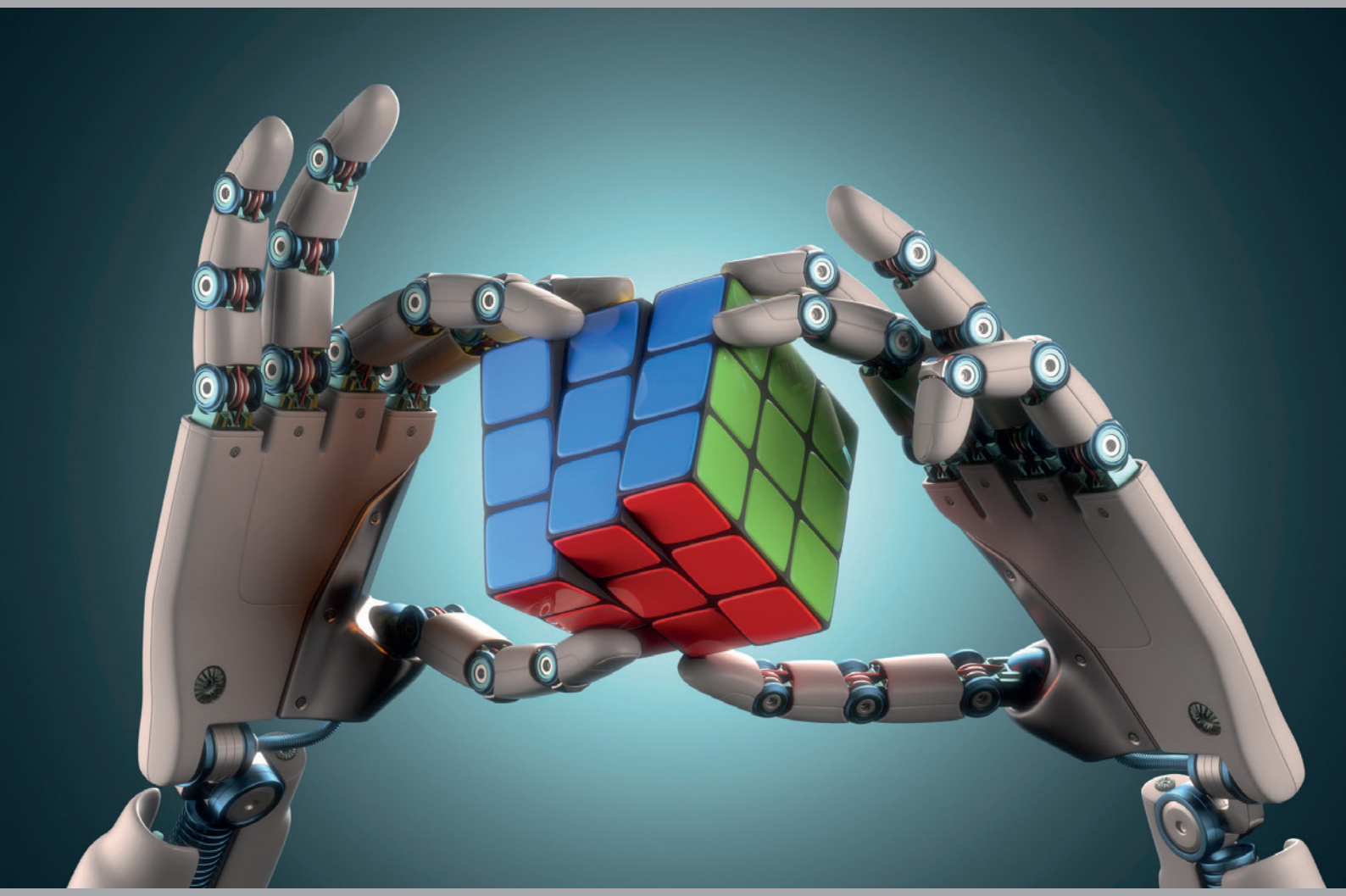
Business News

DOAG Zeitschrift für die Anwender von Oracle Business- und BI-Lösungen

BS



Künstliche Intelligenz

Frühling 2018 | ISSN 09 36-0360 | http://bs.doag.org

Save the Date

20. - 23. Nov 2018

in Nürnberg

2018.doag.org

Eventpartner: AOUG Austrian Oracle User Group · SOUG swiss oracle user group · iJUG Verbund · ORACLE

Rolf Scheuch
DOAG-Vorstand und Leiter
der Data Analytics Community

Liebe Leserinnen und Leser,
diese Ausgabe ist ein Hit! Wir beschäftigen uns fast ausschließlich mit dem Thema „Künstliche Intelligenz". Laut einer Studie des kalifornischen Unternehmens Evans Data setzen bereits 6,5 Millionen (!) Entwickler in ihren Projekten Technologien für künstliche Intelligenz und Machine Learning ein. So konnten wir für diese Ausgabe auf das Schwarm-Wissen der Entwickler der DOAG-Anwendergemeinschaft zurückgreifen, um Ihnen das gesamte Spektrum von künstlicher Intelligenz und Machine Learning in dieser Ausgabe näherzubringen.

Die aktuelle DOAG Business News beleuchtet dieses Hype-Thema aus unterschiedlichen Winkeln, also „Food for thought", und bietet Ihnen durch die unterschiedlichen Sichtweisen und Anwendungsbiete eine breite Palette von Ansätzen für Ihr Unternehmen.

Viel Spaß beim Lesen.

Ihr

Impressum

DOAG Business News wird von der DOAG Deutsche ORACLE-Anwendergruppe e.V. (Tempelhofer Weg 64, 12347 Berlin, www.doag.org), herausgegeben. Es ist das User-Magazin rund um die Applikations-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.

DOAG 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.)
Chefredakteur (ViSdP): Wolfgang Taschner
Kontakt: redaktion@doag.org
Weitere Redakteure: Lisa Damerow, Mylène Diacquenod, Marina Fischer, Fried Saacke, Rolf Scheuch, Dr. Frank Schönthaler

Druck:
adame Advertising and Media GmbH, Berlin, www.adame.de

Fotonachweis:
Titel: © ktsdesign/123RF
S. 5: © Michal Bednarek/123RF
S. 10: © Andriy Popov/123RF
S. 14: © Allan Swart/123RF
S. 17: © Kittipong Jirasukhanont/123RF
S. 22: © phonlamaiphoto/Fotolia
S. 27: © Kheng Ho Toh/123RF
S. 31: © gstockstudio/123RF
S. 35: © lightwise/123RF

Titel, Gestaltung und Satz:
Caroline Sengpiel,
DOAG Dienstleistungen GmbH
(Anschrift s.o.)

Anzeigen:
Simone Fischer, DOAG Dienstleistungen GmbH
(verantwortlich, Anschrift s.o.)
Kontakt: anzeigen@doag.org

Mediadaten und Preise unter: www.doag.org/go/mediadaten

AI hat das Potenzial, den Markt in den meisten Branchen grundlegend zu ändern

Die Wirkung der KI in eine von uns gewünschte Richtung zu lenken

Data Lakes oder Schema-on-read stehen für neue Ansätze, die Flexibilität, Skalierbarkeit und Performance versprechen

Unsere Inserenten

Künstliche Intelligenz – das Mögliche und das Unmögliche

Dimitri Gross, OPITZ CONSULTING Deutschland GmbH

Der Begriff „künstliche Intelligenz" kommt uns heute sehr oft zu Ohren; etwa in Brettspielen, bei denen technische Systeme bereits seit den 1980er-Jahren einen guten Gegenspieler nachahmen konnten. Diese wurde seitdem so perfektioniert, dass ein Computer im Jahr 1996 das erste Mal einen Schachweltmeister, Gari Kasparow, im Schachduell besiegte.

Die Entwicklung geht weiter: Im März 2016 wurde ein weiterer Meilenstein für die künstliche Intelligenz im Spielesektor gesetzt. Google AlphaGo besiegte den weltbesten Go-Spieler. Das stellte den bis dato etablierten Glauben auf den Kopf, das Spiel Go könne aufgrund seiner Kombinationsmöglichkeiten und der sich daraus ergebenden Komplexität von einem Algorithmus nicht in Perfektion gespielt werden.

Was hören wir noch aus der Welt der künstlichen Intelligenz? Immer wieder tauchen Schlagzeilen auf, die künstliche Intelligenz in ein negatives Licht stellen. So sorgte eine Meldung für Entrüstung, in der es darum ging, wie ein japanischer Versicherer seine Prozesse mithilfe von künstlicher Intelligenz optimierte und damit angeblich seine Mitarbeiter ersetzte. Dabei wurde nicht geklärt, ob der Versicherer diesen Mitarbeitern vielleicht nur eine andere Aufgabe gegeben hat und nur routinelastige Tätigkeiten einer Maschine überließ [1].

Die Frage, ob der Einsatz einer Maschine der Allgemeinheit oder nur dem Nutzen eines Einzelnen dient und welche Rahmenbedingungen für einen sozialförderlichen Einsatz notwendig sind, müssen letztendlich Gesellschaft und Politik beantworten. Nichtsdestotrotz sind ethische Grundfragen für die Marktakzeptanz einer Technologie von Bedeutung. Daher interessieren uns Anwendungsfälle, die zeigen, was künstliche Intelligenz im unternehmerischen Kontext zu leisten imstande ist und welchen Mehrwert sie damit schafft.

Dieser Artikel zeigt die aktuellsten Technologien im Bereich der künstlichen Intelligenz und teilt diese in Domänen auf. Es gibt

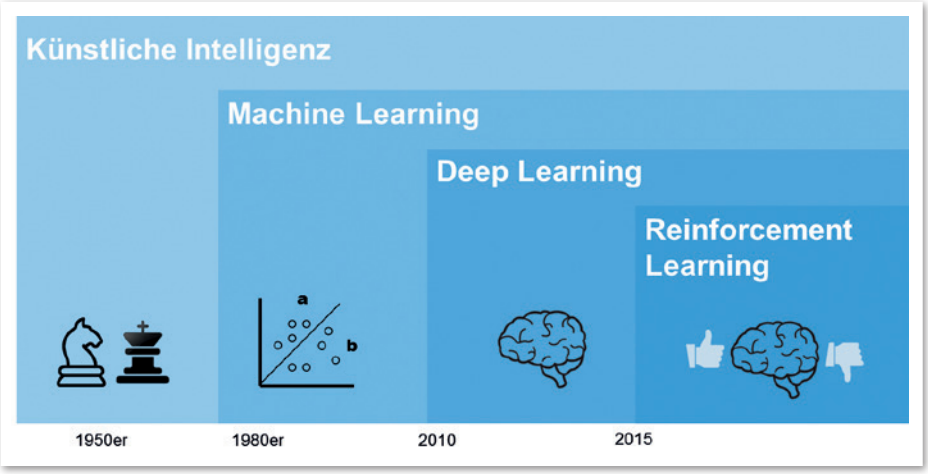


Abbildung 1: Domänen der künstlichen Intelligenz (erweiterte Eigendarstellung nach [2])

Beispiele, bei denen der Mehrwert für unseren Alltag klar sichtbar ist. Anschließend gehen wir der Frage nach, ob eine Singularität im Bereich der künstlichen Intelligenz, und damit der vollständige Ersatz eines Menschen durch eine Maschine, in naher Zukunft denkbar sein wird.

Grundlegende Begrifflichkeiten

Künstliche Intelligenz nimmt ihre Wurzeln bereits in den 1950er-Jahren, als das erste Schach-Computerprogramm entwickelt wurde. *Abbildung 1* zeigt die permanente Entwicklung der künstlichen Intelligenz ab dem Jahr 1980. Ungefähr zu der Zeit, als die Hardware leistungsfähiger wurde, begannen Experimente mit Klassifizierungsalgorithmen. Diese und viele weitere Algorithmen aus der nahen Vergangenheit werden heutzutage unter dem Begriff „Machine Learning" zusammengefasst. Hier wird zwischen „Supervized" und „Unsupervized Learning" unterschieden. Bei Ersterem stehen dem Algorithmus bekannte Resultate unterstützend zur Verfügung, während die Algorithmen beim Unsupervized Learning die Klassifizierungen im aktuell vorhandenen Datenset jedes Mal aufs Neue durchführen, ohne die Resultate aus historischen Daten zu kennen. Beide Arten sind in hochautomatisierten Prozessen weitverbreitet, die ein Mensch rein reaktionstechnisch nicht mehr ausführen kann – Denkvermögen oder Intelligenz spielen dabei keine Rolle, denn solche Algorithmen erlauben nur, eine bestimmte Aufgabe automatisiert und in hoher Geschwindigkeit abzuarbeiten.

Deep Learning

Dem Supervized Machine Learning wird seit etwa zehn Jahren der neu definierte Bereich des „Deep Learning" zugeordnet. Es basiert auf künstlichen neuronalen Netzen, die man auch als gefaltete neuronale Netzwerke (Convolutional Neural Networks, CNN) bezeichnet.

Wie *Abbildung 2* anhand eines neuronalen Netzwerkes zur Bilderkennung zeigt, besteht ein CNN aus mehreren Stufen, die miteinander verknüpft sind. Jeder Block in diesem Netzwerk bildet sozusagen ein komplexes System aus Filtern. Ein großes neuronales Netzwerk besteht aus mehreren solcher Blöcke, wobei jeder Block ein eigenes neuronales Netzwerk darstellt. Aufgrund dieser Quasi-Rekursion spricht man von einem gefalteten Netzwerk.

Mit dieser Herangehensweise lassen sich dynamische Verfahren realisieren, die im Gegensatz zum Machine Learning, in dem die Algorithmen immer nur eine Problemstellung gut lösen können, auf mehrere Problemstellungen gleichermaßen gut angewendet werden können. Bei einer Bildanalyse könnte der Algorithmus zum Beispiel nicht nur klare geometrische Formen vor einem eintönigen Hintergrund erkennen, sondern auch beliebig komplexe Objekte vor beliebig komplexen Hintergründen oder Kombinationen von beidem. Anwendungsszenarien gibt es viele: von der perfekt klingenden Sprachsynthese oder einer sehr robusten Spracherkennung bis zur dynamischen Objekterkennung und deren Derivaten.

Reinforcement Learning

Eine ganz neue Entwicklung bei der künstlichen Intelligenz ist das „Reinforcement Learning". Diese Technik ist dem Bereich „Deep Learning" zugeordnet. Allerdings wird dem System (hier „Agent" genannt) in diesem Fall eine gänzlich unbekannte Umgebung präsentiert, wie zum Beispiel die Bildschirmausgabe eines Computerspiels. Der Agent analysiert die Pixel und versucht, wie dies auch ein Mensch tun würde, auf eine bestimmte

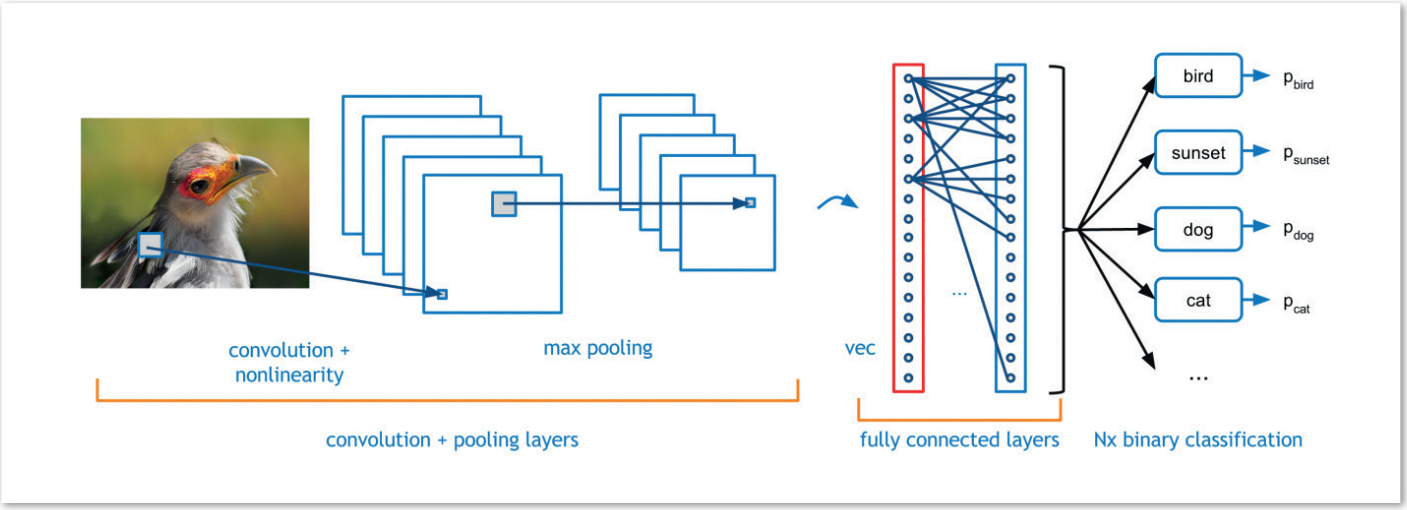


Abbildung 2: Schematische Abbildung eines gefalteten neuronalen Netzwerks (CNN) nach [3]

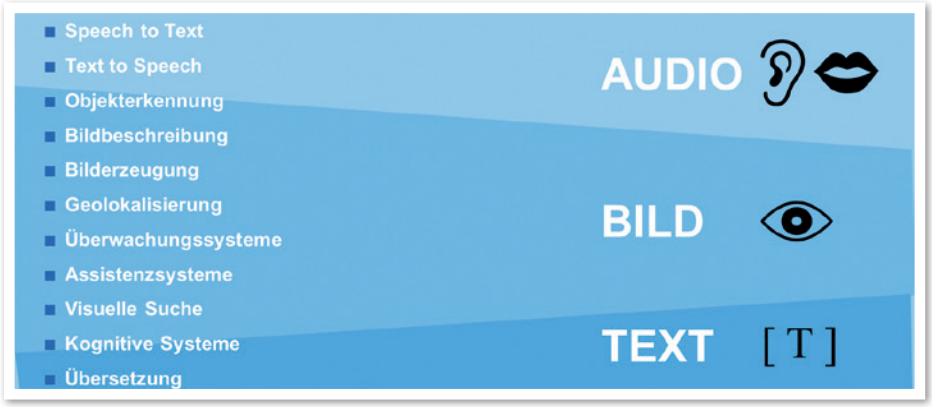


Abbildung 3: Domänen der künstlichen Intelligenz

Art und Weise Zusammenhänge zwischen diesen zu finden. Dabei erhält er in regelmäßigen Abständen positives oder negatives Feedback auf seine Aktionen. Auf dieser Grundlage berechnet der Algorithmus im Laufe der Trainingszeit eine Kostenfunktion, die es ihm ermöglicht, seine Aktionen so an die Umgebung anzupassen, dass diese möglichst nah an das positive Feedback herankommen.

Domänen der künstlichen Intelligenz

Die bekannten Anwendungsfälle von künstlicher Intelligenz lassen sich grob in drei Domänen aufteilen: Audio, Bild und Text *(siehe Abbildung 3)*. Im Audio-Bereich dominieren Speech-to-Text- und Text-to-Speech-Systeme. Die Verwendung von CNN in diesem Bereich verbesserte die Konvertierung von Sprache in Text erheblich. Das zeigt auch die Verbreitung von Sprachassistenten wie Siri, Google Now etc. In diesen Mensch-Maschine-Interfaces wurde die Sprachausgabe erheblich weiterentwickelt.

Die Sample-basierten Systeme werden durch die Verwendung neuer Ansätze aus dem Bereich des Deep Learning obsolet. Dies zeigt auch das Funktionsprinzip WaveNet von Deepmind *(siehe Abbildung 4)*. Hier ist zu sehen, wie mithilfe eines neuronalen Netzwerks eine Wave-Form entsteht. Der Vorteil liegt auf der Hand: Menschlich klingende Aussprache, Nachbildung verschiedener Dialekte und Akzente oder einer beliebigen Stimme sind möglich.

Als Anwender genießen wir seit längerer Zeit die technologischen Errungenschaften aus dem Bereich der Bild-Analyse. Hier dominiert die Objekt-Erkennung, die mithilfe von Deep Learning in kurzer Zeit auf eine neue Ebene gehoben werden konnte. Als Produkt dieser Forschung gibt es zum Beispiel intelligente Assistenzsysteme im Auto. Die durch neue Verfahren modernisierte Gesichtserkennung ermöglicht effiziente Anwendungsszenarien im Sicherheitsbereich. Im Bereich der Gesichtserkennung in Echtzeit sowie bei der Suche nach gleichen Gesichtern aus Millionen von Gesichtern zeigt beispielsweise die Forschung des Unternehmens Findface besonders vielversprechende Ergebnisse [5]. Besonders interessant sind in dieser Domäne zwei Technologien: Text to Image und Geolokalisierung anhand von Bildern (PlaNet).

Text to Image kehrt quasi die Objekterkennung und die textuelle Beschreibung der Objekte um. Nicht das Bild dient als Input, sondern Text. Dementsprechend wird im Output kein Text generiert, der das Bild beschreibt, sondern es wird ein Bild erzeugt, das der angegebenen Beschreibung am nächsten kommt. Dieses System zeigt deutlich das Potenzial eines CNN. Aber auch ganz allgemein eröffnet dieses Verfahren einen neuen Anwendungsbereich beispielsweise in der Kriminalistik. Der Erstellungsprozess eines Phantombilds kann deutlich beschleunigt werden, wenn ein Augenzeuge sofort passende Bildvorschläge zu seiner Personenbeschreibung bekommt und bewerten kann, welches Bild am besten zu der Beschreibung passt. Hier entfällt das bislang übliche Aussuchen in der Bibliothek passender Gesichtsmerkmale *(siehe Abbildung 5)*.

Ähnlich revolutionär ist der Versuch, mithilfe von Deep-Learning-Methoden anhand von Bildern Geo-Koordinaten zu schätzen. Hier nutzen die Wissenschaftler ebenfalls ein CNN, das in der Trainingsphase mit knapp 90 Millionen Bildern und passenden Geo-Koordinaten beladen wurde. Das Netzwerk und seine Filter wurden jedoch so angepasst, dass auch Landschaftsmerkmale, Flora und Wetterphänomene, die für bestimmte Gebiete gewöhnlich sind, bei der Modellbil-

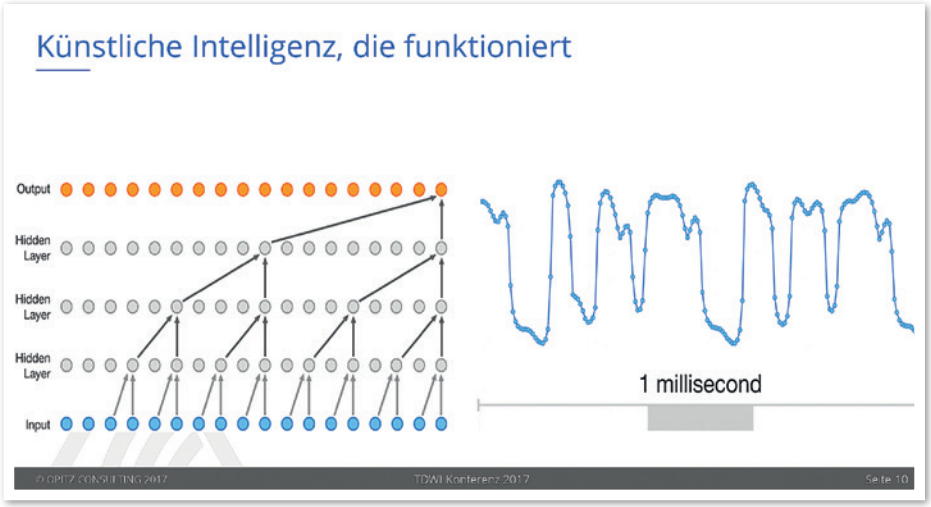


Abbildung 4: WaveNet-Funkionsprinzip [4]

dung berücksichtigt wurden. Auch architektonische Merkmale wurden berücksichtigt; somit ist das System in der Lage, die Geo-Koordinaten auf Straßenebene zu ermitteln, aber auch auf der Landes- und Kontinent-Ebene. Die Verlaufskurven in *Abbildung 6* vergleichen die Treffergüte beim Schätzen von Geo-Koordinaten von PlaNet mit der eines Menschen. Hervorzuheben ist noch die Modellgröße, die mit unter 1 GB sehr gering ausfällt. Damit ist es möglich, dieses oder ähnliche Verfahren auch auf mobilen Endgeräten einzusetzen, was das Einsatzgebiet noch etwas erweitert.

Im Bereich der Text-Analyse gibt es momentan zwei herausragende Bereiche. Zu einen die maschinelle Übersetzung von Texten von jeder beliebigen Sprache in eine andere, zum anderen kognitive Systeme, die mehrere Verfahren aus den Bereichen Machine Learning und Deep Learning in sich vereinen und somit zur Prozessautomatisierung im Aktenwesen beitragen können.

Neu und interessant ist der Einsatz von neuronalen Netzwerken im Bereich der Textdaten. Hier hat Google vor Kurzem mit seiner „Multilingual Neural Machine Translation" für heiße Diskussionen gesorgt [8]. Dabei entfällt die klassische „1:1"-Verknüpfung in den Wörterbuch-Datenbanken und auch deren aufwendige Pflege.

Bereiche mit Ausbaupotenzial

Den Bereichen mit Ausbaupotenzial sind einige Technologien zuzuordnen, die aktuell zwar erst rudimentär funktionieren, jedoch aufgrund ihrer Weiterentwicklungsmöglichkeiten in den nächsten Jahren von immer größerer Bedeutung werden können. Eine davon ist der Chatbot. Alexa und andere Chatbots nutzen als Mensch-Maschine-Interface die natürliche Sprache und profitieren damit besonders von den Errungenschaften im Audio-Bereich. Sprachverständnis und Sprachausgabe scheinen hier schon auf einem guten Niveau zu sein, sodass man den Eindruck bekommen könnte, ein Chatbot wisse bereits auf jede Frage eine Antwort. Leider wird der Nutzer hier spätestens dann enttäuscht, wenn er vom erwarteten Frage-Schema abweicht. Dann versteht der Chatbot seine Fragen nicht mehr oder antwortet mit einer belanglosen Floskel.

Der Grund für diese technologische Diskrepanz mit hoher Sprachqualität auf der einen Seite und der etwas hinterherhinkenden Logik auf der anderen Seite ist das etwas in die Jahre gekommene Frage-Antwort-Konzept [9]. Auf X mögliche Fragen werden Y mögliche Antworten vorgegeben. Weicht man davon ab oder verlässt man den Kontext, ist der Chatbot hilflos. In dieser Technologie steckt ein großes Potenzial. Es fließen bereits viele Forschungsgelder in diese Richtung, sodass wir sicher schon bald mit etwas flexibleren Chatbots rechnen können, auch wenn sie noch lange nicht jede erdenkliche Frage richtig beantworten können werden.

