Nr. 2 | April 2013 | ISSN 09 36-0360

DOAG News

Deutsche ORACLE-Anwendergruppe e.V.

Reporting

www.doag.org

Stefan Kinnen
Leiter Development Community

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

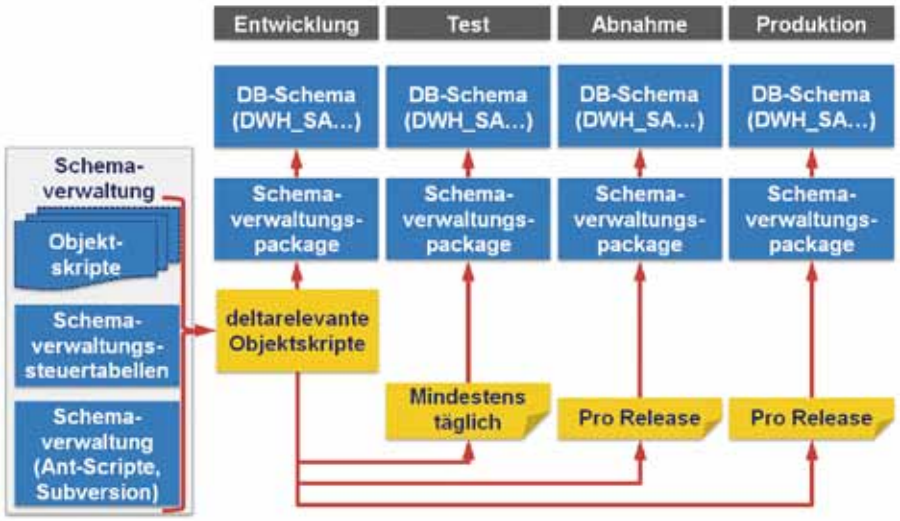
eine erfolgreiche Steuerung der Unternehmensleistung hängt heutzutage vor allem von aussagefähigen und beständigen, konsolidierten Informationen ab. Die Herausforderungen, denen die Unternehmen heute gegenüberstehen, sind vielfältig: Globalisierte Märkte, zeitnahe Berichterstattung und getrennte Berichtswesen, darüber hinaus müssen eine hohe Datenqualität, nachvollziehbare Abschlüsse und parallele Abschlussszenarien gewährleistet sein.

Mit diesem und anderen Themen beschäftigt sich nicht nur diese Ausgabe der DOAG *News*, sondern auch die Community-Konferenz DOAG 2013 BI am 17. April 2013 in München, die in diesem Jahr den Fokus auf den wichtigen Aspekt der Daten- und BI-Konsolidierung sowie die damit verbundene Lifecycle-Optimierung bis hin zur Steigerung des Nutzwerts legt.

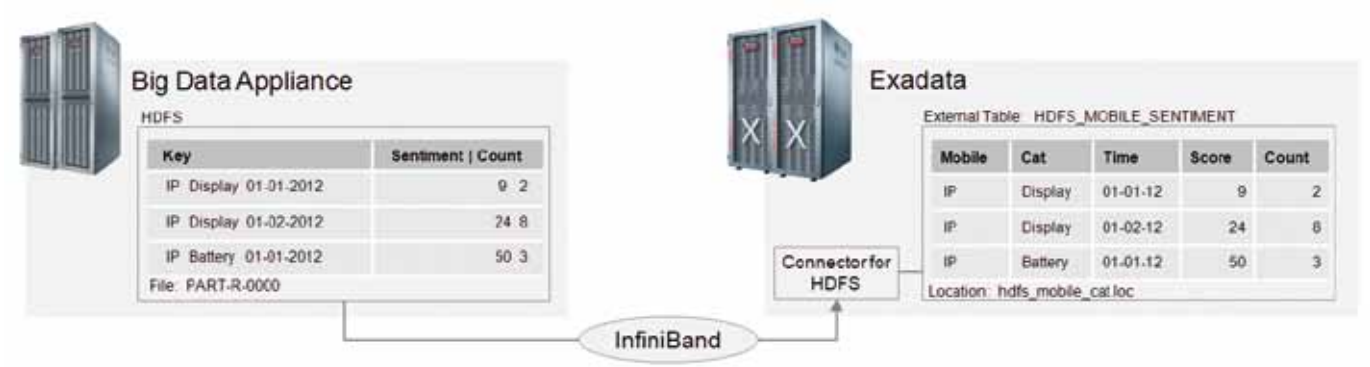
Dieses Event ist aber nur der Auftakt zur bislang umfangreichsten Reihe an Frühjahrs-Fachkonferenzen, die die DOAG je angeboten hat. Mithilfe unserer vielfältigen Kommunikationskanäle können Sie sich leicht über die einzelnen Programme und Mehrwerte informieren. Es erwarten Sie spannende Themen und viel Erfahrungsaustausch in den jeweiligen Communities.

Anfang Juni 2013 schauen wir dann alle gespannt nach Mainz, wo im Anschluss an das DOAG 2013 Infrastruktur und Middleware Community Summit die erste Delegiertenversammlung nach der neuen Satzung zusammentrifft, um die weitere Ausrichtung der DOAG zu prägen. Wir sind sehr zuversichtlich, die DOAG damit für Ihre individuellen Erwartungen künftig noch attraktiver gestalten zu können.

Ihr



Agile Methoden kommen bei der Neu- und Weiterentwicklung von BI-Projekten immer häufiger zur Anwendung, Seite 48



Aufbau agiler BI- und Discovery-Applikationen mit Oracle Endeca, Seite 40

Spotlight

Freitag, 18. Januar 2013

Das Nominierungsverfahren zur Wahl der Delegiertenversammlung beginnt. Die ersten Kandidaten melden sich gleich noch am ersten Tag.

Freitag, 1. Februar 2013

Das Nominierungsverfahren zur Wahl der Delegiertenversammlung ist abgeschlossen. Der Wahlausschuss registriert 62 Bewerber auf die 37 zu wählenden Delegierten. Die DOAG-Leitung freut sich über das große Interesse an diesem wichtigen Amt. Jeder Kandidat stellt sich bis zum Wahlbeginn auf den Webseiten des Wahlsystems der DOAG vor.

Montag, 18. Februar 2013

Werner Keller, pensionierter und erfahrener Oracle Vice President, möchte zukünftig ehrenamtlich in der DOAG mitwirken und insbesondere die DOAG Business Solutions Community unterstützen.

Dienstag, 19. Februar 2013

Im Kick-off-Meeting legt das Team im Berliner DOAG-Office den Grundstein für ein erfolgreiches Jahr auf der operativen Seite.

Mittwoch, 20. Februar 2013

Das Leitungsmeeting der DOAG Business Solutions Community stellt die Weichen für die DOAG 2013 Applications. Die Fachkonferenz für die Anwender der Oracle Business Solutions findet in diesem Jahr erstmals im Herbst vom 9. bis zum 11. Oktober in Berlin statt. Der schwierige Oracle-Applications-Markt in Deutschland stellt eine große Herausforderung für das gesamte Team dar.

Freitag, 15. März 2013

Die DOAG-Leitungssitzung bereitet die Delegiertenversammlung sowie die Community-Konferenzen der vier Communities vor.

Montag, 18. März 2013

Die Wahl der Delegiertenversammlung beginnt. Bis zum 22. März um 20 Uhr sind alle Mitglieder aufgerufen, ihre Stimme elektronisch abzugeben.

Freitag, 22. März 2013

Die Wahlbeteiligung zur Delegiertenversammlung übertrifft alle Erwartungen. Um 20.15 Uhr gibt der Wahlausschuss das vorläufige Wahlergebnis bekannt.

Sonntag, 31. März 2013

Nach Einholung der notwendigen Zustimmung aller gewählten Kandidaten steht das endgültige Ergebnis der Wahl zur Delegiertenversammlung fest.

Dienstag, 2. April 2013

Die DOAG-Leitung lädt zur ersten Delegiertenversammlung am 7. und 8. Juni 2013 in Mainz ein. Insgesamt 55 Delegierte sind aufgerufen, dort die Interessen der Mitglieder zu vertreten.

Das fehlende Interview

Normalerweise finden Sie auf der nachfolgenden Seite immer ein Interview. Für diese Ausgabe fand es bei der Metro Systems GmbH in Düsseldorf statt. Leider warteten wir anschließend vergeblich auf die Freigabe. Nach mehrfachen Nachfragen erhielten wir Wochen später die Auskunft: „Wir möchten von einer Veröffentlichung des Interviews absehen. Wir halten es aus Unternehmenssicht für bedenklich, dass in dem Interview zu Themen Stellung genommen wird, bei denen nicht immer erkennbar ist, wann es sich um persönliche Ansichten des Mitarbeiters und wann es sich um strategische Ausrichtungen der Metro Systems handelt. Zudem werden Aspekte thematisiert, die wir lieber im persönlichen Austausch mit dem Dienstleister besprechen." Da die Fragen bereits vor dem Interview der Metro Systems GmbH vorlagen, hätten wir uns den Aufwand sparen können.
In der nächsten Ausgabe finden Sie wieder wie gewohnt ein interessantes Interview, diesmal mit der Regio IT in Aachen.

Von außen kommend hat man es bei der Beobachtung von Systemen und Anwendungen immer leicht. Dann wundert man sich über diese und jene Eigentümlichkeit. „Warum machen die das so und nicht anders?" oder „Es wäre doch so einfach, wenn ..."

Einblicke in die Praxis gewachsener Data-Warehouse-Systeme

Alfred Schlaucher, ORACLE Deutschland B.V. & Co. KG

Die Bedingungen, unter denen die Data-Warehouse- und Business-Intelligence-Systeme aufgebaut und verwaltet werden, sind meist von außen nicht erkennbar. Schon deswegen sollte man sich mit Bewertungen zurückhalten. Aber von außen kommend, hat man auch eine gewisse Distanz und läuft nicht Gefahr „den Wald vor lauter Bäumen nicht mehr zu sehen". Deshalb sind die Data-Warehouse-Reviews von Oracle auch eine so interessante Quelle von Aspekten aus der gelebten Praxis. Der Startpunkt ist ein standardisierter Fragebogen, den man in einem lockeren Gespräch zwischen Systemverantwortlichen und Oracle-Mitarbeitern beantwortet. In der Regel münden die Gespräche dann nach Stunden in intensive Diskussionen über das „Für und Wider" von Techniken und Verfahren. Nicht selten fällt der Satz: „Das haben wir schon immer so gemacht." Spätestens jetzt ist ein Ansatz für Verbesserungen gefunden, denn etwas Verfestigtes ist meist schon überholt.

Ansprüche und Ziele von Data-Warehouse-Verantwortlichen

Unbestritten gehören Analyse-Systeme heute zu den Top-Anwendungen. Entsprechend heftig ist demzufolge auch die Diskussion um Technologien, Architekturen und Tools. Zwei Hauptziele sind in den Diskussionen erkennbar:

- Die Flexibilität der Informationen soll gesteigert werden. Gemeint ist damit die Verzahnung von Daten aus unterschiedlichen Sachgebieten sowie eine variable Detailtiefe, von grob aggregiert bis auf die Ebene operativer Daten.
- Man will in immer kürzeren Zeitabständen aktuelle Informationen bekommen, also im Extremfall die geschäftliche Entwicklung bis kurz vor Berichterstellung sehen.

Unausgesprochen stehen auch immer die Kosten im Raum. Das Senken von Kosten wird nicht als Hauptziel genannt. Sie schränken jedoch den Handlungsspielraum ein und entscheiden als gesetzte Zwangslage bei der Umsetzung von Ideen meist mit.

Die Ziele sind nicht neu, aber die Wege und Irrungen dorthin. Nicht selten verselbstständigt sich beispielsweise ein spezieller Tool-Einsatz und versperrt den Weg zu einer optimierten Anwendung. Einmal getroffene Entscheidungen zu revidieren, fällt dann schwer.

Bei der Herangehensweise an Data-Warehouse-Aufgaben und auch bei der Bewertung über Sinnhaftes und Unsinniges nutzt Oracle meist ein entsprechendes Schichtenmodell (in der Regel „Inmon"). Die Anzahl der Schichten und die Menge der Schritte sind dabei weniger entscheidend. Wichtig ist die Lösung der beiden Hauptaufgaben:

- Das technische, aber auch fachliche Zusammenführen von Informationen aus mehreren Sachgebieten
- Die Informationen so bereitzustellen, dass sie auch ein Fachfremder versteht

Die Rahmenbedingungen sind so zu wählen, dass Datenflüsse möglichst kurz und redundanzfrei bei geringstem Hardware-Einsatz sind. Hinter diesen pauschalen Formulierungen steckt ein komplettes Konzept, das der Autor auf Anfrage gerne diskutiert.

Immer wieder „Stove Pipes"

Vielen Systemen fehlt der sogenannte „Enterprise-Bezug" beziehungsweise ist die „Enterprise-Fähigkeit" oft nicht ausgeprägt genug. Damit meint man die Möglichkeit, über eine einzige Abfrage Informationen aus unterschiedlichen und über das Aufgabenspektrum des Unternehmens verteilte Sachgebiete zu erfahren. Offenbar sind die Systeme zunächst nur für ein Sachgebiet entwickelt worden (einen Data Mart). Daten wurden direkt aus den Vorsystemen in die Auswerte-Modelle geladen, ohne sie zuvor zu granularisieren und vergleichbar zu machen. Als die Anforderungen nach weiteren Informationen hinzukamen, wurde erweitert, indem man einfach nur einen weiteren Data Mart neben den bestehenden stellte, ohne die vorhandenen Informationsobjekte (und auch ETL-Prozesse) miteinzubeziehen. Es entstand ein System ohne einheitlichen Informationshaushalt, aber mit vielen autarken Data Marts.

Diesen auch „Stove Pipes" genannten Effekt findet man gerade in großen Warehouse-Systemen in größeren Unternehmen. Die Folgen werden leider erst Jahre nach der Erstimplementierung sichtbar. Nachteilig ist hier:

- Die fehlende Flexibilität für die Endanwender, die keine zusammenhängenden und stimmigen Informationen erhalten.
- Die hohen Kosten für die Weiterentwicklung, wenn das System er-

weitert werden muss. Oft muss man komplett neue ETL-Strecken und Data Marts entwickeln.

Eine teure Neuimplementierung des gesamten Systems ist oft die einzige Lösung. Es ist erstaunlich, wie wenig das Drei-Schichten-Architektur-Prinzip in der Praxis diskutiert und konsequent angewendet wird.

Starre Implementierung des Schichtenmodells

Es gibt aber auch das entgegengesetzte Extrem: Eine sachgebietsübergreifende, zentrale Warehouse-Schicht ist zwar vorhanden, sie ist aber gegenüber den Auswertemodellen (Data Marts) schon fast hermetisch abgeschottet: „In mein zentrales Data Warehouse kommt kein Anwender rein." Wenn jedoch kein Anwender hineinkommt, dann muss man eben alle benötigten Daten in die Data Marts kopieren, auch wenn die Tabellen extrem groß sind und auch wenn die Daten bei dem Weg in die Data Marts nicht mehr modifiziert werden, also keine Mehrwerte erhalten.

Solche Systeme erzeugen unnötigen ETL-, Hardware- und Verwaltungs-Aufwand. Agilität (schnelles Handeln bei Neuanforderungen) wird erschwert.

Aus Gründen der Ressourcen-Schonung kann man folgende Regel formulieren: „Keine „1:1"-Kopien im Data Warehouse". Das bedeutet, dass große Bewegungsdaten- und Fakten-Tabellen nur einmal in dem System vorzuhalten sind und Referenzdaten, die sich (im Gegensatz zu Stammdaten) nicht beziehungsweise kaum verändern, ebenfalls nur einmal vorhanden sein sollten.

Dies gelingt nur, wenn eine zentrale Warehouse-Schicht auch für Endbenutzer-Zugriffe geöffnet bleibt. Das bedeutet auch, dass Dimensionen über Schichtengrenzen hinweg eine Fakten-Tabelle referenzieren, die aus Kostengründen nur einmalig in der zentralen Warehouse-Schicht liegt und nicht im Data Mart.

Auch wenn es manchem DWH-DBA widerstrebt: Alle Schichten sollten nur logisch als Schicht begriffen werden und physisch nicht getrennt (etwa auf unterschiedlichen Rechnern / Datenbanken) liegen.

Entfremdung von Data Warehouse und Business Intelligence

„Fachanwendungen (BI) für die Fachanwender in der Fachabteilung und Data Warehouse in die IT-Abteilung." Je größer ein Unternehmen, desto mehr verfestigt sich diese Vorgehensweise.

Es entsteht folgende Situation: Neben dem zentralen und von der technischen IT verwalteten Data Warehouse gibt es mehrere und zum Teil voneinander unabhängige Business-Intelligence-Anwendungen mit unterschiedlichen Teams und oft auch Tools. Die Zentral-IT beschränkt sich auf die Service-Leistung „die Bereitstellung von Data-Warehouse-Daten /-Schnittstellen" auf Anforderung. Das erfolgt über einen formalen Prozess. Bei ungenügenden Ergebnissen (falsche Daten, nicht zusammenhängend, nicht die nötige Detailtiefe etc.) erfolgen neue, zeitaufwändige Anforderungen an die IT, bis die Anwender resignieren und eigene Wege gehen.

Diese Vorgehensweise führt zu vielen Sonderwegen, Sonderzonen und Sonderverantwortlichkeiten, letztlich zu doppelter Arbeit, mehrfachen Ressourcen und leider oft auch nicht abgestimmten Kennzahlen. Während sich die IT in der technischen Optimierung der Datenbank austobt, verheddern sich die Fachanwender in einem Wust nicht abgestimmter Einzeldaten, die sie oft nicht mehr verstehen. Und allzu oft glaubt man, durch die Anschaffung eines neuen BI-Tools mehr Flexibilität zu schaffen, etwa durch Zugriffe auf Daten der Vorsysteme und schnelle In-Memory-Bearbeitung. Teilweise werden auch ETL-Aufgaben in die BI-Tools verlagert. Die Anforderung nach Flexibilität für die Endanwender ist die Legitimation. Der Datenaufbereitungs-Prozess wird zum Glücksspiel.

Richtig wäre es, den gesamten Datenfluss von den Vorsystemen bis hinein in die Berichte/Analysemodelle als einen zusammenhängenden Informationsbeschaffungs-Prozess zu verstehen und diesen Prozess durch eine Hand verwalten/modellieren zu lassen. Dies setzt voraus, dass in der IT mehr über die fachlichen Anforderungen (Informationsbedarfe) bekannt ist und in den Fachabteilungen mehr Kenntnisse über die IT-Systeme beziehungsweise existierenden Datentöpfe (Quellen/Vorsystem) vorhanden sind. Helfen würde sicher schon eine bessere Zusammenarbeit zwischen den Data-Warehouse-Verantwortlichen und den Fachabteilungen.

In kaum einem beobachteten Data Warehouse konnten die Verantwortlichen spontan erklären, welche Daten für welche Zielgruppe im Data Warehouse gespeichert sind. Es existieren kaum Daten-Inventare und kaum Listen über die Benutzergruppen mit ihren Daten-Interessen. Eine Liste von zugelassenen Datenbank-Benutzern reicht nicht. Nur wenn man die Interessen und Nutzungsgewohnheiten der Anwender kennt und permanent überprüft, kann man die DWH-Informationen passgenau bereitstellen.

Fehlende Planung des ETL-Prozesses

Sind die Schichten und Datenmodelle im Data Warehouse schon zufällig, so sind es erst recht die ETL-Prozesse. Vor allem, wenn ETL-Tools zum Einsatz kommen, verselbstständigt sich dieser Bereich. Dann ist zu sehen, dass ETL-Entwicklung, Datenmodellierung und Informationsbedarfs-Planung auf unterschiedlichen Schreibtischen stattfinden und nicht Hand in Hand gehen.

Aufgrund fehlender Ausrichtung an einem Schichtenmodell und Informationsbedarfs-Planung entsteht oft ein chaotischer Wirrwarr von einzelnen Lade-Jobs und ETL-Strecken. In den meisten Data-Warehouse-Systemen wird zu viel und zu umständlich geladen. ETL-Prozesse könnten mit weniger Redundanzen, mit geringeren Laufzeiten, mit weniger Hardware und weniger Volumen laufen, wenn man sie besser planen würde und diese Aufgabe nicht nur den ETL-Tool-Spezialisten überlassen würde.

Nachgelagerter Stellenwert des Data Warehouse

Man sollte die besondere Bedeutung eines Data Warehouse für heutige Unternehmen nicht mehr betonen müssen. Aber es gibt noch zu viele Verantwortliche, die das nicht wissen: „Ohne DWH geht vieles in den Unternehmen einfach nicht mehr." Dieses Unwissen

herrscht oft in Rechenzentren und in den administrativen IT-Bereichen vor, wo es um Hardware, Storage und Administration geht, Datenbank-Administration nicht ausgeschlossen.

Data-Warehouse-Systeme stehen in der Wahrnehmung nicht an der operativen Front des Unternehmens, es sind Sekundär-Systeme. Sie werden nachgelagert hinter vielen OLTP-Anwendungen irgendwie auch noch mitbetreut. Die technische Administration des DWH wird über die normale DBA-Gruppe erledigt, die allzu oft keinen Unterschied zwischen einer OLTP- und einer DWH-Datenbank macht. Aber: Ein Data Warehouse ist kein OLTP-System. Diese einfache Aussage muss auch in den Rechenzentren und der IT-/DB-Administration berücksichtigt werden. Das bedeutet dedizierte Ressourcen und Verfahren für ein Data Warehouse:

- Separater Storage
- Separate Rechner
- Separate Netze
- Separates Backup-Verfahren
- Separate DBA-Administration (zumindest sollten DBAs über die Besonderheiten in einem DWH Bescheid wissen)

Schwacher Einsatz von Datenbank-Techniken

Wichtige DWH-Features der Datenbank werden immer noch zu wenig eingesetzt. Nach einer Schätzung kommen in der Praxis zum Einsatz:

- 70 Prozent Partitioning
- 40 Prozent Materialized Views
- 30 Prozent DWH-spezifische Indizierung (Bitmap, Star Transformation)
- 30 Prozent analytische Funktionen
- 10 Prozent Mining
- 10 Prozent OLAP
- 50 Prozent ETL-Funktionen

Gerade die Features „Materialized Views“, „DWH-Indizierung“ und „analytische Funktionen“ könnten mehr eingesetzt werden. Der Grund für diese Zurückhaltung liegt zum Teil in der Organisation der Teams. Da es sich um technische Features der Datenbank handelt, fallen sie meist in das Aufgabengebiet eines OLTP-orientierten Datenbank-Administrators. Der Nutzen eines Materialized-View-basierten Kennzahlensystems oder analytisch aufbereiteter Werte ist jedoch nur für Fachanwender erkennbar. Diese wissen wiederum oft nicht um die technischen Möglichkeiten der Datenbank und akzeptieren eine auch nur spartanisch bereitgestellte Lösung. Beide Gruppen müssten stärker zusammenarbeiten.

Alfred Schlaucher
alfred.schlaucher@oracle.com

Es entstehen immer größere Datenmengen, die aus immer unterschiedlicheren Formaten und aus immer mehr Datenquellen gespeist werden. Die Erkenntnisse, die aus diesen Daten gewonnen werden können, sind das Gold des Informationszeitalters.

Aktuelle Trends bei Business Intelligence und Data Warehouse

Klaus Rohrmoser, data2fact

Soziale Medien wie Facebook oder Google stellen nicht umsonst kostenlos ihre Dienste zur Verfügung. Daten intelligent zu verwenden, wird künftig immer stärker im Fokus stehen. Hierbei gibt es verschiedenste Methoden und Technologien, die unter den Begriffen „Business Intelligence" (BI) und „Data Warehouse" zusammengefasst werden können. Dieser Artikel gibt einen Überblick über aktuelle Trends.

Business Intelligence und Data Warehouse werden oft als Synonym verwendet, sie unterscheiden sich jedoch grundsätzlich. Während der Begriff „Data Warehouse" Technologien zur optimierten Datenspeicherung umfasst, ist Business Intelligence als ein Prozess zu verstehen, der relevante Informationen aus Daten gewinnt und diese an operative Systeme zurückspielen kann (Closed Loop). Business Intelligence kann IT-Systemen und -Anwendern ein Lernen aus verfügbaren Daten ermöglichen, um effizientere Entscheidungen zu treffen.

Self Service Business Intelligence

In den meisten Unternehmen ist die IT an vorgegebene Release-Zyklen, einzuhaltende SLAs und Kostenlimits gebunden. Fachanwender möchten jedoch eine schnelle und dynamische Umsetzung ihrer Anforderungen, um ihr Geschäft zu betreiben. Dieser Widerspruch lässt sich durch Self Service BI auflösen, sofern einige wichtige Aspekte beachtet werden.

Mit Self Service BI kann mehr Agilität in der Analyse und Auswertung von Unternehmensdaten generiert werden. Reporting-Anwender können Anforderungen mit Ad-hoc-Analysen und Reports schnell und flexibel mit den vorhandenen Reporting-System(en) um setzen. Damit sind Unternehmen in der Lage, schneller auf Anforderungen der Kunden und des Geschäftsumfelds zu reagieren. Die wichtigsten Aspekte von Self Service BI sind:

- *Benutzerfreundlichkeit*
  Einfache Bedienbarkeit des Reporting-Systems bei der Erstellung von Ad-hoc-Reports oder Analysen. Möglichkeit zur Zusammenarbeit zwischen Benutzern (Collaboration). Integration von eigenen, oft Excel-basierten Auswertungen in ein bestehendes Dashboard.
- *Rollenverständnis*
  Fachanwender lieben Excel und können damit ausgefeilte Auswertungen erzeugen. Daten-Analysten sind SQL-affin, verstehen Datenmodelle und können komplexe Analysen erstellen. Die IT muss eine Infrastruktur finden, um beiden Rollen einen Mehrwert bieten zu können.
- *Data Governance*
  Qualität, Herkunft und Aktualität der Daten sowie Kennzahl-Definitionen sind entscheidende Aspekte, damit bei der Anwendung von Self Service BI keine Äpfel mit Birnen verglichen werden, sondern verlässliche und nachweisbare Informationen entstehen.
- *Agilität und Business Intelligence*
  Ein ständiger Austausch zwischen Fachanwendern, Daten-Analysten und IT, um mithilfe der gefundenen Erkenntnissen IT-Systeme weiterzuentwickeln und/oder Geschäftsprozesse zu verbessern.
- *Gemeinsame Datenbasis*
  Eine über mehrere Quellen integrierte Datenbasis für Reporting Stakeholder, die in einem Data Warehouse vorhanden sein kann. Flexibel auswertbar und um zusätzliche (etwa Fachbereich-bezogene) Datenquellen erweiterbar, unter Einhaltung der Data Governance.
- *Sandboxing*
  Durch die einmalige Bereitstellung von produktiven Daten zur Daten- und Anforderungs-Analyse können Erkenntnisse sowohl für das operative Geschäft als auch für künftige Anforderungen gewonnen werden.

Nicht alle Anwender wollen SQL verstehen, sondern vielmehr schnell Antworten auf ihre Fragen erhalten. Zum Erfüllen ihrer Kernaufgaben im Unternehmen benötigen sie Informationen. Um Daten lesen und verarbeiten zu können, braucht man technisches Wissen über SQL und Datenmodelle, erst dann entstehen wertvolle Analysen. Rollenverständnis und Data Governance sind zwei wesentliche Aspekte, um erfolgreich Self Service BI im Unternehmen anzuwenden.

Die IT sollte sich darauf konzentrieren, die Schnelligkeit und Qualität ihres Lösungsportfolios stetig zu verbessern, und die Fachseite sollte Data Governance als wichtigen Bestandteil von Business Intelligence anerkennen. Gemeinsam kann den Anwendern dadurch ermöglicht werden, ihre Kernaufgaben im Unternehmen wahrzunehmen.

Trends bei Datenbank-Technologien

Aufgrund der steigenden Datenmenge und der höheren Ansprüche an Anzahl

und Dauer von Zugriffen entwickeln sich neue Datenbank-Technologien, die als Alternative zu relationalen Datenbanken eingesetzt werden können. Wichtig ist, die Eigenschaften dieser neuen Technologien im Kontext des Gesamtsystems, dessen Bestandteil die Datenbank ist, zu verstehen.

In-Memory-Datenbanken (IMDB) halten Daten primär im Hauptspeicher eines Rechners, um dadurch schnellere Zugriffszeiten auf die gespeicherten Daten zu ermöglichen, da ein Hauptspeicher in der Regel wesentlich höhere Zugriffsgeschwindigkeiten und effizientere Zugriffsalgorithmen als eine Festplatte vorweisen kann. Jedoch gehen bei einem Systemausfall Daten im Hauptspeicher verloren. Ein wesentliches Merkmal von Datenbanken ist Transaktions-Konsistenz, die durch Persistenz der Daten erreicht wird. IMDBs stellen dazu folgende Methoden bereit:

- Zustände der Daten werden in Zeit-Intervallen erfasst und auf persistente Speichermedien geschrieben (Snapshots). Daten, die zwischen diesen Zeit-Intervallen anfallen, können verloren gehen.
- Transaktions-Protokolle werden ausgelesen und auf persistente Speichermedien geschrieben (Replikation):
  - Bei einer asynchronen Replikation sind Transaktion und Persistenz getrennt, was eine hohe Performanz und auch ein hohes Risiko des Datenverlusts mit sich bringt.
  - Bei einer synchronen Replikation sind Transaktion und Persistenz zusammengeführt, was bei Schreibzugriffen zu längeren Laufzeiten führen kann, dafür hohe Datensicherheit durch eine ACID-konforme Transaktionssteuerung sicherstellt (siehe http://de.wikipedia.org/wiki/ACID).
- Der Einsatz von NVRAM-Speicher („Non-Volatile Random-Access Memory“, Hauptspeichermodule mit eigener Stromversorgung), der bei Systemausfällen den letzten Datenzustand wiederherstellen kann.
- Hybride Ansätze, die die oben genannten Methoden kombinieren, um damit ACID und Hochverfügbarkeit bei gleichzeitig schnellen Datenzugriffen zu erreichen.

NoSQL-Datenbanken sind strukturierte Datenspeicher, die keine relationalen Algorithmen wie Datenmodelle in der dritten Normalform verfolgen und teilweise auch auf SQL oder ACID verzichten. Sie skalieren horizontal und arbeiten verteilt, damit werden Ziele wie schnelle Zugriffe, Hochverfügbarkeit oder niedrige Hardwarekosten erreicht. Folgende Ansätze werden als NoSQL-Datenbanken bezeichnet:

- „Key-Value-Paare“ sind einfache Lookup-Strukturen, wobei die Schlüssel je nach Datenbank-Hersteller auch gruppiert oder erweitert werden. Diese können In-Memory oder auf Festplatte gespeichert sein.
- „Columnar“ speichern Daten spaltenorientiert und besitzen die Eigenschaften von NoSQL. Dies unterscheidet diesen Ansatz von den relational spaltenorientierten Datenbanken.
- „Document“ verwenden auch den Key-Value-Ansatz, wobei der Wert ein Dokument darstellt.
- „Graph“ modelliert die Verbindungen zwischen Daten-Objekten, die wiederum als Key-Value-Paare dargestellt sind.

NoSQL-Datenbanken sind für einfache, schnell zugängliche Datenstrukturen bei gleichzeitig hohen Datenvolumen und einer hohen Anzahl an Zugriffen optimiert. Es gibt wenige Restriktionen bei Datenmodellen, Änderungen sind damit auch bei großen Datenmengen leicht umsetzbar.

In **relational spaltenorientierte Datenbanken** sind die Daten spaltenorientiert in Blöcke geschrieben, mit dem Ziel, weniger I/O zu generieren und damit einen schnelleren Zugriff auf Daten zu erreichen. Konventionelle Datenbanken schreiben Daten zeilenorientiert in Blöcke.

Beim Auslesen der Daten werden in spaltenorientierten Datenbanken nur die Spalten gelesen, die in der Query enthalten sind, bei den zeilenorientierten Datenbanken werden hingegen Zeilen – und damit alle Spalten – gelesen. Dies kann bei der hohen Spaltenzahl in Star-Schemata einen entscheidenden Unterschied darstellen.

Zudem komprimieren spaltenorientierte Datenbanken besser als zeilenorientierte, da direkt auf komprimierte Daten zugegriffen wird, ohne diese über die CPU zu dekomprimieren. Durch interne Indizes wird ein schneller Zugriff bei Filtern mit „n>1“ Spalten erreicht, oft werden CPU-Algorithmen direkt von der Datenbank effizient eingesetzt. Spaltenorientierten Datenbanken generieren damit oft weniger I/O als zeilenorientierte, insbesondere bei Reporting und Analyse. Die Eigenschaften der relational spaltenorientierten Datenbanken sind:

- spaltenorientierte Speicherung
- Komprimierung der Daten
- effizienter CPU-Einsatz
- Sortierung der Daten
- hohe Zugriffsgeschwindigkeiten

Das Beispiel in Tabelle 1 verdeutlicht die Speicherung, wobei am Bonus auch die Komprimierung nachvollzogen werden kann.

Zeilenorientierte Speicherung:
Eine Zeile ist zusammenhängend in „n>0“ Blöcken gespeichert:
1,Müller,7000; 2,Meier,3000; 3,Berg, 3000

Spaltenorientierte Speicherung:
Eine Spalte ist zusammenhängend in „n>0“ Blöcke gespeichert, wobei die Daten sortiert nach Zeilen abgespeichert sind:
1,2,3; Müller,Meier,Berg; 7000,3000, 3000

Im Unterschied zu „NoSQL Columnar“ enthalten die relational spaltenorientierten Datenbanken eine wichtige Eigenschaft: Sie bilden Daten relational ab. Dies unterstützt insbesondere Auswertungen und Analysen mit konventionellen SQL-Abfragen, auch die meisten Reporting- und Analyse-Anwendungen unterstützen SQL.

Big Data bedeutet, große Datenmengen in unterschiedlichsten Datenformaten (Bilder, Dokumente,

Geo data etc.) ausreichend schnell zugreifbar und auswertbar zu halten. Man spricht auch von den drei „V"s: „Volume", „Variety" und „Velocity". Relational spaltenorientierte Datenbanken, NoSQL-Datenbanken, „Massive Parallel Systems" und flexible Daten-Schemata sind Bestandteile der Technologie von Big Data.

ID	Nachname	Bonus
1	Müller	7000
2	Meier	3000
3	Berg	3000

Tabelle 1: Beispiel

Weitere Trends

Weitere, hier nur stichwortartig angesprochene Trends sind:

- *Predictive Analytics*
  Computergestützte, statistische Vorhersagemodelle
- *Mobile BI*
  Verteilung von Analysen und Reports auf mobile Endgeräte
- *Business Intelligence 3.0.*
  Kollaboration, interaktive und leicht bedienbare Reporting-Systeme, Vorhersagemodelle, Cloud

Fazit

Business Intelligence wird bei zunehmender Dynamik in der Geschäftswelt und mit einer stetig steigenden Datenmenge immer wichtiger in Unternehmen. Der Artikel hat anhand der zwei Themen „Business Intelligence" und „Data Warehouse" einige Trends vorgestellt, mit der diese Herausforderungen gemeistert werden können.

Klaus Rohrmoser
klaus.rohrmoser@
data2fact.de

Bridge Tables bilden in der dimensionalen Modellierung Dimensionen mit Mehrfach-Attributen (Multi Valued Dimensions) oder rekursive Hierarchien in einer Dimension ab. Diese Erweiterung des Star-Schemas ist zwar mächtig, aber auch komplex in der Anwendung.

Brücken bauen im dimensionalen Modell

Dani Schnider, Trivadis AG

Der Artikel zeigt anhand von konkreten Beispielen, wie Bridge Tables modelliert, geladen und abgefragt werden können, warum Bridge Tables nicht in jedem Fall die beste Lösung sind, wo ihre Risiken liegen und wie diese durch geeignete Alternativen vermieden werden können. Nehmen wir an, die DOAG möchte Auswertungen über die Anzahl von Teilnehmern an den einzelnen Vorträgen an der DOAG-Konferenz machen und erstellt dafür ein Star-Schema mit verschiedenen Dimensionen, unter anderem mit einer Dimension „DIM_SESSION“, in der die verschiedenen Sessions (Vorträge) aufgeführt sind. Die Erstellung eines solchen Star-Schemas stellt kein Problem dar, mit Ausnahme eines kleinen, aber aus Modellierungssicht unschönen Details: Es gibt Vorträge mit mehr als einem Referenten.

Wie immer gibt es mehrere Möglichkeiten, einen solchen Sachverhalt in einem dimensionalen Datenmodell abzubilden. Eine davon ist, die Namen der Referenten als kommaseparierte Liste in einem Attribut abzuspeichern. Diese nicht sehr elegante Lösung ist allerdings schwerfällig für die Abfragen. Andere Varianten sind mehrere Attribute („SPEAKER_1“, „SPEAKER_2“, „SPEAKER_3“) in der Dimensions-Tabelle oder eine separate Dimension „DIM_SPEAKER“, die aus der Fakten-Tabelle mehrfach referenziert wird. Nachteil dieser Lösungen – neben den ebenfalls nicht ganz trivialen Abfragen – ist die Beschränkung auf eine maximale Anzahl von Referenten. Ein pragmatischer Ansatz besteht darin, pro Vortrag einen Haupt-Referenten zu definieren und nur diesen in der Dimensions-Tabelle zu speichern. Diese Lösung ist zwar einfach zu realisieren, führt aber zu fehlenden Informationen bei den Auswertungen.

