

# Virtual RAC?

**Sebastian Solbach**  
**Oracle Deutschland b.v. & Co.Kg**  
**München**

## **Schlüsselworte**

12c, RAC, ASM, ACFS, Real Application Clusters, Automatic Storage Management, Grid Infrastruktur, Oracle VM, VMWare, OVM, ODA

## **Einleitung**

Heutige Server werden immer leistungsfähiger, so dass nach Mitteln und Wegen gesucht wird, diese zusätzliche Leistung sinnvoll zu nutzen. Die einfachste und schnellste Art bietet dabei die sog. Servervirtualisierung, wie Sie von den gängigen Hypervisoren, wie Xen, Oracle VM, VMWare, KVM zur Verfügung gestellt wird. Eine weitere Art der Konsolidierung auf anderer Ebene wäre Datenbankkonsolidierung auf einem Betriebssystem oder mit Oracle 12c und Multitenant sogar die Konsolidierung innerhalb einer Datenbank. In Konsolidierten Umgebungen wird aber Hochverfügbarkeit um so wichtiger, da der Ausfall eines Servers nun nicht mehr nur einige wenige Datenbanken betrifft, sondern gleich mehrere große Systeme.

Zwar hat jede Konsolidierungslösung Ihre eigenen HA Strategie, im Falle von Servervirtualisierung ist diese allerdings häufig nicht ausreichend, um einen kontinuierlichen Datenbankbetrieb zu gewährleisten. Auf der anderen Seite bietet die Datenbank mit Real Application Cluster eine sehr ausfallsichere Lösung, allerdings lässt der Real Application Cluster dafür andere Vorzüge von Servervirtualisierung außer Acht. Was liegt also näher, als die beiden Technologien zu verknüpfen?

## **Vorteile Server Virtualisierung**

Bei Servervirtualisierung unter x86 übernimmt ein Bare-Metal Hypervisor die Funktion des Betriebssystems. Dieser stellt dann in eigenen isolierten Gast Systemen unterschiedliche Betriebssysteme zur Verfügung, in der Datenbanken und andere Applikationen betrieben werden können. Diese Gast Systeme lassen sich sehr einfach aus bestehenden Systemen kreieren und so kann man sehr schnell auf dieser Umgebung die unterschiedlichsten Systeme konsolidieren. Damit einher geht eine Standardisierung und Flexibilisierung der einzelnen Deployments, was eine schnelle Installation von Umgebungen möglich macht. Nicht selten betrachtet man deswegen reine Servervirtualisierung als den ersten Schritt in Richtung Cloud Computing.

Damit liefern die Virtualisierungslösungen eine Unabhängigkeit zur Hardware und bieten rudimentäre HA Funktionalitäten, wie das Starten der Systeme nach einem Betriebssystem Crash. Ebenfalls können mit Hilfe von Live Migration Funktionalitäten die Systeme zu Hardware Wartungszwecken auf andere, allerdings prozessorgleiche Hardware verschoben werden.

Dennoch sollte generell nicht außer Acht gelassen werden, dass die x86 Hypervisor mit 2 großen Problemen zu kämpfen haben: Context Switches im Memory bei Multiusersystemen und I/O auf Netzwerk und Storage. Genau diese beiden Dinge sind aber bei Datenbanken eher an der Tagesordnung, weshalb nicht selten Performanceeinbußen zwischen 5 und 20% zu beobachten sind. Dies bezieht sich nur auf ein virtualisiertes Gastsystem und lässt sich häufig schlecht auf wirklich konsolidierte Server übertragen, da hierzu von den VM Herstellern selten Messungen vorliegen.

Das größte Problem existiert aber im Hochverfügbarkeitsbereich, dass viele HA Funktionalität rein bei der Lauffähigkeit des Gast Systems ansetzen, weniger aber die eigentliche interne Applikation überwachen können.

### **Oracle Real Application Cluster & Clusterware**

Bei der Oracle Clusterware, ein Bestandteil der Grid Infrastruktur, handelt es sich um eine vollwertige Clusterware, die auf allen gängigen Plattformen verfügbar ist. Neben der Überwachung und Steuerung von Clusterdatenbanken können auch andere Applikation unter die Überwachung der Clusterware gestellt werden. Insbesondere Oracle 12c Grid Infrastruktur hat hier einige einfache Neuerungen gebracht.

