

# Oracle ASM-Cluster-File-System: Das lang erwartete fehlende Feature

Matthias Pölzinger, GNC Global Network Communication

**Mit 11g R2 wurde Oracle ASM um eine wesentliche Komponente erweitert, das ASM-Cluster-File-System (ACFS). Dieser Artikel dient als Einführung in diese lang erwartete Erweiterung und beschreibt deren Funktionalität sowie die möglichen Einsatzgebiete und erläutert die Konfiguration anhand eines Beispiels, basierend auf dem zum Redaktionsschluss vorhandenen Beta-Release.**

Seit der Veröffentlichung von Oracle Parallel Server in der Version 6.2 im Jahr 1989 gab es verschiedenste Lösungsvarianten, um Datenbank-Dateien zentral zu speichern, damit die im Cluster befindlichen Knoten Zugriff darauf haben. Anfangs war dies auf die Verwendung von RMS unter VMS beziehungsweise Raw Devices unter Unix-Systemen eingeschränkt. Mit der Weiterentwicklung von Parallel Server und den Betriebssystemen wurden von Oracle weitere Cluster-File-Systeme zertifiziert, insbesondere ab der Version 9i.

Allerdings existierte zum damaligen Zeitpunkt vor allem im Linux-Umfeld nur eine Unterstützung von Raw Devices, die zwar optimale Performance boten, jedoch zahlreiche Einschränkungen aufwiesen. In erster Linie wird für jedes Data File, Redo Log und Control File ein separates Raw Device benötigt. Zusätzlich gilt per Default eine Einschränkung von 256 Raw Devices. Dies kann die Administration von stark wachsenden Datenbanken schnell vor ein Problem stellen. Zudem gilt für Datenbanken mit Raw Devices ein erhöhter beziehungsweise komplexerer Administrationsaufwand, da diese nicht wie Dateien verwaltet werden können.

Um dem entgegen zu wirken sowie die Verbreitung von Real-Application-Cluster-Installationen unter Linux voranzutreiben, initiierte Oracle das Projekt „Oracle Cluster File System“ (OCFS). Dieses stellte mit Veröffentlichung der Oracle Version 9i R2 im Jahr 2002 die erste Version von OCFS und somit das erste kostenlose Cluster-File-System von Oracle zur Verfügung. Unterstützt wurde hierbei die Ablage

von Oracle Datenbank-Dateien, die eine komfortablere und einfachere Administration von RAC-Installationen ermöglichte.

Es wurden jedoch bei der damaligen Veröffentlichung der ersten Version lediglich Oracle Datenbank-Dateien wie Data Files, Control File und Redo Logs unterstützt. Im Falle von Applikationsdateien, BFILES, Oracle Homes etc., die zwischen den Clusterknoten geteilt wurden, bestand weiterhin die Notwendigkeit eines Cluster-File-Systems von Drittherstellern. Ebenso fehlte eine Unterstützung für die Ablage von Archive-Log-Dateien. Diese fehlenden Funktionalitäten wurden jedoch mit dem Erscheinen von OCFS2 im Jahr 2005 behoben.

## Weiterentwicklung zu ASM

OCFS war beziehungsweise ist immer noch ein nur unter Windows und Linux unterstütztes File-System. In anderen Betriebssystem-Umgebungen war es stets erforderlich, das vom jeweiligen Betriebssystem-Hersteller zur Verfügung gestellte Cluster-File-System zu verwenden oder auf das eines Drittherstellers zurückzugreifen. Setzte ein Unternehmen auf mehr als ein Betriebssystem und wollte Raw Devices vermeiden, so war es in den meisten Fällen notwendig, mehr als ein Cluster-File-System zu verwenden und zu administrieren.

Dies erschwerte auch den Umstieg von einer Plattform auf eine andere, da hierzu das Personal entsprechend geschult werden musste und der Administrationsaufwand stieg. Zusätzlich sah Oracle zu wenig Entwicklungspotenzial

auf File-System-Ebene und entschloss sich, mit Automatic Storage Management (ASM) einen intelligenten, betriebssystemunabhängigen Volume/Storage-Manager zu entwickeln, der es ermöglichte, alle Oracle Datenbank-Dateien im OMF-Format abzulegen – unabhängig von Single-Node oder RAC.

