

GI und ASM – Einsatz und Administration nicht nur für RAC

Timo Giese, Fiducia & GAD IT AG

Mit Einführung von ASM in der Version 10g bietet Oracle der Datenbank eine optimale und performante Anbindung an das zugrunde liegende Storage-System. Es ist für den Einsatz von Single Instance bis RAC konzipiert. Ursprünglich zugehörig zur Datenbank, ist es seit 11g Release 2 in der Grid Infrastructure verankert, die für RAC und RAC One Node das Cluster-Framework bildet. Dieser Artikel zeigt den Einsatz von ASM und Grid Infrastructure jenseits von RAC und RAC One Node und stellt Werkzeuge, Methoden sowie Best Practices vor, basierend auf der Ausprägung seit Oracle 11g Release 2.

Im RAC- und RAC-One-Node-Umfeld ist der Einsatz von Grid Infrastructure (GI) und ASM das Mittel der Wahl, an dem kein Weg vorbeiführt. Konzipiert sind ASM und GI von Oracle für den Einsatz bei allen Aufbauvarianten der Datenbank, zu denen auch Single Instance und Failover zählen. Dafür stellt Oracle GI und ASM auch für Nicht-RAC-basierte Datenbanken unter dem Namen „Grid Infrastructure Standalone“ (alternativ auch als „Oracle Restart“ bezeichnet) zur Verfügung. Es beinhaltet eine Ressourcen-Steuerung zum automatischen Restart der verwalteten Single-Instance-Datenbanken sowie ASM als Storage-Management-Layer, Volume Manager und Filesystem-Provider (ACFS).

ASM ist seit 11g Release 2 integraler Bestandteil der Grid Infrastructure und kann ohne diese nicht verwendet werden. Durch den Verzicht auf Filesysteme für die Ablage der Oracle-Datenbankfiles bietet ASM speziell für Oracle-Datenbanken eine optimale Ergänzung. ASM ist optimiert für eine effiziente und performante Anbindung von Storage (SAN/NAS/lokaler Storage).

Seit der Datenbank 12c Release 1 sind RAW-Devices „desupported“ und können nicht mehr direkt verwendet werden. Wer aufgrund der Performance noch immer auf Datenbanken mit RAW-Devices setzt, ist mit dem Wechsel zu ASM gut beraten. ASM ist ein Metadaten-Repository zur Verwaltung der Devices, die in weitestem Sinne immer noch RAW-Devices entspre-

chen, aber eine einfachere Verwaltung und Administration bieten.

Ein weiterer Vorteil ist ein einheitlicher Administrationslayer beim Einsatz des gleichen Technologiestacks wie in RAC-Umfeldern. Dies bedeutet, dass die gleichen Tools zur Administration von GI, ASM und Datenbanken auch für Single Instance zum Einsatz kommen. Im Datenbank-Umfeld mit RAC/RAC One Node und klassischen Single-Instance- oder Failover-Datenbanken wird DBAs, die bisher wenig mit RAC-Umfeldern zu tun hatten, die Fähigkeit gegeben, diese ohne große Probleme im täglichen Betrieb zu bedienen.

Die GI bietet zudem die Möglichkeit, weitere Oracle- und Nicht-Oracle-Applikationen einzubinden, etwa Golden Gate als Oracle-Applikation sowie ein Apache. Third-Party-Applikationen lassen sich mit eigenen Skripten einbinden, für die es einen eigenen Ressourcen-Typ gibt. Für diese Ressourcen ist es möglich, Abhängigkeiten zu anderen Ressourcen zu definieren. GI und ASM bieten vielfältige Möglichkeiten des Einsatzes und sind deshalb nicht nur ein überflüssiges Beiwerk bei Single-Instance- und Failover-Datenbanken.

Aufbau-Varianten

Eine Umgebung auf Basis von GI und ASM im Nicht-RAC-Umfeld lässt sich auf folgenden Basis-Plattformen realisieren:

- Bare Metal auf jedem unterstützten Betriebssystem
- Oracle VM Server for Sparc (LDOM)
- Virtualisiert als virtuelle Maschine (VM) mit
 - Oracle VM (OVM)
 - Microsoft Hyper-V VM
 - Vmware ESX VM
 - Solaris-Zonen (Flying Zones)
 - Jede andere von Oracle unterstützte VM-Plattform
- Kombinationen der genannten mit Third-Party-Clusterware oder Oracle-eigener Clusterware für Failover-Szenarien

Zu beachten sind die jeweiligen Lizenzbedingungen, vor allem beim Einsatz von Datenbanken in virtualisierten Umgebungen. Die Betrachtung der Lizenzierung ist nicht Gegenstand dieses Artikels.

