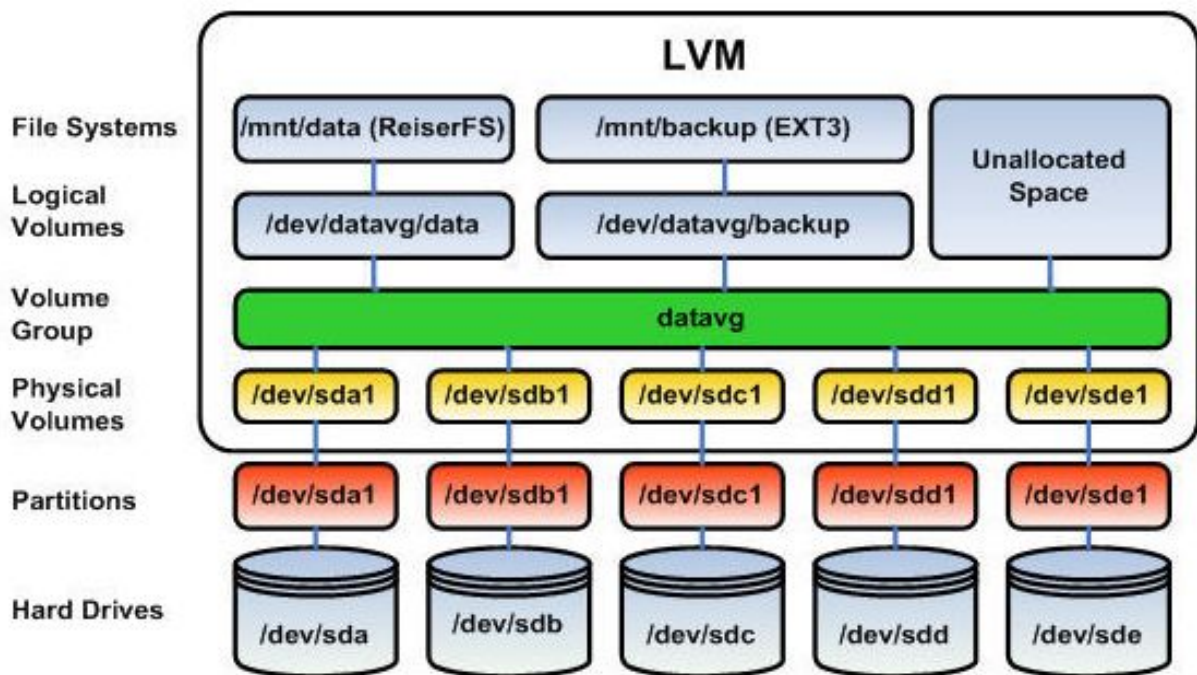


LVM (Logical Volume Manager)

1. LVM이란?

Logical volume management는 디스크나 대용량 스토리지 장치를 유연하고 확장이 가능하게 다룰수 있는 기술이며 이를 리눅스 커널에 구현한 기능은 Logical Volume Manager 이라고 부른다. LVM을 사용하면 여러 물리적인 디스크를 하나의 논리적인 파일시스템으로 구성을 해서 사용할 수 있다. 기존에 파일시스템이 블록 장치에 직접 접근해서 읽고/쓰기를 했다면, LVM은 파일 시스템이 LVM이 만든 가상의 블록 장치에 읽고/쓰기를 하게 된다.



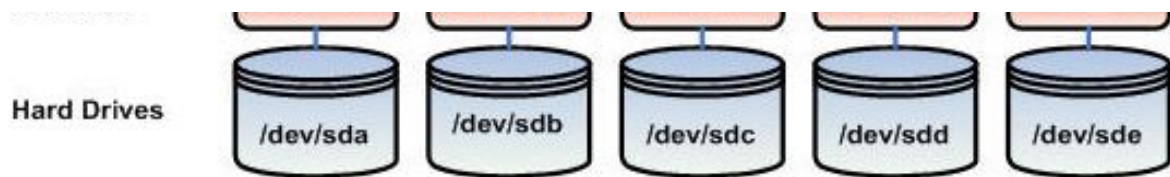
[LVM 구조]

2. 적용과정

- 1) Hard Drives 추가
- 2) 저장장치에 Partitions 설정
- 3) Physical Volumes (PV, 물리볼륨) 생성
- 4) Volume Group (VG, 볼륨그룹) 생성
- 5) Logical Volumes (LV, 논리볼륨) 생성
- 6) 파일 시스템 포맷

