

To: T10 Technical Committee
 From: Rob Elliott, Compaq Computer Corporation (Robert.Elliott@compaq.com)
 Date: 24 April 2001
 Subject: SPC-3 Letting persistent reservations ignore initiator ports

Revision History

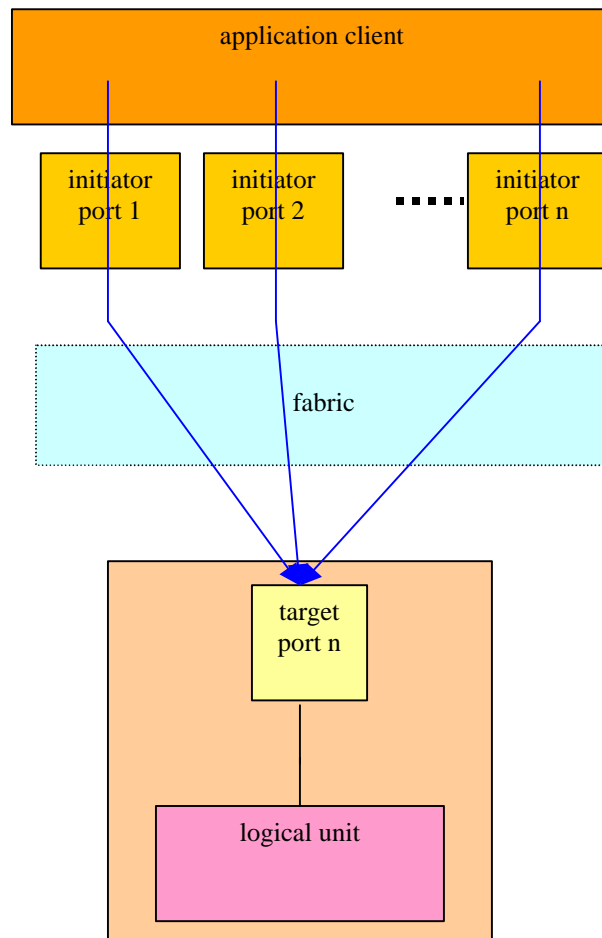
Revision 0 (7 March 2001). First revision. Not complete, but as presented in CAP working group.
 Revision 1 (24 April 2001). Reflects input from the March CAP working group. Settles on separate PR OUT service action to register an initiator ID.

Related Documents

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

Overview

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



In many cases, the logical unit doesn't care which initiator port the reservation came through. It just wants to distinguish between applications or operating systems or clusters. Requiring reservation commands for each initiator port burdens the application clients with issuing extra

commands and burdens the logical unit with extra non-volatile storage. Cases include a single host with multiple ports and multiple hosts in a cluster.

This proposal allows persistent reservation commands to use the higher level identifier rather than initiator port identifier.

Two approaches:

1) Depend on native protocol support. In some protocols (SRP and iSCSI) the initiator port definition is at a high level and the problems can be minimized. Applications can share identifiers if they want. In Fibre Channel, however, the initiator port is the Fibre Channel port. A higher level identifier such as the Fibre Channel node name or FC-GS-3 platform name would be preferable. Unfortunately, node names have a history of inconsistent implementations and platform name usage has not been standardized.

There is no generic term for this level of name in SAM-2. It could be defined as the initiator device identifier hinted at in SAM-2. For SRP and iSCSI, the initiator device identifier would just be the same as the initiator port identifier. For Fibre Channel, it would be the new identifier.

Table 1. Possible initiator device identifiers

Protocol	initiator device identifier
SPI	SCSI device ID
FCP	FC-GS-3 platform name (256 byte string)
SRP	initiator port identifier (a 128-bit worldwide unique ID)
iSCSI	initiator port identifier (255 byte string)

When communicating across multiple SCSI transports (e.g. InfiniBand SRP to Fibre Channel FCP to TCP/IP iSCSI), the endpoint transports might support device identifiers but the intermediate transports do not. Just adding a bit to the PERSISTENT RESERVE OUT CDB or adding a field in a mode page is insufficient.

