DZone

THE 2018 DZONE GUIDE TO

Internet of Things

HARNESSING DEVICE DATA

VOLUME V

BROUGHT TO YOU IN PARTNERSHIP WITH

Couchbase

Digi-Key ELECTRONICS

FLEXERA

influxdata

outsystems

Dear Reader,

IoT is the silent world-changer. Unless you're actively using it, either as a developer or a consumer, you probably don't think much about it.

But the behind-the-scenes growth and adoption of the Internet of Things has been truly staggering. There are now more connected devices on the planet than people according to Gartner, and that number will only explode as the usefulness of IoT is further realized.

Perhaps IoT's greatest benefit is bringing more data than ever before to bear so people can make better decisions — or automate them altogether. IoT data can help recognize trends common in machinery that signify a breakdown is imminent. It can put deals in the hands of store customers, saving time and money while also securing their loyalty. And it's on the way to creating the cars and cities of the future, where urban optimization is baked into the start of the planning process.

And that's not even mentioning the slew of partner technologies that can help IoT really take root. On the conference circuit, blockchain is considered an essential component to creating connected and secure supply chains. Meanwhile, AI has roared to life over the past couple of years, bringing new analytics capabilities to IoT's already considerable support for decision making.

Of course, despite IoT's usefulness, it's fascinating to see how little has truly changed in the ecosystem. Platforms have come and gone, and organizations have more finely tuned various protocols like MQTT and AMQP to make them even more useful, but an increasingly familiar set of problems remains.

IoT device and data security is still a top concern — and a top barrier to wider adoption. Meanwhile, a lack of standardization has made it difficult to bring disparate devices and their data together, further limiting IoT's impact. And there's also a talent shortage that's slowing growth as well.

But those challenges have merely slowed IoT down, not stopped it. That's why it's the silent world-changer — it might not be as flashy or as well-understood by the public as other technologies, but its impact is just as real.

That's why this guide will provide you with the knowledge you need to efficiently and securely use sensors to gather data. Meanwhile, we'll also show you how the burgeoning field of AI is helping IoT create the next great wave of technology, and how that technology will filter down into everyone's increasingly connected lives.

Get ready to help change the world.

By Mike Gates

SR. CONTENT COORDINATOR., DZONE

DZONE IS...

PRODUCTION
CHRIS SMITH
DIR. OF PRODUCTION

ANDRE POWELL
SR. PRODUCTION COORD.

G. RYAN SPAIN
PRODUCTION COORD.

ASHLEY SLATE
DESIGN DIR.

BILLY DAVIS
PRODUCTION ASSISSTANT

TOM MARTIN
SR. ACCOUNT MGR.

JASON BUDDAY
ACCOUNT MGR.

MICHAELA LICARI
ACCOUNT MGR.

MARKETING
LAUREN CURATOLA
MARKETING SPECIALIST

KRISTEN PAGÀN
MARKETING SPECIALIST

JULIAN MORRIS
MARKETING ASSOCIATE

SALES
CHRIS BRUMFIELD
SALES MANAGER

FERAS ABDEL
SALES MANAGER

ALEX CRAFTS
DIR. OF MAJOR ACCOUNTS

JIM HOWARD
SR. ACCOUNT EXECUTIVE

JIM DYER
SR. ACCOUNT EXECUTIVE

ANDREW BARKER
SR. ACCOUNT EXECUTIVE

BRIAN ANDERSON
ACCOUNT EXECUTIVE

BUSINESS AND PRODUCT
RICK ROSS, CEO

MATT SCHMIDT, PRESIDENT

JESSE DAVIS, EVP, TECHNOLOGY

KELLET ATKINSON, MEDIA PRODUCT MANAGER

EDITORIAL
CAITLIN CANDELMO
DIR. OF CONTENT & COMMUNITY

MATT WERNER
PUBLICATIONS COORD.

SARAH DAVIS
PUBLICATIONS ASSOCIATE

MICHAEL THARRINGTON
CONTENT & COMMUNITY MGR. II

KARA PHELPS
CONTENT & COMMUNITY MGR.

TOM SMITH
RESEARCH ANALYST

MIKE GATES
SR. CONTENT COORD.

JORDAN BAKER
CONTENT COORD.

ANNE MARIE GLEN
CONTENT COORD.

ANDRE LEE-MOYE
CONTENT COORD.

Executive Summary

BY MATT WERNER
PUBLICATIONS COORDINATOR, DZONE

The Internet of Things has been at the center of an equally large number of hype cycles and possible disastrous futures. For all who look at the potential of technology to improve industrial processes, city life, and decision-making abilities, there are those who point to security issues, device reliability, and abuse by corporations and governments, some of which is already happening. Regardless, IoT is here to stay, and those who are experimenting with its potential are finding that the challenges are tied to properly utilizing big data and securing their devices. To find out how IoT was impacting the careers of developers and tech professionals, we asked 522 DZone members for their thoughts and plans for the future.

NEW AND OLD CHALLENGES

Data: Between 2017 and 2018, there was a 9% decrease in challenges with device reliability and a 6% decrease in dealing with both power and latency issues. A new answer choice in 2018, device security, is the greatest IoT challenge, according to 52% of respondents.

Implications: This data is corroborated by additional survey data that states that security is the biggest concern for the IoT space according to 71% of respondents. While strides have been made in power consumption and network latency, ensuring that devices and their data are secure is still a major challenge.

Recommendations: Invest in devices that make tampering difficult, and make sure to secure not just devices, but also data in transit. One technology that's beginning to make waves as more than just the mechanism behind bitcoin is blockchain, which can allow for secure transfer of data or contracts that let all parties involved in the transaction know if it has been tampered with. For a fun explanation on how Blockchain, IoT, and AI all intersect, see our infographic, "Sailing on the Sea of Data."

IOT POWERS THE CITIES OF THE PRESENT

Data: 20% of respondents live in areas that have implemented smart city projects, while 29% do not know if their city has done so. Of those, 68% of cities have traffic sensors installed, 41% have parking sensors, and 36% have climate monitoring and air quality sensors.

Implications: Smart cities are not yet widespread, but those that have begun adopting projects are focusing first on transportation optimization, which can boost or sink a city's reputation as being an enjoyable play to live. After that are more environmental concerns.

Recommendations: Encouraging your city or town to pursue smart city initiatives is a small but important first step to growing the industry. In particular, encourage an open data initiative so more people can interact with a city's plan and suggest new solutions. When doing so, focus on ROI for both the city and for taxpayers. For another look at smart cities and some of the challenges that can arise, particularly in working with the taxpayers, check out Cate Lawrence's article, "Smart Cities — Who Wins and Who Loses?"

FUTURE IOT

Data: 41% of respondents are planning on adopting a new IoT technology in 2018, compared to 23% in 2017. Of those, 61% plan on adopting an AI/ML tool, 54% plan on adopting sensors, and 50% are considering an IoT development platform.

Implications: IoT adoption is growing among developers, but with AI/ML being the most popular technology being adopted, it's clear that there are issues with fast, effective data analysis that need to be addressed.

Recommendations: The key to unlocking the potential of IoT is using the data that sensors collect to help businesses and users make more effective decisions. Since data is collected at an enormous rate, AI/ML will be necessary to quickly analyze and aggregate this data into useful insights. While it's still a relatively new field, similar to IoT, there is a wealth of information on neural networks, algorithms, theory, and software. For some introductory material into the world of artificial intelligence and machine learning, check out our **Guide** and **Zone** on the subject.

Key Research Findings

BY G. RYAN SPAIN
PRODUCTION COORDINATOR, DZONE

DEMOGRAPHICS

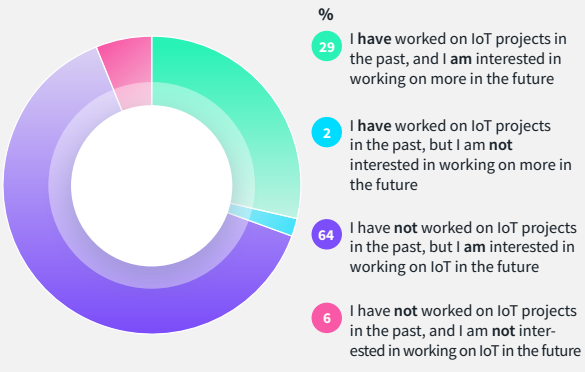
522 software professionals completed DZone's 2018 IoT survey. Respondent demographics are as follows:

- 39% of respondents identify as developers or engineers, 17% identify as developer team leads, and 12% identify as software architects.
- The average respondent has 14 years of experience as an IT professional. 52% of respondents have 10 years of experience or more; 18% have 20 years or more.
- 40% of respondents work at companies headquartered in Europe; 28% work in companies headquartered in North America.
- 19% of respondents work at organizations with more than 10,000 employees; 20% work at organizations between 1,000 and 10,000 employees; and 19% work at organizations between 100 and 1,000 employees.
- 78% develop web applications or services; 43% develop enterprise business apps; and 26% develop native mobile applications.
- 79% work at companies using the Java ecosystem; 60% at companies that use client-side JavaScript; 40% at companies that use Python; and 35% at companies that use server-side JavaScript. 55% of respondents use Java as their primary language at work.

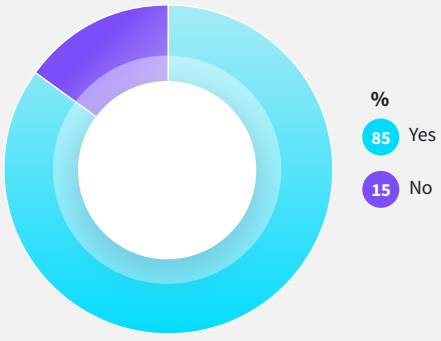
LOSING THE SHEEN

It's no doubt that connected Things will continue to gain prominence in the IT landscape as technologies continue to advance. In the consumer world, IoT devices continue to grow in popularity as more and more products come out that use network connectivity to simplify daily tasks like turning on the lights or seeing what's in the fridge. For developers, though, the idea of IoT is losing the sheen it has had for the past several years, especially regarding its relevance in developers' organizations.

GRAPH 01. How would you describe your relationship with IoT?



GRAPH 02. Do you believe IoT will be relevant to your organization in the future?



Our IoT survey in 2018 showed a 9% drop in respondents who said they are interested in IoT projects within their organization, while IoT interests in other contexts (hobby/personal development, building a start up, consumer or industrial products) remained consistent (variations of 1-3% year over year, within the survey's margin of error); we also saw a 10% swing in respondents who consider IoT to be currently relevant to their organization (65% yes and 35% no in 2017; 60% yes and 40% no in 2018).

Industry certainly has some impact here: respondents working in software vendor or finance/banking industries were less likely than average to believe IoT is currently relevent to their organization–by 4% and 18%, respectively–representing respective drops of 14% and 11% from respondents within those demographics in 2017. On the other hand, respondents in healthcare/pharmaceutical (10% above average) or telecom/cable/utilities (26% above average) industries were much more likely than average to believe IoT is relevent to their organization, representing respective increases of 5% and 15% from last year's respondents within those demographics. So while the overall feeling of "relevance" of IoT to many organizations may be decreasing, its benefits in certain use cases seems to be growing more apparent.

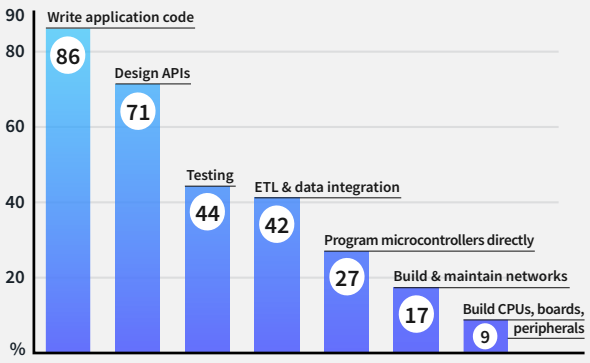
TECHNOLOGY ADOPTION

Overall interest in working with IoT has remained mainly static between 2017 and 2018. Respondents saying that they have worked on IoT projects in the past and are interested in working on more in the future (29%), and those saying they have not worked on any IoT projects before, but are nonetheless interested in working on IoT in the future (64%), changed only a fraction of a percent from last year's results, showing no significant difference year over year. Interestingly, however, respondents who said they are planning on adopting a new IoT technology over the next six months increased dramatically–an 18% year-over-year increase, or a 36% swing, closing the gap between affirmative and negative responses to 18%. Among the related technologies these respondents plan to adopt, AI/ML was the most popular category (61% among respondents planning to adopt), followed by sensors (54%) and IoT development platforms (50%).

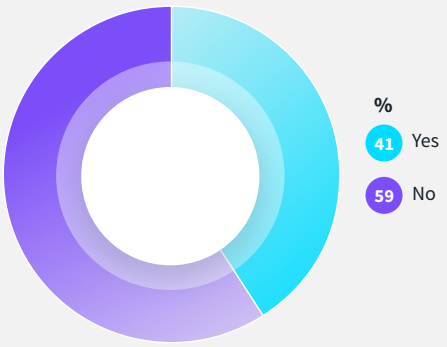
CHALLENGES

71% of respondents to this year's IoT survey are "very concerned" about security in an IoT context, in line with the 74% we saw in last year's survey. When asking respondents who have experience with IoT development about challenges in this area, we added the choice "device security" to this year's survey, and it became the most popular challenge among responses, with 52% of IoT-experienced respondents saying they experience challenges with security. The respondents saying that unpredictable physical environments are a challenge in IoT development remained about the same in 2018 (48%) as it was in 2017 (46%). Other challenges, however, had significant decreases; challenges with unreliable data in IoT contexts fell 9% from last year (26% to 17%); and there was a 6% drop in responses claiming challenges in power constraints (44% to 38%), latency (38% to 32%), and device unreliability (43% to 37%). As such, it seems that as enhancements in hardware technology allow for faster, more energy efficient, closer to lossless connections between connected machines, finding adequate security for these devices is another issue entirely, with–potentially–more serious ramifications.