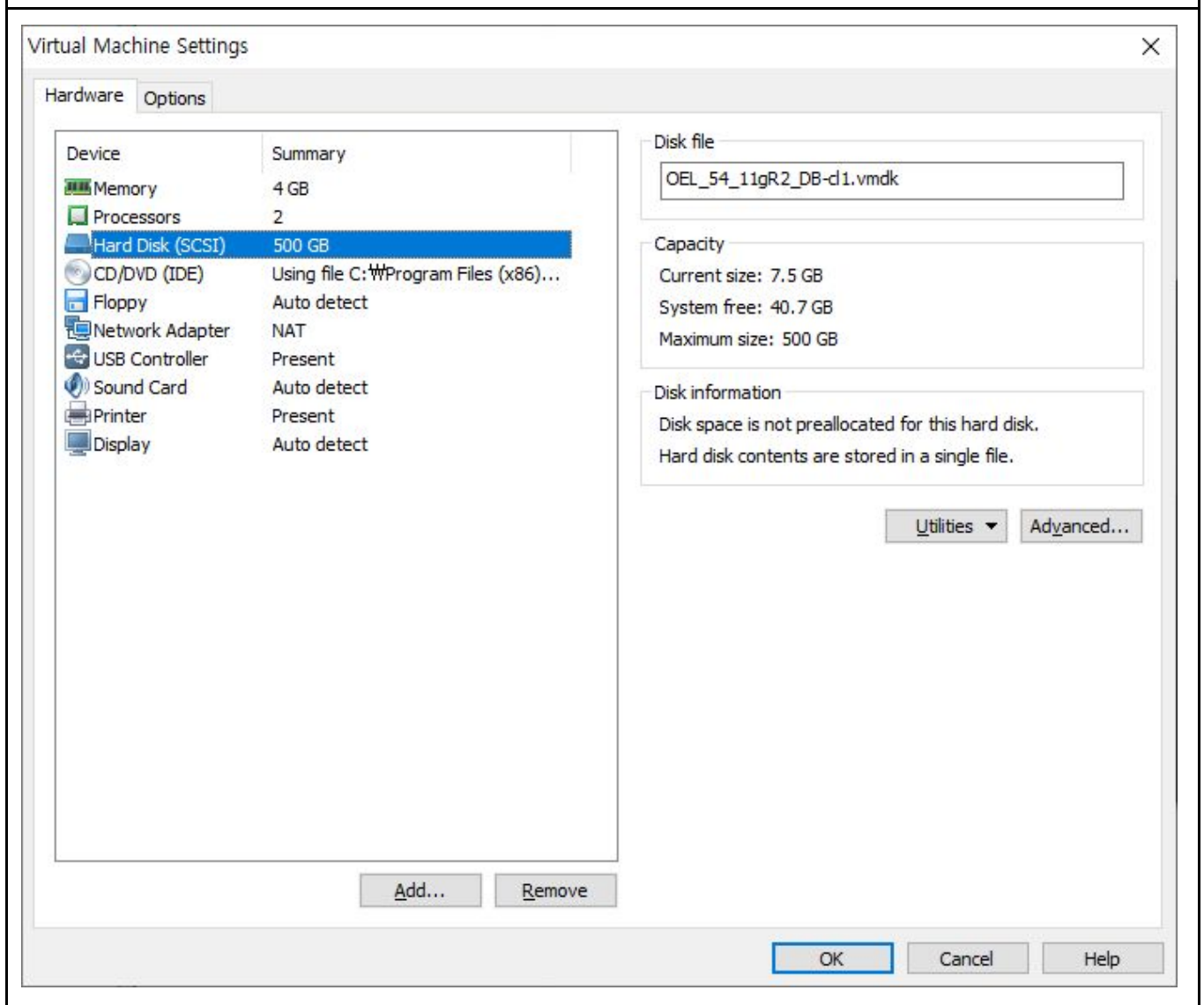
3. LVM 적용

1) Hard Drives 추가



[해당단계]

Edit virtual machine settings 에서 하드 디스크 추가



Add Hardware Wizard

Hardware Type
What type of hardware do you want to install?

Hardware	Explanation
<input checked="" type="radio"/> Hard Disk	Add a hard disk.
<input type="radio"/> CD/DVD Drive	
<input type="radio"/> Floppy Drive	
<input type="radio"/> Network Adapter	
<input type="radio"/> USB Controller	
<input type="radio"/> Sound Card	
<input type="radio"/> Parallel Port	
<input type="radio"/> Serial Port	
<input type="radio"/> Printer	
<input type="radio"/> Generic SCSI Device	

< Back **Next >** Cancel

Add Hardware Wizard

Select a Disk
Which disk do you want to use?

Disk

☒ Create a new virtual disk
A virtual disk is composed of one or more files on the host file system, which will appear as a single hard disk to the guest operating system. Virtual disks can easily be copied or moved on the same host or between hosts.

☐ Use an existing virtual disk
Choose this option to reuse a previously configured disk.

☐ Use a physical disk (for advanced users)
Choose this option to give the virtual machine direct access to a local hard disk.

< Back **Next >** Cancel

Add Hardware Wizard

Select a Disk Type
What kind of disk do you want to create?

Virtual disk type

☐ IDE

☒ SCSI (Recommended)

Mode

☐ Independent
Independent disks are not affected by snapshots.

☒ Persistent
Changes are immediately and permanently written to the disk.

☐ Nonpersistent
Changes to the disk are discarded when you power off or restore a snapshot.

< Back **Next >** Cancel

Add Hardware Wizard

Specify Disk Capacity
How large do you want this disk to be?

Maximum disk size (GB):

Recommended size for Oracle Enterprise Linux 64-bit: 20 GB

☐ Allocate all disk space now.
Allocating the full capacity can enhance performance but requires all of the physical disk space to be available right now. If you do not allocate all the space now, the virtual disk starts small and grows as you add data to it.

☒ Store virtual disk as a single file

☐ Split virtual disk into multiple files
Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.

< Back **Next >** Cancel

Add Hardware Wizard

Specify Disk File
Where would you like to store the disk file?

Disk file
This virtual disk file will store the configuration details of the physical disk.

< Back **Finish** Cancel

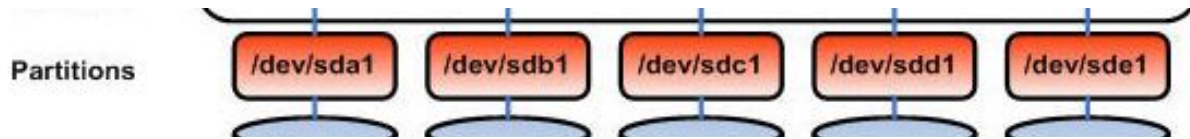
Virtual Machine Settings

Hardware **Options**

Device	Summary
Memory	4 GB
Processors	2
Hard Disk (SCSI)	500 GB
New Hard Disk (S...)	10 GB
New Hard Disk (S...)	10 GB
New Hard Disk (S...)	10 GB
CD/DVD (IDE)	Using file C:\Program Files (x86)...
Floppy	Auto detect
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

※ 동일한 방법으로 disk01 ~ disk03 까지 추가한다.

