

DOAG SOUG News

Schnell, einfach,

APEX



Ein Insider berichtet

Acht große Oracle-Datenbank-Mythen

Im Interview

Lothar Lux, Mitglied der DATEV-Geschäftsleitung



Aktuell

Die neue Generation der Engineered Systems

Reden Sie mit der Nummer 1,
wenn es um APEX geht!



MT AG



Platin Sponsor
der APEX connect

Besuchen Sie uns auf der APEX connect.

Mehr Infos unter <http://apex.doag.org>



Niels De Bruijn
Verantwortlich für den
Themenbereich Apex

#letswreckthistogether

Liebe Mitglieder, liebe Leserinnen und Leser,

Oracle Application Express geht in die nächste Runde. Dieses kostenlose Add-on der Oracle-Datenbank bietet in der Version 5.0 eine Vielzahl an neuen Möglichkeiten, die den Entwickler noch produktiver werden lassen. Wer hätte vor zehn Jahren gedacht, dass sich eine schicke, gut bedienbare und leicht wartbare Web-Applikation in nur wenigen Minuten realisieren lässt? Noch erstaunlicher ist die Tatsache, dass das Apex-Kernteam aus lediglich zwölf Entwicklern besteht.

Ein gutes Produkt zu haben, ist natürlich eine Sache. Apex bekannt zu machen, eine andere. Hier spielt die Apex-Community eine zentrale Rolle, da sie über die Regionaltreffen, die DOAG Konferenz oder Meet-ups den persönlichen Austausch untereinander fördert. Besonders hervorheben möchte ich die zweitägige Apex-Konferenz „APEX Connect 2015“, die am 9. und 10. Juni 2015 in Düsseldorf stattfindet. Bei dieser Premiere erwartet die Teilnehmer ein Programm mit Top-Rednern aus aller Welt.

Auch im Online-Bereich ist die Apex-Community sehr aktiv. So gibt es mehr als einhundertfünfzig Blogger, zwanzig Bücher, Webinare, den bekannten Twitter-Hashtag „#orclapex“ und eines der populärsten OTN-Foren.

Wer übrigens Apex noch nicht kennt, kann auf „apex.oracle.com“ einen kostenlosen Workspace beantragen und sofort loslegen.

„Let's wreck this together“ (Lasst uns diese Sache gemeinsam anpacken), lautete der Hashtag im Tweet Ende Januar, als Joel Kallman, der Apex-Projektleiter, das Early Adopter Release 3 von Apex 5.0 ankündigte. Recht hat er: Nur gemeinsam mit der Community kann Apex noch erfolgreicher werden!

Viel Spaß beim Lesen der vielen Apex-Artikel wünscht Ihnen Ihr

Niels de Bruijn

ORACLE Gold
Partner

Specialized
Oracle Database

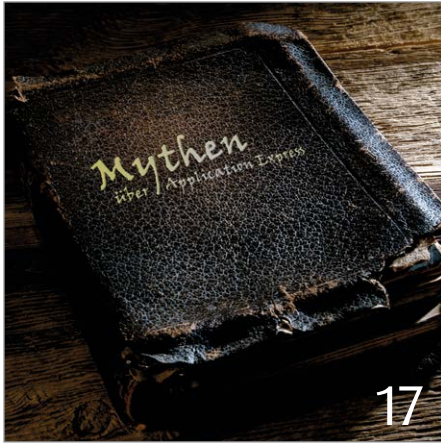
MUNIQSOFT
Datenbanken mit iQ

APEX 5 steht in den Startlöchern, wir auch ...

5 Gründe, die für uns sprechen:

1. 100 durchgeführte APEX-Schulungen
2. 10 Jahre APEX-Erfahrung
3. 300 geschulte Teilnehmer
4. 5 verschiedene APEX-Kurse im Programm
5. Mehr als 5 Oracle APEX-Consultants
in unserem Haus

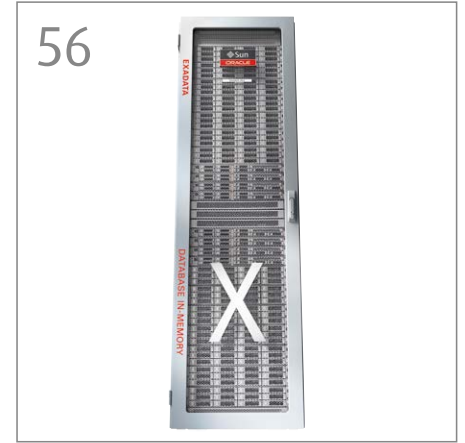
Besuchen Sie uns unter
www.munisoft.de/apex



Seit es Apex gibt, kursieren allerlei Mythen und Legenden darüber. Manche davon sind eher persönliche Vorlieben, andere völliger Unsinn



Die Erfahrungen eines Einsteigers widerlegen den Irrglauben, dass Apex ohne spezielles Wissen genutzt werden kann



Die neue Server-Generation X5-2 mit 2-Sockel-x86-Architektur steht jetzt in den Oracle Engineered Systems zur Verfügung

Einleitung

- 3** Editorial
- 5** DOAG/SOUG Timeline
- 7** Ergebnis der Wahl zur Delegiertenversammlung

Apex

- 8** „Datensicherheit und Datenschutz sind auf höchstem Niveau gewährleistet ...“
Interview mit Lothar Lux
- 11** Das richtige Tool – ein Plädoyer für Apex
Denes Kubicek
- 17** Mythen über Apex
Markus Hohloch und Tobias Arnhold
- 20** Apex 5.0: Anwendungsentwicklung völlig neu definiert
Carsten Czarski
- 25** Enterprise Apex bei Vodafone – eine Erfolgsgeschichte
Tobias Strohmeyer

- 29** Apex im DWH/BI-Umfeld bei der GFKL
Kai Donato
- 34** Erfahrungen eines Apex-Einsteigers
Rüdiger J. Schulz
- 39** D3: Data Driven Documents
Ottmar Gobrecht
- 45** Zehn nützliche Apex-Tipps aus der Praxis
Andreas Wismann
- 52** Apex Community – da ist mehr drin
Carsten Czarski

Aktuell

- 13** Oracle verlängert Support für Forms und Reports 11gR2
- 54** Mobile My Oracle Support
Karl-Heinz Urban
- 56** Eine neue Generation von Engineered Systems
Frank Schneede

Datenbank

- 63** Acht große Oracle-Datenbank-Mythen
Robert Barić
- 68** Oracle Restart: Die Grid Infrastructure für die Single Instance
Dr. Frank Haney
- 72** Tipps & Tricks aus Gerds Fundgrube Heute: Eine optimale Frame-Größe erzeugen
Gerd Volberg

DOAG/SOUG intern

- 73** Neue Mitglieder
- 74** Impressum
- 74** Termine
- 74** Inserentenverzeichnis

✦ DOAG/SOUG Timeline

1. Februar 2015

Zum 1. Februar 2015 übernimmt Frank Obermeier als neuer Country Leader die Verantwortung für das Deutschland-Geschäft von Oracle. Zuletzt war er als Vice President Worldwide Channel Sales für das weltweite Channel-Geschäft der Firmenkunden-Sparte von Hewlett-Packard verantwortlich. In seiner über 20-jährigen Karriere in der IT-Industrie hatte er Leitungsfunktionen unter anderem bei Dell und HP inne. Frank Obermeier berichtet an Jürgen Kunz, der als Senior Vice President Northern Europe die Geschäfte von Oracle in Nordeuropa im europäischen Management Board verantwortet.



Frank Obermeier, Country Leader Oracle Deutschland

5. Februar 2015

In einem Nürnberger Hotel dreht sich wortwörtlich „from Noon to Noon“ alles rund um die kostenfreie Version Express Edition (XE) der Oracle-Datenbank und die bekannteste Open-Source-Datenbank MySQL. Das erstmals von der DOAG durchgeführte, ungewöhnliche Format zielt auf mehr Praxis- und Lösungsorientiertheit. Bei den interaktiven Sessions heißt es: Ärmel hochkrempeln und mitmachen. Die Veranstaltung startet um 12 Uhr und endet um 12 Uhr des Folgetags. Zunächst gibt es Kurzvorträge zu den verschiedensten Themen, wobei die Unterschiede in den Datenbanken im Mittelpunkt stehen. Anschließend werden die Inhalte in einer gemeinsamen Paneldiskussion praxisnah vertieft. Am Abend sind lockeres Networking und Erfahrungsaustausch beim winterlichen Grillbuffet angesagt. Parallel dazu setzen Interessierte die neugewonnenen Erkenntnisse am eigenen Laptop in die Praxis um. Am nächsten Morgen geben die Teilnehmer in den interaktiven Sessions den Ton an und bringen ihre Themen aufs Tapet. Es entwickelt sich nochmals ein reger Austausch über Oracle XE versus MySQL.

9. Februar 2015

Das Organisationskomitee der JavaLand 2015 legt die letzten Aktivitäten für die Java-Konferenz am 24. und 25. März im Phantasieland in Brühl fest. Nachdem bereits in der Voranmeldung die Teilnehmerzahl des Vorjahres übertroffen wird, zeichnet sich ein Besucherrekord ab.

12. Februar 2015

Die ersten DOAG PL/SQL Days finden in München statt. Rund fünfzig Teilnehmer aus ganz Deutschland finden den Weg in die bayrische Landeshauptstadt, um über SQL, PL/SQL und Datenbank-Programmierung zu reden. Obwohl schon mehr als zwanzig Jahre alt, sind diese Themen für viele Oracle-Entwickler nach wie vor hochinteressant. In einem Punkt sind sich alle Dozenten einig: Um die Leistung einer modernen Oracle-Datenbank effektiv zu nutzen, reicht es heutzutage nicht mehr aus, die altbewährten SQL-beziehungsweise PL/SQL-Konstrukte zu verwenden. Sie legen den Teilnehmern ans Herz, eine Reihe von Punkten zu beachten, etwa generell immer die New Features eines neuen Datenbank-Release zu studieren oder auf bewährte Funktionalitäten zurückzugreifen, die vorinstallierte Datenbank-Packages bieten, statt selbst solche Funktionalitäten zu programmieren.

Ein Highlight am Folgetag ist das von Jürgen Sieben vorbereitete Hands-on zum PL/SQL-Instrumentation-Toolkit (PIT). Die Teilnehmer können nicht nur das komplexe PIT live einsetzen, sondern profitieren auch davon, dass Jürgen Sieben zahlreiche architektonische Details von PIT verständlich erläutert. Er hält an seinem Vorhaben fest, PIT in Kürze der Open-Source-Gemeinde zur Verfügung zu stellen.

Damit das Networking nicht zu kurz kommt, gibt es einen Community-Abend beim Augustiner in München. In angenehmer Atmosphäre werden zahlreichen Gesprächen geführt.

Schon während der Veranstaltung melden die Teilnehmer ein positives Feedback. Sowohl die präsentierten Themen als auch das Format des Treffens finden breite Zustimmung. Daher fällt es Christian Schwitalla, Leiter der DOAG PL/SQL Community, nicht schwer, eine Fortsetzung im Jahr 2016 anzukündigen.

24. Februar 2015

In einem ersten Telefonat mit Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG, signalisiert Frank Obermeier, der neue Oracle Country Leader für Deutschland, dass er an einer engeren Zusammenarbeit mit der DOAG sehr interessiert ist. Beide vereinbaren, sich regelmäßig alle drei Monate persönlich auszutauschen. Gleichzeitig reden sie über wichtige Themen, die derzeit die DOAG-Mitglieder besonders bewegen, etwa die VMware-Lizenzierung sowie der Stellenabbau im Oracle Support in Deutschland. Frank Obermeier nimmt auch eine Einladung auf die DOAG 2015 Konferenz + Ausstellung in Nürnberg an. Abschließend wird ein Interview in einer der nächsten Ausgaben der DOAG News vereinbart. Dr. Dietmar Neugebauer ist zuversichtlich, dass mit Frank Obermeier wieder eine intensivere Zusammenarbeit mit Oracle auf Geschäftsleitungsebene entsteht.

24. Februar 2015

Die Leitung der Business Solutions Community stellt in einer Telefonkonferenz die Weichen für die DOAG 2015 Business Solutions

Konferenz, die in diesem Jahr vom 9. bis 11. Juni in Darmstadt stattfindet. Die Metropolregion Frankfurt/Rhein-Main, insbesondere die Wissenschaftsstadt Darmstadt und das architektonisch beeindruckende Darmstadtium, eignen sich hervorragend für diese Veranstaltung. Die Vorbereitungen laufen bereits auf Hochtouren; das Top-Programm mit den neuesten Themen rund um das ganze Spektrum der Oracle-Business-Applikationen steht fest. An den beiden Konferenztagen werden wieder viele praxisnahe und anwenderorientierte Vorträge präsentiert. Im Anschluss daran findet ein Schulungstag statt, der für gezielte und fundierte Wissensvermittlung garantiert. Zudem können auch in diesem Jahr wieder namhafte Aussteller gewonnen werden, die mit ihren Exponaten die ideale Ergänzung zum Konferenzprogramm darstellen.

Dr. Frank Schönthaler informiert die Runde über einige personelle Veränderungen in der Community. Bereits im vergangenen Jahr wurde Michael Baranowski als Themenverantwortlicher für Logistik und Supply Chain Management ernannt. Er ist vielen in der Community seit längerem bekannt, da er seit 2008 im Programm- und Planungskomitee der DOAG Fachkonferenz Logistik + IT aktiv ist. Beruflich ist er seit über 20 Jahren Geschäftsführer der TEAM GmbH in Paderborn, die als Oracle Platinum Partner ein eigenes Oracle-basiertes Produkt für die Intralogistik anbietet.

Eugen Visinescu ist ebenfalls seit vielen Jahren aktives Mitglied der DOAG und seit diesem Jahr zusammen mit Detlef Kubusch für die Themen Oracle CRM und Customer Experience (CX) zuständig. Außerhalb der DOAG ist Eugen Visinescu Teamleiter Entwicklung für CRM im Konzern der Deutschen Telekom AG, T-Systems.

In der Primavera Community hat sich ebenfalls ein Wechsel vollzogen. Alexander Neumann musste aufgrund seiner hohen beruflichen Belastung sein Amt als Themenverantwortlicher niederlegen. Sebastian Schweinle ist nun – zusammen mit Sebastian Hunke – Themenverantwortlicher für Primavera. Herr Schweinle verfügt über mehr als zehn Jahre Erfahrung im Bereich IT-Beratung in verschiedenen Branchen und wechselte im Jahr 2011 zur Siemens AG. Dort ist er im Bereich „Energy Management“ als Projektleiter für die Primavera-Implementierung zuständig.

Dr. Frank Schönthaler heißt die neuen Verantwortlichen herzlich in der BSC-Leitung willkommen und wünscht einen guten Start und gutes Gelingen. Es bedankt sich bei Alexander Neumann für sein Engagement und die jederzeit gute Zusammenarbeit sowie für seinen persönlichen Einsatz zum Wohl der Primavera Community.



Das Motiv der DOAG 2015 BSK

25. Februar 2015

In Frankfurt findet das Kick-off-Meeting der neuen Java Community der DOAG statt. Neben ersten Planungen gilt es, einen geeigneten Kandidaten für das Amt des Vorsitzenden zu finden, der dann die Community im DOAG-Vorstand vertritt. Mit Markus Eisele, André Sept und Andreas Badelt bekunden drei Kandidaten ihr Interesse dafür. Sie einigen sich in einer nachfolgenden Telko auf Markus Eisele.

26. Februar 2015

In Baden Dättwil findet die erste Vorstandssitzung der SOUG im Jahr 2015 statt. Ein Highlight ist die Begrüßung der neuen Vorstandsmitglieder Patrick Schäfer und Jürgen Vitek. Zur französischen Beilage für die Romandie der DOAG/SOUG News gibt es viel positives Feedback. Der Prozess für die Erstellung der Beilage und der Versand hat noch Verbesserungspotenzial, aber grundsätzlich ist der Schritt mit der Zusammenlegung ein Erfolg. Neue strategischen Themen stehen bei der nächsten Vorstandssitzung am 12. Mai 2015 auf der Tagesordnung.



4. März 2015

In der Berliner DOAG-Geschäftsstelle treffen sich Günther Stürner, Vice President Server Technologies und Sales Consulting bei Oracle, und Duncan Harvey, Director Business Development Technology Oracle EMEA, mit Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG, und Fried Saacke, DOAG-Vorstand und Geschäftsführer, sowie den DOAG-Lizenz-Experten Michael Paege und Patrick Wilke. Thema ist die nach wie vor unbefriedigende Lösung bei der Lizenzierung von Virtualisierungssoftware. Die Oracle-Vertreter sagen zu, das Thema stärker zu kommunizieren. Die DOAG ist mit dem Ergebnis des Gesprächs nicht zufrieden und startet eine Umfrage unter den Anwendern, um die Wichtigkeit der Lizenzierungsproblematik zu untermauern.

5. März 2015

Das Kick-off-Meeting der neuen BI Community findet in Frankfurt statt. Es treffen sich alle, die sich bisher in der DOAG zu BI-Themen engagiert haben, oder bei der Delegiertenwahl für die BI Community kandidiert hatten. Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG, und Fried Saacke, DOAG-Vorstand und Geschäftsführer, möchten mit der Besprechung sicherstellen, dass unabhängig von der noch anstehenden Wahl des Vorstands für die BI-Community die Zeit schon jetzt genutzt wird. Alfred Schlaucher, für Oracle im Bereich „Data Warehouse“ aktiv, hat sein großes Interesse an einer engen Zusammenarbeit und seine Unterstützung und Mitarbeit bekundet. Als Vorstand für die BI Community wird Michael Klose vorgeschlagen, der die Kandidatur gerne übernimmt. Weiterer Programmpunkt ist die DOAG 2015 BI. Für die am 23. April in München stattfindende Konferenz wird mit rund 100 Teilnehmern gerechnet.

**VOTE**

Ergebnis der Wahl zur Delegiertenversammlung

Der DOAG-Wahlausschuss, vertreten durch Oliver Pyka, Andreas Ellerhoff und Reiner Büniger, stellt das endgültige Ergebnis der Wahl zur Delegiertenversammlung fest. Abgestimmt wurde im elektronischen Verfahren zwischen dem 19. und 30. Januar 2015. Alle 47 Delegiertenämter sind besetzt. Der DOAG-Vorstand bedankt sich bei allen Kandidaten, dem Wahlausschuss und allen, die sich an der Wahl beteiligt haben.

Die Delegierten werden aus dem Kreis der Mitglieder für eine Amtszeit von zwei Jahren gewählt. Ein Doppelamt ist nicht vorgesehen, im Falle einer Mehrfachkandidatur oder einer Absage rücken die nächstplatzierten Kandidaten nach. Neben den 47 gewählten Delegierten besteht die Delegiertenversammlung aus den 18 regionalen Repräsentanten. Diese sind Kraft ihres Amtes Mitglieder der Delegiertenversammlung. Gewählt sind:

Gruppe der persönlichen Mitglieder

- Dr. Dietmar Neugebauer
- Andrea Held
- Andreas Buckenhofer
- Gunther Pippèr
- Gerhild Aselmeyer

Gruppe der studentischen Mitglieder

- Svenja Schriever
- Marcel Schönthaler

Gruppe der korporativen Mitglieder mit bis zu 500 Mitarbeitern

- Michael Paege
- Henriette Cebulla
- Stefan Kinnen
- Franz Hüll
- Arne Hattendorf

Gruppe der korporativen Mitglieder mit mehr als 500 Mitarbeitern

- Kathleen Hock
- Dagmar Förster
- Claudia Gedamke
- Karl Forsterpointner
- Carsten Kaftan

Gruppe der Business Solutions Community

- Dr. Frank Schönthaler
- Sebastian Hunke
- Dirk Blaurock
- Kasi Färcher-Haag
- Dennis Giese

Gruppe der Datenbank Community

- Christian Trieb
- Johannes Ahrends
- Markus Flechtner
- Robert Marz
- Jürgen Vitek

Gruppe der Development Community

- Sabine Heimsath
- Robert Szilinski
- Niels de Bruijn
- Christian Schwitalla
- Gerd Volberg

Gruppe der Infrastruktur & Middleware Community

- Johannes Michler
- Stefan Panek
- Frank Hoppe
- Markus Lohn
- Reinhold Boettcher

Gruppe der Java Community

- Tina Bauspieß
- Andreas Badelt
- Rolf Scheuch
- Markus Eisele
- André Sept

Gruppe der Business Intelligence Community

- Peter Welker
- Sven Bosinger
- Michael Klose
- Ingo Frost
- Dr. Jens Bleiholder

Regionale Repräsentanten

- Michel Keemers, Regio Berlin / Brandenburg
- Ralf Kölling, Regio Bremen
- Matthias Reimann, Regio Leipzig / Halle
- Oliver Pyka, Regio Würzburg
- Andreas Stephan, Regio NRW
- Bernd Tuba, Regio Trier / Saarland / Luxemburg
- Jens-Uwe Petersen, Regio Stuttgart
- Thomas Tretter, Regio Rhein-Main
- Volker Deringer, Regio Freiburg / Südbaden
- Helmut Marten, Regio Dresden / Sachsen
- Andreas Kother, Regio Osnabrück / Bielefeld / Münster
- Martin Klier, Regio Nürnberg / Franken
- Frank Stöcker, Regio Rhein-Neckar
- Andreas Ströbel, Regio München / Südbayern
- Reiner Büniger, Regio Karlsruhe
- Jörg Hildebrandt, Regio Jena / Thüringen
- Jan-Peter Timmermann, Regio Hamburg / Nord
- Andreas Ellerhoff, Regio Hannover

„Datensicherheit und Datenschutz sind auf höchstem Niveau gewährleistet ...“

Bei der DATEV laufen mehrere Petabyte an Daten zusammen und werden dort verarbeitet und archiviert. Wolfgang Taschner, Chefredakteur der DOAG News, sprach darüber mit Lothar Lux, Geschäftsleitungsmitglied der DATEV.



Lothar Lux (links) im Gespräch mit Wolfgang Taschner

Was ist das Geschäftsmodell der DATEV?

Lux: DATEV versorgt Steuerberatungskanzleien und Unternehmen mit Software und IT-Dienstleistungen. Unser Leistungsspektrum umfasst vor allem Rechnungswesen, Personalwirtschaft, betriebswirtschaftliche Beratung, Steuern, Enterprise Resource Planning, IT-Sicherheit sowie Weiterbildung und Consulting. Mit unseren Lösungen verbessern wir gemeinsam mit unseren Mitgliedern – das sind Steuerberater, Wirtschaftsprüfer und Rechtsanwälte – die betriebswirtschaftlichen Prozesse von zweieinhalb Millionen Unternehmen, Kommunen, Vereinen und Institutionen.

Was sind dabei die größten Herausforderungen an die IT und wie haben Sie diese gelöst?

Lux: Das Leistungsspektrum der DATEV reicht von mehr als zweihundert PC-Programmen über Cloud-Dienste wie Online-

Anwendungen, Datenverarbeitung und Archivierung im Rechenzentrum bis hin zu Outsourcing-Leistungen sowie Sicherheitsdienstleistungen. Unser Rechenzentrum dient außerdem als zentrale Datendrehscheibe, die sensible Informationen zwischen mittelständischen Unternehmen und deren Steuerberatern sowie rund 200 Institutionen in Deutschland austauscht – darunter Finanzverwaltungen, Sozialversicherungsträger, Krankenkassen, Banken, Berufsgenossenschaften oder statistische Ämter. Von elementarer Wichtigkeit sind daher für uns Datenschutz und Datensicherheit. Diese permanent zu gewährleisten, ist für uns die größte Herausforderung.

Welchen Umfang hat Ihre gesamte IT?

Lux: Das Rechenzentrum besteht aus vier IBM-Großrechnern, 1.167 Unix- und 5.704 Windows-Servern. Allein die Großrech-

ner kommen auf eine Gesamtleistung von 40.435 MIPS, das sind mehr als 40 Milliarden Operationen pro Sekunde. Auch wenn solch eine Zahl letztlich nur Experten etwas sagt, zeigt sie doch, dass hier gewaltige Kapazitäten am Werke sind. Es gibt 5,6 Petabyte Festplattenspeicher – synchron gespiegelt wohl-gemerkt – sowie ein Kassettenarchiv mit 15,1 Petabyte archi-vierten Daten. Auf eine handelsübliche CD passen etwa 650 Megabyte. Um den gesamten Festplattenspeicher auf CDs zu sichern, bräuchte man rund 8,6 Millionen CDs sowie eine ge-waltige Spindel zur Aufbewahrung, denn der Turm hätte mit 12.800 Metern eine Höhe, die den Mount Everest deutlich über-ragen würde. Apropos CD: Es gehören auch 31 CD- und DVD-Fertigungssysteme zur Ausstattung, die für die Bereitstellung der DATEV-Software bei Mitgliedern, Kanzleien und Mandan-ten sorgen. Darüber hinaus ist das Rechenzentrum auch direkt mit dem Druck- und Versandzentrum verbunden, das – wie der Name sagt – den Druck und Versand der übermittelten Daten übernimmt. Das sind 10 Millionen Lohn- und Gehaltsabrech-nungen, die jeden Monat ohne jede Fehlertoleranz ihre Emp-fänger erreichen müssen. Dafür stehen 48 Druckmaschinen bereit. Hinzu kommen Versand-Anlagen wie ein automatisches Versandlager, Fördertechnik und Maschinen zum Kommissio-nieren, Kuvertieren, Banderolieren, Verpacken und Freimachen – ein gewaltiger technischer Aufwand im Dienste der Berufsträ-ger und damit letztlich im Dienste von uns allen.

In welchem Umfang können Sie hier Standard-Software einsetzen?

Lux: Unsere gesamte DATEV-Software ist selbst entwickelt. Dafür sind derzeit rund 1.400 Entwickler im Einsatz. Unser Geschäfts-modell und unsere Anforderungen sind so speziell, dass sich der Einsatz von Standard-Software nicht anbietet.

Wie stellen Sie die Sicherheit der bei Ihnen abgelegten sensiblen Daten sicher?

Lux: Für uns stehen zwei Kernfragen im Vordergrund. Einmal, wie Externe an unsere Daten kommen können. Dafür sind wir mit Maßnahmen wie mehrstufigen Firewalls und der Abtrennung der internen Netze sehr gut aufgestellt. Die andere Seite ist der so-genannte „Feind von innen“, wo wir mit ausgefeilten, Rollen-basier-ten Zugriffsrechten unsere Daten schützen. Ich persönlich würde beispielsweise gar nicht an sensible Daten herankommen. Zudem ist der Zugang zum Rechenzentrum durch mehrere Schleusen geschützt, die jeweils eine entsprechende Zugangsberechtigung erfordern. Darüber hinaus praktizieren wir das „Clean Desk“-Ver-fahren, bei dem jeder Mitarbeiter zum Feierabend seinen Schreib-tisch leer räumt und die Arbeitspapiere wegschließt.

Wie sichern Sie die Datenübertragung zu Ihren Kunden ab?

Lux: Die DATEV machte mit der elektronischen Datenfernüber-tragung bereits im Jahr 1974 von sich reden. Dieser Tradition verpflichtet, haben wir die Technologie kontinuierlich auf dem neuesten Stand gehalten. Die Internet-Verbindung zu den Steuer-beratern, Wirtschaftsprüfern und Rechtsanwälten ist heu-te über VPN abgesichert; hinzu kommt eine Hardware-basierte Zwei-Faktor-Authentifizierung. Das bedeutet, der Anwender be-nötigt immer die Besitz-Komponente, bei uns eine Smartcard, und eine Wissens-Komponente, also das gültige Passwort.

Wie stellen Sie die Hochverfügbarkeit der bei Ihnen abgelegten Daten sicher?

Lux: Wir betreiben unsere Rechenzentren an vier verschiedenen Standorten. Jeweils zwei davon sichern die anderen beiden als Backup ab und sind in der Lage, die gesamte anfallende Last zu fahren. Dies wird auch immer wieder getestet und geübt.

Wie bereiten Sie sich auf Spitzenlasten vor, beispielsweise wenn alle Steuerberater gleichzeitig die Steuervoranmeldung machen möchten?

Lux: Die Kapazitäten unseres Rechenzentrums sind auf Spitzen-last ausgelegt, darüber hinaus gibt es bei unseren IBM-Main-frames einen On-Demand-Ansatz. Dieser bietet die Möglichkeit, bei ungewöhnlichen Spitzenlasten gegen eine entsprechende Li-zenzgebühr weitere CPUs hinzuzuschalten.

Im Grunde bieten Sie bereits seit vielen Jahren das an, was heu-te unter dem Begriff „Cloud“ als modern gilt. Wie ist Ihr Modell aufgebaut?

Lux: Obwohl damals noch niemand von einer Cloud geredet hat, haben wir bereits Mitte der 1970er Jahre unseren Mitglie-dern Rechenleistung per Datenfernübertragung angeboten. Heute pflegen wir verschiedene Cloud-Ansätze. Einer unserer Kernprozesse, die Lohn-Applikation, läuft auf dem Mainframe. Die Steuerberater liefern ihre Daten per Internet und erhalten zeitnah das Ergebnis; also eine Quasi-Cloud. Für andere Auf-



Zur Person: Lothar Lux

Lothar Lux leitet bei der DATEV eG als Mitglied der Ge-schäftsleitung den Bereich Produktion, der die Unter-nehmens-IT, das Rechenzentrum sowie das Druck-, Ver-sand- und Logistikzentrum umfasst. Er studierte an der Gesamthochschule Siegen Elektrotechnik und an der Friedrich Alexander Universität Erlangen Nürnberg Be-triebswirtschaftslehre. Nach dem Studium startete er sei-ne Karriere bei Siemens in München. Bereits kurze Zeit später wechselte er zur DATEV in Nürnberg. Dort war er zunächst verantwortlich für den Bereich Kommunikati-onstechnik, bevor er im Jahre 2003 die Gesamtverantwortung für den Produktionsbereich übernommen hat.



gaben gibt es zum einen den Application-Service-Providing-Ansatz, bei dem die Server hier bei uns stehen und die Steuerberater beispielsweise per DSL auf ihre Systeme zugreifen. Dieser Ansatz ist sehr flexibel und skalierbar. Das IT-Management der Software sowie der dafür nötigen IT-Infrastruktur wird von uns erledigt, inklusive Server- und Netz-Monitoring, der Wartung und Administration der Server sowie des Einspielens der Software-Updates und der Datensicherung. Rund 1.100 Kunden nutzen inzwischen diese umfassende Outsourcing-Lösung, die als „Private Cloud“ bezeichnet werden kann. Zum anderen gibt es noch die DATEV-Cloud-Software, über die ein Steuerberater auf standardisierte Funktionen zugreifen kann. Darüber hinaus bieten wir noch zahlreiche Browser-basierte Cloud-Lösungen an. Den Wachstumstrend in Richtung „Cloud“ kann ich nur bestätigen. Mehr als 95.000 kleine und mittelständische Unternehmen nutzen beispielsweise inzwischen gemeinsam mit ihren steuerlichen Beratern die Möglichkeit, ihre Buchführungsbelege über das DATEV-Rechenzentrum in digitaler Form zentral zu verwalten und zu bearbeiten. Digitalisierte Bilder der Originaldokumente werden dabei sicher im Rechenzentrum archiviert, wo sie sowohl der Kanzlei als auch dem Unternehmen jederzeit zur Verfügung stehen.

Was unterscheidet Ihre Cloud-Lösung von anderen kommerziellen Cloud-Lösungen?

Lux: Alle Daten sind hier vor Ort abgespeichert und verlassen Deutschland nicht. Darüber hinaus sind bei uns Datensicherheit und Datenschutz auf höchstem Niveau gewährleistet.

In welchem Rahmen setzen Sie Produkte der Firma Oracle ein?

Lux: Wir benutzen verschiedene Oracle-Datenbanken der Version 11g im File-System, im Data Warehouse und in unserem Online-Portal. Rechner aus der Oracle-Sun-Reihe kommen bei uns nicht mehr zum Einsatz. Zurzeit wird die Datenbank-Version 12c getestet, der produktive Einsatz ist noch für das Jahr 2015 geplant. Die

Oracle Engineered Systems sind momentan kein Thema für uns, da sie nicht in unsere homogene IT-Plattform passen.

In welchem Bereich sehen Sie die Stärken der Oracle-Produkte, wo die Schwächen?

Lux: Die Datenbank ist innovativ, stabil und für uns einfach zu administrieren. Die Business-Applikationen von Oracle hingegen können unsere speziellen Anforderungen nicht erfüllen.

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

Lux: Wir beobachten natürlich alle neuen Trends und Ansätze. Dabei geht es darum, für alle drei Plattformen „Mainframe“, „Unix“ und „Office“ die optimale Lösung zu haben. Dies ist insofern interessant, da sich die drei Hersteller IBM, HP und Microsoft neuerdings auf das Gebiet der jeweiligen anderen begeben.

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

Lux: Oracle sollte weiterhin zuverlässige Datenbanken entwickeln und vor allem für uns besser passende Lizenzmodelle anbieten.

Wie beurteilen Sie den Stellenwert einer Anwendergruppe wie der DOAG?

Lux: Viele unserer Mitarbeiter sehen in der DOAG eine hervorragende Plattform für den Erfahrungsaustausch und besuchen gerne die Veranstaltungen.

Die DATEV

Die DATEV eG ist das Softwarehaus und der IT-Dienstleister für Steuerberater, Wirtschaftsprüfer und Rechtsanwälte sowie deren zumeist mittelständische Mandanten. Das Unternehmen zählt zu den größten Informationsdienstleistern und Softwarehäusern in Europa. So belegt es zum Beispiel Platz 3 im bekannten Lünendonk-Ranking der deutschen Softwarehäuser. Im Geschäftsjahr 2013 (1. Januar bis 31. Dezember) war die DATEV trotz schwächerer Marktentwicklung kräftig gewachsen. Mit einem Umsatzplus von 5,7 Prozent wurde erneut das Wachstum der IT-Branche (1,5 Prozent) sowie der deutschen Gesamtwirtschaft (0,4 Prozent) deutlich übertroffen und mit einem Umsatzvolumen von 803 Millionen Euro (Vorjahr 760 Millionen Euro) erstmals die Marke von 800 Millionen Euro übersprungen.

"Das Wachstum wird von nahezu allen Produktgruppen getragen. Insbesondere bei sicheren Rechenzentrumsdienstleistungen konnten wir eine deutlich gestiegene Nachfrage verzeichnen. Dies zeigt, dass wir mit unserem Fokus auf die Digitalisierung betriebswirtschaftlicher Prozesse mit besonderem Augenmerk auf Vertrauen und Sicherheit richtig liegen", so Prof. Dieter Kempf, Vorstandsvorsitzender der DATEV eG.

Das richtige Tool – ein Plädoyer für Apex

Denes Kubicek

Als der Autor Ende 2004 mit Apex angefangen hat, war das für ihn eine wirkliche Entdeckung. In der Zeit davor war alles viel zu umständlich. Applikationen konnte man nur mithilfe von Forms erstellen und das hat eine richtige Infrastruktur erfordert. Man musste einen Application Server installieren. Auf der Client-Seite waren Applets erforderlich und für das einfache Berichten musste man sich mit Oracle Discoverer oder mit Reports auskennen.

Apex hat neue Möglichkeiten aufgezeigt. Ganz neu war Apex jedoch nicht. Der Vorgänger „WEB DB“ hat schon gezeigt, in welche Richtung es mit Anwendungsentwicklung in Zukunft gehen wird: Entwicklung und Deployment ausschließlich im Browser; keine aufwändige Installation und keine Add-ons. Schlank und schnell. Leider wurde die Weiterentwicklung von WEB DB im Jahr 2000 gestoppt und danach herrschte einige Jahre Stille. Mit der Version 1.4 lebte Apex wieder auf und begann seinen Platz in der Oracle-Community einzunehmen.

Wie bei allen anderen Sachen gibt es auch bei Apex zwei Lager – Befürworter und Skeptiker. Die Befürworter waren immer damit beschäftigt, die guten Eigenschaften von Apex in den Vordergrund zu stellen. Im Fokus stand immer das, was Apex kann und eines Tages können wird. Die Skeptiker dagegen beschäftigten sich mit dem, was Apex nicht kann. Dadurch sind viele Vorurteile entstanden, mit denen man auch heute noch kämpfen muss.

Vorurteile und Fakten

In Bezug auf Apex gibt es einige Vorurteile. Diese sind in den Zeiten entstanden, als Apex noch relativ unbekannt war. Auch Oracle selbst wusste das Potenzial von Apex nicht vom Anfang an richtig einzuschätzen. Es war als ein Ersatz für Microsoft Access gedacht und somit unsichtbar in eine kleine Nische gedrängt. Dadurch ist auch das Vorurteil entstanden, Apex sei nur für einen bestimmten Typ von Anwendungen geeignet. Die Frage des „Geeignet sein“ begleitet Apex heute noch. Fragen wie „Ist Apex für das Programmieren der großen Anwendungen geeignet?“ sind immer wieder zu hören.

Die Antwort „Ja, durchaus“ ist nicht einfach eine pauschale Einschätzung, sondern basiert auf Fakten. Der Autor selbst war an mindestens einem Dutzend solcher Anwendungen beteiligt. Es ging hier unter anderem um drei DAX-Unternehmen, die sich für Apex als Entwicklungsplattform entschieden haben und bei denen er die Gelegenheit hatte, an solchen großen Anwendungen mitzuarbeiten.

Die Befürchtung, die Anwendung gehe in die Knie, wenn sich mehr als eine Handvoll Benutzer einloggt, stammt noch aus den Zeiten von Microsoft Access. Dieses Programm war nicht unbedingt für ein Netzwerk gedacht und hatte genau diese Probleme. Der Vergleich mit Access hat dazu geführt, dass diese Zweifel auch bei Apex aufkommen. Apex kennt jedoch keine Beschränkung der Benutzerzahl,

höchstens ein Limit in der Default-Einstellung des aktuellen Webserver-Dienstes. Apex selbst kommt mit einer großen Anzahl von Benutzern klar. Der beste Beweis dafür ist auf „<https://apex.oracle.com>“ zu finden. Dort wird ein kleiner Server mit bescheidenen Ressourcen eingesetzt, um mehr als 12.000 Workspaces und etwa sechs Millionen Seitenaufrufe im Monat zu bedienen.

Auch die Frage „Wird Apex von Oracle unterstützt?“ lässt sich mit „Ja“ beantworten. Apex ist ein Datenbank-Feature und wird wie alle anderen Bausteine voll unterstützt. Es wird regelmäßig gewartet und mit Patches beliefert; die Updates erscheinen in der Regel alle eineinhalb Jahre. Alle drei Jahre gibt es ein Major Release. Nicht ohne Grund steht auf der offiziellen Apex-Seite: „Oracle Application

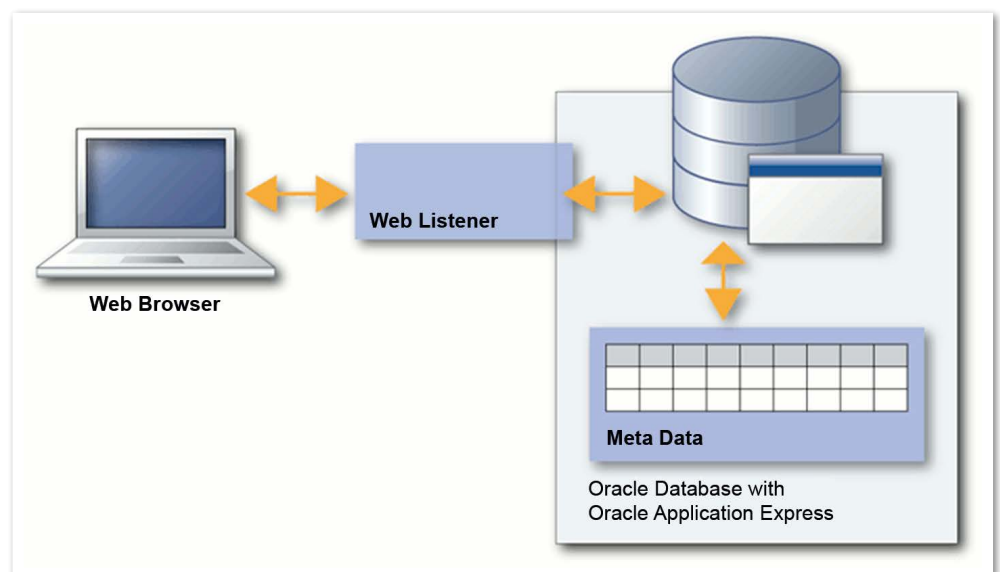


Abbildung 1: Die Apex-Architektur

Express (Apex) is Oracle's primary tool for developing Web applications with SQL and PL/SQL. Using only a web browser, you can develop and deploy professional Web-based applications for desktops and mobile devices."

Dass Apex in Zukunft kostenpflichtig wird, glaubt der Autor nicht. Wobei Apex nicht kostenlos ist; es ist ein Bestandteil der Datenbank und diese kostet Geld. Apex kam als kostenloses Feature dazu. Es ist höchst unwahrscheinlich, dass Apex eines Tages von Oracle separat berechnet wird.

Mancher Entwickler stellt auch die Frage: „Kann ich Apex in Subversion einchecken und jederzeit wieder auf den alten Stand bringen?“ Das geht, aber wozu? Es geschieht sehr häufig, dass Apex mit Entwicklungsumgebungen verglichen wird, die aus dem Java-Umfeld stammen. Sehr häufig wird Apex auf dieser Basis beurteilt – funktioniert es denn genauso wie bei Java? Falls nicht, dann ist das keine gute Idee. Diese Einstellung beruht auf dem mangelnden Wissen über das Konzept, die Architektur und die Technologie. Apex ist kein externes Tool, sondern Bestandteil der Datenbank, das dort verwaltet und gesichert wird. Apex wird zentral und nicht

dezentral entwickelt. Das ist seine Stärke und natürlich, wenn man das dezentrale Konzept mag, auch seine Schwäche.

Eine Technologie, die begeistert

Das Konzept von Apex ist sehr einfach und begeistert deswegen umso mehr. Apex ist Bestandteil der Datenbank und wird dort verwaltet. Wenn man keinen weiteren Webserver-Dienst installieren möchte, übernimmt die Datenbank mit dem „Embedded PL/SQL Gateway“ auch diese Rolle. Das Ganze ist sehr schnell installiert und sofort startbereit. Man kann nur mithilfe eines Browsers die ganze Arbeit erledigen. Auch die Benutzung einer Anwendung geschieht im Browser (siehe *Abbildung 1*). Es sind keine weiteren Tools und Add-ons zwingend notwendig – im Gegensatz dazu, was man aus den anderen Frameworks kennt.

Für diejenigen, die keine Vollversion der Oracle-Datenbank haben, steht die kostenlose Version XE11 zur Verfügung (*Download siehe „<http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html>“*). Diese Version lässt sich ohne zusätzliche Konfiguration auf jedem Rechner installieren und

bietet einen Schnellstart in die Apex-Welt. Nachdem die Software installiert wurde, genügt ein Link-Aufruf im Browser und man steckt sofort mittendrin.

Vor nicht allzu langer Zeit hat der Autor ein Artikel darüber gelesen, für wen Apex als Entwicklungsplattform geeignet ist. Man spottete über Apex und sagte, es sei am besten für sogenannte „Halbwisende“ geeignet und die meisten, die sich damit beschäftigen, gehörten zu dieser Gruppe. Es mag sein, dass dem so ist. Das ist aber auch das Schönste an Apex. Es erfordert keine langjährige Studienzeit, bevor man überhaupt eine Seite in einer Anwendung erstellen kann. Dem Autor fällt dabei als Beispiel die Anleitung im „Oracle Magazine“ zur Erstellung der Wertelisten für ADF ein. Es waren etwa vier komplette Seiten an Text notwendig, um zu erklären, wie man eine Werteliste erstellen und in den Anwendungen bereitstellen kann. Das Erstellen von Wertelisten in Apex ist dagegen ein Selbstläufer. Wie für die meisten anderen Features braucht man dafür keine separate Anleitung.

Mit Apex kann man seine Vorstellungen sehr schnell verwirklichen. Man kann es jemandem geben, der über kein

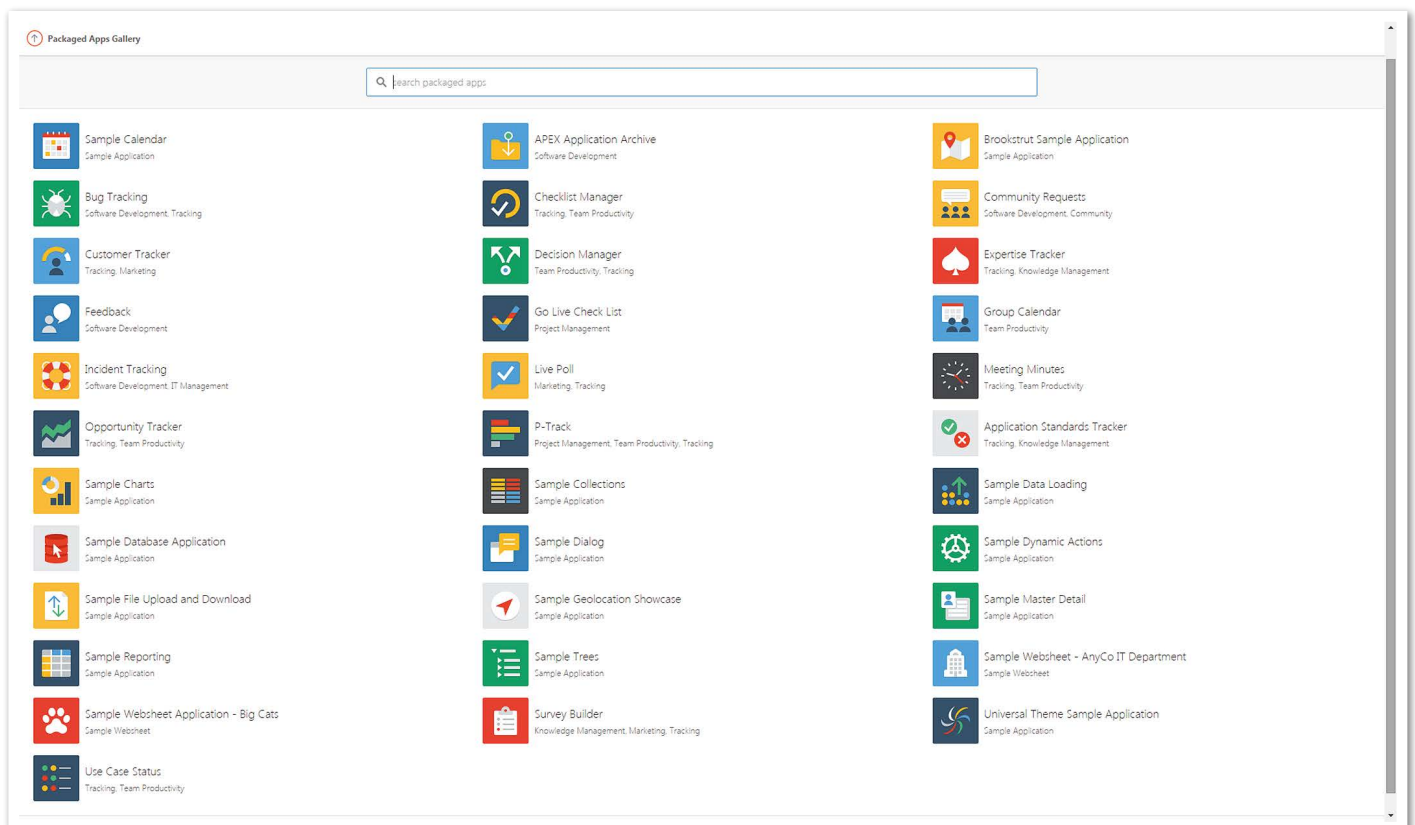


Abbildung 2: Apex Packaged Applications

Oracle-Wissen verfügt, oder jemandem, der schon sehr viel über SQL und PL/SQL weiß. Die Ergebnisse werden gleich sein. Beide können in wenigen Stunden eine brauchbare Anwendung auf die Beine stellen. Apex kann auch als Mockup-Tool sehr gut eingesetzt werden. Wer bereit ist, etwas Neues kennenzulernen, keine Vorurteile hat und nichts von vornherein ausschließt, ist für Apex geeignet und sollte es unbedingt ausprobieren.

