

Only if Reserved Proposal

Roger Cummings
VERITAS Software

Background

- An idea that arose from the PR work
 - But is completely separate from it
- This feature adds value even if you will only ever use (original) Reservations

Problem

- Used to think our apps guys were worried about a reservation disappearing & the backup app not completing
 - That was 180 degrees wrong
- They're actually worried about a reservation disappearing and the app still working
 - Opportunities for data corruption
 - Almost undetectable

Problem

- Applications, Platforms, and Driver ALL issue many Reserves and Releases in traversing some code paths
 - Each one paranoid about the other
 - No way of testing that a reservation exists OR NOT
 - RESERVE returns exactly the same status if an identical reservation already exists
 - RELEASE returns exactly the same status if no reservation previously exists
 - Our record is 26 reserves in opening a tape volume

Test for A Reservation?

- Being able to read if a reservation exists (like PR) would be useful
- Inherent problem – there’s always a “hole” or window between the test and sending a command
- Would probably only mean that 26 reserves get replaced by 1 reserve and 25 tests
- There has to be a better way.....

What do the apps guys want

- For commands to succeed Only If (a device) is Reserved to the issuing Initiator
 - If there's no reservation, then command not performed & status (new ASCQ) returned
 - Conflicts work as they always did
- Proposing a bit in the Control Mode Page called OIR to control this behavior
 - Byte 5 bit 7 is our suggestion

OIR Bit Part 1

A only if reserved (OIR) bit set to one specifies that the device server shall only perform a command if a reservation or persistent reservation exists which allows access to the I_T nexus from which the command was received. When OIR is one and a command is received from an I_T nexus for which no reservations exists, the device server shall not perform the command.

OIR Bit Part 2

When OIR is one and a command is received from an I_T nexus for a logical unit or element upon which no reservation or persistent reservation exists, the device server shall terminate the command with CHECK CONDITION status, and shall set the sense key to NOT READY and the additional sense code to NOT RESERVED.

OIR Bit Part 3

For a list of commands affected by the OIR bit, see the tables that define the commands allowed in the presence of various reservations in this standard (Table 31) and in the command standard (see 3.1.17) for the specific device type. Any command which has “Conflict” in any column is affected by OIR bit.

Reuse Conflict check

Case	Today	With OIR
Reserved by another I	Conflict check RC if found	Conflict check RC if found
No Reservation	No check	Conflict check NR if found
Reserved by this I	No check	No Check

Effect

- This can remove the need for an app to ever issue a reservation in normal codepath
 - Have one simple subroutine – if ever get “Not Reserved” then issue a Reservation

Pros and Cons

- Pro
 - No holes as in testing case
 - Permits Significant simplification of apps
 - Reuses existing conflict checking logic
- Con
 - When OIR set, need to make a Reservation to get the Mode Set issued to be able to reset it

Requests of CAP

- Accept the definition and request that the editor assign a Control Mode page bit for this function, and include the OIR bit text
- Assign a new ASC/ASCQ for “Not Ready”/”No Reservation”