Monday, March 31. 2008

Less known Solaris Features: Remote Mirror with AVS - Part 5: Replication Groups

A new scenario: Okay, the filesystem gets replicated now. Let´s assume that we use /dev/rdsk/c1d0s1 for a database. The filesystem and the database partition are used from the same application and it´s important, that the metadata in the database an the binary objects are in sync even when you switched over to the remote site, albeit it´s acceptable to lose the last few transactions, as both sites are 1000km away from each other and synchronous replication is not an option.The problemWhen you use synchronous replication, all is well. But let´s assume you´ve choosen asychronous replication. Under this circumstances a situation can occure, where one queue is processed faster than another, thus the on-disk states of each volume may be consistent in itself, but both volumes may have the state at different point in time, thus leaving the application data model inconsistent.

Such an behaviour is problematic, when you have a database volume and a filesystem volume working together for an application, but the results can be catastropic when you use a database splitted over several volumes.

The solution of this problem would be a mechanism, that keeps the writes to a group of volumes in order for the complete group. Thus is inconsistencies can´t occur.

Replication GroupTo solve such problems, AVS supports a concept called Replication Group. Adding volumes to a replication group has some implications:

- All administrative operations to this group are atomic. Thus when you change to logging mode or start an replication, this happens on all volumes in the group
- The writes to any of the primary volumes will be replicated in the same order to the secondary volumes. The scope of this ordering is the complete group, not the single volume.
- Normally every replication relation has it´s own queue and it´s own queue flusher daemon. Thus multiple volumes can flush their queue in parallel to increase the performance. In case of the Replication group all I/O operations are routed trough a single queue. This may reduce the performace.

How to set up a replication group?Okay, at first we login at theoden, our primary host in our example. We have to add the existing replication to the replication group and configure another replication relation directly in the correct group. I will create an replication group called importantapp.

```
[root@theoden:~]$ sndradm -R g importantapp gandalf:/dev/rdsk/c1d0s1
Perform Remote Mirror reconfiguration? (Y/N) [N]: yWe´ve added the group to the existing group, now we create the new one:
[root@theoden:~]$ sndradm -e theoden /dev/rdsk/c1d1s1 /dev/rdsk/c1d1s0 gandalf /dev/rdsk/c1d1s1 /dev/rdsk/c1d1s0 ip sync g importantapp
Enable Remote Mirror? (Y/N) [N]: yWith sndradm -P you can look up the exact configuration of your replication sets:[root@theoden:~]$ sndradm -P
/dev/rdsk/c1d0s1      ->      gandalf:/dev/rdsk/c1d0s1
autosync: off, max q writes: 4096, max q fbas: 16384, async threads: 2, mode: sync, group: importantapp, state: syncing
/dev/rdsk/c1d1s1      ->      gandalf:/dev/rdsk/c1d1s1
autosync: off, max q writes: 4096, max q fbas: 16384, async threads: 2, mode: sync, group: importantapp, state: syncingOkay, both are in the same group. As before, we have to perform this configuration on both hosts: So we repeat the same steps on the other hosts as well:[root@gandalf:~]$ sndradm -e theoden /dev/rdsk/c1d1s1 /dev/rdsk/c1d1s0 gandalf /dev/rdsk/c1d1s1 /dev/rdsk/c1d1s0 ip sync g importantapp
Enable Remote Mirror? (Y/N) [N]: y
[root@gandalf:~]$ sndradm -R g importantapp gandalf:/dev/rdsk/c1d0s1
Perform Remote Mirror reconfiguration? (Y/N) [N]: yNo we start the replication of both volumes. We can to this in a single step by using the name of the group.[root@theoden:~] sndradm -m -g importantapp
Overwrite secondary with primary? (Y/N) [N]: yVoila, both volumes are in synchronizing mode:[root@theoden:~]$ dsstat
name              t  s    pct role   ckps   dkps   tps  svt
dev/rdsk/c1d0s1   P  SY  89.37  net      -    Inf     0 -NaN
dev/rdsk/c1d0s0              bmp      0     28     0 -NaN
dev/rdsk/c1d1s1   P  SY  88.02  net      -    Inf     0 -NaN
dev/rdsk/c1d1s0              bmp      0     28     0 -NaNTwo minutes later the replication has succeeded, we have now a
```

fully operational replication group:[root@theoden:~]$ dsstat
name t s pct role ckps dkps tps svt
dev/rdsk/c1d0s1 P R 0.00 net - 0 0 0
dev/rdsk/c1d0s0 bmp 0 0 0 0
dev/rdsk/c1d1s1 P R 0.00 net - 0 0 0
dev/rdsk/c1d1s0 bmp 0 0 0 0Now both volumes are in replicating mode. Really easy, it´s just done by adding the group to the replication relations.

Posted by Joerg Moellenkamp in English, Solaris at 16:53

This is the second part 4
Anonymous on Mar 31 2008, 22:14