GRAPH 03. What task(s) are you likely to perform while developing an IoT product?



GRAPH 04. Are you planning to adopt a new IoT-related technology in the next 6 months?



Introduction to IoT Sensors

BY FRANCESCO AZZOLA
IT ARCHITECT

QUICK VIEW

01. A sensor is a device that can detect events or changes in the environment. It reacts to the changes of a physical property that it measures.

02. A good sensor must be sensitive to the property that it measures while also being independent from other physical properties.

03. Sensors can be connected to prototyping boards to acquire data from the environment.

04. There are several types of sensors with different behaviors (Passive or Active, Digital or Analog, etc).

The Internet of Things is one of the most important and promising technological topics today. Some market researchers estimate that there are more than 20 billion connected devices and counting. Around us, there are smartphones, wearables, and other devices, all of which use sensors. Nowadays, sensors play an important role in our everyday life and in IoT. Sensors monitor our health status (e.g. a heartbeat), air quality, home security, and are widely used in the Industrial Internet of Things (IIoT) to monitor production processes. For these reasons, it is important to know how they work and how we can use them to acquire information.

WHAT IS A SENSOR?

Generally speaking, a sensor is a device that is able to detect changes in an environment. By itself, a sensor is useless, but when we use it in an electronic system, it plays a key role. A sensor is able to measure a physical phenomenon (like temperature, pressure, and so on) and transform it into an electric signal. These three features should be at the base of a good sensor:

- It should be sensitive to the phenomenon that it measures
- It should not be sensitive to other physical phenomena
- It should not modify the measured phenomenon during the measurement process

There is a wide range of sensors we can exploit to measure almost all the physical properties around us. A few common sensors that are widely adopted in everyday life include thermometers, pressure sensors, light sensors, accelerometers, gyroscopes, motion sensors, gas sensors and many more. A sensor can be described using several properties, the most important being:

- **Range:** The maximum and minimum values of the phenomenon that the sensor can measure.
- **Sensitivity:** The minimum change of the measured parameter that causes a detectable change in output signal.
- **Resolution:** The minimum change in the phenomenon that the sensor can detect.

SENSOR CLASSIFICATION

Sensors can be grouped using several criteria:

- Passive or Active. Passive sensors do not require an external power source to monitor an environment, while Active sensors require such a source in order to work.
- Another classification is based on the method used to detect and measure the property (mechanical, chemical, etc.).
- Analog and Digital. Analog sensors produce an analog, or continuous, signal while digital sensors produce a discrete signal.

There are other ways and methods to group sensors but the classifications shown above are the easiest.

HOW TO USE SENSORS IN IOT

The development of prototyping boards and the low price of sensors allow us easily use them in IoT projects. There are several prototyping boards on the market, suited for different projects depending on features and specifications. In this context, we will consider the two most popular boards: the Arduino Uno and Raspberry Pi 2.

This article will explore how to connect different sensors to these boards and how to interact with them.

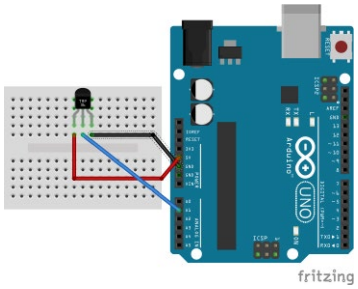
Before diving into the details on how to use sensors with these boards, it is important to note that every sensor has its own operating voltage range. This parameter is very important because the voltage supplied by the board must not be higher than the maximum voltage allowed by the sensor. Therefore, it is important to read the sensor data sheet carefully before connecting it to the board to avoid damage. The same principle is valid for the output signal, which must be lower than the maximum voltage that the board can tolerate.

HOW TO USE ARDUINO WITH SENSORS

The first and the most popular board is the Arduino Uno. It is a microcontroller board based on an ATmega328P. It is very easy to use, and a good starting point. This board provides 6 analog and 14 digital pins. It is perfect to use with analog and digital sensors.

HOW TO MEASURE THE TEMPERATURE USING ARDUINO

The easiest way to start is to connect an *analog* sensor to the Arduino. An analog sensor, as stated before, is a sensor that provides a continuous signal. For our first basic example, we will connect a simple temperature sensor, a TMP36. For more information, you can refer to the sensor data sheet. Generally speaking, the output voltage of this sensor is directly proportional to the environmental temperature. Arduino provides several analog input pins, labeled with an "A," that are suitable for accepting analog signals coming from a sensor. The schema below describes how to connect the sensor:



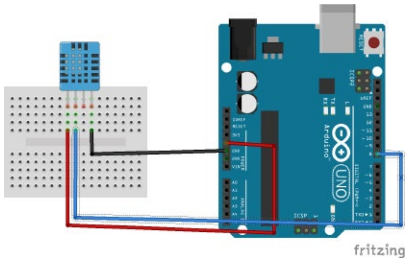
The code to read the temperature is very simple:

```
const int tempSensorPin = A1;
void setup() {
    Serial.begin(9600);
}

void loop() {
    int pinValue = analogRead(tempSensorPin);
    Serial.println("Pin value: " + String(pinValue));
    float voltage = (pinValue / 1024.0) * 5.0;
    Serial.println("Voltage: " + String(voltage));
    float temperature = (voltage - 0.5) * 100; // °C
    Serial.println("Temperature: " + String(temperature)
);
    delay(5000);
}
```

HOW TO MEASURE THE TEMPERATURE AND HUMIDITY USING ARDUINO

Now it is time to connect a **digital** sensor to an Arduino. There are several digital sensors available, but for the sake of simplicity, we will consider a common digital sensor called the DHT11. This sensor measures the temperature and humidity. It is a very cheap sensor that provides a digital output. In this scenario, the sensor data pin has to be connected to the digital Arduino pin, as shown below:



The code is very simple. Even though we can parse the digital signal and read the temperature and the humidity, we will use a library to simplify development. The library is available in the Arduino IDE in the Sketch->Include library menu item.

```
#include "DHT.h"
#define PIN 8
#define DHTTYPE DHT11 // sensor type

DHT dht(PIN, DHTTYPE);

void setup() {
   Serial.begin(9600);
}

void loop() {
   int temp = dht.readTemperature();
   int hum = dht.readHumidity();
   Serial.println("Temperature: " + String(temp));
   Serial.println("Humidity: " + String(hum));
   delay(5000);
}
```

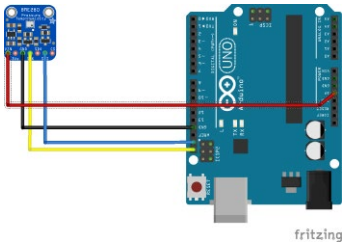
Running the code above, Arduino will log the temperature and the humidity every 5 seconds.

HOW TO CONNECT A I2C SENSOR

An I2C sensor is a serial bus used to connect peripherals to microprocessors. It is widely-used and it requires four different pins:

- Vin
- GND
- CLK (Clock)
- SDA (Data)

To experiment with the I2C sensor with Arduino, we'll analyze the BMP280/BME280 sensor. This sensor measures, among other properties, the barometric pressure. The diagram below shows how to connect a BMP280 to Arduino:



As you can see, there are four different connections. The same connection can be used with a BME280. Do not forget to connect the sensor's CLK pin to the Arduino CLK and the SDA pin (the data) to the Arduino SDA. In addition, the SDO pin cannot be left floating, so you should connect it to the ground or to Vcc. The source code to read the pressure is shown below:

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BMP280.h>
//BMP280
Adafruit_BMP280 bmp;

void setup() {
    Serial.begin(9600);
        if (!bmp.begin()) {
            Serial.println("Could not find a valid BMP280
sensor, check wiring!");
            while (1);
        }
}

void loop() {
    float pressure = bmp.readPressure();
    Serial.println("Pressure: " + String(pressure));
    delay(5000);
}
```

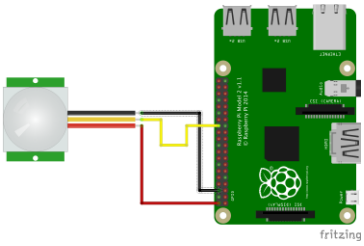
Before running the code above, you have to import a library to handle the sensor, as described in the previous example.

HOW TO USE SENSORS WITH RASPBERRY PI

Raspberry Pi is a single-board computer developed by the Raspberry Pi Foundation. There are several versions of Raspberry Pi with different specifications, but they all have their own operating system based on Linux. It is similar to a PC because it supports video output, USB ports, and keyboards. It is a very powerful board, and the examples below show only a little bit of its power.

HOW TO USE MOVEMENT SENSORS

To monitor movements, we will use a PIR sensor, which stands for *Passive Infrared*. It uses an infrared sensor to detect low level radiation emitted by a warm body. Put simply, when the radiation level changes, it means a warm body is moving into its detection area. This sensor uses a digital pin that gets low (or high) when motion is detected. The schema below shows how to connect the sensor to Raspberry Pi. The connection may change if you use a different PIR version or a different Raspberry Pi board:



The Python code is shown below:

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

sensorPin = 7
GPIO.setup(sensorPin, GPIO.IN)

while True:
    if GPIO.input(sensorPIN) == GPIO.LOW:
        print "Motion detected"
    else:
        print "No motion"
    time.sleep(0.5)
```

When the PIR detects a movement, this simple app will log "Motion detected."

HOW TO DETECT GAS

Another interesting sensor is the MQ-4 sensor. The MQ-4 has a high sensitivity to natural gas. It can respond quickly, and is very easy to use. The connections between the sensor and Raspberry Pi are the same as in the PIR example. Be sure to use the sensor's digital pin, and be sure of the output voltage, which must be lower than 3V. If the sensor has an output greater than 3V, you have to use a logic-level converter. The code to use the MQ-4 is the same as the previous example.

SUMMARY

At the end of this tutorial, we have learned how to use different types of sensors and how to connect them to the two most popular prototyping boards. It is possible to reuse the same sensors with different boards. Once you know how to read data from sensors, you have endless possibilities, and you're ready to explore IoT. You can use data from sensors by storing it in the cloud and using it later to create dashboards.

FRANCESCO AZZOLA is an electronic engineer with over 15 years of experience in computer programming. He is the author of the book "Android Things Projects." He loves creating IoT projects using Arduino, Raspberry Pi, Android, and other platforms, and is mostly interested in the convergence between IoT and mobile applications. He is SCEA, SCWCD, and SCJP certified. He runs a blog called Surviving with Android.

Enterprise and Industrial IoT Edge Processing With Apache NiFi, MiNiFi, and Deep Learning

QUICK VIEW

01. Enterprise IoT has several more requirements, including security, scaling with devices, GPUs, power, command and control, and networking.

02. Learn how to create an IoT data processing pipeline.

03. An effective pipeline includes an ingestion platform, a data bus, a stream processing platform, a scalable storage platform, and a data science platform.

BY TIM SPANN
BIG DATA SOLUTIONS ENGINEER, HORTONWORKS

Enterprise and industrial IoT projects have several more intense requirements than the standards for personal and home devices. The security needs are tighter with full lockdown from source to cloud or on-premise endpoint. At a minimum, solutions need to use SSL or encrypted channels for communication.

Another major feature required is to have support for many device types, from very tiny to large-scale industrial devices costing thousands of dollars. Many devices may include a GPU like the NVidia Jetson TX1, or an add-on compute device like the Movidius. Important features for rugged hardware are airtight cases for devices and backup power supplies. The ability to run without power for extended periods of time in at least a minimal logging mode can be important in remote locations.

These devices need to be remotely monitored, controlled, and updated. These command and control abilities become crucial when patches are required or changes to functionality occur. This can often happen frequently due to security requirements. These processes need to be automated.

Another key requirement is to have full end-to-end data provenance of every change to every piece of data as it travels through the system for full auditing and data governance purposes. GDPR and other laws may be applicable in many sensor-capturing situations, especially regarding camera ingestion.

Sensor and device data that cannot be used with enterprise analytic tools or combined with corporate data in a Hadoop data lake is nearly worthless. Combining device data with other data sources, such as weather or transactional data, is critical for prescriptive and predictive analytics at scale.

Another key requirement is to have full end-to-end data provenance of every change to every piece of data as it travels through the system for full auditing and data governance purposes.

For my clients, these are some common use cases: container truck location monitoring, delivery truck monitoring, service truck and driver monitoring, security camera monitoring, utility asset anomaly detection, and temperature/humidity filtering for devices.

The thing to remember is that while you can start small with a couple of inexpensive devices, a few sensors, a few data points,

hourly data, and no SLAs, you should not plan your system this way. Enterprise and industrial IoT will quickly spread to millions of sensors, millions of devices, and continuous data streams. If you do not plan to handle the volume, velocity, variety, and veracity of data, you will be doomed. If this sounds familiar, this is the big data use case. We just found the motherload of all data. With so many sensors packaged into so many devices located everywhere, IoT data can dwarf all other sources combined. Every truck, every item on a manufacturing floor, and every field sensor can quickly produce billions of streams of data per second with no end in sight. So, I am not giving you a quick start. This is your future-proof infrastructure to scale to massive industrial IoT use cases. This is a proven approach, so let's begin.

Sensor and device data that cannot be used with enterprise analytic tools or combined with corporate data in a Hadoop data lake is nearly worthless.

