

Red Stack

Magazin

DOAG

SOUG
swiss oracle
user group

AOUG
AUSTRIAN ORACLE USER GROUP

Jetzt inklusive BUSINESS NEWS



Aus der Praxis

Fallen Sie nicht
in die Cloud



Im Interview

Holger Nicolay,
Business Development
Manager, Interxion

BUSINESS NEWS

Auf Datenschatzsuche
mit Deep Learning



2020
DOAG
Konferenz + Ausstellung

SAVE THE
DATE

17. - 20. Nov 2020





Robert Marz
Themenverantwortlicher
Cloud, DOAG DB Community

Liebe Mitglieder, liebe Leserinnen und Leser,

seit 2011 steht der Begriff Cloud-Computing bereits im deutschen Duden. Die große Mehrheit der deutschen Unternehmen nutzt mittlerweile Cloud-Dienste in unterschiedlichem Umfang. Die Zahl der Totalverweigerer sinkt stetig. Die meisten Schreckensszenarien haben sich nicht bewahrheitet: Die Daten sind in der Cloud so sicher wie im eigenen Rechenzentrum und Administratoren werden immer noch gebraucht. Die meist überzogenen Einsparpotenziale konnten im Gegenzug jedoch auch nicht realisiert werden. Es hat Zeit gebraucht, aber langsam wird die Cloud als das akzeptiert, was sie ist: flexible Servicemodelle mit virtuell unendlichen Ressourcen, die on demand bereitgestellt werden können. Im Gartner Cloud Hype Cycle von 2019 sind SaaS und Cloud-Office auf dem Plateau der Produktivität angekommen und eine ganze Reihe von Kandidaten befinden sich bereits auf dem Pfad der Erleuchtung. In derselben Veröffentlichung hat Multicloud den Gipfel der überzogenen Erwartungen bereits passiert. Die Multicloud-Allianz von Oracle Cloud Infrastructure und MS Azure funktioniert gut, wird von den Kunden angenommen und in weitere Regionen ausgerollt.

Wir von den deutschsprachigen Oracle-Usergruppen müssen Ihnen, unseren Mitgliedern, nicht mehr erklären, was die Cloud ist oder wie man mit ersten Anwendungen dahin kommt. Stattdessen bieten wir Ihnen in diesem Heft eine breite Vielfalt an Themen von Strategie bis hin zu technischen Detaillösungen aus vielen Bereichen. Und ich finde, besser kann man nicht deutlich machen, dass die Cloud zur Normalität geworden ist.

In der Business News der vorliegenden Ausgabe dreht sich diesmal alles um künstliche Intelligenz. Dabei erläutern wir die unterschiedlichen Begrifflichkeiten wie Machine Learning und Deep Learning, stellen ein Start-up vor und schauen uns an, wie die KI in der Cloud den Content analysiert und verwaltet.

Viel Spaß beim Lesen dieser Ausgabe und bleiben Sie neugierig.

Ihr Robert Marz



Training

Training

MUNIQSOFT

TRAINING

ORACLE
Silver Partner

20 Jahre Oracle-Datenbankschulungen von Experten
Effizient und kundenorientiert

Offene Kurse demnächst:

| | | |
|------------------------|-------------------|-----------------|
| SQL I incl. Vertiefung | 11.05.-15.05.2020 | 1.780,- € netto |
| APEX Kompakt | 18.05.-20.05.2020 | 1.590,- € netto |
| DBA III | 22.06.-26.06.2020 | 1.990,- € netto |
| PL /SQL Grundlagen | 06.07.-10.07.2020 | 1.990,- € netto |
| DBA Sommerspecial | 20.07.-24.07.2020 | 1.590,- € netto |
| Postgres SQL | 27.07.-29.07.2020 | 1.690,- € netto |

☎ 089 679090-40

Website: www.muniqsoft-training.de

Tipps: www.muniqsoft-training.de/tipps

Schulungszentrum

Muniqsoft Training GmbH
Grünwalder Weg 13a
82008 Unterhaching/München

Mehr Oracle Schulungstermine unter
muniqsoft-training.de

Auf Anfrage bieten wir auch gerne individuelle Inhouse Schulung an



7

Interview mit Holger Nicolay



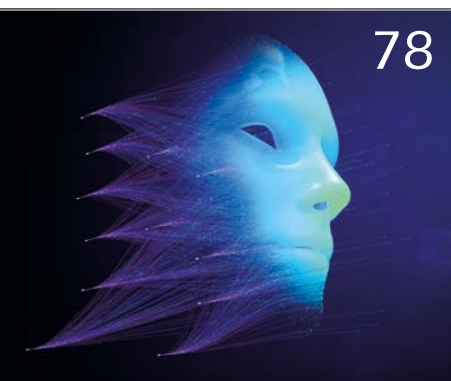
11

Mit der Nutzung einer Multi Cloud kann man die Vorteile der einzelnen Cloud-Anbieter voll ausschöpfen



24

Über den Einsatz von OpenShift hat das Entwicklerteam seine eigene Cloud mit den Vorteilen aus allen Welten bekommen



78

Content-Veredelung mit Hilfe von KI: Das Start-up aiconix macht es möglich

Einleitung

- 3 Editorial
- 5 Timeline
- 6 Aus der Ferne betrachtet: Cloud-Computing ist ein Bombengeschäft
Günther Stürner
- 7 „Technologisch werden Unternehmen stets IT-Experten brauchen, die Cloud-Technologie verstehen und betreiben können, selbst wenn weite Teile der Planung, Migration und des Betriebs an einen Cloud-Partner ausgelagert werden.“
Interview mit Holger Nicolay

Cloud

- 11 Fallen Sie nicht in die Cloud
Michael Skowasch
- 18 Infrastruktur-Integration für hybride Oracle Clouds
Holger Nicolay
- 24 Peterchens Mondfahrt an der Cloud vorbei
Wolf G. Beckmann

Development

- 31 Cloud-native Microservice-Entwicklung mit Helidon
Sven Bernhardt
- 37 Die eigene ADF-Anwendung auf Wolke 7 bringen
Markus Klenke

Engineered Systems

- 42 Oracle Private Cloud Appliance – Einfach ein bisschen Hardware mit Oracle VM?
Martin Bracher
- 48 Hilfreiche Tools für den täglichen Exadata-Betrieb (Teil 2)
Michael Schulze

APEX

- 55 Die passende Architektur für APEX wählen
Niels de Bruijn

- 59 fabe – for all a beautiful earth
Tobias Arnhold, Florian Graßhoff und Julia Chevalier

Datenbank

- 62 Spielerisch leichte Security für Oracle Datenbanken – komplexe Oracle Features genial einfach nutzen
Thomas Petrik und Wolfgang Klinger
- 68 Modernes Datenmanagement mit der Open-Source-basierten Daten- und KI-Plattform IBM Cloud Pak for Data
Nadine Brehm und Claus Huempel
- 70 Das Pink Database Paradigma (PinkDB): Die Datenbank durch die rosa Brille
Interview mit Philipp Salvisberg
- 72 Das war die DOAG 2020 Noon2Noon
Martin Klier
- 73 „Die Cloud an sich ist keine Lösung, sondern eine Technologie“
Interview mit Frits Hoogland

Auf Datenschuttsuche mit Deep Learning

BUSINESS NEWS

- 74 Künstliche Intelligenz, Machine Learning und Deep Learning: Was sind die Unterschiede und wozu kann was verwendet werden?
Björn Bröhl
- 78 Jäger des verborgenen Schatzes: Mithilfe von KI schürft aiconix Daten-Gold aus Archiven. Inklusive aiconix-Interview
Elvira Steppacher
- 83 Cloud-Computing als Fundament für Analytics
Rolf Scheuch
- 88 Machine und Deep Learning mit Data Science in der Oracle Cloud
Alfred Schlaucher

Intern

- 93 Termine und neue Mitglieder
- 94 Impressum und Inserenten

Timeline

06. März 2020

Die erste DOAG-Vorstandssitzung des Jahres findet in Berlin statt. Einen Schwerpunkt bildet die Fortführung der Entwicklung der Strategie DOAG 2024 mit dem Schwerpunkt Veranstaltungen. Diese Ergebnisse sollen in das anstehende Leitungskräfteforum einfließen und gemeinsam weiterbearbeitet werden. Eine wesentliche Maßnahme wird sein, Arbeitsgruppen und Programmkomitees zu bilden, um insbesondere die DOAG Konferenz + Ausstellung neu und stärker auszurichten sowie die CloudLand im nächsten Jahr zu einem der bedeutendsten Events zu entwickeln. Die vorgeschlagenen Maßnahmen sollen dann auch der Delegiertenversammlung vorgelegt werden, welche den Auftrag zur Entwicklung der Strategie DOAG 2024 gegeben hat.

10. März 2020

Der DOAG Database Day zur Oracle Standard Edition Two findet am 10. März in Hannover statt. Die Teilnehmer erfahren in praxisorientierten Vorträgen, was die Standard Edition Two (SE2) von ihrer großen Schwester unterscheidet und wie man die Datenbank auch ohne RAC hochverfügbar macht.

13. März 2020

Die Corona-Krise erreicht die DOAG. Alle Mitarbeiter der DOAG-Geschäftsstelle arbeiten seit heute im Home-Office und setzen dabei auf bewährte Tools wie Redmine oder GotoMeeting, um alle Aktivitäten neu zu definieren und zu koordinieren.

16. März 2020

Shutdown der Geschäftsstelle. Die Corona-Krise zwingt die DOAG zu drastischen Maßnahmen. Alle geplanten Veranstaltungen müssen abgesagt werden. Alle Mitarbeiter der Geschäftsstelle müssen in Kurzarbeit. Die Geschäftsstelle konzentriert sich für den Rest des Jahres auf die Printmedien und Online Services, die verstärkt werden sollen..

17. bis 19. März 2020

Die JavaLand 2020 ist abgesagt. Zum siebten Mal sollte sie in dieser Woche mit über 2.200 Entwicklern und Java-Enthusiasten ihre Tore im Phantasialand Brühl öffnen. Die Vorbereitungen für das große Wiedersehen der Community mit zahlreichen Vorträgen von Top-Referenten, interaktiven Community-Aktivitäten und jeder Menge Spaß sind getroffen.

Schon in der Vorbereitungszeit wurden täglich die Risiken des neuen Coronavirus (SARS-CoV-2) bewertet. Erforderliche Maßnahmen sind in enger Abstimmung mit der Konferenzleitung – und damit der Java-Community – in das Veranstaltungskonzept eingeflossen. Die schnelle Entwicklung der Epidemie zur Pandemie hat aber vor der Javaland nicht haltgemacht.

Angesichts der Entwicklung sehen wir uns als Veranstalter, im Interesse der Gesundheit der Teilnehmer, schweren Herzens gezwungen die JavaLand 2020 abzusagen. Der Erlass der Landesregierung NRW, zunächst Veranstaltungen mit mehr als 1000 Teilnehmenden, dann auch Veranstaltungen mit weniger als 1000 Teilnehmern behördlich zu untersagen, bestätigt uns in der Entscheidung.

17. März 2020

Die als kleiner Ersatz für die JavaLand kurzfristig ins Leben gerufene Cyberland 2020 findet als Online-Event rund um aktuelle Java-Themen statt. Innerhalb von 24 Stunden haben sich 1000 Teilnehmer hierfür registriert. JavaLand unterstützt dieses Event, das von Mitgliedern einiger deutscher JUGs und der CyberJUG organisiert wird. Fried Saacke, Konferenzleiter der JavaLand und Vorsitzender des iJUG begrüßt pünktlich um 9:00 Uhr die Teilnehmer des kostenlosen Events.

18. März 2020

Die JUG Darmstadt hat zu ihrem ersten virtuellen Treffen drei JavaLand-Referenten aus der Region eingeladen. Die Teilnahme ist frei und für alle offen.

19. März 2020

Das CloudLand Pre-Event 2020 im Phantasialand wird aufgrund der Corona-Pandemie und der damit verbundenen Schließung des Phantasialand ebenfalls abgesagt und findet nicht statt. Die Veranstaltung sollte sich den großen Themen Cloud und Digitalisierung widmen.

6. April bis 19. April 2020

Vom 6. April bis einschließlich 19. April bleibt die DOAG-Geschäftsstelle wegen der Corona-Krise aufgrund einer Betriebschließung über Ostern geschlossen. In der Zeit ist für dringende Angelegenheiten eine Notbesetzung durch die Geschäftsleitung gewährleistet.

16. bis 18. April 2020

Die DOAG 2020 Delegiertenversammlung am 18. April 2020 einschließlich des Leitungskräfteforums vom 16. bis 17. April in Berlin findet aufgrund der Corona-Krise nicht statt.

18. April 2020

Der Vorstand der DOAG hat alle Delegierten zu einer Virtuellen Delegiertenversammlung eingeladen. Diese ist formal nicht beschlussfähig. Dem Vorstand war es aber wichtig, alle Delegierten zur Situation und den Maßnahmen zu informieren und sich mit den Delegierten ein Meinungsbild zu den eingeleiteten Maßnahmen, mit denen die DOAG die Krise überwinden möchte, einzuholen.

Aus der Ferne betrachtet: Cloud-Computing ist ein Bombengeschäft



Günther Stürner

Es gibt keine Zweifel mehr: Cloud-Computing ist ein Bombengeschäft.

Ein Geschäft, das nicht etwa die etablierten IT-Giganten wie IBM, HP oder hierzulande T-Systems dominieren, die ja jahrzehntelang für viele Kunden deren IT-Systeme betrieben haben und die das Metier aus dem „Effeff“ kennen müssten. Nein, IBM und Co. spielen heute in diesem Geschäft lediglich eine Nebenrolle. Die Hauptrollen sind vergeben an Amazon, Microsoft und Google. Das zumindest ist der aktuelle Stand, wenn man die Cloud-Umsätze als Maßstab heranzieht. Auch chinesische Anbieter wie Alibaba sind groß im Kommen. Es ist anzunehmen, dass sich dadurch noch einiges zurechtschütteln wird in den nächsten wenigen Quartalen. Doch der \$230-Milliarden-Markt ist nicht nur den ganz Großen vorbehalten. Nahezu jeder spielt mit, der heute im IT-Umfeld aktiv ist. Jeder Anbieter hat eine Cloud-Strategie. Ohne Cloud, meint man, geht gar nichts mehr.

Das Spielfeld ist groß, wird immer größer und ist enorm lukrativ. Und ich dachte immer, Software wäre das lukrativste Geschäft.

Was die einzelnen Firmen anbieten, ist dabei höchst unterschiedlich und in hohem Grade vielfältig bis unübersichtlich. Allein bei Amazon finden sich Hunderte von Services von Analytik über Quantentechnologie bis hin zu klassischen Speicherangeboten.

So richtig vergleichbar ist das Ganze nicht. Für Kunden ist das große Angebot Segen und Fluch zugleich. Irgendein Cloud-Anbieter hat sicher den Service, den ich suche (für kleines Geld) bei der Hand und ein Prototyp ist schnell gebaut. Da auch in der IT, wie überall, nichts ist zählebiger als ein Provisorium, hat man schnell einen Sack von Systemen, basierend auf unterschiedlichen Sprachen und betrieben in unterschiedlichen Cloud-Umgebungen. Halleluja!

Auf einem völlig anderen Blatt stehen die großen Unternehmenssysteme, die man nicht an einem Tag aufbaut und wieder fallen lässt.

Hier sieht Oracle seine große Chance und das ist sicher auch richtig so. Das, was jetzt als Generation-2-Cloud vorliegt, hat im Gegensatz zu den ersten Gehversuchen eine echte Chance am Markt.

Oracle bietet Oracle-Produkte als Cloud-Services auf konkurrenzfähiger Basis und mit einer konkurrenzfähigen Preisstruktur an. Die Autonome-Datenbank-Technologie ist richtig gut und cool! Jetzt lohnt es sich, einzusteigen und die Services auch zu benutzen, und ich bin mir sicher, dass Oracle mit seinen Cloud-Angeboten ein fundiertes und erfolgreiches Geschäftsmodell etablieren kann.

Die eigenen Services in der eigenen Cloud zu betreiben ist gut, aber noch nicht ausreichend.

Wichtig erscheint mir auch, dass sich Oracle an seine Stärken erinnert. Das Erfolgs-Motto lautete: Eine Software für alle Plattformen. Damit ist Oracle groß geworden. Leicht modifiziert und der heutigen Zeit angepasst sollte das Motto lauten: eine Software (z.B. Datenbank) als Basis für alle Services, auf allen Cloud-Plattformen und auf allen wichtigen HW-/OS-Plattformen, virtualisiert oder nicht, völlig egal.

Dass hier noch einiges im Argen liegt, ist offensichtlich. Wichtige IT-Bereiche werden von Oracle mit einer Hartnäckigkeit, die an Halsstarrigkeit grenzt, ignoriert. Es sei nur an das leidige Thema Lizenzierung in virtuellen Umgebungen erinnert oder das Ignorieren von Marktgegebenheiten wie beispielsweise der aktuellen Dominanz von Amazon. Einzig mit Microsoft hat sich Oracle ein wenig geschmeidiger gezeigt. Diese Strategie zahlt sich mittelfristig aus und wäre auch für andere Partnerschaften angebracht. AWS als Basis für alle Oracle-Software-Produkte? Ja, warum denn nicht! GCP als Basis für alle Oracle-Produkte? Ja, bitte, ist doch egal, dass Thomas Kurian dort jetzt das Sagen hat! Alle Oracle-Produkte auf Private Clouds, die zu 100% virtualisiert sind? Aber klar doch – und das mit einem modernen und fairen Lizenzmodell!

Manchmal denke ich, es wäre an der Zeit, dass Oracle auch seinen „Satay Nadella“ bekommt. Einen, der die Handbremsen löst. Einen, der ein anderes, frisches, nicht-dogmatisches Denken in die Organisation bringt. Ich meine, Oracle hätte es verdient und vor allem die Oracle-Kunden hätten es verdient.

Kommentare, Anregungen, Lob – und wenn es sein muss, auch Tadel – an: guenther.stuerner@dbms-publishing.de



„Technologisch werden Unternehmen stets IT-Experten brauchen, die Cloud-Technologie verstehen und betreiben können, selbst wenn weite Teile der Planung, Migration und des Betriebs an einen Cloud-Partner ausgelagert werden.“

Martin Meyer, Redaktionsleiter des Red Stack Magazin, sprach mit Holger Nicolay, Business Development Manager bei Interxion, über die Vorzüge, die Akzeptanz und die Risiken von Cloud Computing.

Was versteht man allgemein unter einer Public Cloud und wer sind die Anbieter?

Public-Cloud-Services sind durch einen Dritten zentral erbrachte IT-Dienste, auf die Unternehmen oder Privatkunden remote und spontan zugreifen können. Diese Services skalieren und werden strikt nach Nutzung abgerechnet. Die großen Player sind der Marktführer Amazon Web Services, Microsoft und Google – aber natürlich auch Oracle mit seiner Oracle Cloud Infrastructure.

Unternehmen nutzen zunehmend die Public Cloud. Was sind die Vorzüge einer Public Cloud? Was die Nachteile?

Bei den Vorzügen einer Public Cloud ist vor allem zu nennen, dass jeder Kunde unangemeldet und spontan auf Ressourcen zugreifen kann. Das ist insbesondere für Nutzer von Vorteil, die im Development- oder Test-Umfeld tätig sind, und auch für solche, die zur Auswertung von Messreihen oder monatlich anfallenden Abrechnungsdaten lediglich sporadisch und zeitweise Ressourcen benötigen. Gerade bei Szenarien mit zeitweiser Nutzung ist es sinnvoll, nicht dauerhaft in eine eigene Infrastruktur investieren zu müssen.

Der Nachteil liegt darin, dass Cloud-Kunden sich auf einer mit anderen Kunden geteilten Infrastruktur bewegen und damit nicht – wie traditionell – die volle Zugriffskontrolle auf die genutzten Kapazitäten haben. Vielmehr sind sie bei der Public-Cloud-Nutzung von der Infrastruktur eines Dritten und dessen Vorhaltung von geeigneten Ressourcen zur Bedienung aller Kunden-Workloads abhängig.



Zur Person: Holger Nicolay

Holger Nicolay ist Business Development Manager bei Interxion, einem führenden europäischen Colocation-Provider mit mehr als 50 Carrier- und Cloud-neutralen Data Centern. Sein Fokus sind die digitale Transformation sowie Connectivity- und Infrastruktur-Konzepte, mit denen Interxion seine Kunden als Infrastrukturpartner und Cloud Connect Provider bei der Implementierung von hochintegrierten hybriden Clouds und flexiblen Multi-Cloud-Lösungen unterstützt. Holger Nicolay ist Diplom-Informatiker (TU) und hat vor Interxion in verschiedenen Sales-, Marketing- und Management-Rollen in der Telekommunikationsbranche gearbeitet. Frühere berufliche Stationen waren die heutigen 1&1 Versatel und Colt Technology Services.

Wie sieht die Akzeptanz der Cloud in Deutschland aus?

Die Anbieter vermelden beachtliche Zuwachsraten. Auch die Interxion-Marktumfrage, die alljährlich im Frühjahr Unternehmenskunden nach den fokussierten Trends und Themen befragt, besagt seit Jahren, dass Cloud Computing in den deutschen Unternehmen angekommen ist.

Dennoch gibt es nach wie vor Bedenken in Bezug auf die Sicherheit der Daten. Hier muss sich jeder Nutzer selbst Gedanken machen, wie er seine Geschäftsgeheimnisse und besonders Schutzwürdiges wie personenbezogene Daten sicher in der Cloud verwaltet.

Ist eine Public Cloud günstiger als eine eigene Lösung?

Das Wesen der Public Cloud ist, dass Ressourcen genutzt werden, wenn sie gerade gebraucht werden. Das heißt aber auch, dass Kapazitäten reduziert oder abgeschaltet werden sollten, wenn gerade weniger oder kein Bedarf nach ihnen besteht. Dies hat zur Folge, dass Nutzer Applikationen anders betreiben müssen als das bislang der Fall war.

Wer seine Applikationen nur mittels Lift & Shift in die Cloud verschiebt, wird kaum solche Kostenvorteile erlangen wie derjenige, der auf das Cloud-spezifische Betriebs- und Abrechnungsmodell hin seine Applikationen gänzlich neu entwickelt oder vor der Migration modifiziert. Automatisierungsmechanismen und Autoskalierung sind die Hebel, mit denen sich in der Public Cloud betriebene Applikationen bedarfsgerecht anpassen und damit kosteneffizient betreiben lassen.

Wie stehen Sie zu den in Deutschland immer wieder geäußerten Sicherheitsbedenken, wenn es um Daten in der Cloud geht?

Daten in der Cloud sind per se sicherer als im unternehmens-eigenen Rechenzentrum, weil Kunden hier auf von Spezialisten bereit gestellte Cloud-Services zurückgreifen, die in professionell gemanagten und vielfach zertifizierten Rechenzentren – wie diejenigen von Interxion – betrieben werden.

Bei der Datensicherheit liegt also ein ausgesprochen hohes Schutzniveau vor, insbesondere wenn ein Kunde geeignete Redundanzkonzepte und Backup-Maßnahmen beauftragt beziehungsweise implementiert. Auch beim Datenschutz existiert diese geteilte Verantwortlichkeit zwischen Kunde und Cloud-Anbieter, gemeinsam geeignete Maßnahmen in Bezug auf Datensicherheit und Datenschutz, Prozesse und Compliance, Zugriffsrechte und Verschlüsselung zu ergreifen.

Aus unserer Sicht geht es bei der Frage der Datensicherheit aber nicht nur um die ruhenden Daten in der Public Cloud, sondern auch um den Zugriffsweg des Nutzers dort. Müssen oder dürfen überhaupt alle Daten über das öffentliche Internet geschickt werden oder möchte man hier nicht ebenfalls auf ein geschütztes, privates Netzwerk mithilfe des „OCI Fast Connect“-Services zurückgreifen?

Wenn Cloud, warum die Oracle Cloud Infrastructure wählen?

Pauschal ist kaum ein seriöser Ratschlag für oder wider einen bestimmten Anbieter zu geben, der für alle Branchen, alle Kunden und alle Anwendungsfälle gleichermaßen gelten kann.

Im Grundsatz geht es bei dieser Fragestellung sowohl um kommerzielle als auch um technologische Rahmenbedingungen. Was uns in erster Linie berichtet wird, ist, dass Oracle-

Kunden bei der Nutzung der Oracle-Cloud massive kommerzielle Vorteile in Bezug auf die Lizenzkosten erleben – verglichen mit dem Konzept, Oracle-Applikationen in einer Amazon-Cloud oder auf Microsoft Azure zu betreiben.

Großen Ausschlag für oder gegen ein bestimmtes Konzept oder einen Anbieter gibt unserer Erfahrung nach zudem das Zusammenspiel der Komponenten untereinander. In hybriden Szenarien kommunizieren Cloud-Services mit Applikationen oder Appliances in der Colocation eines Data-Center-Providers oder gar im weit entfernten, unternehmenseigenen Rechenzentrum. Ein mögliches Zusammenspiel von Cloud-Applikationen untereinander stellt die Cloud-basierte Oracle-Datenbank mit Microsoft „Office 365“ dar.

Welche Alternativen sehen Sie bei Cloud-Lösungen?

Prinzipiell sehen wir eine Vielzahl an Cloud-Angeboten am Markt. Da gibt es Cloud-Technologien für das eigene Rechenzentrum, wie die „Oracle Cloud Machine“. Es gibt mehr oder weniger regional oder branchen-spezifisch aufgestellte Anbieter, die mit Kundennähe und Branchenkenntnis punkten. Es muss also nicht immer der weltweit agierende Cloud-Konzern sein, das Spektrum ist vielfältig.

Am Ende des Tages verschwimmt aber auch die Trennlinie zwischen Hosting und Cloud, Private Cloud und Managed Services. Welche Lösung für welches Unternehmen am besten passt, sondiert man am besten mit den spezifischen Anbietern. Spannend hierzu auch die Ergebnisse der Interxion-Marktstudie von 2018: Die Anbietersauswahl wird auf Applikationsebene getroffen, weil es für zahlreiche Anwendungen bereits sehr ausgereifte „Software-as-a-Service“-Lösungen gibt.

Woraus besteht ein typischer Cloud-Service? / Welche Arten von Cloud-Services gibt es?

Ich bin mir gar nicht sicher, ob es den typischen Cloud-Service überhaupt gibt, zu groß und damit zugegebenermaßen auch unübersichtlich ist das Angebot hier mittlerweile geworden.

Grundsätzlich gibt es für die verschiedenen Kundenszenarien vielfältige Angebote. Vereinfacht gesprochen bilden die „Infrastructure-as-a-Service“-Angebote mit ihren virtuellen Maschinen den eigenen Server nach, während der Kunde bei „Platform-as-a-Service“-Angeboten die Laufzeitumgebung mitgeliefert bekommt und sich somit auch nicht mehr um Dinge wie die Betriebssystem-Ebene kümmern muss. „Software-as-a-Service“-Applikationen liefern sozusagen „Software aus der Steckdose“.

Hinzu kommen zahlreiche Abstufungen und Varianten wie beispielsweise „DataBase-as-a-Service“ und „Security-as-a-Service“. Richtig spannend wird es für den Cloud-Kunden, wenn er sich tiefer in das Angebot der Public-Cloud-Hyperscaler einarbeitet: Hier findet er dann nicht selten Hilfestellungen zu Low-Code-Implementierungen, Baukästen für IoT- und Analytics-Anwendungen und für die Integration mobiler User. Diese Add-on-Services bilden die eigentliche Stärke und das Differenzierungsmerkmal der globalen Public-Cloud-Provider: Ihre Nutzer müssen für eine Vielzahl von Problemstellungen und Anwendungsszenarien das Rad nicht mehr neu erfinden, sondern können in Modulbauweise auf Fertiges und Bewährtes aufsetzen.

Wie unterscheidet sich Cloud Computing von SaaS?

Cloud Computing im Sinne einer „Infrastructure-as-a-Service“-Umgebung ermöglicht seinen Nutzern, wie im eigenen Rechenzentrum Server zu betreiben, ohne in diese investieren zu müssen. Man konfiguriert sich seine virtuelle Maschine mit all ihren Parametern wie Speicherplatz per Mausklick und verbindet diese durch ein virtuelles Cloud-Netzwerk entweder mit der realen Welt außerhalb der Cloud oder mit anderen Cloud-Services.

Das Grundprinzip von „Infrastructure-as-a-Service“ ist, dass der Nutzer seine Server und sein Netzwerk ähnlich einem realen Rechenzentrum zusammenbaut. Ehe also eine Applikation für den Nutzer läuft, muss ein Administrator die Infrastrukturumgebung konfiguriert und gestartet haben.

„Software-as-a-Service“-Applikationen hingegen liefern eine fertige Software aus der Steckdose. Die Einwirkungsmöglichkeiten für den Anwender sind gering, dieselbe Art Software wird für Tausende von Anwendern in ihrer Funktionalität immer gleich zur Verfügung gestellt. Das Praktische hierbei ist, dass sich Administratoren um Dinge wie lokale Installation und Updates nicht mehr kümmern müssen, die Software steht dem Anwender umgehend, ähnlich einem über Webbrowser zugänglich gemachten Internetdienst, zur Verfügung.

Wie sicher sind Cloud-Services hinsichtlich Datenschutz und Verfügbarkeit?

Prinzipiell unterliegen Public-Cloud-Provider mit ihren deutschen Rechenzentren dem deutschen und dem europäischen Datenschutzrecht, gleichzeitig sind US-Anbieter an US-Recht, zum Beispiel den Cloud-Act gebunden. Hier besteht ein Spannungsverhältnis und juristisch gesehen Neuland.

Grundsätzlich nie verkehrt ist die Verschlüsselung der Daten in der Cloud bei gleichzeitiger Aufbewahrung der Schlüssel außerhalb der Public-Cloud-Umgebung. Hier gibt es dedizierte Hardware-Lösungen für das kundeneigene Rechenzentrum am Markt, aber auch HSM-as-a-Service-Angebote, also Hardware-Security-Module wie Interxions „Key Guardian“-Service.

Die Verfügbarkeit von Public Cloud Services ist grundsätzlich sehr hoch – gerade hier lohnt sich beim Anbietervergleich aber der Blick ins Kleingedruckte. Oracle beispielsweise betreibt seine „Oracle Cloud Infrastructure“ in drei Rechenzentren bei drei verschiedenen Colocation-Providern quer über das Frankfurter Stadtgebiet verteilt. Wie der Kunde von dieser Mehrfach-Redundanz profitiert, sollte bei der Auswahl und der Konfiguration der Cloud-Services beachtet werden.

Wie finde ich den passenden Cloud-Partner?

Das ist eine schwierige Frage in einem Markt, der stark in Bewegung ist und von einer Vielzahl an Angeboten gekennzeichnet sind, die alle ihre Daseinsberechtigung haben. Insgesamt gilt, dass viele Cloud-Partner über Spezialisierungen in bestimmten Branchen oder Technologien verfügen. Wenn das alles zu den eigenen Anforderungen passt, lohnt ein Blick auf die Referenzkunden sowie die Art und Anzahl der erfolgreichen Cloud-Projekte, schließlich will man sich als Kunde nicht nur theoretische Expertise, sondern auch tatsächliche Erfahrung zukaufen.

Welche Prozesse im Unternehmen verändern sich durch den Einsatz einer Cloud und wie muss ich mich organisatorisch vorbereiten?

Hier gibt es mindestens drei Dimensionen zu beachten: die technologische, kommerzielle und die organisatorische.

Technologisch werden Unternehmen stets IT-Experten brauchen, die Cloud-Technologie verstehen und betreiben können, selbst wenn weite Teile der Planung, Migration und des Betriebs an einen Cloud-Partner ausgelagert werden.

In der kommerziellen Dimension sollten Verträge, beispielsweise für die Zusammenarbeit mit dem Cloud-Partner, flexibel gehalten sein, um mit der Veränderungsdynamik bei den Cloud-Angeboten Schritt halten zu können. Cloud-Nutzung wird zudem verbrauchsabhängig abgerechnet und ist damit schwer vorhersehbar.

Das leitet zu den organisatorischen Herausforderungen über, denn Verbrauchsabhängigkeit in einem dynamischen Angebotsfeld passt schwerlich in traditionelle Budgetplanungsprozesse. Zudem wird vermehrt Provider-Management-Expertise und nicht mehr nur IT-Administratoren-Erfahrung benötigt. Insgesamt sind Einkaufs- und Freigabeprozesse zu überdenken, da kostenpflichtige Ressourcen mittels Mausclick über das Portal des Cloud-Providers oder sogar vollständig autoskalierend und damit spontan und flexibel gebucht werden.

Welche Migrationsstrategie wähle ich?

Für die Migrationsstrategie bieten sich mehrere Varianten an, die jeweils von mehreren Parametern abhängen. Idealerweise vermeiden Unternehmen den Lift-&-Shift-Ansatz, wenn sie optimale Ergebnisse sowohl kommerzieller als auch technologischer Natur von ihrer Cloud-Nutzung erwarten. Dennoch mag ein solcher Ansatz genau dann sinnvoll sein, wenn beispielsweise die Aufgabe des eigenen Rechenzentrums oder auslaufende Wartungsverträge Zeitdruck auf den Migrationsplan ausüben.

In derart zeitkritischen oder sehr großen Migrationsprojekten kann es durchaus sinnvoll sein, erst per Lift & Shift die Cloud-Migration vorzunehmen, um zu einem späteren Zeitpunkt Applikations-spezifische Optimierungen durchzuführen.

Für optimale Performance und Kosten-Nutzen-Ratio innerhalb der Public Cloud sorgt jedoch die umgekehrte Herangehensweise: Applikationen werden zuerst auf die Cloud-Nutzung hin optimiert und dann migriert. Dies ist die deutlich aufwendigere Herangehensweise, gerade wenn eine Vielzahl an Bestandsapplikationen involviert sind.

Üblich sind daher auch Migrationsszenarien, die nicht die Verlagerung der gesamten Bestands-IT adressieren, sondern die Optimierung einzelner Applikationen fokussieren. Infrage kommen dabei solche IT-Services, für die sowieso eine Neuentwicklung oder ein Generationswechsel ansteht.

Wie einfach lässt sich Cloud-Monitoring implementieren, konfigurieren und warten?

Wer sein Monitoring mithilfe der vom Cloud-Provider bereitgestellten Tools, Schnittstellen und APIs nicht selbst aufbauen möchte, findet für diese Fragestellungen zahlreiche Anbieter am Markt, die ihre Lösungen wiederum als Cloud-Service anbieten. Solche Plattformen können zum Teil die Clouds verschiedener Provider monitorieren – auch hybride Szenarien Ende-zu-Ende abbilden zu können ist hilfreich.

Wie kann ich die Cloud-Lösungen, die bei verschiedenen Anbietern gehostet werden, verbinden?

Auf Applikationsebene lassen sich solche Fragestellungen von Interoperabilität mit APIs (Application Programming Interfaces) und Konnektoren lösen. Eine zusätzliche Herausforderung liegt im Netzwerkbereich auf der Infrastrukturebene. Beim Austausch großer Datenmengen unter zeitkritischen Rahmenbedingungen bieten sich private Netzwerkanbindungen an, die zudem für Betriebssicherheit und Datenschutz auf der Netzwerkebene sorgen. Oracle bietet einen solchen Service mit seinem „OCI Fast Connect“ an – auch Amazon, Microsoft oder Google haben Pendant im Portfolio.

Zu beachten ist, dass diese (virtuellen) Ports am Netzwerkrand der Cloud-Plattform enden, also ein Cloud-Partner in Form eines Colocation-Providers oder eines Telekommunikationsanbieters benötigt wird, um die Ende-zu-Ende-Verbindung in das Unternehmensrechenzentrum zu verlängern.

Im Falle von Multi-Cloud-Szenarien bieten sich insbesondere die Cloud-Connect-Plattformen derjenigen Data-Center-Provider an, in deren Rechenzentren die Public-Cloud-Provider ihre Netzknoten betreiben. So können Cloud-zu-Cloud-Netzwerke lokal zusammengeschaltet werden.

Aber Achtung: Außer Oracle berechnen viele der Public-Cloud-Anbieter das ausgehende Datenvolumen. Neben dem technischen Design gilt es also, auch die kommerziellen Auswirkungen einer solchen Implementierung im Auge zu behalten.

Wie sieht die Zukunft des Cloud-Computings in Deutschland aus?

Die Zukunft des Cloud-Computings liegt sicherlich in Partnerschaften. Unsere Kunden signalisieren uns, dass räumliche Nähe zur Reduzierung von Zugriffszeiten und Netzwerkkosten für sie von ganz entscheidender Bedeutung ist und sie diesen mit Colocation-Services herstellen können. Dies ist aber auch der Grund dafür, dass die Public-Cloud-Angebote in immer mehr Rechenzentren immer näher an die Nutzer ausgerollt werden, wie Oracle das auch auf der DOAG-Konferenz im November angekündigt hat.

Richtig engmaschige Angebote weltweit wird ein Anbieter allein aber nicht schaffen können, daher sind Partnerschaften zwischen der Oracle Cloud Infrastructure und Microsoft Azure – auch für Europa – bereits angekündigt. Derart partnerschaftliches Vorgehen kommt letztendlich allen Nutzern zugute, den Unternehmenskunden wie auch den Oracle-Partnern. Deren Expertise bei der Begleitung von Anwendern in die Cloud wird zukünftig mehr denn je gefragt sein.

Trotz aller zu erwartenden räumlichen Nähe der Cloud-Plattformen zu den Anwendern gehen wir davon aus, dass die Zukunft der Cloud hybrid sein wird. Kosten- und Performance-Vorteile lassen sich dadurch heben, dass Daten dort verarbeitet werden, wo sie am besten aufgehoben sind. Hinzu kommen Regulatorisches oder Vertragsgestaltungen, die einen bestimmten Aufbewahrungsort der Daten und damit der Applikationen vorsehen.

Es lohnt sich also, die Weiterentwicklungen und Trends sowohl für die Public Cloud, als auch in Bezug auf die Cloud-Appliances für das eigene Rechenzentrum parallel im Auge zu behalten.



Fallen Sie nicht in die Cloud

Michael Skowasch, ORDIX

Das Thema Cloud ist ein äußerst komplexes Thema. Viele Entscheider und Administratoren überlegen, ob ihre Oracle-Datenbanken in einer On-Premises-Umgebung (in den eigenen Räumlichkeiten) oder in einer Public-Cloud-Umgebung (externe Anbieter) besser aufgehoben sind.

Oracle wirbt mit dem Cloud-only-Merkmal „Autonomous Database“, das es nur in der „Oracle Cloud Infrastructure (OCI)“ gibt. Ebenso bietet Oracle dort TDE (Transparent Data Encryption) sogar für die Standard Edition 2 (SE2) an. Dies ist eine Datenverschlüsselung auf Blockebene im Datenfile. Auf der anderen Seite fallen Features in Versionen weg, die gerne

von Kunden in der On-Premises-Variante genutzt werden. So kann der Real Application Cluster (RAC) ab 19c nicht mehr für die SE2 eingesetzt werden. Oracle-Releases erscheinen früher in der Cloud als auf On-Premises (Beispiel: Release 18c erschien im März 2018 in der Cloud und auf On-Premises erst im Juli 2018).

Einige Unternehmen wollen bestimm-

te Oracle-Datenbanken, wie zum Beispiel eine Einwohnermeldedatenbank, aufgrund der Personendaten auf gar keinen Fall in die Public Cloud migrieren. Andere Unternehmen entscheiden wiederum „alles“ in die Cloud zu geben.

Viele Institute, darunter Gartner [1], sehen zweistellige Wachstumsraten in den Jahren 2019 und 2020 im Public Cloud

Service. Aber sind hier wirklich auch die Oracle-Datenbanken gemeint oder doch eher die Nutzung von anderen Cloud Services, wie beispielsweise Mailservices?

Die Oracle-Cloud-Zahlen werden auch dadurch beeinflusst, dass man mittlerweile bei Oracle oft nur noch rabattierte On-Premises-Lizenzen erhält, wenn auch gleichzeitig Cloud-Lizenzen erworben werden. Viele dieser Cloud-Lizenzen werden jedoch nicht genutzt.

Auch hört man immer wieder Pressemeldungen: „Cloud-Migrationen langwieriger und teurer als gedacht“ [2] oder „Firmen unterschätzen die Cloud-Migration“ [3].

In diesem Artikel betrachte ich die Oracle-Datenbanken und keine anderen Cloud-Services, die man in der Public Cloud nutzen könnte. Ebenfalls werde ich versuchen, die folgende Frage zu beantworten: Wann ist es sinnvoll, mit Oracle-Datenbanken in die Cloud zu gehen? Dabei werde ich die Probleme und mögliche Fallstricke aufzeigen.

Beginnen möchte ich mit einigen Grundlagen: Cloud-Anbieter, -Formen, -Modelle und die Lizenzmodelle in der Public Cloud. Alle aufgelisteten Preise beziehen sich auf den Stand 09/2019.

Cloud-Anbieter

Für den Betrieb beziehungsweise die Nutzung von Oracle-Datenbanken gibt es eine Reihe von Public-Cloud-Anbietern:



- Oracle Cloud Infrastructure (OCI)



- Amazon Web Services (AWS)



- Open Telekom Cloud (OTC)



- Microsoft Azure

Im Artikel werde ich hauptsächlich auf die Oracle Cloud Infrastructure (OCI) und die Amazon-Web-Services-Cloud (AWS) ein-

gehen und die Vor- und Nachteile dieser beiden Anbieter aufzeigen.

Cloud-Formen

Eine Private Cloud kann man sich selbst im Unternehmen beispielsweise mit Virtualisierungs-, RAC- beziehungsweise Exadata-Lösungen schaffen. Unter einer Public Cloud versteht man die Verwendung der Ressourcen von externen Anbietern.

Die Private und die Public Cloud kann man auch mischen; dann spricht man von der Hybrid Cloud. In der Public Cloud könnten das Backup der Datenbanken oder auch Standby-Datenbanken liegen. Die produktiven Datenbanken bleiben beim Kunden. Ebenfalls kann die Public Cloud nur für die Test- und Entwicklungsdatenbanken verwendet werden.

Von einer Multi Cloud spricht man bei der Nutzung unterschiedlicher Public-Cloud-Anbieter.

Cloud-Modelle

In der Public Cloud werden verschiedene Modelle angeboten.

- **IaaS: Infrastructure as a Service**
Hier können eigene physikalische (Bare Metal bei der OCI) oder virtuelle Server beim Cloud-Anbieter exklusiv genutzt werden.
- **PaaS: Platform as a Service**
Bei diesem Modell wird nur ein Service, zum Beispiel eine Datenbank auf physikalischen (Bare Metal) oder virtuellen Servern zur Verfügung gestellt.
- **SaaS: Software as a Service**
Mit SaaS werden bestimmte Programme vom Cloud-Anbieter angeboten, wie beispielsweise Mailprogramme.

Lizenzmodelle

In der Public Cloud gibt es verschiedene Lizenzmodelle. Es kann nur die tatsächliche Nutzung, zum Beispiel Pay-as-you-go oder eine feste Abnahmemenge, wie etwa mit Universal Credits – Monthly Flex gekauft werden.

Entweder sind die Oracle-Lizenzierungskosten im Preis bereits eingerechnet oder es können eigene Lizenzen mitgebracht

werden (BYOL (Bring your own licenses)). Dies ist für IaaS und PaaS möglich.

Oracle Cloud Infrastructure (OCI) – Angebot

Wie auf der Open World 2019 mitgeteilt wurde, ist Oracle mit Microsoft Azure eine Kooperation eingegangen. Beide Cloud-Anbieter teilen sich demnächst Rechenzentrums-Ressourcen. Oracle will bis Ende 2020 bis zu 36 Regionen weltweit für die Public Cloud anbieten.

In der OCI sind alle Cloud-Modelle IaaS, PaaS und SaaS verfügbar.

IaaS / PaaS

- **Bare Metal Cloud**
Hier können dedizierte Server pro Kunde gebucht werden
- **Virtuelle Server**
Keine dedizierten Server
- **Regionen in verschiedenen geografischen Lokationen**

SaaS

- Hier können unterschiedliche Software-Produkte in der Public Cloud verwendet werden


Oracle offeriert für unterschiedliche Anforderungen verschiedene Editionen und Pakete: Standard Edition 2, Enterprise Edition (EE), EE High Performance und Extreme Performance. In der *Abbildung 1* sind die inkludierten Features aufgelistet.

In allen Editionen und Packages ist Transparent Data Encryption (TDE) enthalten. In der Enterprise Edition (EE) sind alle oben aufgelisteten Features dabei, wenn PaaS oder IaaS inklusive der Lizenzen gekauft werden.

Im Enterprise Package (Name bei BYOL) müssen die Lizenzen für die Enterprise Edition und bei einer Verwendung der Features auch die aufgelisteten Features separat lizenziert werden.

Mit dem Tarif Pay-as-you-go können die Ressourcen ohne vertragliche Verpflichtung und Vorauszahlungen in der Public Cloud angemietet werden. Die nutzungsbezogene Bezahlung erfolgt per monatlicher Rechnung. Hier zahlt man den Listenpreis, aber man ist maximal flexibel.

Der Tarif Monthly Flex bietet deutlich günstigere Preise, allerdings ist hier eine



| Standard Edition 2 (SE2) | Enterprise Edition (EE) | EE High Performance | EE Extreme Performance |
|---|--|---|---|
| <ul style="list-style-type: none"> SE2 Datenbank Max. 16 OCPUs Transparent Data Encryption (TDE) | <ul style="list-style-type: none"> EE Datenbank mit allen EE Features Hybrid Columnar Compression Data Masking und Subsetting Pack Diagnostic und Tuning Real Application Testing | <ul style="list-style-type: none"> Multitenant Partitioning Advanced Compression Advanced Security, Label Security, Database Vault OLAP, Analytics, Spatial und Graph Database Lifecycle und Cloud Management Packs | <ul style="list-style-type: none"> RAC, RAC One Node In Memory Active Data Guard |

Abbildung 1: Oracle Cloud Editions / Pakete (Quelle: © Oracle)

Mindestabnahme pro Monat von 1.000 Dollar und eine Bindung von mindestens einem Jahr erforderlich.

Als Ressourcen können sogenannte OCPU (One Database Oracle Compute Unit) ausgewählt werden. Dabei entspricht eine OCPU einem physikalischen CORE eines Intel Xeon-Prozessors.

In der *Tabelle 1* zeige ich einen Preisvergleich von Pay-as-you-go und Universal Credits (Monthly Flex) mit und ohne BYOL. Es soll ein virtueller Server mit Oracle (PaaS) in der Enterprise Edition mit 16 Oracle OCPUs verwendet werden.

Sollen Ressourcen in der Cloud für die Enterprise Edition auf virtuellen Servern angemietet werden, bietet der Monthly-Flex-Tarif inklusive Lizenzen das beste Preis-Leistungs-Verhältnis. Werden eigene Lizenzen mitgebracht (BYOL) müssen auch zusätzlich die verwendeten Packs und Features mit eingebracht werden. Für die Option BYOL müssen jedoch weiterhin die Wartungskosten der mitgebrachten Lizenzen bezahlt werden.

Hinzu kommen noch Storage-, Backup-, Netzwerk- und Datenübertragungskosten. Dabei können bis zu 10 TB Daten ausgehend kostenfrei pro Monat genutzt werden. Dies ist im Gegensatz zu anderen Cloud-Anbietern vergleichsweise großzügig. Trotzdem wird es richtig schwierig und aufwendig, die Preise für die Nutzung einer Datenbank in der Public Cloud zu ermitteln.

Versuchen Sie es selbst über den Oracle-Kostenrechner: <https://www.oracle.com/de/database/vm-cloud-pricing.html>

Mittlerweile bietet Oracle einen Always-Free-Cloud-Service an. Hier können Sie über einen Account ein Leben lang 2 Datenbanken auf jeweils einer virtuellen Maschine mit jeweils 1 OCPU und 20 GB Speicher testen: <https://www.oracle.com/de/cloud/free/#always-free>

Oracle Cloud Infrastructure (OCI) – Sicherheit

Es ist eine Vertrauenssache, seine Daten in eine Public Cloud zu legen. Auf der

DOAG Konferenz + Ausstellung 2019 hat Oracle mehr Transparenz für die Sicherheit der Daten in der OCI geliefert. Die Region Frankfurt wird von Administratoren aus den USA und Irland administriert. Neben TDE-Verschlüsselung der Daten auf Blockebene im Datenfile wird auch das Produkt Database-Vault eingesetzt. Dies schützt die Kundendaten vor dem Auslesen der Administratoren. Ebenfalls kann die OCI mit verschiedenen Zertifikaten und Compliances (ISO 27001, SOC 1, SOC 2, SOC3 etc.) aufwarten, siehe auch: https://cloud.oracle.com/en_US/cloud-compliance

Oracle Cloud Infrastructure (OCI) – Autonomous Database

Oracle stellt mit der „Autonomous Database“ eine spezielle Variante ab 18c exklusiv (Cloud-only) in der Public Cloud bereit. Diese Datenbank soll sich selbst reparieren, selbst administrieren, Indices selbstständig anlegen und sich selbst schützen

| Oracle Datenbank-Lizenz | Pay-as-you-go OCPU pro Stunde inkl. Lizenzen | Monthly Flex OCPU pro Stunde inkl. Lizenzen | Pay-as-you-go OCPU pro Stunde ohne Lizenzen (BYOL) | Monthly Flex OCPU pro Stunde ohne Lizenzen (BYOL) |
|--|--|---|--|---|
| Database Enterprise Edition (inkl. Lizenzen) | 0,72422784 € | 0,4828185624 € | | |
| Enterprise Package (BYOL) | | | 0,26071843 € | 0,1738122875 € |
| Pro Jahr = 365 Tage x 24 Stunden x 16 OCPUs x Enterprise Edition / Package | 101.508 € | 67.672 € | 36.542 € | 24.362 € |

Tabelle 1: Preisbeispiel, Preise ohne Gewähr, Stand 09/2019

(Self-Driving, Self-Repairing, Self-Securing). Die „Autonomous Database“ soll mit einer Verfügbarkeit von 99,995% aufwarten.

Auf der Oracle-Cloud-Seite kann man sich für eine Data-Warehouse- oder Transaction-Processing-Variante entscheiden. Diese unterscheiden sich im Wesentlichen durch unterschiedliche Parametrisierungen und unterschiedliche Komprimierungsvarianten.

Bei der „Autonomous Database“ ist die „EE-Extreme-Performance-Option“ automatisch dabei. Letztendlich wird die Datenbank auf einer Exadata in einem Real Applikation Cluster (RAC) mit Active-Data-Guard über zwei Regionen betrieben.

Oracle Cloud Infrastructure (OCI) – Oracle Cloud at Customer

Mit der „Oracle Cloud at Customer“ bietet Oracle an, die Cloud beim Kunden abzubilden. Das bedeutet, dass somit auch die „Autonomous Database“ direkt beim Kunden eingesetzt werden könnte. Die Daten liegen dann beim Kunden, aber der Support erfolgt direkt von Oracle über eine VPN-Leitung. Ob dies nun mehr Sicherheit für den Kunden bedeutet, muss jeder selbst entscheiden.

Oracle Cloud Infrastructure (OCI) – Vor- und Nachteile

Vorteile:

- Aktuelle Oracle-Datenbank-Versionen mit speziellen Features, zum Beispiel Autonomous Database
- In der Enterprise Edition inklusive der Lizenzen sind auch Features und Optionen dabei, wie etwa das Diagnostic- und Tuning Pack
- TDE auch für SE2
- Ausgehendes Datenvolumen 10 TB / Monat kostenlos

Nachteile:

- Als Datenbanken nur Oracle im Portfolio
- Erst seit einigen Jahren Anbieter für Cloud

Bewertung:

Für den Betrieb der Oracle-Datenbanken ist OCI ideal geeignet. Mit der Kooperati-

on mit „Microsoft Azure“ werden die Angebote mit Sicherheit noch interessanter werden.

Amazon Web Services (AWS) – Angebot

Amazon (AWS) ist ein langjähriger etablierter Public-Cloud-Anbieter. Das Angebot von Amazon Web Services (AWS) beschränkt sich auf die Modelle IaaS (Amazon EC2) und PaaS (Amazon RDS).

Aktuell sind die Oracle-Versionen 11.2.0.4, 12.1.0.2, 12.2.0.1, 18.0.0.0 und 19.0.0.0 im Angebot.

IaaS

- Amazon EC2
 - Physikalische Server
 - Dedizierte/r Server pro Kunde
- Virtuelle Server
 - Keine dedizierten Server

PaaS

- Amazon RDS
- Die ausgehenden Datenübertragungskosten betragen ab 1 GB bei bis zu 10 TB – 0,09 US-Dollar pro Monat. Kalkulator: <https://awstcccalculator.com>

Amazon Web Services (AWS) – Prozessorlizenzen bei BYOL

Oracle hat die BYOL-Regeln für die Konkurrenten Amazon EC2 und RDS und Microsoft Azure nachteilig geändert (siehe Abbildung 2).

Zwei vCPUs bei Hyper-Threading (Standard bei den Cloud-Anbietern) entsprechen einer Prozessorlizenz, das heißt ein physikalischer CORE entspricht somit einer Prozessorlizenz.

Für die Enterprise Edition wird also der übliche CORE-Umrechnungsfaktor nicht berücksichtigt. Bei On-Premises und der OCI mit Intel-Prozessoren gilt der CORE-Umrechnungsfaktor von 0,5. Zwei physikalische Cores entsprechen hier nur einer Prozessorlizenz.

Amazon Web Services (AWS) – Vorteile/Nachteile

Vorteile:

- PaaS mit verschiedenen Datenbanken

im Angebot, zum Beispiel Amazon Aurora oder MySQL, deutlich günstiger als Oracle-Datenbanken

- Langjähriger und etablierter Anbieter von Cloud-Produkten

Nachteile:

- Mit BYOL ist die Oracle-Prozessor-Berechnung doppelt so teuer wie in der OCI
- Oracle-Versionen oft nicht auf dem neuesten Stand
- Datenübertragungskosten ausgehend im Vergleich zur OCI sehr hoch

Bewertung:

Oracle-Datenbanken sollte man besser in der OCI betreiben. Möchte man die eigenen Lizenzen einsetzen (BYOL), müsste man die doppelte Anzahl der Lizenzen mit einbringen.

Wenn es allerdings nur um einen Datenbankservice geht, könnten günstigere Datenbanken in Betracht gezogen werden, beispielsweise MySQL. Da Amazon AWS auch ausgehende Datenübertragungskosten ohne Freivolumen in Rechnung stellt, wird der Weg raus aus der Cloud wiederum sehr teuer werden.

On-Premises gegen Cloud

In einer Gegenüberstellung von On-Premises vs. OCI Public Cloud möchte ich an einem Beispiel die Kosten bewerten.

Beispiel-Konfiguration:

- Standard Edition 2
- Server 2 CPUs à 8 Cores, insgesamt 16 Cores (Intel)
- Memory 128 GB, SGA 46 GB
- Plattenplatz für die Datendateien 400 GB
- Backup-Platz 2 TB
- Datenübertragung ausgehend 50 GB/ pro Tag

On-Premises

Auf On-Premises wird ein Server mit 2 CPUs mit jeweils 8 Cores mit 128 GB Memory und 2 x 1 TB SSDs angenommen. Mit einer Wartung über 5 Jahre und einer 4-stündigen Reaktionszeit sind Kosten von ca. 20.000 € (5 Jahre) anzusetzen.

Hinzu kommen die Oracle-Lizenzen und -Wartungskosten. Hier werden bei

Licensing Oracle Software in the Cloud Computing Environment

Approved Vendors

This policy applies to cloud computing environments from the following vendors: **Amazon Web Services – Amazon Elastic Compute Cloud (EC2), Amazon Relational Database Service (RDS) and Microsoft Azure Platform (collectively, the ‘Authorized Cloud Environments’)**. This policy applies to [these Oracle programs](#).

For the purposes of licensing Oracle programs in an Authorized Cloud Environment, customers are required to count as follows:

- **Amazon EC2 and RDS** – count two vCPUs as equivalent to one Oracle Processor license if hyper-threading is enabled, and one vCPU as equivalent to one Oracle Processor license if hyper-threading is not enabled.
- **Microsoft Azure** – count two vCPUs as equivalent to one Oracle Processor license if hyper-threading is enabled, and one vCPU as equivalent to one Oracle Processor license if hyper-threading is not enabled.

When counting Oracle Processor license requirements in Authorized Cloud Environments, the Oracle Processor Core Factor Table is not applicable.

Abbildung 2: Lizenzregelung BYOL für Oracle-Konkurrenten [4] (Quelle: © Oracle)

der SE2 zwei Prozessorlizenzen pro Sockel mit 15.717 € und somit insgesamt einmaligen Kosten in Höhe von 31.434 € berechnet. Die Oracle-Wartungskosten belaufen sich für zwei Prozessorlizenzen auf 34.570 € über 5 Jahre gerechnet (2 x 3.457 € x 5 Jahre).

Insgesamt ermitteln wir somit Kosten in Höhe von 17.200 € pro Jahr (20.000 € + 31.434 € + 34.570 € / 5 Jahre).

