

Red Stack

Magazin

Mobile/ Internet der Dinge

Moderne Technologien



Mobil werden

- Forms-Anwendung
- PL/SQL-Logik

Im Interview

Stefan Kinnen, neu gewählter Vorstandsvorsitzender der DOAG



Datenbank

Cloning-Lösungen im Überblick

Oracle Cloud Day

Innovation
to grow.
Insight to
control.



15. November 2016
in Nürnberg

Jetzt anmelden:

oracle.com/goto/de/Cloud-Day



Der Oracle Cloud Day findet parallel zur
DOAG KONFERENZ + AUSSTELLUNG
(15. bis 18. November 2016) statt.



Robert Szilinski
DOAG-Vorstand und Leiter
der Development
Community

Liebe Mitglieder, liebe Leserinnen und Leser,

in den Städten ist derzeit ein interessantes Phänomen zu beobachten: Jugendliche und Erwachsene streichen durch die Straßen und jagen kleine Monster, die sich an Hausecken oder Sehenswürdigkeiten verstecken. Pokémon Go ist da! Auch wenn man es inhaltlich nicht gut finden muss, was da passiert, Fakt ist: Endlich sorgt mal wieder eine mobile App für Furore.

Spannende Apps gibt es immer mal wieder, aber gefühlt ist das Trendthema „Mobile“ nicht mehr so omnipräsent wie noch vor wenigen Jahren. Dies gilt weniger für den Consumer-Bereich als vielmehr für die Unternehmen. „Jede Firma braucht eine Mobile-Strategie“, heißt es. Entwickelt wird natürlich „Mobile first“, bestenfalls braucht es gar keine anderen Anwendungen mehr – die Zukunft ist mobil, in Apps gegossen und optimal vernetzt.

Auch wenn es ganz so extrem bisher nicht gekommen ist: Der Mobile-Trend hat definitiv zum Umdenken geführt. Mehr Fokus auf Innovation, Einfachheit und die tatsächlichen Bedürfnisse der Nutzer haben etablierte Prozesse infrage gestellt und neue Dienstleistungen ermöglicht. „Mobile“ hat inzwischen seinen festen Platz und sich vielleicht tiefer in unserem Alltag verankert, als wir es wahrhaben wollen.

Trotzdem scheint es so, als ob die große Zeit der Apps langsam zu Ende geht. Moderne JavaScript-Frameworks bieten hervorragende Möglichkeiten, um mobile Anwendungen zu entwickeln. Apps sind nicht mehr reine IT-Sache, Business-Anwender können sich ihre Apps zukünftig selbst zusammenklicken. Beide Trends greift Oracle mit neuen Cloud-Services-Angeboten und Oracle JET auf. Es wird spannend sein zu sehen, ob es diesmal auch gelingt, einen signifikanten Footprint zu hinterlassen.

Herzlichst,
Ihr

R. Szilinski

MUNIQSOFT

Support

Mit **IQ** vermeiden Sie Notfälle
und erhalten Hilfe, wenn's brennt.

+49 (0) 89 6228 6789-0

www.munisoft.de/support

Unser Kundenservice

- ▶ Deutschsprachiger Telefonsupport
- ▶ Garantierte Reaktionszeiten
- ▶ 3rd- und 2nd-Level Support
- ▶ Regelmäßige Datenbank-Healthchecks
- ▶ Proaktives Datenbank-Monitoring und Patch-Management
- ▶ Incident- und Problemmanagement



Unternehmensanwendungen in die eine oder andere Richtung überdenken



Die zweite Generation der Event-Processing-Tools betritt die Bühne



Das Internet of Things stellt vermehrt frühere Geschäftsmodelle infrage

Einleitung

- 3 Editorial
- 5 Timeline
- 8 „Die Struktur des Vereins überdenken und modernisieren ...“
Interview mit Stefan Kinnen

Mobile

- 10 Mobile Security
Michael Fischer
- 16 Oracle Forms goes Mobile
Jan-Peter Timmermann
- 22 PL/SQL-Logik Beine machen: mobil durch Webservices
Daniel Kampf

Internet der Dinge

- 26 Von Big Data bis IDS – Informationslogistik im Internet der Dinge
Jens Leveling und Christian Prasse
- 31 Industrie 4.0: Analytisch, Prozess-orientiert und transformativ
Marcel Amende und Thorsten Schulz

- 35 Event-Processing und Stream-Analyse
Arne Brüning
- 40 IoT-Architektur – sind traditionelle Architekturen gut genug oder braucht es neue Ansätze?
Guido Schmutz
- 48 Die ersten Schritte in die IoT-Welt
Philipp Wang
- 54 Oracle und Industrie 4.0 – zurück in die Zukunft
Gerhild Aselmeyer

Datenbank

- 60 Cloning-Lösungen im Überblick
Miguel Anjo, Ludovico Caldara, Catalin Motaitianu und Dawid Wojcik
- 63 E-Mail vom DB-Server: „Brauche neues Power Supply!“
Elke Freymann
- 68 Die Effizienz von SmartScan für die Datenbank einschätzen
Franck Pachot
- 73 Tipps und Tricks aus Gerds Fundgrube Heute: PLL vs. PLX
Gerd Volberg

Intern

- 74 Neue Mitglieder
- 74 Impressum
- 74 Inserenten

Timeline

20. Mai 2016

Christian Trieb, Leiter der Datenbank Community, ist zu Besuch in der DOAG-Geschäftsstelle in Berlin. Ziel ist es, die optimale Unterstützung der Community abzustimmen, um die Zusammenarbeit zu optimieren.

26./27. Mai 2016

Das Team der DOAG Dienstleistungen GmbH trifft sich zum jährlichen Workshop. Hauptthema ist diesmal ein Kommunikationstraining, um die Mitarbeiterinnen und Mitarbeiter für die Zusammenarbeit mit dem Verein und für eine erfolgreiche Steuerung der Projekte fit zu machen.

31. Mai 2016

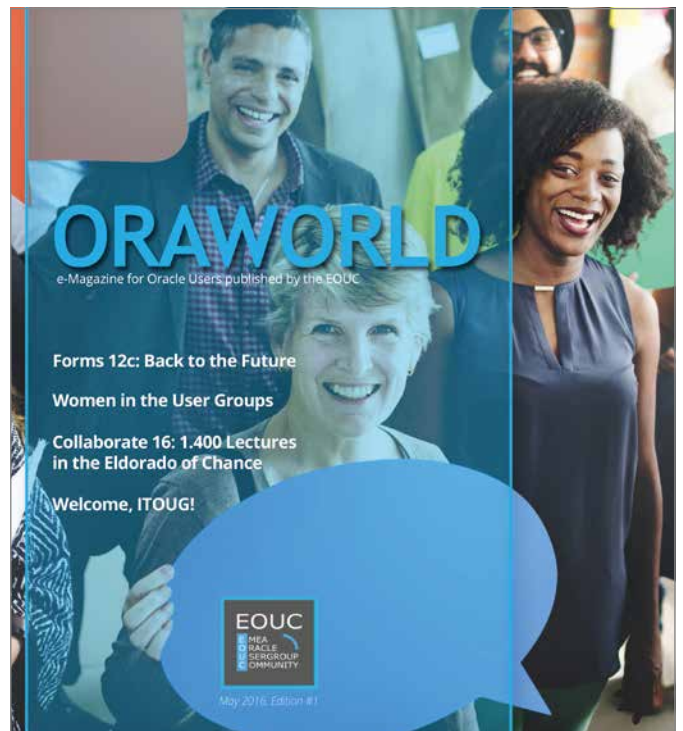
Stefan Kinnen, der neu gewählte Vorstandsvorsitzende der DOAG, stellt sich in der Geschäftsstelle in Berlin vor. Alle sind gespannt auf seine persönlichen Ziele, um ihn bei der Umsetzung bestmöglich zu unterstützen.



Der neue DOAG-Vorstandsvorsitzende Stefan Kinnen

7./8. Juni 2016

Dr. Dietmar Neugebauer, ehemaliger Vorstandsvorsitzender der DOAG, und Ralf Kölling, Repräsentant der Regionalgruppe Bremen, vertreten die DOAG auf dem von Oracle veranstalteten Summit der European, Middle-East and African Oracle User Group Leaders Community (EOUC) in Bukarest. Fried Saacke, DOAG-Vorstand und Geschäftsführer, komplettiert die DOAG-Delegation als Vertreter des Interessenverbands der Java User Groups e.V. (iJUG). Die wichtigsten Themen der DOAG sind die Präsentation und Vermarktung des neuen E-Magazine ORAWORLD sowie die Einladung der Vertreter der Usergroups am Vorabend der DOAG 2016 Konferenz + Ausstellung nach Nürnberg.



Das E-Magazine ORAWORLD

10. Juni 2016

Dr. Clemens Appl von der Wirtschaftsuniversität Wien präsentiert im Rahmen des AOUG Regional-Forums in Graz die Ergebnisse der Studie zur Rechtssituation der Oracle-Lizenzierung in VMware-Umgebungen.

16./17. Juni 2016

Die erste Vorstandssitzung unter der Leitung des neu gewählten Vorstandsvorsitzenden Stefan Kinnen findet in Berlin statt. Stefan Kinnen und DOAG-Geschäftsführer Fried Saacke heißen die

beiden neuen Vorstände Svenja Schriever-Festerling und André Sept herzlich willkommen. Im Mittelpunkt der Sitzung steht die Ausarbeitung der auf der Delegiertenversammlung beschlossenen Ziele. In deren Rahmen werden die Gründung eines Anwenderbeirats sowie die Gründung der neuen Next Generation Community vorbereitet.

17. Juni 2016

Vertreter des Interessenverbands der Java User Groups e.V. (jUG) kommen nach Berlin in die DOAG-Geschäftsstelle, um die Einführung des Planungs- und Kommunikationssystems Redmine vorzubereiten. Am Abend findet gemeinsam mit dem DukeCon-Projektteam eine Bootsfahrt auf der Spree statt.



Auf der Spree unterwegs

18. Juni 2016

Das DukeCon-Projektteam ist zu einem Arbeitstreffen in der DOAG-Geschäftsstelle in Berlin. Ziel ist es, den als Open-Source-Projekt für die JavaLand entwickelten Konferenzplaner auch für die DOAG zu etablieren. Im Gegenzug übernimmt die DOAG den Betrieb der Plattform.

20. Juni 2016

Das Organisationsteam der JavaLand 2017 trifft sich zusammen mit der Projekt-Managerin Kirsten Böttcher zum Kickoff-Meeting auf dem Münchener Flughafen. Besondere Herausforderung ist es, für die schnell wachsende Konferenz ausreichend Räumlichkeiten im Phantasialand bereitzustellen.



Die Community auf der JavaLand 2016

21. Juni 2016

Auf der AOUG-Anwenderkonferenz in Wien präsentiert Penny Avril, Vice President Server Technology bei Oracle, in ihrer Keynote die Vision des Next-Generation-Data-Managements. Anschließend gibt Bryn Llewellyn, Product Manager für PL/SQL und EBR bei Oracle, zahlreiche Argumente dafür, warum man PL/SQL nutzen soll. Rund 130 Teilnehmer informieren sich einen Tag lang im Wiener Austria Trend Hotel Savoyen über aktuelle Themen und Trends.

22. Juni 2016

Bryn Llewellyn hält in Wien am Tag nach der AOUG-Anwenderkonferenz ein exklusives Ganztages-Seminar zu den Themen „Transforming one table to another: SQL or PL/SQL?“ und „Edition based redefinition (ERB)“.

23. Juni 2016

Im Segelhof in Baden findet der zweite SOUG Day 2016 statt. Die Keynote von Johannes Ahrends dreht sich um die Standard Edition 2. Wie nach jeder Präsentation gibt es Zustimmung dafür, dass Zeit, die in dieses Thema investiert wird, sich in bare Münze auszahlen kann. Anschließend folgen drei Themen-Streams, die jeweils gut besucht sind. Neben Oracle-Next-Generation und Cloud gibt es einen interessanten MySQL-Stream. Der von Dell Software und dbi services gesponserte Apéro ermöglicht die Vertiefung der Themen bis in den Abend hinein.



Auf dem SOUG Day 2016 gibt es viel zu diskutieren

27. Juni 2016

Dr. Dietmar Neugebauer, ehemaliger Vorstandsvorsitzender der DOAG, Ralf Kölling, Repräsentant der Regionalgruppe Bremen, und Fried Saacke, DOAG-Vorstand und Geschäftsführer, telefonieren mit Tom Scheirsens, EMEA User Groups Relationship Manager von Oracle. Sie informieren ihn über die anstehenden Projekte der DOAG und laden ihn offiziell nach Nürnberg zur DOAG 2016 Konferenz + Ausstellung und zum am Vorabend stattfindenden Treffen der User Group Leader ein.



Tom Scheirsens, EMEA User Groups Relationship Manager von Oracle

28. Juni 2016

DOAG-Vorstand und Geschäftsführer Fried Saacke ist beim Notar, um die JavaLand GmbH zu gründen. Die neue, 100-prozentige Tochter der DOAG Dienstleistungen GmbH soll die mit einer solchen großen Veranstaltung verbundenen Risiken einschränken und somit die finanzielle Situation der DOAG weiterhin stabil halten sowie mit einem an den Veranstaltungszeitpunkt angepassten Wirtschaftsjahr vom 1. Juni bis zum 31. Mai die betriebswirtschaftliche Abbildung optimieren.

26. Juni bis 1. Juli 2016

Christian Trieb und Niels de Bruijn nehmen als Vertreter der DOAG an der KSCOPE 16 in Chicago teil, veranstaltet von der amerikanischen Oracle Development Tools User Group (ODTUG). In einem Gespräch mit dem ODTUG-Vorstand geht es um die Zusammenarbeit mit der DOAG. Dabei wird auch eine Einladung zum Treffen der europäischen User Group Leader auf der DOAG 2016 Konferenz + Ausstellung in Nürnberg ausgesprochen. Besonders angetan ist Christian Trieb von der neuen App, die dort zum Einsatz kommt. Damit können die Teilnehmer beim Besuch der Ausstellung Punkte sammeln, Vorträge bewerten, Social-Media-Aktivitäten durchführen und am Ende einen Preis gewinnen.



Chicago – Standort der KSCOPE

6. Juli 2016

Am Vortag des Java Forums Stuttgart findet die Mitgliederversammlung des Interessenverbands der Java User Groups e.V. (ijUG) statt. Inzwischen sind 30 User Groups aus Deutschland, Österreich und der Schweiz im ijUG organisiert. Die Mitgliederversammlung in Stuttgart bespricht die Meilensteine zur JavaLand 2017. Darüber hinaus wird sehr viel Wert auf eine gemeinsame Öffentlichkeitsarbeit gelegt, um die Interessen der Java-Entwickler optimal vertreten zu können. Insbesondere um den Fortbestand von Java FX und Java EE sind Aktivitäten geplant.

8. Juli 2016

Der DOAG-Vorstand beschließt auf einer Sitzung in München mit Vertretern der Swiss Oracle User Group (SOUG), der Austrian Oracle User Group (AOUG) und von Oracle sowie mit den Stream-Leitern das Programm zur DOAG 2016 Konferenz + Ausstellung. Die ausgewählten Vorträge versprechen wieder eine interessante und spannende Veranstaltung.

„Die Struktur des Vereins überdenken und modernisieren ...“

Stefan Kinnen ist seit Ende April 2016 neuer Vorstandsvorsitzender der DOAG. Wolfgang Taschner, Chefredakteur des Red Stack Magazin, sprach mit ihm über seine Ziele.

Wann hast du zum ersten Mal von der DOAG erfahren?

Kinnen: Nach einigen Jahren in der Automatisierungstechnik habe ich im Jahr 1999 zu einem Hersteller von Labor-Informations- und Management-Systemen gewechselt, der auch im Oracle-Consulting aktiv war. Dort haben viele Kollegen von der DOAG geschwärmt und mir zur Einarbeitung Magazine und dicke Kataloge der früheren DOAG-Jahreskonferenzen in die Hand gedrückt.

Wie war dein Eindruck von deiner ersten DOAG-Veranstaltung?

Kinnen: Im Jahr 1999 war ich zum ersten Mal bei der Mitgliederversammlung und der Jahreskonferenz in Fellbach. Als Newcomer in Sachen Oracle war alles damals sehr beeindruckend. Thematisch prägend für mich war das Motto „The Internet changes everything“. Vieles war völlig neu – manche Diskussionen und Themen wie die Zukunft von Forms oder die Zufriedenheit mit Lizenzbedingungen und dem Support begleiten mich aber bis heute.

Wie bist du DOAG-Aktiver geworden und was war deine erste Funktion?

Kinnen: Irgendwie bin ich mit der damaligen Vorstandsvorsitzenden Agnes Hombrecher ins Gespräch gekommen, die mich zur Mitarbeit in der Regionalgruppe NRW motiviert hat, die damals etwas uneinig darüber war, ob eine Teilung in Nord und Süd Sinn macht. Hildgund von Alten-Krämer und Dierk Lenz haben mich dabei sehr unterstützt und in diesem Dreier-Team haben wir in ganz NRW viele tolle Treffen durchgeführt.

Die Regionalgruppe NRW zählt zu den meistbesuchten in Deutschland. Was steckt hinter eurem erfolgreichen Konzept?

Kinnen: NRW zählt erstmal zu den Ballungszentren in Deutschland. Eine Fokussierung auf eine einzelne Stadt hat es aber nie gegeben; somit war das Einzugsgebiet immer groß. Hilfreich war auch immer die Bereitschaft von verschiedensten Anwendern und Partnern, uns als Gruppe einzuladen. Die häufig wechselnden



Stefan Kinnen blickt positiv in die Zukunft

den Gastgeber und ein hohes Maß an Networking und interaktiver Diskussion haben die Treffen meist sehr lebendig gemacht.

Im Jahr 2005 wurdest du in den DOAG-Vorstand gewählt. Was waren damals deine Schwerpunkte?

Kinnen: Im Gegensatz zu heute gab es noch keine feste Zuordnung zwischen der Mitgliedschaft im Vorstand und einer speziellen Rolle. Meine Hauptaufgaben waren lange Zeit die Finanzen und auch die Mitwirkung bei der Schaffung der neuen Struktur mit GmbHs und Communities.

Die DOAG hat während deiner Zeit als Vorstand etliche einschneidende Neuerungen durchgeführt. Mit welchen Veränderungen fühlst du dich am meisten verbunden?

Kinnen: Die Einführung der Communities war sicherlich eine der wichtigsten Veränderungen. Auch wenn es immer wieder Überschneidungen und unscharfe Abgrenzungen gibt, war und ist es wichtig, dass wir wirklich alle Themenbereiche des Oracle-Portfolios abdecken – was wahrlich kein leichtes Ziel ist.

Nach Gründung der Development Community im Jahr 2011 warst du deren erster Leiter. Wie hat sich die Community damals entwickelt?

Kinnen: Das hat wirklich richtig Spaß gemacht. Es hat sich schnell ein lebendiges Team gebildet, mit dem wir erfolgreiche Fachkonferenzen durchführen konnten und ein bis heute noch lebendiges Networking in der Community angestoßen haben.

Im April 2016 bist du zum Vorstandsvorsitzenden der DOAG gewählt worden. Was sind deine Ziele, wohin willst du die DOAG führen?

Kinnen: Nun, ich sehe für meine neue Rolle fünf Punkte, die ich erreichen will: Stärkung der Communities, Förderung des Nachwuchses, stärkeren Einfluss der Anwender, Modernisierung des Vereins und verstärkte internationale Zusammenarbeit der Anwendergruppen.

Wie sollen sich die Communities weiterentwickeln?

Kinnen: Die Communities und auch die Querschnittsgruppen sollen noch eigenständiger im Rahmen ihres Handlungsspielraums arbeiten und agieren. Die Delegierten haben kompetente Vorstände für die jeweiligen Rollen gewählt, die mit ihrem jeweiligen Team kreativ und interaktiv arbeiten können und sollen.

Wie schaut es mit dem DOAG-Nachwuchs aus?

Kinnen: Die Förderung des Nachwuchses steht zwar schon länger auf dem Plan – wurde aber noch nicht optimal umgesetzt. Der Aufbau einer Nachwuchs-Community „Next Generation“ soll das ändern. Dadurch kann die DOAG Impulse und Ideen bekommen, um auch künftig die richtigen Zielgruppen zu erreichen.

Das „A“ in DOAG steht für Anwender. Wie stark repräsentieren diese den Verein?

Kinnen: Viele Impulse und vor allem auch die aktive Mitarbeit kommen zunehmend von Oracle-Partnern. Um dennoch auf fundiertes Praxiswissen und Erfahrungen als Anwender (und Kunde) von Oracle zurückgreifen zu können, möchte ich den Aufbau eines reinen Anwenderkreises als Beirat oder Leadership-Board vorantreiben.

In welche Richtung soll sich die DOAG weiterentwickeln?

Kinnen: Wir finden uns immer wieder in Diskussionen, die eine Öffnung unserer Tätigkeiten und unseres Wissens fordern. Gleichzeitig ist es aber wichtig, dass wir den Verein stärken und unabhängig von Oracle oder anderen Förderern arbeiten können. Von daher stelle ich mich auch der Herausforderung, die Struktur des Vereins zu überdenken und zu modernisieren.

Was sind deine internationalen Ziele mit der DOAG?

Kinnen: Neben dem eigenen Engagement der DOAG werden auch die internationalen Kooperationen immer wichtiger. Austausch und Zusammenarbeit mit anderen europäischen Usergroups sind wichtige Ziele für mich. Diese Zeitschrift beispielsweise ist ja schon eine Gemeinschaftsausgabe mit der Swiss Oracle User Group und der Austrian Oracle User Group. Vor Kurzem ist der neue OraWorld-Newsletter der EOUC gestartet, den ich sehr begrüße. Überhaupt wird die Zusammenarbeit der regionalen Usergroups im EOUC immer aktiver, was ich sehr gern fördern möchte.

Was erwartest du dir zukünftig von Oracle?

Kinnen: Zunächst einmal möchte ich meine Freude darüber ausdrücken, wie einladend Oracle auf mich zugegangen ist. In kurzer Zeit habe ich auf nationaler und internationaler Management-Ebene mehrere Gespräche führen und Kontakte knüpfen können, die einheitlich die Zusammenarbeit mit der DOAG begrüßt haben. Ich wünsche mir, dass diese Kanäle offen bleiben

und wir so einen regen Austausch von Erwartungen und Erfahrungen mit Oracle teilen können.

Du bist ja hauptberuflich als Vice President Business Development beim Oracle Platin Partner Apps Associates. Wie bekommst du das alles unter einen Hut?

Kinnen: Die Frage kommt von meiner Frau (lacht) ... Im Ernst: Das ist einer der Punkte, über die ich auch lange nachgedacht habe. Mir ist bewusst, dass ein Vorsitz nochmal deutlich mehr Zeitaufwand bedeutet. Erfreulicherweise hat mein Arbeitgeber dem Vorhaben sofort zugestimmt und mir den notwendigen Freiraum eingeräumt. Für die Regionalgruppe NRW habe ich mit Martin Schmitter von RWE Supply & Trading einen kompetenten und engagierten Nachfolger gefunden.

Wie zufrieden bist du als Oracle Platin Partner mit den Oracle-Produkten. Wohin sollte sich Oracle weiterentwickeln?

Kinnen: Nachdem ich viele Jahre mit den eher klassischen Datenbank- und Development-Produkten beschäftigt war, habe ich heute auch viel mit Standard-Software wie BI-Lösungen oder der E-Business Suite zu tun. Ich sage es mal so: Auch die DOAG würde nicht kontinuierlich wachsen, wenn Partner und Anwender nicht überzeugt von den Produkten wären. Momentan steht der Wechsel in die Cloud im Fokus: Oracle ist auf diesen Zug nicht sofort, dafür aber dann absolut konsequent aufgesprungen. In Europa sehen wir heute schon recht unterschiedliche Reaktionen darauf – in den USA erst recht. Oracle wird diesen Weg sicher weitergehen, sollte aber die Kunden im Tempo nicht überfordern und ein Gleichgewicht zwischen „as a Service“ und Lizenzprodukten schaffen.

Bleibt dir bei so viel beruflichem Engagement noch Zeit für private Aktivitäten?

Kinnen: Zum Glück, ja! Den Ausgleich zur Arbeit und auch zur DOAG schaffe ich doch noch regelmäßig.

Zur Person: Stefan Kinnen

Stefan Kinnen ist seit Mai 2016 Vorstandsvorsitzender der DOAG Deutsche Oracle-Anwendergruppe e.V. Zuvor war er stellvertretender Vorsitzender und Vorstand Finanzen und ist seit dem Jahr 2005 – mit kurzer Unterbrechung – Vorstandsmitglied, unter anderem als Leiter der Development Community. Außerdem war er Leiter der Regionalgruppe Nordrhein Westfalen.

Als Vice President Business Development beim Oracle-Platin-Partner Apps Associates schafft er mit seinem Team Lösungen in den Bereichen Business Software, Custom Development, Business Intelligence und Managed Services. In seiner Freizeit ist Stefan Kinnen gerne sportlich aktiv – zu seinen liebsten Sportarten gehören Golf, Wandern und Radeln. Außerdem interessiert er sich sehr für Musik.

SECURITY

Mobile Security

Michael Fischer, ORACLE Deutschland B.V. & Co. KG

Mit der zunehmenden Nutzung mobiler Endgeräte sowohl im privaten als auch im beruflichen Umfeld und den täglichen Berichten über Daten-Missbrauch und neue Regularien wie das EU-Datenschutzgesetz steigt das Interesse an mobiler Security. Dieser Artikel zeigt eine End-to-End-Betrachtung von Konzepten und Lösungen für eine Absicherung bei der Verwendung von mobilen Devices wie Smartphones oder Tablets.

Wenn man nach „Mobile Security“ im Internet sucht, erscheinen in den Trefferlisten hauptsächlich Lösungen für Anti-Virus- und Anti-Malware-Lösungen. Diese betrachten Mobile Security aus Sicht des Anwenders, der sich wenig darum kümmert, wie die Absicherung außerhalb seines Device aussieht. In etwa so, wie er es vom Desktop her kennt, und vielleicht ist er auch deshalb froh, dass er nicht mehr so viele Wahlmöglichkeiten hat, etwa automatisiertes Patchen zu aktivieren, Passwörter in einem Password-Safe abzuspeichern oder über die Desktop-Firewall selektiv Dienste abzuriegeln. Unternehmen stecken bei der Endgeräte-Absicherung in einem Zwiespalt, einerseits nur möglichst sichere und kontrollierte Endgeräte zuzulassen, andererseits ein attraktives Arbeiten auch mit eigenen Devices zu ermöglichen. Das Paradigma „work from anywhere – with anything“ tut sein Übriges dazu.

Die Absicherung der Endgeräte, ob Smartphone, Tablet oder Desktop, be-

trachtet nur einen Teil der potenziellen Einfallstore. Studien von Analysten und Beratern beschäftigen sich bei Mobile Security oft lediglich damit, die Apps aus zugänglichen App Stores zu analysieren und zu bewerten. Dies deckt Probleme auf, wie eine implizite Weitergabe von Daten, betrachtet aber wieder nur einige Aspekte. Eine Lösung bedarf nicht nur Korrekturen im App-Design, sondern benötigt ein ganzheitliches Security-Konzept.

Der Aufbau

Mithilfe einer logischen Architektur werden die einzelnen Schutzmechanismen erläutert. Parallel dazu erfolgt eine Betrachtung aus der Sicht der Akteure – hier der Endbenutzer, Entwickler, Betreiber sowie Unternehmen. Über diese Zerlegung kann ein Mapping auf einzelne Produkte verschiedener Hersteller erfolgen oder eine bestehende Architektur lässt sich auf Lücken prüfen.

Exemplarisch sind das Oracle-Portfolio gemappt und ein Kundenbeispiel aufgeführt. Werden bei einer Architektur zwischen diesen Schichten standardisierte Schnittstellen und Protokolle berücksichtigt, lassen sich fehlende Funktionen nachrüsten. Ein Vergleich aus Anbietersicht mit Komplettlösungen findet nicht statt, jedoch könnte eine Betrachtung über die vorhandenen Protokolle erfolgen. Komplettlösungen ergeben durchaus Sinn, wenn es um vollständig kontrollierbare und abgeschottete Einsatzszenarien geht. Wermutstropfen dabei ist die Nutzbarkeit; obwohl etwa das Phone der Kanzlerin eine solche abgesicherte Lösung bietet, erfolgt parallel die Nutzung als normales Smartphone.

Nicht alle Nutzungsszenarien sind gleich, daher sind die skizzierten Funktionen und Komponenten eine nicht vollständige Obermenge. Der Einstieg mit verschiedenen Layern und Funktionsbausteinen ist jedoch schon relativ früh sinnvoll, so kann eine Anmelde-möglichkeit an

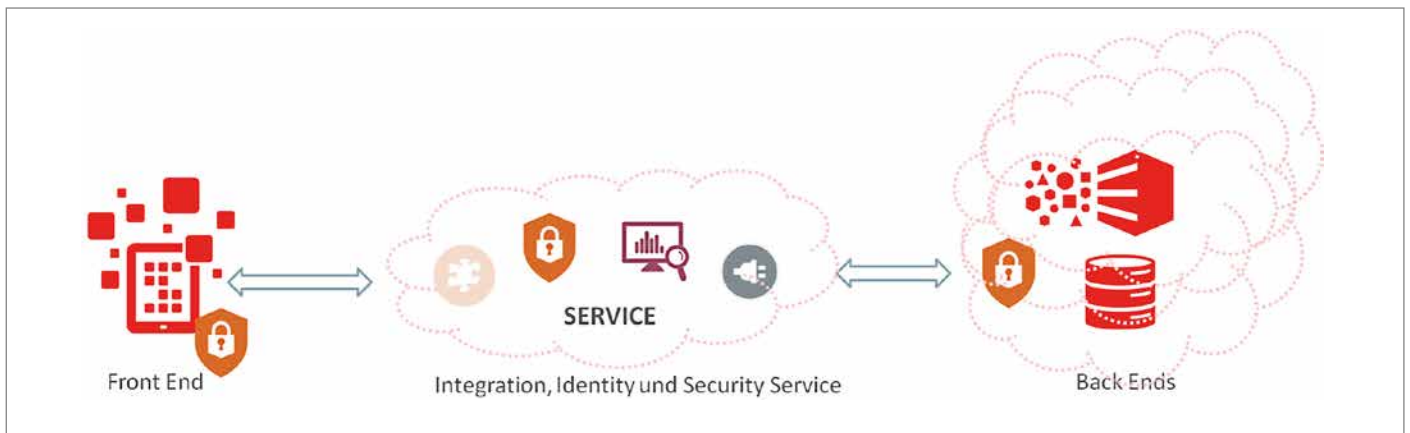


Abbildung 1: Architektur-Überblick Mobile Security

einer App oder das Übertragen von Bewegungsdaten bedingt durch das Bundes- oder EU-Datenschutzgesetz bereits einen Großteil dieser Funktionen impliziert fordern. Beispiele sind verschlüsselte Übertragung und Speicherung, Zugriffsbeschränkung für Administratoren, Nachweis für den Nutzer darüber, was gespeichert wurde, oder das Recht auf Löschungen.

Beteiligte Komponenten

Die beteiligten Komponenten lassen sich grob vereinfacht in einem Drei-Schichten-Modell darstellen (siehe Abbildung 1). Das Frontend beschreibt das Device, unabhängig davon, ob ein Browser oder eine App genutzt wird. Service- und Back-End-Layer sind Cloud-artig dargestellt, da es ja keine Rolle spielt, woher sie kommen, aus dem eigenen Rechenzentrum und/oder in Kombination mit anderen Anbietern aus der Cloud.

Klassischerweise ist ein Back-End-Layer vorhanden, bei traditionellen Unternehmen sind das die bestehenden Rechenzentren, bei neuen disruptiven Firmen kann dieser auch beim gleichen Betreiber liegen wie der Service-Layer oder ein Mash-up aus vorhandenen verteilten Services sein.

Der Service-Layer kann verschiedene Back-Ends integrieren, etwa Google Maps zusammen mit einem Asset-System. Als Beispiel würde der Servicetechniker eines Unternehmens bei einer Reparatur von Aufzügen Informationen zum Aufzug aus dem Asset-System bekommen und über Google Maps zum Reparaturort oder Ersatzteillager geleitet werden. Ein analo-

ges Endkundenbeispiel ist die Suche nach einem Buchhändler, der ein bestimmtes Buch vorrätig hat, inklusive einer Lotsenfunktion unter Berücksichtigung der Erreichbarkeit und der Öffnungszeiten.

Über den Service-Layer erfolgt auch eine Integration, die den Benutzer identifizieren kann und personalisierte Zugriffe auf Services erlaubt. Der Benutzer kann dabei entscheiden, wer bei Bedarf seine Identität verifiziert (etwa Facebook, Microsoft oder ein Unternehmen) und welche Daten an wen weitergegeben werden dürfen.

In der folgenden Unterteilung sind die Security-relevanten Funktionen aufgeführt. Funktionen zum Absichern des Front-Ends sind:

- **Gesicherte Ablauf-Umgebung**
Basis ist der Schutz des Device, etwa mit PIN und/oder Fingerprint. Zudem erfolgt durch Anti-Virus- und Anti-Malware-Lösungen eine bestmögliche Sicherung der Ablauf-Umgebung. Die geprüften Apps an sich und der Benutzer bleiben ohne größere Einschränkungen.
- **Sichere Ablauf-Umgebung/Container**
Eine sichere Ablauf-Umgebung, auch Container, verhindert den Zugriff von anderen Apps auf die laufende App oder deren Daten und steuert zudem die Möglichkeit, Daten zu exponieren, wie Drucken oder das Speichern in eine Cloud-Umgebung. Bei der Definition werden Device-seitig zulässige Apps festgelegt und zur Erkennung des Geräts in der Regel Zertifikate ausgerollt. Der Speicherbereich ist verschlüsselt, um die Daten auch offline

beziehungsweise auf dem Datenspeicher zu schützen. Die Verwaltung der Devices erfolgt über Policies. Vor der erstmaligen Verwendung ist ein Enrollment notwendig.

- **Schutz der gespeicherten Daten und Passwörter**
Passwörter sollten immer geschützt sein. Das kann durch Ablage in die vom mobilen Betriebssystem vorgesehenen Bereiche (wie Keychain) erfolgen, durch die App selbst oder nur Server-seitig (mit der Einschränkung bezüglich Offline-Funktionen). Bei der App selbst stellt sich wie bei jedem Security-Code die Frage, wie sicher eine Eigenentwicklung gegenüber einem fertigen Paket/SDK sein kann. Gespeicherte und schützenswerte Daten sollten schon wegen potenziellen Verlustes verschlüsselt sein, etwa durch eine mit AES verschlüsselte Datenbank.
- **Integration der Device-Funktionen wie Entsperrung per Fingerprint, Voice oder Selfie**
Zum Entsperren sollten die vom Hersteller vorgesehenen Mechanismen zum Einsatz kommen. Neben der PIN bieten biometrische oder Mehrfaktor-Authentifizierungen einen höheren Schutz. Dies sollte in der Anwendung zum Tragen kommen, damit der Vertrauensgrad der Umgebung mit berücksichtigt werden kann, etwa als aufsteigende Level: keine Sperre, PIN, Zertifikat, Bio, Einmal-Passwort. Auch kann eine Applikation weitere Authentifizierungen selbst verlangen, wenn entweder der Verdacht auf unsachgemäße Nutzung besteht (etwa Mehr-

fachaufruf von Personendaten) oder besonders sensible Funktionen (wie Buchung oder Überweisung) ausgeführt werden. Hier können auch zusätzliche, nicht vom Device bereitgestellte Verfahren integriert werden.

- *Single-Sign-on-Funktionen*

Um dem Benutzer ein echtes Single-Sign-on zu ermöglichen, müssen die Zugriffstoken zwischen den Apps und dem App-Browser austauschbar sein. Dies lässt sich durch entsprechende SDKs erreichen. Basis ist immer ein Anmeldeverfahren, das beispielsweise OAuth-Token erstellt, die dann von der jeweiligen App dem Server präsentiert werden.

Der Service-Layer stellt die Kontaktpunkte der Front-end-App dar:

- *Anmeldeverfahren, Step-up-Möglichkeiten, Single-Sign-on*

Wird für die App eine Identifikation des Benutzers serverseitig benötigt, würde sie diesen Service beziehen. Das Service-Prinzip ermöglicht den Austausch und die Erweiterung der Funktionalität, ohne die Apps ändern zu müssen, beispielsweise um eine stärkere Authentifizierung einzufordern oder ein Google-/Facebook-Login aus beziehungsweise einzuschalten. Der Service arbeitet in der Regel über Token.

- *Abstraktion Back-End-Schnittstellen über mobile, verträgliche Schnittstellen (wie REST, OAuth) und Nutzerbezug beziehungsweise Trust-Verhältnis mit den Schnittstellen*

Mobile Apps setzen typischerweise auf REST-Service-Aufrufe auf. Meist müssen diese noch propagiert oder erstellt werden. Im Zuge nicht trivialer Services ist auch hier ein Zugriffsschutz notwendig, der dann dort umgesetzt wird und den Zugriffstoken für das Back-End an sich in ein passendes Format (wie SAML) umwandelt.

- *Aggregation von Einzel- zu Compound-Services*

Bestehende Services haben in der Regel eine zu breite Schnittstelle und stellen oft Einzelfunktionen dar. Im Bereich „Mobile“ sind wegen der Datenrate schmale Services erforderlich;

um Entwicklern eine schnelle und einfache Erstellung zu ermöglichen, werden Services zusammengefasst. Auch hier gilt es, die entsprechenden Zugriffsberechtigungen zu propagieren.

- *Kontrollierte Weitergabe von Benutzer-Informationen*

Der Service-Layer steuert die kontrollierte Weitergabe von Benutzer-Informationen und führt den Nachweis darüber. Er berücksichtigt dabei die jeweilige Einverständniserklärung des Benutzers (etwa mittels User Consent). Der Nachweis ist durch das kürzlich verabschiedete EU-Datenschutzgesetz gefordert und wird künftig eine stärkere Rolle spielen.

- *Verifikation berechtigter Benutzer unter Berücksichtigung des Kontexts*

Um Risiko und Betrug zu entgegnen, erfolgt eine technische Bewertung des Aufrufs (etwa offenes WLAN vs. Firmennetz, registriertes vs. nicht-registriertes Device) und des Kontexts (wie zuvor durchgeführte Aktionen) dieses Aufrufs. Die Service-Komponente führt diese Bewertung durch und stellt das Ergebnis der App (oder deren Serverseite) zur Verfügung oder schreitet direkt ein (etwa durch nur noch Lesezugriffe oder Re-Authentifizierung mit Einmal-Passwort).

- *Administrationsmöglichkeit und Auditierung*

Zur Verwaltung der Benutzer-Accounts und Berechtigungen ist eine Komponente notwendig. Abhängig vom Service muss auch der Nachweis geführt werden, wer Berechtigungen zugeordnet oder genehmigt hat. Für kritische Berechtigungen ist eine Rezerifizierung erforderlich. Im Blickfeld sind nicht nur Anwendungs-Accounts, sondern auch die Accounts der Betreiber und Entwickler. Der Übergang von Consumer-Services zu Unternehmensanwendungen ist hier fließend; diese Anforderungen betreffen auch Consumer-Anwendungen, wenn es etwa zum Abfluss von Accounts/Daten durch einen der privilegierten Accounts kommt.

- *Service gemäß CIA-Prinzip*

CIA steht für „Confidentiality“, „Integrity“ und „Availability“. Dies muss über die Services, je nach gewünschtem Service-Level, gewährleistet sein und

kann über die „state of the art“-Mechanismen wie Zugriffsbeschränkung (Anmeldung, Autorisierung, rollenabhängige, zweckgebundene Nutzung), Verschlüsselung oder Anonymisierung, Auditierung sowie optional Hochverfügbarkeit erreicht werden.

Die Funktionen des Back-End-Layers stimmen größtenteils mit denen des Service-Layers überein und sind dort bereits beschrieben:

- Verifikation des Aufrufers (Trust, Weitergabe Benutzer-Informationen)
- Verifikation berechtigter Benutzer beziehungsweise Service-Schnittstelle unter Berücksichtigung des Kontexts
- Administrationsmöglichkeit und Auditierung
- Service gemäß CIA-Prinzip

Die Akteure

Eine grobe Unterscheidung in die verschiedenen Akteure „Endbenutzer“, „Entwickler“, „Betreiber“ und „Unternehmen“ hilft, weitere Aspekte darzustellen. Die Funktionen überschneiden sich teilweise mit den den jeweiligen Layern zugeordneten Funktionen. Für den Endbenutzer gilt:

- Auswahl des Identity-Providers je Nutzung
- Auswahl, ob Credentials lokal gespeichert werden sollen
- Einmalige Anmeldung und damit SSO zwischen Apps und App-Browser
- Auskunft über gespeicherte Daten
- Erlaubnis für einen Service, persönliche Daten nutzen zu können; einmalig, befristet oder bis auf Widerruf
- Steuerung der Security: soviel Security, wie der Benutzer als notwendig empfindet (mit dem Risiko, manche Funktionen nicht nutzen zu können)

Für den Betreiber gilt:

- Unterstützung bei Administration und Auditierung
- Unterstützung bei CIA

Für das Unternehmen gilt:

- Koppelung an das Unternehmens-Identity- und Access-Management

- Abbildungsmöglichkeiten von Schutzklassen
- Definition und Pflegemöglichkeiten für Policies
- Steuerung zulässiger Apps und Konfigurations-Einstellungen, wie verschlüsselte Verbindungen, Daten nur verschlüsselt gespeichert und nur nach Freigabe von anderen Apps nutzbar

Für den Entwickler gilt:

- Nutzung bereitgestellter Sicherheits-Funktionen
- Konzept/Vorgaben Best Practices
- Keine Entwicklungsmöglichkeit in der Produktion
- Möglichkeit, anonymisierte Testdaten zu nutzen
- Zugriffsmöglichkeit auf Notfall-Accounts

Umsetzung mit Oracle

Für das Front-End bietet Oracle zur Umsetzung der skizzierten Funktionen eine Entwicklungsumgebung zur Entwicklung mobiler Apps für iOS, Android und Windows (Desktop-Tool: MAF – Mobile Application Framework, Server-basiert: MAX – Mobile Application Accelerator, siehe *Abbildung 2*). Hier sind die grundlegenden Funktionen für Authentifizierung und Integration in ein Server-seitiges Identity- und Access-System. Eine verschlüsselte Datenspeicherung und Synchronisation mit einer Server-seitigen Datenhaltung ist möglich. Die Verbindungen sind optional verschlüsselt. Da dies native Apps sind, ist auch eine Offline-Funktion möglich. Optional kann diese App in einem 3rd-Party-Container betrieben werden.

Browser-basierte Anwendungen können mit den Fusion-Middleware-Komponenten auf WebLogic-Basis und beispielsweise mit dem JDeveloper entwickelt werden. Dort stehen alle Identity- und Access-Funktionen zur Verfügung (siehe Service-Layer). Natürlich lassen sich diese Anwendungen auch aus der Datenbank heraus mit Apex programmieren. Dabei kann Apex für Authentifizierung („ootb“ mit Oracle) und Autorisierung („build“) in das Identity und Access Management eingehängt werden.

Im Oracle-Cloud-Angebot gibt es Werkzeuge, um einfache Anwendungen ohne



Abbildung 2: Front-End-Beispiel mit MAF und MAX

Funktion	Cloudbasierte Lizenz	On-Premise-Lizenz
Mit Securityfunktionen integrierte Entwicklungsumgebung	Mobile Application Accelerator (MAX)	Mobile Application Framework (MAF), Oracle JDeveloper, Oracle APEX
Gekapselte oder standardisierte Security-Funktionen zur Verwendung in beliebigen Umgebungen	Oracle Identity Cloud Service (IDCS)*	SDK der Oracle Access Management Suite bzw. die Standardschnittstellen wie OAuth, SAML etc.,
Verschlüsselte Datenhaltung auf Smartphone, Tablet		SQLite von MAF (DB), Oracle Mobile Security Suite (Container)
Verschlüsselte Übertragung	https	https

*) noch nicht verfügbar

Tabelle 1

Entwickler-Know-how erstellen zu können (MAX – Mobile Application Accelerator).

Oracle bietet SDKs, um die zentralen Identity- und Access-Funktionen unabhängig der verwendeten Entwicklungsumgebung in native Apps zu integrieren. Das geht über pure OAuth- und User/PW-Funktionen hinaus, etwa mit Device-Registrierung, „step-up“-Authentifizierung oder kontextbasierter Autorisierung. Darüber hinaus wird ein Token-basiertes SSO zwischen Apps sowie zwischen App und Browser bereitgestellt.

Oracle und Partner bieten eigene SDKs an, um die Funktionen des Oracle Mobile Cloud Service (der auch Identity- und Access-Funktionen enthält) nativ in Entwicklungsumgebungen zu nutzen. Dies sind zurzeit Xamarin (jetzt Microsoft

und Sencha. *Tabelle 1* zeigt die Security-relevanten Funktionen im Überblick.

Der Service-Layer stellt viele der Sicherheits-relevanten Funktionen bereit. Manche Funktionen sind in mehreren Schichten vorhanden, um eine mehrschichtige Security (bei Oracle sogenanntes „defense in depth“) umzusetzen (*siehe Abbildung 3*).

Identity- und Access-Funktionen für die Anmeldung über Browser oder REST/App mittels Standard-Protokollen wie OAuth, SAML, OpenID, „Step-up“-Authentifizierung, OTP, TOTP, SSO und kontextbasierter Autorisierung werden bereitgestellt. Eine Administration der Accounts und Berechtigungen inklusive Governance-Funktionen steht zur Verfügung. Diese Services laufen auf einer WebLogic-Plattform und können mit bestehenden Funktionen (An-

meldeverfahren -> SSO, Userstore -> Konnektoren, Genehmigungs- und Rezertifizierungs-Workflows) verknüpft sein.

Die Identity- und Access-Services werden zeitnah auch stand-alone als Cloud Services (Identity Cloud Service) zur Verfügung stehen. Integriert in andere Oracle Public Cloud Services sind sie heute schon verfügbar.

Das API-Gateway sichert die Service-Aufrufe ab, es ist so etwas wie der erste Security-Layer, den die App oder -Browser-basierte Applikation aufruft. Hier wird bei Bedarf nochmals geprüft, ob eine ausreichend starke Anmeldung und Autorisierung vorliegt. Zudem ist dies der Punkt, bei dem „on the fly“ in den Datenstrom eingegriffen werden kann (etwa zum Filtern oder Anonymisieren von Daten). Weitere Funktionen hinsichtlich Virenabwehr oder DoS sind ebenfalls vorhanden.

Die Integrations-Plattform nutzt die gleichen Mechanismen wie die genannten Identity- und Access-Funktionen. Sie ist sowohl als On-Premise-Installation als auch als Cloud Service verfügbar. Der Mobile Cloud Service bietet eine vorgefertigte Plattform, um Apps und Integration in Back-Ends zu bauen. Dieser Service wurde schon in einer der früheren Ausgaben vorgestellt. *Tabelle 2* zeigt die Security-relevanten Funktionen im Überblick.

