

# DOAG SOUG News

## Konsolidierung räumt die IT auf



### Best Practice

- Inventur im ERP-System
- Faktenbasierte Indexierung

### Im Interview

Prof. Dieter Kempf,  
Präsident des Branchenverbands BITKOM



### Experten-Tipp

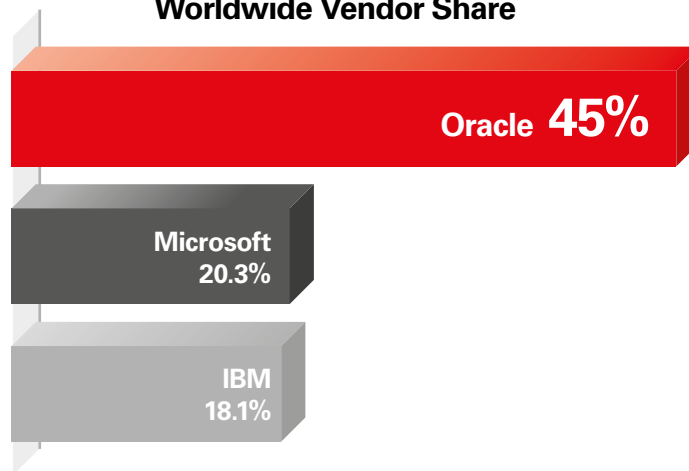
Jetzt oder später: Upgrade  
nach Oracle Database 12c

**STILL**

**#1**

# Database

## Worldwide Vendor Share



## Oracle Database

Trusted by 308,000 Customers Worldwide

**ORACLE®**

[oracle.com/database](http://oracle.com/database)  
or call 0800 1 81 01 11

Source: IDC, „Worldwide Relational Database Management Systems 2013- 2017 Forecast and 2012 Vendor Shares,” IDC #241292, May 2013; Table 1 (Relational Database Management Systems). Vendor share based on software license and maintenance revenue.

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.



Dietmar Neugebauer  
Vorstandsvorsitzender  
der DOAG

### Herzlich willkommen SOUG

Mit dieser Ausgabe starten wir eine Zusammenarbeit mit der Swiss Oracle User Group (SOUG). Die bisherige DOAG News wird unter dem Namen DOAG/SOUG News jetzt auch an die SOUG-Mitglieder verschickt.

Die Kooperation ist für mich eine klassische Win-win-Situation: Die Schweizer Oracle-Anwender profitieren von mehr und umfangreicheren Ausgaben im Jahr und die DOAG-Mitglieder erhalten ab sofort acht Seiten mehr Inhalt pro Zeitschrift. Ein Doppelsieg also für den Erfahrungsaustausch und die Wissensvermittlung der Anwender technischer Oracle-Produkte.

Aber auch die Schweizer Anwender der Oracle-Business-Solutions kommen nicht zu kurz. Für sie wird künftig der DOAG/SOUG News die DOAG Business News beigelegt. Ein weiteres Highlight aus dem DOAG-Verlagsprogramm – die Zeitschrift Java aktuell – ist schon seit Jahren für Interessierte in der DOAG und in der SOUG im Lieferumfang enthalten, da ja beide Vereine auch Mitglied im Interessenverbund der Java User Groups e.V. (iJUG) sind.

Ihr *Dietmar Neugebauer*

### Grüezi mitenand

Auch aus der Schweiz ein herzliches „Grüezi“ an alle DOAG- und SOUG-Mitglieder. Es freut mich sehr, dass die DOAG und die SOUG die bis jetzt schon enge und gute Zusammenarbeit mit dem gemeinsamen Newsletter verstärken können.

Wir haben im SOUG-Vorstand lange diskutiert, wie wir den von unseren Mitgliedern sehr geschätzten Newsletter optimieren können: Braucht es noch eine ausgedruckte Variante oder genügt es, einen elektronischen Newsletter zu versenden? Wollen wir weiterhin Mehrsprachigkeit oder wäre alles in Englisch optimal? Sollen wir die Fachbeiträge nur noch in einer Online-Plattform darstellen? Fragen, die sich viele Publikationsmedien aktuell auch stellen.

Eindeutige Antworten haben wir darauf nicht gefunden. Die künftig gemeinsame Produktion des SOUG-Newsletters mit der DOAG in gedruckter Form sowie als elektronischer Download ist zum Schluss der Abklärungen aus unserer Sicht die optimale Variante.

Für beide Organisationen erhalten wir so große Mehrwerte für unsere Mitglieder und Partner. Das Fach-Know-how beider geografischer Bereiche fließt in ein gemeinsames Publikationsmedium. Die SOUG-Mitglieder erhalten jetzt sechs Newsletter pro Jahr mit wesentlich mehr Artikeln pro Ausgabe. Für unsere Werbe- und Businesspartner ergibt sich ein wesentlich größerer Leserkreis und die zur Produktion notwendige Infrastruktur und die personellen Ressourcen werden nur einmal benötigt.

Wir hoffen, mit unserer Entscheidung, einen gemeinsamen Newsletter zu produzieren, haben wir Ihre Informationsbedürfnisse erkannt und können diese optimal erfüllen.

Ihr *P. Gübeli*



Peter Gübeli  
Präsident SOUG

## Oracle-Datenbanken 12c & Oracle Java SE 8 – Erleben Sie die Neuerungen!

Finden Sie Ihren passenden Kurs mit garantierter Durchführung:

- [www.digicomp.ch/java](http://www.digicomp.ch/java)
- [www.digicomp.ch/oracle](http://www.digicomp.ch/oracle)

**DIGICOMP**

☎ 0844 844 822, [info@digicomp.ch](mailto:info@digicomp.ch), [www.digicomp.ch](http://www.digicomp.ch)

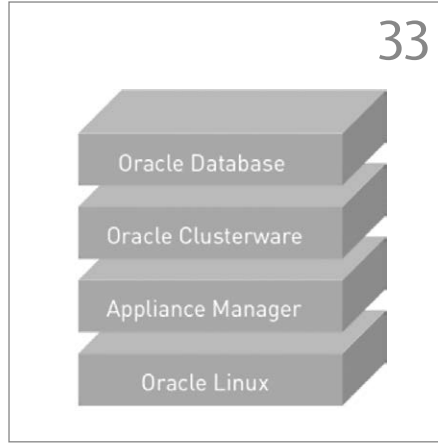
Digicomp ist «Oracle  
University Partner  
of the Year 2014 DACH»  
und einziger autorisierter  
Oracle Education  
Partner in der Schweiz





14

Nach erfolgreicher Einführung einer Exadata-Maschine ergeben sich im täglichen Betrieb interessante Weiterentwicklungen



33

Die für Datenbanken-Konsolidierung und Hochverfügbarkeit gedachte Oracle Database Appliance ist eine leistungsstarke Lösung



72

Die rund 2.000 Fachbesucher bekamen auf der 27. DOAG Konferenz + Ausstellung in Nürnberg wieder eine ganze Menge geboten

## Einleitung

- 3 Editorial
- 5 DOAG/SOUG Timeline
- 6 DOAG Vorstandssitzung am 12. und 13. Dezember 2014 in Berlin

- 26 SAP-Systeme mit Oracle-Technologien konsolidieren  
Jan Brosowski
- 33 Oracle Database Appliance – ein Deep Dive  
David Hueber

- 64 Data Guard 12c: „Zero-Data-Loss“ trotz großer Distanz der Standby-Datenbank  
Uwe Hesse
- 71 Tipps & Tricks aus Gerds Fundgrube Heute: Änderungen in Journal-Tabellen hervorheben  
Gerd Volberg

## Konsolidierung

- 7 „Die IT-Industrie in Europa ist stärker, als man annimmt ...“  
Interview mit Prof. Dieter Kempf
- 10 Was DBAs über virtualisierte Umgebungen wissen sollten  
Siegfried Langer
- 14 Aus der Exadata-Konsolidierung wird eine Oracle Engineered Architecture  
Christian Trieb
- 16 Datenbank-Konsolidierung mit Multitenant und In Memory  
Jens-Christian Pokolm
- 18 Richtigstellung zum Artikel von Andrew Lacy in der letzten Ausgabe
- 20 Pragmatischer Einstieg in Daten-Analyse und Konsolidierung: Hadoop meistert den Spagat zwischen Big Data und Kosteneffizienz  
Oliver Herzberg und Carsten Herbe

## Entwicklung

- 38 Inventur im ERP-System  
Christian Wille
- 42 Exception Handling in ADF  
Dr. Albert Angele

## Aktuell

- 68 Upgrade nach Oracle Database 12c – oder lieber noch warten?  
Mike Dietrich

## Datenbank

- 47 Database Resource Manager – Stand der Dinge  
Ulrike Schwinn
- 53 Faktenbasierte Indexierung – ein Erfahrungsbericht  
Lothar Flatz
- 56 Wir haben doch keine Zeit – Tipps zum Oracle-Text-Index in Webcenter Content  
Gunther Thielemann
- 60 Redo Logs – Informationen, so weit der Logminer reicht  
Thomas Klughardt

## DOAG intern

- 52 S3-Programm – Klappe, die erste  
Yasmin Misch
- 59 DOAG DevCamp – Software Upcycling
- 72 DOAG 2014 Konferenz + Ausstellung mit neuem Besucherrekord  
Mylène Diacquenod und Marina Fischer
- 73 Neue Mitglieder
- 74 Impressum
- 74 Termine
- 74 Inserentenverzeichnis



# DOAG/SOUG Timeline



## 7. November 2014

Das Organisationskomitee der JavaLand 2015 legt die Community-Aktivitäten für die Java-Konferenz am 24. und 25. März im Phantasialand in Brühl fest. Nach oder zwischen den Vorträgen können die Teilnehmer in einer Reihe von Workshops und Unplugged Sessions die neu gewonnenen Kenntnisse sofort aktivieren. Das Community-Hacking bietet unter anderem Coding Dojo, Architektur-Kata, Tinkerforge, Lambdaify Your Project, Open Space – Tic Tac Toe, Early Adopters Area, Java Innovation Lab und den Hackergarden. Für die Fitness gibt es ein JavaLand-Jogging und bei „Meet the JUGs“ kann man sich über die Aktivitäten der Java User Groups in seiner Region informieren.

## 11. November 2014

Für seine Eröffnungsk keynote zur Oracle CloudWorld in Frankfurt wählt Thomas Sattelberger, ehemaliger Vorstand Telekom AG, Continental AG und Lufthansa Passage, das Thema „Digitale Transformation: die Arbeit der Zukunft – die Zukunft der Arbeit“. Er weist darauf hin, dass Deutschland Gefahr läuft, die digitale Revolution zu verschlafen, gering zu schätzen oder politisch früh zu Tode zu regulieren. Mit scharfem Blick, unternehmerischem Mut und transformativer Kraft gilt es gegenzusteuern. Das Vormittagsprogramm der Veranstaltung stößt damit auf großes Interesse.

## 18. November 2014

Die 27. DOAG Konferenz + Ausstellung öffnet im Nürnberger Convention Center ihre Tore. Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG, begrüßt die zahlreichen Besucher. Im Anschluss daran geht Markus Flierl, Oracle Vice President Software Development, auf das Zusammenspiel zwischen Oracle-Database und der Oracle-Infrastruktur ein. Die Zuhörer erfahren aus erster Hand, wie zum Beispiel die neue Version von Solaris oder SPARC unter anderem höhere Leistung, Skalierbarkeit und Verfügbarkeit der Datenbank oder Oracle-Applikation erzielt.



Markus Flierl, Oracle Vice President Software Development und verantwortlich für das Solaris-Betriebssystem

Am Stand der SOUG gibt es feine Schöggli zum Kaffee, den man sich bei einem kleinen Schwatz mit dem SOUG-Vorstand und -Sekretariat gönnt. Der Schweizer-Abend am Eröffnungstag gehört zur festen Tradition der DOAG Konferenz, und hier wird diesmal eine Rekordbeteiligung erreicht. Der Bus, der in die Altstadt von Nürnberg fahren soll, ist bis auf den letzten Platz belegt, sodass die restlichen Teilnehmer mit einem PW den Weg zu den Markt- und Lebkuchen-Frauen unternehmen. Leider ist das Wetter nicht so gut. Bei leichtem Regen werden die zwei Gruppen durch die Altstadt geführt und hören viel Spannendes und Unterhaltsames zur Lebkuchenstadt. Beim anschließenden Nachtessen können sich alle mehr als genug verköstigen – der Wirt hat wohl eine Truppe Schwerarbeiter erwartet. Interessante Gespräche lassen die Zeit wieder wie im Fluge vergehen und einige sollen gar die erste Session am Mittwochfrüh verpasst haben ...



## 19. November 2014

Christian Trieb, Leiter des Competence Center Support der DOAG, stellt im Rahmen der DOAG 2014 Konferenz + Ausstellung die Ergebnisse der aktuellen Umfrage zur Qualität des Oracle-Supports vor. Fast jeder zweite Befragte ist demzufolge mit Qualität, Reaktionszeit und Prozessen unzufrieden. Lediglich das Serviceportal „My Oracle Support“ erhält von 43 Prozent der Befragten die Note „gut“. Die Ergebnisse kommen nicht überraschend, denn am grundsätzlichen Ergebnis des vorletzten Jahres hat sich nichts geändert. Im Detail beurteilen 46 Prozent der Befragten die Qualität des Supports als „schlecht“, nur 28 Prozent vergaben die Note „gut“. Ähnliche Ergebnisse erzielt die Umfrage bei den Reaktionszeiten (42 Prozent „schlecht“) und den dahinterliegenden Prozessen (45 Prozent „schlecht“). 57 Prozent der Befragten ziehen sogar den Support durch einen Drittanbieter in Erwägung. Eine neue Brisanz erhält das Thema aufgrund der von Oracle unbestätigten Informationen über eine mögliche Verlagerung des Oracle-Supports nach Rumänien. Kundennähe und Service-Qualität stehen auf dem Spiel.

## 20. November 2014

Fried Saacke, DOAG-Vorstand und Geschäftsführer, gibt einen neuen Besucherrekord für die DOAG 2014 Konferenz + Ausstellung bekannt. Unter dem Motto „Experience Passion“ kommen mehr als 2.000 erfahrene Oracle-Spezialisten aus aller Welt zusammen, um ihre Leidenschaft für den IT-Beruf zu teilen. Die mittlerweile größte Oracle-Konferenz in Europa ist die ideale Plattform für den Erfahrungsaustausch zwischen den Oracle-Anwendern. Rund 450 Fachvorträge vermitteln neueste Erkenntnisse zum erfolgreichen Einsatz der Oracle-Produkte; das Rahmenprogramm bietet viel Raum zum Networking. Es sind auch auffallend viele Oracle-Mitarbeiter vor Ort, die im Rahmen der Veranstaltung Kontakt zu ihren Kunden suchen. Wichtige Themen, die die Community gerade bewegen, sind „Cloud“, „Engineered Systems“ und „In Memory“.



Reger Austausch unter den Teilnehmern der DOAG 2014 Business Solutions Konferenz in Berlin

## 2. Dezember 2014

Das Leitungsmeeting der DOAG Business Solutions Community bereitet die DOAG 2015 Business Solutions Konferenz vor, die in diesem Jahr von Berlin in die Wissenschaftsstadt Darmstadt inmitten der Metropolregion Frankfurt/Rhein-Main umzieht und dort vom 9. bis 11. Juni stattfindet. Unter dem Motto „Digitale Transformation mit Oracle-basierten Unternehmens-Applikationen“ erhalten die Anwender sofort umsetzbares Expertenwissen und Brandaktuelles direkt aus den Oracle-Entwicklungslaboren. Die intensive Einbindung des Herstellers Oracle, der globalen Kooperationspartner Quest International Users Group und Oracle Applications Users Group (O AUG) sowie von Anwendergruppen aus dem benachbarten europäischen Ausland garantiert eine optimale Breite und Tiefe des Vortrags- und Ausstellungsangebots.

## 10. Dezember 2014

Die Vorstandssitzung der SOUG dreht sich vor allem um die drei Themen „Abgänge/Neuzugänge und Umverteilung einzelner Ämter im Vorstand“, „Newsletter-Beilage für die Kollegen aus dem

französischen Sprachgebiet“ und „Erste Planungsschritte für die Anlässe und Sitzungen 2015“. Da die Sitzung im Gasthof Bären stattfindet, wird anschließend auch passend zum dezemberlichen Dauerschmaus diniert.



## 12./13. Dezember 2014

Die Vorstandssitzung der DOAG stellt die Weichen für das Jahr 2015. Zum ersten Mal in der Geschichte der DOAG wird es vier mehrtägige Großveranstaltungen geben: DOAG 2015 Konferenz + Ausstellung, JavaLand 2015, DOAG 2015 Business Solutions Konferenz und – neu – die APEX connect. Somit kommt wieder ein Jahr mit großen Herausforderungen auf die Mitarbeiter(innen) der DOAG Dienstleistungen GmbH zu, die mit der Organisation der Veranstaltungen beauftragt ist.

## 12. Dezember 2014

Vorstand und Geschäftsstelle der DOAG treffen sich zum offiziellen Jahresabschluss. Der Vorstandsvorsitzende Dr. Dietmar Neugebauer sowie der DOAG-Vorstand und -Geschäftsführer Fried Saacke blicken auf ein erfolgreiches Jahr zurück.

## 31. Dezember 2014

DOAG und SOUG wünscht allen Mitgliedern und Interessenten ein gesundes und erfolgreiches neues Jahr!

## 25. Januar 2015

Das Team der DOAG-Geschäftsstelle trifft sich zum Kick-off-Meeting, um sich auf die anstehenden Projekte im Jahr 2015 einzustimmen. Mit Schwung und Elan sollen die vielen Herausforderungen angepackt werden, die das neue Jahr bringt.

# DOAG Vorstandssitzung im Dezember 2014 in Berlin

Hauptthemen der letzten Vorstandssitzung im Jahr 2014 waren die Planungen für das kommende Jahr. So stehen fünf große mehrtägige Konferenzen im Mittelpunkt. Daneben gibt es eine Reihe von eintägigen Veranstaltungen zu organisieren wie den Noon2Noon-Event in Nürnberg, die BI-Konferenz in München und den Exaday in Frankfurt.

Zum Status der aktuellen Finanzen gaben Finanzvorstand Urban Lankes für den Verein und Fried Saacke für die DOAG Dienstleistungen GmbH einen für das abgelaufene Jahr positiven Ausblick. Die genauen Zahlen werden im 1. Quartal 2015 veröffentlicht.

Als weiteren wichtigen Punkt der Vorstandssitzung präsentierte Myléne Diacquenod den Status zum Thema „Online Services der DOAG“. Schwerpunkte im Jahr 2015 werden die Themenseiten der einzelnen Communities sein sowie der komplette Bereich von MyDOAG als zentrale Drehscheibe des DOAG-Webauftritts.

Am zweiten Tag der Vorstandssitzung berichtete Dietmar Neugebauer über die internationalen Aktivitäten der DOAG. Es zeigt sich immer mehr, dass die DOAG international – vor allem auf europäischer Ebene – verstärkt mit anderen Usergruppen zusammenarbeiten muss. Dies betrifft einerseits den Erfahrungsaustausch durch die zuneh-

mende Anzahl von ausländischen Referenten auf unseren Konferenzen und andererseits auch die innerhalb von Oracle immer stärker werdende Verlagerung von Verantwortungen und Entscheidungen auf europäische Ebene. Die DOAG wird deshalb bereits im Rahmen der nächsten Jahreskonferenz eine Oracle-unabhängige Plattform für den Austausch von Usergruppen bereitstellen.

Zum Abschluss der Vorstandssitzung stellt Michael Paege seine Verantwortungsbereiche „Regionalgruppen“ und „Competence Center“ vor. Gerade die Veranstaltungen in den Regionalgruppen haben im Jahr 2014 wieder sehr guten Zuspruch erhalten und sind ein wichtiger Bestandteil der Arbeit in der DOAG.



# „Die IT-Industrie in Europa ist stärker, als man annimmt ...“

Deutschland soll in den nächsten Jahren zum digitalen Wachstumsland Nr. 1 werden. Wolfgang Taschner, Chefredakteur der DOAG/SOUG News, sprach darüber mit Prof. Dieter Kempf, Präsident des Branchenverbands BITKOM.



Wolfgang Taschner (links) im Gespräch mit Prof. Dieter Kempf

*Was sind die wichtigsten Ziele des BITKOM?*

**Prof. Kempf:** Als Branchenverband der Informations- und Kommunikationstechnologie steht für uns das Schaffen innovations- und investitionsfreundlicher Rahmenbedingungen in der Politik, aber auch in der Gesellschaft, an oberster Stelle. Darunter gliedern sich eine Vielzahl inhaltlicher Schwerpunkte auf. Aktuell sind dies der Ausbau einer Infrastruktur zur flächendeckenden Versorgung mit leistungsfähigen Breitbandanschlüssen und darauf aufbauend die Schaffung intelligenter Hochleistungsnetze für die Bereiche „Gesundheit“, „Bildung“, „eGovernment“, „Energie“ und „Verkehr“. Hinzu kommen Themen mit übergreifendem Charakter wie „Datensicherheit und Datenschutz“ sowie in die Zukunft gerichtete Aspekte wie „Industrie 4.0“ und die Gestaltung der Informationstechnologie

über unterschiedliche Industriezweige hinweg. Über allem steht natürlich eine gesamtgesellschaftliche Sichtweise. Wenn unsere Geschäftsmodelle auch morgen noch funktionieren sollen, müssen wir heute dafür sorgen, dass der Kunde sie akzeptiert.

*Wie setzen Sie diese Ziele um?*

**Prof. Kempf:** Zunächst stellt unsere hervorragend besetzte Geschäftsstelle eine gut funktionierende Infrastruktur bereit. Unser Hauptvorstand umfasst Repräsentanten aus rund hundert der wichtigsten ITK-Unternehmen in Deutschland. Hinzu kommen zahlreiche ehrenamtliche Mitarbeiter, ohne die eine Umsetzung in den mehr als einhundertfünfzig Arbeitskreisen nicht möglich wäre. Die Arbeit ist also auf viele Schultern verteilt.

*Wie schaffen Sie es, die gesamten Aktivitäten im Blick zu behalten?*

**Prof. Kempf:** Das funktioniert nur durch die intensive Zuarbeit aller Gremien, in denen die Arbeit gemacht wird. Darüber hinaus ist jedes unserer fünfzehn Präsidiumsmitglieder in der Regel für drei bis fünf Themen verantwortlich.

*Ihre Amtszeit endet bald und Sie dürfen laut Satzung nicht mehr wiedergewählt werden. Was haben Sie während Ihrer Amtszeit erreicht?*

**Prof. Kempf:** Verglichen mit anderen Branchenverbänden ist der BITKOM noch im Teenager-Alter. Dennoch habe ich vor fast vier Jahren von meinem Vorgänger einen hervorragend aufgestellten Verband übernommen. Die Wahrnehmung und Anerkennung in der Öffentlichkeit ist seitdem nochmals gestiegen, sodass wir zu Recht von uns sagen können, die Interessen der Branche nachhaltig zu vertreten. Unsere Stimme hat politisches Gewicht bekommen, weil wir mehrfach zeigen konnten, dass wir die industriespezifischen Interessen mit dem Gemeinwohl verknüpfen.

*Wie schätzen Sie das Wachstum der IT-Industrie in Europa ein?*

**Prof. Kempf:** Die IT-Industrie in Europa ist stärker, als man im Allgemeinen annimmt, da IT meist nur anhand von Consumer-Themen wie Suchmaschinen oder Kommunikationsplattformen wahrgenommen wird, die zumeist aus den USA kommen. Die wahre Stärke der europäischen IT-Industrie liegt im fertigungsnahen Bereich und die treibenden Länder sind Deutschland und Großbritannien, während die IT-Unternehmen in Frankreich etwas unter der gesamtwirtschaftlichen Situation leiden. Die Zukunft der europäischen IT-Industrie liegt bei den Embedded Systems, also überall dort, wo es in Richtung „Industrie 4.0“ beziehungsweise „Datensicherheits-Lösungen“ geht. Insbesondere deutsche Unternehmen haben hier sehr gute Chancen. Es gilt, diese zu nutzen und offensiv anzugehen.

*Der BITKOM hat Anfang dieses Jahres neunzig Maßnahmen für Wachstum, Infrastruktur und Sicherheit vorgestellt, damit Deutschland in den nächsten Jahren zum digitalen Wachstumsland Nr. 1 wird. Was sind die Kernpunkte dieses Konzepts?*

**Prof. Kempf:** Diese neunzig Maßnahmen gliedern sich in die drei Handlungsfelder „Digitale Wirtschaft“, „Digitale Infrastruktur und intelligente Netze“ und „Vertrauen und Sicherheit“, die natürlich nicht nur für Deutschland, sondern für die gesamte internationale Branche gelten. Wir wollen mit unseren konkreten Umsetzungsvorschlägen die Bundesregierung bei ihren ambitionierten Plänen für eine Digitale Agenda unterstützen.

*Wie muss ein IT-Unternehmen sich positionieren, um im globalen Markt erfolgreich zu sein?*

**Prof. Kempf:** Das Wichtigste ist, zu lernen, zu verstehen und zu akzeptieren, dass es in unserer Branche nahezu unmöglich sein wird, Dinge vollständig alleine tun zu können. Das Kunstwort „Kooperation“ gewinnt immer mehr an Bedeutung – Kooperation und Konkurrenz treten also in Beziehung zueinander. Zukünftige Lösungen müssen untereinander kompatibel und wie die berühmten Lego-Bausteine zusammensetzbar sein. Das bedeutet für viele Unternehmen einen Kulturwandel, denn sie haben jahrzehntelang gut von einer proprietären Aufstellung gelebt. Wir

müssen uns öffnen und flexibler werden. Zukunftstechnologien wie das Cloud-Computing können mit proprietären Ansätzen kaum funktionieren.

*Welche Anforderungen wird die klassische Industrie in den nächsten Jahren an die IT stellen?*

**Prof. Kempf:** Dazu gehört in erster Linie ein hohes Maß an Flexibilität. Die IT-Branche wird eine ähnliche Beziehung erleben,



#### Zur Person: Prof. Dieter Kempf

Im Jahr 2011 hat Dieter Kempf die Nachfolge von Prof. Dr. August-Wilhelm Scheer als Präsident des BITKOM angetreten. Zuvor engagierte er sich seit dem Jahr 2001 als Mitglied im Geschäftsführenden Präsidium des Branchenverbands als Schatzmeister.

Dieter Kempf begann seine Karriere als einer der ersten Mitarbeiter von McDonald's, wo er während seines Studiums der Betriebswirtschaftslehre an der Ludwig-Maximilians-Universität München für die Fast-Food-Kette auch im Leitungsteam der ersten deutschen Filiale in München-Giesing tätig war. Nach seinem Studium wechselte Dieter Kempf zur Wirtschaftsprüfungsgesellschaft Arthur Young (später: Ernst & Young). Dort spezialisierte er sich als Wirtschaftsprüfer im Bereich der Revision elektronischer Datenverarbeitung und wurde 1989 zum Partner (Geschäftsführer und Mitgesellschafter) berufen.

Prof. Dieter Kempf ist seit Juli 1996 Vorsitzender des Vorstands der DATEV eG. Zuvor verantwortete er als stellvertretender Vorsitzender des Vorstands der DATEV die Ressorts Produktentwicklung und Produktion. Er war im Juni 1991 als Mitglied des Vorstands in die DATEV eingetreten und verantwortete zunächst die Ressorts Produkt- und Software-Entwicklung.

Zusätzlich zu seiner Tätigkeit für die DATEV lehrt Kempf als Honorarprofessor für Betriebswirtschaftslehre an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Darüber hinaus setzte er sich bis zum Jahr 2012 auch als Vizepräsident der Steuerberaterkammer Nürnberg für den Nachwuchs des freien Berufsstands ein.



wie sie zwischen der Automobil-Industrie und ihren Zulieferern besteht, wo es mittlerweile eine sehr hohe Arbeitsteilung und Verzahnung gibt. Auch in der klassischen Industrie wird der Konsument zukünftig in die Produktgestaltung einbezogen werden, beispielsweise über soziale Medien.

*Wie wird der Job eines IT-Professionals in Deutschland in der Zukunft aussehen?*

**Prof. Kempf:** Ich rede häufig mit Studierenden genau über dieses Thema und gebe ihnen jedes Mal als wichtigsten Kernsatz mit: „Behaltet eure Neugierde und eure Bereitschaft, Neuem gegenüber offen zu sein“. Kombiniert mit einer profunden Informatik-Ausbildung wird so jemand immer eine interessante Arbeit finden.

*Wie lautet Ihre Empfehlung an Unternehmen hinsichtlich der Datenspeicherung in der Cloud?*

**Prof. Kempf:** Es gilt, die Daten so sicher wie möglich abzulegen. Dabei geht es primär weniger um die Datensicherheit als vielmehr darum, inwiefern ein Datenverlust das Geschäft gefährdet. Von daher spielt das Verhältnis zum Provider eine große Rolle und ebenso, welche Daten man ihm anvertraut. Insbesondere Private-Cloud-Strukturen werden dabei zum Zug kommen.

*Welche anderen Sicherheitsvorkehrungen sollten Unternehmen unbedingt beachten?*

**Prof. Kempf:** Das beginnt bei Viren-Prophylaxe und Firewall, führt über ein ausgefeiltes Zugangskonzept und endet bei der Aufklärung der Mitarbeiter über Risiko-Potenziale. Jedes Unternehmen braucht unabhängig von seiner Größe eine solche Sicherheitsstruktur, um daraus die richtigen Schutzmechanismen für die Daten abzuleiten.

*Wie beurteilen Sie den Markt für Standard-Applikationen, beispielsweise von SAP, Microsoft oder Oracle?*

**Prof. Kempf:** Ich glaube, dass es zukünftig sowohl einen Markt für Standard-Applikationen als auch einen für individuelle Soft-

ware-Entwicklung geben wird. Beide Richtungen werden jedoch intensiver zusammenwachsen, damit man fertige Bausteine mit individuellen Services besser verknüpfen kann. Standard-Software muss daher künftig noch offener sein.

*Sollten Unternehmen ihre IT-Strategie nach dem Muster „Alles aus einer Hand“ ausrichten oder halten Sie die „Best of Breed“-Lösung für besser?*

**Prof. Kempf:** Es wird auch in Zukunft für beide Strategien Märkte und Modelle geben. Wir entwickeln uns jedoch immer mehr von einem Anbieter- zu einem Kunden-Markt. Früher konnte der IT-Anbieter noch entscheiden, welche Lösung der Kunde bekommen soll; zukünftig wird das genau umgekehrt der Fall sein.

### Der BITKOM

Der Branchenverband BITKOM vertritt mehr als 2.200 Unternehmen der digitalen Wirtschaft, davon gut 1.400 Direktmitglieder. Sie erzielen mit 700.000 Beschäftigten jährlich Umsätze von 190 Milliarden Euro, darunter 50 Milliarden Euro Exporte. Zu den Mitgliedern zählen 1.000 Mittelständler, mehr als 200 Start-ups und nahezu alle Global Player. Sie bieten Software, IT-Services, Telekommunikations- oder Internetdienste an, stellen Hardware oder Consumer Electronics her, sind im Bereich der digitalen Medien oder der Netzwirtschaft tätig oder in anderer Weise Teil der digitalen Wirtschaft. Mehr als drei Viertel der Unternehmen haben ihren Hauptsitz in Deutschland, jeweils knapp zehn Prozent kommen aus sonstigen Ländern der EU und den USA, fünf Prozent aus anderen Regionen. BITKOM setzt sich insbesondere für eine innovative Wirtschaftspolitik, eine Modernisierung des Bildungssystems und eine zukunftsorientierte Netzpolitik ein.

Der Branchenverband bildet ein großes, leistungsfähiges Netzwerk und vereinigt die besten Köpfe und Unternehmen der digitalen Welt. BITKOM organisiert einen permanenten Austausch zwischen Fach- und Führungskräften und stellt seinen Mitgliedern Plattformen zur Kooperation untereinander und für den Kontakt mit wichtigen Kunden bereit.

Die Schaffung innovationsfreundlicher Rahmenbedingungen hat für BITKOM höchste Priorität. Bildung und Fachkräftenachwuchs, Netzpolitik, Datenschutz und -sicherheit, E-Government und E-Health, Mittelstand und Start-ups, Urheberrecht, Softwaretechnologien, Consumer Electronics, Klimaschutz und Nachhaltigkeit, Barrierefreiheit und altersgerechte Gestaltung sowie eine neue Telekommunikations- und Medienordnung sind Kern der politischen Agenda des BITKOM. Im Sinne der digitalen Konvergenz fördert BITKOM die Zusammenarbeit aller Unternehmen mit ITK-Bezug.

Weitere Informationen unter [www.bitkom.org](http://www.bitkom.org)



# Was DBAs über virtualisierte Umgebungen wissen sollten

Siegfried Langer, IBM Deutschland Research & Development GmbH

Server-Virtualisierung ist eine etablierte Methode, um Kosten zu sparen und Ressourcen flexibler zu nutzen. Dies bedingt aber, dass Hardware-Ressourcen gemeinsam genutzt werden und einzelne Server sich in diese „Gemeinschaft von Servern“ einfügen müssen.

Aus der Server-Virtualisierung ergeben sich neue Herausforderungen an das Kapazitäts- und Performance-Management. Das Tuning einzelner Anwendungen kann Auswirkungen auf andere Anwendungen in der virtuellen Umgebung haben und ein Systemadministrator muss sicherstellen, dass eine faire Zuteilung von Ressourcen erfolgt, gleichzeitig aber auch Service Level Agreements (SLA) eingehalten werden können. Dieser gesteigerten Komplexität stehen allerdings auch Vorteile gegenüber: Neben Kosteneinsparungen durch effizientere Nutzung der Infrastruktur wird der individuelle Wartungsaufwand für einzelne, jetzt virtuelle Server geringer und das Betriebsmanagement – wie Backup und Vorsorge für den Katastrophenfall – lässt sich einheitlicher gestalten.

Für den DBA ist es wichtig zu verstehen, dass Performance-Probleme aufgelöst werden sollten, anstatt sie mit mehr Ressourcen (Cores, Speicher) herunterzuspielen, was zu erheblichen Mehrkosten führen kann (etwa durch Software-Lizenzkosten). In einer virtuellen Umgebung kann „mehr“ (Aufwand) manchmal sogar „weniger“ (Performance) sein.

Einzelne physische Server müssen für Lastspitzen ausgelegt sein. Die erforderliche Leistung muss die erwartete Spitzenauslastung sowie zukünftige Wachstumsreserven berücksichtigen, auch wenn diese nur kurzzeitig auftreten. Aufgrund der relativ geringen Hardwarekosten und der Verfügbarkeit von Multi-Core-Servern stellt dies meist kein unmittelbares Problem dar.

Wenn man allerdings die Software-Lizenzkosten einbezieht, die sich typi-

scherweise an der Anzahl der Prozessorkerne (Cores) bemessen, ergibt sich ein anderes Bild. Die Konsolidierung vieler, teils nur wenig ausgelasteter Prozessoren auf einen hoch virtualisierten Server führt zu einer wesentlich besseren Nutzung der Ressourcen und ermöglicht erhebliche Einsparungen bei den Software-Lizenzkosten. Darüber hinaus ergeben sich teils erhebliche Einsparungspotenziale bei den operativen Kosten (Strom, Kühlung, Stellfläche, Netzwerk, Servicepersonal). Ein zentralisiertes Management reduziert den Verwaltungsaufwand und erlaubt zentralisierte Datensicherung, bessere Vorsorge für den Katastrophenfall, Hochverfügbarkeit sowie die Nutzung von Cloud-Konzepten mit hoher Flexibilität und schneller Aktivierung neuer Server. Virtualisierung ist ein wich-

## Using the Oracle Performance Method

Performance tuning using the Oracle performance method is driven by identifying and eliminating bottlenecks in the database, and by developing efficient SQL statements. Database tuning is performed in two phases: proactively and reactively.

- In the proactive tuning phase, you must perform tuning tasks as part of your daily database maintenance routine, such as reviewing ADDM analysis and findings, monitoring the real-time performance of the database, and responding to alerts.
- In the reactive tuning phase, you must respond to issues reported by users, such as performance problems that may occur for only a short duration of time, or performance degradation to the database over a period of time.

SQL tuning is an iterative process to identify, tune, and improve the efficiency of high-load SQL statements.



Abbildung 1: Auszug aus dem Oracle Tuning Guide

tiger erster Schritt, um IT-Ressourcen für Cloud-Services nutzen zu können. Nur so kann ein hochflexibler und kosteneffektiver Betrieb gewährleistet werden.

### Performance-Management

Das Tuning eines Systems sollte von gemessenen Daten (Baseline) ausgehen und als sich ständig wiederholender Kreislauf von „Messen“ – „Evaluieren“ – „Verbessern (Verändern)“ – „Messen“ betrachtet werden. Veränderungen sollten sich auf einen oder wenige Parameter beziehen, da verschiedene Tuningmaßnahmen sich gegenseitig beeinflussen und sogar kompensieren können (siehe Abbildung 1, Quelle „[http://docs.oracle.com/cd/E24628\\_01/server.121/e17635/tdppt\\_method.htm#TDPPT006](http://docs.oracle.com/cd/E24628_01/server.121/e17635/tdppt_method.htm#TDPPT006)“).

Endbenutzer beurteilen den Durchsatz typischerweise über die beobachtete Antwortzeit. Hier ist zu beachten, dass die Datenbank-Zeit nur einen Teil der Antwortzeit ausmacht. Die Ursache für solche Probleme kann auch in der Anwendung oder im Netzwerk liegen.

Häufig ist zu beobachten, dass Performance-Probleme durch mehr Ressourcen adressiert werden. Natürlich lässt sich eine ineffektive Anwendung häufig durch mehr Prozessoren und/oder Speicher „erschlagen“, aber es werden in diesem Fall nur die Symptome beseitigt, die Ursache bleibt bestehen. Eine solche Vorgehensweise kann schnell zu einer Kostenexplosion und möglicherweise zu komplexen Folgefehlern führen.

### Virtualisierte Umgebungen

Virtualisierte Server teilen sich physische Ressourcen. Dies hilft bei Lastspitzen, es ist jedoch kein Rezept, um Performance-Engpässe zu beseitigen. Da nun mehrere Server um die physischen Ressourcen konkurrieren, kann die Suche nach solchen Flaschenhälsen wesentlich komplizierter werden, insbesondere wenn einzelne Server nicht optimal konfiguriert sind. Es gelten folgende Grundsätze:

Definiere nicht mehr virtuelle CPUs für einen Linux-Gast als nötig:

- Die Nutzung mehrerer Prozessoren benötigt Software-Locks, sodass Daten oder Kontrollblöcke nur von einem Prozessor zu einer Zeit geändert werden können.
- Linux nutzt ein globales Lock. Wenn das Lock gehalten wird und ein anderer Prozessor es benötigt, muss er warten.
- Die Zahl der virtuellen Prozessoren sollte nach dem Bedarf gesetzt werden und nicht einfach der Anzahl der realen Prozessoren entsprechen.
- Vorsicht beim Klonen: Einige Linux-Gäste brauchen mehr virtuelle CPUs als andere, etwa Oracle-Datenbankservers.

Definiere den (virtuellen) Speicher für Linux nicht größer als nötig:

- Exzessive virtuelle Speichergrößen haben eine negative Auswirkung auf die Performance.
- Linux nutzt freien Speicher für das Caching von Daten. Für gemeinsam genutzte (shared) Ressourcen hat dies negative Auswirkungen.



Specialized Oracle Database



Datenbanken mit iQ



# Zeit ist kostbar!

## Lösungen finden Sie bei uns – Ihrem RAC-Spezialisten.

[www.muniqsoft.de/rac12c](http://www.muniqsoft.de/rac12c)

- Reduziere die Größe des Linux-Gastes, bis er beginnt, Speicher auszulagern (Swap).
- Benutze virtuelle Plattenspeicher (VDISK) für Swap, wenn genügend realer Speicher verfügbar ist.
- Vergleiche die Linux-Speichernutzung mit den im Hypervisor definierten Größen des Gastes.

**Oracle-Datenbanken auf VMware**

VMware gibt Empfehlungen im „Oracle Databases on VMware Best Practices Guide“ (siehe „[http://www.vmware.com/files/pdf/solutions/oracle/Oracle\\_Databases\\_VMware\\_Best\\_Practices\\_Guide.pdf](http://www.vmware.com/files/pdf/solutions/oracle/Oracle_Databases_VMware_Best_Practices_Guide.pdf)“). Diese haben durchaus allgemeingültigen Charakter:

- *Definiere so wenig virtuelle Prozessoren (vCPUs) wie möglich*  
 Sofern das Monitoring der aktuellen Arbeitslast keine Verbesserung des Durchsatzes der Oracle-Datenbank durch mehr virtuelle Prozessoren zeigt, führen die zusätzlichen vCPUs zu Engpässen im Scheduler und können die Gesamt-Performance des virtualisierten Servers beeinträchtigen.
- *Speicher-Reservierungen sollten gleich der Oracle-SGA-Größe gesetzt sein*  
 Der reservierte Speicher sollte groß genug sein, um Speicherauslagerungen (kernel swapping) zwischen ESX und den Gastbetriebssystemen zu vermeiden.
- *Nutze Oracle Automatic Storage Management (ASM)*  
 ASM bietet integriertes Cluster File System und Platten (Volume) Management für Oracle-Datensätze. Es vereinfacht das Anlegen von Datensätzen und ermöglicht einen Daten-Durchsatz, der nahe an die Roh-Datenrate der Platteneinheiten herankommt.
- *Folge den „Best Practices“-Empfehlungen der Plattenhersteller beim Anlegen von Oracle-Datenbanken*  
 ASM ist nicht in der Lage, die optimale Platzierung der Daten oder die LUN-Auswahl für das verwendete Plattenspeichersystem zu bestimmen. Daher ist es kein Ersatz für die enge Abstimmung zwischen Plattenspeicher- und Datenbank-Administratoren (siehe Abbildung 2).
- *ASM-Plattenspeicher-Gruppen (disk groups) sollten gleiche Plattentypen mit gleicher Geometrie beinhalten*  
 Es sollten mehrere ASM-Disk-Gruppen basierend auf den I/O-Charakteristika

angelegt werden. Minimum sind zwei ASM-Disk-Gruppen, eine für Log-Files, die sequenzieller Natur sind, und eine andere für Daten, die von Natur aus wahllos (random) sind. Für hohe Durchsatzraten wird empfohlen, mehrere parallele Datenpfade zu den Datenplatten zu definieren beziehungsweise die Daten auf mehrere ASM-Gruppen zu verteilen.

**Oracle-Datenbank-Konsolidierung auf dem IBM System z**

„IBM System z“ heißt die Produktfamilie für den IBM-Mainframe. Neben den traditionellen Mainframe-Betriebssystemen wie z/OS oder z/VSE ist auch Linux unterstützt und nutzt spezielle System-z-Prozessoren, die Integrated Facility for Linux (IFL). Die System-z-Architektur unterstützt zwei Virtualisierungsebenen: Das physische System kann in bis zu sechzig logische Partitionen (LPARs) aufgeteilt sein, wobei Prozessoren mehrfach genutzt werden können (shared). Diese erste Ebene bietet eine sehr hohe Isolation zwischen den Partitionen, die nach Common Criteria „EAL5“ zertifiziert ist, was als äquivalent zu physisch getrennten Systemen gilt. Dadurch ist es möglich, Produktions-, Test- oder Entwicklungssysteme gleichzeitig auf dem gleichen Rechner zu betreiben ohne

das Risiko, dass Abstürze im Testbetrieb zu Ausfällen des Produktionssystems führen. Eine weitere, wesentlich granularere und flexiblere Virtualisierungsebene bietet der Hypervisor z/VM.

**Der IBM System z Hypervisor z/VM**

„Linux on System z“ kann sowohl „bare metal“ im LPAR laufen als auch unter z/VM. Als Besonderheit kann z/VM auch die traditionellen Betriebssysteme virtualisieren. Es ist sogar möglich, den z/VM Hypervisor unter z/VM zu betreiben, was beispielsweise für Schulungszwecke gerne genutzt wird. z/VM bietet höchste Skalierbarkeit für eine virtuelle Serverumgebung durch die Kombination von virtuellen und realen Ressourcen für jede virtuelle Maschine.

Der Hypervisor z/VM erlaubt es, den einzelnen Gastsystemen mehr Speicher zuzuweisen als physisch vorhanden (memory overcommitment). Dies ist eine sehr praktische Funktion bei vielen Gästen, die nur geringe oder seltene Anforderungen stellen (etwa Test- oder Entwicklungssysteme), da dieser Speicher den aktiven Prozessen zur Verfügung gestellt werden kann.

