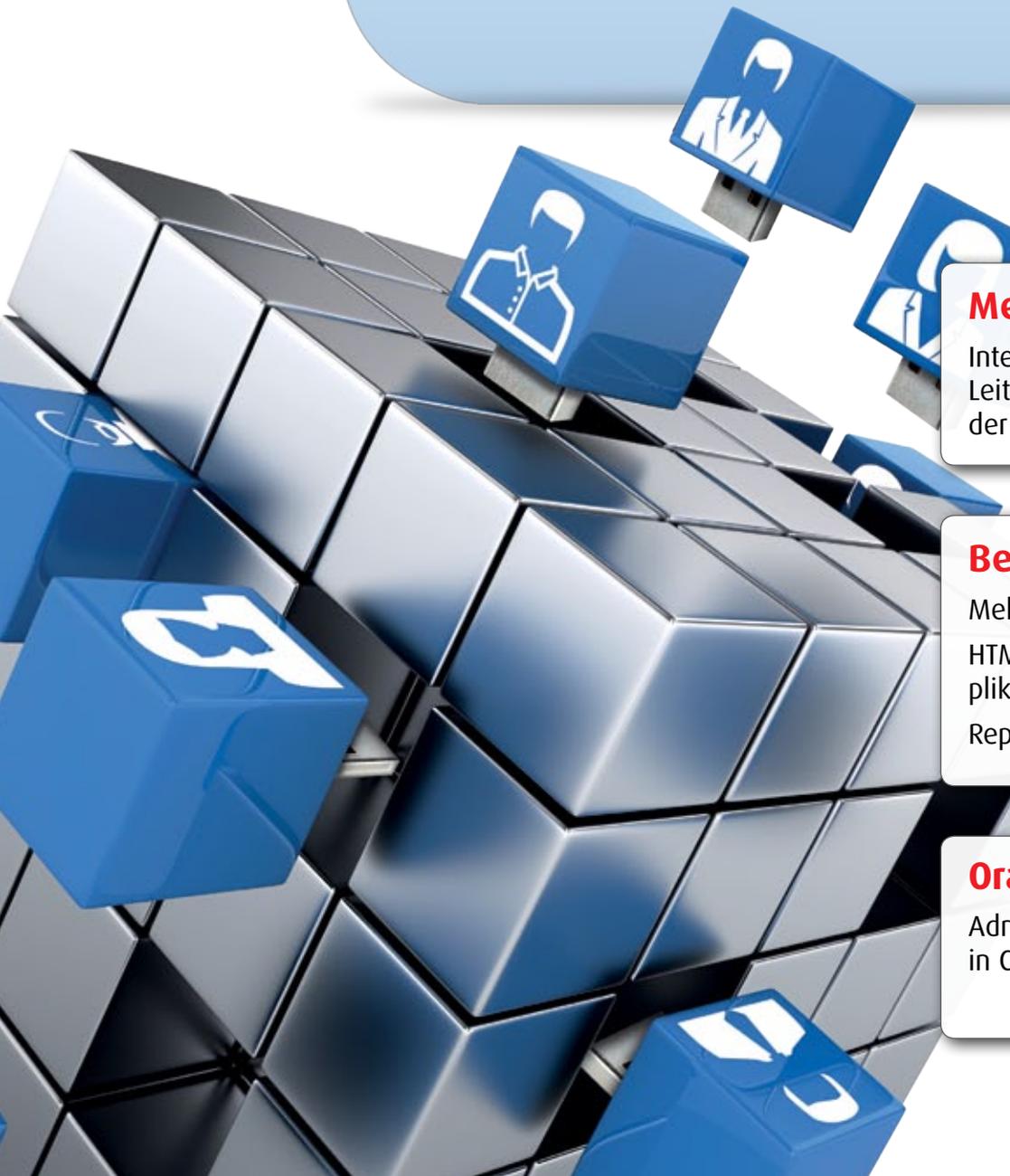


DOAG

Deutsche ORACLE-Anwendergruppe e.V.

News

Plug into



Meinung

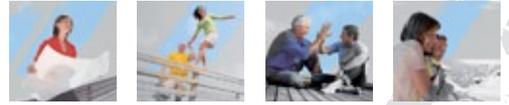
Interview mit Karsten Heinrichs, Leiter Team Database Systems der regio IT

Best Practice

Mehrsprachigkeit in der BIEE 11g
HTML-Anker für eine Apex-Applikation
Replikation großer Datenmengen

Oracle Hidden Secrets

Administrations-Skripte in Cloud Control nutzen



Treffen Sie uns auf der DOAG 2013
Sie finden uns an Stand 238

Wir haben vielleicht keine Superkräfte, aber dafür bieten wir Ihnen IT-Betrieb in Bestform!



Entlasten Sie wertvolle Ressourcen und stellen Sie dennoch die Verfügbarkeit aller Systeme sicher:

Unser OC|MSI® Team unterstützt Sie beim alltäglichen Betrieb Ihrer IT-Landschaft und sorgt für maßgeschneiderte Systemverfügbarkeit, Administration Ihrer IT-Systeme, Pflege, Überwachung, Optimierung – und Sie haben Zeit für Ihre innovativen IT-Projekte!

Managed Services Infrastructure bietet Ihnen herstellerunabhängige Beratung und individuelle auf Sie angepasste Lösungen, remote oder vor Ort, 24/7 oder als Urlaubsvertretung.

Auf Wunsch betreuen wir Ihre IT-Infrastruktur in einer sicheren Service Cloud und sichern so Ihre Business Continuity.



Erfahren Sie mehr über unsere Leistungen im Bereich Managed Services Infrastructure unter www.opitz-consulting.com/msi

Ihr direkter Ansprechpartner ist Najibullah Rajab.
Telefon: +49 2261 6001-0 · E-Mail: Najibullah.Rajab@opitz-consulting.com

Treffen Sie uns auf der
DOAG 2013

19. - 21.11.2013 Nürnberg

Besuchen Sie die Vorträge unserer Experten und unseren Stand. Sie finden uns an Stand 238.

Wir erwarten Sie vor Ort mit Themen wie Agile BI, BPM & SOA, Oracle Forms uvm. Reinschauen lohnt sich!



Wir freuen uns auf Sie!



Christian Trieb
Leiter der Datenbank
Community

Liebe Mitglieder der DOAG Deutsche ORACLE-Anwendergruppe,
liebe Leserinnen und Leser,

sie ist endlich da! Die neue Version 12c der Oracle-Datenbank. Es hat lange gedauert; die ersten Informationen gab es für Vertreter der Oracle-Anwendergruppen und für Beta-Tester vor ziemlich genau zwei Jahren während der Oracle OpenWorld 2011. Die Einladung kam damals sehr kurzfristig und alle Teilnehmer mussten eine Vertraulichkeitserklärung unterschreiben, es durfte nicht fotografiert werden und auch die Kommunikation über das Präsentierte war gegenüber Dritten nicht erlaubt.

Das Hauptaugenmerk lag damals wie heute auf der Pluggable Database, die eine der umfassendsten Änderungen der Oracle-Datenbank in den letzten Jahren darstellt. So war es auch dieses Feature, das nach der Verfügbarkeit der ersten Beta-Version im Februar 2012 intensiv getestet wurde.

Aber die rund fünfhundert weiteren neuen Eigenschaften sollen darüber nicht vergessen werden. So ist beispielsweise „recover table“ eine Möglichkeit, die lange vermisst wurde. In diesem Heft werden viele dieser neuen Möglichkeiten beschrieben und erläutert.

Erfahrungsberichte und Beschreibungen der neuen Features wird es auch während der DOAG 2013 Konferenz + Ausstellung vom 19. bis zum 21. November 2013 in Nürnberg geben. Andrew Mendelsohn, Senior Vice President Oracle Database Server Technologies und Entwicklungs-Chef der Datenbank, wird das Konzept in seiner Keynote vorstellen und es werden zudem einige Oracle-Produktmanager aus der Konzernzentrale anwesend sein.

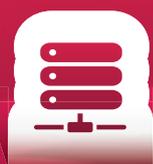
Spezialisten, die sich im Netzwerk OAK-Table zusammengefunden haben, gestalten einen ganzen Stream über alle drei Tage. Speziell für Einsteiger werden am 21. November 2013 durchgängig Vorträge angeboten.

Ich wünsche Ihnen viel Spaß beim Lesen dieser Ausgabe und freue mich, Sie in Nürnberg persönlich treffen zu können.

Ihr

ORACLE Platinum
Partner

HUNKLER
GmbH & Co. KG



„ Sicher, schnell, kompakt verpackt:
Das Datenbanksystem
für Ihre Höchstleistung “

Aktuelle Infos und Aktionen
zu Oracle unter
www.oracle-on-oracle.de

LIZENZBERATUNG &
-VERTRIEB



HOCHVERFÜGBAR-
KEITSLÖSUNGEN &
PERFORMANCE
TUNING



DATA WAREHOUSING &
BUSINESS
INTELLIGENCE
LÖSUNGEN



ORACLE
APPLIANCES



Oracle Database Appliance: perfekt abgestimmt – bis zum letzten Treiber

Server, Storage, Software, Netzwerk: Das ist die Oracle Database Appliance (ODA). Das Datenbank-Komplettsystem ist vorkonfiguriert und bis ins Detail perfekt abgestimmt. Keine aufwändige Verkabelung, keine fehleranfällige Anpassung. Dafür maximale Zuverlässigkeit und Ausfallsicherheit – sofort betriebsbereit!

ODA heißt: Pay as you grow. Sie schalten erstmal nur die Leistung frei, die Sie brauchen, und kaufen die entsprechenden Lizenzen. Dann wächst die ODA mit Ihren Anforderungen mit: Sie erwerben einfach weitere Lizenzen und aktivieren die zusätzliche Power per Knopfdruck – installieren müssen Sie dafür nichts!

Oracle Database Appliance: Das clevere System aus einer Hand. Fragen Sie uns!

Hauptsitz Karlsruhe
Bannwaldallee 32, 76185 Karlsruhe
Tel. 0721-490 16-0, Fax 0721-490 16-29

Geschäftsstelle Bodensee
Fritz-Reichle-Ring 6a, 78315 Radolfzell
Tel. 07732-939 14-00, Fax 07732-939 14-04

info@hunkler.de, www.hunkler.de

ODA AUF EINEN BLICK

- Komplettlösung: Datenbanksoftware, Server, Storage, Netzwerk in einer Box
- Vollständig vorkonfiguriert und perfekt abgestimmt
- Auf Knopfdruck betriebsbereit
- Wächst mit: für mehr Leistung einfach weitere Lizenzen erwerben und aktivieren



Interview mit Karsten Heinrichs, Leiter Team Database Systems der regio iT, Seite 7



Neue Funktionen in der Datenbank 12c, Seite 18



„IDENTITY table clause“- Feature und Apex, Seite 36

Inhalt

- 3** Editorial
Christian Trieb
- 5** Spotlight
- 6** Die DOAG in der Presse
- 7** „Mittlerweile fühle ich mich im Oracle-Bereich sehr wohl ...“
Interview mit Karsten Heinrichs, Leiter Team Database Systems der regio iT

Oracle Database 12c

- 10** Oracle Multitenant: Einfachere Möglichkeiten für die Datenbank-Konsolidierung und mehr
Manuel Hoßfeld
- 13** Automatisierung des Information Lifecycle Managements
Ulrike Schwinn
- 18** Neue Funktionen für Entwickler
Carsten Czarski
- 22** Hochverfügbarkeit
Sebastian Solbach

- 26** Security Inside Out
Heinz-Wilhelm Fabry
- 30** Management der Oracle Datenbank 12c
Ralf Durben
- 34** Die Datenbank 12c auf Solaris 11.1
Franz Haberhauer
- 36** „IDENTITY table clause“-Feature und Apex
Alain Lacour und Arnaud Barbier
- 40** Einsatz von Partial Indices im Data Warehouse
Reinhard Mense

Best Practice

- 42** Replikation großer Datenmengen in einem OLTP-System
Uwe Simon
- 49** Mehrsprachigkeit in der Oracle Business Intelligence Enterprise Edition 11g
Dr. Götz Gleitsmann
- 52** Einen HTML-Anker in eine Apex-Applikation einbauen
Philip Ton
- 55** Oracle SQL – Das umfassende Handbuch
gelesen von Bernd Tuba, regionaler DOAG-Repräsentant Trier/Saarland/Luxemburg

Tipps & Tricks

- 57** Hidden Secrets: Administrations-Skripte in Cloud Control nutzen
Ralf Durben
- 60** Gerds Fundgrube Heute: Visuelles Attribut beliebiger Records ändern
Gerd Volberg

DOAG intern

- 61** „Jede Frau kann in der IT mit ihren speziellen Stärken ihren Platz finden ...“
Interview mit Sabine Heimsath
- 63** Die Oracle-Community trifft sich auf der DOAG 2013 Konferenz + Ausstellung
- 64** Aus dem Verein
- 64** Neue Mitglieder
- 65** Impressum
- 66** Termine
- 66** Inserentenverzeichnis



Spotlight

Mittwoch, 18. Juli 2013

In Frankfurt findet das Kick-off-Meeting für die neue DOAG-Konferenz „JavaLand“ statt. Unter dem Motto „Zwei Tage das JavaLand besiedeln“ bietet die Veranstaltung ein innovatives Konzept für die Teilnehmer. JavaLand wird von der DOAG sowie vom Heise Zeitschriften Verlag vom 25. bis 26. März 2014 im Phantasialand in Brühl präsentiert. Der Java-Community bietet sich dort ein attraktiver Rahmen zum Lernen, Austauschen und Erfahren von Neuigkeiten. Dafür haben die Organisatoren die im Interessenverbund der Java User Groups e.V. (iJUG) vertretenen Java User Groups eingeladen, die Konferenz mitzugestalten.

Dienstag, 23. Juli 2013

Die Regionalgruppe München beschäftigt sich bereits mit dem neuen Datenbank-Release 12c. Aufgrund der vielen Anmeldungen muss der regionale Repräsentant Andreas Ströbel einen größeren Vortragsraum reservieren. Wolfgang Thiem von Oracle Deutschland zeigt die interessantesten Neuigkeiten der Datenbank. Die Teilnehmer stellen viele Fragen und diskutieren vor allem die Vorteile der Pluggable Database.

Donnerstag, 8. August 2013

Peter Doschkinow, Senior Java Architekt, und Michael Bräuer, leitender Systemberater der Oracle Business Unit Server Technologies, stellen in der DOAG-Konferenz-Lounge das zukunftsweisende Thema „HTML5 und Bleeding-Edge Java Enterprise Technologien“ aus Oracle-Sicht dar. Die Zuhörer nehmen interessante Aspekte und Anregungen für ihre Entwicklungsprojekte mit.

Freitag, 9. August 2013

Bei einem Workshop im Berliner DOAG-Büro machen sich die Beteiligten aus der Online- und Print-Redaktion Gedanken, um den Content-Lifecycle zu intensivieren und die vorhandenen interessanten Fachbeiträge besser zu verbreiten.

Dienstag, 13. August 2013

Die Launch-Termine für die neue Datenbank 12c gehen in Hamburg zu Ende. Michael Paege, der die DOAG vor Ort vertritt, kann erstmals den Flyer für die DOAG 2013 Konferenz + Ausstellung verteilen. Die Exemplare werden ihm buchstäblich aus den Händen gerissen.

Montag, 20. August 2013

Die neue Webseite der DOAG-Konferenz „JavaLand“ geht unter „www.javaland.eu“ online. Jetzt sind die Referenten aufgefordert, möglichst viele und gute Vorträge einzureichen.

Mittwoch, 4. September 2013

Herber Rückschlag für Oracle-Chef Larry Ellison bei der Verteidigung des America's-Cup. Das Team Oracle wird wegen unerlaubter Bootsmanipulation hart bestraft. Larry Ellison will jedoch weiterhin alles daransetzen, die begehrte Segel-Trophäe nach Amerika zu holen, um seinem Land einen Dienst zu erweisen.

Freitag, 8. September 2013

Hauptthema der DOAG-Vorstandssitzung sind die von der Delegiertenversammlung verabschiedeten Ziele. Die Vorstände erarbeiten daraus einen Fahrplan für die Agenda 2017.

Donnerstag, 19. September 2013

Die DOAG-Delegation macht sich auf den Weg zur OpenWorld, der Hauskonferenz von Oracle in San Francisco. Alles in allem werden dort sowie auf den parallel dazu stattfindenden Veranstaltungen JavaOne und MySQL Connect, außerdem beim Partner-Event Oracle PartnerNetwork Exchange knapp 3.300 Referenten mehr als 2.300 Sessions abhalten. Der DOAG-Vorstandsvorsitzende Dr. Dietmar Neugebauer, Geschäftsführer Fried Saacke und DOAG-News-Chefredakteur Wolfgang Taschner sorgen in der kalifornischen Metropole dafür, dass die DOAG-Mitglieder über DOAG Online stets aktuell zu allen relevanten Aspekten des Events informiert sind.



Die DOAG in der Presse

Die nachfolgenden Ausschnitte reflektieren die Einschätzung der Fach- und Wirtschaftspresse zu bestimmten Themen über Oracle; die Veröffentlichungen geben nicht die Meinung der DOAG wieder und sind auch nicht im Vorfeld mit der DOAG abgestimmt. Lediglich die Zitate einzelner DOAG-Vorstände geben die Meinung der DOAG wieder.

Heise online vom 16. Juni 2013

Umfrage: Interesse für Oracles „Engineered Systems“

In einer gemeinsamen Umfrage mit der Experton-Group hat die Deutsche Oracle Anwendergruppe (DOAG) Interesse und Einschätzung der von Oracle heftig beworbenen Appliances untersucht. Rund 43 Prozent der als „Endanwender“ eingestuften Antwortenden halten die mit Software gebündelten Maschinen für „uninteressant“. 34 Prozent hingegen setzen sie bereits ein oder prüfen eine Investition in die Technik.

Eins der wichtigsten Ziele der Befragten ist die Senkung der Lizenzkosten. Über deren Höhe klagen Oracle-Kunden immer wieder, vor allem im Zusammenhang mit der Verwendung virtueller Maschinen. Andererseits erwarten die Teilnehmer an der Umfrage durch den Einsatz der diversen Exa-Angebote keinen Rückgang der Lizenzausgaben. Sie hoffen vor allem auf eine Vereinfachung des IT-Systems und schnellere Reaktionszeiten des Support.

Als größte mögliche Nachteile von Oracle-Appliances gilt die Abhängigkeit von einem Hersteller, und zwar sowohl bei zukünftigen Investitionsentscheidungen als auch bei Wartung und Überwachung der Installationen. Den höchsten Anteil unter den bereits installierten Appliances haben die Datenbank-Server der Exadata-Reihe. Aller-

dings läuft sie lediglich bei 9 Prozent der befragten Unternehmen, und ihre Anschaffung kommt für 44 Prozent nicht in Frage. Bei allen anderen Paket-Angeboten von Oracle lag dieser Anteil deutlich unter 40 Prozent.

Befragt wurden 500 Unternehmen und Personen. 60 Prozent der Firmen hatten 1000 und mehr Mitarbeiter, 30 Prozent weniger als 500. Fast alle Befragten setzten Oracles Datenbank ein (94 Prozent), die Middleware verwendet rund die Hälfte. Hardware und Applications nutzen nur jeweils ein Viertel der Teilnehmer.

Computer Reseller News vom 18. Juli 2013

Oracle will Rückstand aufholen

Mit der neuen Version 12c seines Datenbanksystems geht der Softwarehersteller Oracle in die Offensive. Kosteneinsparungen bei den Anwendern, neue Möglichkeiten für Partner und überraschende Cloud-Allianzen sollen für rasche Verbreitung sorgen.

„Daten und Datenbanken sind wieder in“, sagt Günther Stürner, Vice President Server Technologies und Sales Consulting bei Oracle Deutschland. Für sein Unternehmen eigentlich eine gute Botschaft: Die Marktforscher von Gartner attestieren für das Jahr 2012 Oracle 48,3 Prozent und die von IDC 46,6 Prozent Anteil an dem rund 25 Milliarden Dollar schweren Datenbank-Weltmarkt. Allerdings dreht sich das neue Interesse um die Themen In-Memory und Big Data. Die Szenerie prägt bei der Haltung von Daten im Arbeitsspeicher SAP, und bei der Speicherung großer Datenmengen stehen die quelloffene Technologie Hadoop und andere NoSQL-Systeme im Rampenlicht. Der Wettbewerber IBM scheint den neuen Optionen gegenüber offen

und baut seine Datenmanagement-Palette aus. Oracle ist in seinem Kerngeschäft unter Druck geraten. Innovativ sind andere, der Weltmarktführer wirkt konservativ und defensiv.

... Stürner erwartet, dass der Umstieg auf das neue Release rascher erfolgen wird als bei der vorangegangenen Version. Die Mandantenfähigkeit (multitenancy, pluggable database) werde von vielen Kunden gewünscht. Dietmar Neugebauer, Vorsitzender der Deutschen Oracle-Anwendergruppe (DOAG), bestätigt die Vereinfachungen, die diese Architektur bei der Administration erlaubt. Man müsse jedoch prüfen, inwieweit sich die Kosteneinsparungen realisieren lassen, weil sich Oracle neue Features oft separat bezahlen lasse ...

Infomarkt vom 20. Juli 2013

Oracle - Kundenvotum

Das Interesse von IT-Anwendern an integrierten Server-Storage-Netzwerk-Systemen („Appliances“) ist groß und Oracle gehört zu den Trendsettern in diesem Bereich. Die Deutsche Oracle Anwendergruppe (DOAG) hat daher gemeinsam mit der Experten Group eine Online-Befragung bei mehr als 500 Anwendern zu diesem Thema durchgeführt. Das Ergebnis: 14 Prozent der befragten Unternehmen haben die Appliances bereits in die Investitionsplanung aufgenommen, weitere 20 Prozent prüfen diesen Schritt. 43 Prozent der Unternehmen dagegen meldeten kein Interesse. Als Nachteil der Appliances wird insbesondere die Abhängigkeit von einem einzigen Hersteller bei der Technologie sowie beim Monitoring und Support gesehen.

Weitere Pressestimmen lesen Sie unter <http://www.doag.org/presse/spiegel>



Stefan Kinnen (links) im Gespräch mit Karsten Heinrichs

„Mittlerweile fühle ich mich im Oracle-Bereich sehr wohl ...“

Der Umgang mit weit mehr als 1.000 Datenbanken ist keine leichte Aufgabe. Stefan Kinnen, Leiter der Development Community, und Wolfgang Taschner, Chefredakteur der DOAG News, sprachen darüber mit Karsten Heinrichs, Leiter Team Database Systems der regio iT gesellschaft für informationstechnologie mbh.

Was sind die größten IT-Anforderungen und Schwerpunkte in Ihrem Unternehmen?

Heinrichs: Die regio iT ist seit zehn Jahren regionaler IT-Dienstleister für Kommunen, kommunale Unternehmen, Schulen, Energieversorger, Entsorger sowie Non-Profit-Organisationen. Wir haben große Erfahrung in den Prozessen unserer Kunden und können sie dadurch mit maßgeschneiderten IT-Lösungen optimal unterstützen. Beispielsweise mit Bürgerportal-Lösungen, mit denen Kommunalverwaltungen ihre Dienstleistungen näher an den Bürger bringen, indem diese unter anderem ihren Bewohner-Parkausweis zu Hause schnell und unkompliziert beantragen, ausdrucken und jetzt auch via PayPal bezahlen können. Oder mit unserem Produkt „VoteManager“, einer Eigenentwicklung zur Organisation, Vorbereitung und Durchführung von Wahlen. Er ist nicht nur Marktführer in Nordrhein-Westfalen, sondern auch bundesweit bereits bei 276 Kommunen im Einsatz. Darüber hinaus bieten wir IT-Lösungen

für unsere Partner in der Energiebranche, auch hier über NRW hinaus.

Gibt es bei so vielen Lösungen noch den sogenannten „roten Faden“?

Heinrichs: Der ist für mich unsere Cross-over-Kompetenz: Wir kennen sowohl die Prozesse in den Öffentlichen Verwaltungen als auch die Wertschöpfungskette der Energieversorger – gerade auch aus unserer Historie heraus. Hierbei konzentrieren wir uns auf die vier Bereiche „Energie und Entsorgung“, „Bildung



Karsten Heinrichs

und Wissen“, „Verwaltung und Finanzen“ sowie „IT-Service und Betrieb“.

Welchen Umfang haben Ihre IT-Systeme?

Heinrichs: Wir betreiben an drei Standorten in Aachen und Gütersloh derzeit rund 1.500 Server-Systeme, davon sind 65 Prozent virtualisiert. Um unserem Wachstum gerecht zu werden, haben wir erst vor zwei Jahren ein neues Rechenzentrum am Standort Aachen in Betrieb genommen, wo wir modernste Data-Center-Technik anbieten, eingebunden in ein energieeffizientes Konzept. Aktuell setzen wir mehr als 1.300 Datenbanken ein – davon entfallen 69 Prozent auf Microsoft SQL Server, 16 Prozent auf MySQL, 8 Prozent auf Oracle, 4 Prozent auf DB/2 und 3 Prozent auf PostgreSQL. Hinzu kommen noch unsere SAP-Systeme, die alle mit MaxDB laufen.

Wie sehen Sie die Oracle-Datenbank im Vergleich zu den anderen Datenbanken?

Heinrichs: Ich habe im Jahr 1996 als Ad-

administrator für die DB/2-Datenbank begonnen und mich im Jahr 2002 zum ersten Mal mit der Oracle-Datenbank beschäftigt. Damals war die Oracle-Lösung für mich sehr komplex und schwierig zu administrieren. Mittlerweile fühle ich mich aber im Oracle-Bereich sehr wohl. Eine Abgrenzung zwischen den einzelnen Datenbank-Anbietern fällt zunehmend schwerer, da sich die Systeme immer ähnlicher werden.

Nach welchen Kriterien entscheiden Sie dann bei der Einführung einer neuen Datenbank?

Heinrichs: Wenn die Software-Lösung keine bestimmte Datenbank vorgibt, ist meist der preisliche Aspekt ausschlaggebend. Wir können unseren Kunden die SQL-Server-Lösung am günstigsten anbieten. Wobei sich Microsoft mit seinem Lizenzmodell seit dem letzten Jahr auch an den anderen Herstellern orientiert und ebenfalls nach Prozessor-Leistung lizenziert, was die Kosten in die Höhe treibt. Bezüglich der Datenbank-Funktionalität fällt mir auf, dass die letzte Version von DB/2 neue Features enthält, die ich bei Oracle schon seit Jahren kenne.

Wie entwickelt sich Ihr Datenbank-Bereich weiter?

Heinrichs: Wir hatten im vergangenen Jahr ein Wachstum von 15 Prozent, wobei lediglich der Einsatz von DB/2 etwas rückläufig ist.

Wie stellen Sie den Betrieb Ihrer Datenbank-Systeme sicher?

Heinrichs: Fünf unserer sieben Administratoren haben eine Ausbildung in unserem Haus durchlaufen. Obwohl wir ein junges Team sind, arbeiten diese Mitarbeiter bereits seit durchschnittlich sieben Jahren bei uns, und jeder beherrscht mindestens drei verschiedene Datenbank-Systeme. So können wir Vertretungen gewährleisten sowie Dienste und Rufbereitschaften optimal abdecken. Die Gewinnung von Mitarbeitern durch eigene Ausbildung oder durch Trainee-Programme für Aachener Hochschulabsolventen ist bei uns seit vielen Jahren Praxis.

Wie schaut es bei den Datenbanken mit der Virtualisierung aus?

Heinrichs: Datenbanken sind bei uns generell nicht virtualisiert. Aufgrund der hohen I/O-Anforderungen setzen wir lieber auf physikalische Systeme und bieten dort Shared-Services an, indem wir mehrere Datenbanken auf einem Server betreiben.

Welche Produkte der Firma Oracle setzen Sie ein?

Heinrichs: Wir haben ausschließlich die Datenbank im Einsatz. Für die zentral am Standort Aachen zur Verfügung gestellten Systeme setzen wir auf die Enterprise Edition inklusive der Tuning- und Diagnostic-Packs. Neben mehreren Data-Guard-Lösungen bieten wir auch Aktiv/Passiv-Cluster auf Basis von Oracle Fail Safe an.

Wie dringend brauchen Sie die neue Datenbank 12c von Oracle?

Heinrichs: Es ist jetzt nichts dabei, von dem ich sage: Das müssen wir unbedingt sofort haben. Natürlich verfolge ich das Konzept der Pluggable Database mit Interesse, wir werden aber nicht sofort auf diesen Zug aufspringen, sondern erst einmal abwarten.

Haben Sie den Einsatz von Exadata schon evaluiert?

Heinrichs: Nein, das ist für uns eine Nummer zu groß. Wir können unsere Datenbank-Anwendung gut mit Standard-Hardware betreiben.

Was sagt Ihnen der Name „Sun“?

Heinrichs: Wir sind fast ausschließlich ein reines HP-Rechenzentrum und damit auch zufrieden, sodass sich bisher kein anderer Hardware-Anbieter durchsetzen konnte.

Wie beurteilen Sie den Oracle-Support im Vergleich mit den anderen Herstellern?

Heinrichs: Bisher haben wir mit Oracle keine negativen Erfahrungen gemacht. Bei IBM gefällt mir, dass es immer noch eine kleine Support-Organisation in Deutschland gibt, an die wir uns wenden können.

Wie garantieren Sie die Hochverfügbarkeit Ihrer Datenbanken angesichts der Notwendigkeit, diese patchen zu müssen?

Heinrichs: Wir haben mit unseren Kun-

den ausreichend Wartungsfenster vereinbart, um Datenbank-Patches oder -Migrationen durchführen zu können. Für die generelle Hochverfügbarkeit setzen wir im Oracle-Umfeld die beiden Produkte Data Guard und Fail Safe ein.

Wie garantieren Sie die Sicherheit Ihrer Daten?

Heinrichs: Unsere Prozesse sind alle qualitätsgesichert und nach drei international gültigen Standards zertifiziert. Diese umfassen neben Qualitätsmanagement und IT-Servicemanagement auch das Informationssicherheitsmanagement. Die Integration der Vorgaben in alle Prozesse sorgt für eine optimale Sicherheit unserer Daten. Alle kritischen Systeme sind in unseren Rechenzentren redundant ausgelegt.

Wie stellen Sie die Überwachung aller Systeme sicher?

Heinrichs: Für die gesamte Verfügbarkeitsüberwachung des SLA-Reporting setzen wir auf eine „nagios“-Verede-

regio iT auf einen Blick

Als IT-Dienstleister ist die regio iT GmbH ein fairer und zuverlässiger Partner für Kommunen und kommunale Unternehmen, Energieversorger und Entsorger, Schulen sowie Non-Profit-Organisationen. Mit Sitz in Aachen und Niederlassung in Gütersloh bietet die regio iT strategische und projektbezogene IT-Beratung, Integration, IT-Infrastruktur und Full-Service. Aktuell betreut sie mit rund 365 Mitarbeitern über 21.500 Clients und mehr als 145 Schulen. Das Unternehmen erwirtschaftete im Jahr 2012 einen Umsatz in Höhe von 46 Millionen Euro.

Die regio iT kann auf Erfahrungswerte aus mehreren Jahrzehnten zurückgreifen, denn viele ihrer Mitarbeiter kommen aus der Verwaltung sowie kommunalen Unternehmen und kennen die individuellen Wertschöpfungsketten ihrer Kunden.

lung, die sich „Open IT Cockpit“ nennt. Damit überwachen wir derzeit rund 17.500 Services. Im Bereich der Datenbank-Administration kommen außerdem Hersteller-Tools wie Oracle Enterprise Manager Grid beziehungsweise Cloud Control zum Einsatz.

Wie reagieren Sie auf Engpässe beziehungsweise Ausfälle?

Heinrichs: Die diensthabenden Kollegen werden direkt per E-Mail benachrichtigt und können dann direkt in die Fehlerbehebung einsteigen. Wie bei fast allen IT-Dienstleistern steigen die Servicezeiten allerdings immer mehr an und die Wartungsfenster an den IT-Systemen nehmen kontinuierlich ab. Manche Datenbanken für Städte und Gemeinden können jedoch abends problemlos nach Absprache mit den Kunden gewartet werden, während im Energiesektor um diese Zeit wichtige Verarbeitungen laufen und eine Wartung der Systeme nur an zwei bis vier Terminen im Jahr möglich ist. Mithilfe der Data-Guard-Lösungen sind wir allerdings in der Lage, Dienste auf ein anderes Rechenzentrum umzuleiten,

wenn ein System vor Ort gewartet werden muss.

Hat der Anwender Transparenz über den Status seiner Prozesse?

Heinrichs: Wir bieten dem Kunden eine Vielzahl von Möglichkeiten, die Qualität der erbrachten Services nachzuvollziehen. Das beginnt bei der Online-Schnittstelle im Incident Management und endet beim Einsatz von Kunden-Portalen, in denen die monatlichen Service Reports einsehbar sind. Service-Level-Agreements, die auf die Kunden eingehen, und deren Messung beziehungsweise Reporting machen unsere Leistungserbringung messbar.

Beraten Sie Ihre Kunden, wenn Sie Performance-Probleme erkennen?

Heinrichs: Wir haben den Anspruch, unsere Kunden gut zu bedienen, und zeigen ihnen deshalb auch, wie sie ihre Anwendungen optimal betreiben. Im Datenbank-Bereich geht es dann beispielsweise darum, ob eine Oracle Express Edition oder eine Enterprise Edition mit Tuning-Pack zum Einsatz kommt.

Welche Möglichkeiten bieten Sie Ihren Kunden hinsichtlich des Aufbaus ihrer Infrastruktur?

Heinrichs: Wir unterstützen den Kunden abhängig von seinen Anforderungen durch den Betrieb dedizierter Individual-Lösungen und stellen ihm Infrastructure-as-a-Service-Lösungen auf Shared-Service-Architekturen zur Verfügung. Auf die Datenbank-Systeme übertragen bedeutet das, dass wir standardisierte Shared-Service-Datenbankdienste genauso betreiben wie individuelle Cluster- oder Data-Guard-Systeme.

Sehen Sie sich auch in der Rolle eines Cloud-Anbieters?

Heinrichs: Wir bieten schon jetzt erste Dienste in der Cloud unter der Marke „stadtwolke“ an. Hier geht es darum, Cloud-Services für Kommunen rechtsicher und geschützt anzubieten, indem wir die Daten hierzulande und nach zertifizierten Qualitätsstandards speichern. In Zukunft werden wir weitere Dienste in diese Stadtwolke integrieren.

Wie beurteilen Sie die Produktstrategie von Oracle?

Heinrichs: Wir verfolgen die Zukäufe schon, setzen allerdings nur die Datenbank-Produkte ein.

Wie interessant sind die Oracle-Business-Lösungen für Sie?

Heinrichs: Wir haben im Business-Bereich zahlreiche SAP-Lösungen. Für die Kommunen ist in erster Linie der Microsoft SQL Server mit seinen Applikationen interessant. Oracle nehme ich in diesem Umfeld nicht als aktiven Player wahr.

Können Sie sich vorstellen, ein Komplettsystem von der Hardware bis zu den Applikationen von einem einzigen Hersteller wie Oracle einzusetzen?

Heinrichs: Das ist für mich mit keinem Anbieter denkbar.

In welche Richtung wird sich Ihre IT in den kommenden Jahren entwickeln?

Heinrichs: Die regio iT wächst weiter. Auf der Agenda steht vor allem die Fortführung von Standardisierung, Konsolidierung und Virtualisierung. Ein weiteres Ziel besteht darin, unsere Dienste Cloud-fähig zu machen. Auch das Thema „Mobile Computing“ ist für unsere Kunden zunehmend wichtiger.

Was erwarten Sie dabei von einem IT-Unternehmen wie Oracle?

Heinrichs: Im Gegensatz zu früher nehme ich die Präsenz von Oracle kaum mehr wahr. Ich vermisse vor allem über einen längeren Zeitraum dieselben persönlichen Ansprechpartner.

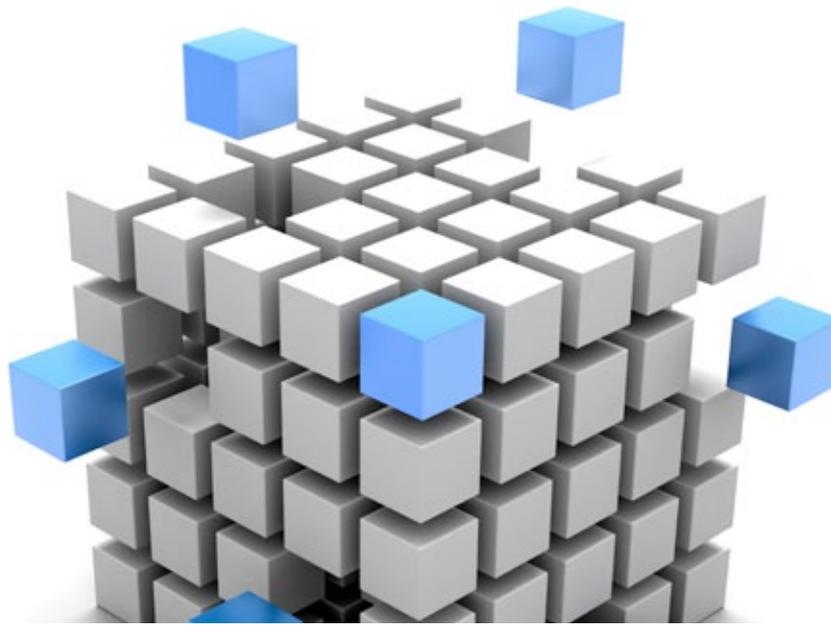
Wie sehen Sie den Stellenwert einer Anwendergruppe wie der DOAG?

Heinrichs: Wir nutzen seit vielen Jahren die Dienste der DOAG beim Wisenserwerb, beim Erfahrungsaustausch, aber auch deren unabhängige Beratung etwa in Lizenzfragen. Die kontinuierliche Fortbildung der Mitarbeiter ist uns sehr wichtig. Deshalb besuchen wir auch gerne die DOAG Konferenz, die Schulungstage sowie die Expertenseminare. Es ist gut, die DOAG als Partner auf unserer Seite zu haben.



Zur Person: Karsten Heinrichs

Karsten Heinrichs absolvierte von 1994 bis 1997 seine Ausbildung zum Datenverarbeitungskaufmann bei der GKDVZ Aachen, einem der Vorgänger der heutigen regio iT. Seit dem Jahr 1996 ist er im Team „Database Systems“ tätig, das er seit 2007 leitet. Seit einem Jahr verantwortet er außerdem das Produktmanagement der regio iT Datenbank-Produkte und -Dienstleistungen. Karsten Heinrichs ist verheiratet, hat zwei Kinder und lebt in Heinsberg.