Der Back-End-Layer ist entweder schon vorhanden und muss nur die Aufrufbarkeit über mobile Schnittstellen ermöglichen oder wird „from the scratch“ zusammen mit der Service-Schicht erstellt. Sind die Service- und Back-End-Funktionen getrennt, bietet sich auch hier eine vorgeschaltete, Gateway-artige API-Funktion zur Entkopplung und als Security-Layer an (siehe Beschreibung Service-Layer und *Abbildung 4*).

Ein Beispiel

Das folgenden Beispiel zeigt die Umsetzung eines Kunden, der für seine Märkte zwei Typen von Apps zur Verfügung stellt: eine für seine Endkunden für Informationen und Bestellungen (zur Abholung) und eine Tablet-App für die Angestellten, damit diese eine persönliche Kunden-Ansprache durchführen können (*siehe Abbildung 5*).

Als Front-End wurde eine native App für iOS und Android mit Oracle MAF umgesetzt. Oracle MAF authentifiziert den Benutzer gegen das bestehende Kunden-

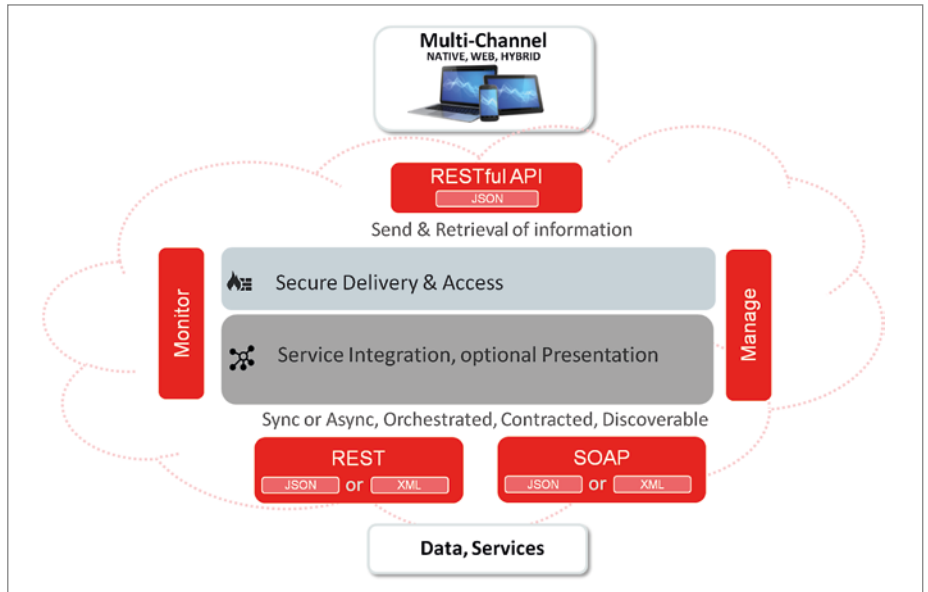


Abbildung 3: Service-Layer-Funktionen IAM und Service-Integration

Funktion	Cloudbasierte Lizenz	OnPremise-Lizenz
Authentifizierung, SSO	Oracle Mobile Cloud Service(MCS), Oracle Identity Cloud Service (IDCS)*	Oracle Access Management Suite
Starke Authentifizierung, Risk & Fraud-Detection	IDCS*	Oracle Access Management Suite
Autorisierung	MCS, IDCS*	Oracle Access Management Suite
API-Schnittstelle, REST/SOAP	MCS, Oracle API Cloud Service (API CS)*	Oracle API Gateway
Service-Composition & Integration	MCS, ICS, SOA CS	Oracle API Gateway, Oracle SOA Suite
Sichere Speicherung von Daten (inkl. sensibler Daten)	Oracle Database as a Service (DBaaS)	Oracle-Datenbank (ASO-Verschlüsselung, DBVault - Gewaltentrennung, AuditVault-Audit)

*) noch nicht verfügbar

Tabelle 2

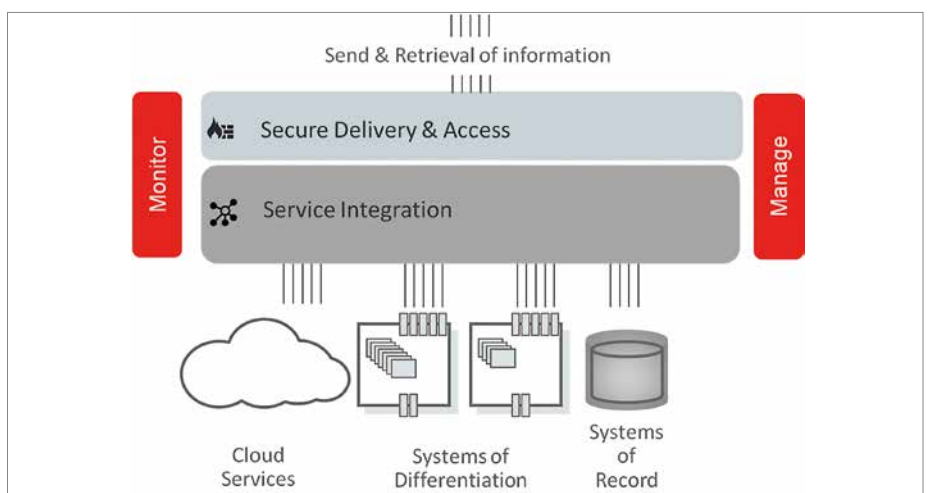


Abbildung 4: Back-End-Funktionen

Portal im Internet. Dabei sind verschiedene Verfahren möglich, etwa über Social Logins. Die App ist Offline-fähig, um Bestellungen auch ohne Netzzugriff in einer der 72 Filialen abholen zu können, ähnlich wie bei Flugtickets. Die Bestellmöglichkeit ist dabei nur einer von vielen bereitgestellten Services.

Die notwendigen Bestell- und Kundenservices bestanden bereits und werden aus dem Rechenzentrum des Kunden abgerufen oder manuell in den Filialen erbracht. Sie werden sowohl über die Kanäle „Web-Portal“ und „Kiosk“ als auch über andere Integrationen (Co-Shopping/Resell und InternetTV) genutzt.

Die bestehenden Services sind durch eine Service-Infrastruktur so zusammengesetzt, dass sie die passende Granularität für die Schnittstelle zur App bereitstellen. Im Back-End wird dabei auf SOAP als Schnittstelle verblieben. Damit konnten ohne Änderung die bestehenden „WS“-Mechanismen zur Absicherung der Services beibehalten werden.

Das API-Gateway im Rechenzentrum nimmt den Request einer App entgegen. Es löst dabei drei Aufgabenstellungen: die Bereitstellung der Services als REST-Services, um diese aus einer App effizient nutzen zu können, die Absicherung der Aufrufe und die Transformation der Security-Informationen in/für nachfolgende Layer. Mit dieser Architektur hat der Kunde seine Multi-Channel Architektur umgesetzt, Silos vermieden und einen hohen Grad an Wiederverwendung erreicht.

Da sensible, personenbezogene Daten verarbeitet werden, ist die Security von Anfang an einbezogen und um möglichst flexibel zu bleiben, hat man auf die entsprechenden Standards gesetzt. Durch die konsequente Service-Orientierung, sowohl bei den Geschäftsprozessen als auch bei den Security-Funktionen lassen sich neue Kanäle und Funktionen effizient bereitstellen. Aktuell erfolgt die Erweiterung um Beacons in den Filialen und IoT-Devices, auch Wearables, um ein optionales persönliches Monitoring im Bereich „Gesundheit“ zu unterstützen.

Fazit

Es ist sinnvoll, Mobile Security von Anfang an zu berücksichtigen, um nicht später Apps mit erhöhtem Aufwand umbauen

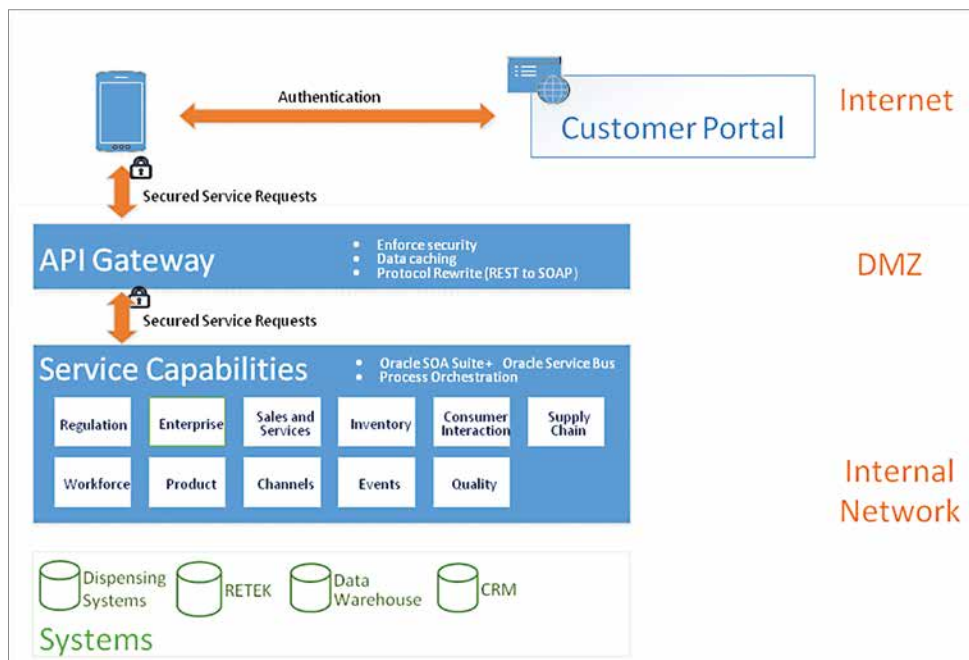


Abbildung 5: Umsetzungsbeispiel eines Oracle-Kunden

oder erweitern zu müssen. Beispiele für eine Erweiterung sind aus technischer Sicht beispielsweise die Grenzen des Anmeldeverfahrens bezüglich Skalierung oder neue Technologien (wie OpenID Connect Integration). Auch von außen bestehen Anforderungen, etwa durch Auflagen von Auditoren oder allgemeine Regularien, ohne die Betrachtung von Breaches. Auch der Betriebsaspekt spielt eine Rolle, vielleicht nicht so sehr bei der ersten App, doch aber mit mehreren Apps, die vielleicht alle ein eigenes User-Management mitbringen. Daher könnten folgende Empfehlungen zum Tragen kommen:

- Ganzheitliches Konzept, keine Silos
- Sicherheit von Anfang an, etwa verschlüsselte Verbindungen bei sensiblen oder personenbezogenen Daten
- Sicherheit nach Bedarf: zum Beispiel „unprotected“, „protected“, „encrypted“ sowie kontextabhängig
- Mindeststandards für Entwicklung
- Zentrale Kontrolle
- Audit und Regularien im Auge behalten
- Integration in Unternehmens-IT, sofern vorhanden
- Nicht alles selber aufsetzen/bauen
- Lifecycle im Auge behalten (Update, Multi-Plattform-Support)
- Gängige Protokolle und Plattformen unterstützen

- Testballons zum Start mit minimalen Kosten auf einer fertigen Plattform

Oracle unterstützt mit seinen Komponenten eine Umsetzung aller dieser Punkte On Premise, in der Cloud oder hybrid. Durch die Verwendung von standardisierten Schnittstellen und Protokollen ist dabei auch ein Mix mit Komponenten möglich, die nicht von Oracle kommen. Viele Kunden in allen Größenordnungen und durch alle Branchen nutzen diese Komponenten. Weitere Informationen unter „<http://www.oracle.com/us/technologies/security/overview/index.html>“.



Michael Fischer
michael.fischer@oracle.com



Oracle Forms goes Mobile

Jan-Peter Timmermann, Jan-Peter Timmermann Consulting

Seit Jahren hört man immer wieder von aktuellen Technologien in der Entwicklung. Da sind Schlagworte wie „IT 4.0“, „Mobile“, „Cloud“ und so weiter im Gespräch. Es gibt mit Sicherheit gute Gründe, sich über diese Entwicklungen Gedanken zu machen und auch die Entwicklung der Unternehmensanwendungen in die eine oder andere Richtung zu überdenken.

Was geschieht jedoch, wenn eine unternehmenskritische Anwendung auf einer alten Technologie basiert – und das schon mehr als zwanzig Jahre? Das gesamte Wissen steckt in dieser Anwendung. Die Anwender arbeiten damit bereits seit einigen Jahren und das nahezu blind. Die Lösung deckt genau das ab, was das Unternehmen braucht.

Nun kommen aber immer mehr Anforderungen auf die Entwickler zu, auch Anwendungen für mobile Endgeräte anzubieten. Vieles, was für das mobile Endgerät gefordert wird, ist bereits in der Forms-Anwendung hinterlegt. Muss man nun alles neu entwickeln, wie schaut es aus mit Security, was ist, wenn kein Internet-Empfang besteht? So kommen komplett neue Fragen auf die Entwickler und die Betreiber zu.



Abbildung 1: Die Vorstellung von „Forms Mobile“



Abbildung 2: Mobile Erwartung

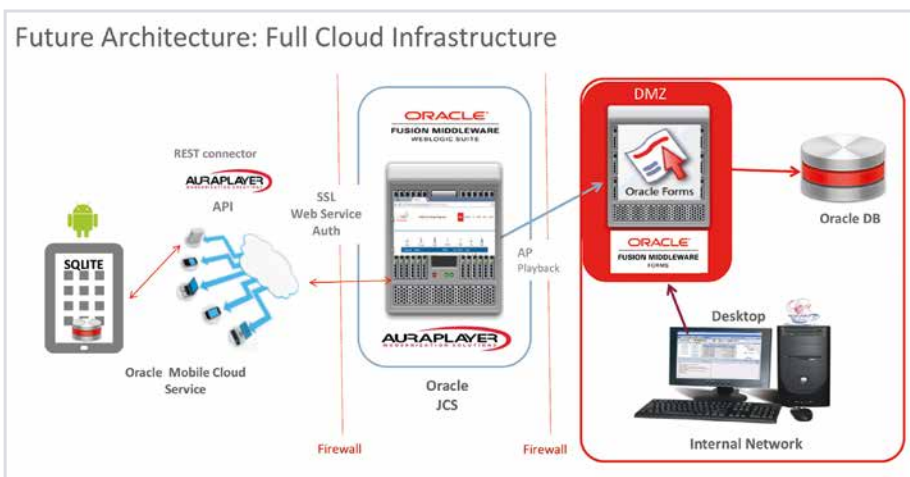


Abbildung 3: Eine mögliche Architektur

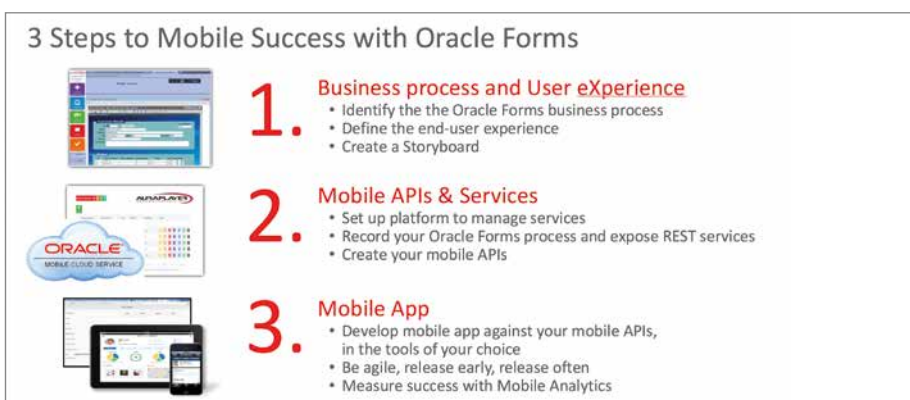


Abbildung 4: Die notwendigen Schritte

Jetzt hat Oracle im letzten Jahr endlich das lange versprochene Release Oracle Forms/Reports in der Version 12c auf den Markt gebracht. Oracle Forms 12c arbeitet mit den aktuellen JDKs zusammen, sodass hier von Seiten der Applika-

tions-Server ein sehr aktueller Stand vorgehalten ist.

Nun gibt es verschiedene Ansätze, mit diesen Themen umzugehen. Viele Entwickler werden auf eine neue Entwicklung hinarbeiten. Es gibt aber genauso den Trend,

bestehende Forms-Anwendungen in Richtung einer neuen Technologie zu migrieren. Leider birgt dieses Vorgehen diverse Risiken, auf die hier nicht weiter eingegangen wird. Trotzdem braucht es auch seine Zeit, eine neue Anwendung zu erstellen oder eine bestehende zu migrieren. Was geschieht in der Zwischenzeit?

Der Artikel beschreibt die Möglichkeit, eine bestehende Forms-Anwendung über einen Webservice für mobile Anwendungen bereitzustellen. Zuvor ist es wichtig zu verstehen, dass man nicht die Forms-Anwendung mobil bereitstellen möchte, sondern immer den Geschäftsprozess. Wenn man sich mit Forms-Anwendern oder -Entwicklern unterhält und sie dann fragt, was sie sich unter einer mobilen Forms-Anwendung vorstellen, stellt man sehr schnell fest, dass meistens ein bestimmtes Bild herauskommt (siehe Abbildung 1).

Man will natürlich die Möglichkeiten ausnutzen, die einem Smartphones und Tablets anbieten. Wer eine Adresse hat, erwartet auch, dass ihm diese in einer Karte angezeigt wird. Ebenso möchte man, dass der Kunde angerufen wird, wenn man auf seine Telefonnummer klickt. So sieht die Erwartungshaltung eher wie in Abbildung 2 aus.

Alle diese Informationen liegen innerhalb der Forms-Anwendung vor. Man muss sie nur als Webservice bereitstellen. Die Schritte sind bei allen Tools ähnlich. Es muss in jedem Falle ein Storyboard erstellt werden, das die Geschäftsprozesse abbildet. Dazu gibt es unterschiedliche Ansätze. Die einen packen in jeden Sourcecode weiteren Code, um den Prozess aufzuzeichnen, andere bevorzugen das Aufzeichnen der Forms-Schritte, etwa mit der Oracle Application Testing Suite.

Wir werden an dieser Stelle das Tool „AuraPlayer“ intensiver betrachten. Es ist als Tool insofern interessant, weil man nicht in den bestehenden Code eingreifen muss, sondern das Ausführen der notwendigen Schritte in einem Storyboard erstellt – ohne eine einzige Zeile Code schreiben zu müssen. Diese Technik funktioniert mit allen notwendigen Operationen, die im Forms-Umfeld bereitstehen. Das ist zum einen natürlich die Abfrage, zum anderen kann man auch Daten einfügen, ändern oder aber auch über den erstellten Webservice löschen.

Was das Tool nicht kann, ist das Erstellen einer mobilen Anwendung. Es ist ein-

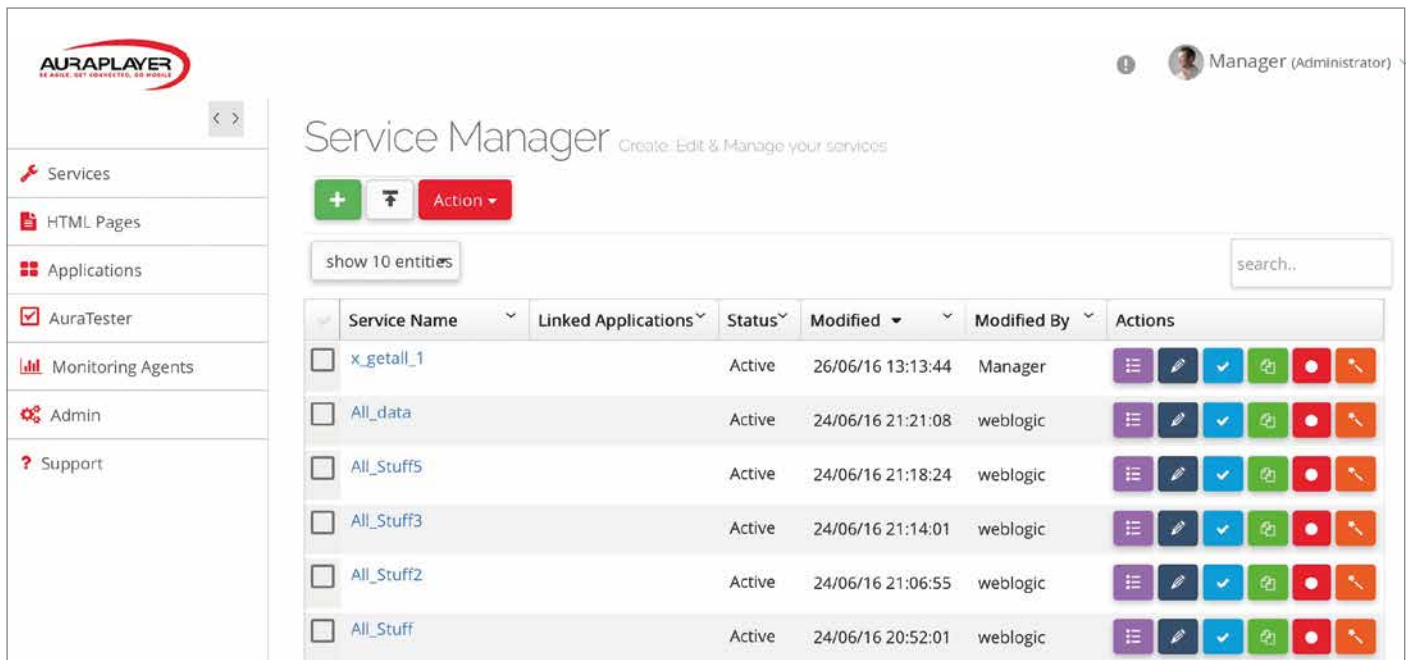


Abbildung 5: Der Service Manager

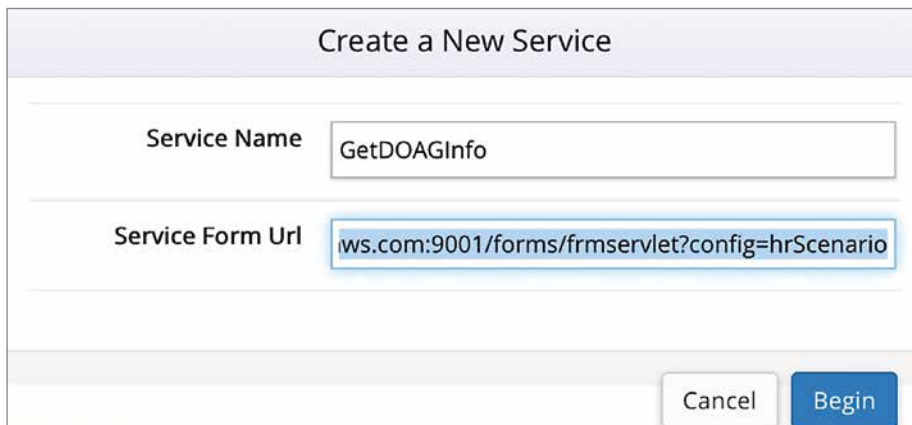


Abbildung 6: Das Storyboard starten



Abbildung 7: Die Anwendung starten

zig und allein dazu ausgelegt, den Webservice bereitzustellen. Dieser greift bei Bedarf immer auf die aktuelle Forms-Maske zu und führt dies zur Laufzeit im Hintergrund aus. Das heißt allerdings auch, dass alle Änderungen, die in dem Forms-Modul erfolgen, sich automatisch auf den Webservice durchziehen. Ebenso werden alle Validierungen, die innerhalb eines Forms-Moduls ausgeführt werden, auch durch den Webservice ausgeführt. *Abbildung 3* zeigt die Architektur der Forms-Umgebung mit dem AuraPlayer.

Es wurde hier der komplette Cloud-basierten Ansatz gewählt. Es ist zwar ebenso möglich, dies „in-house“ zu machen, der Oracle Mobile Cloud Service bietet jedoch verschiedene Vorteile, die man nutzen sollte. Bevor wir uns nun den einzelnen Schritten zuwenden, hier noch einmal ein Überblick darüber, was zu tun ist (*siehe Abbildung 4*). Der System-Manager ist das Steuerelement, um die Webservices zu erstellen und zu verwalten (*siehe Abbildung 5*).

Im ersten Schritt wird immer über den Service Manager ein Storyboard erstellt. Dazu startet man einfach eine Forms-Session, in der man sein Geschäftsmodell abbilden möchte. Aufrufparameter ist die URL der Forms-Anwendung (*siehe Abbildung 6*). Dieser Parameter kann sich sowohl auf eine Test- als auch auf eine Produktiv-Umgebung beziehen, da hier keine Eingriffe in der Anwendung erfolgen. Mit

diesem Aufruf werden die Forms-Anwendung gestartet und alle anschließend getätigten Schritte aufgezeichnet (siehe Abbildung 7). Das wird über die Steuerleiste bestimmt (siehe Abbildung 8).

Mit „Capture Parameter“ werden alle Items einer Maske in den Webservice übernommen, ebenso kann man durch Anklicken der Items entscheiden, welche Informationen bereitgestellt werden sollen. Die Entscheidung fällt hier für die Übernahme aller Items. Daraufhin wird der Webservice genau mit den Informationen erstellt, die man haben möchte (siehe Abbildung 9).

Es sind nun alle Items der Abfrage hinterlegt. Zudem ist die Abfrage aufgezeichnet worden. Man kann jetzt daraus über den Create Service einen Webservice erstellen und sich über die URL „`http://<HOSTNAME>:7001/ServiceManager/Macro/ExecMacro/GetDOAGInfo?wsdl`“ anzeigen lassen (siehe Listing 1).

Mit diesen Informationen lässt sich nun eine Web-Anwendung erstellen. Das

kann eine MCS-Anwendung sein, genauso könnte man die Information auch in eine Apex-Lösung einbauen. Automatisches Testen mit diesen Informationen ist ebenso möglich, denn man verfügt über alle Informationen, die zum Testen einer Forms-Anwendung nötig sind. Dazu muss man nur noch diesen Webservice zum Test Case machen (siehe Abbildung 10).

Dieser Test Case versetzt einen in die Lage, über Parameter/Items Validierungen daraufhin zu testen, ob die Anwendung Fehler erzeugt oder genau das Ergebnis liefert, das erwartet wird. Es besteht die Möglichkeit, über Textdateien verschiedene Eingaben zu steuern und die unterschiedlichen Eingaben gegen Parameter zu validieren. Zudem kann man feststellen, wie schnell die Forms-Anwendung antwortet.

Über die Validierungen legt man die Bedingungen dafür fest, ab wann ein Test erfolgreich durchlaufen ist. Natürlich gibt es hier die Möglichkeit, mehrere Test Cases zu einer Teststrecke zusammenzufas-

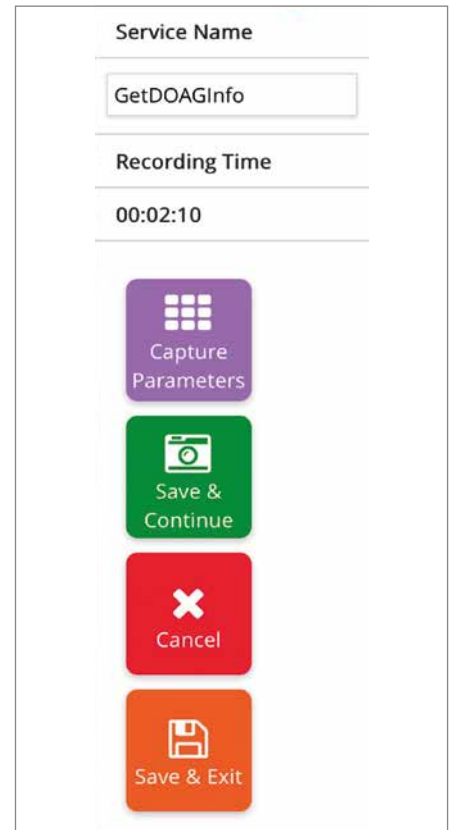


Abbildung 8: Die Steuerung

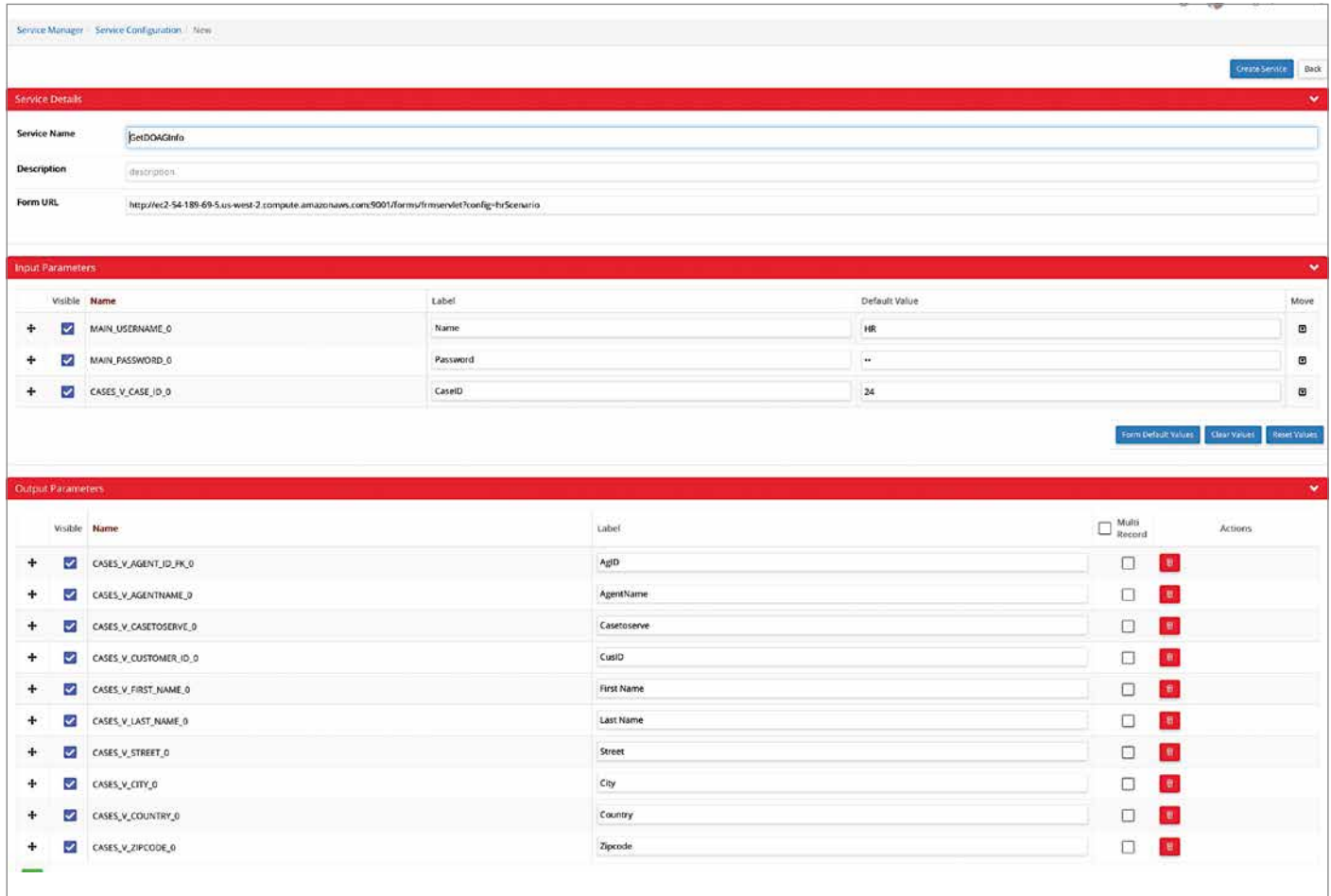


Abbildung 9: Den Webservice erstellen

```

<wsdl:definitions targetNamespace="http://services.auraplayer.com">
  - <wsdl:types>
    - <schema elementFormDefault="qualified" targetNamespace="http://services.auraplayer.com">
      - <element name="GetDOAGInfo">
        - <complexType>
          - <sequence>
            <element name="MAIN_USERNAME_0" type="xsd:string"/>
            <element name="MAIN_PASSWORD_0" type="xsd:string"/>
            <element name="CASES_V_CASE_ID_0" type="xsd:string"/>
          </sequence>
        </complexType>
      </element>
      - <element name="GetDOAGInfoResponse">
        - <complexType>
          - <sequence>
            <element name="GetDOAGInfoReturn" type="impl:GetDOAGInfoResponse"/>
          </sequence>
        </complexType>
      </element>
      - <complexType name="GetDOAGInfoResponse">
        - <sequence>
          <element name="GetDOAGInfoMessage" type="impl:GetDOAGInfoMessage"/>
          <element name="GetDOAGInfoElements" type="impl:GetDOAGInfoElements"/>
        </sequence>
      </complexType>
      - <complexType name="GetDOAGInfoMessage">
        - <sequence>
          <element name="Error" type="string"/>
          <element name="PopupMessages" type="string"/>
          <element name="StatusBarMessages" type="string"/>
        </sequence>
      </complexType>
      - <complexType name="GetDOAGInfoElements">
        - <sequence>
          <element name="CASES_V_AGENT_ID_FK_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_AGENTNAME_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_CASETOSERVE_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_CUSTOMER_ID_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_FIRST_NAME_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_LAST_NAME_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_STREET_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_CITY_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_COUNTRY_0" nillable="true" type="xsd:string"/>
          <element name="CASES_V_ZIPCODE_0" nillable="true" type="xsd:string"/>
          <element name="CASES.V_CASE_ID_0)" nillable="true" type="xsd:string"/>
        </sequence>
      </complexType>
    </schema>
  </wsdl:types>

```

Listing 1

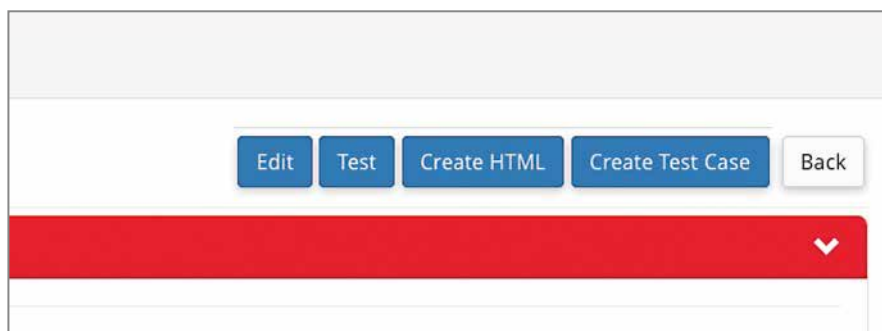


Abbildung 10: Den Test Case erstellen

sen und diese als Gesamtes zu betrachten (siehe Abbildung 11).

Fazit

Der Autor ist seit Jahren ein Freund von Oracle Forms, sieht sich aber immer häufiger dem Thema gegenüber, modern arbeiten zu müssen. Mit dem AuraPlayer hat er eine Möglichkeit gefunden, mit überschaubarem Aufwand seine bestehenden

Output Parameters		
Name	Label	Validations
CASES_V_AGENT_ID_FK_0	AgID	
CASES_V_AGENTNAME_0	AgentName	
CASES_V_CASETOSERVE_0	Casetoserve	
CASES_V_CASE_ID_0	CaseID	
CASES_V_CUSTOMER_ID_0	CusID	
CASES_V_FIRST_NAME_0	First Name	
CASES_V_LAST_NAME_0	Last Name	
CASES_V_STREET_0	Street	
CASES_V_CITY_0	City	
CASES_V_COUNTRY_0	Country	
CASES_V_ZIPCODE_0	Zipcode	
PopupMessages	popup messages	
StatusBarMessages	status bar messages	
Error	error	Fail if Response is NOT empty

Response Time Validation	
Timeout	10 seconds

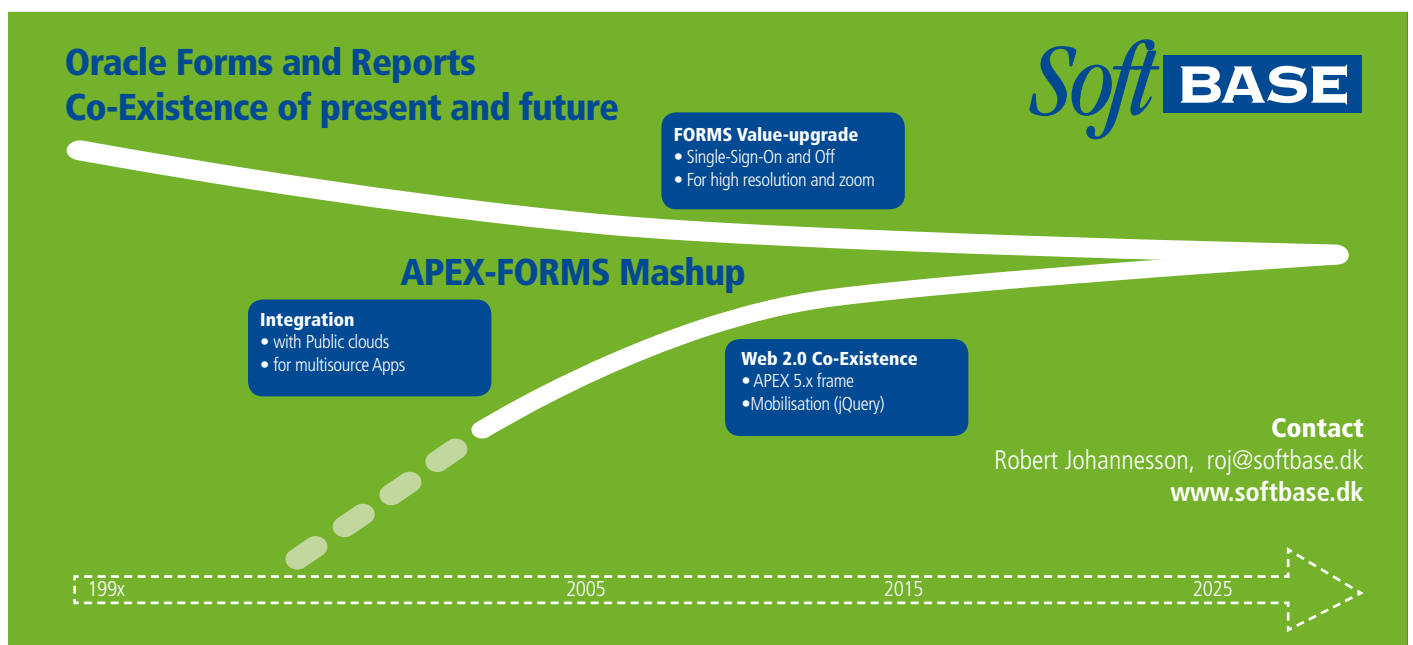
Abbildung 11: Validierungen

Forms-Information zu nutzen und für mobile Anwendung bereitzustellen.

Das Tool dient nicht dazu, Forms abzulösen, sondern besteht vielmehr aus dem Zusammenspiel von Forms und Webservice. Es bietet damit die Möglichkeit, sehr schnell Anwendungen für das Web bereitzustellen. Der Webservice lässt sich als Plug-in für die MCS nutzen, man spricht ihn direkt an oder implementiert ihn in Apex – die Möglichkeiten sind vielfältig.



Jan-Peter Timmermann
jan.timmermann@jptu.de





PL/SQL-Logik Beine machen: mobil durch Webservices

Daniel Kampf, PITSS GmbH

Wer seine Business-Logik in der Datenbank organisiert hat, stellt sich häufig die Frage, wie diese für mobile Anwendungen genutzt werden kann. Der Artikel gibt einen Überblick über die Möglichkeiten der Webservice-Generierung zur Erschließung von PL/SQL-Logik.

Um das User Interface unabhängiger von Technologie-Wechseln zu machen, haben viele Unternehmen ihre Business-Logik in die Datenbank überführt. Damit dieser Code anderen Anwendungen zur Verfügung steht, eignen sich besonders Webservices als universelles Werkzeug. Sowohl Enterprise-Anwendungen als auch mobile Apps kommen als deren Konsumenten infrage. Dieser Artikel zeigt, wie PL/SQL von einem Webservice aus aufgerufen und dieser Service wiederum in den Oracle Mobile Cloud Service eingebunden wird.

Webservice ist nicht gleich Webservice

Zunächst ein Blick auf die verschiedenen Möglichkeiten zur Implementierung ei-

nes Webservice. Abhängig von der Ziel-Anwendung ist es wichtig, den richtigen Webservice zu wählen. Es gibt aktuell im Prinzip zwei verschiedene Typen von Webservices: den etwas in die Jahre gekommenen SOAP- und den RESTful-Webservice. SOAP zeichnet sich durch seine XML-Struktur aus und ist durch eine Vielzahl von Meta-Informationen schnell aufgebläht. Vorteil ist jedoch, dass der Verwender eines SOAP-Webservice über die WSDL-Seite gut einschätzen kann, in welcher Form die Daten vorliegen. Dies ist aber auch der größte Nachteil: Die Payload, also die Daten, die beim Aufruf eines Webservice übermittelt werden, ist recht groß.

Bei RESTful-Webservices ist dieses Problem nicht vorhanden – sie sind deutlich leichtgewichtiger. Hier ist die Be-

ziehung zwischen Service-Anbieter und -Konsument sehr lose: Eine WSDL-Datei, die den Service und alle Methoden beschreibt, ist in Form einer WADL nur rudimentär vorhanden. Diese Eigenschaften machen den RESTful-Service zur ersten Wahl, wenn es um die Entwicklung von mobilen Lösungen geht, da er seine Vorteile bei der Leichtigkeit der Datenübertragung ausspielt.

Zugriff auf die Datenbank

Neben der Wahl des Webservice für die Anbindung der PL/SQL-Logik stellt sich auch die Frage, mit welchen Tools und auf welcher Plattform der Webservice erzeugt beziehungsweise betrieben werden soll. Oracle bietet zwei grundsätzliche

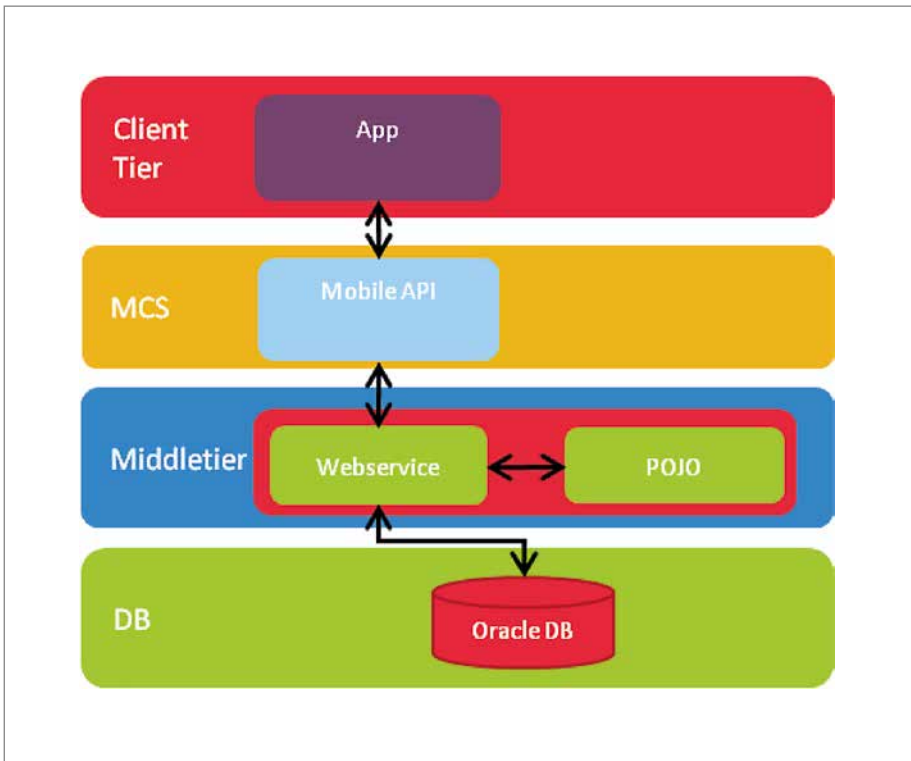


Abbildung 1: Szenario einer mobilen Lösung

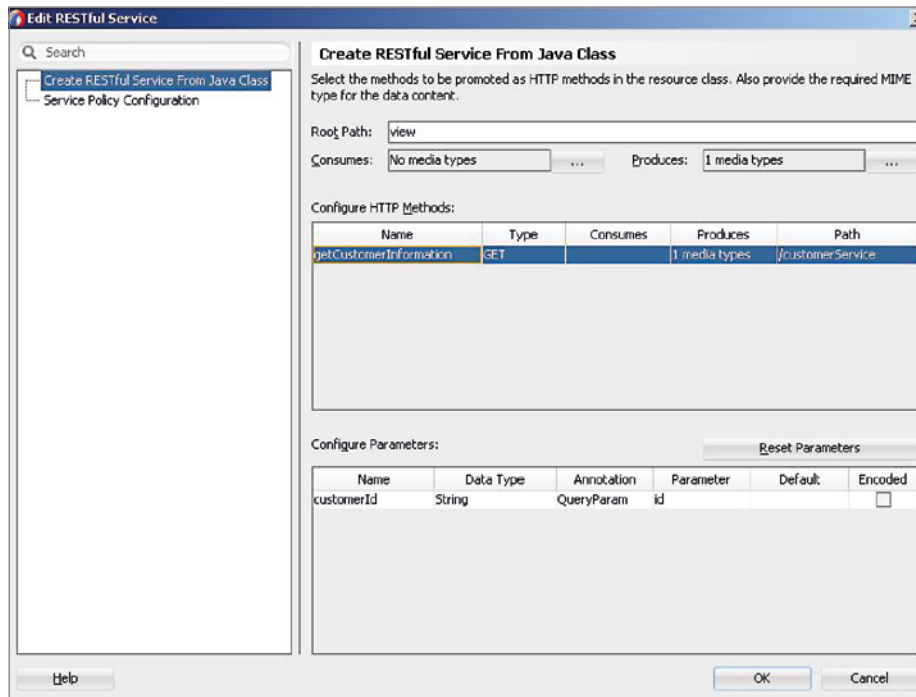


Abbildung 2: Optionen für die Webservice-Generierung

```

27 public class CustomerInformation {
28
29     @GET
30     @Produces("application/json")
31     @Path("/customer")
32     public Customer getCustomerInformation(@QueryParam("id") String customerId){
33

```

Abbildung 3: Annotationen im Java-Code

Möglichkeiten an. Auf der einen Seite lässt sich die Generierung mithilfe des JDeveloper durchführen. Die andere Alternative ist der Oracle REST Data Service (ORDS). Er unterstützt, wie der Name schon sagt, nur die schlankeren RESTful-Webservices, wohingegen im JDeveloper beide Varianten möglich sind. In diesem Artikel liegt der Fokus auf der Implementierung eines Java-Webservice, basierend auf JAX-RS und dem WebLogic-Server in der Version 12.2.1.0.0 als Plattform (siehe Abbildung 1).