Ein weiterer Anwendungsfall, der vielversprechend klingt, kommt aus der Machine-Learning-Ecke und nennt sich „Predictive Policing". Mit diesem Verfahren wurden bemerkenswerte Resultate bei der Vorhersage von Kriminalität in Großstädten erreicht. Doch sobald dieses Verfahren von problematischen Vierteln auf ganze Städte ausgedehnt wurde, funktionierte die Vorhersage nicht mehr. Es stellte sich heraus, dass die Daten, die als Grundlage für die statistischen Analysen dienten, aus sozial schwachen Gegenden stammten. Die darauf errechneten Modelle konnten daher nicht in allen Stadtteilen die gleiche gute Performance erreichen.

Auch in diesem Bereich wird derzeit viel geforscht. Aktuell zeichnen sich bereits Synergien zwischen Deep Learning mit Bilderkennung und kognitiven Systemen ab, die aus gewonnenen Objektdaten Handlungsmuster extrahieren und somit eine Datengrundlage schaffen, die für den zu überwachenden Ort repräsentativ ist. Diese sogenannten „Action-Prediction-Algorith-

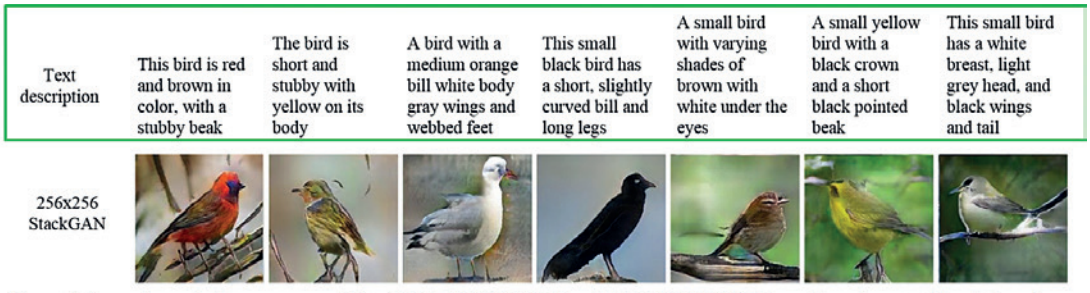


Figure 3. Example results by our proposed StackGAN, GAWWN [20], and GAN-INT-CLS [22] conditioned on text descriptions from CUB test set. GAWWN and GAN-INT-CLS generate 16 images for each text description, respectively. We select the best one for each of them to compare with our StackGAN.

Abbildung 5: Text to Image führt zu generierten Bildern [6]

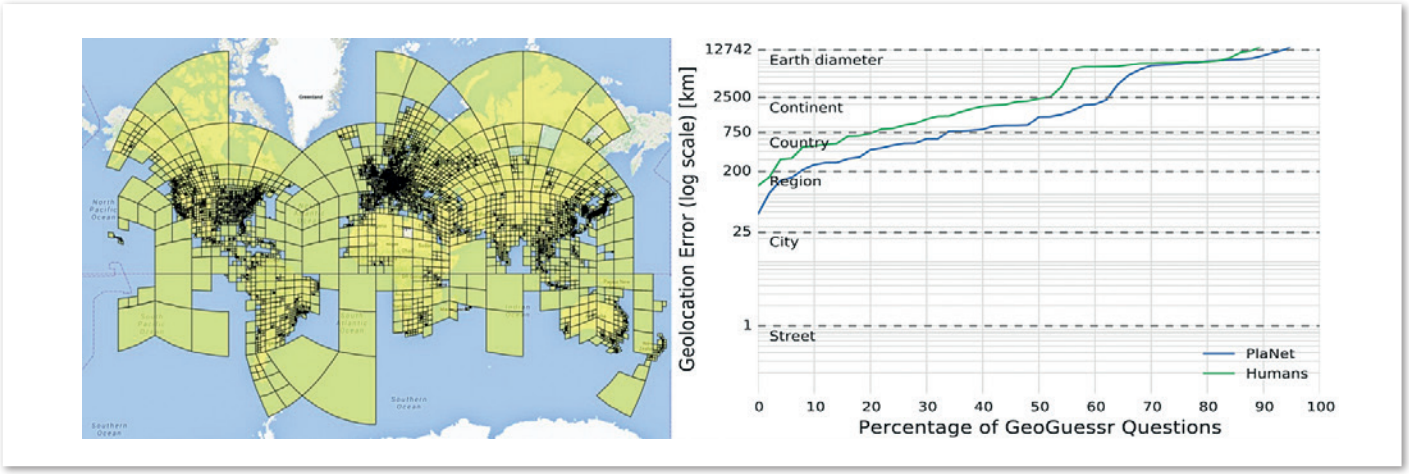


Abbildung 6: Darstellung der Modellgüte im Vergleich zum Menschen [7]

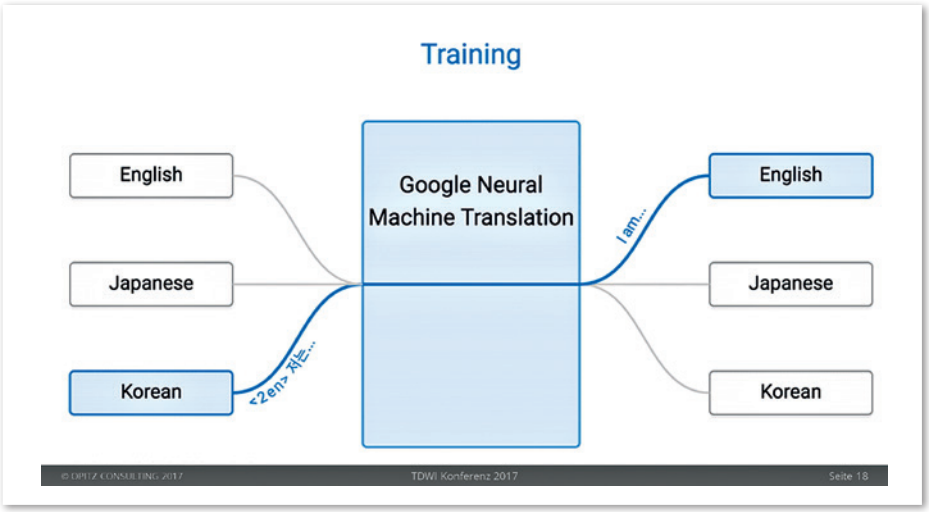


Abbildung 7: Google Neural Machine Translation [8]

men" nutzen Deep-Learning-Techniken, um anhand weniger Frames einer Bildsequenz beispielsweise einer Überwachungskamera die möglichen Handlungen von Personen zu berechnen, die sich im Bildausschnitt befinden [10].

Singularität oder das Unmögliche

Nehmen wir die besprochenen Anwendungsfälle und versuchen, diese analog zur menschlichen Sinnesverarbeitung in einem geschlossenen System zusammenzuführen, werden wir schnell feststellen, dass dies nicht möglich ist. Ein neuronales Netzwerk, das die gesprochene Sprache versteht, gleichzeitig die gezeigten Bilder interpretiert und je nach Kontext diese Interpretation mit der aktuell gesprochenen Sprache verknüpft, parallel dazu noch Handlungen passend zu der errechneten Situation ableitet und so weiter, gibt es nicht. Mathematisch gesehen würde es sich um ein kombinatorisches Problem handeln. Wie wir sehen konnten, bestehen selbst im Bereich der Bild-Analytik, in der bei Objekterkennung oder Objekterzeugung Deep-Learning-Ansätze verwendet werden, technologisch nennenswerte Unterschiede zwischen Maschine und Mensch.

Der Versuch, alle aktuell mit künstlicher Intelligenz abgebildeten Use Cases in ein Modell oder ein Gesamtkonstrukt zu quetschen, muss scheitern. Es sei denn, es würde ein komplett neues Verfahren entwickelt, das in der Lage ist, viele Facetten in einem Modell abzubilden. Aber das wird uns nicht so schnell einholen, wie uns Science-Fiction-Filme prophezeien.

Fazit

Wie wir sehen, lassen sich einige Prozesse durch künstliche Intelligenz beschleunigen und gleichzeitig in ihrer Effizienz steigern. Zu behaupten, dass künstliche Intelligenz uns Menschen intellektuell überlegen ist, wäre nicht ganz richtig. Bei allen genannten Beispielen geht es um die Steigerung der Prozessgeschwindigkeit. Angst vor Singularität müssen wir also nicht haben. Diese kann es in absehbarer Zeit nicht geben. Vor uns stehen derweil noch viele interessante Baustellen, wie zum Beispiel Chatbots, die in den nächsten Jahren weitere Fortschritte machen und unseren Alltag erleichtern werden.

Weiterführende Literatur

[1] *http://www.spiegel.de/wirtschaft/unternehmen/fukoku-mutual-life-versicherer-ersetzt-mitarbeiter-durch-ibms-ki-watson-a-1128670.html*
[2] *https://blogs.nvidia.com/wp-content/uploads/2016/07/Deep_Learning_Icons_R5_PNG.jpg.png*
[3] *http://www.kdnuggets.com/2016/09/beginners-guide-understanding-convolutional-neural-networks-part-1.html*
[4] WaveNet, A Generative Model for Raw Audio, Deepmind: *https://arxiv.org/pdf/1609.03499.pdf*
[5] *https://findface.pro/en/#technology*
[6] Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks, Han Zhang, Tao Xu, Hongsheng Li: *https://arxiv.org/pdf/1612.03242v1.pdf*
[7] PlaNet - Photo Geolocation with Convolutional Neural Networks, Weyand, Kostrikov, Philbin: *https://arxiv.org/pdf/1602.05314.pdf*
[8] Zero-Shot Translation with Google's Multilingual Neural Machine Translation System: *https://research.googleblog.com/2016/11/zero-shot-translation-with-googles.html*
[9] ELIZA - A Computer Program For the Study of Natural Language Communication Between Man And Machine, in Communications of the ACM, 1. Auflage, Juni 1966
[10] *http://news.mit.edu/2016/teaching-machines-to-predict-the-future-0621*

Dimitri Gross
dimitri.gross@opitz-consulting.com

Von Mobile First zu AI First

Andreas Dohren und Tobias Huber, LumaBIT GmbH

In der IT-Branche war „Mobile First“ in den letzten zehn Jahren immer wieder ein großes Schlagwort. Wenn es nach den Großen in der IT-Branche geht, neigt sich dieses Kapitel dem Ende zu und mit „AI First" wurde vor einiger Zeit auch schon das neue Ziel ausgegeben. Es hat den Anschein, als ob in der IT-Branche wieder Goldgräberstimmung herrsche.

War Mobile First erfolgreich? Diese Frage lässt sich mit einem ganz klaren „Ja“ beantworten. Für iOS und Android gibt es mittlerweile mehr als drei Millionen Apps. Weltweit zählen wir an die drei Milliarden Smartphone-Benutzer, die täglich ihr Gerät dafür verwenden, Dinge zu tun, für die sie vor zehn Jahren noch einen Computer benötigt hätten oder die nicht einmal möglich gewesen wären. Die ganze Welt scheint vernetzt und ohne Smartphone scheint man nicht mehr Teil dieser Welt zu sein.

Bei den drei Millionen Apps hat man aber auch das Gefühl, dass es von jeder App mindestens tausend verschiedene Varianten gibt. Seit Jahren ist keine neue App mehr erschienen, bei der man das Gefühl hatte, dass hier etwas Grandioses entwickelt wurde. Eine Kopie von etwas Vorhandenem machen, eine kleine Anpassung durchführen und die App unter einem neuen Namen auf den Markt werfen. Das scheinen die aktuellen Trends der IT-Firmen und Start-ups zu sein, die im mobilen Bereich tätig sind.

Es hat den Anschein, als ob der Markt gesättigt sei, weshalb sich die großen IT-Unternehmen bereits nach einer neuen Herausforderung umsehen und diese nun mit künstlicher Intelligenz (AI) anscheinend gefunden haben. Auf den Entwicklerkonferen-

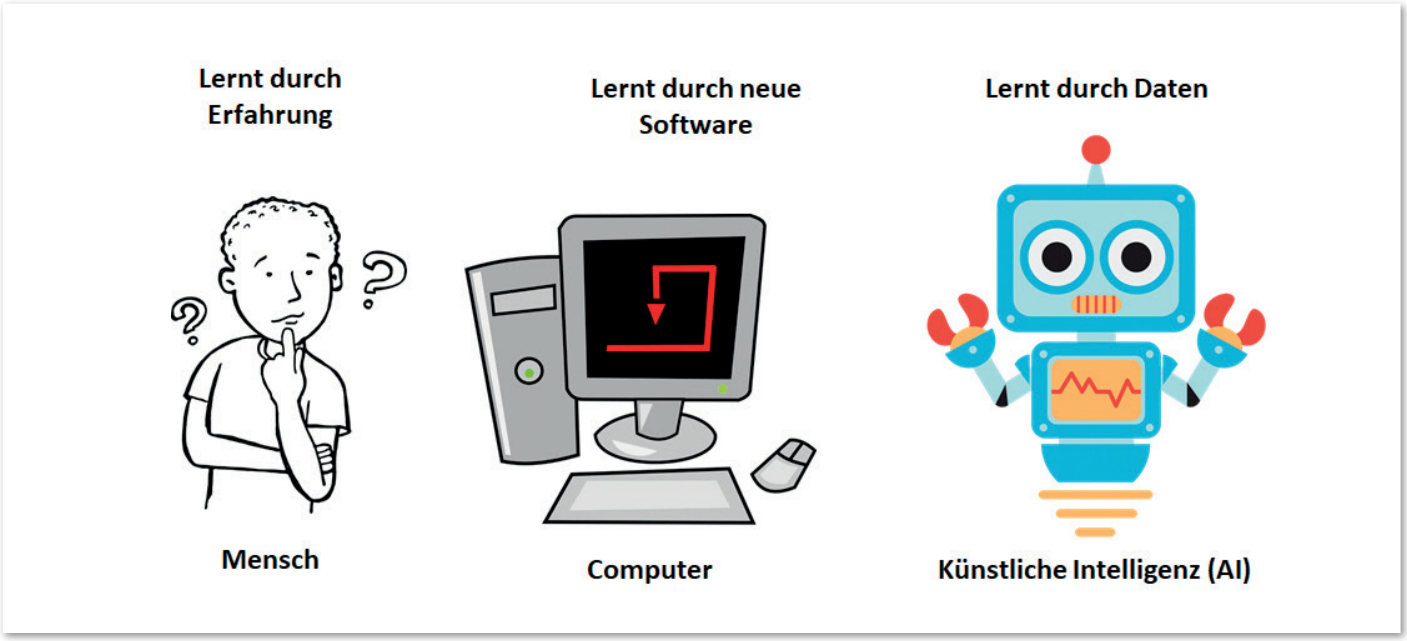


Abbildung 1: Daten sind für Computer das Äquivalent zur gesammelten Erfahrung eines Menschen

zen wird diese neue Richtung auch immer wieder vorgegeben. AI First ist überall zu lesen und zu hören. Aber wird AI First den gleichen Erfolg haben wie Mobile First? Bereits in den 1950er-Jahren hat IBM angefangen, an künstlicher Intelligenz zu arbeiten. Der erste Erfolg war ein Computer, gegen den man Mühle spielen konnte. Im Jahr 1996 hat der Computer Deep Blue das erste Mal den damaligen Weltmeister im Schach geschlagen und gezeigt, dass Computer auch in einem komplexen Spiel dem Mensch überlegen sein können.

Es gab immer wieder große Erfolge im Bereich der künstlichen Intelligenz, aber der ganz große Durchbruch ist nie gelungen. Wieso ist man im Silicon Valley gerade jetzt so zuversichtlich, dass der große Wurf gelingen wird? Die drei Gründe hierfür sind mehr Daten dank Big Data, mehr Rechenleistung dank Moore's Law und bessere Tools. Der Artikel zeigt, in welchen Gebieten AI zum Einsatz kommen kann, wie AI funktioniert und weshalb mehr Rechenleistung und noch mehr Daten so unglaublich wichtig sind.

Einsatzmöglichkeiten für AI

AI hat das Potenzial, den Markt in den meisten Branchen grundlegend zu ändern. Im ersten Schritt haben große IT-Unternehmen in den letzten Jahren bereits ihre Anwendungen überdacht und dem Endanwender mithilfe von AI das Leben einfacher gemacht, ohne dass dieser groß etwas davon mitbekommen hat, dass AI im Einsatz ist. Chatbots wurden entwickelt, die intelligente Antworten auf Fragen geben, Suchmaschinen, die Suchabfragen bereits im Vorfeld erahnen, soziale Netzwerke, die eine persönlich zugeschnittene Nachrichtenseite erstellen, oder Onlinestores, die bereits vorab wissen, was man einkaufen möchten. Doch dies ist erst der Anfang, denn AI kann weitaus mehr.

Eine der ersten Sparten, in denen sich AI als Möglichkeit bewiesen hat, war die Texterkennung mithilfe neuronaler Netze. Bereits einfache Netze haben bisherige konkurrierende Lernverfahren übertroffen, was zum Ergebnis hatte, dass in den letzten fünf Jahren, gerade bei handgeschriebenen Texten, wesentlich bessere Ergebnisse erzielt werden konnten.

Mithilfe der Texterkennung ist es möglich, Bilder aus der Umwelt, die mit der Kamera eines Smartphones aufgenommen wurden, dem Nutzer in einer neuen Form bereitzustellen, etwa die Aufnahme einer Speisekarte in einem Restaurant, die dem Endanwender im Anschluss übersetzt wird.

Texterkennung kann auch für die grundsätzliche Regenerierung von Text aus Bildern herangezogen werden, damit dieser Text anschließend weiterverarbeitet oder nach Wörtern durchsucht werden kann. Beim Einsatz von Texterkennung innerhalb eines Dokumentenmanagementsystems können Metadaten aus der Bilddatei gezogen werden, damit eine spätere Suche nach dem Bild vereinfacht wird. Szenarien wie diese, in denen die Texterkennung sowohl das tägliche Arbeiten als auch die Freizeit eines Menschen erleichtern, gibt es bereits viele.

Im Bereich der Spracherkennung wurden – genauso wie bei der automatischen Texterkennung – seit dem Einsatz von neuronalen Netzen hervorragende Fortschritte erreicht, die die bis dahin bestehenden Systeme in den Schatten stellen und aus dem Markt verdrängen. Die Spracherkennung scheint neben der Bilderkennung einer der wichtigsten Bausteine bei der Entwicklung von Maschinen zu sein, die mit dem Menschen kommunizieren, weil die Spracheingabe als die neue Benutzerschnittstelle zwischen Mensch und Maschine angesehen wird.

Mit Google Home und Amazon Echo sind Produkte auf den Markt gekommen, die nur noch per Spracheingabe gesteuert werden können. Auch wenn der Leistungsumfang dieser Produkte aktuell nur aus trivialen Wetterabfragen oder dem Abspielen von Musik besteht und dadurch etwas eingeschränkt ist, kann man sich sicher sein, dass sich in absehbarer Zukunft die Funktionalität dieser Sprachassistenten, die man sich ins Wohnzimmer stellt, steigern wird.

Natural Language Prozessor (NLP) ist ein Bestandteil von intelligenten Sprachassistenten und eng mit der Spracherkennung verknüpft. Während es bei der Spracherkennung darum geht, die einzelnen gesprochenen Wörter zu erkennen, geht es bei NLP darum, den Sinn zu verstehen, der hinter diesen gesprochenen Wörtern steckt. Mithilfe von NLP können Texte ausgewertet, von Computersystemen verstanden und gedeutet werden. Wichtige Bestandteile dabei sind Syntax und Semantik. Im Bereich NLP ist

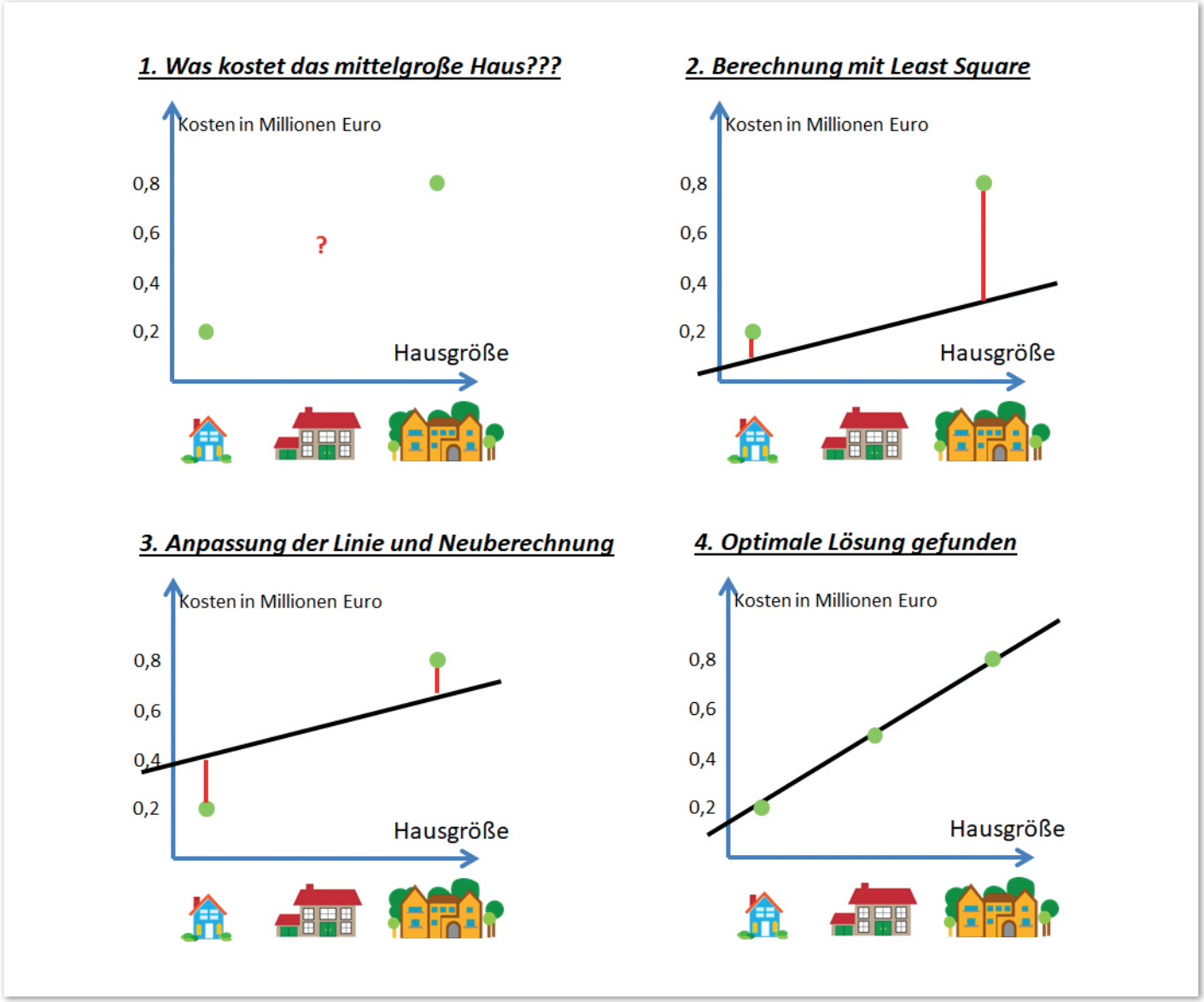


Abbildung 2: Least-Square-Algorithmus in seiner einfachen Form

noch sehr viel Luft nach oben und es muss in Zukunft viel Forschung bezüglich neuronaler Netze betrieben werden, bis man das Ziel eines wirklich intelligenten Systems erreicht hat, das jeden Text interpretieren kann.

Nicht nur das Erkennen von Texten spielt eine Rolle, sondern auch das Auswerten von Bildern. So ist die Gesichtserkennung bei Facebook und Google ein wichtiger Bestandteil geworden, der bei der Gestaltung des Newsfeed oder zur automatischen Erstellung von Fotoalben eingesetzt wird. Auch für das autonome Fahren von Autos ist die Bilderkennung die Grundlage, damit Verkehrsschilder ausgewertet oder damit Menschen, Fahrradfahrer und andere Autos erkannt werden können, um gefährliche Situationen frühzeitig zu erkennen und zu vermeiden.

Krankheiten können in Zukunft früher prognostiziert und damit erfolgreicher behandelt werden. Für einen Menschen ist es schwierig, die Symptome, die beispielsweise auf Krebs hinweisen, sehr früh zu erkennen. Für eine Maschine ist diese Prognose wiederum machbar, weil die Maschine die Informationen von Millionen von ausgewerteten Bildern hat und deshalb auf jedes kleinste Detail achten wird. Da es sich bei Krankheiten, wie beispielsweise Krebs, eben auch um etwas handelt, das sich biologisch entwickelt haben muss und nicht von heute auf morgen da war, glauben viele, die in dem Bereich der IT-Medizin arbeiten, daran, dass AI die Medizin zum Positiven verändern wird.

Auch wenn es viele verschiedene Einsatzmöglichkeiten von AI gibt, so ist der grundlegende Gedanke, wie AI funktioniert, immer der gleiche. In Schritt 1 muss man eine unglaublich große Datenmenge ansammeln, die in Schritt 2 mithilfe verschiedener Algorithmen analysiert und in Schritt 3 dem AI-System zugespielt wird.

Wie AI funktioniert

Menschen lernen durch die tägliche Erfahrung, die sie im Alltag machen, dazu. Computer lernen durch Software-Updates dazu. Von künstlicher Intelligenz spricht man, wenn der Computer kein Software-Update benötigt, um dazuzulernen, sondern in der Lage ist, durch neue Erfahrungen zu lernen. Ein Computer kann zwar keine echten Erfahrungen machen, aber er kann die Ergebnisse von ausgewerteten Daten zugespielt bekommen, auf Basis derer er in Zukunft bessere Entscheidungen treffen kann. Dieser Prozess wird auch „Training eines Modells“ genannt.

Damit ein Computer mit den Daten etwas anfangen kann, müssen diese, wie bereits

erwähnt, mithilfe verschiedener Algorithmen zuerst ausgewertet werden. Nehmen wir zum Beispiel an, ein Haus kaufen zu wollen und den optimalen Preis von der AI vorhersagen zu lassen. Der Einfachheit halber haben wir uns in diesem Artikel dafür entschieden, als vorhandene Daten im Vorfeld nur zwei Häuser zu haben, deren Preis uns bekannt ist. Wir wissen, dass ein kleines Haus 200.000 Euro und ein großes Haus 800.000 Euro kostet. Nun möchten wir mithilfe des Least-Square-Algorithmus herausfinden, was ein mittelgroßes Haus kostet. Dafür zeichnen wir die vorhandenen Daten in ein Diagramm ein (siehe *Abbildung 2*, links). Die grünen Punkte stellen die Preise der beiden Häuser dar. Anschließend zeichnen wir eine schwarze Linie in das Diagramm ein, berechnen die Abstände zwischen der schwarzen Linie und den grünen Punkten und addieren die Summe aller Abstände im Quadrat zusammen.