The first step is to determine what you need to monitor and to obtain a device that has the proper sensors, processing power, environmental suitability, and connectivity your use case demands. The good news is that you do not have to make a difficult decision on software. Apache NiFi is the choice for ingesting any type of data from any source, and it's trivial to connect to these devices. Depending on the size and type of the device, you can choose from a MiNiFi C++ Agent or MiNiFi Java agent. If the device is too small (like an Onion Omega) to support those, you can install a Micropython or C-based library to send MQTT messages. These messages can be sent to an aggregator, say, a Raspberry Pi-sized device attached to your truck. This will allow for localized aggregation, routing, filtering, compression, and even execution of machine learning and deep learning models at that edge. You will also have full control over how and when data is sent remotely to control data transmission costs, energy usage, and unnecessary data propagation. Another feature that makes the MiNiFi and NiFi combination a no-brainer is data provenance. This is built into these tools, and transparently tracks all of the hoops that data travels through, from ingest on a device until it lands in its final home in the cloud or an on-premise data lake. Having encrypted the data and using HTTPS is great, but not knowing who touched the data — and when they did so — along the way is a weakness in most IIoT dataflows, but not in our software.

EDGE DEVICE

Let's dive into a use case with an NVidia Jetson TX1 device, with camera enabled, as our edge device. In my example setup, we have 4GB of RAM, 128GB of storage, WiFi, a USB web camera, and a 256-core Maxwell GPU. We are running a MiNiFi Java Agent along with Python, Apache MXNet, and NVidia's TensorRT. We run deep learning models on the edge device and send images, GPS data, sensor data, and deep learning results if values exceed norms. Using the site-to-site protocol over HTTPS, data is sent to an Apache NiFi cluster (HDF 3.1).

INGESTION PLATFORM

The data arrives securely for further processing, additional TensorFlow processing, and data augmentation in the cases of weather and geolocation. This data is streaming into a Hadoop-based big data platform for analysis, additional machine learning with Apache Spark, and queries via Apache Hive. The primary ingestion method is using Apache NiFi, which handles hundreds of data sources and many data types, and is ideal for simple event processing.

DATA BUS

There are many ways to process our filtered data for storage and machine learning. The most common — and my recommended method — is using Apache Kafka. This is well integrated with Apache NiFi, Apache Storm, Streaming Analytics Manager, Apache Spark, Apache Beam, Apache Flink, and more. This data bus allows for the decoupling of the ingestion platform from our streaming and processing engines. Apache Kafka 1.0 also has support for schemas that make it easy for us to treat data as records from end-to-end when we have data structured enough to include a schema. We often have time series-oriented data with many small values and a timestamp.

STREAM PROCESSING PLATFORM

The two main tools I recommend for most processing use cases are Streaming Analytics Manager and Apache Spark Streaming.

The combination of the two supports most main use cases, SQL processing, joins, windowing, and executing PMML machine learning models. The Stream Processing platform is ideal for processing data in "real-time" as it comes out of the Apache Kafka topics. In SAM, for example, we can use Apache Calcite to query and manipulate these records via SQL in-stream.

The two main tools I recommend for most processing use cases are Streaming Analytics Manager and Apache Spark Streaming. The combination of the two supports most main use cases, SQL processing, joins, windowing, and executing PMML machine learning models.

SCALABLE STORAGE PLATFORM

We need to store several types of data, including key-value, time series, structured table data, unstructured data like images and videos, and semi-structured data like tweets and text blobs. The perfect, safest place to do this is in Apache Hadoop. We can store trillions of rows and petabytes of data and still query it as needed. With the upcoming Hadoop 3.0 release, the platform will support even more data, more files, and more capabilities. We store data as files in HDFS, as well as in Apache Hive Acid tables and in Apache HBase. For some of the faster ingest cases, we store data in Apache Druid for sub-second OLAP analytics.

DATA SCIENCE PLATFORM

In our case, our data science platform leverages Apache Zeppelin for notebooks to experiment, explore, and run analytics and machine learning. We use Apache Hive and Apache Phoenix to run SQL queries to analyze, transform, and organize our data. We use Apache Spark to run various machine learning algorithms and Spark SQL queries, and we have access to a steady stream of real-time data, as well as the massive historic datasets stored in our Apache Hadoop data lake. It is very easy to deploy our models trained on our massive datasets to the streaming processing engines to provide real-time insights with predictive models.

ARCHITECTURE

The nice thing is that, as shown below in the chart, this is all one platform running a common security and authentication system and common administration via Apache Ambari.

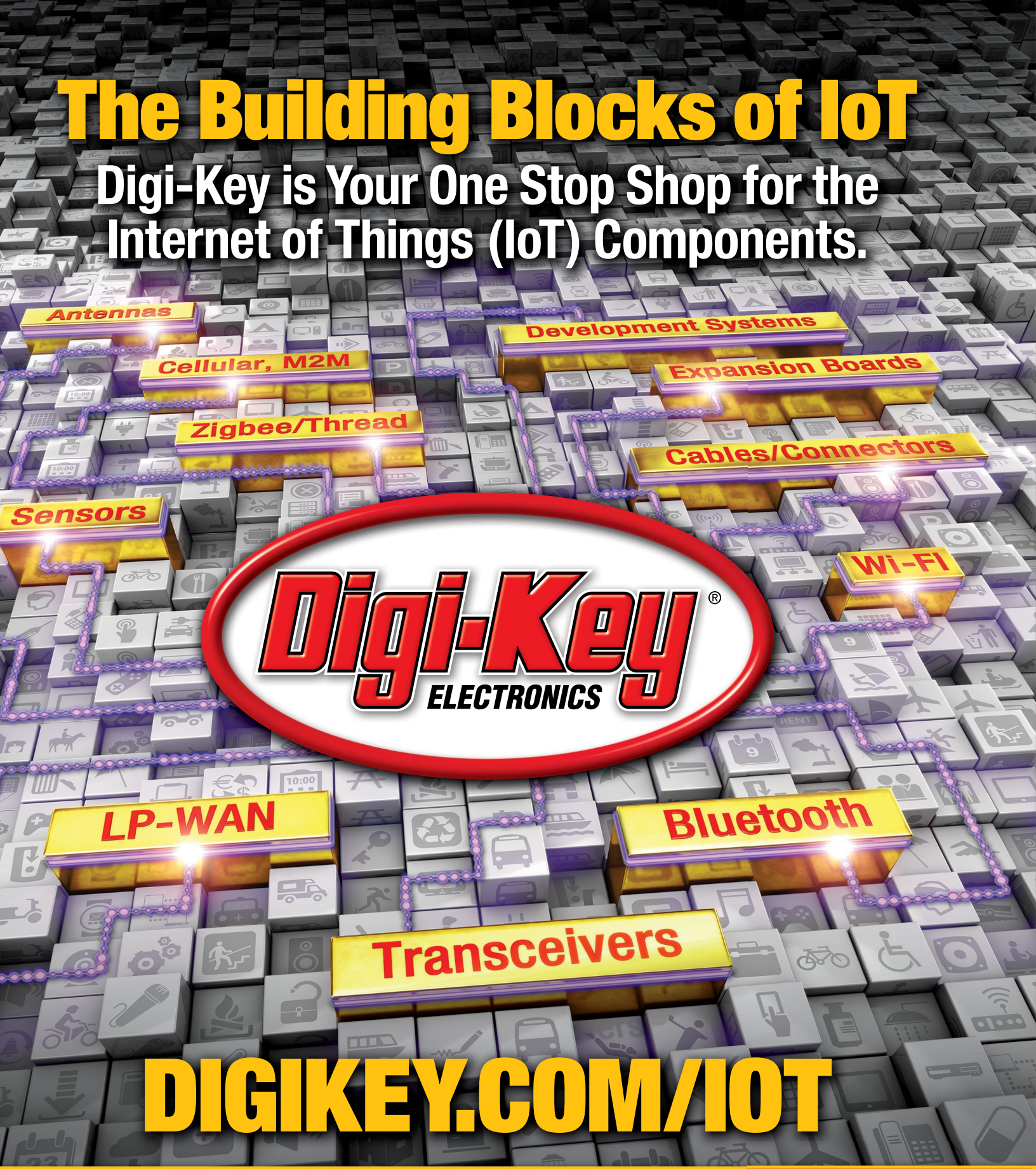
Our global data management platform includes everything that is needed for enterprise and industrial IoT. The GDMP is made up of HDP, HDF, DPS, and services that are built around an open-source system.

DEEP LEARNING

At each layer in the architecture, we can run various deep learning libraries as needed. At the edge, we run Nvidia TensorRT, Apache MXnet, and TensorFlow prebuilt models to scan web camera images for anomalies. In the ingestion phase, Apache NiFi can use TensorFlow, Apache OpenNLP, Apache Tika, and Apache MXnet for sentiment analysis, image analysis, document analysis, and other processing. The streaming engines are all well integrated with deep learning packages. Finally, our query and analytics platform notebooks can run various Apache MXnet and TensorFlow models, as well. We can also run Apache HiveMall for machine learning in our Apache Hive queries.

In the end, we have a continuously growing, always-learning, always-on, scalable platform for developing real solutions for IoT. The funny part is that except for the little piece on the device and some of the ingestion logic, it's the same platform that addresses the same use cases for real-time financial information, real-time social media data, real-time CDC, REST feeds, and thousands of other data sources, types, and origins. In the final analysis, we see that enterprise and industrial IoT are not that much different in their requirements once we get past the first ten meters.

TIM SPANN is a Big Data Solution Engineer. He helps educate and disseminate performant open source solutions for Big Data initiatives to customers and the community. With over 15 years of experience in various technical leadership, architecture, sales engineering, and development roles, he is well-experienced in all facets of Big Data, cloud, IoT, and microservices. As part of his community efforts, he also runs the Future of Data Meetup in Princeton.



Cellular network options tailored to IoT Applications- CIoT (Cellular IoT)

The availability of cellular connectivity nationwide in any country would seem to lend itself well to IoT applications. Prior to the release of LTE-M and NB-IoT network availability, costs and power usage were limiting factors for most applications. LTE-M and NB-IoT service plans are approaching 2G costs, while power usage is being greatly reduced from previous 2G architectures with the introduction of PSM (Power Saving Mode) and eDRX (Extended Discontinuous Reception).

PSM provides the ability for the device to tell the network it will not be active for an indefinite period of time. This significantly reduces the power usage while the device is in a low-power mode. For a device that only needs to transmit daily or weekly, we are now approaching a 10-year lifespan with a couple of AA batteries.

eDRX is a mode that extends the time needed between LTE "network checks." A typical network check is done every 1.28 seconds. Each check involves one of the radios powering up and transmitting a packet, which inevitably wastes energy. eDRX provides the ability to extend the window in between these network checks to 40 minutes or more, providing significant power savings.

A common question between NB-IoT and LTE-M is, "which of these will work best for my application?" The big difference is that LTE-M can work with mobile applications, where NB-IoT must remain in one location, it does not support handover between towers. Another major difference is the transmission speed, LTE-M supports up to 1 Mbps while NB-IoT is approximately 200 Kbps. Network coverage may vary widely because as new networks are deployed, many global carriers are deploying either LTE-M or NB-IoT and do not currently support both.

WRITTEN BY JOSH MICKOLIO
PRODUCT MANAGER – SEMICONDUCTORS, DIGI-KEY ELECTRONICS

PARTNER SPOTLIGHT

mangOH™ Red

Open source rapid proto to production platform supporting cellular connectivity from Sierra Wireless

CATEGORY
Open Source Hardware

OPEN SOURCE?
Yes

NEW RELEASE
Supporting new technologies including Cellular LPWA

CASE STUDY

The mangOH Red platform offers 2G, 3G, and 4G LTE connectivity including LTE-M, NB-IoT, and CAT-1 connectivity using the CF3™ Common flexible form factor modules from Sierra Wireless. Wi-Fi and Bluetooth are on-board with other connectivity options available through the IoT expansion card slot. The board includes built-in accelerometer/gyroscope, pressure, and light sensors. With 90 percent of the hardware and software development already done for you, the mangOH allows you to put your development effort into getting your product to market faster. Legato® Linux integrated on the CF3 modules from Sierra Wireless provides the necessary drivers, allowing you to start writing software immediately.

STRENGTHS

- 2G to 4G & LTE-M/NB-IoT
- IoT Expansion slot
- Wi-Fi and Bluetooth
- Accel/Gyro, Pressure, Light sensors

WEBSITE digikey.com | **TWITTER** @digikey | **DOWNLOADS** digikey.com/en/blog

Pushing IoT Data Gathering, Analysis, and Response to the Edge

QUICK VIEW

01. Use an IoT architecture for edge-based data collection and processing.

02. Discover open source solutions to push IoT data collection to the Edge.

03. Reduce bandwidth usage by pushing data collection and processing to the edge in IoT.

BY DAVID SIMMONS
IOT DEVELOPER EVANGELIST, **INFLUXDATA**

INTRO TO THE IOT

The Internet of Things is not so much a *thing* as it is a *concept.* It's a concept that enables us to instrument our world with sensors and respond to the data coming in from those sensors in meaningful ways. It's about adding sensors to all the *things* in our world so that we can measure, analyze, visualize, predict, and react to the environment around those things. The IoT is not a product because it's not a single thing. If that sounds fairly abstract, it is, but we'll figure it all out.

In addition to all of that, there are multiple segments, or markets, within the IoT. The one most people are familiar with is consumer IoT. You may have a smart thermostat in your house, some smart switches, or other internet-connected devices and appliances. These are generally considered part of consumer IoT. Then there's the Industrial IoT, or IIoT. This segment includes things like smart buildings, industrial automation, and monitoring of industrial processes. This is a part of the IoT that most people never see, and rarely hear about, but it's where the most growth and innovation happens, and it's what we will focus on in this article.

AN IOT ARCHITECTURE

Almost any IoT architecture is going to involve a few basic components like sensors, a place to collect and store data, some way to visualize and interact with the data, and often some way for actions to be taken based on events in the data.