RESTful Java-Webservice mit JAX-RS

Alle modernen IDEs sind in der Lage, auf Basis einer Java-Klasse einen Webservice zu generieren – so auch der JDeveloper. Zunächst sind ein neuer Workspace und ein neues Java-Projekt erforderlich. In diesem Beispiel wurde danach im JDeveloper eine Java-Klasse mit dem Namen „CustomerInformation.java“ erstellt, die über den Connection-Pool des Middleware-Servers eine Datenbank-Verbindung zugewiesen bekommt und mit deren Hilfe das Datenbank-Package „pkg_customer“ aufruft (siehe Abbildung 2). Die Prozedur „get_customer_information“ liefert Kundendaten anhand der Kunden-ID.

Daneben wurde ein Plain old Java Object (POJO) angelegt, um die Struktur der Daten vorzugeben. Mit diesen zwei simplen Klassen sind an und für sich schon alle Voraussetzungen gegeben, um einen Java-basierten RESTful-Webservice zu generieren. Mit einem Rechtsklick auf „CustomerInformation.java“ lässt sich der Kontextmenüeintrag „Create RESTful Service“ auswählen.

Es öffnet sich ein weiteres Fenster, in dem verschiedene Optionen ausgewählt werden können; unter anderem, welche HTTP-Operationen der Webservice bereitstellt (GET, POST, PUT und DELETE) und welche Form der MIME-Type für den Dateninhalt hat. In diesem Fall wurde die Option „application/json“ unter „Produces“ ausgewählt, um die Daten in Form von JSON zu strukturieren.

Im unteren Teil des Fensters lassen sich gegebenenfalls Parameter definieren. Es wurde ein Query-Parameter hinzugefügt, um die entsprechenden Informationen für die übergebene ID zurückgeben zu können. Wählt man „OK“ und schaut sich

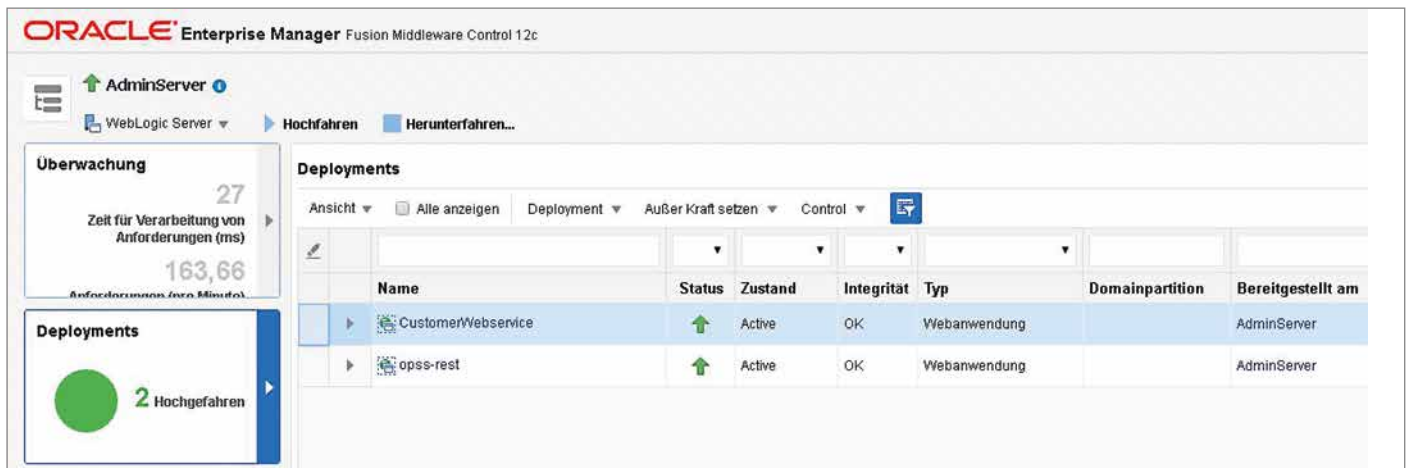


Abbildung 4: „CustomerWebservice“ auf WebLogic-Server eingerichtet

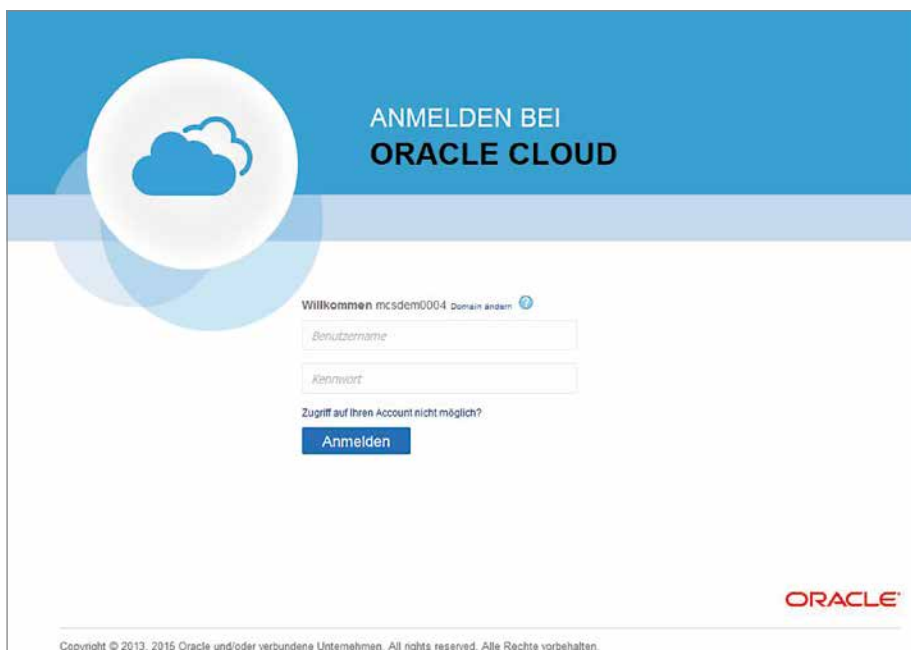


Abbildung 5: Login des Mobile Cloud Service

danach die Java-Klasse an, stellt man fest, dass JDeveloper automatisch alle notwendigen Annotationen in der Java-Klasse hinzugefügt hat (siehe Abbildung 3).

Nach der Generierung des Webservice lässt sich dieser direkt aus dem JDeveloper heraus testen. Dafür kann mit einem erneuten Rechtsklick auf die Java-Klasse „CustomerInformation“ geklickt und „Test Webservice“ ausgewählt werden. Der Webservice wird dann auf dem internen WebLogic-Server eingerichtet und anschließend der HTTP-Tester des JDeveloper geöffnet (siehe Abbildung 4). Im HTTP-Tester lassen sich unter „Parameter“ die Übergabeparameter einstellen und mit einem Klick auf „Send Request“ erfolgt ein Test des Webservice.

Ist der Test erfolgreich, liefert er den HTTP-Status „200“ und die Kunden-Informationen zurück. Jetzt kann der Webservice auf einem eigenständigen Webserver eingerichtet werden. In diesem Fall kommt der Oracle WebLogic Server in der Version 12.2.1.0.0 zum Einsatz. Dies kann direkt aus dem JDeveloper heraus geschehen, es muss hierfür lediglich ein Deployment-Profil angelegt und der entsprechende WebLogic-Server ausgewählt werden. Nachdem der Webservice auf dem Server eingerichtet worden ist, kann er von außen angesprochen werden, die PL/SQL-Logik ist jetzt erreichbar.

Zum Testen von Webservices gibt es neben dem HTTP-Tester des JDeveloper unzählige kostenfreie Werkzeuge. Exem-

plarisch sei hier die Browser-Erweiterung Postman für Google Chrome erwähnt.

Einbindung des Webservice Oracle MCS

Wenn der Test des Webservice erfolgreich war, lässt er sich in den Oracle Mobile Cloud Service einbinden. Hierfür muss man sich an der entsprechenden Domain des MCS mit den Zugangsdaten anmelden (siehe Abbildung 5). Nach der Anmeldung wird ein Dashboard angezeigt, das dem Benutzer die abonnierten Cloud-Services anzeigt – mit einem Klick auf einen Service lassen sich Details wie der Service-Status einsehen.

Die Auswahl des assoziierten Service öffnet auf der nächsten Seite die Service-Konsole, in der die einzelnen Komponenten des MCS verwaltet werden (siehe Abbildung 6). In der Übersicht wird unter „Erste Schritte“ die App-Entwicklung mithilfe des MCS in aller Kürze erläutert. Um den generierten Webservice für die App bereitzustellen, ist zunächst ein sogenanntes „Mobile Backend“ zu erzeugen. Es dient der Server-seitigen Verwaltung der App.

Erstellen eines Connectors

Nachdem das Mobile Backend angelegt ist, können wir einen Connector einrichten. Dieser bietet die Möglichkeit, Backend-Systeme einzubinden, so wie in dem hier vorliegenden Fall, oder aber auch externe Webservices wie Google Webservices. Ein Connector kann dann später in einem sogenannten „Custom-

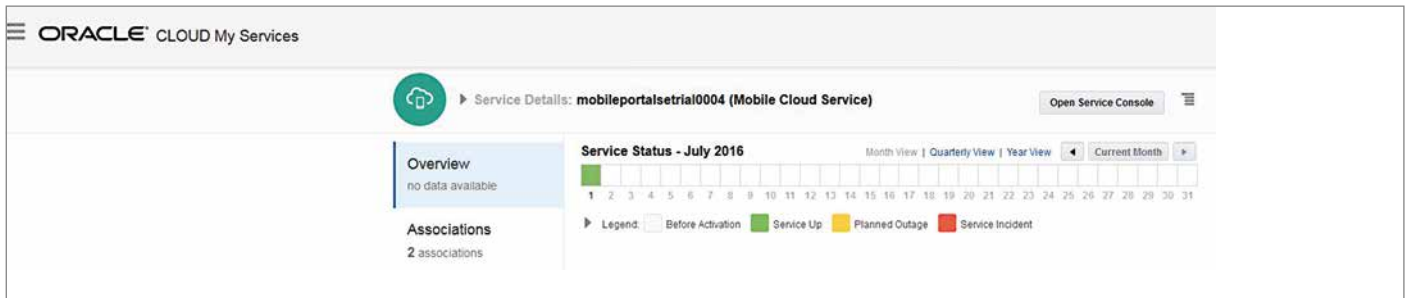


Abbildung 6: Service Console öffnen

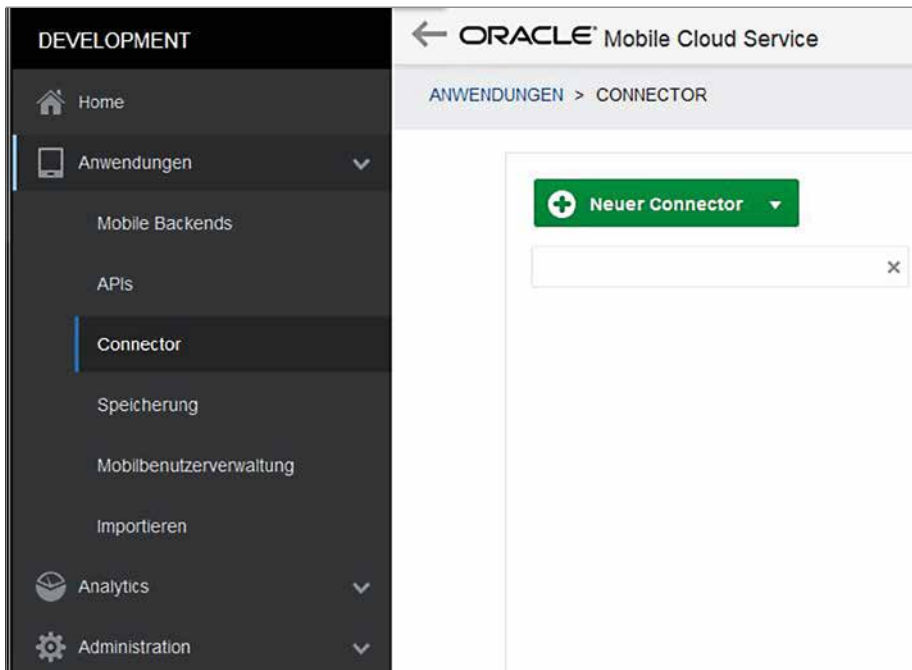


Abbildung 7: Erstellen eines neuen Connectors

Aufruf dieser erweitert und entsprechende HTTP-Methoden zugewiesen werden.

Fazit

Egal, ob selbstentwickelte Java-Klasse im JDeveloper oder ORDS – viele Wege führen zum Ziel, PL/SQL-Logik auf der Datenbank für andere Anwendungen oder mobile Apps verfügbar zu machen. Mit den richtigen Werkzeugen und ein wenig Know-how lässt sich dies bereits umsetzen. Mit dem Mobile Cloud Service bietet Oracle darüber hinaus die Möglichkeit, APIs basierend auf Webservices in eine App mit einzubinden, was dem App-Entwickler eine Vielzahl von Aufgaben abnimmt. Aber auch ohne MCS lassen sich solche Services gut in mobile Apps integrieren. Erwähnt seien an dieser Stelle das Oracle Mobile Application Framework (MAF) und Apex Mobile.

API“ verwendet werden. Ein Klick auf „Neuer Connector“ zeigt eine Auswahl an möglichen Connectoren an: REST, SOAP und Integration Cloud Service.

Nach der Auswahl des zu erzeugenden Service wird eine Eingabemaske geöffnet. Hier müssen folgende Pflichtfelder gefüllt werden: API-Anzeigename, API-Name, Remote-URL und eine Beschreibung. Nachdem diese Felder ausgefüllt sind, lässt sich das API konfigurieren und testen. In den allgemeinen Einstellungen können die Namen für das API nochmals überarbeitet werden. Ebenfalls kann hier ein HTTP-Lese- und Verbindungs-Time-out gesetzt werden.

Im folgenden Menüpunkt „Regeln“ lassen sich Standard-Parameter und die HTTP-Methoden für den Connector vorkonfigurieren. Unter dem Menüpunkt „Sicherheit“ lassen sich verschiedene Sicherheits-Policies wie zum

Beispiel die Authentifizierung über SSL durchsetzen.

Zu guter Letzt lässt sich der Connector testen. Im vorliegenden Fall soll getestet werden, ob für den Kunden mit der ID=„203“ Daten vorliegen. An die Service-URL wird hierfür lediglich „?id=203“ angehängt und auf „Test-Endpunkt“ geklickt. Oracle MCS führt die Anforderung aus und zeigt im unteren Teil der Webseite das Ergebnis an (siehe Abbildung 7).

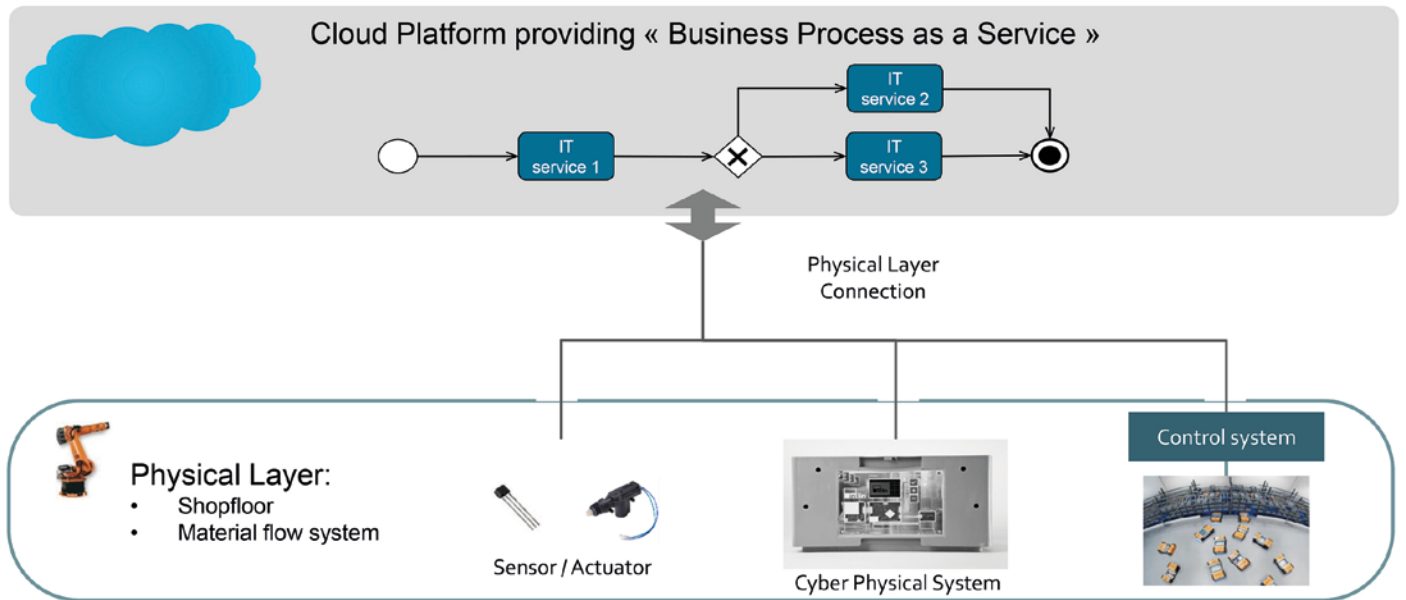
Nach dem Einbinden des Webservice in den MCS ist es möglich, diesen in der weiteren Entwicklung der App aufzurufen. Der hier exemplarisch eingebundene Webservice für die Abfrage von Kunden-Informationen könnte je nach Anforderung weitere Methoden für das Abfragen aller Kunden oder das Anlegen neuer Kunden erhalten. Sind bereits entsprechende Programm-Einheiten in der Datenbank vorhanden, müssen lediglich der Webservice um den

Weiterführender Link

- Oracle MCS Tutorial: http://www.oracle.com/webfolder/technetwork/tutorials/cloud/ocms/OCMS_MBE_OBE_tutorial.html



Daniel Kampf
dkampf@pitss.com



Von Big Data bis IDS – Informationslogistik im Internet der Dinge

Jens Leveling und Christian Prasse, Fraunhofer-Institut für Materialfluss und Logistik (IML)

Wenn man heute von Logistik spricht, muss man neben der klassischen Bewegung von Gütern und Menschen vor allem den Austausch von Daten betrachten. Dieser Artikel zeigt die neuen Konzepte und Methoden im Rahmen der Informationslogistik. Schwerpunkte sind der Umgang mit großen Datenmengen und die horizontale Kooperation über Unternehmensgrenzen hinaus.

Die Welt hat sich geändert und ändert sich noch – immer schneller. Die klassische Rollenenteilung zwischen Produzent, Händler und Logistiker verschmilzt zunehmend. Unter anderem führen steigende Variantenvielfalt und e-Commerce zu einer nie dagewesenen Komplexität in Produktions- und Logistik-Systemen. Hochdynamische Märkte und sich ebenso schnell ändernde Kundenanforderungen verstärken diesen Effekt. Die Zukunft der Produktion und des Handels ist geprägt von agilen Wertschöpfungs-Netzwerken mit einer steigenden Anzahl von Teilnehmern beziehungsweise einer hohen Dynamik bei der Änderung der Player. Um die genannte Komplexität und die Dynamik handhabbar zu machen, sind moderne Hilfsmittel im Sinne einer Informationslogistik und die Dezentralisierung

von Steuerungsprozessen notwendig. Dieser Artikel gibt – wenn auch nicht vollumfänglich – einen Überblick über einige neue Entwicklungen.

Industrie 4.0 und das Internet der Dinge in der Logistik

Es gibt zahlreiche Definitionen von Industrie 4.0 – der vierten industriellen Revolution. Genauso zahlreich sind auch die Interpretationen der Ziele und des Nutzens. Die Autonomie beziehungsweise Selbststeuerung ist das wesentliche Grundprinzip der vierten industriellen Revolution und die wesentliche Grundlage für das Management wachsender Komplexität und Dynamik. Auslöser und Lösungsprinzip

der vierten industriellen Revolution ist der Ansatz des Internets der Dinge und Dienste. Das Internet und Kleinstrechner, die aufgrund ihres geringen Preises in Dinge des täglichen Lebens eingebaut werden können, kommunizieren miteinander. Technologische Basis für diese „denkenden“ Dinge und Maschinen sind Cyber-physische Systeme, mit deren Hilfe logistische Objekte wie Pakete, Behälter, Paletten und Container ihren Weg alleine durch inner- und außerbetriebliche Netze finden, ähnlich wie die Datenpakete im Internet. Durch die ganzheitliche Verschmelzung der physischen und der virtuellen Welt entsteht damit ein Internet der Dinge und Dienste.

Dies wird zu einem nachhaltigen und disruptiven Wandel der intralogistischen



Abbildung 1: Alles wird autonom – Beispiele für lokale Entscheidungen und autonomes Handeln in der Anwendungsdomäne „Logistik“

Materialfluss-Systeme und logistischen Netzwerke führen. Vollziehen wird sich dieser Wandel als Evolution vom reinen Monitoring der Gutbewegungen und der zentralen Steuerung und Optimierung hin zu einem selbst-organisierenden Materialfluss durch autonomes Verhalten der logistischen Objekte entlang der Supply Chain. Einige Beispiele für diese logistischen Objekte und Gewerke, die sich aufgrund von lokalen Entscheidungen selbst organisieren, sind in *Abbildung 1* dargestellt. Regale, Behälter, Container und Fahrzeuge kommunizieren über kabellose Schnittstellen und Software-Agenten miteinander und mit externen Systemen, treffen lokale Entscheidungen und handeln dadurch autonom ohne zentrale Instanz.

Daten sind in Cloud-Speichern verfügbar und ermöglichen einen Zugriff von verschiedenen Teilnehmern. Unersetzliches Element in effizienten Logistiksystemen ist jedoch weiterhin der Mensch, er trifft strategische Entscheidungen, plant und steuert Prozesse. Dazu muss er in anderer Weise in die neuartigen, sozio-technischen Systeme eingebunden sein.

Neben dem grundsätzlichen Steuerungsansatz sind nachfolgend Tools und Services im Rahmen der Informationslogistik vorgestellt. Wichtig in diesem Zusammenhang: Logistik und IT muss zusammengedacht werden.

Big Data

In der Logistik eröffnen sich mit der Verwertung großer Datenmengen neue Möglichkeiten. In den Unternehmen, in Liefer-Netzwerken und im Internet existieren viele verschiedene Datenquellen, die zur Entscheidungshilfe sowie zur Erstellung von Prognosen herangezogen werden können und auf diese Weise einen Mehrwert für die Unternehmen generieren. Hierbei spielen Big-Data-Technologien und Daten-Auswertungen eine entscheidende Rolle (*siehe Abbildung 2*).

Diese Technologien sind vor allem für die Auswertung und Nutzung von Sensor- und AutoID-Daten (Daten von RFID- und Barcode-Lesegeräten) interessant. Vor allem letztere werden immer häufiger Standort- und Unternehmens-übergreifend erfasst und zentral zusammengeführt. Für diese Daten stellt sich im Zuge der Integration die Frage, inwieweit sie vorab verdichtet oder als Rohdaten in einen zentralen Datenspeicher – eine No-SQL-Datenbank oder ein verteiltes Dateisystem – überführt und dort für Analysen und Auswertungen vorgehalten werden. Verdichtung kann zum Beispiel die Berechnung von Durchschnittswerten über einen bestimmten Zeitraum oder die Erzeugung von Events bedeuten, die Zusammenfassungen bestimmter Vorgänge enthalten.

Eine Lösung für entsprechende Fragestellungen ist stark vom jeweiligen Anwendungsfall abhängig. Für eine Governance von Logistik-Infrastrukturen, also der Verwaltung und des Monitorings von IT-Systemen, Fahrzeugen und Materialfluss-Anlagen, reichen verdichtete Sensordaten aus. Für eine detaillierte Analyse von Fehlern oder Unterbrechungen der Auftragsbearbeitung ist jedoch eine Betrachtung von Rohdaten interessant.

Ein Big-Data-Problem ist die Betrachtung von vielen unterschiedlichen Sensoren und AutoID-Readern mit vielen Konfigurationsmöglichkeiten sowie deren Geschwindigkeit beim Erzeugen von Events – in der Regel von mehreren Hertz. Diese Daten sind zusammen mit Auftrags-, Produkt- und Transaktionsdaten aus den Warehouse-Management- (WM) und ERP-Systemen sowie mit weiteren Statistikdaten über zurückliegende Vorgänge zu verarbeiten und auszuwerten.

Für Anwendungsfälle, die nicht nur Wartung und Instandhaltung, sondern auch die Planungen von Anpassungen und Erweiterungen der Anlagen bis hin zur Entwicklung von neuen Logistik-Dienstleistungen betrachten, ist die Nutzung aller dieser Daten von großem Interesse. Der Mehrwert ist im Besonderen die Verbesserung der Sichtbarkeit über den Material- und Informationsfluss. Engpässe in Vorgängen und Materialbewegungen können



einfacher aufgedeckt und Transporte neu und besser geplant werden.

Cloud

Die Wertschöpfung verlagert sich zunehmend von Kernaufgaben hin zu höherwertigen, IT-gestützten Dienstleistungen, sogenannten „Smart Services“. Wettbewerbsfähigkeit zeichnet sich durch eine schnelle Konzeption und Umsetzung sowie durch flexible Gestaltung dieser Dienstleistungen aus. Die Technologie des Cloud Computing erlaubt die Bereitstellung und Nutzung verschiedener Dienste nach Bedarf.

Für die meisten Anwender ist ein unkomplizierter und schneller Bezug aller zur Ausführung von individuellen Dienstleistungen benötigten Software-Kompo-

Abbildung 2: Big-Data-Technologien ermöglichen eine Komplexitätsreduzierung für den Anwender

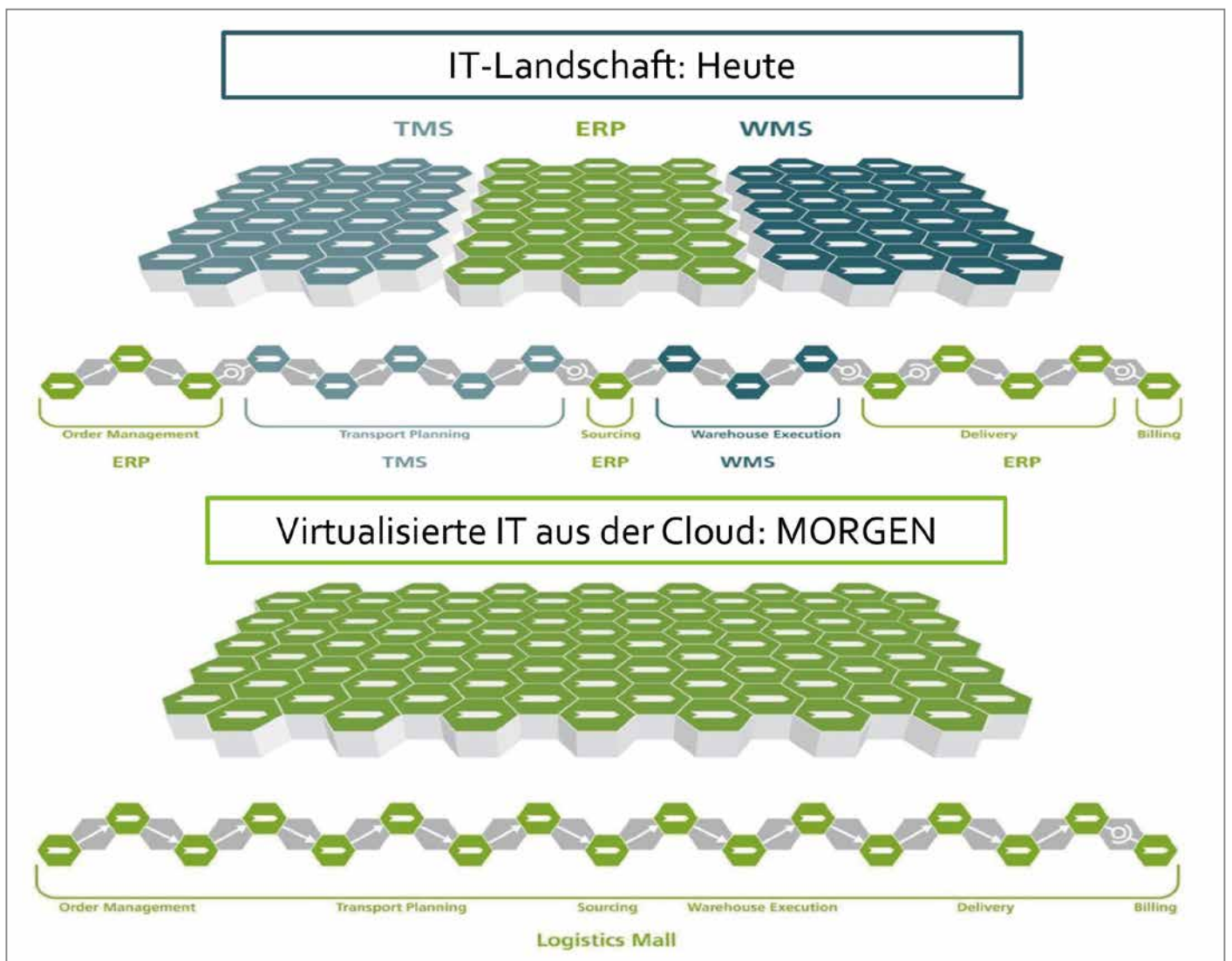


Abbildung 3: Cloud Computing – Daten austauschen und verfügbar machen

nenten (Anwendung, Datenbanken, mobile Schnittstellen, Datenkonverter etc.) aus der Cloud besonders interessant. Dies ermöglicht die in der Industrie häufig auftretenden, saisonbedingten Geschäftsprozessänderungen und erlaubt eine verbrauchsabhängige Abrechnung, die insbesondere bei den häufig vertretenen KMU (kleine und mittlere Unternehmen) Investitionsspitzen und -risiken minimiert.

Heutzutage ist zum Beispiel in einem typischen Logistikprozess eine Vielzahl unterschiedlicher IT-Systeme involviert, sodass eine heterogene Systemlandschaft zugrunde liegt. Bei den IT-Systemen selbst handelt es sich heute überwiegend um Standard-Software (WMS, ERP, PPS etc.), die den Anforderungen der Logistikanwender nach kurzen Vertragslaufzeiten, verbrauchsabhängiger Bezahlung und schneller Bereitstellung individueller IT-Funktionen nur unzureichend gerecht wird.

Die bereitgestellten Funktionen der Standardsoftware werden oftmals nicht im vollen Umfang benötigt, müssen

aber eingekauft werden. Individuell zugeschnittene Software bedingt dagegen in der Regel hohe Anpassungsaufwände auf Seiten des Anbieters und ist deshalb kostenintensiv. Die Flexibilität des Produkts im Hinblick auf spätere Anpassungen wird dabei nicht erhöht. Durch die Nutzung von Cloud-Umgebungen besteht die Möglichkeit, unterschiedliche IT-Dienste nach Bedarf zu nutzen und durch auf neue Anforderungen angepasste Systeme flexibel und dynamisch aus der Cloud abzulösen.

Voraussetzung für die Integrationsfähigkeit der im Rahmen der Prozessmodellierung zusammengestellten IT-Services ist ein gemeinsamer Standard, der die zwischen den IT-Services auszutauschenden Informationen syntaktisch und semantisch definiert. Nur so wird ein konsistenter Informationsaustausch zwischen IT-Services unterschiedlicher Anbieter möglich. Eine Implementierung von Konvertern oder Mapping-Werkzeugen ist nicht mehr notwendig und die

Austauschbarkeit von IT-Services damit gewährleistet (siehe Abbildung 3).

Industrial Data Space

Unternehmen können insbesondere durch den Austausch, die Verknüpfung und die Anreicherung von Daten über Wertschöpfungsketten und Branchengrenzen hinweg profitieren. In den letzten Jahren sind verschiedene Lösungsansätze für diese „Datenökonomie“ entstanden. Aber ein umfassendes Konzept, das sowohl technische als auch Sicherheits- und Datendienst-Aspekte oder auch Fragen der Zusammenarbeit zwischen den verschiedenen Akteuren adressiert, liegt bis heute nicht vor. Genau hier setzt der Industrial Data Space an. Er beschreibt einen virtuellen Datenraum und bürgt für die digitale Souveränität von Daten und Diensten. Zudem sichert er die digitale Identität aller Beteiligten. Alle Unternehmen, die sich an die gemeinsamen Spiel-

Cloud oder On-Premise? Mit Expertise ans Ziel.



DOAG Konferenz + Ausstellung
Treffen Sie uns auf dem Stand 242

Wann ist es Zeit für Cloud? Was bedeutet das für Ihre IT? Profitieren auch Sie vom umfassenden Know-how unserer Experten bei dieser Entscheidung. Erreichen Sie Ihre Ziele mit dbi services.

Phone +41 32 422 96 00 · Basel · Nyon · Zürich · dbi-services.com



Infrastructure at your Service.



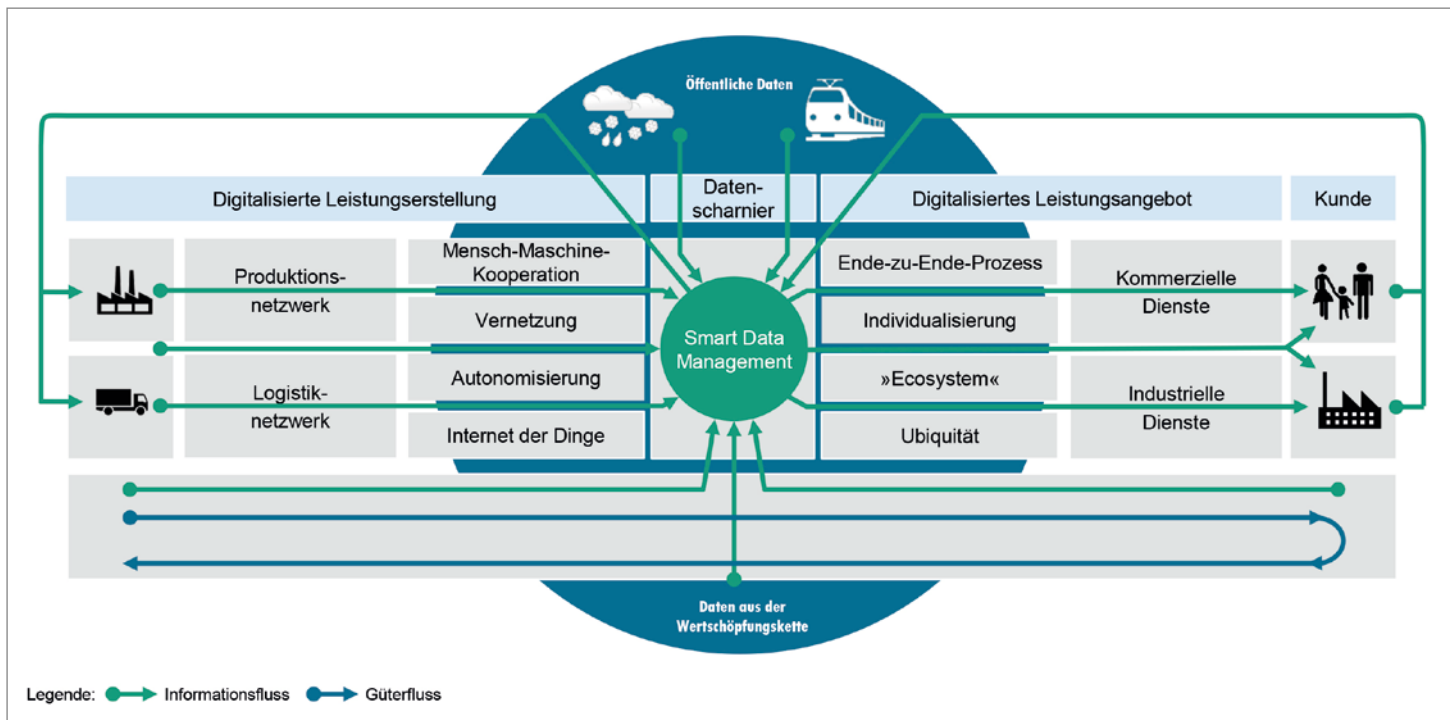


Abbildung 4: Industrial Data Space: Bindeglied zwischen digitaler Produktion/Logistik und Smart Services

regeln halten, können Daten sicher und vertrauensvoll anreichern, austauschen und verknüpfen.

Der Industrial Data Space stellt Datendienste bereit, die beispielsweise die Anonymisierung von Daten und Integrationsdiensten umfassen oder die Einstellung von Verfallsdaten für die Datennutzung realisieren. Auf diese Weise unterstützt er die Unternehmen bei ihrer digitalen Transformation und der Einführung und Weiterentwicklung durch die Industrie 4.0 (siehe Abbildung 4).

Der Industrial Data Space stellt einen ergänzenden Architektorentwurf dar. Bisherige Konzepte und Teillösungen werden durch die Initiative aufgegriffen und zu einem Ökosystem für den standardisierten und sicheren Datenaustausch zwischen Unternehmen zusammengeführt. Die bisherige Flexibilität im Internet bleibt dabei erhalten. Dies wird durch Mechanismen gefördert, die die Integrität und Verfügbarkeit der Dienste und Daten schützen. Der Daten-Eigentümer behält somit die komplette Souveränität über seine Daten und entscheidet eigenständig über deren Verbreitung und den Zugriff durch Partner. Bei Bedarf verbleiben die Daten beim Dateneigner und werden nur unter strengen Schutzmaßnahmen ausgetauscht.

Ziel des Industrial Data Space ist entsprechend die Integration bestehender Prozesse und Systeme. Die Innovation besteht im Nutzen für Anbieter und Verwender, indem bestehende IT-Infrastrukturen weiter genutzt und gleichzeitig sicher und souverän für Kollaboration mit Unternehmenspartnern im Rahmen von Wertschöpfungsprozessen einen Datenaustausch mit diesen standardisiert erlauben.

Der Nutzen lässt sich durch folgende Schlüsselmerkmale benennen: sichere Daten-Wertschöpfungskette, standardisierter Datenaustausch zwischen Unternehmen, Referenz-Architekturmodell und ein offener Entwicklungsprozess.

Fazit

Die Beherrschung der Komplexität und die Schaffung von Transparenz entlang der Supply Chain sind wesentliche Herausforderungen, die die gesamte Wirtschaft betreffen. Es wird entscheidend sein, die notwendigen Soft- und Hardware-Entwicklungen als vertikale und horizontale Lösungen voranzutreiben. Alle genannten Entwicklungen befähigen Unternehmen, die Herausforderungen zukünftiger Produktions- und Logistiksysteme – nämlich hochdynamischer

agiler Wertschöpfungs-Netzwerke – zu bestehen. Über die breite Anwendung in der Logistik und Produktion müssen sich neue Technologien und Methoden bewähren und etablieren. Das Fraunhofer IML unterstützt seine Partner dabei.



Jens Leveling
jens.leveling@iml.fraunhofer.de



Christian Prasse
christian.prasse@iml.fraunhofer.de



Industrie 4.0: Analytisch, Prozess-orientiert und transformativ

Marcel Amende, ORACLE Deutschland B.V. & Co. KG, und Thorsten Schulz, Bosch Rexroth AG

„Unsere Maschine hört sich irgendwie merkwürdig an.“ So oder ähnlich beginnen viele Anrufe von Kunden bei unserer Service-Hotline, weiß Thorsten Schulz als Produkt Manager bei Bosch Rexroth zu berichten, einem führenden Hersteller von Industrie-Steuerungen. Um das Problem näher zu analysieren, war bisher der Vor-Ort-Einsatz eines Servicetechnikers erforderlich.

Im Industrie-4.0-Zeitalter und mithilfe des Oracle Internet of Things Cloud Service, dem Plattformdienst für die Einbindung von Sensoren und Geräten aller Art in die IT eines Unternehmens, lassen sich Problem-Analysen sofort und aus der Entfernung starten. Direkt aus der CRM Cloud können zusätzliche analytische Softwarepakete über die IoT-Cloud in die Steuerung übertragen und während des laufenden Betriebes der Anlage gestartet werden. Im vorliegenden Fall kann dies eine Oszilloskop-Software sein, die eine Achsbewegung einer an die Bosch Rexroth Steuerung angeschlossenen Maschine im Detail aufzeichnet. Eine Unwucht oder Beschädigung ist so aus der Entfernung zu erken-

nen und zu bewerten. Der Einsatz eines Servicetechnikers und der Bedarf an Ersatzteilen kann vorab gut geplant werden, sollte er nötig sein.

Die analytische, ingenieurmäßige Betrachtung von Maschinen-, Geräte- und Sensordaten ist nur der erste Schritt bei einer strategischen Umsetzung von IoT-beziehungsweise Industrie-4.0-Projekten. Fachanwender werden vor allem die Unternehmensprozesse und bestehende Anwendungslandschaft im Blick haben. Die Unternehmensführung ist an innovativen und bestenfalls transformativen Geschäftsmodellen und Diensten interessiert, die neuen Umsatz bringen und das Unternehmen in das Digitalzeitalter führen.

Schritt 1: Analytische Industrie 4.0

Spricht man mit Ingenieuren über das Thema „Industrie 4.0“, steht meist der Wunsch im Vordergrund, ein möglichst umfassendes und detailliertes Wissen um die Funktion von Maschinen und Anlagen zu gewinnen. Er resultiert aus der primären Anforderung, eine hohe Gesamtverfügbarkeit der Anlage und eine bestmögliche Produktionsqualität zu gewährleisten.

Das Herz einer komplexen Industrieanlage ist die Industriesteuerung. Sie kommuniziert über moderne industrielle Feldbus-Systeme wie Profinet oder Sercos mit den elektrischen und hyd-

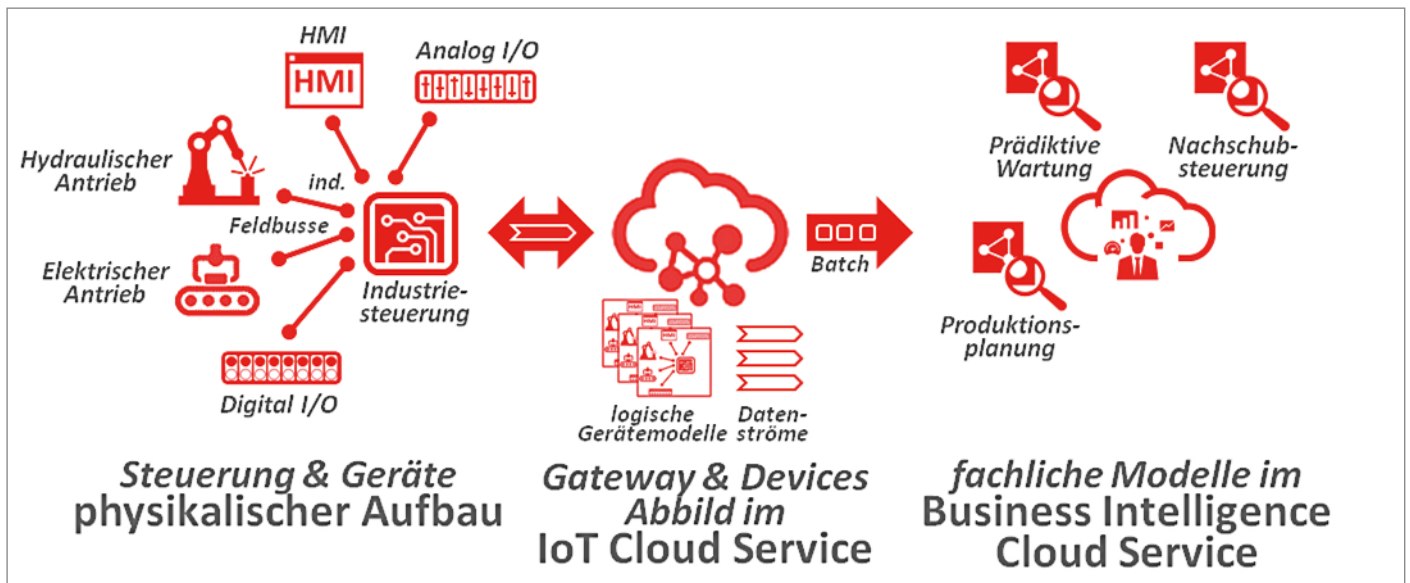


Abbildung 1: Maschinen-Aufbau, logisches Modell in IoT Cloud Service und analytische Modelle in Business Intelligence Cloud Service

raulischen Antrieben, Getrieben, Ventilen und Zylindern, steuert Roboter und erfasst Sensordaten. Die Besonderheit der Bosch Rexroth MLC Steuerungen ist, dass sie neben der industriespezifischen SPS-Programmierung mit ihrem Konzept des Open Core Engineering [1] ein zweites Ökosystem in Form einer Java-8-Embedded-Laufzeitumgebung bereitstellen. Dieses ermöglicht der breiten Java-Entwicklergemeinschaft leichten Zugang zu Industrie-Anwendungen.

In Kombination mit einem speziellen Plattformdienst (Platform-as-a-Service, PaaS) für die Gerätevernetzung wie dem Oracle IoT Cloud Service lassen sich Industrie-4.0-Anwendungen einfach, schnell und risikofrei erproben und umsetzen. Ohne langfristige Investitions- und Wartungskosten sowie Installations- und Konfigurationsaufwände kann der Cloud Service zunächst kostenlos erprobt und bei Erfolg flexibel und skalierbar als Subskription bezogen werden [2].

Mithilfe der quelloffenen Java-Client-Bibliotheken [3] ist die sichere Anbindung der Steuerung an den Oracle Internet of Things Cloud Service eine einfache Aufgabe: Die Steuerung ist im Cloud Service als Gateway abgebildet, das die bidirektionale Kommunikationsbrücke zwischen den Industrie-Bussen und dem Cloud Service darstellt. Die an die Steuerung angeschlossenen Maschinen und Geräte sind als virtuelle Devices repräsentiert.

Die Steuerung hat Zugriff auf alle Daten, die über die Industrie-Busse fließen,

und kann diese an die IoT-Cloud kommunizieren. Dafür sind dort Modelle für die verschiedenen Gerätetypen (analoge I/O, digitale I/O, elektrische Antriebe, hydraulische Antriebe, HMI-Anzeigen) vorkonfiguriert, die die Formate und Attribute von Nachrichten, Alarmmeldungen und im Gerät ausführbare Aktionen beschreiben. Im Cloud Service entsteht so ein logisches Abbild des physikalischen Aufbaus der Maschinen in der Fabrikhalle (siehe Abbildung 1).