## ASM Cluster File System – die fehlende Komponente

Seit der Veröffentlichung von Oracle ASM in 10g R1 kamen zahlreiche Erweiterungen hinzu. Die Unterstützung für Nicht-Oracle Datenbank-Dateien blieb jedoch bisher aus. So galt es für Real-Application-Cluster-Applikationen, die Daten via BFILE speicherten, Beladungsprozesse direkt auf den Clusterknoten durchführten oder ähnliche zentrale Datenspeicherungen erforderten, ein Cluster-File-System eines Drittherstellers oder OCFS zu verwenden beziehungsweise alternativ eine Anbindung via NFS durchzuführen.

Mit Veröffentlichung von 11g R2 wird diese Limitierung durch das ASM Cluster File System (ACFS) behoben. ACFS ist ein auf ASM basierendes Multiplattform Cluster File System, das es ermöglicht, alle Nicht-Oracle Datenbank-Dateien wie BFILES, Oracle Homes, Dumps etc. abzulegen.

## Prinzipieller Aufbau

Um die Ablage von Oracle Database-Binary-Installationen im ACFS zu ermöglichen, musste Oracle aufgrund des „Henne-Ei-Problems“ eine wesentliche Änderung im Aufbau von ASM vorneh-



men. Bisher wurden ASM-Instanzen stets aus einer Oracle Database-Binary-Installation betrieben. Dies wurde nun dahingehend geändert, dass die erforderlichen Binaries Bestandteil der Clusterware Oracle Homes sind beziehungsweise die Clusterware auch den Betrieb von ASM übernimmt. Diese Zusammenschließung wird als „Grid Infrastructure“ bezeichnet (siehe Abbildung 1).

Der Schritt ermöglichte es Oracle, nun einerseits die Oracle Cluster Registry (OCR) und Voting Devices innerhalb ASMs zu speichern, andererseits die Speicherung der Datenbank Oracle Homes in einem ASM-Cluster-File-System vorzunehmen.

Abbildung 1 zeigt auch eine weitere wesentliche Erweiterung, den „ASM Dynamic Volume Manager“ (ADVM). Dieser ermöglicht es, Volumes in einer ASM-Disk-Group anzulegen und dem Betriebssystem als physische Devices zur Verfügung zu stellen. Die angelegten Devices dienen anschließend als Basis für das ASM-Cluster-File-System oder können ebenso für ein Dritthersteller-Dateisystem verwendet werden. Hierbei gilt es jedoch zu beachten, dass bei Nichtverwendung eines Cluster-File-Systems der Mount des Dateisystems nur auf einem Clusterknoten erfolgen darf.

Um Zugriff auf die entsprechenden ADVM-Devices zu ermöglichen beziehungsweise das ASM-Cluster-File-System verwenden zu können, wurden von Oracle entsprechende Kernel-Module/Treiber entwickelt. Diese werden in Abhängigkeit von der Clusterware-Ressource „ora.asm“ automatisch nach einem Neustart geladen und können beispielsweise auf Linux-Systemen mit „lsmod“ gelistet werden (siehe Listing 1).

```
[root@beta1 ~]# lsmod | egrep
„oracle|Module“
Module                Size
Used by
oraacleacfs           781604 2
oraacleadv            212736 6
oraacleoks            224864 2
oraacleacfs,oraacleadv
oraacleasm            46100
```