It's a fairly simple concept that's been around for a very long time: a sensor collects data and sends it to a server to store it. That data is then made available for analysis and, based on that, some action is taken.

But as IoT deployments grow in size and complexity, the ability to simply have a sensor send all its data to a single, monolithic backend begins to become less and less practical. First, the sheer amount of data begins to become overwhelming for any one sensor. Second, there are few systems that can handle the volume of sensor data. To illustrate this point, let's look at a modest-sized IoT deployment of, say, 10,000 sensors deployed across an enterprise. Each sensor takes a series of 5 readings every second. That's 50k readings/second streaming over the internet to a single backend server. If each reading is 1kB of data, that's 5kB per sensor/second, or 250kB/s of data per second overall. It's fairly reasonable to assume that almost any competent backend system could handle this fairly modest amount of data. But now let's begin to scale that to something that would actually go into production, and the numbers rapidly begin to grow out of control. Sensors that collect 1,000 readings per second, at 1kB of data per reading, grows to 1MB of data per second, per sensor. At 10,000 sensors, you're streaming a GB/second of data. Still think it sounds reasonable to stream all of that data to a single backend system in real time? We need to look for alternatives.

PUSHING DATA COLLECTION TO THE EDGE

You could compress your data, but there's a compute overhead in doing so. You could scale back the frequency of data collection, but this could impact your ability to detect and respond to anomalies. Or, you could push your data collection, analysis, and response out from the data center or cloud to the edge.

In the scenario above, with 10,000 sensors, it would be reasonable to segment the deployment into groups, each group connecting to the internet via a gateway device. If you could turn each gateway device into a mini data collection, analysis, and response machine, that would help with the overall scaling problem. If you segment each gateway device to service as many as 1,000 sensors, each gateway device would be at the 250kB/s data rate and could easily handle the load.

Now you've got 1,000 gateway devices, all collecting data from their 1,000 individual sensors, which has solved your data-rate scaling problem, but it has created another problem: distributed data. Now your data is on

1,000 different devices, and you have no way to see aggregated data from all of your sensors. As always, the left hand giveth while the right hand taketh away!

Now, on top of your data collection and scaling problems, you've got a data aggregation problem! I know it sounds like I'm creating more problems than I am solving here, but there is a way to apply a solution across the entire deployment that makes all of these problems go away: the **TICK Stack**.

DEPLOYING THE TICK STACK TO SOLVE THE SCALING PROBLEM

The TICK Stack is made of 4 open source software components designed specifically to make the collection, storage, management, visualization, and manipulation of time series data easy and scalable.

The 'T' in TICK stands for **Telegraf**. Telegraf is a plugin-based high-performance data ingestion engine designed to gather incoming data streams from multiple sources, and in an extremely efficient manner, output those streams to data storage platforms. One of those platforms, the one we're going to focus on here, is InfluxDB, the 'I.'

InfluxDB is a **Time Series Database** designed from the ground up for performance and ease of use when dealing with time series data—-and what is IoT data if not time series data?

The 'C' in TICK is for **Chronograf**, the data visualization and management front-end for the other components of the stack. Using Chronograf, you can quickly and easily build stunning dashboards for data monitoring:

These dashboards can help you easily monitor your sensor data and spot anomalies and trends in your data that you might otherwise miss.

The 'K' in TICK is **Kapacitor**. Kapacitor is our streaming processing engine that runs alongside InfluxDB to do more complex data processing, process alerts, etc.

Great, so how will this help? Well, one thing I've been working on lately is deploying the entire TICK Stack from edge to data center for complete IoT data collection, analysis, reporting, and alerting—-and I can say that it has been wildly successful. I took a $30 Pine-64 LTS board, added a $35 7" touchscreen display, a $9 Bluetooth/Wifi card, and built an edge device that is capable of collecting sensor data via WiFi, wired Ethernet, or Bluetooth LE (I've since added a LoRA radio to it as well, just for fun). That device, with a 32GB MicroSD card, collects sensor data and displays it on a dashboard (in fact, that image above is the dashboard on the device). In addition, it processes that data and sends alerts when the temperature from one of the sensors changes more than 1°F, or whenever the CO2 concentration in the room changes by more than 100ppm.

But I said I was going to conquer the distributed data problem, and with Kapacitor I have done exactly that. I've used Kapacitor to generate the alerts discussed above, but I'm also using Kapacitor to do some fairly sophisticated downsampling of the data. I'm collecting about 100 sensor readings every second on this device. But because I am handling the data visualization and anomaly detection on this edge device, I don't actually need to send this highly granular data to my backend system. So, I'm downsampling my data before sending it back upstream.

What is downsampling of data? It's reducing the granularity, or resolution, of the data while preserving the overall trends in that data. Some refer to this as a "rollup" of the data. So, I am taking a rolling 5-minute mean of the temperature data over a 5-minute window, and rolling that data up to the cloud for long-term storage and further analysis.

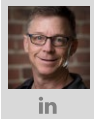
The second-by-second analysis of the temperature data is important locally, but it becomes less important on the backend system. Further, on the backend system I am analyzing the aggregated data from multiple gateway devices collecting this data—each of which is doing its own local processing, visualization, and alerting—so overall trend analysis is more important. In this way I am distributing the load for data collection, data processing and analysis, and action to the closest point to where the data is actually generated, while still preserving the data to a persistent backend data store for further, higher-level, analysis.

The best part of the whole thing? I'm deploying the *exact same code base* at all levels of the architecture. It's the same code—-the same stack, from edge to data center, which reduces complexity and makes deployment easier and more economical. **Here's what it all looks like as a gif**.

It's a simple solution to a complex architecture designed to maximize data collection and scalability while reducing complexity and maintenance issues.

I say quite often that IoT data must be timely, accurate, and actionable in order to be useful. This architecture and use of the TICK Stack maximizes the ability to be timely and accurate in your data collection, and take action on that data as close as possible to the point of data generation where it makes the most impact.

DAVID SIMMONS is the IoT Developer Evangelist at InfluxData, helping developers around the globe manage the streams of data that their devices produce. He is passionate about IoT and helped to develop the very first IoT Developer Platform before "IoT" was even 'a thing.' David has held numerous technical evangelist roles at companies such as DragonFly IoT, Riverbed Technologies, and Sun. He studied Computer Science at the University of New Mexico and has a BA in Technical Writing from Columbia University.



In the Age of Customer Experience, Isn't it Time For a New Database?

If you still harbor any doubts about the power of digital disruption, or whether it's going to transform your business, consider this one simple fact: Today's S&P 500 is swapping out old companies for new at the mind-boggling rate of one every two weeks.

Digital disruptors in the mold of LinkedIn, Uber, and eBay are reinventing one business model after another, and they're doing it first and foremost by revolutionizing the customer experience. As a result, the ability to deliver exceptional customer experiences has become the single most important competitive differentiator in practically every industry.

The problem is, traditional databases simply weren't built to deliver the fast, seamless, and personalized experiences that modern customers demand. Old-school analytical systems are backwards-facing, while transactional systems are focused only on the moment of purchase. But, today's customer engagements depend on a lengthy and complex sequence of interactions that need to be executed flawlessly in real time.

That's why Couchbase specifically designed the world's first and only Engagement Database to let any business – from startup to enterprise – create and execute brilliant customer experiences every time. The Engagement Database responds instantly to fast-changing data, market factors, and business demands. It scales quickly and easily to handle spikes and rapid growth without loss of performance or outages. And it's incredibly flexible and easy to manage, even with limited resources. Now, you can be next to start a revolution.

WRITTEN BY RAVI MAYURAM
SVP OF ENGINEERING, CTO, COUCHBASE

PARTNER SPOTLIGHT

Couchbase Data Platform

From digital transformation to customer engagement – start a revolution with the world's first Engagement Database

Couchbase

CATEGORY
NoSQL Engagement Database

OPEN SOURCE?
Yes

NEW RELEASE
Annual

CASE STUDY

In a world where customer expectations are constantly rising, businesses have to deliver continuously amazing digital experiences if they want to stay on top. That's why hundreds of leading companies from nearly every industry count on Couchbase to solve their toughest digital innovation challenges.

Couchbase designed the world's first Engagement Database with the most powerful NoSQL technology, so that any organization can drive digital innovation with ease. Designed to be a joy to work with, only Couchbase delivers unparalleled performance at any scale, while also providing the unmatched agility and manageability that today's businesses require to continually reinvent their customer experiences and thrive with an adaptive competitive edge.

STRENGTHS

- **JSON with N1QL at any scale** – Leverage existing SQL skills with N1QL (SQL for JSON)
- **Memory-first speed** – Rich data access, in-memory replication, 99.999% availability
- **Security across the stack** – Encrypt, audit, protect, and secure your data anywhere
- **Cloud-native for global scale** – Cross datacenter replication (XDCR) in any cloud
- **Fully synced and offline-first** – Real-time data sync even when users are offline

NOTABLE CUSTOMERS

- AT&T
- Cisco
- eBay
- General Electric
- Marriott

WEBSITE couchbase.com | **TWITTER** @couchbase | **BLOG** blog.couchbase.com

Introduction to Security and TLS

BY ERICH STYGER
PROFESSOR, **LUCERNE UNIVERSITY**

QUICK VIEW

01. The Internet-of-Things is all about connectivity and sending data.

02. A critical part of this is to ensure that the connection is secure.

03. This article shows how TLS (Transport Layer Security) and certificates are used between a client and a server to establish a secure communication channel.

IoT (Internet of Things) is all about connecting to the internet and even more about security. Without security and encrypted communication, everyone can possibly see what I send or receive. And this is especially bad if passwords or user names are sent in an unencrypted way. So, encryption and secure communication is key. The solution to that is to use a connection which uses the TLS (Transport Layer Security) protocol, which I want to use for my MQTT communication (see **MQTT with lwip and NXP FRDM-K64F Board**). I'm still learning MQTT, and I'm learning more about the fundamentals of security and security protocols. So, this article is about what I have learned recently, and what I can use to make my data communication secure: network stack architecture, symmetric and asymmetric encryption, and certificates.

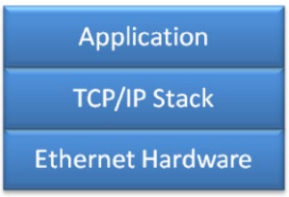
OUTLINE

This article walks though the basic principles for secure communication using TLS with MQTT in mind. TLS is the successor of SSL (Secure Sockets Layer), and the two are often used together (TLS/SSL). TLS (as the name indicates) is an encryption on the transport layer: that means that the application layer does not have to implement the encryption itself. Instead, it configures the transport layer to use the encryption protocol.

ARCHITECTURE

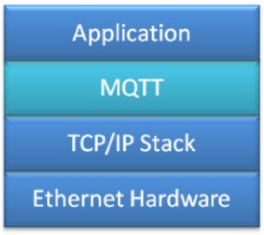
For an application (e.g. running on a microcontroller) to communicate with the internet, it requires a communication stack (TCP/IP, e.g. **lwIP**) plus the needed hardware to communicate with the physical layer (e.g. the FRDM-K64F board). TCP/IP itself is the common language that the application uses to communicates with the other side.

The same principle applies to running the MQTT Mosquitto broker (or any server) on my host machine.



Unencrpyted Communication Stack

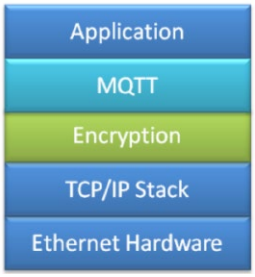
MQTT is a special "language" using TCP (Transmission Control Protocol), and sits between the application and the TCP/IP stack. The application uses the MQTT layer to talk and understand the MQTT language:



Application stack with MQTT

TCP/IP uses "sockets" or "ports." By default, Mosquitto is using the port 1883 (see **MQTT with lwip** and **NXP FRDM-K64F Board**),

which is not using an encryption protocol. That means that everyone can potentially see the exchange of data between the MQTT broker and clients. Having an unencrypted connection is a great with MQTT, as this is a simple and easy way to start exploring MQTT. Configuring encryption and using an encrypted connection, on the other hand, is much more complex. But once I have things working unencrypted for testing purposes, I definitely want to use encrypted communication. Instead of doing "end-to-end" encryption in the application itself, a better approach is to put an encryption layer on top of the communication stack:



MQTT Application with Encryption

This way the application does not need to implement the encryption protocol itself, it simply talks to the encryption layer and that will do all the work.

SYMMETRIC AND ASYMMETRIC ENCRYPTION

Encryption relies a lot on math, random number generators, and cryptographic algorithms. With encryption, there is the need for "keys:" sequences of bits and bytes which are used to lock (encrypt) and unlock (decrypt) the data.

With **symmetric** encryption, the same key is used to encrypt and decrypt a message. It means that everyone having that (blue) key will be able to decrypt the message. So, security depends how securely I can distribute and keep that key.

With asymmetric encryption, I have a pair of mathematically connected keys: a shared green key and a private red key. I keep the red key private and do not disclose and distribute it. The green key is public: everyone can use it. Everyone can encrypt a message with the green public key, but only the one with the red private key is able to decrypt it. The public and private key build a pair of keys. They are different but mathematically related. That way, only the private key is able to decrypt a message encrypted with the public key.

CERTIFICATES

But how can I know that the public key I have received is really from the person I think it is coming from? Maybe I think I have received the public key from someone I know, but instead it's a "man in the middle" intercepting all messages, so I have that public key instead.

Here, certificates come into play. Certificates are a kind of passport, provided by a Certification Authority (CA), which testify that the certificate holder is really that person. The certificate itself can be used to encrypt and verify a key, similar to packing a key into a certificate.

