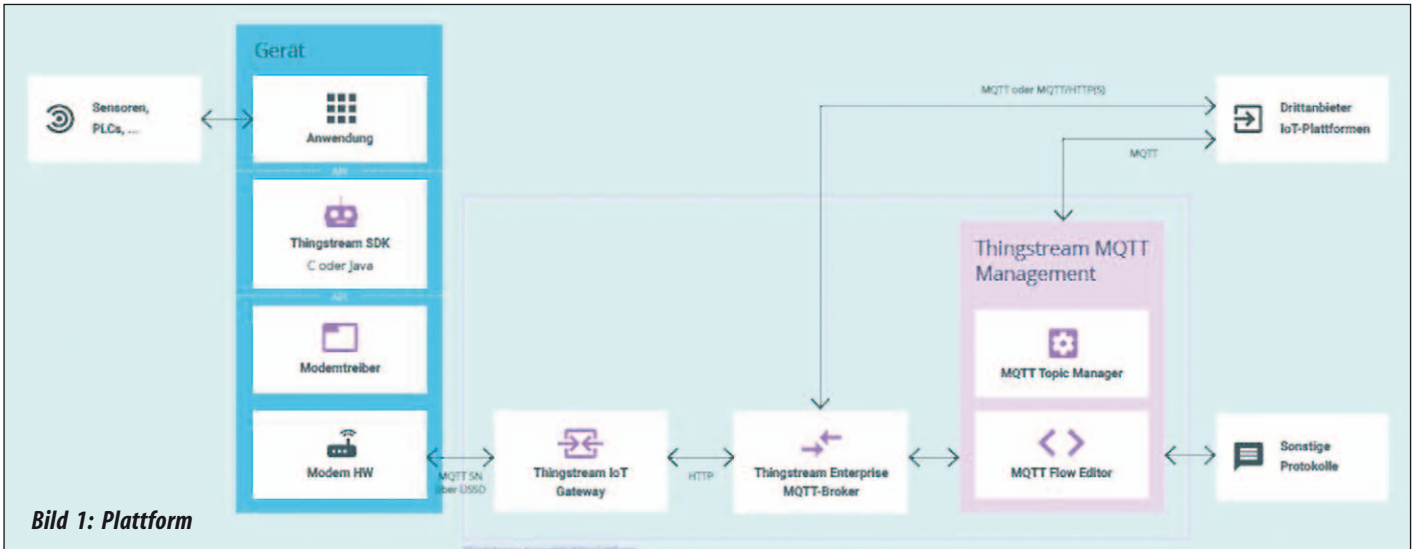


Entwicklungshilfe für das IoT



IoT-Projekte von Unternehmen werden derzeit vor allem durch teils erhebliche Einschränkungen bei der Netzabdeckung und Unterstützung von mobilen IoT-Komponenten ausgebremst. Ebenso gestaltet sich die Entwicklung von Hardware bis zur Einsatzreife als schwierig. Die kosteneffiziente Realisierung von IoT-Projekten ist nicht zuletzt von der Konnektivität abhängig. Die sogenannten LPWA- (Low Power Wide Area), also „Niedrigenergie-Weitverkehrs“-Netzwerke werden hier immer wichtiger. Die tatsächliche Verfügbarkeit und Nutzung von LPWA-Netzen in der Praxis ist jedoch derzeit noch stark begrenzt. Die meisten dieser Konnektivitätsangebote decken nur Ballungsregionen ab. Neben der Netzabdeckung gibt es zudem Unterschiede hinsichtlich Übertragungsrates und Stromverbrauch, aber auch Sicherheit, Skalierbarkeit und Interoperabilität.

Nicht-lizenzierte Technologien wie Sigfox, Ingenu und LoRa erfordern einen großangelegten Netzwerk-

Rollout auf nationaler und internationaler Ebene, was Zeit braucht. Lizenzierte LPWA-Varianten wie die zwei derzeit konkurrierenden Standards NB-IoT (Narrowband-IoT) und Cat-M1 nutzen das LTE-Mobilfunknetz, die grenzüberschreitende Interoperabilität ist jedoch noch nicht geklärt.