Somit ist die Oracle Clusterware ein zentraler Baustein für die Hochverfügbarkeit nicht nur für Oracle Real Application Cluster Datenbanken. Oracle ermöglicht damit eine rollierende Wartung (ein Knoten nach dem anderen) nicht nur für die Hardware, sondern auch für das Betriebssystem, die Clusterware selber und sogar Patch Set Updates (PSU) der Datenbank.

Allerdings ist die Installation von Clustern recht aufwändig und in vielen Fällen von der Hardware abhängig.

### **RAC & Virtualisierung**

Genau hier setzt die Idee an, die Vorteile beider Welten zu verknüpfen. Neben den Hochverfügbarkeitmöglichkeiten des RACs spielen die VM Funktionen, wie Isolation, einfaches Bereitstellen von Umgebungen eine große Rolle. Zusätzlich gibt es bei Oracle eigenem Hypervisor (Oracle VM) noch die Zusatzfunktion eine Teillizenzierung der Umgebung durchzuführen.

Ein gutes Beispiel wie dies Integration des schnellen Installation von RAC unter OVM aussehen kann, bieten die Real Application Clusters Templates für OVM (1), die sowohl auf dem OTN zum ausprobieren, als auch auf My Oracle Support (2) in den jeweilig verfügbaren Patches zur Verfügung stehen. Hierbei sind viele Best Practices bereits in den Templates vorhanden und die Installation der Umgebung funktioniert komplett automatisch, nach der Eingabe einiger Parameter. So kann auch nicht RAC geschultes Personal sehr einfach einen funktionstüchtigen RAC installieren.

### **Zertifizierung von RAC & VM**

Große Unklarheit besteht immer zwischen der Zertifizierung von Oracle Datenbanken unter generischen Hypervisoren und dem generellen Support. Zertifiziert bedeutet, dass der Hersteller der Virtualisierungslösung bestimmte Testszenarien der Datenbank durchgeführt hat und Oracle die Ergebnisse des Test abgesegnet hat. Dies ist insbesondere bei RAC Umgebungen interessant, da die Clusterware auf bestimmte Zeitparameter sehr sensibel reagiert und diese in zertifizierten Umgebungen selten selber richtig eingestellt werden müssen. Oft sind die Ergebnisse solch einer Zertifizierung auch Best Practice Papiere, in denen genau diese Ergebnisse festgehalten sind. Welche Lösungen im Moment zertifiziert sind, ist in der Supported Virtualization and Partitioning Technologies for Oracle Database and RAC Product Releases (3) (Ausschnitt Abb1) zu finden.

Platform	Virtualization Technology	Operating System	Certified Oracle Single Instance Database Releases	Certified Oracle RAC Database Releases
Linux x86-64	Oracle VM	Oracle VM v3	•11gR2 <a href="#">Note</a> •12cR1 <a href="#">Note</a>	•11gR2 <a href="#">Note</a> •12cR1 <a href="#">Note</a>
	Oracle VM Live Migration	Oracle VM v2	•10gR2 <a href="#">Note</a> •11gR1 <a href="#">Note</a> •11gR2 <a href="#">Note</a> •12cR1 <a href="#">Note</a>	•10gR2 <a href="#">Note</a> •11gR1 <a href="#">Note</a> •11gR2 <a href="#">Note</a>
Microsoft Windows x86-64	Microsoft Hyper-V Server 2012	Windows Server 2008 R2	•11gR2 (11.2.0.4) •12cR1 (12.1.0.2)	•No
		Windows Server 2012	•11gR2 (11.2.0.4) •12cR1 (12.1.0.1)	•No
		Windows Server 2012 R2	•12cR1 (12.1.0.2)	•12cR1 (12.1.0.2)

Abb1: Zertifizierungsmatrix Virtualisierung und Oracle Datenbank

Aber auch wenn nicht zertifiziert, gibt es auch von anderen Hersteller solche Best Practice Papiere und Oracle supported diese Umgebungen. Allerdings versucht Oracle die Fehler intern nachzustellen und wenn der Fehler nicht nachgestellt werden kann, muss sich der Kunde an den Support des VM Herstellers wenden. Dies ist auch so in der bekannten Note der Support Position zu VMWare nachzulesen „Support Position for Oracle Products Running on VMWare Virtualized Environments (Doc ID 249212.1)“ und kann auch auf andere VM Layer durchaus übertragen werden. Eigentlich die gleiche Situation, wie bei Fehlern im Betriebssystem, wenn man auf den Betriebssystemhersteller bzw. Hardwarehersteller angewiesen ist.