A new command could be defined whose payload is the purported initiator device identifier as known to the application client. When the device server receives the command, it checks the initiator device identifier reported by the transport and compares it to the payload. If the identifiers match, the device server reports command success. If they do not match, the device server reports command failure and the initiator device identifier cannot be used.

However, if the application client is going to run such a command, it might as well run another command that explicitly registers the initiator device identifier with the target. This avoids the need for intermediate transports to pass through the initiator device identifier. This makes it like the AccessID used by access controls. Any initiator that has registered with this ID is granted the reservation.

Application clients only enable this mode after they have tested the SCSI domain for successful initiator device identifier delivery.

2) An identifier could be passed as data in a new service action for the PERSISTENT RESERVE OUT command. An application would need to run this one time through each initiator port so the target can correlate the provided initiator device identifier with the dynamic initiator port identifier. Once done, the application only needs to run PR OUT commands through one initiator port, provided that it sets a new bit in the PR OUT CDB to indicate that the initiator device identifier needs to be used instead of the initiator port identifier.

This case works over any transports.

Case 2 is detailed below.

Suggested Changes

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.

...

An initiator may choose to identify itself with an initiator device identifier rather than the initiator port identifier defined by the transport. The initiator may create an association between an initiator device identifier and its initiator port identifier with the PERSISTENT RESERVE OUT command's ASSOCIATE INITIATOR DEVICE IDENTIFIER service action. Subsequent PERSISTENT RESERVE OUT commands with the USE INITIATOR DEVICE IDENTIFIER bit set will apply to all initiator ports that have associated with the same initiator device identifier.

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 or initiator device 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 or initiator device 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.

...

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 port identifier ~~creation- or initiator device identifier~~ 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 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 the selected logical unit for all initiators. Most other logical unit states remain unmodified, including MODE SELECT parameters, reservations, and ACA (see SAM-2).

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	<u>USE INITIATOR DEVICE IDENTIFIER</u>	<u>Rsvd see 01-099</u>	SERVICE ACTION				
2	SCOPE				TYPE			
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	PARAMETER LIST LENGTH (18h)							
8								
15	CONTROL							

The USE INITIATOR DEVICE IDENTIFIER bit indicates that the device server shall use the initiator device identifier rather than the initiator port identifier.

Commands shall be checked for reservation conflicts using the initiator device identifier rather than the initiator port identifier.

...

7.11.2 PERSISTENT RESERVE OUT Service Actions

When processing the PERSISTENT RESERVE OUT service actions, the device server shall increment the generation value as specified in 7.10.3.

The PERSISTENT RESERVE OUT command service actions are defined in table 74.

Table 74 — PERSISTENT RESERVE OUT service action codes

Code	Name	Description	GENERATION field incremented (see 7.10.3)
00h	REGISTER	Register a reservation key with the device server (see 5.5.3.4).	Yes
01h	RESERVE	Creates a persistent reservation having a specified SCOPE and TYPE (see 5.5.3.5). The SCOPE and TYPE of a persistent reservation are defined in 7.10.4.2 and 7.10.4.3.	No
02h	RELEASE	Releases the selected reservation for the requesting initiator (see 5.5.3.6.2).	No
03h	CLEAR	Clears all reservation keys and all persistent reservations (see 5.5.3.6.5).	Yes
04h	PREEMPT	Preempts persistent reservations from another initiator (see 5.5.3.6.3).	Yes
05h	PREEMPT AND ABORT	Preempts persistent reservations from another initiator and aborts all tasks for all initiators registered with the specified reservation key (see 5.5.3.6.3 and 5.5.3.6.4).	Yes
06h	REGISTER AND IGNORE EXISTING KEY	Register a reservation key with the device server (see 5.5.3.4).	Yes
<u>07h</u>	<u>ASSOCIATE INITIATOR DEVICE</u>	<u>Associate an initiator device identifier with the initiator port identifier in the device server.</u>	<u>No</u>

	<u>IDENTIFIER</u>		
<u>08h</u>	<u>UNASSOCIATE INITIATOR DEVICE IDENTIFIER</u>	<u>Remove the association of an initiator device identifier with the initiator port identifier in the device server.</u>	<u>No</u>
<u>07h-09h</u> – 1Fh	Reserved		

The parameter list values for the association service actions are specified in 7.11.4. The parameter list values for ~~each~~ the other service actions are specified in 7.11.3.