Oracle Multitenant: Einfachere Möglichkeiten für die Datenbank-Konsolidierung und mehr

Manuel Hoßfeld, ORACLE Deutschland B.V. & Co. KG

Die neue Option namens „Oracle Multitenant“ basiert auf einer neuen Datenbank-Architektur, die es optional erlaubt, mehrere sogenannte „Pluggable Databases“ in einem „Multitenant Container“ – also einer „Container-Datenbank“ – zu betreiben. Der Artikel zeigt, was das genau für Datenbank-User und DBAs bedeutet und welche Möglichkeiten sich bei der Nutzung dieser Option ergeben.

Ein Oracle-Datenbanksystem besteht bis zur Version 11 grundsätzlich aus zwei Komponenten: der Datenbank selbst in Form von Dateien und einer Instanz (oder mehrerer im Fall von RAC) mit den dazugehörigen Betriebssystem-Prozessen und Hauptspeicher. Werden mehrere Datenbank-Systeme auf einem Server betrieben, sind entsprechend viele Instanzen aktiv.

Nun ist diese klassische Architektur zwar schon lange in dem Sinne mandantenfähig (engl. „multitenant“), dass man einzelnen Anwendungen beziehungsweise Mandanten verschiedene Datenbank-Schemata zuweisen kann. Dieses Konzept stößt in der Praxis jedoch schnell an seine Grenzen, wenn man daran denkt, dass es jeden Schema-Namen pro Datenbank-Instanz nur einmal geben kann, ebenso wie sämtliche Parameter für deren Betrieb.

Im Rahmen von Oracle Multitenant wird dieses Konzept geändert: Für bessere Ressourcen-Nutzung und neue Möglichkeiten der Konsolidierung innerhalb einer Datenbank wurde nun eine neue Ebene eingeführt, die gewissermaßen zwischen Instanz und Schema angesiedelt ist: die sogenannte „Pluggable Database“ – eine nach außen hin vollständige Datenbank mit einem eigenen Satz Einstellungen und Schemata, die aber nicht autark auf einem Datenbank-Server Ressourcen beansprucht, sondern sich diese mit anderen solcher „Pluggable Databases“ innerhalb einer „Container-Datenbank“ teilt.

Grundsätzlich ändert sich bei der Nutzung einer Pluggable Database aus Sicht eines Endanwenders nichts: Für ihn verhält sie sich genauso wie eine normale Datenbank auch. Wichtig zu wissen ist lediglich, dass der Connect immer über einen Service erfolgt, da

nur so eine eindeutige Identifikation der gewünschten Pluggable Database möglich ist.

Neue Architektur und Nomenklatur

Mit der Multitenant-Architektur gibt es einige neue Begrifflichkeiten, die man im Hinterkopf behalten sollte (siehe [Abbildung 1](#)):

- *Pluggable Database (PDB)*
Eine für Anwendungen genutzte Datenbank, die unter Verwendung der neuen Architektur betrieben wird. Sie enthält alle Anwendungsdaten samt Data Dictionary, Datenbank-Benutzern etc.
- *Container Database (CDB)*
Der Container, in den die PDBs eingeklinkt werden. Alle PDBs innerhalb einer CDB benutzen die gleiche Datenbank-Version, die durch die CDB vorgegeben ist.

Ohne Umwege zu optimaler Performance.

areto hat die Lösung.



ORACLE Gold
Partner

Strukturierte Datenmodellierung, konsequente Technologie-Nutzung und zielführende Projektstandards ziehen sich von Anfang an als roter Faden durch die Entwicklung unserer Business-Intelligence-Lösungen. Sodass am Ende selbst bei komplexen Anforderungen überraschend einfache und intelligente Lösungen stehen.

Bringen Sie auch in Ihrem Unternehmen den Knoten zum Platzen.

**Rufen Sie uns an:
0221 66 95 75-75**

areto consulting gmbh · Data Warehouse · Business Intelligence
Julius-Bau-Straße 2 · 51063 Köln · 0221 66 95 75-0 · www.areto-consulting.de

areto
CONSULTING. IT WORKS.

- **Non-CDB**
Eine Oracle-Datenbank, die nach herkömmlicher Architektur unter Oracle 12c betrieben wird.
- **Seed-DB**
Ein Datenbank-Template, mit dem sehr schnell eine neue PDB erzeugt werden kann. Jede CDB enthält genau eine Seed-DB mit dem Namen „PDB\$SEED“.

Eine PDB unterscheidet sich von einer Non-CDB dadurch, dass sie keine der folgenden eigene(n) Komponenten besitzt:

- Instanz (Memory und Hintergrundprozesse)
- Kontrolldatei
- Redo-Log-Dateien
- Undo Tablespace

Diese Komponenten werden zentral von der CDB bereitgestellt, in der die PDB betrieben wird. Auch beim Inhalt der Datenbanken gibt es Unterschiede zum klassischen Konzept. Im Rahmen der Pluggable Databases wird das Data Dictionary in folgende Bereiche aufgeteilt:

- Einen statischen Bereich, der die für die eingesetzte Software feststehenden Informationen beinhaltet. Dieser Bereich enthält Informationen, die während der Lebenszeit der Datenbank mit der jeweils gegebenen Version nicht mehr verändert werden. Dieser wird auch als „root“ bezeichnet.
- Einen anwendungsbezogenen Bereich, dessen Inhalte Veränderungen unterliegen können.

Der statische Bereich ist Bestandteil der CDB und wird über eine Verlinkung in den anwendungsbezogenen Bereich, der Bestandteil der PDB ist, einbezogen. Aus der Sicht einer PDB hat diese also ein komplettes Data Dictionary, auch wenn die nicht veränderlichen Teile aus der CDB stammen.

Vorteile der neuen Multitenant-Architektur

Die neue Architektur bietet folgende interessante Vorteile und Anwendungsmöglichkeiten:

- **Konsolidierung und Einsparung von Ressourcen**
Dadurch dass Prozesse, RAM und große Teile des Data Dictionary nur einmal vorhanden sind, können wesentlich mehr Datenbanken auf einem System betrieben werden als mit der klassischen Architektur
- **Mobilität**
Schnelles und einfaches „Unplug“ und „Plug“ einer kompletten PDB
- **Cloud-Fähigkeit**
Schnelles Bereitstellen einer PDB durch Erzeugen aus der Seed-DB oder als Klon einer bestehenden PDB sind die ideale Basis für DBaaS (Database-as-a-Service)
- **Patchen/Migration**
Wenn eine CDB einmal gepatcht ist, sind dies automatisch auch alle darin enthaltenen PDBs
- **Effizientere Möglichkeit für Disaster Recovery sowie Backup & Recovery**
Sowohl Data Guard als auch Backups werden zentral auf Ebene einer CDB eingerichtet. Ein Restore eines Backups ist dennoch auch für einzelne PDBs möglich

Der Weg zur Pluggable Database

Die einfachsten Möglichkeiten stellen natürlich die Wahl der entsprechenden Installationsoption im „Oracle Universal Installer“ oder die Nutzung des „Database Creation Assistant“ (DBCA) dar. Diese ergeben sich allerdings nur für neue Installationen. Da die Oracle-Landschaft vieler Anwender jedoch eher nicht der sprichwörtlichen „grünen Wiese“ entspricht, seien an dieser Stelle kurz zwei Möglichkeiten

erwähnt, wie man bereits in einer Version vor 12c bestehende Datenbanken „pluggable“ machen kann. Voraussetzung ist jeweils das Vorhandensein einer installierten 12c-Datenbank, um überhaupt eine CDB zur Verfügung zu haben. Diese muss lediglich laufen und kann gegebenenfalls auch vollkommen leer sein. Die folgenden Schritte stellen nur einen groben Leitfaden dar – die Details stehen in der Dokumentation sowie in dem am Ende des Artikels beschriebenen Oracle-Dojo Nr. 7.

Direktes Einklinken einer Non-CDB als PDB

Hierzu muss zunächst ein Upgrade der Quell-Datenbank auf 12c erfolgen, da nur ab 12c das für eine Umwandlung von Non-CDB in PDB verwendete Package „DBMS_PDB“ überhaupt vorhanden ist. Nach dem Upgrade erzeugt man mithilfe dieses Package einen Satz Metadaten im XML-Format, der Informationen zu den Einstellungen und Datendateien der Quell-Datenbank beinhaltet. Für den gängigen Fall, dass die neue 12c-Datenbank nicht auf dem gleichen Server läuft wie die Quell-Datenbank, kopiert man die gerade erzeugte XML-Datei sowie die Daten-Dateien auf den neuen Server.

Im nächsten Schritt wendet man das Kommando „CREATE PLUGGABLE DATABASE <Name> USING <Pfad-zur-XML-Datei>...“ an. Je nachdem, wo die Dateien liegen, kommen dabei verschiedene Optionen zum Einsatz, die zum Beispiel die Dateinamen an die neuen Lokationen anpassen. Mit dem Skript „\$ORACLE_HOME/rdbms/

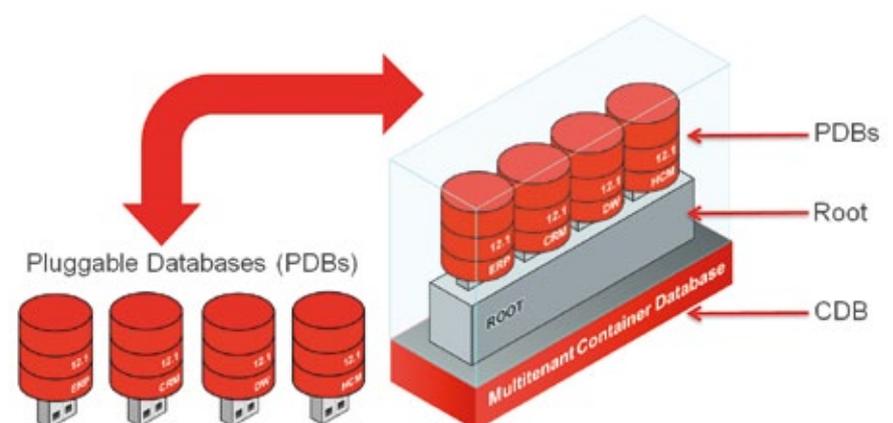


Abbildung 1: Komponenten und Begriffe von Oracle Multitenant

| Pro CDB | Pro PDB |
|------------------------------------|---|
| Eine Oracle-Software-Version | RMAN Point-in-time Recovery |
| Geplante RMAN-Backups | „Ad hoc“-RMAN-Backups |
| Data Guard | Möglichkeit für „Flush Shared_Pool“ |
| Einige Parameter wie Character Set | Parameter, für die gilt „IsPDB_Modifiable = "TRUE"“ |
| Redo und Undo | |

Tabelle 1

admin/noncdb_to_pdb.sql“ wird nun der statische Anteil der (ehemaligen) Non-CDB endgültig entfernt, damit diese als PDB laufen kann.

Importieren eines Full Database Exports in eine leere PDB

Hierbei wird zunächst eine neue leere PDB in der 12c-CDB erzeugt – einschließlich des dazugehörigen Service, um diese überhaupt ansprechen zu können. Anschließend wird ein mittels „expdp ... FULL=Y“ erzeugter „Full Database Export“ in die soeben erzeugte PDB importiert. Hierbei kann einem ein weiteres neues Feature von 12c behilflich sein: Data Pump unterstützt jetzt nämlich auch die Verwendung von Transportable Tablespaces, sodass das eigentliche Dump File sehr klein bleibt und schnell erzeugt ist. Hinweis: Ein Backport dieses Features ist auch in 11.2.0.3 eingeflossen, sodass es entsprechend auch für eine Migration einer 11g-Non-CDB in eine 12c-PDB verwendet werden kann.

CDB vs. PDB

Wie bereits eingangs erwähnt, verhält sich eine PDB nach außen hin grundsätzlich wie eine Non-CDB. Wegen der zugrunde liegenden Architektur mit geteilten Ressourcen aller PDBs innerhalb einer CDB existieren gleichzeitig aber auch Merkmale und Funktionen, die sich demzufolge nicht auf der Ebene einzelner PDBs abspielen. **Tabelle 1** listet die wichtigsten auf.

Lizenzrechtlicher Hinweis

Wie schon aufgrund des Namens „Oracle Multitenant Option“ zu erwarten ist, handelt es sich bei den in diesem Artikel vorgestellten Möglichkeiten um eine zusätzlich zur Datenbank zu lizenzierende Option. Weniger offensichtlich ist die Tatsache, dass dies für den Sonderfall von nur einer einzelnen PDB in einer CDB nicht gilt. Im Umkehrschluss bedeutet dies, dass auch ohne die entsprechende Option sämtliche 12c-Datenbanken mit einer

einzelnen PDB betrieben werden können – auch in der Standard Edition.

Weitere Informationen

1. Umfangreiches (englischsprachiges) Whitepaper zu Oracle Multitenant: <http://www.oracle.com/technetwork/database/multitenant-wp-12c-1949736.pdf>
2. Aus der Reihe der deutschsprachigen „Oracle Dojos“ die Ausgabe Nr. 7 zum Thema „Multitenant“ – sowohl in gedruckter Form als auch online: <http://www.oracle.com/webfolder/technetwork/de/community/dojo/index.html>
3. Oracle Database Days: <http://tinyurl.com/odd12c>

Manuel Hoßfeld
manuel.hossfeld@oracle.com



Automatisierung des Information Lifecycle Managements

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

Was ist neu in Oracle Database 12c im Bereich „Information Lifecycle Management“? Was sind die sogenannten „Heat Maps“? Wie sieht die Automatisierung aus? Wie aufwändig ist das Setup? Dieser Artikel beantwortet diese und weitere Fragen und illustriert sie an einigen Beispielen.

Information Lifecycle Management (ILM) ist kein neues Schlagwort, sondern ein gängiger Begriff, um den Lebenszy-

klus von Daten zu beschreiben. Ziel ist es, Informationen entsprechend ihrem Wert und ihrer Nutzung optimal auf

dem jeweils kostengünstigsten Speichermedium im besten Fall automatisch zu platzieren – so die Definitionen im Web.

Die Techniken zur Realisierung von ILM wie Partitionierung, Speicherplatz-Einsparung durch Komprimierung, Verlagerung von Daten und Virtual Private Database (VPD) für die unterschiedlichen Sichtweisen auf die Daten sind schon lange Bestandteil der Oracle-Datenbank und werden mit jedem Release weiterentwickelt. Allerdings musste eine Automatisierung bisher über eigene Programme oder spezielle Werkzeuge implementiert werden, nachdem man die Daten vorab manuell kategorisiert hat. Neu in Oracle Database 12c ist nun die vollständige Integration von ILM-Features in die Datenbank. Zur Implementierung sind keine zusätzlichen Werkzeuge oder Skripte erforderlich.

Wichtige Voraussetzung für den Einsatz dieser neuen Technologie ist die Lizenzierung der Advanced-Compression-Option. Im Wesentlichen handelt es sich dabei um zwei neue Features: Heat Map und automatische Daten-Optimierung (Automatic Data Optimization). Heat Map verfolgt Veränderungen und Abfragen auf Zeilen- und Segment-Ebene und gibt einen detaillierten Überblick über den Zugriff auf die Daten. Die automatische Daten-Optimierung verlagert und/oder komprimiert die Daten gemäß Nutzer-definierter Regeln (Policies), basierend auf den Informationen, die sich aus der Heat Map ergeben.

Wichtig zu wissen ist, dass in Release 1 noch einige Einschränkungen existieren, die unbedingt bei der Nutzung zu berücksichtigen sind. Die ILM-Funkti-

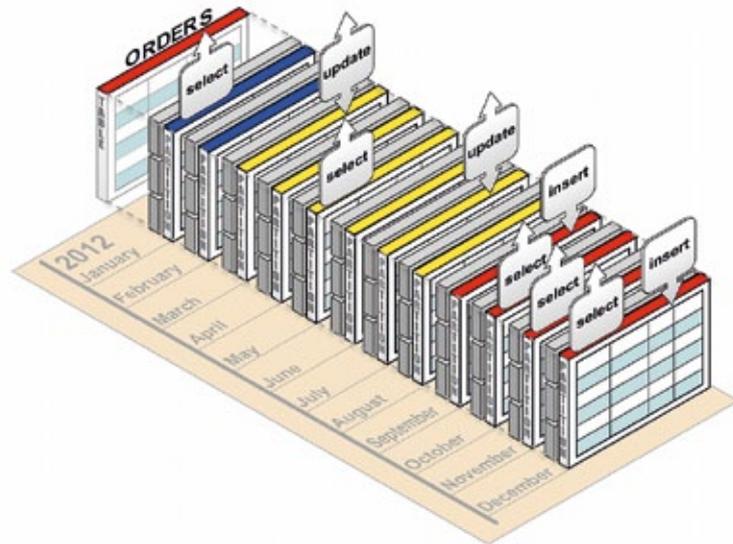


Abbildung 1: Heat Map – Zugriffe auf ein Segment

onen stehen beispielsweise momentan nur in einer Nicht-Multitenant-Architektur (NON CDB) zur Verfügung.

Automatische Klassifizierung von Daten

Bevor über eine Automatisierung der optimalen Datenablage nachgedacht werden kann, sind die Daten zu kategorisieren. Im Klartext bedeutet dies, dass sie je nach Zugriffsstatistik in unterschiedliche Kategorien eingeteilt werden – natürlich automatisch ohne Interaktion durch den Nutzer oder DBA. In Oracle Database 12c ist diese Funktion über das sogenannte „Heat-Map-Feature“ implementiert. Heat Maps geben einen Überblick über die Aktivi-

täten auf den unterschiedlichen Objekten. Dabei wird nicht nur die Segment-Ebene berücksichtigt, sondern sogar die aktuellste Veränderung auf Block-Ebene dokumentiert (siehe Abbildung 1). Die Aktivitäten bestehen aus Lese- und Schreib-Operationen. Sogar Table beziehungsweise Index Lookups werden mitgeschrieben. Damit kein verfälschtes Bild entsteht, sind operative Eingriffe wie Statistik-Management, Verlagerungen etc. nicht vermerkt.

Eingeschaltet wird das Heat-Map-Feature über einen einzigen dynamischen Initialisierungs-Parameter „HEAT_MAP“ („ON“ bzw. „OFF“). Die Default-Einstellung ist „OFF“: „SQL> ALTER SYSTEM set HEAT_MAP=ON;“. Danach werden automatisch alle Zugriffe über einen speziellen In-Memory-Zugriff geloggt. Zugriffe auf Objekte im SYSTEM- und SYSAUX-Tablespace werden dabei ausgelassen.

Der Zugriff auf die Heat Map kann anschließend über Standard-Schnittstellen wie Data-Dictionary-Views, Fixed Tables oder PL/SQL-Packages erfolgen. Eine grafische Schnittstelle in Cloud Control 12c ist geplant (Stand August 2013). Die nachfolgenden Beispiele geben einen kleinen Einblick in die verschiedenartigen Verwendungsweisen. Einen ersten aktuellen Einblick liefert die View „V\$HEAT_MAP_SEGMENT“. Sie zeigt Realtime-Informationen über die Segment-Zugriffe. Listing 1 zeigt ei-

```
SELECT h.object_name, o.object_type, h.track_time, h.segment_write
write, h.segment_read read, h.full_scan, h.lookup_scan
FROM v$heat_map_segment h join dba_objects o
ON (o.object_id=h.obj#);
```

| OBJECT_NAME | OBJECT_TYPE | TRACK_TIME | WRI | REA | FUL | LOO |
|--------------|-------------|------------------|-----|-----|-----|-----|
| PRODUCTS | TABLE | 11.07.2013 15:19 | YES | NO | NO | NO |
| DEPARTMENTS | TABLE | 11.07.2013 15:19 | NO | NO | NO | NO |
| CONFIG\$ | TABLE | 11.07.2013 15:19 | NO | NO | NO | NO |
| TAB_300_300 | INDEX | 11.07.2013 15:19 | NO | NO | NO | NO |
| TAB_300 | TABLE | 11.07.2013 15:19 | NO | NO | YES | NO |
| TAB_300_OLTP | TABLE | 11.07.2013 15:19 | NO | NO | NO | NO |
| TAB_300_OLTP | TABLE | 11.07.2013 15:19 | NO | NO | NO | NO |
| ... | | | | | | |

Listing 1

```
SELECT object_name, segment_write_time write, full_scan, lookup_scan FROM dba_heat_map_segment
WHERE owner = 'SH' AND object_name != 'SALES' ORDER BY 2,3;
```

| OBJECT_NAME | WRITE | FULL_SCAN | LOOKUP_SCAN |
|------------------|------------------|------------------|------------------|
| TAB_300_OLTP | 10.07.2013 17:21 | 10.07.2013 17:21 | |
| TAB_300 | 10.07.2013 17:21 | 11.07.2013 15:01 | |
| TABLE_300_SELECT | 10.07.2013 18:22 | | |
| PRODUCTS | 11.07.2013 15:01 | 26.06.2013 12:30 | 25.06.2013 12:48 |
| PRODUCTS_PK | 11.07.2013 15:01 | | 11.07.2013 15:01 |
| CUSTOMERS_PK | | | 25.06.2013 22:48 |
| ... | | | |

Listing 2

```
SELECT tablespace_name, segment_count, allocated_bytes, min_writetime, max_writetime
FROM dba_heatmap_top_tablespaces;
```

| TABLESPACE_NAME | SEGMENT_COUNT | ALLOCATED_BYTES | MIN_WRITE | MAX_WRITE |
|-----------------|---------------|-----------------|-----------|-----------|
| ADOTEST | 16 | 135266304 | | |
| EXAMPLE | 334 | 190775296 | 18-JUN-13 | 05-JUL-13 |
| USERS | 20 | 5235408896 | 21-JUN-13 | 10-JUL-13 |

Listing 3

nen Auszug zur Track-Zeit „15:19 Uhr am 11. Juli“. Die Objekte, deren Spalten den Wert „YES“ beinhalten, sind aktuell im Zugriff.

Mit „ALL_, DBA_“ und „USER_HEAT_MAP_SEGMENT“ kann man den letzten Zugriff auf die Segmente anzeigen. Folgendes Code-Beispiel listet die Zugriffe auf einige der Objekte des Users „SH“ auf. Wie zu erkennen ist, sind Lese- und Schreib-Zugriffe dokumentiert und darüber hinaus die Full Table

Scans beziehungsweise die zugehörigen Index Scans aufgelistet (siehe Listing 2).

Aus Gründen der Übersichtlichkeit ist die Spalte „SUBOBJECT_NAME“ nicht selektiert. Im Fall von Zugriffen auf partitionierte Tabellen wie zum Beispiel bei der Tabelle „SALES“ ist dies natürlich zur genauen Segment-Analyse unbedingt erforderlich.

Interessiert man sich nur für die Objekte und Tablespaces, die am häufigsten im Zugriff waren, kann man

die Views „DBA_HEATMAP_TOP_OBJECTS“ beziehungsweise „DBA_HEATMAP_TOP_TABLESPACES“ verwenden. In Listing 3 sind die drei Top-Tablespaces unter Angabe der jeweiligen Objektanzahl und des verwendeten Speicherplatzes aufgelistet.

So wird beispielsweise der erste beziehungsweise letzte Schreibzugriff auf Objekte im entsprechenden Tablespace mitgeschrieben. Die Funktion „BLOCK_HEAT_MAP“ des Package „DBMS_HEAT_MAP“ listet die zuletzt erfolgten Schreibzugriffe sogar bis auf Block-Ebene genau auf. Listing 4 zeigt, welche Blöcke der Tabelle „TAB_300“ des Users „COMP_TEST“ zuletzt geändert worden sind.

Automatische Daten-Optimierung

Um zu erklären, was mit automatischer Daten-Optimierung (Automatic Data Optimization) gemeint ist, skizziert folgendes Beispiel einen möglichen Anwendungsfall (siehe Abbildung 2). Zuerst werden Daten mit Bulk-Load-Operationen oder anderen konventionellen Methoden in die Ta-

```
SELECT tablespace_name, relative_fno fno, block_id, to_
char(writetime, 'MM/DD/YY HH24:MI.SS' )
FROM TABLE(dbms_heat_map.block_heat_map('COMP_TEST','TAB_300'));
```

| FNO | BLOCK_ID | TO_CHAR(WRITETIME) |
|-----|----------|--------------------|
| 5 | 251 | 07/11/13 16:20.33 |
| 5 | 252 | 07/11/13 16:20.33 |
| 5 | 253 | 07/11/13 16:20.33 |
| 5 | 254 | 07/11/13 16:20.33 |
| 5 | 255 | 07/11/13 16:20.33 |
| 5 | 6656 | 07/11/13 16:20.33 |

Listing 4

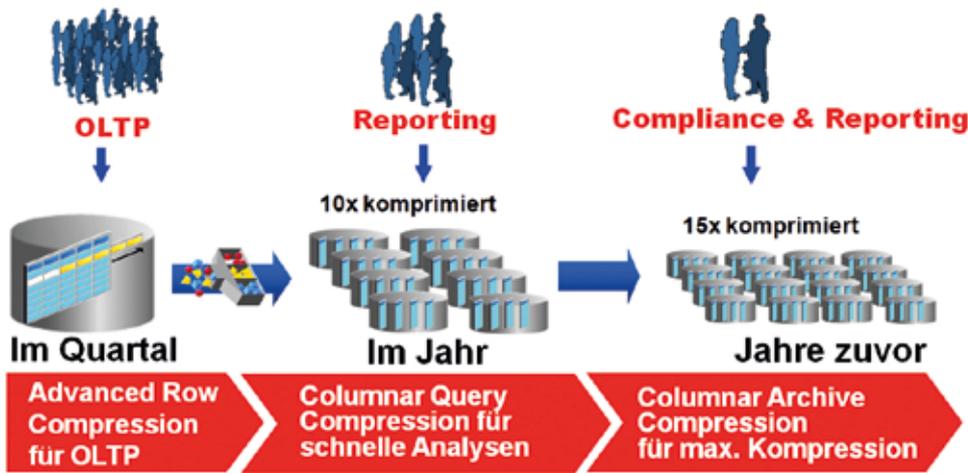


Abbildung 2: Beispiel für automatische Daten-Optimierung

bellern geladen und unterliegen dabei starken Veränderungen. Zu diesem Zeitpunkt ist eine Komprimierung noch nicht erwünscht, somit werden

die Daten unkomprimiert gespeichert. Nach einer gewissen Zeit erfolgen nur noch vereinzelt Veränderungen über OLTP-Transaktionen. Nun kann es sinnvoll sein, die Daten aus Platzgründen in das OLTP-Komprimierungsformat zu konvertieren. Nach einiger Zeit werden die Daten nur noch selten genutzt; nun könnten sie auf ein anderes Speichermedium ausgelagert werden – vielleicht sogar auf ein Speichermedium wie ZFS oder Pillar, die eine höhere Komprimierungsrate durch HCC-Komprimierung erlauben, oder einfach nur auf ein beliebiges anderes Low-Cost-Storage-Medium.

Dieses Szenario lässt sich ab Oracle Database 12c einfach in der Datenbank abbilden und kann sogar in automatisierter Form ablaufen. Wichtiges Hilfsmittel zur Implementierung von automatischer Daten-Optimierung sind die sogenannten „Policies“. Eine Policy-Spezifikation beinhaltet dabei eine Aktion wie zum Beispiel Compression oder Storage Tiering und zusätzlich eine Bedingung, unter der die Aktion ausgeführt werden soll. Policies können auf Segment- oder Row-Ebene implementiert sein. Listing 5 zeigt eine Segment-Level-Policy (Schlüsselwort „SEGMENT“), die automatisch eine Tabelle in das OLTP-Komprimierungsformat umwandelt, nachdem zwanzig Tage lang keine Veränderungen (Schlüsselwort „NO MODIFICATION“) erfolgt sind. „ROW STORE COMPRESS ADVANCED“ ist dabei die neu eingeführte Syntax für OLTP-Compression in 12c.

```
ALTER TABLE sh.sales
ILM ADD POLICY ROW STORE COMPRESS
ADVANCED SEGMENT
AFTER 20 DAY OF NO MODIFICATION;
```

Listing 5

```
ALTER TABLE sh.customers
ILM ADD POLICY ROW STORE COMPRESS
ADVANCED ROW
AFTER 1 DAY OF NO MODIFICATION;
```

Listing 6

```
ALTER TABLE sh.sales
ILM ADD POLICY TIER TO adotest;
```

Listing 7

```
SELECT SUBSTR(name,1,32) name,
value
FROM dba_ilmparameters WHERE name
LIKE ',TBS%';
```

| NAME | VALUE |
|------------------|-------|
| TBS PERCENT USED | 70 |
| TBS PERCENT FREE | 30 |

Listing 8

Das nächste Beispiel zeigt eine „ROW“-Policy. Oracle evaluiert in regelmäßigen Abständen die Blöcke der „ORDERS“-Tabelle. Jeder Block, der den Anforderungen – in diesem Fall „1 DAY OF NO MODIFICATION“ – genügt, wird komprimiert, um wieder ausreichend Platz zu schaffen (siehe Listing 6).

Diese Policy erlaubt höchst mögliche Performance beim Laden von Daten und zusätzlich die Vorteile der Platzersparnis durch die Komprimierung. Im Gegensatz zum Segment-Policy-Beispiel muss nicht darauf gewartet werden, bis eine ganze Partition den Anforderungen genügt.

Nun stellt sich die Frage, wann diese Policies evaluiert werden. Da die Aktionen automatisch im Hintergrund ablaufen sollen, bietet sich das Maintenance Window des „Default Maintenance“-Plans zur Evaluierung und Ausführung an. Möchte der DBA eingreifen, ist dies natürlich auch über die entsprechenden Kommandos möglich.

Zusätzlich zu den gerade illustrierten Aktionen über Komprimierung gibt es auch die Möglichkeit der Verlagerung auf einen anderen Storage (Storage Tiering). Die Implementierung ist einfach, wie Listing 7 zeigt.

Diese Art von Policy wird durch den Füllgrad des entsprechenden Tablespace getriggert. Ist der Quell-Tablespace (hier „USERS“) annähernd voll, werden die Daten in den Tablespace „ADOTEST“ verlagert. Der DBA kann dabei die Eigenschaft „voll“ beeinflussen. Im Beispiel ist der Tablespace bei 70 Prozent voll, wie die eingestellten Parameter zeigen (siehe Listing 8).

Policies können natürlich ein- und ausgeschaltet beziehungsweise auch gelöscht werden. Ein Monitoring ist wie beim Heat-Map-Feature über die entsprechenden Views wie „DBA_ILMDATAMOVEMENTPOLICIES“ und „DBA_ILMOBJECTS“ möglich. Das abschließende Beispiel zeigt einen Ausschnitt aus den im Artikel genutzten Policies (siehe Listing 9).

Fazit

Die neuen ILM-Features ermöglichen eine automatische Datenkomprimierung beziehungsweise automatisierte

Verlagerung der Daten auf ein anderes Speichermedium. Wichtige Voraussetzung ist dabei die Nutzung beziehungsweise das Einschalten des Heat-Map-Features auf System-Ebene. Die Operationen erfolgen dabei im Hintergrund und können im Fall von Partitionen sogar vollständig online ausgeführt werden.

Der Artikel hat nur einen kleinen Ausschnitt der Funktionalität demonstriert. So ist es zum Beispiel auch möglich, eigene Funktionen zu definieren, um den Zeitpunkt des Storage Tiering oder der Komprimierung festzulegen. Wer mehr darüber erfahren möchte, kann Handbücher, Blogs und White Paper konsultieren.

Weitere Informationen

1. Tipps in der DBA Community: http://blogs.oracle.com/dbacommunity_deutsch
2. VLDB and Partitioning Guide
3. SQL Language Reference (Syntax wie „ALTER TABLE“)

| OBJECT_NAME | SUBOBJECT_NAME | ACTION_TYPE | SCOPE | TIER_TABLESPACE |
|------------------------|----------------|-------------|---------|-----------------|
| CONDITION_TYPE | DAYS | | | |
| CUSTOMERS | | COMPRESSION | ROW | |
| LAST MODIFICATION TIME | 1 | | | |
| SALES | SALES_1995 | STORAGE | SEGMENT | ADOTEST |
| | | 0 | | |
| SALES | SALES_1996 | STORAGE | SEGMENT | ADOTEST |
| | | 0 | | |
| SALES | SALES_H1_1997 | STORAGE | SEGMENT | ADOTEST |
| | | 0 | | |

Listing 9

4. PL/SQL-Packages and Types-Reference
5. Automatic Data Optimization with Oracle Database 12c
6. Deutschsprachige Veranstaltungen zum Thema: tinyurl.com/odd12c

Ulrike Schwinn
ulrike.schwinn@oracle.com



www.dba-im-urlaub.de

MUNIQSOFT
Datenbanken mit iQ



Neue Funktionen für Entwickler

Carsten Czarski, ORACLE Deutschland B.V. & Co. KG

Das neue Datenbank-Release bringt insgesamt mehr als 500 neue Funktionen mit. Auch für SQL- und PL/SQL-Entwickler sind mit SQL Pattern Matching, Identity Columns, unsichtbaren Spalten und anderen neuen Funktionen eine Menge nützlicher Dinge dabei. Dieser Artikel gibt einen Einblick in die Neuerungen.

32K VARCHAR überall

„VARCHAR2“-Spalten einer Tabelle können ab Oracle 12c bis zu 32.767 Bytes aufnehmen und verhalten sich damit genauso wie Variablen dieses Typs in PL/SQL. Wenn man also abschätzen kann, dass die benötigte Länge unterhalb von 32 KB bleibt, kann man sich den (komplizierteren) Einsatz von CLOBs sparen ([siehe Listing 1](#)).

Ganz ohne den DBA geht es dann aber doch nicht. Damit alles funktioniert, muss dieser den Datenbank-Parameter „MAX_STRING_SIZE“ auf „EXTENDED“ setzen und pro Datenbank einmal das Skript „\$ORACLE_HOME/rdbms/admin/utl32k.sql“ ablaufen lassen. Ausführliche Informationen dazu

```
create table meine_tabelle(  
  id          number(10),  
  langertext  varchar2(10000)  
);
```

Listing 1

stehen in der Oracle Database Reference unter „MAX_STRING_SIZE“. In einer Apex-Umgebung sollte danach zusätzlich das Skript „\$Apex_HOME/core/collection_member_resize.sql“ gestartet werden, damit das neue Limit auch in Apex Collections möglich ist.

Sequences „By Default“ und Identity Columns

Der Umgang mit Sequences und das Generieren eindeutiger Werte für eine Primärschlüssel-Spalte werden in Oracle 12c wesentlich einfacher. So ist es nun möglich, den nächsten Wert der Sequence als Default für eine Ta-

bellenspalte zu hinterlegen. Das bislang nötige und lästige Erstellen des Triggers gehört damit der Vergangenheit an ([siehe Listing 2](#)).

Es geht sogar noch einfacher: Die „GENERATED AS IDENTITY“-Klausel nimmt die Definition der Sequence zur Tabelle. Das explizite Erzeugen einer Sequence fällt damit weg ([siehe Listing 3](#)).

Die „GENERATED ... AS IDENTITY“-Klausel kennt noch einige Variationen. Die dargestellte Variante „GENERATED ALWAYS“ nimmt immer den Wert der Sequence für die Tabellenspalte, auch wenn im „SQL INSERT“ explizit ein Wert vorgegeben wurde. Alternativen

```
create sequence seq_id start with 1 increment by 1;  
  
create table meine_tabelle(  
  id          number(10)      default seq_id.nextval,  
  text        varchar2(10000)  
);
```

Listing 2

tiv lässt sich ein „GENERATED BY DEFAULT“ oder „BY DEFAULT ON NULL“ spezifizieren (siehe [Abbildung 1](#)).

Invisible Columns

Tabellenspalten können in Oracle 12c quasi unsichtbar gemacht werden – dazu dient das neue Spaltenattribut „INVISIBLE“ (siehe [Listing 4](#)).

Die Tabellenspalten (hier „HIRE-DATE“ und „COMM“) sind tatsächlich nicht mehr sichtbar. Invisible Columns sind dann sinnvoll, wenn es darum geht, dass andere Nutzer mit einer Spalte nicht arbeiten sollten. Das könnten Spalten sein, die allein für Partitionierung angelegt wurden, deren Inhalte nur für bestimmte PL/SQL-Pakete von Belang sind oder ähnliche Gründe. Eine Spalte unsichtbar zu machen, ist allerdings kein Schutz vor Zugriffen: Spricht man die Spalte explizit an, kann man normal damit arbeiten (siehe [Listing 5](#)).

