

Failover-Lösung für Oracle VM (SOA Suite&ORDS) bei Carglass

Johannes Michler
PROMATIS SOFTWARE GMBH
Ettlingen (TechnologieRegion Karlsruhe)

Keywords

Hochverfügbarkeit, Failover, Oracle VM, Oracle SOA Suite, Oracle Application Express APEX, Oracle REST Data Services (ORDS), Oracle E-Business Suite, Linux

Abstract

Steinschlagschäden im Fahrzeugglas sind lästig und passieren immer zum falschen Zeitpunkt. Bei Carglass werden täglich mehr als 3.000 Autoscheiben repariert oder ausgetauscht. Im Hintergrund werden diese Reparaturen rund um die Uhr durch eine Vielzahl von IT-Systemen unterstützt: Insbesondere in Form der Oracle E-Business Suite 12.1.3, erweitert um diverse Oracle APEX-Applikationen. Die elektronische Anbindung an Kunden und Lieferanten erfolgt dabei unter Nutzung der Oracle SOA Suite 12.2.1.3.

Carglass betreibt diese Applikationen auf Solaris SPARC und Intel X86-Servern in einem firmeneigenen Rechenzentrum am Hauptstandort Köln; hierbei erfolgt eine Virtualisierung mittels „Solaris SPARC Zonen“ und „Oracle VM Server for x86“. Ob hoher Verfügbarkeitsanforderungen ist dabei sicherzustellen, dass auch bei einem Totalausfall des Rechenzentrums in Köln ein Weiterbetrieb der kritischen Applikationen sichergestellt werden kann. Hierzu wurde eine Failover-Umgebung in einem zweiten Rechenzentrum aufgebaut.

Der Vortrag beschreibt dieses Vorgehen, insbesondere für die auf Oracle VM-betriebene Oracle SOA Suite und des APEX-Listeners ORDS. Im Hintergrund wurde hierzu die Storage-Replikationstechnologie DELL/EMC RecoverPoint genutzt. Es wird detailliert auf die Erfahrungen eingegangen, die in Kombination mit EMC RecoverPoint für die Replikation von virtuellen Umgebungen auf Oracle VM gemacht wurden.

Einführung

Zielsetzung

Carglass betreibt diverse unternehmenskritische Applikationen in einem lokalen Rechenzentrum am Hauptstandort Köln. Ob der großen Bedeutung dieser IT-Applikationen für den geregelten Unternehmensbetrieb wurde ein „Fallback-Rechenzentrum“ an einem zweiten, räumlich getrennten Carglass-Standort (in Bonn) aufgebaut. Aus historischen und Lizenztechnischen Aspekten werden bei Carglass drei verschiedene Virtualisierungstechnologien eingesetzt:

- Oracle E-Business Suite Umgebungen (12.1.3) und ihre Datenbanken werden auf Solaris SPARC betrieben. Hierbei wird zur einfacheren Verwaltung die Container-Technologie von Solaris SPARC – sogenannte „Zonen“ – genutzt.
- Um langfristig auch eine einfachere Migration in eine (Public) Cloud zu ermöglichen, werden seit einiger Zeit neu aufgesetzte Oracle-Anwendungen unter Linux unter Nutzung der Oracle-eigenen Virtualisierungslösung „Oracle VM 3.4“ (OVM) aufgesetzt. Die Nutzung von sowohl OVM als auch von Solaris-Zonen bei Carglass erfolgt dabei insbesondere unter dem Aspekt, dass hierbei im Gegensatz zu vielen anderen Virtualisierungstechnologien eine sogenannte „harte“ Partitionierung möglich bzw. anerkannt ist: So müssen dann nur die CPU-Kerne für das jeweilige Oracle-Produkt lizenziert werden, auf der tatsächlich eine VM mit diesem Produkt ausgeführt wird.
- Sämtliche weitere (nicht Oracle-Anwendungen betreffende) Umgebungen werden bei Carglass – sowohl mit Windows als auch mit diversen Linux-Derivaten – unter Verwendung eines VMWare-Clusters betrieben.

Die Ziele der gesamten Hochverfügbarkeitsstrategie waren hierbei:

- Im Falle eines (Komplett- oder Teil-)Ausfalls des Haupt-Rechenzentrums in Köln, binnen weniger Stunden

eine vollständige und aktuelle Serverumgebung im Ausfall-Rechenzentrum in Bonn bereitzustellen.