Packaged Applications

Packaged Applications sind ein sehr nützliches Apex-Feature. Mit jeder neuen Version wird es verbessert und ausgebaut. Packaged Applications sind verschiedene Anwendungen, die sofort und uneingeschränkt zur Verfügung stehen. Mit einigen Klicks kann man eine Anwendung installieren. Dabei werden auch alle dafür notwendigen Objekte und Daten installiert (siehe Abbildung 2).

Der Code und die Lösung eines Problems können direkt betrachtet und kom-

plett oder zum Teil wiederverwendet werden. Die Anwendungen sind thematisch unterschiedlich und decken eine ganze Reihe verschiedener Lösungen ab. In der derzeitigen Version 4.2.0.6 sind es insgesamt vierzig Anwendungen. In der neuen Version 5.0 wird das konsolidiert und es kommen neue Themen dazu.

Die wichtigsten Apex-Features

Oracle Apex stellt alles bereit, was man für eine gute Web-Anwendung benötigt. Die wichtigsten Features sind:

- Entwicklung und Deployment von Desktop- und mobilen Anwendungen ausschließlich im Browser
- jQuery-Support als Standard
- Die Möglichkeit, alle Datenbank-Features in Apex einzusetzen (Spatial, Oracle Text, File Handling, Emailing, XML, JSON etc.)
- Apex ist genauso sicher wie die Datenbank selbst
- Authentifizierungs- und Autorisierungsmöglichkeiten können an alle be-

liebigen Systeme angebunden werden (SSO, LDAP, Active Directory)

- SQL-Abfragen sind viel schneller als mit anderen Tools, weil sie direkt in der Datenbank ausgeführt und durch Optimizer verwaltet werden
- Plug-ins und Dynamic Actions bieten die Möglichkeit, externe Ressourcen (JavaScript Libraries) so in die Anwendung einzubinden, als wären sie eine Standard-Komponente
- Interactive Reports sind eine sehr gute Möglichkeit, Excel-ähnliche Mittel bereitzustellen, mit denen Benutzer die Datenbank-Tabellen nach dem Excel-Prinzip abfragen können. Die neuen Interactive Reports in der Version 5.0 bieten auch die Erstellung von Pivot-Ansichten im Standard
- Die Erstellung von Charts „on the fly“ ist als Standard-Komponente in einem Interactive Report enthalten
- Die Bereitstellung von RESTful-Webservice ist in der Kombination mit den Oracle-Rest-Data-Services möglich (sie-

Oracle verlängert Support für Forms und Reports 11g R2

In den aktuellen Support Policies hat Oracle die Verlängerung des Supports von Forms und Reports 11g R2 angekündigt (siehe „<http://www.oracle.com/us/support/library/lifetime-support-middleware-069163.pdf>“). Der „Premier Support“ für beide Tools wird demzufolge um zwei weitere Monate bis Dezember 2016 verlängert, der „Extended Support“ um 14 Monate bis Dezember 2018 ausgeweitet. Die

Version 11g R1 ist davon nicht betroffen (siehe Tabelle).

Die Ankündigung des verlängerten Supports könnte wie schon in der Vergangenheit bei früheren Versionen ein Hinweis auf das baldige Erscheinen von Forms und Reports 12 sein. Michael Ferrante, Oracle Product Manager Forms, hatte bereits auf der DOAG 2014 Konferenz + Ausstellung das Erscheinen von Version 12 von Forms und Reports

im Kalenderjahr 2015 angekündigt. Damit widerlegte er gleichzeitig auch unbestätigte Informationen darüber, dass Oracle Reports nicht in Version 12 erscheinen könnte.

Ferrante legte außerdem auf Twitter allen Kunden, die noch Oracle Forms 6 nutzen, ein Upgrade auf Version 11 nahe. Oracle stellt hierfür einen Upgrade Guide zur Verfügung (siehe „http://docs.oracle.com/cd/E48391_01“).

Release	GA Date	Premier Support Ends	Extended Support Ends	Sustaining Support Ends
Portal 11gR1, Forms 11gR1, Reports 11g R1 and Discoverer 11g R1 (11.1.1.x)	Jun 2009	Jun 2014	Jun 2017	Indefinite
Forms and Reports 11g R2 (11.1.2.x)	Oct 2011	Dec 2016	Dec 2018	Indefinite

he „<http://www.oracle.com/technetwork/developer-tools/rest-data-services/overview/index.html>“)

- Native Excel-Upload ist ebenfalls in Kombination mit den Oracle-Rest-Data-Services möglich. Durch eine einfache Erweiterung der Parameter lassen sich Excel-Dateien im Original-Format hochladen und die Daten einlesen. Auch das Einlesen von beliebig vielen Datenblättern funktioniert problemlos
- Apex eignet sich für die Entwicklung und Bereitstellung von multisprachigen Anwendungen. In der neuen Version 4.2.0.6 ist es nun möglich, die Anwendungen auch ohne neues Deployment aus der Benutzeroberfläche der primären Anwendung heraus zu übersetzen
- Apex ist eine offene Plattform. Man kann den Code nicht verstecken. So steht für alles, was man in Apex einsetzen kann, auch das dazugehörige API zur Verfügung. Dadurch lässt sich alles, was man in der Apex-Oberfläche erledigen kann, genauso im Backend anstoßen

Diese Liste ist nicht vollständig und ein Teil ist sicherlich subjektiv empfunden. Sie

```
BEGIN
    package_name.funktion_oder_procedure (:p1_variable_1,
                                           :p1_variable_2,
                                           :p1_return_wert
                                           );
END;
```

Listing 1

```
SELECT column1, column2, column3
FROM my_table
WHERE column_2 = :p1_variable_1
```

Listing 2

zeigt aber, dass Apex durchaus eine sehr ernsthafte Alternative zu anderen älteren Frameworks geworden ist.

Community und Ressourcen

Eines der großartigen Dinge bei Apex ist die herausragende Community. Apex bedeutet Web – es wird im Web gelernt, gelebt und weiterentwickelt. Wenn man eine Lösung für sein Problem benötigt, ist das Web die Stelle, an die man sich wendet. Das Know-how wird gerne weitergereicht

und so etwas wie Geheimhaltung ist dort völlig unbekannt.

Nachfolgend sind einige der wichtigsten Ressourcen im Web aufgelistet, allesamt ein Muss. Sie beinhalten schon alles, was man über Apex wissen muss. Von dort kommt man dann auch zu den anderen Ressourcen, für die letzten Details:

- **Oracle Apex Forum**
https://community.oracle.com/community/database/developer-tools/application_express

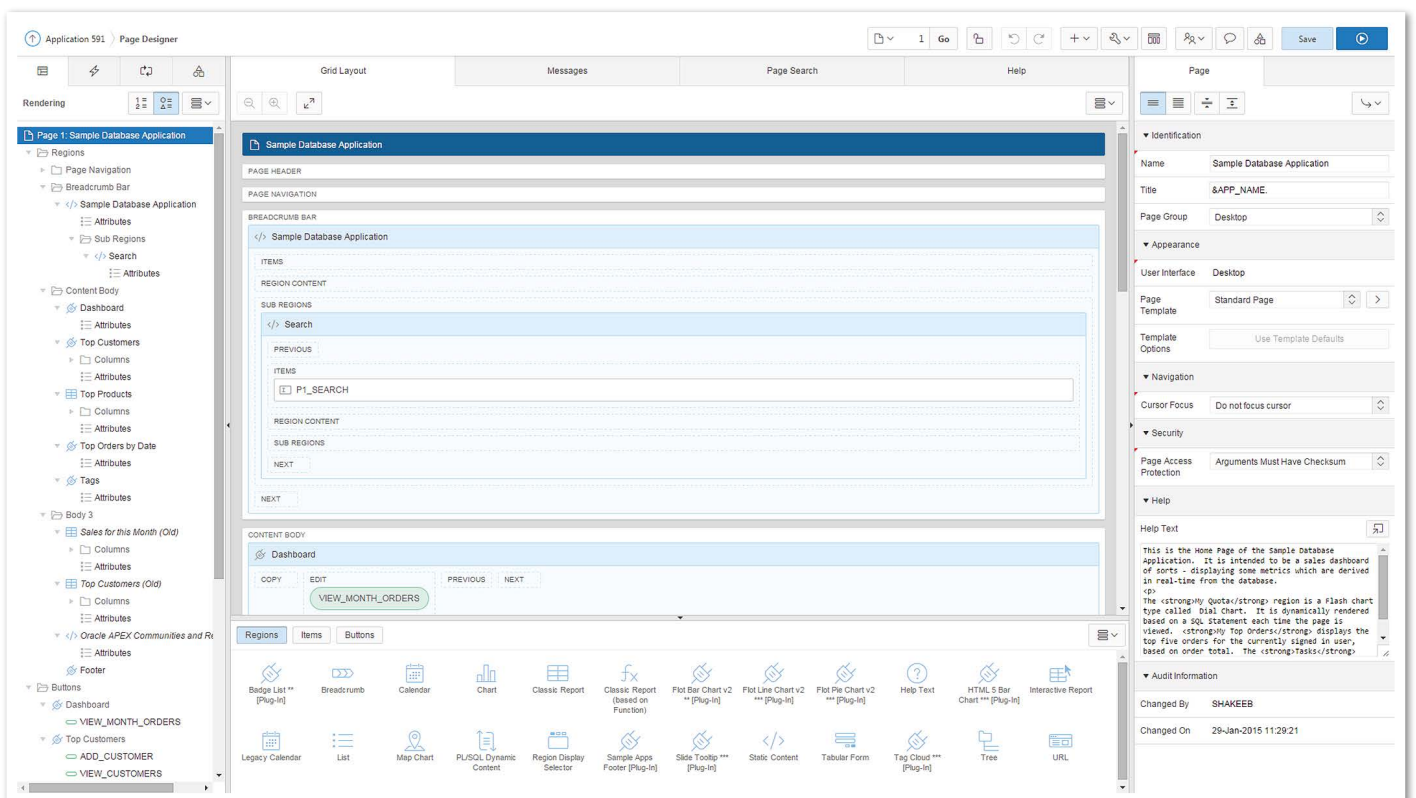


Abbildung 3: Der neue Page-Designer

Spiegelung kompletter Systemumgebungen

- **Deutsche Oracle Apex Community**
<http://www.oracle.com/webfolder/technetwork/de/community/apex/index.html>
- **All Apex Blogs**
<http://www.odtug.com/apex>
- **Oracle Apex Documentation**
<http://www.oracle.com/technetwork/developer-tools/apex/documentation/index.html>
- **Oracle Apex Workspace**
<https://apex.oracle.com/pls/apex/f?p=4550:1>
- **jQuery Ressourcen**
<http://stackoverflow.com>

Zu beachten

Apex ist kein Wundermittel für das Programmieren von guten Anwendungen. Nur deswegen, weil eine Anwendung mit Apex umgesetzt wurde, wird sie nicht automatisch gut. Wenn man mit Apex komplexere Anwendungen programmieren möchte, müssen zwei Voraussetzungen erfüllt sind:

- Sehr gute und fundierte SQL- und PL/SQL-Kenntnisse
- Sofern das Layout und die Ergonomie eine größere Rolle spielen, sind Kenntnisse in jQuery, CSS und HTML5 unausweichlich

In Apex gibt es viele Stellen, die Code aufnehmen können. Doch nur deshalb sollte man das nicht machen. Wenn keine Planung und Best-Practices vorliegen, wird dies zur sogenannten „Code-Redundanz“ führen. In der Praxis kommt es sehr häufig vor, dass jeder Entwickler für sich mit den anonymen PL/SQL-Blöcken arbeitet. Wenn eine komplexe Seite auf diese Weise entsteht, wird sie zwangsweise überladen und mit der Zeit langsam. Die Auswirkungen werden dann erst in der Produktion sichtbar, wenn der Code mit einer richtigen Datenmenge arbeitet.

Komplexe Dinge auslagern

Der Autor empfiehlt an dieser Stelle, die Business-Logik in die Packages auszulagern. Der Code sollte an bestimmten Stellen nie zum Einsatz kommen (auch wenn es möglich wäre) und idealerweise nur aus einem einfachen Aufruf bestehen (siehe Listing 1). Was die SQL-Abfragen betrifft, sollten komplexe Abfragen ebenfalls zuerst in die Views ausgelagert werden (siehe Listing 2).

Sämtliche Joins, Funktionsaufrufe und Unterabfragen sollten innerhalb einer darunter liegenden View verborgen sein. Man sollte diese Logik auf die Spitze treiben und immer darauf achten, es beim nächsten Mal noch ein wenig besser zu machen. Die Gründe dafür sind ganz praktisch und einfach:

- Die Code-Redundanz wird vermieden
- Der Code läuft schneller, da er in die Packages und Views ausgelagert ist
- Das Deployment der Anwendung muss nicht so häufig durchgeführt werden, da es oft reicht, nur das Package oder die View anzupassen
- Durch das Vermeiden der Code-Redundanz reduziert sich auch das Debugging, da sich der versteckte Code nicht wiederholt, sondern zentral verwaltet wird

Blick in die Zukunft

Als Befürworter von Apex ist der Autor natürlich sehr an der Zukunft von Apex interessiert. Durch seine Aktivitäten in der Community hatte er die Möglichkeit, sich an der Entwicklung von Apex 5.0 aktiv zu beteiligen. Die neuen Features sind bereits bekannt, darunter sind:

1. Universal Theme
2. Multiple Interactive Reports
3. Ein neuer Page-Designer mit Drag & Drop und einer Arbeitsweise, die an
4. Oracle Forms erinnert (siehe Abbildung 3)
5. Static File Upload
6. Modale Dialoge

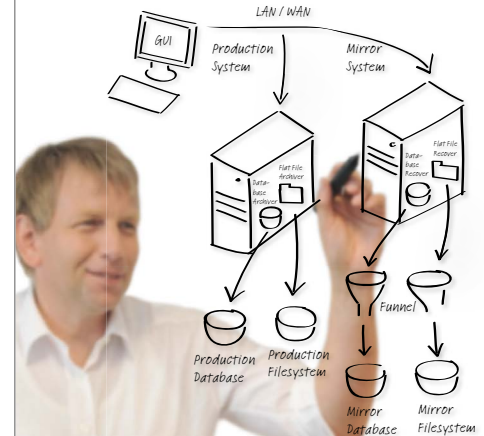
Man könnte sagen, dass Apex mit der Version 5.0 so richtig erwachsen wird. Wenn mit den Versionen 5.1 und 5.2 dann noch „jQuery Grid“ hinzukommt und die Collections auf mehr als 50 Spalten (200) ausgeweitet werden, wird Apex vom jetzigen Standpunkt aus kaum noch Wünsche übriglassen.

Warum Apex das richtige Tool ist

Wer auf einer Oracle-Datenbank arbeitet und seine Prozesse mit SQL und PL/SQL steuert, für den ist Apex ohne Zweifel die richtige und die beste Wahl.

Apex bietet eine einmalige Architektur, die es ermöglicht, sowohl Desktop- als auch mobile Anwendungen aus einer Hand zu

Libelle BusinessShadow®



Unabhängig bezüglich

- ✓ Fehlerursache
- ✓ Entfernung
- ✓ Hardware / Architektur
- ✓ Komplexer Systeme

Schnelle Arbeitsaufnahme

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

Hans-Joachim Krüger
Chief Technology Officer
Libelle AG

Erfahren Sie mehr:
www.Libelle.com/business



ORACLE Gold Partner



Libelle

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

liefern. Zudem ist alles noch sehr deklarativ und standardisiert.

Es ist ganz klar, dass auf dem Markt viele Frameworks zur Auswahl stehen. Apex wird diesen Konkurrenzkampf nicht immer für sich entscheiden können – manchmal auch deswegen, weil die Unternehmens-Entscheidungen häufig nicht mit rationalen Gründen und Argumenten getroffen werden.

Feedback aus der Community

Der Autor bekommt ab und zu E-Mails wie diese: „My Manager decided to cancel Apex from his list of solutions for the application programming and I tried in many ways to convince him that Apex is a true RAD tool and has very good reporting capabilities, etc... However, this doesn't work and his reasons are not clear to me. Probably based on his .NET background. The only thing he tells me all the time is that .Net is object oriented programming and oracle procedure programming. Please help me to find some things to convince him. I like Apex and I want to continue doing that. Any articles or any professional reasons would be a great help. Thank you very much,...“ Seine Antwort darauf ist, sich sofort einen neue Stelle zu suchen, an der man weiter mit Apex arbeiten kann.

Keine gezielte Begründung

Die Gründe für die Argumentation gegen Apex können verschieden sein. Häufig stehen die Architektur-Vorgaben im Vordergrund, die viele potenzielle Kandidaten ausschließen. Zum Beispiel wird als Voraussetzung festgelegt, dass es einen separaten Datenbank- und einen separaten Applikations-Server geben muss und Anwendungen und Business-Logik getrennt von der Datenbank zu verwalten sind.

Eine ausführliche Begründung, warum das so sein muss, gibt es meistens nicht. Es ist aber klar, für welche Technologien das ausgelegt ist und welche Beweggründe dahinterstehen. Dann hilft hier auch keine Argumentation, dass dies nur Geld kostet und keine Vorteile bringt.

Auf der anderen Seite existieren auch Fälle, bei denen die Kosten und die Zeit eine entscheidende Rolle spielen. Es gibt dafür genügend Beispiele aus der Praxis

und dort gewinnt Apex meistens mit einem klaren Vorsprung.

Der Autor kann sich an mehrere Situationen wie diese erinnern:

- Eine Anwendung wurde bei einem Kunden mit Java erstellt und musste nun erweitert werden
- Apex hat dort eine untergeordnete Rolle gespielt und nur Zusatzfunktionen zur Verfügung gestellt, die mit Java aus zeitlichen beziehungsweise Kapazitätsgründen nicht erledigt werden konnten. Das Layout war identisch und somit war die Plattform nur noch am Link im Browser erkennbar
- Der Kunde wollte eine relativ komplexe Verwaltung von Benutzern, Rollen und Rechten realisieren und dabei Pflege-masken haben, mit denen er den Zugriff auf seine Anwendungen gezielt steuern kann
- Dabei wurden Kalkulationen der Aufwände für eine Lösung in Java und eine Lösung mit Apex gemacht
- Der Vergleich fiel für Java vernichtend aus. Die Kalkulation des Aufwands für Java bezifferte man mit etwa 250 Mann-Tagen. Die Kalkulation für Apex dagegen lag bei drei Wochen
- Die Entscheidung war ganz klar und die Umsetzung dauerte am Ende etwas weniger als fünfzehn Mann-Tage

In einem anderen Fall musste aus politischen Gründen eine Funktionalität mit JSF umgesetzt werden. Der Kunde hatte schon Apex im Einsatz und für den Zweck wären Interactive Reports die richtige Wahl. Die Lösung hätte auch mit normalen Reports umgesetzt werden können. Die Anforderung war ziemlich trivial:

- Eine Seite beziehungsweise ein Bericht für das Suchen nach bestimmten komplexeren Informationen war auf Basis eines Header-Datensatzes und einiger Detail-Datensätze zu erstellen
- Vorgabe war, dass die Benutzer ihre Berichte jederzeit speichern und wieder abrufen können
- Einige Prozesse für die nachträgliche Änderung der Daten sollten in die Seite eingebunden werden

Der geschätzte Aufwand für die Umsetzung dieser Anforderungen mit Apex lag

bei etwa 20 und für JSF bei 75 Mann-Tagen. Die tatsächliche Realisierung mit JSF dauerte dann rund 200 Mann-Tage.

Fazit

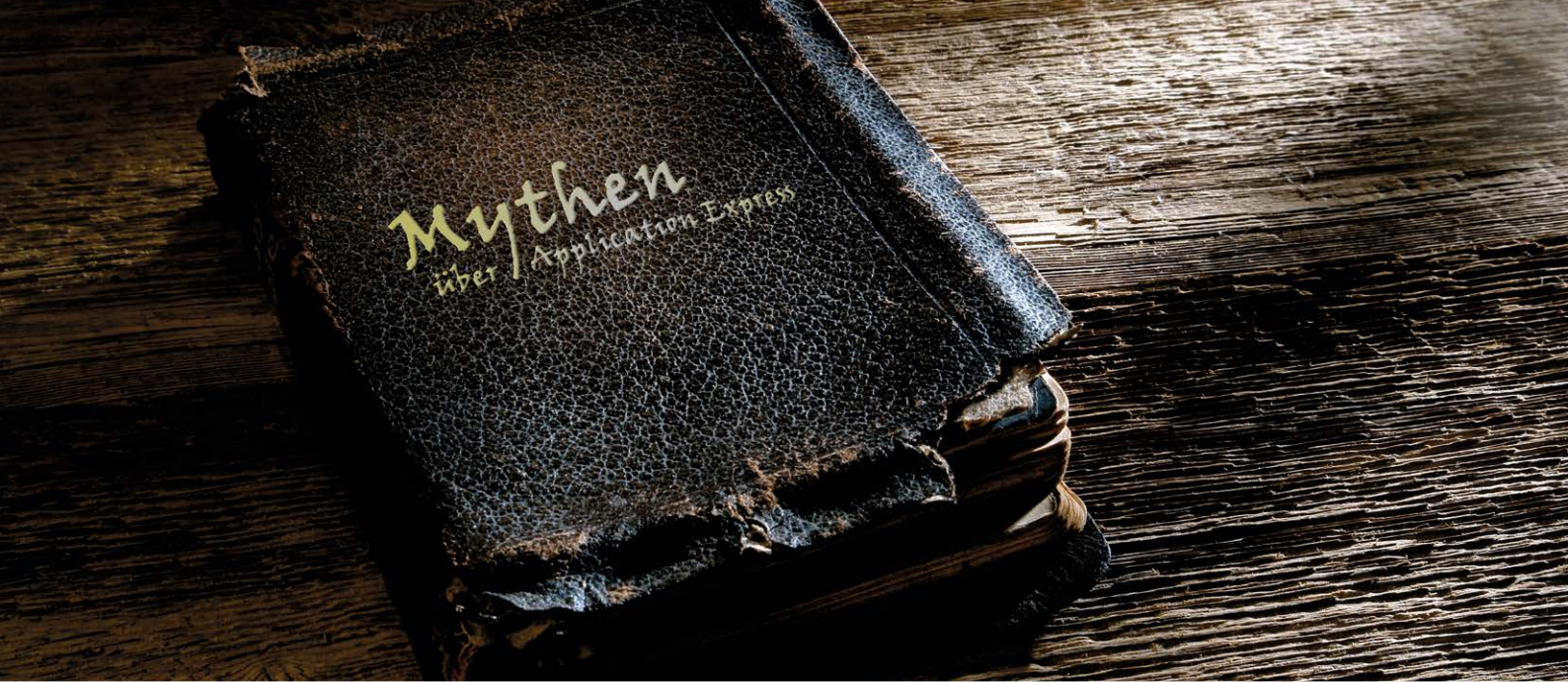
Wenn Kosten, Zeit und Transparenz im Vordergrund stehen, hat Apex gute Chancen, sich durchzusetzen. Je häufiger es sich durchsetzt, desto offensichtlicher werden die Vorteile auch für die Entwicklung aller anderen Anwendungen.

Selbstverständlich ist Apex kein Allheilmittel und kann nicht alles am besten lösen. Auch bei Apex trifft die „80:20“-Regel ganz gut zu. In 80 Prozent der Fälle wird Apex eine ernsthafte Alternative, die man auf jeden Fall berücksichtigen sollte. Und das ist schon sehr viel.

Die aufwändigen Application-Server, die bereits vor der Datenabfrage gewaltige Ressourcen beanspruchen, gehören der Vergangenheit an. In naher Zukunft wird sich auch in der Anwendungsentwicklung die „80:20“-Regel durchsetzen – die meisten Anwendungen, die wir benutzen und kennen, werden im Browser laufen. Nichts spricht mehr für Apex als das.



Denes Kubicek
deneskubicek@yahoo.de



Mythen über Apex

Markus Hohloch, DB Systel GmbH, und Tobias Arnhold, IT Consulting

Apex ist rund zehn Jahre alt. Genauso lange gibt es allerlei Mythen und Legenden darüber und die damit erstellten Anwendungen. Was stimmt daran? Was sind eher persönliche Vorlieben oder völliger Unsinn? Woher kommen diese Stimmen und warum halten sich diese zum Teil beharrlich? Die Autoren haben die häufigsten Vorurteile gesammelt und untersucht.

Mythos 1: **Apex sollte nicht in Produktiv-Umgebungen verwendet werden**

Dieses Vorurteil kommt vor allem aus zwei Richtungen. Einerseits sind es die Software-Entwickler und -Architekten. Ursachen sind meist Unwissenheit, andere Präferenzen oder die Angst vor Neuem. Wie nachfolgend gezeigt wird, ist Apex weder unperformant noch unsicher. Ein guter Entwickler wird deshalb eine ebenso leistungsstarke und sichere produktive Anwendung mit Apex erstellen können wie mit einer anderen Entwicklungsumgebung. Die entsprechende Erfahrung im jeweiligen Gebiet natürlich vorausgesetzt.

Zur zweiten Personengruppe, die sich gegen den produktiven Einsatz von Apex ausspricht, gehört der eine oder andere Datenbank-Administrator. Für ihn spielen natürlich Performance, Wartung und Sicherheit eine große Rolle. Apex liefert viele neue Datenbank-Objekte, die er als eine

Art Blackbox vielleicht nicht einschätzen kann. Die Installation von Apex macht die Datenbank allerdings nicht automatisch langsamer und unsicherer. Apex würde nicht standardmäßig mit der Datenbank installiert werden, wenn das der Fall wäre.

Beim Aufruf einer Apex-Seite wird die zugrunde liegende Datenbank-Sitzung stets mit dem niedrig privilegierten Datenbank-User „Apex_PUBLIC_USER“ aufgebaut. Für sicherheitskritische Systeme wird zudem das Runtime-Environment von Apex empfohlen. Dadurch ist auf der Datenbank keine Administrations- und Entwicklungsumgebung für Apex mehr vorhanden, sondern nur noch die zwingend notwendigen Pakete für den Betrieb der produktiven Anwendung.

Mythos 2: **Apex ist langsam und nicht skalierbar**

Diesen Mythos hört man tatsächlich immer wieder. Er ist aus Sicht der Autoren

absolut haltlos. Prinzipiell kann man sagen: Apex ist so skalierbar wie die Datenbank selbst – und wie allgemein bekannt ist, gehört die Oracle-Datenbank zu den schnellsten auf dem Markt.

Je besser der Datenbank-Server ausgestattet ist, desto mehr Last verträgt damit auch Apex. Der Overhead von Apex besteht im dynamischen Generieren von Webseiten via „http.p“ und in der Submit-Verarbeitung. Bei den meisten anderen Programmiersprachen übernimmt dies ein Web- oder Application-Server, der ebenfalls gewisse Ressourcen benötigt. Die von Apex erzeugte Last auf der Datenbank ist ziemlich gering. Wie hoch genau die Dauer dieser einzelnen Verarbeitungsschritte ist, kann man sehr schön mit der Debug-Funktion ablesen.

Natürlich ist es möglich, eine langsame Apex-Anwendung zu bauen. In diesen Fällen ist aber nicht die Apex-Verarbeitung langsam, sondern vielmehr der meist un-

performante SQL- oder PL/SQL-Code. Um solchen Problemen vorzubeugen, sollte daher der Fokus auf die folgenden beiden Themen gelegt werden:

- Optimales Datenbank-Modell
- Performanter SQL- und PL/SQL-Code

Das klingt einfach, aber dahinter steckt der Schlüssel zum Erfolg. Viele Apex-Entwickler lassen sich wegen der hohen Entwicklungsgeschwindigkeit zu undurchdachten SQL-Statements hinreißen. *Listing 1* zeigt das Beispiel einer Abfrage, mit der die Anzahl der Mitarbeiter je Abteilung ausgegeben werden soll. Ein typischer Anwenderfehler ist, dass die Anzahl über ein „Sub-Select“ anstelle eines „GROUP BY“ selektiert wurde. Die Lösung in *Listing 2* ist sehr viel besser geeignet.

Weitere Ansätze zur Optimierung der Geschwindigkeit sind:

- Caching für Regions und Pages aktivieren
- Anwendungsspezifische Bilder, JS- und CSS-Dateien auf dem Application-Server/Webserver auslagern
- Datenbank-Ausstattung und -Konfiguration prüfen
- Default-Parameter des Webserver prüfen (Beispiel „concurrent connections“)

Beispiele für hochskalierbare Apex-Anwendungen gibt es genug. Schon im Jahr

2007 liefen stark frequentierte Webseiten wie „metalink.oracle.com“ oder „asktom.oracle.com“ auf einer Frühform des heutigen Apex. Die meisten Apex-Anwendungen laufen jedoch intern, um unternehmensspezifische Aufgaben zu lösen. Einige davon sind unternehmenskritisch und haben beachtliche Größen von tausend und mehr Seiten angenommen.

Mythos 3: Apex ist für Anwendungen mit komplexer Geschäftslogik nicht geeignet

Woher kommt dieses Vorurteil? Wahrscheinlich daher, dass es durch Assistenten sehr einfach möglich ist, aus einer Datenbank-Tabelle eine vollständige Web-Applikation zu erstellen. Ohne Übertreibung kann man behaupten, dass man in wenigen Minuten einen Report und ein Formular zur Bearbeitung von Datensätzen erstellen kann – ohne SQL- und PL/SQL-Kenntnisse und inklusive einfacher Validierungen. Vor allem für die Erstellung von ersten Prototypen und Stammdaten-Tabellen (CRUD = Create, Read, Update, Delete) ist das schon ganz ordentlich. Es ist jedoch nur der erste Schritt. Mit SQL, PL/SQL und den üblichen Web-Sprachen sind der Weiterentwicklung kaum Grenzen gesetzt. Als einfachen Excel- oder Access-Ersatz kann man Apex damit nicht mehr bezeichnen.

Aber wie geht man nun mit komplexer Geschäftslogik um? Eine höhere Komplexität senkt natürlich die Entwicklungsgeschwindigkeit. Das ist allerdings normal und unabhängig von der verwendeten Programmiersprache. Man sollte auch bei der Entwicklung mit Apex Standards festlegen und sich mit Kollegen darüber abstimmen, wer welchen Teil übernimmt und wie dabei vorgegangen werden soll.

Damit die Apex-Applikation weiterhin übersichtlich bleibt, sollte man komplexen PL/SQL-Code nicht in Apex-Seiten speichern, sondern in PL/SQL-Packages auslagern. Das erhöht zusätzlich die Test- und Wiederverwendbarkeit.

Aber wo liegen die Grenzen? Apex ist am besten geeignet für datenzentrierte Anwendungen, also für die Anwendungen, bei denen die Datenbank eine Schlüsselrolle spielt. Wenn Spezial-Anforderungen den Hauptteil ausmachen, verwendet man natürlich auch dafür konzipierte Software beziehungsweise Frameworks. Ereignisbasierte Anwendungen erstellt man weiterhin mit Workflow-Management-Systemen und eher selten würde man das x-te Content-Management-System mit Apex erstellen. Auch beim nächsten YouTube wird nicht Apex unter der Haube stecken.

Zusammenfassend kann man sagen: Ja, man kann mit Apex genial einfach CRUD-Seiten erstellen. Aber genauso gut ist Apex für die Erstellung von komplexen Anwendungen geeignet. Vor allem dann, wenn es um die Verarbeitung strukturierter Daten geht.

Mythos 4: Apex-Skills sind notwendig

Der Einstieg in Apex ist sehr einfach. Der Grund dafür ist das deklarative Konzept der Entwicklungsumgebung. Über die Browser-basierte Oberfläche ist es möglich, nach kurzer Einführung eine erste Anwendung zu erstellen. Wichtig ist vor allem zu wissen, dass Apex seitenorientiert aufgebaut ist. Man erstellt Seiten mit Reports, Formularen, statischen Textelementen etc. Für diese Seiten und Unter-elemente kann man dann verschiedene Eigenschaften deklarativ festlegen. Beispielsweise, in welchem Layout und für welche Benutzerrollen diese dargestellt werden sollen.

Von diesen Schaltern und Konfigurationsoptionen gibt es in Apex eine ganze Men-

```
select a1.id,
       a1.name,
       (
         select count(m1.id)
         from t_mitarbeiter m1
         where m1.abteilung_id = a1.id
       ) as anzahl_mitarbeiter
from t_abteilung a1
```

Listing 1

```
select a1.id,
       a1.name,
       count(m1.id) as anzahl_mitarbeiter
from t_abteilung a1
left join t_mitarbeiter m1
  on m1.abteilung_id = a1.id
group by
  a1.id,
  a1.name
```

Listing 2

ge. Mit der Zeit sollte man diese Funktionen alle einmal gesehen haben. Dann stellt man fest, dass vieles sehr einfach umzusetzen ist – auch mit wenig Code. Dazu findet man im Internet eine Menge Tutorials, Blogs und Foren, die bei der Beantwortung von Fragen helfen. Alles Weitere hat nichts mit Apex-Skills im engeren Sinne zu tun.

Um komplexere Anwendungen zu erstellen, sollte man einfach folgende fünf Dinge kennen: SQL, PL/SQL, HTML, CSS und JavaScript. Die Abfragesprache SQL kennt jeder Entwickler, der mit relationalen Datenbanken arbeitet. PL/SQL ist das Java, PHP oder C# der Oracle-Datenbanken. Sowohl Apex selbst als auch die damit entwickelten Anwendungen bestehen aus PL/SQL-Code. Wenn man mit Oracle noch nichts zu tun hatte, sollte man sich hier einarbeiten. Die Sprache ist allerdings ziemlich einfach zu erlernen. HTML, CSS und JavaScript sind Standards im Umfeld von Web-Applikationen. Die Grundlagen von HTML und den Ablauf einer „http“-Anfrage und -Antwort sollte man kennen. Und für nutzerfreundliche Anwendungen sind CSS und JavaScript ebenfalls sinnvoll.

Zusammenfassend kann man sagen, dass die Lernkurve beim Einstieg in Apex sehr steil ist. Aus eigener Erfahrung behaupten die Autoren, dass man nach wenigen Stunden bereits einfache Anwendungen erstellen kann. Für komplexere Anforderungen sind erweiterte Kenntnisse in PL/SQL und Web-Techniken Pflicht.

Mythos 5: Apex ist unsicher

Apex selbst bietet eine sehr sichere Umgebung. Es ist aber durch fehlendes Wissen seitens der Apex-Entwickler möglich, gravierende Sicherheitslücken in die Anwendungen einzubauen.

Grundsätzlich lässt sich sagen: Apex-Anwendungen bieten die gleichen Fallstore wie Web-Applikationen, die mit Java, .Net, PHP oder Ähnlichem erstellt wurden. Folgende Regeln sollte man daher beachten, um möglichst sichere Apex-Anwendungen zu bauen:

- Session-State-Protection (Schutz vor URL-Manipulation) aktivieren
- Falls möglich, die Apex-Standardprozedur für Insert, Update und Delete anstelle von manuellem PL/SQL verwenden

- Keine „&P1_ID. Notation“ in SQL verwenden (Einfallstor für SQL-Injection)
- Den Apex-Advisor nutzen
- Für Internet-Anwendungen möglichst das Apache-Modul „ModSecurity“ in Verbindung mit Black- und Whitelists verwenden (zum Beispiel nur bestimmte „APP_Ids“ für den Zugriff freigeben)

Im Allgemeinen sollte man immer darauf achten, die neueste Apex-Version innerhalb eines Major-Release zu verwenden, also anstatt Apex 4.2.2 heute Apex 4.2.6 einzusetzen.

Die beschriebenen Beispiele gehen nur auf die wichtigsten Themen ein. Es geht darum, ein Gefühl dafür zu vermitteln, auf was man achten sollte und welche Möglichkeiten Apex dabei bietet.

Mythos 6: Niemand verwendet Apex

Dieser Mythos sollte aus unterschiedlichen Perspektiven betrachtet werden. Hinsichtlich der Verbreitung in Unternehmen ist Apex kein exotisches Produkt, sondern viel weiter verbreitet, als manchem bewusst ist. Ein überwiegender Teil der deutschen Großunternehmen hat Apex mehr oder weniger intensiv im Einsatz. Grund dafür ist die hohe Verbreitung der Oracle-Datenbanken. Apex bietet sich dann als gern genutztes Feature an, um direkt mit den Daten zu arbeiten.

Neben dem Selbststudium mithilfe von Dokumentationen sowie Best-Practice-Beispielen in Blogs, Videos und Foren erhalten interessierte Einsteiger auch direkte Unterstützung von Experten. Eine Vielzahl von Beratern und Unternehmen haben sich mittlerweile auf Apex spezialisiert und bieten Lösungen, Entwicklungsleistungen und Schulungen rund um das Thema an. Die starke Nachfrage nach Apex-Lösungen hat aber auch dazu geführt, dass aktuell viele Unternehmen Apex-Entwickler zur Feststellung suchen.

Die Apex-Community ist eine der aktivsten Communities innerhalb von Oracle. Das Apex-Forum beispielsweise ist direkt nach dem SQL-Forum das mit den meisten Foreneinträgen überhaupt. Ein weiteres Beispiel von der letzten DOAG Konferenz: Wenn andere Teilnehmer der Konferenz am Abend bereits feiern, dann macht die Apex-Community eine Open MIC Night.

Nicht, weil die Teilnehmer müssen, sondern weil sie es wollen. Zu guter Letzt ist auch die hohe Anzahl an weltweiten Apex-Events mit und für Apex-Begeisterte zu nennen: Oracle OpenWorld, ODTUG KScope, DOAG Konferenz, DOAG Regionaltreffen, Apex Connect, Apex World und viele regionale Meetup-Gruppen.

Fazit

Die Autoren hoffen, die teilweise plumpen Vorurteile entkräftet zu haben. Ihrer Meinung nach ist Apex sehr wohl produktiv einsetzbar, sicher, schnell, einfach zu erlernen und weitverbreitet. Die aktive und begeisterte Community motiviert umso mehr, mit Apex zu arbeiten.



Tobias Arnhold
tobias-arnhold@hotmail.de



Markus Hohloch
markus.hohloch@deutschebahn.com



Application Builder



SQL Workshop



Team Development



Packaged Apps

Apex 5.0: Anwendungsentwicklung völlig neu definiert

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

Das neue Release 5.0 des populären Entwickler-Frameworks Oracle Application Express sieht nicht nur anders aus, es bringt auch eine Fülle neuer Funktionen mit: So nähert sich Apex mit dem Page Designer einer Desktop-Entwicklungsumgebung an – und zwar im Browser.

Das „Universal Theme“ macht die Erstellung von Anwendungen mit modernem „Look & Feel“ einfach – und mit dem „ThemeRoller“ können Apex-Entwickler zum ersten Mal das Farbschema ihrer Anwen-

dung ändern, ohne CSS- oder HTML-Code schreiben zu müssen. Doch das ist nicht alles, es sind auch eine ganze Menge kleinerer neuer Funktionen dabei. Bei Drucklegung dieses Artikels steht Apex 5.0 noch

nicht produktiv zur Verfügung. Die neuen Funktionen werden daher anhand des „Early Adopter“-Release vorgestellt – im Vergleich zum endgültigen Release können sich noch Änderungen ergeben.

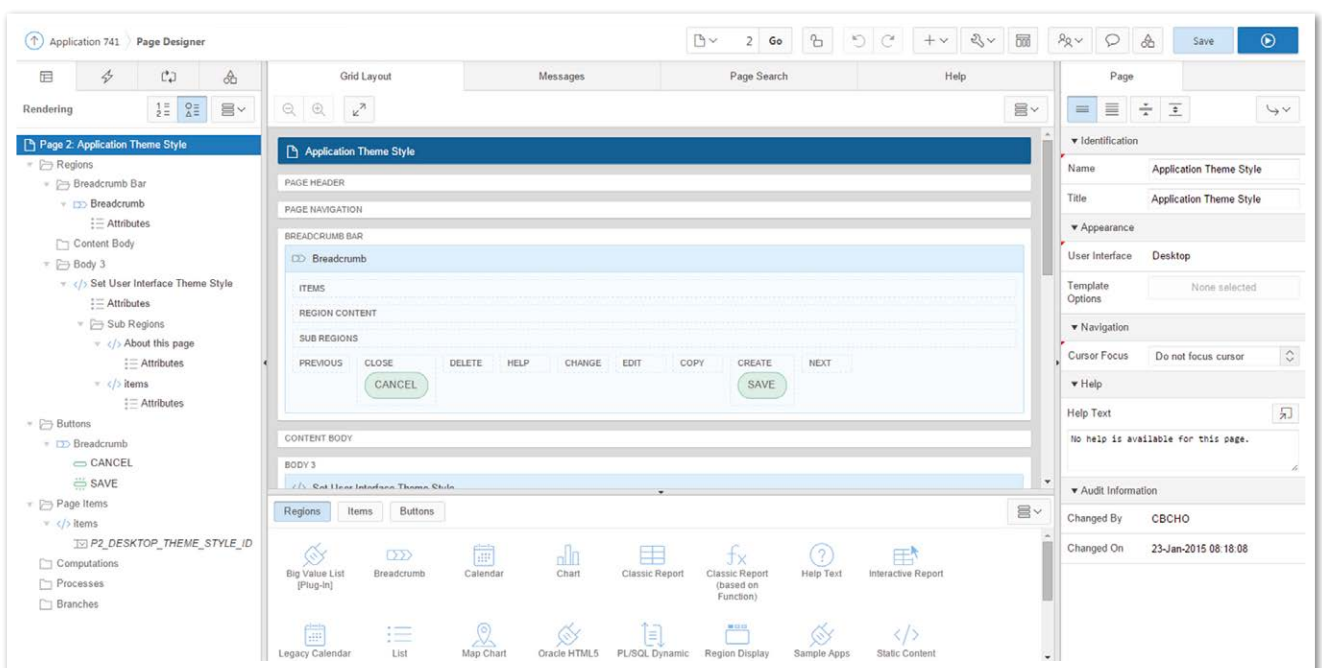


Abbildung 1: Apex 5.0 Page Designer

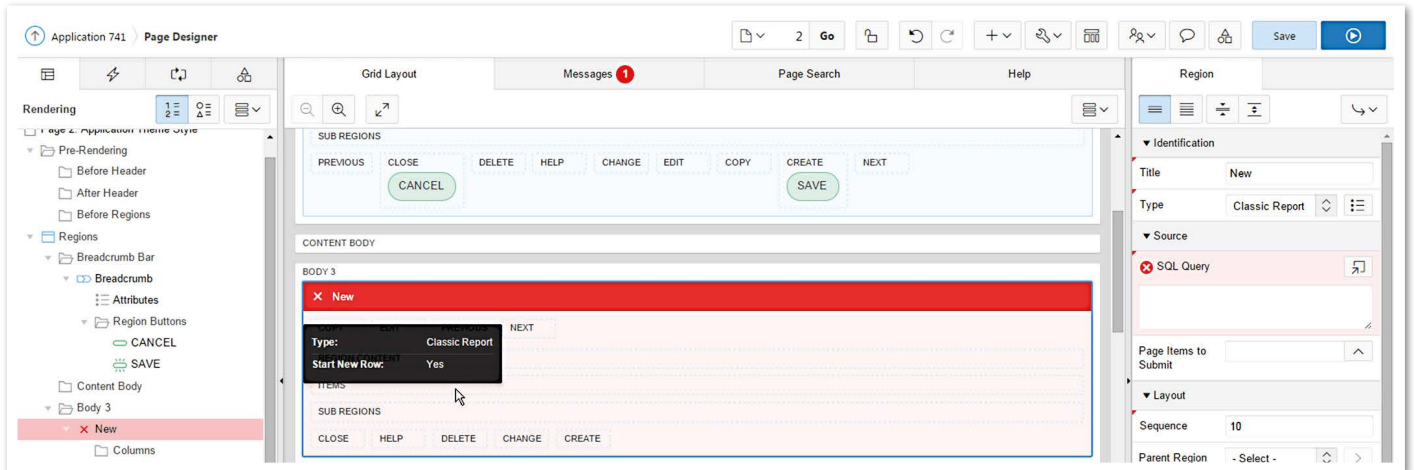


Abbildung 2: Der Page Designer weist den Entwickler auf fehlende Angaben hin

Apex Page Designer

Der neue Page Designer ist die auffälligste Neuerung für den Apex-Entwickler – das Arbeiten an einer Anwendungsseite ändert sich völlig: Steigt man in die Bearbeitung einer Anwendungsseite ein, wird dem Entwickler eine Umgebung präsentiert, die man ansonsten von Desktop-Werkzeugen gewohnt ist. Der Bildschirm wird in miteinander integrierte Bereiche aufgeteilt – in diesen kann der Entwickler die Seitenstruktur und die Eigenschaften der Komponenten, unter anderem auch per „Drag & Drop“, bearbeiten (siehe Abbildung 1):

- Der Strukturbereich links gibt die Seitenstruktur als Baum-Ansicht wieder; der Entwickler erhält hier einen Überblick über die Zuordnung der Komponenten untereinander; also beispielsweise darüber, welche Elemente zu welcher Region gehören. Außerdem enthält der Baum Komponenten, die im Seitenlayout nicht sichtbar sind, wie versteckte Elemente, Berechnungen und Prozesse.
- Der Preview-Bereich in der Mitte stellt die Seite schematisch dar; man kann erkennen, wie die Regionen, Schaltflächen und Elemente zueinander angeordnet

- sind. Es ist zwar kein echtes „What you see is what you get“, aber man erkennt sehr gut, ob zwei Regionen nebeneinander oder untereinander angeordnet sind.
- Aus dem Bereich der verfügbaren Komponenten unten kann der Entwickler per „Drag & Drop“ Komponenten auf den Preview-Bereich ziehen – und sie so direkt der richtigen Position in der richtigen Region zuordnen.
- Im Property Editor rechts sind die Eigenschaften der Komponenten dargestellt.

