



Fried Saacke
Vorstand und Geschäftsführer
der DOAG

Liebe Mitglieder der Deutschen ORACLE-Anwendergruppe,
liebe Leserinnen und Leser,

die Vorstellung der neuen Fusion Middleware 11g war ein voller Erfolg. Das Produkt zählt für mich zu den wichtigsten Innovationen aus dem Hause Oracle, für das sich die lange Wartezeit gelohnt hat. Oracle hat es jetzt geschafft, eine allen Ansprüchen gerecht werdende Middleware-Lösung zur Verfügung zu stellen. Skalierbarkeit, Performance, Offenheit und Integrierbarkeit sind aus Sicht der Anwender wichtige Bausteine, mit denen Oracle die Marktführerschaft bei den Middleware-Lösungen erreichen könnte.

Jürgen Kunz, Geschäftsführer der ORACLE Deutschland GmbH, hat in unserem aktuellen Interview (siehe Seite 9) betont, dass die Kunden mit der neuen Fusion Middleware Prozesse Schritt für Schritt in eine moderne Architektur überführen können. Die Projekte sind auf diese Weise schnell umsetzbar, weniger risikobehaftet und gut zu kalkulieren. Die DOAG wird auf ihren Veranstaltungen den Wissenstransfer zu diesem Thema fördern.

Fusion Middleware steht natürlich auch auf der umfangreichen Agenda der DOAG 2009 Konferenz + Ausstellung. Gerade erst haben wir aus mehr als 550 Bewerbungen die 300 besten Vorträge zu einem hochinteressanten Konferenzprogramm zusammengestellt. Von daher freue ich mich ganz besonders, Sie auf dem größten Event der Oracle-Community in Zentraleuropa von 17. bis 19. November 2009 in Nürnberg begrüßen zu dürfen.

Ich wünsche Ihnen viel Erfolg bei der Arbeit mit Ihren Oracle-Produkten.

Ihr



**Schneller, höher, sicherer: Remote Administration Service (RAS)
von Hunkler – der effizienteste Fitness-Check für Oracle**

ORACLE CERTIFIED ADVANTAGE
PARTNER

- Optimale Konfiguration Ihrer Oracle-Datenbanken
- Fernüberwachung der Performance
- Früherkennung und Behebung von Fehlerquellen und Systemstörungen
- Zugang über VPN
- Telefonischer Support
- Individuell gestalteter Leistungsumfang

Best Solutions Based on Oracle
HUNKLER
GmbH & Co. KG

Hauptsitz Karlsruhe
Geschäftsstelle Bodensee

Bannwaldallee 32
Fritz-Reichle-Ring 2

76185 Karlsruhe
78315 Radolfzell

Tel. 0721-490 16-0
Tel. 07732-939 14-00

Fax 0721-490 16-29
Fax 07732-939 14-04

info@hunkler.de
www.hunkler.de

Aus der DOAG

- 5 Spotlight
- 6 Die DOAG in der Presse
- 9 Interview mit Jürgen Kunz
„Wir haben im Vergleich zum Wettbewerb sehr gut abgeschnitten ...“
- 26 Wichtige Informationen zu aktuellen Lizenzfragen
Michael Paege

Fusion Middleware

- 11 Die neue Fusion Middleware 11g
Wolfgang Taschner
- 14 Migrationsstrategie für Kernbestandteile der Fusion Middleware
Georg Molter, Christian Kücherer und Matthias Pfau
- 19 Entwicklung von Benutzeroberflächen mit ADF Faces
Detlef Müller
- 24 Überwachung von BPEL-Prozessen mittels Business Activity Monitoring
Gert Schüßler

Datenbank

- 27 RMAN, Toad und LiteSpeed
Johannes Ahrends
- 30 Variable Datenselektion für OLTP- und OLAP-Systeme
Volker Rieke
- 34 Entfernte Dateien mit SSHFS zentral und sicher verwalten
Christian Ballweg
- 36 Kostensenkungen und Effizienzsteigerungen im Oracle Datenbank-Umfeld mittels Capacity Management unter ITIL und ISO20000
Christian Wischki und Peter Stalder
- 41 Automatisierte Backup-Überprüfung mit dem Recovery-Katalog
Björn Wolter
- 44 Replikation mit Oracle Streams
Martin Bach
- 47 Elegante und effiziente Umsetzung der Best Practices mit dem Oracle Data Integrator
Karol Hajdu

Entwicklung

- 55 Effiziente Historisierung großer Tabellen
Tobias Stark
- 58 Mehr als Metadaten: Das Application Express Repository
Carsten Czarski
- 62 Oracle ADF 11g – Komponenten zur Daten-Visualisierung
Luc Bors und Hans-Chr. Kaller
- 67 DBMS_COMPARISON zum asynchronen Vergleich von Tabellen
Dr. Elke Fritsch
- 71 ECM-Performance und Oracle
Sebastian Wenzky

Aktuell

- 32 Die DOAG beim Beta-Test von 11gR2 der User Groups in Redwood Shores
Michael Pfautz
- 43 Wegweiser durch Oracle
Till Brügelmann

Tipps & Tricks

- 76 Heute: Wie unterbreche ich in Forms einen Loop?
Gerd Volberg

Aus der DOAG

- 12 Impressum
- 40 Inserentenverzeichnis
- 46 DOAG-Regionaltreffen in München
Franz Hüll
- 61 ITIL-Prozesse optimieren und den Betrieb effizient umsetzen
- 77 Oracle Exadata – ein Blick hinter die Kulissen
Christian Trieb
- 78 Critical Patch Update – eine Erfolgsstory?
Franz Hüll
- 80 Berliner Experten-Seminare
- 81 DOAG Termine
- 82 Wir begrüßen unsere neuen Mitglieder



„Wir haben im Vergleich zum Wettbewerb sehr gut abgeschnitten ...“

Interview mit Jürgen Kunz, Senior Vice President und Geschäftsführer der ORACLE Deutschland GmbH

Seite 9



Die neue Fusion Middleware 11g

Am 2. Juli 2009 stellte Oracle in München das von vielen Anwendern erwartete Release 11g der Oracle Fusion Middleware vor. Dazu gehören neue innovative Funktionen in der SOA Suite, der WebLogic Suite, der WebCenter Suite und beim Identity Management.

Seite 11



Kostensenkungen und Effizienzsteigerungen im Oracle Datenbank-Umfeld

Die Lösung stellt das unter ITIL und ISO/IEC20000 beschriebene Capacity Management dar.

Seite 36



Spotlight

Dienstag, 4. Mai 2009

Fried Saacke, Vorstand und Geschäftsführer der DOAG, besucht die Collaborate 09 in Orlando, Florida (USA). Das größte und bedeutendste Event der amerikanischen Oracle-Usergroups ist mit 4.300 Teilnehmern zwar deutlich weniger besucht als im Vorjahr, dennoch herrscht eine sehr konstruktive Stimmung.

Mittwoch, 5. Mai 2009

Im Rahmen der Collaborate 09 kommen Fried Saacke und vier weitere Usergroup-Vertreter mit Oracle-President Charles Phillips zu einem kurzen Austausch zusammen. Bei dieser Gelegenheit kommt die Enttäuschung der Anwendergruppen zum Ausdruck, dass Oracle immer weniger Geld im Umfeld der Usergroups ausgibt. Charles Phillips begründet das mit der gestiegenen Zahl der Anwendergruppen. Er betont gleichzeitig, dass er die Arbeit der Usergroups sehr schätzt.

Dienstag, 12. Mai 2009

Die DOAG-Fachkonferenz „Logistik & SCM 2009“ ist mit mehr als doppelt so vielen Teilnehmern als im Vorjahr ein großer Erfolg. Besonders erfreulich ist die hohe Zahl an Endanwendern.

Montag, 25. Mai 2009

Bei einem Treffen mit Vertretern von Sun wird eine gemeinsame Roadshow von DOAG und Sun geplant, um die Anwender von Sun über die Möglichkeiten der Oracle-Technologie aufzuklären.

Mittwoch, 3. Juni 2009

Die beiden Veranstaltungen „Oracle 11g Performance Forecast“ und „Oracle 11g Performance Firefighting“ der ersten Berliner Experten-Seminare waren komplett ausgebucht. Die DOAG wird die erfolgreiche Veranstaltungsreihe bereits im September 2009 mit den Themen „Advanced Oracle Objekt Monitoring“ und „Advanced Oracle Datenbank Monitoring“ fortsetzen.

Freitag, 19. Juni 2009

Der Vorstand der DOAG beschließt auf seiner Vorstandssitzung in Berlin die Gründung der DOAG Konferenz und Ausstellung GmbH. Diese neue Gesellschaft wird als Tochterunternehmen der DOAG Dienstleistungen GmbH die betriebswirtschaftliche Abwicklung der jährlich im November stattfindenden Anwenderkonferenz übernehmen.

Samstag, 20. Juni 2009

Im Rahmen der Vorstandssitzung werden mit Günther Stürner, Vice President Server Technologies und Sales Consulting der ORACLE Deutschland GmbH, umfangreiche neue Projekte besprochen. Ein Schwerpunkt liegt auf dem Thema „Java“.

Montag, 6. Juli 2009

Im Rahmen eines Presse-Roundtables stellt die DOAG die diesjährige Umfrage zum Thema „Oracle-Lösungen – Chancen und Risiken für Unternehmen“ vor und diskutiert die Ergebnisse mit Vertretern der Fachpresse und dem Management von Oracle.

Montag, 13. Juli 2009

Nach Gummersbach und Karlsruhe gründet die DOAG in Münster die dritte Hochschul-Regionalgruppe. Im Herbst folgen die Hochschulen und Ulm und München. Damit besitzt die Hochschul-Community der DOAG eine hervorragende Basis.

Freitag, 17. Juli 2009

Das Call for Presentation für die DOAG 2009 Konferenz + Ausstellung war ein voller Erfolg. Die Streamverantwortlichen stellen aus mehr als 550 Vortragsbewerbungen ein attraktives Programm zusammen, das der Vorstand auf seiner Sitzung verabschiedet.



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.

**Computerwoche online
vom 15. Juli 2009:**

Was Oracle-Kunden bewegt

Anwender von Oracle-Produkten interessieren vor allem die Themen Virtualisierung, Compliance, Business-Intelligence und die Automatisierung des IT-Betriebs, so die Deutsche Oracle Anwendergruppe e.V. (DOAG).

Von einer Automatisierung beispielsweise des Datenbankbetriebs erhoffen sich Firmen Kosteneinsparungen. Sie erwarten, damit den Anteil der Betriebskosten an den gesamten Ausgaben für IT drücken zu können, um mehr Luft für IT-Innovationen übrig zu haben. Die Situation verschärft sich, weil so manche Firma mit einem geringeren IT-Budget auskommen muss. Die Aufgaben des Lifecycle-Management, sprich Rollout, Überwachung von Wartung von Software sollen weniger manuell erledigt werden.

Gespannt sind die Oracle-Anwender daher auf die Neuerungen der unlängst präsentierten „Fusion Middleware 11g“. Sie enthält unter anderem einen erweiterten „Enterprise Manager“. Dieses System-Management-Werkzeug könnte Unternehmen dabei helfen, Verwaltungsaufgaben im Datenbankumfeld zu automatisieren, die heute mit eigenen Tools oder selbst geschriebenen Scripts erledigt werden. Jedoch wollen viele Kunden auf das Release 2 der neuen Produktgeneration 11g warten. Das ist nicht neu: Viele Firmen nehmen erst das „Terminal-Release“ einer Hauptversion in Betrieb.

Nach Angaben der DOAG stellt die Oracle-Plattform mit „AIA“ (Applica-

tion Integration Architecture) eine Integrationskomponente bereit, mit der sich verschiedene Anwendungen integrieren lassen. Die Lösung dient Oracle-Nutzern beispielsweise dazu, das CRM-Produkt „Siebel“ mit einem ERP-System von SAP zu verbinden. Diese Kopplung wurde früher durch zahlreiche Schnittstellenprobleme erschwert.

Virtualisierung spielt der DOAG zufolge auch bei Oracle-Nutzern eine große Rolle. Mit Hilfe der Technik hoffen die Firmen, ihre IT flexibler machen und ihre Hardware besser auslasten zu können. Jedoch sind den Nutzern die Lizenzbestimmungen des Softwarekonzerns diesbezüglich nicht verständlich genug. Sie fragen sich zum Beispiel, wie ein Lizenzvertrag auf Basis der Anzahl der CPUs eines Rechners zu bewerten ist, wenn die betreffende Software in einer virtuellen Maschine läuft, ohne dabei einem dedizierten Prozessor zugeordnet zu sein.

Hinzu kommt, dass Firmen oft einen Mix aus verschiedenen Softwareprodukten von Oracle und anderen Anbietern in virtuellen Umgebungen nutzen möchten.

Doch die wirklichen Probleme der Oracle-Kunden betreffen nicht die Technik oder die Lizenzmodelle: Woran es in ihren Unternehmen trotz Krise mangelt, sind IT-Fachkräfte, und zwar solche, die zwar einerseits Technik-affin sind, jedoch interdisziplinär denken und im Team arbeiten können. Den Mangel an Experten, die sowohl Geschäftsprozesse der Fachabteilungen als auch die IT-Lösungen verstehen, nannten die in der Studie befragte Mitglieder der DOAG als eines

von drei Risiken in Sachen IT. Darüber hinaus hoffen sie, dass ihre Budgets nicht weiter schrumpfen. Insgesamt nahmen 182 Oracle-Kunden an der Befragung teil. Die DOAG wollte in einer Umfrage von ihren Mitgliedern wissen, wie sie Oracle-Software einsetzen und welche Chancen sowie Risiken sie dabei sehen.

Die befragten Firmen sagten aus, dass der Druck steige, gesetzliche Anforderungen in IT umzusetzen (Compliance) und Lösungen für das Risiko-Management bereitzustellen. Dazu zählt beispielsweise, in den IT-Systemen die Vorgaben des US-amerikanischen Sarbanes-Oxley-Act („SOX“) sowie der deutschen GdPDU abzubilden. Oracle selbst macht es da den Kunden auch nicht immer leicht: So hatte der Anbieter erst nach Druck der Anwender die Zertifizierung der „E-Business Suite Release 12“ gemäß den „Grundsätzen ordnungsgemäßer Buchführung“ (GoB) erneuert – eigentlich sollte das für einen ERP-Anbieter, der eine integrierte Buchhaltung entwickelt, selbstverständlich sein. Die E-Business Suite ist eine ERP-Komplettlösung, die für internationale Unternehmen geeignet ist. Sie steht in direktem Wettbewerb mit der „Business Suite“ des Konkurrenten SAP.

Das Interesse an Business-Intelligence-Software rührt nicht zuletzt her von dem Bedarf nach Funktionen für das Risiko-Management. Softwarelösungen sollen Firmen helfen, durch die Analyse von Geschäftsdaten die eigene Situation einschätzen und Risiken für das Unternehmen abschätzen zu können. Doch auch diese Initiativen lassen

sich nur mit entsprechenden Experten umsetzen, da gerade hier eine enge Zusammenarbeit zwischen IT und der Fachabteilung vonnöten ist.

Dem Mangel an Fachkräften wollen weder die DOAG noch Oracle tatenlos zusehen. Die DOAG hat hierzu eine „Hochschul Community“ etabliert. Die Anwendervereinigung will über ihre Plattform Professoren, Studenten, den Softwarehersteller, dessen Partnern und Anwender zusammenbringen.

Der Softwarekonzern hält seit einigen Jahren über die „Oracle Academic Initiative“ Kontakt zu insgesamt 260 Fachbereichen an Universitäten, Fachhochschulen und anderen Bildungseinrichtungen. Die beteiligten Institute erhalten Oracle Datenbank und Middleware-Software beziehungsweise einen Zugriff auf gehostete Geschäftsapplikationen.

silicon.de vom 7. Juli 2009:

Fachkräftemangel drückt Oracle-Anwender

Oracle hat mit dem Release von Fusion Middleware 11g als Anbieter einen riesen Schritt getan. Aber gibt es auch Risiken, die ein Anwender beim Einsatz in Betracht ziehen muss? Und welche Trends sehen die Mitglieder der Deutschen Oracle-Anwendergruppe in der Krise? Diese und andere Fragen hat die DOAG Deutsche Oracle Anwendergruppe, seinen Mitgliedern gestellt.

Das Ergebnis: Die absolute Mehrheit der Oracle-Anwender verwendet (natürlich) eine Oracle-Datenbank. Auch Entwicklungstools sind bei den in der DOAG organisierten Anwendern weit verbreitet. Bei Middleware wird die Zahl der Anwender schon bedeutend kleiner. Bisher am wenigsten Zuspruch finden die Business Applications von Oracle.

„Wir sehen für das Jahr 2009 keine gewaltigen Sparmaßnahmen“, erklärte Fried Saacke, Vorstand und CEO der DOAG bei der Vorstellung der Ergebnisse der traditionellen Mitgliederbefragung in München. 46 Prozent der insgesamt 182 Anwender halten ihre Investitionen konstant. Knapp

ein Drittel wird laut eigenen Angaben weniger in die IT investieren. Denen stehen jedoch etwa ein Viertel der Anwender gegenüber, die 2009 wieder mehr Geld für Software-Lösungen ausgeben wollen.

Saacke hat dafür auch eine Erklärung parat: „Viele Anwender haben bereits im vergangenen Jahr saniert.“ Andere Unternehmen versuchten derzeit auch Budgets aus anderen Bereichen in die IT umzuschichten um somit langfristige Ziele erreichen zu können.

Für über ein Drittel der Anwender ist es derzeit sehr wichtig, in neue Software zu investieren. Etwa die Hälfte der Befragten steht dieser Frage neutral gegenüber. 16 Prozent gaben an, dass es derzeit nicht besonders relevant sei, zu investieren. Immerhin ein Viertel glaubt, mit Oracle-Lösungen zum wirtschaftlichen Erfolg des Unternehmens einen sehr großen Beitrag leisten zu können. 42 Prozent sehen einen großen Beitrag der Oracle-Produkte. Knapp ein Drittel stuft den Beitrag als neutral ein.

Saacke erklärte, dass die Befragung keine repräsentative Umfrage wäre, sondern nur stichprobenartig geführt worden sei. Dennoch würden die Ergebnisse einen guten Querschnitt über die Befindlichkeiten der Anwender geben.

Fehlende IT-Fachkräfte seien laut Umfrage auch und gerade in Zeiten der Krise eines der größten Probleme für Unternehmen. 44 Prozent sehen darin ein großes Hindernis in der Umsetzung der Business-Pläne.

Saacke: „Trotz Krise haben wir bei den Fachkräften keinen Überhang.“ Oracle engagiere sich zwar stark bei der Rekrutierung und Weiterbildung von Datenbank-Fachkräften, doch ist das der DOAG noch lange nicht genug. Denn gerade bei den Business Applications und im Bereich Middleware könnte sich der Hersteller noch weiter engagieren, mahnt Saacke. Um hier gegenzusteuern, habe die DOAG eine DOAG-Hochschul-Community gestartet.

Als Ziel dieser Community nennt Saacke, möglichst viele Studenten für Oracle-Technologien zu gewinnen. Günther Stürner, Vice President Busi-

ness Unit Server Technology und Sales Consulting bei Oracle, stellte jedoch klar, dass Oracle die derzeit bestehenden Programme und Trainings nicht ausbauen werde.

Aber was muss ein Bewerber denn eigentlich mitbringen? Gesucht würden derzeit vor allem „Grenzgänger zwischen Business und IT“, wie Helene Lengler, Vice President Sales Fusion Middleware bei Oracle erklärte. Die DOAG definiert den perfekten Kandidaten als: „Teamplayer, interdisziplinär, technikaffin, jemand mit Überblick über komplexe Systeme, der das Business und die Technik versteht und zudem bereit ist, permanent zu lernen.“

Bei den IT-Budgets jedoch bekommen auch die DOAG-Mitglieder die Krise voll zu spüren: 42 Prozent geben an, dass fehlendes Kapital bei der Realisierung von Business-Plänen ein großes Problem ist. Doch auch hier scheint es Hoffnung zu geben. Der Tiefpunkt bei den Investitionen, so Saacke, scheint erreicht zu sein. Die Erfahrungen der DOAG-Experten zeigen, dass Unternehmen derzeit vor allem in Projekte investieren, die kurzfristig „einen schnellen Wertbeitrag oder das schnelle Ausnutzen von Marktchancen versprechen“. Langfristige Projekte, deren Nutzen nicht auf den ersten Blick ersichtlich werde, würden derzeit überhaupt nicht angegangen.

Neben gekürzten Budgets weiß die DOAG auch von steigenden Anforderungen beim Risikomanagement sowie in Governance und Compliance. „Daher sehen wir vor allem die Bereitschaft, in Business Intelligence so wie entsprechende Tools zu investieren“, erklärt Frank Schönthaler, Leiter der Special Interest Group E-Business Suite bei der DOAG.

Wer den so genannten Vendor Lock-in nicht fürchtet, für den bietet Oracle vor allem mit der neuen Fusion Middleware 11g mehr denn je Chancen. So hält die DOAG – wie auch viele Analysten – fest, dass der Hersteller hier eine klare Strategie verfolgt und auch die nötigen integrierbaren Lösungen liefert. Mit neuen Applikationen decke Oracle auch immer mehr branchenspezifische Anforderungen ab.

Derzeit gibt es also produktseitig nur noch wenige Fragen bei den Nutzern. Doch damit sind noch längst nicht alle praktischen Probleme gelöst. „Für viele Anwender sind derzeit die Migrationspfade noch nicht so klar“, berichtet Dietmar Neugebauer, Vorstandsvorsitzender der DOAG. So müssten sich derzeit viele Anwender fragen: „Wie komme ich mit meiner derzeitigen Infrastruktur auf die neuen von Oracle vorgegebenen Technologien.“

Ungereimtheiten scheint es zwischen der DOAG und Oracle auch bei einigen Detailfragen bei der Lizenzierung von Anwendungen in virtualisierten Umgebungen zu geben. Doch wie Günther Stürmer von Oracle versichert, arbeite man derzeit an einem Paper, das die letzten Unwägbarkeiten ausmerzen solle.

Channelpartner vom 30. April 2009:

Ist Oracle an Suns Hardware interessiert?

Der überraschende Zugriff Oracles auf Sun wirft gewichtige Fragen auf. Sowohl die Deutsche Oracle Anwendergruppe als auch die Sun User Group Deutschland haben darauf Antworten.

Während Verkäufer Jonathan Schwartz, CEO von Sun Microsystems, die neue Zukunft des von Oracle geschluckten Server- und Java-Anbieters in rosiges Licht zu tauchen versucht („Diese Übernahme ist eine Neudefinition der gesamten Industrie“), versuchen in Deutschland die beiden Anwenderorganisationen der Unternehmen, die „Sun User Group Deutschland“ (SUG) und die „Deutsche Oracle Anwendergruppe“ (DOAG) die neue Konstellation zu verstehen. Im Gespräch mit ChannelPartner erklärten Stefan Koospal, 1. Vorsitzender der SUG, und Fried Saacke, Vorstand und Geschäftsführer der DOAG, ihre Perspektiven.

Java: Darin sind sich beide Vertreter einig: Die millionenfach verwendete Programmiersprache hat bei Oracle eine große Zukunft. Sie ist als „Programmier-technik“ weltweit im Einsatz, „nicht wegzudenken“ (Koospal), und ferner für Oracle und seine Middleware-Pläne zentral“ (Saacke). Kritisch sehen beide, wie IBM, neben Sun der wichtigste

Java-Beiträger, sich nach dem Kauf verhalten wird. Die Gefahr von Java-Dialekten sehen sie noch nicht.

Solaris: Auch dem quelloffenen Betriebssystem Solaris räumen beide gute Chancen ein. Allerdings: Während Koospal das System mangels Nachwuchsförderung nicht auf Desktops sieht, sagt Saacke, Oracle habe es auf Solaris „abgesehen“. Es passe zu den Plänen, einen „kompletten Stack“ für die Unternehmens-IT anbieten zu können. Welche Ressourcen Oracle in die Entwicklung stecken werde, sei offen, doch hängt es laut Saacke deutlich von dem Willen Oracles ab, sich als verläSSLicher Partner bei Suns Kunden zu präsentieren.

Open Office: Beide befürchten, dass Oracle die „lange hin und her geschobene Office-Alternative (Koospal) nicht weiter beachten wird. Saacke sagt, Oracle sei an geschlossenen Lösungen interessiert, weshalb es die Büro-Suite lediglich als Abwehr oder Ausgliederung von Microsoft aus Oracle-Umgebungen einsetzen werde.

Middleware/MySQL: Da Oracle seit vier Jahren seinen Middleware-Bereich stetig ausgeweitet habe und sich damit gegenüber den Erzrivalen SAP (Netweaver) und IBM (WebSphere) mehr als behaupten wolle, sei hier wenig zu befürchten. Durch den Kauf könne es sowohl Open-Source-Entwickler an sich binden wie SQL- und Applikations-Entwickler. Allein für die offene Datenbank MySQL müsse Oracle ein plausibles Modell finden. Saacke schlägt vor, MySQL als Einstiegsdatenbank gegenüber Microsoft SQL zu platzieren; Koospal sagte, die Datenbank könne in Webumgebungen gut eingesetzt werden. Beide verneinen, MySQL-Server könne aufgegeben werden.

Sparc-Server/Storage: Uneins sind sich die beiden bei der Hardware. Während Koospal sagte, Sparc-Rechner plus Software eigneten sich als Out-of-the-box-Lösung für viele Anwendungsgebiete, erklärte Saacke, Oracle könne die Unix-Server mitsamt Storage „ausgliedern und weiterverkaufen“. „Bedenken Sie die Software-Margen von bis zu 50 Prozent!“ Ebenso denkbar sei, dass Oracle mit der Kombination Hard- und Software sich neue Kundensegmente

erschließen könne. Insgesamt zweifeln beide am Hardware-Interesse Oracles.

Partner: „Beide Firmen brauchen verlässliche Partner“, sagen Koospal und Saacke übereinstimmend. Daran werde sich nichts ändern.

Computerwoche vom 23. April 2009:

Oracle kann Hardware nicht

Mit der Übernahme von Sun Microsystems hat Oracle die Branche überrascht. Das sagen Analysten und Kunden zum Deal.

Aus Sicht der Analysten hat es Oracle1 in erster Linie auf das Softwaregeschäft von Sun abgesehen. Das Schicksal der Hardware rund um Suns Sparc-Architektur und das von StorageTek zugekaufte Speichergeschäft sei dagegen mehr als unklar. Jim McGregor, Chief-Technology-Strategie von In-Stat, ist skeptisch, was die Zukunft von Suns Hardware betrifft. Die Sparc-Architektur habe im Markt der Unix-Server kontinuierlich Marktanteile gegen die konkurrierenden Plattformen von IBM und Hewlett-Packard verloren. Das sei Grund genug für Oracle, die Sparc-Entwicklung einzustellen beziehungsweise das Geschäft ganz abzustoßen. Der Markt für Unix-Server ziehe sich mehr und mehr in bestimmte Nischen wie das High Performance Computing (HPC) zurück. „Für viele Anbieter gibt es deshalb keinen Sinn, an diesem Geschäft festzuhalten.“ McGregor verweist auf das Schicksal von Unternehmen wie Cray und SGI, die mittlerweile von der Bildfläche verschwunden seien.

Es gebe für Oracle kaum Gründe, die Sparc-Entwicklung voranzutreiben, meint auch Jack Gold, Principal Analyst von Gold Associates. Der Hersteller habe in der Vergangenheit immer betont, seine Software basiere auf Standards und stelle keine besonderen Ansprüche an die darunterliegende Hardware. „Es kann daher nicht im Interesse Oracles liegen, seine Software für eine bestimmte Hardwareplattform zu verbieten.“

Weitere Pressestimmen zur DOAG finden Sie unter <http://www.doag.org/presse/spiegel>



Von links: Jürgen Kunz, Dr. Dietmar Neugebauer und Fried Saacke

„Wir haben im Vergleich zum Wettbewerb sehr gut abgeschnitten ...“

Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG, sowie DOAG-Vorstand und Geschäftsführer Fried Saacke sprachen mit Jürgen Kunz, Senior Vice President und Geschäftsführer der ORACLE Deutschland GmbH, unter anderem über die Marktentwicklung in Deutschland.

Sie haben am 31. Mai 2009 als Geschäftsführer der Oracle Deutschland GmbH das Geschäftsjahr 2009 abgeschlossen. Welche der gesteckten Ziele haben Sie erreicht?

Kunz: Das weltweite Ergebnis war mit einem Umsatz von 23,3 Mrd. US-Dollar sehr erfolgreich. Die prozentualen Steigerungsraten in Deutschland decken sich ungefähr, so dass wir auch hier von einem guten Resultat sprechen können.

Wir haben im Vergleich zum Wettbewerb sehr gut abgeschnitten und das realisiert, was wir uns auch vorgenommen haben.

Im Datenbank-Bereich konnten wir unsere Marktführerschaft weiter ausbauen, bei der Middleware sind unsere Erwartungen deutlich übertroffen worden, und bei den Business-Applications haben wir sowohl weltweit als auch hierzulande Marktanteile gewonnen.

Weltweit sind bei Oracle die Software-Erlöse aus Neulizenzen deutlich gesunken. Worauf führen Sie das zurück?

Kunz: Das hängt natürlich mit der derzeitigen Wirtschaftslage zusammen, die auch an der IT-Branche nicht spurlos vorbeigeht. Dennoch haben uns die Analysten im Vergleich zum Wettbewerb ein absolutes Top-Ergebnis bestätigt.

In der Presse ist von Entlassungen bei Oracle in Europa zu lesen. Inwiefern ist die deutsche Niederlassung davon betroffen?

Kunz: Um immer optimal aufgestellt zu sein, gleichen wir uns ständig den Geschäftsgegebenheiten an. Das ist gerade in diesen Zeiten zwingend notwendig. Wir veröffentlichen dazu allerdings keine konkreten Zahlen.

Welche Ziele hat Oracle für das nächste Geschäftsjahr?

Kunz: Wir haben mit Sun eine große Akquisition angekündigt, deren Integration bevorsteht. Ansonsten werden wir die eingangs erwähnten Geschäftsziele verfolgen, um weiter auf der Erfolgsspur zu bleiben.

Gibt es Änderungen in der Organisationsstruktur?

Kunz: In Deutschland haben wir eine sehr große Stabilität in der Organisation; hier wird es keine großen Änderungen geben.

Wie weit ist die Übernahme von Sun fortgeschritten?

Kunz: Zunächst muss das Angebot von den Sun-Aktionären angenommen werden. Auch die Kartell-Behörde muss noch zustimmen. Erst nach dem Erwerb der Aktien-Mehrheit, der voraussichtlich im August stattfinden wird, können wir mit dem Unterneh-

men in Kontakt treten, über das „Goto-Market-Modell“ reden und in den jeweiligen Ländern mit der Zusammenführung beginnen.

VMWare ist unumstrittener Marktführer bei Virtualisierungslösungen. Warum ist kein Oracle-Produkt unter VMWare zertifiziert?

Kunz: Wenn ein Kunde Oracle-Software unter Virtualisierungslösungen wie VMWare, XEN, Hyper-V oder Oracle VM nutzt, erhält er auch die entsprechende Unterstützung vom Support.

Immer mehr DOAG-Mitglieder fordern eine faire Lizenzierung bei Einsatz von VMWare wie die Anerkennung von Hard-Partitioning. Wird Oracle sich hier bewegen?

Kunz: Momentan gibt es keine Anzeichen für eine Veränderung. Ich glaube auch nicht, dass das ein gängiges Problem ist. Wenn jemand den Oracle-Pool lizenziert hat, besitzt er die volle Flexibilität.

Wie erklärt Oracle die Preiserhöhung der Preisliste vom 18. Juni 2009?



Kunz: Es gibt jedes Jahr aus unterschiedlichen Gründen eine Preisanpassung. Von daher ist dies keine außergewöhnliche Aktion.

Ist eine solche Preiserhöhung kein Hindernis, um auf dem Markt erfolgreich zu sein?

Kunz: Es handelt sich hier um weltweite Listenpreise. Wir bieten allerdings auch entsprechend flexible Bundlings und Rahmenverträge an.

Wann werden die BI-Tools der Fusion Middleware 11g produktiv verfügbar sein?

Kunz: Etwa ab Dezember 2009.

Wann ist Forms 11g produktiv verfügbar?

Kunz: Es steht bei Drucklegung dieser Ausgabe zum Download bereit.

Wie sind die lizenztechnischen Migrationspfade vom alten Oracle Application Server zu Fusion Middleware?

Kunz: Klassischen Komponenten des Oracle Application Servers 10g existieren unter Fusion Middleware 11g und bleiben auf der Preisliste (Forms, Reports, Discoverer und Portal). Diese Produkte nutzen eine „Restricted Licence“ des WebLogic Servers. Kunden mit Support können ohne zusätzliche Kosten auf 11g upgraden und als Grundlage diese Version des WebLogic Servers nutzen.

Worin liegen die Vorteile der Oracle Middleware?

Kunz: Durch unsere offenen Standards sind wir in der Lage, bestehende Anwendungen in relativ kurzer Zeit in unsere Middleware zu integrieren. Damit kann der Kunde Schritt für Schritt seine Prozesse in eine moderne Architektur überführen. Die Projekte sind auf diese Weise schnell umsetzbar, weniger risikobehaftet und gut zu kalkulieren.

Wie stark liegen On-Demand-Lösungen im Trend?

Kunz: Wir haben im letzten Jahr einige On-Demand-Projekte sehr erfolgreich realisiert. Wir sehen hier eine hohe Nachfrage, sofern sich ein Kunde dazu entschieden hat, die Verantwortung für IT-Infrastruktur und den Applikationsbetrieb zu delegieren. Inzwischen stellen verstärkt auch große Unternehmen hier Überlegungen an.

Was werden die Highlights zur Oracle Open World 2009 sein?

Kunz: Das „Goto-Market-Modell“ für Sun wird natürlich eine große Rolle spielen. Speziell für die deutsche Community wird es zudem wieder entsprechende Events geben.

Welche Ziele haben Sie sich persönlich für die kommenden Jahre gesteckt?

Kunz: Wir haben hier in Deutschland ein sehr starkes Management-Team, das gemeinsam den Erfolg herbeiführt. Unser Ziel ist es, in allen Produkt-Bereichen weiterhin erfolgreich zu sein.

Wo sehen Sie für Oracle die Vorteile und wo die Nachteile einer starken Anwendervertretung wie der DOAG?

Kunz: Die DOAG gibt uns sehr wichtige Hinweise aus dem Kreis der Anwender, die wir sehr ernst nehmen. Auf der anderen Seite können wir über die DOAG auch viele Dinge gezielt an unsere Kunden kommunizieren. Abgesehen von einigen konzeptionellen Dingen, die man vielleicht verbessern könnte, sehe ich keine Nachteile.

Werden Sie an der DOAG 2009 Konferenz + Ausstellung von 17. bis 19. November teilnehmen?

Kunz: Ich werde am 18. November den ganzen Tag über anwesend sein.

Die neue Fusion Middleware 11g

Wolfgang Taschner, Chefredakteur der DOAG News

Am 2. Juli 2009 stellte Oracle in München das von vielen Anwendern erwartete Release 11g der Oracle Fusion Middleware vor. Dazu gehören neue innovative Funktionen in der SOA Suite, der WebLogic Suite, der WebCenter Suite und beim Identity Management.

Mit der Vorstellung von Oracle Fusion Middleware 11g erweitert Oracle seine Verpflichtung zu seinen Design-Prinzipien und bietet Innovationen an, um die Komplexität der IT zu reduzieren. Die neue Fusion Middleware 11g orientiert sich an den Anforderungen moderner Anwendungs-Infrastrukturen und erleichtert damit die Arbeit bei Entwicklung, SOA, Cloud Computing, Business Process Transformation, Service-orientierted Security Collaboration sowie IT Governance. Zu den Erweiterungen der Plattform zählen:

- Die SOA Suite 11g stellt eine zentrale Prozess-Plattform dar, die System-, Personen- und Dokumentenzentrierte Prozesse sowie eine Event Driven Architecture (EDA) mit einer breiten Palette an SOA-Funktionen – von Entwicklung über Sicherheit bis hin zu Governance – vereinigt. Die SOA Suite 11g, die auf einer Oracle Application Grid Infrastruktur betrieben wird, ist optimiert dafür, Dienste in privaten und öffentlichen Clouds aufzubauen und zu integrieren.
- Die WebLogic Suite 11g (inklusive WebLogic Server) bietet neue Möglichkeiten, den Betrieb zu überwachen und zu automatisieren, was zu höherer Server-Verfügbarkeit und zu geringerem Aufwand für Administratoren führt. Neue Funktionen, wie Fusion Middleware GridLink für Oracle Real Application Clusters und Fusion Middleware Enterprise Grid Messaging, verbessern zudem die Zuverlässigkeit und Verfügbarkeit. Fusion Middleware ActiveCache gestattet die schnelle Skalierung, um auf wechselnde Anwendernachfrage und Systemlast zu reagieren.
- Die WebCenter Suite 11g bietet eine breite Palette von wiederverwendbaren, Out-of-the-Box WebCenter-Ser-

vices-Komponenten, die in nahezu jede Art von Portal integriert werden können. Dazu gehören beispielsweise Intranets, Composite Applications oder Web-basierte Communities, bei denen die Social-Networking-Funktionen und die Produktivität verbessert werden. Oracle Composer, ein selbsterklärendes, Browser-basiertes Tool, erleichtert es Endanwendern und Entwicklern, Anwendungen, Portale und soziale Webseiten zu entwickeln, zu teilen und zu personalisieren. Bei dem neuen Oracle WebCenter Spaces handelt es sich um eine Lösung, die vorgefertigte Social Networks (Group Spaces und Personal Spaces) enthält, die von Endanwendern aufgebaut und verwaltet werden können, um Produktivität, Kommunikation und Effizienz zu erhöhen.

- Das Identity Management 11g enthält die ersten Komponenten einer voll integrierten Identity Management Suite und verfügt über eine verbesserte Integration mit anderen Oracle Fusion Middleware Lösungen. Dazu gehören neue Funktionen wie Deployment Accelerators, Universal Federation Framework sowie eine moderne Benutzerschnittstelle, die auf Oracle Application Development Framework (ADF) Faces basiert.
- Die Oracle Development Tools: Oracle Fusion Middleware 11g baut auf den bereits angekündigten strategischen Entwicklungstools auf, einschließlich JDeveloper, Application Development Framework und TopLink.

„Mit Oracle Fusion Middleware 11g erhalten mehr als 90.000 bestehende Kunden eine Basis für Innovationen, mit denen sie sich von ihren Konkur-

renten abheben, Veränderungen ihres Geschäfts schnell erkennen und darauf reagieren können“, so Thomas Kurian, Oracle Senior Vice President Fusion Middleware. „Mit der engeren Integration über die gesamte Produktlinie hinweg und mit großen technischen Innovationen in jeder Produktkategorie unterstützt Oracle seine Kunden dabei, von der Effizienz zeitgemäßer IT zu profitieren.“

Die Einschätzung der DOAG-Experten



Perry Pakull, Leiter der Special Interest Group Middleware:

Mit Fusion Middleware 11g setzt Oracle einen Meilenstein im Bereich ihrer Middleware-Produktpalette. Obwohl die Integration der BEA-Produkte zusätzliche Zeit in Anspruch genommen hat, bringen gerade die Bausteine WebLogic Server und Service Bus zusätzliche Möglichkeiten und vor allem weitere Stabilität. Die enge Verzahnung aller Produkte basierend auf einer einheitlichen Infrastruktur ist ein Vorteil, der das Warten gerechtfertigt hat.

Im Kern von Fusion Middleware 11g steht die SOA Suite als Basis für die Realisierung von service-orientierten Architekturen und Systemen. Die SOA Suite wurde durch die Service Component Architektur (SCA) in 11g nochmals erweitert, der Reifegrad der

Software macht aber einen Einsatz im Projekt sofort möglich.



Björn Bröhl, OPITZ CONSULTING GmbH, und Torsten Winterberg (Foto), Leiter der Special Interest Group SOA:

Das sind die Highlights der lang erwarteten Produktsuite von Oracle: Das Herz der neuen Plattform und damit die Basis für alle Fusion-Middleware-Komponenten bildet ab sofort der von BEA übernommene Weblogic Application Server 11g. Damit steht nun anerkanntermaßen einer der besten verfügbaren Application Server für Projekte auf Oracle-Basis zur Verfügung.

Das Killer-Feature in der neuen Oracle SOA Suite 11g ist ebenfalls leicht zu benennen: Die extrem enge Integration aller Komponenten, die in den einzigartigen Monitoringmöglichkeiten im neuen EM ihren Höhepunkt findet, ist so auf dem Markt ansonsten nirgendwo zu haben. Das Tracing über alle Komponenten (ESB, BPEL, HumanTasks, Events etc.) ist ein Traum – und dringend benötigt, bedenkt man die doch stark gestiegenen Anforderungen im Umfeld lose gekoppelter Systeme.



Hajo Normann, Oracle ACE Director, Co-Leiter Special Interest Group SOA

Seit drei Jahren heißt es „Mit SOA Suite 11 wird alles gut!“. Und es ist ja auch so. Rules Designer und Decision Tables im JDeveloper, größere Flexibilität im BPEL-Prozess durch leichtgewichtige Events etc. Das Gesamtpaket – aus einem Guss – ist groß geworden und ein wesentlicher Meilenstein für Enterprise SOA/BPM. Bei der Featurefeier kann man leicht das Hauptplus übersehen: Die Komplexitätsreduzierung durch die Komponententechnologie Service Component Architectur (SCA), die BPEL, Routing, Human Interaction grafisch in einen Gesamtkontext stellt, wo man vorher durch mehrere JDeveloper-Reiter browsen musste, um sich ein Bild vom Zusammenspiel zu machen. Das zieht sich durch bis ins Deployment, durch SCA gibt es einen konsolidierten Ort, um URL-Roots und andere Kontextvariablen zu speichern. Das macht die Arbeit leichter.

Es ist nun einfach, alles in ein SCA Composit hineinzupacken. Hier gilt es darauf zu achten, ob da konzeptionell auch wirklich alles hineingehört, ich denke hier besonders an technische

Aspekte wie Integrationslogik und fachliche Aspekte wie Prozesse und Geschäftsregeln. Sollte man diese zusammenbündeln? Es hängt davon ab: Wo man einen SOA-Layer oberhalb von Applikationen legen möchte, kann es sinnvoll sein, die Integrationsaspekte in den neuen „Mediatoren“ mit der Fachlichkeit in BPEL zu verdrahten, da der Austausch der Applikation nicht wahrscheinlich ist.

Wie sieht jedoch das Design aus, wenn man lose Kopplung erreichen möchte, also Prozesse etablieren, die Business Services orchestrieren, deren Implementierung leicht austauschbar sein soll? Dann sollten die fachliche Aspekte von den Integrationsaspekten getrennt deployt werden, also nicht in einem SCA-Composit versammelt sein. Hier bietet sich entweder ein eigenes SCA-Composit pro Business Service an – oder, bei weitreichenden Integrationsaufwänden, die Realisierung der Fassade auf zu integrierende Endsysteme mithilfe des neuen, von BEA übernommenen Oracle Service Bus (OSB). Dieser gehört zur Oracle SOA Suite, ist aber leider noch nicht voll integriert. Gerade hier bleibt Oracle also noch etwas zu tun, so dass der eine oder andere vielleicht, bei aller Freude über das neue Release, bald sagt: „Mit Oracle SOA Suite 11gR2 wird alles gut“. Loslegen sollte man jedoch schon heute!

Kontakte:

Perry Pakull
sig-middleware@doag.org
Torsten Winterberg
Hajo Normann
sig-soa@doag.org

Impressum

Herausgeber:

DOAG – Deutsche ORACLE Anwendergruppe e.V.
Tempelhofer 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

Chefin von Dienst (CvD):

Carmen Al-Youssef,
office@doag.org

Gestaltung und Satz:

Claudia Wagner,
DOAG Dienstleistungen GmbH

Anzeigen:

Yvonne de Andrés, office@doag.org
DOAG Dienstleistungen GmbH

Mediadaten und Preise finden Sie unter:
www.doag.org/publikationen/

Druck:

adame Advertising and Media
GmbH Berlin
www.adame.de

13x COMPUTERWOCHE

Jetzt testen

Sie sparen über 30 %

2 GB-Stick GRATIS



**2 GB SanDisk Cruzer®
Micro USB-Stick**

- Versenkbarer USB-Stecker
- Zertifiziert für High-Speed USB 2.0 (auch für USB 1.1)
- Für U3-Smart geeignet
- Umfasst die folgenden U3-Programme:
 - CruiserSync-Synchronisierungssoftware
 - SignupShield-Passwortverwaltung
 - SKYPE „Video- und Telefongespräche von PC zu PC“
 - McAfee-30 day Trial-Antivirussoftware



13 Ausgaben für nur 39 €

- + Sie sparen über 30 %
- + Gebündeltes IT-Wissen
- + Jeden Montag pünktlich ins Haus
- + Keine Ausgabe verpassen
- + **2 GB USB-Stick gratis**

Die **COMPUTERWOCHE** ist die führende deutschsprachige Wochenzeitung für die Märkte der Informations- und Kommunikationstechnologie. Sie informiert aktuell, kompetent und anwendungsorientiert. Die **COMPUTERWOCHE** liefert strategische Informationen und bietet fundierte Analysen über Trends, neue Technologien, Produkte und Märkte. IT-Entscheider und IT-Profis in Unternehmen erhalten eine verlässliche Grundlage für Entscheidungen über betriebliche IT-Investitionen.

Jetzt COMPUTERWOCHE testen und Geschenk sichern!

☑ Telefon
0711/7252-276

☑ Fax
0711/7252-377

☑ E-Mail
kundenservice@computerwoche.de

Am schnellsten geht's online unter:
www.computerwoche.de/shopcode



COMPUTERWOCHE

COMPUTERWOCHE erscheint im Verlag IDG Business Media GmbH, Lyonel-Feining-Str. 26, 80807 München, Handelsregister München, HRB 99187, Geschäftsführer: York von Heimburg. Die Kundenbetreuung erfolgt durch den IDG Business Kundenservice, Postfach 810580, 70522 Stuttgart, Telefon: 0711/7252-276, Fax: 0711/7252-377, E-Mail: kundenservice@computerwoche.de

Migrationstrategie für Kernbestandteile der Fusion Middleware

Georg Molter, Christian Kücherer und Matthias Pfau, Zühlke Engineering GmbH

Nach der Übernahme von BEA hat Oracle eine Produktstrategie vorgelegt, die auf Grundlage von „Best of Breed“-Entscheidungen sowohl ehemalige Oracle- als auch BEA-Produkte enthält. Vielen Anwendern sind die konkreten Konsequenzen für ihre eigenen IT-Umgebungen unklar. Dieser Artikel beschreibt eine Vorgehensweise für die Migration von Kernbestandteilen der Fusion-Middleware-Plattform, darunter die WebLogic- und Oracle Application Server, die SOA-Produkte und die Portalserver WebCenter, WebLogic Portal und Oracle Portal.

Neben der Entwicklung und Wartung von Anwendungen besteht eine Hauptaufgabe des Dienstleistungsbereichs in der Migration von Altapplikationen. Dieser Bedarf ist durch die Zusammenführung des Produktportfolios von Oracle und BEA besonders akut. In den folgenden Abschnitten werden daher Grundlagen zur Migration zwischen den Produkten im Bereich Application Server, Enterprise Service Bus und Portal beschrieben.

Methodik für Migrationsprojekte

Migrationsprojekte stellen aufgrund der natürlichen Altlasten eine stärkere Herausforderung dar als Neueinführungen. Zum einen stecken in Legacy-Code große Investitionen, die es möglichst zu erhalten gilt. Und noch wichtiger: Es sind die existierenden Anwendungen, mit denen im Tagesgeschäft eines Unternehmens Geld verdient wird. Wenn diese nicht weiterhin funktionsfähig gehalten werden, dann muss eine Ablösung ohne signifikante Unterbrechungen erfolgen.

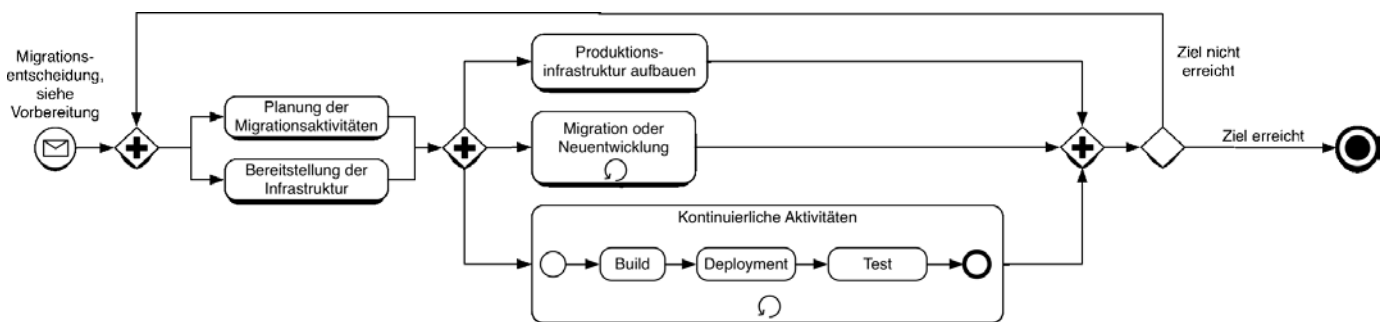
Die Basis für eine erfolgreiche Migration wird durch eine sorgfältige Planung gelegt (siehe Abbildung 1). Zunächst müssen die Ziele des Migrationsprojektes klar definiert und die Randbedingungen festgehalten sein, unter denen dieses durchgeführt wird. Des Weiteren wird erfasst, wie die Ziel-IT-Landschaft des Unternehmens in Zukunft aussehen soll und wie die Roadmap für die Verwendung und Weiterentwicklung der betroffenen Applikationen aussieht. Zudem ist eine separate Betrachtung der zu migrierenden Applikationen erforderlich, um spezifische Eigenschaften und Randbedingungen zu erfassen.

Die so erarbeitete Strategie adressiert neben technologischen Aspekten und konkreten Ansätzen auch die Kosten und Risiken. Sie ermöglicht somit fundierte Entscheidungen und definiert den Rahmen für die Durchführung.

