

copyright 김성우

LINUX

**파일 시스템과
하드디스크**

파일 시스템 (File System)

- 파일 시스템: HDD에 파일들을 어디에 어떻게 저장하고 어떻게 관리할 것이가 하는 자료 시스템 구조 (Windows: NTFS, Android: YAFFS, Linux: EXT)
- 포맷(Format): 운영체제를 설치하거나 운영체제에서 다른 파티션에 있는 데이터를 읽어 들이고자 할 때 파티션이 파일시스템을 갖도록 해 주는 사전 작업. 리눅스에서는 파일 시스템 생성이라고 함.

운영체제	지원하는 파일 시스템
윈도 XP까지	VFAT
윈도 XP 이후	NTFS
리눅스	EXT 파일 시스템(EXT2, EXT3, EXT4)
리눅스	저널링 파일 시스템(EXT3, ReiserFS, XFS)
리눅스	가상 파일 시스템(proc, sys)
리눅스	네트워크 파일 시스템(NFS, CIFS)
Andriod	YAFFS
Mac	HFS+, APFS

파일시스템 용량 체크 명령어

df

- 하드 디스크의 용량이 얼마나 남아 있는지 확인하는 명령어

옵션/명령어	의미	사용 예
-a	접근불가 등 모든 파일 시스템 정보를 출력	df -a
-i	파일 크기를 block대신 inode로 출력	df -i
-k	파일 크기를 kb로 출력	df -k
-T	파일 시스템의 종류와 함께 디스크 정보를 출력	df -T
-h	쉬운 용량 표시 단위로 표시해서 출력 (1G, 250M, 1K 등)	df -h

du

- 특정 디렉토리내의 파일 용량에 대한 정보를 확인하는 명령어

옵션/명령어	의미	사용 예
-a	해당 경로에 대한 사용한 디스크 용량 출력	-a
-k	용량을 킬로바이트 단위로 출력	-k
-m	용량을 메가바이트 단위로 출력	-m

마운트(Mount)

- 마운트: 물리적인 장치를 특정한 위치(디렉토리)에 연결시키는 과정
- 리눅스에서는 HDD의 파티션, CD/DVD, USB 메모리 등을 사용하기 위해서는 지정한 디렉토리에 연결하는 작업이 필요함

mount

#mount [옵션] [디바이스(장치)] [디렉터리]
#umount [옵션] [디바이스(장치)]
/etc/fstab : 마운트할 파일시스템과 옵션을 담고 있는 파일
/etc/mtab : 현재 마운트된 파일시스템과 옵션을 담고 있는 파일
/proc/mounts: 현재 마운트된 파일시스템에 대한 정보를 담고 있는 파일

옵션/명령어	의미	사용 예
-a	fstab에 정의되어있는 파일 시스템을 마운트 할 때 사용	mount -a /dev/sda2 testfolder
-r	읽기만 가능하게 마운트	mount -r /dev/sda2 testfolder
-w	쓰기 모드로 마운트	mount -w /dev/sda2 testfolder
-t [파일시스템]	파일 시스템의 형식을 지정 (생략시, /etc/fstab 파일을 참조) (auto 가능)	mount -t ext3 /dev/sda2 testfolder
-l	현재 마운트된 목록을 출력	mount -l /dev/sda2 testfolder
-o	추가적인 설정을 적용할 때 사용, 다수의 조건은 콤마(,)로 구분	mount -o loop /dev/sda2 testfolder
-f	실제로는 마운트 하지 않고 마운트가 가능한지 확인	mount -f /dev/sda2 testfolder
-n	/etc/mtab 파일에 쓰기 작업을 하지 않고 마운트한다. /etc가 읽기전용 파일시스템인 경우에 필요	mount -n /dev/sda2 testfolder

-o 옵션에서 사용할 수 있는 추가설정

옵션/명령어	의미
async	파일시스템에 대한 I/O가 비동기적으로 이뤄지도록 한다
auto	-a 옵션으로 마운트한다.
defaults	rw, suid, dev, exec, auto, nouser, async를 기본옵션으로 한다.
dev	파일시스템 상의 문자, 블록 특수 장치를 해석한다.
exec	바이너리의 실행은 허가한다.
noauto	-a 옵션으로는 마운트되지 않는다.
nodev	파일 시스템 상의 문자, 블록 장치에 대한 해석을 하지 않는다.
noexec	파일시스템에 실행권한을 주지 않는다.
nosuid	set-UID, set-GID를 무시하게 한다.
nouser	일반 사용자는 마운트를 하지 못하게 한다.
remount	이미 마운트된 파일시스템을 다시 마운트한다.
ro	파일 시스템을 읽기만 가능하게 한다.
rw	읽기/쓰기 모두 가능하게 마운트한다.
suid	set-UID, set-GID가 효력을 발휘할 수 있게 해준다.
vsync	파일 시스템에 대한 I/O가 동기적으로 이뤄지게 한다.
user	일반 사용자도 마운트할 수 있게 허용한다.
loop	CD-ROM 이미지 파일인 iso 파일을 마운트 할 때 사용

/etc/fstab

- 파일 시스템 마운트 테이블이 기록되어 있는 파일
- 시스템이 부팅될 때 이 파일에 기록된 마운트 정보대로 파일시스템이 자동으로 마운트

```
# <file system> <mount point> <type> <options> <dump> <pass>

/dev/sdb1 /userHome ext4 defaults,usrjquota=aquota.user,jqfmt=vfsv0 0 0
```

[장치명] [마운트할 디렉터리] [파일시스템] [옵션] [덤프] [부팅시 파일시스템 점검순서]

필드 항목	설명
장치명	마운트할 블록장치 또는 원격 파일시스템 장치명
마운트할 디렉터리	파일시스템을 마운트할 마운트 포인트, 즉 디렉터리
파일시스템	파일시스템의 유형
옵션	파일시스템에 관련된 마운트 옵션
덤프	dump 명령에 의해 덤프할 파일 시스템을 지정한다. 만일 0값이면 덤프가 필요가 없다고 판단함
부팅시파일시스템 점검순서	부팅 시 파일시스템의 이상 여부를 확인 후 이상이 있을 경우 자동으로 시스템 검사를 할 것인지를 지정 0은 점검하지 않으며, 1은 / (root파티션), 나머지는 2로 지정함 (병렬로 처리한다는 뜻)

ext4 파일시스템

- ext -> ext2 -> ext3 (파일 저널링 시스템) -> ext4 (현 리눅스 표준)
- ext2: 높은 안정성으로 오랫동안 리눅스의 표준으로 활약하였지만 데이터 손실 방지와 데이터 구조 일치성을 위해 반드시 fsck에 의해 점검을 받도록 되어있었음. 불필요한 시간이 많이 걸린다는 단점.
- ext3: 부팅 시 파일 시스템을 체크하지 않고 index에 대한 로그를 기록하여 복구할 수 있도록 만듬 (파일 저널링 시스템)
- ext4: ext3의 기능에 다른 유용한 기능들을 추가한 버전

1. 대용량 파일 지원

- 최대 1EB(엑사바이트)의 볼륨크기 지원
- 단일 파일 크기는 16TB를 지원

2. 기존 파일시스템 호환성

- 기존의 ext2, ext3을 호환하여 ext4시스템에서도 기존 파일 시스템 사용가능

3. 온라인 조각 모음

- ext3에서 발생하는 단편화를 ext4에서는 온라인 조각모음을 사용하여 성능 개선

4. 빨라진 fsck

- 데이터 무결성이 깨어졌을 때 fsck를 이용하여 파일 시스템을 체크.
- 노드 테이블과 노드 테이블의 체크섬을 사용하여 기존보다 2~20배 정도 속도 향상

5. extends 지원

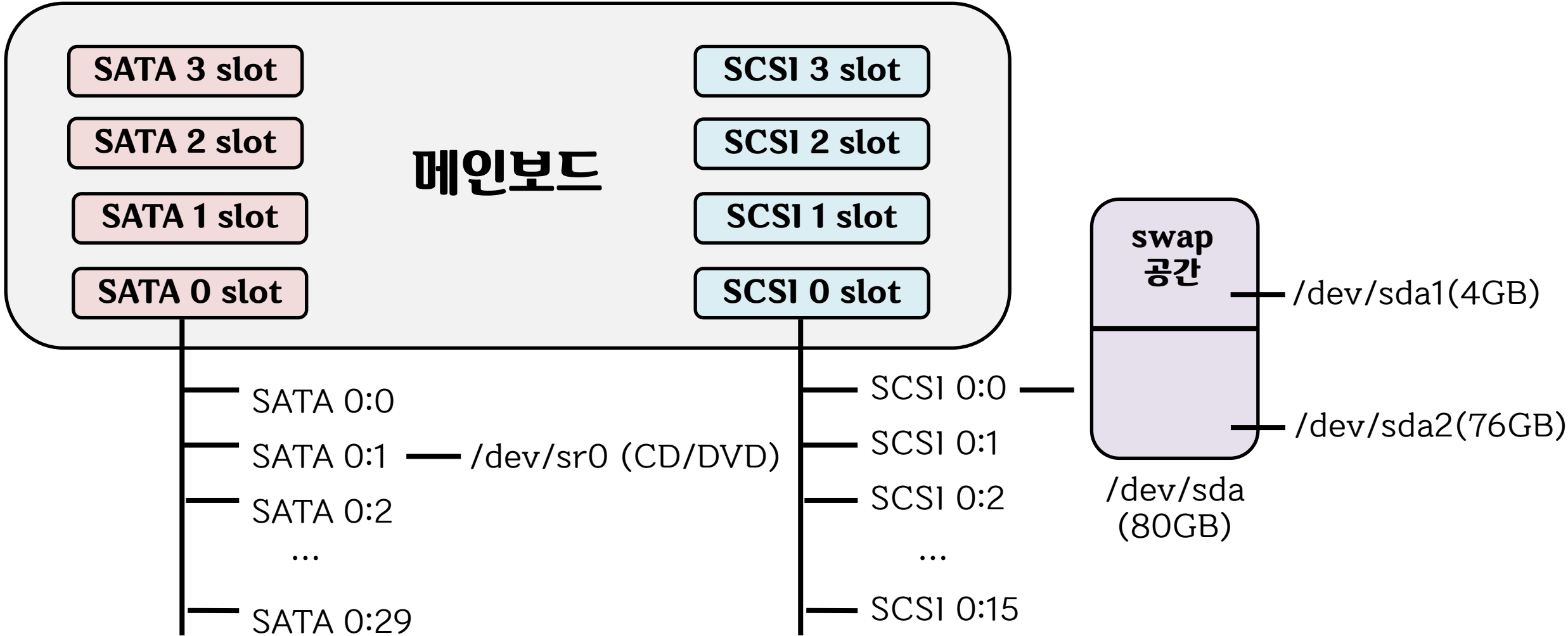
- 디스크 파일 기술자 효율을 증가시켜 큰 사이즈의 파일을 삭제할 때 시간을 단축시킴