Ist die Entscheidung für die grundsätzliche Architektur gefallen, so gilt es nun festzulegen, wie ASM eingesetzt werden soll. ASM bietet mehrere Möglichkeiten, Storage der Datenbank bereitzustellen. Gebündelt werden einzelne Festplatten, NFS-, SAN- oder NAS-Devices in einer Diskgruppe. Die Konfiguration der Diskgruppe entscheidet über die Ausfallsicherheit. Dafür bietet ASM drei Varianten:

- *External Redundancy*
Ausfallsicherheit ist über Betriebssystem, SAN oder NAS sichergestellt (alle RAID-Varianten möglich); verfügbarer

Nutzdatenplatz = Größe nach Spiegelung der Platten

- **Normal Redundancy**
Alle Datenbank-Blöcke werden auf zwei Platten geschrieben, der Ausfall einer Platte ist also abgedeckt; verfügbarer Nutzdatenplatz = Anzahl Platten / 2
- **High Redundancy**
Alle Datenbank-Blöcke werden auf drei Platten geschrieben und somit der Ausfall von zwei Platten abgedeckt; verfügbarer Nutzdatenplatz = Anzahl Platten / 3

Ein weiterer wichtiger Baustein für ASM ist das Sicherstellen der richtigen Zugriffs-Eigenschaften auf die verwendeten Devices sowie gleichbleibende Device-Namen. Damit ASM auch nach einem Neustart des Servers/der VM zuverlässig startet, gibt es mehrere Varianten, dies sicherzustellen:

- **UDEV**
standardmäßig in Linux vorhanden
- **ASMLib**
Oracle-Utility zum Labeln von ASM-Devices (im Oracle Linux Kernel (UEK) bereits integriert, sonst als Installationspaket für andere Linux-Systeme vorhanden)
- **ASM Filter Driver**
Wie ASMLib, erweitert um die Funktionalität, Nicht-Oracle-Datenbank-Schreibzugriffe zu blockieren. Verfügbar ab Oracle-Version 12.1.0.2 (kann in 12.2 bei der Installation direkt mit installiert werden)

Grid Infrastructure und ASM-Installation

Allen im vorigen Kapitel dargestellten Varianten folgt die Installation der Grid Infrastructure. Dies kann mit dem grafischen Installer „runInstaller“ (bis 12.1.0.2) oder mittels „gridSetup.sh“ (12.2.0.1) erfolgen. Mit Letztgenanntem geht Oracle neue Wege bezüglich der Installation: er führt nur noch die Konfiguration der GI und der ASM-Komponenten durch, während „runInstaller“ auch für die Installation der Software-Binaries zuständig ist. Dieser Schritt wurde ersetzt durch das Entpacken des GI-ZIP-Files in 12.2, wodurch sich die Installationszeit verkürzt.

```
[oracle@oel7 ~]$ crsctl stat res -t
-----
Name                               Target  State        Server        State details
-----
Local Resources
-----
ora.DATA.dg                        ONLINE  ONLINE      oel7          STABLE
ora.LISTENER.lsnr                  ONLINE  ONLINE      oel7          STABLE
ora.asm                             ONLINE  ONLINE      oel7          Started,STABLE
ora.ons                             OFFLINE OFFLINE      oel7          STABLE
-----
Cluster Resources
-----
ora.cssd                            1      ONLINE  ONLINE      oel7          STABLE
ora.diskmon                         1      OFFLINE OFFLINE      oel7          STABLE
ora.evmd                             1      ONLINE  ONLINE      oel7          STABLE
ora.orcl.db                          1      ONLINE  OFFLINE      oel7          Open,STABLE
-----
```