Die Durchführung der Migration läuft wie in Abbildung 2 dargestellt ab. In der Grobplanung sind die Ziele der einzelnen Schritte, unter Berücksichtigung der Risiken und der zuvor fest-

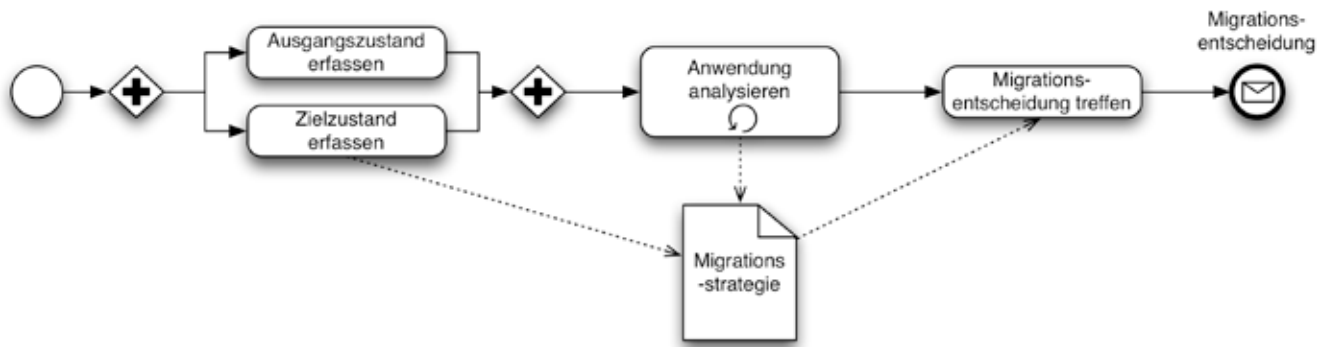
gehaltenen Randbedingungen, festgelegt. Zudem muss die erforderliche Infrastruktur bereitgestellt werden – die Zielumgebung für die Produktion ebenso wie die Integrations- und Testumgebungen. Wichtig sind die rechtzeitige Verfügbarkeit einer effizienten Entwickler-Toolchain sowie einer auf die Zielplattform zugeschnittenen Build- und Deployment-Umgebung. Dies ist für eine effiziente Durchführung der Migration und für die kontinuierliche Qualitätssicherung unerlässlich. Ebenfalls nicht zu unterschätzen sind die Bedeutung des konsequenten Wissensaufbaus bei allen Beteiligten und die Mitarbeit von Wissensträgern des Altsystems.

Wenn diese Voraussetzungen erfüllt sind, kann man die eigentliche Migration angehen. Da im Einzelfall die Neuentwicklung einer Komponente sinnvoller sein kann als eine Portierung, etwa wenn eine verwendete zentrale Bibliothek in der Zielumgebung nicht einsetzbar ist, wird auch dieser Aspekt adressiert. Ein häufig schwieriger, aber dennoch erfolgskritischer Bestandteil



© Zühlke 2009

Abbildung 1: Vorbereitungsphase der Migration (in BPMN Notation)



© Zühlke 2009

Abbildung 2: Durchführungsphase der Migration (in BPMN Notation)

einer Migration ist die Validierung. Wurden die Migrationsziele erreicht? Diese Aussage sollte aus fachlicher und technischer Sicht (funktionale und nichtfunktionale Anforderungen) über entsprechende – idealerweise automatisch durchführbare – Tests bestätigt werden.

Entsprechend der Planung arbeitet man die Migrations-Teilziele sukzessive ab. Der iterative Ansatz mit kontinuierlichen Tests ermöglicht es, den Fortschritt des Migrationsprojekts zeitnah zu überwachen und beim Auftreten von Problemen kurzfristig zu reagieren. Parallel zu den eigentlichen Migrationsarbeiten müssen Lösungen für die Produktionsumgebung gefunden werden. Dazu gehören das Aufsetzen, Anpassen oder Erweitern von Monitoring-, Clustering- und Load-Balancing-Lösungen.

Der Application Server stellt das Rückgrat jeder Enterprise-Applikationslandschaft dar. Dass dieser nicht ohne Aufwand austauschbar ist, sollte jedem IT-Verantwortlichen klar sein. Trotzdem gibt es für Anwender des Oracle Containers für J2EE (OC4J) zwingende Gründe, diesen Schritt zu gehen (siehe [7]):

Application Server

Der Application Server stellt das Rückgrat jeder Enterprise-Applikationslandschaft dar. Dass dieser nicht ohne Aufwand austauschbar ist, sollte jedem IT-Verantwortlichen klar sein. Trotzdem gibt es für Anwender des Oracle Containers für J2EE (OC4J) zwingende Gründe, diesen Schritt zu gehen (siehe [7]):



Trainings bei TEAM:
www.team-pb.de

TEAM

TEAM - Ihr Partner für innovative IT-Lösungen

- Business Intelligence
- ADF Rich Client
- Enterprise 2.0
- WebLogic Server
- Application Express
- DB Administration
- Hochverfügbarkeit
- richtig Lizenzieren

Starten Sie den Dialog mit uns!

TEAM
Partner für Technologie und
angewandte Methoden der
Informationsverarbeitung
GmbH

Hermann-Löns-Straße 88
33104 Paderborn

Mail oracle@team-pb.de
Web www.team-pb.de

Fon +49 5254 8008-0
Fax +49 5254 8008-19

ORACLE CERTIFIED ADVANTAGE
PARTNER

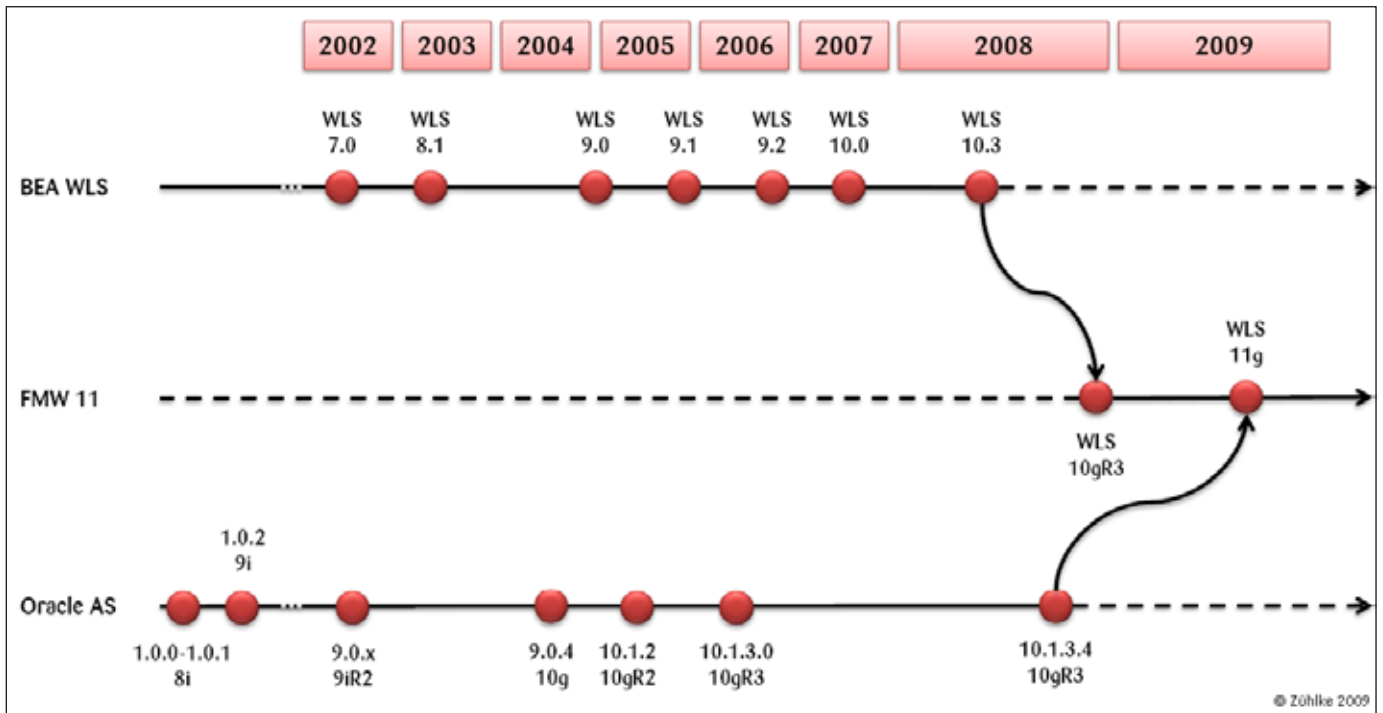


Abbildung 3: Merge der Application Server von BEA und Oracle

- Der OC4J ist kein strategisches Produkt und wird künftig nicht mit derselben Intensität wie der WebLogic Server weiterentwickelt
- Die Kernbestandteile der Oracle Fusion Middleware basieren in Zukunft auf dem WebLogic Server
- Der OC4J bietet im Vergleich zu WebLogic nur eingeschränkte Administrationsunterstützung
- WebLogic bietet im Vergleich zu OC4J eine umfangreichere Deployment- und Clustering-Funktionalität

Abbildung 3 stellt die Zusammenführung von Oracles OC4J und dem WebLogic Server dar. Aus dem OC4J werden lediglich ausgewählte Features übernommen.

Die Migration einer Anwendung von OC4J zu WebLogic verursacht wesentlich mehr Aufwand als das Anpassen des Deployment-Deskriptors. „Write once, run anywhere“ ist Suns Werbeslogan für Java. Nicht ganz zu Unrecht wurde unter Entwicklern daraus „write once, test everywhere“. Letzteres trifft besonders dann zu, wenn Java-Anwendungen in unterschiedlichen Umgebungen ausgeführt werden. So wird neben dem Application Server in vielen Fällen durch die Hintertür auch

die JVM ausgetauscht, da der WebLogic Server im Normalfall mit der BEA-JVM JRockit betrieben wird.

Fairerweise muss an dieser Stelle ergänzt werden, dass Applikationen, die sich vollständig an Java-EE-Standards halten, wesentlich weniger Migrationsaufwand verursachen. In der Praxis gibt es allerdings nur wenige dieser Muster-Anwendungen. So erfordert der hohe Konfigurationsanteil im EJB2.x-Standard herstellerspezifische Einstellungen. Außerdem werden im Entwicklungsalltag häufig nicht standardisierte Features der Application Server verwendet, um Zeit zu sparen und das Rad nicht permanent neu zu erfinden. Letztlich sind es gerade die Server-spezifischen Features, die die Stärken der einzelnen Produkte ausmachen. Einige Beispiele für häufig eingesetzte, nicht standardisierte APIs im OC4J-Umfeld sind:

- Spezielle Persistenz-Features (oracle.toplink.*)
- Security (z.B. oracle.security.jazn.*)
- Support-Klassen für JSP (oracle.jsp.*)

Enterprise Service Bus

Nach den SOA-Initiativen der vergangenen Jahre haben viele Unternehmen

einen Enterprise Service Bus (ESB) im Einsatz. Der Oracle Enterprise Service Bus (OESB) wird künftig als Mediator, also Vermittlungskomponente in der SOA-Suite aufgehen. Wie in Abbildung 4 dargestellt, wurde der BEA Aqualogic Service Bus (ALSB) in „Oracle Service Bus“ (OSB) umbenannt und zum strategischen ESB-Produkt bestimmt. In diesem Zusammenhang lohnt sich in folgenden Fällen eine Migration zum OSB:

- Mittlere bis hohe Anforderungen an die Performanz des Service Bus
- Erweiterte Routing- und Transformationsfunktionalität notwendig
- Ein vollständiger und integrierter SOA-Stack (Service Bus, Registry, Repository, BPM) erforderlich

Die Migration eines Enterprise Service Bus (ESB) ist besonders kritisch, da er einen „Single Point of Failure“ darstellt. Während die Migration mehrerer Applikationen von OC4J zum WebLogic Server sukzessive erfolgen kann und somit immer nur eine Anwendung betroffen ist, hat die Migration des ESB Auswirkungen auf eine Vielzahl von Anwendungen. Um die Risiken einer solchen Migration zu reduzieren, müs-

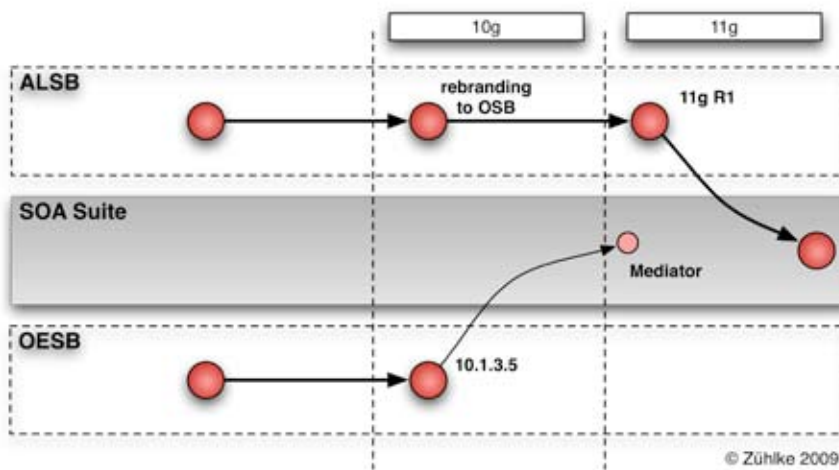


Abbildung 4: Der OESB wird zum Mediator, der ALSB als strategisches Produkt positioniert

sen zunächst die Mitarbeiter im Umgang mit dem neuen Produkt geschult werden, denn die Konfiguration von OESB und OSB ist grundverschieden. Auch die Integration von Registry, Repository und BPM-Suite verläuft unterschiedlich, so dass in diesen Bereichen zusätzlicher Aufwand entsteht. Für das Monitoring des ESB muss eine

bereits existierende Lösung bei Bedarf angepasst werden. Die Migration vom OESB zum OSB umfasst vor allem die folgenden fünf Schritte:

- **Erzeugen von Testdaten**
Dazu werden in einem definierten Zeitraum alle Nachrichten, die über das Altsystem laufen, aufgezeichnet.

- **Aufbau der neuen Umgebung**
Die Konfiguration des OESB wird auf den OSB abgebildet. Dieser Schritt ist sehr einfach zu bewerkstelligen, falls lediglich Gateways (OESB) auf Proxy-Services (OSB) abgebildet werden müssen. Aufwändiger wird der Migrationsschritt, sofern erweiterte Funktionalität des OESB wie XSLT-Transformationen zum Einsatz kam. Diese müssen manuell auf XQuery-Transformationen im OSB abgebildet sein.
- **Test der Routingkonfiguration**
Ob das Routing im neuen System wie erwartet funktioniert, wird mit gemockten Services und den in Schritt 1 erfassten Testdaten überprüft.
- **Last- und Performance-Test**
Der Service-Bus muss ausführlich getestet werden, um die erreichbare Performance und die Lastgrenzen des Systems zu ermitteln.
- **Inbetriebnahme**
Bereitstellung der Produktivumgebung und Umschwenken des Traffics auf den neuen ESB, beispielsweise durch Änderung der DNS-Einträge.

MuniQSoft GmbH – Ihr ORACLE® Spezialist!

Consulting

- IT-Consulting:**
- Datenbankmanagement
 - Administration
 - Migration
 - Backup & Recovery
 - Patch-Management
 - Betriebsführung
 - Performance Tuning
 - Hochverfügbarkeit:
 - Real Application Clusters
 - Failsafe, Data Guard
 - Oracle Datenbank Security
 - Oracle Grid Control

- Services:**
- Remote-DBA Service
 - Telefonhotline

Schulung

- Überblick:**
- Oracle SQL & PL/SQL
 - Datenbankadministration
 - Backup & Recovery
 - Neu in Oracle 10g/11g
 - Datenbank Tuning
 - Datenbank Security
 - Datenbank Monitoring
 - APEX

Wir bieten öffentliche Kurse sowie Inhouse-Schulungen

Entwicklung

- Individualsoftware:**
- PL/SQL
 - Application Express (HTML DB)
 - Visual Basic
 - .NET
 - J2EE
 - Java

- Produkte:**
- OraDB.SchemaDoc

Lizenzen

- Oracle Produkte:**
- Oracle Datenbank:
 - Enterprise Edition
 - Standard Edition (One)
 - Personal Edition
 - Internet Application Server
 - Enterprise Manager
 - Oracle Tools
 - u.v.m.

- Lizenzinformation über:**
- Named User Plus Lizenzen
 - Prozessorlizenzen
 - Mindestlizenzierung
 - Entwicklungsumgebung
 - Standby- und Clusterumgebung
 - u.v.m.



Kompetenz ist unsere Stärke. Kundenservice unsere Philosophie.

MuniQSoft GmbH Grünwalder Weg 13a 82008 Unterhaching Telefon: 089 / 67 90 90 40
<http://www.muniqsoft.de> E-Mail: info@muniqsoft.de



[6])

Das Umschwenken ist möglich, da sich die Schnittstellen für die Anwendungen als Service-Konsumenten nicht ändern. Die Altumgebung bleibt verfügbar, bis der neue Service Bus mit allen betroffenen Anwendungen ausführlich getestet wurde. Im Fehlerfall kann wieder auf das Altsystem zurückgerüstet werden.

Portal

Im Portal-Bereich existieren die Produkte Oracle Portal, WebLogic Portal und WebCenter. Das strategische Produkt ist Oracle WebCenter. Das WebLogic Portal wird zunächst als selbstständige Software weiter entwickelt und ist ebenfalls als ein Bestandteil der WebCenter Suite erhältlich. Eine Migration zu beziehungsweise eine Einführung von WebCenter lohnt sich vor allem, wenn zusätzliche Kollaborationswerkzeuge (Blog, Wiki, Instant Messaging und Social Networking) in das Portal integriert werden sollen.

Ein Portal fasst eine Vielzahl von Anwendungen (Portlets) zu einer personalisierten Webseite zusammen. Diese unterstützt den jeweiligen Anwender bei der Bewältigung seiner spezifischen Aufgabenpalette. Eine Migration vom Oracle oder Weblogic Portal zum WebCenter besteht demnach vor allem aus einer Migration der Portlets. Diese wird vereinfacht, falls man bei der Implementierung der Portlets auf Standards gesetzt hat, die von beiden

Portallösungen unterstützt werden, typischerweise dem Java Specification Request 168 (JSR-168, siehe [6]). Allerdings deckt JSR-168 nicht alle bei Portal-Implementierungen relevanten Aufgabenstellungen ab, so dass sich in vielen Fällen ein Anpassungsbedarf neben dem JSR-168 ergibt. Portlets, die sich nicht an den JSR-168 halten, sind mit hoher Wahrscheinlichkeit zu großen Teilen neu zu entwickeln.

Ein generelles Problem von Portal-Lösungen besteht darin, dass zusammen mit dem Portal alle Portlets eingesetzt werden. Deshalb muss bei jeder kleinen Änderung an einem einzigen Portlet das gesamte Portal neu getestet, in die Produktionsumgebung übergeben und eingesetzt werden. „Web Services for Remote Portlets“ (WSRP) bietet eine Entkopplung zwischen Portlet und dem Portalserver (siehe Abbildung 5). Dank WSRP kann das eigentliche Portlet (z.B. CRM) auf einem vom Portalserver über Webservices erreichbaren System ausgeführt werden. Der Portalserver kommuniziert mit dem Portlet über SOAP und übernimmt nur noch dessen Darstellung.

Daher sollte man während der Migration von Portlets prüfen, ob diese auf WSRP umgestellt werden kann, was sich vor allem bei änderungsanfälligen Anwendungen lohnt. Der größte Nachteil von WSRP besteht in den höheren Hardware-Anforderungen, da für die Darstellung einer Webseite mehrere Webservice-Aufrufe zu tätigen

sind. Der Mehraufwand ist abhängig von der Anzahl der Portlets, die über WSRP eingebunden werden. Detaillierte Informationen zu WSRP sind unter [2] und [3] verfügbar.

Fazit

Standards wie Java EE und JSR-168 vereinfachen die Migration, falls sich die zu migrierende Applikation vollständig an diese Standards hält, was jedoch nur selten der Fall ist. Besonderes Augenmerk muss auf Anforderungen wie Skalierbarkeit, Verfügbarkeit und Performance der Anwendungen gelegt werden. Zur Umsetzung dieser Anforderungen wird häufig auf spezifische Mechanismen der eingesetzten Produkte zurückgegriffen; gerade diese sind bei der Migration sorgfältig zu beachten.

Migrationsprojekte werden häufigen passant angegangen, nicht methodisch und strukturiert – und schlagen genau deshalb fehl. Ein konkreter Grund dafür ist der Verzicht auf die Ausarbeitung einer Strategie. Auch die Definition von Abnahmekriterien und Testfällen wird häufig vernachlässigt. Dass an eine erfolgreiche Migration bei späterer Bereitstellung der Infrastruktur nicht mehr zu denken ist, wird gerne ignoriert. Der Weg zur erfolgreichen und effizienten Migration führt hingegen über eine sorgfältig geplante Vorgehensweise, mit starkem Fokus auf Qualitätssicherung und Risiken.

Quellen

- [1] Oracle Service Bus Strategy: <http://www.oracle.com/technology/products/integration/service-bus/docs/Oracle-Service-Bus-SOD.pdf>
- [2] WSRP Specification: <http://docs.oasis-open.org/wsrp/v2/wsrp-2.0-spec-os-01.html>
- [3] Introduction to WSRP: <https://www.ibm.com/developerworks/webservices/library/ws-wsrp/>
- [4] Netbeans Website: <http://www.netbeans.org/>
- [5] Eclipse Website: <http://www.eclipse.org/>
- [6] Java Specification Request 168: <http://jcp.org/aboutJava/community-process/final/jsr168/>
- [7] Oracle Middleware Briefing: <http://www.oracle.com/products/middleware/docs/oracle-middleware-strategy-briefing-072008.pdf>

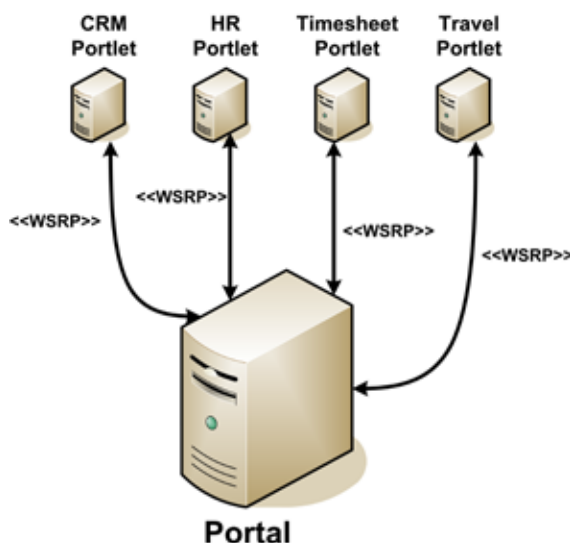


Abbildung 5: Web Services for Remote Portlets

Kontakt:

Matthias Pfau
matthias.pfau@zuehlke.com

Entwicklung von Benutzeroberflächen mit ADF Faces

Detlef Müller, ORACLE Deutschland GmbH

Der letzte Teil der Artikelserie über das Oracle Application Development Framework (ADF) stellt die Entwicklung von Benutzeroberflächen in den Mittelpunkt. Anhand der im ersten Teil vorgestellten Beispielanwendung „Veeva Vacation“ wird die Entwicklung von komponentenbasierten Oberflächen mit ADF Faces sowie deren Anpassung in Funktion und Layout erläutert. Das aktuelle ADF Release 11g enthält mit den ADF Rich Client Components eine Vielzahl von funktional angereicherten Oberflächenkomponenten, mit denen sich moderne, hoch interaktive Anwendungen erstellen lassen können.

In diesem vierten Artikel der Serie über ADF geht es darum, die Datenstrukturen und die Geschäftslogik einer Anwendung in modernen, webbasierten Benutzeroberflächen darzustellen. Zuvor eine Zusammenfassung über die vorangegangenen Teile der Artikelserie.

Der erste Artikel (DOAG News Q3/2008) gab einen Überblick über Oracle ADF. Im Mittelpunkt standen Fragen wie „Was ist ADF?“, „Woraus besteht ADF?“, „Was kennzeichnet ADF?“, „Was ist ein Model View Controller (MVC) Entwurfsmuster?“. Es wurde erläutert, wie der Oracle JDeveloper als Entwicklungsumgebung eine visuelle, deklarative Arbeitsweise durch viele Assistenten besonders unterstützt. Darüber hinaus wurde eine kleine Reisebuchungsanwendung „Veeva Vacation“ vorgestellt, die in allen Artikeln als durchgängiges Beispiel diente.

Der darauffolgende zweite Teil (DOAG News Q4/2008) beleuchtete Oracle ADF aus Sicht eines klassischen Forms-Entwicklers. Er stellte heraus, dass ADF eine moderne Alternative zu Oracle Forms darstellen kann und mit ADF eine Transition hin zu einem JEE-basierten Softwaredesign ermöglicht wird, ohne automatisch in die tiefen Gründe der Java/JEE-Programmierung einsteigen zu müssen. Besonderer Wert wurde auf eine Analogie zur gewohnten Entwicklungsweise mit Oracle Forms gelegt und bestimmte ADF Konzepte mit denen von Oracle Forms verglichen. Es wurde auch eine Möglichkeit der Transformation von Oracle Forms Anwendungen nach Oracle ADF über das Framework JHeadstart vorgestellt.

Der dritte Artikel (DOAG News Q2/2009) beschäftigte sich mit der Persistenzierung von Daten- und Objekt-Strukturen, also wie man Datenbanken oder sonstige Backend-Systeme in der JEE-Anwendungswelt adressiert oder wie ein Caching erfolgt. Einen großen Raum nahm dabei die Java Persistence API (JPA) Implementierung „Eclipse-Link JPA“ ein. Daten und Datenstrukturen werden so für ADF-Applikationen verfügbar gemacht.

Das „V“ in MVC – die View Ebene

Dieser Artikel zeigt zunächst noch einmal das ADF-Architekturschaubild und die einzelnen Ebenen des Model View Controller (MVC) Entwurfsmusters (siehe Abbildung 1). Hier wird der Sinn eines MVC Design Patterns deut-

lich. Man kann verschiedene Typen von Oberflächen auf unterschiedlicher technologischer Basis und für verschiedene Zugangskanäle entwickeln, ohne die eigentliche Business-Logik der Anwendung verändern zu müssen. Die Beispiel-Anwendung „Veeva Vacation“ könnte somit sowohl als Swing-Applikation, als auch als reine HTML-Anwendung oder mobile Applikation bereitgestellt werden, ohne die darunter liegende Logik der Datenverarbeitung verändern zu müssen. Im Weiteren wird aber auf Swing oder mobile Frontends nicht näher eingegangen.

Komponentenbasiertes Entwickeln von Oberflächen

Die Beispiel-Anwendung „Veeva Vacation“ wird hier mit ADF Faces in Form

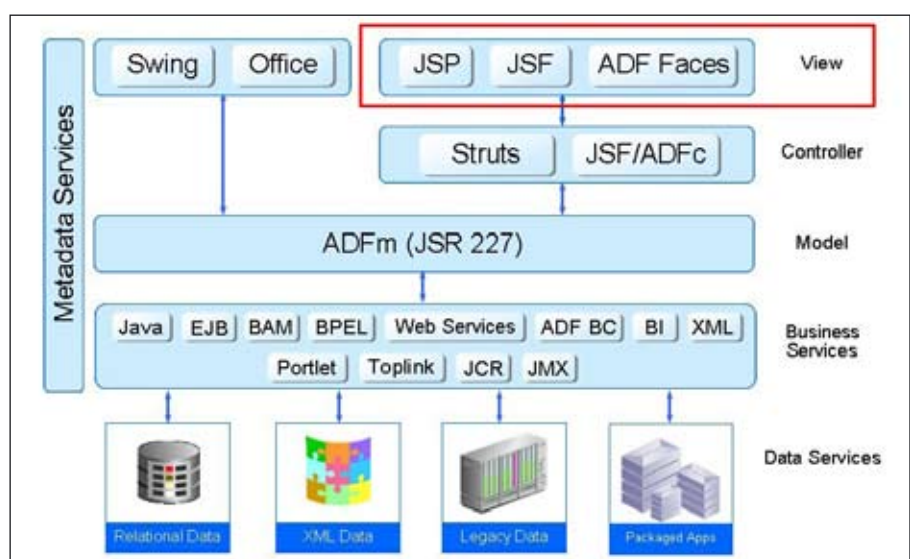


Abbildung 1: Die Prinzip-Architektur von ADF

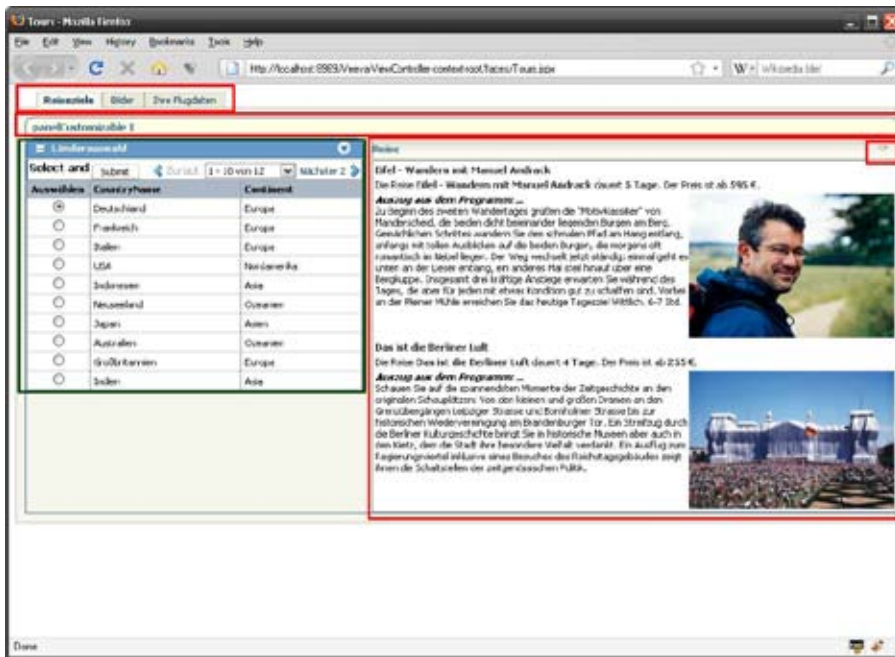


Abbildung 2: Oberflächenkomponenten in der Beispiel-Anwendung „Veeva Vacation“

einer HTML-basierten Web-Oberfläche dargestellt. Wir sehen farblich gekennzeichnete Bereiche zur Strukturierung des Seitenaufbaus, wie zum Beispiel Tab-Reiter, Seitenregionen zum Auf- und Zuklappen, Paneele oder Arbeitsbereiche, Portlet-ähnliche Seitenelemente etc.

Insgesamt präsentiert sich dem Anwender die gesamte Oberfläche als eine in sich geschlossene Einheit, obwohl die Inhalte aus verschiedenen Service-Quellen bereitgestellt sind. Ein Beispiel ist die Auflistung einzelner Reiseländer auf der linken, grün umrandeten Bildhälfte. Diese Daten entstammen einem Datenbank-Zugriff über ADF Business Components (siehe Teil 1 und 2 der Artikelserie) und werden als HTML-Tabelle mit Auswahloptionen dargestellt. Dahinter steckt ein HTML-Form-Tag mit dazugehöriger Radio Group, dem Submit-Button, sowie der JavaScript-Funktion zur Spaltensortierung und zum Abhandeln der Zeilenanzahl pro Seite. Der rechte, rot umrandete Bereich ist ein Auszug aus einem Content Store, der, architektonisch betrachtet, außerhalb dieser Anwendung in einem Content Management System entsteht und als fertiges HTML-Fragment in die Oberfläche eingebunden werden soll. Die Einbettung erfolgt hier in Form eines Portlets.

Aus Sicht des Entwicklers ist es sinnvoll, den modularen Aufbau der gesamten Applikations-Architektur (siehe Abbildung 1) auch bis zur Ebene der Darstellung in entsprechender Baukastenweise fortzusetzen. Eine Read-Only-Tabelle, wie hier grün gekennzeichnet, sollte beispielsweise schon eine fertige Oberflächen-Komponente sein, die nur noch in die Applikation eingesetzt und mit der Datenbehandlung im Model verbunden wird. Funktionale Anreiche-

rungen in der Tabelle, wie zum Beispiel die JavaScript-Funktion zur Spaltensortierung oder die Verwendung von Stylesheets, etc. werden bereits innerhalb der Komponente umgesetzt. Für eine komponentenbasierte Oberflächenentwicklung gibt es nun verschiedene JEE-basierte UI-Frameworks wie Java Server Faces JSF oder eben auch ADF Faces.

Komponentenbasierte Entwicklung zielt auf Wiederverwendbarkeit ab. Jede Komponente muss daher eine bestimmte Flexibilität aufweisen, um für verschiedene Einsatzfälle wiederverwendet werden zu können. Daher sind die Komponenten in der Regel parametrisierbar. Diese Parameter, auch Properties genannt, werden entweder zum Zeitpunkt der Verarbeitung durch den Anwendungsentwickler belegt oder können dynamisch zur Laufzeit über Java-Klassen in den sogenannten „Managed Beans“ verändert werden. Dazu mehr im nächsten Abschnitt. Dem Entwickler stellt sich die Bibliothek von Layout-Komponenten im JDeveloper wie in Abbildung 3 dar.

Neben ADF Faces stehen hier gleichermaßen und gleichberechtigt auch UI-Komponenten aus weiteren Standard-Frameworks wie zum Beispiel JSF oder Apache Trinidad zur Verfügung. Der Entwickler kann also aus verschiedenen Frameworks auswählen und diese bei Bedarf mischen. Um trotz ho-

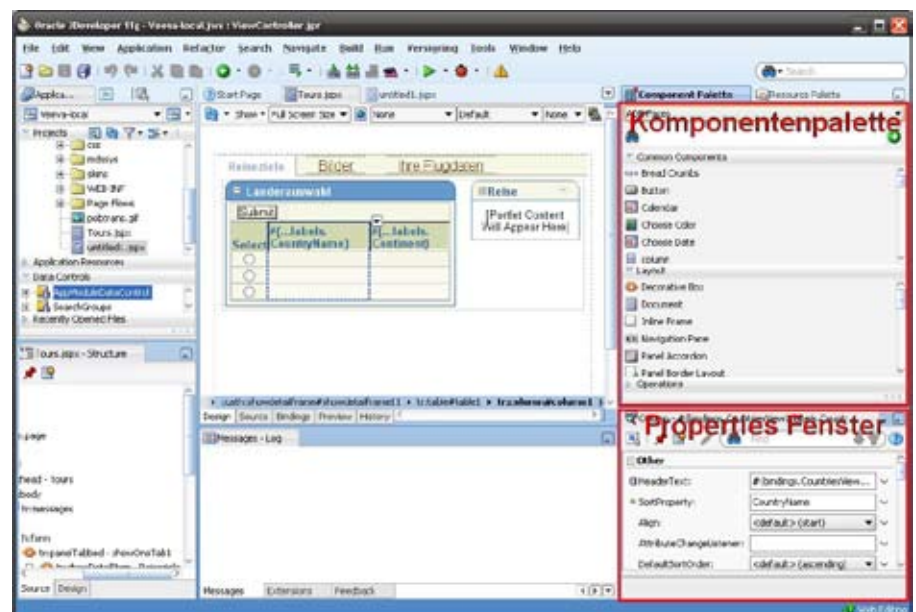


Abbildung 3: Auswahl von UI-Komponenten und Property-Palette im JDeveloper

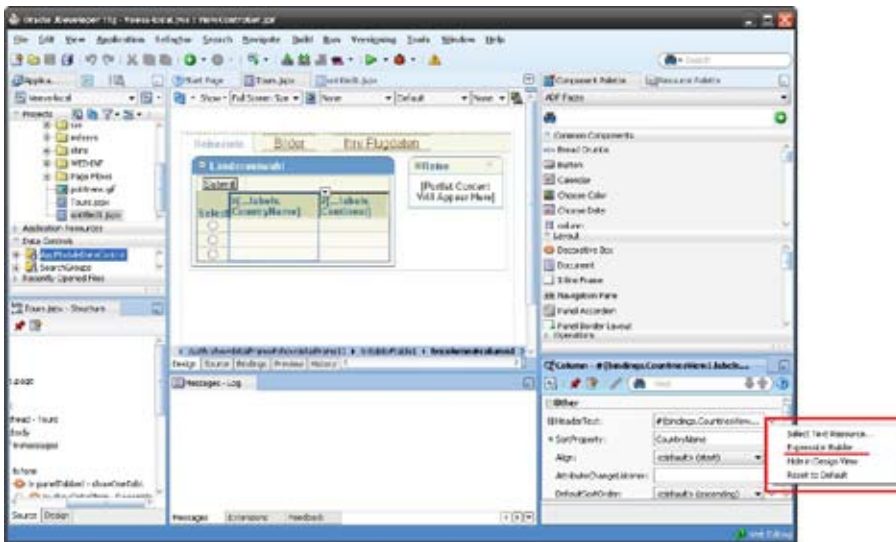


Abbildung 4: Verwendung eines Ausdrucks (Expression) für Properties

her Vielfalt an ADF Faces UI-Komponenten die Übersichtlichkeit zu wahren, werden diese im JDeveloper nach Typen strukturiert. Eine komplette, interaktive Übersicht über alle ADF Faces Komponenten inklusive Property-Liste ist entweder online unter http://www.oracle.com/technology/products/adf/adffaces/11/doc/demo/adf_faces_rc_demo.html zu finden oder kann dort als Offline-Beispielanwendung geladen werden.

Dynamisierung der UI-Komponenten – Expression Language und Managed Beans

Die Properties der jeweiligen Layout-Komponente lassen sich, wie bereits genannt, statisch zum Zeitpunkt der Entwicklung der Oberfläche festlegen. Damit ist die Verarbeitung der UI-Komponenten durch den Entwickler definiert. In vielen Fällen aber muss sich die UI-Komponente dynamisch verhalten, das heißt, bestimmte Properties kann man erst zur Laufzeit festlegen. Das trifft dann zu, wenn beispielsweise durch Interaktionen des Anwenders, wie die Selektion von Daten aus einer Werteliste, bestimmte Textfelder einer anderen UI-Komponente ein- oder ausgeblendet, durch Sicherheitsanforderungen bestimmte Felder von „Read Write“ auf „Read Only“ umgesetzt werden sollen oder bestimmte Informationen, wie Beschriftungen, auf Berechnungen basieren. Eine solche

Dynamisierung kann man durch Verwendung der sogenannten „Expression Language“ in der Property-Definition der UI-Komponente erreichen. Das bedeutet, dass bestimmte Parameterwerte der betreffenden Komponente nicht statisch festgelegt sind, sondern dafür ein Ausdruck (Expression) verwendbar ist. Dieser legt dann zur Laufzeit die entsprechende Property fest. Expressions deklariert man mit #{funktionaler Ausdruck}. Sie können auch Aufrufe von Java-Klassen beinhalten.

In diesem Zusammenhang lassen sich sogenannte „Managed Beans“ verwenden. Dies sind durch das Framework verwaltete Java-Klassen mit einem Bezug zu Oberflächen-Kom-

ponenten. JS- Seiten können also eine Verknüpfung mit solchen Managed Beans erhalten, die gegebenenfalls bestimmte Eigenschaften von UI-Komponenten zur Laufzeit verändern. So ist es zum Beispiel möglich, Wertevorgaben eines Formularfeldes dynamisch zu ermitteln und als Property Value in die UI-Komponente zu übertragen oder es kann bei Verlassen des Eingabefeldes eine Aktion ausgelöst werden, die andere Seitenelemente beeinflusst. Die Vielfalt solcher Dynamisierungen ist nahezu unbegrenzt.

Formatierung über Stylesheets

Nachdem der grundsätzliche Aufbau der Seiten über UI-Komponenten geklärt ist, stellt sich nun die Frage, wie man das Erscheinungsbild der Komponenten in die gewünschte Form bringen kann. ADF Faces ermöglicht hierzu die Definition von sogenannten „Skins“ auf Basis der CSS 3.0 Syntax. Jede UI-Komponente beinhaltet neben der funktionellen Ausgestaltung auch die Vorgabe von CSS-Klassen. Diese können in einer eigenen CSS-Definitionsdatei (Skin-Definition) angegeben und der Anwendung beigefügt werden. Die Skin-Definition behandelt ausschließlich Layout- und Stileigenschaften wie Fonts, Farben, Hintergründe, Icons und sonstige grafische Eigenschaften der UI-Komponenten. Die Einträge in der Skin-Definition werden

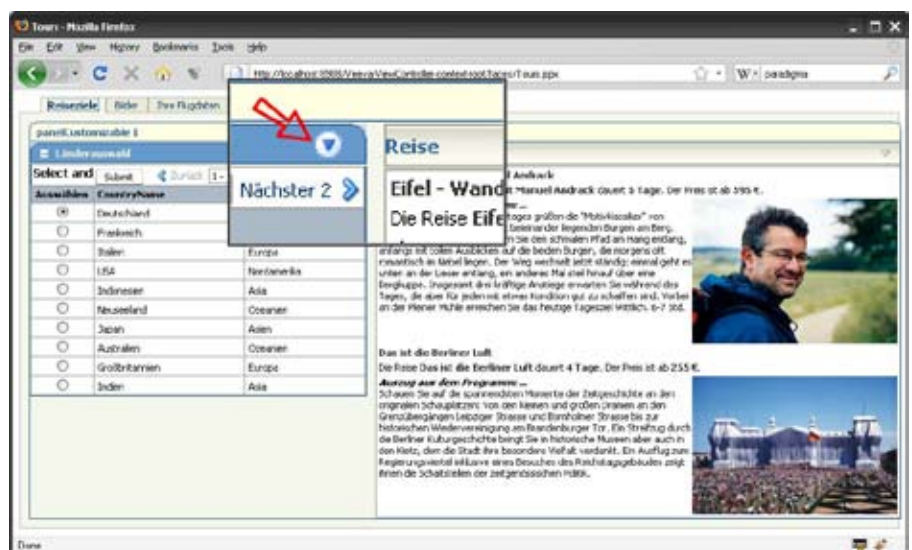


Abbildung 5: Ein Icon Selector der showDetailFrame-Komponente

als „Selectors“ bezeichnet. Man unterscheidet zwischen Style Selectors und Icon Selectors.

Ein Beispiel für Icon Selectors ist in Abbildung 5 zu sehen. Hier wurde die Komponente ShowDetailFrame verwendet, die ein Auf-und Zuklappen des Seitenbereiches mit der Wertetabelle ermöglicht. Über den hier rot gekennzeichneten Schalter kann man die Komponente bedienen. Dieser Schalter wird über ein Icon „Selector“ im Skin definiert. Das Listing zeigt einen Auszug aus der CSS-Datei für die showDetailFrame-Komponente (siehe Listing).

In dem Beispiel ist zu erkennen, dass jede Komponente drei verschiedene gra-

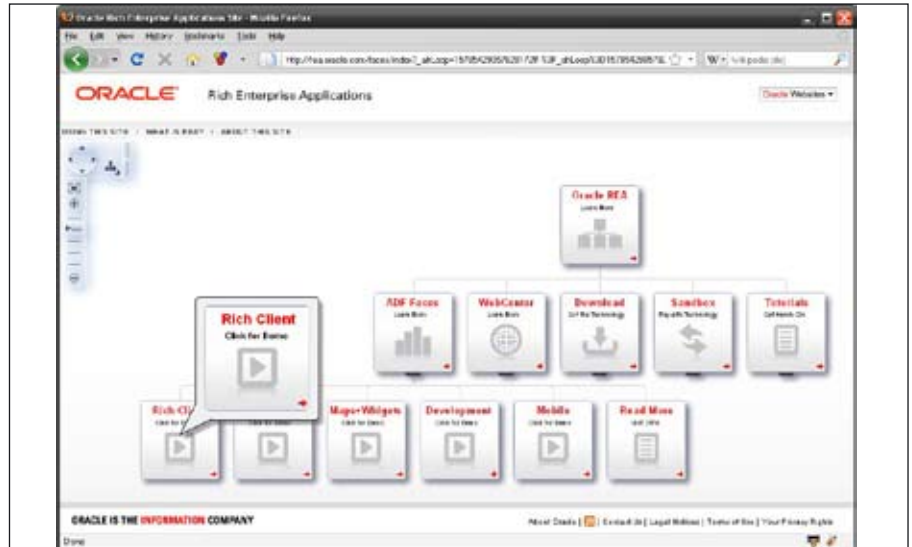


Abbildung 6: Beispiel eines Hierarchie-Browsers auf Basis von Adobe Flash

```

...
/** ----- **/
/** showDetailFrame **/
/** ----- **/
/** showDetailFrame Style Selector **/
howDetailFrame::content-light
{
  -ora-rule-ref:selector(„.BackgroundLight:alias“);
  -ora-rule-ref:selector(„.ColorBlack:alias“);
  border-left:1px -ora-rule-ref:selector(„.ValueDark:alias“) solid;
  border-right:1px -ora-rule-ref:selector(„.ValueDark:alias“) solid;
  border-bottom:1px -ora-rule-ref:selector(„.ValueDark:alias“) solid;
}

af|showDetailFrame::content-medium
{
  -ora-rule-ref:selector(„.BackgroundLight:alias“);
  -ora-rule-ref:selector(„.ColorBlack:alias“);
  border-left:1px -ora-rule-ref:selector(„.ValueDark:alias“) solid;
  border-right:1px -ora-rule-ref:selector(„.ValueDark:alias“) solid;
  border-bottom:1px -ora-rule-ref:selector(„.ValueDark:alias“) solid;
}

af|showDetailFrame::content-dark
{
  -ora-rule-ref:selector(„.BackgroundMedium:alias“);
  -ora-rule-ref:selector(„.ColorWhite:alias“);
  border-left:1px -ora-rule-ref:selector(„.ValueLight:alias“) solid;
  border-right:1px -ora-rule-ref:selector(„.ValueLight:alias“) solid;
  border-bottom:1px -ora-rule-ref:selector(„.ValueLight:alias“) solid;
}

...

/** showDetailFrame Icon Selector **/
showDetailFrame::light-ActionsIcon:alias {content:url(“/skins/mySkin/
images/ShowDetailFrame/light-ActionsIcon.png“);}
showDetailFrame::medium-ActionsIcon:alias {content:url(“/skins/mySkin/
images/ShowDetailFrame/medium-ActionsIcon.png“);}
showDetailFrame::dark-ActionsIcon:alias {content:url(“/skins/mySkin/
images/ShowDetailFrame/dark-ActionsIcon.png“);}
...

```

fische Abstufungen enthält (Dark, Medium, Light), die über eine entsprechende Komponente „Property“ aktiviert sind und für die es dann auch entsprechende Einträge in der Skin-Definition geben muss. Außerdem wird hier sichtbar, dass man CSS-Klassen auch über „-ora-rule-ref:selector(„<klassenname>“);“ referenzieren kann. So sind beispielsweise wiederkehrende Farb-Definitionen für die Layout-Modi „dunkel“, „mittel“ und „hell“ bereits an anderer Stelle in der CSS-Definitionsdatei angegeben und werden hier wiederverwendet.

Man muss übrigens keine komplette Skin-Definition einrichten. Es gibt im ADF Renderer (dem Baustein, der für die Erstellung des HTML-Codes zuständig ist) Standard-Formatierungen, die dann angewendet werden, wenn kein eigener Skin vorhanden ist.

Rich Enterprise Applications

Immer häufiger findet man hochdynamische Webanwendungen vor, die Client-seitig Funktionalitäten ausführen lassen. Dabei setzen sich AJAX (Asynchronous JavaScript and XML) und Adobe Flash als Technologie immer stärker durch. Oracle ADF Faces besitzt eine Vielzahl solcher dynamischen Komponenten. Gerade mit dem Release 11g sind viele neue Komponenten zur Datenvisualisierung eingeführt worden. Ein Beispiel für die dynamische Anwendung von hierarchisch

Houston, wir haben ein Problem!

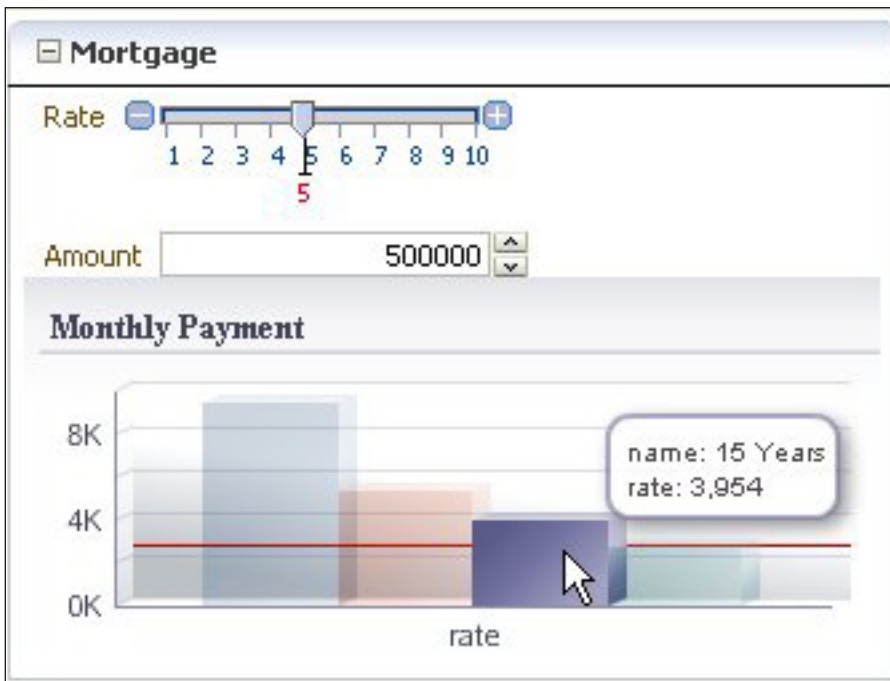


Abbildung 7: UI-Komponenten mit AJAX

strukturierten Daten ist auf der Seite <http://rea.oracle.com> zu finden.

Hierbei wird ein Flash-basierter Hierarchie-Browser verwendet. Man hätte hier auch auf herkömmliche Weise einen JavaScript-Hierarchiebaum anbieten können, wie er oft als klassisches Navigationselement in Websites zu finden ist. Aber mit dieser dynamisch gehaltenen Variante der Darstellung inklusive Anzeige von Detailinformationen werden ganz neue, moderne Nutzungskonzepte möglich, ohne größere Entwicklungsaufwände einzugehen. Ein Beispiel für AJAX-basierte Interaktion zwischen Komponenten ist in Abbildung 7 dargestellt. Hier sind drei verschiedenen UI-Komponenten zu sehen, die dynamisch interagieren. Die Kombination aus Werten der Slider-Komponente oben und dem darunter befindlichen Eingabefeld bewirkt eine sofortige Aktualisierung des Diagramms. ADF Faces UI-Komponenten enthalten automatisch diese Dynamisierung als Option. Von daher muss kein eigenes AJAX Framework extra eingebunden werden.

Fazit

Innerhalb der ADF-Framework-Familie sind webbasierte Benutzeroberflächen über den Oracle JDeveloper mit ADF

Faces realisiert, einer Erweiterung des JSF Frameworks. ADF-Faces-Oberflächen bestehen aus Oberflächen-Komponenten, die das eigentliche Rendering der Details übernehmen. Diese Komponenten sind parametrisch gehalten und lassen sich so variieren. Der funktionale Aufbau und die Darstellung der Oberflächendetails sind in den Komponenten implementiert. Eine grafische Anpassung erfolgt über sogenannte „Skins“, eine auf CSS-3.0-Syntax basierende Stylesheet-Definition.