Übrigens: Intern kennt Oracle die unsichtbaren Spalten schon lange – sobald Objekt-Typen zum Einsatz kommen, werden die Tabellen mit unsichtbaren Spalten versehen – an der Spalte „INVISIBLE“ in der Dictionary-View „USER_TAB_COLS“ lässt sich das leicht nachvollziehen.

```
create table meine_tabelle(
  id      number(10)
         generated always as identity start with 1 increment by 1,
  text    varchar2(10000)
);
```

Listing 3

```
SQL> alter table EMP modify (COMM invisible, HIREDATE invisible);
```

```
SQL> desc EMP
```

| Name | Null? | Typ |
|--------|----------|--------------|
| EMPNO | NOT NULL | NUMBER(4) |
| ENAME | | VARCHAR2(10) |
| JOB | | VARCHAR2(9) |
| MGR | | NUMBER(4) |
| SAL | | NUMBER(7,2) |
| DEPTNO | | NUMBER(2) |

Listing 4

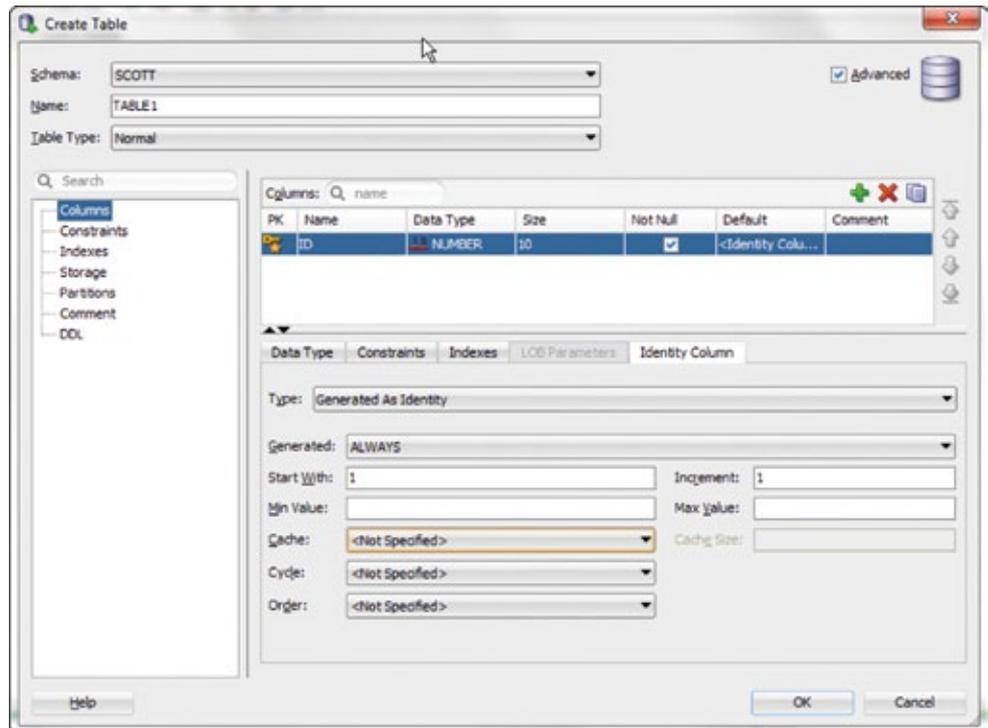


Abbildung 1: Oracle12c Identity Columns im SQL Developer

Weitere Verbesserungen in SQL und PL/SQL

Zwei neue SQL-Klauseln für Abfragen machen das Blättern in einem Bericht wesentlich einfacher: „LIMIT“ sorgt dafür, dass nur eine bestimmte Anzahl

von Zeilen abgerufen wird, „OFFSET“ bewirkt, dass vorher eine bestimmte Anzahl übersprungen wird. Was man bislang aufwändig mit Unterabfragen und „ROWNUM“ gemacht hat, ist nun wesentlich einfacher und klarer. Die Abfrage in [Listing 6](#) selektiert demnach die Zeilen aus der Tabelle „EMP“ mit dem viert-, fünft- und sechsthöchsten Gehalt.

In der WITH-Klausel einer SQL-Abfrage können nun Funktionen definiert werden; diese gelten nur für diese Abfrage – es gibt keine Einträge im Data Dictionary (siehe [Listing 7](#)).

Die „ACCESSIBLE BY“-Klausel legt fest, dass ein PL/SQL-Package, eine PL/SQL-Prozedur oder -Funktion nur von einem (oder mehreren) anderen PL/SQL-Objekt(en) – und nicht direkt – aufgerufen werden kann. Das ist besonders nützlich für „Helper“-Packages, die nicht zum direkten Aufruf und nur zur Nutzung durch andere Packages vorgesehen sind. Die neue Klausel verhindert das versehentliche (oder absichtliche) Aufrufen und dadurch eventuell entstehende Seiteneffekte (siehe [Listing 8](#)).

SQL Pattern Matching

Eine der interessantesten Neuerungen im Bereich der SQL-Funktionen ist das

```
SQL> select ename, hiredate, sal, comm from emp
```

| ENAME | HIREDATE | SAL | COMM |
|-------|---------------------|------|------|
| SMITH | 17.12.1980 00:00:00 | 800 | |
| ALLEN | 20.02.1981 00:00:00 | 1600 | 300 |
| WARD | 22.02.1981 00:00:00 | 1250 | 500 |
| : | : | : | : |

Listing 5

```
select empno, ename, sal
from emp
order by sal asc
offset 3 rows fetch first 3 rows only;
```

| EMPNO | ENAME | SAL |
|-------|--------|------|
| 7521 | WARD | 1250 |
| 7654 | MARTIN | 1250 |
| 7934 | MILLER | 1300 |

Listing 6

```
with
function half_sal(p_sal in number) return number is
begin
return p_sal/2;
end;
select empno, sal, half_sal(sal) from emp;
```

| EMPNO | SAL | HALF_SAL(SAL) |
|-------|------|---------------|
| 7369 | 800 | 400 |
| 7521 | 1250 | 625 |
| 7566 | 2975 | 1487,5 |

Listing 7

SQL Pattern Matching. Dieser Satz an SQL-Funktionen erlaubt es, Muster in Tabellendaten zu finden – und hier geht es um Muster, die sich über mehrere Tabellenzeilen hinweg ergeben. Ein Beispiel für solche Muster wären Web-Sessions, die durch Klicks in einer Log-Datei repräsentiert werden. Die Log-Datei enthält nur die einzelnen Klicks (siehe Listing 9).

Wie man sehen kann, enthält die Log-Tabelle nur Einzelaufnahmen – eben das Abrufen einer Webseite, was auf einem Klick auf einen Link basie-

ren kann. Eine Web-Session wird durch mehrere Zeilen in dieser Tabelle repräsentiert – explizit ist die Information nicht enthalten. Man muss die Sessions aus den Daten rekonstruieren. Dass die „CLIENT_IP“ in einer Session dieselbe sein muss, ist klar. Das Ende der Session ist jedoch nicht direkt aus den Daten ableitbar – für das Beispiel wird daher angenommen, dass die Session endet, wenn die Zeitspanne bis zum nächsten Klick eine bestimmte Dauer (5 Minuten) übersteigt (natürlich vereinfacht dieses Beispiel stark). Die Zeilen, die

zu einer Web-Session gehören, müssen also wie folgt gefunden werden:

- Zunächst nach „CLIENT_IP“ und „ZEITSTEMPEL“ sortieren
- Die allererste Zeile markiert den Beginn der ersten Session
- Wenn die nächste Zeile die gleiche „CLIENT_IP“ enthält und die Differenz der Zeitstempel nicht mehr als fünf Minuten beträgt, dann gehört die nächste Zeile zur gleichen, ansonsten ist sie der Beginn der nächsten Session

Man kann sich nun vorstellen, dass es nicht einfach ist, eine solche Aufgabe mit Standard-SQL zu lösen. Die neue SQL-Funktion „MATCH_RECOGNIZE“ erleichtert die Aufgabe (siehe Listing 10).

In guter alter SQL-Manier wird der Datenbank nicht mehr gesagt, wie die Aufgabe zu erledigen ist, sondern dass ein „Pattern Matching“ durchgeführt werden soll. Die nötigen Angaben erfolgen deklarativ. Als Ergebnis liefert diese SQL-Abfrage eine Übersicht über die gefundenen Sessions zurück (siehe Listing 11).

Java in der Datenbank

Es ist bekannt, dass die Oracle-Datenbank Stored Procedures und Functions nicht nur in PL/SQL ausführen kann, sondern auch in Java. Seit der Datenbank-Version 8i (das war im Jahr 2000) ist eine Java-Engine Teil der Datenbank. Das gilt für alle Datenbank-Editionen ab der Standard-Edition (also nicht XE). Mit Java in der Datenbank lassen sich viele Dinge erreichen, die mit PL/SQL nicht machbar sind:

- E-Mails abrufen
- Die Datenbank als FTP-Client einsetzen
- Dateisystem-Zugriffe und Betriebssystem-Kommandos ausführen

In Oracle 12c wurde die Java-Engine auf Java SE6 aktualisiert. Erstmals ist es auch möglich, selbstständig ein Upgrade der Datenbank-JVM in der gleichen Datenbank durchzuführen. Wer also Java 7 braucht, findet in der Dokumentation „Java Developers‘ Guide“ eine Anleitung zum Upgrade der JVM.

```

create or replace function lowlevel return varchar2
accessible by (toplevel) is
begin
  return 'lowlevel() called.';
end lowlevel;
/

create or replace function toplevel return varchar2 is
begin
  return 'Call via function: ' || lowlevel;
end toplevel;
/

SQL> select toplevel from dual;

TOPLEVEL
-----
Call via function: lowlevel() called.
SQL> select lowlevel from dual;

FEHLER in Zeile 1:
ORA-06552: PL/SQL: Statement ignored
ORA-06553: PLS-904: insufficient privilege to access object
LOWLEVEL

```

Listing 8

```

10.165.244.126 - - [05/May/2010:12:33:02 +0200] "GET / HTTP/1.1" 200 10003
10.165.244.126 - - [05/May/2010:12:33:02 +0200] "GET /spatialde_logo.png HTTP/1.1" 200 2094
10.165.244.126 - - [05/May/2010:12:33:02 +0200] "GET /oraclemaps.png HTTP/1.1" 200 113295
10.165.244.126 - - [05/May/2010:12:34:02 +0200] "GET / HTTP/1.1" 200 10003
:
:
:

```

Listing 9

```

SELECT client_ip, start_tstamp, end_tstamp, sess_clicks
FROM ohs_access_log
MATCH_RECOGNIZE (
  PARTITION BY client_ip
  ORDER BY zeitpkt asc
  MEASURES  STRT.zeitpkt AS start_tstamp,
            LAST(NXT.zeitpkt) AS end_tstamp,
            FINAL COUNT(nxt.zeitpkt) as sess_clicks
  ONE ROW PER MATCH
  AFTER MATCH SKIP PAST LAST ROW
  PATTERN (STRT NXT+)
  DEFINE
    NXT AS NXT.zeitpkt < PREV(NXT.zeitpkt) + INTERVAL '300'
  SECOND
) MR
where sess_clicks > 2
order by client_ip, start_tstamp

```

Listing 10

```

10.165.115.30  26.04.11  11:06:13  26.04.11  12:42:14  166
10.165.115.9  26.04.11  08:11:30  26.04.11  08:11:36  47
:
:
:

```

Listing 11

Fazit

Entwickler können massiv von den neuen Funktionen in Oracle 12c profitieren. Viele sehen auf den ersten Blick zwar recht einfach aus, der Nutzen der „IDENTITY COLUMNS“, der „LIMIT“-beziehungsweise „OFFSET“-Clauses oder des SQL Pattern Matching ist nicht zu unterschätzen: Bislang mussten solche Aufgaben mit mehr oder weniger aufwändigem Coding umgesetzt werden – das bedeutet jedoch nicht nur Mehraufwand beim Entwickeln, sondern auch bei der Wartung oder bei der Übergabe an neue Mitarbeiter im Projekt. Diese müssen den Code schließlich verstehen und ein einfaches „MATCH_RECOGNIZE“ ist leichter zu verstehen als selbstgeschriebener PL/SQL-Code.

Natürlich kann dieser Artikel nur einen kleinen Ausschnitt aus den neuen Funktionen betrachten; umfassendere Informationen stehen auf der Webseite der deutschsprachigen Apex- und

PL/SQL-Community oder in der Oracle-Dokumentation.

Weitere Informationen

1. Deutschsprachige Apex- und PL/SQL-Community: Oracle12c New Features: https://apex.oracle.com/pls/apex/GERMAN_COMMUNITIES.SHOW_TIPP?P_ID=941
2. Oracle12c Dokumentation, New Features Guide: http://docs.oracle.com/cd/E16655_01/server.121/e17906/toc.htm

Carsten Czarski

carsten.czarski@oracle.com<http://sql-plsql-de.blogspot.com>

Hochverfügbarkeit

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

Bereits die Oracle Maximum Availability Architecture (MAA) in der Datenbank 11g R2 lässt im Bereich „Hochverfügbarkeit“ kaum Wünsche offen, um einen 7x24-Stunden-Betrieb mit perfektem Disaster Recovery zu gewährleisten. Hier stellt sich die Frage, wo 12c noch Verbesserungen einbringen kann.

Eine Lücke hatte die Architektur: Es gab keine vollständige Integration der Clients, um mit der Hochverfügbarkeit der Datenbank transparent umgehen zu können. Genau hier ist nun die größte Änderung der Hochverfügbarkeits-Funktionalitäten in 12c zu finden: „Transaction Guard“ und „Application Continuity“ verbessern für Applikationen und Applikationsserver den Umgang mit der Datenbank. Aber auch in den anderen Bereichen wie „RMAN“, „Grid Infrastruktur“ und „Data Guard“ hat sich einiges getan. Dabei können selbst kleine Features großes Potenzial aufweisen. Dazu gehört auch die Möglichkeit, Partitionen und Datendateien online zu verschieben – eine Grundlage für Advanced Data Optimization.

Recovery Manager

Auch wenn kein Real Application Cluster (RAC) oder Data Guard im Einsatz ist, ein Backup der Datenbank ist Pflicht. Dementsprechend gibt es auch den Recovery Manager (RMAN). Die RMAN-Neuerungen sind deshalb für jeden Datenbank-Administrator von besonderem Interesse. Zu den wichtigsten gehört sicherlich die Möglichkeit, dass RMAN mit Oracle 12c nun eine einzelne Tabelle aus dem Backup wiederherstellen kann. Zwar hatte auch schon 11g R2 mithilfe der Flashback-Funktionalität entsprechende Möglichkeiten, Inhalte gelöschter Tabellen oder komplette Tabellen wiederherzustellen, diese basierten aber auf den Flashback-Mechanismen. Sie waren daher entweder von der UNDO-Retention (Flashback Table) oder dem verfügbaren Platz für den Recycle Bin (Flashback Drop) beziehungsweise dem zur Verfügung stehenden Platz für die Flashback-Logs (Flashback Database) abhängig. Waren alle Zeiträume abgelaufen, half nur ein auf-

wändiges Point-in-Time-Recovery der Datenbank (PITR) oder des Tablespace (TSPITR).

Mit dem neuen „RECOVER TABLE“-Kommando ist es nun möglich, einzelne Tabellen direkt aus dem Backup wiederherzustellen, wahlweise direkt zurück in die Datenbank oder zuerst einmal in eine Export-Dump-Datei, um die Tabelle später manuell einzulesen oder in ein anderes Schema zu importieren. Die Vorgehensweise ist dabei ähnlich wie bei einem TSPITR, es ist also eine Auxiliary-Datenbank-Instanz erforderlich, die RMAN aber selbstständig erzeugt, startet und wieder löscht. Listing 1 zeigt das Recover-Table-Kommando, um die Tabelle „EMP“ wiederherzustellen. Das Ergebnis ist eine neue Tabelle „EMP_RECVR“ mit dem Stand von vor vier Tagen.

Die zweite RMAN-Neuerung betrifft die Portabilität von Backups. So lässt sich ein komplettes Backup einer Datenbank etwa unter AIX auf Solaris wiederherstellen. Allerdings muss das Backup vorher für den Transport gekennzeichnet sein. Dies geschieht unter der Angabe „for Transport“ oder

besser noch durch die Angabe der Zielplattform (siehe Listing 2).

Hat die Zielplattform eine andere Byte-Reihenfolge (Endianness), können zwar nicht das gesamte Backup, aber die einzelnen Tablespaces transportiert werden. Andere RMAN-Verbesserungen betreffen die Möglichkeit, Image-Kopien einzelner Datendateien zu parallelisieren, um zum Beispiel mithilfe der Multi-Sektion Definitionen sowie Datenbank-Kopien mit inkrementellen Backups zu aktualisieren. Dazu dient der Befehl „RECOVER DATABASE ... FROM SERVICE“.

Nicht nur diese Änderungen sind insbesondere für Migrations- und Data-Guard-Umgebungen interessant, da nun das Datenkopieren schneller und detaillierter geschehen kann, sondern auch die Tatsache, dass bei einer Klon-Operation in 12c die eigentlichen Prozesse auf dem Ziel ablaufen und nicht mehr auf dem Primärsystem.

Transaction Guard and Application Continuity

Ein Kernproblem in der Entwicklung von Applikationen, die mit dem Aus-

```
RECOVER TABLE SCOTT.EMP
UNTIL TIME 'SYSDATE-4'
AUXILIARY DESTINATION '/tmp/backups'
REMAP TABLE 'SCOTT'. 'EMP': 'EMP_RECVR';
```

Listing 1

```
BACKUP
TO PLATFORM='Linux x86 64-bit'
FORMAT '/tmp/xplat_backups/db_trans_lin.bck'
DATABASE;
```

Listing 2

fall der Datenbank transparent umgehen sollen, ist der sogenannte „ungewisse Transaktionskontext“. Denn stürzt die Datenbank-Session ab, sind alle Informationen über die aktuelle Transaktion, alle Session-Variablen etc. verloren. Zwar konnten schon seit der Datenbank-Version 8i Funktionalitäten wie TAF ein abgebrochenes Select-Kommando weiterführen, bei abgebrochenen DMLs oder sogar beim Ausbleiben der Commit-Bestätigung war die Applikation aber auf sich allein gestellt. Auch die mit 10g eingeführte Fast-Connect-Failover-Technologie (FCF) hatte eher im Fokus, Applikationen mit Connection Pools so schnell wie möglich zu benachrichtigen, als Applikationen bei der Fehlerbehebung der abgebrochenen Statements zu unterstützen.

Diese fehlenden Informationen über den ungewissen Transaktionskontext kann nun die Datenbank 12c liefern. Zusätzlich bietet sie mithilfe von Transaction Guard sogar ein API, um eine ausgebliebene Commit-Bestätigung zu überprüfen. Hierzu führt die Datenbank eine logische Transaktions-ID für die Session der Clients ein, die der Client kennt und jederzeit (auch nach einem Verbindungsverlust) abfragen und überprüfen kann. Dazu muss die Applikation allerdings über einen Service mit der Datenbank verbunden sein, bei dem der sogenannte „Commit Outcome“ protokolliert wird (siehe Listing 3).

Mithilfe des Transaction-Guard-API kann nun jeder Client, egal ob OCI oder Java, den Erfolg der letzten Transaktionen abfragen. Im Java-Bereich geht die Implementation von Transaction Guard sogar noch einen Schritt weiter. Muss unter OCI die Applikation noch selbst den Status über das API abfragen, was eine Applikationsänderung wäre, ist im neuen Universal-Connection-Pool diese Funktionalität durch die sogenannte „Application Continuity Technologie“ (AC) implizit implementiert.

Für diese Funktionalität merkt sich der Connection-Pool die zuletzt ausgeführten Statements inklusive des abschließenden Commits und kann damit im Fehlerfall genau diese Statements wiederholen. Nachdem Application Continuity im Connection-

```
srvctl modify service -database orcl -service GOLD -commit_
outcome TRUE
```

Listing 3

Pool zuvor über Transaction Guard den Transaktionsstatus geprüft hat, startet AC dann die Wiederholung der Statements. Allerdings mit allen Eigenheiten eines Replay: Wurden also in diesem Context andere Aktionen durchgeführt, die nicht im Verantwortungsbereich der Datenbank liegen (etwa ein externer Aufruf einer Prozedur), so würde dieser auch nochmals durchgeführt werden. Deshalb ist AC nicht automatisch transparent und erfordert immer noch einen Test daraufhin, ob die Applikation für AC geeignet ist. Andere Voraussetzungen sind:

- Verwendung von J2EE oder JPA unter Einsatz des Standard-JDBC-API

- Verwendung des Universal-Connection-Pools
- Keine externen Aktionen, die nicht wiederholt werden können/dürfen
- Keine Verwendung von XA

Selbstverständlich steht die Funktionalität in den aktuellen Versionen des aktuellen WebLogic-Servers als Erweiterung von Active Grid Link zur Verfügung.

Global-Data-Services

Einen zentralen Einstiegspunkt für alle Datenbanken innerhalb eines Unternehmens bieten die Global-Data-Services (GDS). Eine ihrer Aufgabe ist es, alle eingehenden Verbindungsanfragen an die optimale Instanz weiterzu-



Abbildung 1: Active Data Guard mit GDS

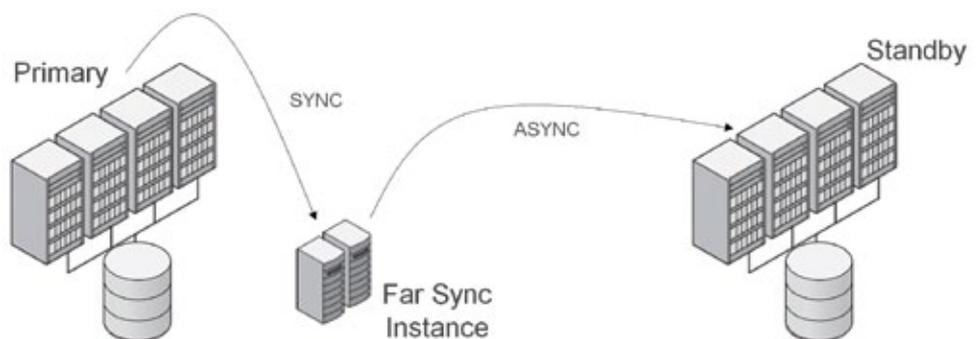


Abbildung 2: Systematische Darstellung von „Far Sync“

leiten und die verfügbaren Datenbank-Services zentral zu verwalten. Das Ziel besteht darin, eine optimale Verteilung zwischen Benutzern und verfügbaren Datenbanken sicherzustellen.

Am besten kann man die positive Auswirkung bei einer Active-Data-Guard- Umgebung (ADG) sehen, in der der Read-Only-Service nur auf der ADG-Seite angeboten wird. Stand die Standby-Datenbank in einer solchen Umgebung unter 11g R2 nicht zur Verfügung, gab es auch keinen Read-Only-Service. Es wäre aber durchaus möglich gewesen, diesen in dem Fall auch auf die Produktiv-Seite oder vielleicht auf eine andere Standby-Datenbank zu verlagern. Genau die Aufgabe erfüllt nun GDS und startet den ausgefallenen Service beispielsweise auf der Primär-Datenbank (siehe **Abbildung 1**).

GDS versorgt also nicht nur die Clients mit Load-Balancing-Informationen, etwa darüber, welche RAC-Instanz, Data Guard oder replizierte Umgebung für den Client die besten Antwortzeiten liefert, sondern kann zusätzlich auch die Services dort starten, wo sie gebraucht werden. Vereinfacht lässt sich die GDS-Technologie mit dem SCAN-Listener in 11g R2 vergleichen, allerdings mit zusätzlichen Funktionalitäten bezüglich unternehmensweiter Service-Verteilung für alle Datenbanken und nicht nur für einen RAC-Cluster.

Data Guard

Der „Zero Data Loss“-Modus von Oracle Data Guard erlaubt es, eine Disaster-Recovery-Umgebung immer auf dem aktuellsten Stand zu halten. Allerdings erfordert dieser Modus, dass die Umge-

bung mit synchronem Redo-Log-Transport arbeitet. Das bedeutet, dass die Primär-Datenbank auf eine Bestätigung der Standby-Umgebung warten muss, bevor sie weiterarbeiten kann. Je größer die Distanz zwischen der Primär-Datenbank und der Standby-Umgebung, desto stärker machen sich die Einbußen bei der Performance bemerkbar.

Oracle 12c Active Data Guard Far Sync behebt dieses Problem, indem es der Primär-Datenbank eine „Far Sync“-Instanz zur Seite stellt. Diese sehr abgespeckte Oracle-Datenbank besteht nur aus folgenden Teilen:

- Standby Control File
- Standby Redo Logs
- Archived Redo Logs

Diese Instanz besitzt keine eigenen Daten-Dateien und dient als eine Art „Redo Log Repeater“, der die Redo-Log-Informationen asynchron an die Standby-Datenbank schickt. Die Primär-Datenbank sendet ihre Redo-Informationen wiederum synchron an diesen Redo Log Repeater (siehe **Abbildung 2**).

Im Fehlerfall erhält die Standby-Datenbank die aktuellen Informationen von der Far-Sync-Instanz und ist somit trotz viel größerer Distanz im „Zero Data Loss“-Modus. Damit ist es auch möglich, ein Netzwerk-Problem zwischen Primär- und Standby-Datenbank abzufedern.

Selbstverständlich kann auch der Standby-Datenbank eine „Far Sync“-Instanz zur Seite gestellt werden, damit Zero Data Loss auch bei umgekehrter Rollenverteilung zur Verfügung steht. Die „Far Sync“-Instanzen erfordern dabei keine Lizenz, nur die Data-

Guard-Umgebung für Active Data Guard muss lizenziert sein.

Im Grunde handelt es sich bei einer „Far Sync“-Standby um eine Art kaskadierendes Standby-System – allerdings bei 12c verbessert, da in 11g R2 die kaskadierende Standby-Datenbank immer auf einen Log-Switch der Primärseite warten musste. Neu im Data-Guard-Umfeld ist auch der folgende Transport-Service „SYNC NOAFFIRM“: „LOG_ARCHIVE_DEST_3=’SERVICE=stdby1 SYNC NOAFFIRM’“. Hier reicht es Data Guard aus, dass die Redo-Log-Informationen im Memory des Standby-Servers angekommen sind. Es wird nicht auf die Bestätigung gewartet, dass die Redo-Log-Informationen auch in die Redo-Log-Files geschrieben wurden.

Erheblich vereinfacht wird auch das Rolling Upgrade mit Active Data Guard und Transient Logical Standby. Ab 12c werden nun auch Datentypen wie „Spatial“, „Image“, „Oracle Text“, „DICOM“, „Varrays“ etc. unterstützt, sodass es nur noch wenige Applikationen geben dürfte, die nicht vom Rolling Upgrade profitieren. Die einzelnen Schritte für ein Rolling Upgrade sind bei Active Data Guard nun auch einfach über das Package „DBMS_ROLLING“ verfügbar. Das vereinfacht die Handhabung des kompletten Prozesses erheblich und schließt viele potenzielle Fehlerquellen aus.

Sonstige Änderungen in Active Data Guard betreffen vor allem die Auslagerung von Read-Only-Services auf Standby-Seite. „Unique Sequences“ in Data-Guard-Umgebungen und die Auslagerung des UNDO von temporären Tabellen in den Temp Tablespace sind dabei von Bedeutung: Dies ermöglicht das Arbeiten mit temporären Tabellen nun auch in einer „Read only“ geöffneten Instanz und bietet somit reinen Reporting-Applikationen einen viel größeren Handlungsfreiraum.

Flex ASM

Im Bereich des Real Application Clusters und der Grid Infrastruktur 12c ist Flex ASM eine der interessantesten Verbesserungen. Es hebt die starre Verknüpfung der Datenbank-Instanzen auf die lokale ASM-Instanz auf und erlaubt somit auch eine Remote-Verbindung

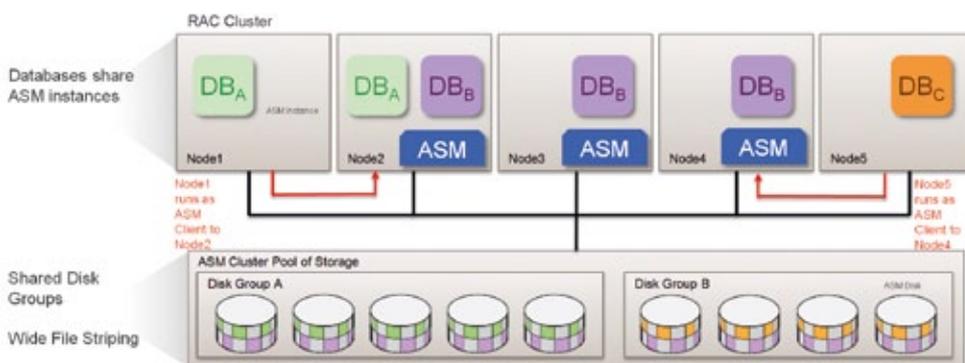


Abbildung 3: Flex ASM

an die ASM-Instanz (siehe [Abbildung 3](#)). Dadurch sind die Datenbank-Instanzen nicht mehr von der lokalen ASM-Instanz abhängig und können auch bei deren Ausfall weiterarbeiten. Nebenbei schont dies bei größeren Clustern die Ressourcen, da nicht mehr auf jedem Knoten eine ASM-Instanz laufen muss.

Oracle Cloud FS und ACFS

Auch auf Seiten der Lizenzierung gibt es interessante Neuerungen. Diese betreffen insbesondere die Lizenzierung und den Datenbank-Support von ACFS. So sind die Basis-Funktionalitäten vom ASM-Cluster-Filesystem und ASM-Dynamic-Volume-Manager nun für alle Dateien kostenfrei verfügbar. Nur bei Verwendung der erweiterten Funktionalitäten wie „Snapshots“, „Replikation“, „Verschlüsselung“, „Security“ und „Auditing“ ist eine Cloud-Filesystem-

Lizenz notwendig. Dies gilt übrigens auch für 11g-R2-ACFS-Installationen.

Darüber hinaus unterstützt 12c-ACFS nun auch Datenbank-Dateien und damit direkt die Oracle-Datenbank. Zwar sollte aufgrund der Performance die Oracle-Datenbank immer noch direkt in ASM liegen, für Test- und Entwicklungssysteme kann dies aber interessant sein, da ACFS eine Snapshot-Funktionalität besitzt und somit das Klonen von Plugable Databases erheblich beschleunigt.

Fazit

Database 12c bietet für jeden Bereich der Hochverfügbarkeit kleine, aber feine Neuerungen. Die größten sind mit Sicherheit „Application Continuity“, um die Anbindung von Applikationen zu verbessern, „Far Sync Standby“ für entfernte Disaster-Recovery-Datenbanken im „Zero Data Loss“-Modus sowie

„online Partition“ und Datendatei-Movement für die Implementation einer echten ILM-Strategie. Aber auch in Standard-Funktionen wie RMAN oder ASM gibt es sinnvolle Erweiterungen, die für jede Datenbank im Unternehmen die Hochverfügbarkeit nochmals steigern können.

Sebastian Solbach
sebastian.solbach@oracle.com





Sicherheit.

EURaix-Rechenzentrum. Und Ihre Daten sind sicher!

Ihre Vorteile:

- Redundante unabhängige Netzanbindung über mehrere Provider
- Leistungsstarke Internetleitungen
- Höchste Betriebssicherheit, 3 unabhängige Rechenzentrumsstandorte
- Betrieb und Betreuung der IT-Infrastruktur rund um die Uhr, 365 Tage im Jahr
- Höchste Verfügbarkeit der gesamten RZ-Infrastruktur
- Videoüberwachung & Einbruchmeldeanlage



Die regio IT ist zertifiziert nach ISO 9001, ISO 20000 und ISO 27001.



Jahre

 regio IT gesellschaft für informationstechnologie mbh
Sitz der Gesellschaft: Aachen · Niederlassung: Gütersloh · www.regioit.de



Security Inside Out

Heinz-Wilhelm Fabry, ORACLE Deutschland B.V. & Co. KG

Die neue Datenbank bietet im Security-Bereich zahlreiche Neuerungen. Diese sind in erster Linie nicht revolutionär, sondern vorwiegend evolutionär. Der Artikel zeigt die Neuerungen im Gesamtkontext der Oracle-Datenbank-Security-Strategie.

Zwei Drittel aller gestohlenen Daten stammen aus Datenbanken. Das ist nicht verwunderlich: Einerseits werden Datenbanken in erster Linie zur Speicherung und Verarbeitung wichtiger Daten eingesetzt. Andererseits verlieren viele Unternehmen aus den Augen, dass Datenbanken, die im Zentrum der EDV stehen, einen erhöhten Schutzbedarf haben. Dieser Schutz ist nicht durch die trügerische Sicherung der Peripherie der EDV-Systeme – etwa durch Firewalls und Virens Scanner – zu erreichen, sondern muss in der Datenbank selbst und in ihrem unmittelbaren Umfeld erfolgen. Dabei sind drei Bereiche strategisch wichtig: Maßnahmen zur Prävention, Detektion und Administration (siehe [Abbildung 1](#)). Datenbank-Features und -Optionen sowie weitere

Produkte wie Enterprise Manager Cloud Control mit seinen Packs sowie Oracle Audit Vault and Database Firewall bieten eine technische Unterstützung für diese Maßnahmen.

Prävention: Verschlüsseln von Daten

Grundsätzlich sind zwei Bereiche zu unterscheiden, die für die Verschlüsselung relevant sind; einerseits die Netzwerk-Verschlüsselung, andererseits die Verschlüsselung von Daten auf Speichermedien. Mit der Database 12c sind alle Maßnahmen zur Verschlüsselung des Netzwerks, also sowohl die Verschlüsselung über SQL*Net als auch diejenige über SSL nicht mehr Bestandteil der Advanced Security Option (ASO), sondern können von jedem Kunden in allen Datenbank-Editionen

(Standard und Enterprise Edition) und in allen Datenbank-Versionen (12.1.0, 11.x.x), die das technisch unterstützen, kostenlos eingesetzt werden. Das gilt übrigens auch für alle Formen der starken Authentifizierung, zum Beispiel über Smartcards, Kerberos-Tickets etc.

Nach wie vor lizenzpflichtig ist der Einsatz von ASO Transparent Data Encryption (TDE). Diese bietet die Verschlüsselung von Tabellen-Spalten und ganzer Tablespaces, ohne dazu Anwendungen verändern zu müssen, die Verschlüsselung von Backups mit RMAN sowie die Verschlüsselung von Dump-Dateien aus Oracle Data Pump.

Die Verschlüsselung von Daten in der Datenbank mit TDE erreicht dabei ein höheres Sicherheitsniveau als die ganzer Festplatten über spezielle Hard- oder Software, da der Zugriff auf die in der Datenbank mit TDE verschlüsselten Daten einen Zugang zur Datenbank und die entsprechenden Objekt-Privilegien erfordert. Dagegen kann jeder, der Zugriff auf das Betriebssystem einer verschlüsselten Festplatte hat, die in der Datenbank gespeicherten Daten über Betriebssystemkommandos auslesen, etwa über den UNIX/Linux-Befehl „strings“.

| PRÄVENTION | DETEKTION | ADMINISTRATION |
|--------------------------------------|--------------------------|---------------------------|
| Verschlüsseln | Aktivitäten überwachen | Privilegien analysieren |
| Redigieren und Maskieren | Auditieren und Berichten | Sensible Daten finden |
| Privilegierte Benutzer kontrollieren | Database Firewall | Konfigurationen verwalten |

Abbildung 1: Security Inside Out – die Oracle-Security-Strategie

```
ADMINISTER KEY MANAGEMENT CREATE KEYSTORE ,,$ORACLE_HOME/net-
work/admin' IDENTIFIED BY einpasswort
```

Listing 1

```
DBMS_REDACT.ADD_POLICY(
  object_schema      => 'scott',
  object_name        => 'emp',
  column_name        => 'empno',
  policy_name        => 'empno_redact',
  function_type       => DBMS_REDACT.PARTIAL,
  function_parameters => '9,2,4',
  expression         => '1=1');
```

Listing 2

Im Bereich von TDE bietet 12c vor allem eine neue Syntax zur Verwaltung der Dateien und Hardware Security Module (HSM), in denen die Schlüssel für die Verschlüsselung gespeichert sind. Diese neue Syntax wurde auch durch die veränderten Anforderungen an ein Oracle-System notwendig, das als Container Database mit mehreren Pluggable Databases aufgesetzt ist. Beispielsweise ist es in solchen Systemen nötig und möglich, Schlüssel aus unterschiedlichen Datenbanken in einem Keystore (früher Wallet) zu verwalten. Listing 1 zeigt ein Beispiel. Neben dieser neuen Syntax wurden weitere Views zum Monitoring und zur Verwaltung der Keystores eingeführt.

Prävention: Redigieren und Maskieren

Unter „Redigieren“ versteht man bei Oracle das Verändern von Daten ausschließlich für die Ausgabe. Die Veränderung kann für unterschiedliche Benutzer unterschiedlich erfolgen. Wesentlich ist, dass dabei die ursprünglichen Daten nicht verändert werden. Sie können sogar im Rahmen aller DML-Befehle (SELECT, INSERT, UPDATE, DELETE) weiterverwendet werden.

Ein bekanntes Beispiel für die Nutzung ist die Ausgabe von Kreditkartennummern: Die ersten Stellen werden durch ein definierbares Zeichen – häufig ein Asterisk (*) – ersetzt. Nur die letzten Stellen zeigen wirkliche Werte an. Ein praktisches Beispiel zur Anzeige der Personalnummer (EMPNO) aus der Ta-

belle EMP verdeutlicht die Vorgehensweise und das Ergebnis (siehe Listing 2).

Die Prozedur „ADD_POLICY“ aus dem Package „DBMS_REDACT“ formuliert eine Regel namens „EMPNO_REDACT“, mit der die Personalnummer zum Teil verändert wird: Alle Ziffern von der zweiten bis zur vierten Stelle

werden durch die Zahl 9 ersetzt. Das geschieht bei jeder Ausgabe („1=1“) und muss nicht mehr in unterschiedlichen Anwendungen neu programmiert werden. Nur bei einer Abfrage durch den Benutzer „SYS“ wird die Regel ignoriert, da dieser über das Privileg „EXEMPT REDACTION POLICY“ verfügt – vergleichbar mit dem aus Virtual Private Database (VPD) und Oracle Label Security (OLS) bekannten Privileg „EXEMPT ACCESS POLICY“. Listing 3 zeigt die Abfrage und das Ergebnis zu diesem Beispiel.

An dem „SELECT“-Befehl wird deutlich, dass die „EMPNO“ im Rahmen der Abfrage durchaus verwendet werden kann. Das Redigieren betrifft ja nur die Ausgabe. Das Ergebnis zeigt dann, dass bei der Ausgabe der „EMPNO“ von der zweiten bis zur vierten Stelle wie gewünscht immer die Zahl 9 statt der tatsächlich vorhandenen Zahl ausgegeben wird. Abschließend noch der wichtige Hinweis, dass das Redigieren nur im Rahmen von ASO zur Verfügung steht.

In der Oracle-Terminologie bedeutet „Maskieren“, dass Produktionsdaten

```
SELECT empno, ename, job, sal, deptno
FROM emp
WHERE empno > 7800;
```

| EMPNO | ENAME | JOB | SAL | DEPTNO |
|-------|--------|-----------|------|--------|
| 7999 | KING | PRESIDENT | 5000 | 10 |
| 7999 | TURNER | SALESMAN | 1500 | 30 |
| 7999 | ADAMS | CLERK | 1100 | 20 |
| 7999 | JAMES | CLERK | 950 | 30 |
| 7999 | FORD | ANALYST | 3000 | 20 |
| 7999 | MILLER | CLERK | 1300 | 10 |

Listing 3

```
CREATE AUDIT POLICY doag2013
PRIVILEGES ALTER ANY TABLE,
           CREATE ANY TABLE,
           DROP ANY TABLE, ...
ACTIONS   ALTER USER,
           CREATE ROLE,
           ALTER ROLE, ...
WHEN ,SYS_CONTEXT
      (('USERENV', 'IP_ADDRESS') <> ('appserver_ip'));
AUDIT doag2013;
```

Listing 4

```

DBMS_PRIVILEGE_CAPTURE.CREATE_CAPTURE(
  name => 'personalanw',
  type => DBMS_PRIVILEGE_CAPTURE.G_CONTEXT,
  condition =>
    'SYS_CONTEXT(''USERENV'', ''MODULE'') = ''PERSANW''');
...
EXEC DBMS_PRIVILEGE_CAPTURE.ENABLE_CAPTURE ('persanw');
...
EXEC DBMS_PRIVILEGE_CAPTURE.DISABLE_CAPTURE ('persanw');
EXEC DBMS_PRIVILEGE_CAPTURE.GENERATE_RESULT ('persanw');

SELECT * FROM dba_used_sysprivs;
SELECT * FROM dba_used_objprivs_path;

```

Listing 5

nachhaltig und in der Regel irreversibel verändert werden. Dabei behalten sie typische Eigenschaften wie Primär-/Fremdschlüssel-Beziehungen oder ihre Verteilungs-Charakteristika. Das leistet schon länger das kostenpflichtige Data-Masking-Pack des Enterprise Manager Cloud Control. Neu ist hier, dass die zu maskierenden Daten nicht zuerst dupliziert werden müssen, sondern in Dateien im Data-Pump-Export-Format geschrieben und dann in Test- oder Entwicklungssysteme importiert werden können. Dieses Feature kann auch kombiniert werden mit der Möglichkeit, nur Teilmengen von Produktionsdaten zu extrahieren („data subsetting“).

