Nr. 6 | Dezember 2016 | ISSN 2366-7915 | www.doag.org

Red Stack Magazin

Topaktuell

Oracle Database Appliance X6-2

Im Interview

Dr. Thomas Petrik, Leiter Technology Consulting, Sphinx GmbH

Best Practice

Kostengünstiges Datenbank-Cloning

DevCamp 2017

8. Februar in Hannover

Development by choice

Moderne Software-Entwicklung mit Oracle

devcamp.doag.org

Christian Trieb
DOAG-Vorstand und Leiter der Datenbank Community

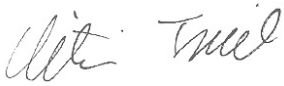
Liebe Mitglieder, liebe Leserinnen und Leser,

Hochverfügbarkeit steht schon seit vielen Jahren auf der Tagesordnung im Applikations- und Datenbank-Betrieb. In einer Welt, in der Informationen jederzeit und überall verfügbar sein müssen, nimmt dieses Thema einen immer größeren Stellenwert ein. Dies spiegelt sich in den Artikeln dieser Ausgabe auch wider. Sie stellen die unterschiedlichen Konzepte und Realisierungen im Oracle-Umfeld vor. Aus unterschiedlicher Perspektive sind die verschiedenen Aspekte der Hochverfügbarkeit – insbesondere im Datenbank-Umfeld – beschrieben und deren Nutzen anhand von Einsatzbeispielen aufgezeigt.

Hochverfügbarkeit gilt für Technik sehr wohl. Aber Sie als Leserinnen und Leser dieser Ausgabe müssen nicht immer verfügbar sein. Nehmen Sie sich in diesem Jahr die Zeit, auch mal nicht verfügbar zu sein. Sie werden sehen, es lohnt sich, um danach mit neuem Schwung Ihre Aufgaben angehen zu können.

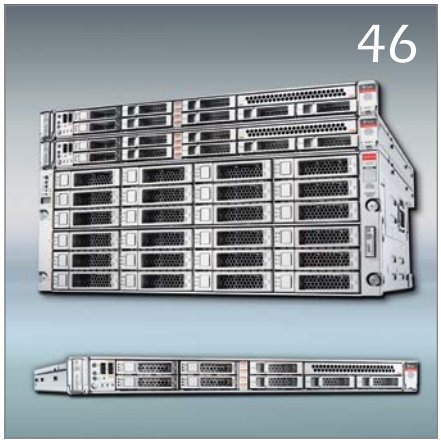
In diesem Sinne wünsche ich Ihnen ein gutes neues Jahr 2017, viel Erfolg bei Ihren Projekten und Spaß beim Lesen dieser Ausgabe.

Ihr



Kritische Geschäftsprozesse brauchen hochverfügbare IT-Systeme

Desaster-Szenarien über Rechenzentrumsgrenzen hinaus abdecken



Die neuen Oracle Database Appliance Server X6-2

3 Editorial

5 Timeline

8 „Oracle selbst bietet keine preisgünstigen Lösungen an …“
Interview mit Dr. Thomas Petrik

Hochverfügbarkeit

11 Hochverfügbarkeit ist kein reines Infrastruktur-Thema
Bernd Usinger

14 Zehn Mal Hochverfügbarkeit
Martin Klier

17 Hochverfügbarkeit oder die Suche nach den 100 Prozent Netzwerk-Sicherheit
circular Informationssysteme

21 Aufbau einer 12*c*-RAC- und Data-Guard-Umgebung mit NFS Storage bei der DEVK
Johannes Ahrends und Tim Hensel

24 Oracle VM und Virtual Shared Storage
Nico Henglmüller & Dr. Thomas Petrik

27 Data Guard Basics
Marco Mischke

32 Oracle Maximum Availability Architecture
Sebastian Solbach

37 Backup-Verfahren in der Praxis – optimiert und intelligent
Roland Stirnimann

41 Bulletproof fail over: Passive first!
Christoph Münch

Aktuell

46 Oracle Database Appliance X6-2 — ein Erfahrungsbericht
Johannes Kraus

52 Optimale Vorbereitung auf Oracle-Zertifizierungen
Rainer Schaub

Datenbank

56 Schnelles und kostengünstiges Datenbank-Cloning mit dem ASM-Cluster-File-System
Sebastian Solbach

Entwicklung

60 Der hierarchische Profiler
Jürgen Sieben

Intern

45 Termine

66 Neue Mitglieder

66 Inserenten

66 Impressum

Timeline

20. September 2016

In Baden-Dättwil findet der dritte SOUG Day in diesem Jahr statt. Es ist einer der bestbesuchten Anlässe der SOUG; mit drei Streams (Datenbank/Konsolidierung, Infrastruktur/Engineered Systems und Java/Security) sowie insgesamt zwölf Vorträgen wird den Besuchern sehr viel geboten. Neben den spannenden Vorträgen im Soft-Bereich ist es auch wieder interessant zu hören, was bei der Infrastruktur mit Oracle möglich ist. So sind die Teilnehmer vom M7-Prozessor, von Oracle DB Appliance und Zero Data Loss Recovery Appliance fasziniert. Die Pausen und der anschließende Apéro dienen zum Netzwerken und Erfahrungsaustausch; der Dank geht an die Sponsoren dbi services und Trivadis.

26./27. September 2016

Im Berliner Expertenseminar geht Andreas Nobbmann kurz und knackig auf die wichtigsten Fragen rund um Administration und Maintenance im Oracle Data Integrator (ODI) ein. Da das Produkt als potenzieller Nachfolger des OWB für Kunden von Interesse ist, stellen sich natürlich auch viele Fragen hinsichtlich Administration und Wartung einer ODI-Installation.

29./30. September 2016

Die DOAG veranstaltet zum dritten Mal die Big Data Days – ein zweitägiges Treffen mit viel Wissens- und Erfahrungsaustausch rund um das Thema „Oracle und Big Data", aber auch mit jeder Menge Networking und einem gemeinsamen Abend. Besonders erfreulich ist der positive Trend bei der Teilnehmerzahl. Im ersten Vortrag „Big Data Architekturen" führt Guido Schmutz von Trivadis die Besucher, von denen die meisten zum ersten Mal bei den DOAG Big Data Days sind, in das Thema ein. Auch alle anderen Dozenten tragen mit ihren Beiträgen zum abwechslungsreichen Programm bei – das Feedback der Teilnehmer fällt entsprechend positiv aus.

Konzentrierte Aufmerksamkeit bei den Vorträgen

29. September 2016

Der Interessenverbund der Java User Groups (iJUG) e.V., in dem die DOAG Mitglied ist, hat sich für einen Sitz im Executive Committee des „Java Community Process" (JCP) beworben. Der Java Community Process (JCP) ist seit dem Jahr 1998 dafür zuständig, in welcher Form sich die Programmiersprache Java und die gesamte Java-Plattform weiterentwickeln. Bei den vom 1. bis 14. November 2016 anstehenden Neuwahlen werden sechzehn der 25 Sitze im Executive Committee neu vergeben. Für einen dieser Sitze hat sich der iJUG beworben, repräsentiert durch Andreas Badelt, den stellvertretenden Leiter der DOAG Java Community. Am „Community Day" der diesjährigen JavaOne fand eine öffentliche Sitzung des Executive Committee statt, bei der die Kandidaten für die zu vergebenden Sitze die Gelegenheit hatten, sich der Community zu präsentieren. Andreas Badelt ist für den iJUG vor Ort und stellt dessen Hauptanliegen vor: den mehr als 20.000 Java-Entwicklern, DevOps etc., die der iJUG vertritt, eine Stimme im wichtigsten Standardisierungsgremium zu geben, damit verbunden mehr Verantwortung der Community zu überlassen (exemplarisch die momentanen Probleme um Java EE 8) sowie eine generell demokratischere Ausrichtung des JCP zu verfolgen.

4. Oktober 2016

Oracle Österreich hält im Rahmen der AOUG ein technisches Frühstück zu den Themen „Oracle Forms 12c" sowie „Oracle Application Builder Cloud Services". Marian Kuna, Technology Sales Consultant bei Oracle Slovakia, zeigt die neuesten Oracle-Forms-12c-Features und die Möglichkeit, Forms-Programme einfach mittels AuraPlayer auf Mobile Devices zu portieren.

4./5. Oktober 2016

Zum ersten Mal findet im Stuttgarter Dormero-Hotel ein neues Konferenz-Format statt, der DOAG Financial Day. Rund 50 Experten und Anwender treffen sich, um über alle relevanten Themen aus dem Finanzbereich für Oracle-Anwender zu sprechen. Wie wichtig ein Anwenderforum speziell für Oracle-Kunden ist, zeigt sich gleich am ersten Tag. Ute Moser und Mirko Duschek, beide Mitarbeiter bei Finanzämtern, präsentieren den Teilnehmern, welche Probleme und Folgen es bei den verschiedenen Zugriffsarten aus Sicht der Finanzverwaltung gibt. Oracle-Anwendungen unterstützen nicht von Haus aus deutsche Standards bei Zugriffen und Datenabfragen und müssen dementsprechend nachgerüstet werden. Neben weiteren hochkarätigen Vorträgen in vier parallelen Streams zu Themen aus den Bereichen „Hyperion" und „E-Business Suite" wird der DOAG-Botschafter-Pokal 2016 für die Business Solutions Community vergeben. Nadia Bendjedou, Leiterin der Oracle-E-Business-Strategie in Paris, erhält die Auszeichnung für ihren Einsatz in der DOAG-Community. Bendjedou ist selbst mit zwei Vorträgen vertreten und spricht darin über die Oracle-Pläne zur Zukunft der E-Business Suite und der Cloud-Anbindung. Am Ende der zweitägigen Konferenz zieht Frank Schönthaler, Leiter der DOAG Business Solutions Community, eine positive Bilanz: „Viele qualitativ hochwertige Vorträge und interessante Gespräche während dieser zwei Tage zeigen, dass ein großer Gesprächsbedarf und viele Unsicherheiten vorhanden sind, gerade auch bezüglich der Oracle-Cloud-Strategie." Wichtig sei es zudem, dass sich die Oracle-Kunden untereinander organisieren,

um den Informationsbedarf zu den Themen des Financial Day zu decken und um sich austauschen zu können. Beste Aussichten also auf eine Fortführung des DOAG Financial Day im Jahr 2017.

Nadia Bendjedou, Leiterin der Oracle E-Business-Strategie in Paris, ist DOAG-Botschafterin 2016 für die Business Solutions Community

5./6. Oktober 2016

Das nächste große Expertenseminar in Berlin zeigt das Informationsbedürfnis der Oracle-Anwender. Oliver Lemm und Kai Donato beschäftigen sich intensiv mit den Neuerungen in Apex 5.1 und den neuen Web-Technologien.

10. Oktober 2016

In der traditionellen monatlichen Telko reden die DOAG-Vertreter mit Tom Scheirsen, EMEA User Groups Relationship Manager von Oracle. Sie informieren ihn über die anstehenden Projekte der DOAG und laden ihn offiziell nach Nürnberg zur DOAG 2016 Konferenz + Ausstellung ein, wo die DOAG am Vortag das International Oracle Usergroup Meeting organisiert. Er sagt die Teilnahme zusammen mit seinem Stellvertreter Klaus Bergius von Oracle Deutschland zu.

13. Oktober 2016

Schwerpunkt beim Treffen der Regionalgruppe München ist die Nachlese zur Oracle OpenWorld. Matthias Weiss, Oracle Deutschland, stellt die Neuheiten vor. Im Anschluss daran gibt Markus Kißling, ebenfalls von Oracle, einen Überblick über die Oracle Datenbank 12c Release 2.

Der Vortragsraum in der Münchener Oracle-Niederlassung ist bis auf den letzten Platz gefüllt

17. Oktober 2016

Fried Saacke, DOAG-Vorstand und Geschäftsführer, klärt mit dem Catering-Unternehmen Lehrrieder die letzten Details für den Bewirtungsservice auf der DOAG 2016 Konferenz + Ausstellung in Nürnberg. Am Community-Abend wird es wieder verschiedene Themenbuffets mit Spezialitäten aus allen Teilen der Welt geben.

Die DOAG 2016 Konferenz + Ausstellung bietet den Teilnehmern ein reichhaltiges Buffet

18. Oktober 2016

Paul Wehner, Senior Director Sales Consulting und als Nachfolger von Günther Stürner Leiter des Consulting-Teams bei Oracle Deutschland, besucht die DOAG-Geschäftsstelle in Berlin. Im Gespräch mit Stefan Kinnen, Vorstandsvorsitzender der DOAG, und Fried Saacke, DOAG-Vorstand und Geschäftsführer, sagt er zu, dass die DOAG auch weiterhin Unterstützung von Oracle erhält, insbesondere bei den Treffen der Regionalgruppen. Er betont, dass er als Ansprechpartner für alle Fragen zur Verfügung steht.

20. Oktober 2016

Der SOUG-Vorstand trifft letzte Vorbereitungen für den Auftritt auf der DOAG 2016 Konferenz + Ausstellung in Nürnberg und freut sich darauf, viele Teilnehmer auf dem Stand der SOUG begrüßen zu dürfen.

Der Stand der SOUG war im Vorjahr gut besucht

21. Oktober 2016

Die Mitgliederversammlung des Interessenverbunds der Java User Groups (iJUG) e.V., in dem die DOAG Mitglied ist, findet im Anschluss an das Java Forum Nord in Hannover statt. Wichtige Themen sind die JavaLand 2017, das vom 28. bis 30. März 2017 im Phantasialand in Brühl stattfindet, sowie die Frage, ob Oracle den Java Community Process (JCP) noch ausreichend unterstützt.

26. Oktober 2016

Stefan Kinnen, Vorstandvorsitzender der DOAG, begrüßt den neuen Anwenderbeirat zum Kick-off-Meeting. Das Gremium aus der Businesswelt wird die DOAG künftig strategisch beraten. Es nehmen Vertreter der Alte Leipziger Versicherung, der Barmenia Versicherung, der Bundesagentur für Arbeit sowie von MyToys teil. Schon nach der Vorstellung ist klar: Die Runde deckt thematisch einen Großteil des Oracle-Stacks ab. Ein erster Erfahrungsaustausch zu den Themen „Lizenzierung", „Support-Änderungen in Deutschland" und „Kommunikation mit dem Oracle-Vertrieb" sowie zu Fragen der Investitionssicherheit durch eine klare, langfristige Produktstrategie zeigt das offenkundig große Interesse und die Motivation zur Zusammenarbeit, denn die Themen kamen dabei aus dem Kreis der Teilnehmer und waren nicht vorgegeben. Die DOAG kann sich also zukünftig auf mehr Praxisrelevanz in der Darstellung und Diskussion freuen. Das nächste Treffen ist für das erste Quartal 2017 geplant. Auf der Agenda stehen dann die Vertiefung der diskutierten Schwerpunkt-Themen sowie die Formulierung möglicher Aktionen gegenüber Oracle.

1. November 2016

Das Organisations-Team der DOAG kommt in der Geschäftsstelle zum Kick-off-Meeting zusammen, um die Durchführung der DOAG 2016 Konferenz + Ausstellung zu finalisieren. Fried Saacke, DOAG-Vorstand und Geschäftsführer, ist überzeugt, auch in diesem Jahr wieder optimale Rahmenbedingungen für die Jahreskonferenz der DOAG bieten zu können.

2. November 2016

Fried Saacke, DOAG-Vorstand und Geschäftsführer, ist in Berlin beim Notar, um das neue Vorstandsmitglied Ingo Sobik, Leiter der Next Generation Community, für das Vereinsregister anzumelden.

3. November 2016

Die DOAG lädt die deutschsprachige Fachpresse zur Online-Pressekonferenz ein. Ein Thema ist die Ausrichtung von Oracle in die Cloud, nachdem der Hersteller während der diesjährigen OpenWorld deutlich gemacht hat, dass er am Ausbau seiner Cloud-Strategie festhalten will. Die Zahl der Cloud-Lösungen steigt stetig an, für die Produkt-Entwicklung proklamiert Oracle eine Cloud-First-Strategie. Die DOAG-Experten zeigen die Herausforderungen auf, die die Cloud für deutsche Unternehmen mit sich bringt. Ein weiteres wichtiges Thema ist die Qualität des Oracle-Supports, nachdem bei der DOAG-Umfrage vor zwei Jahren fast jeder zweite Oracle-Anwender mit Qualität, Reaktionszeit und Prozessen des Oracle-Supports unzufrieden war.

11. November 2016

Fried Saacke, DOAG-Vorstand und Geschäftsführer, sowie Ralf Kölling, vom Vorstand beauftragter Organisator des User Group Leaders Summit, treffen letzte Vorbereitungen für das International Oracle User Group Meeting, das am Vortag der DOAG 2016 Konferenz + Ausstellung in Nürnberg stattfindet. Für das Treffen haben sich 24 Anwendergruppen aus Europa, Afrika, dem Mittleren Osten und den USA angemeldet. Die DOAG 2016 Konferenz + Ausstellung erwartet zusammen mit dem Oracle Cloud Day rund 2.500 Teilnehmer und ist damit wieder die größte Oracle-Veranstaltung in Europa.

Das NürnbergConventionCenter, Standort der DOAG 2016 Konferenz + Ausstellung

Dr. Thomas Petrik (Mitte) im Gespräch mit Ingrid Kriegl (rechts) und Klaus-Michael Hatzinger (links)

„Oracle selbst bietet keine preisgünstigen Lösungen an …“

Hochverfügbarkeit ist für kritische Geschäftsprozesse unabdingbar. Ing. Klaus-Michael Hatzinger, Vorstandsvorsitzender der Austrian Oracle User Group (AOUG), AOUG-Vorstand DI Ingrid Kriegl und Wolfgang Taschner, Chefredakteur des Red Stack Magazin, sprachen darüber mit Dr. Thomas Petrik, Leiter Technology Consulting, Sphinx GmbH.

Was bedeutet Hochverfügbarkeit?

Petrik: Streng nach Definition bedeutet Hochverfügbarkeit, dass ein System gegen den Ausfall einer Komponente abgesichert ist, sodass es zu einer festzulegenden Betriebszeit nur minimal ausfällt. Hier gilt es zwei Dinge zu beachten: Es geht nur um den Ausfall einer Komponente, auch wenn theoretisch zwei Komponenten gleichzeitig ausfallen könnten, und für die minimale Ausfallzeit gibt es fest definierte Verfügbarkeitslevel. Für die meisten Anwender bedeutet „eine Komponente“ immer ein Netzteil, eine Disk, eine CPU oder Ähnliches. Eine Komponente kann aber auch ein komplettes Rack, die gesamte Stromzufuhr oder das ganze Rechenzentrum sein. Es gilt also, in jedem Anwendungsfall exakt zu definieren, was man unter Hochverfügbarkeit versteht. Für den Kunden geht es in erster Linie darum, dass seine Applikationen unterbrechungsfrei funktionieren.

Wie viel Hochverfügbarkeit benötigt welcher Geschäftsbetrieb?

Petrik: Es gibt den alten Spruch in der IT: „Machbar ist fast alles, es ist immer nur eine Frage des Geldes“. Ein Tischler beispielsweise kann seine Produkte auch dann herstellen, wenn sein Computer über Stunden ausfällt, eine Apotheke hingegen wird ohne funktionierendes Kassensystem nicht lange das Geschäft aufrechterhalten. Ein Unternehmer muss sich daher genau überlegen, wie lange er sein Geschäft weiterführen kann, wenn das IT-System ausfallen sollte.

Wie lässt sich Hochverfügbarkeit generell umsetzen?

Petrik: Das hängt natürlich in erster Linie vom Umfang der IT ab. Generell wird die IT in unterschiedliche Bereiche wie Netzwerk, Speicher etc. unterteilt, die getrennt voneinander betrachtet und für sich ausfallsicher gemacht werden. Anschließend nimmt man den gesamten Stack unter die Lupe und schaut, ob die einzelnen Schichten ausfallsicher zueinander sind.

Worin besteht der Unterschied zu Disaster Recovery?

Petrik: Hochverfügbarkeit ist kein Schutz beispielsweise gegen einen Stillstand aufgrund eines Datenfehlers, der durch eine fehlerhafte Software entstanden ist. Wenn eine Datenbank zum Beispiel Block-Corruption liefert, kann eine Hochverfügbarkeitsumgebung dies nicht absichern. Dafür benötigt man zusätzlich entsprechende Disaster-Recovery-Szenarien, die beim Backup beginnen oder beispielsweise eine Data-Guard-Lösung beinhalten. Interessant sind auch Varianten, die die Snapshot-Funktionalität von Storage-Systemen oder Hypervisoren mit Flashback Database verbinden, oder das Setup dedizierter Cloning Environments, die sich in weiterer Folge der Pluggable-Database-Funktionalität aus der Version 12*c* bedienen.

Welche Hochverfügbarkeits-Lösungen stellt Oracle seinen Kunden bereit?

Petrik: Im Bereich „Hochverfügbarkeit" steht der Real Application Cluster an erster Stelle. Das System spiegelt sich auch in der Exadata wider. Für Disaster Recovery kommen dann zwei Exadatas zum Einsatz. Die Cloud-Maschine hingegen verfügt über den klassischen Hypervisor-Ansatz auf Basis von Oracle VM.

Wo sollte Oracle noch nachbessern?

Petrik: Mich stört immer schon, dass Oracle hier nur auf seine Großkunden fokussiert ist, weil die Hochverfügbarkeitslösungen immer die Enterprise Edition erfordern. Ein Kunde mit der Standard Edition geht leer aus. Es besteht einfach ein eklatanter Preissprung zwischen der Standard und der Enterprise Edition. Für Kunden mit der Standard Edition müsste es die Möglichkeit geben, bestimmte Optionen zu einem akzeptablen Preis hinzufügen zu können. Ansonsten ist den kleineren und mittleren Anwendern die Möglichkeit zur Hochverfügbarkeit und zum Disaster Recovery mit Oracle-Mitteln verbaut.

Dr. Thomas Petrik

Muss Hochverfügbarkeit teuer sein?

Petrik: Oracle selbst bietet keine preisgünstigen Lösungen an. Hochverfügbarkeit lässt sich aber generell günstig bauen. Dafür

Die Sphinx GmbH

sphinx IT Consulting wurde von DI Ingrid Kriegl und Mag. Friedl Ebner gegründet. Als eigentümergeführtes Unternehmen wird sphinx seit mehr als zwanzig Jahren als Technologieberater für heterogene Infrastrukturen, Experte für Software-Modernisierung und verlässlicher Betriebsführer von Top-Unternehmen in Deutschland, Österreich und der Schweiz geschätzt. Nach dem Motto „Mastering the GAP" optimiert sphinx komplexe IT-Landschaften übergreifend – von der Infrastruktur bis zu den Anwendungen. Inhaltliche Schwerpunkte sind unter anderem Performance, Daten- und Betriebssicherheit sowie Integration smarter Technologien. Der technologische Schwerpunkt liegt vor allem im Bereich „Oracle".

sind verschiedene Technologien auf dem Markt vorhanden, unter anderem auch Oracle VM.

Ist Hochverfügbarkeit in der Cloud ein praktikabler Weg?

Petrik: Im Prinzip bieten alle Cloud-Angebote die Hochverfügbarkeit quasi out of the box. Abgesehen von der Frage, ob ein Unternehmen seine Daten in die Cloud geben will, besteht das große Problem, beispielsweise von Oracle ein entsprechendes Service-Level-Agreement hinsichtlich Hochverfügbarkeit in der Cloud zu bekommen.

Was erwarten Sie von Oracle in der Zukunft?

Petrik: Ich wünsche mir künftig in erster Linie interessantere Lizenzmodelle, die für kleinere und mittlere Kunden attraktiv sind. Außerdem sollte Oracle seine Lizenz-Audits mit mehr sozialer Intelligenz durchführen. Die Kunden haben ein Verständnis für Lizenz-Audits, doch deren Ablauf schürt immer wieder die Emotionen.

Welche Rolle spielt für Sie eine Anwendergruppe wie die AOUG?

Petrik: Die Veranstaltungen der Anwendergruppen sind für mich wichtige Foren für den Austausch auf allen Ebenen. Hinzu kommt, dass Oracle in den Anwendergruppen einen natürlichen Widerpart findet, gerade was die Lizenzierung oder sonstige Marktentwicklungen betrifft. Ich sehe hier ein hohes Vertrauen der Community in die Arbeit der Anwendergruppen.

Zur Person: Dr. Thomas Petrik

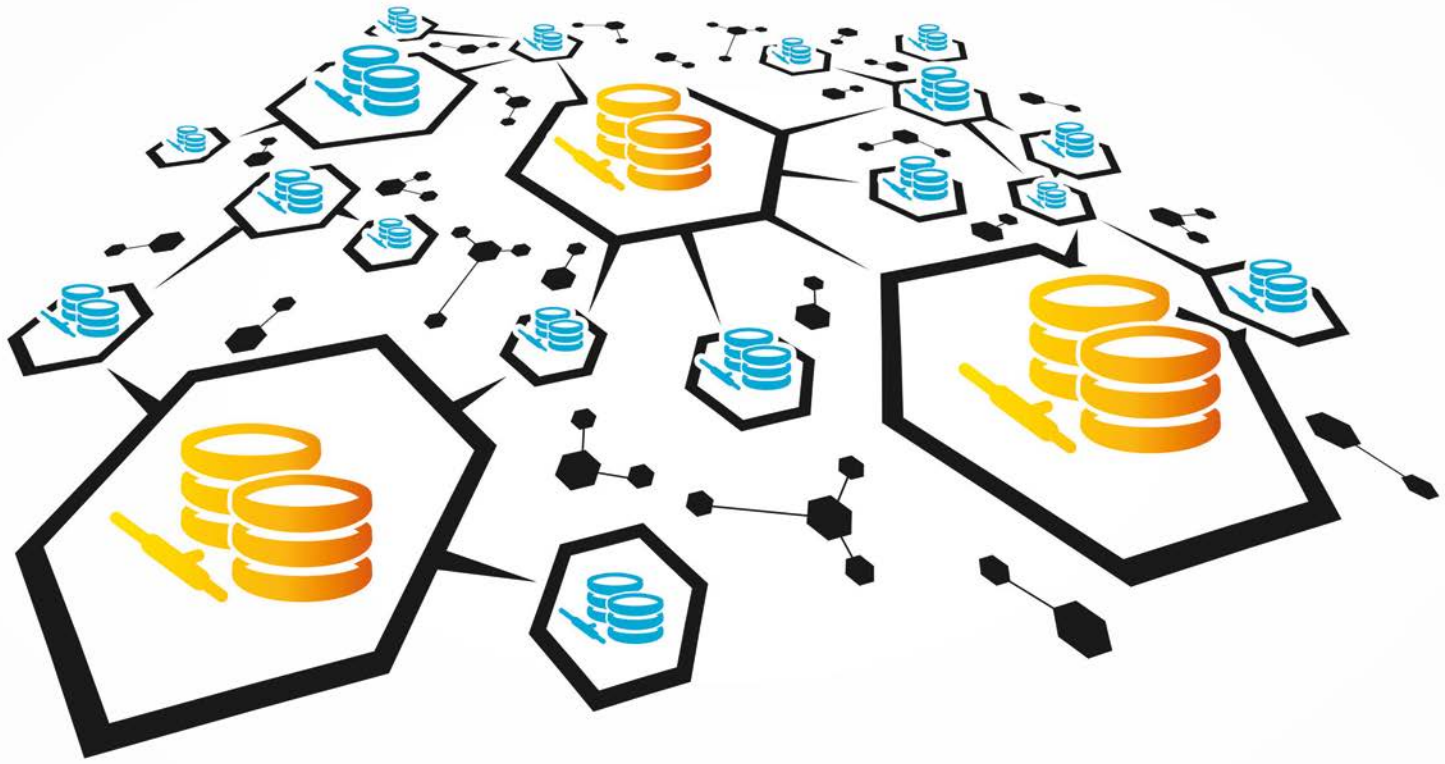
Dr. Thomas Petrik leitet das Technology Consulting bei sphinx und befasst sich seit mehr als fünfzehn Jahren mit Hochverfügbarkeits-Architekturen, Security und Performance-Tuning rund um Datenbanken und Betriebssysteme. Seine Sicht auf die Themen „Hochverfügbarkeit" und „Disaster Recovery" sind vor allem durch seine langjährigen Erfahrungen im Banken-Bereich und im öffentlichen Sektor geprägt. Thomas Petrik hat diese Anforderungen der Enterprise-Kunden in den letzten Jahren zunehmend auch auf mittelständische Betriebe transformiert, was zu einer intensiven Beschäftigung mit kostengünstigeren Infrastrukturen geführt hat, die dennoch dem Anspruch an High Performance und High Availability gerecht werden.

Dr. Jürgen Menge wird DOAG-Botschafter für Technologien

Der Preis „DOAG-Botschafter für Technologien" geht in diesem Jahr an Dr. Jürgen Menge, IT-Consultant bei „Der IT-Macher GmbH". Der ehemalige Oracle-Mitarbeiter hat sich bei der DOAG durch hohes Engagement verdient gemacht.

Erste Bekanntschaft mit Oracle-Technologien hat der gebürtige Erfurter als Berater in einem mittelständischen Münchner Softwarehaus gemacht. Sehr schnell kam er schon in Berührung mit der DOAG: 1993 organisierte er in Absprache mit dem DOAG-Vorstand die ersten Treffen eines Stammtisches mit, aus dem die spätere Regionalgruppe München hervorgeht. Seit dieser Zeit ist der begeisterte Bergsteiger und Skifahrer dem Verein treu geblieben, zuerst als Oracle-Ansprechpartner für die Development-Community der DOAG; seit seinem Ausscheiden aus dem Konzern als Themenverantwortlicher für Oracle Forms. „Die DOAG bietet eine einzigartige Plattform für die Zusammenarbeit und den Erfahrungstausch mit Oracle-Anwendern und Partnern", sagt der Botschafter über den Verein.

Stefan Kinnen (links) überreicht Dr. Jürgen Menge den Pokal



Hochverfügbarkeit ist kein reines Infrastruktur-Thema

Bernd Usinger, GEBHARDT Sourcing Solutions AG

Kritische Geschäftsprozesse brauchen hochverfügbare IT-Systeme. Andernfalls können durch unvorhergesehene Ausfälle immense Schäden entstehen. Aber welche Hochverfügbarkeitsklasse ist die richtige, um einen digitalen Prozess optimal auszurichten? Und wie lässt sie sich erreichen? Intelligente Infrastruktur-Lösungen wie Oracle RAC sind effektive Hilfsmittel, um Unternehmens-Applikationen möglichst ausfallsicher zu gestalten, sie nähern sich dem Thema jedoch allein von der technischen Seite. Das greift schlicht zu kurz: Hochverfügbarkeit braucht ein ganzheitliches Denken.

Die Digitalisierung der Wirtschaft schreitet mit großen Schritten voran – man könnte fast meinen, sie trage Siebenmeilenstiefel: Kaum ein Geschäftsprozess kommt heute noch ohne steuernde IT-Systeme aus, die entscheidend zu seiner Wertschöpfung beitragen. Insbesondere in der Fertigungsindustrie entstehen durch die zunehmend IT-gestützte Vernetzung von Teilprozessen, Maschinen oder ganzen Werken neue Umsatzpotenziale, die über die zukünftige Konkurrenzfähigkeit der hiesigen Industrie entscheiden.

Mit dem Grad der Digitalisierung eines Unternehmens steigt deshalb auch der Anspruch an die Verfügbarkeit seiner Systeme, weil das Schadenspotenzial bei Ausfall einzelner IT-Komponenten inzwischen sprunghaft angestiegen ist. War beispielsweise der Ausfall eines E-Mail-Systems noch vor einigen Jahren auch über mehrere Tage hinweg gut zu verkraften, entsteht Unternehmen heute durch den Ausfall einer solchen IT-Anwendung oder eines ERP-Systems innerhalb weniger Stunden bereits ein riesiger Schaden – nicht selten in existenzgefähr-

dender Höhe. Insbesondere kritische Geschäftsprozesse erfordern deshalb einen verlässlichen IT- Betrieb, der vor riskanten Ausfällen sicher ist.

Das Stichwort lautet „Hochverfügbarkeit". Es bezeichnet die Fähigkeit eines Systems, seinen Betrieb auch dann mit einer hohen Wahrscheinlichkeit aufrechtzuerhalten, wenn eine seiner Komponenten plötzlich ausfällt. Der Begriff findet heutzutage eine fast inflationäre Verwendung. Dabei ist Hochverfügbarkeit nicht gleich Hochverfügbarkeit: Eine Verfügbarkeit von 99 Prozent mag auf den ersten Blick hoch klingen, bedeutet aber für ein System, das rund um die Uhr im Einsatz ist, dass es letztlich pro Jahr immer noch stolze 87,7 Stunden lang ausfallen darf – das sind drei Tage und mehr als 15 Stunden.

Mit jedem Zehntelprozentpunkt steigt die Ausfallsicherheit um 8,76 Stunden pro Jahr. Von Hochverfügbarkeit spricht man allerdings erst ab einem Verfügbarkeitsniveau von 99,99 Prozent, womit die durchschnittliche Ausfallzeit nur noch rund 52 Minuten pro Jahr beträgt. Für die meisten Use Cases ist das ein tragbares Risiko – es gibt aber auch Anwendungsfälle, in denen selbst 52 Minuten Ausfall im Jahr nicht akzeptabel sind, etwa in der Notfallversorgung in Krankenhäusern, in der Luft- und Raumfahrt oder in der Kraftwerk-Steuerung. Hier muss eine Verfügbarkeit von 99,9999 Prozent gegeben sein, damit gewährleistet ist, dass die Funktion des Systems unter allen Umständen verfügbar ist. Aufgrund der vielen Nachkommastellen ist diese Hochverfügbarkeitsklasse im Englischen auch als „six nines" bekannt. Die potenzielle Ausfallzeit lässt sich damit auf eine halbe Minute pro Jahr minimieren.

Je höher die Hochverfügbarkeitsklasse, desto höher die Investition

Dass Hochverfügbarkeit nicht ohne Investitionen zu erreichen ist, liegt auf der Hand. Dennoch sollten auch mittelständische Unternehmen sich davon nicht abschrecken lassen. Denn eine einzige Stunde Server-Ausfall kann im Zeitalter der Digitalisierung mitunter mehr kosten als eine komplette Hochverfügbarkeitslösung. Klar ist allerdings: Je höher die Hochverfügbarkeitsklasse, desto höher die Investitions- und Betriebskosten für ein System. Aus diesem Grund kann es nicht das Ziel sein, jede einzelne Anwendung so verfügbar wie möglich auszulegen. Es gilt vielmehr, die maximal tragbare Ausfallzeit gegen den maximal tragbaren Schaden beziehungsweise Datenverlust abzuwägen, um zu entscheiden, wie verfügbar das jeweilige System sinnvollerweise sein sollte.

Eine gute Entscheidungsgrundlage hierfür liefert die „Availability Environment Classification" (AEC) der Havard Research Group, eine Einteilung in sechs Hochverfügbarkeitsklassen: Die niedrigste Klasse, AEC-0 (Conventional), besagt, dass die Funktion eines Systems unterbrochen werden darf und die Datenintegrität nicht essenziell ist. Auch bei der Hochverfügbarkeitsklasse AEC-1 (Highly Reliable) kann die Funktion unterbrochen werden, die Datenintegrität muss jedoch gewährleistet sein. AEC-2 (High Availability) gibt an, dass die Funktion einer Anwendung nur innerhalb festgelegter Zeiten beziehungsweise zur Hauptbetriebszeit allenfalls minimal unterbrochen werden darf.

Noch weiter geht AEC-3 (Fault Resilient): Anwendungen in dieser Verfügbarkeitsklasse müssen innerhalb festgelegter Zeiten oder in der Hauptbetriebszeit ununterbrochen aufrechterhalten sein. Fehlertolerante Systeme dagegen (AEC-4/ Fault Tolerant) müssen ihre Funktion ununterbrochen halten, sodass ein 24/7-Betrieb gewährleistet ist. An höchster Stelle steht schließlich die Verfügbarkeitsklasse AEC-5 (Disaster Tolerant): Hier muss die Funktion unter allen Umständen verfügbar sein.

Wie aber findet ein Unternehmen nun heraus, welche Verfügbarkeitsklasse es im Einzelfall anstreben sollte? Um diese Frage sinnvoll beantworten zu können, ist an erster Stelle eine professionelle Analyse der betroffenen Geschäftsprozesse gefragt, wie sie ein fachkundiger IT-Dienstleister vornimmt: Welche von ihnen sind tatsächlich kritisch? Welchen Schaden erleidet das Unternehmen bei einem Ausfall der Anwendung? Oder anders gefragt: Wie lange darf die Anwendung im Störfall maximal ausfallen, damit der Schaden noch tragbar ist?

Wenn im Hochfrequenzhandel an der Börse mehrere Millionen Transaktionen pro Minute online durchgeführt werden, ist der Schaden schon bei einem zehnminütigen Ausfall des Systems immens und damit keineswegs tragbar. In anderen Geschäftsfeldern dagegen, beispielsweise im Reporting, entstehen allenfalls Schäden in tolerierbarer Höhe, selbst wenn eine Anwendung einmal vier Stunden lang ausfällt. Hier für teures Geld in die höchstmögliche Verfügbarkeitsklasse zu investieren, steht schlicht nicht im richtigen Verhältnis.

Redundanz kritischer Systemkomponenten ist gefragt

Erst wenn ausgehend vom Geschäftsprozess eruiert wurde, wie ausfallsicher ein System tatsächlich sein muss, steht die Überlegung an, wie die entsprechende IT- Infrastruktur aufgebaut sein muss. In diesem Zusammenhang ist auch zu berücksichtigen, welchen Reifegrad die betroffene Anwendung hat: Soll sie gerade erst eingeführt werden oder ist sie bereits in der Optimierungsphase? Je höher der Reifegrad einer Anwendung ist, desto größer ist auch der Aufwand, um nachträglich eine höhere Verfügbarkeitsklasse zu erreichen.

Bei aller Stabilität und Qualität muss eine jede Unternehmens-IT deshalb zugleich ausreichend flexibel ausgelegt sein, um auf technische Neuerungen, veränderte Regularien oder die Entwicklungen des Marktes binnen kürzester Zeit reagieren zu können. Im Zeitalter der fortschreitenden Digitalisierung von Geschäftsprozessen stehen Unternehmen also vor der Herausforderung, einen Spagat zwischen der hohen Qualität und Verlässlichkeit ihrer IT-Systeme sowie deren Flexibilität schlagen zu müssen.

Hohe Verfügbarkeiten lassen sich im Allgemeinen erreichen, indem sogenannte „Single Point of Failure"-Risiken eliminiert werden. Dahinter verbergen sich diejenigen Systemkomponenten, deren Versagen den Ausfall des gesamten Systems nach sich ziehen würde. Soll eine Anwendung hochverfügbar ausgelegt sein, ist deshalb eine Redundanz aller kritischen Systemkomponenten anzustreben, um damit zugleich die Robustheit und Fehlertoleranz des Gesamtsystems zu erhöhen. Wie lassen sich die betroffenen Anwendungen also fehlertolerant innerhalb der geplanten Architektur abbilden? Wie lässt es sich erreichen, dass eine Applikation auch dann noch verfügbar ist, wenn ein Server oder ein Plattensystem im Storage ausfällt?