Listing 1

```
[oracle@oel7 ~]$ srvctl add database -db orcl \
-oraclehome /u01/app/oracle/product/12.1.0.2/dbhome_1 \
-spfile +DATA/ORCL/spfileorcl.ora

[oracle@oel7 ~]$ srvctl status database -db orcl
Database is running.

[oracle@oel7 ~]$ srvctl status asm -v
ASM is running on oel7

[oracle@oel7 ~]$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s): oel7

[oracle@oel7 ~]$ srvctl config database -db ORCL
Database unique name: ORCL
Database name: ORCL
Oracle home: /u01/app/oracle/product/12.1.0.2/dbhome_1
Oracle user: oracle
Spfile: +DATA/ORCL/PARAMETERFILE/spfile.270.914079395
Password file:
Domain:
Start options: open
Stop options: immediate
Disk Groups: DATA
Database instance: ORCL

[oracle@oel7 ~]$ srvctl config listener
Name: LISTENER
Type: Database Listener
Home: /u01/app/12.1.0.2/grid
End points: TCP:1521
Listener is enabled.

[oracle@oel7 ~]$ srvctl config asm
ASM home: <CRS home>
Password file: +DATA/orapwasm
ASM listener: LISTENER
Spfile: +DATA/ASM/ASMPARAMETERFILE/registry.253.914078551
```

Listing 2

Das kommt besonders zum Tragen bei einer Silent-Installation mit Skripten.

Als Installationsvariante wählt man „Install and Configure Oracle Grid Infrastructure for a Standalone Server“ aus und folgt der geführten Installationsroutine. Das Ergebnis ist eine fertige Umgebung mit GI inklusive Listener (wird beim Einsatz von ASM aus dem GI-Home gestartet), ASM (Instanzname ist „+ASM“) und einer Diskgruppe.

Grid-Infrastructure-Administration

Die Administration unterteilt sich grob in zwei Schwerpunkte:

- Administration der Grid-Infrastructure-Basis und Third-Party-Komponenten
- Administration von Datenbank-Komponenten

Zum Basisteil gehören Aufgaben wie das Starten und Stoppen der GI sowie die Konfiguration und Steuerung von Third-Party-Cluster-Ressourcen. Diese Aufgaben erledigt das Utility „crsctl“. Der Grid-Infrastructure-Stack (inklusive aller Ressourcen) wird mit „crsctl start has“ gestartet. Zum Stoppen wird „crsctl stop has“ ausgeführt. Ein automatisches Starten der Grid Infrastructure etwa bei einem Neustart des Ser-

vers wird mit „crsctl disable has“ unterbunden und lässt sich mit „crsctl enable has“ wieder aktivieren.

Praktische Anwendung kann dies beispielsweise beim Einspielen eines Betriebssystem-Patches bieten, wenn bei dieser Aktion ein oder mehrere Reboots nötig sind und dabei GI, ASM sowie Datenbanken offline bleiben sollen. Einen Gesamtstatus aller durch die Grid Infrastructure verwalteten Komponenten erhält man mit „crsctl stat res -t“ (siehe Listing 1).

Eigene oder Third-Party-Komponenten wie ein Apache-Webserver werden ebenfalls mit „crsctl“ gesteuert; sie können mit „crsctl start res <Ressourcenname>“ gestartet und mit „crsctl stop res <Ressourcenname>“ gestoppt werden.

Bis zur Version 11g Release 2 war es zudem möglich, alle Ressourcen mit „crsctl start/stop res <resource_name>“ zu starten und zu stoppen. Diese Funktionalität ist ab 12.1.0.2 mit dem Hinweis unterbunden, dass diese Aktionen nur noch mit „srvctl“ durchzuführen sind. Mittels „srvctl“ werden Datenbank-Komponenten gesteuert, zu denen Datenbanken, Listener, Services, ASM und Diskgruppen gehören.

Eine Datenbank kann der GI mit „srvctl add database -db <db_unique_name>“ hinzugefügt werden. Bei Verwendung des grafischen Utilitys „dbca“ wird die Datenbank bei ihrer Erstellung automatisch in

Grid Infrastructure integriert. Der Status einer Datenbank wird mit „srvctl status database -db <db_unique_name>“ angezeigt. Für die ASM-Instanz ist dies „srvctl status asm“ und für den Listener „srvctl status listener“.

Datenbanken werden mit „srvctl start|stop database -db <db_unique_name>“ gestartet und gestoppt. Gleiches ist für den Listener mit „srvctl start|stop listener“ und für ASM mit „srvctl start|stop asm“ möglich.

Die Konfiguration und Parameter werden mit „srvctl config [database -db <db_unique_name>] | [listener] | [asm]“ für die Datenbank, ASM und den Listener abgefragt. Eine allgemeine Übersicht über alle möglichen Aktionen und Parameter kann mit „srvctl [<Aktion>] -h“ dargestellt werden (siehe Listing 2).