Prävention: Privilegierte Benutzer kontrollieren

Security-Experten haben lange bemängelt, dass viele DBA-Aufgaben die Privilegien des Benutzers SYS erforderten. Das neue Datenbank-Release bietet hier jetzt wirksame Alternativen. Es können nämlich eigene Gruppen für die Bereiche Backup, Keystore und Standby Management festgelegt werden. Das geschieht – unter UNIX und Linux – auf die gleiche Weise, in der schon bisher die DBA-Gruppe festgelegt wird. Das heißt, es werden Betriebssystemgruppen für die einzelnen Bereiche eingerichtet und diesen Gruppen Benutzer zugewiesen. Diese können sich dann bei der Datenbank etwa mit „sqlplus / as syskm“ als Keystore-Manager anmelden. Sie haben alle Berechtigungen, die sie für ihre Arbeit mit den Keystores benötigen, aber zum

Beispiel keinerlei Rechte zum Anlegen oder Ändern von Benutzern.

Außerdem gibt es analog zum „SYSDBA“-Privileg noch die Privilegien namens „SYSDG“, „SYSBACKUP“ und „SYSKM“. Diese wirken allerdings nur bei geöffneter Datenbank. Auch das ist vom „SYSDBA“-Privileg bekannt, das allein ja zum Beispiel ein Starten der Datenbank nicht ermöglicht. Dazu ist die Zugehörigkeit zur „DBA“-Gruppe erforderlich.

Auch die Optionen Database Vault (DV) und Label Security (OLS) dienen der Kontrolle privilegierter Benutzer. In der neuen Datenbank-Version sind sie automatisch installiert – wie die übrigen Optionen auch. DV und OLS müssen nur noch aktiviert werden. Das vermeidet Probleme, die durch das nachträgliche Installieren immer wieder auftraten, weil die in den entsprechenden Handbüchern beschriebenen Schritte nicht korrekt durchgeführt wurden. Zudem können nun Datenbanken aus einem Oracle-Home-Directory mit und ohne Database Vault betrieben werden.

Detektion: Aktivitäten überwachen

Im Laufe der Jahre hat sich im Bereich des Auditing ein Wildwuchs an Möglichkeiten und Speicher-Formen entwickelt. Das neue Release führt nun mit dem neuen Feature „unified auditing“ alles auf eine einheitliche Vorgehensweise und einen einheitlichen Speicherort zurück. Das neue Auditing ist bei Neu-Installationen grundsätzlich konfiguriert; bei älteren Datenbanken, die auf die neue Software umgestellt

werden, wird es durch Hinzulinken eines speziellen Codeteils aktiviert.

Alle Audit-Daten – auch die von Database Vault – sind beim „unified auditing“ innerhalb der Datenbank in einer einzigen Tabelle gespeichert. Diese ist im Tablespace „SYSAUX“ abgelegt, lässt sich aber problemlos in ein anderes Tablespace verlagern.

Die Tabelle gehört dem Benutzer „AUDSYS“. Die Rollen „AUDADMIN“ und „AUDVIEWER“ berechtigen zur Einstellung beziehungsweise zum Lesen des Audit-Trails. Der Eigentümer eines Objekts kann das neue Auditing nur dann verwenden, wenn er auch über die Rolle „AUDADMIN“ verfügt.

Das Auditieren mit „unified auditing“ ist unabhängig von Initialisierungsparametern – vergleichbar mit dem bekannten Fine Grained Auditing (FGA). Gesteuert wird es ebenfalls wie beim FGA über Regeln („Policies“) und deren Aktivierung mit dem Befehl „Audit“ (siehe Listing 4).

Das Beispiel listet zunächst eine Reihe von Privilegien und Aktionen auf, die auditiert werden sollen. Im letzten Teil wird eine Bedingung formuliert, die erfüllt sein muss, damit ein Audit-Eintrag erfolgt. Die Steuerung der Einträge ist zwar über die Bedingungen vom FGA bereits bekannt, aber dort funktioniert das nur für DML-Befehle (INSERT, UPDATE, DELETE, SELECT). Im „unified auditing“ funktioniert diese Steuerung dagegen für alle Audit-Möglichkeiten. Das ist natürlich ein deutlicher Fortschritt gegenüber dem bisherigen Standard-Auditing. Außerdem ist das Auditing performanter geworden und unterstützt auch SQL*Loader, Data Pump und RMAN. All diese Punkte sollten dafür sorgen, dass sich „unified auditing“ sehr schnell als Standard-Variante des Auditing durchsetzen wird.

Detektion: Auditieren und Berichten/ Database Firewall

Die Beschreibung der Oracle-Strategie im Bereich „Security“ wäre unvollständig, wenn man nicht auch auf das relativ neue Produkt Oracle Audit Vault and Database Firewall (AVDF) eingehen würde. Diese beiden Komponenten, die das Produkt ausmachen, wurden zunächst separat mit deutlichen

Überschneidungen eingesetzt. Die Audit-Vault-Komponente bietet die Möglichkeit, Audit-Daten unterschiedlicher Oracle- und auch Nicht-Oracle-Datenbanken sowie beliebiger weiterer IT-Komponenten zu konsolidieren. Hinzu kommt die Möglichkeit, Alarme auszulösen und Berichte auf Basis von Application Express sowie Test-Daten über Workflows zu erstellen.

Die Database-Firewall-Komponente schützt Oracle- und eine Reihe von Nicht-Oracle-Datenbanken vor SQL*Injection-Angriffen. Zusätzlich kann sie ein Protokoll aller Zugriffe auf diese Datenbanken erzeugen – allerdings lediglich ein Protokoll der Zugriffe, die über das Netzwerk erfolgen. Das bedeutet, dass lokale Zugriffe anders, etwa bei Oracle-Datenbanken mit „unified auditing“, erfasst werden müssen.

Beide Komponenten, also Audit Vault Server und Database Firewall, werden als Software-Appliance geliefert: Der Kunde stellt die Hardware (X86 64 Bit) und auf dieser wird exklusiv die von Oracle gelieferte Software installiert. Das Ergebnis ist eine Appliance, die nur nach den Vorgaben von Oracle verändert werden darf.

Administration: Privilegien analysieren, sensible Daten finden, Konfigurationen verwalten

Die neueste Version von Database Vault erfasst über das Package „DBMS_PRIVILEGE_CAPTURE“, welche Privilegien ein Anwender oder eine Anwendung tatsächlich nutzt. Dies ist ein bedeutender Schritt auf dem Weg, das Konzept des „least privilege“ durchzusetzen. Das folgende Beispiel erläutert die Vorgehensweise zum Erfassen der Privilegien, die eine fiktive Anwendung zur Personalverwaltung namens „PERSANW“ benötigt.

Zunächst wird mit „CREATE CAPTURE“ das Erfassen der Privilegien definiert, dann mit „ENABLE_CAPTURE“ und „DISABLE_CAPTURE“ das Erfassen gestartet und nach angemessener Zeit, in der mit der Anwendung gearbeitet wird, gestoppt. Schließlich wird das Ergebnis mit „GENERATE_RESULT“ in Hilfstabellen abgelegt. Diese sind als Views aufbereitet und über „SELECT“-Befehle auswertbar. Damit erhält man

Auskunft über die verwendeten Privilegien – hier die System-Privilegien, die in „DBA_USED_SYSPRIVS“ abzufragen sind, – und Informationen darüber, auf welchem Weg, zum Beispiel direkt oder über Rollen, diese Anwendung die vorhandenen Objektprivilegien erhalten hat („DBA_USED_OBJPRIVS_PATH“, siehe Listing 5).

Nachdem sich die Frage nach verwendeten Privilegien also durch den Einsatz eines Package aus dem Lieferumfang von Database Vault klären lässt, bleibt die Frage, wie man schützenswerte Daten findet. Die Frage mag auf Anhieb banal klingen, aber für viele, vor allem große Unternehmen stellt sie sich massiv. Sie lässt sich mit der Funktionalität des „sensitive data finding“ aus den Enterprise-Manager-Cloud-Control-Packs „Data Masking“ und „Test Data Management“ klären. Diese sind zwar kostenpflichtig, allerdings lässt sich „sensitive data finding“ kostenlos nutzen, sofern für die betroffene Datenbank eine Security-Option (ASO, DV oder OLS) lizenziert ist.

Als letzte Facette der Oracle-Security-Strategie sei noch auf ein weiteres kostenpflichtiges Pack des Enterprise Manager Cloud Control hingewiesen, das Oracle-Database-Lifecycle-Management-Pack. Es ermöglicht, Konfigurationen von Oracle-Datenbanken zu überwachen, zum Beispiel im Hinblick auf die dort verwendeten Initialisierungs-Parameter zur Sicherheit der Datenbank. Sobald diese Parameter verändert werden, signalisiert das Pack die Veränderung und der (Security)-DBA kann darauf reagieren. Dies unterbindet den sogenannten „configuration drift“, also das Abweichen von individuell festzulegenden Normen im Bereich der Datenbank-Konfiguration.

Heinz-Wilhelm Fabry
heinz-wilhelm.fabry@oracle.com



PROMATIS Appliances

Prozessoptimierung & Simulation

Oracle Applications

Oracle BI Suite

Usability

Enterprise 2.0

Enterprise Content Management

Accelerate-Mittelstandslösungen

Fusion Applications

Business Intelligence Applications

Managed Services

Oracle Infrastruktur

Oracle E-Business Suite

Oracle BPM Suite

Application Integration Architecture

Social BPM

Oracle CRM On Demand

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 fast 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

Management der Oracle Datenbank 12c

Ralf Durben, ORACLE Deutschland B.V. & Co. KG

Die neue Version 12c der Oracle Datenbank bringt nicht nur viele neue hilfreiche Features, sondern muss auch verwaltet werden. Dazu steht neben den bekannten Tools jetzt mit Enterprise Manager Database Express auch ein neues Mitglied aus der Enterprise-Manager-Familie zur Verfügung.

Enterprise Manager (EM) ist eine Sammlung von Tools zur Verwaltung von Oracle-Produkten. Im Zentrum steht Cloud Control 12c als zentrale Verwaltungskonsole. Der modulare Aufbau wird dazu genutzt, bei neuen Produkten oder Produkt-Versionen eine rasche Verwaltungsunterstützung anzubieten. Das gilt auch und gerade für die neue Database 12c. Seit Februar, und damit schon vor Erscheinen der neuen Datenbank, beinhaltet das Modul zur Verwaltung einer Oracle-Datenbank die Unterstützung von prominenten Features wie Multitenant, Data Redaction, neuem Auditing, neuen Resource-Manager-Plänen und vielem mehr. Damit sind diese neuen Features seit dem ersten Tag der Verfügbarkeit sofort in Cloud Control 12c nutzbar (siehe Abbildung 1).

Die vollständige Unterstützung aller neuen Features der Oracle Datenbank 12c wird mit den folgenden neuen Versionen des Datenbank-Moduls in Cloud Control zeitnah realisiert. Wie für die Datenbank 10g beziehungsweise 11g gibt es auch in 12c eine Einzel-Datenbank-Verwaltungslösung, die als Web-Anwendung genutzt wird. Diese neue Lösung mit dem Namen „EM Database Express“ löst das alte „EM Database Control“ ab.

EM Database Express

Da bei der Implementierung von EM Database Express auf die in der Datenbank beinhaltete XDB zurückgegriffen wird, ist im Gegensatz zum alten EM Database Control kein OC4J-Container mehr notwendig. Dies verschlankt nicht nur die Installation der Datenbank, sondern macht auch die Web-An-

wendung effizienter und schneller. EM Database Express wird mit jeder Oracle Datenbank 12c installiert und bei Verwendung des Database Configuration Assistant auch konfiguriert. Sollte dies für eine Datenbank mal nicht funktionieren, so muss nur ein Port für die XDB geöffnet werden. EM Database Express gibt schnell einen Überblick über die Datenbank und ermöglicht es, Basis-Verwaltungsaufgaben durchzuführen. Es gibt folgende Sektionen:

- Überblick
- Configuration
- Storage
- Security
- Performance

Die Überblicks-Sektion erscheint mit einem Klick auf den Namen der In-

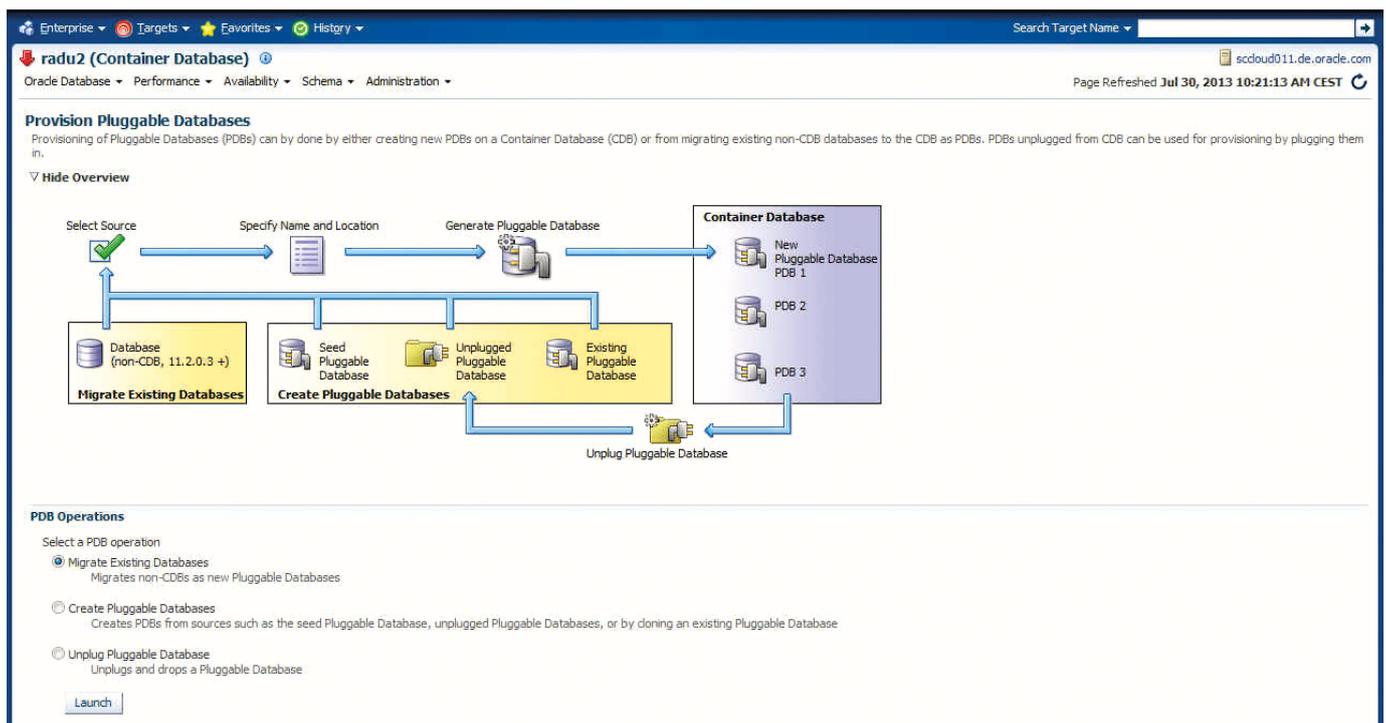


Abbildung 1: Umgang mit Pluggable Databases in Cloud Control

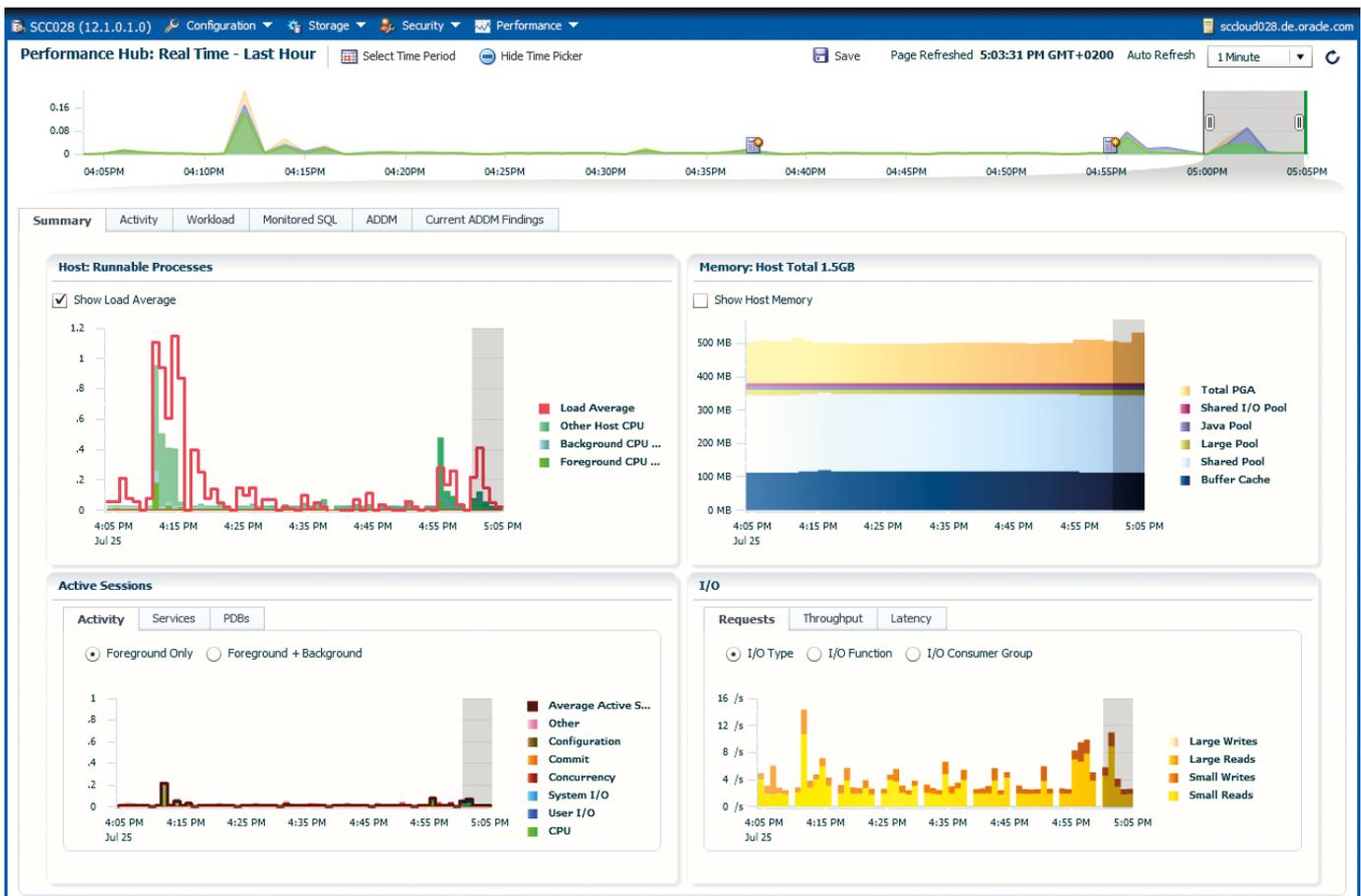


Abbildung 2: Performance Hub in EM Database Express

stanz oben links. Man sieht den Status der Datenbank, einige Angaben zu Performance, Incidents (nur CDB), laufende Jobs und laufende SQL-Kommandos. Im Bereich „Configuration“ kann man je nach Art der Datenbank verschiedene Konfigurationen vornehmen: Instanz-Parameter einstellen, die SGA konfigurieren (nicht bei PDB) und die in der Datenbank genutzten Features einsehen.

Unter „Storage“ lassen sich das Undo Management betreiben, Redo-Log-Einstellungen vornehmen sowie Archive logs, Kontrolldateien und Datendateien verwalten (letztere nicht bei CDBs). Im Bereich „Security“ können Datenbank-Benutzer, -Rollen und -Profile (letztere nicht bei CDBs) eingestellt werden. Der Performance-Bereich, nur nutzbar bei bestehender Lizenz für Diagnostic- beziehungsweise Tuning-Pack, gibt einen Performance-Überblick über die Datenbank; man kann zudem den Tuning Advisor nutzen (siehe Abbildung 2).

Die Aktivierung von EM Database Express ist sehr einfach. Mit der Listener-Control-Utility lässt sich er-

mitteln, welche Ports auf dem Server schon geöffnet sind (siehe Listing 1). Man erkennt jedoch nicht, welcher Port für welche Datenbank verwendet wird. Das wird in der jeweiligen Datenbank abgefragt (siehe Listing 2).

Sollten die Ergebnisse „0“ oder „NULL“ sein, sind die jeweiligen Ports nicht geöffnet. Im obigen Fall gibt es sowohl einen SSL- als auch einen Nicht-SSL-Port. Wenn es einen Port gibt, ist EM Database Express aktiviert und kann im obigen Fall mit „http://hostname:5500/em“ beziehungsweise „https://hostname:5501/em“ aufgerufen werden.

```
lsnrctl status
:
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)
(HOST=scccloud028.de.oracle.com)(PORT=5500))
(Security=
(my_wallet_directory=/opt/oracle/admin/scc028/xdw_wallet))
(Presentation=HTTP)(Session=RAW))
:
```

Listing 1

Man öffnet einen Port (zum Beispiel Port 5506) für den Betrieb mit SSL einfach mit „SQL> exec DBMS_XDB_CONFIG.SETHTTPSPORT(5506);“. Zum Deaktivieren von EM Database Express setzt man einfach den Port auf „NULL“.

Weitere Utilities

Neben Oracle Enterprise Manager gibt es auch die altbekannten Utilities „Data Pump“ und „SQL*Loader“, für die es auch neue Features gibt. So kann man jetzt für verschlüsselte Export-Dateien, die mit einem Passwort ge-

schützt werden sollen, dieses auch interaktiv angeben und nicht wie zuvor nur als Parameter in der Kommandozeile beziehungsweise als Parameter in einer Datei (siehe Listing 3).

Darüber hinaus kann man Views als Tabellen exportieren, wobei nicht wie früher nur die View-Definition exportiert wird, sondern die Definition der View als „SELECT“-Abfrage für einen Daten-Export herangezogen wird. Das Ergebnis ist eine Export-Datei, die bei einem Import die entsprechenden Daten als Tabelle importiert (siehe Listing 4). Mit dem „TRANSFORM“-Parameter lässt sich jetzt auch beim Import eine Komprimierungsmethode wählen (siehe Listing 5).

Gerade bei initialen Imports ist es oft nicht sinnvoll, die dadurch normalerweise anfallenden Redo-Log-Daten generieren zu lassen – vor allem, wenn erst danach eine Backup-Strategie aufgesetzt und die Datenbank produktiv wird. Aus diesem Grund ist jetzt der auch von anderen Kommandos bekannte „NOLOGGING“-Modus für den Import möglich, wobei dieser auf Tabellen oder Indizes eingeschränkt werden kann (siehe Listing 6).

Schon seit langer Zeit gibt es die Möglichkeit, eine Datenbank per Export/Import zu transportieren. Dabei werden per Full-Database-Export alle Nicht-System-Daten in eine Export-Datei ausgelesen. Bei großen Datenbanken wird diese entsprechend umfangreich und der Vorgang dürfte auch seine Zeit benötigen. Dann wird diese große Export-Datei, die auch in kleine Stücke aufgesplittet werden kann, von Server zu Server kopiert. Beim Full-Database-Import werden zunächst eine leere Datenbank mit allen System-Daten erstellt und dann die in der Export-Datei gespeicherten Nicht-System-Daten importiert, zumeist auch ein zeitraubender Vorgang.

Mit der Oracle Datenbank 12c gibt es nun einen neuen Weg für diese Aufgabe: den Full Transportable Export/Import. Dabei werden beim Export keine Daten aus der Quell-Datenbank ausgelesen, sondern nur Metadaten, also die Beschreibung, welche Daten-Dateien zur Datenbank gehören. Die Idee dabei ist, dass nicht eine Export-

```
SQL> select DBMS_XDB_CONFIG.GETHTTPPORT, DBMS_XDB_CONFIG.
GETHTTPSPPORT from dual;
```

```
GETHTTPPORT  GETHTTPSPPORT
-----
           5500             5501
```

Listing 2

```
expdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR schemas=hr
encryption=all encryption_mode=password encryption_pwd_
prompt=y
```

Listing 3

```
expdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR views_as_
tables=[schema].tabelle,...
```

Listing 4

```
impdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR
transform=table_compression_clause:COMPRESS_FOR_OLTP
```

Listing 5

```
impdp hr/hr@ORCL1 dumpfile=hr.dmp directory=WGDGDIR
transform=disable_archive_logging:Y|N[:TABLE|INDEX]
```

Listing 6

Datei mit den Benutzerdaten von Server zu Server kopiert wird, sondern die Daten-Dateien selbst. Dieses Prinzip ist seit geraumer Zeit als „Transportable Tablespace“ bekannt, was jetzt auf die ganze Datenbank ausgeweitet wird. Beim Import werden wieder eine leere Datenbank erzeugt und die kopierten Daten-Dateien dieser neuen Datenbank hinzugefügt. Damit entfällt das zeitraubende Aus- und Einlesen der

Benutzerdaten. Voraussetzung für dieses Verfahren ist, dass alle Tablespace der Quell-Datenbank „READ ONLY“ gesetzt sind.

Auch beim SQL*Loader gibt es Neues zu vermelden: Für einfache Ladevorgänge mit Comma-Separated-Value-Dateien (CSV) benötigt man jetzt keine Steuerungsdatei mehr. Der SQL*Loader ist in der Lage, die CSV-Datei automatisch zu analysieren und zu laden. Vor-

```
SQL> create table wdg1 (nu number,text varchar2(100),datum
date);
```

Tabellendefinition

```
1,Hans,01-jan-13
```

```
2,Peter,04-feb-13
```

Datendatei wdg1.txt

```
sqlldr hr/hr@ORCL1 table=wdg1 data=wdg1.txt
```

Ladekommando

Listing 7

```

REM Welche Patches sind installiert?
SQL> select dbms_qopatch.get_opatch_list from dual;
GET_OPATCH_LIST
-----
<patches/>

REM Das Ergebnis hier: Keine

```

Listing 8

```

REM Ist ein spezieller Patch installiert?
SQL> select dbms_qopatch.is_patch_installed(123456) from
dual;
DBMS_QOPATCH.IS_PATCH_INNSTALLLED(123456)

REM Das Ergebnis ist NULL, also ist der Patch nicht
REM installiert

```

Listing 9

aussetzung dafür sind die Verwendung von Kommas als Trennzeichen in der CSV-Datei sowie die Verwendung skalarer Datentypen (CHAR/VARCHAR, Number, Date), die den Datentypen in der Tabelle entsprechen müssen, also keine Konvertierung erfordern. Ein kleines Beispiel macht dieses neue Feature deutlich (siehe Listing 7).

Neue Management-Features in der Datenbank

Auch ganz ohne Tools gibt es neue Management-Features in der Oracle Datenbank 12c. So können jetzt die Undo-Daten für temporäre Tabellen in eigenen Undo-Segmenten gespeichert werden, die im temporären Tablespace und nicht wie sonst im Undo-Tablespace liegen. Dies hat den großen Vorteil, dass der Umgang mit großen temporären Tabellen nicht mehr unnötigerweise die normalen Undo-Segmente aufbläht, die dann im weiteren Verlauf größer sind als notwendig. Auf diese Weise können auch temporäre Tabellen in einer Active-Data-Guard-Datenbank schreibend verwendet werden, denn sowohl die temporäre Tabelle als auch deren Undo-Daten liegen ja im temporären Tablespace, der im Gegensatz zu den anderen Tablespaces auch in einer sonst „READ ONLY“ geöffneten Active-Data-Guard-Datenbank beschreibbar ist.

Der Administrationsaufwand für dieses Feature ist denkbar gering: Man stellt es einfach mit dem Instanzparameter „temp_undo_enabled“ ein. Das Anlegen erfolgt dann automatisch und mit dem Löschen einer temporären Tabelle wird auch deren Undo-Segment automatisch gelöscht: „SQL> alter system|session set temp_undo_enabled=true|false;“.

Die Verwendung dieser Undo-Segmente ist in „V\$tempundostat“ einsehbar. Auch bei der Steuerung der SGA gibt es Neues: Der in früheren Versionen eingeführte Parameter „PGA_AGGREGATE_TARGET“ gibt eine Zielgröße für die Summe aller PGA-Inhalte an. Diese kann aber in Extrem-Situationen nicht eingehalten werden, wenn die Nutzungsart der Datenbank mehr PGA benötigt als angegeben. Bevor eine Datenbank-Sitzung einen Fehler aufgrund mangelnden PGA-Memorys bekommt, überschreitet die Datenbank das per Instanz-Parameter vorgegebene Ziel. Mit dem neuen Instanz-Parameter „PGA_AGGREGATE_LIMIT“ lässt sich dies verhindern und eine stärkere Kontrolle bezüglich des genutzten Hauptspeichers ausüben, im Ernstfall natürlich zu Lasten von Datenbank-Sitzungen, die dann Memory-Fehler bekommen.

Von vielen DBAs wurde schon lange eine Möglichkeit gewünscht, aus SQL

beziehungsweise PL/SQL heraus einen Blick in das Inventory einer Oracle-Installation zu werfen. Aber auch aus Anwendungssicht ist es sicherlich interessant, Prüfungen hinsichtlich vorhandener Patches vorzunehmen und im Ernstfall Anwendungsmodule zu blockieren. Dieser Zugriff war auch früher schon recht aufwändig möglich; das neue PL/SQL-Package „DBMS_QOPATCH“ zeigt jetzt zum Beispiel sehr einfach an, welche Patches installiert sind (siehe Listing 8). Man kann damit auch gezielt suchen (siehe Listing 9).

Wer das Diagnostics-Pack lizenziert hat, kann sich auf eine neue Variante des Automatischen Datenbank Diagnostic Monitor (ADDM) freuen, die den Namen „Spot ADDM“ trägt. Dabei wird bei speziellen Ereignissen wie zum Beispiel Deadlocks, Hochlast, CPU-Bound etc. automatisch ein Diagnostik-Lauf gestartet, der später jederzeit eingesehen werden kann. Damit ist gewährleistet, dass man Performance-kritische Situationen später genauestens analysieren kann.

Ralf Durben

Ralf.Durben@oracle.com



Kurz gemeldet: Oracle ZFS Storage ZS3

Die neue Speicher-Lösung unterstützt Anwender dabei, entscheidende Geschäftsprozesse schneller abzuwickeln oder Datenbank-Abfragen zu beschleunigen. Sie erreicht Bestleistungen bei verschiedenen Benchmarks und lässt Konkurrenten wie HP, IBM und NetApp hinter sich. Die ZS3-Serie basiert ausschließlich auf Oracle-Storage-Komponenten und ermöglicht somit eine einfachere Bereitstellung von Datenbanken.

Die Datenbank 12c auf Solaris 11.1

Franz Haberhauer, ORACLE Deutschland B.V. & Co. KG

Schon vor der Übernahme von Sun durch Oracle bestanden enge Kontakte zwischen den Entwicklungsgruppen für das Betriebssystem Solaris bei Sun und für die Datenbank bei Oracle. Solaris unterstützt seit Langem betriebssystemseitig gezielt das Lastprofil der Oracle-Datenbank.

Im Jahr 1993 wurde in Solaris 2.2 Intimate Shared Memory (ISM) eingeführt. ISM kann große, im Hauptspeicher residente und durch eine Vielzahl von Prozessen gemeinsam genutzte Speicherbereiche – genau die Anforderungen für die System Global Area (SGA) der Datenbank – besonders effizient verwalten. Seit der Übernahme arbeiten die Teams noch enger zusammen – etwa hinsichtlich eines frühzeitigen uneingeschränkten Austauschs über neue, bahnbrechende Technologien bis hin zum direkten Zugriff auf den aktuell entwickelten Quellcode. Mit der Freigabe der Datenbank 12c und von Solaris 11.1 werden erste Ergebnisse dieser engen Zusammenarbeit nun sichtbar.

Einige dieser Optimierungen sind für den System- wie auch den Datenbank-Administrator transparent. So wurde durch eine Parallelisierung das Anlegen und Auflösen großer Shared-Memory-Bereiche allgemein um den Faktor sechs bis elf beschleunigt und damit auch das Starten und Herunterfahren von Datenbanken [1]. Während das bereits angesprochene ISM nur statisch dimensionierte SGAs unterstützte, war bei der dynamischen Variante DISM die Performance nicht ganz auf

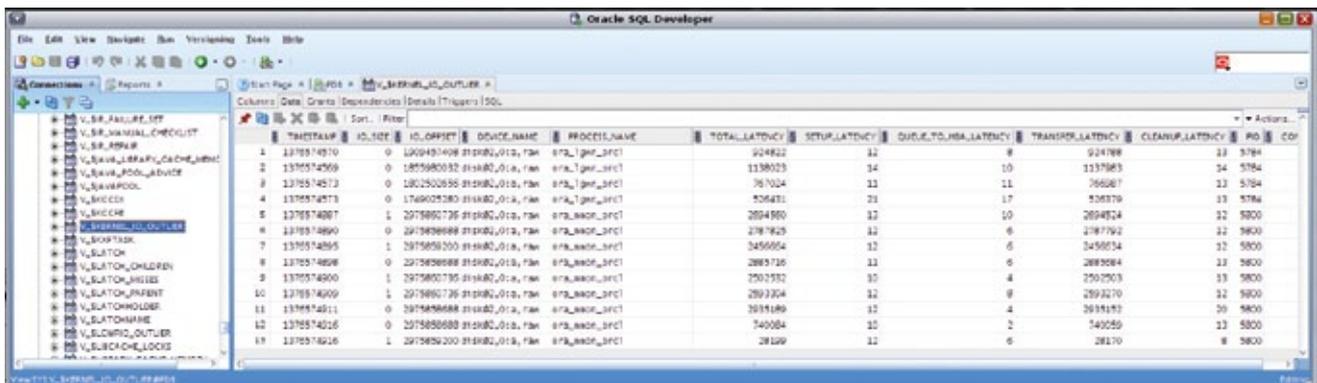
demselben Niveau. Das Automatic Memory Management (AMM) der neuen Datenbank 12c nutzt nun auf Solaris die neue Schnittstelle Optimized Shared Memory (OSM), um die Größe der SGA ohne Zugeständnisse an die Performance online ohne Neustart zu ändern. Damit kann eine Datenbank schnell mit einer kleinen SGA starten und diese nach Bedarf vergrößern.

In Solaris 11.1 wurde zudem eine neue Architektur zur virtuellen Speicherverwaltung eingeführt, die insbesondere die großen Hauptspeicher heutiger Systeme von bis zu mehreren Terabytes (32TB in einer M5-32) wesentlich effizienter verwalten kann und die auf die NUMA-Charakteristika moderner Hauptspeicher-Architekturen abgestimmt ist. Auch hierbei ist eine schnellere Bereitstellung großer Speicherbereiche – etwa für SGAs – und damit ein beschleunigter Systemstart ein unmittelbarer Vorteil.

Durch eine von zehn Millisekunden auf eine Millisekunde verfeinerte Auflösung von Anwendungs-Timern – ermöglicht durch eine effizientere und besser skalierbare Implementierung – wurde die Performance von Anwendungen verbessert, die Polling-Mechanismen nutzen. Davon profitiert auch der Oracle-Log-Writer-Prozess, der bei hoher Last in einen Polling-Modus schaltet [2].

Eine weitere Optimierung in Solaris kommt auf aktuellen SPARC-Servern (T4, T5 und M5) zum Tragen, wo in den Datenbank-Versionen 12c und ab 11.2.0.4.0 (11g R2 Patchset 3) die Log Writer und LMS-Prozesse in Solaris als „Critical Threads“ priorisiert sind. Dabei weist der Solaris-Scheduler diesen Performance-kritischen Threads auf nicht voll ausgelasteten Systemen möglichst ganze Cores exklusiv zu. So müssen sie Ressourcen wie Caches und Ausführungseinheiten nicht mit anderen Threads teilen und können damit die maximal mögliche Hardware-Performance abrufen [3,4].