Überdimensionierte Gäste binden wertvolle Ressourcen, die der Hypervisor im Gesamtsystem mühsam suchen müsste. Außerdem ist die Zeit, die für das Verschie-

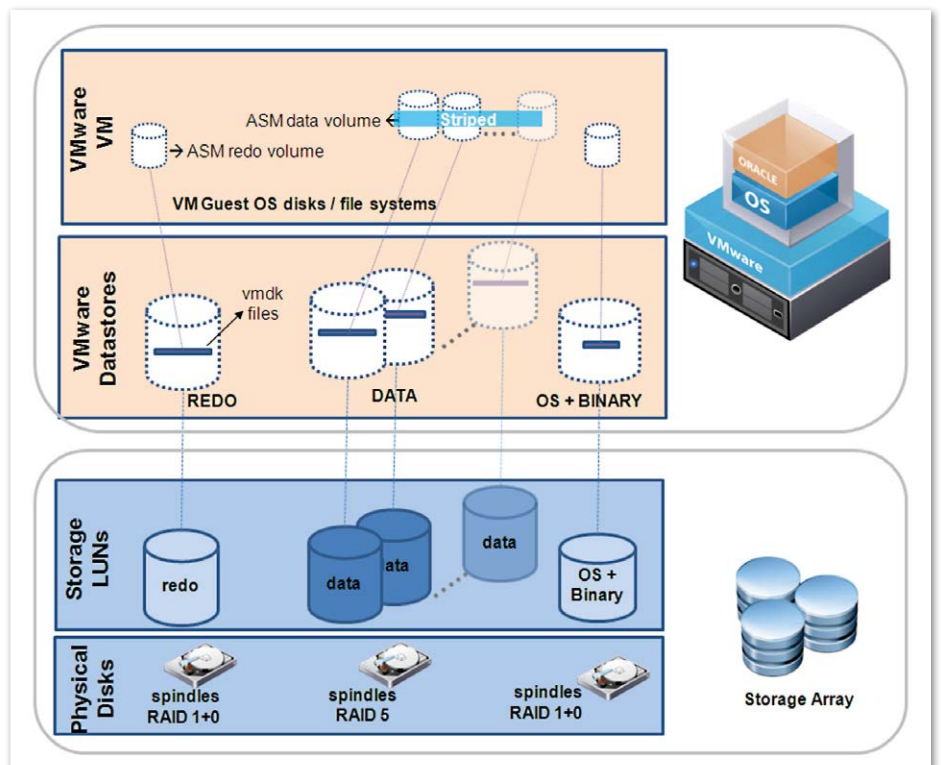


Abbildung 2: Speicher-Layout-Beispiel für OLTP-Datenbank mit VMware



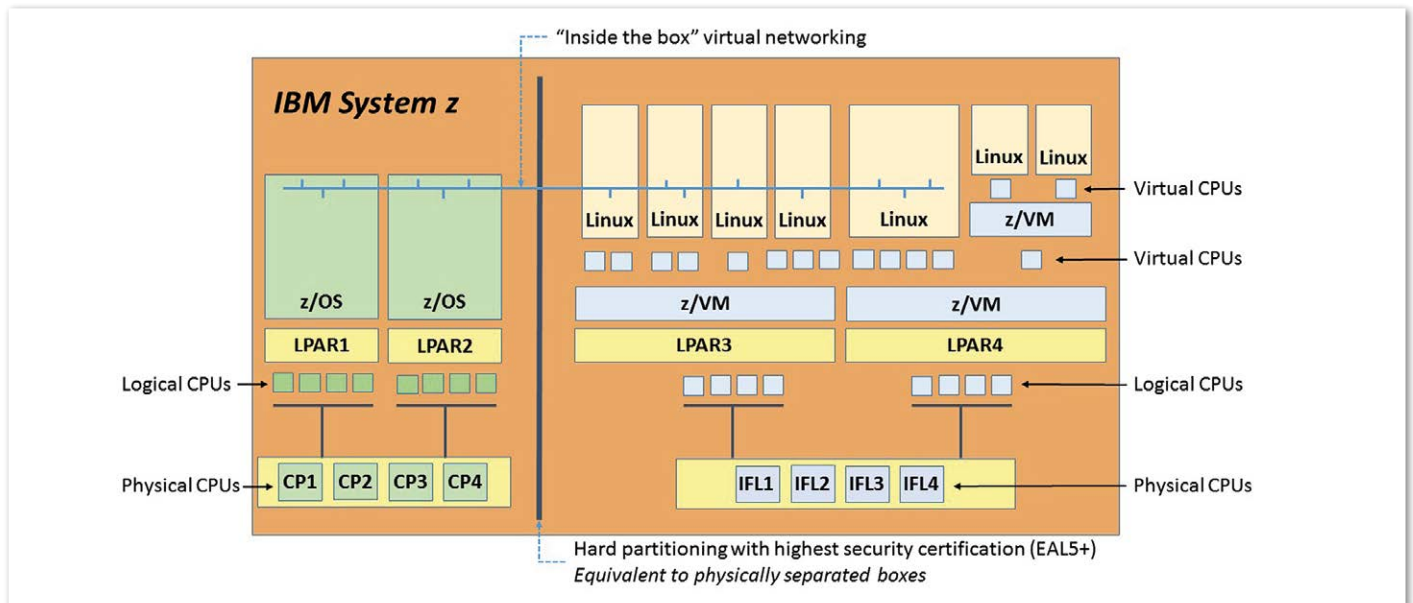


Abbildung 3: Extreme Virtualisierung mit z/VM

ben von Gigabyte-großen Speicherinhalten notwendig ist, unter Durchsatzgesichtspunkten nicht zu vernachlässigen. Für Linux-Gäste gilt daher, dass diese nur so groß dimensioniert werden sollten, wie für eine gute Funktionalität notwendig ist.

Zusätzliches Caching im Linux-Gast, insbesondere I/O-Caching mit laufenden Updates, bindet wertvollen Speicher, der anderen Prozessen nicht zur Verfügung steht. Da der Hypervisor in erster Linie nach dem „least recently used“-Algorithmus vorgeht, wird dieser Cache-Speicher nicht angefasst. Die empfohlene Vorgehensweise ist hier, Direct-I/O zu verwenden und die Linux-Gäste speicherseitig so zu dimensionieren, dass das individuelle Linux im Normalbetrieb gerade noch nicht auf den externen Speicher auslagert (swapped). Dies stellt sicher, dass z/VM das Gesamtsystem optimal mit Ressourcen versorgen kann. Für die Speicher-Dimensionierung hat sich die Faustformel „Startgröße = SGA + PGA + Linux + ASM“ für die virtuelle Linux-Server-Speichergröße bewährt:

- Speicher für Oracle SGA und PGA: laut DBA-Abschätzung
- Speicher für Linux Kernel: 512 MB
- Speicher für Oracle ASM: 256 MB bis 512 MB (falls ASM verwendet wird)

Das Verhältnis zwischen virtueller Speichergröße (Summe der definierten Spei-

chergrößen aller virtuellen Server des Hypervisor) und realem Speicher, also „Virtual:Real“, sollte kleiner als „3:1“ sein. Dieser Wert ist durchaus praxisrelevant für Test- und Entwicklungsumgebungen. Für Oracle-Datenbanken im Produktionsbetrieb hat sich ein Startpunkt von „1,5:1“ als brauchbarer Kompromiss erwiesen. Für besonders performancekritische Produktionsanwendungen kann es sinnvoll sein, das Verhältnis bis auf „1:1“ zu ändern.

Ähnliche Überlegungen gelten auch für die Zuordnung physischer und virtueller Prozessoren (CPU). Auch hier führt eine Überdimensionierung zu mehr Verwaltungsaufwand und damit Overhead im Hypervisor. Zusätzlich besteht das Risiko, dass sehr CPU-aktive Prozesse das Gesamtsystem dominieren und andere Gäste nur noch unzureichend Service geben können, da sie nicht mehr die benötigten Prozessor-Zeitscheiben zugewiesen bekommen. Die vorab zitierten Empfehlungen, bezogen auf virtuelle CPU, Plattenspeicher und ASM, die für Oracle-Datenbanken auf VMware gegeben wurden, gelten prinzipiell auch für z/VM.

### Automatisierung des Resource Managements

Die Größenbestimmung von Linux-Gästen kann ein komplexes Unterfangen sein, insbesondere in einem dynamischen Umfeld mit wechselnden Anforderungen und schnell wachsenden Anwendungen. Zu groß dimensionierte Linux-Gäste kosten zusätzlichen Management-Aufwand im Hypervisor, zu klein dimensionierte Gäste führen zu Performance-Problemen, insbesondere in Auslastungsspitzen.

Es besteht die Möglichkeit, das Management der Ressourcen zu automatisieren. Basierend auf den Anforderungen des Gastes kann das System CPUs und Speicher nach vordefinierten Regeln hinzufügen oder entfernen. Diese Funktion wird durch den Linux „cpuplugd daemon“ (auch „hotplug daemon“ genannt) zur Verfügung gestellt und ist für Linux on System z ab SLES 11 SP2 oder RHEL 6.2 verfügbar (siehe „[http://www-01.ibm.com/support/knowledgecenter/linuxonibm/iaag/10cpup00\\_2012.htm?cp=linuxonibm%2F0-4-3-1-2](http://www-01.ibm.com/support/knowledgecenter/linuxonibm/iaag/10cpup00_2012.htm?cp=linuxonibm%2F0-4-3-1-2)“).

### Fazit

Die Befolgung der genannten Hinweise und Empfehlungen allein garantiert noch nicht, dass eine virtualisierte Umgebung alle Anforderungen der Benutzer erfüllt, sie macht es jedoch einfacher, Ursachen zu ergründen und Abhilfe zu schaffen. Zusammenfassend soll an die Basisregeln des Performance-Managements erinnert werden:

- Etabliere ein permanentes Monitoring

- Sammle Systemdaten als Basisbewertung für gute Performance
  - Implementiere einen Change-Management-Prozess
  - Führe zu einer Zeit so wenige Änderungen wie möglich durch
  - Performance ist oft nur so gut wie das schwächste Glied
  - Die Beseitigung eines Flaschenhalses führt zu weiteren neuen Flaschenhälsen
- Erwarte Veränderungen an anderer Stelle, wenn eine Ressource verändert wird

Grundsätzlich gilt, dass die Ursachenforschung bei Durchsatzproblemen am Anfang stehen sollte. Das beste Tuning von System-Ressourcen kann Probleme der Anwendung oder der Netzwerk-Anbindung nicht beseitigen.



Siegfried Langer  
siegfried.langer@de.ibm.com



# Aus der Exadata-Konsolidierung wird eine Oracle Engineered Architecture

Christian Trieb, Paragon Data GmbH

Nach einer mehr oder weniger längeren Zeitspanne sind viele Datenbanken auf einer Exadata-Datenbankmaschine konsolidiert, das System ist ausgelastet und es gibt keine Datenbanken mehr, die auf die Exadata konsolidiert werden sollen. Letzteres ist sicher selten, kann aber (theoretisch) vorkommen. Dieser Artikel betrachtet den Betrieb einer Exadata-Maschine und die aus den anstehenden Anforderungen resultierenden Weiterentwicklungen.

## Betrieb

Im Rechenzentrumsbetrieb stellt sich eine Exadata auf den ersten Blick wie ein normales Server-Rack dar. Allerdings merkt man schon in der Planung eines Exadata-Betriebs schnell, dass doch einiges zu beachten ist. So muss bei der Integration und dem Betrieb einer Exadata in einem Rechenzentrum die Vorgehensweise gut geplant sein. Mehrere Abteilungen (Betriebssystem-, Storage-, Netzwerk- und Datenbank-Administration) müssen diese gemeinsam sicherstellen. Das klassische Denken in Silos muss also aufgebrochen werden. Nur

so lassen sich Synergie-Effekte nutzen. Im Unternehmen des Autors wurde dies unter der Leitung der Datenbankadministratoren gut umgesetzt. Die Verantwortung für die komplette Maschine liegt daher bei den DBAs. Sie wurden weitergebildet und können selbstständig die meisten Aufgaben lösen. Bei komplexeren Fragestellungen und Detailproblemen werden die Kollegen der anderen Abteilungen hinzugezogen.

Für den Betrieb einer Exadata-Maschine kommen die gleichen Methoden und Werkzeuge wie für den Betrieb von anderen Servern und Datenbanken zum Einsatz.

Die Maschine wird regelmäßig gesichert und mit dem Tool „Nagios“ überwacht. Datenbanken werden mit RMAN gesichert, auch mit „Nagios“ überwacht und mit Oracle Enterprise Manager Cloud Control administriert. Das Patchen einer Exadata-Maschine funktioniert mithilfe des Oracle Platinum Supports und einer sehr langfristigen detaillierten Planung relativ gut.

## Anforderungen

Im Laufe des Betriebs der Exadata-Maschine (X3-2 Quarter Rack) stellte sich nach Integration von immer mehr Datenbanken (mehr

als in der ursprünglichen Planung vorge-  
sehen) heraus, dass die Größe des Haupt-  
speichers ein limitierender Faktor war. Zu-  
dem mussten die Anforderungen an eine  
Hochverfügbarkeitslösung für die wichtigs-  
ten Datenbanken umgesetzt werden. Hinzu  
kam die Implementierung der Nutzung von  
speziellen Exadata-Eigenschaften wie Hyb-  
rid Columnar Compression (HCC).

Nach etwa eineinhalb Jahren Betrieb  
wurde ein steigender Plattenplatzbedarf  
für die Zukunft prognostiziert. Diesen Be-  
darf können die Platten in den drei vor-  
handenen Exadata Storage Servern nicht  
abdecken. Für die Migration der Datenban-  
ken von der Version 11g nach 12c wird zum  
Testen der Applikationen mit der neuen  
Version Platz benötigt, da es erforderlich  
ist, für eine Übergangszeit beide Daten-  
bank-Versionen parallel zu betreiben.

### Lösung

Die Anforderung nach mehr Hauptspeicher  
konnte durch dessen Verdoppeln relativ  
einfach erfüllt werden. Jeder der beiden Ex-  
adata-Datenbank-Server verfügt nun über  
512 GB RAM. Um die Hochverfügbarkeits-  
anforderungen sicherzustellen, wurden die  
wichtigsten Datenbanken mit Oracle Data  
Guard abgesichert. Da keine weitere Exada-  
ta-Maschine vorhanden ist, ist die Standby-  
Datenbank als konventionelles Zwei-Kno-  
ten-RAC aufgebaut. Als Platten-Subsystem  
dient ein NetApp 6080 Metro Cluster.

Um die Performance einiger Abfragen zu  
verbessern und gleichzeitig Plattenplatz ein-  
zusparsen, kommt sukzessive immer stärker  
Hybrid Columnar Compression zum Einsatz.  
Diese führte zu den gewünschten Erfolgen,  
hatte aber im Zusammenspiel mit Data Gu-  
ard zur Folge, dass HCC nicht in den durch  
Data Guard abgesicherten Datenbanken  
verwendet werden konnte, da auf der sekun-  
dären Datenbank kein HCC zur Verfügung

stand. Dies und der Umstand, dass der Plat-  
tenplatz in Zukunft knapp zu werden droht,  
führte zur Entscheidung, auch aufgrund des  
Preises die Oracle-Speicherlösung ZS3-2 zu  
erwerben. Damit war es möglich, dass alle  
gewünschten Test- und Entwicklungs-Da-  
tenbanken sowie die Standby-Datenbanken  
ihre Daten auf diesem Speichersystem ab-  
legen und somit auch HCC nutzen können.  
Somit können die Datenbank-Versionen 11g  
und 12c auch gleichzeitig betrieben werden.  
Die Standby-Datenbanken liegen auf einem  
Non-Exadata Zwei-Knoten-RAC und sind an  
die ZS3 über Fibre Channel angeschlossen.  
Die Test- und Entwicklungs-Datenbanken  
werden zwar weiterhin auf der Exadata be-  
trieben, nutzen aber den Speicherplatten-  
platz auf der ZS3. Die Nutzung der Exadata  
und des ZS3-Storage ermöglicht in Zukunft  
auch intelligentes Lifecycle-Management,  
möglicherweise auch unter Zuhilfenahme  
der Oracle-Heat-Map-Funktionalität.  
Somit ist aus der Exadata eine Engineered  
Architecture entstanden – bestehend aus ei-  
ner Exadata und dem ZS3-Storage.

### Fazit

Die Engineered Architecture erlaubt es  
nun, zu einem sehr guten Preis/Leistungs-  
Verhältnis die vorhandenen Ressourcen  
optimal auszunutzen und Platz für zukünf-  
tige Entwicklungen zu haben. So sind jetzt  
die wichtigsten Datenbanken hochverfü-  
gbar abgesichert. Die Entwicklungs-Daten-  
banken sind in der Lage, HCC zu benutzen,  
belasten aber nicht den Exadata-Storage,  
und auf der Exadata ist Platz, um die nächs-  
ten Schritte anzugehen, wie die Migration  
der Datenbanken von 11g nach 12c. Inso-  
fern ist diese Engineered Architecture eine  
sehr gute Möglichkeit, die Anforderung an  
Konsolidierung abzudecken und darüber  
hinaus kostenbewusst zu agieren.

### Technische Rahmenbedin- gungen

- Oracle Exadata X3-2 Quarter Rack
- 2 Linux DB-RAC-Knoten mit je  
512 GB RAM
- 3 Linux-Storage-Server mit 12  
Platten mit 3 TB
- ZS3-2-Cluster mit 2 x Sata Shelves  
mit jeweils 20 Platten à 900 GB



Christian Trieb  
c.trieb@paragon-data.de

## Sparen Sie Zeit, Geld und Nerven.



### Effizient und preiswert: DBConcepts.

Wir unterstützen Sie remote beim  
Betrieb von Oracle Datenbanken.

SLA ab 10hx5 bis 24hx7 inklusive

- proaktiver Überwachung
- rascher Reaktionszeit
- periodische Health Checks
- Backup und Recovery Tests



Die Oracle Experten

[www.dbconcepts.at](http://www.dbconcepts.at)  
Tel.: +43 1 890 89 990  
[office@dbconcepts.at](mailto:office@dbconcepts.at)

**ORACLE** Platinum  
Partner

# Datenbank-Konsolidierung mit Multitenant und In Memory

Jens-Christian Pokolm, Postbank Systems AG

Bei der Konsolidierung von Datenbanken auf Servern kommt es immer wieder zum Problem mit der Menge an Real-Time-Prozessen und einer steigender Anzahl an Instanzen. Mit Nutzung der In-Memory-Option wächst der Wunsch, die im Verhältnis zu Festplatten hochpreisige Ressource „Hauptspeicher“ möglichst permanent auszunutzen. Wenn beide Anforderungen aufeinandertreffen, helfen nachfolgende Informationen ein Stück weiter.

Heute verfügbare Datenbank-Server haben oft eine hohe Anzahl an Prozessoren. Das Unternehmen des Autors setzt bei der Datenbank-Konsolidierung in der Regel Systeme mit 16 oder 32 Cores für ihre Belange ein – grundsätzlich als Stretched-RAC. Jede Datenbank, das darunterliegende ASM sowie die Clusterware (und gegebenenfalls weitere Überwachungsmonitore und Tools) benötigen auf den Servern dazu Real-Time-Prozesse (*siehe Abbildung 1*).

Auf extrem vielen Prozessoren stehen inzwischen HyperThreads zur Verfügung und präsentieren sich im Betriebssystem und teilweise auch in der Datenbank als angeblich vollständige CPUs. Dies ist aus Sicht der Real-Time-Prozesse jedoch nicht korrekt. Letztendlich stehen ausschließlich die physischen Cores für diese Prozessklasse zur Verfügung. Auf der Datenbank-Ebene sprechen wir an dieser Stelle primär über die LMS-Prozesse auf dem System. Als Daumenregel sollten auf einem Server niemals mehr Real-Time-Prozesse als physische Cores ihren Dienst verrichten. Damit war der massiven Konsolidierung oftmals ein frühes Ende beschert – obwohl eigentlich zumeist noch ausreichend freie Prozessorleistung zur Verfügung stand (*siehe auch MOS-Notes 558185.1 und 1392248.1*).

## Multitenant

Die Einführung der Multitenant-Option (Mandantenfähigkeit) – zumeist einfach „Pluggable-Database“ genannten – hat diese Grenzen ein Stück weit verschoben.

Deren Konstrukt besteht darin, systeminterne Objekte sowie Prozesse wie das Data Dictionary, die PL-SQL-Objekte und die Hintergrund-Prozesse in einer globalen (geteilten) Umgebung zur Verfügung zu stellen – ähnlich einem HyperVisor bei virtuellen Servern. Diese Umgebung nennt Oracle den „Root-Container“ (CDB) und die darunter laufenden Datenbanken dann „Pluggable Database“ (PDB).

In den Datafiles der PDB werden nur noch die Deltas zur CDB gespeichert – dies gilt für die PDB-eigenen vollständigen Objekte sowie auch für die zusätzlichen Daten im Data Dictionary. Aus Sicht der Konsolidierung eine sehr interessante Funktion – denn nun können sich mehrere PDBs einen Großteil der initialen Datenbestände einer Datenbank quasi teilen. Dies gilt neben den klassischen Tablespace (TEMP / SYSTEM / SYSAUX / UNDO etc.) auch für die SGA und die Basis-Prozesse der Datenbank. Dadurch sind die Konsolidierungssysteme sowohl im Storage als auch bei den Real-Time-Prozessen entlastet, denn nun teilen sich mehrere Instanzen auf einem Knoten einen LMS.

In dem Konstrukt gibt es gute Möglichkeiten, um die Ressourcen von PDBs gegeneinander zu schützen – die Datenobjekte jeder PDB sind immer vollumfänglich gegeneinander gesichert. Vorhandene Ressourcen lassen sich weit besser managen als dies bei eigenständigen (autarken) Datenbanken der Fall war – denn hier bestand nur die Option mittels „CPU\_COUNT“ einen Maximalwert zu definieren.

Sobald ein CPU-Over-Provisioning-Ansatz gefahren wurde, war das Ganze dann ein Stückweit Makulatur.

Mittels Ressourcen-Manager können den PDBs nun untereinander garantierte sowie maximal mögliche Ressourcen definiert werden. Der zugrunde liegenden CDB wird wie bisher mittels „CPU\_COUNT“-Limitierungen nach oben gesetzt. Die feingranulare Abstufung von Benutzern innerhalb einer PDB ist weiterhin über Consumer-Groups möglich.

## In Memory

Die mit der Version 12.1.0.2 eingeführte In-Memory-Option ist abweichend von den Buffer-Konzepten kein „Pool“ sondern ein In-Memory-Store. Beim Pool werden durch LRU-Mechanismen Objekte auch wieder verdrängt, sofern diese nicht mehr genutzt werden oder der Platz im Speicher nicht mehr ausreicht. Die In-Memory-Option hingegen gibt einmal belegten Speicher nur auf explizite Anweisung wieder frei. Im Auslieferungszustand ist die Option (endlich einmal) deaktiviert (*siehe Listing 1*).

Zum Einschalten der Option ist mittels „SQL> alter system set inmemory\_size=16G scope spfile sid=“\*“ die „inmemory\_size“ zu setzen. Danach muss die Datenbank neu gestartet werden, um die Option zu aktivieren. Zur Kontrolle kann man die Werte dann natürlich auch in der „v\$sga“-View finden. Zur Ermittlung, welche Objekte im Speicher residieren ist, die Abfrage „select owner,segment\_name,populate\_status from v\$im\_segments;“ nützlich.



## Standard Setup für alle Anwendungen: eGrid 2.0 → RAC

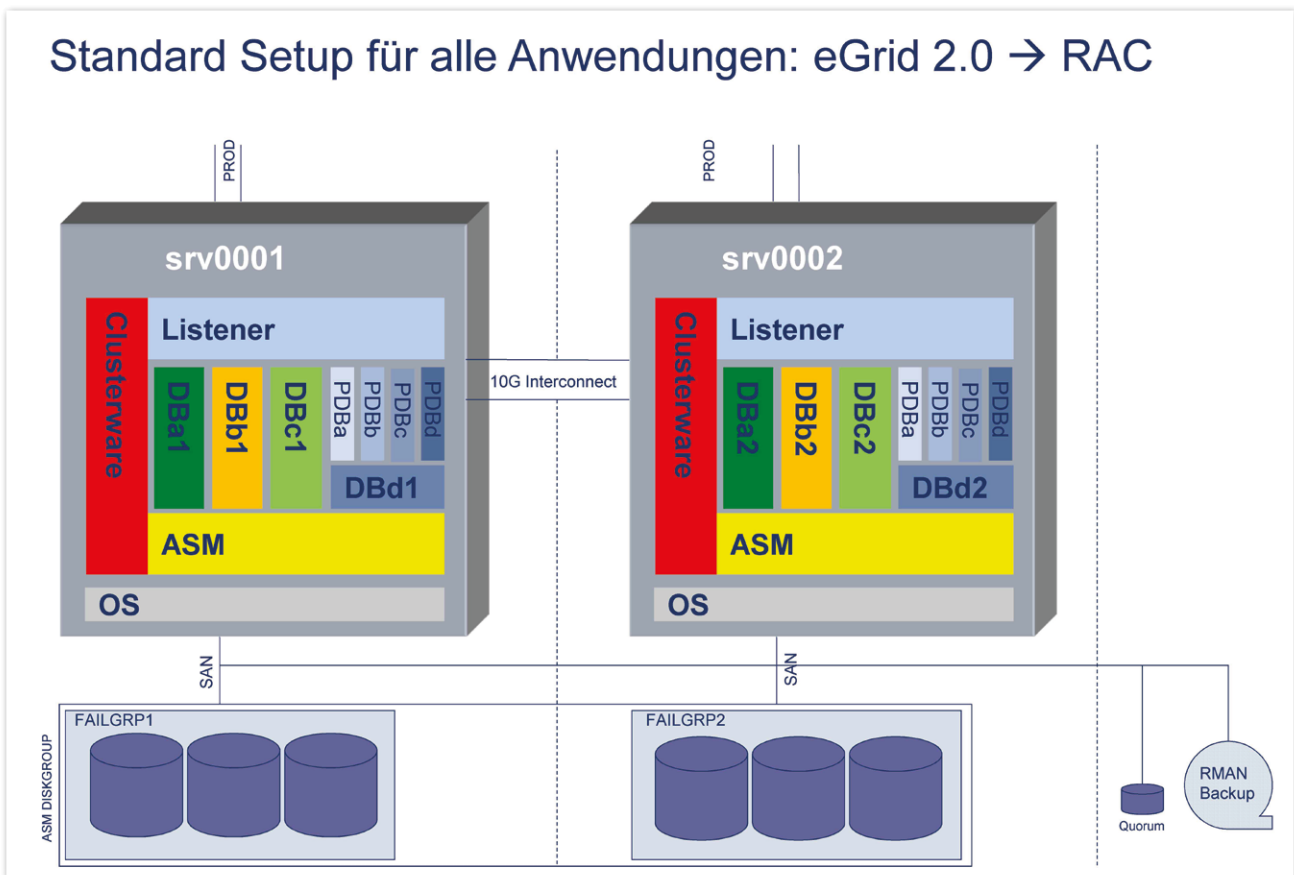


Abbildung 1: Konsolidierungsumgebung – RAC-Stretched über 20 km Distanz

Wie beschrieben ist das Oracle-In-Memory Konzept ein „Store“ – es muss also gezielt ausgewählt werden, welche Objekte in den Speicher kommen. Hierbei ist es interessant, dass Oracle einen „Tiering“-Ansatz unterstützt. Eine Tabelle muss daher nicht zwingend komplett in den Hauptspeicher passen. Fehlende Elemente werden bei Bedarf „on thy fly“ vom Storage nachgeladen.

Es ist sehr gut, dass die Datenbestände beim Laden in den Hauptspeicher komprimiert werden. In Tests hat der Autor mit Echtdateien in der Regel einen Kompressionsfaktor zwischen 2 und 10 (median bei 5) feststellen können. Dabei ist keine „Advanced Compression“-Option erforderlich – der interne Kompressions-Algorithmus ist völlig unterschiedlich.

Eine Datenbank-Tabelle wird im System mit der Syntax „ALTER TABLE TGRP\$\_IPOS\_AZV\_\$00001 INMEMORY MEMCOMPRESS FOR QUERY NO INMEMORY (COL9\_PB15\_ANME, COL10\_PB15\_ENME);“ als „In Memory“ markiert. Im Beispiel wurden für Abfragen uninteressante Spalten einfach ausgeklammert, sie belegen da-

mit auch keinen kostbaren Hauptspeicher. Ohne weitere Angaben werden die so markierten Datenbestände erst beim ersten Zugriff in den Speicher kopiert. Wenn dies unmittelbar erfolgen soll, so ist das Statement um den Parameter „PRIORITY LOW|MED|HIGH|CRITICAL“ zu ergänzen. In dem Fall lädt die Datenbank beim Start die Tabellen in der Reihenfolge der gesetzten Priorität in den Hauptspeicher.

Soll die In-Memory-Option im RAC genutzt werden, sind zuvor einige Parameter zwingend zu beachten, um keine negativen Nebeneffekte zu produzieren. „PARALLEL\_FORCE\_LOCAL“ muss auf „FALSE“ (default) stehen, denn leider steht nur auf den Engineered-Systems die Option zur Duplizierung des In-Memory-Stores zur Verfügung; in sonstigen Umgebungen werden die Daten in den Caches verteilt abgelegt. Anderenfalls würde der Remote-Knoten die Daten von der Disk lesen und damit die Performance massiv einbrechen. „PARALLEL\_DEGREE\_POLICY“ muss auf „AUTO“ oder „ADAPTIVE“ stehen (zur Not auch auf „LIMITED“ – wegen der

Funktion „In-memory-parallel execution“, default = „MANUAL“) und „PARALLEL\_DEGREE\_LIMIT“ dementsprechend auf einen Wert größer/gleich „2“ gesetzt sein.

Oracle setzt den Parameter „INMEMORY\_MAX\_POPULATE\_SERVERS“ (Server, die den In-Memory-Store beladen) auf den Wert gleich der Anzahl CPUs. Dies ist bei Systemen mit sehr vielen Prozessoren oftmals etwas zu gut gemeint und sollte je nach Leistungsfähigkeit des Storage unter Umständen angepasst werden.

### Der Nutzen

Das Lastverhalten von Datenbanken wird bei der Konsolidierung gerne zunutzen gemacht – bei klassischen Umgebungen packt der Autor Datenbanken mit unterschiedlichen Lastprofilen (Tages-/Nachtverarbeitung) sehr gerne zusammen. Dabei profitieren beide Umgebungen von den jeweils wechselseitig ungenutzten CPU/IO-Kapazitäten der anderen Datenbank.

Um ein vergleichbares Verhalten auch bei der Nutzung des In-Memory-Stores zu haben, werden Tabellen, die beispiels-

```

SQL> show parameter inmemory
NAME                                     TYPE                                VALUE
-----
inmemory_clause_default                 string                              DEFAULT
inmemory_force                           string                              DEFAULT
inmemory_max_populate_servers            integer                             4
inmemory_query                           string                              ENABLE
inmemory_size                            big integer                         0
inmemory_trickle_repopulate_servers_    integer                             1
percent
optimizer_inmemory_aware                 boolean                             TRUE

```

Listing 1

weise in der Nacht von einer PDB aus dem Tagesbetrieb nicht mehr erforderlich sind, mittels „ALTER TABLE TGRP\$\_IPOS\_AZV\_\$\$00001 NOINMEMORY;“ aus dem Speicher entfernt. Damit steht der Hauptspeicher nun für andere Objekte (PDBs) wieder zur Verfügung und kann von diesen genutzt werden, ohne dass die Funktion grundsätzlich beschränkt ist. Die für die nachaktive Datenbank benötigten Objekte werden mit dem Parameter „INMEMORY“ markiert und wahlweise mittels SQL-Statement oder „PRIORITY“-Parameter in den RAM kopiert.

Leider bringt dieser Ansatz bei klassischen Non-PDBs nichts. Auch wenn man hier Objekte aus dem In-Memory-Store entlädt, wird die SGA deswegen nicht verkleinert. Da PDBs sich die SGA teilen, kann man dies dort zum Vorteil nutzen. Das

Verfahren lässt sich über den Datenbank-Scheduler gut automatisieren und die vorherigen Zustände in entsprechenden eigenen Steuertabellen speichern. Somit ist das Verfahren auch dynamisch in der Nutzung und bedarf keiner separaten aufwändigen manuellen Pflege.

Leider ist zum aktuellen Zeitpunkt die Aufrüstung von Hauptspeicher bei steigender Modulgröße nicht ansatzweise linear. In den kommenden Jahren wird RAM jedoch Schritt für Schritt deutlich günstiger werden – parallel aber auch die Anforderungen der Anwendungen und die verarbeiteten Datenmengen massiv steigen. Daher ist davon auszugehen, dieser kleine Trick wird länger überleben – sofern denn Datenbanken mit unterschiedlichen Lastprofilen vorhanden sind. In Tests zeigten die PDBs keine messba-

ren Performance-Unterschiede zu den klassischen Datenbanken, daher sind sie auch mit In-Memory ein hervorragendes Mittel zur Konsolidierung.



Jens-Christian Pokolm  
jens-christian.pokolm@postbank.de

## Richtigstellung zum Artikel von Andrew Lacy in der letzten Ausgabe

Nachdem mein Artikel „Praxisbericht: Downgrade Datenbank Enterprise Edition auf Standard Edition One“ in der DOAG News 06/2014 erschienen ist, wurden wir darauf angesprochen, dass der letzte Satz des Artikels den Eindruck vermittelt, Kunden könnten aufgrund einer Datenbank-Migration durch einen Oracle-Platinum-Partner ein Audit vermeiden. Im

besprochenen Fall hat Oracle tatsächlich das Audit vereinfacht, da wir in Abstimmung bestätigen konnten, dass durch die Migration und die damit verbundene Neulizenzierung nun korrekt lizenziert sei.

Generell ist es natürlich nicht so, dass aufgrund der Durchführung der Migration durch einen Platinum-Partner Audits vermieden werden können. Das Recht und

die Initiative zu einem Lizenz-Audit liegt ausschließlich bei Oracle und wird durch diese wahrgenommen. Ich bitte um Entschuldigung, sollte dieser Satz missverstanden worden sein.

Andrew Lacy  
OPITZ CONSULTING Deutschland GmbH

# Mit dem **DOAG EX**aday zur *Pole-Position*

*28. April 2015 in Frankfurt*

*Was sich unter der Haube der Exadata versteckt, fasziniert und wer die Maschine zum Laufen bringt, spricht in den höchsten Tönen über ihre extreme Leistungsfähigkeit. Mit der Maschine alleine ist die Pole-Position allerdings noch lange nicht gesichert. Exadata und Oracle Database Appliance (ODA) brauchen extra ausgebildete Spitzenfahrer.*

*Die DOAG trommelt am 28. April in Frankfurt erfahrene Exa- und ODA-Piloten zusammen die zeigen, wie die integrierten Systeme im Einsatz funktionieren und was für die Evaluierung berücksichtigt werden muss.*



# Pragmatischer Einstieg in Daten-Analyse und Konsolidierung: Hadoop meistert den Spagat zwischen Big Data und Kosteneffizienz

Oliver Herzberg und Carsten Herbe, metafinanz GmbH

Die Finanz- und Versicherungsbranche steht vor großen Herausforderungen. Einerseits müssen kostspielige historische Systeme konsolidiert werden, andererseits steigt mit neuen digitalen Geschäftsmodellen und Regulatorik der Bedarf an Big-Data-Lösungen. Hadoop eröffnet hier neue Chancen, um das Datenmanagement strategisch neu zu überdenken und den Spagat zwischen Innovation und Kosten zu meistern.

Es ist eine alte Grundsatzfrage: Ist die IT in den Unternehmen nur ein Kostenfaktor oder bildet sie die Basis für den Betrieb und stellt als Innovator die Wettbewerbsfähigkeit her? Zumindest in der Finanz- und Versicherungswirtschaft ist diese Frage schon lange entschieden. Ohne IT würde heute der Betrieb zum Erliegen kommen, die Entwicklung von Produkten, die allesamt auf ausgeklügelter Software basieren, wäre nicht mehr vorstellbar. Dennoch sind auch in dieser Branche die

Zeiten vorbei, in denen für Technologie nahezu grenzenlose Budgets zur Verfügung stehen und Individualfertigungen als Status quo gelten. Die Zeichen stehen auf Industrialisierung, was für die IT bedeutet, dass sie Service-orientierter wird und im Zusammenspiel mit den Geschäftsbereichen neu organisiert.

Als große Herausforderung für die IT-Verantwortlichen gilt dabei weiterhin die Verteilung der großen Kostenblöcke. Laut der Gartner-Studie „The 2014 IT Agenda“

fließen 41 Prozent der IT-Kosten in die Rechenzentren, 26 Prozent gehen in die Netzwerk-Infrastruktur und 20 Prozent in den Betrieb der Clients. Für innovative Themen bleibt dabei nicht mehr viel übrig. Genau an diesem Punkt setzen viele CIOs mit Restrukturierungsmaßnahmen an, wie das aktuelle Beispiel bei einem großen Versicherungskonzern zeigt. Das IT-Management initiierte hier ein Konsolidierungsprojekt, das sich über alle weltweiten Rechenzentren erstreckt, gleichzeitig wer-

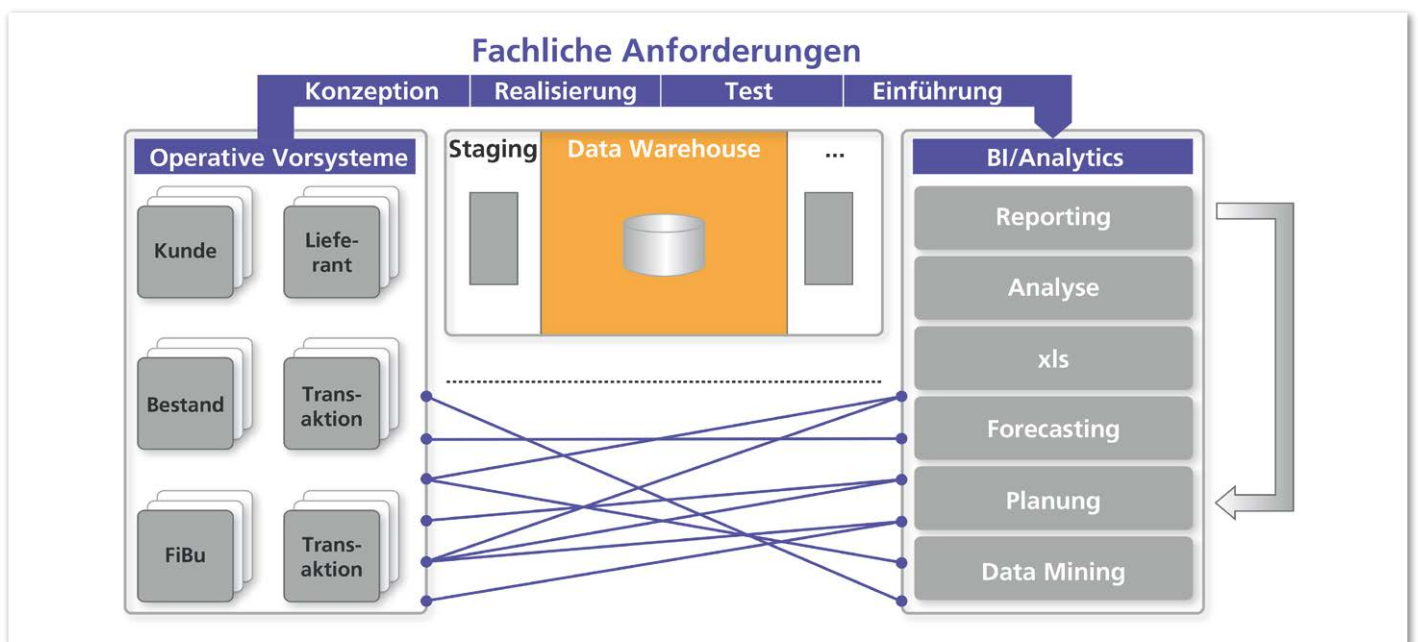


Abbildung 1: Leistungsfähige, flexible und elastische DWH- und BI-Infrastrukturen wirken vor allem bei künftigen Change-Kosten



den auch die Netzwerk-Infrastruktur erneuert und die Arbeitsplätze auf zentral bereitgestellte virtuelle Clients umgestellt.

**Datenbeschaffung sorgt weiterhin für hohe Kosten**

Auch die Modernisierung historisch gewachsener, teilweise veralteter IT-Anwendungen steht in der Finanzwirtschaft seit einiger Zeit ganz oben auf der Agenda. Viele Unternehmen konsolidieren ihre Kernsysteme und betreiben mit großem Aufwand die Erneuerung ihrer Data Warehouses. Hier rechnen Experten auch weiterhin mit steigenden Kosten; als dominierende Bereiche gelten leistungsfähige Data Warehouses und BI-Infrastrukturen (siehe Abbildung 1). Den Löwenanteil der Kosten verschlingen mit 60 Prozent die Datenbeschaffung, -validierung und -transformation, zudem schlagen aufwändige Testphasen überdurchschnittlich zu Buche.

Man ist sich in der Branche einig, dass die Datenmengen weiterhin stark wachsen und damit die Kosten weiter nach oben treiben. Als Hauptgrund gelten die umfangreichen regulatorischen Anforderungen wie Solvency II, Basel III, IFRS, EBA Reporting oder aktuell BCBS239. Wie das Marktforschungsunternehmen Lünendonk herausfand, wird allein die Umsetzung der Regulierungsaufgaben von Solvency II die Versicherungswirtschaft bis zum Jahr 2020 belasten. Neben der Beherrschung der Datenvolumina wird es aber auch verstärkt um den Ausbau der Analyse-Anwendungen gehen, mit denen

Versicherer beispielsweise Kunden- oder Risikogruppen besser segmentieren können.

**Solvency II & Co. verursachen enorme Datenmengen**

Zu den derzeit größten Herausforderungen zählt die Umsetzung der wachsenden Zahl an Compliance-Vorgaben. Diese Aktivitäten binden Ressourcen in der IT und den Fachbereichen; die daraus entstehenden Kosten belasten die Gesamtbudgets und schränken den Spielraum für Innovationen weiter ein. Da meist auch ein hoher Umsetzungsdruck besteht, kommen oft Zwischenlösungen zum Einsatz, die im Betrieb und in der späteren finalen Lösung weitere Ausgaben nach sich ziehen.

Wie sich die Anforderungen an eine zukünftige IT im Versicherungsgewerbe ändern, hat die Lünendonk-Studie ebenfalls untersucht. Demnach wird es bis zum Jahr 2020 weitere Firmenfusionen geben und die Kooperationen mit Banken werden ausgeweitet, um das Geschäft mit dem Asset- und Vermögensmanagement auszubauen. Zunehmend werden mobile Technologien zum Einsatz kommen, um etwa die Kunden-Kommunikation über Apps anzubieten und den direkten Vertrieb von Versicherungsprodukten auszubauen.

Explodierende Datenmengen kommen aber auch von der generellen Digitalisierung immer weiterer Lebensbereiche. Unternehmen sind auch hier gefordert, solche Quellen gewinnbringend zu analysieren und zu veredeln. Daten mutie-

ren damit zum neuen Rohstoff, den sich Unternehmen im Sinne neuer Geschäftsmodelle zunutze machen müssen. Das strategische Management interner und externer Daten gewinnt weiter an Bedeutung, Datenmanagement wird zu einem elementaren Bestandteil der IT-Strategie von Unternehmen. Zusammenfassend ergeben sich aus den geänderten Rahmenbedingungen des Marktes und der IT folgende drei Herausforderungen für CIOs:

- Sie müssen für die Umsetzung regulatorischer Vorgaben einen großen Aufwand betreiben, was mit hohen Kosten ohne Ertrag verbunden ist
- Sie müssen die Digitalisierung aller gesellschaftlichen und wirtschaftlichen Bereiche erkennen und sich zunutze machen, um neue Märkte zu erschließen und Kunden zu binden
- Sie müssen Initiativen ergreifen, um die gewaltigen, weiter wachsenden Datenschatze zu heben und für geschäftliche Szenarien zu nutzen

**Das Big-Data-Missverständnis**

Die Auflistung macht deutlich, dass praktisch alle Konsolidierungs- und Innovationsthemen unmittelbar oder indirekt mit Daten-Management zu tun haben. Seit einigen Jahren kursiert dafür das Schlagwort „Big Data“ – als Schublade für alles, was mit umfangreichen Datenvolumina und Analysen zu tun hat. Allerdings hat sich nach der ersten Big-Data-Welle auch etwas Ernüchterung breitgemacht. Zu oft

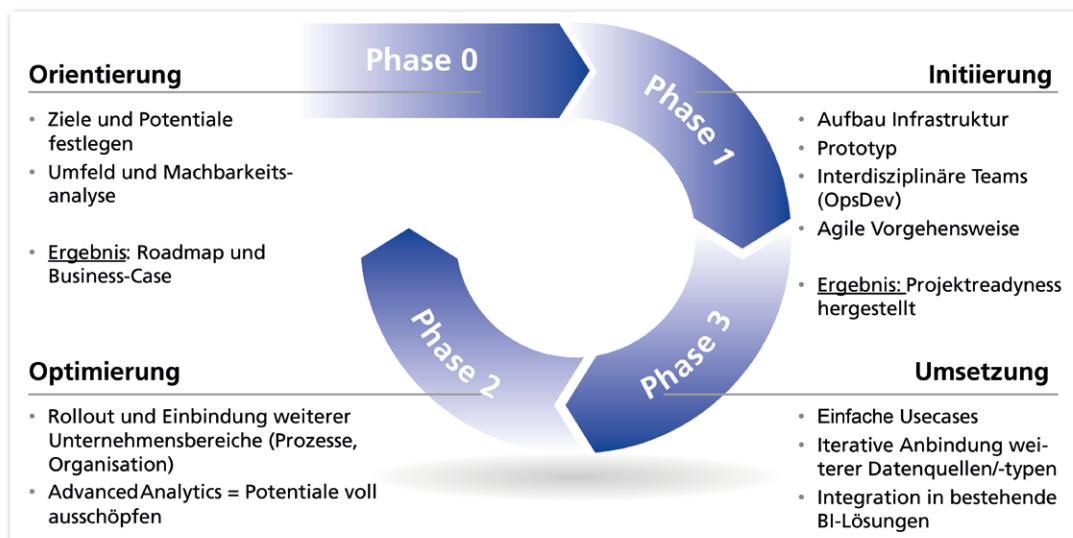


Abbildung 2: Ein iteratives Vorgehensmodell zur Einführung von Hadoop umfasst vier Kernphasen

wurden prestigeträchtige Megaprojekte vorgeführt, mit dem Effekt, dass viele Entscheider heute Big Data mit kostspieligen Initiativen assoziieren, die erst nach längerer Zeit geschäftliche Mehrwerte und einen Return on Investment einfahren.