Hinzu kommen Kosten für das Personal und das eigene Rechenzentrum und dessen Nutzung. Weitere Storage- und Backup-Kosten und Datenübertragungen müssen nicht weiter berücksichtigt werden, weil die Ressourcen im Rechenzentrum selbst bereits zur Verfügung stehen.

Oracle Cloud Infrastructure (OCI) – Datenbank auf Bare Metal Server x7

Als erste Alternative zu On-Premises tritt der physikalische Bare Metal Server x7 an. Dieser Server steht dem Kunden exklusiv zur Verfügung.

Diesen kann man in der OCI mit folgender Konfiguration buchen: 768 GB RAM, 51,2 TB NVMe SSD raw, ~16 TB mit Zwei-Wege-Spiegelung, ~9 TB mit Drei-Wege-Spiegelung.

Es werden die Preise für die Option Universal Credits – Monthly Flex (mind. 1 Jahr Laufzeit) inkl. Lizenzen für die SE2 kalkuliert.

Auf diesen Servern kosten 2 OCPUs 6,434 € pro Stunde, max. 6 weitere OCPUs

| Variante | Kosten pro Jahr | Bemerkung |
|--|-----------------|---|
| On-Premises | 17.200 € | 16 physikalische CPUs (ohne Hyperthreading) |
| PaaS – Datenbank auf dedizierten Bare-Metal-Server | 69.050 € | 8 OCPUs (16 vCPUs) |
| PaaS – Datenbank auf virtuellem Server | 16.917 € | 8 OCPUs (16 vCPUs) |

Tabelle 2: Vergleich On-Premises vs. OCI, Preise ohne Gewähr, Stand 09/2019

können in der SE2 zu je 0,2414092812 € pro Stunde nachgebucht werden. Also können insgesamt nur 8 OCPUs auf dem Bare Metal Server für die SE2 konfiguriert werden. Dies entspricht mit Hyper-Threading 16 vCPUs (virtuellen CPUs) – Also insgesamt 16 Threads, die auch nur in der SE2 genutzt werden können.

Acht OCPUs kosten 7,8824556872 € pro Stunde. Hier ist also auch der Betrieb für den Server mit abgedeckt.

Die Kosten für diesen dedizierten Bare Metal Server betragen pro Jahr 69.050 € (7,8824556872 € x 24 Stunden x 365 Tage).

Hier fallen jedoch auch weiterhin Personalkosten an.

Weitere Posten sind die Storage- und Backupkosten. Die Datenübertragungskosten sind mit ausgehend 10 TB pro Monat inklusive.

Über den speziellen Kostenkalkulator für Bare Metal Server kann man die Preise ausrechnen: <https://www.oracle.com/de/database/bare-metal-cloud-pricing.html>

Oracle Cloud Infrastructure (OCI) – Datenbank auf virtuellem Server

Die zweite Alternative zum physikalischen Bare Metal Server ist ein virtueller Server. Dieser ist deutlich günstiger als die Bare-Metal-Variante. Hier werden allerdings die Ressourcen auch mit anderen geteilt.

Als Storage sind 256 GB–40 TB remote NVMe SSD Block Volumes auswählbar.

Wir nehmen hier auch wieder die Preise im Rahmen der Universal Credits – Monthly Flex (mind. 1 Jahr), inklusive Lizenzen – Variante an.

Eine OCPU kostet 0,2414092812 € pro Stunde. Auch hier werden wieder 8 OCPUs berechnet. Diese kosten insgesamt 1,9312742496 € pro Stunde.

Die Kosten pro Jahr betragen mit 8 OCPUs (16 vCPUs) auf einem virtuellen Server für die SE2 **16.917€** (1,9312742496 € x 24 Stunden x 365 Tage).

Weiterhin müssen hier auch Personal-, Storage- und Backupkosten berücksich-

tigt werden. Die Datenübertragungskosten ausgehend 10 TB pro Monat wären hier ebenfalls wieder inklusive.

Kalkulator: <https://www.oracle.com/de/database/vm-cloud-pricing.html>

Ergebnis:

Auch wenn der Vergleich hinkt und nicht alle Einsparungen der Public Cloud berücksichtigt werden konnten, bietet er jedoch einen ersten Eindruck bezüglich der Basis-Kosten. Die On-Premises- und die betrachtete Virtuelle-Server-Lösung liegen in dieser Kalkulation nahezu gleich auf (siehe Tabelle 2).

In der Bare-Metal- und Virtuellen-Server-Lösung der OCI können nur 8 OCPUs (mit Hyper-Threading 16 vCPUs) konfiguriert werden. In der On-Premise-Lösung kann man hingegen 16 physikalische Cores ohne Hyper-Threading nutzen und somit hat man potenziell auch mehr Leistung.

Die Kosten für den exklusiven Bare Metal Server sind deutlich höher als die Lösung auf On-Premises oder die virtuellen Server in der OCI.

Cloud-Fallen – Die Kosten

Einen neuen Server in ein Rechenzentrum zu integrieren, kann schon von der Beschaffung bis zur Installation und Konfiguration sehr aufwendig sein. Sicherlich fallen daher einige Kosten bei der Cloud-Nutzung weg.

Eine Übersicht der Kosten mit allem Drum und Dran in der Public Cloud ist schwierig zu ermitteln. Das Backup-, Storage- und das ausgehende Datenvolumen und vieles mehr muss bei der Kalkulation berücksichtigt werden.

Wenn beispielsweise Applikationen aufgrund von Fehlern zu viele Daten lesen oder die Datenbanken wieder raus aus der Public Cloud sollen, kann es teuer werden. In der Amazon AWS-Cloud muss ab dem ersten GB-Datenübertragungsvolumen bezahlt werden.

Weiterhin müssen die Migrationskosten berücksichtigt werden, um in die Cloud zu gelangen.

Die Preise in der Public Cloud ändern sich gerne mal und werden nach oben korrigiert. Man muss sich die Frage stellen: Wie hoch sind die Preise nach Ablauf einer vertraglichen Bindung, zum Beispiel

bei Monthly Flex nach mehreren Jahren?

Bei der Option BYOL müssen die Supportgebühren für On-Premises weiterhin jährlich bezahlt werden.

Bei den Cloud-Konkurrenten von Oracle müssen die Prozessorlizenzen mit BYOL doppelt gezählt werden. Zu beachten ist hier auch, dass die Prozessorleistung (Taktung) häufig schlechter auf den Cloud-Servern ist als bei aktuell angeschaffter Hardware auf On-Premises.

Cloud-Fallen – Rechenzentrum ablösen?

Ein kompletter Umzug von sehr heterogenen Rechenzentrums-umgebungen in die Public Cloud ist eher unwahrscheinlich. In der OCI sollte/muss das Verschlüsselungs-Wallet für TDE (Verschlüsselung der Daten in den Oracle-Blöcken der Datendateien) im Rechenzentrum des Kunden liegen. Viele Oracle-DBAs haben jedoch mit einer Wallet-Verwaltung keinerlei Erfahrung, insbesondere bezüglich Disaster Recovery und einer Master-Key-Änderung. Geht das Wallet verloren und kann nicht mehr wiederhergestellt werden oder der Master-Key im Wallet passt nicht mehr mit dem Master-Key der Datenbank überein, so kann niemand mehr die Daten aus der Datenbank lesen. Hier würde man sich bessere Hinweise von Oracle wünschen.

Cloud-Fallen – Personal, hier Oracle-DBA

Die Rolle des Oracle-DBAs wird weiter benötigt, nur die Aufgaben verändern sich. Dies wird häufig bei der Kostenkalkulation von Entscheidern nicht berücksichtigt. Lediglich das Patchen und die Installation der Oracle-Software fallen weg, wenn man nur die Datenbank als Service kauft.

Der Oracle-DBA wird zum Cloud-DBA. Der Cloud-DBA wird zum Vordenker, Lenker und Berater für die strategische Planung und Auswahl von Cloud-Technologien. Er muss sich in neuen Cloud-Tools, HV- und Cloud-Lösungen sowie Preisstrukturen der Cloud auskennen. Weiterhin muss er die Migration der Datenbanken in die Public Cloud durchführen. Zusätzlich ist die Koordination von

Patching und Downtimes mit dem Cloud-Anbieter seine Aufgabe. Er muss die Verwaltung der hybriden Umgebungen (On-Premises und Cloud) übernehmen.

Viele Standard-Aufgaben eines DBAs fallen nicht weg und müssen weiter durch den Cloud-DBA bearbeitet werden: das Sizing der Datenbanken, die Datenbank erstellen und konfigurieren, die Benutzerverwaltung, Backup-/Recovery- und Performance-Analysen und Tuning, Fehleranalysen und noch vieles mehr.

Cloud-Fallen – Servicezeiten/Patching

Patches und Sicherheits-Patches in IaaS-beziehungswise Cloud-Umgebungen werden gerne vom Cloud-Anbieter mit kurzen Vorlaufzeiten mit einer Downtime angekündigt. Dies führt zu ungeplanten Ausfallzeiten, wenn man sich nicht gerade die „Autonomous Database“ leistet.

Angenommen Sie haben einen kritischen Fehler in ihrer Oracle-Datenbank und Oracle muss für ihre spezielle Situation einen Interims-Patch entwickeln. Auf On-Premises können sie den Interims-Patch sofort nach Erhalt einspielen. Doch in der Public Cloud werden sie nicht in jeder Konstellation direkt eingespielt. Im schlimmsten Fall warten sie, bis der Interims-Patch im nächsten Release-Update zur Verfügung steht.

Die Ansprechpartner, Service-Levels und Response-Zeiten müssen hier auch hinterfragt werden. Klar ersichtlich sind sie nicht auf den Seiten der Public-Cloud-Anbieter.

Cloud-Fallen – Netzwerk-Anbindung

Die Wahl der Netzwerkanbindung zwischen dem Kunden und dem Public-Cloud-Anbieter kann sehr entscheidend für die Performance der Gesamtapplikation sein. Aber auch die Region der Public Cloud und die Entfernung zum Kunden sind wichtig für eine gute Netzwerk-Latenz.

Wenn die Applikations- und Datenbankserver gemeinsam in der Public Cloud stehen, sind weniger Probleme zu erwarten. Bei einem Zugriff auf die Public Cloud über das frei zugängliche Inter-

net können Performance-Schwankungen auftreten.

In der OCI kann man zwischen der normalen Internetanbindung, einem Fast-Connect- oder einer VPN-Verbindung auswählen. Empfehlenswert wäre also eine eigene Standleitung (Fast Connect oder VPN) zur Public Cloud.

Cloud-Fallen – Performance

Bei einer Auswahl einer Public Cloud liegen zu Beginn erst mal keine Performance-Werte vor. Bei den Cloud-Anbietern kann man zwischen dedizierten physikalischen und virtuellen Servern wählen. Bei den virtuellen Servern stehen die Ressourcen dem Kunden nicht exklusiv zur Verfügung. Interessant wäre hier eine eigene, langfristige Messung der Performance-Werte und keine Stichprobenmessung.

Cloud-Fallen – Migration in die Cloud

Die Migration der Oracle-Datenbanken in die Cloud geht nur einfach und fast ohne Downtime, wenn Kunde und Cloud-Anbieter die gleiche Oracle-Version haben. Soll hier auch gleichzeitig die Version angehoben werden, so wird es schwieriger und aufwendiger. Besser wäre es dann, das Update der Datenbank bereits auf On-Premises vorher durchzuführen. Zu beachten ist hier, dass Oracle in der OCI bei bereitgestellten Datenbanken bereits die Container-Datenbank-Technik (CDB) anbietet. Ab der Oracle-Version 20c wird dies zwar auch auf On-Premises Standard, jedoch müssen einige DBAs noch mit dieser Technik vertraut gemacht werden.

Die klassischen Möglichkeiten mit einer Downtime sind:

- Export/Import mit Data Pump
- Transportable Tablespace
- CDB: Unplugging/Plugging beziehungsweise Remote Cloning
- Manuelle Standby-Variante mit SE2

Methoden mit wenig Downtime:

- DataGuard (Enterprise Edition) beziehungsweise Logical DataGuard (Logical Rolling Upgrade), selten praktiziert

- Zero Data Loss Recovery Appliance (ZDLRA), Cross-Platform-Datenbank-Migration
- (Big Endian to Small Endian)
- GoldenGate auf On-Premises und GoldenGate Cloud Services

Wann ergibt die Cloud Sinn?

Grundsätzlich muss man sich die Frage stellen, ob man personenbezogene und unternehmenskritische Daten in die Public Cloud geben will beziehungsweise darf.

Sinnvoll ist die Verwendung der Public Cloud für kleine Datenbanken, die wenige Ressourcen benötigen, wie etwa Entwicklungs-, Test- und Schulungsdatenbanken. Optimal wäre es dabei, wenn der Kunde die Daten bereits selbst anonymisiert in die Public Cloud legt. Auch ist die Public Cloud hervorragend für Datenbanken, die nur wenige Male im Monat verwendet werden, geeignet.

Für kleinere Unternehmen ohne Rechenzentrum bietet die Public Cloud eine gute Möglichkeit, die Datenbanken professionell zu betreiben. Man muss aber auch einen Blick links und rechts von Oracle wagen. Denn hier können sie noch mehr Geld sparen, wenn sie günstige Datenbanken, wie zum Beispiel MySQL, verwenden würden.

Fazit

Die Public Cloud ist nicht mehr aufzuhalten. Bevor Sie jedoch mit ihrer Oracle-Datenbank umziehen, ist eine gute Analyse und Beratung notwendig. Die Preise sind in der Public Cloud schwer im Vorfeld zu kalkulieren. Die Cloud-Kosten sind häufig höher als erwartet und die Kosteneinsparungen nicht so hoch wie gedacht. Die Public Cloud ist auch eine Vertrauenssache und hier müssen die Anbieter für noch mehr Transparenz und Vertrauen sorgen.

Das ganze eigene Rechenzentrum komplett durch die Public Cloud zu ersetzen, wird nur schwer realisierbar sein. Es wird daher auch weiterhin noch hybride Umgebungen (On-Premises und Public Cloud) geben.

Mit der Nutzung einer Multi Cloud kann man die Vorteile der einzelnen Cloud-Anbieter voll ausschöpfen. Aber

all dies muss auch von Menschen verwaltet und administriert werden. Für den Betrieb von Oracle-Datenbanken hat die Oracle Cloud Infrastructure (OCI) die Nase vorn. Testen Sie auf jeden Fall ihren Cloud-Anbieter bezüglich Netzwerk-Latenzen und Service Levels, bevor Sie in die Public Cloud gehen.

Quellen

- [1] Gartner, 11/2019
- [2] Cloud-Migration langwieriger und teurer als gedacht, crn.de, 09/2018
- [3] Firmen unterschätzen Cloud-Migration, cloudcomputing-insider.de, 09/2018
- [4] Oracle, Lizenzregelung BYOL für AWS und Azure <https://www.oracle.com/assets/cloud-licensing-070579.pdf>

Über den Autor

Michael Skowasch ist Principal Consultant bei der ORDIX AG. Seit über 20 Jahren berät er Kunden in Oracle-Themen, speziell in den Bereichen Hochverfügbarkeit und Lizenzberatung. Zusätzlich zu seiner Beratertätigkeit ist Michael Skowasch auch als Referent im ORDIX-Seminarzentrum im Einsatz.



Michael Skowasch
msk@ordix.de



HYBRID

Infrastruktur-Integration für hybride Oracle Clouds

Holger Nicolay, Interxion Deutschland

Der Siegeszug der Public Clouds ist nicht mehr aufzuhalten, es ist eher eine Frage des ‚Wann‘ und nicht mehr des ‚Ob‘ ein Unternehmen geeignete Workloads aus der Public Cloud konsumiert. Üblicherweise werden Anwendungen aber nur Schritt für Schritt in eine Public Cloud verlagert, zudem Services von verschiedenen Cloud-Providern bezogen. Hierdurch ergeben sich völlig neue Herausforderungen, die sich nicht auf der Applikationsebene oder mittels Cloud-Migration lösen lassen, sondern vielmehr auf der Infrastrukturebene und im Netzwerkbereich stattfinden.

Was passiert mit IT-Workloads ?

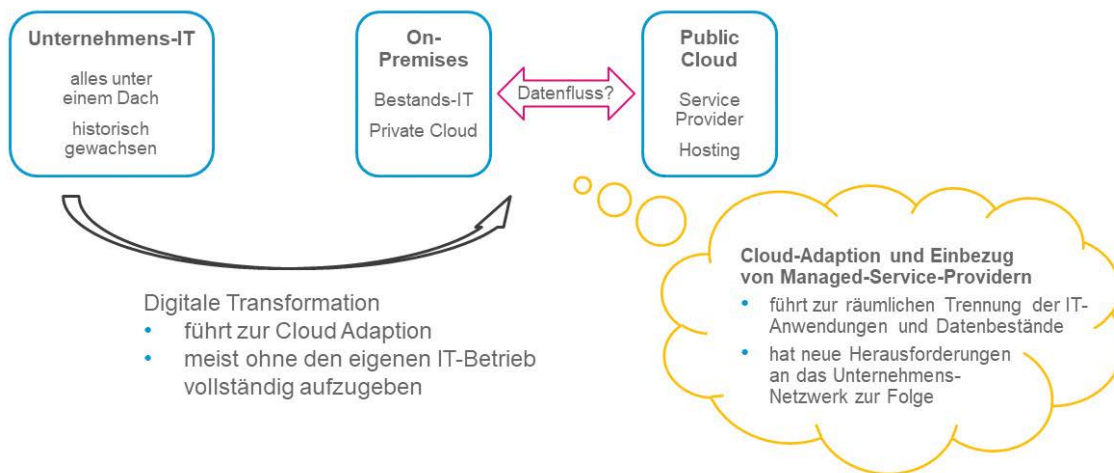


Abbildung 1: Der Einbezug von Public Cloud oder Managed Services, die im Rechenzentrum des Providers erbracht werden, hat unweigerlich Implikationen auf der Netzwerkebene zur Folge. (Quelle: Holger Nicolay)

Natürlich hoffen die Public-Cloud-Anbieter, dass Unternehmen möglichst all ihre IT-Services aus ihrer Cloud beziehen. Das hätte unzweifelhaft Vorteile – vornehmlich für den Cloud Provider, oberflächlich betrachtet wegen geringerer Ausbildungs-, Migrations- und Integrationsaufwände, aber auch für den betreffenden Kunden. Üblicherweise besteht ein Migrationsprozess – egal, ob Cloud oder ganz traditionell – aus verschiedenen Teilschritten, und so ist der Betrieb einer hybriden Cloud oder hybrider IT zumindest als Teil des oft jahrelangen Migrationsprozesses Realität.

Immer mehr Unternehmen erkennen in der hybriden Cloud zudem eine strategische Option, erlaubt dieses Konzept doch die Verarbeitung von Daten genau dort, wo diese am besten aufgehoben sind. Die Eignung von Applikationen zur Verlagerung in die Public Cloud ist aus vielerlei Perspektiven zu analysieren, denn nur so wird offensichtlich werden, dass aus Performanz-, Kosten- und Compliance-Gründen nicht alle Services und Applikationen gleich zu behandeln sind. Und so mag es mal die Public Cloud sein, mal Equipment-Housing bei Colocation-Providern und mal ganz traditionell das selbst betriebene Rechenzentrum vor Ort, welches als Sieger einer solch ganzheitlichen Betrachtung hervorgeht.

Vorüberlegungen

Die allerwenigsten Applikationen sind als Insellösung zu betreiben, sie weisen Interaktion und Datenfluss mit anderen IT-Services auf. Damit steht die Integration der verschiedensten selbst betriebenen mit den in die Cloud verlagerten Applikationen an. Da die relevanten globalen Cloud-Provider ihre Services mittlerweile von deutschen Rechenzentren aus erbringen, sind der Ort der Datenhaltung transparent und Signallaufzeiten zwischen Anwender und Cloud überschaubar geworden. Und doch bleibt stets eine räumliche und nicht zu vernachlässigende Entfernung zwischen Nutzern und der Cloud, die im Sinne einer guten Planung und Ende-zu-Ende-Betrachtung der aus User-Sicht zu erwartenden Performanz analysiert werden muss.

Befand sich in der „alten IT-Welt“ die Hauptverwaltung auf demselben Firmengelände wie Produktion und unternehmenseigenes Rechenzentrum, so entsteht mit der Verlagerung von Services zu Dritten unweigerlich eine zuvor nicht existente räumliche Distanz. Dabei ist es unerheblich, ob Hosting, Managed Services oder Public-Cloud-Services genutzt werden, die Bedeutung der räumlichen Entfernung steigt dadurch, dass nicht mehr in sich mehr oder weniger abgeschlossene Systeme verlagert werden,

sondern Teile Business-kritischer Applikationen (siehe Abbildung 1).

Insofern wäre es fatal, für die Cloud-Migration die einzelnen Applikationen nur aus IT-Sicht isoliert zu betrachten. Vielmehr steht das Zusammenspiel der IT-Services untereinander im Fokus: der Frontend-Webserver mit der Backend-Datenbank, die Applikation mit der Speicherumgebung, ganz allgemein vor allem der Datenfluss zwischen Public-Cloud-Applikationen mit solchen, die noch im eigenen Rechenzentrum betrieben werden.

Für den IT-Spezialisten lassen sich solche Fragestellungen von Interoperabilität scheinbar mit APIs und Konnektoren lösen. Die eigentliche Herausforderung jedoch liegt im Netzwerkbereich auf Infrastrukturebene, also bei denjenigen Unternehmenseinheiten, die in der eigenen Wahrnehmung stets zu spät in Projekte eingebunden werden. Aber auch das Betätigungsfeld der Netzwerk-Spezialisten verlagert sich: Von SAN und LAN ins WAN (Storage, Local und Wide Area Network).

Schritt für Schritt die Applikationen analysieren

Doch der Reihe nach: In der idealisierten „alten On-Premises-IT-Welt“ befinden sich weite Teile der IT-Infrastruktur – bildlich gesprochen – unter einem Dach.

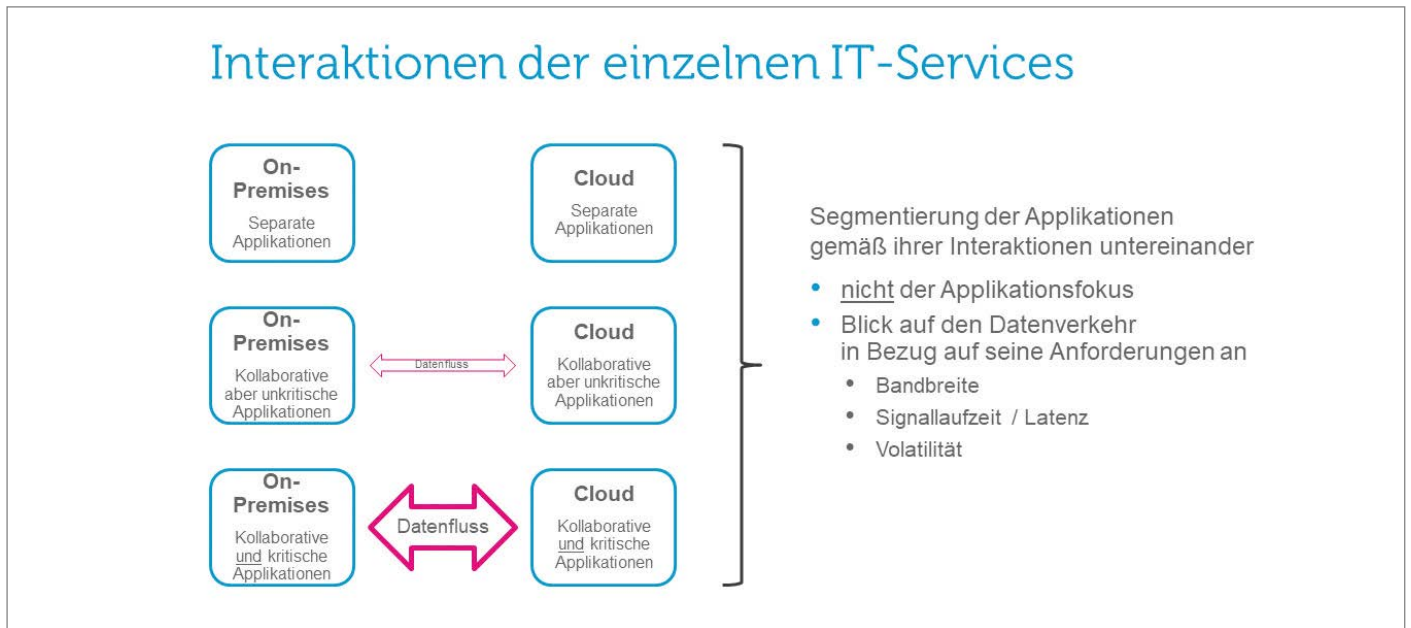


Abbildung 2: Neben der Eignung von Applikationen zur Verlagerung in die Public Cloud sollten deren Interaktionen mit anderen, beispielsweise im eigenen Rechenzentrum verbleibenden Applikationen analysiert werden (Quelle: Holger Nicolay).

Mit Verlagerung in die Cloud wird nun ein Teil der Applikationen am bisherigen Standort, ein anderer Teil aus der Public Cloud erbracht. Unternehmen in den Glasfaser-erschlossenen Ballungsgebieten des Rhein-Main-Gebietes mögen diese Distanz und die daraus resultierende Signallaufzeit vernachlässigen können, Unternehmen in von Telekommunikationsunternehmen eher spärlich erschlossenen Regionen oder mit großer Distanz zur deutschen „Cloud-Hauptstadt“ Frankfurt am Main sind in Sachen Datenübertragung und Signallaufzeit gefordert.

Doch nicht alle Applikationen, die an unterschiedlichen, räumlich voneinander entfernten Standorten betrieben werden, werden zum Problem für die Netzwerk-Abteilung. Vielmehr ist angeraten, nicht nur die Eignung von Applikationen zur Verlagerung in die Cloud aus IT-Sicht zu erörtern, sondern auch detailliert deren Zusammenspiel mit anderen Komponenten in Hinblick auf den Datenfluss derselben untereinander zu analysieren (siehe Abbildung 2).

So wird man Applikationen vorfinden, die weitgehend autonom von anderen zu betreiben sind. Andere haben Datenfluss-Interaktionen untereinander, die aber keinerlei Kritikalität aufweisen. Kollaborative Applikationen mit kritischem Datenfluss zu anderen Instanzen sind jedoch diejenigen, auf die das Augenmerk

bei der Infrastruktur-Integration hybrider Clouds gelegt werden sollte.

„Kritischer Datenfluss“ kann dabei mehrere Ausprägungen aufweisen: Applikationen mit viel Datenaustausch untereinander erfordern Bandbreite, solche mit zahlreichen oder bereits in der Einzelbetrachtung zeitkritischen Interaktionen rufen nach zuverlässig konstanter oder gar geringstmöglicher Latenz. Und wiederum andere sind aus Datenschutz- oder Compliance-Gründen gesondert zu betrachten.

Netzwerk-Spezialisten einbinden

Unkritische Applikationen ohne relevanten Datenaustausch untereinander können leicht über das öffentliche Internet miteinander verbunden werden. Zwar nimmt man hier die Nachteile des öffentlichen Internets wie dessen variables Routing, unsicheren Datendurchsatz und kaum voraussagbare Signallaufzeiten in Kauf. Dem stehen jedoch die unbestreitbaren Vorteile der einfachen Implementierung gegenüber. Von einer wirklichen Integration ist aber hierbei kaum zu sprechen, solche Applikationen erfordern es eben gerade nicht, sich hier weitergehende Gedanken zu machen.

Anders verhält es sich mit den „Bandbreiten-hungrigen“ oder zeitkritischen Ap-

plikationen, hier reicht die unvorhersehbare Performanz des öffentlichen Internets nicht aus, einen verlässlichen Datenaustausch untereinander zu gewährleisten. Dies mag an Unklarheiten über die Tag und Nacht zuverlässig zur Verfügung stehende Bandbreite liegen oder aber an den je nach Beanspruchung des Internets durch andere Nutzer variablen Übertragungszeiten, der Latenz. Auch die Gefahr von DDOS-Attacken, bei denen Netzwerke mit Anfragen überhäuft und damit bis zum faktischen Stillstand in die Knie gezwungen werden, sowie der grundsätzlich unsichere Übertragungsweg, weil eben die öffentliche Infrastruktur Dritter genutzt wird und kaum Einfluss auf den letztendlichen Weg des Datenpakets genommen werden kann, mögen von der Internetnutzung abhalten.

Dort, wo beispielsweise durch regulatorische Vorgaben ein Datenaustausch über das öffentliche Internet trotz aller Sicherheitsmechanismen, die moderne Verschlüsselungsmethoden bieten, ausgeschlossen wird, schaffen Telekommunikationsunternehmen Abhilfe. Diese stellen ihren Kunden eine Standleitung in Form von privaten Festverbindungen oder Ethernet-Kopplungen bereit, die zudem konstant hohen Datendurchsatz und verlässliche Signallaufzeiten gewährleisten.

Die eigentlichen Herausforderungen, welche die räumliche Distanz birgt, kön-

nen aber auch private Cloud-Anbindungen durch Telekommunikationsprovider nicht lösen. Schneller als „Distanz geteilt durch Lichtgeschwindigkeit“ werden hier keinerlei Daten transportiert, hinzu kommen Verzögerungen durch die Signalverarbeitung in den einzelnen Netzelementen. Wer hier für Optimierung sorgen muss oder möchte, wird Colocation-Services eines solchen Anbieters in Erwägung ziehen, der auch die betreffenden Rechenzentren der Public-Cloud-Provider – meist in Frankfurt am Main – beheimatet (siehe Abbildung 3).

In solchen Rechenzentren können Serverschränke in Form von Equipment-Housing-Services angemietet werden. So können Unternehmen ihre Applikationen in Eigenregie und „Tür an Tür“ mit den Public-Cloud-Providern betreiben. Bei einigen wenigen Colocation-Providern ist nicht nur ein einziger Public-Cloud-Provider wie die Oracle Cloud Infrastructure (OCI) präsent, sondern zudem weitere globale Hyperscaler wie

Amazon Web Services, Microsoft Azure oder die Google Cloud.

In der Colocation schrumpft nicht nur die räumliche Distanz zwischen den eigenbetrieblenen und den in der Cloud gehosteten Applikationen auf die Länge einer Inhouse-Verkabelung, es stehen zudem die Cloud-Networking-Services wie der „OCI Fast Connect“ zur Verfügung. Mit solchen Services stellen die Cloud-Provider ihren Kunden einen eigenen (virtuellen) Cloud-Port zur Verfügung – also eine private Direktanbindung mit SLA-Zusage (bis zu 99,999% Verfügbarkeit) und zugesicherter Bandbreite. Da dies unter vollständiger Umgehung des öffentlichen Internets geschieht, sind hier keinerlei Schwankungen in der Signallaufzeit oder der zur Verfügung stehenden Bandbreite zu befürchten. Zudem bestehen keinerlei Angriffsmöglichkeiten für Dritte. Mehr noch: Bei Colocation-Providern, welche die Public Clouds zu ihrem Kundenkreis zählen und auf demselben Campus Equipment-Housing für Unternehmens-

kunden anbieten, sind Signallaufzeiten in der Größenordnung von nur einer Millisekunde zu erwarten.

Optionen bei der Implementierung

Hat also ein Unternehmen die prinzipiell zur Migration in die Public Cloud geeigneten Applikation identifiziert und diese zudem nach ihrem Anbindungsbedarf an andere Applikationen und Rechenzentren klassifiziert, steht der eigentlichen Integration auf Infrastrukturebene nichts mehr im Wege.

Üblicherweise verfügen Unternehmen bereits über einen Internetzugang. Dennoch sollte nicht außer Acht gelassen werden, dass der Bandbreitenbedarf mit Verlagerung von Applikationen in die Public Cloud regelmäßig ansteigt. Die Diskussion mit dem Internet-Service-Provider, welche Optionen für Upgrades bestehen und wie das Routing

Robotron-Schulungszentrum

Kompetente Wissensvermittlung mit Durchführungsgarantie

ORACLE APPROVED
EDUCATION CENTER

Java SE: Programming I

▶ 28.09. - 02.10.2020

Java SE: Programming II

▶ 11.05. - 15.05.2020

▶ 19.10. - 23.10.2020

NEU: Praxisworkshops auf der Datenbankversion 19c!

Basis-Workshop Administration Oracle 19c SE2 (NONCDB)

Aufbau-Workshop Administration Oracle 19c SE2 (CDB)

Kompakt-Workshop Administration Oracle 19c SE2 (NONCDB und CDB)

▶ 25.05 - 29.05.2020

Buchen Sie
jetzt Ihren
Kurs!



Mehr Kurse in unserem Schulungszentrum in Dresden finden Sie hier

robotron[®]

Robotron Datenbank-Software GmbH
www.robotron.de/schulungszentrum

Cloud-Anbindungsoptionen

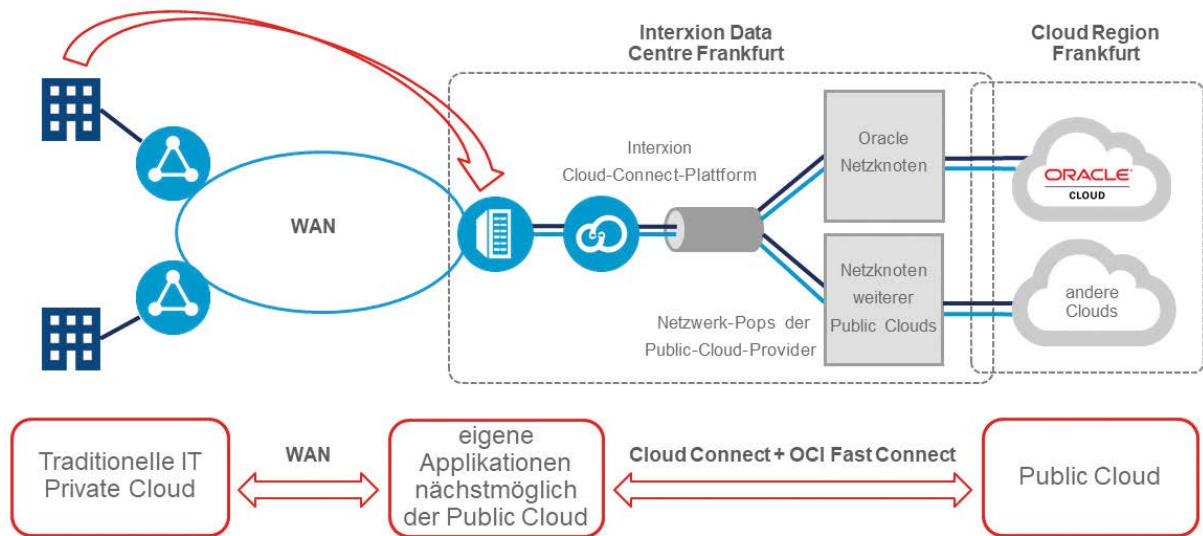


Abbildung 3: Applikationen mit kritischem Datenfluss untereinander haben einen hohen Integrationsbedarf, der mittels Cloud-Anbindung des eigenen Rechenzentrums durch einen Telekommunikationsprovider oder die Verlagerung der Applikation selbst in eine geeignete Colocation realisiert werden kann (Quelle: Holger Nicolay).

optimiert werden kann, sollte frühzeitig angegangen werden. Wohl dem, der zudem die Auswahl zwischen Providern hat und so zu demjenigen Internet-Anbieter wechseln kann, der mit dem besser ausgebauten Backbone in Richtung Public Clouds aufwartet (siehe Abbildung 4).

Deutschland verfügt, gerade auf kommunaler Ebene, über eine Vielzahl an lo-

kalen und regionalen Telekommunikationsunternehmen. Meist sind es aber nur die großen oder internationalen Anbieter, welche von Oracle als „Fast Connect-Partner“ akkreditiert wurden, also einen Ende-zu-Ende gemanagten Bandbreiten-Service in die Oracle Cloud anbieten können. Doch auch die kleineren Anbieter können, gerade im Zusammenspiel mit ihrem Colocation-Partner, bei dem

sie bereits Internetverkehr austauschen und WAN-Anbindungen koppeln, Anbindungen in die Public Clouds auf Projektbasis herstellen.

Auch die meisten Colocation-Provider verfügen nicht über direkte Anbindungsmöglichkeiten an die Public-Cloud-Provider, sondern bedienen sich zur Überwindung dieser „letzten Meile“ Dritter wie Telekommunikationsanbietern oder In-

Entscheidungsbaum

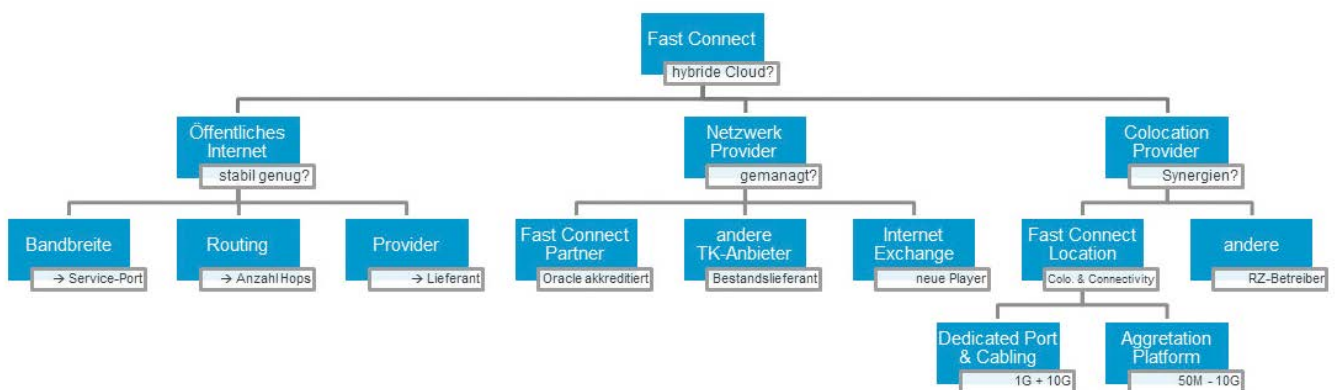


Abbildung 4: Für die Integration selbst betriebener Applikationen mit denjenigen aus Public Clouds konsumierten stehen zahlreiche Anbindungsoptionen und deren Optimierungsmöglichkeiten zur Verfügung, hier als Entscheidungsbaum am Beispiel des „Oracle Cloud Infrastructure Fast Connect“-Services illustriert (Quelle: Holger Nicolay).

Multi Cloud im Colocation-Data-Center

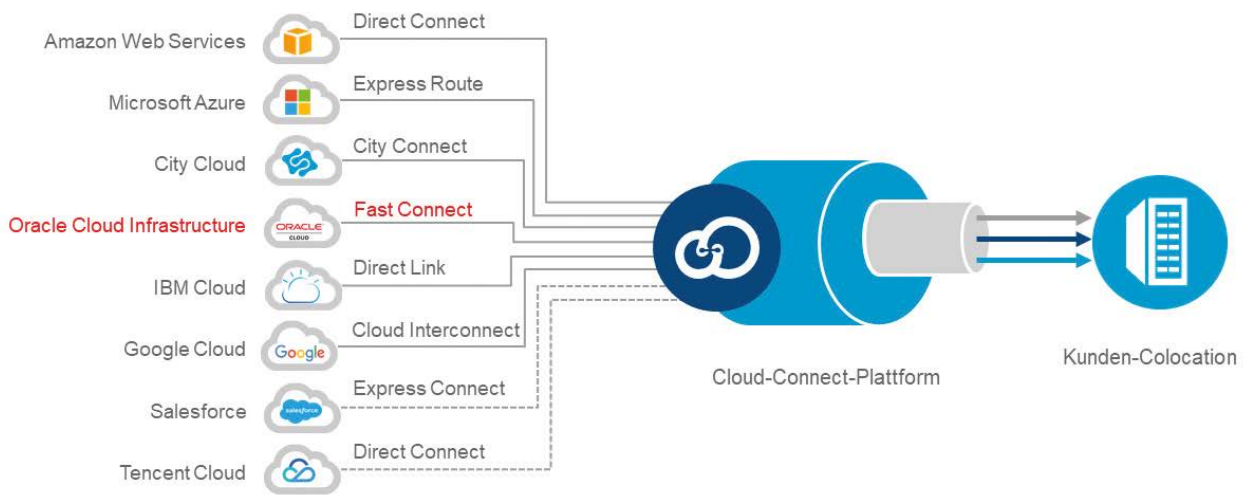


Abbildung 5: Die Integration mehrerer Public Clouds zu einer hybriden oder Multi-Cloud-Umgebung werden am besten in einem Colocation-Data-Center implementiert, das all diese Cloud bereits zu seinem Kundenkreis zählt und eine entsprechende Aggregationsplattform zur Anbindung derselben offeriert (Quelle: Holger Nicolay).

ternet Exchanges. Um diese zusätzliche Schnittstelle zu vermeiden, lohnt sich der Blick auf die sogenannten „Fast Connect Locations“, die Oracle auf seiner Webseite veröffentlicht. Auch Oracles Marktbegeleiter in Sachen Public-Cloud-Services veröffentlichen diese Informationen, so findet man ebenso leicht die „Direct Connect Pops“ von Amazon Web Services oder die „Express Route Locations“ von Microsoft (siehe Abbildung 5).

Bei einem solchen Colocation-Provider haben Kunden zwei Anbindungsoptionen zur Verfügung. Zum einen die dedizierten, aber ungemanteten Glasfaser-Verkabelungen, welche Endkunden über sogenannte Meet-me-Räume an die physischen Ports der Public-Cloud-Plattformen anbinden. Hierbei stehen jedoch ausschließlich die vom Cloud-Anbieter vorgegebenen Bandbreiten – meist 1 oder 10 Gbit/s – zur Verfügung, zudem beschränken sich diese Verbindungen auf einen Campus. Daher betreiben die führenden Colocation-Provider Aggregationsplattformen, mit denen sie hochverfügbare und gemantete Bandbreite mit einer Vielzahl an Zwischengrößen und paneuropäischer Verfügbarkeit anbieten.

Derartige Cloud-Connect-Plattformen erleichtern zudem Multi-Cloud-Implementierungen, weil sie auf einem physischen

Ethernet-Port zum Kunden hin dessen verschiedene, VLAN-basierte Anbindungen an mehrere Cloud-Provider und Cloud-Regionen aggregieren können, ohne dabei die Bandbreite zu überbuchen.

Zusammenfassung

Die Cloud-Migration sieht auf den ersten Blick wie eine klassische IT-Aufgabe aus – die Anpassung und Migration von Applikationen auf eine neue Plattform. Natürlich ist dieser Aspekt die Grundlage allen Handelns und nicht zu vernachlässigen.

Dort jedoch, wo auch im zukünftigen Betrieb das Zusammenspiel von alter und neuer Welt, von Bestands-IT mit Cloud-Applikationen reibungslos vonstatten gehen soll, ist zudem Netzwerk-Expertise gefragt. Die räumliche Trennung der verschiedenen Instanzen macht aus dem ursprünglichen IT-Projekt eine Netzwerk-Erweiterung.

Wer seine zukünftigen Applikationen in der Public Cloud im Sinne einer Hybrid-Implementierung hochperformant an Bestandssysteme anbinden möchte, muss sich mit genau dieser Integration auf Infrastrukturebene befassen. Die gute Nachricht ist, dass es dafür sowohl geeignete und bewährte Konzepte als auch Anbieter am Markt gibt.

Über den Autor

Holger Nicolay ist Business Development Manager bei Interxion, einem führenden europäischen Colocation-Provider mit mehr als 50 Carrier- und Cloud-neutralen Data Centern. Sein Fokus sind die digitale Transformation sowie Connectivity- und Infrastruktur-Konzepte, mit denen Interxion seine Kunden als Infrastrukturpartner und Cloud Connect-Provider bei der Implementierung von hochintegrierten hybriden Clouds und flexiblen Multi-Cloud-Lösungen unterstützt. Holger Nicolay ist Diplom-Informatiker (TU) und hat vor Interxion in verschiedenen Sales-, Marketing- und Management-Rollen in der Telekommunikationsbranche gearbeitet. Frühere berufliche Stationen waren die heutige 1&1 Versatel und Colt Technology Services.



Holger Nicolay
HolgerN@interxion.com



Peterchens Mondfahrt an der Cloud vorbei

Wolf G. Beckmann, TEAM

Die Cloud ist ein weites Feld und bietet viele neue Möglichkeiten, vor allem für die Anbieter. Natürlich werden wir zukünftig an der Cloud in den meisten Fällen nicht vorbeikommen, gerade deswegen lohnt es sich, genau anzuschauen, was man als Anwender wirklich braucht. Hier wird zum einen der Entscheidungsweg gegen den Einsatz der Cloud aufgezeigt und dargelegt, mit welchem System dennoch alle Ansprüche des Szenarios an Flexibilität erfüllt wurden und werden. Anschließend wird erläutert, wie die „On-Premises Cloud“ funktioniert.

Cloud oder nicht Cloud ...

Der Anstoß für die folgenden Betrachtungen war ein Szenario, wie es jetzt typisch für den Entwicklungsbereich eines Unternehmens ist. Die Serverlandschaft soll modernisiert werden und es stand die Entscheidung an, wie weit Server in die Cloud verlagert werden sollten. Vom Ergebnis waren wir selbst überrascht.

Um eine gemeinsame Sicht auf „die Cloud“ zu bekommen, beginnen wir mit einer groben Einteilung, welche Ebenen aus Sicht der Software-Entwicklung die Cloud bietet:

VM / Bare Metal

- Virtuelle Maschinen oder ganze Server.
- Einbindung in eigene Infrastruktur (VPN).
- Entspricht mehr oder weniger dem klassischen Hosting mit mehr Flexibilität.
- Man macht alles selbst, eignet sich daher gut, um vorhandene On-Premises-Szenarien in die Cloud zu verlagern.
- Anbieter (größtenteils) austauschbar, da es jeder im Portfolio hat.

Docker / Kubernetes / Serverless

- Hochskalierbare Systeme, Cloud-Native-Applikationen.
- Aber auch: in Docker gepackte Applikationen.
- Erfordert Um- oder Weiterdenken der aktuellen Entwicklungsstrukturen.

- Anbieter (größtenteils) austauschbar, gerade im Kubernetes-Umfeld.

Special Services

- KI-Systeme, Sprach-Dienste, Übersetzungsservices, Map-Services, alles Mögliche
- auch Datenbanken
- stark anbieterspezifisch

Das Software- und Consulting-Team hat schon früh damit begonnen, die Entwicklung nicht mehr direkt auf dem eigenen Rechner zu erledigen, sondern auf lokale virtuelle Maschinen zu verlagern, um zum einen neue Entwickler durch einfaches Kopieren einer VM in ein Projekt zu holen und zum anderen Projektumgebungen zu konservieren.

Zusätzlich wurde der Fokus für neue Projekte auf die Paketierung mit Containern wie Docker gelegt, da so die Installation und der Betrieb stark vereinfacht werden. Die gemeinsam genutzten Docker-Container wurden vom Team selbst mit einem Open-Source-Produkt von Red Hat betrieben: OpenShift.

So sind auch die ersten Tools fürs Development nicht wie üblich auf einem virtuellen Server, die von der internen Systembetreuung vergeben werden, installiert worden, sondern in Container gewandert. Einer der ersten Server, die in einem Container betrieben wurde, war beispielsweise ein Nexus-Server (hält die Maven Repositories für die Java-Entwicklung).

Nicht zuletzt hat das Team [1] auch mit „Special Services“ zu tun (beispielsweise der Datenbank oder durch die Entwicklung von Voice- und Chatbot-Systemen mit Digital-Assistant-Services von verschiedenen Cloud-Anbietern). Allgemein wurden bei verschiedenen Cloud-Anbietern diverse Systeme mit unterschiedlichen Komplexitätsgraden aufgesetzt; natürlich auch in der Oracle-Cloud.

Grundsätzlich waren für die Modernisierung der Infrastruktur des Software- und Consulting-Teams alle Zeichen für die Cloud gesetzt. Es wurde eine Diskussionsrunde mit der Systembetreuung, der Geschäftsführung (als Geldgeber) und dem Software- und Consulting-Team gestartet. In dieser sollte auf Basis unserer Anforderungen eine konkrete Lösung definiert werden.

Die abzudeckenden Anforderungen, die in der Diskussionsrunde besprochen wurden, sind wohl typisch für Entwicklungsabteilungen:

Für Projekte sollen eigenständige Umgebungen aufgebaut werden

Meist besteht ein Projekt nicht nur aus einem Server. Es umfasst verschiedene Komponenten. In der klassischen Entwicklung wären das ein Datenbank-Server, ein Application-Server, gegebenenfalls native Dienste und eine Handvoll Shell-Scripts, die Zeit- oder Event-gesteuert ablaufen.

Für solch ein Projekt werden in der Entwicklung mindestens zwei solcher Umgebungen benötigt: Development und Test. Diese Umgebungen müssen unabhängig voneinander laufen und natürlich gibt es nicht nur ein Projekt.

Systeme für die Unterstützung der Entwicklung werden benötigt

Die Entwicklung wird immer abhängiger von zusätzlichen Systemen, wie der Versionsverwaltung (GIT), Application-Lifecycle-Management-Tools (Jira oder Polarion), Continuous Integration Server (Jenkins), aber auch ein Wiki für das Wissens-Management wird benötigt.

Beispiel-Systeme sollen Kunden zur Verfügung gestellt werden

In Akquise-Phasen sagt ein Bild oft mehr als tausend Worte; in der Software-Entwicklung eine kleine Beispiel-Anwendung. Diese muss natürlich auch für den potenziellen Kunden zugreifbar sein.

Einfach und schnell sollen Test-Systeme auf- und wieder abgebaut werden können

In der Entwicklung muss man sich häufig mit neuen Technologien auseinandersetzen, da sich die Welt weiterdreht. Auch hierbei werden häufig komplexe Umgebungen gebraucht. Da soll der Aufbau der Umgebung nicht länger als der Test dauern. Deshalb wären Templates oder der Umgebungsaufbau über Skripte wünschenswert. Auch eine Beantragung einer Testumgebung, die sich über viele Diskussionsrunden erstreckt, sorgt dafür, dass die Weiterbildung auf der Strecke bleibt.

Konkret wollte der Entwicklungs-Bereich eine eigene „Spielwiese“ haben.

Tatsächlich hat sich in dieser Besprechungsrunde herauskristallisiert, dass sich ein anderes Konzept als die Cloud für das Team besser rechnet, besser passt

und auch für die Infrastruktur (Systemadministration) der Firma weniger Aufwand bedeutet:

Der interne OpenShift-Server wird vergrößert und es wird bei einem klassischen Hostler (kein Cloud-Anbieter) ein weiterer OpenShift-Server aufgesetzt.

Wie kam es zu der Fokussierung auf OpenShift?

Dazu müssen mehrere Punkte beachtet werden. Der wohl wichtigste Punkt ist:

Es ist keine Entscheidung gegen die Cloud, sondern eine Entscheidung für die konsequente Nutzung moderner Cloud-Technologien.

Ressourcenschonend

Es sollte nicht mehr in klassischen Strukturen, also in Servern (Rechnern) gedacht werden, sondern in flexiblen, ressourcenschonenden Systemen.

Aufgrund des Einsatzes von Containern statt von VMs wird der benötigte Bedarf an Speicher und CPU drastisch reduziert. In der Entwicklung wird beispielsweise als GIT-Server „Gitea“ verwendet. Ein OpenSource-Server, der eine ähnliche Funktionalität wie GitHub mit Managementinterface, Pullrequests etc. zur Verfügung stellt. Wenn dieser Server in einer eigenen VM betrieben wird, verbraucht das Betriebssystem mehr Speicher und CPU als Gitea selbst. Bei anderen Systemen ist die Elastizität groß; wenn das System eingesetzt wird, braucht es kurzzeitig viel Ressourcen, aber die meiste Zeit nicht. Bei einem einfachen VMware-System muss CPU und Speicher fest zugewiesen werden (zumindest bis zum Neustart), das bedeutet, es muss auf die Lastspitzen ausgerichtet werden. Die Virtualisierung

über Docker sieht vor, dass alle Systeme gemeinsam laufen und nur dahingehend abgegrenzt werden, dass sich die Systeme, vereinfacht gesagt, gegenseitig nicht sehen. Alle Container teilen sich CPU und Speicher dynamisch und somit wird auch die Last dynamisch verteilt.

Natürlich kann eingestellt werden, welche maximalen Ressourcen ein Prozess bekommt, damit ein System nicht alle anderen herunterziehen kann.

Um es konkret aufzuzeigen, das externe System besitzt 12 Kerne, 64 GB RAM und 1,5 TB Plattenplatz. Es laufen auf dem System mehrere Oracle-XE-Datenbanken, Payara und JBoss Server, ein Eclipse-Che-System, ein Node.js-Server, ein Wiki-System, ein Gitea-Server, ein Jenkins-Server, Postgres-Datenbanken und noch einiges mehr.

Dennoch ist der Server noch lange nicht ausgelastet, wie in *Abbildung 1* zu sehen ist.

Es ist kein Produktiv-System, deshalb werden alle Systeme nur kurzfristig verwendet und alle Systeme reagieren schnell.

Vereinfachung von Installation und Betrieb

Tatsächlich hat sich gezeigt, dass die meisten Systeme, die wir benötigen, als Docker-Images angeboten werden oder sich leicht selbst in Containern paketieren lassen.

Durch den Einsatz von OpenShift sind die Administration und das Monitoring sehr einfach, es lassen sich auch ohne großen Aufwand ganze Infrastrukturen aufbauen und für wiederkehrende Situationen können Templates angelegt werden.

Ein weiteres Feature ist die Betriebssicherheit. OpenShift kann die Container überwachen und sobald ein Container nicht mehr reagiert, wird er neu gestartet. Im einfachsten Fall gibt man eine Test-URL an, die nur irgendeine Webseite zurückliefern muss.

```
top - 15:35:17 up 110 days, 7:11, 1 user, load average: 2.41, 2.03, 2.08
Tasks: 1110 total, 1 running, 1066 sleeping, 0 stopped, 43 zombie
%Cpu(s): 11.8 us, 8.0 sy, 1.2 ni, 78.0 id, 0.9 wa, 0.0 hi, 0.1 si, 0.0 st
KiB Mem : 65773264 total, 6407396 free, 33043040 used, 26322828 buff/cache
```

Abbildung 1: TOP eines OpenShift-Test-Servers (Quelle: Wolf G. Beckmann)

Kosten

Aufgrund anderer Anforderungen außerhalb des Teams war die Weiterführung des Betriebs einer internen Virtualisierungslösung auf Basis von VMware fest vorgegeben. Deshalb kam es zu der Zerteilung in eine interne und eine externe OpenShift-Installation. Dienste, die ausschließlich intern verwendet werden, bleiben intern. Ansonsten werden sie auf dem gehosteten OpenShift-Server installiert.

Da es sich bei dem externen Server also primär um ein Testsystem handelt, konnten wir günstige Systeme suchen. Fündig wurden wir bei einem Server-Hoster, der Systeme, die für Kundenwünsche zusammengestellt wurden, nach ihrer Laufzeit günstig anbietet. So wurde das oben beschriebene System (12CPUs, 64 GB RAM 1,5 TB Platte) dort für 50€ pro Monat angeboten. In diesem Preissegment konnten die Cloud-Anbieter nicht mithalten.

Selbst der Einwurf, Äpfel mit Birnen zu vergleichen, „Gebraucht gegen Neu“, hilft nicht, da das System für die Anforderungen passend ist. Bei einem Produktivsystem wäre der Einsatz eines Servers gegebenenfalls auch von einem Cloud-Anbieter in Betracht gezogen worden.

OpenShift – die „On-Premises Cloud“

OpenShift ist ein Produkt von Red Hat und wird sowohl kostenfrei als OKD als auch kostenpflichtig angeboten. Das Cloud-Angebot von Red Hat basiert auf OpenShift.

OpenShift ist ein Aufsatz auf Kubernetes. Kubernetes selbst ist ein System, um (Docker)- Container zu managen. Dazu werden unter anderem folgende Features angeboten:

- Virtuelle Umgebungen
Es werden mehrere Container in einem privaten Netzwerk zusammenge-

fasst. Dabei können die Container sich über Namen gegenseitig adressieren.

- (Auto-) Skalierung
Jeder Container kann mehrfach gestartet werden. Intern findet ein Loadbalancing statt. Die Skalierung kann auch aufgrund von Ressourcennutzung automatisch über Kubernetes erfolgen.
- Ausfallsicherheit
Kubernetes kann testen (z.B. über einen URL-Aufruf), ob der Container funktioniert. Wenn nicht, wird er automatisch neu gestartet.
- Clusterfähigkeit
Die Container können über mehrere Nodes (Cluster-Knoten) verteilt werden.

OpenShift erweitert Kubernetes noch um weitere Features, beispielsweise:

- Einfaches Management und Monitoring-Web-Interface
Durch OpenShift können im Browser einfach Projekte angelegt und administriert werden. Dazu gleich mehr.

Cloud oder On-Premise? Mit Expertise ans Ziel.