Listing 1: Auflisten der ASM-Kernel-Module unter Linux

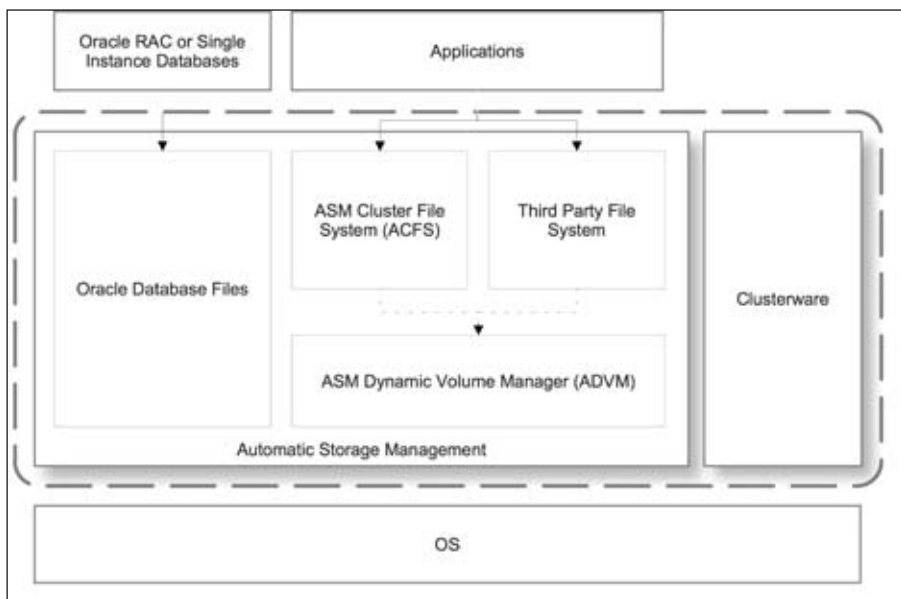


Abbildung 1: Übersicht ASM / ADVM / ACFS

### Die Vorteile

Die Entwicklung des ASM-Cluster-File-Systems ermöglicht es Applikationen, externe Daten zentral ablegen zu können, ohne auf eine der bisher zusätzlich benötigten Komponenten wie NFS, OCFS oder auf das Cluster-File-System eines Drittherstellers zurückgreifen zu müssen. Diese können nun über das ASM-Cluster-File-System direkt in ASM gespeichert sein und dabei den vollen Funktionsumfang von ASM nutzen, wie das Rebalancen von Daten für eine bessere I/O-Verteilung, das Online-Hinzufügen oder -Entfernen einer Disk aus einer Disk-Group oder die Spiegelung über mehrere Fail-Groups.

Die Durchführung dieser Operationen wird aufgrund der internen Speicherung eines ADVM-Volumes als ASM-Datei ermöglicht. Damit wird auf den bereits bekannten Eigenschaften von Disk-Groups aufgesetzt, die dadurch für die Implementierung von ACFS-Dateisystemen genutzt werden können.

Aus administrativer Sicht ergeben sich wesentliche Vorteile. Basierend auf den bisherigen ASM-Administrationskenntnissen lassen sich nun auch ASM-Cluster-File-Systeme rasch implementieren. Zudem ergeben sich im Vergleich zu NFS wesentliche Performance-Vorteile. Unter Verwendung von ACFS kann hierbei auf ein Speichernetzwerk (SAN) mit höherer Zu-

griffsgeschwindigkeit zurückgegriffen werden.

Aus Management-Sicht ergibt sich noch ein weiterer wichtiger Vorteil – die Kosteneinsparung. Konnte man auf Grund der bisher benötigten Performance nicht auf NFS zurückgreifen und hatte daher zur Speicherung von externen Applikationsdaten ein Cluster-File-System eines Drittherstellers im Einsatz, so können diese Daten nun im ASM-Cluster-File-System abgelegt sein. Entsprechendes Einsparungspotenzial ergibt sich dadurch im Hinblick auf die Cluster-File-System-Lizenzen.

### Einrichtung eines ASM-Cluster-File-Systems

Ausgangsbasis für die Beispiel-Implementierung eines ASM-Cluster-File-Systems ist ein Real Application Cluster unter Linux, der aus drei Knoten besteht und die letztgültige Beta-Version von 11g R2 verwendet. Die Installation der Grid Infrastructure (Clusterware und ASM) wurde bereits durchgeführt und ASM zur Speicherung der Oracle Cluster Registry und Voting-Informationen ausgewählt. Hierbei wurde bereits im Rahmen des root.sh-Skripts auf jedem Knoten eine entsprechende ASM-Instanz konfiguriert.