### Setup von RAC unter OVM

Oracle unterscheidet 2 unterschiedliche Setups für RAC Cluster in OVM Umgebungen und hat unterschiedliche Anforderungen an diese. Ein Testsystem kann somit auf einem einzigen Server installiert werden und für die Datendateien der Datenbank gibt es keine besonderen Anforderungen, genauso wenig wie für den Interconnect.

Bei Produktionsumgebungen ist es aber eigentlich selbsterklärend, dass Oracle auf einem ähnlichen Setup wie unter physikalischer Hardware setzt. Somit darf auch ein RAC Gast System niemals mit dem anderen RAC Gast System desselben Clusters zusammen auf demselben Server laufen. Dies ist eigentlich eine rein logische Überlegung, da sonst keine Hochverfügbarkeit gewährleistet ist.

Interessant in diesem Zusammenhang ist noch die Art der Virtualisierung. Unter Linux unterscheidet man generell zwischen Paravirtualisiert und Hardwarevirtualisiert. Auch wenn im offiziellen Whitepaper zu Oracle RAC und OVM (4) immer noch von PVM die Rede ist, verwendet Oracle auf seinem eigenen Engineered Systems, der Oracle Database Appliance kurz ODA, in der Zwischenzeit eine HVM mit Paravirtualisierten Treibern.

Einschränkungen innerhalb des RACs betreffen dynamische vCPU und Memory Änderungen und das „pausieren“ einer VM. Aber auch dies ist bei einem RAC eigentlich kein Problem, da dies mit Hilfe des RAC selber (Rolling Upgrade) geändert werden kann. Das gleiche gilt eigentlich auch für Livemigration. Obwohl wie in der Zertifizierungsmatrix zu sehen, Livemigration auf OVM freigegeben ist, rate ich persönlich davon eher ab, da dies eigentlich durch RAC Funktionalität besser und zuverlässiger abgedeckt werden kann.

Die eigentliche Installation eines RACs unter VM sollte denselben Best Practices folgen, wie eine Installation auf Bare Metal. Vom Folgen der Dokumentation über die Best Practices aus den RAC Starterkits (5) bis hin zu CLUVFY und dem ORACHK Utility gleicht diese anderen Installation. Insbesondere ORACHK sollte ein spezielles Augenmerk gegeben werden, da hier viele Healthchecks enthalten sind, die unter anderem auch auf inperformanten Storage schließen lassen. Gerade für einen

RAC kann ein unzuverlässiger Storage zu ungewünschten Seiteneffekten führen, von denen eine langsame Datenbank das kleinere Problem wäre.

## Netzwerk und Platten I/O

Im Vergleich zur physischen Hardware gibt es bei Verwendung eines Hypervisors erhebliche Unterschiede bei den I/O Zugriffen. Egal ob es sich um das Netzwerk oder um den Storage Zugriff handelt. Einer der größten Unterschiede, ist dass der Hypervisor für die Redundanz der entsprechenden Zugriffe sorgt und innerhalb des Gastsystems keine Mechanismen hierfür zum tragen kommen – mit Ausnahme des ASM Mirroring im Falle der Redundanz der Storages. Hierzu aber später mehr.

Im Netzwerkbereich sollte sowohl für das Public Netzwerk, als auch für den Interconnect Bonding im Hypervisor eingesetzt werden. Da mehrere VMs unter Umständen die selben physikalischen NICs verwenden, ist es sehr ratsam einen Quality of Server auf den Netzen zu ermöglichen, um Performance Engpässe auszuschließen. Dies ist insbesondere für den Interconnect wichtig, der im übrigen bei produktiven Systemen zwingend über eine andere physikalische NIC abgebildet werden muss, als der Public Netzwerk Verkehr – im übrigen nicht anders als bei einem normalen RAC.