Aus Sicht der CIOs jedenfalls sind diesbezüglich die Prioritäten klar definiert, wie eine andere Gartner-Studie herausfand. Gefragt nach den größten Herausforderungen bei Big Data und Analytics nannten die Entscheider mit großer Mehrheit die Antwort: „Wie man am schnellsten Mehrwerte aus großen Datenbeständen generieren kann.“

### Den Spagat zwischen Innovation und Kosten meistern

Mehrwerte aus großen Datenbeständen – aber ohne Großprojekte und Riesensbudgets? Dass dieser vermeintliche Widerspruch auflösbar ist, beweist die noch relativ junge Technologie „Hadoop“. Deren großer Vorteil als Daten-Plattform liegt darin, dass Einführungen mit kleinen Projekten starten können, die zunächst einmal auf aktuelle und pragmatische Ziele wie Konsolidierung und Kostensenkung fokussiert sind. Bei Hadoop handelt es sich um ein Open-Source-Framework, das sich vor allem durch zwei Bereiche auszeichnet:

- Es bietet neue, bisher nicht umsetzbare Analytics-Möglichkeiten
- Es ermöglicht konkrete Einsatz-Szenarien, um die Infrastruktur- und Speicherkosten zu senken

Die Stärken von Hadoop liegen vor allem im dezentralen Speichern und parallelen Verarbeiten sehr großer Datenmengen, wobei hier das horizontal verteilte Dateisystem HDFS zum Einsatz kommt. Es besteht aus einem Cluster mit Standardservern und kann daher beliebige Datenformate in beliebigen Größenordnungen speichern. Ein weiteres Charakteristikum ist das dazugehörige MapReduce-Framework, das eine parallele Verarbeitung der Daten ermöglicht. Somit realisiert Hadoop ein kostengünstiges Speichern von beliebigen strukturierten und nicht-strukturierten Daten sowie eine parallele Verarbeitung riesiger Datenmengen.

Hadoop-Cluster skalieren übrigens linear, was bedeutet, dass zehn Prozent mehr Knoten die Speicherkapazität um zehn Prozent erweitern und damit die Leistung um zehn Prozent steigt. Ein weiterer Vorteil von Hadoop ist das mittlerweile große Ökosystem an Tools, das vielfältige Erweiterungsmöglichkeiten wie beispielsweise Datenauswertungen mit SQL bietet.

### Einsatzszenarien von Hadoop

Die möglichen sinnvollen Einsatzszenarien von Hadoop im Banken- und Versicherungsumfeld lassen sich in folgende drei Bereiche einteilen:

- **RDBMS Offload**  
Hadoop-Cluster ermöglichen die Optimierung von Speicherplatz, um etwa bestehende, komplexe ETL-Prozesse

oder klassische Datenbank-Systeme zu ersetzen

- **DWH-Extension**  
Bestehende Data Warehouses lassen sich durch Hadoop funktional erweitern, indem vorgeschaltete ODS-Datensammler eingesetzt werden
- **Big Data Exploration**  
Einsatz von Big-Data-Anwendungen, die Daten unterschiedlichster Quellen sammeln sowie Data Mining und Machine Learning ermöglichen

Auch wenn Hadoop von der Zielsetzung her für viele klassische BI-Szenarien infrage kommt, erfordert es teilweise völlig andere Methodiken und Herangehensweisen. Typisch für die klassische BI-Welt sind beispielsweise klar definierte Projektziele – ob es sich dabei um den Aufbau eines Near-Time-Reportings, eines Management-Cockpits oder eines Dashboards für das GuV-Stress-Testing handelt. Hadoop-Projekte hingegen gestalten sich zumindest am Anfang oftmals eher als Forschungs- und Entwicklungsprojekte, beispielsweise wenn es um Themen wie „Data Exploration“ geht. Die IT stellt hierbei eine Plattform zur Verfügung, auf der das Business herausfinden kann, welche Fragen sinnvoll sind.

### Beispiel-Szenarien für erfolgreiche Hadoop-Einführungen

Um Hadoop erfolgreich als Big-Data-Lösung einzuführen, empfiehlt sich ein iteratives Vorgehensmodell, das aus folgenden vier Phasen besteht (siehe Abbildung 2):

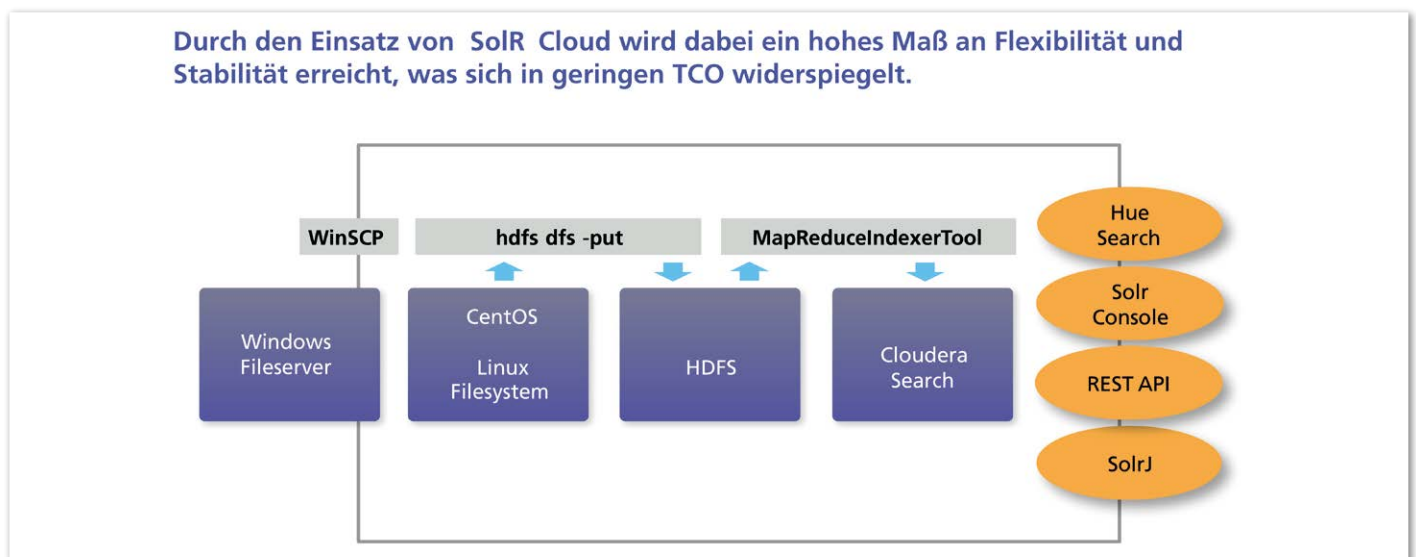


Abbildung 3: Das Hadoop-Search-API analysiert performant den Personalstamm

- Orientierungsphase: Strategie-Entwicklung, Ermittlung des Business-Value und Beispiele für Use-Cases
- Aufbau der Infrastruktur, Prototyp, Durchstich „front to back“
- Umsetzung mit einfachen Use-Cases für einen Quick-Win, stufenweiser Ausbau (Daten) und Integration (Prozesse und Organisation)
- Optimierung: unternehmensweite Integration von Big Data und Advanced Analytics

Anhand von Beispielszenarien lassen sich die Möglichkeiten von Hadoop sowie deren konkreter geschäftlicher Nutzwert anschaulich machen. Die folgenden drei Szenarien bieten einen Einblick in den Charakter typischer Hadoop-Lösungen.

### Szenario 1: Proof of Concept „HR – strategische Personalentwicklung“

Bei der Personalsuche für IT-Projekte muss eine HR-Abteilung schnell auf Kun-

denanfragen reagieren und die vorhandenen Skills abfragen können. Wenn es um die Entwicklung des eigenen Personals geht, müssen die Skills der Mitarbeiter ständig mit den Marktbedürfnissen verglichen und daran ausgerichtet werden.

Informationen darüber liegen sowohl strukturiert als auch unstrukturiert in Projekt-Historien, Schulungsplänen, Gehalts spiegeln der Branche und internen wie externen Stellenausschreibungen vor. Zu Projektbeginn ist unklar, ob die Daten sich in Beziehung setzen lassen.

Im ersten Schritt wird ein Prototyp auf Hadoop-Basis in einer definierten Projektumgebung entwickelt. Dabei wird auf den Einsatz von Open-Source-Technologien geachtet, um ein Vendor-Lock-in zu vermeiden. Mitarbeiterprofile werden in das Hadoop Distributed File System (HDFS) geladen und mit MapReduce indiziert. Auswertungen sind über Velocity, Solr Console, REST API und Sorj auf den Originaldaten möglich. Durch den Einsatz von Solr Cloud wird ein hohes Maß an Flexibi-

lität und Systemstabilität erreicht mit der Folge geringer TCO. Bereits in dieser Phase kann das Unternehmen bei Kundenanfragen feststellen, ob das Know-how vorhanden ist.

Im zweiten Schritt werden die Daten aus dem ERP-System integriert, um Mitarbeiter mit Projekt und Laufzeit zu ermitteln. Im dritten Schritt integriert man Daten aus öffentlichen Quellen wie Stellenbörsen. Das Unternehmen kann dann schnell auf angefragte Ressourcen reagieren und Skills strategisch und an den Kundenbedürfnissen orientiert weiterentwickeln (siehe Abbildung 3).

### Szenario 2: Zentralisierung der Archivierungsmedien

Zur Erfüllung der Grundsätze ordnungsgemäßer Buchführungssysteme (GOBS) oder der Grundsätze zum Datenzugriff und zur Prüfbarkeit digitaler Unterlagen (GdPdU) müssen geschäftsrelevante Daten zehn Jahre archiviert werden. Die Archivierung von Daten erfolgt oft im Data Warehouse



Mehr Zeit für andere Dinge.  
Experten für Cloud und Exadata.



oder in speziellen Archivierungs-Marts und verursacht so hohe Kosten. Durch Weiterentwicklung von Software und Dateiformaten steigt der Aufwand, gespeicherte Daten erneut zu laden und zu verarbeiten.

Die in den entsprechenden Data-Mart gehaltenen Daten werden in den Hadoop-Cluster repliziert. Der Quickwin besteht darin, dass sich die Speicherkosten reduzieren, da alle älteren Daten aus dem Data-Mart gelöscht werden. Im nächsten

Schritt lädt man die Archivdaten, die in unterschiedlichsten Formaten vorliegen, in das Hadoop-Cluster, um sie zu analysieren und zu verarbeiten. In der Regel erfolgt dies mit den gleichen BI-Tools und Reports, mit denen auch schon die Daten in der relationalen Datenbank ausgewertet werden. Möglich machen dies Technologien wie Hive, für die viele Tool-Hersteller schon eine Schnittstelle anbieten (siehe Abbildung 4).

**Szenario 3: Hadoop als Staging-Area im ETL-Prozess**

Im Staging- und ETL-Prozess eines Finanzdienstleisters werden im Rahmen der Tagesend-Verarbeitung große Datenmengen zwischengespeichert. In der Praxis erweisen sich die vorhandenen Daten-Infrastrukturen als nicht ausreichend, um den Bedarf an schneller und genauer Information abdecken zu können. Daher setzen die Fachbereiche oft individuelle Lösungen ein – die

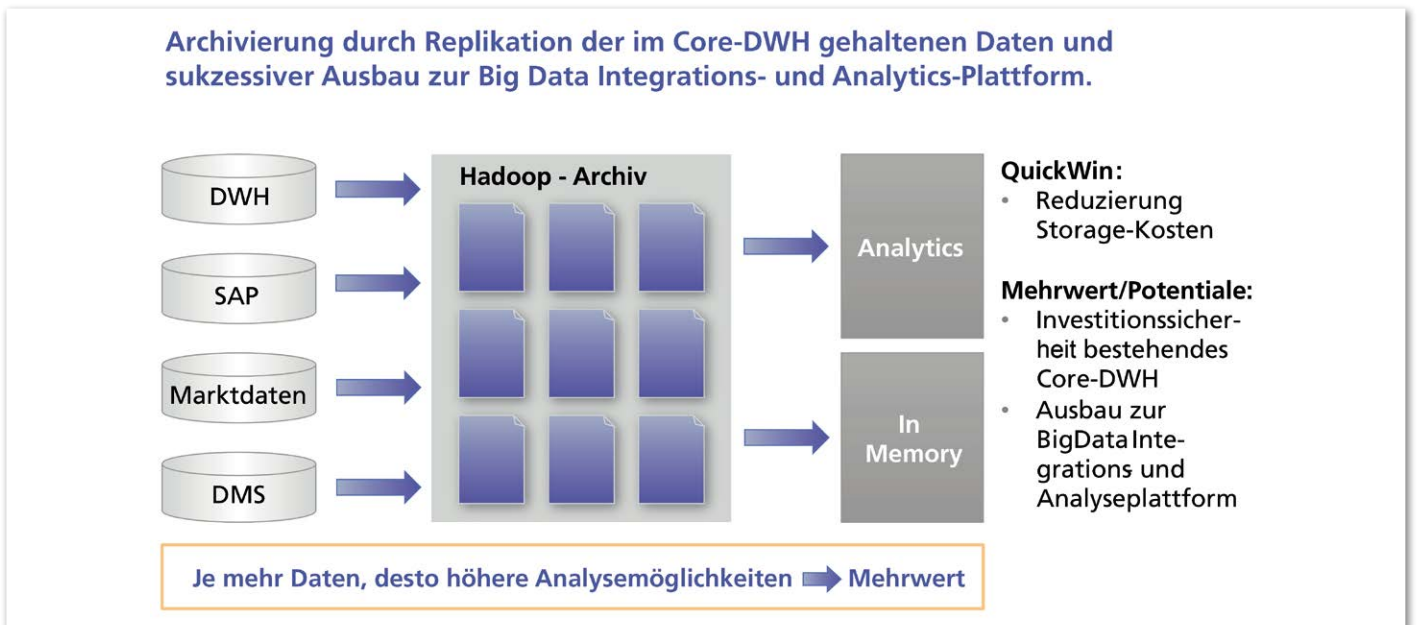


Abbildung 4: Zentrale Archivierung mit Hadoop

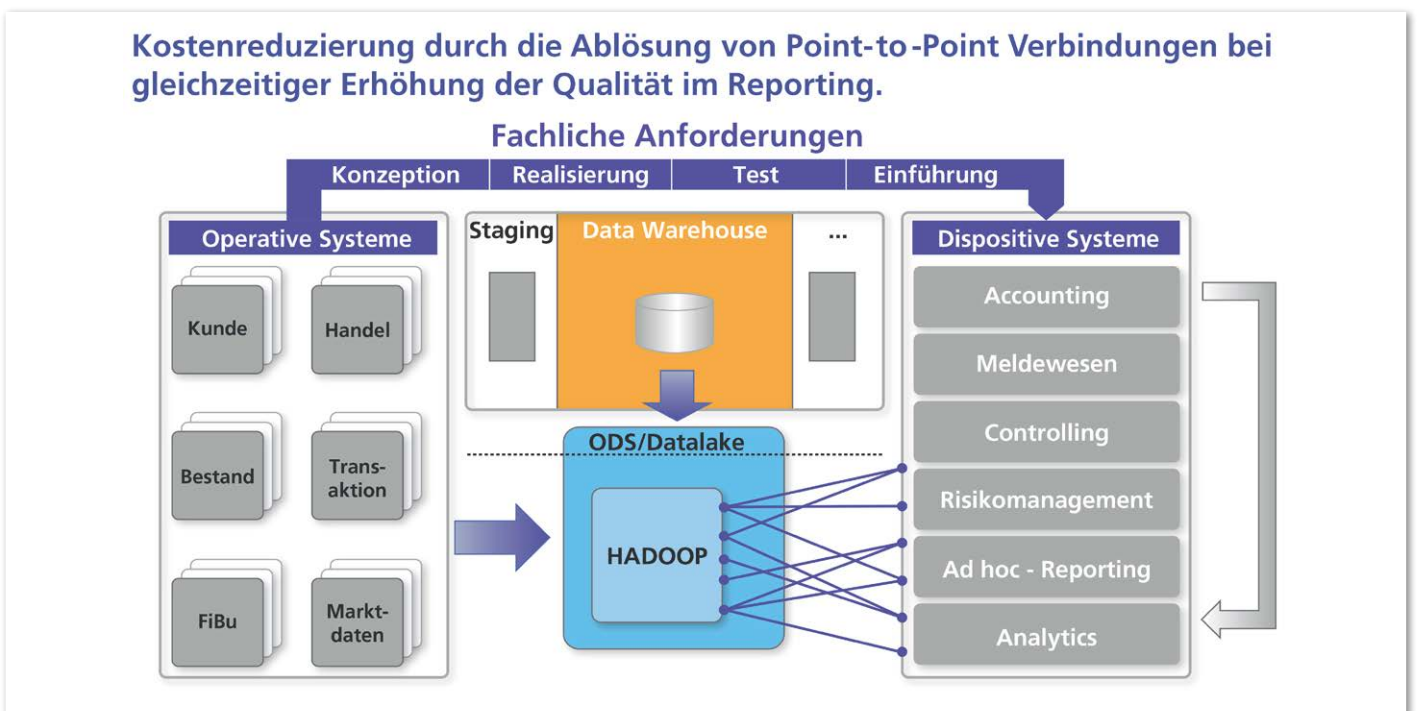


Abbildung 5: Ergänzung des DWH durch den Aufbau eines Hadoop ODS



allerdings den regulatorischen Anforderungen nur bedingt entsprechen. Strukturelle Änderungen des Datenmodells an den operativen Vorkomplexen, zum Beispiel zur Anpassung an regulatorische Anforderungen, sind im Change sehr teuer, da die komplette ETL-Strecke konzipiert, geändert und durch alle Nutzer getestet werden muss.

Ziel ist ein dem Core-DWH vorgelagerter Data-Lake – ein Speicher-Repository, das zunächst große Mengen an Rohdaten speichert. Dezentrale Anwendungen in den Fachbereichen können über klare Schnittstellen sowohl auf Rohdaten aus den Vorkomplexen als auch auf die Daten des DWH und seines Data Mart zugreifen. Die fachlich erforderliche Flexibilität bleibt (siehe Abbildung 5).

Durch den Einsatz von Hadoop und HDFS lassen sich die Kosten der auf günstigen File-Servern und SANs abgelegten Daten deutlich reduzieren. Gleichzeitig stehen sie für performante Auswertungen zur Verfügung. Der Vorteil für Analytics besteht darin, dass die Daten durch Hadoop in der Granularität vorliegen, wie sie aus Sicht der fachlichen Nutzer benötigt werden. Somit lassen sich dezentrale Anwendungen reduzieren.

### Fazit

Wie am Vorgehensmodell und an den Beispielszenarien zu erkennen ist, erfordert der Umgang mit Hadoop ein Umdenken und die Bereitschaft, unbekanntes Terrain zu beschreiten. Best Practices müssen sich erst etablieren oder an die jeweiligen Unternehmensziele angepasst werden. Generell lautet die Empfehlung, klein anzufangen, ein Gefühl für die Technologien und ihre Möglichkeiten zu entwickeln und darauf aufbauend eine dem Unternehmen adäquate Strategie zu erarbeiten.

Keinesfalls werden klassische Data Warehouses oder relationale Datenbanken überflüssig, stattdessen werden sie ergänzt und erweitert. Damit sind die Investitionen in bestehende DWH-, BI- und Reporting-Infrastrukturen gesichert. Gleichzeitig eröffnet Hadoop neue Potenziale für Analytics auf der einen und Kosteneinsparungen auf der anderen Seite. Vor dem Hintergrund steigender Ansprüche im Bereich Datenmanagement empfiehlt es sich damit als ideale Plattform für Konsolidierungen, innovative Lösungen und Geschäftsmodelle der Zukunft.



Oliver Herzberg  
oliver.herzberg@metafinanz.de



Carsten Herbe  
carsten.herbe@metafinanz.de

## Die (Oracle-)Welt wächst zusammen: Meeting mit der Taiwan Java User Group in Taipei

Gunther Pippèrr, freiberuflicher Berater bei GPI Consult, verbrachte im Oktober 2014 zwei Wochen in Taiwan und initiierte dort ein Treffen mit der Taiwan Java User Group. Neben einem regen Erfahrungsaustausch mit der taiwanesischen Community konnte er auch viele kulturelle Eindrücke sammeln.

Dass er ausgerechnet am anderen Ende der Welt auf Gleichgesinnte treffen würde, die genau wie er große Anhänger des NoSQL-Ansatzes sind, hätte sich Pippèrr nun wirklich nicht träumen lassen. Doch bei genauerer Betrachtung sei das in einem Land wie Taiwan gar nicht so weit hergeholt: „Das schnelle Marktwachstum sowie der hohe Marktdruck in den asiatischen Ländern erfordern, alles möglichst schnell umzusetzen und möglichst schnell fertig zu

werden – ganz anders als in Deutschland, wo Wartbarkeit, Sorgfalt und niedrige Betriebskosten entscheidend sind“, berichtet er von seinen Beobachtungen. „Gerade das könnte ein Grund dafür sein, weshalb der NoSQL-Ansatz in den asiatischen Ländern so beliebt ist“, vermutet er. Denn auch bei NoSQL gehe es darum, schnell Lösungen für Applikationen zu finden, die in der Regel nur kurz zum Einsatz kommen.

Pippèrrs Reise begann eigentlich mit der Einladung eines befreundeten Software-Entwicklers, der schon seit einigen Jahren in Taiwan lebt. Pippèrr hatte bald eine Idee: Zusammen mit seinem Reisegefährten, Slava Schmidt, ebenfalls Software-Entwickler, wollte er ein Treffen mit der Oracle Anwendergruppe vor Ort or-

ganisieren. Auf der Suche nach einem Ansprechpartner wandte sich Pippèrr an Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG. Dieser freute sich über das Engagement: „Der von Pippèrr initiierte Austausch ist ein gutes Beispiel für eine gelungene Kommunikation über Ländergrenzen hinweg und zeigt, wie die Usergruppen auf internationaler und persönlicher Ebene zusammenarbeiten können. Die DOAG unterstützt bei solchen Vorhaben gerne bei der weltweiten Kontaktvermittlung.“ So konnte er Pippèrr nach kurzer Zeit einen Kontakt zur taiwanesischen Java User Group vermitteln.

Marina Fischer  
marina.fischer@doag.org

# SAP-Systeme mit Oracle-Technologien konsolidieren

Jan Brosowski, Oracle Deutschland B.V. & Co KG

Die Konsolidierung betrieblicher Informationssysteme wie ERP, CRM oder SCM ist in Anbetracht der immer stärker werdenden Infrastruktur-Komponenten seit vielen Jahren Standard. Während es ursprünglich nur darum ging, den Hardware-Footprint zu verkleinern, steht man heute vor der schwierigeren Aufgabe, den Betrieb der Landschaft zu vereinfachen. Dieser Artikel beschreibt verschiedene Szenarien zur Konsolidierung von SAP-Systemen auf Basis einer Kunden-Installation.

SAP-Systemlandschaften sind in der Regel über Jahre gewachsen und unterliegen einem ständigen Wandel. Sie bestehen aus einer großen Anzahl unabhängiger und doch miteinander vernetzter SAP-Systeme. Das Unternehmen hat klein angefangen und später Systeme für bestimmte Anwendungen ergänzt. Aus unterschiedlichen Gründen hat man dabei Funktionen in getrennten Systemen aufgesetzt, die dann über Schnittstellen miteinander kommunizierten. So entstand im Laufe der Zeit eine große Anzahl von Varianten, wie diese Systeme technisch aufgesetzt sind. So sind durch die zeitlich versetzte Installation, aber auch durch unterschiedliche Update-Zyklen, Patching-Reglementierungen, applikatorische Abhängigkeiten und betriebliche Anforderungen üblicherweise verschiedene SAP-Releases in unterschiedlichen architektonischen Ausprägungen vorhanden.

Man versucht jetzt, dieser kostenintensiven Vielfalt durch eine Konsolidierung der Systeme Einheit zu gebieten. Dabei werden unterschiedliche Herangehensweisen verfolgt, die sich auf verschiedene Ebenen auswirken und somit unterschiedliche Einsparpotenziale eröffnen. Auf der anderen Seite sind diese Herangehensweisen mit unterschiedlichen Aufwänden bei der Umsetzung verbunden:

- Konsolidierung auf Applikationsebene**  
 Man hebt die Trennung auf und konsolidiert beispielsweise alle ERP-Lösungen in einem einzigen zentralen System. Diese Konsolidierungsvariante bietet deutliche Potenziale und Einsparungen, da sie die Anzahl der zu pflegenden Software- und Hardware-Komponenten deutlich reduziert. Auf der anderen Seite ist sie die mit Abstand aufwändigste Methode,

da die diversen in den Systemen abgebildeten Prozesse in einem gemeinsamen System standardisiert werden müssen.

- Konsolidierung auf Betriebssystem-Ebene**  
 Moderne Betriebssysteme bieten unterschiedliche Methoden zum Betrieb mehrerer SAP-Systeme in einem gemeinsamen Betriebssystem, etwa um die zur Verfügung stehenden Ressourcen an die verschiedenen Systeme kontrolliert zu verteilen. Zusätzlich gibt es beispielsweise mit Oracle Solaris Zonen-Technologien, die im Betriebssystem Virtualisierung ermöglichen. Diese Variante ist deutlich schlanker als eine Virtualisierung mit einem Hypervisor, sie erfordert eine Standardisierung auf Betriebssystem-Ebene statt auf Applikations-Ebene.
- Konsolidierung mittels Hypervisor**  
 Auf moderner Hardware können mit Hypervisoren mehrere Betriebssysteme

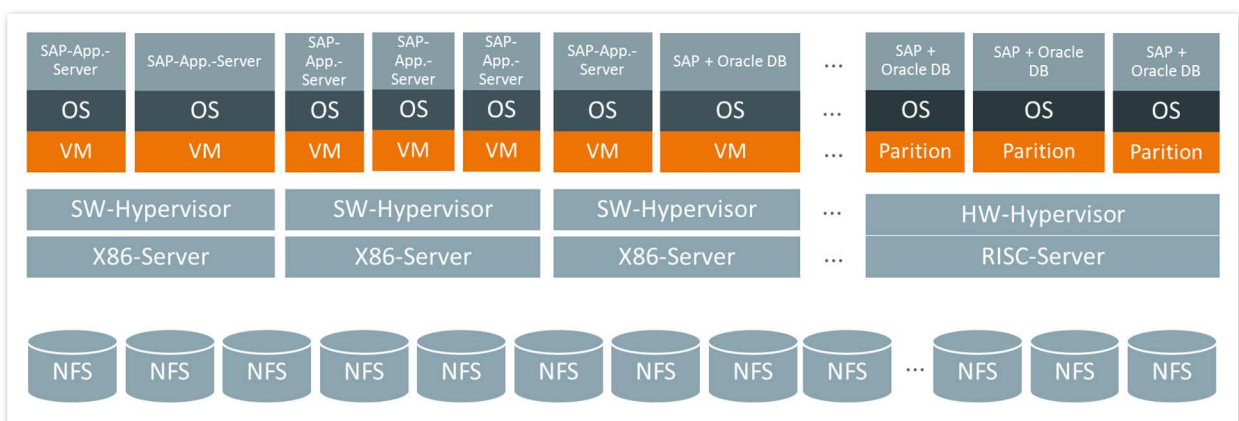


Abbildung 1: Schematische Darstellung der heutigen Architektur

	Systemschiene	SID	SAP-Release	Unicode/Non-Unicode	Datenbank	Tiers	Betriebssysteme	
Aktuell genutzt	ERP	P20	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	3	DB/CS: Solaris 10 App: 10x Linux 5.5	
		Q20	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	3	DB/CS: Solaris 10 App: 2x Linux 5.5	
		D20	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	2	Solaris 10	
		J20	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	2	Linux 5.5	
		S20	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	2	Linux 6.0	
	SCM	P21	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	2	Solaris 10	
		Q21	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	2	Solaris 10	
		D21	NetWeaver 7.0	Non-Unicode	Oracle 11.2.0.4	2	Linux 5.5	
	PLM	P28	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	3	DB/CS: Solaris 10 App: 6x Linux 6.5	
		Q28	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	3	DB/CS: Solaris 10 App: 2x Linux 6.5	
		D28	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	2	Solaris 10	
		S28	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	2	Linux 6.5	
	BW	P22	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	2	Linux 6.0	
		Q22	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	2	Linux 6.0	
		D22	NetWeaver 7.20ext	Unicode	Oracle 11.2.0.4	2	Linux 7.0	
		...	...	...	...	...	...	
	Altsysteme	ERP	P05	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10
			P10	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10
		P92	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10	
BW		P07	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10	
		P12	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10	
		P94	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10	
PLM		P08	WebAS 6.40	Non-Unicode	Oracle 11.2.0.2	2	Solaris 10	
		P98	4.70	Non-Unicode	Oracle 10.2	2	Solaris 10	

Tabelle 1

simultan gefahren werden, in denen jeweils unterschiedliche SAP-Systeme laufen. Die Zuordnung von Betriebssystemen und SAP-Systemen kann also wie bei physischer Hardware unverändert bleiben und es ist auch möglich, unterschiedliche Betriebssysteme verschiedener Hersteller simultan zu verwenden. Letztendlich findet die Standardisierung auf der Ebene des Hypervisors statt. Diese Variante ist am flexibelsten, bietet jedoch das geringste Vereinfachungspotenzial. Die Anzahl der eingesetzten Komponenten steigt sogar, da mit dem Hypervisor eine neue Ebene in den Applikations-Stack eingeführt wird.

In den letzten Jahren wurde in vielen Rechenzentren Virtualisierung durch einen Hypervisor zur Konsolidierung angewandt und damit eine höhere Auslastung der Hardware-Systeme bei gleichzeitig reduzierter Anzahl der genutzten Komponenten erreicht. Allerdings wurde im gleichen Moment die Anzahl der zu wartenden Komponenten erhöht. Es stellt sich die Frage, wie man zukünftig den Betrieb von SAP-Systemen weiter verändern kann, um ihn einfacher und zuverlässiger zu gestalten.

#### Fallstudie aus der Praxis

Um die Effekte der unterschiedlichen Konsolidierungsszenarien zu verdeutlichen,

wird in der nachfolgenden Fallstudie die SAP-Landschaft der Firma GerCos (Pseudonym) betrachtet. Deren primäres Ziel besteht darin, eine neue SAP-Landschaft zu entwerfen, die folgende Ziele erfüllt:

- Ausreichend Leistung, um das erwartete Wachstum von 75 Prozent in drei Jahren aufzufangen
- Die Anzahl physischer Systeme reduzieren, da diese in der internen Verrechnung ein bedeutender Kostentreiber ist
- Weniger komplexer Betrieb, indem man die Anzahl der Komponenten mit zu pflegenden Konfigurationen (Configuration Items) reduziert

- Verbessertes Lizenzmanagement, da man verschiedene Softwareprodukte von Oracle und anderer Hersteller nach einer Core-Metrik lizenziert. Im konkreten Fall waren es neben Datenbank-Lizenzen von Oracle und einem anderen Hersteller auch Lizenzen für Backup-Clients, die nach Anzahl der CPU lizenziert wurden.

Die heutige Landschaft besteht aus mehreren Systemschienen, die unterschiedliche SAP-, Datenbank- und Betriebssystem-Releases verwenden. *Tabelle 1* und *Abbildung 1* geben jeweils einen Ausschnitt der gesamten Landschaft wieder.

Man erkennt bereits die durchgeführten Konsolidierungsmaßnahmen: Applikationsserver werden überwiegend auf einer Linux-VM-Farm betrieben und produktive Datenbanken laufen in mehreren Partitionen großer RISC-Systeme. Die gesamte zur Verfügung stehende Prozessorleistung entspricht ca. 540.000 SAPS (gemessen mit dem aktuellen SAP-

Benchmark) und insgesamt stehen ca. 3,5 TB Arbeitsspeicher auf den verwendeten Systemen zur Verfügung. Zudem belegen Oracle-Datenbanken und andere Filesysteme nochmals 140 TB auf diversen Plattensystemen. Erbracht wird diese Leistung von insgesamt 85 virtuellen Linux-Systemen sowie 16 Partitionen der verwendeten SPARC-Landschaft.

Die eingesetzten SAP-Releases der verschiedenen Systeme ziehen unterschiedliche Restriktionen nach sich. Unter anderem kann nicht jedes System ohne Weiteres von der RISC-Plattform auf eine x86-basierte Infrastruktur migriert werden. In den vergangenen Jahren hat man dies für alle Systeme, für die das mit vertretbarem Aufwand durchgeführt werden konnte, bereits erledigt und ist daher zu dem Schluss gekommen, Systeme auf der gleichen Prozessor-Infrastruktur zu halten.

GerCos setzt alle Systeme am gleichen Standort ein und hat einen mit entsprechenden Technologien gut gesicherten

und verfügbaren Serverraum. Für den Katastrophenfall sieht man einen manuellen Wechsel auf einen zweiten, entfernten Standort vor. Dieser ist nicht Teil dieser Fallstudie.

**Zusätzliche Anforderungen**

Die GerCos ist mit seiner heutigen Landschaft nicht zufrieden. Insbesondere folgende Aspekte stören:

- Es gibt zu viele Downtimes oder Zeiten für Wartungsarbeiten. So sind insbesondere im Bereich der Applikationsserver regelmäßig Wartungsfenster für das Patchen der Hypervisor-Technologie und auch der Betriebssysteme vorgesehen. Zwar werden diese Arbeiten rollierend durchgeführt, sodass nur geringe Auswirkungen auftreten, doch die Arbeiten sind dadurch praktisch permanent in der Wahrnehmung der Endnutzer. Man möchte Wartungsarbeiten zukünftig seltener für den Endnutzer sichtbar durchführen.

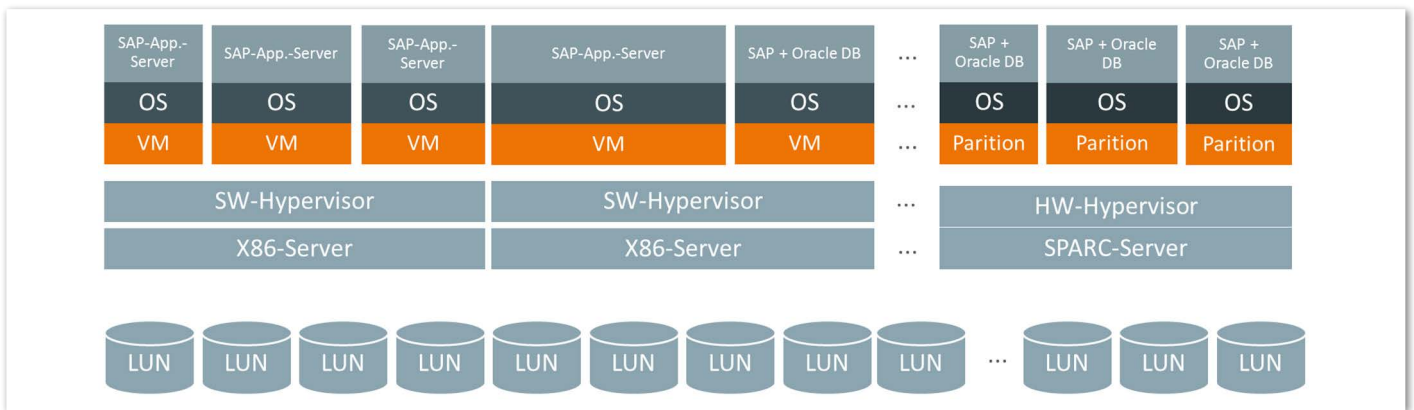


Abbildung 2: Schematische Darstellung des Szenarios 1 mit größeren VMs und SAN statt NAS

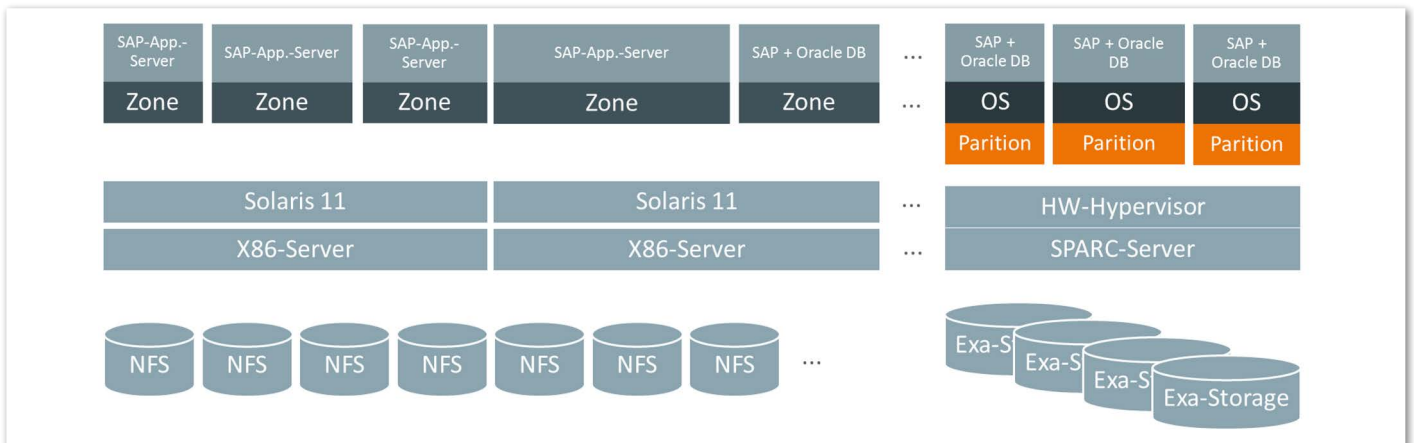


Abbildung 3: Schematische Darstellung des Szenarios 2



- In den großen, produktiven SAP-Umgebungen wurde die Leistung der Datenbank in den letzten Jahren zum Engpass bei Monatsabschlüssen und anderen lang laufenden Operationen. Die Datenbank-Performance wiederum ist vor allem durch das IO-System limitiert. Daher möchte man zumindest für die produktiven Systeme auf ein schnelleres System als die heute genutzten NFS-Filer wechseln.
- Die bislang angewandte Patchmethode bindet Mitarbeiter insbesondere am Wochenende, um die Wartungsarbeiten durchzuführen. Künftig sind Wartungsarbeiten zu einem großen Teil während der normalen Arbeitszeiten durchzuführen, ohne die Produktion zu beeinflussen.
- Eine kleine Anzahl älterer SAP-Systeme, liebevoll „Methusalems“ genannt, laufen unter sehr alten Versionen und müssen nur aus Compliance-Gründen vorgehalten werden. Für diese Systeme ist eine Migration auf moderne Betriebssysteme sehr aufwändig, da ein komplexes Upgrade der SAP-Versionen erforderlich ist. Man möchte daher eine Art Altenteil einrichten, auf dem die Methusalems in einer Umgebung laufen, die laut SAP Product Availability Matrix (PAM, siehe „<http://service.sap.com/pam>“) für sie supportet ist. Heute besteht das Altenteil aus sechs Partitionen auf den SPARC-Systemen.

### Zwei Szenarien im Vergleich

Die Konsolidierung auf Applikationsebene wurde zwar nicht von vornherein als unmöglich ausgeschlossen, doch die entsprechenden Projekte sind sehr lang laufend und würden in den kommenden Jahren nur langsam zu Resultaten führen. Daher wurde beschlossen, dass für einen Vergleich der Angebote die Anzahl der SAP-Systeme vorerst konstant bleibt. Daher wurden zwei Szenarien betrachtet:

- Die evolutionäre Weiterentwicklung der heutigen Strategie der Virtualisierung
- Ein Wechsel auf eine Betriebssystem-Virtualisierung mit Solaris Zones bei gleichzeitiger Nutzung eines Engineered Systems zur Beschleunigung des Datenbankzugriffs

### Szenario 1: Evolution

Im Bereich der Virtualisierung mittels Hypervisor hatte es seit dem letzten Technologie-Refresh der GerCos Weiterentwicklungen gegeben, die auch Auswirkungen auf die evolutionäre Weiterentwicklung der Landschaft haben würden. So ist die technische Restriktion der maximal erreichbaren Systemgröße überwunden worden, sodass in einer neuen Landschaft auch VMs mit 40.000 SAPS am Stück denkbar sind. Damit wäre es möglich, die Anzahl der notwendigen VMs, insbesondere aber auch die Anzahl der

notwendigen SAP-Applikationsserver zu reduzieren.

Der technologische Fortschritt auf CPU-Ebene lässt die Anzahl der verwendeten Hardware-Systeme verringern, wenngleich somit auch die Verfügbarkeitsanforderungen an die einzelnen Systeme gestiegen sind. Insgesamt wären statt der bislang verwendeten zweiundzwanzig Server als Hosts für virtuelle Linux-Maschinen zukünftig im Endausbau nur noch vierzehn erforderlich, und im Bereich der RISC-Server würden bei konstanter Anzahl der Partitionen zukünftig zwei statt acht physischer Server ausreichen (siehe *Abbildung 2*).

Ebenfalls stark weiterentwickelt wurden die Methoden zur Administration einer solchen Landschaft. Covered Systems wie beispielsweise die Oracle Virtual Compute Appliance bieten moderne Management-Tools, die den gesamten Lebenszyklus der VMs unterstützen. Diese vereinfachen beispielsweise auch die Wartung der gesamten Landschaft und der darin verbauten Komponenten wie Switches, Server und Hypervisoren. In Sachen Lizenz-Management wäre die Landschaft identisch mit der bisherigen, wenngleich auf der höheren Anzahl an Cores zusätzliche Lizenzen notwendig würden.

Um die geforderte I/O-Leistung zu erreichen, ist in der evolutionären Weiterentwicklung ein SAN-Speichersystem vom Typ „Oracle FS1“ in einer kleinen Ausbau-

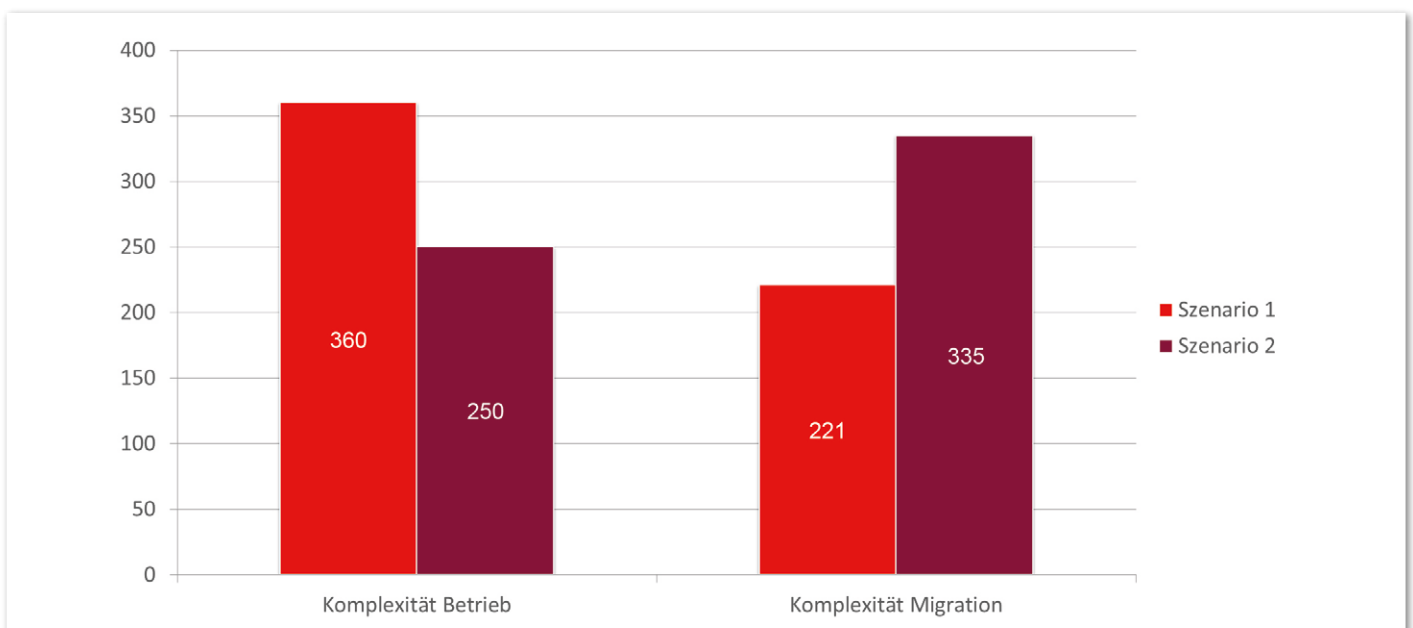


Abbildung 4: Komplexitätsbewertung

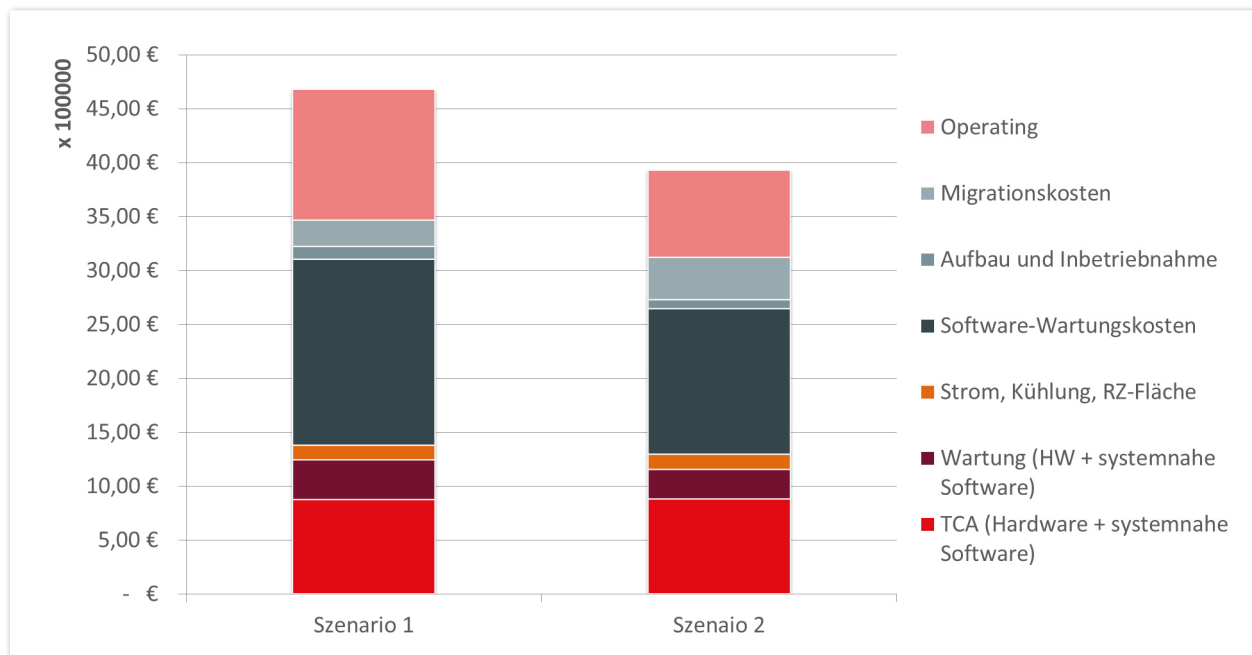


Abbildung 5: TCO-Übersicht

stufe vorgesehen. Dies verfügt gegenüber dem heute verwendeten System über ein deutliches Leistungsplus und die vorhandene QoS-Technologie stellt sicher, dass die produktiven Systeme über entsprechende Leistungsreserven verfügen. Eine weitere Stärke der Lösung besteht in den geringen Aufwänden zur Einführung, da die Ebene der Standardisierung erhalten bleibt. Die eingesetzten Linux-Betriebssysteme ließen sich im Laufe der Zeit bei Auslaufen der jeweiligen Support-Fenster durch neuere Versionen ersetzen.