6. 하위 디렉터리 수 향상

- 하위 디렉터리 수 제한이 기존 32,000개에서 2배 확대

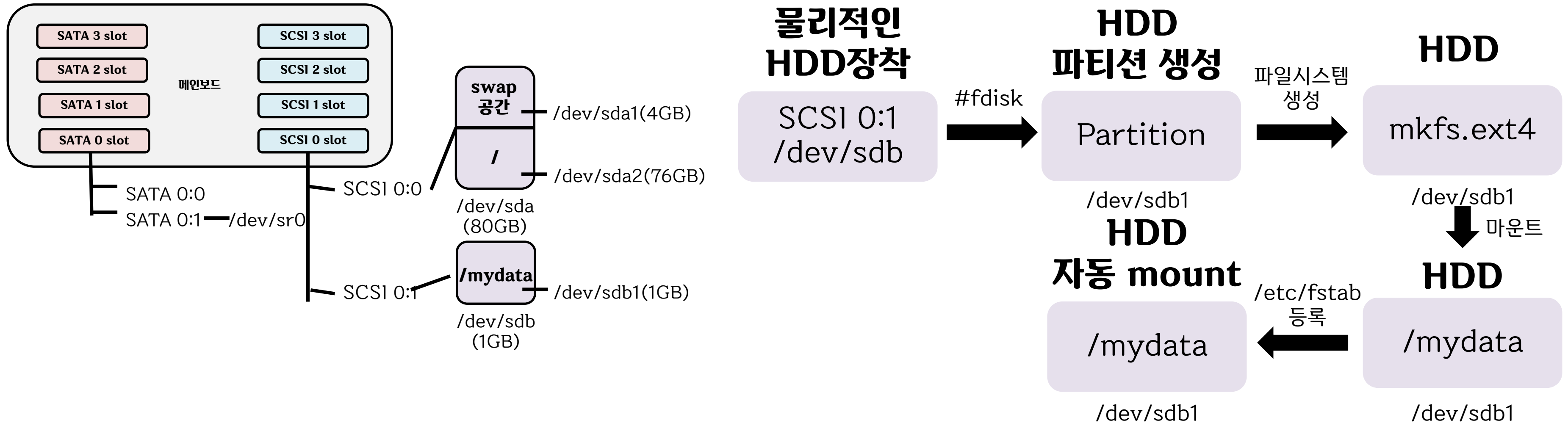
SATA 장치와 SCSI 장치의 구성

- VMware는 총 120개의 SATA HDD와 60개의 SCSI HDD 장착 가능
- **IDE(EIDE):** 데이터 전송을 위해 메인보드와 HDD를 연결하는 방식으로 현재는 사용하지 않음. 데이터 전송속도가 느림
- **SATA(Serial Advanced Technology Attachment):** 어댑터와 장치들을 비교적 속도가 빠른 직렬 연결을 이용해 연결하는 방식으로, SATA2가 가장 많이 사용됨
- **SCSI(Small Computer System Interface):** 주변장치의 연결과 분리가 쉽고, 데이터의 전송속도가 매우 빠르지만 가격이 높다는 단점이 있음
- **NVMe(Non Volatile Memory express):** SSD의 성능을 최대한 활용할 수 있도록 개발된 초고속 데이터 전송 규격으로, SATA 인터페이스 대비 최대 6배 이상의 속도를 낼 수 있어 초고속, 대용량 데이터 처리에 적합.



HDD 추가

- SWAP Partition: HDD의 일부를 RAM처럼 사용할 수 있게 만드는 기술로, RAM의 용량이 꽉 찼을 경우 중요도가 낮은 데이터들을 SWAP공간에 저장하여 사용함
- Linux에서는 파티션을 그냥 사용할 수 없으며 특정한 디렉터리에 마운트(Mount)시켜야 사용 가능



HDD 추가

1. 새로운 1GB의 SCSI HDD 추가

- Store virtual disk as a single file

2. 파티션 설정

- hdd를 사용하기 위해서는 먼저 파티션(Partition)을 설정해야함.
- 만약 hdd를 통째로 사용하려면 1개, 2개로 나눠서 사용하려면 2개의 파티션을 생성
=> Primary 파티션 / Extended 파티션으로 나뉨
=> 1개의 hdd에는 4개의 Primary파티션까지 설정 가능
ex) 5개의 파티션 생성 -> 3개의 Primary 파티션 + 1개의 Extended파티션으로 설정 후, Extended 파티션을 2개 이상의 Logical 파티션으로 설정
- 각 파티션은 섹터로 나누어지며, 한 섹터는 512byte로 설정되어 있음.
1GB를 가지는 파티션은 $1024\text{MB} = 512\text{byte} * 2097152\text{byte}$ 므로,
2097152개의 섹터로 나누어짐
시작 섹터가 2048인 이유는 0~2047(1MB) 부분은 시스템 성능 향상을 위해 사용하지 않기 때문. Blocks의 단위는 1KB

#fdisk /dev/sdb

Command: n (새로운 파티션 분할)

Select: p (Primary 파티션 선택)

Partition number: 1 (파티션 번호 1번 선택(4개까지 선택 가능))

First sector: Enter (시작 섹터 번호 입력)

Last sector: Enter (마지막 섹터 번호 입력)

Command: p (설정된 내용 확인)

Command: w (설정 저장)

```
root@Server: ~/바탕화면
root@Server:~/바탕화면# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).
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 0x30ea49b1.

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 (2048-2097151, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097151, default 2097151):

Created a new partition 1 of type 'Linux' and of size 1023 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: 0x30ea49b1

Device      Boot Start      End Sectors  Size Id Type
/dev/sdb1             2048 2097151 2095104 1023M 83 Linux

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

root@Server:~/바탕화면#
```

HDD 추가

3. 파일 시스템 형식 설정(ex4로.) (포맷)

- ext2, ext3, ext4, xfs 등의 파일 시스템이 있음. ex4, xfs는 다른 것들보다 향상된 파일 시스템이므로, swap을 제외하고는 전부 ext4 혹은 xfs를 사용함. xfs를 사용하기 위해서는 #apt install xfsprogs로, 관련 패키지를 설치해야 함.

```
#mkfs -t 파일시스템 파티션장치 or #mkfs.파일시스템 파티션장치  
#mkfs.ext4 /dev/sdb1
```

```
root@Server:~/바탕화면# mkfs.ext4 /dev/sdb1  
mke2fs 1.45.5 (07-Jan-2020)  
Creating filesystem with 261888 4k blocks and 65536 inodes  
Filesystem UUID: 3ed8c286-c093-45e6-8edd-8c030248af89  
Superblock backups stored on blocks:  
        32768, 98304, 163840, 229376  
  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (4096 blocks): done  
Writing superblocks and filesystem accounting information: done
```

4. 마운트(mount)

- lost + found 디렉터리는 파일 시스템을 점검할 때 생성되는 파일이 저장되는 곳.

```
#mkdir /mydata (=> 마운트할 디렉터리 생성)  
#cp /boot/vmlinuz-[tab] /mydata/test1 (=> vmlinuz로 시작하는 파일 복붙)  
#mount /dev/sdb1 /mydata (=> 포맷이 완료된 sdb1을 mydata에 mount)  
#cp /boot/vmlinuz-[tab] /mydata/test2 (=> vmlinuz로 시작하는 파일 복붙2)
```

```
root@Server:/dev# mkdir /mydata  
root@Server:/dev# cp /boot/vmlinuz-5.13.0-  
vmlinuz-5.13.0-30-generic vmlinuz-5.13.0-40-generic  
root@Server:/dev# cp /boot/vmlinuz-5.13.0-30-generic /mydata/test1  
root@Server:/dev# mount /dev/sdb1 /mydata  
root@Server:/dev# cp /boot/vmlinuz-5.13.0-30-generic /mydata/test2  
root@Server:/dev# ls -l /mydata/  
합계 9952  
drwx----- 2 root root    16384  6월  2 18:31 lost+found  
-rw-r--r--  1 root root 10171040  6월  2 18:41 test2
```

-> /mydata 디렉터리는 /dev/sda2가 아닌 /dev/sdb1에 있음. /mydata 디렉터리에 파일을 생성하면 /dev/sdb1에 파일을 생성하게 됨.
test1은 잠시 /dev/sda2에 숨어있음.

HDD 추가

5. 마운트 해제

```
#umount /dev/sdb1 (=> sdb1을 마운트 해제)
#ls -l /mydata (=> test1확인)
```

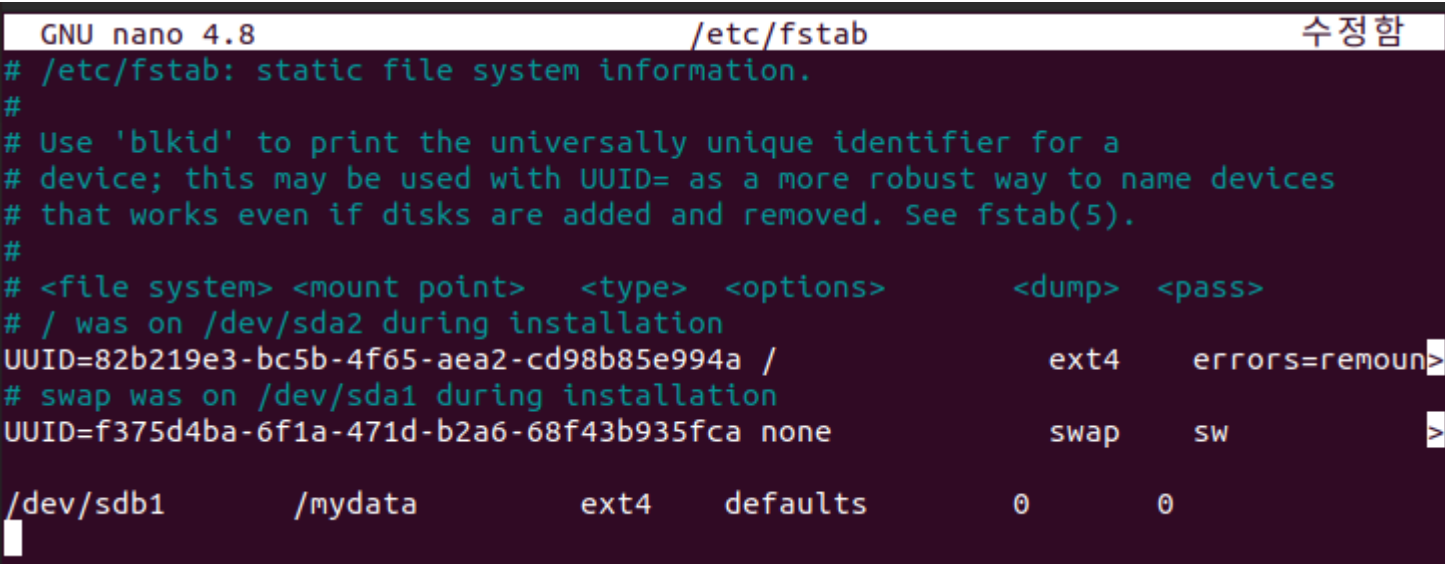
6. 컴퓨터 부팅 시 sdb1 장치가 mydata 디렉터리에 마운트되어 있도록 설정