Eine Besonderheit gibt es bei der Installation noch für das Public Netzwerk: Wurde in 11gR2 der Netzwerkausfall über ein normales „link down“ festgestellt, so hat sich gezeigt, dass insbesondere in VM Umgebungen dies nicht funktioniert und somit die Virtuelle IP nicht auf einen anderen RAC Knoten geschwenkt wurde. Ursache hierfür ist, dass bei einem Netzwerkausfall auf dem Hypervisor die sogenannte Brücke der VM auf die NIC bestehen bleibt. Dies ermöglicht, dass die VM auch mit anderen Gastsystemen auf demselben Rechner weiter kommunizieren kann. Für eine RAC VIP ist dieser Umstand natürlich ärgerlich. Daher führt 12c die Möglichkeit ein, ein sogenanntes PING Target zu spezifizieren, welches vom Cluster für die Netzwerkprüfung angezogen wird. Dies sollte eine immer erreichbare IP außerhalb der VM Umgebung sein. Ein Gateway ist dabei nur bedingt geeignet, besser die IP des Switches oder ähnliches. Bei der Installation wird dieses Ping Target direkt dem Installer mitgegeben:

```
./runInstaller oracle_install_crs_ping_Targets=Host1/IP1,Host2/IP2
```

Auch für den Interconnect sollte die Redundanz auf dem Hypervisor festgelegt werden. Die Cluster eigene Hochverfügbare Interconnect IP (kurz HAIP) eignet sich aus ähnlichen Gründen wie beim Public Netzwerk nicht für die Ausfallsicherheit. Ansonsten muss für den Interconnect aber keine spezielle VM Änderung vorgenommen werden, da die Mechanismen zur Split Brain Ermittlung in der Clusterware anders verankert sind, als nur durch einen Link Down Erkennungsmechanismus. Einzig das Verwenden von Jumbo Frames bleibt auch in VM Umgebungen empfohlen.

Zur Erkennung von I/O Problemen gibt es im Cluster 2 Timeouts. Den CSS misscount des Netzwerk Heartbeats (Default 30 Sekunden) und den CSS Disktimeout für den Zugriff auf die Voting Disks (Default 200 Sekunden). Gerade der erstere erscheint in VM Layern als etwas eng gewählt, da ein Ausfall und Umschalten von I/O Pfaden (Netzwerk und Platte) durch den zusätzlichen Stack länger braucht. Daher zeigten meine persönlichen Erfahrungen, dass der Misscount auf 60 oder 90 Sekunden gesetzt werden sollte (was im Übrigen auch im RAC auf OVM Whitepaper für Livemigration beschrieben wird). Natürlich sollte man sich der Implikation dieser Erhöhung bewusst sein (6).

Bei der Storage Anbindung wird die Redundanz ebenfalls auf dem Hypervisor sichergestellt, es sei denn man verwendet NFS oder iSCSI, das direkt im Gastsystem angebunden wird. D.h. für ein SAN System wird das Multipathing ebenfalls im Hypervisor hinterlegt, wobei die Multipathing Einstellungen vom Storage Hersteller kommen sollten. Trotzdem ist es wichtig, diese zu prüfen, da zu

hohe Timeouts beim RAC zu Fehlern führen können. Diese einzelnen Settings richtig zusammenzuführen, ist eine recht große Herausforderung und hoffentlich bringt ein künftiges RAC auf OVM Papier hier einige Hinweise. In diesem Zusammenhang fällt häufig eine Änderung im ASM Layer auf, die mit 11.2.0.3.4 eingeführt wurde. ASM hat neben den oben erwähnten Disktimeout noch einen eigenen PST Timeout bei Normal- oder High-Redundancy Diskgruppen, der im Moment auf 15 Sekunden steht. Sehr häufig reicht dieser in VM Umgebungen nicht aus und sollte auf 120 Sekunden über den Parameter `_asm_hbeatiowait` angepasst werden. Genaueres hierzu findet man unter MOS in Note 1581684.1: ASM diskgroup dismount with "Waited 15 secs for write IO to PST (7). Dies betrifft übrigens nicht nur RAC Umgebungen, sondern auch Single Instanz Umgebungen, bei denen über mehrere Storages gespiegelt wird.

Die einzelnen LUNs für die Oracle Datenbank für ASM sollten bei XEN basierten Hypervisor als PHY: Device (Block Device) direkt durchgereicht werden. Image Files auf Hypervisor eigenen Filesystemen eignen sich nur selten und können gerade im Bezug auf die Timeoutproblematiken Fehleranalysen sehr schwer machen. So verfügt das von OVM verwendete OCFS2 selbst auch über eigene Timeouts, die dann in Einklang mit den ASM Timeouts gesetzt werden sollten. Ein unnötig komplizierter Vorgang, der auch noch zusätzliche I/O Latenzzeiten mit einfügt.

