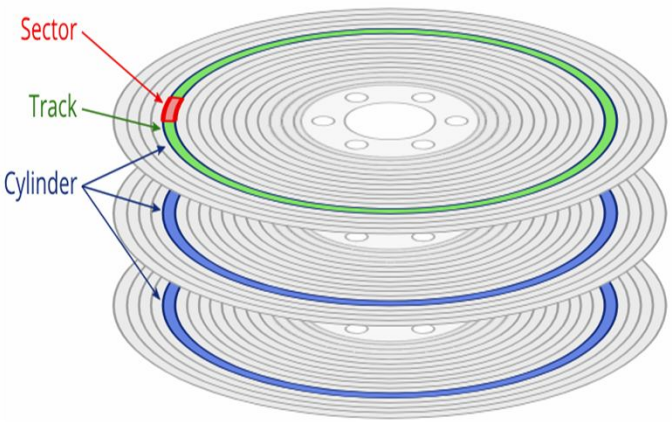
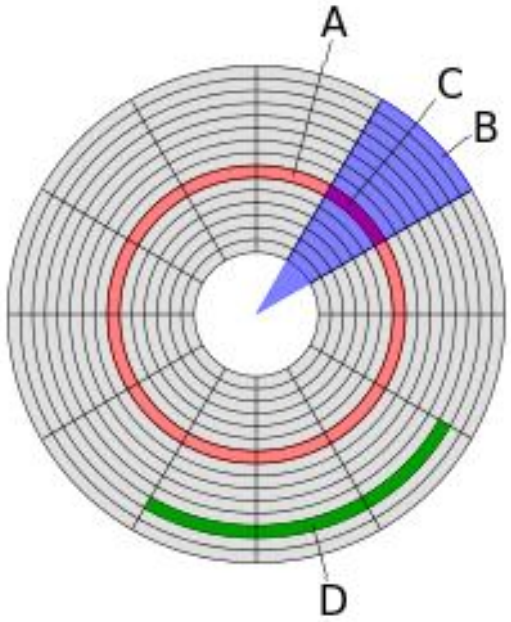
하드 디스크

하드 디스크 구조





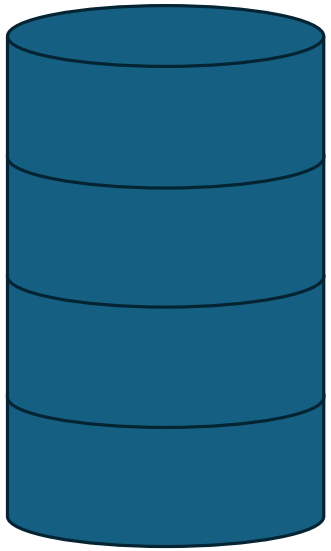
트랙 (A)	섹터 단위의 모음. 원심 전체가 트랙이 됨
섹터 (B)	하드디스크의 물리적인 최소 단위 (512byte)
트랙 섹터 (C)	같은 구역에 있는 섹터의 집합
클러스터 (D)	섹터 단위를 묶어 놓은 데이터의 입출력 단위 기본 4096byte

Disk Partition

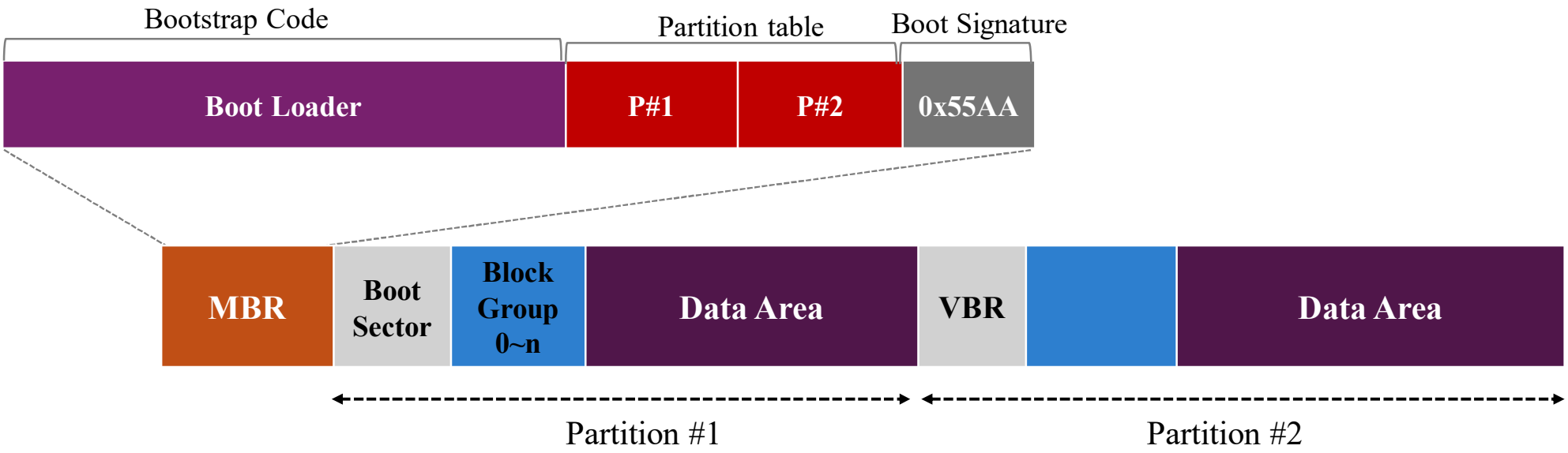
MBR	Primary Partition 1	Primary Partition 2	Primary Partition 3	Primary Partition 4