Der Funktionsumfang der Administrationsutilities ist mit 11g R2 wesentlich erweitert. So wurde mittlerweile

ein eigener Configuration Assistant implementiert, der den Namen „asm-ca“ trägt. Mit diesem Assistenten kann nun die Konfiguration der ASM-relevanten Themen, analog zum „dbca“ für Datenbanken, über eine grafische Oberfläche durchgeführt werden (siehe Abbildung 2).

Darüber hinaus wurden das auf Command Line basierende „asmcmd“ um eine Vielzahl von Administrationsmöglichkeiten erweitert sowie entsprechende Kommandos für die Administration von ACFS und ADVM implementiert. Da in den durchgeführten Beta-Tests nicht immer eine reibungslose Administration mittels „asmca“ funktionierte, beziehungsweise um eine leichtere Nachvollziehbarkeit zu ermöglichen, wurden in den weiteren Schritten die Konfigurationen mittels Command Line durchgeführt.

Um nun ein ASM-Cluster-File-System einzurichten, wird im ersten Schritt eine entsprechende Disk Group benötigt. In diesem Szenario wird hierzu eine neue Daten Disk Group mit zwei Fibre Channel Disks angelegt. Listing 2 führt die entsprechenden Candidate Disks an.

Anschließend wird über ein neues asmcmd-Feature die entsprechende Disk Group mittels XML-File erstellt (siehe Listing 3).

Bei Durchsicht der Anlageattribute wird ersichtlich, dass Oracle mit „compatible.advm“ einen zusätzlichen Kompatibilitätsmodus definiert hat. Um ein entsprechendes ADVM-Volume anlegen zu können, muss zumindest die Kompatibilität für ASM und ADVM auf den Wert 11.2 gesetzt sein. In der nun entstandenen Disk Group „ACFS\_ODATA“ kann man anschließend ein neues ADVM-Volume ablegen. Listing 4 zeigt ein Beispiel dazu. Dieses definiert ein neues Volume namens „appdata1“ mit einer Größe von 4 GB und Default-Werten für das Striping.

Listing 4 schließt mit der Überprüfung des vorhandenen Devices. Im nächsten Schritt wird nun ein ASM-Cluster-File-System auf das frisch erstellte Volume/Device gelegt. Dies kann unter Linux analog wie bei anderen File-Systemen mit dem entspre-

```

ASMCMD> lsdsk -k -m c
Total_MB Free_MB OS_MB Name Failgroup Library
Label UDID Product Redund Path
0 0 5115 ASM Library - Generic
Linux, version 2.0.4 (KABI_V2) ACFS01 UNKNOWN
ORCL:ACFS01
0 0 5115 ASM Library - Generic
Linux, version 2.0.4 (KABI_V2) ACFS02 UNKNOWN
ORCL:ACFS02
ASMCMD> lsdsk -p -m c
Group_Num Disk_Num Incarn Mount_Stat Header_Stat Mode_Stat
State Path
0 0 3915942132 CLOSED PROVISIONED ONLINE
NORMAL ORCL:ACFS01
0 1 3915942133 CLOSED PROVISIONED ONLINE
NORMAL ORCL:ACFS02
ASMCMD>

```

Listing 2: Candidate Disks

```

[oracle@gbeta1 bin]$ more /home/oracle/create_ACFS_dg.asm
<dg name="ACFS_ODATA" redundancy="external">
  <dsk string="ORCL:ACFS01"/>
  <dsk string="ORCL:ACFS02"/>
  <a name="compatible.asm" value="11.2"/>
  <a name="compatible.rdbms" value="11.2"/>
  <a name="compatible.advm" value="11.2"/>
</dg>
[oracle@gbeta1 bin]$ ./asmcmd
ASMCMD> mkdg /home/oracle/create_ACFS_dg.asm
ASMCMD> lsdg
State Type Rebal Sector Block AU Total_MB Free_MB
Req_mir_free_MB Usable_file_MB Offline_disks Voting_files Name
MOUNTED EXTERN N 512 4096 1048576 10230 10176
0 10176 0 N ACFS_ODATA/
ASMCMD>