In Bezug auf die gewünschte Verkürzung von Wartungsarbeiten machen sich vor allem die reduzierte Anzahl von Komponenten sowie die Weiterentwicklung der Administrationswerkzeuge bemerkbar. Es sind weiterhin alle Wartungstätigkeiten notwendig, doch diese müssen nicht mehr so häufig wie bisher durchgeführt werden. Auch ein „Altenteil“ ist technisch auf die gleiche Art wie bisher realisierbar, da für alle Altsysteme entsprechende Betriebssysteme auf der RISC-Plattform verfügbar sind.

### Szenario 2: Oracle Solaris-Zonen und Oracle SuperCluster

Das alternative Szenario besteht aus zwei Elementen. Zum einen die Farm von x86-Server auf Basis von Solaris-Zonen für Applikationsserver und kleinere Stand-Alone-Systeme und zum anderen der Aus-

tausch der bisher genutzten SPARC-Server durch einen Oracle SuperCluster, der mit der integrierten Exadata-Technologie die Datenbankzugriffe beschleunigt (siehe Abbildung 3).

Solaris-Zonen kamen bislang bei GerCos in der SAP-Umgebung noch nicht zum Einsatz, obwohl sie seit Solaris 10 3/05 auch für die Verwendung mit SAP freigegeben sind. Im Grundprinzip stellen Oracle Solaris-Zonen eine Virtualisierungsebene für Anwendungen dar, die oberhalb des Solaris-Kernels realisiert ist. Sie stehen auf allen verwendeten Systemen zur Verfügung, somit können sowohl in der x86-Farm als auch im SuperCluster die gleichen Technologien zur Virtualisierung genutzt werden. Dies vereinfacht Wartungsprozesse und reduziert die Fehleranfälligkeit.

Solaris-Zonen unterscheiden sich in nicht-globale und globale Zonen. Nicht-globale Zonen sind voneinander isoliert, teilen sich jedoch gemeinsam die globale Zone. Diese enthält neben dem Betriebssystem-Kernel die Device-Treiber, die Devices, das Memory-Management-System, die Filesystem-Treiber und in vielen Fällen auch den Netzwerk-Stack. Funktionalitäten des Ressourcen-Managements begrenzen den Ressourcen-Verbrauch der einzelnen Zonen. Lediglich die Ressourcen, die durch die globale Zone für eine nicht-globale Zone bereitgestellt

sind, können durch diese genutzt werden. Durch diesen hohen Anteil von gemeinsam genutzten Komponenten erzeugen Solaris-Zonen praktisch keinen Overhead, was auch im Rahmen von SAP-SD-Benchmarks bewiesen wurde (siehe „www.sap.com/benchmark“).

Solaris-Zonen bedeuten eine noch weitergehende Reduktion der Anzahl der VMs, da technisch die gesamte Leistung eines Servers ohne Overhead für die VM genutzt werden kann. Somit ergibt sich im vorliegenden Fall aufgrund der Festlegung auf eine bestimmte maximale Servergröße, um bei einem Ausfall nicht zu viel Leistung zu verlieren, eine maximale Größe von 60.000 SAPS.

Bei der vorgesehenen Nutzung von Solaris-Zonen auf den SuperCluster-Systemen ergibt sich auch dort eine Reduktion der Anzahl der Partitionen. Durch den geringen Aufwand zur Wartung einer Solaris-Zone – verglichen mit einem vollständigen Betriebssystem – kommen weitere Vereinfachungseffekte zum Tragen. In Summe ergibt sich somit gegenüber dem Szenario 1 eine deutliche Reduktion der Anzahl der zu konfigurierenden Elemente und somit eine Reduktion der Komplexität.

Dem gegenüber steht allerdings ein Aufwand für die Standardisierung des Betriebssystems. Der Wechsel des Betriebssystems im Bereich der VMs erscheint nur auf den ersten Blick schwierig. Gerade

aber bei standardisierten Applikationsumgebungen, die den Großteil der Landschaft ausmachen, ist dieser Aufwand überschaubar. Auch gestalten sich die Prozesse zur Wartung von SAP-Systemen auf allen freigegebenen Betriebssystemen sehr ähnlich, sodass nur geringe Änderungen an Prozessen notwendig sind.

Gegenüber dem Szenario 1 ist auch der Storage-Layer verändert. Statt durchgängig auf SAN-Storage zu setzen, nutzen die Performance-kritischen Systeme den Exadata-Storage des Superclusters, der optimale Performance für die Produktion sicherstellt. Für unkritische Systeme sowie Shared-SAP-Filesysteme wird eine ZFS-Storage Appliance genutzt, die mit einer auf Performance ausgelegten Bestückung mit Flash-Devices immer noch die bisher genutzte Infrastruktur übertrifft.

Die Storage-Infrastruktur wird durch diese Optimierung für die Oracle-Datenbanken in den Exadata Storage Cells leistungsfähiger als die SAN-Architektur in

Szenario 1. Gleichzeitig vereinfacht sich auch ihre Wartung. Der hohe Grad an Integration in den Datenbankbetrieb erfordert beim Exadata-Storage sehr wenig eigenen Administrationsaufwand. Patch-Arbeiten werden simultan mit denen der Datenbank durchgeführt, die angewendeten Patches sind Teil der Datenbank-Patches. Zudem ist die Provisionierung des Storage für eine Datenbank einfacher.

Auf Server-Ebene unterscheidet sich Oracle SuperCluster von den im Szenario 1 eingesetzten SPARC-Servern vor allem durch die hohe Vorintegration und die noch weitergehenden Vereinfachungen beim Patchen durch vorgefertigte und an SAP angepasste Patch-Bundles. Diese schlagen vor allem im Betrieb, aber auch in der Implementierungsphase zu Buche.

In Bezug auf die gewünschte Verkürzung von Wartungsarbeiten machen sich sowohl die reduzierte Anzahl von Komponenten (der Hypervisor fällt weg) als auch das Wartungsmodell von Sola-

ris bemerkbar. Die Verwendung unterschiedlicher Boot-Environments ermöglicht komplexe Wartungsarbeiten, etwa ein Betriebssystem-Upgrade der globalen und nicht-globalen Zonen während normaler Arbeitszeiten vorzubereiten, um es dann mit einer sehr kurzen Downtime live zu schalten. Dazu wird ein neues Boot-Environment erzeugt, während das alte noch produktiv genutzt wird. Im neuen Boot-Environment werden dann alle Änderungen, Patches und sonstigen Arbeiten durchgeführt, ohne die Produktion zu beeinträchtigen. Zu einem späteren Zeitpunkt schaltet man dann mit einem einzigen schnellen Reboot auf das neue Boot-Environment um. Zudem bietet es in jedem Fall ein einfaches und bewährtes Verfahren für ein Rollback, falls unvorhergesehene Schwierigkeiten während des Umschaltens entstehen. Das alte Boot-Environment bleibt erhalten und kann bei Problemen direkt wieder aktiviert werden. Dieses Verfahren kommt in allen Teilen

**Sie wollen wissen. Sie wissen was. Wir wissen das.**

**dbi** InSite  
Workshops

Insider-Wissen von IT-Experten: Unsere massgeschneiderten Workshops für Oracle, SQL Server, MySQL, Linux & mehr.

Phone +41 32 422 96 00 · BaselArea · Lausanne · Zürich

[dbi-services.com/InSite](http://dbi-services.com/InSite)



Infrastructure at your Service.

**dbi** services

der Architektur zu tragen, sodass identische Prozesse für die gesamte Landschaft genutzt werden können.

Auch in Sachen Lizenzmanagement spiegelt sich der reduzierte Overhead der Solaris-Zonen wider. Die Landschaft benötigt weniger Lizenzen als die VM-Landschaft, auch weil Solaris-Zonen als Lizenzgrenzen im Sinne der Oracle-Partitionierungsregeln gelten. Somit ist mit geringeren Ausgaben für Lizenzen und Softwarewartung zu rechnen. Das „Altenteil“ ist technisch ebenfalls mit Solaris-Zonen auf dem SuperCluster realisierbar, da in Solaris-10-Zonen ein älteres Betriebssystem für die „Methusalems“ bereitgestellt werden kann.

### Bewertung der Szenarien

Zur Bewertung der Alternativen wurden vier Kriterien herangezogen:

- Reduktion der Anzahl der genutzten physischen Systeme
- Reduktion der Komplexität der Landschaft
- Erfüllung zusätzlicher Anforderungen wie Reduktion der Anzahl der Downtimes, Leistungssteigerung des I/O-Systems, Wartungsarbeiten ohne Einfluss auf Produktion sowie das Konzept für die Altsysteme
- Zur Reduzierung der Supportkosten durch intelligentes Lizenzmanagement ist eine TCO-Analyse durchgeführt worden, die auch die übrigen Aspekte betrachtet

Die Anzahl der genutzten physischen Systeme wird bei beiden Szenarien reduziert, mit einem leichten Vorteil für Szenario 2 (14 beziehungsweise 16 Systeme). Aufwändiger gestaltet sie die Bewertung der Komplexität. Dabei wurden zwei getrennte Bewertungen für die Komplexität des späteren Betriebs und die Komplexität der Implementierung herangezogen. Die Anzahl und Ähnlichkeit der Configuration Items wird als Maß der Komplexität des späteren Betriebs herangezogen:

- Die heutige Landschaft besteht aus etwa 890 Configuration Items, deren Ähnlichkeiten recht beschränkt sind. Dadurch ergibt sich eine Bewertung der Komplexität des Betriebs von 590 Punkten.

- Für Szenario 1 wurden in Summe etwa 670 Configuration Items ermittelt, die aber häufig recht ähnlich sind. Dadurch ergibt sich in der Metrik der GerCos eine Punktzahl von 360.
- Für Szenario 2 wurden insgesamt etwa 480 Configuration Items ermittelt, wobei der Ähnlichkeitsgrad etwas höher ist als im ersten Fall. Hier wirken sich vor allem das Fehlen des Hypervisors in der x86-Serverumgebung sowie die Standardisierung des Betriebssystems aus. Somit ergibt sich eine Bewertung von 250 Punkten.

Die Komplexität der Migration wird über die Anzahl der zu verändernden Configuration Items berechnet. Diese ist bei Szenario 2 höher, die angesprochene Standardisierung der Betriebssysteme sorgt hier für einen größeren Aufwand. Die Kriterien nach einer Reduktion der Anzahl der Downtimes sowie der Durchführung von Wartungsarbeiten ohne Einfluss auf die Produktion erfüllt Szenario 2 besser als Szenario 1. Hier kommen zum einen die oben beschriebenen Verfahren unter Nutzung von Solaris Boot Environments zum Tragen, die die Wartungsarbeiten vereinfachen. Zum anderen sind die auf den SAP-Betrieb abgestimmten Bundle-Patches eine Unterstützung für die GerCos. Das Kriterium „Leistungssteigerung für Datenbanken“ entscheidet ebenfalls Szenario 2 für sich, wegen der dort genutzten Exadata-Features. Diese versprechen gegenüber klassischen SAN-Infrastrukturen ein deutliches Mehr an Performance.

Eine TCO-Analyse für eine Laufzeit von drei Jahren wurde zur Unterstützung der Entscheidung durchgeführt, um Aspekte wie Wartungskosten oder die Vereinfachungen der Wartungsarbeiten zu quantifizieren (siehe Abbildung 4).

Die Analyse zeigt einen Vorteil für das Szenario 2. Die Anschaffungskosten für Hardware und Software liegen bei beiden Szenarien auf ähnlichem Niveau, die Unterschiede schlagen sich in drei Bereichen nieder:

- Szenario 2 besitzt durch den kleineren Hardware-Footprint geringere Software-Wartungskosten. Auch Storage-spezifische Softwareprodukte sowie der Hypervisor auf x86-Ebene fallen weg.
- Szenario 1 hat Vorteile bei der Migration der Systeme. Diese entstehen, da

in Szenario 2 auch heterogene SAP-Migrationen durchgeführt werden müssen.

- Szenario 2 hat durch die Komplexitätsreduktion verringerte Operating-Kosten. Dies kommt im Falle der GerCos vor allem daher, dass viele Aufgaben von Wochenenden in die Woche zurückverlegt werden können.

Die hier überblicksartig dargestellte TCO-Studie zeigt vor allem aber auch, dass selbst, wenn die zu erwartenden Einsparungen durch die Vereinfachung des Operating nicht im vollen Umfang eintreten, die Vorteile für Szenario 2 überwiegen (siehe Abbildung 5).

### Fazit und Ausblick

Durch die Entscheidung für das zweite Szenario ist man bewusst den Weg gegangen, eine einmalig aufwändigere Migration durchzuführen, um anschließend einen deutlich vereinfachten Betrieb zu haben. Gleichzeitig wurden alle Ziele erreicht und die Wünsche an das neue Konzept umgesetzt. Momentan ist der reale Kunde hinter dieser Fallstudie gerade bei der Umsetzung. Zunächst geht die Applikationsserver-Farm in Betrieb, nach deren Fertigstellung man die SPARC-Infrastruktur angehen wird. Dort startet ein Assessment, um zunächst den Performance-Vorteil des SuperClusters zu evaluieren.



Jan Brosowski  
jan.brosowski@oracle.com



# Oracle Database Appliance

## – ein Deep Dive

David Hueber, dbi services

Im Jahr 2011 wurde die Oracle Database Appliance als neues Engineered System eingeführt. Die für Datenbanken-Konsolidierung und Hochverfügbarkeit gedachte Plattform ist so gut wie eine „Plug and Play“-Lösung. Dieser Artikel gibt einen Überblick über die ODA-Architektur und -Basisnutzung sowie Schlüssel, um die Black Box zu umgehen und von der vollen Leistung der Lösung zu profitieren.

Die Oracle Database Appliance (ODA) gehört dem Portfolio der Engineered Systems an. Von einer „low-cost Exadata“ ist definitiv nicht die Rede, denn Exadata steht für höchste Performance und Volumen. ODA wurde vielmehr als einfache, zuverlässige und verfügbare Lösung zur Konsolidierung von Datenbanken entwickelt.

Das System ist auf einer vollredundanten Hardware aufgebaut. Sie beinhaltet ein neues Framework, um Einrichtung und Management von Datenbanken zu vereinfachen, sowie auch einige (aber nicht alle) Technologien zur Hochverfügbarkeit wie beispielsweise RAC One Node und RAC. Außerdem ist die ODA die erste Plattform mit einem „Pay as you grow“-Lizenzmodell für Oracle-Datenbanken.

### Die ODA-Architektur

Seit ihrer Version X3-2 (Q3 2013) ist die Oracle Database Appliance auf drei individuellen Komponenten aufgebaut: zwei Server und ein Speicher-Array (siehe Abbildung 1). In ihrer letzten Version (ODA X4-2) enthält jeder Server:

- 2 Intel Xeon E5-2697v2 12 Cores
- 256 GB Memory
- 2 HDD SAS 10k 600 GB (gespiegelt in RAID 1, für „boot“ und Binaries benutzt)
- 2 SAS HBA Adapters
- 2 Twinnax 10 GB Interconnect
- 4 Ethernet 100 MB Ports

Das Speicherarray beinhaltet 24 Disks, wie folgt verteilt:

- 20 HDD SAS 10k 900 GB (benutzt für Data und Recovery Files)
- 4 HDD SSD 200 GB – (benutzt für Redo Logs und Control Files)

Die komplette Hardware-Architektur von ODA, einschließlich Lüfter und Netzteile, ist voll redundant. Der Speicherzugriff erfolgt durch zwei getrennte Controller, die



Abbildung 1: ODA-Frontplatten-Ansicht

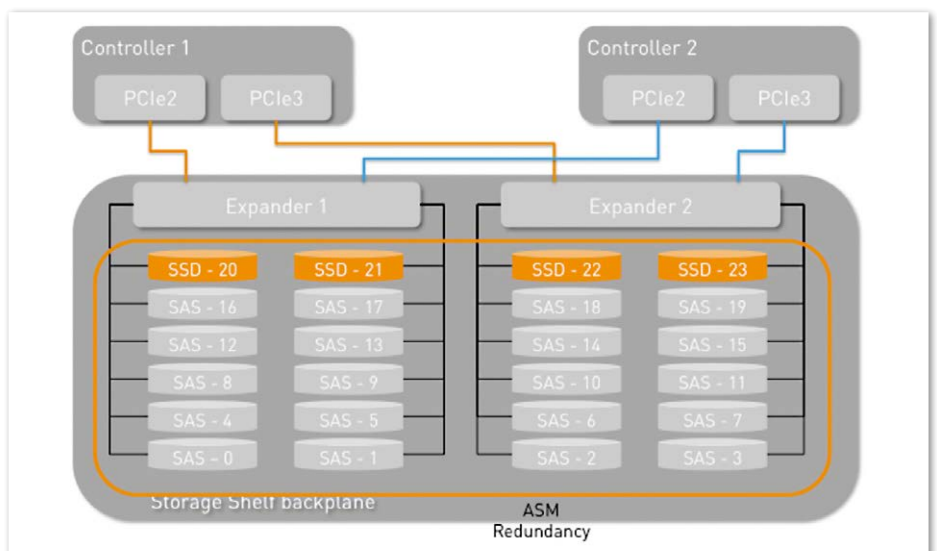


Abbildung 2: ODA-Speicherzugriff-Architektur

mit zwei Expandern innerhalb des Speicherarrays verknüpft sind. Die Daten-Redundanz erfolgt durch Oracle Automatic Storage Management (ASM). Bei Ausfall einer der Komponenten wird der Speicherzugriff weiter gewährt (siehe Abbildung 2). Hinsichtlich der Speicherkapazität werden zwei Parameter berücksichtigt:

- ASM Redundancy
- Backup storage location

Es steht zur Wahl, entweder den Modus „Normal Redundancy“ oder „High Redundancy“ vom ASM zu nutzen, also entweder die Originalspeicher-Kapazität von 9 TB bis 18 TB (Normal Redundancy) oder darunter bis 6 TB (High Redundancy). Zum anderen wird die Speicherkapazität direkt davon beeinflusst, ob das Backup der Datenbanken auf der ODA selbst gespeichert ist (in der Fast Recovery Area) oder auf einem externen Speicherort wie zum Beispiel Tapes oder NFS-Shares. Je nach Wahl wird das ASM-Disk-Group-Sizing angepasst (siehe Tabelle 1).

Demnach wird festgestellt, dass die ODA-Speicherkapazität für Oracle-Datenbanken

Backup Modus	+DATA (TB)	+RECO (TB)	+REDO (GB)
<b>Normal Redundancy</b>			
Local Backup	3,6	4,5	248
External Backup	7,2	0,98	248
<b>High Redundancy</b>			
Local Backup	2,4	3	248
External Backup	4,8	0,65	248

Tabelle 1

je nach Einstellung von Redundanz und Backup-Ort von 2,4 TB bis 7,2 TB schwankt. Anmerkung: Diese Einstellungen werden während der ODA-Einrichtung konfiguriert und lassen sich im Nachhinein ohne Neuinstallation der ODA nicht mehr ändern.

### Die Software-Architektur

Grundsätzlich kann jede ODA seit Version 2.5 nach zwei verschiedenen Architekturtypen installiert werden:

- Bare Metal
- Virtualized

Die Version 1.0 von ODA wurde mit dem Architekturtyp „Bare Metal“ entwickelt. Dieser ist der Datenbank ausschließlich vorbehalten. Er basiert auf Oracle Linux und beinhaltet Grid Infrastructure sowie Oracle Database (siehe Abbildung 3). Der in Version 2.5 eingeführte virtuelle Modus ermöglicht bei ODA das Betreiben von Oracle VM und infolgedessen von virtuellen Maschinen für Applikationen wie WebLogic, JD Edwards oder Tomcat neben einem virtuellen Server für Datenbanken (siehe Abbildung 4).

Bei dieser Architektur laufen alle Datenbanken in einer einzigen und dedizierten virtuellen Umgebung, nämlich im ODA BASE. Je nach Bedarf an Software-Komponenten können daneben zusätzliche virtuelle Maschinen eingerichtet werden. Die virtuelle Maschine ODA BASE hat innerhalb der gesamten Plattform die führende Funktion und wird zur Verwaltung weiterer Komponenten benutzt:

- Datenbanken
- Binaries
- virtuelle Maschinen

Verschiedene Templates stehen für Applikationen virtueller Maschinen bereits zum Herunterladen unter „[edelivery.oracle.com/linux](http://edelivery.oracle.com/linux)“ zur Verfügung. Zudem liefert Oracle eine WebLogic-Anwendung für ODA mit Java-basiertem Installationsprogramm, das eine hochverfügbare WebLogic-Struktur einschließlich Traffic Director automatisch aufstellt (siehe „<http://www.oracle.com/technetwork/middleware/weblogic-oda/downloads/index.html>“ und Abbildung 5).

### Der Appliance Manager

Für Oracle-DBAs ist die größte Veränderung bei der ODA der Appliance Manager

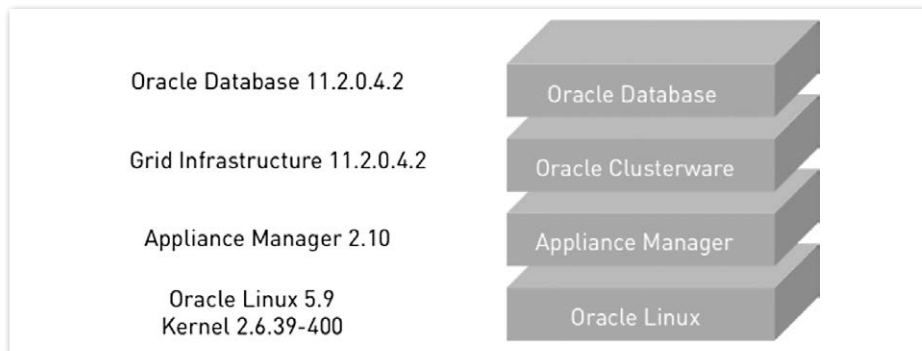


Abbildung 3: ODA – Bare-Metal-Software-Architekturtyp

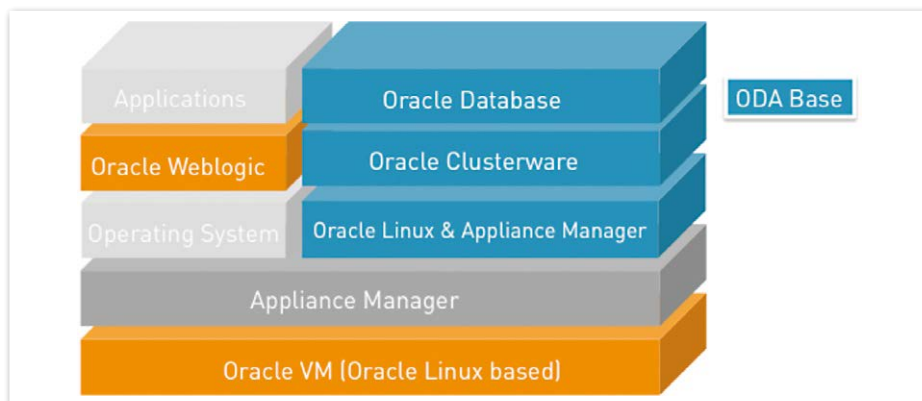


Abbildung 4: ODA – virtuelle Software-Architektur

„OAKCLI“, ein neues Command Line Interface, das folgende Funktionen ermöglicht:

- Hardware-Komponente auflisten
- Hardware-Komponente validieren und diagnostizieren
- Software installieren und upgraden
- Patches anwenden
- Datenbanken erstellen und entfernen
- Oracle Homes installieren und entfernen
- Virtuelle Maschinen einrichten und verwalten

Der OAKCLI-Befehl soll immer als Root-Benutzer vom ersten Knoten gestartet werden („node 0“), sonst gibt es eine Fehlermeldung (siehe Listing 1). Listing 2 zeigt die Festplattenüberprüfung mit OAKCLI und Listing 3 die Validierung der Speicherstruktur und Verkabelung.

## Verwaltung von Datenbanken unter Appliance Manager

Sobald die ODA konfiguriert wurde (Netzwerk, Hostname, Software-Einrichtung), können Datenbanken erstellt werden. Dies wird über den Appliance Manager realisiert, es muss also als „root“-Benutzer vom Knoten 0 gestartet werden (siehe Listing 4). Beim Erstellen von Datenbanken werden verschiedene Parameter bestimmt:

- Name der Datenbank (obligatorisch)
- Version der Datenbank
- Datenbank-„NLS“-Einstellungen

Der CREATE-Befehl startet einen Installationsassistenten, der Basis-Informationen wie „root“- , „oracle“- und „sysasm“-Benutzerkennwort benötigt, aber auch den Typ von Datenbanken (siehe Listing 5) sowie die „Database Class“ (siehe Listing 6).

Die „Database Class“ definiert in der Tat, welches „dbca template“ für das Erstellen der Datenbank benutzt werden soll. ODA wird mit acht Templates geliefert; weitere Templates sind nicht möglich (siehe Tabelle 2).

Diese Einstellungen werden nur der Information halber angegeben, insbesondere die Datenbanken-Anzahl oder die IOPS. In den nächsten Schritten geht es dann darum, die Umgebung anzupassen.

## Über die Limits hinaus

Die ODA bietet auf jeden Fall eine schnelle und einfache Methode, Datenbanken

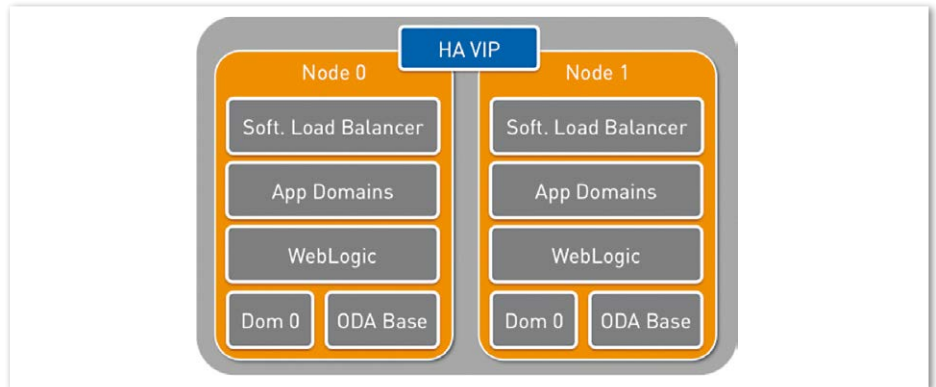


Abbildung 5: WebLogic-Anwendung für ODA

```
[oracle@dbi-oda2 ~]$ oakcli create database -d DBITEST2
sh: dmidecode: command not found
ERROR: 2014-01-06 22:53:25: Insufficient privileges to create the database

[root@dbi-oda2 ~]# oakcli create database -d DBITEST2
ERROR: 2014-01-06 22:54:18: oakcli create database should be invoked
from the first node
```

Listing 1

```
[root@dbi-oda1 ~]# oakcli show disk
```

NAME	PATH	TYPE	STATE	STATE_DETAILS
e0_pd_00	/dev/sda	HDD	ONLINE	Good
e0_pd_01	/dev/sdb	HDD	ONLINE	Good
e0_pd_02	/dev/sdaa	HDD	FAILED	DiskRemoved
e0_pd_03	/dev/sdab	HDD	ONLINE	Good
e0_pd_04	/dev/sdac	HDD	ONLINE	Good

Listing 2

```
root@dbi-oda2 ~]# oakcli validate -c storagetopology

It may take a while. Please wait...
INFO      : ODA Topology Verification
INFO      : Running on Node1
INFO      : Check hardware type
SUCCESS   : Type of hardware found : X3-2
INFO      : Check for Environment (Bare Metal or Virtual Machine)
SUCCESS   : Type of environment found : Bare Metal
INFO      : Check number of Controllers
SUCCESS   : Number of Internal LSI SAS controller found : 1
SUCCESS   : Number of External LSI SAS controller found : 2
INFO      : Check for Controllers correct PCIe slot address
SUCCESS   : Internal LSI SAS controller      : 50:00.0
SUCCESS   : External LSI SAS controller 0      : 30:00.0
SUCCESS   : External LSI SAS controller 1      : 40:00.0
INFO      : Check if JBOD powered on
SUCCESS   : 0JBOD : Powered-on
INFO      : Check for correct number of EBODS (2 or 4)
FAILURE   : Check for correct number of EBODS (2 or 4) : 1
ERROR     : 1 EBOD found on system, which is less than 2 EBODS with 1
JBOD
INFO      : Above details can also be found in log file=/
opt/oracle/oak/log/dbi-oda2/storagetopology/StorageTopology-
gy-2014-01-07-23:28:52_16620_29386.log
```

Listing 3

```
[root@ODADBI1-base ~]# oakcli create database -db TTOP1 -version 11.2.0.3.10 -params toplconf
```

Listing 4

```
Please select one of the following for Database Deployment [1 .. 3]:
1 => EE : Enterprise Edition
2 => RACONE
3 => RAC
1
Selected value is: EE
Please select one of the following for Node Number [1 .. 2]:
1 => ODADBI1-base
2 => ODADBI2-base
2
Selected value is: ODADBI2-base
```

Listing 5

```
Specify the Database Class (1. Medium 2. Others) [1]:2
Please select one of the following for Database Class [1 .. 8] :
1 => Very Very Small
2 => Very Small
3 => Small
4 => Medium
5 => Large
6 => Extra Large
7 => Extra Extra Large
8 => Extra Extra Extra Large
3
Selected value is: Small
```

Listing 6

```
[root@ODADBI1-base ~]# oakcli show db_config_params -detail
Available DB configuration files are:
default
DATABASE_BLOCK_SIZE => 8192
DATABASE_LANGUAGE => AMERICAN
DATABASE_CHARACTERSET => AL32UTF8
DATABASE_TERRITORY => AMERICA
COMPONENT_LANGUAGES => en
```

Listing 7

Setting	XXS	XS	S	M	L	XL	XXL	XXXL
CPU_COUNT	2	2	4	8	12	24	32	48
SGA (GB)	2	4	8	16	24	48	63	96
PGA (GB)	1	2	4	8	12	24	32	48
Processes	200	200	400	800	1200	2400	3200	4800
Redo (GB)	1	1	1	2	4	4	4	4
Nb DBs	24	24	12	6	4	2	1	1
IOPS	137	137	275	550	825	1650	3300	3300

Tabelle 2

in einer hochverfügbaren und effizienten Umgebung zum Laufen zu bringen. Da jede Anwendung unterschiedlich sein kann, können gewisse Anpassungen doch notwendig sein. Dafür lässt sich die Datenbank-Einstellung unter „OAKCLI“ vom DBA individuell verbessern.

Als Erstes nehmen wir die NLS-Einstellungen der Datenbank unter die Lupe. Diese werden von einer Konfigurationsdatei von OAKCLI während der Datenbank-Erstellung definiert. Eine Ausgangskonfiguration wird hierfür zur Verfügung gestellt (siehe Listing 7). Sollten andere NLS-Einstellungen nötig sein, kann individuell eine Konfigurationsdatei erzeugt und während der Datenbank-Erstellung verwendet werden (siehe Listing 8). Nach Erstellung der Datenbank sollten verschiedene Parameter überprüft werden, etwa die Einstellungen für „control files“ und „redo logs“ (siehe Listing 9).

Der Autor macht darauf aufmerksam, dass auch, wenn beide in der +REDO Disk Group (SSD Disks) gespeichert sind, weder der Control File noch die Redo Logs gespiegelt werden. Als Schutz gegen logische Korruption empfiehlt es sich, einen zusätzlichen Control File und einen Redo Member pro Redo Log Group hinzuzufügen. Dies sollte allerdings in der +REDO Disk Group angebracht werden, um Leistungseinbußen zu vermeiden.

Als Nächstes soll der Arbeitsspeicher-Verbrauch der Datenbank beobachtet werden. Seit Oracle 11g wurde das Automatic Memory Management (AMM – „memory\_target“) eingeführt. Dieses ist allerdings nicht mit „huge pages“ kompatibel, daher nutzt ODA das Automatic Shared Memory Management (ASMM – „sga\_target & pga\_aggregate\_target“). Standardmäßig kommt jede ODA aus dem Werk



mit einer „huge pages“-Einstellung von 50 Prozent des physischen Arbeitsspeichers des ODA Servers (siehe Listing 10). Wer über die volle Leistung des Arbeitsspeichers auf einer ODA verfügen möchte, folgt der MOS-Note „401749.1“, um die Größe der „huge page“ zu erweitern.

Sobald die Datenbanken erstellt und angepasst sind, können Prinzipien zur Hochverfügbarkeit eingeführt werden. Hierfür stellt ODA „out of the box“ entweder RAC-One-Node- oder RAC-Lösungen zur Verfügung. Diese Optionen sind unter der Enterprise Edition kostenpflichtig. Folgende Optionen sind ebenfalls verfügbar:

- Data Guard (in der Enterprise Edition beinhaltet)
- Failover Cluster (in der Grid Infrastructure beinhaltet)

Während Data Guard eine zweite ODA benötigt, kann Failover Cluster mit einem einzigen Befehl als „root“-Benutzer implementiert werden (siehe Listing 11). Dann kann die Instanz von einem Knoten zum anderen gewechselt werden (siehe Listing 12).

## Lizenzierung

Der Vollständigkeit halber möchte der Autor noch ein kurzes Wort zur Lizenzierung hinzufügen, ein spannendes Thema bei Oracle-Umgebungen. ODA ist die erste Plattform mit „Pay-as-you-grow“-Lizenzmodell. Bei Auslieferung ist sie für den Standard-Oracle-Prozessor lizenziert inklusive aller Regeln bezüglich Intel-Prozessoren (wie Core Factor). Dennoch besteht ein relevanter Unterschied zu den auf fremder Hardware basierenden Umgebungen (HP, IBM, DELL etc.), bei denen nicht alle verfügbaren Cores zwangsläufig lizenziert werden müssen. Von 24 verfügbaren Cores kann mit einer Minimum-Anzahl gestartet und von da an je nach Bedarf weitere Cores freigeschaltet werden. Aufstocken geht, reduzieren allerdings nicht. In diesem Zusammenhang besteht ein Unterschied zwischen der Bare-Metal- und der virtuellen Architektur, da die Mindestanzahl der zu lizenzieren Cores abweicht:

- *Bare Metal*  
Minimum von vier Cores pro Knoten, Erweiterung in Stufen von jeweils vier zusätzlichen Cores pro Knoten

```
[root@ODADBI1-base ~]# oakcli create db_config_params -conf dbprod1
Please select one of the following for Database Block Size [1 .. 4]:
1      => 4096
2      => 8192
3      => 16384
4      => 32768
2
Selected value is: 8192

Specify the Database Language (1. AMERICAN 2. Others) [1]:1
Selected value is: AMERICAN

Specify the Database Characterset (1. AL32UTF8 2. Others) [1]:2

Please select one of the following for Database Characterset [0 .. 10] :
0      => Others
1      => AL32UTF8
...
90     => UTF8
...
96     => WE8ISO8859P15
97     => WE8ISO8859P9
98     => WE8MACROMAN8S
96
Selected value is: WE8ISO8859P15

Specify the Database Territory (1. AMERICA 2. Others) [1]:1
Selected value is: AMERICA

Specify the Component Language (1. en 2. Others) [1]:1
Selected value is: en

Successfully generated the Database parameter file 'top1conf'

[root@ODADBI1-base ~]# oakcli create database -db DBPROD -params dbprod
```

Listing 8

```
SQL> show parameter control_files
NAME                TYPE                VALUE
-----
control_files       string              +REDO/dbitest/control01.ctl

SQL> select group#,members from v$log;
GROUP#MEMBERS
-----
1          1
2          1
3          1
```

Listing 9

```
[root@dbi-oda1 ~]# grep -i huge /proc/meminfo
HugePages_Total:      64000
HugePages_Free:       56200
HugePages_Rsvd:       393
HugePages_Surp:       0
Hugepagesize:         2048 kB
```

Listing 10

```
[root@ODADBI2-base bin]# ./crsctl modify res ora.dbprod.db -attr
"PLACEMENT=favored"
```

Listing 11

```
[root@ODADBI2-base bin]# ./crsctl relocate res ora.dbprod.db
CRS-2673: Attempting to stop 'ora.dbprod.db' on 'ODADBI2-base'
CRS-2677: Stop of 'ora.dbprod.db' on 'ODADBI2-base' succeeded
CRS-2672: Attempting to start 'ora.dbprod.db' on 'ODADBI1-base'
CRS-2676: Start of 'ora.dbprod.db' on 'ODADBI1-base' succeeded
```

Listing 12

- **Virtuelle Architektur**  
Minimum von zwei Cores pro Knoten, Erweiterung in Stufen von jeweils zwei zusätzlichen Cores pro Knoten

### Fazit

ODA ist eine ziemlich einfache, zuverlässige und verfügbare Plattform für das Kon-

solidieren von Oracle-Datenbanken. Sie bietet eine intuitive und integrierte Methode zu deren Erstellung und Verwaltung. Nichtsdestotrotz muss hinter die Kulissen geschaut werden, um zu sehen, wie die ODA funktioniert und wie sie anzupassen ist, um so viele Vorteile wie möglich zu erzielen.



David Hueber  
david.hueber@dbi-services.com

## Inventur im ERP-System

Christian Wille, PITSS GmbH

Oracle-Forms-Anwendungen haben in der Regel eine lange Historie hinter sich und viele Änderungen durchlaufen – umgesetzt von unterschiedlichen Entwicklern, sodass meist ein kaum überschaubares und nur noch schwer zu beherrschendes System übrig geblieben ist. Nur selten werden Qualitätsvorgaben berücksichtigt, die aufgrund ihrer Verwendung für mehr Klarheit sorgen. Ein Kundenbeispiel zeigt, wie man an diese Problematiken herangehen kann, um eine solch gewachsene Anwendung wieder wartbar und beherrschbar zu machen.

Ein deutsches Handelsunternehmen kaufte bei seiner Gründung im Jahr 2004 ein ERP-System eines anderen Unternehmens, um seine Geschäftsprozesse abbilden zu können. Analog zu der rasanten Unternehmensentwicklung auf mehr als 1.400 Filialen in fünf Ländern (Stand 2014) wuchs und veränderte sich auch das ERP-System. Zahlreiche Prozesse wurden optimiert, neue Prozesse geschaffen und alte abgeschaltet. Auch der Ersatz einzelner Komponenten, etwa des Logistiksystems oder der Verwaltung der Mietobjekte, durch Standardsoftware hatte große Auswirkungen auf das System, da diverse Objekte nicht mehr benötigt und durch neue Schnittstellen ersetzt wurden.

Ein solcher Produkt-Lifecycle ist bei Forms-Anwendungen durchaus verbreitet. Es gelingt in der Regel so gut wie nie, die nicht benötigten Komponenten aus dem System zu entfernen und die zunehmende Komplexität herauszuhalten. Das Unternehmen beauftragte eine detaillierte Code-Analyse, um einerseits einen Überblick über das bestehende System zu erhalten und andererseits nicht mehr benötigte Komponenten aus der Anwendung entfernen zu können.

### Vorgehensweise

Allein die Größe eines ERP-System erschwert eine allumfassende detaillierte Analyse und macht ein manuelles Vor-

gehen unmöglich. Daher wurden alle Komponenten über die API-Schnittstelle von Forms oder das XML-Format von Reports extrahiert und in einem Repository gespeichert. Dieses wurde über einen simplen Datenbank-Connect zusätzlich mit allen Objekten der Datenbank angereichert. Abschließend hat man die SQL-Aufrufe externer Programme als Textfiles ergänzt. Somit waren in dem Repository alle Objekte der Anwendung mit ihren Eigenschaften und ihrem Code gespeichert, um die Basis einer detaillierten Analyse zu schaffen.

In einem nachgelagerten Arbeitsschritt zerlegt ein Parser den gesamten Sourcecode der Anwendung bis hin zu einzelnen

LOCATION	TYPE	COUNT	LINES OF CODE	REFERENCED LINES OF CODE	COMMENT RATIO	PROGRAM UNITS	LINES PER PROGRAM UNITS
MODULE	FMB	219,00	228.347,00	7.792,00	21,17	7.475,00	31,00
	MMB	52,00	7.040,00	21,00	1.497,00	2.792,00	3,00
	OLB	2,00	2.808,00	0,00	0,00	56,00	50,00
	PLL	15,00	12.380,00	0,00	19,22	564,00	22,00
	RDF	836,00	135.794,00	0,00	5,20	13.468,00	10,00
DB	FUNCTION	192,00	9.850,00	0,00	1.013,00	192,00	51,00
	PACKAGE	164,00	9.113,00	0,00	54,46	816,00	11,00
	PACKAGE BO	162,00	95.232,00	0,00	13,88	1.408,00	68,00
	PROCEDURE	1.130,00	207.066,00	0,00	15,04	1.130,00	183,00
	SCRIPT	1.601,00	398.447,00	0,00	0,45	801,00	497,00
	TRIGGER	1.419,00	86.256,00	0,00	3,52	1.419,00	61,00
	VIEW	1.173,00	30.606,00	0,00	6,81	1.173,00	26,00
		6.965,00	1.222.939,00	7.813,00	220,81	31.294,00	84,42

Abbildung 1: Applikationsübersicht

Operatoren und Operanden. Unter der zusätzlichen Berücksichtigung der Eigenschaften einzelner grafischer Objekte lassen sich alle erdenklichen Abhängigkeiten innerhalb des Systems vollständig darstellen. Zusätzlich werden allseits anerkannte Software-Metriken wie Halstead oder McCabe der weit mehr als 200.000 Programmeinheiten berechnet.

Da das System nun in die kleinsten Einzelteile zerlegt ist, können auf dem Repository die vom Unternehmen gewünschten Analysen detailliert gezogen und das gesamte System bewertet werden. Die Analyse erstreckt sich dabei über einen allgemeinen Status quo des Systems, eine Bewertung des Systems unter Qualitätsaspekten und die Identifizierung von Optimierungs- und Entwicklungspotenzialen.

### Der Status quo

Dem Titel des Artikels folgend kann man die „Status quo“-Analyse als die Inventur des ERP-Systems bezeichnen. Mittels Kennzahlen, die über die gesamte Anwendung erhoben werden, entsteht eine Basis, die es erlaubt, das System und dessen Umfang sowie deren Komplexität zu beurteilen. Weitere tieferegehende Detail-Analysen lassen sich auf den ermittelten Kennzahlen aufsetzen und in Relation mit beliebigen Objekten bringen. Zuerst wird die Anzahl der einzelnen Objekte mit ihrer Zeilenanzahl, Comment Ratio, Anzahl von Program Units und der durchschnittlichen

Zeilenanzahl je Programmeinheit aufgeführt (siehe Abbildungen 1 und 2). Ergänzend wird die Anzahl an grafischen Objekten ausgewertet. Anschließend wird die Verteilung und Häufigkeit von Business Logik, der Programmzeilen, der Argumente, der Variablen sowie der verwendeten Exceptions erhoben.

Aus diesen Zahlen, die das Entwicklerteam nicht einmal annähernd schätzen konnte, lassen sich genaue Rückschlüsse über die Beschaffenheit der vorhandenen Architektur des Systems schließen. Dabei sind die Bestrebungen der letzten Jahre, nämlich Business-Logik und Logik im Allgemeinen in die Datenbank auszulagern, bereits deutlich zu erkennen. Es fällt auch auf, dass die einzelnen Forms-Module zwar recht viel Code beinhalten, aber in Verbindung mit der Anzahl grafischer Elemente eher überschaubar aufgebaut sind. Erstaunlich ist der geringe Anteil an Referenzen innerhalb der Forms-Module. Durchaus gut stellen sich die Comment-Ratio und die durchschnittliche Zeilenanzahl je Programmeinheit der Anwendung dar.