Eine verlässliche Lösung liefert das Oracle Real Application Cluster (RAC). Als zusätzliche Option der Oracle-Datenbank ermöglicht es Unternehmen, ihre Datenbanken ausfallsicher einzurichten, indem mehrere Knoten eines Rechnerverbunds auf dieselbe Datenbank zugreifen. Im Falle eines Ausfalls einer dieser Knoten können sich Clients ohne Wiederanlaufzeit über einen der anderen Knoten mit dem System verbinden. Für den Anwender wiederum macht es keinen Unterschied, auf welchem der Knoten sein Zugang erfolgt.

Anders als bei klassischen Failover-Clustern wird die redundante Hardware bei der Oracle-Lösung nicht nur im Fehlerfall eingesetzt, sondern kann auch im Normalbetrieb genutzt werden: Alle Lasten können auf sämtliche Clusterknoten verteilt sein, sodass das RAC nicht nur für eine hohe Verfügbarkeit sorgt, sondern auch sehr flexibel skalierbar ist. Für Unternehmen entsteht dadurch mitunter ein deutlicher Performance-Gewinn.

Hochverfügbarkeit erfordert ein Ende-zu-Ende-Denken

Es ist allerdings zu kurz gedacht, das Thema „Hochverfügbarkeit" allein von Seiten der Infrastruktur anzugehen. Denn es nützt in der Tat wenig, eine Datenbank mit dem Oracle-RAC von technischer Seite her hochverfügbar auszulegen, wenn die Kühlung des Server-Raums bei hohen Temperaturen nicht sichergestellt ist – oder wenn die zuständige IT-Belegschaft um 17 Uhr Feierabend macht. Hochverfügbarkeit betrifft Menschen und Systeme gleichermaßen und sie erstreckt sich über den gesamten Geschäftsprozess. Wer sich allein auf die technische Seite konzentriert, deckt damit nur einen Teilbereich ab und investiert teures Geld, ohne Systemausfälle zuverlässig verhindern zu können.

Stattdessen ist bei der Hochverfügbarkeit – wie so häufig in der Optimierung von Geschäftsprozessen – ein Ende-zu-Ende-Denken gefragt. Dieses lässt sich am besten von außen steuern, etwa durch einen versierten Dienstleistungspartner, der die internen Abläufe von einer externen Perspektive aus betrachten kann. In einem Auftakt-Workshop sollten der beratenden Dienstleister und das betroffene Unternehmen zunächst eine klare Zielsetzung entwickeln, um welche Use Cases es eigentlich geht, welche Anwendungen also überhaupt hochverfügbar ausgerichtet werden sollen. Dazu müssen zunächst einmal die Gefahrenpotenziale definiert werden, die einem Unternehmen durch eventuelle Systemausfälle entstehen können. Über ein fundiertes Risikomanagement lässt sich dann eruieren, welche IT-Abläufe damit innerhalb der Organisation im Einzelnen verbunden sind und auf welcher Rechenzentrums-Infrastruktur sie aufbauen, aber auch, welche räumlichen und personellen Ressourcen es braucht, um die Ausfallsicherheit so weit wie nötig zu minimieren und das System dabei dennoch so flexibel wie möglich zu halten.

Nicht immer ist es dabei sinnvoll, die eigene Infrastruktur kostenintensiv hochverfügbar auszurichten. Es kann unter Umständen besser sein, über eine verlängerte Werkbank im Sinne einer Co-Location oder gar einer Infrastruktur as a Service nachzudenken. Ein erfahrener Sourcing-Partner, der den Markt und seine Preise genau kennt, kann bei der „Make or Buy"-Entscheidung fundierte Hilfestellung leisten.

Fazit

Es ist deutlich absehbar, dass die Hochverfügbarkeit in den kommenden Jahren weiter an Brisanz gewinnt, denn im zunehmend vernetzten Industrie-4.0-Umfeld steigen die Anforderungen an die IT immer weiter. Hochverfügbare Systeme werden für Unternehmen gleich welcher Branche und Größe zum entscheidenden Wettbewerbsfaktor. Wer diese Dringlichkeit versteht und das Thema ganzheitlich denkt, kann mit den rasanten Entwicklungen der Digitalisierung auch in Zukunft Schritt halten.

Bernd Usinger
bernd.usinger@gebhardt-ag.de

Zehn Mal Hochverfügbarkeit

Martin Klier, Performing Databases GmbH

Betrachtungen zu einem alten Buzzword mit den Augen des modernen Datenbank-Administrators

01 Hochverfügbarkeit

Ein System ist hochverfügbar, wenn es mit einer hohen Wahrscheinlichkeit in der Lage ist, den Betrieb auch bei Störungen aufrechtzuerhalten. Schnell geht es dabei um harte Fakten, Zahlen und Service Level Agreements (SLA). So bedeuten die berühmten „five niners" 99,999% Verfügbarkeit. Erstreckt sich der betrachtete Zeitraum auf das volle Jahr, muss das besprochene System bis auf 315 Sekunden (5,26 Minuten) jederzeit betriebs- und leistungsbereit sein.

Da viele Upgrades und Patches einesteils zur Absicherung des Betriebes notwendig sein können, andererseits aber auch oft mit der zeitweisen Abschaltung einer Umgebung einhergehen, entsteht ein Dilemma. Gelöst wird dies in der Praxis unter erhöhtem Einsatz von Ressourcen durch parallel betriebene Umgebungen oder vereinfachend und kostensparend mit einer Reduzierung des für die Verfügbarkeit relevanten Zeitraumes (etwa durch Nichtbetrachtung von Urlaubszeiten).

Es ist nicht automatisch ausgeschlossen, menschliche Eingriffe zur Ermöglichung oder Erhaltung dieser Fähigkeit einzukalkulieren. Oft jedoch beziehen sich entsprechende Angaben auf die autonome Betriebsfähigkeit ohne Eingriffe.

02 Fehlertoleranz

Die einem System eigene Fehlertoleranz beschreibt, wie stark es gegen Bedrohungen seiner Verfügbarkeit abgeschirmt ist. So wird etwa definiert, wie viele Server, Netzwerk-Komponenten oder Storage-Spiegel gleichzeitig ausfallen können, ohne die Verfügbarkeit des Systems einzuschränken.

03 Backup und MTTR

Im Falle eines Desasters – wie Verlust von Nutzdaten oder Konfigurationen – ist für den Wiederanlauf der Umgebung eine unabhängig abgelegte Datensicherung unabdingbar. Ist dies noch der allgemeinen Absicherung zuzurechnen, wird spätestens die Mean Time To Recover (MTTR) relevant für die Beschreibung der Verfügbarkeit. Diese „mittlere Zeit für ein Recovery" bestimmt, wie schnell ein zerstörtes System wieder betriebsfähig ist. Eine verlustfreie Wiederherstellung in möglichst kurzer Zeit erfordert gute Planung von Hardware und Implementierung. Regelmäßige Tests von Konzept und Implementierung sollten eine Selbstverständlichkeit sein: „An untested backup is just a prayer on tape."

04 Faktor „Mensch"

Bei vielen, scheinbar rein technischen Problemen wird der Mensch oft zu schnell als begrenzender Faktor angesehen. In der Praxis aber benötigen zuverlässige und zweckmäßige IT-Anlagen die Betreuung und Pflege durch den kompetenten und engagierten Menschen. Ein elegantes Design wird es verstehen, die Stärken von Automaten und Menschen jeweils auszunutzen und die Schwächen gegenseitig zu kompensieren. Analog zum Betrieb anderer komplexer Maschinen (Beispiel: Flugzeug) sind dabei vor allem die Arbeitsbelastung und die Möglichkeit zur Fehlentscheidung durch den Bediener in Stress-Situationen zu begrenzen.

Um schon bei der Planung die optimale, von menschlichen Fehlern möglichst freie Architektur zu erreichen, ist ein guter Prozess unabdingbar. Vier Augen sollten das absolute Minimum sein und eine Kombination von zeitgemäßen Technologien und bewährten Konzepten zur Anwendung kommen.

05 Features, Komplexität und Beherrschbarkeit

Alle Technologien zur Hochverfügbarkeit, die als Standard-Produkt auf dem Markt platziert sind, müssen dafür ein weites Feld von Anwendungen abdecken. Dies führt unweigerlich zu einer Fülle von Features, die zur Auswahl stehen.

Aber mit jeder Funktion, mit jedem „Gimmick" erhöht sich der Freiheitsgrad für Störungen ebenso wie der für den Nutzen. Je mehr schief gehen kann, desto mehr wird auch schief gehen, dies drückt schon das sprichwörtliche Gesetz von Murphy treffend aus.

Ein High-Availability-System (HA-System) muss einfach zu verstehen und leicht zu beherrschen sein – so lassen sich Fehler in allen Phasen vermeiden: bei Planung, Implementierung, Test, Inbetriebnahme, Operating und auch im Fall der Fälle, wenn es seinen Wert im Ernstfall beweisen muss.

Merke: Komplexität ist der natürliche Feind der Hochverfügbarkeit!

06 Monitoring

Die dauerhafte Betriebsbereitschaft einer hochverfügbaren Umgebung ist ohne fortlaufende, den Menschen entlastende Überwachung nicht zu erreichen. Was nützt das solideste Cluster, die smarteste Replikation oder die verlässlichste Spiegelung, wenn völlig unbemerkt die Ressourcen aufgebraucht sind? Auch dann ist die Verfügbarkeit plötzlich beeinträchtigt und in vielen Fällen provoziert eine überraschende und vermeidbare Notsi-

tuation fehlerträchtige Stress-Entscheidungen.

Das durchdachte und erprobte Monitoring- und Alerting-Konzept ist daher als Grundbaustein der Hochverfügbarkeit zu betrachten: die unter dem Kosten/Nutzen-Aspekt ganz sicher effektivste Methode, die Anwender eines Systems unterbrechungsfrei zu versorgen.

Gewarnt sei erneut vor einem menschlichen Effekt: Fehlalarme oder nichtssagende Meldungen stumpfen den Administrator ab und schaffen mittelbar die noch vor dem Fehlen von Sicherheit gefährlichste Situation – die Illusion von Sicherheit: „Kann gar nichts passieren, wir haben ja ein Monitoring."

07 Applikation

Die Applikation dient vorrangig einem Zweck, den sie gemäß ihrer Spezifikation erfüllen muss, und an dem wird sie gemessen. Leider ist aber die Unterstützung von Hochverfügbarkeit eines der oft vernachlässigten „Soft"-Features, die am Ende über den vollen Erfolg − oder Misserfolg − einer Lösung mitentscheiden. Um für den Betrieb in einer HA-Umgebung fit zu sein, sollte Software einige Mindestanforderungen erfüllen:

- Einfaches, vorhersehbares und transparentes Verhalten der (Netzwerk-) Kommunikation
- Robustes und selbstheilendes Error Handling. Zum Beispiel: abgebrochene Verbindungen automatisch wieder aufbauen, ohne blockierende Fehlermeldungen
- Bei feststehenden Kommunikationspartnern mehrere Ziele unterstützen und Strategie zur Zielauswahl implementieren und dokumentieren (wie Round Robin)
- Hoher Reifegrad, dadurch weniger Fehler
- Optimierter Leistungs-Footprint: Wer (unnötig) viele Ressourcen benötigt, fällt (unnötig) schnell aus, sobald diese einmal knapp werden

Die Implementierung weiterer Features wie die Unterstützung und Behandlung von Events, um auf Veränderungen der HA-Umgebung zu reagieren, sind meist herstellerspezifisch, bei richtiger Umsetzung jedoch sehr effizient.

08 Datenbank

Das relationale Datenbank-System muss für seine Anwender viele Versprechen einhalten, nicht zuletzt ACID. Hochverfügbarkeit ist ein weiterer Eintrag im Lastenheft; es stehen von Oracle und anderen renommierten Herstellern viele verschiedene Technologien zur Verfügung, um die diversen Szenarien abzudecken. Dazu zählen vorrangig:

- Konsistente Online-Backup-Mechanismen, um zur Datensicherung nicht abschalten zu müssen
- Überwachungsschnittstellen, um frühzeitig auf Engpässe oder Ausfälle aufmerksam zu machen
- Replikation, um Teilmengen von Daten logisch auf ein anderes System zu transportieren
- Standby-Datenbanken, um Datenbanken logisch oder physikalisch auf ein anderes System zu transportieren
- Cluster-Technologien, um ausfallende Komponenten automatisch zu ersetzen. Dabei unterscheiden sich die Produkte der großen Datenbank-Hersteller vor allem hinsichtlich möglicher Cache-Kohärenz und damit der Frage, ob alle Knoten gleichwertig Anfragen und Änderungen beantworten können.

09 Infrastruktur

Wo der Unterbau eines hochverfügbaren Systems beginnt, ist reine Definitionssache. In jedem Fall ist es natürlich sinnvoll, dort mindestens gleich hohe Verfügbarkeitsanforderungen zu stellen. Allerdings ist in der Regel eine technische Betrachtung des Gesamtpakets sinnvoll, um mögliche Beschränkungen von Vornherein zu erkennen.

Beim Netzwerk lohnt es sich beispielsweise, die Verfahren zum Load Balancing, Routing und NAT auf Lücken bezüglich

Oracle übernimmt DynDNS-Anbieter

Das amerikanische Unternehmen Dyn sorgt für dynamisches DNS (DynDNS), um Domains im Domain Name System (DNS) dynamisch zu aktualisieren. Dadurch kann ein PC oder ein Router nach dem Wechsel seiner IP-Adresse automatisch und schnell den dazugehörigen Domain-Eintrag einstellen. Mit dieser Technik ist der Rechner immer unter demselben Domain-Namen erreichbar, selbst wenn die aktuelle IP-Adresse für den Nutzer unbekannt ist.

Für Oracle ist der DNS-Betreiber eine sinnvolle Ergänzung für das Internet-as-a-Service- (IaaS) und Platform-as-a-Service-Portfolio (PaaS), wie Thomas Kurian, Präsident der Oracle-Produkt-Entwicklung in einer Pressemitteilung mitteilt. Über den Kaufpreis gibt es keine Angaben.

„Dyns immens skalierbares und globales DNS ist eine kritische Kernkomponente und eine natürliche Erweiterung unserer Cloud-Computing-Plattform", erklärt Kurian. Auch andere Cloud-Anbieter betreiben eigene DNS-Dienste. Dyn hat nach Angaben von Oracle etwa 3.500 Kunden, darunter große Unternehmen wie Netflix, Twitter, Pfizer und CNBC. Der DNS-Dienst wurde im Oktober dieses Jahres Opfer einer DDoS-Attacke, die mehrere Dienste lahmgelegt hat.

Weitere Informationen unter *https://www.oracle.com/corporate/acquisitions/dyn/index.html*

der Verfügbarkeitsplanung zu untersuchen und die HA-Funktionen des Netzes kennenzulernen. So brauchen Lösungen wie STP, BGP, OSPF oder EIGRP regelmäßig einige Zeit, bis das Netz nach Ausfällen wieder funktionsbereit ist und Pakete mit der erforderlichen Latenz und Qualität zur Verfügung stehen. Diese sogenannten „Konvergenzzeiten" von Routing- und Switching-Konzepten mit den angestrebten maximalen Ausfallzeiten der Applikation oder Datenbank zu vergleichen, ist wärmstens zu empfehlen.

Im Bereich „Massenspeicher" bietet der Markt eine nahezu unüberschaubare Fülle an hochverfügbaren SAN- und Storage-Lösungen. Wird bei einem Ausfall „zero data loss", also verlustfreies Weiterarbeiten gewünscht (was in der Regel der Fall sein dürfte), müssen Schreib-Vorgänge an allen Zielen angekommen sein, bevor der darauf aufbauende nächste Schritt beginnt. Damit kommen grundlegend zwei Architekturen in Betracht:

- *Stern*
  Server schreibt auf zwei oder mehrere Massenspeicher (Auswirkung auf Latenz: Langsamster Pfad definiert die Antwortzeit)
- *Kette*
  Server schreibt auf ein Storage-System und dieses repliziert auf ein weiteres etc. (Summe der Pfade definiert die Antwortzeit)

Hier definiert die Qualität der Integration zwischen Hardware-Lösung, Betriebssystem und HA-Software entscheidend die Eigenschaften des Gesamtsystems. Umschalt- oder Erholungszeiten nach Ausfällen und das Lastverhalten bei Schäden (wie Degraded Performance eines RAID- Arrays) sind sorgfältig zu betrachten, zu beeinflussen oder mindestens zu berücksichtigen.

10 Cloud

Die Cloud ist in aller Munde – und doch ist ein Cloud-Service nichts anderes als ein herkömmliches IT-System, das sich in der Hoheit eines anderen befindet und das idealerweise eine Form von Self-Service-Portal anbietet. In der Praxis ist damit der gesamte Technologie-Stack vom Metall bis zur Applikation inklusive Operating der Infrastruktur zuzurechnen. Betrachtet man nun die Cloud unter dem Aspekt der Hochverfügbarkeit, so ergibt sich die Notwendigkeit, die eigenen Anforderungen zunächst in Form von Service Level Agreements (SLAs) zu formulieren. Danach erst kann ein Cloud-Anbieter diese umsetzen und garantieren.

Auch rein technisch stellen sich Herausforderungen. So ist mindestens die Verfügbarkeit der Anbindung an den Cloud-Service zu prüfen: Wie hochverfügbar ist die Anbindung an Internet beziehungsweise Weiterverkehrsnetz, das die User mit dem Backend verbindet? Kein kleines Problem ist die sogenannte „letzte Meile" und die gegebenenfalls erforderliche Anbindung durch zwei oder mehrere Internet Service Provider.

Durch SLAs definierte, harte Schnittstellen zum Cloud-Service-Provider stellen sich in der Praxis oft als hinderlich heraus: Ist der Cloud-Service einmal nicht wie vereinbart verfügbar, erhält man den Schaden zwar (hoffentlich ohne Anwalt und Richter) ersetzt, „down" und zur Untätigkeit verurteilt ist man zunächst trotzdem. Verschiedene Cloud-Anbieter redundant zu nutzen, ist dahingehend die große Herausforderung, der sich Architekten und Planer zu stellen haben. Es ist für jeden Fall sorgfältig zu prüfen, ob der Dienst in der Cloud gut aufgehoben ist oder ob sich das geforderte Level der „End-to-end-Hochverfügbarkeit" nicht im eigenen Rechenzentrum („on premise") sicherer und kostengünstiger umsetzen lässt.

Martin Klier
martin.klier@performing-db.com

Korrektur für die letzte Ausgabe

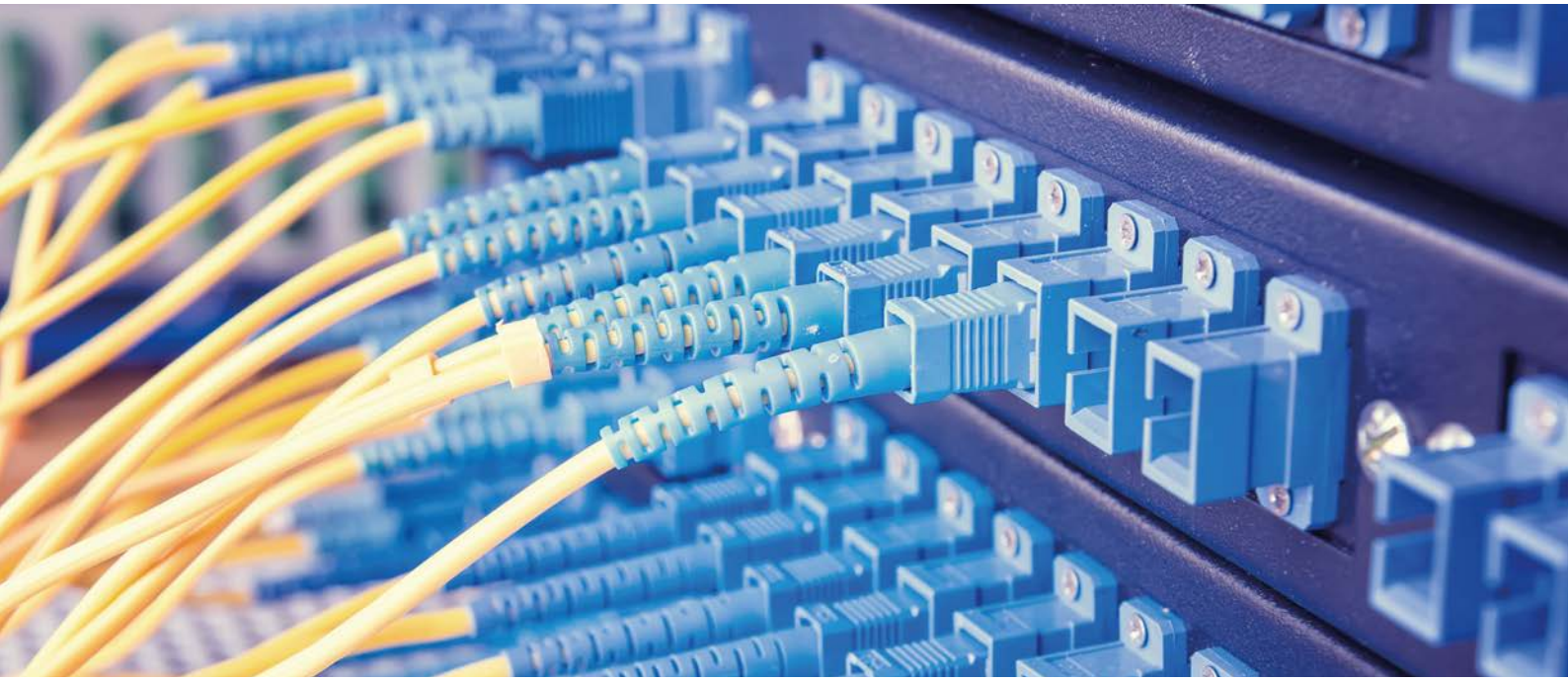
In der letzten Ausgabe habe wir leider ein Autorenfoto falsch zugeordnet. Wir bitten das Versehen zu entschuldigen. Hier die beiden richtigen Fotos.

Seite 24

Simon Hahn
simon.hahn@opitz-consulting.de

Seite 36

Rainer Willems
rainer.willems@oracle.com



Hochverfügbarkeit oder die Suche nach den 100 Prozent Netzwerk-Sicherheit

circular Informationssysteme GmbH

Laut einer weltweit durchgeführten Studie betragen die Kosten für einen IT-Ausfall durchschnittlich 555.000 US-Dollar. Das kann neben Umsatz-Einbußen und Reputationsverlust im schlimmsten Fall den Ruin des Unternehmens bedeuten. Darum gilt für ein hochverfügbares IT-Netzwerk, Single Points of Failure (SPoF) in den einzelnen Netzwerk-Komponenten zu vermeiden, also die Schwachpunkte im System, die bei einer Fehlfunktion zum Ausfall führen. Sowohl aktive Komponenten (Firewall, Router, Switch etc.) als auch passive Komponenten (Patchpanel, Verkabelung, Stecker, Glas oder Kupfer etc.) sind in diesem Zusammenhang zu beachten. Das Thema „Hochverfügbarkeit" in einem lokalen (Local Area Network, LAN) oder weiter ausgelegten Netzwerk (Wide Area Network, WAN) sollte dabei nicht nur schichtweise, sondern im Ganzen betrachtet werden.

Ein Faktor ist entscheidend, um überhaupt eine der hohen Verfügbarkeitsklassen (ab 99,9 Prozent, was immerhin noch eine Ausfallzeit von acht Stunden und 45 Minuten im Jahr entspricht) erreichen zu können. Das Ziel muss sein, eine durchgehende, systemübergreifende Redundanz in Hardware, Software, Anwendungsmanagement etc. aufzubauen, also mindestens die doppelte Ausführung jeder Systemkomponente beziehungsweise jedes Daten-Transportwegs zu garantieren. Eine aktuelle ECC-Studie mit deutschen Unternehmen zeigt die Schäden, die durch den Komplettausfall von IT- und Netzwerkkomponenten entstehen können: Bei bis zu einem Tag Stillstand gehen bereits mehr als zwanzig Prozent der Befragten von einer Schadenshöhe von über 20.000 Euro aus *(siehe „https://de.statista.com/statistik/daten/studie/150900/umfrage/geschaetzte-schadenshoehe-im-betrieb-bei-it-ausfall-in-deutschland")*.

Hochverfügbare Netzwerke: Schicht für Schicht weniger Ausfallrisiko

Die Hochverfügbarkeit im LAN oder WAN ist eine sehr komplexe Angelegenheit, die verschiedene Bereiche in der Netzwerk-Architektur betrifft. Die folgenden Ausführungen orientieren sich am „Open Systems Interconnection"-Modell (OSI). Es besitzt einen hierarchischen Aufbau, der die Kommunikation zwischen den verschiedenen

Systemen, Geräten, Clients und Hosts über Protokolle strukturiert und regelt. Dabei sind sieben OSI-Schichten definiert. Im LAN- oder WAN-Umfeld kommen aber nur die ersten drei Layer in Betracht:

- *Layer 1 (Physical Layer)*
  Beschäftigt sich primär mit der physischen Infrastruktur und Hardware-Redundanz
- *Layer 2 (Data Link Layer)*
  Steht für die Sicherungsebene, die eine geschützte Übertragung der Daten gewährleisten soll
- *Layer 3 (Network Layer)*
  Ist die Vermittlungsschicht; sie schaltet Verbindungen über Leitungen frei oder vermittelt Datenpakete an den richtigen Empfänger

Die erste Schicht – doppelt hält besser

Die physikalische Schicht steht ganz am Anfang jedes modernen, hochverfügbaren IT-Netzwerks. Sie sorgt dafür, dass einzelne Datenpakete, Symbole oder Bits elektrisch, mechanisch, per Schall etc. übertragen werden können. Durch eine physikalische Redundanz versuchen Netzwerk-Experten, die Funktion aufrechtzuerhalten. Eine Möglichkeit bietet hier das Link-Aggregation-Verfahren, mit dem mehrere physische LAN-Schnittstellen zu einem logischen Kanal definiert werden und diesen damit hochverfügbar machen.

Die zweite Schicht – die richtige Weichenstellung entscheidet

Im Data-Link-Layer beziehungsweise in der Sicherungsschicht geht es um das Thema „Switching". Layer-2-Switches funktionieren wie mechanische Weichen, sind hardwarebasiert (dadurch sehr schnell) und verbinden verschiedene Netzwerk-Komponenten über die entsprechenden angesteuerten Ports miteinander. Die Switches entscheiden über den Weg, den ein Datenframe durch das Netzwerk nimmt. Sie bestehen aus Ziel- und Quell-Adressen (MAC-Adressen), aus Steuer-Informationen zur Datenfluss-Steuerung, Nutzdaten des Pakets der Vermittlungsschicht und Prüfsummen zur Gewährleistung der Daten-Integrität. Auch in dieser Schicht ist auf Redundanz zu achten.

Die dritte Schicht – alles eine Frage der Route

Netzwerke unterscheiden sich in den Bereichen „Core" (Backbone), „Distribution" und „Access". Im Access-Bereich, der noch zu Layer 2 gehört, existiert eine hohe Portdichte. An diesen Ports hängen die verschiedenen Arbeitsplätze, die aber meist nicht redundant angebunden sind, da der Ausfall eines Rechners die Funktionsweise des Netzwerks nicht beeinträchtigt. Allerdings kann der Ausfall eines Servers schwerwiegende Folgen haben. Deshalb sind sie beispielsweise über ein Port-Bündelungsverfahren oder Link Bundling stets redundant aufgebaut.

Entscheidend für Layer 3 beziehungsweise die Vermittlungsschicht ist das IP-basierte Routing: Wenn ein Client oder Computer-Programm eines Endgeräts ein Datenpaket an eine bestimmte IP-Adresse versenden will, wird es zunächst an einen Router übermittelt. Dieser prüft dann, ob er eine Route kennt, die zu dieser IP-Adresse führt. Meist gibt es in dieser Schicht auch eine Default-Route ins Internet.

Redundancy-Protokoll – nur ein Geräte-Setup ist hochverfügbar

Das Routing-Verfahren lässt sich um Redundancy-Protokolle erweitern. Sind Übertragungsrouten überlastet oder fehlerhaft, ermitteln sie einen Alternativweg, der durch eine entsprechende Netz-Infrastruktur und redundante Komponenten bereits angelegt ist.

Technisch funktionieren Redundancy-Protokolle nach folgendem Schema: Jedes der beiden gedoppelten Geräte erhält eine eigene IP-Adresse. Eine dritte virtuelle IP-Adresse dient als Gateway für das Datenpaket. Sie entscheidet in einem Fail-Over-Szenario, an welches Gerät das Paket schließlich geht. Dabei gibt es zwei Device-Setups: Es existieren ein aktives und ein passives Gerät, wobei letzteres nur bei Ausfall des bisher aktiven Geräts einspringt.

In der zweiten Variante sind beide Geräte aus Load-Balancing-Gründen aktiv und teilen sich damit die Paket-Übermittlung untereinander auf. Dieses Konzept ist technisch und finanziell sinnvoll, da so die relativ teuren Switch-Ports und deren verfügbare Bandbreite effizient sowie ressourcenschonend genutzt werden können. Allerdings hat diese Variante den Nachteil, dass sich im schlechtesten Fall die Last des Traffic-Aufkommens von einem einzelnen Gerät nicht mehr abfangen lässt. Diese „Überbuchung" kann zu Betriebs-Einschränkungen beispielsweise bei der Bereitstellung von Bandbreite bis hin zum Ausfall eines Dienstes führen. Aus der Hochverfügbarkeitsperspektive ist deshalb das erste Setup zu empfehlen.

IT- und Netzwerk-Sicherheit

Netzwerke bestehen aus verschiedenen Komponenten wie Switch, Router, Load Balancer, Firewall und Intrusion Prevention System (IPS) sowie Intrusion Detection System (IDS) und können unterschiedlich aufgebaut sein. Im Hinblick auf das Thema „Hochverfügbarkeit" spielt dabei der Schutz des Netzwerks eine entscheidende Rolle. Hier beschäftigt sich die Netzwerk-Sicherheit grundsätzlich mit allen Maßnahmen zur Planung, Ausführung und Überwachung der Sicherheit in Netzwerken, was sowohl technologische Maßnahmen, die redundant vorliegen sollten, als auch organisatorische Schritte umfasst, wie etwa Mitarbeiterschulungen und Compliance-Regeln.

Ein Teil ist die Identifikation und Abwehr der verschiedenen Angriffe auf die eingesetzten Komponenten/Protokolle. Beispiele dafür sind etwa Man-in-the-Middle-Angriffe, die die Kommunikation zwischen Client und Zielsystem mitlesen oder beispielsweise durch falsche Routen (Route Poisoning) Verkehr umleiten und kompromittieren, oder Tunneling, mit dem sich Firewalls umgehen lassen. Um eine unerlaubte Ressourcen-Nutzung zu erkennen und zu verhindern, werden Maßnahmen zur Authentifizierung, Autorisierung und Identifikation eingesetzt.

Firewalls sind obligatorisch

Laut einer globalen Studie kosten Cybersicherheitsvorfälle große Unternehmen durchschnittlich über eine halbe Million US-Dollar, kleine und mittelständische Firmen müssen im Schnitt mit rund 38.000 US-Dollar rechnen *(siehe „http://media.kas-*

persky.com/pdf/it-risks-survey-report-cost-of-security-breaches.pdf"). Deshalb sollte das Primärziel jedes IT-Verantwortlichen sein, das eigene Netzwerk möglichst sicher gegen externe und interne Cyber-Bedrohungen abzuschotten. Eine der immer noch entscheidenden State-of-the-Art-Sicherheitsmaßnahmen für Netzwerke sind Firewalls, die es hardwarebasiert in der Vermittlungsschicht und softwarebasiert in der Anwendungsschicht gibt.

Firewalls zählen heute wie Daten-Backup-Systeme zum Sicherheitsstandard jedes Unternehmens, so eine aktuelle Studie *(siehe „https://www.bundesdruckerei.de/digitalisierung/system/files_force/bundesdruckerei_studie_zu_it-sicherheit_und_digitalisierung.pdf")*. Doch im Dark Net finden sich immer neue Gefahren, wie etwa Baukästen für Spionage-Software oder Advanced Persistent Threats (APT), die in komplexen, mehrstufigen Attacken ins Netzwerk eindringen und oft erst zu spät oder gar nicht bemerkt werden. Ziel dieser Angriffe sind meist unternehmenskritische Informationen.

Heute benötigt es jedoch eine modernere Netzwerk-Architektur, da sich die Anforderungen an Firewalls durch die Zunahme von mobilen Endgeräten, Anwendungen, Homeoffice etc. geändert haben. Erlaubt ein Unternehmen unter anderem das Surfen im Internet, lässt sich dafür über herkömmliche Firewalls nur eine pauschale Freigabe erteilen. Deshalb stellt der Aus- und Eingang zum Internet, der bei den meisten Netzwerken über Port 80 oder bei verschlüsselten Webseiten über Port 443 erfolgt, einen besonders beliebten Angriffspunkt für Schadsoftware und Hacker dar. Hat ein Anwender, der immer noch das größte Sicherheitsrisiko für ein Netzwerk darstellt, beispielsweise versehentlich einen Trojaner, Wurm etc. installiert, versucht das Schadprogramm, die erbeuteten Informationen gegebenenfalls über diese beiden Ports aus dem Netzwerk zu schleusen.

Dagegen findet bei den Next-Generation-Firewalls eine tiefergehende Analyse des Traffics beziehungsweise der Datenpakete statt. Zu den herkömmlichen Funktionalitäten kommen IDS, IPS, QoS-Maßnahmen (Quality of Service), Applikationskontrollen, SSL- und SSH-Inspection, Deep Packet Inspection, reputationsbasierte Malware-Abwehr und Application Awareness etc. hinzu. Dabei werden unter anderem Datenpakete und Verbindungen genauer geprüft, etwa daraufhin, welche Anwendungen sowie Dienste sich dahinter verbergen und angesteuert werden.

Vor allem die applikationsspezifischen Abwehrmaßnahmen kümmern sich um die steigende Anzahl von Angriffen auf den Ebenen vier bis sieben des OSI-Stacks. Der Aspekt „QoS" kommt besonders bei verstärkt genutzten Applikationen wie VoIP oder Video-Konferenzsystemen zum Tragen. Deren ausfallsicherer Betrieb kann beispielsweise durch Maßnahmen wie Bandbreiten-Regulierung oder der Priorisierung der Applikationen bei der Bandbreiten-Bereitstellung zentral gemanagt werden.

Mit Next-Generation-Firewalls lassen sich Internet-Anwendungen managen,

erlauben und sperren, wie etwa die Facebook-Nutzung oder ein Freeware-Download. Dadurch werden das Herunterladen von Schadprogrammen oder der Aufbau einer unsicheren Schatten-IT an der eigenen IT-Abteilung vorbei verhindert.

Im Zusammenhang mit Next-Generation-Firewalls steht ebenfalls das Thema „Unified Threat Management" (UTM). UTM-Lösungen lassen sich als Firewall-Appliances definieren und stehen für ein multifunktionales Sicherheitskonzept, mit dem sich unternehmensspezifische Sicherheitsstrategien im Unternehmensnetz durchsetzen lassen. Ein Vorteil ist, dass Sicherheitstechniken und -funktionen, die bisher auf verschiedene Systeme verteilt waren, nun in einer Lösung vereint vorliegen. Damit sollen Angriffe, Datendiebstähle, Spams, Viren und Würmer, trojanische Pferde und andere Attacken kontrollierbar werden.

Zu den UTM-Features und -Schutzmechanismen gehören unter anderem die Content- und Spam-Filterung, Applikations- und Web-Kontrolle sowie Einbruchserkennung und Antiviren-Aktivitäten. Zudem lässt sich ein Security- und Policy-Management für Gruppen oder Anwender definieren. UTM-Lösungen sollen vor allem die Netzwerksicherheit zentral steuerbar machen und die Applikations-Schicht vor den Next-Generation-Bedrohungen schützen, ohne die Performance oder das Netzwerk zu belasten.

Mit Systemen für das Security Information and Event Management (SIEM) lässt sich der Traffic in Echtzeit überwachen. SIEM-Appliances wie etwa IBM QRadar oder Splunk sammeln und indexieren dabei Logs oder Ereignisse beziehungsweise Events, korrelieren diese und werten sie aus. Damit lassen sich anomales Anwenderverhalten, Eindringlinge und komplex vorgehende Malware auch innerhalb des Netzwerks identifizieren und gegebenenfalls sofort Gegenmaßnahmen einleiten.

Die Informationen betreffen das Zugangsmanagement, das Schwachstellen-Management und Compliance-Tools, Betriebssysteme, Datenbanken und Logfile-Analysen. Aus den gesammelten Ereignis-, Bedrohungs- und Risiko-Daten lassen sich Maßnahmen für die adaptive Verwaltung von Sicherheitsrisiken ableiten. Sie basieren auf unternehmensspezifischen Anforderungen – also auf klaren und umfassenden Definitionen, welche Ereignisse sicherheitsrelevant sind und wie mit welcher Priorität darauf zu reagieren ist. Das Ziel ist, anhand eines Regelwerks kontinuierlich die Standards für Sicherheit, Compliance und Qualität des IT-Betriebs zu verbessern.

Empfehlungen für ein modernes, hochverfügbares Netzwerk

Um ein möglichst hohes Maß an Hochverfügbarkeit und Netzwerk-Sicherheit zu schaffen, sollten grundsätzlich alle Komponenten im Netzwerk redundant aufgebaut sein. So lassen sich mögliche SPoF weitestgehend ausschließen. Damit Unternehmen auf heutige sowie künftige Herausforderungen angemessen reagieren können, müssen Netzwerke der nächsten Generation Aspekte wie Flexibilität, Netzwerk-Intelligenz und eine verteilte Steuerung in sich vereinen. Sich stetig verändernde IT-Landschaften und geschäftliche Anforderungen verlangen dabei dynamische und schnell anpassbare Systeme.

Durch die Fokussierung auf ein eventuell Cloud-optimiertes, skalierbares und adaptives Netzwerk können Netzwerk-Betreiber die Einschränkungen bewältigen, mit denen sie konfrontiert werden. Cloud-Services ermöglichen eine flexible Nutzung der IT-Ressourcen. Gleichzeitig stellen sie aber auch ein Sicherheitsrisiko dar, weil die Netzwerk-Sicherheit in fremde Hände gegeben wird. Wenn man jedoch von Anfang an den Sicherheitsaspekt berücksichtigt – und hier die Sicherheit des Dienstes und der Verwaltung –, können sich deutsche Unternehmen für Cloud-Dienste entscheiden, die sich in die bestehende IT-Landschaft integrieren lassen. Zudem sollten sie zumindest kritische Unternehmensdaten auf dem eigenen Firmenserver belassen und nur Cloud-Provider wählen, die den deutschen Datenschutz-Richtlinien unterliegen.

