

## Re: Partitioned cluster question (reboot during lost quorum)

---

*Source:* <http://unix.derkeiler.com/Newsgroups/comp.os.vms/2006-04/msg00954.html>

---

- *From:* Hoff Hoffman <[~~hoff-remove-this@xxxxxx~~](mailto:hoff-remove-this@xxxxxx)>
  - *Date:* Thu, 20 Apr 2006 14:27:12 GMT
- 

JF Mezei wrote:

Hoff Hoffman wrote:

If you want to boot outside the cluster, then I tend to prefer to avoid enabling NISCS\_LOAD\_PEA0 and I don't load VAXCLUSTER.

No, the idea is to be able to boot one node first before you bring in the other ones, so that first one has to form the cluster and the others join in later.

Welll, my own idea is to be able to boot distinctly and completely and fully outside the cluster, and to thus reduce the chances of creatively configured corruptions.

Consider the last time this has happened, too. I've been at this for over a decade, and I can't recall ever having run into this case.

What I've come to realise is that it is very hard to predict every possible problem/situation.

That's a central treatise of Disaster Tolerance (DT). It's also understood that small failures or critical procedures can derail the entire effort.

The ability to find solutions out of some unpredicted situation requires good understanding of the clustering process and full understanding your configuration/applications so that you can "cheat" the config without jeopardizing data integrity.

My goal is to win with the least effort and the least risk, and — as a rule, I prefer to configure and operate the

## Re: Partitioned cluster question (reboot during lost quorum)

configuration such that I don't have to get "creative". I'd prefer to win by avoiding the battle, because battles are expensive and wasteful. (Yes, I've been re-reading a translation of a very old book, but I digress.) If I can configure to avoid the problem (through hardware or software upgrades, or documented management procedures and sequences), then I will.

VMS cannot make any assumptions when it loses connection with another node. It could be ethernet down, it could be the other node down. VMS (rightly) protects against worse case scenario. But the system manager can obtain additional information (such as confirmation that the others nodes are in fact powered down) which then allows him to gauge the situation and consider that the first node can be up without jeopardizing data integrity, at which point knowledge on how to "cheat" the predefined rules comes in handy to get out of that situation. (and later restore the safe settings).

You're definitely headed toward DT here, and DT is far more involved than it looks and that most folks initially assume. I've been doing DT for a number of years both in the technology area and in the emergency services area, and no plan I've seen has fully survived a disaster — some have done better than others (and the core infrastructure has stayed available), but some plans have been colossal failures.

As a rule, the node with the valid copy will have the highest instantiation, and will be the source for the shadow copy operation,

Your "as a rule" applies well when all disks are local and if you have access to one disk, you have access to all disks. (aka, you always mount all members of a shadowset together in the same mount command, so the shadowing software can make the proper determination of which physical disk has the more recent valid copy.

But when shadowset members appear gradually as nodes boot, then the order in which you boot the nodes becomes critical to ensuring the right physical drive is first mounted into the shadowset. Knowledge of which physical drive has the valid contents comes from knowing the config, the exact state of the cluster before the mishap and its current state. AKA: full situation awareness. And it isn't something which VMS can ascertain by itself. It takes a human to diagnose the situation.

You haven't hit a case here where the automatic recovery will fail, BTW. If the cluster configuration is correct, then the recovery will be correct.

Again, how often is the case you are looking at likely to arise, and are there ways to avoid it entirely?

## Re: Partitioned cluster question (reboot during lost quorum)

So knowing how `expected_votes` works, means you can cheat the default config to allow one node to boot first when you know this is what has to be done.

Cheating is not without its costs and its risks, and the penalties can be severe. I prefer to configure and to operate in a manner that avoids the need for configurational creativity, particularly as forced creativity during the usual chaos can cause its own class of problems.

DT configurations do inherently depend on voice and data communications links, and the critical "fail-over" decision — to split the lobes and to punt the processing in the affected lobe(s) — is left up to the operator(s) in most every full DT configuration I've worked with.

If you want or believe you need to perform a sequence as you appear to want to and if you believe you must bypass the blade guards and if there is no other more supportable means available to either prevent or to provide for or to avoid the requirement, then you will want to practice and document and test and carefully follow the sequence, as you will want to make every effort to avoid the corruptions that can ensue. If you can avoid it, you don't want to require yourself or others — and do remember that your creativity may not be fully appreciated or may not be correctly executed by others — to become reactive within a difficult situation.