© nadla | iStock

Big-Data-Technologien in der Fahrzeugentwicklung

AUTOR

Dr. Tobias Abthoff
ist Mitglied des Vorstands und Leiter Technology & Business Development bei der NorCom IT AG in München.

Während einer Testfahrt können mehrere Terabyte an Messdaten anfallen, und in der gesamten Entwicklungsphase entstehen somit Datenmengen im Petabyte-Bereich oder darüber. Es ist weder praktikabel, noch langfristig realistisch umsetzbar, diese Dimensionen über Datenleitungen abzudecken. NorCom schlägt vor: Anstatt die Daten zu bewegen, werden die Algorithmen zu ihnen transferiert. Das IT-Unternehmen liefert auf Basis dieses Prinzips eine Lösung für die Analyse von Mess- und Felddaten.

Sonderdruck aus ATZelektronik 05|2016 | Springer Vieweg
Springer Fachmedien Wiesbaden GmbH

MOTIVATION

Die Analyse großer Datenmengen ist ein immer wichtiger werdender Bestandteil der industriellen Wertschöpfung. Dies gilt auch für die Automobilindustrie und für alle Phasen des Produktzyklus. So entwickeln sich Fahrzeuge immer mehr zu mobilen Rechen- und Datenzentren, die immer stärker vernetzt sind und sich sowohl untereinander, als auch mit stationären Rechenzentren, austauschen.

Insbesondere die Fahrzeugentwicklung wird immer datenintensiver, was im hohen Maße durch das Autonome Fahren und Connected Car getrieben wird. So erzeugt beispielsweise ein autonom fahrendes Fahrzeug einen Rohdatenstrom von über einem Gigabyte pro Sekunde. Während einer Testfahrt können so bereits mehrere Terabyte an Messdaten anfallen, und in der gesamten Entwicklungsphase entstehen somit Datenmengen im Petabyte-Bereich oder darüber. Ähnliche Größenordnungen erreicht man auch in anderen Bereichen, wie etwa der Motorenentwicklung oder bei der Abgasmessung.

Hinzu kommt, dass Feldtests und Testläufe auf der ganzen Welt verteilt stattfinden, woraus sich die Frage ergibt, auf welche Weise Entwickler und Ingenieure am besten auf die Daten zugreifen können, **BILD 1**.

Allein aufgrund ihrer Größe ist es nicht mehr praktikabel, die Daten per Datenleitungen zu übertragen. Sie werden stattdessen auf Datenträger kopiert und physisch zwischen Standorten ausgetauscht. Neben der offensichtlichen Zeitverzögerung, die Tage oder gar Wochen betragen kann, bedeutet dies auch erhebliche Einbußen an Flexibilität und an der verfügbaren Datenmenge für die Analyse. Vor allem wäre wünschenswert:

- schnelle Auswertungen erstellen zu können, beispielsweise um bei Auffälligkeiten sofort weitere Tests fahren zu können
- den Zugriff auf alle bisher eingefahrenen Daten zu haben, etwa um schnell zu ermitteln, ob ein Problem bereits früher an einem anderen Teststandort aufgetreten ist.

DASENSE: ALGORITHMEN ZU DEN DATEN

Die grundlegende Idee, um diese Anforderungen zu realisieren, besteht darin, den umgekehrten Weg zu gehen: Anstatt die Daten zu bewegen, werden die Algorithmen zu den Daten transferiert.

Damit ist die Grundphilosophie von DaSense beschrieben, eine Lösung für die Analyse von Mess- und Felddaten im automobilen Umfeld, die von der NorCom IT AG entwickelt wird.

BILD 2 beschreibt die Unterschiede zu bisherigen Arbeitsabläufen. Im traditionellen Arbeitsablauf finden Entwicklung und Analyse auf lokalen Rechnern statt. Dabei werden zuerst Daten von einem zentralen Dateiarchiv angefordert, die dann auf den Rechner übertragen werden, wo schließlich die Analyse durchgeführt wird. Mit DaSense werden stattdessen die Analysealgorithmen zu den Daten übertragen, die dann in vielen Instanzen auf einem Cluster parallel ausgeführt werden. Anschließend brauchen nur noch die Ergebnisse an die Entwickler zurückübermittelt zu werden.

