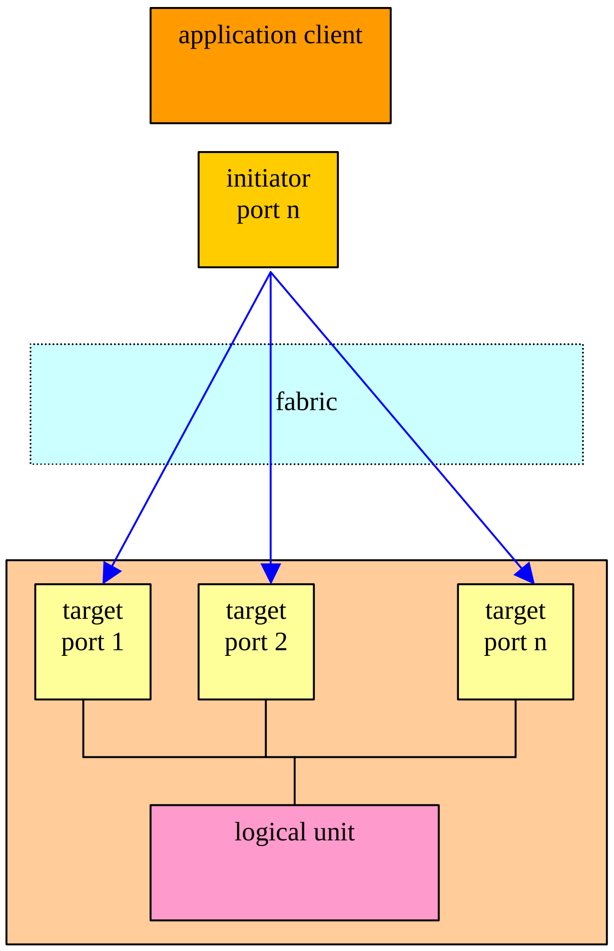
To:	T10 Technical Committee
From:	Rob Elliott, Compaq Computer Corporation (Robert.Elliott@compaq.com)
Date:	17 April 2001
Subject:	SPC-3 Letting persistent reservations ignore target ports

Revision History
Revision 0 (5 March 2001) first revision. Not complete, but as presented in CAP working group.
Revision 1 (17 April 2001). Removed mode page option; use a bit in the CDB. Removed target port group relationship. Add PR IN report capability.

Related Documents
T10/spc2r19 – SCSI Primary Commands revision 19 (by Ralph Weber)
T10/00-232r6 – Asymmetric target behavior (by Ken Moe) (defines target groups)

Overview
SPC-2 requires that logical units remember the target port through which a reservation was made in addition to the initiator port that originated the reservation. An application client wishing to make a reservation needs to run the PERSISTENT RESERVE command through each target port that can route commands to the logical unit.



In many cases, the logical unit doesn't care which target port the reservation came through. It just wants to distinguish between initiators. Requiring reservation commands for each port burdens the application client with issuing extra commands and burdens the logical unit with extra non-volatile storage.

This is only true if all the target ports are in the same SCSI domain. There is no way for the logical unit or a target port to determine this; the application client must determine this (e.g. by reading how many relative ports are available and making sure it has a path through the fabric to each one).

This proposal allows persistent reservation commands to ignore the target port. It adds a bit to the PERSISTENT RESERVE OUT command indicating the reservation command applies to all target ports within the target port group.

It also adds text highlighting the requirement that the target port be included in the reservation identification data. This was previously only mentioned in the multiport section (outside the persistent reservation model section) and in the SAM-2 definition of initiator identifier.

Additionally, a new PERSISTENT RESERVE IN service action is proposed to report persistent reservation feature capabilities. There are several other ways to do this:

- If the target checks reserved bits and returns an error, software knows if it is supposed. Not all targets do this, since it is optional behavior.
- The command support data (CmdDt) bit may be set in INQUIRY returns a bit mask of which fields are supported in a specified opcode's CDB. Not all targets support this feature, since it is optional behavior.
- Add a bit to the standard INQUIRY data indicating support. Unused bits are getting rare.