```

Listing 3: Anlage ACFS\_ODATA Disk Group

```

ASMCMD> volcreate -d ACFS_ODATA -s 4G appdata1
ASMCMD> volinfo -d ACFS_ODATA -a
Diskgroup Name: ACFS_ODATA

Volume Name: APPDATA1
Volume Device: /dev/asm/appdata1-436
State: ENABLED
Size (MB): 4096
Resize Unit (MB): 256
Redundancy: UNPROT
Stripe Columns: 4
Stripe Width (K): 128
Usage:
Mountpath:

ASMCMD> exit
[oracle@gbeta1 ~]$ ls -al /dev/asm/appdata1-436
brwxrwx--- 1 root asmadmin 252, 223234 Aug 30 00:46 /dev/asm/appdata1-436
[oracle@gbeta1 ~]$

```

Listing 4: Anlage des Volumes „appdata1“

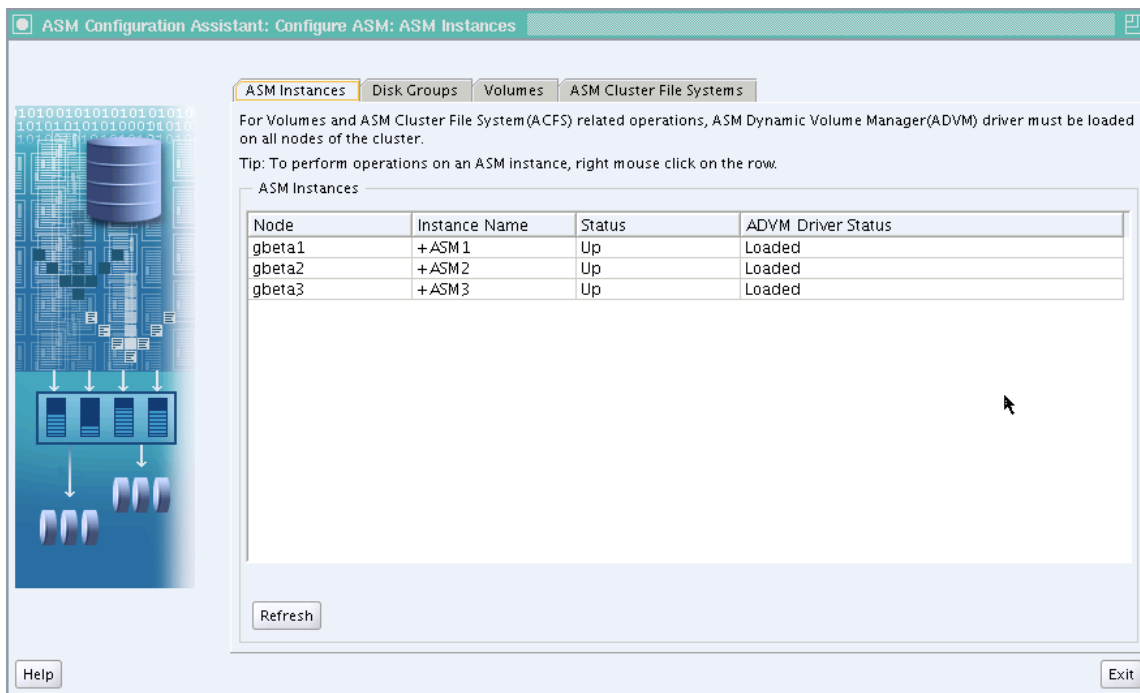


Abbildung 2: Übersicht ASM / ADVM / ACFS

chenden „mkfs“-Kommando erfolgen (siehe Listing 5).