Ein anderer wichtiger Punkt ist die Einbindung sowohl lokaler als auch auswärts tätiger Mitarbeiter. Dank moderner und zentraler Netzwerksystem-Lösungen lassen sie sich zuverlässig in das Netzwerk integrieren. Die Systeme sind schnell und einfach an aktuelle sowie kommende Business Cases anpassbar. So können Mitarbeiter unternehmensweit auf dieselben Geschäftsanwendungen und -dienste zugreifen – mobil oder stationär. Weitere Vorteile sind:

- Nahtlose und sichere Vernetzung von Mitarbeitern, Kunden und Informationen
- Bestmögliche Nutzung qualitativer Echtzeit-Anwendungen wie ERP-, Audio- oder Video-Systeme
- Zugriff auf Dateien und Ressourcen – jederzeit und von überall
- Senkung der Betriebskosten
- Unterstützung nachhaltiger Geschäfts-, IT- und Netzwerkprozesse

Fazit

Die wichtigsten Bestandteile eines modernen, hochverfügbaren Netzwerks sind diese:

- Redundant aufgebaute Netzwerk-Komponenten
- Mindestens zweistufiger Firewall-Cluster von verschiedenen Herstellern (Diversivität)
- Redundant ausgelegte Internet- oder WAN-Verbindung
- UTM- und SIEM-Appliances für die Identifikation von Anomalien im Netzwerk
- Monitoring-Lösungen, um Netzwerke und Dienste zu überwachen, zu überprüfen und sich per Alerts automatisiert einen Ausfall anzeigen zu lassen (etwa ob ein Webserver oder eine Datenbank überhaupt noch verfügbar ist)
- Reporting-Lösungen, um über einen längeren Zeitraum beurteilen zu können, ob die bisherige Netzwerk-Infrastruktur oder die Bandbreite noch ausreicht oder ob skaliert werden muss
- Quality-of-Service-Funktionen und -Vorgaben: Dadurch lassen sich beispielsweise Applikationen (wie etwa die Bandbreiten-Nutzung) priorisieren und gegebenenfalls einschränken
- Out-of-Band-Management: ein Managementnetz getrennt vom Datennetz
- Policy based Routing/Forwarding: Damit können Daten bestimmt werden, die nicht innerhalb des normalen Datenstromes geroutet werden, sondern einen gesonderten Weg nehmen, etwa über eine zweite Internet-Anbindung

circular Informationssysteme GmbH
info@circular.de

Aufbau einer 12*c*-RAC- und Data-Guard-Umgebung mit NFS Storage bei der DEVK

Johannes Ahrends, Carajan DB, und Tim Hensel, DEVK Versicherungen

Im Rahmen eines Projekts wurden die vertriebsunterstützenden Systeme der DEVK-Versicherungen auf eine „always online"-Lösung ausgerichtet. Als Basis sollte eine neue, stabile sowie hochverfügbare Oracle-Infrastruktur dieser Anforderung entsprechen. Das Ziel-Design wurde zudem zugunsten einer schnellen Wiederherstellbarkeit und einer möglichst unterbrechungsfreien Wartbarkeit erarbeitet.

Vor der endgültigen Architektur-Entscheidung der zugrunde liegenden Infrastruktur wurden diverse Szenarien zur Realisierung detailliert betrachtet, gewichtet und anschließend bewertet. Dazu gehörten neben Virtualisierungs-Lösungen mit VMware oder Oracle VM auch die Oracle Engineered Systems ODA und Exadata.

Letztendlich fiel die Entscheidung – nicht zuletzt aufgrund der zu diesem Zeitpunkt teilweise unklaren Lizenz-Politik im VMware-Umfeld und der damit verbundenen Risiken – auf eine Hardware-Lösung mit Cisco UCS Blades.

Die anfängliche Idee, hierauf ein sich über beide Rechenzentren erstreckendes Oracle-Stretched-Cluster zu realisieren, wurde verworfen, da sich im Laufe des Projekts im Hinblick auf die Hochverfügbarkeit der Applikationsserver frühzeitig für ein NetApp-Metrocluster im Master-Slave-Modus entschieden wurde. Dies hätte im Fehlerfall ein manuelles Umschalten des zugrunde liegenden Storage bedeutet.

Somit sollte die Verfügbarkeit der drei benötigten Datenbanken durch die Oracle-eigenen HA-Komponenten RAC in Kombination mit Data Guard abgebildet werden (Maximum Availability Architecture). Realisiert wurde das Ganze in der Produktionsumgebung durch jeweils zwei physikalische Server im RAC-Verbund, wobei die beiden RZ-Standorte über Data Guard miteinander verbunden wurden *(siehe Abbildung 1)*. Von Active Data Guard und Fast Start Failover wurde abgesehen; von letzterem primär aufgrund des fehlenden dritten Standorts für den Observer.

Um einen ausgereiften Staging-Prozess realisieren zu können, gibt das Betriebskonzept der DEVK vor, einer produktiven Umgebung technisch gleichartige Systeme

in anderen Ebenen voranzustellen. Somit galt es, zusätzlich eine Vorproduktions-, eine Entwicklungs- und eine sogenannte „Maintenance"-Umgebung aufzubauen. Letztere kann als reine „Spielwiese" für Oracle-DBAs bezeichnet werden, auf der beispielsweise das Einspielen von Patches als Erstes vorgenommen wird, bis diese dann sukzessive von Stage zu Stage bis in die Produktion eingerichtet werden.

Einzig in der Entwicklung wurde ursprünglich auf die HA-Realisierung durch RAC und Data Guard verzichtet, um Lizenzen einzusparen. In Summe wurden also dreizehn Oracle-Datenbank-Server (vier für Maintenance, einer für Entwicklung, vier für Vorproduktion und vier für Produktion) mit jeweils mindestens drei Datenbanken aufgebaut, ohne hierbei auf die neue Multitenant-Architektur zurückzugreifen.

Bei der Auswahl des zugrunde liegenden Filesystems für die Real Application Cluster hat man sich bewusst nicht für das bei der DEVK bisher eingesetzte und bei den DBAs stets als sehr zufriedenstellend und stabil angesehene ASM, sondern für NFS entschieden. Grund ist die Vorgabe der geringen Wiederherstellungszeiten und die damit verbundene Anforderung, die bis zu 1TB großen Datenbanken im Bedarfsfall möglichst schnell recovern zu können. Dieser Herausforderung konnte man durch Einsatz der Snapshot-Technologie, die wiederum NFS-Shares voraussetzt, problemlos gerecht werden, was sich im späteren Verlauf des Projekts bestätigen sollte. Von der Alternative, ASM auf NFS zu implementieren, wurde abgesehen; man entschied sich hingegen für den Einsatz von direct NFS (dNFS) für Oracle12*c*.

Als Sicherungstool sollte das von NetApp Anfang dieses Jahres veröffentlichte Plug-in des SnapCenter für Oracle-Datenbanken dienen. Zu Beginn des Projekts nahm man an einem Beta-Programm dieser neuen Komponente für das SnapCenter teil, wobei selbige ausführlich geprüft wurde und anfängliche Fehler im direkten Austausch mit NetApp ausgemerzt werden konnten.

Filesystem-Architektur der Datenbanken

Auf den Datenbank-Servern wurde Oracle 12.1.0.2 auf SUSE Linux Enterprise Server 11 mit Service Pack 4 installiert, wobei diese Konstellation der Versionen primär den recht strengen Release-Vorgaben des SnapCenter geschuldet war. Zudem ist SUSE die strategische Linux-Distribution der DEVK.

Die Basis-Installation und -Konfiguration der Serversysteme erfolgte mithilfe der Plattform „Ansible". Durch diese Automatisierungsmethode ist sichergestellt, dass sich die Konfigurationen – angefangen von OS-Usern und -Gruppen über Kernel-Parameter bis hin zu Filesystemen – auch im kleinsten Detail nicht unterscheiden. Später wurden zusätzlich definierte Checks durch das Tool „ServerSpec" durchgeführt, um auch zukünftig Stage-übergreifend vor ungewünschten Änderungen bewahrt zu bleiben.

Bei der Verzeichnisstruktur orientierte man sich an der Oracle Flexible Architecture und entschied sich für die Installation der GI-

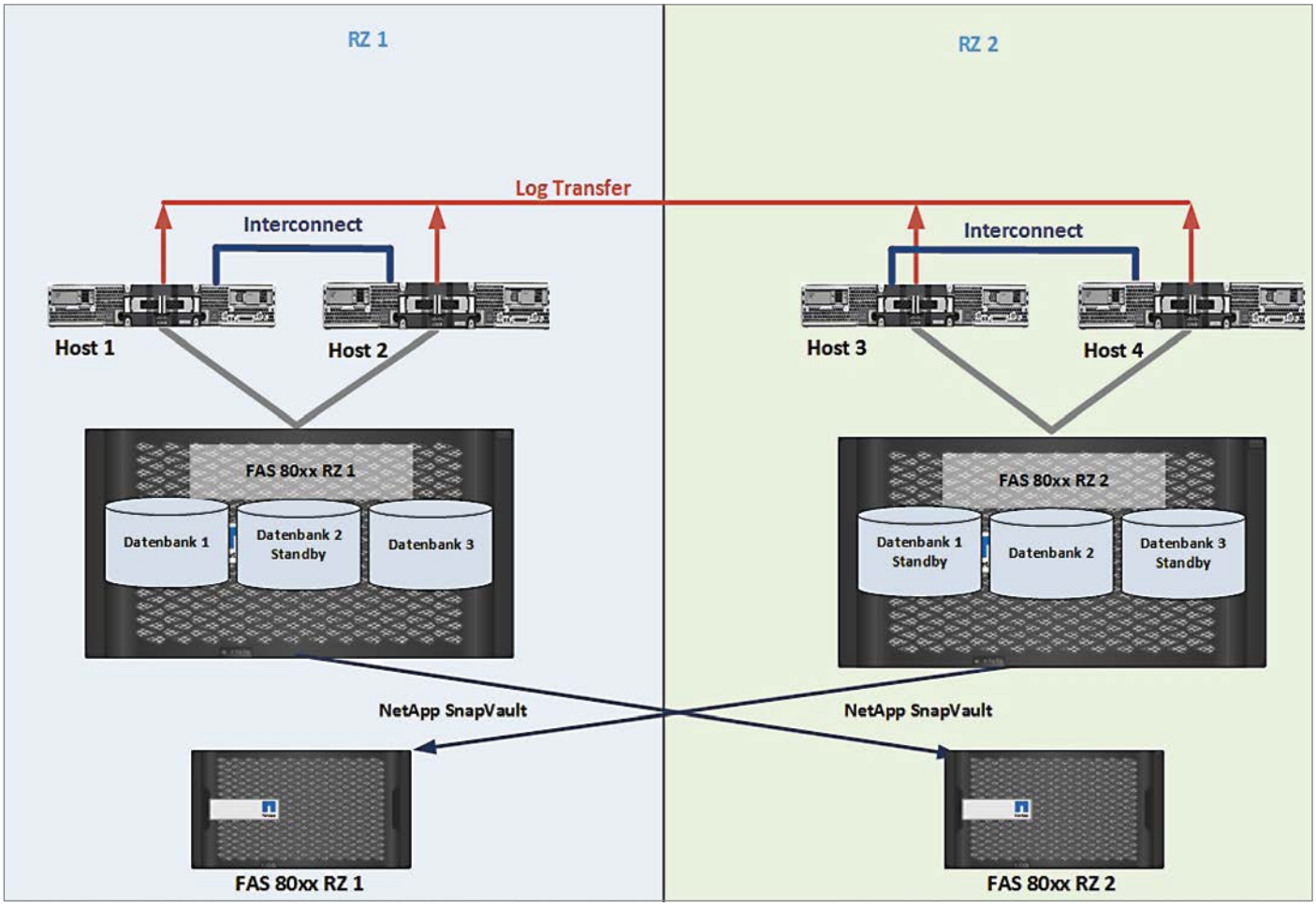


Abbildung 1: Jeweils zwei physikalische Server im RAC-Verbund

und Datenbank-Software auf einem gemeinsamen NetApp-Volume „/u01/app“. Hier ist bewusst auf ein Shared Oracle Home pro RAC verzichtet worden, um einen möglichen SPoF zu vermeiden und im Patching- und Upgrade-Verfahren flexibler zu sein.

Für jede Datenbank wurden jeweils sieben NetApp-Volumes eingerichtet. Dabei wurde zum einen auf Best Practices von NetApp beziehungsweise SnapCenter Rücksicht genommen, zum anderen ergab sich im Laufe des Projekts noch Optimierungspotenzial. So wurde zum Beispiel für die Flashback-Logs ein eigenes NFS-Volume eingerichtet, damit diese nicht unnötig Snapshot-Platz belegen. Das Sichern der Archive-Logs hingegen ist für das SnapCenter im Hinblick auf ein mögliches Recovery oder Cloning obligatorisch, wodurch diese also zwingend auf einem Volume abgelegt werden müssen, von dem Snapshots erzeugt werden. Aufgrund dieser Trennung und der gleichzeitigen Verwendung einer Fast Recovery Area wurden im Filesystem symbolische Links angelegt, die aus der FRA heraus auf ein separates Volume verweisen. Final stellen sich die Filesysteme für eine Datenbank wie in *Tabelle 1* dar, wobei lediglich für die ersten beiden Volumes Snapshots erzeugt werden.

File	Zweck
/u03/oradata/	Datenbank-Dateien, Spfiles und Password-Files
/u04/orabackup	Archive-Log-Dateien
/u03/redoA/	Redo- und Standby-Redo-Logs, Control-Files, Broker-Files
/u03/redoB/	Redo- und Standby-Redo-Logs, Control-Files, Broker-Files
/u03/temp/	Temp-Files
/u04/flashback	Flashback-Logs
/u04/rmanbackup	RMAN-Backups

Tabelle 1

Interessant ist dabei, dass die Control-Files nicht zwangsläufig auf einem Volume liegen müssen, von dem auch Snapshots generiert werden. Das SnapCenter holt sich zum Zeitpunkt des Sicherns der Datenbank die Information über die Lokation der Control-Files aus der Datenbank und sichert diese einmal als Kopie und einmal als Tracefile in den Snapshot der Daten-Dateien.

Damit die Datenbanken das Direct NFS verwenden, wurde die Datei „/etc/oranfstab“ erstellt. Entgegen der Vorgabe von Oracle „keep each „/etc/oranfstab“-file synchronized on all nodes“ (Oracle Grid Infrastructure Installation Guide 12.1) unterscheiden sich die „oranfstab“-Dateien auf den einzelnen RAC-Servern zwangsläufig, da durch redundante Storage-Anbindungen für die „local: Parameter“ jeweils vier unterschiedliche IP-Adressen pro Volume definiert werden müssen.

Das zuletzt aufgeführte Volume wird genutzt, um die Standby-Datenbanken mit dem RMAN zu sichern. Dies hat zum einen den Vorteil, dass Backups mit einem Oracle-eigenen Tool durchgeführt, und zum anderen − dem weitaus wichtigeren Aspekt −, dass hierdurch eventuelle korrupte Blöcke ausfindig gemacht werden können. Alternativ müsste man entsprechende Jobs wie DBVerify oder Analyze Validate Structure definieren, die die Primär-Datenbank belasten würden. So kann man davon ausgehen, dass auf der Standby-Seite keine korrupten Blöcke vorhanden sind und bei Bedarf wäre ein Block Media Recovery auf der Primär-Datenbank möglich. Voraussetzung ist ein vorhandener RMAN-Catalog.

Änderungen der Konfiguration

Nachdem die Infrastruktur bereits aufgebaut war, stieß man im laufenden Betrieb auf gewisse Fehler beziehungsweise Effekte, aufgrund derer die ursprünglichen Konfigurationen weniger Komponenten noch einmal angepasst wurden. Ursprünglich war eines der drei für die RAC-Cluster angelegten Volumes für die Voting Disks in dem jeweils entfernten Rechenzentrum angelegt. Die Voting Disks sind also von einem entfernten Storage gemountet worden, wodurch man sich eine höhere Ausfallsicherheit versprach. Die Praxis zeigte jedoch, dass dies bei Stromausfall oder Netzwerk-Unterbrechungen zum entfernten RZ auf dem hiesigen RAC unmittelbaren Einfluss hat. Zwar laufen die RAC-Cluster auch ohne die dritte Voting Disk, jedoch gibt es allein bei simplen Linux-Befehlen wie einem „df“ Probleme, da das gemountete NFS nicht zugreifbar ist. Infolgedessen wurde das Volume mit der dritten Voting Disk wieder auf das jeweils lokale Storage verlegt.

Entgegen der Oracle-Empfehlung für die OFA-Struktur wurden die Verzeichnisse für die Grid Infrastructure anders benannt: „/u01/app/grid/product/12.1.0.2/gridhome” für die Grid-Software und „/u01/app/gridbase“ für das Grid-Base. Das funktioniert auch ganz gut und ist nach Ansicht der Autoren so besser strukturiert. Allerdings führt dies gegebenenfalls dazu, dass das Grid Infrastructure Management Repository (GIMR) zunächst „falsch“ aufgebaut wird. Laut Oracle-Dokumentation (Oracle Grid Infrastructure Installation Guide 12.1) erkennt der Installer das „ORACLE_BASE“ selbst, wenn das Verzeichnis die Struktur „/u[0-9][1-9]/app/<osuser>“ hat. Da dies in der vorliegenden Konfiguration nicht der Fall war, wurde das GIMR fälschlicherweise mit dem „ORACLE_BASE /u01/app/oracle“ aufgebaut. Daraufhin wurde vorsichtshalber in der „.profile“ des Benutzers „grid“ die Variable „ORACLE_BASE“ auf „/u01/app/gridbase“ gesetzt.

Da für die Entwicklungsumgebung nur ein Server bereitgestellt wurde, installierte man hier eine Single-Instance-Konfiguration. Dies erwies sich allein bei der Ansible-Automatisierung durchweg als Spezialfall. Zudem hatte man sich einen Bruch in der Abbildung des Staging eingebaut. Später hat man die Konfiguration dieses einen Servers auf RAC geändert; zukünftig soll ein zweiter Server hinzukommen, um ein vollwertiges RAC, wenn auch kein Data Guard, in der Entwicklung zu haben.

Fazit

Nachdem die grundlegenden Planungen abgeschlossen waren, erstreckte sich der Aufbau der neuen Infrastruktur bis hin zum produktiven Betrieb der Datenbanken über einen Zeitraum von etwa einem halben Jahr. Hierzu sei aber gesagt, dass zwischenzeitlich jede der vier Umgebungen im Rahmen von streng definierten Abnahmetests umfangreich geprüft wurde. Hinzu kamen die im Artikel erwähnten an-

fänglichen Schwierigkeiten mit dem SnapCenter oder beispielsweise eine in ihrer Gänze neu implementierte Überwachung über den Oracle Enterprise Manager.

Die Zielsetzung der hohen Verfügbarkeit wurde erreicht. Bei einer geplanten Stromabschaltungs-Übung des gesamten Primary-Rechenzentrums beispielsweise wurden die Datenbanken und das Storage absichtlich nicht in den normalen Ablaufplan integriert. Dabei hätte man vorab einen Data Guard Switchover durchführen können; stattdessen konnte problemlos ein Failover initiiert werden, nachdem der Strom der gesamten Infrastruktur im ersten RZ gekappt wurde. Es sei erwähnt, dass zu diesem Zeitpunkt die Datenbanken noch keinen produktiven Status hatten. Durch die Aktivierung von Flashback auf allen Datenbanken ist man in der Lage, nach einem Failover durch ein Reinstate das Data Guard zeitnah wieder in einen konsistenten Zustand zu versetzen. Hierbei wird übrigens der Modus „Maximum Availability" eingesetzt.

Auch im Rahmen von Wartungsarbeiten zeigten sich die Vorteile des Data Guard insofern, als dass der Juli-PSU durch Standby First Apply nahezu ohne Downtime für die Applikationen implementiert werden konnte. Unschön ist weiterhin das leidige Thema der Java-Patches. Hier musste zwingend eine Downtime von etwa zehn Minuten her, um das Datapatch in die mit der OJVM betriebene Datenbank korrekt einzuspielen.

Das Backup- und Recovery-Verfahren durch das SnapCenter – wohlgemerkt das erste offizielle Release von NetApp – erledigt seine Aufgaben zuverlässig. Um die Datenbank per Snapshot zu sichern, wird diese kurzzeitig in den Backup-Modus („alter database begin backup;") gesetzt und die „Verpointerungen" auf Storage-Ebene gesetzt. Hat man sich einmal in der browserbasierten GUI zurechtgefunden, kann man schnell und einfach neben den regelmäßigen Sicherungsjobs auch Tasks wie Point-In-Time-Recovery oder gar einen Clone einer laufenden Datenbank erstellen. Dieser Clone kann sogar zeitlich in der Vergangenheit liegen, je nach Vorhaltezeit der Snapshots. Somit ist es durch SnapVault innerhalb weniger Minuten möglich, auch einen bis zu 35 Tage alten Snapshot für den Aufbau eines Clone zu verwenden. Wichtig in der Gesamt-Konstellation ist jeweils der Einsatz eines Secondary-Storage, worauf die aktuell erstellten Snapshots der Datenbank per SnapVault-Funktion kopiert werden, da ein Snapshot allein noch kein sicheres Backup darstellt.

Leider war es – entgegen den Angaben von NetApp – nicht möglich, die Standby-Datenbanken über das SnapCenter brauchbar zu sichern. Dies wäre nur über ein Offline-Backup möglich gewesen.

Alles in allem wurde eine stabile, zuverlässige Oracle-Infrastruktur geschaffen, die ihrer Bezeichnung „alway online" bisher voll gerecht wird.

Tim Hensel
tim.hensel@devk.de

Johannes Ahrends
johannes.ahrends@carajandb.com

Oracle VM und Virtual Shared Storage

Nico Henglmüller und Dr. Thomas Petrik, Sphinx IT Consulting GmbH

Die Architektur von Oracle VM sieht die Verantwortlichkeit der Speicherverwaltung bei externer Hardware. Eine solche ist für Klein- und Mittelbetriebe oft unerschwinglich. Dies führte die Autoren zur Frage: „Wie kann eine neuartige Infrastruktur konzipiert werden, die gleichzeitig hochausfallsicher, universell einsetzbar und einfach zu betreiben ist; noch dazu kostengünstiger als eine vergleichbare Enterprise-Lösung?" Eine ambitionierte Truppe machte sich auf die Suche und berichtet in diesem Artikel von der verwendeten Technologie und vom innovativen Ergebnis.

Zu Beginn stand die Idee, einen hochverfügbaren und leistbaren, Multipurpose Cluster mit Hard-Partitioning-Unterstützung von Oracle-Datenbanken zu entwickeln. Dieser Idee folgte ein Prototyp, bestehend aus zwei Oracle-VM-Knoten und einem Virtual Shared Storage. Die eingesetzten Knoten basieren auf Standard-x86-Architektur und verwenden ei-

nen Verbund aus redundanten Festplatten als Speicher.

Die Überlegung war, dass im Falle eines unvorhergesehenen Ausfalls eines Knotens sichergestellt ist, dass der zweite Knoten mit minimaler Unterbrechung (Cold failover) den Betrieb fortsetzt. Das Konzept ging auf. Oracle VM bringt für diesen Einsatzzweck die notwendigen Kern-Features mit: High Availability out of the box mit OCFS2 als Cluster-Filesystem und Heartbeat-Komponente sowie eine zentrale Verwaltung des gesamten Clusters in Form von Oracle VM.

Das in *Abbildung 1* dargestellte Design erlaubt es, dass die Speicher-Architektur für den Hypervisor transparent ist. Dem Oracle VM Server wird ein blockbasiertes Device zur Verfügung gestellt, auf dem ein OCFS2-Dateisystem für den Clusterbetrieb von Oracle VM angelegt ist. Durch den Einsatz von Virtual Shared Storage werden Leseoperationen immer lokal durchgeführt. Im Gegensatz hierzu werden schreibende Operationen erst als geschrieben angesehen, sobald die Daten auf beiden Knoten persistiert wurden.

Netzwerk-Architektur

Bei der Gestaltung des Netzwerks ist es notwendig, eine redundante Verbindung exklusiv für die Synchronisation zwischen den Knoten miteinzubeziehen *(siehe Abbildung 2)*. Um einem Bottleneck entgegenzuwirken, werden diese zwei Verbindungen im Link-Aggregation-Modus mit jeweils 10 GBit/s entweder direkt oder über einen „low latency"-10-GBit-Switch verbunden.

Das Netzwerk für die virtuellen Maschinen sowie das Management-Netzwerk werden in einem Active-Passive-Failover-Modus, mit einer Bandbreite von mindestens einem GBit/s betrieben. Das Management und das VM-Netzwerk werden unverändert im Sinne von Oracle VM verwendet. Darüber hinaus kann der VSS-Link durch Mapping eines V-LANs als 20-Gbit-„low latency Interconnect" für virtualisierte Active-Active-Cluster wie beispielweise Oracle RAC verwendet werden.

Reaktionen bei unerwarteten Hardware-Ausfällen

Virtual Shared Storage ermöglicht es, bei einer unerwarteten Betriebs-Unterbrechung des kompletten Storage eines Knotens alle I/O-Operationen transparent auf das Storage des zweiten Knoten umzuleiten. Die Folge ist, dass der Betrieb nicht unterbrochen wird und sich die Daten zu jedem Zeitpunkt in einem konsistenten Zustand befinden.

Wie bei allen Clustern ist die Split-Brain-Problematik auch beim Zwei-Knoten-Cluster ein heiß diskutiertes Thema. Unter „Split Brain" versteht man den kompletten Ausfall der Kommunikation zwischen den Knoten. Die Problematik ist, dass sich beide Knoten als neuer Master deklarieren. Dies führt zu Inkonsistenzen, da beide Datenmengen weiterhin gepflegt werden, allerdings keine Synchronisation zwischen diesen stattfindet.

Die Herausforderung bei einem unvorhergesehenen Hardware-Ausfall ist, schnellstmöglich die Abwesenheit eines Knotens zu erkennen und die Datenkonsistenz zwischen den Knoten zu gewährleisten. Wird nun die Replikationsverbindung komplett unterbrochen, muss der zu schreibende Datenverkehr auf Knoten 1 eingefroren werden. Anschließend stellen sich die folgenden Optionen:

- *Szenario 1*
  Falls das Management-Netzwerk ebenfalls keine Verbindung aufweist, wird Knoten 1 heruntergefahren, um die Daten-Integrität sicherzustellen.

- *Szenario 2*
  Falls sich die Knoten über das Management-Netzwerk erreichen, wer-

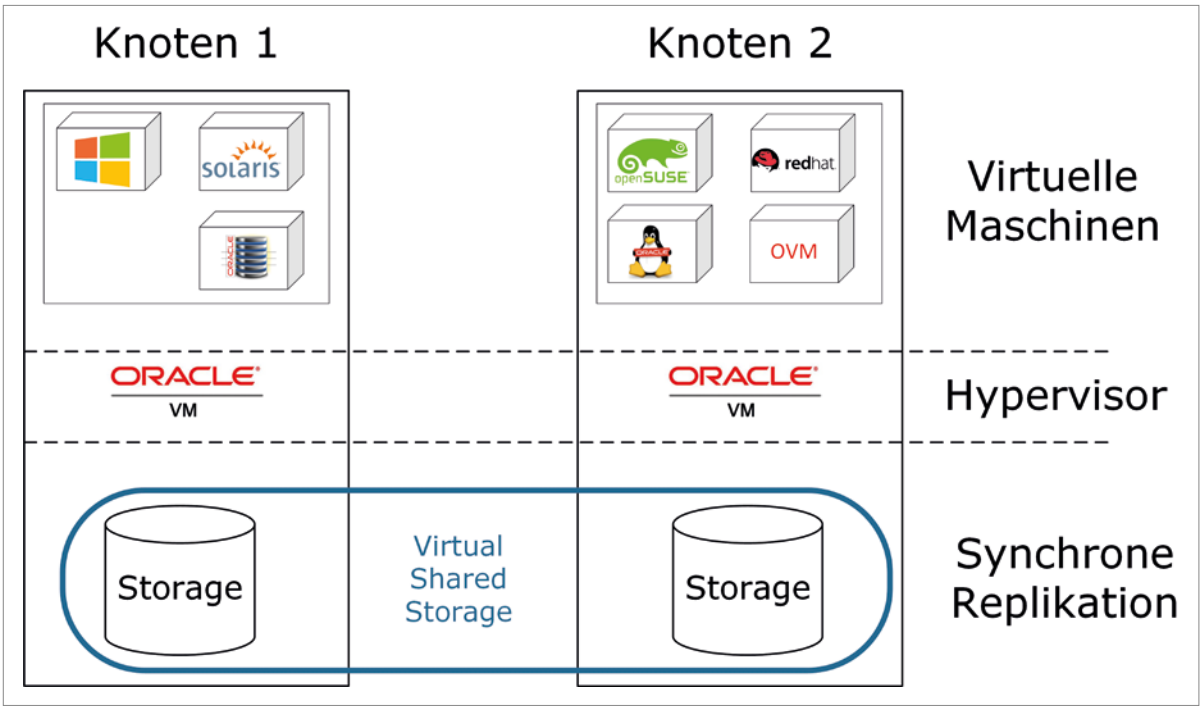


Abbildung 1: Aufbau des Zwei-Knoten-Clusters mit Virtual Shared Storage

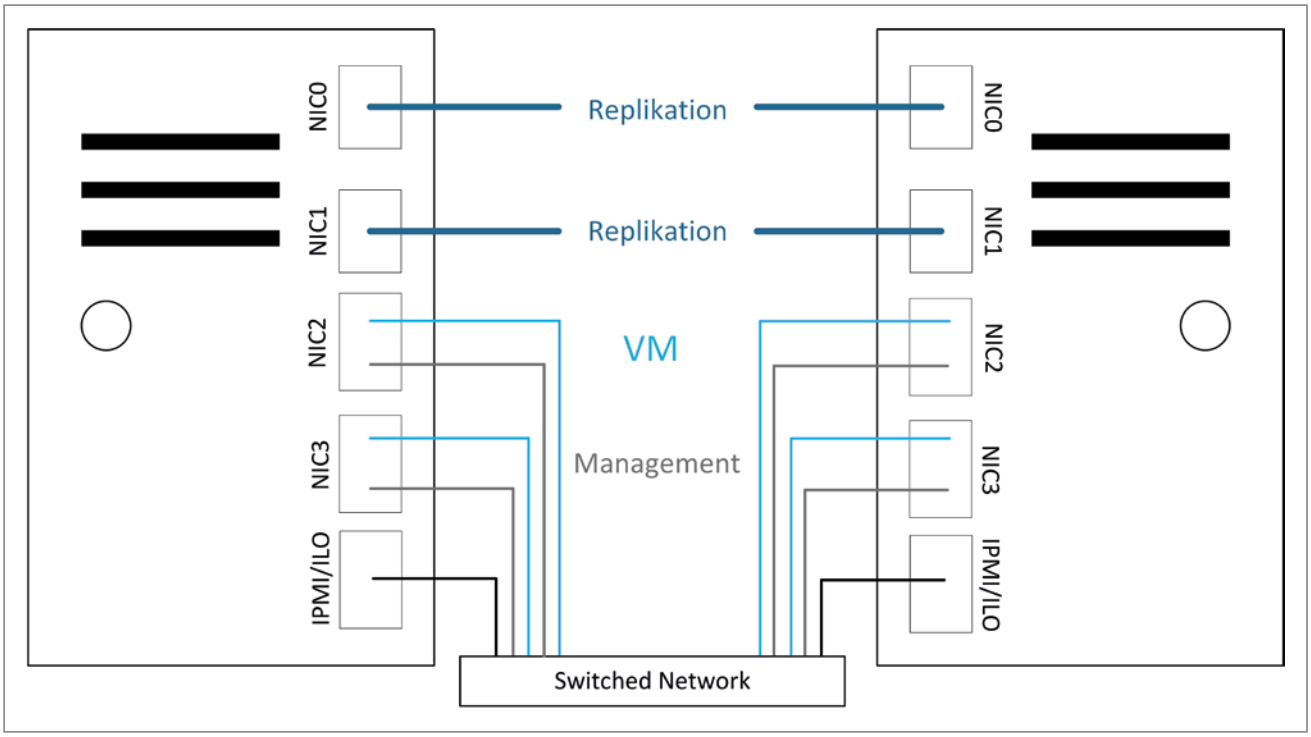


Abbildung 2: Netzwerk-Architektur

den beide Knoten heruntergefahren, da die Synchronisation über das Replikations-Netzwerk nicht mehr erfolgen kann.

- *Szenario 3*
  Falls das Management-Netzwerk erreichbar, jedoch Knoten 2 nicht erreichbar ist, wird Knoten 1 zum neuen Master des Clusters. In der Rolle des Masters werden die virtuellen Maschinen, die als hochverfügbar markiert sind, auf Knoten 1 gestartet. Der Status von Knoten 2 bleibt weiterhin unklar. Falls dieser ebenfalls eine aufrechte Verbindung in das Management-Netzwerk besitzt, ist dies die einzige Situation, in der ein Split Brain unausweichlich ist.

Betrieb

Für den Betrieb sind Werkzeuge für Monitoring, Backup und Recovery sowie Updates und Support essenziell. Mithilfe des Monitorings ist eine detailgenaue Überwachung des Cluster-Zustands ersichtlich und erlaubt eine übersichtliche Erstdiagnose. Die Backup-Lösung ist in den Oracle VM Manager voll integriert und individuell konfigurierbar. Als Sicherungsziel wird ein in OVM eingebundener Netzwerkspeicher verwendet. Die Sicherung erfolgt, wie die Wiederherstellung von virtuellen Maschinen, genauso nahtlos im Oracle VM Manager und erfordert keinerlei zusätzliche Software.

Fazit

Gerade im digitalen Zeitalter können sich auch kleine Unternehmen Unterbrechungen des Betriebs oder Service-Ausfälle nicht erlauben. Die logische Schlussfolgerung ist daher, auf intelligente, ausfallsichere Lösungen zu setzen. Das hier vorgestellte schlanke Produkt BlueBoxx ermöglicht es heute Klein- und Mittelbetrieben, Enterprise-Funktionalitäten zu nutzen. Gerade im Umfeld von Oracle sind smarte Lösungen gefragter denn je.

Die Virtualisierung mithilfe von Oracle VM ermöglicht die vollständige Zertifizierung der Supportkette von Oracle- und Microsoft-Produkten. Durch die Anerkennung der Hard-Partitioning-Fähigkeiten des Hypervisors seitens Oracle birgt die Lizenzierung von Oracle-Datenbanken großes Potenzial.

Nico Henglmüller, BSc
nico.henglmueller@sphinx.at

Mag. Dr. Thomas Petrik
thomas.petrik@sphinx.at

Data Guard Basics

Marco Mischke, Robotron Datenbank Software GmbH

Wenn es um Hochverfügbarkeit geht, ist der erste Gedanke meist RAC. Dieser deckt jedoch nur ein Teilgebiet ab, nämlich die Server-Redundanz. Um Desaster-Szenarien über Rechenzentrumsgrenzen hinaus abzudecken, führt an der Vorhaltung einer Standby-Datenbank und damit an Data Guard praktisch kein Weg vorbei. Der Artikel zeigt die Grundlagen, um die häufigsten Stolpersteine bereits von Beginn an umgehen zu können.

Zuerst einmal stellt sich die Frage, warum man Data Guard überhaupt benötigt. Der Real Application Cluster ist doch bereits hochverfügbar durch die Server-Redundanz und das Storage ist ebenfalls gespiegelt, sei es nun mittels ASM oder direkt im Storage selbst. Typischerweise befinden sich diese Komponenten allerdings in einem Serverraum, wenn nicht sogar im gleichen Schrank. Das trifft insbesondere auf Engineered Systems wie die Oracle Database Appliance zu.

Für die Ausfallsicherheit über Rechenzentrumsgrenzen hinweg ist ein zweites System notwendig, das gegebenenfalls die Aufgaben übernehmen kann. An diesem Punkt kommt Data Guard ins Spiel. Man hält auf einem entfernten System eine Kopie der Produktiv-Datenbank durch permanentes Nachspielen der in den Redologs aufgezeichneten Transaktionen auf dem aktuellen Stand. Diese Umgebung mit Primär- und Standby-Datenbank wird durch Data Guard verwaltet und ermöglicht geplantes sowie ungeplantes Umschalten zwischen beiden Datenbanken. Im Desaster-Fall lässt sich somit innerhalb von Minuten die volle Funktionalität durch die Aktivierung des Standby-Systems wiederherstellen.

Namensgebung

Bei der Einrichtung einer Data-Guard-Umgebung ist einiges zu beachten. Das beginnt bereits bei der Namensgebung. Dazu muss man wissen, welche Namen und Parameter es überhaupt gibt *(siehe Tabelle 1)*.

In einer Data-Guard-Umgebung hat die Standby-Datenbank den gleichen „db_name" wie die Primär-Datenbank, da sie eine byteweise Kopie ist. Die Unterscheidung der beiden Systeme erfolgt daher über den „db_unique_name". Oftmals werden die Rollen in den „db_unique_name" eingebaut, also beispielsweise „DBPRIM" und „DBSTBY". Es ist aber zu beachten, dass man die Rollen auch tauschen kann – dann sind diese Namen irreführend. Man sollte daher die Namen unbedingt unabhängig von der Rolle festlegen und stattdessen ein Kürzel zur Identifizierung des Rechenzentrums verwenden, also „DBRZ1" und „DBRZ2". Der Service-Name wird dann entsprechend unabhängig von Rollen etc. vergeben, um das Konstrukt vor den Client-Anwendungen zu maskieren. *Abbildung 1* zeigt die Namensgebung in einem Data-Guard-Verbund mit zwei RAC-Systemen.

Diese Art der Namensgebung sollte also von Anfang an konsequent verfolgt werden, auch wenn man zu Beginn noch keinen Data Guard verwenden will. Man ist jedoch bereits für zukünftige Entwicklungen gerüstet. Weiterhin empfiehlt es sich, die Namen in Großschreibung zu verwenden, insbesondere bei Benutzung von Oracle Managed Files (OMF). Für einige Pfade werden die Namen automatisch in Großschreibung umgewandelt, andere nicht, sodass man mit einer generellen Großschreibung für mehr Übersicht sorgen kann.