In our example, Sue then can use the trusted certificate to extract the key. If that fails, Sue knows that the certificate with the key is not coming from Joe.

Instead of doing "end-to-end" encryption in the application itself, a better approach is to put an encryption layer on top of the communication stack.

There are several different ways and protocols to distribute the keys and certificate. Usually the keys/certificates are pre-distributed (e.g. pre-installed on the devices), or the secret information is exchanged using a multi-stage protocol like TLS.

TRANSPORT LAYER SECURITY (TLS) PROTOCOL

TLS has a few different versions (current version is 1.2, with 1.3 in the draft state). TLS is using a special protocol called 'Handshake' to agree on the protocol and to exchange keys (see this link for details). Basically, the certificate is used to verify the server identity, and the asymmetric encryption is used to exchange a shared secret key for symmetric encryption:

1. Client sends a clear (unencrypted) message to the server, asking for an encrypted session.
2. Server responds with their server certificate, which includes the server public key in it.

3. Client verifies the certificate and extracts the public key.
4. Client uses the public key to send a pre-master key they have generated to the server.
5. The server uses its private key to extract the pre-master key.
6. Both the client and the server use the pre-master key to compute a shared secret key.
7. Client sends a message to the server encrypted by that shared secret key.
8. The server decrypts the received message and checks it.
9. If that passes, the server sends back an encrypted message using the shared secret key to the client to confirm that everything is ok.
10. From this point on, both the client and server are using the shared secret key for their communication.

The graph at **ibm.com/support/knowledgecenter/en/SSFKSJ_7.1.0/com.ibm.mq.doc/sy10660_.htm** provides a good overview of the handshake process.

Certificates are a kind of passport, provided by a Certification Authority (CA), which testify that the certificate holder is really that person.

ENCRYPTION MIDDLEWARE

I don't plan on implementing TLS or the cryptographic algorithms. Instead, I was looking for an open source library I could use. There are different vendors providing encryption middleware. For my project with MQTT, lwIP, and the FRDM-K64F board, I have found the following options:

- **OpenSSL** (**openssl.org/**): A very complete and capable open source implementation with good documentation. It's mostly targeting desktop and Linux machines, as well as bigger embedded devices. I have a permissive license, and it's and moving to Apache license version. But because of the general overhead I didn't consider it for use with the FRDM-K64F.
- **wolfSSL** (**wolfssl.com/wolfSSL/Home.html**), formerly CyaSSL: Targeting embedded devices, it would fit on the FRDM-K64F. The free Open Source (GPLv2 and GPLv3) version is restrictive, and the commercial one is outside of my budget.
- **CycloneSSL** (**oryx-embedded.com/cyclone_ssl.html**): Good features, but same as wolfSSL: GPL license and commercial version only.
- **mbedTLS** (**tls.mbed.org**), formerly PolarSSL: owned by ARM, good documentation, and its Apache 2.0 license allows me to use it in both commercial and open source projects for free. And it seems to be used with lwip too.

The mbedTLS has been the most versatile and open library I have found, and this is why I have started using it in my project. You can check my site for more information.

SUMMARY

To use secure data transport, I have to use encryption. Cryptographic algorithms are provided with several open source libraries, and the mbedTLS library seems to fit my needs best. The key to encryption is the distribution and handling of keys and certificates. TLS (or Transport Layer Security) is a protocol which manages key verification and distribution, which is provided in the mbedTLS library.

ERICH STYGER researches and teaches computer science and electrical engineering at the Lucerne University of Applied Sciences and Arts in Switzerland. As director, principal engineer, and university professor, he has designed microcontroller instruction sets, developed several C/C++ compilers with advanced optimizations, contributed and extended debuggers with real-time features, and developed dedicated tools for embedded systems. His current focus and research area is everything from small sensor systems up to large distributed embedded systems.

DIVING DEEPER

INTO IoT

#IoT TWITTER ACCOUNTS

 @thedavedev

 @AmyxIoT

 @JoeSpeeds

 @iotwatch

 @RobTiffany

 @gigastacey

 @andysc

 @Fisher85M

 @Cate_Lawrence

 @theRab

IoT ZONES

IoT

dzone.com/iot

The Internet of Things (IoT) Zone features all aspects of this multifaceted technology movement. Here you'll find information related to IoT, including Machine to Machine (M2M), real-time data, fog computing, haptics, open distributed computing, and other hot topics. The IoT Zone goes beyond home automation to include wearables, business-oriented technology, and more.

AI

dzone.com/ai

The Artificial Intelligence (AI) Zone features all aspects of AI pertaining to Machine Learning, Natural Language Processing, and Cognitive Computing. The AI Zone goes beyond the buzz and provides practical applications of chatbots, deep learning, knowledge engineering, and neural networks.

Big Data

dzone.com/bigdata

The Big Data/Analytics Zone is a prime resource and community for Big Data professionals of all types. We're on top of all the best tips and news for Hadoop, R, and data visualization technologies. Not only that, but we also give you advice from data science experts on how to understand and present that data.

IoT REFCARDZ

Messaging Infrastructure for IoT at Scale

IoT technology is all around us and can no longer be ignored. At the core of IoT technology is the messaging infrastructure. Download this Refcard to learn more about the foundation of an IoT platform, how to get started with EnMasse, and open source projects focused on IoT device connectivity.

AMQP Essentials

Practical introduction to AMQP — a binary, multiplexed, symmetric, secure, and lightweight message transport protocol designed for enterprise applications and particularly useful for IoT.

Getting Started With Industrial Internet

Introduces basic concepts and technologies of the Industrial Internet, including sensors and actuators, industrial control systems, human-machine interfaces, real-time streaming data, device security, and more.

IoT PODCASTS

The Internet of Things Podcast

Learn about a wide range of topics in the world of IoT, like smart homes, connected cars, IoT security, and more.

IoT This Week

With this weekly podcast, keep yourself up-to-date with the worlds of IoT, technology, and security.

Industrial IoT Podcast

In this interview-based podcast, get insight into the planning and implementation of IIoT systems.

IoT RESOURCES

How It Works: Internet of Things

In this brief video from IBM Think Academy, find out how IoT works and what we can do with the data accessed from millions of IoT devices.

IoT Tutorial for Beginners

Get a grasp of the basic concepts of IoT and learn how IoT is trying to revolutionize the world.

The Internet of Things TED Talk

Learn from a technology innovation expert how the internet of things is changing our lives in ways we don't even recognize.

Protect and *Monetize* the IoT

from the Edge to the Cloud

Protect your APPS and DEVICES and monetize more effectively

PROTECT
Your Apps and Devices

MANAGE
Software, Features and Updates

ANALYZE
Product and Usage Data

MONETIZE
and Grow Recurring Revenue

Get ahead of the game with the market leader for IoT Monetization and award-winning technology.

WINNER
IoT GLOBAL AWARDS 2018
Recognising excellence in IoT innovation

IoT BREAKTHROUGH AWARD 2018
IOT ENABLEMENT COMPANY OF THE YEAR ENTERPRISE MARKET

FLEXERA

FLEXERA.COM // 1.800.374.4353

Software Composition Analysis Helps Keep the IoT Secure

Recent botnets have shone a light on the vulnerability concerns around IoT. IoT devices have characteristics that make them a target for botnet and malware authors. These devices have high-speed network connections, run Linux, and lack monitoring systems that alert users to a hack.

The typical embedded Linux system uses hundreds of open source packages. While these components are typically high quality, all software contains defects and over time vulnerabilities in these components are discovered.

Often, these devices are not designed to be auto-updated, and depend on components from commercial and open source organizations that have vulnerabilities discovered every few weeks.

It is becoming a best practice to pay attention to a device's Software Bill of Materials, especially components with known vulnerabilities. By keeping track of this list, a company can stay ahead of malware authors.

Today, products are available that are designed to help system designers keep track of their use of open source and commercial dependencies, as well as get alerts when new vulnerabilities are discovered. This allows them to create products that do not contain known vulnerabilities when shipped, and to stay on top of components as they age out when deployed. This type of scanning and management software is known as Software Composition Analysis (SCA).

Such software contains scanning and workflow features designed to help technology companies discover, manage, upgrade and comply with their use of open source components. By scanning and comparing the files used on the devices to a database of billions of known open source files, the system can discover usage of third-party components for the purposes of vulnerability management as well as open source license compliance. Managing these requirements allows a developer to ship a device that respects the open-source community, as well as protects the company's users from attacks.

WRITTEN BY JEFF LUSZCZ
VICE PRESIDENT OF PRODUCT MANAGEMENT, FLEXERA

PARTNER SPOTLIGHT

Flexera IoT Monetization Platform and FlexNet Code Insight

Application and device security, open source scanning, and software vulnerability management at all stages of the product lifecycle

CATEGORY
IoT Monetization and Software Composition Analysis

OPEN SOURCE?
No

NEW RELEASE
Quarterly

CASE STUDY

Flexera serves thousands of customers with millions of devices, and enables billions of IoT revenue monetized through its IoT Monetization platform. Customers use FlexNet Code Insight to reduce software vulnerability risk and manage license compliance for open source software (OSS) and third-party components.

- Development: Manage Open Source and 3rd Party Components; implement tamper resistance to prevent hacking and piracy
- Delivery and Deployment: Ensure only eligible customers can use software/devices; enable secure activation to prevent grey market abuse
- Management of Security Threats: Detect tampering and react, detect cloning and report, and vulnerability scanning
- Update: Send software and firmware updates to applications and devices

STRENGTHS

- The industry standard for IoT monetization
- Largest open source knowledge base with over 13 million components
- Manage, track, and update devices

NOTABLE CUSTOMERS

- Siemens Building Technologies
- ADVA Optical
- Schneider Electric
- Ciena
- PolySync

WEBSITE flexera.com/producer **TWITTER** @flexera **BLOG** blogs.flexera.com/sca

SAILING ON THE Sea of Data

As IoT devices make their way across the world, those devices and sensors generate an enormous amount of data that needs to be processed and transferred in a secure way. Of 522 developers interviewed for DZone's Guide to the Internet of Things, 31% of respondents have issues with large amounts of unprocessed data, and 50% are concerned with device security. Fortunately, two emerging technologies have become almost synonymous with effective IoT data analysis, and transfer: Artificial Intelligence and Blockchain. Let's explore the ways in which these three technologies work together as IoT starts to expand its territory across the world.

Internet of Things

IoT, which refers to a network of objects that can capture data autonomously and self-configure intelligently, is a very important bridge between AI and blockchain. With more connected devices than people in the world, the world is more immersed in the Internet of Things than most of us may realize. IoT is only going to continue trekking along its path to becoming the wave of the future, with nearly 64% of respondents being interested in working on IoT projects. IoT will depend heavily on both blockchain and AI to be successful.

Blockchain

A blockchain is a type of database that records transactions securely, making it a major source of fuel for secure systems that will be key as IoT continues to bring more data than ever to the forefront. With 50% of DZone members saying that device security is a major challenge that they have faced when developing IoT applications, blockchain will prove to be extremely important for IoT.

Artificial Intelligence

61% of DZone survey respondents are looking to adopt an AI solution, and all the data collected by IoT devices will be what helps keep AI – which refers to a machine's ability to make decisions and perform tasks that simulate human intelligence and behavior – afloat. Closely tethered to IoT, artificial intelligence is helping to enhance decision-making and analytics capabilities in IoT systems. AI is going to be the driving force behind making IoT and connected devices the next best (and biggest) technology. 48% of survey respondents have claimed that AI has already helped them make more effective decisions.

Why Time Series Matters for IoT Sensor Data

Download the e-book

"InfluxDB has been great for our environment because it seemed to be built for our specific use case. While testing other solutions, we tried our current production loads against InfluxDB and found that it exceeds our needs."

Michal Knizek, Head of Server Development, tado°

influxdata®

The Modern Engine for Metrics and Events

IoT Data is Time Series Data

Time Series Databases are the fastest growing database category, according to independent research, and store measurements tracked over time. Time series are simply measurements that are tracked, monitored, downsampled, and aggregated over time. This could be server metrics, application performance monitoring, sensor data, or trades in a market. The key difference between time series data and regular data is that you're always asking questions about it over time.

If you think about it, sensor data, or IoT data, is time series data where the data collected is time-stamped to show change over time in order to improve efficiency for things such as energy use, structural health, or even material quality levels. You can see how useful this data is, but one issue that many companies face is around the massive amounts of data that all of these devices produce. Companies need to figure out a way to store, track, analyze, and make sense of the vast amounts of data that is generated.

Because of this, Time Series Platforms are becoming a critical architectural component in IoT implementations.

THE TIME SERIES WORKLOAD

Time series data is very different from any other workload: millions of writes per second, the ability to do real-time queries on huge datasets, time-based functions that help measure change over time, optimal disk compression with these large data sets, and the need to keep high-value data accessible for real-time access, while storing older data with potentially different time precision available for historical analysis.

InfluxData is the leading Time Series Platform and comes out of the box ready to use. Learn more.

WRITTEN BY CHRIS CHURILO
DIRECTOR OF PRODUCT MARKETING, INFLUXDATA

PARTNER SPOTLIGHT

InfluxData

The modern engine for metrics and events

CATEGORY
Time Series Data Platform

OPEN SOURCE?
Yes

NEW RELEASE
Quarterly

CASE STUDY

tado° connects cooling and heating systems with the Internet to enable consumers to intelligently control them. Their app adjusts to the residents' behavior in real time and also takes current weather forecasts and building characteristics into account. tado° uses InfluxData to gather analytics data collected from its hundreds of thousands of units across the globe. tado° uses this data to power its smartphone apps which help their customers understand their energy usage and determine their ideal setting. This IoT monitoring use case shows how using the right tool to power tado°'s IoT platform gives the company the proper insights to generate energy savings for its customers.