Ergänzend zu den aufgeführten Administrationskommandos gibt es zum Anzeigen der Konfiguration noch die Möglichkeit, eine ausführliche Ansicht mit „crsctl stat res <Ressourcen Name> -p“ zu erhalten. Dabei ist zu beachten, dass der Ressourcen-Name dem Namen der Ressource aus der Statusanzeige von „crsctl stat res -t“ entspricht (siehe Listing 3).

ASM-Administration

Zusätzlich zur GI-Administration gibt es für die Administration von ASM noch weitere Utilities. Diese benötigen das Setzen der „ORACLE_SID“ auf den ASM-Instanz-Namen, der in der Nicht-RAC-Variante immer „+ASM“ ist. Zusätzlich ist das „ORACLE_HOME“ zu setzen, das dem Grid-Infrastructure-Home entspricht. Administrations-Tasks können mit folgenden Utilities durchgeführt werden:

- „sqlplus“
- „asmcmd“
- „asmca“

Der Klassiker „sqlplus“ aus dem Datenbankumfeld und seit Einführung von ASM verfügbar, erlaubt es, alle Administrations-Tasks durchzuführen. Die beiden Utilities „asmcmd“ und „asmca“ bieten ein Command-Line-Interface zur einfachen Administration von ASM. Bei „asmca“ gibt es daneben noch eine grafische Oberfläche. Die beiden Utilities ersetzen „sqlplus“ nicht in Gänze und bieten auch

```
[oracle@oel7 ~]$ crsctl stat res ora.orcl.db -p
NAME=ora.orcl.db
TYPE=ora.database.type
CLUSTER_DATABASE=false
DATABASE_TYPE=SINGLE
DB_UNIQUE_NAME=orcl
DESCRIPTION=Oracle Database resource
ENABLED=1
GEN_USR_ORA_INST_NAME=orcl
LOGGING_LEVEL=1
ORACLE_HOME=/u01/app/oracle/product/12.1.0.2/dbhome_1
SPFILE=+DATA/orcl/PARAMETERFILE/spfile.270.914079395
START_DEPENDENCIES=hard(ora.DATA.dg) pullup(ora.DATA.dg) weak(type:ora.
listener.type, uniform:ora.ons)
START_TIMEOUT=600
STOP_DEPENDENCIES=hard(shutdown:ora.DATA.dg, intermediate:ora.asm)
STOP_TIMEOUT=600
USR_ORA_OPEN_MODE=open
USR_ORA_OPI=false
USR_ORA_STOP_MODE=immediate
...
```

Listing 3

```
[oracle@oel17 ~]$ asmcmd lsdg
State      Type      Rebal ...      AU          Total_MB  Free_MB ...      Name
MOUNTED   EXTERN   N          4194304      20480     14200     DATA/

[oracle@oel17 ~]$ asmcmd lsdsk -k -G DATA
Total_MB  Free_MB  OS_MB  Name          Failgroup ... Redund  Path
10240    7100    10240  DATA_0000    DATA_0000  UNKNOWN /dev/sdb
10240    7100    10240  DATA_0001    DATA_0001  UNKNOWN /dev/sdc

[oracle@oel17 ~]$ asmcmd lsdsk -k --candidate
Total_MB  Free_MB  OS_MB  ...          Failgroup_Type ... Redund  Path
0         0        10240  ...          REGULAR      UNKNOWN /dev/sdd
0         0        10240  ...          REGULAR      UNKNOWN /dev/sde
```

Listing 4

in aktuellen Releases nicht alle Möglichkeiten der Administration.

Das Utility „asmca“ ist primär für die Konfiguration von ASM, Erstellen von Diskgruppen und Hinzufügen neuer Devices zu Diskgruppen zuständig. Zu seinen Stärken zählen die Statusanzeige von Diskgruppen und Devices sowie das Kopieren von Datenbank-Dateien nach ASM und aus ASM heraus. Zusätzlich bietet es eine Art „Datei-Explorer“, der Befehle wie „cd“, „ls“ etc. unterstützt und die Datenbank-Dateien und -Verzeichnisse sichtbar macht.

Der Status von Diskgruppen wird mit „asmcmd lsdg [-g <DG>]“ angezeigt. Eine detaillierte Sicht über die zugrunde liegenden Devices einer Diskgruppe erhält man mit „asmcmd lsdsk -k [-G <DG>]“. Sind noch unbenutzte Devices verfügbar, also Devices, die im ASM nicht verwendet werden, aber dem Betriebssystem bekannt sind, so findet man diese mit „asmcmd lsdsk -k --candidate“ (siehe Listing 4).