- fstab파일은 리눅스가 부팅될 때마다 자동으로 읽는 중요한 파일으로 마운트 정보가 수록되어있음. (수정 시 주의)
 - 6개의 필드: '장치명', '마운트될 디렉터리', '파일 시스템', '속성', 'dump 사용 여부', '파일 시스템 체크 여부'를 의미
 - 파일 시스템과 속성
- default: 읽기/쓰기/실행 등 대부분 작업이 가능

- dump
- 1: 리눅스 dump 명령을 통해 백업 가능

- 파일 시스템 체크 여부
- 1, 2: 부팅 시 해당 파티션을 체크. 일반적으로 '/' 파일 시스템을 '1'로 설정하고 이외에는 '2'로 설정하거나 '0'으로 설정.
'0'으로 설정시에는 파일 시스템 체크를 생략하므로 부팅 속도가 향상됨

```
#nano /etc/fstab (=> 파일 open)
/dev/sdb1      /mydata ext4    defaults      0      0 (=>추가 및 저장, reboot로 재시작)
#ls -l /mydata로 test2확인
```



HDD 추가

6. 재부팅 후 확인

#reboot

```
root@Server:~/바탕화면# df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            960676         0    960676   0% /dev
tmpfs           198772      1396    197376   1% /run
/dev/sda2       78106292 7775384 66320260  11% /
tmpfs           993856         0    993856   0% /dev/shm
tmpfs           5120         0     5120   0% /run/lock
tmpfs           993856         0    993856   0% /sys/fs/cgroup
/dev/loop0        128        128         0 100% /snap/bare/5
/dev/loop1        66816     66816         0 100% /snap/gtk-common-themes/1519
/dev/loop2       254848    254848         0 100% /snap/gnome-3-38-2004/99
/dev/loop3        45824     45824         0 100% /snap/snapd/15904
/dev/loop4        63488     63488         0 100% /snap/core20/1494
/dev/loop5        63488     63488         0 100% /snap/core20/1434
/dev/loop6        55552     55552         0 100% /snap/snap-store/558
/dev/loop7        83328     83328         0 100% /snap/gtk-common-themes/1534
/dev/loop8        45824     45824         0 100% /snap/snapd/15534
/dev/sdb1       1014680     12500    933420   2% /mydata
tmpfs           198768         16    198752   1% /run/user/0
root@Server:~/바탕화면# ls -l /mydata
합계 9952
drwx----- 2 root root    16384 6월  2 18:31 lost+found
-rw-r--r--  1 root root 10171040 6월  2 18:41 test2
```

RAID (Redundant Array of Inexpensive/Independent Disks)

- 여러 개의 하드디스크를 하나의 하드디스크처럼 사용하는 방식.
- 하드웨어 RAID와 소프트웨어 RAID로 나뉨

- **하드웨어 RAID**

HW 제조업체에서 여러 개의 하드디스크를 연결한 장비를 만들어 그 자체를 공급하는 것. 각 제조업체에서 기술 지원을 받을 수 있음. 고가의 경우 SA-SCSI, 중저가는 SATA를 사용하여 생성됨. 각 제조업체마다 조작법이 다르고 고가라는 특징을 가짐.

- **소프트웨어 RAID**

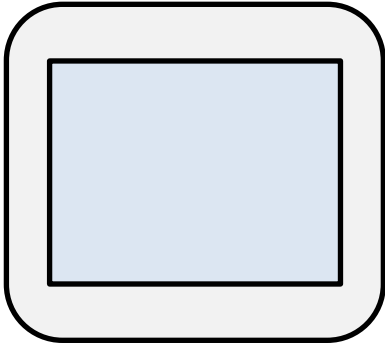
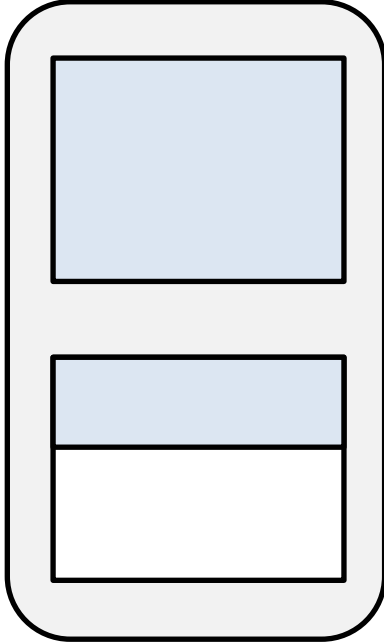
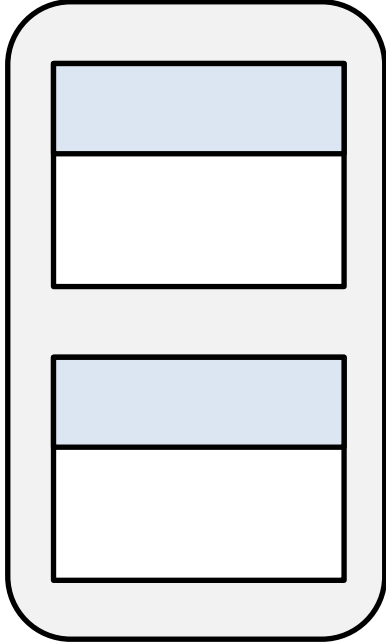
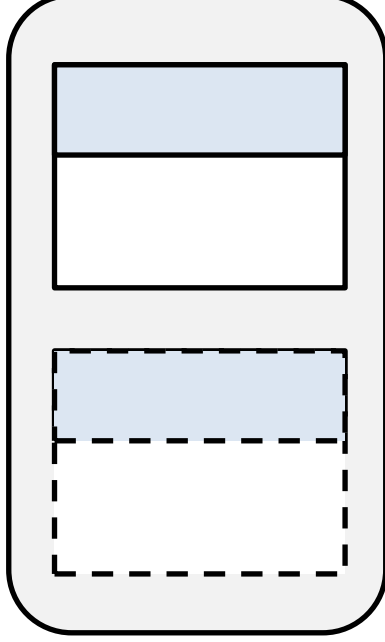
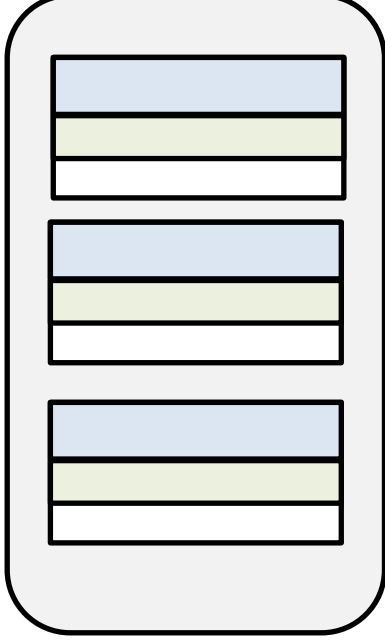
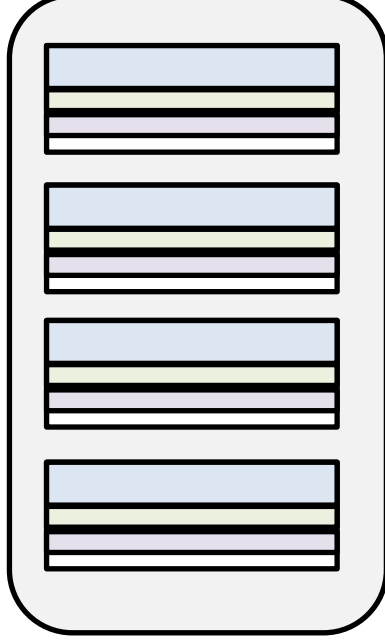
하드디스크만 여러 개 있으면 운영체제에서 지원하는 방식으로 RAID를 구성하는 것. HW RAID와 비교했을 때, 신뢰성과 속도가 낮음. 비용이 저렴하고 안전하게 데이터를 저장할 수 있다는 장점.

- **단순 볼륨**

HDD를 하나의 Volumn(묶음)으로 사용하는 방법. RAID방식에 포함되지 않음.

RAID LEVEL

- 구성하는 방식에 따라 Linear RAID, RAID 0, RAID 1, RAID 2, RAID 3, RAID 4, RAID 5의 7가지가 존재.
- 실무에서는 Linear RAID, RAID 0, RAID 1, RAID 5, RAID 6(RAID 5의 변형), RAID 1+0(RAID 0과 1을 혼합)

단순 볼륨	Linear RAID	RAID 0	RAID 1	RAID 5	RAID 6
					
사용량: 1(N개) 1개 HDD사용	사용량: 2(N개) 2개 이상 1번째부터 저장 디스크 추가 가능	사용량: 2(N개) 2개 이상 동시 저장 속도가 가장 빠름 Stripping	사용량: 1(N/2개) 2개 이상 동시 저장 결함 허용 공간 효율 좋음	사용량: 2(N-1개) 3개 이상 결함 허용 패리티 정보 사용 공간 효율 좋음	사용량: 2(N-2개) 4개 이상 RAID 5의 개선판 결함 허용 중복 패리티 정보 사용

Linear RAID / RAID 0 / RAID 1

Linear

첫 번째 HDD에 데이터가 완전히 저장된 후 다음 HDD에 데이터를 저장함.
첫 번째 HDD에 데이터가 완전히 저장되지 않는다면 다음 HDD는 사용되지 않음.
HDD의 용량이 달라도 전체 용량을 문제없이 사용가능.