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.

Mehr zu diesem Thema erfahren Sie an unserem Event über Oracle, Azure und AWS Cloud-Technologien
Save the date: 12. Mai 2020, Olten (CH)

- Benutzerverwaltung
- Templates
Wir haben beispielsweise Templates für eine einfache Oracle-XE-Datenbank oder eine Oracle-XE-DB mit Apex (inkl. https-Zugang).
- Rolling Deployments
Bei einem Update eines Containers wird erst der neue Container hochgefahren. Erst wenn er voll funktionsfähig ist, werden die Zugriffe auf den neuen Container umgeleitet und der alte Container heruntergefahren. Das vermindert die Ausfallzeit bei Updates.

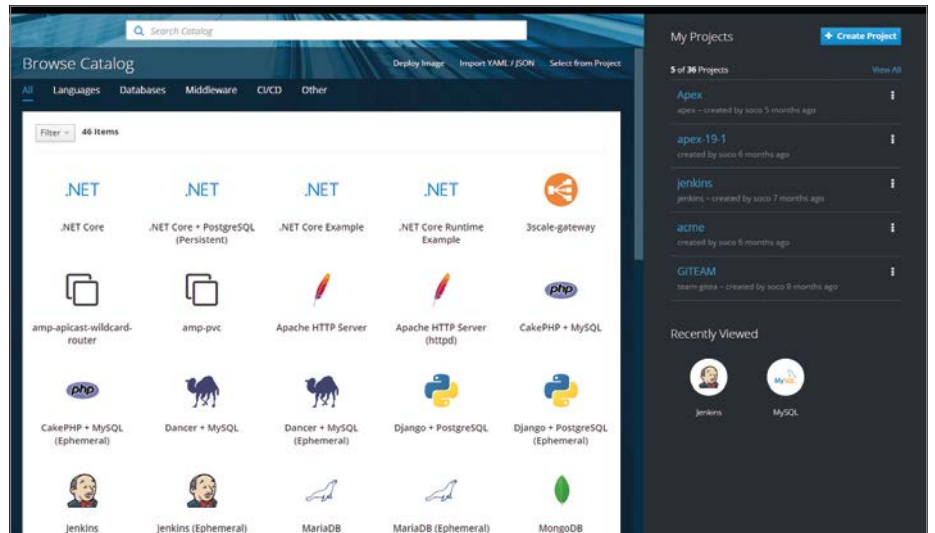


Abbildung 2: Einstiegsseite von OpenShift (Quelle: Wolf G. Beckmann)

Es gibt noch eine Menge weiterer Features, die im Rahmen der Entwicklung sehr spannend sind, die hier allerdings aufgrund von Platzmangel nicht weiter behandelt werden.

Die Installation eines einfachen Ein-Server-Systems war kein Problem. Neueinsteiger finden dazu auch YouTube-Videos, die von Red Hat selbst angeboten werden.

Nach der Installation von OpenShift (OKD) steht dem Benutzer eine Einstiegsseite, wie sie in *Abbildung 2* zu sehen ist, zur Verfügung.

Die Templates werden links angezeigt, über die Leiste rechts kann direkt auf die bereits vorhandenen Projekte zugegriffen werden. Die Darstellung des Status eines Containers ist in *Abbildung 3* zu sehen (hier eine Jenkins-Anwendung).

Um beispielsweise so einen Jenkins-Server in OpenShift aufzusetzen, muss lediglich im Startfenster das Jenkins-Template angeklickt werden, im Browser wird ein Projektname eingegeben sowie einige (wenige) Grundeinstellungen vorgenommen und der Jenkins-Server steht in weniger als fünf Minuten zur Verfügung.

Erstellen einer individuellen Projektumgebung

In OpenShift stellen Projekte Umgebungen dar. In einem Projekt existieren mehrere Container, die sich gegenseitig über Aliasse ansprechen können. Von außen besteht aber kein direkter Zugriff. OpenShift bietet

Router, die erlauben, dass ein Server von außen über eine konfigurierbare URL erreichbar ist. In der Umgebung aus *Abbildung 4* gibt es beispielsweise keinen direkten Zugriff auf den MySQL-Container.

Um darzustellen, wie einfach mit OpenShift-Projektumgebungen aufgebaut werden können und dass sich OpenShift auch wie „Cloud“ anfühlt, wird im Folgenden der Aufbau einer einfachen Projektumgebung Schritt für Schritt erläutert. Das Beispielprojekt besteht aus folgenden Bausteinen:

- Datenbank-Server (MySQL)
- Management-Interface für die Datenbank (MyPHP-Admin)
- PHP-Server mit der Anwendung

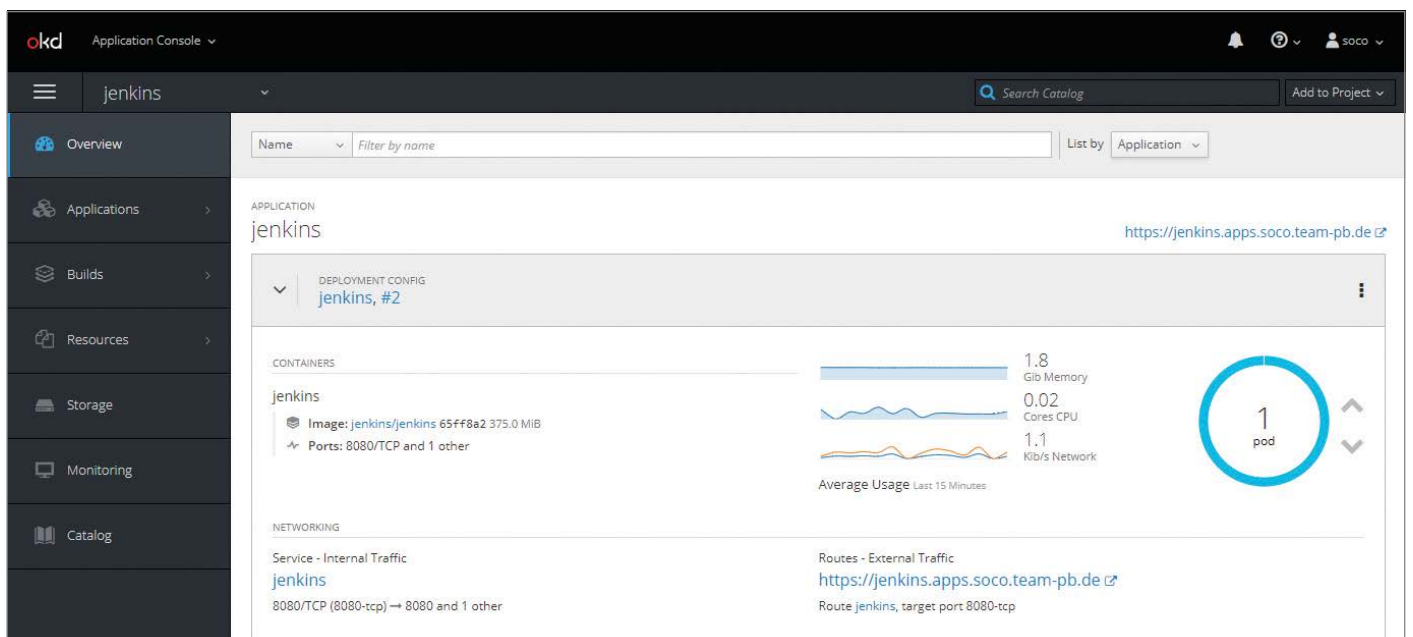


Abbildung 3: Status eines Jenkins-Servers (Quelle: Wolf G. Beckmann)

1. Erstellung eines Projektes

Auf der Startseite wird über den Button „+ Create Projekt“ der Wizard für die Projekterstellung aufgerufen. Hier braucht lediglich der Name des Projektes eingegeben werden. Danach wechselt man in das neu erstellte Projekt.

2. Erstellen der Datenbank

Über den Button oder das rechte „Add to Project“-Menü mit „Browse Catalog“ das Template „MySQL“ auswählen. Die wichtigsten Eingaben sind Connection Username, Password und das Root Password, natürlich können auch Datenbank-Name etc. angepasst werden. Auch hier sind keine weiteren Angaben notwendig.

3. Erstellen einer Management-Oberfläche für die Datenbank

Es soll MyPHP-Admin eingesetzt werden, aber OpenShift bietet dafür kein Template, also wird ein Image aus dem Docker Hub „phpmyadmin/phpmyadmin“ verwendet. Es wird direkt von den Autoren bereitgestellt. Laut der Dokumentation im Docker Hub (<https://hub.docker.com/r/phpmyadmin/phpmyadmin>) wird die Datenbank-Verbindung über folgende Variablen gesetzt: PMA_HOST=mysql (war vorbelegt bei der Anlage der Datenbank), MYSQL_ROOT_PASSWORD=rootpassword (wurde angegeben bei der Anlage der Datenbank).

Da ansonsten der Standard passend ist, werden keine weiteren Parameter gesetzt. Um das Image dem Projekt hinzuzufügen, wählt man im OpenShift-Pro-

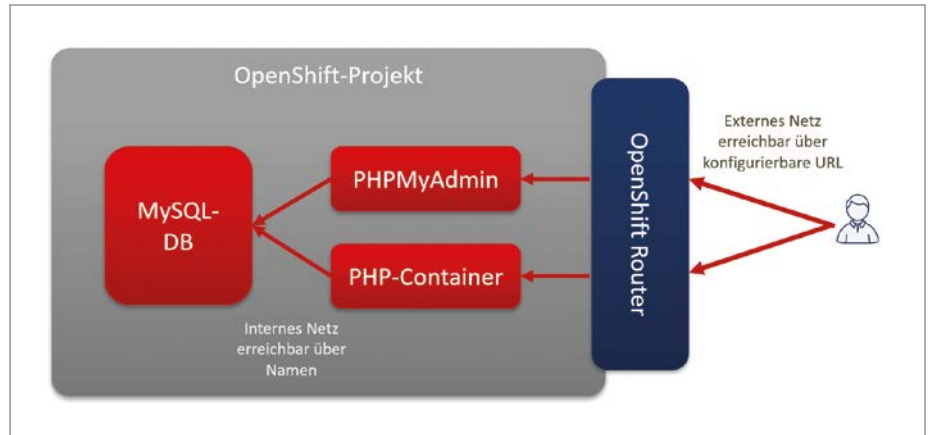


Abbildung 4: Beispielumgebung in OpenShift (Quelle: Wolf G. Beckmann)

jekt über das „Add to Project“-Menü oben rechts „Deploy Image“ und gibt das Image an, wie in *Abbildung 5* zu sehen ist.

Nach dem Anklicken der Lupe können weiter unten in dem Wizard die Variablen eingegeben werden, wie es in *Abbildung 6* zu sehen ist. Über „Deploy“ wird das Image deployt.

Nach kurzer Zeit ist das Deployment fertig. Um von außen auf diese Admin-Oberfläche zugreifen zu können, muss eine Route erstellt werden. Auch hier sind nicht mehr als drei Klicks notwendig, über „Create Route“ wird die Route erstellt. Die primäre Angabe ist die gewünschte URL, unter der der Service erreichbar sein soll. Die URL ist typischerweise eine Subdomain des Servers.

Anschließend kann im Browser nun PHP MyAdmin aufrufen werden. Wenn die Management-Oberfläche nicht mehr benötigt wird, kann die Anzahl der PODs (Container) auf 0 gesetzt werden. Dann existiert lediglich noch die Konfiguration,

die Software selbst läuft aber nicht mehr.

Bis zu diesem Punkt dauert es ca. fünf Minuten.

4. Installieren der PHP-Anwendung

OpenShift bietet hier für eine Vielzahl von Programmiersprachen (Java, JavaScript (node.js), Python, PHP, Django...) vorgefertigte Templates, die den Source einer Anwendung in einer Laufzeitumgebung als Container paketieren. OpenShift nennt die Technologie S2i (Source to Image). Als Beispielanwendung fungiert eine einfache CRUD-Demo in PHP, sie ist zu finden unter <https://github.com/wgbeckmann/php-mysql-crud>. Die Anwendung liest über Umgebungsvariablen des Containers (DBSERVER, DBUSER, DBPASSWORD, DBNAME) die Information zur Datenbank-Verbindung aus.

Somit wird die Anwendung über das PHP-Template installiert. Die Angaben zur Installation sind das GIT-Repository und in

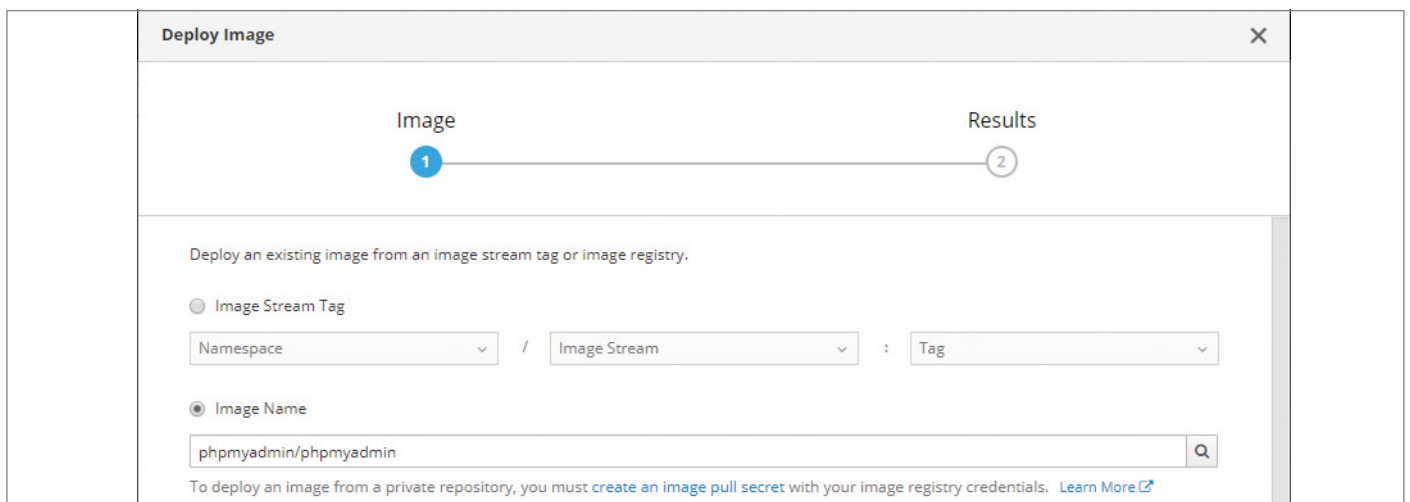


Abbildung 5: Erstellen eines Containers über Image aus Docker Hub (Quelle: Wolf G. Beckmann)

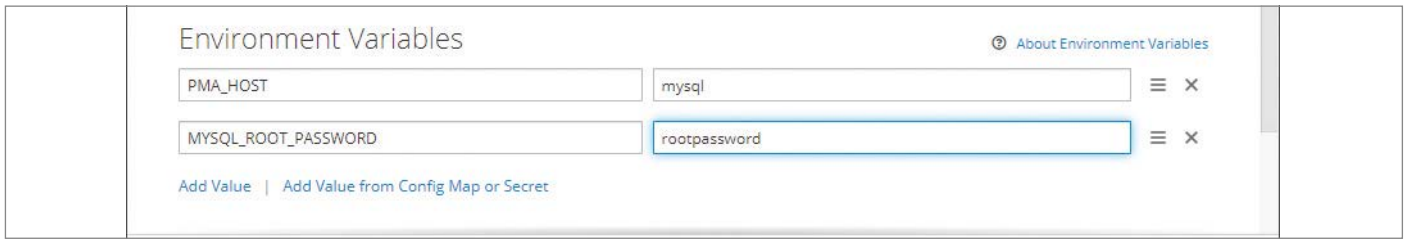


Abbildung 6: Angabe von Umgebungsvariablen für einen Container (Quelle: Wolf G. Beckmann)

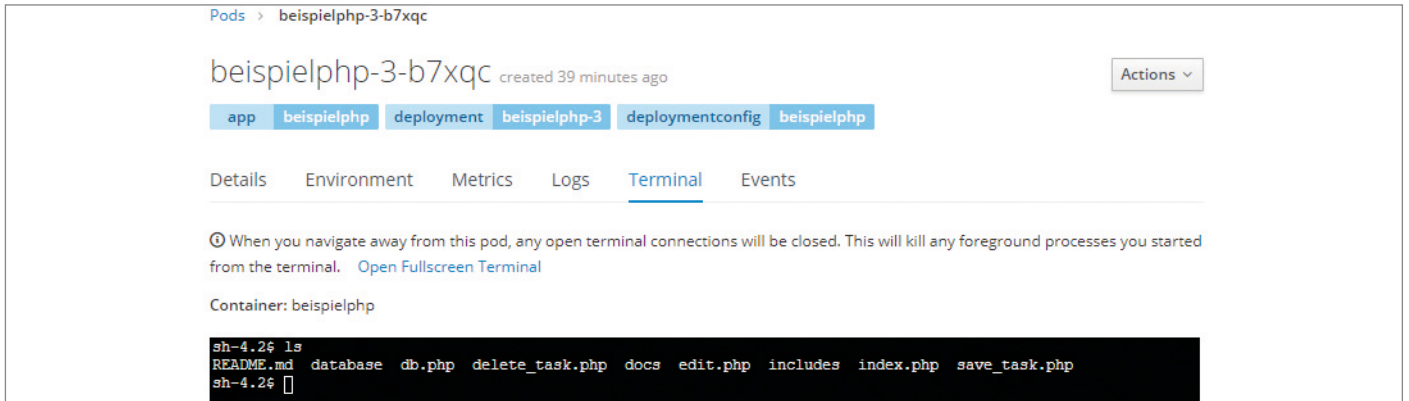


Abbildung 7: Über den Browser kann direkt eine Shell im Container ausgeführt werden. (Quelle: Wolf G. Beckmann)

der erweiterten Konfiguration werden die vier Umgebungsvariablen gesetzt. Danach erstellt OpenShift selbstständig einen passenden Container. In dem Template ist schon das Erstellen der Route vorhanden.

Nachdem über die Management-Oberfläche (PHP MyAdmin) die notwendigen Tabellen in der MySQL-Datenbank erzeugt wurden, kann die Anwendung aufgerufen werden. Alles zusammen dauert weniger als zehn Minuten.

Einbinden beliebiger Applikationen als Container

Wie bei der Installation von PHP MyAdmin gezeigt wurde, können nahezu beliebige Container in OpenShift aus dem Docker Hub installiert werden. Natürlich können auch selbsterstellte Container verwendet werden. Dazu müssen sie lediglich im Repository der OpenShift-Installation deployt werden. So haben wir beispielsweise Container für eine XE-Datenbank mit und ohne Apex erstellt. Zum einfachen Deployment können wiederum OpenShift-Templates erstellt werden.

Für einfache Tests rechnet sich der Aufwand für die Erstellung von Containern nicht immer. Deshalb haben wir einen Container mit einer Oracle-Linux-Installation als Template erstellt. Da OpenShift, wie in *Abbildung 7* zu sehen, den Zugriff auf die

Shell innerhalb des Containers über die Administrationsoberfläche erlaubt, kann dann einfach in dem Container beliebige Software installiert werden.

Fazit

Über den Einsatz von OpenShift hat das Entwicklerteam seine eigene Cloud mit den Vorteilen aus allen Welten bekommen:

- Konsequenter Einsatz von Cloud-Technologien
- Kein administrativer Aufwand (Kostenkalkulation und Beantragung), um Testprojekte aufzusetzen
- Festigung und Erweiterung des Dev-Ops-Wissens im Team
- Einfache Bedienung und Betrieb über eine Web-Oberfläche
- Schnelles Aufsetzen von Umgebungen
- Sehr kostengünstig

Ausblick

Kubernetes ist die typische Betriebsumgebung für Cloud-Native-Applikationen. Somit ergibt es Sinn, bei neuen Applikationen direkt Cloud-Native-Architekturen anzuwenden, selbst wenn sie (noch) nicht in der Cloud betrieben werden.

Über den Autor

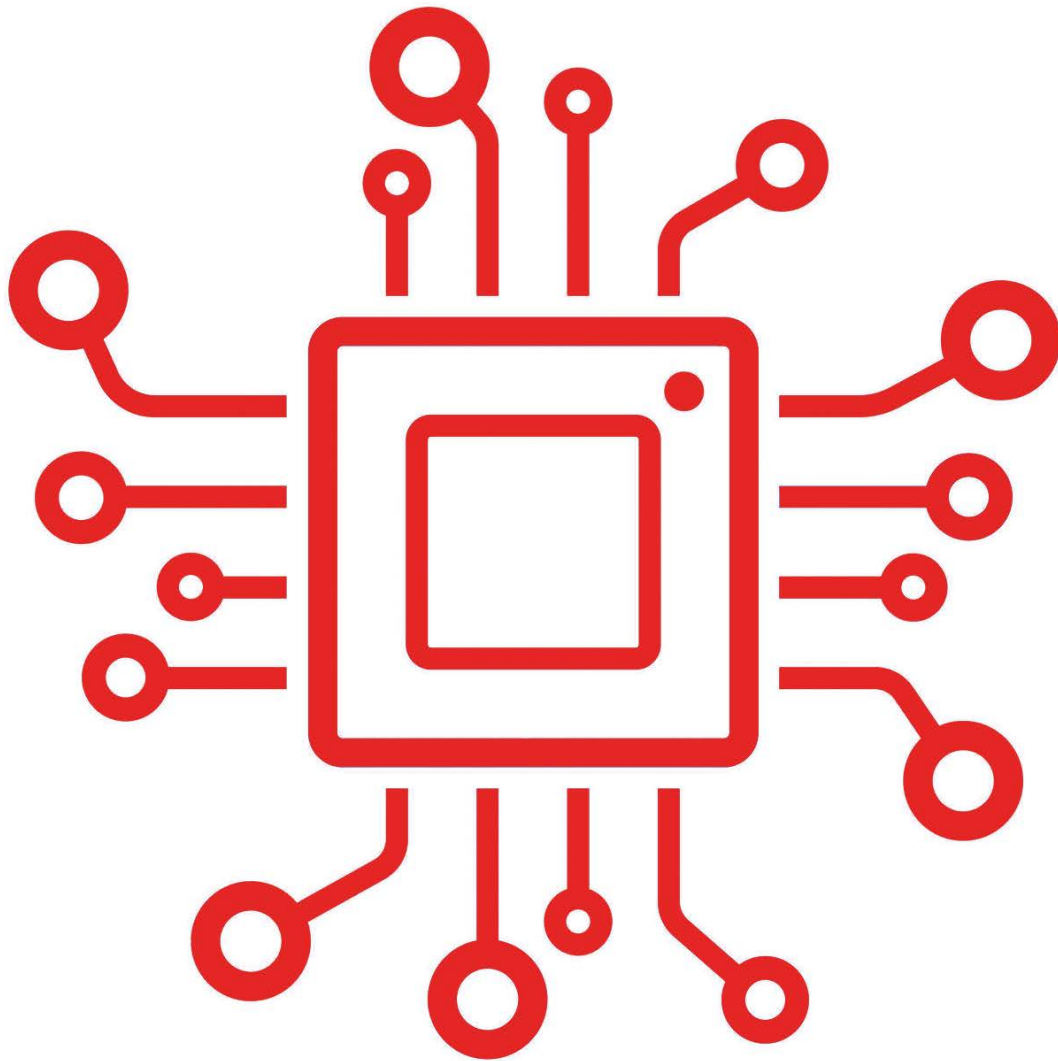
Wolf Beckmann ist Teamleiter des Software- und Consulting-Teams bei der TEAM GmbH und Consultant primär in Migrationsprojekten aus der PL/SQL-Welt in die Java-Welt.

Anmerkung

[1] Der Begriff Team ist in unserem Hause nicht einfach, da einmal von TEAM (unserer Firma) und auch vom Team (dem Software- und Consulting-Team) gesprochen wird. Der Leser möge auf die Schreibweise mit Großbuchstaben achten.



Wolf G. Beckmann
wb@team-pb.de



Cloud-native Microservice-Entwicklung mit Helidon

Sven Bernhardt, Opitz Consulting Deutschland

Schneller, besser und innovativer – das sind Kernattribute, die Unternehmen heute befähigen, langfristig rentabel und konkurrenzfähig zu bleiben. Cloud-Plattformen bilden hierfür eine wichtige Basis, denn die Cloud ist ein wichtiges Vehikel, um innovative Ideen schnell und effizient voranzubringen. Dafür braucht es allerdings zwingend Applikationen, die in der Lage sind, die Cloud richtig zu nutzen und ein Umdenken in der Art und Weise, wie Applikationen implementiert und betrieben werden: Das Zauberwort heißt Cloud-native.

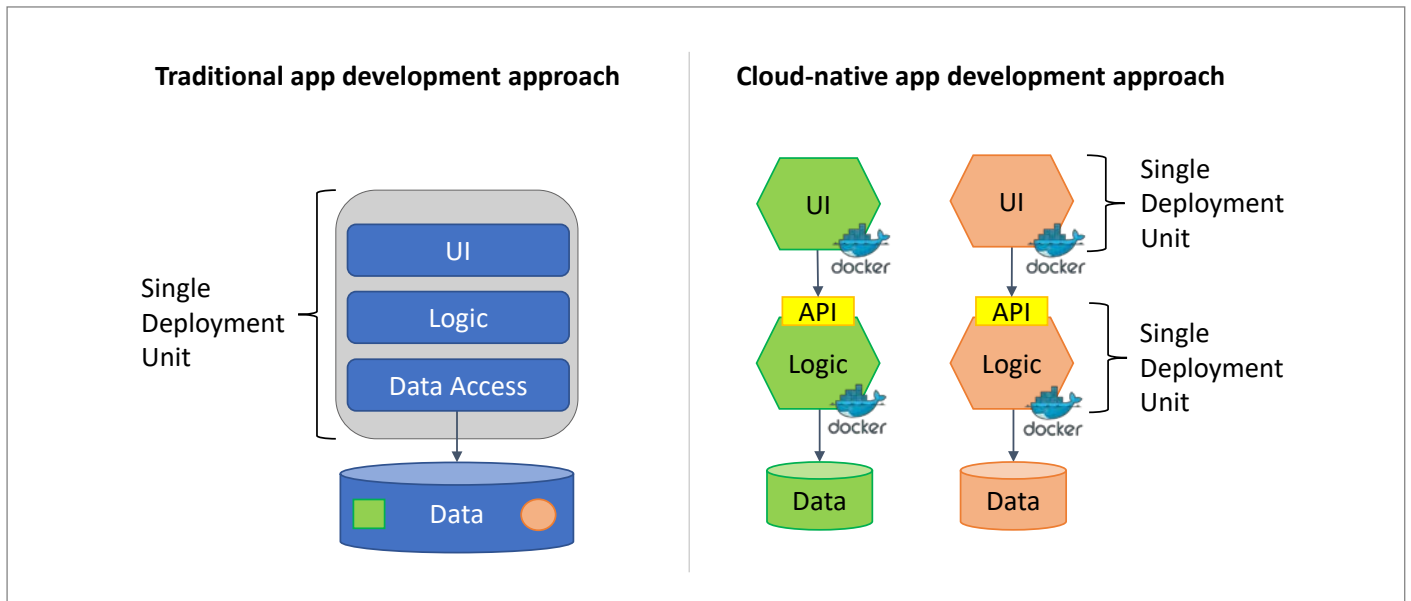


Abbildung 1: Traditionelle vs. Cloud-native Applikationen (Quelle: Sven Bernhardt)

Cloud-native Applikationen sind Applikationen, die von Anfang an mit speziellem Fokus auf die Cloud entwickelt werden. Im Gegensatz zu klassischen On-Premises-Applikationen, die mit „Lift and Shift“ ohne jegliche architektonische Veränderungen in die Cloud gebracht werden, sind Cloud-native Applikationen in der Lage, die besonderen Möglichkeiten der Cloud-Technologie optimal auszuschöpfen, wie zum Beispiel die quasi unbegrenzte Skalierbarkeit. Dafür braucht es eine neue Form der Applikationsarchitektur und -implementierung.

Cloud-native Applikationen

Cloud-native Applikationen sind speziell für die Cloud konzipiert. Aber was bedeutet das eigentlich konkret?

Die Cloud Native Computing Foundation (CNCF) beschreibt Cloud-native Applikationen als

- lose gekoppelt,
- belastbar („resilient“),
- handhabbar („manageable“)
- und beobachtbar („observable“).

Zudem müssen Änderungen an solchen Applikationen schnell und zuverlässig durchgeführt werden können, was durch eine robuste und durchgängige Automatisierungsstrategie zu untermauern ist. [1]

Abbildung 1 zeigt, was die CNCF-Definition in Bezug auf eine Operationalisie-

rung konkret bedeutet. Technologisch setzt Cloud-native massiv auf Containerisierung. Architektonisch stehen vor allem Konzepte wie Microservices, APIs und DevOps im Vordergrund.

Die Implementierung von Cloud-nativen Applikationen erfolgt unter Berücksichtigung der sogenannten Twelve-Factor-App-Prinzipien. Hierbei handelt es sich nicht etwa um neue, bahnbrechende Entwicklungsprinzipien, sondern eher um eine Sammlung von bereits etablierten Entwicklungspraktiken, wie beispielsweise die Verwaltung des Sourcecodes in einem Versionsmanagementsystem oder die Externalisierung von Konfigurationen in Umgebungsvariablen. [2]

Cloud-native Java

Wie bereits erwähnt, werden Cloud-native Applikationen containerisiert betrieben. Dies sollte bereits bei der Entwicklung beachtet werden. Bei einer Java-basierten Anwendung bedeutet das beispielsweise, dass diese „standalone“ in der JVM gestartet und betrieben werden kann (zum Beispiel mittels Kommandozeilen-Befehls `java -jar app.jar`). Das verwendete Frameworks sollte also entsprechende Mechanismen bereitstellen.

Ein prominentes Beispiel hierfür ist Springboot [3], das im Umfeld der Cloud-native Entwicklung bei Java-Entwicklern sehr beliebt ist. Springboot kann auf alles zurückgreifen, was das Spring-Ökosystem

zu bieten hat und kann somit eine Vielzahl verschiedenster Anwendungsszenarien abdecken. Das bedeutet leider Segen und Fluch zugleich: Entwickler, die das Spring-Framework nur wenig kennen, verlieren ob des Umfangs und der Komplexität des Ökosystems schnell den Überblick. Die Lernkurve für Spring-Neulinge ist dementsprechend hoch. Das Schreiben einfacher CRUD-Services kann so zu einer großen Herausforderung werden.

Da aber die Java-Community stets aktiv und kreativ ist, hat sich im Bereich der Microservice-Entwicklung in den letzten Jahren einiges getan. Neben Springboot existieren daher mittlerweile eine Vielzahl leichtgewichtiger Frameworks, die die Entwicklung von Microservices einfach und effizient machen. Abbildung 2 zeigt eine Übersicht der aktuell prominentesten Frameworks in diesem Bereich. Dabei handelt es sich bei der Übersicht um eine Momentaufnahme, die keinen Anspruch auf Vollständigkeit hat.

Neben den sogenannten Full Stack Frameworks zeigt die Abbildung auch MicroProfile-Frameworks sowie Microframeworks.

MicroProfile ist eine Community-getriebene Spezifikation zur Entwicklung von Java-Enterprise-basierten Microservices. Diese Spezifikation wird von der Eclipse Foundation [4] gehostet und umfasst eine Sammlung von Einzelspezifikationen. Dabei bedient sie sich aus existierenden Spezifikationen des klassischen Java EE beziehungsweise Jakarta EE-



Abbildung 2: Cloud-native Java Frameworks, eine Übersicht (Quelle: Sven Bernhardt)

Umfelds. Anders als bei Java Enterprise existiert bei MicroProfile keine Referenzimplementierung. Das macht den Spezifikationsprozess leichtgewichtig und ermöglicht kurze Release-Zyklen.

Umfang und Ökosystem sind bei den MicroProfile-Frameworks bei Weitem nicht so ausgeprägt wie bei Springboot. Dieser Umstand kann für bestimmte Anwendungsfälle Einschränkungen bedeuten, macht sie aber intuitiver und erleichtert Neulingen den Einstieg.

Neben den MicroProfile-Frameworks zeigt *Abbildung 2* auch sogenannte Microframeworks. Microframeworks sind charakterisiert durch eine reaktive, nicht-blockierende („non-blocking“) Architektur. Sie sind optimiert auf schnelle Startup- und Verarbeitungszeiten. Frameworks dieser Kategorie verzichten meist auf jegliche Form impliziter „Framework Magic“ wie Dependency Injection. Damit übernehmen Entwickler dann allerdings selbst die Verantwortung für bestimmte

Abläufe, wie beispielsweise die korrekte Initialisierung von Klassen- und Objektinstanzen oder die Freigabe bestimmter Ressourcen. Abhängig vom Anwendungsszenario erhöhen sich damit einerseits die Menge des Boilerplate Code und der Testaufwand. Auf der anderen Seite sind diese Frameworks durch wenige externe Abhängigkeiten jedoch extrem leichtgewichtig und flexibel.

Helidon Framework-Architektur

Bei Helidon handelt es sich um eine Sammlung von Bibliotheken für die Implementierung Java-basierter Microservices, deren Entwicklung maßgeblich durch Oracle getrieben wird. Das Open Source Framework wurde 2018 in der Version 1.0 released. Interessant dabei ist, dass das Framework, wie in *Abbildung 2* dargestellt, zwei unterschiedliche

Varianten unterstützt: Helidon MP und Helidon SE. Basis für beide Framework-Varianten ist Netty [5], ein asynchrones, Event-getriebenes Framework.

Bei **Helidon SE** handelt es sich um ein leichtgewichtiges, reaktives Microframework, das ein funktionales Programmiermodell unterstützt. Wie in *Abbildung 3* dargestellt beinhaltet das Framework einen reaktiven Webserver sowie Features für ein flexibles Konfigurationsmanagement und Security.

Der reaktive Webserver zeichnet sich durch ein einfaches, funktionales Routing-Modell aus und bietet unter anderem Support für Open Tracing, Metrics sowie Healthchecks. Für die Implementierung von RESTful-Services werden zudem JAX-RS (REST-Service Support) und JSON-P (JSON Parsing) unterstützt. Da Cloud-native Applikationen dem Mantra: „Build once, run anywhere“ aus den Twelve-Factor-App-Prinzipien [1] folgen, wird ein flexibler Konfigurationsmechanismus angeboten, der unterschiedliche Formate wie YAML oder Property-Files und auch dynamische Konfigurationsanpassungen zur Laufzeit unterstützt. Des Weiteren werden Funktionalitäten für die Umsetzung von Security-Anforderungen in den Bereichen Inbound- und Outbound-Authentifizierung, Autorisierung sowie Auditing bereitgestellt.

Bei **Helidon MP** handelt es sich um eine MicroProfile-Implementierung. Wie *Abbildung 3* zeigt, setzt Helidon MP auf die im vorherigen Abschnitt erläuterten Basisbausteine von Helidon SE auf, was eine effiziente Weiterentwicklung der beiden

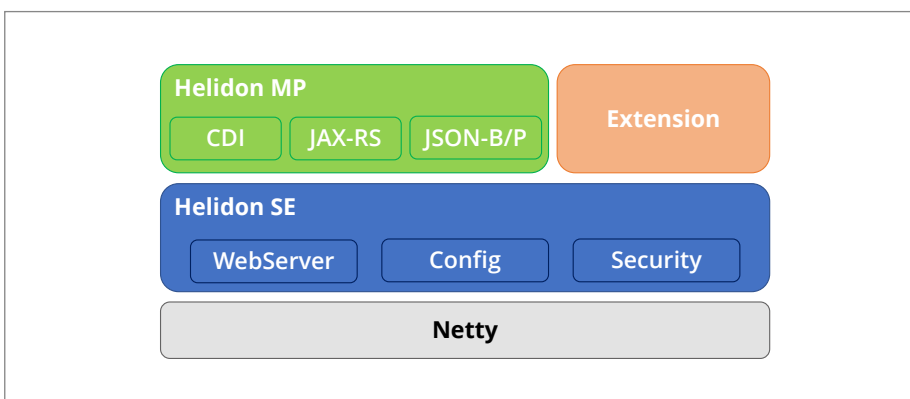


Abbildung 3: Architektur des Helidon Frameworks (Quelle: Sven Bernhardt)

Framework-Varianten ermöglicht. Der MicroProfile-Spezifikation folgend werden die Java EE-Spezifikationen für Context and Dependency Injection (CDI), JAX-RS, JSON-B und JSON-P unterstützt.

Darüber hinaus ist Helidon MP über CDI-Extensions [6] flexibel erweiterbar. Derzeit existieren etwa CDI-Extensions für Java Persistence API (JPA) oder Java Transaction API (JTA). Beide Spezifikationen sind nicht Bestandteil der MicroProfile-Spezifikation, werden aber bei der Entwicklung von datenbankzentrischen Services benötigt; zusätzlich existieren weitere Extensions für den Zugriff auf Oracle Cloud Resources, wie z.B. Storage.

Aus Entwicklersicht ist der Ansatz von Helidon hochspannend: Entwickler können sich je nach Use Case für eine Variante entscheiden. Das Grundframework bleibt unverändert, einzig das Programmiermodell ändert sich. Das macht die Entwicklung effizienter und bietet mehr Flexibilität bei der Umsetzung von Business-Anforderungen.

Quickstart in die Helidon-Entwicklung

Der initiale Start in die Microservice-Entwicklung mit Helidon ist einfach: Für die Erzeugung eines initialen „Hello World“-

```
mvn archetype:generate -DinteractiveMode=false \
-DarchetypeGroupId=io.helidon.archetypes \
-DarchetypeArtifactId=helidon-quickstart-mp \
-DarchetypeVersion=1.4.1 \
-DgroupId=com.opitzconsulting.helidon \
-DartifactId=greeting-service-mp \
-Dpackage=com.opitzconsulting.helidon
```

Listing 1: Projekt-Setup für einen Helidon Service (Helidon MP)

Projekts als Startpunkt für die weitere Entwicklung können entsprechend vorhandene Maven-Archetypen für Helidon SE und Helidon MP verwendet werden.

Der Maven-Aufruf in Listing 1 erzeugt eine Verzeichnisstruktur für einen Helidon MP Service, die neben den üblichen Artefakten wie einer pom.xml und einer initialen Java-Projektstruktur sowie den erforderlichen Konfigurationsdateien (beispielsweise logging.properties) ein Dockerfile sowie ein Kubernetes-Deployment-Manifest enthält. Abbildung 4 zeigt dies sehr detailliert. Analog zu dem in Listing 1 dargestellten Aufruf zur Erzeugung eines Helidon MP-Projekts, wird ein Helidon SE-Projekt unter Verwendung des SE-spezifischen Maven-Archetyps durchgeführt.

Wie Abbildung 4 zeigt, sind die resultierenden Projekte für Helidon SE und MP vom Aufbau her sehr ähnlich. Zur Ausführungszeit funktionieren beide Services sogar exakt gleich: Sie bieten eine

HTTP-Ressource an, die einen „Hello World“-String zurückzuliefert. Die Implementierung hingegen unterscheidet sich aufgrund der Programmiermodelle.

Unterschiedliche Programmiermodelle

Helidon SE unterstützt ein funktionales, Helidon MP ein deklaratives Programmiermodell. Was dies konkret bedeutet, wird im Folgenden anhand von Beispielen erläutert. Der vollständige Code des in diesem Artikel verwendeten Beispiels findet sich in [7].

Helidon-Applikationen haben immer eine zentrale Klasse, die sich für das Applikations-Bootstrapping verantwortlich zeichnet. In der Projektstruktur, die in Abbildung 4 gezeigt wird, ist dies die Klasse Main.java. Hier erfolgen unter anderem die Runtime-Konfiguration und der Start

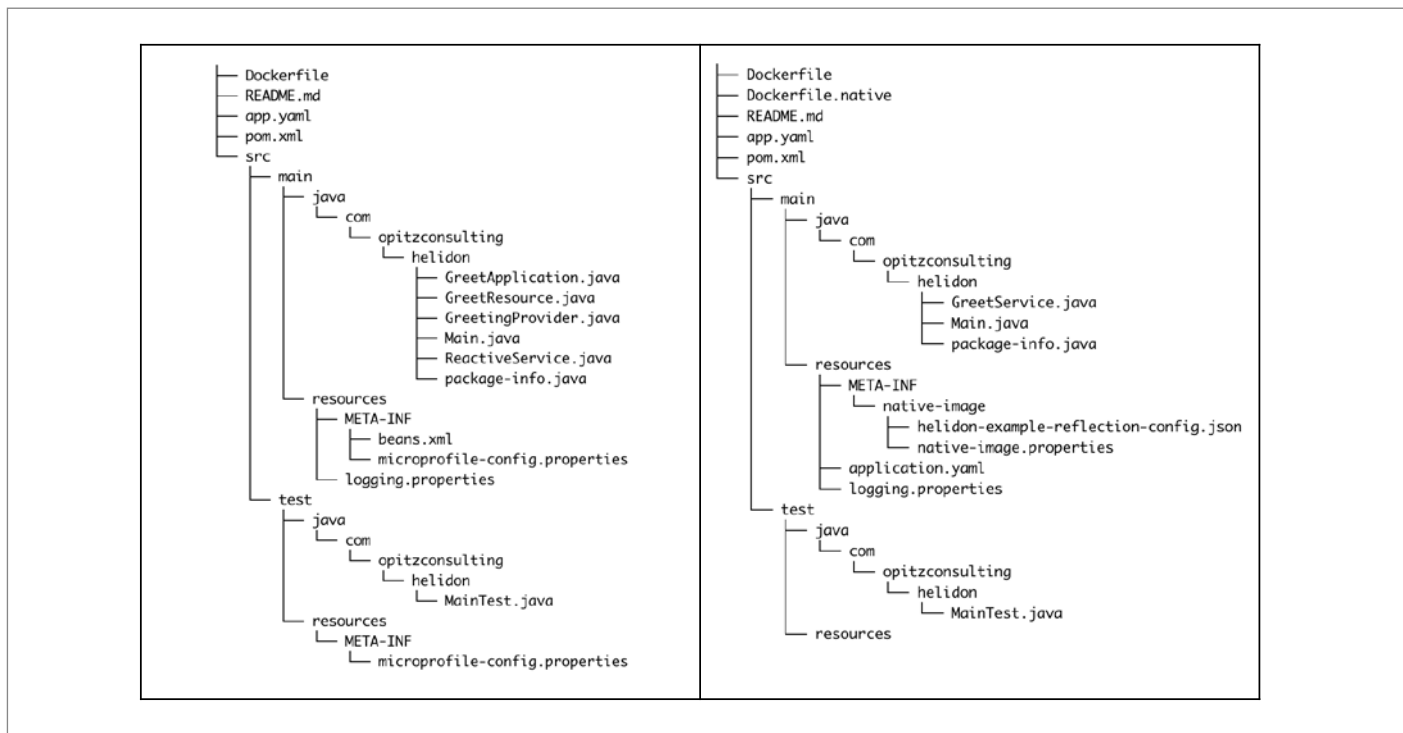


Abbildung 4: Projektstruktur Helidon MP (links) und Helidon SE (rechts) (Quelle: Sven Bernhardt)

```

static WebServer startServer() throws IOException {
    Config config = Config.create();
    ServerConfiguration serverConfig =
        ServerConfiguration.create(config.get("server"));
    WebServer server = WebServer.create(serverConfig, createRouting(config));
    ...
}

private static Routing createRouting(Config config) {

    MetricsSupport metrics = MetricsSupport.create();
    GreetService greetService = new GreetService(config);
    HealthSupport health = HealthSupport.builder()
        .addLiveness(HealthChecks.healthChecks())
        .build();

    return Routing.builder()
        .register(JsonSupport.create())
        .register(health)
        .register(metrics)
        .register("/greet", greetService)
        .build();
}

```

Listing 2: Auszug aus dem Applikations-Bootstrapping einer Helidon SE-Applikation

der Applikation. Für Helidon MP sieht diese Klasse übersichtlich aus, da große Teile des Bootstrapping implizit durch das Framework erledigt werden. Hierzu zählen unter anderem das Laden der Server-Konfiguration aus der Datei `microprofile-config.properties` sowie die Registrierung von Healthcheck- und Metrics-Endpunkten, über welche das Servicemonitoring erfolgt. Da in Helidon SE-Applikationen wie erwähnt weitestgehend auf Framework-Magic verzichtet wird, müssen solche Dinge explizit im Code deklariert werden.

Listing 2 zeigt, wie eine Helidon SE-Applikation konfiguriert wird. Die wichtigste Komponente in diesem Zusammenhang ist die Routing-Komponente, die im Listing 2 in der Methode `createRouting` initialisiert wird. In dieser Methode werden der „Hello World“-Endpunkt sowie die Monitoring-Endpunkte der Applikation initialisiert und entsprechend registriert. Auch die Definition von Request Routings muss explizit ausimplementiert werden. Zu diesem Zweck werden in der Routing-Komponente entsprechende Routing-Regeln und zugehörige Request-Handler definiert, wie Listing 3 zeigt.

Die Registrierung von Routing-Regeln und Request-Handlern erfolgt in der `update`-Methode, die jeder Helidon SE REST-Serviceendpunkt implementieren muss.

In Helidon MP erfolgt die Definition von Request Routings über die aus dem

```

public class GreetService implements Service {
    ...
    @Override
    public void update(Routing.Rules rules) {
        rules
            .get("/", this::getDefaultMessageHandler);
    }

    private void getDefaultMessageHandler(ServerRequest request,
        ServerResponse response) {
        sendResponse(response, "World");
    }
    ...
}

```

Listing 3: Routing-Regeln in Helidon SE

Java EE-Umfeld bekannten JAX-RS-Mechanismen. Hierbei werden die entsprechenden Annotationen (zum Beispiel `@GET`, `@ApplicationPath`, `@Path`) verwendet, um HTTP-Methoden-Mappings, Applikationspfade und anderes zu definieren.

Und es gibt noch weitere Unterschiede: Sollen beispielsweise zusätzliche Framework Features wie Tracing zur Erhöhung der Observability aktiviert werden, muss in einer Helidon SE eine entsprechende Maven-Dependency hinzugefügt werden. Die Verwendung einer Tracing-Komponente, wie sie in Listing 4 gezeigt wird, muss dafür explizit im Code implementiert werden.

In einer Helidon MP-Applikation genügt es, die entsprechende Maven-Dependency für Tracing in der `pom.xml` zu

ergänzen und die Applikations-Konfiguration entsprechend anzupassen.

Wie die Beispiele verdeutlichen, stellt Helidon MP viele Funktionen implizit bereit. Das muss nicht immer von Vorteil sein: Die implizite Bereitstellung kann unter anderem Fehleranalysen erschweren, wenn die Grundannahmen des Frameworks nicht erfüllt werden. Das Umbenennen der Datei `microprofile-config.properties` ist dafür ein gutes Beispiel.

Helidon und GraalVM

Die Java Virtual Machine (JVM) gilt als Standard-Laufzeitumgebung für Helidon-Applikationen. Diese eignet jedoch nicht gleichermaßen für alle Einsatzszenari-

```

static WebServer startServer() throws IOException {
    ...
    final Tracer appTracer = TracerBuilder.create("helidon-se").build();

    ServerConfiguration serverConfig = ServerConfiguration.builder(config.get("server"))
        .tracer(appTracer).build();
    ...
}

```

Listing 4: Open Tracing in Helidon SE-Applikationen

en im Cloud-native-Umfeld. Aus diesem Grund bietet Helidon auch Support für GraalVM an. [8]

Bei GraalVM handelt es sich um eine von Oracle entwickelte hochperformante, flexible Virtual Machine, die als Laufzeitumgebung für verschiedene Non-JVM-Sprachen (C/C++, JavaScript etc.) sowie für JVM-Sprachen (Java, Scala etc.) genutzt werden kann. GraalVM bietet einen Native Image Support, über den mithilfe des GraalVM Ahead-of-Time Compilers ein spezielles, für die Zielplattform kompiliertes Image erzeugt werden kann. Dieses kann schließlich nativ auf der Zielplattform ausgeführt werden. Dieses native Image enthält alle Abhängigkeiten und macht die Installation einer Laufzeitumgebung überflüssig. Im Ergebnis führt dies zu kleineren Container-Images. Auch die Startup-Zeiten einer Applikation können so extrem gesenkt werden, da keine Laufzeitumgebung gestartet werden muss.

GraalVM unterstützt nicht alle JVM Features vollständig. So ergeben sich bei der Verwendung spezieller JVM-Features, wie zum Beispiel Java Reflection, Einschränkungen bei der Benutzung der Virtual Machine (für eine vollständige Auflistung siehe [9]). Aus diesem Grund wird GraalVM aktuell nur für Helidon SE- Applikationen supportet. Bei der Erzeugung eines Helidon SE-Projekts wird, wie in *Abbildung 4* zu sehen ist, auch ein `Dockerfile.native` angelegt, was für den Build eines GraalVM Native-Image-basierten Containers verwendet werden kann.

Fazit

Wie der Artikel zeigt, ist Helidon ein spannendes Framework und eine ernsthafte Alternative zu Springboot, wenn es um die Entwicklung Cloud-nativer Applikationen geht. Vor allem die Unterstützung

der beiden unterschiedlichen Framework-Varianten macht Helidon interessant, eröffnet dies doch mehr Spielraum bei der Lösungsfindung für spezifische Anforderungen.

Allerdings existieren derzeit viele Open Source Frameworks, die einen speziellen Fokus auf die Entwicklung Java-basierter Microservices haben. Stärkste Konkurrenten neben Springboot sind Quarkus und Micronaut. Es darf also mit Spannung beobachtet werden, wie sich Helidon gegen die Konkurrenz behauptet. Das Framework verfolgt eine großartige Vision, die Releases erfolgen sehr regelmäßig und die Community ist aktiv und wachsend. Beste Voraussetzungen also, um die ersten Schritte mit Helidon zu wagen!

Quellen / Zusatzmaterialien

- [1] Quentin Hardy (2019): The problem with „Cloud Native“, <https://bit.ly/36Aph5S>, zuletzt abgerufen am 14.01.2020
- [2] Adam Wiggins (2017): The Twelve-Factor App, <https://12factor.net/>, zuletzt abgerufen am 14.01.2020
- [3] Pivotal Software (2020): Spring Boot 2.2.2, <https://spring.io/projects/spring-boot>, zuletzt abgerufen am 14.01.2020
- [4] Eclipse Foundation (2020): Eclipse MicroProfile, <https://microprofile.io/>, zuletzt abgerufen am 14.01.2020
- [5] Netty Project (2019): Netty Project, <https://netty.io/>, zuletzt abgerufen am 14.01.2020
- [6] Antoine Sabot-Durand, et al.: JSR 365: Contexts and Dependency Injection for Java 2.0, <https://docs.jboss.org/cdi/spec/2.0/cdi-spec.html#spi>, zuletzt abgerufen am 14.01.2020
- [7] Sven Bernhardt Github, <https://github.com/svenbernhardt/helidon-greeting-service>, zuletzt abgerufen am 14.01.2020
- [8] Oracle (2018, 2019): GraalVM, <https://www.graalvm.org/>, zuletzt abgerufen am 14.01.2020
- [9] Oracle GitHub (2019): Native Image Java Limitations, <https://github.com/oracle/graal/blob/master/substratevm/LIMITATIONS.md>, zuletzt abgerufen am 14.01.2020

Über den Autor

Sven Bernhardt arbeitet als Senior Solution Architect für die Opitz Consulting-Deutschland GmbH. Er konzipiert und implementiert zukunftsorientierte, robuste Anwendungen. Sven Bernhardt arbeitet in verschiedenen Projekten, die in den Bereichen Cloud, Microservices und API-Management angesiedelt sind. In seiner Rolle ist er an der Entwicklung und Konzeption von Best Practices in Bezug auf moderne Lösungsarchitekturen (<http://omesa.io>) beteiligt. Weiterhin ist Sven Bernhardt Oracle ACE Director und als Referent auf verschiedenen IT-Konferenzen aktiv.



Sven Bernhardt

sven.bernhardt@opitz-consulting.com



Die eigene ADF-Anwendung auf Wolke 7 bringen

Markus Klenke, TEAM

Viele Unternehmen haben in der Vergangenheit mithilfe von visuellen und deklarativen Oracle Frameworks geschäftskritische Anwendungen gebaut, welche über viele Jahre weiterentwickelt wurden. Parallel haben sich Leitarchitekturen und Technologien allerdings ebenfalls weiterentwickelt oder betreffen Konzepte, die zu Beginn der Entwicklung der Geschäftsanwendungen noch gar nicht im Sichtfeld der Entwicklung lagen. Insbesondere die Anwendungsentwicklung und -Bereitstellung in der Cloud nimmt heutzutage immer höhere Stellenwerte ein. Und obwohl das Oracle Application Development Framework zu Zeiten weit vor dem Cloudhype konzipiert wurde, stellen viele Komponenten eine sehr gute Grundlage dar, um die eigene Anwendung „Cloud-ready“ zu machen.

Die grundlegende Idee des Application Development Frameworks (ADF) von Oracle ist, dass mithilfe eines Frameworks der gesamte Stack einer Webanwendung mit Datenzugriff im MVC Pattern via Java realisiert werden kann, ohne dass sich Entwickler mit einer Vielzahl von einzelnen Tools, Libraries oder verschiedenen Sprachen auseinandersetzen müssen. Diese Anwendung wird dann als Fat-WAR / EAR auf einem Java EE-Server bereitgestellt. Innerhalb der

Anwendung wird allerdings schon zur Designzeit vollständig auf abstrakte Interfaces und nur schwache Klassenreferenzen gesetzt (vollqualifizierte Namen), was zwar den Einstieg in die Entwicklung etwas verkompliziert, dem erfahrenen Entwickler allerdings ein enormes Flexibilitätskonzept bietet.

Wie in *Abbildung 1* zu sehen, bietet ADF eine Implementierung für jede der Schichten in der MVC-Architektur, es ist allerdings auch möglich, mit ADF eine An-

wendung zu bauen, welche nur in den datenverarbeitenden Schichten ADF-Komponenten verwendet und die visuellen Schichten durch alternative Implementierungen bereitstellt. Genau diese Flexibilität der Schichten innerhalb der Applikation erlaubt es uns, die Anwendungen auf unterschiedliche Arten und Weisen effizient in die Cloud zu bringen. Dieser Artikel zeigt mehrere Lösungswege auf, um unterschiedlichen Anforderungen an den Übergang in die Cloud gerecht zu werden.

Oracle Java Cloud-Service – Die „native“ Lösung

Oracle bietet mit dem Java Cloud-Service eine Möglichkeit, einen vollwertigen Weblogic-Server in der Cloud zu betreiben – also den kanonischen Weg, die ADF-Anwendung als Java EE-Applikation in die Cloud zu bringen. Während der Aufwand eines Deployments in die Cloud über diesen Service nahezu keine Komplexität mit sich bringt, bietet dieser Ansatz allerdings auch nur wenige native Vorteile, wie die Cloud sinnvoll eingesetzt werden kann, da sowohl der Weblogic-Server als auch die Full-Stack-Anwendung allein durch ihre Größe bedingt meist unpraktisch sind (Beispiel: Startgeschwindigkeit eines Docker-Containers mit Oracle Fusion Middleware). Hier obliegen Themen wie Ausfallsicherheit und Elastizität eher dem Cluster-Konzept des Weblogic-Servers (beziehungsweise dem Oracle JCS) als der Topologie selbst.

Dennoch ist dieser Ansatz ein sehr valider, insbesondere wenn die Cloud-Strategie ist, die selfhosted Server abzuschaffen. Die Administratoren der Weblogic-Domäne finden sich in ihrem üblichen Metier sofort zurecht und müssen nicht in neue Strukturen angelernt, für den Build/Deployment-Prozess müssen nur Hostnamen angepasst werden; die Anwendung ist also grundsätzlich Cloud-fähig. Um einen wirklichen Mehrwert der Cloud zu erhalten, bietet es sich allerdings an, dennoch Restrukturierungen der Anwendung durchzuführen. Ein möglicher Ansatz der Umformung der Applikation kann durch das Konzept der Remote Regions [2] erreicht werden. Hiermit kann beispielsweise eine große monolithische Anwendung in viele kleine für sich selbst lauffähige Anwendungsteile zerlegt werden, ohne die Wiederverwendung durch bekannte ADF-Strukturen zu verlieren (siehe Abbildung 2). Regions werden nicht mehr durch zur Design-Zeit festgelegte interne Library-Referenzen gefüllt, sondern über ein spezielles Java-Servlet, welches die Inhalte der Region und dessen Sicherheitskonzept über ein Servlet einer anderen ADF-Anwendung lädt. Das ermöglicht deutlich einfachere Ansätze für Ausfallsicherheit, Skalierbarkeit und Austauschbarkeit.