2) 저장장치에 Partitions 설정



[해당단계]

LVM 패키지 설치여부 확인

```
[root@host1 ~]# rpm -qa | grep lvm
system-config-lvm-1.1.5-1.0.el5
lvm2-2.02.46-8.el5
```

현재 디스크 및 파티션 확인 (만든 디스크를 확인)

```
[root@host1 ~]# fdisk -l
```

```
Disk /dev/sda: 536.8 GB, 536870912000 bytes
255 heads, 63 sectors/track, 65270 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	13	104391	83	Linux
/dev/sda2		14	535	4192965	82	Linux swap / Solaris
/dev/sda3		536	65270	519983887+	83	Linux

```
Disk /dev/sdb: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Disk /dev/sdb doesn't contain a valid partition table
```

```
Disk /dev/sdc: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Disk /dev/sdc doesn't contain a valid partition table
```

```
Disk /dev/sdd: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Disk /dev/sdd doesn't contain a valid partition table
```

fdisk는 파티션을 나누는 기능이다. fdisk로 sdb, sdc, sdd의 파티션을 나눈다.

```
[root@host1 ~]# fdisk /dev/sdb
```

Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.

The number of cylinders for this disk is set to 1305.

There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:

- 1) software that runs at boot time (e.g., old versions of LILO)
- 2) booting and partitioning software from other OSs
(e.g., DOS FDISK, OS/2 FDISK)

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): **n** (새로운 파티션 추가)

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): **1**

First cylinder (1-1305, default 1): **(Enter)**

Using default value 1

Last cylinder or +size or +sizeM or +sizeK (1-1305, default 1305): **(Enter)**

Using default value 1305

Command (m for help): **t** (파티션 시스템 id 변경)

Selected partition 1

Hex code (type L to list codes): **8e** (포맷을 8e로 변경)

Changed system type of partition 1 to 8e (Linux LVM)

Command (m for help): **w** (변경사항 저장)

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

나눈 파티션 확인

```
[root@host1 ~]# fdisk -l
```

Disk /dev/sda: 536.8 GB, 536870912000 bytes

255 heads, 63 sectors/track, 65270 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	13	104391	83	Linux
/dev/sda2		14	535	4192965	82	Linux swap / Solaris
/dev/sda3		536	65270	519983887+	83	Linux

Disk /dev/sdb: 10.7 GB, 10737418240 bytes

255 heads, 63 sectors/track, 1305 cylinders

Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	1305	10482381	8e	Linux LVM

Disk /dev/sdc: 10.7 GB, 10737418240 bytes
 255 heads, 63 sectors/track, 1305 cylinders
 Units = cylinders of 16065 * 512 = 8225280 bytes

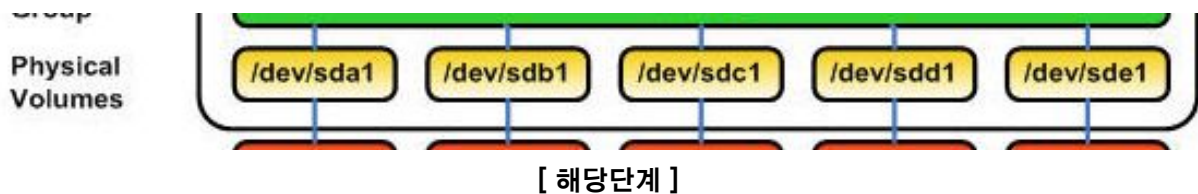
Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		1	1305	10482381	8e	Linux LVM

Disk /dev/sdd: 10.7 GB, 10737418240 bytes
 255 heads, 63 sectors/track, 1305 cylinders
 Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sdd1		1	1305	10482381	8e	Linux LVM

3) Physical Volumes (PV, 물리볼륨) 생성

LVM에서 블록 장치를 사용하려면 PV로 초기화를 해야 한다. 블록 장치를 이루고 있는 파티션들을 LVM에서 사용할 수 있게 물리적으로 변환한 것이다. PV는 일정한 크기의 PE(Physical Extent)들로 구성이 된다.



물리 볼륨 생성

```
[root@host1 ~]# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
[root@host1 ~]# pvcreate /dev/sdc1
Physical volume "/dev/sdc1" successfully created
[root@host1 ~]# pvcreate /dev/sdd1
Physical volume "/dev/sdd1" successfully created
```

물리 볼륨 확인

```
[root@host1 ~]# pvdisplay

/dev/cdrom: open failed: No medium found
"/dev/sdb1" is a new physical volume of "10.00 GB"
--- NEW Physical volume ---
PV Name                /dev/sdb1
VG Name
PV Size                10.00 GB
Allocatable            NO
PE Size (KByte)        0
Total PE              0
Free PE                0
Allocated PE           0
PV UUID                cQK31E-0S1R-kdME-bg8r-FVYZ-IZjX-FSVydf
```



```

"/dev/sdc1" is a new physical volume of "10.00 GB"
--- NEW Physical volume ---
PV Name           /dev/sdc1
VG Name
PV Size           10.00 GB
Allocatable       NO
PE Size (KByte)   0
Total PE         0
Free PE          0
Allocated PE      0
PV UUID          3B9hSu-uEZq-aT2j-sSNQ-R9oV-1Xcn-Hnd7Ru

"/dev/sdd1" is a new physical volume of "10.00 GB"
--- NEW Physical volume ---
PV Name           /dev/sdd1
VG Name
PV Size           10.00 GB
Allocatable       NO
PE Size (KByte)   0
Total PE         0
Free PE          0
Allocated PE      0
PV UUID          5rbCCr-mi2X-2Qhx-876m-Waey-i64z-zbVKNm

```

4) Volume Group (VG, 볼륨그룹) 생성

PV들의 집합으로 LV를 할당할 수 있는 공간이다. 할당할 수 있는 디스크 공간의 풀(Pool)을 생성하는 것으로 보면 된다. 사용자는 VG 안에서 원하는대로 공간을 쪼개서 LV로 만들 수 있다.



볼륨 그룹 생성