In *Abbildung 2*, Mitte links, ergibt das eine Summe von 0,1^2 + 0,6^2 = 0,37. Im Anschluss wird die schwarze Linie nach oben oder nach unten verschoben und die Berechnung wird neu durchgeführt.

In *Abbildung 2*, Mitte rechts, führt dies zu der Summe 0,2^2 + 0,1^2, was 0,05 ergibt. Die Linie wird nun so lange mit dem Ziel verschoben, die Summe der Abstände so gering wie möglich zu halten. Dieser Vorgang wird auch als „Training des Modells" bezeichnet. In *Abbildung 2*, ganz rechts, kann man als Ergebnis nun sehen, dass man für ein mittelgroßes Haus in etwa 450.000 Euro zahlen sollte.

An dem Beispiel kann man sehr gut erkennen, weshalb eine große Menge an Daten und eine hohe Computer-Performance von Vorteil sind. Je mehr Daten vorliegen, desto genauer wird die schwarze Linie (Modell), die auch nicht zwingend immer gerade sein muss. Je mehr Daten es gibt, desto länger dauert allerdings auch das Training des Modells, weshalb Rechenleistung wichtig ist. Von vielen IT-Unternehmen werden Frameworks und Clouds zur Verfügung gestellt, mit deren Hilfe das Training der Modells effizient und schnell vorgenommen werden kann, weil man mit dem eigenen Desktop-Computer oder Laptop recht schnell das Ende des Möglichen erreicht.

Wer mit AI Geld verdient

Möglichkeiten, um mit AI Geld zu verdienen, gibt es viele. Bisher waren die Profiteure hauptsächlich AI-Start-ups, die an Chatbots oder an Frameworks für A-Engines gearbeitet haben. Das Ziel dieser Start-ups war oftmals von Anfang an darauf ausgelegt, dass man nicht selbst den Durchbruch schaffen wollte, sondern dass man das eigene Unternehmen an einen der Global Player (Google, Amazon, Oracle, Facebook etc.) weiterverkaufen wollte. Diese Geschäftsidee hat in den letzten Jahren auch sehr gut funktioniert, wie die folgenden Zahlen zeigen. So wurden mehr als 140 Start-ups seit dem Jahr 2011 verkauft, 40 davon allein im Jahr 2016.

Nicht nur die Software-Entwicklung verdient an AI, sondern auch die Hardware-Branche. Dies liegt daran, dass für das Training von AI-Modellen mit GPUs eine neue Generation von Prozessoren eingesetzt wird. GPUs sind gegenüber CPUs im Vorteil, weil sie eine schnellere Abarbeitung von AI-Tasks ermöglichen – dank Tausender Cores, die parallel arbeiten.

Chip-Hersteller wie Nvidia haben deshalb in den letzten Monaten einen regelrechten Boom erlebt, der durch deren Aktienwert widergespiegelt wird, der sich im letzten Jahr mehr als verdreifacht hat, weil die Nachfrage an GPUs derart gestiegen ist. Google ist nun noch einen Schritt weitergegangen und hat die TPU entwickelt, eine Prozessoreinheit, die speziell für die optimale Zusammenarbeit mit dem von Google entwickelten AI-Framework Tensorflow ausgerichtet ist.

Da die AI-Frameworks und die notwendigen Prozessoreinheiten für eine einfache Entwicklung von AI-Software nun vorhanden sind, wird auf vielen Entwicklerkonferenzen dafür geworben, diese Frameworks einzusetzen und mit deren Hilfe Software zu entwickeln, die das Leben des Kunden grundlegend verändern und einfacher machen soll.

Wie diese Software aussehen wird, kann ich Ihnen nicht sagen, aber mit einem Blick in die Vergangenheit lassen sich Beispiele von Software nennen, die bereits das Verhalten des Kunden verändert haben. Im Internet-Zeitalter hat Amazon das Kaufverhalten des Nutzers geändert und Facebook die Art und Weise, wie wir unser Privatleben teilen. Im mobilen Zeitalter hat Uber das Taxifahren revolutioniert und WhatsApp die Kommunikation untereinander.

Hinweise, in welche Richtung die AI-Software gehen könnte, wurden bereits in einem vorherigen Kapitel beschrieben. Den Autoren scheint es sehr wahrscheinlich, dass in den nächsten Jahren einige IT-Firmen den großen Durchbruch dank AI schaffen und bald zu den Großen im Silicon Valley gehören werden.

Nicht nur IT-Unternehmen werden von AI profitieren. Auch ein Großteil der Unternehmen aus vollkommen anderen Bereichen werden dank AI Geld sparen können. Auf den ersten Blick denken viele Unternehmen daran, mit AI Geld zu verdienen, weil sie sich dadurch Arbeitsplätze sparen können. Die Autoren sehen dank AI allerdings einen noch größeren Vorteil für Unternehmen darin, dass sie die Qualität ihrer Produkte und Dienstleistungen enorm steigern können, was zu zufriedeneren Kunden führen wird – das ist, was was langfristigen Erfolg ausmacht.

Fazit

AI hat in den letzten sechzig Jahren ein Auf und Ab erlebt. Das Interesse an AI hat bereits in den 1980er-Jahren einen kurzfristigen Boom ausgelöst. Allerdings fehlte es an Geschäftsideen, die mit der damals mäßigen AI hätten umgesetzt werden können, was hauptsächlich an den mangelnden Daten und der geringen Rechenleistung lag und dazu geführt hat, dass sich der AI-Boom einen ganz langen Winterschlaf gegönnt hat. Aus diesem Winterschlaf ist AI vor ein paar Jahren wieder aufgewacht, um diesmal richtig durchzustarten, was auch an einem Investitionsboom zu erkennen ist.

Im Jahr 2016 wurden 1,2 Milliarden Dollar in AI-Start-ups investiert, was in etwa das Zehnfache von dem im Jahr 2011 ist. Die Autoren sind sehr zuversichtlich, dass der vor etwa fünf Jahren gestartete Boom diesmal nicht so schnell wieder in den Winterschlaf gehen wird, und freuen sich auf die nächsten Jahre.

Andreas Dohren
dohren@lumabit.de

Tobias Huber
huber@lumabit.de

Ethik und künstliche Intelligenz

Michael Mörike, Integrata-Stiftung für humane Nutzung der Informationstechnologie

Künstliche Intelligenz (KI) wird große Veränderungen bringen: spürbare Verbesserungen unserer Lebensqualität und unerwartete Gefahren. Es ist höchste Zeit, sich mit möglichen Folgen vertraut zu machen. Um die Wirkung der KI in eine von uns gewünschte Richtung zu lenken, müssen wir ihr ein ethisches Verhalten beibringen. Damit das gelingen kann, gilt es, unsere eigene Ethik zu analysieren und zu begreifen. Beim Kongress „Ethik und KI" kann jeder daran mitwirken, der etwas dazu zu sagen hat.

Künstliche Intelligenz (KI) ist aktuell ein Hype. Alle reden davon, dass sie große Veränderungen unserer Arbeits- und Lebenswelt bringen wird. Da der Begriff der Intelligenz nicht klar definiert ist, unterscheidet man in der Technik zwei Arten von künstlicher Intelligenz: schwache KI („weak AI") und starke KI („strong AI"). Die schwache ist keine Intelligenz in dem Sinn, wie wir sie uns Menschen zuschreiben. Die starke wäre es schon, wenn es sie denn (schon) gäbe. An ihr wird noch geforscht, ohne dass aktuell klar ist, wie sie einmal aussehen oder funktionieren könnte.

Wenn man von KI spricht, meint man fast immer die schwache Form. Schwache KI wird immer als Software realisiert, die so programmiert ist, dass sie lernen und entscheiden kann. Zunächst wird sie trainiert und lernt beispielsweise, Muster zu erkennen. Einmal trainiert, kann sie neue Muster mit dem gelernten Material vergleichen und daraus Schlüsse ziehen oder Entscheidungen treffen. Oberflächlich betrachtet sieht das dann so aus, als ob die Maschine intelligent wäre. Daher der technische Begriff „künstliche Intelligenz".

Künstliche Intelligenz war in den vergangenen Jahrzehnten der Informatik schon mehrfach ein Hype, der nach einiger Zeit aus Enttäuschung über die damals noch schwachen Ergebnisse immer wieder abgeklungen ist. Diesmal aber sieht es anders aus: Vor etwa drei Jahren ist beim Lernen mit neuro-

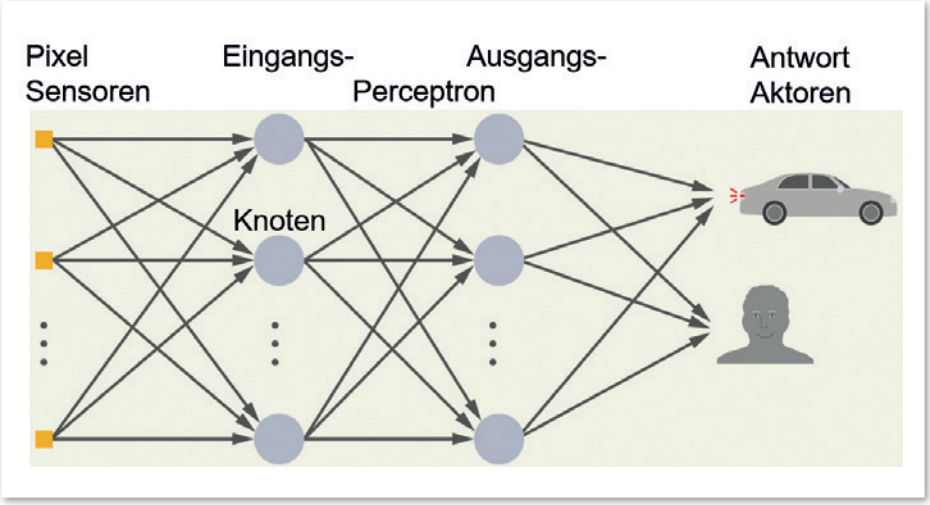


Abbildung 1: Die Berechnung von Entscheidungen ist sehr komplex

nalen Netzen ein technischer Durchbruch gelungen. Man hat nicht – wie früher - nur einige wenige (etwa ein Dutzend) Schichten eingesetzt, sondern sehr viele (mehrere Hundert). Das machte die seit früher deutlich gestiegene Rechenpower moderner Großrechner möglich. Schließlich sind bei jedem einzelnen kleinen Lernschritt viele Millionen von Datenpunkten zu berechnen *(siehe Abbildung 1)*.

Wie KI funktioniert

Die erfolgreichsten heutigen KI-Systeme bestehen aus neuronalen Netzen (NN), die in gewisser Hinsicht den Neuronen im menschlichen Hirn nachgebildet sind (nicht wirklich). NN bestehen aus Knoten, die in Schichten angeordnet sind, sogenannten „Perceptrons". Jeder Knoten multipliziert die Eingangssignale mit einem Gewicht, summiert sie, normiert sie und gibt sie an die nächste Schicht weiter. „Wissen" ist somit in den Gewichten gespeichert. Einfache NN bestehen aus mindestens drei, meist aus zehn bis zwanzig Schichten, neuerdings auch aus Hunderten von Schichten, die ihrerseits jede aus Hunderten bis Tausenden von Knoten zusammengesetzt sein können.

Lernen bedeutet, die Gewichte in den Schichten zu verändern. Dafür gibt es verschiedene mathematische Methoden und Algorithmen, die mehr oder weniger gut lernen lassen. Sie sorgen dafür, dass die Antwort – bei gegebenem Input-Signal – am Ausgang richtig wird (im Sinne des gewünschten Lern-Ergebnisses). Eine davon heißt „Back-Propagation", was deutlich macht, dass die Gewichte von hinten, also vom Ausgang her, in kleinen Schritten passend verändert werden. Bis auf vielen Bildern alle Gegenstände richtig erkannt werden, braucht es viele Tausend Durchläufe mit jeweils kleinen Änderungen.

Das „Wissen" eines NN ist, wie schon genannt, in seinen Gewichten gespeichert; es ist also holografisch: Nicht ein einzelnes Gewicht verursacht eine bestimmte Antwort, sondern alle Gewichte gemeinsam ergeben die richtige Antwort. Es ist nicht mehr einfach nachvollziehbar, wie ein bestimmtes Gewicht zu einer bestimmten Antwort beiträgt, also auch nicht, wie eine bestimmte Antwort zustande kommt. Es kann auch sein, dass die Verteilung der Gewichte sich unterscheidet, je nachdem, in welcher Reihenfolge etwas gelernt wurde, obwohl das in den Gewichten des NN gespeicherte „Wissen" das Gleiche ist.

Da sehr viele sehr einfache Rechenoperationen parallel ausgeführt werden müssen, eignen sich Grafik-Prozessoren (GPU) von Computern recht gut dafür. Sie werden auch von entsprechender Software eingesetzt, die man als Open-Source von Google, Facebook oder Microsoft kostenlos aus dem Netz herunterladen kann. Jeder begabte junge Mensch kann damit also seine eigene KI bauen.

Bestehende KI-Anwendungen

Wenn man sich heute existierende Anwendungen vor Augen führt, wird deutlich, dass KI vor allem dann stark ist, wenn es um Muster- oder Bild-Erkennung geht, wie der folgende Überblick zeigt:

- Bäume, Blumen, Pflanzen und Pilze bestimmen wir heute per Smartphone-App
- PlantVillage erkennt per Smartphone-Fotos Krankheiten bei Pflanzen
- Fußabdrücke von Tieren im Wald kann man per Smartphone-App bestimmen
- Eine Smartphone-App nennt uns die Namen von Bergen in den Alpen
- Nahrungsmittel auf dem Teller – dank der vielen Fotos von Mahlzeiten im Netz
- Die „VIP-Kamera" erkennt Prominente in Ladengeschäften und macht das Personal darauf aufmerksam, damit dieses sie bevorzugt bedienen kann
- Die Gesichtserkennung dient als Zugangsberechtigung oder zur Alarmauslösung für die Personensuche an Berliner Plätzen
- SeeingAI oder AIpoly sagt per Smartphone-App einem Blinden beim Gang durch die Stadt, was er da vor sich hat: einen Bekannten, ein falsch parkendes Auto, eine Apotheke, einen Bäcker oder auch spielende Kinder im Park etc.
- Schadensmeldungen von Verkehrsunfällen werden heute teilweise vollautomatisch anhand von Smartphone-Fotos abgewickelt
- In der Medizin werden bestimmte Krebsarten (Beispiel Kehlkopfkrebs) von KI anhand von Aufnahmen besser erkannt als von den meisten Ärzten
- DNA-Sequenzierungen lassen sich per KI im Hinblick auf Krebsarten schneller auswerten
- Erkennung von Autismus ist schon bei Säuglingen aufgrund von Gehirn-Scans möglich
- Aus Langzeit-EKGs erkennt KI, ob Herzprobleme vorliegen und welche
- KI zeichnet die Ränder von Tumoren auf CT-Scans nach und bereitet damit deren Bestrahlung vor
- Per Kinect (einem Modul für die Gestensteuerung von Spielen) kann man Störungen von depressiv Kranken diagnostizieren
- KI überwacht Senioren und lernt deren übliche Gepflogenheiten. Bei Abweichung vom oft sehr eigenwilligen individuellen Muster alarmiert sie Hilfe
- Die Ebay-Bildersuche erlaubt, im Netz per Foto ein Produkt zu suchen (und zu bestellen)
- Durchforsten von Urteilen oder anderen juristischen Dokumenten hilft den Anwälten, schneller einen Prozess vorzubereiten und Gewinnchancen abzuschätzen. Nebenbei macht es den Nachwuchs von Anwälten quasi arbeitslos
- KI kann Plagiate bei Texten automatisch erkennen

- KI beherrscht das Übersetzen von Texten in andere Sprachen inzwischen sehr gut
- Plagiate in der Musikindustrie kann KI schnell aufdecken
- In der Forschung hilft KI bei der Identifizierung von Erdbebenwellen und deren Unterscheidung von Wellen, die von unterirdischen Atomexplosionen ausgelöst wurden
- KI wird zur Erkennung von Gravitationswellen eingesetzt
- Chatbots: Alexa, Cortana, Echo, OK-Google, Siri etc. arbeiten mit KI
- Bots zur glaubhaften Verbreitung von Fake News setzen KI ein
- In Spielen ist KI dem Menschen längst überlegen: Bei Schach schon seit Langem, bei Go seit dem Jahr 2017 und sogar beim Pokern ist KI jedem menschlichen Spieler weit überlegen
- Eine derzeit viel diskutierte Anwendung ist das autonom fahrende Auto. Gleiches gibt es natürlich auch für Lastkraftwagen und Ähnliches für Schiffe
- Eine weitere (fürchterliche) Variante sind autonome Waffensysteme, die selbstständig entscheiden, wen sie umbringen sollen

Die Liste ließe sich weiter fortsetzen. Insbesondere aber verlängert sie sich täglich durch immer neue Anwendungen.

Wie geht es weiter?

Die Muster-Erkennung allein bringt (wie beschrieben) bereits einen deutlichen Nutzen. Sie kann aber auch zur Wahrnehmung der Umgebung eingesetzt werden. So geschieht es zum Beispiel beim autonom fahrenden Auto. Mehrere Erkennungssysteme für die Umwelt lassen sich kombinieren, sodass ein KI-System in einem Roboter (im autonom fahrenden Auto) seine Umwelt im Prinzip beliebig genau erfassen kann. Wenn alle (wichtigen) Objekte aus dem Umfeld erfasst sind, können sie mithilfe einer Ontologie miteinander verknüpft und eingeordnet werden.

Wenn die KI, etwa in einem Roboter dann noch etwas bewirken soll, braucht sie ein klares Ziel und einen Satz von möglichen und erlaubten Handlungen, die sie mithilfe von Aktoren ausführt, um das Ziel zu erreichen oder ihm näherzukommen. Für nützliche Handlungen des ausführenden Roboters benötigt die KI eine (nützliche) Zielsetzung. Das autonom fahrende Auto zum Beispiel soll das ihm genannte Ziel ansteuern. Diese Vorgabe allein reicht aber nicht. Es soll ja auch sicher fahren, also alle Verkehrsregeln beachten und dabei auch auf andere Verkehrsteilnehmer Rücksicht nehmen. Kurz: Es braucht einen ganzen Satz von Verhaltensregeln. Davon sind einige sehr klar formulierbar und daher relativ einfach umzusetzen, wenn sie beispielsweise durch die Verkehrsordnung vorgegeben sind. Es gibt aber auch Fälle, in denen die Verkehrsordnung nicht weiterhilft. Beispiel: Das Auto steht im Stau und von hinten kommt offensichtlich ein schnelles Fahrzeug. Es droht ein Auffahrunfall. Soll es nun nach vorne und ganz dicht auffahren, um dem auffahrenden Auto mehr Platz zum Bremsen zu geben? Was ist, wenn das nicht reicht? Soll es ganz schnell nach rechts oder links ausscheren? Dabei eventuell einen hoffentlich kleinen Schaden in Kauf nehmen, um größeren Schaden zu verhindern? Jeder Mensch wünscht, dass es den möglichen Schaden minimiert. Was bedeutet das aber? Kann es bedeuten, dass es die Insassen opfert, um noch größeren Schaden zu vermeiden?

Um dieses und ähnliche Themen lässt sich trefflich streiten. Letztlich läuft es aber immer darauf hinaus, dass der Roboter auch ethische Regeln und nicht nur Verkehrsregeln befolgen soll. Damit die ethischen Regeln nicht zu einer Maschinen-Ethik verkommen, sondern dem Menschen nützen, sollen die Regeln unserer menschlichen Ethik folgen. Damit ist eine Ethik gemeint, die wir Menschen uns gegeben haben, um gut miteinander zusammenleben zu können.

Ethik für die KI

Um die Problematik zu verdeutlichen, ein weiteres Beispiel aus dem Bereich der Pflege, wo bereits heute Roboter eingesetzt werden. Soll der Roboter darauf achten, dass der Patient regelmäßig seine Medikamente einnimmt? Soll er ihn dazu zwingen? Bei lebenswichtigen Medikamenten kann das zweifellos richtig sein. Was aber ist mit Medikamenten, die nur dazu dienen, dass der Patient sich wohlfühlt? Wenn der Patient lieber auch mal auf die Einnahme von weniger wichtigen Medikamenten verzichtet, weil es ihm halbwegs gut geht, wie soll sich der Roboter verhalten? Bei diesem Dilemma wird deutlich, dass die Ethik prinzipiell situationsabhängig ist und dass sie viele Ziele mit unterschiedlicher Priorität zu verfolgen hat.

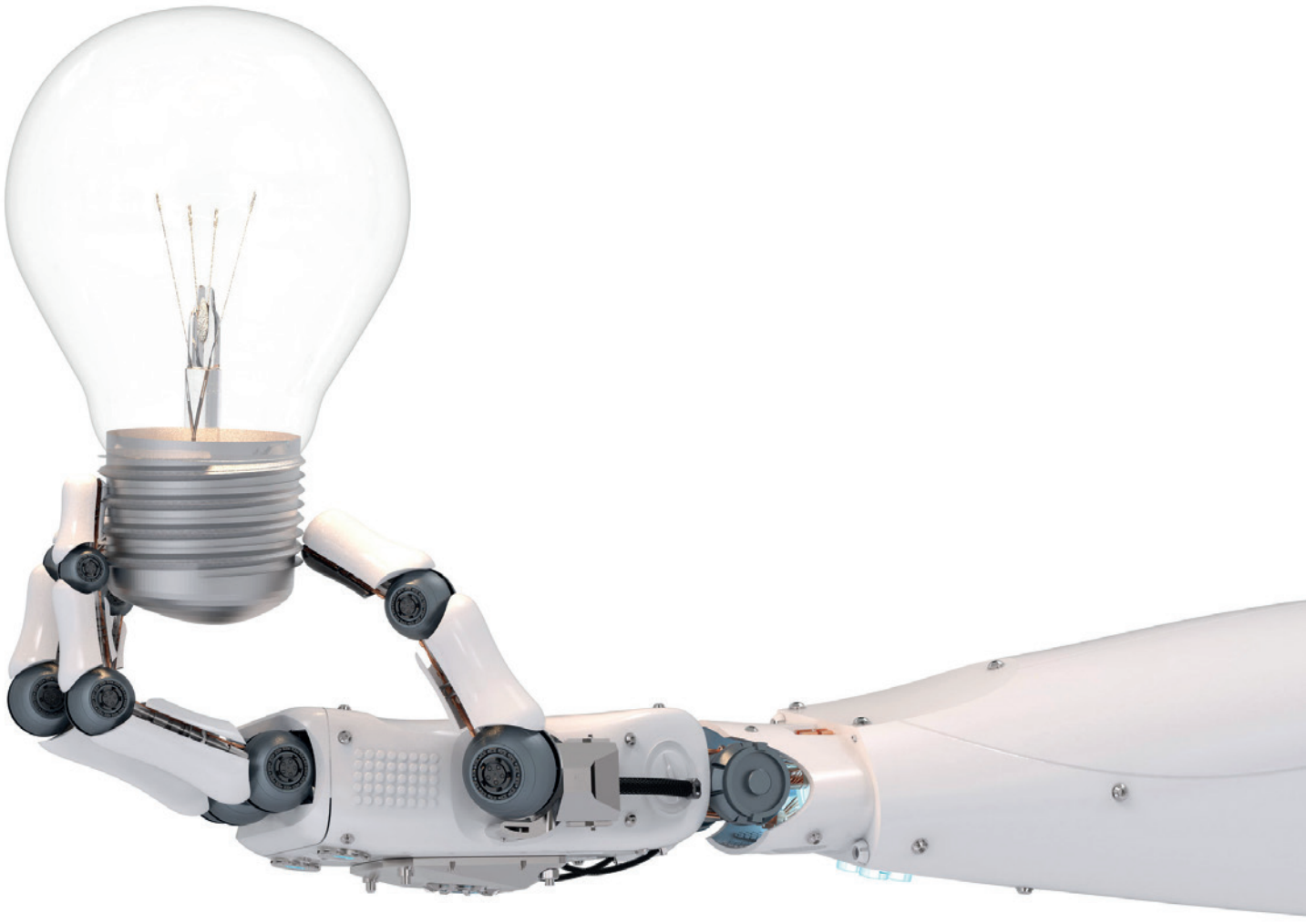
Die Ethik relativiert (situationsabhängig) die vorgegebenen Ziele und bringt sie in eine Prioritäts-Reihenfolge. Wir Menschen haben Ethik im Laufe unseres Lebens gelernt, ohne sie in klar definierte Regeln zu gießen. Viele große Philosophen haben Jahrhunderte lang darüber nachgedacht, ohne ein sauber definiertes Regelwerk wie etwa die Physik zu hinterlassen. Genau das aber benötigen wir jetzt für die Technik der KI. Noch ist sogar unklar, ob man Ethik überhaupt in ein endliches, sauber definiertes Regelwerk fassen kann. Ist es eine gute Idee, dafür Fuzzy Logic einzusetzen?

Ethik technisch fassbar machen

Kann man ethisches Handeln einer KI durch Lernen beibringen? Schließlich bedeutet es eine Prioritätsreihenfolge von Zielen, die situationsabhängig ist. Welche ethischen Regeln soll die KI lernen? Wie soll sie die Regeln lernen? Soll sie es aus dem Verhalten von Menschen lernen, auch wenn diese immer wieder eine gewisse Schieflage zeigen? Beispiel: Bei der Begnadigung von Menschen, die eine Gefängnisstrafe in den USA abzusitzen haben, hat sich gezeigt, dass grundsätzlich Weiße besser beurteilt werden als Schwarze. Wie kann man solche Schieflagen vermeiden?

Diese und ähnliche Themen behandelt der jährliche Kongress Ethik und KI, der von der Integrata-Stiftung für humane Nutzung der IT *(siehe „www.integrata-kongress.de/kongress/2017-euki")* gemeinsam mit dem Weltethos-Institut Tübingen und der Giordano-Bruno-Stiftung jährlich im Herbst durchgeführt wird. Das Kongresskonzept vertraut auf die Weisheit der Vielen und richtet sich bewusst an die Zivilgesellschaft und nicht an die Wissenschaft. Schließlich müssen die ethischen Regeln für die KI eines Tages von allen Menschen akzeptiert werden. Daher ist es gut, wenn möglichst viele Menschen dabei mitmachen.

Michael Mörike
michael.moerike@integrata-stiftung.de



Von Big Data zu künstlicher Intelligenz – maschinelles Lernen auf dem Vormarsch

Andreas Koop, enpit GmbH & Co. KG

Seit Jahrzehnten fasziniert das Gebiet der künstlichen Intelligenz (KI). Neue Lernverfahren auf großen Datenbeständen (wie Deep Learning) haben in den letzten Jahren insbesondere die Sprach- und Bilderkennung essenziell verbessert. Wie können Unternehmensanwendungen von den Errungenschaften Gebrauch machen? Neue Frameworks und Plattform-Services abstrahieren die Komplexität darunterliegender Algorithmen und machen maschinelle Lernverfahren für Entwickler intelligenter Assistenzsysteme einfach zugänglich. Auf Basis ausreichend großer Trainingsdaten können Muster erkannt, Handlungsempfehlungen abgeleitet oder autonome Entscheidungen getroffen werden.