Zur Aktivierung der Remote-Region-Funktionalität müssen die jeweils einzel-

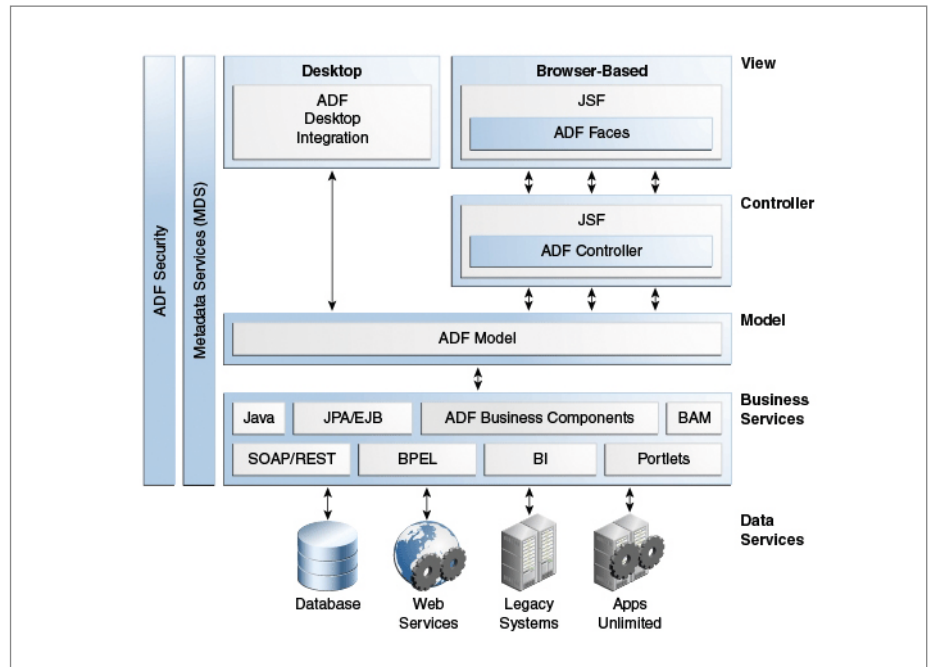


Abbildung 1: ADF Architektur (Quelle © Oracle, [1])

nen Projekte nur als Remote Region Producer resp. Consumer definiert werden, wodurch die entsprechenden Servlets in die web.xml eingetragen werden (siehe Listing 1).

Zusätzlich müssen noch einige Parameter am Task-Flow selbst eingestellt werden, beispielsweise das Transaktionsmanagement.

Das Feature Remote-Regions benötigt allerdings eine Single Sign On-Lösung für die Applikationen und ist zusätzlich in der kostenfreien Variante von ADF (ADF Essentials) nicht enthalten ist. Daher kann dieses Feature nur auf einem Oracle Weblogic-Server verwendet werden, was wiederum nicht notwendigerweise gewollt sein muss, da man sich so an eine fixe Instanz eines Java EE-Servers bindet.

Separation der Schichten – ADF BC Rest Services

Wie in Abbildung 1 gezeigt, sollte jede ADF-Webanwendung nach dem MVC Pattern entwickelt worden sein. Es besteht durch die Trennung dieser Schichten und dem ADF Model Layer als Abstraktionsschicht der Daten ohnehin schon nur eine schwache Abhängigkeit zwischen der Daten-/Businessschicht und der „Frontend“-Schicht. Nimmt man nun noch die Tatsache hinzu, dass nach Best-Practice der Großteil der fachlichen Komplexität und Business-Logik in der Business Service-Schicht implementiert sein sollte, so befindet sich der Kern der meisten ADF-Anwendungen in den Business Components. Für diese besitzt

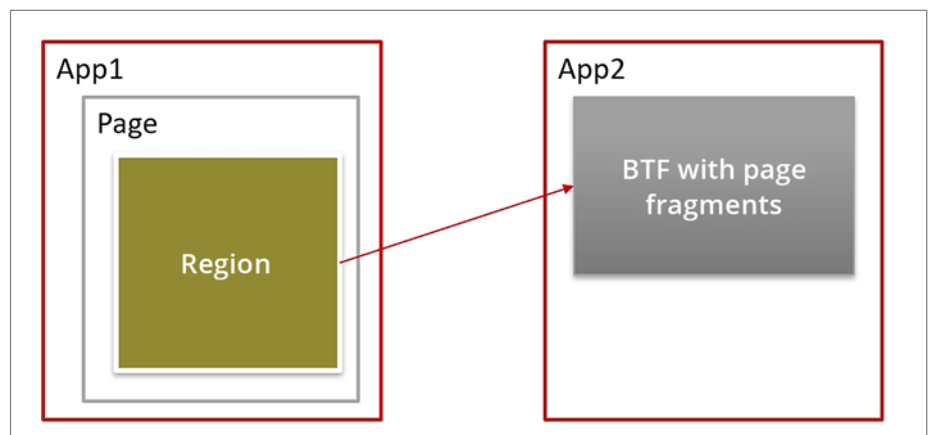


Abbildung 2: Schematische Darstellung des Remote-Region-Konzeptes (Quelle: Markus Klenke)

```

<servlet>
  <servlet-name>rtfqueryServlet</servlet-name>
  <servlet-class>oracle.adfinternal.controller.rtfquery.RemoteTaskFlowQueryServlet</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
  <servlet-name>Faces Servlet</servlet-name>
  <url-pattern>/rr/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>rtfqueryServlet</servlet-name>
  <url-pattern>/rtfquery</url-pattern>
</servlet-mapping>

```

Listing 1: Servlet Properties für den Remote Region Producer

Oracle eine deklarative Möglichkeit, diese Services als REST- (oder SOAP) Schnittstelle nach außen verfügbar zu machen, wodurch nicht nur Datensätze, sondern auch Metadaten wie mehrsprachige Labels, Auswahllisten, Validatoren und selbst fachliche Funktionen propagiert werden können. Wird diese Schnittstelle genutzt, so können beliebige Frontend Frameworks (sofern sie Webservice-fähig sind) mit diesen angereicherten Informationen beladen werden.

In *Abbildung 3* ist schematisch zu sehen, dass statt der JSF-Architektur für das Frontend ebenso eine JavaScript-basierte Architektur (hier Oracle JET) gewählt werden kann, um anwenderspezifische Frontend-Anforderungen an die Geschäftsprozesse abzubilden. Weiterhin kann nach diesem Muster die Anwendung

selbst nach und nach von einem Monolithen in separate Anwendungsbereiche (oder auch nur Funktionen) aufgespalten werden.

Ein valider Ansatz ist hier, die bestehende Anwendung so umzubauen, dass jedes Application Module dupliziert und entsprechend eines der beiden auf die Nutzung für REST Webservices restrukturiert wird. Insbesondere folgende Punkte sind dabei zu beachten:

- Alle Operationen auf Application Module-Ebene (Client Interface des Application Modules) sollten so umgebaut werden, dass sie innerhalb des Client Interfaces eines View-Objekts definiert sind, damit sie über eine REST-Ressource zugreifbar gemacht werden können.

- Methoden auf View-Objekten im Client Interface sind per Definition immer über das REST-Interface erreichbar. Sollen Methoden nicht via REST verfügbar gemacht werden, so muss dies explizit gesetzt werden (*siehe Listing 2*).
- Zur Verwendung von ViewAccessors sollten diese ebenfalls im DataModel des Application Modules sowie als ReadOnly Ressource zur Verfügung gestellt werden. So können deklarativ individuelle Voreinstellungen bezüglich der Datenakquise erstellt werden.

Ist die Migration des Modules abgeschlossen, kann die Umstrukturierung der Anwendung beginnen. Nach und nach werden nun View Controller-Projekte / Page Definitions so umgestellt, dass diese nicht mehr wie üblich auf den ADF BC Data Con-

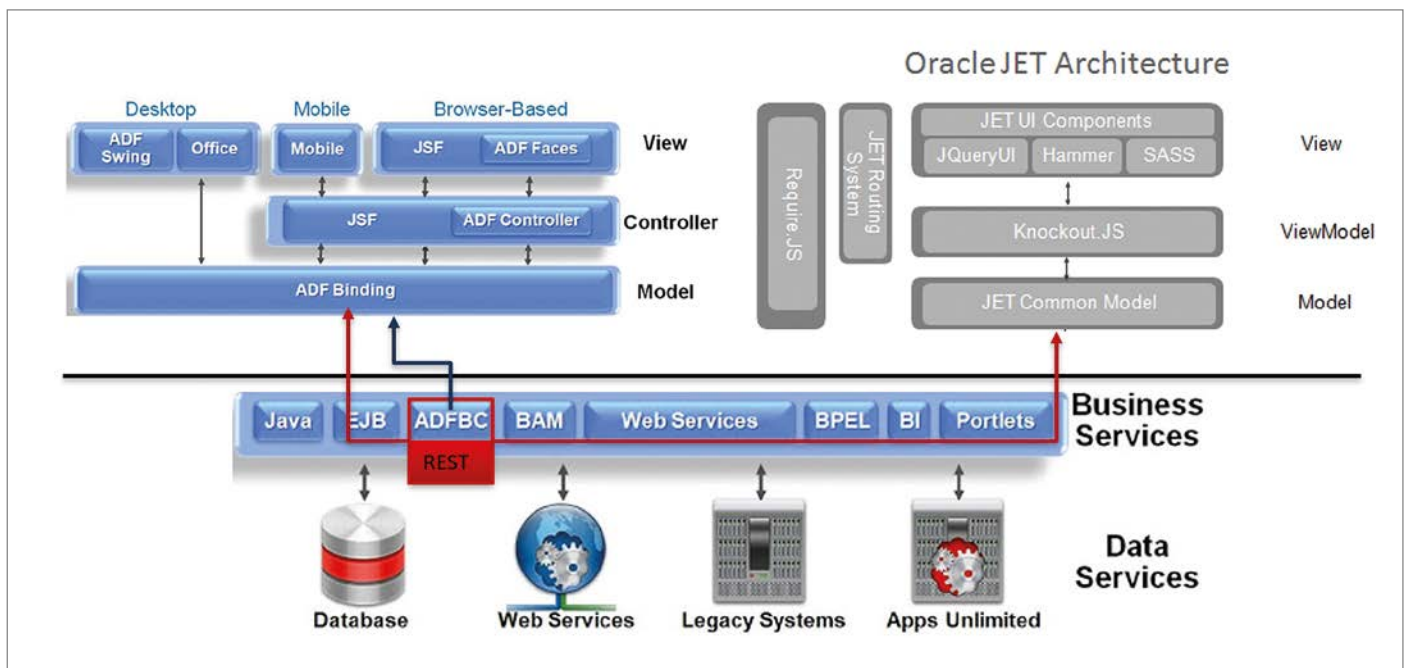


Abbildung 3: ADF Business Components REST Service-Architektur (Quelle: © Oracle)

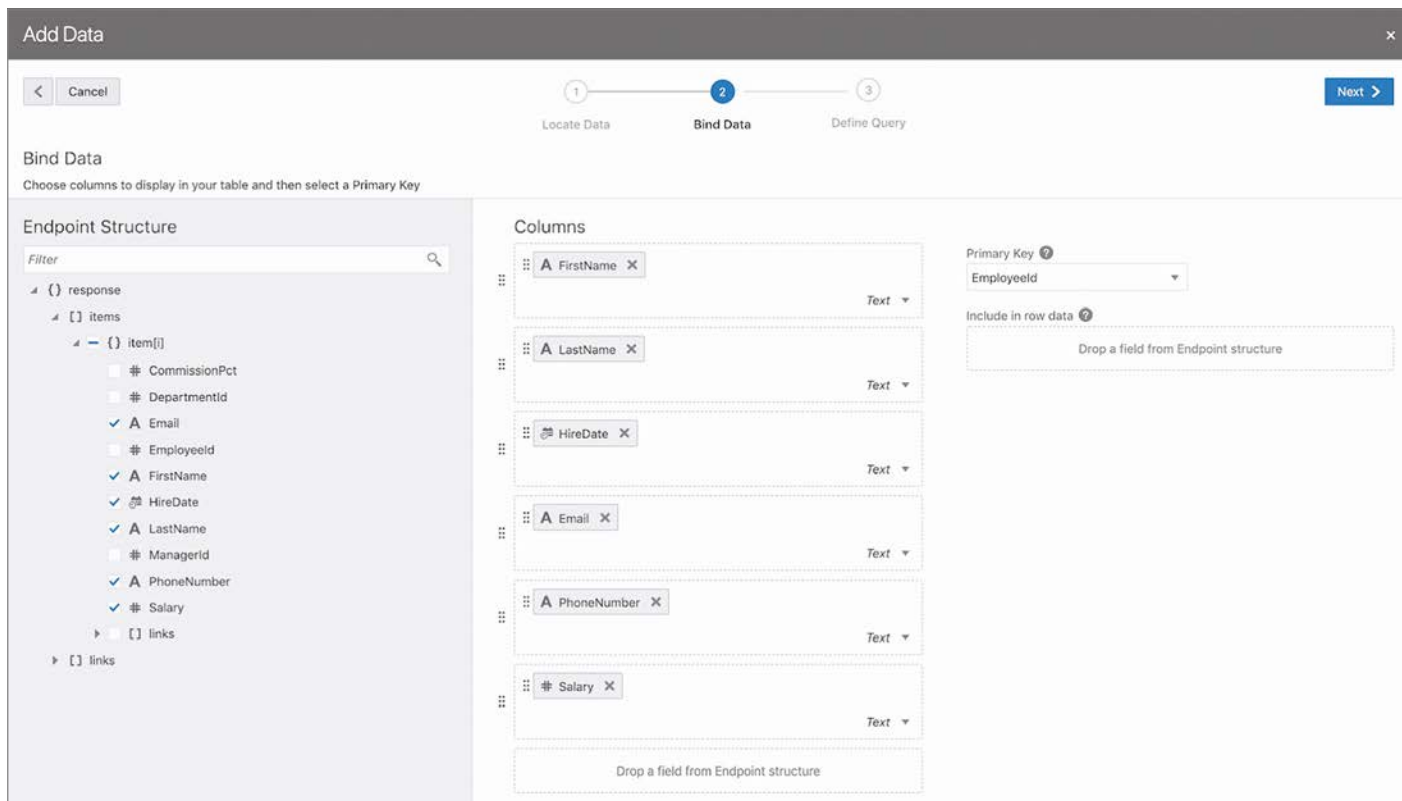


Abbildung 4: Visual Builder Data Wizard auf ADF BC Rest (Quelle: © Oracle)

zugreifen, sondern via ADF REST Data Control auf die neuen Module, welche separat auf dem Server als REST Web Application deployt werden. Auf Grund der Abstraktionsschichten und der angereicherten REST Responses ist dieser Schritt sowohl für den Endanwender kaum ersichtlich (Insbesondere, wenn Fetch-Sizes entsprechend angepasst werden) als auch der Implementierungsaufwand für die Entwickler minimal. Wichtig ist hier natürlich auch wieder, dass bei der Implementierung der Java Beans darauf geachtet wird, dass nicht an den Interfaces vorbeiprogrammiert wurde (Beispiel: Direktzugriff auf oracle.jbo extended-Klassen im Controller), da sonst die Komplexität einer Umstellung um ein Vielfaches erhöht wird.

Oracle VBCS – ADF-like JavaScript Development

Eine der größten Herausforderungen für die Frontentwicklung mit ADF ist die serverbasierte Architektur von JSF. Diese bietet durch den komponentenbasierten Fokus zwar enorm viele Vorteile in Sachen Uniformisierung, Abstraktion und Komplexitätsreduktion, hat durch

```
<Method Name="hello">
  <Return
    Type="java.lang.String"/>
  <Properties>
    <CustomProperties>
      <Property
        Name="PAYLOADHINT"
        Value="Hide"/>
    </CustomProperties>
  </Properties>
</Method>
```

Listing 2: Explizites Deaktivieren einer Methode des Client Interfaces für REST

die Querkommunikation zwischen Client und Server aber durchaus Schwierigkeiten in puncto Geschwindigkeit und User Experience, wenn es um Vergleiche mit Javascript Frameworks geht. Der Fokus auf Client-Side Web Applications und Progressive Web Apps erfordert von den Entwicklern fast schon eine Umstellung auf JavaScript. Die Einarbeitung in eine neue, sehr technische und Code-affine Programmiersprache stellt allerdings für viele Entwickler mit 4GL-Hintergrund eine große Herausforderung dar, welche im Tagesgeschäft häufig nicht gemeistert werden kann.

Glücklicherweise bietet Oracle mit dem Oracle Visual Builder Cloud-Service

eine Möglichkeit, den Übergang von Java zur JavaScript-Entwicklung für ADF-Entwickler so einfach wie möglich zu machen. VBCS ist eine deklarative Entwicklungsumgebung, in der Oracle JET-Anwendungen nicht nur via WYSIWIG- Editor visuell gebaut werden können, sondern insbesondere der JavaScript Lifecycle deklarativ erstellt werden kann (ähnlich wie TaskFlows in ADF für Java). Somit wird es für viele Entwickler deutlich leichter, sich mit Themen wie Callbacks, dynamischen Seitenwechseln und Fehlerbehandlung auseinanderzusetzen. Da es sich beim VBCS grundsätzlich aber auch um einen Service zur Entwicklung von Frontend handelt, benötigt man noch einiges an Arbeit, um die Datenaufbereitung durchzuführen - es sei denn, der datengebende Service ist ein ADF REST-Service, dann werden vom VBCS zusätzlich die Metadaten des REST-Service interpretiert. Insbesondere Labels und Auswahllisten können so direkt verwendet werden, ohne diese mit großem Aufwand neu definieren zu müssen. (Wizard gesteuert, siehe Abbildung 4)

Sind die ADF REST-Services entsprechend gut vorbereitet, können einfache Oberflächen oder spezielle Services so sehr einfach als JavaScript-Anwendung bereitgestellt werden. Über diesen Weg

können Teile der ADF-Anwendung beispielsweise sehr effizient ohne großen Aufwand als PWA für mobile Endgeräte entwickelt werden oder Teile der Anwendung je nach Kontext sehr effizient skaliert werden, um beispielsweise die Performance der Anwendung zu verbessern.

Die Zukunft von ADF ist heiter in den Wolken

Die beiden beschriebenen Ansätze können fast auf jede ADF-Anwendung, welche nach bekannten ADF-Architekturmustern umgesetzt ist, angewandt werden. Ob nun eine Anwendung nach Pillar-Architektur oder eine monolithische Anwendung; je nach Anwendungsfall ergeben sich unterschiedlichste Begründungen, überhaupt in die Cloud zu gehen. Den größten Vorteil erreicht man allerdings, wenn die Applikation nicht nur in die Cloud migriert, sondern diese mit sinnvollen Anpassungen so umstrukturiert wird, dass

sie eine möglichst flexible Anbindung an andere Systeme erlaubt. Der Kern der Anwendung mit seiner Geschäftslogik bleibt weitestgehend unverändert, Schnittstellen für Benutzerinteraktion oder Einbindung in Service- Prozesse werden allerdings deutlich vereinfacht.

Somit bietet ADF (eventuell mit Unterstützung durch Oracle VBCS) auch 2020 noch effizienteste deklarative Datenbank-anwendungsentwicklung mit modernsten Techniken, und als Entwickler bleibt man somit für die Zukunft auch ohne große Umschulungen bestens für Web-Anwendungsentwicklung gerüstet.

Quellen

- [1] Oracle ADF Documentation Library : <https://docs.oracle.com/en/middleware/developer-tools/adf/12.2.1.4/concepts/overview-oracle-adf.html#GUID-D3AFB48B-94D7-471D-A3A9-B7DC9E09C2E1>
- [2] Oracle ADF Remote Regions : <https://docs.oracle.com/middleware/1221/adf/develop/GUID-D7FDF023-1373-47CB-B349-DC24826441B0.htm#ADFFD54447>

Über den Autor

Markus Klenke unterstützt als Oracle Consultant unterschiedliche Kunden bei der Entwicklung von Business-Software und der Erstellung und Analyse von Softwarearchitektur. Dabei liegt der Fokus auf modellgetriebenen Architekturen und der Entwicklung sowie der Oracle Fusion Middleware, wobei insbesondere die deklarativen Frameworks Oracle ADF und Oracle VBCS hier die zentralen Interessensgebiete darstellen.



Markus Klenke
mke@team-pb.de



Das E-3 Magazin

Information und Bildungsarbeit von und für die SAP-Community

Wir leben alle unter dem gleichen Himmel, aber wir haben nicht alle den gleichen Horizont.

Konrad Adenauer

Meinung der
Community

Szene

Coverstory

Management

Infrastruktur

Lizenzen

E-3 – die Wissensplattform
für die SAP-Community

Wirtschaft

Human Resources

Community Info



Oracle Private Cloud Appliance – Einfach ein bisschen Hardware mit Oracle VM?

Martin Bracher, Trivadis

Die Swisscom (größter Schweizer Telecom-Provider) baut gerade zusammen mit Trivadis (größter Schweizer Oracle-Dienstleister) ein Oracle-Cloud-Angebot für Firmenkunden auf. Die Highend-Variante läuft auf Oracle Exadata, die Regular-Variante auf der Oracle Private Cloud Appliance (PCA). Wie ich gemerkt habe, kennen viele DBAs die Exadata, nicht aber die PCA. Deshalb möchte ich in diesem Beitrag nun die PCA etwas genauer vorstellen, auch hinsichtlich der Lizenzierung. Im zweiten Teil werde ich erläutern, wie wir in dem Projekt mithilfe von Ansible automatisiert virtuelle Maschinen für die Kunden auf der PCA erstellen.

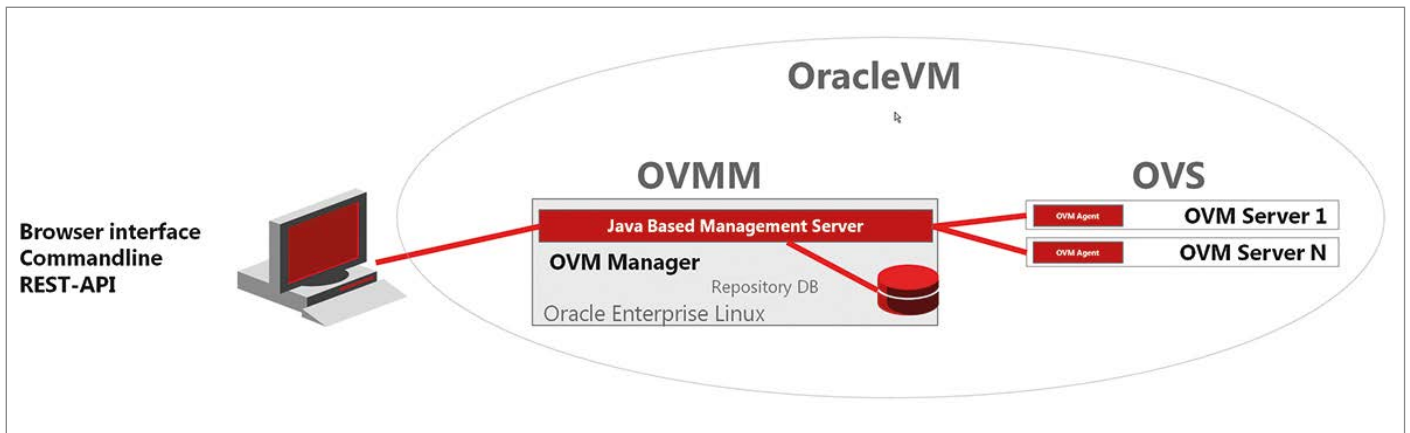


Abbildung 1: Architektur von Oracle VM (Quelle: Oracle)

Oracle VM

Oracle VM (OVM) ist eine Virtualisierungslösung von Oracle, basierend auf dem Oracle Linux (welches wiederum auf Redhat Enterprise Linux basiert). Als Hypervisor zur Virtualisierung wird XEN eingesetzt. OVM besteht aus zwei Komponenten:

- dem OVM Manager, einer Weblogic-Applikation, auf welche man per Browser, Commandline Interface oder einem REST-API zugreifen kann. Der Manager wird auf einem bestehenden RedHat/Oracle Linux installiert.
- den 1..n OVM-Servern. Es handelt sich um Server mit x86 CPUs, auf welchen die virtuellen Maschinen betrieben werden. Auf den Servern wird ein speziell angepasstes Oracle Linux installiert.

Sämtliche Arbeiten werden über den OVM Manager erledigt. Also die Konfiguration und Verwaltung der OVM-Server, als auch die Erstellung und Verwaltung der virtuellen Maschinen wird vom Manager aus gesteuert (siehe Abbildung 1).

Private Cloud Appliance (PCA)

Die Private Cloud Appliance ist ein skalierbares Engineered System von Oracle zum Betrieb von virtuellen Maschinen. Für die Virtualisierung wird OVM verwendet. Oracle plant in seiner Virtualisierungsstrategie jedoch den Wechsel auf KVM. Auf das Design der PCA wird dies kaum große Auswirkungen haben. Oracle wird zukünftig wohl einfach den Virtualisierungslayer austauschen; KVM

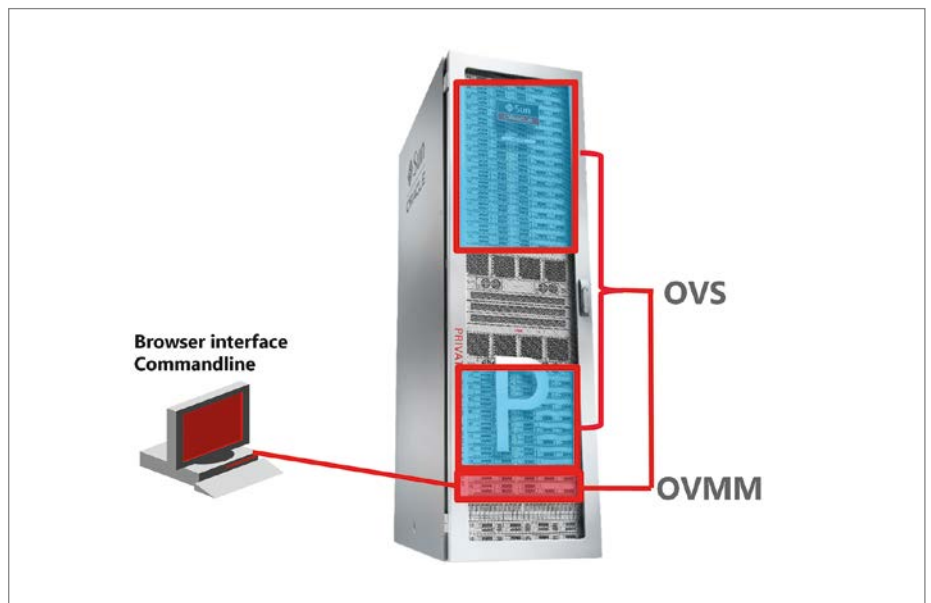


Abbildung 2: Oracle VM auf der PCA (Quelle: Oracle)

wird bereits seit einiger Zeit auf der ODA eingesetzt, und neu auch auf der Exadata X8-M2.

Delivered wird ein Rack mit Oracle SUN-Hardware, welche mit OVM vorkonfiguriert ist (siehe Abbildung 2).

Die Konfiguration ist erhältlich mit 2 - 25 Compute-Nodes (den OVM-Servern). Wem das nicht genügt, der kann auch noch bis zu zwei Expansion Racks mit je bis zu 30 Compute Nodes hinzufügen, also insgesamt bis zu 85 Compute-Nodes mit 3740 Cores (siehe Abbildung 3).

Nun werfen wir einen etwas genaueren Blick auf die einzelnen Komponenten der PCA. Zu beachten ist, dass wir in unserem Projekt noch die etwas ältere Version der PCA verwenden. Diese hat noch einen viel kleineren internen Storage mit 15 statt 100TB, sowie 40Gbit Infiniband statt neu 100Gbit Ethernet.

Management Server

Die PCA enthält zwei Nodes für die Administration, welche aktiv/passiv konfiguriert sind. Den aktiven Node erreicht man über eine virtuelle IP. Darauf läuft der PCA Controller, welcher die initiale Konfiguration der Umgebung übernimmt. Als Web-GUI steht uns das PCA Dashboard (Weblogic-Anwendung) zur Verfügung.

Auf dem aktiven Server läuft auch der originale, unveränderte OVM Manager (siehe Abbildung 4). Somit stimmt der Titel „Einfach ein bisschen Hardware mit Oracle VM“. Gegenüber einem Nicht-Engineered System entfällt jedoch die Grundkonfiguration der OVM-Server, da dies schon durch den PCA Controller gemacht wird.

Ebenso steht ein Enterprise Manager auf diesen beiden Nodes zur Verfügung.

ZFS Storage Appliance

Als interner Shared Storage steht eine ZFS Storage Appliance ZS7-2 zur Verfügung (siehe Abbildung 5). Sie besteht aus zwei Controller Servern und bietet 15TB nutzbaren Platz, wovon initial 3TB zur PCA präsentiert sind. Der Rest kann frei genutzt werden. Für das Storage-Netz werden die 2x40Gbit Infiniband verwendet.

Netzwerk

Es kommt ein sogenanntes Software Defined Network zum Einsatz (siehe Abbildung 6). Folgende Netzwerke stehen zur Verfügung:

- Internal Storage Network (Infiniband 40Gbit)
- Internal Management Network: Wird unter anderem für die Kommunikation des OVM Managers mit den Compute Nodes verwendet
- Internal Underlay Network: Datenverkehr zwischen den Compute Nodes
- External Underlay Network: Datenverkehr zwischen der PCA und dem Datacenter

Diese Netzwerke werden durch den PCA Controller konfiguriert und nicht durch den OVM Manager. Zusätzliche VLANs auf den Underlay-Netzwerken werden jedoch durch den OVM Manager erstellt.



Abbildung 3: Erweiterungsmöglichkeiten der PCA (Quelle: Oracle)

Lizenzierung

Oracle lizenziert man üblicherweise nach Prozessoren. Bei der x86-Plattform (Intel, AMD) benötigt man bei der Enterprise Edition für 2 Cores eine Lizenz. Was ist nun aber bei OVM eine „CPU“?

Im Nicht-Hyperthreading-Modus ist eine CPU ein Core (siehe Abbildung 7). Im Hyperthreading-Modus wird bei x86-Prozessoren 1 Core als 2 Hyperthreads präsentiert und im OVM-Betriebssystem ist 1 Hyperthread als 1 CPU sichtbar (und dieser hat eine Leistung von 0.5 bis 1 Thread,

je nachdem wie der andere Thread gerade ausgelastet ist) (siehe Abbildung 8).

Eine virtuelle CPU einer VM kann standardmäßig eine beliebige, möglichst wenig belastete CPU des Servers verwenden. Zu lizenzieren sind deshalb alle vorhandenen physischen Cores. Bei OVM gibt es jedoch die Möglichkeit des sogenannten Hard-Partitionings, also der expliziten Zuweisung einer bestimmten CPU zu einer oder mehreren VMs (siehe Listing 1). Zu lizenzieren sind dann nur die zugewiesenen Cores. Wenn man also mit Hyperthreading 2 CPUs/Threads desselben Cores

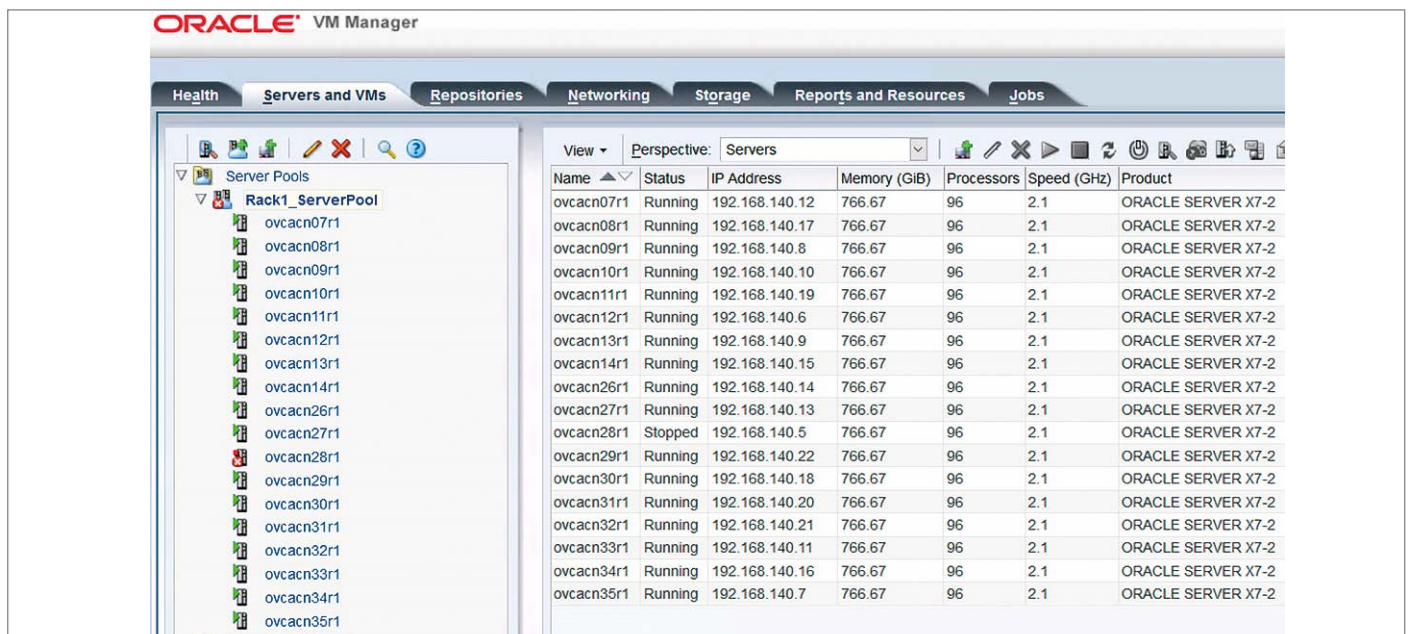


Abbildung 4: Ansicht des OVM Managers auf einer PCA mit 18 Compute Nodes. Die Hostnamen sind vorgegeben, z.B. „ovcacn35r1“ OVM Cloud Appliance Compute Node im 35. Slot des Rack 1. (Quelle: Oracle)

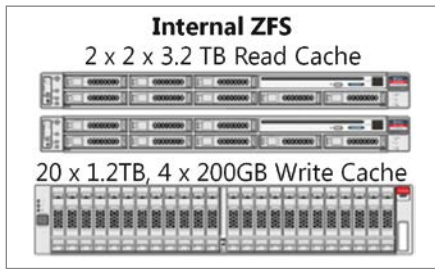


Abbildung 5: Interner ZFS Storage
(Quelle: Martin Bracher)

zuweist, zählt dies für die Lizenzierung als 1 Core. Nachteil ist, dass man dann die Live-Migration, also das Verschieben von VMs im laufenden Betrieb auf einen anderen Server, nicht mehr nutzen darf.

Es gibt noch eine weitere Möglichkeit, um nicht alle vorhandenen Cores lizenzieren zu müssen: das **Trusted Partitioning**, das bei bestimmten Engineered Systems (wie etwa der PCA) verwendet werden kann (<https://www.oracle.com/assets/partitioning-070609.pdf>). Zu lizenzieren ist hierbei die maximale Anzahl gleichzeitig laufender virtueller CPUs (vCPU) der VMs. 2 vCPU zählen lizenztechnisch als 1 Core, also benötigen 4 vCPU 1 EE- Lizenz (siehe Abbildung 9). Folgende Bedingungen sind einzuhalten: Hyperthreading muss aktiviert sein, und die VMs müssen durch den Enterprise Manager (EM) überwacht werden. Im sogenannten „Connected Mode“ wird die Benutzung durch den EM direkt an My Oracle Support (MOS) rapportiert. Im „Disconnected Mode“ müssen vierteljährliche Auswertungen für zwei Jahre aufbewahrt und auf Anfrage an Oracle geliefert werden. Der große Vorteil dieser Lizenzierungsvariante ist, dass man weiterhin die Live-Migration verwenden darf.

Public Cloud mit der Private Cloud Appliance

Bei dem in der Einleitung erwähnten gemeinsamen Cloud-Angebot von Swisscom und Trivadis kann der Kunde einen Oracle-Datenbank-Service selbstständig bestellen. Danach wird automatisch eine neue VM mit einer Single- oder Multi-Tenant- Datenbank der gewünschten Version erstellt.

Den Schritt 3 (siehe Abbildung 10) der VM-Erstellung schauen wir nun etwas genauer an. Dabei nutzen wir die Vorteile der OVM-Templates. Ein Template ist im

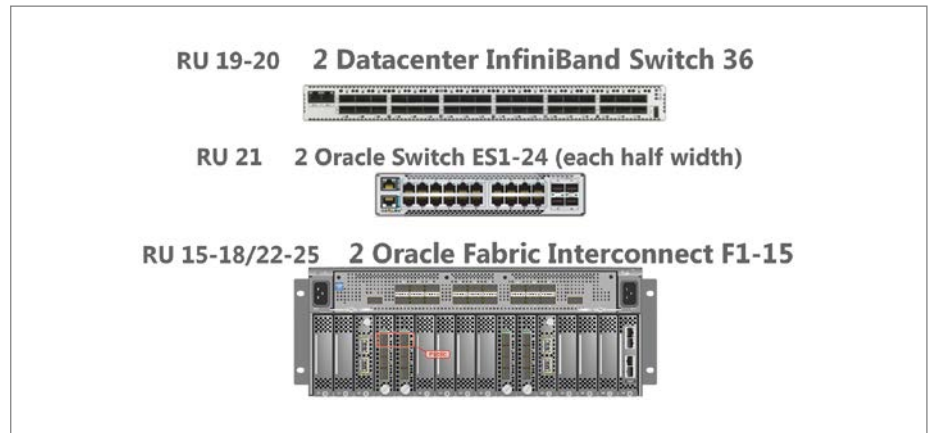


Abbildung 6: Netzwerk-Hardware (Quelle: Martin Bracher)

Prinzip eine VM, die zum Klonen verwendet wird. Beim Klonen werden die virtuellen Disks dupliziert (physische Kopie oder Snapshot) und das Konfigurationsfile ko-

piert und sinngemäss angepasst (Neue MAC-Adressen, neues Disk-Mapping). Andere Settings wie vCPU und Memory bleiben unverändert (siehe Listing 2). Ebenso

| Socket 0 | | Socket 1 | |
|----------|--------|----------|--------|
| Core 0 | Core 1 | Core 0 | Core 1 |
| CPU0 | CPU1 | CPU2 | CPU3 |

Abbildung 7: Nicht-Multithreaded: 1 Core = 1 CPU (Quelle: Martin Bracher)

| Socket 0 | | | | Socket 1 | | | |
|----------|------|--------|------|----------|------|--------|------|
| Core 0 | | Core 1 | | Core 0 | | Core 1 | |
| HT 0 | HT 1 | HT 0 | HT 1 | HT 0 | HT 1 | HT 0 | HT 1 |
| CPU0 | CPU1 | CPU2 | CPU3 | CPU4 | CPU5 | CPU6 | CPU7 |

Abbildung 8: Multithreaded: 1 Hyperthread = 1 CPU (Quelle: Martin Bracher)

```
[root@ttcovms02 ~]# xenpm get-cpu-topology
CPU      core  socket  node
CPU0     0     0       0
CPU1     0     0       0
CPU2     1     0       0
CPU3     1     0       0
CPU4     9     0       0
CPU5     9     0       0
CPU6    10     0       0
CPU7    10     0       0
```

Listing 1: Wenn man durch Hard-Partitioning die CPUs 0-3 den VMs zuweist, werden Core 0 und 1 verwendet. Dies erfordert somit 1 EE-Lizenz.

| | | | |
|----------------------|----------------------|-----------------------|----------------------|
| # vm001 vcpus = 4 | # vm002 vcpus = 8 | # vm003 vcpus = 20 | # vm004 vcpus = 4 |
|----------------------|----------------------|-----------------------|----------------------|

Abbildung 9: In diesem Beispiel werden 36 vCPU verwendet. Bei Trusted Partitions sind 18 Cores zu lizenzieren, also werden 9 EE-Lizenzen benötigt. (Quelle: Martin Bracher)

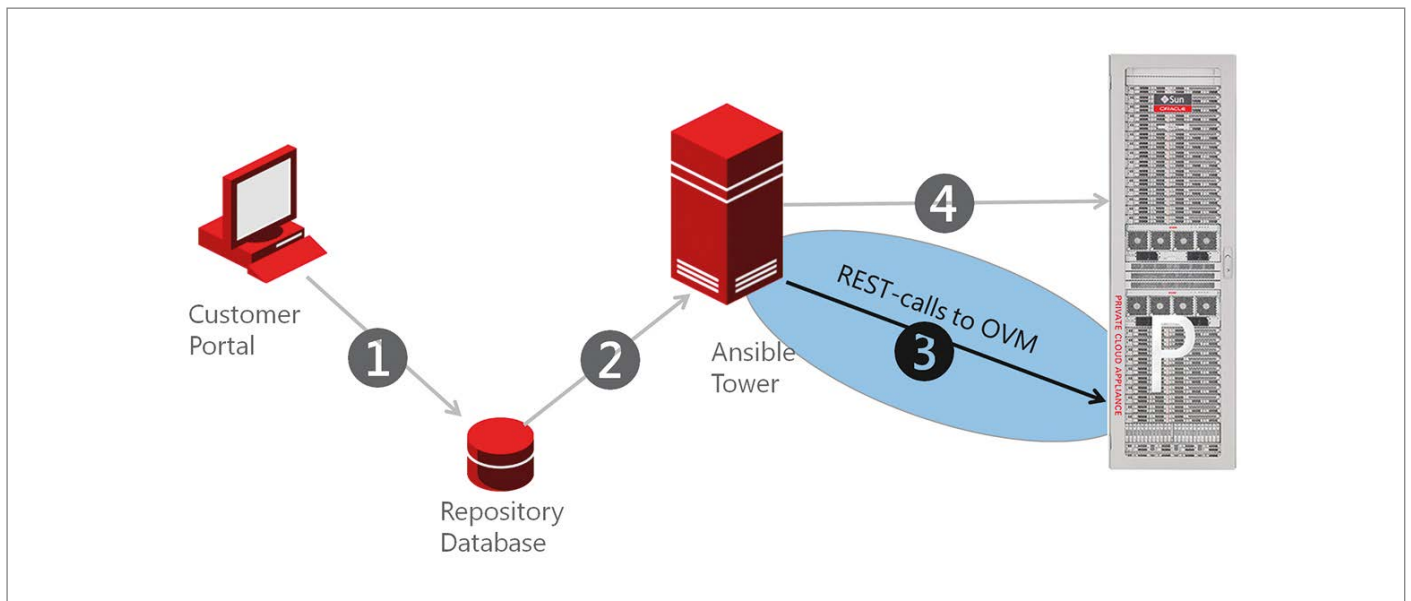


Abbildung 10: ❶ Kunde bestellt DB-Service; ❷ Ansible-Workflow wird angestoßen; ❸ Ansible macht REST-Calls zum OVM zur VM-Erstellung; ❹ Ansible erstellt die DB in der neuen VM. (Quelle: Martin Bracher)

wird der Inhalt der Disks nicht verändert. Was ist nun der Unterschied zu einer normalen VM? Ein Template ist so eingerichtet, dass sich nach dem Klonen die neue VM beim ersten Booten selbst konfiguriert (Hostname, IP-Adresse, ssh-Keys). Den entsprechenden Template-Mechanismus liefert Oracle im [ol7_addon] Repository. Damit können wir uns eigene Templates bauen.

Nicht jeder Kunde braucht gleich viel CPU und Memory, und auch nicht jede Datenbank verwendet dieselbe Oracle-Version. Deshalb verwenden wir in unserem Projekt nicht genau das Prinzip des Klonens eines Templates. Wir ver-

```
yum install libovmapi ovmd ovm-template-config* xenstoreprovider
systemctl enable ovm-template-initial-config
systemctl enable ovmd
systemctl start ovmd
systemctl start ovm-template-initial-config
```

Listing 2: Installation der Template-Konfiguration.

wenden stattdessen ein Baukasten-Prinzip. Wir definieren zuerst eine neue VM mit der gewünschten Memory- und CPU-Ausstattung, sowie den benötigten Netzwerken (VLAN). Danach klonen wir von einer Master-VM die OS-, Swap-, Temp- und u01-Disk. Die OS-Disk ist mit der Template-Konfiguration eingerich-

tet, die /u01 Disk ist für Oracle Log- und Konfigurationsdateien bestimmt (der Platz für Datenbank-Files wird später per NFS von einer ZFS Storage Appliance zur Verfügung gestellt). Die Disk mit der passenden Oracle-Version (12.2 bis 19 stehen aktuell zur Auswahl) wird ab einer anderen Master-VM geklont. Diese

```
def createVmDiskMap(restsession, baseUri, module):
    vmid=get_id_from_name(restsession,baseUri,'Vm',module.params['vmname'])['value']
    uri4='{base}/Vm/{VmId}/VmDiskMapping'.format(base=baseUri, VmId=vmid)
    Data4={
        "vmId": {
            "type": "com.oracle.ovm.mgr.ws.model.Vm",
            "value": vmid
        },
        "virtualDiskId": {
            "type": "com.oracle.ovm.mgr.ws.model.VirtualDisk",
            "value": diskid
        },
        "name": module.params['name'],
        "diskTarget": module.params['slot']
    }
    createVmDiskMap=restsession.post(uri4, data=json.dumps(Data4))
    createVmDiskMapJson=createVmDiskMap.json()
    wait_for_job(createVmDiskMapJson['id']['uri'], restsession)
```

Listing 3: Funktion im Python-Modul zur Erstellung des Diskmappings

Disk-Kopien werden nun an die neue VM gehängt. Anschließend wird die neue VM gestartet, wobei dann die Template-Konfiguration aktiv wird. Mittels des Messaging-Systems von OVM übermitteln wir die für die Konfiguration benötigten Informationen (Hostname, Netzwerkdaten).

Ansible und OVM REST-API

Der ganze Ablauf wird mit Ansible automatisiert. Ansible ist ein Open Source Tool für Provisionierung, Konfigurationsmanagement und Applikations-Deployment. Als Job-Scheduling System verwenden wir Ansible Tower (kommerziell).

Oracle VM stellt ein REST-API zur Verfügung. Die Dokumentation dazu findet sich nicht im Internet, sondern befindet sich gut versteckt auf dem Installations-ISO des OVM-Managers. Sie ist auch nicht sonderlich benutzerfreundlich, da sie fast keine Beispiele enthält.

Glücklicherweise bietet Oracle bereits ein Ansible-Modul mit einem minimalen Funktionsumfang zum Klonen, Starten, Stoppen und Löschen von VMs an. Durch das Studium dieses Modul-Codes kann man die Funktionsweise des REST-APIs besser verstehen. Ansible-Module werden in Python geschrieben. Zu beachten ist, dass in dieser Sprache die Einrückung wesentlich ist.

Das Modul ist übrigens nicht auf die PCA beschränkt, es kann auch auf einer selbst gebauten OVM-Umgebung eingesetzt werden.

Basierend auf diesem Modul konnten wir für unser Projekt noch weitere Module erstellen:

- klonen einer einzelnen Disk
- anhängen der Disk an eine VM
- erstellen/entfernen von Kundennetzwerken (VLAN)
- erstellen/entfernen von VLAN Ports
- senden von Informationen an die VM

Für die eigenen Funktionen konnte ich einige der bestehenden Funktionen weiterverwenden, wie beispielsweise in *Listing 3* dargestellt `get_id_from_name` und `wait_for_job`.

Für diejenigen, die gern selbst solche Module entwickeln wollen, möchte ich noch ein paar Tipps weitergeben:

```
import os
file = open("/tmp/out.txt", "w")
file.write(json.dumps(Result.json(), indent=4 ))
```

Listing 4: Ausgabe von Debug-Informationen.

Während der Entwicklung empfiehlt es sich, Debug-Output in ein File zu schreiben, da es schwierig ist, die Informationen bei der Ausführung mit Ansible anzuzeigen (*siehe Listing 4*).

Wenn jemand den Code des bestehenden Oracle-OVM-Moduls nicht bis ins letzte Detail versteht, geht es ihm wohl ähnlich wie mir... Aber auch so kommt man zum Ziel. Ich kopierte eine ähnliche Funktion im bestehenden Modul und passte sie iterativ (also mit Try-And-Error) an, bis sie lief. Um beispielsweise herauszufinden, was man in `Data4` (*siehe Listing 3*) übergeben muss, fragt man am besten zuerst mal ein bestehendes Diskmapping mit `restsession.get` ab und betrachtet das Resultat, welches man mit der beschriebenen Debug-Methode in ein File schreiben kann.

Fazit

Mit der PCA bekommt man ein Engineered System, welches gut skalierbar ist und aufgrund des Trusted Partitioning auch lizenztechnisch die nötige Flexibilität bietet, ohne dass man diesbezüglich einen großen Aufwand betreiben müsste. Dank des REST-APIs lässt sich das Provisionieren der Services automatisieren. Die Plattform eignet sich somit gut für ein Cloud-Angebot.

Quellen

- [1] <https://www.oracle.com/technetwork/server-storage/private-cloud-appliance/overview/index.html>
- [2] <https://www.oracle.com/technetwork/server-storage/vm/ansiblevm lifecyclemodule-5106627.pdf>
- [3] <https://www.oracle.com/virtualization/technologies/vm/downloads/ovm-tools-downloads.html>

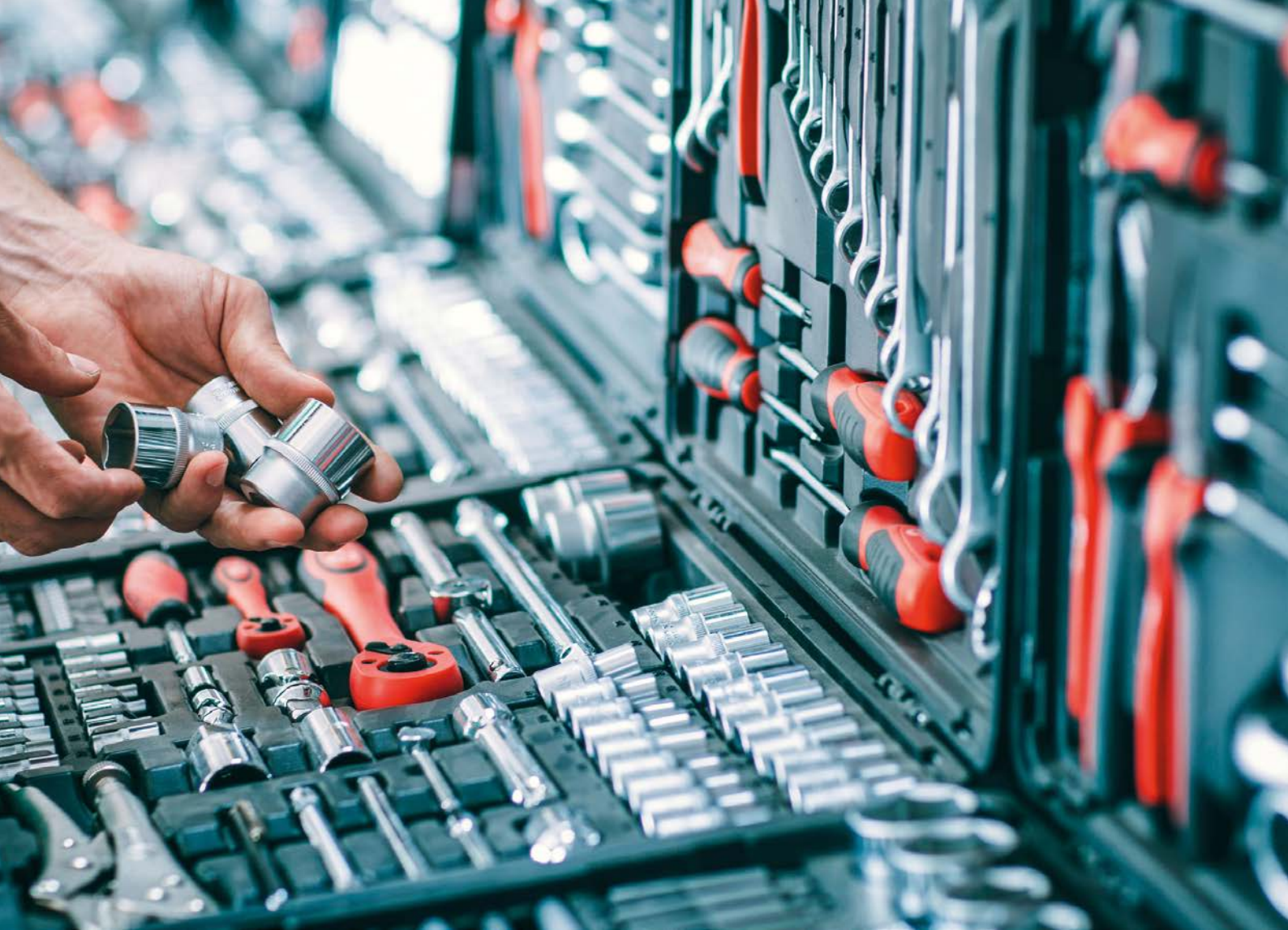
Über den Autor

Martin Bracher ist seit über 20 Jahren beruflich in der Informatik tätig, davon über 15 Jahre bei Trivadis. Nach dem Informa-

tikstudium an der Universität Bern erfolgte ab 1997 der Umstieg in die Oracle-Welt. Er war als Analytiker, Software Engineer und DBA in verschiedenen Oracle-Projekten in verschiedensten Branchen tätig. Wegen seiner sehr guten Kenntnisse in Unix/Linux wird er gern bei konzeptioneller Beratung, Installation und Konfiguration eingesetzt. Zudem hat er große Erfahrung im Bereich Automatisierung/Schnittstellen zwischen Operating Systems (Unix), Datenbank und der Entwicklung. Er ist auch Spezialist in Sachen Oracle-Hochverfügbarkeit (Data-guard, RAC). Bei der Trivadis Toolbox-Software BasEnv zum Setzen von Datenbankumgebungen ist er der Hauptentwickler.



Martin Bracher
martin.bracher@trivadis.com



Hilfreiche Tools für den täglichen Exadata-Betrieb (Teil 2)

Michael Schulze, Opitz Consulting Deutschland

Wer sich im Zuge der Konsolidierung seiner Oracle-Datenbanken für eine Exadata-Datenbanklösung entscheidet und diese in den Betrieb bringt, weiß, dass sie als Cluster-Umgebung für besonders kritische Datenbanken intensiv genutzt und in naher Zukunft ein Wachstum erfahren wird. Nicht selten müssen Administratoren hier mehr als 100 Datenbankinstanzen versorgen, verteilt auf verschiedene Systembereiche, wie zum Beispiel Test, Stage und Prod, mit den unterschiedlichsten Verfügbarkeitsanforderungen. Für die Sicherstellung der Verfügbarkeit kommt oft RAC und Oracle Data Guard (Standby-Lösung 2. Standort) zum Einsatz. In diesem komplexen Kontext sind passende Analyse-Tools eine Grundvoraussetzung, um den Überblick zu behalten und einen sicheren sowie reibungslosen System- und Datenbankbetrieb zu gewährleisten. Geeignete Tools liefern eine strukturierte Dokumentation von Einzelkomponenten und Systemständen und helfen bei Patch-Unterstützung, Housekeeping, Monitoring und vielem mehr. In dieser zweiteiligen Artikelserie stelle ich Ihnen einige Werkzeuge vor, die sich bei der Unterstützung des Exadata-Betriebs in unserem Kundenumfeld bewährt haben. Darunter finden sich nicht nur Oracle Tools für die Betriebsunterstützung, sondern auch eigene oder frei verfügbare Skripte. Im ersten Teil der Artikelserie, der im Red Stack Magazin 2/2020 erschienen ist, ging es um Tools zur Dokumentations- und Patchingunterstützung. Der zweite Teil in dieser Ausgabe behandelt Hilfsmittel für das Housekeeping, Monitoring und zentrale Analysewerkzeuge.

Housekeeping auf Exadata

Die Verwaltung komplexer Oracle-Systeme wie Exadata ist für Datenbankadministratoren keine leichte Aufgabe. Eine der größten Herausforderungen ist die stetige Bereinigung von Traces und Protokollen, die von einer Vielzahl von Komponenten und Prozessen des Systems erstellt werden. Zu nennen wären hier Komponenten wie Cluster, Grid Infrastructure (GI), RDBMS, AHF/TFA etc. So können bei Systemen mit mehr als 100 laufenden Datenbankinstanzen schon innerhalb sehr kurzer Zeit große Mengen an Logfile-Informationen entstehen. Werden diese Informationen nicht periodisch konsequent bereinigt, kann dies schnell zu Problemen mit dem Filesystem und zu Betriebsausfällen führen. Deshalb ist ein sauberes Housekeeping als Grundvoraussetzung für einen stabilen Datenbankbetrieb auf der Exadata unerlässlich.

Ich möchte hier zunächst auf das Automatic Diagnostic Repository (ADR) eingehen, das mit der Oracle-Datenbank 11g eingeführt wurde. Oracle hat mit dem ADR eine einheitliche hierarchische Repository-Verzeichnisstruktur für Log- und Trace-Daten eingeführt. Diese Struktur vereinfacht die Verwaltung der Daten erheblich. Zur zentralen Verwaltung kann das Commandline-Interface ADRCI genutzt werden, das viele Befehle zur Verwaltung enthält, wichtige wären hier:

- view log / tracefiles
- purge log / tracefiles
- show incidents, packaging for Oracle Support

Housekeeping mit purgeLogs

Eine weitere einfache Möglichkeit ist der Einsatz eines von Oracle bereitgestellten Perl-Skriptes namens `purgeLogs.pl`. Für ein zentrales Housekeeping auf Exadata-Systemen mein Favorit!