Für eine fachliche Aufbereitung und Analyse der Maschinendaten ist entscheidend, ob die zu beantwortenden Fragestellungen zum Zeitpunkt der Datenerfassung bereits bekannt sind oder im Nachhinein auf Basis der Rohdaten immer neue Aspekte beleuchtet werden sollen. Im Falle bekannter Fragestellungen bieten sich klassische Business-Intelligence-Werkzeuge an. Besonders einfach ist die Nutzung des Oracle BI Cloud Service, der mit dem Oracle IoT Cloud Service vorintegriert ist. Im IoT Service können Datenströme für die Batch-Übertragung ausgewählt werden. Die passenden Bereitstellungstabellen werden im BI Cloud Service automatisch angelegt. Unter Hinzunahme weiterer externer Datenquellen können daraus auswertungsorientierte Datenmodelle konfiguriert werden, meist in Form eines Stern-Schemas.

Ein typischer Anwendungsfall ist die prädiktive Wartung für die gezielte Überwachung, Optimierung und Leistungssteigerung technischer Anlagegüter. Ausfälle von Komponenten sollen präzise

vorausgesagt werden, um die Gesamtverfügbarkeit von Anlagen und Maschinen, ihre Auslastung und Lebensdauer zu erhöhen. Dafür wird gezielt nach Anomalien gesucht, indem etwa das Verhalten einzelner Komponenten mit einer Referenz verglichen wird. Im Fehlerfall erfolgt eine Ursachen-Analyse und der Einfluss von Umgebungsparametern wird untersucht, um ähnliche Fehlerfälle in Zukunft zu vermeiden. Aus unternehmerischer Perspektive kann man auf diesem Wege Wartungskosten sparen, die Produktionsqualität und Kundenzufriedenheit erhöhen oder Rückrufaktionen limitieren.

Aus Sicht der BI-Datenmodellierung sind Temperaturen, Drücke, Vibrationen, Leistungsaufnahme, Produktionszähler und Betriebsstunden wichtige Kennzahlen. Die Dimensionen, nach denen in diesem Fall eine Daten-Aggregation erfolgen kann, sind (hierarchisch) Produkt-Gruppe, -Typ, -Version und -Komponente, (hierarchisch) Branche und Kunde, Einsatzort, Verwendungszweck und Zeit. Auf diesem Modell lassen sich flexibel KPIs und Abfragen definieren sowie in Berichten und Dashboards visualisieren. Auf Basis derselben Maschinendaten lassen sich ebenso gezielte Aufbereitungen für das Energiemanagement, die Produktions- oder Nachschubsteuerung bewerkstelligen.

Ist zum Zeitpunkt der Datenerhebung unklar, welche Fragestellungen man zu späteren Zeitpunkten beantworten will, bieten sich Big-Data-Verfahren an. Diese nutzen möglichst große und günstige

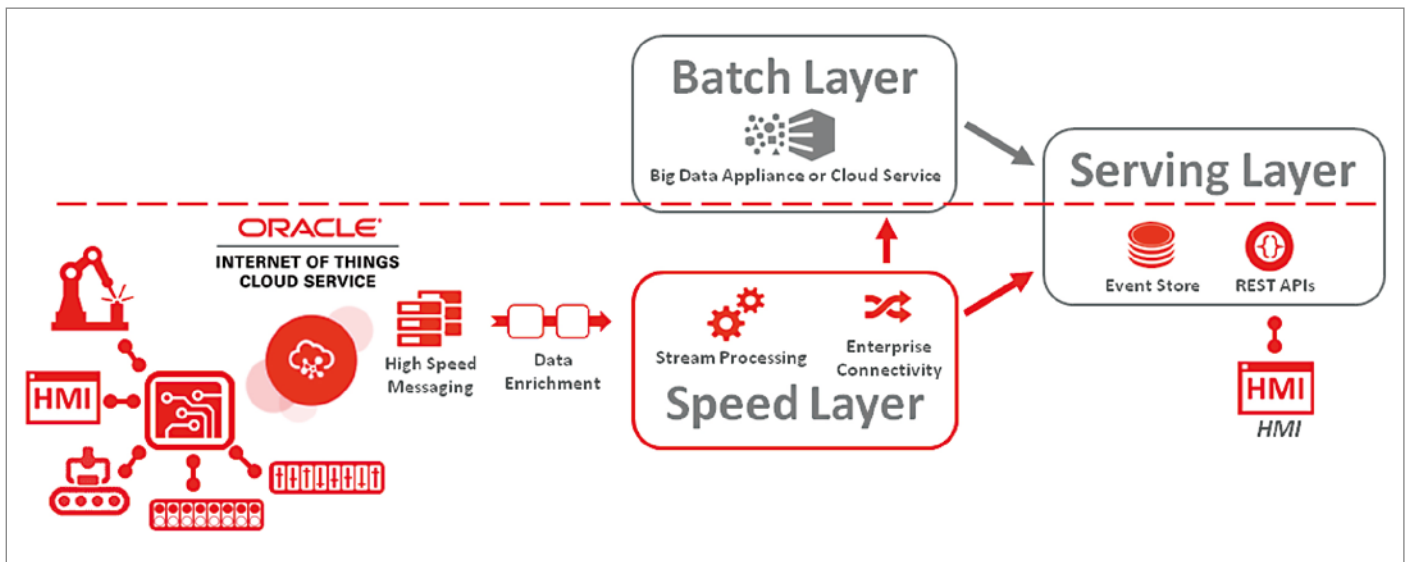


Abbildung 2: Lambda-Architektur für Echtzeit-Analysen im Big-Data-Umfeld

Festplatten mit dem verteilten Hadoop-Dateisystem, um Massendaten unstrukturiert und in großen Blöcken zu speichern. So kann man es sich im wahrsten Wortsinne leisten, Massendaten mit all ihrer Redundanz langfristig aufzubewahren.

Kommt es bei einer Maschine zu einem unerwarteten Defekt, werden die Zustände und Betriebsparameter der Maschine kurz vor Auftreten des Defekts im Datenbestand betrachtet. Findet man in den Daten Hinweise auf mögliche Fehlerursachen, kann man in weiteren Analyseläufen proaktiv nach Maschinen suchen, bei denen ähnliche Defekte drohen, und diese durch gezielte Rückrufe oder Serviceeinsätze verhindern.

Ein Manko der Big-Data-Verarbeitung sind die Latenzzeiten: Es benötigt einige Zeit, bis die Datenblöcke geschrieben und im Batch verarbeitet werden können. Man ist aus unternehmerischer Sicht für einige Minuten blind und kann nicht schnell reagieren. Die Erkenntnis, dass eine Maschine bei zu hoher Drehzahl überhitzt, kann nach wenigen Sekunden bereits zu spät kommen, um korrigierend eingreifen zu können. Daher ergänzt man die Big-Data-Batch-Verarbeitung im Rahmen der Lambda-Architektur üblicherweise durch einen sogenannten „Speed Layer“ (siehe Abbildung 2). Dort wird eine Echtzeitverarbeitung in Form einer Muster-Erkennung auf den Eingangsdatenströmen vorgenommen, um schnell auf besondere Vorkommnisse reagieren zu können. Diese Form der Datenstrom-Analyse, wie auch die Speicherung der Analyse-Ergebnisse in einer No-

SQL-Datenbank und Bereitstellung über ein REST-API, ist bereits integraler Bestandteil des Oracle IoT Cloud Service.

Schritt 2: Prozess-orientierte Industrie 4.0

Fachanwender haben vor allem die Prozesse eines Unternehmens im Blick. Sie sind an größtmöglicher Transparenz interessiert und möchten jederzeit wissen, was aktuell im Unternehmen passiert. Die eingesetzten Applikationen bieten oft nur Einblick in einen bestimmten Ausschnitt des Unternehmens: den Auftragseingang, die Produktionsplanung oder die Lagerhaltung.

Bei fehlender Vernetzung von Maschinen, Anlagen und Transportmitteln ist die Fertigung oder Logistik aus Prozesssicht ein blinder Fleck. Wird die Vernetzung der Betriebsmittel mit der IT im Sinne der Industrie 4.0 hergestellt, können die zusätzlich erhobenen Daten den Wert der vorhandenen Applikationen erheblich steigern: Durch den Einblick in die aktuelle Produktion und die Auslastung der Maschinen kann eine bessere Produktionsplanung erfolgen. Werden Fahrzeuge in Echtzeit verfolgt, kann die Transportplanung optimiert werden. Ein genauer Einblick in Bestände und Bedarfe bietet die Möglichkeit, kapitalbindende Überbevorratung zu vermeiden.

Der Oracle IoT Cloud Service schlägt die Brücke zwischen Werkshalle und IT. Im Cloud Service lassen sich gezielt Daten-

Weiterleitungen konfigurieren. Die von den Industrie-Steuerungen gelieferten Daten können ungefiltert oder als Ergebnis einer Datenstrom-Analyse an beliebige Applikationen und Systeme übertragen werden. Über das REST-API des Cloud Service können Applikation auch bi-direktional mit den Geräten, Maschinen, Anlagen und Sensoren kommunizieren oder Nachrichten aus dem Zwischenspeicher des Cloud Service abfragen.

Schritt 3: Transformative Industrie 4.0

Das Internet, insbesondere die mobile Internetnutzung, hat das Nutzungserlebnis für Konsumenten nachhaltig verändert. Heute wird erwartet, immer und überall Zugriff auf Informationen und Dienste zu haben und diese als Self-Service nutzen zu können. Diese Erwartungshaltung lässt klassische Märkte verschwinden und neue entstehen. Unternehmen und ganze Branchen sind dadurch einem riesigen Veränderungsdruck ausgesetzt. Kaum jemand ist heute noch bereit, Ware aus einem Katalog telefonisch oder postalisch zu bestellen. Online-Plattformen und Shops sind der Standard, eine Anzeige der Verfügbarkeit und Lieferzeit wird erwartet, ebenso die Online-Verfolgung der Lieferung nahezu in Echtzeit. Unternehmen, die diese Erwartung nicht erfüllen, werden vom Markt verdrängt.

Kluge Strategen und Unternehmensführer sind daher immer auf der Suche

nach neuen, innovativen Ideen und Geschäftsmodellen, die dem Unternehmen die Zukunft sichern. Im Idealfall sind diese so transformativ, dass sie den Markt nachhaltig verändern oder komplett neue Märkte definieren. Jede Veränderung ist aber auch mit Investitionen und Risiken verbunden. Cloud-Applikationen (Software-as-a-Service, SaaS) und -Dienste (Platform-as-a-Service, PaaS) eignen sich daher ideal als Innovationsplattformen. Sie stehen auf Knopfdruck zur Verfügung und werden entsprechend ihrer Nutzung abgerechnet. Man vermeidet hohe Investitions- und Anlaufkosten für den Fall, dass ein neues Geschäftsmodell oder ein neuer Dienst die gesteckten Erwartungen nicht erfüllen.

Der deutschen Industrie fehlt oft eine gewisse Startup-Mentalität und Innovationsbereitschaft. Die großen, transformativen Ideen scheinen eher dem amerikanischen Silicon Valley zu entspringen. Forschungen an der Universität St. Gallen haben aber überraschenderweise ergeben, „... dass mehr als 90 Prozent aller Geschäftsmodell-Innovationen lediglich Rekombinationen aus bekannten Ideen, Konzepten und Elementen von Geschäftsmodellen aus anderen Industrien darstellen“ [4]. Innovation lässt sich demnach viel methodischer angehen, als weithin vermutet. Die in der Publikation „Geschäftsmodelle entwickeln“ [4] beschriebenen 55 Muster für Geschäftsmodellinnovation verweisen auf Erfahrungswerte in den verschiedensten Branchen. Für einige ist die direkte Relevanz einer Vernetzung im Sinne der Industrie 4.0 erkennbar: „Das ... Unternehmen Flyeralarm wendete das E-Commerce-Muster erfolgreich an und entwickelte sich zu einer der größten Online-Druckereien in Europa“ [4]. Die Druckbranche ist durch den Trend zur Digitalisierung massiv bedroht, da der Bedarf an Druck-Erzeugnissen generell rückläufig ist. „Durch die fast komplette Automatisierung der Betriebsprozesse kann Flyeralarm schnelle und kostengünstige Abwicklungen von Plakat- und Flyer-Druckaufträgen garantieren“ [4]. Man nutzt den technologischen Wandel aktiv, um durch intelligente Vernetzung die Kluft zwischen Web, IT und Fertigung zu schließen und sich von den Konkurrenten abzuheben.

Ebenfalls interessant für die Industrie sind Geschäftsmodelle, die darauf ba-

sieren, dem Kunden einen einzigartigen Service zu bieten. Garantierte Verfügbarkeiten oder schneller Ersatz bei Defekt sind gute Beispiele. Die Möglichkeit der kontinuierlichen Überwachung und der prädiktiven Wartung von Anlagegütern durch Vernetzung im Sinne der Industrie 4.0 bildet hier die Grundlage.

Denkbar ist auch, für industrielle Geräte, Maschinen und Anlagen ein flexibles, nutzungsbezogenes Abrechnungsmodell zu etablieren. Dies ist interessant, wenn Kunden ansonsten vor zu hohen Investitionskosten zurückschrecken. Teure Maschinen können Kunden sogar kostenlos zur Verfügung gestellt werden, wenn ihre Vernetzung dem Hersteller laufend Einblick in die Nutzung gewährt. Die Refinanzierung kann dann über das benötigte Verbrauchsmaterial oder die fortlaufende Wartung sichergestellt werden, wenn eine Mindestnutzung vereinbart ist. Ein Beispiel hierfür ist ein Röntgenapparat, für den das Krankenhaus das Fotomaterial regelmäßig vom Hersteller bezieht und die Fallkosten mit der Krankenkasse abrechnet.

Ein weiterer Trend in der Industrie ist die kundenspezifische Fertigung. Diese wird durch immer flexiblere Fertigungsmaschinen und 3D-Drucker möglich. Artikel, die von Kunden online konfiguriert und bestellt werden, können direkt in die Fertigung überführt werden. Selten benötigte Ersatzteile müssen nicht mehr vorgehalten, sondern können bei Bedarf gefertigt werden. Ein Kunde kann zum Designer seines individuellen Turnschuhs werden.

Fazit

Dem Thema „Industrie 4.0“ kann man sich aus vielen Perspektiven nähern. Ein Ingenieur wird eine eher technische, die Unternehmensführung eine eher strategische Sicht einnehmen. Nutzen lässt sich in allen Fällen erschließen. Mal stehen Kostenersparnis und Qualitätssteigerung im Vordergrund, in anderen Szenarien Umsatzsteigerung und Innovationskraft. Als Hemmnis stehen dem die Angst vor der Komplexität und die Scheu vor Investitionen gegenüber.

In beiden Fällen ist die Kombination der Megatrends IoT/Industrie 4.0 und Cloud Computing die Lösung. Mit dem

Oracle IoT Cloud Service steht eine hochspezialisierte Plattform zur Verfügung, die sämtliche Grundfunktionalitäten für die Umsetzung von Industrie-4.0-Anwendungen auf Knopfdruck bereitstellt. Dies gilt insbesondere in Kombination mit einer Bosch-Rexroth-Industriesteuerung, um die Reichweite bis in die Fabrikhallen hinein auszudehnen. Der Cloud Service wird flexibel nach Nutzung abgerechnet, Betriebsaufwände entfallen, das unternehmerische Risiko wird minimiert. Der Weg in das neue Industriezeitalter ist geebnet.

Weitere Informationen

- [1] Bosch Rexroth Open Core Engineering: <https://www.boschrexroth.com/de/de/produkte/engineering/open-core-engineering/open-core-engineering>
- [2] Oracle IoT Cloud Service: <https://cloud.oracle.com/internet-of-things>
- [3] Oracle IoT Cloud Service Client Software: <http://www.oracle.com/technetwork/indexes/downloads/iot-client-libraries-2705514.html>
- [4] Geschäftsmodelle entwickeln – 55 innovative Konzepte mit dem St. Galler Business Model Navigator (Gassmann, Frankenberger, Csik – Carl Hanser Verlag München 2013)
- [5] Marcel Amende, Internet of Things: Referenz-Architektur, DOAG/SOUG News 04-2015



Marcel Amende
marcel.amende@oracle.com



Thorsten Schulz
thorsten.schulz2@boschrexroth.de



Event-Processing und Stream-Analyse

Arne Brüning, ORACLE Deutschland B.V. & Co. KG

Industrie 4.0 oder das Internet der Dinge, egal aus welcher Perspektive man die Aufgabe betrachtet: Die echtzeitnahe Analyse von hochvolumigen Daten, das Durchsuchen nach relevanten Mustern und die zeitnahe Reaktion sind üblicherweise Bestandteil eines jeden Szenarios. Die Werkzeuge dafür sind der Kinderstube entwachsen und die zweite Generation der Event-Processing-Tools betritt die Bühne. Auch bei Oracle ist eine neue Generation am Start und wirbt auf der einen Seite mit erweiterten Fähigkeiten wie Geo-Processing und Machine-Learning. Auf der anderen Seite wurde die Administrier- und Benutzbarkeit verbessert. Dieser Artikel gibt einen Überblick und eine Einführung in das aktuelle Oracle-Portfolio.

Grundsätzlich handelt es sich bei Lösungen zur Stream-Analyse um Systeme, die große Mengen an Daten aus verschiedenen Quellen zur Laufzeit nahezu in Echtzeit filtern, aggregieren und nach bestimmten Mustern durchsuchen können. Die Resultate werden dann an Geschäftsanwendungen weitergeleitet. Laut dem Beratungsunternehmen Forrester Research [1] besteht eine Streaming-Analytics-Lösung aus den folgenden Bausteinen:

- **Transformation**
Die hochvolumigen Datenströme können zu einem großen Teil nicht benötigte Daten („noise“) und teilweise nur zu einem kleinen Teil Nutzdaten („signal“) enthalten. Über geeignete Filtermechanismen filtern die Systeme die anfallenden Daten entsprechend, um eine Konzentration auf die relevanten Anteile zu ermöglichen. In dieser Phase lassen sich auch fehlerhafte Daten aussortieren. Meldet beispielsweise ein Temperatursensor für ein paar Millisekunden eine um einige Hundert Grad erhöhte Temperatur, wird es sich wahrscheinlich um einen Messfehler handeln.
- **Correlation**
Nachfolgend werden die erhobenen Daten mit anderen Datenströmen in Bezug gebracht. In einem Verkehrsdaten-System lässt sich so aus den Daten der einzelnen Sensoren eine Verkehrslage erstellen.
- **Enrichment**
Zusätzlich zur Korrelation der Messdaten untereinander kann eine Anreicherung durch zusätzliche Daten von verschiedenen Datenbanken nützlich sein. Beispielsweise lassen sich die erhobenen Daten des Netzwerk-Monitorings eines Telekommunikations-Providers gegen eine Datenbank mit vorhandenen Netzwerk-Kapazitäten abgleichen.
- **Time Windows**
Wichtig ist die zeitliche Komponente der Events, weshalb diese als zeit-

liche Ströme vorgehalten werden. Im Finanzwesen ist zum Beispiel bei der Chart-Analyse das sogenannte „W-Pattern“ von Belang. Dafür ist wichtig zu wissen, dass innerhalb eines bestimmten Zeitraums ein Kurs fällt, steigt, wieder fällt und wieder steigt, sodass der Graph eine „W“-Form annimmt. Rein durch die aktuellen Werte ist dies nicht zu ermitteln, deshalb ist die zeitliche Komponente notwendig.

- Pattern Matching**
Ist die beschriebene Vorverarbeitung getan, lässt sich gezielt nach Mustern suchen. Diese sind von der fachlichen Anwendung abhängig und die durchlaufenden Daten werden ständig nach den definierten Mustern durchsucht. Dabei kann auch das Ausbleiben eines Events ein wichtiges Muster sein. Wird am Flughafen ein Koffer an einem Fließband aufgegeben und taucht dieser nicht nach einer definierten Zeit am geplanten Ausgang wieder auf, ist er vermutlich irgendwo vom Band gefallen.
- Business Logic**
Der eigentliche Wert der Streaming Analysis entsteht, wenn Muster gefunden und infolgedessen Geschäftsanwendungen informiert werden, sodass diese in geeigneter Form darauf reagieren. So kann etwa das Verkehrs-

leitsystem auf einen Stau reagieren und die dynamische Verkehrsführung ändern, das gefundene W-Pattern veranlasst das Trading-System, dem Analysten eine Empfehlung für eine Finanz-Transaktion zu geben, oder der verlorene Koffer wird gesucht.

Gerade im Umfeld der ausgerufenen Industrie 4.0 ist hier am Industriestandort Deutschland eine gesteigerte Nachfrage nach Systemen zur Predictive Maintenance zu beobachten. Dies mag daran liegen, dass es sich hierbei nicht um ein abstraktes Forschungsprojekt, sondern um eine Maßnahme zur kurzfristigen Kosteneinsparung handelt. Die Erwartung ist, über ein Stream-Analyse-System einen bevorstehenden Maschinenausfall besser als bisher vorhersagen zu können. Wenn dann ein Wartungstechniker das System zeitnah reparieren kann, lassen sich Ausfallzeiten und gegebenenfalls Folgeschäden vermeiden, was direkt einen Kostenvorteil ergibt. Da die Ausfall-Szenarien durchaus komplex sein können, wird diese Analyse aber nicht alleine von einem an die Maschine angeschlossenen Steuer-PC durchgeführt. Oftmals sind weitere Daten erforderlich, etwa Wetterdaten, Daten von anderen Systemen oder auch Planungsdaten. Daher ist ein zentrales System notwendig, das die Anforderungen gemäß obiger Definition erfüllt.

Streaming Analytics mit Oracle

Event-Processing-Produkte haben bei Oracle eine lange Historie. Das Produkt Complex Event Processing (CEP) wurde bereits zu BEA-Zeiten entwickelt, kam durch die Übernahme zu Oracle und wurde hier später in Oracle Event Processing (OEP) umbenannt. Entsprechend der langen Zeit am Markt ist auch die Liste der Referenzkunden. So nutzt zum Beispiel Japans größte Telefongesellschaft NTT OEP für das Netzwerk-Monitoring.

Auch schon sehr lange ist OEP Embedded verfügbar, das auf kleinen Devices wie dem Raspberry Pi läuft. Da die Embedded-Variante weitestgehend identisch mit der Server-Variante ist, können auf dem Device die gleichen Event Processing Networks ablaufen wie auf dem Server. Ausgenommen sind lediglich ein paar komplexe Operationen, mit denen die Device-Hardware überfordert wäre.

Intern setzt OEP auf Frameworks wie Spring und OSGi auf. Sie werden genutzt, um OEP extrem leichtgewichtig umzusetzen. Die Kehrseite war bislang: Zur Nutzung von OEP waren Programmierkenntnisse und Kenntnisse der Frameworks notwendig. In der Praxis hat sich aber gezeigt, dass die Algorithmen zur Pattern-Suche oftmals noch gar nicht fertig sind und eher explorativ entwickelt werden. Diese Algorithmen werden nicht von Entwicklern, sondern von den Experten der jeweiligen fachlichen Domänen entwickelt. Daher hat Oracle den Stream Explorer, heute Stream Analytics, entwickelt. Einmal richtig aufgesetzt, können damit auch Anwender ohne Programmierkenntnisse die entsprechenden Analysen interaktiv erstellen.

Oracle Stream Analytics 12.2.1

Das aktuelle Server-Produkt für Event-Processing und Stream-Analyse ist OSA Version 12.2.1. Voraussetzung sind laut Preisliste Coherence Enterprise Edition oder die WebLogic Suite, die Coherence wiederum enthält. Die Installation ist sehr einfach. Der Download enthält wie bei Oracle üblich eine „.jar“-Datei. Diese wird einfach mit „java -jar /media/sf_Share/OSA/fmw_12.2.1.0.0_osa_generic.jar“ aufgerufen, um den Universal Installer zu starten.

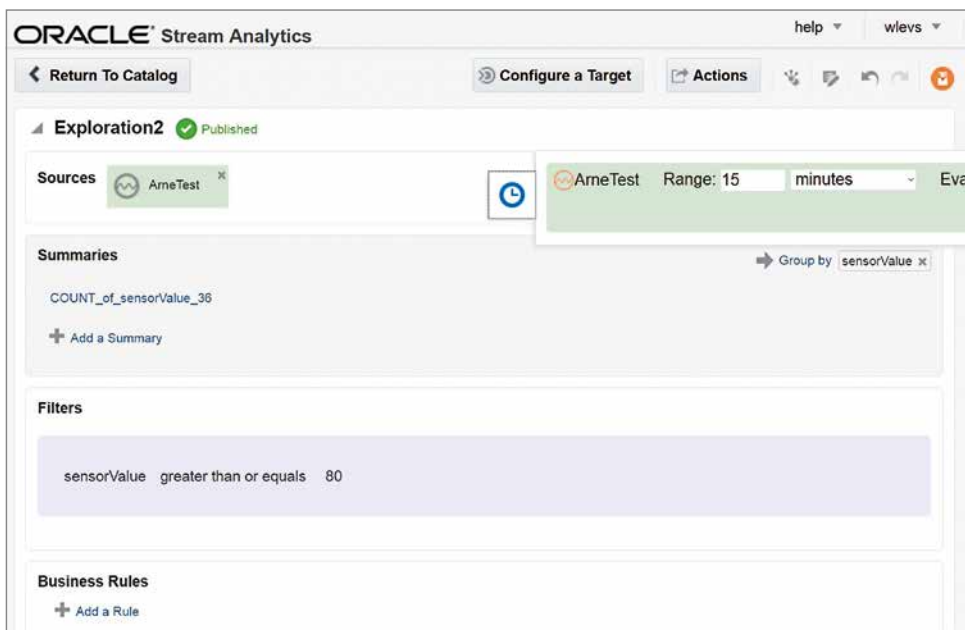


Abbildung 1: Live-Operationen auf den Stream

Neben dem Zielverzeichnis kann man als einzige Option auswählen, ob die Beispiele dazu installiert werden sollen.

Wie vom WebLogic Server bekannt, muss als Nächstes eine Domain angelegt werden. Dies erfolgt wie üblich über das „config“-Skript, unter Linux beispielsweise mit „oep/common/bin/config.sh“. Die wenigen Optionen sind auch hier schnell geklärt: Domain erstellen oder ändern, mit oder ohne Beispielen, Password für Administrator und Keystore – eine Minute später steht die Domain bereit. Gestartet wird der Server dann per „startwlevs.sh“ aus dem Domainverzeichnis.

Überprüft werden kann der laufende Server, wenn man per Browser den bei der Installation eingegebenen Port (Standard: „9002“) öffnet. War die Installation erfolgreich, wird man auf die URL „/wlevs“ umgeleitet. Dort meldet sich der Event Processing Visualizer. Ändert man die URL auf „/sx“, gelangt man auf die Hauptseite von Oracle Stream Analytics (siehe Abbildung 1).

Am Design der beiden Einstiegsseiten lassen sich sofort die Zielgruppen erkennen. Die Hauptseite von Oracle Stream Analytics ist bewusst im typischen Oracle-JET-Design einfach gehalten, das auch die Oracle Cloud Services prägt. Anhand von Templates, die typische Use Cases abbilden, kann der Fachanwender in die Modellierung einsteigen. Die darunterliegende Technik ist hier bewusst versteckt. Der Event Processing Visualizer hingegen bietet dem Entwickler oder Administrator Zugriff auf alle darunterliegenden Details.

Einstieg in Stream Analytics

Der Einstieg in Stream Analytics lässt sich leicht an einem Beispiel demonstrieren. Dazu wechselt man von der Willkommenseite auf den Reiter „Catalog“. Sind die Beispiele mit installiert, finden sich hier bereits einige Beispielobjekte. Um einen Stream zu analysieren, wird zunächst natürlich ein solcher benötigt. Entwicklerfreundlich ist hier als ein möglicher Stream auch eine „.csv“-Datei vorgesehen. Beim Oracle Technology Network sind einige Beispieldateien zum Download angeboten, sodass man ohne große Vorbereitungen schnell erste Erfahrungen mit dem System machen kann.

In den Beispieldateien findet sich eine Datei „smalltrends.csv“, die für dieses Bei-

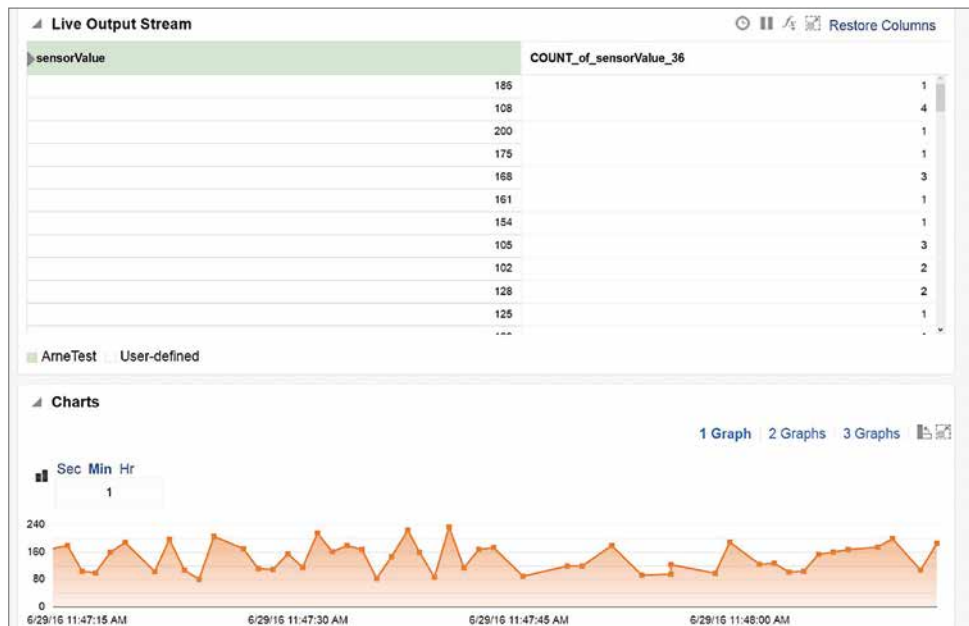


Abbildung 2: Ausgabe des Beispiel-Streams

spiel genommen wurde. Im „Catalog“ erstellt man per „Create New Item“ einen neuen Stream und muss hier als Erstes den Typ auswählen. OSA bietet eine große Auswahl möglicher Streams. Neben Oracle-eigenen Technologien wie Coherence und offenen Standards wie REST findet sich hier auch Unterstützung für das im IoT-Umfeld wichtige Protokoll „MQTT“ oder für das populäre Open Source Messaging System „Apache Kafka“. Hier gibt es auch die Option, eine „.csv“-Datei als Stream-Quelle zu nehmen. Nach Auswahl der Datei wird man von OSA noch aufgefordert, einen Shape auszuwählen oder neu zu definieren. Dabei handelt es sich um eine Endbenutzer-Abstraktion des zu verwendenden Datentyps; hier wurde bewusst die Entwicklersprache vermieden.

Außerdem wird gefragt, ob gleich eine Exploration erstellt werden soll. Die Exploration ist die Abbildung des Streams mit allen darauf getätigten Operationen sowie den Live-Daten. Hat man ausgewählt, dass eine solche Exploration mit erzeugt werden soll, öffnet sich diese im Anschluss direkt im Exploration-Editor und der Anwender sieht sofort die Daten des Streams live durchlaufen. In diesem einfachen Beispiel sind es die Wertepaare für „sensorID“ und „sensorValue“. Hier kann der Anwender nun direkt anfangen, mit den Daten zu experimentieren, exploratives Vorgehen ist hier ausdrücklich vorgesehen.

Im Bereich „Sources“ lassen sich auch weitere Streams mit dem vorhandenen verknüpfen („Correlation“). Der Bereich darunter enthält die der Summenbildung und Gruppierung. In diesem Beispiel erfolgt ein Summary auf „COUNT OF sensorValue“ und ein „GROUP BY sensorValue“. Ähnlichkeiten der darunterliegenden CQL-Abfragesprache zu SQL sind absolut gewollt.

Wie eingangs erwähnt, spielt bei der Stream-Analyse aber auch die zeitliche Komponente eine wichtige Rolle. Ohne weitere Änderungen würde das Aggregat nämlich immer „1“ betragen. Grund dafür ist, dass sich innerhalb eines betrachteten Augenblicks immer nur ein Wert im Stream befindet. Um anzuzeigen, wie häufig ein bestimmter Wert auftrat, muss der Exploration also ein Zeitfenster mitgegeben werden. Dieses kann vorgeben werden, wenn man das Uhren-Symbol am rechten Rand anwählt. In diesem Beispiel sind 15 Minuten vorgegeben. Im dritten Bereich lassen sich Filter definieren („Transformation“); zu Demonstrationszwecken wurde ein einfaches „sensorValue greater than or equals 80“ eingefügt. Im letzten Bereich lassen sich dann über Business Rules auch komplexere Muster suchen. Das Ergebnis wird dann live dargestellt, sowohl als durchlaufende Wertetabelle als auch als Graph (siehe Abbildung 2).

Bislang sind die resultierenden Werte nur in der Design-Oberfläche dargestellt.

Ist der gewünschte Ergebnis-Stream gefunden, muss dieser zur weiteren Verwendung an ein verarbeitendes System weitergeleitet werden. Das kann, wie im obigen Beispiel, wieder ein neuer Datenstrom sein. Explorations lassen sich auch beliebig verketteten. So kann der resultierende Strom auch wieder ein Eingabestrom für eine weitere Exploration sein oder es wird nur nach bestimmten Mustern gesucht („Pattern Matching“). Typischerweise enthält der Ergebnis-Strom dann nur selten Daten, nämlich dann, wenn das gesuchte Muster gefunden wurde.

In jedem Fall lässt sich für die Exploration ein Target definieren, dies können unter anderem ein Service-Endpunkt oder ein Queuing-System sein. Dahinter können dann die Geschäftsanwendungen auf die gefundenen Daten reagieren („Business Logic“). Auch hier gibt es wieder die Option, den Ergebnis-Strom in eine „.csv“-Datei zu schreiben; die wurde in diesem Beispiel verwendet. Ist die Exploration dann soweit fertig, wird sie via „Publish“ live geschaltet, bearbeitet die eingehenden Datenströme wie vorgegeben und meldet die Ergebnisse an die Geschäftsanwendungen.

Weitergehende Funktionalitäten

Dieses simple Beispiel zeigt nur einen kleinen Teil des Umfangs von OSA, die

grundsätzliche Arbeitsweise von OSA sollte daran allerdings erkennbar sein. Einen Eindruck von der Mächtigkeit dieses Werkzeugs bekommt der Anwender beim Öffnen des dritten Reiters „Patterns“. Diese Patterns bilden eine Vielzahl typischer Stream-Analysen ab. Beim Klick auf eines der Patterns bekommt der Anwender eine Kurzbeschreibung, einen Link auf die Dokumentation und ein YouTube-Video, das das Pattern kurz vorstellt.

Hier findet sich auch das Pattern „K-means Anomaly Detection“, ein Machine-Learning-Algorithmus, der ein automatisches Clustering durchführt, also eine Gruppierung der Daten. Cluster mit einer geringen Menge von Werten werden als Anomalie angesehen. Hiermit lassen sich schnell und einfach Ausreißer im Datenstrom finden.

Machine-Learning-Verfahren werden voraussichtlich in Zukunft stark an Bedeutung gewinnen. So lange nur nach vordefinierten Mustern gesucht wird, werden auch nur diese gefunden. Das kann natürlich auch schon sehr hilfreich sein. Wenn beispielsweise das bekannte Muster für einen bevorstehenden Maschinenausfall erkannt und der Wartungstechniker rechtzeitig auf den Weg geschickt wurde, hat sich das System bewährt.

Interessant ist jedoch auch, nach Mustern zu suchen, die noch gar nicht bekannt sind. Wenn etwa im vorigen Beispiel der Wartungstechniker immer zu spät kommt, weil das Muster erst direkt vor dem Ausfall gefunden wird,

dann müssen weitere Daten hinzugezogen werden. Beispielsweise von anderen Maschinen, die weiter vorne in der Verarbeitungskette stehen, oder Umgebungsdaten wie das aktuelle Wetter. Mit der Komplexität der Daten wächst aber auch der Aufwand für die Daten-Analysten, hierin nach geeigneten Mustern zu suchen. Machine-Learning-Algorithmen können hier eine große Hilfe sein.

Eine weitere Neuerung ist das sogenannte „Geofencing“ inklusive Visualisierungs-Komponente. Die Datenströme können auch Tupel von Geo-Koordinaten enthalten. Oracle Event Processing kann überprüfen, ob sich die Koordinaten inner-/außerhalb von definierten Regionen befinden. Ein Ereignis kann also ausgelöst werden, wenn ein Objekt eine dieser Regionen verlässt. In der aktuellen Version bietet auch OSA das Geofencing an und liefert mit der Map gleich eine passende Darstellung dazu. Auf der Map lassen sich dann live die empfangenen Koordinaten und die definierten Regionen darstellen (siehe Abbildung 3).

Die Engine unter OSA heißt „Oracle Event Processing“ (OEP). Somit finden sich die Explorations auch im Event Processing Visualizer des OEP wieder. Auch können die Explorations aus OSA exportiert, in den JDeveloper importiert und hier als Event Processing Networks bearbeitet werden, etwa wenn externe Libraries zur Datenauswertung einbezogen werden sollen, was unter der OSA-Oberfläche nicht möglich ist.

Oracle Edge Analytics ist der Nachfolger von OEP Embedded. Auch hierauf können die gleichen Event Processing Networks ablaufen wie auf der Server-Variante, sodass über diesen Weg auch die Explorations auf einer Vielzahl von Devices zur Ausführung gebracht werden können, da als Systemvoraussetzung lediglich eine passende JVM benötigt wird.

In der aktuellen Version ist OEP aber nicht mehr die einzige Engine, die von OSA genutzt werden kann. Alternativ ist als Ablaufumgebung auch Apache Spark möglich. Beim aktuellen Stand der Entwicklung bietet OEP eine bessere „single node“-Performance sowie eine umfassendere Funktionalität. Spark hingegen kann die Last sehr gut in einem Cluster verteilen. Beide Engines haben unterschiedliche Stärken – gut für den Anwender, dass er hier die Wahl hat.



Abbildung 3: Geofencing-Visualisierung

Coherence kann sowohl als Quelle als auch als Ziel von OSA dienen. Am Beispiel von Predictive Maintenance könnten etwa alle Geräte ihre Messdaten in einem Coherence-Cluster ablegen. Aus diesem führt OSA dann die Stream-Analyse durch und eine mobile Wartungsanwendung liest beispielsweise benötigte Live-Daten, während Coherence mittels „Write-Through“ oder „Write-Behind“ die Daten in die Zielsysteme schreibt.

Selbstverständlich hat Oracle auch eine Integration mit Datenbanken im Angebot. Mit „Oracle GoldenGate for Big Data“ können Änderungen aus einer Datenbank über eine Kafka-Queue direkt in OSA weiterverarbeitet werden. Mit dem IoT Cloud Service (IoTCS) bietet Oracle einen PaaS Service an, der an anderer Stelle in dieser Ausgabe beschrieben wird. OSA ist ein wichtiger Bestandteil des IoTCS, umgekehrt ist der IoTCS eine Möglichkeit, OSA als Cloud-Service zu nutzen.

Fazit

Oracle Stream Analytics ist innerhalb von Minuten installiert und nach kurzer Zeit lassen sich Stream-Analysen durchführen, dank der neuen Oberfläche jetzt auch von Fachanwendern. Trotz der einfachen Endbenutzer-Oberfläche liegt unter Stream Analytics die volle Mächtigkeit von Oracle Event Processing, sodass ein Entwickler gegebenenfalls auch komplexere Event-Processing-Networks erstellen kann. Diese können dann auch auf einer Vielzahl von Devices laufen. Durch die SQL-ähnliche Abfragesprache CQL werden sich Datenbank-Entwickler hier schnell heimisch fühlen. Mit dem aktuellen Release wird die Funktionalität mit Machine-Learning und Geofencing wesentlich erweitert, durch die Öffnung in Richtung populärer Open-Source-Werkzeuge ist OSA jetzt auch einfacher in heterogene Umgebungen zu integrieren.

Quelle

- [1] The Forrester Wave: Big Data Streaming Analytics, Q1 2016



Arne Brüning
arne.bruening@oracle.com

Ihre Oracle Datenbanken können Sie vergessen



Unser Team von zertifizierten, deutschsprachigen Oracle Datenbank Administratoren/innen übernimmt rasch und professionell via Fernwartung alle Aufgaben im Oracle Datenbank- und Middleware Umfeld.

Durch pro-aktives Monitoring, garantierte Reaktionszeiten und schnelle Problemlösungen helfen wir Ihnen rund um die Uhr den ungestörten und fehlerfreien Betrieb Ihrer Oracle Datenbanken zu gewährleisten.



IoT-Architektur – sind traditionelle Architekturen gut genug oder braucht es neue Ansätze?

Guido Schmutz, Trivadis AG

„Internet of Things“ (IoT) ist eines der zentralen Themen der Digitalisierung. Es handelt sich dabei um die Vision, dass sämtliche Dinge auf der Welt das Potenzial besitzen, sich über das Internet zu verbinden und ihre Daten zur Verfügung zu stellen. Damit lassen sich neue, innovative Produkte und Services bauen, mit denen man sich von der Konkurrenz differenzieren kann.

Das Internet der Dinge wird unser Leben grundlegend verändern. Es wird Auswirkungen haben auf die Art und Weise, wie wir wohnen, arbeiten, einkaufen, zusammenleben und reisen. Unternehmen, die heute erste IoT-Anwendungsfälle umsetzen, können noch als „Early Adopter“ bezeichnet werden. Dies wird sich aber rasch ändern und man wird früher oder später mit der Aufgabe konfrontiert werden, eine Integration von IoT in eine bereits existierende Anwendungslandschaft zu schaffen.

Dieser Artikel zeigt, ob und wie sich eine IoT-Anwendung in eine bestehende Architektur integrieren lässt und wo die Grenzen einer solchen Lösung lie-

gen. Es wird darauf eingegangen, wie sich eine bestehende Systemumgebung durch neue, spezialisierte Komponenten so erweitern lässt, dass das äußerst hohe Nachrichtenvolumen, das durch die IoT-Devices verursacht wird, ohne Probleme verarbeitet werden kann. Bevor wir mit der Diskussion starten, hier die drei zentralen Fragen, auf die es bei einer IoT-Anwendung eine Antwort zu finden gilt:

- *Verbinden*

Wie werden die Daten von IoT Devices abgegriffen und wie kann die Komplexität der Verbindung zu den unterschiedlichen Geräten möglichst gering gehalten werden?

- *Analyse*

Wie können die IoT-Daten rasch und effizient analysiert werden, um die interessantesten von den nicht interessanten Ereignissen zu trennen?

- *Integrieren*

Wie werden die IoT-Daten beziehungsweise die wichtigen Ereignisse in die (bestehende) Unternehmens-Infrastruktur integriert, sodass sowohl bestehende als auch neue Prozesse genutzt werden können?

Wir nehmen traditionelle Service- beziehungsweise Nachrichten-orientierte Architektur als Ausgangspunkt und starten

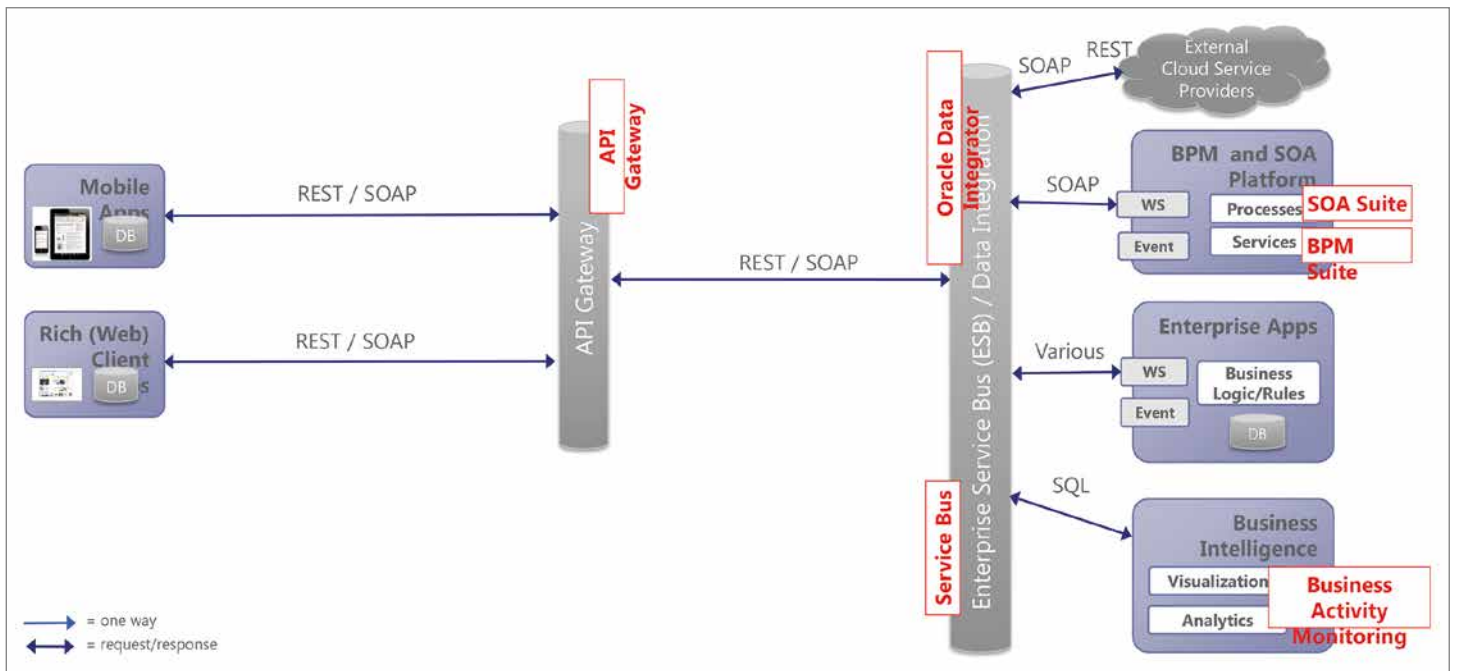


Abbildung 1: Traditionelle, SOA-basierte Architektur mit Oracle Fusion Middleware (in Rot)

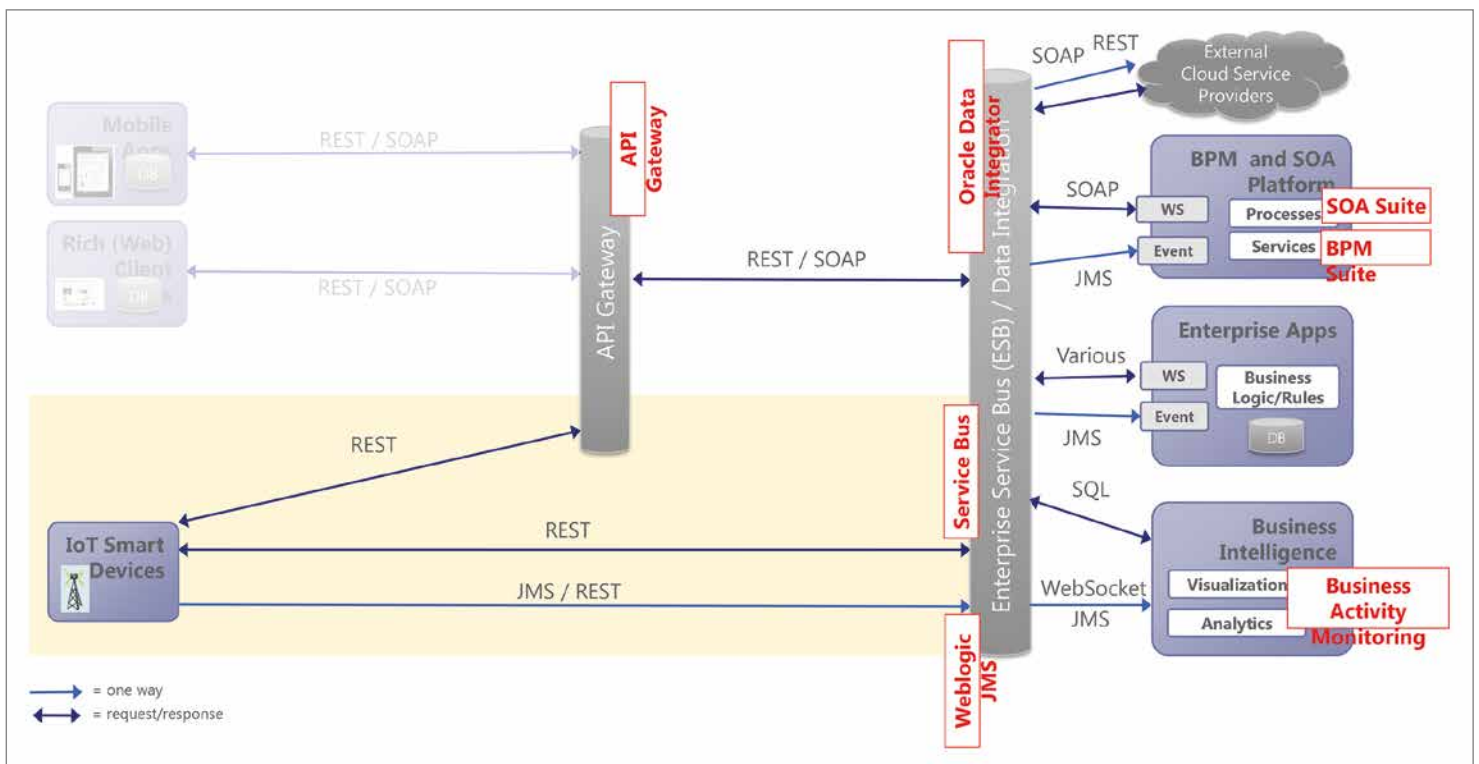


Abbildung 2: Erweiterung der traditionellen Architektur mit IoT-Smart-Devices

mit einer heute gängigen Architektur, wie sie in *Abbildung 1* vereinfacht dargestellt ist. Die Applikationen auf der linken Seite nutzen Dienste, die von den Systemen auf der rechten Seite über Services angeboten werden. In der Mitte dient ein Enterprise Service Bus (ESB) zur Entkoppelung der Services von den Nutzern (Ser-

vice-Virtualisierung) und das API-Gateway zur Sicherheit der Services. Es werden sowohl SOAP- wie auch REST-basierte Services unterstützt. Um es ein bisschen konkreter zu machen, sind mögliche Produkte aus dem Oracle Stack in Rot dargestellt. Es handelt sich dabei um die Produkte aus der Oracle Fusion Middleware.