Nach Erstellung des File-Systems erfolgt nun eine Eintragung in die Cluster Mount Registry. Dies bewirkt, dass das erstellte File-System vom Service „ora.registry.acfs“ alle dreißig Sekunden überprüft und gegebenenfalls automatisch gemountet wird. Ebenso erfolgt dieser automatische Mount-Vorgang nach dem Neustart der Oracle Clusterware.

Alternativ kann man mittels „srvctl“ ein ASM-Cluster-File-System als eigenständige Ressource hinzufügen. Dies ist zwingend erforderlich, wenn im erstellten File-System später ein Shared Oracle Home abgelegt werden soll, um eine Abhängigkeit zur Datenbank-Instanz abbilden zu können.

In diesem Beispiel ist jedoch die Anlage eines File-Systems für Applikationsdaten beschrieben. Listing 6 beinhaltet daher die erforderlichen Schritte für eine Eintragung in die Cluster Mount Registry.

Nach Abschluss der Eintragung erfolgt automatisch ein Mounten des File-Systems auf allen Cluster-Knoten. Um nicht die Dauer dieses automatischen Mounts abwarten zu müssen, kann man auch als User „root“ einen manuellen Mount auf den restlichen

```
[oracle@gbeta1 ~]$ /sbin/mkfs.acfs /dev/asm/appdata1-436
mkfs.acfs: version                = 11.2.0.0.2.0
mkfs.acfs: on-disk version        = 39.0
mkfs.acfs: volume                 = /dev/asm/appdata1-436
mkfs.acfs: volume size            = 4294967296
mkfs.acfs: Format complete.
[oracle@gbeta1 ~]$
```

Listing 5: Formatierung des Devices mit dem ASM-Cluster- File-System

```
[oracle@gbeta1 ~]$ mkdir /u01/app/oracle/acfsmounts/appdata1
[oracle@gbeta1 ~]$ ssh gbeta2 „mkdir /u01/app/oracle/acfsmounts/ap-
pdata1“
[oracle@gbeta1 ~]$ ssh gbeta3 „mkdir /u01/app/oracle/acfsmounts/ap-
pdata1“
[oracle@gbeta1 ~]$
[oracle@gbeta1 ~]$ /sbin/acfsutil registry -a /dev/asm/appdata1-436 /
u01/app/oracle/acfsmounts/appdata1
acfsutil registry: mount point /u01/app/oracle/acfsmounts/appdata1
successfully added to Oracle Registry
[oracle@gbeta1 ~]$ /sbin/acfsutil registry
Mount Object:
  Device: /dev/asm/appdata1-436
  Mount Point: /u01/app/oracle/acfsmounts/appdata1
  Disk Group: ACFS_ODATA
  Volume: APPDATA1
  Options: none
  Nodes: all
[oracle@gbeta1 ~]$ df -h /u01/app/oracle/acfsmounts/appdata1
Filesystem                Size  Used Avail Use% Mounted on
/dev/asm/appdata1-436
                          4.0G   45M  4.0G   2% /u01/app/oracle/acfs-
mounts/appdata1
[oracle@gbeta1 ~]$
```

Listing 6: Hinzufügen eines ACFS in die Cluster Mount Registry

Knoten durchführen. Listing 7 zeigt hierbei den Mount-Vorgang auf einem Cluster-Knoten.

Als letzter Schritt ist nun nur noch eine Anpassung der Berechtigungen erforderlich. Anschließend wird der Zugriff auf allen Cluster-Knoten überprüft (siehe Listing 8).

### Snapshot-Funktionalität

Wie bei vielen Storage-Lösungen üblich, wurde auch für das ASM-Cluster-File-System eine entsprechende Snapshot-Funktionalität implementiert. Oracle verwendet dafür den Copy-On-Write-Mechanismus, der vor der Änderung eines Extents eine entsprechende Kopie erstellt, um den Point-In-Time-Zustand des Snapshots zu gewährleisten.

