CIplus
Band 4/2017

Trinkwasser-Sicherheit mit Predictive Analytics und Oracle

Thomas Bartz-Beielstein, Steffen Moritz, Jan Strohschein, Dimitri Gross, Ralf Seger

Technology
Arts Sciences
TH Köln

Trinkwasser-Sicherheit mit Predictive Analytics und Oracle

Prof. Thomas Bartz-Beielstein, Steffen Moritz und Jan Strohschein, Technische Hochschule Köln, sowie Dimitri Gross und Ralf Seger, OPITZ CONSULTING GmbH

Verunreinigungen im Wassernetz können weite Teile der Bevölkerung unmittelbar gefährden. Gefahrenpotenziale bestehen dabei nicht nur durch mögliche kriminelle Handlungen und terroristische Anschläge. Auch Betriebsstörungen, Systemfehler und Naturkatastrophen können zu Verunreinigungen führen.

Um die Auswirkungen von unbeabsichtigten und vorsätzlichen Kontaminationen des Trinkwassers so gering wie möglich zu halten, arbeiten Wasserversorger, Forschungsinstitute und private Unternehmen gemeinsam an Schutzkonzepten. Dabei setzen die Versorger zum Beispiel sogenannte „Event Detection Systems" ein und installieren Online-Sensorik zur Überwachung der Wasserqualität an verschiedenen Positionen im gesamten Trinkwassernetz. Für die Ermittlung von Anomalien werden meist Verfahren des maschinellen Lernens eingesetzt. Der erste Teil des Artikels gibt einen Einblick in den Aufbau des Event-Detection-Systems für die Trinkwasserversorgung, während im zweiten Teil aufgezeigt ist, wie sich die Anforderungen an Big Data und Predictive Analytics mit Oracle-Software umsetzen lassen.

Trinkwassernetze stehen unter besonderer Aufmerksamkeit. Sie gehören zu den Ein-

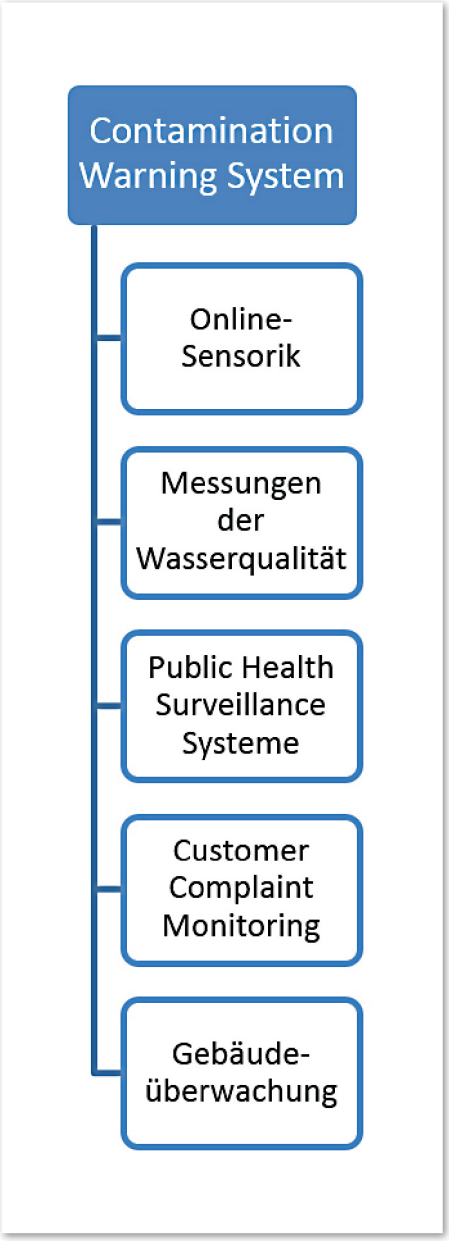

Abbildung 1: Die Bestandteile eines Contamination-Warning-Systems (CWS)

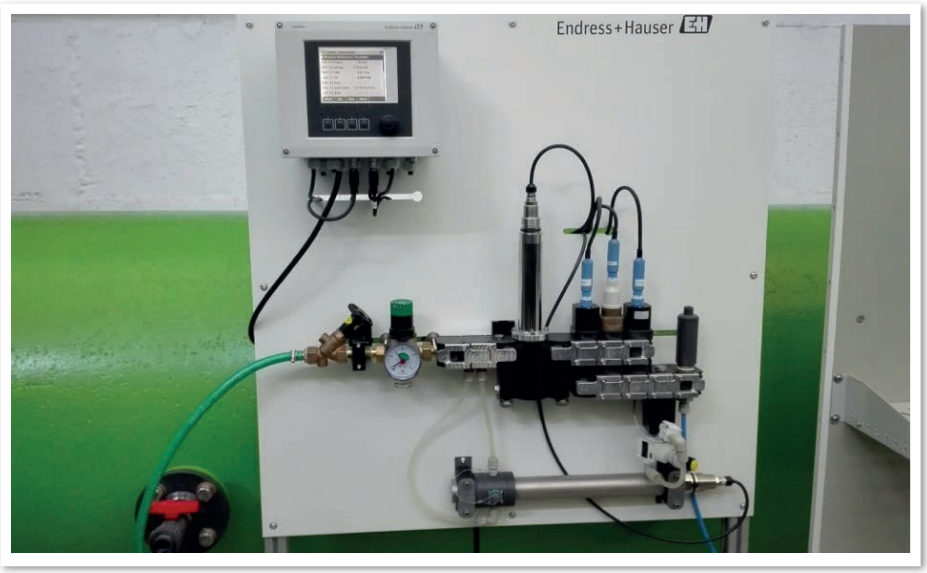

Abbildung 2: Sensorpanel mit Sensoren für Redox-Potenzial, Leitfähigkeit, Chlordioxid, Temperatur, pH-Wert und Trübung