Dieses Bild dient als Ausgangslage, um die Frage zu diskutieren, ob für eine IoT-Anwendung traditionelle Architekturen gut genug oder ob neue Architektur-Ansätze notwendig sind.

Mit der Umsetzung einer ersten IoT-Anwendung wird man sich zwangsläufig die Frage stellen müssen, ob eine beste-

hende Infrastruktur beziehungsweise Architektur dafür genutzt werden kann. Eine IoT-Anwendung unterscheidet sich von herkömmlichen Client-Applikationen primär dadurch, dass Daten ausschließlich automatisch, ohne menschliche Interaktionen generiert werden (vom Sensor), aber weniger durch die Business-Logik, die es situationsbedingt auszuführen gilt. Viele der bestehenden Backend-Applikationen wollen also auch bei IoT-Anwendungen weitergenutzt werden. Können wir die Geräte nicht einfach als neue Quelle sehen und analog zu den Web- oder Mobile-Applikationen in unsere traditionelle Architektur integrieren, wie dies in *Abbildung 2* dargestellt ist?

In diesem Fall kommunizieren die IoT-Devices über die Services, die vom API-Gateway oder vom ESB exponiert werden und in den Backend-Systemen implementiert sind. Für die „One-way“-Kommunikation, also vom Device zum Backend, könnte auch der Einsatz einer Nachrichten-orientierten Middleware ins Auge gefasst werden, mit dem Vorteil der besseren Entkopplung und eines zuverlässigeren Nachrichten-Transfers. Im Falle von Oracle wäre dies ein WebLogic JMS Server, der das Java Message API (JMS) unterstützt. Funktional gesehen ist dies

kaum ein Problem, man wird allerdings mit folgenden Problemen beziehungsweise Herausforderungen konfrontiert:

- Besitzen die IoT-Devices genügend Ressourcen, um direkt über ein Internet-Protokoll (HTTP) beziehungsweise über JMS zu kommunizieren? Dagegen kann sowohl der erhöhte Stromverbrauch als auch der Formfaktor eines IoT-Device sprechen. Man spricht hier auch von Smart Devices, falls diese genügende Intelligenz haben, um direkt mit dem Backend zu sprechen. Andere Devices sollen möglichst lange mit einer Batterieladung auskommen, weil sie einen Schlafmodus besitzen und diesen nur kurzzeitig für das Senden eines neuen Messpunkts verlassen.
- Soll das Device nur Daten aufnehmen und weiterleiten (Sensor) oder ist auch der Weg zurück notwendig, um den Kontext des Device zu verändern (Aktuator)? Wie kann in diesem Fall das Device erreicht werden?
- Lässt sich JMS von den externen IoT-Devices direkt nutzen? Sind die Firewalls für das JMS-Protokoll offen?

- Wie viele IoT-Devices sind insgesamt angedacht und in welcher Periodizität senden sie Nachrichten an das Backend? Kann das daraus resultierende Volumen durch den Service Bus oder JMS Server verkraftet werden? Hier gilt es zu beachten, dass 300 Nachrichten pro Sekunde für eine JMS Queue bereits die obere Grenze darstellen können. Falls also JMS genutzt wird, dann wird der JMS Server den ersten möglichen Engpass darstellen.
- Welche Operationen sind auf den IoT-Nachrichten (Ereignisse) notwendig, um entsprechende Aktionen abzuleiten? Sind es ausschließlich Filter-, Transformation- und Routing-Operationen, dann kann dies funktional auch von einem ESB problemlos bewältigt werden.

Es ist also sehr wohl möglich, erste IoT-Anwendungen über eine bestehende Service- beziehungsweise Nachrichten-orientierte Infrastruktur anzugehen; man sollte aber jederzeit die Grenzen der Plattform im Hinterkopf haben. Was geschieht, wenn eine IoT-Anwendung, die klein startet, zum Erfolg wird und

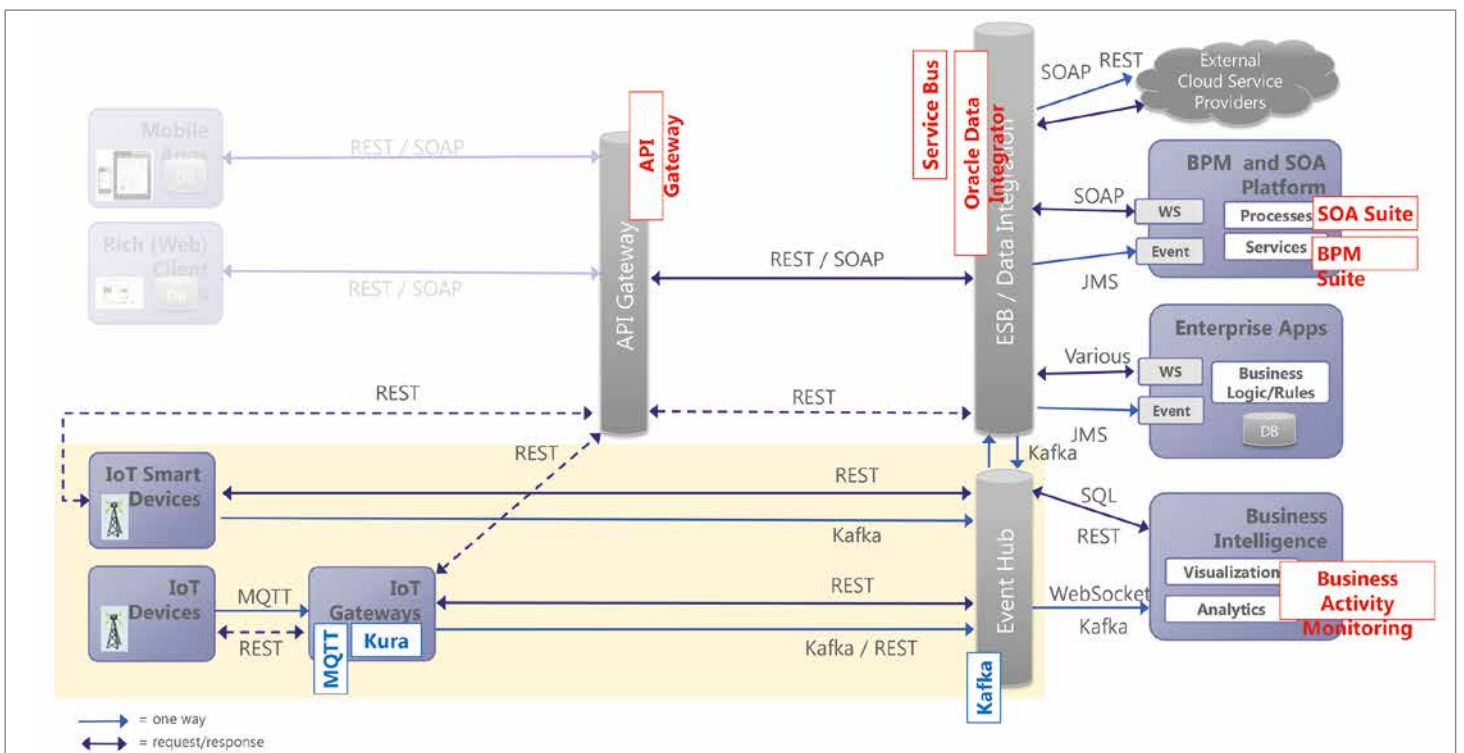


Abbildung 3: Erweiterung der traditionellen Architektur mit IoT-Gateways und modernen Message-Broker

sich die Nachrichtenmenge exponentiell vergrößert? Wie wirkt es sich aus, wenn die einzelnen IoT-Devices mit geringerer Frequenz messen und anstatt im 15-Sekunden-Takt neu im 1-Sekunden-Takt neu im 1-Sekunden-Takt Daten übermitteln?

Eine bestehende Service- oder Nachrichten-orientierte Infrastruktur ist meis-

tens von zentraler Wichtigkeit für ein Unternehmen und die IoT-Anwendung darf die bisher garantierten Service-Level auf keinen Fall gefährden. Das Skalieren der bestehenden Infrastruktur kann rasch teuer werden, wenn es darum gehen soll, um Faktoren größere Nachrichtenmengen verkraften zu können.

Event Hubs und optionale IoT-Gateways einführen

Ein Event Hub ist darauf ausgelegt, riesige Datenströme effizient und zuverlässig entgegenzunehmen und weiterzuleiten. Er stellt dabei eine noch relativ neue Alternative zu einem klassischen JMS Server dar,

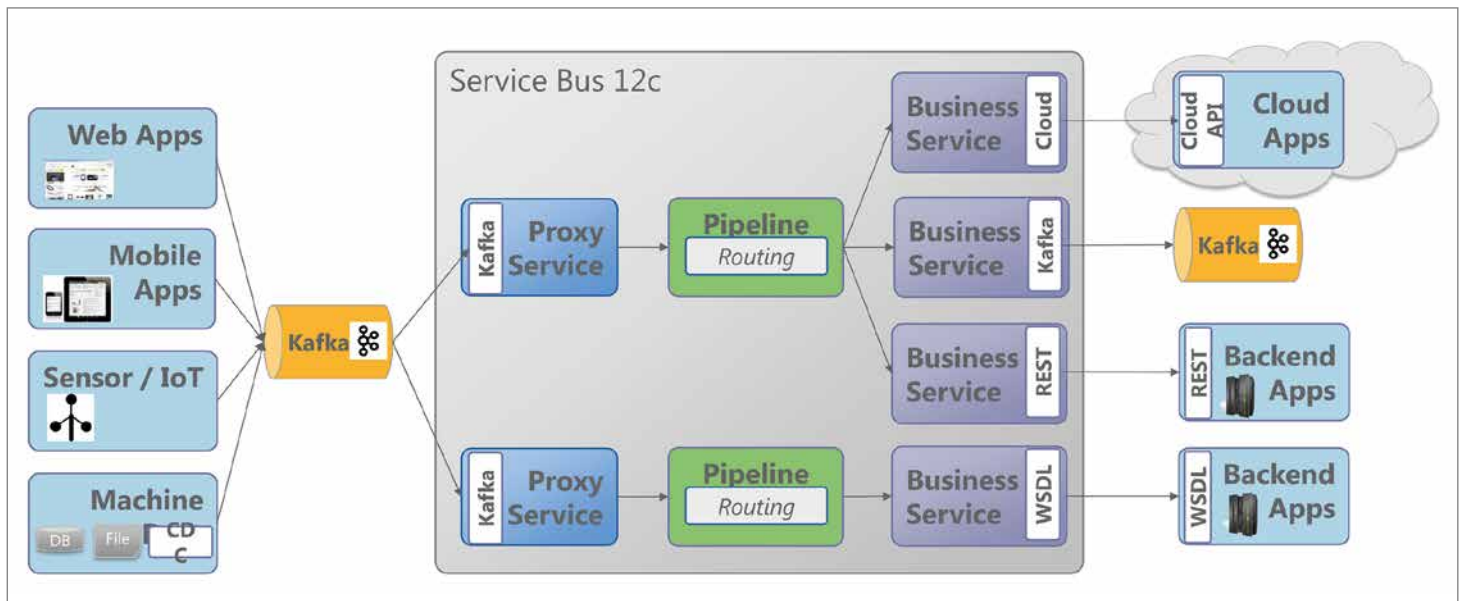


Abbildung 4: Integration von Apache Kafka mit dem Service Bus von Oracle

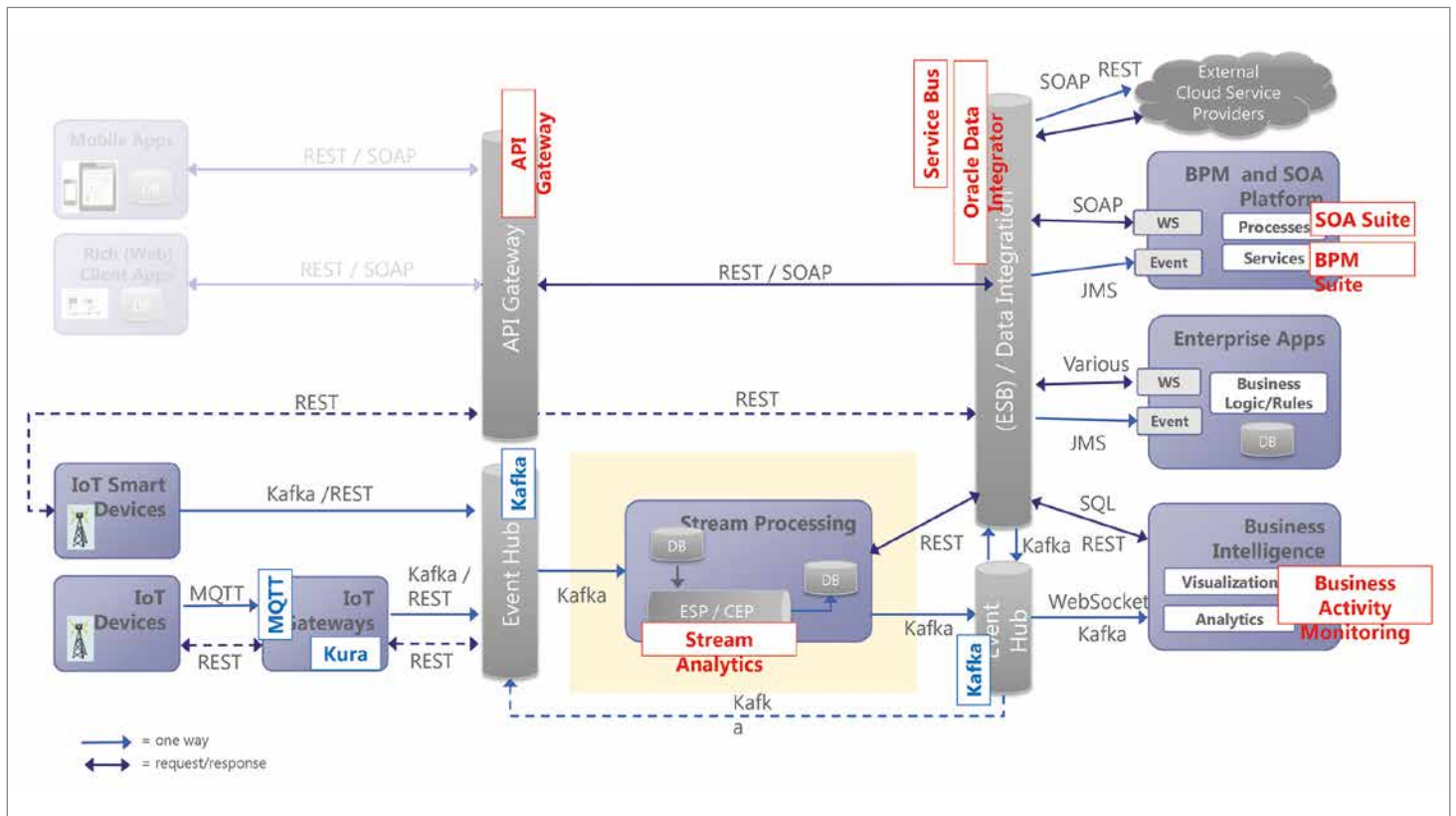


Abbildung 5: Erweiterung um Stream-Processing-Funktionalitäten

da er mit gleichen Ressourcen einen um Faktoren größeren Nachrichten-Durchsatz gewährleistet. Um auch nicht intelligente Devices unterstützen zu können, also Devices, die mit geringen Ressourcen auskommen müssen, können IoT-Gateways in die Kommunikation eingebunden werden. Ein Gateway stellt dabei die Konnektivität mit dem Backend sicher, wird möglichst nahe zu einer Gruppe von Devices platziert, die sich dann über spezialisierte, leichtgewichtige Protokolle mit dem Gateway verbinden. Dadurch sinken die Kosten und die Komplexität der Devices wird drastisch reduziert. *Abbildung 3* zeigt die Erweiterung der Architektur mit IoT-Gateways und dem Event Hub.

Bei dieser Variante kommuniziert das IoT-Device mit dem Gateway über MQTT, einem Vertreter der neuartigen IoT-Protokolle. Alternativen dazu wären ZigBee, BLE oder CoAP. Für die Gateway-Logik kann Eclipse Kura zum Einsatz kommen, ein Java-basiertes Framework für die Implementierung von IoT-Gateways. Als Event Hub agiert Apache Kafka, das ursprünglich von LinkedIn entwickelt und später der Apache Foundation zur Verfügung gestellt wurde. Es erfreut sich großer Popularität und wird mittlerweile von verschiedenen Oracle-Produkten unterstützt.

Apache Kafka organisiert Nachrichten über verschiedene Topics. Ein Produzent sendet Nachrichten in ein Topic und jede

einzelne Nachricht lässt sich immer von mehreren Konsumenten konsumieren (Publish-Subscribe-Verfahren). Konsumenten können auch erst Stunden oder Tage später Nachrichten am Stück konsumieren, ohne dass dies einen Einfluss auf die Performance hat.

Abbildung 4 zeigt, wie sich Kafka mit dem Service Bus von Oracle integrieren lässt. Wie man sieht, kann der Service Bus sowohl von einem Kafka Topic lesen als auch in ein Kafka Topic schreiben.

Wir können also die Nachrichten von Kafka direkt vom Service Bus konsumieren und dann wie bisher an die Backend-Systeme weiterleiten. Darüber hinaus können Dashboards oder BI-Applikationen Informationen auch direkt aus Kafka konsumieren, ohne über den ESB gehen zu müssen. Mit dem Ersetzen von JMS durch Kafka entfällt ein Engpass. Die Frage bleibt nun aber, ob und wann der Service Bus zum nächsten Flaschenhals wird.

Wie bereits erwähnt, lässt ein ESB nur einfache Operationen auf dem Ereignisstrom zu, da er zustandslos agiert, was auch für den Service Bus gilt. Für die Analyse des Ereignisstroms wären die Funktionalitäten des sogenannten „Complex Event Processing“ (CEP) sehr hilfreich, wie

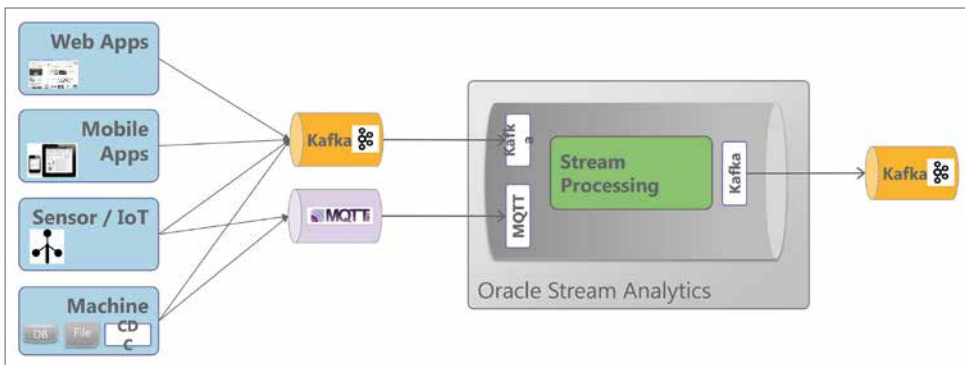


Abbildung 6: Integration von Apache Kafka mit Oracle Stream Analytics

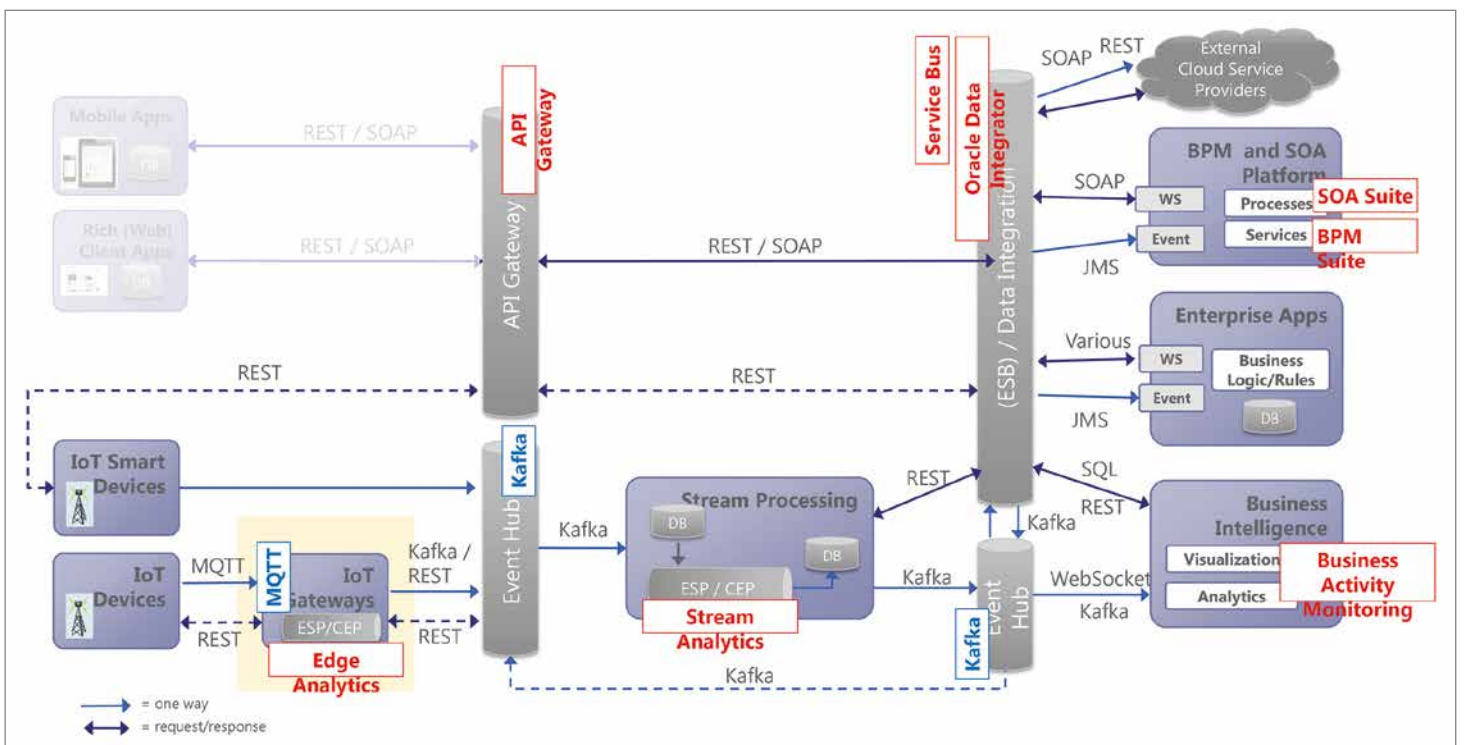


Abbildung 7: Erweiterung mit Stream Processing auf der Ebene des IoT-Gateway

sie von Stream-Processing-Plattformen angeboten werden. Dazu zählen unter anderem Aggregationen von mehreren Ereignissen über Zeitfenster, Erkennen von Patterns über mehrere unterschiedliche Ereignisse und raumbezogenen Operationen. Der nächste Schritt ist daher die Erweiterung der Architektur um eine Stream-Processing-Infrastruktur.

Stream-Processing-Infrastruktur für Analysen des Ereignisstroms

Um den riesigen Ereignisströmen Herr zu werden, die von den IoT-Smart-Devices beziehungsweise IoT-Gateways im Event Hub landen, lässt sich eine Stream-Processing-Infrastruktur einsetzen. Diese erlaubt es, die Ereignisse minimaler Latenzzeit zu analysieren und zu entscheiden, ob und – wenn ja – welche Aktion ausgeführt werden soll. *Abbildung 5* zeigt die Erweiterung der Architektur um eine Stream-Processing-Komponente.

Oracle bietet dabei mit Oracle Stream Analytics (OSA) eine sehr mächtige Plattform, die alle oben erwähnten CEP-Funktionalitäten anbietet. Sie unterstützt Apa-

che Kafka sowohl als Konsument wie auch als Produzent (*siehe Abbildung 6*).

In der Architektur agiert das Stream Processing quasi als Vorfilter. Die Nachrichten werden vom Event Hub konsumiert und durch einfache oder komplexe Analytik-Verfahren auf die Menge reduziert, für die es zu reagieren beziehungsweise zu agieren gilt. Dies können beispielsweise alle Ereignisse über einem bestimmten Schwellenwert sein oder aber das Erkennen eines ausbleibenden/ fehlenden Ereignisses aufgrund eines komplexen Pattern Matching über mehrere Ereignisse und die Zeit hinweg. Diese Erkenntnisse führen dann zu Aktionen in den Backend-Systemen, die vom Stream Processing über einen Service-Aufruf auf dem ESB ausgelöst werden.

Mit dem Stream Processing wurde die Architektur um eine mächtige Komponente erweitert, die hochgradig skalierbar ist und auch fehlertolerant aufgesetzt werden kann. Die Erweiterung der Backend-Architektur um Event Hub und Stream Processing macht es möglich, die massiven Ereignisströme effizient und in Echtzeit abzuarbeiten.

Die Frage, die sich als Nächstes stellt, ist, ob sich das Nachrichtenvolumen, das

von den IoT-Smart-Devices und IoT-Gateways erzeugt wird, überhaupt über das bestehende Netzwerk transferieren lässt. Falls nicht, lässt sich das Netzwerk in der Kapazität mit vertretbarem Aufwand erweitern? Falls dies auch nicht möglich oder finanziell gangbar ist, dann gilt es, die Nachrichtenmenge bereits auf den IoT-Gateways oder Smart-Devices sinnvoll zu reduzieren.

Stream Processing auf den IoT-Gateways

Mit dem IoT-Gateway steht ein Gerät mit Intelligenz zur Verfügung; im Prinzip ein kleiner Computer, der zu mehr genutzt werden kann als zum reinen Empfangen und Weiterleiten der Daten von den angehängten IoT-Devices. Wie in *Abbildung 7* gezeigt, kann ein Event Processing auch auf dem Gateway erfolgen.

Es handelt sich hier allerdings um eine wesentlich leichtgewichtige Variante als im Backend. Oracle bietet mit Oracle Edge Analytics genau so eine Komponente an, quasi den kleinen Bruder von Oracle Stream Analytics, die sich auch auf Embedded Rechnern wie einem Raspberry

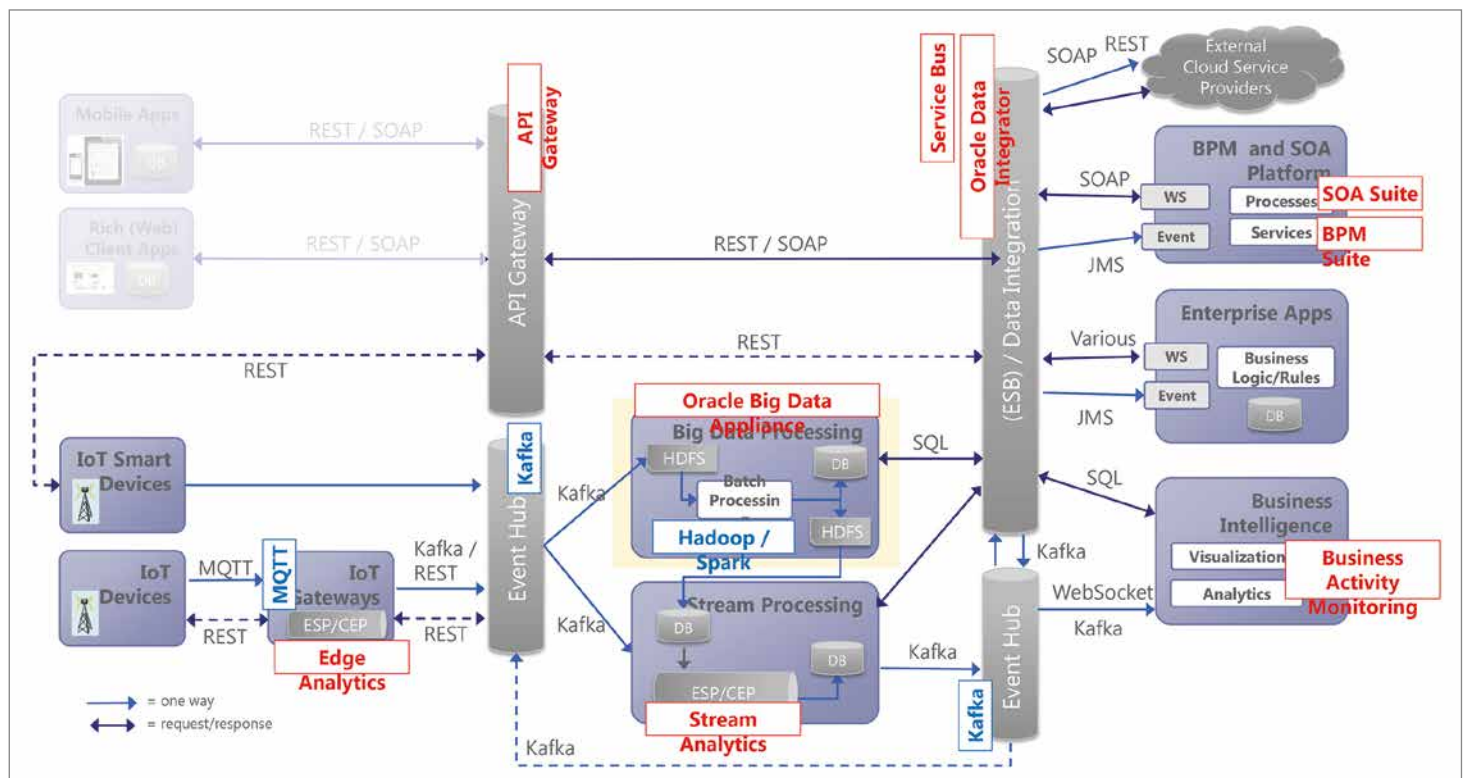


Abbildung 8: Erweiterung um eine Big-Data-Infrastruktur für die Speicherung der Rohdaten und mögliche Big-Data-Batch-Verarbeitung

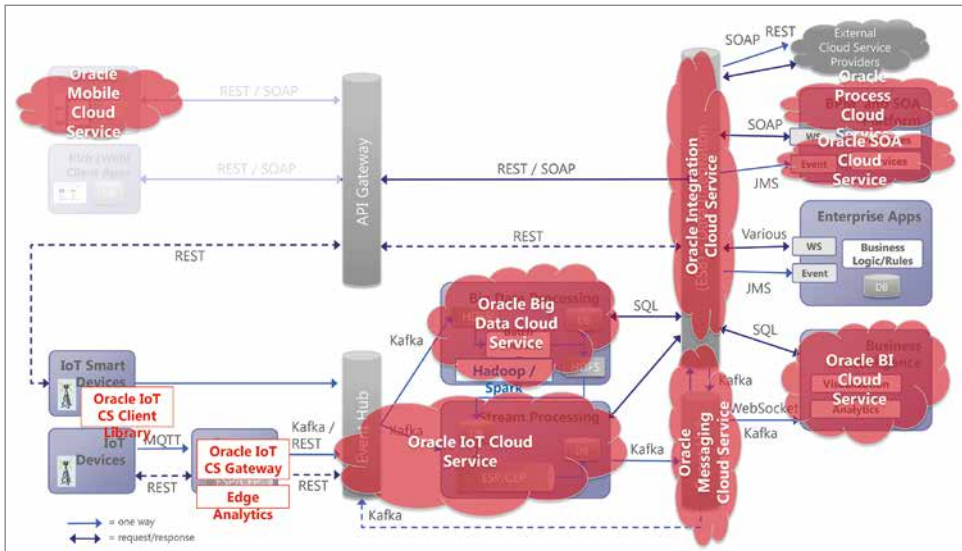


Abbildung 9: Oracle Cloud Services und Internet of Things

PI betreiben lässt. Damit lassen sich erste Filter, Aggregationen oder aber ein Event-Pattern-Matching direkt auf dem Gateway erledigen, sodass sich die Datenmenge, die es ins Backend zu übertragen gilt, entsprechend reduziert. Dies wird auch als „Edge Computing“ beziehungsweise „Fog Computing“ bezeichnet.

Nachteil einer solchen Erweiterung ist jedoch, dass man im Backend nicht mehr die Roh-Informationen des Sensors kennt, sondern nur noch die abgeleiteten Informationen vom Gateway. Rohdaten sind jedoch oft Grundlage für das Machine-Learning im Big-Data-Umfeld; das wird im nächsten Schritt thematisiert.

Erweiterung um eine Big-Data-Infrastruktur für Batch-Analysen

Mit der bisherigen Ergänzung der Architektur um Stream Processing (sowohl zentral wie auch auf dem Gateway) und Event Hub ist sichergestellt, dass die Ereignisse von den IoT-Devices rasch und zuverlässig verarbeitet werden können. Die Analyse beschränkt sich allerdings auf die aktuellen, gerade eintreffenden Daten. Um auch eine Analyse über historische Daten zu ermöglichen, kann die Architektur, wie in *Abbildung 8* gezeigt, um ein Big-Data-Processing ergänzt werden. Eine Infrastruktur dafür sollte zurzeit die Komponenten aus dem Hadoop- und Spark-Ökosystem unterstützen.

Oracle bietet mit der Big Data Appliance eine Möglichkeit, eine Hadoop-beziehungsweise Spark-Plattform als Appliance zu nutzen, also inklusive der Hardware. Wie in *Abbildung 8* dargestellt, konsumiert die Big-Data-Plattform ebenfalls die Ereignisse aus dem Event Hub, jedoch lediglich mit dem Ziel, diese als Rohdaten im Hadoop Distributed Filesystem (HDFS) für die spätere Nutzung abzuspeichern. Diese Daten dienen dann als Input für komplexe Analysen und Verarbeitungen. Dies sind immer Batch-Verarbeitungen, die vom Hadoop-Cluster parallel ausgeführt werden. Die entsprechenden Resultate, etwa ein Modell aus dem Machine-Learning, können dann zum Beispiel auch aus dem Stream Processing verwendet werden, um in Echtzeit die Ereignisse gegen das Modell zu testen (etwa für Predictive Analytics Use Cases).

Oracle Cloud Services und Internet of Things

Die Architektur lässt sich wie gezeigt im eigenen Data Center (On-Premise) aufbauen, es gibt aber auch entsprechende Cloud-Dienste von Oracle, mit denen sich das Ganze ebenfalls umsetzen lässt. *Abbildung 9* zeigt das Oracle-Cloud-Angebot auf die Architektur abgebildet.

Mit Ausnahme der Devices und Gateways können alle Funktionalitäten auch als Cloud-Dienst genutzt werden. Dabei ist natürlich auch eine Kombination von Cloud-

Dienst und eigenem Data Center denkbar. So kann der Oracle IoT Cloud Service für den Empfang und die Filterung der Nachrichten genutzt werden. Die Ergebnisse könnte man dann über den Oracle Integration Cloud Service in das eigene Data Center leiten, um sie den Backend Services zur Verarbeitung zuzuführen.

Fazit

Es lassen sich mit einer bestehenden Infrastruktur sehr wohl erste IoT-Anwendungen umsetzen. Dabei ist allerdings wichtig, sich jederzeit über das zu erwartende zusätzliche Nachrichtenvolumen im Klaren zu sein und die Grenzen der eingesetzten Komponenten zu kennen. Es gibt heute keine einzelne Architektur, die allen Anforderungen von IoT gerecht werden kann – zu vielfältig sind die Anwendungsfälle.

Gefordert ist eine modulare und skalierbare Architektur, die es ermöglicht, je nach Bedarf zusätzliche Komponenten hinzuzufügen. In den letzten Jahren sind einige neue, spezialisierte Produkte im Big-Data- und Fast-Data-Umfeld entstanden, um das Nachrichtenvolumen einer IoT-Lösung zu bewältigen. Diese lassen sich vor eine bestehende Anwendungslandschaft schalten; damit werden die Nachrichten auf eine Menge reduziert, mit der die bestehenden Backend-Systeme umgehen können. Entsprechende Angebote gibt es dabei sowohl als Cloud-Dienst als auch für die eigene Backend-Infrastruktur.




Guido Schmutz
guido.schmutz@trivadis.com

Alle Ausgaben auf Ihr Smartphone und Tablet – ganz bequem mit einer App! Die neue IDG Media Kiosk App

- Eine App für alle Publikationen: COMPUTERWOCHE, CIO, TecChannel, ChannelPartner
- Erste Ausgabe gratis testen
- Inklusive multimedialen Inhalten, Bilderstreifen und weiterführenden Links
- Duales Layout mit interaktivem Lesemodus
- App erhältlich für    



 0711-72 52 276

 www.idgmediakiosk.de/info



COMPUTERWOCHE, CIO, TecChannel und ChannelPartner erscheinen im Verlag IDG Business Media GmbH, Lyonel-Feiningger-Str. 26, 80807 München, Registergericht München, HRB 99187, Geschäftsführer: York von Heimburg. Die Kundenbetreuung erfolgt durch den Kundenservice, ZENIT Pressevertrieb GmbH, Postfach 810580, 70522 Stuttgart, Geschäftsführer: Joachim John, Tel. 0711/72 52 276, Fax 0711/72 52 377, E-Mail: shop@cio.de, shop@computerwoche.de, shop@tecchannel.de, shop@channelpartner.de.



Die ersten Schritte in die IoT-Welt

Philipp Wang, OPITZ CONSULTING Deutschland GmbH

Die Digitalisierung und nun das Internet of Things (IoT) stellen vermehrt frühere Geschäftsmodelle infrage. Für viele Unternehmen erschließen sich darüber hinaus neue Umsatzbereiche außerhalb ihrer angestammten Geschäftsfelder, zum Beispiel durch Kooperationen mit jungen Unternehmen [1]. Gerade Startups können sich mit neuen Ideen als gefährliche Herausforderer für die Geschäftsmodelle etablierter Unternehmen herausstellen [2].

Mit dem Oracle IoT Cloud Service (IOTCS) steht den Unternehmen ein flexibler, Cloud-basierter PaaS für den schnellen Einstieg in die IoT-Welt zur Verfügung. Startups und größere Unternehmen können sich ganz auf die Realisierung ihrer Ideen konzentrieren, da sie sich keine Gedanken mehr über eine geeignete IoT-Infrastruktur und die Verfügbarkeit machen müssen. Dieser Artikel zeigt anhand eines Beispiels einige anfängliche Handgriffe für die Kommunikation eines Geräts mit dem IOTCS und demonstriert, wie der Cloud Service den Nutzern die Analyse der ankommenden Daten erleichtert.

Das Internet der Dinge

Zunehmend werden einst geschlossene Einheiten wie Haushaltgeräte, Fahrzeuge und mit Sensoren ausgestattete Geräte vernetzt. Deren gesendete Daten müssen

anschließend verarbeitet und analysiert werden, um neues Wissen über die Nutzung zu erhalten [3]. Zurzeit konzentrieren sich noch über 50 Prozent der IoT-Aktivitäten auf die Bereiche „Herstellung“, „Transportwesen“, „Smart Cities“ und „Konsumgüter“. Bis zum Jahr 2020 erwarten viele IT-Unternehmen jedoch auch massive Auswirkungen auf den Retail-Bereich, Utilities und den öffentlichen Sektor [4, 5].

Aufgrund der hohen Anzahl der angeschlossenen Geräte und der großen Datenmengen entstehen hohe Anforderungen an das Back-End. Der IOTCS unterstützt Unternehmen bei ihrem Weg in die IoT-Welt, indem er eine stabile Plattform, Bibliotheken für die Integration und viele Analyse-Tools direkt in einer Web-Oberfläche zur Verfügung stellt. Er wurde für die Verarbeitung sehr großer Datenmengen entwickelt und kann für Real-Time-Analysen der ankommenden Daten eingesetzt werden [6]. Als Szenario dient die fiktive Verleihung ei-

nes namhaften Musikpreises. Der Ausrichter möchte während der Veranstaltung einen Publikumspreis vergeben.

Exemplarisches Szenario für den IoT Cloud Service

Während der Verleihung wird dem Publikum im Saal über kleine Geräte mit Display („Wahlmaschinen“) an ihren Sesseln sowie den Zuschauern im Fernsehen und im Internet über Twitter erlaubt, für einen Künstler aus einer vorher festgelegten Liste zu stimmen. Der Künstler mit der höchsten Anzahl an Stimmen erhält am Ende der Veranstaltung den Publikumspreis.

Der Nutzer im Saal sieht auf dem Display seiner Wahlmaschine eine Liste von Künstlern, aus der er eine Person auswählt und dieser seine Stimme gibt. Diese Information wird dann an den IOTCS zur weiteren Speicherung und Verarbeitung geschickt.

Im Fernsehen und im Internet werden die Zuschauer gebeten, ihre Stimme per Twitter-Tweet abzugeben. Sie sollen dabei einen bestimmten Hashtag verwenden.

Zur Vereinfachung des Szenarios kommt eine kleine Scala-Anwendung zum Einsatz, die das Abgeben der Stimmen sowohl mithilfe der Wahlgeräte als auch auf Twitter simuliert [7]. Sie sucht in den Tweets im Twitter-Stream nach einem String in einem bestimmten Format, sammelt diese Tweets und leitet sie an den IOTCS weiter. Mit diesen Daten lässt sich zeigen, wie die Oberfläche des IOTCS den Nutzer bei der Anbindung eines Geräts unterstützt.

Anschließend wird ein Suchmuster auf diese Daten angewendet. Sobald ein bestimmter Schwellenwert bezüglich der Menge an Stimmen für einen Künstler erreicht wird, wird über den Integration Cloud Service (ICS) ein Prozess im Process Cloud Service (PCS) aufgerufen [8]. Dieser Prozess leitet automatisch die weitere Kommunikation mit dem Künstler ein (siehe Abbildung 1).

Ein Gerät mit dem IOTCS verbinden

Für die Anbindung eines Wahlgeräts an das IOTCS sind nur wenige Schritte notwendig. Zuerst muss ein neues Gerät registriert werden. Dazu wird dieses entweder manuell über ein Browser-Interface oder per Batchmodus (für mehrere Geräte) registriert. Für die Sicherheit wird später das Feld „Shared Secret“ mit einer gerätespezifischen ID für den Verbindungsaufbau mit der IOTCS verwendet. Nach der Registrierung wird das Gerät in der Geräteliste als „Registered“ mit dem Typ „not activated“ aufgelistet.

Für die Kommunikation zwischen Gerät und Cloud Service muss man ein Device Model definieren. Dieses enthält Spezifikationen unter anderem zum Nachrichtenformat und den verfügbaren Web-Ressourcen sowie weitere Metadaten zum Gerät. In diesem Szenario ist das sehr einfach: Das Device Model enthält nur ein Attribut vom Typ „String“ namens „artistName“. Zudem könnte man auch Callback-Actions für Prozesse, die aus der Perspektive „process 2 device“ kommunizieren, definieren. Um das Beispiel einfach zu halten und nicht den Rahmen zu sprengen, liegt der Fokus auf „device 2 process“ (siehe Abbildung 3).