Parameter	Bedeutung	Data Guard relevant
db_name	Formt aus Datendateien die Datenbank	Ja
instance_name	Identifiziert die Prozesse	Nein
db_unique_name	Eindeutiger Name der Datenbank zur Identifizierung	Ja
service_name	Abstraktionsschicht für Applikationen	Ja
ORACLE_SID	Ermittelt das Parameter-File sowie die Passwort-Datei beim Instance-Start	Nein
global_name	Globaler Name der Datenbank (für Datenbank-Links)	Nein

Tabelle 1

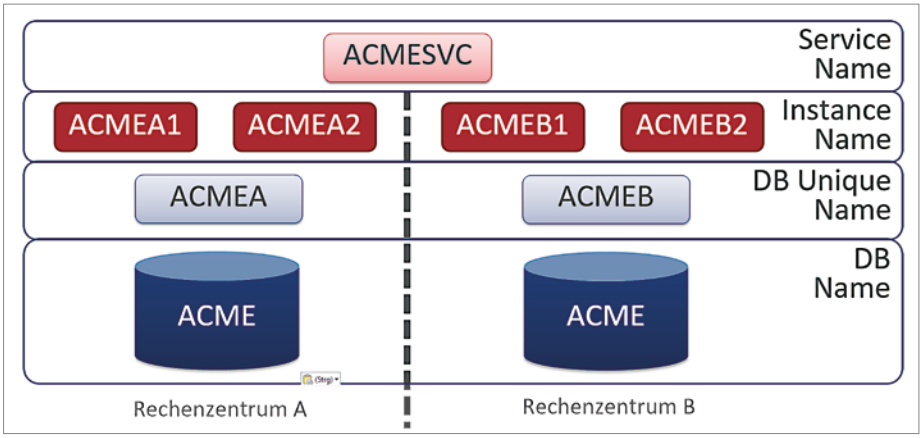


Abbildung 1: Beispielhafte Namensgebung

```
SQL> shutdown immediate
SQL> startup mount
SQL> alter database archivelog;
SQL> alter database open;
SQL> alter database force logging;
SQL> alter database flashback on;
```

Listing 1

```
SQL> alter database add standby logfile thread 1 group 10 size 500M;
SQL> alter database add standby logfile thread 1 group 11 size 500M;
SQL> alter database add standby logfile thread 1 group 12 size 500M;
SQL> alter database add standby logfile thread 1 group 13 size 500M;
```

Listing 2

```
SID_LIST_LISTENER =
  (SID_LIST =
      (SID_DESC =
          (GLOBAL_DBNAME = db12cb.robotron.de)
          (ORACLE_HOME = /u01/app/oracle/product/12.1.0.2/db)
          (SID_NAME= db12cb)
      )
  )
```

Listing 3

Vorbereitung der Quell-Datenbank

Bevor man nun eine Data-Guard-Umgebung aufbaut, bereitet man die Quell-Datenbank soweit es geht vor, um den Aufwand möglichst gering zu halten. Die Parameter zur Namensgebung wurden bereits erläutert. Dazu kommen noch das Aktivieren des Archivelog-Modus und vor allem das Erzwingen des Logging. Ansonsten können durch Nologging-Operationen Informationen zu Transaktionen an den Redologs vorbei geschrieben werden,

```
connect target sys/oracle@db12ca
connect auxiliary sys/oracle@db12cb

startup clone nomount

duplicate target database for standby
from active database
spfile
   set db_name 'db12c'
   set db_unique_name 'db12cb '
   set db_create_file_dest '/u01/app/oracle/oradata '
   set db_create_online_log_dest_1 '/u01/app/oracle/oradata '
   set db_recovery_file_dest '/u01/app/oracle/fra '
   set control_files ' '
   set diagnostic_dest '/u01/app/oracle '
   set audit_file_dest '/u01/app/oracle/admin/db12cb/adump '
   set dg_broker_start 'FALSE '
   set filesystemio_options 'ASYNCH '
   reset log_archive_config
   reset log_archive_dest_1
   reset log_archive_dest_2
   reset fal_server
   reset fal_client
   reset db_domain
   reset memory_target
   set sga_target '1200M '
   set pga_aggregate_target '200M '
noresume
dorecover
;
```

Listing 4

Parameter	Wert	Bedeutung
standbyfile_management	auto	Datendateien, die auf der primären Datenbank erzeugt werden, werden automatisch auch auf der Standby-Datenbank angelegt
db_block_checking	medium	Führt logische Checks der Datenbank-Blöcke durch (bei „full“ auch für Index-Blöcke mit Performance Overhead)
db_block_checksum	typical	Berechnet vor dem Schreiben die Check-Summen für Daten- und Redo-Blöcke
db_lost_write_protect	typical	Zeichnet Cache-Reads in den Redologs auf, um SCNs vergleichen zu können

Tabelle 2

```
SQL> select force_logging, log_mode, flashback_on from v$database;

FORCE_LOGGING    LOG_MODE       FLASHBACK_ON
---------------  ------------   ------------
YES              ARCHIVELOG     NO
```

Listing 5

```
SQL> alter system set dg_broker_config_file1='/u01/app/oracle/oradata/DB12CA/dr1db12ca.dat';
SQL> alter system set dg_broker_config_file2='/u01/app/oracle/fra/DB12CA/dr2db12ca.dat';
SQL> alter system set dg_broker_start=true;
```

Listing 6

was ein Recovery unmöglich macht. Um ein Re-Instanziieren nach einem Failover zu vereinfachen, aktiviert man weiterhin Flashback Database *(siehe Listing 1)*.

Um die Standby-Datenbank möglichst synchron zur Primär-Datenbank halten zu können, sind Standby-Redologs erforderlich. Die Primär-Datenbank schreibt dann parallel zu den eigenen Redologs die Transaktionen auch in die Standby-Redologs der Standby-Datenbank. Diese kann die Transaktionen dann direkt einspielen und nicht erst beim Archivieren eines Redologs auf der Primärseite.

Es empfiehlt sich, eine Gruppe mehr an Standby-Redologs anzulegen, als es Online-Redolog-Gruppen gibt. Im Fall von RAC muss man selbstverständlich für jeden Thread entsprechende Gruppen anlegen. Am besten verwendet man einen separaten Zahlenbereich für die Gruppennummer, um die Dateien direkt unterscheiden zu können *(siehe Listing 2)*. Zur maximalen Sicherheit gegenüber Datenverlust sind noch einige weitere Parameter relevant, die in *Tabelle 2* aufgelistet sind.

Anlegen der Standby-Datenbank

Ist die Quell-Datenbank soweit vorbereitet, wird die zukünftige Standby-Datenbank per „RMAN DUPLICATE“ als Kopie erstellt. Auf dem Standby-Server müssen dazu lediglich die Datenbank-Software installiert und das Verzeichnis für „audit_file_dest“ angelegt sein. Um sich initial an der Zieldatenbank auch remote anmelden zu können, ist per „orapwd file=$ORACLE_HOME/dbs/orapwdb12cb password=oracle force=y“ ein Passwort-File anzulegen. Zudem benötigt man einen statischen Listener-Eintrag in der „listener.ora“ *(siehe Listing 3)*.

Nun kann das Duplikat gestartet werden, dabei setzt man am besten gleich alle Parameter mit den passenden Werten oder setzt diese zurück, um mögliche Probleme direkt auszuschließen *(siehe Listing 4)*. Überprüft man nach erfolgreicher

Duplizierung nun den Zustand der Kopie, so stellt man fest, dass alle Einstellungen übernommen wurden, mit Ausnahme des Flashback-Database *(siehe Listing 5)*. Flashback Database ist also auf der Kopie nochmals separat zu aktivieren.

Data-Guard-Konfiguration

Da die Quell- und Ziel-Datenbank nun vorhanden ist, kann man mit der Einrichtung des Data Guard beginnen. Die Konfiguration wird in zwei gespiegelten Dateien verwaltet, ähnlich den Kontroll-Dateien der Datenbank. Den Namen und Speicherort dieser Dateien legt man entsprechend über Initialisierungsparameter fest. Die eigentliche Arbeit in einer Data-Guard-Umgebung übernimmt der Data-Guard-Broker-Prozess, den man durch Initialisierungsparameter startet *(siehe Listing 6)*.

Natürlich muss man das sowohl auf der Primär- wie auch auf der Standby-Datenbank einstellen. Anschließend kann die Konfiguration per „dgmgrl" erstellt werden. Man verwendet für die Verbindung zu Primär- und Standby-Datenbank am besten die Easy-Connect- Syntax, da man so unabhängig von Einstellungen und Fehlern in der tnsnames.ora oder Ähnlichem ist: „DGMGRL> connect sys/oracle@"oel6u4:1521/db12ca.robotron.de"". Initial wird die Konfiguration mit einer Primär-Datenbank erstellt *(siehe Listing 7)*, als Nächstes fügt man die Standby-Datenbank hinzu *(siehe Listing 8)*.

Nun stellt man den Redolog-Transport in beide Richtungen auf „synchrone Übertragung", um die Transaktionen möglichst ohne Zeitverzug in die Standby-Redologs zu übertragen *(siehe Listing 9)*.

Als Letztes stellt man noch den Protection Mode auf die gewünschte Stufe. Typischerweise ist das „Maximum Availability", dabei werden die Transaktionen nach Möglichkeit synchron übertragen, die Primär-Datenbank stoppt aber nicht, wenn die Standby-Datenbank gerade einmal nicht erreichbar ist. Die Konfiguration ist damit vollständig und kann per „DGMGRL> enable configuration" aktiviert werden. Ab diesem Moment überträgt die Primär-Datenbank ihren Redolog-Stream zur Standby-Datenbank, wo die Transaktionen direkt eingespielt werden. Den Status kann man nach einigen Sekunden entsprechend überprüfen *(siehe Listing 10)*. Damit ist die Konfiguration abgeschlossen.

```
DGMGRL> create configuration db12c as
> primary database is db12ca
> connect identifier is 'oel6u4:1521/db12ca.robotron.de';
```

Listing 7

```
DGMGRL> add database db12cb as
> connect identifier is 'oel6u4:1521/db12cb.robotron.de'
> maintained as physical;
```

Listing 8

```
DGMGRL> edit database db12ca set property logxptmode='SYNC';
DGMGRL> edit database db12cb set property logxptmode='SYNC';
```

Listing 9

```
DGMGRL> show configuration verbose

Configuration - db12c

  Protection Mode: MaxPerformance
  Members:
  db12ca - Primary database
      db12cb - Physical standby database

  Properties:
      FastStartFailoverThreshold       = '30'
      OperationTimeout                 = '30'
      TraceLevel                       = 'USER'
      FastStartFailoverLagLimit        = '30'
      CommunicationTimeout             = '180'
      ObserverReconnect                = '0'
      FastStartFailoverAutoReinstate   = 'TRUE'
      FastStartFailoverPmyShutdown     = 'TRUE'
      BystandersFollowRoleChange       = 'ALL'
      ObserverOverride                 = 'FALSE'
      ExternalDestination1             = ''
      ExternalDestination2             = ''
      PrimaryLostWriteAction           = 'CONTINUE'

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS
```

Listing 10

```
srvctl add database -db db12ca -dbname db12c \
  -oraclehome /u01/app/oracle/product/12.1.0.2/db \
  -domain robotron.de \
  -spfile /u01/app/oracle/product/12.1.0.2/db/dbs/spfiledb12ca.ora \
  -pwfile /u01/app/oracle/product/12.1.0.2/db/dbs/orapwdb12ca \
  -role PRIMARY -startoption open -stopoption immediate
```

Listing 11

Betriebsthemen

Um die beteiligten Datenbanken immer automatisch mit dem Server zu starten und besonders um die korrekten Start-Optionen zu gewährleisten, sollte die Oracle Grid Infrastructure verwendet werden. Auch wenn weder Cluster noch

```
srvctl add database -db db12cb -dbname db12c -oraclehome /u01/app/oracle/product/12.1.0.2/db \
   -domain robotron.de \
   -spfile /u01/app/oracle/product/12.1.0.2/db/dbs/spfiledb12cb.ora \
   -pwfile /u01/app/oracle/product/12.1.0.2/db/dbs/orapwdb12cb \
   -role PHYSICAL_STANDBY -startoption mount -stopoption immediate
```

Listing 12

```
srvctl add service -db <unique name> -service <service name> \
   -role primary -failovertype SELECT \
   -failovermethod BASIC -cardinality uniform \
   -failoverdelay 180 -failoverretry 5
```

Listing 13

```
<TNS alias>=
   (DESCRIPTION_LIST=
       (LOAD_BALANCE=off)
       (FAILOVER=on)
       (DESCRIPTION=
           (CONNECT_TIMEOUT=5)
           (TRANSPORT_CONNECT_TIMEOUT=3)
           (RETRY_COUNT=3)
           (ADDRESS_LIST=
               (LOAD_BALANCE=on)
               (ADDRESS=(PROTOCOL=TCP)
                   (HOST=<primary scan>)(PORT=1521))
       )
           (CONNECT_DATA=(SERVICE_NAME=<service name>))
       )
       (DESCRIPTION=
           (CONNECT_TIMEOUT=5)
           (TRANSPORT_CONNECT_TIMEOUT=3)
           (RETRY_COUNT=3)
           (ADDRESS_LIST=
               (LOAD_BALANCE=on)
               (ADDRESS=(PROTOCOL=TCP)
                   (HOST=<standby scan>)(PORT=1521))
       )
           (CONNECT_DATA=(SERVICE_NAME=<service name>))
   )
   )
```

Listing 14

ASM zum Einsatz kommen sollen, ergibt das durchaus Sinn, denn über die Datenbank-Ressourcen in der Grid Infrastructure kann der richtige Modus zum Starten definiert werden.

Wer keine Lizenz für Active Data Guard besitzt, darf die Standby-Datenbank nur im „MOUNT" starten. Per Standard würde die Datenbank aber „Read only" geöffnet werden, was einen Lizenzverstoß bedeutet. Man legt also die Datenbank-Ressourcen mit den passenden Einstellungen an, einmal für die Primär- *(siehe Listing 11)* und einmal für die Standby-Datenbank *(siehe Listing 12)*.

Bei geplanten oder ungeplanten Umschaltungen benutzt der Data Guard Broker dann die Grid Infrastructure zum Stoppen und Starten der Datenbanken. Das erspart zum einen gesonderte Konfigurationen für SQL*Net und zum anderen passt der Broker die Start-Optionen mit an. Bei einem Rollentausch startet also auch die neue Standby-Datenbank nur im „MOUNT".

Um die Data-Guard-Umgebung für die Applikationen unabhängig von der Rollenverteilung zugänglich zu machen, müssen sich die Applikationen über einen separat einzurichtenden Service verbinden. Dieser wird in der Grid Infrastructure auf beiden Seiten eingerichtet und startet abhängig von der Rolle der Datenbank. Zusätzlich kann der Service noch verschiedene Einstellungen für das Failover enthalten *(siehe Listing 13)*.

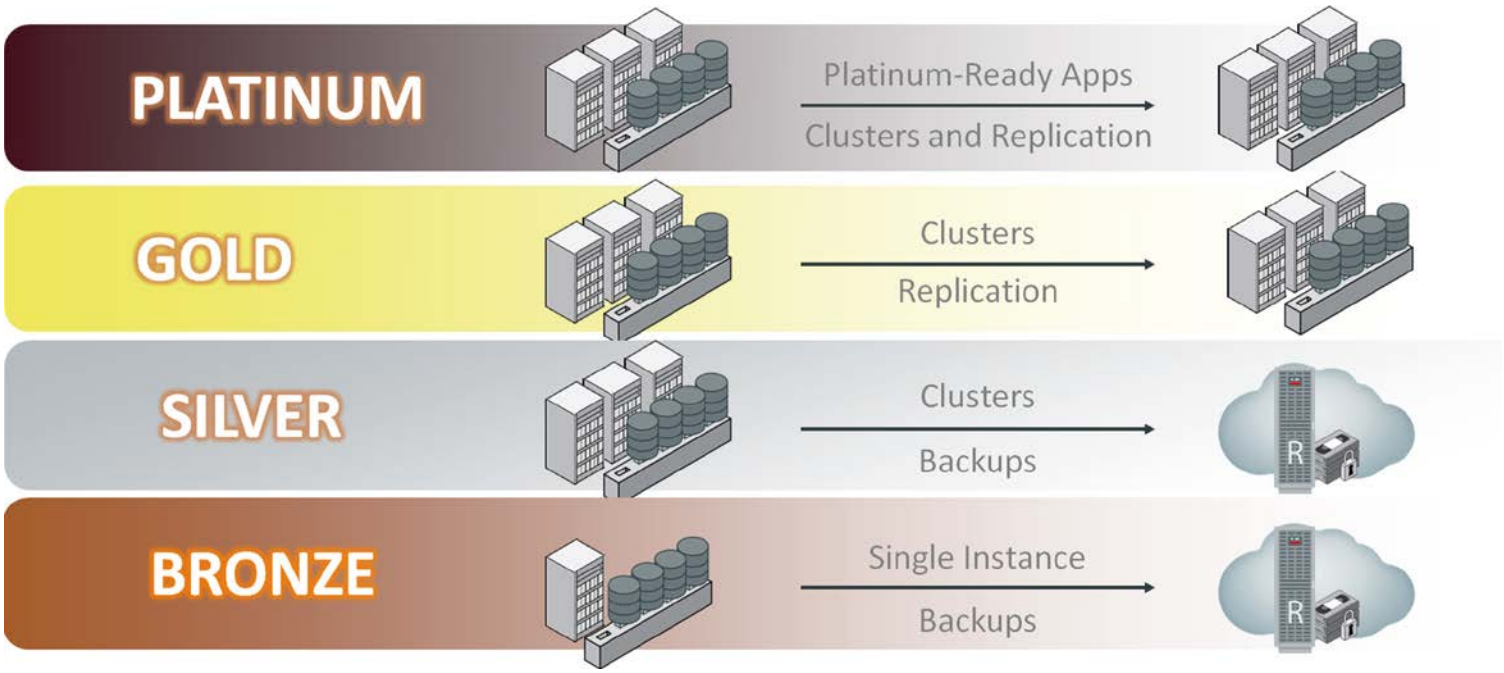
Diese Definition bewirkt, dass der Service nur startet, wenn die zugrunde liegende Datenbank die Rolle „PRIMARY" hat. Für die Applikationen muss man nun nur noch einen entsprechenden TNS-Alias einrichten, der mit einer Description-Liste beide Seiten im Data-Guard-Verbund definiert *(siehe Listing 14)*.

Benutzen die Applikationen nun diesen TNS-Alias, wird die Verbindung immer mit der aktuellen Primär-Datenbank hergestellt, ohne dass irgendwelche Anpassungen erforderlich sind.

Fazit

Mit Data Guard wird eine komplexe Technologie durch einfache Werkzeuge handhabbar. Man kann damit seine Datenbanken auf einfache Art und Weise auf ein anderes System spiegeln und dort aktuell halten. Im Desasterfall, also dem kompletten Ausfall des Datenbank-Systems, braucht man dann lediglich ein einziges Kommando, um die Standby-Datenbank zu aktivieren und weiterarbeiten zu können.

Marco Mischke
marco.mischke@robotron.de



Oracle Maximum Availability Architecture

Sebastian Solbach, ORACLE Deutschland B.V. & Co. KG

Daten sind für viele Unternehmen von zentraler Bedeutung. Damit steigen auch die Anforderungen an die Verfügbarkeit von Datenbanken. Deshalb bietet Oracle viele Technologien an, um die Daten in der Datenbank zuverlässig aufzubewahren und zu jeder Zeit Zugriff darauf zu ermöglichen. Dabei stehen nicht nur ungeplante Ausfälle von Rechnern im Fokus, sondern auch die Bereitstellung der Daten während eines Updates.

Oracle fasst alle diese Technologien in der Oracle Maximum Availability Architecture (MAA) zusammen und bietet für jeden Bereich die passenden Informationen mit entsprechenden Best Practices. Neuerdings ist es dabei auch egal, wo sich die Daten befinden – ob im Unternehmen oder in einer Public Cloud.

Mit jeder eingesetzten Funktionalität gehen andere Voraussetzungen und auch unterschiedliche Anforderungen an den Betrieb einher, sodass man leicht die Übersicht verlieren kann. Außerdem eignet sich nicht jede Technologie für jede Ausfallart gleich gut. Ebenfalls benötigt nicht jede Applikation und Datenbank dieselben Sicherheitsmaßnahmen. Denn je mehr Absicherung gewählt wird, desto mehr Aufwand und natürlich auch Kosten stecken dahinter.

Um die passende Hochverfügbarkeits-Architektur zu wählen und damit den Aufwand abzuschätzen, sollte erst einmal für jede Applikation die Auswirkung von Datenverlust geschätzt werden. Dabei sind insbesondere drei Kennzahlen von höchstem Interesse:

- *Recovery Time Objective (RTO)*
  Wie lange darf die Datenbank nicht verfügbar sein
- *Recovery Point Objective (RPO)*
  Welcher Datenverlust kann toleriert werden
- *Performance*
  Wie viel Performance-Verlust ist im Falle eines Problems in der Umgebung akzeptierbar

Sind diese Umstände für jede Applikation analysiert, kann man auch die passenden Oracle-MAA-Technologien auswählen. Damit das nicht zu komplex wird und nicht das Risiko besteht, einen Fehler bei der Konfiguration und Implementation zu machen, bietet es sich an, die Datenbanken in unterschiedliche Risiko-Gruppen einzuteilen. Diese ordnet man dann entsprechenden Referenz-Architekturen zu. Einen Vorschlag solcher Referenz-Architekturen bietet Oracle MAA *(siehe Abbildung 1)*.

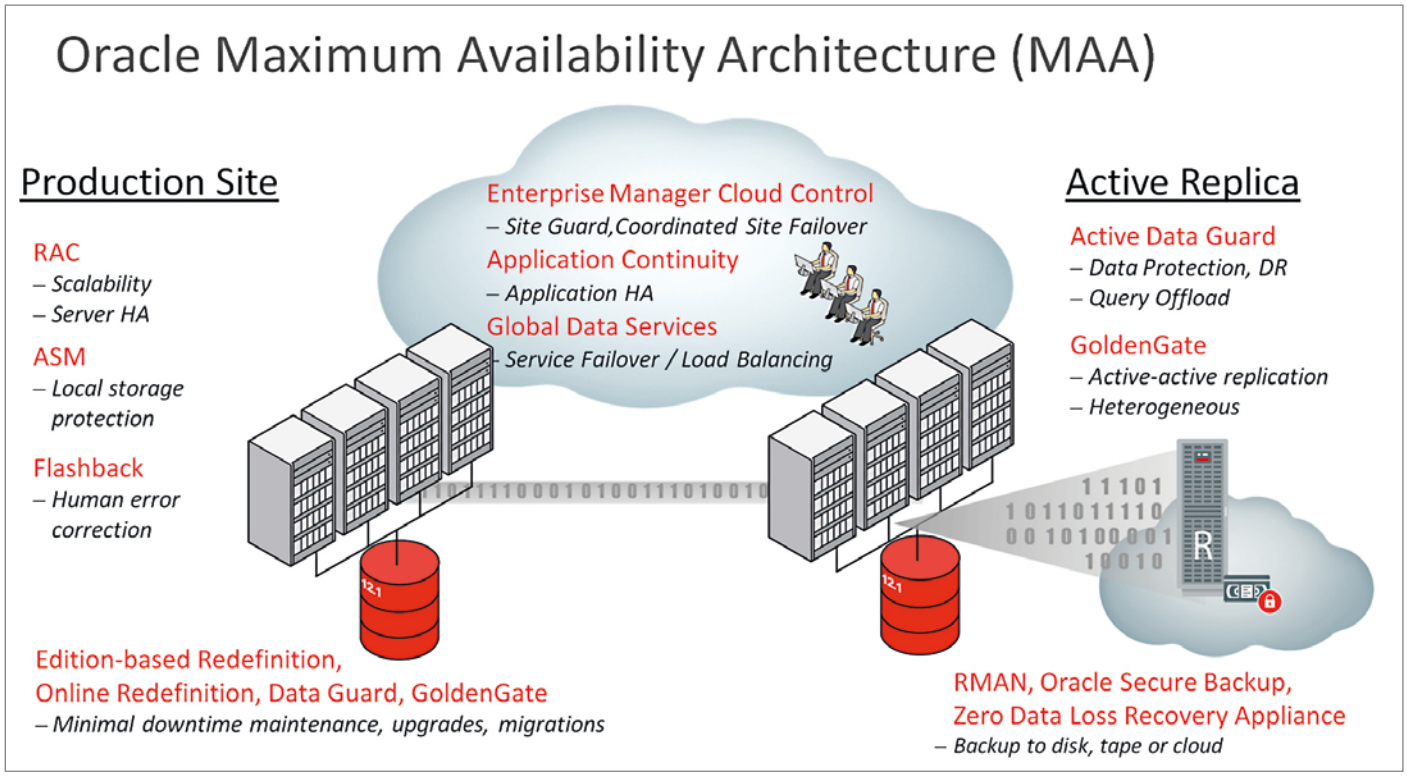


Abbildung 1: Oracle-MAA-Technologien

Oracle gruppiert dazu die unterschiedlichen Technologien in vier Level: Bronze, Silber, Gold und Platin. Letztendlich ist dies jedoch nur ein Vorschlag; welche Technologie man in welchem Bereich einsetzen möchte, ist letztendlich jedem selber überlassen. In jedem Fall ist es sinnvoll, sich auf einige wenige Level zu beschränken.

Bronze: RTO im Stunden-Bereich, RPO auf das letzte Backup

Im Bronze-Level sieht Oracle eine einfache Single-Instanz-Datenbank. Für das Erreichen der Verfügbarkeit dienen die integralen Bestandteile der Datenbank Enterprise Edition und als einzige zusätzliche Maßnahme das ganz normale Backup. Die reinen Basis-Funktionalitäten von Oracle-Datenbanken sollten allerdings nicht unterschätzt werden, denn viele laufen auch ohne zusätzliche HA-Maßnahmen jahrelang ohne Probleme. Eine Ursache dafür sind die in jeder Datenbank vorhandenen Mechanismen zum Schutz vor Daten-Korruption.

Um vor physikalischer Datenkorruption geschützt zu sein, verwendet die Datenbank Checksum-Verfahren, die man über die Parameter „DB_BLOCK_CHECKSUM“ und „DB_BLOCK_CHECKING“ aktiviert. Dazu wird eine Check-Summe während der Laufzeit ermittelt. Damit lassen sich Daten-Korruption im Memory verhindern, beim Lesen auf Platte Korruptionen erkennen und, wenn möglich, vor dem Schreiben vermeiden. Implizit dienen alle diese Mechanismen dazu, Daten-Korruption vorzubeugen.

Wird die Datenbank, wie von Oracle empfohlen, auf Automatic Storage Management (ASM) betrieben, kommen auch hier weitere Schutzmechanismen hinzu. Dazu gehören das Erkennen von Blockfehlern sowie die automatische Reparatur der Datenblöcke, wenn mit ASM gespiegelt wird. Noch

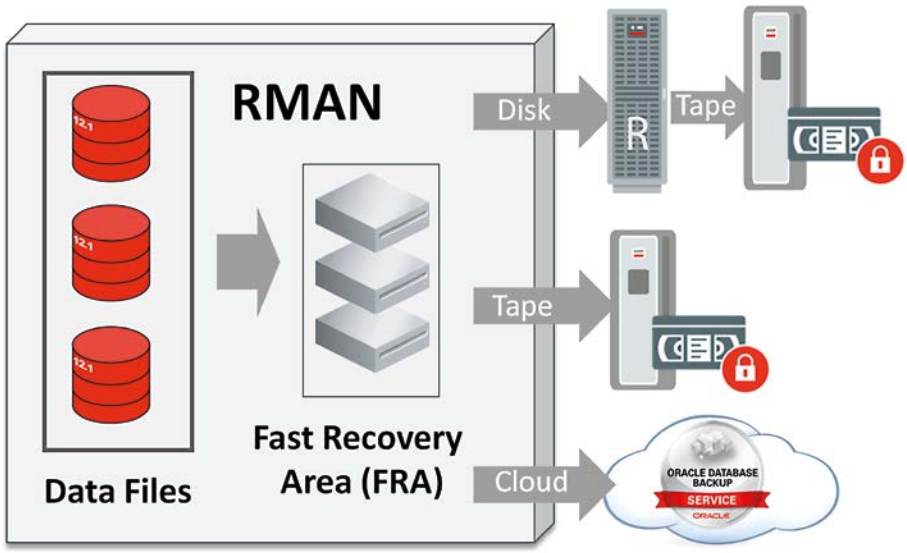


Abbildung 2: RMAN Backup RA, Tape und Cloud

weiter geht der Schutz bei den Engineered Systems, da hier der Storage und die Festplatten über Hardware Assitied Resiliant Data (HARD) ebenfalls in die Checksummen-Funktionalität einbezogen sind.

Zur externen Sicherung der Datenbanken dient der Recovery Manager, der die Datenbank im laufenden Betrieb sichert und somit generell eine vollständige Wiederherstellung erlaubt. Sollte das komplette System inklusive aktueller Redologs nicht mehr verfügbar sein, wäre der letzte Zeitpunkt zur Wiederherstellung das letzte Backup. Dabei kann neben dem Backup auf Festplatte oder Band-Laufwerken das Backup auch automatisch verschlüsselt in die Oracle Cloud gelegt sein. Damit wäre Oracle für das weitere Vorhalten der Sicherungen verantwortlich *(siehe Abbildung 2)*.

Noch weiter verbessern kann man die RMAN-Sicherung, wenn diese auf eine Zero Data Loss Recovery Appliance gelegt wird. Die Recovery Appliance prüft die Blöcke eines RMAN-Backups an vielen Stellen:

- Bei Sicherung
- Automatisch in bestimmten Zeit-Intervallen
- Beim Zurücksichern
- Beim Kopieren auf Band
- Beim Replizieren zu einer weiteren Recovery Appliance

Daneben bietet die Recovery Appliance für die Sicherung der Datenbanken auch an, komplett auf ein inkrementelles Backup-Verfahren umzustellen. Damit werden nicht nur die Datenbanken bei der Sicherung entlastet, sondern es kann auch viel Plattenplatz eingespart werden. Eine Recovery Appliance sichert viele Hunderte Datenbanken schnell und einfach. Trotz inkrementellen Backups ist das Wiederherstellen der Datenbanken so schnell wie bei einem Full Backup, die inkrementellen Backups müssen bei einer Recovery Appliance also nicht nachgefahren werden.

Noch ist das Backup auf eine Recovery Appliance in der Oracle-Cloud nicht verfügbar. Wenn dies jedoch im Laufe des nächsten Jahres der Fall sein wird, können auch große Datenbanken über schwächere Internet-Leitungen gesichert werden, da nur inkrementelle Backups auf die Recovery Appliance gelegt werden müssen. Es empfiehlt sich aber in jedem Fall, ein lokales Backup für das schnelle Zurücksichern und ein Backup in der Cloud oder auf Band extern für den Disaster-Fall vorzuhalten.

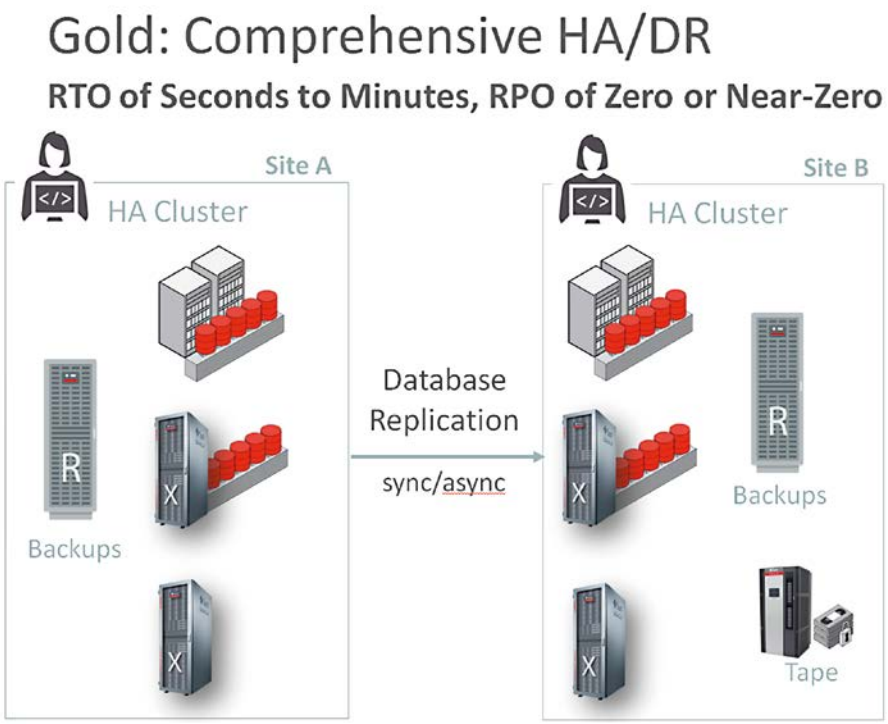


Abbildung 3: Gold Level

Zum Schutz vor menschlichem Versagen bietet Oracle die Datenbank-Flashback-Technologien. Damit lassen sich:

- einzelne Datensätze von Tabellen wiederherstellen
- komplette Transaktionen zurückrollen
- die komplette Datenbank in der Zeit zurücksetzen

Silber: RTO im Sekunden-Bereich für Server-Ausfälle, RPO vom letzten Backup

Zu den Technologien des Bronze-Levels kommt in der nächsten Referenz-Architektur hauptsächlich Oracle Real Applikation Clusters (RAC) hinzu. Dies ermöglicht, Server-Ausfälle im Sekunden-Bereich abzufedern und somit einen kontinuierlichen Zugriff auf die Datenbank zu erlauben. Das gilt nicht nur für ungeplante Ausfälle, RAC bietet auch die Möglichkeit der sogenannten „rollierenden Upgrades". Bei diesem Verfahren wird ein Knoten nach dem anderen im RAC gepatcht (Betriebssystem, Hardware-Wartung oder sogar Datenbank) und somit der Datenbank-Service für die Applikation ständig zur Verfügung gehalten.

RAC besitzt dabei neben der vollständigen Aktiv-Aktiv-Implementierung auch noch eine Abwandlung in Form eines Cold-Failover-Clusters, den sogenannten „RAC One Node". Dieser bietet dieselben Vorteile bei einem rollierenden Upgrade wie ein Aktiv-Aktiv-RAC, da die Instanz hier online während des Upgrades auf den anderen Knoten verschoben wird *(siehe „https://support.oracle.com/epmos/faces/DocumentDisplay?id=1593712.1")*.

RAC bietet auch für die neue 12*c*-In-Memory-Technologie der Datenbank eine Hochverfügbarkeitslösung an. Allerdings unterscheidet sich diese für Engineered Systems und generische RAC-Systeme. Auf den Engineered Systems stehen die In-Memory-Bereiche der Datenbank über die RAC-Knoten dupliziert zur Verfügung. Nach einem Ausfall kann sofort ohne jegliche Performance-Einbuße auf dem anderen Knoten weitergearbeitet werden.

Auf generischen Systemen sind die In-Memory-Bereiche auf die Knoten verteilt. Damit müssen die In-Memory-Bereiche des ausgefallenen Knotens erst einmal von Platte neu geladen werden. Dies geschieht zwar automatisch, allerdings dauert der Zugriff auf die Daten entsprechend länger, da, solange nicht alle Daten im Memory populiert wurden, auch auf einen Plattenzugriff ausgewichen werden muss *(siehe „http://*

www.oracle.com/technetwork/database/availability/dbim-maa-2658757.pdf").

Falls für das RMAN-Backup eine Recovery Appliance verwendet wird, empfiehlt es sich, für Datenbanken des Silber-Levels den Zero-Data-Loss-Modus der Recovery Appliance zu verwenden. Damit sendet die Datenbank ihre Online-Redologs direkt asynchron dorthin, um im Falle eines Komplett-Verlustes kaum Daten zu verlieren. Diese Funktion ist dabei Plattform- und Endianness-unabhängig, da im Gegensatz zu Data Guard das Onlinelog nur gespeichert und nicht angewandt wird.

Gold: Echtzeit-Datensicherheit mit Data Guard, Replikation mit Golden Gate

Im Gold-Level werden die RAC-Datenbanken zusätzlich mit Data Guard und/oder Replikation gesichert. Interessanterweise findet man hierzu auch die meisten Whitepaper der MAA-Architektur. Dies liegt daran, dass es viele Best Practices für Data Guard gibt, nicht nur im Bereich des Zero-Data-Loss-Modus *(siehe Abbildung 3)*.

Da Data Guard generell nur ein Siebtel des Netzwerk-Volumens gegenüber Storage-basierten Replikations-Mechanismen benötigt und nebenbei auch die gesendeten Blöcke prüft, ist diese Lösung ein viel besserer Schutz als herkömmliche Standby-Varianten, wie man sie häufig von Drittanbietern findet. Außerdem ermöglicht Data Guard für geplante Wartungsarbeiten auch das rollierende Upgrade über Versions-Grenzen hinweg. In diesem Umfeld hat Data Guard in der Version 12*c* viele Verbesserungen gebracht, da nun fast alle Daten-Typen bei diesem Vorgehen unterstützt werden. Somit können zukünftige Upgrades problemloser mit „Transient Logical Standby" und dem „RDBMS_ROLLING"-Package der Datenbank durchgeführt werden. Ebenfalls neu bei Data Guard ist, die Standby-Datenbank nun recht einfach in die Oracle Public Cloud stellen zu können. Damit liegt die Datensicherung bei Oracle selbst *(siehe „http://www.oracle.com/technetwork/database/availability/dr-to-oracle-cloud-2615770.pdf")*.

Neben vielen neuen Whitepapern gibt es seit Kurzem auch ein Utility („oratcptest"), um die Netzwerk-Umgebung zu prüfen. Dies ist insbesondere interessant, um die Latenzzeiten und die notwendige Bandbreite für alle Datenbanken zu ermitteln, die mit Data Guard gesichert werden sollen. Selbstverständlich kann damit auch die Verbindung zur Oracle Public Cloud getestet werden. Weitere Whitepaper zu Data Guard Tuning, Data Guard und Multitenant sowie Informationen zu „oratcptest" gibt es hier:

- Data Guard Synchronous Redo Transport: *(siehe „http://www.oracle.com/technetwork/database/availability/sync-2437177.pdf")*

- Data Guard Asynchronous Redo Transport: *(siehe „http://www.oracle.com/technetwork/database/availability/async-2587521.pdf“)*
- Data Guard Redo Apply: *(siehe „http://www.oracle.com/technetwork/database/availability/redo-apply-2745943.pdf“)*
- Switchover and Failover: *(siehe „http://www.oracle.com/technetwork/database/availability/maa-roletransition-bp-2621582.pdf“)*
- MOS Note 2064368.1 Measuring Network Capacity using oratcptest: *(siehe „https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=2064368.1“)*
- MOS Note 1916648.1 Using Deferred PDB Recovery and STANDBYS=NONE with Oracle Multitenant: *(siehe „https://support.oracle.com/epmos/faces/DocumentDisplay?id=1916648.1“)*
- MOS Note 2049127.1 Data Guard with Oracle Multitenant: *(siehe „https://support.oracle.com/epmos/faces/DocumentDisplay?id=2049127.1“)*

Eine physikalische Replikation, wie sie Data Guard vornimmt, ist manchmal nicht ausreichend. In den Fällen, in denen eine aktive logische Replikation gewünscht ist, bietet sich Oracle Golden Gate an. Es ermittelt aus den Datenbank-Redologs die jeweiligen Transaktionen und fährt diese im aktiven offenen Standby-System nach. Im Gegensatz zu Data Guard ist es immer asynchron und aufwändiger zu implementieren, bietet aber den Vorteil, unterschiedlichste Versionen zu unterstützen, nicht nur die Oracle-Datenbank. Häufig findet man Golden Gate als Ergänzung zu Data-Guard-Umgebungen, da es das rollierende Patching noch verbessert und im Gegensatz zu Data Guard auch eine Fall-Back-Lösung implizit mit anbietet.

Selten sind Hochverfügbarkeits-Umgebungen auf Datenbanken beschränkt und beim Umschalten auf die Standby-Datenbank ergibt sich die Notwendigkeit, auch Applikationen und Applikationsserver umzuschalten. Damit dies nicht manuell geschehen muss, bietet der Enterprise Manager den sogenannten „Site Guard“ an, der den kompletten Umschaltvorgang automatisieren kann. Deswegen gehört Site Guard ebenfalls mit zur MAA-Technologie und findet häufig in der Gold-Referenz-Architektur seine Daseins-Berechtigung.