Bemerkenswert und ein wesentlicher Unterschied zu früheren Versionen ist, dass

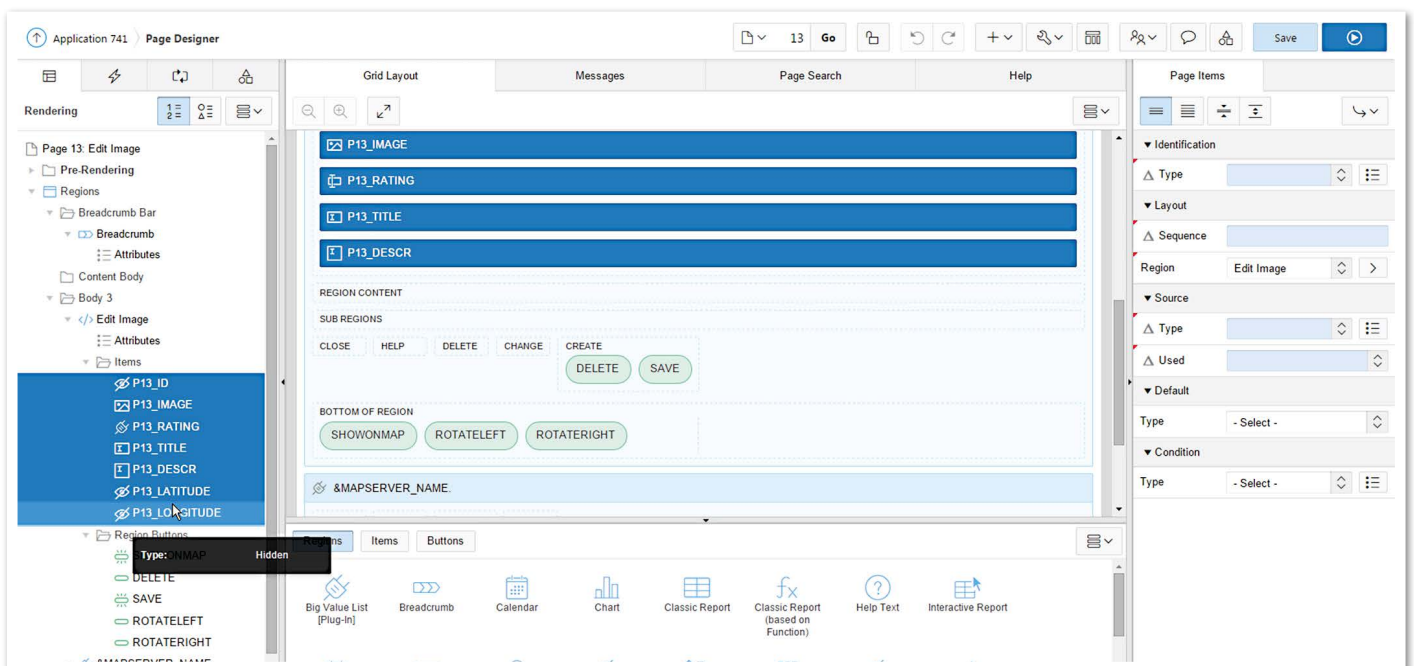


Abbildung 3: Multi-Selection im Page Designer

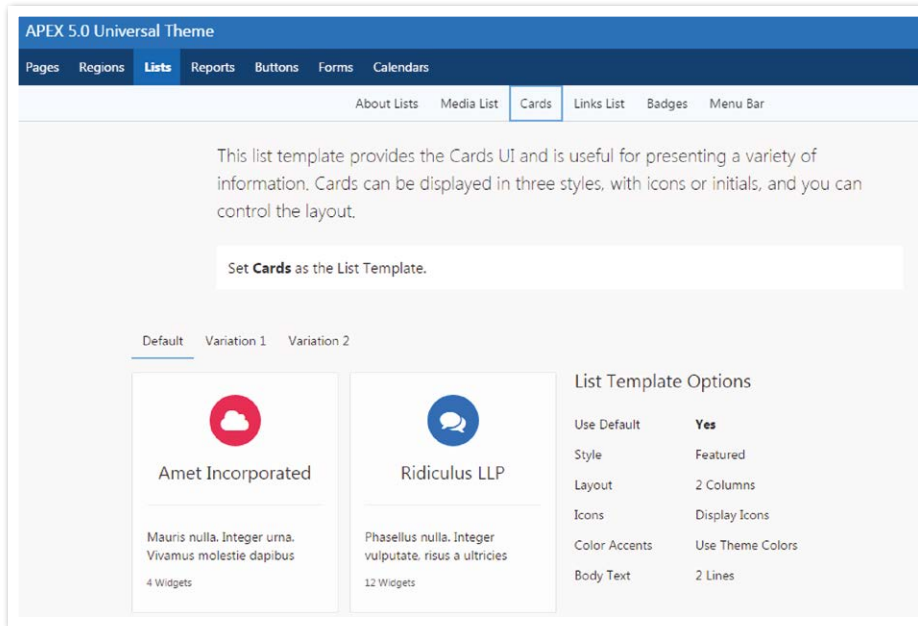


Abbildung 4: Eine Übersicht über die Möglichkeiten des Universal Theme

Apex alle Änderungen sofort prüft und den Entwickler auf fehlende Eigenschaften oder gemachte Fehler hinweist. Wenn man zum Beispiel der Seite per „Drag & Drop“ einen Bericht hinzufügt, so weist Apex unmittelbar auf die nun fehlende SQL-Abfrage hin (siehe Abbildung 2).

Man kann der Seite beliebig Komponenten hinzufügen, diese löschen oder verändern; all diese Änderungen werden zunächst jedoch nur im Browser – und nicht auf dem Server – gespeichert. Folgerichtig sehen andere Entwickler nichts davon.

Erst ein Klick auf die Schaltfläche zum Speichern oder zum Starten der Anwendung (rechts oben) sorgt dafür, dass die Änderungen aktiv werden. Ein Entwickler kann so wesentlich schneller arbeiten, denn bei größeren Änderungen an einer Seite entfallen die vielen Round Trips zum Server.

Es ist zu erkennen, dass der Page Designer eine ganze Menge Intelligenz in den Browser bringt, denn Hinweise auf fehlende oder fehlerhafte Attribute bekommt der Entwickler jetzt unmittelbar. Vor allem für erfahrende Apex-Nutzer ist die neue Fähigkeit

zur Multi-Selection bemerkenswert; es können also mehrere Komponenten auf einmal angewählt und verändert werden. So ist zum Beispiel die Änderung des Label Template für alle Elemente eines Formulars mit wenigen Klicks erledigt (siehe Abbildung 3).

Der Page Designer bringt noch weitere Funktionen mit, die hier jedoch nicht im Detail erläutert sind, unter anderen Syntax Highlighting für SQL-, PL/SQL- und JavaScript-Code, Code Completion, Undo- und Redo-Funktionen sowie kontextsensitive Hilfe.

Universal Theme

Ein Apex-Entwickler ist mit dem Template- und Theme-Konzept von Apex vertraut: Das Aussehen jeder Komponente wird durch ein Template gesteuert; dieses enthält den HTML-Code, der zum Browser gesendet wird. So gibt es Seiten-, Regions-, Element-, Label- und andere Templates. Zusammengehörige Templates, die ein konsistentes „Look & Feel“ ergeben, sind zu einem Theme zusammengefasst. Apex bringt etwa 30 Themes mit. Möchte man ein eigenes „Look & Feel“ erstellen, muss man für die einzelnen Apex-Komponenten Templates bauen und diese zu einem Theme zusammenfassen. Apex 5.0 führt das neue Universal Theme mit der Nummer 42 ein: Dieses sorgt schon mal aus dem Stand für ein modernes „Look & Feel“ (siehe Abbildung 4).

Das Universal Theme ist auch konfigurierbar: Der Entwickler kann auf Ebene der Apex-Komponente sogenannte „Template

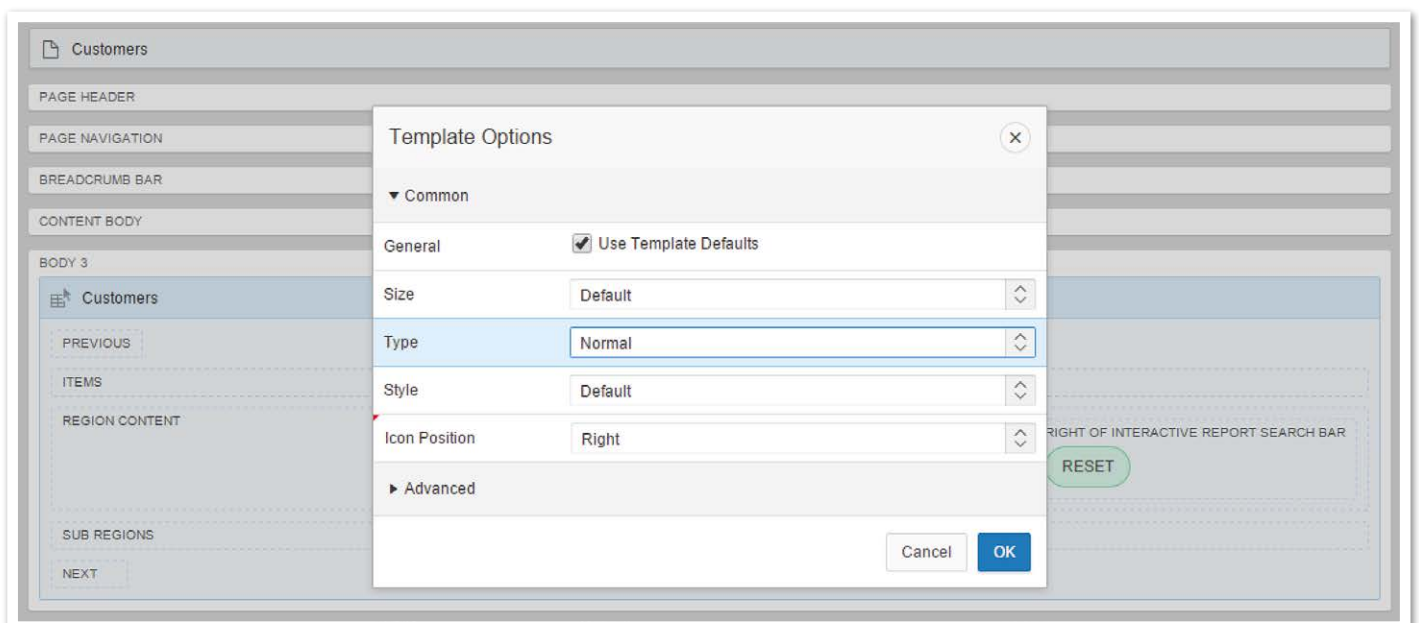


Abbildung 5: Template Options für eine Schaltfläche

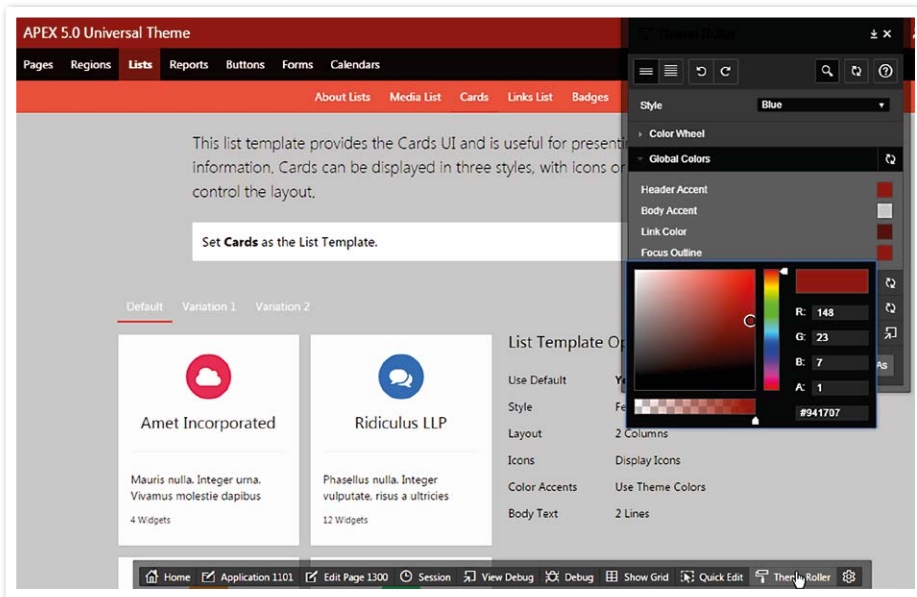


Abbildung 6: Universal Theme Roller

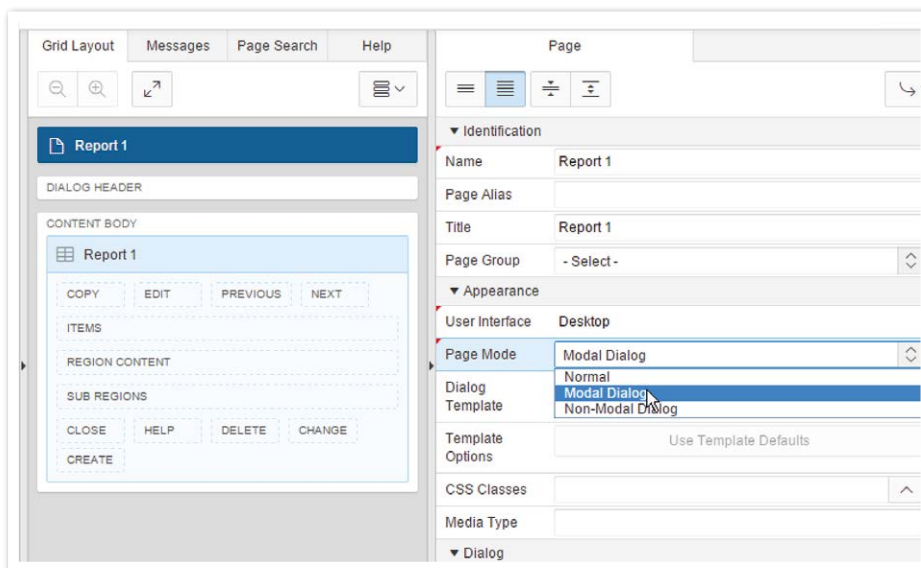


Abbildung 7: Eine Apex-Seite wird zum modalen Dialog

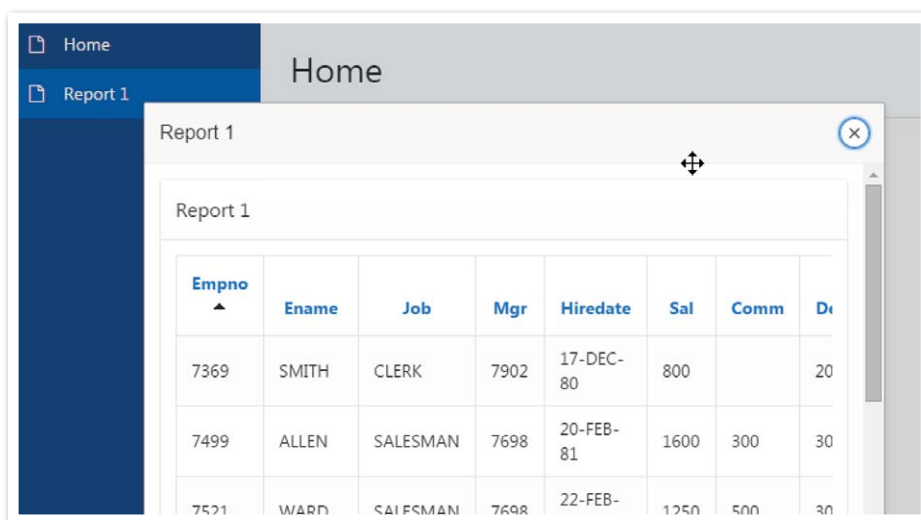


Abbildung 8: Modaler Dialog in Aktion

Options“ einstellen. Damit werden Darstellungsvarianten wie Scrollbars, links- oder rechtsbündige Ausrichtung und Ähnliches realisiert. Für eine Komponente bietet Apex nun nur noch wenige Templates an, die dann mit Template Options konfiguriert werden können (siehe Abbildung 5).

Mit dem neuen Theme Roller lassen sich alle Farben bequem einstellen, ohne auch nur eine Zeile CSS zu programmieren (siehe Abbildung 6). Der so erstellte Theme Style kann gespeichert und für die Anwendung genutzt werden. Ein Theme kann mehrere Theme Styles haben, es ist jedoch pro Anwendung immer nur einer aktiv.

Die meisten Apex-Entwickler werden ab Version 5.0 nur noch ein Theme benötigen: Das Universal Theme – und das zeitgemäße Aussehen einer Apex-Anwendung bekommt man „frei Haus“ dazu. Übrigens hat das Apex-Team im Universal Theme auch auf die etwas sperrigen und unflexiblen Reiterkarten („Tabs“) verzichtet – die Navigation ist nun mithilfe von „Lists“ und „List Templates“ realisiert.

Verwendete Tabs in bestehenden Anwendungen laufen natürlich auch in Apex 5.0 problemlos. Am besten lernt man das Universal Theme kennen, indem man sich die Packaged Application „Universal Theme Sample Application“ installiert und betrachtet.

Modale Dialoge

Modale Dialoge sind ein sehr wichtiges Instrument für einen Anwendungsentwickler, gerade für Web-Anwendungen. Ein modaler Dialog muss vom Endanwender abgeschlossen werden, bevor eine andere Aktivität möglich ist – dies verhindert, dass der Endanwender einen Dialog öffnet, dann aber an anderer Stelle weiterarbeitet. In Apex 5.0 kann jede Anwendungsseite als modaler Dialog deklariert sein (siehe Abbildungen 7 und 8).

ZIP-Unterstützung

Apex 5.0 führt eine konsequente Unterstützung von ZIP-Archiven ein. Diese können als statische Dateien nicht nur hochgeladen, sondern von Apex auch direkt ausgepackt werden. Die Verzeichnisstruktur des ZIP-Archivs bleibt nicht nur erhalten, die Dateien dürfen sich nun auch untereinander mit relativen Pfaden referenzieren. Das Hochladen eines Pakets aus HTML-, JavaScript- und CSS-Dateien in

den Apex-Workspace ist mit Apex 5.0 kein Problem mehr (siehe Abbildung 9).

Die neue ZIP-Unterstützung steht auch dem Apex-Entwickler zur Verfügung. Mit dem PL/SQL Paket „Apex_ZIP“ kann man ZIP-Archive nun auch in einer eigenen Apex-Anwendung ein- und auspacken, wie Listing 1 zeigt.

Apex-Workspace-Login

Das Anmelden am Apex-Workspace als Entwickler oder Administrator wurde flexibler gestaltet. Mussten diese Nutzerkonten bisher stets im Apex-Workspace eingetragen sein, ist es nun möglich, dafür ein anderes Login-Verfahren zu verwenden. Nutzer des Early-Adopter-Programms konnten diese Funktion in Aktion sehen, denn für diese Umgebung erfolgt der Login mit dem Konto des Oracle Technology Network (OTN) – nach erfolgreichem Login werden die Workspaces, auf die man Zugriff hat, präsentiert. Ein Klick auf den Workspace führt schließlich zur Apex-Homepage. Im Gegensatz zu Apex-Anwendungen, bei denen ja ein beliebiges Login-Verfahren gewählt werden kann (bis hin zu einem selbst programmierten), steht zum Login am Workspace nur eine eingeschränkte Auswahl an Login-Verfahren bereit.

Dies und Das

Neben den vorgestellten Funktionen bringt Apex 5.0 noch mehr Neuerungen mit; teilweise sind das Neuerungen im Detail, diese sind jedoch nicht weniger nützlich. Beispielhaft sind:

- Das neue PL/SQL-Paket „Apex_JSON“ erlaubt es dem Entwickler, JSON mit PL/SQL-Logik zu bearbeiten oder zu erzeugen
- Es können nun mehrere interaktive Berichte auf einer Seite platziert werden
- Der Endanwender kann einen interaktiven Bericht als Pivot-View betrachten; dabei werden die Berichtsspalten als neue Zeilen gekippt
- Die Kalender-Region wurde überarbeitet. Unter anderem wurde das „Look & Feel“ verbessert und der Endanwender kann die Daten als ICAL abrufen und so in seinen eigenen Kalender integrieren

Eine vollständige Aufzählung der einzelnen neuen Funktionen steht online zur Verfügung (siehe „Weitere Informationen“).

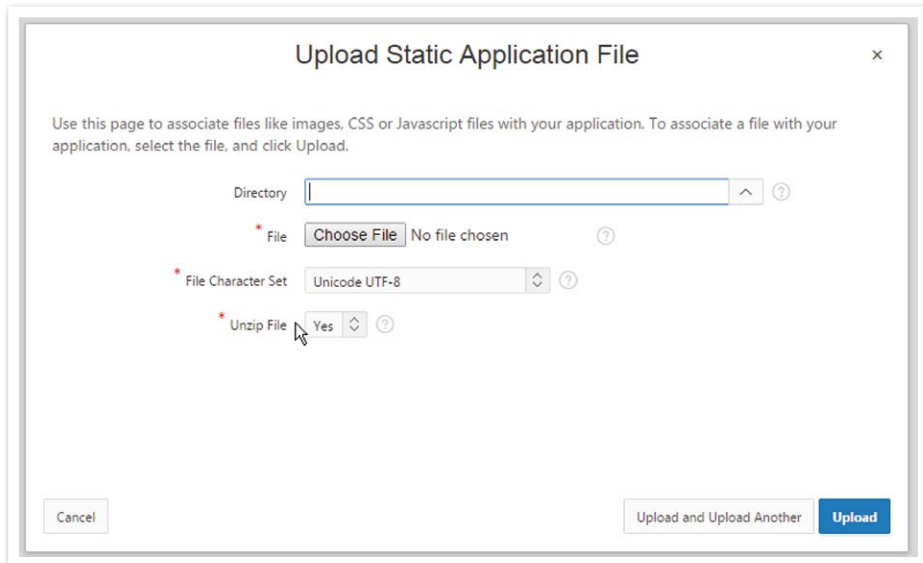


Abbildung 9: ZIP-Archive in den Apex-Workspace hochladen

```

declare
    l_zipfile      blob;
    l_files        apex_zip.t_files;
begin
    -- Hochgeladenes ZIP-File holen
    select blob_content into l_zipfile
    from apex_application_temp_files
    where name = :PX_UPLOADED_FILE;

    -- Inhalte abrufen
    l_files := apex_zip.get_files(l_zipfile);

    -- Inhalte auspacken und in Tabelle speichern
    for i in 1..l_files.count loop
        insert into my_files (file_name, content) values (
            l_files(i),
            apex_zip.get_file_content(l_zipfile, l_files(i))
        );
    end loop;
end;
```

Listing 1: PL/SQL-Prozesscode zum Auspacken einer hochgeladenen Datei

Fazit

Von den vielen neuen Funktionen in Apex 5.0 stehen vor allem der Page Designer und das Universal Theme im Mittelpunkt – das ist logisch; schließlich beeinflussen diese den Entwickler am meisten. Allerdings sind darüber hinaus viele neue Funktionen enthalten, die den baldigen Einsatz von Apex 5.0 absolut rechtfertigen.

Weitere Informationen

1. Apex im OTN: <http://www.oracle.com/technetwork/developer-tools/apex/overview/index.html>
2. Neue Funktionen in Apex 5.0: <https://apex.oracle.com/pls/apex/f?p=65339:1>
3. Deutschsprachige Apex Community Webseite: https://blogs.oracle.com/apexcommunity_deutsch



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

Enterprise Apex bei Vodafone – eine Erfolgsgeschichte

Tobias Strohmeier, MT AG

Apex wird meist mit kleinen Anwendungen und Projekten in Verbindung gebracht. Immer mehr größere Unternehmen fassen aber mittlerweile den Mut, dieser Technologie eine faire Chance für Größeres zu geben. Ein konkretes Projekt bei Vodafone zeigt, wo die Stärken liegen und welche Hindernisse es zu überwinden gilt.

Alles beginnt mit einer Datenbank am Rande ihrer Kapazität im Jahr 2013 im Financial Clearing House (FCH) des Telekommunikationskonzerns Vodafone in Düsseldorf. Ein Teil des Tagesgeschäfts besteht aus der Verarbeitung von Wholesale-Roaming-Rechnungen internationaler Netzbetreiber. Sie beinhalten abrechnungsrelevante Informationen, die durch die Nutzung von Endgeräten im fremden Netz verursacht werden.

Im Zuges eines Offshorings nach Indien häuften sich die Abstürze einer Datenbank. Diese beinhaltet nichts Geringeres als genau diese Rechnungen. Eine Ursache war schnell gefunden; die Datenbank hatte ihre limitierte Kapazität von 2 GB erreicht – ein akutes und massives Problem, dessen Lösung alles andere als trivial war. Unmittelbar war allen Beteiligten klar, dass eine Migration unumgänglich war.

Oracle 11g stand bereits zur Verfügung, sodass die Entscheidung für eine Ziel-Datenbank naheliegend war.

Was sollte jedoch mit den zahlreichen existierenden Anwendungen geschehen, die ebenfalls migriert werden mussten? Ein Gedanke lag nahe: „Lasst uns Apex nutzen. Das ist bereits mit der Datenbank vorinstalliert und kostet nichts.“ Weitere positive Argumente folgten, insbesondere zum geringen Entwicklungsaufwand. Aber ist Apex auch der Komplexität und dem gleichzeitigen Zugriff durch die vielen Nutzer gewachsen?

Migration der Daten

Ein entscheidender Vorteil war die bereits vorhandene Trennung von Frontend und Daten in dem bestehenden System. Ein erster großer Schwung von Tabellen konnte so sehr komfortabel nach Oracle

migriert werden. Gleichzeitig nutzte man diese seltene Möglichkeit, um die Datenmodelle auf den Stand der Zeit zu bringen und sinnvoll zu normalisieren. Vollständig reorganisiert und auf die veränderten Anforderungen angepasst, wurden die neuen Konto-Informationen beispielsweise um IBAN und BIC erweitert. In den bestehenden Frontends wurden die Tabellen-Links nun auf die migrierten Tabellen in der Oracle-Datenbank umgelenkt. Die Anwendungen konnten so erst einmal unverändert bleiben, bevor sie dann schrittweise in Apex nachprogrammiert wurden.

Aller Anfang ist schwer

Noch bevor überhaupt ein Gedanke an die erste Anwendung im Raum stand, musste ein Grundgerüst aufgebaut werden. In kleineren Projekten ist es selten notwendig, beispielsweise ein zentrales Rechtekonzept

The screenshot shows a complex data entry form with the following sections:

- Header Fields:** Service Code (VOICE), Client (Vodafone D2), Partner, EVO Creditor -Nr., VAT-Rate (0,19), Due in Days (30 Days), Decimals (4), VAT-ID-Nr.
- Invoice Information:** Invoice Reference Nr. (2655), Invoice -Date (01.11.2013), Received (02.11.2013), Due on (02.12.2013), Block Payment, Main Document Type (ER).
- Document Archive:** Document Archive Name (DEUD2_E), Invoice period from / to (01.10.2013 / 31.10.2013), Total Events (2.611.924), Total Minutes (8.312.640).
- Financial Summary:** Total (150.730,89), Net Amount (150.730,89), VAT Amount (28.638,87), Gross Amt (179.369,76).
- Qualifiers and Billing:** Sort Nr. (1), Billing-Period From / To (01.10.2013 / 31.10.2013), POP, PDI, and Tariff Zone Qualifiers.
- Buttons:** Close Form, New Invoice, Save Invoice, Rechnung wurde belastet!, Rechnung prüfen, Rechnungsabwicklung, Clear for New Record, Import UDRP Invoice Detail File (highlighted in red), Write Record to Disk, Show tariff updates needed, Update VF Tariff in Invoice Details.

Abbildung 1: Erfassungsmaske in der ursprünglichen Anwendung

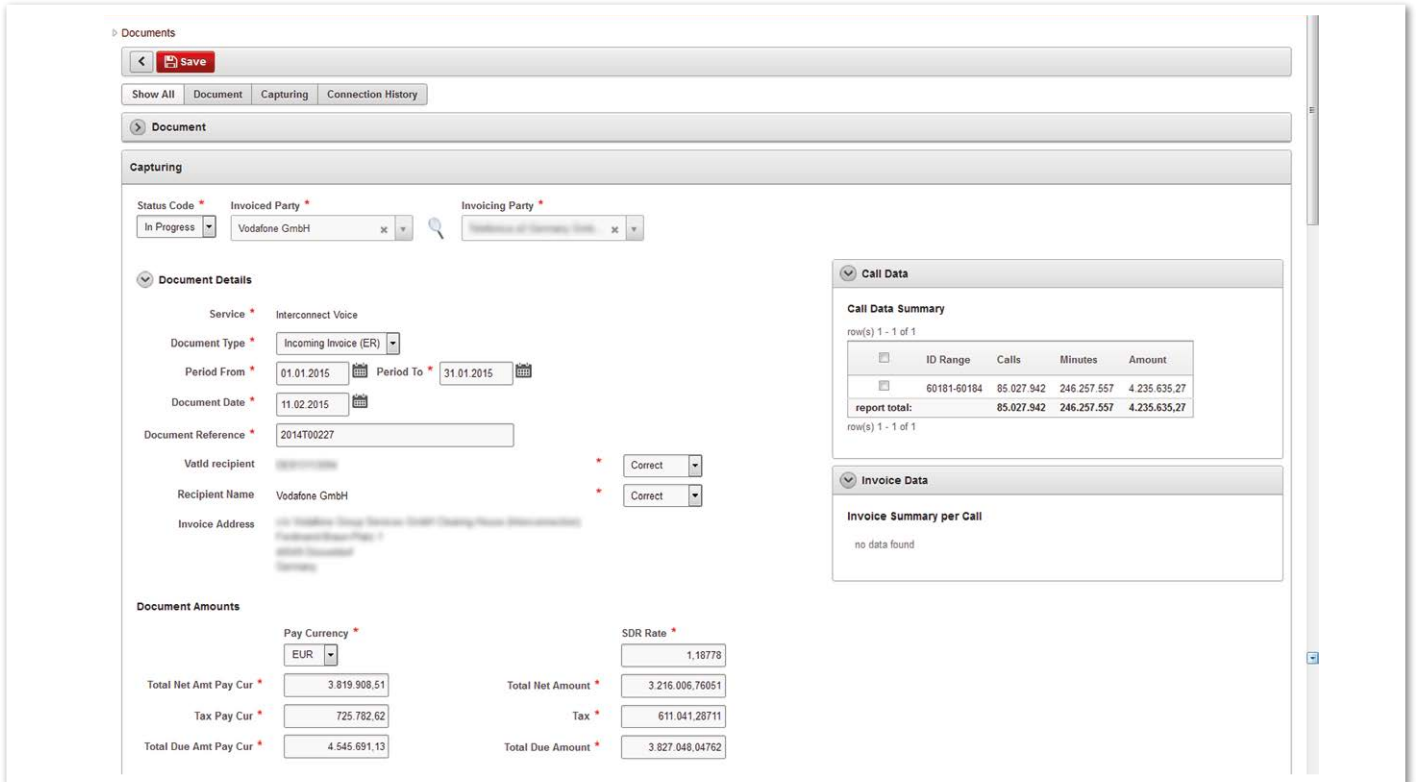


Abbildung 2: Übersichtlichere Erfassungsmaske in Apex

zu entwerfen, da von Apex zur Verfügung gestellte Funktionen meist ausreichen. Im FCH wurden weiterhin das existierende LDAP zur Nutzerverwaltung eingebunden, ein flexibler Navigationsbaum mit JSTree implementiert und ein Corporate-Design-Template samt Qualitätssicherung gestaltet. Die erste Anwendung war damit unmittelbar gefunden und schnell umgesetzt – eine Administration für die Verwaltung eben genau dieser Daten.

Es häuften sich Beschwerden über Wartezeiten, die das alte Frontend bei der Erfassung von Eingangs-Rechnungen verursachte. Ein Problem bei der Oracle-Datenbank? Nein, eher ein Netzwerk-Problem. Viele Nutzer befinden sich nicht vor Ort, sondern loggen sich remote mittels VPN in das System ein. Jeder Verarbeitungsschritt erforderte daher das Laden von Unmengen an Daten aus der Datenbank, damit diese Client-seitig zur Verfügung standen. Die erste zu migrierende produktive Anwendung war damit identifiziert, um dieses Problem zu beseitigen.

Die erste Anwendung in Apex

Nicht nur bei den Daten, sondern auch bei den Anwendungen nahm man sich im FCH Zeit, die eigentlichen Prozesse zu

hinterfragen und zu optimieren, um keine Kopie der vorhandenen Anwendung zu erhalten. Die User Experience stand hier klar im Vordergrund (siehe Abbildung 1).

Kurz vor der Umstellung beanspruchte die Erfassung einer einzelnen Rechnung mehrere Minuten Zeit – hauptsächlich bedingt durch die beschriebenen Warte-

zeiten. Bei Verarbeitungsfehlern war dies langwierig und kostenintensiv, da Daten zurückgesetzt und anschließend die Rechnung neu erfasst werden musste. Im Extremfall dauerte die Erfassung dadurch bis zu einem vollen Arbeitstag. Die Zurücksetzung wurde mit einer einfachen E-Mail an einen Administrator beauftragt; ein Um-

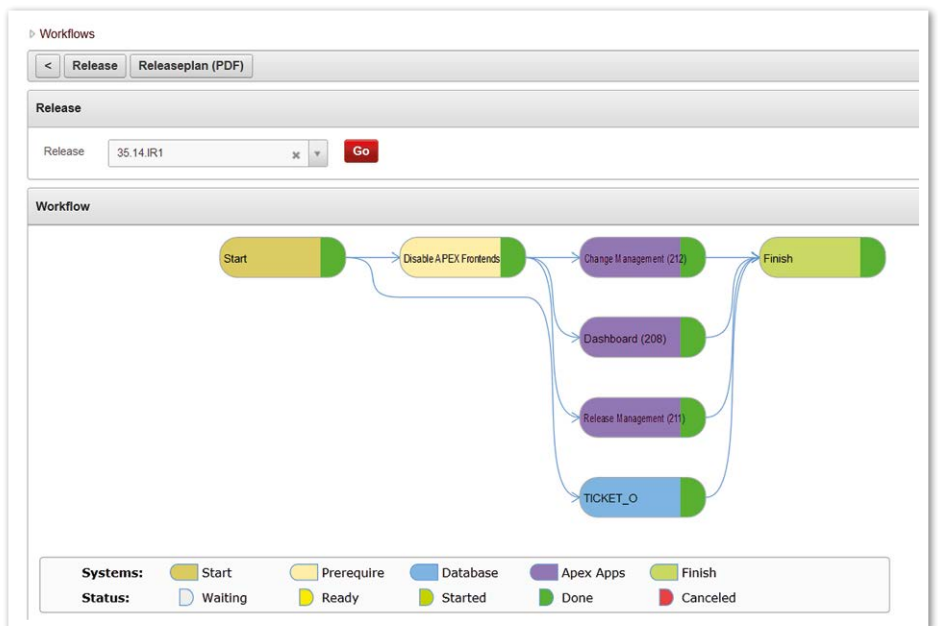


Abbildung 3: Interaktiver Deployment-Workflow in Apex

stand, der eine Auswertbarkeit unmöglich machte.

Um dieses Manko zu beseitigen, wurde die Implementierung eines Ticket-Systems beschlossen, mit dem Incidents und Service Requests getrackt werden konnten. Beide Anwendungen gingen dann im September 2013 produktiv, wobei das Deployment hier noch manuell erfolgte. Jedes Datenbank-Objekt musste einzeln und damit sehr zeitaufwändig aus der Entwicklungsumgebung in die Produktion übernommen werden. Doch diese Maßnahme reduzierte zur Überraschung Aller die Zeit für die Erfassung einer Rechnung um ein Vielfaches (siehe Abbildung 2).

Ein wesentlicher Faktor für diesen Effekt war, dass Apex nur die sichtbaren Daten an den Client sendet und somit die zu transferierende Datenmenge minimiert. Die Bereitstellung der Daten geschieht bereits innerhalb der Oracle-Datenbank, was einen entscheidenden Unterschied zur abgelösten Anwendung darstellt. Das aufgeräumte

und klar strukturierte Frontend fand bei den Nutzern ebenfalls sehr großen Zuspruch.

Die nun ermittelbare hohe Anzahl an Incidents übertraf die Erwartungen deutlich. Diese Informationen waren für das Management enorm wichtig, da sich dadurch Themenschwerpunkte mit hohen manuellen Aufwänden identifizieren ließen. Darüber hinaus konnten nun Lösungen konkret bereitgestellt und das Personal geschult werden; dies führte zu einer deutlichen Entlastung der Administratoren.

In den Wochen darauf wurden weitere Anwendungen migriert. Nach und nach verbesserte man dabei den Deployment-Prozess. Jede Anwendung lässt sich nun mittels Skript einrichten, sodass manuelles Kopieren der Datenbank-Objekte für den Administrator entfiel. Mit der Einführung von Subversion war es möglich geworden, auch einzelne Versionen einer Anwendung zu historisieren und nachzuhalten. Ein weiterer Umstand erschwerte die Deployments jedoch zunehmend.

Einführung von ITIL

Anfang 2014 hatte bereits eine gute Handvoll an Apex-Applikationen erfolgreich den Weg in die Produktion gefunden. Bedingt durch Abhängigkeiten zwischen den Applikationen und ihren zugehörigen Datenbank-Objekten ergaben sich immer wieder Konflikte während des Deployments. Auch fanden diese in unregelmäßigen Abständen und spontan beziehungsweise ungeplant statt. Außerdem wurde die Entscheidung zur Einführung der ITIL-Prozesse „Release-„ und „Changemanagement“ getroffen. Hierbei orientierte man sich stark an den Richtlinien, die ITIL vorschlägt und implementierte dies mit minimalen Anpassungen, die speziell auf das FCH zugeschnitten waren. Dies war ohne Weiteres möglich, da beide Anwendungen von Grund auf neu entwickelt wurden.

Apex selbst besitzt mit dem Team Development bereits ein mächtiges Tool, das genau diese Dinge anspricht. Aufgrund des implementierten Rechtekonzeptes und be-

Flexible SLAs. Weil Ihre IT einmalig ist.

dbi FlexService
ISO 20000

Unsere kosteneffiziente Wartungsverträge für Datenbanken & Middleware passen sich Ihren IT-Bedürfnissen an. Berechnen Sie jetzt Ihren SLA!

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

dbi-services.com/SLA



Infrastructure at your Service.

dbi services

stimmter Attribute beziehungsweise Abhängigkeiten konnte diese Lösung jedoch nicht adaptiert werden. Das Deployment ist durch spezielle Funktionen gezielt unterstützt. So werden jeweils zu Beginn und zum Ende E-Mail-Benachrichtigungen an alle Nutzer versendet. Die Deployer nutzen zudem einen interaktiven grafischen Workflow zur Unterstützung ihrer Arbeit (siehe Abbildung 3).

Während das allererste Release noch mehrere Stunden Aufwand und Produktionszeit verursachte, dauern sie jetzt meist weniger als dreißig Minuten. Mithilfe des Change Managements kann jeder Nutzer nun Änderungswünsche einreichen. Diese werden erst fachlich, dann technisch bewertet, bevor das Management entscheidet, ob eine Umsetzung stattfindet oder nicht. Das Release Management ist wiederum mit dem Change Management und auch mit dem Ticket System verbunden, sodass jederzeit Transparenz herrscht und erkennbar ist, wann eine Änderung oder ein Service Request umgesetzt und final eingerichtet worden ist.

Eine weitere wichtige Veränderung verbesserte dann nochmals deutlich die Geschwindigkeit und die Qualität neuer Releases – die Einführung einer Test-Umgebung. Neben Entwicklungs- und Produktions-Umgebung wurde diese ein wichtiges Verbindungsstück. Dedizierte Nutzer können Vorab-Versionen neuer oder geänderter Applikationen testen. Ebenso ermöglicht sie beim Deployment, Fehler in Skripten vor dem eigentlichen Release zu erkennen. Stressiges Debugging gehört damit der Vergangenheit an.

Produktivitätssteigerung dank Enterprise Apex

Problemlos konnten in den folgenden Monaten weitere wichtige Produktions-

Anwendungen migriert werden, unter anderem betraf dies die Zuordnung von Eingangszahlungen zu Ausgangsrechnungen. Auch hier wieder mit einem unglaublichen Erfolg. Optimierungen in den Prozessen führten nämlich dazu, dass Rechnungen und Zahlungen nun teilautomatisch zugeordnet werden konnten – eine große Entlastung für die Anwender.

Die Geschwindigkeit, mit der die Applikationen in Apex migriert beziehungsweise neu entwickelt werden, begeistert zunehmend andere Abteilungen, mit der Konsequenz, dass diese ihre neuen Projekte in Zusammenarbeit realisieren. Darüber hinaus setzen nun sogar auch andere Vodafone-Tochtergesellschaften auf das FCH-Team und damit auf Apex bei der Umsetzung internationaler Projekte. Grund genug für die Bezeichnung „Enterprise Apex“. Was bedeutet das für die Anwender? Zum einen werden Ressourcen frei, indem Prozesse optimiert und Wartezeiten reduziert werden. Auf der anderen Seite kommen, bedingt durch neue Projekte, zusätzliche Aufgaben hinzu, die es zu bewältigen gilt. Ohne die Einführung von Apex wären die gestiegenen Anforderungen nicht zu realisieren gewesen.

Spannende Zukunft

Aktuell stehen weitere interessante Anwendungen kurz vor der Fertigstellung, unter anderem ein Portal, das den ehemaligen SharePoint ablösen wird. Es hat zur Aufgabe, Dateien möglichst einfach und automatisiert zwischen Abteilungen auszutauschen und zu dokumentieren. Notwendiges manuelles Kopieren von Dateien in den SharePoint entfällt dadurch komplett, da die Daten bereits in der Datenbank zur Verfügung stehen.

Die Implementierung spezieller Workflows reduziert die manuellen Tätigkeiten

zusätzlich auf ein Minimum. Bedingt durch die diversen neuen Projekte wird auf diesem Weg auch das bestehende Rechtssystem erweitert. Bis dato bestand dieses ganz klassisch aus Rechten und Rollen. Neue Merkmale wie Organisation und Abteilung werden nun das Konzept weiter verfeinern. Das ist zwingend notwendig, da Anwender aus allen Kontinenten mit dem aufgebauten System arbeiten werden.

Fazit

Oracles Apex hat dem FCH geholfen, sehr erfolgreich und effizient ein großes und überaus komplexes Migrationsprojekt zu bewältigen. Zwei Dutzend neue Anwendungen haben mittlerweile fast alle alten abgelöst und führten zu strukturierterem Arbeiten und der Bewältigung gänzlich neuer Aufgaben. Durch die Akquise neuer Projekte mit internationalem Charakter befindet sich Apex in einem Aufwärtsschwung unerwarteten Ausmaßes. Kurz gesagt, die Anwender sind begeistert und die Entwickler freuen sich bereits auf die neue Version 5.0, mit der die Entwicklung mit Apex noch produktiver wird.



Tobias Strohmeier

tobias.strohmeier@mt-ag.com

Oracle Social Cloud jetzt mit Instagram und Weibo Unterstützung

Oracle Social Relationship Management (SRM) bietet ab sofort weitreichende Interaktions- und Analyse-Funktionen für die Foto- sowie Video-Sharing-Plattform Instagram und den Microblogging-Dienst Weibo. Somit ist eine Beobachtung, Analyse und Ansprache von Nutzern in zwei der weltweit

am stärksten wachsenden sozialen Netzwerken möglich.

Mit der neuen Funktionalität für Instagram schöpfen Unternehmen das volle Potenzial eines bei jungen Erwachsenen besonders beliebten Netzwerkes aus. Die bedeutende Rolle von Bildern im interak-

tiven Austausch ist gleichermaßen Wachstumsmotor und Fundament des Erfolgs der Foto- und Video-Sharing-Plattform. Mehr als die Hälfte der im Internet aktiven jungen Erwachsenen zwischen 18 und 29 Jahren nutzt das Netzwerk bereits, Tendenz steigend.

Apex im DWH/BI-Umfeld bei der GFKL

Kai Donato, MT AG

Die schnellen Ergebnisse, die man mit Apex und mithilfe des Rapid Prototyping erzielen kann, sind häufig der Anlass für den Einsatz dieser Technologie. Wie dieser Praxisbericht zeigt, gilt dies auch im DWH/BI-Umfeld.

