ACFS Replikation à la 12.2

Mathias Zarick
Trivadis Delphi GmbH
Wien

Schlüsselworte
Grid Infrastructure, ASM, ACFS, Replikation, Hochverfügbarkeit

Einleitung
Eine Replikation für das ASM Cluster Filesystem wurde bereits mit der Version 11.2.0.2 eingeführt. Oracle Database Version 12.2 bringt hier nun wesentliche Neuerungen in der Architektur der ACFS Replikation mit. Dieser Vortrag beleuchtet diesen Wechsel, sowie das Setup und Betriebsaspekte. Mögliche Use Cases für den Einsatz werden diskutiert.

ACFS
Ein ASM cluster file system, kurz ACFS gibt es seit 11.2. Mit ACFS ist es möglich geworden block devices im ASM anzulegen, auf welche dann Filesysteme angelegt werden können und clusterweit als shared Filesystem zu mounten. Eine Replikation für so ein Filesystem gibt es seit 11.2.0.2, dieser Vortrag zeigt die bisherige und die neue Architektur, welche mir Version 12.2 eingeführt wurde. Von 2009 bis 2014 existierte ein spezielles Produkt „Oracle Cloud File System“, welches je nach Art der Verwendung von ACFS zu lizensieren war. Dieses Produkt ist nun jedoch Geschichte, derzeit gelten folgende Lizenzregeln:

- Wenn ACFS für beliebige Files aber keine Oracle DB-Files eingesetzt wird ist es kostenfrei
- Wenn ACFS für DB-Files eingesetzt wird, so ist natürlich die Datenbank zu lizensieren, falls hier auch noch Snapshots verwendet werden, so ist dies nur mit Hilfe der Oracle Enterprise Edition zulässig
- Die Features ACFS auditing, encryption, replication, security sind für die Benutzung zusammen mit Oracle DB-Files nicht supported
- Support für ACFS wird nur dann zur Verfügung gestellt, wenn auf dem jeweilgen Server zumindest ein Oracle Produkt und Support-Vertrag steht, das kann die Oracle Database aber auch Oracle Linux oder Oracle Solaris sein

ACFS Replication
Durch ein Setup von ACFS Replication kann man ein Filesystem, durch ein Standby Filesystem absichern. Die Quelle heißt dann Primary-Filesystem. Jegliche Änderungen (bis auf bewusst gewählte Ausnahmen) auf der Primärseite werden auf die Standby-Seite übertragen. Bis 12.1 wurde die Änderungen im versteckten Unterverzeichnis <mount point>/.ACFS/repl zwischen gespeichert und dann mittels dbms_backup_restore.networkFileTransfer zwischen zwei ASM Instanzen ausgetauscht. Diese Art der Replikation wird durch CRS-background Prozesse verwaltet und überwacht, sie funktioniert nur zwischen echten Clustern und nicht zwischen Oracle Restart Installationen. Um diese Limitierung zu umgehen, kann man One-Node-Cluster-Installationen verwenden.

Grid Infrastructure 12.2
12.2 ist die aktuelle Version der Oracle Datenbank, mit ihr kam natürlich auch eine neue Version der Grid Infrastructure, welche für ACFS zuständig ist. Und es kamen fundamentale Änderungen für die ACFS Replikation. Allgemeines zur neuen Grid Infrastructure Version: Vor der Installation eines Clusters sollte man eine neues OS-Gruppe anlegen (typischerweise racdba), welche verwendet wird,

um den oraagent über das OS mit dem neuen administrativen Recht SYSRAC auszustatten. Dieses Recht ist wesentlich restriktiver als das bisher verwendete SYSDBA Privilege und implementiert auf diese Weise ein für die Security wichtiges Least-Privileges-Prinzip. Mit 12.2 wird die mit Version 12.1 eingeführte Flex-ASM Architektur verbindlich, selbst für Cluster auf NFS, welche bisher gar kein ASM brauchten. Die Installation ist jetzt image-basiert, d.h. man entpackt das fertige Grid Home und startet dann einen Konfigurationsprozess.