7.11.3 PERSISTENT RESERVE OUT parameter list

The parameter list required to perform the PERSISTENT RESERVE OUT command for all service actions except the association service actions is defined in table 75. All fields shall be sent on all such PERSISTENT RESERVE OUT commands, even if the field is not required for the specified service action and scope values.

Table 73. PERSISTENT RESERVE OUT parameter list

Byte\Bit	7	6	5	4	3	2	1	0
0	RESERVATION KEY							
7								
8	SERVICE ACTION RESERVATION KEY							
15								
16	SCOPE-SPECIFIC ADDRESS							
19								
20	RESERVED							APTPL
21	RESERVED							
22	OBSOLETE							
23								
24	INITIATOR DEVICE IDENTIFIER (16 bytes)							
39								

7.11.4 Association service action parameter list

The parameter list required to perform the PERSISTENT RESERVE OUT command with the ASSOCIATE INITIATOR DEVICE IDENTIFIER and UNASSOCIATE INITIATOR DEVICE IDENTIFIER service action is defined in table 755.

Table 755. PERSISTENT RESERVE OUT parameter list

Byte\Bit	<u>7</u>	<u>6</u>	<u>5</u>	<u>4</u>	<u>3</u>	<u>2</u>	<u>1</u>	<u>0</u>
<u>0</u>	<u>INITIATOR DEVICE IDENTIFIER</u>							
<u>15</u>								
<u>16</u>	<u>RESERVED</u>							
<u>23</u>								
<u>39</u>								

The INITIATOR DEVICE IDENTIFIER field is a value provided by the application client to identify the initiator device (i.e. a collection of initiator ports and application clients).

For an ASSOCIATE INITIATOR DEVICE IDENTIFIER service action, the device server shall associate the initiator port identifier of the initiator port used to send the PERSISTENT RESERVE OUT command with the specified initiator device identifier. Any previous association for that initiator port identifier shall be replaced.

For an UNASSOCIATE INITIATOR DEVICE IDENTIFIER service action, the device server shall remove the association of the initiator port identifier with the specified initiator device identifier. If no such association exists, the device server shall return a CHECK CONDITION status. The

sense key shall be set to ILLEGAL REQUEST and the additional sense data shall be set to NO INITIATOR DEVICE IDENTIFIER ASSOCIATION [new ASCQ].

7.10.2 PERSISTENT RESERVE IN service actions

[Editor's note: 01-099 also proposes adding the PR IN REPORT CAPABILITIES service action. This proposal adds another bit to those proposed.]

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	<u>REPORT CAPABILITIES</u>	<u>Returns capability information</u>
<u>02h-03h</u> – 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	<u>Rsvd</u>	<u>Rsvd</u>	<u>Rsvd</u>	<u>Rsvd</u>	<u>USE INITIATOR DEVICE IDENTIFIER</u>	<u>SEE 01- 099</u>	<u>SEE 01- 099</u>	<u>SEE 01- 099</u>
1	<u>RESERVED</u>							
2	<u>RESERVED</u>							
3	<u>RESERVED</u>							
4	<u>RESERVED</u>							
5	<u>RESERVED</u>							
6	<u>RESERVED</u>							
7	<u>RESERVED</u>							

The USE INITIATOR DEVICE IDENTIFIER bit indicates that the device server supports the USE INITIATOR IDENTIFIER bit and the IDENTIFY INITIATOR service action in the PERSISTENT RESERVE OUT command.