Hat man neue Devices verfügbar und möchte diese einer Diskgruppe hinzufügen, so ist dies mit „asmca -silent -addDisk -diskGroupName <DG> -disk <DiskName>“ möglich (siehe Listing 5).

So schön diese Utilities sind, eine Funktionalität ist exklusiv „sqlplus“ vorbehalten: das Entfernen von Devices aus einer Diskgruppe. Dies erfolgt mit „alter diskgroup <DG> drop disk <DiskName>;“. Das Löschen einer Disk aus einer Diskgruppe geschieht asynchron. Der Fortschritt dieser Aktion lässt sich durch Abfrage der View „v\$asm_operation“ oder mit dem Aufruf „asmcmd lsdsk -p -G <DG>“ verfolgen (siehe Listing 6).

Oracle hat in 11g Release 2 eine Änderung am zentralen Rechtekonstrukt

```
[oracle@oel17 ~]$ asmca -silent -addDisk -diskGroupName DATA -disk /dev/sdd
Disks added successfully to disk group DATA

[oracle@oel17 ~]$ asmcmd lsdsk -k -G DATA
Total_MB  Free_MB  OS_MB  Name          Failgroup ... Redund  Path
10240    7108    10240  DATA_0000    DATA_0000  UNKNOWN /dev/sdb
10240    7100    10240  DATA_0001    DATA_0001  UNKNOWN /dev/sdc
10240    10220   10240  DATA_0002    DATA_0002  UNKNOWN /dev/sdd
```

Listing 5

```
[oracle@oel17 ~]$ sqlplus / as sysasm
SQL> alter diskgroup DATA drop disk DATA_0002;
Diskgroup altered.

SQL> select group_number,operation,state,sofar,est_work from v$asm_operation;

GROUP_NUMBER  OPERA  STAT      SOFAR  EST_WORK
-----
1             REBAL  RUN       493    528
1             REBAL  WAIT      0      0

SQL> select group_number,operation,state,sofar,est_work from v$asm_operation;

no rows selected

[oracle@oel17 ~]$ asmcmd lsdg
State      Type      Rebal ...      Name
MOUNTED   EXTERN   Y          DATA/

[oracle@oel17 ~]$ asmcmd lsdsk -p -G DATA
Group_Num  Disk_Num ... Mode_Stat State      Path
1          0      ONLINE   NORMAL    /dev/sdb
1          1      ONLINE   NORMAL    /dev/sdc
1          2      ONLINE   DROPPING  /dev/sdd
...
[oracle@oel17 ~]$ asmcmd lsdsk -p -G DATA
Group_Num  Disk_Num ... Mode_Stat State      Path
1          0      ONLINE   NORMAL    /dev/sdb
1          1      ONLINE   NORMAL    /dev/sdc
```

Listing 6

von ASM vorgenommen. Dies entfernt Rechte von der Rolle „sysdba“ und verschiebt diese in die neue Rolle „sysasm“. Das bedeutet, dass zur Administration von ASM das „sysasm“-Recht benötigt wird und bei der Verwendung von „sqlplus“ eine Verbindung mit dem Connect-String „/ as sysasm“ durchgeführt werden muss. Für „asmcmd“ ist die Verbindung mit der Rolle „sysasm“ standardmäßig hinterlegt und somit muss nichts Weiteres beim Aufruf angegeben werden.

Logfiles

Es kann auch in der stabilsten Umgebung zu Fehlern kommen. Aus diesem Grund ist es immer gut zu wissen, an welcher Stelle sich die Logfiles befinden. Wie bei der Datenbank sind diese für GI und ASM im Automatic Diagnostic Repository (ADR), das im Verzeichnis „ORACLE_BASE/diag“ zu finden ist. Alle Logfiles zu Grid Infrastructure stehen im Verzeichnis „diag/crs/...“ und alle Logfiles zur ASM-Instanz im Verzeichnis „diag/asm/...“.

Bei Problemen mit der GI ist das zentrale GI-Alert-Logfile „diag/crs/<hostname>/crs/trace/alert.log“ (Text Alert-Log) die erste Anlaufstelle. Dort finden sich im Problemfall auch die Referenzen zu den weiteren Logfiles der GI-Komponenten, die sich ebenfalls im „trace“-Verzeichnis befinden. Bei Problemen mit ASM ist „diag/asm/asm/+asm/+ASM/trace/alert_+ASM.log“ das zentrale Logfile.