Platin: Hochverfügbarkeit bis zur Applikation, Zero Data Loss auf große Distanzen

In der letzten Referenz-Architektur finden sich neben den bisher genannten Technologien auch Funktionen, damit Applikationen ohne Unterbrechung bei geplanten und ungeplanten Ausfällen weiterarbeiten können. War es Applikationen vor der Datenbank 12*c* nicht möglich, einfach weiterzuarbeiten, da der Transaktionsstatus der letzten Transaktion nicht ersichtlich war, so löst Application Continuity nun dieses Problem *(siehe „http://www.oracle.com/technetwork/database/options/clustering/application-continuity-wp-12c-1966213.pdf“)*. Eine weitere Technologie, um Applikationen auch das Arbeiten in unterschiedlichen Applikationsversionen zu erlauben, ist Edition Based Redefinition. Dabei kann die Applikation zur Laufzeit auf unterschiedliche Applikations-Packages zugreifen. Dies muss aber von der Applikation programmiert worden sein und geht deshalb etwas über reine Datenbank-Funktionalität hinaus. Es ist auch ein gutes Beispiel dafür, dass die Platin-Technologien durchaus mehr Aufwand erfordern, möchte man wirklich vom kompletten Technologie-Spektrum profitieren.

In der Platin-Referenz-Architektur mit Active Data Guard Far Sync wird für die Datenbanken die synchrone Replikation über weite Entfernungen möglich. Die Master-Master-Replikation mit Golden Gate findet ebenso Einzug in die Referenz-Architektur, da dies auch ein sukzessives Upgrade von Applikationen ermöglicht. Damit können Benutzer auf der alten Datenbank weiterarbeiten, während neue Benutzer schon gegen die neue Umgebung verbunden sind *(siehe „http://www.oracle.com/technetwork/database/availability/ogg-adg-2422372.pdf“)*. Global Data Services kann insbesondere bei letztem Vorgehen die Funktionalität erweitern, da es das Routing für Applikationen an den besten Service übernimmt.

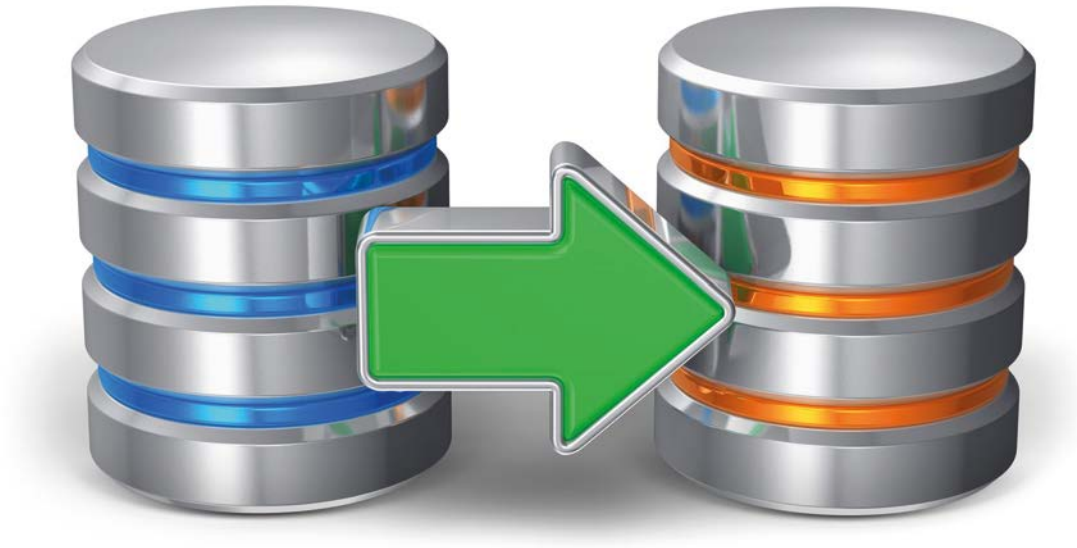
Fazit

Welche MAA-Technologien letztendlich für welche Datenbanken zur Verfügung stehen und wie die unterschiedlichen Level dabei aussehen, entscheidet jeder selbst. Das komplette Spektrum ist nur bei wenigen Kunden im Einsatz. Die Oracle-MAA-Technologie zeigt aber die Möglichkeiten auf und gibt gleichzeitig die Grenzen der einzelnen Technologien an.

Für Oracle hat MAA einen recht hohen Stellenwert, wie man auch an der Vielzahl der Whitepaper erkennen kann, die in der letzten Zeit veröffentlicht wurden. Die aktuelle Oracle-12*c*-Version hat schon viele Verbesserungen im MAA-Bereich gebracht, als Beispiel sei nur die fast 100-Prozent-Unterstützung aller Datentypen beim Versions-Upgrade genannt. Auch mit neuen Datenbank-Versionen wird es weitere Entwicklungen in diesem Bereich geben, so sind zum Beispiel einige Beschränkungen von Application Continuity bald nicht mehr relevant.

Die MAA-Seite *(siehe „http://www.oracle.com/goto/maa“)* hält immer aktuelle Informationen parat und auch auf der deutschsprachigen Datenbank-Community *(siehe „https://blogs.oracle.com/dbacommunity_deutsch/“)* gibt es immer aktuelle MAA-Informationen.

Sebastian Solbach
sebastian.solbach@oracle.com



Backup-Verfahren in der Praxis – optimiert und intelligent

Roland Stirnimann, Trivadis AG

Ein zuverlässiges Backup-Verfahren ist für jedes Unternehmen ein Muss. Dieser Artikel erläutert aktuelle Herausforderungen im Backup-Kontext und beschreibt einen bewährten Ansatz aus der Praxis, um diese zu meistern.

Oracle selbst bietet von Haus aus verschiedene technische Möglichkeiten zur Ressourcen-Optimierung, etwa in Bezug auf die Volumen-Reduktion. Dieses Potenzial nutzen bereits viele Unternehmen, doch darüber hinaus ist viel mehr möglich. Durch intelligente Automatisierung und Standardisierung bietet ein modernes Backup-Verfahren massive Mehrwerte bezüglich Transparenz, Risiko-Minimierung und Effizienz beim täglichen Betrieb.

Hintergrund

IT-Ausfälle und Datenverlust sind für Unternehmen ein Albtraum. Häufig sind Software- oder Netzwerk-Probleme, aber auch menschliche Fehler mögliche Ursachen. Die Kosten können – je nach Kritikalität und Dauer – in die Millionenhöhe gehen. Entsprechende Risiken gilt es durch ein verlässliches „Backup & Recovery"-Verfahren der Datenbanken zu minimieren. Trotz angemessener Sicherheit soll sich allerdings die Effizienz nicht verschlechtern, da mehr Sicherheit oft in Verbindung mit mehr Daten und Ressourcen gesehen wird und somit letztendlich Mehrkosten verursacht. Es stellt sich also die Frage: Wie können Risiken minimiert werden, ohne die Effizienz zu vernachlässigen? Ein optimales Backup-Verfahren ist erforderlich, das so wenig sichert wie möglich, aber so viel wie nötig.

Aktuelle Backup-Situation im Unternehmen

Die Sicherung von Datenbanken wird in den Unternehmen oft sehr pauschal durchgeführt und generiert deshalb unverhältnismäßige Betriebskosten. Alle sind sich einig, dass es ein Backup braucht, jedoch ist es in der Praxis häufig schwierig, Argumente zu finden, um Investitionen in Verbesserungen/Erneuerungen tätigen zu können. Erst im Fehlerfall werden Entscheider und Business wachgerüttelt und sich der Wichtigkeit einer zuverlässigen, performanten Sicherung und Wiederherstellung bewusst. Die nachfolgende Zusammenstellung nennt aktuelle Problemzonen aus der Praxis, denen es zu begegnen gilt:

- *Backup-Performance*
  Jede Sicherung generiert spürbare Systemlast, primär I/O-Operationen, und Datenvolumen im entsprechenden Speichersystem. Da bei einigen Storage-Herstellern volumenbasiert zu lizenzieren ist, lohnt es sich, beim Back-

up-Volumen genauer hinzuschauen. Das Volumen ist jedoch nur eine Seite der Medaille, denn die Last durch das Lesen der Daten ist in Bezug auf die Datenbank-Performance und letztendlich für die Applikation entscheidend. Backup-bedingte Lese-Operationen belasten direkt das Speichersystem der Datenbank, während die eigentlichen Sicherungsdateien meistens in ein dediziertes System geschrieben werden. In der Praxis ist die Backup-Last zudem oftmals schlecht verteilt, sodass massive Engpässe entstehen, die sich negativ auf die Performance auswirken.

- *Backup-Steuerung*
  Die Steuerung und somit die Verteilung der Backup-Jobs ist bei mehreren Hundert Datenbanken nicht ganz einfach. Häufig fehlt der Überblick darüber, was wann startet, und so kommt es zu Lastspitzen aufgrund zu vieler paralleler Backup-Operationen. Ein weiteres Problem der verschiedenen Job-Steuerungen sind die statischen Sicherungs-Intervalle. Die Steuerung kann nicht auf veränderte Situationen reagieren, etwa wenn unerwartet viele Transaktionslog-Daten geschrieben werden und der Plattenplatz ausgeht. Dies führt unweigerlich zum Stillstand der Datenbank. Ein weiteres Beispiel sind wachsende Datenbanken, deren Backup-Dauer zunimmt, wodurch die Sicherungen langsam aber sicher die dafür vorgesehenen Zeitfenster überschreiten und den Geschäftszeiten in die Quere kommen.

- *Backup-Verwaltung*
  Selbst große Umgebungen mit Hunderten von Datenbanken verwenden häufig lokale Scheduler wie Cron oder den Windows Task Scheduler. Dies erschwert die Backup-Verwaltung, wenn der Administrator beispielsweise fünf Datenbank-Server in ein Wartungsfenster versetzen möchte und dazu auf jeden Server einzeln verbinden muss. Generell finden viele Aufgaben direkt auf jedem einzelnen Datenbank-Server statt und kosten den Administrator Zeit und Nerven. Auch bei der Konfiguration unterschiedlicher Backup-Anforderungen (wie seitens SLA) in großen Umgebungen ist es eine Herausforderung, den Überblick jederzeit zu behalten.

- *Überwachung und Reporting*
  In irgendeiner Form existiert meistens eine Backup-Überwachung. Doch was kann diese beziehungsweise was kann sie nicht? Häufig basiert die Überwachung auf Mail-Nachrichten, die im Fehlerfall versendet werden. Übergreifende Ad-hoc-Auswertungen zum Backup-Volumen, zu Backup-Laufzeiten, zu laufenden und fehlgeschlagenen Sicherungen müssen manuell erstellt werden und sind somit zeitaufwändig. Auch den unterschiedlichen Zielgruppen als Report-Empfänger wird wenig Beachtung geschenkt. Interessante Auswertungen für Management und Entscheider fehlen gänzlich oder lassen sich nur mühsam in manueller Arbeit erstellen. Umfassende Transparenz auf Knopfdruck ist vielerorts ein Wunschdenken.

- *Wiederherstellung*
  Weil ein Backup seit Jahren funktioniert, ist dies keine Garantie, dass auch die Wiederherstellung einwandfrei klappt. Die Erfahrung zeigt, dass solche Vorgänge länger dauern als angenommen oder nicht so reibungslos funktionieren wie gedacht. Dies führt zu längeren Ausfallzeiten im Fehlerfall und letztendlich zu mehr Kosten.

Anwendungszwecke einer Sicherung

Bevor die Frage geklärt wird, wie eine optimale Sicherung aussehen könnte, sollte sich ein Unternehmen darüber im Klaren sein, wozu die Backups überhaupt verwendet werden. Natürlich kommt den meisten Leuten der Desaster-Restore als erster Gedanken in den Sinn. Jedoch kann ein Backup in der Praxis für weit mehr verwendet werden, wodurch je nach Anwendungsfall unterschiedliche Anforderungen bestehen.

Insbesondere für den Desaster-Fall glauben teilweise Administratoren, dass eine Hochverfügbarkeitslösung wie Data Guard ein Backup ersetzt. Diese Aussage ist natürlich nicht allgemein gültig. Klar ist die Hochverfügbarkeitslösung die erste Wahl, wenn beispielsweise das primäre Rechenzentrum ausfällt. Reelle Beispiele aus der Praxis lehrten aber den Autor allerdings, dass Murphy's Law existiert, im Fehlerfall auch die Hochverfügbarkeitslösung versagen kann und somit ein Unternehmen einen Plan B braucht, das Backup.

Ein weiterer Anwendungsfall für das Backup ist die Wiederherstellung der Datenbank nach logischen Korruptionen, etwa nach einem fehlerhaften Applikations-Update. Die Datenbank ist eigentlich noch funktionsfähig, jedoch ist der Inhalt (Daten) fehlerhaft. Hier kommt die klassische Wiederherstellung auf einen bestimmten Zeitpunkt zur Anwendung (Point-in-time-Recovery). Das Klonen von Datenbanken kann ebenfalls auf RMAN-Sicherungen basieren.

Für das Duplizieren von Datenbanken können als Beispiele zwei Gründe ausgemacht werden: Erstens werden produktive Daten in die Entwicklungsumgebungen geklont oder zweitens wird ein bestimmter Datenstand der Produktion geklont zwecks Fehler-Analyse. Letzteres kann sehr hilfreich sein, indem ein Klon vor einer bestimmten Fehler-Situation erstellt wird und die Datenstände verglichen werden können.

Möglichkeiten zur Backup-Optimierung mit Oracle-Bordmitteln

Oracle bietet von Haus aus schon seit Längerem diverse Funktionalitäten, um Backup- und Restore-Vorgänge zu optimieren. Inkrementelle Sicherungskonzepte helfen bei der Reduktion des Backup-Volumens (Output) und belasten somit den Backup-Storage weniger. Die gesamte Datenbank wird somit nicht mehr täglich gesichert, sondern nur alle paar Tage.

In der Oracle Enterprise Edition steht zusätzlich die Funktionalität «Block Change Tracking» zur Verfügung. Damit lässt sich auch das Input-Volumen, also die Lese-Operationen, massiv verringern, indem bei inkrementellen Backup-Operationen nur die effektiv geänderten Blöcke gelesen werden anstatt der gesamten Datenbank.

Weitere Optimierungen beim Backup-Output sind durch Komprimierung möglich, die direkt in RMAN eingebaut ist. Jedoch ist beim Aktivieren der Komprimierung darauf zu achten, dass je nach verwendetem Algorithmus die Lizenz für die Advanced-Compression-Option zu erwerben ist. Lediglich die Komprimierungsstufe „BASIC" ist in Standard Edition und Enterprise Edition ohne Folgekosten verfügbar. Die Stufen „LOW", „MEDIUM" und

HIGH“ können ausschließlich für die Enterprise Edition als Option erworben werden.

Für die zentrale Steuerung, Konfiguration, Überwachung und Verwaltung der Backups bietet sich Oracle Enterprise Manager an. Daneben gibt es auf dem Markt diverse Tools zur zentralen Steuerung, jedoch sind diese bei der Verwaltung der Backups häufig limitiert und dem DBA eher fremd.

Letztendlich ist bei zentralisierten Lösungen die Verfügbarkeit sehr wichtig, denn bei den meisten Lösungen muss der zentrale Management-Server verfügbar sein, damit die Agenten die Sicherungsaufgaben ausführen. Ein längerer Stillstand des Management-Servers kann somit zum Datenbank-Stillstand führen (wie Archiver Stuck) und stellt einen Single-Point-of-Failure dar.

Im Bereich der Restore- und Volumen-Optimierung bietet Oracle die Zero Data Loss Recovery Appliance (ZDLRA) an. Dieses Engineered System verringert das Backup-Volumen, indem lediglich ein initiales Komplett-Backup erforderlich ist. Danach reichen ausschließlich inkrementelle Sicherungen aus. Basierend auf der initialen Komplettsicherung und den nachfolgenden Änderungen kann sich die Appliance über die logische Verknüpfung der Datenbank-Blöcke auf jeden Zeitpunkt innerhalb der Aufbewahrungsdauer ein „Incremental Level 0“-Backup zusammenbauen. Dieses logische Gebilde wird gegenüber RMAN präsentiert, wodurch jede Wiederherstellung ohne inkrementelle Sicherungen auskommt. Insbesondere ist auch die Recovery-Dauer mit Archive-Log-Dateien sehr kurz, weil die logische Komplettsicherung der Appliance sehr nahe am gewünschten Recovery-Zeitpunkt liegt. Dieser Umstand schlägt sich in einer kürzeren Wiederherstellungsdauer spürbar nieder. Zudem validiert die Appliance im Hintergrund kontinuierlich die Konsistenz der Blöcke und entdeckt so Korruptionen frühzeitig. Natürlich lässt sich eine ZDLRA äußerst performant via InfiniBand anbinden und wirkt somit auch auf Netzwerk-Ebene potenziellen Engpässen entgegen.

Die intelligente und zentrale Backup-Steuerung

Trotz all der hilfreichen technischen Funktionen wurde bisher eine Komponente zur Optimierung nicht betrachtet, nämlich die Steuerung der Backup-Vorgänge. Fragen Sie sich einmal: Wann tanke ich mein Auto? Jeden Montag exakt um sieben Uhr oder wenn der Tankinhalt zur Neige geht. Wohl eher das Letztere oder mit anderen Worten: Sie tanken bedarfsgerecht, wenn es nötig wird. Genauso verhält sich eine moderne Backup-Steuerung, die auf starre Sicherungs-Intervalle verzichtet und stattdessen anhand von Richtlinien entscheidet, wann ein Backup erfolgen muss. Für diese auf Richtlinien basierende Entscheidung braucht die zentrale Steuerung einen intelligenten Algorithmus. *Abbildung 1* zeigt grafisch die Architektur einer möglichen und bewährten Implementation, die spürbar die Anzahl der Backup-Vorgänge und somit die Gesamtlast reduziert.

Auf jedem Datenbank-Server läuft ein Agent, der kontinuierlich die Situation zur System- und Datenbank-Auslastung ermittelt und an das zentrale Repository sendet. Darunter fallen beispielsweise der CPU-Verbrauch, die I/O-Belastung, das Transaktions-Volumen, der Füllgrad der Archive-Log-Destination, die Rolle der Datenbank (Primary oder Standby) etc.

Im Repository selbst läuft in kurzen Abständen ein Algorithmus in Form eines Datenbank-Jobs, der aufgrund der aktuellen Situation und der definierten Richtlinien für eine bestimmte Datenbank entscheidet, ob ein Backup erfolgen soll oder nicht. Entsprechende Backup-Jobs werden erstellt, vom Agenten abgeholt und ausgeführt. Abschließend meldet der Agent den Status über das Backup an das Repository zurück und der Kreislauf schließt sich.

Jede Datenbank ist einer bestimmten Richtlinie zugewiesen und somit gilt es, initial zu überlegen, wie die Datenbanken kategorisiert werden sollen (Produktion, Test, DWH etc.). Mit anderen Worten widerspiegelt eine Richtlinie letztendlich die Anforderungen an eine bestimmte Kategorie. Nachfolgend werden abschließend ein paar Anforderungen genannt inklusive eines Beispiels, wie sich diese auf die Backup-Steuerung auswirken:

- *Zeitfenster für die Sicherung*
  „Incremental Level 0"-Backup-Operationen sollen nicht zu Geschäftszeiten laufen. Deshalb definiert die Richtlinie ein erlaubtes Backup-Zeitfenster über die Nacht.

- *Transaktionsvolumen*
  Es werden Schwellwerte dafür definiert, dass beispielsweise nach 30 GB Transaktionsvolumen beziehungsweise bei einer Änderungsrate von 40 Prozent ein komplettes oder ein inkrementelles Backup erfolgen soll.

- *Füllgrad der Archive-Log-Destination*
  Beim Überschreiten eines Schwellwerts in Prozent wird eine Archive-Log-Sicherung angestoßen. Somit kann diese Lösung dynamisch auf ein verändertes Lastverhalten reagieren, indem die Sicherungs-Intervalle temporär verkürzt werden und dadurch einen potenziellen Stillstand durch einen Archiver-Stuck verhindern.

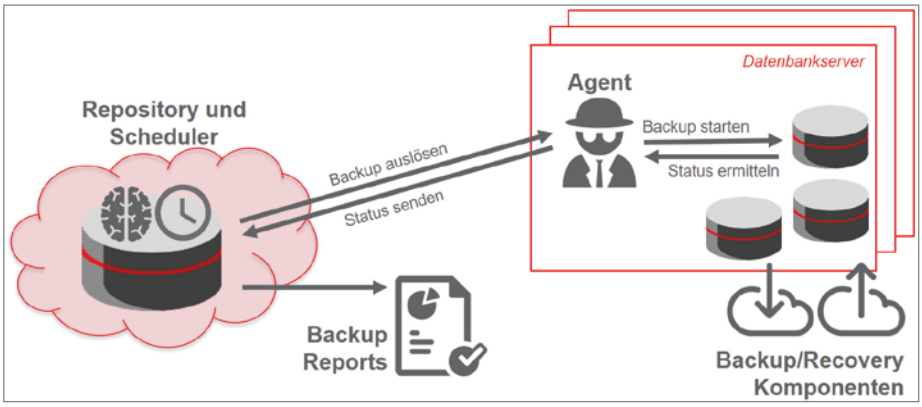


Abbildung 1: Architektur einer intelligenten Backup-Steuerung

- *CPU- und I/O-Last auf RAC-Nodes*
  Durch eine maximale Obergrenze der CPU-Auslastung kann ein Backup komplett unterbunden werden, um ein System nicht vollends in die Knie zu zwingen. Zudem sollte im RAC-Umfeld dynamisch jener Knoten mit der geringsten Last für das Backup gewählt werden.

- *Recovery Point Objective (RPO)*
  Die Vorgabe des maximal akzeptierten Datenverlusts definiert für den Scheduler implizit die maximale Intervalllänge zwischen zwei Archive-Log-Backups.

Eine intelligente Backup-Steuerung entscheidet aber nicht nur aufgrund der Anforderungen einer Richtlinie, sondern plant die Sicherungen innerhalb des konfigurierten Zeitfensters optimal ein. Dank dieser Verteilung der Backup-Operationen werden Performance-Engpässe vermieden und die Backup-bedingte Last ist ausgeglichener ohne massive Schwankungen.

Neben dem Herzstück der intelligenten Steuerung sollten bei der Implementation der Architektur in *Abbildung 1* weitere Punkte beachtet werden. Der Agent muss in der Lage sein, bei einem temporären Ausfall der zentralen Steuerung eigenständig weiter zu sichern, indem er sich lokal die Sicherungs-Intervalle der letzten Tage merkt und gleich weiterfährt. Dadurch wird einem Datenbank-Stillstand durch einen Archiver-Stuck vorgebeugt und das Risiko des Single Point of Failure eliminiert.

Ebenfalls sollten neue Datenbanken durch den Agenten automatisch erkannt, zumindest einer Standard-Richtlinie zugewiesen und somit gesichert werden. Die Praxis zeigt, dass es noch viele Erweiterungen gibt wie beispielsweise eine Schnittstelle zu Enterprise Manager für die Alarmierung oder das Ausführen alternativer Aktionen im Fehlerfall („Corrective Actions").

Der zentrale Ansatz bietet insbesondere im Bereich der Backup-Verwaltung riesige Vorteile, indem der DBA mit einem einzelnen Kommando beliebige Server/Datenbanken in den Wartungsmodus setzen kann. Auch im Sinne der Transparenz lässt die Lösung ihre Muskeln spielen, denn sämtliche Informationen rund um das Backup sind im Repository zentral gespeichert und können für Auswertungszwecke weiterverwendet werden.

Fazit

Die Praxis und die Erfahrung zeigen, dass die Nutzung von Oracle-Funktionalität bereits einiges an Optimierung ermöglicht. Doch erst die bedarfsgerechte und intelligente Steuerung der Backup-Operationen bringt die entscheidenden Mehrwerte in den Bereichen „Effizienz", „Risiko-Minimierung" und „Transparenz" zum Vorschein. Trivadis bietet mit dem Produkt „TVD-Backup" eine umfassende Backup-Lösung an, die mit der Komponente „Trivadis Intelligent Backup" (TIB) genau den erwähnten Mehrwert bringt.

Roland Stirnimann
roland.stirnimann@trivadis.com

Bulletproof fail over: Passive first!

Christoph Münch, virtual7 GmbH

Wer sich mit hochverfügbaren IT-Systemen beschäftigt, steht oft vor dem Problem, die zu planende Umgebung so aufzubauen, dass sie im besten Fall 24/7 verfügbar ist. Gleichzeitig ist es aber notwendig, darauf zu achten, dass sich dieses Gesamtwerk für Wartungsarbeiten sowie Backup-und Recovery-Maßnahmen eignet. Wer offen gegenüber völlig neuen Konzepten ist und gleichzeitig nicht nur Spaß an der IT hat, sondern auch eine gehörige Portion Humor mitbringt, den wird dieses revolutionäre Konzept sicher begeistern.

Um ein IT-System wie eine Web-Applikation für einen Kunden hochverfügbar bereitstellen zu können, stehen einem zwei grundsätzliche Konzepte zur Verfügung, die eine sehr hohe Ausfallsicherheit bieten. Diese beiden Topologien – Active-Passive sowie Active-Active – sind in diesem Artikel kurz beschrieben, um dann im Nachgang auf die offensichtlichen Vorteile des neuen, revolutionären Konzepts einzugehen. Die Basis der beschriebenen Varianten stützt sich jeweils auf eine vollständige Trennung auf zwei oder mehrere verteilte Data Center.

Active-Passive-Topologie

Als Erstes ein Blick auf die hochverfügbare Variante, bei der das Aufspannen des Clusters sich auf einen aktiven und einen passiven Standort verteilt *(siehe Abbildung 1)*. Die gesamte Last der Anwendung wird von einem Standort aufgenommen. Der zweite Standort ist redundant aufgebaut und steht im Fehlerfall bereit. Gibt es am Active-Standort einen nicht zu behebenden Ausfall, kann der passive Teil die gesamte Verarbeitung übernehmen. Er wird dann zum Active Data Center. Der fehlerhafte Strang kann somit überprüft beziehungsweise instand gesetzt werden, ohne eine für den Kunden transparente Beeinträchtigung zur Folge zu haben.

Zudem können mit diesem Konzept Wartungsarbeiten oder Ähnliches am Passiv-Rechenzentrum durchgeführt wer-

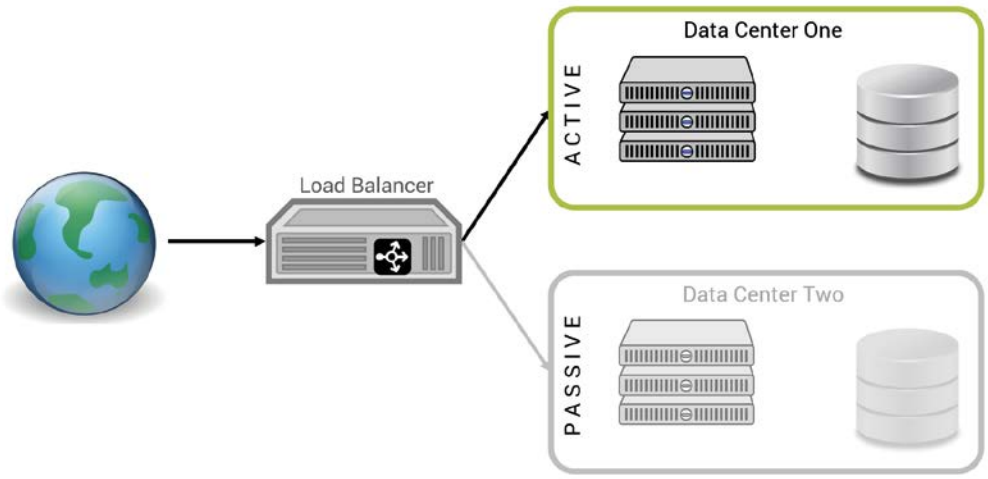


Abbildung 1: Active-Passive-Topologie

den, ohne einen Einfluss auf die Anwendung selbst zu haben. Kommt es während der Wartungsarbeiten im abgeschalteten Zweig zu einem Ausfall im aktiven Strang, ist ein Failover nicht möglich, was den Ausfall der gesamten Anwendung zur Folge hätte. Ein weiterer negativer Aspekt dieser Konstellation sind die hohen Hardware-Kosten. Jeder Standort im Cluster muss gleich stark ausgebaut werden, obwohl nur ein Data Center aktiv ist.

Active-Active-Topologie

Die zweite Variante, um eine Hochverfügbarkeit zu gewährleisten, ist die Active-Active-Topologie, bei der beide Teile eines Clusters online sind und die Last auf die Anwendung gemeinsam übernehmen *(siehe Abbildung 2)*. Damit erreicht man beim Ausfall eines Strangs eine Ausfallzeit der Anwendung von annähernd null, da das zweite Data Center nicht erst aktiv geschaltet werden muss, sondern die Anfragen sofort verarbeiten kann.

Zu den Nachteilen dieser Lösung gehört unter anderem, dass im Fehlerfall eines Knotens die Performance stark beeinträchtigt ist. Zudem kann nicht jedes Verfahren beziehungsweise jede Software in einem Active-Active-Umfeld betrieben werden.

Etabliertes und Bewährtes überdenken

Die beiden in kompakter Form beschriebenen Konzepte bringen einige Herausforderungen mit sich, die den Autor und seine Kollegen dazu gebracht haben, das ganze Thema neu zu betrachten: In Zeiten immer komplexer werdender System-Landschaften ist es auch immer komplizierter, diese zu betreiben. Um die Infrastruktur stabil und redundant zu halten, sind viele Experten erforderlich und somit wird der Personalaufwand entsprechend immer größer. Der Aufwand, dies alles zu monitoren und das Zusammenspiel der verschiedenen Komponenten aufeinander abzustimmen, stellt die Experten vor immer neue Herausforderungen. Die Hard- und Software-Lizenzen sowie der steigende Personalbedarf lassen die Kosten schnell ansteigen. Das hat eine kleine Gruppe erfahrener IT-Berater und Freidenker zum Anlass genommen, über ein komplett neues und revolutionäres Konzept nachzudenken.

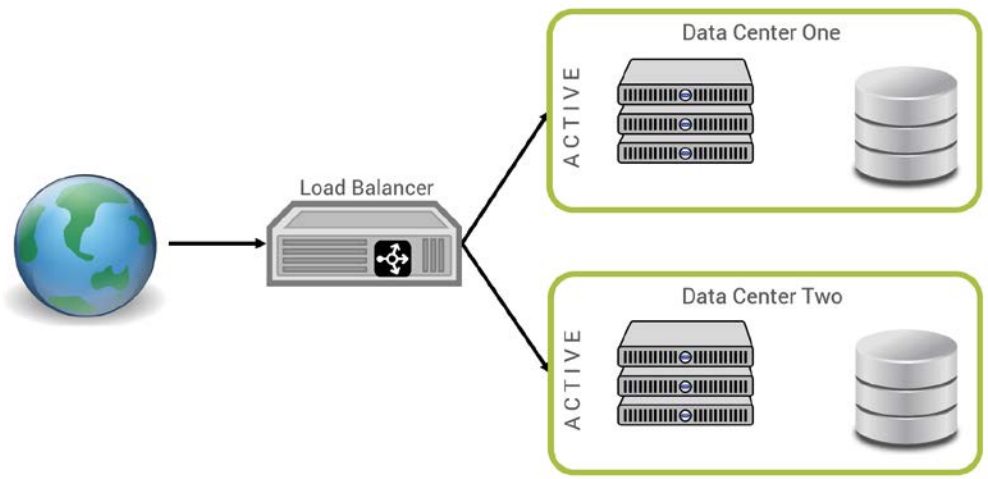


Abbildung 2: Active-Active-Topologie

Ideen, hin zur neuen, bahnbrechenden Strategie

Zu Beginn stand ein Brainstorming *(siehe Abbildung 3)*. Was sind die größten Pain-points der aktuellen Topologie und wie können sie beseitigt werden? Was sind die Eckpfeiler eines revolutionären, neuen Konzepts und welche Möglichkeiten ergeben sich daraus?

Am wichtigsten sind neben der Kostenreduzierung und der absoluten Datensicherheit sicherlich die Aspekte der Mitarbeiterzufriedenheit sowie Green-IT in Perfektion. Die Lösung kann nur zu einem Schluss führen, zur Passive-Passive-Topologie *(siehe Abbildung 4)*.

Ein fast fehlerfreies Konzept, das sowohl für kleine Unternehmen als auch bei allen Global Playern tragfähig wäre. Nachfolgend ist das bahnbrechende Konzept näher erklärt, um zu der Erkenntnis zu kommen: Bulletproof fail over lässt sich nur mit dem Ansatz „Passive-First" erreichen.

Outage-Driven-Architecture

Die gesamte Architektur basiert auf dem Konzept der Outage-Driven-Architecture (ODA). Bei diesem OPS-Pattern handelt es sich um das etwas unbekanntere Konzept, bei dem vor allem betrachtet wird, wie Systeme offline geschaltet werden können. Die gesamte Planung und Um-

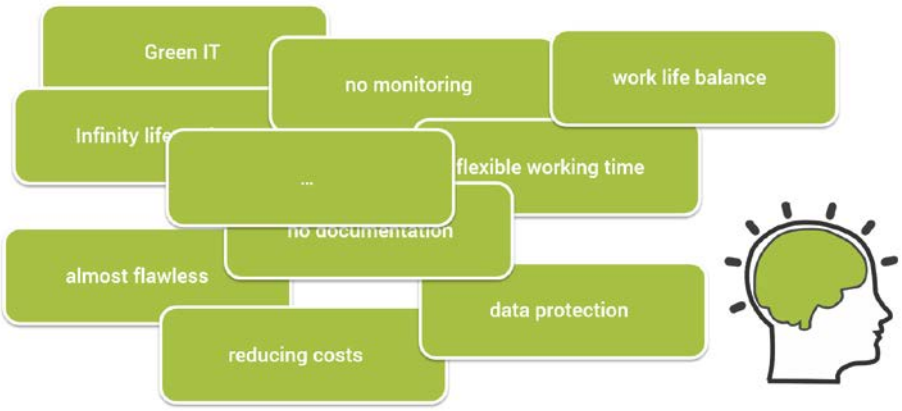


Abbildung 3: Bahnbrechende Ideen, die zu Passive-Passive führten

setzung der Architektur verfolgt einzig und allein das Ziel, dem Operations-Team die Arbeit so einfach und unkompliziert wie möglich zu machen.

Die Vorteile liegen somit klar auf der Hand. Eine hohe Mitarbeiter-Zufriedenheit und gutes Betriebsklima sind garantiert. Wie bei jeder Umstellung oder Neuausrichtung gilt auch bei der Einführung von ODA: Die Schwierigkeit ist nicht, die Betriebsmannschaft zu überzeugen, sondern auf Management-Ebene für das Konzept zu werben. Im Laufe des Artikels wird noch weiter auf diese Problematik eingegangen und man wird sicher das eine oder andere Argument zum Start in die ODA-Architektur finden.

Aufbau einer Non-Availability-Zone

Wer sich mit Anwendungen beschäftigt, die im firmeneigenen Netz, aber auch gleichzeitig im Internet erreichbar sind, weiß um die Schwierigkeit, vor allem die Verbindung ins Internet abzusichern. Um Kunden dennoch einen Zugriff aus der freien Welt zu bieten, wird getrennt durch Firewall und andere Sicherheitssysteme eine Demilitarized Zone (DMZ) aufgebaut. Auch dafür bietet Passive-Passive konzeptionell die beste Lösung. Es wird über die DMZ sowie das Intranet eine Non-Availability-Zone (NAZ) gespannt. Diese sorgt unter anderem für maximale Sicherheit.

Eine NAZ ist sowohl als Multi-Site Passive-Passive als auch als Single-Site Passive-Passive aufbaubar. Auch ein moderner Aufbau, basierend auf einer Container-Plattform, ist denkbar. Verfolgt man dieses Thema aktuell, gehen die Bestrebungen von Experten auf diesem Gebiet sogar hin zum Aufbau von No-Site-Passive-Passive-Umgebungen. Hier spricht man dann sozusagen von einem nahezu perfekten Aufbau einer Non-Availability-Zone.

Soft- und Hardware Lifecycle

Sicherlich ist jetzt noch nicht jeder Leser von dem Konzept überzeugt. Wer jetzt also immer noch nachdenkt und nicht mit einem Lächeln im Gesicht den Artikel liest, den werden hoffentlich die folgenden Eckdaten überzeugen. Dazu folgende Frage: „Wie organisieren Sie Ihre Soft- und Hardware Lifecycles?" In den meisten Fällen wird die Antwort sein, dass es zwar eine Herausforderung für das Betriebsteam ist und alles gut geplant sein muss, es aber mit wenigen Ausnahmen reibungslos läuft. Ausfälle gibt es sehr wohl ab und an, aber diese sind mit dem Service-Level-Agreement (SLA) meist vereinbar.

Passive-Passive bietet auch hier einen neuen, revolutionären Ansatz. Lifecycle-Maßnahmen müssen nicht mehr akribisch geplant sein. Es ist nicht mehr notwendig, Mitarbeiter an Wochenenden oder zu ähnlichen Zeiten bereitzustellen. Die Topologie bietet die Möglichkeit, fast immer und zu jeder Zeit Lifecycle-Maßnahmen an der Softwareund an der Hardware durchzuführen oder auch neue Software einzurichten.

Wenn der Passive-Passive-Ansatz in die Unternehmensstruktur richtig integriert ist, kann ganz leicht 99,9% Nichterreichbarkeit umgesetzt werden. Das wichtigste, wie in jeder anderen Topologie, ist auch hier das Monitoring. Selbstverständlich muss ein Monitoring stattfinden, um den Offline-Status zu überwachen. Im Allgemeinen gestaltet sich dies sehr einfach. Zustände wie „Server ist gestartet", „Die Anwendung reagiert nicht" gehören der Vergangenheit an. Offline ist offline! Somit kann auch das Betriebsteam sich auf das Passive-Sein konzentrieren.

Backup und Recovery

Im Rahmen heterogener Systemlandschaften kann es nicht das eine Backup- und Recovery-Konzept geben. Das Zusammenspiel jeder einzelnen Komponente muss betrachtet werden, um im Falle eines Fehlers eine geeignete Recovery-Maßnahme einleiten zu können.