BILD 1 Die Entwicklung und Analyse finden an weltweit verteilten Standorten statt; Entwickler stehen vor dem Problem, an die Daten zu gelangen (© iStock I adventtr)

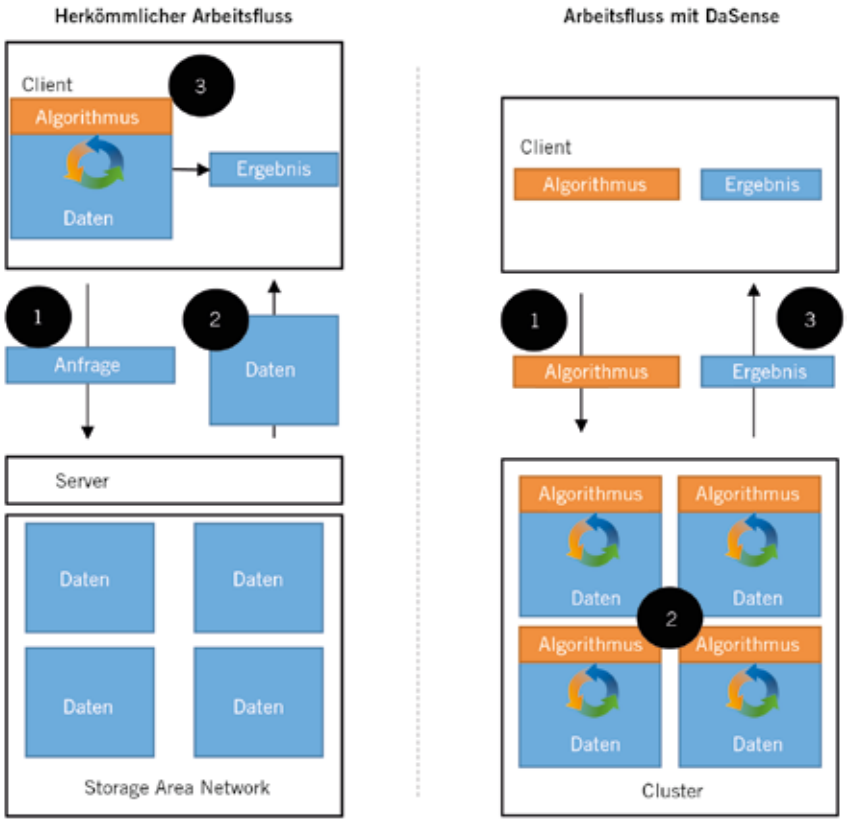

BILD 2 Unterschiede zwischen heutigen traditionellen Arbeitsabläufen und vereinfachten Prozessen mit DaSense (© NorCom)

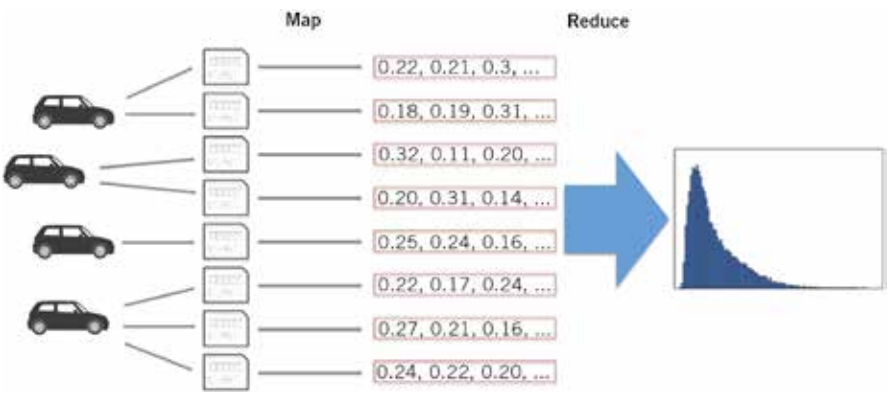

BILD 3 Eine Fahrzeugflotte erzeugt über einen Zeitraum hinweg Messdaten, die beispielsweise im MDF4- oder ATFX-Format auf dem Cluster liegen; in dieser vereinfachten Darstellung werden im Mapper-Schritt viele Map-Prozesse parallel ausgeführt, die jeweils eine Datei analysieren und die gewünschte Information extrahieren; im Reduce-Schritt werden diese Ergebnisse zusammengeführt und dem Benutzer als Gesamtergebnis zur Verfügung gestellt (© NorCom)