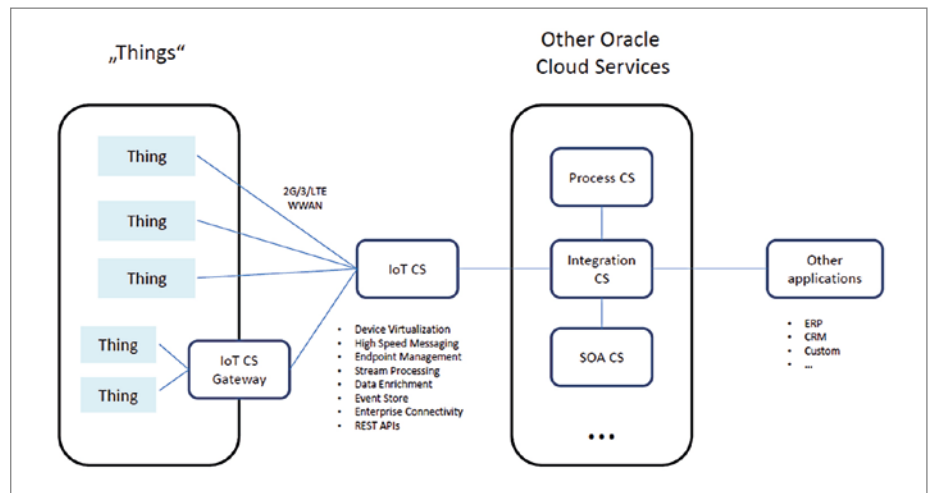


Abbildung 1: Oracle IoT Cloud Service

Abbildung 2: Registrierung eines neuen Geräts

Name	Description	Type	Range	Alias	Access
artistName		String			Yes

Abbildung 3: Definieren des Device Model

Der Sicherheitsaspekt spiegelt sich in der Client-Implementierung wider. Um Daten zur Cloud zu schicken, muss die Java-Klasse, die hier ein Gerät repräsentiert, eine Aktivierungsanfrage an den IOTCS schicken. Diese beinhaltet die Geräte-ID und das „Shared Secret“, die sicher in einem Trusted Keystore gespeichert wurden. Der Truststore wird mit einer Befehlszeile erstellt (siehe Listing 1).

```
java -cp lib/device-library.jar com.oracle.iot.client.impl.trusted.TrustedAssetsProvisioner -serverHost url_to_cloud_instance -serverPort 7102 -truststore config/trustStore.jks -truststorePassword changeit -taStorePassword changeit -sharedSecret <SHARED_SECRET> -deviceId <DEVICE_ID>
```

Listing 1

Nach erfolgreicher Aktivierung erhält das Gerät einen privaten Schlüssel für die weitere Kommunikation, der ebenfalls im Truststore gespeichert wird. Der Code-Ausschnitt in Listing 2 zeigt, wie das IOTCS-API in der Client-Implementierung verwendet wird. Der Status des Geräts verändert sich zu „Activated“, wenn die Anwendung gestartet wurde und das Gerät die erste Anfrage an das IOTCS gestellt hat. Nun ist das Gerät bereit für die Aufzeichnung von Tweets (siehe Abbildung 4).

```
public void connect() throws GeneralSecurityException, IOException {
    // Create the device client instance
    device = new DirectlyConnectedDevice();

    // Activate the device
    if (!device.isActivated()) {
        device.activate(TWITTER_MODEL_URN);
    }

    // Create a virtual device implementing the device model
    DeviceModel deviceModel =
        device.getDeviceModel(TWITTER_MODEL_URN);

    virtualDevice = device.createVirtualDevice(device.getEndpointId(), deviceModel);
    ...
}
```

Listing 2

Erstellung einer Exploration auf einen Datenstrom

Erfolgreich an das IOTCS versendete Daten sind über die Browser-Oberfläche unter „Data and Explorations“ sichtbar. Die aufgelisteten Datensätze enthalten einen Timestamp und die empfangenen Daten. In Abbildung 5 ist ersichtlich, dass

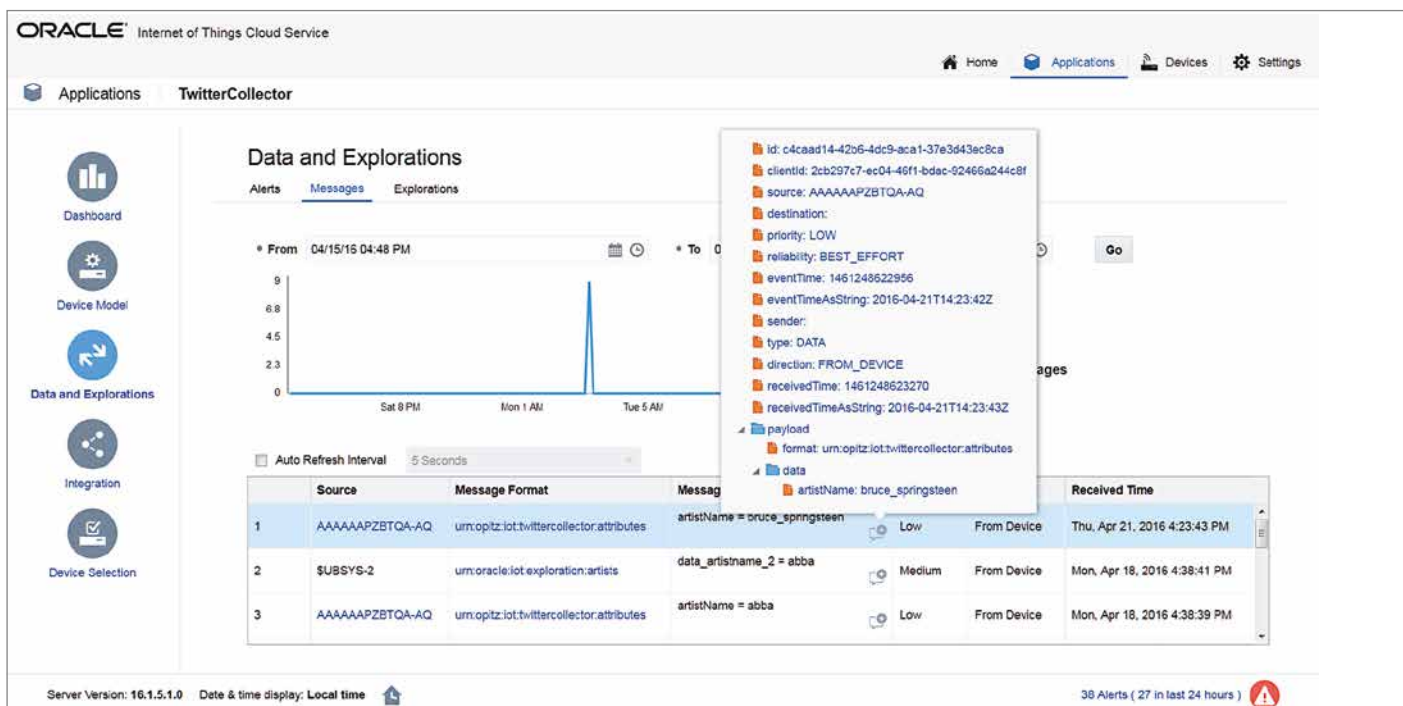


Abbildung 4: Die Übersichtsseite

ein Künstler namens „bruce_springsteen“ Stimmen erhielt.

Mithilfe der Oberfläche ist die Suche nach Künstlernamen in den gespeicherten Daten möglich. Dazu wird in diesem Fall ein Suchmuster definiert, indem man beispielsweise einen Schwellenwert bezogen auf die Anzahl der erhaltenen Stimmen in einem bestimmten Zeitraum festlegt. Dieser kommt anschließend bei der Event-Analyse oder beim sogenannten „Stream-Processing“ zur Anwendung. Auf der Seite „Data and Explorations“ fügt man eine neue „Exploration

Source“ hinzu. Der Name ist „Artist names“. Diese neue Source wird mit dem Device Model für diese Anwendung verknüpft (siehe Abbildung 6). Der neuen Source wird nun eine sogenannte „Exploration“ hinzugefügt. Beim Öffnen dieser Exploration öffnet sich eine Dialogbox, die den Nutzer bei der Erstellung eines Suchmusters unterstützt. In Abbildung 7 sieht man die auf den Datenstrom angewendeten Filterkriterien, mit denen man nach dem Ereignis „Mindestens drei Wählerstimmen für einen Künstler innerhalb von 60 Sekunden“ sucht.

In der unteren Tabelle kann man sofort sehen, ob die Suchfilter angewendet wurden. Das System analysiert dazu die ankommenden Datenströme, filtert diese und zeigt davon nur diejenigen an, die die Suchkriterien erfüllen. Auf dem gleichen Weg kann man auch Aggregations-Funktionen, typische Funktionen wie die TOPN-Funktion oder Muster zur Erkennung von W-Formationen in Charts einsetzen. Auch das Nicht-Eintreffen eines Ereignisses innerhalb eines bestimmten Zeitraums ist für die Suche verfügbar.

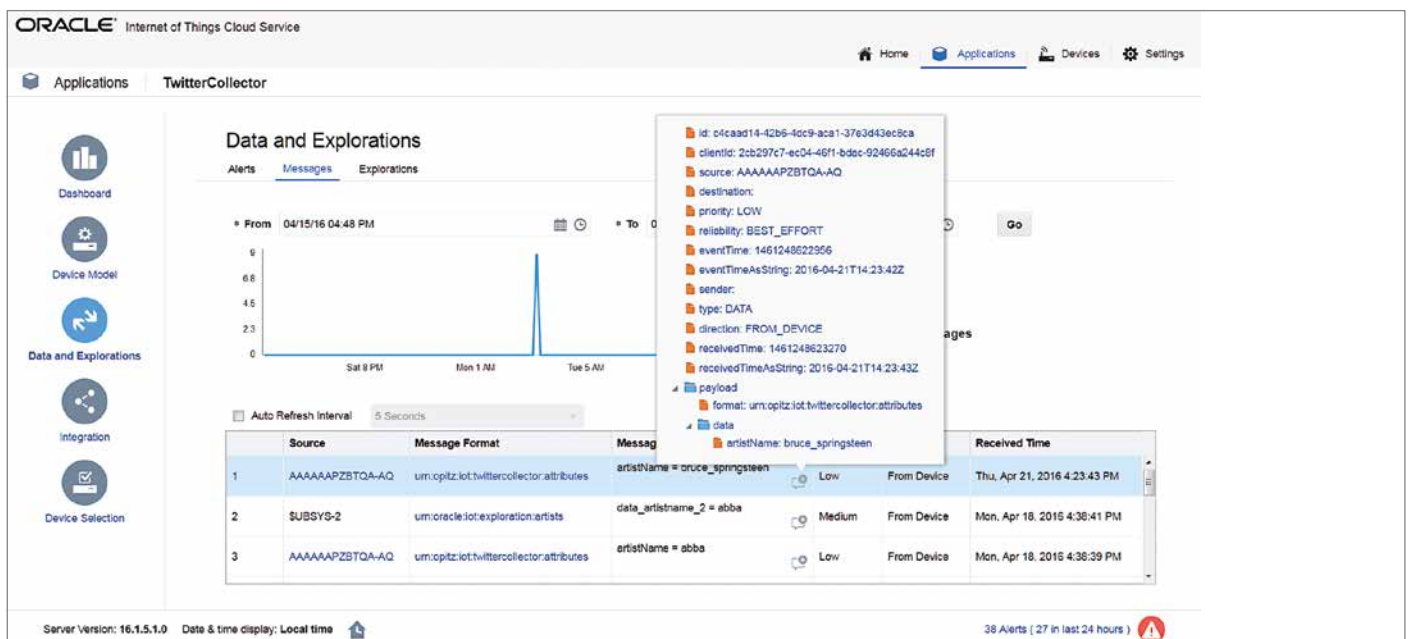


Abbildung 5: Data und Explorations

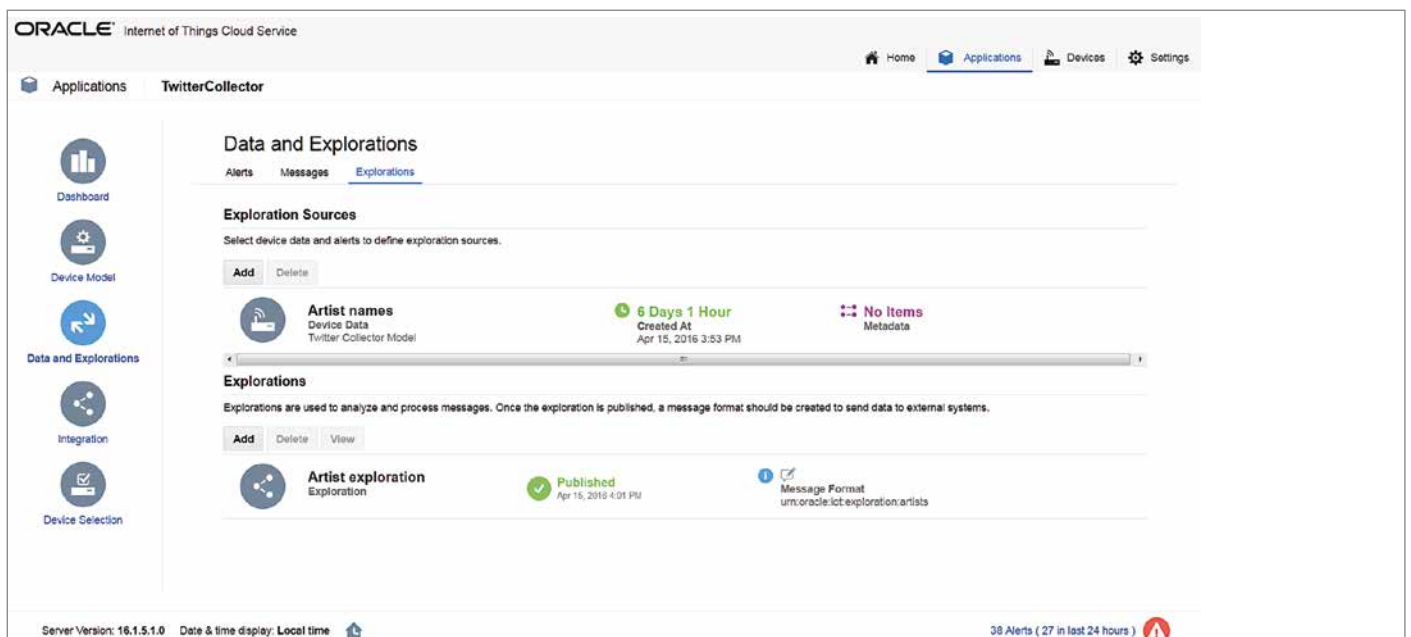


Abbildung 6: Darstellung der bisher erstellten Exploration Sources und Explorations

Weiterverarbeitung der Ergebnisse im ICS

Den gefilterten Datenstrom können wir nun weiter zur externen Verarbeitung an einen REST-Endpoint weiterleiten, der durch den Oracle Integration Cloud Service (ICS) bereitgestellt wird. Dieser Service verbindet sowohl zwei Cloud-Anwendungen als auch Cloud-Anwendungen mit On-Premise-Anwendungen unter anderem per SOAP oder REST. Mit seiner Hilfe werden die Daten ei-

ner Exploration nach außen weitergeleitet, was eine flexible Architektur mit Fokus auf lose Kopplungen zwischen den Endpoints externer Systeme ermöglicht [9]. Dazu ist zuerst eine „Integration“ im IOTCS zu erstellen. Dafür sind nur der Name für die Integration, die URL des REST-Endpoints und die Exploration, die verwendet wird, erforderlich.

Wenn das Publikum im Saal per Gerät und die Internetnutzer über Twitter mindestens drei Mal innerhalb einer Minute für einen bestimmten Künstler stimmen,

wird diese Information aus dem ankommenden Datenstrom herausgefiltert, live als Eintrag in der Tabelle (siehe Abbildung 8) dargestellt und gleichzeitig an einen REST-Endpoint der ICS weitergeleitet.

Ein Nutzer ist somit durch die Web-Oberfläche in der Lage, ohne das Schreiben einer einzelnen Codezeile weitere Anwendungen in den Workflow einzubinden. Der ICS verfügt dazu über eine große Anzahl vordefinierter Adapter, die nach zusätzlicher Konfiguration einen Datenaustausch mit Produkten anderer Hersteller ermöglichen.

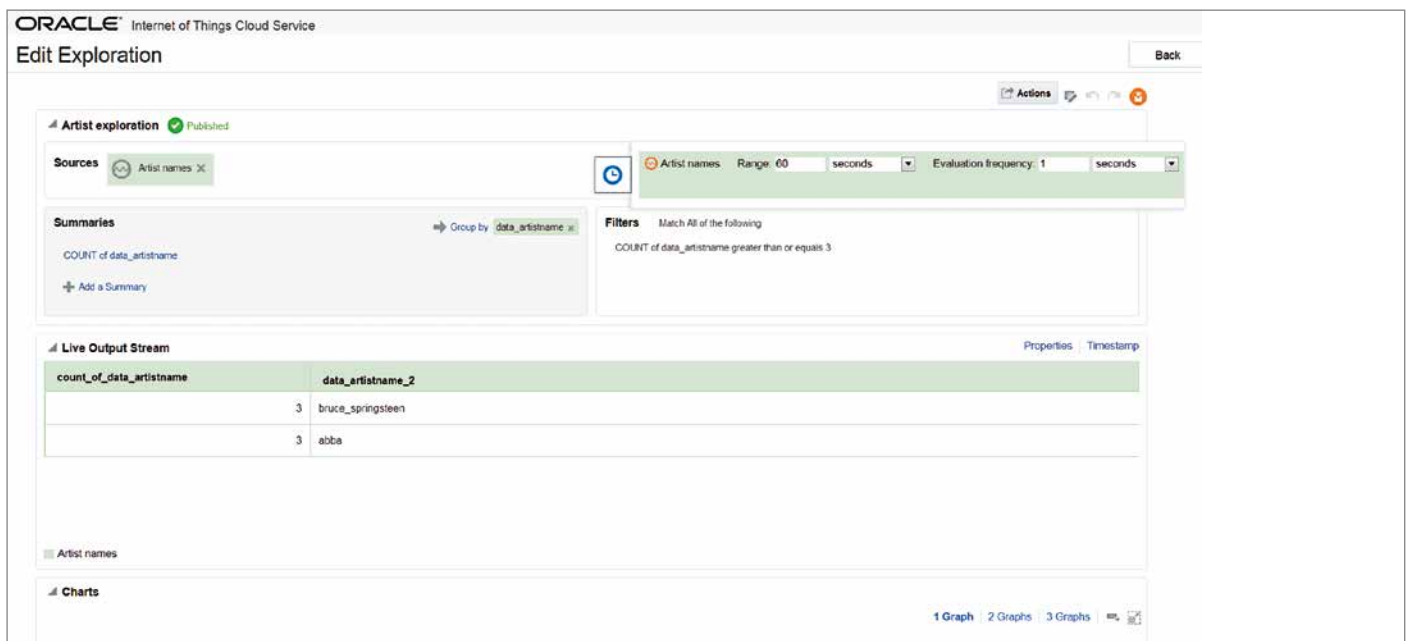


Abbildung 7: Bearbeitung einer Exploration

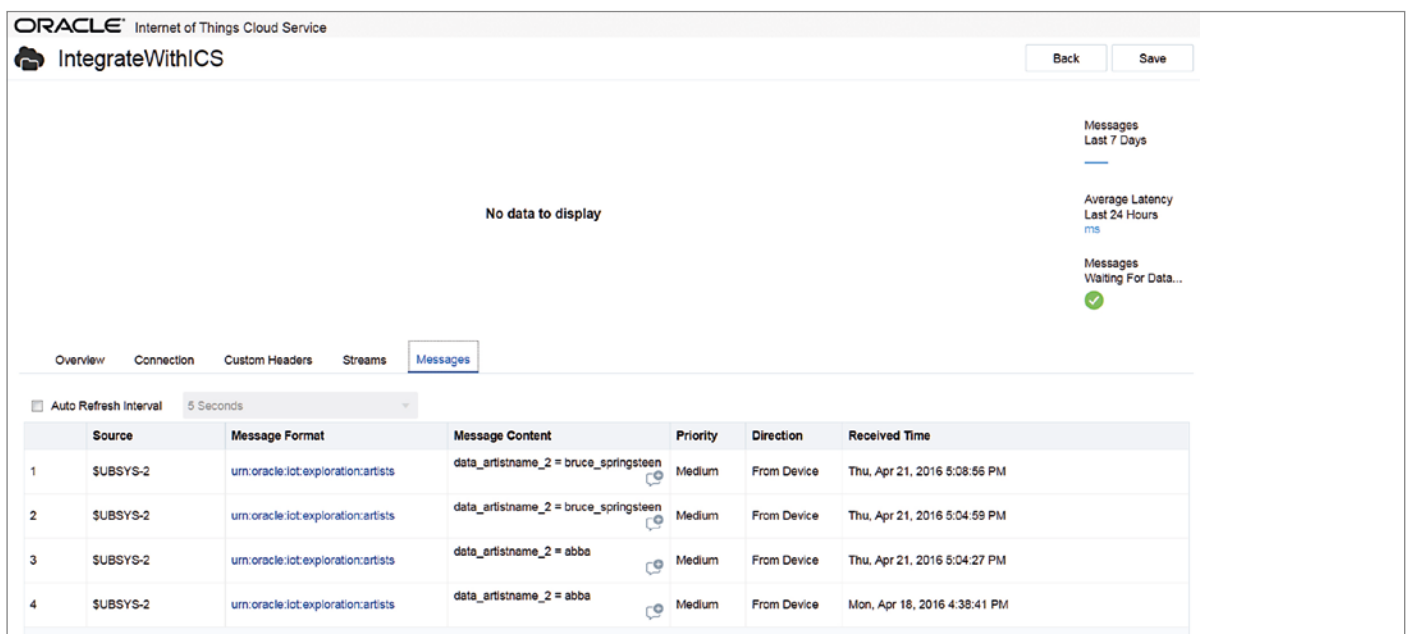


Abbildung 8: Darstellung der beobachteten Datenquellen im ICS

Fazit

Die ständige Datengenerierung durch IoT-Geräte bedingt eine neue Herangehensweise in den Unternehmen. Die saubere Aufbereitung und die Analyse der Daten gewinnt an Bedeutung. Der IOTCS unterstützt Unternehmen nicht nur bei der Anbindung von IoT-Geräten, sondern auch im Pre-Integration-Stage bei der Überwachung geschäftsrelevanter Ereignisse und der Verbesserung der Datenqualität [10]. Aufgrund seiner Schnittstellen nach außen ist er einfach in eine vorhandene IT-Landschaft zu integrieren und umfasst an sich alle Werkzeuge, die ein Unternehmen für seinen Weg in die IoT-Welt benötigt.

Die Geräte können durch die Einbindung der IOTCS-Bibliotheken direkt mit dem Service kommunizieren. Auch weniger intelligente Geräte werden durch die Verwendung eines IOTCS-Gateways eingebunden, das eine Vielzahl von Kommunikationstechnologien wie Bluetooth LE, Zigbee, Z-Wave etc. unterstützt. Der

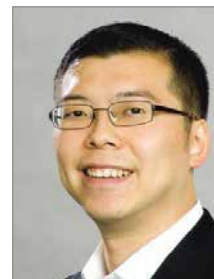
IOTCS ermöglicht nicht nur das Verwalten einer großen Anzahl von Geräten und Gateways, sondern auch die Verarbeitung großer Datenmengen und die Real-Time-Analyse durch den Einsatz verfügbarer Funktionen.

Ohne irgendeine Code-Implementation stehen den Nutzern direkt über die Browseroberfläche viele Funktionen der Fast-Data- und Big-Data-Technologien zur Verfügung, sodass sie mit dieser Cloud-Plattform dynamisch neue Ideen und Lösungen entwickeln können.

Quellen

- [1] <http://iotbusinessnews.com/2016/03/15/31645-mobilock-worlds-first-bike-lock-with-iot-connection-based-on-lora-technology/>, Zugriff Juni 2016
- [2] <http://www.proglove.de/>, Zugriff Juni 2016
- [3] S. Sinha: Making Big Data Work for Your Business. Impact Publishing 2014
- [4] http://internetofeverything.cisco.com/sites/default/files/docs/en/ios_public_sector_vas_white%20paper_121913final.pdf, Zugriff Juni 2016

- [5] <http://www.oracle.com/us/solutions/internetofthings/iot-asset-tracking-brief-2890501.pdf>, Zugriff Juni 2016
- [6] <https://cloud.oracle.com/iot>, Zugriff Juni 2016
- [7] <https://github.com/lucasjellema/aced-cloud-demo-ofmw2016-valencia/tree/master/loTCS-Twitter/twitter-collector>, Zugriff Juni 2016
- [8] <https://cloud.oracle.com/process>, Zugriff Juni 2016
- [9] http://docs.oracle.com/cloud/latest/intcs_gs/, Zugriff Juni 2016
- [10] Bial, D. / Scheuch, R.: IoT erfordert neues Denken. BI-Spektrum 01-2016
- [11] <https://community.oracle.com/docs/DOC-996733>, Zugriff Juni 2016



Philipp Wang

philipp.wang@opitz-consulting.com

Oracle bringt SPARC in die Cloud

Oracle kündigt wesentliche Erweiterungen seiner SPARC-Plattform an. Ab sofort sind die einzigartigen Sicherheitsfeatures, die Effizienz und die Einfachheit von Oracle SPARC auch in der Cloud verfügbar. Auf Basis des neuen SPARC-S7-Mikroprozessor bietet die SPARC-Plattform neue Cloud Services, Engineered Systems und Server.

Die Neuerungen der SPARC-Plattform basieren auf dem 4.27 GHz, 8-Core/64-Thread, SPARC-S7-Mikroprozessor mit „Software in Silicon“-Eigenschaften wie „Silicon Secured Memory“ und „Data Analytics Accelerators“. Dieser bietet den branchenweit höchsten Level an Effizienz pro Core und ermöglicht es Unternehmen, Anwendungen jeglicher Größe auf dieser SPARC Plattform zu betreiben – zum Preis herkömmlicher Commodity Plattformen. Alle kommerziellen und kundenspezifischen Anwendungen laufen auf den neuen SPARC-Produkten ohne Änderung, aber mit signifikanten Verbesserungen im Hinblick auf Sicherheit, Effizienz und Einfachheit.

Die neuen, auf dem SPARC-S7-Prozessor basierenden Cloud Services und Systeme bieten das Preisniveau von x86-Servern sowie dank „Software in Silicon“-Technologie herausragende Enterprise-Class-Funktionalitäten im Bereich „Sicherheit und Analytics“. Sie umfassen neue Oracle-Cloud-Compute-Plattform-Services, das MiniCluster S7-2 Engineered System und die SPARC S7 Server. Diese neuen Produkte sind so konzipiert, dass sie sich nahtlos in existierende Infrastrukturen integrieren lassen und über voll integrierte Virtualisierung und Verwaltung für die Cloud verfügen.

Der neue SPARC Cloud Service, nun ein Teil der SPARC-Plattform, ist ein dedizier-

ter Compute Service, der Unternehmen eine einfache, sichere und effiziente Compute Plattform in der Cloud bereitstellt. Sie erweitert das umfassende Cloud-Service-Portfolio, das Organisationen hilft, leistungsstarke Anwendungen schnell zu entwickeln und in Betrieb zu nehmen oder Oracle Cloud-Anwendungen zu erweitern – auf einer „Enterprise ready“-Cloud-Plattform. Um Sicherheits- und Leistungsvorteile der SuperCluster Engineered Systeme auch im Midsize-Computing-Markt bereitzustellen, hat Oracle zudem den MiniCluster S7-2 entwickelt. Weitere Informationen dazu in der nächsten Ausgabe.

Oracle und Industrie 4.0 – zurück in die Zukunft

Gerhild Aselmeyer, Analysen, Konzepte & Anwendungsentwicklung für EDV

Im Rahmen von Industrie 4.0 könnte zukünftig häufiger mal die Anforderung nach Kenntnissen älterer Oracle-Versionen (7 bis 9) gestellt werden. Aufgrund der sehr viel längeren Lebenszyklen industrieller Anlagen und Maschinen werden diese häufig auf Industrie-Rechnern mit älteren Betriebssystemen als Bedienerkonsolen betrieben. Produktionsdaten lassen sich auch unter diesen Umständen automatisch in eine aktuelle Oracle-Datenbank übertragen, wenn man gewisse Besonderheiten beachtet.

Dieser Artikel zeigt, welche Möglichkeiten die Oracle-Software bietet, um Rechner mit geringerer Hardware-Ausstattung und älterem Betriebssystem (DOS/ Windows NT/ 2000/ XP) als Client für eine aktuelle Oracle-Datenbank einzusetzen. Grundlegend dafür sind die von Oracle gepflegte Rückwärts-Kompatibilität sowie die Werkzeug-Konstanz für Netzwerk (SQL*Net) und Programmierung (Precompiler PRO*C, PRO*Cobol und OCI). Außerdem unterstützt Oracle ein Betriebssystem über dessen gesamten Lebenszyklus mit neuen Versionen. Die Client-Version 10g hat die offizielle Freigabe; für die Professional- und Enterprise-Editionen Windows 2000, 11g ab Windows XP und für 12c benötigt man mindestens Windows 7.

Eines gilt es bei den nachfolgenden Ausführungen zu beachten: Bei den Clients 9i oder älter handelt es sich um nicht offiziell von Oracle unterstützte Versionen für Datenbanken unter neuer Server-Software (11g/12c). Die installierten Clients lassen sich (ausschließlich) für die angegebenen Operationen unter Beachtung der beschriebenen Beschränkungen nutzen.

Welche Oracle-Version für welches Betriebssystem

Betrachten wir zunächst ältere Betriebssysteme, die Hardware-Voraussetzungen und die jeweils letzten dafür freigegebenen Versionen der Oracle-Client-Software (siehe Tabelle 1). Neueren Windows-Betriebssystemen braucht keine besondere Aufmerksamkeit geschenkt zu werden, da sich unter diesen Client-Software von Versionen einsetzen lässt, die keine Kompatibilitätsprobleme erwarten lassen. Die Interoperabilität ist laut der von Oracle veröffentlichten Matrix gegeben, auch wenn die älteren einsetzbaren Versionen nur noch bedingt unterstützt werden. Damit erhält man zwar keine Fehler-Korrekturen, doch ist es unwahrscheinlich, dass

Fehler bei einfachen Datentransfers (SQL, DML) bei korrekter Konfiguration von SQL*Net überhaupt auftreten.

Für neuere Linux-Kernel (ab 2.4.21 / glibc 2.2.4) als RHEL- oder SuSE-Distribution steht bereits die Client-Software 9.2.0.4 oder neuer zur Verfügung. Allerdings gelten für einige Pakete besondere Anforderungen an deren Version – ob auch bei einer reinen Client-Installation, hat die Autorin bisher nicht getestet. An-

triebsystemen braucht keine besondere Aufmerksamkeit geschenkt zu werden, da sich unter diesen Client-Software von Versionen einsetzen lässt, die keine Kompatibilitätsprobleme erwarten lassen. Die Interoperabilität ist laut der von Oracle veröffentlichten Matrix gegeben, auch wenn die älteren einsetzbaren Versionen nur noch bedingt unterstützt werden. Damit erhält man zwar keine Fehler-Korrekturen, doch ist es unwahrscheinlich, dass

Fehler bei einfachen Datentransfers (SQL, DML) bei korrekter Konfiguration von SQL*Net überhaupt auftreten.

Für neuere Linux-Kernel (ab 2.4.21 / glibc 2.2.4) als RHEL- oder SuSE-Distribution steht bereits die Client-Software 9.2.0.4 oder neuer zur Verfügung. Allerdings gelten für einige Pakete besondere Anforderungen an deren Version – ob auch bei einer reinen Client-Installation, hat die Autorin bisher nicht getestet. An-

Betriebssystem	Hardware-Voraussetzungen	Oracle-Version
DOS/ Windows 3.1 Windows NT 3.5.1 (16-bit)	Prozessor 80486 RAM 8 MB Platte < 300 MB	7.3.4 (16-bit Required Support Files)
Windows NT 4.0 SP1 - SP4	Prozessor mindestens: 80486 empfohlen: Pentium um 133/ 166 RAM mindestens: 32 MB empfohlen: 54 MB Platte < 300 MB	8i / 8.1.5
Windows NT 4.0 SP5	Prozessor mindestens: Pentium 166/ 200 empfohlen: Pentium 266 RAM mindestens: 128 MB empfohlen: 256 MB Platte FAT: 450 MB (Runtime) NTFS: 200 MB	9.2.0.1
Linux Kernel 2.2	Prozessor X86 RAM 64 MB Platte (Programmer) 275 MB	8i / 8.1.7

Tabelle 1

dererseits hat sie im Zusammenhang mit Industrie-Rechnern noch nichts von Linux als Betriebssystem gehört. Unabhängig von Betriebssystem und Oracle-Client-Version gehören zu den notwendigen Hard- und Software-Voraussetzungen die Netzwerk-Fähigkeit des Rechners/der Bedienkonsole, also die Existenz einer Netzwerkkarte und Unterstützung von TCP/IP.

Was sich kombinieren lässt

Die Frage nach den möglichen Kombinationen aus Oracle-Client- und Server-Versionen/-Editionen muss sowohl technisch als auch lizenzrechtlich betrachtet werden. Bei der technischen Interoperabilität von Versionen benötigen die Editionen (XE, PE, SE1, SE, EE) keine gesonderte Behandlung, da die Code-Basis für alle identisch ist; lizenzrechtlich hingegen müssen die Bedingungen für einen weiteren Client gemäß den Bestimmungen für die einzelnen Editionen beachtet werden. Zunächst die technischen Voraussetzungen: *Abbildung 1* zeigt die in My Oracle Support veröffentlichte Matrix für die Unterstützung zur Interoperabilität.

Es gibt folgende generelle Anmerkungen, die Oracle zur Interoperabilität macht:

- Für Datenbank-Links muss die Interoperabilität in beide Richtungen gewährt sein.
- Nicht unterstützte Kombinationen können funktionieren. Fehlermeldungen hierzu werden aber nicht bearbeitet.
- Da neuere Datenbank-Server mit einem begrenzten Set älterer OCI-Clients kompatibel sind, muss die Client-Software nicht unbedingt gleichzeitig mit der Server-Software ein Update erfahren. Allerdings lassen sich neue Eigenschaften der Server-Software dann nicht nutzen.
- Die Matrix gilt auch über verschiedene Plattformen und Prozessor-Releases (32/64-Bit) hinweg, außer wenn die von Oracle veröffentlichte Notiz zur Unterstützung einer Plattform oder eines Prozessors etwas anderes besagt.

Im Folgenden gilt das Hauptaugenmerk den Client-Versionen, die in der Mat-

rix für Datenbankserver der Versionen 11g/12c mit „No“ gekennzeichnet sind, da hierfür nie Unterstützung von Oracle geboten wurde. Die übrigen Markierungen geben lediglich den Stand der von Oracle zum gegenwärtigen Zeitpunkt noch geleisteten Unterstützung wieder; es waren oder sind aber von Oracle freigegebene Kombinationen.

Auf den ersten Blick scheinen die Entwickler bei Oracle von den bisher beobachteten Prinzipien der Rückwärts-Kompatibilität bei der neuesten Version 12c abzuweichen, wenn man „#8“ in der Legende zu *Abbildung 1* liest. Ein Blick auf die Beschreibung der Parameter für SQL*Net in der „sqlnet.ora“ lässt die Hoffnung zu, dass zumindest Clients bis zurück zu 8i (8.1.7) konfigurierbar sind:

- SQLNET.ALLOWED_LOGON_VERSION (ab 10g)
- In 12c ersetzt durch
- SQLNET.ALLOWED_LOGON_VERSION_SERVER
- SQLNET.ALLOWED_LOGON_VERSION_CLIENT

Ob und welche Probleme mit noch älteren Client-Versionen auftreten, hat die Autorin bisher nicht verifiziert. In diesem Fall lässt sich mithilfe einer Gateway-Datenbank der Version 11g der Datentransport bewerkstelligen; die Verbindung von 11g- und 12c-Datenbanken über Datenbank-Link wird nämlich unterstützt. Mit Gateway-Datenbanken zur Überbrückung von Versionen beim Einsatz von Datenbank-Links hat die Autorin bereits positive Erfahrungen gemacht. Aber auch bei der Konfiguration von SQL*Net für eine 11g-Datenbank müssen spezielle Einstellungen für ältere, nicht offiziell unterstützte Client-Anwendungen vorgenommen beziehungsweise überprüft werden. Das betrifft vor allem die Datei „sqlnet.ora“.

Da mit Version 8i eine Zuverlässigkeitsprüfung („Authentisierung“) bei der Netz-Kommunikation eingeführt wurde, gibt es seitdem den Parameter „SQLNET.AUTHENTICATION_SERVICES“, den ältere Clients nicht kennen. Außerdem haben sich die zulässigen Werte von 8i über 9i bis 10g teilweise geändert:

- 8i
beq, none, all, kerberos5, cybersafe, securid, identix

- 9i
beq, none, all, kerberos5, cybersafe, radius
- 10g
beq, none, all, kerberos5, radius, nts

Client Version	Server Version				
	12.1.0	11.2.0	11.1.0	10.2.0	10.1.0
12.1.0	Yes	Yes	ES	LES #7	No #3
11.2.0	Yes	Yes	ES	LES #7	No
11.1.0	ES	ES	ES	LES #7	Was #8
10.2.0	LES #7	LES #7	LES #7	LES	Was
10.1.0(#4)	No	Was #6	Was #6	Was	Was
9.2.0	No #0	Was #5	Was #5	Was #5	Was
9.0.1	No	No	No	No	Was
8.1.7	No	No	No	Was	Was
8.1.6	No	No	No	No	No
8.1.5	No	No	No	No	No
8.0.6	No	No	No	No	No
8.0.5	No	No	No	No	No
7.3.4	No	No	No	No	No

Abbildung 1: Aktuelle Unterstützung für Interoperabilität. Legende: „Yes“ = unterstützt, „ES“ = Bugfixes nur für Kunden mit „Extended Support“, „LES“ = Bugfixes nur für Kunden mit „Limited Extended Support“- oder „Market-Driven Support“-Vertrag, „Was“ = Unterstützung für mindestens eine Seite ausgelaufen, „No“ = keine Unterstützung. Spezielle zusätzliche Anmerkungen: „#3“ = Korrekte Anzeige des Fehlers ORA-3134, „#4“ = Probleme, wenn eine der beiden Seiten auf EBCDIC Characterset basiert, „#5“ = Client-Version min. 9.2.0.4, „#6“ = Patchsets bei Datenbank-Links beachten, „#7“ = Patchsets bei Datenbank-Links beachten, „#8“ = der Zugriffsversuch von Client 9.2 auf eine Datenbank 12.1 schlägt fehl mit der Meldung „ORA-28040: No matching authentication protocol“ error.

Abbildung 2: Passwort-Versionen

Betreibt man den Datenbankserver mit einem Linux-Betriebssystem, dann gibt es für die serverseitige Einstellung von „SQLNET.AUTHENTICATION_SERVICES“ mehrere Möglichkeiten:

- SQLNET.AUTHENTICATION_SERVICES=(all)
- SQLNET.AUTHENTICATION_SERVICES=(beq, all)
- SQLNET.AUTHENTICATION_SERVICES=(nts, all)
- SQLNET.AUTHENTICATION_SERVICES=(none)

Mit „all“ lässt man alle Netz-Authentisierungen zu, die der Server kennt, mit „none“ keine außer der Datenbank-gestützten mit Username/Passwort, „nts“ steht für das proprietäre Verfahren NTLM zur Authentisierung von Microsoft und „beq“ für die native Authentisierung über das Betriebssystem für alle außer Microsoft Windows.

Damit steht für Client-Anwendungen basierend auf älteren Oracle-Clients der Versionen 7.3.x, 8.0.x immer die ihnen bekannte Anmeldeform zur Verfügung.

Läuft der Datenbank-Server mit einem Windows-Betriebssystem, gibt es für Client-Anwendungen der Version 7.3.x/8.0.x nur die Einstellungsmöglichkeit „SQLNET.AUTHENTICATION_SERVICES=(none)“. Alle anderen Einstellungen führen dazu, dass vor der Datenbank-gestützten Anmeldeform eine Authentisierung über NTLM versucht wird; dies bewirkt allerdings auf Client-Seite ein unerwartetes Ende der Kommunikation (ORA-03113), der Datenbank-Prozess beansprucht daraufhin 100 Prozent der Server-CPU und legt diese damit lahm. Bei etwas jüngeren Client-Anwendungen (8i, 9i) kann dann auch „kerberos5“ zum Einsatz kommen.

Ab der Datenbank-Version 10g gibt es weiterhin den Parameter „SQLNET.ALLOWED_LOGON_VERSION“ mit „8“ als

Voreinstellung bis 11g, ab 12c ist die Voreinstellung dann „11“. In der von Oracle mitgelieferten Beispieldatei sucht man diesen vergeblich; aber wenn er in der „sqlnet.ora“ steht (beziehungsweise ab 12c), muss der Eintrag „SQLNET.ALLOWED_LOGON_VERSION=8“ lauten, um älteren Client-Anwendungen den Zugriff zu ermöglichen.

Für die Konfiguration von SQL*Net auf dem Client-Rechner können weder „sqlnet.ora“ noch „tnsnames.ora“ einfach vom Server kopiert werden – das wohl häufigste Vorgehen beim Nutzen dieser Methode. Die „sqlnet.ora“ lässt sich unbearbeitet aus der Installation nutzen. Die Einträge für Aliasse mit TCP/IP in der „tnsnames.ora“ haben sich nicht umfangreich verändert, es sind nur weitere Möglichkeiten der Adressierung hinzugekommen. Daher lassen sich die aktuellen Einträge für die notwendigen Datenbanken weitgehend übernehmen, jedoch ist der Eintrag „GLOBAL_NAME“ zu entfernen, denn globale Namen gibt es erst seit Version 8i. Zuletzt gilt es noch, mit der Abfrage „select username, PASSWORD_VERSIONS from DBA_USERS“ die Passwort-Version für das Schema zu prüfen, auf das die älteren Client-Anwendungen zugreifen sollen (siehe Abbildung 2).

Zugriff mit älteren Client-Anwendungen gibt es nur, wenn das entsprechende Schema auch eine Passwort-Version „10G“ besitzt, wie das Schema „STRECKO“ in Abbildung 2. Da es sich hierbei um den schwächsten Verschlüsselungs-Algorithmus handelt, sollten mindestens die Schemata „SYS“ und „SYSTEM“ ausschließlich die höchste Passwort-Version besitzen; deshalb auch die Markierung „!“ bei diesen Schemata. Nachdem SQL*Net jetzt die entsprechende Konfiguration besitzt, kann als erster Test eine Anmeldung mit SQL*Plus 3.3.3 (Datenbank-Version 7.3.3) erfolgen (siehe Abbildung 3).

Da der Datenbank-Katalog immer nur erweitert wurde und die Basis-Tabellen dafür unverändert blieben, lassen sich bekannte Katalog-Views wie „USER_TABLES“ fehlerfrei abfragen. Neuere Katalog-Tabellen und damit die darüberliegenden Views enthalten allerdings auch für das alte SQL*Plus Unverständliches (siehe Abbildung 4).

Anwender-Regeln haben frühestens mit Version 8 Einzug gehalten, denn bei dem undefinierten Datentyp für „RULE_CONDITION“ handelt es sich um einen CLOB, diesen Datentyp gibt es ab Version 8.0.4. Weitere Details zu Einschränkungen bei Cli-

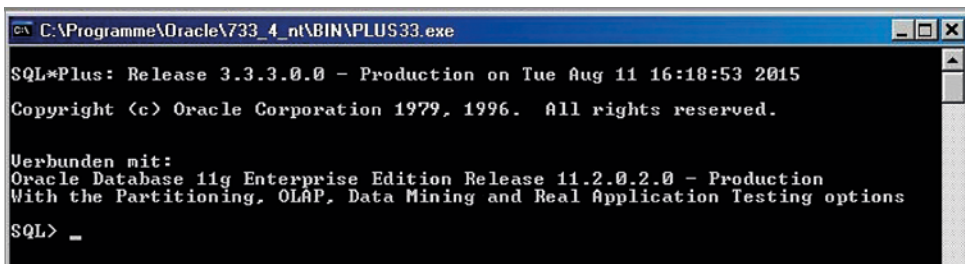


Abbildung 3: Anmeldung mit SQL*Plus 3.3.3 an einer Oracle-Datenbank 11g

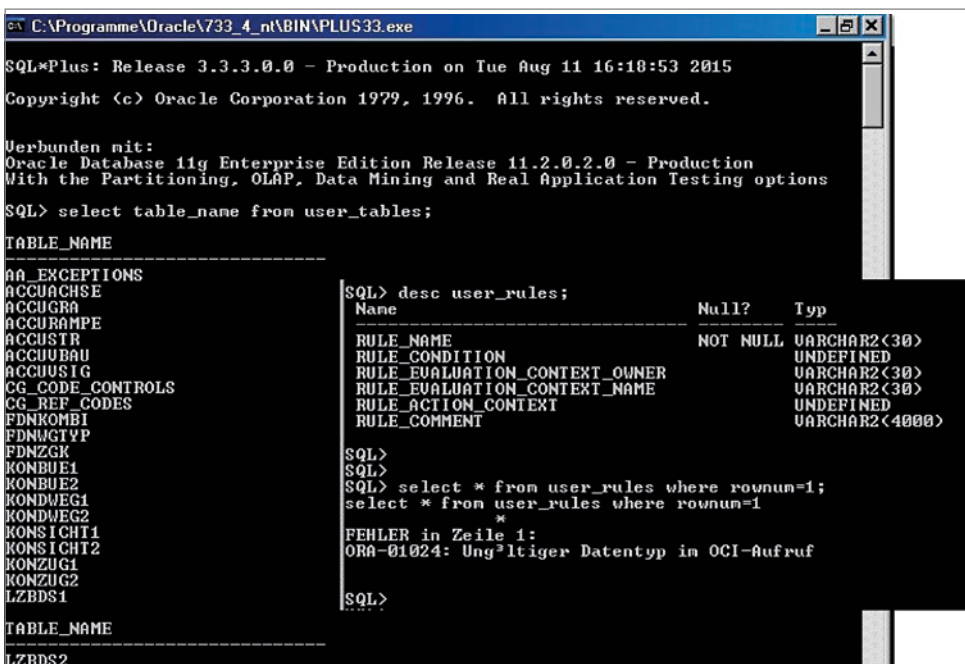


Abbildung 4: Katalog abfragen mit SQL*Plus 3.3.3 bei einer Datenbank 11g

ent-Anwendungen, basierend auf den Bibliotheken älterer Oracle-Versionen, folgen später, da dies bei der Entwicklung von Anwendungen in Programmiersprachen der dritten Generation (3-GL) zu beachten ist.

Die Lizenzierung

Wenden wir uns an dieser Stelle zunächst noch der lizenzrechtlichen Seite zu. Am einfachsten sind die beiden Fälle Personal Edition (PE) und Enterprise Edition (EE) mit Prozessor-Lizenz:

- **PE**
Hier kann wohl zu Testzwecken (quasi unter OTN-Lizenz) ein älterer, noch verfügbarer Client beziehungsweise SQL*Net und SQL*Plus installiert sein, ein Betrieb ist ausgeschlossen.
- **EE/Prozessorlizenz**
Liegt ein Installationsmedium für eine ältere Version vor, können die Client-Software installiert und Client-Anwendungen betrieben werden – solange der Server dafür nicht um einen Prozessor erweitert wird.