Die GFKL Financial Services AG ist einer der führenden Dienstleister für Forderungsmanagement in Deutschland. Die rund 950 Mitarbeiter betreuen ein Forderungsvolumen von derzeit circa 16,6 Milliarden Euro. Das Angebot der GFKL richtet sich an Handelskonzerne, Banken, Versicherungen, Energieversorger, Telekommunikationsunternehmen und öffentliche Institutionen ebenso wie an kleine und mittelständische Unternehmen. Die Kompetenz der GFKL im Forderungsmanagement umfasst ein breites Portfolio mit hohem Qualitätsanspruch. Die Ratingagentur Standard & Poor's verlieh ihr dafür das höchste Rating als Servicer: „Strong, Outlook Stable“.

Das neu konstruierte Data-Warehouse des Unternehmens stellt eine enorme Menge an Daten bereit, die in Zukunft in Form eines BI-Tools analysiert und unter anderem grafisch aufbereitet werden sollen. Die Kollegen des DWH-Teams haben bereits vordefinierte Szenarien/Berechnungen als Views in der Datenbank zum Abruf angelegt und bilden hiermit das Fundament sowohl für Apex als auch für ein anstehendes BI-Tool.

Ziel dieses Projekts war es, die Zeit bis zum Einsatz eines BI-Tools zu überbrücken und schnellstmöglich erste Ergebnisse in Form von visualisierten Daten zu präsentieren. Der Zeitraum dieses Projekts sollte zweieinhalb Monate umfassen und es sollten neunzehn vordefinierte Szenarien in grafischer Form ausgearbeitet werden. Die Apex-Applikation sollte schnell und zielführend eine Rolle im Unternehmensreporting für die definierten Objekte übernehmen, um damit die Vorzüge des neu geschaffenen Data Warehouse aufzuzeigen.

Die Stärken von Apex

Die Vorteile, die Apex in diesem Zusammenhang mit sich bringt, liegen zum ei-

nen darin, dass der Einsatz bei einer vorhandenen Oracle-Datenbank-Lizenz kostenlos und die Implementierung in die eigene Infrastruktur keinen großen Aufwand darstellt. In diesem Projekt war bereits eine Apex-Umgebung vorhanden, die schon für kleinere administrative Zwecke genutzt worden war. Die Tatsache, dass man mit Apex in kürzester Zeit erste Ergebnisse erzielen kann, erleichtert die Abstimmung mit dem Kunden und die Erstellung von funktionalen Prototypen, die zur weiteren Orientierung genutzt werden können.

Die Auswahl der Charting-Option fiel in diesem Projekt relativ leicht, da die gewünschte grafische Darstellung der Daten unter Umständen auch auf mobilen Endgeräten verfügbar gemacht werden soll. Dies ist allein schon aufgrund der Restrik-

tionen in der Produktlinie von Apple nur mit HTML5-Charts flächendeckend zu realisieren. Zusätzlich zu der Kompatibilität sagten die HTML5-Charts dem Kunden visuell deutlich besser zu.

Grundgerüst der Applikation

Zum Entwickeln der Applikation wurde das Apex-Theme 25 verwendet. Dieses bietet von Haus aus responsive Fähigkeiten und eignet sich somit sowohl für Desktop-Applikationen als auch für den Einsatz auf mobilen Endgeräten. Um eine inhaltliche Kategorisierung der verschiedenen Szenarien vorzunehmen, hat man sich für das Plug-in „Enkitec NavBar“ entschieden, das auf Basis einer Apex-Liste in den Shared Components funktioniert.

Da der Benutzerkreis der Applikation erweitert werden kann, ist anzumerken, dass

The screenshot shows the 'Region Attributes' configuration window for the 'NavBar' region. At the top, there are tabs for 'Region Definition' and 'Region Attributes', with 'Region Attributes' selected. Below the tabs, the 'Region Name' is set to 'NavBar'. There are 'Cancel' and 'Apply Changes' buttons. Below this is a 'Settings' section with a 'Show All' button and a 'Settings' header. The settings are as follows:

* List Name	NavigationBar List
Display Brand	Yes
* Brand Label	DWH
Brand URL	f?p=&APP_ID.:1:&APP_SESSION.::&DEBUG.::::
Make Responsive	Yes

Abbildung 1: Einstellungen im Plug-in „Enkitec NavBar“

es über die in der Navigationsliste hinterlegten Conditions möglich ist, die Sichtbarkeit einzelner Menüpunkte zu steuern. So lässt sich ein dynamisches Menü erstellen, das ausschließlich die Menüpunkte anzeigt, die der Benutzer auch einsehen darf (siehe *Abbildung 1*). In den Einstellungen der „Enkitec NavBar“ ist es ebenfalls möglich, die responsive Eigenschaft der Navigation zu aktivieren. Dadurch wird das Menü beim Aufruf mit einem mobilen Endgerät in einer bedienbaren Form angezeigt.

Die Datenbasis, auf der die Apex-Applikation arbeitet, wird durch ein Oracle Data Warehouse und alle relevanten Daten in Form von vordefinierten Views bereitgestellt. Aus Performance-Gründen kamen Materialized Views zwischen Apex und den Views zum Einsatz. Diese gewährleisten einen schnelleren Zugriff auf die Daten, die visualisiert werden sollen, da die teilweise komplexen Berechnungen nur einmalig nach dem Bezug neuer Daten durchgeführt werden. Mithilfe des täglichen Ladelaufs des Data Warehouse werden die Materialized Views mit neuen Daten befüllt, sodass Apex stets auf die aktuellsten Daten des Tages zugreift.

Diagramme

Nachdem die Wahl der einzusetzenden Diagramm-Werkzeuge getroffen war, konnte die Umsetzung beginnen. Da die anzuzeigenden Daten in vielen Fällen nach Tochtergesellschaft gruppiert sind, musste für jede Gesellschaft eine Series angelegt werden (siehe *Abbildung 2*).

Um die enorme Menge an Daten übersichtlich darzustellen, war es notwendig, ver-

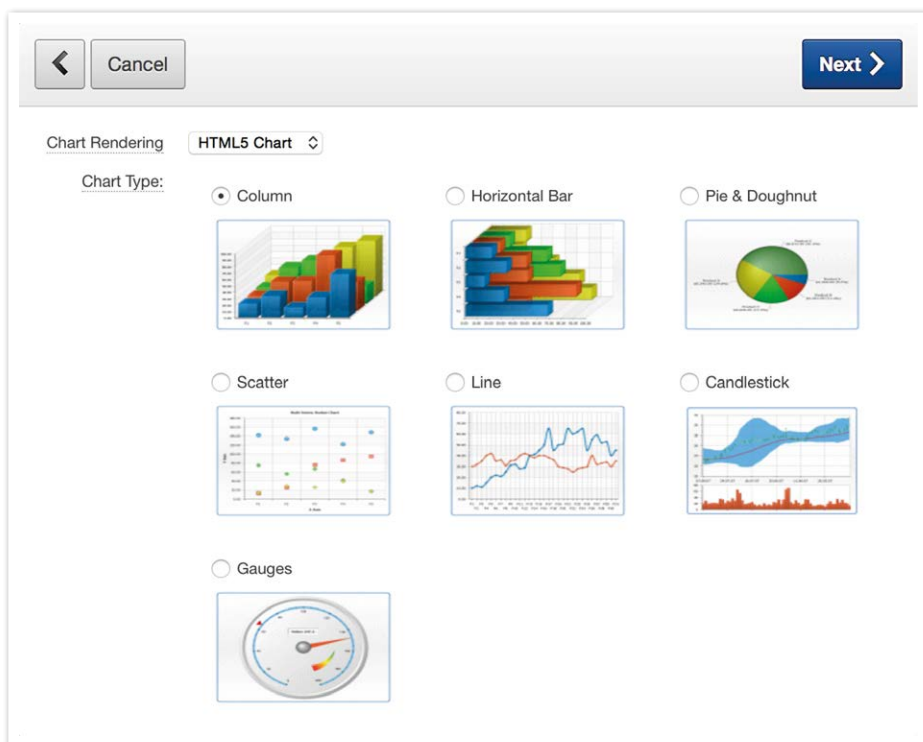


Abbildung 2: Apex-Ansicht für die Erstellung von Diagrammen

schiedene Drill-Down-Stufen zu implementieren. Die Diagramme, die die jeweiligen Daten aufbereitet anzeigen, können über eine Select-List oberhalb des Diagramms auf verschiedenen zeitlichen Ebenen betrachtet werden. Je nach Kontext kann man ein konkretes Jahr oder auch eine Zeitspanne auswählen. In einigen Diagrammen ist es auch möglich, bis auf eine stundengenaue Ansicht zu wechseln (siehe *Abbildung 3*).

Aus technischer Sicht funktioniert ein Drill-Down mithilfe einer Dynamic Action, die das Diagramm und somit auch die da-

hinterstehende Auswahl der Daten neu lädt. Dazu war es notwendig, eine dynamische „WHERE“-Klausel an das „SELECT“-Statement der jeweiligen Series anzuhängen, um die Daten, die visualisiert werden sollen, auf die ausgewählte Drill-Down-Ebene einzuschränken (siehe *Abbildung 4*).

Diagramme mit einer weiteren Y-Achse

Das Hinzufügen einer weiteren Y-Achse ist mit Apex möglich, indem man in den Eigenschaften eines Diagramms die

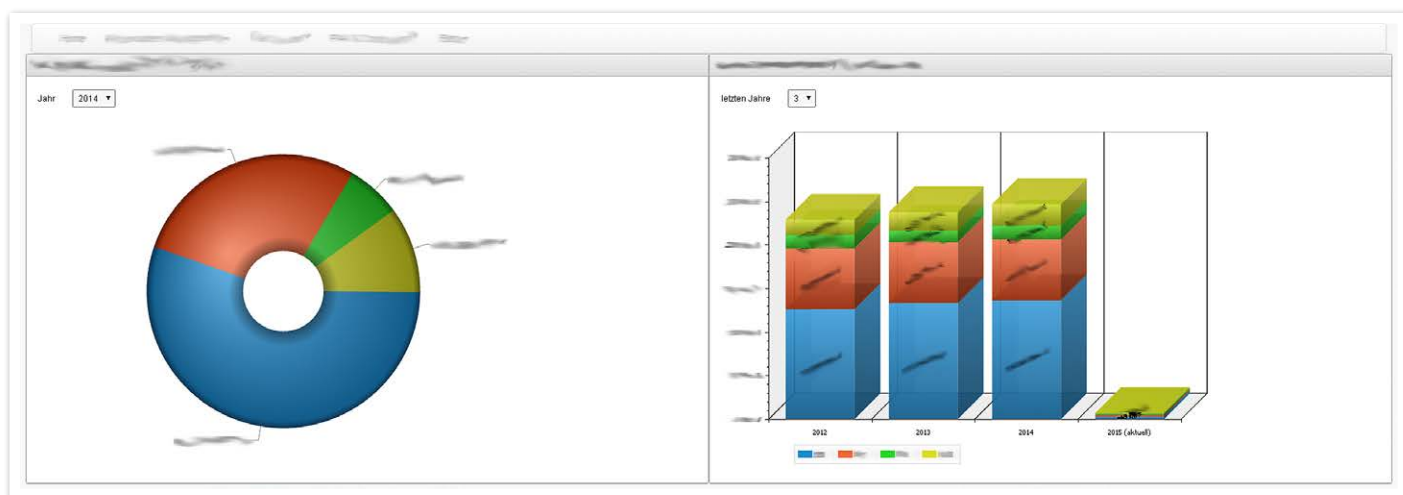


Abbildung 3: Drill-Downs in Form eines konkreten Jahres beziehungsweise der Anzahl der zu betrachtenden Jahre

The screenshot shows the configuration interface for an Apex chart. It has several tabs: 'Show All', 'Series Attributes', 'Series Query', 'Action Link', 'Conditions', 'Security', and 'Configuration'. The 'Series Attributes' tab is active, showing 'Chart Type: 3D Column', 'Series Name: Series 1', 'Series Type: Bar', and 'Sequence: 10'. The 'Series Query' tab is also visible, showing 'Query Source Type: SQL Query' and a text area containing the SQL query: 'SELECT null LINK, to_char(pricing_date) LABEL, CLOSING_VAL "clos" FROM database where YEAR = :P10_SELECT_LIST;'. A 'Build Query' button is at the bottom.

Abbildung 4: Beispiel für eine dynamische Drill-Down Query

Funktion „Multiple Y-Axis“ aktiviert. Es ist wichtig darauf zu achten, dass der zusätzliche Graph, der dem Diagramm hinzugefügt wird, in einer gesonderten Serie und unbedingt mittels Sequenz-Nummer an der zweiten Stelle eingeordnet wird (siehe Abbildungen 5 und 6). Bewusst fiel die Entscheidung für die Darstellung von mehreren korrelierenden Daten innerhalb eines Diagramms auf eine zwei-

dimensionale Darstellung, da es bei der Vermischung von 3D- und 2D-Graphen zu Problemen kommt.

Wie *Abbildung 7* zeigt, sind innerhalb der Diagramme „3D-Stacked-Columns“ (Gelb/Blau) und darüber 2D-Lines. Durch die Vermischung eines drei- mit einem zweidimensionalen Charting-Element entstehen Ungenauigkeiten in der Darstellung. An dieser Stelle sollte man deshalb

auf eine rein zweidimensionale Darstellung wechseln.

Erweiterung der Applikation um Lade-Indikatoren

Das Neu-Laden der Diagramme geschieht, ohne dass die gesamte Apex-Seite neu gerendert werden muss, und bringt hierbei einen asynchronen Prozess mit sich. Während der Anwender sich noch auf der aktuellen Seite befindet, wird ein Teilbereich neu geladen, dies inkludiert das Beziehen neuer Daten aus der Datenbank. Da sich dieser Prozess je nach Menge der Daten auch über mehrere Sekunden erstreckt, bekommt der Anwender in dieser Zeit keinerlei Hinweis darauf, dass im Hintergrund gerade ein asynchroner Prozess läuft.

Diese Problematik lies den Autor auf ein JavaScript-Framework („Pace“, siehe „<http://github.hubspot.com/pace/>“) zurückgreifen, das solche Prozesse automatisch erkennt und mittels Lade-Indikator innerhalb der Applikation anzeigt. Dies ist für sein Empfinden ein elementarer Bestandteil einer JavaScript-getriebenen Applikation. Wenn ein Benutzer einen langwierigen Prozess anstößt, muss er darüber informiert werden, dass Apex beziehungsweise vielmehr der Browser gerade aktiv und ein erneutes Klicken des Buttons oder einer anderen Schaltfläche nicht erforderlich ist.

The screenshot shows the configuration interface for an Apex chart. It has two main sections: 'Chart Series' and 'Display Settings'. The 'Chart Series' section has a table with two rows: 'Series 1' with the query 'SELECT null LINK, to_char(pricing_date) LABEL, CLOSING_VAL "Genorate" FROM EBA_DEMO_CHART_STOCKS where stock_code="GENO"' and 'Series 2' with the query 'SELECT null LINK, to_char(pricing_date) LABEL, CLOSING_VAL "Metro" FROM EBA_DEMO_CHART_STOCKS where stock_code="METR"'. The 'Display Settings' section has various options: 'Marker: Square', 'Style: Aqua Light', 'Background Type: Transparent', 'Include on Chart: Hints, Values, Labels, Multiple Y-Axis (checked), Group By Series, Major Ticks, Minor Ticks, Tick Labels, Invert X-Axis Scale, Invert Y-Axis Scale, Invert Scale, Overlay Y-Axis', and 'Show Grid: Y-Axis (selected)'. The 'Multiple Y-Axis' checkbox is highlighted with a red box.

Abbildung 5: Diagramm-Eigenschaften für das Hinzufügen eines zweiten Graphen

Apex für die Verwaltung von Stammdaten

Im Anschluss an die visualisierten Objekte war es möglich, ein Stammdaten-System zu realisieren, das auf der Basis von Apex

gerade eingeführt wird. Beim Ladeprozess des DWH werden an verschiedenen Punkten Werte-Mappings eingesetzt, um fachliche Bezüge, die auf Seiten der Tochterunternehmen genutzt werden, auf das

DWH der GFKL umzuschlüsseln. Um diese Zuordnungstabellen in der Datenbank zu verwalten, wurde eine eigenständige Applikation mit rund zweihundert Apex-Seiten entwickelt.

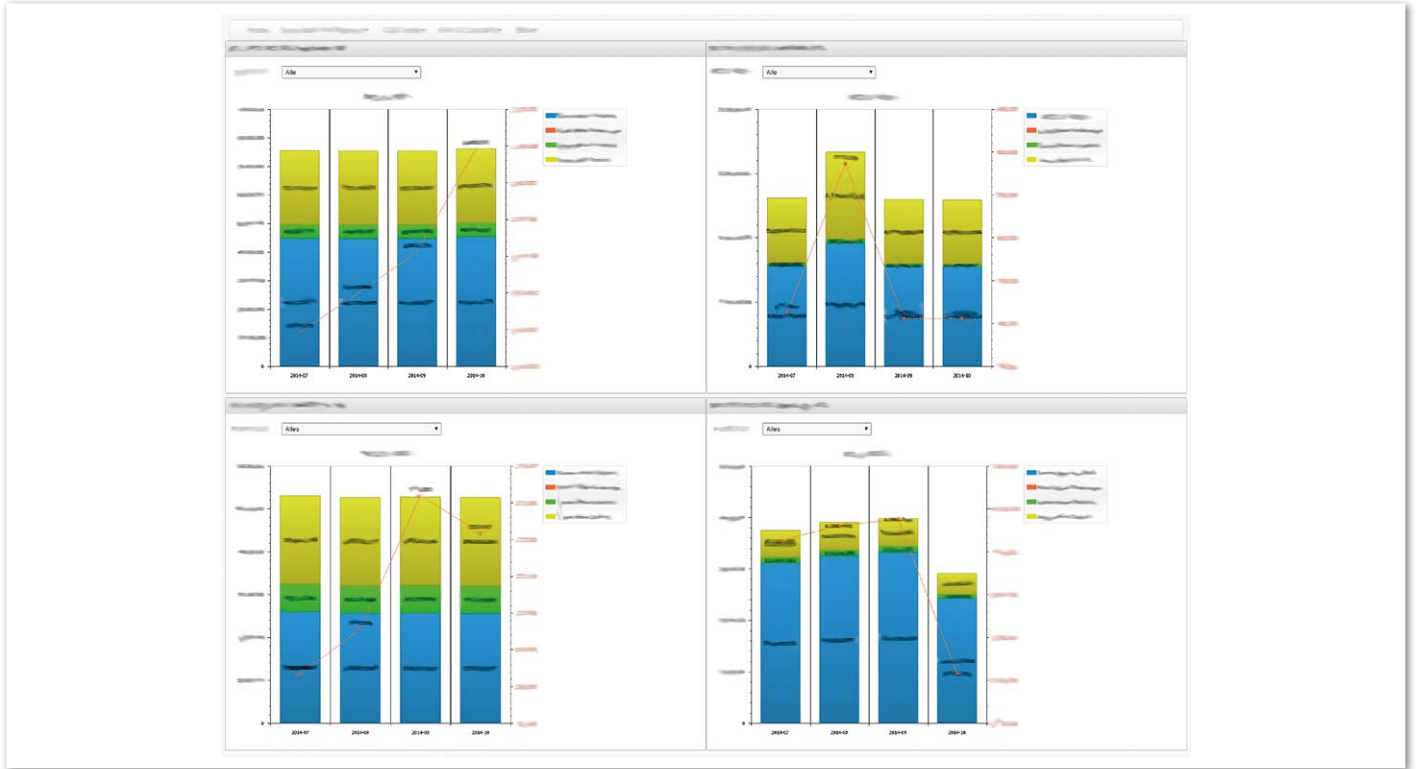


Abbildung 6: Zweidimensionale Darstellung mit einer weiteren Y-Achse

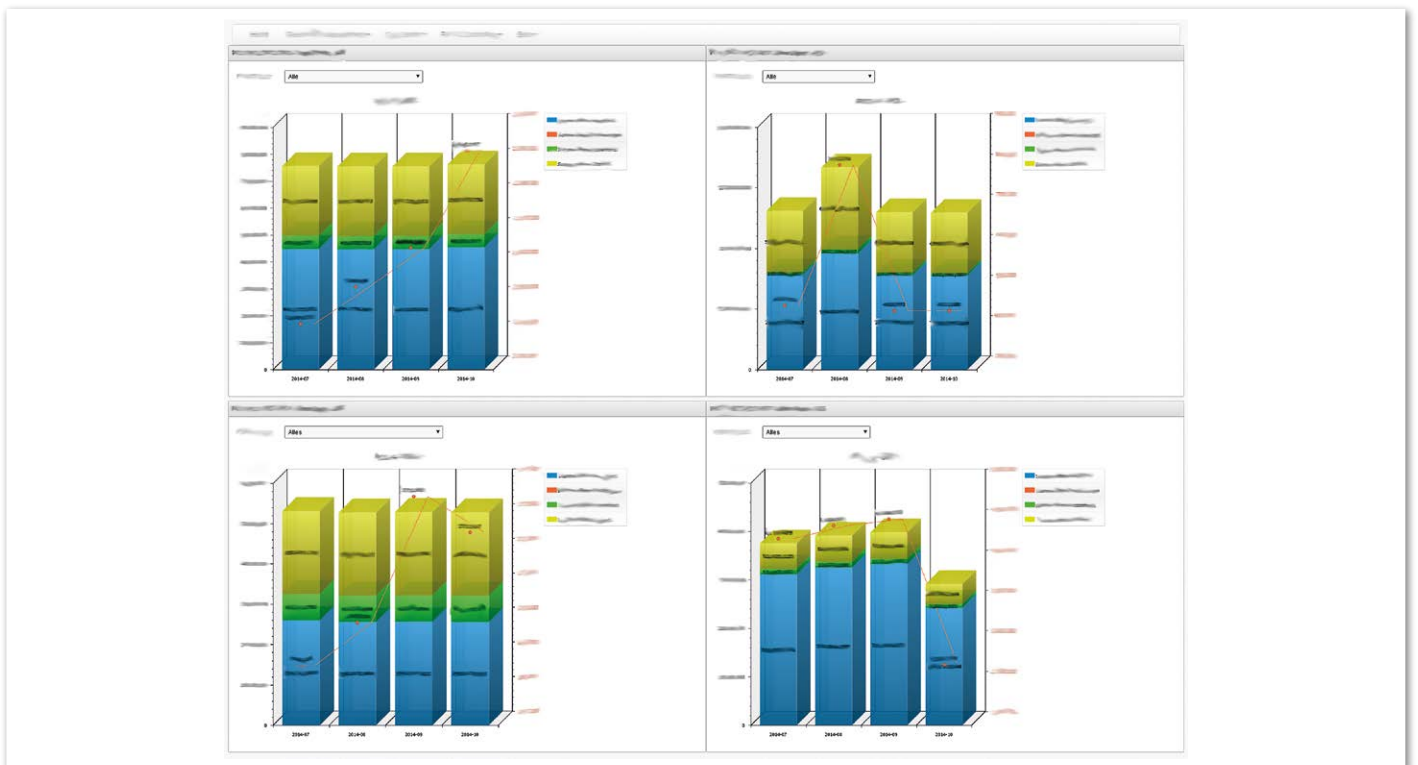


Abbildung 7: Dreidimensionale Darstellung mit einer weiteren Y-Achse.

Das Berechtigungskonzept

Ein Authentifizierungsschema auf Basis des bestehenden Lightweight Directory Access Protocol (LDAP) der GFKL und ein Autorisierungsschema, das darauf aufsetzt, stellen ein restriktives Berechtigungssystem dar. Durch die Implementation des Features „Virtual Private Database“ (VPD) konnte darüber hinaus der Zugriff auf das DWH eingeschränkt werden. Nach der Anmeldung eines Benutzers in der Apex-Applikation wird ein Session-Kontext an die Datenbank übermittelt, den anschließend die Prädikatsfunktion nutzt. Jede Abfrage, die der Benutzer dann über die Oberfläche (oder über ein SQL-Tool) tätigt, wird mit deren Hilfe und mit dem übermittelten Session-Kontext gefiltert. Darüber hinaus werden lediglich die Daten bereitgestellt, die für den Zugriff durch den Benutzer vorgesehen sind.

Fazit

Die Anforderung, schnellstmögliche Ergebnisse liefern zu können, konnte mit

Apex erwartungsgemäß sehr gut erfüllt und dem Kunden regelmäßig der Fortschritt präsentiert werden. Auch um umfangreiche und komplexe Sachverhalte aus einer Vielzahl von Daten aufbereitet darzustellen, ist nicht immer ein BI-Tool notwendig. Die kostenlose Apex-Umgebung ermöglicht es, mithilfe der umfangreichen Diagramm-Bibliotheken unterschiedlichste grafische Darstellungen zu erstellen.

Auch eine tabellarische Darstellung, die von Haus aus einen großen Funktionsumfang mitbringt, ist mit Apex im Handumdrehen zu verwirklichen. Dazu kommt der Interactive Report, über den der Endanwender seine Daten selber auswerten kann (Self-Services). So erlauben es die Standardfeatures des Interactive Report, ohne umfangreiche Programmierung des Entwicklers Gruppierungen und Aggregationen an den bereits angezeigten Daten vorzunehmen, zu speichern und zu exportieren. Auch das individuelle Anpas-

sen der Ausgabe wie Ausbeziehungsweise Einblenden von Spalten ist direkt auf der Oberfläche im Browser möglich.

Apex ist gut geeignet, um einige Aufgaben eines BI-Tools zu übernehmen, wird aber aufgrund der angeforderten Funktionalität und der benötigten Leistung, die ein BI-Tool liefert, nicht in der Lage sein, dieses vollständig zu ersetzen.



Kai Donato
kai.donato@mt-ag.com

documents.
done.
smart.

Exportieren Sie Ihre Daten in vollwertige Excel-, PowerPoint und Word-Dokumente – unmittelbar aus Ihrer Datenbank



Ad hoc oder jobgesteuert, einzeln oder in Serie, via API aus der Datenbank

Zentrale Konfiguration, detaillierte Zugriffskontrolle, hohe Sicherheit

Prozesse automatisieren, Effizienz steigern, Kosten sparen

Informationen, Videos, Trial-Version:
www.smaxt.com oder +49 (0)611 – 900 36 40

 smaxt

Excel, Office, PowerPoint und Word sind eingetragene Marken oder Warenzeichen der Microsoft Corporation in den Vereinigten Staaten und / oder anderen Ländern.



Erfahrungen eines Apex-Einsteigers

Rüdiger J. Schulz, DB Systel GmbH

Am Anfang war das PL/SQL-Package „http“. Damit konnte man in Verbindung mit einem Apache Webserver und dem Modul „mod_plsql“ bereits einfache Web-Anwendungen entwickeln. Später kam die „HTML-DB“ und mittlerweile ist daraus das sehr umfangreiche Apex geworden, das im Grunde nur auf puren, wenn auch sehr komplexen PL/SQL-Packages basiert. Oracle Apex ist unbestritten ein hervorragendes RAD-Werkzeug, um sehr schnell stabile, webbasierte Anwendungen zu erstellen.

Dieser Artikel beschäftigt sich mit den Einstiegsschwierigkeiten in Apex, dem Irrglauben, dass Apex ohne spezielles Wissen genutzt werden kann, und damit, wie sich die DB Systel GmbH diese kostengünstige Technik in der Abteilung „Small Solutions“ zunutze gemacht hat.

Anfangshürden und Stolpersteine

Die letzten Jahre meiner beruflichen Laufbahn hatte ich weniger mit der Entwicklung von Anwendersoftware zu tun. Meine Aufgaben waren vielmehr die Programmierung von Oracle-Datenbanken, das „shell“-Scripting und die Entwicklung kleinerer „Stand-alone“-Lösungen mit Java. Ich hatte also, vom Oracle-Datenbank-Kernel abgesehen, immer die direkte Kontrolle über meine Systeme. Mit Apex stand ich seit Jahren erstmalig wieder von einer Blackbox, die ich erst im Laufe der Zeit einigermaßen durchblicken sollte. Auch wenn die folgenden Darstellungen zunächst eventuell ne-

gativ wirken, sei jetzt schon gesagt, dass man jedes komplexere System nicht von Beginn an beherrscht. Genauso verhält es sich mit Apex: Erst nach längerer Verwendung zeigen sich die großen Stärken.

Ich fing also, wie jeder, der sich mit der Entwicklung von GUI-Systemen beschäftigt, mit der Erstellung eines Beispiels an. Eine einfache Stammdaten-Verwaltung basierend auf einer Datenbank-Tabelle sollte erst einmal reichen. Ich hatte das

Glück, als Grundlage eine Referenz-Anwendung verwenden zu können. Deren Installation lief reibungslos ab, sodass ich sofort mit der Entwicklung beginnen konnte. Zu dieser Referenz-Anwendung später mehr.

Vorweg gesagt: Es ist nicht immer leicht, die richtigen Stellen in der Apex-Entwicklungsumgebung zu finden, und wenn sich diese aufspüren lassen, geschieht es durchaus, dass man sich nach einer Wei-

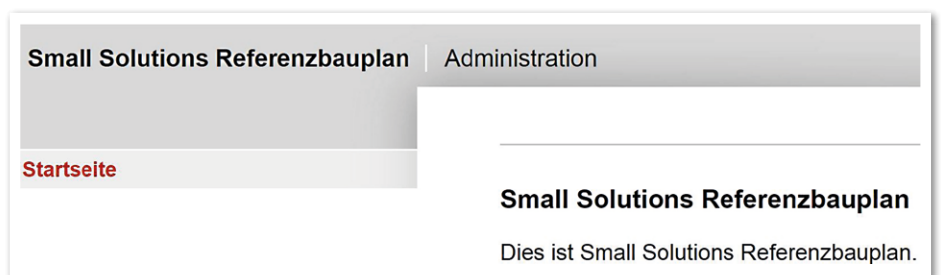


Abbildung 1: Anfangssituation

le fragt: „Wo stand das noch gleich?“ Häufig werden zu Beginn der Arbeit mit Apex-Optionen die günstigsten ganz einfach in der Fülle der Möglichkeiten übersehen. An drei Beispielen möchte ich diese Anfangsschwierigkeiten deutlich machen.

Beispiel 1: Menü-Führungen erstellen

Ich hätte gerne einen Haupt-Menüpunkt „Datenpflege“ und darunter zwei Menüpunkte „Datenpflege 1“ und „Datenpflege 2“. Das klingt erst einmal nicht schwierig. Ich vermutete, dass es mit der Verbindung zwischen dem Haupt- und dem Unter-Menüpunkt getan wäre. Da lag ich aber falsch ... *Abbildung 1* zeigt die Anfangssituation. Nun baue ich die Menüstruktur (siehe *Abbildung 2*).

Die Anwendung hat jetzt schon mal den Haupt-Menüpunkt „Datenpflege“. Also lege ich die Seiten „Datenpflege 1“ sowie „Datenpflege 2“ an und ordne diese den entsprechenden Menüpunkten zu (siehe *Abbildung 3*).

Das Ergebnis ist unbefriedigend, alle anderen Menüs sind nicht sichtbar, beim Klick auf „Datenpflege 1“ lande ich wieder bei der Ausgangssituation, nämlich der Startseite, und die neu erstellten Seiten sind nicht mehr erreichbar (siehe *Abbildung 4*). Nun habe ich versucht, unter „Edit Standard Tabs“ die Tabs den Seiten zuzuordnen. Immer noch kein Erfolg. Bis ich endlich einen Dialog fand (siehe *Abbildung 5*).

Nach weiterer Suche kam ich dann noch zu einem Dialog, der das gleiche Ergebnis brachte (siehe *Abbildung 6*). Des Pudels Kern war also, dass dem Haupt-Menüpunkt eine „Default“-Seite zugeordnet werden muss. Von meinen Kollegen ließ ich mir bestätigen, dass dieser Umstand noch heute immer wieder zu Verwirrungen führt, selbst bei sehr erfahrenen Entwicklern.

Beispiel 2: Änderung bestehender Seiten

Applikationen unterliegen mitunter Änderungen. Das zweite Beispiel betrachtet einen Stolperstein, der einem dabei häufig im Weg liegt. *Listing 1* zeigt eine einfache Struktur.

Auf der dazugehörigen Seite werden nun zunächst nur die Namen bearbeitet. Später wird mit „ALTER TABLE namen ADD vorname varchar2(100);“ noch das

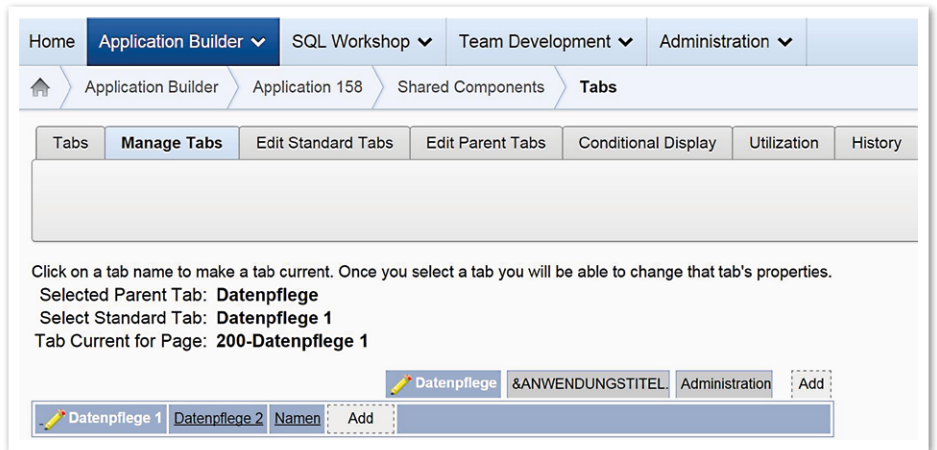


Abbildung 2: Definition der Tabs

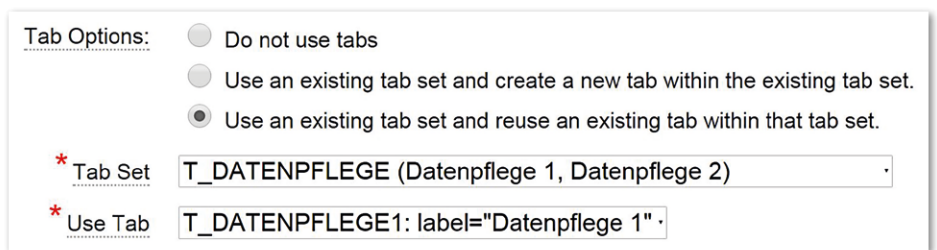


Abbildung 3: Zuordnung der Seite zu den Tabs

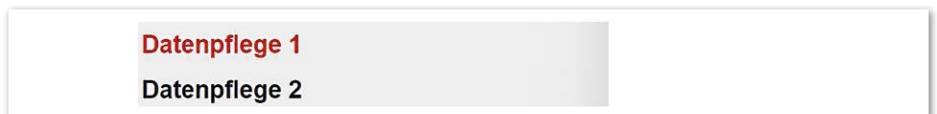


Abbildung 4: Erstes Ergebnis

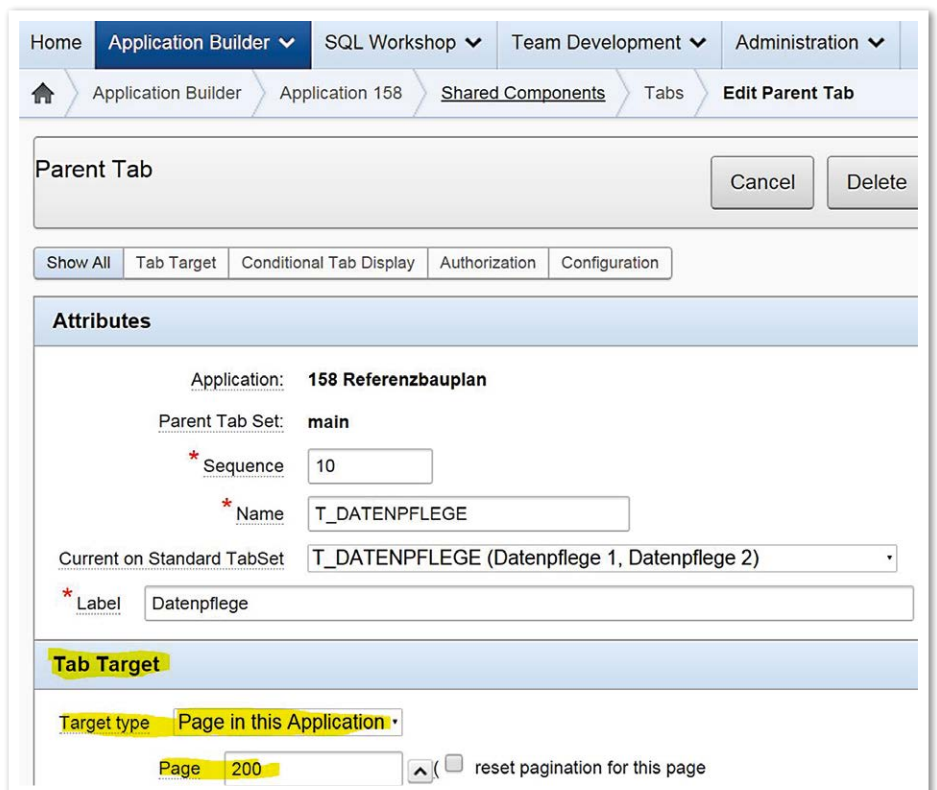


Abbildung 5: Die Lösung: Tab Target

Feld „VORNAME“ in die Tabelle eingefügt. In der Regions-Definition ist nun unter „Source“ das entsprechende SQL-Select-Statement auf „select ID, NAME, VORNAME from #OWNER#.NAMEN“ zu ändern. Der Report und die Bearbeiten-Seite werden damit um das Feld „VORNAME“ erweitert. Nun kommt der folgenschwere Anfängerfehler, der mir häufig passiert ist. *Abbildung 7* zeigt dazu die erweiterte und schon editierte Namensliste.

Ich will nun den Namen „Anton Meier“ in „Christian Meier“ ändern und direkt danach den Namen „Lehman“ (mit einem „n“) auf „Lehmann“ korrigieren (mit zwei „n“) (siehe *Abbildung 8*). Doch ab jetzt hat Herr Lehmann nicht mehr den Vornamen „Karl“, sondern „Christian“ (siehe *Abbildung 9*).

Es ist Folgendes passiert: Jedes Mal, wenn ein Name bearbeitet wird, erscheint im Feld „VORNAME“ immer der Vorname, der zuvor editiert wurde. Worin lag der Fehler? Ich hätte das Source-Used-Attribut (unter Page Item) von „Only when current value in session state is null“ auf „Always, replacing any existing value in session state“ ändern müssen. Übrigens ist auch das ein Fehler, der selbst erfahrenen Entwicklern noch unterläuft.

Beispiel 3: Apex-Fehlermeldungen

Auch Fehlermeldungen wie in *Abbildung 10* sind für den Anfänger sehr frustrierend. Ohne die Hilfe erfahrener Kollegen kommt man schwer damit klar. Das klingt zunächst einmal mehr nach einer Feindseligkeit, als dass man sich mit Apex anfreundet, aber das täuscht: Man sollte sich von diesen Umständen nicht entmutigen lassen. Irgendwann werden diese Schwierigkeiten keine mehr sein, wenn man sie durch Routine überwindet.

Know-how

Nun möchte mit dem Irrglauben aufräumen, dass kein spezielles IT-Wissen für die Entwicklung von Apex erforderlich sei. Das ist selbst dann nicht wahr, wenn man nur eine einfache Stammdaten-Verwaltung zusammenklickt. Spätestens bei der Auswahl einer Tabelle und der Bestimmung deren Primary Keys muss der Entwickler zwangsläufig wissen, was erstens ein Primary Key ist und wo zweitens diese Information zu finden ist (siehe *Abbildungen 11 und 12*).

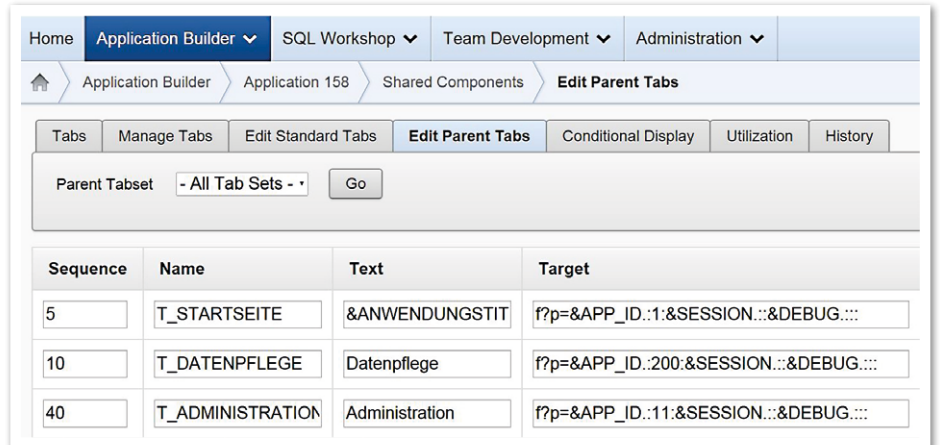


Abbildung 6: Lösungsalternative

```
create table namen
(
  id number primary key
  ,name varchar2(100)
);
create sequence seq_namen start with 1 increment by 1;
```

Listing 1

Also ist zunächst einmal ein Minimum an Datenbank-Know-how gefragt. Von der Modellierung eines der Anwendung zugrunde liegenden Daten-Modells und dem damit verbundenen theoretischen Wissen sehe ich hier völlig ab. Neben umfangreichen Beispielen, die man im Netz findet, ist das folgende Buch zur PL/SQL-Programmierung empfehlenswert: „Oracle PL/SQL Programmierung“, erschienen im Jahr 2014 bei O'Reilly.

Es ist darüber hinaus wichtig, einige HTML/CSS-Grundlagen zu beherrschen, denn ohne beispielsweise das Wissen, dass ein Link mit einem „<a href...>-Tag erstellt wird, kommt man zwar auch weiter, aber es hilft ungemein zu erkennen, dass die Darstellung unter anderem vom CSS abhängt. Die Website „selfhtml.org“ sei an dieser Stelle als Nachschlageort empfohlen.

Nicht ganz so entscheidend ist das Wissen um JavaScript, obwohl es äußerst nützlich sein kann. Ein Beispiel für einen hilfreichen Einsatz ist, wenn aus einer Tabelle mittels Checkbox ein einzelner Wert ausgewählt werden soll (siehe *Listing 2*).

Bei der Analyse von Seiten ist es teilweise ebenso notwendig zu wissen, warum sich eine Seite so verhält, wie sie sich verhält. Empfehlenswert für JavaScript-Einsteiger ist die Site „<http://www.w3schools.com/js>“.

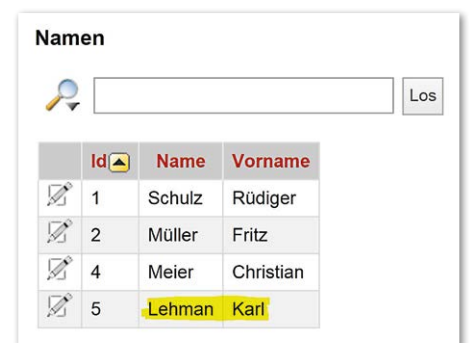


Abbildung 7: Liste der Namen, nun mit Vornamen

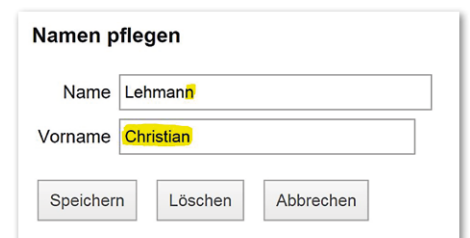


Abbildung 8: Namensänderung

Schließlich ist das Wissen um die Architektur von Web-Anwendungen wichtig. Man muss zwar nicht immer selbst bei der Konfiguration der Komponenten Betriebssystem, Datenbank und Webserver oder einer Firewall Hand anlegen, aber

es ist gut zu wissen, wie diese einzelnen Teile miteinander interagieren. Empfohlen sei ein Blick auf Oracles Website (*siehe „<http://www.oracle.com/technetwork/developer-tools/apex/apex-arch-086399.html>“*).

Wie man sieht, ist weit mehr als nur deklaratives Entwickeln erforderlich, sobald die Komplexität eines Systems zunimmt.

Glücklicherweise können wir uns in unserem Team gegenseitig mit dem notwendigen Wissen aushelfen.

Das Team

Seit einigen Jahren besteht innerhalb der DB Systel GmbH die Abteilung Small Solutions, die es sich zur Aufga-

be gemacht hat, kleinere Anwendungen zu erstellen. Diese Anwendungen, dies es bisher als MS-Excel- oder MS-Access-Anwendungen unter den Schreibtischen gab, wurden Stück für Stück nach Apex migriert. Allerdings sind heute mehr neue Anwendungen als Alt-Ablösungen angefragt. Egal, um welche Art von Systemen es sich handelt, es war und ist von großer Bedeutung, auf verschiedene Rahmenbedingungen zu achten:

- *Corporate Design*

Es liegt auf der Hand, dass eine Anforderung innerhalb eines Konzerns wie der Bahn die Wiedererkennbarkeit von Anwendungen ist, damit die Anwender sich nicht immer an neue Bedienungselemente gewöhnen müssen.

- *Schaffung wiederverwendbarer Komponenten*

Ohne Schnittstellen kommt heute kaum noch ein System aus, daher war es bei der Entwicklung der Apex-Anwendungsvorlage, die häufig auch als „Bauplan“ bezeichnet wird, wichtig, immer wiederkehrende Schritte wie Daten-Importe via XML- oder „csv“-Dateien zu vereinfachen beziehungsweise konfigurierbar, und damit zu einem Quasi-Standard zu machen.

- *Schnelle Bereitstellung von Umgebungen*

Damit das Deployment von Anwendungen nicht jedes Mal zu einem Abenteuer wird, wurde innerhalb der DB Systel ein Standard-Verfahren etabliert, das dedizierte Oracle-Instanzen mit Apex auf Knopfdruck installiert. Diese Datenbanken werden dann automatisch Bestandteil einer betriebsgeführten Umgebung, die sich um Themen wie „Ausfallsicherheit“, „Datensicherungen“ etc. kümmert.

- *Integration in bestehende Umgebungen*

Eine häufige Anforderung ist das Single Sign-on (SSO). Viele der bisher erstellten Anwendungen hatten die Anforderung, sich mittels Betriebssystem-User anzumelden, also ohne zusätzliches weiteres Passwort. Auch diese Technik ist im Referenz-Bauplan enthalten.