Mit ADF Faces 11g wurden eine Vielzahl neuer UI-Komponenten veröffentlicht, die hochinteraktive, auf Flash und AJAX aufbauende Web-Anwendungen erstellen lassen. Das ADF Faces Rendering ist in der Lage, verschiedene Endgeräte mit passendem Source Code zu bedienen, so dass beispielsweise Anwendungen beziehungsweise Bausteine aus Anwendungen auch auf mobilen Endgeräten genutzt werden können.

Kontakte:

Detlef Müller
detlef.mueller@oracle.com
Jürgen Menge
juergen.menge@oracle.com
Kersten Mebus
kersten.mebus@oracle.com
Michael Bräuer
michael.braeuer@oracle.com

...und wir die Lösung!

IT im Alarmzustand! Fachbereiche rufen nach neuen und besseren IT-Services. Governance, Risikomanagement & Compliance sind auf der Tagesordnung. Und das IT-Budget steht auf dem Prüfstand.

Zeit, mit einem verlässlichen Partner zu sprechen, der mit Ihnen gemeinsam wirtschaftliche Lösungen entwickelt. Höchste Zeit, mit PROMATIS zu sprechen!

PROMATIS verbindet intelligente Geschäftsprozesse mit modernen Oracle Technologien und Applikationen:

- Oracle E-Business Suite
- Oracle Accelerate Mittelstandslösungen
- Oracle SOA und Application Integration Architecture
- Business Intelligence und Beehive
- Bewährte Projekt-Vorgehensmodelle und Best Practice-Prozesse

Profitieren Sie von der PROMATIS Lösungskompetenz und unserer internationalen Erfahrung. Sprechen wir darüber!

PROMATIS

Knowledge Powered Business Processes

PROMATIS software GmbH
Tel.: +49 7243 2179-0 · Fax: +49 7243 2179-99
www.promatis.de · hq@promatis.de
Ettlingen/Baden · Hamburg · Berlin

Überwachung von BPEL-Prozessen mittels Business Activity Monitoring

Gert Schüßler, ORACLE Deutschland GmbH

Das Monitoring von Aktivitäten und Prozessen in Service-orientierten Architekturen wird immer wichtiger. Informationen über Key-Performance Indikatoren und die Einhaltung von Service Level Agreements müssen in kürzester Zeit zur Verfügung stehen. Dieser Artikel zeigt anhand der SOA Suite 10g, wie man die dafür notwendigen Daten aus Prozessen an ein Business-Activity-Monitoring-Werkzeug für eine Real-Time-Überwachung übergibt.

In einer Service-orientierten Architektur (SOA) stehen unterschiedlichste Funktionen in Form von Services zur Verfügung. Diese müssen kontrolliert in einer bestimmten Reihenfolge aufgerufen und ausgeführt werden, das heißt, sie sind in klar definierte Prozesse eingebunden.

Oracle BPEL

In einer Oracle SOA steuert der BPEL Process Manager diese Prozesse. Dieser nutzt den Business-Process-Execution-Language-Standard (BPEL) und unterstützt alle gängigen SOA-Standards und -Protokolle. So ist sichergestellt, dass Prozesse schnell entwickelt und über einen langen Zeitraum hinweg überwacht und optimiert werden können.

Ein BPEL-Prozess ruft nicht nur Services auf, er führt auch unterschiedliche Aktivitäten wie die Bearbeitung von Variablen oder Schleifen aus. Bei jeder Aktivität in einem BPEL-Prozess können Sensoren und Sensor-Aktionen definiert sein, um Informationen über diese Aktivität zu sammeln. Diese Information kann beispielsweise der Zeitpunkt des Aufrufs dieser Aktivität sein (Activation) und/oder der Zeitpunkt, zu dem diese Aktivität beendet wurde (Completion).

Oracle Business Activity Monitoring (BAM)

Oracle BAM ist ein Monitoring-System, das Daten und Ereignisse aus vielen Quellen sammelt und analysiert. Standard-Anwendungen, Web-Services, Messaging Backbones und sensor-gesteuerte Netzwerke wie RFID liefern

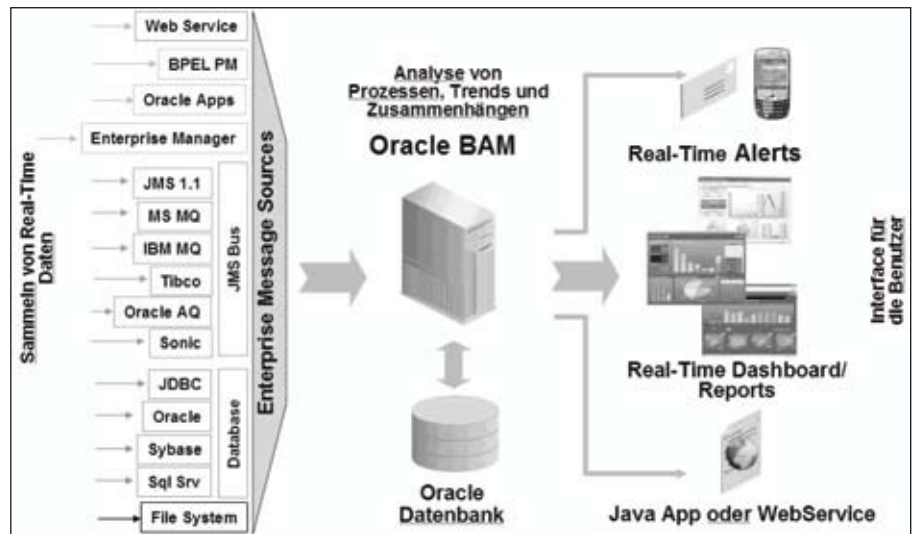


Abbildung 1: BPEL-Sensor bei einer Receive-Aktivität

diese Auskünfte. Die Informationen und Meldungen werden innerhalb weniger Sekunden ausgewertet und auf einem Dashboard angezeigt. So erhält man einen aktuellen Überblick über

Key Performance Indikatoren (KPI) sowie über die Einhaltung von Service Level Agreements (SLA) innerhalb des Unternehmens. Auf Grundlage dieser Informationen können sofort Ent-

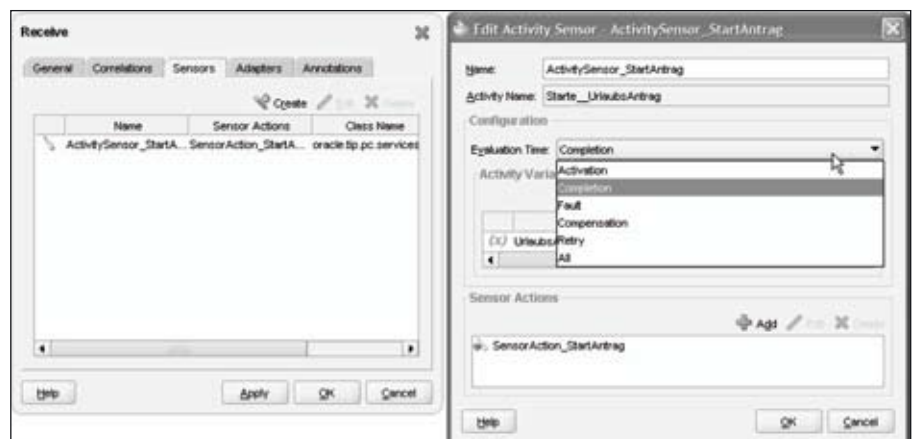


Abbildung 2: Oracle BAM – Integration unterschiedlicher Datenquellen



Abbildung 3: Oracle BAM-Dashboards

scheidungen fallen, um Korrekturen vorzunehmen. Dadurch lassen sich zum Beispiel Engpässe in der Bearbeitung von Anträgen oder Bestellungen sehr schnell erkennen und beheben.

Der Anwender sieht die Daten und Ergebnisse über Reports oder Dashboards. Die automatische Verteilung von neuen Informationen an laufende Reports und Dashboards ohne Aktion des Anwenders ist ein wichtiges Merkmal von Oracle BAM.

Alerts sind Ereignisse in einer BAM-Applikation, die im Hintergrund überwacht und ausgeführt werden. Regeln bei einem Alert legen fest, wann dieser ablaufen soll. Ein Alert kann KPIs, SLAs, Lagerbestände etc. überwachen. Als Aktion wird dann beispielsweise festgelegt, dass eine Person per Mail einen Report mit bestimmten Daten erhält, damit er diese interpretieren und daraufhin eine Aktion veranlassen kann. Wenn bei einem Bestellvorgang festgestellt wird, dass im Lager nicht mehr genügend Produkte vorhanden sind, wird ein Mitarbeiter sofort benachrichtigt, dass es eventuell zu einer Verzögerung bei der Auslieferung kommen kann. Dieser kann dann die Behebung dieses Engpasses veranlassen.

Oracle BAM stellt alle Daten für die Auswertungen in den Reports und Dashboards sowie für die Alerts online im Hauptspeicher zur Verfügung. Alle Informationen und Daten, die von den unterschiedlichen Quellen an Oracle BAM gesendet werden, sind zusätzlich in einem Repository in der

Datenbank in „Data Objects“ abgelegt. Ein „Data Object“ ist vergleichbar mit einer Tabelle in einer Datenbank. Es besteht aus verschiedenen Spalten mit unterschiedlichen Datenformaten wie Integer oder DateTime.

Kommunikation zwischen BPEL und BAM

Informationen aus BPEL-Prozessen werden über Sensor-Aktionen an BAM übermittelt. In Oracle BPEL gibt es dafür den speziellen Typ „BAM Sensor Action“. Im Dialog zum Anlegen einer „BAM Sensor Action“ bekommt man Zugriff auf das BAM-Repository, in dem man das „Data Object“ zum Ablegen der Daten aus dem BPEL-Prozess auswählen kann.

In einer XSL-Transformation legt man fest, welche Daten aus dem BPEL-

Prozess an Oracle BAM übergeben werden, indem man ein Mapping von Feldern aus den Datenstrukturen in BPEL auf die Spalten in einem BAM „Data Object“ definiert. Es können Daten über den BPEL-Prozess selber wie die InstanceID des Prozesses oder die Startzeit der Aktivität übermittelt werden. Außerdem hat man über die Sensoren Zugriff auf die Inhalte von XML-Strukturen, die in Form von Variablen im BPEL-Prozess bearbeitet werden, so dass auch Applikationsdaten wie Antragsteller oder Antragsdatum in Oracle BAM auswertbar sind.

Fazit

Die Definition von BPEL-Sensoren und der BPEL BAM Sensor Actions schafft die Voraussetzungen, um Daten aus

Field name	Field ID	Field type	Max length	Scale	Nullable	Public	Lookup	Calculated	Tip Text
Antragsnummer	__Antragsnummer	integer	-	-	Yes	Yes	-	-	-
Benutzer	__Benutzer	string	100	-	Yes	Yes	-	-	-
Antragsdatum	__Antragsdatum	datetime	-	-	Yes	Yes	-	-	-
Beginn	__Beginn	datetime	-	-	Yes	Yes	-	-	-
Ende	__Ende	datetime	-	-	Yes	Yes	-	-	-
Dauer	__Dauer	integer	-	-	Yes	Yes	-	-	-
Bemerkung	__Bemerkung	string	100	-	Yes	Yes	-	-	-
Manager	__Manager	string	100	-	Yes	Yes	-	-	-
AntragResult	__AntragResult	string	100	-	Yes	Yes	-	-	-
AntragBeginn	__AntragBeginn	datetime	-	-	Yes	Yes	-	-	-
AntragEnde	__AntragEnde	datetime	-	-	Yes	Yes	-	-	-
Antragart	__Antragart	string	100	-	Yes	Yes	-	-	-
AntragLaufzeit	__AntragLaufzeit	datetime	-	-	Yes	Yes	-	-	-

Abbildung 4: Oracle BAM Data Object

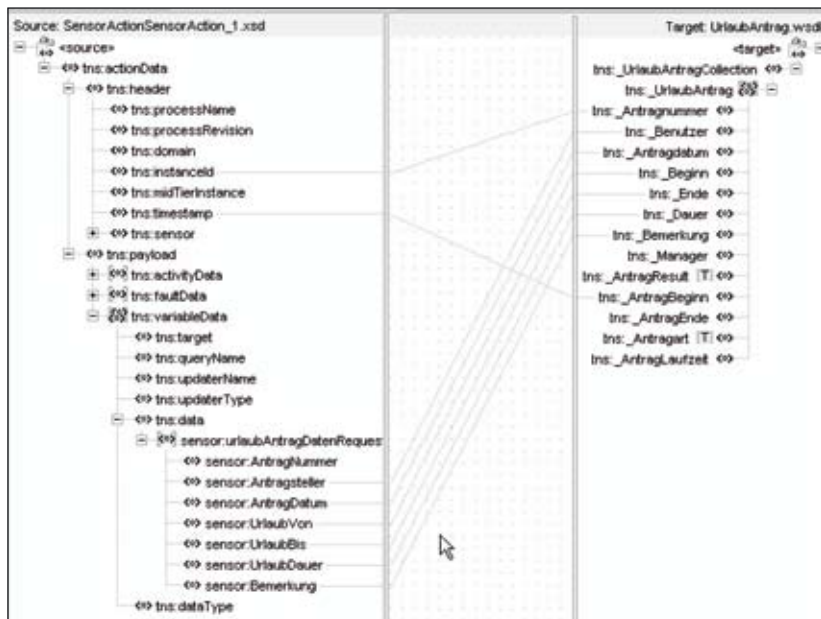


Abbildung 5: Oracle BPEL – XSL-Transformation

einem BPEL-Prozess an Oracle BAM zu schicken. Zur Laufzeit eines Prozesses sendet der BPEL-Sensor die Daten an Oracle BAM. Dazu kommt die Web-Service-Schnittstelle von BAM zum Einsatz, über welche die Daten in die „Data Objects“ eingetragen werden. Danach sind die Reports, Dashboards und Alerts für die Auswertungen definiert und stehen den Benutzern zur Verfügung. Über den BAM „Active Viewer“ oder über eine Webadresse können die Anwender auf die Reports und Dashboards zugreifen.

Kontakt:

Gert Schüssler

Gert.Schuessler@oracle.com

Wichtige Informationen zu aktuellen Lizenzfragen

Michael Paege, DOAG Competence Center Lizenzierung

SE1 Datenbank

Darf man eine SE1 Datenbank auf einem Server mit zwei Dualcore- beziehungsweise mit zwei QuartCore-Prozessoren betreiben? Beide Male handelt es sich um Intel Prozessoren (Faktor: 0,5). Wie wird für die obige Entscheidung gezählt, mit Faktor oder ohne?

Antwort: SE1 ist einsetzbar, wenn der Server nicht mehr als zwei Prozessor-sockets besitzt (bei SE maximal vier Sockets). Bei SE1 und SE werden der gefüllte Socket als Prozessor gewertet, Cores werden nicht gezählt, Faktoren kommen bei SE1 und SE nicht zur Anwendung. Für das Beispiel ist Voraussetzung, dass der Server auch nur zwei Sockets hat. Bei beiden Beispielen sind zwei CPUs zu lizenzieren.

Lizenzierung des Oracle Application Servers (Standard Edition)

Ein Unternehmen setzt zwei ESX-Server mit VM Ware im Verbund ein. Fällt einer dieser Rechner aus, werden die produktionsrelevanten virtuellen Rechner auf das

andere System geschwenkt. Wenn auf einem dieser Rechner ein virtueller Rechner mit Oracle Application Server eingesetzt wird, müssen dann die physikalischen Prozessoren dieses Rechners lizenziert oder die Prozessoren beider ESX lizenziert werden?

Antwort: Es müssen die Prozessoren aller ESX-Rechner, die im Verbund stehen, lizenziert beziehungsweise für die Berechnung des Minimums bei NUP-Lizenzierung herangezogen werden. Beim Einsatz von VM Ware sind immer alle Prozessoren des einen großen Servers beziehungsweise des gesamten Serververbunds zu lizenzieren.

VM Ware, Oracle VM und Sun Solaris Container 10

Wie sind VM Ware, Oracle VM und Sun Solaris Container 10 zu lizenzieren?

Antwort: VM Ware ist immer Softpartitioning. Bei CPU-Lizenzierung sind dann alle CPUs (entweder ein großer Server mit einzelnen VM-Ware-Parti-

tionen oder ein Server-Verbund aus mehreren Einzelmaschinen) zu lizenzieren. Bei NUP-Lizenzen und DB EE sind alle CPUs für die Berechnung des Minimums heranzuziehen. Oracle VM kann sowohl Soft- als auch Hardpartitioning, die Lizenzierung erfolgt je nach Art der Installation. Weitere Informationen dazu siehe <http://www.oracle.com/corporate/pricing/partitioning.pdf> und <http://www.oracle.com/technology/tech/virtualization/pdf/ovm-hardpart.pdf>. Sun Solaris 10 Container ist Hardpartitioning. Hierzu die Aussage von Sun: „Solaris capped Container (Zonen mit Resource-Management und zugewiesenen CPUs) gelten aus Sicht von Oracle zum Typ Hardpartitioning. Diese werden dann auch so lizenziert. Das gleiche gilt auch für LDOM und wird in sehr naher Zukunft im Metalink so erscheinen.“

Kontakt:

Michael Paege

lizenzfragen@doag.org

www.doag.org/cc/lizenzfragen

RMAN, Toad und LiteSpeed

Johannes Ahrends, Quest Software

Eine ganze Reihe von Unternehmen, die Oracle-Datenbanken im Einsatz haben, nutzen seit vielen Jahren den RMAN für ihre Sicherungen. Es gibt jedoch immer noch Administratoren, die eine Oracle-Sicherung in der klassischen Form über die Kopie von Datendateien durchführen oder den Oracle Export als Backup-Methode einsetzen. Die Gründe hierfür sind vielfältig. Oftmals ist es aber die unzureichende Kenntnis über den RMAN, die einen Administrator davon abhält, dieses sehr flexible Werkzeug einzusetzen.

Wie viele der Werkzeuge von Oracle ist auch der RMAN in gewisser Weise eine „Eierlegende Wollmilchsau“, das heißt, es gibt kaum etwas, was man mit dem RMAN nicht sichern kann. Das bedeutet aber auch, dass für 90 Prozent aller Datenbanken nur etwa 10 Prozent der Funktionalität des RMAN benötigt werden.

Vorteile von RMAN gegenüber Kopie und Export

Eigentlich sollte man den Export (auch Data Pump Export) in diesem Zusammenhang gar nicht erst erwähnen. Dies ist kein Backup&Recovery-Werkzeug, sondern dient einzig dazu, Daten zwischen Oracle Datenbanken auszutauschen. Natürlich kann es genutzt werden, um im Falle eines logischen Fehlers (DROP, TRUNCATE, DELETE etc.) einen alten Stand einer Tabelle wiederherzustellen. Allerdings gelingt dies nur für einen bestimmten Zeitpunkt (Start der Sicherung) und nur mit erheblichem zeitlichen Aufwand. Im Falle des Verlustes der gesamten Datenbank ist es außerdem erforderlich, zunächst eine Rumpf-Datenbank, bestehend aus SYSTEM, SYSAUX, UNDO und TEMP-Tablespace, zu erstellen. Erst dann kann man mit dem Importieren der Daten beginnen. Nicht selten dauert der Import mehr als 24 Stunden, für eine produktive Datenbank sicherlich keine akzeptable Zeit.

Mit einer Kopie der Datenbank-Dateien war bereits in der Version 6 auch im laufenden Betrieb (Online) eine Sicherung möglich. Dafür war und ist es erforderlich, dass sich die Datenbank im Archive-Modus befindet. Natürlich kann man die Datenbank im geschlos-

senen Zustand (offline) sichern und dementsprechend auf die Archivierung der Redo-Log-Dateien verzichten. Allerdings lässt sich die Datenbank im Fehlerfall dann auch nur auf den letzten Stand der Sicherung zurückspielen; neuere Transaktionen gehen verloren. Dieses Verfahren sollte daher in Produktionsumgebungen nicht zum Einsatz kommen.

Bei der Online-Kopie (ohne RMAN) sollte man nicht vergessen, die Datendateien zuvor in den sogenannten „BEGIN BACKUP“-Modus zu setzen. Nur so ist sichergestellt, dass die Kopie im Fehlerfall auch wieder zurückgespielt werden kann. Dieses Kommando hat nichts damit zu tun, dass die Dateien dann konsistent sind. Nur über die archivierten Redo-Log-Dateien ist es möglich, tatsächlich eine konsistente Sicherung wieder einzuspielen. Die Gründe für den Einsatz von RMAN sind naheliegend:

1. Der RMAN ist das von Oracle präferierte Werkzeug für das Backup und Recovery von Oracle Datenbanken
2. Die Benutzung des RMAN ist einfach
3. Mit dem RMAN werden standardmäßig nur die Tablespaces gesichert, die für den Betrieb der Datenbank erforderlich sind (also beispielsweise kein TEMP-Tablespace)
4. RMAN sichert nur Blöcke, in denen tatsächlich Daten liegen. Gerade bei vordefinierten Datendatei-Größen, kann dies den Umfang des Backups drastisch reduzieren
5. Es lassen sich automatisch sogenannte Backup-Sets bilden, die aus der Sicherung der Datenbank-Dateien, der bis dahin angefallenen

Redo-Log-Dateien und der notwendigen Strukturdateien (spfile, Control-Files) bestehen

6. Das Recovery ist denkbar einfach
7. Ab 11g besteht mit dem RMAN Data Recovery Advisor sogar die Möglichkeit, sich Tipps für die effektivste Recovery-Strategie geben zu lassen
8. Man kann jederzeit Informationen über die letzten Sicherungen abrufen, da RMAN Informationen über durchgeführte Backups in die Control-Files einträgt
9. RMAN verwaltet auch die archivierten Redo-Logs

Es gibt sicherlich noch weitere Gründe, die für die Benutzung des RMAN sprechen.

Praktisches Arbeiten mit RMAN

Die folgende Beschreibung beschränkt sich auf das Backup von Datenbanken ohne die Benutzung des optionalen Katalogs. Dieser ist vor allen Dingen in größeren Umgebungen mit vielen Datenbanken interessant, da es hierdurch eine zentrale Stelle gibt, in der alle relevanten Daten über das Backup eingetragen sind. Allerdings ist die Verwaltung eines Katalogs aufwendig (gerade in Bezug auf die Releases gibt es hier einige Fallstricke) und zudem nicht unbedingt erforderlich.

Zunächst einmal sollte man die Datenbank, sofern sie sich noch nicht im Archive-Modus befindet, entsprechend umstellen. Ab Version 10g bietet Oracle ein Standardverzeichnis für die Archivierung an. Wenn die Flash-Recovery-Area (Parameter `db_recovery_file_dest`) aktiviert ist, zeigt der Parameter `log_archive_dest_10` automatisch auf

dieses Verzeichnis, so dass archivierte Redo-Log-Dateien automatisch in den entsprechenden Unterverzeichnissen gespeichert werden.

Für die Sicherung der Datenbank mit RMAN sind jetzt einmalig einige Parameter zu setzen. Hierbei handelt es sich um Konfigurationsparameter für den RMAN, die in den Control-Files abgelegt sind.

Im Toad Database-Browser ruft man über die rechte Maustaste das Menü „Generate RMAN Script“ auf und wählt dann das Skript „Basic Script to setup RMAN for backups“ aus. Das jetzt im RMAN-Editor angezeigte Skript lässt sich ohne weitere Änderungen ausführen (siehe Abbildung 1). Sscript 1 zeigt die standardmäßig ausgelieferte Konfiguration:

```
CONFIGURE CONTROLFILE AUTOBACKUP ON;
CONFIGURE DEFAULT DEVICE TYPE TO DISK;
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO '/u02/orabackup/%d/%F.bck';
CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/u02/orabackup/%d/%d-%Y-%M-%D-%p-%s.bck'
```

Skript 1: RMAN-Konfigurationsskript

Damit erfolgt zunächst einmal ein automatisches Backup des Control-Files und des SP-Files gemeinsam mit dem Backup der Datenbank. Die weiteren Kommandos beziehen sich auf die

Lage und den Namen der RMAN-Dateien. Wichtig ist, dass der Name des Control-File Backups das „%F“ enthält, da es neben dem Datum auch die Datenbank-ID als Kennung beinhaltet. Dieses Konfigurationsskript kommt jetzt über „Execute Statement“ zum Ablauf. Eine einmalige Ausführung genügt, da anschließend alle Parameter in die Control-Files eingetragen sind. Die aktuelle Konfiguration ist über V\$RMAN_CONFIGURATION einsehbar. Damit sind alle Voraussetzungen für ein Backup der Datenbank, einzelner Tablespace oder Datendateien (auch inkrementell) erfüllt. Bei kleineren Datenbanken lässt sich jetzt täglich die Sicherung über das Skript 2 „Hot Backup of Database“ durchführen.

```
CROSSCHECK ARCHIVELOG ALL;
BACKUP DATABASE;
BACKUP ARCHIVELOG ALL NOT BACKED UP 2 TIMES;
DELETE ARCHIVELOG ALL COMPLETED AFTER ,SYSDATE - 3';
Skript 2: Backup-Skript
```

Mit CROSSCHECK wird überprüft, ob alle Archivelogs, die noch nicht gesichert wurden, auch tatsächlich existieren. Dieser Befehl ist sinnvoll, da es möglich ist, dass ein DBA aufgrund nicht ausreichenden Platzes im Archiv-Verzeichnis kurzfristig archivierte Redo-Log-Dateien löschen musste, ohne sie vorher zu sichern. Anschließend werden die Datenbank und alle archivierten Redo-Log-Dateien gesichert,

mit denen dieses nicht schon mehr als zweimal durchgeführt worden ist. Zum Schluss erfolgt das Löschen der archivierten Redo-Logs, die älter als drei Tage sind.

Hierbei handelt es sich um ein Beispiel, wie ein Backup aussehen kann. In den Toad-Optionen findet man unter „RMAN Templates“ die Lage der Skripte, um sie nach eigenen Vorgaben anzupassen und zu erweitern.

Komprimierung von Backups

Für kleinere Datenbanken ist diese Vorgehensweise sicherlich ausreichend. Bei größeren Systemen steht man jedoch vor dem Dilemma, dass das Backup zu groß wird und somit nicht einfach auf eine Disk passt. In vielen Fällen führt dies dazu, dass ein Backup direkt auf Tape erfolgt. Auch wenn das im ersten Moment opportun erscheint, ist zu bedenken, dass ein Tape-Device im Falle eines Restores bei weitem nicht die Performance erreicht, die ein Disk-Zugriff bietet. Speziell das Restore einzelner Blöcke, das mit RMAN möglich ist, kann unverhältnismäßig lange dauern, wenn man die Backup-Sets zunächst vom Tape zurückspielen muss.

Als Alternative bietet es sich an, beim bewährten Disk-Backup zu bleiben und eine Komprimierung zu verwenden. Die neue Quest LiteSpeed Engine for Oracle lässt sich sehr einfach in RMAN integrieren und erlaubt eine hohe Komprimierung und Verschlüsselung der Backups, ohne dass die Laufzeiten ansteigen. Im Gegenteil: In vielen Fällen wird sich die Gesamtlaufzeit des Backups sogar reduzieren lassen, da durch die starke Komprimierung von 70 bis 90 Prozent wesentlich weniger Schreibvorgänge notwendig sind.

Die LiteSpeed Engine for Oracle verhält sich gegenüber RMAN wie eine Tape-Library. Das bedeutet, dass hier die offiziell unterstützte Oracle-Methode für die Integration von Media Management Libraries mittels der libobk.so verwendet wird. Die Installation ist denkbar einfach. Das Installationskript wird aufgerufen und installiert die Software ins Verzeichnis \$ORACLE_HOME/Quest/LEO. Zusätzlich wird im ORACLE_HOME/

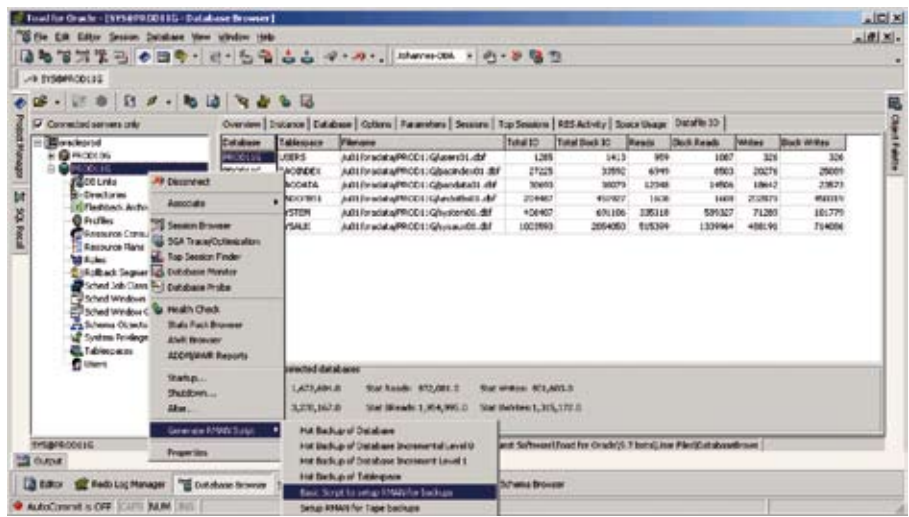


Abbildung 1: Aufruf von RMAN aus dem Database-Browser

bin-Verzeichnis ein Executable mit Namen leo für die Konfiguration und das Reporting der Backups angelegt. Außerdem wird, wie schon erwähnt, die Library libobk.so auf die entsprechende Library (libleoxxx.so) gelinkt. Der RMAN erhält die Information, dass nicht mehr auf Disk, sondern auf Tape gesichert werden soll. Im Gegensatz zu einem „normalen“ Tape wird jetzt jedoch definiert, wie die Dateien heißen sollen und wo sie liegen.

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE
,SBT_TAPE' TO
        ,/u02/orabackup/%d/%F.
tbck';
RMAN> CONFIGURE CHANNEL DEVICE
TYPE ,SBT_TAPE' FORMAT
        ,/u02/orabackup/%d/%d-%Y-
%M-%D-%p-%s.tbck';
RMAN> CONFIGURE DEFAULT DEVICE
TYPE TO ,SBT_TAPE';
```

Skript 3: Backup-Skript für Tape-Backup

Das gezeigte Beispiel verwendet wiederum das Verzeichnis „/u02/orabackup“ für die Sicherung, nur die erstellten Dateien enden auf „tbck“ und nicht auf „bck“ wie bei einem Disk Backup. Der letzte Befehl „CONFIGURE DEFAULT DEVICE TYPE TO ,SBT_TAPE“ gibt an, dass fortan alle Backups auf das Tape-Device erfolgen sollen. Alle zukünftigen Backups, die mit dem Default-Device-Type arbeiten, werden jetzt komprimiert und gegebenenfalls verschlüsselt. Die Konfiguration der Verschlüsselungstechnik beziehungsweise unterschiedliche Komprimierungslevel lassen sich über die Kommandozeile leo anpassen.

```
% leo configure
Please enter the following values. Type „?“ for help.
Target Type [Disk]:
Compression Level (0-4) [1]:
Encryption Type ES256:
Encryption Password :
Confirm Password :
```

Skript 4: Backup-Skript für Tape-Backup

Neben Verschlüsselung und Komprimierung besteht auch die Möglichkeit, die von leo komprimierten und/oder verschlüsselten Daten an ein Tape-

```
Starting backup at 06-FEB-09
current log archived
using target database control file instead of recovery catalog
allocated channel: ORA_SBT_TAPE_1
channel ORA_SBT_TAPE_1: SID=80 device type=SBT_TAPE
channel ORA_SBT_TAPE_1: LiteSpeed Engine for Oracle
channel ORA_SBT_TAPE_1: starting archived log backup set
channel ORA_SBT_TAPE_1: specifying archived log(s) in backup set
input archived log thread=1 sequence=17 RECID=406 STAMP=676969239
...
channel ORA_SBT_TAPE_1: finished piece 1 at 06-FEB-09
piece handle=/u02/orabackup/PROD11G/PROD11G-2009-02-06-1-144.tbck
tag=TAG20090206T165309 comment=API Version 2.0,MMS Version 1.1.0.168
channel ORA_SBT_TAPE_1: backup set complete, elapsed time: 00:01:57
Finished backup at 06-FEB-09
```

Skript 5: RMAN-Ausgabe

```
Leo> show history
Start Time      Raw      Comp      Ratio  Filename
-----
5-Feb-09 00:03  9728K    100K    99.0% /u02/orabackup/PROD11G/c-416
5139185-20090205-00.tbck
5-Feb-09 00:03   643M    202M    68.5% /u02/orabackup/PROD11G/
PROD11G-2009-02-05-1-137.tbck
5-Feb-09 00:04  9728K    100K    99.0% /u02/orabackup/PROD11G/c-416
5139185-20090205-01.tbck
6-Feb-09 00:00   987M    268M    72.8% /u02/orabackup/PROD11G/
PROD11G-2009-02-06-1-139.tbck
6-Feb-09 00:02  9728K    101K    99.0% /u02/orabackup/PROD11G/c-416
5139185-20090206-00.tbck
6-Feb-09 00:02   309M    108M    65.1% /u02/orabackup/PROD11G/
PROD11G-2009-02-06-1-141.tbck
6-Feb-09 00:02  9728K    101K    99.0% /u02/orabackup/PROD11G/c-416
5139185-20090206-01.tbck
6-Feb-09 15:49   514M    176M    65.7% /u02/orabackup/PROD11G/
PROD11G-2009-02-06-1-143.tbck
6-Feb-09 16:53   985M    339M    65.5% /u02/orabackup/PROD11G/
PROD11G-2009-02-06-1-144.tbck
6-Feb-09 16:55   987M    268M    72.8% /u02/orabackup/PROD11G/
PROD11G-2009-02-06-1-145.tbck
6-Feb-09 16:56   512K    118K    76.8% /u02/orabackup/PROD11G/
PROD11G-2009-02-06-1-146.tbck
6-Feb-09 16:56  9728K    102K    99.0% /u02/orabackup/PROD11G/c-416
5139185-20090206-02.tbck
```

Skript 6: LEO-Historie

Device (derzeit TSM) weiterzuleiten. In diesem Fall wird jedoch nur noch ein Betriebssystem-Adapter für die Tape Library verwendet. Der Oracle-Adapter ist nicht mehr erforderlich. Wenn man nach dieser Konfiguration das Backup-Skript (wie in Skript 2 gezeigt) ausführt, erfolgt das Backup der Datenbank, der Control-Files und der archivierten Redo-Log-Dateien mit AES256 verschlüsselt und komprimiert. Die Ausgabe von RMAN sieht dann wie in Skript 5 aus. Über die Befehlszeile leo

kann man sich anschließend die Details ansehen (siehe Skript 6).

Fazit

Auch wenn es möglich ist, direkt auf Band zu sichern, sollte man sich immer überlegen, welche Konsequenzen dies für ein – offensichtlich nie – notwendiges Restore hat.

Kontakt:

Johannes Ahrends
johannes.ahrends@quest.com

Variable Datenselektion für OLTP- und OLAP-Systeme

Volker Ricke, POINT. Consulting GmbH

Wäre es nicht gut, eine wiederverwendbare Selektions-Komponente zu haben, die für beliebige Datenquellen eingesetzt werden kann? Eine Antwort auf diese Fragen ist die im Rahmen eines Kundenprojekts entstandene Variable Selektion.

Wie oft haben Sie schon Abfragedialoge für Ihren Kundenstamm geschrieben? Hier eine Suchmaske für das Stammdaten-Modul, dort eine Parametermaske für die Ausgabe von Reports, dann noch ein Filter für Data-Warehouse-Auswertungen? Wie oft haben Sie an mehreren Stellen Programmänderungen vornehmen müssen, nachdem ein wichtiges neues Kriterium gefordert wurde?

Für die Entwicklung neuer Planungs- und Auswertungssysteme hatte ein Unternehmen großen Bedarf an fein granulierten Abfragemöglichkeiten über Stamm- und Bewegungsdaten. Abfragen sollten vom Anwender gespeichert und wieder verändert werden können, ihre Ergebnisse sowohl in Online-, als auch in Batchprogrammen zur Verfügung stehen. Zusätzlich geforderte Abfragemöglichkeiten und Strukturänderungen an den abgefragten Tabellen waren möglichst ohne Programmieraufwand zu implementieren.

Zielplattform war zunächst die für die OLTP-Systeme genutzte Oracle Datenbank, als Clients waren .NET Windows Forms-Applikationen vorgesehen, die auch als Parameter-Formen für Crystal-Reports dienen sollten. Stand zunächst nur die Selektion von Kundendaten im Fokus, wurde die Variable Selektion in der Folgezeit auch zur Definition von Berechtigungsgruppen im HR-Bereich sowie mit Konsumenten-Daten im Kampagnen-Management eingesetzt. Auch in die Data-Warehouse-Welt schaffte die Variable Selektion den Sprung, hier wurde sie als vorgeschalteter Datenfilter für multidimensionale Analysen integriert.

Die Idee hinter der Variablen Selektion

Die Variable Selektion wird über Metadaten konfiguriert, die in einem Datenbank Repository gespeichert sind. In ihnen ist definiert, was grundsätzlich selektiert werden kann (Selektionsquelle) und wie es sich dem Anwender gegenüber darstellen soll (Selektions-sicht). Dank der Trennung von Quelle und Sicht muss eine Datenbasis nur einmal in der Quelle eingepflegt werden, während die je nach Einsatzzweck wechselnden Anforderungen an Auswahl und Anordnung der abfragbaren Felder in den Sichten berücksichtigt sind.

Der Anwender speichert seine Abfragen als Selektionsregeln ebenfalls in der Datenbank, die Regeln lassen sich somit sowohl von Dialoganwendungen, als auch von Batchprozessen verwenden. Aus den eingestellten Regeln und den Metadaten werden zur

Laufzeit SQL-Abfragen generiert und als Ergebnismenge ausgegeben. Zum Erstellen der Selektionsregeln und für die Pflege der Metadaten existieren Dialoganwendungen, das Generieren der dynamischen SQL Statements und Bereitstellen der Ergebnismengen übernimmt ein PL/SQL Datenbank-Package.

Selektionsquellen, Attribute und Lookups

Herzstück der Variablen Selektion ist die Selektionsquelle. Im einfachsten Fall basiert sie auf einer Tabelle mit einer Schlüsselspalte, die für die Selektion als Rückgabewert definiert ist. Auch Views, Materialized Views oder frei formulierte SQL-Abfragen können eine Quelle bilden.

Abfragefähige Spalten oder Single Row Functions sind als Selektionsattribute definiert, in denen neben dem

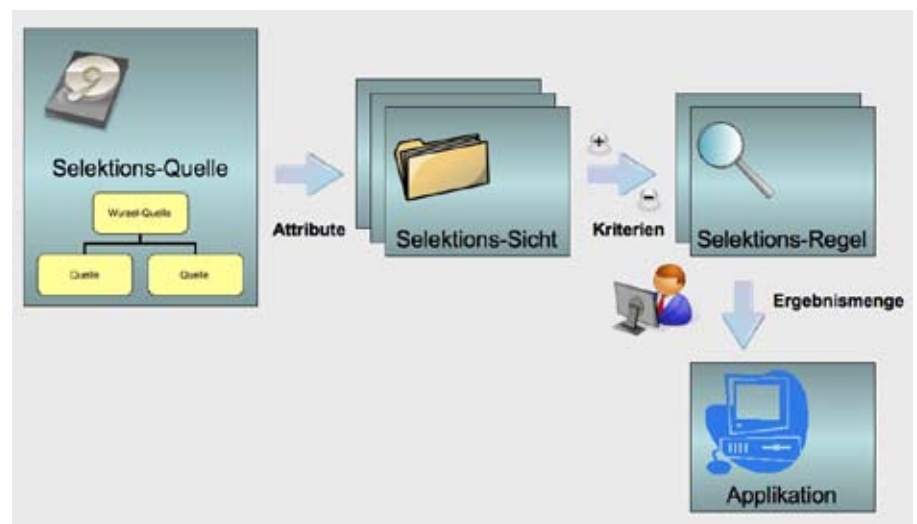


Abbildung: Informationsstruktur der Variablen Selektion

Datentyp auch die Darstellung im Abfragedialog (Eingabefeld, Checkbox, Liste etc.) oder eventuelle Formatmasken hinterlegt sind. Zur Anzeige von Auswahllisten oder zur Validierung von Suchkriterien lassen sich Lookups definieren und mit Attributen verknüpfen. Aus der bisher beschriebenen Wurzel-Quelle wird das treibende SQL für die Abfrage generiert. Es können mit ihr per JOIN-Kriterium noch weitere Quellen verknüpft sein, deren Attribute in eine spätere Abfrage per Subselect eingebunden sind.

Selektionssichten

Ist in der Selektionsquelle erst einmal die Datenbasis definiert (zum Beispiel der OLTP-Kundenstamm mit seinen Detail-Tabellen und einigen Bewegungsdaten), wird in den Selektionssichten zusammengestellt, welche Attribute den Anwendern in den unterschiedlichen Anwendungen zur Suche angeboten und in welcher Struktur sie dargestellt werden sollen. Bindet eine Applikation die Variable Selektion ein, verweist sie immer auf eine solche Selektionssicht. Die selektierbaren Attribute können in der Sicht nach fachlichen Aspekten und unabhängig von der ursprünglichen Datenbankstruktur in einer Baumstruktur organisiert und bezeichnet werden.

Die angebotenen Abfragekriterien lassen sich so einfach dem jeweiligen Anwendungszweck anpassen – der Anwender erhält nur die gerade sinnvollen Kriterien angeboten, bei abweichendem Fachvokabular können die Attribute unterschiedlich benannt werden. Mit der Option einer Standard-WHERE-Klausel lässt sich die Ergebnismenge der Sicht weiter einschränken, per Verknüpfung mit einer Vorschau-Definition sind Felder für einen Export freischaltbar.

Die Selektionsregel enthält die Informationen, welche Attribute mit welchen Bedingungen (zum Beispiel „=“, „LIKE“, „>=“, „<=“, „BETWEEN“, „IN“ oder „IS (NOT) NULL“) und Werten der Anwender aus einer Selektionssicht abfragt. Die einzelnen Kriterien können „UND“- , „UND NICHT“- oder „ODER“-verknüpft sein. Auch bereits

zur gleichen Sicht gespeicherte Regeln lassen sich hier einbinden und somit wiederverwenden.

Variable Selektion aus Anwendersicht

Der Anwender kann den Selektionsdialog einerseits direkt starten und auswählen, über welche der verfügbaren Sichten er eine Selektion öffnen oder erstellen möchte.

Andererseits kann er per „Variable Selektion“-Schaltfläche aus einer Partner-Applikation heraus, aber auch direkt in die passende Sicht wechseln und eine Regel erstellen, deren Ergebnis er anschließend in seiner Applikation zum Filtern der Daten verwenden kann. Zur Plausibilitätskontrolle kann im Selektionsdialog die Anzahl der Treffer angezeigt werden. Außerdem steht eine in Excel exportierbare Vorschau der Daten zur Verfügung, über die so ein eigenes Ad-hoc-Reporting möglich ist.

Eine Selektionsregel ist standardmäßig privat, ihr Eigentümer kann sie aber anderen Benutzern freischalten. In den meisten Projekten wurden die ein bis zwei gängigsten Abfragefelder innerhalb der Applikation implementiert, während die Variable Selektion als erweiterter Filter angeboten wird.

Variable Selektion aus Administratorsicht

Der größte Administrationsaufwand besteht im Einrichten der Datenquellen. Hier spielt natürlich eine Rolle, wie aktuell die abgefragten Daten sein sollen. Greifen die Datenquellen in der OLTP-Welt üblicherweise direkt auf die relationalen Tabellen der operativen Systeme zu, wird in der OLAP-Welt meist auf leichter auswertbare Materialized Views zurückgegriffen. Da die Datenquellen und ihre Verknüpfungen per SQL-Syntax definiert sind, können sie aber auch nachträglich umkonfiguriert und optimiert werden. Das Administrationsmodul bietet Unterstützung beim Import von Tabellen- und Spaltendefinitionen sowie der Spaltenkommentare in das Repository der Variablen Selektion an. JOIN-Krite-

rien lassen sich aus bestehenden Foreign Key Constraints heraus ebenfalls generieren. Im Konfigurationsdialog für die Datenquellen können auch die Lookups definiert und Attributen zugeordnet werden.

Setzt die Definition der Selektionsquellen noch gute SQL-Kenntnisse und Wissen über die Datenbank-Strukturen voraus, können Power-User oder der Application-Support die Selektionssichten auch pflegen. Für die Sichten steht daher ein eigener Konfigurationsdialog zur Verfügung. Hier werden die aus einer Datenquelle benötigten Attribute zur Verfügung gestellt, in Ordnern gruppiert und – falls gewünscht – umbenannt.

Variable Selektion aus Entwicklersicht

Dem Entwickler stehen PL/SQL- und .NET-Schnittstellen zur Verfügung. Über die .NET-Schnittstellen bindet er die Variable Selektion in seinen Client ein, per PL/SQL können Datenbankprozeduren auf Selektionsergebnisse zugreifen. Existiert bereits die für ein neues Projekt passende Quelle, muss der Entwickler sie nur noch um möglicherweise zusätzlich benötigte Attribute erweitern und initial eine neue Sicht definieren. Spätere Erweiterungen werden durch den Administrator eingepflegt, eine Programmänderung ist nicht erforderlich.

Fazit

Zwischen 2004 und 2008 ist die Variable Selektion mit den Projekten gewachsen und wird in vielen seiner Systeme als zentrale Suchkomponente eingesetzt. Der zunächst hohe Aufwand zur Entwicklung einer generischen, deklarativen Lösung hat sich gelohnt. Als „Abfallprodukt“ ist mit der konfigurierbaren Vorschau der Variablen Selektion ein intensiv genutztes Ad-hoc-Reporting-Werkzeug entstanden.

Kontakt:

Volker Ricke

Volker.Ricke@point-gmbh.com

Die DOAG beim Beta-Test von 11gR2 der User Groups in Redwood Shores

Michael Pfautz, Leiter der Special Interest Group Database, führte Tagebuch als Vertreter der DOAG beim Beta-Test der neuen Datenbank-Version in den Oracle-Headquarters.

Samstag, 18. Juli 2009

Anreise zum Flughafen Frankfurt, Abflug um 9:10 Uhr mit Lufthansa Economy Class direkt nach San Francisco. Der Flieger umkreist beim Landeanflug das Oracle Gelände, ich bekomme einen wunderbaren Blick auf das gesamte Areal. Transfer mit dem Shuttle-Service nach Redwood City.

Das Hotel ist sehr komfortabel und angenehm, das Wasser im Pool etwas kalt und es weht ein stets frischer Wind. Nach der langen Reise bin ich sehr müde und falle regelrecht ins Bett.

Sonntag, 19. Juli 2009

An das amerikanische Frühstück muss ich mich erst gewöhnen, obwohl auch hier schon ein gesundes Ernährungsbewusstsein Einzug gehalten hat. Danach gehe ich zu Fuß auf Erkundungstour auf das Gelände der Oracle Headquarters. Ich unterschätze die kalifornische Sonne, die an diesem Tage vom strahlend blauen Himmel scheint, so dass ich am Abend einen kräftigen Sonnenbrand im Gesicht habe. Ich kühle mich bei ein paar Runden im Pool und teste anschließend auf dem Zimmer nochmals meinen Laptop, um für den nächsten Tag gut gerüstet zu sein.

Montag, 20. Juli 2009

Ich steuere zu Fuß das Schulungsgebäude 301 auf dem Oracle Campus an und werde von den Oracle-Mitarbeiterinnen Debbie Migliore und Lynn Snyder begrüßt. Ein Teil der anderen Usergroup-Tester ist schon anwesend, wir lernen uns beim gemeinsamen Frühstück kennen.

Nach und nach treffen weitere Oracle Mitarbeiter ein, die uns später gut unterstützen. Roy Sworner von Oracle begrüßt mich herzlich, er ist durch Mike Dietrich von Oracle Deutschland von meiner Teilnahme informiert worden.

Roy hält gleich den ersten Vortrag zum Thema „Migration“. Anschließend bauen wir unsere Testumgebungen für die erste Migration auf. Am Nachmittag folgen weitere Präsentationen über die neuen Features in Release 2.

Dienstag, 21. Juli 2009

Ich starte meinen Testcase für den ETL-Prozess ins Data Warehouse, der sich schwerpunktmäßig mit dem Thema „Partitionierung“ beschäftigt. Hier gilt es, einen früheren Fehler zu überprüfen. Er tritt in der neuen Version nicht mehr auf.

Mittwoch, 22. Juli 2009

Ablauf des zweiten Testcases zum Thema „Flashback Data Archive“. Ich erkenne die Verbesserungen, die das neue Release bietet und führe im Anschluss sehr konstruktive Gespräche mit dem Produktmanagement über dieses Thema. Lassen Sie sich überraschen, was es zukünftig zu diesem Thema alles geben wird.

Donnerstag, 23. Juli 2009

Ich starte meinen dritten Testcase mit der Funktionalität „Advanced Compression“. Auch hier hat sich gegenüber dem ersten Release einiges getan. Die erzielten Ergebnisse verblüffen nicht nur mich, sondern auch meine Kollegen von der AUOUG (Australien und Neuseeland), EOUG (England) und IOUG (USA).

Am Nachmittag findet eine Video-Session statt. Parallel dazu erstellen wir die Präsentationen für die Feedback-Session am nächsten Tag. Abends besuchen wir in San Francisco den berühmten Pier 39.

Freitag, 24. Juli 2009

Zur Feedback-Session ist der Raum im Conference Center regelrecht über-

füllt. Die meisten von uns stehen, um die Slides mit Ergebnissen, Verbesserungswünschen und Anregungen sehen zu können. Auch Oracle Vice President und Chefentwickler Andrew Mendelsohn ist mit vielen seiner Mitarbeiterinnen und Mitarbeitern vor Ort. Nicht nur für die teilnehmenden Tester, auch für Oracle ist der gemeinsame Beta-Test der User Groups ein spannendes Ereignis.

Unser Fazit lautet: 11gR2 ist wirklich gut und es lohnt sich, schnellstmöglich upzugraden, zumal der Upgrade-Prozess selbst noch besser durch die Software unterstützt wird.

Nach dem Transfer zum Flughafen startet mein Flug mit über 1,5 Stunden Verspätung, weil die Gangway nicht vom Flugzeug entfernt werden konnte, um 20:25 Uhr in Richtung Frankfurt.

Samstag, 25. Juli 2009

Wieder zu Hause bin ich zwar müde, aber dennoch glücklich, an dieser Veranstaltung teilgenommen zu haben. Alles in allem habe ich eine sehr gelungene Veranstaltung von Oracle erlebt, mit viel neuen Erfahrungen und Eindrücken. Besonders positiv überrascht war ich von der optimistischen Einstellung und der sehr kooperativen Zusammenarbeit mit den Oracle-Mitarbeitern sowie den schnell erzielten Ergebnissen. Details zu 11gR2, die wir hier noch nicht veröffentlichen dürfen, gibt es dann in der nächsten Ausgabe der DOAG News sowie in zahlreichen Vorträgen im Rahmen der DOAG 2009 Konferenz von 17. – 19. November 2009 in Nürnberg. Ein Besuch, der sich bestimmt lohnt.

Kontakt:

Michael Pfautz
sig-database@doag.org

Die neue Fachzeitschrift für Datenprofis!

■ Für Entwickler ■ Administratoren ■ Softwarearchitekten

Bestellen Sie
sich Ihr
kostenloses
Kennenlern-
Exemplar unter:

www.databasepro.de/probelesen



Entfernte Dateien mit SSHFS zentral und sicher verwalten

Christian Ballweg, OPITZ CONSULTING Essen GmbH

Mit SSHFS und einem Userspace-Modul namens FUSE ist eine transparente Einbindung entfernter Verzeichnisse mit der impliziten Nutzung von SFTP möglich, eine FTP-Entsprechung des SSH-Protokolls. Auf diese Weise sind die Dateien verschiedener Server auch über das Internet zentral und sicher zu erreichen und direkt modifizierbar.

Es gehört zum täglichen Leben eines Administrators, bei Problemen seiner Server in die Logdateien zu schauen – außerdem wird er von Zeit zu Zeit Änderungen an Konfigurationsdateien vornehmen müssen. Auch die Erstellung von Skripten, die Bereitstellung von Dateien auf den Servern oder der Datei-Download gehören zu den Routineaufgaben eines Administrators. Oft kommt es dabei vor, dass nicht nur lokale Dateien durchsucht oder modifiziert werden müssen, sondern auch entfernte. Dabei liegen diese womöglich nicht nur im eigenen LAN auf verschiedenen Servern, sondern zusätzlich auf solchen, die über WAN oder sogar ohne besonderen Schutz über das Internet kontaktiert werden müssen. Hier nimmt das Thema „Verschlüsselung“ einen hohen Stellenwert ein.

Einerseits hat der Administrator Möglichkeiten, die ihm althergebrachte Mittel wie FTP oder NFS bieten und die im eigenen Netz sicher noch legitim erscheinen. Andererseits existiert im Web-Umfeld häufig ein WebDAV-Zugang, der einen Dateitransfer über den Webserver ermöglicht.

Doch je exponierter sich eine Verbindung darstellt, desto mehr Wert wird auf ihre Sicherheit gelegt. Schnell kommt man zu SSH-Alternativen wie SCP auf der Kommandozeile oder dem grafischen WinSCP beziehungsweise Integrationen des „File transferred over Shell“-Protokolls (FISH), das vor allem in jüngeren Linux-Derivaten zum Einsatz kommt. Dieses Protokoll nutzt das sicher verschlüsselte ssh (auf Wunsch auch das unverschlüsselte RSH) und bietet eine gewisse Transparenz bei der Verwendung des Datei-Browsers. Keines der bisher genannten Verfahren ermöglicht aber eine einfache, schnel-

le und gleichzeitig sichere Integration entfernter Verzeichnisse in eine bestehende Umgebung, die dem NFS vergleichbar wäre. Samba und OpenAFS wären zwar Alternativen, benötigen jedoch zur Konfiguration Administrator-Rechte auf dem Quellserver.

Eine Lösung für dieses Dilemma findet man in SSHFS, der mittels FUSE (Filesystem in USErspace) flexiblen Einbindung verschiedenster Dateisysteme mit Userrechten. Mögliche Dateisysteme sind beispielsweise das VMWare-Filesystem (<http://fuse.sourceforge.net/wiki/index.php/FileSystems>). Auf dieser Seite findet sich auch eine Liste weiterer Dateisysteme. SVNFs zum Einbinden von SVN-Repositories (Codeverwaltung mit Subversion) findet man unter <http://www.jmadden.eu/index.php/svnfs/>.

Zunächst sind die benötigten Pakete mit root-Rechten zu installieren:

```
# apt-get install fuse-utils
sshfs # Debian und Ubuntu bzw.
# yum install fuse-sshfs
# RedHat Sever, Fedora...
bzw.
# yast --install sshfs fuse
# SuSE
# modprobe fuse
# ggf. Modul einbinden
# usermod -a -G fuse meinuser
# User in Gruppe fuse
```

Danach folgt das manuelle Einbinden mit normalen Userrechten (User in Gruppe fuse). Jetzt ist „meinuser“ in der Lage, mit SSHFS zu arbeiten. Er kann beispielsweise den Verzeichnisbaum /var/log von den entfernten Servern A und B auf dem lokalen Server einbinden und durchsuchen. Die Passwörter werden interaktiv abgefragt:

```
$ mkdir -p ~meinuser/logA ~meinuser/logB
$ sshfs root@ServerA:/var/log ~meinuser/logA
$ sshfs root@ServerB:/var/log ~meinuser/logB
$ grep „Suchstring“ ~meinuser/log?/*
```

Damit er wirklich nahtlos in die Lage versetzt wird, fremde Dateisysteme zu mounten, kann „meinuser“ noch einen Schlüsselaustausch mit dem entfernten Server vornehmen und sich so eine Passworteingabe ersparen. Die Generierung der öffentlichen und privaten Schlüssel sowie der Austausch des öffentlichen Schlüssels der Quelle mit dem Zielsystem (vorgenommen von der Quelle) geht wie folgt vonstatten:

```
$ cd $HOME
$ ssh-keygen -t rsa
Enter file in which to save the key (.ssh/id_rsa): <ENTER>
Enter passphrase (empty for no passphrase): <ENTER>
Enter same passphrase again: <ENTER>
Your identification has been saved in id_rsa.
Your public key has been saved in id_rsa.pub.
The key fingerprint is:
6a:h3:7f:ab:5f:1c:87:3b:c6:a2:05:01:20:05:h1:3f meinuser@pc
```

Dann folgt das Kopieren des öffentlichen Schlüssels auf den Zielsystem (hier im Beispiel gleicher Username):

```
$ scp -p .ssh/id_rsa.pub ServerA:~/.ssh/authorized_keys
$ ssh ServerA chmod 600 ~/.ssh/authorized_keys
```


Der letzte Schritt stellt eine Demonstration der passwortlosen Funktionsweise dar und setzt die Dateirechte auf dem Zielsever wie gefordert. Nun können wir ein Verzeichnis wie gehabt, jedoch ohne Passworteingabe, einbinden:

```
$ sshfs ServerA:/var/log ~mei-
nuser/logA
```

Alternativ kann auch ein permanenter, für jeden zugänglicher Mountpoint (beispielsweise read only: „ro“) in der /etc/fstab des lokalen Rechners eingetragen werden, wie in folgender Zeile beschrieben:

```
sshfs#user@remotemachine:/path/
remote /path/local fuse [opti-
ons,...] 0 0):
sshfs#meinuser@ServerA:/var/
log /home/meinuser/logA fuse
ro,IdentityFile=/home/meinu-
ser/.ssh/id_rsa,uid=1001,gid=10
0,umask=022,allow_other 0 0
```

Das manuelle Aushängen funktioniert übrigens mit dem gleichen User einfach mittels

```
fusermount -u /home/meinuser/
logA.
```

Szenario 1

Ein Administrator hat mehrere Server zu verwalten, von denen sich ein Teil im lokalen, ein anderer in einem entfernten Rechenzentrum befindet (WAN). Weitere Server in Außenstellen sind nur über das Internet erreichbar. Der Administrator mountet die Quell-Verzeichnisse, jeweils wie gewünscht „read only“ (ro) oder mit Schreibberechtigung (rw), in eine neue logische Struktur, beispielsweise in ein Hauptverzeichnis, das per Samba bestimmten Domänenbenutzern freigegeben wurde und verschiedene Server-Mountpoints als Unterverzeichnisse hat.

Szenario 2

Unter Verwendung der gemounteten Verzeichnisse aus Szenario 1 können nun neue Verzeichnisse mit Dateien bestückt werden. Der Administrator kann zusätzlich bestimmte Dateien mittels „symbolic links“ als Verweise in einem neuen Verzeichnis bereitstellen.

So können die Oracle alert.log-Files aller Server des Unternehmens („ro“ per Mountpoint-Definition) in einem neuen Verzeichnis zusammengeführt werden, während eine weitere Auflistung die init.ora-Files der Datenbanken aller Server (dieses mal „rw“) enthält.

Szenario 3

Die schon recht flexible Struktur aus Szenario 2 kann der Administrator nun noch erweitern, indem er einzelnen Usern der lokalen Windows-Domäne Teile dieser neuen Verzeichnisstruktur wie die schreibfähigen Share-Verzeichnisse bewusst (etwa per Samba) mit eingeschränkten Rechten freigibt. In solchen können diese User dann beispielsweise ihre CSV-Dateien für den nächtlichen Batch-Import in die Datenbank ablegen. Damit entfällt das lästige Ausprogrammieren von Schnittstellen, die letztlich doch manuell gefüttert werden müssen.

Szenario 4

Es können auch einzelne Verzeichnisse zu Backup-Zwecken gemountet werden. Wenn man dazu „rsync“ einsetzt, können ein einmaliger Abzug (eine Kopie aller Files in ein Backup-Verzeichnis) und darauf folgende rsync-Vorgänge (schnell, da nur geänderte Dateien ko-

piert werden) ein adäquates Backupmittel für statische Dateien darstellen.

Fazit

Durch die Flexibilität von SSHFS sind noch weit mehr als die genannten Anwendungsfälle denkbar. Je nach Anwendungsfall wird man noch zusätzliche Optionen oder weitere Techniken verwenden wollen. Als Hilfestellung für den ersten Zugang sind neben den Manual-Seiten die genannten Links und die Suche nach den nachfolgenden Stichwörtern sehr nützlich:

- ssh: ssh-copy-id, ssh-add (ssh-keygen-Schlüssel mit Passwort: <http://www.linux-community.de/Internal/Artikel/Print-Artikel/LinuxUser/2007/03/Daten-sicher-teilen>), ...
- weitere interessante Optionen von SSHFS: reconnect, cache, ServerAliveInterval, PermitRootLogin with out-password
- FUSE automounter: afuse
- ZFS on FUSE: Zugriff auf das moderne 128-bit-Dateisystem ZFS von Sun
- mysqlfs: Ermöglicht es, Daten in einer MySQL-Datenbank abzulegen

Kontakt:

Christian Ballweg
christian.ballweg@opitz-consulting.com

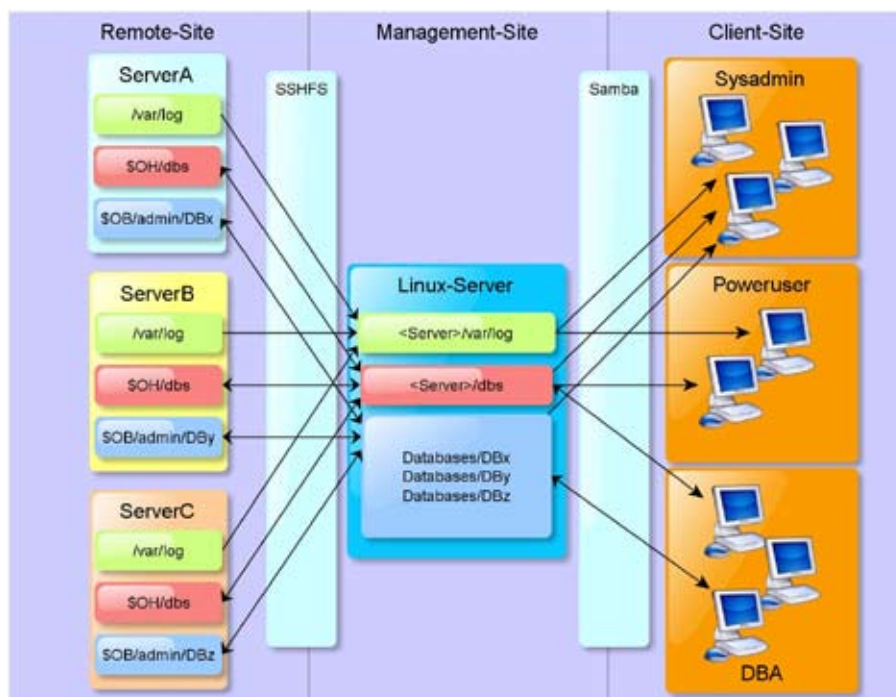


Abbildung: Beispiel für die Anwendung

Kostensenkungen und Effizienzsteigerungen im Oracle Datenbank-Umfeld mittels Capacity Management unter ITIL und ISO20000

Christian Wischki und Peter Stalder, Trivadis AG

Der einfache Ansatz, für jede neue Applikation immer den neuesten und größten Rechner mit möglichst vielen CPUs und entsprechend viel RAM zu verwenden, ist im Oracle Datenbank-Umfeld aus Gründen der Kosteneffizienz nicht möglich, da diese Vorgehensweise stets einen entsprechend hohen Hardware-Aufwand und vor allem auch sehr hohe laufende Lizenz-, Betreuungs- und Wartungskosten zur Folge hätte. Die Lösung für eine nachhaltige Kosten- und Effizienzoptimierung stellt das unter ITIL und ISO/IEC20000 beschriebene Capacity Management dar – vor allem das proaktive Kapazitäts- und Ressourcen-Management.

Bei genaueren Analysen im Oracle Datenbank-Umfeld stellt man leider sehr oft fest, dass selten mehr als 30 Prozent der vorhandenen Datenbank-Kapazitäten und Ressourcen genutzt werden beziehungsweise für das Business überhaupt notwendig sind. Abbildung 1 zeigt ein gravierendes Beispiel: Der komplette rechte untere Quadrant kann konsolidiert und somit die Effizienz der gesamten Datenbank-Landschaft optimiert werden – das spart Lizenz-, Betreuungs- und Wartungskosten. Ein Viereck entspricht einem Server, dessen Fläche die Summe des CPU-Verbrauchs der darauf befindlichen Datenbanken darstellt. Die CPU-Last wird durch die Farbe veranschaulicht – wenig (schwarz), mittel (dunkelrot) und voll (hellrot). Die Fläche der inneren Vierecke stellt den jeweiligen CPU-Verbrauch der einzelnen Datenbanken auf dem Server dar.

Ein proaktives Kapazitätsmanagement – das auch seitens ITIL und ISO/IEC20000 gefordert wird –, und ein darauf basierendes, permanentes oder zumindest periodisches und stets auf das Business hin ausgerichtetes Forecasting von Kapazitäten und Ressourcen versetzen ein Unternehmen in die Lage, brachliegende Potenziale zu reaktivieren oder bestehende laufende Kosten durch entsprechende Konsolidierungsmaßnahmen zu senken und somit die Effizienz von bestehenden

Datenbanklandschaften zu steigern (siehe Abbildungen 2 und 3).

Mittels statistischer Prognosen und Queuing-Analysen lassen sich im Oracle Datenbank-Umfeld die nicht-linearen Variablen „Business-„ und „Service-Anforderungen“ sowie „Res-

ourcen“ in Beziehung setzen. Sind zwei dieser Variablen bekannt, so lässt sich die dritte ermitteln. Wenn beispielsweise eine Bank 8.000 Zahlungsauftragsverarbeitungen pro Stunde fordert, wobei 80 Prozent aller einzelnen Zahlungsaufträge binnen zwei Se-

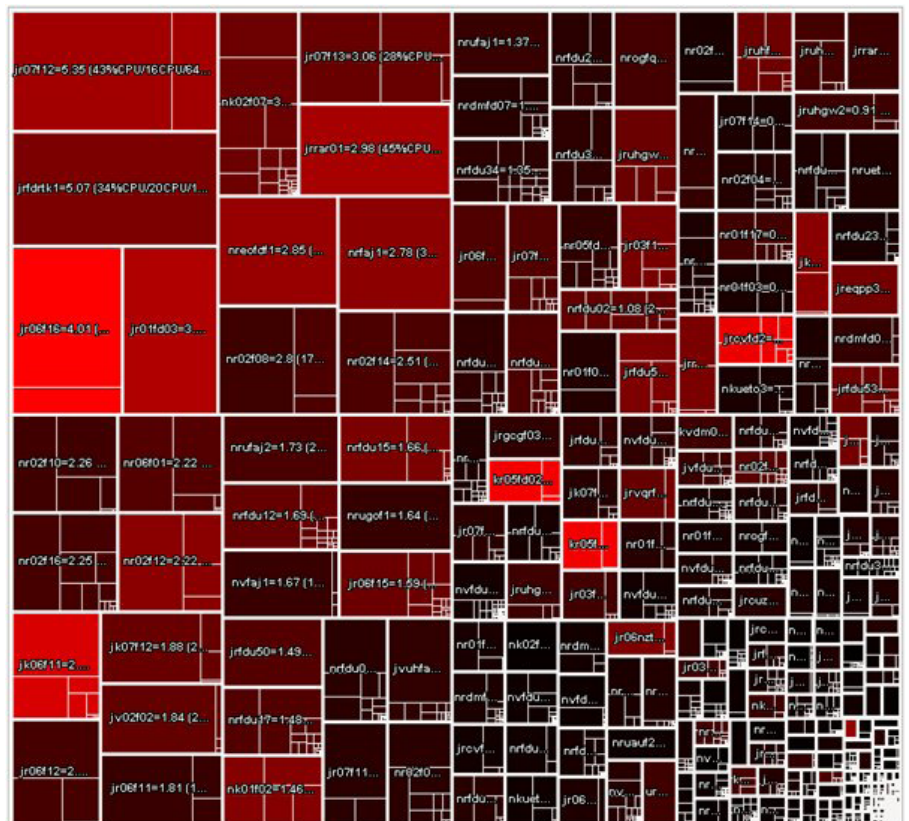


Abbildung 1: Wenn die Ressourcen- und Kapazitätsauslastung einer Datenbank-Landschaft wie abgebildet aussieht, lassen sich signifikant Kosten sparen

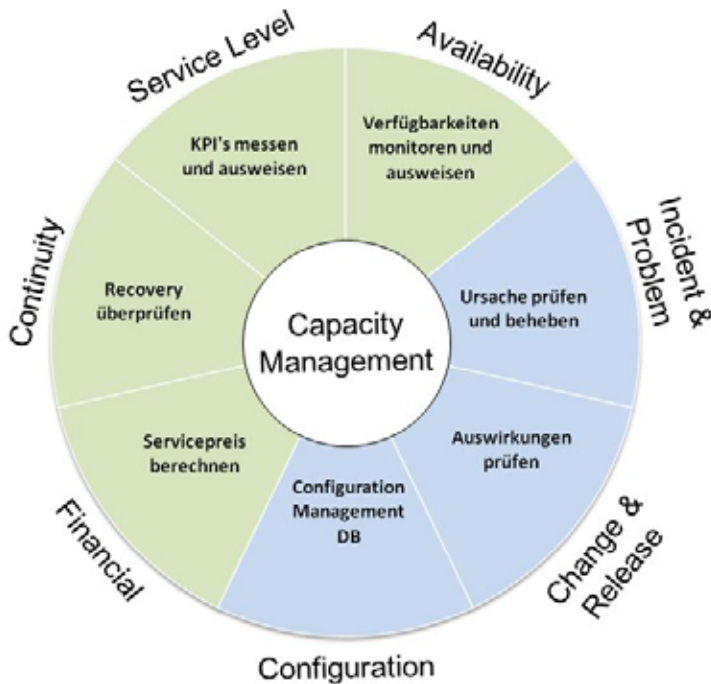


Abbildung 2: Capacity Management unter ITIL ist die Drehscheibe zu allen anderen Prozessen

kunden verarbeitet sein müssen, kann man darauf basierend die Kapazitäts- und Ressourcen-Anforderungen an die Datenbanken errechnen und diese somit auch entsprechend dimensionieren – sowohl auf die aktuellen als auch auf die zukünftigen Geschäftsanforderungen bezogen, was auch einen optimierten TCO (Total Cost of Ownership) im Datenbank-Bereich zur Folge hat.

Capacity Management ist der Hebel und die Drehscheibe für die Effizienz

Bei genauer Betrachtung von ITIL und ISO/IEC20000 stellt man fest, dass das Capacity Management einen signifikanten Einfluss auf alle anderen Prozesse hat. Durch Unterkapazitäten entstehen Störungen, durch Überkapazitäten unnötig hohe Kosten, durch zu große Oracle Datenbank-Umgebungen wird das Problem Management erheblich erschwert und vieles mehr.

Vor allem in heutigen Zeiten, in denen sich die Business-Anforderungen an die IT sehr viel schneller ändern als früher – alleine schon aus dem Grund, dass das von der IT zu unterstützende Business viel dynamischer geworden

ist –, sollte das Kapazitäts- und Ressourcen-Management immer proaktiv sein. Antworten auf nachfolgende Fragen sowie auch auf deren Kombinationen sind erforderlich, damit einerseits die Kapazitäten stets auf die aktuellen und zukünftigen Bedürfnisse des Business hin ausgerichtet sind und andererseits in der IT die hierfür nur wirklich notwendigen Kosten und Aufwände erzeugt werden, zum Beispiel:

- Wo befinden sich nicht genutzte Potenziale und wo kann man Ein-

sparungen realisieren – etwa im Lizenz- und Serverbereich durch Konsolidierungen?

- Welche Datenbank-Server und Cluster-Knoten sind über- beziehungsweise unterlastet und wie kann die Verteilung von Datenbanken auf vorhandene Server effizienter erfolgen?
- Wie ist die langfristige Trend-Entwicklung?
- Welche Anwendungen verwenden die meisten Ressourcen, was sind die Ursachen hierfür und wie wirkt sich das bezüglich der Performance aus?
- Wie viele zusätzliche User und/oder Services bewältigt das System durch wie viele mehrere oder schnellere CPUs?
- Vor allem, was wäre, wenn ...

... mehr User auf dem System wären?

... zusätzlich CPUs für das System verfügbar wären?

... mehr Datenbanken auf dem System laufen würden?

... ein Wechsel auf eine neue Hardware-Plattform stattfinden würde – wie verändert das die Performance beziehungsweise das Systemverhalten?

Proaktives Capacity Management ist Pflicht im Oracle Datenbank-Umfeld

Datenbank-Verantwortliche sind mit entsprechendem Know-how und Tools in der Lage, die Kosten im Datenbank-Bereich entsprechend zu optimieren.

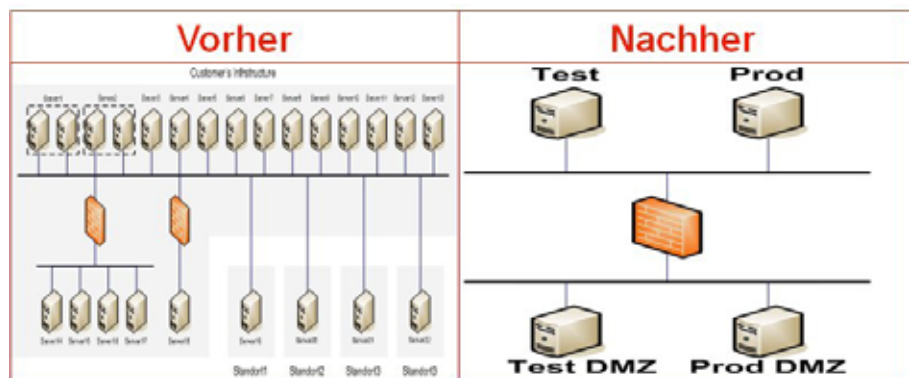


Abbildung 3: Eine Datenbank-Infrastruktur vor und nach einer Optimierung mittels einem auf den Business-Anforderungen basierenden Kapazitäts-, Ressourcen- sowie entsprechenden Configuration Management

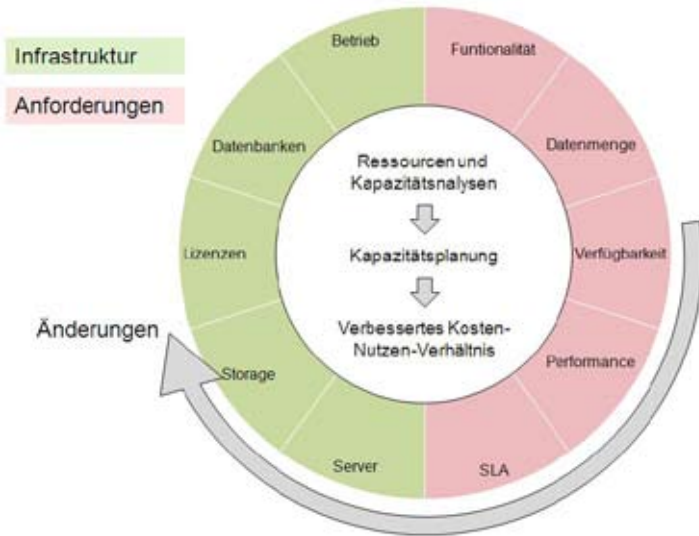


Abbildung 4: Folgen wachsender und schnellerer Änderungsraten – diese führen zu einem nicht optimalen Kosten/Nutzen-Verhältnis

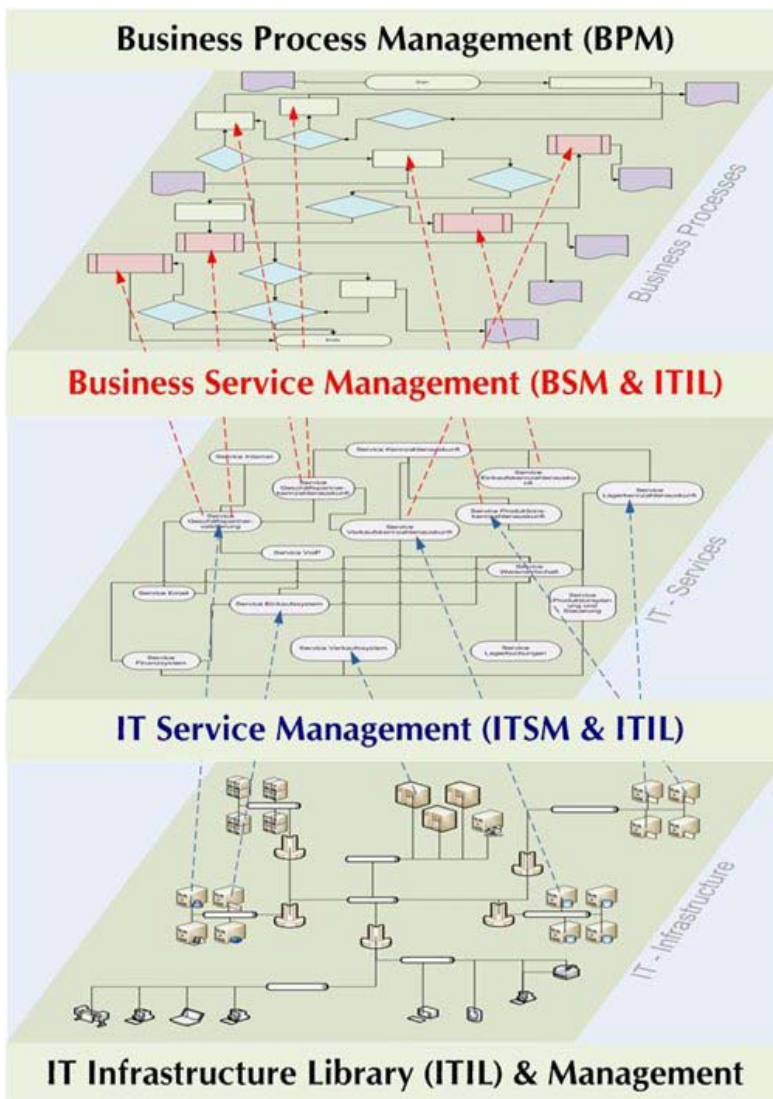


Abbildung 5: Das Capacity Management im Datenbank-Bereich wirkt sich über alle Bereiche hinweg bis zum Business aus – sowohl positiv, als auch negativ

Jeder Euro, der in folgende Bereiche investiert wird, zahlt sich im anschließenden Betrieb oft um ein Vielfaches aus (siehe Abbildung 4):

- „The quality must be built in“ – das heißt, in der Architektur- und Projektphase wird die Grundlage für einen kostengünstigen Betrieb geschaffen. Die Qualität und vor allem die richtige Dimensionierung der Datenbank spielen hier eine entscheidende Rolle.
- Ein permanentes oder zumindest periodisches, aber immer proaktives Kapazitäts- und Ressourcen-Management im IT-Betrieb stellt die Grundlage für die Sicherung der Effizienz dar. Da sich aber die Anforderungen an einen Datenbankservice laufend ändern – meistens durch die sich ändernden Businessanforderungen –, muss man auf diese stets entsprechend reagieren können. Die teure Alternative ist es, Kapazitäten und Ressourcen von Datenbanken stets um ein Vielfaches zu groß auszulegen, um auf der sicheren Seite zu sein.

Vorgehensmodelle, Methoden und Best Practices

Die allgemeinen Best Practices zum Thema Capacity Management sind durch ITIL relativ gut dokumentiert, jedoch verständlicherweise hier sehr allgemein gehalten (siehe Abbildung 5). Methoden haben meistens eine allgemeine Gültigkeit, während sich Rezepte jedoch an spezifischen Fällen ausrichten. Daher müssen die Methoden des Capacity Managements immer auf die spezifischen Anforderungen hin appliziert werden – in diesem Falle die Methode des allgemeinen Capacity Managements auf das Rezept für das Capacity Management für Oracle Datenbanken.

Um die Planbarkeit von Ressourcen in einer IT-Landschaft überhaupt zu ermöglichen, muss das Inventar aller Server, Datenbanken und deren Applikationen sowie vor allem deren Beziehung und Abhängigkeit untereinander zwingend bekannt sein. So einfach das auch klingt, aber die Erfahrung lehrt

uns leider, dass nicht wirklich jede IT-Organisation in der Lage ist, diese Frage per Knopfdruck aktuell, vollständig und richtig zu beantworten. Eine IT-Landschaft sollte aber keinesfalls in statischen Handbüchern dokumentiert werden, sondern stets mittels einer Configuration Management Database (CMDB), was die allererste Voraussetzung für ein erfolgreiches Capacity Management darstellt. Eine zweite Bedingung ist, dass eine Kapazitätsplanung stets ganzheitlich erfolgen muss, das heißt auf Business-, Service- und Infrastruktur-Ebene (siehe Abbildung 6).

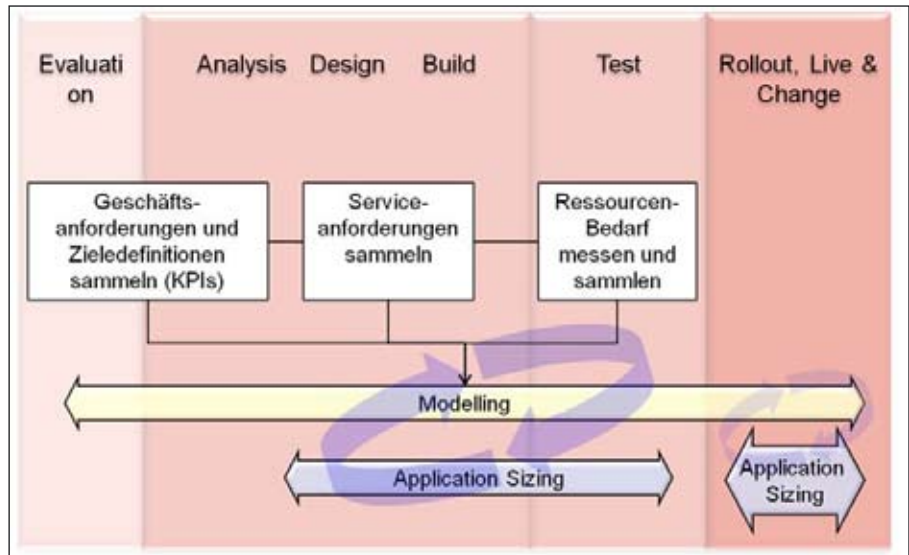


Abbildung 6: Das Capacity Management (die Modellierung und das Sizing) im Kontext des Application Lifecycle

- Das Business Capacity Management hat das Ziel, Veränderungen in der Geschäftsentwicklung zu verstehen, welche einen Einfluss auf den System-Load haben.
- Das Service Capacity Management quantifiziert unter anderem die User- und Applikations-Nachfrage (Service-Demand) und definiert auch die Größenordnungen in Bezug auf Transaktionsvolumen, Antwortzeiten und Verfügbarkeit.
- Auf der Ebene der Infrastruktur ist das Capacity Management für die Überwachung von Auslastung, Transaktionsvolumen und Durchsatz der genutzten Komponenten verantwortlich (CPU, I/O, Memory, Netzwerk).

welche Fragen die Sizing-Prognosen Antworten geben sollen:

- Geht es um den Ausbau eines bestehenden Servers oder um die Konsolidierung mehrerer Server auf einen einzigen?
- Sind Präferenzen bezüglich Hardware-Herstellern zu beachten oder macht der Applikationshersteller bestimmte Vorgaben?

- Wie groß ist das Business-Volumen und welche Antwortzeiten werden erwartet?

Sind diese Festlegungen getroffen, startet man mit der Erhebung von Performance-Daten, wobei darauf zu achten ist, dass hier auch die wirklich relevanten Daten gesammelt werden – einerseits um keinen unnötigen Daten-Overhead zu erzeugen, andererseits

Wenn man an einem zukünftigen Ereignis interessiert ist, so muss man zunächst in die Vergangenheit schauen, was neben einer CMDB auch die historisierten Performance-Daten in Bezug auf beispielsweise Antwortzeiten, Transaktionsvolumen und Auslastung erfordert. Diese Daten müssen permanent gesammelt und am besten in einer Capacity Database (CDB) verwaltet werden, was die dritte Voraussetzung für ein erfolgreiches Capacity Management ist.

Wenn alle oben genannten Voraussetzungen gegeben sind, startet man in ein Capacity Management oder ein Sizing-Projekt, in dem man das Ziel und die Rahmenbedingungen definiert (siehe Abbildungen 7 und 8). Es muss anfangs immer bekannt sein, auf

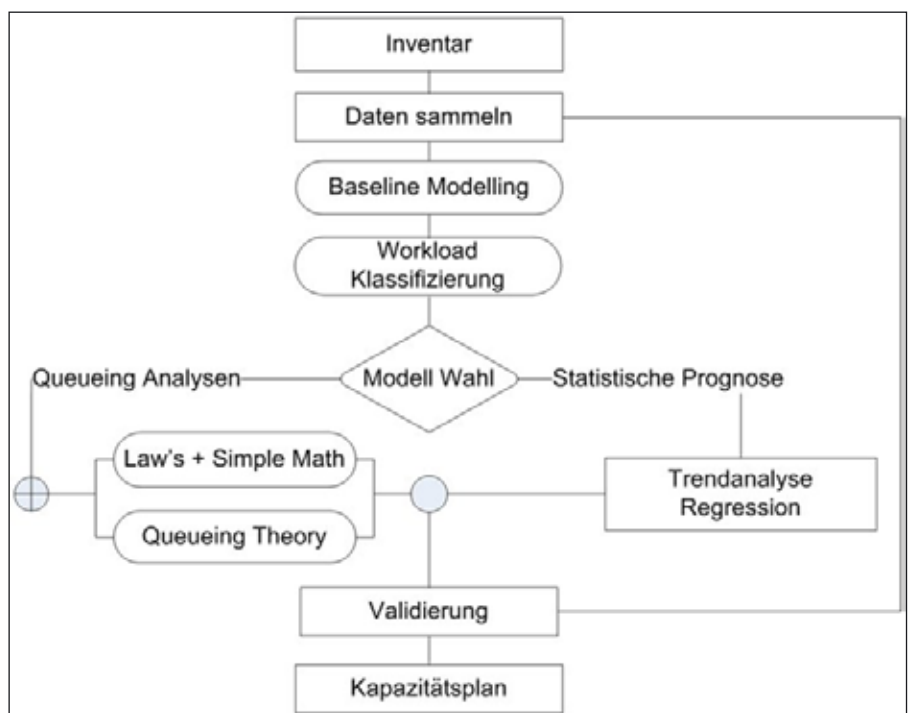
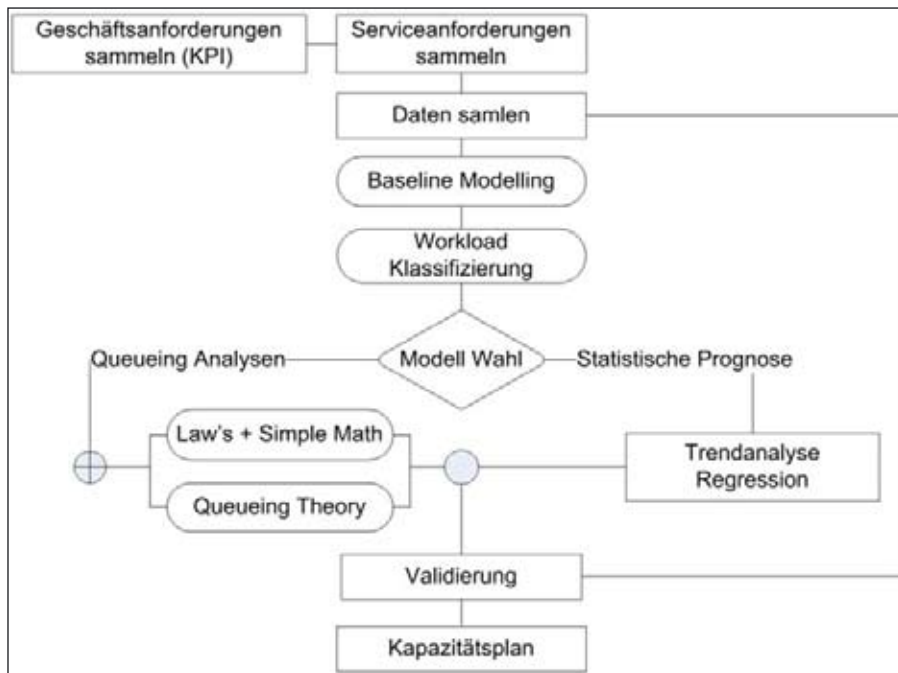


Abbildung 7: Die Vorgehensweise beim Capacity Management für bestehende Systeme



um die hierfür wirklich relevanten Daten zu erhalten.

Anschließend werden die gesammelten Messdaten mittels statistischer Methoden wie der Korrelations- und Regressions-Analyse ausgewertet. Unter Verwendung analytischer Modelle wie der Queuing-Analyse werden dann die Berechnungen von Auslastungen, Service- und Wartezeiten sowie entsprechender Prognosen durchgeführt (siehe Abbildung 9).

Kontakte:

Christian Wischki
 christian.wischki@trivadis.com
 Peter Stalder
 peter.stalder@trivadis.com

Abbildung 8: Die Vorgehensweise beim Capacity Management für neue Systeme

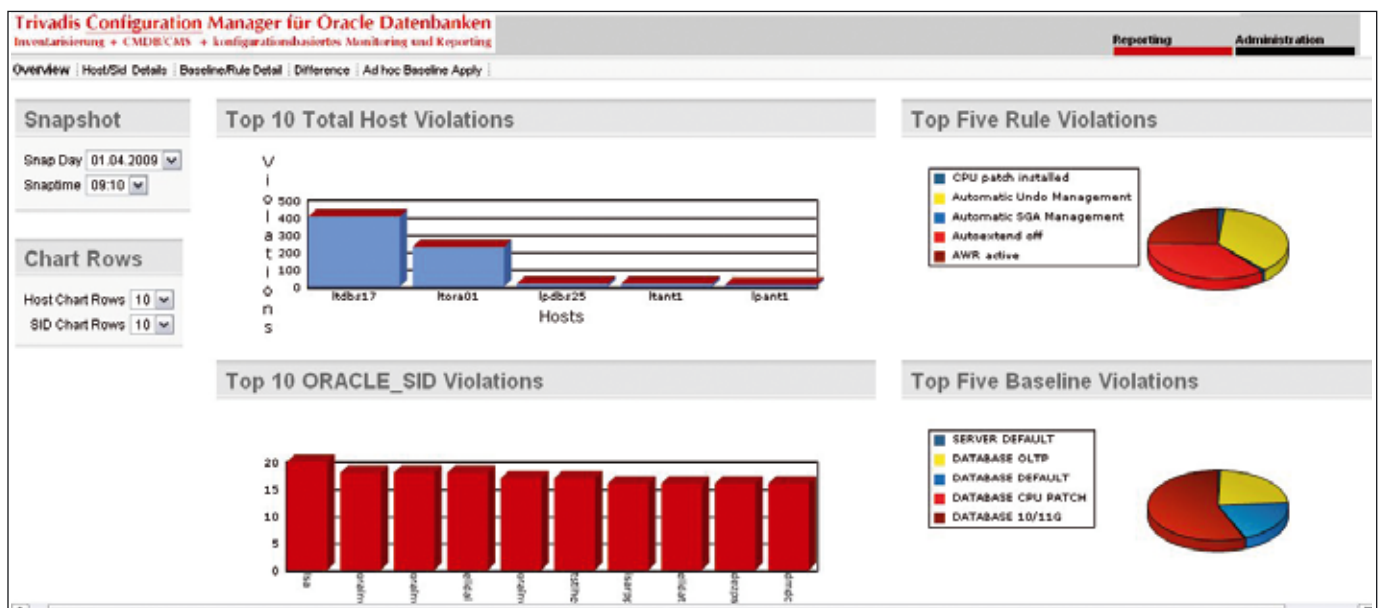


Abbildung 9: Ein Tool hilft bei der automatischen Inventarisierung und Verwaltung von Oracle Datenbank-Landschaften und -Servern

Unsere Inserenten

BzYxS.com Database Optimisation,
 www.bzyxs.com Seite 77
 DOAG 2009 Konferenz + Ausstellung
 www.doag2009.org Seite 54
 IDG-Verlag
 www.computerwoche.de Seite 13
 Neue Mediengesellschaft Ulm mbH
 www.databasepro.de Seite 33
 exensio GmbH,
 www.exensio.de Seite 79

Hunkler GmbH & Co KG,
 www.hunkler.de Seite 3
 MT AG,
 www.mt-ag.com Seite 61
 MuniQsoft GmbH,
 www.munisoft.de Seite 17
 OPITZ CONSULTING GmbH,
 www.opitz-consulting.de Umschlagseite 2
 ORACLE Deutschland GmbH,
 www.oracle.com Umschlagseite 3

ORDIX AG,
 www.ordix.de Seite 69
 PROMATIS software GmbH,
 www.promatis.de Seite 23
 Team GmbH Paderborn,
 www.team-pb.de Seite 15
 Trivadis GmbH,
 www.trivadis.com Umschlagseite 4
 VENTARA AG,
 www.ventara.de Seite 57

Automatisierte Backup-Überprüfung mit dem Recovery-Katalog

Björn Wolter, OPITZ CONSULTING Gummersbach GmbH

Dieser Artikel zeigt, wie man mittels der Repository-Daten des RMAN Recovery-Katalogs eine automatisierte Backupüberwachung implementieren kann. Dazu werden wichtige Views des RMAN-Schemas erläutert. Skriptauszüge und Programmierbeispiele dienen als Anregung zur eigenen Implementierung.

Wer sich bei der Sicherung einer Datenbank für das RMAN-Tool entscheidet, hat die Möglichkeit, optional einen sogenannten „Recovery-Katalog“ zu verwenden. Dieser besteht aus einem Schema, das in einer Oracle Datenbank abgelegt ist. Sämtliche Informationen über die Backups werden bei Einsatz des Katalogs nicht nur in den Controlfiles der Datenbank, sondern zusätzlich in den Tabellen des Katalog-Schemas gespeichert. Ein Vorteil ergibt sich beim Verlust der Controlfiles. Sind diese durch einen Plattencrash verloren gegangen, müssten sie zuerst aus dem Backup manuell zurückgespielt werden, bevor man über RMAN ein Recovery durchführen kann. Bei Verwendung eines Katalogs lässt sich das Recovery dagegen direkt starten, mitsamt der Rücksicherung der Controlfiles. Der Recovery-Katalog bietet sich gerade in großen Umgebungen zur Vereinheitlichung und Standardisierung der Backup-Strategie an. Darüber hinaus können die Backup-Skripte im RMAN-Katalog zentral gespeichert werden.

Backup-Monitoring

In Umgebungen mit vielen Datenbank-Systemen kann der Datenbank-Administrator (DBA) schnell den Überblick über den Backup-Status seiner Datenbanken verlieren. Oft werden RMAN-basierte Backups über Cronjobs angestoßen oder mittels des Grid Controls erzeugt. In beiden Fällen verlässt sich der DBA hierbei auf die Rückmeldung seines eingesetzten Tools oder Skripts. Doch was passiert, wenn keine Meldung kommt? Oft wird gar nicht

überprüft, zu welchen Datenbanken keine Meldung angekommen ist, falls etwa vergessen wurde, den Cronjob anzulegen, oder falls ein unbemerkter Fehler im Skript vorliegt. Die Lösung ist, den Recovery-Katalog als Monitoring-Quelle zu nutzen, weil hier alle Informationen sämtlicher Backups einfließen, die man mit einem geeigneten Analyse-Skript auswerten kann. Statt auf eine Fehler- oder Erfolgsmeldung zu warten, schaut der DBA einfach zu einem gegebenen Zeitpunkt nach, ob im Recovery-Katalog zu einer bestimmten Datenbank Backup-Sets vorhanden sind. Um einen Abgleich zu machen, benötigt er nur eine Liste seiner Datenbanken, die beispielsweise aus dem Grid Control stammen kann.

Wichtige Views

RMAN-Views bieten die Grundlage für das später vorgestellte Verfahren. Das Recovery-Katalog-Schema beinhaltet Tabellen, in denen Informationen über Backup-Sets, Retention Policies, Konfigurationsparameter und Log-Meldungen abgelegt sind. Der Zugriff sollte jedoch auf die schon vorhandenen Views geschehen, da diese auch mit entsprechenden Namen benannt sind und gebündelt die wichtigen Informationen vorhalten. Die wichtigsten Views sind:

- **RC_DATABASE**
Über diese zentrale View des Recovery-Katalogs können Informationen über die registrierten Datenbanken abgefragt werden. Sie enthält neben dem Datenbanknamen die eindeutige Datenbank-ID sowie Informationen über Inkarnationen.
- **RC_BACKUP_SET**
Diese View zeigt Informationen über alle bekannten Backup-Sets aller Datenbanken an. Über die Spalte `BACKUP_TYPE` kann nach dem Backup-Typ gefiltert werden. Die Schlüssel der Backup-Typen sind:
 - D: Full-Backup
 - I: Incremental-Backup
 - L: ArchiveLog-Backup
 Wichtig ist die Spalte „`completion_time`“, mit der man später Auswertungen machen kann, beispielsweise darüber, wann das letzte Backup-Set einer Datenbank gesichert wurde.

Das Statement in Listing 1 gibt die Backup-Sets einer Datenbank in absteigender Reihenfolge aus. Die letzten Backup-Sets stehen dabei oben.

```
select name,db_id, bs_key,
decode(backup_type, 'L','Archive
Log','D','Full','I', 'In-
cremental') TYP,incremental_
level,completion_time from
rc_backup_set a, rc_database b
```

NAME	DB_ID	BS_KEY	TYP	INCREMENTAL_LEVEL	DATUM
DB1	2933979454	486	Full		27.05.2009 20:17
DB1	2933979454	366	Archive Log		27.05.2009 20:14
DB1	2933979454	331	Incremental	1	27.05.2009 20:13

Tabelle 1: Backup-Sets einer Datenbank im RMAN-Katalog nach Aufruf des Listings 1

```
where b.db_key= a.db_key AND
name='<DBNAME>' order by completion_time desc;
```

Listing 1

• RC_BACKUP_FILES

Diese View zeigt Details aller bekannten Backup-Dateien einer Datenbank an. Hier sind unter anderem gesicherte Datendateien innerhalb eines Backup-Sets zu sehen. Um diese View abfragen zu können, muss zuvor eine Prozedur aufgerufen werden, die die Datenbank-ID intern setzt. Es werden nur Backup-Files von jeweils einer Datenbank angezeigt.

Die nachfolgenden Statements dienen der Abfrage von Backup-Files einer Datenbank innerhalb eines gesetzten Zeitraums:

- Ermitteln der Datenbank-ID

```
select dbid from rc_database
where name='<dbname>';
```

- Ausführen der Prozedur

```
CALL DBMS_RCVMAN.SETDATABASE
(null,null,null,<dbid>,null);
```

- Anzeigen der gesicherten Dateien innerhalb eines Zeitraums

```
select BACKUP_TYPE,FILE_
TYPE,FNAME, COMPLETION_TIME
zeit,BYTES, bs_incr_type from
rc_backup_files where
completion_time>to_
date(,05.06.2009 00:00:00', 'dd.
mm.yyyy hh24:mi:ss')
and completion_time<to_
date(,05.06.2009 23:59:00', 'dd.
mm.yyyy hh24:mi:ss')
order by pkey desc;
```

• RC_RMAN_BACKUP_JOB_DETAILS

Diese View enthält Informationen zu bereits gelaufenen Backups. Das Attribut „Status“ gibt dabei an, ob das Backup erfolgreich oder fehlerhaft ausgeführt worden ist. Die dazu gehörenden Fehler- oder Erfolgsmeldungen werden in der nachfolgend beschriebenen View RC_RMAN_

OUTPUT angezeigt. Listing 2 zeigt den Status sämtlicher Backup-Jobs einer Datenbank an.

```
select db_name,start_time,
end_time,status from rc_rman_
backup_job_details where db_
name='<dbname>' order by start_
time desc;
```

Listing 2

• RC_RMAN_OUTPUT

Diese View zeigt den Output der RMAN Sessions an. Über den „Session_Key“ der View RC_RMAN_BACKUP_JOB_DETAILS lässt sich ein Output referenzieren (siehe Listing 2).

```
Select output from rc_rman_out-
put where session_key=<key>;
```

Listing 3

Überprüfungskonzept

Die vorgenannten Views bieten zwei verschiedene Verfahren zur Überprüfung von Backups an. Die erste Möglichkeit besteht darin, das Attribut „Status“ aus der VIEW RC_RMAN_BACKUP_JOB_DETAILS zu überprüfen. Bei einem Fehler wird das Feld mit „error“ gekennzeichnet und ein Output-Text hilft, eine Diagnose zu stellen. Jedoch ist zu bedenken, dass eventuell gar kein Output existiert, falls der Backup-Job aus technischen Gründen nicht ausgeführt worden ist. In diesem Fall ist weder eine Erfolgs- noch eine Fehlermeldung vorhanden. Somit lässt sich eine gute Übersicht über die gelaufenen Backup-Jobs pro Datenbank erstellen.

Eine weitere Möglichkeit besteht in der Auswertung der View RC_BACKUP_SET.

Aus Performance-Gründen empfiehlt es sich, eine materialisierte View in einem zweiten Schema zu erstellen und die Anzahl der Datensätze auf die zurückliegenden 30 Tage zu begrenzen (siehe Listing 4).