- Hierzu sollte, wo möglich, eine (Teil-)Automatisierung des Umzugs von Köln nach Bonn möglich sein.
- Diverse Anwendungen (Oracle-Datenbanken, sonstige Datenbanken, aber auch diverse sonstige Anwendungen von Oracle und anderen Anbieter) und Server-Technologien „einheitlich“ von Köln nach Bonn zu sichern.
- Es sollte hierbei insbesondere nicht notwendig sein, für jede Anwendung/Datenbank ein eigenes Hochverfügbarkeitskonzept zu entwickeln.
- Schließlich soll es möglich sein, im „Normalbetrieb“ zumindest „weniger wichtige Test-/Entwicklungsumgebungen“ auf den Ausfallservern im Rechenzentrum in Bonn zu betreiben.

Auf dieser Grundlage wurde – insbesondere wegen guter Erfahrungen im Bereich der VMWare-Virtualisierung – als zentrale Basistechnologie der Hochverfügbarkeitslösung eine „Spiegelung“ auf Ebene der Speichersystem-LUNs gewählt. Im Folgenden werden zunächst Oracle VM und die dort verwendeten Begrifflichkeiten und Konzepte eingeführt; anschließend erfolgt eine Einführung in die genutzte „Storage-Spiegelungs“-Technologie „EMC RecoverPoint“.

Überblick Oracle VM

Oracle VM – genauer Oracle VM for x86 in Version 3.4 – stellt eine Virtualisierungslösung aus dem Hause Oracle dar. OVM besteht aus zwei Hauptkomponenten: dem OVM Manager zur Verwaltung einerseits und einem oder (in der Regel viele) OVM Server, auf denen die eigentlichen virtuellen Maschinen ausgeführt werden, andererseits.

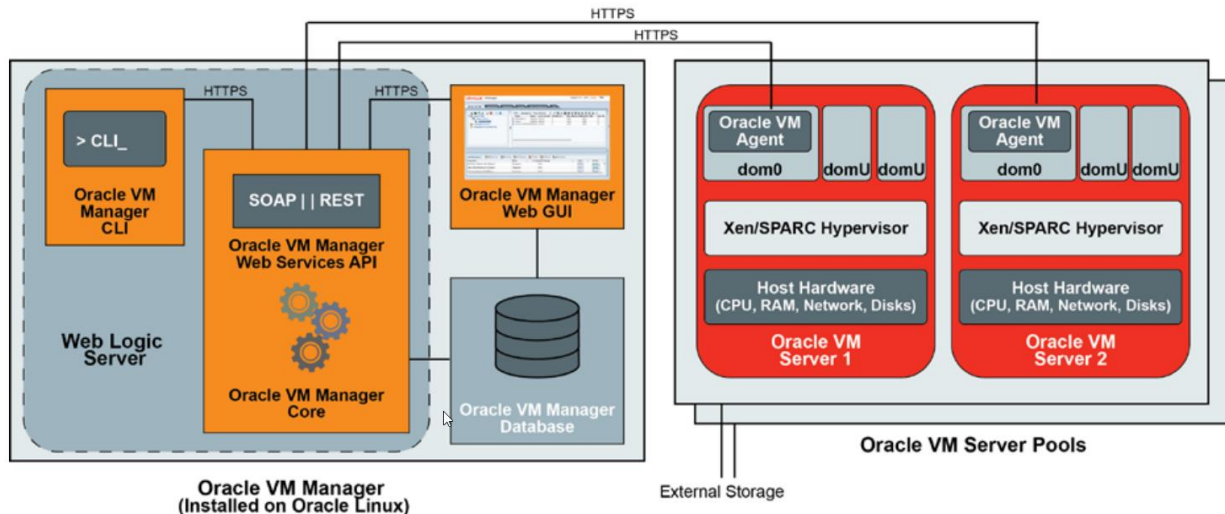


Abbildung 1: Oracle VM Architektur (entnommen aus [OVM_VirtualizeEnterpriseApps])