Diese Rahmenbedingungen wurden in mehr als vier Personenjahren in einer Referenz-Anwendung zusammengefasst, die sich im Laufe der Zeit immer weiterentwickelt hat und noch weiterentwickelt. Jene Blaupause hat mir als Apex-Einsteiger

Namen

	Id	Name	Vorname
	1	Schulz	Rüdiger
	2	Müller	Fritz
	4	Meier	Christian
	5	Lehmann	Christian

Abbildung 9: Ärgerliche Namensänderung

1 error has occurred

- This Menu Identification Page already has an associated Menu Option. You must select another page.

Abbildung 10: Breadcrumb-Fehler, kleine Geplänkel erhalten die Feindschaft ...

Table Name: **KUNDEN**

* Primary Key Type: Managed by Database (ROWID) Select Primary Key Column(s)

* Primary Key Column 1:

Primary Key Column 2:

Abbildung 11: Auswahl des Primary Key – weiß jeder, was eine ROWID ist?

Primary Key Column 1: **ID**

* Define the source for the primary key columns:

Existing trigger
 Custom PL/SQL function
 Existing sequence

Abbildung 12: Wie wird der Primary Key bestimmt?

das Entwicklerleben sehr viel leichter gemacht, weil ich das Rad nicht immer neu erfinden musste.

Man könnte nun meinen, dass die geschaffenen Standards die Entwickler-Kreativität stark einengen. Das ist definitiv nicht der Fall. Im Gegenteil, denn es wurde von vornherein darauf geachtet, dass jeder entscheidende Teile zum Gesamtsystem beiträgt, sodass das Ergebnis von allen Teammitgliedern getragen und verwendet wird. Die Motivation im Team ist außerordentlich hoch und die Fluktuation in gleichem Maße gering.

Die Mitarbeiter

Meist werden unsere Projekte von der initialen Beratung bis zur Fertigstellung von einer Person betreut. Daher sind die Team-Mitglieder von Small Solutions eher Generalisten als Spezialisten. Das technologische Wissen der Kollegen ist über das gesamte Team verteilt. Der Austausch erfolgt permanent.

Ein neuer Mitarbeiter wird nicht zwingend Apex-Erfahrungen besitzen müssen, diese kann er durch „Learning by doing“ erlangen. Die Anforderungen an jeden Einzelnen sind hoch: Herausforderungen aus den Bereichen „Beratung“, „Projektmanagement“ und „Anwendungsentwicklung“ müssen beherrscht werden. Am wichtigsten ist jedoch die Kommunikation mit dem Kunden. Wenn diese nicht funktioniert, wird das Projekt erfahrungsgemäß nicht erfolgreich abgeschlossen. Ein „Small Solutions Leitfaden“ unterstützt jeden Mitarbeiter, die Projekte durchzuführen.

Das Vorgehensmodell

Abbildung 13 zeigt das etablierte Vorgehensmodell. Es handelt sich hierbei um einen iterativen Prozess, in dem der Kunde mit seinen Anforderungen im Mittelpunkt steht. In der initialen Beratung, die den Anfang eines jeden Projekts bildet, werden diese mit dem Kunden gemeinsam ermittelt. Es folgt die Erstellung einer ersten Kostenschätzung, die nach dem „Vier-Augen-Prinzip“ innerhalb des Teams „Small Solutions“ verifiziert wird.

Wenn der Kunde diese Schätzung akzeptiert, erfolgt die Beauftragung auf Basis eines Dienstvertrags. Die Entwicklung kann dann ohne Verzögerung beginnen, da für den Entwickler Apex-Workspaces auf einer speziellen Umgebung zur Ver-

```
$(„.BEAUFTRAGT“).each(function() {
    $(,input[class= BEAUFTRAGUNG]`).attr(,checked`, false);
    $(„.BEAUFTRAGT“).val(0);
});
$(this.triggeringElement).attr(,checked`, true);
$(this.triggeringElement).val(1);
```

Listing 2

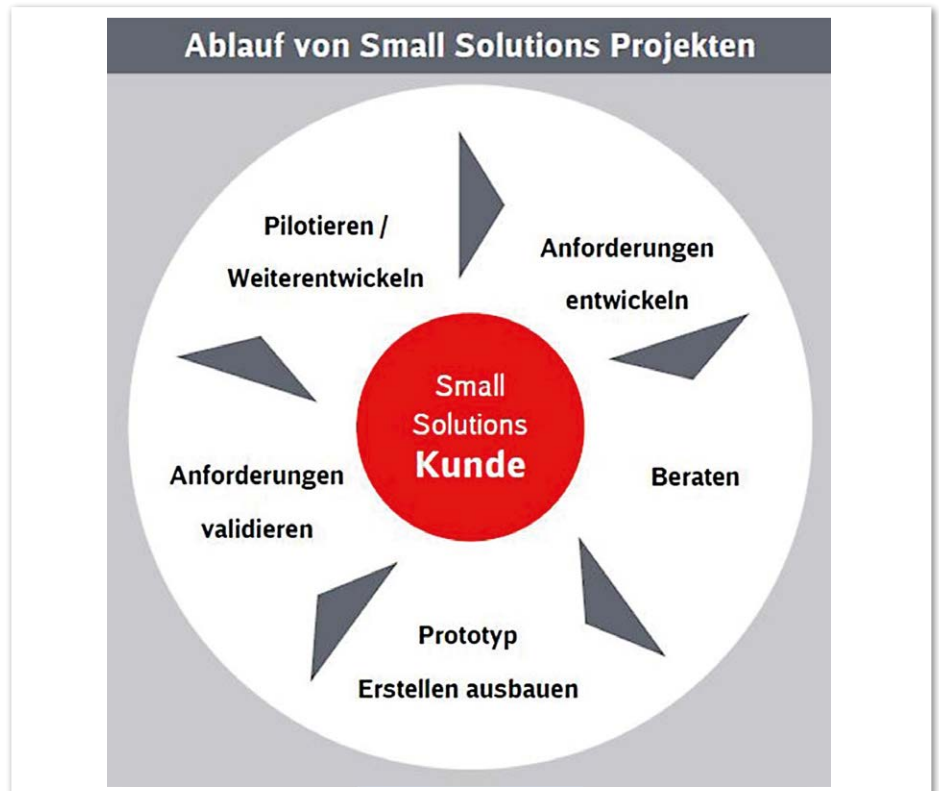


Abbildung 13: Das iterative Vorgehen innerhalb von Small-Solutions-Projekten

fügung stehen. Jeder entwickelte Anwendungsteil wird dem Kunden zum Test zur Verfügung gestellt und eventuelle Anpassungen sind sofort vorgenommen. Die gesamte Applikation wird also in mehreren Schritten prototypisch aufgebaut. Wenn die Entwicklung abgeschlossen ist, erfolgt die Installation der Anwendung auf einer Produktivumgebung und das Projekt ist abgeschlossen.

Die Software-Entwicklung selbst wird durch das Framework getrieben („Framework driven development“). Dies bedeutet, dass die Anforderungen möglichst auf Features der eingesetzten Technologie umgesetzt werden. Dadurch bleiben die Kosten der Entwicklung sehr gering.

Die Kombination aus dem Vorgehen von Small Solutions und der Technologie

von Oracle-Apex hat sich in bislang über 350 Projekten bewährt und zu einer hohen Kundenzufriedenheit geführt.



Rüdiger J. Schulz

ruediger.schulz@deutschebahn.com

D3: Data Driven Documents

Ottmar Gobrecht, Linde AG

D3 ist eine JavaScript-Bibliothek zum Manipulieren von HTML-Dokumenten auf Basis von Daten und setzt dabei auf die Web-Standards HTML, SVG und CSS. Dieser Artikel zeigt die Grundlagen von D3 und weckt das Interesse am Thema „Datenvisualisierung mit Web-Standards“. Vorab nur so viel: Auch D3 kennt den Begriff des Joins.

D3 kennt alle üblichen Charts und beim Anblick der Startseite unter „d3js.org“ wird einem klar, dass es wohl keine Grenzen bei der Umsetzung zu geben scheint. Deshalb hier der unvollständige Versuch, D3 etwas genauer zu definieren:

- D3 ist eine JavaScript-Entwicklerbibliothek, die es erlaubt, Daten mit grafischen Elementen (SVG) im Browser zu kombinieren und diese so auf vielfältige Art und Weise zu visualisieren
- Durch seinen Data-Join-Ansatz ist D3 im Verhältnis zu anderen Bibliotheken auch bei großen Datenmengen sehr schnell
- Aufgrund der Unterstützung von Webstandards ist D3 zukunftssicher und auch auf mobilen Geräten lauffähig
- D3 basiert auf HTML5 und CSS3, es benötigt also einen modernen Browser (Firefox, Chrome, Safari, IE9 aufwärts)
- D3 ist keine Chart-Engine, bei der man fertige Layouts auswählt und konfiguriert

Das Fundament: SVG

SVG steht für „Scalable Vector Graphics“ und basiert auf XML. Die zweidimensionalen SVG-Zeichnungselemente haben gegenüber Raster-Grafiken den Vorteil, verlustfrei skaliert werden zu können. Ihnen fehlt also der berühmte Treppenstufen-Effekt, der bei extremer Vergrößerung in Raster-Formaten auftritt. SVG im Browser wurde im Rahmen von HTML5 standardisiert.

Im Gegensatz zu HTML-Elementen, die im Grunde nur die Rechteckform kennen, gibt es bei SVG-Elementen Pfade und zur einfacheren Handhabung alle grafischen Grundformen wie Kreis, Ellipse, Rechteck,

Linie und Polygon. Zusätzlich kann man noch Text und externe Rastergrafiken verwenden. Alle genannten Elemente können durch das Gruppenelement zusammengefasst werden.

Elemente und Gruppen von Elementen können über Style-Attribute in ihrer Erscheinung angepasst und über Transformationen in Position, Orientierung und Form verän-

dert werden. An Transformationen stehen Parallelverschiebung, Rotation, Skalierung und Scherung zur Verfügung, die über CSS3-Transitions auch in einem vorgegebenen Zeitraum verändert werden können. Daneben gibt es noch Animationen sowie grafische Effekte und Filter. Wer sich tiefergehend dafür interessiert, findet im Web eine Menge Informationen und Beispiele.

```
<!DOCTYPE html><html><head><title>SVG Beispiel</title></head><body>
<script src="http://d3js.org/d3.v3.min.js"></script>

<svg style="width:490; height:140;">
  <rect x="10" y="10" height="120" width="160" fill="#ff6600"/>
  <ellipse cx="270" cy="70" rx="120" ry="40" fill="green"/>
  <line x1="40" y1="40" x2="480" y2="120" stroke="blue"/>
</svg>

<script>
  d3.select('svg').append('circle')
    .attr('cx', '420')
    .attr('cy', '70')
    .attr('r', '60')
    .style('stroke', 'red')
    .style('fill', 'lightsteelblue')
    .style('fill-opacity', 0.5);
</script>

</script></body></html>
```

Listing 1: Ein einfaches SVG-Beispiel

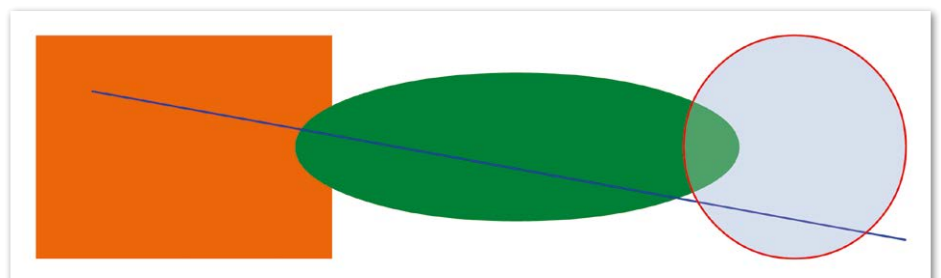


Abbildung 1: Das Ergebnis aus Listing 1

```
// Schleife über alle Elemente: JavaScript
var p = document.getElementsByTagName('p');
for (var i = 0; i < p.length; i++) {
  var pi = p.item(i);
  pi.style.setProperty('color', 'red', null);
}

// Schleife über alle Elemente: D3
d3.selectAll('p').style('color', 'red');
```

Listing 2: JavaScript versus D3

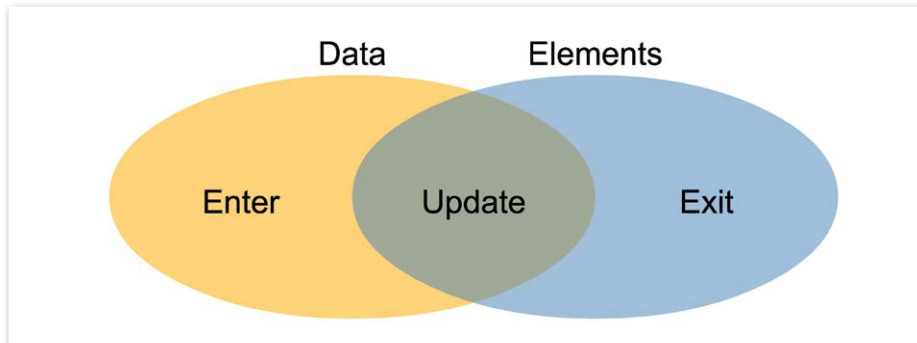


Abbildung 2: Der D3-Data-Join

Man kann SVG-Grafiken auf mehreren Wegen in HTML einbinden: als externe Grafik oder direkt im HTML mit dem Tag „<svg>“. Bei einer externen Grafik ist wie bei XML der Namespace zu definieren. Bei direkter Einbindung in HTML ist das nicht erforderlich – was im Zusammenhang mit D3 den Standardfall darstellt. Bis auf die ungewohnten Tag- und Attribut-Namen wird einem also nicht viel auffallen, wenn man sich eine HTML-Seite mit eingebettetem SVG im DOM-Inspector eines modernen Browsers anschaut. Insofern ist SVG recht pflegeleicht und man kann die gewohnten Werkzeuge des Browsers nutzen.

Wer jQuery kennt, wird sich schnell in D3 zurechtfinden. D3 unterstützt auch die Methodenverkettung; die Basismethoden wie „append“ und „attr“ haben die gleichen Namen oder zumindest ähnliche wie „style“ bei D3 und „css“ bei jQuery. Listing 1 zeigt ein Beispiel, drei SVG-Elemente direkt in HTML und ein Element prozedural mit D3 (siehe Abbildung 1).

Wie Listing 1 zeigt, heißt bei SVG das Attribut für die Füllfarbe „fill“ und für die Konturfarbe „stroke“. Der Bezugspunkt für das Positionieren von Elementen ist die linke obere Ecke des SVG-Elements. Das gilt auch beim Rechteck-Element; beim Kreis und bei der Ellipse ist der Bezugspunkt hingegen die Mitte.

zunehmen, muss man mit einer Schleife über alle Elemente iterieren. D3 verfolgt einen anderen Ansatz, wie man schon an der Bezeichnung erkennen kann – hier wird von „Selections“ gesprochen. Jede Selection ist ein Array, auch wenn es nur ein oder gar kein Element enthält. Wenn man dieses dann mit einer Methode aufruft, iteriert D3 – wie übrigens jQuery auch – automatisch über alle Elemente. Listing 2 zeigt den Vergleich.

Daten an das DOM binden

Um Daten an das DOM binden zu können, müssen diese erst einmal zur Verfügung stehen. D3 stellt für das Lesen von Daten XHR-Methoden („XMLHttpRequest“) zur Verfügung. Damit können Daten per JavaScript nachgeladen werden. Der Vorteil dieses Vorgehens ist, dass sich das Laden der eigentlichen Webseite bei großen Datenmengen nicht verzögert.

Zur einfacheren Handhabung gibt es darauf aufbauend vier Wrapper-Methoden, die für das Laden von CSV-, TSV-, XML- und JSON-Daten genutzt werden können. Häufig kommt es aber auch vor, dass man diese auch „AJAX-Calls“ genannten Aufrufe mit den Methoden des jeweils benutzten Entwicklungs-Frameworks er-

Selectors versus Selections

In JavaScript kann man mit den aus CSS bekannten Selektoren Elemente im Document Object Model (DOM) auswählen. Um dann Änderungen an den Elementen vor-

```
<!DOCTYPE html><html><head><title>Data Join</title></head><body>
<script src="http://d3js.org/d3.v3.min.js"></script>

<p style="color:green;">Ein bereits existierendes Element</p>

<script>

  var body = d3.select('body');

  var p = body.selectAll('p').data([1,2,3])
    .style('color', 'red');
  // http://bost.ocks.org/mike/join/
  // http://bost.ocks.org/mike/circles/

  p.enter().append('p')
    .text( function(d){return ,Neues Element aus Daten , + d; } );

  p.exit().remove();

</script></body></html>
```

Listing 3: Data Join

Ein bereits existierendes Element

Neues Element aus Daten 2

Neues Element aus Daten 3

Abbildung 3: Das Ergebnis aus Listing 3

ledigt, wobei sich das Datenformat JSON durchgesetzt hat.

Bei Apex gibt es zum Beispiel fertige Methoden in den mitgelieferten JavaScript-Bibliotheken und bei kleineren Datenmengen mag es auch ok sein, diese direkt in die Seite zu rendern. Wege gibt es viele. Am Ende braucht man JavaScript-Arrays, die die Daten enthalten und über die D3 iterieren kann. Wie die Zeilen beziehungsweise Objekte des Arrays aussehen, hängt stark vom Anwendungsfall ab.

Nun zur eigentlichen Frage: D3 bindet Daten an das DOM, indem man einer Selection über den Data Operator die Daten in Form eines Arrays übergibt. Hier kommt der eingangs erwähnte Data Join zum Einsatz. Zurückgeliefert werden drei neue Selections, die D3 „Update“, „Enter“ und „Exit“ nennt. *Abbildung 2* zeigt schematisch die Zusammenhänge.

In der Update Selection sind die Elemente des DOM mit den Elementen des Data Arrays in der Reihenfolge ihres jeweiligen Erscheinens verknüpft. Wie man unschwer errät, kann jedoch eine unterschiedliche Anzahl von Elementen im DOM und in den Daten vorhanden sein. Wenn im DOM mehr Elemente vorhanden sind als Daten, dann liegen diese Elemente in der Exit Selection, mit der man die Elemente aus dem DOM entfernen kann.

Gibt es mehr Daten als DOM-Elemente, dann befinden sich diese Daten in der Enter Selection, mit der die fehlenden Elemente im DOM erstellt werden können. *Listing 3* zeigt ein Beispiel und *Abbildung 3* das Ergebnis.

Wie man in *Listing 3* sieht, steht die Update Selection sofort in der Methodenverkettung zur Verfügung und sie hat keinen eigenen Namen. Es wäre aber auch möglich, die Enter oder Exit Selection zuerst aufzurufen. Wichtig ist zu wissen, dass nach Abarbeitung der ersten Selection die anderen Selections gesondert aufgerufen werden müssen – eine weitere Verkettung ist nicht möglich. Außerdem stehen einem die Elemente der Enter Selection nach Ausführung in der Update Selection zur Verfügung. Man könnte also beispielsweise zuerst die neuen Elemente anlegen, ohne weitere Attribute oder Styles zu definieren, und dann mit einem Aufruf der Update-Selection alle Elemente auf einmal aktualisieren.

Für den Fall, dass der einfache Data Join über die Reihenfolge der Elemente in DOM und Array nicht genügt, gibt es eine sogenannte „Key Function“, anhand derer die Elemente im DOM identifiziert werden. Die anonyme Funktion wird bei Bedarf als zweiter Parameter dem Data Operator übergeben. *Listing 4* zeigt ein

```
<!DOCTYPE html><html><head><title>Data Key Function</title></head><body>
<script src="http://d3js.org/d3.v3.min.js"></script><script>

var data = [
  {"id":1, "x":40, "y":40},
  {"id":2, "x":120, "y":40},
  {"id":3, "x":200, "y":40}];

var svg = d3.select('body').append('svg');

var circle = svg.selectAll('circle')
  .data(data, function(d) {return d.id;});
  // Wie all das funktioniert: http://bost.ocks.org/mike/selection/

circle.exit().remove();

circle.enter().append('circle')
  .attr('id', function(d) { return d.id; })
  .attr('r', 10);

  circle
  .attr('cx', function(d) { return d.x; })
  .attr('cy', function(d) { return d.y; });

</script></body></html>
```

Listing 4: Data Key Function

Sparen Sie Zeit, Geld und Nerven.



Effizient und preiswert:
DBConcepts.

Wir unterstützen Sie remote beim
Betrieb von Oracle Datenbanken.

SLA ab 10hx5 bis 24hx7 inklusive

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



Die Oracle Experten

www.dbconcepts.at
Tel.: +43 1 890 89 990
office@dbconcepts.at



Beispiel einer Data-Key-Funktion – hier wird zur Identifikation die ID verwendet. Wer wissen möchte, wie das alles ganz genau funktioniert, der findet unter dem Data Join im Listing einen Link, in dem der Autor von D3 das sehr ausführlich erklärt.

Das Geheimnis der Geschwindigkeit

Die Konsequenz aus dem Vorgehen von D3 ist, dass man immer zuerst eine Selection erstellt, auch wenn man weiß, dass im DOM keine Elemente vorhanden sind. An die Selection werden die Daten gebunden und im Falle einer statischen Visualisierung reicht es dann aus, über die Enter Selection die DOM-Elemente zu erstellen.

Im Falle von dynamischen Visualisierungen spielt dieses Vorgehen sein Stärken aus. Ändern sich nicht alle Daten, muss D3 nur die neuen Elemente im DOM anlegen und alte entfernen; vorhandene können aktualisiert werden. Dies spart Ressourcen im Browser, da nur die nötigste Arbeit zu tun ist, anstatt alle Elemente aus dem DOM zu entfernen, um sie dann wieder neu anzulegen.

Außerdem speichert sich D3 in den Selections die Referenzen auf die DOM-Elemente; dadurch muss für weitere Updates des DOM keine weitere Elementsuche auf dem DOM ausgeführt werden. Nun ist es gelüftet, das Geheimnis hinter der hohen Geschwindigkeit – D3 verhindert ganz einfach unnötige Arbeit für den Browser, womit Rechenkapazität für das Bearbeiten größerer Datenmengen frei wird.

Chart-Code wiederverwenden

Der Schwerpunkt von D3 liegt nicht auf dem Ausliefern von fertigen Chart-Funktionen, in die man einfach seine Daten kippt. D3 konzentriert sich auf grundlegende, immer wieder benötigte Dinge und bietet dafür generisch verwendbare Hilfs-Funktionen an. Man muss zwar Arbeit in seine Charts stecken, hat aber durch den generischen Ansatz der Basis-Funktionen praktisch keine Einschränkungen bei der Umsetzung, wie man an den vielen Beispielen im Internet sehen kann.

Man kommt am Anfang mit diesen Beispielen schnell zu ansehnlichen Ergebnissen. Spätestens wenn man mit dem gleichen Code einen zweiten Chart auf derselben HTML-Seite einbauen möchte, sieht man jedoch die Grenzen: Globa-

Closures in JavaScript

In JavaScript ist alles ein Objekt und Funktionen können Objekte, also auch Funktionen, zurückgeben. Eine Closure ist am Ende nichts anderes als ein Funktionsaufruf, bei dem eine Funktion zurückgegeben wird. Die zurückgegebene Funktion behält ihren vorherigen Kontext, kann also auf die vor der Rückgabe vorhandenen Variablen innerhalb ihrer Ursprungsfunktion zugreifen. Sie kapselt quasi ihre Umge-

bung, daher der Name. Die Folge eines solchen Vorgehens ist, dass man dadurch private Variablen erhält, die von außen nicht einsehbar sind. Um trotzdem an diese Variablen zu kommen, muss man dann entsprechende Get- und Set-Funktionen definieren. *Listing 5* zeigt die Prinzipien. *Abbildung 4* zeigt das Ergebnis aus *Listing 5* und kann einfach in der JavaScript-Konsole des Browsers nachvollzogen werden.

```

<!DOCTYPE html><html><head><title>Prinzip Closure</title></head><body>
<script>

  function myChart() {

    var conf = { "width": 600, "height": 400 };

    function chart(){/*create chart with conf*/}

    chart.render = function(){
      chart();
      return chart;
    };

    chart.width = function(value) {
      if (!arguments.length) return conf.width;
      conf.width = value;
      return chart;
    };

    return chart;
  }

</script></body></html>

```

Listing 5: Prinzip Closure

Console

```

> my = myChart().render();
function chart(){/*create chart with conf*/}
> my.width();
600
> my.width(500);
function chart(){/*create chart with conf*/}
> my.width();
500

```

Abbildung 4: Test Closure in Browser-Konsole

le Variablen und Funktionen sind schnell redeclariert und ein zweiter Chart stört die Funktion des ersten. Hier ist es dann

an der Zeit, über Kapselung und Wiederverwendung von Chart-Code zu sprechen. Dazu muss man wissen, wie JavaScript mit

Variablen und Funktionen umgeht. Das Stichwort hier sind Closures (*siehe Kasten*).

Wenn man nun noch wie in *Listing 5* bei diesen Get- und Set-Funktionen wiederum die Funktion selbst zurückliefert, dann hat man die Basis für eine Methoden-Verkettung geschaffen. Wie man sieht, lohnt es sich, ein wenig Arbeit in ein kleines API zu stecken – als Gegenleistung bekommt man universell einsetzbare Charts, die je nach Gegebenheit initialisiert und zur Laufzeit angepasst werden können, um zum Beispiel auf User-Interaktionen zu reagieren.

Die Königsklasse: Physik-Simulationen

D3 bietet für verschiedene Anwendungsfälle sogenannte Layouts an. Das sind im Grunde die angesprochenen wiederverwendbaren Charts, aber auf einem generischen Level. Wir gehen das an dem Beispiel eines Force-Layouts durch, einer beliebigen, kräftebasierten Darstellung von Netzwerken.

Die Besonderheit einer kräftebasierten, selbstorganisierenden Netzwerk-Visualisierung ist die dahinterliegende physikalische Simulation. D3 berechnet fortwährend die Positionen der Netzwerk-Knoten. Damit der Eindruck einer Animation entsteht, muss man natürlich die Knoten seines Graphen irgendwann einmal neu positionieren. Diese Arbeit nimmt einem D3 nicht ab. D3 bietet aber ein sogenanntes „Tick Event“, das ungefähr alle zwanzig Millisekunden feuert. Wenn man dann bei jedem Tick-Event alle Knoten und Verbindungen des Netzwerks an die jeweils aktuelle Position der physikalischen Si-

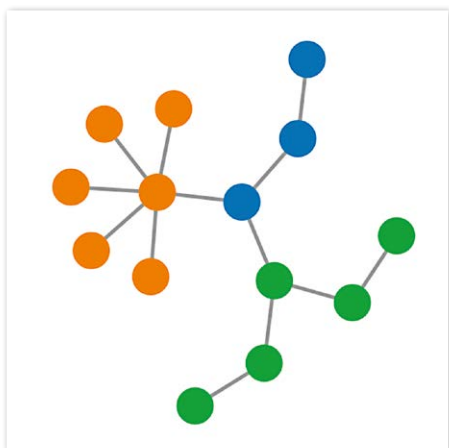


Abbildung 5: Das Ergebnis aus Listing 5

mulation verschiebt, bekommt man den Eindruck einer Animation. Die Animation des Netzwerks hört dann auf, wenn sich die Simulation in einem kräftemäßig ausgewogenen Zustand befindet.

Listing 6 zeigt ein minimales, vollständig lauffähiges Skript für ein Force Layout. Zu-

erst die Daten – sie dürften den meisten bekannt sein. Als Knoten nehmen wir die Mitarbeiter der EMP-Tabelle, als Links die Beziehung zum Vorgesetzten. Die Links beziehen sich auf die Knoten in der Reihenfolge ihres Erscheinens, es wird also keine Key Function eingesetzt, um die

```
<!DOCTYPE html><html><head><title>Force Layout</title></head><body>
<script src="http://d3js.org/d3.v3.min.js"></script><script>

  var nodes = [
    {"name": "King", "dept": 10}, {"name": "Blake", "dept": 30},
    {"name": "Clark", "dept": 10}, {"name": "Jones", "dept": 20},
    {"name": "Scott", "dept": 20}, {"name": "Ford", "dept": 20},
    {"name": "Smith", "dept": 20}, {"name": "Allen", "dept": 30},
    {"name": "Ward", "dept": 30}, {"name": "Martin", "dept": 30},
    {"name": "Turner", "dept": 30}, {"name": "Adams", "dept": 20},
    {"name": "James", "dept": 30}, {"name": "Miller", "dept": 10}
  ];

  var links = [
    {"source": 1, "target": 0}, {"source": 2, "target": 0},
    {"source": 3, "target": 0}, {"source": 7, "target": 1},
    {"source": 8, "target": 1}, {"source": 9, "target": 1},
    {"source": 10, "target": 1}, {"source": 12, "target": 1},
    {"source": 13, "target": 2}, {"source": 4, "target": 3},
    {"source": 5, "target": 3}, {"source": 6, "target": 5},
    {"source": 11, "target": 4}
  ];

  var width = 600, height = 400;

  var svg = d3.select("body").append("svg")
    .attr("width", width).attr("height", height);

  var color = d3.scale.category10();

  var force = d3.layout.force().size([width, height]);

  var link = svg.selectAll("line").data(links)
    .enter().append("line").style('stroke', '#999');

  var node = svg.selectAll("circle").data(nodes)
    .enter().append("circle")
    .attr("r", 5)
    .style("fill", function(d) { return color(d.dept); })
    .call(force.drag);

  force.on("tick", function() {
    link.attr("x1", function(d) { return d.source.x; })
      .attr("y1", function(d) { return d.source.y; })
      .attr("x2", function(d) { return d.target.x; })
      .attr("y2", function(d) { return d.target.y; });

    node.attr("cx", function(d) { return d.x; })
      .attr("cy", function(d) { return d.y; });
  });

  force.nodes(nodes).links(links).start();

</script></body></html>
```

Listing 6: Force Layout

Knoten zu identifizieren. Dies ist in Ordnung, solange keine Updates auf den Graphen erfolgen sollen, ansonsten wird es einem schwer fallen, die Knoten im DOM zu identifizieren.

Will man zum Beispiel die Knoten in Farbe und Größe passend zu den Daten gestalten, kann man auf die vielen Hilfsfunktionen von D3 zurückgreifen. Als Beispiel sei hier die Farbe der Knoten genannt. In *Listing 6* kann man erkennen, wie die Variable „color“ mit der Funktion „d3.scale.category10()“ belegt ist. Diese Funktion liefert eine aus zehn verschiedenen Farben für den übergebenen Wert zurück. Wir benutzen dies dafür, um über die Abteilungsnummer eines Mitarbeiters eine Farbe zurückzubekommen – somit haben alle Mitarbeiter einer Abteilung die gleiche Knotenfarbe, ohne dass wir dafür etwas programmieren müssen.

Nach der Variable „color“ wird eine Variable „force“ mit dem entsprechenden Layout belegt. Dann folgt schon die Variable „link“. Hier werden alle Linien im SVG-Element selektiert – in unserem Fall ergibt das ein leeres Array, weil wir ja wissen, dass wir zum ersten Mal den Graphen erstellen.

Dann folgt in derselben Zeile der Data Join mit unseren Links und gleich darauf wird in der Enter Selection für jeden Eintrag in unserem Array „link“ eine Linie im SVG-Element gezeichnet. Das Ganze wiederholen wir jetzt für die Knoten. Daraufhin definieren wir noch für das Tick Event die bereits erwähnte Funktion, die dann jeweils unsere Knoten und Links innerhalb des SVG-Elements positionieren. Man kann

hier schön erkennen, wie D3 jeweils implizit über das Array „link“ und „node“ iteriert und für jeden Eintrag in der darauf definierten anonymen Funktion als ersten Parameter die Daten des Eintrags übergibt, die einem dann für die weitere Verarbeitung zur Verfügung stehen. In unserem Fall geben wir einfach nur die jeweilige Position für den Link oder den Knoten zurück.

Als letzter Schritt wird noch die eigentliche physikalische Berechnung gestartet, die natürlich unsere beiden Knoten und Link Arrays benötigt, um vernünftig arbeiten zu können. Als Ergebnis erhalten wir den Graphen aus *Abbildung 5*, der schon ein wenig an den ersten Graphen unter dem Artikelbild erinnert.

Wie man sieht, muss man oft nur minimal Code erstellen, um Charts an seine eigenen Bedürfnisse anzupassen. Und sollte man doch mal etwas Komplizierteres benötigen, so kann man davon ausgehen, im Internet viele Beispiele für die gleiche oder eine ähnliche Problemstellung zu finden.

Netzwerke: ein Plug-in (nicht nur) für Apex

Wer sich dafür interessiert, was man noch so alles machen muss, um vom eben gezeigten Minimalbeispiel zu der großen Lösung aus dem Artikelbild zu kommen, findet eine vollständige Implementierung eines Netzwerk-Charts als Apex-Plug-in auf GitHub. Allen Nicht-Apex-Entwicklern sei gesagt, dass trotz seines Namens nicht zwingend Apex erforderlich ist.

Wie im Artikel empfohlen, existiert ein vollständiges JavaScript-API. Darüber hin-

aus gibt es einen interaktiven Customize Wizard und die Chart-Funktion stellt bei nicht vorhandenen Daten selbstständig Beispieldaten zur Verfügung. Mit diesen Voraussetzungen kann man nach dem Einbinden der benötigten Source-Dateien sofort loslegen, Parameter verändern und live die Auswirkungen testen.

Fazit

Mit D3 stehen einem eine Menge Wege offen, eigene Charts zu erstellen. Man muss zwar zuerst ein wenig Arbeit investieren, bekommt aber bei überlegter Umsetzung wiederverwendbaren Code und hat praktisch keine Einschränkungen in der Umsetzung.

Weitere Informationen

1. <http://d3js.org>
2. <https://github.com/ogobrecht/d3-force-apex-plugin>
3. https://apex.oracle.com/pls/apex/GERMAN_COMMUNITIES.SHOW_TIPP?P_ID=3481
4. <http://www.apex-plugin.com/search-plugin-directory> (nach D3 suchen)



Ottmar Gobrecht
ottmar.gobrecht@linde.com

ADF-Buch der deutschen ADF-Community erschienen

Der Wunsch nach Wissensaustausch stand an erster Stelle – herausgekommen ist ein umfangreiches Referenzwerk zum Thema „Oracle ADF“. Rund 1.400 Seiten ist das Kompendium stark, das die deutsche ADF-Community zusammengestellt hat. Es steht kostenfrei als PDF-Dokument unter „<http://stream.doag.org/adf/ADF-Buch.pdf>“ zum Download bereit.

Für das ADF-Buch haben die Autoren zahlreiche Artikel, Vorträge und Workshop-Unterlagen einer Vielzahl von Referenten

zusammengestellt. Die Leser bekommen somit das gesammelte Entwicklungs-Know-how der letzten fünf Jahre und einen breiten, praxisorientierten Querschnitt durch alle ADF-Themengebiete geboten.

Neben den Grundlagen umfasst das Nachschlagewerk zahlreiche Lernbeispiele sowie Erfahrungsberichte zu konkreten Beispielen. Ein Überblick über Architektur, Oracle Mobile Application Framework (MAF) und zahlreiche Spezialthemen wie beispielweise „ADF mit Eclipse“ und „Mig-

ration Forms zu ADF“ vervollständigen das ADF-Buch.

Robert Szilinski, Leiter der DOAG Development Community, resümiert begeistert die Zusammenarbeit aller Beteiligten: „Alle haben sich mit sehr großem Engagement in dieses Buch eingebracht und waren bereit, ihr Wissen und ihre Erkenntnisse im Sinne einer Community untereinander zu teilen. Das ist wirklich fantastisch und es zeigt den einzigartigen Spirit, der sich inzwischen in der Community entwickelt hat.“

Zehn nützliche Apex-Tipps aus der Praxis

Andreas Wismann, WHEN OTHERS

Als Apex-Entwickler staunt man bisweilen nicht schlecht, welche neuen, spannenden Features und Arbeitsweisen man immer wieder entdeckt.



Mit User Interface Defaults viel Zeit sparen

Nicht immer bleibt es dabei, dass man pro verwendete Tabelle nur genau eine Pflegemaske baut. Die gleichen Tabellen werden oftmals in unterschiedlichen Formularen oder Reports benutzt, lediglich die Ansicht oder die verwendeten Spalten unterscheiden sich. Zu diesem Zweck gibt es ein extrem hilfreiches Apex-Feature, das im SQL-Workshop ein ungerechtfertigtes Schattendasein fristet, nämlich die User Interface Defaults. Die Idee ist, vornehmlich Spalten-Label, also die Feldnamen, die dem Benutzer angezeigt werden, und aussagekräftige Hilfetexte festzulegen, bevor man die Tabellenspalten erstmals in Apex-Seiten verwendet. Der große Vorteil besteht darin, dass bei Wiederverwendung derselben Tabelle erneut dieselben Spalten-Label und Hilfetexte zum Einsatz kommen und man dadurch eine konsistente Benutzerführung erreicht. Wer User Interface Defaults verwendet, wird eine Tabellenspalte namens „KDNR“ in allen anschließend erzeugten Formularen einheitlich beispielsweise als „Kunden-Nr.“ sehen und nicht wechselweise „Kundennr.“, „Kundennummer“, „Kd-Nr“, „KNr“ etc.

Unter „SQL Workshop -> Utilities -> User Interface Defaults“ lässt sich rechts

oben das gewünschte Schema einstellen, falls es mehrere davon gibt (siehe *Abbildung 1*). Unter „Manage Table Dictionary“ findet man bereits erzeugte Vorgaben,

diese sind in vielen Apex-Projekten erfahrungsgemäß leer. Durch die Synchronisierung werden das Repository und eventuell neu hinzugekommene Tabellen

Abbildung 1: Die User Interface Defaults

miteinander abgeglichen. Dabei liest Apex unter anderem die bestehenden Spaltenkommentare der Tabellen aus und verwendet sie als Hilfetext-Vorgabe – das ist ziemlich hilfreich, selbst wenn man sicher einige Kommentare anschließend noch umformulieren möchte. Man kann auf jeden Spaltennamen klicken und dann jeweils für Formulare und Reports die passenden Beschriftungen definieren.

Die Detailfülle der Einstellungen geht sogar noch weiter: Man legt fest, mit welchem Item-Typ das Feld in einem Formular grundsätzlich gerendert wird (beispielsweise „Date Picker“ oder „List of Values“) oder wie es sich verhalten soll (etwa „Submit when Enter pressed“); sogar die relative Anzeigereihenfolge und die horizontale Ausrichtung in Reports lassen sich bestimmen. Wer möchte, kann an dieser Stelle auch eine statische oder dynamische LOV definieren – das erspart den Sprung in die Shared Components.

Was auf der Seite „Table Dictionary“ pro Tabelle einstellbar ist, lässt sich in der Kategorie „Attribute Dictionary“ tabellenunabhängig einrichten. Sofern man in einem Formular oder Report ein Item mit einem hier festgelegten Namen anlegt, gelten die User Interface Defaults als Vorgabe.

Die Vorgaben in den User Interface Defaults werden zum Erstellungszeitpunkt eines Formulars oder Reports berücksichtigt; bei Änderungen der Defaults erfolgt keine nachträgliche Synchronisierung der bereits erstellten Formulare. Um den größtmöglichen Nutzen aus diesem Feature zu ziehen, sollte man also seine User Interface Defaults möglichst früh ausarbeiten – idealerweise schon bevor das erste Formular überhaupt vom Band rollt.



Hilfefenster ohne Hilfetext finden

Es ist für die Akzeptanz einer Apex-Anwendung nicht förderlich, wenn der Benutzer auf das Label eines Felds klickt und sich daraufhin ein animiertes Hilfefenster einblendet, in dem lediglich steht: „No help is available for this item“. In einigen Templates ist aber genau dieser Überraschungseffekt vorprogrammiert, weil das standardmäßig verwendete Label-Template, das Apex allen neuen Items zuweist, eben auf „... with help“ endet – und dieses Template bietet immer einen Hilfefenster-Link an, egal ob es einen Hilfetext

gibt oder nicht. Man kann nun alle Items seiner Anwendung durchklicken, um diejenigen zu finden, die keinen Hilfetext besitzen, mit dem Ziel, entweder dort etwas Hilfreiches zu formulieren oder das Label-Template auf „... without help“ zu ändern. Diese Items findet man allerdings wesentlich einfacher, indem man eine der zahlreichen Apex-internen Views abfragt (siehe Listing 1). Vorschlag: Man erstellt aus diesem SQL-Statement einen Classic Report nur für Entwickler, der direkt zur entsprechenden Anwendungsseite verlinkt.

```
SELECT *
FROM apex_application_page_items
WHERE item_label_template LIKE
,'%help'
AND item_help_text IS NULL
```

Listing 1



Hilfetexte mit HTML anreichern

Der schmucklose Hilfetext eines Page-Items lässt sich durchaus mit HTML-Code aufwerten. Gegen das Einbinden von Grafiken jeglicher Art spricht ebenfalls nichts. Alles, was man dazu benötigt, ist HTML-Handwerkszeug. Das Hilfefenster, das sich beim Anklicken eines Labels öffnet, beherrscht (da es Teil des DOM-Baums

der Apex-Seite ist) alle Varianten der Gestaltung und Formatierung, die HTML und CSS landläufig mitbringen. So kann man längere Erläuterungen mit Absätzen und Tabellen anreichern (siehe Abbildung 2). Wie zu sehen ist, sind durchaus Style-Angaben in Form von Inline-CSS erlaubt (siehe Abbildung 3).



Bestimmte Regionen nur für Entwickler anzeigen

Zur Qualitätssicherung der Anwendungen eignen sich Abfragen und Reports wie im zweiten Tipp, doch diese Informationen sollten natürlich nur Entwicklern zugänglich sein. Die Regionen werden also entweder vor der Auslieferung gelöscht (Nachteil: Dann muss man sie gegebenenfalls später wieder neu erstellen.) oder grundsätzlich dem Zugriff des Anwenders entzogen. Zunächst könnte man auf die Idee kommen, dies über eine Condition zu formulieren (etwa „User is developer“), aber diesen Bedingungstyp bietet Apex in der Liste der vordefinierten Bedingungen nicht an. Macht aber nichts, denn man kann ja immer noch eine geeignete Condition per SQL schreiben, wenn man sie denn nur wüsste ...

Eine Möglichkeit, den Entwicklerstatus abzufragen, ist der SQL-Ausdruck „apex_application.g_edit_cookie_session_id IS NOT NULL“. Würde man den Bedingungstyp eines Entwickler-Reports auf „SQL Expression“ stellen und diese Bedingung ins Feld

Help Text

Help Text

<p>Ändern Sie bei Bedarf den Rückgabewert der Funktion, wobei die folgenden Abkürzungen verwendet werden können:</p>

A	ANYTYPE
B	BOOLEAN
BL	BLOB
CL	CL
D	DATE
I	INTEGER
N	NUMBER
P	PLS_INTEGER
T	TIMESTAMP
V	VARCHAR2
X	XMLTYPE

Abbildung 2: Längere Erläuterungen mit Absätzen und Tabellen anreichern

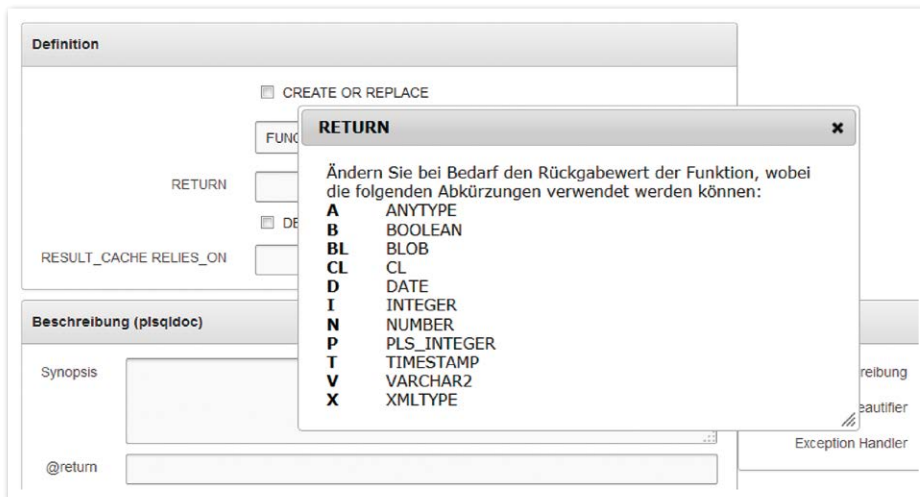


Abbildung 3: Style-Angaben in Form von Inline-CSS

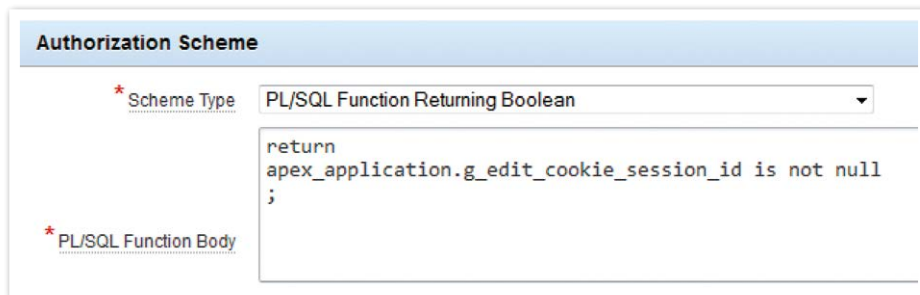


Abbildung 4: Authorization Scheme

„Expression 1“ eingeben, käme man zwar zum gewünschten Ergebnis: Nur noch die Entwickler sehen den Report.

Die Sache hat allerdings einen kleinen Haken: Sollte sich die Implementierung dieses Features jemals ändern, und das könnte sie tatsächlich mit jeder neuen Apex-Version, so müsste man alle Stellen in den Apex-Programmen suchen und ersetzen, die diese Formulierung verwenden, einschließlich der leicht abgewandelten Schreibweisen mit eingefügtem Zeilenumbruch etc.