Multi Valued Bridge Tables

Eine vollständige und einfache Lösung für die Abbildung von Mehrfach-Attributen ist in einem klassischen Star Schema mit Dimensions- und Fakten-Tabellen nicht möglich. Um solche Datenbestände abzubilden, kann jedoch eine weitere Art von Tabellen verwendet werden: die Bridge Table. Wie der Name besagt, bildet diese eine Brücke zwischen zwei Dimensionen oder zwischen einer Dimensions- und einer Fakten-Tabelle. Diese beiden

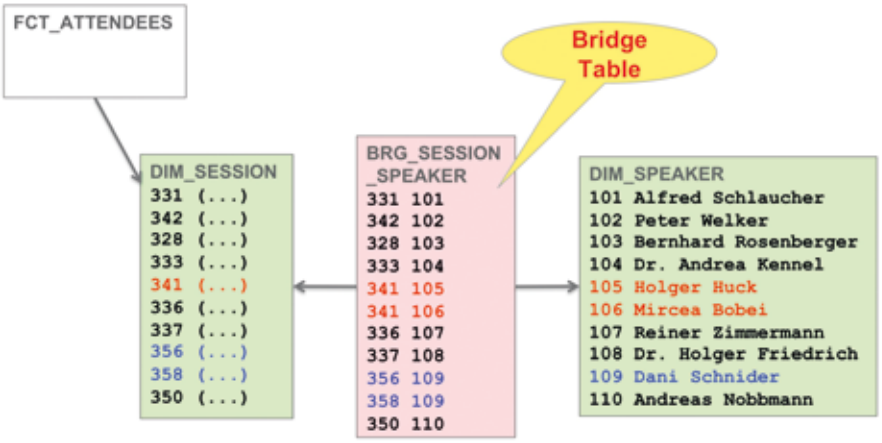


Abbildung 1: Das Beispiel mit Multi Valued Attribute Bridge Table

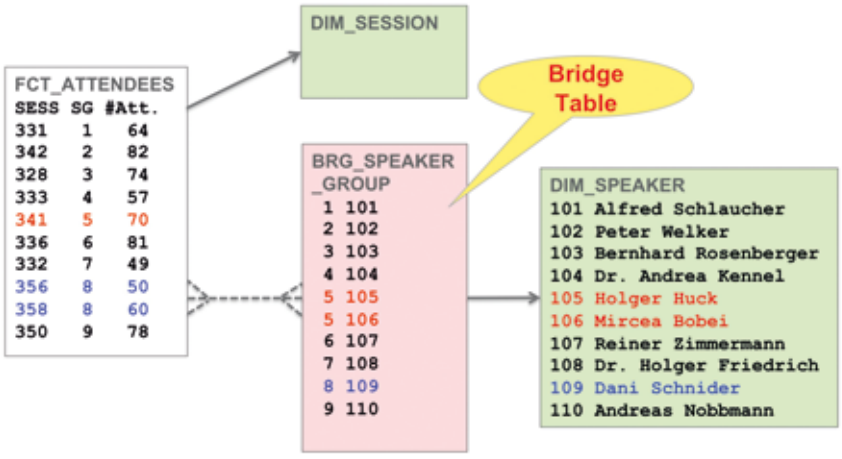


Abbildung 2: Das Beispiel mit Multi Valued Dimension Bridge Table

Möglichkeiten werden anhand unseres Beispiels mit den DOAG-Vorträgen genauer erläutert.

Um das Beispiel zu illustrieren, wurde eine Reihe von Vorträgen aus dem Stream „DWH & BI“ der DOAG-Konferenz 2012 ausgewählt. Die für unser Problem mit den Mehrfach-Attributen interessanteste Session ist dabei der Vortrag „Oracle Essbase Backup & Recovery“. Warum? Weil der Vortrag von zwei Referenten, Holger Huck und Mircea Bobei, gehalten wurde. Um im dimensionalen Datenmodell Sessions mit zwei (oder mehr) Referenten abbilden zu können, wird die Dimension „DIM_SESSION“ durch eine Bridge Table sowie eine zusätzliche Dimensions-Tabelle erweitert (siehe Abbildung 1).

Zusätzlich zur Dimensionstabelle „DIM_SESSION“ wird eine weitere Dimensions-Tabelle „DIM_SPEAKER“ angelegt, in der sämtliche Referenten der DOAG-Konferenz (hier nur ein Ausschnitt) abgespeichert sind. Durch die Bridge Table „BRG_SESSION_SPEAKER“ werden die „n:n“-Beziehungen zwischen „DIM_SESSION“ und „DIM_SPEAKER“ abgebildet, wie wir es aus der relationalen Datenmodellierung kennen. Durch diese sogenannte „Multi Valued Attribute Bridge Table“ lassen sich sowohl Vorträge mit mehreren Referenten als auch Referenten mit mehreren Vorträgen abbilden (Details siehe [1] Seite 205 und [2] Seite 210).

Eine andere Möglichkeit besteht darin, eine Multi Valued Dimension Bridge Table zwischen Fakten- und Dimensions-Tabelle zu verwenden. Dazu ändern wir das Datenmodell unseres Beispiels so, dass die Dimensionen „DIM_SESSION“ und „DIM_SPEAKER“ als unabhängige Dimensionen modelliert und somit separat aus der Faktentabelle referenziert werden. Um Vorträge mit mehreren Referenten abbilden zu können, wird zwischen Fakten- und Dimensions-Tabelle „DIM_SPEAKER“ eine Bridge Table gelegt (siehe Abbildung 2).

Die Einträge in der Fakten-Tabelle „FCT_ATTENDEES“ enthalten die Anzahl der Teilnehmer für die einzelnen Sessions. Anmerkung: Die hier aufgeführten Zahlen sind frei erfunden und entsprechen nicht den tatsächlichen Teilnehmerzahlen. Die Fakten referenzieren jedoch nicht einen einzelnen Referenten in „DIM_SPEAKER“, sondern eine „Speaker Group“ (Attribut „SG“), die in der Bridge Table definiert ist. Auf diese Weise ist es ebenfalls möglich, Vorträge mit beliebig vielen Referenten abzubilden.

Zu beachten ist in diesem Beispiel die „n:n“-Beziehung zwischen Fakten-Tabelle und Bridge Table. Sie verhindert die Definition von Foreign Key Constraints zwischen den Tabellen. Dieses Problem kann jedoch durch eine zusätzliche Dimensions-Tabelle (zum Beispiel „DIM_SPEAKER_GROUP“) mit nur einem Attribut und einem künstlichen Schlüssel gelöst werden, der dann sowohl von der Fakten-Tabelle als auch von der Bridge-Table referenziert wird.

Hohe Flexibilität und hohe Komplexität

Der Vorteil von Bridge Tables liegt in der Flexibilität: Die fachlichen Zusammenhänge mit Mehrfach-Attributen können vollständig abgebildet werden und es gibt keine Limitierung der Anzahl der Werte. Auch ein Vortrag mit zehn oder mehr Referenten könnte in beiden oben erwähnten Daten-Modellen abgebildet werden. Die Flexibilität hat allerdings ihren Preis. Im Falle von Bridge Tables äußert sich dieser durch eine höhere Komplexität, sei es beim Datenmodell („n:n“-Beziehung), in der ETL-Logik oder bei den Abfragen auf das Star-Schema. Hier müssen

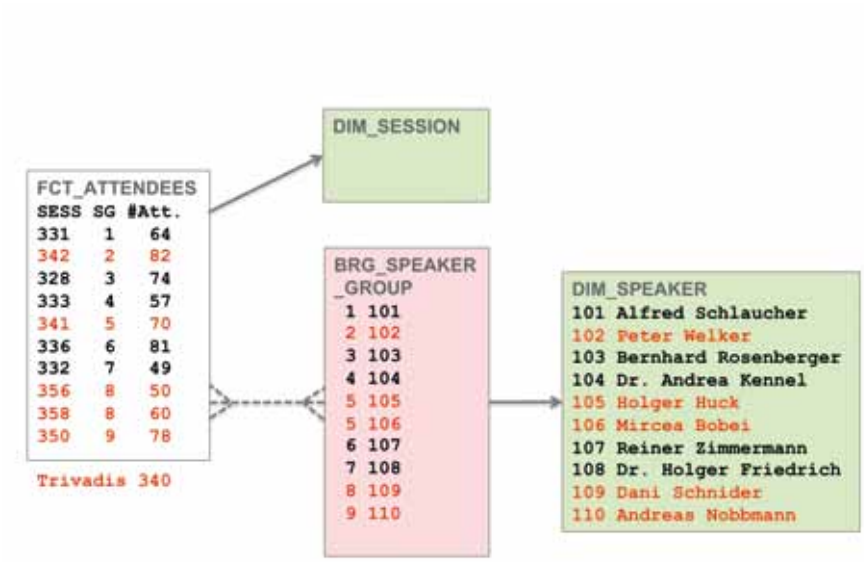


Abbildung 3: Anzahl Vortragsteilnehmer bei Trivadis-Referenten

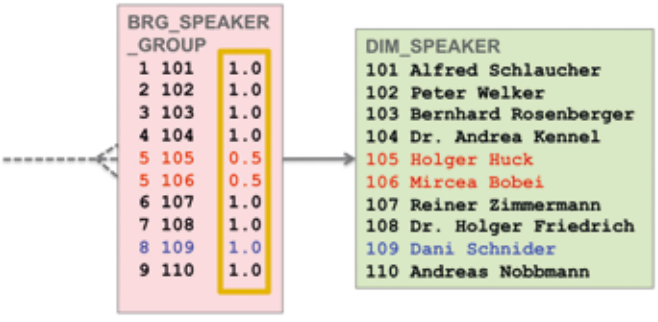


Abbildung 4: Gewichtung der Zuordnungen in der Bridge Table

dann spezielle Vorkehrungen getroffen werden, um Mehrfach-Zählungen zu vermeiden, wie im nächsten Abschnitt beschrieben.

Wo liegt die zusätzliche Komplexität bei den ETL-Prozessen? Neben dem Einfügen oder Ersetzen von Dimensions-Einträgen müssen auch die zugehörigen Datensätze in der Bridge Table bewirtschaftet werden. Das kann zum Beispiel heißen, dass nachträglich ein zusätzlicher Referent für einen bereits angemeldeten und ins DWH geladenen Vortrag angekündigt wird. Dies führt zu einem neuen Eintrag in der Bridge Table. Bei Absage eines Referenten muss die entsprechende Zuordnung aus der Bridge Table gelöscht werden. Lösch-Operationen in einem Data Warehouse gibt es normalerweise nicht – bei Bridge Tables können sie jedoch durchaus zweckmäßig und notwendig sein. Die hier aufgeführten Beispiele gehen von der einfachen Annahme aus, dass keine Historisierung der Dimensionsdaten nötig ist, dass wir es also mit Slowly Changing Dimensions Typ 1 (SCD 1) zu tun haben.

Bei SCD 2 wird es um einiges komplexer. So hat das Einfügen einer neuen Version in die Dimensions-Tabelle auch die Erstellung neuer Versionen aller zugehörigen Einträge in der Bridge Table zur Folge. Eine versionierte Bridge Table wächst dadurch typischerweise sehr rasch, da für jede Änderung eines Dimensionseintrags sämtliche Gruppen-Zugehörigkeiten kopiert werden müssen. Bei Änderungen von Gruppen-Zugehörigkeiten (wie nachträgliche An- und Abmeldungen von Referenten) müssen in der Bridge Table neue Versionen erstellt und teilweise bestehende Einträge kopiert werden. Bei Multi Valued Bridge Tables müssen je nach Art der Änderung auch zusätzliche Versionen in die Dimensions-Tabelle eingefügt werden. Schließlich ist bei Bridge Tables in Kombination mit SCD 2 zu beachteten, dass bei Abfragen immer eine Einschränkung des Datums-Intervalls auf die Bridge Table notwendig ist, da sonst mehrere Versionen aus der Dimensions-Tabelle selektiert werden. Die Einschränkung aufgrund des Joins mit der Fakten-Tabelle, wie sonst bei SCD2-Dimensionen üblich, genügt hier nicht.

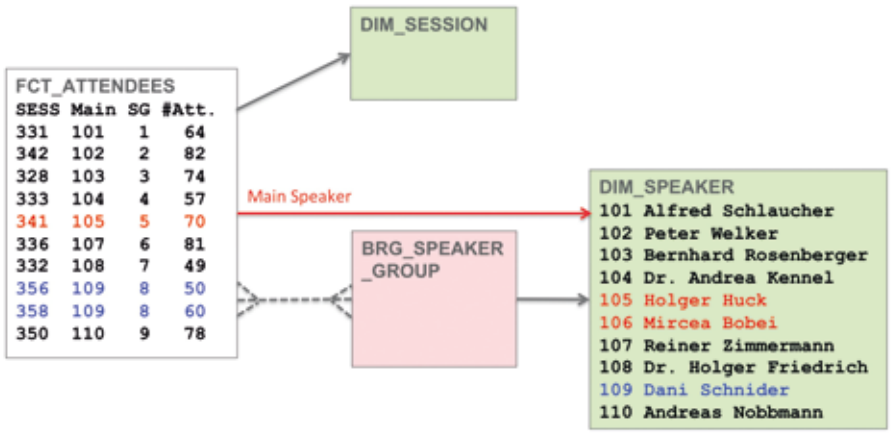


Abbildung 5: Vereinfachung durch View über Bridge Table

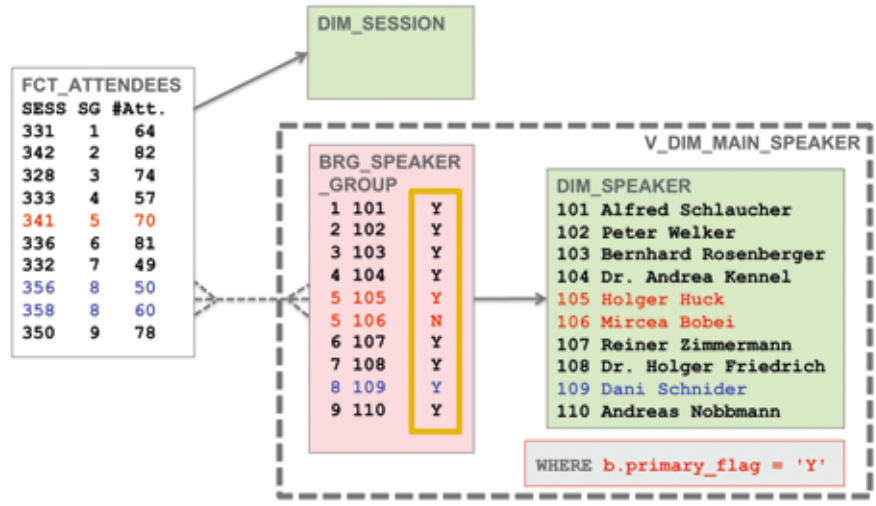


Abbildung 6: Vereinfachung durch zusätzliche Beziehung auf Dimensionstabelle

Abfragen auf Bridge Tables

Der letzte erwähnte Punkt führt uns zu einer wesentlichen Fehlerquelle im Zusammenhang mit Bridge Tables: Mehrfachzählungen bei Abfragen. Um die Problematik zu erläutern, führen wir ein paar SQL-Abfragen auf das Beispielschema aus Abbildung 2 aus. Zuerst möchten wir wissen, wie viele Teilnehmer jeder Referent in seinen Vorträgen hat (siehe Listing 1).

```
SELECT d.speaker_name
     , SUM(f.num_attendees)
  FROM fct_attendees f
  JOIN brg_speaker_group b
ON (b.speaker_group_id =
f.speaker_group_id)
  JOIN dim_speaker d ON
(d.speaker_id = b.speaker_id)
GROUP BY d.speaker_name
```

Listing 1

Die Query liefert für alle Referenten korrekte Resultate. Dass Holger Huck und Mircea Bobei je 70 Zuhörer haben, liegt daran, dass sie einen gemeinsamen Vortrag präsentieren. Aus Sicht jedes einzelnen Referenten ist die ermittelte Anzahl der Teilnehmer korrekt.

Nun möchten wir die Abfrage so ändern, dass die Anzahl der Teilnehmer nicht pro einzelnen Referenten, son-

dern pro Firma, bei der die Referenten angestellt sind, ermittelt wird. Dieser „Drill-Up“ wird üblicherweise so realisiert, dass einfach nach einem anderen Attribut der Dimension – hier nach dem Firmennamen – aggregiert wird (siehe Listing 2).

```
SELECT d.company_name
     , SUM(f.num_attendees)
  FROM fct_attendees f
  JOIN brg_speaker_group b
ON (b.speaker_group_id =
f.speaker_group_id)
  JOIN dim_speaker d ON
(d.speaker_id = b.speaker_id)
GROUP BY d.company_name
```

Listing 2

Doch liefert diese SQL-Abfrage das korrekte Resultat? In der für das Beispiel willkürlich zusammengestellten Liste von Referenten sind „zufälligerweise“ die Hälfte der Personen Trivadis-Mitarbeiter (siehe Abbildung 3).

Werden die (erfundenen) Teilnehmerzahlen der fünf Trivadis-Vorträge zusammengezählt, ergibt die Summe 340 Teilnehmer. Die SQL-Query gibt jedoch als Resultat die Zahl 410 zurück. Wo liegt der Fehler?

Die Ursache liegt bei der Doppelzählung der 70 Teilnehmer, die dem Vortrag von Holger Huck und Mircea Bobei folgen. Da dieser Vortrag von zwei Referenten gehalten wird, ergibt die SQL-Query für diesen Vortrag die doppelte Anzahl an Teilnehmern – also 70 zu viel.

Zur Vermeidung von Mehrfachzählungen wird in der Bridge Table ein zusätzliches Attribut mit einer Gewichtung eingeführt (siehe Abbildung 4). Vorträge mit einem Referenten erhalten die Gewichtung 100 Prozent (beziehungsweise 1.0), bei Vorträgen mit mehreren Referenten wird die Gewichtung prozentual auf die Referenten verteilt – bei zwei Referenten also je 50 Prozent (beziehungsweise 0.5).

Diese Gewichtung wird für die Korrektur von Mehrfachzählungen bei Abfragen auf übergeordnete Aggregationsstufen (wie Referenten einer Firma, eines Landes oder für das Gesamttotal) verwendet (siehe Listing 3). Aber aufgepasst: Bei Abfragen auf der untersten Stufe (Teilnehmerzahl pro Referent) darf die Gewichtung nicht verwendet werden.

```
SELECT d.company_name
     , SUM(f.num_attendees *
b.allocation_factor)
  FROM fct_attendees f
  JOIN brg_speaker_group b
ON (b.speaker_group_id =
f.speaker_group_id)
  JOIN dim_speaker d ON
(d.speaker_id = b.speaker_id)
GROUP BY d.company_name
```

Listing 3

Abbildung 7: Dimensions-Tabelle mit rekursiver Hierarchie

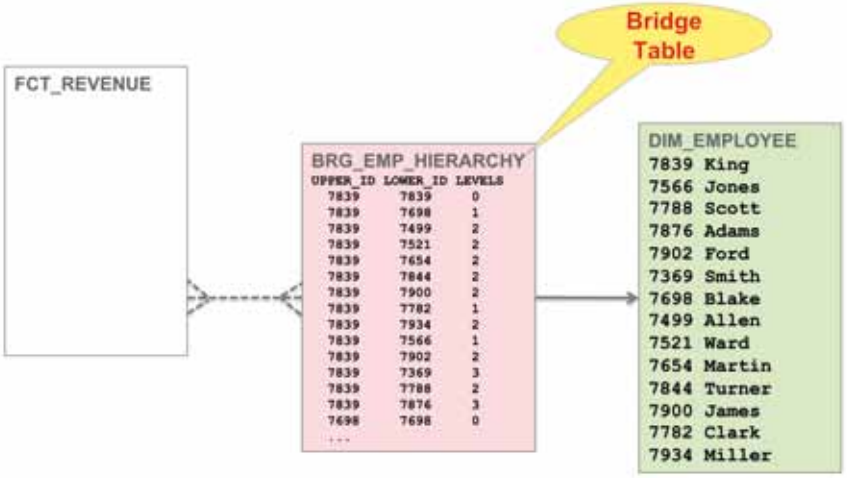


Abbildung 8: Beispiel mit Hierarchy Bridge Table

Vereinfachung der Abfragen

Einmal mehr zeigt sich hier das Dilemma zwischen Flexibilität und Komplexität. Für erfahrene Power-User, die unterschiedlichste Auswertungen nach verschiedenen Kriterien durchführen möchten und in der Lage sind, entsprechende Ad-hoc-Queries zu formulieren, bietet ein Datenmodell mit Bridge Tables zahlreiche Möglichkeiten. Doch die meisten Endanwender – und viele BI-Tools – scheitern an der Komplexität der Abfragen. Hier sind Vereinfachungen gefragt.

Eine Möglichkeit zur Vereinfachung besteht darin, die Komplexität der Bridge Table hinter einer View zu verstecken. Dazu wird die Bridge Table um ein zusätzliches Attribut „PRIMARY_FLAG“ ergänzt. Für jede Referenten-Gruppe ist eine Person als Haupt-Referent markiert. Die View schränkt nun den Datenbestand so ein, dass pro Vortrag nur der jeweilige Hauptreferent angezeigt wird (siehe Abbildung 5). Die meisten Endanwender arbeiten mit dieser View wie mit einer „normalen“ Dimensions-Tabelle. Für spezielle Auswertungen, in denen auch die zusätzlichen

Referenten gefragt sind, wird hingegen direkt auf die Bridge Table und die zugehörige Dimensions-Tabelle zugegriffen.

Als weitere Variante kann eine zusätzliche Beziehung zwischen Fakten- und Dimensions-Tabelle definiert werden, die den Haupt-Referenten jedes Vortrags identifiziert (siehe Abbildung 6). Die Standard-Abfragen der Endanwender verwenden ausschließlich diese Verbindung zur Dimensions-Tabelle „DIM_SPEAKER“, während die Bridge Table nur für spezifische Abfragen durch entsprechend geschulte Power-User zur Anwendung kommt.

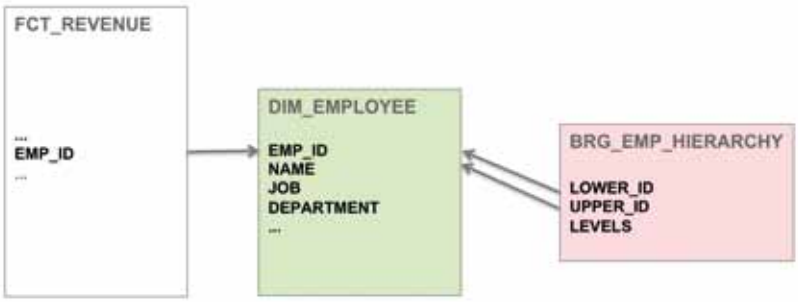


Abbildung 9: Eliminierung der „n:n“-Beziehung einer Hierarchy Bridge Table

Rekursive Hierarchien

Wir haben uns nun ausführlich mit einem Einsatzgebiet von Bridge Tables befasst, nämlich mit der Abbildung von Mehrfach-Attributen in Dimensionen. Daneben gibt es aber noch einen weiteren typischen Anwendungsbereich: rekursive Hierarchien, wie sie zum Beispiel in Mitarbeiter-Organigrammen, Organisationseinheiten, Stücklisten oder Kostenstellen zum Einsatz kommen. Eine rekursive Hierarchie besteht aus Dimensions-Einträgen, die auf übergeordnete Dimensions-Einträge (wie den Vorgesetzten eines Mitarbeiters) verweisen.

Typisch für solche Hierarchien ist, dass die Anzahl der Hierarchie-Stufen nicht fix ist. Eine flexible Möglichkeit besteht in der Implementierung mittels Self-Relationship (auch „Schweinsohr“ genannt), also einer Fremdschlüssel-Beziehung auf die gleiche Tabelle (siehe Abbildung 7). In Oracle SQL lassen sich darauf hierarchische Abfragen ausführen (siehe Listing 4). Neben der Einschränkung, dass diese Abfrage Oracle-spezifisch ist, besteht auch der Nachteil, dass solche Abfragen in vielen BI-Tools nicht oder nur mit erheblichem Aufwand realisiert werden können.

Ein häufig gewählter und bewährter Ansatz besteht darin, die rekursive Hierarchie als flache Dimensions-Tabelle zu implementieren und fehlende Hierarchiestufen durch Wiederholung der übergeordneten Einträge zu füllen (siehe [2] Seiten 224 – 227). In vielen Fällen ist diese Lösung zweckmäßig, hat allerdings die Eigenschaft, dass die Anzahl der Hierarchie-Stufen durch das Design der Dimensions-Tabelle beschränkt wird. Falls diese Einschränkung ein Problem darstellen sollte, lässt sich eine rekursive Hierarchie auch mit einer Bridge Table abbilden.

Hierarchy Bridge Tables

Eine Hierarchy Bridge Table ist eine Tabelle, die für jede Kombination von Dimensions-Einträgen eine Referenz auf den übergeordneten und den untergeordneten Datensatz sowie auf die Anzahl der Hierarchie-Stufen dazwischen festhält. Das Beispiel in Abbildung 8 zeigt eine Mitarbeiter-Dimension, die die 14 Mitarbeiter der altbekannten EMP-Tabelle aus dem Oracle-Beispielschema „SCOTT“ enthält. Um die gesamte Mitarbeiter-Hierarchie abzubilden, sind in der zugehörigen Bridge Table 39 Einträge erforderlich, die nicht alle hier dargestellt sind. Soll nun zum Beispiel der Umsatz aller Mitarbeiter ermittelt werden, die Mr. Jones unterstellt sind, lässt sich dies mit einer einfachen SQL-Abfrage formulieren (siehe Listing 5).

Durch Vertauschen der Attribute „LOWER_ID“ und „UPPER_ID“ der Bridge Table lassen sich auch ähnliche Abfragen formulieren, die die übergeordneten Datensätze aufsummieren (etwa „Mr. Jones und alle seine Vorgesetzten“). Wie Abbildung 8 zeigt, gibt es zwischen der Fakten-Tabelle und der Bridge Table wiederum eine „n:n“-Beziehung. Bei einer Hierarchy Bridge Table kann diese auf einfache Weise eliminiert werden, indem das Datenmodell wie in Abbildung 9 modelliert wird.

Literatur

[1] Ralph Kimball, Margy Ross: The Data Warehouse Toolkit, Second Edition John Wiley and Sons, Inc., 2002, ISBN 978-0471200246

[2] Christopher Adamson: Star Schema, The Complete Reference McGraw-Hill Companies, 2010, ISBN 978-0071744324

Dani Schnider
dani.schnider@trivadis.com

```
SELECT emp_id, name, parent_
emp_id
  FROM dim_employee
 START WITH name = 'Jones'
 CONNECT BY PRIOR emp_id =
parent_emp_id
```

Listing 4

```
SELECT SUM(f.amount)
  FROM fct_revenue f
  JOIN brg_emp_hierarchy b ON
(b.lower_id = f.emp_id)
  JOIN dim_employee d ON
(d.emp_id = b.upper_id)
 WHERE d.name = 'Jones'
```

Listing 5

In Data-Warehouse-Systemen kommen meist Komponenten verschiedener Software-Hersteller zum Einsatz. Dieser Artikel beschreibt das Zusammenspiel zwischen Oracle und SAS im Bereich der Legitimation, insbesondere, wie die technischen und fachlichen Anforderungen zur benutzerspezifischen Beschränkung der Datenzugriffe umgesetzt werden. Die Anforderungen basieren auf den Erfahrungen aus einem Projekt, in dem eine Oracle-Datenbank zur Datenhaltung und die BI-Lösung von SAS für die Auswertungen und Analysen zum Einsatz kommen.

Zusammenspiel von SAS und Oracle beim Steuern von Datenzugriffen

Christian Schütze, metafinanz-Informationssysteme GmbH

Das Prinzip ist einfach: Anwender in verschiedenen Themengebieten (Sales, Human Ressources, Produktion etc.) nutzen eine gemeinsame SAS-Umgebung für Auswertungen und Analysen. Im Infobereich findet man weitere Informationen zu den verwendeten SAS-Komponenten. Die Datenhaltung erfolgt ausschließlich in einer Oracle-Datenbank in Form verschiedener Stern-Schemata (siehe Abbildung 1). Diese enthalten Fakten, Mess-Größen und Kennzahlen, Dimensionen und Merkmale, nach denen Fakten gruppiert werden (siehe Abbildung 2).

In der Fakten-Tabelle „F_ORDERS" sind verschiedene Kennzahlen enthalten:

- ORDER QUANTITY
- ORDER SALES PRICE
- PRODUCT PRODUCTION COSTS

Der Datenbank-Zugriff erfolgt für eine überschaubare Anzahl von Power-Usern über ein persönliches Login auf die Datenbank. Diese Benutzer können zusätzlich SQL-Abfragen erstellen, deren Zugriff auf die Oracle-Tabellen durch Festlegungen der Daten-Eigentümer eingeschränkt ist. Die Berechtigungen auf die Datenbank-Objekte werden über Oracle-Rollen gesteuert, die gemäß den Themengebieten definiert sind. Der Daten-Eigentümer hat die Möglichkeit, in einer Excel-Datei für jedes Datenbank-Objekt Lesezugriffe durch die Oracle-Rollen anzugeben. Diese Informationen werden von der IT in die Datenbank übernommen. Die Anwender mit einem persönlichen Datenbank-Login werden einer oder mehreren Oracle-Rollen zugeordnet.

Der Datenbank-Zugriff aus SAS erfolgt für alle Anwender über einen einzigen technischen User. Das reduziert den Administrationsaufwand, weil eine umfangreiche Benutzerverwaltung nur in SAS notwendig ist. Der technische User hat generell Zugriff auf alle Oracle-Tabellen. Es ist also notwendig, in SAS den Zugriff je Themengebiet zu steuern. Dazu sind je Themengebiet Ordner definiert, deren Zugriff mittels SAS-Rolle abgesichert wird. Jede Tabelle, die in SAS verwendet werden soll, muss in SAS registriert sein. Andernfalls ist sie unbekannt und bleibt unsichtbar. Die registrierten Tabellen werden in den jeweiligen Ordnern in SAS abgelegt.

Die Dimensions-Tabellen wurden gemeinsam in einem Ordner abgelegt, auf den es keine Zugriffsbeschränkung gibt. Die Anwender besitzen ein SAS-Login und sind – je nach Themengebiet – einer SAS-Rolle zugeordnet. Die Dimension „ORGANIZATION" enthält eine Hierarchie aus „REGION" und „COUNTRY". Jeder Anwender soll eine individuelle und flexibel zu definierende Sicht auf die Daten der „ORGANIZATION" erhalten (siehe Abbildung 3). Bestimmte Anwender können somit in ihren Auswertungen auf Regionen wie Europa, einzelne Länder wie Deutschland oder beliebige Kombinationen aus Regionen und Ländern beschränkt werden.

In Auswertungen der Anwender soll diese individuelle Dimension verwendet werden. Um dies zu realisieren, wurde für jede Ausprägung der Hierarchie („REGION" und „COUNTRY") jeweils eine SAS-Rolle definiert. Die Anwender sind einer oder mehreren dieser SAS-Rollen zugeordnet.

Für die Anbindung in den Auswertungen der Anwender wird in Oracle eine eigene Dimension „SAS_ORGA-

Abbildung 1: System-Architektur mit SAS und Oracle

```
DBMS_RLS.ADD_POLICY(
object_schema         => 'SASUSER',
object_name           => 'SAS_ORGANIZATION',
policy_name           => 'POL_ORGANIZATION',
function_schema       => 'SASUSER',
policy_function              => 'F_AUTH_ORGANIZATION',
statement_types       => 'SELECT',
update_check          => false,
enable                => true,
static_policy         => false
);
```

Listing 1

```
DBMS_RLS.ADD_POLICY(
object_schema         => 'SASUSER',
object_name           => 'F_ORDERS',
policy_name           => 'POL_HIDE_COLUMNS',
function_schema       => 'SASUSER',
policy_function              => 'F_AUTH_COLUMNS',
sec_relevant_cols     => 'PRODUCT_PRODUCTION_COSTS',
sec_relevant_cols_opt => dbms_rls.ALL_ROWS
);
```

Listing 2

```
begin
sasuser.PKG_AUTH_SAS_SESSION.sas_login(,&_metaperson','&clientuserid
','&sysuserid');
end;
```

Listing 3

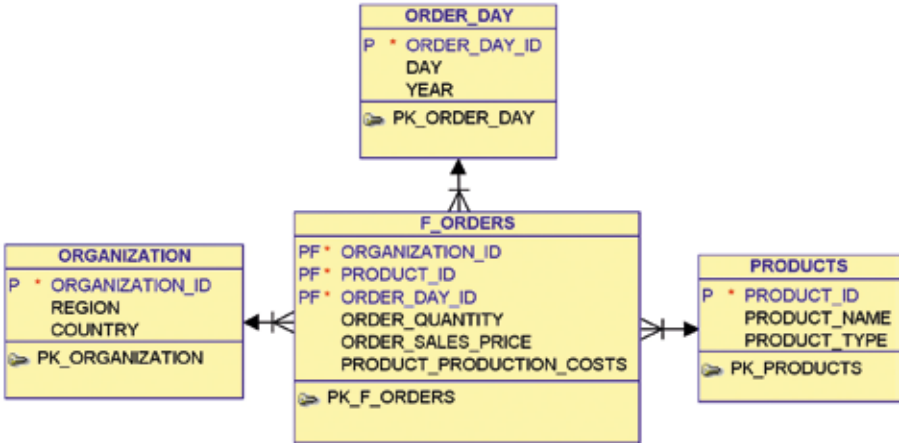


Abbildung 2: Beispiel für Stern-Schemata

REGION	COUNTRY
Europe	Switzerland
Europe	Austria
Americas	United States of America
Europe	Germany
Europe	United Kingdom
Europe	The Netherlands

REGION	COUNTRY
Europe	Switzerland
Europe	Austria
Europe	Germany
Europe	Others

REGION	COUNTRY
Europe	Germany

Abbildung 3: Sichten auf Dimension „ORGANIZATION“

Oracle-Legitimation

Eine Anmeldung auf der Datenbank ist Voraussetzung für den Anwender-Zugriff. Sie geschieht typischerweise durch Angabe von Benutzername und Passwort. Zur Steuerung von Daten-Zugriffen verwendet man in der Regel Oracle-Rollen. Diese haben Zugriff auf Datenbank-Objekte. Mögliche Zugriffsrechte sind „Lesen“ (select), „Schreiben“ (update/delete/insert) oder „Ausführen von PL/SQL-Packages“ (execute). Die Datenbank-Anwender bekommen eine oder mehrere Rollen zugewiesen.

Für die Einschränkung von Daten-Inhalten existiert in Oracle-Datenbanken das Feature „Virtual Private Database“ (VPD). Es erlaubt die Beschränkung der Spalten-Inhalte und Datensätze. Zur Umsetzung wird eine Policy pro Tabelle definiert, die auf eine Policy-Funktion verweist. Diese wird bei jedem Zugriff ausgeführt und erzeugt zusätzliche Filter beziehungsweise blendet Spaltenwerte aus.

In unserem Beispiel führen zwei Anwender unabhängig voneinander die gleiche Abfrage aus (siehe Abbildung 6). Auf dieser Tabelle existiert eine VPD, die die Daten filtert. Beim Anwender „A“ wird durch VPD zusätzlich die Beschränkung auf „AT“ ergänzt. Im Ergebnis erhält der Anwender eine Zeile mit den entsprechenden Daten.

Anwender „B“ hat keinen Zugriff, angedeutet durch die Bedingung „1=2“. Dadurch ist die Ergebnismenge für ihn leer.

select REGION, COUNTRY, "USER" from sas_organization

REGION	COUNTRY	USER
Europe	Switzerland	Müller
Europe	Austria	Müller
Europe	Germany	Müller
Europe	Switzerland	Schmidt
Europe	Austria	Schmidt
Europe	Germany	Schmidt
Europe	Others	Müller

Abbildung 4: Spezielle Dimension „SAS_ORGANIZATION“ in Auswertungen

NIZATION“ erstellt (siehe Abbildung 4). Diese beinhaltet die individuellen Sichten auf die Struktur für jeden Anwender. Beim Befüllen der neuen Dimension mittels PL/SQL werden die Zuordnungen der SAS-Rollen je Anwender berücksichtigt.

Um sicherzustellen, dass beim Ausführen von Berichten und Analysen die korrekte Sicht verwendet wird, kommt Virtual Private Database (VPD) zum Einsatz. Allgemeine Informationen zur Legitimation in Oracle findet man im Infobereich. Zuerst wird die notwendige Policy auf dieser Tabelle definiert (siehe Listing 1).

Die zugehörige Policy-Funktion „F_AUTH_ORGANIZATION“ filtert für alle Anwender die Datensätze. Der technische User „SASUSER“, in dem die Tabelle liegt, erhält keine Beschränkung und sieht alle Datensätze. Für jeden weiteren User wird auf der Spalte „USER“ ein Filter angewendet.

In den Auswertungen der Anwender wird immer die „SAS_ORGANIZATION“ verwendet. Die Originaltabelle „ORGANIZATION“ wird für die ETL-Prozesse genutzt und dient als Basis für die „SAS_ORGANIZATION“. Einige Fakten-Tabellen und Berichte werden von mehreren Anwendern unterschiedlicher Themenbereiche verwendet. Allerdings sollen nicht alle enthaltenen Kennzahlen auswertbar sein. Im Beispiel enthält die Tabelle „F_ORDERS“ verschiedene Kennzahlen:

- ORDER QUANTITY
- ORDER SALES PRICE
- PRODUCT PRODUCTION COSTS

Sie wird von Anwendern der Themenbereiche „Sales“ und „Produktion“ verwendet. Allerdings soll „Product Production Costs“ ausschließlich für Anwender aus dem Themenbereich „Produktion“ sichtbar sein. Um die gesamten Daten zu sehen, müssen die Anwender Mitglied der SAS-Rolle „SAS PRODUCTION“ sein (siehe Abbildung 5). Die Umsetzung erfolgte ebenfalls durch VPD. Auf der Fakten-Tabelle wurde eine Policy definiert (siehe Listing 2).

Zur Definition der betroffenen Spalte werden zusätzlich die Parameter „sec_relevant_cols“ und „sec_relevant_

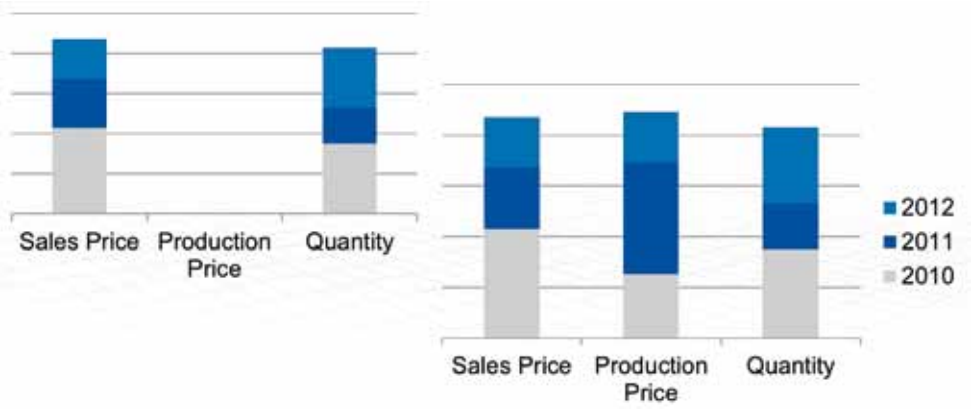


Abbildung 5: Darstellung beim Ausblenden von Kennzahlen in den Berichten für „Sales“ (links) und „Produktion“ (rechts)

Abbildung 6: Funktionsweise von VPD

Überblick SAS

Statistic Analysis System, besser bekannt als „SAS“, bietet verschiedene Komponenten für Daten-Bereitstellungen, Durchführung von Analysen sowie Definition von Berichten. Das SAS Information Delivery Portal kann im Browser dazu genutzt werden, Dashboards zu erstellen oder vorhandene Berichte zu öffnen. Zusätzlich kann das SAS Web Report Studio verwendet werden, um einfache Berichte zu erzeugen. Für komplexere Berichte dient die Windows-Anwendung Enterprise Guide. Basis der Berichte sind sogenannte „Information Maps“. Sie werden im SAS Information Map Studio definiert (siehe Abbildung 7).
„Information Maps“ sind Metadaten-Modelle und erlauben den Anwendern, einfach und ohne Kenntnisse der Datenbank-Struktur SQL-Datenabfragen durchzuführen. Die Möglichkeit einer Business-Sicht auf die Daten ermöglicht es, Informationen in Ordnern zu strukturieren, fachliche Bezeichnungen zu verwenden und Kennzahlen zu definieren.
Datenbank-Verbindungen werden in sogenannten „SAS Libraries“ definiert. Sie enthalten alle relevanten Informationen zum Erstellen der Verbindung. Dies ist vergleichbar mit der Definition einer Verbindung im SQL Developer. Die SAS-Daten (Information Maps, Berichte etc.) sind in den SAS-Metadaten abgespeichert – vergleichbar mit der Ablage von DDL-Informationen in einer Oracle-Datenbank.
Die Steuerung der Zugriffe auf die SAS-Metadaten erfolgt über Zuordnungen von SAS-Rollen und SAS-Gruppen. Der Anwender ist bei entsprechenden „Mitgliedschaften“ in der Lage, bestimmte SAS-Komponenten zu verwenden oder auf Objekte in den SAS-Metadaten zuzugreifen.