richtungen mit zentraler Bedeutung für das staatliche Gemeinwesen, bei deren Ausfall oder Beeinträchtigung nachhaltig wirkende Versorgungsengpässe, erhebliche Störungen der öffentlichen Sicherheit oder andere dramatische Folgen eintreten. Die öffentliche Wasserversorgung gehört daher neben Sektoren wie Energie und Gesundheit sowie Informationstechnik oder Telekommunikation zu den sogenannten „kritischen Infrastrukturen" [1].

In aktuellen Forschungsvorhaben arbeiten Wasserversorger, Forschungsinstitute und private Unternehmen gemeinsam daran, die Auswirkungen von unbeabsichtigten und vorsätzlichen Kontaminationen des Trinkwassers so gering wie möglich zu halten. Insbesondere die amerikanische Umweltschutzbehörde EPA nimmt hier eine Vorreiterrolle ein und treibt das Thema in mehreren Forschungsprojekten voran. Auch in Deutschland gibt und gab es bereits mehrere Forschungsprojekte dazu; unter anderem beschäftigt sich die Arbeitsgruppe SPOTSeven der TH Köln im Verbundprojekt „IMProvT" in Zusammenarbeit mit mehreren Wasserversorgern und dem Sensorhersteller E+H Conducta mit der rechtzeitigen Erkennung von Qualitätsbeeinträchtigungen im Trinkwasser.

Ein vielversprechender Ansatz ist der Einsatz von Event-Detection-Systemen (EDS). Diese analysieren die anfallenden großen Datenvolumina der Wasserqualitäts-Sensoren und sollen Anomalien zeitnah erkennen. Weil die Parameter der Wasserqualität bereits im Normalbetrieb stark schwanken, ist dies keine einfache Aufgabe. Machine-Learning-Konzepte mit Modellen wie neuronalen Netzen oder Support Vector Machines spielen dabei eine wichtige Rolle.

Das Contamination-Warning-System

Um die Auswirkungen von Verunreinigungen so gering wie möglich zu halten, ist eine zügige Einleitung von Gegenmaßnahmen essenziell. Dies setzt voraus, dass die Kontaminierung bereits wenige Minuten nach deren Auftreten entdeckt wird. Dafür wurde das Konzept des Contamination-Warning-Systems (CWS) entwickelt. Es bezeichnet das Zusammenspiel verschiedener proaktiver Überwachungs- und Kontrolltechnologien [2]. Daten aus verschiedenen Bereichen und Systemen wie Online-Sensorik zur Messung der Wasserqualität, Kundenbeschwerden, Labormessungen oder Gebäudeüberwachung werden gesammelt und analysiert *(siehe Abbildung 1)*.

Momentane CWS-Ansätze sind allerdings noch sehr aufwändig. Neben den großen Mengen von anfallenden Daten, die verarbeitet werden müssen, ist vor allem die eingesetzte Sensorik teuer und wartungsintensiv. So belaufen sich die Anschaffungskosten für eine einzelne Messstation im Wasserverteilungsnetz bereits auf mehrere Tausend Euro. Auf die Online-Sensorik zu verzichten, ist trotzdem nicht ratsam, denn durch die teilweise mehrmals pro Minute durchgeführten Online-Messungen ist es möglich, Beeinträchtigungen der Wasserqualität schon aufzuzeichnen, bevor das Wasser den Kunden erreicht.

Das reine Aufzeichnen der Daten ist allerdings nur eine Seite der Medaille. Zusätzlich ist noch ein Event-Detection-System (EDS) nötig, um die Daten auszuwerten, automatisiert Anomalien zu erkennen und gegebenenfalls Alarm zu schlagen [4]. Da die Parameter bereits im Normalbetrieb stark schwanken, ist dies keineswegs trivial. Auch Änderungen der Betriebsparameter, wie die Menge des abgegebenen Wassers, können zu ähnlichen Mustern führen wie tatsächliche Qualitätsbeeinträchtigungen. Das EDS muss also einerseits große Datenmengen verarbeiten, andererseits aber die Ergebnisse möglichst zeitnah bereitstellen. Damit stellt es den wichtigsten und anspruchsvollsten Bestandteil eines CWS dar.