Zukunftsweisendes Konnektivitätskonzept für gängige IoT-Anforderungen

Etwas, das bei der Auswahl einer IoT-Konnektivitätslösung berücksichtigt werden soll, ist, wie viel Aufwand erforderlich ist, um die Daten von den „Dingen“ in ein sinnvolles Format zu bringen, damit sie von einer IoT-Plattform erfasst werden können. Darüber hinaus stellt sich die Frage, wie sich Servicequalität erzielen lässt, also sicherzustellen, dass Nachrichten von den „Dingen“ auch die zuverlässig die IoT-Plattform erreichen.

Das häufigste Protokoll für IoT-Netzwerkverkehr ist Message Queue Telemetry Transport for Sensor Net-

works (MQTT-SN), ein schlankes Publish-/Subscribe-Messaging-Protokoll für M2M-Telemetrie (Machine-to-Machine) in Umgebungen mit geringer Bandbreite entwickelt. MQTT-SN verfügt über integrierte Servicequalität (Quality of Service, QoS), die für IoT-Anwendungen wichtig ist. Es macht die Kommunikation in unzuverlässigen Netzwerken viel einfacher, weil das Protokoll die erneute Übertragung abwickelt und die Zustellung der Nachricht garantiert, unabhängig davon, wie unzuverlässig der zugrundeliegende Transport ist. Außerdem ermöglicht es MQTT-SN einem Client, den QoS-Level zu wählen, abhängig von seiner Netzwerksicherheit und Anwendungslogik.

Die Kombination aus kommunizierendem MQTT-SN über USSD (Unstructured Supplementary Service Data), ein Übermittlungsdienst für GSM-Netze, und dem globalen, nahezu flächendeckend verfügbaren GSM-Netz ist für gängige IoT-Anforderungen ein zukunftsweisendes Konzept. Damit steht erstmals kostengünstige IoT-Konnektivität mit internationaler Abdeckung und Mobilitätsunterstützung als End-to-End-Konnektivitätslösung bereit. Mit diesem Ansatz lassen sich viele der üblichen Schwierigkeiten bewältigen, mit denen Hersteller beim Aufbau verbundener IoT-Geräte konfrontiert sind, wie Kosten, Strom und Daten. So gelingt es, Geräte mit geringem Stromverbrauch zu bauen, die ein entgegen-

wärtiges, globales Netzwerk nutzen. Als Konnektivitätsbasis dient dabei das bewährte GSM-Netz, das in über 190 Ländern verfügbar ist, ohne dabei auf SMS oder eine Mobilfunk-Datenverbindung angewiesen zu sein. IoT-Endgeräte können sich mit praktisch jedem GSM-Netz weltweit verbinden. Über MQTT-Messaging können kleine Datenmengen schnell und äußerst zuverlässig zu geringen Kosten versendet werden. Unterstützt wird zudem uneingeschränkte bidirektionale Kommunikation. Eine derartige Technologie eignet sich für den Einsatz in vielen vertikalen Sektoren, einschließlich Asset-Tracking, Logistik, Supply Chain, Facility Management, Energie und Umweltüberwachung.

Konnektivität wird in der Anfangsphase von IoT-Projekten oft vernachlässigt

Ingenieure, die an IoT-Projekten beteiligt sind, sind nicht unbedingt von Anfang an bereits am Thema Konnektivität interessiert. Sie suchen normalerweise die perfekte Kombination von Halbleitern für eine Firmware-Anwendung, um damit eine elegante Anwendung zu schaffen. In diesen Fällen arbeiten zwei Disziplinen zusammen, ein Hardware-Spezialist und ein Firmwarespezialist. Zunächst wird mithilfe der gemeinsamen Fähigkeiten ein Prototyp-Format basierend auf Arduino, Raspberry Pi oder einer anderen Physical-Computing-Plattform erstellt.