```

[root@host1 ~]# vgcreate vg01 /dev/sdb1 /dev/sdc1 /dev/sdd1
Volume group "vg01" successfully created

```

볼륨 그룹 확인

```

[root@host1 ~]# pvdiskdisplay
--- Physical volume ---
PV Name           /dev/sdb1
VG Name           vg01
PV Size           10.00 GB / not usable 717.00 KB
Allocatable       yes
PE Size (KByte)   4096
Total PE         2559
Free PE          2559
Allocated PE      0
PV UUID          cQK31E-0S1R-kdME-bg8r-FVYZ-IZjX-FSVydf

```

```

--- Physical volume ---
PV Name                /dev/sdc1
VG Name                vg01
PV Size                10.00 GB / not usable 717.00 KB
Allocatable            yes
PE Size (KByte)        4096
Total PE               2559
Free PE                2559
Allocated PE           0
PV UUID                3B9hSu-uEZq-aT2j-sSNQ-R9oV-1Xcn-Hnd7Ru

--- Physical volume ---
PV Name                /dev/sdd1
VG Name                vg01
PV Size                10.00 GB / not usable 717.00 KB
Allocatable            yes
PE Size (KByte)        4096
Total PE               2559
Free PE                2559
Allocated PE           0
PV UUID                5rbCCr-mi2X-2Qhx-876m-Waey-i64z-zbVKNm

```

5) Logical Volumes (LV, 논리볼륨) 생성

사용자가 최종적으로 다루게 되는 스토리지



[해당단계]

볼륨 그룹을 논리적 볼륨으로 생성

-L : 볼륨 그룹의 크기 생성

-n : 볼륨 그룹의 이름

※ 데이터파일 생성

```

[root@host1 ~]# lvcreate -L 1G -n lv1000M01 vg01
Logical volume "lv1000M01" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M02 vg01
Logical volume "lv1000M02" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M03 vg01
Logical volume "lv1000M03" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M04 vg01
Logical volume "lv1000M04" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M05 vg01
Logical volume "lv1000M05" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M06 vg01
Logical volume "lv1000M06" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M07 vg01
Logical volume "lv1000M07" created
[root@host1 ~]# lvcreate -L 1G -n lv1000M08 vg01
Logical volume "lv1000M08" created

```


※ 컨트롤파일 생성

```
[root@host1 ~]# lvcreate -L 300M -n lv0300M01 vg01
Logical volume "lv0300M01" created
[root@host1 ~]# lvcreate -L 300M -n lv0300M02 vg01
Logical volume "lv0300M02" created
```

※ 리두로그파일 생성

```
[root@host1 ~]# lvcreate -L 500M -n lv0500M01 vg01
Logical volume "lv0500M01" created
[root@host1 ~]# lvcreate -L 500M -n lv0500M02 vg01
Logical volume "lv0500M02" created
[root@host1 ~]# lvcreate -L 500M -n lv0500M03 vg01
Logical volume "lv0500M03" created
[root@host1 ~]# lvcreate -L 500M -n lv0500M04 vg01
Logical volume "lv0500M04" created
[root@host1 ~]# lvcreate -L 500M -n lv0500M05 vg01
Logical volume "lv0500M05" created
[root@host1 ~]# lvcreate -L 500M -n lv0500M06 vg01
Logical volume "lv0500M06" created
```

논리적 볼륨 확인

```
[root@host1 ~]# lvscan -v
Finding all logical volumes
ACTIVE                '/dev/vg01/lv1000M01' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M02' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M03' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M04' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M05' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M06' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M07' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv1000M08' [1.00 GB] inherit
ACTIVE                '/dev/vg01/lv0300M01' [300.00 MB] inherit
ACTIVE                '/dev/vg01/lv0300M02' [300.00 MB] inherit
ACTIVE                '/dev/vg01/lv0500M01' [500.00 MB] inherit
ACTIVE                '/dev/vg01/lv0500M02' [500.00 MB] inherit
ACTIVE                '/dev/vg01/lv0500M03' [500.00 MB] inherit
ACTIVE                '/dev/vg01/lv0500M04' [500.00 MB] inherit
ACTIVE                '/dev/vg01/lv0500M05' [500.00 MB] inherit
ACTIVE                '/dev/vg01/lv0500M06' [500.00 MB] inherit

[root@host1 ~]# lvdisplay
--- Logical volume ---
LV Name                /dev/vg01/lv1000M01
VG Name                vg01
LV UUID                yP4jos-SpI5-1NhH-pLZv-rRlw-O9up-tsgOiI
LV Write Access        read/write
LV Status              available
# open                 0
LV Size                1.00 GB
Current LE             256
Segments              1
Allocation             inherit
Read ahead sectors     auto
- currently set to    256
Block device           253:0
```

....

6) 파일 시스템 포맷

File Systems

/mnt/data (ReiserFS)

/mnt/backup (EXT3)

Unallocated

[해당단계]

서버가 기동될때마다 항상 등록된 명령어가 작동할 수 있도록 데몬파일을 수정한다.