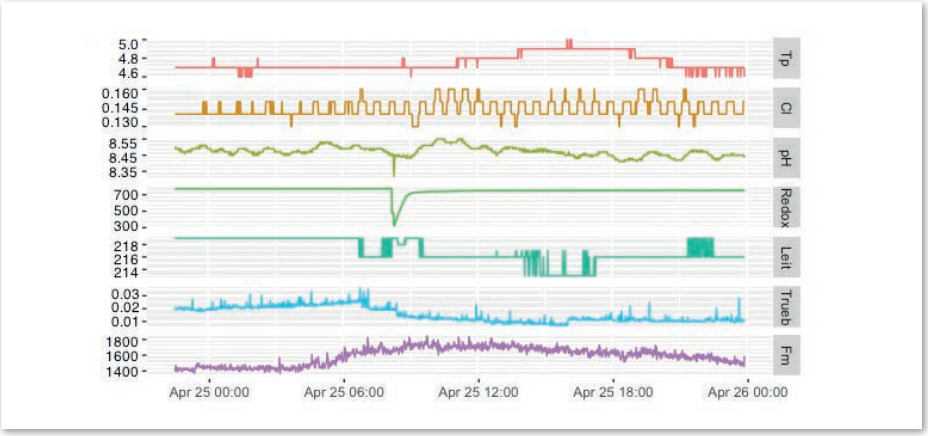

Abbildung 3: Beispielhafte Zeitreihe über 24 Stunden mit Messwerten für Temperatur, Chlordioxid, pH-Wert, Redox-Potenzial, Leitfähigkeit, Trübung und Wasserabgabe-Menge

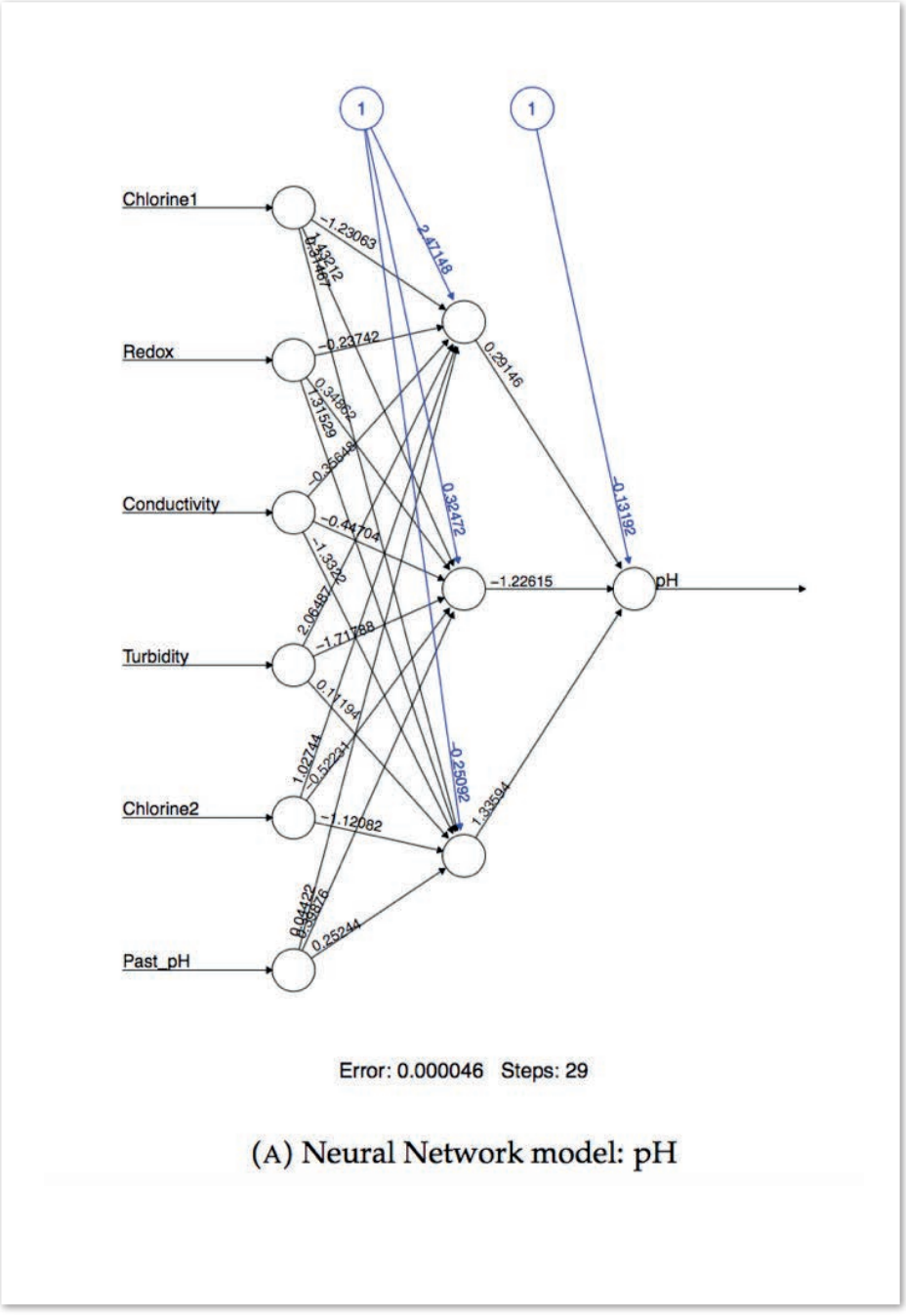

Abbildung 4: Neural Network Model: pH-Wert

Online-Sensorik für die Wasserqualität

Nicht alle Qualitätsparameter, die für eine Analyse sinnvoll sind, können auch wirklich online gemessen werden. Bakterienbelastungen etwa lassen sich nur im Labor oder mit Tests vor Ort zuverlässig ermitteln. Dieser Umstand erschwert die Erkennung von Events, weil die Parameter, die gemessen werden, oftmals nur indirekt auf die Parameter hinweisen, die tatsächlich relevant sind *(siehe Abbildung 2)*.