### Qualitätsanalyse

Eine Qualitätsanalyse nutzt und optimiert die Ergebnisse der Status-quo-Analyse. So wird die allgemeine Comment-Ratio durch eine Analyse einzelner Programmeinheiten ohne Inline-Dokumentation verfeinert. Sicherlich kann bei den in Forms zahlreichen sehr kleinen Triggern zur Na-

LIBRARY	Attached Library	17	0
FORM	Item	11.252	379
	Block	1.308	4
	Canvas	443	4
	Graphics	3	0
	Tab Page	232	0
	Popup Menu	75	0
	Form Module	219	0
	Object Group	78	4
	Record Group	437	0
	List of Values	228	0
	Property Class	1	0
	Menu	Popup Menu	782
Menu Module		52	0
Object Library	Item	17	0
	Block	5	0
	Library Tab	12	0
	Object Child	58	0
	Object Group	5	0
Report	Object Library	2	0
	Frame	5.706	0
	Group	2.239	0
	Section	2.508	0
	Cross Product	24	0
Report Module	836	0	

Abbildung 2: Übersicht GUI-Objekte

vigation auf eine Inline-Dokumentation verzichtet werden. Mit steigender Anzahl an Statements und damit zunehmenden Entscheidungswegen wird es jedoch unerlässlich, diese entsprechend zu dokumentieren.

Die vom Kunden eingeführten Namenskonventionen für Programmeinheiten und Datenbankobjekte wurden ebenfalls über das gesamte System auf Einhaltung geprüft. Sie dienen dazu, die Lesbarkeit des Sourcecodes zu erhöhen. In Anbetracht

des Zeitraums seit deren Einführung wurde bereits ein nennenswerter Teil der notwendigen Umbenennungen erreicht. Die Objekte, die den Regeln nicht entsprechen, sind dank erfolgter Analyse dokumentiert und zur Änderung aufgelistet.

Eine weitere interessante Aussage zur Qualität des Systems ergibt sich aus der Berechnung der statischen Software-Metriken jeder Programmeinheit. Neben der Anzahl der Statements werden auch die allseits anerkannten Metriken zur zyklomatischen Komplexität von McCabe, Halstead und der gemischte Metrik-Maintainability-Index berechnet. Diese Kennzahlen geben klare Anzeichen dafür, wie wartbar und testbar die einzelnen Programmeinheiten sind.

Insgesamt ergab sich anhand der prozentualen Verteilung von wartbaren bis hin zu nicht wartbaren Programmeinheiten ein Gesamtbild der Qualität des ERP-Systems. Nach den Metriken sollten nicht wartbare Programmeinheiten in den nächsten Entwicklungszyklen überarbeitet und zerlegt werden, um die Qualität zu erhöhen. Da Software-Metriken in Forms, nicht zuletzt wegen der proprietären Programmstruktur, in PL/SQL keine so große Beachtung finden wie in objektorientierten Programmiersprachen, bieten die Kennzahlen bei dem Unternehmen analog zu fast allen anderen Systemen eini-

gen Verbesserungsspielraum (siehe Abbildung 3).

Zu den Qualitätsanalysen werden neben tabellarischen und grafischen Übersichten auch detaillierte Listen mit Objekten und Programmeinheiten erstellt, die die gewünschten Prüfkriterien nicht erfüllen. So ist es dem Unternehmen möglich, die Qualität kontinuierlich zu erhöhen.

### Die Entwicklungsanalyse

Nach den bewertenden Analysen des Status quo und der Qualität identifiziert die Entwicklungsanalyse weitere Verbesserungspotenziale. Da das Unternehmen System-Komponenten deaktiviert und deren Forms-Code aus der Anwendung entfernt hat, bleiben ungenutzte Datenbank-Objekte und Reports zurück. Diese zu identifizieren, fiel dem Kunden aufgrund der engen Verzahnung unterschiedlicher Systemkomponenten sehr schwer.

Zuerst wurden ausgehend von Triggern oder SQL-Skripten, die einen Programmablauf starten, alle abhängigen Programmeinheiten sowie zugegriffenen Tabellen und Views identifiziert. Aus diesen Abhängigkeiten ergab sich eine Anzahl von Programmeinheiten und Tabellen, die nicht mehr für den Programmablauf relevant sind.

Bei der Analyse der nicht mehr aufgerufenen Programmeinheiten der Daten-

bank wurde ersichtlich, dass ein Viertel der Programmeinheiten nicht mehr in Verwendung war. Noch überraschender war, dass die Hälfte der Programmeinheiten in Forms nicht verwendet wurde. Dies ist durch die vorangegangene Auslagerung von Logik in die Datenbank erklärbar. Wenn die nicht mehr aufgerufenen Programmeinheiten aus dem ERP-System entfernt werden, wird sich die Gesamtzeilenzahl um fast 40 Prozent verringern. Dies wird die Übersichtlichkeit und Wartbarkeit des ERP-Systems verständlicherweise erhöhen. Einfach betrachtet bedeutet dies auch fast 40 Prozent weniger Aufwand für Wartung und Pflege.

Die Anzahl der nicht mehr verwendeten Tabellen und Views ist ebenfalls sehr hoch, da nicht mehr produktive System-Komponenten viele Stammdaten enthielten. Die außergewöhnlich hohe Anzahl unbenutzter Views deutet auf Fremdsysteme hin, wie die bei dem Unternehmen verwendeten Auswertungstools (Qlikview, Access), die über Views auf Systemdaten zugreifen, aber keinen Programmcode verwenden (siehe Abbildung 4).

Durch die Analyse kann die Anzahl der Tabellen um 40 Prozent gesenkt werden. Zusätzlich wird sich der Datenbestand des Systems durch das Entfernen der Tabellen um insgesamt drei Prozent reduzieren.

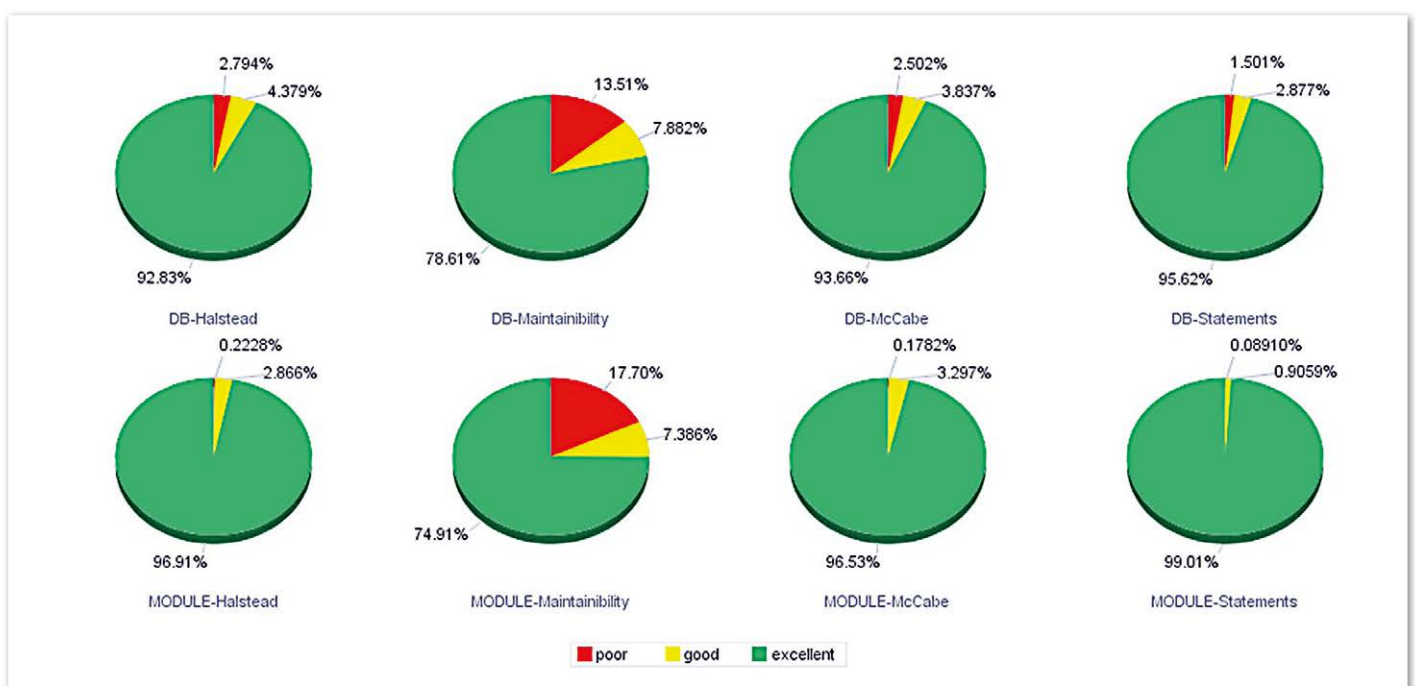


Abbildung 3: Übersicht der Software-Metriken



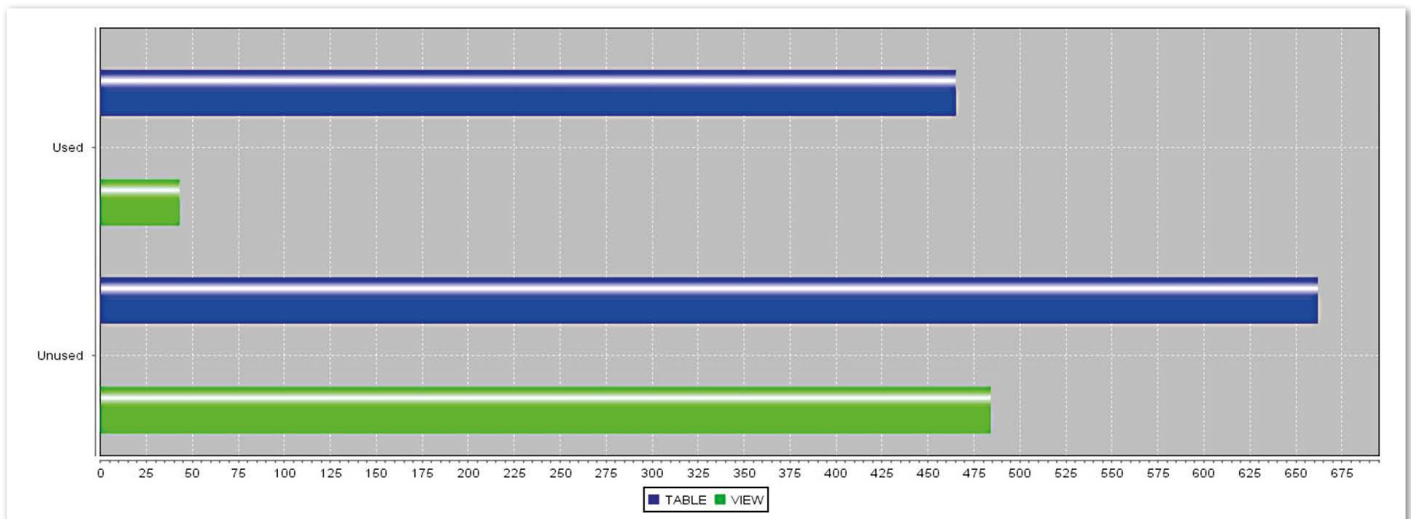


Abbildung 4: Übersicht der unbenutzten Tabellen

Da das Unternehmen lediglich Forms-Module aus der Anwendung entfernt hat, wurde eine Analyse der nicht mehr aufgerufenen Berichte erstellt. Auf der einen Seite hat man alle fest codierten Report-Aufrufe bewertet und andererseits über einen längeren Zeitraum die Aufrufe protokolliert, die den Reportnamen dynamisch ermitteln. Dabei konnten zwölf Prozent der Berichte identifiziert werden, die nicht mehr aus dem ERP System aufgerufen werden.

Im Detail wurden das System auf redundante Programmeinheiten geprüft und die Ergebnisse in Verbindung mit der allgemeinen Verwendung von Libraries und Packages gesetzt. Mehrfach identischer Programmcode in unterschiedlichen Sources sollte analog zu anderen Programmiersprachen gebündelt und referenziert werden. Einerseits ist dies besser für die Speicherauslastung und andererseits verringert es die Gefahr von unerwünschtem oder unterschiedlichem Verhalten. Besonders im Hinblick auf die geringe Anzahl von referenzierten Programmteilen überraschte das ERP-System mit einem sehr geringen Teil an Redundanzen.

Einen zusätzlichen Aspekt der Entwicklungsanalyse stellt eine Analyse Formsbedingter „Security Issues“ dar. Dabei wird das ERP-System nach Befehlen durchsucht, die eine SQL-Injektion oder JavaScript-Ausführung ermöglichen. Nicht enthalten, aber mindestens genauso wichtig zum Thema „Sicherheit“, sind natürlich auch administrative Einstellungen und Aufgaben. Das Unternehmen hat als

Ergebnis eine Auflistung aller sicherheitsrelevanten Stellen erhalten und kann anhand dieser Liste prüfen, ob Angriffe auf das System möglich sind.

Als letzten Teil der Entwicklungsanalyse wurde die Verwendung von globalen Variablen betrachtet. Sie sind ein beliebtes Mittel der Entwickler, um komplexere und verteilte Prozesse zu koordinieren, haben allerdings neben der Unübersichtlichkeit auch aus Aspekten der Laufzeit und des Speicherverbrauchs negativen Einfluss auf die Anwendung. Daher sollte sich die Verwendung auf ein Minimum beschränken. Erfreulicherweise erfolgt deren Benutzung bei dem Unternehmen sehr selten. Allerdings wurden 17 globale Variablen lediglich mit einem Wert initialisiert, dieser Wert jedoch nie verwendet. Diese Variablen-Definitionen sollten entfernt werden.

### Fazit

Mit der Analyse hat das Handelsunternehmen eine Inventur seines ERP-Systems durchführen lassen. Analog zur Inventur in einer Filiale oder dem Lager ist so ein detaillierter Überblick über die zentrale Unternehmensanwendung entstanden. Das reine Bauchgefühl des Entwicklerteams ist durch fundierte Kennzahlen ersetzt worden. Diese Kennzahlen schaffen die Basis für einen kontinuierlichen Verbesserungsprozess und ermöglichen nachhaltig, den eingeschlagenen Kurs nachvollziehbar und messbar zu verfolgen. Diverse in der Analyse erhobene Kennzahlen werden in ein regelmäßiges

Reporting an den CIO einfließen und so wird der Qualitätssicherungsprozess der bestehenden Arbeitsabläufe des ERP-Systems sichtbar vorangetrieben.

Sicherlich sind alle qualitätssichernden Maßnahmen in einem ersten Schritt mit Zusatzaufwand verbunden, der sich aber durch die Einsparungen in Wartung und Weiterentwicklung schnell amortisiert. Da das ERP-System allerdings auch einen zentralen Teil des wertschöpfenden Prozesses eines Unternehmens darstellt, ist dessen Qualität äquivalent zur Qualität der Produkte zu setzen.



Christian Wille  
cwille@pitss.com

# Exception Handling in ADF

Dr. Albert Angele, IKB Deutsche Industriebank AG

Oracle Application Development Framework (ADF) bietet ausreichend Möglichkeiten, Exceptions abzufangen und zu behandeln. Standardmäßig ist jedoch nicht in allen Schichten des Frameworks eine Fehlerbehandlung konfiguriert/implementiert. Dieser Artikel zeigt, wie Fehler aller Schichten des Frameworks abgefangen und einheitlich behandelt werden können. Zudem ist beschrieben, wie solche Fehler automatisch mitgeloggt und die Log-Einträge mit nützlichen Informationen angereichert werden können. Als Entwicklungsumgebung diente JDeveloper 11.1.1.5.

In ADF, einem modernen Framework zur Entwicklung von Web-Anwendungen, kommt eine Vielzahl verschiedener Technologien zum Einsatz. Die möglichen Fehlerquellen sind für Einsteiger oft unüberschaubar. Sie werden oftmals – wenn überhaupt – nur punktuell in Einzel-Implementierungen abgehandelt, deren Programmierung insgesamt sehr aufwändig ist und häufig kein einheitliches Verhalten zeigt. Die einzelnen Fehlerbehandlungsroutinen sind mit steigendem Umfang und Komplexität der Anwendung immer schwerer zu warten; sie verteilen sich gewöhnlich unübersichtlich auf viele Stellen des Codes.

Die Anwender werden bei einer Fehlermeldung typischerweise mit für sie verständlichen technischen Informationen konfrontiert, wohingegen für Entwickler, die den Fehler beheben sollen, nur unzureichende oder gar keine Informationen verfügbar sind. Oftmals ist die Applikation nach Auftreten eines Fehlers in einem instabilen Zustand und muss neu gestartet werden. Vor diesem Hintergrund soll das Exception Handling in ADF mindestens folgende Leistungsmerkmale bieten:

- Stabiler Zustand für die Applikation
- Sprechende Fehlermeldungen für den Anwender
- Einfache Zuordnung von Fehlerklassen zu Fehlermeldungen
- Mehrsprachige Fehlermeldungen
- Einheitliches Fehler-Reporting
- Eindeutige ID für jede Exception
- Präzise und umfangreiche Informationen für den Entwickler

- Automatisches Logging
- Geringer Implementierungsaufwand

In einem Projekt hat der Autor eine Error Library entwickelt, die alle notwendigen Komponenten für das Exception Handling enthält, um diese Eigenschaften zu realisieren. Diese Bibliothek wird in einer großen ADF-Anwendung aus vielen Teil-Applikationen eingesetzt. Um sie in einer neu erstellten ADF-Anwendung zu verwenden, ist ein einmaliger Aufwand von weniger als zehn Minuten erforderlich. Danach werden alle Exceptions abgefangen, gegebenenfalls automatisch mit aussagekräftigen Informationen geloggt, der Anwender mit einfachen, sprechenden Nachrichten informiert und die Applikation in einem stabilen Zustand gehalten. Darauf wird im Folgenden eingegangen. Der Artikel stellt eine Kurzfassung des Vortrags „All Inclusi-

ve: ADF Error Handling als Rundum-sorglos-Paket“ auf der DOAG Development 2014 und die Grundzüge der Implementierung vor.

## Das ADF-Schichtenmodell

Eine ADF-Applikation wird in einem Servlet Container ausgeführt (siehe Abbildung 1). Dieser kann auf Exceptions einer Applikation reagieren; unter anderem dann, wenn die Applikation ihre Fehler nicht selbst abfängt. Die Fehlerbehandlung durch den Servlet Container ist allerdings beschränkt, außerdem findet sie außerhalb der ADF-Applikation statt. Man kann jedoch zum Beispiel auf eine allgemeine Fehlerseite navigieren. Diese ist in der Datei „web.xml“ spezifiziert. Auf die Fehlerbehandlung durch den Servlet Container wird hier nicht näher eingegangen, da Exceptions schon in der ADF-Applikation behandelt werden

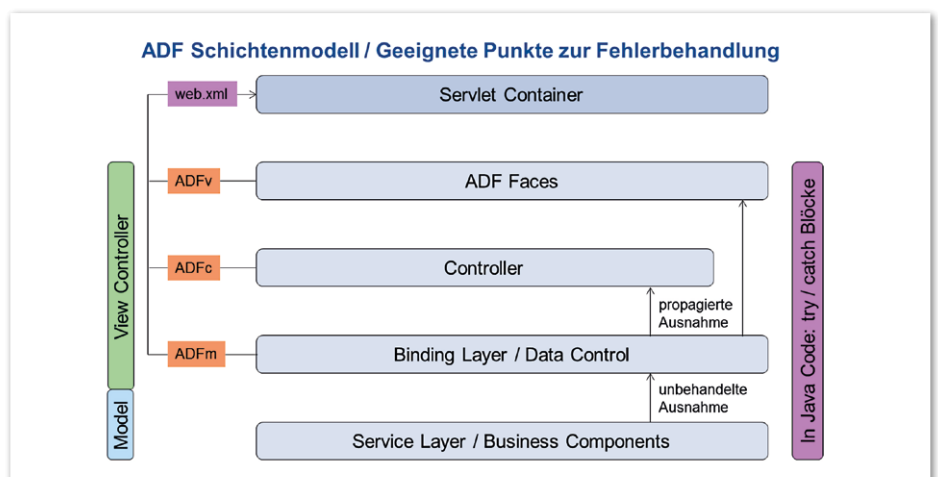


Abbildung 1: Schichtenmodell einer ADF-Anwendung im Servlet Container

## Hierarchie der Exception Klassen und Interfaces

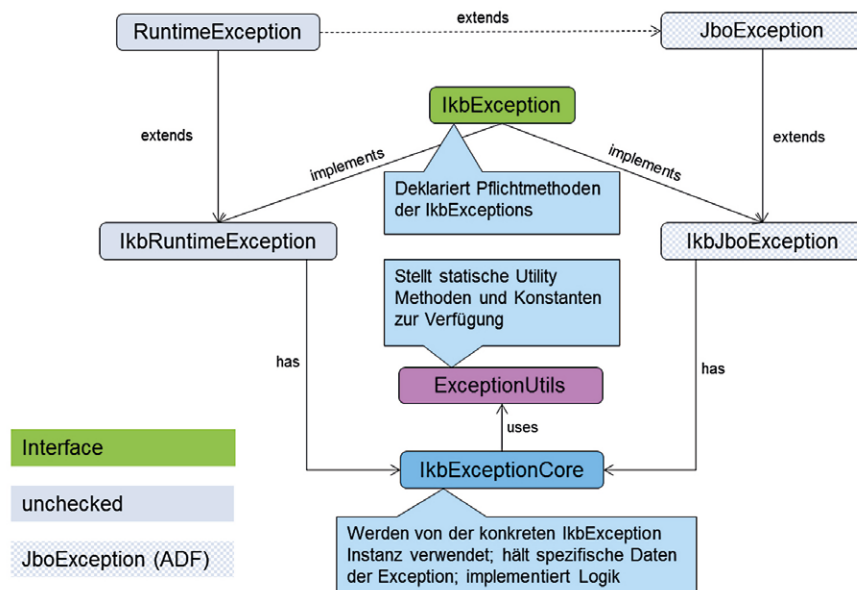


Abbildung 2: Hierarchie der Exception-Klassen und Interfaces

sollen. „ADFm“, „ADFc“ und „ADFv“ bezeichnen die möglichen Punkte, um eigene ADF Exception Handler für die Model(m)-, Controller(c)- und View(v)-Schicht zu implementieren. Darüber hinaus kann eine Fehlerbehandlung auch durch den Servlet Container („web.xml“) und im Java-Code („try/catch“-Blöcke) erfolgen.

Eine ADF-Applikation besteht gewöhnlich aus einem Model- und einem View-Controller-Projekt. Das Model-Projekt enthält den Service Layer, in dem Business Components (BCs) wie Application-Module und Entity- beziehungsweise View-Objekte zu finden sind, die mit der Datenbank kommunizieren. Treten dort Fehler auf (sowohl aus der Datenbank als auch aus den BCs selbst), werden sie an die aufrufende Schicht weitergeleitet. Hält man sich an die Best Practice des ADF, erfolgt der Zugriff auf den Service Layer über den Binding Layer. Dort besteht die erste Möglichkeit, einen zentralen Exception Handler zu implementieren, den „ADFm Exception Handler“. Dieser ist in der Standard-Konfiguration bereits aktiv und in der Klasse „DCErrorHandlerImpl“ implementiert. Diese Klasse stellt gegebenenfalls eine auftretende Exception in eine JboException und reicht sie an die aufrufende Schicht weiter.

In der Controller-Schicht („Task Flows“) kann ein weiterer Exception Handler implementiert sein („ADFc Exception Handler“). Er wird durch eine „Task Flow Activity“ aufgerufen, die als „Error Handler“-Task markiert ist. Sie repräsentiert einen Methoden-Aufruf einer Bean, wofür es keine Standard-Implementierung gibt.

Als dritte Möglichkeit kann in der View-Schicht („ADF Faces“) ein Exception Handler implementiert werden („ADFv Exception Handler“). Er reagiert auf diejenigen Exceptions, die der ADFc Handler nicht abfängt. Dies sind im Übrigen auch Exceptions, die in der Render-Response-Phase des JSF Lifecycle auftreten, da hier der ADFc Handler nicht mehr greift. Für den

ADFv Exception Handler existiert ebenfalls keine Standard-Implementierung.

Sind die genannten drei Exception Handler „ADFm“, „ADFc“ und „ADFv“ implementiert, können alle Exceptions, die innerhalb der ADF-Applikation auftreten, einheitlich behandelt werden. Deren Implementierung und Konfiguration wird in den nachfolgenden Abschnitten schematisch beschrieben. Darüber hinaus können im Java-Code aller Schichten „try/catch“-Blöcke zur Fehlerbehandlung genutzt werden. Hier bleibt es dem Entwickler überlassen, ob und wie er eine Exception behandelt oder weiterreicht.

Um den Umgang mit Exceptions zu standardisieren, insbesondere auch im

```

public interface IkbException {
    public String getMessage();
    public String getDisplayMessage();
    public String getExceptionId();
    public Calendar getErrorDateTime();
    public boolean isLogging();
    public void writeLogEntry(String loggingSourceClassName);
    public Throwable getCause();
    public void show();
    ...
}

```

Listing 1

```

public void show() {
    String errorMessage = getDisplayMessage();
    FacesContext fc = FacesContext.getCurrentInstance();
    FacesMessage facesMessage =
        new FacesMessage(FacesMessage.SEVERITY_ERROR, errorMessage, null);
    fc.addMessage(null, facesMessage);
    fc.renderResponse();
}

```

Listing 2

```

@Override
public void reportException(DCBindingContainer container, DCBindingContainer, Exception exception) {
    if (exception instanceof ValidationException || exception instanceof IkbException) {
        super.reportException(container, exception);
    } else {
        IkbJboException ikbJboException = new IkbJboException(exception);
        super.reportException(container, ikbJboException);
    }
}

```

Listing 3

Zusammenspiel mit den ADFx Exception-Handlern, ist es sinnvoll, eigene Exception-Klassen zu entwickeln. Diese sollen von bereits vorhandenen Exception-Klassen ableiten, um deren gebräuchliche Eigenschaften zur Verfügung zu stellen. Darüber hinaus implementieren sie einheitlich Fehler-Logik, Logging, Darstellung etc.

### Eigene Exception-Klassen

Der ADFm Exception Handler stellt auftretende Exceptions in eine „JboException“. Um keine unerwünschten Seiteneffekte zu erhalten, hat der Autor für diesen Handler eine Exception-Klasse entwickelt, die von „JboException“ ableitet, die „IkbJboException“ (siehe Abbildung 2).

Die View-Controller-Schicht benötigt nicht zwingend JBO-Klassen, sodass für diesen Fall eine Exception entwickelt wurde, die von „RuntimeException“ ableitet, die „IkbRuntimeException“.

In beiden Fällen handelt es sich um „unchecked“-Exceptions, sodass es keinen weiteren Programmieraufwand erzwingt, wenn sie geworfen werden.

Da beide Klassen grundsätzlich die gleiche Funktionalität mitbringen sollen, implementieren sie ein Interface „IkbException“ und nutzen eine Core-Klasse „IkbExceptionCore“, die die Logik der Exceptions implementiert und auf die im Adapter-Pattern

zugegriffen wird. „IkbJboException“ und „IkbRuntimeException“ stellen im Grunde also nur Hüllen dar, um die Eigenschaften und Funktionalitäten ihrer Oberklassen zu erhalten. Als weitere Klasse ist „IkbExceptionUtils“ implementiert, die statische Hilfsmethoden und Konstanten enthält.

Eine konkrete „IkbException“-Klasse bietet zwei Konstruktoren. Am Beispiel der „IkbJboException“ lauten sie:

- IkbJbo-Exception(Throwable throwable)
- IkbJbo-Exception(String message)

Der erste Konstruktor dient zum Aufnehmen gefangener Exceptions, der zweite zum Erzeugen neuer Exceptions unter An-

gabe einer Fehlermeldung, beispielsweise zum Werfen interner Fehler in selbst entwickeltem Code. Beim Erzeugen einer „IkbException“-Instanz werden klassenintern einige Eigenschaften initialisiert:

- Anhand der ursprünglichen Exception (siehe erster Konstruktor) wird ein lokalisierter kurzer Meldungstext ermittelt, den auch der Anwender erhält
- Eine eindeutige Exception-Id wird generiert, die ebenfalls dem Anwender angezeigt und auch im Logging verwendet wird
- Der aktuelle Zeitpunkt wird festgehalten
- Anhand der ursprünglichen Exception wird ermittelt, ob diese „IkbException“-Instanz automatisch geloggt wird

Diese Eigenschaften sind über geeignete Methoden zugänglich, die im Interface „IkbException“ deklariert sind (siehe Listing 1). Die konkreten Implementierungen der Methoden haben folgende Eigenschaften:

- `getExceptionId()`  
Liefert eine eindeutige Id für jede Exception
- `getErrorDateTime()`  
Liefert den Zeitstempel der Instanziierung der Exception-Klasse
- `getMessage()`  
Liefert eine lokalisierte sprechende Fehlermeldung, die in einem „ErrorResourceBundle“ hinterlegt ist
- `getDisplayMessage()`  
Wie „getMessage()“, jedoch angereichert mit der ExceptionId zur Darstellung für den Anwender
- `isLogging()`  
Flag, ob die Exception geloggt werden soll

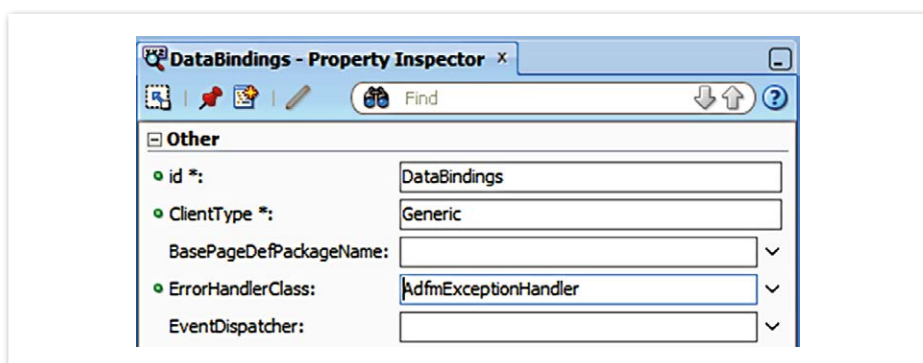


Abbildung 3: Konfiguration des ADFm Exception Handler in „DataBindings.cpx“ der ViewController-Projekte



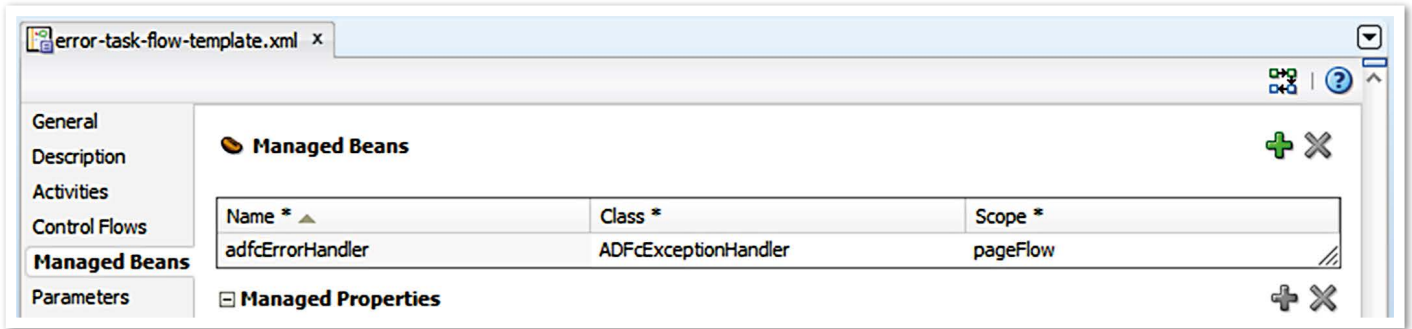


Abbildung 4: Die ADFc-Exception-Handler-Klasse wird im „error task flow template“ als „Managed Bean“ registriert

```
public String handleError() {
    ControllerContext cc = ControllerContext.getInstance();
    Exception exception = cc.getCurrentViewPort().getExceptionData();
    IkbException ikbException = null;
    if(exception instanceof IkbException) {
        ikbException = exception;
    } else {
        ikbException = new IkbRuntimeException(exception);
    }
    ikbException.writeLogEntry(this.getClass().getName());
    ikbException.show();
    cc.getCurrentRootViewPort().clearException();
    return "return";
}
```

Listing 4

- *writeLogEntry(String loggingSourceClassName)*  
Schreibt einen Logfile-Eintrag, der mit nützlichen Informationen angereichert ist und auch ein „StackTrace“ enthält. Als „loggingSourceClassName“ kann der Name der Klasse übergeben werden, der die „IkbException“ verarbeitet, etwa die ADFx-Exception-Handler-Klassen
- *getCause()*  
Ruft „Throwable.getCause()“ auf
- *show()*  
Zeigt die Fehlermeldung in einem Dialog an

Die Klasse „IkbExceptionCore“ enthält die Implementierungen dieser Methoden. Dort ist auch eine Mapping-Tabelle hinterlegt, die Fehler-Klassen einfachen Meldungstexten zuordnet. Wenn beispielsweise bei einer „SQLException“ der Text „Es trat ein Fehler in der Datenbank auf“ angezeigt werden soll, so lautet der Mapping-Eintrag „SQLException“ <-> ERRMSG\_DATABASE“, wobei „ERRMSG\_DATABASE“ der Key des „ErrorResourceBundles“ ist, der den gewünschten Meldungstext repräsentiert.

Dieses „ResourceBundle“ enthält auch den Key „ERRMSG\_STD“, der einen Standardtext repräsentiert. Dieser wird für alle Exceptions ausgegeben, für die kein explizites Mapping erfolgt ist. „IkbExceptionCore“ setzt auch die Eigenschaft „isLogging“ auf „true“, es sei denn, es handelte sich um eine „ValidationException“, bei der man Validierungsfehler nicht protokollieren will.

Zur Ermittlung des Logfile-Eintrages wird „getCause()“ der Exception beziehungsweise der „Causes“ so lange durchlaufen, bis „null“ erhalten wird. So kann der Logfile-Eintrag mit Informationen über jeden „Cause“ angereichert werden. Geeignete Methoden sind beispielsweise „getMessage()“ und „getStackTrace()“ der Klasse „java.lang.Throwable2“ sowie „getErrorCode()“, „getProductCode()“, „getSeverity()“ und „getDetailMessage()“ der Klasse „oracle.jbo.JboWarning“. Die ECID („execution context Id“ des WebLogic Servers) lässt sich über „weblogic.diagnostics.context.DiagnosticContextHelper.getContextId()“ ermitteln.

### Darstellung der Fehlermeldung

ADFc und ADFv Exception Handler führen die Methode „IkbException.show()“ aus.

Dadurch sollen der Fehlertext sowie die Exception-Id einer „IkbException“ in einem Dialog dargestellt werden, den der Anwender – gegebenenfalls nach Erstellung eines Screenshots – wieder schließen kann. Eine einfache Implementierung dieser Methode nutzt die Klasse „javax.faces.application.FacesMessage“ (siehe Listing 2).

Bei Bedarf kann man die Funktionalität der Darstellung erweitern, sodass beispielsweise eigene Dialoge verwendet werden, die dem Corporate Design entsprechen, oder der Anwender mehrere Buttons zum Schließen des Dialogs hat, die zusätzlich eine geeignete Navigation auslösen.

### Implementierung und Konfiguration der ADFx Exception Handler

Die „IkbExceptions“ kommen in den drei ADFx Exception Handlern zum Einsatz. Für jeden davon ist es notwendig, eine Java-Klasse zu erstellen und gegebenenfalls bestimmte Methoden zu überschreiben. Im Fall des ADFc Exception Handler erfolgt die Erstellung als Java Bean, die in einem TaskFlow (-Template) ausgeführt wird. Die Implementierung und die Verwendung (Konfiguration) sind nachfolgend für jeden Exception Handler einzeln beschrieben.

Die Eigen-Implementierung des ADFm Exception Handler leitet von „oracle.adf.model.binding.DCErrorHandlerImpl“ ab. Die wichtigste zu überschreibende Methode ist „reportException“. Hier wird geprüft, ob die aufgeworfene Exception in eine „IkbException“ übergeben wird. Bei „ValidationExceptions“ soll dies unterbleiben, da diese vom Framework in unveränderter Form verarbeitet werden sollen (siehe Listing 3).

Die Verwendung des ADFm Exception Handler ist in jedem ViewController-Projekt in der Datei „DataBindings.cpx“ konfiguriert (siehe Abbildung 3). Wählt man im Strukturfenster des JDevelopers „DataBin-

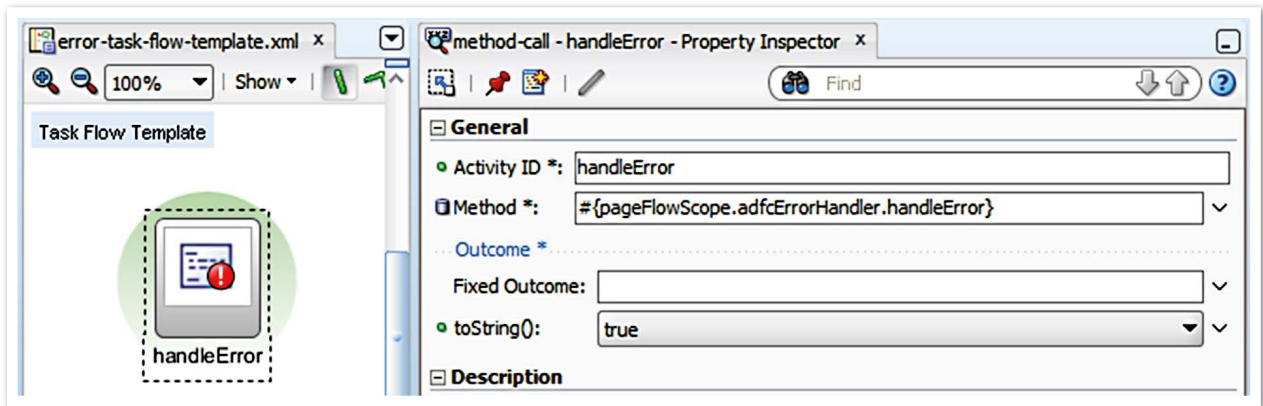


Abbildung 5: „ADFExceptionHandler.handleException()“ wird als „Method Call“ im „error task flow template“ aufgerufen und als „Exception Handler“-Activity markiert

dings.cpx“ aus, kann man im Properties-Editor den (voll qualifizierten) Namen der „ADFExceptionHandler“-Klasse als Error Handler eintragen.

Für den ADFc Exception Handler wird eine Java Bean mit beliebigem Namen erstellt (beispielsweise „ADFExceptionHandler.java“). Die Fehlerbehandlung erfolgt durch eine parameterlose Methode mit Rückgabe-Typ „String“. Dessen Name ist ebenfalls frei wählbar (etwa „handleError()“). Wie im vorherigen Beispiel wird die auftretende Exception gegebenenfalls in eine „IkbException“ übergeben. Hier werden außerdem automatisch ein Logfile-Eintrag erzeugt, die Fehlermeldung angezeigt sowie der Fehlerstack aufgeräumt (siehe Listing 4).

Nun erstellt man das „error task flow template“ und registriert die ADFc-Exception-Handler-Klasse als „Managed Bean“ (siehe Abbildung 4). „handleError()“ wird im „error task flow template“ als „Method Call“ aufgerufen und als „Exception Handler“-Activity markiert (Rechtsklick auf „Method Call“ -> „Mark Activity“ -> „Exception Handler“, siehe Abbildung 5). Der ADFc Exception

Handler wird in ADF-Applikationen dadurch verwendet, dass alle TaskFlows, die mit Exception Handling ausgestattet werden sollen, vom „error task flow template“ ableiten.

Der ADFv Exception Handler leitet von „oracle.adf.view.rich.context.ExceptionHandler“ ab. Es muss die Methode „public abstract void handleException(FacesContext p1, Throwable p2, PhaseId p3)“ implementiert sein, die analog zu „ADFExceptionHandler.handleError()“ erfolgen kann (siehe oben). Die Verwendung des ADFv Exception Handler wirkt auf eine gesamte ADF-Applikation und wird demzufolge nur einmal für jede ADF-Applikation konfiguriert:

1. Im Filesystem legt man im Verzeichnis „[application-root]/.adf/META-INF“ ein Unterverzeichnis „services“ an
2. Dort erstellt man eine Datei mit dem Namen „oracle.adf.view.rich.context.ExceptionHandler“
3. Der einzige Inhalt dieser Datei ist der voll qualifizierte Name der „ADFvExceptionHandler“-Klasse, also beispielsweise „ADFvExceptionHandler“

Im JDeveloper ist wird diese neu erstellte Datei unter Application Resources dargestellt (siehe Abbildung 6).

**Fazit**

ADF bietet ausreichend Möglichkeiten, um ein einheitliches und effektives Exception Handling zu implementieren. Dafür müssen drei Exception Handler („ADFm“, „ADFc“ und „ADFv“) für Model-, Controller- und View-Schicht implementiert sein. Es ist sinnvoll, in den Exception Handlern eigene Exception-Klassen zu verwenden, um eine einheitliche Logik und Darstellung der Fehlermeldungen zu gewährleisten und Wartungsaufwände durch die zentrale Implementierung zu minimieren.

Der anfängliche Entwicklungsaufwand ist überschaubar und zahlt sich erfahrungsgemäß schon nach kurzer Zeit aus, da die Anwendung eines zentralen Exception Handling pro ADF-Applikation nur noch wenige Minuten Implementierungs- beziehungsweise Konfigurationsaufwand erfordert.

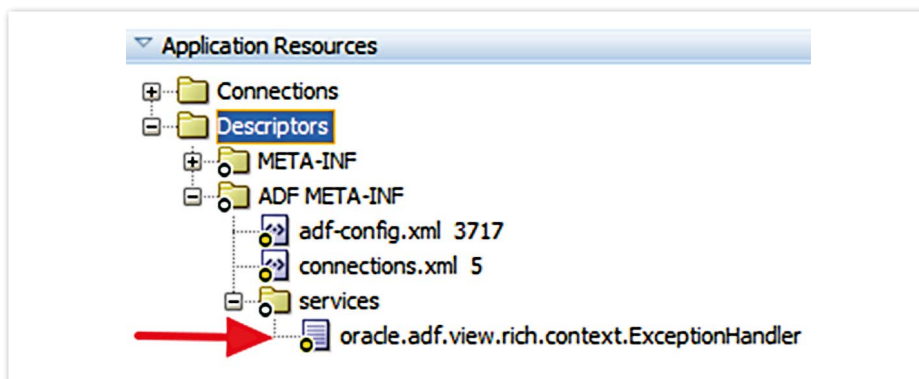
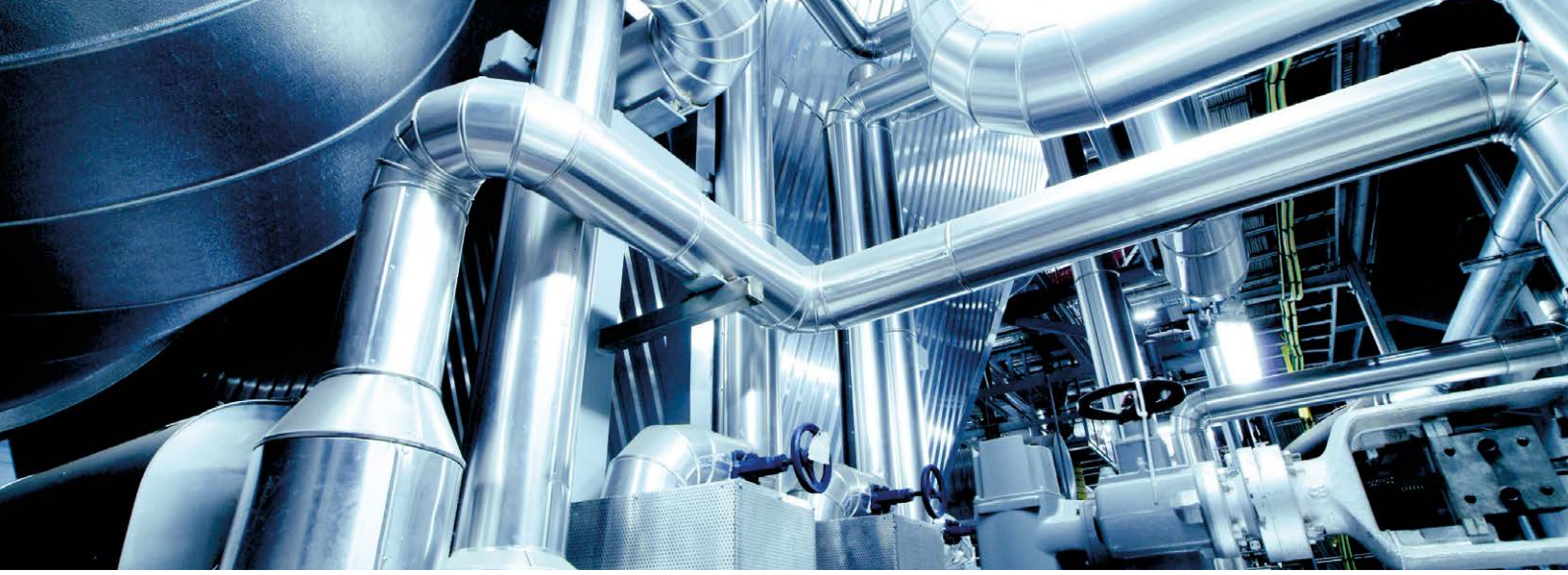


Abbildung 6: Der ADFv Exception Handler ist zentral für die gesamte ADF-Applikation definiert



Dr. Albert Angele  
albert.angele@ikb.de



# Database Resource Manager – Stand der Dinge

Ulrike Schwinn, ORACLE Deutschland B.V. & Co.KG

Schon seit der Version 8i ist es möglich, Ressourcen in der Datenbank mit der Enterprise Edition und dem Database Resource Manager zu verwalten. Damit lassen sich Datenbank-Ressourcen-Engpässe lösen, die beispielsweise mit hoher CPU-Last in Zusammenhang stehen.