```
[root@host1 ~]# cat >> /etc/rc5.d/S91ora_start <<EOF
> !/bin/bash
>
> su - root -c '/bin/raw /dev/raw/raw1 /dev/vg01/lv1000M01'
> su - root -c '/bin/raw /dev/raw/raw2 /dev/vg01/lv1000M02'
> su - root -c '/bin/raw /dev/raw/raw3 /dev/vg01/lv1000M03'
> su - root -c '/bin/raw /dev/raw/raw4 /dev/vg01/lv1000M04'
> su - root -c '/bin/raw /dev/raw/raw5 /dev/vg01/lv1000M05'
> su - root -c '/bin/raw /dev/raw/raw6 /dev/vg01/lv1000M06'
> su - root -c '/bin/raw /dev/raw/raw7 /dev/vg01/lv1000M07'
> su - root -c '/bin/raw /dev/raw/raw8 /dev/vg01/lv1000M08'
> su - root -c '/bin/raw /dev/raw/raw9 /dev/vg01/lv0300M01'
> su - root -c '/bin/raw /dev/raw/raw10 /dev/vg01/lv0300M02'
> su - root -c '/bin/raw /dev/raw/raw11 /dev/vg01/lv0500M01'
> su - root -c '/bin/raw /dev/raw/raw12 /dev/vg01/lv0500M02'
> su - root -c '/bin/raw /dev/raw/raw13 /dev/vg01/lv0500M03'
> su - root -c '/bin/raw /dev/raw/raw14 /dev/vg01/lv0500M04'
> su - root -c '/bin/raw /dev/raw/raw15 /dev/vg01/lv0500M05'
> su - root -c '/bin/raw /dev/raw/raw16 /dev/vg01/lv0500M06'

> su - root -c 'chmod 666 /dev/raw/raw1'
> su - root -c 'chmod 666 /dev/raw/raw2'
> su - root -c 'chmod 666 /dev/raw/raw3'
> su - root -c 'chmod 666 /dev/raw/raw4'
> su - root -c 'chmod 666 /dev/raw/raw5'
> su - root -c 'chmod 666 /dev/raw/raw6'
> su - root -c 'chmod 666 /dev/raw/raw7'
> su - root -c 'chmod 666 /dev/raw/raw8'
> su - root -c 'chmod 666 /dev/raw/raw9'
> su - root -c 'chmod 666 /dev/raw/raw10'
> su - root -c 'chmod 666 /dev/raw/raw11'
> su - root -c 'chmod 666 /dev/raw/raw12'
> su - root -c 'chmod 666 /dev/raw/raw13'
> su - root -c 'chmod 666 /dev/raw/raw14'
> su - root -c 'chmod 666 /dev/raw/raw15'
> su - root -c 'chmod 666 /dev/raw/raw16'

> su - root -c 'chown oracle:dba /dev/raw/raw1'
> su - root -c 'chown oracle:dba /dev/raw/raw2'
> su - root -c 'chown oracle:dba /dev/raw/raw3'
> su - root -c 'chown oracle:dba /dev/raw/raw4'
> su - root -c 'chown oracle:dba /dev/raw/raw5'
> su - root -c 'chown oracle:dba /dev/raw/raw6'
> su - root -c 'chown oracle:dba /dev/raw/raw7'
> su - root -c 'chown oracle:dba /dev/raw/raw8'
> su - root -c 'chown oracle:dba /dev/raw/raw9'
> su - root -c 'chown oracle:dba /dev/raw/raw10'
> su - root -c 'chown oracle:dba /dev/raw/raw11'
> su - root -c 'chown oracle:dba /dev/raw/raw12'
> su - root -c 'chown oracle:dba /dev/raw/raw13'
> su - root -c 'chown oracle:dba /dev/raw/raw14'
> su - root -c 'chown oracle:dba /dev/raw/raw15'
> su - root -c 'chown oracle:dba /dev/raw/raw16'
> EOF
```

부팅될 때마다 데몬파일이 자동으로 실행될 수 있도록 설정

```
[root@host1 ~]# chmod 777 /etc/rc5.d/S91lora_start  
[root@host1 ~]# reboot
```

재시작 후 확인

```
[root@host1 ~]# /bin/raw -qa  
/dev/raw/raw1: bound to major 253, minor 0  
/dev/raw/raw2: bound to major 253, minor 1  
/dev/raw/raw3: bound to major 253, minor 2  
/dev/raw/raw4: bound to major 253, minor 3  
/dev/raw/raw5: bound to major 253, minor 4  
/dev/raw/raw6: bound to major 253, minor 5  
/dev/raw/raw7: bound to major 253, minor 6  
/dev/raw/raw8: bound to major 253, minor 7  
/dev/raw/raw9: bound to major 253, minor 8  
/dev/raw/raw10: bound to major 253, minor 9  
/dev/raw/raw11: bound to major 253, minor 10  
/dev/raw/raw12: bound to major 253, minor 11  
/dev/raw/raw13: bound to major 253, minor 12  
/dev/raw/raw14: bound to major 253, minor 13  
/dev/raw/raw15: bound to major 253, minor 14  
/dev/raw/raw16: bound to major 253, minor 15  
  
[root@host1 ~]# ls -l /dev/raw/raw*  
crw-rw-rw- 1 oracle dba 162, 1 Jul 13 16:24 /dev/raw/raw1  
crw-rw-rw- 1 oracle dba 162, 10 Jul 13 16:24 /dev/raw/raw10  
crw-rw-rw- 1 oracle dba 162, 11 Jul 13 16:24 /dev/raw/raw11  
crw-rw-rw- 1 oracle dba 162, 12 Jul 13 16:24 /dev/raw/raw12  
crw-rw-rw- 1 oracle dba 162, 13 Jul 13 16:24 /dev/raw/raw13  
crw-rw-rw- 1 oracle dba 162, 14 Jul 13 16:24 /dev/raw/raw14  
crw-rw-rw- 1 oracle dba 162, 15 Jul 13 16:24 /dev/raw/raw15  
crw-rw-rw- 1 oracle dba 162, 16 Jul 13 16:24 /dev/raw/raw16  
crw-rw-rw- 1 oracle dba 162, 1 Jul 13 16:24 /dev/raw/raw2  
crw-rw-rw- 1 oracle dba 162, 3 Jul 13 16:24 /dev/raw/raw3  
crw-rw-rw- 1 oracle dba 162, 4 Jul 13 16:24 /dev/raw/raw4  
crw-rw-rw- 1 oracle dba 162, 5 Jul 13 16:24 /dev/raw/raw5  
crw-rw-rw- 1 oracle dba 162, 6 Jul 13 16:24 /dev/raw/raw6  
crw-rw-rw- 1 oracle dba 162, 7 Jul 13 16:24 /dev/raw/raw7  
crw-rw-rw- 1 oracle dba 162, 8 Jul 13 16:24 /dev/raw/raw8  
crw-rw-rw- 1 oracle dba 162, 9 Jul 13 16:24 /dev/raw/raw9
```

4. DB 소프트웨어 (Oracle) 설치

oracle 사용자로 로그인하여 ./runInstaller 로 설치한다.