MBR	Primary Partition 1	Primary Partition 2	Primary Partition 3	Extended Partition (Logical Partition)

- 파티션
  - Primary 파티션 4개 또는 Primary 파티션 3개와 1개의 확장 파티션
  - 확장 파티션은 하나의 물리디스크에 하나만 설정할 수 있으며 논리 파티션은 제한되지 않음

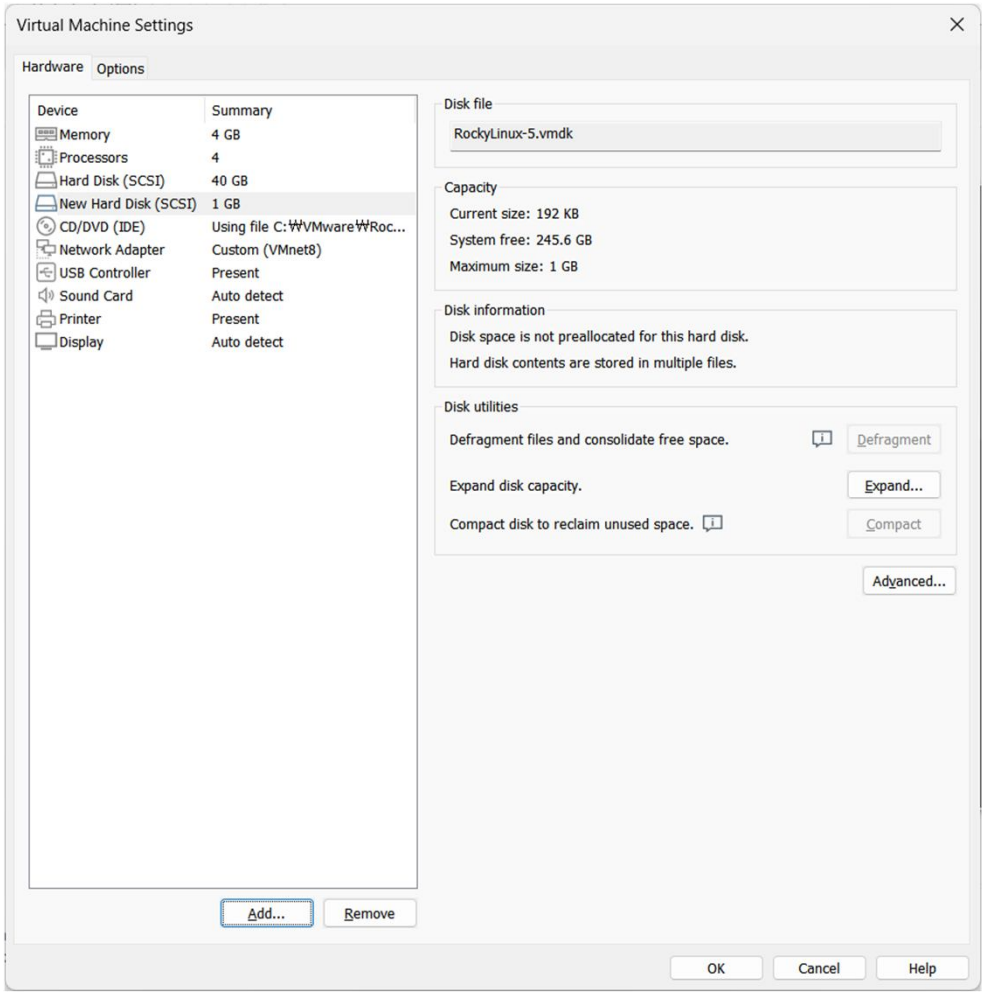


MBR(Master Boot Record) 구성



하드 디스크 추가

1단계. 하드 디스크 추가(SCSI)