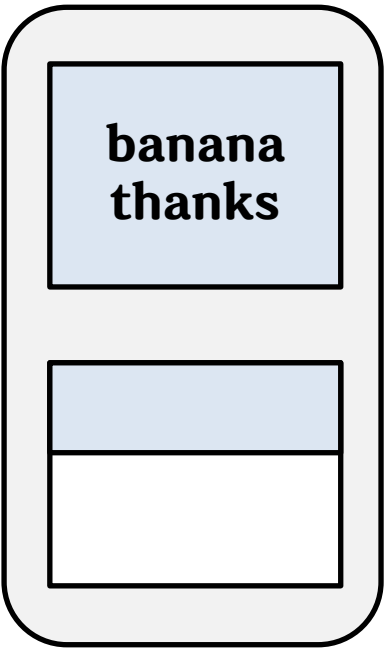
RAID 0

모든 HDD를 동시에 사용하는 '스트라이핑(Stripping)'방식을 사용.
(bit단위 혹은 block단위로 저장됨)
저장되는 시간 or 속도 면에서 모든 RAID 중 가장 성능이 뛰어나며,
모든 HDD를 사용하므로 공간 효율도 좋음.
단점은 하나의 HDD에서 문제가 발생하면 DATA를 알 수 없게 되어 모두 쓸모가 없어짐.
각 HDD의 크기가 다르면 작은 HDD기준으로 동작하여 큰 HDD의 공간이 남음.
'빠른 성능을 요구하되, 데이터가 전부 없어져도 문제가 되지 않는 자료'를 저장하는데 적합.

RAID 1

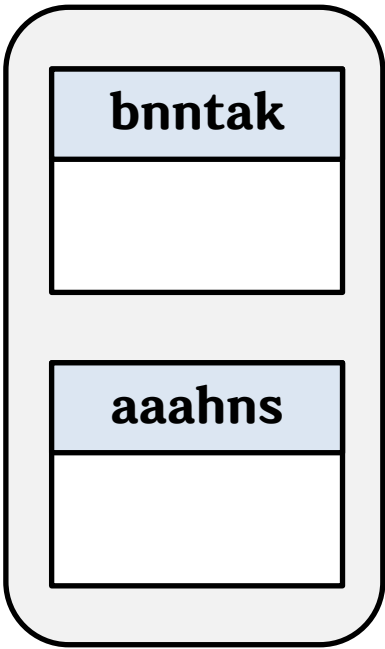
똑같은 데이터를 복제해놓는 '미러링(Mirroring)'방식을 사용.
하나의 data를 저장할 때 필요한 공간은 해당 data의 2배가 필요.
HDD 중 하나가 고장 나도 data가 손실되지 않는 '결함 허용(Fault-tolerance)' 특징을 가짐.
속도는 하나의 data를 저장하는데 걸리는 속도와 동일.
'HDD가 고장 나도 없어져서는 안될 중요한 데이터'에 적합.

Linear RAID



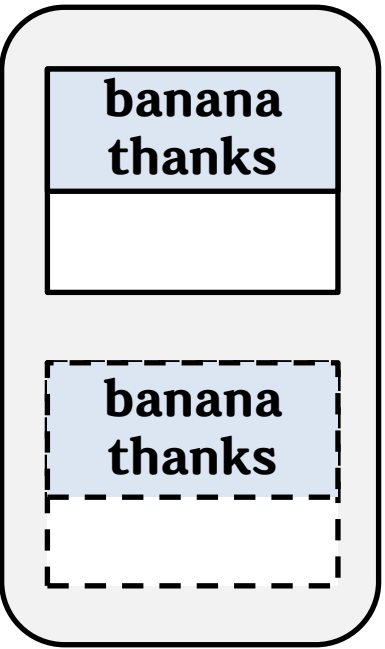
총 12byte 사용

RAID 0



총 12byte 사용

RAID 1



총 24byte 사용

RAID 5

RAID 5

RAID 0 + Linear 1을 생각하여 구성된 것.
최소 3개 이상의 HDD가 있어야 구성 가능. (대부분 5개 이상으로 구성)
HDD에 오류가 발생하면 '패리티(Parity)'를 이용해서 데이터 복구 가능.
어느정도 결함을 허용하며 저장 공간의 효율이 좋음.
data를 저장할 때, Parity를 위한 공간이 추가로 필요함.
HDD를 여러 개를 사용할 수록 공간 효율이 좋아짐. (HDD 1개 고장 ok)

000 111 010 011 12bit 저장 시

각 행이 짝수가 되도록 만들기 위해
숫자를 구성하는 '짝수 패리티' 사용

RAID 5

저장되는 방향 -->			
sda	sdb	sdc	sdd
0	0	0	parity
1	1	parity	1
0	parity	1	0
parity	0	1	1

RAID 5

저장되는 방향 -->			
sda	sdb	sdc	sdd
0	0	0	0
1	1	1	1
0	1	1	0
0	0	1	1

RAID 6 / 그 외 RAID

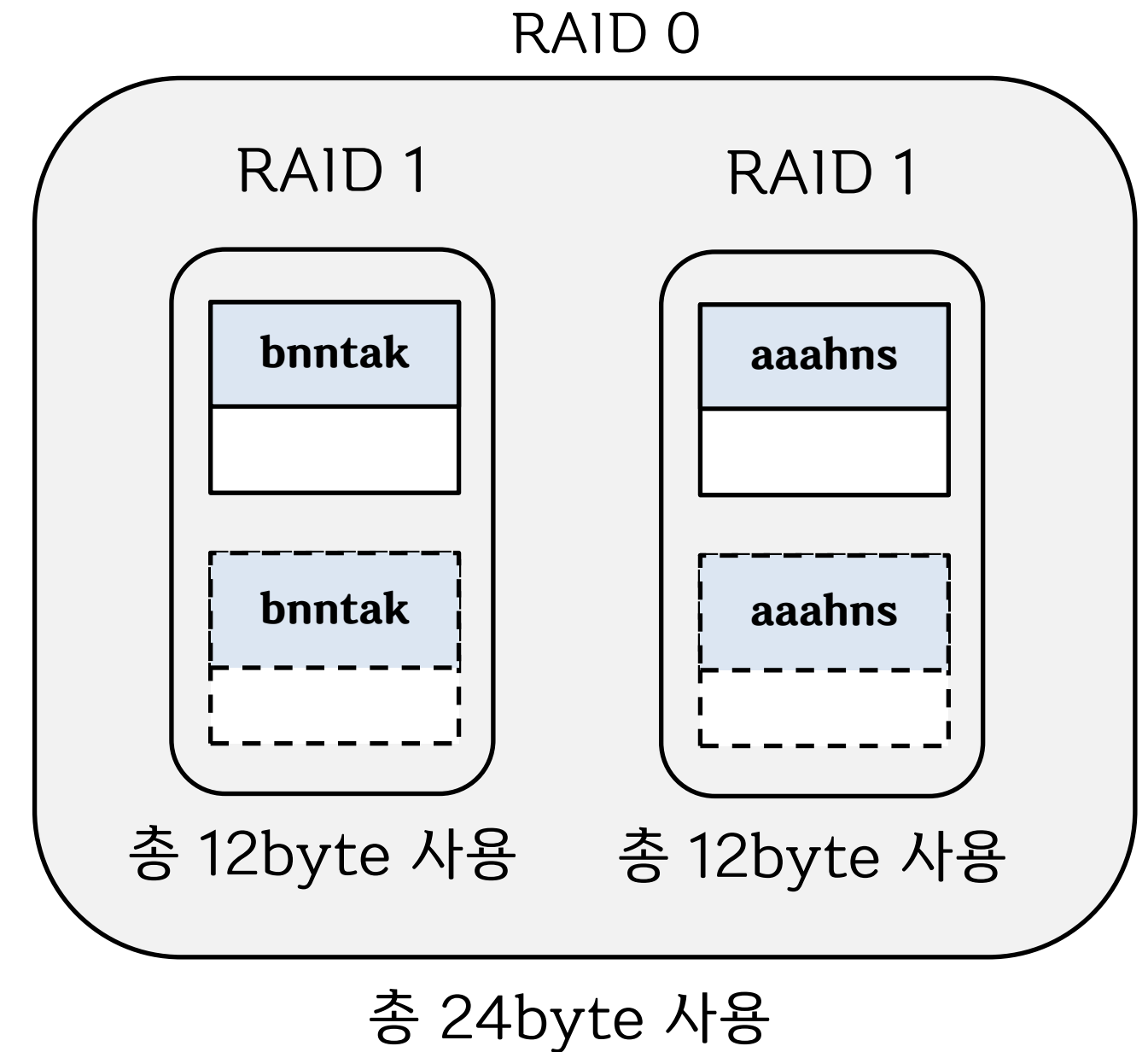
RAID 6

RAID 5방식을 개선한 것으로, 2개의 Parity를 사용. (HDD 2개 고장 ok)
따라서 최소 4개 이상의 HDD로 구성해야 함.

Parity를 생성하기 위한 알고리즘이 복잡하기 때문에 RAID 5보다 성능(속도) 효율이 약간 낮음.