STRENGTHS

- Developer Happiness
- Faster Time to Awesome
- Ease of Scaleout and Deployment

NOTABLE CUSTOMERS

- Tesla
- Siemens
- BBOXX
- Spiio

WEBSITE influxdata.com | **TWITTER** @InfluxDB | **BLOG** influxdata.com/blog

Smart Cities: Who Wins and Who Loses?

BY CATE LAWRENCE
ZONE LEADER, **DZONE**

QUICK VIEW

01. Smart cities projects need to connect and intersect to provide solutions to existing problems.

02. ROI needs to be considered instead of merely funding opportunities.

03. Open data can be a catalyst for new app development, research projects, and targeted local campaigns.

04. Who benefits and owns smart city data and the issue of citizen privacy and engagement cannot be under estimated.

Since the development of the public internet, Wi-Fi, Bluetooth, sensor technology, and the Internet of Things, we have seen smart cities around the world develop and flourish from Austria, to Zagreb, and everywhere in between. The evolution of smart cities is based on the concept that connected technology can not only make a city more progressive, but also generate data to create insights that can improve the safety, quality of life, and economic opportunities for those that live, work, or visit a city. There are certain characteristics that seem synonymous with smart cities and it's worth taking a step back from the hype to consider why and how we are building them. What can we blueprint from earlier developments, how can we learn from mistakes, and how can we build better smart cities?

INTERSECTIONS WITHIN CITY SECTORS

One of the keys underpinning smart cities is intersection. Smart cities don't (and can't) work when projects and data are siloed. One example is city transport, where a variety of solutions can provide the means for people to travel around the city more efficiently, safely, and sustainably. Smart traffic management can be used to monitor and analyze traffic flows to optimize streetlights to prevent roadways from becoming too congested during typical rush hours. Connected parking hubs can connect with public transport to enable park and ride customers to find free parking spots on the way to the morning train. Solar lights can be utilized in parking meters and smart streetlights will dim when there aren't cars or pedestrians on the roadways to save energy. In public spaces such as sports stadiums, smart surveillance can also be utilized to provide safety evacuation information such as instructions and directions in the case of emergency, and analytics can be coordinated with weather and traffic information outside of the stadium. This means fans can leave happy, with the knowledge of their fastest route home. Each of these initiatives, whilst impressive as a stand-alone project, would not really succeed without the others.

MALLEABLE, ADAPTABLE TECHNOLOGICAL INFRASTRUCTURE THAT CAN EVOLVE AS TECHNOLOGY ADVANCES

The UK City of Bristol was recently awarded the Judges' Choice Smart City Award at the GSMA's 2018 Global Mobile Awards. They were commended for Bristol is Open, a joint venture between the University of Bristol and Bristol City Council. Three networks are integrated through software-defined control: fiber in the ground; a wireless heterogeneous network along the Brunel Mile area of Bristol with Wi-Fi, 3G, and 4G; and a radio frequency mesh network deployed on 2,000 of the city's lamp posts. The University is also carrying out 5G research and delivering the UK's first 5G urban deployment. This platform is allowing companies of all sizes to come and test new technology in a real-world environment rather than just inside a laboratory and demonstrates the value of an underlying structure that thoughtfully considers future tech, cross-discipline collaboration, open test grounds, and meaningful connections.

FUNDING NEEDS TO CONSIDER ROI

At present, many smart cities are funded through grants such as the Horizon 2020 EU fund, University research funding, or in-kind benefits from telcos or utilities. To succeed beyond these 3-5-year funding rounds, a combination of public and private funding is necessary. For example, Cisco announced $1b in grant funding for Smart City investment in November last year, bringing private funding to a number of US municipal councils. But what happens when private funders want ROI?

Some projects have the advantage of cost savings to a city that

outweigh the price of infrastructural investment- such as a connected utility systems, For example, in Melbourne, Australia, South East water use digital metering technology to take regular readings of network flows received through data transmissions. This enables the company to improve network efficiency by accounting for water loss, and to better detect service failures and repair them the damage spreads. Yet this is saving money rather than earning money. Who pays the extra costs? Some smart cities, such as in India, have begun applying parking fees, water and sewage surcharges, telecom fees, and utility (gas/electric power) surcharges to help pay for the available Smart City technologies to provide stronger ROI and a viable approach to paying back secured funds.

OPEN DATA IS KEY

Open data is one of the keys to successful smart city execution. It can help identify and solve civic problems, ensure accountability of city officials, and create new business opportunities. Data about education, healthcare, transport, and tourism can be a catalyst for new app development, research projects, and targeted local campaigns. Amsterdam has had open city data since 2012, including census data, neighborhood growth, electricity usage, and roadworks.

Further, for many companies – both in the software and creative sector – it is typically difficult to develop smart city applications and applications because there is a lack of good testing environments. You require the consent of all parties (municipality, companies, etc.) to create an effective test environment. As the Amsterdam data and platform are open, it means developers can use the data to develop new applications and concepts.

SMART CITIES NEED TO BE BY THE PEOPLE, NOT SIMPLY FOR THE PEOPLE

Traditionally, urban problems have been solved through policy generated by municipal officials and urban planners in collaboration with the private sector. One of the great benefits of smart cities is that information and communications technologies like predictive algorithmic software, Big Data, and IoT can be utilized to streamline local government, transportation infrastructure, and the local environment to make it more sustainable and livable. At a local level, this means creating local solutions for local problems rather than just a top-down approach.

There's a certain irony that often upon attending pretty much any smart city event, the audience is comprised largely if not solely of stakeholders who have a vested financial or employment interest in smart cities: town planners, tech companies, energy companies, university academics, and the like without representation of local citizens whose daily lives are purported to improve by their creation. Most events require payment to attend and are held during business hours, making them prohibitive to those who are working or studying during such times. Cities need to creative in their efforts to engage citizens and ensure those less tech-savvy are not left behind.

A REAL UNDERSTANDING OF DATA PRIVACY, CITIZEN DATA AND CITIZEN ROI

The issue of CCTV and facial recognition show that smart city technology can be used for good and bad, depending on your opinion. For example, Moscow's smart city efforts were boosted last year with the implementation of 160,000 outdoor cameras focused on traffic and areas of possible crime. This is part of the Moscow Traffic Control Center, the headquarters of an elaborate monitoring and control system that also includes 40,000 traffic lights and a vast data storage facility that contains all the video data transmitted from the streets. The data has been used to fine citizens for violating road signs and signals with cameras recording license plates on cars.

China is already in the news for its Black Mirroresque Social Credit system. It's reported that by 2020, China will have completed a nationwide facial recognition and surveillance network that achieves near-total surveillance of urban residents, including in their homes via smart TVs and smartphones.

The capabilities of connected technology are expanding rapidly with the trajectory of the Internet of Things moving at a rapid pace bringing monumental benefits to the industry. If this isn't disturbing enough, there's another layer to consider: monetization.

It's well-known that data is the new oil, a particularly interesting idea driving the monetization of the Internet of Things and the creation of a "Machine Economy," where devices will trade everything from storage, computation, and analytics to electricity and sensor data. The trading of anonymized data from sensors may seem like a straightforward enough way to generate income, but what does it mean for residents already paying taxes to the city? Should they be reaping some of the rewards from data sold to third parties, particularly since they can't opt out? What does it mean for the concept of open data, one of the pinnacles of smart city development? It shows that if anything, smart cities are only going to get more complex as they mature and their technology advances. It's not all doom and gloom, I have high hopes for many aspects of smart cities, but we can expect some challenges along the way.

CATE LAWRENCE An experienced technology journalist and writer based in Berlin, Cate brings an understanding and focus on IoT, biohacking, and future technology to DZone. She likes to look behind the technology to see the challenges and opportunities its creation poses for developers, users, law makers, and society.

IoT, AI, and Low-Code: A Perfect Trifecta

Key technological developments have come of age in 2018, opening the door to a perfect trifecta:

1. With the widespread availability of cheap sensors, IoT is estimated to connect over 23 billion devices by the end of this year, up almost 4 billion from 2017, according to statistics firm Statistica.
2. Sophisticated cognitive services can now be accessed from the cloud on inexpensive computing devices, meaning artificial intelligence is available to anyone.
3. The low-code development platform market is predicted to grow from USD 4.32 billion in 2017 to USD 27.23 billion by 2022, at a CAGR of 44.5%.

With billions of connected devices comes the need to understand and process the data being generated. Enter AI algorithms to feed on the huge data sets to find patterns. But what's missing is a way to to harness the combined power of IoT and AI. Low-code platforms are the perfect glue, enabling developers to quickly connect different types of technology and build new solutions.

Why OutSystems?

1. Visual, model-driven development means "intelligent" applications can be built without expensive, hard-to find resources.
2. Rapid experimentation and built-in feedback enables organizations to test and validate new concepts.
3. Pre-built integrations with leading IoT and AI cloud services accelerate solution development.

Organizations taking advantage of the perfect trifecta can disrupt established industries, breaking new ground while bringing more value to customers.

WRITTEN BY MIKE HUGHES
PRODUCT MARKETING, OUTSYSTEMS

PARTNER SPOTLIGHT

OutSystems Low-Code Development Platform

OutSystems is the #1 low-code platform for digital transformation - build mobile apps, web portals, mission-critical systems, and more.

CATEGORY
High productivity, low-code application development and delivery platforms

OPEN SOURCE?
No, but we offer a free personal edition

NEW RELEASE
Annual major releases and monthly feature releases

CASE STUDY
ISB Global is disrupting the waste collection industry by using IoT and AI to optimize schedule and route planning for residential and commercial waste bin collection. Using OutSystems, ISB Global developed an advanced app to read IoT-enabled bin fill levels and forecast future fill rates. It determines when bins will reach the maximum fill level and defines the best collection time - allowing the routing engine to generate routes that meet service levels, minimize logistics cost, and maximize truck-load capacity. The app gives waste companies the visibility into collection requirements, including volume and weight, to achieve dramatic cost savings. This IoT app is the first big step towards a full 'Uberization' of the Waste Industry.

STRENGTHS

- Visual, low-code full-stack web and mobile application development
- Integration with everything: Pre-built connectors to leading AI/ML and IoT platforms
- Change tracking and automatic change impact analysis
- One-click deployment and continuous delivery for even the most complex apps
- High availability, scalability, and enterprise-grade security for running apps

NOTABLE CUSTOMERS

- FICO
- AXA Insurance
- Vodafone
- Volkswagen
- Randstad

WEBSITE outsystems.com | **TWITTER** @outsystems | **BLOG** outsystems.com/blog

Executive Insights on the State of IoT in 2018

BY TOM SMITH
RESEARCH ANALYST AT **DZONE**

QUICK VIEW

01. Understand how you are going to get value from all of the data you collect and how this fulfills a business need that will improve the customer experience.

02. Companies can get more out of IoT by solving a specific business problem. Doing so will generate revenue, reduce costs, and have an obvious ROI.

03. More companies are pursuing IoT strategies as they see use more cases driving ROI.

To gather insights on the current and future state of IoT, we talked to 23 executives involved with IoT. Here's who we spoke to:

- **Adam Fingerman**, CXO and Co-founder, and **Troy Petersen**, Marketing Director, **ArcTouch**
- **Andreas Pettersson**, CTO and CPO, **Arcules**
- **Sean Grundy**, CEO and Founder, **Bevi**
- **Jeff Bonnell**, V.P. of Industry Solutions, **Coresystems**
- **Eli Feldman**, CTO, Advanced Technology, **EPAM**
- **Brent Pietrzak**, V.P. Producer Solutions and Strategy, **Flexera**
- **Scott Allen**, CMO, **FreeWave**
- **Mark Herring**, CMO, **Tim Hall**, V.P. of Products, **Brian Mullen**, V.P. of Business Development, **InfluxData**
- **Dipti Borkar**, V.P. of Product Marketing, **Kinetica**
- **Crystal Valentine**, V.P. of Technology Strategy, **MapR**
- **Jesse Robbins**, CEO, **Orion Labs**
- **Lars Knoll**, CTO, **Qt**
- **Olivier Pauzet**, V.P. and General Manager IoT Solutions, **Sierra Wireless**
- **Jens-Ole Graulund**, CTO, **Spiio**
- **Monte Zweben**, CEO, **Splice Machine**
- **Shawn Reynolds**, CMO, **Telit**
- **Yu Xu**, CEO, **TigerGraph**
- **Ray Wu**, Founder and CEO, **Wynd**
- **Alex Kubicek**, CEO, **Understory**
- **Jeff Finn**, CEO, **zvelo**

KEY FINDINGS

1. The keys to a successful IoT strategy are to **understand how you are going to get value from all of the data you collect and how this fulfills a business need that will improve the customer experience.** Answer the following questions as a team: What is the product or service being created? What is the benefit being provided? What is the value you are adding? What customer needs are we fulfilling? How can we monetize it?

Start small, then scale and iterate. Use business insights from the data to solve problems and improve the solution you have built. Learn what data will be useful to customers, and what they're willing to pay for. Put the customer first and make it easy for the customer to see the benefits of your product or service.

2. Companies can get more out of IoT by **solving a specific business problem.** Doing so will generate revenue, reduce costs, and have an obvious ROI. Solve problems simply and easily, but don't mistakenly believe that IoT is easy, because it's not. There is a lot of complexity to make IoT solutions a reality, and most companies do not have the skillsets in-house to do so. In addition, most employees do not have the strategic vision to think about consumer or business use cases that drive revenue, efficiency, and ROI.