Etwas genauer hinsehen muss man bei EE mit Lizenzierung benannter Anwender (Named User Plus) sowie bei Standard Edition (SE) und Standard Edition One (SE1). Die über die Konsole (Industrie-Rechner) gesteuerte Maschine gilt für die Oracle-Lizenzierung als ein Benutzer; daher ist für diese Installation eine zusätzliche Nutzer-Lizenz erforderlich. Im Fall von SE und SE1 muss noch überprüft werden, ob man durch den zusätzlichen Nutzer eine vorgegebene Höchst-Anzahl für die jeweilige Edition überschreitet. Eine dafür notwendige Erweiterung des Servers mit zusätzlicher CPU kann zur Folge haben, dass die lizenzierte Edition nicht mehr ausreicht.

Liegt eine gültige Lizenz vor, dann ist die Installation einer älteren Software-Version ein Downgrade, was lizenzrechtlich unbedenklich sein sollte, allerdings benötigt man natürlich noch ein Installationsmedium. Firmen, die bereits seit Langem Oracle-Datenbanken einsetzen, haben ältere Versionen möglicherweise im Archiv. Wer einen gültigen Support-Vertrag besitzt, kann über „My Oracle Support“-Medien oder einen Link zum Herunterladen ältere Software-Versionen anfordern.

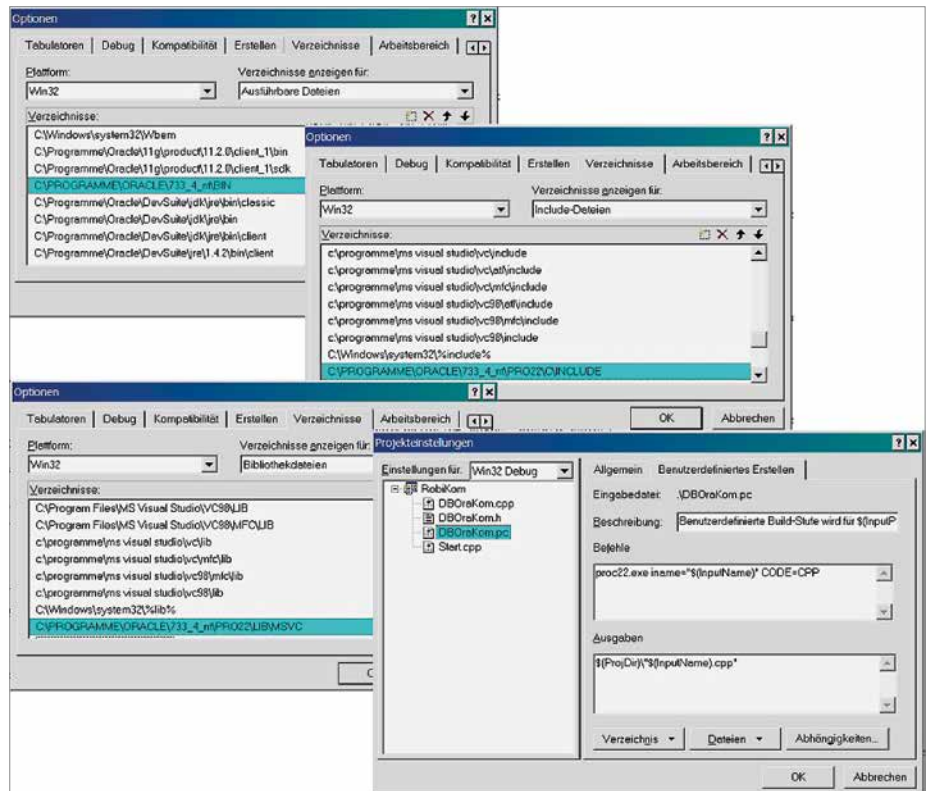


Abbildung 5: Pro*C 2.2 einbinden in Visual Studio

Eine Besonderheit bildet Express Edition (XE). Es gibt zwar keine Benutzer-Beschränkung, aber die Lizenz enthält nicht die Precompiler (Pro*C, Pro*Cobol) sowie SQLJ. Damit berechtigt diese Lizenzierung nicht zur Programmierung. Nach Auskunft von Oracle LMS gibt es aber keine Beschränkungen bezüglich der eingesetzten Client-Version für extern entwickelte Software, wenn entsprechende Installationsmedien vorliegen.

Eingebettetes SQL in 3-GL-Programmiersprachen

Zunächst benötigt man die Programmierumgebung der alten Oracle-Version, bestehend aus Header-Dateien, Bibliotheken und Pre-Compiler. Dafür muss das zur Oracle-Version gehörende Pro*C (Pro*Cobol ist wohl eher ungeeignet) in der modernen Entwicklungsumgebung installiert sein.

Da Microsoft wenig von Rückwärts-Kompatibilität hält, hat die Autorin die Installation unter Windows 7 nur über den Umweg über einen Rechner mit Windows XP und nachträglichem Kopieren der Installation (Ordner sowie die notwendigen Einträge in der Registrierung) geschafft.

Anschließend ließ sich der Pre-Compiler dann aber anstandslos ins Visual Studio einbinden. Dabei ist darauf zu achten, dass man alle Pfade berücksichtigt und beim benutzerdefinierten Erstellen für die PC-Dateien besser die meist auch vorhandene „PROC[nn].EXE“ verwendet, um sicher zu sein, den richtigen Pre-Compiler anzusprechen (siehe Abbildung 5).

Leider erhält man unter Windows 7 bei der Ausführung der erzeugten Programme schon bei der Ausführung des eingebetteten Befehls „EXEC SQL CONNECT :psUser IDENTIFIED BY :psPW USING :psDB;“ eine „nicht abgefangene Ausnahme in ... (NTDLL.DLL): 0xC0000008; Invalid Handle“. Daher ist das Debuggen höchstens remote möglich.

Die Programmierung

Bei der Programmierung gilt die besondere Beachtung den verfügbaren Datentypen und Datenbankobjekten. In allen Versionen ab 7.3.3 gibt es folgende Datentypen:

- VARCHAR2 bis mindestens 2.000 Zeichen
- CHAR bis mindestens 255 Zeichen

- NUMBER
- LONG bis 2 GB
- DATE
- RAW bis mindestens 255 Zeichen
- LONG RAW bis 2 GB
- ROWID

Ab Version 8.0.4 können ein „VARCHAR2“ bis 4.000 Zeichen sowie „CHAR“ und „RAW“ bis 2.000 Zeichen Länge definiert sein. Außerdem gibt es ab dieser Version zusätzlich „NCHAR“ bis 2.000 Byte sowie „CLOB“, „NCLOB“, „BLOB“ und „BFILE“ mit jeweils bis 4 GB Fassungsvermögen. In Version 8i (8.1.5) existiert zusätzlich „UROWID“ mit einer Länge bis zu 4.000 Byte.

Ab Version 9i (9.0.1) muss bei den Längen-Angaben für „VARCHAR2“ und „CHAR“ noch nach „BYTE“- und „CHAR“-Semantik unterschieden werden. Außerdem gibt es jetzt auch „TIMESTAMP“ sowie die Datums-/Zeit-Intervalle. Auf diese Beschränkungen ist je nach der eingesetzten Client-Version nicht nur bei der Deklaration im Quellcode zu achten, sondern auch beim Anlegen der für das Projekt infrage kommenden Datenbankobjekte wie Tabellen, Views und Funktionen; weitere Objekte wie Prozeduren oder Packages hat die Autorin bisher nicht eingesetzt.

Bei Funktionen müssen die Ein- und Ausgabeparameter den Beschränkungen genügen; das Innere der Funktion interessiert lediglich die Datenbank und läuft damit in der Umgebung der Server-Software. Wenn man diese Einschränkungen beherzigt, kann mit einer Funktion die Stelle eines Werts in einer durch Doppelpunkte separierten Liste in einem SQL*Plus der Oracle-Version 7.3.3 abgefragt werden (siehe Listing 1).

Obwohl zurzeit von Oracle 7.3.3 keiner etwas von HTMLDB/Apex auch nur ahnte, erhält man eine fehlerfreie Ausgabe, wie *Abbildung 6* zeigt. Die Funktion in Listing 2 ist Teil eines einfachen Daten-Transfers von Statusmeldungen eines Roboters in eine Datenbank-Tabelle „ROBI_LOG“.

„Pre-compiliert“ mit „PROC22“ und gebunden mit den Bibliotheken „SQLLIB18.LIB2“ und „SQLLIB18.LIB“, liefert die Anwendung einwandfrei Daten aus einer alten 7er-Oracle-Client-Umgebung in einer 11g-Datenbank ab – allerdings nur unter älteren Windows-Betriebssystemen (Testumgebung: Windows XP). *Abbildung 7* zeigt die in der Tabelle „ROBI_LOG“ abgespeicherten Daten.

```
create or replace FUNCTION GetIndex4Value
(ValueString IN varchar2, Value IN varchar2)
return number
AUTHID CURRENT_USER
is
    TableOfValues HTMLDB_APPLICATION_GLOBAL.VC_ARR2
                := HTMLDB_UTIL.STRING_TO_TABLE(ValueString);
    nInd          number := 1;
begin
    while ( nInd <= TableOfValues.count and
           TableOfValues(nInd) <> Value ) loop
        nInd := nInd +1;
    end loop;
    if ( nInd > TableOfValues.count ) then nInd := 0; end if;
    return nInd;
end;
```

Listing 1

```
int speicherLog(char* pcBez, double nWert)
{
    EXEC SQL BEGIN DECLARE SECTION;
    char* psBezeichnung = pcBez;
    double nWertangabe = nWert;
    double nZeitDiff;
    int nLfdNr;
    EXEC SQL END DECLARE SECTION;

    EXEC SQL select sysdate - RL_ZEITSTEMPEL Zeit_Diff,
                    max(RL_LFD_NR)
                    into nZeitDiff, nLfdNr
                    from robi_log
                    where RL_ZEITSTEMPEL
                        = (select max(RL_ZEITSTEMPEL)
                          from robi_log)
                    group by sysdate - RL_ZEITSTEMPEL;
    if ( nZeitDiff != 0 ) { nLfdNr = 0; }
    else { nLfdNr ++; }

    EXEC SQL insert into ROBI_LOG
                (RL_ZEITSTEMPEL, RL_LFD_NR, RL_BEZ, RL_WERT)
                values (sysdate, :nLfdNr, :psBezeichnung,
                       :nWertangabe);

    if ( sqlca.sqlcode == 0 )
    {
        EXEC SQL commit;
    }

    return sqlca.sqlcode;
}
```

Listing 2

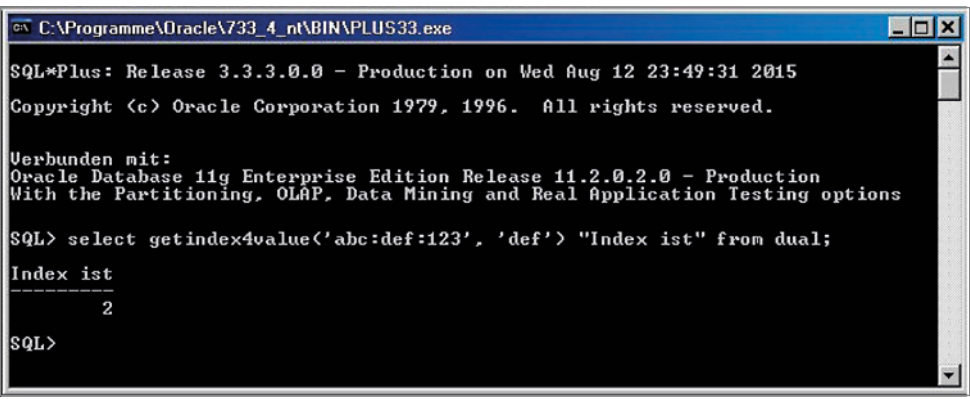


Abbildung 6: Funktionsabfrage in SQL*Plus 3.3.3

```

1 alter session set nls_date_format = 'dd.mm.yyyy hh24:mi:ss';
2 select * from ROBI_LOG where to_char(RL_ZEITSTEMPEL, 'yyyymmdd') = 20150812
3 order by RL_ZEITSTEMPEL desc, RL_LFD_NR desc;
4

```

RL_ZEITSTEMPEL	RL_LFD_NR	RL_BEZ	RL_WERT
1 12.08.2015 20:49:44	10	Maschine Stop	1
2 12.08.2015 20:49:44	9	Aktueller Lärmpegel	541
3 12.08.2015 20:49:44	8	Aktueller Lärmpegel	517
4 12.08.2015 20:49:44	7	Aktueller Lärmpegel	539
5 12.08.2015 20:49:44	6	Aktueller Lärmpegel	552
6 12.08.2015 20:49:44	5	Spannung	5,324
7 12.08.2015 20:49:44	4	Maschine Start	0
8 12.08.2015 20:49:44	2	Maschine Stop	1
9 12.08.2015 20:49:44	1	Aktueller Lärmpegel	557
10 12.08.2015 20:49:44	0	Aktueller Lärmpegel	540
11 12.08.2015 20:47:18	10	Maschine Stop	1

Abbildung 7: Daten in der Tabelle „ROBI_LOG“

Die Spalte „RL_ZEITSTEMPEL“ ist selbstverständlich als „DATE“ deklariert; deshalb benötigt man zusätzlich die Spalte „RL_LFD_NR“, um eine exakte Abfolge der Meldungen zu erhalten.

Fazit

Die meisten Probleme bereiten Windows-Betriebssysteme bei der Aufgabe, ältere Maschinen durch Datenbank-Anbindung

in eine automatisierte Umgebung zu integrieren. Erhält man von Oracle Unterstützung oder zumindest ein gewisses Entgegenkommen bei der Beschaffung der notwendigen Installationsmedien für ältere Oracle-Versionen (7 bis 9), dann ist die Aufgabe lösbar. Es lässt sich auch in solchen Umgebungen ein weiterer Schritt in Richtung „Industrie 4.0“ machen.



Gerhild Aselmeyer
g.aselmeyer@aka-edv.de

Alles, was die SAP-COMMUNITY wissen muss, finden Sie monatlich im E-3 MAGAZIN.
Ihr WISSENSVORSPRUNG im Web, auf iOS und Android sowie PDF und Print:
e-3.de/abo

Wer nichts weiß,
muss alles glauben!

Marie von Ebner-Eschenbach



SAP® ist eine eingetragene Marke der SAP AG in Deutschland und in den anderen Ländern weltweit.

www.e-3.de

Cloning-Lösungen im Überblick

Miguel Anjo in Zusammenarbeit mit Ludovico Caldara, Catalin Motaitianu und Dawid Wojcik, Trivadis AG

Datenbanken werden immer größer und größer. Häufig sind die Archivierung und der Lebenszyklus der Daten nicht im ursprünglichen Design eingeplant. Das führt zu Problemen, wenn Entwickler Kopien zum Testen, zur Qualitätssicherung, zur Kontrolle oder aus anderen Gründen benötigen. Entwickler, Datenbank-Administratoren und Fachabteilungen brauchen eine aktuelle Kopie der Datenbank im Zugriff, um Patches, Tests oder Reports ohne Auswirkungen auf die Produktion durchführen zu können.

Die Auswahl und Implementierung einer Cloning-Lösung sollte entsprechend sorgfältig erfolgen, damit über das reine Datenbank-Cloning hinausgehende zusätzliche Anforderungen berücksichtigt werden können: Daten maskieren, isolieren oder mit weiteren Systemen oder Datenbanken synchronisieren sind mögliche erweiterte Funktionen. Parameter wie die Größe der Datenbank, die Übertragungsgeschwindigkeit, die verfügbare Festplattenkapazität sowie der mögliche Zeitaufwand für das Klonen spielen ebenfalls eine wichtige Rolle. Das Wiederherstellen von Sicherungen, ein RMAN-Duplikat oder Export/Import sind mögliche Lösungen, wenn die Datenmenge begrenzt, die Zeitdauer für das Klonen kein Problem und das Budget begrenzt ist.

Oracle verbessert laufend die Cloning-Funktionen des Enterprise Manager. Die-

ser bietet eine einfach zu bedienende gute Lösung. Firmen wie Delphix oder ActifIO haben sich auf virtuelle und physische Anwendungen spezialisiert, um Klone sowohl auf Oracle- als auch auf Speicher-Technologien schnell zu erstellen. Man sollte jedoch bedenken, dass Klone nicht immer die beste Lösung für die Bedürfnisse des Unternehmens sind und man deshalb auch andere Alternativen betrachtet werden sollten.

Wie Datenbank-Cloning funktioniert

Datenbank-Cloning ermöglicht das Erstellen einer Kopie der Produktionsdaten mit möglichen Untermengen, Maskierung und Isolation (siehe Abbildung 1). Aus einer

Sicherung oder der eigentlichen Datenbank entsteht eine erste Kopie im Staging-Bereich. Dies kann eine vollständige Kopie oder nur eine Teilmenge der Tablespace sein. In dieser Kopie erfolgt das Maskieren sensibler Daten. Zudem wird das Senden von E-Mails an bestehende Kunden sowie die Verbindung zu anderen Datenbanken unterbunden. Am Ende steht eine Master-Kopie, die sich beliebig klonen lässt.

Zuvor sollte man jedoch möglichen Anwendungsfälle und deren Häufigkeit planen. Dabei hilft der folgende Fragenkatalog:

- Wie wird die geklonte Datenbank genutzt?
- Wie oft werden Klone benötigt?
- Wie viele unabhängige Klone sind notwendig?
- Wie schnell sollen die Klone vorliegen?

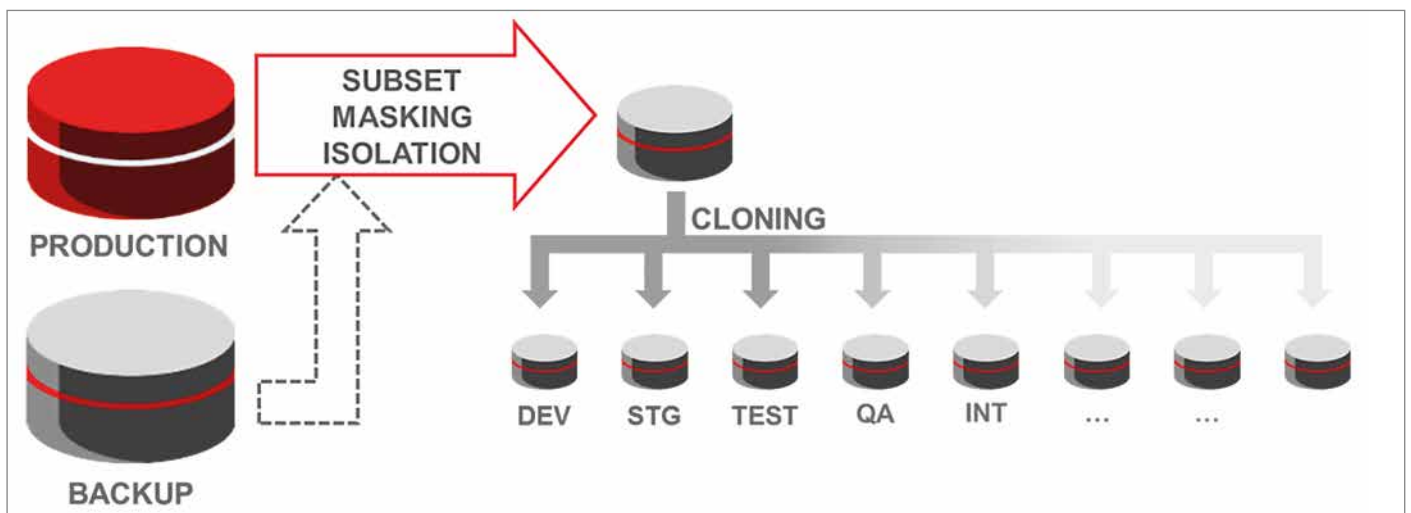


Abbildung 1: Datenbank-Cloning

- Wie lange wird ein Klon benötigt?
- Ist Daten-Anonymisierung erforderlich?
- Ist eine Self-Service-Lösung für die Bereitstellung der Datenbank-Kopien notwendig?
- Gibt es im Unternehmen andere Datenbanken oder Anwendungen mit ähnlichen Anforderungen?
- Hat das Unternehmen bereits ein Storage-System mit Cloning-Features?
- Sind Features wie das Klonen eines Klons zu beliebigen Zeitpunkten erwünscht?
- Gibt es andere Datenbanken/Systeme, die zur gleichen Zeit geklont werden müssen?
- Wird die Enterprise Edition der Datenbank verwendet?
- Ist der Oracle Enterprise Manager Cloud Control im Unternehmen im Einsatz? Wenn ja, welche Management Packs sind lizenziert?

Frei verfügbare Lösungen

„RMAN Restore“ bringt den großen Vorteil, gleichzeitig die Backups zu überprüfen. RMAN kann auch eine vollständige Kopie der Datenbank mit der gleichen Fragmentierung erstellen, was für Performance-Tests wichtig ist. Um das Klonen zu automatisieren, bedarf es jedoch einiger Skripte. Der erforderliche Speicherplatz sowie die Zeit für das Klonen sind abhängig von der Anzahl der Klone, sodass diese Lösung nur für eine begrenzte Menge von Klonen oder für kleine Datenbanken interessant ist.

„RMAN Duplicate“ steigert die Geschwindigkeit des Klonens aus einer aktiven Datenbank im Vergleich zu einer Wiederherstellung einer Sicherung, allerdings werden die Wiederherstellung der Datenbank und Auswirkungen auf die Produktion nicht getestet (außer, man dupliziert aus einer physischen Standby-Datenbank).

Der wichtigste Vorteil von „Datapump Export und Import“ ist die einfache Bedienung. Es ist auch möglich, Untermenüen und Maskierung gleichzeitig zu erledigen. Jedoch ist es nicht einfach, einen Katalog von Klonen zu unterhalten. Der verwendete Speicherplatz ist abhängig von der Größe der Datenbank. Die Fragmentierung der Daten geht verloren, was die Möglichkeit verringert, den Klon für Performance-Tests einzusetzen, weil der

Optimierer evtl. andere Ausführungspläne bevorzugt.

Fazit: Die frei verfügbaren Lösungen sind gut, wenn man eine begrenzte Zahl von Klonen benötigt und die Datenbanken klein sind. RMAN verifiziert gleichzeitig der Restore-Vorgang und überprüft die Backups. Die erwähnten Lösungen bedürfen Skripte zum Automatisieren und benötigen so viel Speicherplatz, wie die Anzahl der Klone beträgt.

Oracle-Lösungen

Mit Aktivieren des Parameters „CLONEDB“ lassen sich einfache Kopien aus einer Read-only-Datenbank erstellen. Seit Version 12.1.0.2 ist dies auf jedem Dateisystem möglich, das sparse files unterstützt, einschliesslich „ext4“ oder „ACFS“. Diese Technologie nutzt die Paket-Funktion „DBMS_DNFS.clonedb_renamefile“, benannt nach ihrer vorherigen Verwendung mit direct NFS. Bei neuen Pluggable-Datenbanken erledigt "Snapshot Copy" die Arbeit, dabei muss aber die Multitenant-Option der Enterprise Edition lizenziert sein.

Diese kostenlose Lösung (auch in der Standard Edition verfügbar) erlaubt es, viele Klone aus der gleichen Sicherung zu erstellen, Tests der Autoren zeigten jedoch eine geringere Leistung der Klone. Bei der Verwendung von Nicht-CDBs sind einige Skripte erforderlich und es gibt keinen Katalog der Klone. Bei Lizenzierung des „Cloud Management Pack“ ist es möglich, mithilfe des Enterprise Manager die Klone zu erstellen und zu katalogisieren. Es ist allerdings eine etwas teure Lösung für diese Anwendung.

Als Teil von Enterprise Manager Cloud Control 12c ist SnapClone eine einfache Cloning-Lösung, basierend auf Hardware (NetApp, Oracle Sun ZFS Storage Appliance EMC VMAX und VNX) oder mit Software (mit ZFS-Dateisystem). Es funktioniert durch die Installation eines Agent- und Speicher-basierten Verwaltungsskripts. Der Klonprozess erfolgt durch einen Assistenten im Enterprise Manager, der viele Optionen bei der Erstellung der Zieldatenbank ermöglicht, sogar RAC-Klone. Die Datafiles der Klone sind aus einem NFS-Mountpoint sichtbar.

Seit dem Datenbank-Release 11.2.0.4 lassen sich Datenbank-Snapshots mit der ACFS-Funktion erstellen. Es empfiehlt sich,

die Version 12.1 der Grid Infrastructure zu verwenden. Die Snapshots entstehen dann mit wenigen Befehlen auf einem ADVM-Volume. Tests der Autoren zeigten große Leistungseinbußen auf der Quell-Datenbank, nachdem ein paar Snapshots erstellt wurden. Sie empfehlen deshalb, die Snapshots aus einer Standby-Datenbank (natürlich nicht derjenigen, die für die Notfallwiederherstellung vorgesehen ist) auszuführen, nachdem vorübergehend der Apply-Modus deaktiviert wurde. Auf diese Weise beeinflusst das Erstellen der Snapshots die primäre Datenbank nicht.

Oracle stellte das Skript „gDBClone.pl“ bereit, das alle Klon-Aufgaben auf einmal ausführen soll. Es scheint aber nicht ein DBA gewesen zu sein und das Skript ist definitiv nicht für die Produktion bereit. Anstatt zu warten, bis eine neue Version dieses Skripts von Oracle freigegeben wird, kann es einfacher sein, ein vergleichbares Skript selbst zu erstellen.

Proprietäre Lösungen

Die wichtigsten proprietären Lösungen für das Oracle-Datenbank-Cloning sind Delphix und ActifIO. Setzt man einige der besten Oracle-Spezialisten und Entwickler von ZFS zusammen, kann ein interessantes Werkzeug entstehen. Dies ist der Fall bei der Delphix-Agile-Datenplattform. Es handelt sich um eine virtuelle Appliance, die auf vSphere unter einem Fork von Open Solaris 5.11 namens „DelphixOS“ läuft und Daten unter einer Abspaltung der OpenZFS unter dem Namen „DxFS“ speichert. Derzeit werden mehrere Datenbank-Anbieter (Oracle, SQL Server, PostgreSQL, Sybase) unterstützt. Backups, inkrementelle Sicherungen sowie Redo-Logs sind auf dem Dateisystem DxFS möglich; daraus lassen sich Point-in-Time Klones mit NFS erstellen.

Es gibt eine benutzerfreundliche Weboberfläche zum Klonen (zusätzliche Option), aber es ist auch möglich, den Zugriff auf die Appliance über ssh oder https zu automatisieren. Eines der Features ist, einen Klon aus einem anderen Klon zu erstellen, sowie einen schnellen Rücklauf zu starten und einen physischen Klon zu kreieren. Hinzu kommt die Möglichkeit, gegen Aufpreis ein Daten-Maskierungs-Modul zu verwenden.

ActifIO ist spezialisiert auf Lösungen für kontinuierliche Datensicherung mit

Lösung	Lizenzierung	Cloning Typ	Einsatz	Anmerkung
RMAN restore	Frei	Full	Kleinere Anzahl Klone	Prüfen der Backups möglich
RMAN duplicate	Frei	Full	Kleinere Anzahl Klone	Schneller als die Wiederherstellung vom Band
Data Pump	Frei	Full	Kleinere Anzahl Klone	Einfache Bedienung
Clonedb, PDB snapshot copy	Frei ohne OEM, sonst Cloud Management Pack erforderlich, bei PDB Snapshot Copy-Multitenant-Option	Thin	Kleinere Anzahl Klone	Einfach mit PDB, ansonsten Scripting oder OEM-Lizenz erforderlich
Snap Clone	Cloud Management Pack	Thin	Für einige Speicher-Typen verfügbar	OEM-Integration
ACFS snapshots	Enterprise Edition	Thin	Einfache zukünftige Lösung	Leistungsstark mit verschiedenen Klonen
Delphix	Kostenpflichtig	Thin	Sehr gut in bestimmten Fällen	Sehr gute Entwicklung
ActifIO	Kostenpflichtig	Thin	Übergreifende Lösung für Unternehmen	
Storage snapshot	Kostenpflichtig	Thin	Frei mit Oracle	Scripting erforderlich
DataGuard snapshot standby	Enterprise Edition	Full	Für wiederholte Tests am besten geeignet	Teil von DataGuard

Tabelle 1

Komprimierung und Deduplizierung. Die meisten Backup-Daten ändern sich im Laufe der Zeit praktisch nicht. ActifIO arbeitet mit mehreren Lösungen wie VMware, Oracle oder Microsoft zusammen. Es ist als Software-Appliance verfügbar, für größere Installationen existiert auch eine physikalische Einheit.

ActifIO erstellt sofortige Klone für Oracle-Datenbanken und kann diese auch in physikalische Kopien umwandeln. RMAN kommt für das Erzeugen einer ersten Sicherung zum Einsatz; anschließend werden dann nur noch die inkrementellen Änderungen erfasst. Die Autoren hatten bisher keine Möglichkeit, sich näher mit diesem Produkt zu beschäftigen, aber es scheint interessant zu sein, falls ein Unternehmen nach einer globalen Backup-Lösung mit einem Web-Interface und der Bereitstellung von einfachen Oracle-Klonen auf ZFS Basis sucht.

Weitere Alternativen

Oracle 12c bietet neue RMAN-Funktionen für Storage-Snapshots. Das zeigt, dass Oracle das Zurückspielen von Snapshots

garantiert, ohne die Datenbank in den Backup-Modus zu versetzen. Damit lassen sich vorhandene Speicher-Lizenzen für Snapshot-Technologie und einfaches Scripting für Datenbank-Klone verwenden. Ein Anwendungsfall für Speicher-Snapshots ist die gleichzeitige Erstellung von Klonen mehrerer Datenbanken. Man kann sich gut vorstellen, den Snapshot einer ASM-Diskgroup oder eine große Partition zu erstellen, während mehrere Datenbanken laufen. Für die Verwendung dieser RMAN-Funktionen ist allerdings die Advanced-Compression-Option der Datenbank erforderlich.

Eine häufige Anforderung ist die Möglichkeit, wiederholt Tests auf einer Datenbank durchzuführen, nachdem sie in den ursprünglichen Zustand zurückversetzt wurde. Dafür sind keine Snapshots erforderlich, denn die vorhandene DataGuard-Snapshot-Standby-Funktion kann dafür genutzt werden. Sie ist mit der Enterprise Edition kostenlos und ermöglicht mit einfachen Skripten, einen Self-Service für Benutzer zu erstellen. Dabei ist es erforderlich, eine Standby-Datenbank pro möglichem Klon haben.

Fazit

Tabelle 1 zeigt eine Übersicht über die wichtigsten Vorteile und Anwendungsfälle der Lösungen, die in diesem Artikel vorgestellt sind. Es gibt viele gute Möglichkeiten für das Datenbank-Cloning; die Auswahl der Lösung hängt vom Anwendungsfall ab.

Zu Beginn eines Cloning-Projekts ist es wichtig, die richtigen Fragen zu stellen: „Wie viele Klones?“, „Wie schnell?“, „Wie oft?“, und „Zu welchen Kosten?“ etc. Erfahrungen anderer Anwender können ebenfalls ein Kriterium bei der Auswahl einer Backup- oder Speicher-Lösung geben. Last but not least, muss auch die Oracle-Lizenzierungssituation in betracht gezogen werden. In den meisten Fällen wird es keine eindeutig beste Lösung geben und jeder Fall ist ein Einzelfall.

Miguel Anjo
miguel.anjo@trivadis.com



E-Mail vom DB-Server: „Brauche neues Power Supply!“

Elke Freyemann, ORACLE Deutschland B.V. & Co. KG

Leider doch schon mal passiert: Der neue Datenbank-Server von Oracle soll selbstverständlich auch in Sachen Ausfallsicherheit punkten. Dazu hat er unter anderem auch zwei redundante Power Supplies. Leider stellt das eine die Arbeit ein – und das hat leider niemand mitbekommen. Sollte jetzt auch noch Nummer zwei ausfallen, werden das jedoch mit Sicherheit alle merken.

Wie also kann die Oracle-Hardware rechtzeitig einen Fehler mitteilen? So rechtzeitig, dass Redundanzen des Systems greifen, der aufgetretene Defekt behoben werden kann und letztendlich wieder alles läuft. Kann die Hardware sich im Rahmen einer effektiven Überwachungslösung das benötigte Ersatzteil nicht am besten gleich selbst bei Oracle anfordern? Ja, auch das lässt sich einrichten.

Integrated Lights Out Manager (ILOM) und eXtended System Control Facility (XSCF) sind diejenigen Komponenten in einem Oracle Server, die die relevanten Daten liefern: Fehlermeldungen einzelner Module, Temperaturwerte der verschiedenen Sensoren, Stromverbrauch oder auch einfach Angaben zu Server-

modell, Seriennummer und Ausbau der Maschine. Oracle Auto Service Request (ASR) empfängt direkt die Meldungen von der Hardware, filtert sie und nimmt eine Bewertung dahingehend vor, ob die Meldung tatsächlich auf einen Defekt hinweist. Sollte dies der Fall sein, wird automatisch ein Service Request bei My Oracle Support eröffnet.

Für Enterprise Manager Cloud Control ist das Oracle-Hardware-Plug-in verfügbar. Damit bindet man die entsprechenden Server auch mit ihren Hardware-nahen Daten in die Überwachung durch Cloud Control oder Grid Control ein. Es werden nicht nur Fehlermeldungen weitergeleitet und verarbeitet, man hat auch den sogenannten „Health State“ der Komponenten

und die Daten der diversen Hardware-Sensoren (Temperaturen, Lüfterdrehzahlen, Leistungsaufnahme etc.) im Blick.

Nutzer von Enterprise Manager Ops Center haben in diesem grafischen Management-Tool diese Funktionalitäten ebenfalls in vollem Umfang zur Verfügung.

Der direkte Draht der Hardware zu My Oracle Support

Üblicherweise sind in modernen Infrastruktur-Komponenten – seien es nun Server, Storage-Systeme, Switches oder gar komplette Engineered Systems – Service-Prozessoren im Einsatz. Diese arbeiten komplett unabhängig vom instal-

lierten Betriebssystem als Rechner im Rechner, meist mit eigenem Netzwerk-Zugang.

Oft auch „LOM-Module“ genannt, überwachen sie den Betriebszustand, erlauben Zugriff auf Kennzahlen wie Typ, verbaute Komponenten, Versionsnummern von Firmware-Ständen sowie Seriennummern und geben Daten weiter, die die diversen Sensoren für Temperatur, Drehzahlen und Leistungsaufnahme melden.

Auch für einen Reset des installierten Betriebssystems oder den Zugriff auf die serielle Konsole des Systems kann man sie in aller Regel nutzen. Ganz originär fällt den Service-Prozessoren auch die Aufgabe zu, Meldungen der Hardware zunächst einmal im eingebauten Event-Log aufzuzeichnen und je nach getroffenen Konfigurations-Einstellungen über die konfigurierte Netzwerk-Schnittstelle nach außen weiterzuleiten. Üblicherweise erfolgt dies als SNMP-Trap oder direkt als E-Mail.

Man kann sich von seinem Datenbank-Server also häufig direkt und ungefiltert mitteilen lassen, wenn er beispielsweise Probleme mit einem seiner Power Supplies verspürt. Eine potenzielle Schwierigkeit besteht allerdings darin, die empfangene Meldung auch korrekt zu interpretieren: War das nun die Mitteilung über einen normalen Reboot des Systems, gab es eine einmalige Warnung über einen korrigierten Fehler (ECC-Mel-

dung vom Memory) oder liegt ein ernstzunehmender Defekt vor, der eine entsprechende Aktion, wie die Eröffnung eines Service Request, auslösen muss?

Diese Interpretationsarbeit kann man sich bei Oracle-Servern und nahezu der kompletten Palette an Infrastruktur-Komponenten abnehmen lassen. Man lässt die SNMP-Traps an einen sogenannten „ASR-Manager“ schicken. Dieser ist eine schlanke Software-Komponente, die auf My Oracle Support (MOS) zum Download bereitsteht und auf einem Solaris- oder Linux-System installiert wird. Nach entsprechender Konfiguration der Service-Prozessoren empfängt der ASR-Manager dann diese Meldungen und nimmt eine Bewertung vor. Dabei verfügt er über ein ständig aktualisiertes Rule-Set, das es ihm ermöglicht zu entscheiden, wann ein wirklicher Hardware-Defekt vorliegt.

Wer sich für die Details zu diesem Rule-Set interessiert, wird in der Dokumentation zu ASR im Abschnitt „Fault Coverage“ fündig. Für jeden Hardware-Typ sind dort die Meldungen (Events Codes), die empfangen werden können, aufgelistet und weitere Informationen, Datenquelle und notwendige Aktion angegeben. Im Beispiel einer „nicht mehr zu tolerierenden ECC-Meldung“ eines X5-2L Servers, Event-Code „SPX86A-8002-XM“ (mit Verlinkung zu weiteren Details in

MOS), bedeutet „Multiple correctable ECC errors on a memory DIMM have been detected“, die Meldung stammt vom ILOM und muss zur Folge haben, dass ein Service Request mit Priorität 2 eröffnet wird (Action „Case“, Severity „2“).

Sollte also tatsächlich ein Hardware-Defekt gemeldet worden sein, schickt der ASR-Manager die Daten der Fehlermeldung sowie Daten zum System (Typ der Hardware, Seriennummer sowie Host-Name) verschlüsselt an die ASR-Backend-Infrastruktur bei Oracle. Dazu benötigt der ASR-Manager natürlich Internet-Zugriff, entweder direkt oder per https-Proxy.

Im ASR-Backend bei Oracle wird dann anhand der gemeldeten Daten ganz klassisch ein Service Request (SR) eröffnet: Der übertragene Datensatz enthält alle notwendigen Daten zur SR-Eröffnung (aber auch nur die wirklich notwendigen Daten) und so kann der SR umgehend automatisch ohne manuelle Nacharbeit und Zeitverzögerung erstellt werden.

Natürlich erfolgt das alles nicht stillschweigend. Die eingetragenen Ansprechpartner (Registrierungsinformation, gesammelt im Zuge der ASR-Manager-Installation, eingetragene Service-Kontakte derjenigen Hardwarekomponente, von der die Fehlermeldung stammt) werden per E-Mail über den eröffneten SR informiert und über den aktuellen Bearbeitungszustand auf dem Laufenden gehalten.

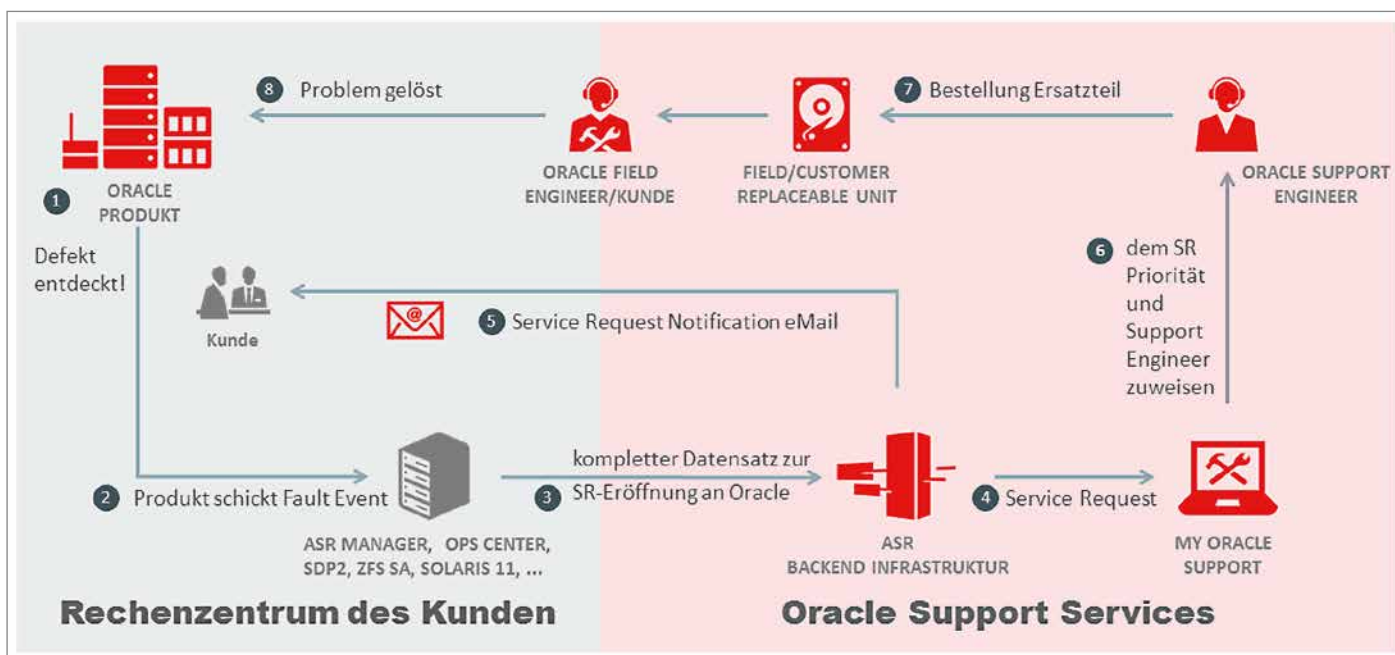


Abbildung 1: Bearbeitungsschritte bei Auto Service Request (ASR)

Der Service Request an sich wird durch einen Support Engineer bearbeitet. Handelt es sich bei dem benötigten Ersatzteil um eine sogenannte „Customer Replaceable Unit“ (CRU) – bei kleineren und mittleren Servern fallen Power Supplies häufig in diese Kategorie – kann es schon mal sein, dass sich der Server sein Ersatz-Netzteil über ASR automatisch selbst bestellt. Fällt das benötigte Ersatzteil in die Kategorie „Field Replaceable Unit“ (FRU), greifen die etablierten Mechanismen, um den Einsatz des Service-Technikers zu koordinieren (siehe *Abbildung 1*).

ASR ist so zu verstehen, wie es die Bezeichnung schon vermuten lässt: ein Mechanismus, mit dem Service Requests automatisch erzeugt werden können. Es ist kein Werkzeug, mit dem ein klassisches Hardware-Monitoring (zum Beispiel Aufzeichnung von Kurven zur Stromversorgung oder Alarmierung bei zu hoher Arbeitslast auf der CPU) implementiert werden kann. Was ASR, genauer der ASR-Manager, aber sehr wohl auch leisten kann: Weiterleitung der empfangenen SNMP-Traps an andere Tools und Weiterleitung der Informationen, wenn ein Service Request eröffnet wurde oder einen neuen Bearbeitungszustand erreicht hat.

In die Überwachung durch ASR fließen bei Servern typabhängig nicht nur die Meldungen des Service-Prozessors mit ein. Auch Benachrichtigungen, die aus der Fault Management Architecture von Solaris oder von einem im Betriebssystem installierten Hardware-Management-Pack stammen, können als Datenquellen dienen. Bei Storage-Systemen werden Phone-Home-Funktionalitäten der Produkte oder bereits in den Common Array Manager eingebaute ASR-Schnittstellen für die Kommunikation mit dem ASR-Backend genutzt. Auch Enterprise Manager Ops Center kann direkt damit kommunizieren.

Enterprise Manager Cloud Control hat auch Hardware fest im Blick

Hält man Ausschau nach einer umfassenden Monitoring- und Management-Lösung, mit der man seine komplette Hard- und Software-Landschaft über eine grafische Oberfläche im Blick halten möchte, stößt man relativ schnell auf den

Oracle Enterprise Manager. Die komplette Produktbezeichnung lautet in der aktuellen Version 12c „Enterprise Manager Cloud Control“. Manchmal wird auch noch der Vorgänger „Enterprise Manager Grid Control 11g“ genannt. Weniger geläufig ist, dass es für diese beiden Werkzeuge, die häufig mit Schwerpunkt Datenbank-Management eingesetzt werden, auch Erweiterungsmodule gibt, mit denen Hardware überwacht werden kann.

Stammt die zu überwachende Hardware aus dem Hause Oracle, kommt das Plug-in für Hardware, aktuelle Version für Cloud Control 12.1.0.5.0, ins Spiel. Das heruntergeladene Plug-in wird zunächst einmal auf dem Oracle Management Service (OMS) ausgerollt. Im Anschluss daran wird es auf jedem Server ausgerollt, dessen Hardware überwacht werden soll, und zwar auf dem Cloud-Control- oder Grid-Control-Agenten, der auf dem Betriebssystem dieses Servers läuft.

Der Agent, der vielleicht bis dato hauptsächlich Datenbank-Monitoring betrieben hat, bekommt über dieses Plug-in Zusatzfunktionen zur Hardware-Überwachung. Die dazu notwendigen Daten und Funktionen liefert der Service-Prozessor, der im Server verbaut ist. Die Kommunikation läuft dabei über die interne Schnittstelle, die der Service-Prozessor zum Betriebssystem hin hat. Das Plug-in unterstützt in der aktuellen Version an dieser Stelle Service-Prozessoren, auf denen ein ILOM läuft. Die genaue Liste der unterstützten Servermodelle ist Teil der Dokumentation; sowohl aktuell angebotene Server als auch ältere Modelle finden sich auf ihr. Als Leistungsumfang deckt das Plug-in ab:

- Basis-Informationen zur Hardware werden erfasst und angezeigt: Modell, Typ und Seriennummer des Systems, Versionsnummer des ILOM, das auf dem Service-Prozessor läuft, IP- und MAC-Adresse des ILOM und teils auch weiterer NICs des Systems
- Direkter Link, über den die Web-Console des ILOM gestartet werden kann
- Erfassung von Daten zur Server-Ausstattung: Anzahl/Typ verbauter CPUs, Memory DIMMs, Lüfter, Power Supplies und Platten beziehungsweise Flash-Module

- Anzeige der Daten zu Stromverbrauch und Temperaturen, die vom ILOM geliefert werden, auch in Chart-Darstellung
- Anzeige eines sogenannten „System Health State“, getrennt nach Funktionsbereichen: Liegen etwa Warnungen von den CPUs, dem Memory, den Lüftern, den Power Supplies oder anderen Komponenten vor?
- Erfassung der Meldungen (Events), die das ILOM liefert, inklusive aller verfügbaren Detail-Informationen

Genau der letzte Punkt dieser Liste zeigt auf: Hat der Datenbank-Server ein Problem mit einem seiner Power Supplies, landet diese Information beim Enterprise Manager, wird zum gelisteten „Open Problem“ und schließlich zum „Incident“. Über das Incident Management von Enterprise Manager kann man sich Benachrichtigungen zustellen lassen. Der Server kann also so seine Botschaft „Brauche neues Power Supply“ mitteilen.

Wenn der Enterprise Manager Verbindung zum Internet hat, können über den Incident Manager, basierend auf den vorliegenden „Open Problems“, auch direkt Service Requests in MOS eröffnet werden. Externe Ticketing-Systeme können auch über den Incident Manager integriert werden. Ist dann das Power Supply im Server getauscht und die Reparatur im ILOM vermerkt, so wird auch die Alarmierung im Enterprise Manager beendet (siehe *Abbildungen 2 und 3*).