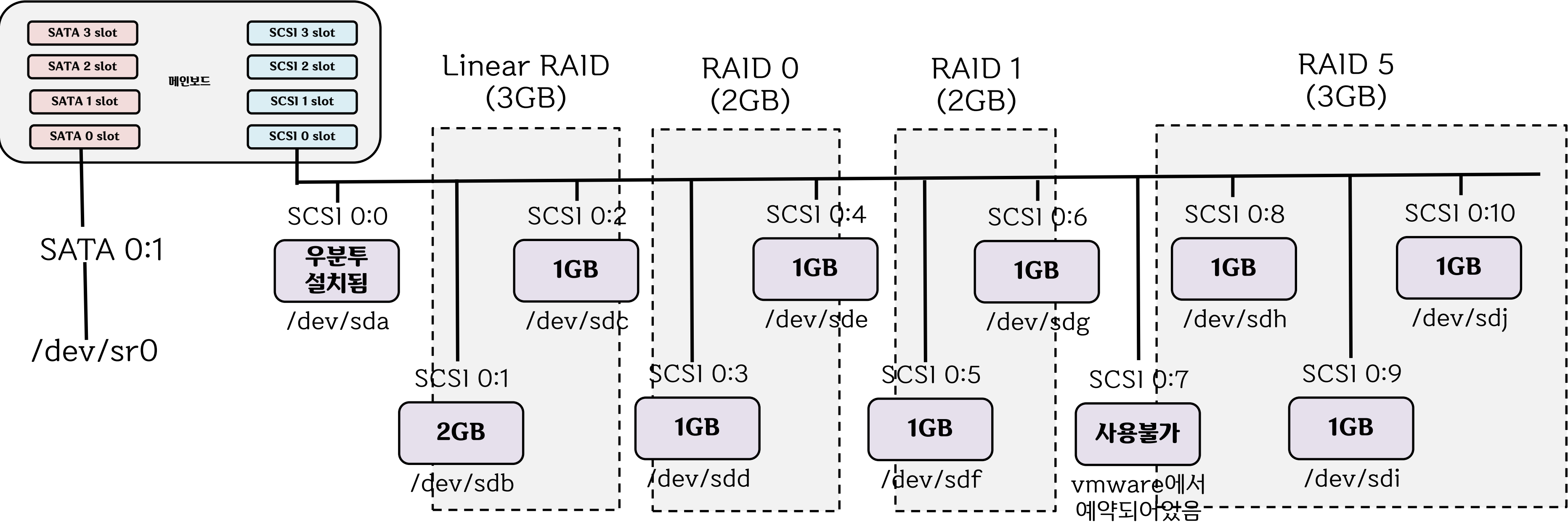
그 외(RAID 1+0)

RAID 1로 구성한 data를 다시 RAID 0으로 구성하는 방법. 안정성과 속도를 동시 확보 가능.



RAID 구현

- 9개의 SCSI HDD 장착 후 Linear RAID, RAID 0, RAID 1, RAID 5 구현

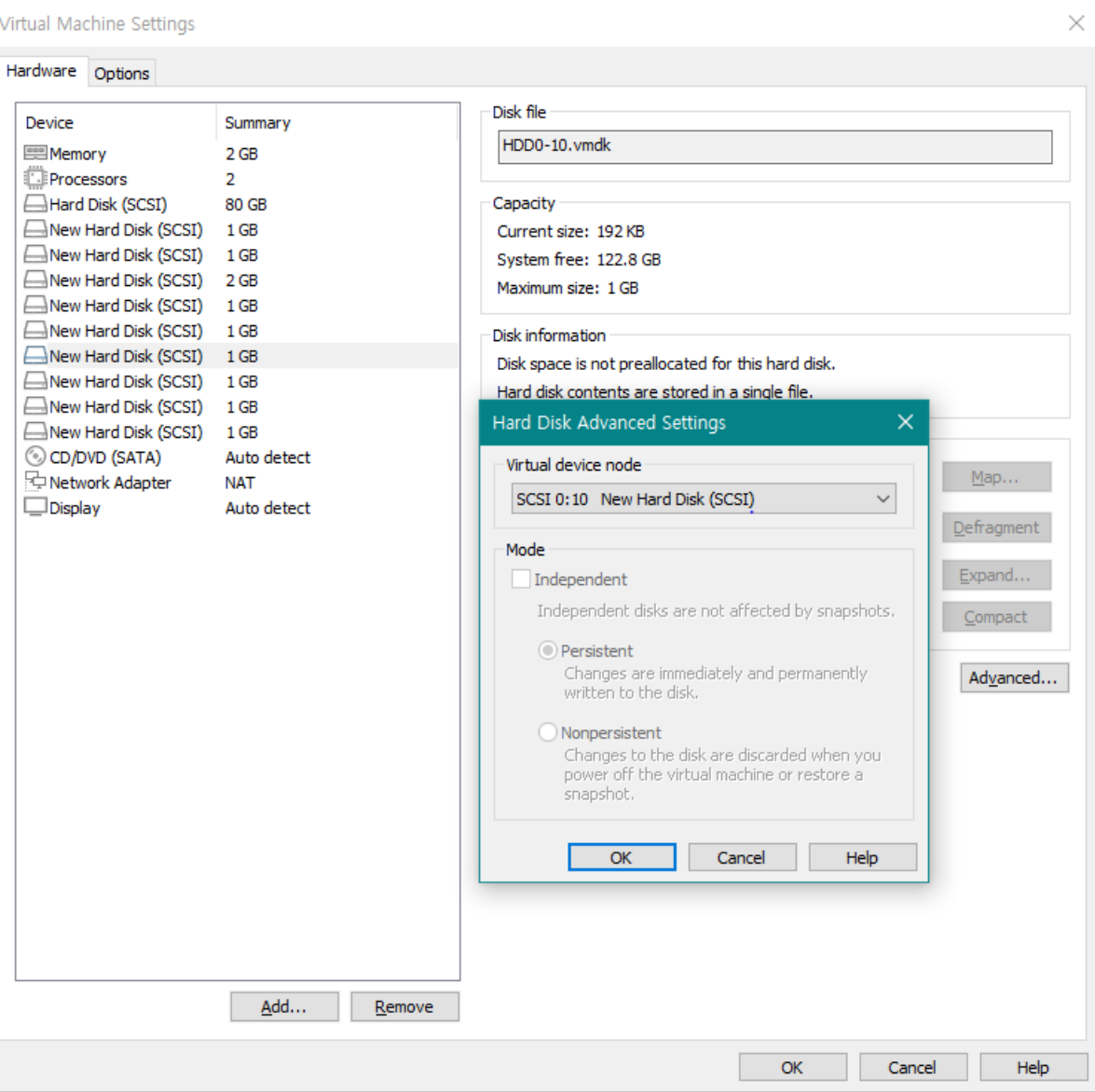


하드디스크 관리

RAID 구현

RAID 구현

1-0.HDD 생성(SERVER)



1-1. HDD 생성 확인

```
root@Server:~/바탕화면# ls -l /dev/sd*
brw-rw---- 1 root disk 8,  0 6월  2 21:49 /dev/sda
brw-rw---- 1 root disk 8,  1 6월  2 21:49 /dev/sda1
brw-rw---- 1 root disk 8,  2 6월  2 21:49 /dev/sda2
brw-rw---- 1 root disk 8, 16 6월  2 21:49 /dev/sdb
brw-rw---- 1 root disk 8, 32 6월  2 21:49 /dev/sdc
brw-rw---- 1 root disk 8, 48 6월  2 21:49 /dev/sdd
brw-rw---- 1 root disk 8, 64 6월  2 21:49 /dev/sde
brw-rw---- 1 root disk 8, 80 6월  2 21:49 /dev/sdf
brw-rw---- 1 root disk 8, 96 6월  2 21:49 /dev/sdg
brw-rw---- 1 root disk 8, 112 6월  2 21:49 /dev/sdh
brw-rw---- 1 root disk 8, 128 6월  2 21:49 /dev/sdi
brw-rw---- 1 root disk 8, 144 6월  2 21:49 /dev/sdj
```

1-2. sdb 파티션 생성

```
root@Server:~/바탕화면# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).
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 0x085f294b.

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 (type L to list all codes): fd
Changed type of partition 'Linux' to 'Linux raid autodetect'.

Command (m for help): p
Disk /dev/sdb: 2 GiB, 2147483648 bytes, 4194304 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: 0x085f294b

Device      Boot Start      End Sectors Size Id Type
/dev/sdb1             2048 4194303 4192256   2G fd Linux raid autodetect

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

root@Server:~/바탕화면# █
```


RAID 구현

1-3. sd 파티션 전부 생성 후 확인

```
root@Server:~/바탕화면# ls /dev/sd*
/dev/sda  /dev/sdb  /dev/sdc1  /dev/sde  /dev/sdf1  /dev/sdh  /dev/sdi1
/dev/sda1 /dev/sdb1 /dev/sdd  /dev/sde1 /dev/sdg  /dev/sdh1 /dev/sdj
/dev/sda2 /dev/sdc  /dev/sdd1 /dev/sdf  /dev/sdg1 /dev/sdi  /dev/sdj1
```

2-1. Linear RAID 생성1

```
root@Server:~/바탕화면# mdadm --create /dev/md9 --level=linear --raid-devices=2 /dev/sdb1 /dev/sdc1
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md9 started.
root@Server:~/바탕화면#
root@Server:~/바탕화면# mdadm --detail --scan
ARRAY /dev/md9 metadata=1.2 name=Server:9 UUID=f6187ebb:ce5dc668:6ebad7f6:7c189f56
root@Server:~/바탕화면#
```

2-2. Linear RAID 포맷 후 mount

```
root@Server:~/바탕화면# mkfs -t ext4 /dev/md9
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 784896 4k blocks and 196224 inodes
Filesystem UUID: 92dfa2b2-a414-42a7-a9f8-d9cd058c99c8
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

root@Server:~/바탕화면# mkdir /raidLinear
root@Server:~/바탕화면# mount /dev/md9 /raidLinear/
root@Server:~/바탕화면# df
Filesystem      1K-blocks      Used Available Use% Mounted on
udev             960676         0     960676  0% /dev
tmpfs           198772        1480     197292  1% /run
/dev/sda2       78106292  7615512   66480132  11% /
tmpfs           993856         0     993856  0% /dev/shm
tmpfs            5120         0         5120  0% /run/lock
tmpfs           993856         0     993856  0% /sys/fs/cgroup
/dev/loop0         128         128          0 100% /snap/bare/5
/dev/loop2        66816       66816          0 100% /snap/gtk-common-themes/1519
/dev/loop3        63488       63488          0 100% /snap/core20/1434
/dev/loop4       254848     254848          0 100% /snap/gnome-3-38-2004/99
/dev/loop5        55552       55552          0 100% /snap/snap-store/558
/dev/loop7        45824       45824          0 100% /snap/snapd/15534
tmpfs           198768         24     198744  1% /run/user/0
/dev/loop8        45824       45824          0 100% /snap/snapd/15904
/dev/loop6        63488       63488          0 100% /snap/core20/1494
/dev/loop1        83328       83328          0 100% /snap/gtk-common-themes/1534
/dev/md9         3024752       9216   2842176  1% /raidLinear
root@Server:~/바탕화면#
```