- 재부팅 없이 추가된 하드 디스크 인식시키는 방법

#echo "- - -" > /sys/class/scsi_host/host0/scan

#ls /dev/sd*

#fdisk -l

2단계. 파티션 나누기

#ls /dev/sd*

#fdisk -l

#fdisk /dev/sdb

command : n

select : p

Patition number(1~4) : 1

First sector :

Last sector : +100M

Command : t

Hex code : 83

Command : p

Command : w

#ls /dev/sd*

#fdisk -l

```
Device     Boot    Start     End Sectors  Size Id Type
/dev/sdb1           2048  206847  204800  100M 83 Linux
/dev/sdb2         206848  821247  614400  300M 83 Linux
/dev/sdb3         821248 1230847  409600  200M 83 Linux
/dev/sdb4        1230848 1230948     101 50.5K 83 Linux

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
[root@localhost ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): n
To create more partitions, first replace a primary with an extended partition.
```

/dev/sdb 파티션 나누기
(sdb1(100MB), sdb2(300M),sdb3(200M),sdb4(100M))

- 확장 파티션 만들기 (4번 파티션 삭제 후 4번 파티션을 확장 파티션으로 지정)

Command : d

Patition number(1~4) : 4

Command : n

Select : e

First sector :

Last sector :

Command : p

Command : w

#fdisk -l

```
Command (m for help): d
Partition number (1-4, default 4): 4

Partition 4 has been deleted.

Command (m for help): n
Partition type
   p   primary (3 primary, 0 extended, 1 free)
   e   extended (container for logical partitions)
Select (default e): e

Selected partition 4
First sector (1230848-2097151, default 1230848):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (1230848-2097151, default 2097151):

Created a new partition 4 of type 'Extended' and of size 423 MiB.
```

```
Command (m for help): p
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x983edd17

Device     Boot   Start     End Sectors  Size Id Type
/dev/sdb1          2048  206847  204800  100M 83 Linux
/dev/sdb2        206848  821247  614400  300M 83 Linux
/dev/sdb3        821248 1230847  409600  200M 83 Linux
/dev/sdb4       1230848 2097151  866304  423M  5 Extended
```

***주파티션은 5개 이상 만들 수 없음**

• 확장 파티션 나누기

#fdisk /dev/sdb

Command : n

First sector :

Last sector : + 100M

Command : p

Command : w

```
[root@localhost ~]# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


Command (m for help): n
All primary partitions are in use.
Adding logical partition 5
First sector (1232896-2097151, default 1232896):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (1232896-2097151, default 2097151): +100M

Created a new partition 5 of type 'Linux' and of size 100 MiB.
```

```
Command (m for help): p
Disk /dev/sdb: 1 GiB, 1073741824 bytes, 2097152 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x983edd17

Device     Boot   Start     End Sectors  Size Id Type
/dev/sdb1          2048  206847  204800  100M 83 Linux
/dev/sdb2        206848  821247  614400  300M 83 Linux
/dev/sdb3        821248 1230847  409600  200M 83 Linux
/dev/sdb4       1230848 2097151  866304  423M  5 Extended
/dev/sdb5       1232896 1437695  204800  100M 83 Linux
```

#df -Th

```
[root@localhost ~]# df -Th
Filesystem          Type      Size  Used Avail Use% Mounted on
devtmpfs            devtmpfs  4.0M     0  4.0M   0% /dev
tmpfs               tmpfs     1.9G     0  1.9G   0% /dev/shm
tmpfs               tmpfs     777M  9.7M  768M   2% /run
/dev/mapper/rl-root xfs        36G  5.8G   30G  17% /
/dev/sda1           xfs       960M  273M  688M  29% /boot
tmpfs               tmpfs     389M  104K  389M   1% /run/user/1000
/dev/sr0            iso9660    11G   11G     0 100% /run/media/gildong/Rocky-9-4-x86_64-dvd
```

3단계. 파일 시스템 생성

- 파티션에 파일 시스템 생성

  #mkfs.ext4 -v /dev/sdb1

```
[root@localhost /]# mkfs.ext4 -v /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
fs_types for mke2fs.conf resolution: 'ext4', 'small'
/dev/sdb1 contains a ext2 file system
        last mounted on Tue Dec  3 07:14:35 2024
Proceed anyway? (y,N) y
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
25584 inodes, 102400 blocks
5120 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=33685504
13 block groups
8192 blocks per group, 8192 fragments per group
1968 inodes per group
Filesystem UUID: e2630f76-3f3b-4694-9a5f-591e647d5f30
Superblock backups stored on blocks:
        8193, 24577, 40961, 57345, 73729
```

4단계. Mount 하기

- 파티션에 파일 시스템 생성

  #mkdir /sbdisk

  #cd /sbdisk

  touch aaa bbb ccc

  #cd ..