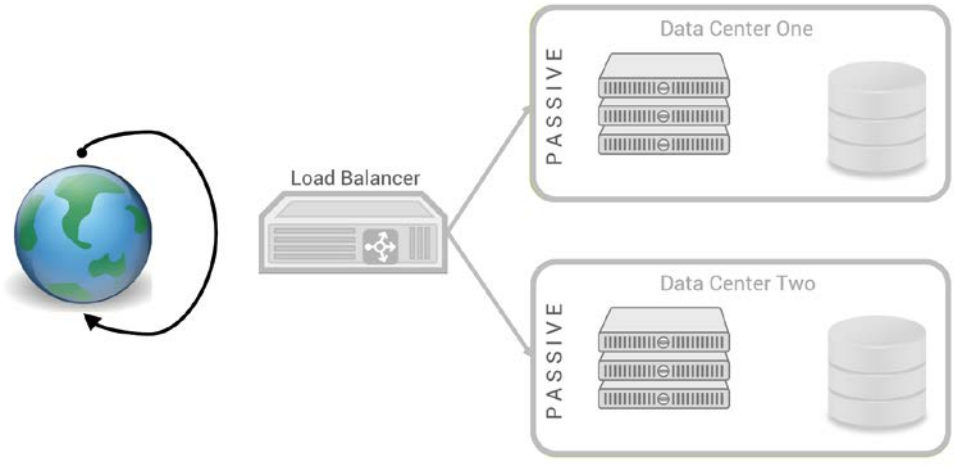


Abbildung 4: Passive-Passive-Topology

Auch hier gilt, wie schon für das Lifecycle-Management erörtert, Backup und Recovery ist zu jeder Zeit möglich. Einzuplanen sind hier nur die Zeitspannen, die erforderlich sind, um die betroffenen Systeme zu starten. Beachtet werden sollte, dass hier weniger mehr ist. Das soll bedeuten, je weniger Server gestartet werden, desto weniger Zeit muss für den Shutdown eingeplant werden.

Den größten Vorteil gibt es allerdings in Bezug auf die Backup- und Recovery-Strategie. Es ist völlig irrelevant, in welcher Reihenfolge man dieses Konzept angeht. Man kann entweder das Backup vor dem Recovery einplanen, so wie es schon seit der IT-Steinzeit umgesetzt ist, oder man geht konzeptionell noch unbeschrittene Wege und startet umgekehrt mit dem Recovery. Das Initial Recovery ist mangels Daten somit auch sehr performant.

Performance

Ein weiterer wichtiger Punkt ist die Performance und Skalierbarkeit. Wir beschäftigen uns oft mit Berechnungen der Anzahl von Prozessoren beziehungsweise Kernen oder damit, wie viel RAM für die Anwendungen zur Verfügung steht. Das alles spielt bei einem Passive-Passive-Aufbau keine Rolle mehr. Achtet man darauf, dass nicht zufällig und unerwartet Systeme starten, ist die Grenze nur der Plattenplatz (hier sei noch einmal auf die Wichtigkeit des Monitorings verwiesen).

Nimmt man als Beispiel das Betreiben von Application Servern, ist die Frage, wie viele Installationen auf die Platte beziehungsweise auf das angeschlossene Plattensystem passen. Es ist selbstverständlich darauf zu achten, dass die notwendige Uptime richtig einzuplanen ist. Es müssen sicher mehrere Komponenten zur gleichen Zeit hochgefahren sein. Zudem muss man sich keine Gedanken über die Bandbreite machen. Weder im Intranet noch im Internet kann es durch Passive-Passive zu Engpässen kommen. Hier gilt ganz klar „less data needs less band width".

Hohe Mitarbeiterzufriedenheit

In unserer heutigen Zeit, da IT-Experten in vielen Bereichen rar sind, muss es Ziel der Unternehmen sein, die angestellten Experten zu halten. Eine Möglichkeit, dies sicherzustellen, ist, darauf Wert zu legen, zufriedene Mitarbeiter zu beschäftigen. Auch hier bietet die Passive-Passive-Topologie ein Patentrezept.

Während des Passiv-Betriebs beschränken sich die Tätigkeiten der Administratoren darauf, sich selbst automatisiert startende Systeme hektisch wieder herunterzufahren. Produkte, die passiv betrieben werden, können jeden Bug verschleiern und machen somit ein Bugfixing völlig überflüssig. Dadurch gibt es weniger Reibungspunkte zwischen Entwicklungs- und Test-Teams. Installationen, Wartungen oder Ähnliches können auf ein für die Mitarbeiter sehr entspanntes Minimum reduziert werden.

Sollte sich dennoch Langeweile breit machen, kann man hier durch gezieltes Starten einzelner Systeme gegensteuern. Allgemein lässt sich hier festhalten, passive Systeme verhindern Fehlverhalten von Anwendern um nahezu hundert Prozent. Dies kann man frei nach Bob Marley mit dem Titel „No user, no cry!" zusammenfassen.

Passive-Passive und Green-IT

Wer die Ausführungen in den vorherigen Abschnitten aufmerksam gelesen hat, dem wird sehr schnell klar, dass die Passive-Passive-Implementierung quasi auf den Ideen der Green-IT basiert. Die Energiekosten stehen bei diesem Konzept mit im Vordergrund und werden ohne zusätzlichen Mehraufwand auf annähernd null reduziert. Jeder passiv betriebene Server spart somit bares Geld. Ebenfalls wird der Bedarf der Kühlung von Rechenzentren minimiert. Teure Wasserkühlungen oder Ähnliches gehören der Vergangenheit an. In Passive-Passive-Rechenzentren genügen selbst zu Spitzenzeiten gekippte Fenster oder eine offene Tür.

How many successful fail overs do you have a day?

Selbstverständlich hat das Konzept auch Schattenseiten. In betrieblicher Hinsicht muss einiges verändert werden, um von einem klassischen auf das Passive-Passive-Konzept umzustellen. Um auf einen lückenlosen, automatisierten Continuous-Fail-over-Prozess zu implementieren, den man unbedingt benötigt, um echtes Passive-Passive umzusetzen, sollte man über eine voll automatisierte Fail-over-Pipeline nachdenken. Damit stellt man bis hin zum automatisch rollierenden Shutdown einen reibungslosen Betrieb sicher.

Neue Rollen bringt die passive Landschaft

Wie so oft beginnt ein Paradigmen-Wechsel nicht nur mit Veränderungen in betrieblicher Hinsicht, sondern vor allem auch unter organisatorischen Gesichtspunkten. Man benötigt ganz neue Fähigkeiten und Rollen in seiner Organisation. Auf der technischen Ebene gibt es einen Downtime Insurance Engineer, der genügend Vorkenntnisse der klassischen Methoden der Hochverfügbarkeit mitbringen sollte. Nur so lässt sich gewährleisten, dass der Mitarbeiter ausreichend kreative Ideen entwickeln kann, um mit dem nötigen Respekt und Geschick ans Werk zu gehen. Ein Einsteiger würde sehr wahrscheinlich den humorigen Hintergrund nicht in vollem Umfang überblicken können.

Sinnvoll wäre sicher auch die Installation eines Accidental Boot Prevention Supervisors, der die notwendige Kontrollfunktion übernehmen sollte. Um auch nur eine geringe Chance zu haben, eine solche Idee auch strategisch im Unternehmen platzieren zu können, benötigt man selbstverständlich auch die Unterstützung durch das Top-Management. Hier wird empfohlen, über die Position des Chief Downtime Officers (CDO) nachzudenken. Dieser kann mit dem Blick für das ganze Desaster sehr einfach ein Hochfahren beziehungsweise In-Betrieb-Nehmen einzelner Komponenten oder im schlimmsten Fall ganzer Rechenzentren per Management-Entscheidung unterbinden. Der CDO ist somit die wichtigste Person der gesamten IT-Abteilung.

Fazit

Wer die Ausführungen zur Passive-Passive-Topologie bis jetzt gelesen hat und

der Meinung ist, dies sei auch ein tolles Konzept für sein Unternehmen, sollte mit der Ausschreibung eines CDO zu beginnen. Hoffentlich sind alle Leser mit dem Autor der gleichen Überzeugung, ein wenig Spaß schadet auch der IT nicht. Er würde sich freuen, alle Lesern mit diesem Artikel etwas unterhalten zu haben und wenn alle den Alltag auch mal mit einem kleinen Augenzwinkern betrachten. Wir leben alle IT und haben Spaß an der Arbeit, es kann also auch nicht schaden, den Alltag mit wirren Ideen etwas aufzulockern.

Weiterführende Links

International Passive-Passive User Group:
http://www.ippug.org

Der Autor bedankt sich für die fundierte fachliche Unterstützung bei folgenden Kollegen:

- Ralf Downtime Ernst
- Sturmius OfflineMostSecure Rippert
- Hans WhoNeedsMonitoring Mehl
- Thorsten NoContent Wussow
- Florian ClosedGate Stoll

Christoph Münch
christoph.muench@virtual7.de

Termine

Januar

Januar 2017
Regionaltreffen Bremen
regio-bremen@doag.org
Bremen

13.01.2017
DOAG Webinar zum Thema: "Oracle Database Cloud Performance"
sig-database@doag.org
online

19.01.2017
Regionaltreffen Nürnberg
Thema: **Oracle** RAC Deep Dive und RAC 12.2 New Features
regio-franken@doag.org
Nürnberg

Februar

26.01.2017
Regionaltreffen Stuttgart
regio-stuttgart@doag.org
Stuttgart

01.02.2017
Regionaltreffen München/Südbayern
regio-muenchen@doag.org
München

02.02.2017 - 03.02.2017
DOAG Noon2Noon
Upgrade nach 12c
sig-database@doag.org
Mainz

08.02.2017
DOAG 2017 DevCamp
Themen: "Development by Choice", Moderne Softwareentwicklung mit **Oracle**, ADF, APEX, JavaScript, Forms, Webcenter
sig-development@doag.org
Hannover

15.02.2017
DOAG 10. Primavera Community Day
bsc-primavera@doag.org
München

Oracle Database Appliance X6-2 – ein Erfahrungsbericht

Johannes Kraus, Herrmann & Lenz Services GmbH

Welche Firma kennt nicht die Situation der Hardware-Entscheidung? Die vorhandenen Server sind in die Jahre gekommen und abgeschrieben, sodass nun ein Ersatz angeschafft werden muss. Dabei soll natürlich die Leistung verbessert werden. Aufgrund des Zuwachses von immer mehr Kernen pro Sockel wird dieser gewünschte Effekt erzielt. Dieser Fakt hat jedoch auch einen kleinen Nachteil. Die Enterprise Edition, die linear von der Anzahl der vorhandenen Kerne abhängig ist, erfordert somit auch einen Zukauf an Lizenzen. Die SE2 Edition darf nur auf Server mit maximal zwei Sockeln betrieben werden und ist zusätzlich softwareseitig auf zwölf Threads begrenzt.

Aufgrund der erwähnten Problematik greifen viele Firmen im ersten Schritt auf die Virtualisierung der Datenbank-Server zurück. Doch genau in dieser Virtualisierung steckt bei der falschen Vorgehensweise die größte Gefahr einer Lizenzkosten-Explosion. Der Grund liegt darin, dass Oracle zwischen Soft- und Hard-Partitioning unterscheidet. Während bei Hard-Partitioning die CPU-Kerne auf dem Server begrenzt werden können, Stichwort „Processor Pinning", ist es bei Soft-Partitioning etwa unter VMware möglich, nach Belieben Kerne einer virtuellen Maschine hinzuzufügen. Aufgrund dieser Möglichkeit muss in den meisten Fällen der gesamte Server lizenziert werden.

Zusätzlich ist dabei zu beachten, dass bei Live-Migrationen, also der Verlagerung der virtuellen Server im laufenden Betrieb, die gesamte VMware-Umgebung lizenziert werden muss. Sind in dieser Umgebung physikalische Server vorhanden, die mehr als zwei Sockel vorweisen, ist eine Enterprise-Edition-Lizenzierung unumgänglich. Das Gravierende dabei: Es muss die gesamte VMware-Umgebung lizenziert werden.

Komponenten	ODA X6-2S	ODA X6-2M
Größe	Eine Höheneinheit	Eine Höheneinheit
Prozessor	Eine CPU mit 10 Kernen (Intel Xeon E5-2630 v4)	Zwei CPUs á 10 Kerne (Intel Xeon E5-2630 v4)
Speicher	Min. 128 GB Max. 384 GB	Min. 256 GB Max. 768 GB
Netzwerk	2 x 10GbE SFP+ (fiber) und 2 x 10GBase-T (bond)	2 x 10GbE SFP+ (fiber) und 4 x 10GBase-T (2x2 bond)
Boot-Festplatte	2 x 480 GB SSD (gespiegelt)	2 x 480 GB SSD (gespiegelt)
Storage	Optionen: A: 6,4 TB NVMe SSD B: 12,8 TB NVMe SSD	Optionen: A: 6,4 TB NVMe SSD B: 12,8 TB NVMe SSD
DB Edition	SE, SE1, SE2 und EE	SE, SE1, SE2 und EE
Virtualisierung (OVM)	Nein	Nein
RAC	Nein	Nein

Tabelle 1

Komponenten	ODA X6-2L	ODA X6-2-HA
Größe	Zwei Höheneinheiten	Zwei Server plus Anzahl der Storage-Shelfs
Prozessor	Zwei CPUs á 10 Kerne (Intel Xeon E5-2630 v4)	Zwei CPUs á 10 Kerne pro Server (Intel Xeon E5-2630 v4)
Speicher	Min. 256 GB Max. 768 GB	Min. 256 GB Max. 768 GB
Netzwerk	4 x 10GBase-T (2x2 bond) 2 x 10GbE SFP+ (fiber)	4 x 10Gb InfiniBand Interconnect, SFP+ (fiber) optional
Boot-Festplatte:	2 x 480 GB SSD (gespiegelt)	2 x 480 GB SSD (gespiegelt)
Storage	19,2 bis 28,8TB NVMe SSD	12 bis 48TB SSD (Storage Shelf)
DB Edition	SE, SE1, SE2 und EE	EE
Storage Management	Nein	Ja
RAC	Nein	Ja

Tabelle 2

Welche Hardware ist die richtige?

Vereinfacht könnte die Aussage getroffen werden, dass bei der Auswahl neuer Server/CPUs darauf geachtet werden sollte, dass diese wenige, jedoch sehr leistungsstarke Kerne besitzen. Dies ist vor allem im Zusammenhang mit der SE2 Edition sinnvoll – Stichwort „zwei Sockel und sechzehn Thread"-Begrenzung.

Die Lösung kommt in diesem Fall – seit Juni 2016 – aus dem Hause Oracle. Es sind die neuen Oracle Database Appliance Server X6-2 (ODA X6-2). Es gibt sie zum aktuellen Zeitpunkt in vier verschiedenen Ausführungen:

- ODA X6-2S
- ODA X6-2M
- ODA X6-2L
- ODA X6-2-HA

Technische Spezifikationen der Server

Tabelle 1 zeigt eine Übersicht der Server ODA X6-2S und ODA X6-2M, *Tabelle 2* eine Übersicht der Server ODA X6-2L und

ODA X6-2-HA. Hinweis: Aufgrund der gesammelten Erfahrung des Autors mit den Maschinen ODA X6-2S und X6-2M beschränkt sich der Artikel auf diese beiden Modelle.

Installation und Konfiguration

Der Einbau in die Serverschränke ist einfach. Zum Standard-Lieferumfang gehören neben dem Server zwei Netzkabel, die Serverschienen sowie ein Gelenkarm für eine ordentliche und saubere Kabelführung. Netzwerkkabel sowie „SFP+"-Adapterstecker sind im normalen Lieferumfang nicht enthalten. Bevor die Appliance jedoch in Betrieb genommen werden kann, sollten die folgenden Informationen zur Verfügung stehen:

- Domain-Name
- DNS-Server-Adresse
- NTP-Server-Adresse
- IP-Adressen sowie die dazugehörigen Subnet-Masken für
  - ILOM-Schnittstelle
  - Netzwerkanbindung
  - Gateway

Darüber hinaus sollten die im MOS-Artikel 2144642.1 beschrieben Patches herunterladen werden.

```
[root@oak ~]# configure-firstnet
Select the Interface to configure the network on (btbond1 btbond2 sfpbond1)
ond1]:btbond1
Configure DHCP on btbond1 (yes/no) [no]:
INFO: You have chosen Static configuration
Enter the IP address to configure :
Enter the Netmask address to configure :
Enter the Gateway address to configure[           ] :
```

Abbildung 1: Ausschnitt der ersten Netzwerk-Schnittstelle

```
Node  Cores  Modified                          Job Status
----- ------ --------------------------------  ---------------
0  4         October 28, 2016 12:00:00 AM      CEST Configured
```

Listing 1

Nach dem Einbau des Servers muss dieser mit beiden Netzkabeln sowie einer USB-Tastatur und einem VGA-Monitor angeschlossen werden. Es folgt ein Selbstcheck, in dessen Anschluss der Server eingeschaltet werden kann. Nachdem der Server erfolgreich gestartet wurde, können die Netzwerkkabel (SFP+/BTBOND) eingesteckt und das erste Netzwerk mit „configure-firstnet" konfiguriert werden.

Wichtig dabei ist, dass dieser Befehl nur einmal ausgeführt wird. Er dient dazu, dass der Server anschließend im Netzwerk verfügbar ist und somit über SSH weiter konfiguriert und installiert werden kann. *Abbildung 1* zeigt einen Ausschnitt der Konfiguration des BTBOND1. Aus Sicherheitsgründen wurden alle eingegebenen Daten ausgeschnitten.

Nachdem die Netzwerk-Konfiguration abgeschlossen ist, müssen die bereits heruntergeladenen Patches installiert werden. Dabei kann nach den mitgelieferten Installationsanleitungen vorgegangen werden. Bevor jedoch nun die Appliance erstellt werden kann, ist es notwendig, dass im Falle eines Einsatzes einer Enterprise Edition die CPU-Kerne auf die vorhandenen Lizenzen eingegrenzt werden. Dieses sogenann-

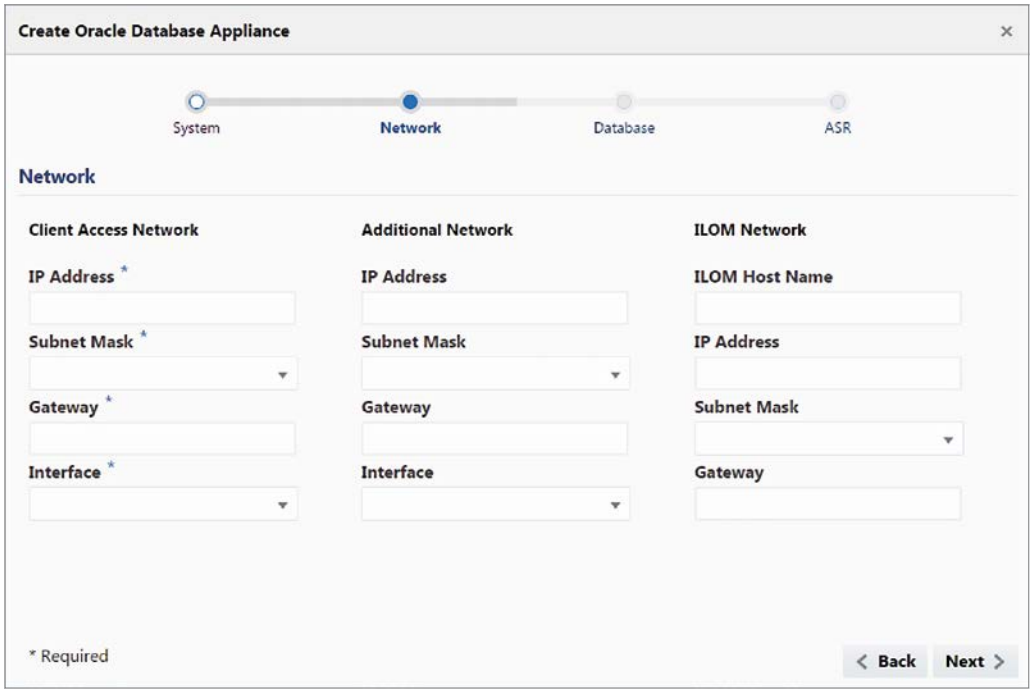


Abbildung 2: Appliance-Konfiguration

te „Capacity on Demand“-Feature ist bei keinem anderen Hersteller, ohne den Einsatz zusätzlicher Software (Oracle VM), in dieser Art und Weise zu finden und somit ein großer Vorteil der Appliance Server.

Mithilfe des Befehls „odacli describe-cpucore“ können die aktivierten Kerne angezeigt werden. Die Ausgabe kann wie in *Listing 1* aussehen.

Die ODA ist nun für die Erstellung der Appliance und der dazugehörigen Datenbank bereit. Die Erstellung der Appliance wird dabei über ein Web-Interface durchgeführt, das unter der URL *„https://<IP_auser_Konfiguration>:7093/mgmt/index.html“* zur Verfügung steht. Nach der Anmeldung am Web-Interface durch den User „ODA-Admin“ kann die Appliance erstellt werden.

Auf der ersten Maske *(siehe Abbildung 2)* sind die Informationen über die Appliance angegeben. Leider kann in diesem Artikel nicht auf alle Eingabefelder eingegangen werden, sondern nur auf jene, die eine Mehrfach-Bedeutung vorweisen. Das Feld „Domain Name“ sorgt unter anderem auch dafür, dass der PFILE/SPFILE-Parameter „DB_DOMAIN“ gesetzt wird.

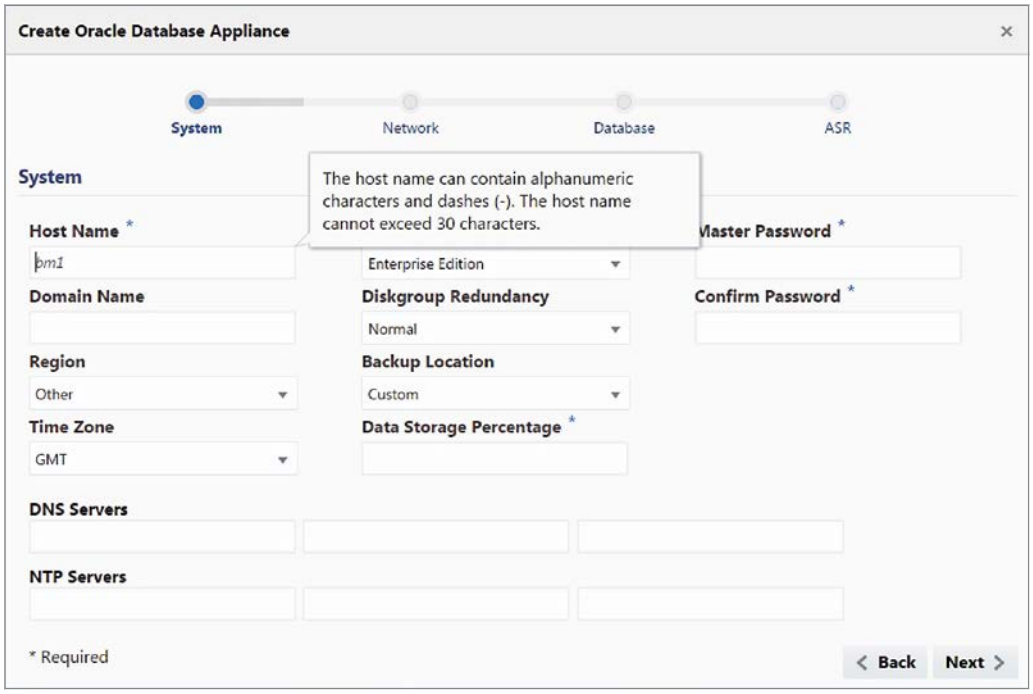


Abbildung 3: Netzwerk-Konfiguration

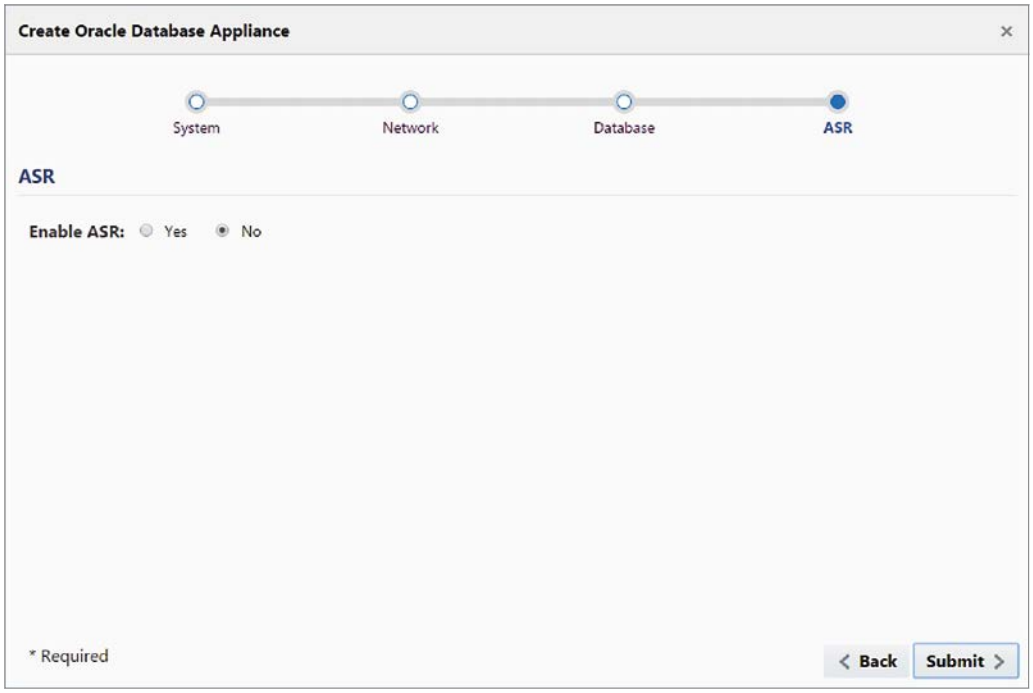


Abbildung 5: ASR-Konfiguration

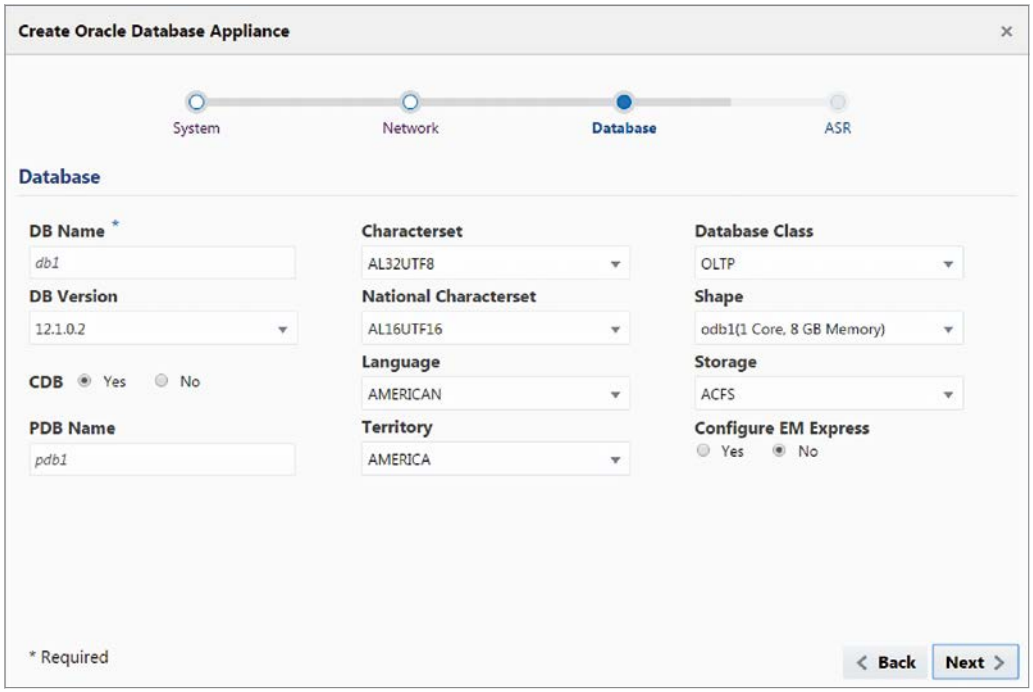


Abbildung 4: Datenbank-Konfiguration

Alle weiteren Felder sollten selbsterklärend sein *(siehe Abbildung 2)*.

Auf der zweiten Maske *(siehe Abbildung 3)* wird die Konfiguration des Netzwerks vorgenommen. Dabei wird zwischen einem Client und einem alternativen Netzwerk unterschieden. Zusätzlich kann, sofern noch nicht geschehen, die ILOM-Schnittstelle („Integrated Light Off Manager") konfiguriert werden. Das Client-Netzwerk ist eine verpflichtende Konfiguration. Sollte diese mit der Konfiguration aus dem „configure-firstnet"-Befehl übereinstimmen, können die gleichen Daten abermals eingetragen werden.

Auf der dritten Maske *(siehe Abbildung 4)* werden die Einstellungen der Datenbank vorgenommen. Die Wahl des „DB Namen" sollte in Bezug auf eine bevorstehende Standby-Konfiguration wohl überlegt sein. Das Feld „Shape" sorgt für eine Eingrenzung der Kerne für die zu erstellende Datenbank. Die RAM-Angaben können jederzeit im Nachhinein angepasst werden.

Sobald alle Informationen eingetragen und ausgewählt wurden, erscheint die letzte Maske *(siehe Abbildung 5)*. Auf ihr kann die Einstellung des Automatic Service Request (ASR) vorgenommen werden. Dies ist ein Feature zur automatischen Erstellung eines Oracle Service Request im Falle eines Fehlers. Nach der letzten Einstellung werden sowohl die Appliance als auch die dazugehörige Datenbank erstellt. Der anschließende Erstellungsstatus lässt sich dabei in Form von einzelnen Tasks beobachten.

Monitoring und Hardware-Ersatz

Mit der ODA X6-2 wird mithilfe der ILOM-Schnittstelle auch ein rudimentäres Hardware Monitoring mitgeliefert, das einen Einblick in den verschiedenen Status der Hardware des Systems wiedergibt. Dabei ist jedoch nur die Appliance selbst in Beobachtung und nicht die darauf laufenden Datenbanken oder etwa die Zustände wie die Auslastung des Plattenplatzes. Des Weiteren sind einige Meldungen der Zustände wie im Falle eines Plattendefekts einer der Boot-SSDs gewöhnungsbedürftig. So wies eine Maschine eine defekte Boot-SSD weiterhin mit den Status „OK" aus. Bei näherer Betrachtung konnte festgestellt werden, dass die Serial-Number fehlte und die Festplatte nur noch als reguläre HDD angezeigt wurde. Eine externe Überwachung des Systems ist somit notwendig.

Das Monitoring-Modul der Firma Herrmann & Lenz Solutions wurde um zusätzliche Services erweitert, die speziell auf die Bedürfnisse der ODA X6-2 abgestimmt sind. So ist das Monitoring-System beispielsweise in der Lage, den Lebenszyklus des NVMe-Speichers zu überwachen. Zudem wird im Falle eines technischen Defekts, ob es nun der Lüfter, die SSDs oder gar der RAID-Controller ist, dieser sofort erkannt und eine entsprechende Meldung angezeigt.

Im Falle eines Hardware-Ausfalls muss bei Oracle ein SR eröffnet werden. Nach der Feststellung der defekten Hardware durch den Support wird ein entsprechender Ersatz zügig versendet und kann entweder durch einen Oracle-Service-Techniker vor Ort oder selbstständig ausgetauscht werden.

Vor- und Nachteile der ODA X6-2S/M Server

Die Vorteile sind:

- Aufgrund ihrer Ausstattung sind diese Server ideal für den Einsatz von Single-Instance-Datenbanken geeignet
- Dank der maximal zwei vorhandenen Sockel ist es möglich, eine SE2 Edition einzusetzen
- Im Falle eines Einsatzes der Enterprise Edition besteht die Möglichkeit, nur die Anzahl der Kerne zu aktivieren, für die auch die entsprechenden Lizenzen vorhanden sind („Capacity on Demand")
- Dank des großen Storage von bis zu 12.8 TB (Raw Capacity) NVMe-SSDs steht genügend Platz zur Verfügung, um die Daten in der Datenbank abzuspeichern.

- Das Storage kann wahlweise zweifach (Normal Redundancy – 3,2 oder 6,4TB verfügbar) oder dreifach gespiegelt (High Redundancy – 6,4 oder 4,2TB verfügbar) werden.
- Durch den Flash-Speicher wird eine extrem gute I/O-Performance erreicht
- Geringe Anschaffungskosten

Nachteile sind

- Beide ODA-Server besitzen keine Hochverfügbarkeit
- Es wird keine Virtualisierung unterstützt

Fazit

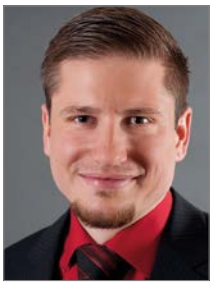
Mit den neuen ODA-X6-2S/M-Maschinen stehen sehr leistungsstarke Server zur Verfügung, die selbst in der kleinen Ausbaustufe dank der verbauten CPUs und bedingt durch den NVMe-Speicher ihresgleichen suchen. Besitzer einer SE2 Edition können diese ohne Probleme auf den neuen ODA X6-2 Server einsetzen. Firmen, die in Besitz einer Enterprise Edition sind, können aufgrund der „Capacity on Demand"-Funktionalität Kerne abschalten und bei Bedarf − sofern Lizenzen vorhanden sind − hinzufügen.

Obwohl diese beiden Server keine Hochverfügbarkeit aufweisen, ist es dennoch möglich, bei einem Einsatz von zwei Servern eine Primary/Standby-Umgebung mithilfe von Data Guard (Enterprise Edition) oder Dbvisit (SE, SE1, SE2) aufzubauen.

Der Einbau sowie die Inbetriebnahme des Servers sind einfach und schnell. So ist es möglich, innerhalb eines Arbeitstages, diesen sowie die darauf laufenden Datenbanken so zu konfigurieren, dass eine Datenübernahme von einem Altsystem vorgenommen werden kann. Ein weiterer Vorteil sind die sehr attraktiven Preise, die selbst für kleine Unternehmen erschwinglich sind. Die Server konnten in dem bisherigen Produktiv-Einsatz durchweg überzeugen, sodass sie auf jeden Fall bei einer Neuanschaffung in Betracht gezogen werden sollten.

Links

[1] ODA X6-2S/M White Paper: *http://www.oracle.com/technetwork/database/database-appliance/downloads/odax6-2sm-wp-3049043.pdf*
[2] ODA X6-2L Data Sheet: *http://www.oracle.com/technetwork/database/database-appliance/learnmore/oda-x6-2l-ds-3242353.pdf*
[3] ODA X6-2-HA Data Sheet: *http://www.oracle.com/technetwork/database/database-appliance/learnmore/oda-x6-2ha-ds-3242361.pdf*
[4] Setup Booklet: *http://docs.oracle.com/cd/E75550_01/doc.121/e76903/*
[5] Deployment and Users Guide: *http://docs.oracle.com/cd/E75550_01/doc.121/e76900/toc.htm*
[6] HL-Monitoring: *http://www.hl-solutions.de/produkte/monitoring-module*



Johannes Kraus
johannes.kraus@hl-services.de

Optimale Vorbereitung auf Oracle-Zertifizierungen

Rainer Schaub, Acceleris AG

IT-Beratungsunternehmen legen bei der Einstellung von Informatikern schon seit längerer Zeit großen Wert auf die Zertifizierung der Kandidaten. Neuerdings findet man auch bei Stellenausschreibungen von Firmen, die interne IT-Experten einstellen möchten, immer häufiger die Anforderung, dass diese Oracle-zertifiziert (OCA, OCP oder OCE) sind. Eine möglichst erst kürzlich erfolgte Oracle-Zertifizierung stellt einen Mehrwert sowohl für den Mitarbeiter/Bewerber als auch für den Arbeitgeber dar.

Um eine Prüfung bestehen zu können, benötigt man sowohl Kenntnisse über das Prüfungsthema (etwa Oracle Database Administration) als auch über die Form der Prüfung (wie Multiple Choice, Sprache). Der Artikel behandelt die Aspekte der Form einer Oracle-Zertifizierungsprüfung; die Hinweise sind lediglich für Multiple-Choice-Aufgaben anwendbar und damit nicht für eine Oracle-Certified-Master-Prüfung (OCM). Informationen über eine generelle Vorgehensweise zur Vorbereitung auf den Inhalt einer Oracle-Prüfung sind beispielsweise im Video von Gwen Lazenby zu „Exam Preparation Tips" *(siehe „https://www.youtube.com/watch?v=RcxXjN-QsbY")* zu sehen.

Der Artikel gibt Hinweise zu Fragetypen und sogenannten „magischen Worten". Mit „Fragetyp" ist eine besondere Art der Fragestellung gemeint, die sich deutlich von anderen Typen unterscheidet und zudem einer eigenen Antwortstrategie bedarf. „Magische Worte"' sind Worte, die spezielle Beachtung verdienen, da sie Hinweise darauf geben, ob diese Antwort eher mehr oder eher weniger infrage kommt.

Markieren und mehrere Durchläufe

Im Gegensatz zu einer GMAT-Prüfung *(siehe „http://www.mba.com/us")* kann bei einer Oracle-Zertifizierungsprüfung eine Frage markiert und später (nochmals) beantwortet werden. Dies ist sehr hilfreich, da es vorkommen kann, dass die Lösung oder eine Teillösung einer Frage aus dem Inhalt einer anderen Frage abgeleitet werden kann. Zudem ist es unter dem Gesichtspunkt der Prüfungsstrategie logisch, schwierige Fragen erst zu bearbeiten, wenn bereits ein gewisser Fundus von Antworten vorliegt, da so Zeit und Energie gespart werden können. Last but not least ist es psychologisch geschickter, in einem ersten Durchgang die einfacheren Fragen zu bearbeiten. Die Ideen stammen zum Großteil aus John Watsons Buch „OCA Oracle Database 12*c* Installation and Administration Exam Guide", 2014, McGraw-Hill Education, und wurden vom Autor erfolgreich angewandt.

Fragetypen

Die Kenntnis unterschiedlicher Fragetypen und der dazugehörigen optimalen Antwortstrategie hilft, kostbare Prüfungszeit zu sparen, die Ressourcen (Zeit, Energie, Konzentrationsfähigkeit) besser einzuteilen und so die Chancen zu erhöhen, eine höhere Antwortquote zu erreichen. Dem Autor sind derzeit vier unterschiedliche Fragetypen bekannt. Diese werden nun der Reihe nach erläutert, danach wird ein Strategievorschlag zur adäquaten Behandlung vorgeschlagen.