Zu Beginn war die Resource-Manager-Funktion der Datenbank noch wenig bekannt und fand nur spärlichen Einsatz. Ein Grund dafür war die eingeschränkte Verwendbarkeit, die ausschließlich über PL/SQL-Packages und für dedizierte Datenbankuser-Sessions möglich war. Spätestens seit der Datenbank-Version 10g mit der Verfügbarkeit über die grafische Oberfläche des Enterprise Manager sowie der Erweiterung der Nutzung über Services, Module, Programme etc. ist dieser Grund passé. So können zum Beispiel CPU-Ressourcen für verschiedene Applikationen priorisiert, lang laufende Operationen unterbunden oder die Anzahl der Sessions begrenzt werden, um nur einige Beispiele zu nennen. Darüber hinaus wird mit der Oracle Database 12c und der neuen Multitenant-Architektur der Einsatz des Resource Manager immer wichtiger.

Mittlerweile verwenden immer mehr Anwender den Resource Manager und haben damit eine zusätzliche wichtige Ebene der Administration kennen und schätzen

gelernt. Nach einer Einarbeitungsphase, die sich durch vorhandene Skripte verkürzen lässt, sowie einer Testphase des Setups können durch den Einsatz eines entsprechenden Resource-Management-Plans drängende Fragen der Administration gelöst werden. Danach ist nur noch ein einfaches Monitoring von Zeit zu Zeit erforderlich. Auch intern findet der Database Resource Manager schon seit der Version 10g seine Anwendung, zum Beispiel in den sogenannten „Automated Maintenance Tasks“.

Um einen schnellen Einstieg in das Thema zu gewährleisten, kommen nachfolgend ein paar grundsätzliche Begriffe wie „Resource Consumer Group“, „Resource Consumer Mapping“ und „Resource

Plan“ zur Sprache. Im ersten Schritt werden Benutzergruppen nach unterschiedlichen Ressourcen-Anforderungsprofilen erzeugt; sie werden als „Resource Consumer Group“ bezeichnet. Es sind also Datenbankuser-Sessions, die als eine Einheit behandelt werden. Defaultmäßig existieren schon einige vordefinierte Resource Consumer Groups wie zum Beispiel die sogenannte „SYS\_GROUP“, die aus den Usern „SYS“ und „SYSTEM“ besteht. „OTHER\_GROUPS“ hingegen besteht aus allen anderen Usern, die keine Zuweisung zu einer Gruppe erhalten haben. Möchte man über die bestehenden Consumer Groups hinaus eigene Gruppen definieren, kann man das Kommando aus *Listing 1* verwenden.

```
dbms_resource_manager.create_consumer_group(  
  consumer_group => 'HPRIO_GR',  
  comment       => 'Sessions for HPRIO_GR');
```

*Listing 1*



Datenbankuser-Sessions können automatisch einer gewissen Consumer Group zugeordnet werden. Diese „Mapping Rules“ enthalten Regeln beziehungsweise Eigenschaften, nach denen die einzelnen Sessions den Gruppen initial zugeordnet sind. Ändern sich die Eigenschaften einer Session oder tritt eine Überschreitung einzelner Ressourcengrenzen auf, kann dies zu einem automatischen Wechsel in eine andere Gruppe führen. Darüber hinaus lässt sich ein Gruppenwechsel natürlich auch manuell anstoßen. Die wichtigsten Mapping Rules sind:

- Oracle-Datenbank-User (dbms\_resource\_manager.oracle\_user)
- TNS-Servicename oder Oracle definierte Services wie SYS\$USERS (dbms\_resource\_manager.service\_name)
- MODULE und ACTION (dbms\_resource\_manager.module\_name etc.)
- Client OS User (dbms\_resource\_manager.client\_os\_user)
- Client Program (dbms\_resource\_manager.client\_program)
- Client-Maschine (dbms\_resource\_manager.client\_machine)

Damit es zu keinen Überlagerungen der einzelnen Regeln kommt, werden diese in einem separaten Schritt priorisiert. Die folgende Prozedur (siehe Listing 2) bildet nun die Sessions des Service „S\_SERVICE1“ auf die Consumer Group „HP\_GROUP“ ab.

Für den Parameter „attribute“ können die in den Mapping Rules angegebenen Konstanten wie „dbms\_resource\_manager.oracle\_user“ etc. verwendet werden. Möchte man beispielsweise erreichen, dass der „APPDBA“-User der „SYS\_GROUP“-Gruppe angehört, ist das in Listing 3 gezeigte Kommando erforderlich.

Damit ein User oder eine Rolle in eine Consumer Group wechseln kann, ist ein zusätzlicher Prozedur-Aufruf erforderlich. Mit der Verwendung von „PUBLIC“ wird beispielsweise allen Usern erlaubt, in die Gruppe „HPRIO\_GR“ zu wechseln (siehe Listing 4).

Kern des Ressourcen-Managements ist der „Resource Manager Plan“. Resource-Pläne enthalten die Regeln (auch „Direktiven“ genannt), nach denen die Ressourcen an die Resource Consumer Groups aufgeteilt werden. Resource-Pläne sind dynamisch und können mit einem einfa-

```
execute dbms_resource_manager.set_consumer_group_mapping(
  attribute      => dbms_resource_manager.service_name,
  value          => 'S_SERVICE1',
  consumer_group => 'HP_GROUP');
```

Listing 2

```
dbms_resource_manager.set_consumer_group_mapping(
  attribute      => dbms_resource_manager.oracle_user,
  value          => 'APPDBA',
  consumer_group => 'SYS_GROUP');
```

Listing 3

```
dbms_resource_manager_privs.grant_switch_consumer_group(
  grantee_name  => 'public',
  consumer_group => 'HPRIO_GR',
  grant_option  => FALSE);
```

Listing 4

chen Befehl aktiviert beziehungsweise deaktiviert werden. Es gilt allerdings immer nur ein Plan zu einer Zeit. Resource-Pläne können aus einem einzigen Plan, also einer Ebene (auch „Single Level Plan“), oder auch aus mehreren Subplänen, das heißt aus mehreren Ebenen (auch „Multi Level Plan“) bestehen. Wie bei den Consumer Groups gibt es auch hier vordefinierte Resource-Pläne wie „DEFAULT\_MAINTENANCE\_PLAN“, der für die oben erwähnten Maintenance-Aufgaben zum Einsatz kommt.

Ein Code-Ausschnitt aus einem selbstdefinierten Resource-Plan (oder Subplan) könnte dann folgendermaßen aussehen (siehe Listing 5). Die Prozeduren „clear\_“, „create\_“, „validate\_“ und „submit\_pending\_area“ verwalten dabei einen temporären Arbeitsbereich für die Dauer der Resource-Management-Konfiguration. Die Änderungen sind so lange nicht sichtbar/wirksam, bis die Pending Area mit „submit\_pending\_area“ abgeschlossen worden ist. Hinweis: Die Verwendung eines temporären Arbeitsbereichs ist bei der Erzeugung von Consumer Groups oder Resource-Plänen obligatorisch. Aus Gründen der Übersichtlichkeit wurde diese Information in den übrigen Beispielen weggelassen.

Der Parameter „mgmt\_p1“ gibt den Prozentsatz an CPU an, der von der Con-

sumer Group alloziert werden kann. Das Maximum an CPU liegt natürlich bei 100 Prozent. Falls die CPU nicht oder nicht vollständig von einer Consumer Group verwendet werden kann, wird dieser Anteil an die anderen Gruppen weitergegeben. Seit der Version 11g R2 gibt es zusätzlich den sehr nützlichen Parameter „max\_utilization\_limit“. Damit kann man die Verwendung an CPU nach oben begrenzen. In vielen Resource-Plänen ist dieser Parameter mittlerweile ein wichtiger Bestandteil. Hinweise: Man sollte „mgmt\_p1“ und nicht den veralteten Parameter „cpu\_p1“ verwenden. Zu beachten ist auch, dass der Parameter „max\_utilization\_limit“ in der Version 12c in „utilization\_limit“ umbenannt wurde.

Der Parameter „parallel\_degree\_limit\_p1“ gibt die Grenze für den genutzten „degree of parallelism“ (DOP) einer Operation an. Da nicht mehr als zwei Operationen simultan parallelisiert werden können, errechnet sich die Gesamtanzahl der Parallel Execution Server pro Statement aus der doppelten Anzahl des DOP. Der Defaultwert ist übrigens „NULL“, was „unbegrenzt“ bedeutet. Soll nur seriell gearbeitet werden, muss explizit der Wert auf „0“ gesetzt sein. Darüber hinaus gibt es noch weitere Möglichkeiten, Direktiven aufzusetzen. Ein detailliertes Parallel Statement Queuing beispielsweise ist seit 11g



R2 über den Resource Manager möglich. Weitere Informationen über mögliche Direktiven finden sich im „PL/SQL Packages and Types Reference Guide“. Zusätzliche Ebenen lassen sich über den Suffix „p2“ bis „p8“ – also „mgmt\_p2“ für die CPU-Ressource auf Ebene 2 steuern.

Es gibt zwei Möglichkeiten, den Plan zu aktivieren: entweder manuell über den „ALTER SYSTEM“-Befehl oder aber regelmäßig und automatisch über die sogenannten „Scheduler Windows“, die der Oracle-Scheduler-Job beispielsweise für die Maintenance-Jobs verwendet. Das Beispiel in Listing 6 zeigt die Verwendung des Resource-Plans „PB\_RES\_PLAN“ im „Monday Window“.

### Monitoring und Einsatz von Werkzeugen

Zusätzlich zu den verwendeten Prozeduren und Funktionen gibt es auch grafische Implementierungen zur Konfiguration und Verwaltung der Ressourcen. Bis einschließlich Oracle Database 11g stehen dabei die beiden Werkzeuge „Enterprise Manager Cloud Control“ und „Database Control“ mit den entsprechenden Menüpunkten zur Verfügung. Ab Oracle Database 12c findet man die Resource-Manager-Funktionalität in „Cloud Control 12c“ unter dem Menüpunkt „Administration => Resource Manager“.

Ein weiteres grafisches Werkzeug, das zur Konfiguration im Ressourcen-Management-Umfeld nützlich sein kann, ist der „SQL Developer“. Er verfügt schon seit der Version 3 (aktuell in der Version 4.0.2) über einen „DBA Navigator“-Bereich, der unter anderem auch die Konfiguration und Administration des Ressourcen-Managements unterstützt.

Ein exaktes und umfangreiches Monitoring lässt sich allerdings immer noch am schnellsten über „SQL-Skripte“ durchführen. Es gibt wie bei jedem technischen Feature auch im Resource-Manager-Bereich eine Reihe von speziellen Data Dictionary Views, mit denen die Konfiguration im Einzelnen überprüft werden kann. Zusätzlich liefern spezielle Metrikdaten aus „V\$Views“ Hinweise auf aktuelle Performance-Daten. Zum Beispiel kann es vorteilhaft sein zu wissen, ob eine Instanz „CPU-bound“ ist – also ob sie mehr CPU benötigt, als zur Verfügung steht. Hinweise auf dieses Verhalten liefern das Event

„resmgr:cpu quantum“ im AWR oder Informationen aus der View „v\$rsrcmgrmetric\_history“, die sogar die aktuelle „CPU Utilization“ pro Consumer Group liefern kann. Das Ergebnis bei voller Last vergleicht man dann mit der Verwendung im Resource Manager (mit „mgmt\_p1“ etc.). Genaue Beispiele und Skripte finden sich im White Paper „Using Oracle Database Resource Manager“ oder in der DBA-Community.

### Resource Manager und Multi-tenant-Architektur

Mit der Datenbank-Version 12c und dem neuen Konzept der Multitenant-Architektur wurde auch der Resource Manager erweitert. Da nicht alle Pluggable Databases (PDBs) gleich sind, kann auch der Ressourcen-Verbrauch unterschiedlich sein. Um auch hier den wichtigen PDBs ein gewisses Minimum an Ressourcen zu garantieren, ist der Resource Manager in der

```

begin
  dbms_resource_manager.clear_pending_area();
  dbms_resource_manager.create_pending_area();
  dbms_resource_manager.create_plan(
    plan      => 'HIGH_Prio_PLAN',
    comment => 'Plan/Subplan for HighPrio');
  dbms_resource_manager.create_plan_directive(
    plan      => 'HIGH_Prio_PLAN',
    group_or_subplan => 'HPrio_GR',
    comment   => 'HighPrio',
    mgmt_p1   => 70,
    parallel_degree_limit_p1 => 16,
    max_utilization_limit   => 75);
  dbms_resource_manager.create_plan_directive(
    plan      => 'HIGH_Prio_PLAN'
    group_or_subplan => 'HLOW_GROUP',
    comment   => 'Hlowgroup',
    mgmt_p1   => 30,
    parallel_degree_limit_p1 => 2,
    max_utilization_limit   => 35);
  ...
  dbms_resource_manager.validate_pending_area();
  dbms_resource_manager.submit_pending_area();
end;
/

```

Listing 5

```

execute dbms_scheduler.set_attribute(
  name      => '"SYS"."MONDAY_WINDOW"',
  attribute => 'RESOURCE_PLAN',
  value     => 'PB_RES_PLAN');

```

Listing 6

PROCEDURE CREATE_CDB_PLAN_DIRECTIVE	Argument Name	Type	In/Out	Default?
	PLAN	VARCHAR2	IN	
	PLUGGABLE_DATABASE	VARCHAR2	IN	
	COMMENT	VARCHAR2	IN	DEFAULT
	SHARES	NUMBER	IN	DEFAULT
	UTILIZATION_LIMIT	NUMBER	IN	DEFAULT
	PARALLEL_SERVER_LIMIT	NUMBER	IN	DEFAULT

Listing 7

CDB Plan			PDB3 Plan	
PDB	Shares	Utilization Limit	Consumer Group	CPU
PDB1	3	100%	OLTP	75%
PDB2	3	100%	Reporting	15%
PDB3	1	70%	OTHERS	10%

Abbildung 1: PDB- und CDB-Resource-Pläne

```
SELECT name, instance_caging FROM v$rsrc_plan WHERE is_top_plan = 'TRUE';
show parameter cpu_count
```

Listing 8

```
SELECT to_char(begin_time, 'HH24:MI')time, sum(avg_running_sessions)
avg_running_sessions, sum(avg_waiting_sessions) avg_waiting_sessions
FROM v$rsrcmrgmetric_history
GROUP BY begin_time ORDER BY begin_time;
```

Listing 9

```
SQL> SELECT username, elapsed_time, plsql_exec_time, sql_text,
cpu_time,rm_last_action, rm_last_action_reason,
rm_last_action_time, rm_consumer_group
FROM v$sql_monitor WHERE username is not null;
USERNAME ELAPSED_TIME PLSQL_EXEC_TIME
-----
SQL_TEXT
-----
CPU_TIME RM_LAST_ACTION
-----
RM_LAST_ACTION_REASON RM_LAST_A RM_CONSUMER_GROUP
-----
SH 378358 0
select /*+ use_nl(c) parallel ordered*/ count(*) from sh.sales s,sh.
customers c where c.cust_id=s.cust_id and cust_first_name='Dina'
10998 SWITCH TO OTHER_GROUPS
SWITCH_CPU_TIME 19-FEB-14 OTHER_GROUPS
```

Listing 10

Lage, den Ressourcen-Verbrauch pro PDB zu verwalten. Dabei gibt es PDB- und CDB-Resource-Pläne. Ein „CDB-Resource-Plan“ enthält Direktiven für alle PDBs; PDB-Resource-Pläne hingegen enthalten Direktiven für die Ressourcen-Verteilung der Consumer Groups innerhalb einer PDB. Ein CDB-Resource-Plan wird immer innerhalb des ROOT-Containers konfiguriert. Dabei wird das neue Konzept der Shares (Anteile) benutzt. Je höher der Share-Wert

ist, desto höher sind die daraus resultierenden Ressourcen-Anteile. Mögliche Ressourcen sind dabei CPU („shares“ und „utilization\_limit“) und Parallel Execution Server („parallel\_server\_limit“). Die neue Prozedur „create\_cdb\_plan\_directive“ erzeugt dann die CDB-Resource-Pläne im ROOT-Container (siehe Listing 7).

PDB-Resource-Pläne sind vergleichbar mit dem Setup eines NON-CDB-Resource-Manager-Plans und werden auch

ebenso erzeugt. Allerdings existieren einige Einschränkungen. Ein PDB-Resource-Plan kann beispielsweise keine Subpläne enthalten und maximal acht Consumer Groups werden pro PDB unterstützt. Die Aktivierung erfolgt dabei in der jeweiligen PDB. Vorab muss ein CDB-Resource-Plan angelegt worden sein (siehe Abbildung 1).

### Weitere Funktionen

Außer den beschriebenen Features gibt es noch weitere interessante Funktionen im Zusammenhang mit dem Resource Manager. So besteht die Möglichkeit, I/O-Ressourcen detailliert auf Datenbank- und Cell-Ebene eines Exadata Storage Servers zu verwalten. Um solche I/O-Anfragen zu regeln oder zu priorisieren, steht diese Funktion unter dem Namen „IO Resource Manager“ (auch IORM) zur Verfügung. Dabei können die Ressourcen nicht nur innerhalb einer Datenbank („INTRA Database Plan“), sondern auch zwischen den Datenbanken („INTER Database Plan“) verwaltet werden. Eine genaue Beschreibung des Setups findet sich im Handbuch „Oracle Exadata Storage Server Software User's Guide“ sowie in der „My Oracle Support Note Master Note for Oracle Database Resource Manager“ (Doc ID 1339769.1).

Ein weiteres wichtiges Feature, das nur im Zusammenhang mit Resource Manager aktiviert werden kann, ist das sogenannte „Instance Caging“. Damit lässt sich seit der Version 11g R2 auf allen Plattformen die maximale Anzahl der CPUs pro Instance einstellen. Das Vorgehen ist einfach: Mit „ALTER SYSTEM“ wird pro Instanz der Wert von „CPU\_COUNT“ dynamisch festgelegt; dabei ist eine Aufteilung der vorhandenen CPUs auf Instanzen oder sogar ein Over-Subscribe der CPUs möglich. Hinweis: Änderungen an „CPU\_COUNT“ haben auch direkten Einfluss auf andere Technologien wie den Optimizer. Nun muss noch ein Resource-Manager-Plan aktiv sein. Falls man keine speziellen Anforderungen an einen eigenen Resource-Plan hat, bietet sich der mitgelieferte Resource-Plan „DEFAULT\_PLAN“ an. Listing 8 zeigt eine einfache Überprüfung des Setups.

Um die Ressourcen zu kontrollieren, eignet sich die vorhin schon erwähnte View „v\$rsrcmrgmetric\_history“. Im Beispiel (siehe Listing 9) wird die durchschnittliche Anzahl von laufenden und wartenden Sessions angezeigt.

Vor Oracle Database 12c war es schon möglich, im Database Resource Manager eine Grenze (Threshold) für lang laufende Queries anzugeben – man spricht hier auch von „Runaway Queries“. Diese Grenze kann in der „create\_plan“-Prozedur über Parameter wie „switch\_io\_megabytes“, „switch\_io\_reqs“, „switch\_for\_call“, „switch\_time“, „switch\_estimate“, „max\_est\_exec\_time“ oder neu in 12c über „switch\_io\_logical“ und „switch\_elapsed\_time“ eingestellt werden. Nach Erreichen dieser Grenze erfolgt eine Aktion. Dauert beispielsweise eine Query oder ein PL/SQL-Aufruf länger als 30 Sekunden (in CPU), dann kann man die Query beenden, einen Wechsel (Switch) zu einer anderen Consumer Group vollziehen oder sogar die ganze Session beenden.

Wer hat diese Queries ausgeführt? Welcher Code (SQL oder PL/SQL) wurde verwendet? Und was für eine Aktion wurde durchgeführt? Eine Erweiterung in Oracle Database 12c im „SQL Monitoring“-Umfeld gibt Antworten auf diese Fragen. Die neuen Spalten „rm\_consumer\_group“, „rm\_last\_action“, „rm\_last\_action\_reason“ und „rm\_last\_action\_time“ in „v\$sql\_monitor“ liefern die entsprechenden Informationen. Im folgenden Beispiel wird nach Erreichen einer Grenze der Wechsel von „GRUPPE\_LOW\_CPU“ zu „OTHER\_GROUPS“ initiiert (siehe Listing 10).

Will man dabei die Runaway Queries nur überwachen und keine zusätzliche Aktion durchführen, gibt es den neuen Wert „LOG\_ONLY“, der mit den Prozeduren „create\_plan\_direktive“ oder „update\_plan\_direktive“ eingestellt werden kann (siehe Listing 11). Das Monitoring kann dann über eine Abfrage erfolgen (siehe Listing 12). Der User „SCOTT“ führte offensichtlich um 16:44 eine lang laufende Prozedur aus und wurde in „v\$sql\_monitor“ aufgelistet.

### Fazit und abschließende Tipps

Der Database Resource Manager bietet eine Vielzahl an Möglichkeiten, Ressourcen zu verwalten und zu überwachen. Weitere Funktionen werden in den nächsten Database Releases sicherlich folgen. Wie bei allen Features gilt auch hier: Bevor man den Resource-Plan aktiviert, sollte man Tests durchführen. Dabei ist wichtig zu wissen, dass die Direktiven erst dann greifen können, wenn die Res-

```
BEGIN
dbms_resource_manager.clear_pending_area();
dbms_resource_manager.create_pending_area();
dbms_resource_manager.update_plan_directive(
    plan                => 'TEST_RUNAWAY',
    group_or_subplan    => 'GRUPPE_HIGH_CPU',
    new_switch_group    => 'LOG_ONLY',
    new_switch_time     => 200);
dbms_resource_manager.update_plan_directive(
    plan                => 'TEST_RUNAWAY',
    group_or_subplan    => 'GRUPPE_LOW_CPU',
    new_switch_group    => 'OTHER_GROUPS',
    new_switch_time     => 30);
dbms_resource_manager.update_plan_directive(
    plan                => 'TEST_RUNAWAY',
    group_or_subplan    => 'OTHER_GROUPS',
    new_switch_group    => 'CANCEL_SQL',
    new_switch_time     => 100);
dbms_resource_manager.submit_pending_area();
END;
/
```

Listing 11

```
SQL> SELECT username, elapsed_time, plsql_exec_time, sql_text,
    cpu_time, rm_last_action, rm_last_action_reason,
    rm_last_action_time, sql_exe_start, rm_consumer_group
    FROM v$sql_monitor WHERE username is not null;
USERNAME                               ELAPSED_TIME  PLSQL_EXEC_TIME
-----
SQL_TEXT
-----
CPU_TIME  RM_LAST_ACTION
-----
RM_LAST_ACTION_REASON                   RM_LAST_ACTION_T SQL_EXEC_START
-----
RM_CONSUMER_GROUP
-----
SCOTT                                     140668120        140658478
BEGIN last1.sortiere(32767); END;
116168340
                                     19.02.2014 16:44
GRUPPE_HIGH_CPU
```

Listing 12

sourcen unter Last stehen. Bevor man mit dem Skripting startet, sollte man zuerst in Ruhe überlegen, welchen Gruppen welche Ressourcen garantiert werden sollen. Man bildet dann die Gruppen am besten auf entsprechende TNS-Services ab; das ist eine gängige Praxis.

Der Plan sollte auf jeden Fall einfach gehalten werden. Zudem muss man dafür sorgen, dass der Administrator in der „SYS\_GROUP“ immer ausreichend Ressourcen zur Verfügung hat. Cloud Control 12c bietet gute Übersichten zur Anzeige von Resource-Plänen und Con-

sumer Groups. Man kann für den eigenen Plan die vorhandenen Skripte (siehe „Weitere Informationen“) nutzen, die sich leicht an eigene Anforderungen anpassen lassen. Ein abschließendes genaues Ressourcen-Monitoring lässt sich mit entsprechenden Skripten durchführen. Wer sich mit Cloud Control ein wenig besser auskennt, kann natürlich auch zusätzlich eigene „user defined metrics“ (beziehungsweise „metric extensions“) im Cloud Control hinterlegen, um ganz automatisch Notifikationen bei bestimmten Ereignissen zu erhalten.

## Weitere Informationen

- *My Oracle Support*  
Master Note for Oracle Database Resource Manager (Doc ID 1339769.1)
- *Deutschsprachige Tipps der DBA Community*  
[http://blogs.oracle.com/dbacomcommunity\\_deutsch](http://blogs.oracle.com/dbacomcommunity_deutsch)
- *Skripte*
  - Im DBA Community Blog: Erfahrungsbericht zur Nutzung des Database Resource Manager
  - Im Blog von Joel Kallman: [joelkallman.blogspot.co.uk](http://joelkallman.blogspot.co.uk)
- *Oracle White Paper*
  - Using Oracle Database Resource Manager (mit guten Monitoring-Skripten)
  - Effective Resource Management Using Oracle Database Resource Manager
- *Handbücher*
  - Oracle Database Administrator's Guide
  - PL/SQL Packages and Types Reference



Ulrike Schwinn  
[ulrike.schwinn@oracle.com](mailto:ulrike.schwinn@oracle.com)

# S3-Programm – Klappe, die erste

Yasmin Misch, DOAG Dienstleistungen GmbH

Fast sechzig Studierende von dreizehn unterschiedlichen Hochschulen, fünfzehn Sponsoren und neun Hochschul-Professoren: So lauten die Zahlen des ersten StudentSponsorShip-Programms (kurz: „S3“). Dieses fand erstmals im Rahmen der DOAG 2014 Konferenz + Ausstellung statt. „Ein voller Erfolg“ und „Wir hoffen auf Wiederholung im nächsten Jahr“ meinten die Teilnehmer. Kurzum: eine gelungene Premiere!

Montag, 14 Uhr: Alle Studierenden sind angereist. Gleich wird das Orga-Team sie einweisen. Raum- und Referentenbetreuung, Zutrittskontrolle oder die Ausgabe von Dolmetscher-Headsets sind nur einige der Tätigkeiten, die Studierende nun schon seit einigen Jahren auf der DOAG Konferenz + Ausstellung im Rahmen des Studenten-Programms übernehmen. Im Gegenzug erhalten die Studierenden mehr als freien Zutritt zu Vorträgen und Ausstellung, mehr als die Möglichkeit, kostenfrei am Schulungstag teilzunehmen, und mehr als einen aufregenden Community-Abend. Wer am S3 teilnimmt, bekommt zusätzlich auch noch Reise- und Übernachtungskosten erstattet.

## Kontakte sind mehr wert als Geld

Der finanzielle Vorteil ist für die meisten Studierenden nicht der alleinige Anreiz für eine S3-Teilnahme. Der Kontakt zu den sponsernden Unternehmen und die Aussicht auf lukrative Job-Angebote seien Gold wert,

berichten die Studierenden einheitlich. Prof-finanzierten Sponsorship vermittelt die DOAG Gespräche mit Studierenden. Doch auch darüber hinaus suchen diese den Kontakt zu den Sponsoren. Auf dem Get-together am zweiten Tag der Konferenz kommen alle Teilnehmer des S3 zusammen. Die Sponsoren stellen sich vor. Wer es schafft, das Interesse der Studierenden zu wecken, ergattert zusätzliche Zeit mit dem Nachwuchs.

## Auszeichnungen für Professor und Studierende

Mittwochabend: Ernennung zum DOAG-Botschafter des Jahres im Bereich „Technologie“. Die Auszeichnung geht an Prof. Dr. Harm Knolle von der Hochschule Bonn-Rhein-Sieg. Das Komitee würdigt sein Engagement bei den studentischen Aktivitäten. Er hatte einmal mehr eine Exkursion zur DOAG Konferenz organisiert und war mit rund dreißig Studierenden angereist. Außerdem war er maßgeblich an der Entstehung des S3 beteiligt. An dieser Stelle nochmal „Herzlichen Glückwunsch!“

Auszeichnungen gibt es auch unter den Studierenden für ihre Teilnahme an einem Test. Zur Auswahl standen Online-Tests in den Bereichen „Java“ oder „SQL“. Alternativ konnten sie an einem eigens für die Konferenz kreierte Quiz teilnehmen. Die zwanzig Fragen stammten direkt von den Sponsoren. Der Erstplatzierte, Tobi-

as Lindner, erhielt eine Mini-Drohne. Für den zweiten (Anja Schäfer) und den dritten Platz (Sebastian Mahlke) erhielten die Studenten jeweils eine Solarblume, mit der sie nun über USB den Akku ihres Handys laden können.

## S3 wird ausgeweitet

Bisher war das S3 ein Privileg der DOAG Konferenz + Ausstellung. Nach der guten Resonanz hat die DOAG beschlossen, diese in Zukunft auch für die Java-Konferenz JavaLand anzubieten. Sie findet vom 24. bis 26. März 2015 im Phantasialand in Brühl statt. Dem bisherigen Feedback folgend, deutet auch hier alles auf einen gelungenen Start für das S3 im JavaLand hin.



Yasmin Misch  
[yasmin.misch@doag.org](mailto:yasmin.misch@doag.org)



# Faktenbasierte Indexierung – ein Erfahrungsbericht

Lothar Flatz, Diso AG

Das Erstellen einer faktenbasierten Indexierung zählt zu den Königsdisziplinen der IT. Der Artikel zeigt, dass diese sich in verschiedener Hinsicht auszahlt.

Während seiner zehnjährigen Tätigkeit für Oracle Consulting war der Autor mit spannenden Aufgaben betraut. Darunter war die Mitarbeit an der Software zur Auswertung der Messdaten beim CERN oder eben die komplette Neuindexierung einer großen Online-Datenbank. Wenn dabei von „Index“ die Rede ist, ist damit der konventionelle „B\*-Index gemeint. Für ein Data Warehouse gelten andere Indexierungsregeln, diese würden allerdings den Rahmen dieses Artikels sprengen.

## Kaffeepause mit Folgen

Man saß gemütlich beim Kaffee bei einem Stammkunden, der dem Autor einen Tuning-Auftrag erteilt hatte. Der erfahrene DBA meinte: „Ich wette, mindestens 30 Prozent unserer Indizes sind überflüssig. Die Folge: Unsere Ausführungspläne sind instabil und brauchen Platz. Am liebsten würde ich die Indexierung komplett überprüfen lassen.“

Fakt ist: Bei eingekauften Applikationen werden die Indizes in der Regel von den Entwicklern erstellt. Diese erleben das System selten unter Volllast. Zudem ist die Indexierung nie an die Bedürfnisse eines bestimmten Kunden angepasst. Es muss also nach-indexiert werden. Die Grundlage, die Entwickler geschaffen haben, wird aber nicht mehr infrage gestellt. Es kommen lediglich neue Indizes dazu. Das Resultat ist ein unübersichtlicher, dichter Dschungel aus Indizes.

Wie wäre es, einmal alles auf Start zu setzen und von Beginn weg richtig zu machen? Die Fakten dazu sind vorhanden. Das Werkzeug auch, schließlich ist die Oracle-Datenbank eine unerschöpfliche Quelle von Informationen. Über die Infor-

mationen, die im Shared Pool und in AWRs gespeichert sind, müsste eine geeignete Indexierung erstellt werden können.

Dieser Gedanke ließ den Autor nicht mehr in Ruhe. Einige Monate später kam die Chance: Einer seiner Kunden hatte genug von halbherzigen Lösungen. Er wollte Ordnung in der Index-Struktur seiner wichtigsten Applikationen. Am Anfang brauchte der Autor seine ganze Überredungskraft, um den Kunden davon zu überzeugen, dass die Indexierung von Grund auf neu gemacht werden muss. Denn er war überzeugt, dass man sich bei Verbesserungen auf Basis der bestehenden Indexierung nur verzetteln würde.

Natürlich hätte man einfach ein großes SQL-Tuning-Set erstellen und dieses durch den SQL-Tuning-Advisor abarbeiten lassen können. Man hätte sich dann allerdings blind auf die Entscheidungen des SQL-Tuning-Advisor verlassen müssen und das Wissen und die Erfahrungen der Entwickler und DBAs nicht verwenden können.

In diesem Zusammenhang die Argumente aus der Fachliteratur: „First, while the automated tools reduce the complexity of the physical design process, it is still nontrivial to identify a representative workload that can be used to drive the physical design in its entirety. Second, automated tools do not consider all factors that impact physical design (e.g., the impact of replication architectures). [1].“ Neue Schweizer Forschungen weisen darauf hin, dass die automatische Lösung mittels Tools für große Gesamtaufgaben problematisch ist [2, 3].

Man hat sich dann entschlossen, ein halbautomatisches Vorgehen zu wäh-

len. So ließ sich jeder Schritt erklären und nachvollziehen. Jedes Teammitglied konnte seine Gedanken einbringen. Am Schluss stand eine klar begründete Lösung, zu der jeder stehen konnte.

## Die Vorbereitung

Zunächst wurde ein Team aus allen Beteiligten gebildet. Dieses bestand aus zwei Vertretern aus dem Entwicklungsteam des Applikationsherstellers, zwei DBAs als Vertreter des Softwarebetreibers und dem Autor als externer Berater für den Datenbank-Hersteller. In zwei Sitzungen wurden das grobe Vorgehen bestimmt und die Grundregeln der Indexierung festgelegt:

- Primär- und Fremdschlüssel bekommen grundsätzlich automatisch einen Index
- Namenskonventionen
- Design, das der Applikation Rechnung trägt
- Art der physischen Speicherung (zum Beispiel Tablespace)

## Datensammelungsphase

In dieser Phase wurden möglichst viele Informationen über die Abfragebedingungen auf der Datenbank gespeichert. Dabei war es essenziell, möglichst alle wichtigen Verarbeitungen in die Auswertung mit einzubeziehen – also mussten nicht nur tägliche Verarbeitungen, sondern auch wöchentliche und monatliche Aktivitäten Berücksichtigung finden. Diese zeitaufwändige Phase zog sich über mehrere Monate hin. Die Arbeit wurde von automatischen Sammel-Tools geleistet, das Team hatte damit relativ wenig zu tun. Im

Wesentlichen werden folgende Informationen gesammelt:

- Häufigkeit und die Vergleichsoperatoren, mit denen nach einer bestimmten Spalte gesucht wird
- Kombination von Spalten, nach denen gleichzeitig gesucht wird, und die Häufigkeit, mit der dies geschieht

Dies sind im Wesentlichen die gleichen Informationen, die die Datenbank zur Unterstützung der automatischen Statistik-Generierung in der „sys.col\_usage\$“ sammelt. Ab der Version 11.2.0.2 werden auch Spalten-Kombinationen unterstützt. Das entsprechende Verfahren beschreibt Maria Colgan aus dem Oak-Table-Netz und bis vor Kurzem Produktmanagerin des Optimizers in ihrem Blog [4]. In früheren Datenbank-Versionen werden die Spalten-Kombinationen nicht unterstützt und müssen daher aus den anderen Quellen hochgerechnet werden. Als Quellen kommen infrage:

- Die aktuellen Abfragen „Shared Pool“. Hier kann man die Suchkriterien ganz einfach aus den Filtern und „Access Predicates“ entnehmen.
- Die „top“-Statements aus dem WAR. Hier muss man leider einen „reparse“ durchführen, da die Filter- die Access Predicates in der „DBA\_HIST\_SQL\_PLAN“-Tabelle leer sind.

Diese Möglichkeiten sind deshalb wichtig, weil man sich auf den Inhalt der „col\_usage\$“ nicht hundertprozentig verlassen kann. Der Autor hat Datenbanken gesehen, bei denen der Inhalt der „col\_usage\$“ nicht brauchbar war, vermutlich aufgrund von Speichermangel. Es ist klar, dass dann auch die Statistikgenerierung in Mitleidenschaft gezogen wird. Außerdem werden Informationen aus dem Dictionary verwendet :

- Die Selektivität der einzelnen Spalten sowie die der verwendeten Kombinationen
- „Primary Key“- und „Foreign Key“-Constraints

### Die Auswertung

In dieser Phase werden die gesammelten Informationen zu einem fertigen In-

dex-Design verdichtet. Dies geschieht über mehrere Etappen. Bei schwierigen Design-Entscheidungen wird immer noch das menschliche Wissen als letzte Instanz hinzugezogen.

Das Vorgehen folgt grob dem Muster des als „Merge and Reduction“ bekannten Algorithmus [3]. Als Grundlage für die Arbeit dienen die in der Vorphase gefundenen Spalten-Kombinationen. Im Grunde genommen könnte man aus jeder Spalten-Kombination, die im „Suchen“ auftaucht, einen Index erzeugen, was aber zu einem Überangebot an Indizes führen würde. Da jeder Index die DML-Operationen verlangsamen kann [6], sollen nur so viele Indizes wie nötig und so wenige wie möglich erstellt werden. In den folgenden Schritten wird daher versucht, die Index-Struktur hinsichtlich Preis/Leistung zu optimieren.

### Grund-Indexierung

Meist wird man von einer Basis-Indexierung ausgehen. Primary Keys erhalten in der Regel ohne große Überlegung einen Index. Foreign Keys sollte man im Allgemeinen auch indexieren, und zwar aus folgenden Gründen:

- Da es beim Löschen eines Satzes zu unangenehmen Sperrungen der abhängigen Sätze kommen kann, sofern ein „Foreign Key“-Constraint existiert [5]
- Damit ein „nested loop“-Join in jede bestimmte Richtung möglich ist und somit der Optimizer in der Wahl des besten Zugriffsplans nicht unnötig eingeschränkt wird

### Elimination

In dieser Phase nimmt man alle Suchkombinationen, die man ohne große Leistungseinbußen weglassen kann, aus der Betrachtung. Das sind insbesondere:

- Foreign- und Primary-Keys, die bereits im vorigen Schritt indexiert worden sind
- Suchkombinationen, die so wenig selektiv sind, dass sich kein Index lohnt
- Suchkombinationen, die bereits in anderen Suchkombinationen gleichwertig enthalten sind

Dazu einige Beispiele aus dem allgemein bekannten Bereich der Adress- und Personendaten: Ein Index „{Geschlecht}“

lohnt sich auf der Tabelle „Person“ nicht, da die Suchspalte zu wenig selektiv ist. Ausnahme: Eine Anwendung für Frauenfragen beim Militär, da hier häufig nach einem seltenen Wert gesucht wird. Dieses Beispiel zeigt, dass selbst scheinbar einfache Fälle nicht ohne Nachdenken entschieden werden können. Einmal mehr wird klar, vor welchen Schwierigkeiten eine vollautomatische Indexierung steht. Ein anderes Beispiel: Der Indexkandidat „{Ort}“ auf der Tabelle „Adresse“ wird eliminiert, wenn es eine andere häufige Suchkombination gibt, zum Beispiel „{Ort, Straße}“, in der „{Ort}“ bereits enthalten ist. Dies kann man natürlich noch weiter führen: Der dreispaltige Index „{Nachname, Vorname, Alter}“ auf der Tabelle „Person“ kann drei verschiedene Suchkriterien unterstützen:

- {Nachname, Vorname, Alter}
- {Nachname, Vorname}
- {Nachname}

Die entsprechenden Suchkombinationen können also eliminiert werden. Folgende Suchabfragen hingegen können nicht unterstützt werden und müssen bestehen bleiben:

- {Vorname, Alter}
- {Alter}
- {Vorname}

An diesen Beispielen erkennt man, dass die Wahl der richtigen Reihenfolge der Spalten entscheidend für die Qualität eines flexiblen Index-Designs ist. Eine optimale Indexierung, unabhängig von der Spalten-Reihenfolge, ist mit dem konventionellen „B\*“-Index nicht möglich. Wer die ultimative Flexibilität und Performance haben möchte, muss den mehrdimensionalen Index, beispielsweise den der Firma „dimensio“, verwenden. Er kann eine beliebige Kombinationen von Suchkriterien parallel auswerten und verknüpfen. Dies erleichtert die optimale Indexierung wesentlich.

### Die Synthese

In diesem Schritt versucht man, übrig gebliebene Suchkombinationen zusammenzulegen und dadurch weitere Indexkandidaten zu eliminieren. So kann man beispielsweise einen „Foreign Key“ um

weitere Suchkriterien erweitern. Wenn zum Beispiel in der Tabelle „Umsatz“ häufig nach „{Artikelnummer, Verkaufsdatum}“ gesucht wird, ist es sinnvoll, einen bestehenden „Foreign Key“-Index auf der Artikelnummer um das Feld „Verkaufsdatum“ zu erweitern.

Manchmal kann man auch durch eine leichte Verschlechterung des Index-Designs einen zusätzlichen Index-Kandidaten eliminieren, zum Beispiel bei den Suchkombinationen „{Ort, Nachname, Vorname}“ und „{Ort, Nachname, Geburtsdatum}“. Der Indexkandidat „{Ort, Nachname, Vorname, Geburtsdatum}“ kann die beiden Kombinationen ersetzen. Zwar ist die Suche nach „{Ort, Nachname, Geburtsdatum}“ nicht mehr so optimal unterstützt wie beim spezialisierten Index, jedoch kann die Verschlechterung wahrscheinlich in Kauf genommen werden, weil „Ort“ und „Nachname“ für sich gesehen bereits gute Suchkriterien sind. Man sieht, dass in diesem Schritt die Vernunftgesteuerte Entscheidung des menschlichen Designers besonders wichtig ist.

## Tests und Umsetzung

Hat man alle diese Schritte durchlaufen, ist es relativ einfach, aus dem Ergebnis ein „Index create“-Skript zu bilden. Natürlich ist das Ergebnis jetzt ausführlich zu testen. „RAT“ bietet sich dabei als Mittel der Wahl an. Wenn man produktiv geht, sollte man auch Überwachungsmechanismen im Einsatz haben, die schnell in der Lage sind, fehlende Indizes zu finden.

Der Autor hatte ein eigenes Überwachungsskript, das Ausführungspläne finden kann, die durch suboptimale Indexierung entstehen, und entsprechende Verbesserungsvorschläge macht. Natürlich kann man auch den „Oracle Index Advisor“ verwenden, der für den Autor allerdings für diesen spezialisierten Zweck etwas umständlicher in der Handhabung ist. Die neue Index-Struktur zeigte sich in der Produktion erstaunlich stabil. In den ersten Monaten musste nur eine einstellige Anzahl von Indizes nacherzeugt werden.

## Fazit

In dem hier geschilderten Fall handelt es sich um eine Applikation, die spezialisierte Consultants der Firma Oracle schon seit Monaten optimiert hatten. Umso erfreu-

licher war es, dass die Gesamt-Indexierung zusätzlich eine Platzersparnis von 30 Prozent und eine Performance-Verbesserung von ebenfalls 30 Prozent erbrachte.

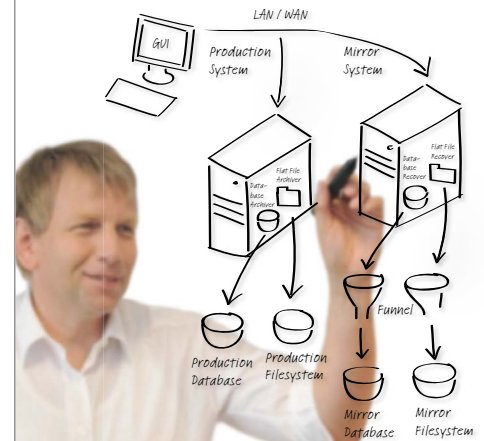
Ein sehr willkommener Nebeneffekt kam überraschend: die verbesserte Stabilität der Execution-Pläne. Da für einen bestimmten Zweck nur noch ein Index zur Verfügung steht und nicht mehrere, bleiben die Pläne stabiler und verändern sich nicht so leicht in Abhängigkeit von den Werten der Bind-Variablen.

Last but not least: Die neue Indexstruktur mit ihren klaren Regeln und ihrer einheitlichen Namensgebung erleichtert die tägliche Arbeit der DBAs.

## Weitere Informationen

- [1] Weikum, Moenkeberg, Hasse, Zaback, „Self-tuning Database Technology and Information Services: from Wishful Thinking to Viable Engineering“, Proceedings of the 28th VLDB Conference, Hong Kong, China, 2002
- [2] Bruno, N. and Chaudhuri, S. 2007. Physical design refinement: The ‘merge-reduce’ approach. ACM Trans. Database Syst. 32, 4 (Nov. 2007), 28.
- [3] Borovica, Alagiannis, Ailamaki. „Automated Physical Designers: What You See in (Not) What You Get“, DBTest’12, May 21, 2012 Scottsdale, AZ, USA.
- [4] Colgan M., How do I know what extended statistics are needed for a given workload? blogentry: [https://blogs.oracle.com/optimizer/entry/how\\_do\\_i\\_know\\_what\\_extended\\_statistics\\_are\\_needed\\_for\\_a\\_given\\_workload](https://blogs.oracle.com/optimizer/entry/how_do_i_know_what_extended_statistics_are_needed_for_a_given_workload).
- [5] Oracle® Database Concepts 12c Release 1 (12.1), Kapitel 9, Locks and Foreign Keys and Kapitel 6, Indexes and Foreign Keys
- [6] Oracle Database 2 Day DBA 12c Release 1 (12.1), Kapitel 8. Managing Indexes

## Libelle BusinessShadow®



Unabhängig bezüglich

- Fehlerursache
- Entfernung
- Hardware / Architektur
- Komplexer Systeme

Schnelle Arbeitsaufnahme

- Mit konsistenten Daten
- Auf Knopfdruck
- Automatisiert
- ...