Erfahrungen

Im täglichen Betrieb einer Grid-Infrastructure-Standalone-Umgebung ergeben sich noch Optimierungsmöglichkeiten – allen voran der ASM-Instanz-Parameter „audit_sys_operations“, der seit 12.1 auf „true“ gesetzt ist. Er sorgt dafür, dass alle ASM-Operationen, die als „sys“ ausgeführt werden, also interaktive durch Administratoren und nicht interaktive durch GI- und ASM-Prozesse, standardmäßig auditiert und in das Verzeichnis „GI_HOME/rdbms/audit“ geschrieben werden. Dieses Verzeichnis unterliegt keinem automatischen Log-Rotate und sollte regelmäßig aufgeräumt werden.

Alternativ lässt sich das „sys“-Auditing auch deaktivieren.

Ein weiterer Kandidat ist der Parameter „processes“, der schon in der Datenbank des Öfteren zu Problemen führt. Er sollte auch in der ASM-Instanz genau beobachtet werden und ist bei aktuellen GI-Patchständen aktuell auf den Wert „200“ gesetzt. Werden nun viele Datenbank-Instanzen in einer einzigen GI-ASM-Umgebung konsolidiert, so kann dies unter Umständen zu Problemen führen, die erst beim Restart des Servers oder beim Restart von GI-ASM + Datenbanken auftreten. Beim Starten werden gleichzeitig zu viele Verbindungen von den Datenbanken zum ASM parallel aufgebaut. Als Folge kann es passieren, dass nicht alle Datenbanken wie vorgesehen automatisch gestartet werden, sondern ein Teil der Datenbanken gestoppt bleiben und einem manuellen Start unterliegen.

Mit Installation der GI ergibt sich auch eine nicht unwichtige Änderung in Bezug auf den Listener. Dieser wird nicht wie bisher aus dem Datenbank-Home gestartet, sondern aus dem Grid-Infrastructure-Home. Dabei gilt es zu beachten, dass es nun mehrere „tnsnames.ora“- und „sqlnet.ora“-Dateien geben kann, jeweils einen Satz für jedes Home. Wichtig ist zu wissen, dass diejenigen Parameter und Konfigurationen verwendet werden, die in der jeweiligen Datei des Oracle-Home der angesprochenen Komponente liegen. Konkret bedeutet dies, dass Einstellungen in den jeweiligen Dateien des GI-Home für die ASM-Instanz und Einstellungen im Datenbank-Home für die daraus gestartete Datenbank-Instanz gelten.

Unterschiede zwischen RAC und RAC One Node

RAC und RAC One Node erweitern die Funktionalität von ASM auf weitere Server, die in diesem Setup ein Cluster formen. Dies erfordert, dass der Storage für ASM als Shared Disk für alle Knoten zur Verfügung steht. Grid Infrastructure wird als Cluster-Setup aufgebaut, um die Kommunikation der Komponenten zwischen den Knoten zu regeln.

RAC One Node dient im weitesten Sinne als Failover-Lösung, die im Wartungs-

fall eine Verlagerung der Datenbank online durchführt und sich beim Ausfall wie ein klassischer Failover-Cluster mit einer kurzen Nichtverfügbarkeit der Datenbank darstellt. Beiden gemein ist eine vorgeschaltete Verbindungsschicht, der SCAN-Listener, der beim RAC für Loadbalancing und beim RAC One Node die Brücke zur Online-Verlagerung bietet.

Grid Infrastructure Standalone bietet eine gute Basis, um sich auch in RAC- und RAC-One-Node-Umgebungen zurechtzufinden. Alle vorgestellten Administrationstools sind auch für diese Ausprägungen nutzbar. Die vorgestellten Befehle sind grundsätzlich gleich; sie werden zusätzlich noch um RAC- und RAC-One-Node-spezifische Parameter erweitert. Mit diesem Grundverständnis fallen der Aufbau und die Administration einer RAC-One-Node- oder RAC-Umgebung deutlich leichter.

Fazit

Grid Infrastructure und ASM sind kein Mysterium und lassen sich sehr gut in Nicht-RAC-Umgebungen einsetzen. Sie bieten gute Performance sowie einfache Administration und können hervorragend zur Konsolidierung von Datenbanken genutzt werden.

Eine Erweiterung der Grid Infrastructure um Nicht-Datenbank-Ressourcen bietet die Möglichkeit, Applikationskomponenten einzubinden und diese zusammen mit der Datenbank als Komplettlösung zu betreiben.



Timo Giese
timo.giese@fiduciagad.de