting retrieval by an XDREAD command until performing one of the following events: a matching XDREAD command, logical unit reset, CLEAR TASK SET, ABORT TASK if the task matches the pending XDREAD, or ABORT TASK SET.

It appears that the method to clear buffered data is command-specific.

4.3.1.9. Preexisting ACA,unit attention,and deferred error conditions

On March 14, the WG decided that SRP should treat these the same as clearing tasks (for that initiator), but worried whether SAM allows this behavior. SAM-2 suggests *not* for ACA (at least not via ABORT TASK):

> **(SAM-2) 6.2 ABORT TASK**
>
> [...] Previously established conditions, including MODE SELECT parameters, reservations, ACA, and CA shall not be changed by the ABORT TASK function.

Deferred Errors -

> **(SPC-3) 7.25.4 Deferred errors**
>
> The deferred error indication may be sent at a time selected by the device server through use of the asynchronous event reporting mechanism (see SAM-2), if AER is supported by both the application client and device server.

I'm unable to find any way to tell a device server to discard unsent deferred error indications. In any case, the LU would associate the indication with an initiator, not an IN.

(SRP has no requirement that such an AER be sent on the IN that delivered the original command.)

3 SAM-2 references

SCSI Ports

3.1.93 SCSI initiator port: A SCSI initiator device object [that] acts as the connection between application clients and the service delivery subsystem through which requests and responses are routed. In all cases when this term is used it refers to an initiator port or a SCSI target/initiator port operating as a SCSI initiator port.

3.1.103 SCSI target port: A SCSI target device object that contains a task router and acts as the connection between device servers and task managers and the service delivery subsystem through which requests and responses are routed. When this term is used it refers to a SCSI target port or a SCSI target/initiator port operating as a SCSI target port.

(4.2) The SCSI distributed service model

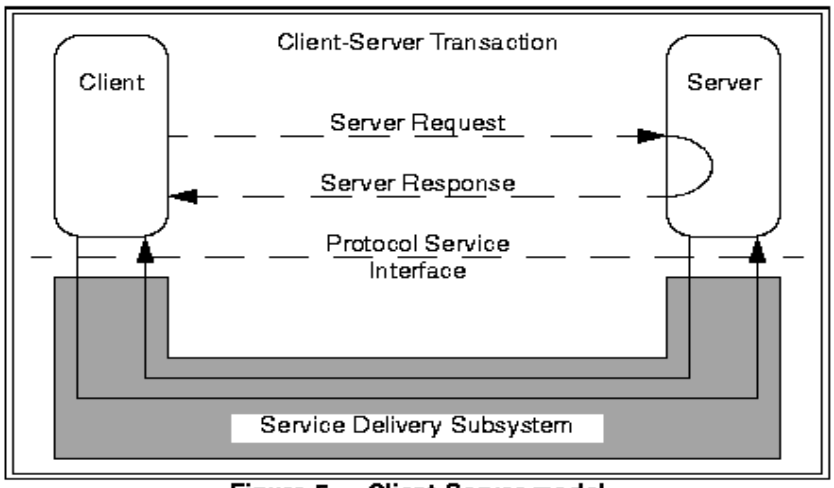


Figure 5 — Client-Server model

Here, the Service Delivery Subsystem is the stuff below the Protocol Service Interface (i.e., Protocol and Interconnect layers).

Service Delivery Subsystem

3.1.112 service delivery subsystem: That part of a SCSI I/O system that transmits service requests to a logical unit or target and returns logical unit or target responses to an initiator.

4.6 The service delivery subsystem

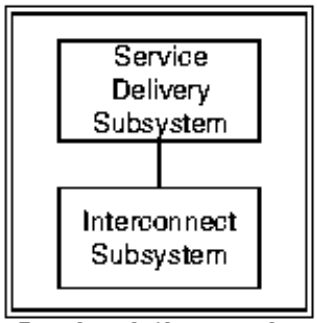


Figure 10 — Service delivery subsystem model

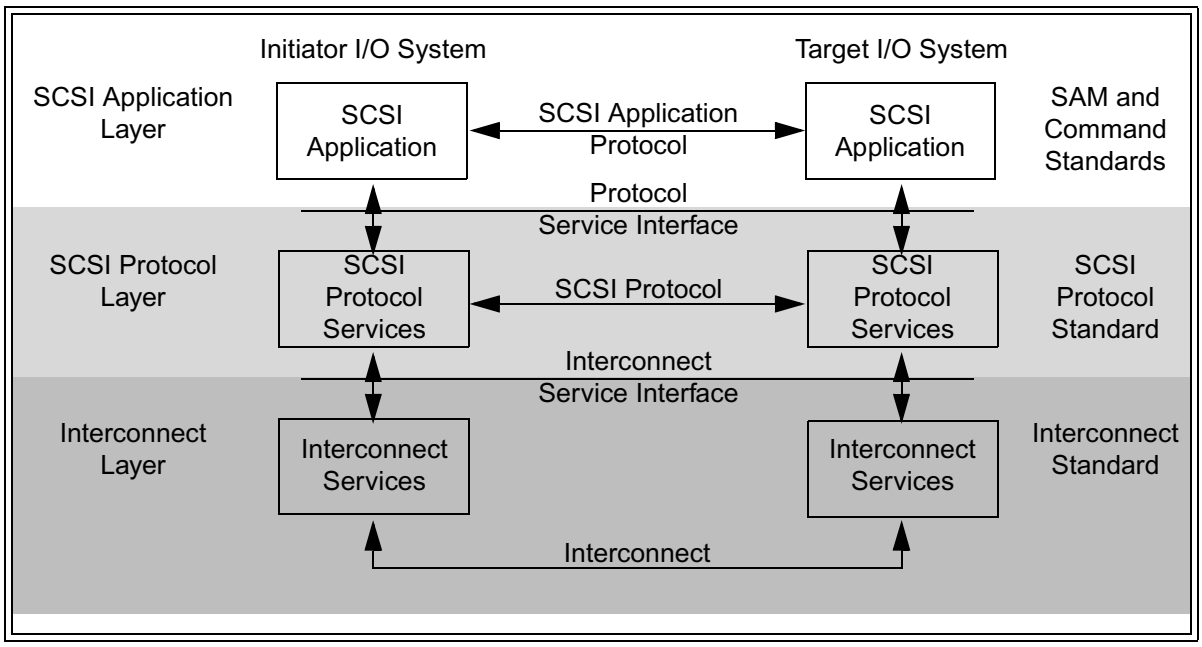
The service delivery subsystem connects SCSI ports (see 3.1.94) and is composed of an interconnect subsystem (see figure 10). *"composed of" implies exclusively.*

Interconnect Subsystem

3.1.54 interconnect subsystem: One or more physical interconnects that appear as a single path for the transfer of information between SCSI devices in a domain.

3.1.97 SCSI protocol layer: The protocol and services used by a SCSI application layer to transport data representing a SCSI application protocol transaction.

4.14 The SCSI model for distributed communications



SAM-2 Figure 25 — Protocol service reference model

SCSI application layer: Contains the clients and servers that originate and process SCSI I/O operations by means of a SCSI application protocol.

SCSI protocol layer: Consists of the services and protocols through which clients and servers communicate; and

Physical interconnect layer: Comprised of the services, signaling mechanism and interconnect subsystem needed for the physical transfer of data from sender to receiver. In the SCSI model, the physical interconnect layer is known as the service delivery subsystem.