Zu den typischen Parametern, die für die Detektion von Qualitätsbeeinträchtigungen relevant sind und die auch online gemessen werden können, gehören Temperatur, Chlordioxid, pH-Wert, Leitfähigkeit, Trübung und Redox-Potenzial [3]. *Abbildung 3* zeigt einen Ausschnitt aus den Daten einer Messstation.

Grundsätzlich können Anomalien durch die Erhöhung der Sensoren im Netz zuverlässiger und schneller detektiert werden. Da die Sensorpanels allerdings relativ teuer sind und auch regelmäßig gewartet werden müssen, muss im Betrieb ein Kompromiss zwischen Wirtschaftlichkeit und Anzahl der Sensoren eingegangen werden.

Die regelmäßige Wartung und Nachkalibrierung der Sensorik ist hierbei wichtig. Wird sie vernachlässigt, erhöhen verfälschte Messungen die Anzahl der Fehlalarme, die durch das EDS ausgegeben werden. Die Tatsache, dass ein Stromanschluss erforderlich ist und die Datenübertragung sichergestellt sein muss, reduziert die Anzahl möglicher Sensor-Standorte zusätzlich. Im Sensorpanel in *Abbildung 2* ist zu sehen, dass die Sensoren keineswegs klein sind und dementsprechend Raum benötigen. Bevorzugte Standorte für die Sensorik sind deshalb Hochbehälter, die einerseits an besonders wichtigen Punkten im Netz stehen und zum anderen auch die für die Infrastruktur benötigten Voraussetzungen mitbringen.

Das Event-Detection-System

Das Event-Detection-System (EDS) dient der Erkennung von Auffälligkeiten in den Daten, die mittels Online-Sensorik aufgenommen wurden. Dazu werden Streaming-Daten verarbeitet [5]. Es ist wichtig, nahezu alle Kontaminationsereignisse zu erkennen und gleichzeitig die Anzahl der Fehlalarme möglichst gering zu halten. Realisiert wird die Detektion von Auffälligkeiten im EDS meist in einem zweistufigen Prozess [4].

Im ersten Schritt wird ein zukünftiger Wasserqualitätswert vorhergesagt. Diese Vorhersage basiert in der Regel auf kurz zu-

vor aufgenommenen Werten. Dafür kommen unterschiedliche Methoden zum Einsatz, etwa modellbasierte Vorhersagen mit neuronalen Netzen oder Support Vector Machines. *Abbildung 4* zeigt beispielhaft ein Modell mit einem neuronalen Netz für den Parameter „pH-Wert". Input sind die pH-Werte aus der jüngeren Vergangenheit und die Werte weiterer Sensoren. In einem zweiten Schritt werden die berechneten Vorhersagen mit den tatsächlich eintretenden Werten verglichen, sobald diese verfügbar sind.
Die Entscheidung, ob eine Anomalie vorliegt, hängt von der Differenz zwischen dem vorhergesagten Wert („Prediction") und dem tatsächlich eingetretenen Wert („Observation") ab sowie davon, wie sich diese Differenz über die vergangenen Minuten entwickelt hat. Einzelne Ausreißer in sonst unauffälligen Daten führen hierbei nicht unbedingt direkt zu einem Alarm, denn sie sind oftmals auf Sensorfehler zurückzuführen. Erst wenn mehrere Messwerte hintereinander signifikant von der Prognose abweichen, erfolgt eine Meldung.

Praktische Umsetzung

Die Überwachung der Werte, die das Sensorpanel im CWS erhebt *(siehe Abbildung 2)*, ist eine typische Streaming-Problematik. Messdaten werden kontinuierlich im CWS erfasst, versendet und wollen verarbeitet sein. Einzelne Beobachtungen könnten auf ein reales Problem hindeuten, allerdings sind in der Praxis einige Fallstricke zu bewältigen, die bei einem idealisierten Labor-Stream nicht anzutreffen sind:

- Fehler und Ausfall von Sensoren
- Störungen wie Zeitversatz im Sende-Netzwerk

Obwohl man schon beim eintreffenden Datenstrom mit Fehlern rechnen muss, wird die gesamte Datenmenge für die Analyse-Pipeline genutzt. Ein persistenter Datenpuffer am Anfang jeder Stream-Verarbeitung stellt sicher, dass die Messdaten nicht verloren gehen, sollte ein technischer Fehler bei der Überwachung auftreten. Eine skalierbare Lösung bietet beispielsweise Apache Kafka *(siehe „https://kafka.apache.org")*. Ein Kafka-Consumer kopiert die eintreffenden Werte in ein Emergency Topic und kann so auf einzelne extreme Sensordaten reagieren.

Im Beispiel in *Abbildung 5* liefert das CWS die Sensordaten für Redox-Potenzial, Leitfähigkeit, Chlordioxid, Temperatur, pH-Wert