Wenn man es sich überlegt, stellt die Rolle eines Benutzers keine Bedingung im eigentlichen Sinne dar; es handelt sich eher um eine Frage des Zugriffsschutzes. Zugriffsberechtigungen würde man in Apex unter „Authorization Schemes“ definieren – also erledigt man das am besten auch dort. Dazu erstellt man unter „Shared Components -> Authorization Schemes“ ein neues Schema namens „User is Developer“. Typ ist „PL/SQL Function Returning Boolean“; man gibt die Bedingung als Return-Befehl formuliert ein (siehe Abbildung 4).

Man vergibt eine Error Message, die im Grunde nur eine Rolle spielt, wenn der Benutzer eine Apex-Seite mit Autorisierungsschema aufruft, ohne dafür berechtigt zu sein. Man stellt sicher, dass unter „Evaluation Point“ der Wert „Once per Session“ ausgewählt ist, und speichert das Schema. Anschließend wird wieder der Entwickler-Report geöffnet und im Feld „Authoriza-

tion Scheme“ das gerade erstellte Autorisierungsschema zugeordnet. Nach dem Abspeichern wird der Report nur noch für Entwickler gerendert.

Tipp: Wer auf Nummer Sicher gehen möchte, weist dem Entwickler-Report noch eine Build-Option namens „Development“ zu und stellt diese beim Export der Anwendung auf „Exclude“ um. Durch diesen doppelten Boden würde der Report auch dann nicht dargestellt, falls sich die Entwicklerstatus-Implementierung irgendwann einmal ändern sollte, bevor man den Code anpassen kann. Im schlimmsten Fall (also wenn „User is Developer“ positiv ermittelt würde, obwohl der Benutzer gar kein Entwickler ist) könnten dann nur die Anwender in der Entwicklungsumgebung diesen Report sehen, denn in den exportierten Anwendungen greift ja außerdem noch die Notbremse mit der Build-Option.



Unmittelbar nach dem Login

Es gibt typische Aufgaben, die man direkt nach jeder (erfolgreichen) Benutzeranmeldung durchführen möchte, so zum Beispiel das Einlesen der Benutzerrechte. In Apex können solche Initialisierungsroutinen an diversen Stellen hinterlegt sein. Dennoch gibt es einen Platz, der sich dafür in besonderer Weise eignet und deshalb bevorzugt werden sollte, das Login-Processing. Um dorthin zu gelangen, klickt man auf „Shared Components -> Authentication Schemes“ und editiert das aktive Schema. Weiter unten findet sich das Eingabefeld „Post-Authentication Procedure Name“. Die Prozedur, die man dort benennt, wird nach dem Login aufgerufen – es kann auch

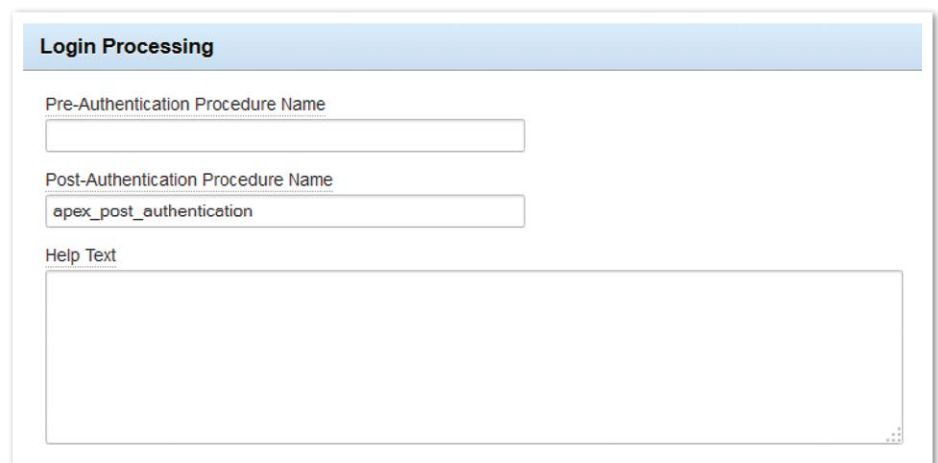


Abbildung 5: Post-Authentication Procedure

ein Package-Name als Qualifizierer vorangestellt werden. So schafft man eine leicht nachvollziehbare Ordnung für alle Initialisierungsskripte (siehe Abbildung 5).

Die Prozedur „apex_post_authentication“ bietet über die „v()“-Funktion Zugriff auf Variablen des Apex-Kontexts wie beispielsweise „v(,APP_USER‘)“. Vielleicht möchte man diese Prozedur als Verteilerstation für weitere Prozeduren einsetzen, um für jede Anwendung eine eigene „POST_AUTHENTICATION“ schreiben zu können – mit Fallunterscheidung anhand „v(,APP_ID‘)“.



Nutzungsbedingungen mittels Shortcuts und Subscription vererben

Zeigt die Apex-Anwendung den Benutzern auf vielen Seiten einen Fußnotentext an (zum Beispiel Nutzungsbedingungen), der überall gleich erscheinen soll? Dann gibt es mehrere Möglichkeiten, diesen Text zu speichern. Einigkeit sollte jedenfalls darüber bestehen, dass es langfristig keine gute Idee ist, eine HTML-Region damit zu bestücken und dann von einer Seite zur nächsten zu kopieren. Stattdessen könnte man die Nutzungsbedingungen aus einer Datenbank-Funktion abrufen und dabei entscheiden, ob man den Text dort fest codieren oder mittels einer Tabellen-Abfrage auslesen möchte. Wie auch immer, der Hinweistext kann in einer PL/SQL-Region als „http.p(disclaimer_text());“ ausgegeben werden.

Wer jedoch anwendungsrelevante Textblöcke lieber in Apex selbst bearbeiten und speichern möchte, und dafür spricht viel, der findet unter den Shared Components die ideale Stelle: Shortcuts. Für diesen Tipp erstellt man zunächst eine Mutter-Anwendung, die ausschließlich dazu dient, solche wiederkehrenden Anwendungsbestandteile zentral im Workspace zu verwalten und an alle verbundenen Anwendungen weiterzugeben. In dieser ansonsten leeren Mutter-Anwendung wird zunächst in den Shared Components (Rubrik „User Interface“) ein Shortcut namens „DISCLAIMER“ vom Typ „HTML Text“ erstellt und in dessen Eingabe-

```
... SELECT count(*)
   INTO v_count
   FROM tabelle;
IF v_count > 0 THEN ...
```

Listing 2

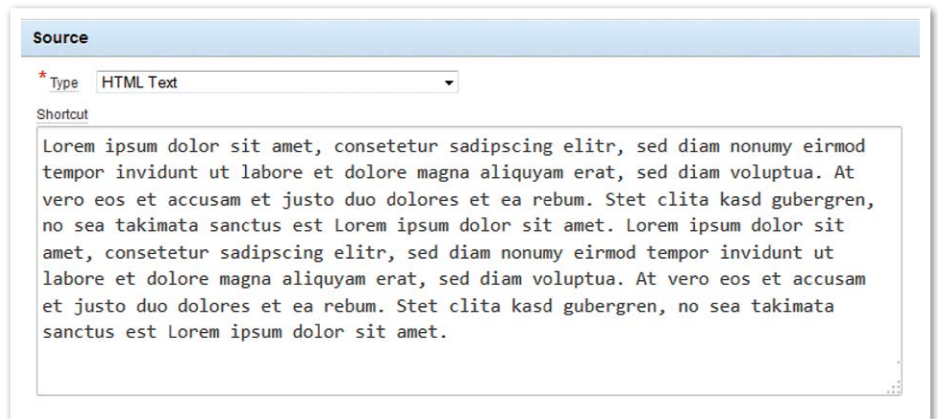


Abbildung 6: Der Shortcut

```
@-moz-document regexp(,.*apex/f\\?p=4(00|55)0:.*'),
  regexp(,.*apex/www_flow\\.accept.*')
{
  /** Unterstriche sind nicht immer sichtbar */
  input[type="text"] {
    line-height: 1.1em !important;
  }

  /** „Comments“ rechts unten fixieren */
  form section#COMMENT
,form section#COMMENTS {
    position: fixed;
    bottom: 10px;
    right: 10px;
  }

  /** Code-Eingabefelder: Schriftgröße erhöhen */
  textarea.textarea {
    font-size: 140% !important;
    line-height: 140% !important;
  }
}
```

Listing 3

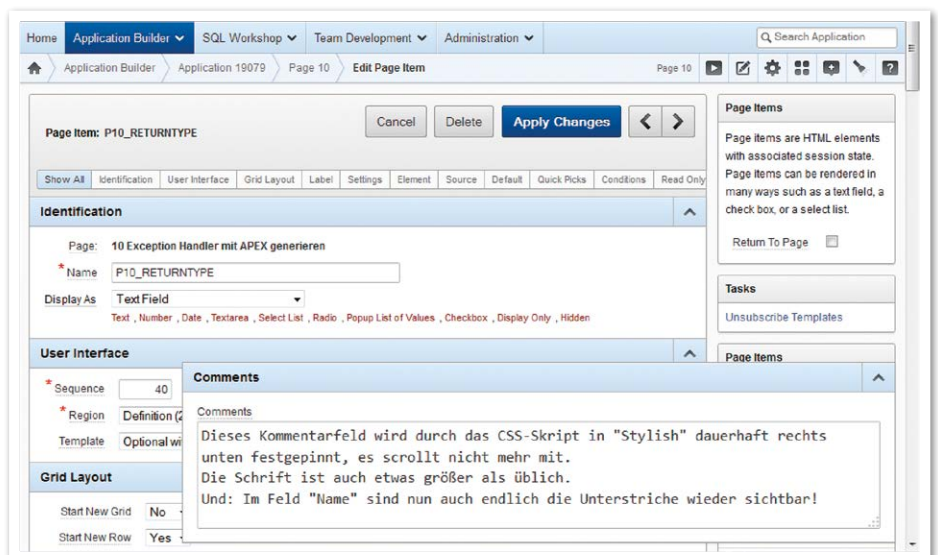


Abbildung 7: Der Nutzerstil-Manager „Stylish“

feld „Shortcut“ der gewünschte Text ohne irgendwelche Formatierungen eingegeben.

Um diesen Shortcut in Aktion zu sehen, erstellt man eine neue Region vom Typ „HTML Text (with Shortcuts)“ und schreibt in die Region Source lediglich „DISCLAIMER“ (in Großbuchstaben und inklusive der Hochkommata), wodurch Apex den Shortcut zur Laufzeit ausliest und dessen Text als Region Source ausgibt (siehe Abbildung 6).

Nun öffnet man eine bestehende Anwendung, die diesen neuen Shortcut verwenden soll. Einzige Voraussetzung dafür ist: Diese Anwendung und die Mutter-Anwendung befinden sich im selben Apex-Workspace. Angenommen, in einer bestehenden Seite ist bereits eine Footer-Region erstellt, in der sich der komplette Text der Nutzungsbedingungen befindet. Nun wird dieser Text gelöscht, durch die Zeichenfolge „DISCLAIMER“ (auch hier wieder in Großbuchstaben und mit Hochkommata) ersetzt und der Typ der Region von „HTML Text“ auf „HTML Text (with shortcuts)“ geändert.

Wohl wissend, dass die Region den Shortcut noch gar nicht auflösen kann,

wird dieser nun erzeugt. Anstatt über „From Scratch“ zu gehen, wählt man den Weg „As a Copy of an Existing Shortcut“, wählt als Quelle die Mutter-Anwendung und dort den „DISCLAIMER“-Shortcut. Der Namen der Kopie muss angepasst werden und der Shortcut in jeder der Zielanwendungen ebenfalls „DISCLAIMER“ heißen. Der entscheidende Klick ist die Auswahl von „Copy and Subscribe“. Dadurch speichert die Apex-Anwendung die Herkunft des Shortcuts ab und kann (auf Wunsch) die jeweils aktuelle Version nachladen. Diese Region kann man nun ohne Bedenken auf weitere Seiten kopieren.

Mit der Zeit entstehen letztendlich viele Anwendungen mit vielen Seiten, auf denen jeweils der Shortcut aus der Mutter-Anwendung zum Einsatz kommt. Um irgendwann den Text des Disclaimers in der Mutter-Anwendung zu ändern, gibt es durch die Subscription drei Möglichkeiten:

- Man ändert zwar den Text in der Mutter-Anwendung, unternimmt danach aber nichts. Dann erfolgt auch keine

Änderung in irgendeiner abhängigen Anwendung.

- Nach der Textänderung in der Mutter-Anwendung klickt man auf „Publish Shortcut“: Dadurch wird dieser Text an alle Anwendungen im selben Workspace publiziert, die einen entsprechenden Shortcut auf Basis dieses Mutter-Shortcuts erzeugt haben. Dies ist eine sehr komfortable Methode, um alles auf einen Schlag zu synchronisieren.
- Um, aus welchen Gründen auch immer, zunächst nur wenige ausgewählte Anwendungen zu synchronisieren, navigiert man innerhalb der betreffenden Anwendungen zum abhängigen Shortcut, klickt dort die Schaltfläche „Refresh Shortcut“ an und speichert anschließend.

Mit Subscription besteht jederzeit die Kontrolle darüber, ob, wann und wie die Quelle die Änderungen an die Zielanwendungen weitergibt. Übrigens, Shortcuts lassen sich sogar im Quelltext der Regions-Templates referenzieren. Vielleicht möchte man ja seinen Apex-Entwicklern lieber eine dedi-



zierte Regionsvorlage namens „DISCLAIMER“ zur Verfügung stellen, die diesen Shortcut intern bereits eingebaut hat. Dadurch ist sogar die Eingabe des Shortcut-Namens in die Region Source überflüssig.



Build-Options für unterschiedliche Skript-Versionen

Beim Rapid Application Development geht die Entwicklung zügig zur Sache. Der Autor gehört zu der Sorte Entwickler, die den jQuery-Code gerne selbst schreibt, anstatt sich nur auf Dynamic Actions abzustützen – Apex bietet halt für jeden Geschmack etwas. Deshalb tummeln sich mitunter Skripte auf seinen Seiten, die er nicht ständig benötigt und dann lieber abschalten würde.

Um zwischen den Skript-Variationen einer Seite auf die Schnelle umschalten zu können, erstellt man zunächst für die in Frage kommenden Skripte jeweils eine „No Template“-HTML-Region, kopiert den Code in die HTML-Source und ordnet jeder Region dann eine Build-Option (siehe „Shared Components“) zu, beispielsweise namens „JavaScript Test“.

Indem man die Build-Option ein- oder ausschaltet, wird gleichsam das entsprechende Skript auf der Seite ausgeführt oder nicht. Da der Autor zusammen mit seinen Skripten oft auch geeignete Anzeige-Elemente programmiert, ist es ganz praktisch, beides zusammen in derselben HTML-Region zu kapseln. Auf „Dynamic Actions“ lässt sich dieser Tipp natürlich ebenso anwenden.



Bedingungen mithilfe unsichtbarer Regionen schachteln

Es gibt die Aufgabe, dass ein Report als solcher nur angezeigt werden sollte, sofern die Basistabelle überhaupt Zeilen besitzt. Darüber hinaus muss man dies vielleicht mit einer weiteren Bedingung koppeln, beispielsweise mit der Auswertung einer Benutzereingabe. In diesem Fall lässt sich die Vorgabe „Exists“ aus der Bedingungs-Dropdownliste nicht verwenden, denn dadurch wären ja alle vorgegebenen Conditions bereits verbraucht. Man sucht einen Weg, beide Bedingungen in einem (PL)/SQL-Ausdruck zu kombinieren. Doch ausgerechnet der „exists“-Operator ist nicht ganz trivial zu codieren, wenn man die Performance der Abfrage im Auge behält. Oft sieht man Ansätze wie in *Listing 2*.

```
SELECT DISTINCT id
-- /// wieso ist die Abfrage nicht eindeutig?
FROM ...
```

Listing 4

SQL, PL/SQL	-- ///
JavaScript, jQuery	///
CSS	/* /// */
HTML, XML	<!-- /// -->

Tabelle 1

Bei großen Tabellen bringt das oft ungeahnte Verzögerungen mit sich, weil ja nicht die genaue Anzahl interessiert (eventuell wird hier ein ineffizienter Full Table Scan durchgeführt), sondern nur die Anwesenheit wenigstens einer Zeile. Eine pragmatischere Lösung wäre, eine neue Region um den betreffenden Report herum zu erstellen, die eine weitere Condition aufnehmen kann. Diese Region hätte selbst keinerlei HTML-Repräsentation (Template = „no Template“). Man würde von außen nach innen zuerst diejenige Condition prüfen, die Apex schneller beantworten kann; also immer die Auswertung der Benutzereingabe, da der Wert eines Items im Cache der Apex-Engine vorgehalten wird.

Erst wenn diese äußere Bedingung erfüllt ist, kommt die Auswertung der inneren Bedingung des Reports (mit der Vorgabe „Exists“) zum Zuge. Ein weiterer Vorteil dieses Ansatzes ist, dass man noch weitere Elemente als Geschwister des Reports anlegen könnte, die dadurch quasi gruppiert sind und so der umgebenden Region und deren Condition gehorchen. Auf diese Weise werden Code-Wiederholungen vermieden.



Stylish – den Browser an Apex anpassen

Für Schulungszwecke (und weil der Autor mittlerweile Brille trägt) wünscht er sich eine etwas größere Schriftgröße in den Apex-Code-Eingabefeldern. Beim Erhöhen der Zoomstufe wächst der Rest der Seite ja ebenfalls mit. Doch auch hier bietet das Web Lösungen. Eine davon ist der Nutzerstil-Manager namens „Stylish“ (siehe „www.userstyles.org“). Er ist als Add-on

für Firefox und als Erweiterung für Chrome verfügbar und ermöglicht es, individuelle CSS-Stile auf ausgewählte Webseiten anzuwenden. Je nachdem, wie individuell man es mag, lassen sich auf diese Weise die Seiten der Apex-Entwicklungsumgebung umfangreich anpassen.

Das Stylish-Skript für Firefox in *Listing 3* erhöht nicht nur die Schriftgröße in Textfeldern (übrigens behebt man damit auch das lästige Problem, dass im Firefox-Browser bisweilen die Unterstriche in Textfeldern nicht dargestellt werden), sondern bringt auch das Comments-Feld permanent rechts unten in den Vordergrund, sodass zum Eingeben und Lesen von Kommentaren nicht mehr ganz nach unten gescrollt werden muss.

Es kommt auf die ersten beiden Zeilen an: Dort wird festgelegt, auf welche Seiten des Apex Application Builder der Stil angewendet wird. Dieser reguläre Ausdruck und auch die CSS-Selektoren im Skript selbst müssen anfangs nicht hundertprozentig passen – denn wenn eine Apex-Seite einmal nicht gehorchen will, dann fügt man die fehlende Regel nach der Analyse der URL einfach im Laufe des Projekts hinzu (siehe *Abbildung 7*).



Dreifach schräg

Der Autor kennt keinen anderen Programmier-Kniff, der gleichzeitig so trivial und doch so hilfreich ist wie dieser: Man markiert alle Stellen im Quellcode, die einen gewissen Baustellen-Charakter haben, mit drei aufeinanderfolgenden Schrägstrichen (siehe *Listing 4*).

In SQL und PL/SQL gehört noch ein doppelter Bindestrich vorangestellt; in

den meisten Skriptsprachen wie etwa JavaScript kann das jedoch entfallen, weil der doppelte Schrägstrich ja ohnehin einen Zeilenkommentar einleitet (siehe Tabelle 1). Wer dahinter einen kurzen Kommentar anfügt, kann projektweit seinen Quellcode nach offenen Punkten durchsuchen. Der Autor hat sich angewöhnt, seine Baustellen je nach Dringlichkeit aufsteigend sogar mit bis zu sechs Schrägstrichen zu markieren. Das handhabt er übrigens auch in sämtlichen Office-Dokumenten so. Kommt er nun als Program-

mierer nach längerer Abwesenheit zurück, ist seine erste Aktion die Suche nach „//“ im Programmcode.

Dies ist gerade im Zusammenhang mit Apex so nützlich, angefangen bei Item-Labels, über die komplette Apex-Verarbeitungslogik, bis hin zu den gut versteckten jQuery-Skripten in diversen Dynamic Actions. Man kann diese Art der Schnellmarkierung überall verwenden, um alle offenen Punkte rechtzeitig vor Fertigstellung des Projekts sicher und bequem über die Apex-Volltextsuche wiederzufinden.



Andreas Wismann
wismann@when-others.com

APEX connect
by DOAG

Zwei Tage, nur ein Thema.

Konferenz für APEX-Begeisterte
9. & 10. Juni 2015 in Düsseldorf

Kein Superheld ...
... nur das richtige
Handwerkzeug

Apex Community – da ist mehr drin

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

Apex zeichnet sich seit Langem durch seine sehr lebhaft entwickelte Community aus. Dort findet nicht nur ein reger Austausch statt; auch die gegenseitige Hilfe bei Problemen und Herausforderungen wird groß geschrieben. Dieser Artikel zeigt, was sich hinter dieser „Apex Community“ verbirgt und wie man sich austauscht.



Apex gehört zur Oracle-Datenbank und ist damit natürlich ein Thema für die DOAG – auch auf den DOAG Regionaltreffen, an denen jedermann kostenlos teilnehmen kann (siehe „<http://www.doag.org/de/events/regionaltreffen.html>“). Dies ist eine hervorragende Gelegenheit, insbesondere auch um andere Oracle-Anwender vor Ort kennenzulernen.

Darüber hinaus treffen sich die Apex-Entwickler (und solche, die es werden möchten) mittlerweile in ganz Deutschland mehr oder weniger häufig zu sogenannten „Meetups“. Diese Treffen, die formlos über die Webseite „www.meetup.com“ organisiert werden, erfreuen sich zunehmender Beliebtheit. Im deutschsprachigen Raum gibt es inzwischen acht Gruppen – Spitzenreiter ist die Münchner Gruppe mit mehr als sechzig Mitgliedern. Auf der Webseite „www.apexmeetups.com“ erhält man eine Übersicht, kommt zur Webseite der lokalen Meetup-Gruppe und kann sich einfach für das nächste Treffen anmelden.

Beispiele, Howtos, Tutorials

Bereits seit dem Jahr 2006 wird die Webseite der deutschsprachigen Apex- und PL/SQL-Community von Oracle betrieben. Etwa alle zwei Wochen gibt es auf „blogs.oracle.com/apexcommunity_deutsch“ einen neuen Tipp zu einem Apex-Thema mit konkreten Beispielen (siehe Abbildung 2). Nicht nur Oracle, auch die MT AG betreibt ein Apex-Portal mit zahlreichen De-

mos, Präsentationsfolien, Videos sowie Tipps & Tricks aus der Praxis. Es ist unter „<https://apex.mt-ag.com>“ erreichbar (siehe Abbildung 3).

Diskussionsforen

An erster Stelle der Diskussionsforen ist natürlich das Apex-OTN-Forum zu nen-

nen. Die eigentliche URL ist sehr lang, daher sei die Abkürzung von Martin D’Souzas Link-Sammlung („www.oracleapex.com“) genannt: „forum.oracleapex.com“. Es ist eins der meistfrequentierten Foren im OTN. Hier diskutieren Apex-Entwickler weltweit und auch das Development-Team ist mit dabei. Wer hier eine (Apex-)Frage stellt, be-



Abbildung 1: Apex-Meetups



Abbildung 2: Deutschsprachige Apex- und PL/SQL-Community



Abbildung 3: Das Apex-Portal der MT AG

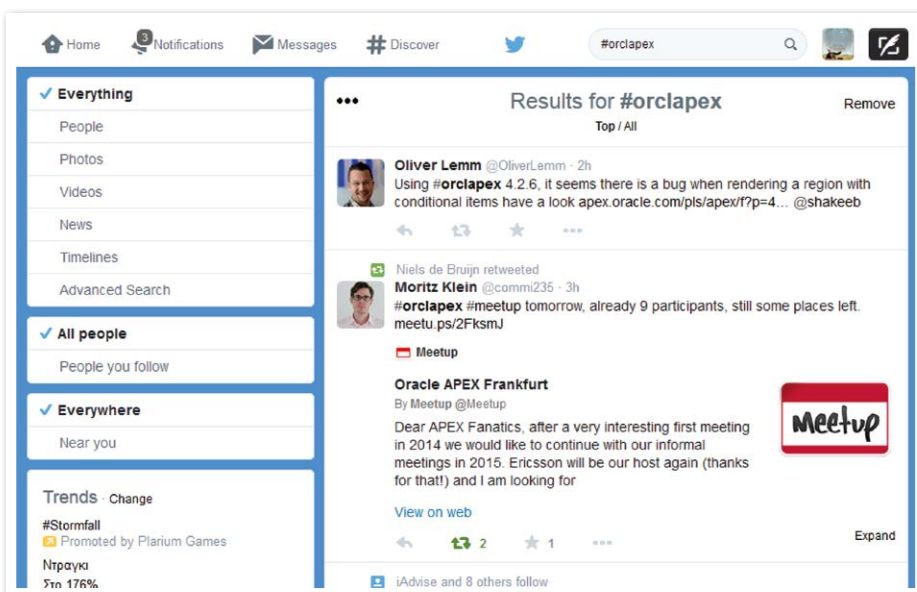


Abbildung 4: Apex auf Twitter

kommt fast immer eine schnelle und gute Antwort.

Der Name ist zwar veraltet, aber das Forum, das Ralf Merker unter „<http://www.htmldb.de/htmldb>“ betreibt, ist immer noch aktuell und man findet dort rege Diskussionen. Hier können Apex-Themen mit gleichgesinnten Entwicklern auf Deutsch diskutiert werden.

Gezwitscher

Der Kurznachrichtendienst Twitter ist eine wahre Fundgrube für interessierte Apex-Entwickler. Das Entwicklerteam und viele international bekannte Experten publizieren regelmäßig aktuelle Informationen auf Twitter.

Apex wird dort unter dem Hashtag „#orclapex“ geführt – sucht man also danach, findet man Tweets mit Informationen zu Apex – teilweise sind Neuigkeiten wirklich zuerst auf Twitter zu finden (siehe Abbildung 4).

Einige Twitter-Handles seien hier beispielhaft genannt – natürlich ist die Community viel größer – und es lohnt sich wirklich, noch mehr Personen zu folgen:

- Joel Kallman, Leiter des Apex-Entwicklerteams, @joelkallman
- Peter Raganitsch, bekannter Apex-Experte aus Wien, @peterraganitsch
- Roel Hartman versendet wöchentlich eine Apex-Zeitung, @roelhartman
- Las but not least, der Twitter Handle der DOAG, @doagev

Blogs

Nicht zu vergessen sind natürlich die zahlreichen Apex-Blogs, über die verschiedene Apex-Experten weltweit ihre Erfahrungen mit der Community teilen. Die amerikanische „Oracle Development & Tools User Group“ (ODTUG) hat alle Apex-Blogs auf „<http://odtug.com/apex>“ zusammengefasst. Ein paar Blogs seien hier dennoch erwähnt:

- Joel Kallmans Blog – was hier steht, kommt direkt vom Development der Oracle Corp., „<http://joelkallman.blogspot.com>“
- Denes Kubicek veröffentlicht ebenfalls regelmäßige Tipps – ein Geheimtipp, „<http://deneskubicek.blogspot.co.uk>“ und „<https://apex.oracle.com/pls/otn/f?p=31517:1>“

- Sehr lesenswert ist der Blog von Dimitri Gielis aus Belgien, „<http://dgielis.blogspot.co.uk>“

Big Events

Hinzu kommen natürlich die großen Treffen der Community: Auf der kommenden DOAG 2015 Apex Connect, am 9. und 10. Juni in Düsseldorf, sowie auf der DOAG 2015 Konferenz vom 17. bis 20. November in Nürnberg gibt es die Gelegenheit, andere Apex-Entwickler persönlich kennenzulernen – bekanntlich die beste Basis für gute Zusammenarbeit.

Fazit

Was auch immer man mit Apex umzusetzen versucht, in der Community findet man mit Sicherheit Unterstützung, wertvolle Ideen oder gar jemanden, der ein ähnliches Thema hat.

Es fehlt etwas? Ein lokales Treffen in Ihrer Gegend? Beispiele zu einem ganz bestimmten Thema? Die Community sind wir alle. Teilen Sie Ihre Erfahrungen, tauschen Sie sich aus, organisieren Sie ein Treffen. Sie werden viele Gleichgesinnte finden, die Sie bei Ihrer täglichen Arbeit weiterbringen.



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

Mobile My Oracle Support

Karl-Heinz Urban, ORACLE Deutschland B.V. & Co. KG

Dieser Artikel stellt einige der Funktionalitäten der neuen Mobile-My-Oracle-Support-Applikation vor. Diese wurde im Rahmen des letzten Support-Portal-Updates aktualisiert.

Mobil My Oracle Support (MMOS) ist eine Web-basierte Anwendung, die für mobile Geräte optimiert wurde und eine Teilmenge der Funktionen bietet, die das My-Oracle-Support-Portal für Anwender bereitstellt, die von unter-

wegs zum Beispiel auf ihre Service Requests zugreifen möchten. Die Applikation funktioniert am besten auf iPhone/iPad sowie Android-Geräten. Die Einstellungen im Browser des Geräts müssen so sein, dass der Browser „Cookies“ nutzen darf.

MMOS wird über den Link „<https://support.oracle.mobi>“ aufgerufen. Hier meldet man sich über den „Sign in“-Button (siehe *Abbildung 1*) wie gewohnt als My-Oracle-Support-Benutzer an. Anschließend erfolgt die Weiterleitung auf das Dashboard (siehe *Abbildung 2*).

Nach dem Anmelden werden basierend auf den jeweiligen Zugriffsrechten die Regionen angezeigt. Das Layout beinhaltet die Schnell-Navigation in den SRs, den Zugriff auf die Wissensdatenbank und die erweiterten Menüs. Folgende Gruppen können MMOS nutzen:

- My-Oracle-Support-Benutzer oder Support-Partner mit SR-View- und Update-

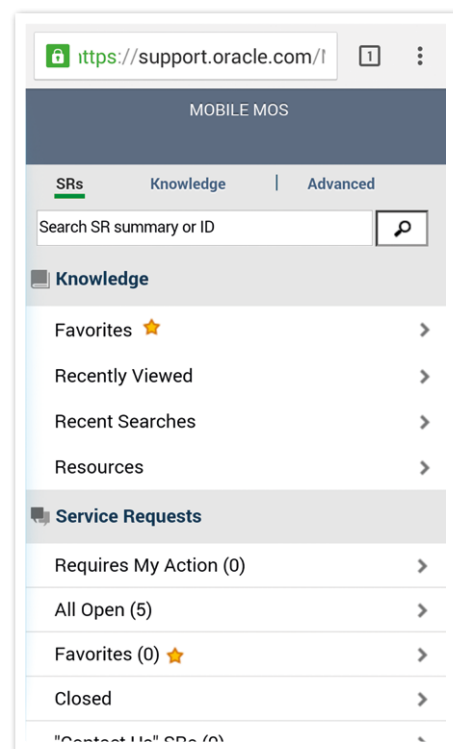


Abbildung 2: Das Dashboard

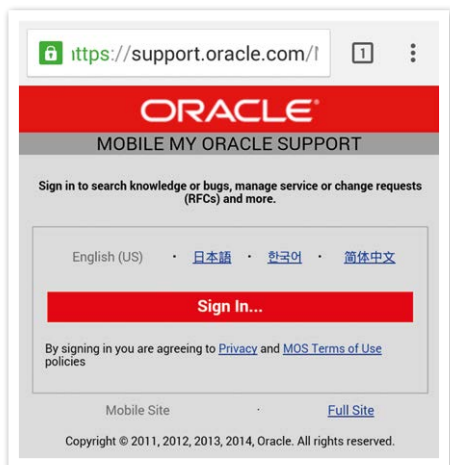


Abbildung 1: Die Anmeldeseite

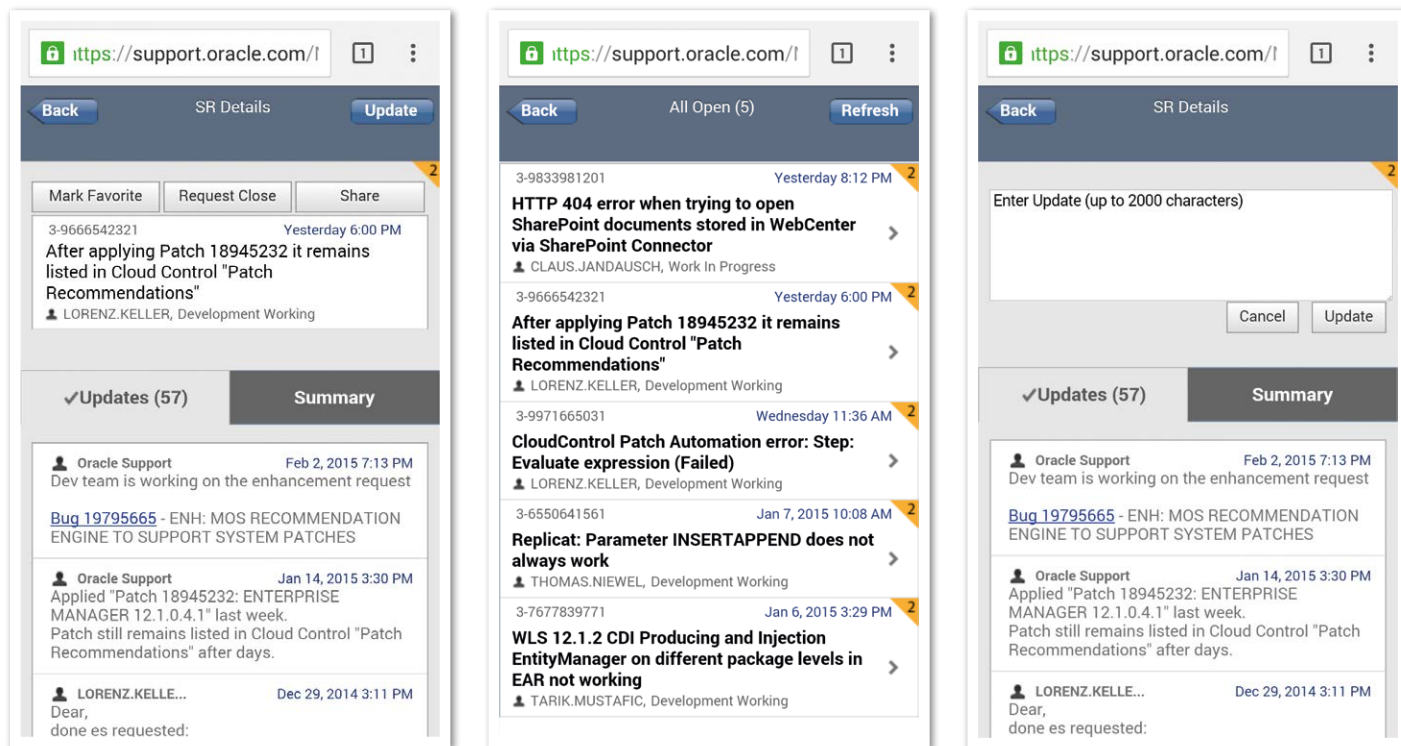


Abbildung 3: SR-Liste, SR-Details und SR Update

Rechten können SRs ansehen und aktualisieren

- Alle Benutzer können in der Wissens- und Fehler-Datenbank suchen
- Customer-User-Administratoren können Benutzerfragen bearbeiten
- Oracle-Managed-Cloud-Services können Requests for Changes (RFCs) ansehen, bearbeiten und planen

Zum Arbeiten mit Service Requests bieten sich folgende Möglichkeiten (siehe Abbildung 3):

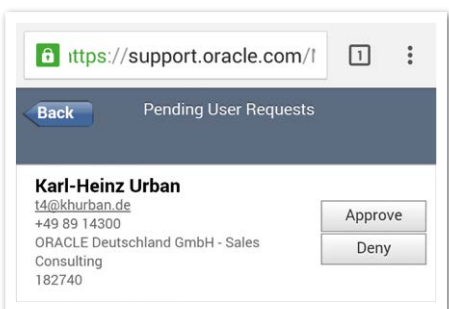
- Zugriff auf die neuesten Updates in eigenen SRs
- Beantwortung von Fragen der Support-Mitarbeiter und Aktualisieren der eigenen SRs
- Anlegen von personalisierten SR-Reports mit der erweiterten Suche
- Nutzung der erweiterten Suche zum Anlegen und Abspeichern der personalisierten SR-Reports

Unter dem Menüpunkt "Managed Cloud Services Change Requests" bieten sich folgende Optionen:

- Lesen und Aktualisieren der Change Requests

- Genehmigung von Change Requests anfordern
- Planen von Change Requests

Oracle bietet eine umfangreiche Wissens-Datenbank, auf die registrierte My-Oracle-Support-Benutzer auch in MMOS zugreifen können. Zur Suche stehen die bekannten Quellen wie die Wissens-Datenbank inklusive Archiv, die Fehler-Datenbank sowie das Sun-System-Handbook zur Verfügung. Es besteht natürlich auch die Möglichkeit, bei der Suche nach Produkt beziehungsweise Produktlinie zu filtern. Als letzte Option bietet sich die Suche nach „Task“ an, wobei dies nur bei der Suche in der Wissens-Datenbank und deren Archiv anwendbar ist.



Um die gefundenen Artikel oder Lösungen dauerhaft zur Verfügung zu haben, kann man sie durch Klicken auf das Sternensymbol als Favorit markieren und somit schnell wieder auf diese Information zugreifen, ohne erneut die Suchkriterien einzugeben. Generelle Einstellungen oder der Zugriff auf verschiedene Quellen wie Twitter, My-Oracle-Support-

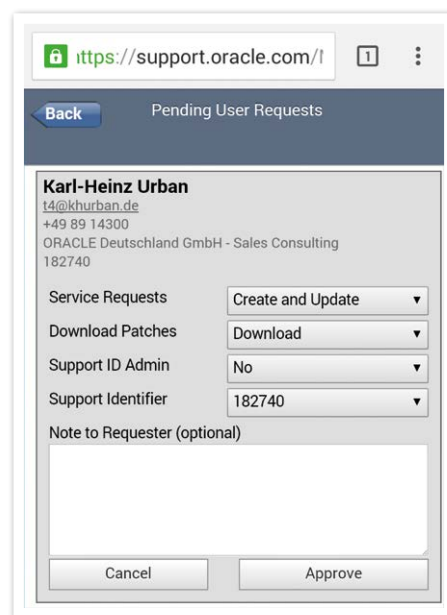


Abbildung 4: Pending User und Vergabe der Benutzerrechte

Blog, Hilfe etc. finden sich im Menüpunkt „More“.

Benutzer-Administration

Die Region „Benachrichtigungen“ erscheint auf der Startseite, wenn eine Aktion des Benutzer-Administrators erforderlich ist. Dies geschieht dann, wenn ein Benutzer Zugriff auf einen Support Identifier des Unternehmens beantragt. Der Administrator kann

diese Anfragen jetzt auch von seinem mobilen Gerät aus bearbeiten; genau wie in der Desktop-Version von My Oracle Support stehen dazu die Buttons „Approve“ und „Deny“ zur Verfügung (siehe Abbildung 4).

Weitere Informationen

Einführungsvideo: My Oracle Support How-to Series – How to use Mobile My Oracle Support (Doc ID 1683083.2)



Karl-Heinz Urban
karl-heinz.urban@oracle.com

Eine neue Generation von Engineered Systems

Frank Schneede, ORACLE Deutschland B. V. & Co. KG

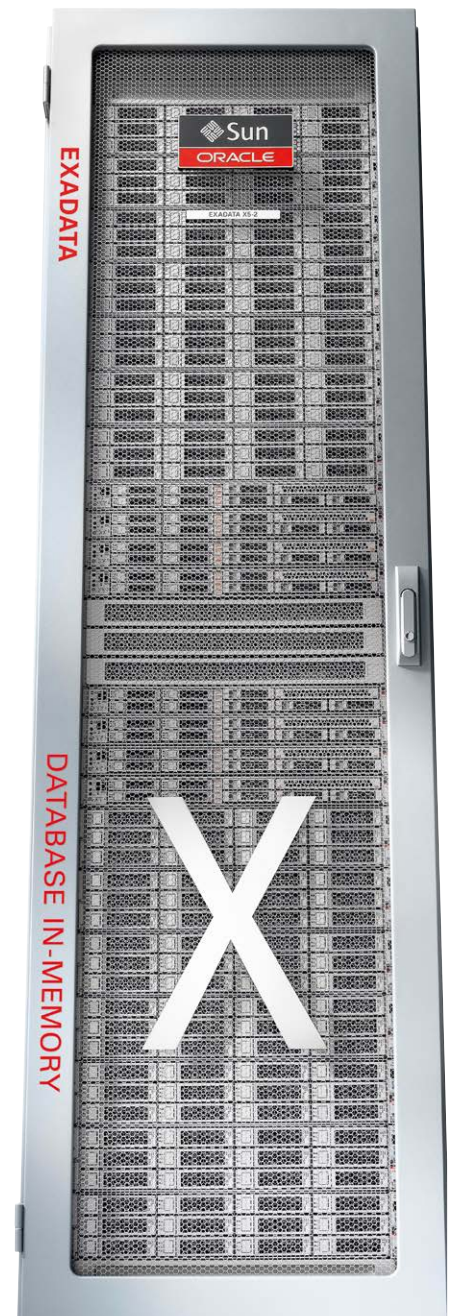
Nach der Vorstellung einer neuen Server-Generation X5-2 mit 2-Sockel-x86-Architektur Anfang Dezember 2014 war es nur eine Frage der Zeit, wann diese Server in der Oracle-Engineered-Systems-Familie zur Verfügung stehen würden.

Am 21. Januar 2015 war es soweit: Sämtliche Engineered-Systems-Modelle wurden auf aktuelle Hardware gebracht und mit ebenfalls aktualisierter Software der Öffentlichkeit vorgestellt. Neben der Exadata Database Machine X5, der mittlerweile sechsten Generation der Oracle-Datenbank-Maschine, und der Oracle Database Appliance stellt dieser Artikel auch das jüngste Engineered System vor, die Zero Data Loss Recovery Appliance. Neben der Vorstellung der Neuerungen in Hard- und Software zeigt eine Einordnung, wie sich die Systeme gegeneinander abgrenzen und welche Möglichkeiten sich durch die Nutzung der neuen Möglichkeiten für den Anwender bieten.

Basis der Neuankündigungen sind die unterschiedlichen Modelle des Oracle Servers X5-2. Ausgestattet sind die Server mit Prozessoren der „Intel Xeon Processor E5-2600 v3 Product Family“ in aktueller

Haswell-Mikroprozessor-Architektur mit bis zu 18 Prozessorkernen. Im Gegensatz zu den bisher verwendeten Prozessoren in Ivy-Bridge-Mikroprozessor-Architektur bieten die neuen Prozessoren eine verbesserte Energie-Effizienz und gesteigerte Leistung. So haben die Topmodelle der neuen Prozessoren nach „SPECint 2006“ eine im Vergleich zum Vorgänger um 47 Prozent gesteigerte und für den Einsatz als Datenbank-Plattform besonders interessante Multi-Thread-Performance.

Auch die Single-Thread-Performance ist um 16 Prozent gestiegen. Darüber hinaus werden mehr DIMM-Module (DDR4) unterstützt, womit die Server auf bis zu 768 GB Hauptspeicher aufgerüstet werden können. Zugleich unterstützt die neue Server-Generation nun auch „Non Volatile Memory Express“- (NVMe) und SAS3-SSDs. Es gibt sie jetzt in unterschiedlichen Bauarten – als Server mit 8,



12 oder 24 Platteneinschüben in ein oder zwei Höheneinheiten.

Exadata Database Machine X5-2

Durch die Verwendung des Exadata X5-2 Database Servers verfügt jeder Datenbank-Server mit zwei Eighteen-Core Intel Xeon E5-2699 v3 Prozessoren (2.3 GHz) nun über insgesamt 36 Prozessorkerne und lässt sich – ausgehend von der Standard-Konfiguration mit 256GB DDR4 DRAM – auf 512 GB oder 768 GB aufrüsten. Wartungsarme Kondensatoren ersetzen Batterien, die bislang den Festplatten-Controller des Servers versorgten und turnusmäßig ausgetauscht werden mussten. Auch der Exadata X5-2L Storage Server erfuhr eine Prozessor-Aufrüstung, es sind jetzt zwei Eight-Core Intel Xeon E5-2630 v3 Prozessoren (2.4 GHz) verbaut.

Die neue Modell-Generation bietet wieder zwei Varianten des Storage-Servers: Der Exadata Storage Server X5-2 High Capacity (HC) enthält die bereits aus dem Vorgänger bekannten 4-TB-SAS-Festplatten und verfügt über vier neue, mit Non Volatile Memory Express (NVMe) Protokoll angeschlossene 1,6TB Flash Accelerator F160 PCIe Cards, die doppelt so viel Speicher für den Exadata-Flash-Cache zur Verfügung stellen wie bisher. In der High-Capacity-Variante können

also je Datenbank-Server 48TB Rohdaten auf Festplatte und 4,8 TB Rohdaten auf Flash untergebracht sein; er bietet einen Durchsatz von 10 GB/s beziehungsweise 400.000 Lese- oder 192.000 Schreib-Operationen (8k, SQL end-to-end).

Der vollständig Flash-basierte Exadata Storage Server X5-2 Extreme Flash (EF) ersetzt das bisherige High-Performance-Modell und verfügt über acht 1.6 TB 2.5 Zoll Flash Accelerator F160 PCIe Drives, die ebenfalls über NVMe-Protokoll angeschlossen sind. Damit haben Kunden erstmals die Möglichkeit, für höchste Performance-Ansprüche auf ein Exadata-System zu setzen, das ausschließlich Flash-Devices als Datenbank-Storage verwendet. Ein Extreme-Flash-Storage-Server hat eine Kapazität von 12,8 TB Rohdaten und bietet einen Durchsatz von 18 GB/s beziehungsweise 400.000 Lese- oder 377.000 Schreiboperationen (8k, SQL end-to-end). Die Exadata Storage Server X5-2 werden mit identischen Konfigurationen auch in den verschiedenen Modellen des SPARC Superclusters eingesetzt.