Reporting the feature in PR IN keeps it in the persistent reservation domain. Reporting the new feature can be required with less impact than requiring checking reserved bits or CmdDt support.

<u>Suggested Changes</u>
Text is from SPC-2 revision 19.

5.5.1 Reservations overview
Reservations may be used to allow a device server to execute commands from a selected set of initiators. The device server shall reject commands from initiators outside the selected set of initiators by uniquely identifying initiators using protocol specific mechanisms.
,,,

5.5.3.1 Overview of the Persistent Reservations management method
…
The PERSISTENT RESERVE OUT and PERSISTENT RESERVE IN commands provide the basic mechanism for dynamic contention resolution in multiple initiator systems using multiple port targets. Before a persistent reservation may be established, an initiator shall register with a device server using a reservation key. Reservation keys are necessary to allow:
a) authentication of subsequent PERSISTENT RESERVE OUT commands;
b) identification of other initiators that are registered;
c) identification of the reservation key(s) that have an associated reservation;
d) preemption of a persistent reservation from a failing or uncooperative initiator; and
e) multiple initiators to participate in a reservation.

The reservation key provides a method for the application client to associate a protocol-independent identifier with an initiator ~~on a specific port of a device server~~. The reservation key is used in the PERSISTENT RESERVE IN command to identify which initiators are registered and which initiator, if any, holds the reservation. The reservation key is used in the PERSISTENT RESERVE OUT command; to register an initiator, to verify the initiator issuing the PERSISTENT RESERVATION OUT command is registered, and to specify which initiator's registration or persistent reservation to preempt.
…

5.5.3.2 Preserving persistent reservations

…

The device server shall preserve the following information for each registration across any reset, and if the persist through power loss capability is enabled, across any power cycle:

a) Initiator port identifier;
aa) relative port identifier of the target port;
b) reservation key; and
c) when supported by the protocol, the initiator port's world wide identification.

The device server shall preserve the following reservation information across any reset, and if the persist through power loss capability is enabled, across any power cycle:

a) Initiator port identifier;
aa) relative port identifier of the target port;
b) reservation key;
c) scope;
d) type; and
e) when supported by the protocol, the initiator port's world wide identification.

For those protocols for which the initiator port's world wide identification is available to the device server the initiator port's world wide identification shall be used to determine if the initiator identifier has changed. This determination shall be made at any time the target detects that the configuration of the system may have changed. If the initiator identifier changed, the device server shall assign the new initiator identifier to the existing registration and reservation of the initiator port having the same world wide identification.

[Editor's note: the references to "initiator port's world wide identification" should change to a generic SAM-2 term like "initiator port name." That is left for another proposal.]

…

5.5.3.4 Registering

…

In response to a PERSISTENT RESERVE OUT with a REGISTER or a REGISTER AND IGNORE EXISTING KEY service action the device server shall perform a registration by doing the following as an uninterrupted series of actions:

a) Process the registration request regardless of any persistent reservations;
b) process the APTPL bit;
c) ignore the contents of the SCOPE and TYPE fields;
d) map the reservation key to the registering initiator using the initiator identification and, if available, the initiator port's world wide identification;
e) register the reservation key without changing any a persistent reservation that may exist; and
f) retain the reservation key and associated information.

5.6 Multiple port and multiple initiator behavior

SAM-2 specifies the behavior of logical units being accessed by more than one initiator. Additional service delivery ports provide alternate service delivery paths through which the device server may be reached and may also provide connectivity for additional initiators. An alternate path may be used to improve the availability of devices in the presence of certain types of failures and to improve the performance of devices whose other paths may be busy.

If a SCSI device has more than one service delivery port, the arbitration and connection management among the service delivery ports is vendor specific. If one service delivery port is being used by an initiator, accesses attempted through other service delivery port(s) may:

a) receive a status of BUSY; or
b) be accepted as if the other service delivery port(s) were not in use.

