



IoT overview

# Microsoft Developer School

30 листопада 2017 року, Київ

Sergiy Poplavskyi,  
Software Engineer  
SCE CEE, Microsoft

<https://aka.ms/msdevschoolnov>

#msdevschool



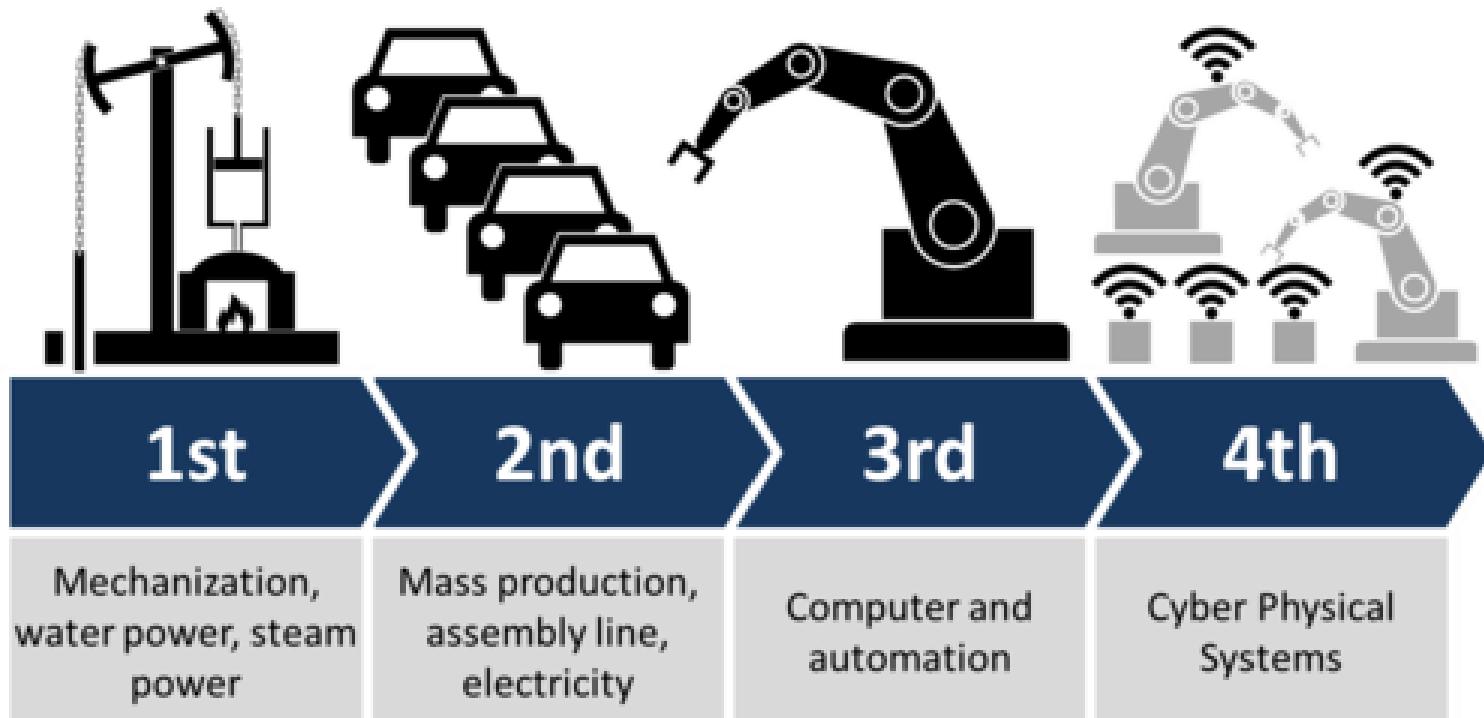
# Industry 4.0 & IoT



# Industry 4.0

Industry 4.0 is a name for the current trend of automation and data exchange in manufacturing technologies. It includes cyber-physical systems, the Internet of things, cloud computing and cognitive computing.

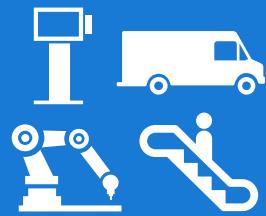
Industry 4.0 creates what has been called a "smart factory". Within the modular structured smart factories, cyber-physical systems monitor physical processes, create a virtual copy of the physical world and make decentralized decisions. Over the Internet of Things, cyber-physical systems communicate and cooperate with each other and with humans in real time.



# Internet of things

Internet of Things: A network of internet-connected objects able to collect and exchange data using embedded sensors.