Jeder Oracle VM Server stellt dabei einen skalierbaren x86 (XEN-basierenden) Hypervisor dar, der jeweils bis zu 384 physikalische CPUs mit bis zu 6 TB Arbeitsspeicher besitzen kann. Jeder Server kann dabei gleichzeitig bis zu 1024 virtuelle Maschinen mit jeweils bis zu 256 CPU-Cores und 2 TB Arbeitsspeicher betreiben. Diese virtuellen Maschinen können dabei sämtliche gängige x86/x86-64-Linux-Distributionen oder Microsoft Windows-Versionen enthalten und auch komplexe Netzwerk- oder Platten-Konfigurationen vorsehen. Mehrere Oracle VM Server können in einem sogenannten „Server Pool“ kombiniert werden. Diese „Pools“ ermöglichen beispielsweise einen automatischen Neustart von VMs auf einem anderen Server des Pools beim Ausfall eines Servers oder auch die Migration einer laufenden VM zwischen zwei Servern – hierbei sind jedoch die Lizenzbedingungen der betriebenen Anwendungen zu beachten! Die Daten der einzelnen VMs können dabei entweder (eher für kleine Umgebungen geeignet) auf lokalen Festplatten der Oracle VM Server abgelegt werden oder wie bei Carglass auf einem externen Storage System verwaltet werden.

Die Verwaltung aller virtuellen Maschinen auf allen Oracle VM Servern erfolgt über eine webbasierte Oberfläche, den „Oracle VM Manager“ (Web GUI). Dieser wird (einmalig) auf einem Linux Server installiert und verwaltet sämtliche Umgebungsparameter (VM-Server, VMs, Storage-Systeme (inkl. LUNs), Netzwerk(-Segmente),...). Diese Informationen werden dabei in einer mysql-Datenbank abgelegt. Neben dem Zugriff über die Weboberfläche ist dabei auch eine Verwaltung mittels Kommandozeile (CLI) oder über (SOAP oder REST) Web Services möglich. Auch wenn

der Betrieb des Oracle VM Managers theoretisch auf einer Oracle VM möglich ist, wird hiervon im Produktivbetrieb abgeraten. Bei Carglass wurde der Oracle VM Manager auf einer Oracle Linux 7 VM unter VMWare bereitgestellt; er ist daher unabhängig von den eigentlichen Oracle-VM-Servern.

Überblick EMC RecoverPoint

EMC RecoverPoint stellt einen kontinuierlichen Schutz von Daten für operationale (insbesondere Destaster-) Wiederherstellungen bereit und ermöglicht die Wiederherstellung von Datenständen zu einem beliebigen Zeitpunkt. Dies ist auch in heterogenen Storage-Umgebungen und insbesondere auch über verschiedene Datenzentren hinweg möglich. Im Falle eines Datenverlusts oder einer „Zerstörung“ von Daten kann dabei – wenn mehrere zu schützende LUNs in einer Konsistenzgruppe zusammengefasst sind – jederzeit im Quell- oder im Ausfall-Rechenzentrum der konsistente Datenstand zu einem beliebigen Zeitpunkt bereitgestellt werden. Diese Konsistenzgruppen sind insbesondere dann wichtig, wenn sich die Daten einer relationalen Datenbank auf viele verschiedene LUNs aufteilen. Dann ist ein problemloser Start nur dann möglich, wenn diese LUNs auch untereinander in einem konsistenten Zustand sind. Im hier beschriebenen Failover-Szenario wird dies in der Regel der jeweils aktuellste Datenstand vor dem Ausfall des primären Rechenzentrums sein. Dieses Vorgehen funktioniert dabei für beliebige vom Storage System unterstützte Umgebungen / Anwendungen. Im Falle von Carglass wird es analog sowohl für VMWare-, Oracle VM- und Solaris-Server genutzt. Technisch werden dazu mit „minimaler Verzögerung“ im Sekundenbereich sämtliche Änderungen an den Datenbestände verteilt.

Die folgende Abbildung zeigt die Konfiguration von EMC RecoverPoint zur Spiegelung der für die SOA Suite notwendigen LUNs von Köln nach Bonn:

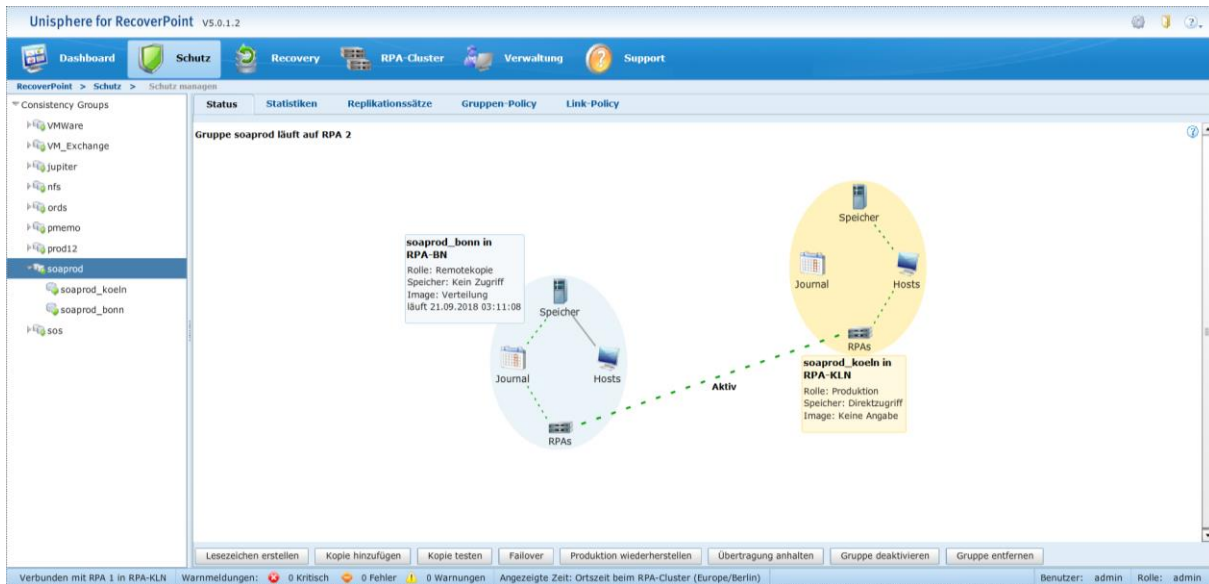


Abbildung 2: Darstellung LUN-Spiegelung von Köln nach Bonn

Grobkonzept / Vorgehensbeschreibung

Die folgende Abbildung zeigt zunächst die Ausgangssituation, wobei aus Gründen der Übersichtlichkeit auf die Auflistung der SPARC Solaris Zonen/Server verzichtet wurde:

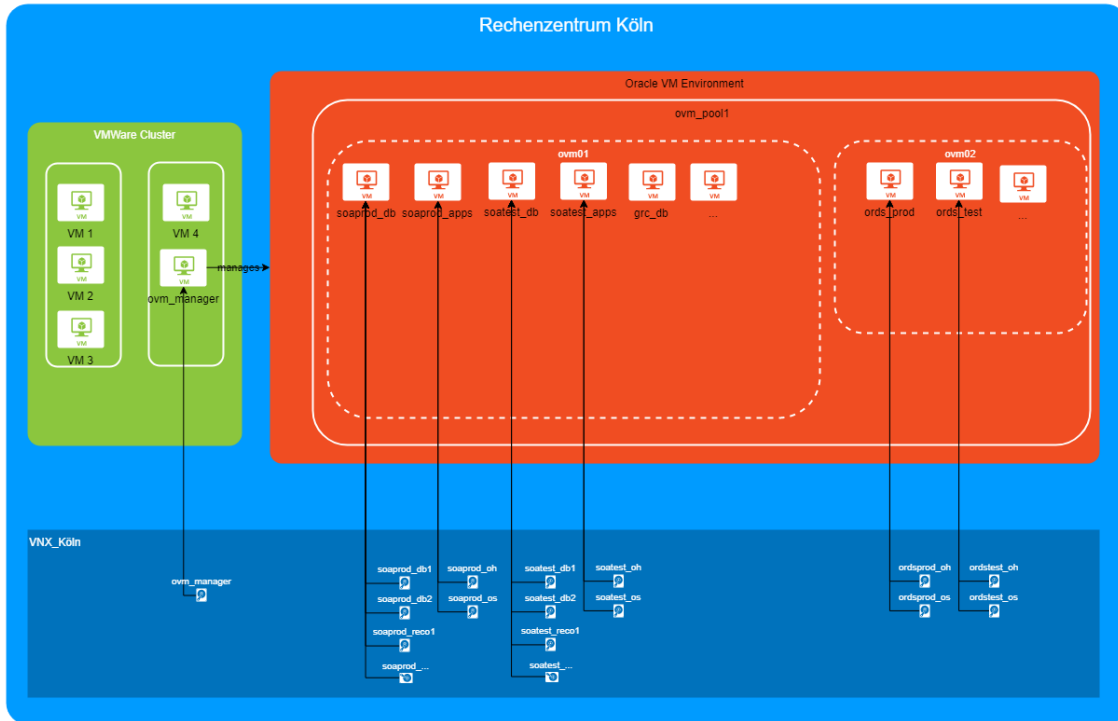


Abbildung 3: Architektur Ausgangssituation

Die Umgebung in Köln besteht also aus einer unter VMWare betriebenen Oracle VM Manager VM (`ovm_manager`), welcher einen einzigen Server-Pool `ovm_pool1` verwaltet. Dieser Pool besteht aus zwei physischen Servern `ovm01` und `ovm02`. In diesen Servern befinden sich VMs für die Test- und Produktivumgebung der SOA-Suite (inkl. Datenbank) sowie von ORDS. Jede dieser VMs ist im Hintergrund auf mehreren LUNs im Storage-System VNX_Köln abgelegt. Weiterhin besitzt jede VM genau ein (virtuelles) Netzwerkinterface, welchem genau eine IP-Adresse und ein DNS-Name zugeordnet sind. Im Zuge der Erstellung einer hochverfügbaren Architektur wurde, wie die folgende Abbildung zeigt, eine ähnliche Infrastruktur im Rechenzentrum in Bonn aufgesetzt:

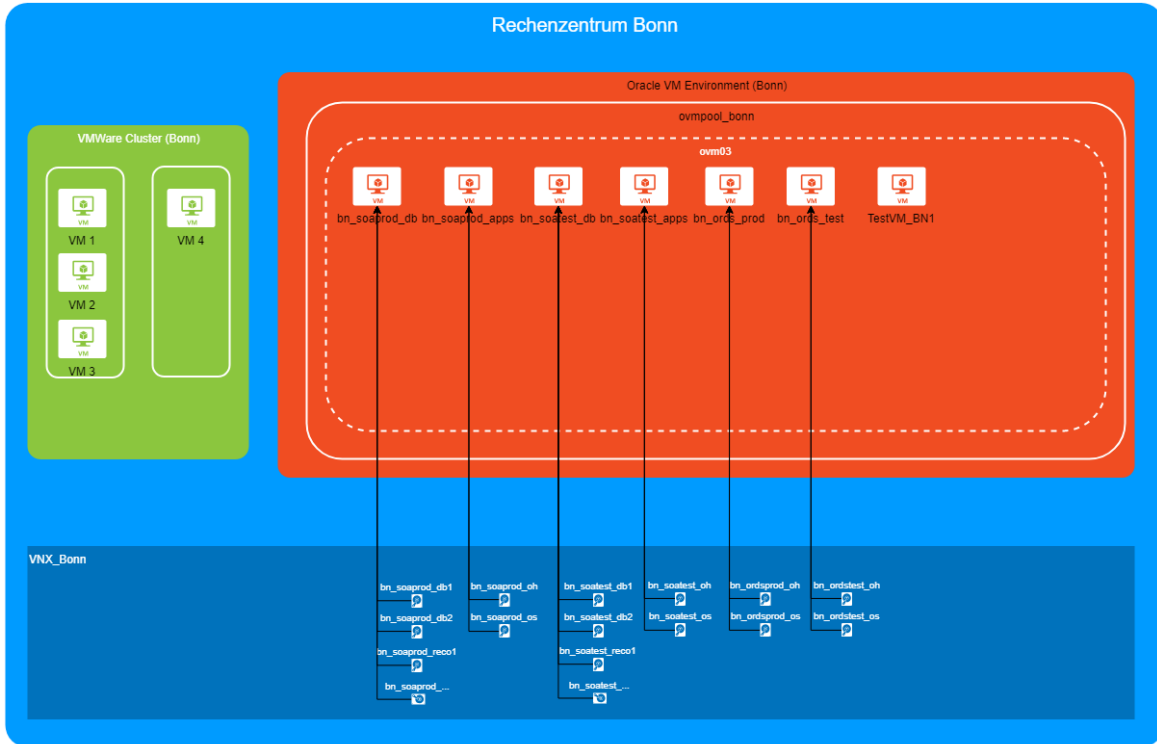


Abbildung 4: Architektur Rechenzentrum Bonn

Hierbei wurden grob folgende Schritte durchgeführt:

- 1) Zunächst jeweils "leere" LUNs gleicher Größe im Storage-System in Bonn angelegt, der Name erhielt einen bn_ Präfix.
- 2) Diese leere Luns wurden auf Storage-Ebene mit einem neuen Server in Bonn (ovm03) verbunden.
- 3) Auf dem Server wurde ein aktuelles Oracle VM installiert; der Server wurde anschließend inkl. einem neuen Server-Pool (ovmpool_bonn) mit dem in Köln laufenden OVM Manager ovm_manager verbunden.
- 4) Anschließend wurden (manuell) im Pool ovmpool_bonn alle hochverfügbar bereitzustellenden VMs mit einem bn_-Präfix angelegt. Die restliche Konfiguration entsprach 100% der Konfiguration der jeweiligen VM im primären Rechenzentrum in Köln.
- 5) Abschließend wurden die jeweiligen LUNs in Köln und Bonn über die EMC RecoverPoint Console miteinander verbunden. Dabei wurden jeweils alle zusammengehörigen LUN einer Applikation (insbesondere sämtliche LUNs einer Datenbank) in einer sogenannten „Consistency Group“ zusammengefasst: Dadurch stellt die EMC RecoverPoint-Lösung sicher, dass alle gespiegelten LUNs innerhalb einer solchen Gruppe stets zueinander konsistent gehalten sind. Nach dieser Einrichtung spiegeln die Storage-Systeme in Köln und Bonn sämtliche Daten (bzw. die Änderungen an den Daten) jederzeit von der primären/offenen Seite (normalerweise Köln) zur passiven/standby Seite (normalerweise Bonn).

Fallstricke bei der Umsetzung und ihre Lösung

Im Folgenden soll gezielt auf einige Besonderheiten / Probleme, die bei der Implementierung der Lösung aufgetreten sind, eingegangen werden.

Umgang mit dem OVM Manager

Eine wichtige Rolle für die Verwaltung der Oracle VM-Umgebung stellt der oben beschriebene Oracle VM Manager dar. Über diesen erfolgt sowohl die Verwaltung, aber auch Operationen wie Start/Stopp oder Neustart einer VM. Für den Umgang mit der (bei Carglass unter VMWare betriebenen) Oracle VM Manager kommen folgende Möglichkeiten in Betracht:

- A) Der Oracle VM Manager wird ausschließlich in Köln (am primären Standort) betrieben. Dort werden auch

die Server/VMs in Bonn verwaltet. Im Disaster-Fall erfolgen Start und Stopp ohne VM Manager mittels lokaler Kommandozeilen-Befehle auf dem Ausfall Oracle-VM-Server in Bonn.

- Vorteil: Einfach einzurichten! Es ist möglich, auch im Regelbetrieb in Bonn VMs zu verwalten – es muss nur eine Oracle VM Manager-Umgebung „gepflegt“ werden.
 - Nachteil: Im Disaster-Fall stehen nur die Xen-Kommandozeilen-Tools zur Verwaltung zur Verfügung.
- B) Ähnlich wie in Variante A werden die VMs/Server in Bonn über den zentralen Manager in Köln verwaltet. Die Oracle VM Manager VM wird im Falle eines Disasters mit Mitteln der VMWare-Virtualisierung „hochverfügbar“ in Bonn bereitgestellt; dann kann hierüber Start, Stopp und Verwaltung erfolgen.
- Vorteil: Siehe A, außerdem steht die Verwaltungsoberfläche auch bei einem Total-Ausfall in Köln zur Verfügung.
 - Nachteil: Minimal höherer Aufwand, was die initiale Einrichtung angeht (die VM muss VMWare-seitig als „in Bonn verfügbar“ markiert werden).
- C) Alternativ könnte auch eine komplett isolierte Oracle VM-Umgebung mit eigenständigem OVM Manager „ovm_manager_bonn“ aufgesetzt werden, wobei ovm_manager_bonn ausschließlich den OVM-Pool/die Server / die VMs in Bonn verwaltet.
- Nachteil: Die Verwaltung wird komplexer! Insbesondere, wenn auch im Tagesbetrieb einzelne VMs auf den Ausfall-Servern in Bonn betrieben werden sollen

Nach obigen Überlegungen wurde bei Carglass Variante B umgesetzt.

Umgang mit Netzwerk MAC-Adressen

Die MAC-Adressen der (virtuellen) Netzwerkkarten werden über den Oracle VM Manager verwaltet. Hierbei stellt der OVM Manager sicher, dass jede MAC-Adresse nur einmal verwendet wird – es ist nicht vorgesehen und möglich, dass innerhalb einer Oracle VM Manager-Umgebung zwei virtuelle Maschinen Netzwerkinterfaces mit derselben MAC-Adressen besitzen.