```
[oracle@host1 ~]$ cd /stage/database

[oracle@host1 database]$ ls
doc      readme.html  rpm          sshsetup    welcome.html
install  response    runInstaller stage

[oracle@host1 database]$ ./runInstaller
```

listener 설치

```
[oracle@host1 ~]$ netca
```

```
Oracle Net Services Configuration:
Configuring Listener:LISTENER
Listener configuration complete.
Oracle Net Listener Startup:
Running Listener Control:
/u01/app/oracle/product/11.2.0.3/dbhome_1/bin/lsnrctl start LISTENER
Listener Control complete.
Listener started successfully.
Oracle Net Services configuration successful. The exit code is 0
```

```
[oracle@host1 ~]$ lsnrctl status
```

```
LSNRCTL for Linux: Version 11.2.0.3.0 - Production on 13-JUL-2020 16:57:35
```

```
Copyright (c) 1991, 2011, Oracle. All rights reserved.
```

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=host1) (PORT=1521)))
STATUS of the LISTENER
```

```
-----
Alias                LISTENER
Version              TNSLSNR for Linux: Version 11.2.0.3.0 - Production
Start Date           13-JUL-2020 16:57:27
Uptime                0 days 0 hr. 0 min. 8 sec
Trace Level           off
Security              ON: Local OS Authentication
SNMP                 OFF
Listener Parameter File
/u01/app/oracle/product/11.2.0.3/dbhome_1/network/admin/listener.ora
Listener Log File     /u01/app/oracle/diag/tnslsnr/host1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=host1) (PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=EXTPROC1521)))
The listener supports no services
The command completed successfully
```

5. 데이터베이스를 OS레벨에 생성한 후 LVM에 파일 포맷

PFILE을 만든 후 SPFILE을 생성한다. 그리고 NOMOUNT 상태로 오픈한다.

```
[oracle@host1 oradata]$ mkdir -p /u01/app/oracle/oradata/PROD
```

```
[oracle@host1 oradata]$ cd $ORACLE_HOME/dbs
```

```
[oracle@host1 dbs]$ ls
```

```
init.ora
```

```
[oracle@host1 dbs]$ orapwd file=orapwPROD password=oracle_4U
```

```
[oracle@host1 dbs]$ ls
```

```
init.ora orapwPROD
```

```
[oracle@host1 dbs]$ cat >> initPROD.ora <<EOF
> db_name=PROD
> service_names=PROD
> control_files='/dev/raw/raw9','/dev/raw/raw10'
> sga_target=400M
> pga_aggregate_target=150M
> db_block_size=8192
> remote_login_passwordfile='EXCLUSIVE'
> undo_tablespace='UNDOTBS1'
> EOF
```

```
[oracle@host1 dbs]$ ls
```

```
init.ora initPROD.ora orapwPROD
```

```
[oracle@host1 dbs]$ sqlplus / as sysdba
```

```
Connected to an idle instance.
```

```
SQL> startup nomount;
```

```
ORACLE instance started.
```

```
SQL> create spfile from pfile;
```

```
File created.
```

```
SQL> startup nomount;
```

```
ORACLE instance started.
```

데이터베이스 PROD 생성

```
SQL> CREATE DATABASE PROD
2  USER SYS IDENTIFIED BY oracle_4U
3  USER SYSTEM IDENTIFIED BY oracle_4U
4  LOGFILE GROUP 1 ('/dev/raw/raw11' ,'/dev/raw/raw12') SIZE 499M,
5  GROUP 2 ('/dev/raw/raw13' ,'/dev/raw/raw14') SIZE 499M
6  CHARACTER SET AL32UTF8
7  NATIONAL CHARACTER SET AL16UTF16
8  EXTENT MANAGEMENT LOCAL
9  DATAFILE '/dev/raw/raw1' SIZE 999M AUTOEXTEND OFF
10 SYSAUX
11 DATAFILE '/dev/raw/raw2' SIZE 999M AUTOEXTEND OFF
12 DEFAULT TABLESPACE USERS
13 DATAFILE '/dev/raw/raw3' SIZE 999M AUTOEXTEND OFF
14 DEFAULT TEMPORARY TABLESPACE TEMP
15 TEMPFILE '/dev/raw/raw4' SIZE 999M AUTOEXTEND OFF
16 UNDO TABLESPACE UNDOTBS1
17 DATAFILE '/dev/raw/raw5' SIZE 999M AUTOEXTEND OFF;
```

```
Database created.
```

CATALOG.SQL : DATA dictionary view, Dynamic performance view 생성

```
SQL> @?/rdbms/admin/catalog.sql
```

CATPROC.SQL : PL/SQL 관련된 패키지 및 프로시저를 생성

```
SQL> @?/rdbms/admin/catproc.sql
```

PUPBLD.SQL : Product User Profile 테이블 및 관련 프로시저를 생성

```
SQL> connect system/oracle_4U  
Connected.
```

```
SQL> @?/sqlplus/admin/pupbld.sql
```