Enterprise Manager Ops Center Hardware Monitoring

Ein weiteres Mitglied der Produktfamilie „Oracle Enterprise Manager“ ist Ops Center. Es ist ein eigenständiges Software-Produkt mit eigener Architektur und separat zu installieren. Ops Center ist ein grafisches Management-Tool, das Hardware-Monitoring und Firmware-Management für die Server aus dem eigenen Haus abdeckt. Darüber hinaus bietet Ops Center Betriebssystem-Monitoring und -Patching sowie Virtualization Management; alles mit Schwerpunkt auf Solaris und in aller Regel gesteuert über Agenten, die im Betriebssystem installiert sind.

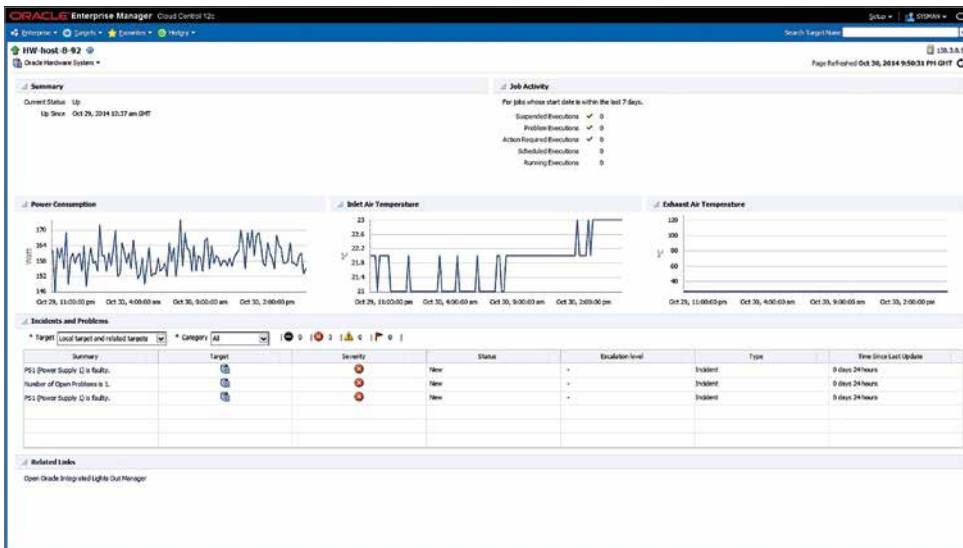


Abbildung 2: Die Ansicht „Oracle Hardware System“ in Enterprise Manager

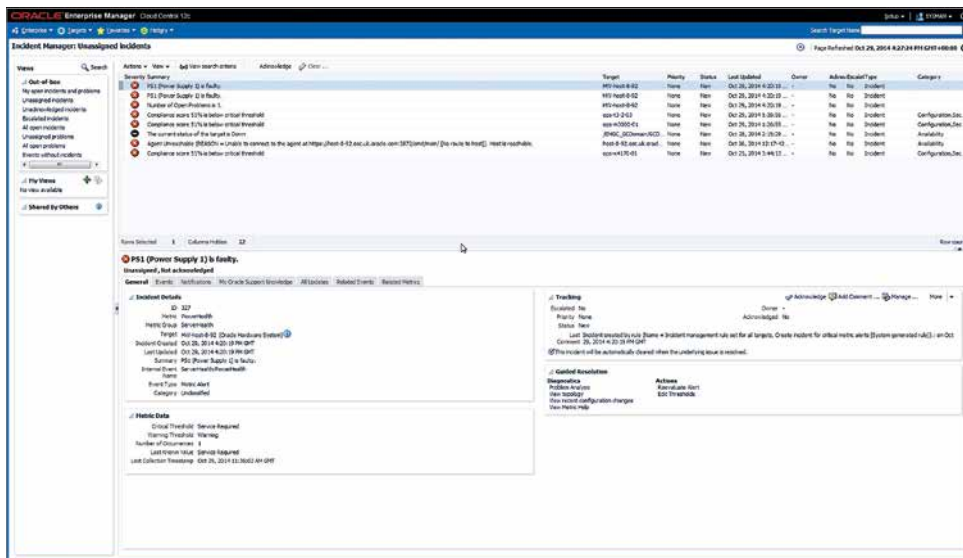


Abbildung 3: EM Incident Manager mit Meldung zu defektem Power Supply

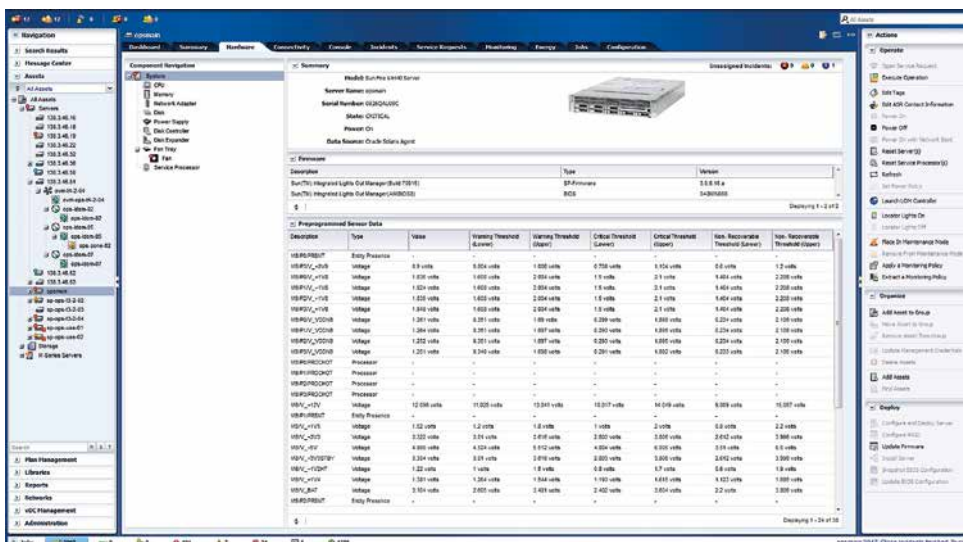


Abbildung 4: Ops-Center-Hardware-Monitoring und Hardware-Management-Funktionen

Auch für Ops Center gilt: Daten für das Hardware Monitoring stammen vom Service-Prozessor. Sie werden durch direkte Kommunikation mit dem verbauten Service-Prozessor erfasst. In aller Regel nutzt dieser dazu SSH- und IPMI-Schnittstellen. Installierte Ops-Center-Agenten sind an dieser Stelle außen vor. Der modulare Ansatz von Ops Center ermöglicht es auch, nur das Hardware-Monitoring zu nutzen und weitere Ops-Center-Funktionen ungenutzt zu lassen. Ops Center stellt die vom Service-Prozessor gelieferten Daten mit höchstmöglicher Detail-Genauigkeit dar. Das geht zum Beispiel so weit, dass von verbauten Memory-DIMMs Herstellerangaben und Seriennummer der einzelnen Komponente, also genau die Detaillierung, die auch der Service-Prozessor nativ selber liefert, gelistet werden. Die „verstandene“ Hardware muss von Oracle, der Partnerschaft mit Fujitsu oder von Sun Microsystems stammen. Systeme anderer Hersteller können nicht in das Hardware-Management von Ops Center eingebunden sein.

Empfängt Ops Center von der überwachten Hardware eine Fehlermeldung, so erfolgt eine entsprechende Alarmierung in der grafischen Benutzeroberfläche. Ops-Center-Benutzer, für die ein Notification-Profile konfiguriert ist, können die generierten Meldungen auch umgehend per E-Mail erhalten. Zusammengehörige aufgelaufene Meldungen, sowohl von der Hardware direkt als auch aus der Fault Management Architecture von Solaris, werden zu Incidents zusammengefasst.

Incidents können ganz klassisch bearbeitet werden: Einem Bearbeiter zuweisen, Anmerkungen zur Problemlösung hinterlegen, vordefinierte Aktionen auf der betroffenen Komponente ausführen, als bearbeitet markieren und schließen. Außerdem kann auf Basis eines Incident auch direkt ein Service Request bei MOS eröffnet werden. Der sogenannte „Enterprise Controller“ von Ops Center muss dazu im „connected mode“ laufen, also Internet-Verbindung haben, und Logindaten zu MOS müssen in Ops Center hinterlegt sein. Über diesen Weg eröffnete Service Requests sind dann auch in Ops Center direkt in der Historie zu der entsprechenden Hardware-Komponente mit ihrem jeweils aktuellen Status einsehbar.

Wünscht man sich einen noch höheren Grad der Automatisierung, kann man

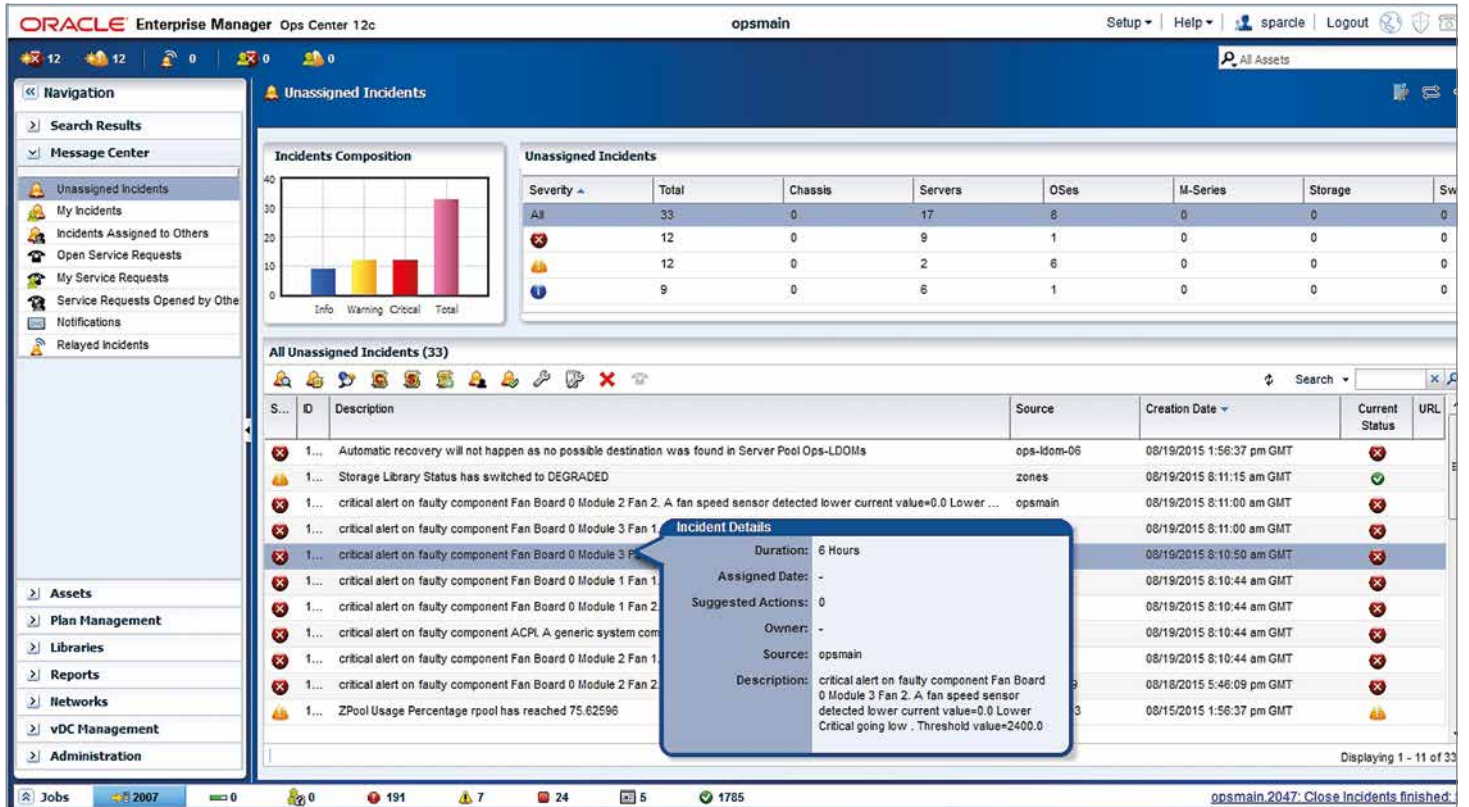


Abbildung 5: Incidents im Message Center des Ops Center

auch eine in Ops Center selber eingebaute Implementierung von ASR aktivieren und nutzen. Diese ist dann alternativ zu der eingangs beschriebenen Stand-alone-Implementierung zu betrachten: Es reicht ja, wenn sich der Datenbank-Server, der mit einem seiner Power Supplies ein Problem hat, per automatisch generierten Service Request ein Ersatzteil bestellen will (siehe Abbildungen 4 und 5).

Die Kosten für diese Lösungen

ASR und Ops Center sind, wie es so schön heißt, „at no additional cost“ im Oracle Premier Support enthalten. Für Betreiber von Ops Center, das neben dem Hardware-Monitoring komplexe Funktionalitäten im Bereich OS- und Virtualization-Management beinhaltet, besteht außerdem Support-Anspruch auf die dokumentierte Produkt-Funktionalität.

Der Licensing Guide zum Enterprise Manager beinhaltet das Regelwerk zur Lizenzierung der Funktionspakete von Cloud Control beziehungsweise Grid Control. Dabei gilt aber gerade für

Cloud Control häufig, dass Infrastrukturnahe Funktionen wie das Hardware-Monitoring in der Lizenzkosten-freien Basis-Funktionalität enthalten sind. Ops Center mit seiner Gesamtfunktionalität, die auch Infrastruktur-Management bietet, ist auch dort beispielsweise im einschlägigen Kapitel des Licensing Guide geführt.

Weiterführende Informationen

- Oracle Auto Service Request: <http://www.oracle.com/asr>
- Dokumentation: http://docs.oracle.com/cd/E37710_01/index.htm
- Cloud Control / Grid Control Plug-in für Oracle-Hardware: <http://www.oracle.com/technetwork/oem/grid-control/downloads/oem-plugin-oracle-hardware-394184.html>
- Enterprise Manager Ops Center: http://docs.oracle.com/cd/E59957_01/index.htm
- Support-Anspruch und Oracle Premier Support: <http://www.oracle.com/us/corporate/features/opscenter-everywhere-program-1567667.html>



Elke Freymann
elke.freymann@oracle.com

Die Effizienz von SmartScan für die Datenbank einschätzen

Franck Pachot, dbi-services

Der Artikel in der letzten Ausgabe hat gezeigt, dass man zuerst „direct-path read“ einführen muss, damit SmartScan wieder ins Spiel kommt. Mit dem Simulationsmodus lässt sich die Effizienz von SmartScan grob abschätzen. Dieser Beitrag geht nun ins Detail, um zu verstehen, warum das betreffende Volume gefiltert werden kann oder nicht. Jenseits von Theorie und Marketing wird gezeigt, ob die Aktivität der Datenbank mit den Anwendungen, Benutzern und Daten beim Umstieg auf Exadata an Leistung gewinnen wird.

SmartScan wird nicht deswegen etwas bringen, weil man viele „direct path read“ durchführt und man überprüft hat, dass ein Großteil der „physical read bytes“ durch „cell simulated physical IO bytes eligible for predicate offload“ abgedeckt sind. Seine Effizienz hängt davon ab, was gefiltert werden soll.

Projection Offloading

Die Projektion ist die Auswahl der Spalten, die für die SELECT-Clause notwendig ist. Man geht beispielsweise mit dem oben genannten Parameter in den Simulationsmodus und startet einen „SELECT *-Befehl ohne WHERE-Clause. Listing 1 zeigt den Ausführungsplan (Format: „allstats last +predicates“). Mit „+projection“ werden alle Spalten des SELECT-Befehls angezeigt. Tabelle 1 zeigt die wesentlichen Statistiken bei der Ausführung.

Alles wurde dem SmartScan unterworfen, ohne dass etwas gefiltert worden wäre. Man hat damit also nichts gewon-

nen. Jetzt folgt nochmal das Gleiche, aber nur mit wenigen Spalten („SELECT ID“, „SEQ FROM DEMO2“, siehe Listing 2).

Die Spalten sind in der Projektion des Vorgangs „STORAGE“ sichtbar und diese Projektion wurde somit – via iDB – mit einem E/A-Abruf übermittelt. Tabelle 2 zeigt das Ergebnis. Auf dem betreffenden Volume wurden von SmartScan nur 23 Prozent zurückgegeben. Andere Spalten als „ID“ und „SEQ“ wurden als Reaktion auf den E/A-Abruf von den an die Datenbank zurückgegebenen Blöcken entfernt. Es ist zu beachten, dass die Statistiken der Simulation zwar „predicate offload“ heißen, tatsächlich aber

alle „offload“ Predicate und Projection enthalten. Das hier ist der Beweis.

Predicate Offloading

Bei der nächsten Abfrage kommt ein Prädikat „WHERE MOD(ID,2)=0“ hinzu, um die Hälfte der Zeilen zu beseitigen (siehe Listing 3). Man sieht im Ausführungsplan, dass das Prädikat durch den Speichervorgang verarbeitet werden kann. Tatsächlich beträgt das von SmartScan zurückgegebene Volumen nur die Hälfte von zuvor (siehe Tabelle 3).

Statistic	Total
cell simulated physical IO bytes eligible for predicate offload	2,340,585,472
cell simulated physical IO bytes returned by predicate offload	2,121,558,080
physical reads bytes	2,346,206,208
physical reads direct	285,716

Tabelle 1

```

-----
| Id | Operation                               | Name | Starts | E-Rows | A-Rows | Buffers | Reads |
-----
| 0 | SELECT STATEMENT                         |      | 1      | 2000K  | 2000K  | 285K    | 285K  |
| 1 | TABLE ACCESS STORAGE FULL              | DEMO2 | 1      | 2000K  | 2000K  | 285K    | 285K  |
-----

Column Projection Information (identified by operation id):
-----
1 - "ID"[NUMBER,22], "DEMO2"."SEQ"[NUMBER,22], "DEMO2"."D"[NUMBER,22],
"DEMO2"."E"[NUMBER,22], "DEMO2"."F"[VARCHAR2,1000], "DEMO2"."G"[VARCHAR2,4000]
    
```

Listing 1

```

-----
| Id | Operation | Name | Starts | E-Rows | A-Rows | Buffers | Reads |
-----
| 0 | SELECT STATEMENT | | 1 | 2000K | 2000K | 285K | 285K |
| 1 | TABLE ACCESS STORAGE FULL | DEMO2 | 1 | 2000K | 2000K | 285K | 285K |
-----

Column Projection Information (identified by operation id):
-----
1 - "ID"[NUMBER,22], "SEQ"[NUMBER,22]

```

Listing 2

```

-----
| Id | Operation | Name | Starts | E-Rows | A-Rows | Buffers | Reads |
-----
| 0 | SELECT STATEMENT | | 1 | 1000K | 1000K | 285K | 285K |
|* 1 | TABLE ACCESS STORAGE FULL | DEMO2 | 1 | 1000K | 1000K | 285K | 285K |
-----

Predicate Information (identified by operation id):
-----
1 - storage(MOD("ID",2)=0)
   filter(MOD("ID",2)=0)

Column Projection Information (identified by operation id):
-----
1 - "ID"[NUMBER,22], "SEQ"[NUMBER,22]

```

Listing 3

Join Offloading

Es wird hier nicht weiter ins Detail gegangen, man muss allerdings bedenken, dass die Reaktionszeit einer Abfrage sich nicht aus dem Lesen der Daten ergibt. Bei komplexen Abfragen gibt es Verknüpfungen, Sortierungen und Aggregation. Das alles geschieht nur auf dem Datenbankservers mithilfe von Tempfiles. SmartScan bringt hier nichts. Für die Filterung der Verknüpfungen gilt eine Ausnahme: Bei der Erstellung der Hash-Tabelle (die kleinere der beiden Tabellen eines „HASH JOIN“) wird ein Bloom-Filter erstellt, mit dem beim Lesen der zweiten Tabelle viele Zeilen entfernt werden können, selbst ohne die Entsprechung mit der Hash-Tabelle vorzunehmen. Dieser Bloom-Filter ist wie ein Prädikat: Er wird auf dem Speicherserver ausgeführt, wenn SmartScan läuft. Man kann diesen Filter als große „IN ()“-Clause verstehen, die als erster Grobfilter funktioniert.

Abbildung 1 zeigt die Funktionsweise: Wenn die Hash-Tabelle erstellt wird, entsteht ein Filter, der diese Tabelle darstellt, aber False Positives enthalten kann. Verwendet SmartScan die zweite Tabelle (was

Statistic	Total
cell simulated physical IO bytes eligible for predicate offload	2,340,585,472
cell simulated physical IO bytes returned by predicate offload	55,272,816
physical reads bytes	2,341,306,368
physical reads direct	285,716

Tabelle 2

Statistic	Total
cell simulated physical IO bytes eligible for predicate offload	2,340,585,472
cell simulated physical IO bytes returned by predicate offload	27,967,088
physical reads bytes	2,343,059,456
physical reads direct	285,716

Tabelle 3

das an die Datenbank zu übertragende Volumen begrenzt), brauchen nur noch die False Positives entfernt zu werden.

Betrachtet man nur die „wait events“ bei Verknüpfungen zwischen zwei großen

Tabellen, wird man viele „direct path read temp“ sehen, die keinem SmartScan unterworfen sind. Wenn man allerdings mit der Simulationsbibliothek fortfährt, wird man sehen, dass der Bloom-Filter viel-

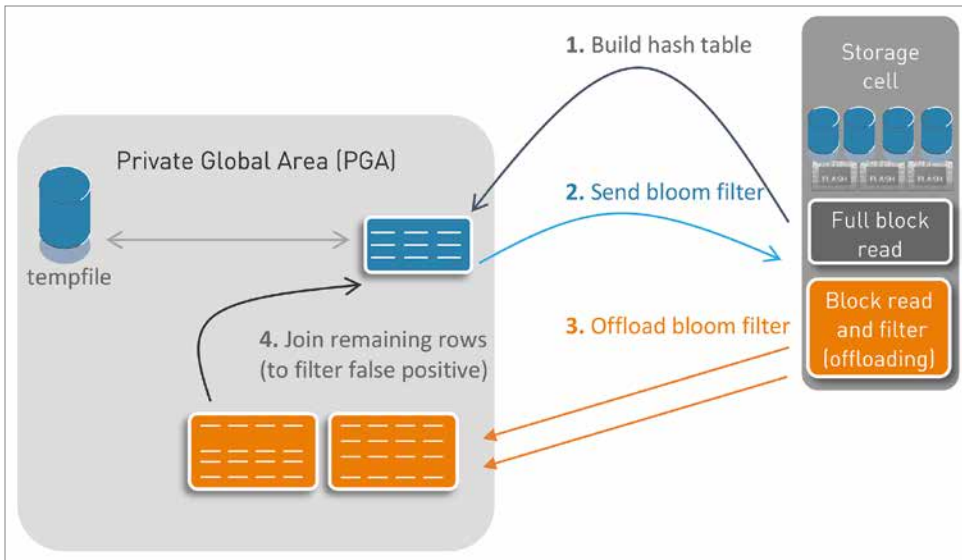


Abbildung 1: Bloom-Filter für Hash Join Offloading

leicht einige dieser Lese-/Schreibvorgänge der Tempfiles ersparen wird.

Betrachten wir nochmals die Simulationsstatistiken. Der Bloom-Filter erscheint nur im Parallel Query bis 12.1.0.2, wo er im Serial für In-Memory verwendet wird. Tatsächlich gibt es jedoch weitere Fälle, in denen man ihn im Serial beobachten kann.

Beim Auslesen einiger Spalten einer Tabelle, beim Lesen von mehr als ein paar Zeilen oder beim „HASH JOIN“ zwischen großen Tabellen – hier wird klar, bei welcher Art von Datenbanken SmartScan seine Vorteile aufzeigt: für Reporting, Data Warehouse (ETL und Abfragen), analytische Abfragen, DSS etc.

Bei einer ERP-Datenbank, die alle Spalten („SELECT*“) einiger Zeilen („WHERE PK=...“) auslesen wird, bringt SmartScan keine Vorteile. Der Autor kennt eine Banking-Anwendung, die alles als CLOB (XML) speichert und die nur über den Primärschlüssel zugreift. Dafür kann man Exadata vergessen, ebenso In-Memory. Beides ist nicht dafür gemacht. Heute gibt es für diese Datenmodelle NoSQL-Datenbanken, die kurz vor den relationalen SQL-Datenbanken entworfen wurden. Bei ERP- oder anderen Datenbanken, die die Parameter „optimizer_index_caching,optimizer_index_cost_adj“ definieren, um den Zugriff über den Index zu erzwingen, wird SmartScan ebenso wenig etwas bringen.

Event	Waits	Time(s)	Avg Wait(ms)	% Total Call Time	Wait Class
read by other session	46,201,741	158,777	3	50.2	User I/O
db file sequential read	87,834,244	102,268	1	32.3	User I/O
CPU-Zeit		27,646		8.7	
db file scattered read	6,313,301	5,369	1	1.7	User I/O
enq: TX - row lock contention					

Tabelle 4

Event	Waits	Time(s)	Avg Wait(ms)	% Total Call Time	Wait Class
DB CPU		1,685		49.63	
direct path read	347,955	347,955	3	27.26	User I/O
db file sequential read	599,906	271	0	7.99	User I/O
enq: TX - row lock contention	185	89	481	2.62	Application
log file sync	23,503	76	3	2.24	Commit

Tabelle 5

Statistic	Total
cell physical IO interconnect bytes	221,656,831,180
cell simulated physical IO bytes eligible for predicate offload	336,000,856,474
cell simulated physical IO bytes returned by predicate offload	187,871,446,630
physical reads bytes	361,291,243,520
physical write bytes	1,785,430,016

Tabelle 6

Den SmartScan-Vorteil bewerten

Man nimmt einen AWR-Bericht und wirft einen Blick auf den Abschnitt „Top Events“. Die Angabe „%DB Time“ gegenüber „direct path read“ zeigt den maximalen Vorteil, den man erwarten kann. Dazu einige Beispiele.

Tabelle 4 zeigt die „Top 5 Timed Events“. Diese Basis hat einen eingeschränkten Puffer-Cache, führt aber meist Single-Block-Lesevorgänge durch. SmartScan bringt hier nichts. Man muss zuerst diese Lesevorgänge verstehen. Ein fehlerhafter Ausführungsplan wird möglicherweise immer wieder die gleichen Zeilen lesen. Auch der Kauf eines neuen Rechners wird die Feinabstimmung nicht vermeiden.

In Tabelle 5 sind die „Top 5 Timed Foreground Events“ dargestellt. Hier kann SmartScan 27 Prozent der Reaktionszeit eindämmen. Es wäre also inter-

essant weiterzugehen, um den Gewinn einzuschätzen. Dazu wird die Simulation mit „_rdbs_internal_fplib_enabled“ aktiviert. Man vergleicht das von SmartScan zurückgegebene Volumen mit den „physical read bytes“, um die Effizienz des Filtervorgangs bewerten zu können.

Tabelle 6 zeigt die interessanten Statistiken. Es mussten 336 GB gelesen werden („physical reads bytes“) und dennoch wurden nur 206 GB zwischen der Speicher- (hier simuliert durch „fplib“) und der Datenbank-Schicht übertragen. Genauer gesagt: Diejenigen Schreibvorgänge müssen entfernt werden, die in „cell physical IO interconnect bytes“ ebenfalls enthalten sind. Das vorgestellte Beispiel hat keine ASM-Redundanz, sodass das durch Schreibvorgänge übertragene Volumen den „physical write bytes“ entspricht. Bei normaler Redundanz hätte man es mit „zwei“ multiplizieren müssen.

Man musste also 336 GB auslesen, von denen 313 GB durch SmartScan verarbeitet werden. 175 GB wurden von SmartScan zurückgegeben; $206 - 1 \times 1,6 - 175 = 29$ GB wurden zurückgegeben, ohne von SmartScan bearbeitet worden zu sein. Durch Übernahme der Gesamtzahl der Lese-/Schreibvorgänge braucht man schließlich nur $206 / (336 + 1,6) = 61$ Prozent des notwendigen Volumens zu übertragen.

Angesichts der Tatsache, dass die E/A 27,26 und 7,99 Prozent der „DB Time“ ausmachen, ist ein Gesamtgewinn von $(0,2726 + 0,0799) \times 0,61 = 21,5$ Prozent der „DB Time“ mit Exadata zu erwarten. Das ist gut, man sollte allerdings gegenüber seinen Benutzern keine Verpflichtung über diesen Wert hinaus eingehen.

Laufende Änderungen

Die Simulation dient nur zur Bewertung des gefilterten Datenvolumens. Die Effizienz von SmartScan hängt auch von den gleichzeitigen Änderungen ab. Zuerst war zu erkennen, dass der Checkpoint im Falle einer Vielzahl veränderter Blöcke zu unwirksam wäre und SmartScan nicht durchgeführt wird. Es gibt jedoch auch das Problem der Konsistenz: Man kann weder die nicht durchgeführten Änderungen noch die nach Beginn der Abfrage durchgeführten Änderungen sehen. So kann es plötzlich sein, dass SmartScan den vollständigen Block zu einem bestimmten Zeitpunkt ohne jede Filterung zurückgeben muss, weil nur

Event	Waits	Time(s)	Avg Wait(ms)	% Total Call Time	Wait Class
DB CPU		2.2		54.3	
enq: KO - fast object checkpoint	1	1.7	1690.24	42.7	Application
direct path read	1,130	.3	0.28	8.0	User I/O
db file sequential read	535	.2	0.30	4.0	User I/O
SQL*Net more data to client	402	.1	0.12	1.3	Network
enq: RO - fast object reuse	2	0	1.56	.1	Application
SQL*Net message to client	213	0	0.01	.0	Network
reliable message	3	0	0.43	.0	Other
control file sequential read	137	0	0.01	.0	System I/O
log file sync	1	0	0.73	.0	Commit

Tabelle 7

der Code der Datenbank ein konsistentes Image rekonstruieren kann. *Tabelle 7* zeigt mit „Top 10 Foreground Events by Total Wait Time“ einen AWR-Auszug dieses Falls.

Man sieht, dass nur 8 Prozent der Zeit auf den SmartScan („direct path read“) entfallen. Man hat aber 42 Prozent der Zeit für die Einrichtung dieses Checkpoints aufgewendet, um diese „direct path read“ durchführen zu können. Es geht allerdings noch schlechter. In *Tabelle 8* sind alle Statistiken zu „cell“ in den Simulationsmodus eingebracht.

Die Statistiken „cell blocks processed by ...“ geben das von jeder der verschiedenen SmartScan-Schichten verarbeitete Volumen an. Es wurden 142.870 Blöcke im Direktmodus gelesen. Alle diese Blöcke wurden durch die „cache layer“ verarbeitet, aber ausgehend von „transaction layer“ ergeben sich weniger Blöcke.

SmartScan hat Blöcke gesehen, deren SCN außerhalb der SCN der Abfrage liegt. Diese Blöcke müssen ein „undo“ anwenden, um ein konsistentes Image zu rekonstruieren. Die „storage cell“ kann dies nicht machen: Sie hat keinen Zugriff auf die Festplatten der anderen und kann daher nicht das gesamte „undo“ einsehen.

SmartScan war hier also deutlich weniger effektiv als erwartet. Beim Betrachten der Zahlen versteht man, warum weiter oben komplizierte Berechnungen durchgeführt wurden, um die Effizienz zu messen. Einige Tools (einschließlich Oracle)

erstellen einfach das Verhältnis „eligible“ zu „returned“, vergessen in diesem Fall aber alle betroffenen Blöcke, die nicht von SmartScan verarbeitet wurden.

Das Verhältnis erscheint gut: Die Hälfte des Volumens wird von SmartScan zurückgegeben. Tatsächlich entspricht aber das ausgetauschte Volumen („cell physical IO interconnect bytes“) selbst nach Entfernen der Schreibvorgänge („physical write bytes“) dem gesamten betreffenden Volumen. In diesem Beispiel gibt es also keinen Vorteil. Nur einen Zeitverlust durch den Checkpoint.

SmartScan ist nicht für die Optimierung einer OLTP geschaffen, bei der die Änderungen genauso wichtig sind wie die Lesevorgänge. Deswegen müssen den Blöcken die Funktionen zugewiesen werden, die auf Ebene der Speicherserver nicht verfügbar sind.

Fazit

Man muss verstehen, wo SmartScan angewendet werden kann und wo nicht. Denn dies ist der wichtigste Faktor der Leistungsverbesserung mit Exadata im Vergleich zu jeder anderen Plattform. Wie man also sieht, ist Exadata eher für Data Warehouse gemacht:

- Full Table Scan oder Index Fast Full Scan
- Parallel Query oder Serial Direct Read

Statistic	Total	per Second	per Trans
cell IO uncompressed bytes	1,170,391,040	213,965,455.21	97,532,586.67
cell blocks helped by minscn optimization	10	1.83	0.83
cell blocks processed by cache layer	142,870	26,118.83	11,905.83
cell blocks processed by data layer	78,128	14,283.00	6,510.67
cell blocks processed by txn layer	78,128	14,283.00	6,510.67
cell commit cache queries	36,182	6,614.63	3,015.17
cell physical IO interconnect bytes	1,215,807,488	222,268,279.34	101,317,290.67
cell scans	5	0.91	0.42
cell simulated physical IO bytes eligible for predicate offload	1,170,391,040	213,965,455.21	97,532,586.67
cell simulated physical IO bytes returned by predicate offload	539,262,144	98,585,401.10	44,938,512.00
physical read bytes	1,174,036,480	214,631,897.62	97,836,373.33
physical reads	143,315	26,200.18	11,942.92
physical reads cache	445	81.35	37.08
physical reads direct	142,870	26,118.83	11,905.83
physical write bytes	23,306,240	4,260,738.57	1,942,186.67

Tabelle 8

- Wenige gleichzeitige Aktualisierungen
- Kompression

Man wird eher zu Exadata raten, wenn man bereits alles getan hat, was für die vorhandene Infrastruktur möglich war. Es gilt zunächst, Parallel Query zu beherrschen, um alle lizenzierten CPU zu verwenden. Eventuell bietet sich auch RAC an, um die Leistung mehrerer Server einzusetzen. Wenn das optimal läuft und es keine weitere Möglichkeit mehr als die E/A-Bandbreite auf den „direct path read“ gibt, kann man wirklich auf die schnellere Geschwindigkeit mit Exadata umsteigen. Der Autor hat einen Kunden, dem genau das passiert ist, als es um die Erfassung von Leistungsproblemen ging. Bei einer OLTP-Last wird SmartScan keine Vorteile bringen, da es folgende Punkte nicht verbessert:

- „SELECT*-Anfragen
- Zugriffe auf wenige Zeilen über den Index (etwa durch PK)
- Optimierungen in „FIRST_ROWS“
- Gleichzeitige Aktualisierungen mit dem Puffer-Cache und Undo, beides sehr wirksamen Mechanismen. Dank derer hat Oracle den Markt der transaktionalen Datenbanken beherrscht (außer den Mainframes).

Natürlich kann man Exadata wählen, um sein Data Warehouse zu beschleunigen, und mit der Absicht einer Konsolidierung auch seine OLTP-Datenbanken integrieren. Selbst wenn SmartScan nicht angewendet wird, kann man dennoch immer von einer leistungsstarken Maschine profitieren. Bei OLTP-Lasten muss man mit Exadata jedoch drei Dinge beachten:

- Nicht vergessen, dass die HCC-Komprimierung für Tabellen erfolgt, die in erster Linie schreibgeschützt sind. Deutlich ausgedrückt: Bis heute gibt es noch keine Kompression, die effizient auf Änderungen reagiert.
- Der Wechsel auf RAC bedeutet nicht, dass alle Dienste auf allen Knoten geöffnet werden müssen. Die Arbeit an den gleichen Blöcken mit verschiedenen Instanzen wird alle vorhandenen Beschränkungen verstärken.
- Nicht alle Index-Dateien löschen. Unter OLTP dienen die Index-Dateien auch dazu, Einschränkungen bei der Eindeutigkeit zu validieren, ein Sperren der Tabellen zu vermeiden etc.

Vor einer Entscheidung über eine neue Infrastruktur gilt es zunächst, die aktuellen Datenbanken zu prüfen und zu versu-

chen, den möglichen Gewinn einzuschätzen. Man kann durch Einblick in einen AWR-Bericht eine erste Vorstellung gewinnen und mehr erfahren, wenn man SmartScan simuliert sowie einige Berechnungen mit den entsprechenden Statistiken durchführt.

Der Artikel soll kein negatives Bild von Exadata vermitteln, bei der es sich um eine außergewöhnliche Maschine handelt. Natürlich kann man auch seine OLTP-Datenbanken einbringen, damit wird man allerdings keine herausragenden Leistungen erzielen. Ein vereinfachender Vergleich: Man kann mit dem Sportwagen auch durch die Stadt fahren, aber dafür hat man ihn nicht erworben.



Franck Pachot

franck.pachot@dbi-services.com

Tipps und Tricks aus Gerds Fundgrube

Heute: PLL vs. PLX

Gerd Volberg, OPITZ CONSULTING Deutschland GmbH

Libraries, die in Oracle Forms eingebunden werden, dürfen zur Laufzeit als PLL oder PLX hinterlegt werden. Beide Formate sind unterstützt. Aber welche Variante ist besser?

Schauen wir uns vor der Beantwortung dieser Frage zwei Fälle an:

Beispiel 1: Bei der Entwicklung werden sowohl PLL als auch PLX genutzt. Der Forms Builder benutzt das Arbeitsverzeichnis „C:\Forms“ und im „FORMS_PATH“ steht der Verweis auf „C:\Forms\Lib“. Die Sourcecodes heißen „lib.pll“, „menu.mmb“ und „emp.fmb“, die generierte Library „lib.plx“. Die Libraries werden in folgender Reihenfolge gesucht (*siehe Abbildung 1*):

1. Wenn die „lib.plx“ im Arbeitsverzeichnis liegt, wird sie genutzt
2. Wird sie nicht gefunden, suche im „FORMS_PATH“ nach der PLX
3. Wird sie nicht gefunden, suche im Arbeitsverzeichnis nach der PLL
4. Wird sie nicht gefunden, suche im „FORMS_PATH“ nach der PLL
5. Wird sie nicht gefunden, erzeugt Forms eine Fehlermeldung

Dies gilt genauso für Forms-Masken und Menüs, jedoch mit der Ausnahme, dass dort nur die kompilierten Versionen FMX und MMX relevant sind.

Wenn das PLX im lokalen Verzeichnis nun eine ältere Version ist als die entsprechende PLL am gleichen Ort, dann führt dies zu großen Problemen während der

Laufzeit. Den Fehler zu finden, ist nahezu unmöglich, solange der Entwickler die Zeitstempel der Libraries nicht überprüft.

Beispiel 2: Bei der Entwicklung wird nur PLL eingesetzt. Während der Entwicklung und zur Laufzeit hat man keine Probleme. Wenn eine Library nicht im Arbeitsverzeichnis liegt, wird sie im „FORMS_PATH“ gefunden.

Fazit

Der Autor nutzt keine PLX mehr, da er in zu vielen Projekten mit den beschrie-

benen Problemen konfrontiert war. Die Fehlersuche bei größeren Forms-Pfaden ist enorm aufwändig, wenn dort irgendwo eine veraltete PLX liegt.

Viele Entwickler sind der Meinung, dass PLX schneller ist. Dies konnte er bisher nicht nachvollziehen und hatte auch noch nie Performance-Probleme. Andere Entwickler sagen, PLX sei kompilierter Code, somit kann niemand den Sourcecode einsehen oder stehlen – in diesem Fall empfiehlt der Autor, einmal Folgendes auszuprobieren: Eine Library mit folgendem Package erzeugen und daraus eine PLX generieren (*siehe Listing 1*).

Sucht man in der PLX mit einem Editor nach „HiddenPW“, wird man sowohl den Namen der Konstanten als auch deren Wert finden. Eine PLX schützt also diesbezüglich nicht. Aus Sicht des Autors überwiegen die Vorteile, wenn man ausschließlich mit PLLs arbeitet.

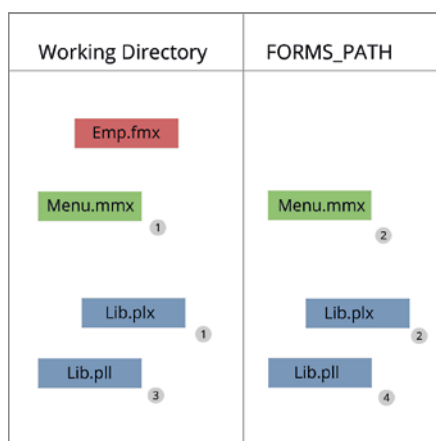


Abbildung 1: Die Such-Reihenfolge

```
PACKAGE Const IS
  HiddenPW      CONSTANT VARCHAR2 (100) := 'HiddenPW';
END Const;
```

Listing 1



Gerd Volberg

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

Wir begrüßen unsere neuen Mitglieder

Persönliche Mitglieder

- Christoph Förster
- Brian Skelton
- Franz Wiczorek
- Dirk Skirde
- Sinem Kolb
- Eva Brüning
- Jörn Scheewe
- Yuriy Romanenko
- Bernhard Pietsch
- Daniel Hochleitner
- Stefan Hess
- Michael Möller
- Sai Janakiram Penumuru
- Alexander Asarin
- Uwe Schierz
- Björn Bröhl
- Alexander Rudolf
- Andreas Ralf Schmidt
- Ingo Peter
- Andreas Steeger
- Sebastian Ramsauer
- Janek Winz
- Oliver Kuiper
- Björn Berg
- Hao Pan

Firmenmitglieder DOAG

- PRODATO Integration Technology GmbH, Markus Schneider
- TimoCom Soft- und Hardware GmbH, Jörg Alexander
- DR. JOHANNES HEIDENHAIN GmbH, Max Beisse
- Siemens AG, Jörg Danzenbächer
- ELDICON Systemhaus GmbH, Jochen Langer
- Promata GmbH, Michael Wolf

Neumitglieder SOUG

- Andy Kielhorn, Liechtensteinische Landesbank AG

Impressum

Red Stack Magazin wird gemeinsam herausgegeben von den Oracle-Anwendergruppen DOAG Deutsche ORACLE-Anwendergruppe e.V. (Deutschland, Tempelhofer Weg 64, 12347 Berlin, www.doag.org), AOUG Austrian Oracle User Group (Österreich, Lassallestraße 7a, 1020 Wien, www.aoug.at) und SOUG Swiss Oracle User Group (Schweiz, Dornacherstraße 192, 4053 Basel, www.soug.ch).

Red Stack Magazin ist das User-Magazin rund um die Produkte der Oracle Corp., USA, im Raum Deutschland, Österreich und Schweiz. Es ist unabhängig von Oracle und vertritt weder direkt noch indirekt deren wirtschaftliche Interessen. Vielmehr vertritt es die Interessen der Anwender an den Themen rund um die Oracle-Produkte, fördert den Wissensaustausch zwischen den Lesern und informiert über neue Produkte und Technologien.

Red Stack Magazin wird verlegt von der DOAG Dienstleistungen GmbH, Tempelhofer Weg 64, 12347 Berlin, Deutschland, gesetzlich vertreten durch den Geschäftsführer Fried Saacke, deren Unternehmensgegenstand Vereinsmanagement, Veranstaltungsorganisation und Publishing ist.

Die DOAG Deutsche ORACLE-Anwendergruppe e.V. hält 100 Prozent der Stammeinlage der DOAG Dienstleistungen GmbH. Die DOAG Deutsche ORACLE-Anwendergruppe e.V. wird gesetzlich durch den Vorstand vertreten; Vorsitzender: Dr. Dietmar Neugebauer. Die DOAG Deutsche ORACLE-Anwendergruppe e.V. informiert kompetent über alle Oracle-Themen, setzt sich für die Interessen der Mitglieder ein und führen einen konstruktiv-kritischen Dialog mit Oracle.

Redaktion:

Sitz: DOAG Dienstleistungen GmbH
(Anschrift s.o.)

Chefredakteur (ViSdP): Wolfgang Taschner

Kontakt: redaktion@doag.org

Weitere Redakteure (in alphabetischer Reihenfolge): Julia Bartzik, Gaetano Bisaz, Mylène Diacquenod, Marina Fischer, Klaus-Michael Hatzinger, Sebastian Höing, Dr. Dietmar Neugebauer, Jan Peterskovsky, Fried Saacke

Titel, Gestaltung und Satz:

Alexander Kermas, DOAG Dienstleistungen GmbH (Anschrift s.o.)

Fotonachweis:

Titel: © iconimage/fotolia.com
Foto S. 6: © JFL Photography/fotolia.de
Foto S. 7: © Songquan Deng/123rf.com
Foto S. 10: © faithie/123rf.com
Foto S. 16: © Freepik/freepik.com
Foto S. 22: © alphaspirit/fotolia.com
Foto S. 31: © Daniel Schwarz/123rf.com
Foto S. 35: © rawpixel/123rf.com

Foto S. 40: © Tommaso Altamura/fotolia.com
Foto S. 48: © Dusanka Visnjican/123rf.com
Foto S. 63: © venimo/123rf.com

Anzeigen:

Simone Fischer, DOAG Dienstleistungen GmbH (verantwortlich, Anschrift s.o.)
Kontakt: anzeigen@doag.org
Mediadaten und Preise unter:
www.doag.org/go/mediadaten

Druck:

adame Advertising and Media GmbH,
www.adame.de

Alle Rechte vorbehalten. Jegliche Vervielfältigung oder Weiterverbreitung in jedem Medium als Ganzes oder in Teilen bedarf der schriftlichen Zustimmung des Verlags. Die Informationen und Angaben in dieser Publikation wurden nach bestem Wissen und Gewissen recherchiert. Die Nutzung dieser Informationen und Angaben geschieht allein auf eigene Verantwortung. Eine Haftung für die Richtigkeit der Informationen und Angaben, insbesondere für die Anwendbarkeit im Einzelfall, wird nicht übernommen. Meinungen stellen die Ansichten der jeweiligen Autoren dar und geben nicht notwendigerweise die Ansicht der Herausgeber wieder.

Inserentenverzeichnis

DBConcepts GmbH www.dbconcepts.at	S. 38	E-3 Magazin www.e-3.de	S. 47	ORACLE www.oracle.com	U 2
dbi services ag www.dbi-services.com	S. 28	IDG / Computerwoche www.promatis.de	S. 58	Softbase A/S www.softbase.com	S. 21
DOAG e.V. www.doag.org	U 3	MuniQsoft GmbH www.muniqsoft.de	S. 3	Trivadis GmbH www.trivadis.com	U 4

DOAG FINANCIAL DAY 2016

4. + 5. Oktober 2016 in Stuttgart



Genau unser Ding: Internet of Things.



■ In Zukunft wird sich alles noch mehr um Daten drehen. Bestes Beispiel: das IoT. Es wird neue intelligente Vernetzungen und Prozesse schaffen. Es wird zu höherer Automatisierung und Effizienz führen. Mehr Komfort und Sicherheit ermöglichen. Wir bringen alle diese Daten und Dinge für Sie zusammen. Sprechen Sie mit uns über Ihre IoT-Projekte.

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. ■ ■ ■