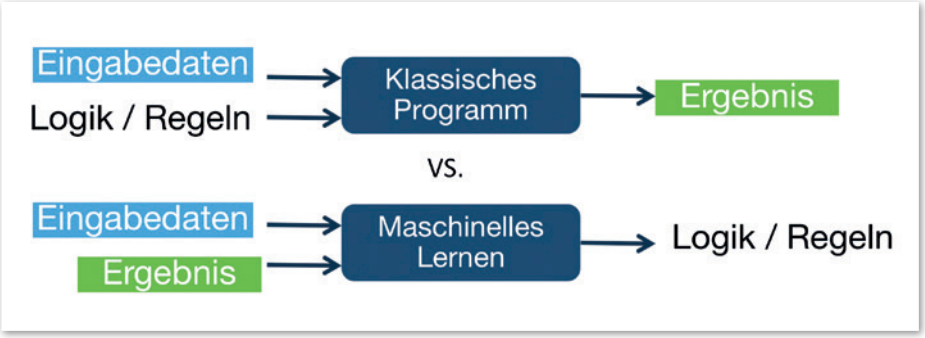


Abbildung 1: Paradigmenwechsel – vom Programm zum maschinellen Lernen [2]

```
func schaetzeVerkaufspreis(baujahr, flaeche, stadt) {
    return baujahr * w1 + flaeche * w2 + stadt * w3 +b;
}
```

Abbildung 2: Einfache Funktion als Machine-Learning-Modell

Der Artikel gibt einen Einblick in die Welt der künstlichen Intelligenz und stellt grundlegende Konzepte und Frameworks für maschinelles Lernen vor. Anhand eines Praxisbeispiels aus der Lego-Welt werden analoge Problemstellungen aus dem Unternehmenskontext greifbar gemacht.

Einführung und Überblick

Mit Big Data assoziiert man der Definition nach Massendaten, die zu groß, zu komplex und oft zu schwach strukturiert sind, um sie mit herkömmlichen Verarbeitungsmethoden auszuwerten [1]. Es werden immer mehr Daten, da sie zunehmend völlig maschinell erzeugt werden – sei es durch Log-Dateien, Transaktionen, Kameras, Mikrofone oder sonstige Sensoren (Geologie, Meteorologie etc.). Klassische Software-Programme (Ein-/Ausgabeverarbeitung) sind zu limitiert, um durch noch so ausgereifte Algorithmen und Regeln Ergebnisse zu prognostizieren oder Hypothesen herzuleiten.

Maschinelles Lernen ermöglicht auf Basis bekannter Ein-/Ausgabepaare, Regeln und Funktionen zu ermitteln, die anschließend auf unbekannte Daten angewendet werden können. Es ist quasi in der Lage, aus bekannten Fakten (Eingabedaten inklusive dazugehörigem Ergebnis – auch „Trainingsdaten" genannt) Muster zu erkennen und die dafür notwendige Logik herzuleiten. Die ermittelte Logik lässt sich anschließend auf herkömmliche Weise anwenden. *Abbildung 1* zeigt den Paradigmenwechsel vom klassischen Programm zum maschinellen Lernen.

Im Wesentlichen unterscheidet man zwei Kategorien von maschinellem Lernen:

- *Überwachtes Lernen*
  Es liegen valide Ein-/Ausgabe-Daten vor; der Algorithmus lernt die passende Funktion/Logik.

- *Unüberwachtes Lernen*
  Es liegen nur Eingabedaten vor; der Algorithmus versucht, charakteristische Muster etwa für eine Clusterbildung zu erkennen und die dafür erforderliche Funktion/Logik zu lernen

Machine Learning verstehen

Angenommen, man möchte Immobilien-Preise in deutschen Städten vorhersagen. Dafür liegen jede Menge Referenzdaten über Verkaufspreise vor. Vereinfacht dargestellt – eine Tabelle mit folgenden Informationen: Baujahr, Wohnfläche, Stadt und letztlich dem erzielten Verkaufspreis. Also Eingabedaten und das zugehörige Ergebnis. Man könnte nun hergehen und Regeln ableiten, doch das wäre viel zu mühsam. Der Fall ruft nach überwachtem maschinellen Lernen.

Im ersten Schritt versucht man, ein geeignetes Modell zu finden, um das Problem zu beschreiben. Vom Prinzip her muss es durch eine mathematische Funktion abbildbar sein – Eingabedaten modelliert man als Parameter und das Ergebnis der Funktion ist die Preis-Vorhersage. Da die Eingaben offensichtlich Einfluss auf das Ergebnis haben, modelliert man dies durch entsprechende Koeffizienten (w1, w2, w3) sowie zusätzlich eine Konstante (b) *(siehe Abbildung 2)*.

Nun kann der Lernprozess starten. Mit den Referenzdaten – auch „Trainingsdaten" genannt – wird der Algorithmus durchlaufen, bis sich ein Optimum eingestellt hat. Man startet mit zufälligen Werten für die Unbekannten w1, w2, w3 und b und vergleicht die damit errechnete Schätzung mit dem tatsächlichen, bekannten Wert. Typischerweise wird es eine Abweichung geben. Diese Abweichung gilt es zu optimieren. Dazu werden die Unbekannten so lange angepasst, bis sich die Abweichung nicht mehr minimieren lässt. Maschinelles Lernen heißt also vom Grundsatz her, ein Optimierungsproblem zu lösen. Das kann gegebenenfalls einige Zeit in Anspruch nehmen, falls sämtliche Kombinationen der Koeffizienten probiert werden müssen. Man spricht daher nicht von ungefähr von der Trainingsphase eines Machine-Learning-Modells. Die auf diese Weise ermittelte Logik lässt sich anschließend auf neue Daten anwenden, um (mit einer Wahrscheinlichkeit behafteten Genauigkeit) Preis-Vorhersagen zu treffen.

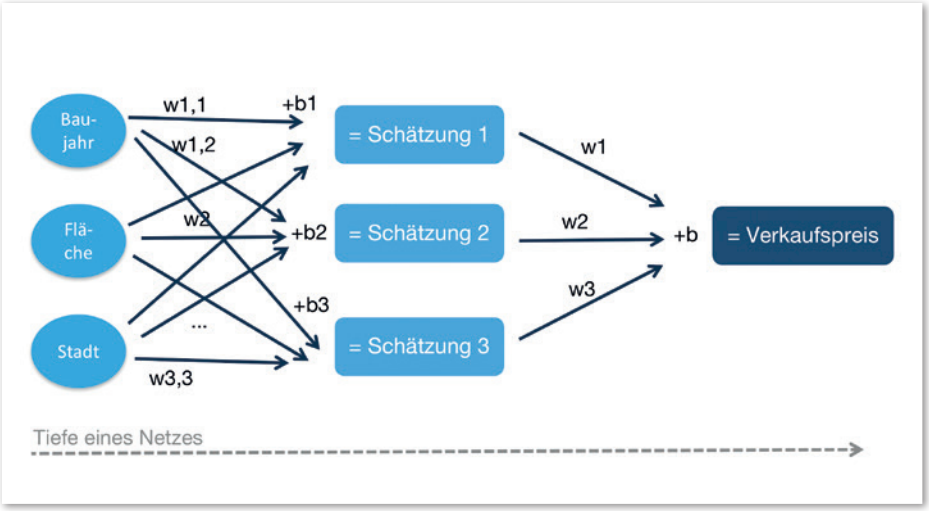


Abbildung 3: Berechnungslogik als künstliches neuronales Netzwerk modelliert

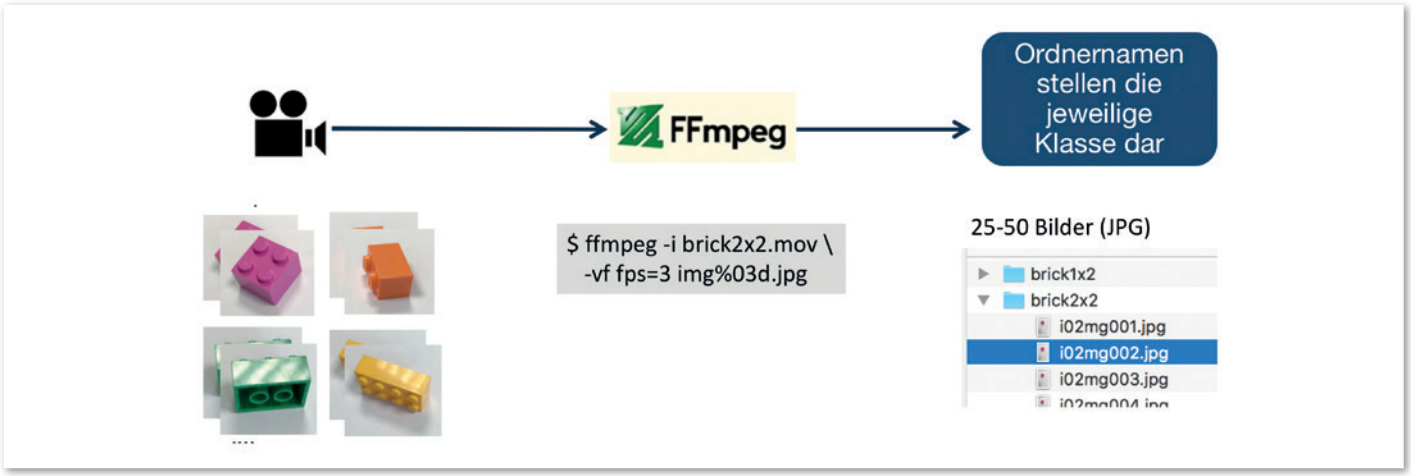


Abbildung 4: Trainingsdaten (Bilder) aus Videos gewinnen

Deep Learning und wofür es geeignet ist

Das bisher ermittelte Modell wird nur in den Fällen funktionieren, in denen eine lineare Beziehung zwischen den Eingabedaten und dem Verkaufspreis existiert. In der Praxis wird es jedoch Fälle geben, die einer komplexeren Modellierung bedürfen, etwa wenn das Baujahr in großen Städten einen anderen Einfluss auf den Preis hat als in kleinen. Um mögliche Sonderfälle abzudecken, könnte der Trainings-Algorithmus für diese Fälle mit unterschiedlichen Koeffizienten zunächst mehrere Preisschätzungen (etwa drei verschiedene) berechnen. Anschließend würde man nach dem gleichen Prinzip aus den drei Preisschätzungen den finalen Verkaufspreis ermitteln.

Um das Prinzip zu visualisieren, hat sich anstelle einer Funktion eine grafische Darstellung etabliert. *Abbildung 3* zeigt auf der linken Seite die Eingabe-Parameter und am rechten Ende das Funktionsergebnis. Man kann leicht erkennen, dass der Trainingsaufwand steigt, da eine weitere Ebene für Kombinationen von weiteren Koeffizienten eingeführt wird. Derart aufgebaute und trainierte Modelle sind jedoch deutlich leistungsstärker. Man spricht auch von einem künstlichen neuronalen Netzwerk (KNN).

Von Deep Learning spricht man also, wenn das zugrunde liegende Modell auf einem neuronalen Netz basiert und mehrere Ebenen umfasst. Leistungsstarke Netze zur Objekterkennung bestehen teils aus dreißig und mehr Ebenen. Man kann sich leicht vorstellen, dass in der Trainingsphase mit Millionen von Eingabedaten und derart ausgeprägtem Netz sehr viele Kombinationen von Parametern exploriert werden müssen. Das Training komplexer Netze mit einer CPU kann mehrere Wochen dauern. Daher kommen verstärkt GPUs und parallele Trainingsalgorithmen zum Einsatz. Wem das nicht ausreicht, der kann auf speziell für Machine Learning entwickelte TPUs (Tensor Processing Units) von Google in der Cloud zugreifen [4], um die Trainingszeiten zu beschleunigen.

```
#!/bin/sh
python retrain.py \
  --how_many_training_steps=250 \
  --model_dir=inception \
  --image_dir=training-data \
  --output_graph=retrained_graph.pb
```

Abbildung 5: Aufruf zum Trainieren des Modells

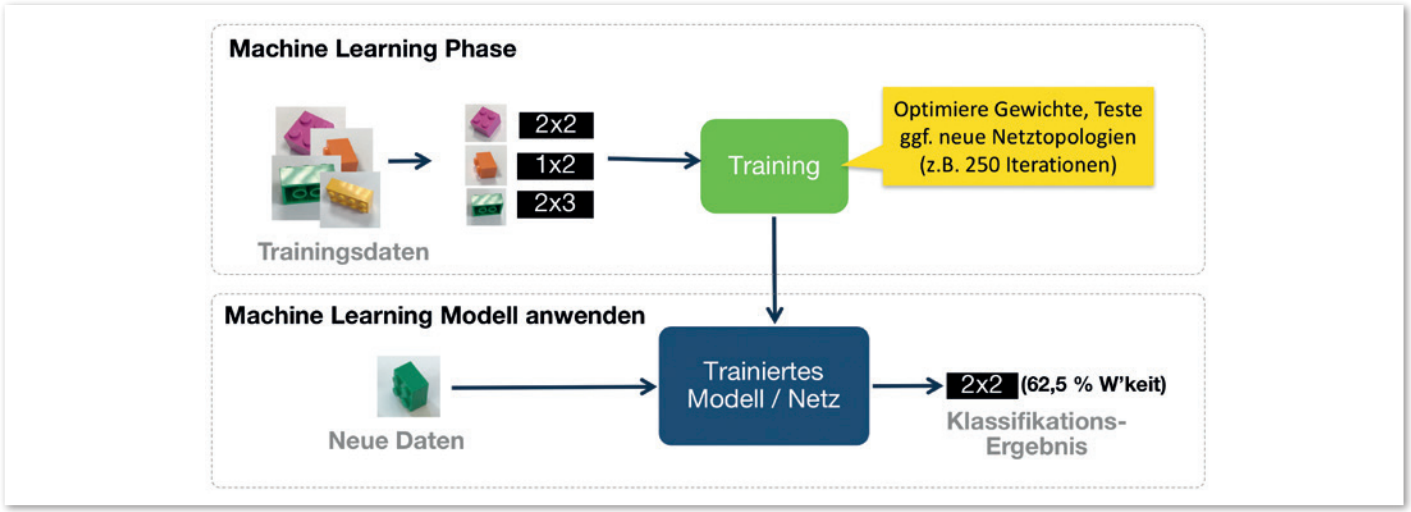


Abbildung 6: Prinzipielles Vorgehen bei überwachtem Machine Learning

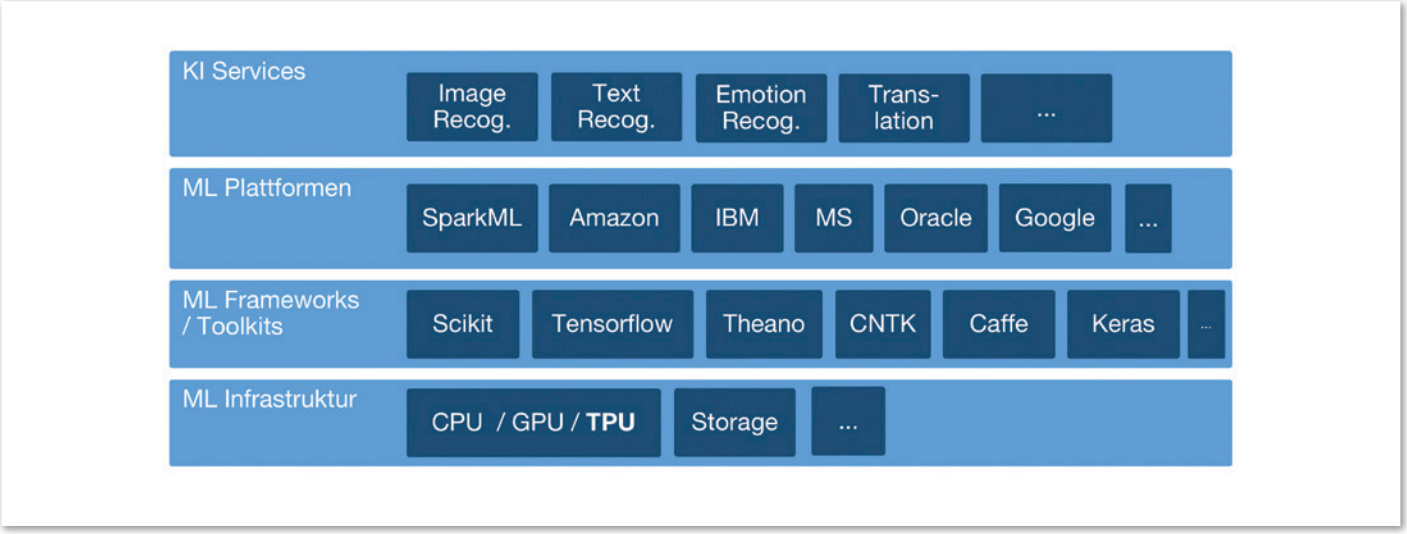


Abbildung 7: Überblick und Einordnung von KI/ML-Werkzeugen und -Plattformen

Lego-Steine mit Deep Learning erkennen

Die bisherige Darstellung war sehr theoretisch. Um zu sehen, wie aufwendig es ist, Machine Learning in der Praxis anzuwenden, ein anschauliches Beispiel aus der Lego-Welt. Es geht um einen Haufen Lego-Steine und die Frage, welche Lego-Projekte sich damit realisieren lassen. Dafür benötigt man in erster Linie ein System, das in der Lage ist, Lego-Steine automatisiert zu erkennen und zu klassifizieren. Um die Prinzipien maschinellen Lernens anzuwenden, wird wie folgt vorgegangen:

- *Trainingsdaten gewinnen*
  Zahlreiche Bilder von ein und demselben Lego-Stein aus unterschiedlichen Perspektiven und gegebenenfalls Lichtverhältnissen erstellen

- *Lernen*
  Das mathematische Modell trainieren, also Koeffizienten/Gewichte ermitteln

- *Anwenden*
  Das erlernte Modell auf neue Daten anwenden

Für die Gewinnung von Trainingsdaten wurden je Lego-Stein ein Video erstellt und anschließend mit dem Kommandozeilen-Tool „ffmpeg" Bilddateien erzeugt; rund fünfzig Bilder je Lego-Stein *(siehe Abbildung 4)*. Die Pixel der jeweiligen Bilder stellen die Eingabedaten dar und der Ordnername repräsentiert das jeweilige zugehörige Ergebnis.

Welche Funktion beziehungsweise welches Machine-Learning-Modell ist nun das geeignete für diese Problemstellung? Sollten wir von Grund auf ein neuronales Netzwerk entwickeln? Ein enormer Aufwand!

Über die Jahre wurden bei Google („Inception v3") und Microsoft („ResNet") Netzwerke entwickelt, die in der Lage sind, Bildmotive zu erkennen. Das Interessante ist, dass die Netze vortrainiert eingesetzt werden können, die Koeffizienten (auch Gewichte genannt) also auf Basis einer riesigen Bild-Datenbank (ImageNet [5] mit mehr als 15 Millionen klassifizierten Bildern) optimiert sind. Das Netz kann demnach bereits Objekte erkennen. Es muss nur noch durch Ergänzung einer weiteren Ebene modelliert werden, welches Objekt welcher Kategorie zugeordnet werden soll.

Aufgrund der weiten Verbreitung fiel die Entscheidung auf TensorFlow [6], ein Open-Source-Machine-Learning-Framework, das bereits vordefinierte Trainings- und Evaluierungsskripte für das Inception-Netz mitbringt. Der in *Abbildung 5* dargestellte Kommandozeilen-Aufruf gibt an, dass der Trainingsfortschritt nach 250 Iterationen beendet werden soll. Die klassifizierten Trainingsdaten werden aus dem Ordner „training-data" gelesen. Das Ergebnis des Trainingslaufes ist ein Berechnungsgraph („retrained_graph.pb"), der die gelernte Logik (also Koeffizienten im Netz) enthält. Aufgrund des vortrainierten Netzes und der geringen Anzahl von Trainingsdaten dauert das Training weniger als eine Minute auf einer aktuellen CPU.

Nach dem Training gilt es, das gelernte Netz auf neue Bilder anzuwenden und zu prüfen, ob und wie gut es funktioniert. Dazu nimmt man Bilder von Lego-Steinen, die nicht in den Trainingsbildern enthalten sind. Die Anwendung geschieht in unserem Beispiel durch den einfachen Aufruf von „python label_image.py neu/eval2x3.png". Als Ergebnis erhält man eine Einschätzung, welcher Lego-Stein mit welcher Wahrscheinlichkeit erkannt wurde: „brick2x3 (score = 0.72559) brick2x2 (score = 0.15992) brick1x2 (score = 0.07629) brick2x4 (score = 0.03821)". Im Beispiel erkennt das gelernte Netz also mit einer Wahrscheinlichkeit von rund 72 Prozent den Lego-Stein. Das ist nicht besonders hoch, aber möglicherweise auf die geringe Anzahl von Trainingsdaten zurückzuführen. In jedem Fall wird mit mehr als 50 Prozent auf den korrekten Stein getippt. Bei Tests mit Bildern von Lego-Steinen, die keiner Kategorie der Trainingsdaten angehören, lag der beste Tipp übrigens stets unter 50 Prozent.

Die Implementierung des skizzierten Lego-Anwendungsfalls inklusive Trainings- und Evaluierungsbildern steht auf GitHub [3] für den Selbstversuch beziehungsweise zur weiteren Optimierung zur Verfügung. *Abbildung 6* zeigt nochmals das grundlegende Vorgehen bei überwachtem Machine Learning. Aus klassifizierten Daten (Eingabe inklusive korrekten Ergebnisses) wird ein geeignetes Modell/Netz erstellt und anschließend in mehreren Iterationen trainiert. Anschließend kann es nach klassischem Programmier-Paradigma als Funktionslogik verwendet werden.

KI/ML-Werkzeuge und -Plattformen

Die Disziplin des maschinellen Lernens ist nicht neu. In den letzten fünfzehn Jahren sind Bibliotheken, Frameworks und Platform-Services gereift. Eines der mächtigsten Toolkits ist „scikit learn" – ein Open-Source-

Produkt mit zahlreichen Algorithmen und Beispielen. Zielgruppe sind Data-Mining- und Data-Science-Spezialisten, die sich im Detail auskennen.

Der aktuelle Trend geht dahin, diese Werkzeuge über den Browser zugänglich zu machen, sodass eine Installation auf lokalen Rechnern entfallen kann. Wer also nicht tief ins Detail zu gehen braucht, kann auf sogenannte „KI/ML-Plattform-Services" in der Cloud zugreifen. Alle großen Cloud-Provider bieten Services zur Nutzung von Machine Learning.

Für jemanden, der bestehende intelligente Services nutzen will, ohne im Detail wissen zu müssen, mit welcher Technik (also ML) sich die Services Intelligenz aneignen, gibt es immer mehr out of the box nutzbare Angebote, insbesondere zu Bild- beziehungsweise Objekt- und Emotions-Erkennung, Text-/Sprach-/Sprecher-Erkennung sowie Übersetzung.

Abbildung 7 zeigt die bekanntesten KI/ML-Werkzeuge. Besonders hervorzuheben ist Keras, ein übergreifendes Framework, das in der Lage ist, Machine-Learning-Modelle auf einem hohen Abstraktionslevel zu beschreiben und per Konfiguration auf Basis von TensorFlow, Theano oder CNTK ausführen zu lassen. Damit lässt sich implizit auch sehr einfach die Leistungsstärke der jeweiligen Toolkits vergleichen. Je nach Anwendungsfall kann das eine oder andere Tool besser abschneiden – sei es in der Genauigkeit oder Trainingsgeschwindigkeit.

Fazit

Maschinelles Lernen hat den akademischen Ruf hinter sich gelassen. Nach jahrelangen wissenschaftlichen Vorarbeiten steht es einem breiten (IT-)Anwenderkreis zur Verfügung. Zahlreiche Open-Source-Werkzeuge ermöglichen eine einfache Anwendung und Handhabung von Machine Learning auf bekannte Anwendungsfälle.Für neue Anwendungsfälle, für die es noch keine Modelle und (vortrainierte) Netze gibt, erfordert die Nutzung von Machine Learning viel Kreativität, Trial & Error und Ausdauer. Die Fehlersuche/-behebung (Debugging) ist extrem schwierig. Liegt der Fehler im Design des Modells, an den Trainingsdaten oder an nicht optimalen Parametern? An welcher Stellschraube müsste im Lego-Beispiel gedreht werden, um auf eine Genauigkeit von mehr als 90 Prozent zu kommen? Liegt es wirklich nur an unzureichenden Trainingsdaten?

Eine besondere Herausforderung bei Anwendung von Machine Learning stellen grundlegende Änderungen (wie Gesetze, Vertragsbedingungen, Diagnose-Merkmale) dar. Denn das bedeutet, dass gegebenenfalls optimale, über mehrere Jahre gültige Musterdaten, die als Basis für das Lernen eines Modells genutzt wurden, nicht mehr korrekt sind und damit als Trainingsdaten nicht die erforderliche Qualität aufweisen. Ab dem Zeitpunkt einer solch gravierenden Änderung sollte das bestehende Modell nicht mehr genutzt werden, da es mit falschen Annahmen trainiert wurde. Dann gilt es, neue Trainingsdaten zu gewinnen, um das Modell Stück für Stück wieder an die ursprüngliche Leistungsfähigkeit zu bringen. In manchen Fällen könnte es möglich sein, das Modell zu erweitern oder die bisherigen Trainingsdaten künstlich um die neu geltenden Merkmale anzupassen. Mit vielen aussagekräftigen Trainingsdaten lassen sich auf Basis von überwachtem Maschinenlernen interessante Aufgabenstellungen im Unternehmenskontext lösen:

- Dokumenten-Analyse und -Klassifizierung (etwa Verträge, Angebote prüfen, Spam erkennen)
- Healthcare – Erkennung von Krankheitsbildern in Diagnosedaten/-aufnahmen
- Finance – Prüfung der Kreditwürdigkeit, Empfehlung von Investment-Strategien
- eCommerce – Optimierung der Conversion-Rate
- Smart City – Intelligente Verkehrssteuerung
- Produktion – Erkennung von Ereignismustern
- Support – Erkennen von Verhaltensmustern (genervter Kunde etc.)