ACFS Replikationsarchitektur in 12.2
Die Replikation ist nach wie vor asynchron, aber sie basiert jetzt auf Snapshots, der Fähigkeit Unterschiede zwischen Snapshots als sogenannte Snapshot Duplication Streams zu erfassen und applizieren und SSH. Selbst auf Windows kommt dafür jetzt SSH zum Einsatz und Oracle erklärt in der Dokumentation wie Cygwin/SSH installiert und konfiguriert werden soll, um es für ACFS Replikation nutzen zu können. Neuerdings ist ACFS Replikation aber auch zwischen verschiedenen Plattformen erlaubt. Erlaubt ist es beispielsweise Primary- und Standby-Filesystem zwischen Linux, Solaris und AIX zu betreiben. Windows jedoch kann hier nicht mit einbezogen werden, Replikationen von Windows sind nur zu einem Windows-System unterstützt. Nach wie vor wird nur ein Standby-Filesystem unterstützt, jedoch kann man mit einer manuellen Sync-Methode auch weitere Standby-Filesystem aufsetzen. Tags können immer noch verwendet werden, um die zu replizierenden Files auszuwählen, d.h. einen Filter in die Replikation einzubauen.
Die Replikation ist jetzt also Snapshot-basiert. Auf beiden Filesystem werden immer wieder Snapshots erzeugt, die Differenz der Daten zwischen 2 Snapshots wird nun als so genannter Snapshot Duplication Stream erfasst und per SSH auf die Standby Seite transportiert und dann dort appliziert. Dadurch schreitet der Stand der Daten auf der Standbyseite auf denselben Stand voran.

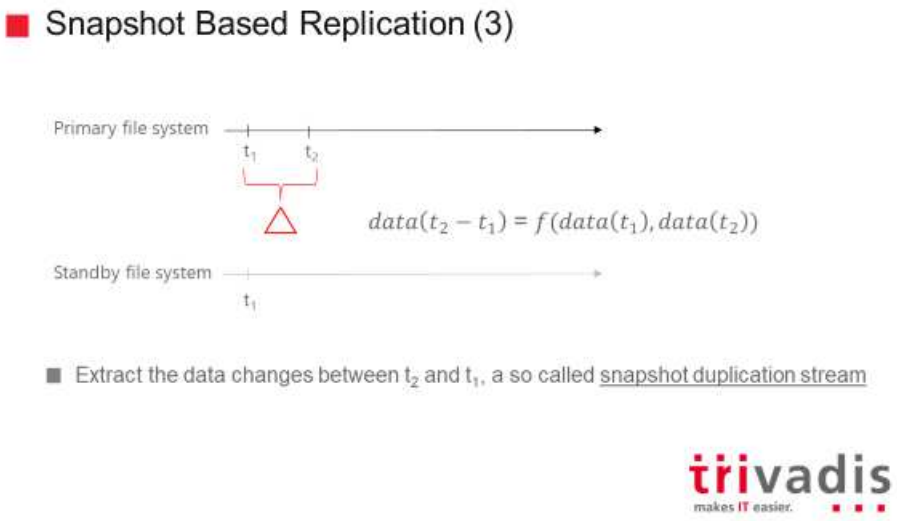

Abb. 1 - Snapshot Based Replication

Setup
Auf der Zielseite muss ein replication apply User (z.B. repluser) angelegt werden, dann müssen ssh-keys so ausgetauscht werden, dass es möglich wird sich vom Cluster mit dem Primary-Filesystem als root auf den repluser des Standby-Clusters anzumelden. Es wird empfohlen hier SCAN für die Kommunikation zu verwenden, und die SSH Host Keys innerhalb eines Clusters identisch zu halten, um eine Hochverfügbarkeit und eine funktionierende ACFS Replication auch nach Ausfall von Cluster

Nodes gewährleisten zu können. Dann wird mit folgenden Schritten als Grid Infrastructure Administrator ein ACFS und eine ACFS Replikation aufgebaut.
Auf Primary- und Standby-Cluster:

```
asmcmd volcreate -G ACFS -s 100G images_fs
sudo /sbin/mkfs -t acfs /dev/asm/images_fs-*
sudo srvctl add filesystem -device /dev/asm/images_fs-* \
  -path /u00/app/grid/acfsmounts/images_fs -user grid,oracle
srvctl start filesystem -device /dev/asm/images_fs-*
```

Jetzt muss das Filesystem auf beiden Seiten nur auf je einem Node gemounted sein, dann auf Standby-Seite:

```
sudo acfsutil repl init standby -u repluser \
  /u00/app/grid/acfsmounts/images_fs
```