Das `purgeLogs`-Skript, das unter DocID: 2081655.1 verfügbar ist, ist ein sehr einfaches One-Button-Skript, das alle anfallenden Trace- und Protokolldateien in einem RAC-Cluster-Umfeld automatisch bereinigt. Die Programmfunktionen des Skripts decken unter anderem die folgenden Komplexe ab:

| | |
|--|---|
| <code>-days <Tage></code> | Bereinigen von orcl, tfa, osw, oak-Protokollen/Traces |
| <code>-orcl <Tage></code> | Nur GI / RDBMS-Protokolle/Traces behandeln |
| <code>-tfa <Tage></code> | Nur TFA-Repository-Daten behandeln |
| <code>-osw <Tage></code> | Nur OSWatcher-Archive behandeln |
| <code>-oda <Tage></code> | Nur OAK / DCS-Protokolle entfernen (nur ODA) |
| <code>-extra '<Ordner>': <Tage></code> | Nur Dateien in benutzerdefinierten Ordnern löschen |
| <code>-aud</code> | Audit-Protokolle basierend auf der Option ' <code>-orcl <Tage></code> ' behandeln |
| <code>-lsnr</code> | Bereinigung des Listener-Logs unabhängig vom Alter |
| <code>-dryrun</code> | Zeigt die Befehle an, ohne sie auszuführen |
| <code>-h</code> | Hilfefunktion |

Tabelle 1: `purgeLogs` – Parameter und Optionen

- Audit Files
- Listener Logs
- Oracle GI/RDBMS Logs und Trace Files
- TFA Repository, OS-Watcher Logs
- Angabe spezifischer Verzeichnisse (extra Option)
- Housekeeping auf ODA (oda Option)

In der Praxis sieht das so aus: Das Skript muss als root-User auf jedem Database-Node der Cluster-Umgebung separat ausgeführt werden. Im besten Fall wird es über die root crontab des Nodes zur periodischen Ausführung eingeplant.

Tabelle 1 zeigt eine Zusammenfassung der `purgeLogs`-Parameter und -Optionen, die teils auch kombiniert werden können. Der Default-Wert für die Datenerhaltung liegt bei 30 Tagen, kann jedoch über den Skript-Tag `<Tage>` modifiziert werden.

Über den Befehl `./purgeLogs -orcl 30 -tfa 30 -aud -lsnr` werden beispielsweise alle GI / RDBMS Logs und Traces, TFA-Repository-Daten und Auditinformationen entfernt, die älter als 30 Tage sind. Auch die Listener-Logs, die stark anwachsen können, werden hier bereinigt.

Exadata Monitoring und Analyzing

Im Folgenden sollen einige Ideen und Tipps für das einfache Monitoring von Exadata-Umgebungen an Beispielen erläutert werden.

Enterprise Manager Cloud Control

Im Exadata-Umfeld findet häufig der Oracle Enterprise Manager Cloud Control (kurz CC) Anwendung. Der CC ist in der Lage, durch die Implementierung entsprechender Plug-ins auch komplexe Exadata-Umgebungen mit ihren Komponenten bis auf Hardwareebene effizient zu verwalten. Jedoch sind die Administration und Verwaltung des CC sehr vielschichtig, aufwendig und komplex.

Additionalales Skript: `rac_status.sh`

Für einen schnellen Überblick über den RAC-Cluster kann ich im Exadata-Umfeld ein frei verfügbares Skript von Fred Denis namens „`rac_status.sh`“ empfehlen. Das Skript steht auf dieser Seite zur Verfügung: <https://github.com/freddenis/oracle-scripts/blob/master/rac-status.sh>

Dieses Skript ist in der Lage, im Terminal eine sehr gute grafische Übersicht über die wesentlichen Bestandteile des Oracle-Database-Clusters darzustellen. Es wird fortlaufend gepflegt und enthält unter anderem auch Informationen über die Exadata-Hardware. Als Alternative oder Ergänzung zum recht komplexen „`crsctl stat res -t`“-Output bekommt man hier im Ergebnis einen Gesamtüberblick über den DB-Cluster, der insbesondere durch die intuitive farbliche Darstellung für Übersicht sorgt. In komplexen Umgebungen, bestehend aus mehreren Clustern, kann das Tool eine sehr gute Hilfe sein.

```

Cluster ffgr is a X5-2 Elastic Rack HC 4TB

-----
Listener | Port | ghtr5643 | fjjgh765 | kliz4530 | Type |
-----
LISTENER | TCP:4020 | Online | Online | Online | Listener |
LISTENER_4021 | TCP:4021 | Online | Online | Online | Listener |
LISTENER_4022 | TCP:4022 | Online | Online | Online | Listener |
LISTENER_DG | TCP:4023 | Online | Online | Online | Listener |
LISTENER_DGSCAN_SCAN1_NET2 | TCP:4023 | - | - | Online | SCAN |
LISTENER_SCAN1 | TCP:4020 | Online | - | - | SCAN |
LISTENER_SCAN2 | TCP:4020 | Online | - | - | SCAN |
LISTENER_SCAN3 | TCP:4020 | - | Online | - | SCAN |
-----

DB | Version | ghtr5643 | fjjgh765 | kliz4530 | DB Type |
-----
ght453 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
ssder4 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
jzt556 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
juz453 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
lop675 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
loi766 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
asde34 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
ert323 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
tzt453 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
mnn765 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
lkit55 | 12.1.0.2 (2) | Open | - | - | RAC (P) |
oiui88 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
rttz45 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
vbgf56 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
ysed44 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
lki564 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
juzrff | 12.2.0.1 (1) | Open | - | - | RAC (P) |
kiu543 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
bnh564 | 12.2.0.1 (1) | Open | - | - | RAC (P) |
sder44 | 12.2.0.1 (1) | - | - | Open | RAC (P) |
bght55 | 12.2.0.1 (1) | - | - | Open | RAC (P) |
ffgrrdb1 | 19.4.0.0 (3) | Shutdown | Shutdown | Shutdown | RAC (P) |
ffgrrdb2 | 12.2.0.1 (1) | Shutdown | Shutdown | Shutdown | RAC (P) |
ffgrrdb3 | 11.2.0.4 (4) | Shutdown | Shutdown | Shutdown | RAC (P) |
ffgrrdb4 | 12.1.0.2 (2) | Shutdown | Shutdown | Shutdown | RAC (P) |
-----

ORACLE_HOME references listed in the Version column ("'" means "same as above")

1 : /u01/app/oracle/product/12.2.0.1/dbhome_1 oracle oinstall
2 : /u01/app/oracle/product/12.1.0.2/dbhome_1 "' "'
3 : /u01/app/oracle/product/19.4.0.0/dbhome_1 "' "'
4 : /u01/app/oracle/product/11.2.0.4/dbhome_1 "' "'

█ : Has been restarted less than 24 hours ago
    
```

Abbildung 1: Ein anonymisiertes Beispiel mit dem Aufruf ./rac_status.sh (Quelle: Michael Schulze)

Und so funktioniert es: Da es sich bei rac_status.sh um ein All-in-one-Skript handelt, muss es lediglich mit dem Skriptnamen aufgerufen werden. Der Aufruf stellt diese Elemente im Überblick dar (siehe Abbildung 1).

- Cluster- und Hardware-Informationen
- Listener mit Name, Port, Status (Node-übergreifend)
- Datenbanken mit Namen, Versionen, Status, DB-Type/-Role (Node-übergreifend)
- Locations entsprechend der Oracle Home(s)

Exadata-Profil

Die neueren Versionen des Remote Diagnostic Agent (RDA) enthalten auch Pro-

file für die Exadata. Die Anwendung dieser Profile hilft bei der Fehlerdiagnose bis auf Hardwareebene der Exadata. Im Folgenden daher mal ein paar Beispiele und Outputs, die zeigen, wie man die Exadata-Profil anwendet und welche Informationen man dem Output entnehmen kann.

Mit dem Befehl ./rda.sh -L | grep Exadata verschaffen wir uns zunächst ein Überblick über die im RDA enthaltenen Exadata-Profil (Stand RDA 19.X) siehe Tabelle 2)

Die einzelnen Profile decken verschiedene Hardwarebereiche im Exadata-Kontext ab und geben in Form von RDA-Reports detaillierte Informationen zum jeweiligen Hardwarebereich aus (siehe Abbildung 2). Die Reports umfassen diese Bereiche:

- Netzwerk/InfiniBand: Diagnose, Host-Informationen, Net Discover, Switch-Informationen, Topology
- Festplatten/Flashdisks: BIOS-Infos, Temperaturinformationen, MegaCLI64 (Raid-Infos), Backup-Batterie-Status, Partitionsinformationen, SCSI-Status, Seriennummer, PCI-Infos
- Cell-Ebene: RDMA-Device-Info, Cell Collection, Failures
- ILOM: Events, Firmware, Netzwerkinfos, Sensoren

Autonomous Health Framework

Unter dem Dach des Autonomous Health Framework (AHF) wurden viele bekannte Tools aus dem Umfeld des Oracle-RAC-

RDA 19.4 Main Index

Executed on test01

- [Overview](#)
- [Operating System Setup](#)
- [Performance](#)
- [Network](#)
- [Oracle Net](#)
- [RDBMS](#)
 - [RDBMS Log/Trace Files](#)
- [Oracle Exadata \(FailedDrives\)](#)

Oracle Exadata (FailedDrives)

- Compute Node
 - [MegaCli64 Information](#)
 - [ADP All Information](#)
 - [BBU Command](#)
 - [ADP Events](#)
 - [Configuration Display](#)
 - [FwTerm Log](#)
 - [LD Information](#)
 - [LdPd Information](#)
 - [PD List](#)
 - [PD Short List](#)
 - [Disk Partition Information](#)
 - [Image Information](#)
 - [IPMI Information](#)
 - [PCI Devices Configuration](#)
 - [RAID Status](#)
 - [RDMA Devices](#)
 - [SCSI Devices](#)
 - [Serial Numbers](#)
- Cell Nodes
 - [Cell Collection Summary](#)

Abbildung 2: RDA-Report, Aufruf über:
`./rda.sh -p Exadata_FailedDrives`
 (Quelle: Michael Schulze)

| Profil | Beschreibung |
|-----------------------|--|
| EXA | Collects Exadata Information |
| cell | Tests Connection to Exadata Cells |
| exa:default | Collects Exadata-specific information |
| Exadata_Assessment | Oracle Exadata assessment collections |
| Exadata_CellBrownout | Oracle Exadata long brownout due to cell |
| Exadata_CellFailure | Oracle Exadata cell failure problems |
| Exadata_DatabaseCrash | Oracle Exadata database crash problems |
| Exadata_DatabaseHang | Oracle Exadata database hang problems |
| Exadata_FailedDrives | Oracle Exadata failed drives problems |
| Exadata_FlashDrives | Oracle Exadata flash drives problems |
| Exadata_IbSwitch | Oracle Exadata IB switch problems |
| Exadata_Ilom | Oracle Exadata ILOM problems |
| Exadata_ListenerHang | Oracle Exadata listener hang problems |
| Exadata_Network | Oracle Exadata general network problems |
| Exadata_NetworkCable | Oracle Exadata network cabling problems |
| Exadata_RacInstance | Oracle Exadata RAC instance problems |
| Exadata_SickCell | Oracle Exadata sick cell problems |

Tabelle 2: Übersicht über die Exadata-Profile (Beispiel)

Clusters sowie weitere neue Werkzeuge in ein einziges Framework integriert (siehe Abbildung 3). Zu nennen wären hier: der Trace File Analyzer (TFA), cluvfy, exachk/orachk, oswatcher. Die Zusammenfassung in einem Toolset (AHF) sorgt für Ordnung im Wirrwarr der vielen verfügbaren Werkzeuge. Zu beziehen ist AHF unter der Doc ID: 2550798.1

Installation des AHF auf Exadata

TFA als ein Bestandteil von AHF, wird unter anderem auch über Grid-Infrastructure-Patches mit ausgeliefert, dann aber nicht in der aktuellsten Version. Es ist daher ratsam, ihn zusätzlich in einer aktuellen Version zu installieren. Die Installation ist nicht kompliziert und verteilt sich automatisch auf alle beteiligten Nodes der Exadata. Zunächst wird dafür die AHF-Software auf dem System bereitgestellt. Dann kann die Installation erfolgen, über den Befehl `/ahf_setup.sh`.

Per Default wird AHF in das Verzeichnis `/opt/oracle.ahf` installiert, man kann aber auch ein anderes Verzeichnis wählen. Das Setup prüft bereits installierte Versionen, bereinigt diese und verteilt die

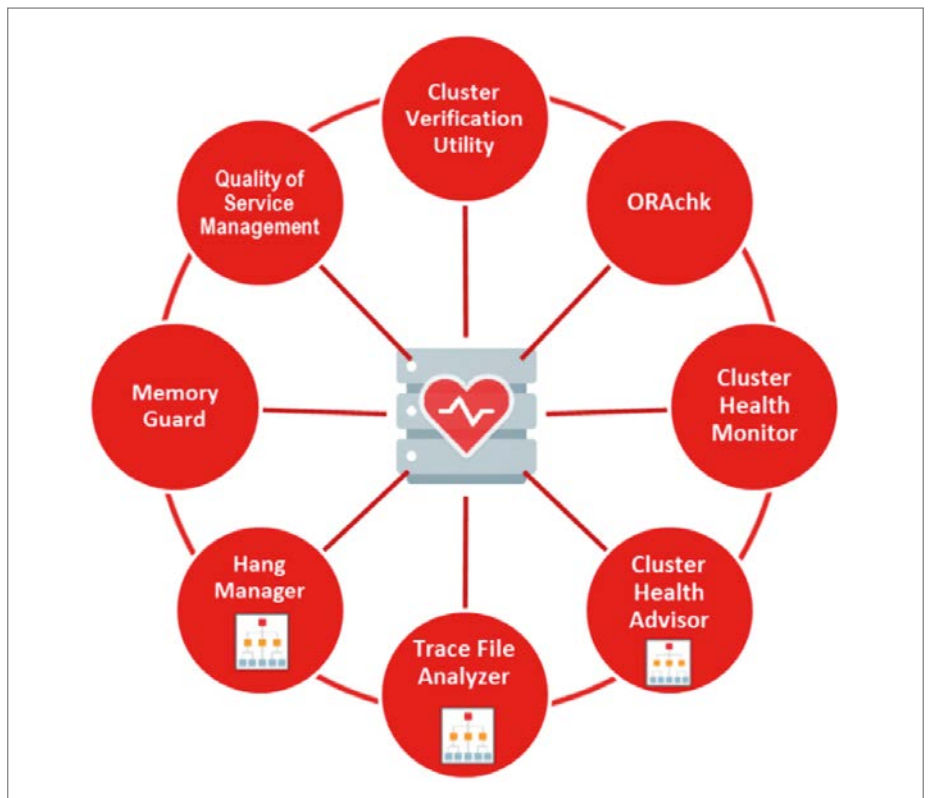


Abbildung 3: Überblick AHF (<https://www.oracle.com/de/database/technologies/rac/ahf.html>)
 (Quelle: Oracle)

```

+-----+-----+-----+-----+-----+-----+-----+
| Host           | Status of TFA | PID    | Port | Version | Build ID           | Inventory Status |
+-----+-----+-----+-----+-----+-----+-----+
| test01        | RUNNING      | 392702 | 5000 | 19.3.0.0.0 | 19300020191018172057 | COMPLETE        |
| test02        | RUNNING      | 192204 | 5000 | 19.3.0.0.0 | 19300020191018172057 | COMPLETE        |
+-----+-----+-----+-----+-----+-----+-----+
    
```

Abbildung 4: Überprüfung des Print-Status mit `opt/oracle.ahf/tfa/bin/tfactl print status` (Quelle: Michael Schulze)

```

+-----+-----+-----+-----+
| TOOLS STATUS - HOST : b-ref-ixsrv-dc-01 |
+-----+-----+-----+-----+
| Tool Type           | Tool          | Version | Status |
+-----+-----+-----+-----+
| Development Tools  | exachk        | 19.3.0.0.0 | DEPLOYED |
+-----+-----+-----+-----+
| Support Tools Bundle | oswbb         | 8.3.2 | NOT RUNNING |
|                   | prw           | 12.1.13.11.4 | NOT RUNNING |
+-----+-----+-----+-----+
| TFA Utilities      | alertsummary  | 19.3.0.0.0 | DEPLOYED |
|                   | calog         | 19.3.0.0.0 | DEPLOYED |
|                   | dbcheck       | 18.3.0.0.0 | DEPLOYED |
|                   | dbglevel     | 19.3.0.0.0 | DEPLOYED |
|                   | grep         | 19.3.0.0.0 | DEPLOYED |
|                   | history       | 19.3.0.0.0 | DEPLOYED |
|                   | ls           | 19.3.0.0.0 | DEPLOYED |
|                   | managelogs   | 19.3.0.0.0 | DEPLOYED |
|                   | menu         | 19.3.0.0.0 | DEPLOYED |
|                   | param        | 19.3.0.0.0 | DEPLOYED |
|                   | ps           | 19.3.0.0.0 | DEPLOYED |
|                   | pstack       | 19.3.0.0.0 | DEPLOYED |
|                   | summary      | 19.3.0.0.0 | DEPLOYED |
|                   | tail         | 19.3.0.0.0 | DEPLOYED |
|                   | triage       | 19.3.0.0.0 | DEPLOYED |
|                   | vi           | 19.3.0.0.0 | DEPLOYED |
+-----+-----+-----+-----+
    
```

Abbildung 5: Überprüfung des Tool-Status mit `opt/oracle.ahf/tfa/bin/tfactl toolstatus` (Quelle: Michael Schulze)

| Funktion | Beschreibung |
|-----------------------------|---|
| ORAchk bzw. EXAchk | Oracle Compliance Checks |
| procwatcher | performance diag, session hangs |
| events | logfile warnings + errors |
| managelogs | disk space, purge ADR logs |
| alertsummary | for 1-n alertfiles from nodes |
| summary | all config |
| ls, vi, tail, grep, findstr | handles alert.logs + traces |
| param | show db + os param for pattern |
| oswatcher | collect os metrics, for node-etic + issues |
| oratop | real time db monitoring |
| pstack | process stack across nodes |
| dbglevel | set CRS trace level |
| history | for tfactl |
| changes | report changes for: system setup, db-param, patches in timeperiod |
| calog | major events from cluster |
| ps, tasklist | find procs |
| trriage | sum of oswatcher / exawatcher data |

Tabelle 3: Komponenten im TFA

aktuelle Softwareversion automatisch auf alle beteiligten Database Nodes. In der Exadata-Umgebung werden hierbei auch automatisch beteiligte Komponenten wie DB-Nodes, Cells etc. entdeckt. *Abbildung 4* und *5* zeigen, wie der Status der Installation geprüft wird.

TFA: Komponenten

Als zentraler Bestandteil des AHF ist Trace File Analyzer (TFA) die Schnittstelle zu sämtlichen relevanten Werkzeugen, die zur Analyse eines Oracle-RAC-Cluster-Datenbanksystems notwendig sind.

Tabelle 3 zeigt die Komponenten, die aktuell zum Umfang des TFA gehören.

`tfactl` ist das Command-Line Interface des TFA, das einige Parameter bietet, um Funktionen direkt aufzurufen. Mit dem Befehl `opt/oracle.ahf/tfa/bin/tfactl menu` können alle Komponenten und Tools des TFA zudem menübasiert erreicht werden. Nach der Eingabe öffnet sich ein ASCII-Menü, über das zum Beispiel in die Bereiche System Analysis, Collections oder Administration mit den oben genannten Einzelfunktionen navigiert werden kann.

TFA – Use Cases

Im Folgenden einige Beispiele der direkten Verwendung von TFA-Funktionen.

Component Summary mit dem TFA

Über den Befehl `opt/oracle.ahf/tfa/bin/tfactl summary -overview` werden Node-übergreifend sämtliche Daten gesammelt (*siehe Abbildung 6*).

Im Default-Modus gelangt man hier anschließend in ein ASCII-Menü, das den Zugang zu den einzelnen gesammelten Sammlungen bietet. Noch viel besser und strukturierter für eine Auswertung finde ich jedoch den HTML-Output, der mit der Option `-html` erzeugt werden kann. Diese Option bietet eine bequeme Navigation der gesammelten Daten im Browser (*siehe Abbildung 7*).

exachk über TFA

Ein bekanntes Tool für die Prüfung von Komponenten der Exadata, Cells, Cluster, GI und enthaltener Datenbanken ist exachk. Es gilt als Best Practice Standard Tool für die Healthcheck-Analyse der Exadata-Umgebung. Es bietet zum Beispiel in der Vorbereitung von Patch-Aktionen wertvolle Unterstützung bei der Bewertung des Exadata-Gesamtsystems. exachk kann im TFA folgendermaßen direkt aufgerufen werden: `/opt/oracle.ahf/tfa/bin/tfactl exachk`

Es folgt eine interaktive Abfrage verschiedener Parameter. Ergebnis ist ein

```
Executing Summary in Parallel on Following Nodes:
Node : test01
Node : test02

Component Specific Summary collection :
- Collecting CRS details ... Done.
- Collecting ASM details ... Done.
- Collecting ACFS details ... Done.
- Collecting DATABASE details ... Done.
- Collecting EXADATA details ... Done.
- Collecting PATCH details ... Done.
- Collecting LISTENER details ... Done.
- Collecting NETWORK details ... Done.
- Collecting OS details ... Done.
- Collecting TFA details ... Done.
- Collecting SUMMARY details ... Done.

Remote Summary Data Collection : In-Progress - Please wait ...
- Data Collection From Node - test02 .. Done.
```

Abbildung 6: Beispiel-Datensammlung (Quelle: Michael Schulze)

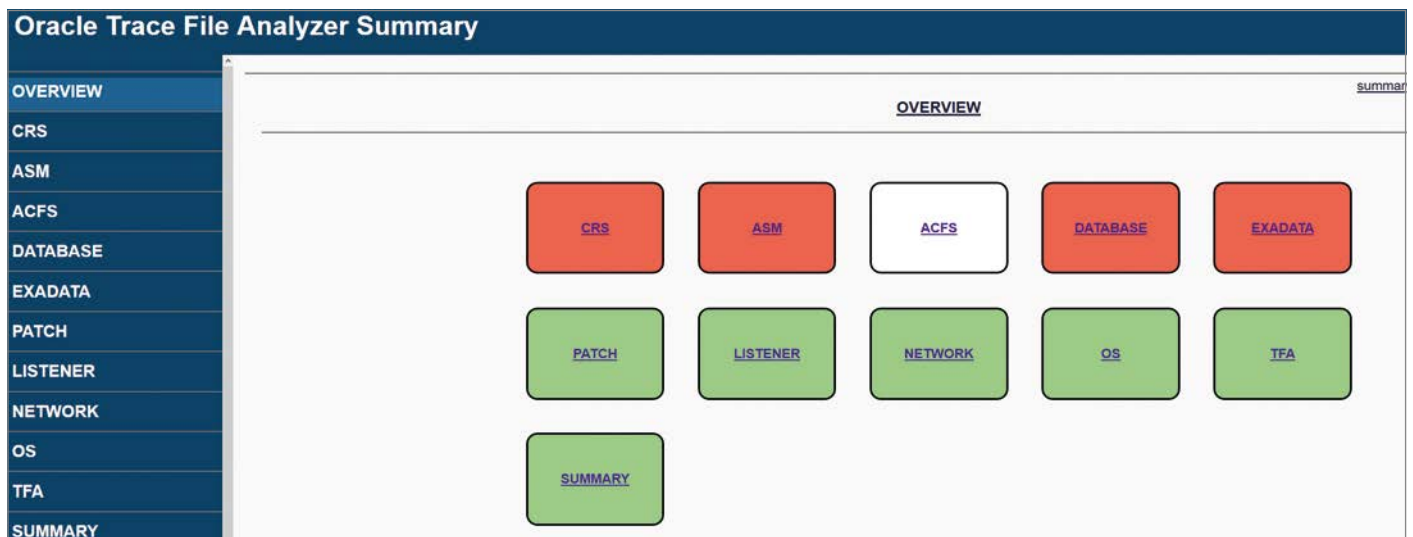


Abbildung 7: Die Browsernavigation im TFA (Quelle: Michael Schulze)

exachk- Report, der statische HTML-Seiten ausgibt und Aufschluss über den Systemzustand und den Zustand einzelner Teilbereiche nach Gewichtung bietet (siehe Abbildung 8).

Housekeeping mit dem TFA

Das Purging von Logfiles und Trace Files im ADR kann auch über den TFA getriggert und gesteuert werden. Dies ist manuell möglich, aber auch das automatische Entfernen von Logfiles lässt sich konfigurieren. Im Folgenden soll ein Beispiel eines manuellen Purgelaufs über TFA erläutert werden:

Zunächst ermittelt das System den Status quo und gibt aus, wie viele Log- und Trace-Informationen auf den Einzelsystemen vorhanden sind: `/opt/oracle.ahf/tfa/bin/tfactl managelogs -show usage`

Database Server

| Status | Type | Message |
|----------|-------------------|---|
| CRITICAL | OS Check | Hardware and firmware profile check is not successful. [Database Server] |
| CRITICAL | ORACLE_HOME Check | Oracle database(s) should be using RDS protocol for inter process communication |

Storage Server

`Status|Type|Message|Status On|Details`

InfiniBand Switch

`Status|Type|Message|Status On|Details`

Cluster Wide

`Status|Type|Message|Status On|Details`

Abbildung 8: Beispiel für den Output eines exachk-Reports (Quelle: Michael Schulze)

Ein Trockenlauf simuliert den Purgelauf und zeigt, wie viel Platz gespart werden könnte, ohne dass der Löschauf real gestartet werden muss. So könnte zum Bei-

spiel mit dem folgenden Befehl eine Vorhaltezeit von 30 Tagen definiert werden: `/opt/oracle.ahf/tfa/bin/tfactl managelogs purge -older 30d -dryrun`

```

-----
|                                     File System Variation : /u01/app/19.0.0.0/grid                                     |
-----+-----+-----+-----+-----+-----+-----+-----
| State | Name                               | Size      | Used      | Free      | Capacity | Mount |
-----+-----+-----+-----+-----+-----+-----+-----
| Before | /dev/mapper/VGExaDb-LVDbOral      | 309506048 | 151935928 | 143901484 | 52%      | /u01  |
| After  | /dev/mapper/VGExaDb-LVDbOral      | 309506048 | 149907436 | 145929976 | 51%      | /u01  |
-----+-----+-----+-----+-----+-----+-----+-----

-----
|                                     File System Variation : /u01/app/oracle/product/19.0.0.0/dbhome_1                                     |
-----+-----+-----+-----+-----+-----+-----+-----
| State | Name                               | Size      | Used      | Free      | Capacity | Mount |
-----+-----+-----+-----+-----+-----+-----+-----
| Before | /dev/mapper/VGExaDb-LVDbOral      | 309506048 | 149906132 | 145931280 | 51%      | /u01  |
| After  | /dev/mapper/VGExaDb-LVDbOral      | 309506048 | 149864764 | 145972648 | 51%      | /u01  |
-----+-----+-----+-----+-----+-----+-----+-----
    
```

Abbildung 9: Beispiel für die Anzeige der möglichen Platzersparnis (Quelle: Michael Schulze)

Das reale Löschen der Trace Files und Logfiles, in unserem Beispiel wieder für Daten, die älter als 30 Tage sind, wird durch den folgenden Befehl initiiert, der im Ergebnis die Platzersparnis noch einmal gegenüberstellt: `/opt/oracle.ahf/tfa/bin/tfactl manageLogs purge -older 30d` (siehe Abbildung 9).

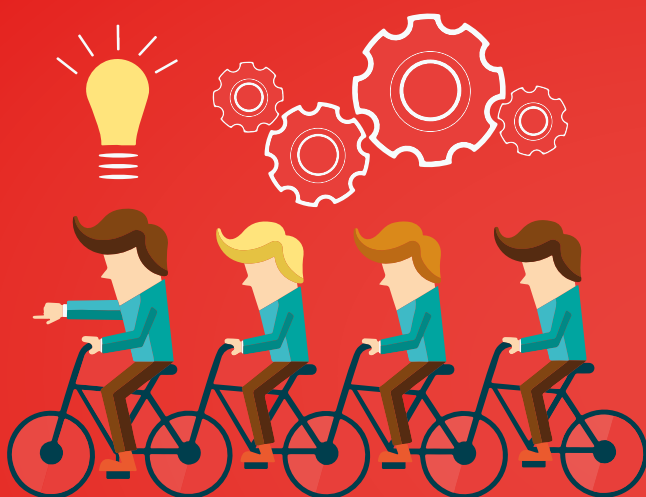
Zusammenfassung und Fazit

In dieser Artikelserie wurden Werkzeuge vorgestellt, die in der Praxis des Exadata-Betriebs unterstützen können. Neben

einigen bekannten Oracle-Werkzeugen wie RDA als Dokumentationsgrundlage, AHF und TFA für Analyse und Monitoring oder purgeLogs fürs Housekeeping wurden in diesem Teil auch interessante Third-Party-Skripte wie rac_status.sh und einige weitere Ideen präsentiert. Im Administratorenalltag kann diese Tool-sammlung dabei helfen, die vielen Komponenten in komplexen Exadata-Umgebungen zu beherrschen und gezielt zu analysieren. Insbesondere die Idee einer zentralen RDA-Dokumentation als strukturierte Dokumentationsgrundlage stellt dies sicher.



Michael Schulze
michael.schulze@opitz-consulting.com



DOAG

Die DOAG Print und Digital: Wir sind online für Sie da!

Um Sie auch weiterhin über aktuelle Informationen und Neuigkeiten aus der IT auf dem Laufenden zu halten, intensivieren wir unsere Aktivitäten im Print- und Onlinebereich.

News, Videos & Webinare

Online veröffentlichen wir News und Videos. Ein Blick in den Terminkalender lohnt sich weiterhin: Die DOAG plant eine Reihe Webinare und erweitert ihr Online-Angebot.

Magazine

Zögern Sie nicht, in unsere Zeitschriften Red Stack (inklusive Business News) und Java aktuell zu stöbern – auch online.

<https://www.doag.org>





Die passende Architektur für APEX wählen

Niels de Bruijn, MT AG

Welche Architektur brauche ich für meine APEX-Applikation? Eine einfache Frage, aber leider nicht einfach zu beantworten. In diesem Artikel gehen wir die üblichen Szenarien durch, damit man den Durchblick im Architektur-Dschungel bekommt.

Der schnellste Weg zur ersten APEX-App

Wer sich nicht selbst mit der Architektur beschäftigen möchte und damit leben kann, dass die Daten außerhalb des Firmennetzwerkes liegen, sollte sich die APEX-Cloudanbieter mal genauer anschauen. So gibt es eine kostenfreie „fully managed“-Alternative wie apex.oracle.com, wo innerhalb von wenigen Minuten ein Workspace mit 25 MB zur Verfügung steht (siehe *Abbildung 1*). Natürlich ohne Support, Service Level Agreement oder eine Datensicherung. Auch besteht kein direkter Zugriff auf die Datenbank, so-

dass man nur mit dem SQL-Workshop in APEX mit den Datenbankobjekten arbeiten kann. Dennoch kann die Umgebung gut für die ersten Apps mit unkritischen Daten für Testzwecke verwendet werden.

Oracle Cloud Free Tier

Wer mehr Speicherplatz braucht und weiterhin auf Support verzichten kann, der kann sich die Oracle Cloud Free Tier anschauen (siehe cloud.oracle.com). Hier bekommt man sogar zwei Pluggable-Autonomous-Datenbanken mit jeweils 20 GB an Datenhaltung dauerhaft kostenlos.

So kann eine APEX-Datenbank für die Entwicklung sowie das Testen von Applikationen verwendet werden und die andere Datenbank als Produktionsumgebung. Auch hier gilt die Einschränkung bezüglich des Supports und der Datensicherung. Außerdem muss die Umgebung mindestens alle sieben Tage eine Aktivität aufweisen, ansonsten wird die Datenbank automatisch heruntergefahren. Nach drei Monaten Inaktivität wird die Umgebung gelöscht.

Mit der Oracle Free Tier bekommt man noch viel mehr Möglichkeiten quasi geschenkt. Eine Schritt-für-Schritt-Anleitung über die Möglichkeiten bieten die Blogposts [1] von Dimitri Gielis.

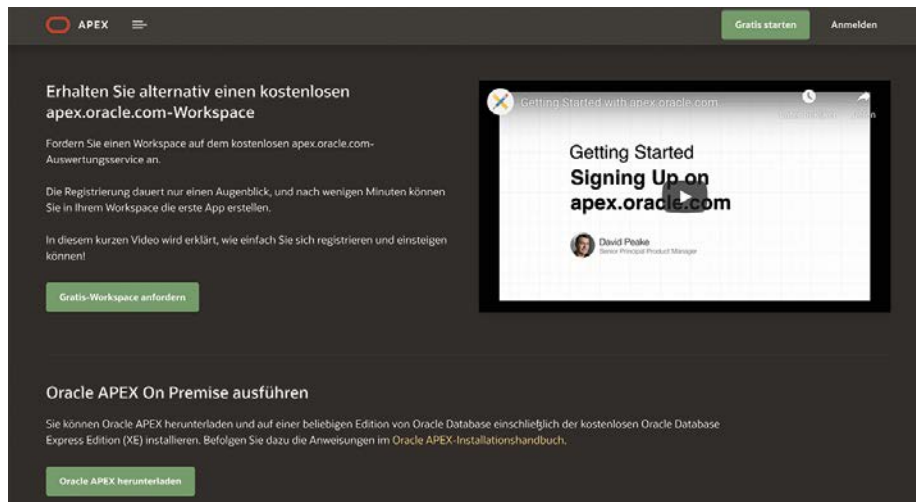


Abbildung 1: apex.oracle.com (Quelle: Oracle)

Support gefällig?

Wachsen die Anwendungen weiter, dann könnte der Schritt von Free Tier in die kostenpflichtige Oracle Cloud sich rechnen. Wer mit minimalen Kosten seine APEX-Apps inklusive Support betreiben lassen möchte, kann auch auf die Angebote von Anbietern wie Maxapex, Enciva, Revision oder Skillbuilders zurückgreifen.

Die Kontrolle behalten

Wer lieber die volle Kontrolle über die APEX-Umgebung behält, kann entweder eine vorinstallierte APEX-Umgebung hosten lassen oder die Software selbst installieren. Im letzten Fall genügt einen Hosting-Anbieter für das Betriebssystem CentOS. Das Betriebssystem kann mit einem Skript [2] zu Oracle Linux migriert werden. Wem 12 GB an Datenhaltung ausreichen und wer auf Support verzichten kann, der wird sich über die kostenlose Express Edition der Datenbank 18c freuen.

In dieser Datenbank lässt sich APEX 19.2 problemlos installieren.

On-Premises

Trotz der vielen Angebote aus der Cloud haben die größeren Unternehmen ihre APEX-Umgebungen nach wie vor im Firmennetz laufen. Die Referenzimplementierung kann man der Abbildung 2 entnehmen. Die Datenbank läuft auf einem Server getrennt vom Oracle-REST-Data-Services-Server (ORDS-Server). Als Proxyserver hat man die Wahl zwischen IIS (Windows) oder NGINX/Apache Webserver (Linux). Der Ablauf im Betrieb ist im Wesentlichen wie folgt: Eine HTTPS-Anfrage wird durch den Proxyserver meist über das binäre AJP-Protokoll an ORDS weitergeleitet. ORDS verwendet eine bestehende Datenbanksession, um eine PL/SQL-Prozedur von APEX aufzurufen. Diese baut die HTML-Seite mittels PL/SQL anhand der Metadaten zusammen und gibt diese an den Browser zurück.

Single Sign-on

Sehr empfehlenswert ist die Konfiguration von IIS (Windows) oder mod_auth_kerb (Linux), sodass die APEX-URL durch Kerberos geschützt ist [3]. Sowohl Entwickler als auch Endanwender müssen zuerst an der Windows-Domäne angemeldet sein, bevor eine Anfrage an die APEX-Datenbank weitergeleitet wird (siehe Abbildung 3). Dabei ist nach der Anmeldung am Betriebssystem auf dem Client keine Eingabe von Kontodaten mehr erforderlich. In nur wenigen Tagen hat man diese sichere APEX-Architektur geschaffen, die sowohl Ihrer IT-Security-Abteilung als auch allen Endanwendern eine Freude bereitet.

Geschützten Zugriff über das Internet ermöglichen

Wenn Entwickler und/oder Endanwender auf geschützte APEX-Applikationen über das Internet zugreifen sollen, dann stellt Abbildung 4 eine Alternative dar. In diesem Fall erfolgt zuerst die Anmeldung mittels SAMLv2 über den ADFS Proxy an der Windows-Domäne, bevor der Benutzer die APEX-Applikation aufrufen kann. Für die Implementierung [4] kann sowohl ein Apache-Webserver als auch ein Hardware Load Balancer wie beispielsweise Big IP-F5 eingesetzt werden.

Externe Benutzer werden in der Regel nicht im Firmenverzeichnis angelegt, besonders dann, wenn anonyme Benutzer sich selbst registrieren können. Microsoft bietet unter anderem für solche Szenarien Active Directory auch in der Cloud an (siehe Abbildung 5). Ein Registrierungsprozess muss dann nicht selbst implementiert werden. Das Verfahren für die Authentifizierung ist jedoch identisch wie bei Abbildung 4.

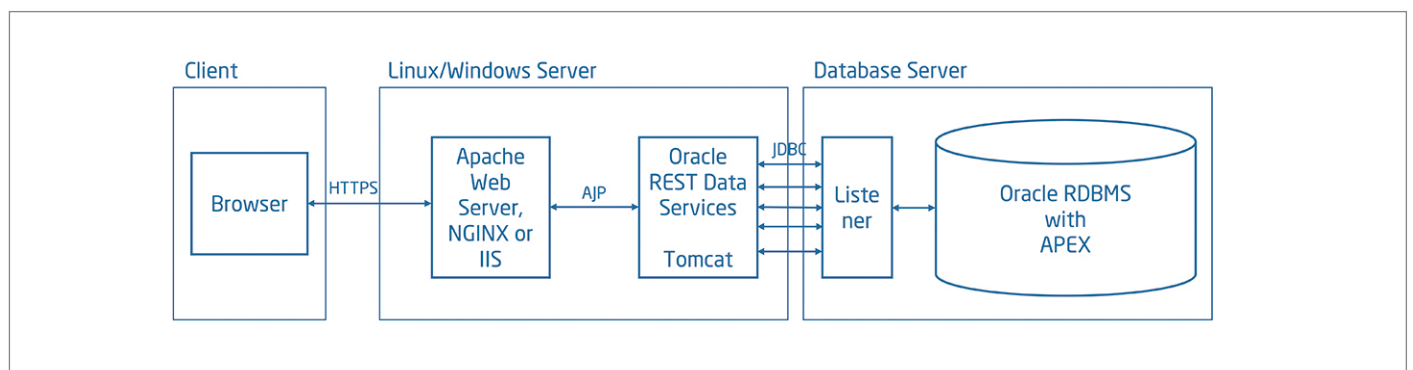


Abbildung 2: Eine APEX-Referenzarchitektur ohne Hochverfügbarkeit/Ausfallsicherheit (Quelle: Niels de Bruijn).

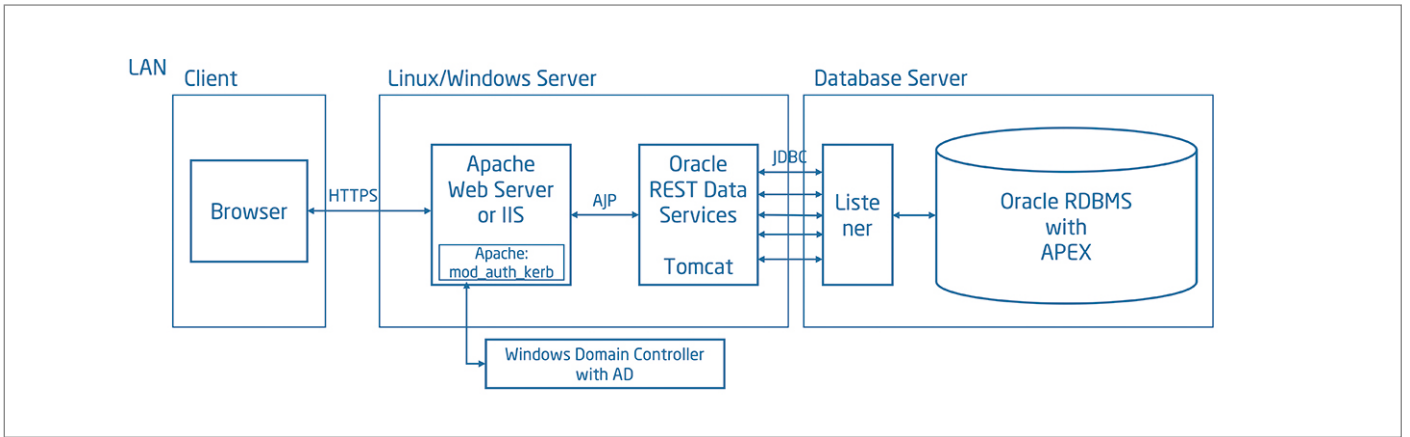


Abbildung 3: Single Sign-on auf Basis von Kerberos für interne Aufrufe (Quelle: Niels de Bruijn).

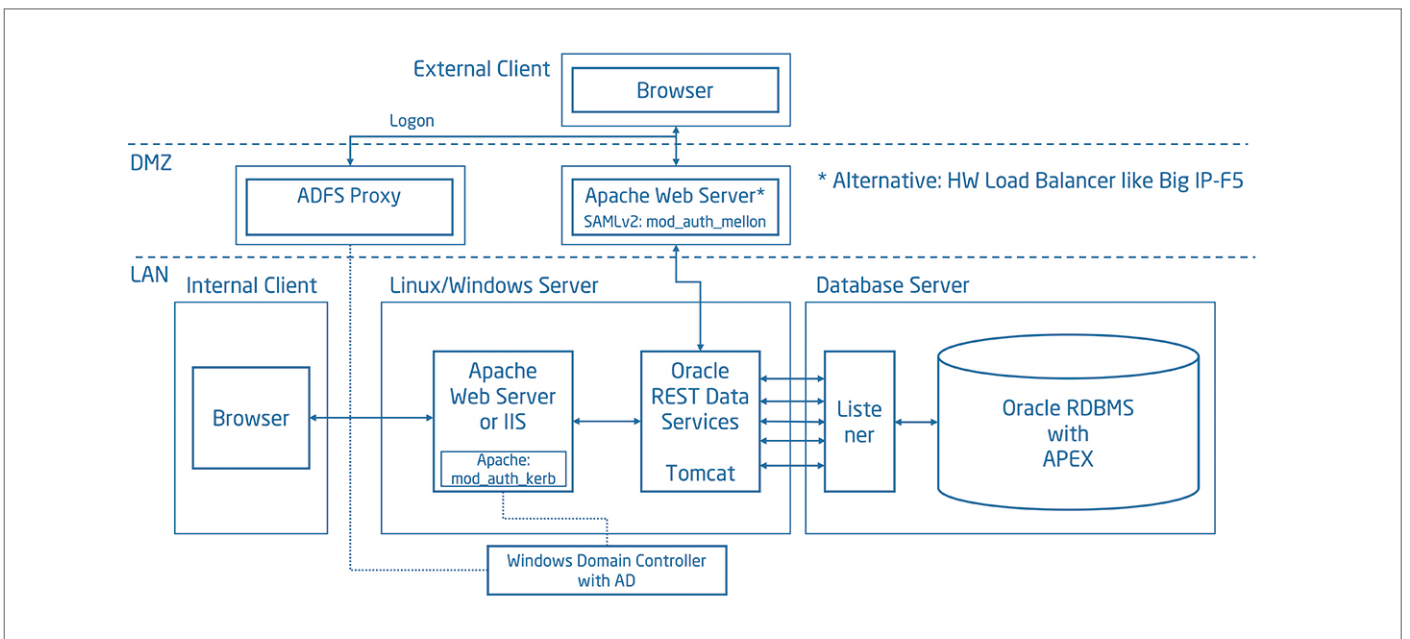


Abbildung 4: Einen sicheren Zugriff über das Internet ermöglichen (Quelle: Niels de Bruijn).

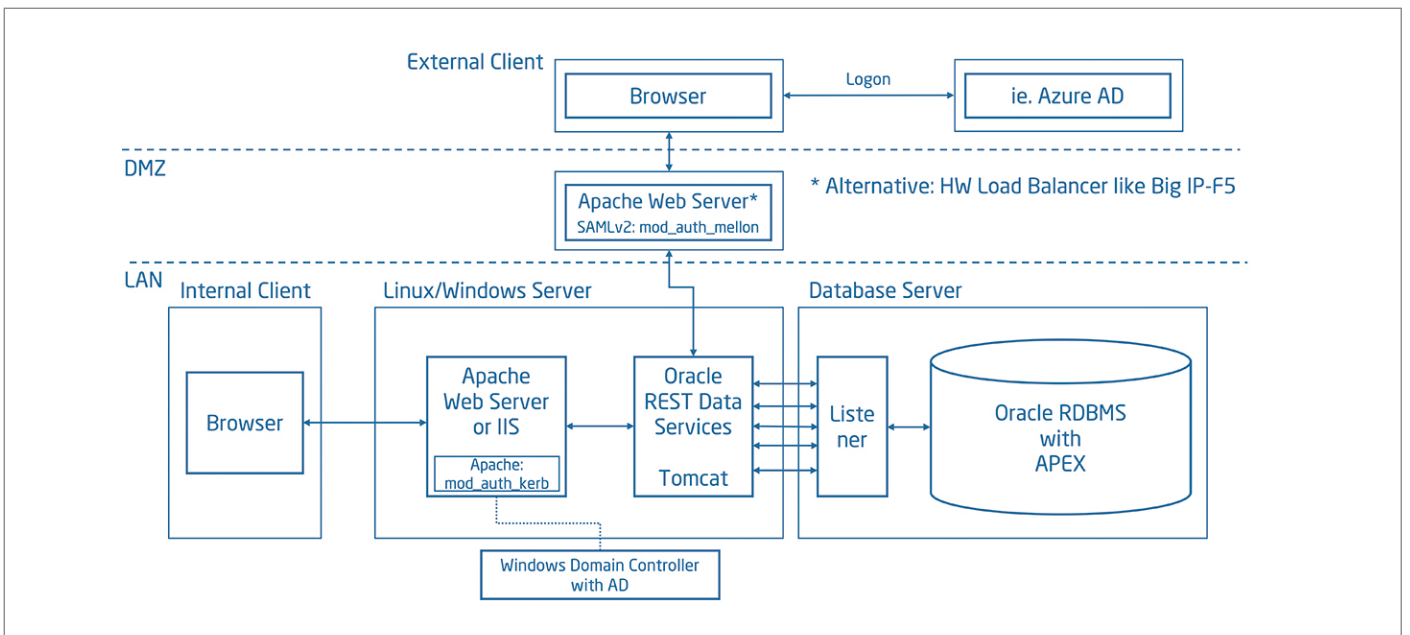


Abbildung 5: Ablage von externen Benutzern in der Cloud (Quelle: Niels de Bruijn).

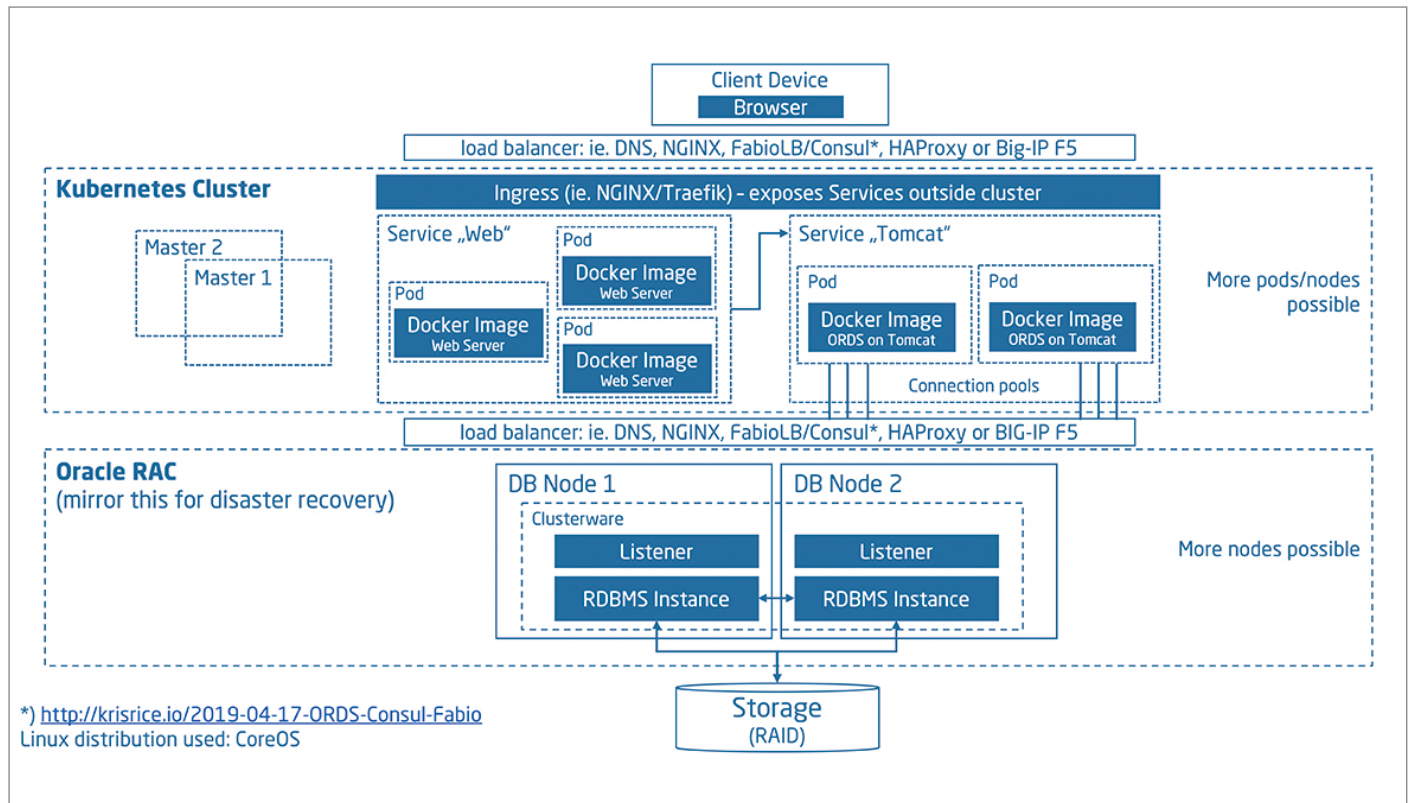


Abbildung 6: Eine mögliche 24x7-Architektur (Quelle: Niels de Bruijn).

Ausfallsicherheit

Damit die Umgebung auch bei einem Ausfall schnell und automatisch wieder zur Verfügung steht, können alle Änderungen in der Datenbank mithilfe von Oracle Data Guard (für Kunden mit einer Enterprise Edition der Datenbank) oder DBVisit auf einen anderen Server gespiegelt werden. Der ORDS-Server wird meist virtuell betrieben und kann ebenfalls aktiv-passiv laufen. Beim Ausfall wird das zweite Image gestartet und übernimmt die Dienste vom ersten Image. Beide Images brauchen keine (Session-)Daten zu synchronisieren, weil bei APEX alle Sessions in der Datenbank gespeichert werden. Sollte die Umschaltung im Fehlerfall schnell erfolgen, bekommt dies der Endanwender in der Regel nicht mit.

Maximale Verfügbarkeit

Wer global mit einer APEX-App vertreten ist und wessen Geschäft sich direkt mit der Verfügbarkeit der Umgebung bedingt, für den kommt eine sehr hochverfügbare Architektur in Betracht (siehe Abbildung 6). Auf Datenbankebene stellt ein RAC-Cluster den reibungslosen Be-

trieb sicher. Dazu werden ORDS sowie der Proxy-Server mittels Docker in einem Kubernetes-Cluster betrieben. Zwischen den Netzwerken werden Load Balancer eingesetzt. Es wird klar sein, dass diese Architektur nicht mal eben aufgesetzt ist, viel Expertise braucht und ein Vielfaches kostet. Was die Installation und den Betrieb angeht, können die großen Cloud-Anbieter den benötigten Aufwand mit fertigen Komponenten jedoch reduzieren und ermöglichen eine bedarfsgerechte Skalierung.

Fazit

Eine passende Architektur für den Betrieb von APEX-Applikationen wählt man nicht mal eben so. Das vorhandene Budget, die Zahl aktiver Benutzer, die gewünschte Ausfallsicherheit/Hochverfügbarkeit, der erforderliche Speicherplatz, die Relevanz der Daten sowie die vorhandene Oracle-Expertise sind nur einige Parameter, die eine Rolle dabei spielen. Dennoch kann mit APEX klein gestartet werden und die Architektur mit dem Geschäft mitwachsen. Single Sign-on erhöht die IT-Sicherheit und sollte bei den Überlegungen für die Architekturwahl mit betrachtet werden.

Weitere Informationen:

- [1] <http://bit.ly/apexcloudblogposts>
- [2] <https://linux.oracle.com/switch/centos>
- [3] https://knowledgebase.mt-ag.com/q/apex_sso_kerberos
- [4] https://knowledgebase.mt-ag.com/q/apex_sso_samlv2

Über den Autor

Niels de Bruijn ist Business Unit Manager Low-Code bei der MT AG und Oracle ACE Director. Er unterstützt gemeinsam mit seinem Team von APEX-Spezialisten bei der Entwicklung im APEX-Umfeld.



Niels de Bruijn
Niels.deBruijn@mt-ag.com

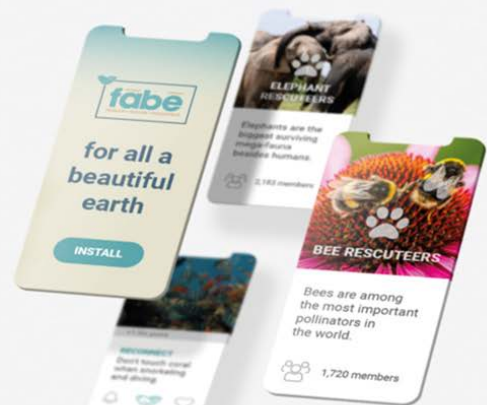


fabē

reduce • rescue • reconnect

Take concrete action
today with **the new 3Rs**

Download the fabe app and discover our library of over 350 unique actions you can take on a regular basis. Join our community and proudly share what you do to help curb climate chaos!



fabe – for all a beautiful earth

Tobias Arnhold (APEX und BI), Florian Graßhoff (DB-Entwickler), Julia Chevalier (UX-Designer)

Der Klimawandel ist nicht erst seit dem Marsch der Millionen, initiiert von Greta Thunberg, in aller Munde. Zwei bekannte Entwickler im Oracle-Umfeld haben im September 2018 einen gemeinnützigen Verein gegründet, der den Namen „fabe“ trägt. Das Ziel von fabe ist es, ein Werkzeug zu schaffen, bei dem jeder von uns die notwendigen Hilfestellungen erhält, um seinen eigenen Lebensstil aktiv zu verändern. Getreu dem Motto: „Wenn einer was macht, was nützt das schon? Aber wenn Millionen sich aktiv verändern, dann ist die Veränderung spürbar.“

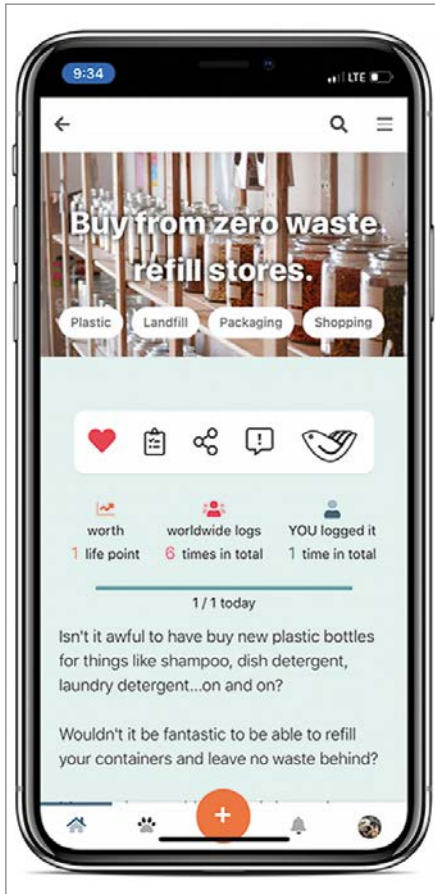


Abbildung 1: Anzeige einer Action in der App (Quelle: fabe)

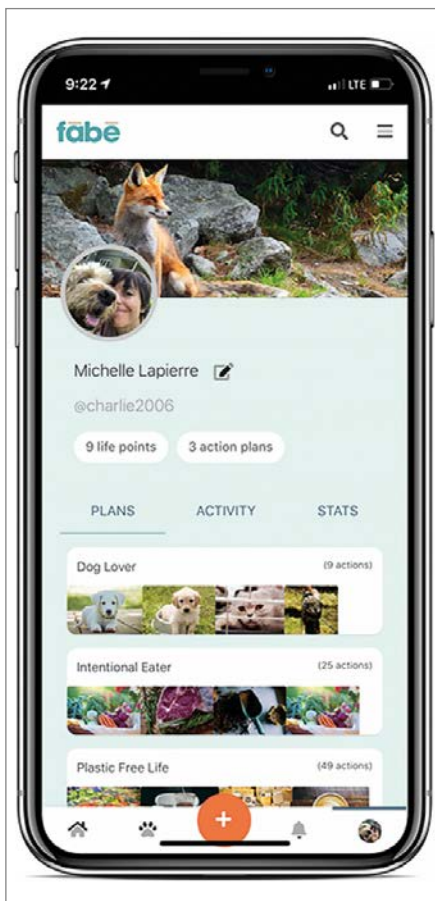


Abbildung 2: User-Profile (Quelle: fabe)

Kein Standard-APEX-Projekt

Die beiden Entwickler aus dem Oracle-Umfeld waren damals Steven Feuerstein und Vincent Morneau. Durch einen öffentlichen Aufruf an mögliche Unterstützer sind auch wir drei nach und nach zu dem Projekt dazugestoßen. Aber was produzieren wir hier eigentlich?