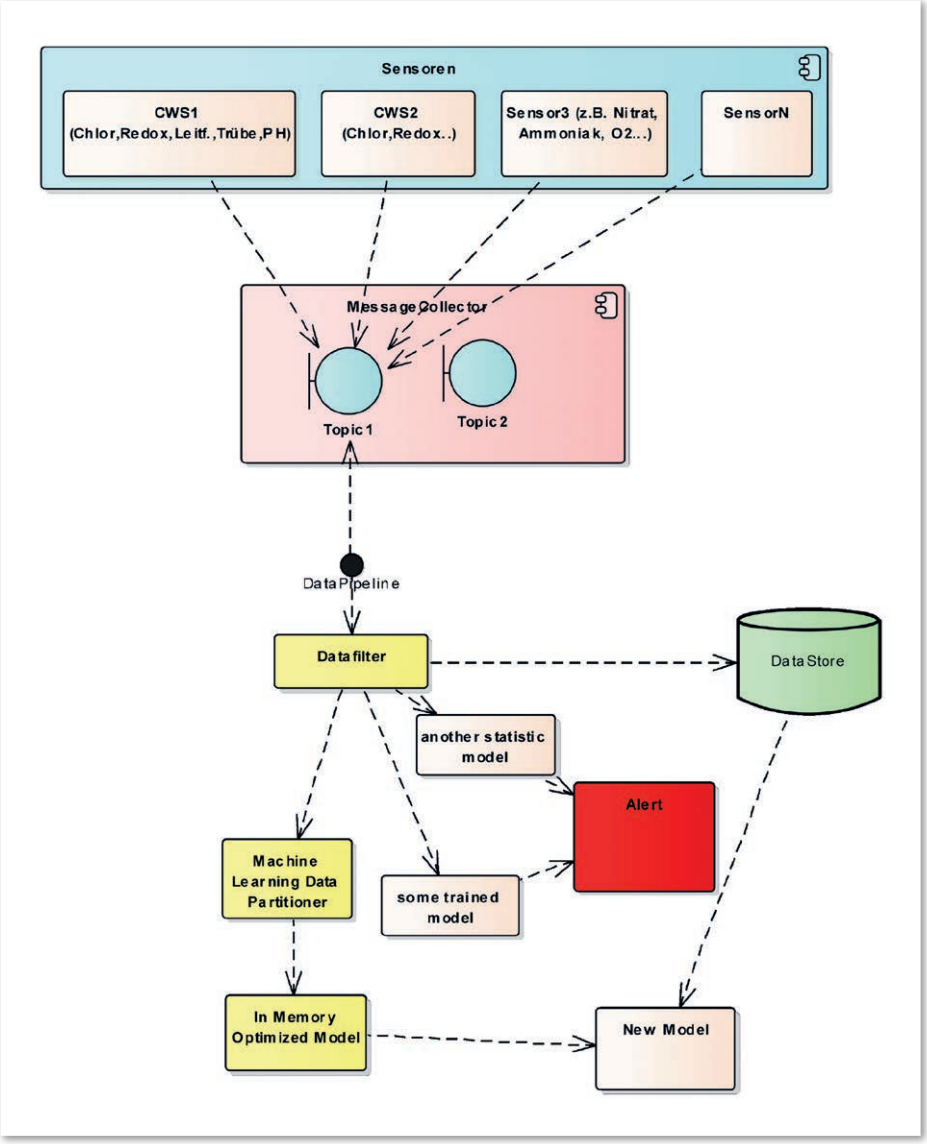

Abbildung 5: Stream-Processing-Architektur mit Einbindung mehrerer CWS-Sensorpanels

und Trübung. Eine Anbindung weiterer Datenquellen ist jederzeit möglich. Die Daten werden entweder aktiv vom Sensor an den MessageCollector gepusht oder die Message Collector-Komponente holt sich die Daten in bestimmten Zeitintervallen.
Auf Architektur-Ebene entkoppelt das System diese technische Abhängigkeit und liefert der nachfolgenden Verarbeitungspipeline (Einstiegspunkt „DataPipeline") eine neutrale Schnittstelle. Zusätzlich stellt eine hochskalierbare Message Queue wie Apache Kafka die wiederholbare Verarbeitung sicher und verhindert das sogenannte „Information Flooding".

Message Queues werden schon lange in Enterprise-Anwendungen eingesetzt, um dort unter anderem als Puffer Daten sicher von der Quelle an ihr Ziel zu bringen. Apache Kafka hat außer der herausragenden Performance noch weitere positive Eigenschaften: Die Daten lassen sich für eine vorkonfigurierbare Zeit zwischenspeichern. Außerdem werden bereits abgeholte Daten nicht automatisch gelöscht. Ein Client kann also jederzeit wieder auf Originaldaten zugreifen. Der Consumer entscheidet, welche Daten er verwenden möchte.

Dem Consumer bieten sich mehrere Optionen an. Eine echte Stream-Verarbeitung kann zum Beispiel mit Apache Storm *(siehe „http://storm.apache.org")* oder Apache Ignite *(siehe „https://ignite.apache.org")* erfolgen. Um Modelle zu trainieren, kann er im sogenannten „Micro-Batch-Betrieb" arbeiten. Im Kontrast zu einem beständigen Datenstrom, der einzelne Tupel an die Topologie sendet, werden beim Micro-Batching mehrere Tupel auf einmal versendet. Die Menge ist meistens über ein Zeitintervall festgelegt. Möglich ist bei manchen Frameworks auch eine fixe Anzahl.