  #mount /dev/sdb1 /sbdisk

  #cd /sbdisk

  #ls

  #dh -h

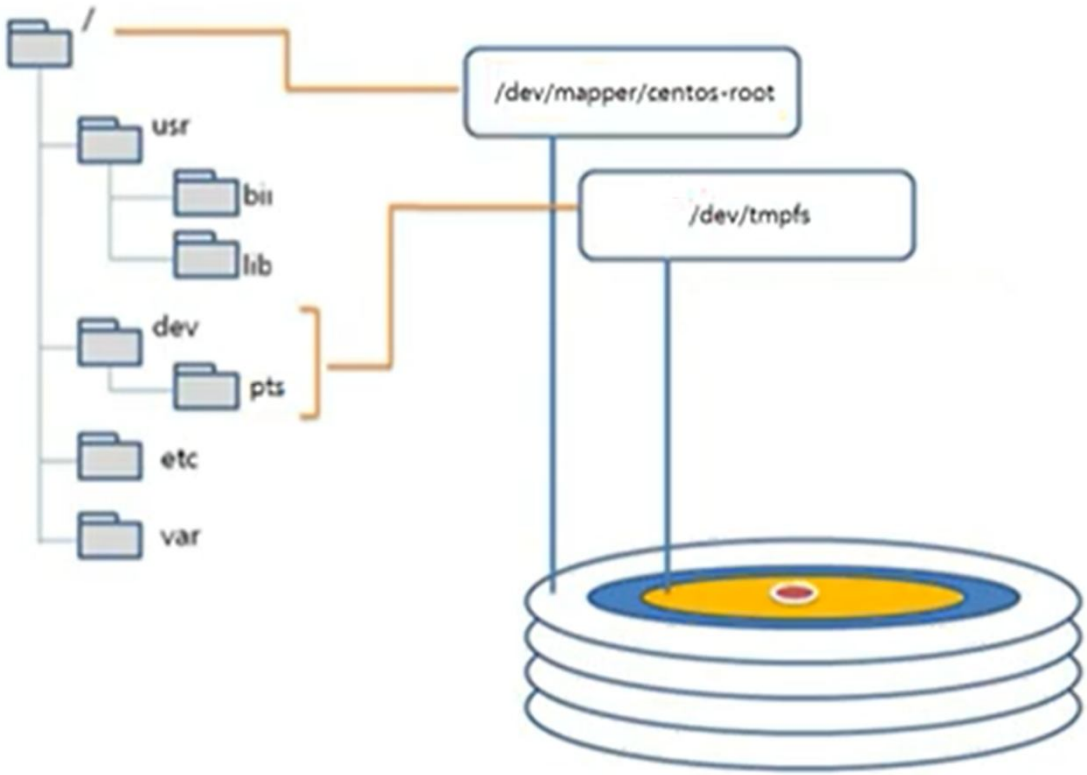
- Mount 해지

  #umount /dev/sdb1

Mount

특정 디렉터리와 특장 장치를 연결시켜 주는 작업

형식) mount [옵션] [장치] [마운트포인트]



#mkdir /DISK

#cd /DISK

#touch aaa bbb ccc

#mount /dev/sdb1 /DISK

#cd /DISK

#ls

```
[root@localhost ~]# mkdir /DISK
[root@localhost ~]# cd /DISK
[root@localhost DISK]# touch aaa
[root@localhost DISK]# touch bbb
[root@localhost DISK]# ls
aaa  bbb
[root@localhost ~]# mount /dev/sdb1 /DISK
[root@localhost ~]# cd /DISK
[root@localhost DISK]# ls
lost+found
[root@localhost DISK]#
```

5단계. 자동 마운트 설정

#nano /etc/fstab

/dev/sdb1 /sbdisk ext4 defaults 0 0

파일 /etc/fstab

- 자동 마운트 설정 파일

```
[root@localhost /]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Wed Nov 27 18:40:42 2024
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rl-root     /                       xfs     defaults        0 0
UUID=ca1050f0-a2c8-4acc-8b13-54b02df95289 /boot                   xfs     defaults        0 0
/dev/mapper/rl-swap     none                    swap    defaults        0 0
```