3. The biggest change in IoT in the past year is that **more companies are pursuing IoT strategies as they see use cases that have driven ROI.** Companies are learning the need to focus on their IoT business model as well as the ROI over three to four

years. Organizations are looking more closely at business cases before jumping into technology and this is important given the complexity and specialization of the technology by vertical. We have a lot of vendor stories, but not a lot of customer stories except in energy and manufacturing. Some companies are pursuing IoT initiatives in conjunction with their digital transformation initiatives since they are both heavily data-driven.

Additionally, some companies are leveraging artificial intelligence (AI) and machine learning (ML) for cross-sensor analytics as well as to generate real-time results and insights for better informed decision making.

4. The **technical solutions used for IoT initiatives vary greatly by industry** with specific sensors, data, and analytics tools are being used based on specific industries and use cases. The industry is becoming very complex, and more specific solutions are becoming necessary.

5. The industries with the most IoT applications shared by our respondents are **healthcare, manufacturing, oil and gas, and automotive.** The most frequent applications are **incident prevention, predictive maintenance, optimization, and autonomy.**

Several respondents are working with medical device manufacturers to perform preventive and predictive maintenance on MRI machines that are also aggregating images and analyzing them with ML to prevent and predict diseases, as well as recommend treatments.

A lot of respondents are working with different automotive OEMs to help build autonomous vehicles that know the best routes to reduce traffic, risk, and ensure safer transportation.

There are also a lot of manufacturing applications to optimize yield management, as well as using preventive and predictive maintenance so companies can bring machines down for maintenance and part replacement on a schedule rather than in an emergency situation, which can cost a production facility millions in lost productivity.

6. The most common issues mentioned preventing companies from fully realizing the benefits of IoT is a lack of talent and vision, as well as organizations' and individuals' **failure to identify the problem they are attempting to solve.**

Organizations do not have the people capable of performing the analysis necessary to take advantage of data to solve problems. There's an unrealized complexity of deploying IoT solutions and a lack of skills to do so. There's a general lack of awareness about the impact IoT can have and what problems it can solve, as well as a lack of understanding of what technologies to use.

The second most frequently mentioned problem has been a recurring theme – failure to identify the specific business need or problem to solve. Focus on the business value being created for customers. Taking a regular product and connecting it increases costs – but you must also increase the value to the end user.

7. The biggest opportunities in the continued evolution of IoT is around **AI, ML, DL (deep learning), predictive analytics with edge computing, and cross-sensor AI/ML models.** There is a clear intersection between AI and IoT, and the benefits for businesses will be transformational. Tasks that used to take humans weeks or months to complete will be done in minutes or seconds.

8. The primary concerns with the state of IoT today are **security, privacy, and the lack of standardization.** There's a concern over privacy and security with the potential of people or companies to have a backdoor into your home network, as well as an inability to update devices already in the field. There is also a lack of standards for device communication. No one seems to be interested in taking a leadership role and solving the issue.

The IoT industry seems to be relearning the same lessons regarding security that have evolved over time in other computing industries. Best practices, like the ability to update a device and restricting access to unauthorized users are rampant problems in the industry today.

9. A broad range of skills is needed for developers to be successful working on IoT projects. Those mentioned more than once include: **understanding the business context, knowing the IoT stack, mobile development, toolsets, data ingestion, cloud, and security.** It used to be OK to be a good developer, but IoT is complex. You need to understand the business context of the solutions you are building. Know mobile and cloud platforms, as well as the wide spectrum of firmware available. Use available tools to shortcut the process. Think about how the product will function and get updated when it's deployed.

TOM SMITH is a Research Analyst at DZone who excels at gathering insights from analytics—both quantitative and qualitative—to drive business results. His passion is sharing information of value to help people succeed. In his spare time, you can find him either eating at Chipotle or working out at the gym.

Solutions Directory

This directory of monitoring, hosting, and optimization services provides comprehensive, factual comparisons of data gathered from third-party sources and the tool creators' organizations. Solutions in the directory are selected based on several impartial criteria, including solution maturity, technical innovativeness, relevance, and data availability.

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
ADLINK Technology IST	Vortex	IoT data analysis	Healthcare, transportation, smart cities, energy	prismtech.com/vortex/overview
Aeris	Aeris	IoT platform, connectivity middleware	Healthcare, logistics, transportation, utilities	aeris.com/solutions
Afero	Afero	IoT platform, security, cloud infrastructure, connectivity middleware	Industrial IoT, wearables, medical, transportation, home automation	afero.io/platform
Altizon	Datonis	Connectivity middleware, device management, big data analytics	Smart city, industrial IoT, utilities	altizon.com/datonis-iiot-platform
Amazon	Amazon Echo	Consumer product	Home automation	amazon.com/echo
Amazon	AWS IoT Platform	IoT platform, connectivity middleware, device management	Smart city, transportation, healthcare	aws.amazon.com/iot
Apache Foundation	MyNewt	IoT operating system	Device management, connectivity	mynewt.apache.org
Apple	Apple HomeKit	Developer program, SDK	Home automation	developer.apple.com/homekit
Applied Informatics	macchina.io	IoT platform, messaging middleware	App development, device management	macchina.io
Arduino	Arduino Uno	Development board	Prototyping, hobbyists, DIY	arduino.cc
ARM	Mbed IoT Device Platform	IoT platform, operating system, device management	Prototyping, app development	mbed.com/en/platform

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Arrow	Dragonboard 410c	Development board	Prototyping, hobbyists, DIY	arrow.com/en/campaigns/the-dragonboard-is-here
Atmel	Atmel Microcontrollers	Microcontrollers	Hardware	atmel.com/products/microcontrollers
Ayla Networks	Ayla IoT Platform	IoT platform, messaging middleware	Home automation, wearables, logistics	aylanetworks.com/products
Ayyeka	Ayyeka Wavelets	Sensors	Utilities, smart city, industrial IoT	ayyeka.com
Beagleboard.org	BeagleBone Black	Development board	Prototyping, Hobbyists, DIY, Mobile	beagleboard.org/black
BestMile	BestMile	Smart car connectivity platform	Transportation	bestmile.com
Blue Pillar	Aurora	IoT platform	Industrial IoT, energy	bluepillar.com/aurora-energy-network-of-things-platform
Bluetooth	Bluetooth	Device connection	Hands-free point-to-point connections	bluetooth.com/bluetooth-technology
Blynk	Blynk	IoT mobile app	Prototyping, hobbyists, DIY, mobile	blynk.cc
Bosch Software Innovations	Bosch IoT Suite	IoT platform, device management	Industrial IoT, agriculture, home automation, transportation, logistics	bosch-si.com/iot-platform/bosch-iot-suite/homepage-bosch-iot-suite.html
Bridgera	Bridgera	Custom IoT Software Platform	Healthcare, Industrial, Security, Transportation	https://bridgera.com/
Buddy	Ohm	Smart building sensors	Smart buildings, smart cities	buddy.com/building-monitoring
Bug Labs	Dweet.io	IoT messaging platform	Device management, monitoring	dweet.io
Bug Labs	Freeboard	IoT device visualization	Monitoring, analytics	freeboard.io
Buoy	Buoy	Water system sensors	Home automation, utilities	buoy.ai
C3 IoT	C3 IoT Platform	IoT rapid application development platform	Logistics, smart cities, manufacturing	c3iot.com/products/c3-iot-platform
Canonical	Ubuntu Core	IoT operating system	Industrial IoT, robotics	ubuntu.com/core
Carmine	Carmine Telematics	Fleet management and monitoring	Logistics, transportation	carmine.io/telematics
Carriots	Carriots	IoT platform, device management, messaging middleware	App development, data analytics	carriots.com/what-is-carriots

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Carvi	Carvi	Smart car sensors	Transportation, logistics, insurance	getcarvi.com
Casa Jasmina	Casa Jasmina	Development community	Home automation	casajasmina.cc
Cisco	Cisco Internet of Things Dev Center	Networking, messaging middleware	Industrial IoT, smart city, data analytics	developer.cisco.com/site/iot
Cisco	Jasper Control Center	IoT platform, networking	Device management, analytics	jasper.com/control-center-for-iot
ClearBlade	IoT Edge Platform	IoT platform	Security, device management, data filtering	clearblade.com
Compology	WasteOS	Sensor network	Logistics, smart city	new.compology.com/technology
Concirrus	Concirrus Platform	IoT platform, device management	Insurance, data analytics	concirrus.com
Connected Technologies	Connect One	Networking	Device management, healthcare, industrial IoT, smart city, home automation	simplifywithconnectone.com
ConnectM	Yantra Cloud	IoT platform, load balancing, data analysis	Home automation, logistics, industrial IoT	connectm.com
Control4	Control4	Developer program, SDK	Home automation	control4.com
CoreRFID	CoreRFID	Sensors	Logistics, manufacturing, monitoring	corerfid.com
Couchbase	Couchbase Server	Engagement database	Key-value, document, data caching	couchbase.com/products/server
Cumulocity	Cumulocity	IoT platform, messaging middleware	Device management, analytics	cumulocity.com
Current	Daintree	Sensors, device management	Utilities, home automation, logistics	products.currentbyge.com/control-systems/wireless-lighting-controls
DataArt	DeviceHive	IoT platform, data analytics	Connectivity, analytics	devicehive.com
DGLogik	DGluk5	IoT platform	Agriculture, analytics, healthcare, industrial IoT, smart city, logistics	dglogik.com/products/dglux5-ioe-application-platform
Digi	Digi XBee3	Networking, sensors	Smart city, industrial IoT	digi.com
Digi-Key Electronics	Digi-Key	IoT electronics, development board	Wearables, hardware, DIY	digikey.com

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
DotMatrix Technologies	DotMatrix Connected Device Language	IoT platform	App development and deployment	dotmatix.net
Eclipse Foundation	Kura	Connectivity middleware	Device management	eclipse.org/kura
Eclipse Foundation	Vorto	IoT platform	Device management	eclipse.org/vorto
Edyn	Edyn Garden Sensor	Sensors	Agriculture	edyn.com
ElasticM2M	Elastic IoT Platform	IoT platform, IoT modules, data analytics	Transportation, marine, energy, smart buildings, climate	elasticm2m.com/platform/
Electric Imp	Electric Imp Platform	IoT platform, connectivity middleware, security	Utilities, industrial IoT	electricimp.com/platform
Embedded Micro	Mojo V3	Development board	Prototyping, hobbyists, DIY	embeddedmicro.com/products/mojo-v3.html
enModus	enModus	Smart lighting modules	Utilities, smart city, home automation	enmodus.com/products/
Eurotech	Everyware Cloud	IoT platform	Device management	eurotech.com/en/products/software+services/
Eurotech	IoT Development Kits	Development board	Prototyping, manufacturing	eurotech.com/en/products/iot/iot+development+kits
Evothings	Evothings Studio	Mobile app development platform, device management	Healthcare, home automation, industrial IoT	evothings.com
Exosite	Murano	IoT platform	Industrial IoT, home automation, healthcare	exosite.com/iot-platform
F5 Networks	Big-IP	Networking, load balancing	Industrial IoT, smart city	f5.com/products/big-ip
Filament	Filament	Networking	Industrial IoT	filament.com
FitBit	Fitbit	Developer program, API	Wearables	dev.fitbit.com
Flexera	Flexera	Software & tech management	IT asset management	flexera.com
Gadget Factory	Papilio Wiki	Development board	Prototyping, Hobbyists, DIY, Mobile	papilio.cc
GE	GE Predix	IoT platform, operating system	Industrial IoT	ge.com/digital/predix-platform-foundation-digital-industrial-applications

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Gizmosphere	Gizmo 2	Development board	Prototyping, Hobbyists, DIY, Mobile	element14.com/community/community/designcenter/single-board-computers/gizmo2
Google	Android Wear	Developer program, consumer product, API	Wearables, mobile	android.com/wear
Google	Android Studio	IoT development platform	Wearables, mobile	developer.android.com/studio/index.html
Google	Nest Developers	Developer program, smart thermostat, API	Home automation	developers.nest.com
Google	Jacquard	Connected clothing	Device management	atap.google.com/jacquard
Greenwave Systems	AXON Platform	Messaging middleware, IoT gateway, analytics	Networking, sensors, mobile	greenwavesystems.com/product/axon-iot
Helium	Helium Smart Sensors	Sensors	Home automation, environmental, analytics	helium.com/solutions
Huawei	Huawei	Networking	Industrial IoT, smart city, agriculture, environmental	huawei.com/minisite/iot
IBM	Bluemix	IoT platform	App development, big data analytics	ibm.com/bluemix
Imprint	RIOT	Operating system	IoT hardware and app development	riot-os.org
InfluxData	InfluxData	Time series platform	Data ingestion, real-time querying	influxdata.com/products
infswift	infswift IoT Platform	IoT platform, stream processing, data analytics	Energy, agriculture, climate, smart cities, home automation, transportation	infiswift.tech
Insteon	Insteon Hub	Sensors, developer program, API	Home automation	insteon.com/which-hub-are-you
Intel	Up Squared Grove IoT Development Kit	Development board	Prototyping, hobbyists, DIY	software.intel.com/en-us/iot/hardware/up-squared-grove-dev-kit
InterSystems	InterSystems IRIS Data Platform	Data platform	Transaction processing, data analytics	intersystems.com/products/intersystems-iris
Itron	Silver Spring	Networking	Industrial IoT, smart city	silverspringnet.com/solutions
Jawbone	Jawbone UP	Developer program, API	Wearables	jawbone.com/up/developer

