



# Advanced Oracle Document

## 11gR2

## 향상된 ASM(ADVM 과 ACFS) (Oracle 11g R2 11.2.0.2.0)

Author: Hyun-Ho, Jung  
Site: <http://www.commit.co.kr>  
E-MAIL: [admin@commit.co.kr](mailto:admin@commit.co.kr)  
[cleanto@naver.com](mailto:cleanto@naver.com)  
Creation Date: 2011-06-01

---

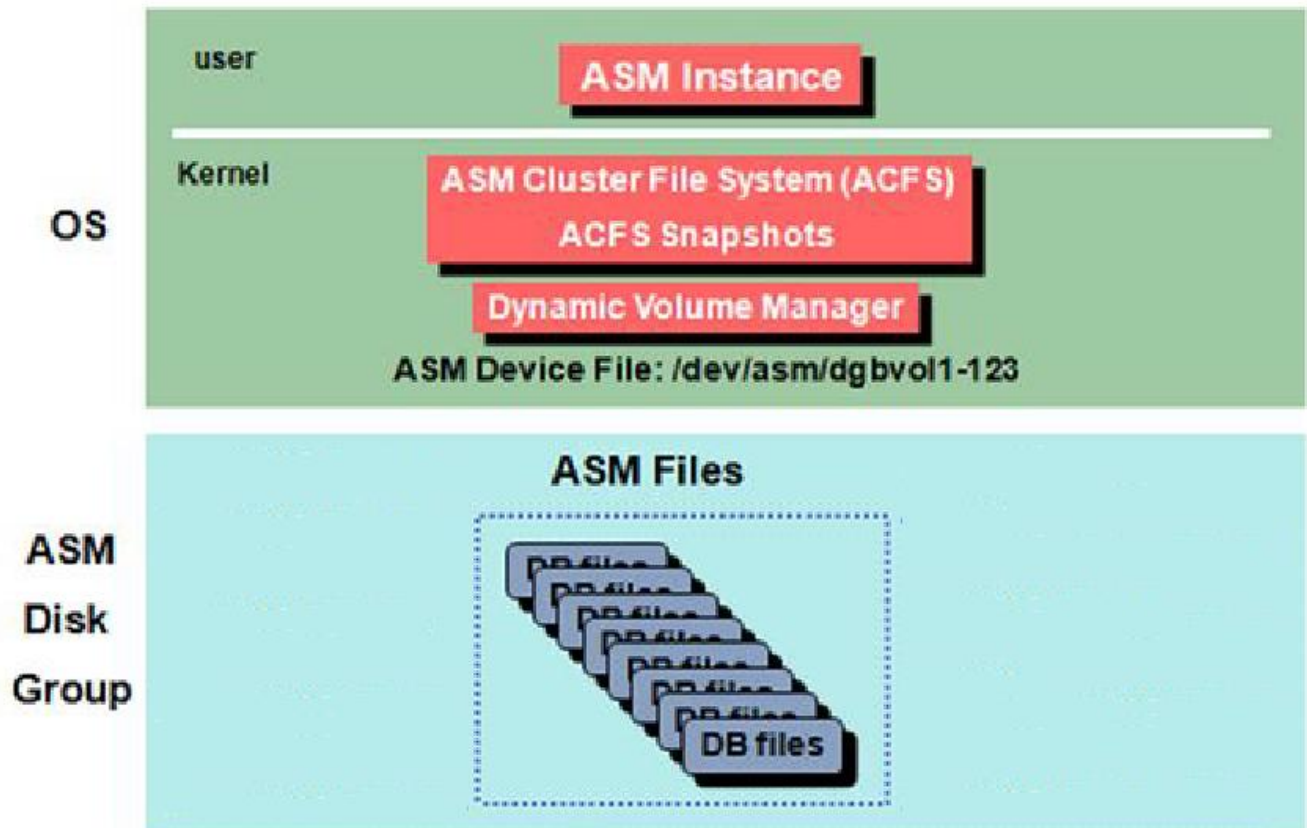
# DBMS Install Document

---

## Server Information

Items	Description
Test Date	2011 / 06 / 01
CPU	Intel(R) Xeon(R) CPU X5620 @ 2.40GHz X 8
Main Memory	1GB
O/S version	RHEL 5.5
Host Name	ATHENA
ORACLE_SID	ATHENA
Oracle version	11.2.0.2

# ADVM



11gR2 ASM에서 새롭게 나온 기능이며 해당 기능 하나로 벤더 혹은 3rd party 볼륨 제품군을 사용하지 않아도 될 만큼 훌륭한 기능이다.