BIG DATA

DaSense basiert auf sogenannten Big-Data-Technologien, wobei der Begriff Big Data eine Sammelbezeichnung für eine ganze Reihe von Technologien ist, die im Umfeld der großen Internetunternehmen entstanden sind. Eine der bekanntesten und am weitesten verbreiteten Big-Data-Technologien ist Hadoop mit seinem zugehörigen Softwareökosystem, das auch als die technologische Grundlage von DaSense dient [1].

CLUSTER

Ein grundlegendes Konzept, das DaSense von Hadoop übernimmt, ist der Computercluster. Ein solcher Cluster besteht in der Regel aus sehr vielen einzelnen Computern, auf denen Analysen parallelisiert abgearbeitet werden. Dazu werden die einzelnen Rechner („Knoten“) so miteinander vernetzt, dass sie für den Benutzer über eine einheitliche Schnittstelle zur Verfügung stehen. Eine wesentliche Eigenschaft eines DaSense-Clusters wird oft durch den englischen Ausdruck „scale-out“ beschrieben, was sich in etwa mit „horizontale Skalierbarkeit“ übersetzen lässt. Konkret ist hier gemeint, dass zum einen die einzelnen Knoten aus Standardkomponenten bestehen, die sich leicht und kostengünstig beschaffen lassen.

Zum anderen lässt sich ein Cluster nahezu nach Belieben mit wachsenden Datenmengen erweitern. So kann man beispielsweise zu Beginn eines Big-Data-Projekts einen kleinen Cluster nutzen, um Verfahren zu implementieren und zu testen. In einer späteren Phase kann dieser Cluster nach Bedarf sukzessive erweitert werden. Dabei sorgt die DaSense-Infrastruktur dafür, dass Analysen skalieren und Analysezeiten pro Dateneinheit nahezu gleich bleiben.

DATA LAKE

Der sogenannte Data Lake umschreibt eine fundamentale Funktion eines DaSense-Clusters, nämlich Daten zu speichern. Abgesehen von den bereits erwähnten Datenmengen, schließt der Begriff weitere Aspekte ein, die einen Data Lake von herkömmlichen Datenspeichermodellen unterscheiden.

Hier ist zunächst die Datenlokalität zu nennen. Um Latenzen innerhalb eines Clusters zu verringern, müssen Analysen und die zugehörigen Daten effizient zusammengebracht werden. Da Datenanalysen letztendlich auf einzelnen Knoten eines Clusters stattfinden, sorgt die DaSense-Architektur dafür, dass Analysesoftware auf dem Knoten ausgeführt wird, der auf die angeforderten Daten den schnellsten Zugriff hat. Folgende Aspekte gilt es zu berücksichtigen:

- Fehlertoleranz: Dies beinhaltet insbesondere, dass Ausfälle von Knoten miteingeplant sind und DaSense diese selbstständig kompensiert. Defekte Knoten können im laufenden Betrieb ausgetauscht werden.
- Datenvielfalt: Automobile Daten sind hochkomplex. Das Spektrum reicht von einfachen zeitbasierten Messreihen, wie sie etwa aus elektronischen Steuergeräten ausgelesen werden, über komplexe Metadaten bis hin zu Sensorda-

ten, die von Kameras oder Radargeräten stammen. Im Gegensatz zu klassischen Datenspeichern werden diese Daten auch direkt der Analysesoftware zur Verfügung gestellt, ohne Umwege, etwa über Datenbankabfragen.
- Unveränderlichkeit: Big Data ist auf schnelles Lesen großer Datenmengen hin optimiert, was zur Folge hat, dass Schreibvorgänge sehr teuer sind. Somit gilt: Sind Daten einmal im Data Lake angekommen, dann werden sie nicht mehr modifiziert.

Data Lakes basieren auf sogenannten Clusterdateisystemen, die auf technischer Ebene die obigen Punkte implementieren. Hier sind insbesondere das Hadoop-spezifische Dateisystem HDFS [2] und die kommerzielle Lösung MapR-FS [3] zu erwähnen, die beide von DaSense unterstützt werden.