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Kentix	Kentix360	Sensors	Home automation, home security	kentix.com/en/solutions-and-more/kentix360
Kontakt.io	Location Engine	Messaging middleware	Networking, smart city, industrial IoT, home automation, agriculture	kontakt.io/location-engine
Kontakt.io	Kontakt Beacons	Beacons, sensors	Networking, smart city, industrial IoT, home automation, agriculture	store.kontakt.io
leakSMART	LeakSmart	Sensors	Home automation, industrial IoT	getleaksmart.com
Lightbend	Fast Data Platform	Fast data applications	Microservices, machine learning, intelligent security management	lightbend.com/products/fast-data-platform
LinkLabs	Symphony Link	Development board modules, routers, API	Prototyping, industrial IoT	link-labs.com/symphony
Litmus Automation	LoopEdge	Edge computing platform for IIoT	Industrial IoT	litmusautomation.com/loopedge-2
Litmus Automation	LoopCloud	Data management platform, device management	Device management	litmusautomation.com/loopcloud
Logmein	Xively	IoT platform, connectivity middleware	Industrial IoT	xively.com
Marvell	Kinoma Create	Development board	Prototyping, hobbyists, DIY	kinoma.com/create
Marvell	Kinoma Element	Development board	Prototyping, hobbyists, DIY, home automation	kinoma.com/element
Mender	Mender	IoT platform	DevOps, automated deployment	mender.io/product/features
Meshdynamics	Meshdynamics	Surveillance, hardware, networking	Security, monitoring, industrial IoT, smart grid	meshdynamics.com
Meshify	Meshify Enterprise	IoT platform, messaging middleware	Device management, industrial IoT	meshify.com/applications
Microduino	mCookie	Development board modules	Prototyping, hobbyists, DIY	microduinoinc.com/products/mcookie

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Microsoft	Azure IoT Suite	IoT platform, data analytics	Device management, analytics, industrial IoT	microsoft.com/en-us/internet-of-things/azure-iot-suite
Microsoft	Windows 10 IoT Core for NXP	IoT operating system	Robotics, industrial IoT, home automation	developer.microsoft.com/en-us/windows/iot
mnubo	SmartObjects	Predictive analytics for devices	Manufacturing, industrial IoT, agriculture, home automation	mnubo.com/solution/platform/
Mojio	Mojio	IoT platform	Transportation, smart cars	moj.io/connected-car-platform
MuleSoft	MuleSoft Anypoint Platform	Connectivity middleware	Device management, connectivity	mulesoft.com/platform/enterprise-integration
Muzzley	Q by Muzzley	IoT platform	Consumer IoT monetization	q.muzzley.com
Muzzley	Muzzley	IoT platform	Device management, home automation	smarthome.muzzley.com
MyDevices	Cayenne	IoT app development platform, messaging middleware	Data analytics	mydevices.com/cayenne/features
Myriad Sensors	PocketLab	Sensors	Environmental	thepocketlab.com
NATS	NATS	Messaging middleware	Connectivity	nats.io
Netvibes	Dashboard of Things	Analytics platform	Home automation	netvibes.com/en/explore/dashboard-of-things
Neura	Neura	IoT platform	Healthcare	theneura.com
Node-RED	Node-RED	IoT platform	Connectivity, device management	nodered.org
NPM	HomeStar	Messaging middleware	Home automation	github.com/dpjanes/node-iotdb/blob/master/docs/homestar.md
NXP	Kinetis Cortex-M MCUs	ARM processors and microcontrollers	Hardware	nxp.com/products/processors-and-microcontrollers
Onion	Omega2	Development board modules	Prototyping, hobbyists, DIY	onion.io/omega2

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Open Hybrid	Open Hybrid	IoT platform	App development, prototyping	openhybrid.org
Oracle	Oracle Internet of Things Cloud Service	IoT platform	Device management, analytics	oracle.com/solutions/internet-of-things
OutSystems	OutSystems	Enterprise app development platform	Build, deploy, & manage enterprise apps	outsystems.com/platform
Particle	Device Cloud	Networking, messaging middleware	Device management	particle.io/products/software/device-cloud
Particle	Particle Photon	WiFi development board	Prototyping, hobbyists, DIY	particle.io/products/hardware/photon-wifi
Pebble	Pebble Watch	Development program, smart watch, SDK	Wearables	developer.getpebble.com
Philips	Philips Hue	Development program, SDK	Home automation	developers.meethue.com
PlatformIO	PlatformIO IDE	Development environment	Embedded development, debugging	platformio.org/platformio-ide
PlatformIO	PlatformIO Core	IDE, IoT platform	App development	docs.platformio.org/en/latest/core.html
Prodea	Arrayent IoT Services Platform	IoT platform, data analytics, device management	App development	prodea.com/iot-services-platform
PTC	Thingworx	IoT platform	App development, device management, big data analytics	ptc.com/en/products/iot/thingworx-platform
PubNub	PubNub Data Stream Network	Networking, messaging middleware	Connectivity, device management	pubnub.com/products/global-data-stream-network
Raspberry Pi	Raspberry Pi 3 Model B+	Development board	Prototyping, hobbyists, DIY	raspberrypi.org
Razer	Razer Nabu	Developer program, SDK	Wearables, video games	razerzone.com/nabu
Red Hat	Red Hat JBoss AMQ	Connectivity middleware	Messaging device management	redhat.com/en/technologies/jboss-middleware/amq
Reekoh	Reekoh	IoT platform, connectivity middleware	Device management	reekoh.com
Relayr.io	Relayr	Device management, machine learning analytics, connectivity middleware	Business scaling	relayr.io/en/iot-middleware-platform

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Remforce	Remforce Boiler and Leak Monitoring	Sensors	Home automation	remforce.com/base-boiler-kit
resin.io	resin.io	IoT platform	DevOps, automated deployment	resin.io/features
RTI	Connext DDS	Connectivity middleware	Industrial IoT	rti.com/products/dds
Runtime.io	Runtime	Device management, Apache MyNewt management	Device management, connectivity	runtime.io
Salesforce	Salesforce IoT	IoT platform, device management, data analytics	Connected devices, app development	salesforce.com/products/salesforce-iot/overview
Samsara	Samsara	Sensors	Logistics, utilities, industrial IoT, environmental, transportation	samsara.com
Samsung	ARTIK Modules	Development board	Prototyping, hobbyists, DIY	artik.io/modules
Samsung	ARTIK Cloud	Data management platform, device management	Data analytics	artik.cloud
Scanalytics	Scanalytics SoleSensors	Sensors	Analytics, industrial IoT, home automation, retail	scanalyticsinc.com
Seeed Studio	Wio	Development boards and modules	Prototyping, hobbyists, DIY	seeedstudio.com/series/Wio-11.html
Sense	Sense	Sensors	Home automation, utilities	sense.com/product.html
Sierra Wireless	Sierra Wireless Embedded Solutions	Embedded modules, routers, IoT gateways	Hardware	sierrawireless.com/products-and-solutions/embedded-solutions
Sigfox	Sigfox	Networking, IoT connectivity	Industrial IoT	sigfox.com
Slock.it	Etherium Computer	IoT device rental	Device management, home automation	slock.it/ethereum_computer.html
Stream Technologies	IoT-X	IoT platform, connectivity middleware, networking	Smart cities	stream-technologies.com/iotx
Structural Health Systems	Concrete Sensors	Sensors	Industrial IoT, construction	concretesensors.com
Telit	Telit IoT Portal	IoT connectivity, IoT platform	Smart transportation, agriculture, retail, healthcare, automotive, smart cities	telit.com/m2m-iot-products/iot-platforms/telit-iot-portal
Telit	Telit IoT Modules	Development board modules, routers, API	Smart cars, cellular communication, networking	telit.com/m2m-iot-products/iot-module-selector

COMPANY	PRODUCT	PRODUCT TYPE	VERTICAL	WEBSITE
Temboo	Temboo	IoT development platform	Smart energy, smart cities, logistics, manufacturing	temboo.com/iot-applications
Tessel.io	Tessel 2	Development board	Prototyping, hobbyists, DIY	tessel.io
Texas Instruments	TI LaunchPads	Development boards and modules	Prototyping, hobbyists, DIY	ti.com/tools-software/launchpads/launchpads.html
thethings.iO	thethings.iO	IoT platform	Agriculture, logistics, industry, smart home, smart cities	thethings.io/iot-dashboards-features
Tridium	Niagra 4	IoT app development platform, analytics	Industrial IoT, smart cities	tridium.com/en/products-services/niagara4
Ubidots	Ubidots	IoT platform, data analytics	Hobbyists, DIY, logistics, home automation, industrial IoT	ubidots.com/features
U-Blox	U-Blox	Development board WiFi modules	Prototssyping, connectivity, networking	u-blox.com/en/product-search
Verdigris	Verdigris	Sensor network and platform	Smart buildings, smart cities	verdigris.co
Verizon	Hum	Sensors, mobile app	Smart cars	hum.com
VersaSense	VersaSense	Sensors	Home automation, industrial IoT, hobbyists, DIY	versasense.com
VSCP	VSCP	IoT platform	Device management, control protocol	vscp.org
WebNMS	WebNMS IoT Platform	Connectivity middleware, analytics	Energy, smart cities, logistics	webnms.com/iot/unified-webnms-iot-platform.html
Wilderness Labs	Netduino 3 WiFi	Development board modules	Prototyping, hobbyists, DIY	wildernesslabs.co/Netduino
Wiring	Wiring	Microcontroller IoT platform	Prototyping, hobbyists, DIY	wiring.org.co
WSO2	WSO2 IoT	Connectivity middleware	Device management, connectivity	wso2.com/iot
Wyliodrin	Wyliodrin STUDIO	IDE, IoT platform	Prototyping, hobbyists, DIY	wyliodrin.com
Yeti	Yeti	IoT platform	Home automation	getyeti.co
Zebra Technologies	Zebra Savanna	IoT platform	Data intelligence, configuration management	zebra.com/us/en/cpn/savanna.html
Zolertia	RE-mote	Development board	Prototyping, hobbyists, DIY	zolertia.io/product/re-mote-professional-pack

GLOSSARY

ACTUATOR: A mechanism that performs a physical task based on input from a connected system.

AGENT: A piece of software that autonomously makes decisions for a user or another agent proactively and reactively.

ADVANCED MESSAGE QUEUING PROTOCOL (AMQP): An open application layer protocol for message-oriented middleware with a focus on queuing, routing (P2P, PubSub), security, and reliability.

ARDUINO SHIELD: An expansion board for Arduino devices that is mounted on top of the original board. Shields add new features in terms of connectivity, computational power, and so on.

BLUETOOTH LOW ENERGY (BLE): A wireless personal area network (PAN) aimed at devices with reduced power consumption and cost while maintaining a similar communication range to regular Bluetooth.

COMPETING CONSUMERS: A messaging patterns in which more consumers get messages from a common source (i.e. queue) but each message is delivered to only one consumer.

CONNECTED DEVICES: Components that make up the Internet of Things. Many have built-in sensors and/or actuators and collect data to help users or other devices make informed decisions and monitor or affect outside events.

DATA FILTRATION: A part of the Edge Layer that reduces the amount of transmitted information but retains the meaning of it.

DIRECT MESSAGING: A messaging mechanism in which the sender and receiver are directly connected or can exchange messages through one or more intermediate hops, which do not take ownership of each message but just forward it (routing).

EDGE GATEWAY: The connecting factor between device analytics and cloud data processing and analytics.

EDGE LAYER: An architectural shift in IoT that breaks the norm of the traditional client-server model. This is the first layer of connectivity for devices to connect to before going to the server. Responsible for the local connectivity of devices and for managing the data collection and connection to this server.

INDUSTRIAL INTERNET: The integration of machine learning, big data technology, sensor data, and machine-to-machine communication automation. This is done with the knowledge that the Internet of Things will be scaled and driven by enterprises. The idea is that smart machines can more accurately capture and communicate data to help corporations find problems sooner and increase overall efficiency.

INTERNET OF THINGS (IOT): A network of objects (such as sensors and actuators) that can capture data autonomously and self-configure intelligently based on physical-world events, allowing these systems to become active participants in various public, commercial, scientific, and personal processes.

IOT CLOUD PLATFORM: A cloud platform that provides a set of services that simplify the integration process between the services provided by cloud platforms and IoT devices. Some platforms include development tools and data analytics capabilities.

IOT DEVELOPMENT BOARD: A board that can be used to prototype and create IoT hardware. There are several boards available on the market with different features.

MESSAGING PROTOCOLS: The way information is transferred and communicated amongst devices, the cloud, and data storage. Different protocols are used for different results.

MESSAGE QUEUING TELEMETRY TRANSPORT (MQTT): A lightweight messaging protocol that runs on the TCP/IP protocol. It is designed for communicating with small devices in remote locations with low network bandwidth.

MICROCONTROLLER (MCU): A small computer on a single integrated circuit designed for embedded applications and used in automatically controlled embedded systems.

MULTI-AGENT SYSTEM: A network of multiple agents which act in an environment and interact or communicate with each other to achieve their design objective.

OPERABILITY: The measure of how well a software system works when operating in production, whether that is the public cloud, a co-located datacenter, an embedded system, or a remote sensor forming part of an Internet of Things (IoT) network.

PEER-TO-PEER COMPUTING: A network or architecture that splits computing tasks between several different nodes, called peers.

RELEASABILITY: The ability to quickly deploy changes to a software system, but also to quickly recover from disaster and adapt to changing technical and business challenges.

SENSOR: A device or component that perceives and responds to physical input from the environment.

SENSOR NETWORK: A group of sensors with a communications infrastructure intended to monitor and collect data from multiple locations.

SMART CITY: A residential or urban area that uses sensors and data to make decisions related to the well-being of its citizens. This can include improved traffic patterns, air quality, and parking lot monitoring.

STORE AND FORWARD: A messaging mechanism in which a broker is involved between sender and receiver, so that the broker gets ownership of the message from the sender, stores it for reliability and then delivers the message itself to the receiver.

TIME-SERIES DATABASE: Databases that are designed to store the time at which data is collected in addition to the data itself.