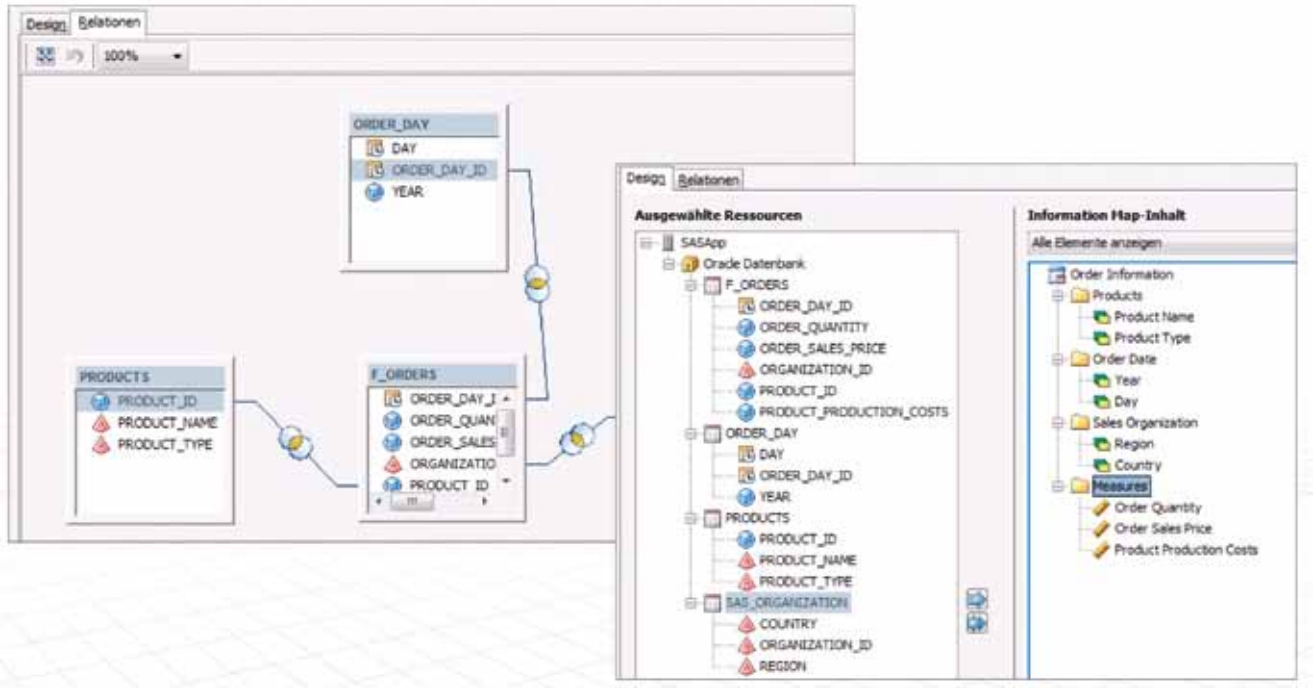


Abbildung 7: SAS Information Map Studio

cols_opt“ gesetzt. In der Policy-Funktion wird geprüft, ob der Anwender Mitglieder der SAS-Rolle „SAS PRODUCTION“ ist. Ist dies der Fall, wird der Inhalt der Spalte angezeigt – andernfalls wird „NULL“ ausgegeben.

Informationsübertragung zwischen SAS und Oracle

Zwei Punkte bleiben noch offen:

- Woher weiß Oracle, welchen SAS-Rollen ein Anwender in SAS zugeordnet wurde?
- Wie weiß Oracle, welcher Anwender in SAS angemeldet ist, wenn der Zugriff immer über einen technischen User erfolgt?

Jede Nacht werden die SAS-Rollen und Anwenderzuordnungen aus den SAS-Metadaten in eine Oracle-Tabelle übertragen. Diese Informationen werden im Rahmen der Erstellung der Dimension „SAS_ORGANIZATION“ beziehungsweise in den Anwendungen der VPD berücksichtigt.

Die Anmeldung aus SAS an der Datenbank erfolgt immer mit dem technischen User. Für die Funktionen der VPD wird allerdings der korrekte Anwender aus SAS benötigt.

In SAS ist in unterschiedlichen Session-Variablen (je nach SAS-Komponente) der Benutzername gespeichert. Beim Aufbau der jeweiligen Datenbank-Verbindung werden eine PL/SQL-Prozedur aufgerufen und die Variablenwerte übergeben.

Oracle ermittelt den korrekten Anwender und hinterlegt ihn in einer Session-Variable. Die Policy-Funktionen lesen die Session-Variable aus (siehe Listing 3).

Fazit

Unter Berücksichtigung der technischen Voraussetzungen und fachlichen Anforderungen konnte eine Lösung gefunden werden, die alle Bereiche erfolgreich abdeckte. Tiefgreifende Kenntnisse der eingesetzten Technologien von Oracle und SAS führten zur beschriebenen Realisierung. Die aufgezeigten Lösungen sind adaptierbar für zukünftig ähnlich gelagerte Anforderungen.

Christian Schütze
christian.schuetze@metafinanz.de

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

Redaktion:
Fried Saacke, Carmen Al-Youssef, Mylène Diacquenod, Dr. Dietmar Neugebauer, Stefan Kinnen, Dr. Frank Schönthaler, Christian Trieb

Titel, Gestaltung und Satz:
Claudia Wagner, Alexander Kermas
DOAG Dienstleistungen GmbH

Titelfoto: Fotolia

Anzeigen:
CrossMarketeam Doris Budwill
www.crossmarketeam.de
Mediadaten und Preise finden Sie unter:
www.doag.org/go/mediadaten

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

Data-Warehouse- und Business-Intelligence-Projekte gibt es seit rund zwanzig Jahren. Nahezu alle Unternehmen jeder Größe und Branche nutzen heute Analyse-Technologien zur aktiven strategischen und/oder operativen Steuerung ihres Geschäfts.

Real-World-Analyse-Szenarien vs. Transformationsflexibilität des Oracle Data Warehouse

Oliver Röniger, ORACLE Deutschland B.V. & Co. KG

Studien zeigen deutlich: Erfolgreiche Unternehmen setzen überdurchschnittlich viel Analyse-Technologie ein und sind damit auch überdurchschnittlich erfolgreicher (siehe [1] Seite 47 und [2]). In der Praxis sind sehr unterschiedliche fachliche Anforderungslagen anzutreffen. Im folgenden Beitrag werden typische Projekt-Anforderungen aus der täglichen Praxis betrachtet und es wird kritisch reflektiert, wie/ob diese in der bestehenden Oracle-Data-Warehouse-Architektur abgebildet werden können.

Langfristiger Erfolg eines Data Warehouse?

Geschäftsprozesse und Geschäftsanforderungen ändern sich ständig – je nach Wettbewerbsdynamik immer schneller. Diese fachliche Volatilität kann nicht ohne technologische Konsequenz für das Analyse-unterstützende Data Warehouse bleiben, es muss nahezu permanent angepasst werden. Zwangsläufig spiegelt das aktuelle Data Warehouse immer nur veraltete Geschäftsanforderungen wider (siehe [3] Seite 47 und Abbildung 1).

Die Fachseite ist unzufrieden, die IT-Seite steht unter Druck. Erfolg kann nur durch gemeinsame Verantwortlichkeiten und eine abgestimmte Projektplanung erzielt werden – sogenanntes „Business IT Alignment" (siehe [4] Seite 93). Der langfristige Erfolg eines Data Warehouse hängt davon ab, inwieweit das Unternehmen in der Lage ist, dieses zu nutzen, um seine strategischen Ziele zu erfüllen [5]. Es geht eben nicht um kleinliche Kosten-Nutzen-Berechnungen mit Blick auf das Data Warehouse, sondern um die Grundsatzfrage, ob es überhaupt noch gebraucht wird.

Für Data Warehouse und Business-Intelligence-Projekte haben sich Reifegradmodelle und Best-Practices-Architekturen herausgebildet und etabliert. Oracle postuliert seit Jahren für Data Warehouses eine datenbankzentrische Ideal-Architektur (siehe Abbildung 2, leicht modifiziert nach [6]).

Ausschlaggebend für dieses datenbankzentrische Paradigma sind gewichtige technische Argumente hinsichtlich enger Integration der Komponenten, einfacherem Betrieb, Datensicherheit, Performance etc. Viele Unternehmen sind diesem An-

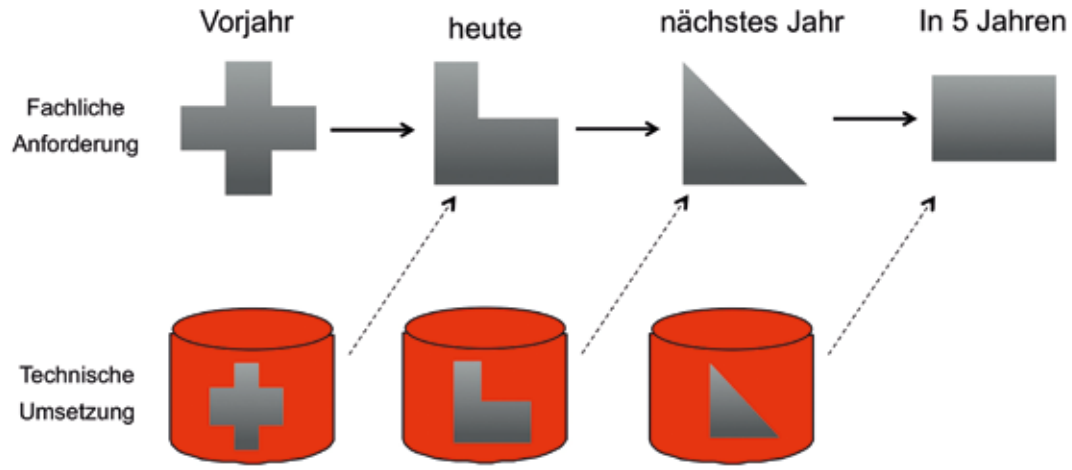


Abbildung 1: Fachliche Anforderung vs. technische Umsetzung

satz gefolgt, Oracle ist einer der Marktführer im Data Warehousing.

Wenn unterstellt werden darf, dass das Data Warehouse den strategischen Geschäftsnutzenbeitrag in der Vergangenheit erbringen konnte, stellt sich nun die Frage nach dessen konkreter Veränderungs- und Transformationsfähigkeit. Dazu sind nachfolgend einige Szenarien skizziert.

Analytische Anforderungen aus der täglichen Praxis

Viele Data-Warehouse-Projektteams sind heute mit einer oder mehreren der folgenden Anforderungen konfrontiert (ungefilterte Wiedergabe aus der Praxis):

ETL untertägig/Real-Time

- *Anforderung*
  Der nächtliche ETL-Prozess genügt nicht mehr, es werden untertägig auch aktuellere Daten für die Analysen benötigt. Teilweise reichen diese Forderungen bis hin zu Real-Time- oder zumindest Near-Time-Analysen (sofortige oder minutengenaue Informationen)
- *Technologie-Antwort Oracle*
  Oracle Golden Gate zur belastungsarmen Real-Time-Datenreplikation oder alternativ selektiver Zugriff über den Oracle BI Server auf das OLTP-System (sofern nur gelegentlich Einzelwerte relevant sind)
- *DWH-Transformationsbedarf*
  - ETL-Modernisierung: geringe Auswirkungen auf das DWH-Kernmodell
  - BI-Server: keine DWH-Architekturänderung
- *Vertiefende Informationen [7], [8]*

Performance

- *Anforderung*
  Die Performance aus Sicht der fachlichen Analyse-Anwender ist zu langsam. „Keine Ad-hoc-Anfrage darf länger als drei Sekunden dauern. Wir sollten uns intensiver mit In-Memory-Technologien beschäftigen."
- *Technologie-Antwort Oracle*
  Exadata (I/O-optimiert) für das DWH-Backend und Exalytics (In-Memory) für das BI-Frontend
- *DWH-Transformationsbedarf*
  Keine DWH-Architekturänderung und keine größeren technisch-fachlichen Anpassungen notwendig
- *Vertiefende Informationen [9]*

Ad-hoc/OLAP

- *Anforderung*
  Die angebotenen Auswertungen sind nicht dynamisch genug, die Fachanwender wollen also frei analysieren und alles mit allem kombinieren. „Wir brauchen OLAP. Die starren Modelle, die wir seit Jahren haben, liefern zwar die notwendige Performance, die Fachseite ist dennoch total unzufrieden und gibt ständig Sonderanalysen in Auftrag, die das IT-Team von der eigentlichen Projektarbeit abhalten, weil sie immer wieder mit Priorität abzuarbeiten sind."
- *Technologie-Antwort Oracle*
  Oracle Essbase oder Oracle OLAP Option anstelle von optimierten Materialized Views
- *DWH-Transformationsbedarf*
  MOLAP-Engine an das DWH anbinden (Essbase) beziehungsweise im DWH integriert (OLAP Option)
- *Vertiefende Informationen [10]*

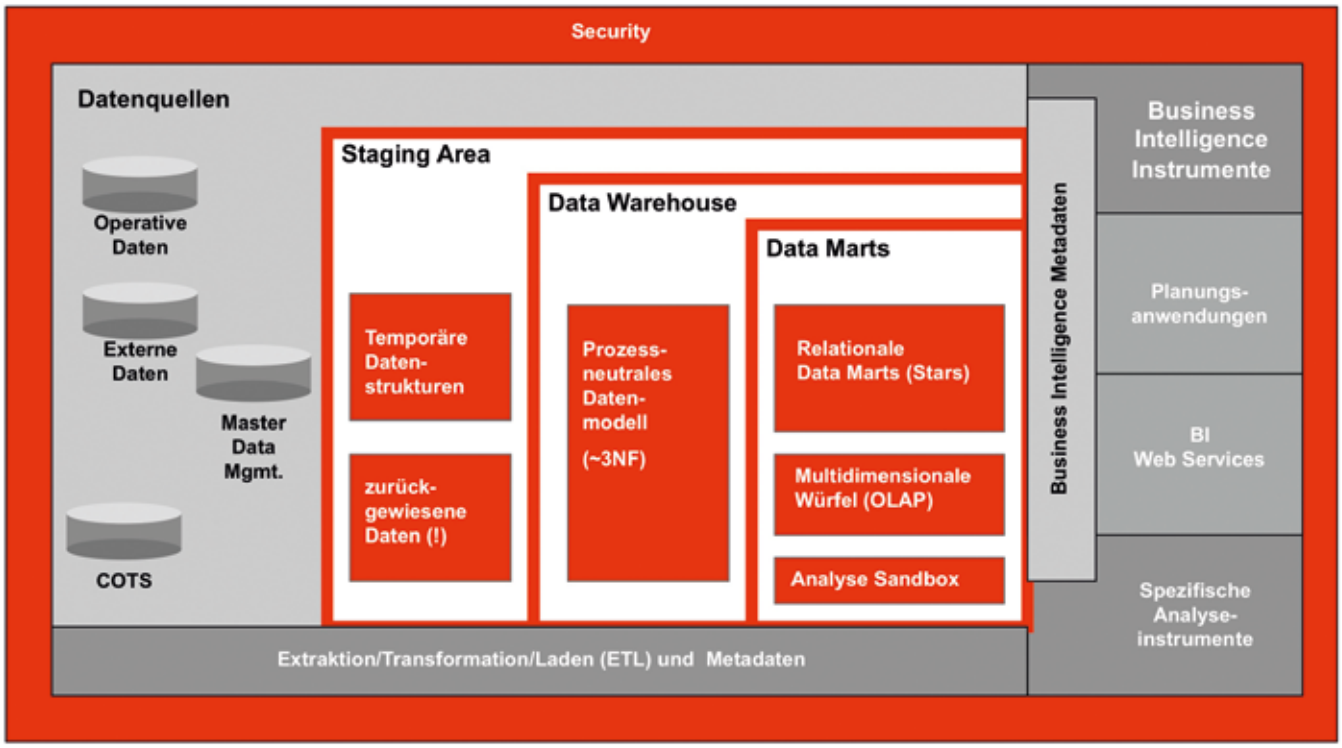


Abbildung 2: Oracle-Data-Warehouse-Architektur

Planung/Simulation

- *Anforderung*
  Das Data-Warehouse-System ist zu rückwärtsgewandt. Neben der Analyse der Vergangenheitsdaten sollen künftig auch Planungen und Hochrechnungen mit dem System möglich sein sowie Simulationen, also mehrere mögliche Zukunftsszenarien parallel. Bisher wird die Planung mit Excel abgebildet, sie soll nun abgelöst und mit dem Data Warehouse enger verzahnt sein
- *Technologie-Antwort Oracle*
  Oracle Essbase als MOLAP-Engine plus Excel-Frontend (SmartView) oder auch komplette Planungsapplikation (Oracle Hyperion Planning)
- *DWH-Transformationsbedarf*
  Zusätzlicher Baustein neben dem DWH, an das Warehouse aber leicht mittels Oracle Data Integrator (ODI) anzubinden
- *Vertiefende Informationen [11]*

Predictive Analytics

- *Anforderung*
  „Auf dem BI-Kongress letzte Woche haben alle nur noch von „Predictive Analytics“ gesprochen. Wir brauchen auch Data Mining und spezifische Algorithmen, nur so können wir auffällige Datenmuster finden, die wir zur Betrugserkennung (Fraud, Compliance) benötigen.“
- *Technologie-Antwort Oracle*
  Oracle Advanced Analytics, bestehend aus Oracle Data Mining und „R“, der Open-Source-Statistik-Bibliothek/-Programmiersprache
- *DWH-Transformationsbedarf*
  Keine DWH-Architekturänderung
- *Vertiefende Informationen [12]*

Unstrukturierte Daten

- *Anforderung*
  Neben den bisherigen Data-Warehouse-Daten sollen auch unstrukturierte Informationen ausgewertet werden, die sich teilweise in Textdokumenten, Webseiten etc. finden. „Auch hinsichtlich Big Data müssen wir eine Strategie entwickeln. Diese Daten sollen gleichwertig in die Analysen einbezogen werden.“
- *Technologie-Antwort Oracle*
  Oracle Endeca Information Discovery (OEID) als ergänzende Analyse- und Such-Engine. Gemischter Zugriff auf externe, unstrukturierte Massendaten und unternehmenseigene Daten (DWH, CRM, ERP)
- *DWH-Transformationsbedarf*
  Zusätzlicher Baustein neben dem DWH
- *Vertiefende Informationen [13]*

SAP-Daten

- *Anforderung*
  „Zur kostenwirtschaftlichen Bewertung unserer Einsätze müssen wir die Daten aus der Finanzbuchhaltung beziehungsweise aus dem Controlling (SAP FI/CO) ins Data Warehouse bringen. Geht das überhaupt mit einem Oracle Data Warehouse?“
- *Technologie-Antwort Oracle*
  Oracle Data Integrator (ODI) besitzt Knowledge-Module in Form von SAP-Adaptern für SAP FI/CO und andere sowie für SAP BW
- *DWH-Transformationsbedarf*
  ETL-Modernisierung: Geringe Auswirkungen auf das DWH-Kernmodell
- *Vertiefende Informationen [14]*

XBRL

- *Anforderung*
  XBRL wird der neue Trend für Datenaustausch-Verfahren, nicht nur bei Fachthemen wie „Solvency II“ und „E-Bilanz“. „Wie können wir derartige XML-ähnliche Daten mit teilweise mächtigen Taxonomien verarbeiten und leicht analytisch auswertbar anbieten?“
- *Technologie-Antwort Oracle*
  Die Oracle XBRL Extension (XML DB) ist Teil der Oracle-Datenbank (Alleinstellungsmerkmal). Der XBRL-Inhalt kann über relationale Views in Richtung DWH weiterverarbeitet oder auch ohne Transformationen direkt mit der Oracle BI Suite analysiert werden.
- *DWH-Transformationsbedarf*
  Abhängig vom Datenintegrations- und Weiterverarbeitungs-Bedarf. Die direkte Analyse der XBRL-Eingangsdaten ist aufwandsminimal möglich.
- *Vertiefende Informationen [15]*

Abbildung 3 zeigt einen Blick auf das technische Gesamtumfeld und verweist auf die skizzierten acht Szenarien.

Fazit

Das Oracle Data Warehouse ist konzeptionell und technologisch robust und flexibel aufgestellt. Neue fachliche Anforderungen können durch Erweiterungen und Fortschreibung mithilfe zusätzlicher Oracle-Technologien in der Oracle-Data-Warehouse-Architektur berücksichtigt werden.

Quellenverzeichnis

[1] Davenport, T. H./Harris, J. G.: Competing on Analytics: The New Science of Winning, Boston 2007

[2] Bettencourt, L. A./Bettencourt, S. L.: Innovating on the Cheap, Harvard Business Review, June 2011, http://hbr.org/2011/06/innovating-on-the-cheap

[3] Inmon, W. H./Strauss, D./Neushloss, G.: DW 2.0: The Architecture for the Next Generation of Data Warehousing, Burlington 2008

[4] Bhansali, N.: Strategic Data Warehousing: Achieving Alignment with Business, Boca Raton 2010

[5] Weir, R.,/Peng, T./Kerridge, J.: Best Practice for implementing a data warehouse: A review for strategic alignment, 5th International Workshop on the Design and Management of Data Warehouses, Berlin 2003

[6] Cackett, D./Bond, A./Lancaster,K./Leiker, K.: Enabling Pervasive BI through a Practical Data Warehouse Reference Architecture, An Oracle White Paper, Februar 2010, http://www.oracle.com/us/solutions/datawarehousing/058925.pdf

[7] http://www.oracle.com/us/corporate/customers/customersearch/ufon-1-goldengate-ss-1865740.html

[8] Röniger, O.: Enterprise Business Intelligence: „nur" IT-Strategie oder ein echter Beitrag zur Unternehmenssteuerung, DOAG News 1/2008, Seite 8-13

[9] http://www.oracle.com/search/customers/browse?Dy=1&Nty=1&Ntk=All&Ntt=nykredit

[10] http://www.oracle.com/us/corporate/customers/customersearch/benxi-municipal-hrss-1-exadata-ss-1735617.html

[11] http://www.oracle.com/us/corporate/customers/customersearch/astrazeneca-1-hyperion-ss-1844164.html

[12] http://www.oracle.com/us/corporate/customers/customersearch/turkcell-1-exadata-ss-1887967.html

[13] Röniger, O./Erb, H.: Analytische Mehrwerte von Big Data, DOAG News 4/2012, Seite 46-50.

[14] http://www.oracle.com/us/corporate/customers/customersearch/gfkl-financial-serv-1-essbase-ss-1378590.html

[15] http://www.oracle.com/technetwork/database-features/xmldb/oracle-xbrlvault-datasheet-131462.pdf

Oliver Röniger
oliver.roeniger@oracle.com

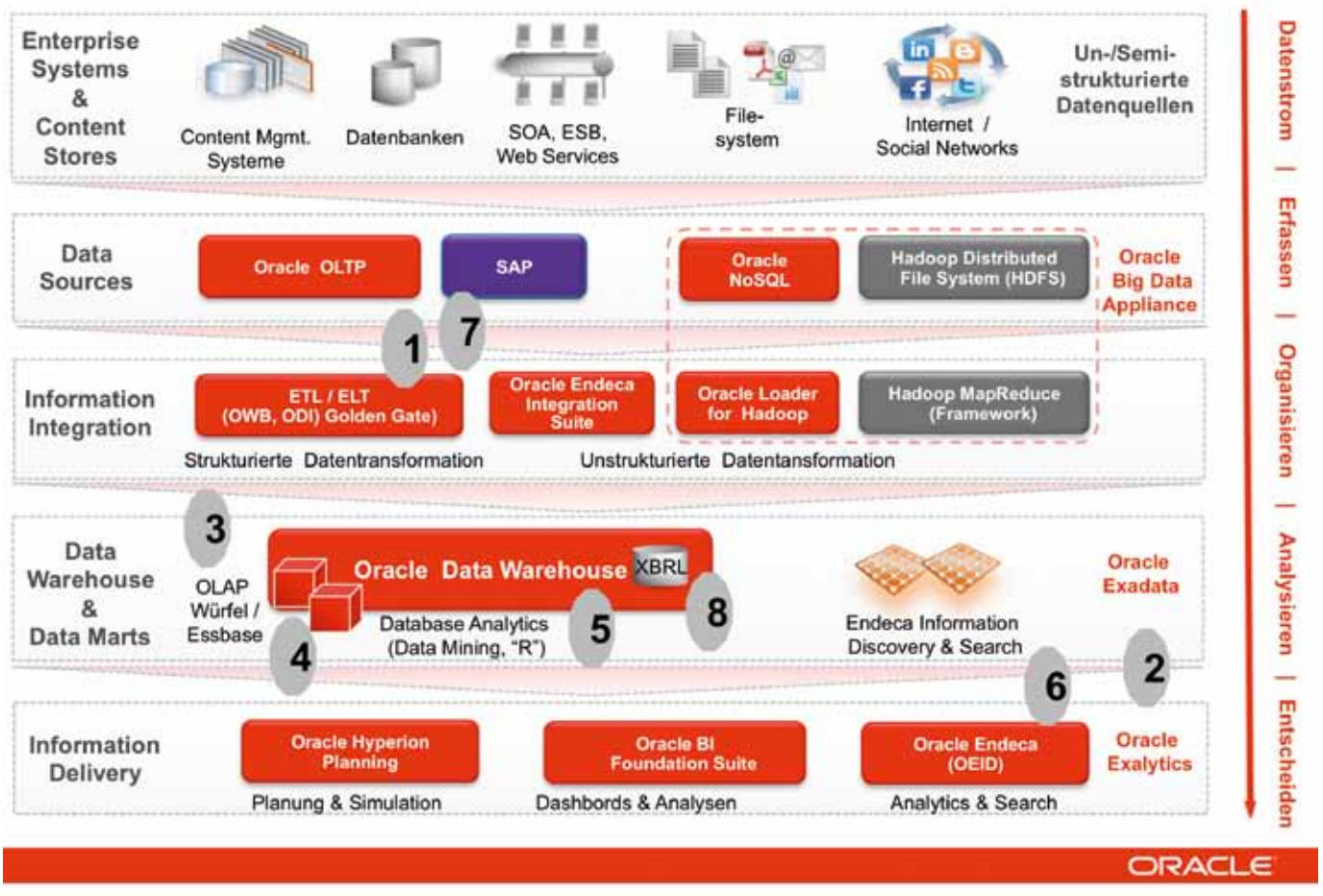


Abbildung 3: Das Big Picture

Zur performanten Ausführung von Berichten und Ad-hoc-Abfragen eines BI-Systems sind beim Oracle Optimizer aussagekräftige und aktuelle Statistiken für die Tabellen und Indizes von essenzieller Bedeutung. Doch die Erstellung der Statistiken benötigt bei großen Datenmengen oft auch sehr viel Zeit. Somit stellt sich die Frage nach einer geeigneten Strategie für die Aktualisierung der Statistiken eines Data-Warehouse-Systems.

Oracle-Statistiken im Data Warehouse effizient nutzen

Reinhard Mense, areto Consulting GmbH

Für die Ausführung eines SQL-Statements untersucht der Oracle Optimizer die möglichen Ausführungspläne und wählt denjenigen mit den geringsten Kosten. Damit der Optimizer die Kosten für einen Ausführungsplan ermitteln kann, benötigt er möglichst detaillierte Informationen über alle beteiligten Tabellen und Indizes. Statistiken stellen diese Informationen zur Verfügung. Anhand derer versucht der Optimizer unter anderem vorherzusagen, wie viele Datensätze in einem Schritt des Ausführungsplans, also etwa bei einem Join, verarbeitet werden müssen. Diese Anzahl an Datensätzen, „Kardinalität" (Cardinality) genannt, bestimmt maßgeblich die Kosten und damit die Auswahl eines Ausführungsplans. Das bedeutet aber auch, dass der ausgewählte Plan nur gut sein kann, wenn die Statistiken dem Optimizer die richtigen Informationen liefern. Da mit jeder Ausführung des ETL-Prozesses umfangreiche Änderungen beziehungsweise neue Daten in das Data Warehouse (DWH) aufgenommen werden, muss großer Wert auf aktuelle Statistiken gelegt werden.

Die folgenden Angaben und Empfehlungen beziehen sich auf die Oracle-Datenbank-Version 11*g* R2. Der Optimizer erfährt mit jeder Datenbank-Version Veränderungen, sodass die hier geschilderten Verhaltensweisen bei älteren, aber auch bei zukünftigen Datenbankversionen zu überprüfen sind.

Welche Statistiken für ein Data Warehouse?

Die typische DWH-Architektur besteht aus einer Staging Area, einem Core und Data Marts. Die Staging Area wird zur temporären Aufnahme der aus den Quellsystemen extrahierten Daten und für die Speicherung von Zwischenergebnissen der Transformationen im ETL-Prozess verwendet. Im Core werden dauerhaft die konsolidierten und homogenisierten Daten historisch gespeichert. Die Data Marts stellen schließlich die Daten in für die Abfragen optimierten Star- oder Snowflake-Schemata zur Verfügung.

Da die Berichte und Ad-hoc-Abfragen insbesondere auf die dafür optimierten Data Marts und bei Bedarf auch auf die Core-Schicht ausgeführt werden, sind für die Datenbank-Objekte dieser Schichten aktuelle Statistiken besonders wichtig. Um eine gute Performance der Abfragen zu erzielen, sollten die Statistiken sowohl für alle Tabellen und Indizes als auch für gegebenenfalls zusätzlich erstellte Materialized Views erzeugt werden.

Auch wenn auf den Tabellen der Staging Area keine Auswertungen erfolgen, ist zu bedenken, dass im Rahmen der ETL-Prozesse intensive Abfragen auf die Tabellen dieser Schichten erfolgen. Insbesondere bei einem Tool wie dem Oracle Warehouse Builder (OWB) werden die ETL-Prozesse vollständig in der Datenbank ausgeführt, sodass auch hier aktuelle Statistiken von großer Bedeutung sind, um eine gute Performance der ETL-Prozesse zu ermöglichen.

Lokale und globale Statistiken

Im Core und in den Data Marts werden die Bewegungsdaten beziehungsweise die Fakten-Tabellen in der Regel partitioniert. Statistiken können sowohl für die einzelnen Partitionen (lokale Statistiken) als auch für die gesamte Tabelle (globale Statistiken) erstellt werden. Das Erzeugen der lokalen Statistiken kann man auf die zuletzt geänderten Partitionen beschränken, sodass der Aufwand relativ gering und nahezu konstant bleibt.

Mit zunehmenden historischen Datenvolumen im DWH wird das Erstellen der globalen Statistiken jedoch immer aufwändiger, da stets die gesamte Datenmenge betrachtet wird. Damit stellt sich die Frage, ob das Erzeugen der globalen Statistiken wirklich notwendig ist. Um diese Frage zu beantworten, muss man die Arbeitsweise des Optimizers betrachten. Greift eine Abfrage nur auf eine Partition zu, werden vom Optimizer lediglich die lokalen Statistiken der einzelnen Partition ausgewertet. Erfolgt jedoch der Zugriff auf mehr als eine Partition, werden sowohl die lokalen als auch die globalen Statistiken vom Optimizer ausgewertet, um die Kardinalität zu bestimmen. Abfragen, die die Daten mehrerer Partitionen auslesen, sind im DWH keine Seltenheit, sodass neben den lokalen auch die globalen Statistiken aktuell zu halten sind.

Ein Beispiel veranschaulicht die Bedeutung aktueller globaler Statistiken: Angenommen, in einer Faktentabelle „FAKT_VERKAUF" sind Verkaufsdaten enthalten und jeden Tag wird eine Million neuer Datensätze aufgenommen. Die „PRODUKT_ID" in der Fakten-Tabelle verweist auf die Produkt-Dimension, bei der sämtliche Änderungen historisiert werden (Slowly Changing

Dimension Typ 2). Die Dimension beinhaltet 100 verschiedene Produkte. Jeden Tag ändern sich jedoch die Attribute von 50 Produkten, sodass aufgrund der Historisierung täglich 50 neue Dimensions-Einträge mit neuen „PRODUKT_IDs“ entstehen. Es wird außerdem angenommen, dass jeden Tag alle 100 Produkte verkauft werden. Täglich nach dem Ausführen der ETL-Prozesse wird die Menge der verkauften Produkte seit dem 1. Januar 2013 ermittelt. Dazu dient das in Listing 1 dargestellte SQL-Statement. Dabei ist <ende> jeweils durch das Datum des zuletzt geladenen Tages zu ersetzen.

Betrachtet man die Kardinalität in Tabelle 1, so erkennt man schnell, dass diese nur bei aktuellen lokalen und globalen Statistiken der exakten Anzahl der tatsächlichen Datensätze entspricht. Werden hingegen nur die lokalen Statistiken erzeugt, kann der Optimizer die korrekte Kardinalität nicht ermitteln. Interessant ist dabei, dass die Existenz der lokalen Statistiken für die stets leere „MAXVALUE“-Partition offensichtlich einen deutlichen Einfluss auf die ermittelte Kardinalität hat.

Werden die lokalen Statistiken für die „MAXVALUE“-Partition erzeugt, wird für die Kardinalität vom Optimizer stets 100 ermittelt. Fehlen jedoch die lokalen Statistiken für die „MAXVALUE“-Partition, nähern sich die ermittelten Werte für die Kardinalität den exakten Werten an.

Man könnte jetzt auf die Idee kommen, das als Workaround zu nutzen, um auf die Erstellung von globalen Statistiken zu verzichten. Listing 2 und Tabelle 2 zeigen jedoch, dass das nicht möglich ist. Verändert man die Abfrage so, dass der abgefragte Zeitraum auch die „MAXVALUE“-Partition umfasst (out of range condition), weichen die Werte für die Kardinalität ohne lokale Statistiken für die „MAXVALUE“-Partition deutlich ab.

Mag das Beispiel zunächst etwas konstruiert wirken, so wird doch deutlich, dass globale Statistiken die Genauigkeit der vom Optimizer ermittelten Kardinalität stark verbessern und sogar zu exakten Kardinalitäts-Werten führen können. Gerade für komplexere Abfragen kann das entscheidend für die Wahl eines guten Ausführungsplans sein.

```
select produkt_id, sum (umsatz)
  from fakt_verkauf
 where datum between to_date (‘01.01.2013‘, ‘dd.mm.yyyy‘)
                 and <ende>
 group by produkt_id;
```

Listing 1: Summe der Umsätze vom 1. Januar 2013 bis zum zuletzt geladenen Tag

		Kardinalität			
Zuletzt geladener Tag und Datum für <ende>	Anzahl Ergebnis-Datensätze	ohne Statistiken	nur lokale Statistiken (nicht für MAXVALUE-Partition)	nur lokale Statistiken (auch für MAXVALUE-Partition)	Lokale und globale Statistiken
01.01.2013	100	989.310	100	100	100
02.01.2013	150	1.912.425	134	100	150
03.01.2013	200	2.868.793	159	100	200
04.01.2013	250	4.220.568	179	100	250
05.01.2013	300	5.878.273	195	100	300
06.01.2013	350	5.602.747	210	100	350
07.01.2013	400	8.255.838	222	100	400

Tabelle 1: Kardinalität für die Abfrage aus Listing 1

		Kardinalität			
Zuletzt geladener Tag	Anzahl Ergebnis-Datensätze	ohne Statistiken	nur lokale Statistiken (nicht für MAXVALUE-Partition)	nur lokale Statistiken (auch für MAXVALUE-Partition)	Lokale und globale Statistiken
01.01.2013	100	889.110	790.371	100	100
02.01.2013	150	2.455.191	1.673.551	100	150
03.01.2013	200	3.309.624	3.672.977	100	200
04.01.2013	250	2.838.101	3.666.418	100	250
05.01.2013	300	5.193.334	4.918.003	100	300
06.01.2013	350	6.776.543	6.988.246	100	350
07.01.2013	400	7.018.066	7.920.052	100	400

Tabelle 2: Kardinalität für die Abfrage aus Listing 2

```
select produkt_id, sum (umsatz)
  from fakt_verkauf
 where datum between to_date ('01.01.2013', 'dd.mm.yyyy')
                 and to_date ('08.01.2013', 'dd.mm.yyyy')
 group by produkt_id;
```

Listing 2: Summe der Umsätze vom 1. bis zum 8. Januar 2013

```
select *
  from dim_produkt
 where produktgruppe = 'Obst'
   and produktkategorie = 'Lebensmittel';
```

Listing 3: Filterung auf korrelierende Spalten einer Produkt-Dimension

Histogramme

Werden die Daten beispielsweise durch eine WHERE-Bedingung gefiltert, versucht der Optimizer vorherzusagen, wie viele Datensätze gefiltert werden. Ohne Histogramme geht der Optimizer von einer Gleichverteilung der Werte aus, das heißt „Kardinalität = Gesamtzahl der Datensätze / Anzahl unterschiedlicher Werte für die Filterspalte" (siehe Abbildung 1).

In der Realität liegt jedoch nicht immer eine Gleichverteilung vor (siehe Abbildung 2), sodass der Optimizer die falsche Anzahl an Datensätzen ermittelt und damit auch die Gefahr besteht, einen ungünstigen Ausführungsplan zu wählen.