RAID 구현

2-3. Linear RAID 부팅 설정

GNU nano 4.8

/etc/fstab

수정함

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>        <dump>  <pass>
# / was on /dev/sda2 during installation
UUID=82b219e3-bc5b-4f65-aea2-cd98b85e994a /          ext4      errors=remount-ro 0      1
# swap was on /dev/sda1 during installation
UUID=f375d4ba-6f1a-471d-b2a6-68f43b935fca none       swap      sw        0      0

/dev/md9    /raidLinear /ext4     /defaults 0      0
```

2-4. 최종 Linear RAID 설정 확인

```
root@Server:~/바탕화면# nano /etc/fstab
root@Server:~/바탕화면# mdadm -D /dev/md9
/dev/md9:
        Version : 1.2
        Creation Time : Thu Jun  2 22:22:12 2022
        Raid Level : linear
        Array Size : 3139584 (2.99 GiB 3.21 GB)
        Raid Devices : 2
        Total Devices : 2
        Persistence : Superblock is persistent

        Update Time : Thu Jun  2 22:22:12 2022
        State : clean
        Active Devices : 2
        Working Devices : 2
        Failed Devices : 0
        Spare Devices : 0

        Rounding : 0K

Consistency Policy : none

        Name : Server:9 (local to host Server)
        UUID : f6187ebb:ce5dc668:6ebad7f6:7c189f56
        Events : 0

        Number Major Minor RaidDevice State
           0     8     17         0   active sync  /dev/sdb1
           1     8     33         1   active sync  /dev/sdc1
```


RAID 구현

3.1 RAID 0 부팅 설정 후 최종 확인

```
root@Server:~/바탕화면# nano /etc/fstab
root@Server:~/바탕화면# mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
    Creation Time : Thu Jun  2 23:07:11 2022
    Raid Level : raid0
    Array Size : 2091008 (2042.00 MiB 2141.19 MB)
    Raid Devices : 2
    Total Devices : 2
    Persistence : Superblock is persistent

    Update Time : Thu Jun  2 23:07:11 2022
    State : clean
    Active Devices : 2
    Working Devices : 2
    Failed Devices : 0
    Spare Devices : 0


    Layout : -unknown-
    Chunk Size : 512K

Consistency Policy : none

           Name : Server:0 (local to host Server)
           UUID : 0de65c0a:b62e9940:467e02a9:68a6e722
           Events : 0

   Number   Major   Minor   RaidDevice State
     0         8       49         0     active sync   /dev/sdd1
     1         8       65         1     active sync   /dev/sde1
root@Server:~/바탕화면#
```

3.1 RAID 0 생성 후 mount

```
root@Server:~/바탕화면# mdadm -C /dev/md0 --level=0 --raid-devices=2 /dev/sdd1 /dev/sde1
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
root@Server:~/바탕화면# mdadm --detail --scan
ARRAY /dev/md9 metadata=1.2 name=Server:9 UUID=f6187ebb:ce5dc668:6ebad7f6:7c189f56
ARRAY /dev/md0 metadata=1.2 name=Server:0 UUID=0de65c0a:b62e9940:467e02a9:68a6e722
root@Server:~/바탕화면# mkfs -t ext4 /dev/md0
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 522752 4k blocks and 130816 inodes
Filesystem UUID: df18340e-369a-4d78-8c8d-466535bd35ca
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

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

root@Server:~/바탕화면# mkdir /raid0
root@Server:~/바탕화면# mount /dev/md0 /raid0
root@Server:~/바탕화면# df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev              960676         0    960676   0% /dev
tmpfs            198772        1484    197288   1% /run
/dev/sda2       78106292 7615520  66480124  11% /
tmpfs            993856         0    993856   0% /dev/shm
tmpfs             5120         0         5120   0% /run/lock
tmpfs            993856         0    993856   0% /sys/fs/cgroup
/dev/loop0         128         128         0 100% /snap/bare/5
/dev/loop2        66816        66816         0 100% /snap/gtk-common-themes/1519
/dev/loop3        63488        63488         0 100% /snap/core20/1434
/dev/loop4       254848       254848         0 100% /snap/gnome-3-38-2004/99
/dev/loop5        55552       55552         0 100% /snap/snap-store/558
/dev/loop7        45824       45824         0 100% /snap/snapd/15534
tmpfs            198768         24    198744   1% /run/user/0
/dev/loop8        45824       45824         0 100% /snap/snapd/15904
/dev/loop6        63488       63488         0 100% /snap/core20/1494
/dev/loop1        83328       83328         0 100% /snap/gtk-common-themes/1534
/dev/md9         3024752       9216   2842176   1% /raidLinear
/dev/md0         2025360       6144   1898284   1% /raid0
root@Server:~/바탕화면#
```

RAID 구현

4. 최종 _etc_fstab

```
GNU nano 4.8 /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>        <dump> <pass>
# / was on /dev/sda2 during installation
UUID=82b219e3-bc5b-4f65-aea2-cd98b85e994a /                ext4      errors=remoun>
# swap was on /dev/sda1 during installation
UUID=f375d4ba-6f1a-471d-b2a6-68f43b935fca none             swap      sw
#
/dev/md9          /raidLinear      ext4      /defaults      0          0
/dev/md0          /raid0            ext4      /defaults      0          0
/dev/md1          /raid1            ext4      /defaults      0          0
/dev/md5          /raid5            ext4      /defaults      0          0
```

4. 최종 df

```
root@Server:~/바탕화면# df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            960676         0     960676   0% /dev
tmpfs           198772        1500     197272   1% /run
/dev/sda2       78106292    7615548    66480096  11% /
tmpfs           993856         0     993856   0% /dev/shm
tmpfs           5120          0        5120   0% /run/lock
tmpfs           993856         0     993856   0% /sys/fs/cgroup
/dev/loop0       128          128          0 100% /snap/bare/5
/dev/loop2       66816        66816          0 100% /snap/gtk-common-themes/1519
/dev/loop3       63488        63488          0 100% /snap/core20/1434
/dev/loop4       254848     254848          0 100% /snap/gnome-3-38-2004/99
/dev/loop5       55552        55552          0 100% /snap/snap-store/558
/dev/loop7       45824        45824          0 100% /snap/snapd/15534
tmpfs           198768         32     198736   1% /run/user/0
/dev/loop8       45824        45824          0 100% /snap/snapd/15904
/dev/loop6       63488        63488          0 100% /snap/core20/1494
/dev/loop1       83328        83328          0 100% /snap/gtk-common-themes/1534
/dev/md9         3024752        9216     2842176   1% /raidLinear
/dev/md0         2025360         6144    1898284   1% /raid0
/dev/md1         1013688         2564     942416   1% /raid1
/dev/md5         2025360         6144    1898284   1% /raid5
```


5. mdama 버그해결

```
root@Server:~/바탕화면# mdadm --detail --scan
ARRAY /dev/md/Server:0 metadata=1.2 name=Server:0 UUID=0de65c0a:b62e9940:467e02a9:68a6e722
ARRAY /dev/md/Server:9 metadata=1.2 name=Server:9 UUID=f6187ebb:ce5dc668:6ebad7f6:7c189f56
ARRAY /dev/md/Server:5 metadata=1.2 name=Server:5 UUID=2b9fa1b3:1d5f144d:6d82e873:3b9f00ef
ARRAY /dev/md/Server:1 metadata=1.2 name=Server:1 UUID=b4d7f995:266cc8c3:2e99ea4b:837e4c65
root@Server:~/바탕화면#
```

복사(C)

HTML로 복사(H)

붙여넣기(P)

읽기 전용(O)

기본 설정(P)

새 창(W)

새 탭(T)

메뉴 모음 보이기(M)

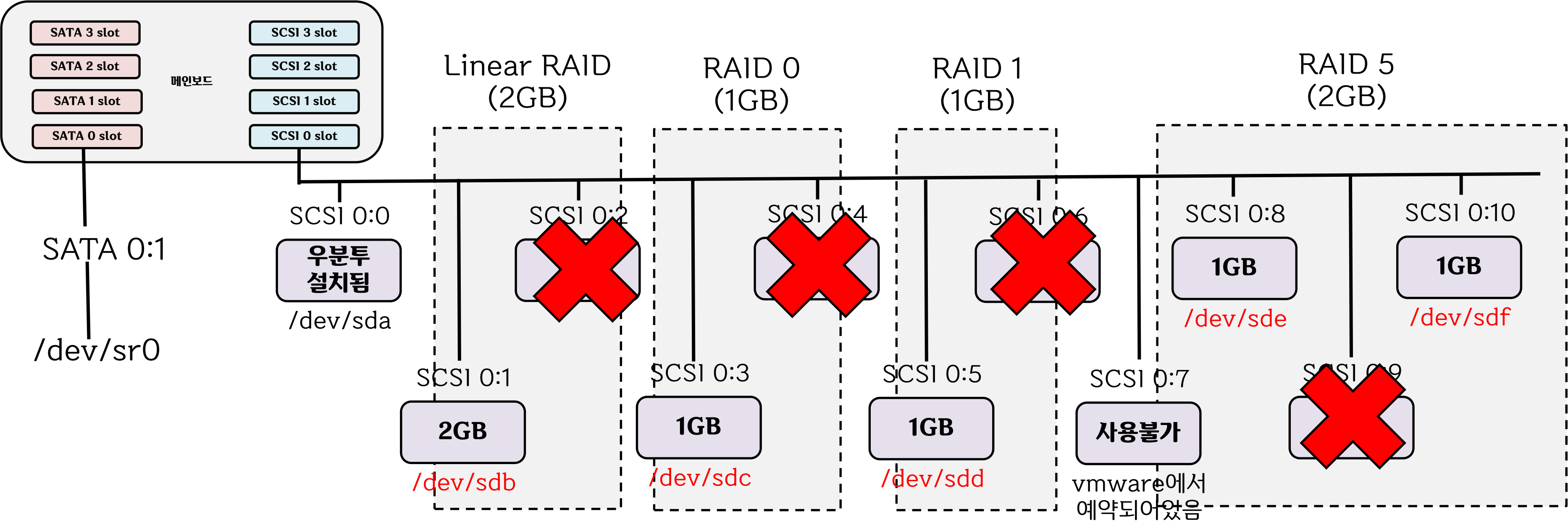
```
root@Server:~/바탕화면# gedit /etc/mdadm/mdadm.conf
24 ARRAY /dev/md/Server:0 metadata=1.2 UUID=0de65c0a:b62e9940:467e02a9:68a6e722
25 ARRAY /dev/md/Server:9 metadata=1.2 UUID=f6187ebb:ce5dc668:6ebad7f6:7c189f56
26 ARRAY /dev/md/Server:5 metadata=1.2 UUID=2b9fa1b3:1d5f144d:6d82e873:3b9f00ef
27 ARRAY /dev/md/Server:1 metadata=1.2 UUID=b4d7f995:266cc8c3:2e99ea4b:837e4c65
28
```

6. 최종 확인 후 스냅샷 저장