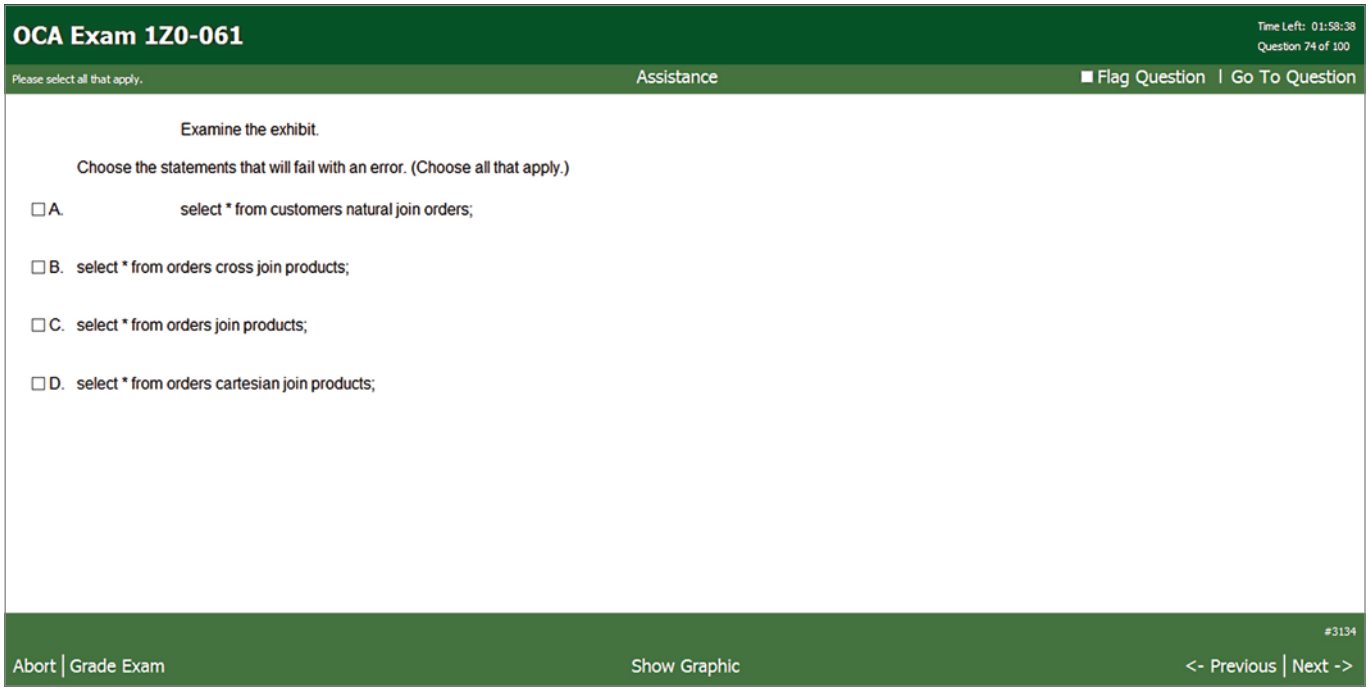


Abbildung 1: Beispiel für eine „Choose all that apply"-Frage

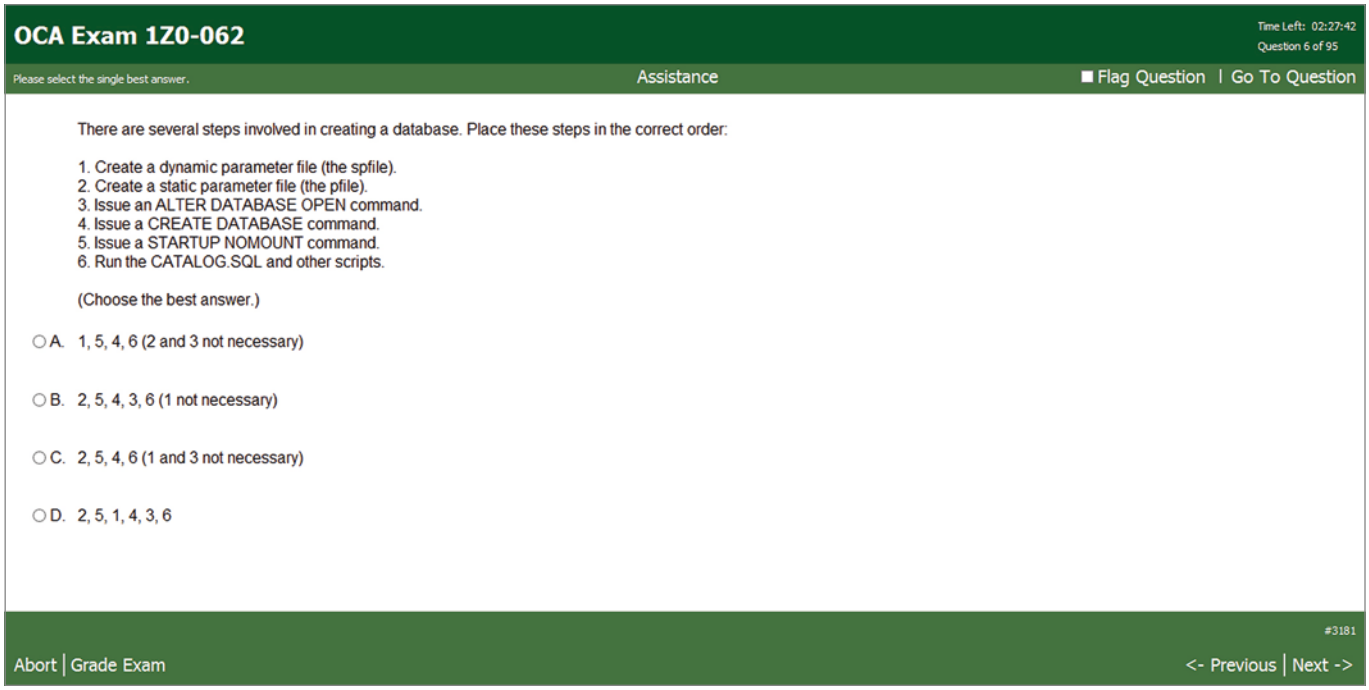


Abbildung 2: Beispiel für eine „Complicato"-Frage

„Choose all that apply"

Dies ist der mit Abstand anspruchsvollste und schwierigste Fragetyp, da er keine Information über die Anzahl der richtigen Antworten gibt und meist viel wertvolle Prüfungszeit und Energie beansprucht. Deshalb ist die Empfehlung klar: Frage umgehend markieren – dies gilt auch für inhaltliche Themen, in denen sich der Kandidat wohlfühlt – und zur nächsten Frage übergehen. *Abbildung 1* zeigt ein Beispiel des Fragetyps „Choose all that apply".

„Complicato"

Dieser Fragetyp zeichnet sich durch eine komplizierte Struktur aus, bietet jedoch den Vorteil, dass nur eine Antwort richtig ist. Zuerst werden der Sachverhalt sowie eine Liste einzelner Schritte beschrieben, die zur Lösung des Sachverhaltes notwendig sein könnten. Anschließend muss der Kandidat die richtige Antwort aus einer Reihe von Möglichkeiten auswählen. Der Versuch, selbst die richtige Reihenfolge der einzelnen Schritte zu finden, benötigt kostbare Zeit und ist meistens zum Scheitern verurteilt. Eine gute Strategie besteht darin,

die Beschreibung des Sachverhalts genau zu lesen und die Liste der einzelnen Schritte darauf zu prüfen, ob sie zur Lösung nicht notwendige Schritte enthält oder einzelne Schritte mehrfach zur Lösung notwendig sind. Mit diesem Wissen kann in der Regel schon die eine oder andere Antwort ausgeschlossen werden; gelegentlich findet man die richtige Antwort, weil ein Schritt mehrfach vorkommen muss und dies nur bei einer Antwort der Fall ist.

Weitere Möglichkeiten zur Lösungsfindung bestehen darin, den jeweils ersten oder letzten Schritt über alle Antwortmöglichkeiten zu vergleichen und so die Anzahl der möglichen Lösungen weiter einzugrenzen. Manchmal werden auch unnötige Schritte („not necessary") aufgeführt. Dies stellt auch eine weitere gute Möglichkeit dar, um die Lösung zu finden. Sind zur Lösung der Aufgabe notwendige Schritte als „not necessary" ausgewiesen, so ist klar, dass diese Antwort falsch ist. Auch hier gilt es, diese Frage sofort zu markieren, zur nächsten überzugehen und die Frage erst in einem späteren Durchlauf zu bearbeiten, da in der Regel die Beantwortung dieses Typs von Fragen zeitaufwändig ist *(siehe Abbildung 2)*.

„Choose 1, 2 or 3"

Auch dieser Fragetyp weist wie „Complicato" die richtige Anzahl der Antworten aus und stellt den einfachsten Typ dar. Hier gilt es, mithilfe des „Process of Elimination"-Verfahrens (POE) falsche Antworten zu eliminieren und dann, falls die Antwort offensichtlich oder bekannt ist, diese auszuwählen *(siehe Abbildung 3)*.

Die Besonderheit dieses Fragetyps liegt darin, dass nicht alle Informationen sofort auf dem Bildschirm ersichtlich sind, sondern erst durch Drücken der Taste „Show" angezeigt werden. Bei den Zertifizierungen „Oracle Database SQL Expert", „Oracle Database 11*g*/12*c*: SQL Fundamentals" oder auch bei PL/SQL-Zertifizierungsprüfungen kommt diese Art von Fragen recht häufig vor.

Wichtig ist zu wissen, dass bei etwa der Hälfte der Fälle die Frage auch ohne Kenntnis der zusätzlichen Informationen gelöst werden kann. Die adäquate Strategie wäre demnach, die Frage ohne Aufrufen der Zusatz-Informationen zu lösen. Nur wenn dies nicht möglich ist, sollten die weiteren Informationen mittels „Show" konsultiert werden. Dies hat zudem den Vorteil, dass man genau weiß, welche Informationen für die Antwort fehlen, und somit gezielt im zusätzlichen Datenbereich gesucht werden kann.

Eine weitere Problematik der Zusatz-Informationen besteht darin, dass viele nicht notwendige Daten angeboten werden. Beim Beispiel in *Abbildung 4* kann die eigentliche Frage auch ohne die zusätzlichen Informationen gelöst werden, da es ausschließlich um die Syntax geht und die Kenntnis der Struktur der Tabellen für die Lösung überflüssig ist.

Gesamtstrategie zu den Fragetypen

Es empfiehlt sich, für die Fragetypen „Choose all that apply" (CATA) und „Complicato" (COMP) im ersten Durchgang keine Zeit aufzuwenden und beide nur zu markieren. Im zweiten Durchgang kann man dann „Complicato"-Fragen lösen und sich erst im dritten Durchgang den „Choose all that apply"-Aufgaben zuwenden.

„Show"-Fragen können bisweilen bei allen Fragetypen vorkommen, obwohl sie meistens bei „Choose 1, 2 or 3" anzutreffen sind. Die Antwortstrategie zu deren Lösung ist somit vom eigentlichen Fragetyp abhängig. Fragen des Typs „Choose 1, 2 or 3" löst man im ersten Durchgang, wenn sie einfach sind, ansonsten im zweiten Durchgang. Auch wenn jeder Kandidat letztlich seine eigene Strategie finden muss, zeigt *Abbildung 5* eine für den Autor optimale Gesamtstrategie.

„Magische Worte"

Es gibt Ausdrücke, die spezielle Beachtung verdienen, da sie Hinweise darauf geben,

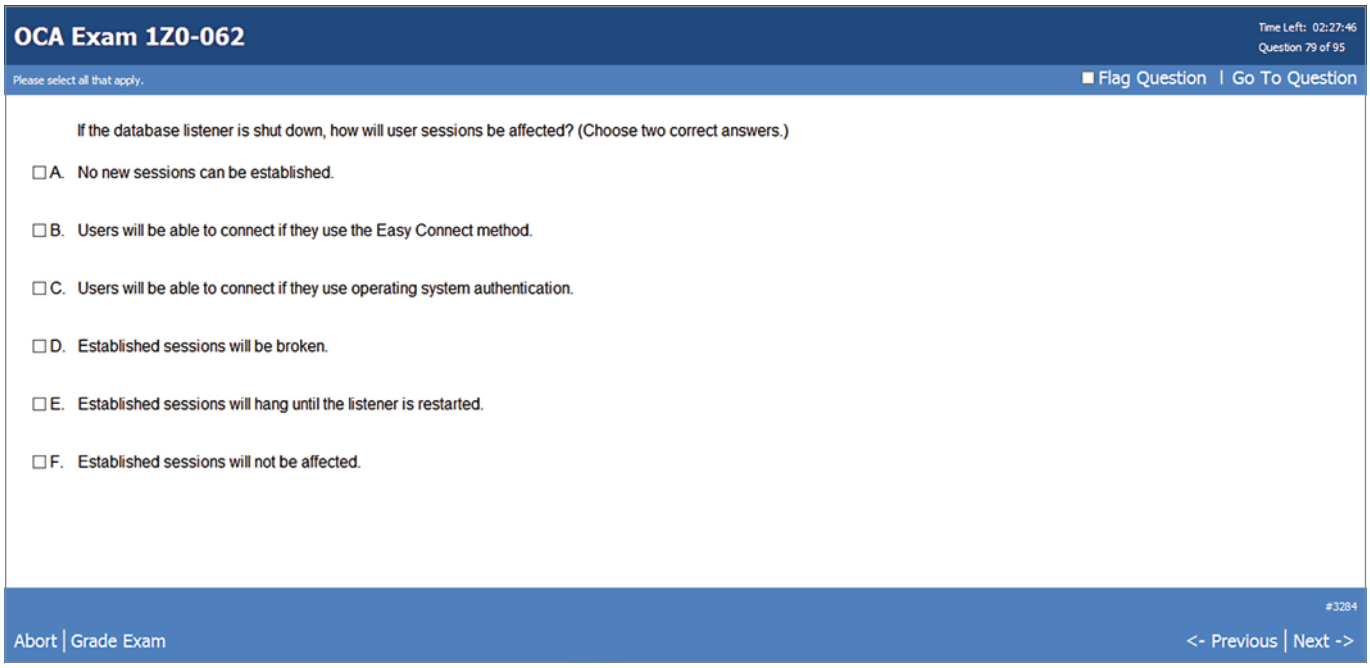


Abbildung 3: Beispiel für eine „Choose 1, 2 or 3"-Frage

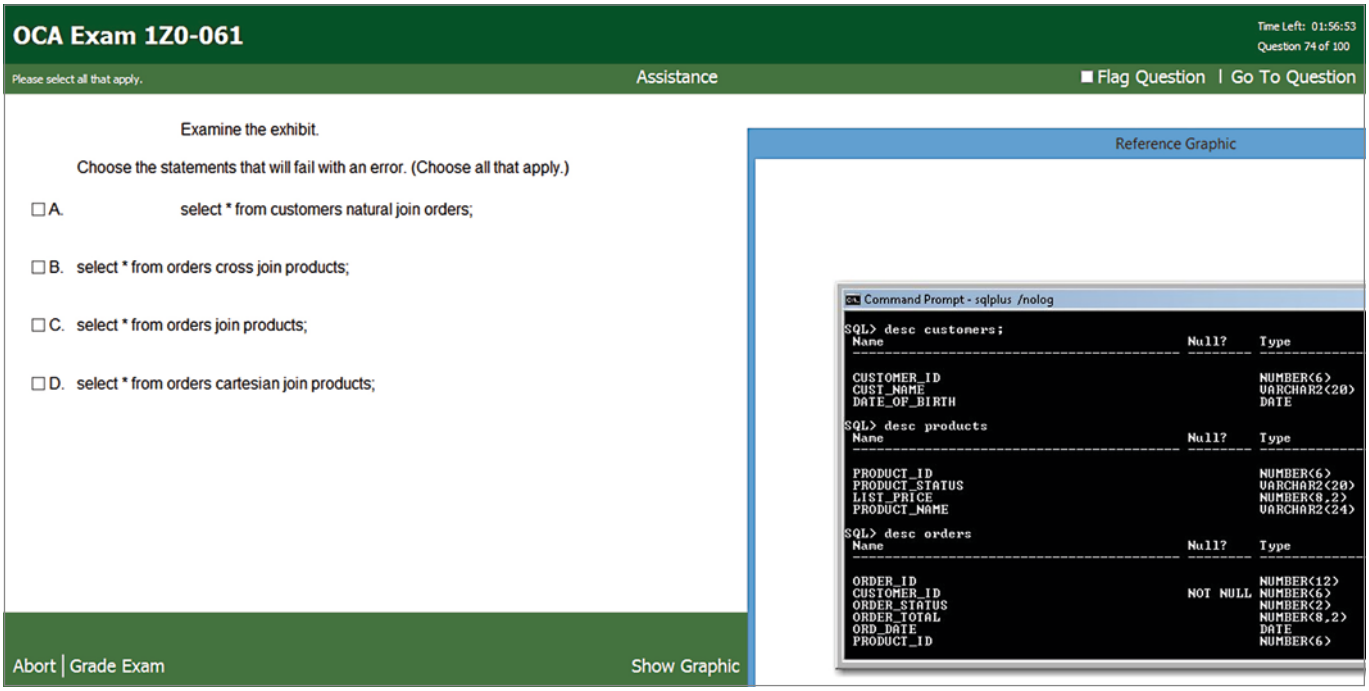


Abbildung 4: Beispiel für eine „Show"-Frage

ob eine Antwort eher mehr oder eher weniger infrage kommt. Nachfolgend eine Liste von magischen Wörtern, die in Oracle-Zertifizierungsprüfungen vorkommen:

- Only
- Must
- Always
- Can
- Can only
- Exactly
- Never
- Automatically
- Might

Worte wie „must", „always", „exactly" und „never" sind Wörter, die einengend sind, da sie auf einer Skala am einen (wie „never") oder am anderen (wie „always") Ende des möglichen Lösungs-Spektrums vorkommen und somit quasi Extremsituationen darstellen. Wörter wie „might" oder „can" kommen im Gegensatz dazu auf einer Skala der möglichen Lösungen im mittleren Bereich vor und präsentieren somit die große Masse.

Das Wort „automatically" kann man auch zu den einengenden Begriffen zählen, da auch hier kein Handlungsspielraum vorhanden ist. Andere einengende Wörter wie „always" oder „never" geben einen Hinweis darauf, dass die Antwort, die dieses magische Wort enthält, tendenziell nicht richtig ist. Dazu ein Anwendungsbeispiel: Man hat bei einer „Choose 2"-Frage mit fünf mögliche Antworten zwei Antworten als falsch und eine weitere als richtig identifiziert. Bei den übrigen zwei Antwortmöglichkeiten hat man jedoch keine Idee, welche die zweite richtige Antwort ist. Wenn nun in einer der beiden Antworten ein einengendes Wort wie „never" oder „always" vorkommt, so sollte man die andere Antwort als zweite richtige auswählen. Kommt in einer der beiden Antworten ein „might" vor, während die andere kein magisches Wort enthält, ist der „might"-Antwort der Vorzug zu geben.

	1. Durchgang	2. Durchgang	3. Durchgang
CATA	nein	nein	ja
COMP	nein	vielleicht	ja
1/2/3	vielleicht	ja	---
SHOW	vielleicht	ja	---

Einfache Fragen im ersten Durchgang bearbeiten. Schwerere Fragen eher im zweiten oder dritten Durchgang lösen.

Abbildung 5: Strategievorschlag zu Fragetypen

Gegenseitiger Ausschluss

Manchmal kommt es vor, dass zwei Antworten sich völlig widersprechen. Dies stellt einen gegenseitigen Ausschluss dar. Ist die eine Antwort richtig, so kann die andere Antwort nicht richtig sein. Beim gegenseitigen Ausschluss ist zudem wichtig zu wissen, dass meistens eine Antwort davon eine richtige Antwort ist. Auch dazu ein Anwendungsbeispiel: Man hat bei einer „choose 2"-Frage mit fünf möglichen Antworten eine als falsch und eine weitere als richtig identifiziert. Bei den letzten drei Antwortmöglichkeiten hat man jedoch keine Idee, welche die zweite richtige Antwort ist. Falls in keiner der drei zur Wahl stehenden Antworten ein magisches Wort vorkommt, jedoch zwei Antworten einen gegenseitigen Ausschluss darstellen, so sollte man eine der beiden sich ausschließenden Antworten wählen.

Doppelte Verneinung

Die doppelte Verneinung ist immer und unabhängig von der vorliegenden Prüfungssituation schwierig und verwirrend. Hier ist es ratsam, die Frage als Bejahung umzuformulieren und anschließend die Lösung der Frage in Angriff zu nehmen. Die doppelte Verneinung ist zwar selten in Oracle-Zertifizierungsprüfungen, aber sie kommt vor.

Fazit

Voraussetzung für das Bestehen einer Prüfung ist immer die Sachkenntnis! Bei einer Oracle-Zertifizierungsprüfung gibt es keine Noten, sondern lediglich „bestanden" oder „nicht bestanden". Es gilt also, die erforderliche Quote der richtigen Antworten zu erzielen.

Die gute Kenntnis der Form einer Prüfung allein reicht zwar nicht aus, um diese zu bestehen, sie kann aber den Ausschlag zwischen „nicht bestanden" und „bestanden" geben. Deshalb ist die Kenntnis möglichst vieler Rahmen-Faktoren in vielen Fällen nützlich und hilfreich. Wenn diese Ausführungen dazu beitragen, so haben sie ihren Zweck erfüllt.

Anmerkung: Die Abbildungen 1 bis 4 sind der CD des Buches „OCA/OCP Oracle Database 12*c* All-in-One Exam Guide" von John Watson entnommen. John Watson hat die Zustimmung hierzu schriftlich erteilt und der Verlag ist darüber informiert.

Rainer Schaub
rainer.schaub@acceleris.ch

Schnelles und kostengünstiges Datenbank-Cloning mit dem ASM-Cluster-File-System

Sebastian Solbach, ORACLE Deutschland B.V. & Co.KG

Für ein Duplikat einer produktiven Datenbank gibt es viele Anwendungsfälle, von der einfacheren Sicherung über das Testen von neuen Funktionalitäten bis hin zur Bereitstellung von Entwicklungsumgebungen. Zwar muss insbesondere Letzteres nicht immer auf aktuellen Daten geschehen, es bietet aber dennoch enorme Vorteile, mit einer Kopie der Produktion zu arbeiten.

Leider stehen den Vorteilen eines Duplikats vor allem der doppelte Plattenplatz-Bedarf pro Kopie und auch die Zeit zur Erstellung einer solchen Kopie entgegen. Eine technische Lösung, um eine komplette Kopie schnell und platzsparend bereitzustellen, wurde deshalb auch schnell gefunden: Datei-basierende Snapshots auf Basis von Copy-on-Write (CoW) beziehungsweise Redirect-on-Write (RoW). Ein Snapshot sieht dabei aus wie eine vollständige Kopie, enthält aber keine Daten, sondern nur einen Verweis auf die Originalblöcke. Erst geänderte Daten werden bei einem Update wirklich geschrieben – entweder an denselben Ort bei CoW oder an einen anderen Ort bei RoW.

Datei-basierende NFS-Storage-Systeme stellen diese Funktionalität schon länger zur Verfügung. Das ist jedoch nicht selten mit zusätzlichen Kosten verbunden. Da es sich bei den Verfahren aber eigentlich um Storage-Technologien handelt, finden diese auch Einzug in moderne Dateisysteme. Das kostenlose ASM-Cluster-File-System von Oracle bietet diese Funktionalität für alle Dateien. Lediglich die Verwendung der CoW-Snapshot-Funktion für Oracle-Datenbanken ist an die Enterprise Edition geknüpft.

Das ASM-Cluster-File-System

Oracle liefert das ASM-Cluster-File-System (ACFS) als ein modernes, generisches, PO-

SIX-kompatibles Dateisystem mit der Grid-Infrastruktur aus. Es basiert auf dem Automatic Storage Management und steht auf allen Plattformen zur Verfügung, für die es die aktuelle Grid-Infrastruktur gibt. Lediglich für HPUX ist es nicht verfügbar. Da ACFS auf ASM basiert, erbt es auch alle dessen Vorzüge: vom automatischen Striping bis hin zum Mirroring.

Seit einiger Zeit steht ACFS mit all seinen Funktionalitäten kostenfrei zur Verfügung. Neben der Snapshot-Technologie, die Bestandteil dieses Artikels ist, bietet ACFS allerdings noch ein paar andere interessante Lösungen:

- *Verschlüsselung*
  Daten im ACFS können automatisch verschlüsselt werden

- *Zugriffsberechtigung*
  Dateien können zusätzliche Rechte erfordern, die über die normalen Betriebssystemrechte hinausgehen

- *Replikation*
  Repliziert ein ACFS-Filesystem auf ein anderes zum Zwecke der Ausfallsicherheit

Im Gegensatz zum Snapshot stehen diese Funktionen aber nur Dateien zur Verfügung, die nicht direkt mit einer Oracle-Datenbank in Verbindung stehen. Die Datenbank bietet für Verschlüsselung, Sicherheit und Replikation eigene Mechanismen. So übernimmt Transparent Data Encryption die Verschlüsselung, Database Vault die strengeren Sicherheitsmechanismen und Replikation wird selbstverständlich durch Data Guard gelöst. Snapshots von Oracle-Datenbank-Dateien sind zwar generell erlaubt, erfordern jedoch eine Enterprise Edition. Näheres zu den ACFS-Funktionen und den Restriktionen im Zusammenhang mit der Datenbank stehen im Lizenzierungs-Handbuch *[1]*.

Ebenfalls ist es erst mit der Grid-Infrastruktur 12.1.0.2 erlaubt, ACFS-Filesysteme für Oracle-Datenbank-Dateien zu verwenden, da hierfür einige Performance-relevante Verbesserungen für Datenbanken eingebaut wurden. Es gibt allerdings noch ein paar andere Voraussetzungen, bevor ACFS für Datenbank-Dateien sinnvoll einsetzbar ist:

- Die ASM-Diskgruppe, die das ACFS-Filesystem beherbergt, muss alle Kompatibilitäts-Attribute mindestens auf 12.1.0.0 stehen haben („COMPATIBLE.ASM“, „COMPATIBLE.RDBMS“ und „COMPATIBLE.ADVM“)
- Das dem ACFS zugrunde liegende Volume sollte acht Stripe Columns und eine Stripe-Width von mindestens 1 MB oder einem Vielfachen davon aufweisen, am besten der Allocation-Unit-Größe der Diskgruppe entsprechen
- Grid-Infrastruktur für ein Cluster muss vorliegen – „Oracle Restart only“ ist nicht unterstützt, da sich diese ACFS nicht im Cluster registrieren lassen
- Oracle-Datenbank Version 11.2. oder 12.1.0.2
- „FILESYSTEMIO_OPTIONS“ muss auf „SETALL“ stehen
- Die Datenbank-Blockgröße sollte 4 K oder ein Vielfaches davon sein

Insbesondere der ADVM-Kompatibilitätslevel ist ausschlaggebend, da erst hiermit Datenbank-Dateien erlaubt sind und damit die Anzahl der Snapshots von 63 auf 1023 erhöht wurde. Die eingestellten Kompatibilitäts-Level und die Parameter des dem ACFS zugrunde liegenden Volume lassen sich am einfachsten über das ASM Command Line Interface (ASMCMD) abfragen *(siehe Listing 1)*.

Nähere Details zur Verwendung von ACFS für Datenbank-Dateien finden sich im Automatic Storage Management Administrator's Guide der Dokumentation im Abschnitt „About Oracle ACFS and Database Data Files“ *[2]*.

ACFS-Snapshots

ACFS-Snapshots stehen wie erwähnt für beliebige Dateien zur Verfügung. Um von einem ACFS-File-System einen Snapshot zu erstellen, dient der Befehl „acfsutil“, der als Benutzer mit „root“-Rechten ausgeführt werden muss *(siehe Listing 2)*. Informationen über bestehende Snapshots und deren Verwaltung erfolgen ebenfalls mit diesem Befehl *(siehe Listing 3)*.

Nun könnte man selbstverständlich die notwendigen Schritte für das Klonen von Datenbanken selbst übernehmen. Diese wären ganz grob für eine 11*g*R2-Datenbank:

- Produktive Datenbank in den Backup-Modus versetzen
- Snapshot des ACFS-File-Systems (mit den Datendateien; Redo Logs und Temp Tablespace werden nicht benötigt)

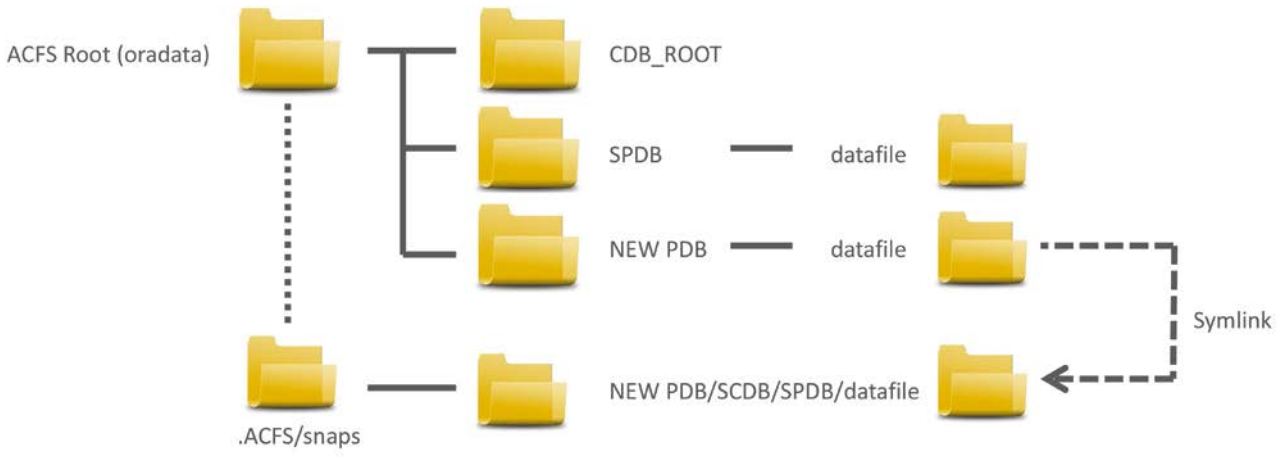


Abbildung 1: Snapshot Clone Directory Structure

- Auf der produktiven Datenbank den Backup-Modus beenden
- Clone-Parameter-File erzeugen
- Neue Datenbank mit den Clone-Dateien starten
- Neue Redo Logs und Temp File erzeugen
- Neue Datenbank mit Resetlogs öffnen

Es gibt jedoch eine einfachere Methode, Snapshots von Oracle-Datenbanken zu erzeugen. Oracle bietet dazu ein vorgefertigtes Skript; mit 12*c* und Multitenant sind diese Funktionen sogar in SQL enthalten.

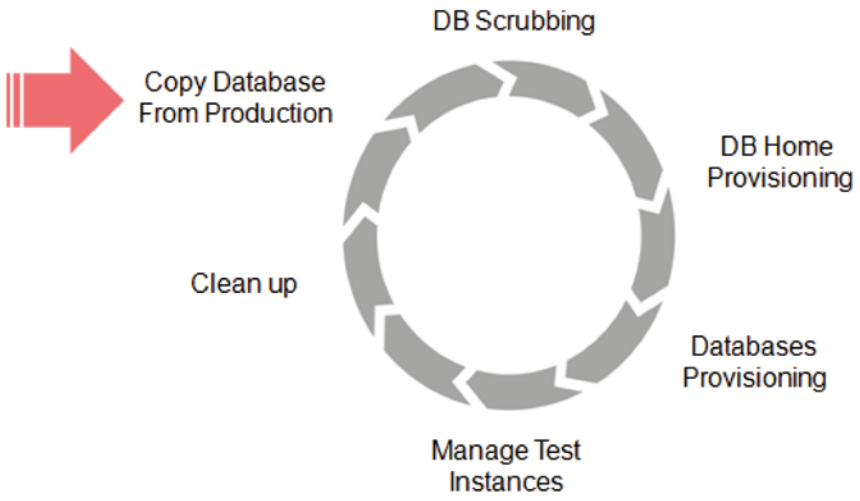


Abbildung 2: gDBClone-Skript

Pluggable Datenbanken und ACFS-Snapshots

Mit 12*c* Multitenant gibt es einen einfachen SQL-Befehl, um einzelne Pluggable Datenbanken (PDB) zu duplizieren. Hierfür muss lediglich die Ausgangs-PDB für die Zeitspanne der Kopie im Read-only-Modus sein. Dies funktioniert etwas anders als das Klonen von 12*c* nonCDB und 11*g*R2-Datenbanken, da hierfür nicht der Backup-Modus verwendet wird. Das Kopieren funktioniert generell auch ohne Snapshot-fähiges Dateisystem, dann muss die Ausgangs-Datenbank aber erheblich länger im Read-only-Modus bleiben *(siehe Listing 4)*.

Befinden sich alle Daten der Original-PDB bereits auf ACFS, sorgt der einfache Zusatz „snapshot copy" dafür, dass der PDB-Klon quasi instant erzeugt wird: „SQL> create pluggable database clonedb from spdb snapshot copy;". Letztendlich ist das alles, was der DBA tun muss, um einen Snapshot einer PDB zu erzeugen. Im Hintergrund passiert aber einiges, um diesen Snapshot zur Verfügung zu stellen:

- Snapshot erstellen
- Für die neue PDB neue Verzeichnisse erzeugen
- Die Daten im Snapshot über symbolische Links verknüpfen
- Den temporären Tablespace neu anlegen

Abbildung 1 zeigt die Abhängigkeiten in der Dateistruktur. Die erzeugten Strukturen erkennt man aus der eindeutigen PDB-ID, der sogenannten „GUID" *(siehe Listing 5)*. Diese Information spiegelt sich auch im Dateisystem wieder und man erkennt die erzeugten symbolischen Links auf den Snapshot *(siehe Listing 6)*. Das Verzeichnis, in dem ACFS Snapshots erzeugt, ist „.ACFS". Da es sich um einen versteckten Systemordner handelt, ist dieser auch nicht so einfach zu finden, wenn man ihn nicht kennt. Er wird weder mit dem Befehl „find" noch mit einem „list –l" entdeckt, man kann nur explizit mit dem Befehl „cd" dorthin wechseln. Selbstverständlich erhält man nun auch mit dem Befehl „acfsutil" die entsprechenden Snapshot-Informationen *(siehe Listing 7)*.

Ein Löschen der PDB löscht im Normalfall auch den erzeugten Snapshot. Umgekehrt funktioniert dies allerdings nicht: Wird versucht, den Snapshot mit dem Befehl „acfsutil" zu löschen, funktioniert dies bei geöffneter PDB nicht.

Snapshots von 11.2-Datenbanken und 12*c* nonCDB

Nicht ganz so komfortabel funktioniert das Klonen von 12*c* nonCDB und 11.2-Datenbanken. Aber auch hierfür gibt es ein Perl-Skript, das Oracle auf dem Technology Network bereitstellt, das sogenannte „gDBClone Skript" *[3]*. Allerdings kann gDBClone weit mehr als nur einen Snapshot einer Datenbank auf ACFS zu erzeugen. Da produktive Datenbanken selten bereits auf ACFS sind, kann gDBClone auch im ersten Schritt eine komplette Kopie mit RMAN-Mitteln auf ACFS erzeugen. gDBClone unterstützt somit einen kompletten Testzyklus, wie *Abbildung 2* zeigt.

gDBClone ist ein frei verfügbares Perl-Skript, das sich auch nach Bedarf anpassen lässt. Es ist von den Entwicklern erstellt worden, die dieselbe Funktionalität im Command Line Interface der Oracle Database Appliance („oakcli") implementiert haben, und verfügt über einen ähnlichen Funktionsumfang.

Das Skript wird als „Root"-User aufgerufen und kann von kompletten RAC-Datenbanken eine Kopie erzeugen. Dabei kann das Ziel eine Single-Instanz oder auch eine RAC-Datenbank sein. Es geht davon aus, dass sowohl auf dem Quell- als auch auf dem Ziel-System die Grid-Infrastruktur existiert.

Im ersten Schritt fertigt gDBClone aus einem RMAN-Backup oder direkt über das Netzwerk von einem produktiven Cluster eine Datenbank-Kopie an. Theoretisch kann gDBClone auch von Datenbank-Dateien direkt von ASM nach ASM oder von einem Filesystem nach ASM kopieren. Das ist offiziell jedoch nicht unterstützt. Ziel (auch für das weitere Klonen) sollte ein ACFS-Filesystem sein. Befindet sich die Datenbank bereits auf ACFS, kann selbstverständlich auch gleich ein Snapshot ohne vorherige Kopie erzeugt werden. gDBClone verfügt über eine gute Hilfefunktion, um die einzelnen Möglichkeiten aufzulisten *(siehe Listing 8)*:

- Die Clone-Funktionalität erstellt (anders als der Name vermuten lässt) eine vollständige Kopie einer Datenbank in den Ziel-Cluster auf ACFS und kann alternativ auch auf ein RMAN-Backup zugreifen
- Snap dagegen ist das Anlegen eines Snapshots in ACFS und verwendet darin implizit die oben genannten Schritte

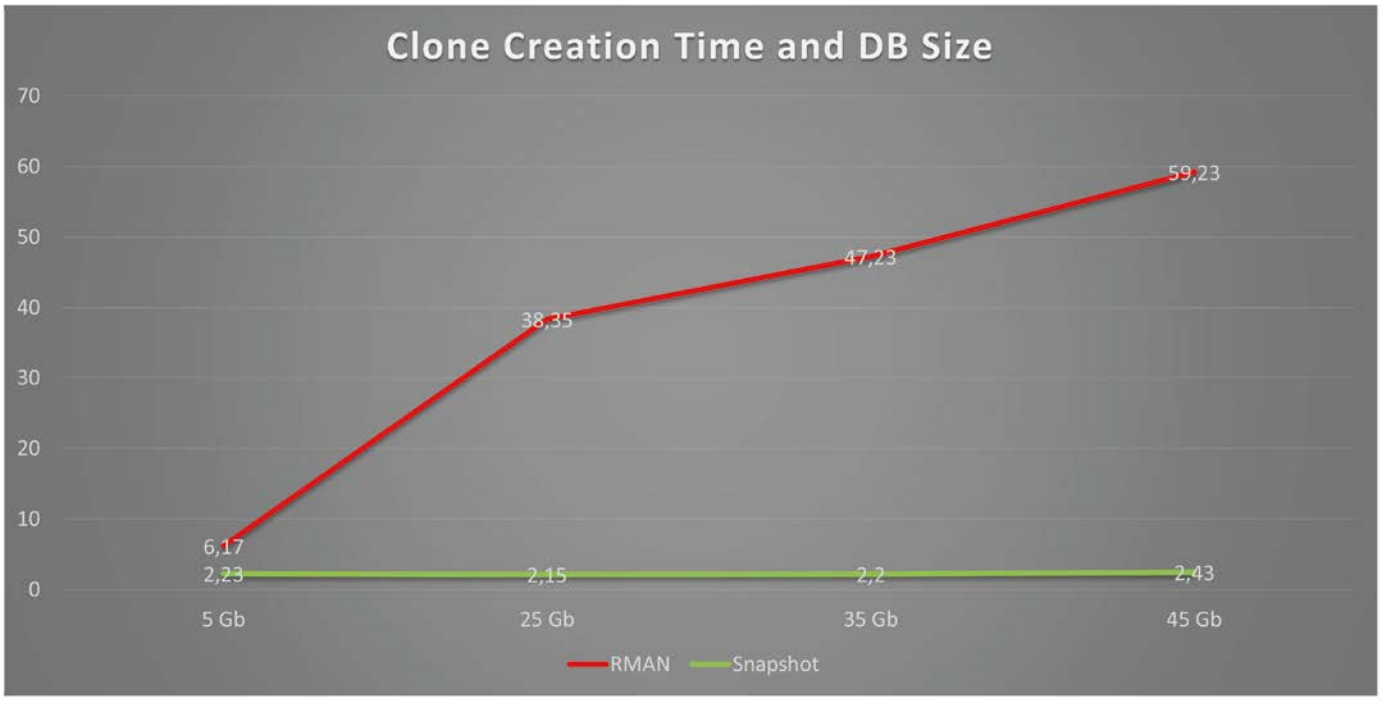


Abbildung 3: RMAN vs. Snapshot

- „ListDB“ listet dabei alle vorhandenen Snapshots auf
- „DelDB“ löscht die kopierten Datenbanken beziehungsweise Snapshots

Im zweiten Schritt werden dann aus der erzeugten Kopie die platzsparenden Snapshots für das Testen und die Entwicklung erstellt. Wie viel Zeit ein Snapshot einsparen kann, hat Oracle mit ein paar Messungen auf einem relativ kleinen System einmal nachgestellt. *(siehe Abbildung 3)*. Je größer die Datenbank ist, desto eher machen sich die Geschwindigkeitsvorteile eines Snapshots bemerkbar. Letztendlich besteht das Erzeugen einer Kopie der Produktions-Datenbank auf ACFS und der Snapshots aus zwei kleinen Befehlen mit gDBClone *(siehe Listing 9)*.