Als Ausblick sei erwähnt, dass aktuelle Entwicklungen in Richtung „automatisiertes Machine Learning" (AutoML) gehen. Das bedeutet im Wesentlichen, dass das Experimentieren mit neuen Modellen und Algorithmen automatisiert erfolgt. Was heute ein Data-Science-Experte durchführt, könnte in Zukunft voll automatisiert erfolgen. Darüber hinaus arbeiten Hersteller an Machine-Learning-Bibliotheken, die auf mobilen/embedded Geräten ablaufen. Damit werden insbesondere folgende Ziele verfolgt: Sicherheit, Offline-Fähigkeit und Performance – die Daten müssen nicht das Gerät (etwa in eine Cloud-Umgebung) verlassen. Zusammenfassend lässt sich folgendes Fazit ziehen:

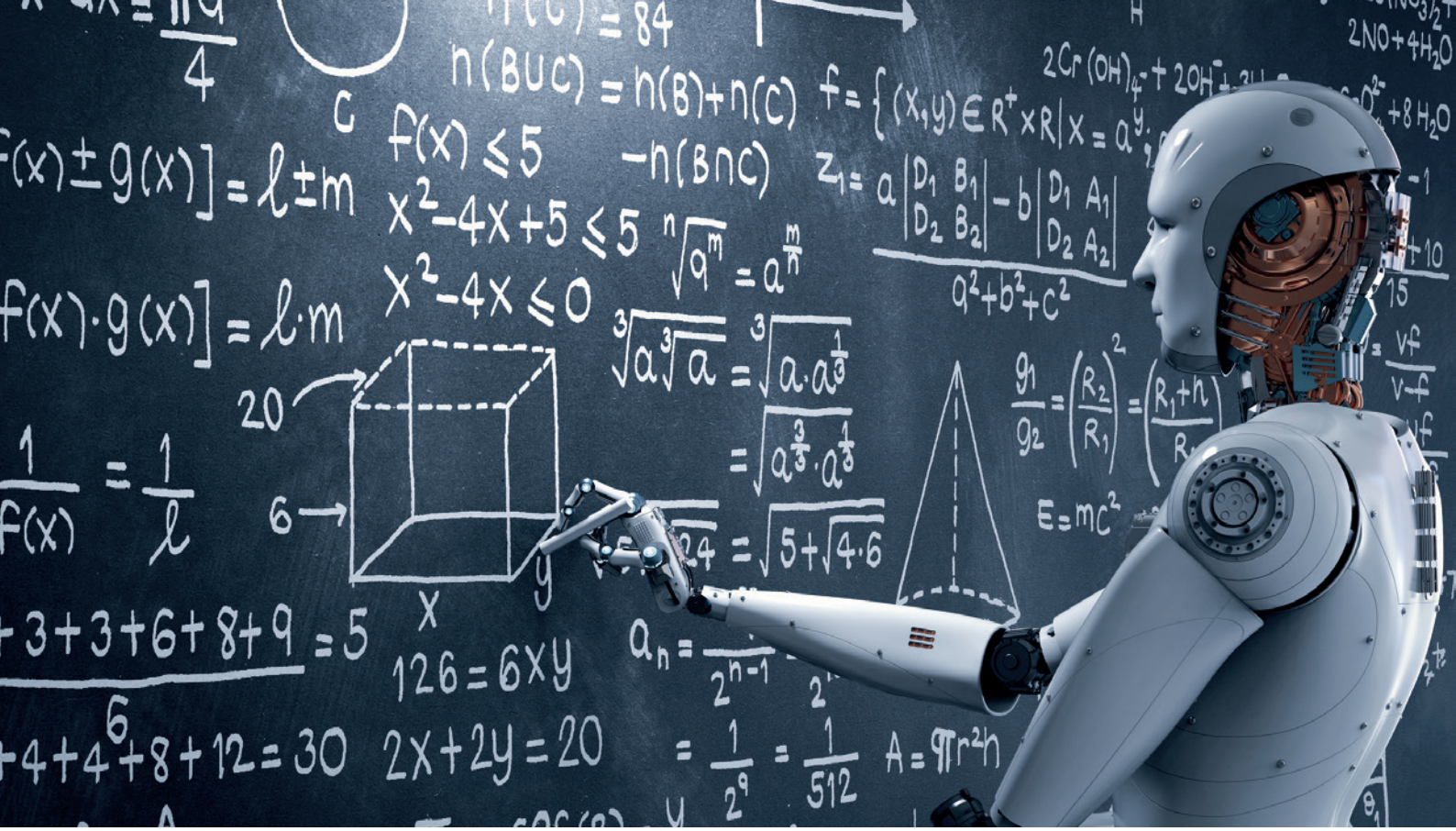
- Maschinelles Lernen lässt Big Data erstrahlen
- Auf Basis vortrainierter Modelle sind schnelle Trainingserfolge möglich
- GPUs oder spezielle Prozessoren wie Tensor Processing Units (TPU) sind für effizientes Lernen erforderlich
- APIs für künstliche Intelligenz nehmen zu und sind per Cloud-Service einfach nutzbar – insbesondere für Bilderkennung, Spracherkennung/-verarbeitung zur Implementierung von Bots und virtuellen Assistenten

Maschinelle Lernverfahren sind ein maßgeblicher Enabler, um intelligente Systeme zu bauen. Es zeichnet sich ab, dass Produkte ohne integrierte KI schon bald der Vergangenheit angehören. Die zunehmende Automatisierung impliziert eine kontinuierliche Selbstoptimierung.

Weitere Informationen

[1] Big Data: *https://de.wikipedia.org/wiki/Big_Data*
[2] Deep Learning with Python Version 4
[3] Code der Lego-Stein-Erkennung: *https://github.com/enpit/tensorflow-for-lego*
[4] Google Cloud TPU: *https://cloud.google.com/tpu*
[5] ImageNet: *http://www.image-net.org*
[6] TensorFlow – Open Source Library for Machine Intelligence: *https://www.tensorflow.org*

Andreas Koop
andreas.koop@enpit.de



Machine Learning zur Zusammenführung heterogener historischer Daten und neuer Datenbestände – ein Anwendungsfall aus der Materialstammdaten-Harmonisierung

Dr. Sebastian Appelhans, thyssenkrupp AG, und Dr. Sebastian Wernicke, ONE LOGIC GmbH

Die thyssenkrupp AG betreibt zahlreiche ERP-Systeme verteilt über mehrere Gesellschaften. Die gleiche Materialart kann in diesen Systemen unterschiedliche Material-IDs besitzen, obwohl sie technisch identisch oder sehr ähnlich ist. Diese Herausforderung lässt sich mit modernen Algorithmen lösen. Dazu hat die thyssenkrupp AG zusammen mit der ONE LOGIC GmbH eine zentrale Data-Science- Plattform für den Konzern geschaffen. Diese kann nun für zahlreiche Anwendungsfälle verwendet werden. Oracle-Big-Data-Technologien sind die Grundlage – im eigenen Rechenzentrum und in der Cloud.

Die thyssenkrupp AG ist ein diversifizierter Industriekonzern mit traditionell hoher Werkstoffkompetenz und einem wachsenden Anteil an Industriegüter- und Dienstleistungsgeschäften mit Hauptsitz in Essen. Weltweit arbeiten über 156.000 Mitarbeiter an ca. 2.000 Standorten in 78 Ländern. Die Geschäftsaktivitäten sind in sechs Business Areas gebündelt: Components Technology, Elevator Technology, Industrial Solutions, Materials Services, Steel Europe und Steel Americas. Die thyssenkrupp AG erwirtschaftete im Geschäftsjahr 2015/2016 einen Umsatz von rund 39 Mrd. Euro (siehe Geschäftsbericht 2015/16).

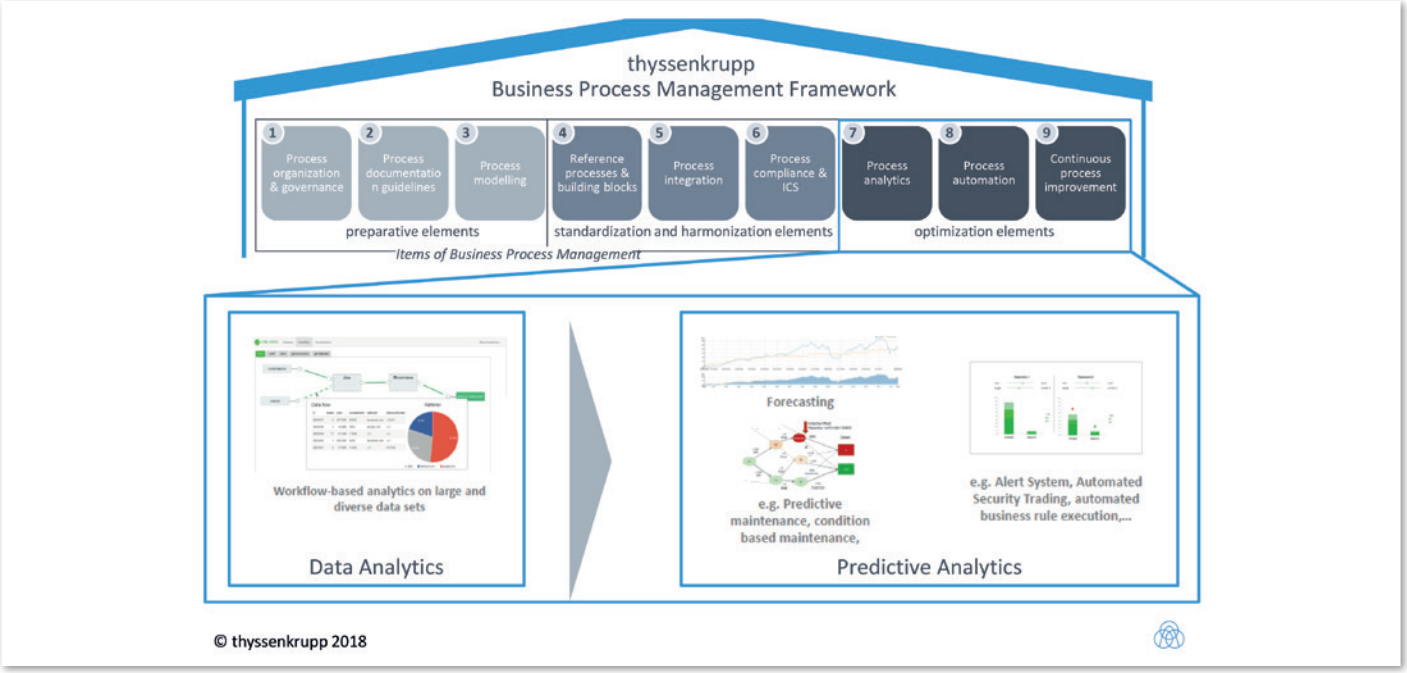


Abbildung 1: Das thyssenkrupp-Business-Process-Management-Framework umfasst die Integration der Data und Predictive Analytics in das Framework

Die ONE LOGIC GmbH ist ein junges, auf Data Science spezialisiertes Unternehmen mit über 70 Mitarbeitern an den Standorten Passau und München. Das Unternehmen entwickelt Lösungen für Kunden in den Bereichen Predictive Analytics, Machine Learning und künstliche Intelligenz. Mit der selbst entwickelten Data-Science-Plattform ONE DATA wird der effiziente Übergang von Prototypen in die Produktivumgebung ermöglicht.

Daten- und Prozess-Harmonisierungsinitiative

Wie bei vielen großen Unternehmen befinden sich bei thyssenkrupp viele Daten in verteilten, teils heterogenen Systemen. Für die übergreifende Analyse oder auch Prognose (Predictive Analytics) muss zunächst einmal eine gemeinsame Datenbasis geschaffen werden. Das Unternehmen zielt mit seiner zentralen Harmonisierungsinitiative „daproh" (data- and process harmonization) auf die Daten-und Prozess-Harmonisierung im gesamten Konzern. Die Initiative ist nicht als reines IT-Projekt angelegt, sondern umfasst neben der Optimierung der Geschäftsprozesse den ganzheitlichen unternehmensweiten Wandel des Mindset zu einer agilen und effektiven Netzwerkorganisation *(siehe Abbildung 1)*. Das Vorhaben der Initiative verfolgt in erster Linie eine Standardisierung von Daten, um folgende vier Ziele zu erreichen:

- Unterstützung für die operative Steuerung
- Erhöhung der Transparenz
- Verbesserung und Automatisierung von Geschäftsprozessen
- Beitrag zur Sicherstellung der Compliance

Für die Harmonisierung wurde ein zentrales Business Process Management Framework mit neun Phasen definiert. Diese setzen sich zusammen aus:

- *Drei vorbereitenden Elementen*
  Prozessorganisation und Einhaltung der Prozesse (1), Prozessdokumentationsrichtlinien (2) und der Prozessmodellierung (3)

- *Drei Standardisierungs- und Harmonisierungselementen*
  Referenzprozesse und Bausteine (4), Prozess-Integration (5), Prozess Compliance & (ICS) (6)

- *Drei Optimierungselementen*
  Prozessanalytik (7), Prozessautomatisierung (8) und kontinuierliche Prozessverbesserung (9)

Data und Predictive Analytics können nun auf diesem Weg in die Geschäftsprozess-Management-Struktur integriert werden, insbesondere in die Schritte Prozessautomatisierung und Prozessverbesserung.

Da End-to-End-Prozesse aufgrund komplexer Rollen- und Zuständigkeitsprobleme schwerer zu optimieren sind, ist thyssenkrupp dazu übergegangen, diese Prozesse in sogenannte „Building Blocks" nach fachlichen Kriterien zu segmentieren. So sind unter anderem Planning & Estimation Accounting oder Overhead Cost Management *(siehe Abbildung 2)* in fachlich abgegrenzte Referenzprozesse (wie Controlling oder Accounting) zusammengefasst. Gemeinsam mit den jeweiligen Fachabteilungen definiert das Business Process Management hier zentrale Angebote und allgemeingültige Geschäftsprozesse, die zusammen mit zentralen Applikationslösungen im Rahmen der Building Blocks zur Implementierung in ihre Businessmodelle an die Konzerngesellschaften ausgeliefert werden.

Um die Harmonisierungsinitiative auch in Zukunft weiterentwickeln und die Effizienz und Einhaltung von Prozessen und damit auch ihre Wirtschaftlichkeit messen zu können, werden Data-Analytics-Methoden eingesetzt. Hierdurch sollen die klassischen Business-Process-Management-Aufgaben, insbesondere Governance und Continuous Improvement, effizient unterstützt werden.

Alle Daten und Prozesse zu harmonisieren und zu optimieren, soll in Zukunft das Zielbild werden. Da dies jedoch in der Vergangenheit noch nicht der Fall war, muss die Harmonisierungsinitiative auch in der Lage sein, heterogene historische Daten zu vertretbaren Kosten zu integrieren. Dies ist eine Herausforderung und hier kommt die Zusammenarbeit mit ONE LOGIC ins Spiel.

Deren Data-Science-Plattform ONE DATA ist darauf spezialisiert, heterogene Datenquellen zu integrieren, auf Basis von mo-

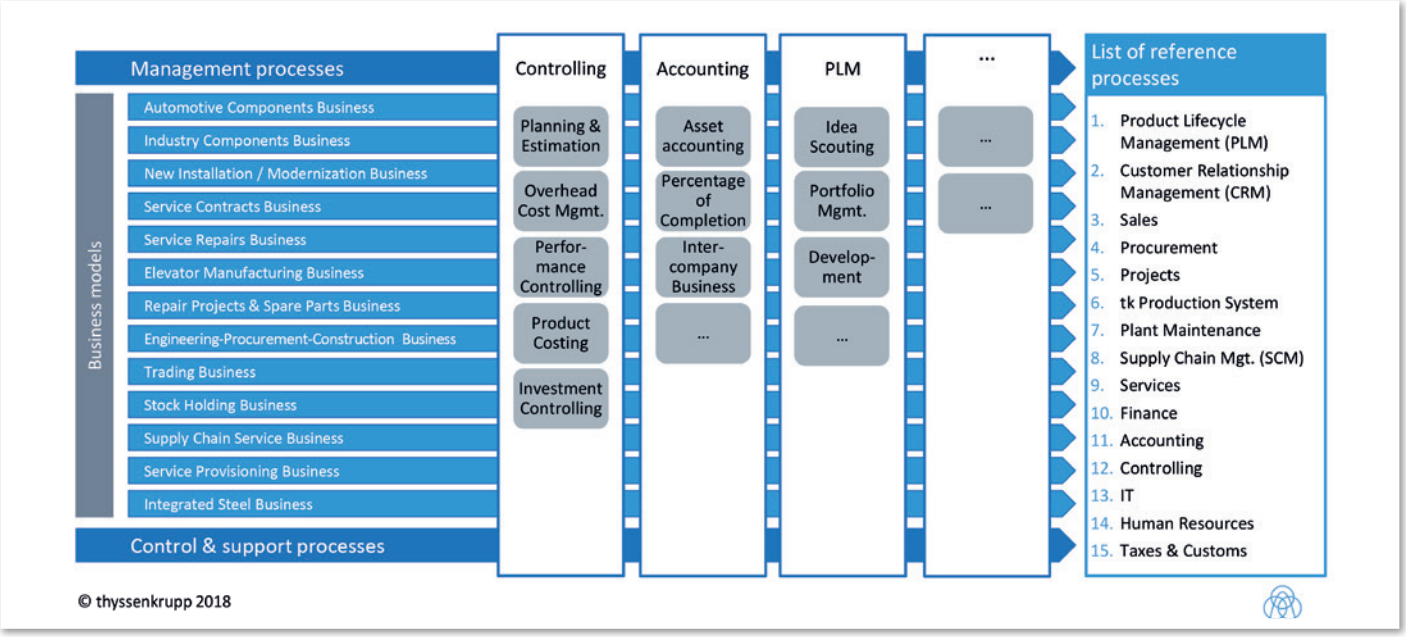


Abbildung 2: thyssenkrupp-Referenzprozesse

dernen Algorithmen Vorhersagen zu treffen und diese im produktiven Einsatz zu nutzen (Data Science „from garage to production"). Die Daten werden dabei modular im Tool aufbereitet und analysiert. Reports und analytische Workflows, deren vielfältige Visualisierung auf aktuellen Web-Technologien basieren, können durch Drag-and-Drop-Technik intuitiv erzeugt werden. Zusammengefasst sind die Eigenschaften:

- Plattform- und Datenbank-unabhängige Software
- Open-Source-basierte Architektur für Cluster-Computing, basierend auf Apache Spark (Verteilung von Rechenoperationen), und damit Skalierbarkeit
- Verteilte Dateispeicher und Datentransfer-Architektur basierend auf Hadoop
- Web-/Browser-basiert, unabhängig vom Gerät
- Reporting und Berechnungen in einem Tool möglich
- Umfangreiche Historie und Revisionssicherheit
- Flexible Grafik-Engine

Es können somit effizient unterschiedlichste Datenquellen angebunden und große Datenmengen einfach verarbeitet werden. Die integrierten Datenanalyse-Werkzeuge und -Algorithmen (wie Frequent-Pattern-Mining-Growth-Algorithmus, Frequent Sequence Mining, Entscheidungsbäume, neuronale Netze oder Support Vector Machines) erlauben auch komplexe Auswertungen direkt im Tool, die im Anschluss mit flexiblen Reporting- und Analyse-Funktionen als statische Reports oder dynamische Dashboards dargestellt werden.

Das Ziel der Zusammenarbeit ist es, historische Datenbestände, heterogene Formate und Plattformen miteinander zu verbinden. Dafür wird eine Datenbank-unabhängige Software benötigt. Zur Überprüfung, ob diese Vorgabe erreicht werden kann, wurde ein Proof of Concept (PoC) im Bereich Materialstammdaten-Harmonisierung durchgeführt *(siehe Abbildung 3)*.

Die Ausgangslage

thyssenkrupp betreibt zahlreiche ERP-Systeme, verteilt über mehrere Gesellschaften. Die gleiche Materialart kann in diesen Systemen unterschiedliche Material-IDs besitzen, obwohl sie technisch identisch oder sehr ähnlich ist. Dies führt beispielsweise zu unnötig höheren Lagerbeständen. Eine erhöhte Transparenz in den Materialbeständen und -flüssen kann zu folgenden Aktionen führen und bietet die Option, neue Business-Modelle zu definieren:

- Signifikante Reduktion des Working Capital
- Höhere Produktverfügbarkeit
- Höhere Liefergeschwindigkeit

Die Machbarkeitsstudie bestand aus drei Anforderungspunkten, die erfüllt werden sollten: der erfolgreichen Datenintegration verschiedener Systeme, dem Beleg für Tracking- und Reporting-Möglichkeit sowie einem intuitiven User-Interface mit einer modernen Visualisierung. Im Rahmen des Proof of Concept hat ONE LOGIC zusammen mit Oracle und thyssenkrupp untersucht, ob sich heterogene Material-IDs aus der Vergangenheit mithilfe von Machine-Learning-Algorithmen zuordnen lassen. Klassische Master-Data-Management-Ansätze werden durch statistische Vorhersagen ersetzt, da die manuelle Zuordnung der Daten aufgrund der hohen Komplexität (insbesondere der unterschiedlichen Standards und Kategorisierungen) einen enormen Aufwand bedeuten würde. Es wird eine Datenbank-unabhängige und vor allem von den Formaten unabhängige Lösung benötigt. Dies hat anderen Anbietern in der Vergangenheit Schwierigkeiten bereitet.

Die ausgewählte Lösung kann alle wichtigen Datenquellen – sowohl heterogene Daten aus der Vergangenheit (Legacy Welt) als auch neue Daten (daproh) – einbinden. Vergangenheit und Zukunft können so miteinander verbunden werden. Dies erlaubt thyssenkrupp ein Maximum an Effizienz und Flexibilität. Der erste Anforderungspunkt konnte erfüllt werden.

Die zeitnahe, flexible Ermittlung und Darstellung klassischer Kennzahlen-Reportings sowie deren Verfügbarkeit für beliebige Endnutzer waren ein weiterer Anforderungspunkt. Zudem bietet das Tool die Möglichkeit, dynamische Kennzahlen (wie Durchlaufzeiten) zu erstellen, ohne das Tool zu verlassen, und im gleichen Report per Klick auf eine weitere Detailebene zu gehen („Drill-Down").

Der dritte Anforderungspunkt des Proof of Concept war ein ansprechendes User-Inter-