```
select db_id,pieces,status,a.
DB_KEY, BACKUP_TYPE, COMPLETI-
ON_TIME,START_TIME,INCREMENTAL_
```

```
LEVEL,
b.name from rman.rc_backup_set
a, rman.rc_database b where
a.db_key=b.db_key AND completi-
on_time> SYSDATE-30
order by Name,DB_
KEY,completion_time desc;
```

Listing 4

Die Idee dabei ist, für jede registrierte Datenbank zu ermitteln, wann das letzte Archivlog-, Incremental- und Full-Backup gelaufen ist und diesen Zeitstempel für weitere Auswertungszwecke in einer Tabelle temporär abzulegen. Die Implementierung kann mittels einer PL-SQL-Schleife erfolgen. Für die Gesamtliste aller Datenbanken können die Daten aus der VIEW RC_DATABASE herangezogen werden. Falls ein Grid Control in der Umgebung genutzt wird, könnte man sich der Management-Views des SYSMAN-Schemas bedienen. Wichtig ist, dass bei der Überprüfung der Zeitstempel die Datenbank-ID als Schlüssel verwendet wird, da der Datenbankname in einigen Umgebungen mehrfach vorkommen kann und sonst keine Unterscheidung der Datenbanken möglich wäre. Nach der Ablage der Zeitstempel eines zuletzt gelaufenen Archivlog-, Incremental- und Full-Backups erfolgt eine Auswertung.

Zunächst definiert man ein Zeitfenster, in dem mindestens ein Archivlog-, Incremental- und Full-Backup für jede Datenbank gelaufen sein muss. Zur Überprüfung wird der Zeitstempel des jeweils letzten Backups einer Datenbank herangezogen und das Delta bestimmt. Sollte die Zeitdifferenz zu groß sein – ein sicheres Anzeichen für ein vorliegendes Problem –, erfolgt eine Warnung. Diese Verfahrensweise kann in verschiedenen Programmiersprachen implementiert werden. Der Vorteil ist, dass nicht Erfolgs- oder Fehlermeldungen ausgewertet werden, sondern wirklich geprüft wird, was dem RMAN-Tool bekannt ist. Wenn die Gesamtliste der Datenbank-Systeme dann noch aus einer unabhängigen Quelle stammt, würden auch die Datenbanken im Monitoring auftauchen, bei denen vergessen wurde,

ein „register database“ auszuführen oder den RMAN-Job einzurichten. Die Kombination beider Verfahrensweisen würde das optimale Monitoring darstellen. Abbildung 1 zeigt ein Beispiel mit beiden Verfahrensweisen als Web-Anwendung.

Hostname	Datenbank	Fehlertyp	Aktion	Grund
GMVDC1	OCQWMT	Archive-Log Backup fehler	Fehlermeldung	⊗ Joblauf: 02.06.09 22:03
GMBRT1	OCTRAQ	Incremental Backup fehler	Fehlermeldung	⊗ Joblauf: 02.06.09 22:01
GMEE3	OCNRA	Kein Full-Backup in Zeitraum		⊗ zuletzt: 24.05.09 22:28
GMVD1	OCBJW	Kein Incremental in Zeitraum		⊗ zuletzt: 01.06.09 22:51
GMQDX2	OCJLA	Kein Incremental in Zeitraum		⊗ zuletzt: 01.06.09 22:27
GMIDES1	OCDSK	Kein Archive-Log in Zeitraum		⊗ zuletzt: 25.05.09 11:38
GMLAS2	OCARK	Kein Archive-Log in Zeitraum		⊗ zuletzt: 02.06.09 09:37

Abbildung 1: Backup-Monitoring als webbasierte Anwendung

Fazit

Der RMAN-Katalog bietet mit seinen Views eine gute Grundlage für die Implementierung eines Backup-Monitorings von Datenbanken. Lediglich über die Darstellung und die

Form der Auswertung muss man sich Gedanken machen. Die Qualität des Datenbank-Betriebs im Unternehmen wird auf diese Weise gesteigert, weil nun Fehler sehr früh erkannt werden können und der DBA präventiv

agieren kann, bevor es zum Daten-GAU kommt.

Kontakt:

Björn Wolter

bjoern.wolter@opitz-consulting.com



Wegweiser durch Oracle

Till Brügelmann, ORACLE Deutschland GmbH

Basierend auf dem Feedback der Kunden und Partner hat Oracle in Deutschland bereits im Jahr 2000 ein Customer Care Center mit dem Namen „Hallo Oracle“ eingerichtet. Ziel ist es, einen zentralen und kundenorientierten Eingang zu Oracle zu bieten.

Wer kennt sie nicht? – Die von Komiker-Ikone Karl Valentin erschaffene Figur des Buchbinders Wanninger, der mit seiner telefonischen Anfrage in einer fiktiven Firma immer nur weiterverbunden wird und nicht den richtigen Ansprechpartner findet. Was für den Zuschauer sehr belustigend wirken mag, kann aber dem Betroffenen schnell den Puls auf 200 treiben.

Zugegeben – auch Oracle mag auf den ersten Blick riesig und komplex wirken. Durch Wachstum und Integration zahlreicher Firmen hat Oracle in den letzten Jahren an Breite und Vielfältigkeit seines Produkt-, Lösungs- und Service-Portfolios deutlich zugelegt. Und wo sollte ein orientierungssuchender Mensch bei Oracle zuerst anrufen: In der Hauptverwaltung München oder in der nächstgelegenen von deutschlandweit neun Geschäftsstellen?

Seit 2000 ist das Customer Care Center mit dem Namen „Hallo Oracle“ eingerichtet. Die vier langjährigen Mitarbeiter/innen dieses Teams, die in der Geschäftsstelle Düsseldorf arbeiten, möchten Ihnen gerne jede Orientierungsfrage rund um Oracle beantwor-

ten (selbstverständlich deutschsprachig) und die Kontaktabbahnung bzw. Zusammenarbeit so einfach und komfortabel wie möglich machen.

Sehr viele Anfragen können vom Team ad hoc beantwortet werden, indem alle verfügbaren Informationssysteme genutzt werden, und nicht zuletzt durch die hohe persönliche Erfahrung der Teammitglieder. In allen anderen Fällen holt „Hallo Oracle“ die notwendigen Informationen bei den entsprechenden Fachabteilungen ein oder stellt zuverlässig den Kontakt für alle weiteren Schritte her. Nicht der Kunde, sondern seine Anfrage wird intern weitergeleitet und geklärt; so wird der eingangs erwähnte „Wanninger-Effekt“ wirkungsvoll vermieden.

Nobody is perfect – daher hat „Hallo Oracle“ selbstverständlich auch stets ein offenes Ohr für Ihr kritisches Feedback oder Beschwerden. In unserer Rolle als „Anwalt des Kunden“ bleibt nichts unversucht, um aufgetretene Problemfälle zu lösen und Ärgernisse aus der Welt zu schaffen (sofern notwendig gegebenenfalls auch unter Einbeziehung der Geschäftsführung).

Die tägliche Herausforderung ist die riesige Bandbreite der eingehenden Anfragen (Tendenz steigend), denn sie stammen auch von sehr unterschiedlichen Quellen: Bestehende oder potentiell neue Kunden und Partner, Entwickler, Journalisten, Bewerber, Studenten und Oracle-Fans – hier ist jeder richtig! Die häufigsten Anfragen beziehen sich auf die Suche nach geeigneten Ansprechpartnern für eine bestimmte Thematik oder Angelegenheit sowie die Unterstützung bei der Web-Recherche auf Oracle.com und anderen Oracle-Internetplattformen.

Auf folgenden Wegen ist „Hallo Oracle“ zu erreichen:

- Telefon: 0180-2-ORACLE bzw. 0180-2-672253
- Telefax: 0180-2-ORAFAX bzw. 0180-2-672329
- E-Mail: info_de@oracle.com
- Web: www.oracle.de, Link auf der Homepage ganz unten unter „Kontakt“

Kontakt:

Till Brügelmann
till.bruegelmann@oracle.com

Replikation mit Oracle Streams

Martin Bach, Markit Group Ltd (London)

Replikation existiert schon lange im Oracle-Umfeld in Form der "Advanced Replication"-Option, die standardmäßig mit der Enterprise Edition der Datenbank ausgeliefert wird. Seit Oracle 9i gibt es noch eine weitere Replikationstechnologie: Oracle Streams – im übertragenen Sinne also Datenströme.

Verfolgt man die Dokumentation der Advanced Replication, kommt man nach Ansicht des Autors nicht umhin zu vermuten, dass diese Technologie nicht mehr aktiv weiterentwickelt wird. Dieser Verdacht scheint sich in Metalink zu bestätigen – viele Artikel dort beziehen sich auf Oracle 9i, eine Version, die schon viele Jahre ihren Dienst verrichtet hat. Streams arbeitet oft unerkannt und still als Grundlage für andere Komponenten und Produkte wie etwa Change Data Capture und Audit Vault. Grund genug, die Technologie hinter Streams etwas näher zu untersuchen.

Oracle Streams verwendet das gleiche Konzept, das schon für die Logical-Standby-Datenbank zum Einsatz kommt. Im Gegensatz zur ursprünglich vorgestellten Physical-Standby-Datenbank, die Bit für Bit der primären Instanz gleicht, kommt bei der logischen Standby-Datenbank der Log-Miner zum Einsatz, um Änderungen der primären Instanz zur Standby-Datenbank zu replizieren. Vereinfacht gesagt extrahiert Log-Miner SQL-Anweisungen aus den Redo-Log-Dateien.

Exkurs: Redo-Log-Dateien

Alle datenverändernden Transaktionen, die Benutzer mittels Commit-Befehl dauerhaft in die Datenbank geschrieben haben, müssen im Fall des Absturzes der Datenbank-Instanz wiederherstellbar sein. Da Oracle nicht mit jedem Commit den Cache zurück auf das Platten-Subsystem schreibt, kommen Redo-Log-Dateien zum Einsatz. Sollte nun die Datenbank-Instanz tatsächlich aus irgendeinem Grund unsauber terminiert werden, ist der Inhalt des flüchtigen Cache verloren und die Datenbank in einem inkonsis-

ten Status. Beim nächsten Start der Instanz erkennt Oracle den Crash und zieht mithilfe der Redo-Log-Dateien die ausstehenden Transaktionen nach. Transaktionen, die zum Zeitpunkt des Crash nicht abgeschlossen waren, werden rückgängig gemacht.

Für logische Standby-Datenbanken wie auch Streams schreibt die Datenbank zusätzliche Informationen in die Redo-Log-Dateien, in der Dokumentation „supplemental logging“ genannt. Dies ist auf jeden Fall ein zusätzlicher Aufwand, Oracle nennt etwa fünf Prozent Overhead. Im Einzelfall ist durch Tests abzuwägen, ob die Anwendung dadurch keine Performance-Einbußen erleidet. Vorsicht ist vor einer Generalisierung geboten: Anders als bei Data Guard wird „supplemental logging“ nur für replizierte Datenbank-Objekte verwendet.

Wie bereits eingangs erwähnt, verwenden die logische Standby-Datenbank und Streams gemeinsame Komponenten, wobei Streams aber einige Vorteile aufzuweisen hat. Die wichtigsten davon sind eine höhere Flexibilität und die plattformübergreifende Replikation.

Vorteile gegenüber Data Guard

Streams bietet eine Menge an Detailverbesserungen, vor allem im Bereich der Administration. Es erlaubt zudem viel mehr Einblick in seine Funktionsweise. Zudem lässt Streams im Gegensatz zu Data Guard die Replikation einer Untermenge an Daten zu. Diese Fähigkeit nutzt zum Beispiel das Change Data Capture (CDC), eine Technik zum effektiven Laden eines Data Warehouse.

Mittels Streams lassen sich beispielsweise sternförmig verteilte Da-

tenbanken realisieren, wobei mehrere Quelldatenbanken ein Zielsystem in Echtzeit mit Daten versorgen. Die Auswertungen mithilfe dieser Daten können ressourcensparend auf dem Zielsystem vorgenommen werden.

Streams ist mit Einschränkungen plattformunabhängig – der Autor hat erfolgreich Daten aus einer Produktionsdatenbank 10gR2 in einem auf 11g basierenden Management Informationssystem bereitgestellt. Streams eignet sich mithin als Migrationsstrategie mit sehr geringer beziehungsweise keiner Ausfallzeit.

Streams besteht im Wesentlichen aus drei Komponenten:

- Aufnahme (Capture)
- Verteilung (Propagation)
- Wiedergabe (Apply)

Ein Teil der System Global Area, also des der Datenbank zur Verfügung stehenden Arbeitsspeichers, wird für die Replikation verwendet. Der sogenannte Streams Pool kann – wenn eingestellt – von Oracle selbst verwaltet werden, sollte aber mindestens 200 MB betragen.

Der schematische Prozess der Replikation mit Oracle Streams läuft wie folgt ab:

- Der Aufnahmeprozess liest Daten aus den Redo-Log-Dateien. Dabei bestimmen Regeln, welche Informationen zu replizieren sind. Diese Regeln werden zu Beginn der Streams-Replikation festgelegt, sie sind aber nicht statisch, sondern lassen sich zur Laufzeit verändern.
- Ausschließende (negative) Regeln existieren seit Oracle 10 und ersparen dem Administrator einiges an Handarbeit. Je komplexer die Re-

geln werden, desto länger dauert logischerweise ihre Ausführung – von der CPU-Belastung ganz zu schweigen.

- Sind die relevanten Informationen erst einmal aus den Redo-Log-Dateien gelesen, werden sie als Nachrichten verpackt. In der Oracle-Dokumentation heißen diese Nachrichten „Logical Change Record“ (LCR).
- Nachdem sie verpackt sind, werden die Informationen zur Ziel-Datenbank auf die Reise geschickt. Hinter den Kulissen arbeitet dabei Advanced Queuing; dies ist wahrscheinlich für viele DBAs der erste Kontakt mit dieser Technologie. Oracle garantiert die Zustellung der Nachrichten, gibt aber nach 16 erfolglosen Sendeversuchen auf und stoppt die Verteilung der Nachrichten an die betreffende Ziel-Datenbank.
- Nach dem Versand der Nachricht ist die Arbeit für die Quell-Datenbank erledigt.
- Nach dem Eintreffen der Nachricht am Ziel wird diese zerlegt, in SQL-Befehle transformiert und angewendet. Die Replikation ist damit erfolgreich abgeschlossen.

Natürlich gibt es viele Variationen dieses Beispiels, und Streams kann wesentlich mehr. Die Aufzählung dieser Möglichkeiten würde aber den Rahmen des Artikels sprengen.

Einschränkungen

Für Streams und logische Standby-Datenbanken gelten ähnliche Einschränkungen bezüglich der unterstützten Datentypen. Der Capture-Prozess kann derzeit Änderungen der folgenden Datentypen nicht verarbeiten:

- SecureFile LOB (schließt BLOB, CLOB und NCLOB ein)
- BFILE
- rowids
- vom Benutzer erstellte Datentypen
- bestimmte XML-Type-Repräsentationen
- Mediendaten-Typen, Geodaten (Spatial Option)

Streams Praxisbeispiel

Im folgenden Beispiel soll nun das Benutzerschema DOAG von der Datenbank PROD in die Datenbank MIS repliziert werden. Eine der Anforderungen dabei ist, dass neue Datenbank-Objekte im Schema der Quell-Datenbank ohne Eingreifen des Administrators zusätzlich zu den anderen Objekten repliziert werden. Eine Transformation der Daten auf der Zieldatenbank ist nicht gefordert.

Seit Oracle 10.2 ist die Konfiguration von Streams recht einfach geworden. Anstatt je ein API pro Komponente zu beherrschen, reicht nun der Aufruf einer Unterprozedur des Pakets DBMS_STREAMS_ADM aus. Zugegeben, die Anzahl der Parameter dieser Prozeduren ist recht umfangreich, dafür wird der DBA aber nach erfolgreicher Ausführung mit einer funktionierenden Replikation belohnt.

Zuvor müssen aber noch einige kleine Vorarbeiten abgeschlossen sein. Wie bei Oracle üblich, gilt es, einen Benutzer anzulegen, der als Administrator für die Streams-Komponente dient. Im Normalfall wird dieser „STRMADM“ genannt, ein solches Benutzerkonto muss sowohl auf der Quell- als auch auf der Ziel-Datenbank angelegt werden. Diejenigen DBAs, die Advanced Replication kennen, werden sich sicher an den REPADM und verwandte Benutzer erinnern.

Es ist empfehlenswert, STRMADM einen dedizierten Tablespace zuzuweisen und auf gleichförmige (uniform) Extents zu verzichten. Nachrichten (LCRs), die aus irgendwelchen Gründen nicht zustellbar sind, werden in diesem Tablespace gespeichert. Je nach Anwendung kann sich eine Menge von Daten ansammeln, es sollte also ausreichend Speicherplatz vorgehalten werden. Zudem ist es notwendig, bidirektionale, private Datenbank-Links zwischen den STRMADMIN-Benutzern anzulegen. Diese werden später für die Instanzierung gebraucht. Sind alle diese Aufgaben abgeschlossen, ruft STRMADMIN die Maintain-Schema()-Prozedur auf der Quell-Datenbank auf. Listing 1 zeigt ein Beispiel für den Aufruf von Maintain Schema().

```
DBMS_STREAMS_ADM.MAINTAIN_SCHEMA(
  schema_names
=> ,DOAG',
  source_directory_object
=> ,NA',
  destination_directory_object
=> ,NA',
  source_database
=> ,PROD',
  destination_database
=> ,MIS',,
  perform_actions
=> ,FALSE',
  script_name
=> ,streams_impl.sql',
  script_directory_object
=> ,STREAMS_SRC_DIR',
  capture_name
=> ,CAPTURE_DOAG',
  capture_queue_table
=> ,CAPT_Q_T_DOAG',
  capture_queue_name
=> ,CAPT_Q_DOAG',
  propagation_name
=> ,PROP_DOAG',
  apply_name
=> ,APPLY_DOAG',
  apply_queue_table
=> ,APPLY_Q_T_DOAG',
  apply_queue_name
=> ,APPLY_Q_DOAG',
  log_file
=> ,streams_DOAG.log',
  bi_directional
=> FALSE,
  include_ddl
=> TRUE,
  instantiation
=>
  DBMS_STREAMS_ADM.INSTANTIATION_SCHEMA_NETWORK);
```

Listing 1

Die Methode INSTANTIATION_SCHEMA_NETWORK bewirkt, dass die Ziel-Datenbank einen Data-Pump-Prozess startet, um die Daten von der Quell-Datenbank zu importieren. Dies ist nicht in allen Fällen praktisch; es gibt weit mehr Methoden zur Instanzierung von Streams.

Will man mehr als ein Schema replizieren, so ist dem Parameter „schema_names“ eine durch Kommata abgesetzte Liste zu übergeben. Die Prozedur darf keinesfalls mehrfach aufgerufen werden, da sonst mehrere Capture-Prozesse entstehen, was nur unnötig Ressourcen verschwendet.

Neben dem Aufruf von DBMS_STREAMS_ADM stehen noch der

Enterprise Manager sowie weitere Methoden zur Instanzierung der Daten in der Ziel-Datenbank zur Verfügung. Der interessierte Leser sei auf die weiterführende Dokumentation verwiesen.

Nach dem Abschluss der Prozedur ist die Replikation erstellt und voll funktionsfähig. Im Schema DOAG auf der Quelldatenbank neu erstellte Tabellen sowie alle weiteren DDL-Operationen werden automatisch und ohne Zutun des Administrators auf der Ziel-Datenbank ausgeführt.

Die Administration der Streams-Replikation erfolgt am einfachsten mittels Enterprise Manager. Anders als eine Data-Guard-Umgebung lässt sich Streams sogar mit der Datenbank-Konsole steuern.

Wer sich intensiver mit den zu Streams gehörenden V\$-Views beschäftigen möchte, findet in der Metalink Note 273674.1 „Streams Configuration Report and Health Check Script“ Links zu Skripten mit jeder Menge an Diagnose-Informationen.

Fazit

Oracle Streams ist eine zwar noch recht neue, aber dennoch viel versprechende Technologie, die sich anschickt, anderen Replikations-Mechanismen den Rang abzulaufen. Die Komplexität kann Administratoren allerdings abschrecken, da drei dicke Handbücher zu dem Thema existieren.

Kontakt:

Martin Bach
martin-bach1@gmx.net

DOAG-Regionaltreffen in München

April 2009

Am 22. April 2009 standen zwei sehr unterschiedliche Themen auf der Agenda des Regionaltreffens, das wie immer im großen Hörsaal der Fachhochschule München stattfand.

Christian Pöhl von der Secaron AG berichtete über „Compliance im Unternehmen“. Compliance hat auch Auswirkungen auf die Arbeit des Administrators. Er legt die Nutzer an, vergibt Zugriffsrechte, sorgt für Backup und Recovery, stellt die Verfügbarkeit sicher und vieles mehr. Seine Tätigkeit muss nachvollziehbar sein, auch zum Schutz des DBA, der damit unter Umständen nicht mehr „in der Schusslinie“ ist.

Martin Schlichenmayer von Oracle Deutschland referierte kurzweilig über SOA und stellte folgende Thesen auf, die er im Laufe seines Vortrages auch belegte:

- SOA ist keine konkrete Technik, sondern ein abstrakter Begriff für ein Managementmodell
- SOA schreibt keine spezielle Technologie vor
- SOA ist nicht neu
- Man kann SOA nicht kaufen
- SOA ist kein reines IT-Thema

Nach beiden Vorträgen gab es noch eine ausführliche Diskussion, die wir, wie immer, im Löwenbräukeller fortsetzten.

Mai 2009

Weil in der Fachhochschule zu Beginn des Semesters kein Raum mit dem erforderlichen zeitlichen Vorlauf zur Verfügung stand, sind wir für die Mai-Veranstaltung wieder in die Oracle Hauptverwaltung ausgewichen. Es stand nur ein Vortrag auf dem Programm. Unter dem Titel „Der Designer ist tot – lang lebe der Designer“ sprach Heinz Mielimonka darüber, wie der Weg vom Oracle Designer hin zu SQL Developer Data Modeling verlaufen könnte.

Da zum Zeitpunkt des Vortrages lediglich eine Betaversion des SQL-Developers zur Verfügung stand, konnte die eine oder andere Funktion noch nicht gezeigt werden, und es wurde auch noch auf manchen Fehler hingewiesen, der aber in der Produktivversion behoben sein soll. Trotzdem erhielten die Zuhörer einen umfassenden Überblick über das Thema und der Vortrag ist gut angekommen. Das zeigte auch die rege Teilnahme im gut gefüllten Raum. Beim anschließenden Büffet (vielen Dank an den Hausherrn) wurde dann weiter gefachsimpelt.

Juni 2009

Wieder zurück in der Fachhochschule trafen wir uns am 22. Juni 2009. „JMS mit Oracle Advanced Queuing“ lautete der Titel des Vortrags von Christian Hartmann. Nach einer kurzen Einführung in die Thematik führte der Referent am Beispiel eines von ihm durch-

geführten Projekts vor, wie mit Java Message Service (JMS) die Kommunikation zwischen verschiedenen Welten (hier PL/SQL und Java) realisiert werden kann. Die gezeigten Beispiele waren leicht verständlich. Insgesamt gab der Vortrag einen sehr guten Überblick über das Thema.

Mit der Gleichung „Oracle + XML = Oracle XML DB“ war der Vortrag von Christoph Burandt, ORACLE Deutschland GmbH, überschrieben. Er zeigte die unterschiedlichen Möglichkeiten auf, wie XML-Daten in der Datenbank abgespeichert werden können. Zu Beginn wurden aber im Rahmen einer kurzen Einführung in das Thema noch einige wesentliche Begriffe erläutert.

Beide Vorträge fanden große Resonanz bei den Zuhörern. Die Referenten standen auch für die Nachbesprechung im Löwenbräu zur Verfügung. An alle Referenten geht der Dank für die Zeit, die sie aufgewendet haben, um ihren Vortrag vorzubereiten und auf dem DOAG Regionaltreffen zu präsentieren.

Die Vorträge liegen auf dem DOAG-Server für Mitglieder bereit. Themenwünsche, die wir bei einem der nächsten Regio-Treffen behandeln sollten, schicken Sie bitte per Mail an regio-muenchen@doag.org. Wir nehmen Ihre Anregungen gerne auf.

Kontakte:

Franz Hüll
Andreas Ströbel
regio-muenchen@doag.org

Elegante und effiziente Umsetzung der Best Practices mit dem Oracle Data Integrator

Karol Hajdu, Trivadis AG

Der Oracle Data Integrator (ODI, ehemals Sunopsis) wird in unternehmensweiten DWH- und EAI-Projekten immer beliebter. Dieser Artikel beleuchtet einen der Gründe: das Konzept der Knowledge Modules. Sie erlauben, die Generierung des Quellcodes weitgehend frei anzupassen.

In Werbeartikeln wird über den Flexibilitätsvorteil hinaus noch eine weitere Botschaft transportiert: Durch den Kauf von ODI erwerbe der Kunde kostenfrei auch eine Library der Knowledge Modules und somit auch die Best-Practice-Lösungen der üblichen Problemstellungen aus dem Umfeld DWH und Datenintegration. Erfahrungen zeigen jedoch: Was für den einen „Best Practice“ heißt, kann der andere auch als „schlecht“ oder „ungenügend“ einstufen. Anhand des aufgeführten Beispiels kann sich jeder seine eigene Meinung bilden. Die Erfahrung des Autors: ODI ist ein geeignetes Tool, um vorhandene Best Practices zu automatisieren. Für den, der noch keine Best Practice hat: Die frei verfügbaren Knowledge Modules von Oracle stellen meistens leider noch keine fertige Lösung dar.

Im ODI wurde das Konzept der deklarativen Software-Entwicklung recht konsequent und erfolgreich umgesetzt. Die Definitionen, „was“ gemacht werden soll, „wo“ und „wie“ es gemacht wird, sind weitgehend getrennt festgelegt. Die Transformationslogik (was) wird in Form von Eigenschaften und Ausdrücken (Expressions) im sogenannten Datenmodell (Metadaten über Datenstrukturen) und in ODI-Interfaces (Mappings) spezifiziert. Anschließend kann für diese Objekte die Ausführungslokation (wo, sogenannte „Execution Location“) zugeordnet werden. Ein durchdachtes, im ODI eingebautes Regelwerk leitet daraus die Datenflüsse ab. Wie (also mit welcher Strategie und mit welchen Methoden) diese Datenflüsse in bestimmter Technologie umgesetzt werden sollen, wird durch die Zuordnung eines sogenann-

ten „Knowledge Modules“ wiederum deklarativ festgelegt.

Dazu ein Beispiel, wie die Aspekte „was“, „wo“ und „wie“ in einem ODI-Interface zusammenspielen: Die Mappings im ODI-Interface spezifizieren, dass beispielsweise zwei Tabellen aus einer Datenbank mit einem JOIN verknüpft werden sollen und dass das Resultat in eine Zieltabelle in einer anderen Datenbank eingefügt werden soll. Die Execution Location des JOINS spezifiziert, in welcher Datenbank der JOIN zur Ausführung kommen soll. Das Knowledge Module definiert auch, „wie“ die Resultatmenge in die Zieltabelle geschrieben wird, entweder als ein LOOP der Einzel-INSERTs (satzori-

entiert) oder als ein INSERT...SELECT-Befehl (mengenorientiert).

Ein Knowledge Module ist eine Sequenz von Schritten (Commands). Diese Schritte definieren, wie der Quellcode für einen Datenfluss im ODI-Interface aufgebaut (generiert) werden soll. Die Aufgabe des Knowledge Modules ist in Abbildung 1 schematisch dargestellt.

Übernehmen und Weiterentwickeln

Bei der Installation von ODI wird auch eine Menge von Knowledge Modules für verschiedenste Ziel-Technologien kostenfrei mitgeliefert. Ein Auszug daraus ist in Abbildung 2 dargestellt.

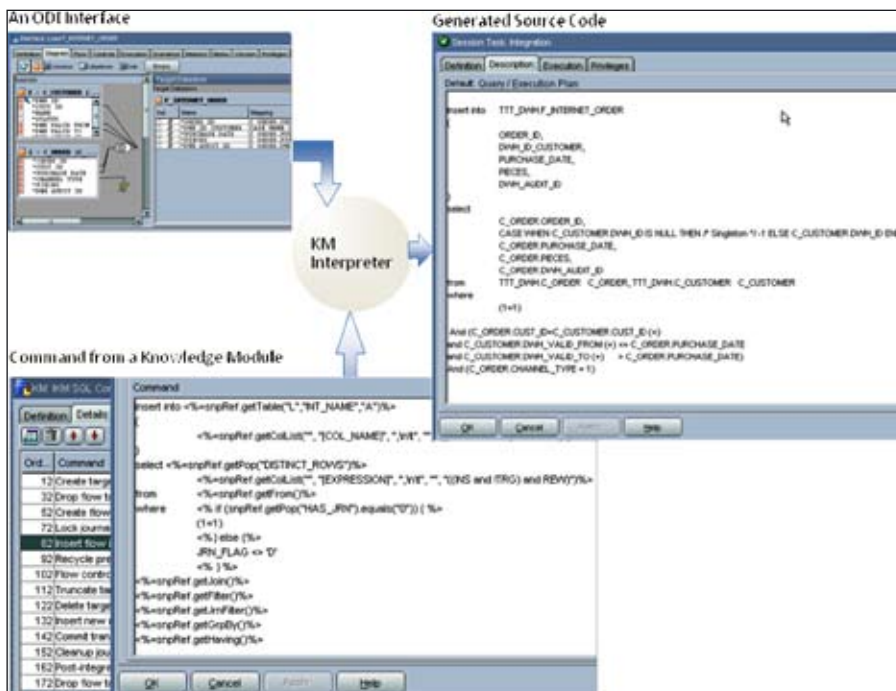


Abbildung 1: Prinzip und Verwendungszweck von Knowledge Modules in ODI

Für spezifische Projektbedürfnisse lassen sich auch ganz neue Knowledge Modules entwickeln oder vorhandene einfach erweitern.



Abbildung 2: Auszug aus der Liste der mitinstallierten, frei verfügbaren Knowledge Modules

Ein Praxisbeispiel: Versionierung der Stammdaten im DWH

Ziel dieses Artikels ist, einerseits aufzuzeigen, wie einfach sich im ODI ein bestehendes Knowledge Module erweitern lässt, andererseits aber auch auf die Gefahren hinzuweisen, falls man mit der Annahme arbeitet, dass die mitgelieferten kostenfreien Knowledge Modules „out of the box“ bereits eine Lösung darstellen. Dazu ein Beispiel, das in der DWH-Praxis sehr verbreitet ist: die Problemstellung der Versionierung.

Bei der Versionierung geht es darum, die Änderungen seit dem letzten Datenabgleich zu erkennen und dieses Delta in geeigneter Form in das Core Data Warehouse abzulegen. Abbildung 3 stellt diesen Abgleichprozess grafisch dar.

Die Versionierung wird nachfolgend anhand einiger Beispieldaten erklärt. Dazu existiert in der Staging Area eine

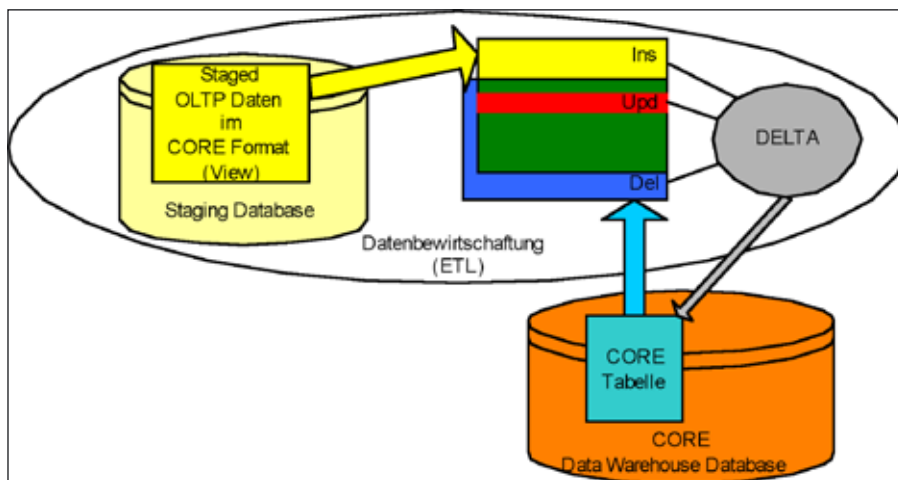


Abbildung 3: Prinzip der Versionierung der Stammdaten

Tabelle SA_CUSTOMER. Struktur und Inhalt sind in Abbildung 4 dargestellt.

Diese Staging-Tabelle enthält einen Abzug (Snapshot) der Daten aus einem OLTP-System für einen bestimmten Zeitpunkt (zum Beispiel den 31.01.2009). Die Spalte CUST_ID bildet den Primärschlüssel dieser Tabelle.

Im Beispiel gibt es im CORE Data Warehouse eine Tabelle C_CUSTOMER. Abbildung 5 zeigt ihre Struktur und ihren Inhalt vor dem Versionierungslauf.

Die Spalte DWH_ID enthält den Surrogate Key, der zugleich auch den Primärschlüssel der Tabelle darstellt. In der Tabelle C_CUSTOMER stehen alle historischen Veränderungen des Attributs STATUS. Das Attribut STATUS ist ein versionierungsrelevantes Attribut (SCD2-Attribut). Die technischen Attribute ValidFrom und ValidTo schränken die Gültigkeit der einzelnen Versionen ein (der spezielle Wert 01.01.2400 bedeutet, dass die Version unbegrenzt gültig ist). Das Attribut NAME ist kein versionierungsrelevantes Attribut (SCD1-Attribut). Das Attribut CUST_ID ist ein Natural Key. Kombiniert mit einer Zeiteinschränkung ermöglicht es die eindeutige Identifikation einer bestimmten Kundenversion. Abbildung 6 zeigt den Inhalt der Tabelle C_CUSTOMER nach dem Versionierungslauf. Folgende Änderungen sind erkennbar:

- Die Version mit DWH_ID=-1000 wurde abgeschlossen (ValidTo auf 31.01.2009 gesetzt), denn der Eintrag mit CUST_ID=4710 kommt in der SA-Tabelle nicht mehr vor (wurde im OLTP System wahrscheinlich gelöscht).

Table SA_CUSTOMER		
CUST_ID	NAME	STATUS
4711	John_4711	active
4712	Thomas_4712	suspended
4713	Lisa_4713	active
4714	Simon_4714	inactive
4715	Robert_4715	inactive

Abbildung 4: Struktur und Inhalt der Staging-Tabelle SA_CUSTOMER

C_CUSTOMER					
DWH_ID	CUST_ID	ValidFrom	ValidTo	NAME	STATUS
-1000	4710	05.01.2009	01.01.2400	Bill_4710	suspended
-1001	4711	01.01.2009	01.01.2400	John_4711	active
-1002	4712	01.01.2009	10.01.2009	ThoSmas_4712	active
-1003	4712	10.01.2009	01.01.2400	ThoSmas_4712	suspended
-1004	4714	10.01.2009	01.01.2400	SiQmon_4714	active
-1005	4715	08.01.2009	01.01.2400	Robert_4715	active

Abbildung 5: Struktur und Inhalt der CORE DWH Tabelle C_CUSTOMER vor dem Versionierungslauf

C_CUSTOMER					
DWH_ID	CUST_ID	ValidFrom	ValidTo	NAME	STATUS
-1000	4710	05.01.2009	31.01.2009	Bill_4710	suspended
-1006	4713	31.01.2009	01.01.2400	Lisa_4713	active
-1001	4711	01.01.2009	01.01.2400	John_4711	active
-1002	4712	01.01.2009	10.01.2009	Thomas_4712	active
-1003	4712	10.01.2009	01.01.2400	Thomas_4712	suspended
-1004	4714	10.01.2009	31.01.2009	Simon_4714	active
-1007	4714	31.01.2009	01.01.2400	Simon_4714	inactive
-1005	4715	08.01.2009	31.01.2009	Robert_4715	active
-1008	4715	31.01.2009	01.01.2400	Robert_4715	inactive

Abbildung 6: Inhalt der CORE DWH Tabelle nach dem Versionierungslauf

- Die Version mit DWH_ID=-1006 wurde neu hinzugefügt, denn es gab in der CORE Tabelle noch keine Version für CUST_ID = 4713.
- Die Version mit DWH_ID=-1004 wurde abgeschlossen (ValidTo auf 31.01.2009 gesetzt), denn beim CUST_ID = 4714 hat sich der STATUS geändert (vom „active“ zu „inactive“) und diese Änderung führt zur Erstellung einer neuen Version (mit DWH_ID=-1007).
- Das Gleiche geschah auch beim Kunden mit CUST_ID = 4715. Dort hat sich der STATUS geändert (vom „active“ auf „inactive“). Die Version mit DWH_ID=-1005 wurde abgeschlossen (ValidTo auf 31.01.2009 gesetzt) und eine neue Version (mit DWH_ID=-1008) erstellt.
- Bei dem Kunden mit CUST_ID = 4712 hat sich der Name geändert (vom „Thomas_4712“ auf „Thomas_4712“). Die Änderung an diesem Attribut ist jedoch nicht versionierungsrelevant, es wurde also keine neue Version gebildet. Alle bestehenden Versionen werden mit dem neuen Wert angepasst.
- Das Gleiche geschah auch beim Kunden mit CUST_ID = 4714. Dort hat sich der Name geändert (vom „Simon_4714“ auf „Simon_4714“).

Mögliche Umsetzungen der Versionierung

Es gibt mehrere Wege, wie die Versionierung umgesetzt werden kann. Die Umsetzungen unterscheiden sich

meistens darin, wie gut sie folgende Ansprüche erfüllen:

- Die Umsetzung implementiert die Zielsetzung korrekt, das heißt, sie enthält keine Denkfehler
 - Sie ist robust genug, so dass auch Randfälle und komplexe Fälle richtig behandelt werden
- Sie ist in der gegebenen Technologie (zum Beispiel Oracle Datenbank 11g) ablauffähig, das heißt, sie enthält keine Syntaxfehler.
- Sie berücksichtigt die Gegebenheiten der Technologie (Stärken, Limitierungen, Bugs etc.)
 - Sie ist effizient im Sinne ihres Verwendungszwecks (Durchsatz bei Mengenverarbeitung; Latenzzeit bei Einzelverarbeitung)
 - Sie führt keinen künstlichen Flaschenhals in die Verarbeitung ein, der Skalierbarkeit blockieren würde

In diesem Artikel werden zwei Umsetzungen der Versionierung in der Zieltechnologie Oracle 11g vorgestellt:

- Die Umsetzung anhand des Standard Knowledge Modules „IKM Oracle Slowly Changing Dimension“, das von Oracle mit der ODI-Standard-Installation mitgeliefert wird
- Best Practice des Autors

Aus Gründen der besseren Verständlichkeit zunächst die Best Practice des Autors. Listing 1 und 2 zeigen die Um-

setzung der Versionierung in der Zieltechnologie Oracle11g:

- Sie verwendet den FULL OUTER JOIN, um das Delta zu finden
- Sie minimiert die Anzahl der Zugriffe auf die Tabellen
- Alle Operationen der generierten SQL-Befehle sind voll parallelisierbar
- Die Schreib-Operationen mit temporärem Inhalt verursachen nur minimales Undo- und Redo-Aufkommen (I/O-Operationen)

In Listing 1 wird das Delta ermittelt und in einer temporären Tabelle festgehalten, weil auf diese Resultatmenge mehrmals zugegriffen werden muss. Die Flags (Attribute der Delta-Tabelle) legen fest, um welche Art der Änderung es sich handelte (DML_I, DML_D, DML_U_SCD2, DML_U_SCD1). Anschließend werden entsprechende DML-Befehle gegen die CORE-Tabelle abgesetzt (siehe Listing 2, Seite 50).

```
-- Step 1
-- insert changes (Delta) into
I$(FULLOUTERJ)

insert /*+ APPEND NOLOGGING */
into TTT_DWH_ETL_RUN.I$_C_CUSTOMER
(
    DWH_ID ,
    CUST_ID ,
    NAME ,
    STATUS ,
    DML_D ,
    DML_I ,
    DML_U_SCD1 ,
    DML_U_SCD2
)
select
    DWH_ID,
    CASE WHEN DML_D=1 THEN
    OLD_CUST_ID ELSE CUST_ID
    END AS CUST_ID,
    CASE WHEN DML_D=1 THEN
    OLD_NAME ELSE NAME END AS
    NAME,
    CASE WHEN DML_D=1 THEN
    OLD_STATUS ELSE STATUS END
    AS STATUS,
    DML_D ,
    DML_I ,
    DML_U_SCD1 ,
    DML_U_SCD2
from ( select
    C.DWH_ID,
```

```

SA.CUST_ID,
SA.NAME,
SA.STATUS,
C.CUST_ID AS OLD_CUST_ID,
C.NAME AS OLD_NAME,
C.STATUS AS OLD_STATUS
, CASE WHEN SA.CUST_ID
IS NULL THEN 1 ELSE 0
END AS DML_D
, CASE WHEN C.CUST_ID
IS NULL THEN 1 ELSE 0
END AS DML_I
, CASE WHEN C.CUST_ID
IS NOT NULL
AND SA.CUST_
ID IS NOT NULL
AND ( C.NAME
<> SA.NAME )
THEN 1 ELSE 0 END AS
DML_U_SCD1
, CASE WHEN C.CUST_ID
IS NOT NULL
AND SA.CUST_ID
IS NOT NULL
AND ( C.STATUS
<> SA.STATUS )
THEN 1 ELSE 0 END
AS DML_U_SCD2
from (SELECT
SA_CUSTOMER.CUST_ID AS
CUST_ID,
SA_CUSTOMER.NAME AS NAME,
SA_CUSTOMER.STATUS AS
STATUS
from TTT_DWH_SA.SA_
CUSTOMER SA_CUSTOMER
) SA
full outer join
(SELECT *
from TTT_DWH.C_CUSTOMER
where DWH_VALID_TO =
to_date ('01-01-2400',
'mm-dd-yyyy')
) C
on
SA.CUST_ID = C.CUST_ID
)
WHERE DML_I+DML_U_SCD1+DML_U_
SCD2+DML_D > 0
;

```

Listing 1: Abgleich und Erkennen der Differenzen zwischen der Stage- und der CORE-Tabelle

In nächsten Abschnitt wird die Umsetzung der Versionierung in ODI gezeigt. Für die Generierung kommt das frei mitgelieferte Knowledge Module „IKM Oracle Slowly Changing Dimension“ zum Einsatz.

Als erstes definiert man im ODI Designer Tool das sogenannte „SCD-Verhalten“: Für jede Spalte der Tabelle

```

-- Step 2
-- update target with SCD1
changes
MERGE
INTO -- target table
TTT_DWH.C_CUSTOMER T
USING
(SELECT * FROM TTT_DWH_ETL_
RUN.I$_C_CUSTOMER WHERE
DML_U_SCD1 = 1) D
ON (T.CUST_ID = D.CUST_ID )
WHEN MATCHED THEN
UPDATE
SET T.NAME = D.NAME
;

-- Step 3
-- close versions of SCD2
MERGE
INTO -- target table
TTT_DWH.C_CUSTOMER T
USING
(SELECT * FROM TTT_DWH_ETL_
RUN.I$_C_CUSTOMER WHERE
DML_U_SCD2 = 1 OR DML_D = 1)D
ON (T.DWH_ID = D.DWH_ID )
WHEN MATCHED THEN
UPDATE
SET T.DWH_VALID_TO = to_
date(<31-01-2009>, <dd-mm-yyyy>)

/* expression for logical_snap-
shot_date */ ;

-- Step 4
-- insert new rows (SCD1+SCD2)
INSERT INTO TTT_DWH.C_CUSTOMER
(
DWH_ID ,
CUST_ID ,
NAME ,
STATUS ,
DWH_VALID_FROM ,
DWH_VALID_TO ,
DWH_AUDIT_ID
)
SELECT
TTT_DWH.C_CUSTOMER_S.nextval,
CUST_ID ,
NAME ,
STATUS ,
to_date(<31-01-2009>, <dd-
mm-yyyy>) /* expression
for logical_snapshot_date */,
to_date (<01-01-2400>, <mm-
dd-yyyy>),
1
FROM TTT_DWH_ETL_RUN.I$_C_CUS-
TOMER D
WHERE DML_U_SCD2 = 1 OR DML_I
= 1
;

```

Listing 2: Durchführung der DML-Befehle, um die Differenzen zwischen Stage- und CORE-Tabellen zu beseitigen

C_CUSTOMER wird definiert, wie sich die Spalte in Bezug auf Versionierung verhält bzw. welche Rolle sie hier übernimmt. Das SCD-Verhalten für Spalten der Beispieltabelle C_CUSTOMER wurde entsprechend der Ausführungen zuvor konfiguriert (siehe Abbildung 7).

Column	SCD Behavior
DWH_ID	Surrogate Key
CUST_ID	Natural Key
NAME	Overwrite on Change
STATUS	Add Row on Change
DWH_VALID_FROM	Starting Timestamp
DWH_VALID_TO	Ending Timestamp

Abbildung 7: Im ODI wurde den Spalten der Tabelle C_CUSTOMER das entsprechende SCD-Verhalten zugeordnet

Diese Angaben werden im Repository von ODI gespeichert. Das Knowledge Module, das für die Generierung des Quellcodes zuständig ist, kann diese Angaben aus dem Repository herauslesen und bei der Generierung berücksichtigen.

In ODI erstellt man dann ein einfaches Interface (Synonym für „Mapping“ in anderen Umgebungen wie Oracle Warehouse Builder oder Informatica’s PowerCenter), das die Daten aus der Tabelle SA_CUSTOMER herausliest, mit dem Inhalt in C_CUSTOMER abgleicht und entsprechend den erkannten Differenzen anpasst.

Die Transformationen des Abgleichs und der Versionierung (zum Beispiel der Lese-Zugriff auf die CORE-Tabelle und der Vergleich der Spaltenwerte) werden nicht direkt im ODI-Interface gezeichnet. Da dies in jedem ODI-Interface für Versionierung gleich (generisch) gehalten werden soll, werden diese im Knowledge Module definiert. Im ODI-Interface erfolgt nur die Zuordnung des entsprechenden Knowledge Modules, das die Versionierung implementiert – also eine indirekte Definition. Das ODI-Interface ist in diesem Fall also trivial (siehe Abbildung 8).

Wie bereits erwähnt, erfolgt im ODI-Interface die Zuordnung des Knowledge Modules, das die Versionierung implementiert. Man ordnet hier das

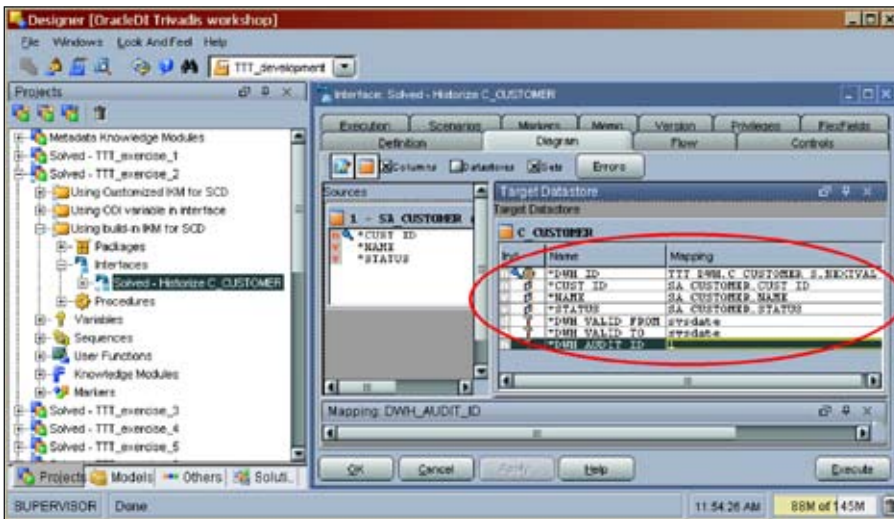


Abbildung 8: Die Versionierung wird im ODI mit einem simplen Interface definiert

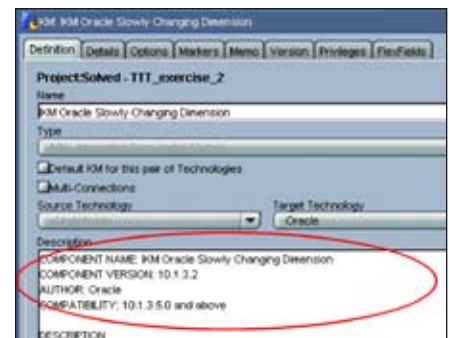


Abbildung 9: Zum Zeitpunkt der Erstellung dieses Artikels war von diesem Knowledge Module die Version 10.1.3.2 verfügbar

von Oracle kostenfrei zur Verfügung gestellte Knowledge Module „IKM Oracle Slowly Changing Dimension“ zu. Anschließend kann man die Qualität der Umsetzung durch dieses Knowledge Module untersuchen. Dazu wird

dessen aktuellste Version verwendet (siehe Abbildung 9).

Ein näherer Blick auf die Struktur: Die Schritte, aus welchen dieses Knowledge Module besteht, sind in Abbildung 10 dargestellt. In den rot

markierten Schritten finden die eigentlichen Datentransformationen der Versionierung statt.

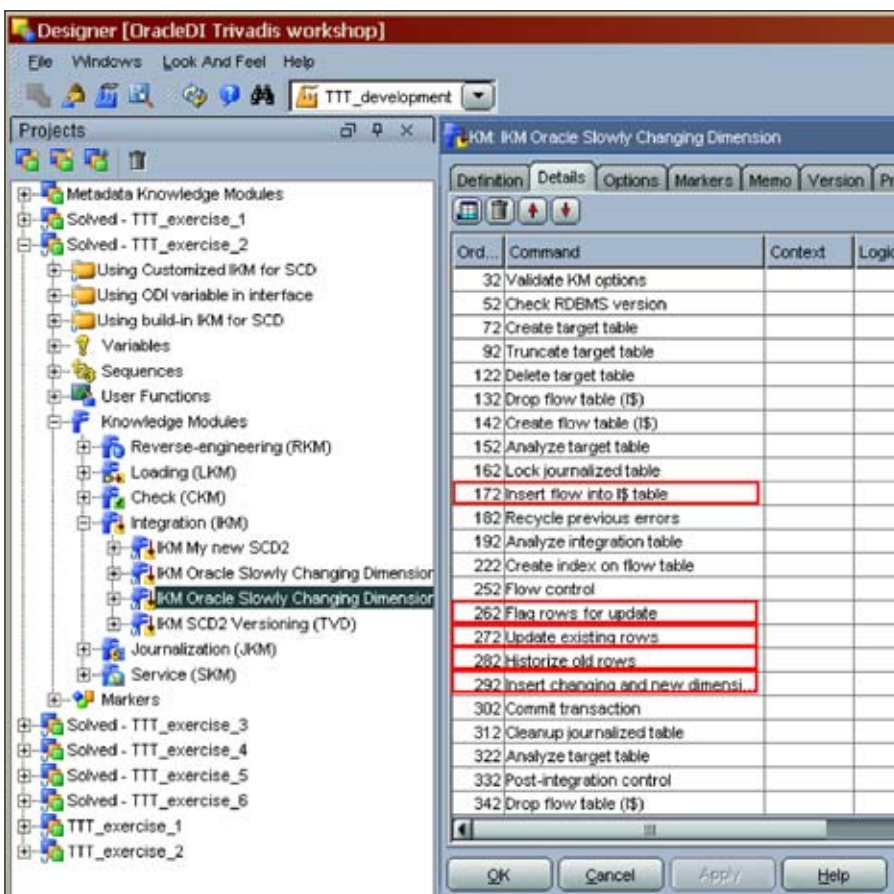


Abbildung 10: IKM Oracle Slowly Changing Dimensions besteht aus den hier erkennbaren Schritten (Commands)