Neben diesen administrativ transparenten Optimierungen können mit 12c Solaris-Funktionalitäten direkt aus der Datenbank heraus genutzt werden. In Solaris 10 wurde mit DTrace eine sehr mächtige und zugleich leichtgewichtige Performance-Instrumentierung eingeführt. Oracle 12c nutzt auf der Solaris-Plattform DTrace, um nähere Informationen über I/Os auf ASM-verwalteten Geräten zu gewinnen, die sehr lange dauern (mehr als 500 Mil-



| ID | INST | IO_OFFSET | DEVICE_NAME | PROCESS_NAME | TOTAL_LATENCY | SETUP_LATENCY | DISK_IO_LATENCY | TRANSFER_LATENCY | CLEANUP_LATENCY | PO | COF |
|----|------------|-----------|----------------------------|---------------|---------------|---------------|-----------------|------------------|-----------------|------|-----|
| 1 | 1378574210 | 0 | 1209491408 s10x80_01a, raw | ora_lgwr_prc1 | 924822 | 22 | 0 | 924798 | 22 | 5784 | |
| 2 | 1378574209 | 0 | 1805980352 s10x80_01a, raw | ora_lgwr_prc1 | 1338023 | 14 | 10 | 1337983 | 34 | 5784 | |
| 3 | 1378574213 | 0 | 1802502855 s10x80_01a, raw | ora_lgwr_prc1 | 767024 | 23 | 11 | 766987 | 23 | 5784 | |
| 4 | 1378574213 | 0 | 1749005380 s10x80_01a, raw | ora_lgwr_prc1 | 556431 | 23 | 17 | 556376 | 23 | 5784 | |
| 5 | 1378574207 | 1 | 2075860735 s10x80_01a, raw | ora_lgwr_prc1 | 269490 | 22 | 10 | 269452 | 22 | 5800 | |
| 6 | 1378574209 | 0 | 2075860888 s10x80_01a, raw | ora_lgwr_prc1 | 2787925 | 22 | 6 | 2787792 | 22 | 5800 | |
| 7 | 1378574205 | 1 | 2075860300 s10x80_01a, raw | ora_lgwr_prc1 | 3456054 | 22 | 6 | 3456034 | 22 | 5800 | |
| 8 | 1378574208 | 0 | 2075860888 s10x80_01a, raw | ora_lgwr_prc1 | 2895726 | 22 | 6 | 2895684 | 22 | 5800 | |
| 9 | 1378574209 | 1 | 2075860735 s10x80_01a, raw | ora_lgwr_prc1 | 2502352 | 22 | 4 | 2502303 | 22 | 5800 | |
| 10 | 1378574209 | 1 | 2075860735 s10x80_01a, raw | ora_lgwr_prc1 | 2932204 | 22 | 8 | 2932170 | 22 | 5800 | |
| 11 | 1378574211 | 0 | 3075860888 s10x80_01a, raw | ora_lgwr_prc1 | 3031189 | 22 | 4 | 3031183 | 20 | 5800 | |
| 12 | 1378574216 | 0 | 2075860888 s10x80_01a, raw | ora_lgwr_prc1 | 740084 | 22 | 2 | 740059 | 22 | 5800 | |
| 13 | 1378574216 | 1 | 2075860200 s10x80_01a, raw | ora_lgwr_prc1 | 28129 | 22 | 6 | 28170 | 8 | 5800 | |

Abbildung 1: V\$KERNEL_IO_OUTLIER im Oracle SQL Developer

lisekunden). Diese werden in der neuen dynamischen View „V\$KERNEL_IO_OUTLIER“ bereitgestellt, die auf Solaris wesentlich detailliertere Informationen bietet als die Views „V\$IO_OUTLIER“ und „V\$LGWRIO_OUTLIER“, die auf allen Plattformen vorhanden sind.

„V\$KERNEL_IO_OUTLIER“ ist dort als View zwar ebenfalls verfügbar, bleibt aber leer. „V\$KERNEL_IO_OUTLIER“ liefert zusätzlich eine Aufspaltung der Zeit-Anteile, die auf den Datenpfad sowie auf das Gerät selbst entfallen, was im Fehlerfall eine Diagnose signifikant vereinfacht und beschleunigt [5]. **Abbildung 1** zeigt ein Beispiel mit einigen Events, dargestellt mit dem SQL Developer. Es ist deutlich erkennbar, welche Prozesse wann I/O-Probleme hatten und welche Platte daran beteiligt war.

Diese gezielten Optimierungen in Solaris für die Datenbank sowie die Nutzung spezifischer Features in Solaris durch die Datenbank stehen noch am Anfang. Bereits bei der Ankündigung von Solaris 11.1 wurde kurz auf ein weiteres Projekt hingewiesen: Die Verlagerung einiger Funktionen aus dem RAC-Lock-Management in ein Solaris-Kernel-Modul („Kernel Mode RAC Acceleration“), wovon eine Durchsatz-Steigerung um 20 Prozent erwartet wird [6].

Solaris dient einerseits als Betriebssystem auf General-Purpose-Servern direkt als Plattform für die Datenbank – 12c ist für Solaris auf SPARC- sowie auf x86-Servern (und damit für Virtualbox) verfügbar –, andererseits dient Solaris auch als Betriebssystem für die ZFS Storage Appliance (ZFS SA, vormals S7000), auf die als Network Attached Storage (NAS) von der Datenbank aus über dNFS zugegriffen wird.

Auf der ZFS SA war als einzigem NAS-System bereits mit der Datenbank 11g Hybrid Columnar Compression (HCC) nutzbar. Mit der Datenbank 12c wird jetzt das Oracle Intelligent Storage Protocol (OISP) eingeführt, das der Datenbank ein automatisches Performance-Tuning von Datasets auf der ZFS SA ermöglicht. Solaris ZFS verwendet Copy-on-Write-Technologien [7], die bereits mit der Datenbank 11g verwendet werden konnten, um Datenbanken auf dNFS etwa für Testzwecke schnell zu

klonen [8], bei 12c ist das schnelle und platzsparende Klonen von Pluggable Databases von besonderem Interesse.

Fazit

Dieser Artikel zeigt einige der wechselseitigen Optimierungen der Datenbank 12c und des Betriebssystems Solaris. In der SPARC-Prozessor-Roadmap werden bereits einige Funktionen für „Software in Silicon“ skizziert, dem nächsten Schritt des „Engineered to Work Together“.

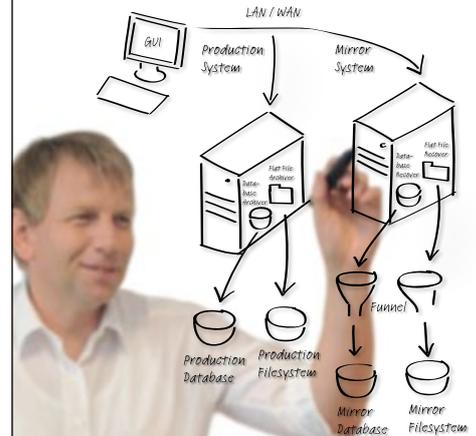
Literatur

- [1] Faster Memory Allocation Using vmtasks: https://blogs.oracle.com/sistare/entry/faster_memory_allocation_using_vmtasks
- [2] High Resolution Timeouts: https://blogs.oracle.com/sistare/entry/high_resolution_timeouts
- [3] Critical Threads Optimization: https://blogs.oracle.com/observatory/entry/critical_threads_optimization
- [4] Reducing High Waits on 'log file sync' on Oracle Solaris SPARC by Increasing Priority of Log Writer (My Oracle Support, Note-ID 1523164.1)
- [5] Oracle Database 12c and Oracle Solaris 11 Dtrace: https://blogs.oracle.com/solaris/entry/database_12c_and_solaris_dtrace
- [6] Oracle Solaris 11.1 – What's New: <http://www.oracle.com/technetwork/server-storage/solaris11-1-what-new-1732377.pdf>
- [7] Franz Haberhauer: Best Practices für Datenbanken auf ZFS, DOAG News Ausgabe 3/2013
- [8] Clone Your dNFS Production Database for Testing (My Oracle Support, Note-ID 1210656.1)

Franz Haberhauer
franz.haberhauer@oracle.com



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

Besuchen Sie uns auf dem
DSAG-Jahreskongress

17. – 19. September 2013
NCC Nürnberg Convention Center
Halle 12, Stand-Nr. B 32

ORACLE Gold Partner



Libelle

Libelle AG

Gewerbestr. 42 • 70565 Stuttgart, Germany
T +49 711 / 78335-0 • F +49 711 / 78335-148
www.Libelle.com • sales@libelle.com



„IDENTITY table clause“-Feature und Apex

Alain Lacour und Arnaud Berbier, dbi-services

Der Artikel beschreibt, wie in Apex Primärschlüssel auf einer „IDENTITY“-Spalte abgebildet werden können, sodass eine harmonische und performante Implementierung mit Oracle 12c gelingt.

Die neue Datenbank bietet für die Tabellenerstellung die neue Klausel „IDENTITY“ für eine numerische Spalte. Diese generiert bei Bedarf für jeden Spalten-Eintrag eine Sequenznummer. Der exakte Bedarf wird mit einem Klausel-Parameter deklariert. Laut Dokumentation gibt es folgende Parameterwerte:

- ALWAYS (Default)
- BY DEFAULT
- BY DEFAULT ON NULL

Je nach Klausel-Parameter werden DML-Operationen unterschiedlich behandelt. Bemerkenswerterweise betrifft diese neue Deklaration der Vorgabewerte teilweise auch „UPDATE-DML“. Dazu ein Beispiel mit der Datenbank-Version 12.1.0.1.0 und Apex 4.2.0.00.27, das bereits in der Datenbank-Software enthalten ist. „ALWAYS“ verhindert, dass ein Wert manuell eingetragen wird. Während der „INSERT“- oder „UPDATE“-Vorgänge darf diese „IDENTITY“-Spalte nicht adressiert werden. Bei Nichtbeachtung wird eine DML-Exception generiert (siehe Listing 1).

```
CREATE TABLE md_country
(
  country_id  NUMBER GENERATED ALWAYS AS IDENTITY,
  country_code CHAR (3 BYTE) NOT NULL,
  country_name VARCHAR2 (200 BYTE) NOT NULL
);
```

Listing 1

```
CREATE TABLE md_country
(
  country_id  NUMBER GENERATED BY DEFAULT AS IDENTITY,
  country_code CHAR (3 BYTE) NOT NULL,
  country_name VARCHAR2 (200 BYTE) NOT NULL
);
```

Listing 2

```
CREATE TABLE md_country
(
  country_id  NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
  country_code CHAR (3 BYTE) NOT NULL,
  country_name VARCHAR2 (200 BYTE) NOT NULL
);
```

Listing 3

„BY DEFAULT“ bietet die Möglichkeit, einen manuellen Wert anzugeben – außer „NULL“. So kann die Spalte während „INSERT“- oder „UPDATE“-Operationen

manipuliert werden, solange ihr neuer Wert nicht „NULL“ ist (siehe Listing 2).

„BY DEFAULT ON NULL“ bietet die Möglichkeit, einen manuellen Wert

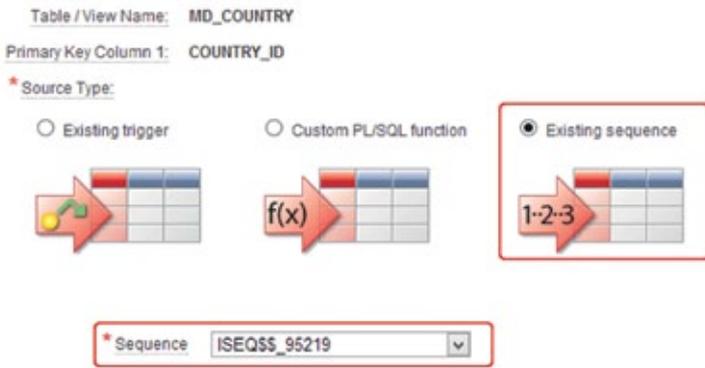


Abbildung 6: Create Page Primary Key



Abbildung 7: Error message

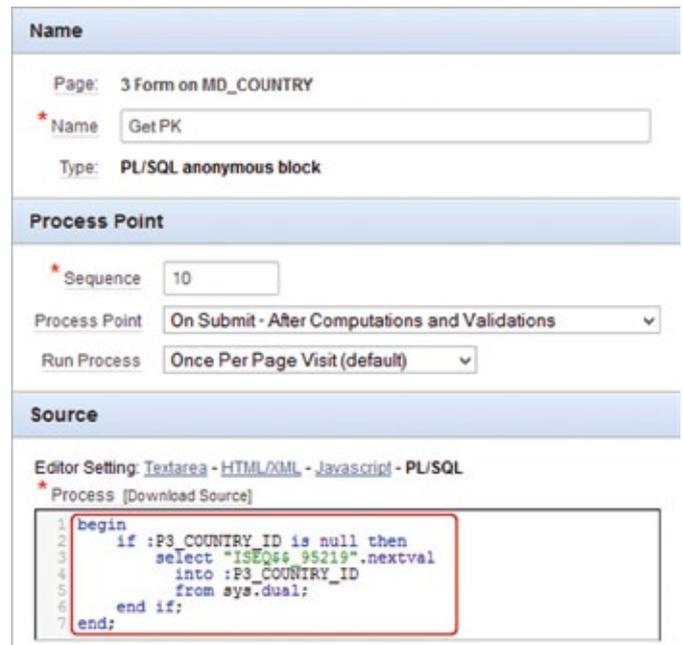


Abbildung 8: Edit Page Process

Hervorzuheben ist die verwirrende Tatsache, dass, obwohl die Sequenz durch die Verwendung einer „IDENTITY“-Klausel entstanden ist, die Tabelle nicht in der Referenzliste erscheint.

Behandlung der „IDENTITY“ als Primary Key in einem Formular

Damit das Zusammenspiel zwischen dem neuen Feature und Apex getestet werden kann, wurde eine Datenbank-Applikation erstellt. Darin verwaltet ein Formular die Daten der „MD_COUNTRY“-Tabelle.

Fall 1, mit „ALWAYS“-Parameter: Bei der Wahl der Primary-Key-Quelle (während der Erstellung des Formulars) wurde manuell „Existing sequence“ gewählt. Auch die Sequenz, die zuvor beim Aufsetzen der Tabelle generiert worden ist, wurde Apex explizit in der Drop-Down-Liste „Sequence“ bekanntgegeben (siehe Abbildung 6).

Beim Anlegen neuer Datensätze mithilfe dieses Formulars kommt es jedoch zu einer Fehlermeldung (siehe Abbildung 7). Der fehlschlagende SQL-Code liegt in der „Get PK“-Prozess-Anweisung, die von Apex für den Formularbereich „Page Processing“ generiert wurde (siehe Abbildung 8).

Man erkennt im Quellcode, dass für die Spalte ein Wert gesetzt wird. Eine „IDENTITY ALWAYS“-Spalte verhindert so etwas bekanntlich und wirft eine DML-Exception. Eine Daten-Än-

derung („update“, „delete“) kann weiterhin ohne Fehlermeldung durchgeführt werden. Wenn man hingegen „Existing Trigger“ statt „Existing sequence“ wählt, werden die Daten auch beim Erstellen richtig behandelt. Die „Get PK“-Prozessanweisung wird bei dieser Variante nicht generiert (siehe Abbildung 9).

Obwohl kein Trigger existiert, behandelt Apex die Tabelle so, als ob es einen gäbe. Somit kümmert sich Apex gar nicht um die Verwaltung der Primary-Key-Spalte und überlässt das der Datenbank. Glücklicherweise stört es Apex nicht, dass es keinen Trigger gibt.

Im Fall 2, mit „BY DEFAULT“-Parameter, macht es keinen Unterschied, ob man „Existing Trigger“ oder „Existing sequence“ während der Erstellung des Formulars wählt, denn hier

ist die Angabe eines Werts für die „IDENTITY“-Spalte erlaubt, solange von „NULL“ verschiedene Werte angegeben sind. So hat die „Get PK“-Prozessanweisung keinen Einfluss auf die korrekte Arbeitsweise der „IDENTITY“-Spalte, solange sie nicht geändert wird, um den „NULL“-Wert zu verwenden.

Im Fall 3, mit „BY DEFAULT ON NULL“-Parameter, ist es ebenfalls nebensächlich, ob man „Existing Trigger“ oder „Existing sequence“ verwendet. Auch hier akzeptiert die „IDENTITY“-Spalte Werte einschließlich „NULL“. Die „Get PK“-Prozessanweisung findet an dieser Stelle keinen Einsatz.

Anmerkung: Die dritte Wahl, „Custom PL/SQL function“, sollte für alle Möglichkeiten und die „IDENTITY“-Klausel-Parameter-Werte sorgfältig erwogen werden, bevor sie aufgesetzt wird.

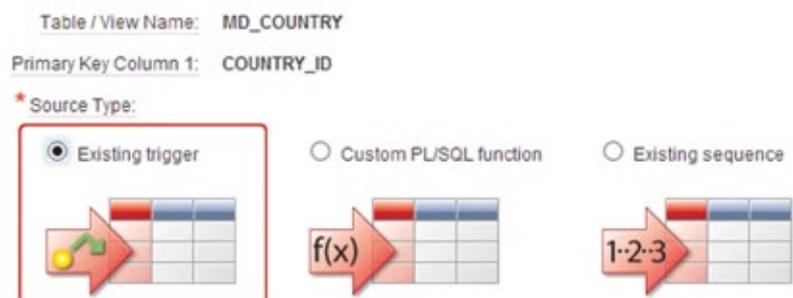


Abbildung 9: Create Page Primary Key

| Anzahl der eingefügten Datensätze | Zeitverbrauch der Identity-Variante (s) | Zeitverbrauch der Trigger-Variante (s) |
|-----------------------------------|---|--|
| 100 | 0.0071 | 0.0232 |
| 1000 | 0.0486 | 0.2043 |
| 10000 | 0.4737 | 2.1988 |
| 100000 | 5.0842 | 21.9390 |
| 1000000 | 53.8582 | 217.648 |

Tabelle 1: Performance-Vergleich

Betrachtung der Performance

Eine Methode für die Handhabung von ID-Spalten ist, dass man ihre Werte mithilfe eines „Before Insert“-Triggers automatisch generieren lässt. Dieser verwendet seinerseits ein Sequenzobjekt. Das wird mit der neuen 12c-IDENTITY-Funktionalität verglichen und hinsichtlich der Performance untersucht. Es wurden Ergebnisse für die vierfache Geschwindigkeit der „IDENTITY“-Generierung im Vergleich zur Trigger-Lösung erzielt, unabhängig von der Anzahl der Datensätze (siehe Tabelle 1).

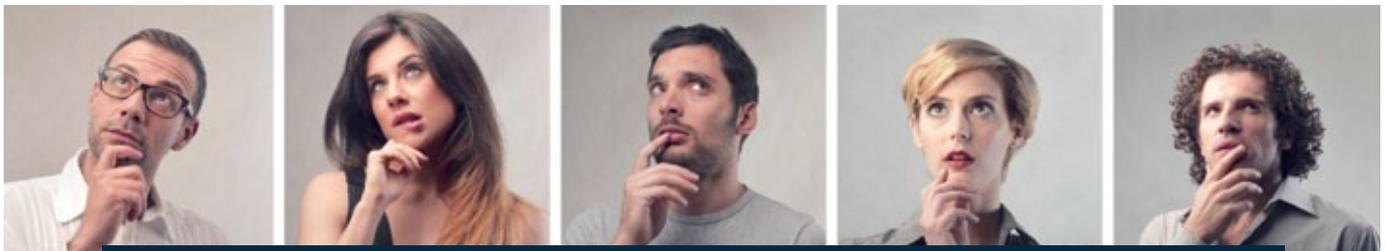
Fazit

Da für die Wahl der Primary-Key-Quelle die „Existing Trigger“-Option in jedem Fall funktioniert, scheint sie die bessere Wahl zu sein, sobald eine „IDENTITY“-Spalte eine Verwendung findet. Hinzu kommen die bemerkenswerten Performance-Vorteile der „IDENTITY“-Spalte, die für ihre Verwendung als ID-Spalte in Oracle 12c sprechen.

Alain Lacour
alain.lacour@dbi-services.com



Arnaud Barbier
arnaud.barbier@dbi-services.com



Was wäre, wenn Sie mehr Zeit für wichtige Projekte hätten?

Wir erledigen **alle DBA Aufgaben** im Oracle Umfeld und gewährleisten rund um die Uhr den reibungslosen Betrieb Ihrer Oracle Datenbanken. Dadurch helfen wir Ihnen **Zeit, Geld und Nerven** zu sparen. Und zwar wesentlich.

Seit über 13 Jahren betreuen wir sehr erfolgreich via Fernwartung zahlreiche Oracle Datenbanken von über fünfzig Unternehmen aus den unterschiedlichsten Branchen.



Die Oracle Experten



Von der kleinsten Datenbank bis zur Exadata, von Oracle 6 bis 12c kennen unsere zertifizierten Spezialisten alle Feinheiten. Wie Sie von unserem Service profitieren können, finden Sie unter http://bit.ly/dbc_dba

Unser **“Oracle Quick Pocket Licence Guide”** enthält eine übersichtliche Zusammenfassung über alle Oracle Datenbank Lizenzmodelle, Faktoren und Metriken sowie eine aktuelle Lizenz Preisliste in Euro.

Zusätzlich erhalten Sie unsere wertvollen **Lizenzierungstipps**, die Sie in dieser Qualität wahrscheinlich nirgendwo bekommen.

Gratis Download:
http://bit.ly/dbc_guide



Einsatz von Partial Indices im Data Warehouse

Reinhard Mense, areto consulting gmbh

Für performante Abfragen in Data-Warehouse-Systemen sind Indizes unverzichtbar. Aufgrund des sehr großen Datenvolumens im Data Warehouse kann deren Erstellen aber aufwändig und zeitraubend sein. Die mit Oracle 12c verfügbaren „Partial Indices“ können die Einführung neuer Indizes in ein Data Warehouse erleichtern.

Partial Indices können nur für partitionierte Tabellen angewendet werden und müssen analog zur Tabelle partitioniert sein. Die Idee dabei ist, den Index nur für einen Teil der Partitionen zu erstellen. Somit kann die benötigte Zeit für den Aufbau oder Rebuild des Index im Gegensatz zur vollständigen Erstellung deutlich reduziert werden. Allerdings ist der Index dann auch nicht beim Zugriff auf allen Partitionen nutzbar. Es lassen sich sowohl „Partial B*Tree“-Indices als auch „Partial Bitmap“-Indices erstellen. Partial Indices können jedoch nicht für Unique Indices erzeugt werden.

Partial Index für Fakten-Tabellen

In DWH-Systemen sind Partial Indices besonders für Fakten-Tabellen geeignet. Diese werden in der Regel partitioniert und meist dient ein Datum als Partitionsschlüssel. Fragen die Benutzer aktuelle Daten häufiger ab, erfolgt der Zugriff somit auch nur auf einige wenige Partitionen. Beim Aufbau eines neuen oder beim Rebuild eines existierenden Index

bringt dieser also den größten Nutzen für die Partitionen, in denen sich die aktuellen Daten befinden. Am Beispiel einer Fakten-Tabelle werden im Folgenden die Erstellung, die Wirkungsweise und der Nutzen eines Partial Index gezeigt.

Um einen Partial Index zu erzeugen, ist die Klausel „INDEXING PARTIAL“ anzugeben (siehe Listing 1). Damit wird der Index als „Partial“ definiert,

```
create bitmap index bx_verkauf_produkat
on fakt_verkauf (produkt_id)
local nologging indexing partial;
```

Listing 1: Erstellen eines Partial Index mit „INDEXING PARTIAL“

```
create table fakt_verkauf
(datum date
, produkt_id number
, filiale_id number
, menge number
, umsatz number)
pctfree 0 nologging
partition by range (datum)
(partition m201301 values less than (to_date ('01.02.2013', 'dd.mm.yyyy')) indexing off
, partition m201302 values less than (to_date ('01.03.2013', 'dd.mm.yyyy')) indexing off
, partition m201303 values less than (to_date ('01.04.2013', 'dd.mm.yyyy')) indexing off
, partition m201304 values less than (to_date ('01.05.2013', 'dd.mm.yyyy')) indexing off
, partition m201305 values less than (to_date ('01.06.2013', 'dd.mm.yyyy')) indexing off
, partition m201306 values less than (to_date ('01.07.2013', 'dd.mm.yyyy')) indexing off
, partition m201307 values less than (to_date ('01.08.2013', 'dd.mm.yyyy')) indexing off
, partition m201308 values less than (to_date ('01.09.2013', 'dd.mm.yyyy')) indexing off
, partition m201309 values less than (to_date ('01.10.2013', 'dd.mm.yyyy')) indexing on
, partition m201310 values less than (to_date ('01.11.2013', 'dd.mm.yyyy')) indexing on
, partition m201311 values less than (to_date ('01.12.2013', 'dd.mm.yyyy')) indexing on
, partition m201312 values less than (to_date ('01.01.2014', 'dd.mm.yyyy')) indexing off
, partition pdefault values less than (maxvalue) indexing off);
```

Listing 2: Definition der Tabellen-Partitionen für den Partial Index

```
create bitmap index bx_verkauf_filiale
  on fakt_verkauf (filiale_id)
  local nologging indexing full;
```

Listing 3: Erstellen eines vollständigen Index mit „INDEXING FULL“

| Partition | Status für Partial Index BX_VERKAUF_PRODUKT | Status für vollständigen Index BX_VERKAUF_FILIALE |
|-----------|---|---|
| M201301 | UNUSABLE | USABLE |
| M201302 | UNUSABLE | USABLE |
| M201303 | UNUSABLE | USABLE |
| M201304 | UNUSABLE | USABLE |
| M201305 | UNUSABLE | USABLE |
| M201306 | UNUSABLE | USABLE |
| M201307 | UNUSABLE | USABLE |
| M201308 | UNUSABLE | USABLE |
| M201309 | USABLE | USABLE |
| M201310 | USABLE | USABLE |
| M201311 | USABLE | USABLE |
| M201312 | UNUSABLE | USABLE |
| PDEFAULT | UNUSABLE | USABLE |

Tabelle 1: Status von Partial Index und vollständigem Index im Vergleich

es wird jedoch noch nicht angegeben, welche Index-Partitionen aufgebaut werden. Dies bestimmen die Klauseln „INDEXING ON“ und „INDEXING OFF“ bei der Definition der Tabellen-Partitionen (siehe Listing 2). Für mit „INDEXING ON“ definierte Partitionen werden die Index-Partitionen aufgebaut, bei „INDEXING OFF“ findet der Aufbau hingegen nicht statt.

Um die Wirkungsweise des Partial Index zu verdeutlichen, wird im hier verwendeten Beispiel noch ein weiterer, diesmal aber vollständiger Index erzeugt. Dazu kann optional die Klausel „INDEXING FULL“ angegeben werden (siehe Listing 3).

Überprüft man nach dem Erstellen der beiden Indizes den Status der Index-Partitionen im Data Dictionary, wird der Unterschied zwischen dem Partial Index und dem vollständigen Index deutlich (siehe Tabelle 1). Während beim vollständigen Index der Status für alle Partitionen „USABLE“ ist, ist das beim Partial Index nur für Partitionen der Fall, die bei der Tabellende-

inition mit „INDEXING ON“ definiert worden sind. Beim Erstellen des Partial Index wurde also nur ein Teil der Index-Partitionen aufgebaut.

Speicherplatz sparen durch Partial Index

„ALTER TABLE“ passt die aufgebauten Partitionen für den Partial Index bei Bedarf an. Ändert man eine Partition von „INDEXING OFF“ auf „INDEXING ON“, wird diese bei allen Partial Indices sofort aufgebaut. Ein expliziter Rebuild der Partition ist für die Partial Indices nicht notwendig. Ebenso lässt sich eine Partition von „INDEXING ON“ auf „INDEXING OFF“ umstellen. In diesem Fall wird die Index-Partition sofort „UNUSABLE“.

Listing 4 zeigt ein Beispiel für die hier verwendete Fakten-Tabelle. Somit kann für diese sehr einfach ein „Moving Window“ für die aufzubauenden Partitionen realisiert werden (siehe Tabelle 2). Im dargestellten Beispiel werden immer nur die drei aktuellsten Partitionen aufgebaut. Durch den Einsatz dieses Verfahrens reduziert sich der Speicherbedarf

```
alter table fakt_verkauf modify partition m201309 indexing off;
alter table fakt_verkauf modify partition m201312 indexing on;
```

Listing 4: Anpassung der Partitionen für Partial Index

| Partition | Status für Partial Index BX_VERKAUF_PRODUKT vor ALTER TABLE | Status für Partial Index BX_VERKAUF_PRODUKT nach ALTER TABLE |
|-----------|---|--|
| M201301 | UNUSABLE | UNUSABLE |
| M201302 | UNUSABLE | UNUSABLE |
| M201303 | UNUSABLE | UNUSABLE |
| M201304 | UNUSABLE | UNUSABLE |
| M201305 | UNUSABLE | UNUSABLE |
| M201306 | UNUSABLE | UNUSABLE |
| M201307 | UNUSABLE | UNUSABLE |
| M201308 | UNUSABLE | UNUSABLE |
| M201309 | USABLE | → UNUSABLE |
| M201310 | USABLE | USABLE |
| M201311 | USABLE | USABLE |
| M201312 | UNUSABLE | → USABLE |
| PDEFAULT | UNUSABLE | UNUSABLE |

Tabelle 2: Status von Partial und Full Index im Vergleich

für den Partial Index im Vergleich zu einem vollständig aufgebauten Index auf ein konstant niedrigeres Niveau.

Schrittweise Einführung eines neuen Index

Bei sehr umfangreichen Fakten-Tabellen kann der vollständige Aufbau eines neuen Index sehr viel Zeit in Anspruch nehmen. Oft steht für die Erstellung eines neuen Index jedoch nur ein begrenztes Zeitfenster zwischen der Ausführung der ETL-Prozesse und dem Zugriff der DWH-Benutzer zur Verfügung, sodass die zur Verfügung stehende Zeit nicht für den vollständigen Aufbau des Index ausreicht.

Ist der neue Index hingegen als Partial Index erzeugt und definiert man zunächst sämtliche Tabellen-Partitionen mit „INDEXING OFF“, lässt sich der Index anschließend partitionsweise aufbauen. Dazu sind die Tabellen-Partitionen schrittweise auf „INDEXING ON“ zu setzen. Somit kann die Einführung des neuen Index auf mehrere Zeitfenster verteilt erfolgen, sodass die begrenzt zur Verfügung stehende Zeit keinen Engpass mehr darstellt. Bei diesem Verfahren sollte man abschließend den Partial Index mit „ALTER INDEX <index_name> INDEXING FULL“ in einen vollständigen Index umdefi-

nieren, damit der Status der Index-Partitionen bei der Einführung weiterer Partial Indices unberührt bleibt.

Schrittweiser Rebuild eines Index

Soll ein bereits existierender vollständiger Index schrittweise aufgebaut werden, kann das zuvor beschriebene Verfahren für die Einführung neuer Indizes ebenfalls angewendet werden. Für den Index „BX_VERKAUF_FILIALE“ aus dem hier verwendeten Beispiel ist dabei folgender Ablauf vorzusehen:

1. Umdefinieren des vollständigen Index in einen Partial Index mit „ALTER INDEX BX_VERKAUF_FILIALE INDEXING PARTIAL“
2. Alle Tabellenpartitionen auf „INDEXING OFF“ setzen
3. Schrittweise alle Tabellen-Partitionen wieder auf „INDEXING ON“ setzen. Dabei findet automatisch der Rebuild der jeweiligen Partitionen des Index „BX_VERKAUF_FILIALE“ statt.
4. Umdefinieren des Partial Index in einen vollständigen Index mit „ALTER INDEX BX_VERKAUF_FILIALE INDEXING FULL“

Mit diesem Verfahren kann der schrittweise Rebuild auch für mehrere Partial Indices gleichzeitig erfolgen.

Fazit

Oracle Database 12c bietet mit der Einführung der Partial Indices mehr Flexibilität beim Aufbau der Indizes. Ihr Einsatz eignet sich besonders für große partitionierte Fakten-Tabellen im Data Warehouse. Sie ermöglichen für die Erstellung neuer und den Rebuild existierender Indizes den schrittweisen Aufbau, sodass auch kürzere Wartungszeitfenster für die Durchführung dieser Aufgaben genutzt werden können. Für Fakten-Tabellen mit besonders umfangreichen historischen Datenmengen können Partial Indices zudem genutzt werden, um den Speicherbedarf zu begrenzen.

Reinhard Mense

reinhard.mense@areto-consulting.de



Replikation großer Datenmengen in einem OLTP-System

Uwe Simon, T-Systems Telekom IT

Bei der Replikation von Daten bietet Oracle mit Read-only-Snapshots, Multimaster-Replikation, Streams oder GoldenGate die passenden Replikationsverfahren für verschiedene Anwendungsfälle an. Eine Replikation mit Read-only-Snapshots ist sehr einfach zu implementieren und für sehr viele Anwendungsfälle gut geeignet. Doch wo sind die Grenzen und wo muss man bei Problemen suchen? Am Beispiel der Replikation in einem großen OLTP-System wird gezeigt, wo diese Grenzen liegen und wie man Probleme identifizieren und lösen kann.

Die Installation einer Replikation mit Read-only-Snapshots, auch „Materialized View“ (MView) genannt, ist recht

einfach und in den Handbüchern gut beschrieben. Sie bietet sich an, wenn man Daten aus einem produktiven

System für reine Lesezugriffe auf einem oder mehreren anderen Systemen zur Verfügung stellen muss. Nachfolgend

```
CREATE MATERIALIZED VIEW LOG ON xxx$ta_xxx WITH PRIMARY KEY;
```

Listing 1

```
CREATE MATERIALIZED VIEW xxx$mv_xxx REFRESH FAST AS
SELECT * FROM xxx$ta_xxx@DBLINK
```

Listing 2

```
CREATE TABLE xxx$mv_yyy (id NUMBER NOT NULL, ...);
CREATE MATERIALIZED VIEW xxx$mv_yyy ON PREBUILT TABLE
REFRESH FAST AS SELECT id, ... FROM xxx$ta_yyy@DBLINK
```

Listing 3

werden nur die „FAST REFRESHABLE MVIEWS“ betrachtet, da sie der einzige Weg zum zeitnahen Replizieren großer Tabellen sind. Die Installation besteht je Tabelle aus zwei Schritten. Zu Beginn wird dabei auf der MView-Site noch zusätzlich ein Database-Link benötigt. Die Installation kurz zusammengefasst:

- Auf der Master-Site den MView-Log anlegen, am besten immer mit „WITH PRIMARY KEY“ (siehe Listing 1)
- Auf der MView-Site die MView anlegen (siehe Listing 2)
- Es ist geschickter, die MViews als Prebuilt Tables anzulegen. Damit kann man bei Datenbank-Kopien die MViews löschen und neu anlegen, ohne dabei die Daten wieder neu laden zu müssen. Zudem kann man dann einfach zusätzliche Spalten anlegen, die über Trigger gefüllt werden, etwa eine Spalte mit dem Zeitpunkt der letzten Replikation (siehe Listing 3)
- Jetzt fehlen nur noch die Jobs, die die MViews konsistent in einer oder mehreren Replikationsgruppen aktualisieren. Der Zuschnitt der Replikationsgruppen und deren Refresh-Intervalle sind typischerweise fachlich getrieben. Dies kann entweder über „DBMS_REPCAT.CREATE_MVIEW_REP_GROUP“ und „DBMS_REFRESH.MAKE“ erfolgen oder, falls die Replikation in Applikationslogik eingebettet werden

soll, innerhalb dieser mit „DBMS_MVIEW.REFRESH (...)“

Hiermit hat man dann sehr schnell eine lauffähige Replikation implementiert, die normalerweise auch über lange Zeit recht problemlos funktioniert. Trotzdem ist es immer gut, wenn man auf Probleme vorbereitet ist. Dazu ein paar technische Daten zur betriebenen Replikationsumgebung:

- Eine Snapshot-Datenbank mit optimierter Partitionierung/Indizierung
- 231 Fast-Refreshable Materialized Views in 15 Replikationsgruppen
- Drei Tabellen werden über Oracle-Streams aktualisiert

```
SELECT owner, mview_name, master_link, build_mode, container_name, last_refresh_type, last_refresh_date
FROM dba_mviews;
```

Listing 4

```
SELECT rowner, rname, j.job, j.failures, j.broken, j.last_date,
       j.this_date, j.next_date
FROM dba_refresh r JOIN dba_jobs j ON(r.job=j.job);
```

Listing 5

```
SELECT job_name, status, actual_start_date, run_duration
FROM dba_scheduler_job_run_details WHERE job_name = 'REPL_JOB';
```

Listing 6

- Sechs Datenbanken, die Massendaten aus Snapshot-Datenbanken abziehen
- Mehrere Systeme, die die neu replizierten Daten lesen
- MViews mit bis zu 256 Partitionen/Subpartitionen
- Die größte MView hat 140 GB, alle in Summe 540 GB
- Das Replikationsvolumen beträgt 100 bis 500 MLOG-Einträge/Sek.
- Eine Replikationsgruppe enthält 50 Prozent der Änderungen
- Besonderheit: Zeitstempel auf MView-Site zur Identifizierung neu replizierter Datensätze

Wenn die Replikation nicht gut läuft

Nachfolgend sind SQL-Statements für die Abfragen aufgelistet. Die meisten Informationen erhält man zwar auch per Mausklick in Grid Control, die SQL-Statements sind aber beispielsweise für die Integration in einem externen Monitoring-Tool. Dass die Replikation nicht wie gewünscht läuft, erkennt man typischerweise daran, dass die Daten auf der MView-Site zu alt sind, also der letzte abgeschlossene Refresh zu lange her ist. Dies zeigt ein Statement (siehe Listing 4). Als Nächstes stellt sich die Frage: „Läuft die Replikation schon/noch oder liefert sie Fehler?“ (siehe Listing 5).

Bei der Replikation ist es immer gut, wenn man die historischen Laufzeiten der Replikationszyklen kennt. Einerseits sieht man dann schnell, dass es

vielleicht ausnahmsweise mal schlechter läuft, andererseits fällt auch frühzeitig auf, wenn sich – etwa bedingt durch steigendes Änderungsvolumen – die Replikationsdauer langsam verlängert. Die Abfrage in Listing 6 zeigt die Laufzeiten in dem Fall, dass die Replikationsjobs mit dem DBMS_SCHEDULER laufen. Falls sie über DBA_Jobs erfolgen, sollte diese Information auf jeden Fall durch entsprechende PL/SQL-Logik in einer eigenen Log-Tabelle gespeichert werden (siehe Abbildung 1).

Wenn die Replikation plötzlich deutlich länger läuft beziehungsweise gar nicht fertig wird, stellt sich die Frage: „Warum dauert es denn gerade jetzt so lange?“ Dafür gibt es zwei Hauptgründe:

- **Fachlich**
Es werden deutlich mehr Daten angelegt/geändert/gelöscht und damit repliziert
- **Technisch**
Die Replikation dauert je Änderung länger

Die Menge der geänderten/angelegten/gelöschten Datensätze kann man bei der MView-Replikation leider nicht direkt erkennen, da es dafür keine Views beziehungsweise Tabellen im Systemkatalog gibt. Mit etwas PL/SQL-Kenntnissen kann man dies zumindest abschätzen. In einem eigenen Replikations-Job wird vor und nach dem eigentlichen „DBMS_MVIEW.RE-

```
SELECT sid, name, value
FROM v$mystat JOIN v$statname USING (statistic#)
WHERE name = ',execute count';
```

Listing 7

```
DBMS_APPLICATION_INFO.SET_MODULE(
    module_name=>'REPL',
    action_name=>'GRUPPE_1');
```

Listing 8

```
SELECT module, action, executions, sql_text
FROM v$sqlarea WHERE module = 'REPL' AND action = 'GRUPPE_1'
```

Listing 9

```
SELECT dmltype$$, COUNT(DISTINCT m_row$$) anzahl
FROM <owner>.mlog$_<tablename>
GROUP BY dmltype$$;
```

Listing 10

FRESH(...)" jeweils eine Abfrage gemacht (siehe Listing 7) und die Differenz dieser Werte berechnet, die man am besten zusammen mit einem Zeitstempel in eine Log-Tabelle schreibt.