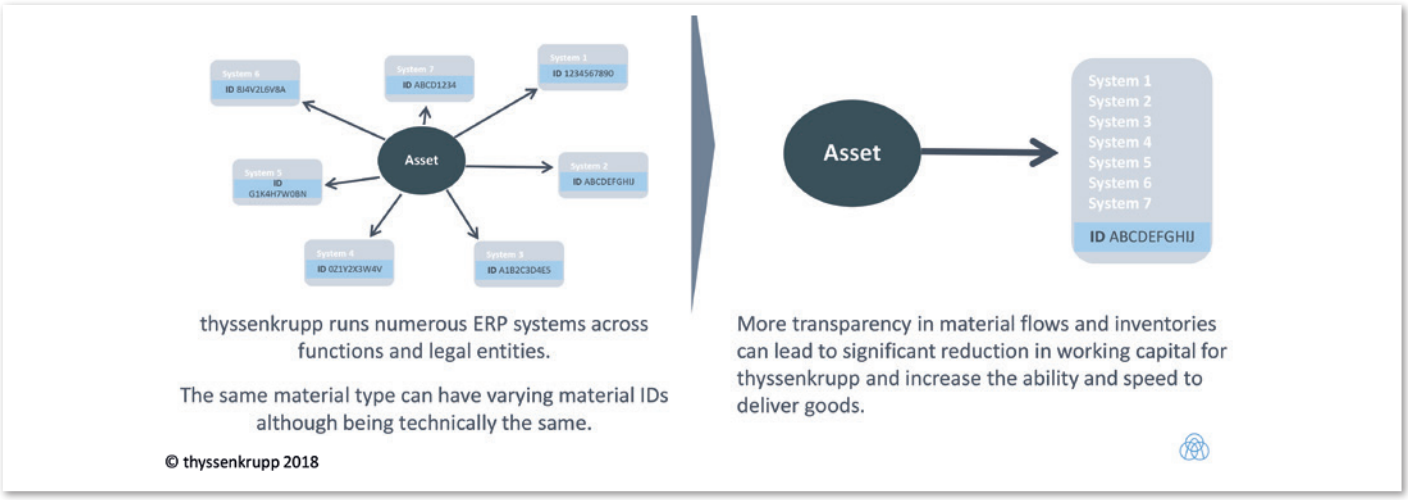


Abbildung 3: Materialstammdaten-Harmonisierung

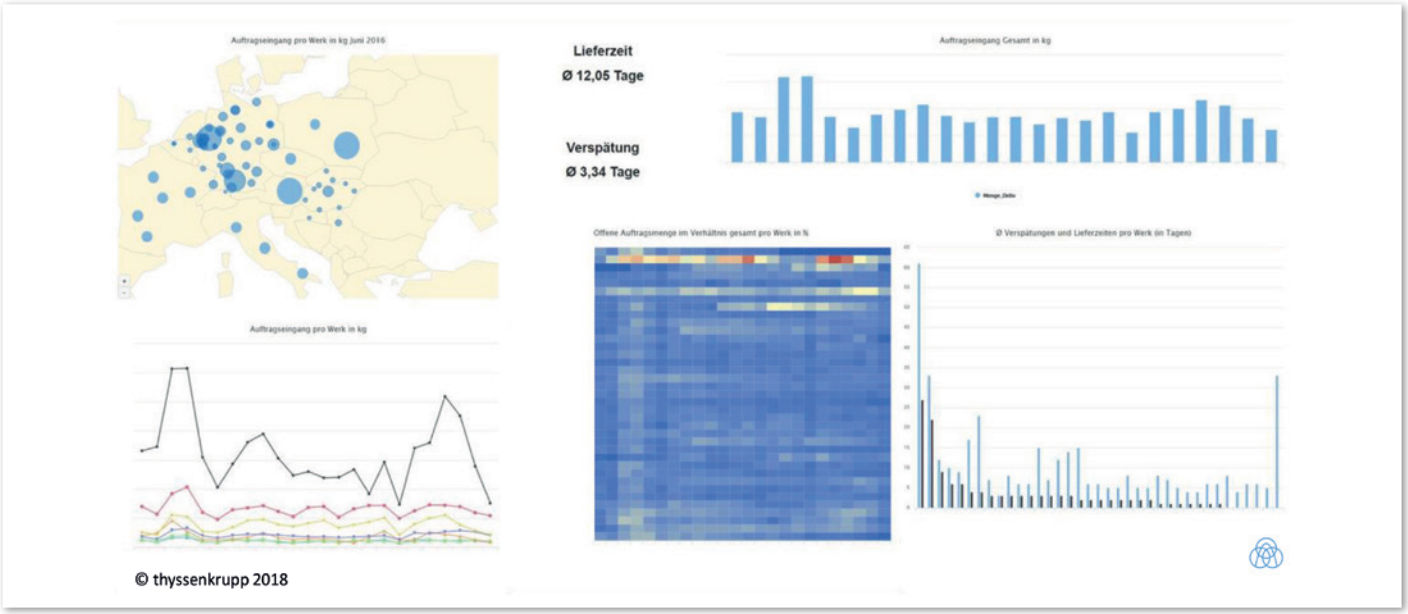


Abbildung 4: Beispiel für ein Reporting-Cockpit

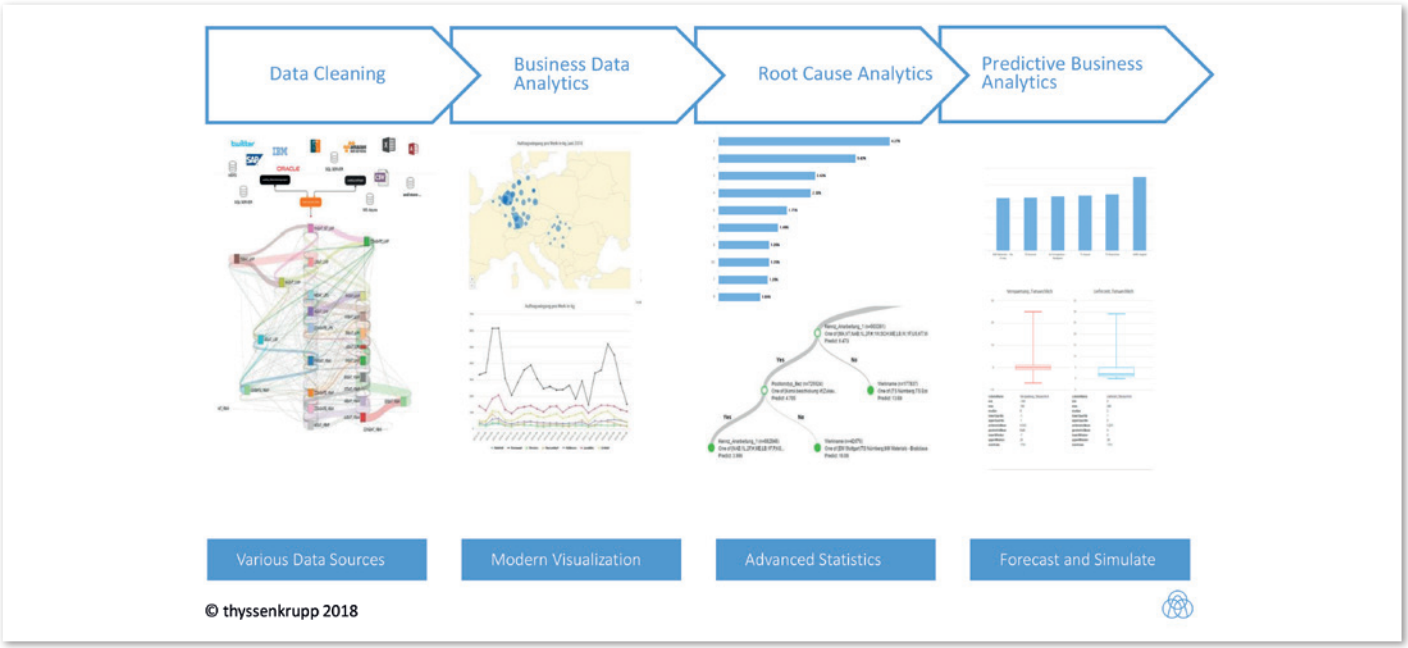


Abbildung 5: Big Data Analytics Services

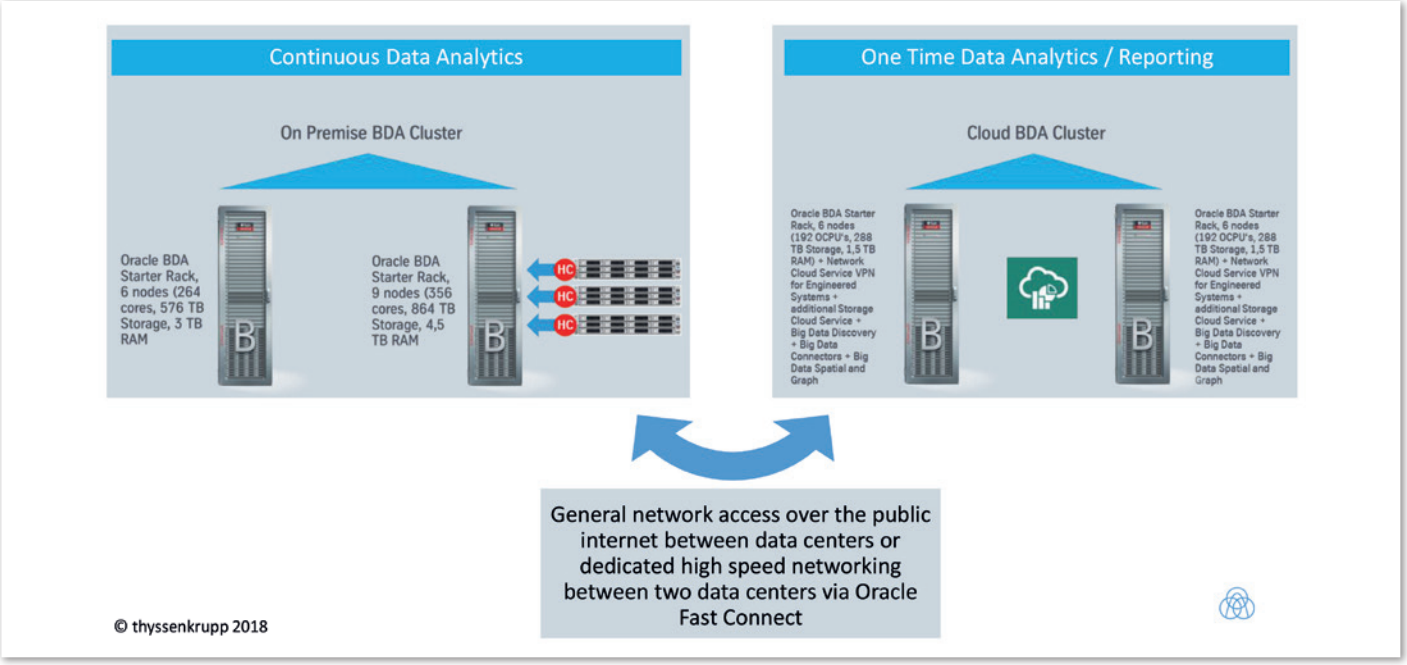


Abbildung 6: Big-Data-Hardware-Infrastruktur

face und moderne Visualisierungsmöglichkeiten basierend auf neuester Technologie. Das Tool soll intuitiv vom User verwendet und verstanden werden können (kein Spezialisten-Tool). Die Visualisierungsmöglichkeiten sollen dynamisch zusammenstellbar sein (Cockpit) und die Möglichkeit zur schnellen Drag-and-Drop-Erstellung von Berichten bieten. Da auch dieser Punkt abgedeckt wird, wurde der Proof of Concept erfolgreich abgeschlossen *(siehe Abbildungen 4 und 5)*.

Das ursprüngliche Ziel der Machbarkeitsstudie war es, Business Data Analytics aufzusetzen. Im Zuge dessen wurde jedoch festgestellt, dass man auf dieser Basis auch weitergehen kann. Wenn Probleme erkannt werden, kann Ursachen-Analytik (Root Cause Analysis) angewendet werden. Die Ursachen für Ergebnisse werden dabei durch Kausalanalysen wie Entscheidungsbäume festgestellt und durch Simulation veranschaulicht. Daraufhin kann auch weiter in den Bereich „Vorausschauende Analytik" (Predictive Analytics) übergegangen werden. Anhand der Ergebnisse aus den Analysen, wenn beispielsweise vorher definierte Leistungskennzahlen (KPIs) nicht erreicht werden, können weitere Fragestellungen und Kausal-Analysen abgeleitet werden.

Ein klassischer Anwendungsfall wäre etwa, wenn eine schlechte Charge (Produktionseinheit) entdeckt würde. Mit der weiterführenden Analyse kann die Frage „Wo wird das in Zukunft wieder auftreten?" bearbeitet werden, ohne dass das Tool beziehungsweise die Plattform gewechselt werden müsste. Dies ist für große Unternehmen wie thyssenkrupp allein aus Sicherheitsgründen ein wichtiger Aspekt. Neben der erfolgreich abgeschlossenen Machbarkeitsstudie in Bezug auf Material-IDs im Themenfeld „Supply Chain Management" gibt es viele weitere Anwendungsfälle, die aufgrund der Flexibilität der Plattform durchgeführt werden können. Beispiele wären die Analyse eines Kundenbeziehungsmanagements (CRM) oder die eines Produkt-Lebenszyklus (PLM). Im ersten Fall können zum Beispiel Kunden über Regionen, Geschäftsfelder und Zeiträume mit abgebildeten Kundennummern analysiert werden, im zweiten können das bestehende Produktdaten-Management-System verbessert und die abgebildeten Produktnummern bereinigt werden.

Oracle Big Data Appliance als Grundlage

Die Grundlage für die zentrale Data-Science-Plattform sind Oracle-Big-Data-Technologien *(siehe Abbildung 6)*. Das System besteht aus einer hybriden Architektur von Big Data Appliances sowohl im eigenen Rechenzentrum (On-Premise) als auch in der Cloud. Anforderungen wie Datensicherheit und Datenhoheit auf der einen Seite und flexible Skalier- und Erweiterbarkeit auf der anderen Seite sind durch dieses Konzept sichergestellt. Die Verfügbarkeit der On-Premise-Lösung ist unter anderem wichtig für die Performance und den Datentransfer, die Geschwindigkeit, aber auch unter Sicherheitsaspekten. So kommunizieren die BDAs im eigenen Rechenzentrum künftig direkt mit Oracle-Exadata-Systemen, auf denen zentrale Datenbanken laufen. Ein weiterer Vorteil dieser Architektur ist ihre einfache Erweiterbarkeit. Die Verfügbarkeit in der Cloud hingegen ist interessant für kleinere Analysen und insbesondere fürs Reporting. Des Weiteren werden Oracle-Technologien zum Komprimieren der Daten oder zur Unterstützung der sicheren Datentransfers eingesetzt.

Fazit

Durch die Machbarkeitsstudie im Bereich Business Data Analytics, die ONE LOGIC mit Oracle bei thyssenkrupp erfolgreich umgesetzt hat, haben sich viele weitere Use Cases aufgetan, die auch im Bereich Predictive Analytics umgesetzt werden können. ONE DATA als Data-Science-Plattform liefert neben einem ansprechenden User Interface den großen Vorteil, auch heterogene historische Daten aus verschiedensten Systemen des Unternehmens in ihre Analysen einbeziehen zu können. Damit unterstützt das Tool die Umsetzung der Daten- und Prozessoptimierungsinitiative daproh von thyssenkrupp, mit der der Konzern seine Geschäftsbereiche dauerhaft für die Anforderungen von Industrie 4.0 /Digitalisierung aufstellen und sich kontinuierlich weiterentwickeln will.

Dr. Sebastian Wernicke
sebastian.wernicke@onelogic.de

Dr. Sebastian Appelhans
sebastian.appelhans@thyssenkrupp.com

Ersetzen Data Lakes die Core DWHs?

Andreas Buckenhofer, Daimler TSS GmbH

Das Ende des Data Warehouse (DWH) oder von ETL wird in immer mehr Artikeln beschrieben. Data Lakes oder Schema-on-read stehen für neue Ansätze, die Flexibilität, Skalierbarkeit und Performance versprechen, um neue Muster in den Daten zu erkennen oder Advanced Analytics anzuwenden. Dieser Artikel stellt Big-Data-Architekturen und -Konzepte vor und vergleicht sie mit dem klassischen DWH.

Die Digitalisierung betrifft alle Branchen; die Software wird immer wichtiger: Robotics, Industrie 4.0, Internet of Things, Connected Cars, um nur einige aktuelle Entwicklungen zu nennen. Moderne Fahrzeuge enthalten heute bereits mehr als hundert Millionen Codezeilen – Tendenz stark steigend. Im Vergleich dazu besteht Facebook aus nur 65 Millionen oder Android aus 15 Millionen Codezeilen. Fahrzeuge werden durch digitale Services angereichert oder neue Services wie das spontane Mieten von Fahrzeugen („car2go“) entstehen. Unternehmen wie Google und Apple dringen immer mehr in traditionelle Branchen ein. Im Mittelpunkt stehen die Daten, die als Asset für datengetriebene Entscheidungen dienen oder selbst das Produkt sind („Data as a Product“). DWHs oder Data Lakes können diese Daten sammeln, speichern, aufbereiten und zur Verfügung stellen.

Das klassische Data Warehouse

Das DWH dient dazu, um Daten aus mehreren Datenquellen zusammenzuführen. Die Daten sind qualitätsgeprüft, integriert und harmonisiert. Bill Inmon und Ralph Kimball

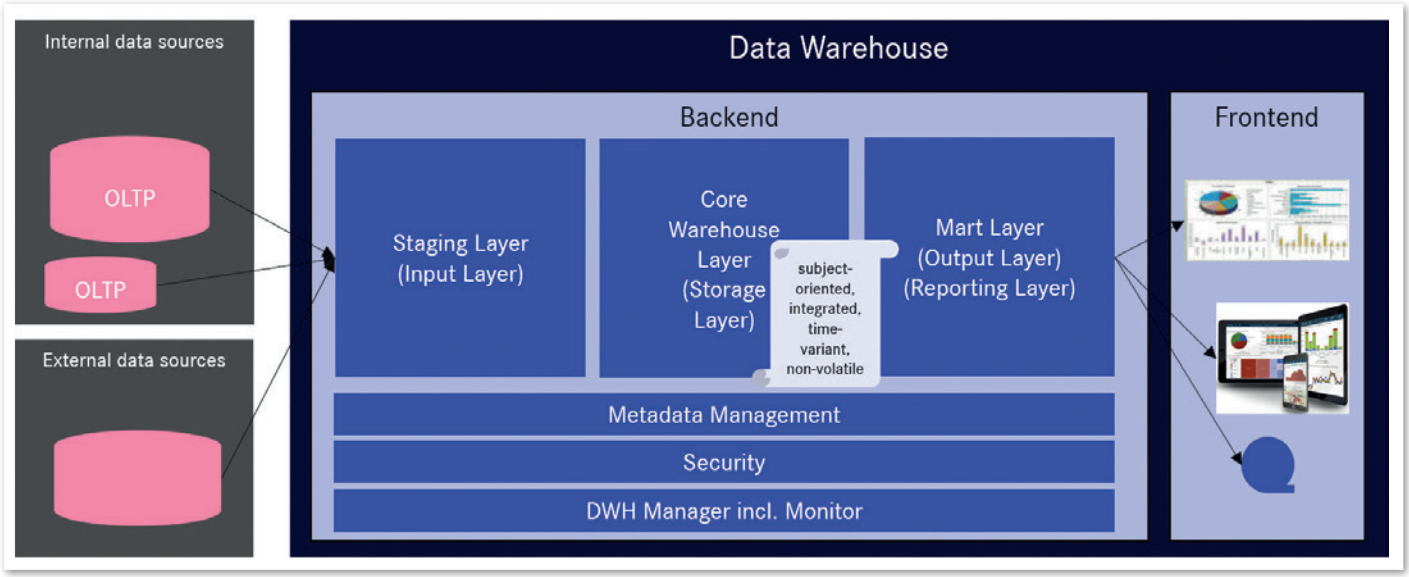


Abbildung 1: Dreischichtige DWH-Architektur

prägten die in der Industrie hauptsächlich anzutreffenden Architekturen. In *Abbildung 1* ist eine klassische dreischichtige Architektur dargestellt. Weitere Architekturen sind in [1] aufgeführt.

Die Datenintegration erfolgt in verschiedenen Layern:

- *Input Layer (Staging Layer, Acquisition Layer)*
  Übernahme der Daten aus den Quellsystemen. Die Daten in diesem Layer sind typischerweise nur temporär gespeichert.

- *Storage Layer (Core Warehouse Layer)*
  Zusammenführung der Daten in ein neues, DWH-spezifisches Datenmodell wie Data Vault, Head-Version-Modell, 3NF inklusive Historie etc. Die Daten sind organisiert nach Themen („Subject Areas") anstatt nach Funktionen. Daten werden nicht gelöscht, sondern stets hinzugefügt, sodass die gesamte Historie verfügbar bleibt.

- *Output Layer (Data Mart Layer, Presentation Layer, Reporting Layer)*
  End-User greifen auf diesen Layer zu, um die Daten in Reports, Dashboards, Scorecards etc. auszuwerten. Das Datenmodell richtet sich am Tool aus, um ein möglichst performantes Lesen der Daten zu ermöglichen. Typischerweise trifft man eine dimensionale Modellierung an, da beispielsweise ein Star-Schema die Daten performant und für den End-User verständlich zur Verfügung stellt.

Das Cleansing der Daten erfolgt üblicherweise vom Input in den Storage Layer, sodass die Daten durch Bereinigungen und Transformationen veredelt zur Verfügung stehen und Widersprüche im Sinne eines „Single Version of Truth" vermieden werden. Architekturen wie Data Vault 2.0 führen das Cleansing der Daten erst vom Storage in den Output Layer durch, um Daten unverändert aus dem Quellsystem abzulegen und Auditierbarkeit sicherzustellen. Mit dem Aufkommen von Data Lakes stieg die Kritik am DWH:

- Zu unflexibel, wenn es um neue Datenformate wie JSON, XML, Audio, Video oder Logs allgemein geht („Data Variety")
- Zu langsam, wenn es um die Verarbeitung schnell eintreffender Daten geht („Data Velocity")
- Zu teuer, wenn es um die Speicherung riesiger Datenmengen geht („Data Volume")
- Zu aufwändig, wenn es um das Cleansing der Daten geht

Begriffs-Wirrwarr: Data Lake, Data Library, Data Swamp, Landing Zone, Hadoop, Spark

Für das klassische DWH bestehen etablierte Methoden und Konzepte. Doch rund um Data Lakes, Hadoop, Data Libraries etc. existieren einzelne Artikel mit jeweils eigenen Definitionen und Begriffen wie Data Library, Data Lake 3.0 oder Data Reservoir.

Insbesondere muss zwischen Data Lakes und Hadoop getrennt werden. Data Lakes beschreiben eine Architektur oder ein Konzept in Analogie zu einem DWH; Hadoop oder Spark sind dagegen Tools analog zu relationalen Datenbanken oder BI Suites. Ein Data Lake kann mithilfe von Hadoop implementiert werden oder auch mit einem anderen Tool wie Spark, Elastic Stack oder einer NoSQL-Datenbank. Hadoop besteht im engeren Sinn aus einem Framework für skalierbare, verteilt arbeitende Software sowie einem geclusterten Dateisystem (HDFS). Im weiteren Sinne umfasst Hadoop einen Zoo von vielen Tools wie zum Beispiel

- Sqoop zur Datenintegration
- HBase zur Speicherung von Daten in einem Wide Column Store (NoSQL)
- Hive zum Zugriff auf Daten in Hadoop mittels SQL
- Oozie zum Scheduling von Jobs

Hadoop ist Open Source, jedoch bieten beispielsweise Cloudera, Hortonworks oder MapR eigene Distributionen an, inklusive Spark-Komponenten. Doch wenn Hadoop oder Spark zum Einsatz kommen, was war/ist dann das Problem?

Der Data Lake

Data Lakes sollen die am Ende des Kapitels über DWH aufgeführten Herausforderungen lösen können. Ein Data Lake dient dazu, Rohdaten in einem zentralen, unternehmensweiten Speicher unverändert abzuspeichern, um die Daten für Analysen verfügbar zu machen. Das Potenzial solch eines Datensees ist groß, jedoch verbergen sich auch Risiken, wenn die Daten ohne jegliche Governance angehäuft werden. Rohdaten abzuspeichern im ursprünglichen Format und die Daten

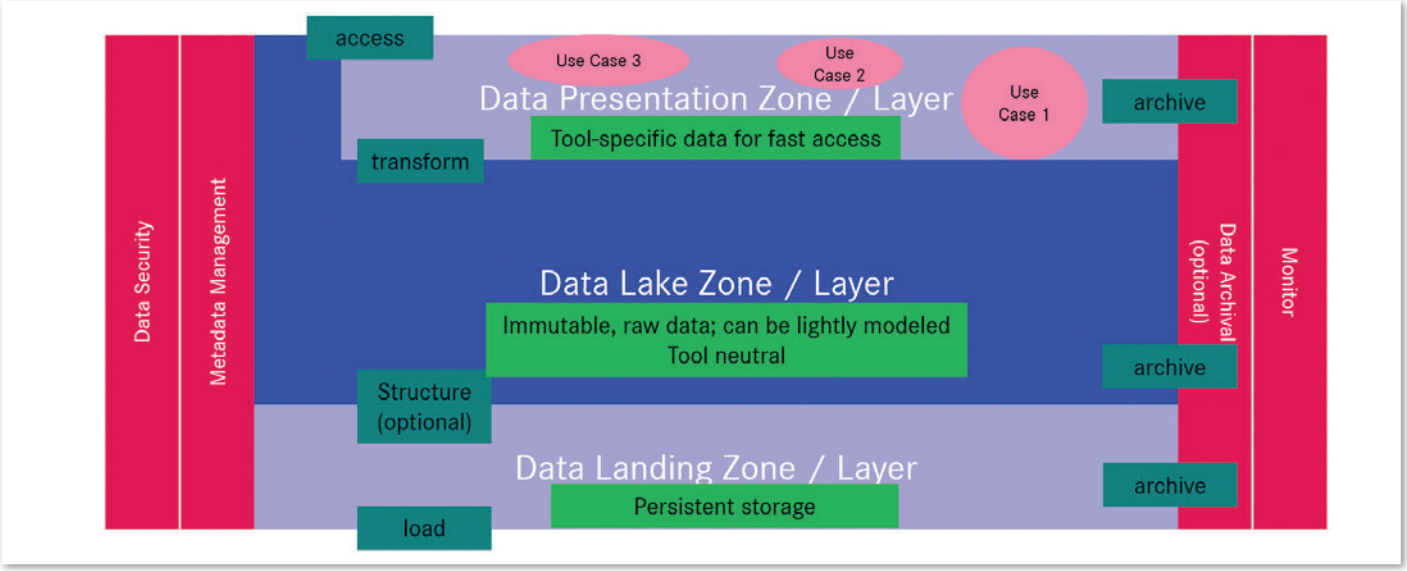


Abbildung 2: Data-Lake-Architektur

erst beim Lesen mit einer Struktur zu versehen („Schema-on-read"), verspricht große Flexibilität, beinhaltet jedoch gleichzeitig das Risiko eines Data Swamp. Sean Martin fasst seine Beobachtungen zusammen: „We see customers creating big data graveyards, dumping everything into HDFS and hoping to do something with it down the road. But then they just lose track of what's there" [2].

Aus Governance-Sicht ist es notwendig, eine Ordnung für einen Data Lake zu definieren. Ein großes, geclustertes Dateisystem wird schnell im Chaos enden. Eine Architektur könnte wie in *Abbildung 2* aussehen.

Data Lake ist dabei in zwei verschiedenen Ausprägungen zu verstehen. Zum einen als übergreifende Data-Lake-Architektur, zum anderen als eine Schicht innerhalb der Architektur. Ein Data Lake lässt sich analog zu einem DWH in verschiedene Schichten oder Zonen einteilen:

- *Data Landing Zone/Layer*
  Übernahme der Daten aus den Quellsystemen. Die Daten werden dauerhaft gespeichert im Sinne einer persistenten Landing Zone. Daten können mehrfach redundant vorliegen, wenn etwa das sendende System die Daten mehrfach übermittelt oder eine Delta-Erkennung nicht unterstützt. Das Dateisystem kann beispielsweise nach Quellsystemen organisiert sein.

- *Data Lake Zone/Layer*
  Speicherung der Daten aus der Landing Zone im Rohformat ohne Datenänderungen und ohne Redundanz. Data Scientists können in dieser Zone ihre explorativen Analysen durchführen. Das Dateisystem kann beispielsweise nach Geschäftsobjekten organisiert sein.

- *Data Presentation Zone/Layer (Data Reservoir)*
  Use Cases werden in dieser Schicht umgesetzt. Die Endanwender bekommen genau die Daten aus dem Data Lake zur Verfügung gestellt, die benötigt werden. Das Dateisystem kann beispielsweise nach Use Cases organisiert sein.

Die größte Flexibilität wird erreicht, wenn die Daten ohne Format-Anpassungen im Data-Lake-Layer gespeichert sind. Erfolgt keine Modellierung der Daten im Gegensatz zu einem DWH, so wird ein Schema erst beim Lesen der Daten angewendet („Schema-on-read").

Doch was bedeutet große Flexibilität bei Schema-on-read genau? Es wird auf eine Datenmodellierung verzichtet, Daten werden also wie angeliefert gespeichert. Diese große Flexibilität hört jedoch beim lesenden Prozess auf, denn der Entwickler des Lese-Prozesses muss nun die Modellierung durchführen und ein Schema anwenden. Viele und vor allem wichtige Daten werden mehrfach gelesen und beispielsweise in eine Presentation Zone überführt. Dies bedeutet, dass das Anwenden des Schemas bei jedem Lesevorgang durchzuführen ist.

Schema-on-read ist inperformanter beim Lesen im Vergleich zu Daten, die in einem modellierten Schema wie Avro oder Parquet vorliegen („Schema-on-write"). Außerdem müssen verschiedene User beziehungsweise Entwickler JSON-Dateien untersuchen und verstehen, um eine Leseoperation zu implementieren. Fehler sind dabei vorprogrammiert.

Nathan Marz, der die Lambda-Architektur geprägt hat, warnt in seinem Buch vor Schema-on-read: „Many developers go down the path of writing their raw data in a schemaless format like JSON. This is appealing because of how easy it is to get started, but this approach quickly leads to problems. Whether due to bugs or misunderstandings between different developers, data corruption inevitably occurs" [3]. Er empfiehlt, Daten in einem Graphen-Modell zu strukturieren.

Alternativ könnten Daten im Data Lake Layer auch mittels Data Vault strukturiert werden, um eine leichtgewichtige Datenintegration durchzuführen und ohne die Daten zu transformieren. Auch Google berichtet von intern gemachten Erfahrungen, dass deren Entwickler ein starkes Schema bevorzugen, um qualitativ hochwertige Software zu liefern [4].

Auch sicherheitsrelevante Anforderungen führen dazu, dass das schemalose Abspeichern von Daten schnell zu Herausforderungen führen kann. Müssen Daten aufgrund von gesetzlichen Bestimmungen gelöscht werden, so ist es sehr hilfreich, wenn die abgespeicherten Daten und deren Struktur bekannt sind.