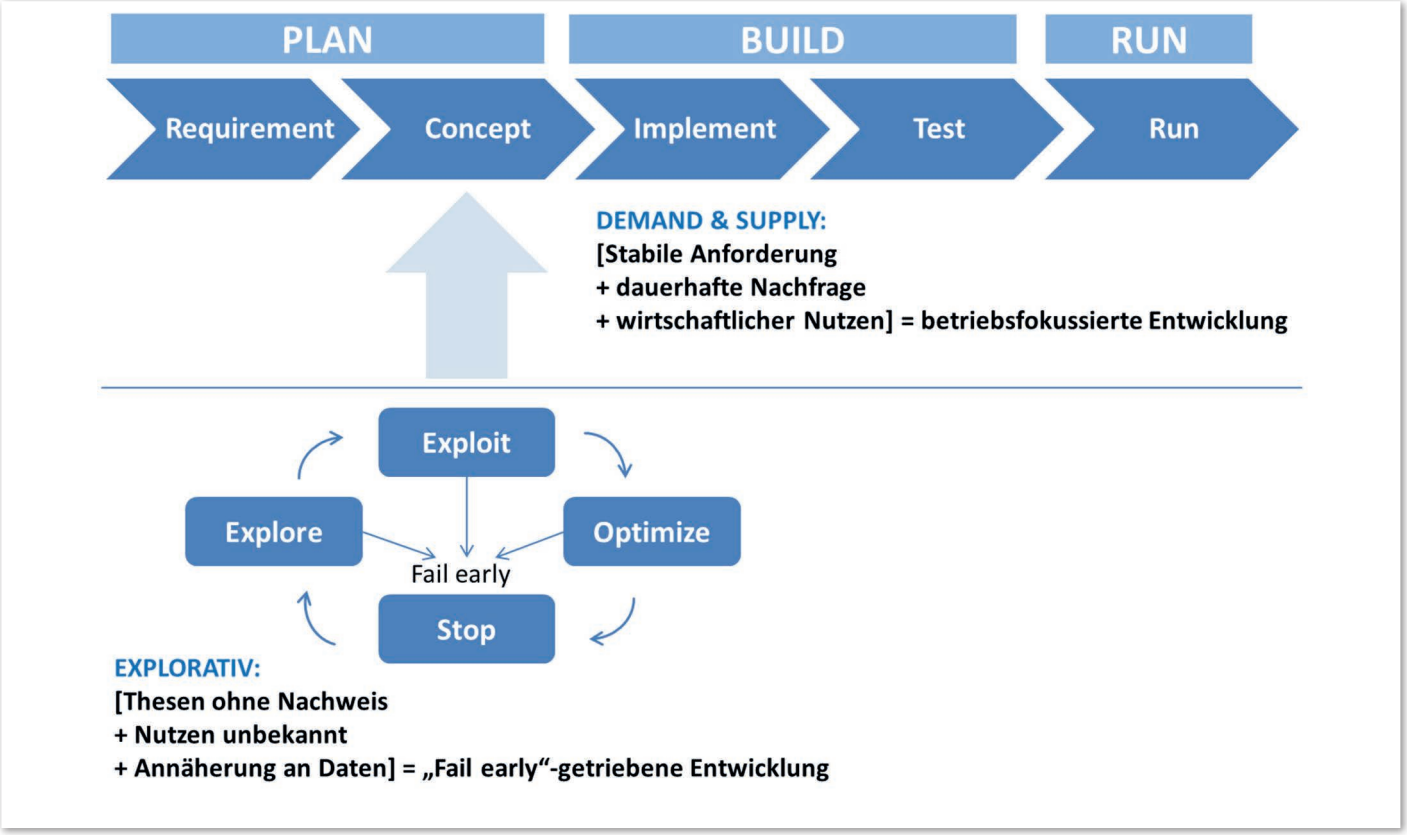

Abbildung 6: „Demand & Supply" vs. explorative Vorgehensweise

Direkt von Kafka mit Daten betanken lassen sich Alternativen wie das auf Storm aufsetzende Trident *(siehe „http://storm.apache.org/releases/1.0.1/Trident-tutorial.html")*, Apache Flink *(siehe „https://flink.apache.org")* oder Apache Spark *(siehe „http://spark.apache.org").* Jedes dieser Frameworks bietet auch eine Integration für Machine Learning Libraries. Im Projekt zur Trinkwasser-Analyse bei der TH Köln kommt das Spark-Streaming zum Einsatz, vor allem aufgrund der größeren Verbreitung und weil die Zeitfenster-Strategien für diesen Use Case ausreichen. In anderen Anwendungen kann Flink mit seinen variableren Zeitfenstern die bessere Wahl darstellen.

Die in *Abbildung 5* dargestellte Verarbeitungspipeline beginnt mit einer Filteroperation. Dieser Filter dient dazu, klare Messfehler aus der Verarbeitungspipeline fernzuhalten. Natürlich sollte dafür bereits Wissen über die Sensorbereiche vorhanden sein. Die ungefilterten Rohdaten fließen deshalb in einen Data Store. Spark unterstützt den schnellen Datentransfer zu einigen NoSQL-Datenbanken wie HBase, Cassandra oder Accumulo.

Der Data Store in *Abbildung 5* liefert die Trainingsdaten für die interaktive Modellierung. Das sogenannte „Supervised Learning" nimmt einen beträchtlichen zeitlichen Aufwand in Anspruch. Allerdings wird durch exploratives Vorgehen außer Referenzmodellen auch Verständnis für Daten generiert. Um mehr Modelle als Kurz- oder Langzeitgedächtnis zu trainieren, sind die Messwerte noch partitioniert. Mit diesen werden „Machine Learner" trainiert. Die Partitionierung dient nicht nur der schnelleren Performance, sondern hält auch einen Teil der Daten zurück, um die Modellgenauigkeit zu validieren.