Wer es genauer haben will, kann die drei SQL-Statements (INSERT/UPDATE/DELETE) auf der MView-Site aus „V\$SQLAREA“ ermitteln. Das geht am einfachsten, wenn man vor dem eigentlichen Refresh eine Abfrage macht (siehe Listing 8), die nach

dem Refresh wieder zurückgesetzt wird. Die hier übergebenen Werte sind dann in „V\$SQLAREA“ in den Spalten „MODULE“ und „ACTION“ zu finden und können dort leicht abgefragt werden (siehe Listing 9). Dabei ist dann die Differenz über die „executions“ zu bilden. Damit kann man dann das Volumen der Änderungen je Replikations-Zyklus abschätzen und später auswerten beziehungsweise vergleichen. Über diesen Weg findet man

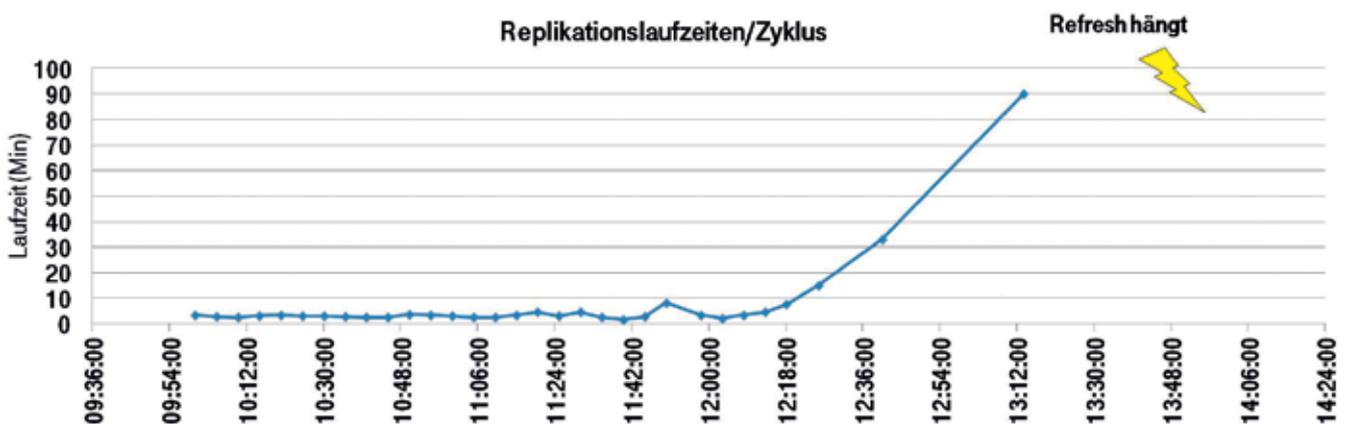


Abbildung 1: Die Replikationsdauer

| | Master-Site | MView-Site |
|-----------------------|--|--|
| Je Replikationsgruppe | | DBMS_REFRESH.REFRESH() |
| | DBMS_SNAPSHOT_UTL.SET_UP() UPDATE MLOG\$_.... COMMIT; | |
| Je MView | SELECT-Zeilen für INSERT/UPDATE | Für jede Zeile UPDATE, wenn nicht vorhanden, INSERT |
| | SELECT der Zeilen für DELETE | Für jede Zeile DELETE |
| Je Replikationsgruppe | DBMS_SNAPSHOT_UTL.WRAP_UP UPDATE MLOG\$_.... DELETE MLOG\$_.... COMMIT; | |

Tabelle 1

| Ursache | Abhilfe |
|--|---|
| Sehr hohe I/O-Zeiten auf MView-Site | Buffer-Cache vergrößern (hilft nur, wenn Blöcke mehr als einmal gelesen werden) I/O beschleunigen (Striping hilft nur begrenzt, da nur ein Prozess schreibt/liest) |
| Replikation verursacht viel I/O auf der Master-Site | Refresh-Intervall zu groß Buffer-Cache zu klein |
| Replikation verursacht viel UNDO auf der Master-Site (Consistent Read) | Replikations-Volumen zu groß Refresh-Intervall zu groß Sehr viele Datenänderungen während eines Replikationszyklus |
| Fachlich | Zu viele Daten werden repliziert (Zeilen und oder Spalten) |

Tabelle 2

in „V\$SQLAREA“ auch leicht die Statements, die für die meiste Laufzeit verantwortlich sind.

Wie viele Daten aktuell repliziert werden müssen, kann man auf der Master-Site über ein Statement erkennen (siehe Listing 10).

Wenn man in den Tabellen auf der Master- und der MView-Site jeweils eine Zeitstempel-Spalte hat, in der auf der Master-Site die Zeit der letzten Änderung (CHANGED_AT) und auf der MView-Site die Zeit der Replikation (REPLICATED_AT) gespeichert werden, kann man relativ einfach die zeitliche Aktualität der MView-Site auf Datensatzbasis ermitteln. Damit dies auf der MView-Site möglich ist, muss man die MViews mittels „PREBUILT“-Table installieren. Dann kann diese zusätzliche Spalte durch einen „INSERT-/UPDATE“-Trigger mit dem „SYSDATE“ gefüllt werden. Wenn das fachliche Volumen keine ausreichende Erklärung für die Replika-

tionsdauer ist, lohnt es sich, die internen Details der Replikation zu betrachten.

Wo die Replikation ihre Zeit verbringt

Um zu verstehen, wo die Replikation Zeit verbraucht, schauen wir zuerst auf den schematischen Ablauf. Alle Aktionen werden beim Refresh dabei von der MView-Site aus angestoßen (siehe Tabelle 1). Nachdem die einzelnen Schritte während eines Replikationszyklus identifiziert sind, stellt sich natürlich die Frage nach deren Laufzeiten.

Im Beispiel von [Abbildung 2](#) fällt auf, dass die Replikation auf der MView-Site hauptsächlich auf I/O wartet. Dies ist auch leicht erklärbar. Auf der Master-Site wurden die Daten vieler Prozesse während der Verarbeitung in den Buffer-Cache geladen und sie bleiben dann auch einige Zeit dort. Daher müssen sie zur Replikation nicht mehr physikalisch gelesen werden. Auf der MView-Site wird jedoch auf diese Daten das erste

Mal während der Replikation zugegriffen, daher gibt es hier die deutlich höheren I/O-Wartezeiten. Wenn die Replikation nicht mehr gut läuft, sieht das Bild etwas anders aus (siehe [Abbildung 3](#)).

Im schlimmsten Fall ist die Replikation auf der Master-Site so lange allein mit dem „UNDO“ beschäftigt, dass pro Sekunde weniger Datensätze durch die Replikation gelesen/verarbeitet werden, als die Applikation an Änderungen produziert. Ab diesem Zeitpunkt wird die Replikation – ohne Zeiten mit deutlich niedrigerem Änderungsvolumen – nicht mehr aufholen können.

Die Replikation beschleunigen

Wenn klar ist, wie die Laufzeit zustande kommt, stellt sich die Frage: „Wie kann man diese Zeiten reduzieren?“ (siehe [Tabelle 2](#)). Falls die aktuell laufende Replikation schon sehr lange überfällig ist, kann es sinnvoll sein, sie zu terminieren und neu zu starten. Damit kann ge-

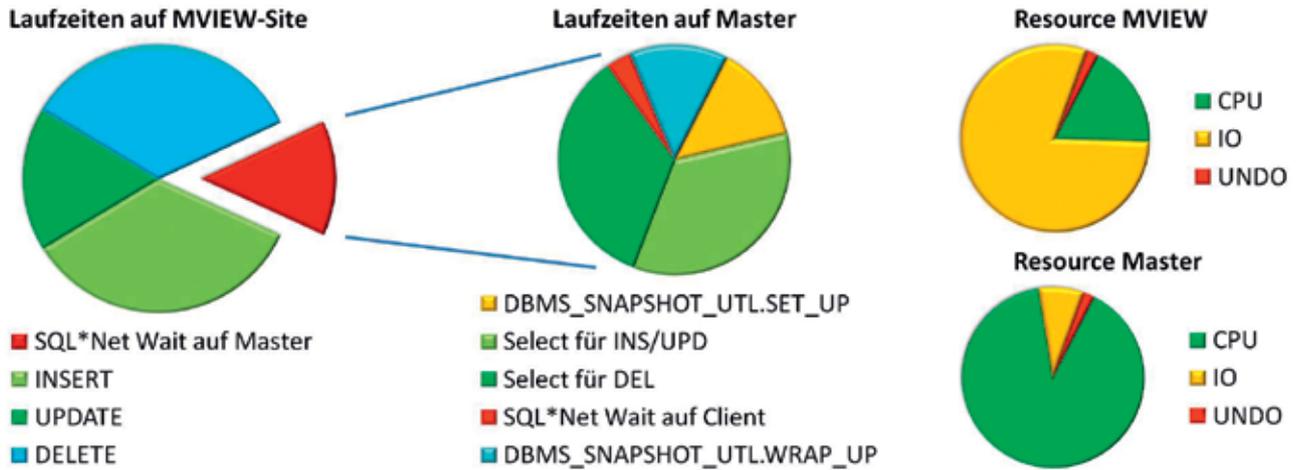


Abbildung 2: Laufzeitverteilung – Replikation läuft gut

gebenfalls wesentlich schneller eine aktuelle MView erreicht werden. Beim Neustart ist zu beachten, dass eventuell wesentlich mehr I/O auf der Master-Site anfällt, da die zu replizierenden Daten nicht mehr im Buffer-Cache liegen.

Zur fachlichen Optimierung stehen folgende Maßnahmen zur Verfügung:

- Nur notwendige Rows/Columns replizieren
- Replikations-Reihenfolge in einer Gruppe von Tabellen, die viel UNDO erzeugen, zu Tabellen, die wenig UNDO erzeugen (reduziert gegebenenfalls Zeiten beim Zugriff auf UNDO für das konsistente Lesen)
- I/O auf MView-Site reduzieren (Buffer-Cache vergrößern)
- Nur notwendige Indizes auf MView-Site
- I/O auf MView-Site beschleunigen (etwa durch Indizes auf Solid State Disk)
- UNDO beschleunigen (UNDO-Tablespace auf Solid State Disk)
- Replikationsgruppen verkleinern
- „Steuern der Änderungsrate auf Master-Site“ in Abhängigkeit der MLOG-Größen (etwa unkritische Batch-Prozesse verlangsamen, damit das Replikations-Volumen pro Zeit immer größer ist als das Änderungs-Volumen pro Zeit)
- Auszeiten der Datenbanken mit MViews so kurz wie möglich halten beziehungsweise mit weiterlaufendem MView-Refresh, sonst werden eventuell alle anderen MViews auf der gleichen Basis-Tabelle deutlich langsamer, da der MLOG wächst
- Regelmäßiges Verkleinern der MView-Logs (nach Migrationen, Auszeiten

etc.) mit „ALTER MATERIALIZED VIEW LOG ON xxx SHRINK SPACE COMPACT“. Dies sollte besonders nach einem starken Anwachsen der MLOGs erfolgen

Security-Aspekte der Replikation

Da die Replikation Datenbank-Links verwendet, gelten die gleichen Security-Anforderungen, wie sie beim sonstigen Einsatz von Datenbank-Links bestehen. Der wichtigste Aspekt dabei ist, dass der Account, mit dem auf die Master-Site zugegriffen wird, nur die unbedingt notwendigen Privilegien hat (auf jeden Fall nicht den Eigentümer der zu replizierenden Tabellen nutzen). Hierbei ist zu beachten, dass die Accounts, die auf der Master-Site für die Replikation angelegt werden, nicht nur Zugriff auf die zu replizierenden Tabel-

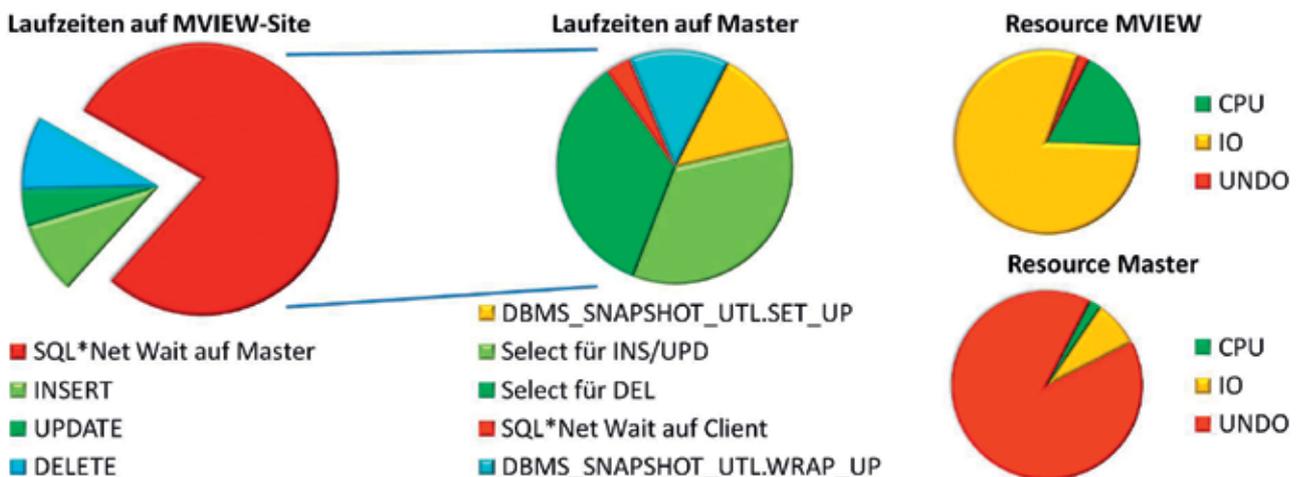


Abbildung 3: Laufzeitverteilung – Replikation läuft schlecht

| Bugs zur Replikation |
|---|
| <p>Mit Oracle 11.2.0.1 ORA-32329 beim Anlegen einer MView Bug 9369183: MVIEW WITH PREBUILT TABLE ON SELECT FROM REMOTE TABLE RETURNS ORA-32349 MView-Name muss unterschiedlich zum Namen der Basistabelle sein. Dafür gibt es Patch 9369183.</p> |
| <p>Master-Site 10.2.0.4, MView-Site 11.2.0.1, Dump während REFRESH Bug 5879082: Dump[kghfrf] during refresh of MView. Dafür gibt es für 10.2.0.4 den Patch 5879082.</p> |
| <p>Es werden keine Refresh mehr ausgeführt. Job-Query-Processes werden nicht gestartet Bug 10103086: wrong result for query with order by and rownum=<constant> Fixed in 11.2.0.3. Workaround Dummy-Job, der kontinuierlich läuft.</p> |
| <p>Refresh liefert ORA-4052: error occurred when looking up remote object. Auslöser ist die Information, dass Tabellen in Master-Site aus Shared-Pool geflogen sind Bug 10210507 : ORA-2019 AFTER ALTER SYSTEM FLUSH SHARED_POOL Fixed in 11.2.0.3, Workaround PUBLIC DATABASE LINK, nicht wirklich nutzbar</p> |
| <p>MViews aus 10.2.0 DBs erscheinen nicht in DBA_REGISTERED_MViews von 11.2.0.1 Bug 9249039 : MVIEW CREATED FROM 10.2 TO 11.2 MASTER IS NOT BEING REGISTERED IN SYS.SLOG\$ Fixed in 11.2.0.2</p> |

Tabelle 3

len, sondern auch auf die MLOGs der Tabellen haben müssen, wenn „FAST-REFRESHABLE MVIEW“ zum Einsatz kommt. Die Replikation macht beim FAST-Refresh ein „SELECT“ mit einem „JOIN“ zwischen Tabelle und zugehörigem „MLOG“. Ferner sollte man auf jeden Fall von „PUBLIC“- Datenbank-Links mit „CONNECT TO“ Abstand nehmen.

Die Erfahrung hat ferner gezeigt, dass es auch für ein Monitoring von Vorteil ist, wenn jedes System einen eigenen Account auf der Master-Site hat. Damit kann man bei Performance-Problemen in „V\$SESSION“ die Replikations-Sessions auf der Master-Site leichter unterscheiden, besonders, wenn mehrere MView-Sites auf einem Server laufen.

Performance-Probleme beim Installieren von MViews

Eine eigene Kategorie von Problemen kann beim Installieren von MViews auftreten. Einige davon findet man leider nicht so einfach auf Testsystemen. Dazu ein kleines Beispiel:

- Auf der Master-Site im laufenden Betrieb, etwa um die Auszeit zu minimieren, eine große neue Tabelle mit „INSERT AS SELECT“ anlegen

- MLOG auf der Tabelle anlegen
- Eine Prebuilt-Table auf der MView-Site anlegen
- MView anlegen
- Complete Refresh starten
- Der Complete Refresh dauert deutlich länger als im Test

Das Problem ist in diesem Fall der „Delayed Block Cleanout“, der sich besonders in Systemen mit vielen Sessions/Transaktionen negativ auswirken kann. Dieser fällt immer dann an, wenn Blöcke mit offenen Transaktionen vom DB-Writer auf Disk zurückgeschrieben werden müssen. Dann wird beim Lesen geprüft, ob der Stand auf Disk noch aktuell ist, weil ein COMMIT ausgeführt wurde, oder ob die Transaktion noch offen ist. Ist die Transaktion abgeschlossen, wird der Block Cleanout durchgeführt und über den DB-Writer auf Platte geschrieben, damit es beim nächsten Zugriff wieder schneller geht. Ob diese Block Cleanouts häufig auftreten, kann man prüfen (siehe Listing 11).

Warum werden die Delayed Block Cleanouts bei der Replikation zum Problem?

Zugriffe über Database-Links führen leider nicht dazu, dass die Blöcke

nach erfolgreichem Cleanout wieder zurückgeschrieben werden. Die Folge ist, dass die Cleanouts mit steigender Laufzeit des SELECT immer mehr Zeit beanspruchen – das Suchen im UNDO dauert dann immer länger – und damit das Lesen der Daten immer langsamer wird. Um den Delayed Block Cleanout während der initialen Replikation zu verhindern, kann vor dem Abzug einfach ein FULLTABLE-Scan auf der zu replizierenden Tabelle gemacht werden (siehe Listing 12). Damit kann man dann den Refresh (Größe der MView im 100-GB-Bereich) deutlich beschleunigen, obwohl vorher noch zusätzlich der FULLSCAN laufen muss. Das hat bis Oracle 10.2 gut funktioniert. Aber auch hier gibt es eine Ausnahme. Ein Select mit Parallel-Query schreibt ebenfalls den Block Cleanout nicht zurück. Hier muss man also gegebenenfalls manuell parallelisieren (etwa n Sessions parallel für n Partitionen).

Mit der Version 11g hat Oracle den FULLTABLE-Scan beschleunigt – was ja eigentlich eine gute Sache ist. Der Scan macht per Default die Disk-Zugriffe statt mit „SCATTERED READ“ nun mit „DIRECT READ“, was den Buffer-Cache umgeht. Dies hat jedoch den Nebeneffekt, dass wiederum kei-

```
SELECT name, value FROM v$sysstat
WHERE name LIKE ,deferred%block cleanout%';
```

Listing 11

```
SELECT /*+ FULL(t) */ count(*) from tab t;
```

Listing 12

```
ALTER SESSION SET EVENTS
'10949 trace name context forever, level 1';
```

Listing 13

```
DBMS_REFRESH.REFRESH('xxx','C',atomic_refresh=>FALSE);
```

Listing 14

ne sauberen Cleanouts gemacht werden. Hierfür biete Oracle einen Workaround (siehe Listing 13).

Ein kleiner Tipp: Bei einem Complete Refresh großer Tabellen, die schon auf der MView-Site gefüllt sind, sollte man, falls fachlich möglich, eine Abfrage machen (siehe Listing 14), dann wird die MView zuerst mit TRUNCATE gelöscht, statt per Default mit „DELETE“. Das beschleunigt den Komplettaufbau der MView deutlich, führt aber dazu, dass auf der MView-Site die MView während des Refresh für alle Sessions erst mal leer ist.

Datenbank-Kopien und Replikation

Beim Benutzen einer Datenbank-Replikation muss man sehr vorsichtig beim Kopieren dieser Datenbank sein. Sowohl Datenbank-Kopien über RMAN als auch solche per Export/Import enthalten alle Replikations-Objekte, Datenbank-Links (inklusive deren Kennworte) etc. Hier muss unbedingt sichergestellt sein, dass die 1:1-Kopie nicht auf die Original-/Produktions-Datenbank zugreifen kann. Folgende Probleme führen sonst im schlimmsten Fall zu Datenverlusten in der produktiven Umgebung:

- MLOG wird durch Refresh auf der falschen Datenbank geleert

- MLOG wächst, weil die 1:1-Kopie noch für den MLOG registriert ist, aber kein Refresh mehr erfolgt

Die sicherste Lösung sind durch Firewalls abgeschottete Umgebungen, bei denen es also keine Zugriffsmöglichkeit von einer Umgebung auf eine andere (wie Test/Entwicklung auf Produktion) gibt. Falls eine Firewall nicht möglich ist, muss die Datenbank-Kopie unbedingt mit abgeschalteten „DBMS_JOBS“ und „DBMS_SCHEDULER“ hochgefahren werden; dann muss man als Erstes alle Datenbank-Links löschen und alle lokalen Kennwörter ändern (was man sowieso tun sollte). Anschließend kann die Datenbank-Kopie gefahrlos gestartet werden. Diese Vorgehensweise sollte man auch bei allen Datenbanken mit Datenbank-Links nutzen.

Bei der Datenbank-Kopie kann man sich die Arbeit deutlich vereinfachen, wenn die Materialized Views über „ON PREBULT TABLE“ angelegt wurden. Dann lassen sich die Materialized Views auch bei geänderten Datenbank-Namen ohne erneute Kopie der Daten anlegen. Die Daten bleiben auch nach dem „DROP MATERIALIZED VIEW“ in der Kopie enthalten. Damit spart man sich besonders bei großen Datenmengen viel Zeit. Dabei ist unbedingt darauf zu achten, dass alle beteiligten Da-

tenbanken transaktionsgenau kopiert werden (wie bei allen Datenbanken, zwischen denen Datenbank-Links genutzt werden), da es andernfalls zu Daten-Inkonsistenzen kommen kann.

Bugs im Umfeld von MView-Replikation

Zu guter Letzt noch ein paar Bugs, über die man beim Einsatz der MView-Replikation (besonders bei größeren Replikations-Umgebungen) stolpern kann (siehe Tabelle 3).

Fazit

Die Read-only-Snapshot-Replikation ist auch in größeren Umgebungen sehr gut nutzbar. Mit ein paar SQL-Abfragen, die man ins Monitoring aufnehmen kann, können alle kritischen Situationen erkannt werden. Man muss aber – wie bei jedem anderen Verfahren auch – die Grenzen der Snapshot-Replikation kennen. Der wesentliche Nachteil ist die fehlende Skalierung, da die Snapshot-Replikation je Replikationsgruppe nur durch einen Prozess erfolgt. Hier muss man aufpassen, dass man die kritische Grenze nicht erreicht oder – noch schlimmer – überschreitet.

Uwe Simon

uwe.simon@t-systems.com



Kurz gemeldet: Oracle WebCenter Portfolio Updates

Die neue Version bringt das Einbinden neuer mobiler Apps, „Bring Your Own Device“-Unterstützung (BYOD) sowie umfangreiche Entwicklungswerkzeuge für mobile Portale und Webseiten.

Mehrsprachigkeit in der Oracle Business Intelligence Enterprise Edition 11g

Dr. Götz Gleitsmann, ORBIT IT-Solutions, und Valentina Speccia, ORBIT IT-Solutions

Dieser Artikel richtet sich an Entwickler und Administratoren im Bereich Oracle BI. Es wird die Version OBIEE 11g vorgestellt. Neben allgemeinen Vorgehensweisen kommen auch typische Fehler und Fallen zur Sprache.

Viele Unternehmen haben neben ihrem Standort in Deutschland weitere Niederlassungen im Ausland. Damit jeder Mitarbeiter die in der Oracle-BI-Anwendung dargestellten Daten verstehen kann, muss er die jeweilige Landessprache einstellen können. Um dies überhaupt zu ermöglichen, müssen zunächst an zwei verschiedenen Stellen die Spracheinstellungen eingetragen beziehungsweise geändert werden. Dieser Prozessschritt wird „Lokalisierung“ genannt. Er beginnt mit der Erzeugung der übersetzten Strings durch einen sprachkundigen Mitarbeiter.

Die Lokalisierung findet in der Oracle-BI-Anwendung an zwei verschiedenen Stellen statt, nämlich im Repository (RPD) und im Web-Katalog. Im Presentation-Layer des RPD sind die Namen der anzuzeigenden Tabellen und der dort enthaltenen Felder lokalisiert, darüber hinausgehende Objekte wie die Namen von Dashboards, Reports, Prompts hingegen im Web-Katalog.

Lokalisierung im RPD

Zunächst sei die grundlegende Funktionsweise erläutert. Für alle Tabellen- und Spaltenbezeichner wird im OBI Repository (RPD) eine Hilfstabelle angelegt. Diese enthält mehrere Spalten, im Einzelnen „SPALTE_BESCHREIBUNG“, „SPALTE_VARIABLEN_NAME“, „LOKALISierter_STRING“, „SPRACH_KUERZEL“, „UEBERSETZUNG_VOLLSTAENDIG“ und „UEBERS_ID“. Bei der Anmeldung eines Benutzers wird zunächst der Initialisierungsblock „SET_LANGUAGE“ ausgeführt, der abhängig von der getroffenen Sprachwahl die entsprechende Session-Variable setzt. Anschließend liest der Session-Varia-

ble-Initialization-Block „GET_L10N_STRINGS“ die Strings aus der Tabelle und macht sie damit verfügbar. In [Abbildung 1](#) ist das Fenster gezeigt, in dem die Initialisierungsblöcke bearbeitet werden können.

Kommen wir nun zur konkreten Vorgehensweise, wie im Presentation-Layer des RPD Spalten- und Tabellennamen lokalisiert sind. Das RPD wird im Online-Modus geöffnet, sodass im Presentation-Layer die verschiedenen Dimensions- und Fakten-Tabellen angezeigt werden. Jeder Bezeichner (Tabellen- oder Feld-Name) verfügt über einen „Benutzerdefinierten Anzeigenamen“ (engl. „Custom display name“), der wie folgt aufgebaut ist: „VALUEOF(NQ_SESSION.CN_NameDerTabelle_NameDerSpalte)“.

Variablen-Namen sollten keine Umlaute oder Sonderzeichen wie „ß“ enthalten. Andernfalls müssten sie in Anführungsstriche gesetzt werden, was jedoch zu Fehlern führen kann. Daher sollte man diese Sonderzeichen in herkömmliche 7-Bit-ASCII-Zeichenfolgen auflösen. Wer über die GUI ([siehe Abbildung 2](#)) einen benutzerdefinierten Anzeigenamen eingetragen hat, sollte anschließend einen weiten Bogen um die zugehörige Checkbox machen – sobald der Benutzer diese anklickt, wird der Anzeigename ohne Rückfrage auf einen von Oracle BI zusammengesetzten String zurückgesetzt.

Für die Werte in der Spalte „SPALTE_BESCHREIBUNG“ der eingangs erwähnten Tabelle sollte der sogenannte „Benutzerdefinierte Anzeigename“ verwendet werden, hiervon allerdings nur die Anteile „<<NameDerTabelle>>“ und „<<NameDerSpalte>>“. So kann man jederzeit sehen, woher die-

ser Spaltenname stammt. Dies ist insbesondere dann hilfreich, wenn in verschiedenen Ordnern der gleiche Spaltenname abgespeichert wird. Die Spalte „LOKALISierter_STRING“ enthält die Übersetzung des ausgewählten Strings und die Spalte „SPRACH_KUERZEL“ das Kürzel der Sprache, die in der Spalte „LOKALISierter_STRING“ eingetragen wurde. Also wird für Deutsch „de“ und für Englisch „en“ eingetragen. Jeder String wird für jede Sprache, in der er später angezeigt werden soll, einzeln aufgelistet. Soll ein String in den Sprachen Deutsch, Englisch und Französisch dargestellt werden, müssen also drei Zeilen angelegt sein ([siehe Tabelle 1](#)). Der Arbeitsaufwand steigt linear mit jeder hinzukommenden Sprache. Zu beachten ist, dass die Kombination aus „SPALTE_VARIABLE_NAME“ und „LOKALISierter_STRING“ eindeutig sein muss, da es ansonsten zu einer Doppelbelegung im Tool kommt.

Leider ist es nicht möglich, dass eine lokalisierte Spalte beim Kopieren innerhalb des gleichen oder in andere Ordner die gleiche Variable behält. Will man hingegen einen ganzen, bereits komplett lokalisierten Präsentationsordner innerhalb des Presentation-Layer kopieren, so behalten die den Spalten zugeordneten Session-Variablen ihren ursprünglichen Namen. Damit vermeidet man die Erzeugung einer unnötig hohen Anzahl an Variablen beziehungsweise Einträgen in der Tabelle.

Besondere Vorsicht ist geboten, wenn in einem Ordner nur ein Teil der Spalten lokalisiert worden ist. Insbesondere muss immer beachtet werden, dass in dem, nach einem Rechtsklick



Abbildung 1: Session-Variable „Initialization Block“, im Administrationstool über Extras à Variablen à Initialisierungsblöcke à GETL10STRINGS zu erreichen

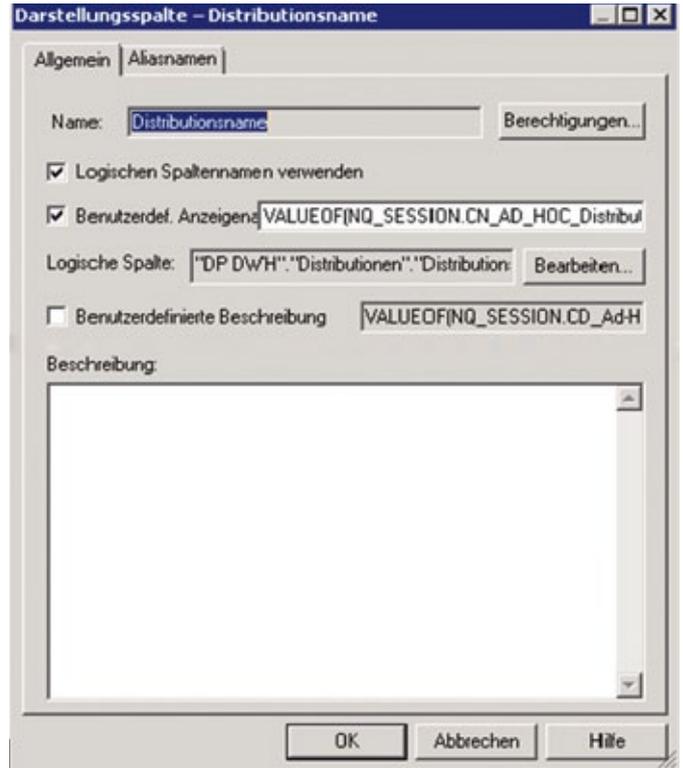


Abbildung 2: Lokalisierung auf Repository-Ebene

auf eine Subject-Area oder Präsentations-Tabelle erscheinenden Menü, das Häkchen „Externalize display names“ beziehungsweise „Externalize descriptions“ nicht gesetzt werden sollte, falls es grau ist. Dann werden nämlich zu allen Spalten Session-Variable erzeugt, auch wenn ihre Lokalisierung gar nicht beabsichtigt war. Passiert dies unbemerkt, werden die entsprechenden Einträge in der Tabelle fehlen. Grundsätzlich sollte daher immer ein ganzer Ordner lokalisiert werden. Auch ist die automatische alphabetische Sortierung im Presentation Layer zu deaktivieren, da sie sich allein auf die Default-Namen der Spalten bezieht.

Lokalisierung im Web-Katalog

Die Lokalisierung im Web-Katalog geschieht im Prinzip dadurch, dass die in XML-Dateien enthaltenen Spaltennamen übersetzt werden und die derart geänderte XML-Datei in einem wei-

teren Betriebssystem-Ordner abgelegt wird. Bei der Bearbeitung des Web-Katalogs ist äußerst sorgfältig und genau vorzugehen, da mitunter neue Strings in die bereits vorhandenen XML-Dateien eingefügt werden müssen. Ist die XML-Datei nach Abschluss der Arbeiten nicht mehr wohlgeformt, so muss der Vorgang wiederholt werden.

Vor Beginn der eigentlichen Lokalisierung wird der Zielordner für die zu exportierenden XML-Dateien beispielweise über den Pfad „ORACLE_INSTANCE\bifoundation\OracleBIPresentationServicesComponent\coreapplication_

obipns\msgdb\l_<<XX>>\captions“ festgelegt.

An Stelle des Platzhalters „<<XX>>“ ist das jeweilige Sprachkürzel derjenigen Sprache einzufügen, die in diesem Ordner abgespeichert werden soll. Zum Beispiel gilt für Englisch „<<XX>> = „en““. Alle weiteren Sprachkürzel finden sich im offiziellen OBI Presentation Service Administration Guide. Für jede Sprache muss also ein eigener Ordner vorhanden sein.

Die eigentliche Lokalisierung beginnt mit einem Export des Web-Katalogs. Für jeden dort enthaltenen Unterordner wird eine eigene XML-Datei erzeugt. Ein Export ist nur möglich, wenn der Web-Katalog im Offline-Modus geöffnet ist.

Für die Spracheinstellungen der Web-Oberfläche wird der Ordner „Ge-

| SPALTE_VARIABLEN_NAME | LOKALISIERTER_STRING | SPRACH_KUERZEL |
|-----------------------|----------------------|----------------|
| SVB_AA_Beschreibung | Beschreibung | de |
| SVB_AA_Beschreibung | description | en |
| SVB_AA_Beschreibung | description | fr |

Tabelle 1: Auszug aus der Übersetzungstabelle

meinsame Ordner“ genutzt. Dort werden die im BI-Tool verwendeten Verzeichnisse in einer Baumstruktur aufgelistet. Nun markiert man den Ordner, dessen Inhalt übersetzt werden soll, klickt auf den Button „Extras“ und wählt „Titel exportieren“ aus.

Es erscheint ein neues Fenster, in dem jetzt ausgewählt werden muss, in welchem Ordner die neu erzeugte XML-Datei gespeichert werden soll. In **Abbildung 3** ist dieses Fenster gezeigt. Es ist ratsam, vorher einen Ordner namens „Captions“ anzulegen, in dem anschließend die verschiedenen XML-Dateien abgespeichert werden können.

Darüber hinaus kann man in dem Fenster auswählen, welche Strings exportiert werden sollen. Hier wird empfohlen, das Häkchen bei „Nur neue oder geänderte Zeichenfolge exportieren“ zu setzen. Beim ersten Export ist dies nicht notwendig, da noch keine Übersetzungen vorgenommen worden sind. Die Auswahlmöglichkeit „Nur neue oder geänderte Zeichenfolge exportieren“ kann besonders bei späteren Exporten genutzt werden. In solchen Fällen will man nur die Strings erhalten, die noch nicht übersetzt wurden, da es sich um Inhalte aus neuen oder geänderten Tabellen handelt.

Darüber hinaus ist festzulegen, wie identische Strings behandelt werden sollen. Entweder können für alle identischen Strings eigene IDs kreiert oder es kann dieselbe ID für alle identischen Strings verwendet werden. Letzteres bietet Vorteile bei der späteren Wartung. **Abbildung 4** zeigt die dazugehörige Eingabemaske.

Im folgenden Schritt wird die erzeugte XML-Datei geöffnet und es werden alle erforderlichen Strings übersetzt. Diesen Schritt wiederholt man für alle gewünschten Sprachen in den jeweiligen XML-Dateien. Nach Ab-

schluss der Übersetzung werden die Daten in der Web-Oberfläche über den Navigationspfad „Administration / Datei und Metadaten neu laden“ neu eingelesen. Anschließend sollte über die BI-Web-Oberfläche getestet werden, ob alle Änderungen korrekt übernommen wurden.

Es besteht außerdem die Möglichkeit, neben dem kompletten Ordner auch nur einzelne ausgewählte Berichte und deren XML-Dateien zu exportieren. Hierfür wählt man lediglich den zu übersetzenden Bericht aus und klickt anschließend wieder über „Extras“ auf „Titel exportieren“. Auch hier kann man wie zuvor auswählen, welche Bestandteile exportiert werden und in welchem Ordner die neu erzeugte XML-Datei abgelegt wird.

Fehlerhafte Übersetzungen können durch neu zugeteilte IDs zustande kommen. Im Katalog-Manager wird jeder String mit einer ID gekennzeichnet, die dem „Attributnamen“ in den XML-Übersetzungsdateien entspricht. Es kann passieren, dass das BI-Tool diese ID infolge der Bearbeitung von Dashboard- und Answers-Berichten ändert. Daher sollten zunächst die bislang nicht übersetzten Strings in der XML-Datei auffindig gemacht und bezüglich der ID überprüft werden. Gegebenenfalls muss man die IDs anpassen. Eine weitere Möglichkeit besteht darin, den BI-Server und den BI-Presentation-Server neu zu starten, um die Übersetzungen abermals zu laden.