```

insert /*+ APPEND */ into TTT_
DWH_ETL_RUN.I$_C_CUSTOMER
(
    CUST_ID,
    NAME,
    STATUS,
    DWH_VALID_FROM,
    IND_UPDATE
)
select
    SA_CUSTOMER.CUST_ID,
    SA_CUSTOMER.NAME,
    SA_CUSTOMER.STATUS,
    sysdate,
    'I' IND_UPDATE
from TTT_DWH_SA.SA_CUSTOMER
SA_CUSTOMER
where (I=1)
and NOT EXISTS (
    select 'x'
    from TTT_DWH.C_CUSTO
    M E R T
    where T.CUST_ID      =
    CUST_ID
    and ((SA_CUSTOMER.
    NAME = T.NAME) or (SA_CU
    STOM E R.NAME IS NULL and
    T.NAME IS NULL))
    and DWH_VALID_TO    =
    to_date ('01-01-2400',
    'mm-dd-yyyy')
)
    
```

Listing 3: Quellcode, welcher aus dem Schritt „insert flow into I\$table“ generiert wurde

```

update      TTT_DWH_ETL_RUN.
I$_C_CUSTOMER S
set        S.IND_UPDATE = <U>
where     (S.CUST_ID)
          in
            (
              select  T.CUST_ID
              from    TTT_DWH.C_
CUSTOMER T
              where  ((S.STATUS
= T.STATUS) or ((S.STATUS
is null) and (T.STATUS is
null)))
              and   DWH_VALID_TO = to_date
(<01-01-2400>, <mm-dd-yyyy>)
            )

```

Listing 4: Quellcode, welcher aus dem Schritt „Flag rows for update“ generiert wurde

```

update TTT_DWH.C_CUSTOMER T
set
(
  T.NAME
) =
(
  select X.NAME
  from   TTT_DWH_ETL_RUN.
I$_C_CUSTOMER X
  where  X.CUST_ID   =
T.CUST_ID
  and    X.IND_UPDATE =
<U>
)
where (T.CUST_ID)
      in(
        select S.CUST_ID
        from   TTT_DWH_ETL_RUN.
I$_C_CUSTOMER S
        where  S.IND_UPDATE =
<U>
        )
      and DWH_VALID_TO = to_date
(<01-01-2400>, <mm-dd-yyyy>)

```

Listing 5: Quellcode, welcher aus dem Schritt „Update existing rows“ generiert wurde

```

update TTT_DWH.C_CUSTOMER T
set
(
  T.DWH_VALID_TO
) =
(
  select
    S.DWH_VALID_FROM
  from   TTT_DWH_ETL_RUN.
I$_C_CUSTOMER S
  where  S.CUST_ID   =
T.CUST_ID
  and    S.IND_UPDATE =
<I>
)
where (T.CUST_ID)
      in
        (
          select X.CUST_ID
          from   TTT_DWH_ETL_RUN.
I$_C_CUSTOMER X

```

```

          where X.IND_UPDATE =
<I>
          )
          and DWH_VALID_TO = to_date
(<01-01-2400>, <mm-dd-yyyy>)

```

Listing 6: Quellcode, welcher aus dem Schritt „Historize existing rows“ generiert wurde

```

insert into TTT_DWH.C_CUSTOMER
(
  CUST_ID,
  NAME,
  STATUS
  ,DWH_ID
  ,DWH_VALID_FROM
  ,DWH_VALID_TO
)
select CUST_ID,
NAME,
STATUS
,TTT_DWH.C_CUSTOMER_S.
NEXTVAL
,DWH_VALID_FROM
,to_date (<01-01-2400>,
<mm-dd-yyyy>)
from TTT_DWH_ETL_RUN.I$_C_CU-
STOMER S
where S.IND_UPDATE = 'I'

```

Listing 7: Quellcode, welcher aus dem Schritt „Insert changing and new dimensions“ generiert wurde

Abbildung 11 (siehe Seite 53) zeigt das Resultat, das ein Ladelauf dieser Umsetzung produziert. Die rot markierten Stellen zeigen, wo die Umsetzung fehlerhaft ist:

- **Befund 1**
In Fällen, in denen sich der STATUS geändert hat (wie etwa bei CUST_ID=4715), erzeugt dieser Quellcode keine neue Version (fehlende Version ist in Abbildung 12 kursiv markiert): Denkfehler im Listing 3 (Schritt „insert flow into I\$table“).
- **Befund 2**
Falls sich das Attribut NAME (SCD1=overwriting history) ändert, wird nur die letzte Version geändert, nicht aber die volle Historie (CUST_ID=4712): Denkfehler im Listing 5 (Schritt „update existing rows“).
- **Befund 3**
Falls sich der STATUS (SCD2) und gleichzeitig auch der NAME (SCD1) ändert (CUST_ID=4714), wird die

Änderung am Attribut NAME in vorherigen Versionen nicht übertragen: Denkfehler im Listing 4 (Schritt „Flag rows for update“).

- **Befund 4**

Falls ein Eintrag in der SA-Tabelle nicht mehr vorkommt, führt dies nicht zum Abschließen der aktuell gültigen Version (CUST_ID=4710). Die Tatsache, dass diese Funktionalität fehlt, ist in dem Knowledge Module nicht dokumentiert.

Ist diese Umsetzung der Versionierung effizient und skalierbar?

In größeren DWH Umgebungen ist es üblich, dass einige der Dimensionstabellen recht viele Einträge enthalten (mehrere Millionen Datensätze). Man stelle sich vor, die Beispieltabelle C_CUSTOMER wäre so eine. In der aufgeführten Umsetzung wird auf die Tabelle C_CUSTOMER fünfmal zugegriffen (zweimal lesend, zweimal mit Update, einmal mit Insert). Dies lässt sich definitiv effizienter gestalten. Wer deshalb davon ausgeht, dass das von Oracle mitgelieferte „IKM Oracle Slowly Changing Dimension“ eine korrekte und effiziente Lösung für die Versionierung darstellt, kann böse Überraschungen erleben. Generell ist sehr zu empfehlen, die Qualität der später zum Einsatz kommenden Knowledge Modules ziemlich genau unter die Lupe zu nehmen, etwa in Form vom „Proof of Concept“ und anhand repräsentativer Use Cases. Wenn man mit der Umsetzung der üblichen Fragestellungen bereits eigene Best Practice zur Verfügung hat, ist man eventuell schneller am Ziel, wenn man eigene Knowledge Modules selber entwickelt.

Eigene Knowledge Modules entwickeln

Wie das aufgeführte Beispiel gezeigt hat, kommt man beim ODI-Einsatz nicht darum herum, eigene Knowledge Modules zu entwickeln. Und hier kommt die gute Nachricht: das Entwickeln ist gar nicht so schwierig. Dazu wieder ein Beispiel, bei dem die vorgestellte Best Practice des Autors für die Umsetzung der Versionierung in ein neues Knowledge Module übertragen wird.

Dazu legt man ein neues Knowledge Module an und nennt es „IKM SCD Versioning (TVD)“. Dann legen wir die einzelnen Schritte (Commands) an (siehe Abbildung 12). Die Definition der unterstützenden Schritte übernimmt man einfach von dem Standard-IKM, man ändert nur das Wesentliche, nämlich die Schritte, die den Quellcode für die Datentransformationen generieren. Diese sind in Abbildung 12 rot markiert.

Um das Prinzip der Entwicklung von Commands der Knowledge Modules zu verstehen, betrachtet man den Schritt „close versions of SCD2“: In Listing 8 ist der Befehl für diesen Schritt zu finden. Er generiert das zweite MERGE-Statement in Listing 2, das für das Schließen der Versionen zuständig ist.

```

MERGE
INTO -- target table
  <%=snpRef.getTable("L","TARG_
  NAME","A")%> T
USING
  (SELECT * FROM <%=snpRef.
  getTable("L","INT_
  NAME","A")%> WHERE DML_U_
  SCD2 = 1 OR DML_D = 1) D
ON (<%=snpRef.getColList("",
  "T.[COL_NAME] = D.[COL_NAME]
  ", " and \n\t", "", "SCD_
  SK")%>)
WHEN MATCHED THEN
  UPDATE
    SET      T.DWH_VALID_TO =
  <%=snpRef.getColList("", "[EX-
  PRESSION]", ",\n\t", "", "SCD_
  START")%>
    
```

Listing 8: Command für den Schritt „close versions of SCD2“, das den Quellcode des zweiten MERGE-Statements in Listing 2 generiert

Die Platzhalter <%...%> werden zum Generierungszeitpunkt ausgewertet. Sie entsprechen den Aufrufen von dokumentierten Java-Methoden. Diese Methoden lesen die Attributwerte des jeweiligen zu generierenden ODI Interfaces (Mappings) aus dem ODI-Repository heraus und ermöglichen, dass der Generierungsprozess diese Werte berücksichtigt. Wer diese dokumentierten Methoden-Aufrufe beherrscht, der kann eigene Knowledge Modules selbst schreiben.

C_CUSTOMER					
DWH_ID	CUST_ID	ValidFrom	ValidTo	NAME	STATUS
-1000	4710	05.01.2009	01.01.2400	Bill_4710	suspended
-1006	4713	31.01.2009	01.01.2400	Lisa_4713	active
-1001	4711	01.01.2009	01.01.2400	John_4711	active
-1002	4712	01.01.2009	10.01.2009	ThoSmas_4712	active
-1003	4712	10.01.2009	01.01.2400	Thomas_4712	suspended
-1004	4714	10.01.2009	01.01.2400	SiQmon_4714	active
-1007	4714	31.01.2009	01.01.2400	Simon_4714	inactive
-1005	4715	10.01.2009	01.01.2400	Robert_4715	active
	4715	31.01.2009	01.01.2400	Robert_4715	inactive

Abbildung 11 : Inhalt der Tabelle C_CUSTOMER nach dem Ladelauf der Versionierung mit „IKM Oracle Slowly Changing Dimension“

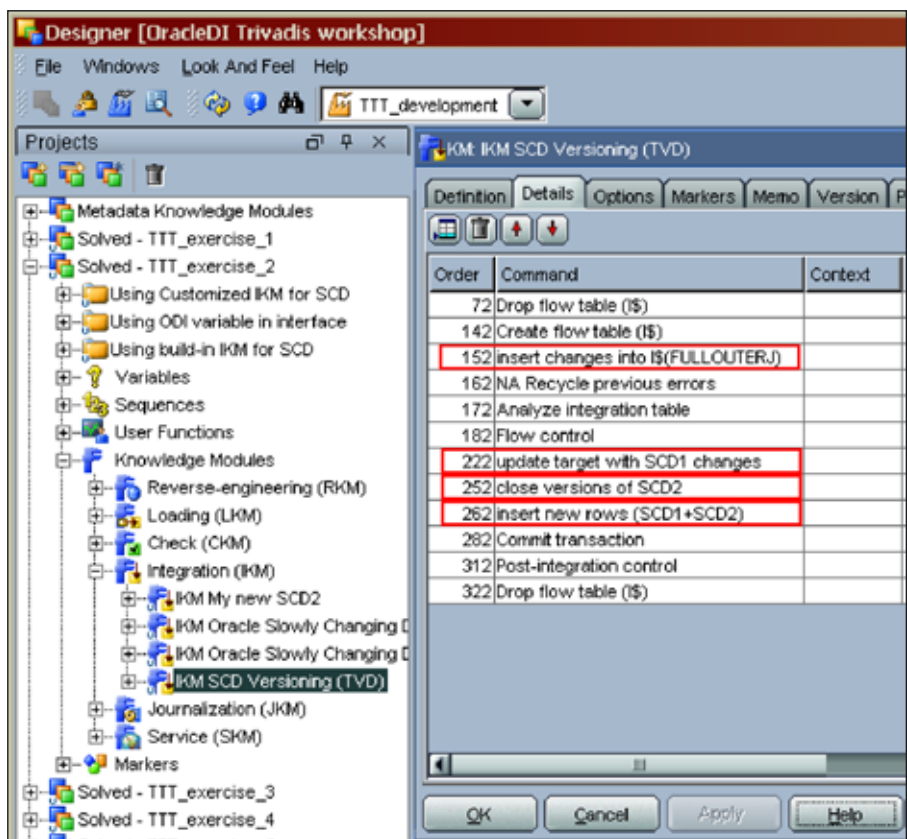


Abbildung 12: Struktur des neu angelegten Knowledge Modules „IKM SCD Versioning (TVD)“

Fazit

Das Know-how, um Knowledge Modules zu entwickeln, ist relativ simpel. Das Know-how darüber, was man mit den Knowledge Modulen erzielen soll, also wie eine korrekte und performante Umsetzung aussehen soll, war und bleibt auch weiterhin komplex. Es braucht vor allem mehrjährige prakti-

sche Erfahrung in der Umsetzung von DWH-Projekten. Die mitgelieferten Knowledge Modules von Oracle bieten eine Basis für die Umsetzung, jedoch keine Lösung.

Kontakt:

Karol Hajdu
karol.hajdu@trivadis.com



2009
DOAG
Konferenz + Ausstellung

IBM

HMK

SAP

KTCG

pdg
PRIMUS DELPHI GROUP

trivadis
makes it easier.

syncsort
PERFORMANCE DRIVEN

Best Solutions Based on Oracle
HUNKLER
GmbH & Co. KG

red database
security

XDEV
XDEV Software Corp.

symantec.

up to data
professional services

Libelle

EMC²
where information lives!

People at Work
Systems

FUJITSU

T24M

metafinanz

WOLF CONSULT
IT-PROJEKTE GMBH

GDV

**Wir führen
die Oracle-Community
zum Erfolg**

sgi

itspeople

ORDIX AG

KUTTIG

CSP
professional solutions

HE DOAG
HOCHSCHULE-KOMMUNITY

sentrigo

digin

ALTRAN
CIS

MicroStrategy

PROMATIS

precise

imining

COMIT

OPITZ CONSULTING

QUEST SOFTWARE

D&I

CipherSoft Inc.
A Subsidiary of Unif's Corporation

Sun
microsystems
The Network is the Computer™

ORACLE

www.doag2009.org

DOAG 2009 Konferenz + Ausstellung
17. – 19.11.2009 in Nürnberg

Effiziente Historisierung großer Tabellen

Tobias Stark, metafinanz GmbH

Bei der nachträglichen Historisierung von Daten in großen Tabellen ist es von Anfang an wichtig, auf die Performance zu achten. Je mehr Datensätze anzupassen sind, desto entscheidender wird die Wahl der technischen Vorgehensweise.

Gerade DML-Operationen wie Updates führen bei großen Datenmengen häufig zu Problemen. Um auch in solchen Situationen akzeptable Verarbeitungsgeschwindigkeiten zu erreichen, stellt die Oracle Datenbank Möglichkeiten wie PL/SQL-Tabellen, BULK-Verarbeitung und Parallel DML zur Verfügung.

Zur Historisierung von Daten in einer Basis-Datenbank wurde von einem Kunden des Autors eine Lösung in PL/SQL entwickelt. Diese PL/SQL-Anwendung hatte dabei zu jedem Datensatz die Gültigkeit bei neuen Datensätzen einzutragen beziehungsweise bei älteren Datensätzen zu aktualisieren, und zwar nach Art einer „Slowly Changing Dimension (SCD) Typ 2“ in „gültig von/gültig bis“-Attribute. Durch dieses Vorgehen lag die Anzahl der erwarteten, anzupassenden Datensätze bei Delta-Ladungen für einige Tabellen täglich im Millionen-Bereich. Um diese Mengen in annehmbarer Zeit zu verarbeiten, wurden mehrere Prozesse parallel per DBMS_JOB gestartet. Ein Master-Prozess sammelte die zu bearbeitenden fachlichen Schlüssel in einer temporären Tabelle und übergab diese dann per Pipes an die parallel gestarteten Jobs als Slave-Prozesse. Die Aufgabe eines solchen Slave-Prozesses bestand darin, zu jedem Datensatz des übergebenen Schlüssels die Gültigkeit zu ermitteln und diese anschließend in die Historisierungs-Attribute zu schreiben.

Dabei wurden die zu dem Schlüssel vorhandenen Daten nebst ROWID per Indexzugriff in eine PL/SQL-Tabelle eingelesen. Danach erfolgte die Ermittlung der „gültig von/gültig bis“-Daten in der PL/SQL-Tabelle im Speicher. Zum Schluss wurden die Daten per Einzelsatzverarbeitung anhand der ROWID in die Tabelle geschrieben. Neben der Gültigkeit waren für die Identifizierung der geänderten Daten

durch Folgesysteme auch noch weitere Attribute wie eine Verarbeitungsnummer anzupassen. Durch die Parallelisierung waren bei Tests mit kleineren Datenmengen in der Entwicklungsumgebung gute Ergebnisse zu erzielen. Aufgrund der Testergebnisse kam diese Lösung in der Produktionsumgebung zum Einsatz.

Nach einer Upladung mit rund 520 Millionen Datensätzen in der Produktivumgebung und einem täglichen Delta von bis zu 21 Millionen Datensätzen kam es jedoch bei mehreren Tabellen zu Performance-Problemen. Damit die Belastung des Datenbank-Servers durch die parallelen Prozesse

nicht zu groß wurde, fiel die Entscheidung, die Historisierung unabhängig von der Ladung immer nur für eine Tabelle durchzuführen. Dabei traten jedoch trotz der Parallelisierung Laufzeiten von mehr als acht Stunden für eine Tabelle auf.

Problem-Analyse

Um die Ursache für die Performance-Probleme zu finden, wurden mehrere Analysen durchgeführt. Zum einen erfolgte während der Historisierung einer Tabelle mit sechs Prozessen eine Auswertung der auftretenden Wait-Events (siehe Tabelle 1).

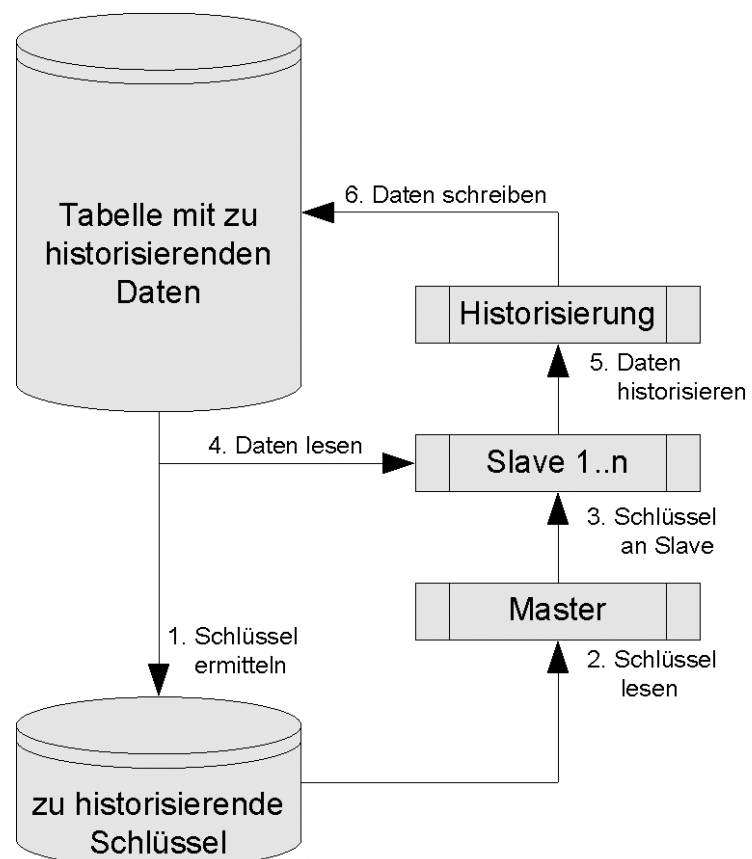


Abbildung 1: Ablauf ursprüngliche Historisierung

Wait-Event	Waits gesamt
db file sequential read	5.637.171
latch: shared pool	463.396
latch: library cache	9.303
latch: row cache objects	3.679
latch: library cache lock	73

Tabelle 1: Wait-Events bei Historisierung mit sechs Prozessen

Zudem erfolgte ein Review des PL/SQL-Codes. Dabei wurde festgestellt, dass bei den dynamischen SQL-Anweisungen Literale wie die ROWID zum Einsatz kamen. Dies deckte sich auch mit den beobachteten Latch-Waits, die ein Hinweis auf das häufige Parsen von SQL-Anweisungen sind.

Auch beim Layout der Tabellen traten einige problematische Punkte auf. Durch das Lesen anhand des fachlichen Schlüssels kommt es gerade bei großen Tabellen beim Index-Zugriff zu längeren Laufzeiten, insbesondere dann, wenn die zu lesenden Datensätze über unterschiedliche Datenbankblöcke verstreut sind. Darauf weist auch die Zahl der beobachteten „db file sequential read“ hin.

Neben dem Zugriff beim Lesen über den Index gab es die Problematik, dass eins der historischen Attribute als Kriterium für die Partitionierung zum Einsatz kam. Dadurch entstand beim Schreiben der Daten zusätzlich ein Verschieben von Datensätzen von einer Partition in eine andere. Zusätzlich war auch noch ein Index auf der Verarbeitungsnummer der geänderten Daten zu pflegen.

Identifizierte Verbesserungsmöglichkeiten

Um das Performanceproblem in den Griff zu bekommen, wurden die folgenden Verbesserungsmöglichkeiten identifiziert:

- *Entfernung des Index auf der Verarbeitungsnummer.*
Da sich der Index für die lesenden DWH-Systeme als nicht hilfreich erwiesen hatte, wurde er entfernt. Dadurch war eine Pflege dieses Index bei der Historisierung nicht mehr notwendig.

- *Anpassung der Partitionierung*
Das historische Attribut wurde aus den Kriterien für die Partitionierung herausgenommen und die Tabellen neu partitioniert. Dadurch war ein Verschieben von Datensätzen zwischen Partitionen nicht mehr notwendig.
- *Nutzung von BIND-Variablen statt Literalen*
Bei der Ausführung der dynamischen SQL-Anweisungen wurden statt der vorher genutzten Literale die Daten per BIND-Variablen übergeben. Darüber hinaus ergab sich, dass der Zusammenbau der dynamischen SQL-Anweisungen nur noch einmal zu Anfang der Verarbeitung durchgeführt werden musste. Die Ermittlung bei jedem zu bearbeitenden Datensatz konnte dadurch entfallen und die CPU-Belastung sank.
- *Vermeidung von Index-Zugriffen*
Das Sammeln der zu bearbeitenden Datensätze in einer temporären Tabelle vermied den Indexzugriff über den fachlichen Schlüssel.
- *Nutzung von BULK-Verarbeitung*
Die zu schreibenden Historisierungsdaten werden in PL/SQL-Tabellen zwischengespeichert und per BULK-Insert in eine Tabelle für die anschließende Nutzung mittels Parallel DML geschrieben. Durch die BULK-Verarbeitung lassen sich die bei Einzelsatzverarbeitung auftretenden Wechsel zwischen PL/SQL- und SQL-Engine minimieren.

- *Einsatz von Parallel DML*
Durch den Einsatz von Parallel DML beim Anpassen der Daten in den Tabellen war es möglich, die Historisierung direkt im Anschluss an das Laden der Deltadaten durchzuführen. So war keine asynchrone, serielle Verarbeitung der Tabellen mehr notwendig.

Aufgrund der Komplexität und um nicht sämtliche Datensätze nochmals verarbeiten zu müssen, hat man den Kernbestandteil zur Ermittlung der Historisierung nicht geändert. Bevor das Anpassen der Daten per Parallel DML implementiert wurde, kam in einer Zwischenversion ein Update per BULK zum Einsatz. Diese Zwischenversion erwies sich auch für nicht partitionierte Tabellen als erfolgreich. Bei partitionierten Tabellen jedoch war die Laufzeit für den Update von 5,4 Millionen Datensätzen mit 9,5 Stunden weiterhin zu hoch. Die Statistiken der oben genannten Latches ließ sich durch die vorgenommenen Anpassungen jedoch bereits signifikant verbessern, so dass eine Entlastung der Ressourcen feststellbar war (siehe Tabelle 2).

Einsatz von Parallel DML

Um auch für partitionierte Tabellen eine Verbesserung der Laufzeiten zu erreichen, wurde die Nutzung von Parallel DML implementiert. Für dessen Einsatz sind folgende Bedingungen zu beachten:

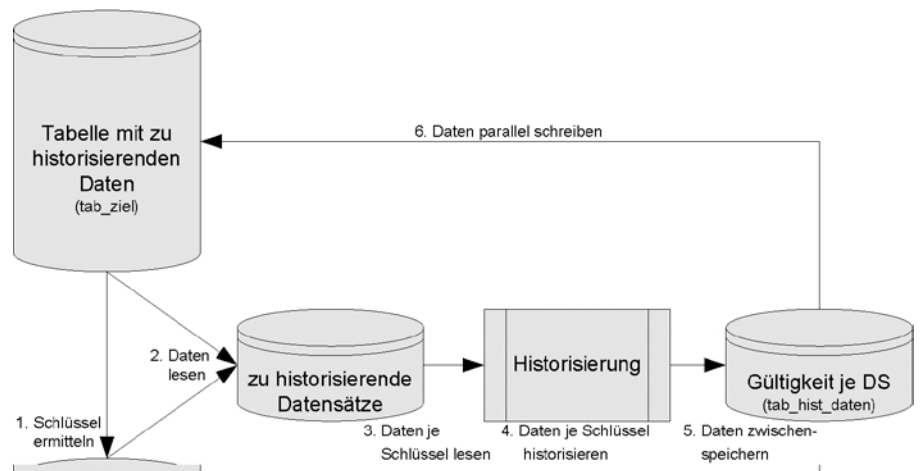


Abbildung 2: Ablauf angepasste Historisierung

	GETS		MISSES		SLEEPS	
	alt	neu	alt	neu	alt	neu
shared pool	2.100.566.433	29.658.667	131.602.673	1.066.005	2.740.417	715
library cache	1.604.674.472	23.173.192	19.110.037	968.176	82.488	577
library cache lock	403.277.029	8.458.705	907.972	631.276	358	157
row cache objects	6.387.481.148	46.192.372	210.970.588	398.618	17.167	159

Tabelle 2: Mess-Ergebnisse

- Parallel DML muss in der Session aktiv sein:
ALTER SESSION ENABLE PARALLEL DML;
- Erfolgt die Verarbeitung mittels eines „updatable join“-Views, ist ein Primary-Key auf einer der Tabellen erforderlich.
- Weitere Voraussetzungen für den Einsatz von Parallel DML sind im Oracle Data Warehousing Guide zu finden.

Da die zu schreibenden Daten der historischen Attribute in einer temporären Tabelle vorliegen, kommt zum Update ein „updatable join“-View zur Anwendung (siehe Listing).

Durch den Einsatz von Parallel DML sank die Laufzeit zur Durchführung des Updates bei partitionierten Tabellen für 7,5 Millionen Datensätze auf 10 Minuten.

Fazit

Derzeit findet in dieser Form täglich bei fast 100 Tabellen erfolgreich eine Historisierung statt. Normalerweise sind Updates von vielen Datensätzen auf großen Tabellen wenn möglich zu vermeiden. Lösungen, bei

```

UPDATE /*+ PARALLEL(upd_view,10) */
      (SELECT /*+
                PARALLEL(tgt,10)
                PARALLEL(src,10)
            */
      src.dat_von AS new_dat_von
      ,src.dat_bis AS new_dat_bis
      ,:p_verarbeitungs_id AS new_verarb_id
      ,tgt.dat_von AS dat_von
      ,tgt.dat_bis AS dat_bis
      ,tgt.id AS id
      FROM   tab_ziel tgt
            ,tab_hist_daten src
      WHERE  0 = 0
            AND tgt.rowid = src.v_rowid
      ) upd_view
SET   upd_view.dat_von = upd_view.new_dat_von
      ,upd_view.dat_bis = upd_view.new_dat_bis
      ,upd_view.vearbeits_id = upd_view.new_ver-
      arb_id
      ;
    
```

Listing: SQL-Anweisung für Parallel DML mit einem „updatable join“-View

denen man mit Inserts arbeitet, sind aufgrund der in den meisten Fällen besseren Laufzeit vorzuziehen. Doch durch die von Oracle in der Enterprise Edition zur Verfügung gestellten Möglichkeiten wie PL/SQL-Tabellen, BULK-Verarbeitung und Parallel DML lassen sich auch große Tabellen per Update mit annehmbaren Laufzeiten bearbeiten.

Weitere Informationen

Parallel DML im Oracle Data Warehousing Guide:
http://download.oracle.com/docs/cd/B19306_01/server.102/b14223/usingpe.htm#sthref2120

Kontakt:
 Tobias Stark
 tobias.stark@metafinanz.de



Oracle-Datenbankadministration einfach, schnell und sicher

Mit DBora vereinfacht sich die tägliche Datenbankadministration spürbar. Dafür wurden viele wichtige Werkzeuge in die Anwendung integriert.

- Wartung der Instanz
- SQL-Plan Base-Lines
- Backup
- Storage Management
- Tablespaces
- SQL-Analyse
- Sessions
- ReDoLog-Dateien
- Reverse Engineering
- Auditing
- Undo/Rollback-Segmente
- Flashback
- Memory Management
- Resource-Management
- Datenbankdokumentation
- Statistics Management
- Security
- und vieles mehr

www.ventara.de

Kostenloser Download Trial-Version

Und das Beste zum Schluss:
 Sie gehen kein Risiko ein. **Testen Sie DBora 30 Tage lang kostenlos** und überzeugen Sie sich.

Mehr als Metadaten: Das Application Express Repository

Carsten Czarski, ORACLE Deutschland GmbH

Application Express (APEX) ist ein metadatengetriebenes Werkzeug, das alle Details der APEX-Anwendungen in den Repository-Tabellen abspeichert. Zur Laufzeit liest APEX die Definitionen aus und generiert die Webseiten der Anwendung. APEX-Anwendungen dokumentieren sich in diesen Repository-Einträgen im wahrsten Sinne des Wortes selbst; es kommt nur darauf an, diese Informationen sinnvoll zu nutzen. Dieser Artikel zeigt einige Ideen und Ansätze dazu auf.

Die APEX Data Dictionary Views bieten einen direkten (lesenden) Zugriff auf das Repository. Damit steht eine offene, dokumentierte Schnittstelle zu den APEX-Metadaten zur Verfügung. In der View APEX_DICTIONARY sind alle Dictionary Views mit allen Spalten samt Erklärungen hinterlegt. Auch die APEX-Entwicklungsumgebung ermöglicht ein Zugriff darauf. Dazu navigiert man von der Startseite des APEX-Workspace zu den Utilities (Werkzeuge) und dort zu den APEX-Views (siehe Abbildung 1).

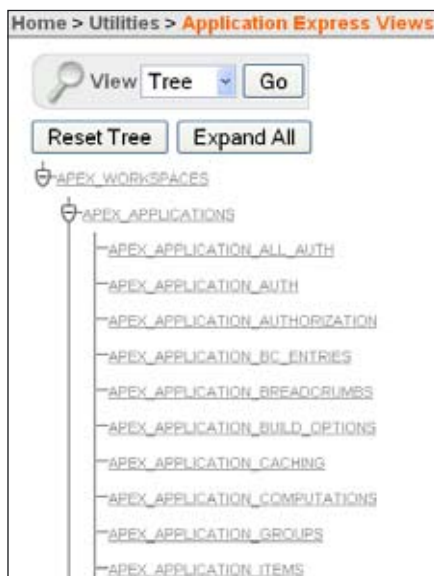


Abbildung 1: Die APEX Dictionary Views sind hierarchisch organisiert

Stellt man die Ansicht auf „Tree“ (Baum) um, so erkennt man die hierarchische Organisation der Views; ein Klick auf einen der Einträge liefert Details zur jeweiligen View. Die Views sind sowohl in der APEX-Entwicklungsumgebung (SQL-Workshop) als auch außerhalb

von APEX (SQL*Plus, JDBC-Verbindungen) nutzbar. Es gelten jedoch stets die APEX-Sicherheitsregeln: So zeigt die Abfrage in Listing 1 (in SQL*Plus ausgeführt) nur den Workspace, dem das Schema des angemeldeten Users zugewiesen wurde – die anderen APEX-Workspaces sind nicht sichtbar.

```
SQL> select workspace_id, workspace, applications
2 from apex_workspaces

WORKSPACE_ID WORKSPACE APPLICATIONS
-----
1841600194840592 PARTNER 14

1 Zeile wurde ausgewählt.
```

Listing 1: Wie viele Anwendungen befinden sich im Workspace?

Ein erster Schritt besteht darin, sich diverse Informationen über APEX-Anwendungen direkt aus dem Repository zu holen. Ein einfaches Beispiel dafür ist in Listing 2 zu finden.

```
SQL> select page_id, page_name
from apex_application_pages
2 where application_id=103
3 order by 1;

PAGE_ID PAGE_NAME
-----
1 Abteilungs-Übersicht
2 Auswertungen
3 Bearbeiten
101 Anmelden

4 Zeilen ausgewählt.
```

Listing 2: Aus welchen Seiten besteht die Anwendung 103?

Qualitätssicherung mit dem APEX-Repository

Das APEX-Repository bietet den strukturierten Zugriff auf die Metadaten einer Anwendung mit SQL. Damit ist es sehr einfach möglich, Qualitätssicherung für Applikationen durchzuführen. Dazu ein Beispiel: Wenn ein Bericht keine Daten zurückgibt, wird normalerweise ein Standardtext ausgegeben; je nach Spracheinstellung lautet dieser „Keine Zeilen gefunden“ oder „no data found“. Wenn die Qualitätssicherung nun verlangt, dass dort aussagekräftige Meldungen wie „Keine Kunden vorhanden“ stehen müssen, könnte man dies anhand des APEX-Repository recht einfach prüfen – hier ist die Spalte APEX_APPLICATION_PAGE_REGIONS.NO_DATA_FOUND_MESSAGE von Interesse. Listing 3 zeigt, wie man mit einer SQL-Abfrage herausfindet, an welchen Stellen der Entwickler die Anforderung noch umsetzen muss.

```
SQL> select application_id, page_id, region_name
2 from apex_application_page_regions
3 and no_data_found_message = 'no data found'

APPLICATION_ID PAGE_ID REGION_NAME
-----
122 1 Modelldetails
122 1 Mining-Modelle
122 2 Analyseergebnis
122 4 Vorhandene Data Mining-Modelle
122 4 Confusion Matrix zeigen

4 Zeilen ausgewählt
```

Listing 3: Berichtsregionen mit Standardmeldung „no data found“

Anhand der anderen Data Dictionary Views lassen sich ähnliche Abfragen zur Qualitätssicherung formulieren wie zum Beispiel:

- Sind alle Labels der Formularelemente linksbündig ausgerichtet?
- Sind alle Seiten, auf die in den Page Branches verzweigt wird, auch wirklich vorhanden?

Ein fertiges Werkzeug für diese Form der Qualitätssicherung ist der APEX-Advisor von Patrick Wolf [1]. Dieser bietet vordefinierte Prüfungen zu gängigen Qualitätsproblemen an (siehe Abbildung 2).

Alle Prüfungen erfolgen anhand der APEX-Metadaten. Einige gehen dabei über eine einfache SQL-Abfrage hinaus. So kann es sein, dass ein in Prozessen oder Bedingungen hinterlegter PL/SQL-Code ungültig ist – APEX prüft den Code nicht immer. Der Check „Is Valid PL/SQL Code“ liest den hinterlegten Code (beispielsweise aus der Spalte PROCESS_SOURCE in APEX_APPLICATION_PAGE_PROC), parst diesen und prüft so, ob er gültig ist oder nicht.

Der APEX-Advisor ist als Einstieg in die Qualitätssicherung mit dem APEX-Repository sehr zu empfehlen, da viele Prüfungen bereits enthalten sind. Ist ein benötigter Check nicht vorhanden, so kann man ihn nach dem hier beschriebenen Prinzip einfach selbst durchführen: Zunächst die Metadaten-Views mit den benötigten Informationen identifizieren sowie anschließend eine SQL-Abfrage oder eine PL/SQL-Prozedur schreiben, welche die Informationen zusammenträgt, aus-

wertet und bei Bedarf eine Fehlermeldung auslöst. Wenn man zufrieden ist, kann man den Check sogar selbst zum APEX-Advisor hinzufügen.

Die Qualitätssicherung zeigt bereits den hohen Wert des APEX-Repository. Klassischerweise würde man diese per Code Review durchführen, ein langwieriger und fehleranfälliger Prozess. APEX erlaubt mithilfe des Repository einfache, schnelle und standardisierte Prüfungen.

Generischen Code mit dem APEX-Repository entwickeln

Die Views des APEX-Repository eignen sich auch sehr gut zur Integration in eigene Applikationen. Ein sehr simples Beispiel ist eine Menüseite für die Anwendungen eines Workspace. Dies ist als einfacher Bericht auf die View APEX_APPLICATIONS realisierbar. Doch auch anspruchsvollere Themen sind denkbar. So enthält die APEX-Community ein Beispiel zu den berühmten „abhängigen Auswahllisten“ (Cascading LOV) [2]. In diesem Beispiel werden abhängige Auswahllisten mit AJAX-Technologie realisiert. Alle nötigen Informationen stehen im APEX-Repository, so dass der Code nur einmal hinterlegt werden muss und für alle Auswahllisten einer Anwendung nutzbar ist. Auch das APEXLib-Framework von Patrick Wolf [3] baut auf diesem Prinzip auf.

Ein weiterer Tipp von Thomas Fuhr [4] ist ebenfalls generisch auf Basis von APEX Dictionary Views realisiert. Dieser Tipp erlaubt es, die Texte, die zu jedem Formularelement als Online-Hilfe hinterlegt werden können, einfach dar-

zustellen, sobald der Fokus auf diesem Feld liegt, und nicht, wie standardmäßig, als Popup-Fenster bei Klick auf das Label (siehe Abbildungen 3 und 4).

Projektkalkulation mit dem APEX-Repository

Da alle Metadaten einer Anwendung im Repository enthalten sind, dient dieses auch der Analyse bestehender APEX-Applikationen. Das Ziel ist, die Informationen als Grundlage zur Analyse und Bewertung vorhandener oder zur Kalkulation neuer Anwendungen zu nutzen. Auch hierzu enthält die APEX-Community bereits einen ausführlichen Tipp [5].

In einem ersten Schritt muss man sich einen Überblick über die Struktur der Anwendungen verschaffen. Da die relevanten Informationen über mehrere APEX Dictionary Views verteilt sind, sind diese zunächst zusammenzufassen. Eine solche Zusammenfassung könnte wie in Listing 4 aussehen. Das SQL zur Erstellung der View APEX_APP_STRUCTURE ist im APEX-Community-Tipp [5] enthalten.

```
SQL> select component from APEX_
APP_STRUCTURE

COMPONENT
-----
APPLICATION: 103 ( )
  PAGE: 2 ( )
    REGION: Abteilungs-Daten ( )
    REGION: Gehaltsaufteilung ( )
  PAGE: 101 ( )
    ITEM: P101_USERNAME (Text
Field)
    ITEM: P101_PASSWORD (Password
(...
PROCESS: Clear (Clear Cache
for all Items on Pages
  PAGE: 1 ( )
    ITEM: P1_DEPTNO (Select List
with Submit)
    REGION: Abteilungen ( )
:
```

Listing 4: Zusammenfassung mehrerer Metadaten-Views in einer einzigen

Die so ermittelten Komponenten lassen sich in einem nächsten Schritt klassifizieren; anschließend kann man die Kategorien auszählen und gewichten. Ziel ist es, die APEX-Anwendung

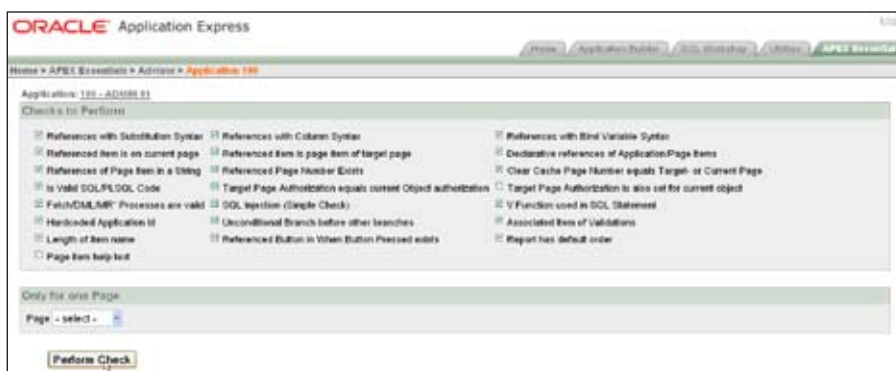


Abbildung 2: APEX-Advisor: Auswahl der Prüfungen



Abbildung 3: Hinterlegung des Hilfetextes für ein Formularelement



Abbildung 4: Die etwas andere Darstellung des Hilfetexts

anhand weniger Kenngrößen zu charakterisieren. Am Ende erhält man wenige Kennzahlen, die Auskunft über die Komplexität einer Anwendung geben. Falls es Aufzeichnungen über die konkrete Entwicklungszeit gibt (Personentage), so hat man einen Zusammenhang zwischen Anwendung und Aufwand, der anhand anderer Anwendungen verifiziert werden und als

Grundlage für die Kalkulation neuer Projekte dienen kann. Natürlich wird ein solches Verfahren keine exakten Voraussagen machen; gerade während der Entwicklung neuer Anwendungen ändern sich die Anforderungen ständig. Ein geübter APEX-Entwickler kann damit allerdings schon während der Diskussion der Anforderungen abschätzen, welche Element- oder Berichtstypen

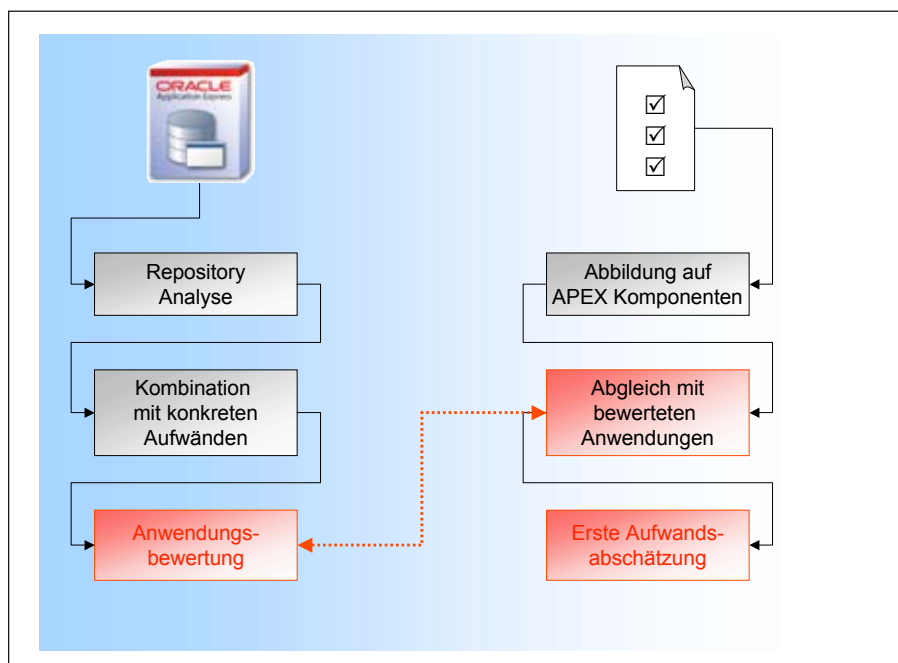


Abbildung 5: Analyse und Schätzung von APEX-Anwendungen

wahrscheinlich notwendig sind. Damit lassen sich die voraussichtlich benötigten Komponenten schon im Vorfeld abschätzen, gewichten und aufsummieren. Das Ergebnis ist eine (wenn auch sehr grobe) Abschätzung über den voraussichtlichen Aufwand. Abbildung 5 macht diesen Prozess deutlich.

Fazit

Da Application Express ein Metadaten-getriebenes Werkzeug ist, dokumentieren sich alle Anwendungen automatisch selbst. Aufbau und Komponenten einer Anwendung sind nicht im Code versteckt, sondern liegen stets in strukturierter Form vor. So muss man keine Code-Analyse durchführen um festzustellen, wie viele Eingabefelder in einem Formular enthalten sind.

Das Repository kann vielfältig genutzt werden: Ob Programmierregeln eingehalten wurden, lässt sich anhand einfacher SQL-Abfragen feststellen – das Repository wird so zur wichtigen Grundlage für die Qualitätssicherung.

Wie die Tipps der APEX-Community zeigen, kann auch generischer Code entwickelt werden, der das APEX-Repository entsprechend nutzt. Und schließlich könnte das Repository Grundlage zur Analyse bestehender und zur Kalkulation neuer APEX-Anwendungen sein – letztere Idee steckt aber sicherlich noch in den Kinderschuhen und wäre weiterzuentwickeln.

Weitere Informationen

- [1] APEX-Advisor (Teil der APEX-Essentials) von Patrick Wolf: <http://www.inside-oracle-apex.com/apex-essentials/>
- [2] APEX-Community: Cascading LOV mit AJAX: <http://www.oracle.com/global/de/community/tips/ajax-jquery-01/index.html>
- [3] APEXLib-Framework von Patrick Wolf: <http://www.inside-oracle-apex.com/apex-lib-framework/>
- [4] APEX-Community: Formularhilfe von Thomas Fuhr: <http://www.oracle.com/global/de/community/tips/formularhilfe/index.html>
- [5] APEX-Community: Anwendungsanalyse mit dem APEX-Repository: <http://www.oracle.com/global/de/community/tips/repo/index.html>

Kontakt:

Carsten Czarski
carsten.czarski@oracle.com

ITIL-Prozesse optimieren und den Betrieb effizient umsetzen

Die DOAG und Oracle veranstalten am 8. September 2009 in Köln gemeinsam die Fachkonferenz **DOAG ITIL & Betrieb 2009**.

Die Veranstalter knüpfen damit an die erfolgreiche Tagung des Vorjahrs an. Experten zeigen Ihnen, wie ein Unternehmen den Betrieb von ITIL vorteilhaft betreiben und damit Kosten sparen kann. Setzen Sie hier den Hebel an:

- In den Betriebsprozessen und deren Schnittstellen, indem die zu erbringenden IT-Services qualitativ und vollständig definiert und optimiert sind
- In der Service-Definition, denn Servicebäume und -graphen bestimmen exakt die Services innerhalb der IT sowie durch SLAs gegenüber dem Business
- Im Mindset der IT, denn die Anwender benötigen IT-Services, die ihr Kern-Business optimal unterstützen
- In der Aufgaben- und Rollendefinition aller Bereiche, die für die vollständige und serviceorientierte Lieferung von IT-Services notwendig sind



Sie erfahren außerdem die optimale Umsetzung Ihrer Strategien durch Unterstützung und Integration der Oracle System- und Software-Lösungen. Neben der eigentlichen Fachtagung haben Sie im Informationsareal der Veranstaltung ausführlich Zeit und Gelegenheit, Ihr Kontakt-Netzwerk zu erweitern und sich mit Oracle-Spezialisten sowie den teilnehmenden Partnern auszutauschen. Weitere Informationen unter www.doag.org/go/itil09

Durch Leistung

überzeugen

Wer heute erfolgreich im Markt agieren will, braucht eine klare Strategie – und eine reibungslos funktionierende, effiziente Informationstechnologie. Als langjähriger IT-Beratungspartner von Großunternehmen und Mittelstand hat die MT AG bereits zahlreiche anspruchsvolle IT-Aufgaben umgesetzt. Die MT AG steht dabei für Technologie-Know-how und praxisnahe, effiziente IT-Dienstleistung – von Strategie und Beratung über Entwicklung und Integration der Lösung bis hin zu Wartung und Administration von IT-Infrastrukturen. Ganz gleich, ob es sich um konzernweite Lösungen oder Lösungen für Fachabteilungen handelt – die Experten der MT AG sind Garanten für maßgeschneiderte, nutzerfreundliche und effiziente Lösungen. Lassen Sie sich doch auch überzeugen!



Besuchen Sie unsere Vorträge auf
der DOAG: 17. – 19. November 2009
Wir freuen uns auf Ihren Besuch!

MT AG
Balcke-Dürr-Allee 9
40882 Ratingen
Tel. +49 (0) 2102 309 61-0
Fax +49 (0) 2102 309 61-10
info@mt-ag.com
www.mt-ag.com

ORACLE CERTIFIED ADVANTAGE
PARTNER

Oracle ADF 11g – Komponenten zur Daten-Visualisierung

Luc Bors, AMIS Services B.V. und Hans-Chr. Kaller, AMIS Deutschland GmbH

Im Oktober 2008 hat Oracle JDeveloper und ADF 11g veröffentlicht. Eine neue Komponente in ADF Faces Rich-Client ist ein Toolpaket für die Visualisierung von Daten, die Data Visualisierung Tools.

Diese Data Visualisierung Tools (DVT) umfassen eine reichhaltige Sammlung interaktiver JSF-Komponenten, die dem Anwender sehr umfangreiche Möglichkeiten für die Analyse und Visualisierung von Daten mittels Diagrammen (Graphen) und Tabellen bietet. Dieser Artikel gibt eine Einführung in DVT und zeigt auf, wie man die einzelnen Komponenten sinnvoll anwendet. Im Mittelpunkt steht das einfache Beispiel „Erhebung von Daten“.

Mit DVT stellt Oracle nun ein zentrales Werkzeug zur Verfügung, das für die gesamte Produktpalette eingesetzt werden kann, um Grafiken oder andere grafische Datendarstellungen zu erzeugen und zu visualisieren – wie zum Beispiel für Enterprise Manager und Hyperion, Business Activity Monitoring (BAM) und natürlich auch Oracle Fusion Anwendungen. Dies wird bei konsequenter Anwendung durch Entwickler auch dazu führen, dass beim

Anwender ein einheitliches „look & feel“ entsteht.

Die Anforderungen für die Benutzung der DVT sind die gleichen wie für andere ADF-Komponenten. Voraussetzung wird eine ADF-Runtime- oder eine Application-Server-Lizenz. Das ADF-Entwicklungspaket (einschließlich DVT) erfordert keine Lizenz; die Nutzung der JDeveloper-Komponente ist kostenlos.

Eingebettet in die ADF-Architektur

Aufgrund der allgemeinen ADF-Architektur ist es möglich, Graphen aus jedem Daten-Konstrukt zu erstellen, der über ADF Data Control zur Verfügung gestellt werden kann. Verknüpfungen (Bindungen) mit einem Standard Rowset und mit hierarchischen Datenstrukturen werden unterstützt. Dies bedeutet, dass man Diagramme sowohl aus Daten einer Datenbank über

ADF Business-Komponenten als auch aus anderen Quellen wie BAM oder dem BI-Server generieren kann. Zudem ist es auch möglich, über Web-Services Daten für die Erzeugung von Diagrammen zur Verfügung zu stellen.

Ein Diagramm wird durch die Zusammenstellung von Daten (Data Collection) per Drag & Drop aus dem Data Control Panel (Datenquelle) erzeugt. Dies erfolgt auf die gleiche Weise, wie auch ein Databound Form oder eine Tabelle auf einer Page entstehen. Daten für ein Diagramm lassen sich auch über Managed Bean erzeugen.

DVT-Komponenten unterstützen sowohl „Skinning“ als auch „Partial Page Refresh“. DVT können Ziele von Partial Page Rendering (PPR) sein, sie können aber auch ein PPR-Ereignis initiieren. Es ist auch möglich, nahezu jede Eigenschaft einer DVT-Komponente dynamisch mithilfe von EL zu verändern.

Eine Sammlung grafischer Funktionen

DVT besteht aus fünf Hauptgruppen:

- *Diagramme (Graphen)*
Anzeige von Daten auf Basis von Standards (einschließlich Balken-, Linien-, Tortendiagramme etc.) und individueller Vorgaben. Die richtige Auswahl eines Diagramms ist abhängig von der Struktur der darzustellenden Datenquelle.
- *Gauge*
Angezeigtes Ergebnis einer Messung; die Messgröße ist relativ zu verschiedenen Grenzwerten.
- *Pivot-Tabellen*
Tabellarische Datendarstellung mit horizontaler und/oder vertikaler Dimension (Zeile / Spalte). Die Darstel-

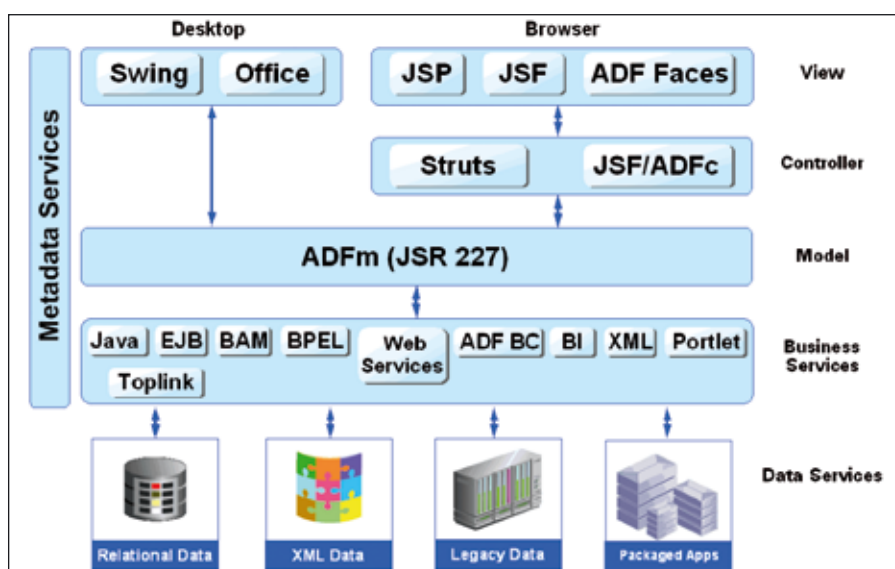


Abbildung 1: Die Architektur von ADF

lung der Dimension kann auf mehreren Ebenen erfolgen, das Vertauschen der Dimensionen von Zeilen und Spalten ist möglich („pivotieren“)
Landkarten

Geografische Darstellungskomponente, die es unter anderem ermöglicht, in mehreren Schichten (Punkt-) Daten auf einer einzigen Karte anzuzeigen.

- **Gantt-Chart**

Balkendiagramm-Tafel (Gantt Chart) zur Zeitplanung von Projekten und Überwachung der Durchführung. Es gibt drei Typen: Projekt, Planung und Nutzung von Ressourcen.

In Abbildung 2 sind einige grundlegende Eigenschaften der DVT-Komponenten aufgezeigt. Die beschriebenen Beispiele wurden auf der Grundlage gemeinsamer Daten (Datenquelle salesresult) erstellt.



Abbildung 2: Platzhalter Datensteuerung

Alle DVT-Funktionen, die in diesem Artikel beschrieben werden, sind auf der gleichen ADF-Page enthalten und die dazugehörige Verknüpfung ist in einer Page-Definitions-Datei festgelegt. Die grafische Anzeige (Icons) der Page-Definitions zeigt, dass jede einzelne DVT-Komponente auch den gleichen Zeiger nutzt. Die gewählte Form muss nicht unbedingt die beste Wahl sein, ist aber für die Darstellung in diesem Artikel gut geeignet.

Diagramm-Repertoire

Die Visualisierungskomponente für Daten besteht aus mehr als fünfzig verschiedenen Arten von Diagrammen. Alle unterstützen Animation (Erstanzeige und Datenveränderung), Vergrößerung und Verkleinerung sowie den Bildlauf, einige auch ein Zeitsuch-

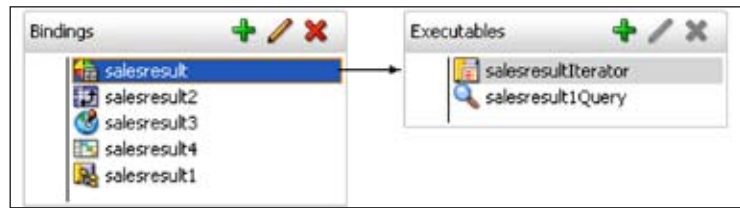


Abbildung 3: Data Binding

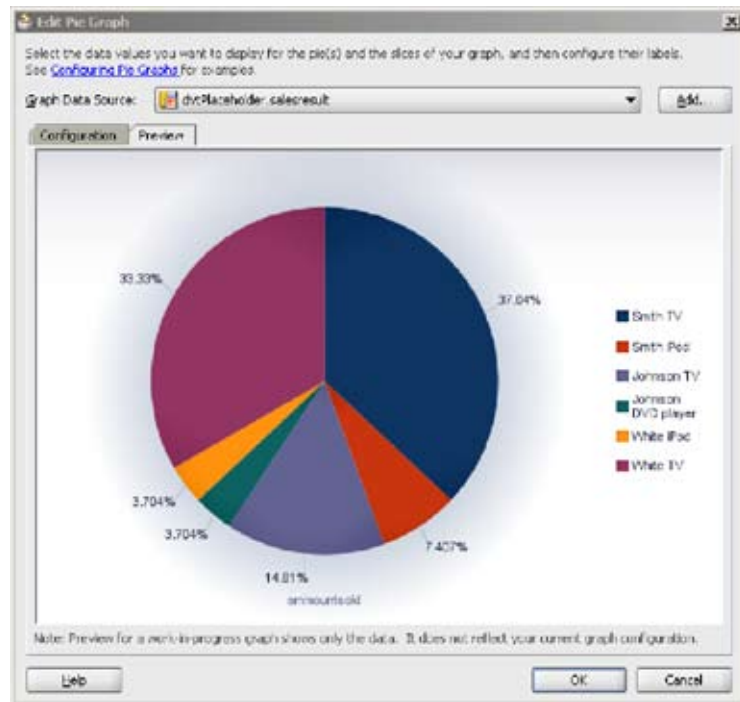


Abbildung 4: Diagrammvorschaufunktion

fenster. Die wichtigste Komponente für Diagramme ist die <dvt:graph/>-Komponente mit insgesamt 77 Eigenschaften (Attributen). Für die am häufigsten verwendeten Diagramme existieren vereinfachte Typen, so dass nicht sämtliche verfügbaren Eigenschaften gesetzt werden müssen.

Tortendiagramm

Ein Beispiel: Das Ziehen der Datenquelle salesresult aus dem Data Control Panel auf die Diagrammseite generiert ein Popup-Fenster, in dem der Diagrammtyp wählbar ist. Für dieses Beispiel wählen wir ein Tortendiagramm aus. Der nächste Schritt ist die Konfiguration des Diagramms. Das Diagramm soll den Prozentsatz am gesamten Umsatz pro Verkäufer aufzeigen. Hierzu wählt man die entsprechenden Werte und überprüft dann im Previewer, ob das Ergebnis den Erwartungen entspricht.

Es ist sehr einfach, das Diagramm durch entsprechende Inhalte und durch Animation anzureichern. Der erste Schritt hierfür wäre zum Beispiel, dem Diagramm eine dreidimensionale Ansicht zu verpassen. Dies geschieht durch Hinzufügen der Eigenschaft threeDEffect="true".

