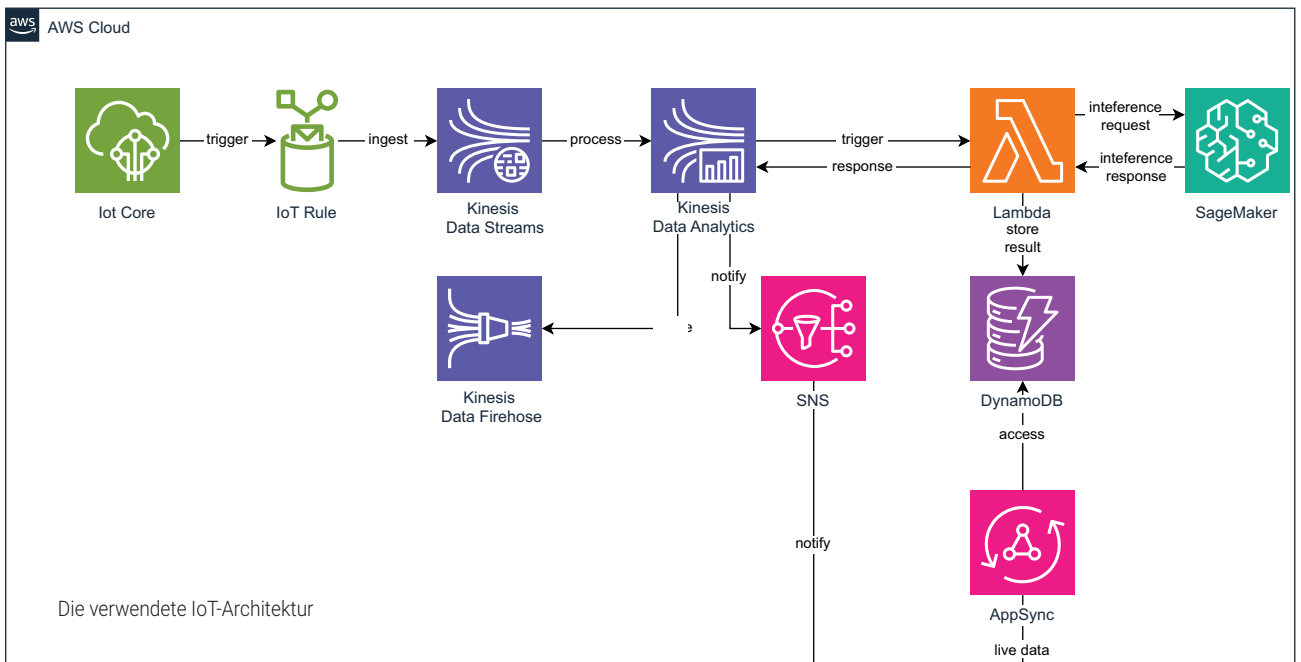


Mit der AWS-Cloud zur vernetzten Produktion (Teil 2/2)

Auf dem Hot-Data-Pfad zur Echtzeit-Datenverarbeitung



Der erste Teil dieser Artikelserie behandelt die Arbeitsschritte, um IoT-Daten in der Cloud entgegenzunehmen, sie aufzubereiten, zu speichern und erste Langzeitanalysen durchzuführen. Jetzt ist es an der Zeit, die Informationssysteme des Unternehmens mit den erfassten Daten anzureichern.

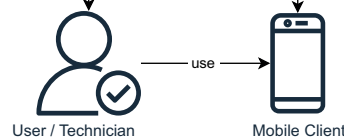
Wie im ersten Teil der Artikelreihe beschrieben, ermöglicht es die Umsetzung des Cold-Data-Pfades, Erkenntnisse aus Langzeitdaten-Analysen zu ziehen. Damit werden frühzeitige Steuerungsmaßnahmen möglich, die eine Minderung der Produktivität, -qualität oder gar einen Produktionsausfall verhindern können. In diesem Teil der Serie steht die Frage im Vordergrund, wie sich eingehende Live-Daten für ad-hoc Analysen, Alarme oder Visualisierungen nutzen lassen. Dazu bauen Anwenderunternehmen ein Stream Processing auf

der AWS-Cloud auf, mit dem Echtzeit-Einblicke in die Produktion gewonnen und diese anschließend gewinnbringend genutzt werden können.