ADVM 과 ACFS 를 사용하기 이전에 ADVM 이라는 의미를 이해 해야 한다.

ADVM 은 ASM Dynamic Volume Manager 의 약자이며 용어대로 ASM 의 볼륨을 관리하는 기능이다.

위의 그림이 ADVM 과 ACFS 를 설명해 주는 가장 손쉬운 그림이다.

ASM DISK GROUP 위에 ADVM 이 존재 하며 ADVM 을 통해 ASM Volume 생성 및 ACFS 기능을 이용할 수 있다.

ADVM 은 linux, aix, hp-ux 에서 사용중인 볼륨 서비스인 LVM 과 유사 하다고 볼 수 있다.

LVM 은 여러개의 Disk 를 Volume Group 으로 만든 후 logical volume 을 생성한 후 Rawdevice 나 mount 하여 filesystem 영역에 사용하게 되는데 이와 유사 하다고 할 수 있다.

ASM 의 경우 여러 개의 Disk 를 하나의 ASM Disk Group 으로 생성 하여 사용하게 된다.

10g,11gR1 까지는 이렇게 생성된 ASM Disk Group 내에 datafile 을 위치 시켜 사용하였다.

11gR2 는 여기서 좀더 진보하여 LVM 과 유사하게 ASM Disk Group 에서 volume 을 생성 할 수 있다.

해당 Volume 은 rawdevice 로 사용가능하며, mount 하여 OS 의 filesystem 영역으로 사용가능하며, ADVM 에서 제공하는 ACFS 를 이용하여 Cluster file system 을 사용할 수 있게 된다.

볼륨은 오라클 데이터베이스의 데이터 파일이 아닌 OS 의 파일시스템의 볼륨으로 표현된다.

ADVM 과 관련 디바이스 드라이버는 동적 볼륨과 ASM 클러스터 파일 시스템 간에 인터페이스를 제공한다.

ASM 동적 볼륨에는 NTFS 와 같은 다른 파일 시스템으로도 운영 될 수 있다.

ASM 의 장점 중에 하나인 ASM Disk GROUP 내에 존재하는 Disk 에 아주 균등하게 Striping 되어 ASM AU 가 분배 되어

어느 Storage 기능보다 I/O 성능을 보장 받을 수 있다는 것은 익히 알려진 내용이며,

ACFS 를 사용하게 되면 Cluster file system 의 편의성과 ASM 의 고성능 I/O 성능을 사용할 수 있는 기술이라고 할 수 있다.

### 1. sysasm => sysdba

11gR2 에서 변경 된 것이 ASM 인스턴스 접속 관리자 권한이 sysdba 에서 sysasm 권한으로 변경 되었다는 것이다.

asm 인스턴스에 접속해서 히든파라미터를 조회 하면 11gR1 과 R2 가 다른 것을 확인 할 수 있다.

```
COL PARAM          FORMAT A60  HEADING "Parameter | Name" WRAP
COL DESCRIPTION    FORMAT A64  HEADING "Parameter | Description" WRAP
COL DFLT           FORMAT A7   HEADING "Default | Values"
COL ISSES_MODIFIABLE FORMAT A10 HEADING "IsSession | Modifiable"
COL ISSYS_MODIFIABLE FORMAT A10 HEADING "IsSystem | Modifiable"
```

```
SELECT RPAD(LKSPPINM, 35) || '=' || V.KSPSTVL AS PARAM
      , I.KSPDESC DESCRIPTION
      , DECODE(V.KSPSTDF, 'FALSE', 'F', 'TRUE', 'T') AS DFLT
      , P.ISSES_MODIFIABLE
      , P.ISSYS_MODIFIABLE
FROM X$KSPP I
      , X$KSPPCV V
      , V$PARAMETER P
WHERE V.INDX = I.INDX
      AND V.INST_ID = I.INST_ID
      AND I.KSPPINM = P.NAME(+)
      AND I.KSPPINM = '_asm_admin_with_sysdba'
ORDER BY I.KSPPINM;
```

위 쿼리의 결과에서 보면

11gR1

Name

-----  
\_asm\_admin\_with\_sysdba = TRUE

11gR2

Name

-----  
\_asm\_admin\_with\_sysdba = FALSE

해당 파라미터의 Description 에는 아래와 같이 ASM 의 관리자 권한을 sysdba 인지 여부를 결정하는 것을 알 수 있다.