```
<dvt:pieGraph id="pieGraph1"
    inlineStyle="width:100%;
    height:400px;"
    value="#{bindings.
    salesresult.graphModel}"
    subType="PIE"
    threeDEffect="true"
    imageFormat="FLASH"
    seriesEffect="SE_GRADIENT"

    seriesRolloverBehavior="RB_
    HIGHLIGHT">
    <dvt:pieLabel
    rendered="true"/>
    <dvt:legendArea
    automaticPlacement="AP_NEVER"
    rendered="false"/>
</dvt:pieGraph>
```

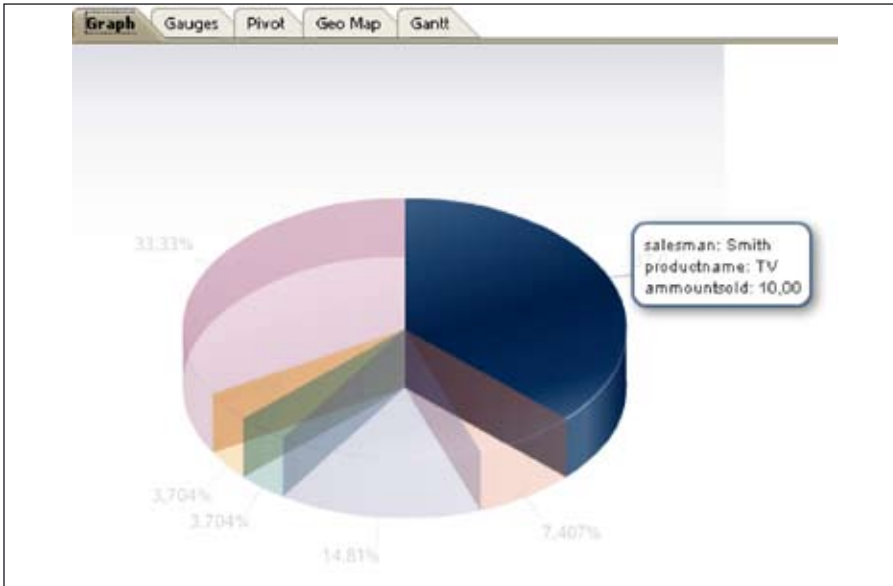


Abbildung 5: Tortendiagramm (angereichert)

Pivot-Tabellen

Diese sehr mächtige Darstellungskomponente zeigt normalisierte relationale Daten in einer benutzerfreundlichen Matrixstruktur an. Es ist zudem möglich, Daten aus unterschiedlichen Blickwinkeln zu betrachten. Ein Beispiel: Man möchte wissen, welchen Auftragseingang welcher Vertriebsbeauftragte eines Sales-Teams in welcher Stadt (Region) erzielt hat. Gleichzeitig soll das verkaufte Produkt differenziert zugeordnet sichtbar sein. Diese Anforderung lässt sich durch unterschiedliche Darstellungsformen realisieren, möglich sind:

Gauge-Messwert-Darstellung

Die Anzeige von Messgrößen stellt beispielsweise kritische Werte dar. Die Werte konzentrieren sich dann auf einen bestimmten Datenpunkt, beispielsweise den Warenbestand in einem Lagerhaus oder die Absatzzahlen eines bestimmten Produkts. Basis für die Anzeige einer Messgröße kann durchaus eine Messgrößen-Reihe sein. Für die Darstellung wird jedoch ein Einzelwert selektiert, da nur ein Wert abhängig vom definierten Wertetyp aus dieser Messreihe pro Darstellungszeile sinnvoll angezeigt werden kann. Der Wertetyp (Klasse) ergibt sich aus der Klassifikation als Minimum-, Maximum- oder Schwellenwert. Damit eignen sich Gaugen hervorragend für das Verfolgen kritischer Werte.

Die Darstellung der Gaugen kann für Klassen farblich unterschiedlich erfolgen, sie lässt sich etwa durch Flash oder SVG anreichern, und sie kann auf drei verschiedene Arten zur Anzeige kommen: Ziffernblatt (dial), Längemaß (status meter) oder LED. Für die Anzeige eines kritischen Wertes ist eine beliebige Anzahl von Schwellenwerten möglich. Das nachfolgende Listing sowie Abbildung 6 zeigen ein Beispiel dafür. Führt man den Mauszeiger über einen der Punkte, so erscheinen zusätzliche Informationen über den aktuell dargestellten Messwert und den nächstgelegenen Schwellenwert.

```
<af:column sortProperty="amountsold" sortable="false"
headerText="+/-">
  <dvt:gauge id="gauge1" inlineStyle="height:20px;
width:20px;"
    gaugeType="LED"
    imageFormat="FLASH" ledStyle="LS_DOT"
    value="{row.bindings.amountsold}"
    dynamicResize="DYNAMIC_SIZE"
    maxValue="20.0">

    <dvt:gaugeBackground>
      <dvt:specialEffects fillType="FT_GRADIENT">
        <dvt:gradientStopStyle/>
      </dvt:specialEffects>
    </dvt:gaugeBackground>

    <dvt:thresholdSet>
      <dvt:threshold fillColor="#ff0000" thresholdMaxVa-
lue="3"/>
      <dvt:threshold fillColor="#ff9c31" thresholdMaxValu
e="6"/>
      <dvt:threshold fillColor="#52ff52"/>
    </dvt:thresholdSet>
    <dvt:metricLabel position="LP_NONE"/>
  </dvt:gauge>
</af:column>
```

salesman	productname	+/-	periodstartdate	periodenddate	city
Smith	TV	●	1-1-2009	1-2-2009	Amsterdam
Johnson	TV	●	1-1-2009	1-2-2009	London
Smith	iPod	●	1-1-2009	1-2-2009	Amsterdam
White	iPod	●	1-1-2009	1-2-2009	Berlin
White	TV	●	1-1-2009	1-2-2009	Paris
Johnson	DVD player	●	1-1-2009	1-2-2009	Rome

Abbildung 6: LEDs in einer Tabelle

- Pivot by City
- Pivot by Product
- Simultane Darstellung

Die Pivot-Tabelle ermöglicht es, alle Darstellungsformen in einer einzigen Tabelle abzulegen und durch unterschiedliche, aber einfachste Handhabung zu betrachten. Wir benutzen wieder die Datenquelle salesresult und erstellen in Anlehnung an das Vorgehen für das Gauge-Diagramm jetzt eine Pivot-Tabelle. Der Sourcecode für die Pivot-Tabellenfunktion ist wirklich sehr einfach:

```
<dvt:pivotTable
id="pivotTable1"
inlineStyle="width:100%; „
value="#{bindings.salesresult2.pivotTableModel}"/>
```

Die Funktionalität ist hingegen deutlich mächtiger. Die Pivot-Tabellenansicht lässt sich durch einfaches Ziehen der Zeilen oder Spalten verändern, dadurch entsteht ohne jede Veränderung der Dateninhalte eine andere Sicht. Die Abbildungen 7 und 8 zeigen, wer was in welcher Stadt verkauft hat und in welcher Stadt was von wem verkauft wurde. Ein eindrucksvolles Beispiel für die Realisierung einer Zellenarstellung (heat map) steht unter <http://www.oracle.com/technology/pub/articles/jellema-adfcellhighlighting.html>. Es ist durch entsprechende Anpassungen für Pivot-Tabellen nutzbar.

	amountsold					
	Smith	Johnson	White			
	TV	iPod	TV	DVD player	iPod	TV
Amsterdam	10	2				
London			4			
Berlin						1
Paris						9
Rome				1		

Abbildung 7: Pivot by City

	amountsold					
	Amsterdam	London	Berlin	Paris	Rome	
	Smith	Johnson	White	White	Johnson	
TV	10	4			5	
iPod	2		1			
DVD player						1

Abbildung 8: Pivot by Product

Geografische Karten

Mit der Option, Daten auf eine (geografische) Karte zu projizieren, lassen sich beispielsweise Regionen farblich differenziert anzeigen. Ohne großen Aufwand sind gleich mehrere Schichten und unterschiedliche Motive (multiple layers) auf einer Karte möglich. Die aktuell vorhandene Darstellungsmöglichkeit geografischer Karten basiert auf Oracle Spatial. Leider können andere Karten wie zum Beispiel Google Maps als Basis (noch) nicht genutzt werden. Wie bei den anderen Ansichten aus dem DVT-Repertoire lassen sich auch geografische Karten anpassen (Kartengröße, Startkoordinaten oder Zoomlevel).

Jedes Kartenmotiv erfordert einen Daten-Kontrollsatz, der die mit einer Karte verknüpften Informationen enthält. Für unser letztes Beispiel ist es ausreichend, die geografischen Koordinaten in die Datenquelle einzufügen (beispielsweise Breiten-/Längengrad 52,2 und 4,5 für Amsterdam). Vor der Nutzung von Geographic Map ist eine Serververbindung zu Mapviewer und zu Geocoder in JDeveloper herzustellen. Durch diese Verbindung ist es dann sehr einfach, die geografische Karte mit der Datenquelle salesresult zu verknüpfen (siehe Listing und Abbildung 9).

```
<mapTheme IterBinding="salesresultIterator" id="salesresult3"
xmlns="http://xmlns.oracle.com/adfm/dvt">
<mapThemeDataMap
mapThemeType="point">
<item type="lat_
long" longitude="longi"
latitude="lat" label="city"/>
<item type="data"
value="amountsold"
label="amount sold"/>
<item type="category"
value="productname"
label="product name"/>
</mapThemeDataMap>
</mapTheme>
```

Gantt-Charts

Die für Projektplanung und Projektverfolgung verwendeten Gantt-Diagramme beinhalten insbesondere Aufgaben

und Ressourcen für einen festgelegten Zeitrahmen mit einem Projektbeginn und mit einem Projektende (Projektdefinition). Es existieren drei Arten von Gantt-Charts:

- Projekt (Aufgaben)
- Ressourcen
- Zeitplan

Das Projekt-Gantt listet die Aufgaben vertikal und zeigt die Dauer für jede Aufgabe als einen Balken auf einer horizontalen Zeitachse. Ressourcen-Gantt stellt grafisch dar, ob die Ressourcen über oder unter dem Soll liegen. Ein Zeitplan-Gantt basiert auf (manuellen) Zeitplanübersichten und gibt die Ressourcen vertikal mit den dazugehörigen Tätigkeiten auf einer horizontalen Zeitachse an.

Die Komponente besteht im Prinzip aus zwei angrenzenden Baumtabellen mit einer Trennleiste dazwischen. Die linke Tabelle zeigt die Aufgaben, die rechte den Balken zwischen Start- und Endpunkt. Beide Tabellen greifen auf dieselbe Datenquelle zu (data sharing) und werden daher automatisch synchronisiert. Das passiert auch, wenn sich Teilaufgaben verändern. Da Gantt PPR verwendet, ist es nicht notwendig, die gesamte Page aufzufrischen, sondern nur die relevanten Elemente.

Wir nutzen für unser Beispiel wieder die Datenquelle salesresult und erstellen in Anlehnung an das Vorgehen



Abbildung 9: GeoMap

für die bisher erstellten Diagramme und Anzeigen auf der Page ein Projekt Gantt (siehe Abbildung 10).

Auf der generierten Page sehen wir rechts als Balken dargestellt die Verkaufsperioden des jeweiligen Verkäufers. Auf der linken Seite können wir – je Selektion der Spalten – die dem Verkäufer zugeordnete Information (hier: Startzeitpunkt, Endzeitpunkt und City) anzeigen (siehe Abbildung 11). Es ist jederzeit möglich, auf einfache Art und Weise Zeitskalen anzupassen oder Spalten der linken Seite zu ergänzen oder zu entfernen.

Blick in die Zukunft

In naher Zukunft wird Oracle die nächste Version von ADF 11g (Codename „Bulldog“) veröffentlichen. Mit diesem Release gibt es dann vielleicht

eine neue, zusätzliche Komponente im DVT Set, den Hierarchy Viewer (siehe Abbildung 12).

Diese Komponente wurde bereits von Oracle veröffentlicht (siehe BI Beans Statement of Direction unter <http://www.oracle.com/technology/products/bib/oraclebibeansod2008.pdf>) und wird vorrangig für das Anzeigen von Organisations-Charts Anwendung finden. Darüber hinaus sind zukünftig auch alle anderen hierarchischen Darstellungsformen möglich.

Entwickler und Anwender hoffen zudem, dass mit dem nächsten Oracle Release eine fortgeschrittene, modernisierte Version von Active Data Services (ADS) freigegeben wird. ADS basiert auf einem Server-Push-Mechanismus, mit dem Clients auf Veränderungen der Daten sofort reagieren können.



Abbildung 12: Hierarchy Viewer

Ein manuelles Refresh ist somit nicht mehr nötig, da der Server Änderungen direkt an die Clients kommuniziert. ADS war bereits im letzten Technical Preview (TP4) enthalten, aber Oracle hat es dann doch nicht im Boxer-Release freigegeben.

Fazit

ADF 11g DVT ist ein sehr mächtiges und dennoch einfach zu nutzendes Toolset. Es erlaubt die Betrachtung von Daten und Informationen aus den verschiedensten Perspektiven. Oracle hat damit eine Grafik-Komponente entwickelt, die in allen Produkten eingesetzt werden kann, wenn Graphen-Darstellungen oder andere Arten von Daten-Visualisierungen erforderlich sind. Der einheitliche Ansatz und die integrale Nutzung werden zu einem einheitlichen „look & feel“ über die gesamte Produktpalette führen.

Die Beispiele in diesem Artikel sind sicherlich nur die Spitze des Eisbergs. Wer heute damit beginnt, sich zu diesem Thema weiterzubilden, und die Funktionen geschickt einsetzt, wird schnell feststellen, welche große Erleichterung DVT dem Entwickler aber auch dem Anwender bringt.

Kontakte:

Luc Bors
 Hans-Chr. Kaller
 kaller@amis-de.com
<http://technology.amis.nl/blog>

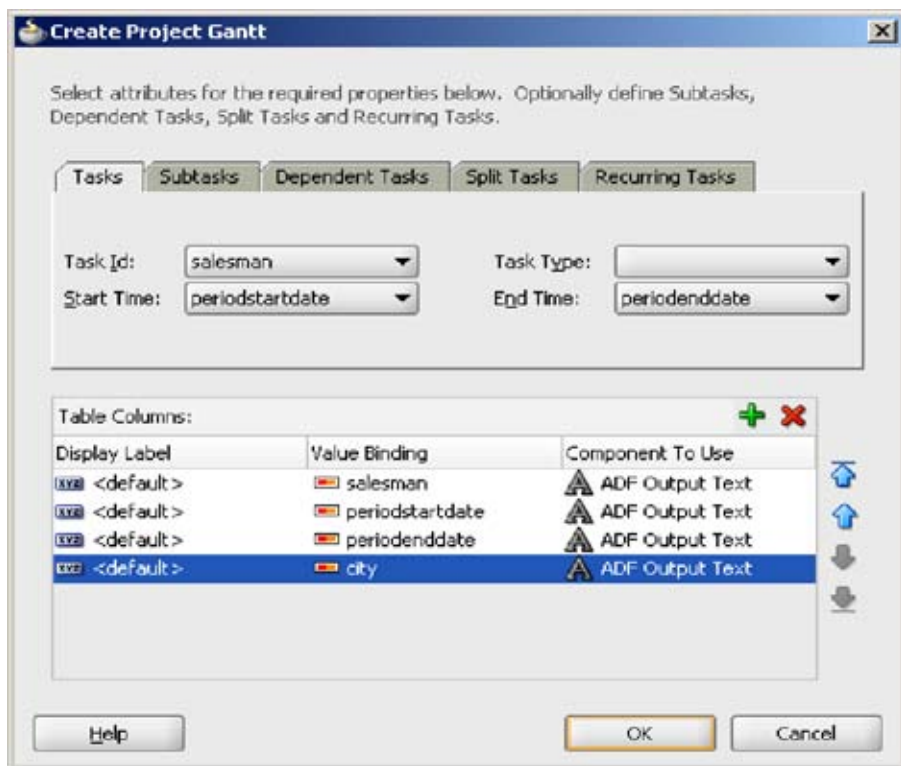


Abbildung 10: Gantt Chart

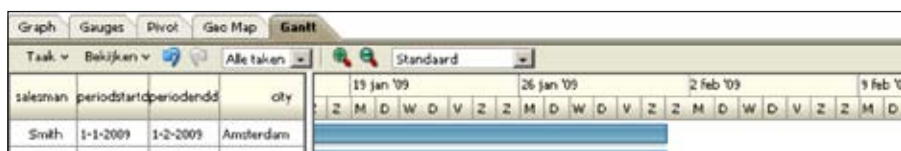


Abbildung 11: Gantt Page

DBMS_COMPARISON

zum asynchronen Vergleich von Tabellen

Dr. Elke Fritsch, MuniQSoft GmbH

Viele Tools zum Vergleich einzelner Datenbank-Objekte vergleichen nicht deren Inhalte, sondern nur die Definitionen dieser Objekte. Seit Version 11g dient das Package DBMS_COMPARISON dem Vergleich und der Synchronisation von Tabellen und in eingeschränktem Maße auch von Views.

Es gibt eine ganze Reihe von Tools zum Vergleich einzelner Oracle-Datenbank-objekte oder ganzer Schemata auf dem Markt [1]. Wenn man einzelne Objekte hingegen mittels Oracle SQL beziehungsweise PL/SQL auf Unterschiede innerhalb der Datensätze überprüfen möchte, gab es bisher nur die Möglichkeit, mit SET-Operatoren zu arbeiten [2,3] oder die Funktionen ORA_HASH, DBMS_UTILITY.get_hash_value [4] oder DBMS_CRYPTO.hash [5] zu verwenden. Mit dem bereits in Version 8 zur Verfügung stehenden Package DBMS_RECTIFIER [6] konnte man lediglich ermitteln, welche Datensätze nur jeweils in einer von zwei Tabellen vorhanden sind. Differenzen innerhalb der Datensätze wurden damit nicht gefunden.

Version 11g bietet für den Vergleich und die Synchronisation von Tabellen sowie in eingeschränktem Maße auch von Views das Package DBMS_COMPARISON [7]. Zur Funktionsweise ist im Web einiges zu finden [8 – 11], wobei sich die Beispiele auf kleine Objekte wie die allseits beliebte emp-Tabelle beziehen. Dieser Artikel gibt einen Einblick in die grundlegenden Funktionen dieses Packages sowie ein paar Tipps für seine Verwendung und vergleicht die Performanz der neuen Methode mit dem in [2] beschriebenen Ansatz.

Um den Umgang mit DBMS_COMPARISON zu vereinfachen und zu automatisieren, hat der Autor ein Package erstellt, das die einzelnen Unterprogramme sowie die Abfragen der Data-Dictionary-Views zusammenfasst. Es ist zusammen mit weiteren Informationen und Beispielen unter http://www.muniqsoft.de/tipps/monatstipps/monattipps_index.htm zu finden.

DBMS_COMPARISON lässt sich nur für den Ver- und Abgleich von Tabellen, Views, die auf einer einzigen Tabelle basieren, und Materialized Views einsetzen. Dabei gilt:

Die zu vergleichenden Objekte können im gleichen Schema, in zwei Schemata der gleichen Datenbank oder in zwei über einen Datenbanklink verknüpften Datenbanken liegen. Die Versionsnummer der „Remote“-Datenbank muss mindestens 10.1 sein. Die Beispiele basieren auf einer 10.2-Express-Edition.

- Wenn die Objekte nicht die gleiche Struktur haben, kann man den Vergleich auch auf einzelne Spalten beschränken. Spalten vom Typ LOB, BFILE, LONG, XML type etc. sind vom Vergleich auszuschließen.
- Jedes Objekt muss einen einfachen oder zusammengesetzten Primärschlüssel beziehungsweise einen Unique-Index auf einer oder mehreren „NOT NULL“-Spalten haben.

Die wichtigsten Views dieses Packages lauten:

- *dba/user_comparison* speichert Namen und Parameter
- *dba/user_comparison_scan* speichert Informationen über jeden durchgeführten Vergleich
- *dba/user_comparison_row_dif* speichert die gefundenen Unterschiede

Der Vergleich und die Synchronisation erfolgen über drei bis vier Stufen:

- *Erstellen eines Vergleichs*
DBMS_COMPARISON.create_comparison
- *Der eigentliche Vergleichsvorgang*
DBMS_COMPARISON.compare
- *Synchronisation*
DBMS_COMPARISON.converge
- *Löschen des Vergleichs beziehungsweise der Ergebnisse*
DBMS_COMPARISON.drop_comparison beziehungsweise purge_comparison

DBMS_COMPARISON.create comparison

Zunächst wird über die Prozedur DBMS_COMPARISON.create_comparison eine Art Template für den Vergleich eingerichtet, in dem festgelegt ist, mit welchen Objekten und wie die Überprüfung erfolgen soll. Als Grundlage für das Beispiel dient eine vergrößerte Kopie der Tabelle dba_objects mit einer Million Zeilen, die sowohl im Schema Scott (unter dem Namen object1) als auch im Schema System einer XE-Datenbank auf einem anderen Rechner (als object2) vorliegt. Ist der Parameter column_list nicht gesetzt, werden alle Spalten verglichen.

Im Beispiel (siehe Listing 1) sind nur die Spalten object_name, object_type und last_ddl_time berücksichtigt. Wird eine Spaltenliste verwendet, so muss sie auch die Indexspalte enthalten.

```
CONN / as sysdba
SET TIMING ON
BEGIN
  DBMS_COMPARISON.create_comparison(
    comparison_name      => 'object1_scott_object2_system',
    schema_name          =>
```

```
'scott',
  object_name      => 'object1',
  -- Basistabelle
  dblink_name      => 'system_xe',
  remote_schema_name => 'system',
  remote_object_name => 'object2',
  -- Vergleichstabelle
  column_list      => 'object_name,object_type,object_id,last_ddl_time');
END;
Abgelaufen: 00:00:09.40
```

Listing 1

Wichtig ist hierbei:

- Innerhalb derselben Datenbank ist kein weiterer Vergleich mit demselben Namen möglich, auch nicht von einem anderen User.
- Nur der Ersteller des Vergleichstemplates darf den Vergleich durchführen und ihn löschen. Insofern sollte man als DBA die Rechte an diesem Package eher zurückhaltend vergeben.
- Wenn die Indizes der Tabellen nicht im selben Schema gespeichert sind, kann man Index-Schema und -Namen über die Parameter `index_schema_name` und `index_name` gesondert angeben. Auch hier sind die Index-Spalten in die Spaltenliste mit aufzunehmen.
- Der Parameter `dblink_name` muss für Vergleiche innerhalb der Datenbank auf NULL gesetzt sein.

DBMS_COMPARISON.compare

Diese Funktion prüft die Objekte auf Unterschiede. Dabei sind die Datensätze einer Tabelle automatisch in Bereiche, sogenannte Buckets, aufgeteilt. Die Zahl der Buckets nimmt mit der Größe des Objekts zu (maximal 1000). Die `ORA_HASH`-Funktion erzeugt einen Hash-Wert pro Reihe, und am Schluss pro Bucket, und vergleicht diesen mit dem Wert des korrespondierenden Buckets des anderen Objekts. Falls Unterschiede auftreten, werden die Buckets weiter unterteilt und gesannt. So erfolgen sogar bei einem Vergleich von zwei Exemplaren der emp-Tabelle, die einen einzigen Unterschied aufweisen,

vier Scans. Die Funktion `compare` gibt zwei Werte zurück:

- Die aktuelle Scan-ID über einen OUT-Parameter vom Typ `comparison_type`, ein Record mit fünf Feldern, von denen nur das `Scan_id`-Feld gefüllt ist
- Einen Boolean-Parameter, wobei TRUE bedeutet, dass die Objekte übereinstimmen

Der Parameter `perform_row_dif` bestimmt, ob die beiden Objekte zeilenweise verglichen werden. Im Beispiel unten fehlt er zunächst, um Zeit zu sparen. Sein Defaultwert ist FALSE, das heißt, es wird nur überprüft, ob Unterschiede vorliegen oder nicht. Treten Differenzen auf, so wird in einem zweiten Durchlauf der Parameter auf TRUE gesetzt. Eine Abfrage der Views `user_bzw. dba_comparison_scan_summary` liefert die Zahl der Unterschiede (siehe Listing 2).

```
DECLARE
  v_count          NUMBER;
  v_gleich         BOOLEAN;
  scan_rec         DBMS_COMPARISON.comparison_type; -- Record mit 5 Feldern
BEGIN
  v_gleich := DBMS_COMPARISON.compare(
    ,object1_scott_object2_system',
    scan_info => scan_rec); -- OUT-Parameter
  IF v_gleich = TRUE THEN
    DBMS_OUTPUT.PUT_LINE('Scan-Nr: '||scan_rec.scan_id);
    DBMS_OUTPUT.PUT_LINE('Die Objekte stimmen überein');
  ELSE
    v_gleich := DBMS_COMPARISON.compare(
    ,object1_scott_object2_system',
    scan_info => scan_rec ,
    perform_row_dif => TRUE);
    SELECT current_dif_count
    INTO v_count FROM user_comparison_scan_summary
    WHERE scan_id = scan_rec.scan_id;
    DBMS_OUTPUT.PUT_LINE('Scan-Nr: '||scan_rec.
```

```
scan_id);
    DBMS_OUTPUT.PUT_LINE('Es wurden '||v_count||' Unterschiede gefunden.');
```

```
END IF;
END;

Abgelaufen: 00:00:24.11
Scan-Nr: 1
Die Objekte stimmen überein
```

Listing 2

Jetzt werden ein paar Unterschiede eingebaut. Dann dauert das Ganze deutlich länger (siehe Listing 3).

```
UPDATE scott.object1 SET object_name = 'A', object_type = 'B', last_ddl_time = sysdate
WHERE MOD(object_id, 13079) = 0;
DELETE scott.object1 WHERE MOD(object_id, 13083) = 0;
INSERT INTO scott.object1 (object_id, object_name) VALUES (1000008, 'Max');
INSERT INTO scott.object1 (object_id, object_name) VALUES (1000009, 'Moritz');
COMMIT;
```

```
Scan-Nr: 84
Es wurden 154 Unterschiede gefunden.
Abgelaufen: 00:06:12.28
```

Listing 3

Über die Views `user_comparison_row_dif` erhält man Informationen darüber, für welche Index-Werte sich die Datensätze in den zwei Tabellen unterscheiden beziehungsweise wo Datensätze fehlen. Ein doppelter Eintrag „vorhanden“ für einen Index-Wert weist auf unterschiedliche Inhalte der Datensätze hin. Welche Werte hier abweichen und in welchen Spalten die Unterschiede liegen, ist allerdings mit `DBMS_COMPARISON` nicht direkt zu ermitteln (siehe Listing 4).

```
SELECT index_value,
CASE WHEN local_rowid IS NULL THEN ,fehlt' ELSE ,vorhanden' END in_basis,
CASE WHEN remote_rowid IS NULL THEN ,fehlt' ELSE ,vorhanden' END in_vergleich
FROM user_comparison_row_dif
WHERE comparison_name = UPPER(object1_scott_object2_
```

```

system')
AND status = ,DIF'
ORDER BY TO_NUMBER(index_value);
→
INDEX_VALUE    IN_BASIS
IN_VERGLEICH
-----
13079 vorhanden      vorhanden
13083 fehlt        vorhanden
26158 vorhanden      vorhanden
....

```

Listing 4

DBMS_COMPARISON.converge

Das Synchronisieren der Objekte erfolgt mittels der Prozedur DBMS_COMPARISON.converge.

Voraussetzung ist natürlich die Vergabe der benötigten Insert-, Update- und Delete-Rechte an den jeweiligen Tabellen. Das geht auch bei großen Tabellen recht schnell.

Man übergibt den Namen des Vergleichs, die beim Vergleich zurückge-

gebene Scan_id und einen Parameter namens converge_options, der entscheidet, ob die Vergleichstabelle oder die lokale Tabelle überschrieben wird. Der Defaultwert ist cmp_converge_local_wins, das heißt, die Werte der lokalen Tabelle werden auf die Vergleichstabelle übertragen. Im Beispiel wird die Richtung umgekehrt und die Basistabelle überschrieben (siehe Listing 5).

```

DECLARE
  scan_rec  DBMS_COMPARISON.
  comparison_type;
BEGIN
  DBMS_COMPARISON.converge(
    comparison_name =>
    'object1_scott_object2_system',
    scan_id => 84,
    scan_info =>
    scan_rec,
    converge_options =>
    DBMS_COMPARISON.cmp_converge_
    remote_wins);
  DBMS_OUTPUT.PUT_LINE(scan_

```

```

rec.loc_rows_merged
                                ||' Zei-
len in der Basistabelle über-
schrieben');
  DBMS_OUTPUT.PUT_LINE(scan_
rec.loc_rows_deleted
                                ||' Zei-
len in der Basistabelle ge-
löscht');
  DBMS_OUTPUT.PUT_LINE(scan_
rec.rmt_rows_merged
                                ||' Zei-
len in der Vergleichstabelle
überschrieben');
END;
→
152 Zeilen in der Basistabelle
überschrieben
2 Zeilen in der Basistabelle
gelöscht
0 Zeilen in der Vergleichsta-
belle überschrieben
Abgelaufen: 00:00:01.64

```

Listing 5



Die ORDIX AG ist ein etablierter, unabhängiger IT-Dienstleister. Unser Leistungsspektrum umfasst Beratung, Entwicklung, Service, Training und Projektmanagement im Umfeld Datenbanken, Unix, Windows und JEE.

Das zeichnet ORDIX aus

- Interessante Projekte bei namhaften Kunden
- Breites persönliches Aufgabenspektrum
- Ausgeprägtes, internes Know-how-Netzwerk
- Stetige Weiterbildung, persönliche Förderung
- Attraktive Vergütung, Firmen-PKW
- Soziale Absicherung

Interesse?

Dann senden Sie Ihre Bewerbung mit Angabe Ihrer Gehaltsvorstellung und dem gewünschten Einstellungstermin an:

ORDIX AG, Christoph Lafeld
 Westernmauer 12 - 16
 33098 Paderborn
 Tel.: 05251/1063-0, personal@ordix.de
 Weitere Infos unter www.ordix.de



Oracle Senior Consultant

Ihre Aufgaben

- Administration, Konfiguration und Betriebssupport bei komplexen Datenbanklösungen
- Anpassungen bestehender und Implementierung neuer IT-Systeme (Change Management)
- Monitoring bestehender IT-Systeme (Operations)
- Backup/Recovery- und Hochverfügbarkeitslösungen
- Migrationen
- Referententätigkeit im Seminarumfeld

Ihre Qualifikationen

- Sehr gute Oracle-Kenntnisse (Unix/Windows)
- Gute Unix-/Windows-Kenntnisse
- Mindestens 5 Jahre Erfahrung in der Administration von Oracle Datenbanklösungen
- Perfektes Deutsch in Wort und Schrift
- Uneingeschränkte Reisebereitschaft innerhalb Deutschlands
- Von Vorteil: Kenntnisse eines anderen RDBMS (Informix/DB2/SQL Server)

DBMS_COMPARISON.drop_comparision und DBMS_COMPARISON.purge_comparision

Die Prozedur DBMS_COMPARISON.drop_comparision löscht einen Vergleich inklusive des Templates. Wenn man dieses weiter benutzen will, weil der Vergleich in regelmäßigen Abständen durchzuführen ist, kann man alternativ mittels DBMS_COMPARISON.purge_comparision nur die Einträge in den zugehörigen Views löschen. Optional lässt sich hier auch eine bestimmte scan_id oder ein Zeitpunkt angeben, zu dem die Daten gelöscht werden sollen.

```
BEGIN
  DBMS_COMPARISON.purge_
  comparison(,object1_scott_ob-
  ject2_system');
END;
```

Vergleich mit anderen Verfahren

Eine der vor 11g zur Verfügung stehenden Möglichkeiten, zwei Tabellen mit Oracle-eigenen Mitteln zu vergleichen, ist der von Tom Kyte beschriebene Einsatz von Set-Operatoren [2]. Auf das Beispiel bezogen ergibt das folgende Abfrage (siehe Listing 6):

```
SELECT object_id, object_name,
  object_type, last_ddl_time,
  CASE COUNT(src1) WHEN 0 THEN
  ,fehlt' ELSE ,vorhanden' END
  in_basis,
  CASE COUNT(src2) WHEN 0 THEN
  ,fehlt' ELSE ,vorhanden' END
  in_vergleich
FROM
  (SELECT a.object_id, a.object_
  name, a.object_type, a.last_
  ddl_time,
  1 src1, TO_NUMBER(null) src2
  FROM scott.object1 a
  UNION ALL
  SELECT b.object_id, b.object_
  name, b.object_type, b.last_
  ddl_time,
  TO_NUMBER(null) src1, 2 src2
  FROM system.object2@system_xe b)
GROUP BY object_id, object_
  name, object_type, last_ddl_
  time
HAVING COUNT(src1) <>
  COUNT(src2)
ORDER BY object_id;
→
OBJECT_ID      OBJECT_NAME
```

```
OBJECT_TYPE    LAST_DDL_TIME
IN_BASIS       IN_VERGLEICH
13079 A         B           03.03.09
vorhanden     fehlt
13079 EXU8COL_TTS_UNUSED_COL
VIEW 15.10.07      fehlt
vorhanden
13083 EXU8COEU      VIEW
15.10.07      fehlt      vorhanden
26158 A         B           03.03.09
vorhanden     fehlt
.....
230 Zeilen ausgewählt.
Abgelaufen: 00:00:30.61
```

Listing 6

Die Abfrage erfolgt im Vergleich zu DBMS_COMPARISON.compare recht zügig und hat auch den Vorteil, dass man genau ersehen kann, welche Werte differieren. Für den manuellen Abgleich – auch über einen Datenbank-Link – ist diese Methode also durchaus interessant.

Das Überschreiben kann man dann mit einem Merge, gefolgt von einem Delete-Befehl, realisieren. Das dauert allerdings bei großen Tabellen länger als die Converge-Prozedur und erfordert einiges an fehleranfälliger Handarbeit (siehe Listing 7).

```
MERGE INTO scott.object1 o1
USING system.object2@system_xe
o2
ON (o1.object_id = o2.object_
id)
WHEN MATCHED THEN      UP-
DATE SET o1.object_name =
o2.object_name, o1.object_type
= o2.object_type, o1.last_ddl_
time = o2.last_ddl_time
WHEN NOT MATCHED THEN
INSERT (owner, object_name,
subobject_name, object_id,
data_object_id, object_type,
created, last_ddl_time, time-
stamp, status, temporary, gene-
rated, secondary)
VALUES (o2.owner, o2.object_
name, o2.subobject_name,
o2.object_id, o2.data_object_
id, o2.object_type, o2.created,
o2.last_ddl_time, o2.timestamp,
o2.status, o2.temporary,
o2.generated, o2.secondary);

1000000 Zeilen integriert.
Abgelaufen: 00:01:45.96

DELETE FROM scott.object1 WHE-
```

```
RE object_id IN
(SELECT object_id FROM scott.
object1 MINUS
SELECT object_id FROM system.
object2@system_xe);
```