The device server shall indicate the presence of multiple ports by setting the MULTIP bit to 1 in its standard INQUIRY data.

For the purposes of handling reservations, other initiators are defined as all initiators on the same service delivery port except the initiator holding the reservation and all initiators on all other service delivery ports. Only the following operations allow an initiator to interact with the tasks of another initiator, regardless of the service delivery port:

a) the PERSISTENT RESERVE OUT with PREEMPT service action preempts persistent reservations for other initiators (see 5.5.3.6.3);
b) the PERSISTENT RESERVE OUT with PREEMPT AND ABORT service action preempts persistent reservations and all tasks for other initiators (see 5.5.3.6.4);
c) the PERSISTENT RESERVE OUT with CLEAR service action releases persistent reservations and removes reservation keys for all initiators (see 5.5.3.6.5);
d) the TARGET RESET task management function releases reservations established by the reserve/release method and removes all tasks for all initiators for all logical units in the target ~~and for all initiators~~ (see SAM-2). Persistent reservations remain unmodified;
e) the LOGICAL UNIT RESET task management function releases reservations established by the reserve/release method and removes all tasks for all initiators for the addressed logical unit and any logical units issuing from it in a hierarchical addressing structure (see SAM-2). Persistent reservations remain unmodified; and
f) the CLEAR TASK SET task management function removes all tasks for all initiators for the selected logical unit ~~for all initiators~~. Most other logical unit states remain unmodified, including MODE SELECT parameters, reservations, and ACA (see SAM-2).

[Editor's note: just making d) e) and f) use the same order]

7.11.1 PERSISTENT RESERVE OUT command introduction

…

Table 73. PERSISTENT RESERVE OUT command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (5Fh)							
1	Rsvd	Rsvd	ALL TARGET PORTS	SERVICE ACTION				
2	SCOPE				TYPE			
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	PARAMETER LIST LENGTH (18h)							
8								
15	CONTROL							

…

The ALL TARGET PORTS bit indicates the command applies to all target ports capable of routing commands and task management functions to the logical unit, and that all relative port identifiers shall be included in the reservation information. Application clients should ensure that all target ports are in the same SCSI domain before setting the ALL TARGET PORTS bit. Application clients may use the PERSISTENT RESERVE IN command's REPORT CAPABILITIES service action to determine whether the device server supports the ALL TARGET PORTS bit.

7.10.2 PERSISTENT RESERVE IN service actions

7.10.2.1 Summary of PERSISTENT RESERVE IN service actions

The service action codes for the PERSISTENT RESERVE IN command are defined in table 67.

Table 67 — PERSISTENT RESERVE IN service action codes

Code	Name	Description
00h	READ KEYS	Reads all registered Reservation Keys
01h	READ RESERVATION	Reads the current persistent reservations
02h	REPORT CAPABILITIES	Returns capability information

02h – 1Fh	Reserved	Reserved

7.10.2.x Report Capabilities

The REPORT CAPABILITIES service action requests that the device server return a parameter page indicating support for various persistent reservation features.

Table 73. REPORT CAPABILITIES parameter data

Byte\Bit	7	6	5	4	3	2	1	0
0	Rsvd	Rsvd	Rsvd	Rsvd	Rsvd	ELEMENT SCOPE	APTPL	ALL TARGET PORTS
1	RESERVED							
2	RESERVED							
3	RESERVED							
4	RESERVED							
5	RESERVED							
6	RESERVED							
7	RESERVED							

The ALL TARGET PORTS bit indicates that the device server supports the ALL TARGET PORTS BIT in the PERSISTENT RESERVE OUT command.

[Editor's note: bits for existing optional features could be added too.]
The APTPL bit indicates that the device server supports the APTPL in the PERSISTENT RESERVE OUT command.

The ELEMENT SCOPE bit indicates that the device server supports a SCOPE value of ELEMENT_SCOPE in the persistent reservation commands.