Der Aufruf stammte aus dem APEX-Umfeld der Oracle Community und Vincent hatte bis dato schon einige beachtliche Community-Projekte auf die Beine gestellt (Material APEX, APEX Nitro, APEX as a PWA).

fabe ist aber mit keinem dieser Projekte vom technologischen Umfang und Aufwand her vergleichbar. Es benötigte Unterstützung aus allen Reihen der IT: Administration, Datenbank-Design, PL/SQL-Entwicklung, APEX-Entwicklung, UI/UX-Design und auch Marketing sowie Datenpflege waren entscheidende Faktoren für die erste Stufe der Entwicklung.

APEX als App geht das überhaupt?

Kann mit dem APEX-Framework eine mobile Anwendung gebaut werden, die vielleicht von Millionen von Nutzern verwendet wird? Wie kann die Anwendung auch als innovative App gestaltet werden? Aller Motivation und Erfahrung zum Trotz stellte sich doch die Frage: „Können überhaupt alle komplexen Anforderungen mit APEX realisiert werden?“

Glücklicherweise haben wir durch die wachsende Aufmerksamkeit für das Projekt die Unterstützung von Oracle erhalten, sodass wir die Oracle Cloud vorerst kostenlos nutzen können. Um die Performance und Hochverfügbarkeit der APEX-Anwendung mussten wir uns daher weniger Gedanken machen.

APEX als Programmierframework für mobile Anwendungen ist bis dato nur bedingt erfolgreich gewesen. Es gibt zwar kleinere App-Projekte, aber den Anspruch, wie es die fabe-App erfordert, so eine mobile APEX-App gibt es bisher nicht.

PWA oder native App?

Nach langer Diskussion und Testphase sind wir am Ende der Version 1 zur Übereinstimmung gekommen, dass das PWA-

Konzept die richtige Entscheidung war und wir auch weiterhin von der Technologie überzeugt sind. PWA bedeutet, dass wir zunächst einmal nur APEX benötigen und keine zusätzlichen komplexen Frameworks für eine auf Web-basierte native App integrieren müssen. Die PWA-Webseite kann selbst verschiedene App-Features nativ verwenden. Die großen Hersteller unterstützen PWAs innerhalb der App Stores bisher nur zu Teilen: Android ja, Apple bald.

Bis dahin muss die Anwendung unter iOS noch etwas umständlicher im Vergleich zum gewohnten App-Store installiert werden.

Was bedeutet APEX-App jetzt schlussendlich?

Die Anwendung, gehostet in der Oracle Cloud, betrieben mit ORDS und Oracle 19c, verwendet die aktuellste Version von APEX. So viel zum Standard. Die APEX-Anwendung selbst verwendet als Basis ein eigenes Theme und ist dadurch nur bedingt mit einer normalen APEX-Anwendung vergleichbar. Warum ist das so?

Die typische APEX-Anwendung im Universal Theme hat den Fokus auf der Verwendung als Business-Anwendung, die zumeist am PC genutzt wird. Die fabe-App hat die Aufmerksamkeit auf das Handy gerichtet und die Navigations- und Designentscheidungen daraufhin fokussiert.

Ein paar Features, die bei der Fokussierung als App komplett neu implementiert werden:

- UX/UI-optimierte Interaktion für die spezifischen fabe-Aufgaben
- Interaktive Page-Navigation (Zero Latency)
- Client Side Single Sign-on (Always Online)
- Bildverwendung und -optimierung (NodeJS-Implementierung)
- Frontend Testing mit cypress.io
- Datenmodell mit optimierten Multi-Language und EBR für Zero-Downtime bei Updates

Wie funktioniert die Entwicklung?

Die einzelnen Entwickler haben dedizierte Aufgaben, wodurch versucht wird, möglichst wenige Überschneidungen in-

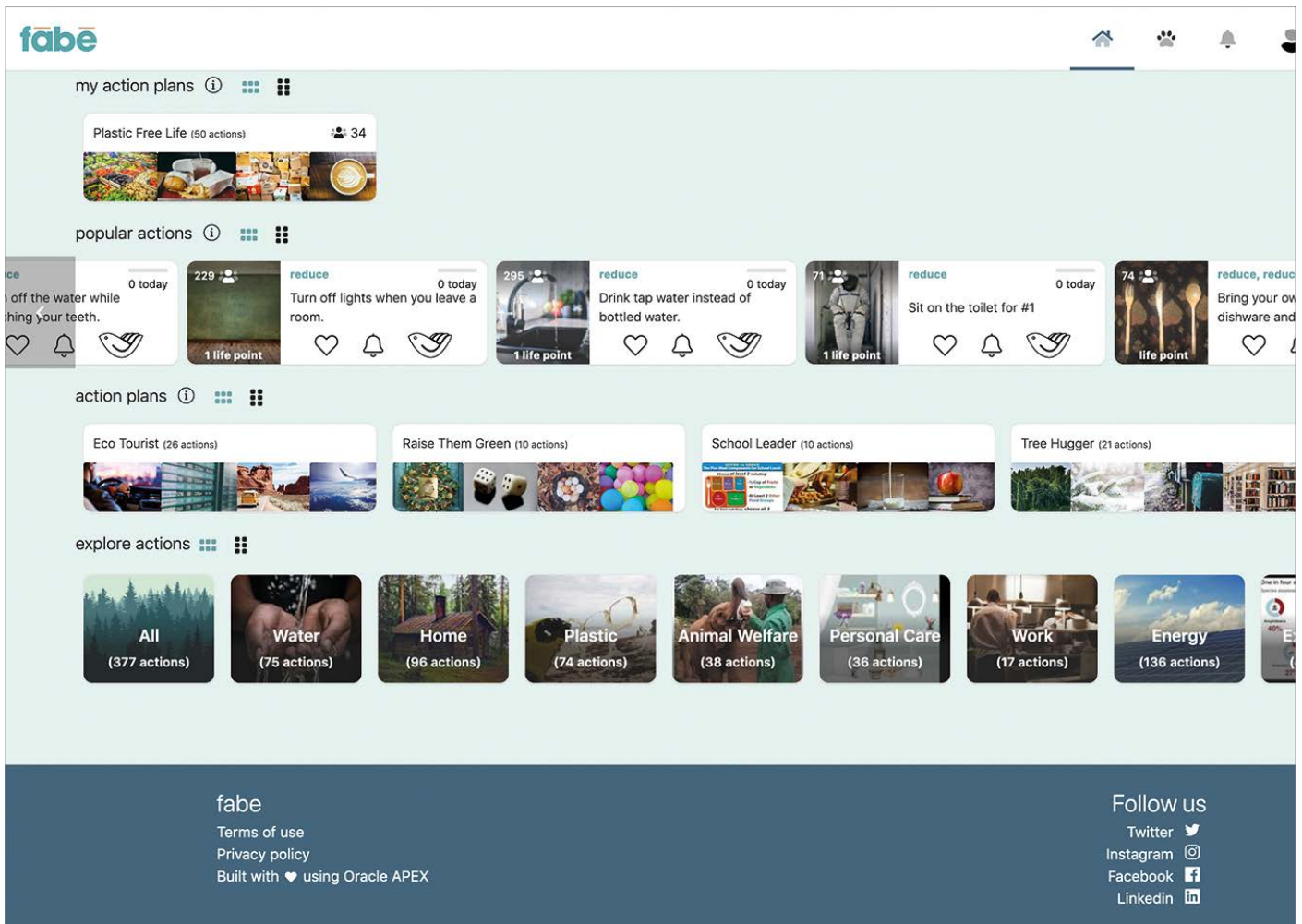


Abbildung 3: Startseite der Anwendung auf einem Tablet oder Desktop (Quelle: fabe)

nerhalb eines Sprints zu verursachen. Besonders spannend ist, dass die Entwickler weltweit verstreut sitzen und niemand zu festen Arbeitszeiten zur Verfügung steht. Zur Erinnerung: Fast alle im fabe-Projekt unterstützen das Projekt in ihrer Freizeit. Die Kommunikation ist daher ein wichtiger Faktor, um sich nicht in die Quere zu kommen. Gleichzeitig laufen die Implementierungen anhand der verschiedenen Merge Requests in einzelnen Branches zum Master zusammen.

Wie geht es weiter?

Der Hauptfokus liegt aktuell auf der Implementierung von Social-Media-Aspekten und der gemeinsamen Interaktion mit Freunden. Die Anwendung befindet sich aufgrund der agilen Entwicklung in ständiger Veränderung und Verbesserung, wodurch die Implementierung des finalen Datenmodells ebenfalls noch aussteht.

Neben der technischen Weiterentwicklung ist der Faktor Marketing und die Verbreitung der App ein sehr wichtiges Thema.

Fazit

fabe ist ein sehr komplexes APEX-Projekt und wird bisher fast ausschließlich von ehrenamtlichen Mitarbeitern umgesetzt. Es geht nicht darum, Millionen an Euros oder Dollars zu verdienen, sondern Millionen davon zu überzeugen, ihre Lebensweise auch mit kleinen Dingen im Leben zu verändern. Ob die App schlussendlich ein Erfolg wird, hängt von vielen Faktoren ab. Wir drei bleiben dabei und versuchen, das gemeinnützige Projekt fabe zum weltweiten Erfolg zu bringen.

Wichtige Links

- fabe: <https://forallbeautiful.earth/>
- fabe-Team: <https://forallbeautiful.earth/home/fabe-team/>

- APEX-App als PWA: <https://github.com/vincentmorneau/apex-pwa>
- Oracle Cloud: <https://www.oracle.com/de/cloud/>
- Cypress.io: <https://www.cypress.io>



Tobias Arnhold
tobias-arnhold@hotmail.de

Julia Graßhoff
jgrasshoff@fab.earth

Florian Graßhoff
fgrasshoff@fab.earth



Spielerisch leichte Security für Oracle Datenbanken – komplexe Oracle Features genial einfach nutzen

Thomas Petrik und Wolfgang Klinger, Sphinx IT Consulting

Die Oracle Database Enterprise-Edition enthält eine Fülle von Sicherheits-Features, deren Nutzung enormen Zeitaufwand bedeutet. Aus diesem Grund werden viele Features nicht verwendet. Durch Automatisierung von Provisionierung und Administration gelingt es dem Security Framework SCURTY, praktisch alle Security-Features mit wenig Aufwand nutzbar zu machen. Virtual Private Database (VPD), Application Context, Secure Application Roles, Proxy Authentication, Datenbank-Trigger u.v.m. stehen plötzlich zur komfortablen Verwendung zur Verfügung.

Die Oracle Database Enterprise-Edition enthält viele hervorragende Sicherheits-Features ohne zusätzlichen Aufpreis. Die Implementierung der Features bedeutet jedoch oft hohen administrativen Aufwand. Ausgezeichnete Funktionen wie VPD (Virtual Private Database) und viele andere werden aufgrund der Wartungsintensität selten produktiv verwendet. Das ist nicht nur schade – es ist vor allem auch schädlich, wenn Bedrohungen nicht entsprechend begegnet werden kann.

Wer schon einmal VPD-Policies in größerem Umfang verwendet hat, weiß, wovon ich spreche. Die Übersicht zu bewahren ist nicht einfach und der administrative Aufwand nimmt schnell Dimensionen an, die die Standard-Security mit klassischen Grants plötzlich wieder sehr attraktiv erscheinen lassen.

Genau hier setzt SCURTY an: Das Produkt der Sphinx IT Consulting GmbH realisiert die vollständige Automatisierung vieler Sicherheits-Features der Oracle Database Enterprise-Edition. SCURTY verbirgt die Komplexität der dahinterliegenden Funktionen und ermöglicht damit den Einsatz höherer Sicherheits-Level in Bezug auf Authentifizierung und Datenzugriff. Der administrative Aufwand wird enorm reduziert, was die Sicherheits-Features oftmals erst verwendbar macht. Die nötigen Policies, Rollen und anderen Datenbank-Objekte generiert und verwal-

tet SCURTY im Hintergrund automatisch. Eine manuelle Wartung dieser Objekte entfällt komplett.

SCURTY enthält ein komfortables PL/SQL-API, über das alle Definitionen vorgenommen werden können. Es ist nicht nötig, Detailwissen über die dahinterliegenden Funktionen wie VPD, Proxy Authentication etc. zu besitzen. Es genügt, das SCURTY API der Dokumentation entsprechend zu verwenden, um Datenbereiche effektiv zu schützen und Zugriffsberechtigungen punktgenau zu vergeben. Oder auch um Sandbox-Umgebungen zu erstellen. Die Dokumentation liefert alle nötigen Informationen auf einer höheren Ebene, nämlich jener der Funktionalität.

Die Funktionalität von SCURTY

Eine Kernfunktionalität von SCURTY ist es, Datenbereiche (in erster Linie Tabellen oder ganze Schemata) festzulegen, die geschützt werden sollen. Auf diese Datenbereiche können feingranulare Zugriffsrechte vergeben werden.

An diesem Punkt entstehen üblicherweise Bedenken bezüglich des damit verbundenen Aufwands – aber genau dieser wird durch SCURTY, wie oben beschrieben, enorm reduziert. Der größte

Aufwand liegt in den Überlegungen, die angestellt werden müssen, um ein sinnvolles Berechtigungskonzept zu erstellen. Die Implementierung ist mit SCURTY dann vergleichsweise rasch erledigt, weil keine manuelle Tätigkeit – ausgenommen der Aufruf des APIs – anfällt.

Ist ein geschützter Datenbereich definiert, existieren aber noch keine in SCURTY vergebenen Zugriffsrechte, sind für die User auch keine Daten sichtbar. Die normalen Berechtigungen (zum Beispiel SELECT-Rechte) erlauben keinen Zugriff auf geschützte Bereiche.

Ein weiteres wesentliches Merkmal von SCURTY ist die Personalisierung der Accounts. Über Proxy Authentication und die Verwendung von Application Context können auch Applikations-User von durchgehender Personalisierung profitieren. Um entsprechende Berechtigungen zu gewähren und sinnvolles Audit zu ermöglichen, wird intern immer die Information verwendet, um welche physischen User es sich handelt.

Feingranulare Zugriffsrechte ohne viel Aufwand

Daten werden mit SCURTY auf einfache Weise für Zugriffe freigeschaltet, wobei die Granularität bis auf Zeilen- und Spaltenebene einstellbar ist. Die Berech-

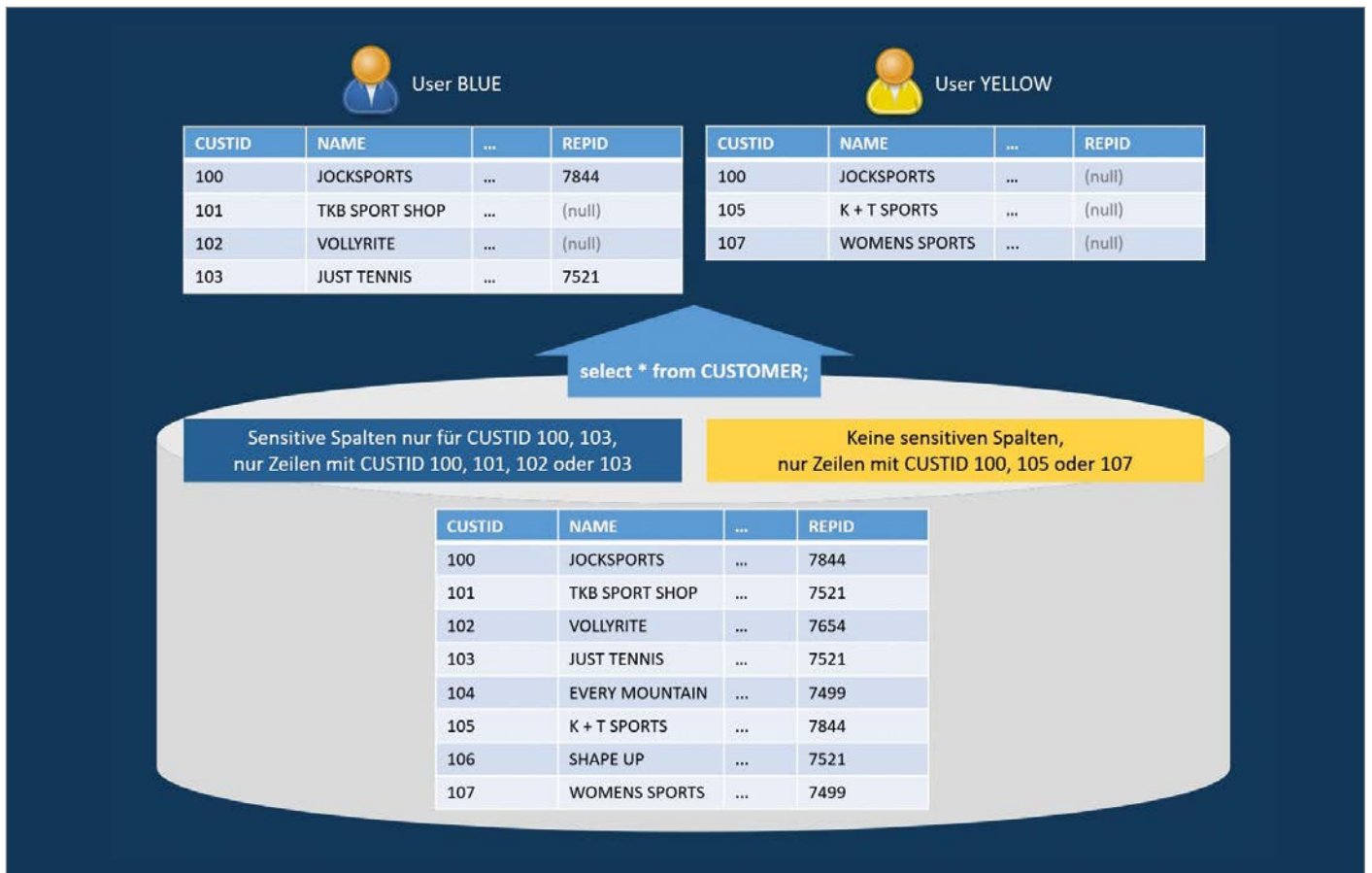


Abbildung 1: Daten einer sensitiven Spalte können ausgeblendet oder abhängig von den Inhalten einer anderen Spalte teilweise angezeigt werden (Quelle: Wolfgang Klinger)

tigungen gehen also weit über die klassischen SELECT-, INSERT-Privilegien etc. hinaus. Spalten können als sensitiv gekennzeichnet werden und gleichzeitig kann auch auf Zeilenebene über das Zugriffsrecht entschieden werden. Werte können bei Bedarf abhängig von Dateninhalten ausgeblendet werden beziehungsweise stehen nur für bestimmte User zur Verfügung (siehe Abbildung 1). Das klingt aufwendig, ist es mit SCURTY aber nicht, weil der arbeitsintensive Teil automatisiert im Hintergrund abläuft.

Beispiel für feingranulare Rechte-Vergabe über API-Aufrufe

Das klingt alles sehr abstrakt. Um ein Gefühl für die Praxis zu bekommen, hier ein Beispiel:

User WILLIAMS besitzt eine Tabelle (siehe Listing 1), von welcher User YELLOW 4 Zeilen sehen soll, aber nur bei zwei Zeilen dürfen auch sensitive Spalten angezeigt werden. User BLUE darf drei andere

Zeilen sehen und überhaupt keine sensitiven Spalten. Die Spalte REPID in diesem Beispiel wurde zuvor als sensitive Spalte gekennzeichnet.

User YELLOW erhält Zugriff auf die CUSTIDs 100,101,102 und 103. Für 100 und 103 sollen auch sensitive Spalten sichtbar sein (letzter Parameter = 'TRUE'), in diesem Fall ist das die Spalte REPID (siehe Listing 2). "U\$WILLIAMS" und "CUSTOMER" sind Referenzen auf existierende Rollen und Objektgruppen, die die Tabelle CUSTOMER enthalten. Die Namen dieser Objekte sind in den Reporting-Tabellen und -Views von SCURTY zu finden.

User YELLOW hat damit entsprechenden Zugriff auf die Daten (siehe Listing 3).

User BLUE erhält Zugriff auf die CUSTIDs 100, 105 und 107 und darf überhaupt keine sensitiven Spalten sehen (letzter Parameter = 'FALSE') (siehe Listing 4).

User BLUE hat damit entsprechenden Zugriff auf die Daten (siehe Listing 5).

So weit, so einfach. Was in der Ausführung einfach aussieht, ist im Hintergrund sehr umfangreich, es läuft unbemerkt eine Fülle von Prozessen ab:

- Mehrere Checks: u.a. ob jemand das Recht sich selbst zuteilen möchte oder ob die korrekte Klassifizierung für die Person in SCURTY besteht (das Recht kann nicht an Applikations-User vergeben werden) etc.
- Erstellen von Secure Application Roles
- Vergabe der Berechtigung auf die Rollen
- Erstellen von Virtual Private Database Policies
- Protokollierung des Aufrufs

Wie in diesem Beispiel zu sehen ist, sind über das API nur die für die Rechte-Vergabe essenziellen Daten einzugeben. Alle anderen, durchaus komplexen Funktionen und Schritte, werden im Hintergrund vollständig automatisiert ausgeführt.

Dezentralisierte Provisionierung

Durch das Verbergen der Komplexität wird es möglich, die Provisionierung an die Fachabteilung auszulagern. Das hat

zwei wesentliche Vorteile: Einerseits wird die DBA-Gruppe entlastet und andererseits kennt die Fachabteilung die genauen Anforderungen selbst am besten und kann ohne zeitliche Verzögerung agieren. Die Fachabteilung ist autonom und flexibel für ihren Bereich unterwegs – ohne Abhängigkeit von DBAs, aber auch ohne die Tätigkeit der DBAs durch eventuelle Fehler negativ beeinflussen zu können.

SCURTY enthält eine interne Berechtigungs-Struktur mit entsprechenden Rollen, die für die Rechte-Vergabe oder das Anlegen neuer User genutzt werden können. Es sind also keine DBA-Privilegien nötig, um die entsprechenden Definitionen vorzunehmen. Damit ist auch sichergestellt, dass es zu keinen Kollisionen zwischen DBAs und Fachabteilung kommt.

Innerhalb von SCURTY ist eine Trennung der Verantwortlichkeiten realisiert, die genutzt werden kann. So sind zum Beispiel das Anlegen neuer User oder neuer Datenbereiche rollenmäßig getrennt von der Vergabe der Rechte oder auch von Reporting-Funktionen. Diese Trennung in mehrere Bereiche existiert; wenn es so gewünscht wird, können die Funktionen aber auch in einer Person zusammenfließen.

Personalisierter Zugriff

Three Tier-Architekturen verwenden häufig Applikations-User, wodurch die Rechte-Vergabe schwierig und ein Audit-Trail bedeutungslos werden. Oft werden Zugriffsrechte durch Applikationen selbst abgehandelt. Wenn aber die Applikations-User außerhalb der Applikation verwendet werden, dann ziehen diese Berechtigungen nicht, und die Daten liegen offen. Die Audit-Daten enthalten in diesem Fall keinen Hinweis auf die wirklichen User hinter dem Keyboard, sondern immer nur die Kennung der Applikations-User.

Personalisierte User-Accounts sind aber die Basis für gute Datenbank-Sicherheit. Ohne eine Form der Personalisierung gibt es keine Möglichkeit, die Datenbank-Sicherheitsfeatures umfassend zu nutzen. Applikationen können jedoch nicht einfach neu geschrieben werden, um dieses Ziel zu erreichen.

Mit Proxy Authentication kann trotz Verwendung der Applikations-User die

```
sqlplus williams/*****@localhost:1522/pdb
select custid, name, repid from williams.customer;
```

| CUSTID | NAME | REPID |
|--------|----------------|-------|
| 100 | JOCKSPORTS | 7844 |
| 101 | TKB SPORT SHOP | 7521 |
| 102 | VOLLYRITE | 7654 |
| 103 | JUST TENNIS | 7521 |
| 104 | EVERY MOUNTAIN | 7499 |
| 105 | K + T SPORTS | 7844 |
| 106 | SHAPE UP | 7521 |
| 107 | WOMENS SPORTS | 7499 |

Listing 1: Abfrage der Tabelle von User WILLIAMS

```
exec p_rep_oper.grant_user_tnt_access(
'YELLOW',
'U$WILLIAMS',
'CUSTOMER',
SCURTY_t_lov('100', '103'),
TRUE);

exec p_rep_oper.grant_user_tnt_access(
'YELLOW',
'U$WILLIAMS',
'CUSTOMER',
SCURTY_t_lov('101', '102'),
FALSE);
```

Listing 2: Erteilen der Zugriffsrechte für User YELLOW. (Ausgeführt als User mit SCURTY "OPERATOR" oder DBA-Rechten.)

```
sqlplus yellow/sx123@localhos t:1522/pdbb
col name format a50
select custid, name, repid from williams.customer;
```

| CUSTID | NAME | REPID |
|--------|----------------|-------|
| 100 | JOCKSPORTS | 7844 |
| 101 | TKB SPORT SHOP | |
| 102 | VOLLYRITE | |
| 103 | JUST TENNIS | 7521 |

Listing 3: User YELLOW sieht vier Zeilen der Tabelle, wobei in zwei Zeilen die sensitive Spalte REPID ausgeblendet ist.

```
exec p_rep_oper.grant_user_tnt_access(
'BLUE',
'U$WILLIAMS',
'CUSTOMER',
SCURTY_t_lov('100','105','107'),
FALSE);
```

Listing 4: Erteilen der Zugriffsrechte für User BLUE. (Ausgeführt als User mit SCURTY "OPERATOR" oder DBA-Rechten.)

```
sqlplus blue/*****@localhost:1522/pdb
select custid, name, repid from williams.customer;
```

| CUSTID | NAME | REPID |
|--------|---------------|-------|
| 100 | JOCKSPORTS | |
| 105 | K + T SPORTS | |
| 107 | WOMENS SPORTS | |

Listing 5: User BLUE sieht drei Zeilen der Tabelle, aber ohne Werte in der sensitive Spalte REPID.

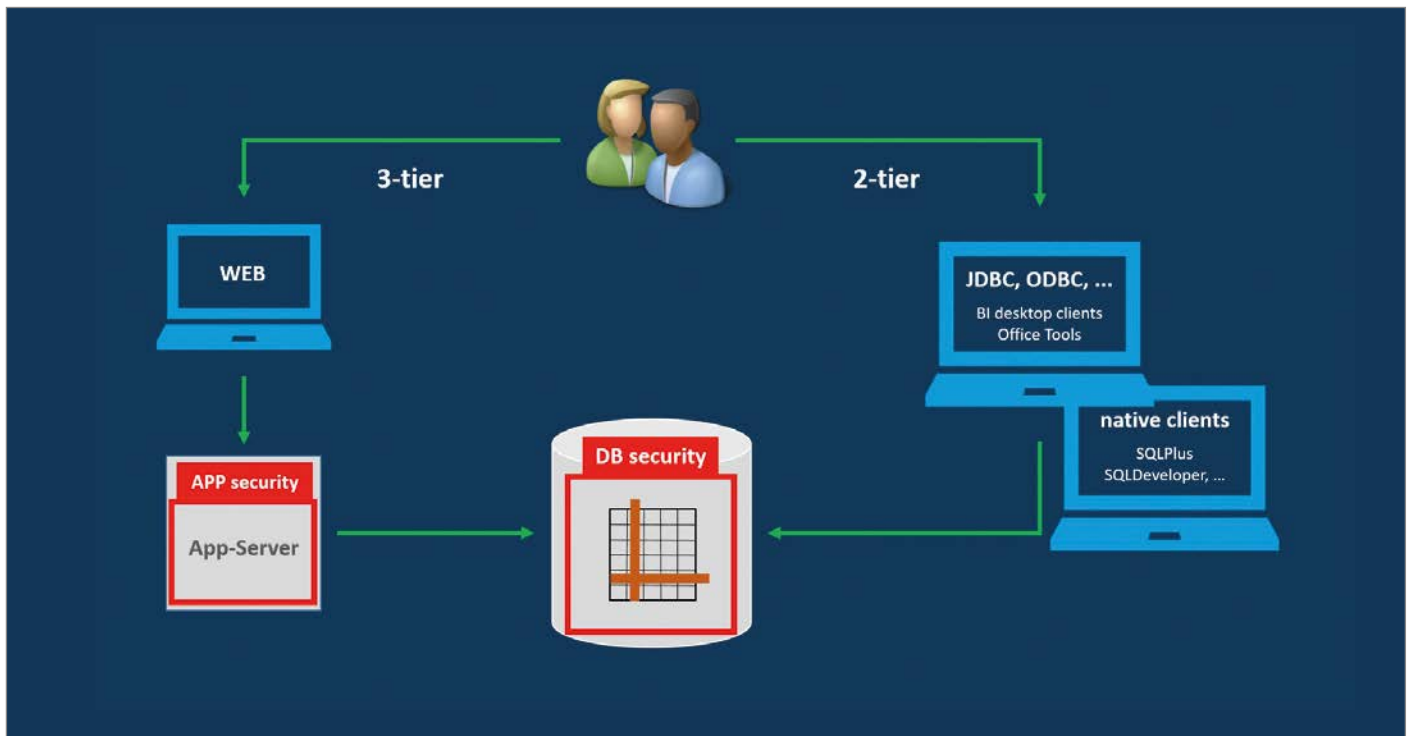


Abbildung 2: Sowohl Zwei- als auch Drei-Tier Architekturen werden unterstützt (Quelle: Wolfgang Klinger)

Identität der physischen User an die Datenbank übermittelt werden. User erhalten dadurch zwei Identitäten gleichzeitig: jene des Applikations-Users und die eigene – und sind damit jederzeit eindeutig identifizierbar. Dadurch können alle Sicherheitsmechanismen der Datenbank eingesetzt werden und die Audit-Daten enthalten die echte Identität der User.

Ein weiterer Vorteil der Proxy Authentication: Nur die Kombination der beiden User führt zum Ziel, wodurch ein höherer Level an Sicherheit erreicht wird. Die Applikations-User haben die Rechte auf die Datenbank-Objekte, aber keines für Login. Die physischen User haben Login-Rechte, aber keine Rechte auf die Datenbank-Objekte. Nur durch das Anmelden als Proxy-User, bei dem beide Kennungen in einem Login verwendet werden, wird das Login und die Verwendung der Daten möglich.

Unabhängig davon, ob Proxy Authentication verwendet wird oder direkte Connects stattfinden, wird immer der Application Context verwendet, um die Identität der User zur Verfügung zu haben.

Sicherheit, die in der Datenbank hinterlegt ist, kann nicht umgangen werden. Auch deshalb ist personalisierte Rechtevergabe von zentraler Bedeutung. Die Berechtigungen sind damit immer die Gleichen, unabhängig davon, welches

Tool oder welche Architektur (Zwei- oder Drei-Tier) eingesetzt werden (siehe Abbildung 2).

Proxy Authentication ist Teil von SCURTY und wird ebenfalls über das PL/SQL API eingerichtet. Wenn ein User nicht provisioniert wurde, ist kein Login möglich. Extern angelegte User, die in SCURTY nicht provisioniert wurden, haben auf die geschützten Datenbereiche keinen Zugriff.

Ein weiteres Feature: Sandboxing

Rasch eine Umgebung für Tests zu erhalten ist oft nicht einfach. Mit SCURTY können Sandbox-Bereiche für Test, Entwicklung und Analyse angelegt werden. Eine Sandbox wird hier als temporäres Datenbank-Schema verstanden, auf das alle oben beschriebenen Sicherheitsmechanismen angewandt werden können. Auf Objekte in der Sandbox kann anderen Usern Zugriff gestattet werden, ausgenommen solche Objekte, die als geschützt ("protected") markiert wurden. Auf geschützte Objekte haben nicht einmal Sandbox Master oder Sandbox Owner Zugriff, sondern nur jene Person, die selbst das Objekt erstellt hat.

Jede Sandbox hat einen Lebenszyklus und wird irgendwann obsolet, das zeigt

die Erfahrung. Eine in SCURTY angelegte Sandbox ist zwingend mit einer Lebensdauer zu versehen. Nach Ablauf dieser Lebensdauer wird die Sandbox automatisch gesperrt und später, nach einer zusätzlichen Grace-Period, automatisch gelöscht. Auf diese Weise wird der übliche Daten-Wildwuchs verhindert.

Wie ist SCURTY implementiert und wie wird es installiert?

SCURTY selbst wurde von der Sphinx IT Consulting GmbH in PL/SQL geschrieben und läuft in der Oracle Database Enterprise-Edition ab Version 12c. Die Installation erfolgt über automatisierte Skripts und ist in wenigen Minuten abgeschlossen. DBA-Berechtigungen sind für die Installation und das Anlegen des ersten Administrator-Accounts in SCURTY nötig. Danach kann von diesem Account aus die weitere Einrichtung erledigt werden, DBAs sind nicht mehr zwingend nötig.

Meist werden weitere Administrator-Accounts angelegt, die jeweils einen, mehrere oder alle Berechtigungsbereiche innerhalb SCURTYs abdecken können. Wie viele Berechtigungen einzelne Administrator-Accounts bekommen, kann frei entschieden werden. Eine Trennung der Ver-

antwortlichkeiten ist möglich, aber nicht zwingend zu implementieren.

Die Installation ist stark parametrisierbar, um beispielsweise Namenskonventionen des Unternehmens bei der Benennung von Accounts und Objekten berücksichtigen zu können.

Nach der Installation gilt es, geschützte Datenbereiche zu definieren und User anzulegen sowie mit Rechten zu versehen. Ab diesem Punkt können User zu arbeiten beginnen und die Sicherheitsmechanismen von SCURTY, beziehungsweise der Oracle Database, wachen im Hintergrund über das Geschehen.

Jeder Konfigurationsschritt wird in SCURTY selbst mitprotokolliert und ist damit auch immer nachvollziehbar. Zusätzlich gibt es Datenbank-Views, die existierende Definitionen für Reporting-Zwecke auflisten.

Ein bestehendes System kann einfach exportiert und in eine andere SCURTY-Installation übernommen werden. Dabei werden die gesammelten API-Aufrufe in

Skript-Dateien geschrieben, die am Zielsystem ausgeführt werden können. Diese Skripts enthalten alle Definitionen von Objekten, User-Accounts, Berechtigungen etc., sodass eine SCURTY-Implementierung rasch auf ein weiteres System kopiert werden kann. Dieser Vorgang ist zum Beispiel beim Wechsel von einem Test- oder Entwicklungssystem auf ein Produktivsystem von Nutzen.

Fazit

Die Oracle Database Enterprise-Edition liefert viele gute Sicherheits-Features wie Virtual Private Database (VPD), Secure Application Role, Proxy Authentication etc., deren Einsatz relativ hohen Wartungsaufwand bedeutet. Das hier vorgestellte Tool reduziert diesen Aufwand durch vollständige Automatisierung der dafür nötigen Wartung enorm. Damit wird die Verwendung der Features für viele Firmen ein realistisches Szenario.



Thomas Petrik
thomas.petrik@sphinx.at



Wolfgang Klinger
wolfgang.klinger@sphinx.at

MUNIQSOFT
CONSULTING

Consulting

Hochverfügbarkeit mit IQ

Sicherheit vor teuren Ausfallzeiten:

Mit dem richtigen Konzept sind Ihre Daten und Server vor Systemausfällen optimal geschützt.

Nutzen Sie die Erfahrung der Muniqsoft Consulting GmbH
www.muniqsoft-consulting.de

ORACLE Gold Partner

Specialized
Oracle Database



Jetzt Beratungstermin vereinbaren:
+49 89 62286789-39



Modernes Datenmanagement mit der Open-Source-basierten Daten- und KI-Plattform IBM Cloud Pak for Data

Nadine Brehm und Claus Huempel, IBM

Im letzten Jahr haben wir IBM Cloud Pak for Data [1] vorgestellt, eine robuste End-to-End-Lösung für die Datenanalyse in Unternehmen. Ein Grundprinzip des Designs dieser Plattform ist es, den Zugang zu allen dem Unternehmen verfügbaren Datenquellen sicherzustellen, sowohl On-Premises als auch in der Cloud.

IBM Cloud Pak for Data ist eine moderne, Open-Source-basierte Daten- und KI-Plattform, die Daten-, Analyse- und Microservices auf Basis einer Cloud-nativen Architektur beinhaltet. Es braucht keine großen Anpassungen, da die Software vorkonfiguriert ist. Unternehmen können damit auf ihre verteilten Daten im

Unternehmen zugreifen, diese physisch oder virtuell sammeln, organisieren und analysieren. Dadurch erhalten sie auch die notwendige Grundlage für KI-Anwendungen. Im Endeffekt ist IBM Cloud Pak for Data eine Multicloud-fähige Daten- und KI-Plattform, welche die grundlegende Systemarchitektur und Services

für jede Form von Künstlicher Intelligenz (KI) liefert.

Ein wesentlicher Vorteil der Lösung ist, dass zahlreiche bestehende Lösungen, etwa Datenbanken wie Db2, MongoDB oder Cockroach, einfach integriert werden können. Somit ergänzt das System bestehende Legacy-Systeme und vereinfacht den Unternehmen die notwendige Informationsarchitektur für Künstliche Intelligenz. Einige Unternehmen haben auch weder die Zeit noch die erforderlichen Ressourcen, um alle notwendigen Hardware- und Software-Anforderungen für den Aufbau einer KI-Lösung zu erfüllen. Hier hilft das neue IBM Cloud Pak for Data System, damit auch diese Unternehmen rasch von den Vorteilen von KI profitieren können. Mit einem neuen Release gehen wir nun noch einen Schritt weiter.

IBM Cloud Pak for Data System Version 1.0

Das neue IBM Cloud Pak for Data System [2] ist eine Intel-x86-basierte, hyperkonvergente Daten- und KI-Plattform. Sie umfasst alle Möglichkeiten von IBM Cloud Pak for Data, ohne dass Unternehmen dafür ihre Hard- oder Software-Infrastruktur anpassen müssten. Das System kombiniert Speicherung, Verarbeitung, Netzwerk und Software in einem einsatzfertigen Gesamtpaket.

Ein wesentliches Ziel des IBM Cloud Pak for Data System ist es, KI-Funktionen gebrauchsfertig, also out of the box zu liefern. Deshalb ist zum Beispiel auch Watson Studio als Open-Source-basierte Lösung für Data Scientists in der Konfiguration dabei. Watson Machine Learning und der Watson Machine Learning Accelerator sind auf Wunsch ebenfalls verfügbar. Außerdem ist ein umfassendes End-to-End Data Science Toolkit enthalten, das Data Scientists aller Qualifikationsstufen dabei unterstützt,

- Daten vorzubereiten,
- KI-Modelle zu bauen, sowohl für Machine Learning (ML) als auch für Deep Learning (DL),
- KI-Modelle zu trainieren, entweder interaktiv oder mithilfe von Batch-Paradigmen,
- KI-Modelle zu betreiben und deren Lebenszyklus zu managen,



Abbildung 1: IBM Cloud Pak for Data System (Quelle: IBM)

- GPU-Beschleunigung zu nutzen, um KI-Modelle zu trainieren,
- GPUs und Field Programmable Gate Arrays (FPGAs) für Folgerungen zu verwenden
- sowie die unternehmensweite Verteilung von ML/DL-Modellen zu unterstützen.

Auf der modernen Daten- und KI-Plattform können auch die IBM Watson Services, die bisher nur in der IBM Cloud verfügbar waren, vor Ort hinter der Firewall oder auf anderen Clouds genutzt werden. Als Watson Services sind unter anderem Watson Knowledge Studio, Speech to Text, Text to Speech und natürliche Spracherkennung enthalten. Außerdem kann man zusätzlich das KI Fairness Tool Watson OpenScale [3] installieren sowie den Watson Assistant und Watson Discovery.

Warum man sich IBM Cloud Pak for Data System anschauen sollte

Was IBM Cloud Pak for Data besonders macht, ist der modulare Ansatz von Da-

tenverarbeitung, Netzwerkfunktionen und Speicherung auf Standard-Hardware – damit ist ein Baukastensystem mit einheitlichem Management möglich.

Zusätzlich zu ihrem Plug-and-Play-Ansatz bietet die Lösung auch „Plug-and-Grow“-Funktionen – das heißt, sie kann einfach erweitert werden. Zusätzliche Rechenkapazitäten und weitere Speicherknoten können so jederzeit hinzugefügt werden. Tests von IBM haben gezeigt, dass man das IBM Cloud Pak for Data System in weniger als vier Stunden zum Laufen bringen kann. So können auch Unternehmen ohne große IT-Ressourcen schnell und einfach anspruchsvolle KI-Kapazitäten aufbauen.

Weitere Informationen

- [1] <https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=8777/ENUSZP18-0296&infotype=AN&subtype=CA>
- [2] <https://www.ibm.com/products/cloud-pak-for-data/system>
- [3] <https://cloud.ibm.com/catalog/services/watson-openscale>

Claus Huempel
claus.huempel@de.ibm.com



Nadine Brehm
nadine.brehm@de.ibm.com



Das Pink Database Paradigma (PinkDB): Die Datenbank durch die rosa Brille

Christian Schwitalla, stellvertretender Leiter der Development Community, hat mit Philipp Salvisberg (Trivadis) über dessen PinkDB-Ansatz gesprochen.

Philipp, datenbankzentrierte Software-Entwicklung gehört zu deinen Schwerpunkten. In deinem Blog [1] beschreibst du PinkDB als „application architecture for database centric applications“. Könntest du die wesentlichen Eigenschaften des PinkDB-Ansatzes beschreiben?

„PinkDB“ steht für „processing in knowing database“. Im Kern geht es darum, die Verarbeitung zu den Daten zu bringen und nicht umgekehrt. Wenn die Daten also in einer relationalen Datenbank gespeichert sind, bringen wir die Verarbeitung im ersten Schritt dorthin. Wir beschreiben mit möglichst wenigen SQL-Befehlen, was zu tun ist, und überlassen das Wie – die Auswahl der geeigneten Algorithmen – der Datenbank. Der Optimizer findet heutzutage in den allermeisten Fällen gute Lösungen.

Falls nicht, können wir ihm immer noch helfen. Der Zugriff auf die Tabellen wird über ein API geschützt. Ein API besteht aus Views und Stored Objects. Dadurch schaffen wir den Spielraum, Datenmodelle und APIs weitgehend unabhängig voneinander zu erweitern. Der Zugriff auf die APIs erfolgt über sogenannte Connect-User, die nur für APIs berechtigt sind und keine eigenen Objekte besitzen.

Wo liegt der Unterschied zu anderen Konzepten, wie zum Beispiel der SmartDB von Bryn Llewellyn?

PinkDB ist nichts Neues und beschreibt, was wir in den 90er Jahren unter „Thick Database“ verstanden haben. SmartDB ist eine extreme Form von PinkDB. Ein SmartDB API besteht aus-

schließlich aus PL/SQL-Packages. Außerdem dürfen Transaction Control Statements (COMMIT, ROLLBACK etc.) nur innerhalb von PL/SQL-Packages ausgeführt werden. Im Gegensatz zu Entity Microservices nutzen SmartDB- und PinkDB-Applikationen die Datenbank als Processing Engine. Dies reduziert Network Roundtrips, den Ressourcenverbrauch auf der Datenbankseite und die Verarbeitungszeiten. Dadurch, dass Views in einem API einer SmartDB-Applikation nicht erlaubt sind, ist das sinnvolle Einsatzgebiet kleiner als bei einer „klassischen“ PinkDB-Applikation. Views sind für mich ein wesentliches Element eines Datenbank-API, um Konsumenten wie APEX die Möglichkeiten von SQL zur Verfügung zu stellen.

Bei Datenbanken denkt man eher an Datenmodellierung. Warum sollte man beim Aufbau einer Datenbank auch die Applikationsarchitektur im Blick behalten?

Die Datenmodellierung ist nach wie vor ein zentraler Aspekt bei Datenbanken. Sich Gedanken über das API einer Datenbank zu machen, hilft bei der Wartung und Weiterentwicklung einer Applikation. Solange sich die bestehenden API-Komponenten nicht anders verhalten, können jederzeit neue Versionen der Datenbank-Komponenten freigegeben werden, ohne dass die entsprechenden Konsumenten (Middle-Tier, GUI) angepasst werden müssen. Das hilft beim Fixen von Bugs, aber auch bei der Einführung von neuer, geänderter Funktionalität mithilfe von versionierten API-Komponenten. In Zeiten immer kürzer werdender Release-Zyklen ist dies ein nicht zu unterschätzender Mehrwert.

Im Rahmen deiner beruflichen Tätigkeit bist du häufig in Kundenprojekten unterwegs. Machen sich die Datenbank-Anwender ausreichend Gedanken über Applikationsarchitektur?

Es kommt darauf an. Auf den Kunden, die Aufgeschlossenheit der Architekten und das Vorhaben. In den letzten Jahren war schon ein Trend weg von der Datenbank festzustellen, was teilweise zu unglaublich ineffizienten Lösungen geführt hat. Datenbanken als dummen Datenspeicher einzusetzen, führt zu mehr Netzwerkverkehr, höherem Ressourcen-Verbrauch in der Datenbank und schlechteren Verarbeitungszeiten. Mein Eindruck ist, dass dies in letzter Zeit mehr und mehr verstanden wird und die Akzeptanz von datenbanknahen Architekturen wieder gestiegen ist.

Bietet die Projektpraxis ausreichend Raum für Gedanken an neue Lösungen, wie zum Beispiel PinkDB? Wie flexibel kann dieser Ansatz ausgelegt werden?

PinkDB ist ein praxisnahes Paradigma. Es besteht aus vier Empfehlungen, die einfach zu befolgen sind. PinkDB erlaubt auch Ausnahmen, solange die Gründe dafür dokumentiert werden. Damit ist ein hoher Grad an Flexibilität gegeben.

Wurde PinkDB bereits in Kundenprojekten umgesetzt? Welche Erfahrungen hast du dabei gemacht?

Ja. ThinkDB oder FatDB gibt es schon lange. Wir haben es in vielen Projekten erfolgreich eingesetzt. Als problematisch habe ich in diesen Projekten hauptsächlich den Einsatz von komplexen oder geschachtelten Instead-of-Triggern in Erinnerung. Vor allem dann, wenn viele Datensätze verarbeitet wurden. Heute achte ich beim Design auf die Notwendigkeit von zusätzlichen, Bulk-fähigen APIs.

Im Gegensatz zu früheren Trends kann man zunehmend Bestrebungen beobachten, datenintensive Anwendungen wieder nah an den Daten zu platzieren. Unterstützt der PinkDB-Ansatz diesen Trend?

Auf jeden Fall. Verarbeitungen nahe an der Datenbank auszuführen, ist ein wesentliches Ziel von PinkDB. Auch wenn in Diskussionen immer wieder das View API gestresst wird, darf man nicht vergessen, dass PinkDB explizit auch Stored Objects, das heißt Packages, Object Types, Functions und Procedures, als Teil des API unterstützt. Damit lassen sich beispielsweise Batch-Verarbeitungen in der Datenbank optimal umsetzen.

Denkst du, dass der PinkDB-Ansatz weiterentwickelt werden kann? Gibt es weitere Ansätze, die du interessant findest?

Ja, ich denke die PinkDB-Empfehlungen lassen sich konkretisieren. Beispielsweise wie genau APIs versioniert werden können, welche Vor- und Nachteile der jeweilige Ansatz hat und welche Rolle EBR dabei spielen kann. Interessant finde ich die aktuellen Entwicklungen die MLE betreffend. Nicht nur wegen der zusätzlich zur Verfügung stehenden Sprachen in der Datenbank und deren effiziente Interoperabilität, sondern auch bezüglich der Möglichkeit, Verarbeitungen zur Laufzeit transparent vom Middle-Tier zur Datenbank zu verschieben.

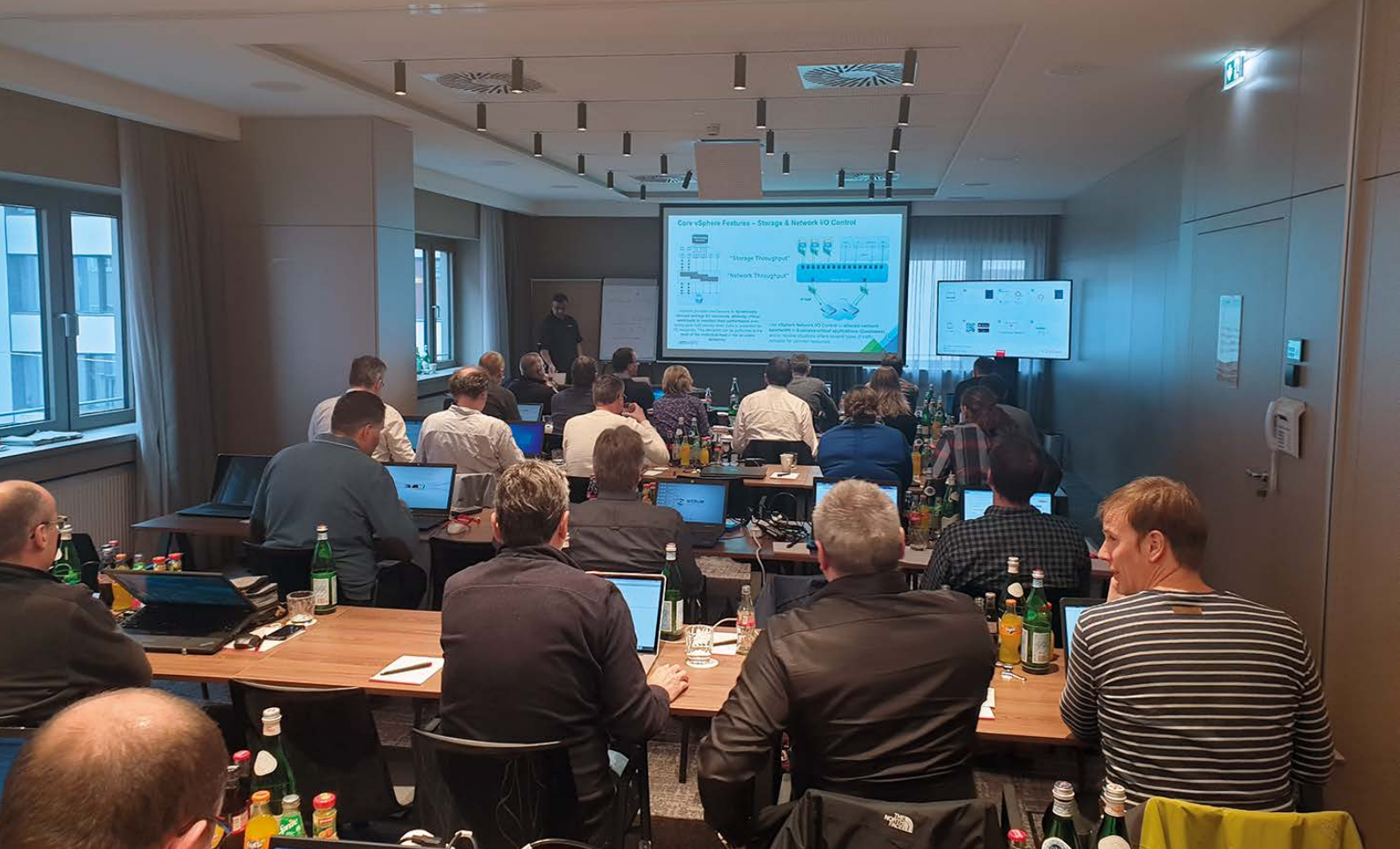
Quellen

- [1] <https://www.salvis.com/blog/2018/07/18/the-pink-database-paradigm-pinkdb/>



Zur Person: Philipp Salvisberg

Philipp Salvisberg ist ein Senior Principal Consultant und Partner der Trivadis-Gruppe sowie ein Oracle ACE. Seit 1988 fokussiert er sich auf datenbankbasierte Lösungen. Seither unterstützt er Kunden beim Design, Aufbau und der Optimierung ihrer datenbankzentrischen Lösungen, unterrichtet Themen der Anwendungsentwicklung oder hält Vorträge auf Konferenzen. Philipp hat ein Faible dafür, möglichst viel in einer einzigen SQL-Anweisung auszuführen und ist an so ziemlich allem interessiert, um die Datenbank so effizient wie möglich zu nutzen. Er entwirft gern modellgetriebene Softwarelösungen, auch unter Verwendung eigener domänenspezifischer Sprachen, wobei er immer bestrebt ist, das Beste aus den zugrunde liegenden Technologien herauszuholen.



Das war die DOAG 2020 Noon2Noon

Martin Klier, DOAG Database Community

Am 23. und 24. Januar 2020 fand die sechste DOAG Noon2Noon statt. Diesmal in München, ganz zentral in einem Hotel am Hauptbahnhof.

Was ist eigentlich ein DOAG Noon2Noon? Ein Noon2Noon steht für 24 Stunden Technik, Technologie und Interaktion. Von 12 Uhr mittags bis 22 Uhr am Abend, und von 9 Uhr morgens bis 16 Uhr am zweiten Tag gab es ein sehr intensives Programm aus Vorträgen, Labs und Hilfe zur Selbsthilfe.

Alles drehte sich um Interna, wie VMware mit realen Ressourcen umgeht, wie Oracle diese virtualisiert nutzt und wie der DBA und der VMware-Administrator die Funktionen und die Features beider Technologien zusammen einsetzen können. Immer mit dem Ziel, die optimale Leistung bei möglichst kleinen Kosten zu erhalten.

Am Donnerstag führte Valentin Bondzio (VMware Cork) in VMware Core-Funktionen ein: Das CPU Accounting, die IO-

Technologien, vMotion, Failsafe und viele mehr. Anschließend leitete Sudhir Balasubramanian (VMware Palo Alto) durch zwei große Laborübungen, die die Teilnehmer teils bis nach 21 Uhr zu intensiver Arbeit und geschäftiger Gruppenarbeit anhielten. Aber natürlich kamen besonders zu späterer Stunde der lockere Austausch, zwangloses Beisammensein und genüssliches Fachsimpeln nicht zu kurz.

Der Freitag begann mit einem tollen Überblick über die Praxis der Lizenzierung von Oracle-Datenbanken unter VMware. Arno Mastik (License Consulting) war eigens aus den Niederlanden angereist, um beim DOAG Noon2Noon von der richtigen Vorbereitung auf mögliche Lizenzaudits („Beat Oracle Before Oracle Beats You“) und über die erfolgversprechendsten Strategien im Um-

gang mit Oracle LMS und Oracle Legal zu sprechen.

Danach gab es wieder technische Hintergrundinformationen, diesmal zu Snapshots, Host Affinity Rules und Performance Troubleshooting für VMware-Umgebungen. Den Ausklang bildeten am frühen Nachmittag dann technische Diskussionen zu den Laborübungen, aus der Praxis und die fundierte Behandlung „unerklärlicher“ Beobachtungen.

Die 38 Teilnehmer, die aus ganz Deutschland und der Schweiz angereist waren, bewiesen großes Engagement, sorgten für gute Stimmung und lieferten hervorragendes Feedback. Die fachliche Tiefe und Relevanz für die Arbeit im Spannungsfeld zwischen Datenbank und virtueller Systemtechnik wurden von vielen ausdrücklich gelobt.



„Die Cloud an sich ist keine Lösung, sondern eine Technologie“

Martin Meyer, Redaktionsleiter des Red Stack Magazin, sprach mit Frits Hoogland (Accenture Enkitech Group) in einem Kurz-Interview über das Thema seiner Keynote „The importance of failing“.

Wie kommen Sie zu Ihrem Keynote-Thema „The Importance of Failing“?