만든 데이터베이스 정보 확인

```
[oracle@host1 ~]$ cat /etc/oratab
```

```
# This file is used by ORACLE utilities.  It is created by root.sh  
# and updated by either Database Configuration Assistant while creating  
# a database or ASM Configuration Assistant while creating ASM instance.
```

```
# A colon, ':', is used as the field terminator.  A new line terminates  
# the entry.  Lines beginning with a pound sign, '#', are comments.
```

```
#
```

```
# Entries are of the form:
```

```
# $ORACLE_SID:$ORACLE_HOME:<N|Y>:
```

```
#
```

```
# The first and second fields are the system identifier and home
```

```
# directory of the database respectively.  The third field indicates
```

```
# to the dbstart utility that the database should , "Y", or should not,
```

```
# "N", be brought up at system boot time.
```

```
#
```

```
# Multiple entries with the same $ORACLE_SID are not allowed.
```

```
#
```

```
#
```

```
PROD:/u01/app/oracle/product/11.2.0.3/dbhome_1:N (자동실행여부)
```

```
[oracle@host1 ~]$ . oraenv
```

```
ORACLE_SID = [PROD] ? PROD
```

```
The Oracle base remains unchanged with value /u01/app/oracle
```

```
[oracle@host1 ~]$ sqlplus / as sysdba
```

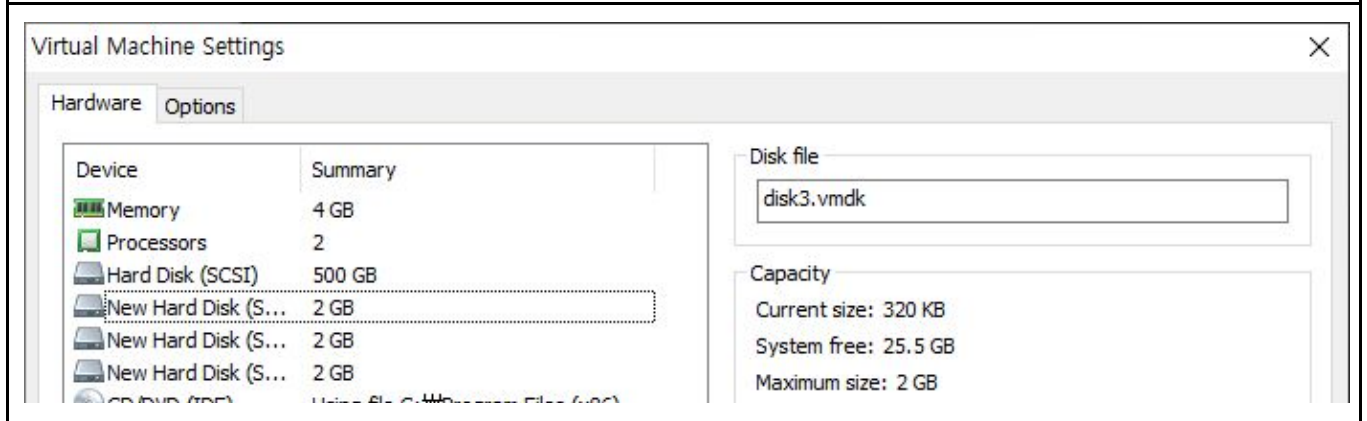
```
SQL> select file_name, tablespace_name  
2 from dba_data_files;
```

FILE_NAME	TABLESPACE_NAME
/dev/raw/raw1	SYSTEM
/dev/raw/raw2	SYSAUX
/dev/raw/raw5	UNDOTBS1
/dev/raw/raw3	USERS

※ LVM에 DB 소프트웨어 (Oracle) 설치하기

1) Hard Drives 추가

2G짜리 하드를 3개 추가한다.



2) 저장장치에 Partitions 설정

현재 디스크 및 파티션 확인 (만든 디스크를 확인)

```
[root@host1 ~]# fdisk -l
```

```
Disk /dev/sda: 536.8 GB, 536870912000 bytes
255 heads, 63 sectors/track, 65270 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	13	104391	83	Linux
/dev/sda2		14	535	4192965	82	Linux swap / Solaris
/dev/sda3		536	65270	519983887+	83	Linux

```
Disk /dev/sdb: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Disk /dev/sdb doesn't contain a valid partition table
```

```
Disk /dev/sdc: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Disk /dev/sdc doesn't contain a valid partition table
```

```
Disk /dev/sdd: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Disk /dev/sdd doesn't contain a valid partition table
```

fdisk로 sdb, sdc, sdd의 파티션을 나눈다.

```
[root@host1 ~]# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.

The number of cylinders for this disk is set to 1305.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n (새로운 파티션 추가)
Command action
   e   extended
   p   primary partition (1-4)
p

Partition number (1-4): 1

First cylinder (1-1305, default 1): (Enter)
Using default value 1

Last cylinder or +size or +sizeM or +sizeK (1-1305, default 1305): (Enter)
Using default value 1305

Command (m for help): t (파티션 시스템 id 변경)
Selected partition 1
Hex code (type L to list codes): 8e (포맷을 8e로 변경)
Changed system type of partition 1 to 8e (Linux LVM)

Command (m for help): w (변경사항 저장)
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

나눈 파티션 확인

```
[root@host1 ~]# fdisk -l

Disk /dev/sda: 536.8 GB, 536870912000 bytes
255 heads, 63 sectors/track, 65270 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1    *           1          13       104391   83   Linux
/dev/sda2             14          535      4192965   82   Linux swap / Solaris
/dev/sda3          536         65270     519983887+   83   Linux
```

```
Disk /dev/sdb: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	261	2096451	8e	Linux LVM

```
Disk /dev/sdc: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		1	261	2096451	8e	Linux LVM

```
Disk /dev/sdd: 2147 MB, 2147483648 bytes
255 heads, 63 sectors/track, 261 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdd1		1	261	2096451	8e	Linux LVM

3) Physical Volumes (PV, 물리볼륨) 생성

물리 볼륨 생성

```
[root@host1 ~]# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
[root@host1 ~]# pvcreate /dev/sdc1
Physical volume "/dev/sdc1" successfully created
[root@host1 ~]# pvcreate /dev/sdd1
Physical volume "/dev/sdd1" successfully created
```

물리 볼륨 확인