```
root@Server:~/바탕화면# ls -l /dev/md*
brw-rw---- 1 root disk 9, 0 6월 9 02:15 /dev/md0
brw-rw---- 1 root disk 9, 1 6월 9 02:15 /dev/md1
brw-rw---- 1 root disk 9, 5 6월 9 02:15 /dev/md5
brw-rw---- 1 root disk 9, 9 6월 9 02:15 /dev/md9
root@Server:~/바탕화면# df
Filesystem      1K-blocks      Used Available Use% Mounted on
udev              960304          0     960304   0% /dev
tmpfs             198772        1532     197240   1% /run
/dev/sda2        78106292    7897788    66197856  11% /
tmpfs             993856          0     993856   0% /dev/shm
tmpfs              5120           0        5120   0% /run/lock
tmpfs            993856          0     993856   0% /sys/fs/cgroup
/dev/loop1         63488        63488          0 100% /snap/core20/1494
/dev/loop2         63488        63488          0 100% /snap/core20/1518
/dev/loop0          128          128          0 100% /snap/bare/5
/dev/loop3         45824        45824          0 100% /snap/snapd/15534
/dev/loop4        260224        260224          0 100% /snap/gnome-3-38-2004/106
/dev/loop5         66816        66816          0 100% /snap/gtk-common-themes/1519
/dev/loop6         83328        83328          0 100% /snap/gtk-common-themes/1534
/dev/loop7         45824        45824          0 100% /snap/snapd/15904
/dev/loop8        254848        254848          0 100% /snap/gnome-3-38-2004/99
/dev/loop9         55552        55552          0 100% /snap/snap-store/558
/dev/md5           2025360        6144    1898284   1% /raid5
/dev/md0           2025360        6144    1898284   1% /raid0
/dev/md9           3024752        9216    2842176   1% /raidLinear
/dev/md1           1013688        2564     942416   1% /raid1
tmpfs              198768          16     198752   1% /run/user/0
```

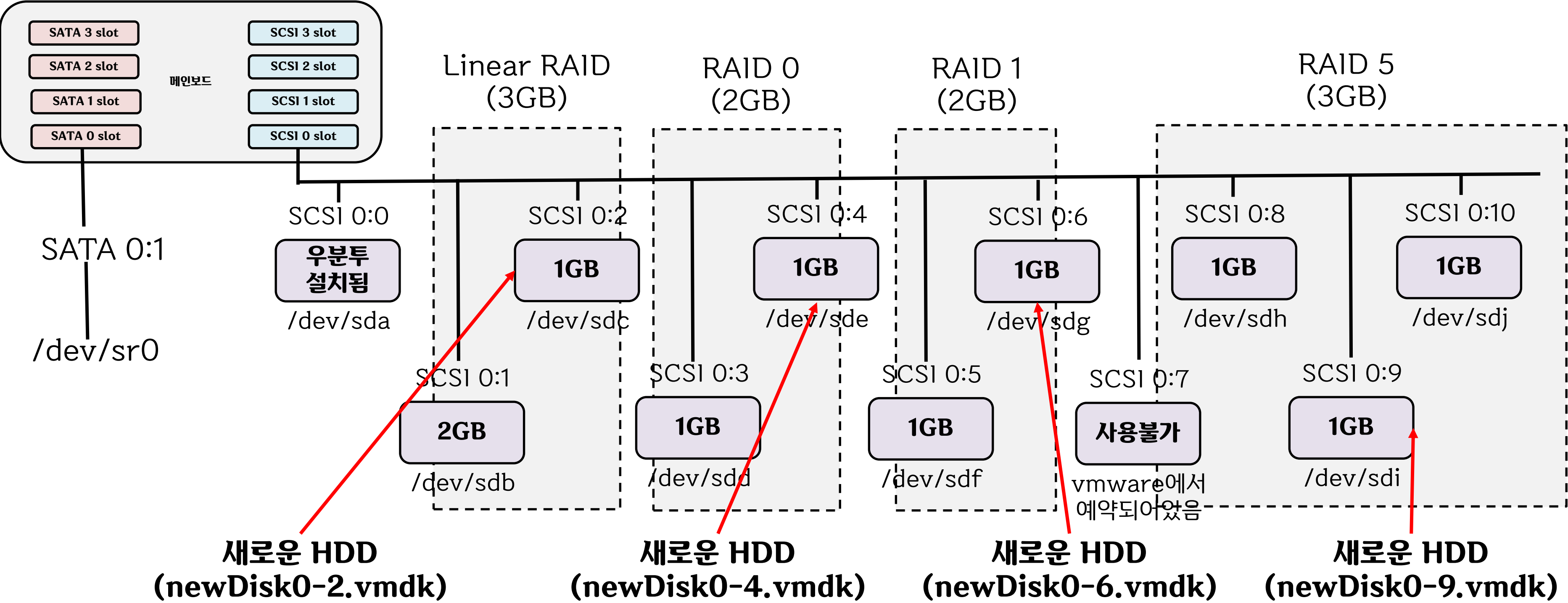
RAID 에서의 문제 발생 테스트

- 9개의 SCSI HDD 장착 후 Linear RAID, RAID 0, RAID 1, RAID 5 에서 HDD가 1개씩 고장났다고 가정



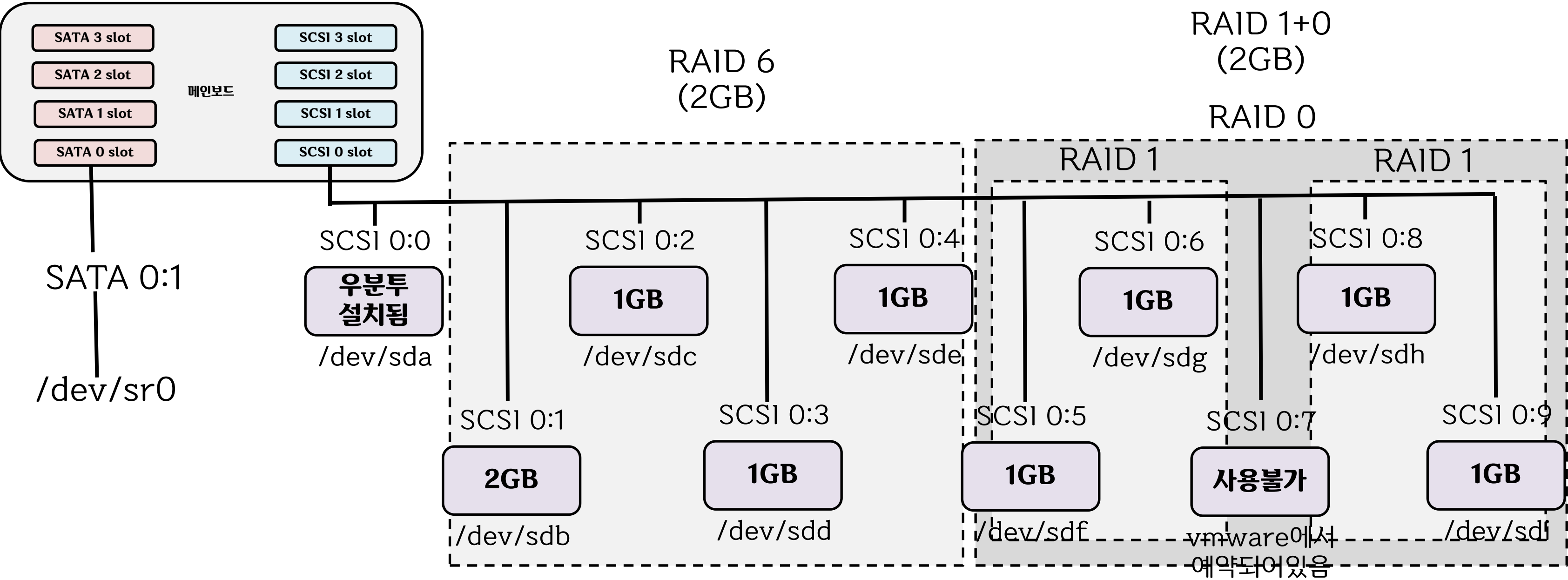
RAID 복구

- HDD가 1개씩 고장난 Linear RAID, RAID 0, RAID 1, RAID 5 를 복구



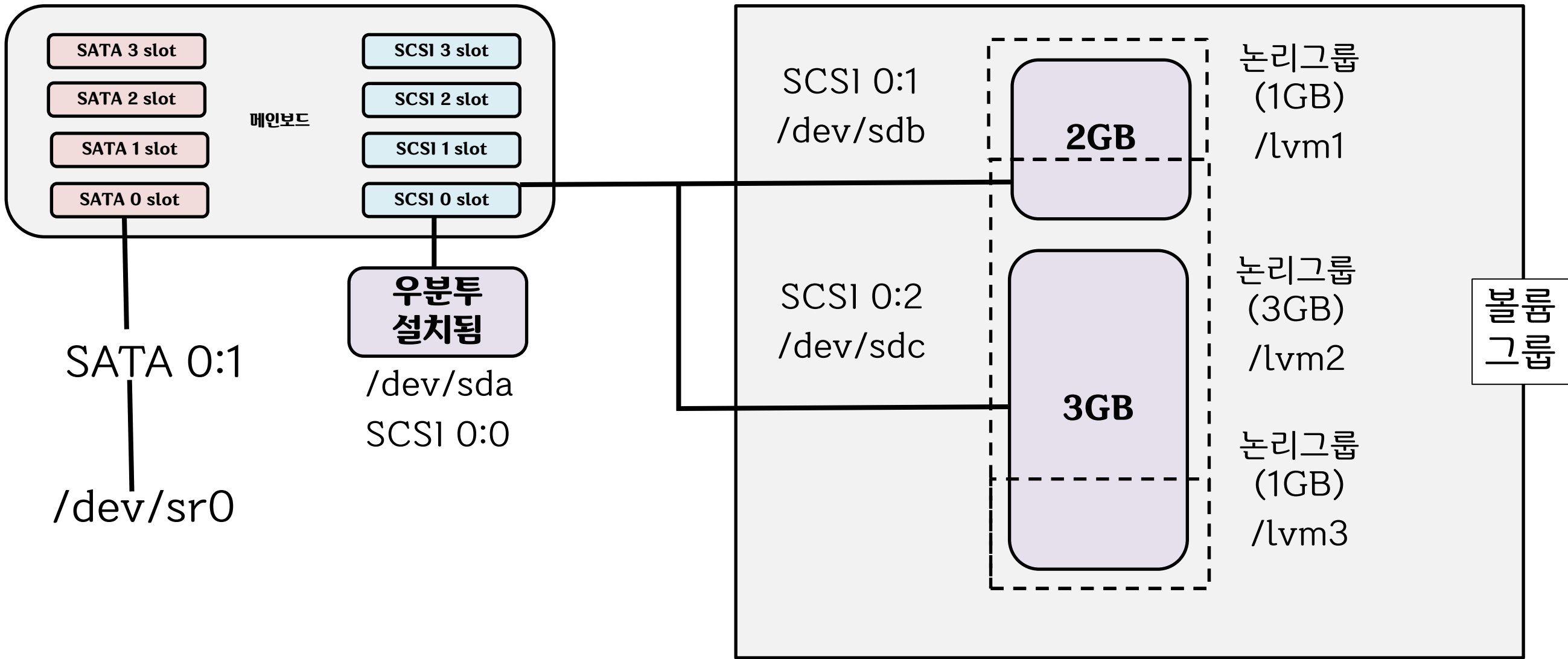
RAID 6 / RAID 1+0 구현

- RAID 5에서 신뢰도를 높인 RAID 6, RAID 1과 RAID 0을 합친 RAID 1+0을 구현

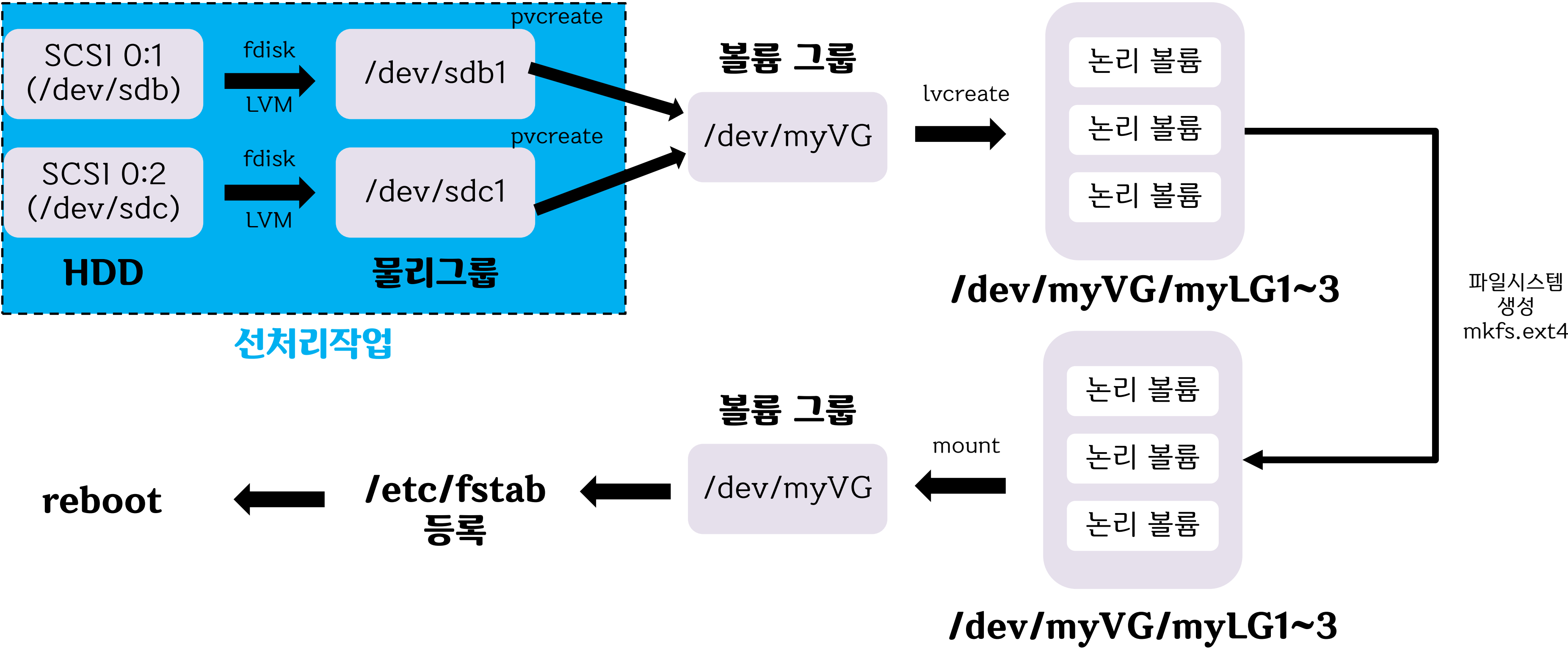


LVM (Logical Volume Manager)

- 논리 하드디스크 관리자로, Linear RAID와 비슷하지만 더 많은 기능을 가짐
- 여러 개의 HDD를 합쳐서 한 개의 파티션으로 구성 후, 필요에 따라 나눠서 사용함
- 물리 볼륨(Physical Volume): /dev/sda1, /dev/sdb1 등의 파티션
- 볼륨 그룹(Volume Group): 물리 볼륨을 합쳐서 1개의 물리 그룹으로 생성
- 논리 볼륨(Logical Volume): 볼륨 그룹을 1개 이상으로 나눈 것으로 논리적 그룹이라고도 함



LVM (Logical Volume Manager)



쿼터(Quota)

- 파일 시스템마다 사용자나 그룹이 생성할 수 있는 파일의 용량과 개수를 제한하는 것
- 일반 사용자들이 사용하는 파일 시스템을 '/'로 지정하지 않고 별도의 파일 시스템을 지정해서 사용하게 함
- 많은 사용자가 동시에 '/'를 사용하면 시스템의 성능이 저하될 수 있음

/etc/fstab
수정



Reboot 또는
Remount



쿼터 DB 생성



개인별 쿼터
설정

/etc/fstab 옵션 부분에
쿼터 관련 부분 추가

#quotacheck,
#quotaon,
#quotaoff

#edquota

#quotacheck

옵션/명령어	의미
-a (All)	모든 파일 시스템을 체크
-u (User)	사용자 쿼터 관련 체크를 시도
-g (Group)	그룹 쿼터 관련 체크를 시도
-m (no-remount)	재마운트를 생략
-n (user-first)	첫 번째로 검색된 것을 사용
-p (print-state)	처리 결과를 출력
-v (Verbose)	파일 시스템의 상태를 보여줌

Filesystem	사용자별 쿼터를 할당하는 파일 시스템
blocks,soft,hard	현재 사용자가 사용하는 블록과 소프트, 하드 사용 한도를 의미 (기본 kb단위) blocks 0: 28KB soft, hard 0: 한도제한이 없음
inodes	inode의 개수를 의미 (파일 개수)
grace	soft 한도를 초과했을 때 사용할 수 있는 기간

1. 다음 중 용량이 2GB 하드디스크 7개를 이용하여 RAID-6로 구성했을 때 가용 공간으로 알맞은 것은?

- ① 8GB ② 10GB
- ③ 12GB ④ 14GB

2. 다음 설명과 같은 상황에서 사용해야 하는 기술로 가장 알맞은 것은?

- ① LVM ② RAID
- ③ Bonding ④ Clustering

/home 디렉터리를 500GB로 구성할 예정이지만, 사용자가 많아질 경우를 대비해서 쉽게 용량 증설이 가능한 환경으로 구축하려고 한다.

3. 다음 설명에 해당하는 RAID 기술로 알맞은 것은?

- ① Volume Group ② Linear
- ③ Striping ④ Mirroring

디스크에 에러 발생 시 데이터의 손실을 막기 위해, 추가로 하나 이상의 장치에 중복 저장하는 기술이다.

4. 다음 그림에 해당하는 명령으로 알맞은 것은?

- ① df ② du
- ③ mount ④ lsblk