Der Titel soll zum Nachdenken anregen. Die Inspiration kommt von der Tatsache, dass, wenn man sich Youtube-Videos, Präsentationen, Demos und Whitepapers ansieht, "jeder" in der Lage zu sein scheint, Dinge aufzuschreiben und ohne langes Nachdenken eine neue Technologie in wenigen Minuten zum Laufen zu bringen. Im wirklichen Leben funktioniert es aber nicht so einfach. Wenn ich das tue, stoße ich sogar sofort auf Probleme wie Betriebssystem- und Paketkompatibilität. Der erste Versuch scheitert mehrfach und ich brauche einen halben Tag, um etwas Anständiges aus meiner Idee zu machen. Ich bin überzeugt, dass ich nicht der Einzige mit diesen Erfahrungen bin, sondern dass dies universell ist. Ich möchte diese Idee den Zuhörern nahebringen und versuche, einige weitere Überlegungen darüber anzustellen, wie man sie nutzen kann und wie man sie nicht nutzen sollte.

Wie beurteilen Sie Europa hinsichtlich des Fortschritts und der Einstellung beim Thema Cloud?

Wir sollten uns wirklich bewusst sein, dass der digitale Prozess nicht immer unbedingt Gutes mit sich bringt. Letztlich sind es die von Menschen getroffenen Entscheidungen, die zählen. Die Cloud ist gekommen und wird auch bleiben, das ist klar. Aber die Cloud an sich ist keine Lösung, sondern eine Technologie. Und wie jede Technologie kann sie sich, wenn man sie falsch einsetzt, als eine schlechte Lösung erweisen. Ich bin auch davon überzeugt, dass es für die On-Premises-Technologie ebenso viel Platz gibt wie für die Cloud. Eigentlich ändern sich Projektmanagement und Projekte im Allgemeinen nicht: Wenn Sie die Situation nicht sorgfältig genug einschätzen und eine Alles-in-die-Cloud-Strategie verfolgen, wird diese einfach scheitern.

Wo sehen Sie Herausforderungen in Gegenwart und Zukunft?

Die Dinge werden sehr unterschiedlich und gleichzeitig ziemlich gleich sein. Technologische Fortschritte werden wahrscheinlich

einige aktuelle technologische Probleme lösen, so dass wir uns auf die Lösung anderer Probleme konzentrieren können. Ich stelle fest, dass weniger Zeit für den sorgfältigen Aufbau von Architektur- und IT-Projekten aufgewendet wird und dass weniger spezifisches Wissen eingesetzt wird, was die Erfolgsaussichten von IT-Projekten verringert.



Zur Person: Frits Hoogland

Frits Hoogland ist ein IT-Experte, der sich auf die Leistung und die Interna von Oracle-Datenbanken spezialisiert hat. Frits präsentiert häufig auf Konferenzen auf der ganzen Welt technische Themen zu Oracle. Im Jahr 2009 erhielt er eine Oracle ACE-Auszeichnung vom Oracle Technology Network und wurde ein Jahr später Oracle ACE-Direktor. Im Jahr 2010 trat er dem OakTable-Netzwerk bei. Neben der Entwicklung seiner Oracle-Kompetenz arbeitet Frits mit modernen Betriebssystemen. Frits arbeitet derzeit in der Gruppe Accenture Enkitech.

BUSINESS — NEWS

NEWS



Künstliche Intelligenz, Machine Learning und Deep Learning: Was sind die Unterschiede und wozu kann was verwendet werden?

Björn Bröhl, Trivadis AG

Einer der Top-IT-Trends ist Hyperautomatisierung. Er ist die Fortsetzung des Trends zur Automatisierung und beinhaltet zum Beispiel eine Form von Robotic Process Automation (RPA), ergänzt um Technologien wie künstliche Intelligenz und maschinelles Lernen. Künstliche Intelligenz und maschinelles Lernen bilden die Basis für fast alle Top-IT-Trends. Erst durch diese wird die Umsetzung ermöglicht oder massiv beschleunigt. Die Umsetzung neuer Geschäftsfelder im Rahmen der digitalen Transformation ist in vielen Fällen erst möglich durch die Analyse großer Datenmengen, die durch menschliche Auswertungen nicht möglich ist. Der Artikel gibt eine Erklärung und Einordnung der Themen künstliche Intelligenz, maschinelles Lernen und Deep Learning.

Vor gar nicht allzu langer Zeit war künstliche Intelligenz etwas für Science-Fiction-Filme oder Serien. Bereits 1982 gab es mit K.I.T.T. in der Serie „Knight Rider“ ein Auto, das sprechen, sehen und selbstständig handeln konnte. Zuvor, 1968, hatte in Stanley Kubricks Weltraumklassiker „2001: Odyssee im Weltraum“ der auf künstlicher Intelligenz basierende Bordcomputer HAL 9000 die Führung übernommen, um seiner bevorstehenden Abschaltung zu entgehen. Und Steven Spielberg widmete sich im Jahre 2001 einem ehemals von Kubrick angegangenen Science-Fiction-Film-Projekt namens „A.I. – Künstliche Intelligenz“. Hier haben durch Umweltkatastrophen und Geburtenkontrolle sogenannte „Mechas“ Einzug gehalten, die nicht mehr von echten Menschen zu unterscheiden sind und sogar emotionale Bindungen zu ihren Besitzern aufbauen können. Im heutigen Alltag erleben wir an vielen Stellen künstliche Intelligenz. Sei es, dass der Media-Player im Auto genau die Musik vorschlägt, die einem gefällt, oder dass das Auto den Unterschied zwischen einem Tier, einem Menschen oder einem anderen Fahrzeug erkennt und entsprechend handelt.

Rund um das Thema künstliche Intelligenz schwirren viele Begriffe umher. Nachfolgend möchte ich etwas Licht in das

Dunkel bringen und mit Beispielen die Unterschiede zwischen künstlicher Intelligenz, Machine Learning und Deep Learning erklären. Von diesen drei Begriffen ist künstliche Intelligenz der gängige Oberbegriff für diverse Techniken und Technologien, zu dem auch Machine Learning und Deep Learning gehören. Diese Technologien eint die Vision, dass eine maschinelle, künstliche Form der Intelligenz der menschlichen Intelligenz überlegen ist.

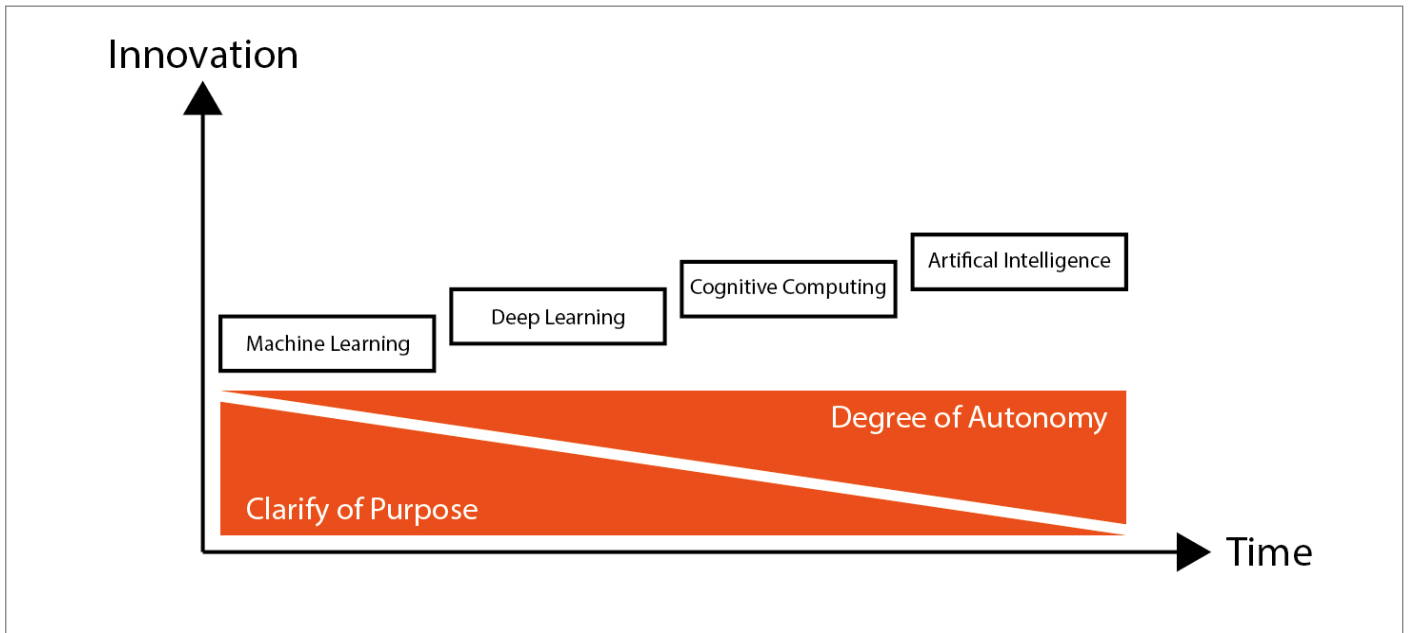
Künstliche Intelligenz

Künstliche Intelligenz bezeichnet Maschinen, die menschliche kognitive Fertigkeiten wie das Lösen von Problemen oder andere Fähigkeiten nachahmen, die Sprache, Sprechen, Sehen und strategisches Denken voraussetzen. KI-Anwendungen versetzen Maschinen in die Lage, bestimmte menschliche Aufgaben genauso gut oder sogar besser auszuführen. In der heutigen Zeit, wo immense Datenmengen in immer schnellerer Zeit generiert werden, wird KI immer bedeutender. Es ist für den Menschen praktisch nicht mehr möglich, die zur Verfügung stehende Menge an Daten zu sichten, zu ordnen, zu analysieren, zu bewerten und logische Schlüsse daraus zu ziehen.

Künstliche Intelligenz finden wir heute in ganz unterschiedlicher Komplexität in vielen alltäglichen Dingen. Ein Beispiel sind Sprachassistenten, die in jedem Smartphone, als intelligente Lautsprecher in vielen Haushalten oder in Autos eingesetzt werden. Sie basieren auf einer Mustererkennung von Sprache und interpretieren daraus auszuführende Aufgaben, zum Beispiel „Schalte die Sitzheizung aus!“. Die KI im Auto erkennt dabei die Schlüsselwörter „schalte“, „Sitzheizung“ und „aus“. In Kombination mit den Informationen aus dem Sensor für die Sitzbelegung auf der Fahrerseite wird dann die passende Aktion ausgeführt.

Machine Learning

Machine Learning ist ein Teilgebiet der künstlichen Intelligenz. IT-Systeme sind in der Lage, durch das Erkennen von Mustern und Gesetzmäßigkeiten in Datenbeständen eigenständig Lösungen für Probleme zu finden. Es wird quasi künstliches Wissen aus Erfahrungen generiert. Die gewonnenen Erkenntnisse aus den Daten lassen sich verallgemeinern und für neue Problemlösungen oder für die Analyse von bisher unbekanntem Daten verwenden. Damit eine Maschine selbstständig lernen und Lösungen finden kann, ist



Einordnung von künstlicher Intelligenz und ihren Techniken Machine Learning und Deep Learning. Quelle: Crisp Research

vorher die Unterstützung von Menschen notwendig. Dafür müssen zum Beispiel die Systeme mit den für das Lernen relevanten Daten und Algorithmen versorgt werden. Zudem sind Regeln für die Analyse des Datenbestands und das Erkennen der Muster aufzustellen. Diese Regeln basieren unter anderem auf bestimmten Schlüsselwörtern, Text- oder Bildmustern. Ein Beispiel findet sich im Bereich der SPAM-Erkennung. Hier hilft maschinelles Lernen indem aus als SPAM markierten E-Mails ein Muster abgeleitet wird, das bei weiteren E-Mails diese immer besser bewerten kann.

Wenn passende Daten vorhanden und Regeln definiert sind, können Systeme mit maschinellem Lernen unter anderem Folgendes:

- Relevante Daten finden, extrahieren und zusammenfassen,
- Vorhersagen auf Basis der analysierten Daten treffen,
- Wahrscheinlichkeiten für bestimmte Ereignisse berechnen,
- sich an Entwicklungen eigenständig anpassen und
- Prozesse auf Basis erkannter Muster optimieren.

Es gibt ein sehr großes Spektrum von Anwendungsmöglichkeiten für Machine Learning. Bleiben wir bei den Beispielen aus dem Automobilbereich. Maschinelles Lernen ermöglicht hier:

- selbstständiges Erkennen von Fehlern in Produktionsabläufen (zum Beispiel Toleranzen bei der Blechverarbeitung)
- Sprach- und Texterkennung für digitale Assistenten im Fahrzeug
- Erkennung von Absatzpotenzialen bestimmter Fahrzeugtypen und Ausstattungen, indem aus den Statistiken des Absatzes Muster abgeleitet werden.

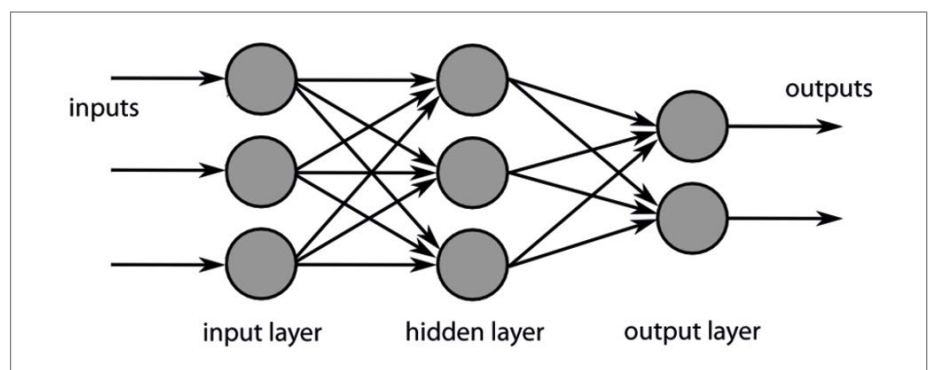
Weitere Anwendungsbereiche des maschinellen Lernens in der Zukunft liegen in der Erkennung von Fahrverhalten oder, über die Gesichtserkennung, des Zustands der Fahrtüchtigkeit.

Deep Learning

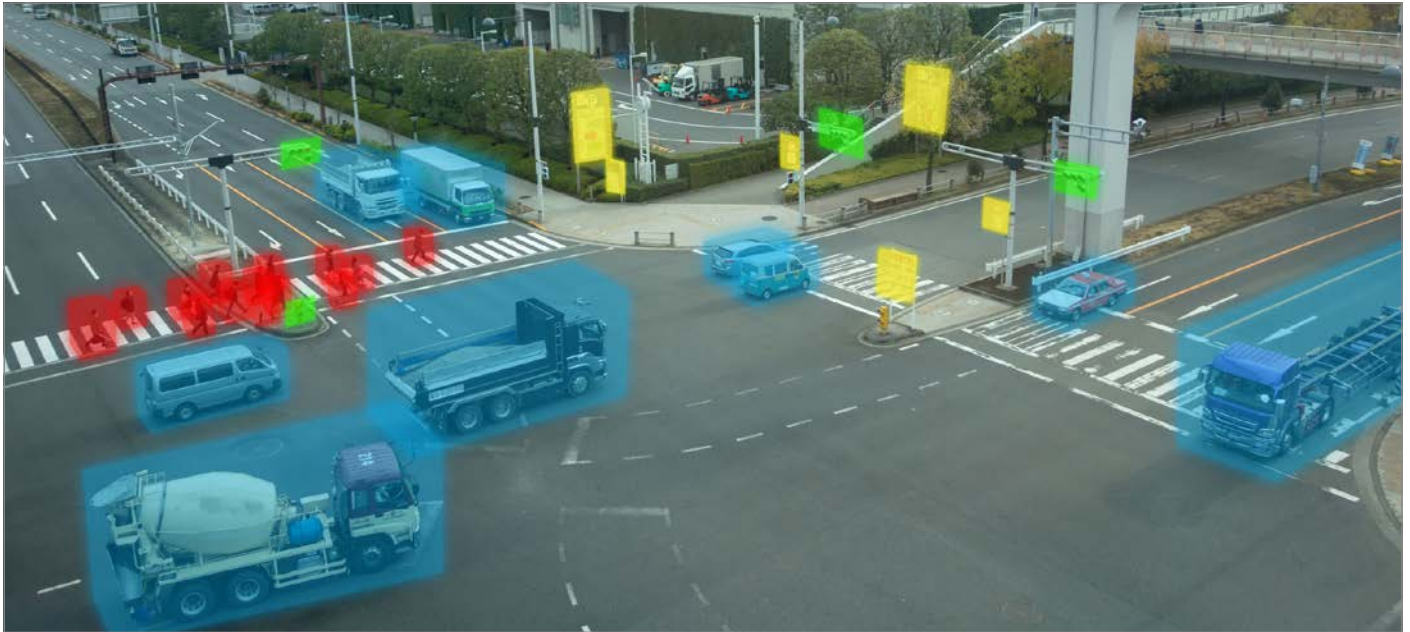
Ein Teilbereich des Machine Learning ist Deep Learning. Dieses basiert auf neuronalen Netzwerken sowie großen Datenmengen. Diese Lernmethode ist der Funktionsweise des menschlichen Gehirns nach-

empfunden. Daraus resultiert die Fähigkeit, eigene Prognosen oder Entscheidungen zu treffen. Die Basis der Methode ist der Aufbau eines neuronalen Netzwerks, das eine Art künstliches Abstraktionsmodell des menschlichen Gehirns darstellt. Es besteht aus künstlichen Neuronen. Im Aufbau besteht es aus sogenannten Eingangs- und Ausgangsneuronen. Dazwischen befinden sich mehrere Schichten an Zwischenneuronen. Die Eingangsneuronen lassen sich durch Lernen auf unterschiedlichen Wegen über die Zwischenneuronen mit den Ausgangsneuronen verknüpfen. Je mehr Neuronen und Schichten existieren, desto komplexere Sachverhalte lassen sich abbilden.

Durch das Verknüpfen wird die Maschine in die Lage versetzt, selbstständig und ohne menschliches Zutun ihre Fähigkeiten zu verbessern. Dies funktioniert, indem aus vorhandenen Daten und Informationen Muster extrahiert und klassifiziert werden.



Aufbau von künstlichen neuronalen Netzwerken. Quelle: Wikipedia



Objekterkennung mittels Deep Learning, der Unterschied zwischen Schildern, Fahrzeugen und Lebewesen.

Die gewonnenen Erkenntnisse lassen sich wiederum mit Daten korrelieren und in einem weiteren Kontext verknüpfen. Schließlich ist die Maschine fähig, Entscheidungen auf Basis der Verknüpfungen zu treffen. Durch das kontinuierliche Hinterfragen der Entscheidungen erhalten die Informationsverknüpfungen bestimmte Gewichtungen. Bestätigen sich Entscheidungen, erhöht sich deren Gewichtung, werden sie revidiert, verringert sich die Gewichtung. Zwischen der Eingabeschicht und der Ausgabeschicht entstehen immer mehr Stufen an Zwischenschichten und Verknüpfungen. Über den eigentlichen Output entscheidet die Anzahl der Zwischenschichten und deren Verknüpfung.

Die Unterschiede

Der Unterschied zwischen Deep Learning und Machine Learning besteht darin, dass beim maschinellen Lernen der Mensch in die Analyse der Daten und den eigentlichen Entscheidungsprozess eingreift. Beim Deep Learning sorgt der Mensch lediglich dafür, dass die Informationen für das Lernen bereitstehen und die Prozesse dokumentiert sind. Die eigentliche Analyse und das Ableiten von Prognosen oder Entscheidungen überlässt der Mensch der Maschine selbst. Der Mensch hat keinen Einfluss auf die Ergebnisse des Lernprozesses. Es lässt sich im Nachhinein nicht mehr vollständig zurückverfolgen, auf Basis welcher genauen Muster eine Maschine eine bestimmte Entscheidung getroffen hat. Zudem werden die

Entscheidungen von den Algorithmen ständig hinterfragt und die Entscheidungsregeln automatisch selbstständig optimiert.

Autonomes Fahren ist ohne den Einsatz von Deep Learning nicht möglich. Die Objekterkennung muss durch eine unendlich große Anzahl von Bildern und Videos trainiert werden, um sicher zwischen anderen Fahrzeugen, Lebewesen oder sonstigen Hindernissen unterscheiden zu können. Ohne den Einsatz dieser Technologie können zum Beispiel in hochkomplexen Verkehrssituationen wie Baustellen oder Hindernisse keine Entscheidungen getroffen werden.

Künstliche Intelligenz in der Zukunft?

Zugegeben, in den meisten verfügbaren Anwendungsfällen erlebt der Nutzer heute noch sehr deutlich, dass es sich bei auf künstlicher Intelligenz basierenden Systemen noch um genau diese handelt. Durch die immer größere Menge an verfügbaren Daten, durch die KI-Systeme lernen können, und die steigende Geschwindigkeit der Verarbeitung dieser Daten werden die Systeme immer flüssiger in der Interaktion. So wird es zukünftig schwierig, unterscheiden zu können, ob ein Mensch ein Auto fährt oder ein System diese Aufgabe autonom übernommen hat. Die Möglichkeit, diese Systeme als nicht-menschliche zu erkennen, schwindet. Die Kombination von künstlicher Intelligenz mit animierten Personen erschafft täuschend echte „künstliche Menschen“ die, wie im Januar 2020 auf der CES in Las Vegas vorgestellt, mit uns interagie-

ren und durch ihre Antworten, aber auch durch ihre Gestik und Mimik, völlig natürlich wirken. Bleibt am Ende nur die Frage, wie schnell der Mensch lernen kann, sich daran zu gewöhnen.



Björn Bröhl

bjoern.broehl@doag.org

Björn Bröhl ist Head of Marketing & Communications Trivadis und verantwortet diesen Bereich für die gesamte Trivadis-Gruppe (Schweiz, Deutschland, Österreich und Rumänien). Zudem beschäftigt er sich mit Themen zu Oracle Infrastrukturen, Data Analytics sowie mit Technologien wie Big Data, IoT und Digital Business Transformation. Er entwirft bis heute die Architektur für kritische Systeme/IT-Umgebungen und berät Kunden hinsichtlich der optimalen Gestaltung des IT-Betriebs und dessen Strategie. Er ist aktuell Vorstand und Leiter der DOAG Data Analytics Community.



Jäger des verborgenen Schatzes: Mithilfe von KI schürft aiconix Daten-Gold aus Archiven

Elvira Steppacher

Hochwertiger Content gleicht einer Goldmine. Nur, wie den Schatz heben? Das Hamburger Start-up aiconix verspricht, das Schürfen zu vereinfachen und dabei das Material zu veredeln. Die KI-Plattform aingine soll nicht nur Verlagen und Medienhäusern helfen, sondern allen Unternehmen, die Content-Berge anhäufen. Das Ziel: (Bewegt-)Bilder zielgruppengenau zu veredeln sowie Autoren und Filmemacher dabei zu unterstützen, relevanten Content zu produzieren.

Geärgert habe er sich. Kameramann Eugen Gross wusste, dass es den Shot gab. Er hatte ihn selbst gedreht, besaß alle Rechte, hatte das Material sogar eigenhändig archiviert – nur finden konnte er den Take auf seinem Server nicht. „Wahnsinn, dass ich mehr Zeit mit Suchen von Videos zubrachte als mit dem Dreh oder dem Schnitt“, erinnert sich Gross. Wenn ihm, einem Freelancer, das Problem teure Zeit stahl, um wie viel mehr mussten Verlage, Medienhäuser, Rundfunkanstalten, Konzerne damit kämpfen? Content wächst unaufhaltsam, jede Minute werden allein auf YouTube 500 Stunden Videos hochgeladen, kaum eine Branche ohne Bewegtbild. Unsere Welt, ein Video – und bald ein Pixellabyrinth. Der Filmemacher erkannte das Potenzial. Diesen ganzen Wust müsste man sichten, kategorisieren, prozessfähiger machen. Per Click verfügbar. Aber nicht von den Künstlern, sondern von künstlicher Intelligenz (KI). Die Kreativen hätten dann endlich wieder Zeit für Kreation, das, was immer zu kurz kommt, wenn man über kein Budget verfügt, das nur aufwendige Werbe- oder Spielfilme erhalten. Er schreibt eine Masterarbeit darüber, wie KI Kreative unterstützen könnte, stellt einen Businessplan auf und erhält gemeinsam mit seinem damaligen Team das EXIST-Gründerstipendium am StartUp Dock der Technischen Universität Hamburg.

Bei dem Hamburger Next Media Accelerator formierte sich wenige Wochen später das Team neu. Neben CEO Gross gab nun der Informatiker Aleksander Koleski als CTO dem Projekt Gestalt. Im April 2018 galt es, das wachsende Entwicklerteam und die Kostenstruktur zu professionalisieren, seither ist CFO Mariel Prange an Bord. 2019 wird das Start-up Mitglied im Oracle Ecosystem. Der Name aiconix ist Programm: Die Silbe Icon (engl. für Bild / Symbol) wird von den Buchstaben A plus I (engl. für Artificial Intelligence) eingerahmt und von einem X (für Experience und Analytics) beendet. Der Anspruch war umrissen: „Wir bauen das Supertool. Unsere KI durchforstet die Content-Stollen, findet, sortiert und wäscht die Nuggets, obendrein sagt sie voraus, wie das Publikum auf sie reagiert.“

Die Plattform *aingine* ist ein One-Stop-Shop für die automatisierte Verarbeitung, Auswertung und Veredelung von Medieninhalten. Ihre Features sind unter anderem Gesichtserkennung (Face recognition), Live-Video-Untertitelung (Speech-to-text), automatisierte Beschriftung von Bildinhalten

und Video-/Audio-SEO. Hierbei aggregiert aiconix verschiedene Best-in-Class-Anbieter, das heißt, alle sind auch ohne aiconix verfügbar, doch ergänzt um proprietäre Technologien erzeugt die aingine drei besondere Benefits. Erstens, die Plattform vereinfacht das Schnittstellenmanagement, weil der Kunde ein, statt zahlreiche APIs verwaltet, eine, statt viele Rechnungen erhält. Zweitens, sie reduziert die Workflow-Kosten durch ein einfach zu implementierendes Frontend. Drittens, aingine hostet sämtliche von ihr genutzten Daten auf Servern des Frankfurter Rechenzentrums von Oracle und verarbeitet die Daten konform zu den europäischen Datenschutzbestimmungen. Hinzu kommen technologische Features, die über die aggregierten Anbieter hinausgehen, wie zum Beispiel die Erkennung der Einstellungsgröße (shot Detection), eine eigene Personenerkennung, bei der die Daten im Besitz des Kunden bleiben, oder den Best-Results-Algorithmus, der den Provider auswählt, der am besten zu dem Use Case passt.

Die ursprüngliche Idee der aingine basiert auf Assistenzfunktionen, die in den Bereich Machine Learning gehören: Wie lassen sich Producer und Kreative von zeitaufwendigen Aufgaben entlasten? Das führte zum nächsten Ziel, der Performance-Vorhersage. Wie oft wird dieses Video voraussichtlich geklickt werden? Warum springen User bei bestimmten Videos ab? Medienproduzent Gross: „Videos präsentieren Inhalte mit bestimmten Mitteln, eben Bild, Ton, Text, Musik, Einstellungsgröße, Schnitt, Licht usw. Erst das Arrangement der Mittel macht aus den Zutaten ein verständliches, fesselndes, unterhaltendes Video. Hier gibt es zielgruppenspezifische Vorlieben. Wenn es uns gelingt, die Parameter zu destillieren, haben wir ein hochwirkungsvolles Instrument in der Hand.“

Dank der Unterstützung der Hamburger Förderbank IFB (Hamburgische Investitions- und Förderbank; Programm InnoRampUp) konnte aiconix sein Neuro-Lab eröffnen. „Wir sind froh, dass wir unsere Idee prototypisieren konnten“, meint Mariel Prange, „jetzt geht es darum, weitere Investoren zu gewinnen. Damit der Prototyp in Serie gehen kann, muss er unaufwendiger und billiger sein. Uns schwebt eine Art App vor, die die Aufmerksamkeit misst.“ Die Meta- und Analysedaten der Videos sollen um aufmerksamkeitspsychologische und



Der grüne Faden für Ihre Digitale Evolution

Wir bei PROMATIS folgen einem selbst entwickelten grünen Faden:

Mit professioneller Beratung und innovativen Digitalisierungslösungen schaffen wir exzellente Geschäftsprozesse: agil, bedarfsgerecht, intelligent und zukunftssicher. Nachhaltige Qualität und Wirtschaftlichkeit sichern wir durch kontinuierliche Verbesserung der eingesetzten Verfahren, Produkte und Services.

Mit unserer Digitalisierungskompetenz und unseren Best Practice-Lösungen begleiten wir Sie auf Ihrer Reise in die Oracle Cloud.

PROMATIS Gruppe
Pforzheimer Str. 160
76275 Ettlingen
+49 7243 2179-0
www.promatis.de

Ettlingen | Hamburg | Berlin | Münster
Wien | Zürich | Denver



Im NeuroLab sollen Gehirnströme von Probanden mit Sensoren ausgewertet werden. Wieviel Aufmerksamkeit verbindet der Zuschauer mit dem Gesehenen?

biosensorische Aspekte erweitert werden. Der Weg zur Vorhersage des Nutzerverhaltens war eingeschlagen, von dort zur automatisch generierten Video-Produktion ist es nurmehr ein kleiner Schritt. Schon heute selektieren manche KIs selbstständig geeignete Sequenzen, aus denen die Trailer von Blockbustern entstehen [1].

Koleski, der Informatik mit den Nebenfächern Psychologie und Medizin studierte, weiß, dass aiconix sich viel vorgenommen hat. Die Beschreibung der sogenannten Basisemotionen Freude, Trauer, Wut, Verachtung, Überraschung, Angst, Ekel gelingt vergleichsweise einfach. Mimetisch lassen sie sich durch bestimmte Patterns von Augen, Brauen, Nase und Mund im Verhältnis zueinander darstellen – siehe Emoticons. Da es sich um Ausdrucksweisen der menschlichen Psyche handelt, die in allen Kulturen vorhanden sind, lassen sie sich interkulturell verstehen und auch maschinell auslesen. Technologisch gesprochen geht es um Klassifizierung. Schaut Angela Merkel überrascht in dem Video, sind die Zoobesucher traurig oder wütend? Dies ist längst Standard.

Doch um Emotionen vorhersagen zu können, bedarf es mehr. Eine ganze Industrie arbeitet bereits daran, schließlich sind zahlreiche Use Cases in fast allen Feldern des Lebens denkbar: Arbeitssicherheit, Telemedizin, Straßenverkehr, Personenkont-

rollen, Strafvollzug. Vermessen wird alles, was geht: Pupillengröße, Augenbewegung, Mikropulsationen, Farbmuster, biochemische Reaktionen und anderes. Big Data verbessert die Prognosefähigkeit erheblich. Allein durch Aggregationsmöglichkeiten von Daten, die es vorher so nicht gab oder die nicht verfügbar waren, etwa durch Realtime-Körper- und -Gesundheitsdaten, Social-Media-Verläufe, Geotagging etc.

Für aiconix sind in dem Riesengebiet derzeit Kurzfilme relevant, also Videos von einer bis zehn Minuten Länge. Sie entsprechen dem nachrichtlichen Kurzbeitrag, lassen sich aber auch auf Tutorials wie Koch-, Schmink-, Krafttraining-, Heimwerker-, Tierpflege- und anderen How-to-Videos anwenden. Im NeuroLab sollen Gehirnströme von Probanden mit Sensoren ausgewertet werden. Wieviel Aufmerksamkeit verbindet der Zuschauer mit dem Gesehenen? Die Neurowissenschaft arbeitet daran seit Jahrzehnten, auch die Werbewirkungsforschung zeigt hier seit jeher Interesse. Aleksander Koleski: „Bei der Betrachtung von Videos lassen Zuschauer jedoch zu 90 Prozent keine Regungen erkennen. Jedenfalls keine, die mit bloßem Auge verwertbar wären. Es gibt jedoch neue Forschungsansätze, die unter anderem mit hochempfindlichen Lichtmesstechniken arbeiten. Einstweilen konzentrieren wir uns auf das Thema Aufmerksamkeit.“

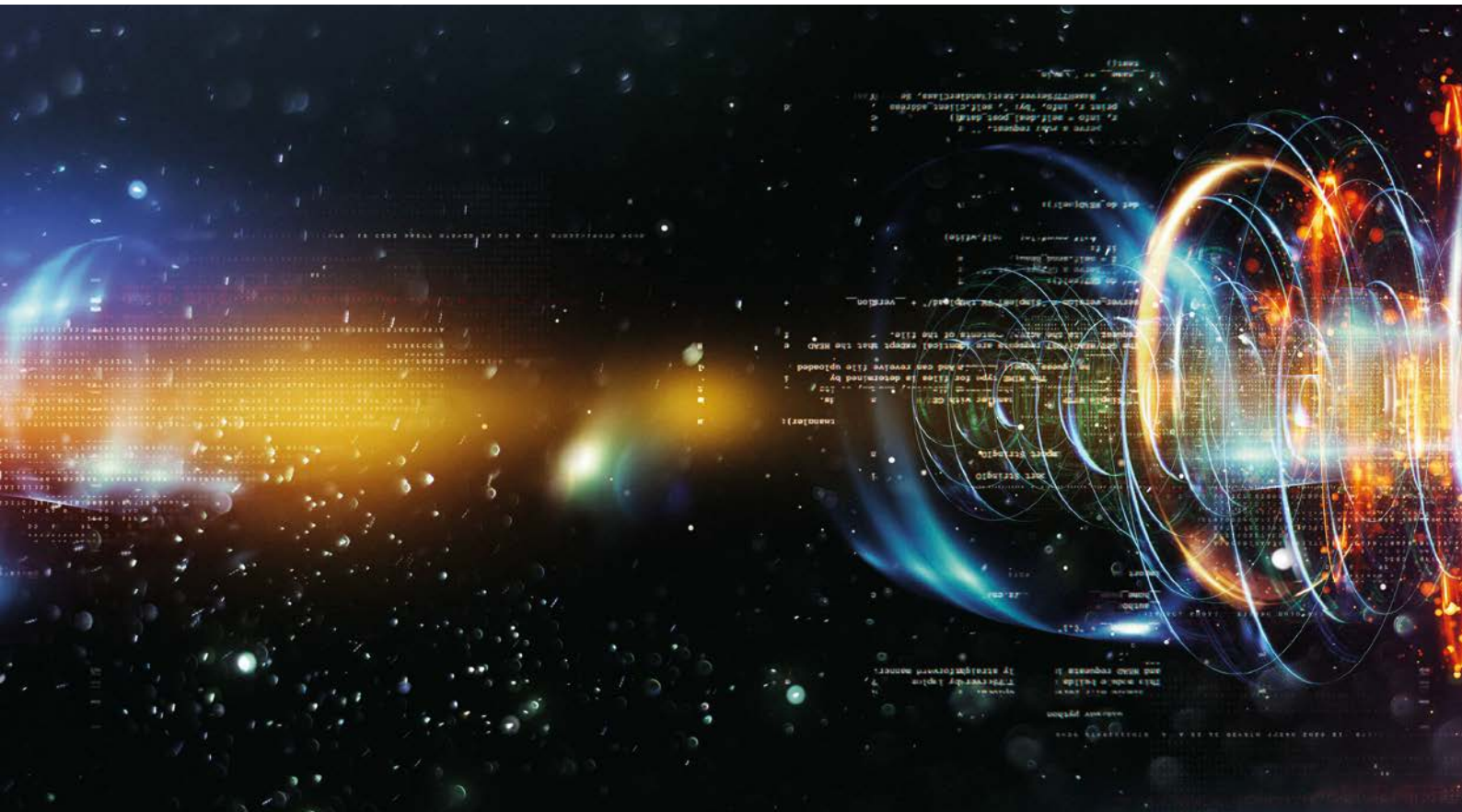
Wenn die Aufmerksamkeit schwindet, so die These, kann das ein Signal für nachlassendes Interesse sein, der Zuschauer springt womöglich ab. Informationen aus drei Ebenen werden daher bei aiconix zusammengeführt: die Analyse des Videos (Inhalt und Komposition), sämtliche aus Analytics verfügbare Daten und als letzte, neueste Erweiterung das neurosensorisch-psychologische Wissen. Je nach Kontext und Zielgruppentypologie kann ein Video die Erwartungshaltung des Zuschauers erfüllen. Das Gesehene stimuliert positiv, die Wahrscheinlichkeit, dass er dieses und vergleichbare Videos bis zum Schluss anschaut, erhöht sich. Dabei macht es einen Unterschied, ob es sich um User der YouTube-Generation, also sogenannte Digital Natives handelt oder um Zuschauer, die mit dem linearen Fernsehen groß geworden sind. Schnittfolgen, Tempo, Kamerabewegung, dies und mehr ändern sich. Tendenziell geht der Trend zum schnellen, subjektiven, live-orientierten visuellen Erzählen. Auch die Wahl der Protagonisten hat mit Präferenzen, Konsum- und Sehgewohnheiten zu tun. „Die Faktoren“, so Gross, „sind multifaktoriell. Wir suchen nach den wenigen relevanten.“

[1] Siehe dazu: <https://katzlberger.ai/2019/09/26/netflix-nutzt-ki-um-personalisierte-trailer-fuer-filme-herzustellen/> und <https://www.ibm.com/blogs/cloud-archive/2016/08/ibm-watson-creates-first-movie-trailer/>



Elvira Steppacher
est@elvira-steppacher.de

Elvira Steppacher, Dr. phil., ist Bloggerin und Autorin mit langjähriger Erfahrung in der journalistischen Aus- und Weiterbildung. Sie schreibt über Fragen der Vermittlung von Digitalisierung, künstlicher Intelligenz (KI), Ethik und berät zu strategischer Kommunikation. Sie unterstützt die Initiative Zukunftsnarrative.com, die den Diskurs über KI in der Gesellschaft fördern will. Überzeugt davon, dass Literatur technologische Zeitenwenden mitgestaltet, schreibt sie Kurzgeschichten und Romane zu den Veränderungen durch Digitalisierung. #gerneMensch lautet ihr Hashtag.



Interview mit den aiconix-Gründern Eugen L. Gross und Aleksander Koleski sowie Mariel Prange, CFO vom Hamburger Start-up

Warum ist aiconix für Medienunternehmen interessant?

Gross: Medienunternehmen stehen unter starkem Kosten- und Zeitdruck. In ihren Archiven liegt Content, der nicht optimal wertgeschöpft wird. Keine andere Branche leistet sich solche Lagerbestände, da liegen riesige Wertstoffe ungenutzt auf Halde. Das gilt übrigens genauso für globale Konzerne. Wer kommuniziert heutzutage nicht über soziale und andere Medien? Coca-Cola, Siemens, Axa, Telekom, das sind längst Content-Fabriken. Kein Unternehmen kommt

mehr ohne mediale Inhalte aus: Für Kunden, Mitarbeiter, Presse, Aktionäre usw. werden ja heute ganze Online-Magazine produziert und bereitgestellt.

Woran liegt es, dass der Content so schlecht online ausgewertet wird?

Gross: Ursprünglich waren es Lizenzfragen; die Urheberrechte schlossen die Web-Nutzung oft generell aus. Inzwischen haben die meisten Verlage nachjustiert. Doch all der Content muss erst mal kategorisiert und auffindbar gemacht werden. Viele Online-Medien, beson-

ders die der Zeitungsverlage, sind per se stark regionalisiert. Da weiß Region A manchmal nicht, was Region B alles im Köcher hat. Zahlreiche Fusionen taten ein Übriges.

Nichts ist so alt wie die Nachricht von gestern. Warum sollten Medienhäuser in ihre Archivbestände investieren?

Koleski: Nicht das Bild oder der O-Ton verliert an Wert, sondern die jeweilige Zusammenstellung. Man muss das Material mit Metadaten versehen, damit es modularisiert und multipel verwendbar ist. Dabei

hilft unsere aingine. Sie zeigt den Journalisten, Autoren oder Regisseuren: Das sind eure vierundzwanzig Videos zu Queen Mom, in fünf davon erscheint sie in der Halbtotale mit grünem Hut und Handtasche. Aber nur in einem lächelt sie. Der Timecode lautet 32:12,14, der Sprechertext an der Stelle ist folgender: „.....“

Worin liegt Ihr Vorteil gegenüber anderen Anbietern?

Prange: Wir automatisieren nicht nur und schaffen so Raum für die Kreativen. Wir helfen auch, Prognosen abzugeben: Wie performt der Content? Wie bringe ich frischen Glanz auf vorhandene Module? Also auf schon bestehende Einstellungen, Bildausschnitte, aber auch ganze Filme. Das gedrehte Material ist ja immer viel reichhaltiger als das verwendete. Wir zeigen: So mache ich aus vermeintlich veraltetem Content einen USP für die Medienmarke.

Können Sie das erklären?

Koleski: Wie häufig kommt es vor, dass Menschen aus dem Nichts heraus prominent werden? Fast nie. Durch KI lässt sich daher exklusives Material generieren. Wenn jemand in der Politik landet, war er vorher vielleicht Schul- oder Pressesprecher. Aus der Region, in der diese Person gelebt hat, existieren Bildvorräte. Bei einem Schützenverein, einem Straßenfest. Mit unserer Technologie *Faice* findet ein Unternehmen auch rückwirkend im Archiv die Bilder dieser Person.

Welche datenschutzrechtlichen Aspekte spielen eine Rolle?

Gross: Grundsätzlich bin ich sehr dafür, dass wir in Europa hohe Datenschutzstandards haben. Ich würde mir wünschen, dass die Politik in manchen Bereichen klarer definiert und sich traut, Grauzonen auszuleuchten, um klare Verhältnisse zu schaffen. Auch kann der Staat mit seinen riesigen Archiven ganz konkret dabei helfen, dass wir Modelle trainieren und wir so als KI-Standort vorankommen. Nach unserer Auffassung sind Unternehmen Content-Anbieter. Ihr Material *als solches* unterscheidet sich wenig von dem der klassischen Medienhäuser. Footage, Hintergrund, alles vergleichbar, zum Teil wird es von Journalisten erstellt.

Wieviel Sorge bereitet Ihnen die Diskussion um das Für und Wider von Kameraüberwachung, nicht nur im öffentlichen Raum?

Gross: Kamera-Erfassung ist kein Selbstzweck.

Nehmen wir das Beispiel der Übergriffe im öffentlichen Raum oder die Hetze auf Fremde. Die Frage ist doch: Was wird als gesellschaftlich höheres Gut angesehen, das Recht auf Sicherheit und Strafverfolgung oder das Recht, auf öffentlichen Plätzen nicht gefilmt zu werden? Das ist jetzt kein Plädoyer gegen eine Diskussion zur Frage: Wollen wir überall „Kamera-erfasst“ werden? Im Gegenteil, ich bin dafür, dass die Diskussion geführt wird, damit über die Verwendung der Daten rechtliche Klarheit entsteht. Durch die DSGVO ist hier auch ganz klar geregelt, wer was aufnehmen darf und wann wir mit KI diese Aufnahmen analysieren dürfen.

Nach Ihren jüngsten Plänen planen Sie einen Vorstoß der aingine in Richtung Vorhersage von Emotionen?

Koleski: Wir sprechen von Aufmerksamkeitsprognose. Dabei spielt Emotion eine Rolle, aber Emotionen sind komplex. Viele Faktoren haben einen Effekt. Dank einer Unterstützung der IFB waren wir in der Lage, unsere Modellanordnung als Prototyp zu zeigen. Das richtige Design dazu gilt es nun zu entwickeln.

Mit welchem Ziel?

Gross: Kleinen und großen Publishern ein Tool an die Hand zu geben, mit denen sie kostengünstig Wirkungsvorhersagen treffen können.

Wann kommt Ihrer Meinung nach die komplett automatisierte Filmgenerierung? Wo alles, also Aufnehmen, Schneiden, Auswerten, von allein erzeugt und auf Knopfdruck eine Nachricht oder ein Feature „ausgespuckt“ wird?

Gross: Teilweise gibt es das schon beim Sport. Ich glaube, dass es zuerst noch eine Zwischenstufe geben wird: Ein Autor macht einen 10-Minuten-Film mithilfe von Hinweisen unseres Superproducers. Der nächste Schritt, Trailer für YouTube, Facebook etc. zu schneiden, vielleicht auch in verschiedenen Schnitffassungen für unterschiedliche Zielgruppen, das wird die KI bald automatisiert machen können. Langfristig glaube ich, dass Kreativität noch lange eine Aufgabe des Menschen sein wird.

In einem Satz: Warum aiconix?

Prange: Wir geben mit der aingine KI-Innovation auf einfache Art und Weise an unsere Kunden weiter.

Das Interview führte Elvira Steppacher.

Die Interviewpartner:



Eugen L. Gross
office@aiconix.ai

Eugen L. Gross hat langjährige Erfahrungen im Bereich Bewegtbildproduktion. Als Kameramann und Produzent kennt er die Herausforderungen der Medienindustrie. Durch ein Executive MBA Studium an der Hamburg Media School entdeckte er sein Interesse an Daten, Artificial Intelligence und der Nutzung dieser bei der Erstellung von Bewegtbild-Content. Bei aiconix ist er für das Business Development sowie für die Produktentwicklung verantwortlich.



Mariel Prange
office@aiconix.ai

Mariel Prange war bereits in diversen Funktionen wie Corporate Development, Strategic Planning, Finance & Planning und Operations bei IBM und Canon Inc. tätig. Infolgedessen hat sie sich als Freiberuflerin in der Beratung kleiner und mittlerer Unternehmen spezialisiert, für die sie, unter anderen, Businesspläne, Durchführbarkeitsstudien, Marktpositionierung und -profilierung ausführt. Als CFO bringt sie wertvolle und weitreichende Kompetenzen in das Unternehmen.



Aleksander Koleski
office@aiconix.ai

Aleksander Koleski studierte Informatik an der Universität Hamburg. In den Nebenfächern Psychologie und Medizin vertiefte er sich in die Gebiete Neurophysiologie und kognitive Psychologie. Seit seinem Studium beschäftigt er sich mit neuronalen Netzen und deren praktische Anwendungen. Bei aiconix ist er für die Bereiche IT, Data Science, Machine Learning sowie die Produktentwicklung verantwortlich.



Cloud-Computing als Fundament für Analytics

Rolf Scheuch, OPITZ CONSULTING

Im Zuge der Digitalisierung der Geschäftsmodelle verschmelzen die eher in die Vergangenheit schauende klassische Business-Intelligence-Welt und das in die Zukunft gerichtete Analyseverständnis von Big Data zu einer gemeinsamen Analytics-Plattform. Diese neue „analytische Plattform“ zeichnet sich durch technologische Vielfalt und stetige Veränderung aus und verbindet in verstärktem Maße unterschiedliche Cloud-Ansätze zu einer hybriden analytischen Infrastruktur. Der Artikel betrachtet die Motive für die Verschmelzung von BI und Big Data und beschreibt einen Blue-Print für eine analytische Architektur. Anschließend geht er näher auf hybride Cloud-Ansätze ein und zeigt, wie Geschwindigkeit, Flexibilität, Skalierbarkeit und Change schon in der Architektur implementiert werden können.

Die Zukunft von BI und Big Data

Reale und virtuelle Welt gehen heute immer mehr ineinander über. Eine radikale Transformation unseres beruflichen und privaten Umfeldes ist bereits in vollem Gange [1]. Gerade die Ex-post-Sicht der Business Intelligence (BI) gerät damit unter Druck, denn der Wert einer Information nimmt in Relation zur Zeit stetig ab [2]. Geschwindigkeit ist gefragt. Nichts ist schließlich so alt wie die Zeitung von gestern!

Zwar hat die IT in den letzten Jahren BI-Lösungen geschaffen, die die Zeitspanne zwischen Verarbeitung und Kommunikation von Informationen reduzieren können. Die Digitalisierung verändert die Anforderungen jedoch noch einmal dramatisch durch ihre Forderung nach Near-Realtime-Analysen und -Nutzung. Somit entfällt der zeitliche Versatz, der den aktuellen BI-Architekturen innewohnt. Dazu kommt, dass der Blick der BI-Welt eher auf die Vergangenheit gerichtet ist. Die Vergangenheit ist bei digitalen Geschäftsmodellen jedoch oft nur ein Teilaspekt: Next-Best-Offer, Predictive Maintenance oder Deep Learning sind als Analyseansätze integraler Bestandteil von operativen Prozessen. Sie verwenden Prognoseergebnisse der Gegenwart zur Steuerung von Entscheidungen.

Parallel dazu verspricht Cloud-Computing „IT-Power aus der Steckdose“. Welche IT träumt nicht davon? Die IT-Kosten beziehen sich damit auf die tatsächliche Nutzung und auf Preismodelle, die nutzungsbezogen abrechnen. Betriebswirtschaftlich ist somit eine Wandlung der Kapitalbindung durch IT-Infrastruktur (CAPEX) in operative Betriebskosten (OPEX) möglich.

Analytics bei der Digitalisierung

Im Folgenden geht es kurz um die grundlegenden Bausteine eines digitalen Geschäftsmodells. Anhand dieser Bausteine zeigt sich, wie dringend ein Regelkreis aus Big Data und BI benötigt wird. Hieraus motiviert sich am Ende die Zusammenlegung zu einer analytischen Infrastruktur.

Ein digitales Geschäftsmodell stellt nur eine Ausprägung eines klassischen Geschäftsmodells dar, mit der Besonderheit, dass die Leistungserbringung von der Nutzung der Informationstechnologie abhängt und der Mehrwert für den Kunden meist ein digitales und somit virtuelles Produkt ist [3]. Doch ganz gleich, welche Klassifikation ein angestrebtes digitales Geschäftsmodell besitzt, es wird in seiner Ausprägung grundlegende fachliche

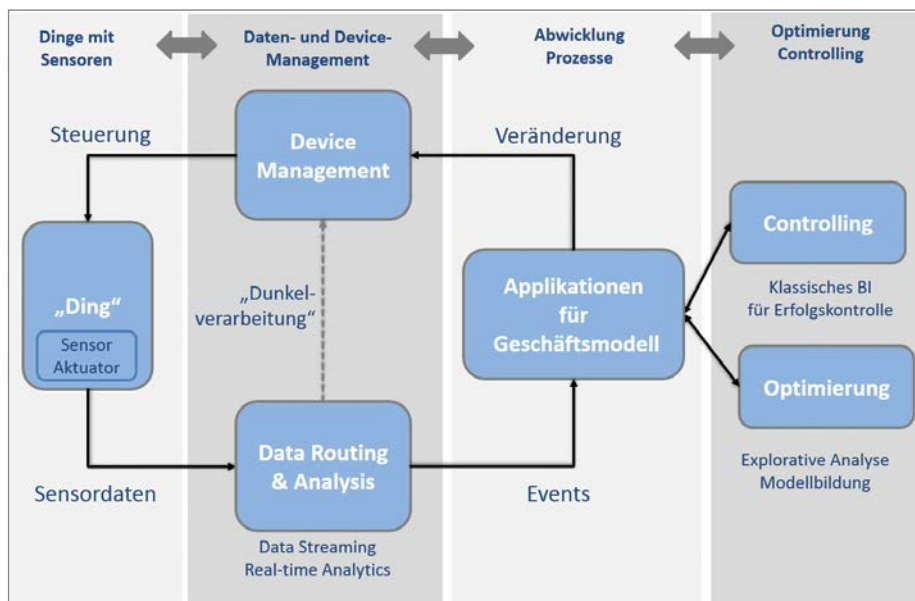


Abbildung 1: Fachliche Komponenten eines digitalen Geschäftsmodells © OPITZ CONSULTING

Komponenten besitzen. Diese Komponenten sind in *Abbildung 1* aufgeführt.

Der Wertschöpfungsprozess eines digitalen Geschäftsmodells beinhaltet einen „lernenden Regelkreis“. Dieser beginnt und endet mit der Nutzung des Gegenstands oder des Endgeräts. Verhalten und Zustand des Produkts werden durch die Sensorik seiner Daten nach außen sichtbar [4].

Die übermittelten Sensordaten der Endgeräte werden im Rahmen des Datenmanagements aufbereitet. Aufgrund der großen Menge und Variabilität der Sensordaten wird die Verarbeitung in der Regel als Fast-Data [5] implementiert. Die notwendige Geschwindigkeit bei der Datenaufnahme und Verarbeitung wird über Architekturen erreicht, die sich über mehrere Ebenen des Netzwerks erstrecken. Was das Datenvolumen angeht, helfen die Architekturkonzepte des Big Data weiter. Die Streaming-Daten werden in Near Realtime gefiltert und analysiert, wobei bereits eine lokale Intelligenz zu einer Veränderung der Nutzung führen kann (siehe auch „Dunkelverarbeitung“ in *Abbildung 1*.)

Im Mittelpunkt vieler digitaler Geschäftsmodelle steht oft ein Verbund an Endgeräten, etwa Beleuchtungseinheiten, Parkplätze, Mietobjekte, Connected Car etc., deren Verhalten gemeinsam an die veränderten Umweltbedingungen angepasst werden muss. Hierzu werden fachliche Nachrichten an ein lokales oder zentralisiertes Device Management versandt. Das Device Management ist nun in der Lage, die logischen Events der unterschiedlichen Endgeräte auf-

zubereiten und im Netzwerk zu propagieren, um das Verhalten der Endgeräte durch eine Ansteuerung der Aktuatoren zu beeinflussen. Diesen Regelkreis zusammen mit dem Datenmanagement stellt der Kasten „Daten- und Device Management“ in *Abbildung 1* dar. Die Frontend- und Backend-Applikationen des digitalen Geschäftsmodells nutzen anschließend die aufbereiteten Daten zur Etablierung des neuen Angebots oder eines Mehrwerts für den Kunden.

Abbildung 2 zeigt den Regelkreis, der durch Analytik-Maßnahmen geschlossen wird. Diese Maßnahmen umfassen neben der Erfolgsmessung, die in Form eines klassischen Controllings über einer BI-Welt stattfindet, auch eine explorative Erkennung von Mustern, etwa des Kunden- und Produktverhaltens. Ziel ist eine Optimierung des digitalen Geschäftsmodells. Die Optimierung erfolgt meist in einem explorativen Ansatz, der zeitlich eher unkritisch ist – einem sogenannten „Batch-orientierten Big-Data-Ansatz“. Hierzu zählen statistische Prognosemodelle ebenso wie die Ansätze des Deep Learning [6]. Die Modelle werden fortlaufend anhand von Testfällen aus der Praxis überprüft und somit kontinuierlich verbessert. Die Erkenntnisse daraus können verwendet werden, um entsprechende Regelwerke auf den Streaming-Daten für eine Near-Realtime-Filterung und -Prüfung zu verwenden. Das eher klassische Controlling rundet den Regelkreis ab und liefert in einer Ex-post- und betriebswirtschaftlichen Sicht Informationen zum Geschäftsmodell selbst.

Die analytische Plattform

Abbildung 3 verdeutlicht das Zusammenspiel der Ansätze von Big Data sowie klassischer BI-Welt und beschreibt die Architekturkomponenten.

Sowohl Near-Realtime-Entscheidungsfindung, klassisches BI als auch Modellbildung für Prognosen fließen beim Aufbau einer analytischen Architektur mit ein. Es wird also deutlich, dass sowohl eine klassische BI/DWH-Sicht als auch eine reine Big-Data-Sicht für sich genommen jeweils unzureichend wären. Auf eben dieser Annahme baut die beschriebene Referenzarchitektur auf. Dieser „analytischen Architektur“ liegt die Annahme zugrunde, dass mittelfristig die klassische, eher Ex-post-orientierte Business Intelligence mit Data-Warehouse-Ansätzen und die neuen Big-Data-Ansätze zu umfassenden Architekturen verschmelzen werden. Auch wenn beide Ansätze hinsichtlich der technischen Implementierung äußerst verschieden sind, so ist die betriebswirtschaftliche Aufgabenstellung bei beiden identisch: Beiden geht es darum, aus Daten Informationen zu gewinnen und das Geschäft durch Analysen besser steuern zu können.