Snapshots sind unmittelbar verfügbar und können wie gewohnt online durchgeführt werden. Derzeit sind diese auf 63 Snapshots pro File-System begrenzt und werden mittels „acfsutil“ erstellt beziehungsweise wieder entfernt (siehe Listing 9).

Listing 9 zeigt die Erstellung eines Snapshots namens „test\_snap“. Zum Zeitpunkt der Erstellung dieses Artikels konnten im Linux Beta-Release 2 keine Detaildaten zu Snapshots mittels „acfsutil“ gelistet werden. Es ist lediglich eine Übersicht über den Platzverbrauch verfügbar. Um die vorhandenen Snapshots zu listen, musste entsprechend das .ACFS/snaps-Verzeichnis im ACFS Mountpoint überprüft werden.

Anschließend wurde der Inhalt der Testdatei verändert, um die Funktionsweise des Snapshots zu überprüfen. Als letzter Schritt hat man den erstellten Snapshot wieder entfernt. Hierbei gilt es zu beachten, dass derzeit keine Möglichkeit für eine automatische Löschung vorhanden ist. Somit sind der Platzverbrauch entsprechend zu überwachen und Snapshots in regelmäßigen Abständen wieder zu entfernen.

### Fazit

Basierend auf vorhandenen ASM-Administrationskenntnissen lässt sich in einfachen Schritten das von Oracle nun präferierte Cluster-File-System

```
[root@gbeta2 ~]# mount -t acfs /dev/asm/appdata1-436 /u01/app/oracle/acfsmounts/appdata1
[root@gbeta2 ~]# df -h /u01/app/oracle/acfsmounts/appdata1
Filesystem                Size Used Avail Use% Mounted on
/dev/asm/appdata1-436    4.0G 109M  3.9G   3% /u01/app/oracle/acfs-
mounts/appdata1
[root@gbeta2 ~]#
```

Listing 7: Mounten des File-Systems auf einem weiteren Knoten

```
[root@gbeta1 ~]# ls -al /u01/app/oracle/acfsmounts/
total 16
drwxr-xr-x 4 oracle oinstall 4096 Aug 30 01:04 .
drwxr-xr-x 8 oracle oinstall 4096 Aug 18 21:29 ..
drwxrwx--- 4 root   asmadmin 4096 Aug 30 01:21 appdata1
[root@gbeta1 ~]# chown oracle:asmadmin /u01/app/oracle/acfsmounts/
appdata1
[root@gbeta1 ~]# ls -al /u01/app/oracle/acfsmounts/
total 16
drwxr-xr-x 4 oracle oinstall 4096 Aug 30 01:04 .
drwxr-xr-x 8 oracle oinstall 4096 Aug 18 21:29 ..
drwxrwx--- 4 oracle asmadmin 4096 Aug 30 01:21 appdata1
[root@gbeta1 ~]#

[oracle@gbeta1 appdata1]$ echo „Testinhalt“ > /u01/app/oracle/acfs-
mounts/appdata1/testacfs.txt
[oracle@gbeta1 appdata1]$ more /u01/app/oracle/acfsmounts/appdata1/
testacfs.txt
Testinhalt
[oracle@gbeta1 appdata1]$

[oracle@gbeta2 ~]$ more /u01/app/oracle/acfsmounts/appdata1/testacfs.
txt
Testinhalt
[oracle@gbeta2 ~]$

[oracle@gbeta3 ~]$ more /u01/app/oracle/acfsmounts/appdata1/testacfs.
txt
Testinhalt
[oracle@gbeta3 ~]$
```

