Saturday, April 26. 2008

Less known Solaris Features: Point-in-time copy with AVS - Part 3: Independent copy

The most important point about independent copies are in their name. The point in time copy is an independent copy of your original data. You can use it on it´s own and the copy doesn´t need the original data to work.
This method is quite similar of doing a copy with dd or cp. At first a full copy is done by the PiT functionality. Now you´ve created the first point in time copy. But now the advantages of tracking the changes in a database come in to the game. Whenever data on the master is changed, the system tracks this in the bitmap volume. At the next resync of the shadow with the master, the system only copies the changed blocks to the shadow. This vastly reduces the time for the update

Deeper diveAt first you have your master volume. It´s the volume with your data. Before configuration of point-in-time copy both shadow and bitmap copy are uninitalized. We use some special manufactured disks for tutorial purposes. They have only five blocks.

When you´ve configured the independent copy, a full sync takes place. Each block is copied from the master volume to the shadow volume, and the bitmap volume is intialized. At the end of this process master and shadow are identical and the bitmap is in the "clean" state. No differences between the both.
There is one important fact: After the initial copy, the bitmap doesn´t have to be clean. During the full sync the independent copy behaves like a dependent copy. This is done to enable you to use the master volume directly after initiating the independent copy. So, when you change the master volume during the full sync, you will have a "dirty" bitmap (i will explain this condition in a few moments).

Okay, now we change the fourth block of the disk. As the old data is already on the shadow volume, we don´t have to move any data. But we log in the bitmap volume, that a block has changed, the block is "dirty". From now the bitmap is in the "dirty" state. The dirty state tells us, that there are differences between the master and the shadow volume.

Okay, we don´t need to move data, why do we need the bitmap volume. The bitmap volume makes the synchronisation of master and shadow much more efficent. With the bitmap volume you known the position of changed blocks. So when you resync your shadow with the shadow you just have to copy this blocks, and not the whole disk. After copying the block, the adjacent bit in the bitmap is set to zero, the system known that the synced block on master and shadow are identical again.

Advantages and DisadvantagesThis has some interesting advantages: You can export this disks in your SAN and use it from a different server. For example you can mount it on a backup server, thus you don´t need to transport all the traffic across you local area network.

But there are some disadvantages. You have a high initial amout of data to sync and the size of the master and the shadow have to be equal. This isn´t much a problem of the time needed for the sync (because of the fact, that it behaves as a dependent copy) but it poses more load to CPU and I/O system to copy all the data.

Posted by Joerg Moellenkamp in English, Oracle at 20:51