Alle so gewonnenen Modelle können zur Alarmierung eingesetzt werden. Aber woran erkennt das System besondere Events, die einen Alarm auslösen können? Welche dieser Events sind relevant? Sensorfehler und fehlende Daten reduzieren die Aussagekraft der gemessenen Daten. Fehlende Messdaten liefern keine valide Aussage über den Zustand des zu beobachtenden Systems. Da immer Zeitfenster betrachtet werden, die eine Zeitreihe von Sensordaten liefern, können einzelne Missing Values ignoriert werden. Ein leeres Zeitfenster ist mit dem Totalausfall eines Sensors gleichzusetzen und zieht als Eskalation eine Reparatur beziehungsweise den Austausch des Sensors nach sich.

Events, die aufgrund ungewöhnlicher Messdaten ausgegeben werden, lassen sich mit den oben erwähnten statistischen Modellen prüfen. Bei klassischen Whitebox-Modellen (wie „lm" oder „glm") lassen sich Konfidenz-Intervalle angeben. Liegen mehrere Messwerte außerhalb dieses Konfidenzbereichs, liegt ein weiterer Alarmierungsfall vor. Bei Black-Box-Modellen (neuronalen Netzen) grenzt man üblicherweise den zulässigen Wertebereich ein. Beobachtungen außerhalb dieser Grenzen werden auf den nächsten Grenzwert normiert [7].

Was noch zu beachten ist

Soweit die Technik. Doch wie sieht es in der Produktion aus? Welche Faktoren können ein Big-Data-Vorhaben zum Scheitern bringen? Das sind Fragen, die viele Unternehmen, die Big Data erproben und in die Produktion überführen möchten, vergessen. Typische Gründe für das Scheitern von Big-Data-Vorhaben sind erfahrungsgemäß:

- Falsche Erwartungen
- Zeitliche Verzögerungen bei der Entscheidungsfindung
- Zu starke Divergenz bei Technologien und vorhandenen Skills

Diese drei Schlüsselaspekte sollten vor dem Start eines Big-Data-Vorhabens geklärt werden. So empfiehlt sich beispielsweise Lean Startup [6] als methodische Vorgehensweise.

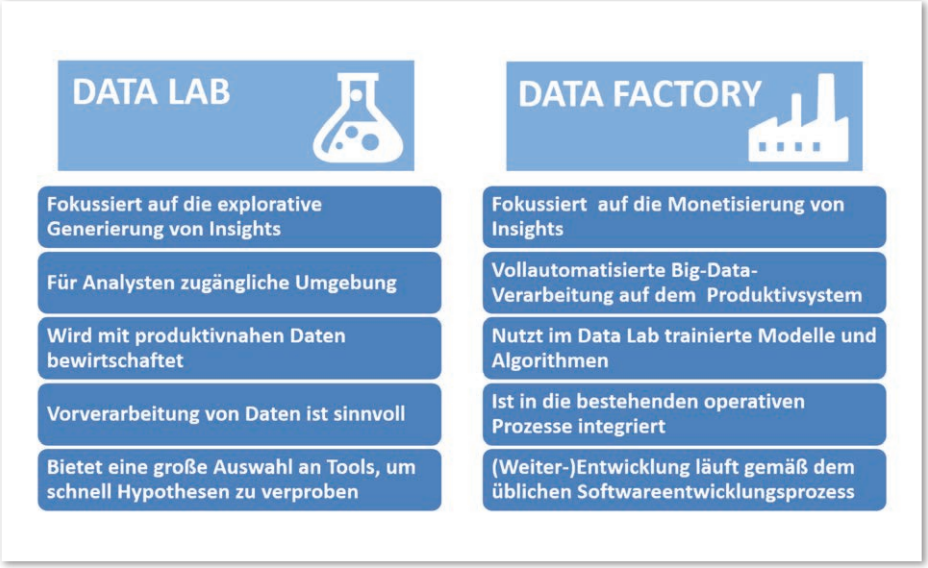

Abbildung 7: Kernprozesse in Data Lab & Data Factory

Es kann beim Management das Bewusstsein dafür schärfen, dass eine Hypothese auch ein negatives Resultat liefern kann.

Zudem ist es wichtig, sich mit organisatorischen Fragen auseinanderzusetzen. Es gibt häufig Unternehmen, die erst angesichts des drohenden Scheiterns eines Pilotprojekts eine organisatorische Anpassung überprüfen und angehen. Wenn das Management erst zu diesem Zeitpunkt feststellt, dass eine etablierte, jedoch rigide Organisationsform (Plan-Build-Run) nicht zu einem hoch agilen Thema wie Big Data passt, kann ein Umschwenken sehr teuer werden.

Rigide Abläufe, lange Zyklen im Anforderungsmanagement und fehlende Möglichkeiten, ein dediziertes Team für Big-Data-Vorhaben zusammenzustellen, lassen ein Projekt scheitern. Letztendlich wären damit also bürokratische Hürden mit zu vielen Schnittstellen der wesentliche Grund. Nur ein Schritt in Richtung einer virtuellen Organisation kann hier bereits Abhilfe schaffen. Die Aufbau-Organisation ist dabei ein individuelles Thema und richtet sich immer an die Gesamtstruktur des Unternehmens. Auch das Gewicht des Big-Data-Vorhabens für die Gesamtunternehmung spielt hier eine Rolle. Ein weiterer wichtiger Punkt sind die vorhandenen Skills und das Know-how zu neuen Technologien im Unternehmen. Der Prozess für den Skill-Aufbau kann parallel mit den organisatorischen Maßnahmen stattfinden. Ein weiterer Faktor, der den explorativen Prozess der Hypothesen-Erprobung beschleunigen kann, ist der Einsatz einer Big-Data-Distribution, die auf einer passenden Hardware läuft. Man stelle sich die Datenbewirtschaftung als zwei in sich greifende Prozesse vor. Der erste Prozess, der sich mit der explorativen Überprüfung von Hypothesen beschäftigt *(siehe Abbildung 6)*, wird in der Praxis oft als „Data Lab" bezeichnet. Dementsprechend definiert der zweite Prozess einen stabilen Produktivbetrieb und wird üblicherweise als „Data Factory" bezeichnet *(siehe Abbildung 7)*.