Does the sysdba role have administrative privileges on ASM?

11gR1 까지는 true 로 되어 있으며 export ORACLE\_SID=+ASM 지정 후  
sqlplus "/as sysdba" 로 ASM 인스턴스 내로 접근할 수 있었지만 11gR2 부터는  
grid 유저로 sqlplus "/as sysasm" 으로 접속해야 한다.

그렇기 때문에 사용의 편의 성을 위해 sysdba 를 사용하기 위해  
11gR2 에서 파라미터 수정을 하겠다.

```
SQL> alter system set "_asm_admin_with_sysdba"=true scope=spfile;
```

db 종료 => asm 재시작 => db 구동 순으로 작업 하면 된다.

필수 사항은 아니며, 11gR2 부터는 sysasm 를 사용해야 한다는 것이 변경된  
점이다.

## 2. Volume Creation

---

ASM 인스턴스를 접속하여 볼륨을 생성한다.

아래쿼리로 조회해보면 사용중인 용량과 여유 용량을 알 수 있다.

```
set lines 400  
col name for a30  
select name,state,type,total_mb,free_mb from v$asm_diskgroup;
```

TOTAL_MB	FREE_MB
32496	26065

약 26000MB 가 여유 공간으로 조회되고 있다. 10000MB 의 볼륨을 할당 하겠다.

명령어는 아래와 같다  
alter diskgroup DATA add volume arch size 10000m;

DATA 는 ASM DISK GROUP 이름이고 arch 는 volume 명이다.

생성 후 OS 에서 /dev/asm 을 조회해보면 OS 디바이스가 생성된 것 을 확인할 수  
있다.

```
ls -l /dev/asm/  
total 0  
brwxrwx--- 1 root dba 252, 226817 Jun 1 23:24 arch-443
```

arch-443 으로 디바이스가 생성되었고 -443 은 디스크 그룹에 대한 고유번호이다.  
이 번호는 시스템에 의해 자동으로 할당된다.

쿼리로 조회해보면

```
set lines 500
col usage for a20
col volume_name for a20
col mountpath for a30
col volume_device for a30
col mountpath for a30
select
volume_name,size_mb,volume_number,state,usage,volume_device,mountpath
from v$asm_volume;
```

아래와 같은 내용이 조회할 수 있다.

VOLUME_NAME	SIZE_MB	VOLUME_NUMBER	STATE
ARCH	10240	1	ENABLED

USAGE	VOLUME_DEVICE	MOUNTPATH
	/dev/asm/arch-443	

### 3. Volume Resize

---

10000MB 로 생성된 arch 볼륨을 resize 시키려면 아래와 같다.

Increase

```
SQL> alter diskgroup DATA resize volume arch size 10005m;
```

Reduce

```
SQL> alter diskgroup DATA resize volume arch size 9500m;
```

### 4. Volume Delete

---

생성 된 볼륨을 삭제 하기 위해서 아래와 같다.

```
alter diskgroup DATA drop volume arch;
```

## 5. Volume Status Change

---

볼륨 활성화

```
SQL> alter diskgroup DATA enable volume arch;
```

=> 볼륨 생성 후 default 는 바로 enable 상태 임으로 활성화 할 필요는 없다.

볼륨 비활성화

```
SQL> alter diskgroup DATA disable volume arch;
```

---

## RAWDEVICE

### 1. RAWDEVICE

---

test 라는 볼륨을 하나 생성하여 rawdevice 로 사용가능한지를 테스트 해본다.

```
SQL> alter diskgroup DATA add volume test size 10m;
```

root 유저로 /dev/asm 으로 이동한 뒤 소유권을 변경한다.

```
# chown oracle:dba test-443
```

oracle 유저로 Database 로 접속하여 테이블스페이스를 생성한다.

```
SQL> create tablespace test datafile '/dev/asm/test-443' size 8m;
```

조회 해보면 정상적으로 Tablespace 가 생성되었음을 확인 할 수 있다

TABLESPACE_NAME	FILE_NAME
SYSTEM	+DATA/athena/datafile/system01.dbf
SYSAUX	+DATA/athena/datafile/sysaux01.dbf
UNDOTBS1	+DATA/athena/datafile/undotbs01.dbf
USERS	+DATA/athena/datafile/users01.dbf
TEST	/dev/asm/test-443



테이블 생성 하고 데이터를 insert 하고 조회 해보면

```
SQL> create table test
  2  (no number)
  3  tablespace test;
Table created.
```

```
SQL>insert into test
  2  values(1);
1 row created.
SQL> commit;
sel
Commit complete.
```

```
SQL> select * from test;
-----
      NO
-----
      1
```

ASM volume 으로 Rawdevice 로 사용가능 하다는 것을 확인 할 수 있다.