Elastic Configuration

Die Exadata Database Machine ist nach dem Prinzip der „Well balanced Architecture“ entwickelt und stand so bislang nur

in fest definierten Ausbaustufen (Eighth, Quarter, Half oder Full Rack) zur Verfügung. Dieses Architektur-Prinzip, in dem zu jedem Zeitpunkt der Lebenszeit des Systems alle relevanten Ressourcen (CPU, Memory, Netzwerk, I/O) im Gleichgewicht sind, hat nach wie vor für die Mehrzahl der Kunden Gültigkeit. Es gibt jedoch Ausnahmen, in denen der Kunde zur Erfüllung besonderer Anforderungen auf eine sogenannte „Elastic Configuration“ zurückgreifen kann. Ausgehend von einer Quarter-Rack-Basiskonfiguration mit zwei Datenbank-Servern und drei Storage-Servern kann das System bis zum Maximalausbau von 22 Servern beziehungsweise 38 genutzten Höheneinheiten bedarfsgerecht um Datenbank- oder Storage-Server (HC oder EF) erweitert werden.

Die Montage einer Exadata mit Elastic Configuration erfolgt wie bei den Standard-Systemen bereits bei Oracle, sodass der Kunde nach wie vor von einer sehr zügigen Inbetriebnahme ausgehen kann. Nachträgliche Erweiterungen erfolgen beim Kunden, die dazu gewünschten Komponenten werden grundsätzlich mit der erforderlichen InfiniBand-Infrastruktur ausgeliefert. Die Elastic Configuration deckt auch extrem hohe Anforderungen wie zum Beispiel bei Database-In-Memo-

Exadata Database Machine X5-2

Elastic Configuration

- Basis-Konfiguration Half Rack
 - 2x X4-8 Datenbank Server (8 Sockel)
 - 3x X5-2 High Capacity Storage Server oder 3x X5-2 Extreme Flash Storage Server
- Zusätzlich bis 2 weitere X4-8 Datenbank Server (Max 4 / jeweils 5 RU)
- Zusätzlich bis 11 weitere X5-2 Storage Server (Max 14 / jeweils 2 RU)
 - Mix zwischen Extreme Flash und High Capacity Storage Servern möglich!
- (internes) Elastic Configuration Tool zur Illustration der Alternativen
- 2x Sun Datacenter InfiniBand Switch 36
- 36-port Managed QDR (40Gb/s) Switch
- 1 "Admin" Ethernet Switch
- Redundante Stromversorgungen

X4-8 In-Memory

DB Memory (MB)	DB Cores	Storage Disk (TB)	Raw Flash (TB)
8,192	480	288.0	38.4

**4x Database Servers
6x HC Storage Servers**

X4-8 Flash OLTP

DB Memory (MB)	DB Cores	Storage Disk (TB)	Raw Flash (TB)
8,192	480	0.0	102.4

**4x Database Servers
8x EF Storage Servers**

Abbildung 1: Exadata X5-2 in der Elastic Configuration

DOAG/SOUG News 02-2015 | 57

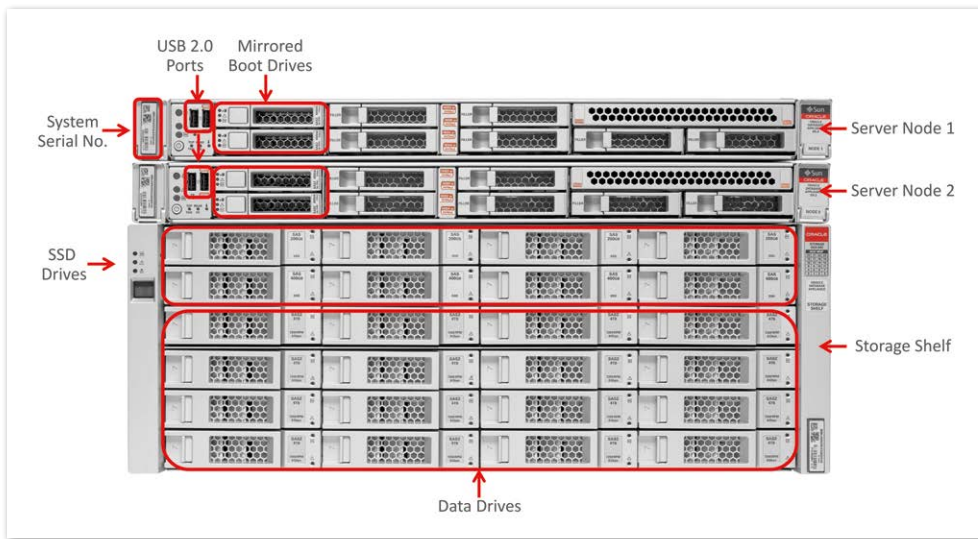


Abbildung 2: Aufbau der Oracle Database Appliance X5-2

bungen zu reduzieren. Das Einrichten der virtuellen Maschinen erfolgt mit dem aktuellen Oracle Exadata Deployment Assistant (OEDA), sowohl bei der initialen Implementierung als auch beim späteren Hinzufügen von virtuellen Maschinen. Der OEDA steht in Form des Exadata-Storage-Server-Patch Nr. 20305033 (Januar 2015) zum Download zur Verfügung.

Natürlich besteht mit der aktuellen Software auch für ältere Exadata-Maschinen ab Modell X2-2 die Möglichkeit, Oracle VM einzusetzen. Hierzu muss allerdings ein Re-Imaging der Datenbank-Server erfolgen, ein bloßes Aktualisieren der Software und des Betriebssystems reicht nicht aus.

Der Einsatz von Oracle VM ist auf der Exadata noch mit gewissen Einschränkungen verbunden. So stehen unter anderem Features wie OVM Live Migration, InfiniBand Partitioning oder OVM Manager nicht zur Verfügung. Als Gast-Betriebssystem ist lediglich Oracle Linux 6 freigegeben. Da die virtuellen Maschinen auf den lokalen Platten der Datenbank-Server platziert sind, ist das angegebene Limit von maximal 63 unterstützten VMs auf der X5-Plattform (16 VMs auf älterer Hardware) eher theoretischer Natur.

Exadata und Oracle VM sollen primär dazu dienen, den Kunden eine Möglichkeit zu bieten, den Isolationslevel konsolidierter Datenbanken zu erhöhen und

ry- oder OLTP-Systemen mit hohem I/O-Durchsatz ab.

Abbildung 1 zeigt beispielhaft zwei mögliche Konfigurationen, deren Leistungsdaten über ein Excel-basiertes Konfigurationsstool ermittelt und grafisch aufbereitet werden können. Die Elastic Configuration ist sowohl für X5-2- als auch für X4-8-Exadata-Maschinen verfügbar.

Software Refresh

Die Exadata-X5-Modelle erfordern die neue Exadata Storage Server Software 12.1.2.1.0. Neben umfangreichen Verbes-

serungen in Hinblick auf Performance und Verfügbarkeit bietet diese Version in Verbindung mit dem ebenfalls aktualisierten Betriebssystem Oracle Linux 6 Update 6 die Möglichkeit, auf der Exadata mit Oracle VM zu arbeiten.

Oracle VM isoliert Datenbanken gegeneinander, wenn eine strikte Trennung auf Betriebssystemebene beziehungsweise die Einhaltung festgelegter Grenzen in Bezug auf Memory/CPU sichergestellt sein muss. Trusted Partitions optimieren die Lizenzierung je VM, um die Lizenzkosten insbesondere für konsolidierte Umge-

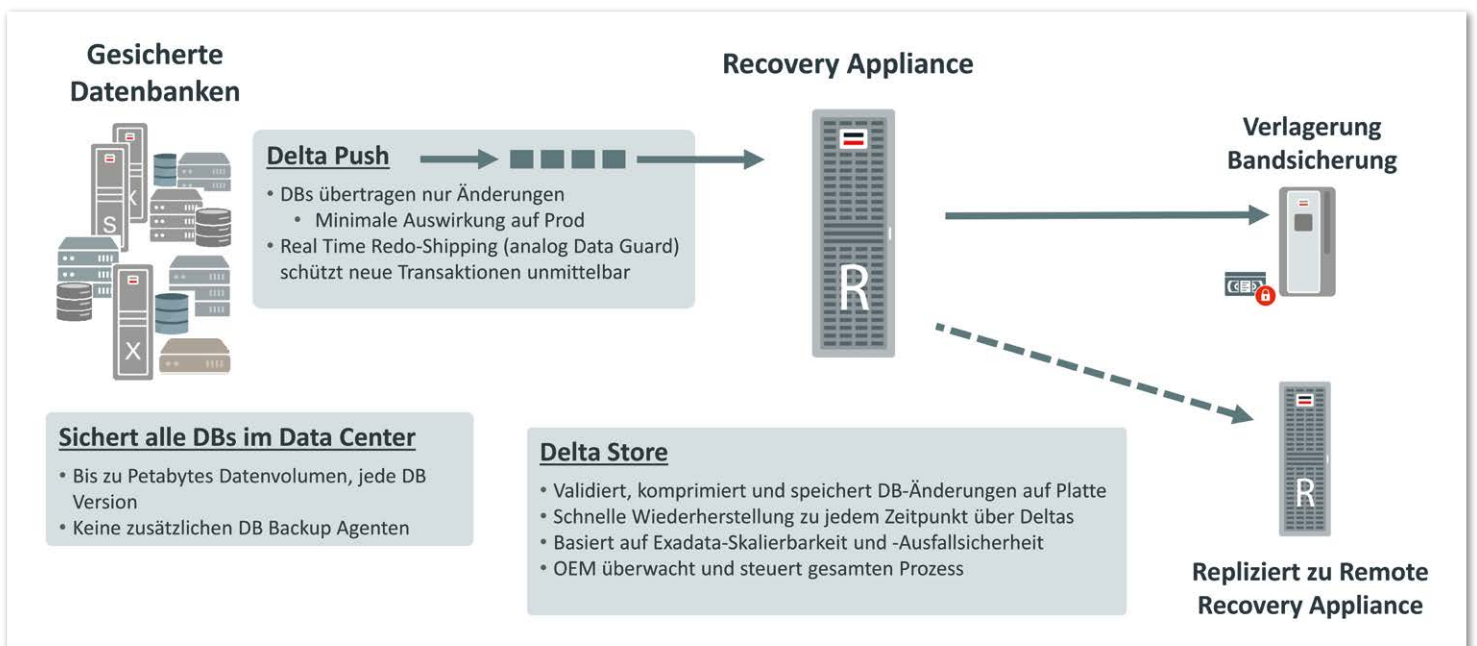


Abbildung 3: Zero Data Loss Recovery Appliance im Überblick

Lizenzkosten zu sparen – und nicht, um die Exadata Database Machine als virtualisierte Plattform für die Ressourcen-intensive Konsolidierung von Applikationsservern zu nutzen. Eine Nutzung als Plattform für schlanke ETL-Tools oder Security-Tools kann jedoch – eine Freigabe für Oracle Linux 6 vorausgesetzt – durchaus sinnvoll sein.

Mit der neuen Exadata Software 12.1.2.1.0 ist Solaris x86 als Betriebssystem auf der Exadata Database Machine abgekündigt worden. Die letzte Exadata-Software-Version, die Solaris x86 unterstützt, ist 12.1.1.1.1. Kunden, die derzeit auf das Solaris-x86-Betriebssystem setzen, werden bei der Migration auf Linux besonders unterstützt.

Die Exadata Database Machine ist das Flaggschiff der Engineered Systems für die Oracle-Datenbank. Mit nahezu unbegrenzter Skalierbarkeit, Virtualisierung und elastischer Konfiguration bietet sie den Kunden Einsatzmöglichkeiten für jedes denkbare Szenario.

Oracle Database Appliance X5-2

Die Oracle Database Appliance X5-2 (ODA X5-2) ist als Plattform für die Oracle-Datenbank sowie für Anwendungen konzipiert und verfolgt hierbei das Prinzip der Einfachheit, indem alle Einrichtungs- und Wartungsarbeiten über die grafische Oberfläche des Appliance Manager ausgeführt werden. Die aktuelle Version der ODA X5-2 verfügt über die gleichen Datenbank-Server-Modelle wie die Exadata mit zwei Eighteen-Core Intel Xeon E5-2699 v3 Prozessoren (2.3 GHz) und bis zu 768 GB DDR4 Hauptspeicher. Der Interconnect zwischen den beiden Datenbank-Servern ist in der X5-2 redundant über latenz-armes QDR-InfiniBand (40Gb/s) realisiert, die Verbindung besteht direkt zwischen beiden Servern und steht daher nur für diesen Zweck zur Verfügung.

Als Storage dient ein Shelf mit vier Höheneinheiten, dessen Kapazität im Vergleich zum Vorgänger erheblich vergrößert worden ist. Es bietet nun Platz für vier 200-GB-SSD-Festplatten, die für schreib-

intensive Redo-Logs vorgesehen sind, für vier 400-GB-SSD-Festplatten, die als Flash-Cache und Datenspeicher genutzt werden, und für 16 4-TB-High-Capacity-Festplatten für Daten oder Backups. Somit bietet das neue Modell rechnerisch also insgesamt Platz für 64 TB Daten, 800 GB Redo Logs und 1,6 TB Flash. Der zur Verfügung stehende Platz lässt sich durch die Erweiterung um ein zusätzliches Shelf im laufenden Betrieb verdoppeln.

Die ODA X5-2 besitzt insgesamt 2 bis 72 Prozessor-Kerne, die ohne Modifikation der Hardware bedarfsgerecht aktivierbar sind. Die Vergrößerung des Flash-Speichers ermöglicht eine vielfältige Nutzung dieses schnellen Speichermediums zur Steigerung der Datenbank-Performance. Der ODA-Flash-Cache fungiert hierbei als Database-Flash-Cache, also als Erweiterung der SGA. Damit werden Schreibvorgänge für Redo Logs auf Flash durchgeführt, das bedeutet eine etwa vierfache Performance-Steigerung im Vergleich zum Schreiben auf herkömmlichen Storage-Systemen.



**Digitale
Transformation**

Eintauchen, verstehen,
verändern

DOAG 2015 Business Solutions Konferenz

9. - 11. Juni 2015 in Darmstadt
Zwei Konferenztage & ein Schulungstag



business-solutions.doag.org

DOAG
Deutsche ORACLE-Anwendergruppe e.V.

ODA Flash Files ermöglichen es auf der ODA X5-2 erstmals, Datenbank-Files komplett auf Flash Storage zu legen. Wie auch bei der Exadata Database Machine ist es nun für die ODA X5-2 möglich, aus Verfügbarkeitsgründen den In-Memory Columnar Cache über RAC-Knoten hinweg zu duplizieren.

Überwachung und Verwaltung der ODA X5-2 ist über ein neues Enterprise-Manager-Plug-in möglich. Der Appliance Manager unterstützt die Einbindung in den Enterprise Manager. Der neue Appliance Manager 12.1.2.2, der in Kürze freigegeben wird, kann für alle bisherigen ODA-Modell-Generationen verwendet werden. Es kön-

nen Oracle-Datenbanken in den Versionen 11.2.0.2, 11.2.0.3, 11.2.0.4 und 12.1.0.2 eingerichtet und betrieben werden.

Die Oracle Database Appliance X5-2 wendet sich an Mittelstandskunden, die eine hochverfügbare, aber eben einfach einzurichtende und mit geringem Wartungsaufwand zu betreibende Plattform für

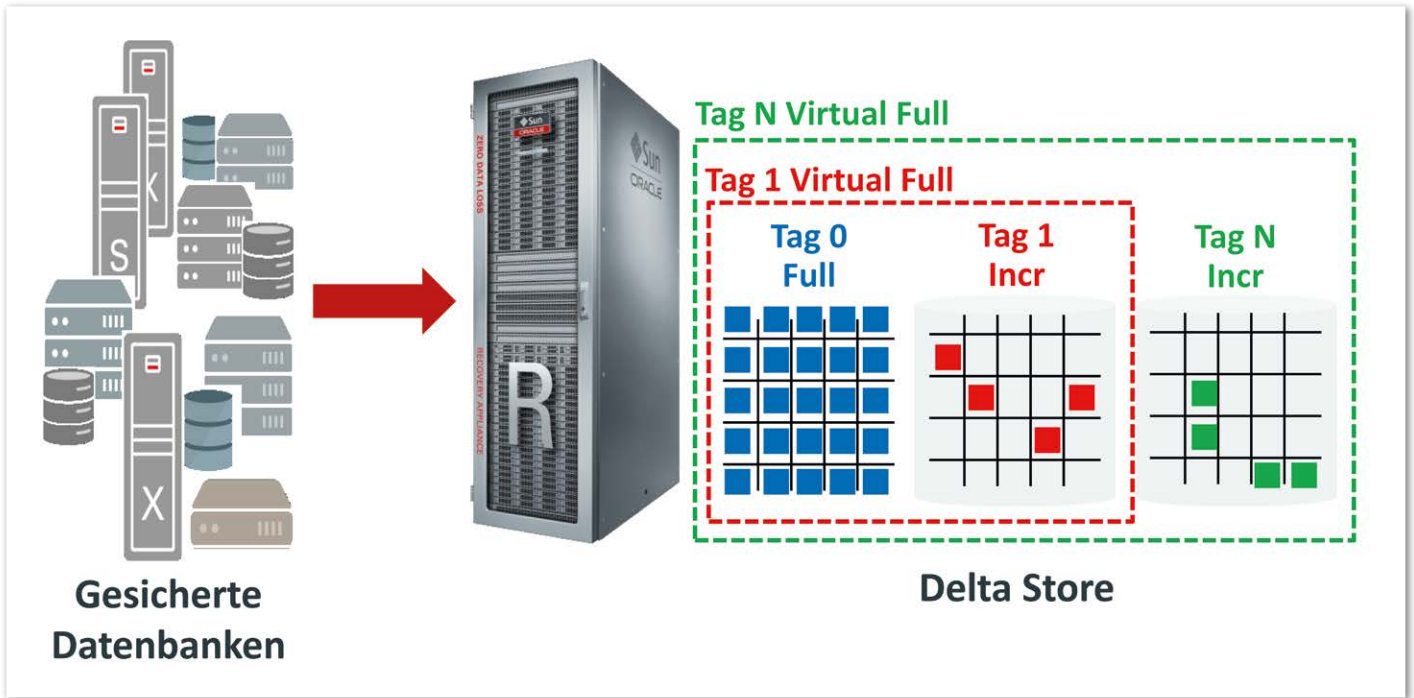


Abbildung 4: Virtual Full Backup

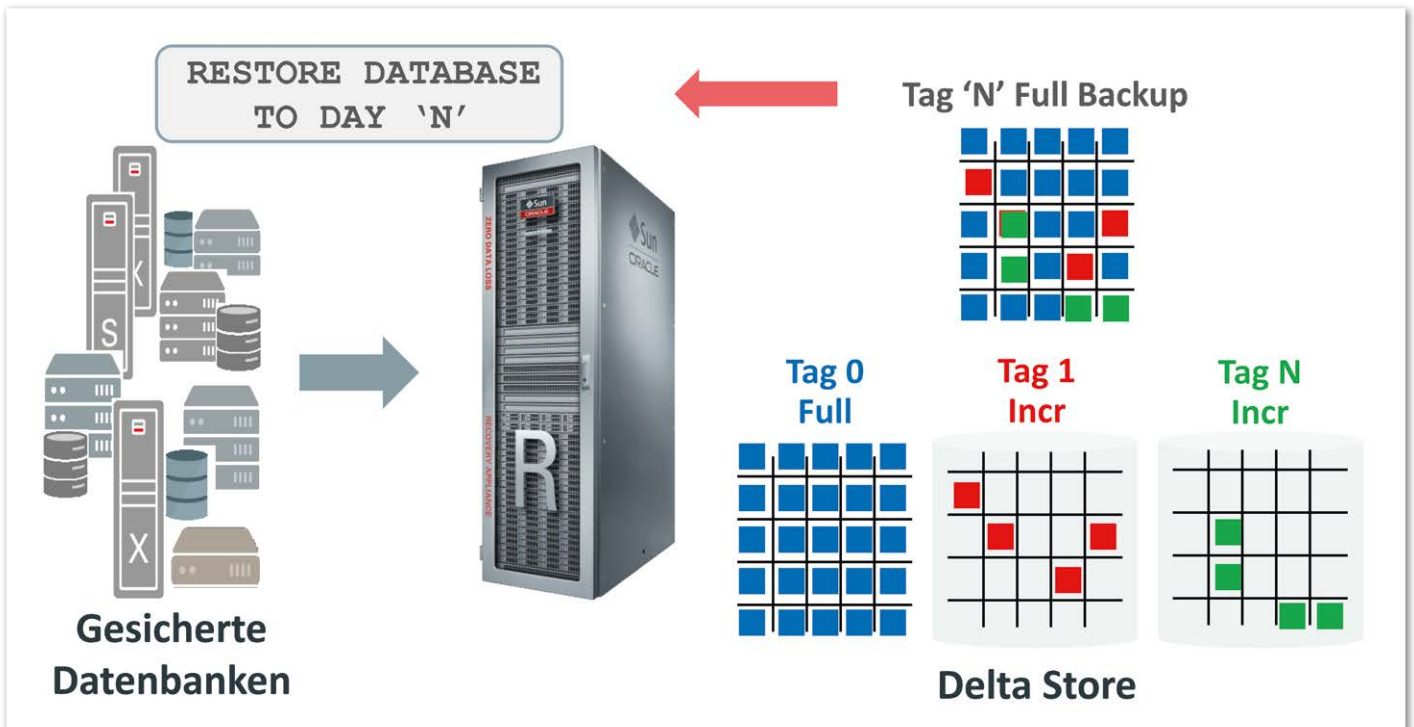


Abbildung 5: Recovery aus einem Virtual Full Backup

	Incrementally Updated Backups	Incremental Forever – Full Restore
Availability	Oracle Database 10g Release 2 >	Recovery Appliance only - Oracle Database 10g Release 2 >
Initial full backup type	Image copy	Backupset
Typical retention on disk	1 – 7 days	Days to months
Overhead on the database server	Yes – Merge (roll-forward) of incremental into full uses DB server resources	No – Delta Store processing is offloaded to the Recovery Appliance
SWITCH TO COPY capability	Yes	No
Recovery Point Objective (RPO)	Amount of time since last backup	Sub-second (when real-time redo transport is enabled)

Tabelle 1: Incrementally-Updated-Backups vs. Incremental-Forever-Strategie

Oracle-Datenbanken oder -Anwendungen suchen. Die ODA X5-2 bietet einschließlich Erweiterungs-Shelf mit einem nutzbaren Plattenplatz von maximal 64 TB (normale ASM-Redundanz) oder 42,6 TB (hohe ASM-Redundanz) für die meisten Mittelstandskunden ausreichend Platz-Reserven, auch wenn eine Erweiterung der Rechenkapazität systembedingt nicht möglich ist.

Zero Data Loss Recovery Appliance X5

Die Zero Data Loss Recovery Appliance (ZDLRA) ist das jüngste Mitglied der Engineered-Systems-Familie. Die Hardware-Ausstattung ist schnell beschrieben, denn diese besteht aus einem Exadata X5-2 Quarter Rack, auch „Base-Rack“ genannt, mit High-Capacity-Storage-Servern. Aufbauend auf den darin enthaltenen zwei Datenbank-Servern und drei Storage-Servern lässt sich das System bedarfsgerecht durch Hinzufügen weiterer X5-2-Storage-Server ausbauen.

Das Base-Rack der ZDLRA bietet Platz für 42 TB Rohdaten, jeder weitere Storage Server kann 18,5 TB Rohdaten aufnehmen. Damit stehen rechnerisch in einem voll ausgebauten Rack auf insgesamt 18 Storage Servern 320 TB Platz für Rohdaten zur Verfügung. Das entspricht bei einer Datenmenge von 320 TB und einer Änderungsrate von täglich zehn Prozent in etwa einem Zeitraum von zehn Tagen, für den ein Recovery zu jedem beliebigen Zeitpunkt möglich ist.

Die Besonderheit der ZDLRA liegt in der Konzeption der Maschine als zentrale Backup & Recovery Appliance für die Oracle-Datenbanken eines Unternehmens. Filesystem-Inhalte können derzeit noch nicht auf der ZDLRA gesichert wer-

den. Aktuelle Backup-Appliances decken den Bedarf, der bei der Sicherung von Datenbanken entsteht, häufig nur unzureichend ab und sind zudem recht kostenintensiv. Die Zero Data Loss Recovery Appliance adressiert genau die Themen, die für IT und Fachbereich wesentlich sind:

- Minimale Beeinträchtigung der Produktion durch Backup-Aktivitäten
- Optimierung von Recovery Time Objective (RTO) und Recovery Point Objective (RPO)
- End-to-End-Überwachung und Steuerung aller B&R-Aktivitäten
- Hohe Skalierbarkeit

Zentrale Komponente der Recovery Appliance ist der sogenannte „Delta-Store“. *Abbildung 3* zeigt dessen prinzipielle Funktionsweise.

In den Delta Store werden Backups übertragen, die mit dem Standard Oracle Tool Recovery Manager (RMAN) erzeugt worden sind. Darüber hinaus sind keine weiteren Backup-Agenten erforderlich. Nach einem initialen Vollbackup werden dann nur noch anfallende inkrementelle Sicherungen zur ZDLRA geschickt, wodurch die Produktion nur minimal belastet wird. Dieses Vorgehen wird als „Incremental Forever Strategie“ bezeichnet.

Im Delta Store erfolgt die Validierung, Komprimierung und Speicherung der inkrementellen Sicherungen in einem sogenannten „Virtual Full Backup“. Das bedeutet, dass auf der ZDLRA über intelligente Verarbeitung aus den unterschiedlichen Sicherungen eine Pointer-basierte Darstellung des Full-Backups zum Zeitpunkt des inkrementellen Backups erzeugt wird. *Abbildung 4* zeigt das Prinzip.

Im Recovery-Fall kann dann auf das benötigte Virtual Full Backup zurückgegriffen werden, ohne dass – wie bei einer herkömmlichen Backup-Strategie – auf das Einspielen eines Full-Backups und mehrerer nachfolgender inkrementeller Sicherungen zurückgegriffen werden muss (*siehe Abbildung 5*).

Neben der äußerst effizienten Ausnutzung des Speicherplatzes (man kann grob von zehnfach effizienterer Nutzung ausgehen) durch die Incremental-Forever-Strategie entsteht ein erheblicher Zeitvorteil durch den Verzicht auf das Restore der inkrementellen Sicherungen. So ergibt sich eine Reduzierung des Zeitaufwands für das Restore um bis zu 50 Prozent.

Ein Beispiel zur Verdeutlichung: Eine Datenbank wird klassisch nach dem System wöchentliches Full-/tägliches inkrementelles Backup gesichert, das vom Umfang her typischerweise jeweils rund zehn Prozent des Full-Backups ausmacht. Nach sechs Tagen haben die inkrementellen Sicherungen damit etwa 60 Prozent des Datenvolumens der Vollsicherung.

Das Restore von inkrementellen Sicherungen ist durch die unterschiedliche I/O-Charakteristik („random I/O“ statt „sequential I/O“) circa 40 Prozent zeitaufwändiger als das Restore eines Full-Backups. Somit ist das Restore nach dem klassischen Konzept in etwa doppelt so zeitaufwändig wie das Restore eines (Virtual)Full-Backups. *Tabelle 1* zeigt die Unterschiede zwischen einer herkömmlichen Sicherung und der Sicherung mittels ZDLRA auf.

Der neben dem Performance-Vorteil zweite große Nutzen der ZDLRA liegt in der Entlastung der Produktions-Datenbank von Backup-Prozessen. Diese laufen fast sämtlich auf der ZDLRA; Aufgaben

Oracle-Datenbank-Version	Möglichkeiten
Ab Oracle-Datenbank 11.1.0.7	ASYNCR redo transport to the Recovery Appliance Supported platforms: Linux, Windows and Solaris x86
Ab Oracle-Datenbank 11.2.0.4	Backup polling from big-endian platforms Backup polling supported from NFS-mounted FRA Redo transport for SPARC, Power, Itanium platforms Redo transport from a Standard Edition database Redo encryption during transport Data Guard Broker and SQL-based support for preserving redo transport on primary database after role transition
Ab Oracle-Datenbank 12.1.0.2	Cascaded redo from standby database SQL-based support for preserving redo transport on either primary or standby database after role transition

Tabelle 2: Redo-Transport auf der ZDLRA

wie „merge backup“, „De-Duplizierung“ oder „Komprimierung“ fallen nicht mehr an und tragen so erfahrungsgemäß zu einer Entlastung der Produktion um etwa 25 Prozent bei. Ausfallzeiten durch nächtliche Backup-Fenster können entfallen. Zudem werden Netzwerk-Ressourcen geschont, denn eine für die ZDLRA erstellte Sicherung wird in Form eines platzsparenden Backup-Sets erzeugt, während eine herkömmliche Sicherung Image-Copies verwendet.

Oracle-Datenbanken können ab Version 10g R2 auf der ZDLRA gesichert werden, ab Oracle-Version 11.1.0.7 aufwärts wird optional Redo-Transport unterstützt, der im „zero data loss“-Modus der ZDLRA den Namen gegeben hat. *Tabelle 2* zeigt die Möglichkeiten, aufgeschlüsselt nach der Datenbank-Version.

Auf der ZDLRA läuft eine Oracle-Datenbank mit RAC und ASM sowie auch Exadata-Software. Die zentrale Administration der ZDLRA erfolgt über ein kostenfreies Enterprise-Manager-Plug-in. Alle für den Betrieb der ZDLRA notwendigen Lizenzen sind bereits in der ZDLRA-Software-Lizenz enthalten, es fallen keine weiteren Gebühren an. Die ZDLRA ist ein Baustein in der „Maximum Availability Architecture“, um Oracle-Datenbanken auf

einem zentralen System schnell sichern und wiederherstellen zu können – eine Absicherung kritischer Produktionssysteme mittels Data Guard wird durch die ZDLRA jedoch nicht ersetzt.

Weitere Engineered Systems X5

Neben den hier ausführlich vorgestellten Engineered Systems aus dem Umfeld der Datenbank wurden in dem großen Announcement vom 21. Januar 2015 auch die Virtual Compute Appliance X5-2, die Big Data Appliance X5-2 und die Exalogic Elastic Cloud X5-2 basierend auf neuer X5-2 Hardware angekündigt. Die Virtual Compute Appliance bietet in Verbindung mit dem ebenfalls neuen Oracle FS1 Series Flash Storage System eine komplette, konvergente Infrastruktur, auf der mit minimalem Zeitaufwand Datenbanken und Anwendungen auch von unterschiedlichen Herstellern ausgerollt werden können.

In der Big Data Appliance sorgen doppelt so viel Hauptspeicher und mehr als doppelt so viele Prozessorkerne dafür, dass Anfragen noch schneller beantwortet werden können. Mit der Erweiterung von Oracle SQL auf Hadoop und NoSQL können nun SQL-Abfragen über alle Daten laufen, ohne zwischen Anwendungen wechseln zu müssen.

Aktuelle Hardware und neue, zum Teil erst in Kürze verfügbare Software-Funktionen machen die neuen Engineered Systems interessant für viele Einsatz-Szenarien. Über eingebautes InfiniBand lassen sich Engineered Systems miteinander verbinden und zum Beispiel im Fall von Exadata und Exalogic über den sogenannten „Exabus“ zu einer High-Performance-Infrastruktur verbinden, wie sie auch Oracle in der Oracle-Cloud anbietet.

Weiterführende Informationen

1. <https://www.oracle.com/engineered-systems/index.html>
2. <https://docs.oracle.com/en/engineered-systems/>



Frank Schneede
frank.schneede@oracle.com

Oracle präsentiert neue und erweiterte Produkte für Big-Data-Analysen

Oracle hat neue Big-Data-Lösungen vorgestellt, die den Zugang zu neuen Informationen und deren Auffindung vereinfachen. Anwender erhalten damit aus Daten

schnell wertvolle Informationen für ihr Geschäft. Zu den neuen Angeboten zählen Oracle Big Data Discovery, Oracle GoldenGate for Big Data, Oracle Big Data SQL

1.1 und Oracle NoSQL Database 3.2.5. Diese Erweiterungen sind Teil der Vision, wonach Hadoop-, NoSQL- und SQL-Technologien zusammenarbeiten.

Acht große Oracle-Datenbank-Mythen

Robert Barić, ITGAIN Consulting Gesellschaft für IT-Beratung mbH

Die IT steckt voller Mythen. Man denke nur an die sich in den Köpfen von Anwendern noch immer haltende Notwendigkeit, Datenträger regelmäßig defragmentieren zu müssen. Mit einer Solid State Disk ist dies überflüssig. Mancher Anwender glaubt auch noch, er surfe im „Privat Modus“ wirklich anonym, was definitiv nicht der Fall ist.

Technische Mythen basieren oft auf falschen Annahmen oder auf einem Wissensstand, der inzwischen durch weitere Entwicklungen überholt ist. Auch die Welt von Oracle steckt voller solcher Mythen. Höchste Zeit, sich einmal mit den überraschendsten eingehender zu beschäftigen.

Mythos 1: Count(*) ist böse

Für lebhaftes Fachgespräche sorgt immer wieder die Frage, ob „count(*)“, „count(1)“ oder „count(rowid)“ performanter sind, wenn es darum geht, die Anzahl der Zeilen in einer Tabelle zu bestimmen. Die beiden letztgenannten gelten bei einigen geradezu als Geheimtipp. Ein Mythos! Das Ergebnis und der Zugriffspfad sind bei allen drei Abfragen absolut identisch. Die Annahme, dass der Stern dabei alle Spalten berücksichtigt, ist schlicht ein Irrglaube.

Doch was passiert, wenn die Funktion „count“ über eine Spalte ausgeführt wird? Im Fokus stehen jetzt die Inhalte der jeweiligen Spalte und nicht die Zeilen. In diesem Fall werden nur die Zeilen gezählt, in denen die Werte der Spalte bekannt (also „not null“) sind. Unbekannte Werte („null“) werden nicht berücksichtigt. Dazu ein konkretes Beispiel, für das man eine neue Tabelle erstellt (siehe Listing 1).

Jetzt erfolgt die Abfrage der Zeilen mit vier Varianten des Count-Befehls:

1. Select count(*) from mytable;
2. Select count(1) from mytable;
3. Select count(rowid) from mytable;
4. Select count(x2) from mytable;

Table 1 zeigt, welches Ergebnis die verschiedenen Abfragen liefern. Das vielleicht

auf den ersten Blick überraschende Ergebnis der vierten Variante mit „count(x2)“ erklärt sich dadurch, dass in diesem Fall keine Null-Werte betrachtet werden und nun dieser Index, der keine unbestimmten Werte enthält, genutzt werden darf (siehe „<https://docs.oracle.com/database/121/SQLRF/functions003.htm#SQLRF20035>“).

Ändern sich jedoch auch die Ausführungspläne, wenn in der Spalte „x2“ keine Null-Werte zugelassen werden? Die Antwort lautet schlicht „ja“. Wenn die Spalte „x2“ mit einem „not null“-Constraint versehen ist und damit nur mit bestimmten Werten gefüllt werden kann, greifen alle vier Abfragen auf den Index zurück und liefern auch das gleiche Ergebnis. Mit „sqlplus“ und „autotrace“ lässt sich diese Aussage einfach verifizieren.

Mythos 2: Indizes speichern keine Null-Werte

In einigen Fundstellen der Literatur existiert immer noch die Aussage, dass Null-Werte nicht in Indizes gespeichert werden (siehe „<https://hoopercharles.wordpress.com/2012/02/28/repeat-after-me-null-values-are-not-stored-in-indexes>“ und „<http://www.toadworld.com/platforms/oracle/w/wiki/5874.indexes-and-null.aspx>“). Ein Mythos!

Null-Werte werden durchaus im B-Tree-Index gespeichert, sofern zumindest eine Index-Spalte keinen Null-Wert besitzt. Der Mythos hält sich wahrscheinlich nur deshalb, weil ein Index mit nur einer Spalte nie einen Null-Wert aufnimmt. Nachfolgend zwei Kommandos, bei denen ein Index Null-Werte enthalten kann. Das Kommando „create index i1 on mytable (x1,x2);“ erstellt einen zusammengesetzten Index, der Zeilen der Tabelle

Variante	Select	Operation	Ergebnis an Zeilen
1	Count(*)	Full Table Scan	300.000
2	Count(1)	Full Table Scan	300.000
3	Count(rowid)	Full Table Scan	300.000
4	Count(x2)	Index Fast Full Scan	270.000

Table 1

```
Create table mytable (x1 number, x2 number, x3 date)
Create index ind2 on mytable(x2)
Insert von 300.000 Zeilen mit beliebigen Werten.
```

Listing 1

```
create index i3 on mytable (x2, '1');
create index i3 on mytable (x2, 1);
```

Listing 2

referenziert, sofern nicht „x1“ und „x2“ gleichzeitig unbestimmt sind. In diesem Fall erfolgt kein Eintrag im Index.

Eine der beiden Varianten des Kommandos erzeugt einen zusammengesetzten Index mit einem Literal (siehe Listing 2). Da das Literal ein bestimmter Wert ist, wird jeder Wert in „x2“, egal ob bestimmt oder unbestimmt, im Index aufgenommen.

Listing 3 zeigt, wie die Ausführungspläne für die oben erwähnten Abfragen mit „count“ aussehen, wenn jeweils ein solcher Index für die im eingangs gezeigten Beispiel angelegte Tabelle „mytable“ ohne „not null“-Constraints erstellt wird.

Die Unterschiede bei den Kosten sind deutlich. Das kommt daher, dass bei der ersten Variante in beiden Spalten unbestimmte Werte vorkommen können und die Datenbank einen Full Table Scan verwendet, um alle Zeilen zu zählen. Bei der zweiten Variante mit einem Literal werden alle Zeilen im Index gespeichert und die Datenbank nutzt diesen, um alle Zeilen zu zählen.

Mythos 3: Deadlocks sind tödlich

Geraten zwei Sessions in der Oracle-Welt in eine Deadlock-Situation, endet dieser Zustand dadurch, dass eine der beiden Sessions terminiert wird und der Konflikt damit gelöst ist. Dieser automatische Killer-Auftrag ist ein weiterer Mythos, wie Charles Hooper aufgezeigt hat (siehe „http://hooper-charles.wordpress.com/2012/01/04/faulty-quotes-7-deadlock-kills-sessions“). Tabelle 2 zeigt das Beispiel.

Das möglicherweise überraschende Ergebnis: Weder Session 1 noch Session 2 werden terminiert. Und keine von beiden wird vollständig zurückgerollt. Lediglich die letzte Deadlock-Aktion wird nicht ausgeführt. Fragt man die Tabelle „T1“ in Session 2 nach dem Deadlock ab, erhält man folgendes Ergebnis (siehe Tabelle 3).

Mythos 4: If Exit = Rollback

Eine der vier sogenannten „ACID-Eigenschaften“ einer Datenbank besagt, dass für den Fall, dass eine Transaktion nicht ordentlich abgeschlossen werden kann, diese zurückgerollt wird. Ein Verhalten, das immer beobachtet werden sollte. Das sollte ja auch bei einer Oracle-Applikation gelten, wenn diese vorzeitig geschlossen oder absichtlich abgeschossen wird. Listing 4 zeigt das Beispiel eines Mythos.

```

create index i1 on mytable (x1,x2);
select count(*) from mytable; -> Resultat ist Full Table Scan

Execution Plan
-----
| Id | Operation          | Name | Rows | Cost (%CPU) |
-----
| 0  | SELECT STATEMENT   |      | 1    | 249 (1)     |
| 1  | SORT AGGREGATE    |      | 1    |             |
| 2  | TABLE ACCESS FULL| MYTABLE | 299K | 249 (1)     |
-----

create index i3 on mytable (x2,1);
select count(*) from mytable; -> Resultat ist Fast Full Index Scan

Execution Plan
-----
| Id | Operation          | Name | Rows | Cost (%CPU) |
-----
| 0  | SELECT STATEMENT   |      | 1    | 208 (1)     |
| 1  | SORT AGGREGATE    |      | 1    |             |
| 2  | INDEX FAST FULL SCAN| IND1 | 210K | 208 (1)     |
-----
    
```

Listing 3

Session 1	Session 2	Kommentar
Create table T1(C1 Number Primary Key, C2 Varchar2(10));		
Insert Into T1 values (1,'1');	Insert Into T1 values (2,'2');	
	Insert Into T1 values (1,'3');	Session 2 hängt
Insert Into T1 values (2,'2');		Session 1 hängt
	ORA-00060 deadlock detected	Session1 hängt weiter, Session 2 ist frei

Tabelle 2

Session 1	Session 2
...	...
	ORA-00060 deadlock detected
	SELECT * FROM T1;
	C1 C2
	----- -----
	2 2

Tabelle 3

Das Ergebnis: Die Insert-Operation wurde persistent übernommen, obwohl „auto-commit“ abgeschaltet ist. Dabei handelt es sich nicht um einen Fehler in der Datenbank, sondern um ein Standardverhalten vieler Oracle-Tools. Beim Verlassen der Applikation werden die Änderungen festgeschrieben. Dahinter steht wohl die Annahme, dass bei vielen Oracle-Tools das Beenden der An-

wendung als ein vom Benutzer gewollt abschließendes Ereignis gesehen wird und deswegen kein explizites Commit erforderlich ist (siehe „http://docs.oracle.com/cd/E29505_01/server.1111/e25789/transact.htm“).

Mythos 5: Weniger ist mehr

Ein klassischer Ansatz, eine Abfrage zu optimieren, besteht darin, die Abfragemen-

ge zu beschränken. Das lernen heute Kinder bereits in der Grundschule, wenn es um das Thema „Suchmaschinen“ geht. Die Oracle-Welt kennt solche Beschränkungen ebenfalls, damit etwa ein Index verwendet werden kann. Aber können zu viele Bedingungen in der „Where“-Klausel zu einer Verlangsamung der Abfragen führen? Dazu ein Beispiel von Marcus Winand (siehe „<https://use-the-index-luke.com/3-minute-test>“). Dabei werden zwei Abfragen angelegt, wobei die zweite durch eine zusätzliche Einschränkung noch präziser ist (siehe Listing 5). Die Abfrage liefert 100 aus 1.000.000 Zeilen.

Die Abfrage aus Listing 6 liefert 10 aus 1.000.000 Zeilen. Inwiefern hat aber die weitere Einschränkung Auswirkungen auf Kosten beziehungsweise Performance? Läuft die Abfrage schneller, langsamer oder macht das keinen Unterschied? Oder hängt das Ergebnis von den Daten selbst ab?

Die eigene Intuition dürfte den meisten sagen, dass eine größere Einschränkung positiv ist. Aber ist die Performance nicht

vielleicht doch auch eher von den Daten abhängig? Das überraschende Ergebnis: Die weitere Bedingung in diesem Beispiel verschlechtert die Performance. Es ist also ein Mythos zu glauben, dass mit zusätzlichen Einschränkungen in den Abfragen die Ergebnisse grundsätzlich schneller oder kostengünstiger vorliegen.

In diesem Beispiel muss durch die zusätzliche Einschränkung auch auf den Index der Tabelle zugegriffen werden. Der Aufwand durch den Tabellen-Zugriff steigt erheblich, wie sich durch einen Vergleich der beiden Ausführungspläne einfach zeigen lässt (siehe Listing 7).

Mythos 6: Der veraltete Index

Ob es etwas damit zu tun hat, dass viele Anwender ein personalisiertes Verhältnis mit Computern pflegen? Der Gedanke, dass Software und Programme wie wir Menschen altern, ist nicht nur im Umgang mit Windows, Word & Co. anzutreffen. Auch manche Administratoren machen sich immer wieder Gedanken über das Alter einer Datenbank. So sollen Indizes mit der

Zeit altern und aus der Balance geraten. Wächst der Index stärker auf einer Seite in einem bestimmten Werte-Bereich, vergrößert sich auf dieser Seite auch die Tiefe. Unterschiedlich tiefe Baum-Strukturen erhöhen aber den Verwaltungsaufwand.

Ein weiterer Faktor, der den Alterungsprozess des Index in die Wege leitet, besteht darin, dass gelöschte Elemente im Index verbleiben und nur als gelöscht markiert werden. Anders als beim Menschen gibt es hier zumindest Linderung dieser typischen Altersbeschwerden. Der Index-Rebuild sorgt für Heilung (siehe Abbildung 1).

Die Sache mit der mangelnden Balance ist ein Mythos. Er gründet auf einem Missverständnis. B-Tree steht bei Oracle nicht für „Binary“, wie erstaunlich viele Nutzer fälschlicherweise annehmen, sondern für „Balanced“ (siehe „http://docs.oracle.com/cd/E11882_01/server.112/e40540/indexiot.htm#CNCPT1170“). Ein nicht ausbalancierter balancierter Baum? Das klingt geradezu nach einem Widerspruch in sich. Ein einfacher Algorithmus sorgt dafür, dass es keine unbalancierten Bäume geben kann.

Die Erklärung: Wenn neue Einträge einen Index-Block vollständig auffüllen, wird der Block geteilt. Es entstehen zwei neue Blöcke und damit ein Split. Allerdings entsteht nur beim Split des ersten Blocks (Root) eine neue Ebene. Alle anderen Splits geschehen auf dem gleichen Level (siehe Abbildung 2).

Beim Verlauf des Vorgangs wird deutlich, dass bei diesem Verfahren nur balancierte Bäume entstehen können. Auf allen drei Ebenen werden Splits bei den Blöcken „R“, „A“ und „4“ ausgeführt. Nur bei „R“ entsteht eine neue Ebene und damit eine größere Tiefe des Baumes (siehe Abbildung 3).

Doch auch die Alterung und der Verlust der Performance durch nicht mehr gebrauchte Index-Einträge ist ein Mythos in der Oracle-Welt. Der Gedanke basiert auf einer oder mehreren der nachfolgenden, falschen Annahmen:

- Entleerte Index-Blöcke können nicht wiederverwendet werden
- Gelöschte Index-Einträge werden nie wiederverwendet
- Ein Index-Eintrag kann nur wiederverwendet werden, wenn ein neuer Eintrag in eine Lücke passt

```
SQL> set autocommit OFF
SQL> show autocommit
autocommit OFF
SQL> create table TEXTIT (a number);
SQL> Insert into TEXTIT values (1) ;
SQL> Exit;

#> SQLPLUS <User/Password>
SQL> select * from textit;

A
--
1
```

Listing 4

```
create index tab_idx on tbl (a, date_column);
Select date_column, count(*)
From tbl
Where a=123
Group by date_column;
```

Listing 5

```
Select date_column, count(*)
From tbl
Where a = 123
And b = 42
Group by date_column;
```

Listing 6

- Index-Blöcke können nur wiederverwendet werden, wenn neue Einträge in die erstellte Lücke passen
- Es ist ein (regelmäßiger) Index-Rebuild notwendig, um den Index performant zu halten

Tatsächlich werden Einträge wiederverwendet und sie müssen auch nicht genau in die zuvor ausgefüllte Lücke passen (siehe „<https://richardfoote.files.wordpress.com/2007/12/index-internals-rebuilding-the-truth.pdf>“).