Für Spalten, die von der Gleichverteilung der Daten deutlich abweichen, kann es deshalb sinnvoll sein, Histogramme zu erzeugen. Histogramme ermöglichen dem Optimizer, auch für die Spalten mit nicht gleichverteilten Werten, eine genauere Vorhersage der zu erwartenden Datensätze.

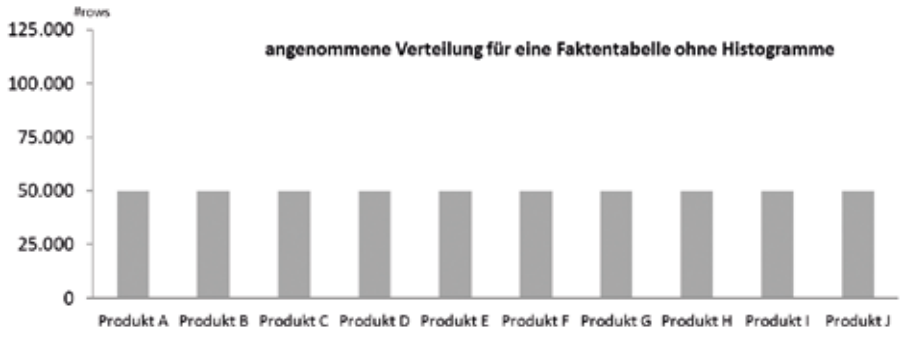


Abbildung 1: Vom Optimizer angenommene Verteilung von Verkaufsdaten ohne Histogramme

Abbildung 2: Tatsächliche Verteilung von Verkaufsdaten

Beim Data Warehouse ist auch zu beachten, dass sich häufig die Verteilung der Daten im historischen Verlauf ändert. So kann sich zum Beispiel die Anzahl der Verkaufs-Datensätze für ein bestimmtes Produkt nach einer durchgeführten Marketing-Kampagne deutlich erhöhen. Aber auch technische Gründe sind für eine Veränderung der Verteilung möglich. So werden bei Änderungen der „Slowly Changing Dimensions" neue Datensätze mit neuen künstlichen Schlüsseln erzeugt, sodass sich in einer Fakten-Tabelle die Verteilung der Produkt-IDs deutlich ändert (siehe Abbildung 3). Deshalb sollte man Histogramme ebenfalls regelmäßig aktualisieren.

Große Datenmengen sind zeitintensiv

Im DWH wird häufig über die Attribute der Dimensions-Tabellen gefiltert, sodass Histogramme für diese Tabellen in Betracht zu ziehen sind. Durch Joins der Dimensions- mit den Fakten-Tabellen findet implizit auch eine Filterung der Fakten-Tabellen statt. Deshalb können Histogramme auch für die Foreign-Key-Spalten der Fakten-Tabellen sinnvoll sein. Da die Erzeugung von Histogrammen bei großen Datenmengen jedoch sehr zeitintensiv sein kann, sollte man Histogramme nur für Spalten mit einer ungleichen Verteilung erzeugen.

Extended Statistics

Erfolgt eine Filterung der Daten über mehr als eine Spalte, nimmt der Optimizer an, dass die Werte dieser Spalten unabhängig voneinander sind. Enthält im DWH eine Produkt-Dimension beispielsweise 25.000 Produkte, die hierarchisch in 500 Produkt-Gruppen und 50 Produkt-Kategorien gleichmäßig verteilt sind, und wird das SQL-Statement aus Listing 3 abgesetzt, so erwartet der Optimizer „25.000 / 500 / 50 = 1 Datensatz" als Ergebnis (siehe Abbildung 4). Da die Werte für Produktgruppen und Produktkategorien aber aufgrund ihrer hierarchischen Abhängigkeit korrelieren, werden tat-

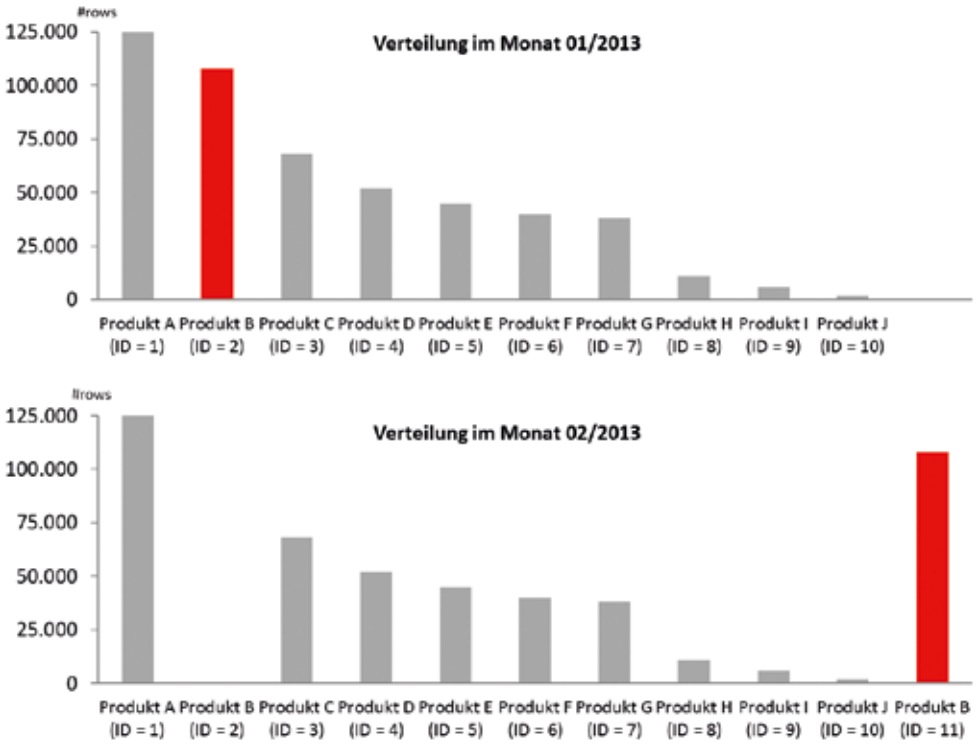


Abbildung 3: Veränderung der Verteilung von Verkaufsdaten durch künstliche Schlüssel einer „Slowly Changing Dimension Typ 2“

```
-------------------------------------------------------------------------------
| Id  | Operation          | Name         | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |              |     1 |    63 |    90   (0)| 00:00:02 |
|*  1 |  TABLE ACCESS FULL | DIM_PRODUKT  |     1 |    63 |    90   (0)| 00:00:02 |
-------------------------------------------------------------------------------
```

Abbildung 4: Kardinalität für korrelierende Spalten ohne Extended Statistiken

```
-------------------------------------------------------------------------------
| Id  | Operation          | Name         | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT   |              |    50 |  3150 |    90   (0)| 00:00:02 |
|*  1 |  TABLE ACCESS FULL | DIM_PRODUKT  |    50 |  3150 |    90   (0)| 00:00:02 |
-------------------------------------------------------------------------------
```

Abbildung 5: Kardinalität für korrelierende Spalten mit Extended Statistiken

ohne Extended Statistics (Laufzeit: 00:00:56.20)

```
-------------------------------------------------------------------------------------------
| Id  | Operation                     | Name           | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT              |                |     1 |    19 |   383   (1)| 00:00:05 |
|   1 |  SORT AGGREGATE               |                |     1 |    19 |            |          |
|   2 |   NESTED LOOPS                |                |       |       |            |          |
|   3 |    NESTED LOOPS               |                |  1073 | 20387 |   383   (1)| 00:00:05 |
|*  4 |     TABLE ACCESS FULL         | DIM_PRODUKT    |     1 |    12 |    90   (0)| 00:00:02 |
|   5 |     BITMAP CONVERSION TO ROWIDS|               |       |       |            |          |
|*  6 |      BITMAP INDEX SINGLE VALUE | BX_PRODUKT_ID  |       |       |            |          |
|   7 |     TABLE ACCESS BY INDEX ROWID | FAKT_VERKAUF  |  1000 |  7000 |   383   (1)| 00:00:05 |
-------------------------------------------------------------------------------------------
```

Abbildung 6: Ausführungsplan bei korrelierenden Spalten ohne Extended Statistiken

sächlich „25.000 / 500 = 50 Datensätze“ als Ergebnis geliefert.

Abfragen dieser Art treten im DWH häufig auf und stellen insbesondere bei Dimensions-Tabellen für den Optimizer ein Problem dar, da die Korrelation der Spalten nicht erkannt wird. Extended Statistics können hier Abhilfe schaffen. Sie erlauben, Statistiken für die kombinierten Werte mehrerer Spalten zu erstellen. Diese Statistiken ermöglichen es dann dem Optimizer, die korrekte Anzahl der Datensätze auch bei korrelierenden Spalten zu ermitteln. Listing 4 erzeugt die entsprechenden Statistiken für das obige Beispiel. Anschließend wird vom Optimizer die korrekte Kardinalität für das SQL-Statement aus Listing 3 ermittelt (siehe Abbildung 5).

Die Bedeutung der Extended Statistics im DWH wird deutlich, wenn man im obigen Beispiel zusätzlich die Fakten-Tabelle mit Verkaufsdaten per Join hinzufügt (siehe Listing 5).

Betrachtet man den Ausführungsplan für diese Abfrage, wird deutlich, dass die Wahl der Join-Methode von der Existenz der Extended Statistics für die Produkt-Dimension abhängt. Ohne Extended Statistics wird für den Zugriff auf die Produkt-Dimension fälschlicherweise eine zu niedrige Kardinalität erwartet, sodass für den Join ein Nested Loop zum Einsatz kommt (siehe Abbildung 6). Mit Extended Statistics hingegen wird die Kardinalität für den Zugriff auf die Produkt-Dimension richtig ermittelt und der Optimizer wählt den performanteren Hash Join (siehe Abbildung 7).

Beim Einsatz von Standard-Berichten für das Reporting werden häufig Filter über korrelierende Spalten definiert. So kann beispielsweise bei einem Umsatz-Bericht der Benutzer zunächst aufgefordert werden, die Produkt-Kategorie und anschließend die zugehörige Produkt-Gruppe auszuwählen. Ohne Extended Statistics für die Spalten können bei solchen aufeinanderfolgenden und voneinander abhängigen Filtern schnell Performance-Probleme für diese Standard-Berichte auftreten.

Bei Ad-hoc-Abfragen stößt der Einsatz der Extended Statistics jedoch an

```
declare
  vResult varchar2 (4000);
begin
  vResult := dbms_stats.create_extended_stats
              (ownname => 'MART'
             , tabname => 'DIM_PRODUKT'
             , extension => '(produktgruppe, produktkate
               gorie)');

  dbms_stats.gather_table_stats
   (ownname => ,MART'
  , tabname => 'DIM_PRODUKT'
  , method_opt => 'for columns (produktgruppe, produktka
    tegorie)');
end;
```

Listing 4: Erzeugung von Extended Statistics für eine Produktdimension

```
select sum (v.umsatz)
  from dim_produkt p
     , fakt_verkauf v
 where p.produkt_id = v.produkt_id
   and p.produktgruppe = 'Obst'
   and p.produktkategorie = 'Lebensmittel';
```

Listing 5: Filterung auf korrelierende Spalten in Verbindung mit einem Join

mit Extended Statistics (Laufzeit: 00:00:02.59)

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	19	7868 (2)	00:01:35
1	SORT AGGREGATE		1	19		
* 2	HASH JOIN		50000	927K	7868 (2)	00:01:35
* 3	TABLE ACCESS FULL	DIM_PRODUKT	50	600	90 (0)	00:00:02
4	TABLE ACCESS FULL	FAKT_VERKAUF	10M	66M	7729 (2)	00:01:33

Abbildung 7: Ausführungsplan bei korrelierenden Spalten mit Extended Statistiken

seine Grenzen, da nicht vorhersehbar ist, welche korrelierenden Attribute die Benutzer in ihren Abfragen in welcher Kombination verwenden. Extended Statistics für sämtliche denkbaren Kombinationen von korrelierenden Attributen zu erzeugen, ist viel zu aufwändig, sodass eine andere Lösung anzustreben ist. Hier kann Dynamic Sampling helfen.

Dynamic Sampling

Wenn keine Extended Statistics vorliegen, kann der Einsatz von Dynamic Sampling den Optimizer trotzdem in die Lage versetzen, die richtige Kardinalität zu ermitteln. Beim Dynamic Sampling werden für das auszuführende SQL-Statement zusätzliche Statistiken für die beteiligten Tabellen ermittelt. Der Datenbank-Parameter „OPTIMIZER_DYNAMIC_SAMPLING" gibt dabei an, in welchem Umfang Dynamic Sampling zum Einsatz kommt. Wird dieser Parameter auf „Level 4" oder höher gesetzt, werden bei Filtern über mehrere Spalten auch Informationen über Korrelationen der Werte dieser Spalten gesammelt. Je höher das Level ist, desto mehr Datenblöcke werden als Stichprobe für das Sammeln der Informationen gelesen (siehe Tabelle 3). Durch Dynamic Sampling benötigt der Optimizer zwar etwas mehr Zeit für das Ermitteln des Ausführungsplans, aber er kann gegebenenfalls den deutlich besseren Plan wählen, was insbesondere im DWH bei Abfragen auf große Datenmengen ein beträchtlicher Vorteil sein kann.

Aufgrund der großen Datenmengen im DWH ist eine möglichst effiziente Erzeugung der Statistiken von großer Bedeutung. Die Dauer dafür hängt wesentlich vom Umfang der für die Analyse verwendeten Stichprobe ab. Für partitionierte Tabellen, wie die besonders großen Fakten-Tabellen, können die globalen Statistiken seit Oracle 11*g* außerdem inkrementell erzeugt werden.

AUTO_SAMPLE_SIZE

Der Umfang der für die Erzeugung der Statistiken zu analysierenden Daten kann mithilfe des „ESTIMATE_PERCENT"-Parameters beim Aufruf der „DBMS_STAT.GATHER_TABLE_STATS"-Prozedur als Prozentsatz (Sample Rate) angegeben werden. Ein niedriger Prozentwert führt zu einem geringen Umfang der Stichprobe für die Analyse. Je geringer der Umfang der Stichprobe, desto schneller erfolgt die Erzeugung der Statistiken, gleichzeitig nimmt man aber auch eine geringere Genauigkeit der Statistiken in Kauf.

Als Default-Wert für „ESTIMATE_PERCENT" ist jedoch nicht ein fester Prozentwert angegeben, sondern der Wert „DBMS_STATS.AUTO_SAMPLE_SIZE". Dieser bewirkt, dass die Oracle-Datenbank selbst den geeigneten Prozentwert für das Erzeugen der Statistiken ermittelt. Wird „AUTO_SAMPLE_SIZE" in der Version 11*g* verwendet, kommt dabei außerdem ein neuer, sehr effizienter Algorithmus für das Erzeugen der Statistiken zum Einsatz. Die von diesem Algorithmus generierten Statistiken haben fast die gleiche Genauigkeit wie die Statistiken auf Basis einer 100-Prozent-Sample-Rate.

Abbildung 8 zeigt die Laufzeiten einer 11*g*-Datenbank für das Erzeugen der Statistiken für eine 40 Millionen Datensätze umfassende Fakten-Tabelle mit unterschiedlichen Sample Ra-

tes im Vergleich zur Verwendung von „AUTO_SAMPLE_SIZE“.

Inkrementelle Statistiken

Das Erstellen der globalen Statistiken ist insbesondere für große Fakten-Tabellen sehr aufwändig und benötigt deshalb oft viel Zeit. Da Fakten-Tabellen jedoch in der Regel partitioniert sind und sich meist nur die Daten einer oder weniger Partitionen ändern, sollte man den Einsatz inkrementeller Statistiken in Betracht ziehen (seit Version 11*g*). Dazu müssen die lokalen Statistiken der einzelnen Partitionen aktuell sein und die inkrementellen Statistiken für die betroffenen Tabellen mit „DBMS_STATS.SET_TABLE_PREFS“ aktiviert werden (Preference „INCREMENTAL“ auf „TRUE“ setzen).

Durch das Aktivieren der inkrementellen Statistiken speichert die Oracle-Datenbank für jede Partition der Tabelle ein sogenanntes „Synopsis-Objekt“ im „SYSAUX“-Tablespace. Dieses enthält statistische Metadaten für die einzelnen Partitionen und Spalten in den Partitionen. Im Vergleich zu den vollständigen Partitionsdaten sind Synopsis-Objekte sehr klein. Wird eine neue Partition hinzugefügt oder eine bestehende Partition geändert, müssen zunächst für diese Partition die lokalen Statistiken aktualisiert werden. Anschließend kann die Oracle-Datenbank anhand der Synopsis-Daten die globalen Statistiken erzeugen, ohne die gesamte Tabelle lesen zu müssen. Dadurch wird die Laufzeit für die Erzeugung der globalen Statistiken erheblich reduziert. Abbildung 9 zeigt die deutlich bessere Laufzeit inkrementell erzeugter gegenüber nicht inkrementell erzeugten Statistiken am Beispiel einer partitionierten Fakten-Tabelle. Betrachtet wird ein Zeitraum von 31 Tagen, in dem jeden Tag 10 Millionen neue Datensätze in die Fakten-Tabelle eingefügt werden.

Wann sollen Statistiken erstellt werden?

Die Oracle-Datenbank bietet die Möglichkeit, die Anzahl geänderter Datensätze zu protokollieren („MONITORING“-Klausel für Tabellen) und mithilfe eines täglich laufenden Jobs die Statistiken der Tabellen auto-

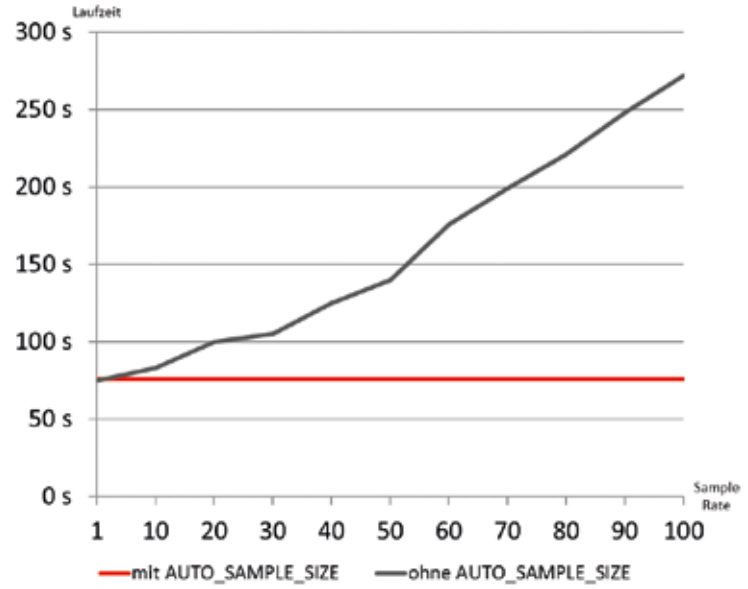


Abbildung 8: Laufzeiten für die Statistik-Erzeugung mit unterschiedlichen Sample Rates im Vergleich zu „AUTO_SAMPLE_SIZE“

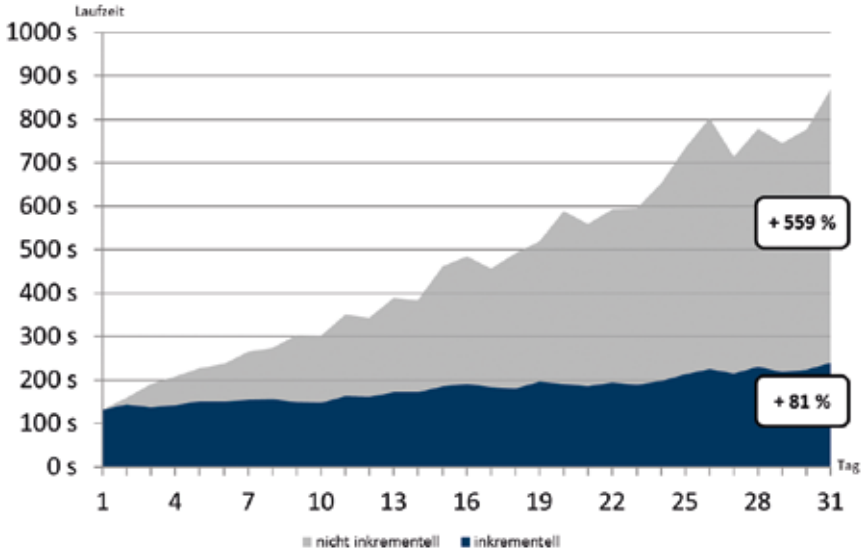


Abbildung 9: Laufzeiten für inkrementell und nicht inkrementell erzeugte globale Statistiken im Vergleich

Level	Umfang der Stichprobe
4	64 Datenblöcke für nicht analysierte Tabellen 32 Datenblöcke für analysierte Tabellen
5	64 Datenblöcke
6	128 Datenblöcke
7	256 Datenblöcke
8	1024 Datenblöcke
9	4096 Datenblöcke
10	alle Datenblöcke

Tabelle 3: Umfang der Stichprobe beim Dynamic Sampling

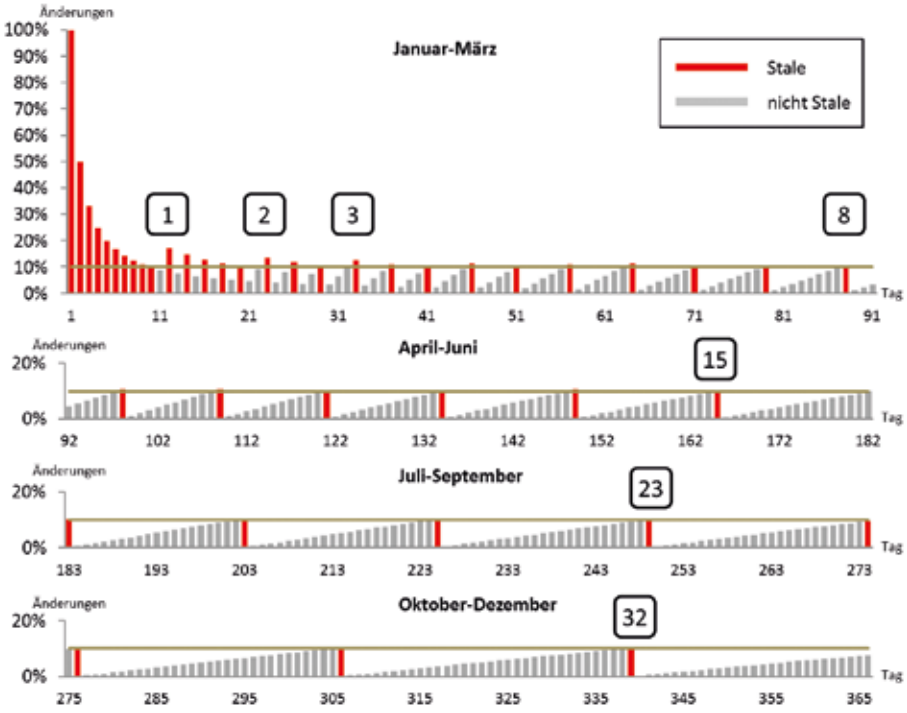


Abbildung 10: Zeitpunkte der Aktualisierung der Statistiken mit „MONITORING“ im Verlauf eines Jahres

matisch zu aktualisieren. Dabei werden nur Tabellen aktualisiert, bei denen sich mindestens ein festgelegter Prozentsatz der Datensätze geändert hat. Die Statistiken werden dann als „Stale“ bezeichnet. Per Default sind 10 Prozent eingestellt. Dieses Verfahren eignet sich für OLTP-Systeme häufig gut, da diese Systeme in der Regel nicht umfangreiche historische Daten vorhalten und somit eine geringe Menge geänderter Daten ausreicht, um das Aktualisieren der Statistiken auszulösen.

Beim DWH hingegen ist mit zunehmender historischer Datenmenge eine immer größere Anzahl an Datensätzen notwendig, um die erforderlichen 10 Prozent Änderungen zu erreichen und somit das Aktualisieren der Statistiken auszulösen. Im DWH kann das insbesondere bei den Fakten-Tabellen dazu führen, dass über einen größeren Zeitraum das Aktualisieren der Statistiken ausbleibt. Abbildung 10 verdeutlicht das Problem anhand eines Beispiels. Dabei wird davon ausgegangen, dass in einer Tabelle (etwa einer Fakten-Tabelle) jeden Tag die gleiche Anzahl neuer Datensätze hinzugefügt wird. In den ersten 10 Tagen werden die Statistiken jeden Tag als „Stale“ angesehen und entsprechend erneuert, aber bereits am 11. Tag wird die erforderliche 10-Prozent-Grenze nicht mehr erreicht und das Aktualisieren der Statistiken somit nicht ausgeführt. Erst am 12. Tag wird diese Grenze wieder überschritten und die Statistiken werden erneuert.

Zu beachten ist, dass die Zeiträume, in denen die Statistiken nicht erneuert werden, mit zunehmenden historischen Datenvolumen immer größer werden. Nach drei Monaten werden bereits 8 Tage lang die Statistiken der Tabelle nicht aktualisiert. Am Ende des Jahres überschreitet der Zeitraum ohne aktuelle Statistiken mit 32 Tagen sogar einen ganzen Monat.

Insbesondere für Berichte und Abfragen, die auch auf die aktuellen Daten des DWH zugreifen, drohen somit aufgrund fehlender Statistiken und unpassender Ausführungspläne erhöhte Laufzeiten. Für DWH-Systeme ist dieses Verhalten nicht akzeptabel. Deshalb sollte das Erzeugen der Statistiken regelmäßig innerhalb des ETL-Prozesses erfolgen. Damit wird sichergestellt, dass die Statistiken stets aktuell sind und die Performance der Abfragen sich nicht verschlechtert.

Fazit

Aktuelle Statistiken sind für das Reporting im DWH unverzichtbar, um performante Berichte und Abfragen zu garantieren. Mit den Möglichkeiten von Oracle 11*g* und der richtigen Strategie ist es möglich, auch für große Datenmengen eines DWH den Aufwand für das Erzeugen aktueller Statistiken gering zu halten. Für eine Strategie zur Erzeugung von Statistiken im DWH sollten folgende Punkte beachtet und anhand der konkreten Situation bewertet werden:

- Stets lokale Statistiken für geänderte Partitionen und globale Statistiken aktualisieren
- Globale Statistiken für partitionierte Fakten-Tabellen inkrementell erzeugen
- Histogramme für Spalten mit ungleichmäßig verteilten Daten, über die häufig gefiltert wird, verwenden
- Extended Statistics für korrelierende Spalten erstellen, die in Standardberichten als Filter genutzt werden
- Dynamic Sampling mit Level 4 oder höher einsetzen, um auch für Ad-hoc-Abfragen gute Laufzeiten beim Filtern über korrelierende Spalten zu erzielen
- Statistiken in der Regel mit „AUTO_SAMPLE_SIZE“ erzeugen. Nur bei sehr großen Fakten-Tabellen gegebenenfalls gezielt kleine Werte für „ESTIMATE_PERCENT“ angeben
- Statistiken im DWH nicht mit dem automatischen Statistik-Job der Datenbank, sondern gezielt im ETL-Prozess erzeugen

Reinhard Mense
reinhard.mense@areto-consulting.de

Das Integrieren neuer Informationsquellen und ihrer Auswertungen gewinnt für Firmen immer mehr an Bedeutung. Dazu gehören auch Daten aus sozialen Netzwerken wie Twitter oder Facebook. Entsprechend steigen die Datenmengen in allen Unternehmensbereichen rasant an.

Social Data Analyse

Norbert Henz, Trivadis GmbH

Die Herausforderung besteht darin, aus dieser Vielfalt an Datenmaterial relevante Informationen herauszufiltern. Es gilt, schnell und einfach neue Erkenntnisse aus einer flexiblen Verknüpfung unterschiedlichster Datenquellen zu gewinnen. Solche Systeme müssen schnell anpassbar, dabei variabel und leicht zu bedienen sein. Der heutige Anwender will nicht mehr monatelang auf seinen Datenzugriff warten.

Mit Oracle Endeca Information Discovery können Daten aus unterschiedlichsten Quellen schnell und einfach zueinander in Bezug gesetzt und dem Anwender umgehend zu Analyse-Zwecken angezeigt werden. Mit seinen „Search and guided Navigation"-Features erlaubt diese Lösung schnelle Antwortzeiten und gleichzeitig die freie Auswahl von Such-Optionen für die Endanwender.

Durch die hohe Flexibilität bei der Daten-Zusammenstellung lassen sich neue Daten zügig zu bestehenden Datastores hinzufügen und stehen dem Anwender somit zeitnah zur Verfügung. Der Artikel stellt anhand eines Lösungsbeispiels die Möglichkeiten von Endeca vor.

Die Ausgangssituation

Die Firma Trivadis sucht immer nach guten, qualifizierten Beratern und stellt entsprechende Anstrengungen an, um geeignete Personen auf sich aufmerksam zu machen. Daher werden die Stellenbeschreibungen nicht nur auf der Homepage veröffentlicht, sondern schon lange zusätzlich unter anderem auch via Twitter beworben.

Als soziales Netzwerk ist Twitter ein ideales Medium, um in direkteren Kontakt mit anderen Menschen zu treten. Mittels der Hashtags, das sind die Textteile mit dem # davor, werden die Meldungen („Tweets" genannt) zusätzlich kategorisiert.

Trivadis verschickt bei Twitter seine Stellenangebote mit dem Hashtag #Jobs. Wer dem Twitter-Stream folgt, wird über unsere Stellenangebote auf diese Art informiert. Aber auch jeder andere Mensch, der bei Twitter nach Jobs sucht, erhält die Anzeigen. Einfach, effektiv und mit wenig Aufwand wird durch Twitter die Reichweite der Stellenanzeigen erhöht.

Die Herausforderung

Aber erreicht man mit dieser Maßnahme auch wirklich jemanden? Wird via Twitter auf die Stellenanzeigen zugegriffen? Wenn ja, wie oft? Um solche Fragen beantworten zu können, muss man die Informationen von Twitter mit den Stellenanzeigen verknüpfen und aus dem Twitter-Stream die relevanten Informationen gewinnen. Nun hat der Twitter-Stream zwar eine Struktur, aber viele interessante Detail-Informationen liegen lediglich in Textform vor. Endeca bietet die Funktionalität, solche unstrukturierten Daten mit strukturierten Daten in einem gemeinsamen Datastore zu verbinden und somit analysierbar zu machen.

Mit der Definition eines Datastores hält der Endeca Server die geladenen Daten im Hauptspeicher. Es kommt also eine In-Memory-Lösung zum Einsatz. Durch diese Form der Datenhaltung im Hauptspeicher und seine sehr flexible Datenstruktur erlaubt Endeca eine schnelle Anpassung an sich ändernde Berichtsanforderungen. Die Entwicklungszyklen können hierbei sehr kurz gehalten sein. Erste Daten laden, ad-hoc auswerten, die Ladestrecke wieder anpassen und erneut analysieren. Das ist schnell machbar und flexibel anpassbar. Das Endeca-ETL-Werkzeug „Integrator" öffnet

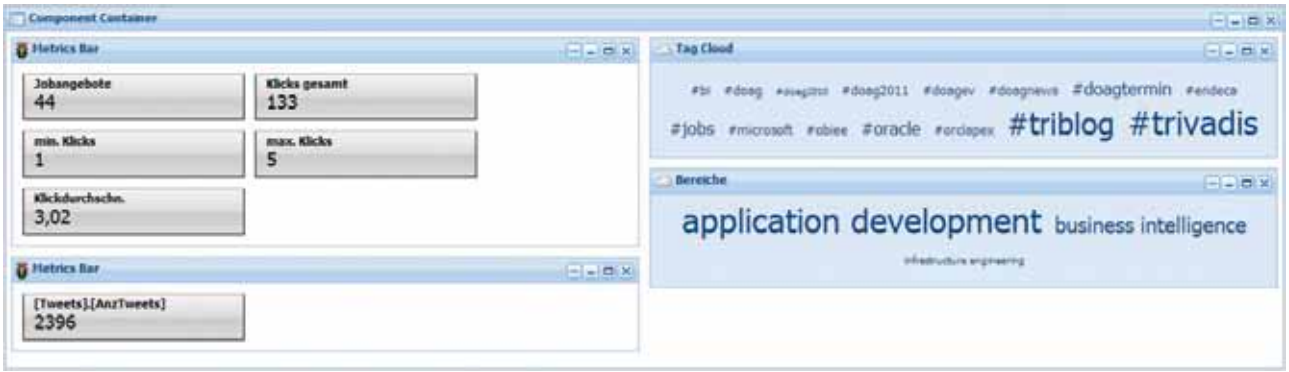


Abbildung 1: Preview auf Twitter-Daten mit Hashtag

die Datenquellen und extrahiert den gewünschten Inhalt. Dahinter versteckt sich eine erweiterte Version des um viele Funktionalitäten ergänzten Open-Source-ETL-Werkzeugs „CloverETL“.

Nach Umformung und Aufbereitung im ETL-Prozess werden die Daten dann in einem Datastore abgelegt. Für die Daten-Transformation liefert Endeca eine umfangreiche Bibliothek an nützlichen Funktionen im Endeca Integrator mit (siehe Abbildung 2).

Die Verknüpfung der Daten aus den verschiedenen Quellen erfolgt ganz simpel über gemeinsame, gleichartige Attribute. Diese ermöglichen dem Anwender bei seinen Auswertungen die übergreifenden Abfragen auf alle geladenen Daten in diesem Datastore. Durch die In-Memory-Technik bietet der Datastore dabei eine hohe Flexibilität und ermöglicht sehr schnelle Abfragen auch bei großen Datenmengen.

Sollten die Quell-Datensätze einmal nicht über die notwendigen Attribute für eine Verknüpfung verfügen, muss der Entwickler eingreifen. Durch Extraktion aus vorhandenen Strings kann er zum Beispiel Teil-Inhalte herauslösen und diese als neues, gemeinsames Attribut zusätzlich in den Datastore laden.

Da die Daten ihre ursprüngliche Daten-Strukturen und -Typen beibehalten, ergibt sich eine sehr unterschiedliche Gesamt-Datenstruktur im Datastore.

Erkennbar ist, dass es keine vordefinierten Tabellen gibt; die Datenstruktur entsteht durch die Daten aus den verschiedenen Quellen von selbst. Die Datensätze einer Quelle müssen dabei noch nicht einmal die gleiche Satzlänge haben. Einzig wichtig für die geplanten Auswertungen sind die gemeinsamen Attribute, hier in der Mitte des Bildes im roten Bereich gezeigt. Aufgrund dieser Schlüssel-Attribute kann später quellübergreifend ausgewertet werden. Ohne diese Verbindungs-Attribute würden die einzelnen Inhalte bezugslos nebeneinander im Datastore liegen.

Ein Beispiel

Zu allererst müssen aus den Twitter-Daten Inhalte extrahiert werden, um als Schlüssel-Attribute oder Filter-Kriterium zur Verfügung zu stehen. Über die Hashtags ist es für den Entwickler sehr einfach, dies zu bewerkstelligen. Die Hashtags verbergen sich im eigentlichen Text einer Twitter-Meldung. Durch einen regulären Ausdruck erlaubt Endeca, diese zu erkennen.

Der Operator „TEXT_TAGGER_REGEX“ im Importer filtert über den regulären Ausdruck die Hashtags heraus und schreibt sie nachfolgend in ein neu definiertes Attribut. In der Vorschau auf die Twitter-Daten sind die Hashtags im Textfeld sehr gut zu erkennen (siehe Abbildung 1).

Nach der Anwendung des Operators sieht man das Ergebnis: Die Hashtags wurden erkannt und separat gespeichert. Damit stehen diese Inhalte jetzt für Auswertungen zur Verfügung. Dies ist nur ein kleines Beispiel für die vielfältigen Möglichkeiten von Endeca Integrator, um Daten aufzubereiten.

Datenauswertung mit Oracle Endeca

Die Auswertungen selbst erfolgen per Web-Browser durch Endeca Information Discovery. In dieser Anwendung kann sich der Anwender flexibel, schnell und interaktiv in den Daten des Datastores bewegen (siehe Abbildung 2). Anpassungen an den Dashboards sind jederzeit selbstständig realisierbar. Durch die geführte Suche in der bereit-

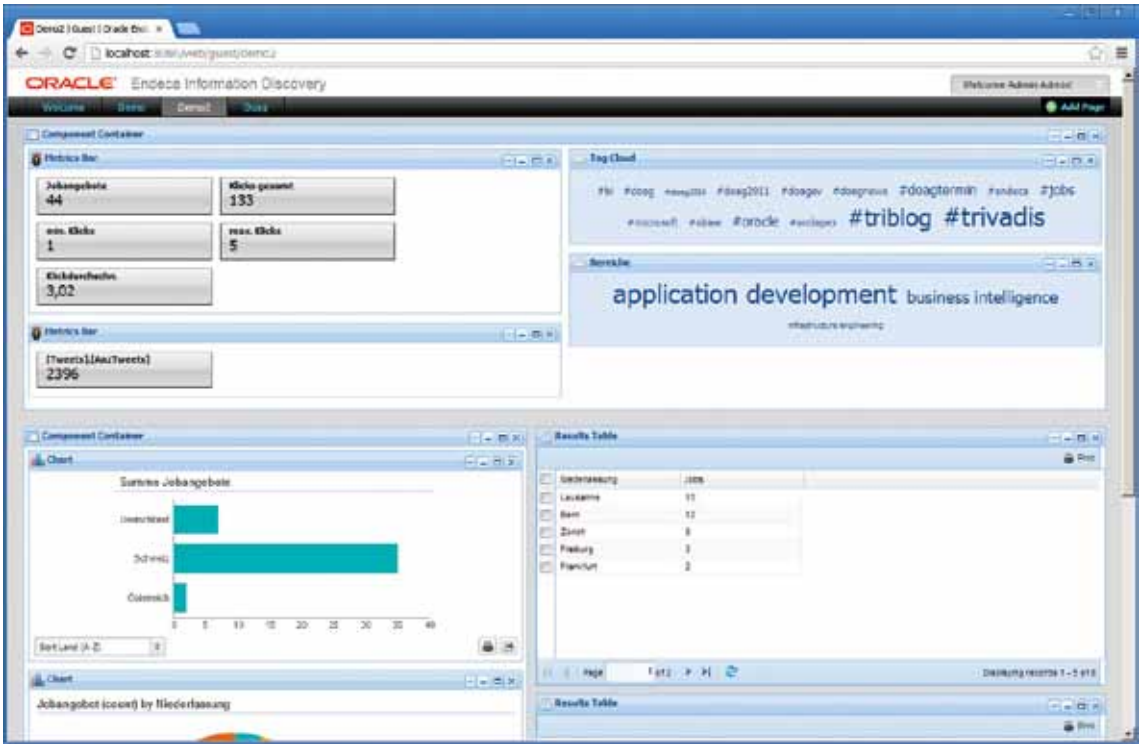


Abbildung 2: Beispiel eines Endeca Dashboards

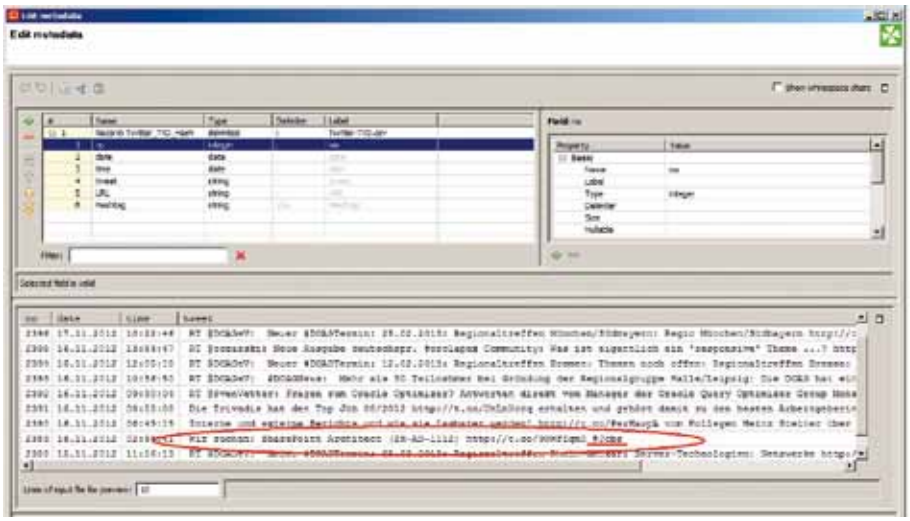


Abbildung 3: Teil einer Auswertung mit Tag-Wolke und Metrik-Auswertung

gestellten Datenwelt bietet Endeca Discovery mehr als nur ein vorgefertigtes Reporting oder eine Analyse. Der Anwender selbst bestimmt die Richtung seiner Auswertungen, er erforscht quasi seine Datenwelt und kann so zu völlig neuen Erkenntnissen kommen.