Kurz gefasst

Der folgende Artikel beschreibt, wie eine Plattform auf Basis von MQTT/SN Entwicklern dabei hilft, kosteneffizient, schnell und komfortabel neue IoT-Anwendungen und -Endgeräte zu realisieren

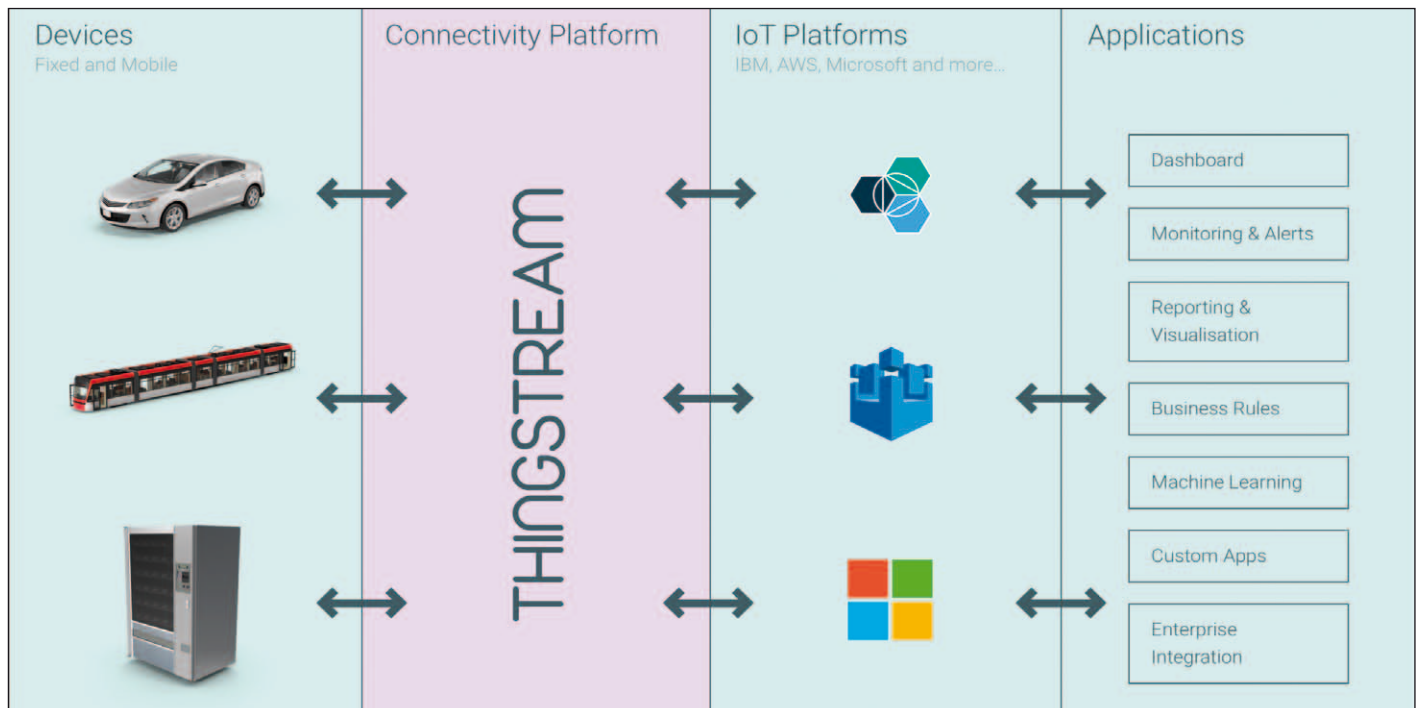


Bild 2: Architektur einer IIoT-Umgebung vom Endgerät über die Connectivity-Plattform bis hin zu den IoT-Plattformen von Drittanbietern und den damit verknüpften Applikationen

Erst in einer späteren Phase im Projekt fällt die Entscheidung, welche Art von Konnektivität für das Offloading der Daten und welche Frequenz verwendet werden sollte.

Dieses klassische Szenario ist nicht gerade ideal für IIoT-Projekte. Dies liegt daran, dass ein Prototyp mit einem kostengünstigen Standard-Bausatz erstellt wird, aber diese Kits typischerweise nicht für eine Massenproduktion geeignet sind. Um den „tatsächlichen“ Arbeitsprototypen in dem für den Feldeinsatz geeigneten Formfaktor mit geeigneten Halbleitern zu erstellen, muss die gesamte Hardware-Entwurfsphase erneut beginnen. Außerdem haben diese Kits oft wesentlich mehr Komponenten an Bord, die in der realen Hardware am Ende gar nicht benötigt werden.