Aber nicht nur die Storageanbindung ist wichtig, auch die Anzahl der LUNs. So ist es zwar bei ASM in der Zwischenzeit Best Practice 4x die Anzahl der Pfade zum Storage als minimale LUN Anzahl zu haben. Dies kann aber in VM Umgebungen, in denen das Limit zwischen 64 und 107 LUNs pro VM liegt, schnell zu einem Problem werden und sollte deswegen immer im Hinterkopf behalten werden. So wäre dann auch bei mehreren Datenbanken in einem Gastsystem von einer ASM Diskgruppe pro Datenbank abzuraten, da man sehr schnell an diese Grenze stoßen würde. Auch das dynamische Vergrößern der LUNs würde dann erheblich erschwert, wenn man einmal an diese Grenze gestoßen ist (da kein leichter Plattenaustausch mehr vorgenommen werden kann).

## Overprovisioning

Ein interessanter Aspekt von VM Layern ist das sogenannte Overprovisioning von CPU und Memory. Memory - insofern überhaupt vom Hypervisor unterstützt - sollte für eine Datenbank VM niemals zum Einsatz kommen. Genau sowenig, wie ein RAC Knoten swappen sollte, sollte auch der Hypervisor ähnliches tun.

Bei der CPU gibt es eine generelle Best Practice Vorgabe für stabile RAC Umgebungen, die besagt, dass niemals mehr als die doppelte Anzahl an CPU Kerne vergeben werden sollten. Allerdings gibt es mehrere Stellen und Möglichkeiten, diese CPUs zu vergeben:

- Via. Hyperthreading der CPU
- Mit Hilfe des Hypervisors
- Über den `CPU_COUNT` der Datenbank, wenn mehrere Datenbanken innerhalb des Gastsystems laufen.

Aus meiner Erfahrung, ist es ratsam, dass Hyperthreading auf der CPU auszuschalten, da nicht jeder Hypervisor optimal damit umgehen kann und lieber den Hypervisor das Overprovisioning machen zu lassen. Eine Ausnahme hierzu wäre, wenn pro physikalischem Server nur ein einziges Gastsystem zum Einsatz kommt. Für die Datenbank(en) innerhalb der VM sollte in jedem Fall Instance Caging (`CPU_COUNT` Parameter + Resource Manager Plan) aktiviert werden, insbesondere dann, wenn mehr als eine Datenbank innerhalb eines Gastsystems laufen. Nur so kann sichergestellt werden, dass die geforderte Anzahl (2x CPU Kerne) nicht überschritten wird.

Generell empfohlen, nicht nur für größere Datenbanken (>10GB Memory), ist die Verwendung von HugePages. Diese Empfehlung gilt selbstverständlich auch für Datenbanken in VM. Allerdings gibt es hier im Gegensatz zur Physik noch zu berücksichtigen, dass nicht jeder Hypervisor mit jeder Version dies zur Verfügung stellt. So kann OVM bei einer PVM erst mit dem UEK Kernel 2.6.39-100.0.21 des Gastsystems sicherstellen, dass Hugepages auch zur Verfügung stehen (8)

### **Sonstige Einstellungen**

Sonstige Besonderheiten bei RAC und OVM gibt es auch bei der Zeitsynchronisation. Jeder der Virtualisierung im Einsatz hat, kennt das Problem, dass die Zeit innerhalb der Gastsysteme leicht unterschiedlich ist. Für einen RAC ist dies absolut nicht wünschenswert, aber in den neueren Linux Versionen muss dafür einfach NTP konfiguriert werden. Setzen besonderer Parameter (wie Sie auch noch im Best Practice Whitepaper RAC auf OVM enthalten sind) sind nicht mehr notwendig.

Ein besonderes Augenmerk sollte noch auf VM HA gelegt werden. Wie oben schon erwähnt dürfen für einen produktiven RAC die RAC Knoten niemals auf demselben Server laufen. Dies darf natürlich auch nicht zufällig passieren, falls die HA Funktionalität im Hypervisor aktiviert wurde und dadurch das Gastsystem im Fehlerfall einfach auf demselben Server landet. Hierfür sollte der VM Administrator Sorge tragen.