Für die Trivadis-Stellenanzeigen wurden einige verschiedene Möglichkeiten der Auswertung getestet. So kann man sehr einfach die Anzahl der Anzeigen zählen, aber auch die Zahl der Klicks darauf ermitteln, was schon interessanter ist. Über die Verteilung auf die Fachbereiche erkennt man dann schnell, welche Angebote am meisten Interesse geweckt haben (siehe Abbildung 3).

Zudem lassen sich die Inhalte durch grafische Darstellungen visuell präsentieren. Sehr schön ist die Interaktivität aller Darstellungskomponenten gelungen. Der Anwender wandert quasi durch die neuen Daten und gibt seiner Analyse selbst die gewünschte Richtung.

Von übergeordneten Sicht-Ebenen sind die Details aller Daten immer erreichbar. Auch kann durch Verlinkung jederzeit auf andere Inhalte verwiesen werden, beispielsweise auf die Original-Stellenanzeige.

Fazit

Die Auswertung der Twitter-Daten hat gezeigt, dass die Stellenangebote über diesen Weg sehr wohl Beachtung finden. Daraus haben sich interessante Einblicke und Handlungs-Optionen ergeben.

In zukünftigen Schritten lässt sich auf dieser Basis die Twitter-Auswertung auf weitere Themen ausdehnen. Zusätzliche Datenquellen können zum Datastore hinzugefügt werden und ermöglichen so eine noch tiefere Analyse zusammen mit unseren internen Daten.

Norbert Henz
norbert.henz@trivadis.com

Im Oracle Warehouse Builder Repository sind die Metadaten zu den im Design Center entwickelten ETL-Strecken gespeichert. Darüber hinaus beinhaltet das OWB-Repository Informationen zur Laufzeit von Mappings und Prozessflüssen, Audit-Details etc. Dieser Artikel stellt als Alternative zum Repository Browser die sogenannten „Public Views" vor.

OWB-Repository – individuelle Reports

Ute Middendorf, metafinanz-Informationssysteme GmbH

Eine einfache Möglichkeit zum Zugriff auf die Warehouse-Builder-Repository-Informationen ist der Repository-Browser. Voraussetzung für dessen Nutzung ist jedoch, dass ein entsprechender Listener-Prozess gestartet ist. Darüber hinaus wird eine Freischaltung auf den Port des Repository-Browser-Listener benötigt. Ein Nachteil dabei ist, dass man auf die zahlreichen Informationen des OWB-Repository nur mit den vorgegebenen Reports zugreifen kann.

Grundlage des Repository-Browsers sind die OWB Public Views. Auf die Public Views für Design- und Runtime-Metadaten kann man mittels SQL zugreifen. Diese alternative Schnittstelle zu den Repository-Informationen ermöglicht die einfache Generierung individueller Reports.

Notwendige Berechtigung

Der Repository-Workspace-Owner besitzt alle notwendigen Berechtigungen, um mit SQL die Public Views abzufragen. In der Praxis ist es sicherlich nicht gewollt, dass alle Endanwender den Zugriff auf die Public Views durch den Logon als Workspace-Owner bekommen. Um einem normalen Datenbank-Benutzer die Abfrage der Public Views zu ermöglichen, muss er die Rolle „ACCESS_PUBLICVIEW_BROWSER" erhalten (siehe Abbildung 1). Ohne diese Rolle wird jede SQL-Abfrage der Public Views grundsätzlich mit „Es wurden keine Zeilen ausgewählt"/„0 rows returned" zurückgeben.

Die „ACCESS_PUBLIC_VIEW"-Rolle kann als Workspace-Owner im OWB Design Center zugewiesen werden. Dazu im Global Navigator erst den Sicherheits- und dann den Benutzer-Zweig erweitern; anschließend den Benutzer auswählen, der Zugriff auf die Repository-Browser-Public-Views bekommen soll. Mit einem Rechtsklick den Benutzer bearbeiten (Öffnen) und im Reiter „System Privilegien" das „ACCESS_PUBLICVIEW_BROWSER"-Recht aktivieren.

Für den Fall, dass es mehr als einen Workspace gibt und die Abfragen nicht in dem Default-Workspace lau-

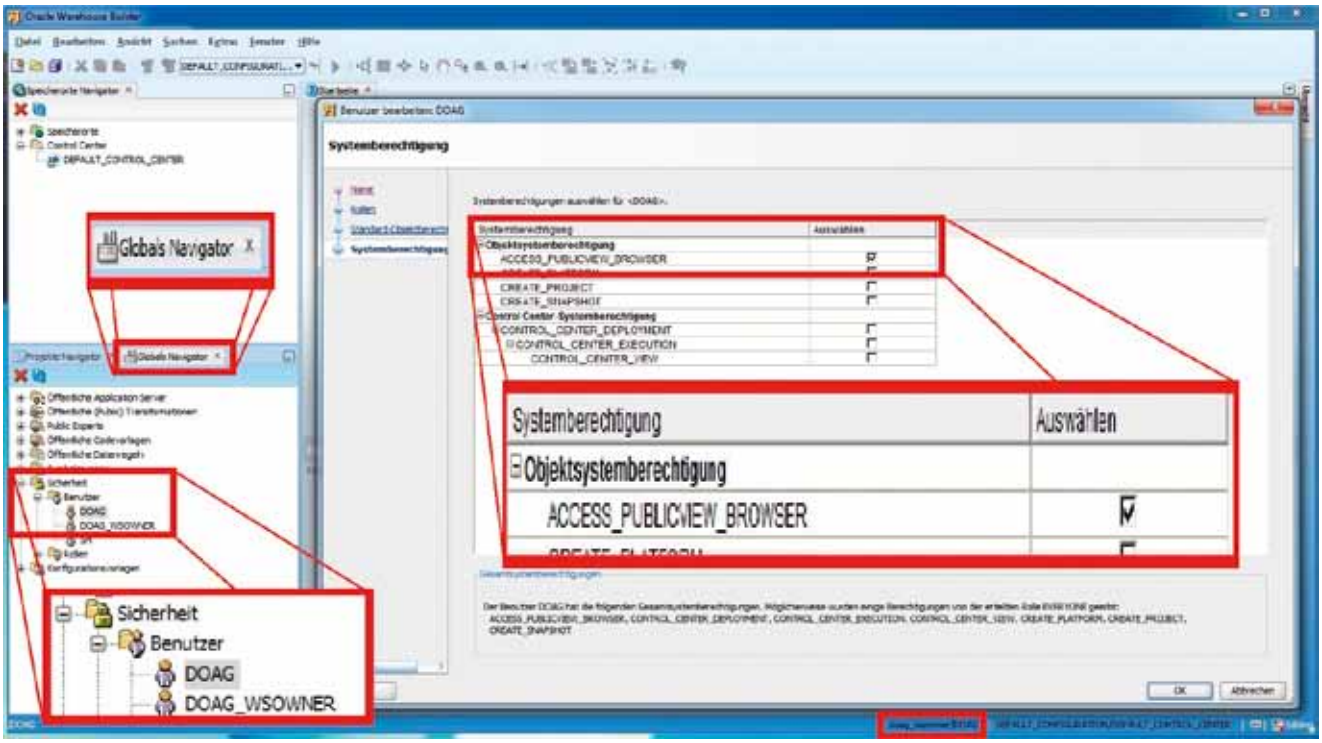


Abbildung 1: „ACCESS_PUBLICVIEW_BROWSER"-Berechtigung vergeben

fen sollen, muss noch zu Beginn der SQL-Session der Workspace mithilfe der Prozedur „WB_workspace_management.set_workspace (<Workspace Name>,<Workspace Owner>)“ gesetzt werden (siehe Listing 1).

Beispiel-Szenario

In den folgenden Abschnitten werden einige der Public Views vorgestellt. Dies geschieht anhand der zwei Mappings „MAP_DOAG_1“ und „MAP_DOAG_2“ (siehe Abbildung 2 und 3). Im ersten Mapping werden die „CUSTOMERS“- und die „COUNTRIES“-Tabellen des SH-Schemas miteinander gejoint. Anschließend findet ein Split nach Geschlechtern in zwei Tabellen „FEMALE“ und „MALE“ statt.

Das zweite Mapping besteht aus einem Deduplikator der Tabelle „FEMALE“ zur „FEMALE_DEDUP“-Tabelle. Die Besonderheit in diesem Mapping ist, dass die „FEMALE“-Input-Tabelle im Mapping „FEMALE_IN“ heißt. Zudem wurde die „FEMALE_DEDUP“-Tabelle so angelegt, dass die Spalte „CUST_FIRST_NAME“ nicht „NULL“ sein darf. Die Spalten „CUST_FIRST_NAME“ und „CUST_GENDER“ wurden absichtlich nicht in den Deduplikator hineingezogen. Das Mapping endet mit einem Fehler.

Zum Design Environment zählen mehr als 200 Public Views, die sich in folgende Bereiche unterteilen lassen:

- General Model Views
- Data Model Views
- Flat Files Views
- Collection Views
- Function Model Views
- Configuration Model Views
- Deployment Model Views
- Mapping Model Views
- Process Flow Model Views
- Profiling Views
- Data Rules Views
- User Defined Object Views
- Expert Views
- Business Intelligence Views
- Real Time Views
- Scheduling Views
- Security Views
- Code Template Views
- Web Services Views
- Others

```
exec owbsys.WB_workspace_management.set_workspace('doag','doag_wsowner');
```

Listing 1

```
SELECT map_id, map_name, is_valid as V, updated_when as U_WHEN,
created_when as C_WHEN, updated_by
FROM all_iv_xform_maps;
```

MAP_ID	MAP_NAME	V	U_WHEN	C_WHEN	UPDATED_BY
79529	MAP_DOAG_1	Y	13.09.12	26.08.12	doag
80400	MAP_DOAG_2	Y	13.09.12	13.09.12	doag

Listing 2

```
SELECT maps.map_name, maps.is_valid as V, comp.map_component_id as ID,
comp.map_component_name, comp.operator_type
FROM all_iv_xform_maps maps
JOIN all_iv_xform_map_components comp
ON maps.map_id = comp.map_id
ORDER BY 1,2;
```

MAP_NAME	V	ID	MAP_COMPONENT_NAME	OPERATOR_TYPE
MAP_DOAG_1	Y	79541	CUSTOMERS	Table
MAP_DOAG_1	Y	79616	COUNTRIES	Table
MAP_DOAG_1	Y	79662	JOINER	Join
MAP_DOAG_1	Y	79940	SPLITTER	Splitter
MAP_DOAG_1	Y	80023	FEMALE	Table
MAP_DOAG_1	Y	80049	MALE	Table
MAP_DOAG_2	Y	80412	FEMALE_IN	Table
MAP_DOAG_2	Y	80439	DEDUPLICATOR	Distinct
MAP_DOAG_2	Y	80463	FEMALE_DEDUP	Table

Listing 3

```
SELECT map_name, map_component_name, data_entity_name
FROM all_iv_xform_map_components
WHERE operator_type = ,Table'
AND data_entity_name = ,FEMALE';
```

MAP_NAME	MAP_COMPONENT_NAME	DATA_ENTITY_NAME
MAP_DOAG_1	FEMALE	FEMALE
MAP_DOAG_2	FEMALE_IN	FEMALE

Listing 4

Ein guter Start zum Erforschen der Public Views bilden die „Mapping Model Views“. Um sich einen Überblick über die bestehenden Mappings eines Workspace zu verschaffen, wird die View „ALL_IV_XFORM_MAPS“ abgefragt (siehe Listing 2).

Meist ist man jedoch nicht nur an einer Übersicht über die einzelnen Mappings interessiert, sondern möch-

```
SELECT deployment_audit_name AS name, repository_user AS user, gene-
ration_time AS gen_time, deployment_audit_status AS status
FROM all_rt_audit_deployments
WHERE deployment_audit_name LIKE ‚%DOAG_1‘;
```

NAME	USER	GEN_TIME	STATUS
MAP_DOAG_1	doag	26.08.12	COMPLETED
MAP_DOAG_1	doag	26.08.12	COMPLETED
MAP_DOAG_1	doag_wsowner	10.09.12	COMPLETED

Listing 5

```
SELECT map_name AS name,
       start_time AS start,
       elapse_time AS e,
       number_errors AS NE,
       run_status AS status,
       number_records_selected AS rec_sel,
       number_records_inserted AS rec_ins,
       number_records_updated AS rec_upd
FROM all_rt_audit_map_runs;
```

NAME	START	E	NE	STATUS	REC_ SEL	REC_ INS	REC_ UPD
MAP_DOAG_2	12.09.12	0	0	COMPLETE	7333	7333	0
MAP_DOAG_2	13.09.12	5	51	COMPLETE	1000	0	0
MAP_DOAG_1	26.08.12	7	0	COMPLETE	55500	55500	0

Listing 6

```
SELECT runs.map_name as name, runs.start_time as start, runs.number_
errors AS ne, executions.return_code AS rc, executions.return_result
AS r_result,          executions.number_task_errors AS no_error,
 executions.number_task_warnings AS no_warn
FROM all_rt_audit_map_runs runs
LEFT OUTER JOIN all_rt_audit_executions executions
ON runs.execution_audit_id = executions.execution_audit_id
ORDER BY map_name DESC, start_time DESC;
```

NAME	START	NE	RC	R_RESULT	NO_ERROR	NO_ WARN
MAP_DOAG_2	13.09.12	51	1	FAILURE	0	51
MAP_DOAG_2	12.09.12	0	0	OK	0	0
MAP_DOAG_1	26.08.12	0	0	OK	0	0

Listing 7

te auch noch deren einzelne Bestandteile kennen. Für diesen Bericht kann die „ALL_IV_XFORM_MAPS“-View mit der View „ALL_IV_XFORM_MAP_COMPONENTS“ verknüpft werden (siehe Listing 3).

Diese Abfrage listet die Komponenten der Mappings genauso, wie sie im OWB zu sehen sind. Für die Komponente „80412“ wird der Name „FEMALE_IN“ angezeigt. Der tatsächliche Name der Tabelle innerhalb der Datenbank steht in der Spalte „DATA_ENTITY_NAME“ der „ALL_IV_XFORM_MAP_COMPONENTS“-View. Mithilfe dieser Spalte lässt sich eine Übersicht über alle Mappings erstellen, die von der Änderung der Struktur einer Tabelle, zum Beispiel „FEMALE“, betroffen sind (siehe Listing 4).

Runtime Environment

Im Gegensatz zum Design Environment ist die Anzahl der Public Views über das Runtime Environment überschaubar. Es gibt je 14 Views zu den Themengebieten „Deployment“ und „Execution“:

Deployment Audit Views
- ALL_RT_AUDIT_LOCATIONS
- ALL_RT_AUDIT_LOCATION_MESSAGES
- ALL_RT_AUDIT_LOCATION_FILES
- ALL_RT_AUDIT_OBJECTS
- ALL_RT_AUDIT_SCRIPT_MESSAGES
- ALL_RT_AUDIT_SCRIPT_RUNS
- ALL_RT_AUDIT_SCRIPT_FILES
- ALL_RT_AUDIT_DEPLOYMENTS
- ALL_RT_INSTALLATIONS
- ALL_RT_LOCATIONS
- ALL_RT_LOCATION_PARAMETERS
- ALL_RT_OBJECTS
- ALL_RT_TASKS

Execution Audit Views
- ALL_RT_AUDIT_EXECUTIONS
- ALL_RT_AUDIT_EXECUTION_PARAMS
- ALL_RT_AUDIT_EXEC_MESSAGES
- ALL_RT_AUDIT_EXEC_FILES
- ALL_RT_AUDIT_MAP_RUNS
- ALL_RT_AUDIT_MAP_RUN_SOURCES
- ALL_RT_AUDIT_MAP_RUN_TARGETS
- ALL_RT_AUDIT_STEP_RUNS
- ALL_RT_AUDIT_STEP_RUN_SOURCES
- ALL_RT_AUDIT_STEP_RUN_TARGETS
- ALL_RT_AUDIT_MAP_RUN_ERRORS
- ALL_RT_AUDIT_MAP_RUN_TRACE
- ALL_RT_AUDIT_PROC_RUN_ERRORS
- ALL_RT_AUDIT_STEP_RUN_ STRUCTS

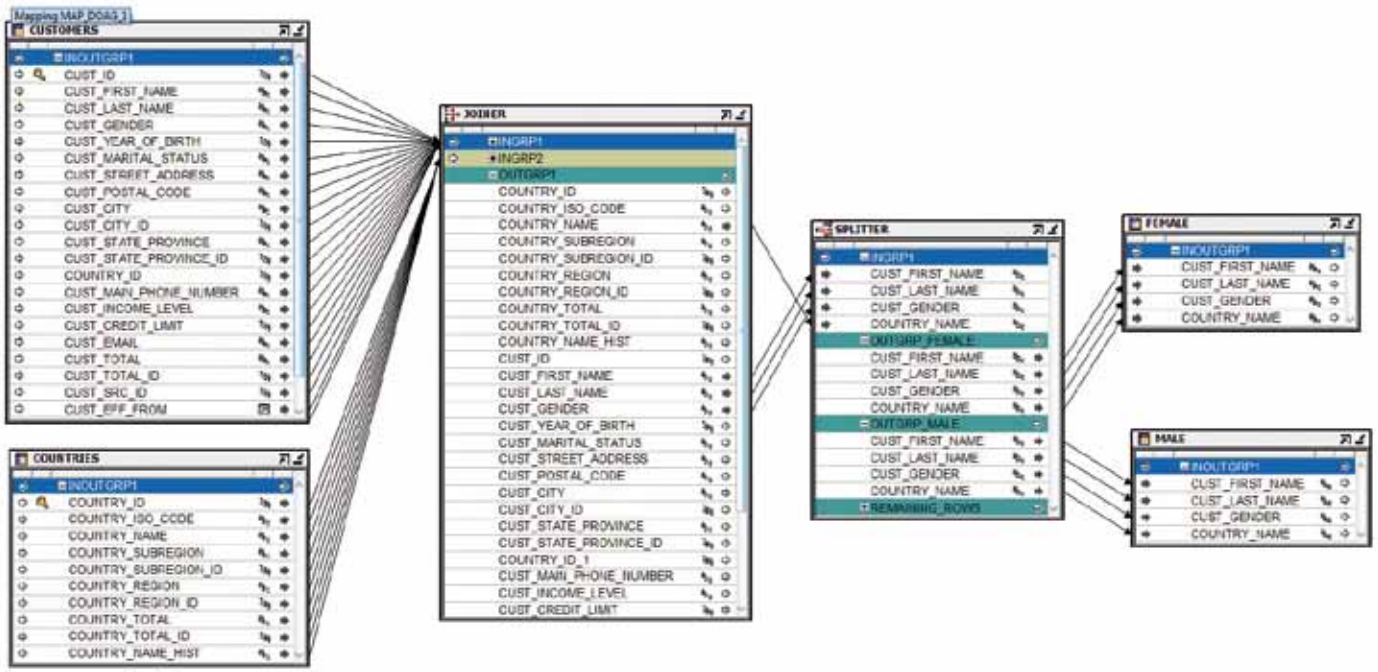


Abbildung 2: Mapping map_doag_1

Im Zusammenhang mit Auditing stellt sich oft die Frage: „Wann wurde durch wen deployt?" Eine Historie von Deployments kann durch eine Abfrage erstellt werden (siehe Listing 5).

Die Public Views, die zum Bereich der Execution Auditing Views zählen, unterstützen bei der Prüfung der Ausführung von Mappings. Neben Ausführungszeiten können auch Informationen über die Anzahl von verarbeiteten Datensätzen in einen individuellen Report aufgenommen werden (siehe Listing 6).

Bei diesem Report ist zu beachten, dass trotz der 51 Fehler während der Ausführung des zweiten Mappings der Status „COMPLETE" angezeigt wird. Der Run-Status gibt also keine Auskunft über den Erfolg oder Misserfolg der Ausführung, sondern lediglich darüber, ob die Ausführung beendet ist oder nicht. Je nach Art der betrachteten Mappings kann man noch die Anzahl der gelöschten (number_records_deleted), der gemischten (number_records_merged) oder der bei einer SQL-Loader-Ladung abgewiesenen (number_records_discarded) Datensätze mit in die Abfrage aufnehmen. Return-Code-Informationen lassen sich mithilfe der View „ALL_RT_AUDIT_EXECUTIONS"

```
SELECT distinct runs.map_name as name, runs.start_time as start,
runs.number_errors AS ne, err.target_name as tname, err.run_error_
number AS err_no, err.run_error_message as err_messsage
FROM all_rt_audit_map_runs runs
LEFT OUTER JOIN all_rt_audit_map_run_errors err
ON err.map_run_id = runs.map_run_id
WHERE trunc(runs.start_time) >
  to_date('11.09.2012','dd.mm.yyyy')
ORDER BY start_time ASC;
```

NAME	START	NE	TNAME	ERR_NO	ERR_MESSAGE
MAP_DOAG_2	12.09.12	0			
MAP_DOAG_2	13.09.12	51	FE-MALE_ DEDUP	-1400	ORA-01400: Einfügen von NULL in („SH"."FEMALE_ DEDUP"."CUST_ FIRST_NAME") nicht möglich

Listing 8

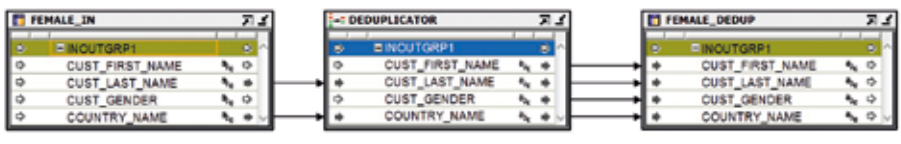


Abbildung 3: Mapping map_doag_2

Vorschau auf die nächste Ausgabe

Das Schwerpunktthema der Ausgabe 03/2013 lautet
Oracle Best Practices auf Infrastrukturen und Plattformen
Sie erscheint am 7. Juni 2013

zusammenstellen (siehe Listing 7). Über diese rein informativen Abfragen hinaus lassen sich Execution Auditing Views auch zur Fehler-Analyse heranziehen. Die View „ALL_RT_AUDIT_MAP_RUN_ERRORS“ beinhaltet Fehlermeldungen zu einem Mapping. Mit dem Statement in Listing 8 lässt sich ein Bericht über die gelaufenen Mappings seit einem bestimmten Tag (alternativ auch „sysdate“) inklusive etwaiger Fehlermeldungen generieren.

Gegebenenfalls mehrfach auftretende Fehler werden nur einmal gelistet. Die Spalte „err.run_error_number“ gibt Auskunft über die Häufigkeit der einzelnen Fehler, während die Spalte „run_error_message“ die ORA-Fehlermeldung enthält.

Fazit

Es gibt zahlreiche Anwendungsbeispiele, wie die individuellen SQL-Reports der Public Views genutzt werden können. Hierzu zählen zum Beispiel die Produktionsüberwachung inklusive Einbindung von Monitoring, Fehler-Analyse, technischer Freigabe, Impact-Analyse, Deployment-Historie und Überprüfung von Namenskonventionen.

Der Repository-Browser ist komplett auf die Public Views aufgebaut. Jede im Repository-Browser angezeigte Information lässt sich somit auch mit SQL-Abfragen generieren. Die Public Views ermöglichen es, Berichte über das Design oder Runtime-Analysen ganz nach den eigenen Bedürfnissen zusammenzustellen. Durch die Möglichkeit, nur relevante Daten herauszufiltern, erhält man gerade auch für große ETL-Umgebungen anwendbare Reports. Diese individuellen Reports lassen sich problemlos automatisieren und ins Scheduling einbinden.

Ute Middendorf
ute.middendorf@metafinanz.de

Unternehmen stützen sich seit Jahrzehnten bei ihren Geschäftsentscheidungen auf Transaktionsdaten, die in relationalen Datenbanken gespeichert sind. Neben diesen kritischen Daten gibt es aber noch eine Vielzahl weiterer Quellen mit zum Teil weniger streng strukturierten Daten wie Office-Dokumenten, E-Mails, Beiträgen aus Internet-Foren, Blogs, sozialen Netzwerken oder Sensordaten. Durch Anbindung dieser meist brachliegenden Datenquellen lassen sich nützliche Zusatz-Informationen gewinnen und für die ganzheitliche Darstellung geschäftlicher Zusammenhänge einsetzen.

Aufbau agiler BI- und Discovery-Applikationen mit Oracle Endeca

Harald Erb, ORACLE Deutschland B.V. & Co. KG

Für den Aufbau agiler BI- und Discovery-Applikationen stellt der Artikel das neue Produkt Oracle Endeca Information Discovery (OEID) vor, setzt es anhand eines durchgängigen Beispiels in den Gesamtkontext von Oracles Business-Analytics-Strategie beziehungsweise des zugehörigen Lösungsangebots und erläutert, wie OEID auf neuartige Weise Funktionen einer Suchmaschine mit der Leistungsfähigkeit eines Business-Intelligence-Werkzeugs kombiniert.

Social Media Monitoring – ein Laborbeispiel

Zu den häufig diskutierten Anwendungsbereichen von Discovery Applikationen gehört das systematische Beobachten und Analysieren von Social-Media-Beiträgen und Dialogen in Diskussionsforen, Weblogs, Communities etc., um unter anderem die Zustimmung oder Ablehnung der Konsumenten zu Produkten und Services besser verstehen zu lernen. Nachfolgend wird auf Basis der Oracle-Business-Analytics-Plattform ein vereinfachtes Szenario beschrieben, das in vier Schritten („Aquire“, „Organize“, „Analyze“ und „Decide“) den Aufbau einer Analyse-Applikation skizziert. Die Oracle-Business-Analytics-Plattform bietet für die Umsetzung solcher Vorhaben die passende Infrastruktur und besteht in unserem Szenario aus folgenden sogenannten „Engineered Systems“ (siehe Abbildung 1):

- *Big Data Appliance*
  Zur Bereitstellung der zu verarbeitenden Massendaten (auch in unbeziehungsweise semistrukturierter Form)
- *Exadata*
  Für die kombinierte Analyse von Big Data mit den traditionellen Unternehmens-Datenquellen wie Data Warehouse, OLTP-Datenbank.
- *Exalytics*
  Für den Aufbau/Betrieb analytischer In-Memory-Applikationen

In unserem Laborbeispiel gehen wir von bereits vorhandenen Unternehmens-Datenquellen (Data Warehouse, PLM-System etc.) aus, Twitter-Kurznachrichten sollen als zusätzlich anzubindende Datenquelle die Basis für Social-Media-Analysen bilden, die später mit OEID erfolgen.

Aquire

In unserem Beispiel müssen zunächst Twitter-Daten – eingeschränkt nach eigenen Suchbegriffen – beschafft und

Abbildung 1: Die Oracle-Business-Analytics-Plattform

Performance by Design.

areto kennt die Stellschrauben.

ORACLE Gold Partner

Wer voraus denkt, ist schneller am Ziel: Mit sorgfältiger Datenmodellierung, versiertem Technologie-Einsatz und nachhaltigen Projektstandards bringen wir Struktur in Ihre Datenbestände. So entstehen genau auf Ihre Anforderungen zugeschnittene BI- und Reporting-Anwendungen mit hoher Akzeptanz und Leistungsfähigkeit.

Entdecken Sie jetzt eine neue Dimension in Business Intelligence und Reporting.

Rufen Sie uns an:
0221 66 95 75-75

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

areto
CONSULTING. IT WORKS.

zur Oracle Big Data Appliance (BDA) übertragen werden. Twitter stellt dazu verschiedene Web-Service-APIs bereit (siehe http://dev.twitter.com), die es uns entweder erlauben, per Bulk-Collect (REST-API) historische Tweets über einen Zeitraum von sieben bis zehn Tagen zu erhalten oder durch Nutzung der Streaming-APIs kontinuierlich Tweets zu vorgegebenen Suchbegriffen oder einzelnen Usern in die Big Data Appliance zu laden. Twitter stellt die angeforderten Daten im XML-/JSON-Dateiformat bereit, daher bietet sich für unser Szenario die dateiorientierte Speicherung der Twitter-Inhalte im Hadoop-Distributed-Filesystem (HDFS) der BDA an. Alternativ steht in der BDA für die satzorientierte Speicherung der Social-Media-Daten die Oracle NoSQL Datenbank als Universal-Key-Value-Speicher zur Verfügung, die technisch auf der Oracle Berkeley Datenbank basiert.

Abbildung 2 zeigt, wie die Beschaffung der Twitter-Daten zum Beispiel mit dem Java-Programm „twitter_search.jar" programmatisch umgesetzt werden kann. Möchte man nahezu in Echtzeit User-Statusmeldungen oder die Ergebnisse eigener Suchanfragen aus dem globalen Twitter-Stream abrufen, dann startet man mit dem Aufruf „Stream" einen Twitter-Streaming-Job, der die resultierenden Tweets in ein XML-Dateiformat wandelt und anschließend für die Weiterverarbeitung in einem Eingangsverzeichnis im verteilten Dateisystem (HDFS) der BDA speichert.

Über diesen Weg erhält man zusammen mit der Twitter-Nachricht den Namen des Users, den Zeitstempel der Nachricht sowie einige User-Metriken wie „Anzahl Follower" und „Anzahl Freunde". Alternativ lassen sich per Aufruf „Search" ältere Twitter-Daten mit eigenen Schlagworten durchsuchen. Im Ergebnis erhält man zusätzlich zu den gefundenen Tweets nur die Zeitangabe und den User-Namen des Verfassers. Die User-Metriken (social importance metrics), die wichtig für die Bestimmung des Einflusses des Users in der Netzwelt sind, fehlen allerdings. Über einen User Lookup lässt sich dieses Informationsdefizit jedoch beheben, indem nachträglich die Ergebnisse der Twitter-Suche um diese Metriken ergänzt werden.

Für unser Beispielszenario werden rund um das Thema Mobilfunk die wichtigsten Suchbegriffe in einer einfachen Konfigurationsdatei hinterlegt und kategorisiert. So bilden die Begriffe „@iPhone", „@iPhone3", „@iPhone4" eine Kategorie „IP". Die vom Twitter-Streaming-Job abgerufenen Ergebnisse werden dann bei der Umformatierung in ein XML-Ausgabeformat noch zusätzlich um eine entsprechende Kategorie-Information erweitert. Dieser Verarbeitungsschritt ermöglicht später das Organisieren der Daten mit dem Hadoop-Framework „MapReduce".

Organize

Nach der gezielten Beschaffung der Twitter-Daten findet in unserem Beispiel nun das Framework „MapReduce" seine Anwendung. Ziel ist dabei die Durchführung einer Sentiment-Analyse, um aus den Twitter-Nachrichten systematisch die Konsumenten-Zustimmung beziehungsweise -Ablehnung zu Produkten oder Services ermitteln zu können (siehe Abbildung 3). Die Sentiment-Analyse selbst ist simpel gehalten: Der Text eines Tweets wird tokenisiert und ein Fuzzy-Match-Algorithmus durchsucht dafür hinterlegte Wörterbücher nach passenden positiven oder negativen Wörtern; die Grammatik der Sätze bleibt unberücksichtigt. Wird ein positives Wort gefunden, erhöht sich der Sentiment Score, bei negativen Treffern reduziert er sich entsprechend. Für die Anwendung anspruchsvollerer Methoden bietet BDA zusätzliche In-Database-Funktionen wie Text Mining, Data Mining oder die Statistikumge-

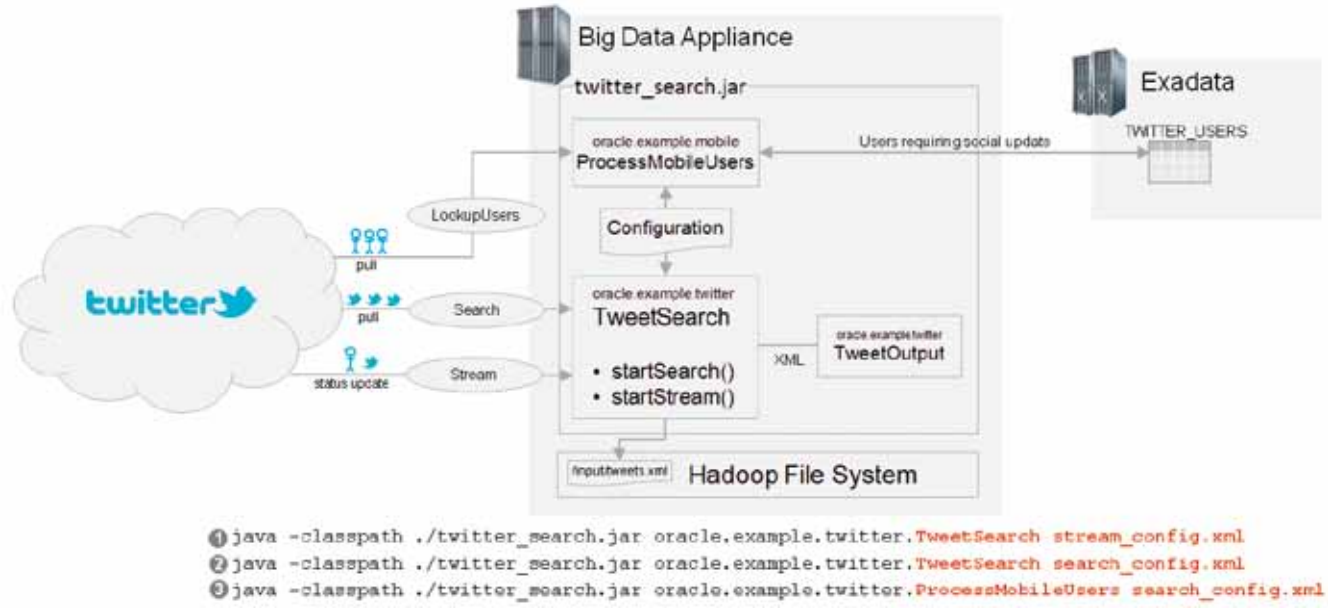


Abbildung 2: Akquisition von Twitter-Daten

Tweet	Sentiment
Would like to take the opportunity to thank @Vodafone customer service for delivering fast at no cost! #yay #awesome.	5
I hate the display. It is broken. Not Good!.	-3

Abbildung 3: Sentiment Score der Tweet-Nachrichten berechnen

bung „Project R“ und unterstützt auch Semantik-Analysen.

Die MapReduce-Jobs fassen in unserem Beispiel in der Mapper-Phase die Tweets und ihre errechneten Sentiment-Scores nach bestimmten Kriterien zusammen, zum Beispiel auf Tagesbasis, nach Telekommunikations-Providern (VF = Vodafone), nach den im Aquire-Schritt festgelegten Tweet-Kategorien (Staff, Contracts, iPhone etc.) oder nach Twitter-Usern. Als Ergebnis produziert der Mapper Key-Value-Paare, die im Anschluss die „Shuffle & Sort“-Phase durchlaufen, bevor sie an den Reducer übergeben werden. In der Abbildung 4 sind die beteiligten Komponenten zu sehen, repräsentiert durch das Java-Programm „mobile_mr.jar“, für unser Beispiel sowie für die in der Mapper-Phase zu erledigenden Aufgaben.

In der Reducer-Phase erfolgt schließlich die Weiterverarbeitung der sortierten Schlüssel und Werte-Arrays. In unserem Beispiel entstehen dabei neue Key-Value-Paare mit nun zwei Metriken: dem aufsummierten Sentiment-Score und der Anzahl der Vorkommnisse pro Schlüssel (siehe Abbildung 5). Der letzte Schritt ist wiederum die Ausgabe der vom Reducer generierten, finalen Key-Value-Paare, die nun als Textdateien im HDFS abgelegt werden.

In unserem Beispiel soll später einmal der Marketing-Bereich eines Telco-Anbieters mit Endeca Information Discovery die Tonalität der Twitter-Posts seiner Konsumenten für eine gezielte Kunden-Ansprache nutzen können. Dafür werden die Ergebnisse der MapReduce-Jobs mit dem Oracle Direct Connector for HDFS (ODCH) via InfiniBand-Netzwerkverbindung in eine Oracle-Datenbank geladen (siehe Abbildung 6).

Die im verteilten Dateisystem der BDA abgelegten Key-Value-Paare können somit in einem Exadata Data Warehouse über eine externe Tabelle bequem mit SQL abgefragt und weiterverarbeitet werden. Für unser Beispiel ist es ein wichtiger Punkt, wenn es gelingt, die neuen Erkenntnisse aus dem Twitter-Kanal mit den Kundendaten etwa im Data Warehouse verknüpfen zu können. Ab jetzt können bereits Unternehmensdaten zusammen mit den aufbereiteten Social-Media-Daten per In-Database-Analytik von einer großen Anzahl von Anwendern genutzt werden.