---

## Filesystem

### 1. Filesystem

---

위에서 설명했듯이 ASM 볼륨으로 OS 일반 파일시스템으로도 사용 가능하다.

백업을 위한 볼륨을 생성하고 mount 를 해서 사용하겠다.

```
SQL> alter diskgroup DATA add volume backup size 10000m;
```

root 유저로 mkfs 후 mount 를 실행한다.

```
mkfs -t ext3 /dev/asm/backup-443
```

```
mount /dev/asm/backup-443 /backup
```

```
[root@athena trace]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/os_vol-root_vol
                          39G  17G  21G  44% /
/dev/hda1                  289M   17M  257M   7% /boot
tmpfs                      2.0G  567M  1.5G  28% /dev/shm
/dev/asm/backup-443      9.9G  151M  9.2G   2% /backup
```

df -h 로 보면 정상적으로 /backup 에 mount 된 것을 확인 할 수 있다.

rman 으로 백업을 수행해서 /backup 파일시스템을 정상 사용가능한지 확인 하겠다.

```
rman> backup incremental level 0 database format '/backup/df_%d_%s_%p.bus'
```

```
[oracle@athena:/backup]$ ls -l
total 449324
-rw-r----- 1 oracle dba 20905984 Jun  2 00:19 df_ATHENA_2_1.bus
-rw-r----- 1 oracle dba 417832960 Jun  2 00:23 df_ATHENA_3_1.bus
-rw-r----- 1 oracle dba 20905984 Jun  2 00:23 df_ATHENA_4_1.bus
```

/backup 위치에 정상적으로 백업되는 것을 확인 할 수 있으며, 해당 파일시스템을 정상적으로 사용 가능하다는 것을 확인 할 수 있다

ASM 볼륨을 이용하여 mount 한 내역을 다음 부팅에서도 mount 되게 하기 위해서는 OS 의 fstab 등에 기록하지 않고 acfsutil 이라는 새로운 utility 를 이용한다.

registry 라는 옵션을 사용하며 커맨드는 아래와 같다.

```
acfsutil registry -a /dev/asm/backup-443 /backup
```

등록해 놓으면 grid 의 resource 기동 순서는 아래와 같이 된다.

```
ASM => DISKGROUP => Mount => Database start
```

## 1. ACFS

---

ACFS는 강인성, 일반 용도, 익스텐트 기반, 저널링 파일 시스템을 제공하여 ASM을 확장한다.

ACFS는 오라클 이진 파일(바이너리), 리포트 파일, 추적 파일(Trace), 정보 로그(alert log), 그 외 애플리케이션 데이터 파일 등을 지원한다.

ACFS가 추가됨으로서 ASM은 오라클 데이터베이스와 비-데이터베이스 파일들을 모두 지원하는 완전한 저장 장소 관리 솔루션이 된다.

ACFS는 소용량 파일에서 대용량(엑사바이트) 파일까지 확장되며, 클러스터 내의 수많은 노드들을 지원한다.

ACFS는 고성능을 제공하고, 파일시스템의 무결성 및 신속한 복구를 위한 로그 기반 메타데이터 트랜잭션 엔진을 사용하기 위한 익스텐트 기반 파일 시스템이다.

ACFS 디스크 구조는 엔디언(edian) 중립 메타데이터를 지원 한다.

ACFS 파일시스템은 NFS 및 CIFS와 같은 산업 표준 프로토콜을 이용하여 원격 클라이언트에게 제공 될 수도 있다.

ACFS는 단일 노드 뿐만 아니라, RAC와 같은 그리드 컴퓨팅 환경에서 모든 파일 타입 관리를 단순화하며, 써드 파티(Third Party) 클러스터 파일 시스템의 도입 및 구입의 필요성을 감소 시킨다.