Use Cases

Log-Daten verschiedener Quellen wie Sensoren oder Roboter liegen häufig als JSON-Daten vor. Verschiedene Softwarestände führen dazu, dass die Geräte Daten in

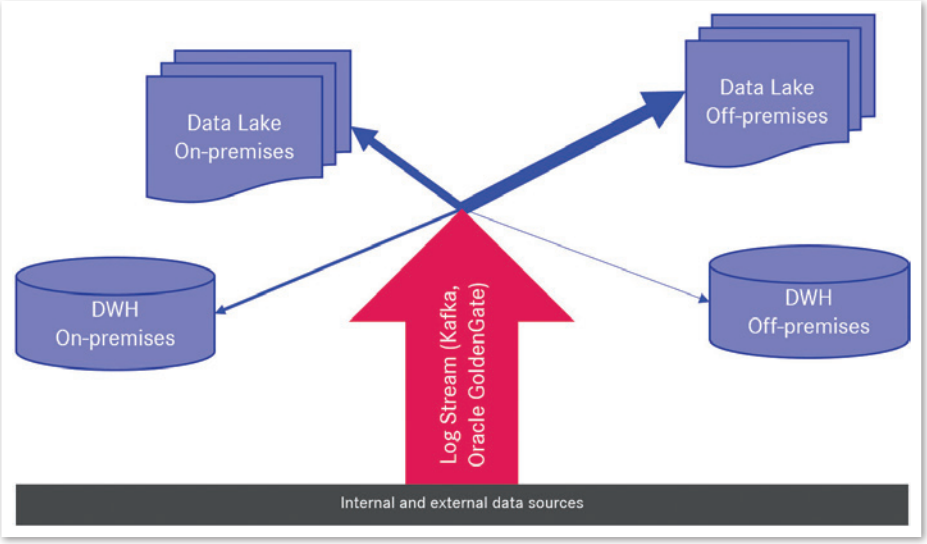


Abbildung 4: Log-zentrierte Architektur

verschiedenen Formaten senden. Durch Messfehler oder Übertragungsfehler sind regelmäßig Messungen fehlerhaft oder unvollständig. Über die Landing Zone werden die Daten in den Data Lake geladen, in dem eine Anreicherung der Maschinendaten mit den Daten zu den bearbeiteten Teilen erfolgen kann. Der Bewegungsablauf und dafür benötigte Zeiten von einzelnen Robotern werden analysiert. Treten Anomalien auf wie etwa überdurchschnittlich lange Stillstandszeiten, so helfen Entscheidungsbäume, eine Erklärung für diese Zeiten zu finden.

In einem anderen Use Case wurden LKW-Stücklisten in den Data Lake integriert. Ziel war die Erkennung von Änderungen im Vergleich zu vorigen Stücklisten-Versionen. Die Stücklisten stammen von einer relationalen Datenbank und das Ergebnis (Änderungen) wird an die relationale Datenbank zurückgeliefert. Zusätzlich werden die Daten im Data Lake persistiert. Für die Erkennung von Änderungen in Stücklisten ist Hadoop sehr gut geeignet, um rechenintensive Stücklisten-Auflösungen durchzuführen und die Teile-Auflösungen abzuspeichern. Dies führt zu einer Entlastung der relationalen Datenbank. Stücklisten können mit Apache Spark GraphX performant abgebildet und analysiert werden.

Ausblick

Data Lakes sind erst am Anfang. Ausgereifte Methoden für die Integration, Speicherung und Analyse von Daten, wie sie von DWHs bekannt sind, sind noch Mangelware. Es gibt noch viel Neuland zu entdecken und Erfahrungen zu sammeln.

Ein einziges zentrales DWH existiert in vielen Unternehmen nicht, da der Aufbau für sehr große, heterogene Unternehmen zu komplex ist und Unternehmen in ständigem Wandel sind, nicht nur durch Zukäufe. Sehr häufig sind mehrere DWHs anzutreffen Auch für Data Lakes kann eine ähnliche Entwicklung erwartet werden, sodass unabhängig voneinander physikalisch getrennte Data Lakes entstehen. Rechtliche Bestimmungen oder auch Agilität tragen zu physikalisch getrennten Data Lakes bei. Ein weiterer Treiber ist der aktuelle Trend zur verstärkten Nutzung von Cloud Services in Ergänzung zu On-Premise Data Lakes.

Hadoop und dessen Zoo verschiedener Tools ist eine Möglichkeit, um einen Data Lake zu realisieren. Die Produktivität und der Reifegrad einzelner Tools sind sehr verschieden. Tools wie Oozie für das Scheduling von Tasks haben aktuell noch eine sehr dürftige Reife und Produktivität. Auch das Error-Handling und die Weitergabe von Fehlern während der Verarbeitung innerhalb von Hadoop sind im Vergleich zu relationalen Technologien noch sehr aufwändig zu realisieren.

Data Lakes lösen einige der im Kapitel zu DWH aufgeführten Herausforderungen – bringen jedoch andere mit. Data Lakes werden DWHs nicht ersetzen, sondern sind eine Ergänzung. Data Lakes und DWHs werden in Zukunft parallel existieren. Die Herausforderung ist, die neue mit der alten Welt sinnvoll zu verbinden, also die Flexibilität und Skalierbarkeit mit den bewährten DWH-Methoden. Schema-on-read hat seine Vorteile, birgt aber auch große Gefahren, die schnell zu einem Data Swamp führen. Eine Datenmodellierung oder Schema-Extraktion ist notwendig; die Frage ist, wann im Prozess und durch wen. Eine mögliche Kombination kann in Anlehnung an die Log-zentrierte Architektur nach Jay Kreps wie in *Abbildung 4* aussehen [5].

Mithilfe von Queuing-Tools wie Kafka werden Daten als kontinuierlicher Log-Stream zur Verfügung gestellt. Alternativ (oder als Ergänzung) kann Oracle Golden Gate verwendet werden, um die Daten aus (relationalen) Vorsystemen abzugreifen. Empfangende Systeme können den Log-Stream komplett oder nur teilweise abgreifen (etwa Maschinen-Logs nur in den Data Lake und nicht in das DWH übertragen), um dann den Input-Layer in einem DWH oder die Landing Zone in einem Data Lake zu versorgen. Das Hosting kann On-, Off-Premise oder auch gemischt erfolgen.

Metadaten-Management ist bereits aus DWH-Projekten bekannt. Ein DWH- und Tool-übergreifendes Metadaten-Management ist in Projekten nur sehr selten anzutreffen und Metadaten sind sehr schnell veraltet, wenn sie nicht gepflegt werden. In Data Lakes wird dieses Thema erneut angegangen – man kann gespannt sein, ob mit mehr Erfolg. Google verfolgt einen anderen Ansatz beim Metadaten-Management: Anstatt die Daten manuell zu pflegen, wird ein interner Crawler verwendet, um interne Systeme nach Metadaten zu durchsuchen und zentral abzulegen [6]. Bei Google geht man davon aus, dass ein manueller Prozess nicht funktioniert.

Quellen und Links

[1] Buckenhofer, Andreas: Introduction to DWH and DWH architecture, Vorlesung an der Dualen Hochschule Baden Württemberg: *https://www.slideshare.net/AndreasBuckenhofer/data-warehousing-lecture-at-bw-cooperative-state-university-dhbw-69300614*

[2] Stein, Brian; Morrison, Alan (2014), Data lakes and the promise of unsiloed data, Technology Forecast, Rethinking integration: *http://www.pwc.com/en_US/us/technology-forecast/2014/cloud-computing/assets/pdf/pwc-technology-forecast-data-lakes.pdf*

[3] Nathan Marz, James Warren, Big Data, Principles and best practices of scalable realtime data systems, Manning Publications 2015

[4] David F. Bacon, Nathan Bales, Spanner, Becoming a SQL System, Proc. SIGMOD 2017, pp. 331-343 (to appear): *https://research.google.com/pubs/pub46103.html*

[5] Jay Kreps, Questioning the Lambda Architecture, 2014: *https://www.oreilly.com/ideas/questioning-the-lambda-architecture*

[6] Alon Halevy, Flip Korn, Goods, Organizing Google's Datasets, Google Research Paper: *https://static.googleusercontent.com/media/research.google.com/de//pubs/archive/45390.pdf*

Andreas Buckenhofer
andreas.buckenhofer@Daimler.com

Business-Intelligence-Cloud-Angebote als Ersatz oder Ergänzung klassischer On-Premises-Data-Warehouses

Jan Schreiber, Loopback.ORG GmbH

Es gibt eine Menge einzelne BI-Angebote in der Cloud, aber ein komplettes Data-Warehouse? Der Artikel zeigt, ob das heute bereits realisierbar ist oder es sich noch nicht lohnt.

Sollen DWH-Architekturen verändert oder neu aufgebaut werden, stellt sich zunehmend die Frage, ob das DWH mitsamt BI-Backend nicht auch in die Cloud verlagert werden kann. Neben den dafür allseitig angeführten Argumenten wie verringerte Bereitstellungszeiten, Elastizität und Kostenersparnis reizen viele Entscheider gerade die Möglichkeiten, die BI-Tools in der Cloud versprechen: Anwender würden immer die aktuelle Version des BI-Stacks nutzen und eine hohe Verfügbarkeit ließe sich trotz überschaubarer eigener IT-Ressourcen garantieren. Ein agiles Self-Service-BI verspräche zudem, die Mauer zwischen IT und Business zu überwinden, weil Anwender neue Auswertungen schnell und selbst entwerfen und auch sogleich testen könnten. Welche Möglichkeiten genau bieten sich hier zur Realisierung an?

Transaktionales BI

Viele Unternehmen sammeln die ersten Erfahrungen auf dem Weg in die Oracle-Cloud mit den BI-Apps. Eigentlich sind diese eine On-Premises-Lösung, um schnell Daten aus einer Anwendung aufzubereiten und so darzustellen, wie man es von Oracle-BI-Lösungen gewohnt ist – inklusive Dashboards und interaktiven Analysen.

Die BI-Apps wurden relativ früh in der Oracle-Cloud angeboten. Für die gerne ebenfalls in der Cloud betriebenen Oracle-Applikationen wie Human Capital Management (HCM) existieren fertige Lösungen, um die gewünschten Ergebnisse bereits fertig aufbereitet zu präsentieren. Selbst die fachliche Logik, also das Wissen darum, welche für die Analyse notwendigen Zahlen in welchen Tabellen der Anwendung verborgen sind, wird bereits mitgeliefert. Der Kunde spart sich die gesamte Wertschöpfungskette des klassischen Data Warehouse: Quellsystem-Analyse, Datenmodellierung, ETL sowie die Erstellung von Dashboards und Reports. Die BI-Apps sind Teil der sogenannten „Oracle Transactional Business Intelligence" (OTBI); transaktional, weil die BI-Elemente direkt aus den operativen Vorsystemen – den transaktionalen Anwendungen – gespeist werden.

Wann immer schnelle Analysen aus den Oracle-Applications erforderlich sind, ist OTBI eine interessante Lösung: Obwohl schnell und schlank, steht die Aufbereitung der erhaltenen Daten denen der klassischen Präsentation, wie man sie aus On-Premises-OBIEE-Systemen gewohnt ist, kaum in etwas nach – und ist mit minimalem Aufwand in der Oracle-Cloud verfügbar. OTBI ist für viele Oracle-Cloud-Apps einfach als Zusatz-Option buchbar.

Was das transaktionale Reporting nicht leistet, ist die Integration verschiedener Quellen, also die klassische Aufgabe des Enterprise-Data-Warehouse (EDWH). Daten können nur dort aufbereitet und dargestellt werden, wo die notwendigen Transformationen zwischen Quelle und BI-Darstellung durch Oracle vorbereitet sind. Sollen eigene Integrationen durchgeführt oder Daten aus verschiedenen Quellen konsolidiert werden, muss weiter ausgeholt werden.

BI Cloud Services und Analytics Cloud

Neben OTBI unterhält Oracle ein weiteres Cloud-basiertes Angebot: die Business-Intelligence-Cloud-Services (BICS), die Cloud-Alternative zur Installation des Oracle-Business-Intelligence-Servers (OBIEE) im eigenen Hause. Zur Anpassung an die Cloud-Umgebung hat Oracle vor allem die Funktionalität des Admin-Tools in eine neue Oberfläche integriert.

Das traditionelle Admin-Tool läuft lokal auf einem Entwickler-PC. Mit ihm wird das OBIEE-Repository erstellt: Den physischen Datenbank-Objekten werden logische Entitäten gegenübergestellt, die beispielsweise Tabellen durch Joins über entsprechende gemeinsame Attribute verbinden, und schließlich wird das für den Benutzer sichtbare Universum aus geschäftlichen Relationen definiert. Diese Funktionalität ist nun mithilfe des neuen Cloud-Admin-Tools weitgehend vom Windows-Client und seiner in die Jahre gekommenen Optik befreit und per Web-Browser nutzbar gemacht.

Einen großen Sprung nach vorne haben die Oracle-BI-Cloud-Angebote durch die Einführung der Analytics-Cloud (OAC) erfahren. Denn BICS wies einige Nachteile auf:

- Es konnten ausschließlich die Inhalte einer Datenbank aufbereitet werden
- Es ließen sich keine Agenten verwenden
- Der BI-Publisher und Essbase sind nicht unterstützt

BICS wurde daher eher als ein abteilungsfokussiertes Angebot eingeschätzt, das für den unternehmensweiten Einsatz nur bedingt tauglich schien. Dementsprechend verhielt sich auch das Preismodell: Für 3.500 Dollar pro Jahr (bei den Preisen sind nachfolgend stets die Listenpreise angeführt) erhielt der Kunde BICS mit einer dazugehörigen Cloud-Datenbank, die auf ein Storage-Volumen von 50 GB und einen Durchsatz von 300 GB begrenzt war.

OAC überwindet all diese Einschränkungen und bringt den zusätzlichen Vorteil, dass der zugrunde liegende Datenbank-Dienst frei gewählt und auch eine bereits vorhandene Datenbank-Lizenz genutzt werden kann. Im Gegensatz zu BICS ist OAC ein für die Cloud neu entwickeltes Produkt. BICS ist aus der Code-Basis des OBIEE entstanden, einem um die Jahrtausendwende entstandenen und mehrmals weiterverkauften Produkt, dessen Code-Qualität Gerüchten zufolge nicht geeignet war, daraus ein echtes Cloud-Produkt zu erzeugen *(siehe „http://redpillanalytics.com/introducing-oracle-analytics-cloud")*. Die Kosten belaufen sich „non-metered", also in einer pauschal abgerechneten Version, auf 3.000 Dollar pro Monat und OCPU (Oracle-Cloud-Version der Prozessormetrik bei der Lizenzberechnung) für die Standard Edition und 6.000 Dollar Listenpreis pro Monat und OCPU für die Enterprise Edition *(siehe „https://cloud.oracle.com/enUS/oac/pricing",* Stand Mitte 2017). Die Enterprise Edition beinhaltet auch Essbase *(siehe Tabelle 1)*.

Wie jedes BI-Tool muss auch OAC mit den darzubietenden Daten bestückt sein. In der physischen Schicht des OBIEE beziehungsweise BICS könnten natürlich transaktionale Daten der Vorsysteme direkt angeboten werden. Nun ist es allerdings so, dass die Data-Warehouse-Community ihre Referenz-Architektur nicht ohne Grund entworfen hat: „Performance", „Konformität" und „Konsistenz" sind nur einige Schlagwörter, die eine Architektur bezeichnen, die sich auf Dauer zwar nicht ganz ohne Aufwand an Modellierung, Ladelogik und Speicherplatz realisieren lässt – die aber eben doch in den allermeisten Fällen langfristig Zeit und Aufwand spart und zudem die entscheidenden Ergebnisse realistisch liefern kann.

DWH – Platform-as-a-Service

Ein klassisches, vollständiges Data Warehouse kann Daten aus verschiedenen Quellen annehmen, aufbereiten und in ein gemeinsames Datenmodell integrieren. Die Daten werden – oft weiter zurückreichend als die Quellsysteme – historisiert; durch Konsolidierung sind Redundanzen vermieden. Für die weitere Verarbeitung werden üblicherweise verschiedene Schichten angelegt: die Stage, der Core und ein oder mehrere Data-Marts.

Angebot	Kostenpunkt	Umfang	Zielgruppe
OTBI (-E)	Je nach Anwendung	Cloud-Fusion-spezifisches, transaktionales Berichtswesen	Abteilung bzw. Organisation (-E)
BICS	USD 30.000/Jahr/10 Nutzer (USD 1.000/Monat + USD 150/Monat/ Named User)	Abgespecktes OBIEE in der Cloud	Abteilung
OAC	USD 36.000(SE) bis 72.000 (EE)/ Jahr/OCPU + DB + IaaS	BI in der Cloud	Abteilung und Unternehmen (mit eigener DB und ETL)

Tabelle 1

Die Daten werden auseinandergenommen, transformiert und in ein dimensionales Modell geladen. Um das zu erreichen, sind eine Datenbank zur Ablage der jeweiligen Datenschichten und ein Extract-Transform-Load-Tool (ETL) erforderlich. Selbstverständlich bietet Oracle auch die Datenbank in der Cloud an. Dieses Angebot existiert in verschiedenen Ausprägungen:

- *Database-Schema-Service*
  Ein Schema in einer Cloud-Datenbank mit 50 GB Volumen

- *Exadata-Express*
  Eine Pluggable-Database (PDB) in einer 12.2-Cloud-Datenbank, ebenfalls mit einem Datenvolumen bis 50 GB

- *Database-Cloud-Service (DBCS)*
  Eine Datenbank auf einer Linux-VM in der Oracle-Public-Cloud, installierbar per Web-Tool in verschieden großen Ausprägungen

- *Exadata-Cloud-Service*
  DBCS auf Exadata

- *(Exadata)-Cloud-Service on Bare-Metal-Servers*
  DBCS auf einer dedizierten Hardware, die exklusiv für den Kunden reserviert ist (Private Cloud)

Sehr schön ist auch der „Oracle Public Cloud Data Transfer Service" zum Migrieren großer Datenmengen in die Cloud: Oracle schickt dem Kunden eine ZS4-Storage-Appliance, auf der die Daten zwischengelagert und dann zurückgeschickt werden.

Für eine DWH-Datenbank empfiehlt sich, solange – etwa aus Sicherheitsgründen, wegen der aktuellen Angriffsmöglichkeiten auf die Isolation virtueller Systeme (Spectre und Meltdown) – keine dedizierte Hardware benötigt wird, aber DWH-typische Anforderungen an Durchsatz und Zuverlässigkeit bestehen, der Exadata-Cloud-Service. Allerdings ist von den verfügbaren Methoden, Daten in DBCS zu laden (SQL-Developer-Anbindung, APEX-Maske, REST- und SOAP-Interface sowie ein spezieller Adapter namens DataSync), nur letztere in der Lage, überhaupt Daten in den in DWH-Umgebungen üblichen Größenordnungen entgegenzunehmen. Es handelt sich dabei um ein Java-Tool, das die Daten aus der Quelle lädt und in die DBCS-Datenbank schiebt.

Der eigentlich angestrebte Weg, Daten der Quell-Systeme in ein DWH zu laden, ist jedoch in der Regel die Nutzung eines ETL-Tools, im Oracle-Umfeld gerne der Oracle-Data-Integrator (ODI). Er verfügt über eine Vielzahl von Lade-Adaptern; die besten Ergebnisse werden allerdings der Erfahrung nach mit der Nutzung von direkten Datenbank-Links erzielt, zumindest wenn das Quellsystem auch eine Oracle-Datenbank verwendet.

Infrastructure-as-a-Service

Viele Anbieter von Cloud-Lösungen haben versucht, den Markt mit einer naheliegenden Strategie aufzurollen: Zunächst werden die Angebote mit überschaubarer Komplexität in die Cloud verlagert: Produkte, die mit möglichst wenig Anpassungen bei einer möglichst großen Kundenzahl erfolgreich einsatzbar sein könnten.

Auch Oracle hatte im ersten Schritt universell einsetzbare Produkte als Software-as-a-Service (SaaS) oder Platform-as-a-Service (PaaS) anzubieten: HCM, BI-Apps, OTBI-Bausteine und BICS. Aber spätestens, wenn ein mehrschichtiges DWH mit selbstmodellierten Schichten – auch mit noch nicht in der Cloud vorliegenden Quellen und angepassten ETL-Strecken, die diese konsolidieren – aufgebaut werden soll, reichen die vorgefertigten Cloud-Angebote nicht mehr aus, und es bleibt nur der Ansatz Infrastructure-as-a-Service (IaaS) übrig. Hier wird zwar die Infrastruktur in der Cloud betrieben, deren Grenzschicht ist jedoch das Betriebssystem. Der Kunde erhält eigene virtuelle Maschinen, auf denen er dann die DWH-Komponenten selbst installieren und betreiben muss.

So gibt es zwar ein Produkt mit dem Namen „Oracle Data Integrator Cloud Service", aber die Installationsanleitung umfasst sämtliche Schritte, die nötig sind, um das Repository, das Studio und die Agenten manuell auf der Kommandozeile in einer VM zu installieren. In diesem Markt ist das Angebot von Oracle vergleichbar, wenn nicht austauschbar, mit dem anderer Anbieter wie AWS. Nichtsdestotrotz läuft die Datenbank in der Oracle-Cloud einer Enkitec-Studie zufolge deutlich effektiver als bei vergleichbaren IaaS-Providern wie AWS *(siehe „https://www.accenture.com/t20161013T060358Z__w__/us-en/_acnmedia/PDF-34/Accenture-Oracle-Cloud-Performance-Test-October2016.pdf")*. Es können zwar einige der ursprünglichen Ziele der Cloud-Migration realisiert werden, der Kunde muss keine eigene Hardware vorhalten und pflegen, aber der Aufwand für die Installation und Wartung der Software bleibt ihm nicht erspart.

Das Endresultat ist ein Gemischtwarenladen

Der Autor hat im vergangenen Jahr ein Cloud-DWH-Einführungsprojekt begleitet, das sämtliche hier diskutierten Aspekte umfasste und auch von Oracle konzipiert wurde.

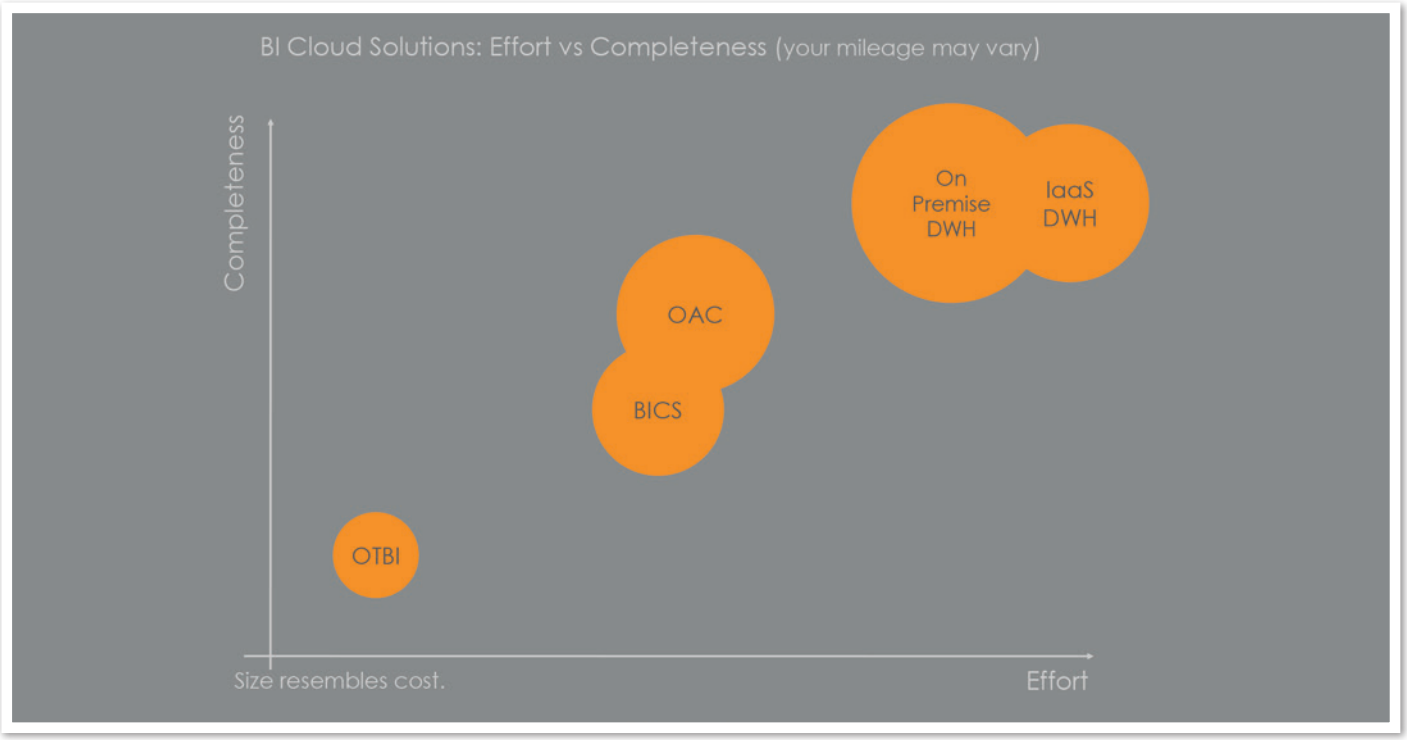


Abbildung 1: Die aktuell bereitstehenden Oracle-Cloud-BI-Angebote

Der Prototyp der neuen BI-Landschaft bestand aus Applikationen, die größtenteils bereits in der Cloud verwendet wurden: HCM, UCM, Field Service, Finance. ETL wurde mit dem ODI ausgeführt, der als IaaS-Komponente in der Oracle-Compute-Cloud installiert war. Die ODI-Studio-Clients liefen lokal oder auf einem Terminal-Server in der Compute-Cloud. Modelliert wurde mit einem lokal auf den Entwickler-PCs installierten SQL-Developer-Data-Modeler. Die Versionskontrolle erfolgte über Subversion, dessen Server wiederum in der Compute-Cloud beheimatet war. Die Datenbanken waren in einer Exadata-Cloud-Service-Quarter-Rack-Umgebung untergebracht. Als BI-Komponente diente BI-Analytics-Cloud-Service. Daten aus bestehenden On-Premises-Systemen wurden über einen ODI-Agenten geladen und teilweise in der Storage-Cloud abgelegt.

Eines der angetroffenen Probleme war, das selbst die Oracle-Cloud-Apps nicht immer über automatisiert benutzbare Schnittstellen verfügten, sondern darauf ausgelegt waren, Exporte als Dateien auszuführen, die per Hand aus einer GUI erzeugt werden sollen. Werden solche Applikationen On-Premises betrieben, existiert meist die Möglichkeit, ETL direkt aus dem Datenmodell der Anwendung zu betreiben – bei den Cloud-Apps ist die Datensenke der Anwendung jedoch ein geschlossenes System.

Fazit

Die aktuell bereitstehenden Oracle-Cloud-BI-Angebote *(siehe Abbildung 1)* sind in erster Linie dort erfolgreich einsetzbar, wo es um schnelle, abgrenzbare Lösungen geht. Das angesprochene Projekt fand im Frühjahr 2017 statt. Ein Enterprise-Data-Warehouse ist allerdings in den meisten Fällen kein einfach abgrenzbares Projekt. Der Gedanke, die komplette Business Intelligence in die Cloud zu verlagern, ist für Entscheider sehr attraktiv, aber nicht immer zufriedenstellend umsetzbar. Hier muss selbst konzipiert, entwickelt und installiert werden.