Ein weiterer wichtiger Punkt ist die Integration von Data Lakes in Enterprise-Sicherheitsarchitekturen. DaSense ist ausgelegt auf eine Single-Point-of-Access-Architektur und bietet eine durchgehende Integration von Kerberos.

MAPREDUCE

Die Beschaffenheit eines Data Lakes erfordert eine neue Sichtweise auf die Art und Weise, wie Daten analysiert und verarbeitet werden. Das am weitesten verbreitete und sehr bekannte Verfahren ist „MapReduce“, das 2004 von Google vorgestellt wurde [4]. Das grundlegende Prinzip ist hier, die einfache Datenanalyse in zwei Schritte einzuteilen: Das „Mapping“ führt eine beliebige Anzahl von Analysen parallel auf dem Cluster aus, deren Ergebnisse dann im „Reduce“-Schritt zusammengeführt werden und das Analyseresultat liefern.

In einem vereinfachten Beispiel könnte es für die Entwicklung von Getriebeelektronik interessant sein, was die typische Dauer eines Kickdowns bei Automatikgetrieben einer bestimmten Baureihe ist. Die Messdaten, die über einen längeren Zeitraum aus Fahrzeugen ausgelesen wurden, liegen als große Ansammlung von Dateien auf einem Cluster vor, etwa im MDF4-Format (in der Praxis werden diese Daten als Erstes in ein für die Verarbeitung auf dem Cluster besser geeignetes Datenformat konvertiert). Ein einzelner Mapping-Prozess ist in der Lage, eine MDF4-Datei zu lesen und die entsprechenden Kanäle nach Kick-down-Ereignissen zu durch-suchen und ihre Dauer zu ermitteln.

Der Mapping-Schritt als Ganzes besteht aus der parallelen Ausführung möglichst vieler Prozesse, bei der alle vorliegenden Dateien nach Kick-downs durchsucht werden. Jeder Mapping-Prozess liefert als Ergebnis eine Statistik über die Dauer von Kick-downs in einer einzelnen Datei zurück, die dann im Reduce-Schritt zu einer Gesamtstatistik zusammengeführt werden.

Eine Fahrzeugflotte erzeugt über einen Zeitraum hinweg Messdaten, die beispielsweise im MDF4- oder ATFX-Format auf dem Cluster liegen, wie in **BILD 3** beschrieben. In dieser vereinfachten Darstellung werden im Mapping-Schritt viele Map-Prozesse parallel ausgeführt, die jeweils eine Datei analysieren und die gewünschte Information extrahieren. Im Reduce-Schritt werden diese Ergebnisse zusammengeführt, und dem Benutzer als Gesamtergebnis zur Verfügung gestellt.

Ausgehend von MapReduce gibt es inzwischen auch allgemeinere und erheblich flexiblere Verfahren zur Datenanalyse auf dem Cluster wie Spark [5], das ebenfalls in DaSense zum Einsatz kommt.