Fazit

Der Artikel beschreibt die Lokalisierung im OBI Repository und im Web-Katalog. Es wurden Beispiele genannt, die in der Praxis zu Schwierigkeiten führen können. Speziell beim Lokalisieren des Web-Katalogs stellt sich die Frage, ob es Hilfsmittel zur Automati-

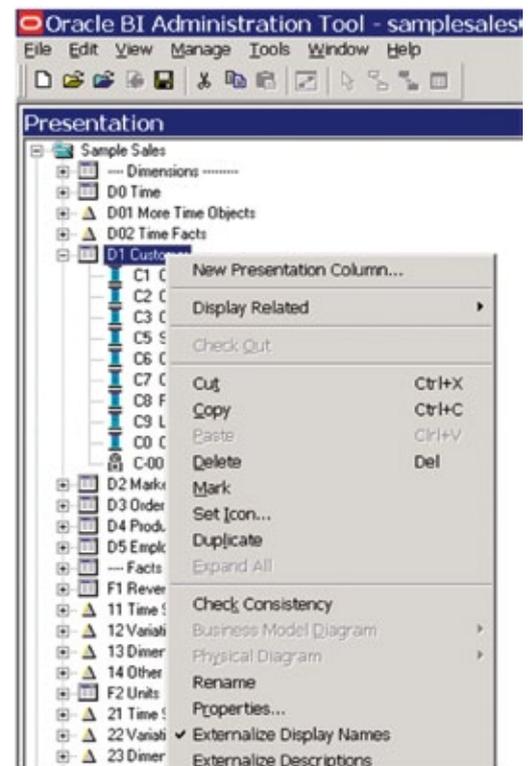


Abbildung 4: Hier kann für die ganze Tabelle der Feld-Name lokalisiert werden. Ist nur ein Teil der Felder lokalisiert, wird das gezeigte Häkchen grau statt schwarz dargestellt.

sierung gibt. In der Tat gibt es ein Tool namens „BI Localizer“, das jedoch kostenpflichtig von einem Dritt-Hersteller erworben werden muss. Es automatisiert bis auf die Übersetzung der Strings alle durchzuführenden Aufgabenschritte. Der Benefit für den IT-Dienstleister beziehungsweise Administrator besteht in einer deutlichen Zeitersparnis und dem Ausschluss praktisch aller Fehlerquellen.

Dr. Götz Gleitsmann
goetz.gleitsmann@orbit.de

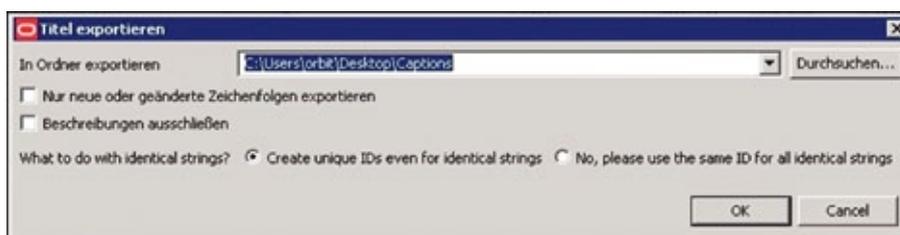
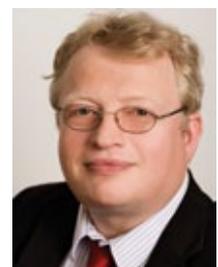


Abbildung 3: Exportieren der XML-Datei

Einen HTML-Anker in eine Apex-Applikation einbauen

Philip Ton, OPITZ CONSULTING GmbH

Jeder Endbenutzer von Application Express kennt das: Man möchte sich mal eben einen Datensatz im Detail ansehen, doch beim Zurückspringen auf die Master-Ansicht landet man statt auf der Höhe des vorher ausgewählten Datensatzes wieder am Seitenanfang.

Dem Endbenutzer die vorherige Scroll-Höhe anzubieten, ist eine Komfort-Funktion, die die Arbeit mit Apex-Anwendungen angenehmer macht. Dieser Artikel zeigt, wie man einen Anker schnell in bestehende oder neue Reports einbauen und ihn bei Bedarf aktivieren kann, ohne dazu Frontend und Code deutlich anpassen zu müssen.

Anwender können mit Oracle Application Express (Apex) Tabellen als Report betrachten. An sich ein schönes Feature, gäbe es da nicht einen kleinen Schönheitsfehler: Da die Zeilen meist auf Datensatz-Ebene angeordnet sind, stellen sie nur einen Bruchteil der möglichen Ansicht dar. Dazu kommt, dass es häufig so viele Datensätze auf einer Seite gibt, dass der Nutzer nach unten scrollen muss, um weitere In-

formationen zu sehen. Um sich ein Element in der Master-Ansicht genauer anzuschauen, steht eine Detail-Ansicht zur Verfügung. Springt man von dort aus allerdings wieder zurück auf die Master-Ansicht, wird die Seite neu geladen und, sofern der Datensatz außerhalb des Sichtbereichs liegt, ist nur der obere Bereich zu sehen.

Bei diesem Problem hilft ein klassisches HTML-Element. Das als „Anker“ bekannte Element wird auf Datensatz-Ebene eingebaut und gewährleistet beim Zurückspringen die komplette Sichtbarkeit des vorher ausgewählten Datensatzes (siehe [Abbildung 1](#)).

In vier leicht zu implementierenden Schritten wird gezeigt, wie schnell man seinen Kunden oder Endbenutzern auf diese Weise eine Freude machen kann.

Erstellen eines Hidden Items

Im ersten Schritt wird ein Hidden Item angelegt, um die Datensatz-ID zwischenspeichern. Die ID für den Anker muss zwischengespeichert werden, wenn die Navigation von der Masterseite auf die Detailseite stattfindet. Damit dieses Page Item nicht auf jeder betroffenen Seite angelegt werden muss, wird es einmalig auf der Page Zero (Global Page) erstellt. Von jeder Seite aus wird dann auf dieses Page Item zugegriffen. Es sollte als Hidden Item angelegt sein, damit die ID nicht zwangsweise vom Endbenutzer gesehen wird. Ein möglicher Name für das Page Item ist „PO_ANCHOR_ID“.

Anpassen der Lade-Prozesse zum Speichern der Datensatz-ID

Im Regelfall befindet sich auf einer Detailseite ein Ladeprozess, der die Daten mithilfe eines erhaltenen Primärschlüssels (die Datensatz-ID) in entsprechende Page Items lädt. An dieser Stelle wird angeknüpft und dem in Schritt 1 angelegten Hidden Item ein Wert zugewiesen (siehe [Listing 1](#)).

Einbau des Ankers im Interactive Report

Der vorher ausgewählte Datensatz soll beim Rücksprung von der Detailseite auf die Masterseite wieder im Sichtfeld des Browsers erscheinen. Damit dies funktioniert, sind die Anker auf Datensatz-Ebene im Interactive Report der Masterseite einzubauen. Im Vorfeld ist zu überlegen, in welcher Spalte der Anker eingebaut werden soll. Folgende Kriterien sind dabei zu beachten:

- Es empfiehlt sich, eine Spalte auszuwählen, die nie vom Endbenut-

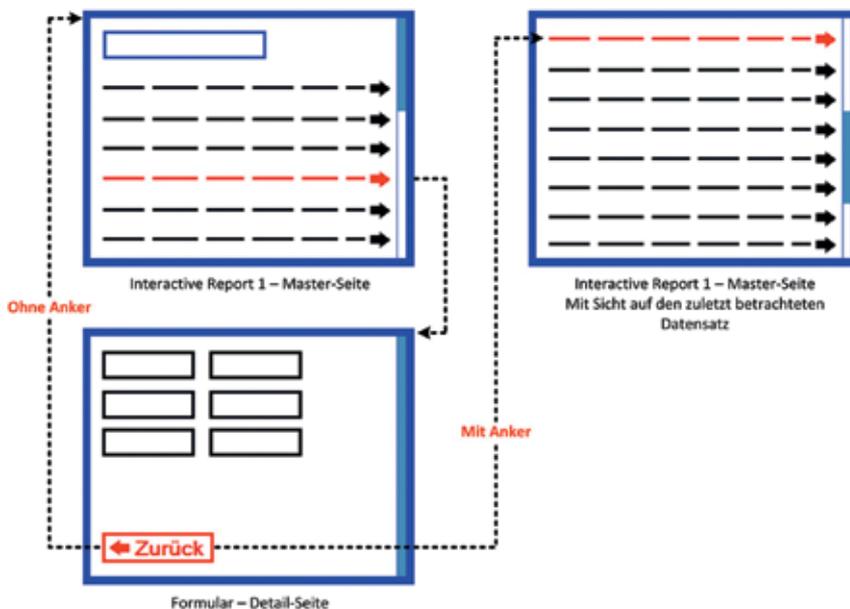


Abbildung 1: Apex-Anker

```

/* Standard Lade-Prozess */
BEGIN
  SELECT Kunden_Name,
         Kunden_Vorname,
         Kunden_Ort
         --[...]
  INTO :P5_KUNDEN_NAME,
       :P5_KUNDEN_VORNAME,
       :P5_KUNDEN_ORT
       --[...]
  FROM Kunden
  WHERE Kunden_ID = :P5_KUNDEN_ID;
END;

/* Lade-Prozess mit Wertzuweisung vom Hidden Item */
BEGIN
  SELECT Kunden_Name,
         Kunden_Vorname,
         Kunden_Ort
         --[...]
  INTO :P5_KUNDEN_NAME,
       :P5_KUNDEN_VORNAME,
       :P5_KUNDEN_ORT
       --[...]
  FROM Kunden
  WHERE Kunden_ID = :P5_KUNDEN_ID;

  -- Es folgt die Wertzuweisung vom Hidden Item:
  PO_ANCHOR_ID := :P5_KUNDEN_ID;
END;

```

Listing 1

zer ausgeblendet beziehungsweise deaktiviert wird, damit der HTML-Code dem Anker auch in jedem Fall zur Verfügung steht.

- Die Spalte, in die der Anker eingebaut wird, sollte von vornherein dem Datentyp „String“ entsprechen. Der Einbau des Ankers in eine Spalte, die später den Datentyp „Number“ oder „Date“ enthält, wird den Typ zunächst unweigerlich zu einem String wandeln. Auch wenn der HTML-Code später nicht mehr sichtbar ist, könnte diese Entscheidung die von Apex angebotene Sortierung dieser Datentypen fatal verändern.
- Die ID des Ankers sollte so eindeutig sein, dass auf den richtigen Datensatz zurückgesprungen werden kann. Deshalb empfiehlt es sich, als ID eine Spalte auszuwählen, die auf Datensatz-Ebene eindeutig ist. Dies trifft zum Beispiel auf den Primär-Schlüssel zu, wie das Quellcode-Beispiel zeigt.

Der Einbau des Ankers ist schnell programmiert und anhand des Quell-

codes leicht ersichtlich (siehe Listing 2).

Die Spalte würde den HTML-Code in der Anwendung demnach zunächst nicht als Code, sondern als String ausgeben. Dafür muss die Spalten-Eigenschaft noch verändert werden. Ziel ist es, die Column-Attribute für die betroffene Spalte zu ändern (siehe Abbildung 2). Dieser Vorgang

```

/* Standard SQL-Query*/
SELECT Kunden_ID,-->Primärschlüssel
       Kunden_Name,
       Kunden_Vorname,
       Kunden_Ort
       --[...]
  FROM Kunden;

/* SQL-Query mit HTML-Anker (id) */
SELECT Kunden_ID,-->Primärschlüssel
       ,<a id="" || Kunden_ID || ,">' || Kunden_Name Kunden_Name,
       Kunden_Vorname,
       Kunden_Ort
       --[...]
  FROM Kunden;

```

Listing 2

wird durch die folgenden Schritte erreicht:

1. Eigenschaften des Interactive Reports öffnen (zum Beispiel per Doppelklick).
2. Wechsel auf die Registerkarte „Report Attributes“
3. Im Abschnitt „Column Attributes“ die entsprechende Spalte auswählen und unter „Display Text As“ die Eigenschaft „Standard Report Column“ auswählen. Dies hat zur Folge, dass der HTML-Code auch als solcher wahrgenommen wird, sich in die Seite integriert und verhindert, dass der Code als String oder Text identifiziert wird und damit lediglich auf der Seite ausgegeben würde.
4. Abschließend mittels „Apply Changes“ die Änderung speichern.

Umstellen der Rückwärts-Navigation auf den Anker

Abschließend ist die Rückwärts-Navigation so umzustellen, dass sie bei Aktivierung in der generierten URL den Anker mit sich führt, der dafür sorgt, dass der Nutzer auf den Datensatz der Masterseite zurückspringen kann. Dafür wird zunächst der Branch oder Button auf der Detailseite ausgewählt, der für die Rückwärts-Navigation auf die Masterseite zuständig ist. Beide haben gemeinsam, dass sie Page-Item-Werte zuweisen können. Diese Zuweisung wird mit der generierten URL wirksam.

Um den technischen Hintergrund des Ankers besser verständlich zu machen, wird an dieser Stelle kurz der

Aufbau der URL in Apex-Anwendungen erläutert, von der bei diesem Vorgang Gebrauch gemacht wird: Die Stelle vor dem letzten Doppelpunkt in der URL dient der Auswahl der Variablen, die zu deklarieren sind. Die Stelle danach beinhaltet die damit zu füllenden Werte in der jeweiligen Reihenfolge. Dazu als Beispiel die Adresse „http://123.1.123.123:7778/pls/apex/f?p=100:2:5498012442492::NO::P2_TEST:4711“. Beim Öffnen dieser URL würde der Benutzer auf Seite 2 gelangen. Das Page Item „P2_TEST“ auf dieser Seite würde mit dem Wert „4711“ deklariert sein.

Auch zu den Anker in URLs an dieser Stelle ein kleiner Exkurs. Ein Anker würde in einer gewöhnlichen URL so aussehen: „http://www.adresse.org#AnkerName“, „http://www.doag.org#top“ würde also bezwecken, dass der Anfang der Seite „www.doag.org“ zu sehen ist.

Dem Beispiel folgend, würde ein Anker in der URL der Apex-Anwendung wie folgt lauten: „http://123.1.123.123:7778/pls/apex/f?p=100:2:5498012442492::NO::#123“. Beim Öffnen dieser URL würde der Benutzer abermals auf die Seite 2 gelangen. Sollte der Entwickler einen HTML-Anker mit der ID „123“ im Interactive Report eingebaut haben, würde der Betrachter auf die Höhe dieses Ankers springen und ihn damit in den sichtbaren Bereich befördern.

In einem Branch („Action“ – „Page in this application“) oder Button („Action when button clicked“ – „Redirect to page in this application“) besteht die Möglichkeit, Page Items einen Wert zuzuweisen. Das Page Item, dessen Wert in der URL als String aufgelöst werden soll, kann man unter „With these values“ auswählen. In diesem Fall wird das Page Item genommen, das die Datensatz-ID zwischengespeichert hat. Zusätzlich wird vor das Item noch eine Raute geschrieben, um den Anker in der URL zu erhalten, zum Beispiel „#&P0_ANCHOR_ID“.

Beim Speichern der Einstellung mittels „Apply Changes“ und beim erneuten Öffnen des Branch sollte sich der Target-Type in URL geändert haben und unter „URL Target“ die URL in der gleichen Schreibweise zu sehen sein, in der sie auch beim Aktivieren des Branch erscheinen würde.

| | Heading | Type | Link | Display Text As |
|--|----------------|----------------|--------|---|
| | KUNDEN_NAME | Kunden Name | STRING | Standard Report Column |
| | KUNDEN_VORNAME | Kunden Vorname | STRING | Display as Text (escape special characters) |
| | KUNDEN_ORT | Kunden Ort | STRING | Display as Text (escape special characters) |

Abbildung 2: Column-Attribute

Achtung: Wenn die oben genannten Änderungen beim Button wirksam werden und funktionieren, jedoch alle Felder in der betreffenden Region leer sind, liegt die Vermutung nahe, dass es sich um den gleichen Prozess wie beim Branch handelt. Nur wurde dann die entsprechende Auswahl nicht in der Entwicklung berücksichtigt. Im Zweifelsfall sollte deshalb immer ein Branch benutzt werden, wie er zum Beispiel beim Aktivieren eines Buttons ausgelöst wird (siehe Abbildung 3).

Noch zu beachten

Beim Navigieren von einer Detailseite auf eine andere kommt es zu Pro-

blemen, weil die „P0_ANCHOR_ID“ überschrieben wird. Um dem vorzubeugen, bedarf es keiner komplexeren Logik. Schon ein weiteres „Page Zero Page Item“ schafft hier Abhilfe. Für einen Endbenutzer ist es verlockend, in Apex-Anwendungen die Rückwärts-Navigation des Browsers zu benutzen. Auch dann muss nicht zwingend der zu bearbeitende Datensatz zu sehen sein.

Fazit

Die Implementierung eines Ankers mit dem Ziel, den zuvor betrachteten Datensatz wieder ohne Scrollen im Blickfeld zu haben, ist kein Hexenwerk. Die

Abbildung 3: Eigenschaften des Branch-Buttons

Anpassungen sind auch bei schon bestehenden Anwendungen sehr gering und als Entwickler können Sie diese Änderungen sehr schnell und routiniert vornehmen. Dieses Vorgehen empfiehlt sich besonders auf Seiten, auf denen sehr viele Datensätze auf einmal angezeigt werden oder wenn

es dazu noch ein Action-Menü gibt, in dem weitere Einstellungen vorgenommen werden können. Die Endbenutzer werden Ihnen für dieses kleine Feature dankbar sein, wenn sie zukünftig nicht mehr für jeden Datensatz umständlich scrollen müssen.

Philip Ton

philip.ton@opitz-consulting.com



Oracle SQL – Das umfassende Handbuch

gelesen von Bernd Tuba, regionaler DOAG-Repräsentant Trier/Saarland/Luxemburg

Gleich vorweg: Es handelt sich nicht um eine „SQL Referenz“, die wie in vielen Fällen mehr oder weniger eine deutsche Übersetzung der Oracle-Dokumentation sowie etwaiger Notes von Oracle Technical Network (OTN) beinhaltet. Vielmehr dient das Buch als praxisnaher Leitfaden mit vielen Beispielen und Übungen.

Das Buch ist gut strukturiert, was sich direkt im Inhaltsverzeichnis prüfen lässt. Auch der Index ist logisch aufgebaut. Ich habe ihn des Öfteren genutzt und bin immer fündig geworden.

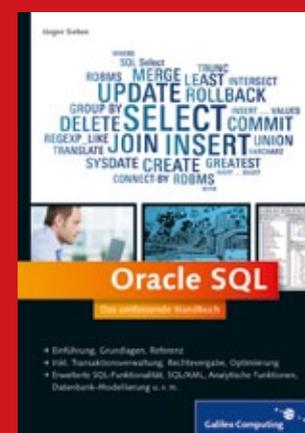
Es gibt an den verschiedensten Stellen Hinweise zu speziell in 11g R2 eingeführten Funktionalitäten und auch bereits Ausblicke auf die nächste Datenbank-Version 12c. Gerade Letzteres steigert schon die Neugierde auf diese neue Version.

Idealerweise sind direkt nach Inhalt, Einführung und dem Dankeschön an die Ehefrau in Kapitel 2 die zum Erlernen von SQL technischen Voraussetzungen Schritt für Schritt aufgezeigt. Es ist beschrieben, von welchen Online-Quellen man die (Übungs-)Datenbank und etwaige Tools (SQL Developer) herunterladen sowie installieren und konfigurieren kann. Dabei wird auch auf feine Details wie die Zeichensatz-

Codierung eingegangen. Zur weiteren Vorbereitung zählt auch das Installieren (Ausführen) der Beispielskripte, die zum Download bereitstehen und nicht manuell eingetippt werden müssen. Zum Abschluss dieses Kapitels wird auch noch auf OTN und die Oracle-Online-Dokumentation verwiesen.

Wie Kapitel 2 liefern auch die beiden Folgekapitel für Einsteiger notwendige Grundlagen. Dies sind zum einen die Basics der relationalen Datenbanken und zum anderen die grundlegenden Architekturen und Prozesse einer Oracle-Datenbank. Gerade bei Letzterem wären ein paar Grafiken hilfreich gewesen. Allerdings rundet die Erklärung der Datentypen zum Abschluss dann wieder perfekt ab. Somit ist Teil I des Buches „Einführung und Grundlagen“ abgeschlossen.

Anschließend wird dann auch erstmals der direkte Kontakt zu SQL gesucht, der dann in den Folgekapiteln auch entsprechend gepflegt und ausgebaut wird. So kann man die Kapitel 5, 6, 7 und 8 als unabdingbares Grundlagenwissen bezeichnen. Gerade die in Kapitel 7 ausreichend kurz und mit verständlichen Beispielen aufgeführten Zeilen-Funktionen werden in der Praxis sicherlich immer wieder als



Nachschlagewerk dienen. Nicht nur für diese Kapitel gilt, dass die spannenden Übungen eine ideale Möglichkeit bieten, das gerade Erlernte zu üben und vertiefen. Ab der Beschreibung der Gruppen-Funktionen in Kapitel 8 wird der Stoff dann allerdings anspruchsvoller, aber auch interessanter.

So zum Beispiel die analytischen Funktionen: Obwohl ich in der Vergangenheit bereits ein paar (DOAG-)Vorträge zu diesem Thema besucht habe, war meine praktische Anwendung eher gering. Und dennoch war Kapitel 9 für mich eines der absoluten Highlights des Buches. Wie der Autor treffend erklärt, sollte man dieses Thema nicht nur im Bereich eines DWH ansiedeln. Die Beispiele sind teilweise komplex, aber detailliert beschrieben. Auch gibt der Autor einen „Step by Step“-Guide, der nochmals die einzelnen Schritte aufführt. Besser kann man dem Leser ein anspruchsvolles Thema nicht nahebringen.

Das abschließende Kapitel „Unterabfragen“ bietet einen gelungenen

Abschluss zum II. Teil des Buches. Alle, die an reinen Abfrage-Möglichkeiten beziehungsweise am Reporting interessiert sind, haben nun die entsprechenden Grundlagen erreicht, eigentlich sogar mehr.

Teil III des Buches startet in Kapitel 11 mit der Daten-Manipulation. Hier findet sich eine gute und verständliche Einführung zum Thema „Transaktionen“. Anschließend werden DML-Befehle erläutert. Allerdings gibt es hier eine kleine Passage zum „Recycle Bin“, die leicht missverständlich sein kann. Natürlich werden Tabellen ab Version 10g in den Recycle Bin verschoben (solange der DBA dies nicht deaktiviert). Man sollte allerdings beachten, dass Objekte dort nicht beliebig lange verbleiben können/müssen (etwa wenn der Speicherplatz des Tablespace zur Neige geht).

Hilfreich sind auch in Kapitel 12 und 13 wieder die Praxis- und Projekt-Erfahrungen des Autors, die regelmäßig einfließen. Sie zeigen immer wieder einen direkten, praktischen Bezug zur grauen Theorie. Auch schafft er an verschiedenen Stellen die ideale Gradwanderung zwischen Anwendungs-Entwicklern und Datenbank-Administratoren. Dann kommt auch mal ein Hinweis auf „Exadata“ im Absatz „Index-Mythenbildung“ oder darauf, dass zu viele Indizes nicht Vor- sondern Nachteile mit sich bringen.

Im ersten Moment würde man Kapitel 15 („Rechte-Vergabe“) wohl eher im Bereich der DBAs und somit auch nicht zwingend in einem SQL-Buch ansiedeln. Aber wie bereits in den vorangegangenen Kapiteln gelingt auch hier der Spagat zwischen Oracle-Allgemein- und DBA-Spezialwissen. Um jedoch Missverständnisse auszuschließen: Dieses Buch ist natürlich kein DBA-Buch.

„Spezielle Abfragetechniken“ werden dann im IV. Teil des Buches behandelt. Dem Titel stimme ich zu – die zugehörigen Kapitel sind recht speziell und sicher nicht für jeden notwendig. Aber sie sind sehr interessant und ermöglichen an den verschiedensten Stellen, die neuen Thematiken mit dem vorher Erlernten zu verknüpfen!

Im 16. Kapitel geht es dann um hierarchische Abfragen. Hier wird dann

ein tolles Beispiel behandelt, in dem der kritische (Projekt-)Pfad über eine hierarchische Abfrage ermittelt wird. Feinheiten bezüglich der zugehörigen Datenmodellierung sind dann in Kapitel 24 („Speicherung hierarchischer Daten“) erläutert.

Kapitel 17 („XML-Abfragen“) und 18 („Die Model-Klausel“) sind dem Titel des Teilbereichs entsprechend speziell. XML-seitig wird detailliert auf die Verwendung des SQL/XML-Standards eingegangen. Auch wird hier sinnvollerweise auf die weiterführenden Kapitel im PL/SQL-Buch des Autors verwiesen.

Interessant sind auch zwischenzeitliche Ausflüge in Begrifflichkeiten, die indirekt im Zusammenhang mit der Materie stehen. Ein gutes Beispiel ist hier der Exkurs zu „DOM“ und „SAX“ im Kapitel „XML-Abfragen“.

Ich würde Kapitel 20 („Umgang mit Datum und Zeit“) eigentlich jedem empfehlen, obwohl es hauptsächlich Entwickler (und natürlich auch DBAs) interessiert, die Anwendungen für den internationalen Einsatz entwickeln beziehungsweise administrieren. So kompakt und dennoch ausreichend detailliert habe ich über diese Thematik bisher nichts gelesen.

Kapitel 21 („Objektorientierung in der Oracle-Datenbank“): Was bringt dieses Kapitel? Wir wissen nun zumindest, dass es sich beim Autor um eine rheinische Frohnatur handelt ... Es ist sicherlich nicht jedermanns Thema, jedoch allen Datenbank- und objektorientierten Entwicklern empfohlen, die in einem gemeinsamen Projekt vor der Problematik des OR-Mappings auf die relationale Welt stehen. Sie werden zwar keine Lösung „out of the book“ erhalten, aber eine neutrale Sicht auf die Probleme beider Welten in diesen Projekten finden. Die jeweilige (Ein-)Sicht auf die Probleme des anderen ist ja schon ein bedeutender Schritt nach vorn. Ergänzend zu diesem Kapitel wird in Kapitel 26 auf die „Abbildung objektorientierter Strukturen“ eingegangen.

Im letzten Teil geht es um den Bereich „Datenmodellierung“. Die Grundlagen sind dann auch gleich im Kapitel 22 aufgezeigt. Auch hier gibt es wieder ein gutes Praxisbeispiel, bei dem aufgrund der alphanumerischen Primärschlüsselwer-

te die Zeichensatz-Konvertierung nicht durchgeführt werden konnte.

Fazit

Nicht nur, aber im Besonderen für Einsteiger, ist dieses Buch perfekt. Die vielen praxisnahen Beispiele und Übungen helfen, den theoretischen Stoff direkt mit der realen (IT-)Welt zu verknüpfen. Es eignet sich daher geradezu zum SQL-Selbststudium. Aber auch erfahrene Datenbank-Entwickler werden hier ihre Freude haben. Sie können sich der eventuell noch unbekannteren Thematik der jeweiligen Kapitel widmen, jedoch auch bei sich stellenden bekannten Aufgaben einfach mal einen Blick ins Buch werfen. Ich könnte mir durchaus vorstellen, dass der eine oder andere einen neuen Blickwinkel oder Ansatz zur Problemlösung findet. Auch für reine DBAs lohnt sich der Blick in Buch. Wie bereits in den einzelnen Kapitelbeschreibungen erläutert, gibt es einige Überschneidungen in den jeweiligen Themengebieten.

Insgesamt ist dieses Buch in Stil und Inhalt außergewöhnlich. Ich kenne keinen Autor, der SQL derart lebhaft niederschreiben hat und den Leser gleichzeitig so gekonnt einfangen kann. Gepaart mit der offensichtlich enormen Praxiserfahrung, die ungefiltert eingeflossen ist, ergibt sich eine absolute Kaufempfehlung.

Für mich jedenfalls ist sicher: Dieses Buch wird einen festen Platz auf meinem Schreibtisch einnehmen und dort stehen tatsächlich nicht viele Bücher. An letzter Stelle noch ein Satz an Frau Sieben: Wahrscheinlich würden Sie viele Menschen äußerst glücklich machen, wenn Sie Ihren Mann beim nächsten Buch nochmals wohlwollend und tatkräftig unterstützen ...

| | |
|-----------------|---|
| Titel: | Oracle SQL – Das umfassende Handbuch |
| Autor: | Jürgen Sieben |
| Verlag: | Galileo Computing |
| Umfang: | 877 Seiten |
| Sprache: | Deutsch |
| Preis: | 59,90 Euro (Online-Ausgabe 54,90 Euro, Buch plus Online-Ausgabe 74,90 Euro) |
| ISBN: | 978-3-8362-1875-7 |

Hidden Secrets: Administrations-Skripte in Cloud Control nutzen

Ralf Durben, ORACLE Deutschland B.V. & Co. KG

Auch wenn Oracle Enterprise Manager Cloud Control viele interaktive Möglichkeiten zur Verwaltung von Datenbanken anbietet, gibt es immer wieder den Wunsch, bereits bestehende Skripte einzubinden, die mehrere Aktionen gemeinsam ausführen. Diese Skripte sind zumeist parametrisiert und sollen über ein Sicherheitskonzept nur bestimmten Administratoren zur Verfügung stehen.

Das Jobsystem von Cloud Control hilft in diesem Fall nicht weiter, da die dort erstellten Jobs keine Parametrisierung erlauben. Dagegen können die Deployment Prozeduren aus dem Lifecycle Management Pack sehr gut für diese Aufgabe verwendet werden. Anhand eines einfachen Beispiel-Skripts zeigt dieser Artikel, wie man dieses umsetzen kann.

Das Skript dient zur Erstellung eines Datenbank-Benutzers. Dabei wird dieser nicht nur einfach erstellt, sondern gleich mit den passenden Privilegien ausgestattet, es findet also mehr als nur ein einfaches „CREATE USER“ statt. Über einen Parameter kann der Typ des neuen Datenbank-Benutzers festgelegt werden, was dann auch automatisch die Privilegien festlegt, die zu vergeben sind. Als Beispiel für ein oben beschriebenes Administrations-Skript dient folgendes Shell-Skript (siehe Listing 1).

```
[oracle@radu1 ~]$ more credbuser.sh
ORACLE_SID=$1
export ORACLE_SID
export ORAENV_ASK="NO"
. oraenv
export PATH=$ORACLE_HOME/lib:$PATH
export LD_LIBRARY_PATH=$ORACLE_HOME/lib
echo $ORACLE_HOME
echo $ORACLE_SID
echo $PATH

sqlplus /nolog << EOF
connect / as sysdba
declare
  username varchar2(100):=to_char('$2');
  password varchar2(100):='$3';
  usermode char(1):='$4';
begin
  execute immediate('create user '||username||' identified by
' ||password);
  if usermode='c' then
    execute immediate('grant connect to '||username);
  elsif usermode='r' then
    execute immediate('grant resource to '||username);
  end if;
end;
/
exit
```

Listing 1

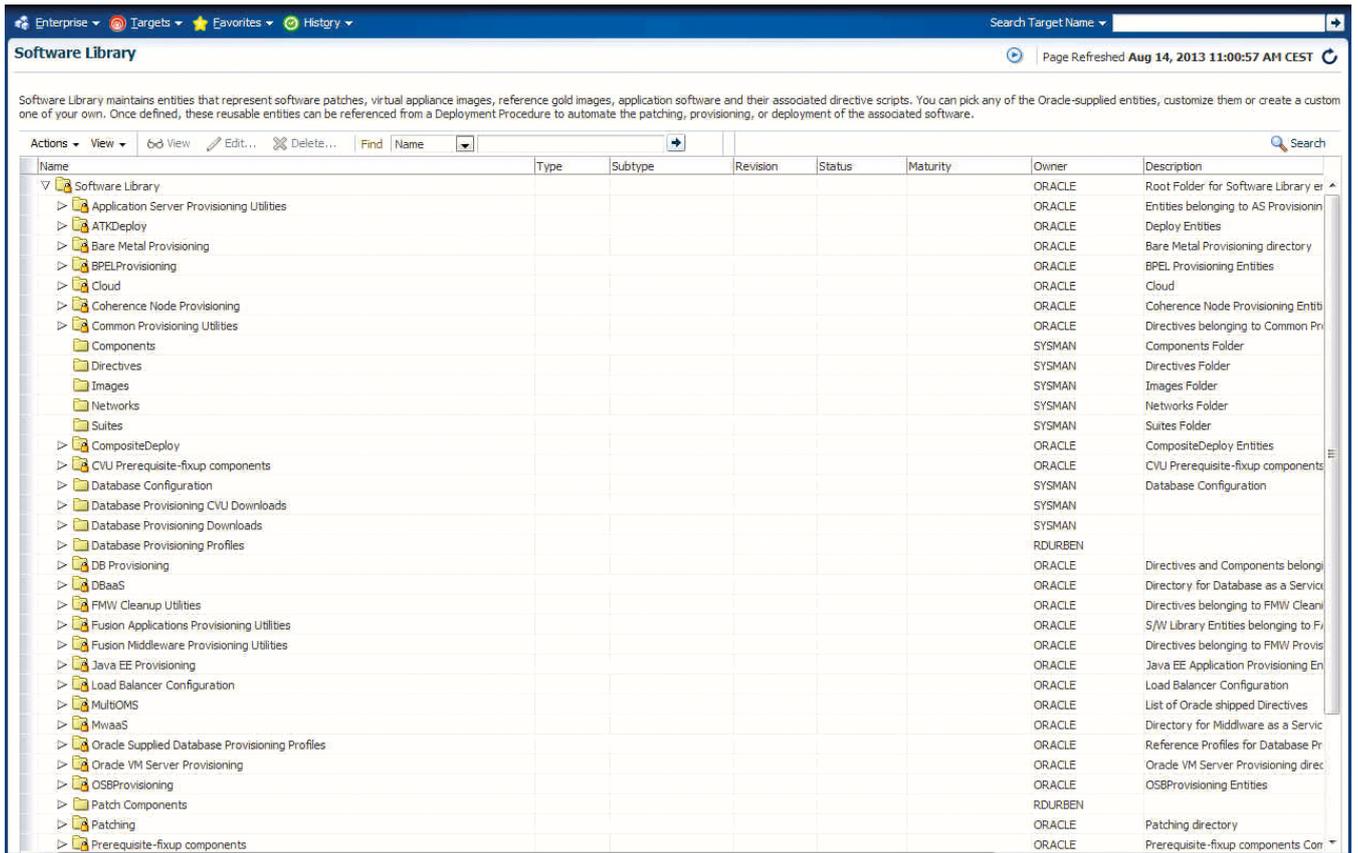


Abbildung 1: Software Library in Cloud Control 12c

Als Parameter für dieses Skript gibt man den Namen des zu erstellenden Datenbank-Benutzers samt Passwort und Typ des Benutzers sowie die betreffende Datenbank-Instanz an: „./credbuser.sh orcl1 testuser1 passwort1 c“. Der angegebene Benutzer-Typ legt die Privilegien fest, die automatisch vergeben werden sollen. Um das Beispiel einfach zu halten, werden hier nur zwei Typen unterschieden und daraufhin entweder die Rolle „CONNECT“ oder „RESOURCE“ vergeben.

Wichtig ist bei diesem Skript das Setzen von „LD_LIBRARY_PATH“, damit das Anmelden als „SYSDBA“ funktioniert. Dieses Skript wird später vom Cloud-Control-Agenten als Benutzer „ORACLE“ ausgeführt. Der Vorteil dieses Verfahrens ist, dass für das Skript volle Rechte vorliegen, diese aber nur innerhalb des Skripts gültig sind. Bei der Entwicklung dieses Skripts ist also darauf zu achten, dass keine unberechtigte Aktion durch Angabe entsprechender Parameterwerte gestartet

werden kann (etwa SQL Injection bei PL/SQL-Prozeduren). Selbstverständlich kann auch ein anderer Benutzer verwendet werden.

Die Software Library

Um das Administrations-Skript durch eine Deployment-Prozedur auszuführen, kann man dieses selbst auf die Zielsysteme kopieren oder die Software Library von Cloud Control nutzen, um das Skript zum Zeitpunkt seiner Ausführung durch die Deployment-Pro-

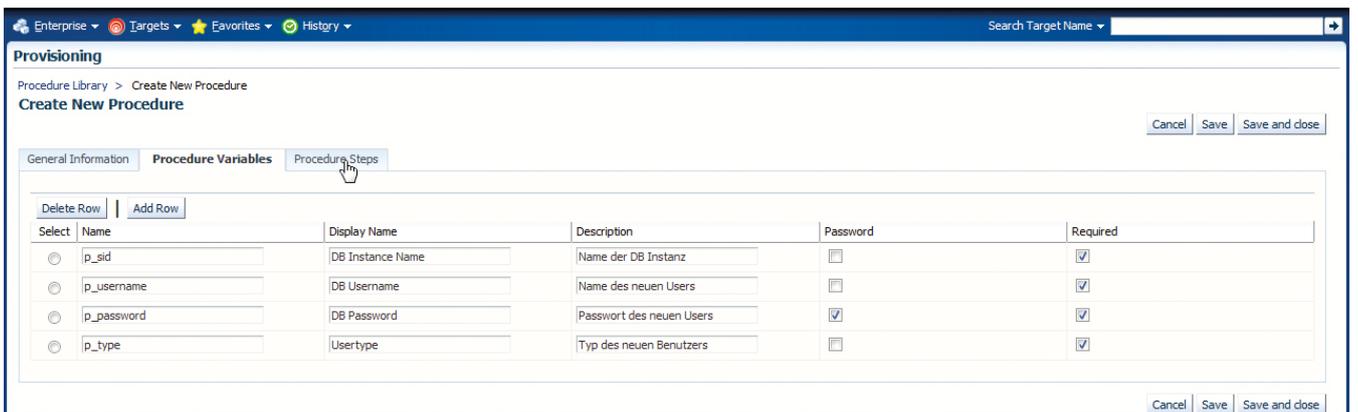


Abbildung 2: Parameter für eine Deployment-Prozedur festlegen

zedur zu übertragen. Letzteres hat den großen Vorteil, dass immer das aktuelle Administrations-Skript ausgeführt wird. Die Software Library erscheint in Cloud Control wie ein normales Dateisystem. Man kann Verzeichnisse anlegen und dann das Skript als generische Komponente hochladen (siehe [Abbildung 1](#)).

Die Deployment-Prozedur

Eine Deployment-Prozedur besteht aus mehreren Schritten, die in Blöcken gebündelt werden können. Darüber hinaus kann man Parameter erstellen, deren Werte dann in den einzelnen Schritten verwendet werden. Dies gibt die Möglichkeit, im obigen Beispiel die Werte für die Instanz, den Namen des neuen Datenbank-Benutzers, das Passwort und den Benutzer-Typ in der Deployment-Prozedur abzufragen und das Administrations-Skript mit diesen Werten aufzurufen (siehe [Abbildung 2](#)). Die Ausführungsschritte in der Deployment-Prozedur sind dabei recht einfach:

- Kopieren des Administrations-Skripts auf den Zielrechner
- Setzen der notwendigen Ausführungsrechte für das Skript
- Starten des Skripts

Kopieren des Administrations-Skripts auf den Zielrechner

Das Kopieren von Bestandteilen der Software-Library durch eine Deployment-Prozedur erledigt man einfach über einen Ausführungsschritt vom Typ „Library:Component“. Dabei ent-

```
/home/oracle/em_scripts/credbuser.sh ${data.p_sid} ${data.p_username}
${data.p_password} ${data.p_type}
```

Listing 2

scheidet man sich für eine Komponente aus der Software-Library, gibt dem Schritt einen Namen und wählt im Dialog noch die Direktive mit dem Namen „Copy“ aus. Für diese gibt man dann noch an, in welches Verzeichnis auf dem Zielsystem diese Komponente kopiert werden soll (siehe [Abbildung 3](#)).