Gerne wird RAC ja als „Extended Distance Cluster“ aufgesetzt und dies ist natürlich auch innerhalb der Gastsysteme möglich. Allerdings findet man z.B. bei OVM, dass OVM keinen Stretch Cluster unterstützt. Damit ist aber nicht die RAC Funktionalität in den Gästen gemeint, sondern dies bezeichnet, dass ein OVM Server Pool nicht mehrere Rechenzentren überspannen sollte. Dies hängt unter anderem damit zusammen, dass das verwendete OCFS2 Pool Filesystem nicht über 2 Storages verteilt werden kann und weil die Timings des OCFS2 für einen Stretch Cluster zu eng gewählt sind. Deswegen spannt man lieber 2 Serverpools (einen pro Rechenzentrum) auf und lässt die Stretch Funktionalität komplett von RAC übernehmen. Damit hat man auch die oben erwähnte HA Problematik gleich mit umgangen.

Insbesondere bei Stretch Clustern sollte aber auf die Timings geachtet werden und darauf dass es bei Stretch Clustern zu einem Split Brain der Knoten und des Storages kommen kann. Dieser Umstand sollte dem RAC bekannt gemacht werden, wie in MOS Note 1623400.1 erklärt (9).

### **Fazit**

RAC auf Basis OVM ist keine Besonderheit, wie man auch an den Oracle eigenen Systemen, der ODA erkennen kann. Auf der anderen Seite kann man von diesen Systemen sehr viel lernen und auf generische RAC auf VM Systeme umlegen. Wenn man insbesondere auf Timings achtet und das VM System dieselben Voraussetzungen erfüllt, wie ein physikalisches RAC System, spricht eigentlich nichts dagegen.

Leider genügen generische VM Umgebungen aber nicht den speziellen Anforderungen einer RAC Umgebung, gerade dann wenn viel konsolidiert wird. Deshalb sollten diese Art der Umgebungen gut überprüft werden und sehr häufig auf Best Practices untersucht werden. Dazu gehört nicht nur aktuelle Software einzusetzen, sondern auch den Tools, wie man Sie von physikalischen Clustern her kennt (z.B ORACHK). Dann sollte auch einem RAC auf VM Umgebung nichts mehr im Wege stehen.

### **Links**

1) Oracle RAC VM Templates für OVM

<http://www.oracle.com/technetwork/server-storage/vm/database-templates-12c-11gr2-1972804.html>

2) Oracle VM Templates for Oracle Database - Single Instance, Oracle Restart (SIHA) and Oracle RAC (Doc ID 1185244.1)

<https://support.oracle.com/epmos/faces/DocumentDisplay?id=1185244.1>

3) Supported Virtualization and Partitioning Technologies for Oracle Database and RAC Product Releases

<http://www.oracle.com/technetwork/database/virtualizationmatrix-172995.html>

4) Oracle Real Application Clusters in Oracle VM Environments

<http://www.oracle.com/technetwork/database/options/clustering/oracle-rac-in-oracle-vm-environment-131948.pdf>

5) RAC Starter Kits

<https://support.oracle.com/epmos/faces/DocumentDisplay?id=810394.1>

6) MOS Note 294430.1: CSS Timeout Computation in Oracle Clusterware

<https://support.oracle.com/epmos/faces/DocumentDisplay?id=294430.1>

7) MOS Note 1581684.1: ASM diskgroup dismount with "Waited 15 secs for write IO to PST

<https://support.oracle.com/epmos/faces/DocumentDisplay?id=1581684.1>

8) How to Enable HugePages for a Para-Virtualized Machine ( PVM ) on Oracle VM 3.2 (Doc ID 1529373.1)

<https://support.oracle.com/epmos/faces/DocumentDisplay?id=1529373.1>

9) Normal Redundancy : Storage split caused crash of ASM diskgroup (Doc ID 1623400.1)

<https://support.oracle.com/epmos/faces/DocumentDisplay?id=1623400.1>

**Kontaktadresse:**

Sebastian Solbach  
Oracle Deutschland b.v. & Co. Kg.  
Riesstrasse 25  
D-80992 München

Telefon: +49 (0) 711-72840 239  
E-Mail: sebastian.solbach@oracle.com  
Internet: www.oracle.de