Analyze

In unserem Beispiel sind nun die Sentiments zu allen Tweets errechnet, die

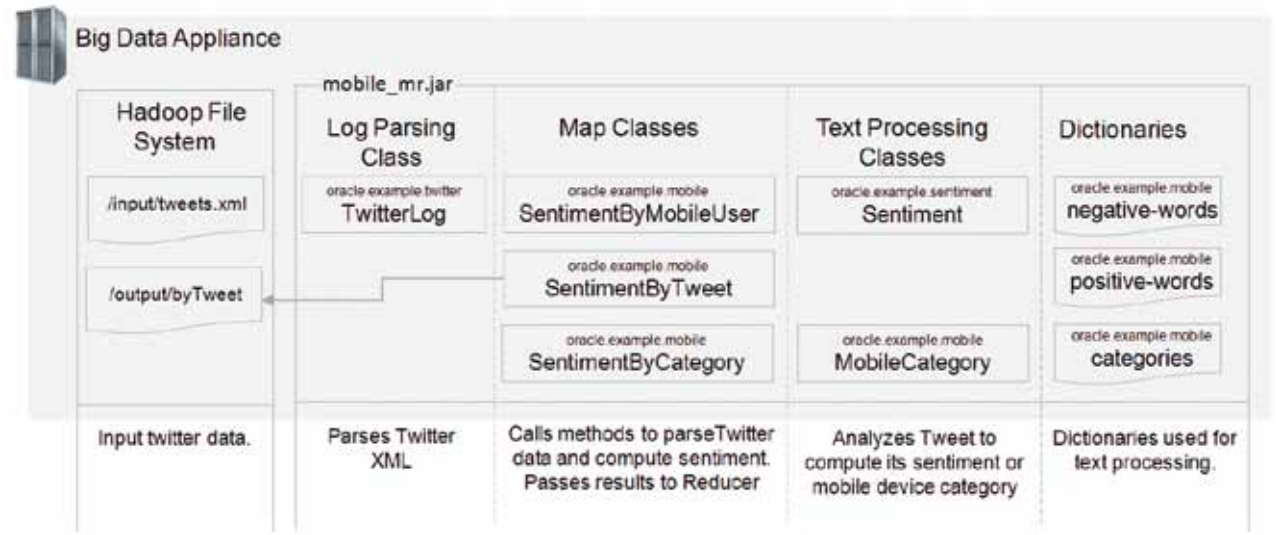


Abbildung 4: Die Mapper-Phase

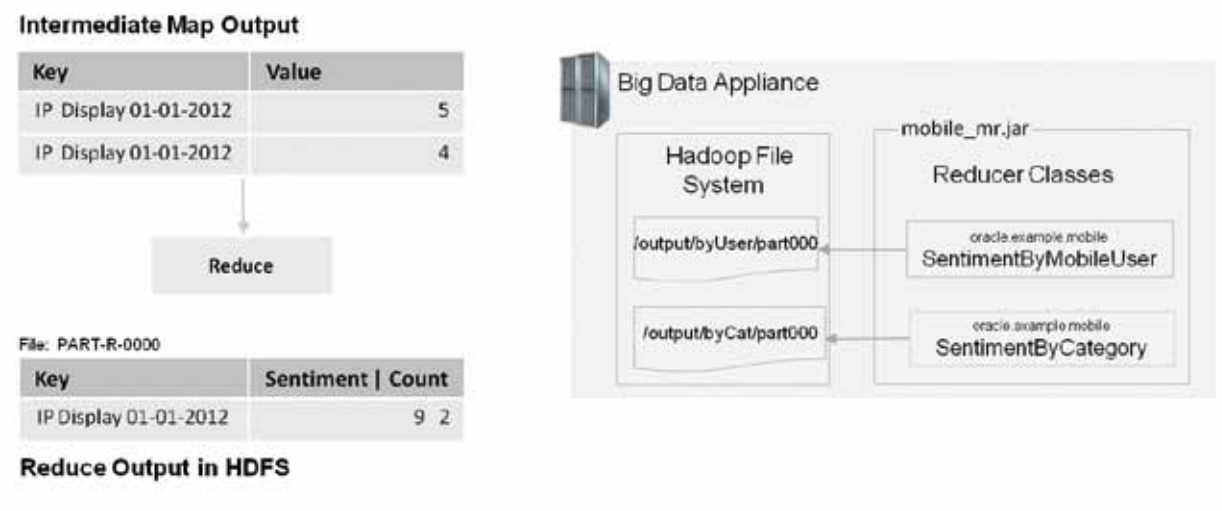


Abbildung 5: Die Reduce-Phase

wir zu diesem Zweck vom globalen Twitter-Stream in die Big Data Appliance übertragen und anschließend nach verschiedenen Kategorien („auf Tagesbasis", „nach Hersteller", „nach Mobilfunkkunden", „nach Service-Aspekten") aggregiert haben. Ferner kennen wir aufgrund einiger Metriken („Anzahl Freunde", „Anzahl Follower") die „Social Importance" der Verfasser zu den untersuchten Tweets.

Durch das Laden dieser Twitter-Daten in das Enterprise Data Warehouse (Exadata) und die Verknüpfung mit den vorhandenen Unternehmensdaten lässt sich nun beispielsweise die Kundenzufriedenheit über die Zeit darstellen. Es können aber auch Trends ermittelt und die Zufriedenheit der eigenen Kunden mit denen des Wettbewerbs verglichen werden. Mit den bekannten analytischen Fähigkeiten der Oracle-Datenbank lassen sich so folgende Fragen beantworten: „Welche ökonomisch wichtigen Kunden sind aufgrund bestimmter Missstände zunehmend frustriert?", „Bei welchen Kundenverträgen ist mit höherer Wahrscheinlichkeit mit einer Kündigung zu rechnen?" oder „Welche Kunden sind – im Social-Media-Kontext betrachtet – Meinungsmacher und nehmen Einfluss auf mein Geschäft?"

Zur Umsetzung der meisten Fragestellungen würde man den klassischen Data-Warehouse- und Business-Intelligence-Ansatz wählen und für die Fachanwender – vom zentralen Enterprise Data Warehouse (Datenmodell) ausgehend – themenspezifische Data Marts (relational/multidimensional) ableiten, die für einen definierten Zeitbereich verdichtete Informationen enthalten. Auf diesen Auswerte-optimierten Datenquellen lässt sich dann In-Memory mit der Exalytics BI Machine auf Basis der Business Intelligence Foundation Suite eine unternehmensweit einsetzbare BI-Plattform aufbauen, die es den Fachanwendern erlaubt, per Self-Service-BI alle relevanten Geschäfts-Informationen abzurufen oder grafisch unterstützt eigene Analysen durchzuführen. Hier stellen sich nun die Fragen: „Wozu also jetzt Endeca einsetzen?" und „Wozu dieser Artikel über agile BI- und Discovery-Applikationen?"

Aus Oracle-Sicht kann man zwischen zwei Arten von Fragestellungen unterscheiden, mit denen analytische Applikationen umgehen müssen. Zum einen gibt es den Typ von Business-Fragestellungen, bei dem im Voraus die entsprechenden Geschäftsprozesse und die dazu benötigten Daten durch die Fachseite bekannt sind: „Wie stellt sich die Umsatzprognose nach Region für einen bestimmten Zeitraum dar?" oder „Wie ist die Performance meiner Organisation im Vergleich zu den gesetzten Zielen?" Zum anderen gibt es Fragestellungen, bei denen weder der entsprechende Geschäftsprozess noch die benötigten Daten vorab durch die Fachseite definiert werden können: „Auf welche Kunden sollen wir uns fokussieren?" oder „Warum gehen meine Verkaufszahlen zurück?" Interessant ist dabei zu sehen, dass der zweite Fragentyp aufgrund seines offenen Charakters im Vergleich zum ersten Typ viel kurzlebiger ist und eher neue Fragen hervorbringt, als abschließend beantwortet zu werden.

Das Interaktionsmodell für die bekannten Fragestellungen kann man ganz gut mit dem Betrachten von aufbereiteten Informationen in einem Standardbericht oder einem BI-Dashboard beschreiben – so wie es heute mit traditionellen Business-Intelligence-Mitteln umgesetzt wird. Bei den heute noch unbekannten, aber morgen schon von den Fachanwendern nachgefragten Analysen ist dagegen ein Interaktionsmodell erforderlich, das eher die Datenerkundung beziehungsweise das Entdecken neuer Zusammenhänge (Data Discovery) unterstützt. Betrachtet man dort zusätzlich den Aspekt der Datenmodellierung, so finden wir bei Business-Intelligence-Lösungen in der Regel den allumfassenden semantischen Layer vor, dessen Aufbau und Pflege Zeit und Geld kostet.

Investitionen dieser Art werden von Unternehmen nur getätigt, wenn sich die Anstrengungen durch Effizienzgewinne bei der Informationsbeschaffung wieder amortisieren. Gleichzeitig sinken weiterhin die Kosten für Speichermedien und mit der Popularität von Hadoop steigen die Aussichten,

dass aus nichtmodellierten Daten Nutzen gezogen werden kann.

Aus diesen beiden Blickwinkeln erkennt man, dass sich traditionelle Business-Intelligence- und Data-Discovery-Lösungen ergänzen können. Die nach den Anforderungen der Fachseite aufgebaute Business-Intelligence-Anwendung liefert qualitätsgesicherte Ergebnisse für bekannte Fragestellungen. Es können aber auch neue Fragen aufgeworfen werden, deren Beantwortung erst mit einem neuen Release der BI-Anwendung oder gar des Data Warehouse möglich ist – im schlimmsten Fall ist die Anforderung bis dahin schon obsolet geworden. Mit Endeca sind dagegen neue fachliche Fragestellungen schneller zu beantworten, insbesondere dann, wenn die dafür notwendigen Informationen in den unterschiedlichsten Formaten vorliegen (strukturiert, semistrukturiert, unstrukturiert) und in den verschiedensten Systemen gespeichert sind (DWH, operative Datenbanken, Office-Dokumente). Stellt sich bei der Arbeit mit Endeca Information Discovery heraus, dass die beantworteten Fragestellungen regelmäßig benötigt werden, kann dies in die Release-Planung für die nächste DWH-Version beziehungsweise die Version der Business-Intelligence-Applikation einfließen.

Zurück zum Beispiel: Abbildung 7 zeigt den typischen Endeca-Fall. Ein Großteil der zu analysierenden Daten (einschließlich der aufbereiteten Tweets) stammen aus dem Data Warehouse (Exadata), weitere Zusatz-Informationen liefern in strukturierter Form die Datenbank eines Product-Lifecycle-Management-Systems (detaillierte Gerätebeschreibungen) und in semistrukturierter Form (Vertragsdokumente) ein Content-Management- oder CRM-System. In einer Integrationsphase werden Daten aus unterschiedlichen Quellen miteinander verknüpft und in der Exalytics-Umgebung als denormalisierter „Endeca Record“ im Endeca-Server, einer spaltenorientierten In-Memory-Datenbank, gespeichert.

Die facettierte Datenhaltung im Endeca-Server kommt ohne Tabellen und vordefiniertes Datenmodell (Schema)

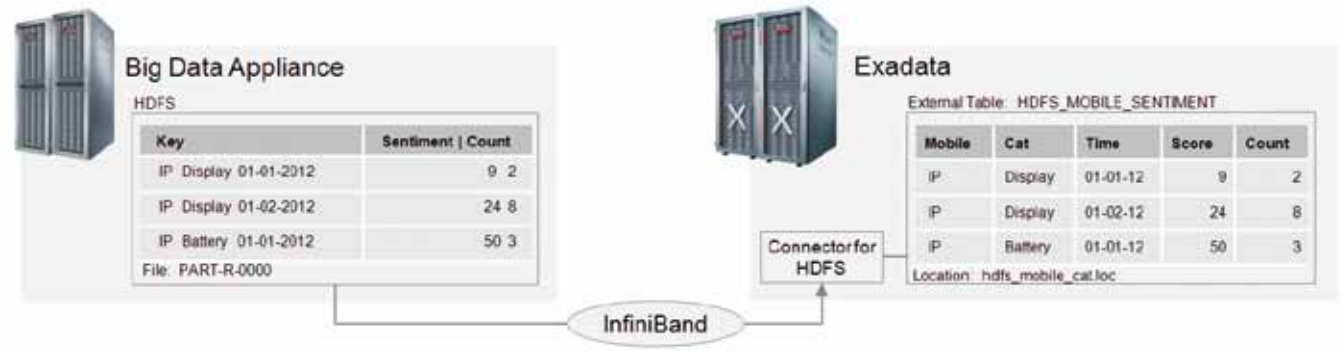


Abbildung 6: Verwendung der MapReduce-Ergebnisse in der Datenbank

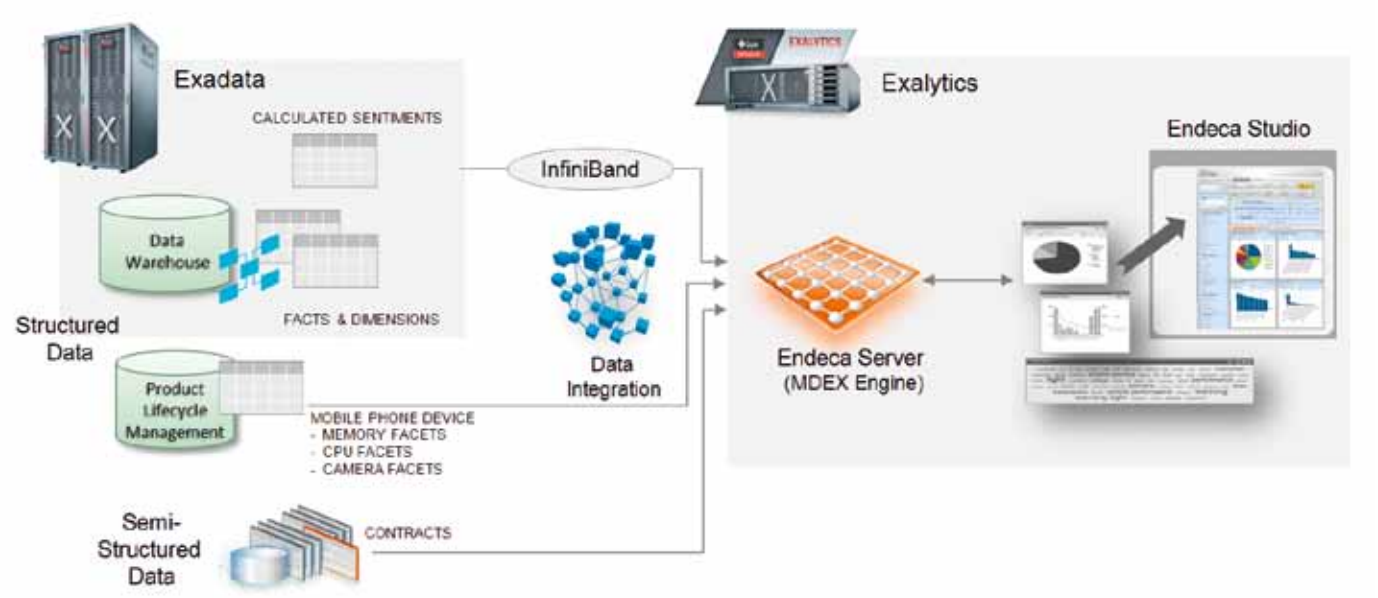


Abbildung 7: „Model as you go“-Ansatz mit Endeca Information Discovery

aus. Die Datensätze selbst werden als Sammlung von Key-Value-Paaren gespeichert, dabei kann jeder Datensatz anders aufgebaut sein – das Facetten-Datenmodell leitet sich automatisch aus den geladenen Daten ab. Das bedeutet zum Beispiel, dass es Daten-Attribute gibt, die exklusiv nur im Data Warehouse, im Product Lifecycle Management System (PLM) oder in den (Meta-)Daten der geladenen Dokumente vorkommen. Andere (globale) Attribute finden sich in mehreren oder allen Datenquellen wieder. Ferner lässt der Endeca-Server auch die Speicherung von semi-strukturierten Daten und Multi-Value-Feldern zu.

Analytische Anwendungen können auf diese Weise schnell implementiert und iterativ weiterentwickelt werden. So kann sich die Erweiterung der Produkt-Dimension in einem relationalen Data-Warehouse-Schema beim ständigen Hinzufügen neuer Produkt-Facetten auf die Dauer als komplex erweisen. Der Endeca Data Store lässt sich dagegen aufgrund seiner flachen, XML-ähnlichen Struktur, mit sich selbst beschreibenden Key-Value-Paaren, beliebig erweitern. Zum Laden verschiedener Datenquellen kommt die Endeca Integration Suite zum Einsatz, die aus dem Werkzeug „CloverETL“ mit Konnektoren und den Content-Enrichment-Bibliotheken für die Zusammenführung und Anreicherung vielfältiger Informationen besteht.

Die Endeca Integration Suite ermöglicht die effiziente Vernetzung strukturierter und unstrukturierter Daten zu einer einheitlichen, integrierten Sicht. Die Kommunikation mit dem Endeca-Server erfolgt über Web-Services, für große Datenmengen gibt es ein Bulk-Loader-Interface. Während des Betriebs können neue Daten zum Endeca Data Store hinzugefügt oder vorhandene Datensätze aktualisiert werden, ohne dass eine Neu-Indexierung aller Daten erforderlich ist.

Für unser Beispiel ist das zur Integration Suite zugehörige „Content Acquisition System“ (CAS) interessant. Dabei handelt es sich um eine Crawling-Umgebung, die verschiedene Konnektoren zur Integration unstrukturierter Daten bietet – in unserem Fall zum Erfassen der Vertragsdokumente im MS-Office- oder PDF-Format. Zum weiteren Leistungsumfang zählt auch ein Webcrawler zur Anbindung von Internet-Sites.

Die Endeca Integration Suite ermöglicht optional auch die Einbindung von Text-Analyse- und Text-Mining-Produkten von Drittanbietern. Auf diesem Weg lassen sich wichtige Begriffe (wie Personen-, Orts- und Firmen-Namen) aus textbasierten Informationsquellen extrahieren oder Sentiment-Analysen durchführen, um die positive/negative Tonalität eines Forenbeitrags oder die Produktzustimmung / -ablehnung von Konsumenten erkennen zu kön nen.

Für das Beispiel wäre dies also eine Alternative zu unserer selbstprogrammierten Sentiment-Analyse in der Big Data Appliance.

Decide

Wie schon gesagt, kann auf der Exalytics BI Machine neben der BI Foundation Suite auch die gesamte Endeca-Infrastruktur betrieben werden. Dazu gehört als Middleware-Komponente „Oracle Endeca Studio“, eine webbasierte Infrastruktur, auf die Anwender per Browser zugreifen können. Endeca Studio stellt eine Bibliothek mit vorgefertigten Portlets zur Verfügung, die per „Drag & Drop“ auf die Anwenderoberfläche gezogen und dort konfiguriert werden können. Im Ergebnis steht den Endbenutzern eine agile Discovery-Anwendung zur Verfügung, in der jedes Attribut, das in dem Endeca-Datenbestand enthalten ist, als Filter-Kriterium dienen kann. Alle Charts und Filtermöglichkeiten berechnen sich direkt nach jedem weiteren gesetzen Abfragefilter neu, per „Faceted Navigation“ sieht man als Analyst stets die aktuell verfügbaren Navigationsoptionen. So werden Resultate immer neu zusammengefasst präsentiert, die Nutzer bekommen durch die integrierte Volltextsuche in den semi- beziehungsweise unstruktierten Daten neue Anhaltspunkte, wie sie die Ergebnisse weiter verfeinern und neue Zusammenhänge in den Daten erkennen können.

Harald Erb
harald.erb@oracle.com

Weiterführende Informationen

[1] Jean-Pierre Dijcks: Oracle: Big Data for the Enterprise, Oracle White Paper, Januar 2012, http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/1453236.pdf

[2] V. Murthy, M. Goel, A. Lee, D. Granholm, S. Cheung: Oracle Exalytics In-Memory Machine: A brief introduction, An Oracle White Paper, Oktober 2011, http://www.oracle.com/us/solutions/ent-performance-bi/business-intelligence/exalytics-bi-machine/overview/exalytics-introduction-1372418.pdf

[3] o.V.: A Technical Overview of Oracle Endeca Information Discovery, Oracle White Paper, Mai 2012, http://www.oracle.com/us/solutions/ent-performance-bi/oeid-tech-overview-1674380.pdf

[4] M. Klein: Informationen mit Oracle Endeca Information Discovery entdecken, DOAG News 4-2012

[5] C. Czarski: Big Data: Eine Einführung, Oracle Dojo Nr. 2, München 2012,http://www.oracle.com/webfolder/technetwork/de/community/dojo/index.html

DOAG 2013 Development 19. Juni 2013, Bonn

Eine Konferenz für den Erfahrungsaustausch von Software-Entwicklern

Themen:
- DB Programmierung: PL/SQL, APEX, Spatial
- BPM & Software-Architektur
- Java & Open Source
- Forms, Reports, ADF und BI Publisher

FRÜHBUCHER BIS 22. MAI 2013

Aussteller: ORACLE t&p trivadis

Im Fokus:

Agile and Beyond – Projektmanagement in der Oracle-Software-Entwicklung
Wohin geht die Reise? (Part Two)

DOAG
Deutsche ORACLE-Anwendergruppe e.V.

http://development.doag.org

Agile Methoden kommen bei der Neu- und Weiterentwicklung von BI-Projekten immer häufiger zur Anwendung. Ein wesentlicher Faktor für den Erfolg dieser Projekte ist das Testen, insbesondere im Backend-Bereich. Die Notwendigkeit für tägliche, automatisierte Testläufe ergibt sich aus den kurzen Release-Zyklen, wie sie in der agilen Entwicklung üblich sind. Zudem macht es die ständig wachsende Zahl an Regressionstests ab einem gewissen Punkt unmöglich, die Tests in einer annehmbaren Zeit manuell durchzuführen. Dieser Artikel zeigt, welche Voraussetzungen für automatisiertes Testen in BI-Projekten geschaffen werden müssen und welche Werkzeuge sich in der Praxis bewährt haben.

Agile BI in der Praxis – agiles Testen

Andreas Ballenthin und Thomas Flecken, OPITZ CONSULTING GmbH

Die Einführung einer erfolgreichen Testautomatisierung bedarf einiger Grundlagen. Zunächst werfen wir einen Blick auf die Team-Zusammensetzung. Das agile Vorgehensmodell, nach dem wir entwickeln, sieht vor, dass die Teammitglieder sämtliche Bereiche des Wertschöpfungs-Prozesses abdecken. Das umfasst sowohl die Konzeption, die Entwicklung im Backend und im Frontend als auch die Unit-, Verbund- und Regressions-Tests. Entwickler und Tester sind als Rollen zu verstehen, die nicht dedizierten Personen zugewiesen sind, sondern von Story zu Story oder sogar von Task zu Task wechseln können. Dem Team sollte der Mehrwert der Test-Automatisierung stets bewusst sein. In einem ersten Schritt ist es hilfreich, die Implementierung von automatisierbaren Testfällen als Element der „Definition of Done" aufzunehmen.

Eine weitere, wichtige Voraussetzung ist die Verantwortung des Scrum-Teams für das Testsystem. Es muss jederzeit in der Lage sein, Abläufe und Inhalte nach den Bedürfnissen des anstehenden Testlaufs anzupassen, ohne dabei auf eine weitere Partei zugreifen zu müssen. Das Team benötigt folglich die DBA-Rolle auf den Entwicklungs- und Test-Datenbanken, Administratoren-Rechte für das eingesetzte ETL-Tool (Informatica PowerCenter) und Rechte auf den User, unter dem das ETL-Tool installiert wurde.

Die Praxis zeigt, dass testgetriebene Entwicklung (TDD) maßgeblich zum Erfolg der Story-Implementierungen beiträgt. Im Wesentlichen geht man bei dieser Methode wie folgt vor:

1. Ein Testfall wird erstellt, bevor der Programmcode implementiert ist
2. Der Testfall wird ausgeführt und schlägt wie erwartet fehl
3. Der Programmcode wird implementiert
4. Der Testfall wird erneut ausgeführt und zeigt, ob die Implementierung des Programmcodes erfolgreich war

In diesem Kontext enthält ein Testfall nicht nur die möglicherweise auftretenden Fehlersituationen, sondern auch die Features des zu erstellenden Codes.

Die Vorteile dieser Vorgehensweise sind vielfältig. So erhalten wir auf diese Art und Weise ein genaues Bild von dem zu erwartenden Ergebnis, an dem wir uns bei der eigentlichen Programm-Implementierung orientieren können. Direkt im Anschluss können wir die Funktionalität testen und gegebenenfalls den Programmcode oder den entsprechenden Testfall anpassen. Zudem stellen wir sicher, dass der Test mit der Entwicklung Schritt halten kann. Durch das Persistieren der Testfälle in Subversion gehen sie nach dem Entwicklertest nicht verloren, sondern werden sofort für den Regressionstest verfügbar gemacht und eingesetzt.

Diese Arbeitsweise ist in der Regel für die Team-Mitglieder ungewohnt und erfordert anfangs eine gewisse Disziplin, bis sie selbstverständlich geworden ist. Ein einmal erstellter Test-

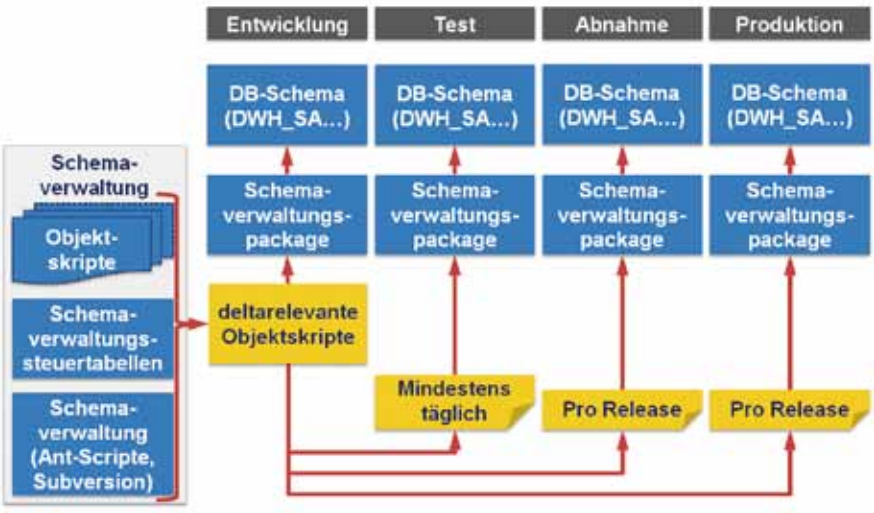


Abbildung 1: Automatisiertes Deployment – DDL, DCL, Daten-Migrationen

fall darf nicht als einzige Wahrheit gesehen werden, da bei der Implementierung des Programmcodes unerwartete Erkenntnisse über die Daten beziehungsweise deren Qualität erlangt werden können, die ein grundsätzliches Umdenken zur Folge haben.

Datenbank-Deployments per OC-Schemaverwaltung

Bevor wir uns Gedanken über die technische Umsetzung der Test-Automatisierung machen können, müssen wir zunächst sicherstellen, dass auch die Deployments der Datenbank- und ETL-Inhalte automatisch durchgeführt werden können.

Um die Datenbank-Inhalte auf einen von ihnen gewünschten Stand zu bringen, haben sich die Autoren für die Verwendung der OC-Schemaverwaltung (OCSV) entschieden. Dieses auf „Ant" basierende Werkzeug vergleicht den Ist-Zustand einer Oracle-Datenbank mit dem in Skripten definierten Soll-Zustand. Sofern Unterschiede festgestellt werden, führt die Schemaverwaltung die erforderlichen DDL-, DML- und DCL-Befehle aus (siehe Abbildung 1).

ETL-Deployments per Informatica-Kommandozeilen-Tools

In ihrem Projekt verwenden die Autoren Informatica für die Entwicklung und Steuerung der ETL-Prozesse. Eine Automatisierung des ETL-Deployments ist nicht nur unabdingbar für eine funktionierende Test-Automatisierung, sie sorgt auch für eine spürbare Reduzierung der Fehler, die beim manuellen Deployment auftreten können. Darüber hinaus ist die Performance der Kommandozeilen-Tools ungleich höher als gleichartige Operationen unter Verwendung der Client-Tools, etwa des Repository-Managers.

Der Ausgangspunkt ist wie bei der Datenbank-Entwicklung der zentrale Testserver. Hier erstellen wir ein Unix-Shell-Skript, das den gesamten Ablauf des automatisierten ETL-Deployments kontrolliert. Der Inhalt eines jeden Deployment-Pakets wird durch je eine Repository-Query im Quell-Repository festgelegt. Dort bestimmen wir die Workflows, die ausgeliefert werden sollen und konfigurieren die Query so, dass auch alle von den Workflows abhängigen Objekte selektiert werden. Das Ergebnis der Query dient als Basis für den vom Shell-Skript ausgelösten XML-Export.

Das dadurch erzeugte Deployment-Paket wird einschließlich Kontrolldatei auf den Server mit dem Ziel-Repository kopiert. Ein weiteres, parametrierbares Shell-Skript auf dem Zielserver stößt schließlich den Workflow an, der die XML-Datei importiert. Die Protokollierung des ETL-Deployments erfolgt in einer Log-Datei, die am Ende des Prozesses auf dem Testserver abgelegt und in Subversion eingecheckt wird.

Frequenz der Deployments

Für die Test-Automatisierung haben die Autoren festgelegt, dass mindestens täglich Datenbank- sowie ETL-Deployments durchgeführt werden. So ist sichergestellt, dass beide Programm-Komponenten zueinander passen. Auch untertägige Deployments sind bei Bedarf problemlos durchführbar. Programmteile, die in einem noch nicht auslieferbaren Zustand sind, werden vom Entwickler nicht in Subversion eingecheckt und sind damit auch vom Deployment ausgenommen.

Baseline-Dumps

Die zentrale Komponente der automatisierten Tests ist der schon mehrfach angesprochene Testserver. Zu Beginn der Test-Automatisierung war dies eine virtuelle Maschine mit Windows-Betriebssystem, die Anfang des Jahres durch eine virtuelle Maschine mit Linux als Betriebssystem abgelöst wurde. Diese virtuelle Maschine wird nun auch von anderen Scrum-Teams des Kunden als Testserver genutzt.

Der Testserver benötigt über Secure-Shell (beziehungsweise WinSCP als skriptbarer Secure-Shell-Ersatz unter Windows) Zugriff auf alle notwendigen Server ohne Passwort-Eingabe. Im Kundenfall sind dies zwei Informatica-Server, ein Subversion-Server sowie der Data-Warehouse-Datenbankserver mit der DWH-Test-Instanz. Zusätzlich benötigt der Testserver SQLNet-Zugriff auf die DWH-Instanz und zur Informatica-Instanz für die Verarbeitung von SQL-Skripten. Im Optimalfall steht ein kompletter Oracle-Client zur Verfügung; existiert nur ein Oracle-Instant-Client, so können beispielsweise die Oracle-Clients der DWH-Instanz verwendet werden.

Eine wesentliche Grundfunktionalität ist das Aufsetzen oder Wiederaufsetzen auf dem letzten produktiv gesetzten Software- und Daten-Stand. Dazu exportieren wir am Tag einer Auslieferung auf das Abnahmesystem den erfolgreich getesteten Datenstand des Testsystems. So besteht in den Tests des nächsten Sprints immer wieder die Möglichkeit, diesen Datenstand zurückzusichern. Den Export dieses Da-

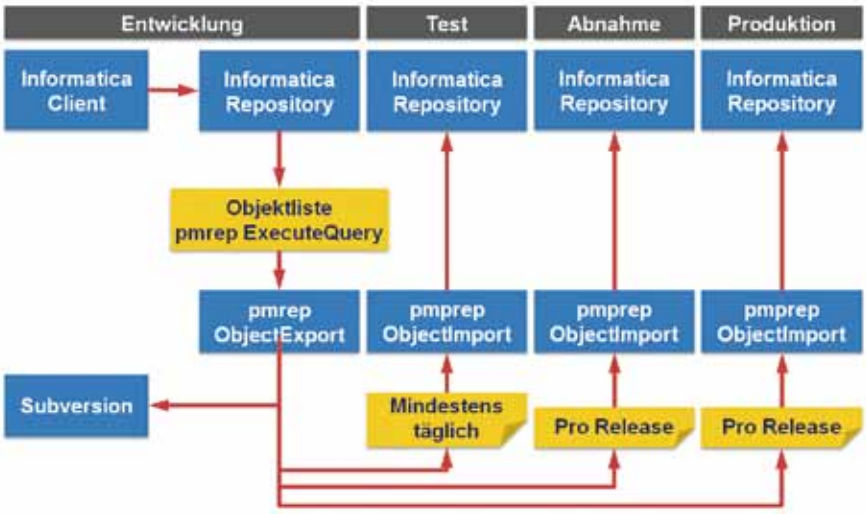


Abbildung 2: Automatisiertes Deployment mit Informatica

tenstands bezeichnen die Autoren als „Baseline-Dumps“. In der Regel werden Datenbank-Schemata um nicht notwendige Daten bereinigt; so enthalten unsere Staging-Tabellen beim Export nur noch genau eine Zeile. Es muss unbedingt darauf geachtet werden, diese Baseline-Dumps auf einem separaten Storage zu sichern, weil sie eine sogenannte „Golden Source“ darstellen.

Statt jeden abhängigen Server an Subversion anzubinden, haben die Autoren entschieden, jedes auf einem abhängigen Server auszuführende Skript vor jeder Ausführung vom Testserver zum abhängigen Server zu transferieren und auszuführen. Führender Server ist also immer der Testserver. Im Fall der Baseline-Dumps besteht diese Notwendigkeit, denn impdp steht auf dem Testserver nicht zur Verfügung. Wenn ein komplettes Release-Upgrade getestet werden soll, so ist der Daten- und DDL-Stand der Baseline-Dumps eine hinreichende Basis.

Initial- & Delta-Loads

Mit Testplänen muss es möglich sein, sowohl Initial-, Teil-Initial- (beispielsweise bei der Initialisierung einer neuen Faktentabelle) sowie Delta-Loads zu testen. Erfahrungsgemäß reicht es nicht, mit Initial- beziehungsweise Teilinitial-Loads zu arbeiten, denn zeitliche Abgrenzungsprobleme wie Nachläufer fand man regelmäßig erst nach mehreren Delta-Loads.

Testfälle

Idealerweise entstehen Testfälle bereits während der zuvor geschilderten testgetriebenen Entwicklung. Testfälle müssen gegen das Entwicklungs- und das Testsystem ausführbar sein. Implementierte Testfall-Kategorien sind beispielsweise:

- Datenabgleiche innerhalb einer Schicht des Data Warehouse (z.B. Aggregate vs. Detail-Fakten)
- Datenabgleiche zwischen den Schichten des Data Warehouse
- Datenabgleiche „End-to-End“ (Fakt oder Dimension zum Quellsystem)
- Prüfen auf Codierungs-Standards (Hier werden Prüfungen im ETL-Repository vorgenommen)
- Prüfen, ob alle Foreign Keys deployt sind und ob Tabellen- oder Spalten-Kommentare fehlen
- Hilfs-Skripte zum Warten auf das Ende von ETL-Prozessen
- Erstellen von Flat-Files, die bei ihrer Verarbeitung definierte Metrikwerte generieren
- Generierung vorsätzlich falscher Flat-Files zur Prüfung von Protokoll-Funktionalitäten
- Prüfen, ob Fehler-Tabellen leer sind

Die Testfälle bestehen in der Regel aus drei Komponenten:

- *Testfallregistrierung*
  Es muss gewährleistet sein, dass ein Testfall unter allen Umständen einen Eintrag im Test-Protokoll generiert, selbst wenn der Testfall syntaktisch falsch ist
- *Aufruf des Testfalls*
  Den Aufruf eines Testfalls codieren wir in der Regel als Shell-Skript (zum Aufruf aus einem Testplan) und als Batch-File (zum Aufruf des Entwickler-Clients, insbesondere im Rahmen der Testfall-Entwicklung)
- *Testfall-Source-Code*
  In allen bisher codierten Testfällen ist dies ein SQL-Statement oder ein anonymer PL/SQL-Block

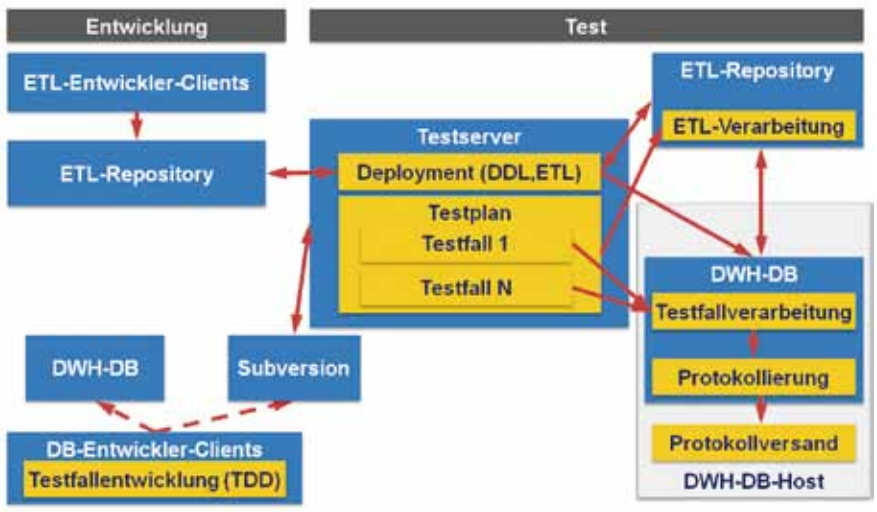


Abbildung 3: Die Architektur der Test-Suite

Diese Methodik wird nachfolgend am Beispiel des Prüfens von Spalten-Kommentaren illustriert. Die Autoren haben sich die Konvention auferlegt, dass jede Tabelle in jeder Spalte einen Spalten-Kommentar erhält, abgesehen von Stage-Tabellen und systemgenerierten Tabellen wie „SYS_EXPORT%“, die von Export Dump generiert werden. Die Testfall-Registrierung „sqlplus /nolog @register_testfall 01010_DDL $1“ sorgt nun dafür, dass der Testfall einen Testprotokoll-Eintrag erzeugt. Später in der Abarbeitung der Testfall-Gruppe wird der Testfall mit „sqlplus /nolog @01010_DDL.sql $1“ aufgerufen und es wird ein Skript ausgeführt (siehe Listing 1). Nun gibt es drei mögliche Testergebnisse:

- *Der Testfall ist erfolgreich*
  Dann wird der Status des Testfalls auf „erfolgreich“ geändert
- *Der Testfall ist nicht erfolgreich*
  Dann wird der Status des Testfalls auf „nicht erfolgreich“ geändert
- *Der Testfall wurde nicht aufgerufen oder ist syntaktisch falsch*
  Dann bleibt der Testfall mit dem Status „registriert“ im Test-Protokoll stehen.
  Dies kann bei Tabellenstruktur-Änderungen, also im Rahmen von Regressionstests, aber auch bei noch nicht fertig codierten Testfällen im Rahmen des Test Driven Developments der Fall sein.