Nähere Informationen zu den einzelnen Schritten von gDBClone stehen im Whitepaper, das ebenfalls auf OTN zu finden ist. Darüber hinaus haben sich folgende Tipps als sehr wertvoll herausgestellt:

- Die Source-Datenbank muss im Archivelog-Modus sein
- Die Source-Datenbank darf sich nicht im Backup-Modus befinden
- gDBClone muss als Root ausgeführt werden
- Datenbank-Namen sollten keine Sonderzeichen enthalten
- Die Befehle am besten, bis man alle Voraussetzungen kennt, mit der Debug-Option („-debug“) starten
- Quell- und Ziel-Datenbank müssen „Administrator managed“ sein

gDBClone loggt während der Ausführung alle Informationen unter „/var/log/gDBClone“. Hier finden sich auch die erzeugten Dateien, etwa für das RMAN-Skript zur Kopie. Grundsätzlich spricht aber auch nichts dagegen, selbst Hand anzulegen, sollte das Skript nicht in Ihrer Umgebung funktionieren.

Fazit

Das ACFS-Filesystem bietet eine kostenlose und schnelle Alternative, um platzsparende Datenbank-Kopien zu erzeugen, zumindest wenn man eine Enterprise Edition einsetzt. Bei 12*c* Multitenant ist diese Funktionalität direkt in SQL integriert. Aber auch für ältere Datenbank-Versionen oder ohne Multitenant lässt sich mithilfe des gDBClone-Skripts viel Arbeit sparen. Selbst eine Kombination von beidem ist denkbar, wenn die komplette Multitenant-Datenbank mit all ihren PDBs erst auf ein Test-System kopiert und dort dann Snapshots zu Test- und Entwicklungszwecken erzeugt werden sollen.

Links & Referenzen

[1] Database Licensing Information Oracle ASM Cluster File System (Oracle ACFS): *http://docs.oracle.com/database/121/DBLIC/editions.htm#CIHDDJCJ*

[2] Automatic Storage Management Administrator's Guide, About Oracle ACFS and Database Data Files: *http://docs.oracle.com/database/121/OSTMG/GUID-EAE5BECA-E057-41B2-9E26-B755B-F7F9947.htm*

[3] gDBClone Database Clone/Snapshot Management Script: *http://www.oracle.com/technetwork/indexes/samplecode/gdbclone-download-2295388.html*

[4] *http://download.oracle.com/otn/samplecode/Managing-Test-Dev-Envs-WP-5-2015.pdf*

Hinweis: Sämtliche Listings finden sie online unter: *http://www.doag.org/go/redstack/solbach_abb*

Sebastian Solbach
sebastian.solbach@oracle.com

Der hierarchische Profiler

Jürgen Sieben, ConDeS GmbH & Co. KG

Der hierarchische Profiler steht seit der Datenbank-Version 11*g* bereit, um die zuvor vorhandenen PL/SQL-Profiling-Tools abzulösen. Neben einer einfachen Bedienung und einer guten Integration in den SQL-Developer bietet er auch neue Einblicke in die Abläufe von PL/SQL-Programmen. Dadurch wird das Werkzeug zu einem unverzichtbaren Hilfsmittel, wäre da nicht eine entscheidende Limitierung …

Der hierarchische Profiler unterstützt die Entwicklung effizienten PL/SQL-Codes durch die Darstellung der Aufruf-Hierarchie und der in den einzelnen Programmteilen aufgewendeten Zeit sowie durch das Zählen der Aufrufe der verschiedenen Methoden und SQL-Anweisungen. Ist er einmal eingerichtet, arbeitet es sich ganz entspannt und die Einblicke in den Code sind durchaus interessant.

Voraussetzungen und Einrichtung

Um den hierarchischen Profiler einrichten zu können, muss der Nutzer ein Ausführungsrecht an dem Package „DBMS_HPROF“ besitzen und in seinem Schema das Skript „<Oracle_Home>/rdbms/admin/dbmshptab.sql“ ausführen. Dieses Skript legt drei Tabellen an:

- „DBMSHP_RUNS“
  Sie zeigt für jeden Profiler-Lauf eine Zeile mit einer ID, dem Zeitpunkt der Messung, der gesamt benötigten Zeit sowie einem optionalen Kommentar, der durch das Package „DBMS_HPROF“ gesetzt werden kann (Details später).

- „DBMSHP_FUNCTION_INFO“
  Die eigentliche Haupt-Tabelle mit detaillierten Angaben über den aufgerufenen Code, die Zeit, die darin verbracht wurde, und mehr.

- „DBMSHP_PRANET_CHILD_INFO“
  Sie zeigt die hierarchische Verbindung der Aufrufe. Als Referenz kommt die Spalte „SYMBOLID“ der Tabelle „DBMHP_FUNCTION_INFO“ zum Einsatz, hier als Spalten „PARENTSYMID“ und „CHILDSYMID“.

Zudem wird die Sequenz „DBMSHP_RUNNUMBER“ angelegt, die für die Vergabe der „RUNID“ in Tabelle „DBMSHP_RUNS“ verwendet wird. Nicht automatisch durch

das Skript angelegt wird ein Directory-Objekt mit dem Namen „PLSHPROF_DIR“, das aber erforderlich ist, um den hierarchischen Profiler zu verwenden. Es muss diesen Namen haben, wenn der hierarchische Profiler aus dem SQL-Developer heraus genutzt werden soll, ansonsten kann ein beliebiges Directory-Objekt durch die Methoden des Package „DBMS_HPROF“ angesprochen werden. Wird der Profiler aus SQL-Developer aufgerufen und das Directory existiert noch nicht, kann auch der SQL-Developer das Verzeichnis anlegen.

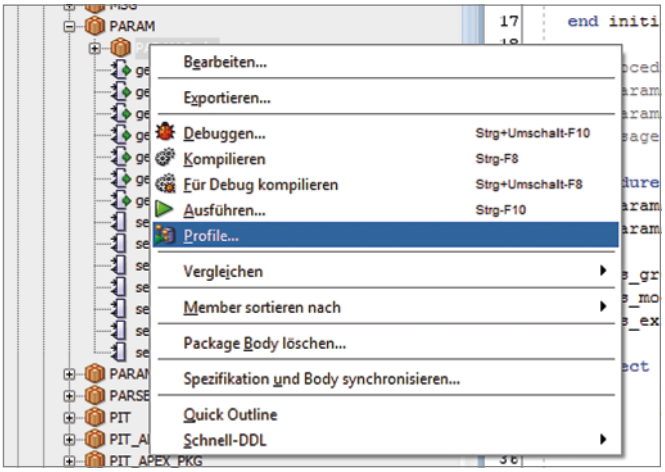


Abbildung 1: Auswahl einer Methode für das Profiling

Nutzung des hierarchischen Profilers aus dem SQL-Developer

Anschließend kann jede vorhandene Methode durch einen Rechtsklick auf deren Namen im SQL-Developer direkt für das Profiling ausgewählt werden *(siehe Abbildung 1)*. Der anschließende Dialog ist analog zum Dialog für das Debugging aufgebaut und erlaubt es, die Methode durch einen anonymen Block mit entsprechenden Parametern aufzurufen. *Abbildung 2* zeigt dies beispielhaft für eine Parameter-Funktion. Nach dem Bestätigen des Dialogs werden der Profiler-Lauf sofort unter dem Reiter „Profile“ des Package eingeblendet und ein fortlaufend nummerierter Eintrag in der oberen Tabelle angezeigt. Nun kann das Ergebnis des Laufs begutachtet und analysiert werden *(siehe Abbildung 3)*.

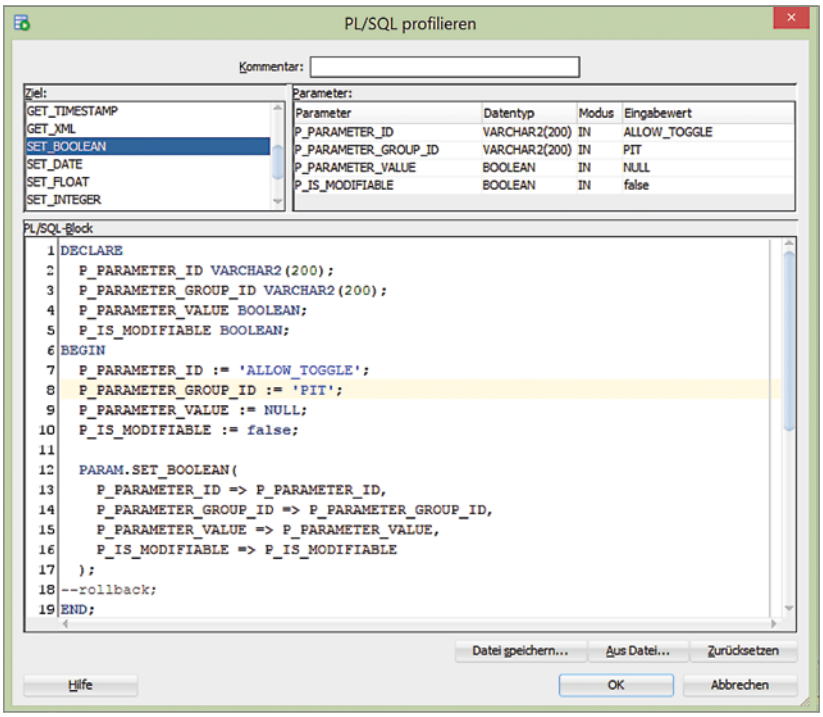


Abbildung 2: Einrichten eines Profiler-Laufs

Soweit, so gut und so hilfreich. Fokus dieser Auswertungen ist es, die Ausfüh-

Code | Berechtigungen | Referenzen | Abhängigkeiten | Details | Profile | Fehler

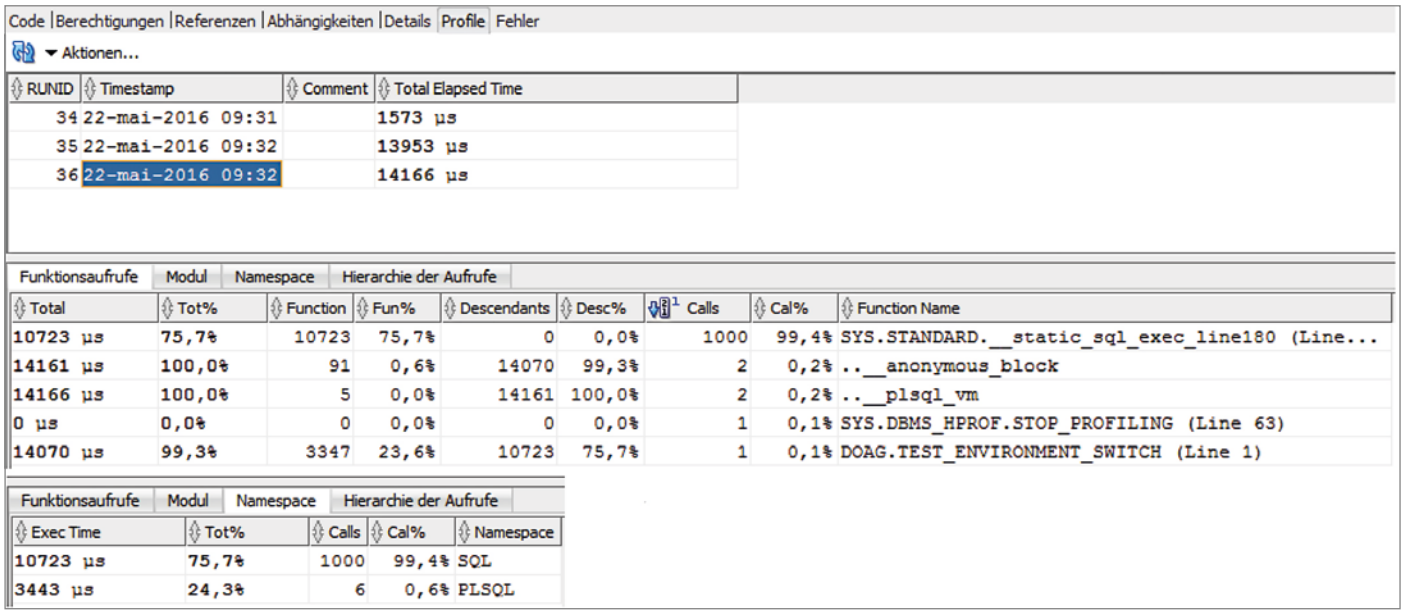
Aktionen...

RUNID	Timestamp	Comment	Total Elapsed Time
32	21-mai-2016 02:12		21371 µs

Funktionsaufrufe | Modul | Namespace | Hierarchie der Aufrufe

Total	Tot%	Function	Fun%	Descendants	Desc%	Calls	Cal%	Function Name
21371 µs	100,0%	7	0,0%	21364	100,0%	2	22,2%	..__plsql_vm
21364 µs	100,0%	232	1,1%	21132	98,9%	2	22,2%	..__anonymous_block
0 µs	0,0%	0	0,0%	0	0,0%	1	11,1%	SYS.DBMS_HPROF.STOP_PROFILING (Line 63)
5238 µs	24,5%	5238	24,5%	0	0,0%	1	11,1%	DOAG.PARAM.__static_sql_exec_line118 (Line 118)
15825 µs	74,0%	15825	74,0%	0	0,0%	1	11,1%	DOAG.PARAM.__static_sql_exec_line138 (Line 138)
21132 µs	98,9%	18	0,1%	21114	98,8%	1	11,1%	DOAG.PARAM.SET_BOOLEAN (Line 278)
21114 µs	98,8%	51	0,2%	21063	98,6%	1	11,1%	DOAG.PARAM.SET_PARAMETER (Line 102)

Abbildung 3: Darstellung der Auswertungsergebnisse



Code | Berechtigungen | Referenzen | Abhängigkeiten | Details | Profile | Fehler

Aktionen...

RUNID	Timestamp	Comment	Total Elapsed Time
34	22-mai-2016 09:31		1573 µs
35	22-mai-2016 09:32		13953 µs
36	22-mai-2016 09:32		14166 µs

Funktionsaufrufe | Modul | Namespace | Hierarchie der Aufrufe

Total	Tot%	Function	Fun%	Descendants	Desc%	Calls	Cal%	Function Name
10723 µs	75,7%	10723	75,7%	0	0,0%	1000	99,4%	SYS.STANDARD.__static_sql_exec_line180 (Line...
14161 µs	100,0%	91	0,6%	14070	99,3%	2	0,2%	..__anonymous_block
14166 µs	100,0%	5	0,0%	14161	100,0%	2	0,2%	..__plsql_vm
0 µs	0,0%	0	0,0%	0	0,0%	1	0,1%	SYS.DBMS_HPROF.STOP_PROFILING (Line 63)
14070 µs	99,3%	3347	23,6%	10723	75,7%	1	0,1%	DOAG.TEST_ENVIRONMENT_SWITCH (Line 1)

Funktionsaufrufe | Modul | Namespace | Hierarchie der Aufrufe

Exec Time	Tot%	Calls	Cal%	Namespace
10723 µs	75,7%	1000	99,4%	SQL
3443 µs	24,3%	6	0,6%	PLSQL

Abbildung 4: Ergebnis des Profiler-Laufs für 1.000 Iterationen

```
create or replace procedure test_environment_switch(
   p_iterations in number)
as
   l_result varchar2(200);
   c_msg_template constant varchar2(100) :=
       'Iteration #i# für Benutzer #USER# ausgeführt.';
begin
   for i in 1 .. p_iterations loop
       l_result := replace(
           replace(c_msg_template, ,#i#', i),
           '#USER#', user);
   end loop;
end test_environment_switch;
/
```

Listing 1

rungszeiten sowie die entstehenden Umgebungswechsel zu analysieren. Unter dem Reiter „Namespace" findet sich zum Beispiel die prozentuale Verteilung der Rechenzeit zwischen den Namensräumen „SQL" und „PL/SQL", im Reiter „Modul" stehen die Ausführungszeiten pro Package etc. Das nachfolgende Beispiel zeigt die Nützlichkeit dieser Auswertungen. Wir testen eine Prozedur, die versteckte Umgebungswechsel enthält, und analysieren, ob der hierarchische Profiler diese aufzeigen kann. *Listing 1* zeigt die Prozedur. In *Abbildung 4* ist das Ergebnis eines Profiler-Laufs mit 1.000 Iterationen dargestellt.

Die Umgebungswechsel sind sowohl in der Darstellung der Funktionsaufrufe als auch in der Verteilung der Rechenzeit im Reiter „Namespace" klar zu erkennen. Nicht zu erkennen ist jedoch, dass der Aufruf der Funktion „USER" für die Umgebungswechsel verantwortlich ist. Die Funktion ist, wie ein Blick in das Package „SYS.STANDARD" zeigt, als „select user from dual;" implementiert. Einen deutlichen Fingerzeig in diese Richtung liefert die Analyse spätestens mit der Angabe der Zeilennummer, bei der der Umgebungswechsel erfolgt.

Ein komplexerer Fall und ein Problem

Listing 2 zeigt einige aufeinander zeigende Prozeduren. Die Idee dahinter: Prozedur „A" ruft zunächst Funktion „C" und anschließend in einer Schleife mehrfach Prozedur

```
create or replace procedure e
as
begin
   null;
end e;
/

create or replace procedure d
as
begin
   null;
end d;
/

create or replace procedure c
as
begin
   null;
end c;
/

create or replace procedure b(
   p_bucket in number)
as
begin
   case mod(p_bucket, 2)
   when 0 then d;
   else e;
   end case;
end b;
/

create or replace procedure a
as
begin
   c;
   for i in 1 .. 30 loop
       b(i);
   end loop;
   c;
end a;
/
```

Listing 2

„B“ auf, die anhand einer einfachen Logik entscheidet, entweder Prozedur „D“ oder „E“ aufzurufen. Zum Abschluss wird noch einmal Prozedur „C“ aufgerufen. *Abbildung 5* zeigt das Ergebnis der hierarchischen Darstellung der Prozedur „A“. Hier wird nun allerdings fälschlicherweise der Eindruck erweckt, es werde zunächst dreißigmal Prozedur „B“ und darin jeweils die Prozeduren „D“ und „E“ aufgerufen, und erst anschließend zweimal die Prozedur „C“. Ein Bug? Um das zu untersuchen, müssen wir die Arbeitsweise des Werkzeugs verstehen und überlegen, ob es andere Möglichkeiten der Auswertung gibt.

Die Arbeitsweise des hierarchischen Profilers

Wie gesagt, erfolgt das eigentliche Profiling einer Prozedur durch das Package „DBMS_HPROF“. Es stellt die Prozeduren „START_PROFILING“, „STOP_PROFILING“ und die Funktion „ANALYZE“ zur Verfügung. Diese liefert als Ergebnis eine „RUNID“, unter der in den angesprochenen Tabellen die Ergebnisse erfragt werden können. Wenn man also im SQL-Developer ein Profil erstellen lässt, wird im Hintergrund in etwa Code gemäß *Listing 3* ausgeführt.

Das Package „DBMS_HPROF“ macht jedoch nicht die eigentliche Arbeit, sondern veranlasst diese nur. Eine externe Funktion führt das Tracing durch und speichert alle Ergebnisse in einer externen Datei, die im Fall des SQL-Developer „test.trc“ heißt und im Verzeichnis „PLSHPROF_DIR“ liegt. Die Daten dieser Datei sind ein wenig technischer als die Tabelle und in *Listing 4* in Auszügen dargestellt.

Es werden folgende Einträge unterschieden:

- „P#V“: Version der hierarchischen Profilers
- „P#!“: Kommentar
- „P#C“: Aufruf einer Prozedur/Funktion (Call-Event)
- „P#R“: Rückkehr aus einer Prozedur/Funktion (Return-Event)
- „P#X“: Zeitnahme in Mikrosekunden

Interessant sind neben den Zeiten vor allem die Call-Events, denn sie enthalten die konkreten Prozedur-Namen, die Zeilennummern und den Typ des Objekts,

```
declare
   l_runid number;
begin
   dbms_hprof.start_profiling(
       location => 'PLSHPROF_DIR'
       filename => 'test.trc');
   <Ausführung des anonymen Blocks>
   dbms_hprof.stop_profiling;
   l_runid := dbms_hprof.analyze(
          location => 'PLSHPROF_DIR',
          filename => 'test.trc');
end;
```

Listing 3

```
P#V PLSHPROF Internal Version 1.0
P#! PL/SQL Timer Started
P#C PLSQL."".""."__plsql_vm"
P#X 3
P#C PLSQL."".""."__anonymous_block"
P#X 77
P#C PLSQL."DOAG"."A"::7."A"#980980e97e42f8ec #1
P#X 26
P#C PLSQL."DOAG"."C"::7."C"#980980e97e42f8ec #1
P#X 1
P#R
P#X 16
P#C PLSQL."DOAG"."B"::7."B"#e17d780a3c3eae3d #1
P#X 24
P#C PLSQL."DOAG"."E"::7."E"#980980e97e42f8ec #1
P#X 0
P#R
...
P#R
P#C PLSQL."".""."__plsql_vm"
P#X 2
P#C PLSQL."".""."__anonymous_block"
P#X 48
P#C PLSQL."SYS"."DBMS_HPROF"::11."STOP_PROFILING"#9... #63
P#R
P#R
P#R
P#! PL/SQL Timer Stopped
```

Listing 4

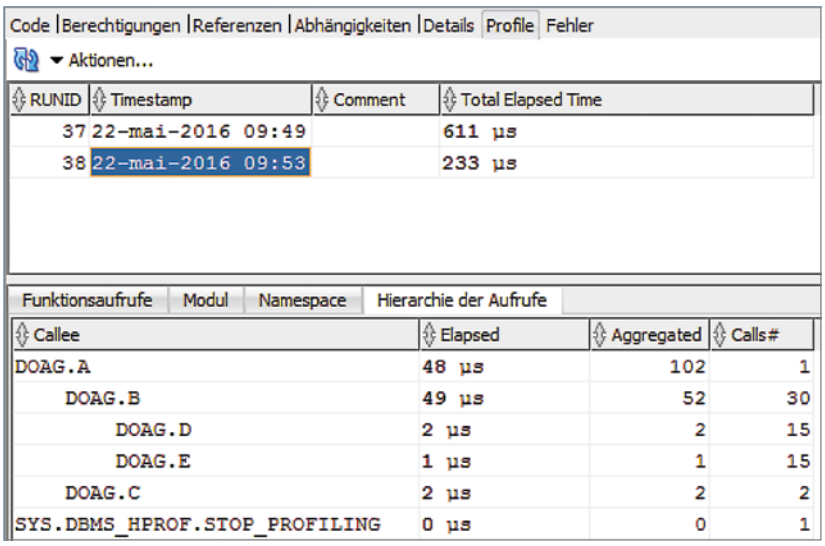


Abbildung 5: Ergebnis des komplexeren Tests

```
select symbolid, owner, module, type, function, line#, subtree_elapsed_time, function_elapsed_time, calls
  from dbmshp_function_info
 where runid = 38;
```

Skriptausgabe | Abfrageergebnis | Abfrageergebnis 1

SQL | Alle Zeilen abgerufen:8 in 0 Sekunden

	SYMBOLID	OWNER	MODULE	TYPE	FUNCTION	LINE#	SUBTREE_ELAPSED_TIME	FUNCTION_ELAPSED_TIME	CALLS
1	1	(null)	(null)	(null)	__anonymous_block	0	227	125	2
2	2	(null)	(null)	(null)	__plsql_vm	0	233	6	2
3	3	DOAG	A	PROCEDURE	A	1	102	48	1
4	4	DOAG	B	PROCEDURE	B	1	52	49	30
5	5	DOAG	C	PROCEDURE	C	1	2	2	2
6	6	DOAG	D	PROCEDURE	D	1	2	2	15
7	7	DOAG	E	PROCEDURE	E	1	1	1	15
8	8	SYS	DBMS_HPROF	PACKAGE BODY	STOP_PROFILING	63	0	0	1

Abbildung 6: Daten des oben gezeigten Profiler-Laufs

```
select *
  from dbmshp_all_functions_vw;
```

Skriptausgabe | Abfrageergebnis

SQL | 50 Zeilen abgerufen in 0,077 Sekunden

	RUNID	ID	PARENT_ID	CALL_LEVEL	SYMBOLID	CATEGORY	CATEGORY_NAME	NAMESPACE	OWNER	MODULE	FUNCTION
1	61	1	(null)	0	1	V	Version	Other	(null)	(null)	(null)
2	61	2	(null)	0	2	!	Comment	Other	(null)	(null)	(null)
3	61	3	2	1	3	C	Call	SQL	KISMON	UI_CREATE_RUN_PKG	__dyn_sql_exec_line196
4	61	4	3	2	4	C	Call	PLSQL	(null)	(null)	(null)
5	61	5	4	3	5	C	Call	PLSQL	(null)	(null)	(null)
6	61	6	5	4	6	C	Call	PLSQL	KISMON	APP_STANDARD	REFRESH_STAY_COMPLETE
7	61	7	6	5	7	C	Call	PLSQL	KISMON	PIT	TRACE_ALL
8	61	8	6	5	9	C	Call	PLSQL	KISMON	PIT	ENTER
9	61	9	8	6	10	C	Call	PLSQL	KISMON	PIT_PKG	ENTER
10	61	10	9	7	11	C	Call	PLSQL	KISMON	PIT_PKG	TRACE_ME

Abbildung 7: Detail-Darstellung, extrahiert aus der Trace-Datei

das aufgerufen wird. Basierend auf diesen Daten füllt die Funktion „ANALYZE" die Tabellen mit Daten und gibt die erzeugte „RUNID" an die aufrufende Umgebung zurück.

Die Grenzen der Auswertbarkeit

Eine Analyse der erzeugten Tabellendaten zeigt, dass der hierarchische Profiler die Informationen, die für die korrekte hierarchische Anzeige der Aufrufe erforderlich sind, gar nicht (mehr) besitzt, denn unter der „RUNID" des gezeigten Laufs sind nur die in *Abbildung 6* gezeigten Daten in der Tabelle „DBMSHP_FUNCTION_INFO".

Aus diesen Daten geht hervor, dass bereits eine Aggregation der Aufrufe erfolgt ist. Zeile 4 zeigt nur eine einzige Zeile für die dreißig Aufrufe der Funktion „B", ebenso wie Zeile 5 nur einen Eintrag für beide Aufrufe der Funktion „C" zeigt. Die hierarchische Beziehung ist nun also nur noch über die aggregierten Aufrufe der Prozeduren möglich, eine detailliertere Betrachtungsweise gibt es nicht mehr. Daraus folgt auch, dass die zeitliche korrekte Wiedergabe der Aufruf-Hierarchie nicht mehr möglich ist.

Dass die Prozedur „C" vor der Prozedur „B" aufgerufen wurde, ist nicht mehr ersichtlich. Natürlich ist diese Aggregation nicht völlig sinnlos, wenn man sich etwa vorstellt, man hätte mehrere Tausend Aufrufe einer Funktion. Wenn aber das Ziel die konkrete Analyse eines komplexen Aufruf-Szenarios ist, reichen die angebotenen Informationen nicht aus.

Ein Lösungsansatz

Da die Tabellen nur noch voraggregierte Daten enthalten und die korrekte zeitliche und hierarchische Abfolge nicht mehr darstellen können, scheiden sie für eine genauere Darstellung des tatsächlichen Verhaltens des Codes aus. Doch es gibt einen naheliegenden Ansatz, das Problem zu lösen: Da die Daten ja als Textdatei zur Verfügung stehen, können wir die Extraktion der Daten in eine Datenbank-Tabelle selbst organisieren und dabei die Schwächen der Standard-Implementierung umgehen. Eins allerdings vorweg: Prozeduren, die durch „Code-Inlining" oder durch Optimierung einer „Deterministic"-Funktion nicht ausgeführt wurden, sind auch in der Trace-Datei nicht zu finden.

Beginnen wir also mit einer externen Tabelle auf die Trace-Datei. Unter *„www.*

doag.org/go/redstack/jsieben/skripte" stehen die Skripte zur Erzeugung der externen Tabelle und der Ziel-Tabelle „DBMSHP_ALL_FUNCTIONS" sowie die „select"-Anweisung zum Kopieren der importierten Daten in die Ziel-Tabelle. Alternativ ist die Abfrage zum Einfügen der Daten auch als View „DBMSHP_ALL_FUNCTIONS_VW" verfügbar, falls man die Daten nur analysieren, aber nicht speichern möchte. *Abbildung 7* zeigt das Ergebnis des Imports.

Ausgehend von dieser Datenbasis, lässt sich nun anhand der Spalte „ID" die genaue Aufruf-Reihenfolge nachvollziehen; die Spalten „MSEC_TOT", „MSEC_PER_METHOD" sowie „CUM_MSEC" enthalten die aufsaldierten Zeitangaben aus der Trace-Datei. Dabei gilt:

- „MSEC_TOT" setzt sich aus „CALL" und der „RETURN"-Zeit der entsprechenden Methode (die als „MSEC_PRE/POST_SUB" auch separat vorhanden ist) zusammen.
- „MSEC_PER_METHOD" gruppiert nach dem Namen (der maßgeblich für die Vergabe der Symbol-ID ist) der Methode.
- „CUM_MSEC" saldiert die Zeiten der einzelnen Methoden in der korrekten Ausführungs-Reihenfolge auf

Um die neuen Möglichkeiten in ein halbautomatisiertes Verfahren zu integrieren, kann man den Aufruf des hierarchischen Profilers aus dem Beispiel leicht um eine entsprechende „insert"-Anweisung zum parallelen Füllen der neuen Tabelle ergänzen *(siehe Listing 5)*.

```
declare
   l_runid number;
begin
   dbms_hprof.start_profiling(
       location => 'PLSHPROF_DIR',
       filename => 'test.trc');
   <Ausführung des anonymen Blocks>
   dbms_hprof.stop_profiling;
   l_runid := dbms_hprof.analyze(
          location => 'PLSHPROF_DIR',
          filename => 'test.trc');
   insert into dbmshp_all_functions
   select *
       from dbmshp_all_functions_vw v;
   commit;
end;
```

Listing 5

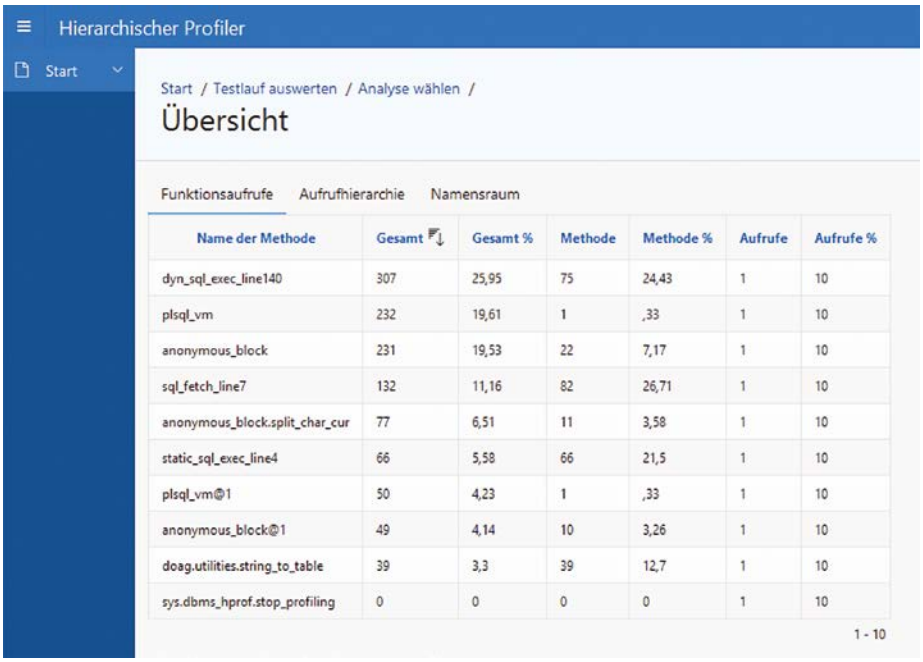


Abbildung 8: Visualisierung der Testläufe

Die Visualisierung der Ergebnisse

Ein großer Vorteil der Integration des hierarchischen Profilers in den SQL-Developer ist die unmittelbare Visualisierung der Test-Ergebnisse. Dies ist naturgemäß mit den neu erhobenen Daten nicht gegeben. Um dieses Manko zu beseitigen, bietet es sich an, eine kleine Apex-Anwendung zu erstellen, die die bestehenden Testläufe visualisieren kann. Dabei ist natürlich dem Entdeckerdrang keine Grenze gesetzt *(siehe Abbildung 8)*.

Die Anwendung kann hier unter *„www.doag.org/go/redstack/jsieben/anwendung"* heruntergeladen werden. Sie stellt allerdings in keiner Weise produktionsfertigen Code dar, sondern ist im Gegenteil eine prototypisch erstellte Version zur Dokumentation der grundsätzlichen Vorgehensweise und der Möglichkeiten. So sind sicher nicht alle interessanten Zusammenstellungen der Daten visualisiert, der Code Editor (der aus dem mitgelieferten Plug-in der Apex-Entwickleranwendung entliehen wurde) implementiert nicht alle Möglichkeiten bezüglich Validierung und Sicherheit. Als Startpunkt für eigene Anwendungen reicht es aber sicher aus. Die Lösung erlaubt die Definition neuer Testläufe (wenn auch nicht mit dem Komfort des SQL-Developer, der entsprechenden Stub-Code zum Aufruf des Tests bereits erzeugt), das Ausführen der Testläufe und die Visualisierung der Testergebnisse. Zu den einzelnen Limitierungen der Anwendung stehen auf den Seiten entsprechende Hinweise.

Jürgen Sieben
j.sieben@condes.de

Wir begrüßen unsere neuen Mitglieder

Persönliche Mitglieder

- Erik Zimmermann
- Bernd Hubert
- Ingo Wevers
- Alexander Weber
- Michael Künzner
- Dominik Watts
- Horst Tessmann
- Stefan Graf
- Jacob Bogers
- Vladimir Lifchits
- Petra Ella Kintzel
- Uwe Hesse
- Philip Riese

Firmenmitglieder DOAG

- SITA Airport IT GmbH, Michael Theil

Neumitglieder SOUG

- CND Computer + Network Division AG, Marco Schmucki
- CND Computer + Network Division AG, Matthias Stähli
- Oracle Switzerland, Andreas Oswald

Impressum

Red Stack Magazin wird gemeinsam herausgegeben von den Oracle-Anwendergruppen DOAG Deutsche ORACLE-Anwendergruppe e.V. (Deutschland, Tempelhofer Weg 64, 12347 Berlin, *www.doag.org*), AOUG Austrian Oracle User Group (Österreich, Lassallestraße 7a, 1020 Wien, *www.aoug.at*) und SOUG Swiss Oracle User Group (Schweiz, Dornacherstraße 192, 4053 Basel, *www.soug.ch*).

Red Stack Magazin ist das User-Magazin rund um die Produkte der Oracle Corp., USA, im Raum Deutschland, Österreich und Schweiz. Es ist unabhängig von Oracle und vertritt weder direkt noch indirekt deren wirtschaftliche Interessen. Vielmehr vertritt es die Interessen der Anwender an den Themen rund um die Oracle-Produkte, fördert den Wissensaustausch zwischen den Lesern und informiert über neue Produkte und Technologien.

Red Stack Magazin wird verlegt von der DOAG Dienstleistungen GmbH,Tempelhofer Weg 64, 12347 Berlin, Deutschland, gesetzlich vertreten durch den Geschäftsführer Fried Saacke, deren Unternehmensgegenstand Vereinsmanagement, Veranstaltungsorganisation und Publishing ist.

Die DOAG Deutsche ORACLE-Anwendergruppe e.V. hält 100 Prozent der Stammeinlage der DOAG Dienstleistungen GmbH. Die DOAG Deutsche ORACLE-Anwendergruppe e.V. wird gesetzlich durch den Vorstand vertreten; Vorsitzender: Stefan Kinnen. Die DOAG Deutsche ORACLE-Anwendergruppe e.V. informiert kompetent über alle Oracle-Themen, setzt sich für die Interessen der Mitglieder ein und führen einen konstruktiv-kritischen Dialog mit Oracle.

Redaktion:
Sitz: DOAG Dienstleistungen GmbH (Anschrift s.o.)
Chefredakteur (ViSdP): Wolfgang Taschner
Kontakt: redaktion@doag.org
Weitere Redakteure (in alphabetischer Reihenfolge): Gaetano Bisaz, Mylène Diacquenod, Marina Fischer, Klaus-Michael Hatzinger, Sebastian Höing, Jan Peterskovsky, Fried Saacke

Titel, Gestaltung und Satz:
Alexander Kermas, DOAG Dienstleistungen GmbH (Anschrift s.o.)

Fotonachweis:
Titel: © 1tjf/123rf.com
Foto S. 11: © madpixblue/123rf.com
Foto S. 17: © wklzzz/123rf.com
Foto S. 21: © stylephotographs/123rf.com
Foto S. 27: © dotshock/123rf.com
Foto S. 37: © scanrail/123rf.com
Foto S. 41: © gekaskr/123rf.com
Foto S. 52: © Le Moal Olivier/123rf.com
Foto S. 60: © Torbz/fotolia.com

Anzeigen:
Simone Fischer, DOAG Dienstleistungen GmbH (verantwortlich, Anschrift s.o.)
Kontakt: anzeigen@doag.org
Mediadaten und Preise unter:
www.doag.org/go/mediadaten

Druck:
adame Advertising and Media GmbH,
www.adame.de

Inserentenverzeichnis

DBConcepts GmbH *www.dbconcepts.at*	S. 51	E-3 Magazin *www.e-3.de*	S. 35	Performing Databases GmbH *www.performing-databases.com*	S. 45
dbi services ag *www.dbi-services.com*	S. 39	Libelle AG *www.libelle.com*	S. 13	Sphinx IT Consulting GmbH *www.sphinx.at*	S. 9
DOAG e.V. *www.doag.org*	S. 19, U 2, U 3	MuniQsoft GmbH *www.muniqsoft.de*	S. 3	Trivadis GmbH *www.trivadis.com*	U 4

JavaLand

Schnell Ticket & Hotelzimmer sichern!
Early-Bird-Preise bis 30.01.2017

28.-30. März 2017 in Brühl

Programm online!

www.javaland.eu

Präsentiert von: DOAG Heise Medien

Community Partner: iJUG Verbund