Vereinfachter Anwendungsfall

Als Szenario dient ein stark vereinfachter Anwendungsfall aus der Instandhaltung: Im Beispiel geht es um ein Asset, das nur in einem bestimmten Temperatur- und Luftfeuchtigkeitskorridor Teile in der gewünschten Qualität produzieren kann. Erweiterte Analyseverfahren können diesen Korridor offenlegen. Nun

möchten die Produzenten frühzeitig eine Warnung an einen Instandhalter schicken, dass sich die Maschine außerhalb dieses Korridors bewegt, oder diesen bald verlassen wird. Zu diesem Zweck setzen sie eine Streaming-Architektur auf. Eine Streaming Architektur besteht immer aus drei Bestandteilen: dem **Event Producer**, dem **Event Router** und dem **Event Consumer**. Dabei veröffentlicht der Producer die Nachricht an den Router, wobei dieser die Daten filtern oder anreichern kann und liefert diese Daten wiederum an die Consumer aus. Dadurch können Producer und Consumer voneinander unabhängig skaliert werden. Um das Szenario umzusetzen – also Warnungen an die Instandhaltung zu schicken – so



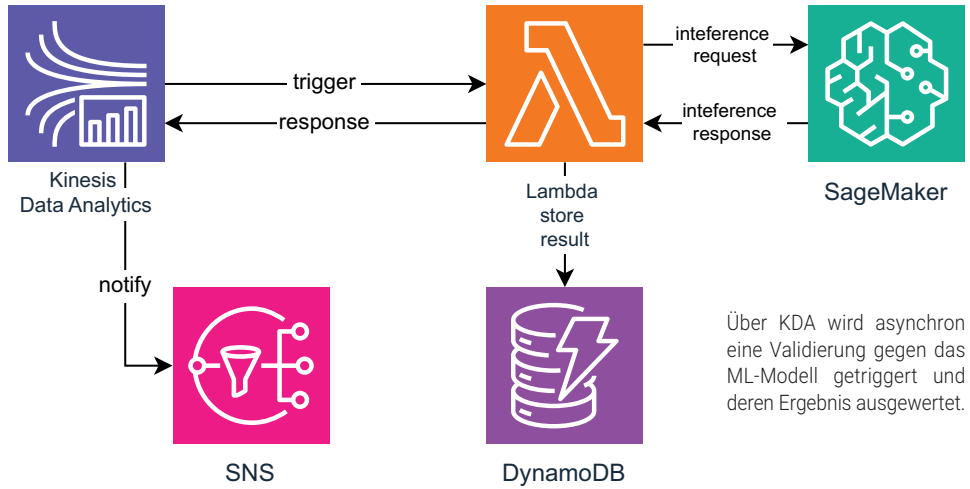
bald Luftfeuchtigkeit und Temperatur drohen ihren Korridor zu verlassen, lässt sich eine Architektur mit folgenden Komponenten aufbauen.

AWS-Software im Einsatz

In der Architektur fließen die Daten aus IoT Core direkt über eine IoT Rule in die Streaming-Anwendung Kinesis Data Streams, die von AWS verwaltet wird. Dabei wird kein zusätzlicher Code benötigt, da die Integration der AWS Services verlässlich ist. Innerhalb der Regel müssen Anwender lediglich einen Kinesis Data Stream angeben, auf den die Daten veröffentlicht werden sollen. Die tatsächliche Verarbeitung der Daten erfolgt in Kinesis Data Analytics (kurz KDA), wobei KDA hierbei das Hosting der vom Anwender als Docker Container bereitgestellten Apache-Flink-Applikation übernimmt. Apache Flink ist ein Framework zum Stream Processing, welches die asynchrone Filterung und die Anreicherung von Daten unterstützt. KDA übernimmt dabei die Skalierung und Überwachung der Applikation.