Synthetische Testdaten anreichern und löschen

Bei einer Reihe von Testfällen, die zum Beispiel ein korrektes Verhalten in Feh-

```
declare
  v_testfall varchar2(10)  := '01010_DDL';
  v_result   varchar2(1)   := 'E';
  v_notes    varchar2(255) := NULL;
begin
  select case when count(*)=0 then 'S' else 'E' end,count(*)||' feh-
lende Kommentare'
    into v_result,v_notes
    from (
           select 1
             from all_col_comments c
            inner join all_tables t
               on c.owner=t.owner
              and c.table_name=t.table_name
            where t.owner like 'DWH%'
              and t.table_name not like 'SYS_EXPORT%'
         );
  result_testfall (v_testfall,v_result,v_notes);
  commit;
exception when others then
  result_testfall (v_testfall,v_result,v_notes);
  commit;
end;
/
```

Listing 1

ler-Situationen prüfen, sind in der Regel Anreicherungen mit synthetischen Testdaten nötig. Im Anschluss werden die Daten gelöscht. Abhängig von der Verarbeitungsdauer der ETL-Prozesskette haben die Autoren zwei Verfahren implementiert:

- Wenn die Prozesskette zeitlich eher lange dauert, werden alle nötigen synthetischen Testdaten im Übernahme-Prozess aus dem Quellsystem in die Staging Area oder als Zwischenschritt zwischen Staging Area und Core angereichert
- In allen anderen Fällen werden zunächst alle synthetischen Daten verarbeitet, geprüft und verworfen, um danach die Nutzdaten zu verarbeiten

Zum Löschen der angereicherten Daten sind drei Verfahren implementiert:

- Löschen per SQL. Dies setzt voraus, dass man eine Abgrenzung der Daten finden kann, was in Aggregationen unter Umständen nicht sinnvoll möglich ist.
- Zurücksetzen auf den vorherigen Datenstand vor der Verarbeitung über Export Dump, hinterher über Import Dump
- Ausführen von Flashback Table nach der Verarbeitung

Testpläne

Eine Testgruppe ist eine sinnvolle Zusammenstellung von Testfällen und die kleinste sinnvoll von Testplänen aufzurufende logische Einheit. Die Autoren haben beispielsweise Testgruppen für folgende Aktivitäten erstellt:

- Das Einspielen von Baseline-Dumps
- Den Initial- oder Delta-Load aus dem Vertragsmanagement-System
- Den File-basierten Load von Umsatzdaten
- Die Verarbeitung von Testfällen zum Vertragsmanagement oder von Umsatzdaten

Die Testpläne sind somit Zusammenstellungen von Testfall-Gruppen. Von einem Test-Operator wird täglich die Entscheidung des Teams darüber herbeigeführt, welche Testfall-Gruppen und somit welcher Testplan in der kommenden Nacht laufen soll. In der Regel wird täglich nachts getestet.

Auf dem Linux-Testserver werden Testpläne ganz klassisch über cron verwaltet. Unter Windows fungiert die Windows-Aufgabenplanung als Scheduling-Komponente.

Fazit

Mit den geschilderten Maßnahmen erreicht das Scrum-Team eine hohe Softwarequalität. Die vollständige Test-Automatisierung sorgt für ein frühes Erkennen von Fehlern und Datenqualitätsproblemen. Die Testabdeckung ist konstant hoch, manuelle Tests werden weitgehend vermieden. Die Investition in eine Test-Automatisierung ist sowohl aus technischen als auch aus betriebswirtschaftlichen Gesichtspunkten insbesondere für Agile BI unverzichtbar.

Andreas Ballenthin
andreas.ballenthin@
opitz-consulting.com

Thomas Flecken
thomas.flecken@
opitz-consulting.com

Wie der BI Publisher in Oracle Forms eingebunden werden kann, wurde von Oracle schon vor längerer Zeit in einem Whitepaper beschrieben [1]. Dieser Artikel zeigt eine für den Applikations-Server ressourcenschonende Alternative des Zusammenspiels zwischen Forms und dem BI Publisher auf.

Alternative Einbindung des BI Publisher in Forms

Stephan La Rocca und Christian Piasecki, TEAM GmbH

Bei einem Kunden mit einer gewachsenen Infrastruktur- und System-Landschaft entstand der Bedarf an einer neuen, unternehmensrelevanten Applikation. Aus der Historie heraus besteht die vorliegende Anwendungs-Landschaft nahezu durchgängig aus Forms-Anwendungen, die auf Basis eines Oracle-Applikations-Servers betrieben werden. Das Forms-Know-how ist stark ausgeprägt und der Kunde will die Pflege der entwickelten Anwendungen selbst fortführen. Auf Basis dieser Randbedingungen entschied man sich erneut für eine Forms-Oberfläche mit einer Oracle-Datenbank als Daten-Basis. Die Anwendung realisiert eine Adressverwaltung mit hochsensiblen Datenbeständen, aus der es den Benutzern möglich sein soll, einzelne oder mehrere Personen unkompliziert per Post oder E-Mail anzuschreiben. Zusätzlich sollen die Adressen für einen Etikettendruck zur Verfügung stehen und exportiert werden können.

Der BI Publisher kommt ins Spiel

Die genannten Reporting-Anforderungen lassen instinktiv an den BI Publisher als passgenaues Werkzeug denken. Durch seine Vorteile, verschiedene Ausgabekanäle mit unterschiedlichen Ausgabeformaten zu bedienen, platziert er sich für die Aufgabenstellung als geeignete Lösung. Insbesondere bedient er mühelos die unterschiedlichen Aufgabenstellungen, etwa einem Endbenutzern zu ermöglichen, Briefe sowohl als PDF-Ausgabe direkt auf einem Drucker ausgeben zu lassen, wenn gewünscht, Personen direkt per E-Mail anzuschreiben oder, abhängig von der Kontaktadresse der Person, beide Aufgaben gleichzeitig zu erledigen.

Da die Benutzer für diese Aufgaben nicht zwischen der Web-Oberfläche des BI Publisher und der Forms-Applikation wechseln sollen, sollten der BI Publisher in die Forms-Anwendung integriert und das Zusammenspiel zwischen Benutzern, Forms und BI Publisher umgesetzt werden. Dieses Zusammenspiel lässt sich in mehrere Aspekte unterteilen.

Berichts- und Vorlagen-Erstellung nur für Administratoren

Neue Berichte und Layout-Vorlagen sollen nur durch Administratoren erstellt werden, der normale Benutzer darf in der Anwendung nur aus den verschiedenen Berichten und Layout-Vorlagen auswählen. Da es Administratoren zumutbar ist, mit der Web-Oberfläche des BI Publisher zu arbeiten, wurden hier keine Anpassungen vorgenommen, sondern die Möglichkeiten genutzt, die die Web-Oberfläche und der Word-Template-Builder bieten. Das war der einfache Part.

Im nächsten Schritt ging es darum, den BI Publisher an Forms anzubinden und den Benutzern das Erstellen von Briefen, Berichten, E-Mails etc. zu ermöglichen.

Die Endbenutzer sollten diese über die Forms-Anwendung möglichst komfortabel erstellen können, indem sie einfach Kontakte, einen Bericht und Layout auswählen und dann aus Forms heraus nur noch das Generieren des Reports anstoßen.

Nutzung des Java-API als MBean

Als Erstes wurde die Lösung aus dem genannten Whitepaper [1] von Oracle über die Nutzung des BI Publisher in Forms eruiert. Die beschriebene Lösung für diesen Use-Case besitzt einen Nachteil: Durch die hier beschriebene Implementierung wäre jeder neu angestoßene Bericht ein neuer JVM-Prozess auf dem Applikations-Server und könnte so bei Benutzern zu Performance-Problemen führen. Um dieses zu umgehen, fiel die Entscheidung auf die Möglichkeit, alles in eine MBean zu kapseln, diese in Forms einzubinden und so den JVM-Prozess auf dem Client ablaufen zu lassen.

Oracle Forms bietet seit Einführung der Web-Variante die Möglichkeit, Java an verschiedenen Stellen einzubinden. Das sind im Einzelnen die Nutzung von Pluggable Java Components, Java Beans oder von PL/SQL-Wrappern. Der BI Publisher stellt ein Java-API zur Verfügung, das sowohl über einen PL/SQL-Wrapper in Forms-PL/SQL-Code eingebunden werden kann (so wie es in dem erwähnten Papier beschrieben wurde) oder aber auch für eine Java Bean ansprechbar ist. Dieser Java Bean wird in Forms eine Implementierungsklasse zugewiesen, die die gesamte Kommunikation zwischen den Forms-Properties und den Attributen des Java-API übernimmt.

Mit den Built-ins „SET_CUSTOM_PROPERTY“ und „GET_CUSTOM_PROPERTY“ können aus der Java Bean Methoden aufgerufen oder Werte abgefragt werden. Die Implementierungsklasse bereitet diese Informationen dann für das BI-Publisher-API auf.

In der Konfiguration des Forms-Servers werden die Implementierungsklasse und das Java-API als Jar-Archiv signiert, ähnlich wie schon für die von Oracle beigestellten Web-Utilities, und zum Download auf den Client zur Verfügung gestellt. Nach dem ersten

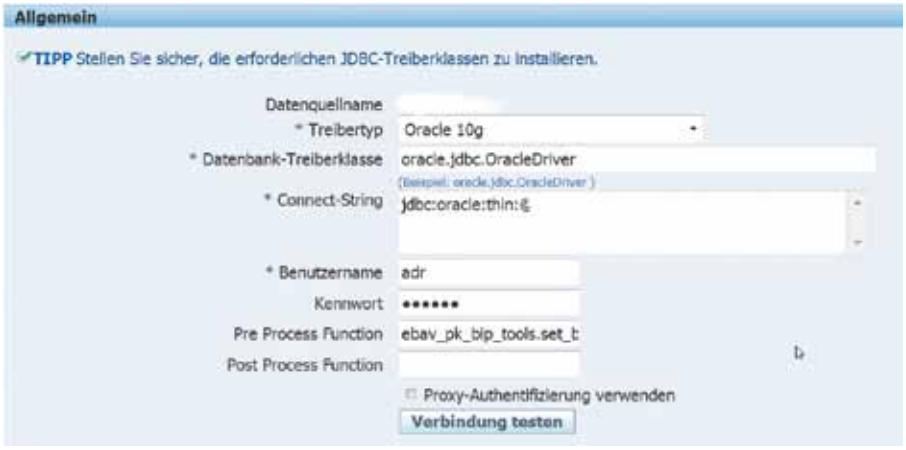


Abbildung 1: Der Pre-Trigger

Download auf den Client übernimmt das BI-Publisher-API die Kommunikation zum BI-Publisher-Server und somit ist der Applikations-Server von dieser Last und dem Traffic befreit.

Pre-Trigger und Bursting

Zwei weitere BI-Publisher-Features, die in diesem Projekt genutzt werden, sind die Pre-Trigger der Datenquellen und das Bursting. Die Pre-Trigger der Datenquellen sind perfekt dafür geeignet, den Kontext zu setzen, in dem der BI Publisher unterwegs ist.

Da in der Datenbank-Verbindung explizit ein Benutzer für den Datenbank-Connect eingegeben werden muss, es aber sicherzustellen gilt, dass jeder Forms-Anwender nur seine eigenen Daten beziehungsweise die Daten, an denen er Rechte besitzt, sehen darf, wird in dem Trigger über eine Datenbank-Funktion der Endbenutzer an die Datenbank weitergegeben, um so die Daten zu filtern (siehe Abbildung 1).

Außerdem ist der Einsatz der Bursting-Funktionalität entscheidend für das Projekt. In Abhängigkeit von der Adress-Art (E-Mail, Brief etc.) müssen verschiedene Ausgabekanäle des BI Publisher angesprochen und zudem unterschiedliche Layout-Vorlagen für die Dokumente genutzt werden. Um dieses möglichst komfortabel für den Endbenutzer zu gestalten, werden die Berichte durch die Bursting-Funktionalität aufgesplittet und mit den passenden Layout-Vorlagen an die entsprechenden Ausgabekanäle verteilt.

Fazit

Durch diese Implementierung und den Einsatz der verschiedenen BI-Publisher-Funktionen ist es gelungen, eine Forms-Anwendung mit einer modernen Reporting-Lösung zu schaffen. Zudem ist es dem Kunden nun möglich, den BI Publisher in Zukunft als zentrales Reporting-Tool in seiner bestehenden System-Landschaft zu nutzen.

Literaturhinweis

[1] How to integrate Oracle BI Publisher via Web Services in Oracle Forms, 2008, Dr. Jürgen Menge, Rainer Willems

Stephan La Rocca
sr@team-pb.de

Christian Piasecki
cpi@team-pb.de

Neu: Oracle stellt Database Appliance X3-2 vor

Die neue Version bietet zweimal höhere Leistungsfähigkeit und viermal mehr Storage-Kapazität im Vergleich zu früher. Kunden können damit das verfügbare Datenvolumen umgehend vergrößern, in dem sie einfach ein Storage Expansion Shelf anschließen, ohne dass weitere Verwaltungsaufgaben notwendig sind. Zudem enthält Oracle Database Appliance X3-2 optional eine virtualisierte Plattform, um auf Basis von Oracle VM mit der ISV vollständige Out-of-the-Box-Lösungen paketieren und ausliefern können.

Die Oracle Database Appliance Virtual Platform nutzt die Möglichkeiten von Oracle VM zum Hard-Partitioning. Außerdem erweitert sie das „Pay-as-you-grow"-Software-Lizenzierungsmodell auf sämtliche Software von Oracle.

Bei der Oracle Database Appliance handelt es sich um ein vollständiges Paket aus Software, Server, Storage und Netzwerk-Komponenten, das auf Einfachheit und Hochverfügbarkeit hin zusammen entwickelt wurde. Damit senken Unternehmen ihren Aufwand für die Installation, die Wartung und den Support ihrer Datenbanken.

Hardwareseitig hat Oracle die Oracle Database Appliance X3-2 mit 512 GB Speicher, 18 TB SAS Festplatten und 800 GB Flash Memory ausgestattet, um die Leistung im Online Transaction Processing und Data Warehousing zu beschleunigen. Im Vergleich zum Vorgängermodell ist sie damit bis zu zweimal schneller, verfügt über viermal mehr Storage-Kapazität, dreimal so viel Flash Speicher und über zweieinhalbmal so viel Arbeitsspeicher.

Weitere Informationen unter http://www.oracle.com/us/products/database/database-appliance/overview/index.html

Die Versorgung eines Data Warehouse (DWH) mit frischen Daten kann zuweilen eine große Herausforderung sein. Es sind in der Regel nicht nur ein DWH-System, sondern zumeist mehrere Systeme wie die Entwicklungs-, Test-, Integrations-, Wartungs- und Produktions-Umgebungen zu bedienen.

Global Staging Area: Implementierung einer zentralen Daten-Drehscheibe

Sven Bosinger, its-people GmbH

Jede der zu versorgenden Umgebungen hat spezielle Anforderungen. Gleichzeitig soll auf der Datenlieferanten-Seite die Anzahl der Schnittstellen, über die Daten abgegeben werden, überschaubar bleiben. Zusätzlich spielen regulatorische und rechtliche Vorgaben eine Rolle. Entwickler dürfen immer häufiger keinen Zugang mehr zu personalisierten Daten erhalten, sondern müssen auf maskierten und verfremdeten Daten entwickeln.

In der hier dargelegten Lösung geht es um die Implementierung einer zentralen Datendrehscheibe, die sogenannte „Global Staging Area" (GSA), die aus verschiedensten Quellsystemen mit Daten bestückt wird. Sie gibt wiederum die gepufferten Daten an die diversen DWH-Systeme gezielt weiter. Dadurch wird in jedem Quellsystem nur noch eine Schnittstelle benötigt, die damit Datenlieferant für alle nachgelagerten DWH-Systeme ist. In der GSA wird nach einem vorgegebenen Regelwerk entschieden, wann, wie und in welcher Form die Daten an die DWH-Systeme weitergegeben werden. So lässt sich ein permanenter Datenstrom mit allen Echtdaten an die Produktionsumgebung einrichten, wohingegen die Entwicklungsumgebung mit einem reduzierten und verfremdeten Datenbestand versorgt wird. Neue Quellsysteme können einfach über Oracle-Standard-Technologien (Streams, CDC, AQ, Trigger etc.) an die GSA angebunden werden.

Ausgangslage

Viele Anwender einer DWH-Lösung haben sich für einen klassischen Aufbau ihres DWH entschieden (siehe Abbildung 1). Dabei werden die Daten der Quellsysteme in einer Staging Area gesammelt, durch einen Batch-Lauf in ein zentrales Enterprise-Modell integriert und abschließend Business-Area-spezifische Data Marts aufgebaut.

In der Regel betreibt ein Anwender aber nicht nur eine DWH-Instanz, sondern mehrere. Je nach Vorgehen werden neben der Produktion noch weitere Instanzen für Entwicklung, Test, Abnahme und Wartung benötigt. Dies bedeutet, dass nicht nur eine Instanz permanent mit Daten aus den Quellsystemen versorgt werden muss, sondern viele. Ausgehend davon kommt man zu folgenden Datenanforderungen:

- *Produktions-Instanz*
  Regelmäßige Belieferung mit vollumfänglichem, unverfälschtem Datenbestand
- *Wartungs-Instanz*
  Regelmäßige Belieferung mit vollumfänglichem, unverfälschtem Datenbestand, um Fehler in der Produktion nachstellen zu können
- *Abnahme-Instanz*
  Regelmäßige Belieferung mit gegebenenfalls eingeschränktem und maskiertem Datenbestand, um Abnahmetests durchzuführen
- *Test-Instanz*
  Bedarfsgesteuerte Belieferung mit eingeschränktem und maskiertem (Test-) Datenbestand, um Tests durchzuführen
- *Entwicklungs-Instanz*
  Bedarfsgesteuerte Belieferung mit eingeschränktem und maskiertem (Test-) Datenbestand, um zu entwickeln

Bei einer DWH-Entwicklung besteht zudem die Besonderheit, dass eine erfolgreiche Entwicklung nur auf produktionsnahen Echtdaten und nicht auf

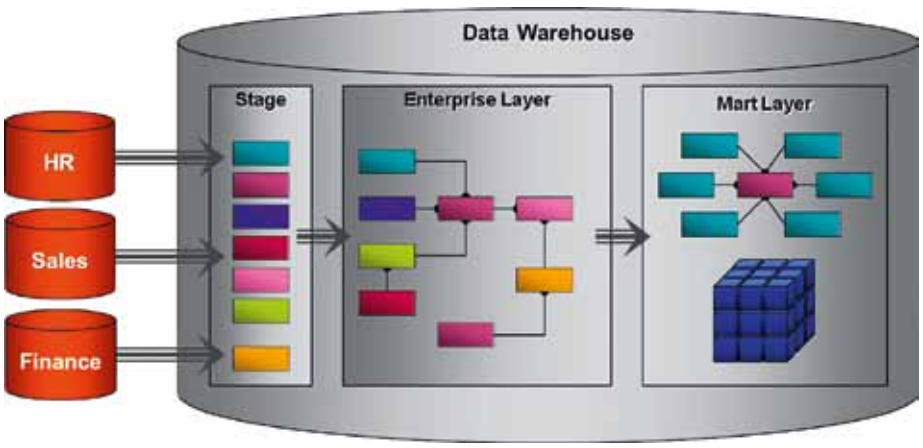


Abbildung 1: Das klassische DWH

Testdaten möglich ist. Jegliche DWH-Entwicklung ist eine Daten-getriebene Entwicklung. Fragen der Performance, statistische Auswertungen und Variationsvielfalt sind durch eingeschränkte Testdaten in der Regel nicht zu beantworten. Es ist also häufig notwendig, schon in den Entwicklungs-Instanzen mit Produktionsdaten zu arbeiten. Spätestens in der Abnahmeumgebung muss auf Produktionsdaten gearbeitet werden, um abnahmefähige Testfälle generieren zu können. Daher ergibt sich die Problematik, dass die produktiven Quellsysteme nicht nur mit dem Produktions-DWH über Schnittstellen verbunden werden müssen, sondern auch mit den übrigen DWH-Systemen. Dies führt zu einem überproportionalen Anwachsen der Anzahl der Schnittstellen (siehe Abbildung 2).

Aufgrund von Datensicherheitsaspekten dürfen häufig sensible Produktionsdaten, wie Kreditkarten-Informationen oder Bankdaten, nicht in eine ungeschützte Entwicklungsumgebung gelangen. Vor allem nicht, wenn bei der Entwicklung Near- oder Offshore-Kräfte eingesetzt werden sollen. Hier müssen Daten gegebenenfalls verfremdet oder ausgeblendet sein. Darüber hinaus bestehen branchenabhängige rechtliche Vorgaben, die eine Modifikation der Daten zwingend erforderlich machen. Diese Anforderungen müssen bei einem klassischen DWH-Ansatz mit lokalen Staging Areas in der jeweiligen Schnittstelle realisiert werden.

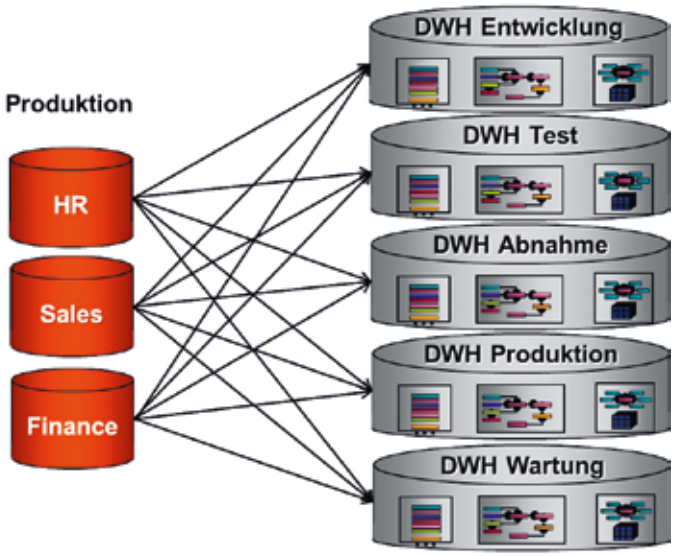


Abbildung 2: Schnittstellen-Explosion

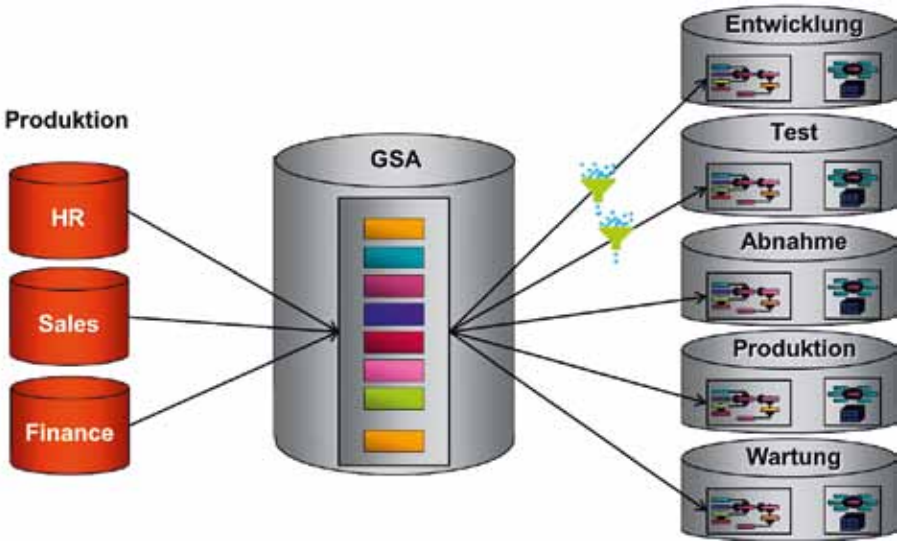


Abbildung 3: Global Staging Area

Global Staging Area

Das Verfahren der Global Staging Area (GSA) ersetzt alle lokalen Staging Areas in den einzelnen DWH-Instanzen (siehe Abbildung 3). Alle DWH-Instanzen verarbeiten die Stage-Daten weiterhin im Rahmen eines klassischen ETL-Prozesses. Insofern wird auf die GSA zugegriffen, als ob es sich um eine klassische, lokale Staging Area handeln würde. Die Quellsysteme werden ausschließlich über Schnittstellen an die GSA angebunden (siehe Abbildung 4). Daher muss für jedes Quellsystem nur noch eine Schnittstelle definiert werden, egal, wie viele DWH-Instanzen bedient werden müssen. Dabei kommt ausnahmslos ein Push-Verfahren zum

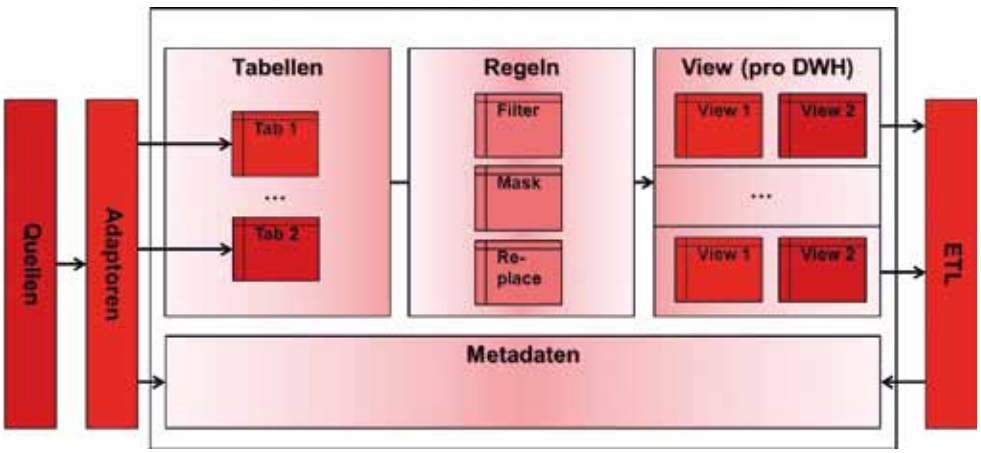


Abbildung 4: Interner Aufbau der GSA

Einsatz, die Quellsysteme stellen also die Datenlieferungen zusammen und übertragen diese direkt in die GSA. Dabei werden die technischen Verfahren „Change Data Capture“, „Advanced Replication“, „Advanced Queuing“, „Streams“ und „Trigger“ (via Database Link) unterstützt.

Die Quellsysteme liefern die Produktionsdaten umfänglich, die Daten werden also so geliefert, wie sie im Produktions-DWH erforderlich sind. Etwaige Filterungen, Daten-Maskierungen oder Verfremdungen erfolgen innerhalb der GSA. Diese Aktivitäten sind DWH-Instanz-spezifisch, die Produktionsumgebung erhält somit die Daten ungefiltert, wohingegen die Daten für die Test-Umgebung gefiltert und Konto-Informationen maskiert werden können. Die Bereitstellung der Daten für die nachgelagerten ETL-Prozesse ist von der GSA durch entsprechende Instanz-spezifische Sichten auf die Stage-Daten gewährleistet. Während des Aufbaus dieser Sichten werden dabei metadatengesteuert die entsprechenden Filterregeln, Maskierungen und Verfremdungen angewandt. Jede DWH-Instanz erhält nur Zugriff auf seine Sichten.

Prozesse

Der Betrieb der GSA erfolgt prozessgesteuert. In der GSA werden hierzu drei (optional vier) Prozesse betrieben:

- *Push in die GSA*
  Die Quellsysteme liefern die Daten in Real/Near Time in die GSA. Dort werden sie ungeprüft in die entsprechenden Stage-Tabellen eingefügt. Jeder Datensatz wird mit einer systemweiten, eindeutigen DWH-ID und einem Liefer-Datum versehen. Zusätzlich wird im Metadaten-Katalog der Status jedes einzelnen Datensatzes DWH-Instanz-bezogen festgehalten. Das Einfügen der Daten in die GSA ist transaktionsgesichert (siehe Abbildung 5).
- *Pull aus den DWHs*
  Die Verarbeitung der Daten aus der GSA in der jeweiligen DWH-Instanz erfolgt mithilfe eines klassischen ETL-Prozesses. Die zu verarbeitenden Sätze werden aus Instanz-spezifischen Views gelesen. Ebenso wie die erfolgreiche Verarbeitung werden Fehler im Metadatenkatalog

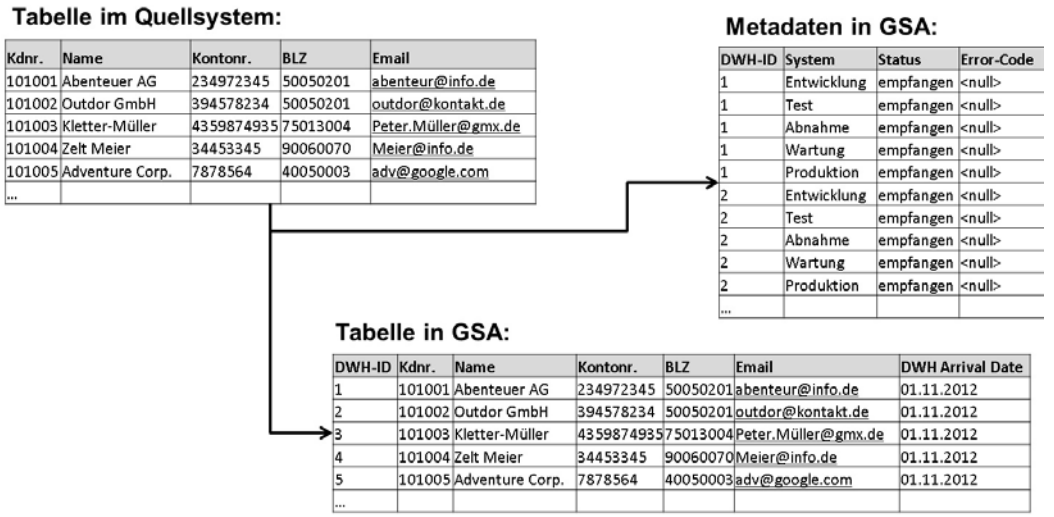


Abbildung 5: Push in die GSA

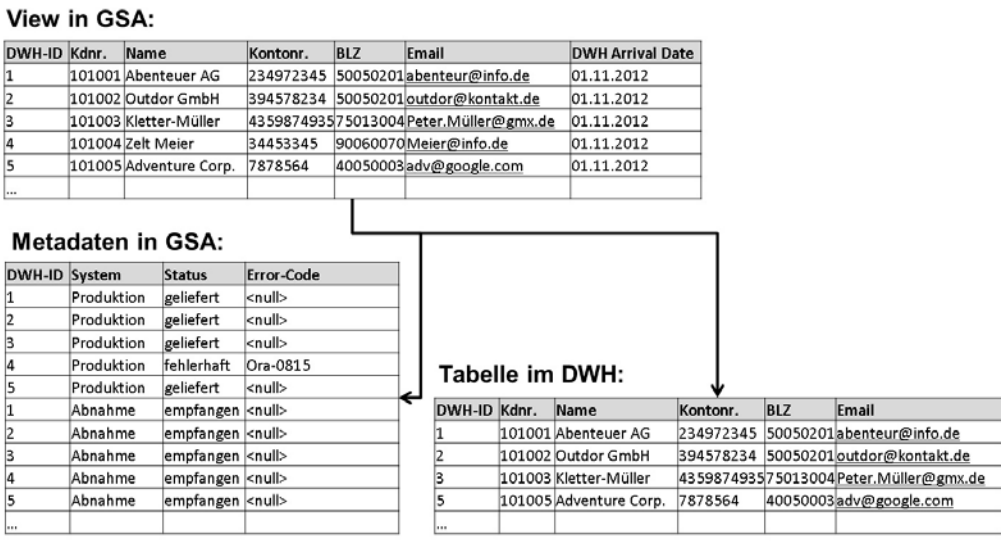


Abbildung 6: Pull aus dem DWH

protokolliert. Die Views werden pro Instanz unter Berücksichtigung der Instanz-spezifischen Filterregeln, Maskierungen und Verfremdungen erzeugt. Der Pull pro Instanz kann völlig unabhängig von allen anderen Instanzen betrieben werden. Unterschiedliche Ladezeitpunkte und Frequenzen sind problemlos realisierbar (siehe Abbildung 6).

- *CleanUp der SGA*
  Alle erfolgreich in die DWH-Instanzen geladenen Daten, erkenntlich über den Status im Metadatenkatalog, werden regelmäßig asynchron gelöscht. Dazu kann eine Vorhaltezeit definiert werden, sodass die Daten zu Nachverfolgungszwecken einen definierten Zeitraum in der GSA verbleiben (siehe Abbildung 7).
- *Real-/Near-Time-Auswertungen (optional)*
  Solange die Daten in der GSA stehen, können sie zusätzlich noch zu Auswertungszwecken genutzt werden. Sobald sie aus der GSA gelöscht wurden, stehen sie im DWH bereit. Real-/Near-Time-Auswertungen können somit auf der SGA einfach realisiert werden.

Performance

Beim Betrieb einer GSA wird mit Massendaten gearbeitet. Daher ist es notwendig, sich beim Design der GSA über die damit verbundenen Performance-Aspekte Gedanken zu machen. Nachfolgend ein paar Hinweise, ohne den Anspruch auf Vollständigkeit:

- *Push in die GSA*
  Der Upload der Daten soll aus den Quellsystemen initiiert werden. Dabei sollten nach Möglichkeit die Change-Data-Capture-Verfahren zum Einsatz kommen. Der Vorteil dieser Technologie besteht darin, dass es zu keiner zusätzlichen Belastung der Quellsysteme kommt. Die Redo Logs der Quellsysteme können hierbei zudem asynchron auf der GSA-Instanz ausgewertet werden.
- *Pull aus den DWHs*
  Die einzelnen DWH-Instanzen verarbeiten die GSA-Daten in einem Batch. Es wird also eine Mengen- und keine Einzelsatzverarbeitung durchgeführt. Das Protokollieren der verarbeiteten Sätze in den Metadaten-Tabellen der GSA erfolgt ebenfalls über Massen-Updates. Durch eine geeignete Partitionierung der zugrunde liegenden Tabellen können alle später zu löschenden Daten gezielt in die entsprechenden Partitionen gelegt werden.
- *CleanUp der SGA*
  Das Löschen von Daten ist in der Regel ein sehr aufwändiger Prozess. Daher sollte ein klassisches Löschen durch ein Delete-Statement vermieden werden. Effektiver ist es, alle zu löschenden Datensätze in Partitionen vorzuhalten und dann die gesamten Partitionen zu entfernen. Dies verhindert auch die Fragmentierung der Daten-Tabellen in der GSA.
- *Architektur*
  Es hat sich bewährt, die GSA in einer eigenen Datenbank-Instanz zu betreiben. Die Daten-Tabellen der GSA sollten partitioniert sein. Als Kriterium dafür kann das Arrival-Datum dienen. Es werden Tages-Partitionen gebildet. Der Vorteil ist, dass später nach der gewünschten Aufbewahrungsfrist die gesamte Tages-Partition komplett entfernt werden kann. Dazu müssen gegebenenfalls vorher die noch fehlerhaften Datensätze, die in der GSA verbleiben sollen, in eine nicht zu löschende Fehlerpartition ausgelagert werden.

Fazit

Wie dargelegt, bietet der Einsatz einer GSA eine Reihe von Vorteilen:

- Die Anzahl der Schnittstellen in den Quellsystemen wird massiv reduziert. Pro Quellsystem ist lediglich eine Schnittstelle notwendig, um beliebig viele DWH-Instanzen mit Daten zu versorgen. Dadurch sinkt die Anzahl der Schnittstellen.
- In den Schnittstellen der einzelnen Liefersysteme wird keine zusätzliche

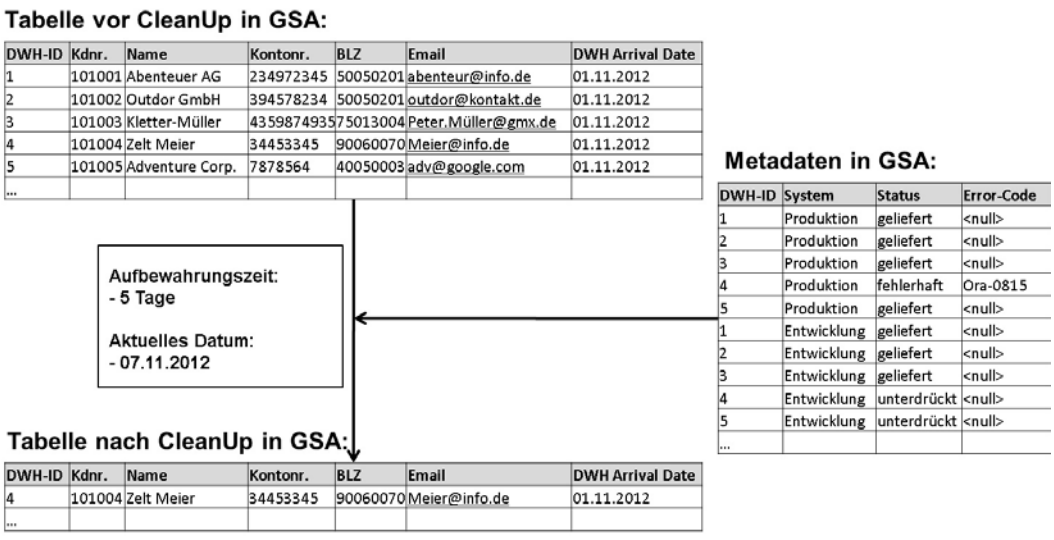


Abbildung 7: CleanUp der SGA

Logik benötigt; Verfremdung und Filterung der Daten wird bei Bedarf in der GSA durchgeführt. Dadurch wird die Komplexität der Schnittstellen reduziert.