```
[root@www ~]#
Filesystem      Type      Size  Used Avail Use% Mounted on
devtmpfs        devtmpfs  2.0G   0    2.0G   0% /dev
tmpfs           tmpfs     2.0G   0    2.0G   0% /dev/shm
tmpfs           tmpfs     2.0G  9.5M  2.0G   1% /run
tmpfs           tmpfs     2.0G   0    2.0G   0% /sys/fs/cgroup
/dev/sda1       xfs       47G   7.4G   40G  16% /
tmpfs           tmpfs     396M   24K   396M   1% /run/user/0
[root@www ~]#
```

5. 다음 설명에 해당하는 RAID의 종류로 알맞은 것은?

- ① RAID-0 ② RAID-1
- ③ RAID-5 ④ RAID-6

최소 3개의 디스크로 구성해야 하고, 패리티 정보를 이용해서 하나의 디스크가 고장이 발생한 경우에도 데이터 사용이 가능한 구성 방식이다. 디스크 3개로 구성 시에 약 33.3%가 패리티 공간으로 사용된다.

6. 다음 중 장착된 디스크들의 파티션 테이블 정보를 확인하는 명령으로 가장 알맞은 것은?

- ① mount -a ② fdisk -l
- ③ df -hT ④ du -h

7. 다음 중 부팅 시에 특정 파티션을 자동으로 마운트 되도록 등록하는 파일로 알맞은 것은?

- ① /etc/mtab ② /etc/fstab
- ③ /etc/partitions ④ /etc/filesystems

8. 다음 중 fdisk 명령 실행 시 파티션 속성을 변경하기 위한 명령으로 알맞은 것은?

- ① d ② n
- ③ p ④ t

9. 다음 중 /etc 디렉터리가 차지하고 있는 전체 용량을 확인할 때 사용하는 명령으로 가장 알맞은 것은?

- ① ls ② df
- ③ du ④ mount

10. 다음 중 분할된 파티션 단위로 사용량을 확인할 때 사용하는 명령으로 알맞은 것은?

- ① df ② du
- ③ mkfs ④ mount

수고하셨습니다