May 20, 2001

To: Deborah Donovan, NCITS Secretariat
Cc: PSA Dept., ANSI
John Lohmeyer, T10 Chairman
Ralph Weber, SPC-2 Technical Editor
From: Roger Cummings, VERITAS Software
Subject: Public Review Comment on dpANS SCSI Primary Commands - 2

Please accept this document as a Public Review comment by VERITAS Software against NCITS 351, SCSI Primary Commands - 2 (SPC-2).

VERITAS will be pleased to have representation at the next meeting of Technical Committee T10 to discuss this comment, and the changes suggested herein, in detail.

Regards,

Roger Cummings
VERITAS Software
roger.cummings@veritas.com
407.531.7257

PUBLIC REVIEW COMMENT on NCITS 351 SPC-2

Introduction

VERITAS has decided to take the serious step of submitting a Public Review comment on this draft standard because of experience gained in testing prototype equipment that is compatible with the Persistent Reservation requirements in the SPC-2 draft along with our clustering applications.

Background

One of the major uses for the Persistent Reservations defined for the first time in SPC-2 is expected to be in clusters. VERITAS has a cluster application that can support up to 32 nodes, and which assumes a Storage Area Network shared by all of those nodes is used to provide the connection to storage. VERITAS there intends to use Persistent Reservations to provide controlled access from a subset of the nodes in each cluster to each storage device, via a Write Exclusive – Registrants Only Persistent Reservation. When the cluster is created, each node will register with a set of storage devices as directed by the management logic, and the reservation will be initiated.

Problem

When one of the nodes in the cluster is to be taken offline for maintenance, the shutdown procedure will cause the node to remove its registration from its set of storage devices. If this node was the one that originally established the reservation, then the following text from subclause 5.5.3.6.1 (page 32) applies:

When a reservation key has been removed, no information shall be reported for that unregistered initiator in subsequent READ KEYS service action(s) until the initiator is registered again (see 5.5.3.4). Any persistent reservation associated with that unregistered initiator shall be released. If that released persistent reservation was of the type Write Exclusive – Registrants Only or Exclusive Access – Registrants Only the device server shall establish a unit attention condition or all registered initiators other than the initiator that issued the PERSISTENT RESERVE OUT command with PREEMPT or PREEMPT AND ABORT service action. The sense key shall be set to UNIT ATTENTION and the additional sense data shall be set to RESERVATIONS RELEASED.

VERITAS has a number of problems with both the definition and the text in this paragraph, as follows:

1) This definition has a significant impact on the remaining nodes in the cluster, each of which now have to execute the following process:
   a. Respond to the resulting Unit Attention condition;
   b. Clear the resulting Auto Contingent Allegiance condition;
   c. Interact with the other nodes to determine which should reissue the reservation;
   d. Reissue the Write Exclusive – Registrants Only Persistent Reservation if selected;
   e. Communicate with all other nodes that the reservation has been re-established;
   f. Reissue all of the commands that completed with ACA Active status.
2) The end result of the process in 1) above is only to recreate a reservation that is almost identical to the reservation that existed before the node was taken offline!
3) The process in 1) above is a significant impediment to scalability. We are attempting to expand our clustering products to support hundreds of nodes, and tens of thousands of disk drives. For each node/disk pair the process in 1) above has to be performed at a time when those nodes are under full load since a configuration change is being processed and cluster resources are reduced. Note that this happens even in the case of a graceful shutdown of a node.
4) There is a time period between steps a. and d. in the process described in 1) above where an unregistered node can access the device and cause data corruption.
5) The text in the first paragraph under Problem above that states "If this node was the one that originally established the reservation" is actually an assumption on the part of VERITAS. The text in the quoted paragraph states "*Any persistent reservation associated with that unregistered initiator*". The verb "associated" is only used elsewhere in the Persistent Reservation section in terms of keys associated

with a reservation. Elsewhere (e.g. in 5.5.3.3.3 b)), the text describes “*the initiator that holds the persistent reservation”.* Is the assumption correct?

Suggested Changes

VERITAS would like to amend the definition of the Registrants Only reservations such that the reservation survives as long as there are registrations in place for that Logical Unit. Thus even if the initiator that holds the reservation unregisters, the reservation survives. This will convert the Registrant Only reservations to a true peer basis, and will mean that there is NO processing required on the remaining cluster nodes after a node unregisters. There will also be no time period during that process during which a competing reservation can be established or data can be corrupted, and no Unit Attention conditions have to be created.

Suggested text for the paragraph quoted from subclause 5.5.3.6.1 (page 32) above is:

When a reservation key has been removed, no information shall be reported for that unregistered initiator in subsequent READ KEYS service action(s) until the initiator is registered again (see 5.5.3.4). Any Registrants Only persistent reservation held by that unregistered initiator shall be released only when no other registrations exist for that Logical Unit, otherwise the reservation shall not be effected. Any non Registrants Only persistent reservation held by that unregistered initiator types held by that initiator shall be released.

Also, the following additional paragraph is suggested following the one above:

When a Registrants Only persistent reservation continues after the initiator that originally created the reservation unregisters, the device server shall report the reservation key of any of the initiators with remaining registrations in a PERSISTENT RESERVE IN command with a READ RESERVATION service action.