BIG DATA IN DER AUTOMOBILENTWICKLUNG

DaSense bringt die Vorteile und Stärken der Big-Data-Technologien in die Automobilentwicklung und übernimmt alle notwendigen Anpassungen. **BILD 4** zeigt die vielfältigen Möglichkeiten zur clusterbasierten Datenanalyse mit DaSense. Die einfache Integration in die bestehenden Arbeits- und Entwicklungsvorgänge beschreiben die folgenden drei Aspekte:
- Automobilspezifische Datenformate: Sehr viele Werkzeuge in der automobilen Datenverarbeitung sind auf die Verarbeitung von standardisierten Datenformaten ausgelegt (wie etwa MDF oder ADTF). Um solche Werkzeuge in ein Big-Data-Umfeld zu integrieren, bietet DaSense Schnittstellen für solche Formate an. Diese ermöglichen insbesondere die Übertragung von speziellen Datenformaten in Big-Data-Formate, die für die parallele Verarbeitung optimal geeignet sind.
- Suche in Messdaten: Suchen in automobilen Messdaten unterscheiden sich deutlich von Suchanfragen, wie man sie etwa an Internetsuchmaschinen stellt. So sind viele der Daten sehr komplex strukturiert und weisen einen hohen Verknüpfungsgrad auf, da sie die internen Zustände komplizierter Maschinen widerspiegeln. Oft ist der Grad der Verknüpfung a priori unbekannt, und es ist Ziel der Suche, Zusammenhänge in Messdaten festzustellen. DaSense bietet eine komfortable Benutzeroberfläche, mit deren Hilfe sich eine Vielzahl von Suchkriterien verknüpfen lässt. Je nach Art der Suche liefert DaSense die Ergebnisse als interaktive Grafiken, oder es ermöglicht anspruchsvolle statistische Auswertungen der Suchergebnisse, **BILD 5**.
- Mobiles Messdatenmanagement: Da einzelne Testfahrten bereits große Mengen an Daten erzeugen, müssen Testfahrzeuge diese Daten häufig entladen. Der zunehmende Datenaustausch zwischen Testfahrzeugen und Rechenzentren schränkt somit zu einem gewissen Grad die Auswahl der verfügbaren Teststrecken ein. Um hier abzuhelfen, hat NorCom eine mobile Lösung im Programm. Die Messdatenmanagement (MDM)-Box ist ein mobiler Cluster, der voll in die DaSense-Architektur integriert ist. Diese Box kann schnell an einen Einsatzort gebracht und über das Internet an die globale DaSense-Clusterstruktur angebunden werden. Entwickler, die in der Regel nicht direkt vor Ort sind, können nun ihre Algorithmen zeitnah an den mobilen Cluster übermitteln, um die frisch gewonnenen Daten zu analysieren. Somit ist es auch von entlegenen Standorten aus möglich, Erkenntnisse aus Analysen umgehend in Tests und standortübergreifende Auswertungen einfließen zu lassen.

DASENSE-ENTWICKLUNGSPLATTFORM

Mit DaSense bietet NorCom eine Plattform der nächsten Generation für die Big-Data-Algorithmenentwicklung in der Automobilindustrie an. **BILD 6** zeigt schematisch die wichtigsten Bestandteile der DaSense-Entwicklungsumgebung. Wie bereits erwähnt bildet „Hadoop“ das Fundament, das für die Datenspeicherung und die effizi-

BILD 4 Möglichkeiten zur clusterbasierten Datenanalyse (© NorCom)

BILD 5 Suchmaske zur Ereignissuche: Der Benutzer kann hier Messkanäle und Bedingungen angeben, die ein Ereignis beschreiben; diese Bedingungen können außerdem logisch verknüpft werden; DaSense startet anhand dieser Bedingungen eine parallele Suche auf dem Cluster und stellt die zurückgelieferten Ergebnisse dem Benutzer zur Verfügung; oben befinden sich Statusinformationen zur Clusterauslastung (© NorCom)

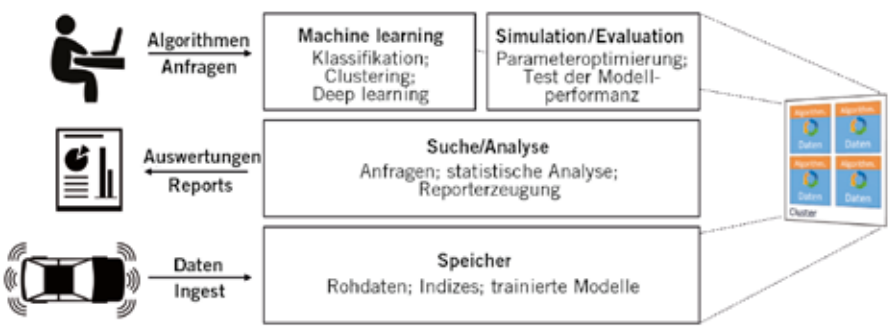

BILD 6 Die DaSense-Entwicklungsplattform: Aufbauend auf Big-Data-Technologien stellt DaSense vielfältige Funktionalitäten speziell für das Umfeld der Automobilentwicklung zur Verfügung (© NorCom)

ente Datenverarbeitung auf Clustern zuständig ist. Darauf aufbauend bietet die Basisfunktionalität von DaSense die Suche und Analyse von automobilen Daten, mit der sich bereits ein großer Teil der praxisrelevanten Analyseaufgaben abdecken lassen.