ACFS는 데이터베이스를 정지하지 않고 동적 파일시스템의 확장 및 축소를 지원한다.

ACFS는 하드웨어 RAID 기능에 추가 하여 ASM 미러링 및 스트라이핑 기능을 사용할 수 있도록 함으로서 고가용성을 제공한다.

## 2. mkfs -t acfs

---

arch 이름으로 9000MB 용량으로 생성한다.

```
SQL> alter diskgroup data add volume arch size 9000m;
```

filesystem 사용하듯이 mkfs 작업을 실행한다.

```
# mkfs -t acfs /dev/asm/arch-443
mkfs.acfs: version          = 11.2.0.2.0
mkfs.acfs: on-disk version  = 39.0
mkfs.acfs: volume          = /dev/asm/arch-443
mkfs.acfs: volume size     = 9663676416
mkfs.acfs: Format complete.
```

mount 시에도 -t(type) 을 지정해준다.

```
# mount -t acfs /dev/asm/arch-443 /arch
```

df -h 로 조회 해보면 아래와 같이 mount 된 것을 확인 할 수 있다.

```
/dev/asm/arch-443  9.0G  55M  9.0G  1% /arch
```

acfsutil 의 info fs 옵션으로 확인해보며 아래와 같다

```
# acfsutil info fs
/arch
ACFS Version: 11.2.0.2.0
flags:      MountPoint,Available
mount time: Thu Jun  2 00:44:41 2011
volumes:    1
total size: 9663676416
total free: 9606504448
primary volume: /dev/asm/arch-443
label:
flags:      Primary,Available,ADVM
on-disk version: 39.0
allocation unit: 4096
major, minor: 252, 226818
size:       9663676416
free:       9606504448
ADVM diskgroup DATA
ADVM resize increment: 268435456
ADVM redundancy: unprotected
ADVM stripe columns: 4
ADVM stripe width: 131072
number of snapshots: 0
snapshot space usage: 0
```

ASM 인스턴스에서 조회하면 USAGE 에는 ACFS 로 MOUNTPATH /arch 로 볼륨 정보 및 ACFS 에 대한 조회가 가능하다.

USAGE	VOLUME_DEVICE	MOUNTPATH
ACFS	/dev/asm/test-443 /dev/asm/arch-443	/arch

위에서 언급 했듯이 mount 정보를 acfsutil 을 이용하여 mount 정보를 등록한다.

```
# acfsutil registry -a /dev/asm/arch-443 /arch  
acfsutil registry: mount point /arch successfully added to Oracle Registry
```

### 3. ACFS Resize

---

위에서 벤더 및 3<sup>rd</sup> party 볼륨 제품군을 사용하지 않아도 될 만큼 고가용성을 느낄 수 있는 부분이다.

HP-UX 기준으로 LVM Version1 을 사용하기 때문에 online 중에 볼륨 추가 및 resize 가 되지 않는다고 한다.

Cluster 일 경우 한쪽노드의 LVM 의 변경사항을 export 한후 import 해줘야하는 번거러움도 존재한다.

Linux 기준으로 LVM 사용시 Volume resize 후 Filesystem 에서 따로 resize2fs 작업이 동반되지만 ACFS 는 명령어 한줄로 아주 빠르게 Online Resize 가 가능하다. 물론 파일시스템 umount 하지 않아도 된다.

ASM DISK GROUP 의 여유 공간이 있다면 해당공간 만큼 증가 시킬 수 있다

ASM 의 장점 중 한가지는 ASM DISK GROUP 에 신규 DISK 를 추가시 Online 추가 및 Rebalance 기능이다.

정리 하면 ASM 을 사용하면서 용량 부족에 대한 부분 Downtime 없이 가능하다는 것이다.

ASM DISK GROUP 에서 새로운 LUN / DISK 를 Online 으로 추가하여 DISK GROUP 의 용량을 증가 시킨 후 ASM Volume 증가 순서로 작업하게 되면 Downtime 없는 고가용성 용량 추가가 가능하다.

위에서 asm volume 의 resize 는 ASM 인스턴스 내에서 아래 커맨드 형태로 진행 하였다.

```
SQL> alter diskgroup DATA resize volume arch 10000m;
```

하지만 볼륨이 ACFS 로 적용 되었다면 acfsutil 로 resize 를 진행한다.  
acfsutil 로 resize 를 하게 되면 volume 도 resize 되면서 Filesystem 도  
resize 가 적용된다.