```
[root@host1 ~]# pvdisplay

/dev/cdrom: open failed: No medium found
"/dev/sdb1" is a new physical volume of "2.00 GB"
--- NEW Physical volume ---
PV Name                /dev/sdb1
VG Name
PV Size                2.00 GB
Allocatable            NO
PE Size (KByte)        0
Total PE               0
Free PE                0
Allocated PE           0
PV UUID                iLlWFq-CmPK-RWmz-Xluv-60uQ-VUWx-DdoeON
```

```

"/dev/sdc1" is a new physical volume of "2.00 GB"
--- NEW Physical volume ---
PV Name                /dev/sdc1
VG Name
PV Size                2.00 GB
Allocatable           NO
PE Size (KByte)       0
Total PE              0
Free PE               0
Allocated PE          0
PV UUID               fDKh7D-VlCk-l8ak-6TVo-Lpj1-qzOE-Sbbrga

"/dev/sdd1" is a new physical volume of "2.00 GB"
--- NEW Physical volume ---
PV Name                /dev/sdd1
VG Name
PV Size                2.00 GB
Allocatable           NO
PE Size (KByte)       0
Total PE              0
Free PE               0
Allocated PE          0
PV UUID               5UXwKG-pLuH-OfsK-kBQL-EOiK-bWK6-BaEYMB

```

4) Volume Group (VG, 볼륨그룹) 생성

볼륨 그룹 생성

```

[root@host1 ~]# vgcreate vg01 /dev/sdb1 /dev/sdc1 /dev/sdd1
Volume group "vg01" successfully created

```

볼륨 그룹 확인

```

[root@host1 ~]# vgdisplay

--- Volume group ---
VG Name                vg01
System ID
Format                lvm2
Metadata Areas         3
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               3
Act PV               3
VG Size                5.99 GB
PE Size                4.00 MB
Total PE              1533
Alloc PE / Size        0 / 0
Free PE / Size         1533 / 5.99 GB
VG UUID               GtzTCT-ixJ3-BvpH-m9hp-m0Wx-PSZ1-diCnyL

```

5) Logical Volumes (LV, 논리볼륨) 생성

오라클(약 4.5G)을 설치하기 위해 5G를 할당한다.

```
[root@host1 ~]# lvcreate -L 5G -n lv5000M01 vg01
Logical volume "lv5000M01" created
```

논리 볼륨 확인하기

```
[root@host1 ~]# lvdisplay

--- Logical volume ---
LV Name                /dev/vg01/lv5000M01
VG Name                 vg01
LV UUID                 5yEL9x-JaTV-jXhi-Gyef-msXc-z2Wq-66Bcz0
LV Write Access         read/write
LV Status                available
# open                  0
LV Size                  5.00 GB
Current LE               1280
Segments                 3
Allocation               inherit
Read ahead sectors      auto
 - currently set to     256
Block device             253:0
```

만든 볼륨 확인

```
[root@host1 ~]# ls -l /dev/vg01/
total 0
lrwxrwxrwx 1 root root 26 Jul 14 12:00 lv5000M01 -> /dev/mapper/vg01-lv5000M01
```

6) 파일시스템 포맷 및 마운트

파일시스템 포맷

```
[root@host1 ~]# mkfs.ext3 /dev/vg01/lv5000M01
```

※ 마운트를 위해 파일시스템 ext3 타입으로 포맷

```
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
655360 inodes, 1310720 blocks
65536 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1342177280
40 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736
```

```
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 32 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
```

오라클 폴더에 논리 볼륨을 마운트한다.

```
[root@host1 ~]# mount -t ext3 /dev/vg01/lv5000M01 /u01/app/oracle

[root@host1 ~]# df -l
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sda3	503696908	7181760	470515956	2%	/
/dev/sda1	101086	11836	84031	13%	/boot
tmpfs	2021220	0	2021220	0%	/dev/shm
/dev/mapper/vg01-lv5000M01	5160576	141440	4756992	3%	/u01/app/oracle

데몬파일로 부팅 시마다 자동으로 켜질 수 있도록 설정

```
[root@host1 ~]# vi /etc/fstab
```

※ /dev/vg01/lv5000M01 /u01/app/oracle ext3 defaults 0 0 추가

LABEL=/	/	ext3	defaults	1	1
LABEL=/boot	/boot	ext3	defaults	1	2
tmpfs	/dev/shm	tmpfs	defaults	0	0
devpts	/dev/pts	devpts	gid=5,mode=620	0	0
sysfs	/sys	sysfs	defaults	0	0
proc	/proc	proc	defaults	0	0
LABEL=SWAP-sda2	swap	swap	defaults	0	0
/dev/vg01/lv5000M01	/u01/app/oracle	ext3	defaults	0	0

부팅 시 자동실행을 위해 데몬파일 수정한 후 권한 부여

```
[root@host1 ~]# cat >> /etc/rc5.d/S91ora_start <<EOF
> !/bin/bash
>
> su - root -c 'chown -R oracle:oinstall /u01'
> su - root -c 'chmod -R 775 /u01'
> EOF

[root@host1 ~]# chmod 777 /etc/rc5.d/S91ora_start
[root@host1 ~]# reboot
```


7) DB 소프트웨어 설치 (Oracle)

oracle 사용자로 로그인하여 ./runInstaller 로 설치한다.

```
[oracle@host1 ~]$ cd /stage/database  
  
[oracle@host1 database]$ ls  
doc          readme.html  rpm          sshsetup    welcome.html  
install      response    runInstaller stage  
  
[oracle@host1 database]$ ./runInstaller
```

설치 후 볼륨 확인

```
[oracle@host1 database]$ df -l
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sda3	503696908	7188600	470509116	2%	/
/dev/sda1	101086	11836	84031	13%	/boot
tmpfs	2021220	0	2021220	0%	/dev/shm
/dev/mapper/vg01-lv5000M01	5160576	4381088	517344	90%	/u01/app/oracle