Bestehende Hardware in die Cloud zu verlagern, funktioniert bei Oracle wie bei Amazon selbstverständlich. Ob die Komplexität der Schnittstellen der unterschiedlichen Cloud-Produkte untereinander am Ende des Tages einfacher zu handhaben ist als technische Schnittstellen von On-Premises betriebenen Systemen, bleibt hingegen abzuwarten. Die Gesamtsumme an Komplexität sowohl technischer als auch vertraglicher Art erhöht sich durch den Cloud-Einsatz sicher nicht. Es gibt noch einen anderen Aspekt: Ein strategischer Vorteil eines EDWH ist, einen konsolidierten und vollständigen Daten-Pool jederzeit im Zugriff zu haben.

Erfolgreiche Unternehmen möchten sich natürlich auf ihr Kerngeschäft konzentrieren und keine Unsummen für Anschaffung und Betrieb im Zweifel schnell veralteter IT ausgeben. Die Verteilung der eigenen Daten auf eine Vielzahl von gemieteten Cloud-Anwendungen birgt jedoch die Gefahr, eben diesen Vorteil aufzugeben. Es droht eine Zersiedelung des Datenbestands, in der die Übersicht und vor allem auch die Hoheit über die eigenen Daten verloren gehen kann. Und die Entscheidung, die eigene IT nur bei einem Cloud-Provider anzusiedeln, erhöht wiederum die Abhängigkeit von eben diesem Anbieter.

IT-Infrastruktur in die Cloud zu verlagern, bedeutet, sie von Fremdanbietern betreiben zu lassen, und bedingt ein engeres Verhältnis als etwa ein Lizenz-Abkommen für Software. Ändert der Anbieter beispielsweise kurzfristig seine Geschäftsbedingungen in einer inakzeptablen Weise, kann es sehr schwer fallen, die Cloud wieder zu verlassen. Für Aufsehen hat in der Oracle-Community im Januar 2017 beispielsweise die Entscheidung gesorgt, überraschend die Core-Factor-Table zu Ungunsten von Nicht-Oracle-IaaS-Providern zu ändern *(siehe „https://oracle-base.com/articles/misc/oracle-databases-in-the-cloud“)*. Alternativ gibt es ja auch noch die Möglichkeit, sich eine private Cloud auf Basis der Engineered-Systems oder Cloud-Machine aufzubauen.

Jan Schreiber
js@loopback.org

Gemeinsam stärker: Oracle und die Graph-Datenbank Neo4j

Stefan Kolmar, Neo4j

Relationale Datenbankmanagement-Systeme (RDBMS) wie Oracle sind in vielen Fällen die erste Wahl für Unternehmensanwendungen. Doch die Anforderungen steigen – vor allem in Hinblick auf die Verarbeitung vernetzter Daten. Gefragt sind Agilität, Funktionalität und Geschwindigkeit, um auch heterogene Daten in Echtzeit abfragen und verwalten zu können. Kommen RDBMS und Graph-Technologie gemeinsam zum Einsatz, profitieren Unternehmen von den Vorteilen beider Systeme, ohne bestehende Investitionen zu beinträchtigen. Wie die Koexistenz beider Technologien im Detail aussieht, hängt vom Anwendungsfall ab.

Big Data, IoT, künstliche Intelligenz – vernetzte Daten sind für viele Unternehmen der Ausgangspunkt für erweiterte Funktionalitäten, effizientere Prozesse und innovative Strategien, die ihre Wettbewerbsfähigkeit auch für die Zukunft sichern. Je schneller die Komplexität und die Menge der zu verarbeitenden Daten steigen, desto deutlicher zeichnen sich jedoch die Grenzen relationaler Systeme ab. Als Alternative zu bestehenden Systemen sind daher in den letzten Jahren verstärkt NoSQL-Datenbanken zum Einsatz gekommen, die im komplexen Datengeflecht besser und vor allem schneller navigieren. Das gilt insbesondere für Graph-Datenbanken, die nicht nur die eigentlichen Daten, sondern auch die Beziehungen zwischen den Daten in den Fokus stellen.

Steht Unternehmen damit ein radikaler Umbruch ihrer Datenbank-Technologien bevor? Keineswegs. Graph-Datenbanken und relationale Datenbanken treten nicht zwangsläufig in Konkurrenz miteinander. Vielmehr ergeben sich durch die polyglotte Persistenz beider Technologien zusätzliche Vorteile. Das Zusammenspiel von Neo4j und Oracle ermöglicht es System-Architekten und Entwicklern, ihre Anwendungen ohne Unterbrechung weiter auf bestehenden Systemen laufen zu lassen und dabei gleichzeitig alle strategischen Möglichkeiten einer Graph-Datenbank zu nutzen.

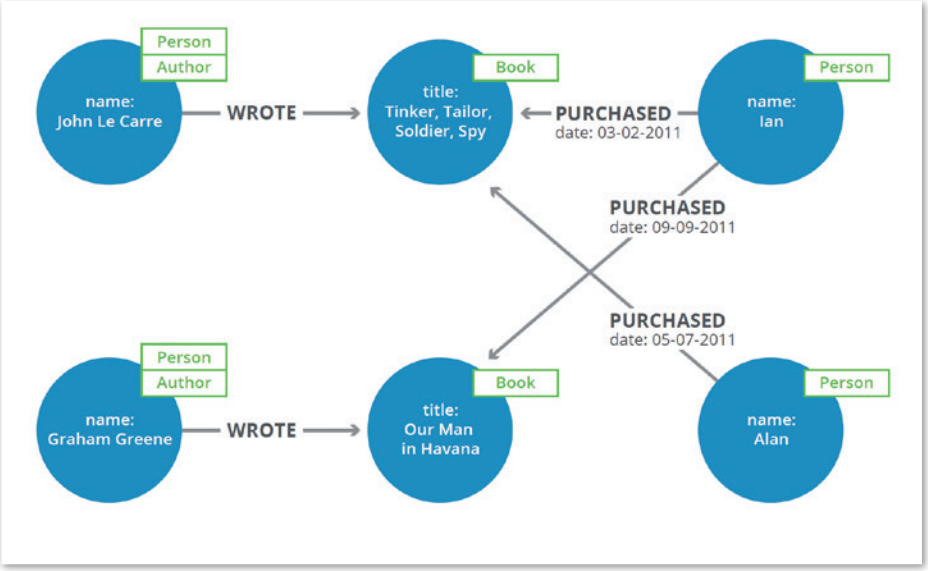


Abbildung 1: Ein einfaches Graphen-Datenmodell: Bücher, Autoren und Eigentümer

Elementare Unterschiede zwischen Graph und RDBMS

Um zu verstehen, wie Graph-Technologie und relationale Datenbank-Systeme parallel und in Kombination genutzt werden können, lohnt sich ein grundsätzlicher Blick auf die Prinzipien beider Technologien. Relationale Datenbanken sind für die Verarbeitung mit Datenbeziehungen nur bedingt geeignet – auch wenn es das Wort „relational" („relationships") anders vermuten lässt. Beziehungen sind durch sogenannte „JOINs" miteinander verknüpft. Je tiefer man sich in der Datenebene bewegt und je mehr JOINs verwendet werden, desto mehr leidet die Performance.

Der Begriff „Relation" bezieht sich vielmehr auf eine mathematische Notation, die aus Attributen und Tupeln besteht und üblicherweise als Tabelle aus Spalten (Attributwerten) und Zeilen (Tupeln) beschrieben wird. Stark strukturierte Daten können in diesen Tabellen mit vordefinierten Spalten problemlos gespeichert werden. Dabei enthalten die Zeilen in diesen Tabellen den gleichen Typ an Informationen. Dieses Tabellen-Format erlaubt nur wenig Spielraum bei der Entwicklung von Anwendungen, die diesem starren Format folgen müssen. Daher eignet sich ein solches festes Schema am besten für stark strukturierte und vorab klar definierte Aufgaben.

Im Gegensatz dazu steht bei einer Graph-Datenbank die Datenbeziehung an erster Stelle. So ist Neo4j ein ACID-kompatibles, transaktionales Datenbankmanagement-System, dessen CRUD-Operationen (Create, Read, Update und Delete) nativ in einem Graph-Datenmodell arbeiten. Dieses ist intuitiv zu verstehen und stellt die Daten in einer Weise dar, die uns beispielsweise von einem U-Bahn-Netz oder einem Familienstammbaum vertraut sind.

Der Graph besteht aus zwei wesentlichen Elementen: Knoten beziehungsweise einzelnen Datensätzen („nodes") und deren Beziehungen untereinander („edges"). Jeder Knoten stellt eine Entität dar. Jede Beziehung zeigt, wie zwei Knoten miteinander verbunden sind. So entsteht aus einer Vielzahl von heterogenen Daten ein semantischer Kontext, der sich flexibel erweitern lässt und ein genaues Abbild einer Problem-Domäne liefert *(siehe Abbildung 1)*.

Da es sich bei Neo4j um eine Schema-optionale Datenbank handelt, entfällt der Aufbau komplexer, vordefinierter Datenmodelle, von denen man annimmt, dass sie den geschäftlichen Anforderungen bestmöglich gerecht werden. Stattdessen sind das Datenmodell und die Validierung auf der Anwendungsschicht abstrahiert. Das verschafft Entwicklern eine höhere Agilität und beschleunigt die Bereitstellung von Anwendungen, ohne dass die nötige Kontrolle über die Definition der eingehenden Daten erforderlich ist.

Der Unterschied zwischen Graph-Datenbanken und RDMBS zeigt sich insbesondere bei der Verarbeitung vernetzter Daten. Mit Neo4j lassen sich beispielsweise Daten mit einer Tiefe von mehreren Zehnmillionen Verbindungen pro Sekunde pro Prozessorkern abfragen. In der Welt der relationalen Datenbanken entspricht dies mehreren Millionen JOIN-Operationen pro Sekunde pro Kern – was unmöglich ist. Das wirkt sich entscheidend auf die Performance aus: Je mehr Daten vorliegen, desto langsamer und schwerer fällt es relationalen Datenbanken, Beziehungen zwischen Daten abzufragen.

Effizientes Zusammenspiel für höhere Performance

Der gemeinsame Einsatz von Neo4j und Oracle ermöglicht es, die optimale Technologie für die jeweilige Anwendung zu nutzen: Tabellarische, strukturierte Daten lassen sich sehr gut mit Oracle managen; unstrukturierte, stark vernetzte Daten oder dynamische Daten wie Hierarchien und Netzwerke können aus Oracle nach Neo4j migriert und dort abgefragt werden. Bei richtig konzipierter Koexistenz verbessert sich die Anwendungsleistung deutlich, da Suchvorgänge („Traversals") im Graphen in Millisekunden erfolgen, während sich Transaktionsdaten aus Oracle auslesen lassen.

Neo4j kann mit Oracle – sowie mit jedem anderen relationalen Datenbankmanagement-System – auf verschiedene Weise zusammenarbeiten. Welchen Ansatz man wählt, hängt vom jeweiligen Anwendungsfall ab. Dazu vier Koexistenz-Ansätze anhand von Beispielen, nämlich das Migrieren oder Synchronisieren eines Teilbereichs der Daten sowie die vollständige Migration oder Synchronisierung aller Daten.

Einen Datenteilbereich nach Neo4j migrieren und synchronisieren

Wenn Abfragen aufgrund der Tiefe und Komplexität der Datenbeziehungen in bestehenden Datenbank-Systemen nicht effizient durchgeführt werden können, empfiehlt es sich, relevante Daten nach Neo4j zu migrieren. Neo4j ist dann der transaktionale und ACID-kompatible Datenspeicher für diesen Teil der Datenmenge. Zum Beispiel verbleiben Informationen zu Kunden und Produkten in Oracle, während ihre Kennungen sowie die Beziehung zwischen ihnen im Graphen abgelegt sind. Abfragen, die auf die Verbindung der Daten abzielen, lässt die Anwendung so zunächst über die Graph-Datenbank laufen und nutzt das ResultSet für eine weitere Detailabfrage in Oracle.

Die Migration der Daten nach Neo4j kann manuell oder automatisch mithilfe eines ETL-Tools erfolgen. Dabei werden Tabellen als CSV-Dateien aus Oracle exportiert und in Neo4j importiert – entweder über ein Befehlszeilen-Tool oder mithilfe der Syntax der Graph-Abfragesprache Cypher.

Ein anderer Ansatz ist das Synchronisieren eines Teilbereichs der Daten. Die Synchronisierung erfolgt hier über Middleware wie beispielsweise Oracle GoldenGate und entsprechenden Adapter.

Die norwegische Telekommunikationsgesellschaft Telenor synchronisierte beispielsweise Daten für die Ressourcen-Autorisierung von Self-Service-Webanwendungen auf Neo4j und konnte so die Anmeldezeiten für Kunden auf nur wenige Millisekunden reduzieren. Bei Ressourcen-Autorisierungen handelt es sich um stark vernetzte Daten, die bei jedem Log-in-Verfahren und für jeden einzelnen Benutzer neu zu berechnen sind. Insbesondere für große Kunden mit vielen Nutzern kam es mit dem bestehenden Datenbank-System zu langen Wartezeiten. Auch der Versuch, die Serviceleistung für Großkunden über eine gespeicherte Prozedur zur Vorabberechnung der Ressourcen-Autorisierungen zu verbessern, konnte das Problem nicht vollständig lösen.

Da die Vorabberechnung nachts im Stapelverfahren durchgeführt und die Ergebnisse teilweise bis zu 24 Stunden zwischengespeichert wurden, konnten Änderungen der Daten (etwa neue oder gelöschte Benutzer) in manchen Fällen erst nach Ablauf eines Tages im System umgesetzt werden. Zudem erfolgten die Ressourcen-Autorisierungen für kleinere Kunden nach wie vor über Ad-hoc-Berechnungen und liefen damit sehr langsam ab. Angesichts des zukünftigen Wachstums des Unternehmens stellt die komplexe Vorabberechnung mit rund 1.500 Zeilen SQL-Code nur eine Übergangslösung dar.

Die Lösung bestand für Telenor in einem NoSQL-Konzept als Ergänzung des vorhandenen relationalen Datenbankmanagement-Systems. Die Daten für Suchfunktionen wurden mit Solr/Lucene synchronisiert, die Daten für Ressourcen-Autorisierungen mit Neo4j. So ließen sich Anmeldeverfahren für alle Kunden in Echtzeit durchführen und Änderungen innerhalb der Daten direkt umsetzen.

Alle Daten nach Neo4j migrieren

Es gibt Szenarien, bei denen es notwendig ist, über eine Neuentwicklung einer Anwendung nachzudenken. Dazu gehören unzumutbare Wartezeiten für Fachanwender sowie Abfragen, die in Echtzeit wechselnde Datenmengen und Datentiefen verknüpfen sollen und bei denen Voraussagen oder Vorausberechnungen nur schwer möglich sind. Auch zu komplexe Abfragen, die insbesondere neue Teammitglieder vor Herausforderungen stellen und entsprechende Schulungen voraussetzen, können ein Grund sein, eine bestehende Anwendung zu ersetzen. In diesen Fällen – wenn also extrem stark vernetzte Daten ins Spiel kommen – empfiehlt es sich tatsächlich, Neo4j anstatt eines relationalen Systems als alleinigen Datenbankspeicher einzusetzen und eine vollständige Migration der Daten vorzunehmen.

Die Migration folgt dabei dem gleichen Prinzip wie beim teilweisen Migrieren. Ergänzt wird das Vorgehen allerdings durch die Überarbeitung des Anwendungscodes, der mit der Datenbank interagiert, und durch die Erstellung der entsprechenden Abfragen. Die Umstellung auf eine Graph-Datenbank kann in unterschiedlichen Bereichen sinnvoll sein und einen echten Mehrwert aus bestehenden Daten generieren.

Der Spielzeughersteller Schleich nutzt Neo4j beispielsweise für ein neues PDM-System, in dem das komplexe Netzwerk aus CAD-Daten, Stücklisten, Regularien, Material- und Teileinformationen, 2D- und 3D-Daten, Fertigungsanweisungen sowie Lieferanten und Zulieferern eines Produkts abgelegt ist. Über fachspezifische Applikationen mit individueller Benutzeroberfläche können Mitarbeiter nun auf die funktionalen Bausteine des PDM-Systems zugreifen und die komplette Supply Chain – vom Design über die Produktion bis hin zum Vertrieb – einsehen und rückverfolgen. Das bringt Pluspunkte für das Material-Management und die Compliance. Darüber hinaus stellen Schnittstellen zu anderen Systemen eine einfache Integration in die heterogene Systemlandschaft des Unternehmens sicher und ermöglichen eine direkte Anbindung an SAP- und ERP-Systeme.

Auch bei Übernahmen und Fusionen, bei denen eine Vielzahl unterschiedlicher Systeme integriert werden muss, um die Daten aus verschiedenen Quellen in einer Anwendung zu nutzen, kann eine vollständige Migration notwendig sein. So zum Beispiel bei der seit 1939 bestehenden schwedischen Druckerei Billes Tryckeri. Eine Reihe vorausgegangener Fusionen und Akquisitionen hatte zu einem Flickenteppich an Tools und IT-Systemen geführt. Mit der nahtlosen Übertragung der Daten aus allen Systemen

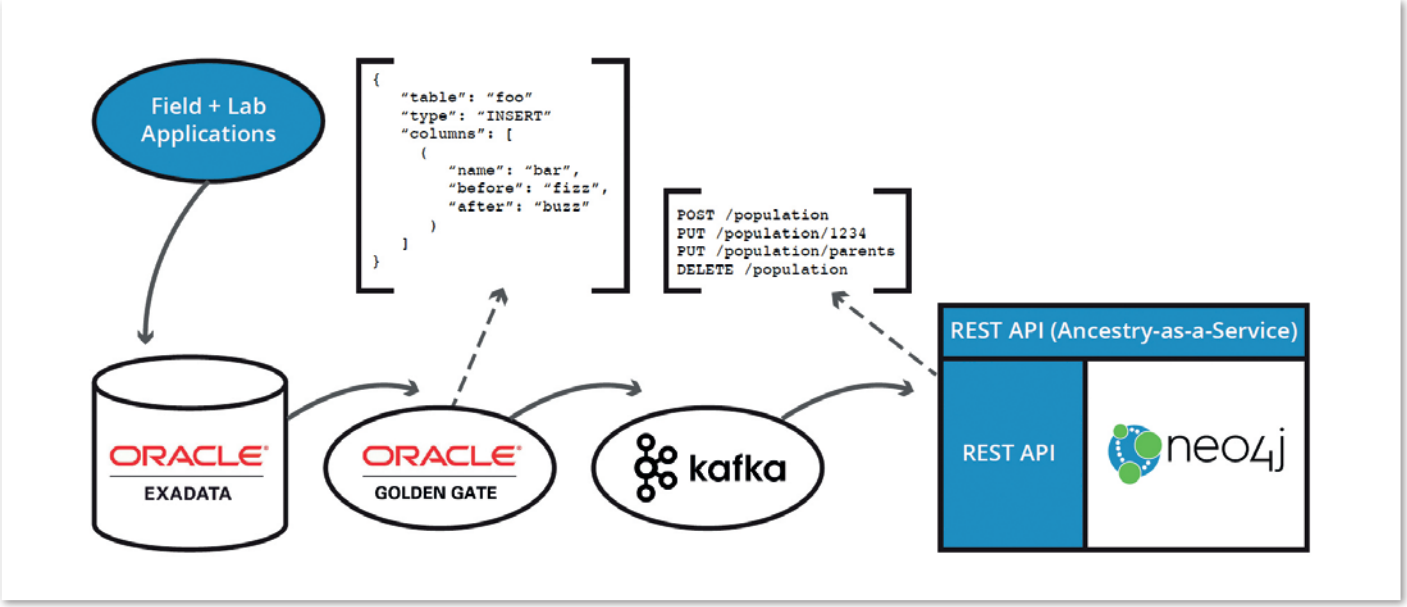


Abbildung 2: Abfragezeiten für Oracle Exadata versus Neo4j bei Monsanto

in das Neo4j-basierte System „Poff“ konnte das Unternehmen Änderungen schneller und einfacher vornehmen und reduzierte so den Aufwand für die Systemintegration erheblich. Insgesamt können nun 40 externe Systeme Kundenaufträge annehmen und diese über ein gemeinsames System bearbeiten. Die effiziente Administration des Systems ermöglichte es darüber hinaus, neue Services wie Online-Bestellungen von Endkunden anzubieten.

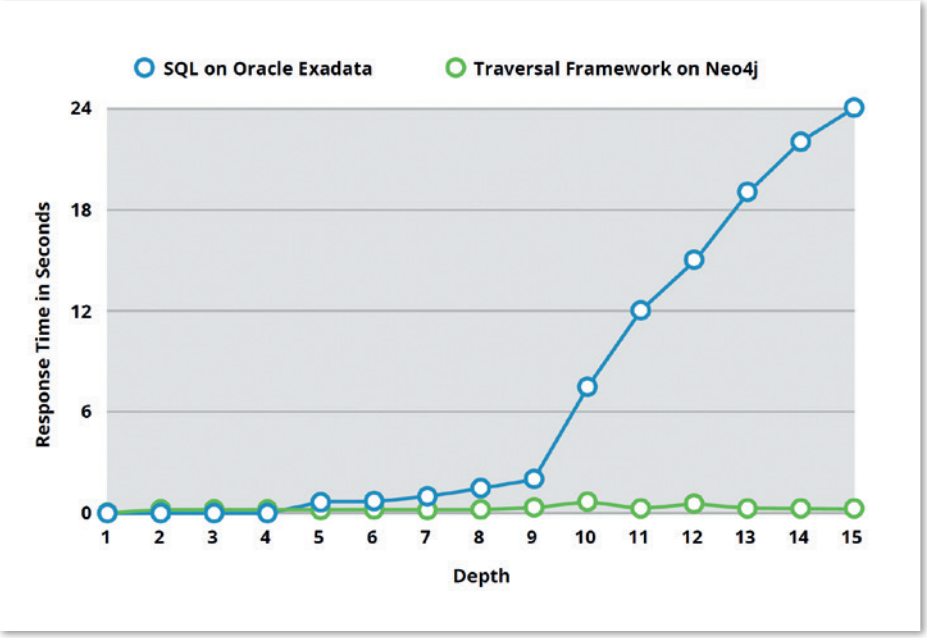


Abbildung 3: Einsatz von Oracle und Neo4j für die Monsanto-Anwendung zur Abfrage der genetischen Abstammung verschiedener Saat-Produkte

Vollständige Synchronisierung der Daten zwischen Oracle und Neo4j

Komplexere Anwendungsszenarien verlangen nach einer vollständigen Synchronisierung aller Daten zwischen Oracle und Neo4j. Dazu zählen beispielsweise Anwendungen mit Daten aus mehreren Datenquellen oder auch Fallbeispiele, in denen mehrere bestehende Anwendungen in eine Oracle-Datenbank schreiben und eine Änderung aus Kostengründen nicht infrage kommt.

Genau dieser Fall traf auf das Agrar- und Biotechnologie-Unternehmen Monsanto zu. Es nutzt Neo4j, um die Kerndaten der genetischen Abstammung verschiedener Saatprodukte zu managen und abzufragen. Ursprünglich arbeitete das Unternehmen mit einer 96 CPUs umfassenden Oracle-Exadata-Installation, die alle Daten mithilfe gespeicherter Prozeduren, JOIN-Tabellen, rekursiver Abfragen und dualer Indizes zur Leistungsoptimierung hostete. Trotz zahlreicher Tuning- und Optimierungsversuche gelang es der Datenbank nicht, die vernetzten Daten in Echtzeit zu verarbeiten. Dies war jedoch eine Grundvoraussetzung für die Einführung eines neuen genomischen Tests im Unternehmen, der die Markteinführungszeit von Produkten um ein volles Jahr verkürzt. In einem ersten Versuch zur Erzeugung von Echtzeitergebnissen erstellte das Team sehr große In-Memory-Graphen und parste diese. Allerdings waren diese Graphen nach Ende der Abfragen nicht mehr verfügbar.

Monsanto entschied sich daraufhin, den gesamten Bestand der genetischen Abstammungsdaten in einer Graph-Datenbank zu hinterlegen. Anstatt die Datenbank-Verbindungen der Exadata-Umgebung auf einen Schlag abzuschalten, baute das Team eine individuelle API-Schicht zur Synchronisierung der Datenströme zwischen Exadata und Neo4j. Ein neues Interface für Abfragen ermöglicht es den Wissenschaftlern, tiefgehende, schlagwortbasierte Abfragen auf einfache Weise auszuführen – eine Abfrage, die mit SQL-Algorithmen nicht möglich war *(siehe Abbildung 2)*.

Die Architektur nutzt Apache Kafka zur Übergabe von Echtzeittransaktionsdaten aus Oracle an Neo4j. Außerdem entwickelte das Team einen Connector für Oracle GoldenGate und Kafka, den es mit einer Open-Source-Lizenz auf GitHub zur Verfügung stellte *(siehe Abbildung 3)*.

Heute verarbeitet die Graph-Datenbank mehr als 600 Millionen REST-Abfragen über etwa eine Milliarde im Graphen hinterlegte Knoten und liefert damit erstmals Ergebnisse innerhalb von Zehntel Millisekunden. Dabei bedient die Lösung für die genetischen Abstammungsdaten etwa 120 verschiedene Anwendungen und steht für Wissenschaftler unternehmensweit zur Verfügung.

Fazit

Das jeweilige Anwendungsszenario entscheidet, welche Art von polyglotter Persistenz letztendlich gewählt wird. Generell gilt: Das Design eines DBMS sollte der Datenstruktur folgen, um sowohl das Datenmodell als auch die Abfrage-Workloads bestmöglich unterstützen zu können. Jede Datenstruktur ist für einen bestimmten Zweck ausgelegt und keine Datenbank ist universell einsetzbar. Das gilt für relationale genauso wie für NoSQL-Datenbanken wie spaltenorientierte, relationale, dokumentbasierte oder eben auch graphbasierte Systeme.

Ein großer Teil von Unternehmensanwendungen läuft auf Oracle. Es ist nicht sinnvoll, vorhandene Investitionen in die Infrastruktur, Tools und Trainings über Bord zu werfen, sobald in der digitalen Welt eine neue Herausforderung für das Daten-Management auftaucht. Das Zusammenspiel bewährter Technologien wie Oracle mit anderen Datenbank-Technologien wie der Graph-Plattform Neo4j bietet hier einen guten Mittelweg, um Nutzern die beste Lösung für ihre jeweilige Anwendung anzubieten – ganz ohne bestehende Prozesse zu gefährden oder unnötige Kosten zu verursachen.

Stefan Kolmar
vertrieb@neo4j.com

DOAG 2018
Exa & Middleware Days

18. & 19. Juni 2018 in Frankfurt

DOAG

exaday.doag.org

DOAG 2018
Logistik + IT

14. Juni 2018 in Köln

logistik.doag.org