Mythos 7: Ein Index-Rebuild tut immer gut

Treffen Administratoren und Entwickler aufeinander, ist die Diskussion rund um den Rebuild nicht fern. Schaden wird der Rebuild ja wohl wenigstens nicht. Das ist aber gerade nicht korrekt. Denn ein Index-Rebuild versucht Kosten:

- Der gesamte Index muss gelesen und neu geschrieben werden
- Kurzfristig wird bis zum Doppelten des Speicherplatzes benötigt; dabei kann der Tablespace anwachsen, wenn die Funktion „Autoextend“ aktiviert wurde
- Änderungen am Index generieren entsprechend Redo
- Beim Offline-Rebuild wird die Tabelle gesperrt
- Insert-Operationen benötigen Platz; ein Rebuild hingegen macht den Index in der Regel kompakter, wodurch weniger Platz zur Verfügung steht. Steht davon nicht genügend zur Verfügung, muss er erst durch neue Index-Block-Splits geschaffen werden, die wiederum Redo und zusätzliches I/O verursachen. Und genau das führt zu längeren Laufzeiten.

Mythos 8: Der Index-Rebuild ändert den Clustering-Faktor

Der Clustering-Faktor ist eine wesentliche Kennzahl für die Entscheidung des Optimizers, ob ein Index-Zugriff oder besser ein Full-Table-Scan angewendet werden sollte. Der Clustering-Faktor ist ein wesentlicher Faktor für die Kosten beim Index-Zugriff. Mit einem Index-Rebuild wird der Clustering-Faktor optimiert – oder doch nicht? Da der Clustering-Faktor ja zum Index gehört, muss sich dieser ja verändern, wenn der Index optimiert wird.

```
select date_column,count(*)
from tbl
where a=123
group by date_column;
```

Id	Operation	Name	Rows	Bytes	Cost (%CPU)
0	SELECT STATEMENT		100	2200	3 (0)
1	SORT GROUP BY NOSORT		100	2200	3 (0)
*2	INDEX RANGE SCAN	TAB_IDX	100	2200	3 (0)

Predicate Information (identified by operation id):

2 - access("A"=123)

```
select date_column,count(*)
from tbl
where a=123
and b=42
group by date_column;
```

Id	Operation	Name	Rows	Bytes	Cost (%CPU)
0	SELECT STATEMENT		126	4410	28 (0)
1	SORT GROUP BY NOSORT		126	4410	28 (0)
*2	TABLE ACCESS BY INDEX ROWID	TB1	126	4410	28 (0)
*3	INDEX RANGE SCAN	TAB_IDX	30		3 (0)

Predicate Information (identified by operation id):

2 - filter("B"=42)
3 - access("A"=123)

Listing 7

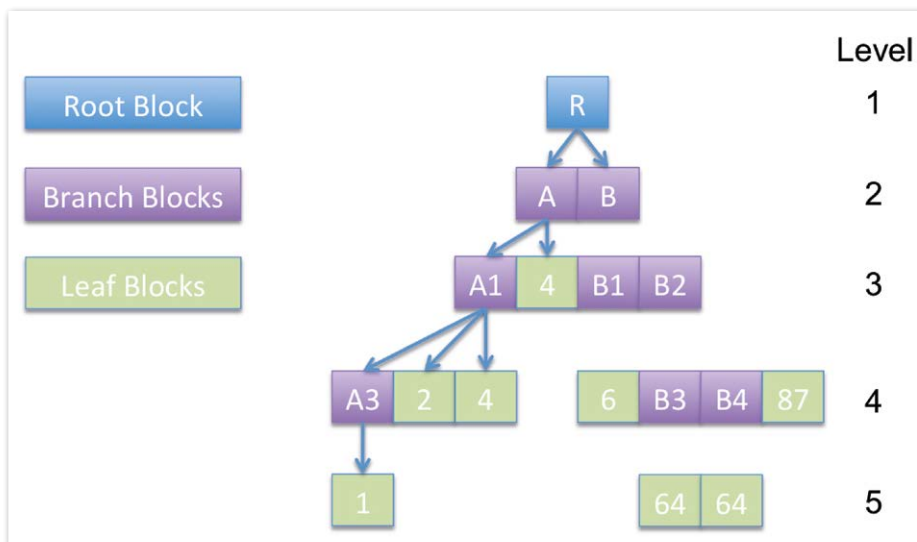


Abbildung 1: Ein unbalancierter Baum mit Blättern sind auf unterschiedlichen Ebenen (Level)

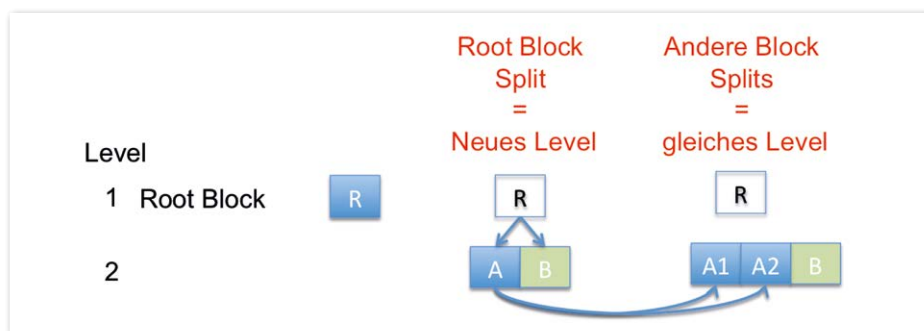


Abbildung 2: Der Split erfolgt nur einmal auf einer neuen Ebene

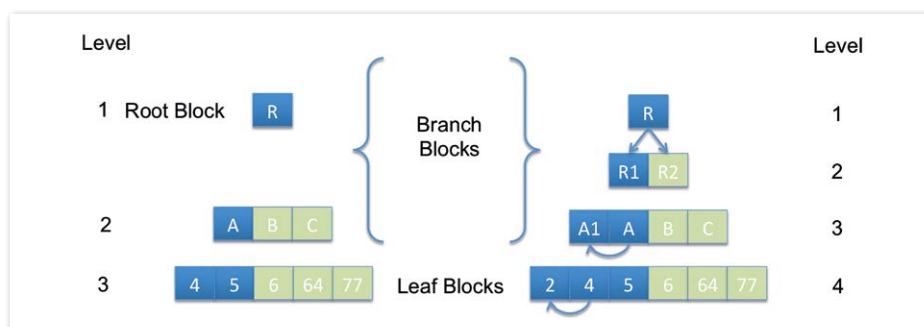


Abbildung 3: Nur bei „R“ entsteht eine neue Ebene

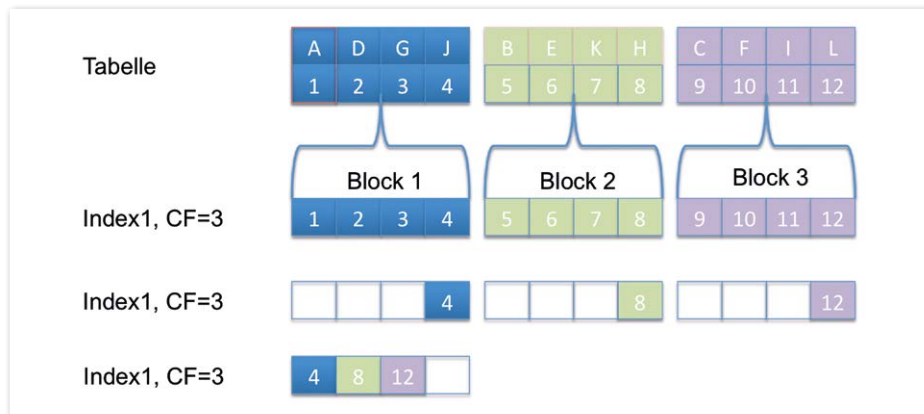


Abbildung 4: Der Clustering-Faktor

Dies ist ein Mythos, der sich aber auch aus alten Support-Dokumenten von Oracle speist (siehe „Support Doc ID 122008.1“). Denn dort wurde der Index-Rebuild tatsächlich als Optimierungsmaßnahme für den Clustering-Faktor beschrieben. Das ist indes schon lange nicht mehr so.

Der Clustering-Faktor beschreibt nichts anderes als die Ordnung zwischen der Tabelle und dem Index.

Dieser ist jedoch immer geordnet. Der Index ändert seine Ordnung weder vor noch nach oder zwischen einem

Rebuild. Das bedeutet dann allerdings auch, dass sich der Clustering-Faktor nicht ändern kann.

Zur Verdeutlichung dient das Schaubild in *Abbildung 4*, anhand dessen die grundlegende Berechnung des Clustering-Faktors aufgezeigt wird. Oben senkrecht sind die Zeileninhalte einer Tabelle, wie (A,1), dargestellt.

Die Blöcke einer Tabelle wurden farbig unterschiedlich markiert. Die Inhalte des Index1 sind mit der korrespondierenden Blockfarbe der jeweiligen Tabelle eingefärbt.

Vereinfacht: Der Clustering-Faktor wird durch den Wechsel der Blöcke der Tabelle bestimmt. Werden nun alle Zeilen außer denen mit den Werten „4“, „8“ und „12“ entfernt, ändert sich der Clustering-Faktor weder dabei noch nach einem Rebuild, da weiterhin der Blockwechsel bestehen bleibt.

Schluss mit den Mythen!

Diese kleine Auswahl an Mythen aus dem Oracle-Umfeld ist nicht neu, sondern gut dokumentiert. Dennoch spuken sie in den Köpfen vieler Anwender herum. Es gibt zwei Sätze, die wie Saatkörner zur Entstehung eines Mythos beitragen: „Das haben wir immer schon so gemacht“ und „Das ist technisch nicht möglich“.

Warum wurde etwas schon immer so gemacht? Warum wird es weiterhin so gemacht? Warum kann etwas technisch nicht sein?

Fazit

Die Entstehung von Mythen kann nur verhindert werden, wenn man das (eigene) Wissen infrage stellt. Nur weil einmal etwas in einer offiziellen Dokumentation geschrieben wurde, muss das nicht den Tatsachen entsprechen oder entsprochen haben. In diesem Sinne: Neugierig bleiben. Infrage stellen. Experimentieren.



Robert Barić
robert.baric@itgain.de

Oracle Restart: Die Grid Infrastructure für die Single Instance

Dr. Frank Haney, Jena

Oracle stellt eine Vielzahl von Hochverfügbarkeitslösungen zur Verfügung. Es ist dabei nicht trivial, die auszuwählen, die den jeweiligen Anforderungen am besten entspricht. Das gilt sowohl hinsichtlich der Verfügbarkeitsanforderungen als auch für wirtschaftliche Aspekte. Für einige ist die Grid Infrastructure die Basis.

Seit Oracle 11g R2 heißt die Clusterware „Grid Infrastructure“. Einer der wichtigsten Unterschiede zu vorherigen Versionen ist, dass ASM jetzt gebündelt mit der Clusterware- und nicht mehr mit der Datenbank-Software installiert wird, was die neue Namensgebung rechtfertigt. Folgende Varianten an Hochverfügbarkeitslösungen lassen sich auf der Basis der Grid Infrastructure realisieren:

- **Real Application Cluster (RAC)**
Die Datenbank läuft aktiv mit mehreren Instanzen auf verschiedenen Servern. Dabei handelt es sich um eine kostenpflichtige Option der Enterprise Edition. In der Standard Edition kann man sie ohne Zusatzkosten nutzen, sofern man die Beschränkungen der Standard Edition für das komplette Cluster einhält.
- **RAC One Node**
Hier darf die Datenbank nur auf einem Knoten mit einer Instanz aktiv laufen. Es handelt sich um eine Aktiv-Passiv-Lösung, die nur zusätzlich zur Enterprise Edition lizenzierbar ist, allerdings zu geringeren Kosten als das volle RAC. Im Bedarfsfall startet die Clusterware die Instanz auf einem anderen Knoten neu.
- **Failover-Datenbank**
Auch das ist eine Aktiv-Passiv-Lösung, bei der durch Skripte der Übergang der Datenbank-Ressource auf einen anderen Knoten erreicht wird. Das ist eigentlich nur noch interessant, wenn man in der Standard Edition die Lizenzbedingungen für RAC nicht erfüllen kann. Die Variante ist aber bezüglich des Supports nicht unproblematisch.

- **Fremdapplikationen**

Diese können mit der Grid Infrastructure hochverfügbar gemacht werden, ohne dass im Cluster eine Datenbank betrieben wird. Zu denken wäre hier an Application Server oder Enterprise Manager.

- **Oracle Restart**

Die Grid Infrastructure wird auf einem Stand-Alone-Server installiert. Es besteht nicht die Absicht, diesen zu einem Cluster zu erweitern.

Oracle Restart im Überblick

Das neue Feature in Oracle 11g R2 ist allerdings nur eingeschränkt in 11.2.0.1 nutzbar. Eine Implementierung ist erst ab 11.2.0.2 sinnvoll. Diese ist mit und ohne Verwendung von ASM möglich. Es gilt auch hier, dass die Grid Infrastructure den höchsten Release-Stand haben muss.

Oracle Restart ermöglicht den automatisierten Restart der Oracle-Instanz und aller mit ihr verbundenen Ressourcen (Listener, Services, ASM, ASM Diskgroups) unter Beachtung der Abhängigkeiten nach einem Crash beziehungsweise einem beliebigen Reboot. Dabei wird auch der Zustand vor dem Crash berücksichtigt.

Gestartet werden die Ressourcen nicht vom „init“-Prozess oder dem jeweiligen Windows-Dienst, sondern vom Oracle High Availability Service (OHAS). Dieser bezieht die Information für die Starteigenschaften der jeweiligen Ressource aus der Oracle Local Registry (OLR). Oracle Restart hat eine Anbindung an Oracle Data Guard, was bewirkt, dass eine Standby-Datenbank immer in der richtigen Rolle startet.

Die Einbindung von Services in die Restart-Architektur ist unter Berücksichtigung der Datenbank-Rolle möglich. Ein Client verbindet sich dann, eine entsprechende TNS-Konfiguration vorausgesetzt, in einer Data-Guard-Umgebung immer mit dem Server, der die primäre Rolle hat. Auf Hochverfügbarkeitsereignisse kann mittels Fast Application Notification (FAN) reagiert werden. Eine Verarbeitung der Events ist dann über den Oracle Notification Service (ONS) oder serverseitige Callout-Skripte möglich.

Auch andere (Oracle- und Nicht-Oracle)-Applikationen können als lokale Ressource in die Restart-Umgebung eingebunden werden. Dadurch lässt sich ihr Start automatisieren. Notwendige Abhängigkeiten werden berücksichtigt und FAN-Callouts sind auch für diese Anwendungen möglich. Besonders sinnvoll ist das beispielsweise für den Grid beziehungsweise Cloud Control Agent oder auch für Database Control.

In der Dokumentation von Oracle 12c R1 ist das Feature überraschend als „deprecated“ deklariert. Die folgenden Ausführungen beziehen sich erst einmal auf 11g. Die Spezifika der Verwendung unter 12c werden später erläutert.

Besonderheiten der Installation

Die Implementierung beginnt mit der Installation der Grid Infrastructure für einen Stand-Alone-Server. Aber Achtung, diese Installationsoption erwartet die Verwendung von ASM. Bei der Installation muss man wenigstens eine Disk für eine initiale ASM-Diskgroup angeben und es wird dann eine ASM-Instanz gestartet. Das ist

jedoch nicht zwingend, weil ein Stand-Alone-Server keinen Shared Storage erfordert. Wenn man kein ASM verwenden will, dann muss die Installations-Option „Software Only“ ausgewählt werden.

Die Installation läuft anschließend komplett durch. Zum Schluss muss wie bei den meisten Installationen das Skript „\$ORACLE_HOME/root.sh“ ausgeführt werden. Wenn man ohne ASM installiert, fordert dieses zu Ausführung eines Perl-Skripts auf, ein analoges Skript gibt es für die Cluster-Installation (siehe Listing 1).

Nach erfolgreichem Lauf des Skripts ist die Installation komplett und der High Availability Service (HAS) läuft. Das kann man mit dem Befehl „crsctl check has“ überprüfen. Anschließend kann man mit dem Net Configuration Assistant („netca“) einen Listener konfigurieren und starten. Dieser verwendet wie bei einer Cluster-Installation das Grid-Infrastructure-Home.

Dann werden die Datenbank-Software in ein eigenes Home installiert und eine Datenbank angelegt. Bei den Grid-Installa-

tionsoptionen muss man „Datenbankinstallation mit nur einer Instanz“ auswählen. Beim Anlegen der Datenbank mit dem DBCA wird diese zum Schluss als HA-Ressource in Oracle Restart registriert, was der einzige Unterschied zu einer Datenbank-Erzeugung ohne Restart ist. Wenn man das Resultat kontrolliert, erscheint folgende Ausgabe (siehe Listing 2).

Verwaltung der Restart-Ressourcen

Genauso wie beim Cluster sind die beiden wichtigsten Werkzeuge „crsctl“ für die Verwaltung der Grid Infrastructure und „srvctl“ für die Verwaltung der Ressourcen. Es ist nicht notwendig, hier die komplette Syntax zu beschreiben; das kann nachgelesen werden. Ein paar Besonderheiten seien jedoch erwähnt. So kennt zum Beispiel Oracle Restart bei „crsctl“ die Argumente „cluster“ und „crs“ gar nicht. Stattdessen wird das Argument „has“ verwendet. Es heißt also „crsctl check|start|stop has“.

Dieses Kommando überprüft den Status des High-Availability-Service, startet bezie-

hungsweise stoppt ihn und startet beziehungsweise beendet auch alle in der OLR registrierten Applikationen unter Berücksichtigung der dort hinterlegten Abhängigkeiten. Was nun, wenn Datenbank und Listener schon vor der Restart-Implementierung vorhanden sind? Dann kann man sie mit „srvctl add database“ beziehungsweise „srvctl add listener“ hinzufügen.

Oracle Restart und Data Guard

Es gibt eine Integration von Oracle Restart und Data Guard. Diese erfordert aber den Data Guard Broker. Oracle Restart startet die Datenbank immer in der registrierten Rolle („PRIMARY“, „PHYSICAL_STANDBY“, „LOGICAL_STANDBY“ oder „SNAPSHOT_STANDBY“) und berücksichtigt gegebenenfalls Abhängigkeiten. Hinzu kommt, dass bei einem Rollenwechsel alle Services gestartet beziehungsweise gestoppt werden, die für die entsprechende Rolle konfiguriert sind. Hier als Beispiel das Anlegen eines Service, der nur läuft, wenn die Standby-Datenbank in der Rolle der primären ist: „srvctl add service -d ora11r2sb -s servsb -l primary“. Der Service muss nach dem Anlegen einmal manuell mit „srvctl start service -d ora11r2sb -s servsb“ gestartet werden. Dabei gibt es jedoch zwei Probleme:

1. Bei einem Failover oder Switchover wird zwar die neue Rolle der Datenbank berücksichtigt, aber der Open-Modus in der Restart-Konfiguration nicht geändert. Das wird allerdings erst ein Problem, wenn es zu einem Reboot des Servers kommt. Für die Lösung gibt es drei Möglichkeiten:
 - Nach jedem Rollentausch (Switchover oder Failover) wird der Open-Modus manuell mit „srvctl“ geändert. Das ist für das Switchover eine akzeptable Lösung, für das Failover kaum.
 - Implementierung eines „FAN-Callout“, der die aktuelle Rolle der Datenbank ausliest und automatisch den Open-Modus ändert.
 - Gelöst in Patch 12596559 (on Top of 11.2.0.3.9 beziehungsweise in 11.2.0.4 etc.)
2. Ein Service für die Rolle „PHYSICAL_STANDBY“ lässt sich nicht starten, wenn die Standby-Datenbank nur ge-

```
hostname:/oracle/oragrid/11.2.0/grid# /oracle/oragrid/11.2.0/grid/perl/bin/perl
-I/oracle/oragrid/11.2.0/grid/perl/lib
-I/oracle/oragrid/11.2.0/grid/crs/install /oracle/oragrid/11.2.0/grid/crs/install/roothas.pl
```

Listing 1

```
$GRID_HOME/bin/crsctl stat res -t
-----
NAME                TARGET STATE         SERVER          STATE_DETAILS
-----
Local Resources
-----
ora.LISTENER.lsnr
                ONLINE ONLINE         ora11r264
...
Cluster Resources
-----
...
ora.ora11r2.db
                1          ONLINE ONLINE         ora11r264      Open
```

Listing 2

```
#!/bin/ksh
umask 022
FAN_LOGFILE=/tmp/`hostname`.log
echo $* "reported="`date` >> $FAN_LOGFILE &
```

Listing 3

mountet ist. Das ist ein erwartetes Verhalten. Es gibt zwei Lösungen:

- Verwendung von Active Data Guard (kostenpflichtige Option)
- Services nur für die Rolle „PRIMARY“ anwenden

Reaktion auf Hochverfügbarkeits-Ereignisse

Statusänderungen in der Restart-Umgebung können an die Clients (Applikationen) propagiert werden. Das ist das Prinzip von Fast Application Notification (FAN) und erfordert die Verwendung von Services. Beispiele für solche HA-Events sind:

- Service nicht erreichbar
- Ausfall des Servers
- Rollenwechsel der Datenbank

Es gibt verschiedene Wege zur Propagierung der Information an die Clients:

- Verwendung FAN-integrierter Oracle-Clients („JDBC“, „ODP.NET“, „OCI“)
- Kommunikation mithilfe des Oracle Notification Service (ONS)
- Server Side Callouts (benutzerdefinierte Callout-Skripte)

Letzteres soll kurz dargestellt werden, um das Prinzip von FAN zu verdeutlichen:

1. Das Skript erstellen. Das hier „callout.sh“ genannte Skript macht nichts anderes als die Events in ein Logfile zu schreiben (siehe Listing 3).
2. Skript nach „\$GRID_HOME/racg/usrco“ kopieren.
3. Testen, etwa mit „ps -ef | grep smon“ und „kill -9 <smon ospid>“.
4. Die generierten Events haben folgende Struktur, etwa für Crash und automatischen Restart der Instanz (siehe Listing 4).
5. Den Logfile parsen und eine Reaktion implementieren: E-Mail, Stopp der Applikation etc. Das kann natürlich auch direkt im Skript erfolgen.

Integration einer „Fremd“-Applikation

Oracle Restart bietet auch die Möglichkeit, andere Applikationen einzubeziehen. Am Beispiel Database Control sind dazu folgende Schritte notwendig:

1. Es muss ein Action-Script erstellt werden und dieses die Funktionen „start“, „stop“, „check“ sowie „clean“ für die zukünftige Ressource enthalten.
2. Das Skript wird getestet und nach „\$GRID_HOME/crs/public“ kopiert.
3. Die Ressource wird in der OLR registriert (siehe Listing 5).
4. Die Ressource muss mit „[oracle@ora11r264 bin]\$ crsctl start resource dbcontrol“ gestartet werden.
5. Man kann die Funktionalität testen, indem man den Prozess killt und nachsieht, ob Oracle Restart Database Control wieder startet.

Quo vadis Oracle Restart?

In der Oracle-12c-Dokumentation steht: „Oracle Restart is deprecated in Oracle Database 12c. Oracle Restart is currently restricted to manage single-instance Oracle databases and Oracle ASM instances only, and is subject to desupport in future releases. Oracle continues to provide Oracle ASM as part of the Oracle Grid Infrastructure installation for Standalone and Cluster deployments.“ Da das zu

Missverständnissen geführt hat, wird es in der MOS-Note 1584742.1 „Support Impact of the Deprecation Announcement of Oracle Restart with Oracle Database 12c“ näher erläutert:

- „Deprecated“ bedeutet in diesem Fall nicht, dass es bereits einen Ersatz gibt. Oracle weicht hier von der üblichen Vorgehensweise deutlich ab, siehe „Streams“.
- „Deprecated“ ist nicht gleich „Desupported“ – Oracle Restart wird nach wie vor voll unterstützt.
- Es gibt keinerlei Hinweis darauf, was Restart ersetzen könnte.
- Es muss aber einen Ersatz geben, schon wegen ASM auf Stand-Alone-Implementierungen.

Bei näherer Betrachtung ist der Stand der Dinge in 12c folgender:

- Es gibt keine grundsätzlich neue Funktionalität gegenüber 11g.
- Restart ist nach wie vor voll in den Installer integriert.

```
INSTANCE VERSION=1.0 service=ora12r1 database=ora12r1 instance=ora12r1
host=ora12r164bit status=down reason=FAILURE timestamp=2014-06-03
14:37:30
INSTANCE VERSION=1.0 service=ora12r1 database=ora12r1 instance=ora12r1
host=ora12r164bit status=up reason=FAILURE timestamp=2014-06-03
14:37:51
```

Listing 4

```
[oracle@ora11r264 bin]$ crsctl add resource dbcontrol -type local_resource
-attr "ACTION_SCRIPT=$GRID_HOME/crs/public/dbcontrol.sh, <weitere
Attribute>"
```

Listing 5

```
[oracle@ol16r01 ~]$ lsnrctl status
...
Service "pdb1" has 1 instance(s).
Instance "ora12r1", status READY, has 1 handler(s) for this service...
```

Listing 6

```
[oracle@ol16r01 ~]$ srvctl add service -h
Adds a service configuration to be managed by Oracle Restart.
Usage: srvctl add service -db <db_unique_name> -service <service_name>
[-pdb <pluggable_database>]
```

Listing 7

- Alle Tools („srvctl“, „crsctl“, „dbca“ etc.) unterstützen Restart auch mit der in 12c veränderten Syntax.
- Meldungen, Ausschriften etc. sind an 12c angepasst.
- Oracle Restart unterstützt Pluggable Database.

Pluggable Database und Oracle Restart

Die Pluggable Database läuft intern als Service (siehe Listing 6). Die Grid Infrastructure kennt diesen Service nicht, man kann jedoch einen Service anlegen, der auf die PDB zeigt und sich mit dieser verbindet (siehe Listing 7). Die überraschende Konsequenz ist: Wenn für die PDB ein Service konfiguriert und gestartet ist, dann wird die PDB beim Start der Container-Datenbank nicht nur gemountet, sondern es wird automatisch „READ/WRITE“ geöffnet. Das ist eine wesentlich elegantere Mög-

lichkeit für ein automatisches Startup der PDB als beispielsweise ein Startup Trigger. Auf diese Weise trägt ein eigentlich abgekündigtes Feature zur besseren Verwaltung neuer 12c-Funktionalität bei.

Fazit

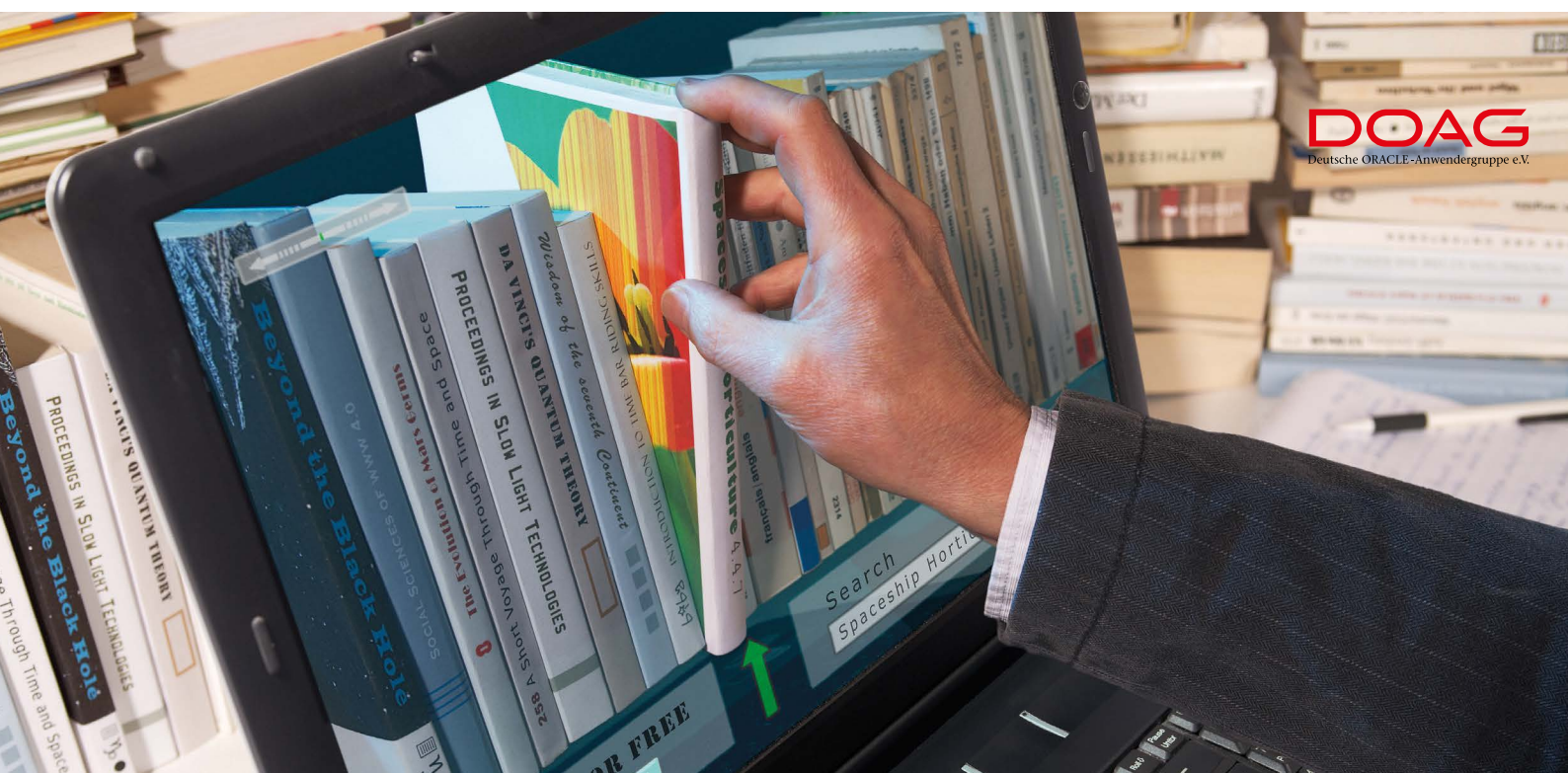
Es besteht kein Anlass, Oracle Restart in 12c nicht mehr zu verwenden. Oracle hätte sich vielleicht die Anmerkung in der Dokumentation verkneifen sollen, um Verwirrung zu vermeiden, zumal nicht erkennbar ist, wohin die Reise gehen soll.

Natürlich schafft Oracle Restart keine wirkliche Hochverfügbarkeit. Wie wir gesehen haben, hat aber der Einsatz der Grid Infrastructure für Single-Instance-Datenbanken durchaus Vorteile, nicht nur, weil damit der Einsatz von ASM möglich wird. Es sind vor allem die erleichterte Verwaltung, der automatische Neustart der Ressourcen, die Data-Guard-Integrati-

on und die übergreifenden Eigenschaften von FAN, die die Implementierung überlegenswert und sinnvoll machen.



Frank Haney
info@haney.it



DOAG
Deutsche ORACLE-Anwendergruppe e.V.

DOAG 2015
Datenbank





Tipps & Tricks aus Gerd's Fundgrube

Eine optimale Frame-Größe erzeugen

Gerd Volberg, OPITZ CONSULTING GmbH

Auf das Erzeugen eines neuen Blocks im Oracle Forms Builder folgt üblicherweise die langwierige Arbeit im Layout-Editor. Hier werden die letzten Änderungen an Größe, Position und Aussehen der darzustellenden Felder vorgenommen.

Die folgenden drei Schritte zeigen, wie man einen Block sehr schnell auf die optimale Breite vergrößern kann. Ausgangsbasis für das Beispiel ist ein Block, der gerade aus dem Data Block Wizard erstellt wurde (siehe Abbildung 1). Ziel ist es, diesen Block so zu vergrößern, dass die Abstände zum Fensterrand gleich groß werden.

In diesem Beispiel wird ein Frame für den Block benutzt, der seine Größe automatisch neu berechnet, wenn man „Update Layout“ aktiviert. Die optimale Breite des Blocks erreichen wir durch eine Vergrößerung des Beschreibungsfeldes. Man nimmt nun dieses Feld und zieht es ein wenig nach unten und ein wenig nach rechts. Wichtig ist, dass die rechte Kante des Feldes anschließend auf der gleichen X-Position ist wie beim Frame (siehe Abbildung 2).

Nun vergrößert man die Breite des Feldes, sodass der rechte Rand dort endet, wo zukünftig der rechte Rand des Frames

enden soll. Im letzten Schritt klickt man auf dem Frame mit der rechten Maustaste auf „Update Layout“ (siehe Abbildung 4). Nun hat man einen Block, der die optimale Breite des Fensters ausnutzt. Einfacher geht es nicht.

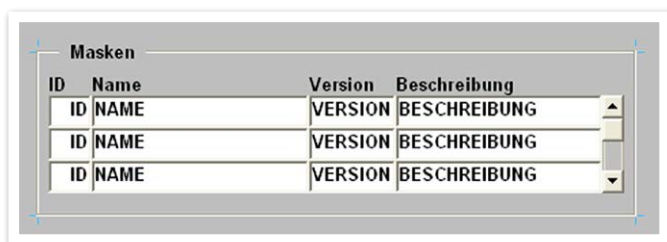


Abbildung 1: Der ursprüngliche Block

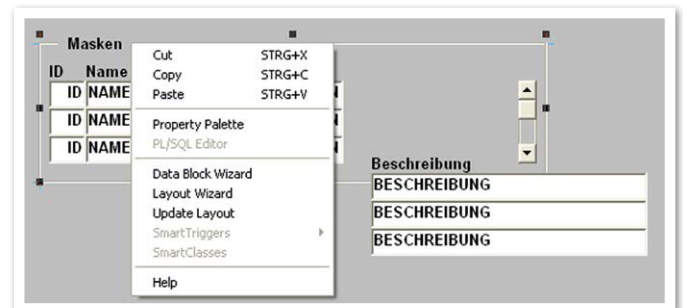


Abbildung 3: Die Breite vergrößern

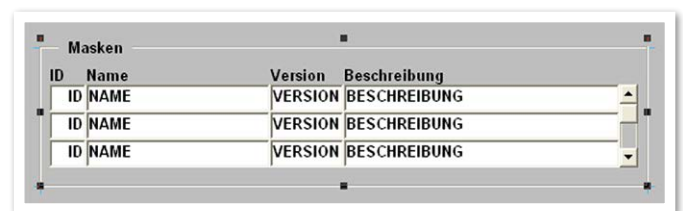


Abbildung 4: Die optimale Breite

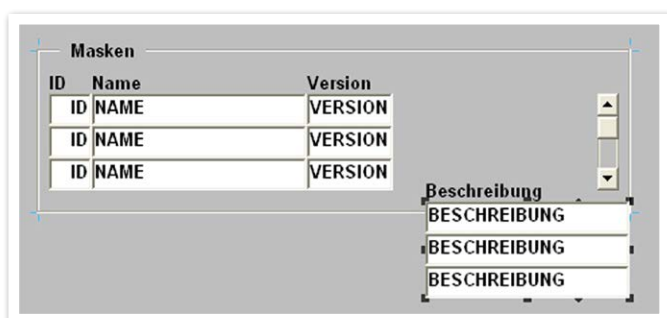


Abbildung 2: Ausrichtung nach rechts und nach unten



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

DOAG unterstützt offenen Brief der Campaign for Clear Licensing



**CAMPAIGN
FOR CLEAR
LICENSING**

Die britische Non-Profit-Initiative „Campaign for Clear Licensing“ (CCL, „<http://www.clearlicensing.org>“) hat in einem offenen Brief an den Oracle-CTO Larry Ellison in einem „Sieben-Punkte-Plan“ dargestellt, wie Oracle zukünftig den Umgang mit seinen Kunden verbessern könnte (siehe „<http://www.clearlicensing.org/open-letter>“). „Generell unterstützen wir diese sieben Punkte“, so Dr. Dietmar Neugebauer, Vorstandsvorsitzender der DOAG. „Insbesondere in den Bereichen Kommunikation und Information gibt es bei Oracle noch Verbesserungspotenzial.“

Ihre Forderungen stützt die „Campaign for Clear Licensing“, die sich für klare Lizenzbestimmungen für die Kunden von Business-Software einsetzt, auf ihre im letzten November veröffentlichte Studie. Rund 100 IT-Manager hatte die CCL zur Oracle-Lizenzpolitik befragt – um anschließend zu einem ernüchternden Ergebnis zu kommen. 92 Prozent der Befragten berichteten von erheblichen Mängeln in der Kommunikation. So sollen in einigen Fällen nach Änderung der Lizenzbestimmungen sogar veraltete Informationen kommuniziert wor-

den sein. Weiterhin bewerteten 88 Prozent der Befragten die Anforderungen in Oracle-Audits als unklar. Lediglich 22 Prozent stufen den Oracle License Management Service (LMS) bei Vertragserneuerungen und -verhandlungen als behilflich ein.

Mit dem Ergebnis der Studie greift die CCL Anliegen auf, die Oracle-Kunden schon seit längerer Zeit beschäftigen. Die DOAG unterstützt daher generell die von der CCL vorgestellten sieben Punkte:

1. Strategischer Fokus auf Kundenzufriedenheit
2. Klarheit bei Audits
3. Informationen aus einer Hand
4. Wissensdatenbank
5. Geringeres Risiko bei Reengineering
6. Motivation und Prozess-Blueprints für Software Asset Management
7. Kommunikation

„Besonders wichtig für uns sind dabei die Punkte Wissensdatenbank und Kommunikation“, so Dr. Dietmar Neugebauer. Kunden machten ihn immer wieder auf die mangelnden Informationen aufmerksam.

Gleichzeitig appelliert Dr. Neugebauer auch an die Kunden, sich rechtzeitig auf die Audits vorzubereiten und sich mit ihren Lizenzen und den Oracle Lizenzregeln auseinanderzusetzen. Den eigentlichen Handlungsauftrag sehe er jedoch beim Hersteller: „Oracle müsste mehr informieren, die Komplexität abbauen und die Kommunikation mit den Kunden verbessern“, schlussfolgert er.

Michael Paege, stellv. DOAG-Vorstandsvorsitzender und Leiter Competence Center Lizenzierung, ergänzt: „Insgesamt wünschen wir uns hier mehr Engagement von Oracle und fordern mehr Kundenorientierung, ein stärkeres Eingehen auf die Wünsche und Probleme der Kunden und ein Anpassen der Oracle-Lizenzpolitik an Marktgegebenheiten wie beispielsweise in Hinblick auf VMware.“

Die DOAG hat hier schon frühzeitig den Handlungsbedarf erkannt und mit dem digitalen Lizenzguide ein Nachschlagewerk herausgebracht, in dem sich DOAG-Mitglieder über die wichtigsten Aspekte der Lizenzierung von technischer Oracle-Software informieren können.

Wir begrüßen unsere neuen Mitglieder

Persönliche Mitglieder

Manfred Warzawa
Kerstin Köpfer
Christopher Krah
Philipp Riemer
Sebastian Schweinle
Stefanie Bartl
Richard Haase
Dieter Käshammer
Werner Sanojca
Young-Hoon Kim
Silvan Tatovsky
Christoph Krampitz
Michael Vollrath
Sebastian Martinka

Stephan Zornow
Jörg Thomas
Salim-Martin Yermani
Dennis Stritzke
Kerstin Glogert
Jan Berens
Tobias Anhalt
Dominic Lindner
Nils Wild
Sergej Reiser
Erida Shkurti
Uwe Schmidt
Martin Weber

Firmenmitglieder DOAG

Mert Kartal, SoftwareONE Deutschland GmbH
Sascha Rudolf, GFI Gesellschaft für Informations- und Datenverarbeitung mbH
Robert Zweers, CAPCON Deutschland GmbH
Jobst Eßmeyer, cronn GmbH
Wolfgang Stoll, INNEO Solutions GmbH

Firmenmitglieder SOUG

Sascha Körner, Rudolph AG
Rolf Maag, Banque Pictet & Cie SA
Ricardo Santandrea, Banque Pictet & Cie SA
David Schoch, Six Group Services AG
Shahid Abdul, Six Group Services AG



24.04.2015

DOAG Webinar: Fusion Reporting Cloud für die E-Business Suite

Dirk Blaurock
bsc-eps@doag.org

27.04.2015

Regionaltreffen München/Südbayern

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

28.04.2015

DOAG Exaday | Frankfurt

Stefan Panek
stefan.panek@doag.org

29.04. – 30.04.2015

DOAG DevCamp | Frankfurt

Upcycling Software – Modernisierung in der Softwareentwicklung
Robert Szilinski
robert.szilinski@doag.org



05.05. – 06.05.2015

Berliner Expertenseminar mit Klaus Reimers zum Thema "Dataguard" | Berlin

Cornel Albert
expertenseminare@doag.org

07.05.2015

DOAG Oracle Spatial & Geodata Day | Frankfurt

Kushtrim Krasniqi
sig-spatial@doag.org

06.05.2015

SOUG bi de Lüt

Die Mobiliar, Bern
sekretariat@soug.ch

08.05.2015

DOAG Webinar: RAC-Test: Welche sind notwendig? Welche sind überhaupt durchführbar?

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

19.05.2015

DOAG Security Day | München

Franz Hüll, Tilo Metzger
sig-security@doag.org

20.05.2015

Regionaltreffen München/Südbayern

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

21.05.2015

Regionaltreffen NRW (Forms Community)

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

21.05.2015

Regionaltreffen Stuttgart

Jens-Uwe Petersen
regio-stuttgart@doag.org

21.05.2015

Regionaltreffen Nürnberg/Franken

André Sept, Martin Klier
regio-franken@doag.org

21.05.2015

SOUG SIG Tuning | Prangins (Schweiz)

sekretariat@soug.ch

28.05.2015

DOAG Middleware Day |

Jan-Peter Timmermann
sig-middleware@doag.org



03.06.2015

Regionaltreffen Berlin/Brandenburg

Michel Keemers
regio-bb@doag.org

08.06.2015

Regionaltreffen NRW (Vorabend APEX.connect 2015)

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

09.06. – 10.06.2015

APEX.connect 2015 | Düsseldorf

Niels de Bruijn
niels.de.bruijn@doag.org

Impressum

Herausgeber:

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

SOUG Swiss Oracle User Group

Im Gundelinger Feld/Bau 5
Dornacherstrasse 192
CH-4053 Basel
Tel.: +41 (0)61 367 93 30
www.soug.ch

Verlag:

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

Chefredakteur (ViSdP):

Wolfgang Taschner, redaktion@doag.org

Redaktion:

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

Titel, Gestaltung und Satz:

Alexander Kermas, DOAG Dienstleistungen GmbH

Titelfoto: © alphaspirt / 123RF.com

Foto S. 7: © Chornii Yevhenii / 123RF.com

Foto S. 34: © Galina Peshkova / 123RF.com

Foto S. 72: © antonprado / 123RF.com

Anzeigen:

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

Druck:

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

Inserentenverzeichnis

avato consulting ag www.avato-consulting.com	S. 49
DBConcepts www.dbconcepts.at	S. 41
dbi services ag www.dbi-services.com	S. 27
DOAG e.V. www.doag.org	S. 51, 59, 71
Libelle AG www.libelle.com	S. 15
MT AG www.mt-ag.com	U 2
MuniQsoft GmbH www.muniqsoft.de	S. 3
ORACLE Deutschland B.V. & Co. KG www.oracle.com	U 3
SYMAX Business Software AG www.symax.de	S. 33
Trivadis GmbH www.trivadis.com	U 4

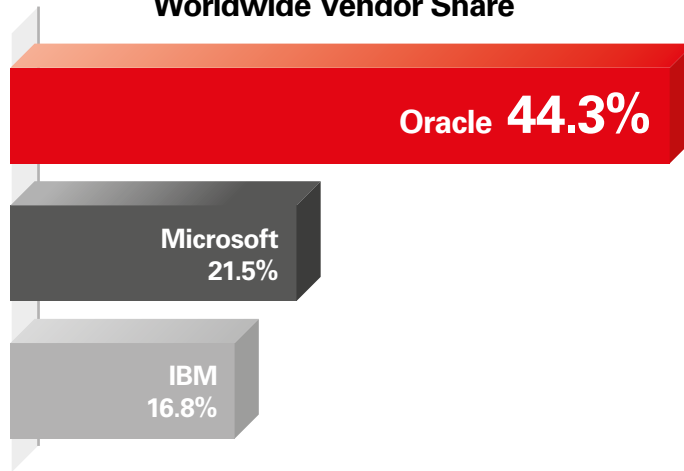
Weitere Termine und Informationen unter www.doag.org/termine/calendar.php
sowie unter www.soug.ch

STILL

#1

Database

Worldwide Vendor Share



Oracle Database

Trusted by 310,000 Customers Worldwide

ORACLE®

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

Source: IDC, "Worldwide Database Management Systems 2014–2018 Forecast and 2013 Vendor Shares," IDC #248952; June 2014, Table 4, (Relational Database Management Systems). Vendor share based on software license and maintenance revenue.

Copyright © 2015, Oracle and/or its affiliates. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Gut zu wissen, dass es in der Firma läuft.



■ Gestalten Sie Ihr Leben sorgenfreier. Und Ihre IT leistungsfähiger. Denn wir haben das richtige Servicemodell für Sie. Von der Pflege und dem Support für Ihre Software und BI-Lösungen über den hochverfügbaren Betrieb Ihrer IT-Infrastruktur bis hin zum Outsourcing oder Cloud-Services. Immer effizient und innovativ. Trivadis ist führend bei der IT-Beratung, der Systemintegration, dem Solution-Engineering und bei den IT-Services mit Fokussierung auf Oracle- und Microsoft-Technologien im D-A-CH-Raum. Sprechen Sie mit uns. www.trivadis.com | info@trivadis.com

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

trivadis
makes IT easier. ■ ■ ■