Am Beispiel eines Geschäftsmodells zum Verleih von Fahrrädern wird diese Verschmelzung gut sichtbar: In Near Realtime analysiert ein Unternehmen die Sensordaten seiner Fahrräder und erkennt Muster, die auf eine notwendige Reparatur schließen lassen. Hierzu generiert das System einen Serviceauftrag an das Backend-

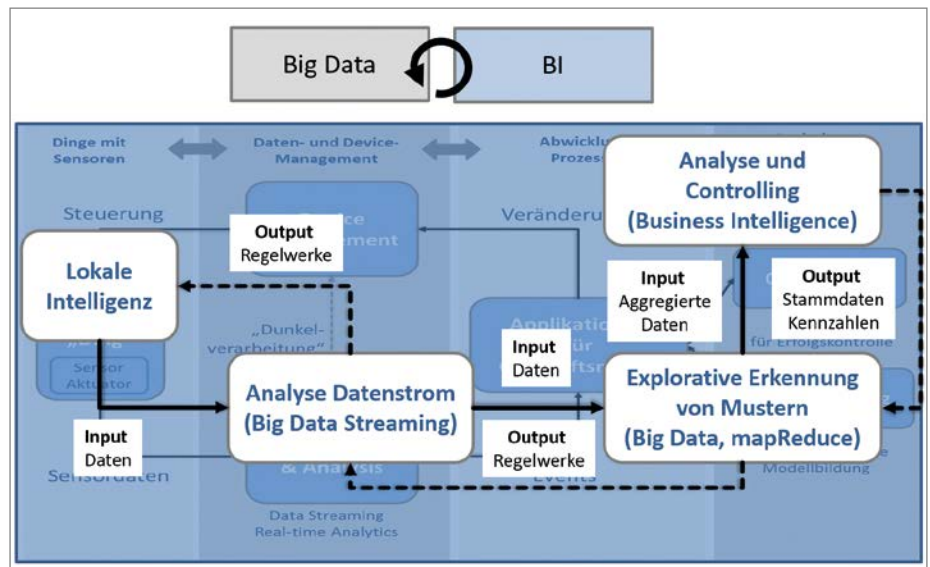


Abbildung 2: Der analytische Regelkreis © OPITZ CONSULTING

Service-Modul, das als SaaS-Lösung implementiert sein kann. Des Weiteren gelingt es, mit zeitlichem Versatz in Batch-orientierten statistischen Verfahren die Positionen der Fahrräder mit ihrer Mietdauer sowie mit zukünftigen Events und Großereignissen zu korrelieren. Aus dieser Korrelation ergeben sich sogenannte „Lokalisierungsaufträge“, die generiert werden, um Fahrräder an bestimmte geographische Punkte in die Nähe potenzieller Kunden zu bringen und damit bessere Ausleihquoten zu erreichen. Letztendlich fließen die Fahrradinformationen nebst Daten aus den betriebswirtschaftlichen Systemen (Materialwissenschaften, Finanzbuchhaltung, Service etc.) aggregiert

in ein Data Warehouse und ermöglichen so ein umfassendes finanzielles Controlling. Ferner werden die Batch-orientierten Ansätze verwendet, um neue Muster zu erkennen und diese, so möglich, auf dem Stream für eine Near-Realtime-Verarbeitung zu implementieren. Das Fallbeispiel zeigt, wie klassische BI-Ansätze und neue Big-Data-Ansätze mit den Spielarten der Batch-orientierten explorativen Simulation mit statistischen Modellen und der Near-Realtime-Analyse des Datenstroms erst in ihrer Kombination zu wesentlichen Erfolgsfaktoren werden.

Die geschilderten fachlichen Komponenten benötigen unterschiedliche System-

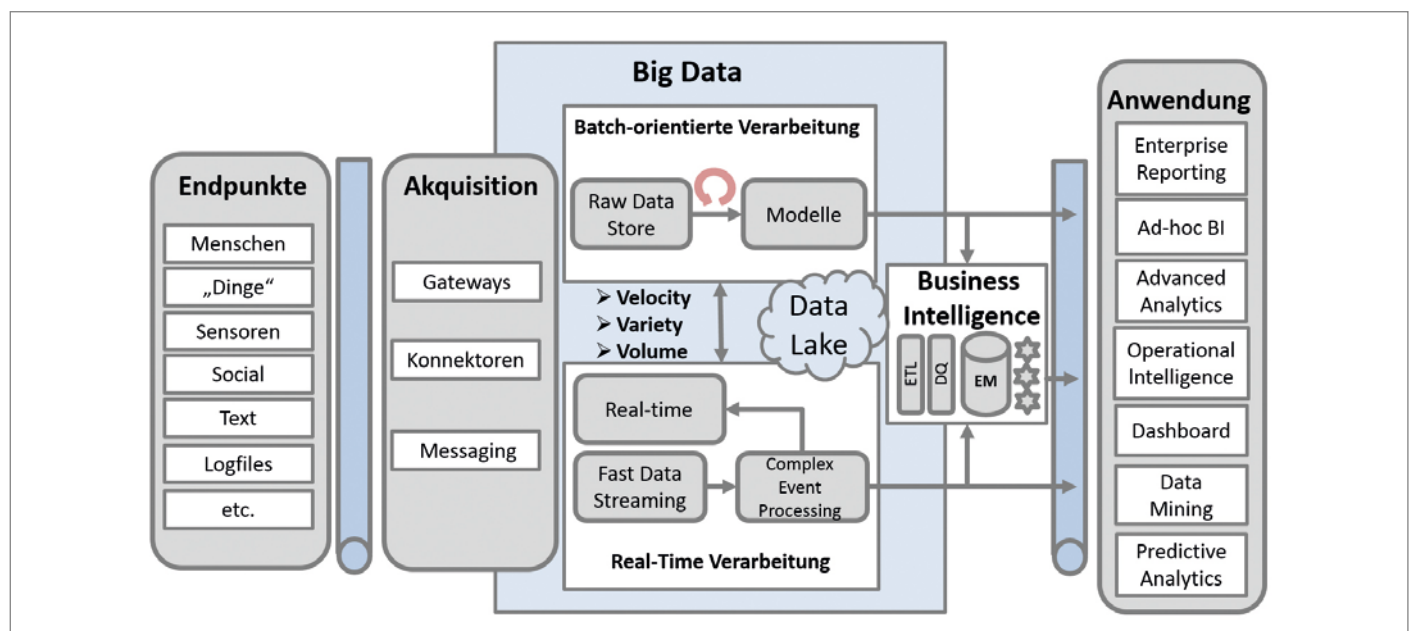


Abbildung 3: Analytische Architekturen bei der Digitalisierung © OPITZ CONSULTING

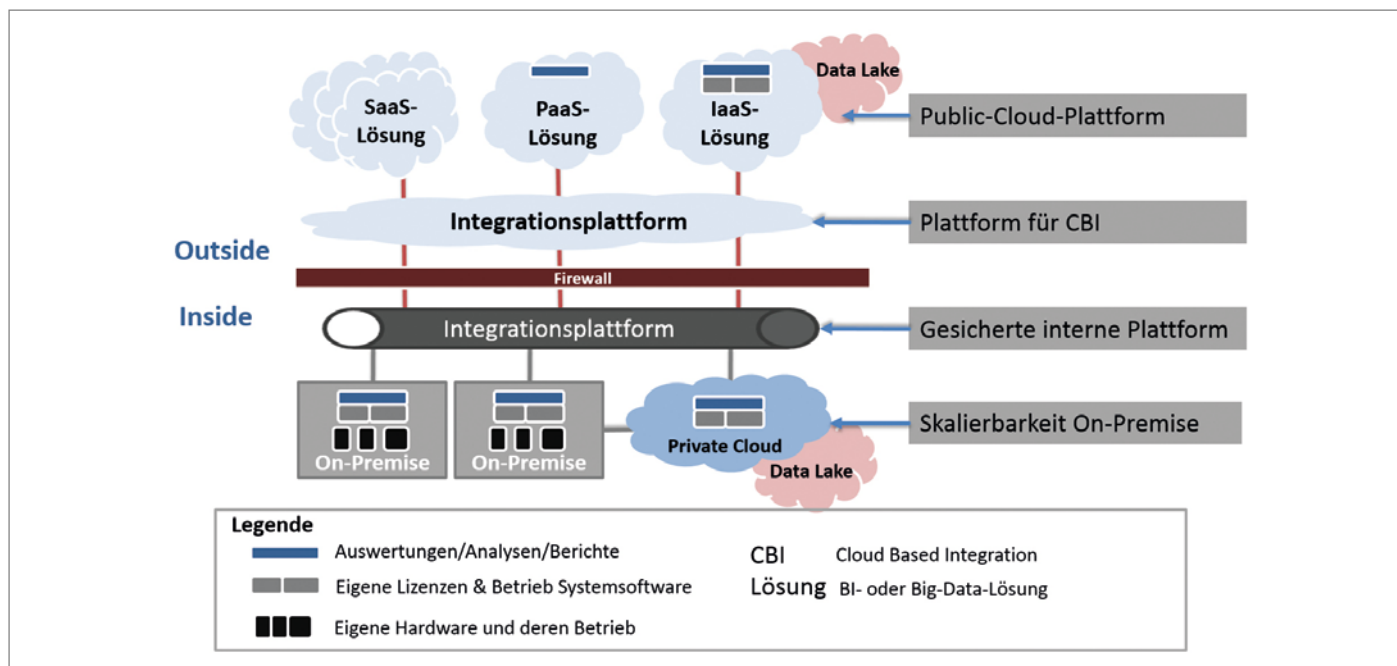


Abbildung 4: Zukunftsweisende hybride Infrastrukturen © OPITZ CONSULTING

infrastrukturen. Daher bilden hybride Infrastrukturanätze unter Einbeziehung von Cloud-Lösungen ein passendes Fundament für die analytische Architektur.

Hybride Cloud-Lösungen

Die meisten Unternehmen nutzen bereits eine Vielzahl von Cloud-Lösungen mit unterschiedlichen Liefer- und Servicemodellen. Dies gilt für BI-Branchenlösungen, die vermehrt als SaaS-Lösungen angeboten werden, aber auch für PaaS-Lösungen für Big Data oder BI-Suiten. So ist es nicht verwunderlich, dass ein ganzheitlicher Ansatz für analytische Lösungen notwendig wird. Der ganzheitliche Ansatz verschafft hybriden Ansätzen bei analytischen Lösungen bei allem Wildwuchs eine ordnende Sicht.

Anstatt auf die sich so entwickelnde Dezentralität mit Bestrebungen zur Zentralisierung und Standardisierung zu antworten und somit wieder Geschwindigkeit und Innovation zu hemmen, kann die IT im Rahmen hybrider Cloud-Lösungen einen „Design for Change“-Ansatz verfolgen und die analytische Architektur auf einer reaktiven Infrastruktur implementieren.

Bei dezentralen Ansätzen treten schnell Herausforderungen an die Datenqualität auf. Insbesondere sind Inkonsistenzen bei der Nutzung von Stamm-, Meta- und Referenzdaten zu beobachten. In der Folge sinkt das Vertrauen in die Analysen und die dezentralen Plattformen geraten unter Druck.

An dieser Stelle liefert die klassische BI-Welt einen Mehrwert als Lieferant von qualitätsgesicherten Daten. Voraussetzung hierfür sind allerdings eine übergreifende Governance beim Datenmanagement sowie die Transparenz über die Nutzung relevanter zentraler Daten.

Abbildung 4 stellt die unterschiedlichen Implementierungsansätze inklusive der Cloud-Ansätze für analytische Lösungen in einem Blue-Print für eine hybride Infrastruktur der analytischen Welt dar. Hier gibt es kein Richtig oder Falsch, sondern Entscheidungen müssen situativ auf Basis aktueller Anforderungen getroffen werden.

Abbildung 4 zeigt unterschiedliche Szenarien für die Implementierung einer BI- oder Big-Data-Lösung:

- Nutzung von Public-Cloud-Lösungen ohne eigene Lizenzen
 - als SaaS-Lösung
 - als PaaS-Plattform für BI oder Big Data
- Nutzung eigener BI- und/oder Big-Data-Lizenzen
 - in einer Public-aaS-Lösung
 - in einer Private-Cloud-Lösung
 - On-Premises auf der eigenen Infrastruktur
- Nutzung einer Integrationsplattform für die Datenbewirtschaftung
 - in einer Public Cloud (Cloud Based Integration)
 - On-Premises

Für eine Cloud-Lösung sprechen einige Vorteile: Beim Betrieb einer Cloud-BI-Lösung mit eigenen Lizenzen im Rahmen einer IaaS-Lösung ist die Unternehmens-IT mit den gleichen IT-Service-Prozessen belastet wie bei einer On-Premises-Lösung. Im Gegensatz dazu entfallen bei einer PaaS-Lösung mit nutzungsabhängiger Abrechnung die direkten Lizenzkosten ebenso wie die Verantwortung für wesentliche IT-Service-Prozesse. Jedoch müssen bei PaaS (wie auch bei SaaS) die internen Verrechnungsalgorithmen angepasst werden.

Bei Pilot-Projekten oder einem Proof-of-Concept (PoC) hat ein Cloud-Ansatz Vorteile für die Time-to-Value [7]. In vielen Fällen ermöglichen die Anbieter von Cloud-Lösungen

- eine schnelle Bereitstellung der benötigten Systemressourcen inklusive der Software selbst,
- eine umfassende Testumgebung,
- eine nutzungsabhängige Abrechnung,
- die leichtere Einführung von modernen Lean-Startup-Ansätzen nach dem „Fail fast, fail early“-Prinzip und deren Nutzung für den PoC.

Unternehmen beginnen verstärkt mit der Implementierung von Data Lakes [8][9]. Eine Vielzahl von neuartigen Daten werden in die Analysen einbezogen: Sensordaten, Lokalisierungsdaten, Geodaten, Text Messages, Log File etc. Die Heterogenität dieser

Daten, die Variabilität der Datenstrukturen und die schnellen Zyklen der Formatänderungen sind mit den klassischen Datenbewirtschaftungsprozessen aus der BI-Welt schwerlich realisierbar. Ferner lassen diese Daten sich nur mit hohem Aufwand in die notwendigen relationalen Strukturen wie zum Beispiel Star-Schemata überführen. An dieser Stelle sind NoSQL-Datenhaltungen [10] auf dem Vormarsch und die Data-Lake-Konzepte lassen die Ablage der Daten in heterogenen Datenhaltungen zu.

Neben den betriebswirtschaftlichen Überlegungen hinsichtlich Investitionsschutz und Abschreibungszeiträumen sind am Ende vor allem zwei Kriterien bei der Entscheidung für oder gegen die Cloud-Lösung ausschlaggebend: der Datenschutz und die Kosten.

Datenschutz

Beim Datenschutz ist es notwendig, die gewachsenen, oft schon veralteten Datenschutzbestimmungen und Regelungen zu hinterfragen, um die Entscheidung gegen eine Cloud-Lösung nicht vorschnell zu treffen. Dabei empfiehlt es sich, keine zentrale übergreifende Entscheidung für alle BI- und Big-Data-Lösungen zu treffen, sondern zunächst die relevanten Datenobjekte für den analytischen Bereich zu definieren und daraufhin eine Entscheidung für das jeweilige Sub-Set zu treffen. Abgesehen davon sind alle in *Abbildung 4* aufgeführten Spielarten der Implementierung realistisch und sollten geprüft werden.

Kostentreiber

Ein wesentlicher Kostentreiber ist der Preis für die Datenspeicherung und den Datenexport von einer Cloud-Infrastruktur in eine andere Infrastruktur. Je größer die Datenmenge ist, die aus der Cloud in BI- und Big-Data-Systeme fließt, desto preiswerter wird eine Inside-Lösung im Unternehmensnetzwerk sein. Aktuell sind die Preismodelle der Cloud-Anbieter nur für konkrete Anwendungsfälle vergleichbar.

Eine Besonderheit besteht in der Near-Realtime-Verarbeitung durch Streaming-Lösungen, Fast-Data-Ansätze, Complex Event Processing und Realtime-Analytik, wie in *Abbildung 3* aufgeführt. Diese Lösungsansätze findet man häufig beim Internet der Dinge mit einer Vielzahl heterogener Sensordaten. Die IoT-Plattformen selbst liegen in der Regel schon außerhalb des Unternehmensnetzwerks und die Security bezieht sich auf einen Zugriffsschutz zu den

Endgeräten und das Einspielen fehlerhafter Daten auf die IoT-Plattform. Normalerweise besteht kein ausgeprägter Datenschutz. Für den Fall eines Datenverlusts bietet sich für das Backup eine Cloud-Lösung als Plattform an.

Für das Balance-Verhältnis zwischen On-Premises-Lösungen und Cloud-Lösungen steht das „Center of Gravity“ [11]. Es setzt den Anteil der Cloud-Lösungen in Bezug zu den On-Premises-Installationen. Aus diesem Verhältnis lassen sich wichtige Schlüsse ziehen, denn: Je höher der Einsatz unterschiedlicher Cloud-Lösungen ist, desto eher verschiebt sich auch die Integrationsplattform in die Cloud und rückt damit näher an den Ursprung der Daten.

Fazit

Die Verbreitung von Cloud-Lösungen ist unaufhaltsam. Unterschiedliche Liefer- und Servicemodelle nehmen Einzug in die Unternehmen, hybride Ansätze beim Cloud-Computing werden die Regel. Die Client/Server-Revolution hat uns gelehrt, die Verbreitung der Cloud-Lösungen seitens der zentralen IT zu unterstützen, anstatt als „Bremser“ von Innovation aufzutreten.

Auch der Big-Data-Zug rollt. Und für eine zentrale BI-Einheit im Unternehmen ergibt es wenig Sinn, die Big-Data-Welle einzudämmen. Wie geschildert ergänzen sich die beiden Facetten analytischer Fragestellungen gerade im Hinblick auf die kontinuierliche Verbesserung von digitalen Geschäftsmodellen. Aus diesem Grunde werden die zentralistische BI-Welt und die eher föderale Big-Data-Welt zu einer neuartigen analytischen Plattform verschmelzen. Das Thema der Governance der analytischen Plattform wird in dem Zusammenhang ein neues, wichtiges Handlungsfeld.

Auch etablierte Denkmuster der zentralen Plattform werden sich wandeln. Zu dynamisch ist der Markt und zu groß sind die Chancen neuer Geschäftsmodelle durch neue Technologien. Der „Design for Change“-Gedanke hält Einzug in die Denkmuster und Cloud-Computing wird zu einem entscheidenden Hilfsmittel, um die Time-to-Value zu reduzieren und Flexibilität zu erhöhen. Cloud-Computing wird ein Fundament der neuen analytischen Plattform werden.

Der Artikel von Rolf Scheuch erschien erstmals unter dem Titel „Die Zukunft von BI und Big Data“ im Rahmen der Publikation „Big Data

& Analytics: KI und Maschinelles Lernen auf dem Vormarsch“ (eBook, deutsch, 70 Seiten, PDF 13,2 MB, IT Verlag für Informationstechnik GmbH 2019). Wir danken für die freundliche Genehmigung.

Quellen und Anmerkungen

- [1] Tim Cole: Digitale Transformation, Franz Vahlen Verlag 2015
- [2] Dominik Bial: IoT erfordert neues Denken, BI-Spektrum 02-2016
- [3] Thomas Bieger, Stephan Reinhold: Innovative Geschäftsmodelle: Konzeptionelle Grundlagen, Gestaltungsfelder und unternehmerische Praxis. 2011 in: T. Bieger, D. zu Knyphausen-Aufseß, C. Krys (Hg.), Innovative Geschäftsmodelle, Springer Verlag 2011
- [4] Ferner kann der Gegenstand auch ein Smart Device wie etwa auch ein Tablet oder Smartphone sein.
- [5] BITKOM, Frauenhofer (Hg.): Big-Data-Technologien – Wissen für Entscheider, <https://www.bitkom.org/Publikationen/2014/Leitfaden/Big-Data-Technologien-Wissen-fuer-Entscheider/140228-Big-Data-Technologien-Wissen-fuer-Entscheider.pdf> (Download Juni 2015)
- [6] Einen guten Einstieg erhält man auf dieser Website: <http://deeplearning.net/> (Download 10/2016)
- [7] Nello Franco: Time To Value: A Key Metric, <https://blog.nellofranco.com/2013/08/25/time-to-value-a-key-metric/> (Download 10/2016)
- [8] Tamara Dull: Data Lake vs Data Warehouse: Key Differences, <http://www.kdnuggets.com/2015/09/data-lake-vs-data-warehouse-key-differences.html> (Download 10/2016)
- [9] Matthias Reiss und Stephan Reimann: Das Data Lake Konzept: Der Schatz im Datensee, <https://www.it-daily.net/it-management/big-data-analytics/11222-das-data-lake-konzept-der-schatz-im-datensee> (Download 10/2016)
- [10] Eine Übersicht und Einführung findet sich unter <http://nosql-database.org/> (Download 10/2016)
- [11] Jim Harris: The Cloud is shifting our Center of Gravity, <http://www.ocdqblog.com/home/the-cloud-is-shifting-our-center-of-gravity.html> (Download am 28.5.2015)



Rolf Scheuch

rolf.scheuch@opitz-consulting.com

Rolf Scheuch ist Diplom-Mathematiker, geschäftsführender Gesellschafter und Mitbegründer von OPITZ CONSULTING; er verantwortet seit 2011 als Chief Strategy Officer die strategische Ausrichtung der Unternehmensgruppe. Heute arbeitet er zudem als Management-Coach, hält Fachvorträge und ist Autor diverser Bücher und Publikationen. Schwerpunkt des Management-Coachings der letzten Jahre war die veränderte Rolle der IT durch die Digitalisierung mit den spezifischen Themen Agilität, Right Sourcing, föderale Governance und Innovationsfähigkeit der IT.



Machine und Deep Learning mit Data Science in der Oracle Cloud

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

Machine Learning und Deep Learning (ML+DL) sind Analyseprozesse, um über Software-gestützte Algorithmen Vorhersageregeln (Modelle) zu entwickeln. Mächtige Hardware, optimierte Algorithmen, die neu entstandene Rolle des Data Scientist und die Digitalisierung von nahezu allen Lebensbereichen lassen die schon recht alte Aussage von den Daten als wichtigste Unternehmensressource heute tatsächlich Wirklichkeit werden, und sie klingt nicht mehr wie eine abgedroschene Floskel. Eine Cloud-Lösung ist dabei nicht einfach nur technische Infrastruktur. Cloud erhöht die Effizienz von Data Science bei Machine-Learning- und Deep-Learning-Analyse-Workflows durch Flexibilität und die Möglichkeit einer heterogenen Datenlandschaft. Wer Machine-Learning- oder Deep-Learning-Analysen bislang manuell und mit individuell gestrickten Umgebungen programmiert hat und die Aufwände hiervon kennt, wird sich vor allem über die vielen Automatismen des neuen Oracle Data Science Cloud Service freuen.

Herausforderungen im Machine-/Deep-Learning-Analyse-Workflow

Unter ML+DL-Analysen fassen wir zunächst Supervised-Learning-Verfahren zusammen, mit dem Ziel, aufgrund bekannter Informationen und bereits eingetretener Ereignisse Vorhersagen für künftige Ereignisse vornehmen zu können. Die Verfahren reichen von altbekannten Methoden wie *Support Vector Machines*, *K-Nearest Neighbors*, *Decision Trees* bis hin zu neueren Methoden wie *LGBM/XDBClassifier*, *RandomForest* oder neuronalen Netzen mit der Ausprägung Deep Learning.

Doch hier entsteht bereits eine erste Herausforderung: Niemand sagt dem Analysten, welcher der vielen Algorithmen für welche Aufgabenstellung und Datenart der passende ist. Das hat weniger mit Unkenntnis des Analysten zu tun. Der Grund liegt eher in den immer wieder unterschiedlichen Datenbeständen. So verschlingt die Phase der Datengewinnung und Datenvorbereitung nach Umfragen mehr als drei Viertel der Projektzeit. Unvollständige Daten, zu starke Korrelationen zwischen Input-Variablen, zu wenig diversifizierte Merkmalsausprägungen sind nur einige der Probleme. Hinzu kommen Anpassungen an die von den Algorithmen geforderten Formate.

Die nächste Herausforderung ist die Sicherstellung der Vorhersage-Güte der entwickelten ML+DL-Modelle. Es ist entscheidend, dass man erklären kann, wie eine Vorhersage-Empfehlung zustande kommt. Nur dadurch entsteht Vertrauen bei der Anwendung der Analyse-Ergebnisse und dies sichert den Erfolg eines ML+DL-Projektes.

Eine ML/DL-Analyse wird als Workflow umgesetzt. Dieser umfasst die in *Abbildung 1* zu sehenden Schritte.

Viele dieser Schritte sind automatisierbar. Und hier kommt die neue Data-Science-Cloud-Lösung von Oracle ins Spiel.

Die „Spagat“-Herausforderung von Data Science

Bevor wir tiefer einsteigen, hier kurz ein paar Worte zu den besonderen Herausforderungen der Rolle „Data Scientist“. Ein

Data Scientist benötigt Fertigkeiten aus mehreren Disziplinen: Einerseits muss er die Betriebsprozesse und den betriebswirtschaftlichen Hintergrund zu den Prozessen verstehen, um die zu analysierenden Daten besser einschätzen und Projektergebnisse bewerten zu können. Andererseits muss er Daten nach Belieben selektieren und anpassen können. Dazu ist technisches Know-how bis hin zur Programmierung nötig. Weit verbreitet sind hier mittlerweile *Python* und das Arbeiten in einem Notebook, wie *Jupyter* oder *Zeppelin*.

Eine weitere Herausforderung sind die von Fall zu Fall wechselnden Infrastrukturanforderungen. Die zu analysierenden Daten sind meist räumlich verteilt oder versteckt in unterschiedlichen Systemen oder Datenbanken. Eine einmalig konfigurierte Arbeitsumgebung reicht nicht. Hier hilft das Cloud-Konzept, wonach je nach Anforderung und Aufwand zielgerichtet Komponenten spontan bereitstehen.

Oracle Data Science Cloud Service

In den letzten zwei Jahren hat Oracle eine konsequente Umsetzung der genannten Workflow-Schritte in der Oracle Cloud Infrastructure (OCI) vorgenommen. Ein wichtiges Ziel war die Automatisierung von Workflow-Schritten, um Zeit und Aufwand zu sparen. Ein Data Scientist soll sich im Wesentlichen um Algorithmen und Modellbewertung und weniger um Programmierung, Datenzugriffe oder Datentransformationen kümmern.

Dieser neue Cloud-Service ist Bestandteil der OCI und verfügt über alle Cloud-Ressourcen. Das sind Storage- und Speicheroptionen wie Cloud-Storage, Datenbanken, aber auch Big Data Cluster und Streaming-Dienste. Die benötigte Rechenleistung ist wahlfrei und reicht von kleinen Zwei-Core-Maschinen bis hin zu großen Bar-Metal-Maschinen oder GPU-basierten Rechenmaschinen.

Oberfläche? Weit gefehlt!

Derjenige, der ein klassisches Produkt mit separater Oberfläche erwartet, wird enttäuscht sein. Ein Data Scientist, der ge-

wohnt ist, mit *Python* und *Jupyter Notebook* zu arbeiten, wird sich freuen, denn genau das findet er in dem Data Science Cloud Service (DS-Service).

Die Stärke des DS-Service wird durch eine Detail-Betrachtung der einzelnen Workflow-Phasen deutlich. Es überrascht besonders durch den Umfang der vielen Automatisierungen.

Schritt 1: Gewinnen von Daten

In Zeiten von Big Data und Data Lakes kann das Gewinnen von Daten eine zeitraubende Tätigkeit sein. Es ist nicht damit getan, eine Excel-Tabelle einlesen zu können. In Data Lakes liegt alles Mögliche in unterschiedlichen Formaten und in der Regel nicht aufbereitet. Kombinieren wir diesen Umstand mit dem Fakt, dass der Analyst meist nicht weiß, welche Daten er überhaupt benötigt und überall „mal eben kurz hinschauen“ muss, wird die Wichtigkeit dieses Schrittes bewusst.

Die DS-Lösung hat alle Datenzugriffsanforderungen in dem Modul *DatasetFactory* zusammengefasst. Dieses Modul beinhaltet:

- Lesen und Schreiben der **wichtigsten Datenformate** wie CSV, TSV, Parquet, Libsvm, JSON, Excel, HDF5, SQL (verschiedene Datenbanken), xml, Log-Files usw. Interessant ist die Fähigkeit, Files mit einem einzigen Aufruf in ein anderes Format zu überführen, also Lesen von CSV und Schreiben nach Parquet oder Lesen von XML und Schreiben nach JSON. Um Missverständnissen vorzubeugen: Es geht nicht darum, dass man dies mit Programmiermitteln überhaupt kann. Es geht darum, es mit einem einzigen Aufruf und spontan zu erledigen.
- Lesen von den diversen **physischen Ablageorten** wie lokales File-System, Google Cloud Storage, AWS S3, Object Storage, SQL Server, Oracle DB, noSQL u.v.a.m.
- **Data Sampling**. Große Dateien will man ungern auf einen Schlag einlesen. Über einen Schwellwert steuert man, ab welcher Anzahl von Sätzen ein Sampling

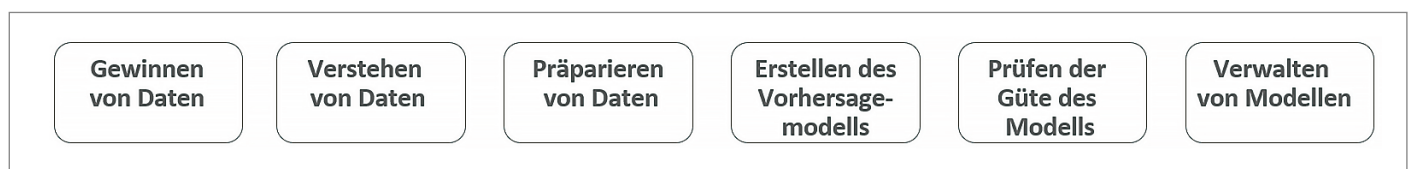


Abbildung 1: Der Workflow einer ML/DL-Analyse © Alfred Schlaucher

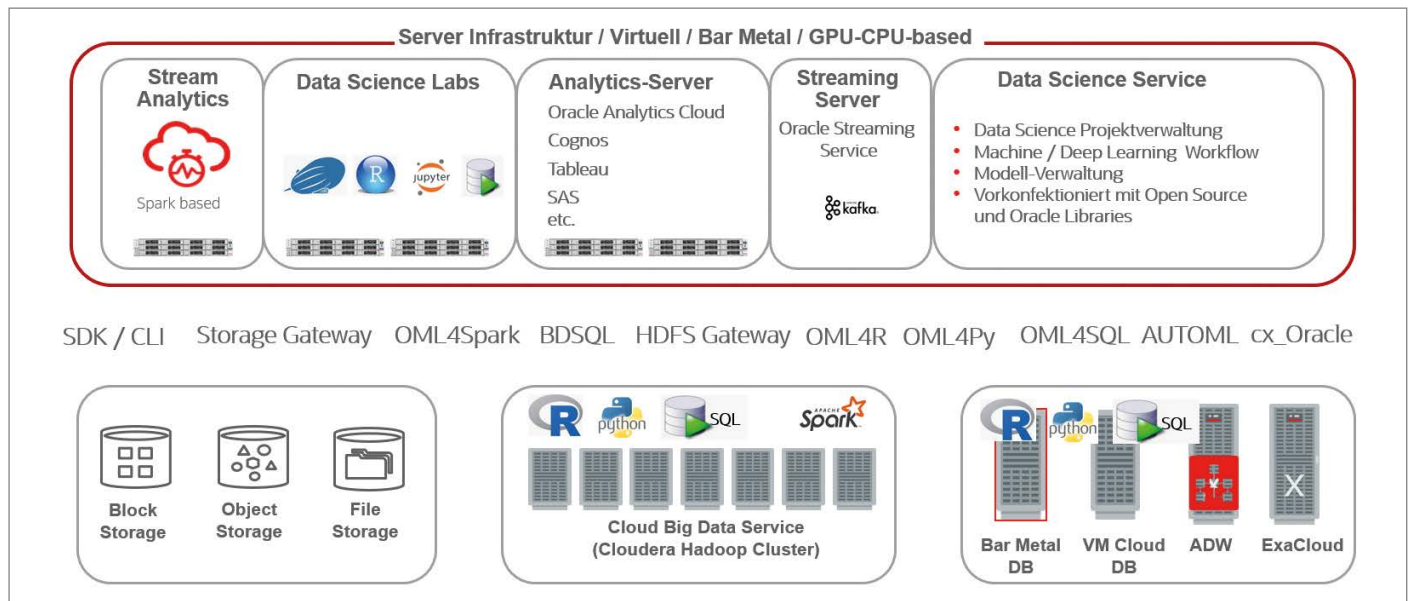


Abbildung 2: Analyse-Umgebung für Data Science und Analytics © Alfred Schlaucher

beginnen soll. Es ist also auch möglich, mit riesigen Dateien zu arbeiten. Allerdings kostet das in der Erkundungsphase sehr viel Zeit.

- **Data Profiling.** Die Daten werden bereits beim Lesen im Sinne von Data Profiling analysiert. Das heißt, unmittelbar nach dem Lesevorgang stehen die wichtigsten Metadaten der Datei und Strukturmerkmale der einzelnen Spalten/Variablen zum Abfragen bereit.

Schritt 2: Verstehen von Daten

Auch wenn man sich in einem *Python Notebook* befindet, arbeitet man dennoch interaktiv. Ein eingelestes Daten-Objekt kann man jetzt interaktiv bezüglich Struktur und Werteverteilung betrachten. Dabei helfen folgende Verfahren:

- Über Funktionen wie *type_of* fragt man die Struktur der Daten ab.
- Die Funktion *show_in_notebook* zeigt mit einem Aufruf:
 - ein Summary des kompletten Datasets,
 - AutoPlot der einzelnen Attribute/Features entsprechen der Datenstruktur (abhängig von continuous/categorical/binary/multi-class),
 - graphische Anzeige des Correlations-Grades zwischen Features,
 - Daten-Sampling-Ausschnitt (Realdaten).

Schritt 3: Präparieren der Daten

Nicht alle Datenattribute sind Analyse-relevant. Indizes, starke Korrelationen, Attribute mit lediglich einem oder gar fehlenden

Werten tragen in der Regel nicht zu einem sinnvollen Vorhersageergebnis bei. Auch dies wird automatisch bereinigt:

- Die Funktion *get_recommendations* zeigt solche Schwachstellen auf:
 - Löschung von Konstanten-Spalten und Primary-Key-Feldern
 - Ersetzen von NULLs durch „angenommene“ Werte
 - Löschen von stark korrelierenden Spalten
 - Ausbalancieren von Skewness
- Die Funktion *auto_transform* kann die Ergebnisse dieser Schwachstellenanalyse automatisch, also ohne weitere Interaktion, bereinigen. Will man das nicht, so klickt man sich interaktiv durch einen Dialog mit möglichen Bereinigungsmaßnahmen durch.

Maßnahmen wie Skalieren oder Transponieren usw. bleiben einem späteren Schritt vorbehalten, da die Notwendigkeit für solche Transformationen abhängig von den später noch zu wählenden Algorithmen ist.

Schritt 4: Erstellen des Vorhersagemodells

Erst jetzt startet die eigentlichen ML+DL-Analyse. Der Analyst kann mehrere Wege wählen:

1. Für einen ersten „Schnellschuss“ bietet sich *AutoML* an. *AutoML* wählt anhand der vorliegenden Daten eine Reihe von

voreingestellten Algorithmen zur Modellerstellung, führt ein Scoring der Modellgüte durch und schlägt den für die Daten am besten geeigneten Algorithmus in einem Ranking vor. Dieser Weg wird für die meisten Anwendungsfälle ausreichende Ergebnisse erzielen. Gerade für Analysten mit wenig langjähriger Erfahrung oder bei wenig zur Verfügung stehender Projektzeit ist dieser Weg vorzuziehen.

2. Alternativ dazu kann man natürlich mit *Python* selbst den Aufruf möglicher Algorithmen programmieren. Diese klassische Vorgehensweise ermöglicht das individuelle Setzen von Parametern für die Algorithmen. Aber auch hier ist es Praxis, nicht nur einen Algorithmus, sondern eine Auswahl unterschiedlicher Algorithmen durchzuspielen.

3. Selbstverständlich ist auch eine Kombination dieser beiden Wege sinnvoll: Zunächst lässt man *AutoML* eine Reihe von Algorithmen automatisch prüfen, um sich einen für seine Daten besonders geeigneten Algorithmus vorschlagen zu lassen. Diesen nimmt man dann und programmiert ihn mit den eigenen Wunschparametern aus. Ein anschließender Vergleich der Vorhersagegüte zeigt, wer „besser war“.

In allen Varianten wird man allerdings die Vorhersagegenauigkeit der gefundenen Modelle im Nachgang überprüfen. Das erledigt man in einem Folgeschritt.

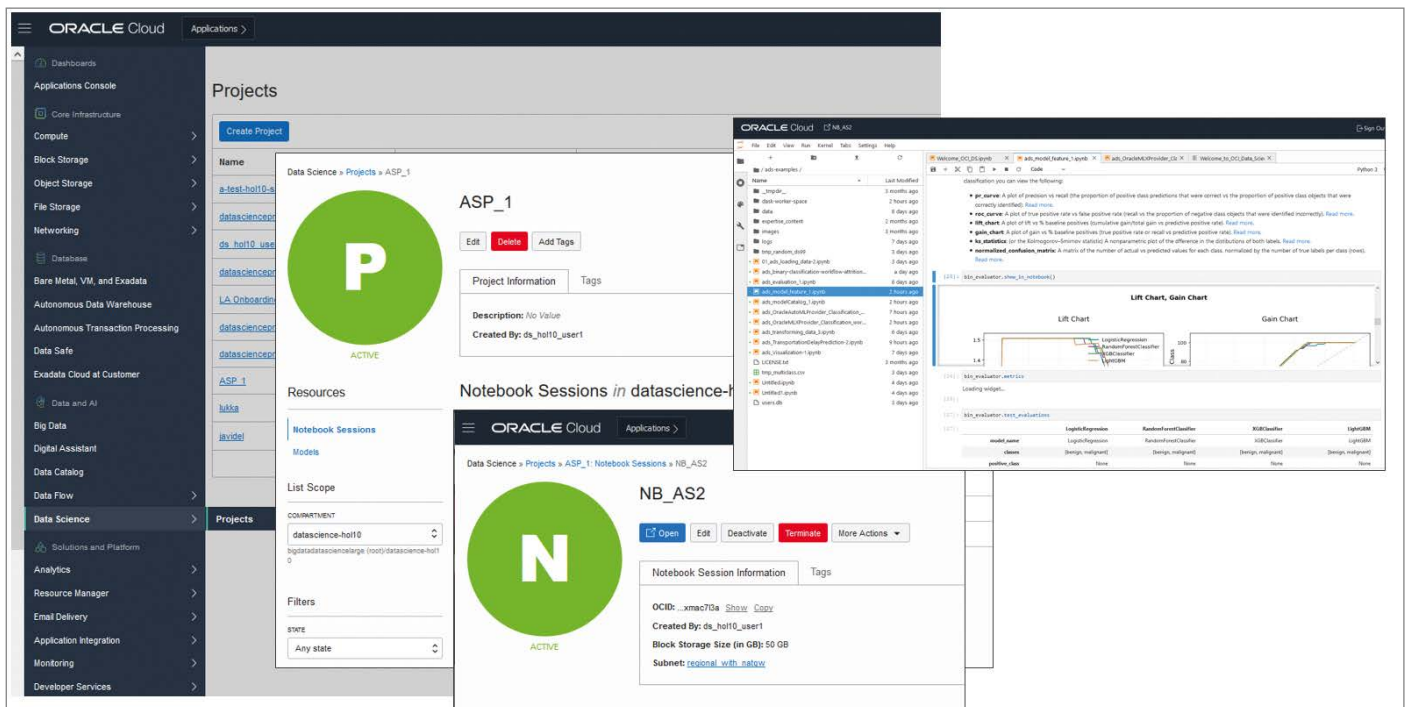


Abbildung 3: Data Science Screens © Alfred Schlaucher

AutoML

AutoML besteht aus vier Modulen und bringt diese im Hintergrund automatisch über einen internen Pipe-Mechanismus zur Ausführung. Der Pipe-Mechanismus nimmt die Ergebnisse von Modul 1 und schiebt sie zu Modul 2 usw.

Modul 1: Zur Lösung einer Klassifizierungsaufgabe zieht ein erstes Modul zum Beispiel die folgenden Algorithmen heran und erstellt für alle Algorithmen ein Modell:

- AdaBoostClassifier
- DecisionTreeClassifier
- ExtraTreesClassifier
- KNeighborsClassifier
- LGBMClassifier
- LinearSVC
- LogisticRegression
- RandomForestClassifier
- SVC
- XGBClassifier

Modul 2: Ein zweites Modul macht gegebenenfalls ein sogenanntes *Adaptiv Sampling*. Das ist gerade bei großen Datenbeständen wichtig, die man aufgrund ihrer Größe nicht komplett analysieren lassen will.

Modul 3: Das dritte Modul unternimmt das *Feature Selection*. Es rankt also die Attribute des Datensatzes nach ihrer Fähigkeit, zu dem Vorhersageergebnis beizutragen.

Modul 4: Im letzten Modul wird noch ein Hyperparameter Tuning durchgeführt. Es sucht die optimale Parametereinstellung der einzelnen Algorithmen.

Die Module werden zum Teil mehrfach und wechselseitig gestartet. AutoML ist wahlweise parallelisierbar. Man kann also die genutzte Hardware optimal einsetzen. Bei vielen Prozessoren lässt sich eine hohe Parallelisierung einstellen. Schritte wie das Erstellen von Trainings-/Testdatenbeständen erfolgen zum Teil implizit. Auch darum muss man sich nicht kümmern.

Manuelle Algorithmen-Anwendung / Deep Learning

Bei der Durchsicht der Algorithmen des AutoML wird sicher aufgefallen sein, dass Deep-Learning-Algorithmen fehlen. Diese sind natürlich optional wählbar, aber AutoML in Kombination mit Deep Learning kann ganz pragmatisch betrachtet in der Praxis zu Schwierigkeiten führen. Wählt man einen Deep-Learning-Ansatz zum Beispiel mit TensorFlow (Keras), so hat man es oft mit extrem hohen Datenmengen beziehungsweise mit speziellen Datenformaten wie etwa aus Pixeln bestehenden Bildern zu tun.

Die neuronalen Netze des Deep Learning wenden wiederholt Abermilliarden von Schwellwertoperationen in ihren Knoten an. Deep-Learning-Algorithmen laufen

heute am besten auf Grafikprozessoren. Der Data Science Cloud Service von Oracle bietet diese Variante auch an. Diese Vorgehensweise ist allerdings nicht für alle Aufgabenstellungen sinnvoll und je nach Anwendungsfall wird der Analyst zwischen den oben gezeigten klassischen Algorithmen und einem Deep-Learning-Ansatz entscheiden. Oder er wählt testweise beide, um die Ergebnisse zu vergleichen. Die Vorgehensweise lautet also eher, zunächst AutoML und dann Deep Learning – und nicht beides zur gleichen Zeit im Rahmen eines AutoML.

Damit haben wir einen Fall gefunden, in dem sich manuelles Anwenden eines Algorithmus empfiehlt. Der Oracle Data Science Cloud Service liefert für Deep-Learning-Verfahren fertig installierte *TensorFlow-Libraries* mit dem eingebetteten *Keras*. Je nachdem, welches *Compute-Shape* man in der Cloud gewählt hat, lassen sich die Berechnungen im Hintergrund auch auf GPUs ausführen und das ganz spontan ohne Vorbereitung der Infrastruktur.

Schritt 5: Modelle evaluieren / Prüfen der Güte von Modellen

Einer der letzten Schritte des Workflows ist die Prüfung, ob ein Modell auch gut vorher-sagen kann, und das Klären der Frage, welches von mehreren berechneten Modellen eine Aufgabenstellung am besten löst. Nehmen wir etwa die oben beschriebene

dritte Vorgehensweise, also die Kombination aus *AutoML* und zusätzlich manuell entwickeltem Modell, so will man möglichst bequem die Qualität aller gefundenen Modelle vergleichen.

Hier kommt das *ADSEvaluator-Modul* ins Spiel. Das ist ein einfacher Funktionsaufruf mit den Namen der zu prüfenden Modelle als List-Parameter (beliebig viele Modelle können dabei gleichzeitig geprüft werden). Als Ergebnis erhält man unter anderem:

- Einen Plot mit einer Precision-Recall-Kurve mit allen Modellen
- Eine ROC-Kurve mit allen Modellen
- Ein Lift Chart mit allen Modellen
- Ein Gain Chart Plot mit allen Modellen
- Für jedes Modell eine Confusion Matrix
- Eine vergleichende Kostenanalyse der jeweiligen Modelle
- Und vieles mehr

Schritt 6: Modelle erklären

Die Vorhersagequalität (Güte) eines Modells festzustellen bedeutet noch nicht, dass man auch versteht, warum ein Modell eine bestimmte Vorhersage macht. Die Erklärung eines Modells (Model Explanation) ermöglicht erst Rückschlüsse auf die Ursachen von Ereignissen, also die besonderen Verhältnisse in den Unternehmensprozessen, die zu den Analysedaten geführt haben. Hinzu kommt die fortschreitende Weiterentwicklung der zunehmend komplexeren Algorithmen. Die Komplexität der Modelle nimmt stetig zu und erschwert das Verstehen eines Modells zusätzlich und zunehmend:

- Welche Features tragen am meisten zum Entstehen eines Ereignisses bei?
- Wie weit ist ein Modell generalisierbar, das heißt, sind die Vorhersagemodelle auch noch passend, wenn neue, bislang unbekannte Input-Daten vorliegen, oder hat das Modell nur bekannte Zustände „auswendig gelernt“ (Overfitting-Problem)?

Den Prozess zum Erklären von Machine-Learning- und Deep-Learning-Modellen nennt man im Englischen *Machine Learning Explainability (MLX)*. Die Aufgabe des *MLX* wird in dem Data Science Cloud Service über den *ADSExplainer* gelöst. Mehrere Optionen stehen zur Auswahl:

- In einer Phase namens *Global Explanation* wird die Bedeutung der einzelnen Attri-

bute für das Vorhersageergebnis näher erklärt. Man untersucht, wieweit sich die Vorhersagefähigkeit eines Modells durch Weglassen und Hinzufügen von Attributen verändert.

- Das *Local Explanation* untersucht die Vorhersage von einzelnen Ereignissen aufgrund von Werteverteilungen in den Attributen und gibt damit punktgenaue Erklärungen für eine einzelne Vorhersage in einem konkreten Fall.

Eine Vielzahl von Einzelmethoden schließt sich hier an, die der *ADSExplainer* mit diversen weiteren Plots anbietet. In dieser Phase muss Domain- und Geschäftsprozesswissen die weitere Analyse begleiten, das heißt, Fachanwender sind wieder mit an Bord und müssen helfen, die Einzelergebnisse zu interpretieren.

Letztlich gilt: Je genauer ein Modell beziehungsweise die Vorhersagen, die durch ein Modell entstehen, erklärbar sind, desto mehr Vertrauen bringen Fachmitarbeiter und Entscheider den Analyseergebnissen und Handlungsempfehlungen entgegen. Wenn man versteht, warum man etwas tun soll, dann wird man es eher tun, als wenn man einen Handlungsvorschlag erhält, dessen Herkunft man nicht erklären kann.

Schritt 7: Verwalten von Modellen / Model Catalog

Der letzte Schritt bei der Herstellung von Machine-Learning- und Deep-Learning-Modellen ist deren Verwaltung (Management) nach Abschluss der Modellentwicklung. Nach Abschluss der Modellentwicklung überführt man Modelle zum Beispiel in operative Ablaufprozesse und nutzt sie dort zur Steuerung von betrieblichen Vorgängen. Den aktuellen Entwicklungsstand eines Modells, also der Stand, der sich aktuell in einem produktiven Einsatz befindet, speichert der Data Science Cloud Service im *Model Catalog*.

Spätestens an dieser Stelle sollte das interne ADS-Modellformat erwähnt werden. In diesem Format speichert der Data Science Cloud Service alle Modelle im *Model Catalog* ab. Dieses interne ADS-Format wurde bereits in den Schritten zuvor, also Modell-Evaluation und Modell-Explanation, verwendet. Verwaltungsoperationen sind unter anderem *save-, load-, publish-model*. Zur besseren Verwaltung ergänzt man gespeicherte Modelle mit weiteren Metadaten wie Beschreibungen, Verwendungszweck, Identifikationsnummer, Autor, Versionen usw.

Solche Metadaten können schließlich in einem Browser oder in der OCI-Oberfläche publiziert werden.

Schritt 8: Model Deployment

Der finale Schritt, das Deployment in die operativen Abläufe eines Unternehmens oder einfach nur in die Ablauflogik einer Anwendung, erfolgt über *Oracle Functions*.

Dieser Artikel ist gekürzt. Eine Langform, in der das Model Deployment mit Oracle Functions erläutert wird, kann direkt über den Autor bezogen werden.



Alfred Schlaucher
alfred.schlaucher@oracle.com

Alfred Schlaucher ist seit 1986 in der IT tätig. Er startete mit der Programmierung von Datenbanken auf Großrechnern von Siemens und IBM. Zu Beginn der 1990er Jahre verlagerte sich der Schwerpunkt seiner Arbeit auf Datenmodellierung und Metadatenmanagement. Seit Ende der 1990er Jahre berät er Oracle-Kunden bezüglich ihrer Data Warehouse- und Analyse-Infrastruktur. In den letzten Jahren beschäftigt sich der Autor verstärkt mit Big Data, Advanced Analytics und Machine Learning sowie den Analysesprachen R und Python.

Wir begrüßen unsere neuen Mitglieder

Persönliche Mitglieder

- Deniz Engin
- Benjamin Knauer
- Andreas Gmyrek
- Julian Driver
- Kinga Bori
- Markus Hampe
- Lisa Nell
- Steven Hertel

Firmenmitglieder DOAG

- Alta Via Consulting GmbH,
Martin Kalkuhl
- menten GmbH,
Ralph Menten
- Qaware GmbH,
Josef Fuchshuber



Termine

Mai

05

05. - 06.05.2020

APEX Connect 2020 Online

apex.doag.org

(Kostenfrei)

Juni

06

04.06.2020

DOAG-VORSTANDSSITZUNG 2/20

17.06.2020

Regionaltreffen München/Südbayern

Andreas Ströbel

17.06.2020

Regionaltreffen Berlin/Brandenburg

Michel Keemers

18.06.2020

Regionaltreffen Dresden/Sachsen

Helmut Marten

18.06.2020

Regionaltreffen Nürnberg

Martin Klier, Thomas Köppel

22. & 23.06.2020

DOAG Construction & Engineering Day

München

Juli

07

10.07.2020

DOAG Datenbank Webinar

Online

15.07.2020

Regionaltreffen München/Südbayern

Andreas Ströbel

16.07.2020

Regionaltreffen Nürnberg

Martin Klier, Thomas Köppel

Impressum

Red Stack Magazin inkl. Business News 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 inkl. Business News 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 inkl. Business News 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: Stefan Kinnen. 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.)
ViSdP: Mylène Diacquenod
Redaktionsleitung Red Stack Magazin:
Martin Meyer
Redaktionsleitung Business News:
Marcos López
Kontakt: redaktion@doag.org
Weitere Redakteure (in alphabetischer Reihenfolge): Tobias Arnhold, Wolf G. Beckmann, Sven Bernhardt, Martin Bracher, Nadine Brehm, Björn Bröhl, Julia Chevalier, Niels de Bruijn, Florian Graßhoff, Eugen L. Gross, Claus Huempel, Markus Klenke, Martin Klier, Wolfgang Klinger, Aleksander Koleski, Martin Meyer, Holger Nicolay, Thomas Petrik, Mariel Prange, Philipp Salvisberg, Rolf Scheuch, Alfred Schlaucher, Michael Schulze, Michael Skowasch, Elvira Steppacher, Christian Schwitalla, Günther Stürner

Titel, Gestaltung und Satz:

Caroline Sengpiel
DOAG Dienstleistungen GmbH
(Anschrift s.o.)

Fotonachweis:

Titel: Bild © Alexander | <https://stock.adobe.com>
S. 7: © scyther5 | <https://de.123rf.com>
S. 11: © rawpixel | <https://de.123rf.com>
S. 18: © Thittaya Janyamethakul | <https://de.123rf.com>
S. 24: © Tithi Luadthong | <https://de.123rf.com>
S. 31: © Валентин Амосенков | <https://de.123rf.com>
S. 37: © Sergey Novikov | <https://de.123rf.com>
S. 42: © Sergey Nivens | <https://de.123rf.com>
S. 48: © golubovy | <https://de.123rf.com>
S. 55: © Somchai Rakin | <https://de.123rf.com>
S. 62: © RAJESH RAJENDRAN NAIR | <https://de.123rf.com>
S. 68: © Nukoonrat | <https://de.123rf.com>

S. 70: © Vitaliy Mateha | <https://de.123rf.com>
S. 73: © Dmitriy Shpilko | <https://de.123rf.com>
S. 74: © Mopic | <https://stock.adobe.com>
S. 77: © Eric Eric | <https://de.123rf.com>
S. 78: © rfsrn | <https://de.123rf.com>
S. 80: © melpomen | <https://de.123rf.com>
S. 81: © fisher4 | <https://de.123rf.com>
S. 83: © Sean Prior | <https://de.123rf.com>
S. 88: © Jozsef Bagota | <https://de.123rf.com>
S. 93: © gmat3r | <https://de.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:

WIRMachenDRUCK GmbH,
www.wir-machen-druck.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

| | | | | | |
|---|-----------------|---|-------|---|-------|
| B4Bmedia.net AG https://e-3.de | S. 41 | MuniQsoft Consulting GmbH www.muniqsoft-consulting.de | S. 67 | Robotron Datenbank-Software GmbH www.robotron.de | S. 21 |
| dbi services sa www.dbi-services.com | S. 27 | MuniQsoft Training GmbH www.muniqsoft-training.de | S. 3 | Trivadis AG www.trivadis.com | U 4 |
| DOAG e.V. www.doag.org | U 2, U 3, S. 54 | PROMATIS software GmbH www.promatis.de | S. 79 | | |

Online

APEX *connect*
by DOAG

Am **5. und 6. Mai 2020** findet die
APEX connect 2020 als **kostenfreie**
Online-Veranstaltung mit abgespecktem
Programm statt. Sei unbedingt dabei!

Mehr Informationen und Anmeldung unter:

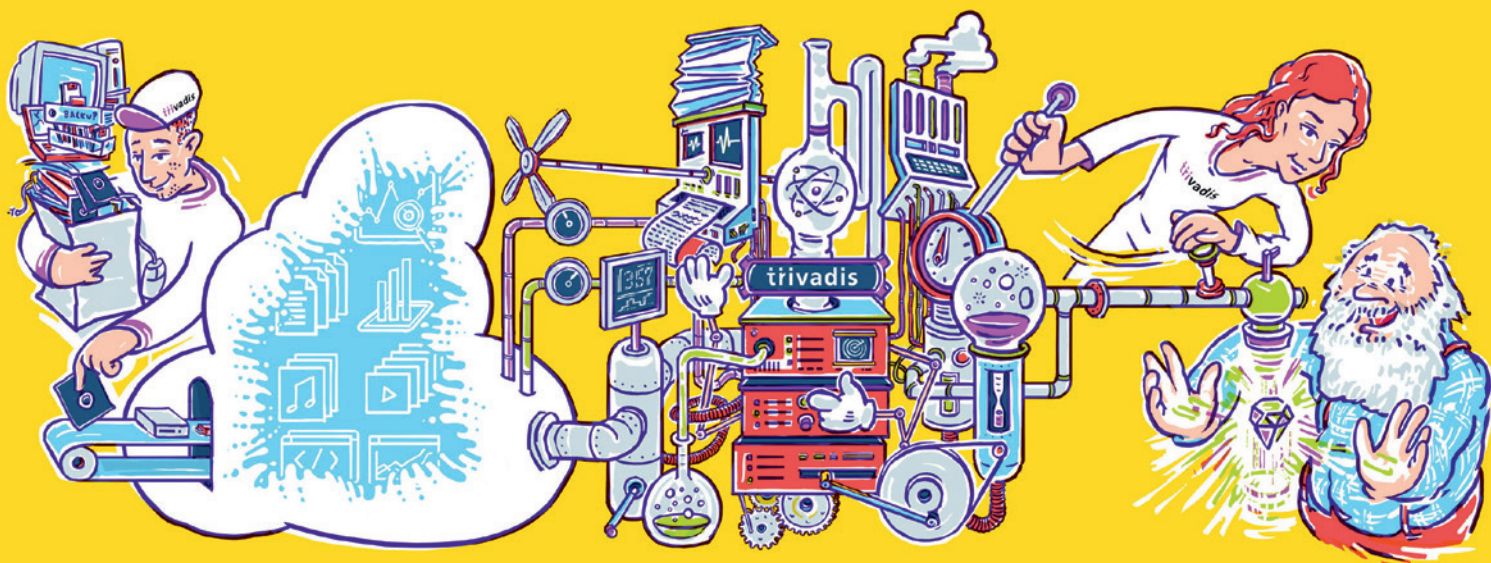
apex.doag.org



DOAG

Einfach clever.

Deine
Cloud-Lösungen
von Trivadis.



Von Rohdaten zu Diamanten – mit Trivadis und Microsoft Azure

Mit unseren Cloud-Lösungen holst du das meiste aus deinen Daten. Doch wie schaffst du den Weg in die Cloud? Wie stellst du sicher, dass du die Übersicht behältst und die Kosten im Griff hast? Und wie bindest du auch Technologien wie IoT oder KI sinnvoll ein?

Mit unserem „Way to Azure“ gelingt dir der Weg in die Microsoft Cloud ganz selbstverständlich. Und mit unserer „Azure Foundation“ kannst du die Microsoft Cloud als sichere und produktive Plattform unkompliziert in deiner Unternehmung einführen. **Einfach clever!**