Auf der Primary-Seite wird das Ganze jetzt abgeschlossen, dabei kann man verschiedene Parameter wählen:

- Refresh Interval (-i) oder Constant Mode (-C)
- Komprimierung oder keine Komprimierung (-z on oder off)
- Tags
- Debug Level (Default 3)
- Einige SSH Optionen

Beispiel für das Setup mit einem „image_data“ Tag, Komprimierung (-z) und constant mode (-C):

```
sudo acfsutil repl init primary -C -s repluser@cluster102 \
  -m /u00/app/grid/acfsmounts/images_fs -z on image_data \
  -o sshStrictKey=n /u00/app/grid/acfsmounts/images_fs
```

Beispiel für das Setup ohne Tag und 2 Stunden Refresh Intervall (-i):

```
sudo acfsutil repl init primary -i 2h -s repluser@cluster102 \
  -o sshStrictKey=n -m /u00/app/grid/acfsmounts/images_fs \
  /u00/app/grid/acfsmounts/images_fs
```

Jetzt kann das Filesystem auch auf allen anderen Nodes gemounted werden.

Was passiert während der Synchronisation?
Folgendermaßen sieht die Synchronisation aus, welche durch die Hintergrund-Prozesse immer wieder angestoßen werden. Dabei wird immer erst auf Primary-Seite ein Snapshot erzeugt, dann der Duplication Stream zum vorherigen Snapshot ermittelt, dieser wird übertragen und dann appliziert. In acfsutil-Kommandos sieht das Ganze so aus:

```
acfsutil snap create -r -R -S REPL_BASEFS_1501851997 \
/u00/app/grid/acfsmounts/images_fs
acfsutil snap dup create -R -i REPL_BASEFS_1501851476 \
REPL_BASEFS_1501851997 /u00/app/grid/acfsmounts/images_fs | \
ssh -o BatchMode=true -o StrictHostKeyChecking=no \
-o Ciphers=aes128-ctr -x repluser@cluster102 \
acfsutil snap dup apply -R -b '/u00/app/grid/acfsmounts/images_fs'
acfsutil snap delete -R -S REPL_BASEFS_1501851476
/u00/app/grid/acfsmounts/images_fs
```

Die “längeren” Zahlen in den Snapshot-Namen sind Unixtimes der jeweiligen Zeitpunkte. Ein Snapshot Duplication Stream wird mittels „acfsutil snap dup create“ erzeugt und direkt nach STDOUT gesendet. Hier wird der Stream jedoch direkt per pipe („|“) und durch SSH an die Empfängerseite

gesendet. „acfsutil snap dupp apply“ liest von STDIN und schreibt die Änderungen direkt in das Ziel-Filesystem.
Diese Kommandoabfolgen sind vollends dokumentiert und daher auch für manuelle Synchronisationen unterstützt, auf diese Art und Weise kann man hier auch mehr als nur ein Standby-Filesystem unterstützen oder ausgeklügelte Provisioning-Methoden für Applikations-Filesysteme umsetzen.

Use Cases?
Wo kann man diese Replikation nun sinnvoll einsetzen? Es gibt hier aus meiner Sicht vielseitige Verwendungsmöglichkeiten. Zum Beispiel kann man es aus Gründen der Hochverfügbarkeit für Oracle Datenbanken, welche exzessiv externe Files (z.B. BFILES) verwenden. Während die DB durch Data Guard abgesichert wird, sichert man das Filesystem mit ACFS Replikation ab. Mit ACFS Replikation kann auch für Applikations-Filesystem eine Redundanz zur Absicherung geschaffen werden. Ein weiterer Anwendungsfall wäre es GoldenGate Trail Files mit dieser Technologie abzusichern. Und zu guter Letzt sei noch die Möglichkeit erwähnt, Deployments und Cloning-Methoden für Applikations-Filesysteme mit effizienten Prozeduren der ACFS Replikation auszustatten, und das Ganze cross-plattform (außer Windows).

Kontaktadresse:

Mathias Zarick
Principal Consultant
Oracle Certified Master

Trivadis Delphi GmbH

Millennium Tower
Handelskai 94-96
A-1200 Wien

Tel.: +43 1 332 35 31 46
Fax: +43 1 332 35 34
Mobil: +43 664 854 42 95
mathias.zarick@trivadis.com
www.trivadis.com