- Alle DWH-Instanzen sind permanent über ein Push-Verfahren mit aktuellen Echtdaten versorgt. Dadurch kann bei der Entwicklung schon auf realistischen Datenmengen gearbeitet werden. Darüber hinaus sind alle vorkommenden Daten-Konstellationen berücksichtigt.
- Die Datenmenge in der GSA ist deutlich geringer als die Datenmenge aller lokalen Staging Areas zusammen. Jeder Datensatz wird nur einmal gespeichert und kann an beliebig viele DWH-Instanzen verteilt werden. Die Sichten sind lediglich logischer Natur und als Datenbank-Views aufgeteilt auf ein Schema pro DWH-Instanz.
- Die Datenmengen lassen sich bei Bedarf für einzelne Instanzen einfach durch konfigurierbare Filterregeln reduzieren.
- Sicherheitsrichtlinien werden durch Metadatenkonfiguration einfach und transparent umgesetzt, so können Maskierung und Verfremdung der Daten Instanz-abhängig eingestellt werden.
- Das komplette Verfahren ist transaktionsgesichert, es können damit keine Datensätze verloren gehen. Kommt es zu Abbrüchen oder Lieferausfällen, so werden keine unvollständigen Lieferungen gespeichert und damit auch nicht im ETL-Prozess verarbeitet.
- Durch das Push-Verfahren sind die Daten in Real/Near Time in der GSA gespeichert, was beispielsweise ein Real-/Near-Time-Reporting einfach macht.
- Zusätzliche DWH-Instanzen, die eventuell sogar nur temporär benötigt werden, sind einfach und schnell mit Daten zu versorgen. Lediglich die Instanz-bezogenen Views sind einmalig zu erzeugen.

Größere Nachteile waren beim Einsatz einer GSA bisher nicht feststellbar. Die Implementierung ist bei Vorliegen der beschriebenen Vorbedingungen (mehrere DWH-Instanzen greifen auf die gleichen Quellesysteme zu) uneingeschränkt zu empfehlen.

Sven Bosinger
sven.bosinger@
its-people.de

Alle Daten, die wir in den Datenbanken speichern, sind der Rohstoff für Informationen. Damit die Informationen vom Empfänger einfach und ohne Nachfrage verstanden werden, würden allgemeingültige Berichtsstandards die Kommunikation erleichtern. Der Artikel zeigt das schrittweise Umsetzen mit der Oracle Business Intelligence Enterprise Edition (OBIEE).

Einheitliche Berichte, damit Empfänger die Informationen (besser) verstehen

Andreas Nobbmann und Heinz Steiner, Trivadis AG

Ohne dass es uns beispielsweise beim Lesen von Straßenkarten wirklich bewusst ist, werden die Karten einheitlich standardisiert dargestellt und wir finden uns sofort zurecht, sowohl zu Lande als auch zu Wasser. Bei der Management-Kommunikation fehlte ein solcher Standard bisher. Seit einiger Zeit wird dem Thema „Informations-Design" zunehmend Beachtung geschenkt, dies dank dem Notationskonzept von Prof. Dr. Ing. Rolf Hichert, das als Hichert®Success zunehmend vom Management und Controlling akzeptiert und gefördert wird (siehe http://www.hichert.com/de/success). Es rückt die Lesbarkeit der Informationen in den Fokus. Hichert bringt das Konzept auf die Kurzformel „Success". In dem Akronym haben die Buchstaben folgende Bedeutung:

- *Say*
  Botschaft vermitteln; was soll dem Empfänger mitgeteilt werden?
- *Unify*
  Bedeutung vereinheitlichen; wird Gleiches gleich und Unterschiedliches anders dargestellt?
- *Condense*
  Informationen verdichten
- *Check*
  Korrekte Darstellung von Informationen in Tabellen und Diagrammen sicherstellen
- *Enable*
  Notationskonzept umsetzen unter Einsatz geeigneter BI- und Reporting-Tools
- *Simplify*
  Redundanzen bei Informationen reduzieren; wird Dekoration und Rauschen weggelassen?
- *Structure*
  Informationen gleichartig, erschöpfend und überschneidungsfrei aufbereiten

Management und Controller sind vom Success-Ansatz einfach zu überzeugen, doch von der Idee bis zur Umsetzung ist es ein steiniger Weg. Die BI-Tools erlauben die Realisierung der Anforderungen nur teilweise. Es braucht ein umfangreiches Tool-Wissen, um den Vorgaben gerecht zu werden. Wir versuchen mit den Standardmitteln von OBIEE, die folgende Analyse nachzubilden, die den Success-Richtlinien folgt.

Die Vorgabe beinhaltet einige schwer zu knackende Nüsse. „Konzentration auf das Wesentliche" bedeutet, dass die Einheiten so gewählt werden, dass die Werte mit max. vier bis fünf Stellen dargestellt werden können. Die Spaltenbeschriftungen sind rechtsbündig, dafür entfallen Gitterrahmen. Die Lesbarkeit wird durch Einrücken und Hervorheben der Summen erreicht. Die aktuellen Werte sind am grauen Hintergrund auf den ersten Blick erkennbar. Bei dieser Notation gibt es keine Farben ohne eine explizite Bedeutung. Typisch für die Hichert-Notation ist die einheitliche Darstellung der Szenarien.

Diese Regel wird in Tabellen und Diagrammen angewendet; so wird sofort ersichtlich, welche Bedeutung die Werte haben.

Die Kennzahlen werden aus Gründen der Lesbarkeit unterschiedlich formatiert – die Summen fett, die Abweichungen immer mit einem Vorzeichen.

Umsetzung im OBIEE

Da OBIEE nicht „out of the box" alle Success-Konzepte umsetzen kann, sind einige Vorbereitungen zu treffen. Wichtig ist zuallererst das Modellieren der Daten. Hierbei sollte darauf geachtet werden, dass die Attribute der Dimensionen in Zeilenform vorliegen. Dies wird erreicht über das Implementieren einer neuen Dimension „Kennzahlen", die Werte wie „Vorjahr", „Actual", „Budget" und „Forecast" enthält. Dazu sind die notwendigen Foreign Keys mit einem ETL-Werkzeug oder per SQL in die Fakten-Tabelle einzufügen.

Danach werden die Symbole vorbereitet, die in Success einheitlich verwendet werden und überall dieselbe Bedeutung haben. Diese müssen in die OBIEE-Installation integriert werden. Dafür können die Symbole im Bildformat in die entsprechenden Skin-Verzeichnisse der OBIEE-Installation kopiert (die Verzeichnisse sind FMW_HOME/Oracle_BI1/bifoundation/web/app/res/s_blafp/images und FMW_HOME/user_projects/domains/bifoundation_domain/servers/bi_server1/tmp/_WL_user/analytics_11.1.1/7dezjl/war/res/s_blafp/images) und nachfolgend über die FMAP-Funktionalität referenziert werden. Listing 1 zeigt, wie die Funktion einer solchen Spalte in OBIEE beispielsweise aussehen kann.

```
CASE  WHEN "Scenario"."Scenario" || , , || "Time"."Jahr" = 'Ist Y' || VALUEOF(NQ_SESSION.AktJahr)-1
      THEN "Scenario"."Scenario" || , 10 <br/><imgsrc="res/s_blafp/meters/success/IstVJ.jpg"'
      ELSE
      CASE WHEN "Scenario"."Scenario" || , , || "Time"."Jahr" = 'Ist Y' || VALUEOF(NQ_SESSION.AktJahr)

             THEN "Scenario"."Scenario" || , 11 <br/><imgsrc="res/s_blafp/meters/success/IstLJ.jpg"'
      ELSE
             CASE WHEN "Scenario"."Scenario"  = ,Bud , || VALUEOF(NQ_SESSION.Year)
                    THEN "Scenario"."Scenario" || , <br/><imgsrc="res/s_blafp/meters/success/BudLJ.jpg"'
             ELSE "Scenario"."Scenario"
             END
      END
END
```

Listing 1

Um das Administrieren der Überschriften und Beschreibungen zu vereinfachen, können die Texte in OBIEE ausgelagert werden. Dies wird im Presentation-Layer mittels rechter Maustaste auf die entsprechende „Subject Area" sowie die Auswahl von „Externalize Display Names" und „Externalize Descriptions" erreicht.

Nachfolgend werden die Texte über die Utilities exportiert, in einer externen Tabelle abgelegt und verwaltet. In der Tabelle können gleichzeitig mehrere Sprachen gespeichert sein. Ein weiterer Vorteil ist, dass man auch HTML-Texte in dieser Tabelle ablegen kann. Ein „Init Block" im Repository übernimmt das Auslesen der Daten, sodass beim Log-in eines Benutzers die passenden Übersetzungen ermittelt und im Browser angezeigt werden (siehe http://blog.trivadis.com/b/andreasnobbmann/archive/2008/09/22/multilanguage-support-in-use-variables-over-variables.aspx und http://blog.trivadis.com/b/andreasnobbmann/archive/2009/08/07/external-strings-in-obiee-not-only-useful-for-localization.aspx).

Um unsere Zielanalyse wie gewünscht zu erstellen, ist es notwendig, eine Pivot-Tabelle in OBIEE zu verwenden; für die Formatierungen sind die Funktionalitäten von OBIEE mehr als ausreichend. Es werden „Conditional Formatting" und die normale Formatierung der Zellen verwendet. Hintergrundfarben und Rahmeneinstellungen können in OBIEE über „Format Values" formatiert werden.

Um die Formatierung nicht mehrfach manuell vornehmen zu müssen, bietet OBIEE als kleines Hilfsmittel die Möglichkeit an, eine Formatierung zu kopieren und auf eine andere Spalte zu applizieren. Für „Conditional Formatting" können wir einmalig die Formatierung konfigurieren und auf die entsprechenden Spalten legen.

Das Ergebnis

Abbildung 1 zeigt das Resultat. Zwar lässt sich die Vorlage allein dadurch noch nicht 100-prozentig nachbilden, denn hierfür fehlen noch die grafischen Abweichungen, das Ergebnis in OBIEE kann sich jedoch schon sehen lassen.

Fazit

Mit OBIEE lassen sich unter Verwendung eingebauter Funktionalitäten mit ein wenig manueller Arbeit Analysen nach den Success-Regeln bis zu einem gewissen Grad einfach standar-

Muster AG

Ist April..Mai 2011
in TCHF

	April					Mai				
	Ist 10	Ist 11	Bud	ΔBud 11	Δ% Bud 11	Ist 10	Ist 11	Bud	ΔBud 11	Δ% Bud 11
T & M Umsatz	4,127.2	5,137.2	4,300.0	837.2	19.0	2,123.1	3,189.2	2,200.0	989.2	45.0
Projektumsatz	10,269.3	10,807.3	11,350.3	-543.0	-5.0	13,288.0	8,461.0	13,350.6	-4,889.6	-37.0
Bruttoumsatz	**14,396.5**	**15,944.5**	**15,650.3**	**294.2**	**2.0**	**15,411.0**	**11,650.2**	**15,550.6**	**-3,900.4**	**-25.0**
Erloesminderungen	156.0	144.0	150.0	-6.0	-4.0	101.0	158.0	112.0	46.0	41.0
Erloesminderungen in % BU	1.1	0.9	1.0	0.0	0.0	0.7	1.4	0.7	0.0	0.0
Nettoumsatz	**14,240.5**	**15,800.5**	**15,500.3**	**300.2**	**2.0**	**15,310.1**	**11,492.2**	**15,438.6**	**-3,946.4**	**-26.0**
Eigenleistungen	10,022.0	10,649.0	10,400.0	249.0	2.0	10,013.0	8,649.0	8,500.0	149.0	2.0
Fremdleistungen	90.0	24.0	50.0	-26.0	-52.0	17.0	13.0	15.0	-2.0	-13.0
Standard Herstellerkosten	10,112.0	10,673.0	10,450.0	223.0	2.0	10,030.0	8,662.0	8,515.0	147.0	2.0
DB 1	**4,128.5**	**5,127.5**	**5,050.3**	**77.2**	**2.0**	**5,280.1**	**2,830.2**	**6,923.6**	**-4,093.4**	**-59.0**
DB 1 in % NU	29.0	32.5	32.6	0.0	0.0	34.5	24.6	44.8	0.0	0.0

Abbildung 1: Hervorhebung von Vorjahr, Laufjahr, Budget und Forecast

disieren. Dazu ist es notwendig, die Features von OBIEE gut zu kennen und gezielt einzusetzen. Aber der Aufwand lohnt sich: Die Berichtsempfänger verstehen die Informationen besser. So wird der Wert der im Data Warehouse gespeicherten Daten besser erkannt.

Andreas Nobbmann
andreas.nobbmann@
trivadis.com

Heinz Steiner
heinz.steiner@
trivadis.com

Tipps und Tricks aus Gerds Fundgrube

Heute: Vererbungs-Probleme und deren Lösung

Gerd Volberg, OPITZ CONSULTING GmbH

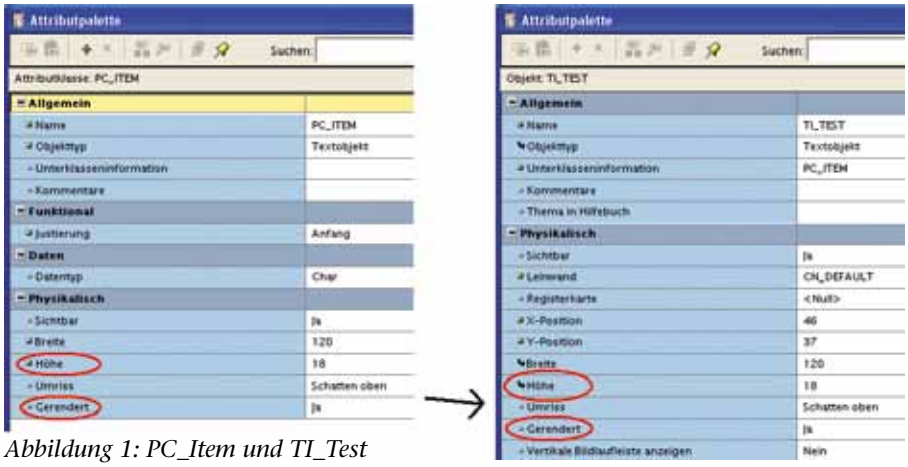


Abbildung 1: PC_Item und TI_Test

Aufgrund eines Versehens haben wir in der letzten Ausgabe den Beitrag mit nur einer Abbildung abgedruckt. Dadurch konnte der Inhalt nicht richtig verstanden werden. Deshalb bringen wir hier nochmals den vollständigen Artikel.

Wenn man in Oracle Forms mit Property-Klassen arbeitet, sollte man auf Default-Werte achtgeben. Betrachten wir das Problem an einem Beispiel: Gegeben seien eine Klasse „PC_Item“ und ein Feld „TI_Test“, das aus dieser Property-Klasse vererbt wurde (siehe Abbildung 1).

Vererbte Properties erkennt man an dem schwarzen Pfeil vor dem Namen – nicht jedoch an der Property „Gerendert“. Dies liegt daran, dass in der Property-Klasse der Defaultwert „Ja“ hinterlegt wurde.

Wenn man nun in der Property-Klasse den Wert „Ja“ auf „Nein“ und danach wieder auf „Ja“ ändert, wird aus dem kleinen runden Kreis ein grünes Quadrat. Dieses Flag zeigt an, ob der Wert ein Default-Wert oder ein veränderter Wert ist (siehe Abbildung 2).

Mit diesem einfachen Workaround sorgt man dafür, dass in allen vererbten Properties die korrekte Vererbungsinformation zu sehen ist (siehe Abbildung 3). Arbeitet die eigene Vererbungsstrategie auf Basis von Property-Klassen, sollte man dafür sorgen, dass jeder Default-Wert in den Klassen überarbeitet wird.

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

Abbildung 2: Default-Wert überschreiben

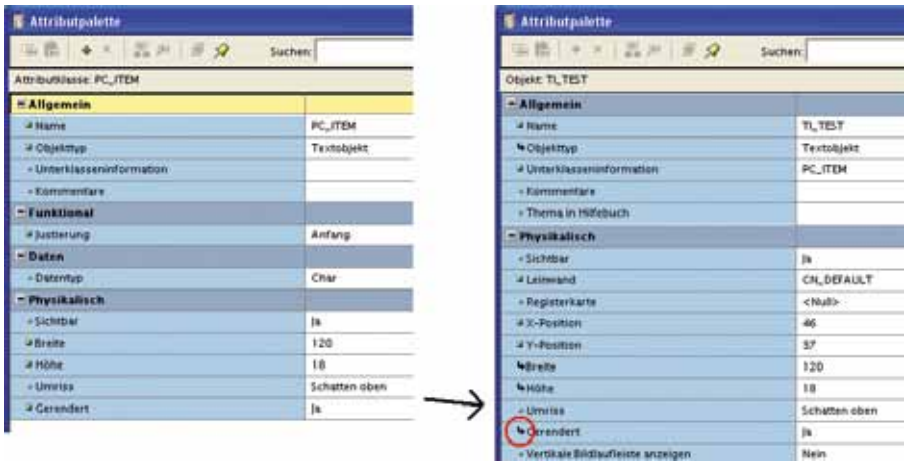


Abbildung 3: Property mit dem korrekten Vererbungshinweis

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

„Die IT bietet zahlreiche Möglichkeiten, sich weiterzuentwickeln …"

Welchen Beruf üben Sie aus?
Haller: Zurzeit bin ich bei der Tieto Deutschland GmbH beschäftigt und arbeite als Projektleiterin in einem großen IT-Projekt. Wir setzen Software eines Anbieters beim Kunden ein. Diese muss für den Kunden angepasst und erweitert werden. Dreimal jährlich werden neue Releases entwickelt und ausgeliefert. Bei jedem Release gibt es neben Verbesserungen bereits existierender Anwendungen auch Neuerungen und Erweiterungen. Meine genaue Stellenbezeichnung ist Solution Consultant. Das Aufgabenspektrum umfasst neben der Projektleitung in verschiedenen Releases auch die Bereiche Beratung, Erfassen von Anforderungen des Kunden und Finden und Designen von Lösungen.

Auf welchem Weg sind Sie dorthin gekommen?
Haller: Nach dem Informatik-Studium, Familienphase und verschiedenen Teilzeitarbeiten führte mich der Weg über eine Weiterbildungsmaßnahme des Arbeitsamts zurück in die IT. Zunächst arbeitete ich als Software-Entwicklerin mit Schwerpunkt „Datenbanken". Nach Tätigkeiten in der Oracle-Datenbank-Administration und der Oracle-Datenbank-Entwicklung war ich Teil-Projektleiterin mit Schwerpunkt „Datenbanken" und schließlich Projektleiterin.

Was hat Sie motiviert, diesen Beruf zu ergreifen?
Haller: Als ich mit dem Studium begann, war Informatik ein neuer Studiengang. Die Nähe zur Mathematik und gleichzeitig die Aussicht, kreativ und gestalterisch tätig werden zu können, waren für mich die wichtigsten Entscheidungsgründe für dieses Fach. Das Finden von Lösungswegen und die vielen verschiedenen Gestaltungsmöglichkeiten zeigten sehr eigenständige Entwicklungs-Chancen auf. Darüber hinaus schien mir die Informatik so vielschichtig – von der reinen Hardware-Technik bis hin zur fachlichen Kundenberatung –, dass sich für das Berufsleben immer wieder neue, alternative Wege aufzeigten und ich mich nicht auf einen vorgegebenen Weg festgelegt sah.

Wie sehen Sie generell die Rolle der Frau in der IT?
Haller: Positiv. Frauen sind oft weniger Technik-affin. Dies hilft in gemischten Teams häufig, Probleme von unterschiedlichen Seiten aus zu betrachten und die bestmöglichen Lösungswege zu erkennen. Durch kommunikative Stärke und Überzeugungskraft schaffen es Frauen oft besser, die einzusetzende Technik dem Kunden verständlich nahe zu bringen. Reinen Männer-Teams fehlt meiner Meinung nach immer ein wesentlicher Blickwinkel – genauso wie reinen Frauen-Teams auch.

Bietet die IT-Branche Frauen die Möglichkeit, ihre Stärken einzusetzen?
Haller: Ja, genauso umfangreich und vielseitig wie Männern auch. IT-Tätigkeiten sind meist in Projekten organisiert. Hier sind kreative und kommunikative Stärken gefordert, genauso wie fundiertes Wissen. Dieses Wissen umfasst neben technischen Kenntnissen auch die Fähigkeit, die Dinge aus Kundensicht zu sehen und Lebenserfahrung aus den unterschiedlichsten Bereichen einzubringen. So sind Menschen mit unterschiedlichen Erfahrungen in Projekten von unschätzbarem Wert: Ein IT-Berater für Bank-Software, der beispielsweise eine Lehre als Bankkaufmann absolviert hat, versteht die Sichtweise und Anforderungen des Kunden besser und schneller; der starke Techniker kann die technische Umsetzbarkeit des Kundenwunschs beurteilen.

Zur Person: Ulrike Haller
Ulrike Haller begann 1979 ihr Studium der Informatik an der TU in München. Nach dem Diplom und einer daran anschließenden Familienphase mit Jobs in verschiedenen Branchen führte durch eine Umschulung im Jahr 2000 der Weg zurück in die IT. Zunächst war sie als Software-Entwicklerin und Oracle-DBA tätig. Als sich die Aufgaben in diesen Bereichen zu wiederholen begannen, suchte und fand Ulrike Haller neue Herausforderungen als Teil-Projektleiterin und schließlich Projektleiterin. Neben und zwischen den Projekten berät sie Kunden und Kollegen in der Anforderungs- und Lösungsspezifikation. In ihrer Freizeit ist sie in den Münchner Hausbergen unterwegs.

Was könnte Frauen motivieren, einen Beruf in der IT zu ergreifen?
Haller: Die IT bietet zahlreiche Möglichkeiten, sich weiterzuentwickeln. In der Informatik gibt es sowohl für technisch versierte Menschen als auch für kreative Köpfe ganz unterschiedliche Entwicklungswege. Gerade im Bereich der Datenbank-Entwicklung sind Ideen und das Finden von Lösungswegen deutlich wichtiger als das technische Wissen. Wer irgendwann keine Lust mehr auf Technik und Entwicklung hat, dem bieten sich Wege als Berater oder Projektleiter an; hier sind vor allem Lebenserfahrung und Kommunikationsfähigkeit notwendig.

Welche Eigenschaften sollte eine Frau mitbringen, um sich in der IT-Branche durchzusetzen?
Haller: Zunächst muss eine Frau in der IT die gleichen Eigenschaften mitbringen wie in allen anderen Branchen auch. Wer sich durchsetzen will, braucht Kraft, Selbstbewusstsein, eine fundierte Ausbildung und Spaß an dem, was sie macht. Darüber hinaus sollte Frau gut mit Männern zusammenarbeiten können.

Was kann eine Anwendervereinigung wie die DOAG tun, damit mehr Frauen in die IT kommen?
Haller: Sicherlich liegt der Schwerpunkt der Interessen der Mitglieder bei Informationen zur Technologie sowie Tipps und Tricks zum täglichen Umgang mit den Produkten von Oracle. Die DOAG sollte darüber hinaus versuchen, die Berufswelt in der IT in ihrer Vielseitigkeit darzustellen.

Die IT ist in Deutschland mit den Stempeln „Technik-verliebt" und „mathematisch" versehen. Hier gilt es, die Möglichkeiten von Karrieren in der IT für Quereinsteiger und kreative Menschen besser herauszustellen, etwa durch geeignete Pressemitteilungen, Blogs, Informationsveranstaltungen an Schulen und Universitäten, außerdem Präsentationen auf Konferenzen. Wenn diese Informationen von Frauen übermittelt werden, so wird hier ein klar sichtbares Signal gesetzt.

Was erwarten Sie von einem IT-Unternehmen wie Oracle?
Haller: Von Oracle erwarte ich, den Frauen-Anteil zu stärken, beispielsweise indem mehr Vorträge auf den Konferenzen von Frauen gehalten oder Blogs von Frauen geführt werden. Für mich sind zu wenige Oracle-Frauen nach außen hin sichtbar.

Was wünschen Sie sich für die Zukunft?
Haller: Ich wünsche mir vor allem, dass die IT sich einbringt und einmischt in gesellschaftliche Themen wie Klimaschutz oder Bildung. In Deutschland ist Informatik oder Technik ein stark unterrepräsentiertes Unterrichtsfach, gerade an den Schulen sollte die Vielseitigkeit der IT viel stärker vermittelt werden.

Dr. Dietmar Neugebauer
Vorstandsvorsitzender der DOAG

Finanzbericht und Delegiertenversammlung

Auf seiner ersten Sitzung im neuen Jahr beschäftigte sich das DOAG Leitungsgremium mit den Finanzberichten des Vereins und der DOAG Dienstleistungen GmbH für das Jahr 2012 sowie mit den Vorbereitungen zur ersten Delegiertenversammlung im Juni dieses Jahres.

Der Verein hat im vergangenen Jahr einen Überschuss von knapp 2.000 Euro und die DOAG Dienstleistungen GmbH einen Gewinn von rund 34.000 Euro erwirtschaftet. Eine detaillierte Zusammenstellung der Bilanzen erfolgt wie immer im jährlichen Finanzbericht, der kurz vor der Delegiertenversammlung allen Mitgliedern zugänglich gemacht wird.

Die erste Delegiertenversammlung der DOAG wird das Ziel haben, die in der Satzung zusammengestellten Vereinszwecke daraufhin zu überprüfen, wie ihre Erfüllung auch in Zukunft gewährleistet werden kann. Hier sind entsprechende strategische Entscheidungen zu diskutieren und zu beschließen. Ein weiterer wichtiger Punkt wird die zukünftige Arbeit in den Regionalgruppen sein. Auch hier stellt sich die Frage, wie die so wichtige lokale Präsenz der DOAG zeitgemäß gestaltet werden kann. Aufgrund der Repräsentanz aller Mitgliedergruppen ist die Delegiertenversammlung das richtige Gremium, um die notwendigen Weichen zu stellen. Der auf der Delegiertenversammlung neu gewählte Vorstand wird dann die Aufgabe haben, Maßnahmen zur Umsetzung der Entscheidungen aufzusetzen.

Christian Trieb
Leiter Datenbank Community

Neues aus der Datenbank Community

Anfang März 2013 trafen sich die Mitglieder der Datenbank Community in Bad Soden/Taunus, um die Aktivitäten des Jahres 2013 zu planen und vorzubereiten. So wurden die letzten Details der Community-Konferenz DOAG

2013 Datenbank, die am 14. Mai 2013 in Düsseldorf stattfindet, besprochen und abgestimmt.

Breiten Raum nahm die DOAG 2013 Konferenz + Ausstellung vom 19. bis 21. November 2013 in Nürnberg ein. Die Vortrags-Streams sind „Oracle Datenbank“, „MySQL“ sowie „Oracle und SAP“. Aufgrund des großen Erfolgs im vergangenen Jahr wird es am Dienstag, 19. November 2013, wieder einen Datenbank-Community-Abend in einem Restaurant in der Nürnberger Altstadt geben. Hinzu kommt ein Workshop der Datenbank Community während der Konferenz.

Im Schwerpunkt „Veranstaltungen“ wurde festgelegt, dass die SIG MySQL ihre Events in Kombination mit anderen DOAG-Veranstaltungen durchführt. So war beispielsweise die Kombination der SIG MySQL mit der SIG Database an zwei aufeinanderfolgenden Tagen am selben Ort ein guter Erfolg. Die Datenbank-Webinare werden sehr gut angenommen. In diesem Jahr gibt es nur noch vier Termine, für die noch Themen und Referenten gesucht werden. Vorschläge und Wünsche bitte an christian.trieb@doag.org.

Auch die DOAG Delegiertenversammlung wurde aus Sicht der Datenbank Community vorbereitet. Das Treffen zeigt, dass die Mitglieder der Community in Zusammenarbeit mit Oracle auf einem guten Weg sind, das Datenbank-Thema innerhalb der DOAG weiterzuentwickeln, damit die Mitglieder weiterhin über eine gute Basis zum Erfahrungsaustausch und über entsprechende Möglichkeiten verfügen. Für Fragen und Anregungen oder auch Wünsche zur Mitarbeit steht der Leiter der DOAG Datenbank Community unter all-dbc@doag.org gerne zur Verfügung.

Die SIG Database

Im Februar 2013 fand in München die SIG Database zum Thema „Tuning/Optimierung“ statt. Nach der Begrüßung durch den SIG-Leiter Johannes Ahrends präsentierte Rainier Kaczmarczyk, OPITZ CONSULTING GmbH, den Vortrag „Oracle-Tuning mit Bordwerkzeugen“. Dabei beschrieb er sehr anschaulich, wie man Performance-Engpässe nur mit Datenbank-eigenen Mitteln erkennt und behebt – ohne Oracle-eigene Tools oder Werkzeuge von Drittherstellern. Im Anschluss daran erläuterte Robert Kruzynski von der Trivadis GmbH die „Unterstützung von Tuning-Maßnahmen mithilfe von Capacity Planning“. In diesem Vortrag ging es um die proaktive Erkennung von Engpässen und die daraus resultierenden Schritte, damit es dazu nach Möglichkeit nicht kommt. Der Referent stellte dar, wie man mit Capacity Planning mögliche Probleme frühzeitig erkennen und prophylaktisch eingreifen kann.

Im Vortrag von Johannes Ahrends von der CarajanDB GmbH stand die Frage „Index oder nicht Index“ im Mittelpunkt der Betrachtungen. Es wurde sehr gut ausgeführt, in welchen Fällen ein Index sinnvoll, notwendig oder überflüssig ist. Ulrike Schwinn von der ORACLE Deutschland B.V. & Co. KG beschrieb in ihrem Vortrag „Alles rund um SQL Tuning Sets“ deren Arbeits- und Wirkungsweise. Dabei ging sie auch darauf ein, in welchen Situationen es sinnvoll ist, dieses Werkzeug einzusetzen.

Den Abschluss bildete die Präsentation von Felix Castillo von oraconsult zum Thema „Quo vadis AWR“. Dabei stellte er sehr detailliert das Automatic Workload Repository vor und erklärte die Möglichkeiten, die der DBA mit diesem Tool hat, um Performance-Herausforderungen zu meistern. Es wurden aber auch die Grenzen des Tools aufgezeigt.

Zu allen Vorträgen entspann sich auch in den Pausen eine rege Diskussion zwischen den zahlreichen Teilnehmen und den Referenten. Insgesamt war es eine gut gelungene Veranstaltung, was sich auch in den Rückmeldungen der Teilnehmer widerspiegelt. Die nächste SIG Database findet am Donnerstag, 12. September 2013, in Frankfurt statt. Zuvor bietet jedoch die DOAG 2013 Datenbank am Dienstag, 14. Mai 2013, in Düsseldorf eine eintägige Fachkonferenz rund um alle Oracle-Datenbank-Themen.

Fragen, Anregungen und Wünsche kann man gerne an die Leiter der SIG Database Christian Trieb und Johannes Ahrends unter sig-database@doag.org richten.

Stefan Kinnen
Leiter der Development Community

Neues aus der Development Community

Im Februar 2013 trafen sich die Aktiven der Development Community in Berlin zum „Frühjahrsputz“, um den Status der bisherigen Arbeit zu ermitteln und vor allem auch sinnvolle Änderungen und Erweiterungen zu finden. Mit den beiden anstehenden Fachkonferenzen – DOAG 2013 BI im April in München und DOAG 2013 Development im Juni in Bonn – besteht seitens der Planung große Zufriedenheit. Die Teilnehmer erwarten vielfältige Streams mit spannenden Vorträgen, gepaart mit reichlich Freiraum für Erfahrungsaustausch und Networking.

Nach einem Vergleich der Oracle-Produktpalette mit dem bisherigen Angebot der DOAG wurden Wege geplant, um auch die beiden wichtigen Produkte „ADF“ und „Apex“ künftig besser und sichtbarer in der Development Community zu platzieren. Neben dem reinen Produktfokus sollen vor allem auch technologisch übergreifende Themen adressiert und kommuniziert werden können. Konkret soll nach zwei erfolgreichen SIG-Veranstaltungen das Thema „Mobile Computing“ regelmäßiger und auch in anderen Formaten weiter vertieft

sowie parallel dazu der Erfahrungsaustausch fokussiert werden.

Die Development Community sieht sich als Vorreiter in der Ansprache und Gewinnung jüngerer Mitglieder. Studenten, Ausbildungsabsolventen und „Young Engineers“ sollen künftig mit neuen Veranstaltungsformaten angesprochen werden. Eines davon ist das sogenannte „Barcamp“, vergleichbar mit der „Unconference“ bei der DOAG-Jahreskonferenz.

Zudem gilt es, das Thema „Java Development“ weiter zu intensivieren. Hier basiert die Strategie weiterhin auf der Kooperation mit dem Interessenverbund der Java Usergroups e.V. (iJUG), in dem die DOAG zu den Gründungsmitgliedern zählt. Auch zum Thema „Java“ werden künftig neue Angebote präsentiert, die von dem Netzwerk des iJUG profitieren und gleichzeitig die Stärken der DOAG bei der Organisation erfolgreicher Veranstaltungen und deren vielfältige Kommunikationswege nutzen.

Die Development Community freut sich auf die neuen Vorhaben. Der Vorstand hat schnell grünes Licht gegeben, sodass erste konkrete Planungen und Vorbereitungen bereits gestartet sind. Wir freuen uns immer über Ihre Unterstützung in Form von Feedback, Kritik oder Vorschlägen an ski@doag.org.

Dr. Frank Schönthaler
Leiter der Business Solutions Community

Neues aus der Business Solutions Community

Die Vorbereitungen der DOAG 2013 Applications Konferenz + Ausstellung vom 9. bis 11. Oktober 2013 schreiten zügig voran. Um wieder eine herausragende Veranstaltung auf die Beine stellen zu können, arbeitet die BSC-Leitung gemeinsam mit den internationalen Kooperationspartnern und Oracle auf Hochtouren: Das Rahmenprogramm ist in Planung und die Keynote-Speaker sind angefragt. Der Call for Presentations ist ebenso wie die Ausstelleranmeldung bereits geöffnet und die ersten Ausstellerplätze wurden auch schon gebucht. Alle Informationen rund um die Teilnahme an Europas führender Oracle-Applications-Konferenz finden Sie unter http://www.doag.org/de/events/konferenzen/doag-2013-applications.html.

Die von der BSC angebotenen Web–inare stoßen auf großen Zuspruch. Bereits am Freitag, 8. März 2013, informierte die Primavera Community interessierte Mitglieder zum Thema „Risk Management“ unter Zuhilfenahme der Applikation „Oracle Primavera Risk Management“. Dieses Webinar ging sowohl auf mögliche Einsatzbereiche als auch auf Maßnahmen zum Umgang mit identifizierten Risiken in Projekten und daraus gewonnene Erfahrungen ein. Am 28. März zeigte Thomas Fricke von Oracle im Rahmen der E-Business Suite Community einen Querschnitt des Oracle Report Manager mit dem Schwerpunkt: „Reporting leicht gemacht in Oracle EBS Release 12“. Fazit: Das leider von vielen noch unentdeckte Tool bietet einiges für EBS-Anwender. Weitere Webinare der DOAG Business Solutions Community folgen in regelmäßigen Abständen. Alle Termine stehen unter http://bs.doag.org/de/events-bs/webinar.html. Mitglieder können die Präsentationsunterlagen der Webinare auch bequem herunterladen.

Wir begrüßen unsere neuen Mitglieder

Persönliche Mitglieder

Gabriele Bethge
Jochen Reinartz
Thomas Nötling
Volker Christ
Kirill Loifman
Petra Durow
Marcel Merz
Nico Schwarzbach
Wolfgang Beranek
Peter Lang
Cornel Brücher
Jan Lütke
Giovanni Cisotta
Franz Josef Jobst
Thomas Heinrich
Klaus Igel
Ronny Roth
Rostislav Kushnirenko
Tamme Reinders
Beate Künneke
Detlev Kockel
Andrè Schuster
Konstantin Lavrentyev

Firmenmitglieder

JThomas Heine, Premium AEROTEC GmbH
Thomas Weiß, Josef Witt GmbH
Karsten Lück, Germanischer Llyod SE
Evgeni Ivanov, INTERSHOP Communications GmbH
Markus Bergholz, regiocom GmbH
Jörg Otto, IDS GmbH
Christoph Dörstel, mSE-GmbH
Theodor Vehndel, Oldenburgische Landesbank AG

16.04.2013
Regionaltreffen Hamburg/Nord
Stefan Thielebein
regio-nord@doag.org

16.04.2013
Regionaltreffen Rhein-Main
Thomas Tretter
regio-rhein-main@doag.org

17.04.2013
DOAG 2013 Business Intelligence
Im aktuellen ökonomischen Umfeld gewinnt der Einsatz von effektiven Business Intelligence- und Data Warehouse-Lösungen immer mehr an Wichtigkeit. Die DOAG Business Intelligence Konferenz bietet in diesem Kontext die ideale Plattform für eine intensive Weiterbildung.
Christian Weinberger
office@doag.org

18.04.2013
Regionaltreffen Nürnberg/Franken
Andrè Sept, Martin Klier
regio-franken@doag.org

22.04.2013
Regionaltreffen München/Südbayern
Andreas Ströbel
regio-muenchen@doag.org

23.04.2013
Regionaltreffen Freiburg
Volker Deringer
regio-freiburg@doag.org

23.04.2013
SIG Security
Franz Hüll
sig-security@doag.org

25.04.2013
DOAG 2013 Logistik
Community-Konferenz: Intelligente Prozesse und IT-Systeme
Simone Fischer
office@doag.org

07./08.05.2013
Berliner Expertenseminar:
„Oracle XML" mit Jürgen Sieben
Cornel Albert
expertenseminare@doag.org

07.05.2013
Regionaltreffen Rhein-Neckar
Frank Stöcker
regio-rhein-neckar@doag.org

13.05.2013
Regionaltreffen NRW (Vorabend Datenbank Konferenz)
Stefan Kinnen, Andreas Stephan
regio-nrw@doag.org

14.05.2013
DOAG 2013 Datenbank
Datenbank-Administratoren und technisch Interessierten eröffnet sich die Gelegenheit, einen Tag rund um die klassischen Themen der Oracle Datenbank sowie MySQL und Embedded Database zu erleben.
Christian Trieb
office@doag.org

14.05.2013
6. Primavera Community Day
Alexander Neumann, Sebastian Hunke
bsc-primavera@doag.org

16.05.2013
Regionaltreffen Nürnberg/Franken
Andrè Sept, Martin Klier
regio-franken@doag.org

16.05.2013
Regionaltreffen Stuttgart
Jens-Uwe Petersen
regio-stuttgart@dag.org

28.05.2013
Regionaltreffen München/Südbayern
Andreas Ströbel
regio-muenchen@doag.org

29.05.2013
Regionaltreffen Bremen
Ralf Kölling
regio-bremen@doag.org

06.06.2013
DOAG 2013 IM Community Summit
Community-Konferenz zum Thema Infrastruktur meets Middleware
Björn Bröhl
office@doag.org

07./08.06.2013
1. DOAG Delegiertenversammlung
office@doag.org

10.06.2013
Regionaltreffen Osnabrück/Bielefeld/Münster
Andreas Kother
regio-osnabrueck@doag.org

11./12.06.2013
Berliner Expertenseminar:
„Engineering Oracle for Performance" mit Dr. Günter Unbescheid
Cornel Albert
expertenseminare@doag.org

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