Hans-Joachim Krüger  
Chief Technology Officer  
Libelle AG

Erfahren Sie mehr:  
[www.Libelle.com/business](http://www.Libelle.com/business)



Lothar Flatz  
lflatz@diso.ch

ORACLE Gold Partner



Libelle

Libelle AG

Gewerbestr. 42 • 70565 Stuttgart, Germany  
T +49 711 / 78335-0 • F +49 711 / 78335-148  
[www.Libelle.com](http://www.Libelle.com) • [sales@libelle.com](mailto:sales@libelle.com)

# Wir haben doch keine Zeit – Tipps zum Oracle-Text-Index in Webcenter Content

Gunther Thielemann, Slix Gesellschaft für Computersysteme mbH

Mit OracleTextSearch (OTS) ermöglicht WebCenter Content (vormals Universal Content Management, UCM) die Suche nach Metadaten und Inhalten auch über große Bestände. Auch wenn mit der Version 11g die Performance gesteigert wurde, dauert der komplette Aufbau eines Index über mehrere Millionen Dokumente immer noch eine Weile. Der Artikel beschreibt, wie man diesen Prozess mit geringfügigen Änderungen weiter beschleunigen kann, indem man die Aufgaben geschickt auf Content Server und Datenbank verteilt.

Eine Anwendung, die schon seit einigen Jahren im Einsatz ist, sollte ein Upgrade von UCM 10g R3 auf UCM 11g erfahren. Die Anzahl der Dokumente war mittlerweile auf etwa vier Millionen angewachsen. Mit dem Upgrade war ein kompletter Neuaufbau der Search Collection verbunden. Da dieser Vorgang in der Vergangenheit mit geringerem Volumen nicht nur Tage, sondern Wochen in Anspruch genommen hatte, wurde nach einer Lösung gesucht, um den Zeitaufwand wesentlich zu reduzieren. Gleichzeitig sollten der Fortschritt besser überwacht und realistische Aussagen zur voraussichtlichen Fertigstellung ermöglicht werden.

## Suche nach dem Flaschenhals

Die Dauer des Neuaufbaus ist neben Anzahl, Größe und Typ der Dokumente von der Umgebung und der verfügbaren Hardware abhängig. Die Untersuchung des vorhandenen Systems ergab, dass der Gesamtprozess nicht, wie etwa zu vermuten, durch die aufwändige Text-Extraktion oder zu geringen Durchsatz im Netzwerk, sondern durch die Synchronisation des Oracle-Text-Index gebremst wurde. Als hinderlich erwiesen sich die beim I/O-Verhalten der Datenbank festgestellten Latenzen. Da sich im Rahmen des Projekts daran nichts ändern ließ, mussten andere Lösungen gefunden werden.

Es wurde nach einer Möglichkeit gesucht, die Synchronisation des Index durch parallele Ausführung zu beschleunigen.

Dass sich damit auch ohne Änderungen am I/O-Subsystem der Datenbank die Geschwindigkeit deutlich steigern ließ, wurde durch Tests bestätigt. Blieb nur noch die Frage, wie sich diese Erkenntnis für den Indexaufbau unter UCM nutzen ließe.

Die im Repository-Manager konfigurierbaren Parameter (siehe *Abbildung 1*) sind übersichtlich. „Content-Objekte pro Indizierungsbatch“ legt die Anzahl der Dokumente je Stapel fest. Der zweite Wert bestimmt, in welchen Intervallen ein Checkpoint erfolgt. Hier empfiehlt Oracle, die Voreinstellung von 100.000 Dokumenten beizubehalten. Größere Werte führen auch nach unserer Erfahrung nicht zu merklichen Geschwindigkeitsvorteilen und wirken sich bei eventuellen Unterbrechungen kon-

traproduktiv aus. Eine Verringerung bedingt häufigere Checkpoints und kostet damit Zeit.

Wenn je Stapel mehr Objekte verarbeitet werden, kann das den Durchsatz durchaus steigern. Beliebig heraufsetzen kann man die Anzahl jedoch nicht. Um den Text zu extrahieren, werden alle Dokumente eines Stapels in den Hauptspeicher (der JVM) geladen; sie teilen sich diesen mit den Threads des Content Servers. Die roten Felder in *Abbildung 2* zeigen die zu durchlaufenden Schritte. In der vorgesehenen Abfolge wird nach Speichern von Metadaten und Text in der Datenbank die Prozedur „CTX\_DDL.SYN\_INDEX“ aufgerufen und auf deren Abschluss gewartet.

Den Aufruf so abzuwandeln, dass die Synchronisation parallel erfolgt, wäre na-

**Neuerstellungszyklus des Suchindex**

Der Dokumentindex-Neuerstellungszyklus erstellt einen neuen Suchindex, der alle freigegebenen Dokumente enthält.

**Contentobjekte pro Indizierungsbatch:**

**Contentobjekte pro Checkpoint:**

**Debug-Ebene der Indizierung:**

Abbildung 1: Parameter im Repository Manager



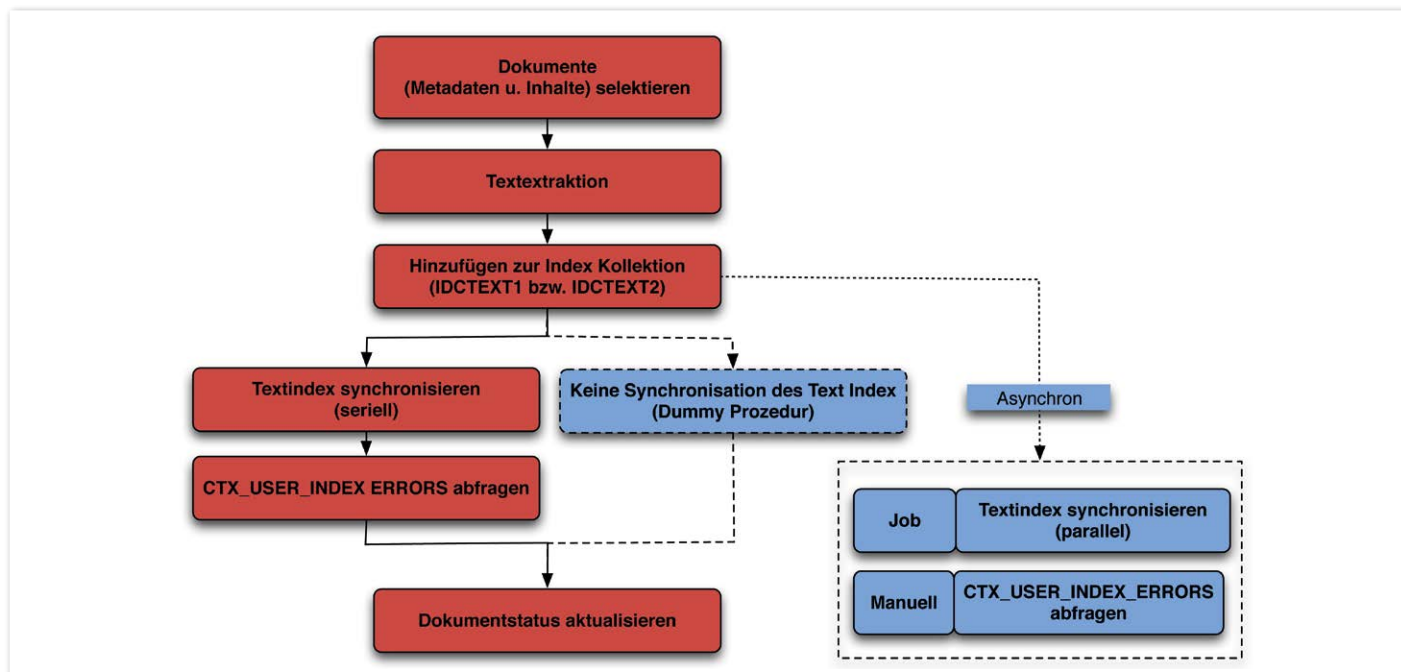


Abbildung 2: Ablauf-Indizierung

heliegend, doch dafür sind die Stapel einfach zu klein. Ein höherer Wirkungsgrad erfordert die parallele Verarbeitung von mehreren Tausend Objekten und damit die Veränderung des Ablaufs. Wie in *Abbildung 2* mit den blauen Feldern dargestellt, kann man diesen Schritt ausgliedern und asynchron durch einen Job in der Datenbank ausführen lassen. Dabei kon-

trolliert der Content Server die Synchronisation des Text-Index in der Datenbank nicht mehr, kann aber, ohne auf deren Abschluss zu warten, den nächsten Stapel in Angriff nehmen. Bei Testläufen traten in diesem Zusammenhang keine Probleme auf. Die Überwachung eines Datenbank-Jobs verursacht keinen außergewöhnlichen Aufwand.

### Umsetzung

Die notwendigen Anpassungen sind mit geringem Aufwand zu implementieren. Welche Prozedur der Content Server bei der Synchronisation von OracleTextSearch aufruft, wird in der Datei „query.htm“ im Verzeichnis „<middleware\_home>/Oracle\_ECM1/ucm/idc/resources/core/tables“ bestimmt (*siehe Listing 1*).

Für Tests ist es ausreichend, nach Erstellen einer Kopie die Anpassungen in der Originaldatei vorzunehmen. Später sollte man dafür eine Komponente verwenden. Die Änderungen (*wie in Listing 2*) und das Erstellen der Prozedur in der Datenbank gemäß *Listing 3* bewirken, dass sich der Content Server nach dem nächsten Neustart beim vollständigen Neuaufbau nicht weiter mit der Synchronisation des Text-Index beschäftigt, sich beim Fast Index Rebuild aber genauso verhält wie vorher. *Listing 4* zeigt, wie der Job erstellt wird. Im diesem Beispiel wird die Synchronisation im Intervall von fünf Minuten gestartet. Der „IndexName“ ist abhängig vom aktiven Index im Content Server mit „FT\_IDCTEXT1“ beziehungsweise „FT\_IDCTEXT2“ anzugeben.

Um die Datenbank besser für Oracle Text optimieren zu können und Prozesse parallel auszuführen, ohne mit dem Content Server um die Ressourcen zu konkurrieren, wurde die Search Collection in eine separate Instanz verlagert (*siehe Abbildung 3*).

```
<tr>
  <td>(ORACLE.CALLABLE)CoracleTextSyncIndex</td>
  <td>{call CTX_DDL.SYNC_INDEX(?, ?, NULL, ?, ?)}</td>
  <td>indexName varchar
    memory:50M varchar
    parallelDegree:1 int
    maxTime:15 int</td>
</tr>
```

Listing 1

```
<tr>
  <td>(ORACLE.CALLABLE)CoracleTextSyncIndex</td>
  <td>{call DEV_OCSSEARCH.DUMMY_SYNC
    (?, ?, NULL, ?, ?)}
</td>
  <td>indexName varchar
    memory:50M varchar
    parallelDegree:1 int
    maxTime:15 int</td>
</tr>
```

Listing 2

## Memory Parameter

Wie viel Hauptspeicher den Prozessen für die Synchronisierung des Text-Index zur Verfügung steht, lässt sich über einen optionalen Parameter beim Aufruf der Prozedur „CTX\_DDL.SYNC\_INDEX“ definieren. Ohne diese Angabe greift der voreingestellte Standard, normalerweise sind das 12 MB. Oracle empfiehlt die Anpassung der Voreinstellungen, sobald der Gesamtumfang des zu indizierenden Content 50 GB übersteigt.

Um einem Prozess mehr als 50 MB zuzuteilen, ist zuvor der Maximalwert „MAX\_INDEX\_MEMORY“ heraufzusetzen, wie in *Listing 5* gezeigt. Die absolute Obergrenze sind 2 GB. Bei der Dimensionierung ist zu berücksichtigen, dass dieser Speicher nicht Bestandteil der SGA ist und sich die Angaben auf jeden einzelnen Prozess beziehen.

## Fortschrittskontrolle

Die im Ressource-Manager-Applet des Content Servers angebotene Fortschrittskontrolle ist für die Überwachung des automatischen Update-Zyklus sicher ausreichend, für den Neuaufbau eines großen Index liefert sie allenfalls eine Momentaufnahme. Wie in *Listing 6* dargestellt, kann man durch periodische Abfrage der Datei „<domain\_home>/cs/search/rebuild /state.hda“ den Status fein granular in einer „.csv“-Datei protokollieren. Die so gesammelten Werte gestatten eine realistische Prognose der voraussichtlichen Fertigstellung und eignen sich auch für die Visualisierung des Verlaufs (*siehe Abbildung 4*).

## Ergebnis

Mit dem beschriebenen Vorgehen war der Neuaufbau schon nach einigen Tagen abgeschlossen. Gegenüber einer Dauer von mehr als fünf Wochen zuvor wurde damit eine Verbesserung um eine Größenordnung erreicht. Der Durchsatz war über den gesamten Zeitraum nahezu konstant. In Tranchen mit mehreren Tausend Dokumenten ließ sich zum einen die Synchronisierung des Text-Index gut parallelisieren und zum anderen wurde der Index nicht so stark fragmentiert. Da keine Anforderung bestand, dass neue und geänderte Dokumente in Echtzeit zur Suche zur Verfügung stehen sollten, wurde dieses Verfahren auch über den Neuaufbau hinaus für den periodischen Aktualisierungszyklus beibehalten.

```
create or replace PROCEDURE DUMMY_SYNC (indexName in VARCHAR2, partition in varchar2, memory in varchar2, parallelDegree in number, maxTime in number)
-- Platzhalter für sync Index
-- Sync Index wird von Job ausgeführt, nicht hier
-- es sei denn, es handelt sich um den Shadow Index
AS
BEGIN
IF upper(indexName) LIKE 'RIO%' THEN
CTXSYS.CTX_DDL.SYNC_INDEX(indexName,memory,null,parallelDegree,maxTime);
else
NULL;
end if;
END DUMMY_SYNC;
/
```

Listing 3

```
BEGIN
DBMS_SCHEDULER.CREATE_JOB (
job_name => 'DEV_OCSSEARCH"."',
job_type => 'PLSQL_BLOCK',
job_action => 'begin ctx_ddl.sync_index(''FT_IDCTEXT1''); end;',
number_of_arguments => 0,
start_date => TO_TIMESTAMP_TZ('2014-10-13 10:00' EUROPE/BERLIN','YYYY-MM-DD HH24:MI TZR'),
repeat_interval => 'FREQ=MINUTELY; INTERVAL=5',
end_date => NULL,
enabled => TRUE,
auto_drop => FALSE,
comments => 'Synchronisieren Index');
END;
/
```

Listing 4

```
begin
ctxsys.ctx_adm.set_parameter ('max_index_memory','1024M');
end;
/

begin
ctxsys.ctx_adm.set_parameter ('default_index_memory','256M');
end;
/
```

Listing 5

```
LOG_DIR=/var/tmp
while [ true ]
do
IDXDOC=$(cat $CS/search/rebuild/state.hda | grep totalAddIndex | cut -d = -f 2)
IDXTIME=$(date "+%d.%m.%Y %H:%M:%S")
echo "$IDXTIME;$IDXDOC;" >> $LOG_DIR/ots_progress.csv
sleep 60
done
```

Listing 6

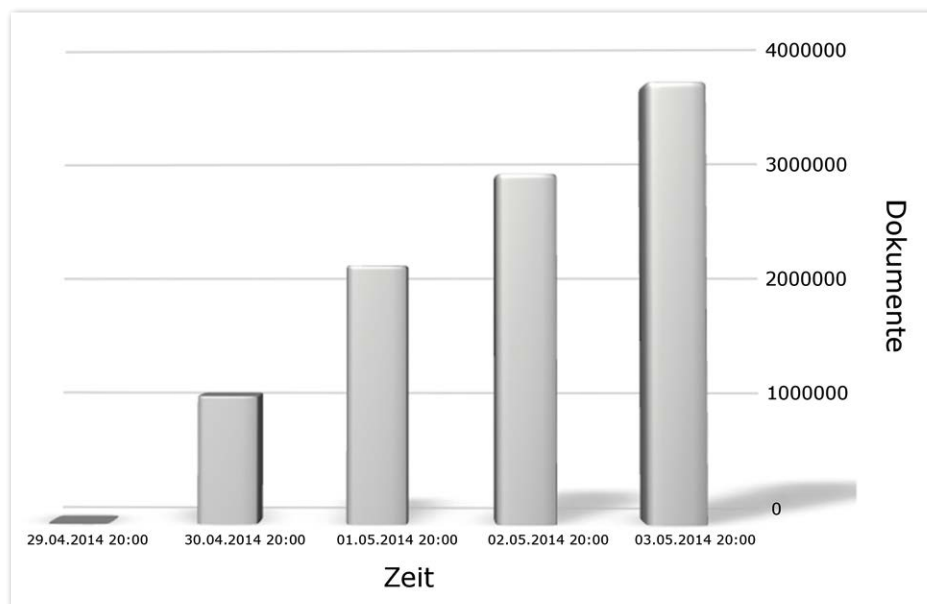


Abbildung 4: Diagramm Index-Neuaufbau

### Fazit

Ein kompletter Neuaufbau der Index-Kollektion für Oracle WebCenter Content kann sich als eine durchaus zeitaufwän-

dige Angelegenheit erweisen und bei großem Bestand schon mal mehrere Tage in Anspruch nehmen. Da der bedeutend schlankere und schnellere Fast Index Re-

build zwar viele Fälle abdeckt, aber eben nicht alle, ist diese Prozedur nicht nur einmalig beim Urknall durchzuführen, sondern hin und wieder unumgänglich.

Mit dem hier vorgestellten Lösungsansatz lassen sich der Zeitaufwand deutlich reduzieren und der Fortschritt besser überwachen. Dass die Synchronisierung des Text-Index asynchron über einen Datenbank-Job erfolgt und nicht mehr der Kontrolle des Content Servers unterliegt, hat sich nicht als spürbar nachteilig erwiesen.



Gunther Thielemann  
gunther.thielemann@slix.de

## DOAG DevCamp – Software Upcycling

Entwickler im Oracle-Umfeld sollten sich den 29. und 30. April 2015 rot im Kalender anstreichen: Unter dem diesjährigen Motto „Software Upcycling“ lädt das DOAG DevCamp in Frankfurt erneut zu einer Runde unter Gleichgesinnten. Die Teilnehmer treffen sich zu zwei interaktiven Tagen mit spannenden Sessions und einer Abendveranstaltung zum Netzwerken in der Jahrhunderthalle, um in lockerer Atmosphäre ihr Wissen auszutauschen.

Bei diesem Veranstaltungskonzept der DOAG heißt es: Ärmel hochkrempeln und mitmachen. Jeder Einzelne ist gefragt, denn es gibt weder Programm noch Speaker. Eine feste Agenda? Fehlanzeige! Auch Beamer und Standard-Präsentationen sind tabu. Stattdessen bringen die Teilnehmer ihre Ideen und Fragen am Veranstaltungsmorgen einfach mit, stimmen dann die Themen untereinander ab und diskutieren sie anschließend innerhalb der Sessions in

Gruppen. Das Ergebnis? Interaktion und lebendiger Wissensaustausch.

### Aus Teilnehmern werden Teilgeber

Im Mittelpunkt der Veranstaltung steht das Thema „Software Upcycling – Modernisierung in der Softwareentwicklung“. Der Schwerpunkt wird auf Architektur-Themen liegen. Alte Relikte aus längst vergangenen Zeiten, wie Oracle Forms oder COBOL, sollen abgestaubt werden und einen neuen Farbanstrich erhalten. Aber auch ADF, APEX, PL/SQL, agile Softwareentwicklung und Mobile werden thematisiert, sofern die Teilnehmer es möchten. Und das alles in lockerer Atmosphäre: Alle Teilnehmer sind an den Veranstaltungstagen „per du“, um echten Team Spirit zu erzeugen und den direkten Austausch zu fördern.

Weiterhin gilt: Eine Session beginnt, wenn sie beginnt, und endet, wenn sie endet. Das freie Format sorgt für eine große

Dynamik und Interaktion. Auch das Wechseln zwischen den einzelnen Sessions ist möglich, und sogar gewünscht: der Teilnehmer ist dann wie eine Biene, die für Befruchtungen zwischen den Sessions sorgt. Findet man einmal keine der laufenden Sessions interessant, bieten sich auch am Buffet noch viele Gelegenheiten, um Informationen zu teilen, Kontakte herzustellen und Ideen zu entwickeln.

Das neue Veranstaltungsformat des DOAG DevCamp wurde bereits im letzten Jahr erfolgreich getestet: Rund 70 Teilnehmer kamen in die Allianz Arena nach München und ließen sich von der einmaligen interaktiven und offenen Atmosphäre anstecken und inspirieren. Der hohe Informationsgehalt der Sessions und der rege Wissensaustausch kamen bei den Teilnehmern besonders gut an.

Mehr Informationen zur Veranstaltung unter <http://barcamp.doag.org>

# Redo Logs – Informationen, so weit der Logminer reicht

Thomas Klughardt, Dell Software

In diesem Artikel geht es um den Persistenz-Mechanismus von Oracle-Datenbanken, die Redo Logs. Dass diese wichtig sind, um verloren gegangene Änderungen wiederherstellen zu können, und dass bei deren Verlust Datenprobleme drohen, ist jedem bekannt. Weil sie so wichtig sind, müssen sie synchron geschrieben werden und können so schnell zum Flaschenhals werden. Deshalb wird hier auch betrachtet, wann sie sich negativ auf die Datenbank-Performance auswirken und was man dann tun kann. Abgesehen vom Recovery-Fall sind Redo Logs auch noch für andere Anwendungen interessant, um die es hier ebenfalls gehen soll.

Oracle-Datenbanken werden dann genutzt, wenn geschäftskritische Anwendungen betrieben werden, die im Fall von OLTP-Anwendungen kurze Antwortzeiten liefern und große Analysen ermöglichen sollen. Das ist nur dann möglich, wenn die Daten auf denen gearbeitet wird, im Speicher gehalten werden können. Dort finden die Zugriffe in Nanosekunden statt und nicht in Millisekunden wie bei klassischen Festplatten oder im Mikrosekundenbereich wie bei Solid State Disks (je nach Typ). Würde ein Vorgang mit vielen Random-Access-Zugriffen, also rein für die IO-Opera-

tionen im Speicher, eine Sekunde dauern, würde die gleiche Anzahl von IO-Zugriffen auf einer klassischen Festplatte etwa elfeinhalb Tage in Anspruch nehmen.

## Persistenz in einer Oracle-Datenbank

Dass möglichst viele Daten im Speicher gehalten werden sollen, dürfte hier klar sein; man geht bei einer OLTP-Anwendung davon aus, dass die Buffer-Cache-Hit-Ratio nicht unter 95 Prozent liegen sollte. Das bedeutet, nur fünf Prozent der Blöcke sollen von der Platte gelesen werden und sogar das kann zu viel sein. Doch Speicher-Inhalte haben einen Nachteil: Sie sind flüchtig, im Falle eines Crashes der Instanz oder eines Serverabsturzes sind sie verloren. Wenn eine Eigenschaft für Oracle-Datenbanken noch wichtiger sein dürfte als kurze Antwortzeiten, dann die, dass abgeschlossene Transaktionen nicht verloren gehen.

Was also tun? Man protokolliert die Änderungen auf einer Festplatte und schreibt dabei sequenziell. Dadurch vermeidet man das ständige, langsame Neupositionieren der Schreib-/Leseköpfe, speichert diese Änderungen aber trotzdem persistent auf der Platte. Diese Transaktionsprotokolle heißen bei Oracle-Datenbanken „Redo Logs“. Sie sind die wichtigsten Dateien in einer Oracle-Datenbank. Hat man lückenlos alle Redo-Log-Dateien zur Verfügung, kann man die Datenbank von der Erstellung auf

jeden beliebigen Stand bringen. Ein Verlust hingegen bedeutet so gut wie immer Datenverlust, wenn dazu ein Instance-Crash auftritt. Sollten also im laufenden Betrieb aus irgendeinem Grund Redo Logs verloren gehen, sollte immer sofort eine Vollsicherung erfolgen, notfalls auch offline.

## Performance-Überlegungen für Redo Logs

Trotz des sequenziellen Schreibens bei Redo Logs sind Festplattenzugriffe relativ langsam und so können die Transaktionsprotokolle leicht zum Flaschenhals in der Datenbank werden. Ist die Datenbank nicht in der Lage, die Änderungen schnell genug in den Redo Logs zu protokollieren, passiert erst einmal nichts weiter. Sie wartet, bis die Änderungen protokolliert sind. Erst dann ist eine Transaktion abgeschlossen.

Das bedeutet, man muss sich beim Anlegen der Redo Logs Gedanken machen. Es gibt mindestens zwei Redo-Log-Gruppen mit meist mehreren Mitgliedern, die zyklisch der Reihe nach beschrieben werden. Ist eine Redo-Log-Gruppe gefüllt, wird die nächste beschrieben. Dort werden wie in einem Ringpuffer die Einträge überschrieben. Um die Daten lückenlos wiederherstellen zu können, müssen alte Redo Logs archiviert sein (siehe Abbildung 1).

Bei Oracle-Datenbanken gibt es dafür den „ARCHIVELOG“-Modus, um sicherzustellen, dass alte Redo Logs archiviert werden, bevor man beginnt, neue zu schreiben.

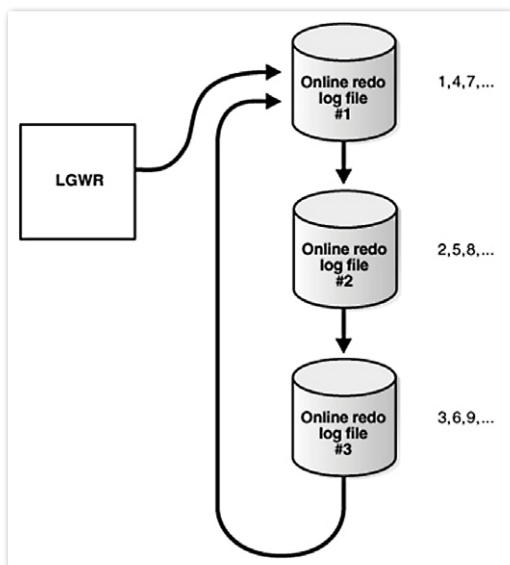


Abbildung 1: Wiederverwendung der Oracle Redo Logs (Oracle 12c Dokumentation)



Diese archivierten Redo Logs sind wichtig, weil man damit bei Datenverlust von der letzten Vollsicherung die Daten bis zum aktuellen Stand wiederherstellen kann, aber natürlich gibt es auch hier einen Flaschenhals. Kommt die Archivierung nicht hinterher, dann muss die Datenbank warten.

Bei der Planung der Redo Logs spielt auch deren Größe eine wichtige Rolle. Oracle führt in mehr oder weniger regelmäßigen Abständen sogenannte „Checkpoints“ durch, bei denen die Änderungen, die erfolgt sind, aus dem Cache in die Datendateien geschrieben werden. Diese finden zum Beispiel statt, nachdem es einen Redo-Log-Switch gegeben hat. Finden sie zu häufig statt, dann kann das nächste Redo Log unter Umständen nicht überschrieben werden, weil noch Daten auf die Platte geschrieben werden müssen, die in diesen Log stehen. Finden Checkpoints zu selten statt, dauert das Herunterschreiben dieser Daten länger und im Falle eines Instance-Crashes zieht es sich natürlich länger hin, bis die Daten wiederhergestellt sind und die Datenbank wieder zur Verfügung steht.

Diese Abwägung ist so wichtig, dass es in der Datenbank eine Sicht („V\$INSTANCE\_RECOVERY“) gibt, über die man sich eine Empfehlung für die Mindestgröße der Redo Logs anzeigen lassen kann. Allerdings sollte man dann auch bedenken, eine vernünftige Zielvorgabe für eine Recovery-Zeit zu setzen („FAST\_START\_MTTR\_TARGET“). Es gibt also einige Dinge, die beim Konfigurieren der Redo Logs beachtet werden müssen, damit die Datenbank durch diesen absolut notwendigen Persistenz-Mechanismus nicht ausgebremst wird.

### Sinnvolle und weniger sinnvolle Praktiken für Redo Logs

Gerade bei Online-Transaction-Processing-Datenbanken (OLTP) ist es wichtig, dass Transaktionen schnell abgearbeitet werden, damit die Performance der Anwendung gut ist. Um die Datenbank-Transaktionen schnell abzuarbeiten, sollten die Redo Logs nicht zum Flaschenhals werden. Dazu finden sich einige mehr oder weniger sinnvolle Praktiken im Internet darüber, wie das Schreiben von Redo Logs optimiert werden kann. So gut wie immer taucht der Hinweis auf, dass die Redo Logs auf schnelle IO-Systeme gehören. Das ist richtig und natürlich sollte man auch beachten, dass dann alle

Redo-Log-Member auf diese schnellen IO-Systeme gelegt werden. Wenn man schon die Online Redo Logs auf die schnellen Platten legt, ist es natürlich auch sinnvoll, die archivierten Redo Logs auf schnelle IO-Systeme zu legen, sonst läuft man direkt in den nächsten Flaschenhals.

In der Praxis ist diese Empfehlung jedoch vollkommen nutzlos. Oracle propagiert seit Jahren das „Stripe and Mirror Everything“-Prinzip (SAME) und fast alle Storage-Systeme für Datenbanken-Dateien befolgen es; damit findet ein Striping über alle Devices statt. In Extremfällen könnte man die Redo Logs zwar tatsächlich auf ein anderes, schnelleres Storage legen, aber für die meisten Datenbanken würde es die Verwaltung erschweren und es ist mitunter sogar gefährlich, wenn die Datenbank (üblicherweise von System-Administratoren statt über RMAN) direkt über Snapshot-Mechanismen des Storage-Systems gesichert wird. Dann kann es nämlich passieren, dass Redo und Archive Logs vergessen werden und die Datenbank nie wieder auf einen konsistenten Stand gebracht werden kann.

Sinnvoller ist es da schon, die Zahl der Redo-Log-Gruppen zu erhöhen. Die Redo Logs werden, wie bereits gesagt, zyklisch überschrieben und wenn es mehrere Gruppen gibt, dauert es einfach länger, bis eine Gruppe überschrieben wird. Das hat zwei Folgen: Zum einen findet nach einem Log-Switch immer ein Checkpoint statt. Das bedeutet, dass die geänderten Blöcke aus dem Buffer-Cache in die Datendateien auf der Festplatte geschrieben werden müssen. Bevor die Daten in den Redo Logs überschrieben werden dürfen, muss der Checkpoint fertig sein. Zum anderen hat dadurch der Archiver mehr Zeit, die Redo Logs zu archivieren, und man vermeidet, dass die Datenbank stehenbleibt, weil auf die Archivierung gewartet werden muss.

Größere Redo-Log-Dateien erfüllen einen ähnlichen Zweck. Auch hier vergrößert man die Zeit bis zum Überschreiben von Redo Logs. Größere Redo-Log-Dateien können aber auch bedeuten, dass ein Checkpoint länger dauert, weil insgesamt mehr Blöcke pro Checkpoint auf Platte geschrieben werden können.

Dazu gibt es noch eine Reihe weniger sinnvoller und teilweise gefährlicher Ratschläge. So wird zum Beispiel vorgeschlagen, RAM-Disks für die Redo-Log-Dateien

# PROMATIS Appliances

Prozessoptimierung & Simulation

## Oracle Applications

Oracle Business Analytics

Usability

Industrie 4.0

## Enterprise Content Management

Best-Practice-Mittelstandslösungen

## Oracle ERP Cloud

Planning & Budgeting Cloud Service

Managed Services

Oracle Infrastruktur

Oracle E-Business Suite

## Oracle BPM Suite

Application Integration Architecture

Social BPM

Oracle Sales Cloud

Besuchen Sie uns vom  
13. - 17. April 2015  
auf der Hannover Messe – Digital Factory  
„BITKOM Innovation Area Industrie 4.0“

# Hier sind wir zuhause

Unser Alleinstellungsmerkmal: Intelligente Geschäftsprozesse und beste Oracle Applikations- und Technologiekompetenz aus einer Hand. Als Oracle Pionier und Platinum Partner bieten wir seit über 20 Jahren erfolgreiche Projektarbeit im gehobenen Mittelstand und in global tätigen Großunternehmen.

Unsere Vorgehensweise orientiert sich an den Geschäftsprozessen unserer Kunden. Nicht Technologieinnovationen sind unser Ziel, sondern Prozess- und Serviceinnovationen, die unseren Kunden den Vorsprung im Markt sichern. Über Jahre gereifte Vorgehensmodelle, leistungsfähige Softwarewerkzeuge und ausgefeilte Best Practice-Lösungen garantieren Wirtschaftlichkeit und effektives Risikomanagement.

# PROMATIS



PROMATIS software GmbH

Tel.: +49 7243 2179-0

Fax: +49 7243 2179-99

www.promatis.de · hq@promatis.de

Ettlingen/Baden · Hamburg · Berlin

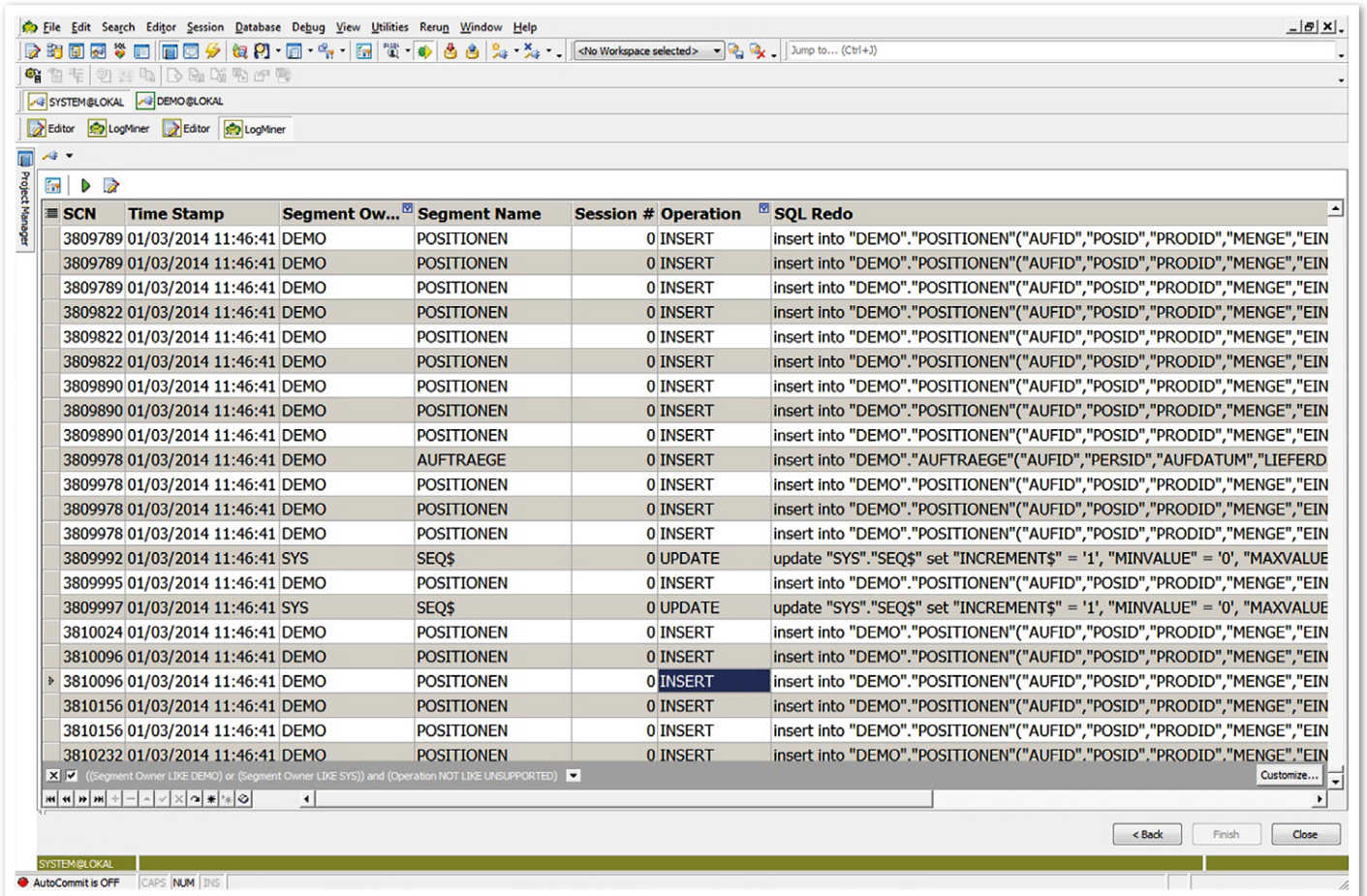


Abbildung 2: Ausgabe des Logminer Interface in Toad for Oracle

zu verwenden. Diese sind tatsächlich relativ schnell, aber man hebt dadurch den kompletten Persistenz-Mechanismus aus und bei einem Instance-Crash sind die Redo Logs verloren. Nicht nur, dass man die Änderungen verliert, sondern man bekommt die Datenbank auch nie auf einen konsistenten Stand und kann beim letzten Backup, das hoffentlich einen relativ aktuellen Stand hat, wieder anfangen. Von RAM-Disks für Redo Logs sollte man also auf jeden Fall die Finger lassen.

In eine ähnliche Kerbe schlägt auch die Empfehlung, Tabellen für die vielen Redo-Log-Informationen durch INSERT-Operationen zu schreiben, im „NOLOGGING“-Modus zu betreiben und die INSERT-Operationen mit dem „/\*+APPEND\*/“-Hint durchzuführen. Das hat zur Folge, dass die Datensätze oberhalb der High-Water-Marke eingefügt werden und dafür keine Redo-Log-Informationen geschrieben werden. Diese Operationen sind damit auch nicht wiederherstellbar, aber zumindest weiß man dann, welche Tabellen betroffen sind. Für Staging-Tabellen, deren Daten sich durch er-

neutes Durchführen der INSERT-Jobs leicht wiederherstellen lassen, ist das durchaus eine gute Option. Ein bisschen Vorsicht ist jedoch geboten, wenn die Datenbank oder der Tablespace in den „FORCE LOGGING“-Modus gesetzt werden, dann werden plötzlich trotzdem Redo-Log-Informationen geschrieben, was dann zu einer Explosion des Redo-Log-Volumens führen kann.

Ein anderer Rat ist, weniger oft Commits auszuführen. Bei „COMMIT“-Operationen muss ja der Log-Buffer-Inhalt in die Redo Logs auf Platte geschrieben werden, bevor es weitergeht. Weniger oft Commits auszuführen, also wenige große statt viele kleine Transaktionen auszuführen, führt zu einem eher asynchronen Verhalten und damit wird die Anzahl der Schreib-Operationen besser verteilt. Bei Anwendungen von Drittanbietern hat man allerdings meist keinen Einfluss darauf, wann Commits durchgeführt werden, deshalb ist dieser Ratschlag nur für bestimmte Umgebungen relevant.

Um überhaupt zu wissen, ob Performance-Probleme mit den Redo Logs zusammenhängen, gibt es eine Reihe von

Wait-Events. Nur wenn man sie ausgewertet, kann man auch gezielt gegen Performance-Probleme vorgehen.

### Performance-Troubleshooting mit Wait Events

Wenn es Performance-Probleme in der Anwendung gibt, lohnt es sich, die Wait Events zu analysieren. Sie erzählen davon, womit die Datenbank ihre Zeit verbringt, beziehungsweise worauf sie warten muss. Im Zusammenhang mit den Redo Logs gibt es dabei fünf besonders wichtige Wait Events, die hier aufgeführt sind:

- *log file sync*  
Bei einem Log-File-Sync-Wait-Event wird erwartet, weil nach dem Beenden einer Transaktion mit „Commit“ oder „Rollback“ die Redo-Log-Einträge noch nicht auf Platte geschrieben worden sind. Der Flaschenhals ist hier wirklich das Storage-System.
- *log buffer space*  
Der Log-Buffer ist voll, weil die Änderungen schneller passieren, als sie in



die Redo Logs geschrieben werden können. Entweder ist der Log-Buffer zu klein, um genug Änderungen zu puffern, oder die Redo-Log-Files werden zu langsam geschrieben.

- *log file switch (archiving needed)*  
Das Redo Log kann noch nicht überschrieben werden, weil die Inhalte noch nicht archiviert sind. Es könnte aber auch sein, dass die Archive-Log-Destination voll ist und keine Archive Logs mehr geschrieben werden können.
- *log file switch (checkpoint incomplete)*  
Das Redo Log kann nicht überschrieben werden, weil die zugehörigen geänderten Blöcke vom Database Writer (DBWR) noch nicht auf Platte geschrieben wurden.
- *log file switch completion oder log file switch (private strand flush incomplete)*  
Je nach Oracle-Version kann man einen der beiden Wait Events sehen, der aber nicht so häufig auftritt wie die anderen. Hier ist eigentlich alles erfüllt, um den Log-Switch durchzuführen, es wird nur darauf gewartet, dass der Redo Log Writer die entsprechenden internen Operationen durchgeführt hat.

## Informationen aus den Redo Logs

Bis jetzt hat es sich so angehört, als seien die Redo Logs ein lästiges, aber notwendiges Übel in der Datenbank, doch sie sind wesentlich mehr als das. Wie schon erwähnt kann man, wenn man die Redo Logs lückenlos zur Verfügung hat, die Datenbank auf jeden beliebigen Stand bringen, man nennt das auch „vorwärts rollen“ (roll forward). Genau diese Möglichkeit wird bei einem Recovery ja benötigt. Jede Änderung, die irgendwann durchgeführt wurde, ist also in den Redo Logs verzeichnet. Dazu ist auch notiert, zu welcher Transaktion sie gehört, inklusive der System Change Number, einer eindeutigen, fortlaufenden Nummer, von der jeder Transaktion genau eine zugeordnet ist. Darüber hinaus ist verzeichnet, wer diese Änderung durchgeführt hat, wann sie passiert ist und sogar, wie man sie rückgängig machen kann. Wurde ein Datensatz gelöscht, sieht man hier die entsprechenden Daten, die vorher vorhanden waren. Wurde ein Datensatz aktualisiert, dann tauchen hier die Werte auf, die er vorher gehabt hat. Das macht die

Redo Logs für viele Anwendungen zu einer Goldgrube an Informationen.

## Nachvollziehbarkeit und Auditing

Oft gibt es rechtliche Gründe oder betriebliche Vorgaben, die es nötig machen, Änderungen zu protokollieren. Dafür gibt es von Oracle Auditing-Mechanismen, doch auch die Redo Logs sind in der Lage, die nötigen Informationen zu liefern. Sie enthalten ja die Mitteilungen darüber, wer wann was geändert hat, und sogar, wie die Werte vor der Änderung waren (before image). Das hört sich sehr gut an, allerdings hat die Sache einen kleinen Haken: Man kann nur Änderungen sehen, die in den Redo Logs protokolliert sind. Wurden Direct Loads durchgeführt, also „INSERT“-Statements mit einem Append Hint abgesetzt, tauchen die INSERTs nicht in den Redo Logs auf. Will man diese Möglichkeit unterbinden, dann gibt es die Möglichkeit, für die Datenbank oder einzelne Tablespace die „FORCE\_LOGGING“-Option zu aktivieren, und damit werden dann auch diese Operationen protokolliert und nachvollziehbar.

Über den Logminer („DBMS\_Logminer“) können die Informationen aus den Redo Logs lesbar gemacht und ausgegeben werden. Allerdings ist die Ausgabe immer noch recht kryptisch, deshalb empfiehlt es sich hier, mit Tools zu arbeiten, die die Daten in Data Grids anzeigen oder in Excel-Sheets laden können. Zum Beispiel hat Toad for Oracle einen Logminer-Assistenten, der die Einträge anzeigen und exportieren kann (siehe *Abbildung 2*).

## Replikation

Nachdem die Redo Logs, wie gesagt, alle nötigen Daten enthalten, sind sie auch für Replikation interessant. So ist es damit möglich, ohne viel Last in der Datenbank zu erzeugen, alle Änderungen auf andere Systeme zu transportieren und dort Replikate zu pflegen. Dabei gibt es physische Replikationslösungen wie Dataguard, der physikalisch identische Replikate verwaltet, die im Falle eines Ausfalls der Primär-Datenbank übernehmen können. Der Nachteil ist, dass an der Ziel-Datenbank nichts geändert werden darf, sie muss ja binär identisch bleiben.

Etwas komplizierter, aber dafür auch flexibler, sind logische Replikationslösungen wie Shareplex, Oracle GoldenGate

oder DBVisit Replicate, die aus den Redo Logs tatsächlich SQL-Statements erzeugen und so auf der Zielseite auch in Datenbanken schreiben können, die ein komplett anderes physikalisches Layout haben. Dort kann auf beide Datenbanken lesend und schreibend zugegriffen werden, sie können andere Versionen und Architekturen haben und zum Beispiel zusätzliche Indizes für große Abfragen enthalten. Dazu sind hier interessante Szenarien wie Migrationen fast ohne Auszeit oder entfernte Replikation der Daten zwischen mehreren Master-Datenbanken möglich. Dabei gibt es teilweise große Unterschiede zwischen der Art, wie diese Lösungen arbeiten; nicht jede Replikationslösung eignet sich für jedes Anwendungsszenario.

## Fazit

Insgesamt kann man sagen, dass die Redo Logs zu den elementaren Bestandteilen einer Oracle-Datenbank gehören. Ohne sie könnte die Datenbank nicht performant und gleichzeitig persistent arbeiten. Dadurch, dass sie synchron geschrieben werden müssen, können sie natürlich zum Flaschenhals werden, zum Glück ist die Datenbank aber auskunftsfreudig darüber, woran es liegen kann.

Da Redo Logs alle wiederherstellbaren Änderungen einer Datenbank enthalten müssen, kann man sie auch wunderbar als Informationsquelle nutzen. Um das zu vereinfachen, gibt es Tools, die über das Logminer-Interface zugreifen, und Lösungen, die das Redo-Log-Format direkt auslesen können und sich diese Informationsfülle so zunutze machen.



Thomas Klughardt  
thomas.klughardt@software.dell.com

# Data Guard 12c: „Zero-Data-Loss“ trotz großer Distanz der Standby-Datenbank

Uwe Hesse, ORACLE Deutschland B.V. & Co. KG

Seit seiner Einführung in Oracle Version 8 ist Data Guard geeignet, Datenbanken gegen regionale Katastrophen mit Standby-Datenbanken in beliebig großer Entfernung abzusichern.

