

Online resizing of Oracle volumes on Pure FlashArray



The screenshot shows a window titled "Edit Volume" with a close button (X) in the top right corner. Below the title bar, there are two input fields. The first is labeled "Name:" and contains the text "ora-test-vol1". The second is labeled "Provisioned Size:" and contains the number "251". To the right of the "Provisioned Size" field is a unit selector dropdown menu currently set to "GB". At the bottom right of the dialog, there are two buttons: a grey "Cancel" button and a blue "Save" button.

While thin provisioning enables higher capacity volumes without the guilt of actually wasting the space, we still see storage admins creating volumes close to what the DBAs have asked for and hence there is always a need for expanding the volume when the current volume reaches its capacity.

When it comes to ASM, lot of DBAs prefer to add a new volume than expanding the existing volume which would address the need but if there is a simple way to expand the ASM volume size without any downtime and not add any overhead with volume management, why not take advantage of it?

This blog post is to showcase the simplicity in resizing the FlashArray volume in ASM environment on Linux platform. I am using Red Hat Linux 7.0 version in my environment. Also I am using UDEV rules to manage the device and not ASMLib which gives the flexibility of not partitioning the device.

Following are the steps for resizing [Oracle](#) volumes on FlashArray.

1. Run the multipath command as root on the Linux host to get the current size of the lun. Note the friendly devicename is mpathb and the current size on the host is 225GB.
[crayon-6424c10ce1e7f744238972/]
2. Extend the LUN by editing the volume in the Pure UI and increase the size.



3. Rescan the SCSI devices to with the following command which will identify luns that were resized. (Need sg3_utils package for this to work, use yum install sg3_utils)
[crayon-6424c10ce1e86106918920/]
4. Resize the multipath device using the following command. Replace the <devicename> with the friendly name of your volume. (mpathb in this example)
[crayon-6424c10ce1e89860176656/]
[crayon-6424c10ce1e8a208346985/]
5. Run the multipath command again to validate the volume with the extended size. (You can see the increased size from 225G to 251G in our example)
[crayon-6424c10ce1e8b436489247/]
6. If you are using ASM, resize the volume at the ASM level by resizing the corresponding diskgroup using the following command which can be done while the diskgroup is mounted.
[crayon-6424c10ce1e8c774357351/]
[crayon-6424c10ce1e8d319809528/]

Try out [FlashBlade](#)

More on FlashArray

If you are using ASMLib, you have two options to extend the space.

1. Create a second partition on the extended space which means you have to create a new disk on the second partition and add it to the diskgroup. This is no different than adding a new volume but certainly has the following advantages.
 1. Less intrusive. No need to attach the volume to the host as it is already part of the volume. Still have to perform rescan-scsi-bus.sh with "s" option.
 2. Does not count against volume count if there is any limit to the volumes on the storage array.
2. Extend the initial partition by deleting the partition and recreating the partition using fdisk (d,n,p with defaults, w) followed by altering the diskgroup with new size which certainly works in theory but is not a preferred or recommended approach whether for production or non-production environment. If you wanted to try this, do it at your risk but at least take a snapshot of the volume before making any change to protect yourself.

If you are using cooked filesystem which again has partitions, you can certainly resize and the steps are

similar to the above #2 where you have to delete the partition and create new partition with the same starting block and run command like `resize2fs` or `xfs_growfs`. This can be done online as well but the same disclaimer and warnings apply. You better know what you are doing and if not you can mess up the data.