Leider leiten viele Linux-Distributionen standardmäßig mittels „persistent udev rules“ aus der MAC-Adresse die „Identität“ und Konfiguration der Netzwerkkarte ab. Besitzen die beiden VM-Konfiguration „in Köln“ und „in Bonn“ jedoch verschiedene MAC-Adressen, so muss sichergestellt werden, dass es keine fest auf die MAC-Adresse prüfenden udev-Regeln gibt. Bei Carglass konnten die entsprechenden Regeln einfach entfernt werden, da jede VM nur ein Netzwerk-Interface besitzt. Ist dies nicht der Fall, so können in udev-Regeln auch reguläre Ausdrücke für zwei Mac-Adressen (der jeweiligen Netzwerkinterfaces an den verschiedenen Standorten) genutzt werden.

Nicht zugreifbare LUNs

Nach Einrichtung einer Spiegelung zwischen zwei LUNs an einem aktiven (schreibbaren) und einem passiven (aktuell nicht schreibbaren) Standort, darf und kann auf der aktuell passiven Seite nicht mehr auf die LUN geschrieben werden. Das Multipath-Setup der Oracle VM Server prüft jedoch auch ohne echten „Nutz“-Zugriff auf die jeweiligen LUNs permanent, ob alle LUNs zugreifbar sind. Stellt der Multipath-Daemon fest, dass dies nicht der Fall ist, führt das zu sehr vielen Fehlern im Kernel-Log. Dadurch werden einerseits die Alarm-Tabellen des Oracle VM Managers sehr schnell vollgeschrieben und andererseits wird bei vielen nicht verfügbaren LUNs der Oracle VM Server so stark blockiert, dass er im schlimmsten Fall nicht einmal mehr sauber neustarten kann. Weiterhin wurde dadurch der Betrieb von „regulär in Bonn betriebenen VMs“ praktisch verhindert.

Da nach Rücksprache mit dem OVM-Development bei Oracle diese Prüfung nicht deaktiviert werden kann, wurden auf der aktuell „passiven“ Seite (also in der Regel in Bonn) über die Multipath-Konfigurationsdatei /etc/multipath.conf sämtliche nicht zugreifbaren Devices „ge-blacklistet“. Dazu wurde im Bereich „blacklist“ der Datei für sämtliche LUNs Einträge wie folgt vorbereitet:

```
blacklist{
...
wwid 3600601607cb02d00a7d6646fafbfe711 #bn_soaprod_db1
wwid 3600601607cb02d00a9d6646fafbfe711 #bn_soaprod_db2
wwid 3600601607cb02d00e59b6188afbfe711 #bn_soaprod_oh
wwid 3600601607cb02d00e79b6188afbfe711 #bn_soaprod_os
...
}
```

Dadurch werden die jeweiligen LUNs, solange sie „passiv“ sind, nicht mehr von Multipath betrachtet. Beim „Schwenken“ der Umgebung von Köln nach Bonn müssen diese Black-List-Einträge entfernt und anschließend Multipath neu geladen werden. Hierzu wurde jeweils ein Skript „unattach.sh“ und „reattach.sh“ implementiert. Dabei

wird in reattach.sh wie folgt vorgegangen:

1. Suchen aller wwids, durch das Suchen von Einträgen in /etc/multipath.conf, die z.B. bn_soaprod enthalten.
2. UN-Blacklisten der LUNs (# zu Beginn der Zeilen in multipath.conf hinzufügen mittels sed).
3. rescan-scsi-bus.sh (nicht zwingend notwendig, solange das Device nicht unpresented wird).
4. service multipathd reload; Hinweis: Das lädt nur die Konfiguration neu; Zugriffe auf bestehende Multipath-Devices sind davon nicht betroffen (siehe <https://access.redhat.com/solutions/261243>).
5. Multipath \$LUNID, um die LUNs anzulegen.
6. dd if=/dev/mapper/\$LUNID of=/dev/null bs=1M count=1 iflag=direct, um den Zugriff auf die LUN zu testen.

In unattach.sh wird ähnlich vorgegangen:

1. Suchen aller wwids, durch das Suchen von Einträgen in /etc/multipath.conf, die z.B. bn_soaprod enthalten.
2. Identifikation von "unter-LUNs" mittels dmsetup ls und Löschen der Unter-LUNs (Partitionen) und der Haupt-LUNs (dmsetup remove LUNID).
3. flush des multipath-Devices: multipath -f \$LUNID.
4. Blacklisten der LUNs (# zu Beginn der Zeilen in multipath.conf entfernen mittels sed).
5. service multipathd reload.