Bislang war bei einer sehr großen Entfernung der Standby-Datenbank der einzig sinnvolle Schutzmodus „Maximum Performance“, während „Maximum Availability“ und „Maximum Protection“ aufgrund der dabei erforderlichen synchronen Übertragung des Redo-Protokolls die Performance der primären Datenbank zu stark beeinträchtigt hätten. Dies hat sich mit der Version 12c geändert: „Far Sync“-Instanzen beziehungsweise „Real Time Cascade“-Standby-Datenbanken ermöglichen nun die höheren Schutzmodi trotz großer Distanz der Disaster Recovery Site. Dieser Artikel beschreibt die dazu nötigen Schritte, ausgehend von einer bereits bestehenden Data-Guard-Konfiguration im „Maximum Performance“-Schutzmodus. Momentan sendet die primäre Datenbank direkt an die Standby-Datenbank, und zwar aufgrund der großen Entfernung asynchron – „prima“ befindet sich beispielsweise in den USA und „physt“ in Deutschland (siehe Listing 1).

## Far Sync

Eine zwischengeschaltete „Far Sync“-Instanz wird sich in für synchrone Übertragung geeigneter Nähe zur primären Datenbank befinden – etwa in einer anderen Stadt, jedoch nicht auf einem anderen Kontinent – und das empfangene Redo mittels asynchroner Übertragung an die eigentliche Standby-Datenbank weiterleiten (siehe Abbildung 1).

Auf diese Weise kann trotz großer Distanz zwischen primärer Datenbank und Standby-Datenbank der Schutzmodus „Maximum Availability“ – und damit Zero-Data-Loss – erreicht werden. Eine „Far Sync“-Instanz benötigt nur ein spezielles Controlfile und

Standby Redo Logfiles – also keine Datendateien – und kann daher auch kein Redo Apply durchführen. Sie wird folgendermaßen implementiert (siehe Listing 2).

Das Ganze wurde in einer Session an der primären Datenbank durchgeführt. Nun wird das „Far Sync“-Controlfile auf den für die „Far Sync“-Instanz vorgesehenen Rech-

```
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxPerformance
  Databases:
    prima - Primary database
    physt - Physical standby database
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
DGMGRL> show database physt logxptmode;
LogXptMode = 'ASYNC'
```

Listing 1

```
SQL> alter database create far sync instance controlfile
  2 as '/stage/control01.ctl';
Database altered.
```

Listing 2

```
SQL> connect sys/oracle@fsl as sysdba
Connected to an idle instance.
SQL> startup mount
ORACLE instance started.
Total System Global Area      521936896 bytes
Fixed Size                     2290264 bytes
Variable Size                  364907944 bytes
Database Buffers               146800640 bytes
Redo Buffers                    7938048 bytes
Database mounted.
SQL> select database_role from v$database;
DATABASE_ROLE
-----
FAR SYNC
```

Listing 3



ner kopiert, ebenso wie das Password-File der primären Datenbank. „Oracle Net Connectivity“ muss ebenfalls hergestellt sein. Diese Schritte unterscheiden sich nicht sonderlich von der üblichen Vorgehensweise zur Implementierung einer Standby-Datenbank, außer dass kein „RMAN duplicate“ erforderlich ist. *Listing 3* zeigt die Eingaben auf dem „Far Sync“-Rechner. Die „Far Sync“-Instanz ist nun bereit, empfängt und sendet aber noch kein Redo. *Listing 4* zeigt die nächsten Schritte. Die Konfiguration entspricht nun der in *Abbildung 1*. Was aber, wenn ein Switchover erfolgen soll (*siehe Listing 5*)?

Dies ist im momentanen Zustand der Konfiguration nicht erlaubt. Es gibt nun zwei Möglichkeiten. Erstens, der Schutzmodus wird vor dem Switchover auf „Maximum Performance“ abgesenkt und die neue primäre Datenbank überträgt anschließend direkt asynchron zur neuen Standby-Datenbank (*siehe Abbildung 2*). Zweitens, eine weitere „Far Sync“-Instanz auf der entfernten Standby-Seite wird für den Fall eines Switchover vorbereitet (*siehe Abbildung 3*). *Listing 6* zeigt die Data-Guard-Broker-Konfiguration gemäß der Abbildung nach erfolg-

```
DGMGRL> add far_sync fs1 as connect identifier is fs1;
far_sync instance "fs1" added
DGMGRL> edit database prima set property redoroutes = '(LOCAL:fs1 SYNC)';
Property "redoroutes" updated
DGMGRL> edit far_sync fs1 set property redoroutes = '(prima:physt ASYNC)';
Property "redoroutes" updated
DGMGRL> enable far_sync fs1;
Enabled.
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxPerformance
  Databases:
    prima - Primary database
      fs1   - Far Sync
      physt - Physical standby database
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
DGMGRL> edit configuration set protection mode as maxavailability;
Succeeded.
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxAvailability
  Databases:
    prima - Primary database
      fs1   - Far Sync
      physt - Physical standby database
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
```

Listing 4



Abbildung 1: Primary mit Far Sync Instance

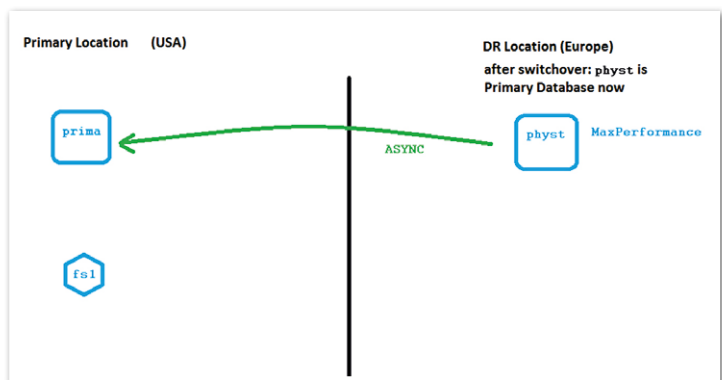


Abbildung 2: Absenkung des Schutzmodus ermöglicht Switchover

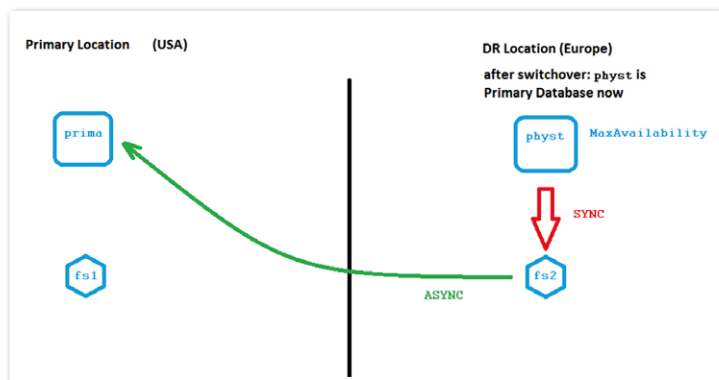


Abbildung 3: Standby mit „Far Sync“ zur Beibehaltung von „MaxAvailability“ bei Switchover

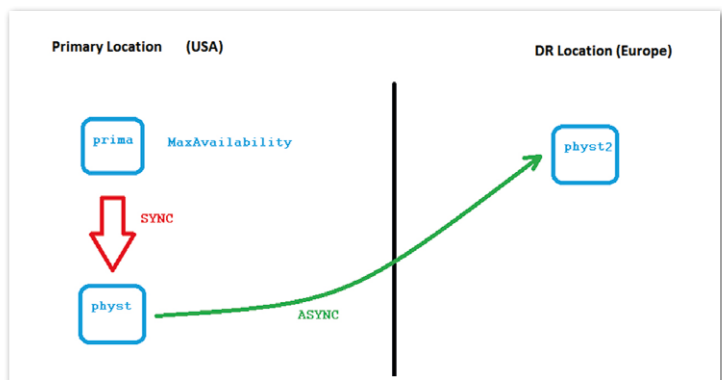


Abbildung 4: Primary mit naher „Real-Time Cascading“-Standby-Datenbank

```
DGMGRL> switchover to physt;
Performing switchover NOW, please wait...
Error: ORA-16627: operation disallowed since no standby databases would remain to support protection mode
Failed.
Unable to switchover, primary database is still "prima"
```

Listing 5

```
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxAvailability
  Databases:
    prima - Primary database
    fs1 - Far Sync
    physt - Physical standby database
    fs2 - Far Sync (inactive)
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
```

Listing 6

```
DGMGRL> switchover to physt;
Performing switchover NOW, please wait...
Operation requires a connection to instance "physt" on
database "physt"
Connecting to instance "physt"...
Connected as SYSDBA.
New primary database "physt" is opening...
Operation requires startup of instance "prima" on da-
tabase "prima"
Starting instance "prima"...
ORACLE instance started.
Database mounted.
Switchover succeeded, new primary is "physt"
```

Listing 7

```
DGMGRL> edit database prima set property redoroutes = '(LOCAL:physt SYNC)';
Property "redoroutes" updated
DGMGRL> edit database physt set property redoroutes = '(prima:physt2 ASYNC)';
Property "redoroutes" updated
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxPerformance
  Databases:
    prima - Primary database
    physt - Physical standby database
    physt2 - Physical standby database (receiving current redo)
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
DGMGRL> edit configuration set protection mode as maxavailability;
Succeeded.
```

Listing 10

ter Implementierung. So ist ein Switchover ohne Absenkung des Schutzmodus erlaubt (siehe Listing 7). Wie in Listing 8 ersichtlich, ist

der Befehl für das Switchover derselbe wie bei Konfiguration ohne „Far Sync“-Instanz, das gilt auch für das Failover.

```
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxAvailability
  Databases:
    prima - Primary database
    fs1 - Far Sync
    physt - Physical standby database
    fs2 - Far Sync (inactive)
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
DGMGRL> failover to physt;
Performing failover NOW, please wait...
Failover succeeded, new primary is "physt"
```

Listing 8

```
DGMGRL> show configuration;
Configuration - myconf
  Protection Mode: MaxPerformance
  Databases:
    prima - Primary database
    physt - Physical standby database
    physt2 - Physical standby database
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
```

Listing 9

## Real-Time Cascade

Kaskadierende Standby-Datenbanken, also Standby-Datenbanken, die ihrerseits Redo-Protokoll an Standby-Datenbanken senden, sind nicht neu. Vor der Version 12c erfolgte diese Übertragung von Standby-Datenbank zu Standby-Datenbank allerdings nur mithilfe des „Archiver“-Hintergrundprozesses. „Real-Time Cascade“ bedeutet, dass die erste Standby-Datenbank per LNS-Hintergrundprozess an die zweite Standby-Datenbank übertragen kann, typischerweise asynchron (siehe Abbildung 4).

Zunächst wird hierzu eine weitere Standby-Datenbank erstellt und der DataGuard-Broker-Konfiguration hinzugefügt. Diese Schritte unterscheiden sich nicht von den bereits vor 12c bekannten. Listing 9 zeigt diese Ausgangslage. „physt“ und „physt2“ empfangen momentan Redo direkt von „prima“. Dies wird nun geändert

```

SQL> connect sys/oracle@prima as sysdba
Connected.
SQL> select database_role from v$database;
DATABASE_ROLE
-----
PRIMARY
SQL> select sequence# from v$log where status='CURRENT';
SEQUENCE#
-----
27
SQL> connect sys/oracle@physt as sysdba
Connected.
SQL> select database_role from v$database;
DATABASE_ROLE
-----
PHYSICAL STANDBY
SQL> select sequence#,status from v$managed_standby where process='MRP0';
SEQUENCE# STATUS
-----
27 APPLYING_LOG
SQL> connect sys/oracle@physt2 as sysdba
Connected.
SQL> select sequence#,status from v$managed_standby where process='MRP0';
SEQUENCE# STATUS
-----
27 APPLYING_LOG
    
```

Listing 11

```

DGMGRL> edit configuration set protection mode as maxprotection;
Succeeded.
DGMGRL> show configuration;
Configuration - myconf
Protection Mode: MaxProtection
Databases:
prima - Primary database
physt - Physical standby database
physt2 - Physical standby database (receiving current redo)
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
    
```

Listing 12

„Far Sync“-Instanz	„Real-Time Cascading“-Standby-Datenbank
Nur Controlfile und Standby Redologs	Vollständige Standby-Datenbank
Redo Apply nicht möglich	Redo Apply wird durchgeführt
„Maximum Protection“ nicht möglich	Alle Schutzmodi möglich

Tabelle 1

(siehe Listing 10). Die Konfiguration entspricht nun der Abbildung 3. Listing 11 verdeutlicht die Redo-Apply-Aktivität der beiden Standby-Datenbanken.

Wie ersichtlich, führen beide Standby-Datenbanken „Real-Time Apply“ durch, obwohl das Redo-Protokoll kas-

kadierend übertragen wird. Genau das war vor 12c nicht möglich. Im Unterschied zur Konfiguration mit „Far Sync“-Instanzen kann bei „Real-Time Cascade“ auch im Schutzmodus mit „Maximum Protection“ operiert werden (siehe Listing 12).

In diesem Fall ist allerdings eine zweite kaskadierende Standby-Datenbank (die Redo synchron empfängt) dringend zu empfehlen, um die Verfügbarkeit der primären Datenbank auch bei Ausfall einer der synchron empfangenden Standby-Datenbanken weiterhin zu gewährleisten. Tabelle 1 stellt „Far Sync“ und „Real-Time Cascade“ gegenüber. Abschließend bleibt zu erwähnen, dass beide Verfahren die zusätzliche Lizenzierung der Active-Data-Guard-Option erfordern.



Uwe Hesse  
uwe.hesse@oracle.com

# Upgrade nach Oracle Database 12c – oder lieber noch warten?

Mike Dietrich, Oracle Corporation

Von Zeit zu Zeit steht ein Datenbank-Upgrade an. Aktuell ist Oracle Database 12.1.0.2 seit mehr als einem halben Jahr verfügbar. Und zusätzlich läuft der Premier Support für Oracle Database 11.2 Ende Januar 2015 aus. Nun stellt sich die Frage: „Datenbank-Upgrade jetzt – oder besser auf das zweite Release warten?“

Ende Januar 2015 endet der Premier Support für Oracle Database 11.2. Das ist noch kein Grund in Panik zu verfallen, aber es sollte zumindest schon ein Plan für das oder die anstehenden Datenbank-Upgrades in der Schublade liegen. Mit dem Ende des Premier Supports endet normalerweise die Fehlerbehebung (Bug Fixing). Allerdings lässt sich relativ günstig, aber kostenpflichtig, Extended Support für ein bis drei Jahre zukaufen. Dies umfasst die Behebung von kritischen Problemen, Sicherheitslücken sowie den Zugang zu den Bundle Patches (BP), den Security Patch Updates (SPU) und Patch Set Updates (PSU), wobei SPUs und PSUs im Quartalszyklus ungefähr in der Monatsmitte im Januar, April, Juli und Oktober erscheinen.

Im Fall von Oracle Database 11.2 offeriert Oracle allen Kunden und Partnern kostenfreien Zugang zum Extended Support im ersten Jahr, allerdings mit ein paar wichtigen Einschränkungen bezüglich des Software-Stands (siehe *Abbildung 1*):

- Kein Extended Support für Oracle Database 11.2.0.1 und 11.2.0.2
- Extended Support kostenfrei bis 27. August 2015 für Oracle Database 11.2.0.3; danach endet der Extended Support für Oracle Database 11.2.0.3
- Extended Support kostenfrei bis 31. Januar 2016 für Oracle Database 11.2.0.4; danach ist Extended Support kostenpflichtig verlängerbar bis maximal 31. Januar 2018 für Oracle Database 11.2.0.4

Seit Oracle Database 11.2 ist jedes Patch Set ein sogenanntes „Full Release“. Es bedarf also keiner Installation des „Base Release“ mehr, um beispielsweise Oracle Database 12.1.0.2 einzusetzen. Oracle Database 11.2.0.4 ist das „Terminal Patch Set“, also das finale Patch Set für Oracle Database 11.2. Weitere Details stehen in der My-Oracle Support (MOS) Note 742060.1.

## Auf Release 2 warten

Nun stellt sich die Frage: „Sollte man das Upgrade oder die Migration nun angehen oder besser noch auf das sogenannte „Release 2“, also Oracle Database 12.2, warten?“ Aus der Vergangenheit werden viele Administratoren und Anwender spontan mit „Warten!“ antworten. Aber ist das noch angebracht oder haben sich die

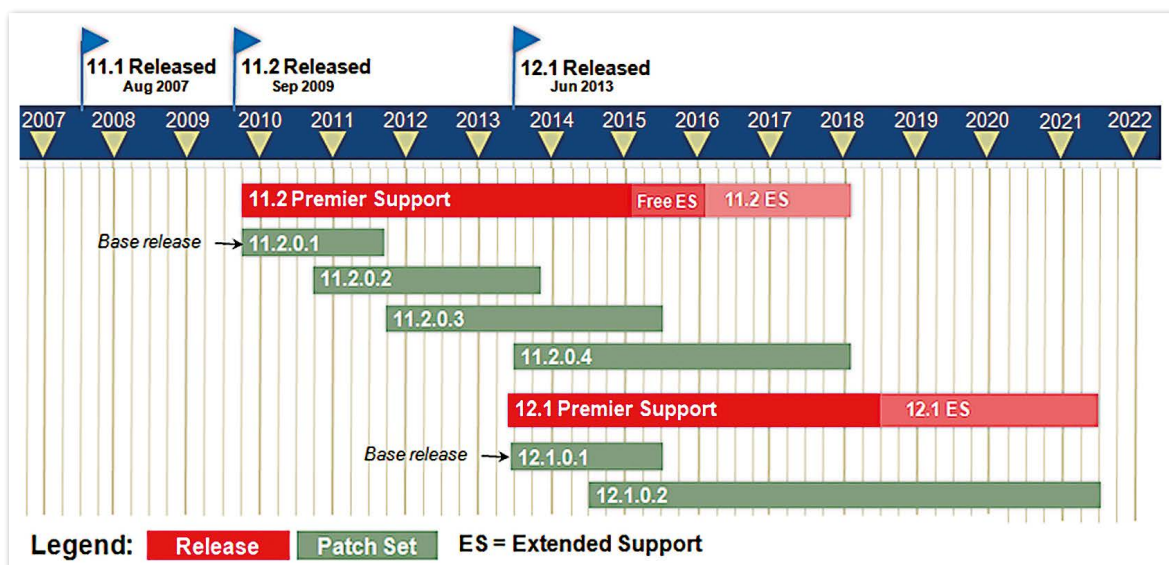


Abbildung 1: Extended Support für Oracle Database 11.2.



Platform	12.2.0.1	12.1.0.2 <sup>11,12</sup>	12.1.0.12	11.2.0.4 <sup>10</sup>	11.2.0.3	11.2.0.2	11.2.0.12	11.1.0.7 <sup>1</sup>
Linux x86	Not planned	Not planned	Not planned	28-Aug-2013	23-Sep-2011	13-Sep-2010	1-Sep-2009	18-Sep-2008
Linux x86-64	1HCY2016	22-Jul-2014	25-Jun-2013	27-Aug-2013	23-Sep-2011	13-Sep-2010	1-Sep-2009	18-Sep-2008
Oracle Solaris SPARC (64-bit)	1HCY2016	22-Jul-2014	25-Jun-2013	29-Aug-2013	1-Oct-2011	24-Sep-2010	6-Nov-2009	06-Oct-2008
Oracle Solaris x86-64 (64-bit)	1HCY2016	22-Jul-2014	25-Jun-2013	29-Aug-2013	1-Oct-2011	24-Sep-2010	25-Nov-2009	Not planned

Abbildung 2: Geplante Release-Verfügbarkeit für Oracle 12.2. aus MOS Note 742060.1

Gegebenheiten im Vergleich zu den Versionen 9i oder 10g nicht doch etwas verschoben? Dazu ein Blick auf die Release-Zyklen:

- 18 Monate zwischen Oracle Database 9.2 und Oracle Database 10.1
- 18 Monate zwischen Oracle Database 10.1 und Oracle Database 10.2
- 25 Monate zwischen Oracle Database 10.2 und Oracle Database 11.1
- 26 Monate zwischen Oracle Database 11.1 und Oracle Database 11.2
- 45 Monate zwischen Oracle Database 11.2 und Oracle Database 12.1

Man sieht, dass sich die Release-Zyklen über die letzten dreizehn Jahre deutlich ausgedehnt haben. Die späte Verfügbarkeit von Oracle Database 12.1 ist sicherlich ein Ausreißer, aber der Trend ist er-

kennbar. Bezüglich Oracle Database 12.2 liefert MOS Note 742060.1 eine Antwort – zumindest zur geplanten Release-Verfügbarkeit (siehe Abbildung 2).

Aller Voraussicht nach werden die meisten Kunden sicherlich die Verfügbarkeit des ersten Patch Sets für Oracle Database 12.2.0.1, also Oracle Database 12.2.0.2, abwarten. Unter Berücksichtigung der obigen Tabelle kann jeder nun für sich selbst eine Abschätzung treffen, wann das erste Patch Set 12.2.0.2 verfügbar sein könnte. In der Vergangenheit lagen etwa zwölf bis vierzehn Monate zwischen dem Base Release und dem ersten Patch Set. Wichtig ist zu bemerken, dass es sich hierbei um keine offizielle Ankündigung eines Oracle Patch Sets handelt, sondern lediglich um ein Gedankenspiel. Würde dieser Zyklus auch für Oracle 12.2 eingehalten werden, befänden wir uns dann schon in der Mit-

te, vielleicht sogar in der zweiten Hälfte des Jahres 2017.

Unter der Annahme, dass Tests erst ab diesem Zeitpunkt vorgenommen werden würden, könnte ein geplanter Go-Live Termin mit Oracle Database 12.2.0.2 frühestens Ende 2017 sein – also sehr nahe am geplanten Ende des kostenpflichtigen Extended Supports für Oracle Database 11.2.0.4 liegen.

Daraus ergibt sich fast schon zwingend, dass das Datenbank-Upgrade auf Oracle Database 12.1.0.2 unumgänglich ist, um nicht potenziell in eine Periode ohne Bug-Fixing-Support zu gelangen.

### Meine Applikation ist nicht für Oracle Database 12.1.0.2 zertifiziert

Was aber nun, wenn die Applikation bisher noch nicht für Oracle Database 12.1.0.2

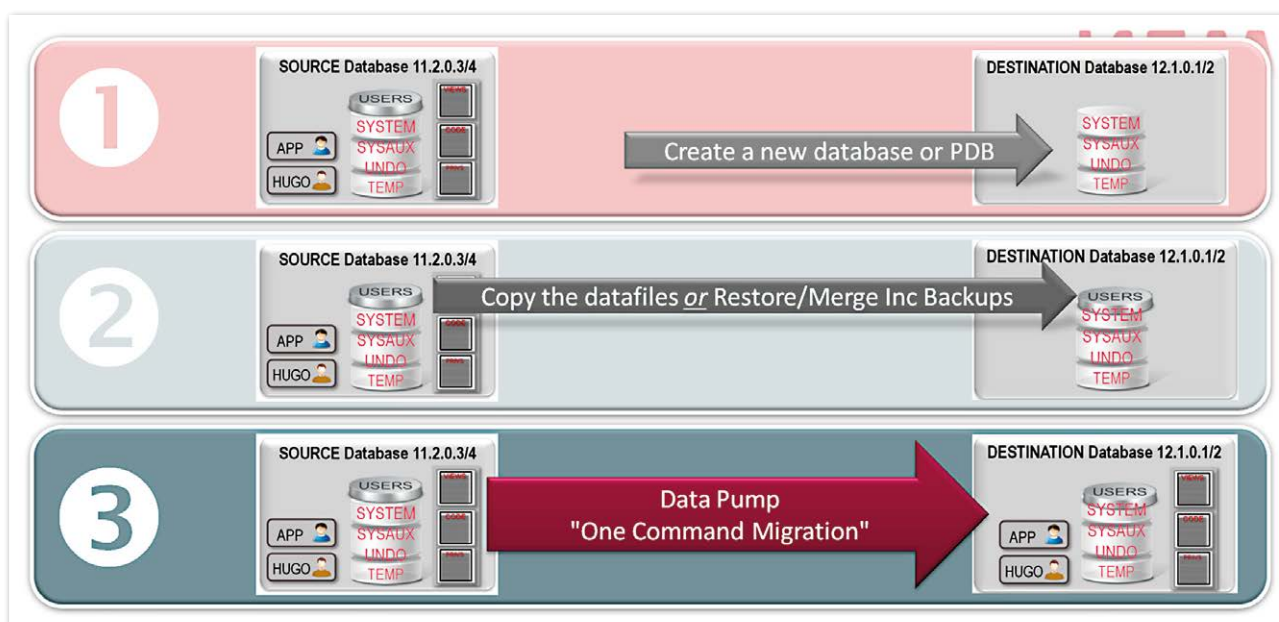


Abbildung 3: Full Transportable Export/Import

freigegeben wurde? Dies ist in der Regel das am häufigsten ins Feld geführte Argument, warum noch kein Upgrade durchgeführt wurde. Es sollte allerdings für alle Applikationshersteller ein sehr deutliches Signal sein, dass SAP Oracle Database 12.1.0.2 zertifizieren wird (siehe „<http://www.oracle.com/us/solutions/sap/sap-database/index.html>“): „Oracle Database 12c will be certified by SAP in CY 2015. By certifying Oracle Database 12.1.0.2, SAP will break the terminal release custom (certification of Oracle Database release x.2, no certification of release x.1) valid to date, to ensure sufficient overlap with Oracle Database 11g.“

Teilweise setzt die Freigabe durch den Software-Hersteller ein Upgrade der Software voraus, aber viele Hersteller haben Oracle Database 12.1.0.2 bereits freigegeben. Oracle hat beispielsweise Siebel ab Version 8.1 und Oracle Ebusiness Suite ab Release 12.1 (inklusive 12.2.2) zertifiziert. Im Einzelfall ist der Software-Anbieter zu konsultieren.

### Wo bitte gehts zum Upgrade?

Von einem Upgrade spricht man, wenn die Datenbank-Version auf der existierenden Betriebssystem-Familie auf ein höheres Release verändert wird. Dabei kann auch die Hardware verändert werden, beispielsweise, wenn ein neuer Server angeschafft wird und die Datenbank dorthin kopiert und dann ein Upgrade erfolgt. Um auf Oracle Database 12.1.0.2 upgraden zu können, müssen folgende Release-Voraussetzungen gegeben sein:

- Oracle Database 10.2.0.5
- Oracle Database 11.1.0.7
- Oracle Database 11.2.0.2 oder neuer
- Oracle Database 12.1.0.1

Neu beim Datenbank-Upgrade ist vor allem das Preupgrade-Package „preupgrd.sql“, das im Verzeichnis „\$ORACLE\_HOME/rdbms/admin“ der neuen 12c-Installation liegt oder – einfacher und in neuerer Version verfügbar – über die MOS Note 884522.1 geladen werden kann. Es liefert im Gegensatz zu den „utlu1nni.sql“-Skripten der vorherigen Releases detaillierte Informationen und vor allem Empfehlungen.

Die wichtigste Neuerung stellt allerdings das „Parallel Upgrade“ dar. Seit Oracle Da-

```
impdp oow/passwd@PDB1
NETWORK_LINK=DB1 VERSION=12 FULL=Y
TRANSPORTABLE=ALWAYS METRICS=Y
LOGFILE=oow_dir:src112fullimp.log
TRANSPORT_DATAFILE='/oradata/ts1.dbf' ...
```

Listing 1

tabase 12c wird das Datenbank-Upgrade nicht mehr aus SQL\*Plus heraus gestartet, sondern über ein PERL-Skript namens „catctl.pl“. Es ist in der Lage, das Upgrade mit maximal acht parallelen Threads durchzuführen („\$> \$ORACLE\_HOME/perl/bin/perl catctl.pl -n 8 catupgrd.sql“). Per Default führen vier „Worker“ das Upgrade durch. Die Zeitersparnis beim Oracle Server selbst beträgt bis zu 40 Prozent. Die Komponenten-Upgrades wie Java oder XDB sind aktuell noch nicht parallelisiert.

### Thema „Migration“

Wenn die Betriebssystem-Plattform geändert werden soll, beispielsweise von einer „Big Endian“-Plattform wie HP-UX oder AIX auf eine „Little Endian“-Plattform wie Oracle Linux, spricht man von einer Migration. Bei Oracle Database 12c funktionieren natürlich die bekannten Methoden wie Export/Import mit Data Pump, Transportable Database, Transportable Tablespaces oder spezielle Oracle-Tools wie Oracle-to-Oracle (O2O). Zusätzlich kommt seit Oracle Database 12.1.0.1 noch ein neuer Weg hinzu, der bei Plattformwechseln sehr hilfreich ist, der sogenannte „Full Transportable Export/Import“.

Dabei muss die Quell-Datenbank mindestens eine Oracle-Datenbank 11.2.0.3 sein. In der Zielumgebung wird – wie bei Transportable Tablespaces auch – eine neue Datenbank oder – wenn Oracle Multitenant verwendet wird – eine Pluggable Database (PDB) angelegt. Nach dem Kopieren der Dateien erledigt Data Pump alle manuellen Schritte, die eine Umstellung per Transportable Tablespaces so mühsam gemacht haben, mit einem Kommando, vom Export/Import der Meta-Informationen über den Tablespace-Inhalt zum Nachbauen der User, Applikationsobjekte, Synonyme, Views etc. (siehe Abbildung 3). Listing 1 zeigt ein Beispiel für eine One-Command-Migration über einen Datenbank-Link.

### Weiterführende Informationen

Wer das Upgrade oder eine Migration zu Oracle Database 12c angehen möchte, findet eine umfangreiche Präsentation auf „<http://blogs.oracle.com/UPGRADE>“ auf der rechten Seite im „Slides Download Center“. Das Thema lautet „Upgrade, Migrate & Consolidate to Oracle Database 12c“.

### Fazit

Viele Kunden stehen gerade vor der Entscheidung, ihre Datenbank-Umgebungen aufzugraden. Es ist dabei sehr wichtig zu bedenken, dass der Aufwand, um auf das Terminal-Patch-Set Oracle Database 11.2.0.4 umzusteigen äquivalent zum Aufwand für ein Upgrade auf Oracle Database 12c ist – nur mit dem entscheidenden Unterschied, dass Oracle 12.1 noch bis Mitte 2018 im Premier Support ist, der sich maximal um drei Jahre per Extending Support verlängern lässt.



Mike Dietrich  
mike.dietrich@oracle.com

# Änderungen in Journal-Tabellen hervorheben

Gerd Volberg, OPITZ CONSULTING GmbH

Journal-Tabellen, in denen die Historie einer Tabelle gespeichert wird, sind sehr unübersichtlich und daher schwer zu analysieren.

Abbildung 1 zeigt die historischen Stammdaten-Änderungen des Mitarbeiters Smith in der Tabelle „EMP“. In dieser Darstellungsweise sind Änderungen nur sehr schwer zu erkennen.

Viel einfacher wird es, wenn alle Änderungen farblich hervorgehoben werden. In Abbildung 2 sind alle Felder, die sich gegenüber dem älteren Datensatz geändert haben, mit grünem Hintergrund unterlegt.

Die einfachste Technik, mit der man dieses Feature implementieren kann, beruht auf einem einzigen „POST QUERY“-Trigger, der auf dem Block „EMP“ erzeugt wird. Für jedes Feld, das man hervorheben möchte, benötigt man in dieser Lösung einen Parameter, in dem der vorherige Wert zwischengespeichert

Empno	Änderung	Ename	Job	Salary	Mgr	Deptno
7369	17.12.1980	SMITH	CLERK	800	7902	20
7369	01.01.2000	SMITH	CLERK	820	7902	20
7369	01.07.2004	SMITH	CLERK	850	7902	20
7369	01.01.2008	SMITH	SALESMAN	1400	7902	20
7369	01.03.2010	SMITH	SALESMAN	1400	7839	10
7369	01.05.2011	SMITH-KING	SALESMAN	1550	7839	10
7369	01.08.2011	SMITH-KING	SALESMAN	1700	7839	10
7369	01.01.2012	SMITH-KING	ANALYST	2800	7839	10

Abbildung 1: Rohdaten einer Journal-Tabelle

```
IF :EMP.ENAME != :Parameter.PA_ENAME THEN
  Display_Item ('EMP.ENAME', 'VA_CHANGED');
END IF;
:Parameter.PA_ENAME := :EMP.ENAME;

IF :EMP.JOB != :Parameter.PA_JOB THEN
  Display_Item ('EMP.JOB', 'VA_CHANGED');
END IF;
:Parameter.PA_JOB := :EMP.JOB;

IF :EMP.MGR != :Parameter.PA_MGR THEN
  Display_Item ('EMP.MGR', 'VA_CHANGED');
END IF;
:Parameter.PA_MGR := :EMP.MGR;

IF :EMP.SAL != :Parameter.PA_SAL THEN
  Display_Item ('EMP.SAL', 'VA_CHANGED');
END IF;
:Parameter.PA_SAL := :EMP.SAL;

IF :EMP.DEPTNO != :Parameter.PA_DEPTNO THEN
  Display_Item ('EMP.DEPTNO', 'VA_CHANGED');
END IF;
:Parameter.PA_DEPTNO := :EMP.DEPTNO;
```

Listing

Empno	Änderung	Ename	Job	Salary	Mgr	Deptno
7369	17.12.1980	SMITH	CLERK	800	7902	20
7369	01.01.2000	SMITH	CLERK	820	7902	20
7369	01.07.2004	SMITH	CLERK	850	7902	20
7369	01.01.2008	SMITH	SALESMAN	1400	7902	20
7369	01.03.2010	SMITH	SALESMAN	1400	7839	10
7369	01.05.2011	SMITH-KING	SALESMAN	1550	7839	10
7369	01.08.2011	SMITH-KING	SALESMAN	1700	7839	10
7369	01.01.2012	SMITH-KING	ANALYST	2800	7839	10

Abbildung 2: Journal-Tabelle mit Änderungs-Hervorhebungen



wird. Die Hervorhebung ist in unserem Beispiel über ein „Visual“-Attribut gesteuert, das die Hintergrundfarbe auf „grün“ setzt.

„POST QUERY“ vergleicht alte und neue Werte miteinander. Sind sie nicht identisch, wird der Hintergrund des Felds mit

dem Visual-Attribut „VA\_CHANGED“ überschrieben (siehe Listing).

Es empfiehlt sich, diese Funktionalität in eine generische Prozedur zu überführen und in einer PL/SQL-Library global zur Verfügung zu stellen, falls man diese Technik mehrfach einsetzen möchte.



Gerd Volberg  
gerd.volberg@opitz-consulting.com  
talk2gerd.blogspot.com

## DOAG 2014 Konferenz + Ausstellung mit neuem Besucherrekord

Mylène Diacquenod und Marina Fischer, DOAG Online



Die rund 2.000 Fachbesucher bekamen auf der 27. DOAG Konferenz + Ausstellung vom 18. bis 20. November in Nürnberg wieder eine ganze Menge geboten: etwa 450 Vorträge in neun unterschiedlichen Streams, sechs Keynotes und eine erstmals ausgebuchte Ausstellung. Kurzum, alles Wichtige rund um die neusten Trends und Entwicklungen der Branche. Neben dem ausführlichen Konferenzprogramm boten die begleitende Aus-

stellung und zahlreiche Community- und Networking-Aktivitäten viele Möglichkeiten zum Austausch.

Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG, eröffnete die Veranstaltung mit einer Begrüßung, die das Motto der diesjährigen Konferenz in den Mittelpunkt rückte: „Experience Passion“. Der Spaß an Technolo-

gien und die Leidenschaft für ihren Beruf sei etwas, das alle Konferenzteilnehmer verbinde – ganz egal, ob Administrator, Entwickler, Architekt oder Manager, betonte Neugebauer. „Diese Leidenschaft ist unser großer, gemeinsamer Nenner und die Stärke der DOAG“, so der Datenbank-Experte, „und diese Leiden-

schaft wird noch stärker, wenn man sie gemeinsam erlebt – wie bei der DOAG Konferenz und Ausstellung.“

Besonders erfreulich ist in diesem Jahr auch die erstmals vollständig ausgebuchte Ausstellungsfläche. Sie wurde sogar über die bisherigen drei Ebenen bis hin zur Ebene null ausgeweitet, um dem Ansturm der Ausstellernanfragen gerecht zu werden. Auch die stetig steigenden Besucherzahlen zeigen, dass die Jahreskonferenz ein fest etabliertes Ereignis und eine beliebte Plattform zum Erfahrungsaustausch darstellt.





Die Eröffnungs-Keynote hielt im Anschluss an Neugebauers Begrüßungsrede



Markus Flierl, Oracle VP Software Development. Dabei ging der Solaris-Verantwortliche insbesondere auf das Zusammenspiel zwischen Oracle Database und der Oracle-Infrastruktur ein. Die Fachbesucher erfuhren so aus erster Hand, wie zum Beispiel die neue Version von Solaris oder SPARC unter anderem höhere Leistung, Skalierbarkeit und Verfügbarkeit der Datenbank oder Oracle-Applikation erzielen. Auch bei Flierl war die Passion für seine Arbeit während seines Vortrags stets spürbar, getreu dem Motto: Experience Passion.



Mylène Diacquenod  
mylene.diacquenod@doag.org



Marina Fischer  
marina.fischer@doag.org

## Wir begrüßen unsere neuen Mitglieder

### Persönliche Mitglieder

Michael Kunze  
Dr. Daria Schymura  
Markus Langlotz  
Rainer Glanzner  
Christian Rügger  
Frank Binkowski  
Ulrich Weber  
Michael Griesser  
Volker Ambach  
Hajo Rühl  
Michael Andresen  
Josef Opitz  
Michel Keemers  
Thomas Geib

Sebastian Preuß  
Christian Henn  
Andreas Kosubek  
Constantin Lotz  
Holger Tenbeitel  
Thomas Ostermeier  
Thorsten Gehrke  
Andreas Brück  
Daniel Baier  
Marcel Stratmann  
Steven Grzbielok  
Moritz Lilleby  
Timo Amling  
Martin Ortel

Philipp Albrecht  
Matthias Schnell  
Simon Buchholz  
Sumit Goyal  
Florian Wieser  
Maximilian Liesegang  
Anton Frank  
Alexandra Bartels  
Arun Kumar Kandhasamy  
Jan Domanski  
Mustafa Bicen  
Tarik Artun  
Burcu Öztürk

### Firmenmitglieder

Bert Horvath, EQUIcon Software GmbH  
Oliver Schran, MediaNet GmbH  
Ralf Böttger, Wincor Nixdorf GmbH  
Matthias Bartels, Haema AG  
Martin Klier, Performing Databases GmbH  
Carsten Schröder, BERIS consulting GmbH  
Svenja Gisbertz, Raynet GmbH  
Benjamin Büchele, ITUnified  
Christoph Oldenburg, Dirk Rossmann GmbH  
Hendrik Schäfer, BISON Schweiz AG  
Lothar Flatz, Diso AG  
Peter Küppers, Axedo Solutions AG



12. - 13.02.2015

### DOAG PL/SQL Community Day

Christian Schwitalla  
sig-development@doag.org

13.02.2015

### DOAG Webinar: Identification of Performance Problems without the Diagnostic Pack

Johannes Ahrends, Christian Trieb  
sig-database@doag.org

19.02.2015

### Regionaltreffen NRW (Forms Community)

Stephan Kinnen; Andreas Stephan  
regio-nrw@doag.org

23.02.2015

### Regionaltreffen München/Südbayern

Franz Hüll, Andreas Ströbel  
regio-muenchen@doag.org



03. - 04.03.2015

### DOAG Berliner Expertenseminar mit Martin Bach zum Thema „Exadata“

Cornel Albert  
expertenseminare@doag.org

04.03.2015

### Regionaltreffen Berlin/Brandenburg

Michel Keemers  
regio-bb@doag.org

05.03.2015

### Regionaltreffen Würzburg

Oliver Pyka  
regio-wuerzburg@doag.org

09.03.2015

### Regionaltreffen Osnabrück/Bielefeld/Münster

Andreas Kother, Klaus Günther  
regio-osnabrueck@doag.org

13.03.2015

### DOAG Webinar: Datenbankadministration Q&A

Johannes Ahrends, Christian Trieb  
sig-database@doag.org

19.03.2015

### Regionaltreffen Stuttgart

Jens-Uwe Petersen  
regio-stuttgart@doag.org

19.03.2015

### SOUG SIG

24. - 25.03.2015

### JavaLand 2015

24.03.2015

### Regionaltreffen München/Südbayern

Franz Hüll, Andreas Ströbel  
regio-muenchen@doag.org

25.03.2015

### Real World Performance Tour 2015

Johannes Ahrends, Christian Trieb  
sig-database@doag.org



10.04.2015

### DOAG Webinar: SAP auf Oracle: Development Update

Johannes Ahrends, Christian Trieb  
sig-database@doag.org

13.04.2015

### Regionaltreffen Jena/Thüringen

Jörg Hildebrandt  
regio-thueringen@doag.org

13. - 14.04.2015

### DOAG Berliner Expertenseminar mit Randolf Geist zum Thema „Parallel Execution Masterclass“

Cornel Albert  
expertenseminare@doag.org

Weitere Termine und Informationen unter [www.doag.org/termine/calendar.php](http://www.doag.org/termine/calendar.php)

## Impressum

### Herausgeber:

DOAG Deutsche ORACLE-Anwendergruppe e.V.  
Tempelhofer Weg 64, 12347 Berlin  
Tel.: 0700 11 36 24 38  
[www.doag.org](http://www.doag.org)

### Verlag:

DOAG Dienstleistungen GmbH  
Fried Saacke, Geschäftsführer  
[info@doag-dienstleistungen.de](mailto:info@doag-dienstleistungen.de)

### Chefredakteur (ViSdP):

Wolfgang Taschner, [redaktion@doag.org](mailto:redaktion@doag.org)

### Redaktion:

Fried Saacke, Julia Bartzik,  
Mylène Diacquenod, Marina Fischer,  
Marius Fiedler, Dr. Dietmar Neugebauer

### Titel, Gestaltung und Satz:

Alexander Kermas, DOAG Dienstleistungen GmbH

**Titelfoto:** © fotomek / Fotolia.com

**Foto S. 47:** © Andrei Merkulov / 123RF.com

**Foto S. 71:** © antonprado / 123RF.com

### Anzeigen:

Simone Fischer, [anzeigen@doag.org](mailto:anzeigen@doag.org)  
DOAG Dienstleistungen GmbH  
Mediadaten und Preise finden Sie  
unter: [www.doag.org/go/mediadaten](http://www.doag.org/go/mediadaten)

### Druck:

Druckerei Rindt GmbH & Co. KG  
[www.rindt-druck.de](http://www.rindt-druck.de)

## Inserentenverzeichnis

avato consulting ag <a href="http://www.avato-consulting.com">www.avato-consulting.com</a>	S. 25
DBConcepts <a href="http://www.dbconcepts.at">www.dbconcepts.at</a>	S. 17
dbi services ag <a href="http://www.dbi-services.com">www.dbi-services.com</a>	S. 33
Digicomp Academy AG <a href="http://www.digicomp.ch">www.digicomp.ch</a>	S. 3
DOAG e.V. <a href="http://www.doag.org">www.doag.org</a>	U 3, S. 21
IT GAIN GmbH <a href="http://www.itgain.de">www.itgain.de</a>	S. 43
Libelle AG <a href="http://www.libelle.com">www.libelle.com</a>	S. 57
MuniQsoft GmbH <a href="http://www.muniqsoft.de">www.muniqsoft.de</a>	S. 13
ORACLE Deutschland B.V. & Co. KG <a href="http://www.oracle.com">www.oracle.com</a>	U 2
PROMATIS software GmbH <a href="http://www.promatis.de">www.promatis.de</a>	S. 63
Trivadis GmbH <a href="http://www.trivadis.com">www.trivadis.com</a>	U 4





# DOAG 2015 Business Intelligence

23. April 2015, in München

*Neueste Trends und Entwicklungen im BI- und DWH-Umfeld*

Big Data und In-Memory – zwei Trends, die sich noch in der Experimentierphase befinden. Die Schonfrist ist jetzt um: Wir holen diese Themen aus den Laboren und nehmen sie auf der DOAG BI Konferenz auseinander. Die In-Memory-Option der Oracle-Datenbank und die Technologien um Hadoop und NoSQL sind selbstverständlich nicht die einzigen Schwerpunkte der kommenden Auflage. Auch die Themen Performance und Datenbankmodellierung werden vertieft.

Für Quereinsteiger wird ein Spezial-Programm geboten, das die Grundlagen erläutert.



# Wir bringen Ihre IT-Infrastruktur auf Kurs.



Trivadis ist führend bei der IT-Beratung, der Systemintegration, dem Solution-Engineering und bei den IT-Services mit Fokussierung auf Oracle- und Microsoft-Technologien im D-A-CH-Raum. Unsere Leistungen erbringen wir auf den strategischen Geschäftsfeldern Application Development, Business Intelligence, Infrastructure Engineering und Training. Darüber hinaus übernimmt die Trivadis Services den korrespondierenden Betrieb Ihrer IT-Systeme. Sprechen Sie mit uns über Ihre Herausforderungen. Wir schaffen die Lösung. [www.trivadis.com](http://www.trivadis.com) | [info@trivadis.com](mailto:info@trivadis.com)

ZÜRICH ▪ BASEL ▪ BERN ▪ BRUGG ▪ GENÈVE ▪ LAUSANNE ▪ DÜSSELDORF  
FRANKFURT A.M. ▪ FREIBURG I.B.R. ▪ HAMBURG ▪ MÜNCHEN ▪ STUTTGART ▪ WIEN

**trivadis**  
makes IT easier. ■ ■ ■