Eine schlanke Plattform auf Basis von GSM und MQTT/SN via USSD für die IIoT-Kommunikation ermöglicht es Ingenieuren, Prototypen für das IIoT schneller zu entwickeln. Hierzu können sie Produktionskomponenten auf einer einfachen Leiterplatte verwenden, wodurch die Markteinführungszeit für eine End-to-End-Anwendung beschleunigt wird. IIoT-Firmware-Designer können sich somit darauf konzentrieren, die mit den notwendigen Sensoren verbundene Anwendung zu schreiben und

einfach die Anwendungslast zu aktivieren, um die Daten an der API an Bord des IIoT-Moduls zu „veröffentlichen“. IIoT-Entwickler müssen sich somit nicht mit dem Verständnis der zugrundeliegenden AT-Befehle (zum Konfigurieren und Parametrieren von Modems) für GPRS etc. herumzuschlagen. Da die Notwendigkeit entfällt, über das Innenleben des Modems Bescheid zu wissen, ist dies eine große Erleichterung. Die Entwicklungsingenieure gewinnen dadurch Zeit, um sich auf Anwendungsdesign und -tests zu konzentrieren.

IIoT-Entwickler müssen sich nicht um Carrier, Bitraten und Datentarife kümmern

Ein entsprechendes IIoT-Modul verfügt über einen leistungsstarken Anwendungsprozessor, der Anwendungen ausführen kann, die in einem gängigen Format wie MCUXpresso von NXP geschrieben wurden. Der entscheidende Vorteil eines solchen IIoT-Moduls ist, dass es in der Lage ist, MQTT zu jedem IIoT-Online-Ziel zu kommunizieren, sobald es aktiviert wird. An Bord ist eine SIM-Karte integriert, die sich mit dem stärksten GSM-Signal verbindet, das sie findet. Dadurch ist die Übertragung der Nutzdaten der Anwendung möglich, ohne dass eine Mobilfunkinfrastruktur

(GPRS, UMTS oder LTE) erforderlich ist. Damit steht auf einfache Weise genau das zur Verfügung, was sich IIoT-Entwickler wünschen: Sie wollen, dass die Konnektivität einfach erledigt wird, ohne sich um Carrier, Bitraten und Datentarife kümmern zu müssen. Das Abrufen der Daten von einem Gerät ist nur ein Teil des IIoT-Puzzles. In den meisten IIoT-Designszenarien sind die Entwickler, die die Hardware und Firmware erstellen, nicht für die Cloud-Seite der IIoT-Anwendung verantwortlich oder darin involviert. Dies stellt ein erhebliches Problem dar, das IIoT-Projekte ebenfalls verlangsamt. Angesichts der Tatsache, dass die Firmware-Entwickler genau wissen, welche Daten tatsächlich ein Gerät verlassen, ist es sinnvoll, dass sie die Hand über den Datenfluss behalten vom Ereignis „Publish“ bis zum Ereignis „Subscribe“. So kann ein Hardwaredesigner ein Prototypgerät nahe am realen Endprodukt erstellen und einen Prototyp-Datenfluss in einer Umgebung in der Art von Node-RED abbilden. Die Übernahme in die industrielle Produktionsumgebung erfolgt dann einfach per Knopfdruck.

Einfaches Testen

Hardwareingenieure haben in der Regel keine Möglichkeit, um

Anwendungsfälle, in denen ein Gerät in realen Szenarien kommunizieren muss, einfach zu testen. Genau dafür ist die beschriebene IIoT-Entwicklungsplattform konzipiert. Es ist ziemlich einfach, eine serverseitige MQTT-„Subscribe“- und „Publish“-Umgebung zu emulieren, um zu testen, wie das Gerät arbeitet, wenn es Daten von einer serverseitigen Anwendung sowohl in Prototyp- als auch in Produktionsumgebungen empfängt. Hardwareingenieure können so bisherige Softwarebarrieren überwinden und die IIoT-Designphase innerhalb kürzester Zeit erfolgreich zum Abschluss bringen.



*Autor:
Neil Hamilton,
VP Business Development bei
Thingstream
thingstream.io*