현재 9000MB 로 되어있는 arch 볼륨을 15000MB 으로 늘려 보겠다.  
root 로 작업하며, 커맨드는 아래와 같고 linux 의 time 명령어로 시간을  
측정 해보았다.

```
# time acfsutil size 15000m /arch
acfsutil size: new file system size: 15837691904 (15104MB)

real  0m0.060s
user  0m0.009s
sys   0m0.014s
```

9000MB 에서 15000MB 로 변경 되었으며 6000MB 로 크지 않은 용량 이지만  
1 초 미만의 빠른 처리를 보여주고 있다. df -h 로 확인해보면 filesystem 이  
15000MB 로 증가 된 것을 확인 할 수 있다

```
/dev/asm/arch-443 15G 67M 15G 1% /arch
```

ASM 인스턴스 내에서 ASM Volume 을 조회하면 아래와 같이 ARCH  
volume 사이즈도 같이 늘어난 것을 확인 할 수 있다.

VOLUME_NAME	SIZE_MB	VOLUME_NUMBER	STATE
TEST	256	1	ENABLED
ARCH	15104	2	ENABLED

---

## Conclusion

### 1. acfstuil

---

리눅스/유닉스 및 윈도우에서 지원되지 않는 새로운 ACFS 명령어는 다음과 같다.

acfsutil info fs - Display ACFS file and file system features and information

acfsutil snapshot - create and display ACFS snapshots

acfsutil registry - Registry an ACFS file system with the ACFS mount registry

마운트 포인트를 등록하는 옵션으로써 다음 부팅때 clusterware 가 올라오면  
마운트가 된다. RAC 의 경우 한쪽노드에서 실행하면 된다.

acfsutil registry -a /dev/asm/backup-443 /backup - mount 정보 등록

acfsutil registry -d /backup - mount 정보 삭제

acfsutil rmfs - Remove unmounted ACFS file system

acfsutil size - resize an ACFS file system

# acfsutil size 20G /backup/arch/

acfsutil 명령어를 실행시키면 더 많은 옵션에 사용에 대한 정보를 볼 수 있다

### 2. only MS-Windows Command

---

아래의 명령어가 윈도우 파일 관리 API 가 ACFS 파일 시스템에 통합되었으며  
다음 명령들이 관리 명령에 추가 되었으며, 윈도우 OS 에서만 가능하다.

- acfschkdsk

- acfsformat

- acfsmountvol

- acfsdiskmount

- ACFS 파일 시스템 생성

acfsformat [/vf] [/b blkksz] [/n name] device [blocks]

- ACFS 파일 시스템 마운트

```
acfsmountvol [/all] [/v]
```

- ACFS 파일 시스템 언마운트

```
acfsdismount [/v] [/p] [/P] path
```

- ACFS 파일 시스템 체크 및 수리

```
acfschkdsk [/a] [/v] [/n] [/f] [info] device
```

### 3. Overall rating

---

모 사이트에서 solaris 에서 Veritas Volume manager 를 사용하는 것을 본적이 있다. LINUX/UNIX 중 유일하게 solaris 만 LVM 이 존재 하지 않고 SVM 을 이용하여 볼륨을 관리하는데 해당 담당자는 SVM 이 불편하여 Veritas 제품을 사용한다고 말한적이 있었다.

위에서 테스트 해봤듯이 ASM 볼륨은 어느 볼륨 시스템 보다 쉽고 간편하게 사용할 수 있는 장점이 있어서 OS 볼륨 관리를 위해서도 충분히 사용해볼 만 하다고 생각되며 grid infrastructure + ASM 이 생성 되어있다면 바로 볼륨서비스 및 ACFS 를 사용가능하기에 가격 면에서도 3rd Party 보다 저렴할 듯도 하다.

Test 해본 결과 상당히 좋은 기능임을 알 수 있으며, 현재 테스트 환경에서는 datafile 이 ASM 영역내에 존재 하지만 db 생성 전 ASM 볼륨 +ACFS 를 준비 해놓는다면 cluster file system 영역에 바로 DB 를 생성 할 수 있다.

이렇게 구성이 된다면 ASM 을 사용하지만 ASM 내에 존재 하는 것은

OCR / VOTE 만이 ASM 내에 존재 하게 되므로 11gR2 RAC 에서도 datafile 에 대한 관리가 쉬워진다고 볼 수 있다.