Listing 8: Anpassung der Berechtigungen und Testen des Zugriffs auf allen Cluster-Knoten

```
[oracle@gbeta1 ~]$ /sbin/acfsutil snap create test_snap /u01/app/
oracle/acfsmounts/appdata1
acfsutil snap create: Snapshot operation is complete.
[oracle@gbeta1 ~]$
[oracle@gbeta1 ~]$ /sbin/acfsutil info fs /u01/app/oracle/acfsmounts/
appdata1
/u01/app/oracle/acfsmounts/appdata1
ACFS Version: 11.2.0.0.2.0
flags:          MountPoint,Available
mount time:     Sun Aug 30 01:05:40 2009
volumes:       1
total size:     4294967296
total free:     4114432000
primary volume: /dev/asm/appdata1-436
label:
flags:          Primary,Available,ADVM
```



```

on-disk version:      39.0
allocation unit:     4096
major, minor:       252, 223234
size:               4294967296
free:              4114432000
ADVM diskgroup      ACFS_ODATA
ADVM resize increment: 268435456
ADVM redundancy:    1
ADVM stripe columns: 4
ADVM stripe width:  131072
number of snapshots: 1
snapshot space usage: 65536
[oracle@gbeta1 ~]$
[oracle@gbeta1 ~]$ ls -al /u01/app/oracle/acfs-
mounts/appdata1/.ACFS/snaps/
total 12
drwxr-xr-x 3 root   root   4096 Aug 30 12:17 .
drwxr-xr-x 5 root   root   4096 Aug 30 01:21 ..
drwxrwx--- 4 oracle asmadmin 4096 Aug 30 12:17
test_snap
[oracle@gbeta1 ~]$
[oracle@gbeta1 ~]$ echo „Geaendert“ > /u01/app/
oracle/acfsmounts/appdata1/testacfs.txt
[oracle@gbeta1 ~]$ more /u01/app/oracle/acfs-
mounts/appdata1/testacfs.txt
Geaendert
[oracle@gbeta1 ~]$ more /u01/app/oracle/acfs-
mounts/appdata1/.ACFS/snaps/test_snap/testacfs.
txt
Testinhalt
[oracle@gbeta1 ~]$
[oracle@gbeta1 ~]$ /sbin/acfsutil snap delete
test_snap /u01/app/oracle/acfsmounts/appdata1
acfsutil snap delete: Snapshot operation is
complete.
[oracle@gbeta1 ~]$ /sbin/acfsutil info fs /u01/
app/oracle/acfsmounts/appdata1
/u01/app/oracle/acfsmounts/appdata1
ACFS Version: 11.2.0.0.2.0
flags:          MountPoint,Available
mount time:     Sun Aug 30 01:05:40 2009
volumes:       1
total size:    4294967296
total free:    4114505728
primary volume: /dev/asm/appdata1-436
label:
flags:
Primary,Available,ADVM
on-disk version:      39.0
allocation unit:     4096
major, minor:       252, 223234
size:               4294967296
free:              4114505728
ADVM diskgroup      ACFS_ODATA
ADVM resize increment: 268435456
ADVM redundancy:    1
ADVM stripe columns: 4
ADVM stripe width:  131072
number of snapshots: 0
snapshot space usage: 0
[oracle@gbeta1 ~]$

```

www.database-consult.de

**Kompetenz  
Training  
Consulting  
Entwicklung**

Telefon 08043 1010 **ORACLE PARTNER**

ACFS für Nicht-Oracle Datenbank-Dateien implementieren. Im Gegensatz zu OCFS als Multi-Plattform-File-System konzipiert, können zukünftig Cluster-File-Systeme unabhängig vom Betriebssystem „ident“ administriert werden. Zusätzlich ergibt sich bei derzeitiger Verwendung von Cluster-File-Systemen eines Drittherstellers bei der Ablage von Applikationsdaten auch Kosteneinsparungspotenzial.

Administratoren, welche die Verwendung eines Shared Oracle Homes im Real-Application-Cluster-Umfeld präferieren, erhalten nun die Möglichkeit, diese über ACFS in ASM abzulegen, ohne hierfür auf NFS oder andere Cluster-File-System-Komponenten zurückgreifen zu müssen.

Anhand der vorhandenen Beta-Version gilt es am Ende nur zu hoffen, dass eine Unterstützung von ACFS bereits mit Erscheinen von 11g R2 auf allen Plattformen erfolgt, um plattformabhängige Einschränkungen zu vermeiden.

**Kontakt:**

Matthias Pölzinger  
mp@gnc.at

Listing 9: Nutzung von Snapshots