Darüber hinaus bietet DaSense auch eine Programmierschnittstelle zur unterliegenden Spark-Architektur, die es den Entwicklern erlaubt, DaSense um beliebige eigene Big-Data-Anwendungen zu ergänzen. Zur Unterstützung bietet DaSense hier APIs, die den Zugriff auf automotive Daten unterstützen, wie zum Beispiel eine domänenspezifische Sprache zum Umgang mit Zeitreihen, **BILD 7**. Eine weitere Funktionalität von DaSense sind grafische Schnittstellen, mit denen sich Analyseschritte zu komplexen Workflows verknüpfen lassen, **BILD 8**.

Auf dieser flexiblen und mächtigen Grundlage können dann die modernsten Methoden der clusterbasierten Datenanalyse für ihren Einsatz im Bereich der Automobilentwicklung aufbauen.

VON ANALYSEN ZU SIMULATIONEN

Bei der Entwicklung von Systemen für hochautonomes Fahren spielen Simulationen eine wichtige Rolle. Diese basieren auf Daten, die in Testfahrten gesammelt werden. Hierbei fallen große Mengen an sensorischen Informationen an, die beispielsweise von Kameras oder Lidar kommen.

Auch hier stehen Entwickler vor dem Problem, dass im herkömmlichen Ablauf immer nur ein Bruchteil der insgesamt eingefahrenen Daten zur Verfügung steht und der Datenaustausch zwischen Fahrzeug und Simulationsrechner ein langwieriger Vorgang ist. Zwei Beispiele zeigen, wie DaSense auch in solchen Entwicklungsabläufen eingesetzt werden kann.

Im ersten Beispiel wird die Ereignissuche in DaSense dazu genutzt, um die für die Simulation benötigten Datenmenge zu reduzieren. Im zweiten Beispiel zeigen wir, dass sich das Prinzip „Algorithmen zu den Daten“ auch auf komplexe Simulationssoftware übertragen lässt.

BEISPIEL: SOFTWAREABSICHERUNG FÜR UNFALLERKENNUNGSSYSTEME

In diesem Beispiel werden für die Entwicklung eines Unfallerkennungssystems in Testfahrten Daten im ADTF-Format mit einer Rate von über 2 GB pro Minute erzeugt. Um das Ereignis „Unfallerkennung“ zu simulieren, wird in der Regel nur der kleine Teil der Daten benötigt, der jeweils kurz vor und nach dem Ereignis eingefahren wurde. Die Aufgabe ist es, diese Ereignisse zu identifizieren und die relevanten Zeitabschnitte aus den gesamten Messdaten zu extrahieren.

Im ersten Schritt des DaSense Workflows werden die ADTF-Dateien auf das Cluster übertragen. Diese enthalten unter anderem Fahrzeugbus-Daten (Bus Traces) im Flexray-Format, in denen die Ereignisse gefunden werden können. Dazu werden die Bus Traces gesondert in das Big-Data-Format „Parquet“ übersetzt, das besonders zur Filterung und Suche in Datensätzen optimiert ist.

Im zweiten Schritt erstellt der Benutzer eine DaSense-Suchabfrage, in der er Bedingungen formuliert, mit deren Hilfe in den Bus Traces ein Unfallereignis identifiziert werden kann. Damit wird eine Suche auf dem Cluster angestoßen, die die relevanten Zeitintervalle zurückliefert. Der letzte Schritt besteht aus der parallelisierten Extraktion genau der Daten, die in den zuvor gefundenen Zeitintervallen aufgezeichnet wurden. Die auf diese Weise um eine Größenordnung verkleinerten Datensätze können jetzt an die Simulation weitergereicht werden.