Nach dem Transfer der Skriptdatei auf das Zielsystem sind je nach Betriebssystem die notwendigen Ausführungsrechte nicht gesetzt. Auf UNIX-Systemen zum Beispiel ist für diese Datei ein „chmod +x dateiname“ erforderlich. Dieses wird in der Deployment-Prozedur in einem eigenen Ausführungsschritt vom Typ „Host Command“ durchgeführt.

Zum Schluss erstellt man einen weiteren Ausführungsschritt vom Typ „Host Command“. Dabei gibt man das Administrations-Skript an, das ausgeführt werden soll, und nutzt die vorbereiteten Parameter der Deployment-Prozedur (siehe [Listing 2](#)).

Ausführungsrechte

Das Ausführungsrecht für Deployment-Prozeduren wird in Cloud Control anders als bei Grid Control individuell vergeben. Damit kann genau festgelegt werden, welcher EM-Benutzer eine Deployment-Prozedur ausführen kann. Bei deren Ausführ-

ung wird nach einem Host-Credential für den Zielsystem gefragt, sodass hier durch die Zuweisung von „Named Credentials“ eine weitere Nutzungseinschränkung vorgenommen werden kann. Weitere Einschränkungen sind durch Erweiterung der Deployment Prozedur möglich. Lizenz-Hinweis: Die Nutzung von Deployment-Prozeduren ist im Rahmen des Lifecycle-Management-Packs separat lizenzpflichtig.

Fazit

In grafischen Tools wie Cloud Control kann man alle Administrationsaufgaben in Form einzelner Aktionen durchführen. Um diese Aktionen in Skripten zu bündeln, weil man das zum Beispiel durch historisch gewachsene Administrations-Skripte gewohnt ist, kann man sie in Cloud Control einbinden und mithilfe von Deployment-Prozeduren ausführen. Durch die Verwendung der Software-Library ist sichergestellt, dass auf den Zielsystemen immer die aktuelle Version eines Skripts verwendet wird.

Weitere Informationen zur Nutzung von Deployment-Prozeduren und alle Schritte zur Einbindung des Beispiels im Detail stehen im „Oracle Online Dojo Nr. 3: Oracle Enterprise Manager 12c Cloud Control“ unter <http://tiny-url.com/dojoonline>.

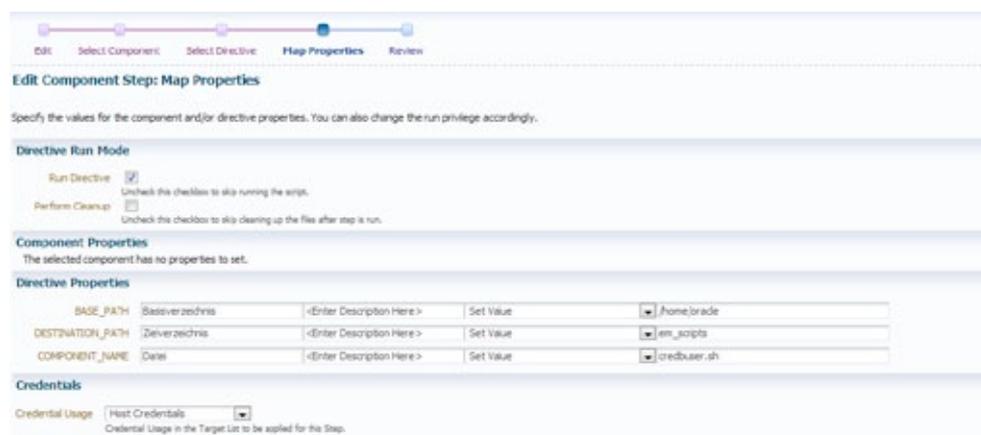


Abbildung 3: Zielverzeichnis für den Kopiervorgang festlegen

Ralf Durben
Ralf.Durben@oracle.com



Tipps und Tricks aus Gerds Fundgrube

Heute: Visuelles Attribut beliebiger Records ändern

Gerd Volberg, OPITZ CONSULTING GmbH

Blöcke, in denen die Aufmerksamkeit auf bestimmte Datensätze mithilfe der Hintergrundfarbe gesteigert wird, sind sehr einfach realisierbar.

Im aktuellen Beispiel (siehe [Abbildung 1](#)) werden alle Datensätze von Mitarbeitern, die seit mindestens drei-
 ßig Jahren im Unternehmen sind, grün eingefärbt. Dazu sind ein „POST
 QUERY“-Trigger auf dem EMP-Block
 sowie eine Prozedur „Set_Record_VA“,
 die einen einzelnen Datensatz mit
 einem neuen visuellen Attribut ver-
 sorgt, erforderlich. Voraussetzung ist,
 dass ein entsprechendes Visual Attri-
 bute namens „VA_GRUEN“ mit grünem
 Hintergrund in der Maske existi-
 tiert.

[Listing 1](#) zeigt das POST QUERY auf
 „EMP“ und [Listing 2](#) die Prozedur zum
 Ändern des visuellen Attributs.

Diese Hilfsprozedur wurde schon
 in der DOAG News, Ausgabe 1/2009,
 in einem anderen Kontext vorgestellt.
 Das zeigt einmal mehr, wie wichtig sol-
 che Hilfsfunktionen sind, die immer in
 einer zentralen PL/SQL-Library gespei-
 chert werden sollten.

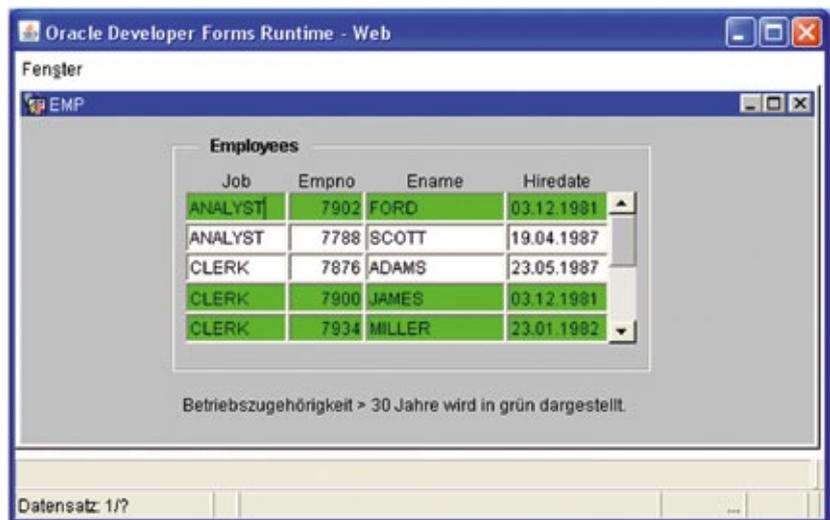


Abbildung 1: Das Beispiel

```
IF (SYSDATE - :EMP.HIREDATE) / 365.25 >= 30 THEN
    Set_Record_VA (P_Block => 'EMP',
                  P_VA    => 'VA_GRUEN');
END IF;
```

Listing 1

```
PROCEDURE Set_Record_VA (P_Block IN VARCHAR2, P_VA IN VARCHAR2) IS
    V_Field  VARCHAR2 (30);
    V_Item   VARCHAR2 (61);
BEGIN
    V_Field := Get_Block_Property (P_Block, FIRST_ITEM);
    V_Item  := P_Block || '.' || V_Field;

    WHILE V_Field IS NOT NULL
    LOOP
        IF Get_Item_Property (V_Item, ITEM_TYPE) IN
            (Const.typ_Text_Item, Const.typ_List_Item) THEN
            Display_Item (V_Item, P_VA);
        END IF;
        V_Field := Get_Item_Property (V_Item, NEXTITEM);
        V_Item  := P_Block || '.' || V_Field;
    END LOOP;
END;
```

Listing 2

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



„Jede Frau kann in der IT mit ihren speziellen Stärken ihren Platz finden ...“

In der Rubrik „Frauen in der IT“ stellt die DOAG News Frauen vor, die erfolgreich im IT-Bereich arbeiten. Ziel ist es, mehr Frauen für die IT-Berufe zu interessieren und ihnen dort auch eine Arbeitsumgebung anzubieten, die Familie und Berufe besser vereinbaren lässt.

Welchen Beruf üben Sie aus?

Heimsath: Ich bin seit zehn Jahren selbstständig und berate Unternehmen rund um die Datenbank. Mein Steckenpferd ist dabei die Apex-Entwicklung.

Auf welchem Weg sind Sie dorthin gekommen?

Heimsath: Eigentlich nicht geplant, sondern indem ich immer meinen Neigungen gefolgt bin. Ich wusste, dass mich die Arbeit mit Computern reizt und mir mathematische Problemstellungen und Logik Spaß machen. Daher lag das Fach „Informatik“ nahe; ich wollte es aber mit einem zusätzlichen Fach kombinieren. Da es mich nach dem Abitur in die Ferne zog, habe ich mich in Großbritannien für verschiedene Studiengänge beworben. Unter anderem auch für „Informatik und Astronomie“. Am Ende hatte die Universität Warwick das beste Angebot und ich entschied mich, dort mein Studium zu beginnen.

Nach dem Diplom habe ich dreieinhalb Jahre bei Oracle Consulting und ein Jahr in der Produktentwicklung bei der Diamos AG gearbeitet. Dann wurde ich gefragt, ob ich nicht Lust hätte, mich innerhalb eines Verbundes selbstständig zu machen. Dies war keine leichte Entscheidung. Einerseits hatte ich etwas Angst, aber andererseits fand ich es unheimlich reizvoll, Anteilseigner eines Unternehmens zu werden und die Selbstständigkeit auszuprobieren. Die Aussicht, zwar selbstständig, aber nicht völlig auf mich allein gestellt zu sein, gab schließlich den Ausschlag, „its-people“-Gesellschafterin (www.its-people.de) zu werden. Und das bereue ich bis heute

nicht – ich mache genau den Job, den ich will.

Was hat Sie motiviert, diesen Beruf zu ergreifen?

Heimsath: Daran haben sicher meine Eltern einen großen Anteil. Sie haben mir nie eingeredet, dass etwas Technisches nichts für Mädchen sei. Während der Schulzeit habe ich mit meinem Vater zusammen DOS-Kurse besucht und ich durfte ätzen, löten und programmieren – damals noch in BASIC. Als dann unsere Schulklasse durch die Berufsberatung vom Arbeitsamt geschleust wurde, wusste ich zum Glück schon, was ich (nicht) wollte.

Wie sehen Sie generell die Rolle der Frau in der IT?

Heimsath: Das ist schwer zu sagen, wenn man Generalisierungen vermeiden will. Angenommen, die weiblichen und männlichen Gehirne sind unterschiedlich verdrahtet – und darauf weist ja einiges hin –, dann sollte man bei der Zusammenstellung eines Teams kein Potenzial verschenken und für eine gute Mischung sorgen. Aber dabei möchte ich nicht die „Quotenfrau“ aus Prinzip sein, sondern die technischen und persönlichen Voraussetzungen müssen natürlich ebenfalls passen.

Bietet die IT-Branche für Frauen die Möglichkeit, ihre Stärken einzusetzen?

Heimsath: Da es in diesem Bereich nicht auf Muskelkraft ankommt, würde ich diese Frage ganz klar mit „ja“ beantworten. Jede Frau kann in der IT mit ihren speziellen Stärken ihren Platz finden. Die Bandbreite ist riesig, und das macht die IT ja gerade so interessant.

Was könnte Frauen motivieren, einen Beruf in der IT zu ergreifen?

Heimsath: Wichtig ist, dass Mädchen nicht so früh im Denken eingeschränkt werden, sodass Alternativen im IT-Bereich überhaupt wahrgenommen werden können. Nur dann hat man die Möglichkeit, ihnen im entsprechenden Alter unterschiedliche Aspekte der IT-Berufe nahezubringen – zum Beispiel die Chance, durch den geschickten Einsatz der IT Menschen bei ihrer Arbeit zu unterstützen.

Welche Eigenschaften sollte eine Frau mitbringen, um sich in der IT-Branche durchzusetzen?

Heimsath: Ein gewisser Humor hilft auf jeden Fall! Ich habe als (fast) einzige Frau in den naturwissenschaftlichen Kursen zwar durchaus doofe Sprüche bekommen, aber auch gerne gekontert. Abgesehen davon, dass man eine Strategie haben sollte, wie man das Geschlechterthema handhabt, sollte man Spaß an Teamarbeit und an kniffligen Aufgabenstellungen haben.

Was kann eine Anwendervereinigung wie die DOAG tun, damit mehr Frauen in die IT kommen?

Heimsath: Ich sehe das als einen sehr langfristigen Prozess. Die Weichenstellung liegt dabei sicherlich in der Kindheit/Schulzeit. Kindern muss vermittelt werden, dass ihnen jeder Bereich offen steht. Das ist jedoch schwierig in Zeiten, in denen es die Industrie darauf anzulegen scheint, aus jedem Mädchen eine Barbie oder eine Prinzessin zu machen. Die DOAG könnte Antworten finden zu verschiedenen Fragen: „Wie bringt man die komplexen Berufsbilder der IT jemandem ohne

Vorkenntnisse nahe?“, „Kann man das an einem „Girls‘ Day“ im Jahr schaffen?“ oder „Braucht es andere Konzepte?“

Was erwarten Sie von einem IT-Unternehmen wie Oracle?

Heimsath: Generell sollten alle Angestellten – unabhängig vom Geschlecht – die Flexibilität bekommen, die sie für ihre Familie brauchen. Präsenz ist zwar wichtig, aber es ist bei den meisten Jobs nicht notwendig, 100 Prozent seiner Arbeitszeit im Büro zu verbringen und dies auch noch in einem starren Zeitfenster. Oracle nimmt bereits am „Girls Day“ teil. Beim vorletzten war ich zufällig anwesend und habe mich sehr gefreut, dass die Mädchen ausgerechnet eine Apex-Anwendung erstellen durften. Auch hier lautet allerdings die Frage „Reicht es aus, einmal im Jahr aktiv zu werden?“

Was wünschen Sie sich für die Zukunft?

Heimsath: Mehr Frauen, die sich trauen, mit its-people den Schritt in die Selbstständigkeit zu wagen. Ich hätte



Zur Person: Sabine Heimsath
Sabine Heimsath begann im Jahr 1992 ihr Studium „Computer & Management Science“ an der University of Warwick und machte ihren Diplom-Abschluss in Wirtschaftsinformatik im Jahr 1999 an der Westfälischen Wilhelms-Universität in Münster. Diverse Tätigkeiten als Hilfskraft im Rechenzentrum und am Lehrstuhl begleiteten das Studium.

Ihr Weg in der Arbeitswelt begann bei Oracle Consulting, wo ihr vor allem die Abwechslung gefiel. Nach einem Ausflug in die Produktentwicklung machte sie sich zusammen mit anderen Freiberuflern als Mitglied im „its-people“-Verbund im Datenbank- und Web-Umfeld selbstständig. Speziell beschäftigt sie sich mit Apex, der objektorientierten Entwicklung in Oracle PL/SQL, objektrelationalen Mappings und Schnittstellenprozessen für Data Warehouses. Ihre Erfahrungen mit Unix, Java, PHP, MS Access und VBA helfen ihr bei der Migration oder Anbindung von bestehenden Systemen. Sabine Heimsath ist verheiratet, hat zwei Kinder, liest alles von c't bis Emma, sucht immer noch ihre Lieblingsportart und ist seit dem Jahr 2011 außerdem auch in der Kommunalpolitik aktiv.

sehr gerne mehr Kolleginnen und es würde mich sehr freuen, wenn es nicht

mehr schick wäre, mit „Mathe konnte ich noch nie“ zu kokettieren.

Die Lizenzierung der Oracle Database 12c

Michael Paege, Leiter des DOAG Competence Center Lizenzierung

Das neue Major Release der Oracle Datenbank bringt leider für die Lizenzierung keine komplett neue Metrik, die Cloud-Ansätze und -Architekturen optimal unterstützt, indem nach Verbrauch von Ressourcen (on demand) oder verarbeiteten Transaktionen oder Datenvolumen abgerechnet wird.

Zeitgleich zu 12c kam dennoch am 25. Juni 2013 eine neue Tech-Preisliste heraus. Darin findet man weiterhin die bekannten Metriken „Named User Plus“ und „Prozessor“, aber eben keine neue Metrik. Bei den Optionen der Datenbank Enterprise Edition gibt es nun die Option „Multitenant“, die die Features der Pluggable Database enthält.

Dennoch hat sich Oracle ein bisschen in Richtung Cloud bewegt, denn Oracle und Microsoft haben eine Kooperation in Sachen Cloud Computing angekündigt. Oracle-Kunden können ihre Software nun auf der Microsoft-Windows-Plattform Azure betreiben – nicht nur technisch, sondern Oracle hat hierzu auch seine Lizenzbedingungen angepasst, indem nämlich Azure in das Dokument „Licensing Oracle Software in Cloud Computing Environments“ aufgenommen wurde, das bisher nur die Amazon Elastic Compute Cloud (E2C) und Amazon Simple Storage Service (S3) beinhaltete. Darin steht, dass bei diesen Cloud-Architekturen die virtuell zu-

gewiesenen Cores als physikalische Cores gelten und auch nur diese lizenziert werden müssen. Es sind auch die Regeln für die Produkte mit „Standard Edition One“ und „Standard Edition“ für diese Cloud-Architekturen genannt: hier gelten jeweils vier zugewiesene Cores als ein Socket beziehungsweise zu lizenzierender Prozessor.

Im Bereich der private Clouds, die Firmen üblicherweise mittels Virtualisierungslösungen erstellen, gibt es keine Änderungen. Virtualisierungsplattformen wie VMWare, HyperV oder Xen sind nach wie vor als Softpartitioning klassifiziert, sodass weiterhin der gesamte Cluster bezüglich Oracle lizenziert werden muss.

Die Netzwerkverschlüsselung wurde allerdings aus der Advanced Security Option genommen und in die lizenzkostenpflichtigen Editionen (Standard Edition One, Standard Edition und Enterprise Edition) überführt. Im Datenbank-12c-Launch-Event wurde gesagt, dass dies für alle Editionen und auch für ältere Releases gilt, die dieses Feature technisch unterstützen.

Weitere Informationen zur Lizenzierung der Datenbank 12c: http://docs.oracle.com/cd/E16655_01/license.121/e17614/editions.htm



Die Oracle-Community trifft sich auf der DOAG 2013 Konferenz + Ausstellung

Das größte Oracle-Event in Europa findet in diesem Jahr zum 26. Mal statt. In rund 400 Fachvorträgen präsentieren vom 19. bis 21. November 2013 in Nürnberg die weltweit besten Experten praktische Erfahrungen und neueste Informationen. Die Veranstaltung bietet die ideale Voraussetzung für den erfolgreichen Einsatz der Oracle-Produkte.

Ein besonderes Merkmal der DOAG Konferenz + Ausstellung sind die zahlreichen Networking-Elemente, die einen optimalen Austausch unter den 2.000 erwarteten Besuchern garantieren. Darüber hinaus besteht die Möglichkeit zu einem persönlichen Gespräch mit erfahrenen Spezialisten aus der Oracle-Welt und führenden Produktmanagern des Herstellers.

Stream „Oak Table“ dreht jedes Bit um

Sie sind moderne Ritter der Tafelrunde, eine Gruppe von Oracle-Besessenen auf der Suche nach dem binären Gral. Sie sind Mitglieder des „Oak Table“. Zur DOAG Konferenz + Ausstellung kommen die Oakies erstmalig mit fünfzehn Mann aus allen Ecken der Welt, um zusammen mit den Konferenz-Besuchern die Datenbank von oben nach unten zu kehren und jedes Bit einzeln umzudrehen.

Der Oak Table existiert seit dem Jahr 2002. Es begann an einem kalten Januartag, als sich eine Clique von Oracle-Verrückten in einem Haus in dem ruhigen, nahe Kopenhagen gelegenen Måløv, zu einem informellen Master Class versammelte. Alle zwanzig Mann saßen am Küchentisch des Dänen Morgens Nørgaard, der bereits zum zweiten Mal in Folge einlud. „Ich fragte an einem Abend, ob wir dieser Versammlung nicht einen Namen geben sollten“, berichtet der Däne in dem Buch Oracle Insights: Tales of the Oak Table. Auf einmal war es wie selbstverständlich: Saßen sie nicht an diesem wunderbaren, mächtigen Eichentisch? Im Gegensatz zu König Artus' Tafel ist das Möbelstück übrigens nicht rund, sondern viereckig und so schwer, dass die

Kraft von vier bis sechs Mann benötigt wird, um ihn zu Veranstaltungen zu transportieren. Zur Konferenz kommen die Oakies ohne Tisch, dafür mit reichlich Man-Power.

Keynote von Peter Kreuz und die Anstiftung zum Querdenken

Er gehört zu einer neuen Generation von Vordenkern in Wirtschaft und Management. „Bildreich und frech plädiert er dafür, anders zu denken und wieder Mut, Spaß und Leidenschaft in den Wirtschaftsalltag zu bringen“, schreibt das Manager Magazin über den diesjährigen Keynote-Speaker Dr. Peter Kreuz. Entwicklern, Administratoren und allen IT-Experten soll dieser Vortrag den Horizont für eine neue Art zu leben und zu arbeiten öffnen.

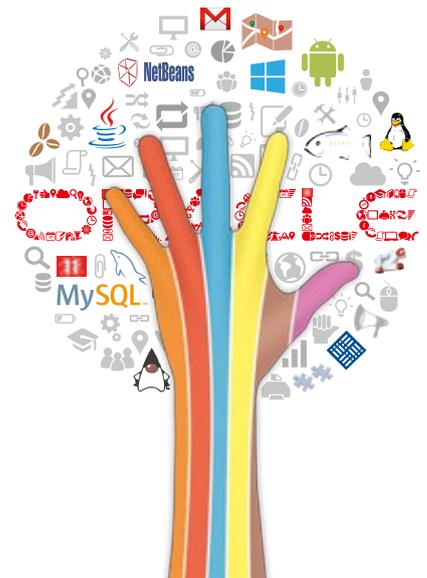
Es geht um den Abschied von alten Reiseführern, überholten Gewissheiten und ausgetrampelten Pfaden. Für Menschen, die etwas bewegen wollen in ihrer Welt und in ihrem Unternehmen. Mit Engagement, Hunger nach Veränderung und dem unbedingten Willen, etwas Besonderes zu schaffen. Die Botschaft des Vortrags lautet: Sucht die Chance, nicht die Hindernisse! Brecht die Regeln! Gebt mehr als ihr nehmt! Seid besser die erstklassige Version eurer selbst, als die zweitklassige Version eines anderen. Im Anschluss an die Keynote findet eine interaktive Diskussionsrunde statt.

Die Datenbank ist noch lange nicht am Ende

Im Fokus der DOAG 2013 Konferenz + Ausstellung steht die neue Oracle Datenbank 12c. Sie bietet einige innovative Features, auf die manche

Unternehmen bereits ungeduldig gewartet haben. Schon lange ist in der Oracle-Community bekannt, dass es sogenannte „Pluggable Databases“ geben wird. Das zweite große Schlagwort heißt „nutzungsorientierte Optimierung der Daten“. Dieses Feature sorgt automatisch dafür, dass Daten – je nach Aktualität mehr oder weniger komprimiert – in entsprechenden Speicherklassen gehalten werden. Highlights der 12c-Präsentationen sind die Keynote von Andrew Mendelsohn, Oracle Senior Vice President und Leiter der Datenbank-Entwicklung, mehr als dreißig spezielle 12c-Vorträge sowie praktische Vorführungen im Oracle Demo-Kino.

Das komplette Programm der DOAG Konferenz + Ausstellung, die Anmeldung sowie weitere Informationen gibt es unter <http://2013.doag.org>



Wir begrüßen unsere neuen Mitglieder

Persönliche Mitglieder

Torsten Grünen
Jörg-Christian Müller
Sebastian Winkler
Günter Neupert
Brigitte Egger
Fred Telaar

Jörg Pekruhl
Ulrich Werning
Bernhard Becker
Graeme Gerber
Björn Behrens

Firmenmitglieder

DataSquare GmbH, Joachim Misdorf
Landesforsten RLP, Christine Sohn
nova ratio AG, Siegward Sanden
IT-P GmbH, Frank Grossheim
ATLAS Dienstleistungen für Vermö-
gensberatung GmbH, Michael Kuhn
PAPSTAR GmbH, Ralf Schneider



Dr. Dietmar Neugebauer
Vorstandsvorsitzender der DOAG

Die Zukunft der DOAG

Auf seiner ersten Vorstandssitzung nach der Sommerpause diskutierte der DOAG-Vorstand am 6. und 7. September 2013 in Hamburg die Ziele und Kern-Botschaften des Vereins für die kommenden Jahre. Basierend auf den Beschlüssen der Delegiertenversammlung vom Juni dieses Jahres wurden folgende Schwerpunkte festgelegt:

- Die DOAG kommuniziert die Herausforderungen der Oracle-Community und Oracle hört der DOAG besser zu (verantwortlich: Dr. Dietmar Neugebauer, Christian Trieb)
- DOAG Konferenz + Ausstellung, DOAG Applications und JavaLand sind führende Konferenzen in Europa. Daneben finden spezialisierte oder regionale Veranstaltungen für den deutschsprachigen Raum statt (verantwortlich: Fried Saacke, Dr. Frank Schönthaler)
- Eine Redaktion für Online und Print berichtet aktuell und zielgruppenorientiert (verantwortlich: Robert Szilinski, Björn Bröhl)

- Die DOAG ist als Wissensvermittler bekannt und insbesondere an Hochschulen präsent (verantwortlich: Urban Lankes, Dr. Frank Schönthaler)
- Die DOAG ist mit den User Groups der Nachbarländer eng vernetzt und fördert einen von Oracle unabhängigen Erfahrungsaustausch der europäischen User Groups (verantwortlich: Dr. Dietmar Neugebauer, Christian Trieb)

Zu allen Punkten werden die Verantwortlichen auf der nächsten Vorstandssitzung Maßnahmen vorstellen, wobei die bereits auf der Delegiertenversammlung gesammelten Ideen mit einfließen. Darüber hinaus legte der Vorstand den Termin für die nächste Delegiertenversammlung fest. Diese wird am 14. und 15. März 2014 in Düsseldorf stattfinden.



Robert Szilinski
Leiter der Development Community

Die Development Community stellt sich für die Zukunft auf

Nachdem Stefan Kinnen sein Amt als Community-Leiter zur letzten Delegiertenversammlung abgegeben hat, freue ich mich, als neuer Vorstand zukünftig die Development Community zu verantworten und im Sinne aller Mitglieder weiterzuentwickeln. Für das Jahr 2013 haben wir uns viel vorgenommen: mehr Interaktion und das Networking untereinander zu fördern, Trends wie „Mobile“ und „Agilität“ stärker aufzugreifen sowie etablierte Themen wie Apex oder ADF weiter auszubauen.

Auch wenn das Jahr noch nicht zu Ende ist, kann man doch feststellen, dass wir richtig in Bewegung sind und sowohl personell als auch inhaltlich bereits jetzt viel erreicht haben. So fand im Sommer die zweite Developer-Konferenz DOAG 2013 Development mit dem Schwerpunkt „Agilität“ statt, bei der schon am Vorabend beim BBQ das Netzwerk gepflegt werden konnte. Auch personell hat sich etwas getan: Seit kurzem freue ich mich, dass wir unsere Community mit Ulrich Gerkmann-Bartels (ADF) und Niels de Bruijn (Apex) gleich mit zwei neuen Themen-Verantwortlichen verstärken konnten. Zudem ist eine enge Kooperation mit der Oracle ADF Community angelaufen. Für das Trendthema „Mobile“, das übrigens auch ein Schwerpunkt auf der diesjährigen DOAG Konferenz + Ausstellung sein wird, planen wir analog dazu ebenfalls, ein aktives Netzwerk in der DOAG zu etablieren.

Die größten Neuerungen wird es im Bereich der Veranstaltungen geben. Sicherlich habe viele schon vom „JavaLand“ gehört, mit dem wir am 25. und 26. März 2014 gemeinsam mit dem Interessenverbund der Java User Groups e.V. (iJUG) und anderen Part-

nen eine neue Java-Konferenz etablieren werden. Dafür setzen wir auf ein neues Format mit vielen spannenden Attraktionen und herausragenden Referenten im Phantasialand in der Nähe von Köln.

Neben dieser großen Konferenz wird die DOAG darüber hinaus am 30. Januar 2014 das erste Mal ein Development Barcamp veranstalten, also eine Fachkonferenz, die kein vordefiniertes Programm und keine festen Streams hat. Dafür gibt es Interaktion und einen intensiveren Austausch über aktuelle Themen, denn das Programm legen alle Teilnehmer zu Beginn gemeinsam fest. Ich selbst habe bereits mehrfach an Barcamps außerhalb der DOAG teilgenommen und war jedes Mal absolut begeistert, was sich entwickeln kann, wenn „Teilnehmer“ zu „Teilgebern“ werden und Wissen in der Community (inter)aktiv untereinander ausgetauscht wird.

In diesem Sinne möchte ich die Arbeit in der Community fortsetzen und freue mich auf den Input aller Mitglieder, die mit Ideen und Engagement die Development Community weiter in Bewegung halten möchten.



*Christian Trieb
Leiter Datenbank Community*

Die etwas andere Österreichische ORACLE- Anwenderkonferenz

Anlässlich des 25-jährigen Jubiläums der Austrian Oracle User Group (AOUG) dauerte die diesjährige Konferenz in Wien zwei Tage. Dafür hatte sich die Anwendergruppe etwas Besonderes ausgedacht: Die Konferenz fand nicht wie in den Jahren zuvor in einem Konferenzzentrum statt, sondern

in den ehemaligen Produktionsräumen einer Brauerei (Bier gab es erst abends). Damit unterschied sich das Ambiente angenehm von der einheitlichen Funktionalität moderner Tagungszentren.

Die Konferenz gliederte sich in die fünf Streams „Technologie“, „Development“, „BI / Big Data“, „University“ und „Führungskräfte“. Die Keynote hielt der blinde Extrembergsteiger Andreas Holzer. Es war beeindruckend mitzuerleben, wie er trotz seines Handicaps die höchsten Berge erklimmt.

Im Stream „Führungskräfte“ wurden sehr interessante Vorträge ohne Technik- beziehungsweise Oracle-Bezug gehalten. Der Zuspruch war entsprechend gut. In den anderen Streams waren Oracle und Partner mit vielfältigen technischen Präsentationen vertreten.

Am Abend des ersten Tages fand die Jubiläumsfeier statt. DOAG-Vorstand Christian Trieb überbrachte die Glückwünsche der DOAG zum Jubiläum.

Die Wahl des Konferenzorts hat die Veranstaltung sicher belebt. Interessante Vorträge, kompetente Referenten und eine vielseitige Ausstellung sorgten dafür, dass die Jubiläumskonferenz gut gelungen war.

Impressum

Herausgeber:

DOAG Deutsche ORACLE-
Anwendergruppe e.V.
Temoelhofer Weg 64, 12347 Berlin
Tel.: 0700 11 36 24 38
www.doag.org

Verlag:

DOAG Dienstleistungen GmbH
Fried Saacke, Geschäftsführer
info@doag-dienstleistungen.de

Chefredakteur (ViSdP):

Wolfgang Taschner, redaktion@doag.org

Redaktion:

Fried Saacke, Carmen Al-Youssef,
Mylène Diacquenod, Dr. Dietmar
Neugebauer, Christian Trieb,
Robert Szilinski

Titel, Gestaltung und Satz:

Alexander Kermas
DOAG Dienstleistungen GmbH

Titelfoto: © Sashkin/Fotolia.com,
raven/Fotolia.com

Foto S. 10: © Tonis Pan/Fotolia.com

Foto S. 18: © Oracle/www.oracle.com

Foto S. 26: © Artsem Martysiuk/Fotolia.com

Foto S. 36: © fixer00/www.oracle.com

Anzeigen:

Simone Fischer, anzeigen@doag.org
DOAG Dienstleistungen GmbH
Mediadaten und Preise finden Sie
unter: www.doag.org/go/mediadaten

Druck:

Druckerei Rindt GmbH & Co. KG
www.rindt-druck.de

Neu: Virtual Compute Appliance

Die Familie der Engineered Systems hat Zuwachs bekommen: Die Oracle Virtual Appliance an eignet sich zur schnellen und einfachen Virtualisierung und ermöglicht einen schnellen und wiederholbaren Infrastruktur-Aufbau, der laut Oracle in weniger als einer Stunde betriebsbereit ist. Die Appliance wurde so konzipiert, dass unter Oracle VM (OVM) mehrere Applikationen und verschiedene Betriebssysteme gleichzeitig auf der Infrastruktur-Plattform laufen, dabei werden Solaris, Oracle Linux sowie andere Linux-Distributionen neben Microsoft Windows unterstützt. Die Software unterstützt bis zu 128 vCPUs.



14./15.10.2013
Berliner Expertenseminar:
 „SOA Suite 11g Deep Dive Best Practices
 für Entwickler und Administratoren mit
 Markus Lohn“
 Cornel Albert
 expertenseminare@doag.org

16.10.2013
Regionaltreffen NRW
 Stefan Kinnen, Andreas Stephan
 regio-nrw@doag.org

17.10.2013
Regionaltreffen Karlsruhe
 Reiner Bünger
 regio-karlsruhe@doag.org

17.10.2013
SIG Database
 Johannes Ahrends, Christian Trieb
 sig-database@doag.org

17.10.2013
Regionaltreffen Stuttgart
 Jens-Uwe Petersen
 regio-stuttgart@doag.org

17.10.2013
Regionaltreffen Nürnberg/Franken
 André Sept, Martin Klier
 regio-franken@doag.org

24.10.2013
Regionaltreffen München/Südbayern
 Andreas Ströbel
 regio-muenchen@doag.org

Aktuelle Termine und
 weitere Informationen finden Sie unter
www.doag.org/termine/calendar.php

07.11.2013
Regionaltreffen Rhein-Neckar
 Frank Stöcker
 regio-rhein-neckar@doag.org

14.11.2013
Regionaltreffen Trier/Saarland/Luxemburg
 Bernd Tuba, Holger Fuchs
 regio-trier@doag.org

19-21.11.2013
DOAG 2013 Konferenz + Ausstellung
 DOAG Geschäftsstelle
 office@doag.org

22.11.2013
DOAG Schulungstag
 DOAG Geschäftsstelle
 office@doag.org



03.12.2013
Regiotreffen Jena/Thüringen
 Jörg Hildebrandt
 regio-thueringen@doag.org

03.12.2013
Nordlichtertreffen
 Stefan Thielebein
 regio-nord@doag.org

04.12.2013
Regionaltreffen Würzburg
 Oliver Pyka
 regio-wuerzburg@doag.org

04.12.2013
Regionaltreffen Berlin/Brandenburg
 Michel Keemers
 regio-bb@doag.org

09.12.2013
Regionaltreffen München/Südbayern
 Andreas Ströbel
 regio-muenchen@doag.org

09.12.2013
**Regionaltreffen Osnabrück/Bielefeld/
 Münster**
 Andreas Kother, Klaus Günther
 regio-osnabrueck@doag.org

10.12.2013
Regionaltreffen Bremen
 Ralf Kölling
 regio-bremen@doag.org

10.12.2013
Regionaltreffen Hannover
 Andreas Ellerhoff
 regio-hannover@doag.org

10.12.2013
Regionaltreffen Nürnberg/Franken
 André Sept, Martin Klier
 regio-franken@doag.org

Unsere Inserenten

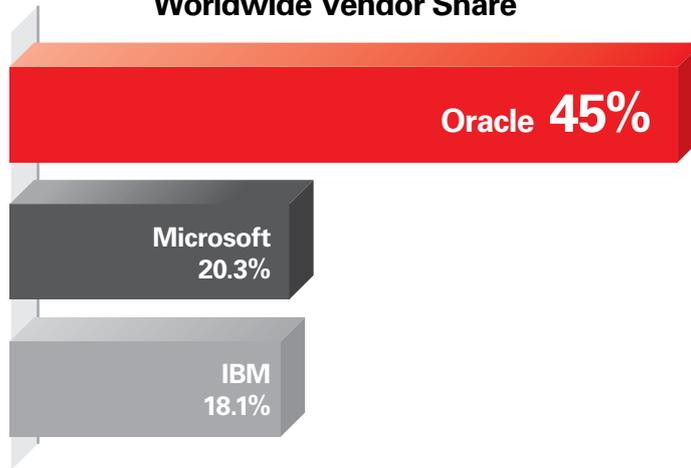
| | |
|---|-------|
| areto gmbh www.areto-consulting.de | S. 11 |
| DBConcepts www.dbconcepts.at | S. 39 |
| Hunkler GmbH & Co. KG www.hunkler.de | S. 3 |
| Libelle AG www.libelle.com | S. 35 |
| MuniQsoft GmbH www.muniqsoft.de | S. 17 |
| OPITZ CONSULTING GmbH www.opitz-consulting.com | U 2 |
| ORACLE Deutschland B.V. & Co. KG www.oracle.com | U 3 |
| Promatis software GmbH www.promatis.de | S. 29 |
| regio iT www.regioit.de | S. 25 |
| Trivadis GmbH www.trivadis.com | U 4 |

STILL

#1

Database

Worldwide Vendor Share



Oracle Database

Trusted by 308,000 Customers Worldwide

ORACLE®

oracle.com/database
or call 0800 1 81 01 11

Schön, wenn man Zeit gewinnt.



■ Mehr Effizienz für Ihr Unternehmen. Aber auch mehr Zeit für Sie persönlich. Mit unseren IT-Lösungen erhalten Sie beides. 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 Business Intelligence, Application Development, Infrastructure-Engineering sowie Training und Betrieb. Sprechen Sie mit uns über Ihre Anforderungen.

Wir nehmen uns Zeit für Sie. www.trivadis.com | info@trivadis.com

ZÜRICH ■ BASEL ■ BERN ■ BRUGG ■ LAUSANNE ■ DÜSSELDORF ■ FRANKFURT A.M.
FREIBURG I.BR. ■ HAMBURG ■ MÜNCHEN ■ STUTTGART ■ WIEN

trivadis
makes IT easier. ■ ■ ■