

Part 2

Manage logical volumes (LVM)

partitioning LVM fdisk stratis
 file system
 file system check
 mounting
 monitoring

resize lvm

storage pools lvm

physical disk
 physical volume
 volume group
 logical volume

create lvm

```
# pvcreate /dev/sdb /dev/sdc
# vgcreate vg0 /dev/sdb /dev/sdc
# lvcreate --name lv0 --size 60G vg0
```

Volume Group: vg0
 Logical volume: lv0
 Physical volume: /dev/sdb
 Physical volume: /dev/sdc
 Physical disk: /dev/sdb 30GB
 Physical disk: /dev/sdc 30GB
 Disk 1
 Disk 2
 Disk 3

pvdisplay → show physical volume
vgdisplay → show volume group
lvdisplay → show logical volume
...
mkfs.ext4 /dev/vg0/lv0 → format the disk
mkdir /data → make directory for mount
mount /dev/vg0/lv0 /data/ → mount disk to directory "data" (run time)
vim /etc/fstab → to mount permanently
/dev/vg0/lv0 /data ext4 defaults 0 0
mount -a → go and mount from file /etc/fstab
df -h → Show all mounted points.

extend /data 130GB

lv0 60

vg0 100 add /dev/sdd 50 GB

→ extend lv0 60GB to 130GB

```
# pvcreate /dev/sdd      → make disk physical volume  

# vgextend vg0 /dev/sdd      → vg0 130GB  

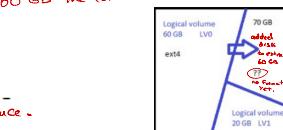
# lvextend vg0/lv0 --size 130G      → extend from 130GB to 130GB  

# resize2fs /dev/vg0/lv0  

# df -h
```

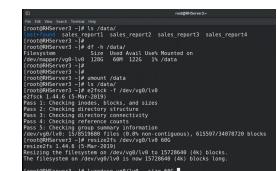
(cont'd) reduce (shrink size) if you reduce the disk size less than data size, data will be lost, so you have to scan firstly before you reduce.
 do be able to scan you have first to umount the disk.

```
# df -h
# umount /data
# e2fsck -f /dev/vg0/lv0      → do scan/check the disk "lv0"
# resize2fs /dev/vg0/lv0 60G      → resize lv0 to 60GB, but firstly you have to list the used space to make sure to not lose any data. using >df -h /mounted/folder/
# lvreduce vg0/lv0 --size 60G      → reduce the disk to 60GB
# mount /dev/vg0/lv0 /data/      → mount it
# df -h /data/
# ls /data/
```



remove logical volume

```
# vim /etc/fstab      remove mount line for logical volume, you have to do this step, because there will be a problem when system boots.
# umount /data
# lvremove /dev/vg0/lv0
# vgremove vg0
# pvremove /dev/sdb /dev/sdc /dev/sdd
# lsblk
```



Stratis (Storage Retrieval Automated Tracking Integrated System)

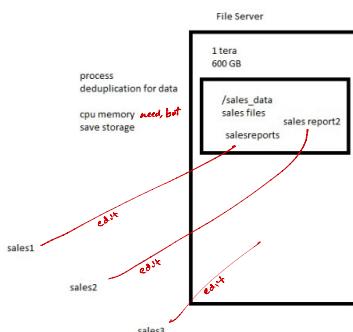
```
# yum install stratisd.x86_64 stratis-cli.noarch
# lsblk
# systemctl enable stratisd
# systemctl start stratisd
# systemctl status stratisd
```

```
# stratis pool create test-pool /dev/sdb /dev/sdc → pool in disk
# stratis blockdev list
# stratis pool add-data test-pool /dev/sdd → pool de disk
# stratis blockdev list
```

```
[root@RHSERVER3 ~]# stratis filesystem create test-pool test-filesystem1
[root@RHSERVER3 ~]# stratis filesystem create test-pool test-filesystem2
[root@RHSERVER3 ~]# stratis filesystem list
[root@RHSERVER3 ~]# mkdir /m1
[root@RHSERVER3 ~]# mkdir /m2
[root@RHSERVER3 ~]# mount /stratis/test-pool/test-filesystem1 /m1
[root@RHSERVER3 ~]# mount /stratis/test-pool/test-filesystem2 /m2
```

```
[root@RHSERVER3 ~]# stratis filesystem
[root@RHSERVER3 ~]# stratis pool
[root@RHSERVER3 ~]# vim /etc/fstab
/stratis/test-pool/test-filesystem1 /m1 xfs defaults 0 0
```

```
# umount /m1
# umount /m2
# stratis filesystem destroy test-pool test-filesystem1
# stratis filesystem destroy test-pool test-filesystem2
# stratis pool destroy test-pool
```



VDO (virtual data optimizer)

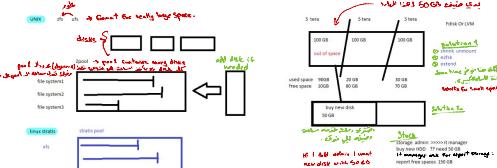
```
# systemctl status vdo
# vdo create --name=vdo1 --device=/dev/sdb --vdoLogicalSize=50G --verbose
# vdo create --name=vdo2 --device=/dev/sdc --vdoLogicalSize=50G --verbose

# pvcreate /dev/mapper/vdo1 /dev/mapper/vdo2
# vgcreate vg0 /dev/mapper/vdo1 /dev/mapper/vdo2
# lvcreate --name lv0 --size 60G vg0 ; mkfs.xfs /dev/vg0/lv0
# mkdir /mount100
# mount /dev/vg0/lv0 /mount100
# vi /etc/fstab
/dev/vg0/lv0 /mount100 xfs defaults,x-systemd.require=vdo.service 0 0

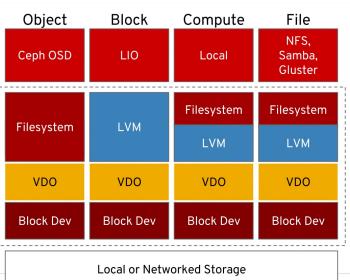
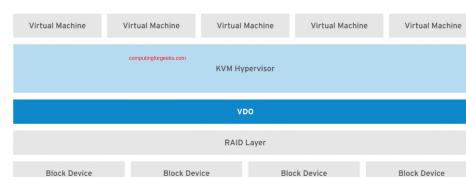
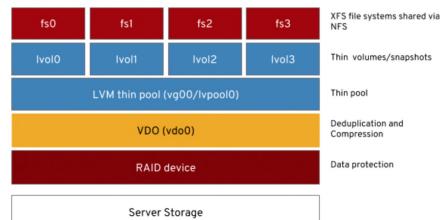
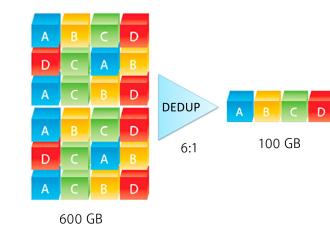
# vdostats --human-readable

# umount /mount100
# lvremove vg0/lv0
# vgremove vg0
# pvremove /dev/mapper/vdo1 /dev/mapper/vdo2
# vdo remove --name vdo1
# vdo remove --name vdo2
# vim /etc/fstab
# lsblk
```

remove mount lines



to check if what you wrote is right
→ umount /data
→ mount -a



W
Vg
Vv

evolve

why
resize
then
reduce \Rightarrow