BILD 7 Mit der DaSense-domänenspezifischen Sprache für Zeitreihenanalysen steht eine komfortable Programmierschnittstelle zur Verfügung, die die Verarbeitung von Zeitreihen erleichtert; derzeit basiert diese Schnittstelle auf Python; für zukünftige Versionen ist auch die Unterstützung von R und Scala geplant (© NorCom)

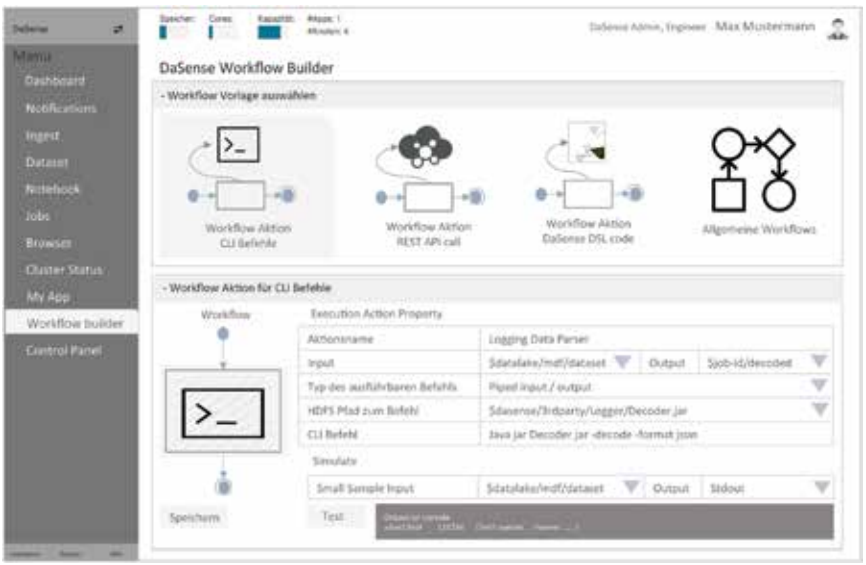

BILD 8 Komplexe Analysen bestehen aus einer Vielzahl von auf einander aufbauenden Schritten; ein einzelner Schritt kann sich einer eingebauten Funktionalität bedienen, externe Werkzeuge zur Anwendung bringen oder sich der Programmierschnittstelle bedienen; mit DaSense lassen sich beliebige Workflows zusammenstellen (© NorCom)

BEISPIEL: PARALLELISIERTE SIMULATION

Ein typisches Vorgehen im Entwicklungszyklus ist es, einen Algorithmus anhand einer Sequenz von Testfahrten zu testen, die gewisse Gemeinsamkeiten aufweisen, etwa Regen oder Bäume am Straßenrand. Diese Tests werden sequenziell durchgeführt, was, je nach Anzahl und Länge der Testfahrten, mehrere Stunden in Anspruch nehmen kann. Konzeptionell ist leicht zu sehen, wie im Prinzip sich solche sequenziellen Simulationsdurchläufe mithilfe von DaSense parallelisieren lassen: Eine Simulation lässt sich einfach als Mapping-Prozess auffassen, der im Cluster auf eine beliebige Testsequenz angewendet werden kann.

In der Praxis jedoch ergeben sich zunächst einige Schwierigkeiten, die vor allem daher kommen, dass die Simulationssoftware sehr komplex und ressourcenhungrig ist, was für einen Mapper in den meisten MapReduce-Anwendungen untypisch ist. Für solche Fälle stellt DaSense einen auf Docker [6] basierenden Container-Service zur Verfügung. Nach der Einbettung der Software in einen Container verhalten sich die Simulationen aus Sicht von DaSense wie ein einzelner Prozess und lassen sich problemlos parallelisieren. Über Details zu dieser DaSense-Anwendung wurde in [7] berichtet.

LITERATURHINWEISE

[1] Online: http://hadoop.apache.org

[2] Ghemawat, S.; Gobioff, H.; Leung, S.-T.: The Google File System. 19th ACM Symposium on Operating Systems Principles, Lake George, NY, 2003

[3] Online: https://www.mapr.com

[4] Dean, J.; Ghemawat, S.; MapReduce: Simplified Data Processing on large Clusters, OSDI'04: Sixth Symposium on Operating System Design, 2004

[5] Zaharia, M.; Chowdhury, M.; Franklin, M. J.; Shenker, S.; Stoica, I.; Spark: Cluster Computing with Working Sets, HotCloud'10 Proceedings of the 2nd USENIX conference on Hot topics in cloud computing, 2010

[6] Online: https://www.docker.com

[7] Abthoff, T.; Horakh, T.; et al.: Accelerate and Industrialize the Development of HAD Algorithms Using Big Data Technologies. Aachener Kolloquium, 2016