[파일시스템 장치] [마운트 포인트] [파일시스템 종류] [옵션] [덤프] [파일체크옵션]

Superblock 손상 시 발생하는 문제

#mkdir /TEST
#cd /TEST
#touch aaa
#dd if=/dev/zero of=/TEST/aaa bs=512 count=20
  512*20=10KB 파일을 생성해서 /TEST에 생성시킴

#umount /dev/sdb1
#dd if=/dev/zero of=/dev/sdb1 bs=512 count=32
  512*32=16,384(superblock 크기), Superblock에 0값으로 채움
#mount /dev/sdb1 /sbdisk

Superblock 복원

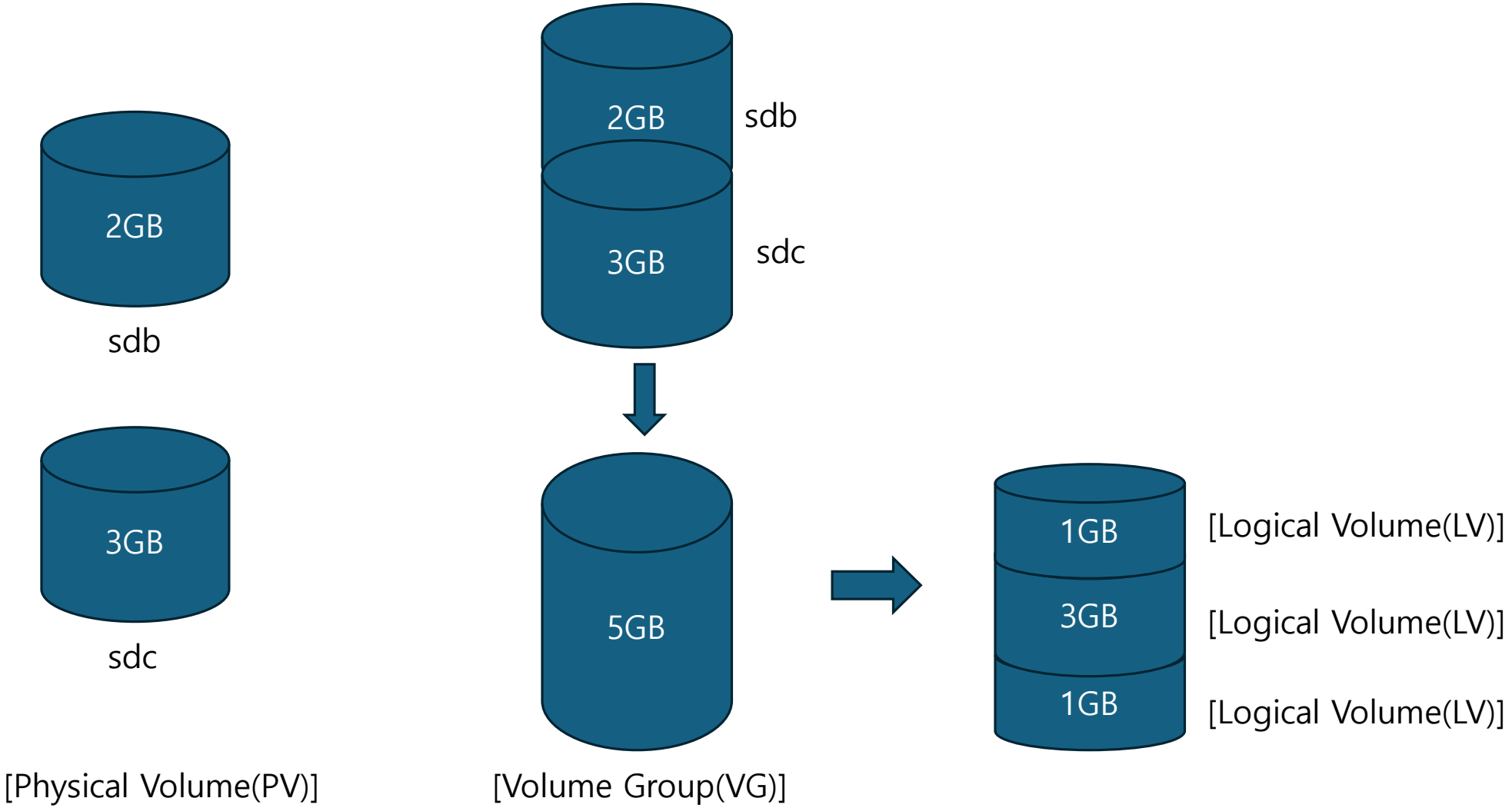
fsck -b [백업블록번호] [복구할 장치]