Vorgehen bei einem Ausfall

Das Vorgehen bei einem Ausfall in Köln ist in folgender Abbildung dargestellt:

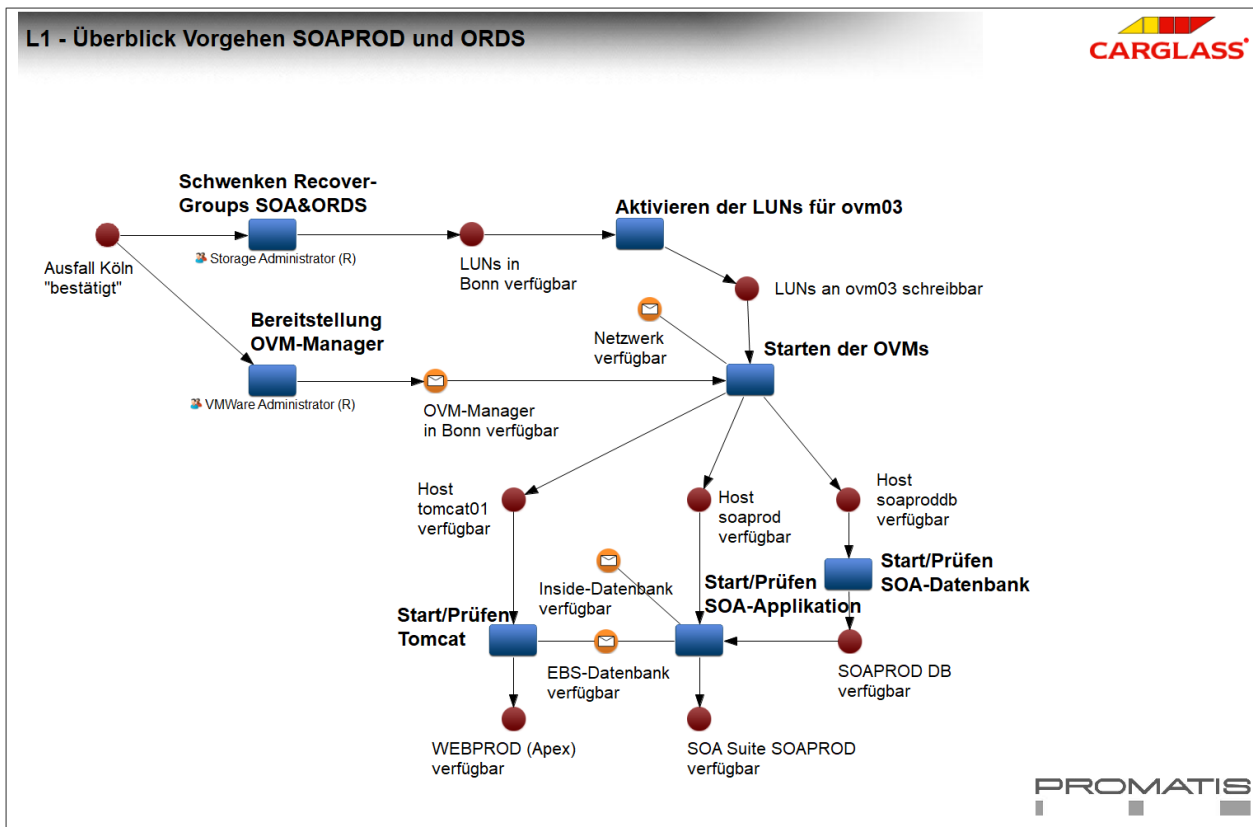


Abbildung 5: Vorgehen beim Failover

Referenzen

[OVM_VirtualizeEnterpriseApps] Oracle VM—Built for Virtualizing Enterprise Applications abgerufen von <http://www.oracle.com/us/technologies/virtualization/oracle-vm-wp-2349820.pdf>

[OVM_Overview] Oracle VM 3 - Architecture and Technical Overview abgerufen von <http://www.oracle.com/technetwork/server-storage/vm/ovm3-arch-tech-overview-459307.pdf>

Kontaktadresse

Johannes Michler

PROMATIS software GmbH

Pforzheimer Str. 160

D-76275 Ettlingen

Telefon: +49 (0) 7243 - 2179 - 260

E-Mail johannes.michler@promatis.de

Internet: www.promatis.de