```
2 Zeilen wurden gelöscht.
Abgelaufen: 00:00:03.39
COMMIT;
```

Listing 7

Fazit

DBMS_COMPARISON bietet eine zuverlässige, wenn auch nicht besonders schnelle Möglichkeit, Tabellen oder auf einzelnen Tabellen basierende Views auf inhaltliche Unterschiede zu überprüfen. Der Einsatz des Packages ist besonders dann von Vorteil, wenn man den Vergleich automatisieren und die Tabellen- bzw. Spaltennamen dynamisch übergeben will. Mit den vor 11g vorhandenen Methoden ist man in diesem Fall auf dynamisches SQL angewiesen, was die Angelegenheit komplex und langsam gestaltet.

Quellen

- [1] <http://dgielis.blogspot.com/2006/01/compare-2-oracle-schemas.html>
- [2] <http://www.oracle.com/technology/oramag/oracle/05-jan/o15asktom.html>
- [3] http://www.idevelopment.info/data/Oracle/DBA_scripts/Database_Administration/dba_compare_schemas.sql
- [4] http://tonguc.wordpress.com/2008/03/06/10gs-ora_hash-function-to-determine-if-two-oracle-tables-data-are-equal/
- [5] http://asktom.oracle.com/pls/asktom/f?p=100:11:0:::P11_QUESTION_ID:41727263557318
- [6] http://www.psoug.org/reference/dbms_rectifier_diff.html
- [7] http://www.psoug.org/reference/dbms_comparison.html
- [8] http://technology.amis.nl/blog/2420/dbms_comparison-to-compare-and-synchronize-tables-new-in-oracle-11g
- [9] http://tonguc.wordpress.com/2008/10/04/DBMS_COMPARISON-supplied-package-after-11g-release-1/
- [10] <http://robertgfreesman.blogspot.com/2007/12/11g-new-feature-dbmscomparison.html>
- [11] Andrea Held, Oracle Database 11g – neue Features, Hanser-Verlag, 2008

Kontakt:

Dr. Elke Fritsch
e.fritsch@muniqsoft.de

ECM-Performance und Oracle

Sebastian Wenzky, Digital Media Center, ECM Business Unit

Dieser Artikel beschäftigt sich mit grundlegenden ECM-Performance-Kriterien am Beispiel des Open Source Enterprise Systems Alfresco (2.2.1). Die Kriterien können einen enormen Einfluss durch benutzerspezifische Anwendungsszenarien auf die darunterliegende Systemlandschaft haben. Dieser Artikel stellt zunächst ganzheitliche Diagnose- und Optimierungsansätze zur Diskussion. Anschließend erfolgt die Fragestellung, in welchem Umfang solche Ansätze mit einer Oracle-Datenbank weiterverfolgt werden können.

Enterprise Content Management – wofür steht dieser Begriff eigentlich? Die Strukturierung von Daten in einer wohldefinierten Art und Weise gehört sicherlich zu einem der Kernaspekte von ECM, ebenso wie umgesetzte Prozesse, die auf Basis des eingestellten Inhaltes bestimmte Regeln oder Verhalten ausführen. Das Versionieren von Dateien mit dem nachhaltigen Anreichern von Meta-Informationen gehört als implizite Datenverwaltung zu jedem ECM-/DMS-System zweifelsfrei dazu. In der Regel ist ein nicht-funktionaler Aspekt sicherlich nicht so häufig vertreten: die Performance.

Typische Probleme

Unternehmen mit vielen Informationsschichten und -strukturen haben sowohl ein logistisches als auch ein infrastrukturelles Problem. Das System muss in der Lage sein, die Strukturen und Prozesse nach Unternehmensrichtlinien abzubilden. Dieser erste Schritt wird immer wichtiger, um keine unnötige Arbeit durch Suchen von Dokumenten oder durch unnötiges Wiederholen von Prozessen durchführen zu müssen. Ein bereits vorhandenes ECM-System verfügt häufig über diese notwendige Flexibilität und unterstützt somit die Unternehmensabläufe aktiv. Jedoch stellt die Performance nach nur kurzer Zeit ein wesentliches Problem dar. Dabei galt das ECM zur Einführungsphase als extrem schnell, hochskalierbar und war zu Beginn der produktiven Phase auch als entsprechend performant ausgewiesen. Für den unternehmensweiten Einsatz ist diesbezüglich eine Oracle Datenbank herangezogen worden. Doch inner-

halb eines Jahres ist das ECM-System regelrecht zum Erliegen gekommen.

Mittlerweile befinden sich drei Millionen Dokumente im ECM-System. Jedes Dokument besitzt eine Vielzahl unterschiedlicher Attribute (dateispezifisch, fachlich). Über 23 Millionen Datensätze existieren in der Tabelle ALF_NODE_PROPERTIES, in der die Dokumenten-Eigenschaften gespeichert sind. 23 Millionen Datensätze können durchaus ein potenzielles Problem darstellen. Um jedoch eine genauere Aussage treffen zu können, sind fortlaufende Tests notwendig (siehe „Wie weit sollte Denormalisierung reichen?“).

Die Problemstellung

Zurück zu den Unternehmen: Vor allem die zu langsame Übertragungsrates von Dokumenten über WebDAV in das ECM-System wird seitens des Unternehmens moniert. Die rund 300 Dateien mit einer zusammengefassten Größe von etwa zehn Megabyte benötigen per externem Zugriff über zehn Minuten. Die Zeit für eine Suchanfrage übersteigt deutlich die Anforderung von unter einer Sekunde. Hier ist die unternehmenseigene IT in vielen Fällen überfordert mit der Frage, auf welche Weise dieser Zustand entstehen konnte. Das ECM-System nutzt explizit viele benachbarte Systeme, um Anwendungsszenarien des Benutzers zu bearbeiten. Schauen Sie sich hierzu nur einmal Ihre eigene Systemlandschaft an. Das darin enthaltene Netzwerk aus Abhängigkeiten zwischen dem Stagesystem, Dienstprogrammen sowie System-Konfigurationen bildet ein komplexes Regelwerk. Vie-

le technische Ebenen gilt es daher zu analysieren, um konkretere Eindrücke des möglichen Performance-Problems zu erhalten. Dabei wird häufig vornehmlich der Hebel am ECM-System selbst angesetzt, obwohl nur ein kleiner Teil durch das ECM selbst bearbeitet wird.

Die Theorie

Eine weiterführende Analyse dieser komplexen Umgebung (Netzwerk, Storage, Betriebssystem, Antivirus) teilt das System in funktionale Blöcke ein. Diese Blöcke sind nach ihrem Systemeinfluss zu gewichten. Der Einfluss lässt sich durch partielle Tests zunächst grob ermitteln. Hierzu nimmt man einen Anwendungsfall (Beispiel: Hochladen von Dokumenten auf den Server) und testet die Bandbreite des Netzwerks sowie die dafür notwendige Übertragungszeit, um diese im ECM-System zu speichern. Anschließend schaltet man das Antivirus-Programm hinzu und misst erneut diesen Anwendungsfall. Die oft einfache Aussage, dass ein ECM-System der Grund des Performance-Problems darstellt, ist nur aus Sicht des Endbenutzers akzeptabel; im Hintergrund ist jedoch die soeben beschriebene Herangehensweise relevanter. Erst wenn diese Systeme als optimiert gelten, sollte überhaupt in Erwägung gezogen werden, das ECM-System zu tunen. Doch wie sollte eine Analyse vollzogen werden? Hier hilft ein verallgemeinertes Modell aus der Datenbankskalationswelt weiter, welches durch verschiedene Fragestellungen das auftretende Problem auf kleinere funktionale Bereiche des Systems eingrenzt:



Abbildung 1: Wesentliche Einflusskriterien bei verschiedenen Anwendungsfällen, gewichtet nach Häufigkeit der Nutzung durch konkrete Hauptfunktionen

- Die Frage nach dem *Was* deutet auf die Art des Vorfalls hin
- Die Konkretisierung des Vorfalls hilft bei den nächsten Fragestellungen bezüglich des räumlichen Auftretens
- Darauf aufbauend sollte ein Diagnose-Tool in der Lage sein, den zeitlichen Aspekt mit zu berücksichtigen, wann ein Performance-Problem im System anzutreffen ist
- Die nächste Fragestellung beschäftigt sich mit zusätzlichen Informationen wie der des Mehrbenutzerbetriebs
- Zum Schluss ist es sinnvoll zu priorisieren, welcher Grad von Abhängigkeit zwischen dem Unternehmen und einem performanten System besteht

Schwierigkeiten der Analyse

Abbildung 1 zeigt wesentliche Hauptfunktionen mit Unterfunktionen des eingesetzten ECM-Systems von Alfresco auf: Erfassung, Verwaltung, Ausgabe, Speichern sowie die Aufbewahrung sind zentrale Bestandteile dieses Systems.

Das Anwendungsszenario bestimmt die beteiligten Funktionalitäten, welche in Abbildung 1 dargestellt sind. Eine Abhängigkeit zwischen der ECM-Performance und den in Abbildung 1 dargestellten Funktionen gilt es daher festzustellen. Bevor deshalb eine tiefgreifende Analyse des ECM-Systems auf diesen Punkt hin erfolgen kann, muss herausgefunden werden, wel-

che Anwendungsszenarien seitens des Kunden im Fokus der Performance stehen. Diese sind in der Regel schon durch vorangegangene Analysen des Systems herausgefunden worden. Beispiel: Wird ein System unter ständiger Anforderung des Speicherns geführt, sollten vor allem das Storage-System sowie das Netzwerk – für die Übertragung – begutachtet werden. Einmal identifiziert, können Abschätzungen und Messungen in diese Richtung erfolgen. Die Fragepunkte aus dem Abschnitt „Die Theorie“ gilt es genauso als Gegenstand der Untersuchungen mit aufzunehmen wie den Workload der Systemkomponenten, welche direkten Einfluss auf die Funktionalitäten besitzen. Vor allem die stärker gewichteten Unterfunktionen spielen bei einem konkreten Anwendungsfall eine große Rolle, da der Hauptteil der Arbeit auf ihnen lastet.

Ein Beispiel aus der Praxis

Als allgemeiner Anwendungsfall dient das Einstellen von Dateien in ein ECM über diverse Schnittstellen. Diese nach außen doch recht übersichtliche Aufgabenstellung mündet in ein komplexes Regelwerk von Prozessen, die gestartet werden müssen, sobald das ECM-System die Handlung übernimmt.

Abbildung 2 zeigt die wichtigsten Aktionen seitens Alfresco, um eine Datei ablegen zu können. Die Versionie-

Zeitlicher Verbrauch	
Objekt erstellen (mit Eigenschaften/ Properties)	6%
Versionierung	62 %
Rechteüberprüfung	8,5%
Andere kopierrelevante Maßnahmen	~ 25%

Abbildung 2: Prozentuale Aufteilung der Aktionen beim Erstellen des Objekts

ung beansprucht dabei einen Großteil der Zeit. Versionierung im ECM-Bereich ist definiert durch die fortlaufende Sicherstellung eines Dokumentenzustands vor einer Veränderung. Jede Veränderung eines Dokuments hat zur Folge, dass ein komplett neues Dokument einschließlich Eigenschaften erzeugt wird, welches eine „neuere“ Version des Ausgangsdokuments darstellt.

Eine mögliche Optimierung dieses Use Cases könnte darin bestehen, die Versionierung zu verbessern. Die Systemlandschaft soll seitens des Kunden unberührt bleiben. Eine Optimierung kann demnach nur im ECM-System erfolgen, obwohl das E/A-System oder das Netzwerk eine ebenfalls entscheidende Rolle bei diesem Anwendungsfall spielen. Eine fachliche Anforderung seitens des Kunden verlangt weiterhin, dass kein Dokument überschrieben werden kann.

Optimierungsmöglichkeit: Zeitliche Verschiebung der Versionierung

Als Gegenstand der Lösung dient der zeitliche Aspekt einer Versionierung. Das bedeutet nichts anderes, als dass die erste Versionierung bei der ersten Durchführung einer Änderung angewandt wird. Die Optimierung von diesem Punkt erzielte in der Praxis eine Zeitersparnis von über 50 Prozent; der zeitliche Rahmen verringerte sich von sechs auf drei Minuten für das Einstellen von elf Megabyte über das Netz. Drei Minuten sind für diese Dokumentenanzahl immer noch zu hoch. An dieser Stelle kann davon ausgegangen werden, dass das ECM-System für dieses Anwendungsszenario optimiert ist. Eine weitere Optimierung nach Abbildung 2 gilt vom Kosten-Nutzen-Faktor her als nicht durchführbar. Denn hier tritt die Gewichtung auf, welche zuvor diskutiert worden ist: Zunächst gilt es zu versuchen, die größtmöglichen identifizierten Performance-Blöcke zu optimieren, dann das ECM-System. Ersichtlich wird dies, wenn das Netzwerk näher betrachtet wird. Analysen des Serversystems und des Netzwerks – beispielsweise mit NetIO – zeigten einen verminderten Datendurchsatz. Hier konnte ein falscher Treiber iden-

tifiziert werden, der zu sehr langsamen Übertragungsraten von Dokumenten auf den Server führte.

Wo bleibt die Datenbank?

An diesem Punkt stellt sich die Frage, auf welche Weise die Oracle-Datenbank unterstützend wirken kann. Vielleicht sollte die Frage auch folgendermaßen lauten: Gilt die Oracle-Datenbank überhaupt als eine Möglichkeit, Alfresco zu tunen oder einer Diagnose zu unterziehen? Am Beispiel von Alfresco spielen drei wesentliche technisch-orientierte Aspekte eine Rolle:

- *Dokumentenspeicherung*
Die Dokumente werden nicht zentral in der Datenbank abgelegt. Es erfolgt daher keine Verwendung von Blobs. Metadaten sowie Verwaltungsinformationen sind jedoch in einer Oracle Datenbank auffindbar.
- *ORM*
Hibernate generiert sämtliche SQL-Statements, die an das Oracle-System transportiert werden.
- *Caching*
Als seien der Platten-, der System- und der SGA-Cache von Oracle nicht genug an Zwischenspeicherung, legt Alfresco intern zwei funktional voneinander unabhängig agierende Caches an. Der Hibernate-Cache dient als zentrales Objektrepository – die Verbindungslinie zwischen Oracle und Alfresco. Der zweite Cache, in dem häufig genutzte Userdaten zwischengespeichert werden, wird von Alfresco direkt verwaltet.
Da die Dokumentenspeicherung direkt im Filesystem erfolgt, die Referenzierungen in der Datenbank jedoch vorgehalten werden, besteht hier kein Ansatz, der als Performanceaspekt betrachtet werden kann. Das „chaining“ des Caching führt zu einem komplexen aber hocheffizienten Regelwerk an Zwischenspeichern, welches keinen wesentlichen Ansatz einer Optimierung mit vertretbarem Aufwand bietet. Hibernate optimiert nach bestem Gewissen Statements, die dann an Oracle weitergegeben werden.

Dient etwa die Oracle-Datenbank nur noch als reiner Metadaten-Platz?

Oracle als Diagnosehelfer?

Es wird bewusst eine Datenbank-Unabhängigkeit seitens Alfresco suggeriert. Doch für die Datenbank, Teil eines jeden ECM-Produkts, gelten die im vergangenen Abschnitt aufgezeigten Punkte nur begrenzt. Alle drei für sich abgeschlossenen Punkte – die Dokumentenspeicherung, das ORM und das Caching – müssen früher oder später mit der Oracle Datenbank synchronisiert werden. Interessant wird es also erst, wenn es herauszufinden gilt, wie viel Workload eine Oracle Datenbank produziert. Um eventuell herauszufinden, ob das ECM-System als Ablage- oder Leseort gehalten wird, bieten sich einige Performance-Views an sowie die Verwendung des Enterprise Managers von Oracle. Diese Views werden mit einfachen SQL-Statements angesprochen.

Wichtig ist der zeitliche Bezug zu den Daten: Hierzu ist die zeitliche Dimension ein wesentlicher Faktor, ohne den die Daten aus den Views wertlos erscheinen. Hierzu sollte der Administrator einen entsprechenden Job auf der Datenbank einrichten (einfach einzurichten über Oracle Enterprise Console), welche eine PL/SQL-Funktion triggert. Dieser PL/SQL-Block speichert in regelmäßigen Abständen Performance-Attribute wie die CPU- und die IO-Time des Systems in einem angemessenen Zeitverhältnis in einer eigens hierfür erstellten Performance-Tabelle (siehe Abbildung 3). Das Zeitverhältnis kann beispielsweise alle 15 Minuten ausgeführt werden. Man sollte mit vier Testzeitpunkten in der Stunde beginnen. Der zeitliche Nutzungsgrad von Oracle lässt sich dadurch genauer identifizieren. Dieser Ansatz verhält sich auch auf dem Produktivsystem deutlich besser als ein zentral eingeschaltetes Select-Auditing.

```
SELECT substr(SQL_TEXT, 0, 80)
|| ,... ' sqltext, USER_IO_
WAIT_TIME/1000000 || ,s' IO_
TIME, CPU_TIME/1000000 || ,s'
CPU_TIME, ELAPSED_TIME/1000000
```

SQLTEXT	IO_TIME	CPU_TIME	ELAPSED_TIME	DISK_READS	PERC	EXECUTIONS	TIME
1 insert into alf_node (version, store_id, uuid, transaction_id, node_deleted, ...	1091,850567s	11,1875s	1106,15405s	21703	1	32781	11:23:47 07-MRZ-2009
2 insert into alf_child_assoc (version, parent_node_id, type_qname_id, child_...	141,127877s	9,0625s	150,808269s	3321	1	32781	11:23:47 07-MRZ-2009
3 insert into alf_node_properties (node_id, qname_id, list_index, locale_id, ac...	116,449418s	12,484375s	130,533694s	2455	0,888917...	15601	11:23:47 07-MRZ-2009
4 select childassoc0_id as id7_ childassoc0_version as version7_ childassoc...	61,832406s	4,265625s	66,222389s	1212	1	60348	11:23:47 07-MRZ-2009
5 select childassoc0_id as col_0_0_ childassoc0_type_qname_id as col_1_0_...	17,248793s	134,28125s	157,636481s	16544	1	17347	11:23:47 07-MRZ-2009
6 select properties0_node_id as node1_0_ properties0_actual_type_n as a...	12,773239s	,296875s	13,031841s	284	1	2411	11:23:47 07-MRZ-2009
7 select nodeimpl0_id as id4_0_ dbaccesscol1_id as id14_1_ nodeimpl0_...	9,074382s	,109375s	9,116503s	232	1	258	11:23:47 07-MRZ-2009
8 select childassoc0_id as col_0_0_ childassoc0_type_qname_id as col_1_0_...	7,198377s	,5s	7,665874s	256	1	68	11:23:47 07-MRZ-2009
9 insert into alf_node_aspects (node_id, qname_id) values (:1, :2)...	5,107875s	,859375s	6,137391s	149	1	6938	11:23:47 07-MRZ-2009
10 insert into alf_transaction (version, server_id, change_txn_id, commit_time...	4,988147s	2,015625s	7,654474s	111	1	6495	11:23:47 07-MRZ-2009
11 select aspects0_node_id as node1_0_ aspects0_qname_id as qname2_0_...	4,453277s	,21875s	4,626613s	119	1	2410	11:23:47 07-MRZ-2009
12 insert into alf_node_assoc (version, source_node_id, target_node_id, type...	4,038152s	,390625s	4,481558s	120	1	1733	11:23:47 07-MRZ-2009

Abbildung 3: Zeitliche Beanspruchung von Systemkomponenten

```
|| ,s' ELAPSED_TIME, DISK_READS, PARSE_CALLS / executions perc, executions, (SELECT TO_CHAR(CURRENT_DATE, 'HH:MI:SS DD-MON-YYYY') FROM DUAL) TIME FROM v$sqlarea WHERE PARSING_SCHEMA_NAME = 'HEINRICH' ORDER BY USER_IO_WAIT_TIME DESC;
```

Das erste Statement dient dem Zwecke, sich einen groben Überblick über die vernetzten Systemkomponenten zu verschaffen. Die selektierten Attribute sprechen für sich, während die IO_TIME implizit von DISK_READS abhängt.

In Abbildung 3 ist eine hohe Auslastung vor allem in eine Richtung (schreibend) gut zu erkennen. Die Vorgängerwerte, welche 120 Minuten vorher gemessen worden sind, waren sehr ausgeglichen und mehr als 1000-fach kleiner. Hier liegt eine akute Spitze im System vor, indem deutlich mehr auf dem ECM-System gearbeitet wird. Die Tabelle ALF_NODE identifiziert ein Dokument oder einen Ordner, so dass auch deutlich vom Tabellennamen gezeigt werden kann, dass es sich hierbei um das Einstellen von neuen Dokumenten handelt. Jetzt stellt sich natürlich die Frage, ob diese Werte Auswirkungen auf die Systemperformance haben. Einen Überblick hierzu können wir durch folgendes Statement erhalten:

```
select SQL_TEXT, LOGON_TIME, WAIT_CLASS, BLOCK_GETS, CONSISTENT_GETS, PHYSICAL_READS,
```

```
BLOCK_CHANGES, CONSISTENT_CHANGES FROM (select SQL_ID, LOGON_TIME, WAIT_CLASS, BLOCK_GETS, CONSISTENT_GETS, PHYSICAL_READS, BLOCK_CHANGES, CONSISTENT_CHANGES from v$session a, v$sqlarea b WHERE a.USERNAME = 'HEINRICH' and a.PROGRAM = 'JDBC Thin Client' and a.sid = b.sid) a, v$sqlarea b where a.sql_id = b.sql_id(+);
```

In der Tat liegt ein Wartezustand seitens eines Clients vor, der im Connection Pool von Alfresco verwaltet wird. In Abbildung 4 erkennt man den aktuellen Wartezustand sowie das betroffene SQL-Statement wieder.

Dieser Zustand gilt als ein Anzeichen dafür, dass das darunterliegende IO-System ausgelastet scheint. Jedoch ist diese Sicht, aufgebaut aus dem zweiten SQL-Statement, eine Momentaufnahme. Viel wichtiger wäre eine Übersicht der gesamten bisherigen USER I/O-Wartezeiten.

```
SELECT WAIT_CLASS, TOTAL_WAITS, ROUND(100 * (TOTAL_WAITS / SUM_WAITS),2) PCT_WAITS, TIME_WAITED, ROUND(100 * (TIME_WAITED / SUM_TIME),2) PCT_TIME FROM ( SELECT WAIT_CLASS, TOTAL_WAITS, TIME_WAITED FROM V$SYSTEM_WAIT_CLASS WHERE WAIT_CLASS != 'Idle' ), ( SELECT SUM(TOTAL_WAITS) SUM_WAITS, SUM(TIME_WAITED) SUM_TIME FROM V$SYSTEM_WAIT_CLASS WHERE WAIT_CLASS != 'Idle' ) ORDER BY PCT_TIME DESC;
```

SQL_TEXT	LOGON_TIME	WAIT_CLASS	PHYSICAL_READS	BLOCK_GETS	CONSISTENT_GETS	BLOCK_CHANGES	CONSISTENT_CHANGES
1 insert into alf_node (version, store_id, ...	07.03.09	User I/O	3570	581581	3951633	318865	205
2 (null)	07.03.09	Idle	4488	744739	4969342	408755	92
3 (null)	07.03.09	Idle	17049	2566576	17100575	1408599	911
4 (null)	07.03.09	Idle	24488	954558	6291149	524028	349

Abbildung 4: Eine User-Session wartet gerade auf die Ausführung des Statements

Das Caching-Problem

Im Abschnitt „Wo bleibt die Datenbank?“ heißt es, dass Alfresco explizit sowie implizit in großem Maße aktuelle Objekte in verschiedenen Caches zwischenspeichert. Dadurch relativiert sich die Aussage bezüglich des Einsatzes von Performance-Tabellen von Oracle wieder etwas. Lesende Zugriffe sind nicht zwingend im DBMS-Cache abzuarbeiten, ergo können auch keine Einträge in der v\$sqlarea vorliegen. Sehr wohl jedoch können schreibende Zugriffe, die vor allem auf die Tabellen ALF_NODE und ALF_NODE_PROPERTIES einwirken, geloggt werden. Auditing seitens Alfresco, welches inhaltsbasierte CRUD-Funktionalitäten loggt, kann hierbei ebenfalls benutzt werden. Das Audit-System von Alfresco nimmt nur einen unwesentlichen Anteil an der System-Performance ein, wie sich in der Praxis gezeigt hat. Natürlich ist der Enterprise Manager eine weitere Möglichkeit, das System weitergehend zu analysieren.

Datenbank-Optimierung

Am Anfang dieses Artikels wurde ein häufiges Problem – die Performance von ECM-Systemen im produktiven Einsatz – aufgezeigt. Die darauf aufbauende Optimierungsmaßnahme hatte diesen Use Case um über 50 Prozent beschleunigt, basierend auf funktionalen Abhängigkeiten, dargestellt in Abbildung 1. Voraussetzung für solche Optimierungsmaßnahmen stellen dagegen nur Schritte dar, die kritischen Performanceeinfluss besitzen. Gemeint ist damit, dass eine Beschleunigung von 50 Prozent keineswegs die Lösung für das allgemeinere Performance-Problem sein kann. Optimiert man eine Komponente, welche nur über geringen Einfluss verfügt, ist der Gewinn, wie eben 50 Prozent, durch eine Optimierung in Frage gestellt. Oracle ist in vielen Fällen auch davon betroffen: Sowohl das Caching als auch die Filesystem-Speicherung von Dokumenten unterbinden de facto die zentrale Stelle eines solchen Systems, so dass Optimierungen in diese Richtung in den Hintergrund rücken. Interessant wird

dieser Aspekt hingegen erst wieder, sobald Millionen von Dokumenten mit unzähligen Eigenschaften und Versionen im System verwaltet werden müssen. Partitionierung ist dann sicherlich eine Möglichkeit, Optimierungen durchführen zu können.

Wie weit sollte Denormalisierung reichen?

Zu Beginn des Artikels ist eine ECM-Anwendung mit mehreren Millionen Datensätzen in einer Tabelle vorgestellt worden. Folgende Darstellung greift auf diese Betrachtungsweise zurück, um technische Ausmaße dieser Größenordnung bezüglich Alfresco vorzustellen.

Abbildung 5 gibt ein Beispiel der Größe einzelner Indizes von Tabelleneinträgen in Alfresco wieder. Diese Standard-Alfresco-Installation beinhaltet gerade einmal 130.000 Dokumente mit 300.000 Properties. Die Akkumulation des Indexspeichers allein der Tabelle ALF_NODE_PROPERTIES führt dennoch zu einer theoretischen DBMS-Cachegröße von über 600 Megabyte.

Im aufgezeigten Beispiel mit 23 Millionen Properties wächst der Index allein dadurch auf über drei Gigabyte an. Diese Daten werden nicht einmal ansatzweise im DBMS-Cache gehalten. An dieser Stelle sollte das Zugriffsverhalten der ECM-Nutzer genauer untersucht werden. In vielen Fällen ist der Arbeitsbereich auf nur wenige Dokumente bzw. Bereiche eines ECM-Systems fokussiert. Hieraus bilden sich dann die Hits im DBMS-Cache nach

dem MRU-Prinzip ab. Ergo finden sich diese Indices auch länger in den Caches wieder.

Durch eine starke Denormalisierung innerhalb der Alfresco Datenbank, indem auf redundante Informationen verzichtet wird, sichert man sich zwar Verwaltungsaufwand im ECM-System. Die Nachteile sind jedoch der physische Verwaltungsaufwand durch das Setzen von Constraints in der Datenbank und das Halten verschiedenster Indizes im Speicher. Der Denormalisierungsgrad von Alfresco 1.4 ist noch nicht so stark ausgeprägt gewesen wie in der Version 2.2.1, so dass beispielsweise die QNames (bilden eine Art Datentypen in Alfresco ab) in der ALF_NODE-Tabelle mit gespeichert worden sind.

Ein Ansatzpunkt zur Minderung der Zeilenanzahl ist das Transponieren von Zeilen in Spalten der ALF_NODE_PROPERTIES-Tabelle. Ziel ist es hierbei, Properties, die als Zeilen abgelegt sind, in Spalten zu transformieren. Der Vorteil besteht darin, dass eine deutlich reduzierte Anzahl an Zeilen in der Tabelle zur Abspeicherung der Dokumenteneigenschaften benötigt wird. Weiterhin ist das Selektieren der Properties auf eine Zeile sehr feingranular, ergo benötigt das DBMS weniger Indexspeicher. Natürlich tauscht man jeden Vorteil gegen einen Nachteil ein. Hibernate ist nicht in der Lage, selektiv Spalten zu akquirieren, die ad-hoc einer Tabelle hinzugefügt werden. Hier müsste ein Workaround erstellt werden. Weiterhin wird die Verlinkung einer Zeile sehr groß und es entstehen Migrated

Rows in großem Maße, da viele Spalten nicht beim ersten Einstellen ausgefüllt werden. Auch eine deutlich erhöhte Anzahl an Chained Rows kann die Folge dieser Umstrukturierung sein.

Fazit

Optimierungen im ECM-Bereich sind auf vielseitige Art und Weise zu erreichen, wie das Beispiel bezüglich der Versionierung zeigt. Es gilt jedoch, die Diagnose- und Tuning-Maßnahmen auf die größten Blöcke zu reduzieren, da man sonst vor beliebig vielen und auch komplexen Analyse- und Tuning-Möglichkeiten steht. Effizienz heißt hier die Devise.

Oracle ist hier in vielen Fällen nicht die zentrale Komponente, an der eine Performance-Optimierung durchführbar ist. Erst bei einer großen Anzahl von Objekten, die ein ECM-System wie Alfresco verwaltet, spielt die Datenbank eine deutlich zentralere Rolle. Aktuell ist es jedoch so, dass mehr Kompetenz in das ECM-System übergehen wird, sodass selbst der Einsatz einer Oracle Datenbank in Frage gestellt werden könnte.

Damit ein Unternehmen langfristig mit einem ECM-System arbeiten kann, gilt es, vor der Integrierung in dessen Firmenstruktur die Anwendungsfälle kritisch zu prüfen. Dies bedeutet, dass zunächst eine reine Standard-Installation von Alfresco auf Performance-Probleme hin untersucht werden sollte. Getestet werden sollte, ob relevante Use Cases, wie das Einstellen von tausenden Dokumenten, in mehreren parallelen Sessions den gestellten Anforderungen entsprechen. Ist dies der Fall, so kann von einem soliden Grund-System ausgegangen werden. Durch eine agile Vorgehensweise wie die Anpassung des ECM-Systems an die Bedürfnisse des Unternehmens ist es dann möglich, aktuelle Messergebnisse mit dem Grundsystem zu vergleichen, um frühzeitig Performance-Probleme erkennen zu können.

Kontakt:

Sebastian Wenzky
sebastian.wenzky@dmc.de

	INDEX_SIZE	INDEX_NAME	TABLE_NAME	COLUMN_NAME
1	208 MB	SYS_C009862	ALF_NODE_PROPERTIES	LIST_INDEX
2	208 MB	SYS_C009862	ALF_NODE_PROPERTIES	QNAME_ID
3	208 MB	SYS_C009862	ALF_NODE_PROPERTIES	NODE_ID
4	208 MB	SYS_C009862	ALF_NODE_PROPERTIES	LOCALE_ID
5	128 MB	SYS_C009814	ALF_CHILD_ASSOC	TYPE_QNAME_ID
6	128 MB	FK_ALF_NPROP_N	ALF_NODE_PROPERTIES	NODE_ID
7	128 MB	SYS_C009814	ALF_CHILD_ASSOC	CHILD_NODE_NAME
8	128 MB	SYS_C009814	ALF_CHILD_ASSOC	PARENT_NODE_ID
9	128 MB	SYS_C009814	ALF_CHILD_ASSOC	CHILD_NODE_NAME_CRC
10	104 MB	FK_ALF_NPROP_LOC	ALF_NODE_PROPERTIES	LOCALE_ID
11	96 MB	FK_ALF_NPROP_QN	ALF_NODE_PROPERTIES	QNAME_ID
12	96 MB	SYS_C009845	ALF_NODE	STORE_ID
13	96 MB	SYS_C009845	ALF_NODE	UUID

Abbildung 5: Beispielindexgrößen bei rund 130.000 Dokumenten und 300.000 Properties

Tipps und Tricks aus Gerds Fundgrube

Heute: Wie unterbreche ich in Forms einen Loop?

Gerd Volberg, OPITZ CONSULTING GmbH

In Oracle Forms ist es nicht möglich, eine PL/SQL-Routine zu unterbrechen. Während eines Loops lassen sich zum Beispiel keine weiteren Forms-Trigger abarbeiten, da der Loop exklusiv arbeitet.

Der folgende Showcase zeigt, wie mit einem einfachen Timer und einem beliebigen Hotkey ein Loop abgebrochen werden kann. Als Hotkey wird hier exemplarisch der KEY-EXIT benutzt. Somit kann über Strg-Q ein Abbruch initiiert werden. In diesem Beispiel werden zwei Buttons und ein Text-Item benötigt:

- Block CONTROL ohne Datenbank-Verwendung
- Button BT_LOOP_Start und BT_LOOP_EXIT
- Text-Item TI_COUNTER, numerisch



Abbildung 1: Demonstration des Loop-Abbruch-Verfahrens

Eine einfache Schleife, die zum Beispiel ein paar Sekunden für die Abarbeitung benötigt und nicht unterbrochen werden kann, wird mittels folgender Schritte in eine unterbrechbare Schleife verwandelt:

```
FOR I IN 1 .. 50
LOOP
  DBMS_LOCK.Sleep (0.1);
END LOOP;
```

Die folgenden Trigger werden dazu auf Form-Ebene benötigt:

- WHEN-NEW-FORM-INSTANCE

```
DEFAULT_VALUE ('0', 'GLOBAL.Loop_Counter');
DEFAULT_VALUE ('FALSE', 'GLOBAL.Cancel_Loop');
```

- WHEN-TIMER-EXPIRED

```
IF :GLOBAL.Cancel_Loop = 'FALSE' THEN
  Main_Loop;
END IF;
```

- BT_LOOP_START : WHEN-BUTTON-PRESSED

```
Main_Loop;
```

- BT_LOOP_EXIT : WHEN-BUTTON-PRESSED

```
DO_KEY ('exit_form');
```

- KEY-EXIT

```
:GLOBAL.Cancel_Loop := 'TRUE';
```

Die zentrale Prozedur MAIN_LOOP steuert den abbrechbaren Loop nun über globale Variablen, die den Loop-Fortschritt zwischenspeichern, und über die lokalen Variablen V_Max und V_Iteration, die den Loop-Endwert und die Anzahl der Loop-Schritte definieren, die zwischen zwei Timer-Fenstern durchlaufen werden:

```
PROCEDURE Main_Loop IS
  V_Iteration NUMBER := 2;
  V_Max NUMBER := 50;
  V_Timer Timer;
```

```
BEGIN
  IF to_number (:GLOBAL.Loop_Counter) < V_Max THEN
    :GLOBAL.Cancel_Loop := ,FALSE';
    FOR I IN to_number (:GLOBAL.Loop_Counter) + 1 .. to_number (:GLOBAL.Loop_Counter) + V_Iteration
    LOOP
      DBMS_LOCK.Sleep (0.1);
    END LOOP;
    :GLOBAL.Loop_Counter := :GLOBAL.Loop_Counter + V_Iteration;

    :Control.TI_Counter := to_number (:GLOBAL.Loop_Counter);
    Synchronize;
    V_Timer := CREATE_TIMER (,TEST_LOOP', 10, NO_REPEAT);
  END IF;
END;
```

Der Loop arbeitet eine feste Anzahl von Iterationen ab und startet dann einen Timer, der ein paar Millisekunden später die eigene Routine erneut startet. Durch dieses Entzerren zwischen PL/SQL-Loop und Timer gibt man Oracle Forms die Möglichkeit, sämtliche Events abzuarbeiten, die in der Zwischenzeit aufgelaufen sind. In unserem Fall wird der Hotkey dazu benutzt, eine globale Abbruch-Variable zu setzen, die die Weiterarbeit des Loops beendet.

Dies ist der einfachste Fall eines Loop-Abbruchs. Komplexere Loops, die auf Basis eines Select-Statements arbeiten, sind ebenfalls denkbar und müssten entsprechend überarbeitet werden.

Kontakt:

Gerd Volberg
talk2gerd.blogspot.com

Oracle Exadata – ein Blick hinter die Kulissen

Am 18. Juni 2009 fand in Köln die DOAG SIG Database zum Thema „Exadata“ statt.

Nach der Begrüßung durch DOAG-Beirat Thomas Tretter stellte Frank Schneede von Oracle das Konzept der Exadata-Technologie vor und gab einen guten Überblick über die Arbeitsweise der neuen Hintergrundprozesse. Dagmar Förster und Malthe Griesel von der Paragon Data GmbH berichteten in ihrem Vortrag „Exadata – Proof of Concept mit OLTP und Data-Warehouse-DB“ über das Proof of Concept, das in diesem Jahr stattfand. Dabei beschrieben sie die heutige Ausgangssituation (sechs Knoten, Oracle RAC 10.2.03, NetAppFiler) und stellten die Vorbereitung, den Ablauf und die Nachbereitung des Tests dar. Die Ergebnisse wurden angeregt diskutiert.

Im Vortrag „Exadata Expedition oder die HP Oracle Database Machine aus Anwendersicht“ beschrieb Rainer Marekvia von Oracle die technische Betriebsicht der Database Machine. Dabei wurden Fragen wie Backup und Recovery, Integration in bestehende Netz- und Speicher-Topologien, Hochverfügbarkeits-Konzepte und Konfigurationsoptionen diskutiert. Eine eindrucksvolle Live-Demo schloss sich an.

In seinem Referat „HP Oracle Database Machine mit I/O-Resource-Management als Integrationsplattform nutzen“ stellte Frank Schneede die Möglichkeiten des Database Resource Managers sowohl innerhalb einer Datenbank als auch über mehrere Datenbanken hinweg sehr anschaulich dar. Die Präsentation mündete in eine Demo per Film.

Die Veranstaltung der SIG Database fand viel Anklang bei den Teilnehmern und es entwickelten sich sehr spannende Diskussionen. Christian Trieb (Paragon Data), stellvertretender DOAG Vorstandsvorsitzender und Leiter der SIG Database, bedankte sich zum Abschluss bei allen Referenten und Teilnehmern und beendete die sehr gelungene Veranstaltung. Die Vorträge stehen für Teilnehmer und DOAG-Mitglieder unter www.doag.org/termine/287745 zur Verfügung. Die nächste SIG Database findet am 1. Oktober 2009 in Berlin und am 7. Oktober 2009 in München statt. Weitere Informationen dazu unter www.doag.org/termine/287746.

Kontakt:

Christian Trieb
sig-database@doag.org



BZYXS.COM
database optimisation

BzYxS.com schenkt Ihnen Zeit und Geld!

Optimierung der Performance und Nutzung von Storage, Datenbank- und Application Servern großer DWHs, Oracle BI / Siebel Analytics und Siebel CRM Installationen. Umfassende Analyse und Problemlösung auf Storage-, Datenbank- und Anwendungsebene.

Troubleshooting bei aktuellen, Erkennung und Beseitigung zukünftiger Engpässe mit unserem Rundum-Sorglos-Programm. Erfahren Sie mehr unter info@BzYxS.com.

Critical Patch Update – eine Erfolgsstory?

Franz Hüll, DOAG Competence Center Security

Im April hat Oracle wieder einen Critical Patch Update (CPU) herausgebracht. Seit der Veröffentlichung des ersten CPU im Januar 2005 sind damit achtzehn CPUs erschienen, die das gemeinsame Ziel haben, die Sicherheit der Produkte aus dem Hause Oracle zu verbessern.

Oracle hat damals nach einigen Startschwierigkeiten – manche erinnern sich noch an den „Oracle Security Alert #68“ – diesen Weg beschritten. Damit wurde offensichtlich, dass auch Software aus dem Hause Oracle Sicherheitslücken enthalten kann und es Aufgabe des Herstellers ist, diese zu schließen. Zuerst ein paar Zahlen (siehe Abbildung 1):

Auf Grund von Oracles dynamischer Einkaufspolitik tauchen immer wieder neue Produkte auf. Ein Zeichen dafür, dass auch diese, zumindest vereinzelt, in den Genuss von CPUs kommen. Auch die Produktbezeichnungen ändern sich im Laufe der Jahre. Über 1000 behobene Sicherheitslücken sind eigentlich eine gute Nachricht und ein gutes Zeichen. Allerdings, und das

Fragezeichen in der Überschrift deutet es schon an, gibt es auch wesentliche Aspekte, die nicht zufriedenstellend sind.

Vereinzelt wurden Fehler nicht in einem CPU behoben, sondern die Behebung zog sich über mehrere CPUs hin. Auch dauerte es manchmal etwas länger, bis ein an Oracle gemeldeter Fehler behoben wurde. Das sind jedoch Punkte, die man unter Berücksichtigung der hochkomplexen Software und der manchmal sicher nicht einfach zu behebbenden Fehlerursache noch nachvollziehen kann.

Problematischer ist es, dass der CPU nicht für alle unterstützten Plattformen zum selben Zeitpunkt herausgegeben wird. Warum ist das ein Problem? Die Tatsache, dass ein Hersteller die Behebung einer Sicherheitslücke bekanntgibt, beinhaltet auch die Information, dass eine Sicherheitslücke vorliegt. Das liegt in der Natur der Sache und lässt sich leider nicht vermeiden. Hält der Hersteller alle Informationen über die mit einem Patch geschlossenen Sicherheitslücken zurück, dann geht der Kunde zu Recht auf die Barrikaden und verlangt mehr Informationen. Er will wissen, ob dieser Patch für ihn relevant und wichtig ist, oder nicht. Gibt der Hersteller Informationen über die Sicherheitslücken preis – und seien diese noch so kurz gehalten –, so liefert er unfreiwillig Hinweise für die Suche nach Möglichkeiten, die vorhandene Lücke in nicht gepatchten Systemen auszunutzen.

Hier sind aber auch die Hersteller von Standard- oder Individualsoftware gefordert, die auf der Basis von

Datum	Database (Server+Client)	Times Ten	Secure Backup	HTTP Server	HTML DB/AppI.Express	Application Server	Collaboration Suite	E-Business Suite	EM/Grid	PS/JDE	BEA	Gesamt
Jan 05	17					5	1	2				25
Apr 05	24					18	33	5	1	7		88
Jul 05	12					12	6	17	2			49
Okt 05	38					19	18	22	1	6		104
Jan 06	37					17	20	27		2		103
Apr 06	14					1	5	15	2	2		39
Jul 06	27					10	1	20	4	3		65
Okt 06	22			6	35	14	12	22		9		120
Jan 07	17			9		10	12	11	6	3		68
Apr 07	16					7	2	11	1	4		41
Jul 07	18				1	4	5	14		7		49
Okt 07	30					11	7	8	3	3		62
Jan 08	9					7	1	10		4		31
Apr 08	16				2	5	1	11	1	3		39
Jul 08	13	3				9		6	2	7	7	47
Okt 08	15					6		5		5	6	37
Jan 09	10	1	9			4	1	4	1	6	5	41
Apr 09	16					12		3		4	8	43
Gesamt	351	4	9	15	38	171	125	213	24	75	26	1051

Abbildung 1: Zahl der Sicherheitslücken seit Januar 2005

Oracle Software-Entwicklung betreiben. Häufig schreiben diese Dritthersteller genau vor, welche Release- und Patch-Stände ihre Software benötigt. Es gehört zu den Aufgaben dieser Firmen, sehr zeitnah die Freigabe (Zertifizierung) des aktuellen CPU für ihre Software durchzuführen.

Es kommt noch schlimmer

Das Hauptproblem liegt aber darin, dass häufig der CPU vom Kunden sehr spät oder gar nicht eingespielt wird. Der Kunde wartet häufig lieber auf das nächste Patchset oder gar Major Release und spekuliert darauf, dass er bei der Migration auch die bis dahin erschienenen CPUs „on the fly“ erledigt und bis dahin schon nichts passieren wird. Viele Kunden gehen auch den Weg, ein oder zwei CPUs im Jahr zumindest auf die kritischsten Datenbanken einzuspielen. Wenn ein CPU vom Kunden als hochkritisch eingestuft wird, dann erfolgt auch schon mal eine Sonderaktion, um diesen CPU zusätzlich zu übernehmen.

Der Grund für all diese Schwierigkeiten liegt darin, dass das Einspielen der CPUs immer noch mit viel Aufwand und auch Risiko verbunden sein kann. Der CPU muss getestet werden:

- Laufen noch alle Anwendungen wie gewohnt, gibt es irgendwo Abweichungen?
- Läuft die Installation problemlos durch?
- Wie und wann plane ich die Downtime ein?
- Brauche ich ein Fall-Back-Szenario?
- Was mache ich mit vielen (mehrere 100 oder gar mehrere 1000) Datenbank-Instanzen?
- Gibt es Quer- und Seiteneffekte?
- Wie hoch ist das Risiko, den CPU einzuspielen?
- Wie hoch ist das Risiko, den CPU nicht einzuspielen?
- Ist der CPU von Drittherstellern zertifiziert?

Dies und vieles mehr sind die Fragen, die den Datenbankverantwortlichen umtreiben. Und das viermal im Jahr.

Was kann getan werden?

Um einem Missverständnis vorzubeugen: Hier soll nicht dafür plädiert werden, die Häufigkeit von CPUs zu reduzieren, oder gar ganz darauf zu verzichten. Nein, die Forderung an den Datenbankhersteller kann nur lauten: Oracle muss technologisch und organisatorisch die Voraussetzungen dafür schaffen, dass CPUs problemlos und sicher übernommen werden können.

Es gilt hier, das Vertrauen der Kunden darin aufzubauen, dass die Übernahme eines CPU kein (oder nur ein ganz minimales) Risiko für den Betrieb darstellt. Dazu gehört auch die Möglichkeit, einen CPU weitestgehend ohne Downtime zu übernehmen sowie einen übernommenen CPU bei Bedarf auch wieder problemlos und restlos entfernen zu können.

Eine Forderung, die leicht gestellt ist, die aber sicher nicht leicht zu re-

alisieren ist. Jeder, der sich schon mal intensiv mit Release-Management und Softwaredesign auseinandergesetzt hat, weiß, dass diese Aufgabenstellung hohe Anforderungen in sich birgt. Das umfangreiche Oracle Portfolio macht die Sache auch nicht leichter.

Es würde Oracle gut zu Gesicht stehen, wenn dem Kunden ein fehlerresistentes und sicheres Verfahren an die Hand gegeben wird, um die Übernahme eines CPU stabil und zuverlässig zu bewerkstelligen, und damit den Aufwand des Kunden erheblich zu reduzieren, die Planungssicherheit aber im selben Maße zu steigern. Die Veröffentlichung von CPUs zur Schließung vorhandener Sicherheitslücken ist ein erster und wichtiger Schritt; es wird Zeit, dass der zweite folgt.

Kontakt:

Franz Hüll
securityfragen@doag.org



Analyse Beratung Projektmanagement Entwicklung

Ihr Spezialist für webbasierte Informationssysteme mit

Oracle WebLogic Server
Oracle WebLogic Portal

exensio ● ● ●
www.exensio.de



Weitere Experten-Seminare bereits angekündigt

Am 14. September 2009 steht „Advanced Oracle Objekt Monitoring“ und am 15. September 2009 „Advanced Oracle Datenbank Monitoring“ auf dem Programm. Der bekannte Referent Marco Patzwahl gibt Lösungen zu folgenden Problemstellungen: Die Produktiv-Datenbank hängt oder ist sehr langsam? Wie findet man die ressourcenintensivsten Statements von Oracle-Applikationen und wie spürt man Objekte auf, die reorganisiert werden sollten? Welche Indizes fehlen in der Datenbank und welche Statistiken sind zu ungenau oder werden nie aktualisiert?

Die Teilnehmer lernen, mit Hilfe von Skripten oder kleinen Prozeduren die Schwachstellen in der Datenbank zu finden, die sonst nicht aufzuspüren sind. Es werden viele praxisnahe Beispiele demonstriert, die auch als Vorlage mitgenommen werden können. Der Referent zeigt schnelle und effektive Wege, die Datenbank auf verschiedene Arten zu monitoren sowie Sperren, Blockaden und Engpässe zu erkennen und zu beheben. Die Teilnehmer lernen die wichtigsten Fehlerfälle kennen, die eine Datenbank zum Hängen bringen kann. Sie bekommen ein Package an die Hand, mit dem mehrere Datenbanken einfach zu überwachen sind. Außerdem werden Sicherheitsprüfungen vorgestellt, um Datenbank-Objekte zu schützen.

Weitere Informationen finden Sie unter www.doag.org/expertenseminare



Berliner Experten-Seminare

Die DOAG bietet zusammen mit ausgewählten Schulungspartnern spezielle Seminare an. Diese Seminare sind keine Grundlagen-Schulungen, sondern richten sich als Ergänzung zu den Schulungsangeboten der Oracle-Partner in der Regel an erfahrene Oracle-Anwender, die ihr Wissen vertiefen möchten. Die Seminare werden in der Regel von sehr erfahrenen, namhaften Dozenten gehalten. Die beiden Veranstaltungen „Oracle 11g Performance Forecast“ und „Oracle 11g Performance Firefighting“ waren komplett ausgebucht.

Das Seminar zum Performance Forecast beschäftigte sich mit der Überwachung und Vorhersage von Datenbank-Performance, was nach ITIL in den Bereich „Capacity Management“ einzuordnen ist. Hier wurden neben mathematischen Methoden für eine Vorhersage die neuen Features von Oracle 11g zur Unterstützung des Performance Monitoring präsentiert. Schwerpunkt war dabei das dreistufige Monitoring sowie eine Analyse, was der Oracle Enterprise Manager in diesem Zusammenhang leistet. Begriffe wie „Moving Window Baseline“ oder „Statistische Schwellwerte“ sind nun für viele keine Fremdwörter mehr. Zu den am häufigsten benutzten Forecast-Methoden „Queuing-Theorie“ und „Skalierung“ wurden mathematische Grundlagen und Einsatzgebiete erläutert. Fallbeispiele für CPU- und I/O-Forecasts sowie für einen System-Forecast rundeten das Thema ab.

„Oracle 11g Firefighting“ begann mit der Herangehensweise und Methoden zur Analyse von Performance-Problemen sowie Empfehlungen, wie typische Probleme vermieden werden können. Als am häufigsten verwendete Methode wurde das Drill-Down-Prinzip sowie dessen praktische Anwendung mit dem Enterprise Manager und von der Kommandozeile vorgestellt. Weitere Werkzeuge und Features wie „oradebug“, „Active Session History“, „Server Result Cache“, „Automatic Memory Management“ oder „ASM“ wurden diskutiert. Auf ein besonders großes Interesse sind die Kapitel „Oracle Internas“ zu den Bereichen Redo Logs, Latche und Mutexe sowie Buffer Cache und Shared Pool gestoßen.

Insgesamt herrschte im Seminar eine sehr offene Atmosphäre, es gab viele Diskussionen und Meinungsbeiträge und es war letztendlich nicht nur ein Seminar sondern gleichzeitig ein Erfahrungsaustausch.

Eine Bootsfahrt auf der Spree als Sightseeing Tour mit gemeinsamem Abendessen war aus Sicht des Dozenten eine Spitzenidee des Veranstalters! Für die Teilnehmer war es beeindruckend zu sehen, wie sich die Stadt seit der Wiedervereinigung entwickelt hat sowie die gelungene Verbindung von historischen und neuen Gebäuden von der Spree aus zu betrachten. An diesem Abend wurden viele persönliche Kontakte geknüpft und man wird sich lange Zeit daran erinnern.

Terminkalender September bis November 2009

September

Dienstag, 01.09.2009

Regionaltreffen Rhein/Main

Thomas Tretter
regio-rhein-main@doag.org

Dienstag, 08.09.2009

Fachkonferenz DOAG ITIL & Betrieb 2009

ITIL-Prozessoptimierung, Betrieb-
effiziente Umsetzung

Fried Saacke
office@doag.org

Mittwoch, 09.09.2009

Regionaltreffen Dresden/Sachsen

Helmut Marten
regio-sachsen@doag.org

Regionaltreffen Berlin Brandenburg

Oracle 11g Upgrade

Michel Keemers
regio-bb@doag.org

14. – 15.09.2009

DOAG Berliner Expertenseminare

Advanced Oracle Objekt Monitoring
Advanced Oracle Datenbank Monitoring

DOAG Office
office@doag.org

Regionaltreffen Hamburg/Nord

Oracle und PHP5

Michael Paege
regio-hamburg@doag.org

Dienstag, 15.09.2009

SIG Business Process Management

BPM unleashed – vom Papiertiger zum
zentralen Instrument der Qualitätssicherung

Sebastian Graf
sig-bpm@doag.org

Mittwoch, 16.09.2009

SIG Fusion Middleware

Betrieb einer SOA Infrastruktur

Perry Pakull
sig-middleware@doag.org

Donnerstag, 17.09.2009

Regionaltreffen Rhein-Neckar

Oracle ApEx: Erfahrungsbericht Bosch
Automotive, ApEx-Einstieg – Von 0 auf 100

Kai Christianus
regio-rhein-neckar@doag.org

Freitag, 18.09.2009

SIG Development

Neue, weniger verbreitete Entwicklungs-
umgebungen für Oracle-Projekte
(Delphi, .NET, PHP, ...)

Christian Schwitalla, Andreas Badelt
sig-development@doag.org

Montag, 21.09.2009

Regionaltreffen Südbayern/München

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

Dienstag, 22.09.2009

SIG Security und Datensicherheit

Frank Stöcker
sig-security@doag.org

Mittwoch, 23.09.2009

SIG JD Edwards

Kasi Färcher-Haag
sig-jde@doag.org

Donnerstag, 24.09.2009

SIG E-Business Suite

Erfolgspotenziale in stürmischen Zeiten:
Oracle E-Business Suite Best Practices und
Serviceinnovationen

Dr. Frank Schönthaler
sig-ebusiness@doag.org

Montag, 28.09.2009

Regionaltreffen Hannover

Henrik Rüniger
regio-hannover@doag.org

Dienstag, 29.09.2009

SIG SOA

Technische Prozessautomatisierung
und Integration

Torsten Winterberg, Hajo Normann
sig-soa@doag.org

Mittwoch, 30.09.2009

Regionaltreffen NRW

Dierk Lenz
regio-nrw@doag.org

Oktober

Donnerstag, 01.10.2009

SIG Database

ORACLE Streams und Verteilte Datenbanken

Christian Trieb, Michael Pfautz
sig-database@doag.org

Dienstag, 06.10.2009

SIG Siebel

Kunden Erfahrungsberichte Siebel 8.1
& CRM in der Wirtschaftskrise

Detlef D. Kubusch
sig-siebel@doag.org

Mittwoch, 07.10.2009

SIG Database

ORACLE Streams und Verteilte Datenbanken

Christian Trieb, Michael Pfautz
sig-database@doag.org

Dienstag, 13.10.2009

Regionaltreffen Jena/Thüringen

Jörg Hildebrandt
regio-thueringen@doag.org

Dienstag, 20.10.2009

Regionaltreffen Nürnberg/Franken

Jürgen Häffner
regio-franken@doag.org

Mittwoch, 21.10.2009

Regionaltreffen Bremen

Ralf Kölling
regio-bremen@doag.org

Donnerstag, 22.10.2009

Regionaltreffen Stuttgart

Jens-Uwe Petersen
regio-stuttgart@doag.org

Dienstag, 27.10.2009

Regionaltreffen Freiburg/Südbaden

Volker Deringer
regio-freiburg@doag.org

Donnerstag, 29.10.2009

Regionaltreffen Trier/Saarland

Hochverfügbarkeit bei Oracle-Datenbanken
Holger Fuchs + Bernd Tuba
regio-trier@doag.org

November

17. – 19. November 2009

DOAG 2009 Konferenz + Ausstellung

Das Treffen der Deutschen Oracle-Community im CongressCenter Ost in Nürnberg

Freitag, 20. November 2009

DOAG 2009 Schulungstag

www.doag2009.org

Weitere, aktuelle Informationen finden Sie unter www.doag.org/termine

Vorschau auf die nächste Ausgabe

Das Schwerpunkt-Thema der Ausgabe Q4/2009 lautet:
„Die neue Datenbank 11g R2“

Gesucht sind folgende Themen:

- Neue Features in 11g R2
- Erfahrungen aus dem Beta-Test

Darüber hinaus finden Sie wie in jeder Ausgabe:

- Fachartikel aus dem Bereich Datenbank
- Best Practice im Umgang mit den Oracle-Produkten
- Aktuelle Informationen über Veranstaltungen der DOAG
- Neues von Oracle

Redaktionsschluss ist am 31. August 2009

Die Ausgabe Q4/2009 erscheint am 30. Oktober 2009



Wir begrüßen unsere neuen Mitglieder!

Firmenmitglieder:

Alexander Doubek	Comit SE
Johannes Abheiden	Prisma Software Engineering GmbH
Thomas Engels	Prisma Software Engineering GmbH
Bruno Dörrer	W&W Informatik GmbH
Steffen Böttinger	W&W Informatik GmbH
Patrick Essich	W&W Informatik GmbH
Matthias Pfau	Zühlke Engineering GmbH
Udo van den Berg	ComDev Systemlösungen GmbH
Enrico Meier	Hilcona AG
Ulrich Schell	ABeam Consulting B.V.
Eric Wolf	Lufthansa CityLine GmbH
Christian Grave	ProLicense GmbH i.G.
Josef Opitz	Personalberatung OPITZ GmbH

Angelika Dye	ORACLE Deutschland GmbH
Christian Weinfurter	ORACLE Deutschland GmbH
Björn Wolter	
Michael Roeder	Independent Royalty Services GmbH
Frank Grobelny	Independent Royalty Services GmbH
Peter Köhler	
Christoph Jansen	Trivadis GmbH
Florian Hacklinger	Zühlke Engineering GmbH
Matthias Papesch	
Alexander Richter	Steria Mummert Consulting AG
Birgit Arndt	
Lutz Bauer	
Norbert Greif	
Vasudevan Raghavan	Cognizant Technology Solutions GmbH
Sebastian Dörr	
Frank Hofmann	
Robin Stenke	

Persönliche Mitglieder:

Ilker Büyükkörüc	
Martin Schlichenmayer	ORACLE Deutschland GmbH
Wim Beulen	Informatica