#fsck -b 8193 -fy /dev/sdb1
#mount /dev/sdb1 /sbdisk

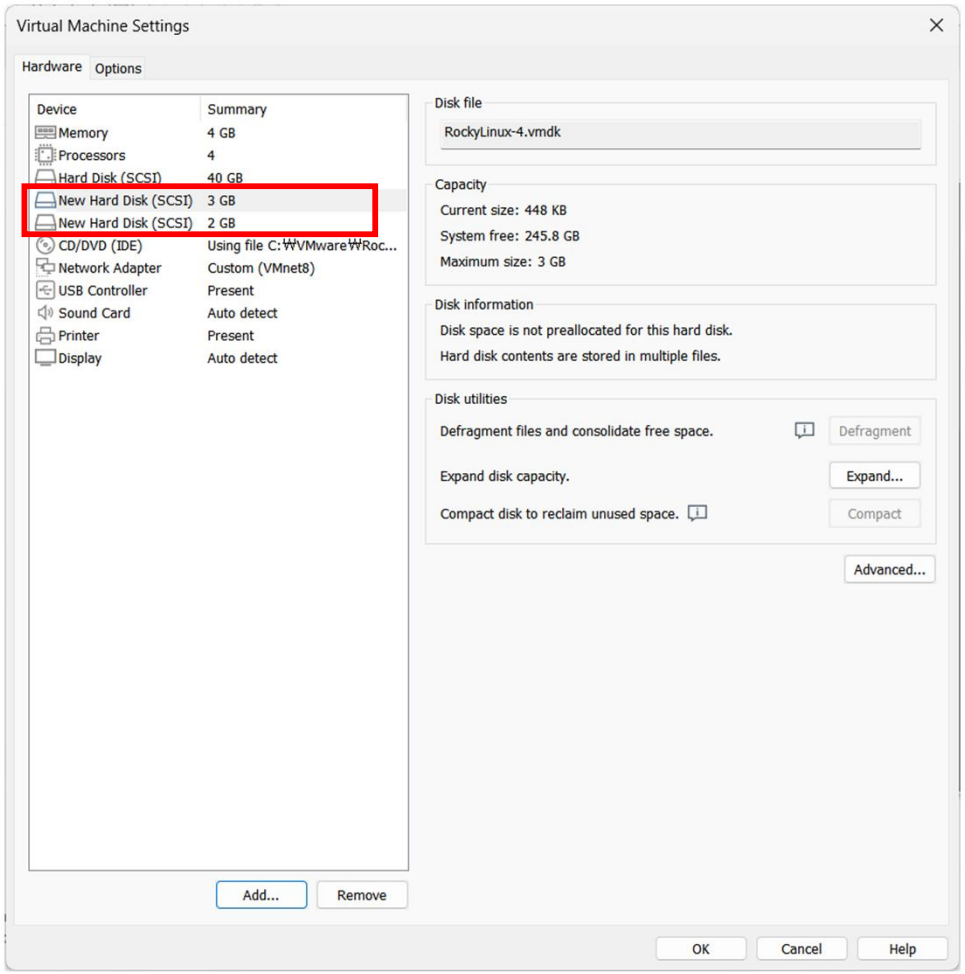
*dumpe2fs /dev/sdb1

```
[root@localhost /]# mkfs.ext4 -v /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
fs_types for mke2fs.conf resolution: 'ext4', 'small'
/dev/sdb1 contains a ext2 file system
        last mounted on Tue Dec  3 07:14:35 2024
Proceed anyway? (y,N) y
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
25584 inodes, 102400 blocks
5120 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=33685504
13 block groups
8192 blocks per group, 8192 fragments per group
1968 inodes per group
Filesystem UUID: e2630f76-3f3b-4694-9a5f-591e647d5f30
Superblock backups stored on blocks:
        8193, 24577, 40961, 57345, 73729
```

LVM(Logical Volume Manager)


2GB
sdb
3GB
sdc
2GB
sdb
3GB
sdc
5GB
1GB
[Logical Volume(LV)]
3GB
[Logical Volume(LV)]
1GB
[Logical Volume(LV)]
[Physical Volume(PV)]
[Volume Group(VG)]

① 하드 디스트 추가 (SCSI)