KI-gestütztes Monitoring

Von hier lässt sich eine asynchrone Anfrage an eine Lambda-Funktion anstoßen, die als Proxy für die Sagemaker Inference API dient, welche in diesem Fall mit einem Micro-Batch angefragt wird. Also wird für ein kleines Datenset angefragt, ob die aktuellen Daten darauf hindeuten, dass die Werte bereits außerhalb des Korridors liegen, oder zu verlassen drohen. Diese API wird genutzt, um die Live-Daten gegen das zuvor trainierte



Über KDA wird asynchron eine Validierung gegen das ML-Modell getriggert und deren Ergebnis ausgewertet.

Modelle zu validieren. In diesem Korridor von Luftfeuchtigkeit und Temperatur ist die Problemstellung nahezu trivial und könnte auch durch einfache Wenn-Dann-Verkettungen gelöst werden. Doch das Vorgehen ist auf weit komplexere Daten mit vielen Einflussvariablen übertragbar.

Daten für den Hot-Data-Pfad

Die um die Inferenz angereicherten Daten werden nun in einer DynamoDB gespeichert, welche nur den aktuellen Status jedes Assets enthält. Die längerfristige Betrachtung der Daten findet nicht im Hot-Data-Pfad statt. Gleichzeitig wird das ermittelte Ergebnis zurück nach KDA gegeben, wo abhängig von dem Ergebnis der Inferenz, eine Mitteilung im Simple Notification Service veröffentlicht wird, die darauf hinweist, dass die Messwerte außerhalb des gewünschten Korridors liegen. Diese Mitteilung können verschiedene Adressa-

ten verschickt werden. Diese reichen von anderen Applikationen, die das Topic abonnieren, bis hin zu dem automatischen Versand per E-Mail an eine abonnierende Adresse. So kann der Instandhalter per E-Mail oder über seine Service App direkt erfahren, dass das Asset bereits außerhalb der definierten Parameter läuft oder droht dies zu tun.

Der Live-Zustand der Assets

Die Instandhaltung kann über diese Service App den aktuellen Zustand der angebotenen Assets überprüfen. Dazu wird die Service App aus der zuvor angelegten DynamoDB Tabelle über App Sync in (nahezu) Echtzeit mit Daten zum Zustand des Assets versorgt. Dabei erfolgt die Anbindung über Standard GraphQL-Schnittstellen, die sich schnell neu entwickeln und integrieren lassen.

IoT-Plattform ist aufgebaut

Am Ende des Artikels ist nachvollziehbar, wie eine IoT-Plattform entsteht, die sowohl die langfristige Speicherung und Analyse (Artikel 1) unterstützt, als auch die Verarbeitung von Daten in nahezu Echtzeit, wie in diesem Artikel beschrieben. Zusammengefasst legen Anwenderunternehmen so ein tragfähiges Fundament für die Digitalisierung ihrer Produktion. ■

```

{
  "topicRulePayload": {
    "sql": "SELECT * FROM 'some/topic'",
    "ruleDisabled": false,
    "awsIotSqlVersion": "2016-03-23",
    "actions": [
      {
        "kinesis": {
          "streamName": "kinesis_stream",
          "partitionKey": "${topic()}",
          "roleArn": "arn:aws:iam:123456789012:role/aws_iam_kinesis"
        }
      }
    ]
  }
}
    
```

Beispiel für die Definition einer IoT Rule, die Daten Richtung Kinesis Data Stream weiterleitet.

Toni Spittel
Senior Business Analyst
Transition Technologies
PSC Germany GmbH
www.ttpsc.com