Ein Engineered System, in dem Hardware und Software kombiniert sind, bietet Oracle mit seiner Big Data Appliance an. Auf dieser mächtigen Hardware läuft Cloudera EDH mit den dazugehörigen Frameworks. Das System ermöglicht zum einen die prototypische Umsetzung und stellt zum anderen den Produktivbetrieb im Rahmen einer Data Factory sicher. Apache Zeppelin, das in der Cloudera EDH mitgeliefert wird, ermöglicht eine explorative Erprobung von Hypothesen, was wiederum gut zum Data-Lab-Konzept passt. In diesem Szenario arbeiten die Data Scientists an der Erprobung mathematischer Verfahren, während ein Team aus Software-Entwicklern und Integratoren die bereits validierten Hypothesen in die produktive Datenbewirtschaftung überführt.

Fazit

Neue Entwicklungen im Bereich der Sensortechnik, des High Performance Computing und der Daten-Analyse (Big Data, Deep Learning) ermöglichen Lösungen in Bereichen, die noch vor wenigen Jahren aufgrund von fehlenden Daten oder wegen der großen Berechnungskomplexität undenkbar gewesen wären. Die in diesem Artikel beschriebene Online-Trinkwasserüberwachung, die im Verbundprojekt „IMProvT" der TH Köln von der Arbeitsgruppe SPOTSeven *(siehe „www.spotseven.de")* entwickelt wird, stellt ein prominentes Beispiel dar.

Die verschiedenen Produkte der Apache Software Foundation erlauben eine skalierbare und ausfallsichere Verarbeitung von Big Data und insbesondere von Streaming-Daten. In Kombination mit neuen, erschwinglichen Cloud-Angeboten für die hochperformante Datenverarbeitung und -speicherung werden solche Projekte auch für große Datenmengen wirtschaftlich sinnvoll – für sensible Daten durch eine Oracle Cloud Machine sogar On-Premise im eigenen Datencenter. Die geringen Kosten erlauben es agilen Teams, neue Ansätze auszuprobieren, explorativ die wirtschaftlichen Möglichkeiten eines Vorhabens auszuloten und bei den Mitarbeitern wertvolle Fähigkeiten aufzubauen.

Literatur

[1] Bundesministerium des Innern, Nationale Strategie zum Schutz kritischer Infrastrukturen, 2009

[2] Roberson, J. Alan, Morley, Kevin M., Contamination Warning Systems for Water: An Approach for Providing Actionable Information to Decision-Makers, American Water Works Association, 2005

[3] Storey, Michael V., van der Gaag, Bram, Burns, Brendan P., Advances in On-line Drinking Water Quality monitoring and early warning systems, Water Research 45.2, 2011 (Seite 741 – 747)

[4] Murray, R., et al., Water Quality Event Detection Systems for Drinking Water Contamination Warning Systems – Development, Testing and Application of CANARY, 2010

[5] Bartz-Beielstein, T., Experimental Algorithmics Applied to On-line Machine Learning, in Papa, G. and Mernik, M., Bioinspired Optimization Methods and their Applications, 2016 (Seite 94 – 104)

[6] Ries, E., The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Crown Publishing, 2014

[7] Klein, B. D., Rossin, D. F., A Preliminary Analysis of Data Quality in Neural Networks, 1997

Dimitri Gross
dimitri.gross@opitz-consulting.com

Kontakt/Impressum

Diese Veröffentlichungen erscheinen im Rahmen der Schriftenreihe "CIplus". Alle Veröffentlichungen dieser Reihe können unter
https://cos.bibl.th-koeln.de/home
abgerufen werden.

Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.
Datum der Veröffentlichung: 06.06.2017

Herausgeber / Editorship

Prof. Dr. Thomas Bartz-Beielstein,
Prof. Dr. Wolfgang Konen,
Prof. Dr. Boris Naujoks,
Prof. Dr. Horst Stenzel
Institute of Computer Science,
Faculty of Computer Science and Engineering Science,
TH Köln,
Steinmüllerallee 1,
51643 Gummersbach
url: www.ciplus-research.de

Schriftleitung und Ansprechpartner/ Contact editor's office

Prof. Dr. Thomas Bartz-Beielstein,
Institute of Computer Science,
Faculty of Computer Science and Engineering Science,
TH Köln,
Steinmüllerallee 1, 51643 Gummersbach
phone: +49 2261 8196 6391
url: http://www.spotseven.de
eMail: thomas.bartz-beielstein@th-koeln.de

ISSN (online) 2194-2870

Technology
Arts Sciences
TH Köln

Supported by:

Federal Ministry
for Economic Affairs
and Energy

on the basis of a decision
by the German Bundestag

Grant No. 03ET1387A

Technology
Arts Sciences
TH Köln