- 재부팅 없이 추가된 하드 디스크 인식시키는 방법

  #echo "- - -" > /sys/class/scsi_host/host0/scan

  #ls /dev/sd*

  #fdisk -l

② 하드 디스크 파티션 나누기 (파일 시스템 LVM 지정)

```
[root@localhost ~]# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xd34f3a5e.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-4194303, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-4194303, default 4194303):

Created a new partition 1 of type 'Linux' and of size 2 GiB.

Command (m for help): t
Selected partition 1
Hex code or alias (type L to list all): 8e
Changed type of partition 'Linux' to 'Linux LVM'.
```

- /dev/sdb와 /dev/sdc 파티션 나누기

  #fdisk /dev/sdb

  command : n

  select : p

  Patition number(1~4) : 1

  First sector :

  Last sector :

  Command : t

  Hex code : 8e

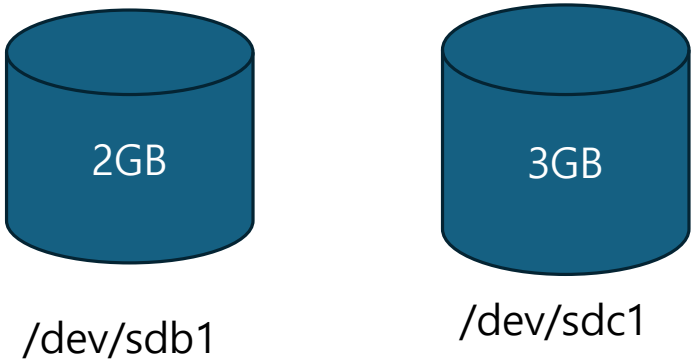
  Command : p

  Command : w

  #ls /dev/sd*

  #fdisk -l

③ Physical Volume 만들기

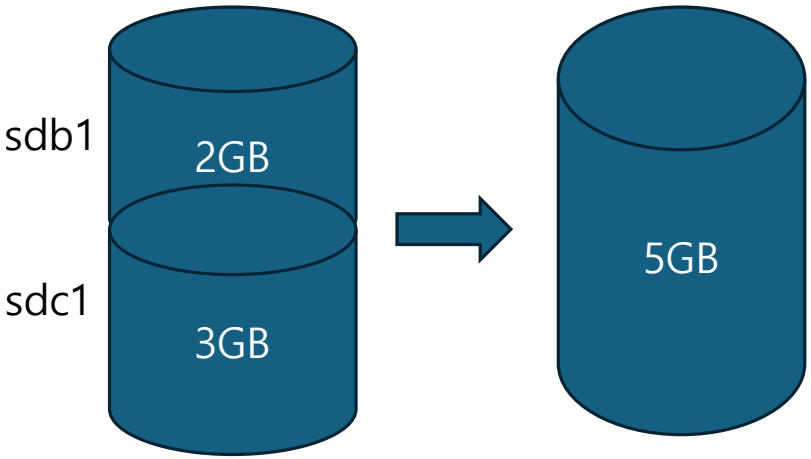


#pvcreate /dev/sdb1

#pvcreate /dev/sdc1

```
[root@localhost ~]# pvcreate /dev/sdb1
  WARNING: adding device /dev/sdb1 with idname /dev/sdb1 which is already used for missing device.
  Physical volume "/dev/sdb1" successfully created.
[root@localhost ~]#
[root@localhost ~]# pvcreate /dev/sdc1
  WARNING: adding device /dev/sdc1 with idname /dev/sdc1 which is already used for missing device.
  Physical volume "/dev/sdc1" successfully created.
```

④ Volume Group 만들기



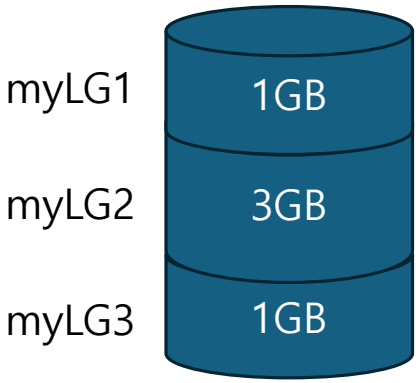
#vgcreate myVG /dev/sdb1 /dev/sdc1

```
[root@localhost ~]# vgcreate myVG /dev/sdb1 /dev/sdc1
  Volume group "myVG" successfully created
```

#vgdisplay

```
[root@localhost ~]# vgdisplay
  --- Volume group ---
  VG Name               myVG
  System ID
  Format                lvm2
  Metadata Areas        2
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                2
  Act PV                2
  VG Size               4.99 GiB
  PE Size               4.00 MiB
  Total PE              1278
  Alloc PE / Size       0 / 0
  Free  PE / Size       1278 / 4.99 GiB
  VG UUID               Tvb8uC-0uTR-Y6uS-7JIj-SnQR-ezNX-sadBAr
```

⑤ Logical Volume 만들기



#lvcreate --size 1G --name mLG1 myVG

#lvcreate --size 3G --name mLG2 myVG

#lvcreate --extents 100%FREE --name mLG3 myVG

```
[root@localhost ~]# lvcreate --size 1G --name myLG1 myVG
  Logical volume "myLG1" created.
[root@localhost ~]# lvcreate --size 3G --name myLG2 myVG
  Logical volume "myLG2" created.
[root@localhost ~]# lvcreate --extent 100%FREE --name myLG3 myVG
  Logical volume "myLG3" created.
```

#fdisk -l

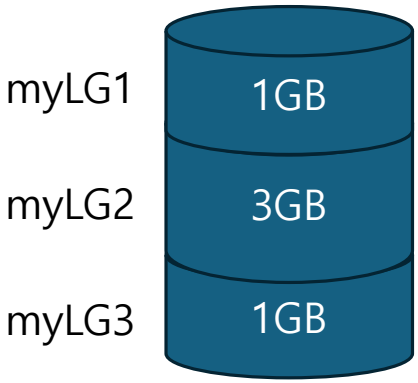
#ls /dev/my*

```
Disk /dev/mapper/myVG-myLG1: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/myVG-myLG2: 3 GiB, 3221225472 bytes, 6291456 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/myVG-myLG3: 1016 MiB, 1065353216 bytes, 2080768 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[root@localhost ~]# ls /dev/my*
myLG1  myLG2  myLG3
```

⑥ 파일 시스템 생성



#mkfs.ext4 /dev/myVG/myLG1

#mkfs.ext4 /dev/myVG/myLG2

#mkfs.ext4 /dev/myVG/myLG3

```
[root@localhost ~]# mkfs.ext4 /dev/myVG/myLG1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 2a11762e-0d9c-4aeb-8d29-a4da5bc27c8b
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```

⑦ mount

myLG1→ /lvm1

myLG2→ /lvm2

myLG3→ /lvm3

#mkdir /lvm1 /lvm2 /lvm3

#mount /dev/myVG/myLG1 /lvm1

#mount /dev/myVG/myLG2 /lvm2

#mount /dev/myVG/myLG3 /lvm3

```
[root@localhost ~]# mkdir /lvm1 /lvm2 /lvm3
[root@localhost ~]# mount /dev/myVG/myLG1  /lvm1
[root@localhost ~]# mount /dev/myVG/myLG2  /lvm2
[root@localhost ~]# mount /dev/myVG/myLG3  /lvm3
```

#df

```
[root@localhost ~]# df
Filesystem             1K-blocks     Used Available Use% Mounted on
devtmpfs                    4096        0      4096   0% /dev
tmpfs                    1988408        0   1988408   0% /dev/shm
tmpfs                     795364     9912    785452   2% /run
/dev/mapper/rl-root     36716544  6028860  30687684  17% /
/dev/sda1                 983040   279152    703888  29% /boot
tmpfs                     397680      104    397576   1% /run/user/1000
/dev/sr0                10660236 10660236         0 100% /run/media/gildong/Rocky-9-4-x86_64-dvd
/dev/mapper/myVG-myLG1    996780       24    927944   1% /lvm1
/dev/mapper/myVG-myLG2   3021608       24   2847916   1% /lvm2
/dev/mapper/myVG-myLG3   1005120       24    936696   1% /lvm3
```

⑦ /etc/fstab에 등록

- 부팅 시 자동으로 myLG1~3 장치가 /lvm1~3에 마운트 되도록 설정

#nano /etc/fstab

/dev/myVG/myLG1 /lvm1 ext4 defaults 0 0

/dev/myVG/myLG2 /lvm2 ext4 defaults 0 0

/dev/myVG/myLG3 /lvm3 ext4 defaults 0 0

```
[root@localhost ~]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Wed Nov 27 18:40:42 2024
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rl-root     /                       xfs     defaults        0 0
UUID=ca1050f0-a2c8-4acc-8b13-54b02df95289 /boot xfs     defaults        0 0
/dev/mapper/rl-swap     none                    swap    defaults        0 0
/dev/myVG/myLG1         /lvm1                   ext4    defaults        0 0
/dev/myVG/myLG2         /lvm2                   ext4    defaults        0 0
/dev/myVG/myLG3         /lvm3                   ext4    defaults        0 0
```