Things



Connectivity



Data

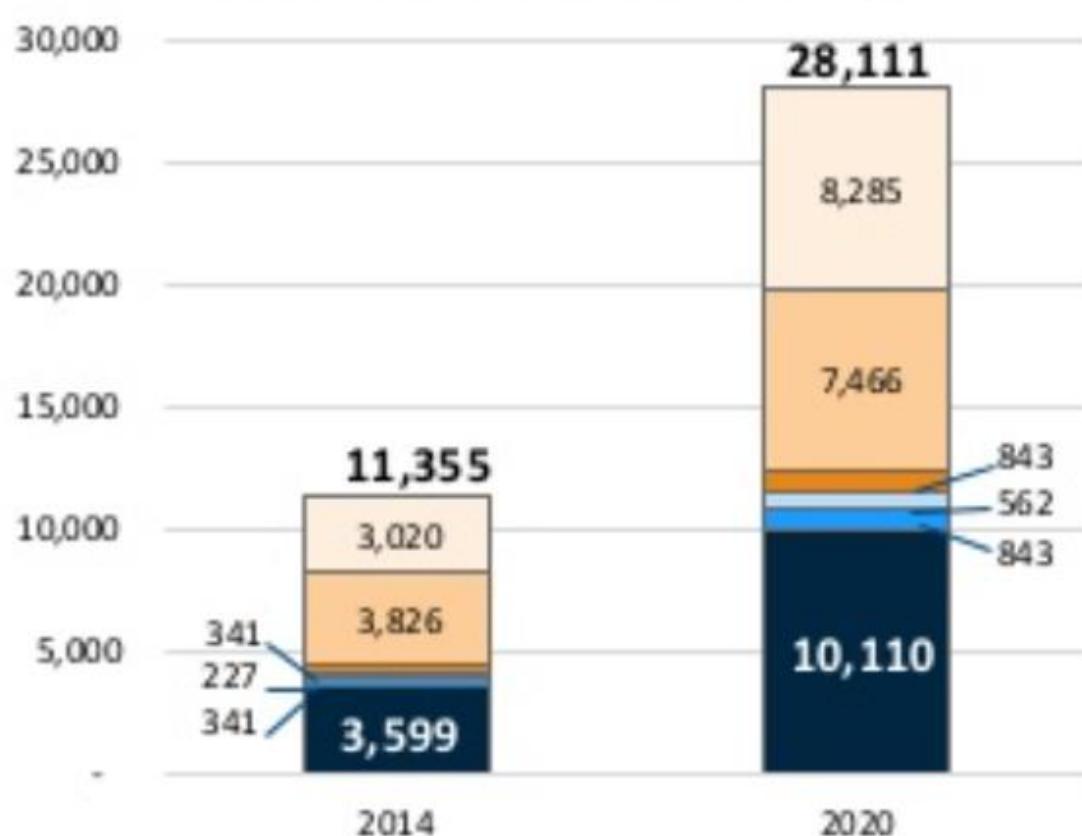
10101  
01010  
00100

Analytics



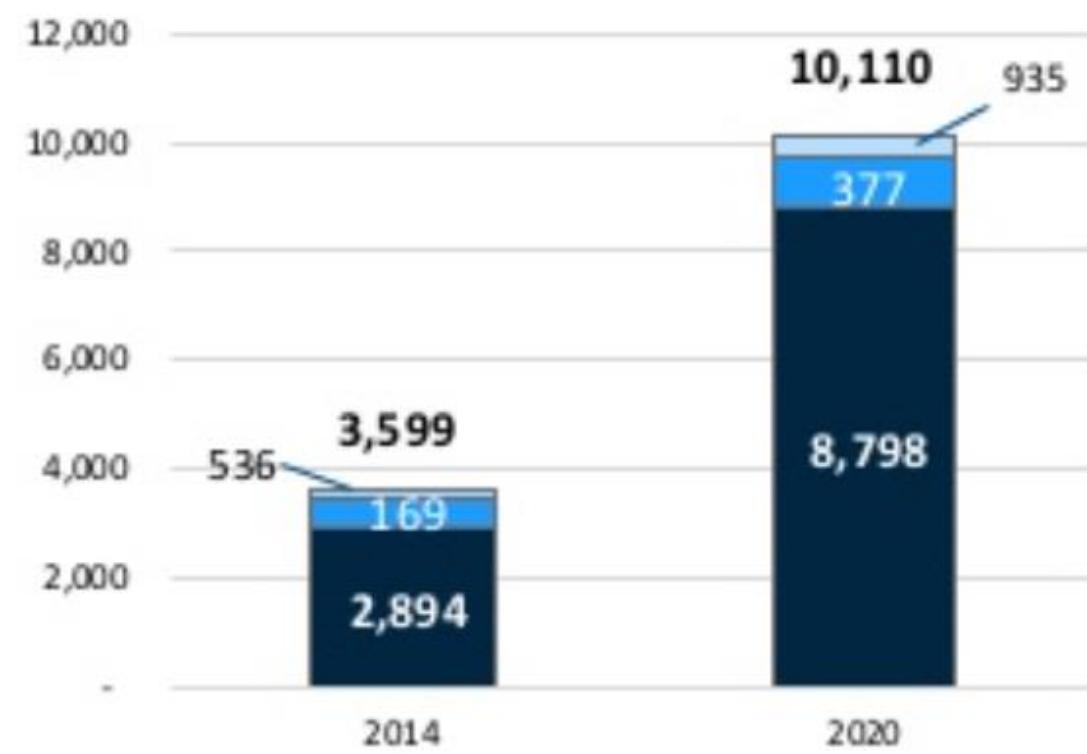
# The Internet of Things Market Size

Millions of IoT Units Installed\*



AP Breakdown

Millions of IoT Units Installed\*



■ AP ■ C&E Europe ■ Latam ■ MEA ■ NA ■ W Europe

■ APeJ ■ Japan ■ Rest of AP

\* Includes Industrial IoT and Non-wearable Consumer IoT, not Wearable Consumer IoT units

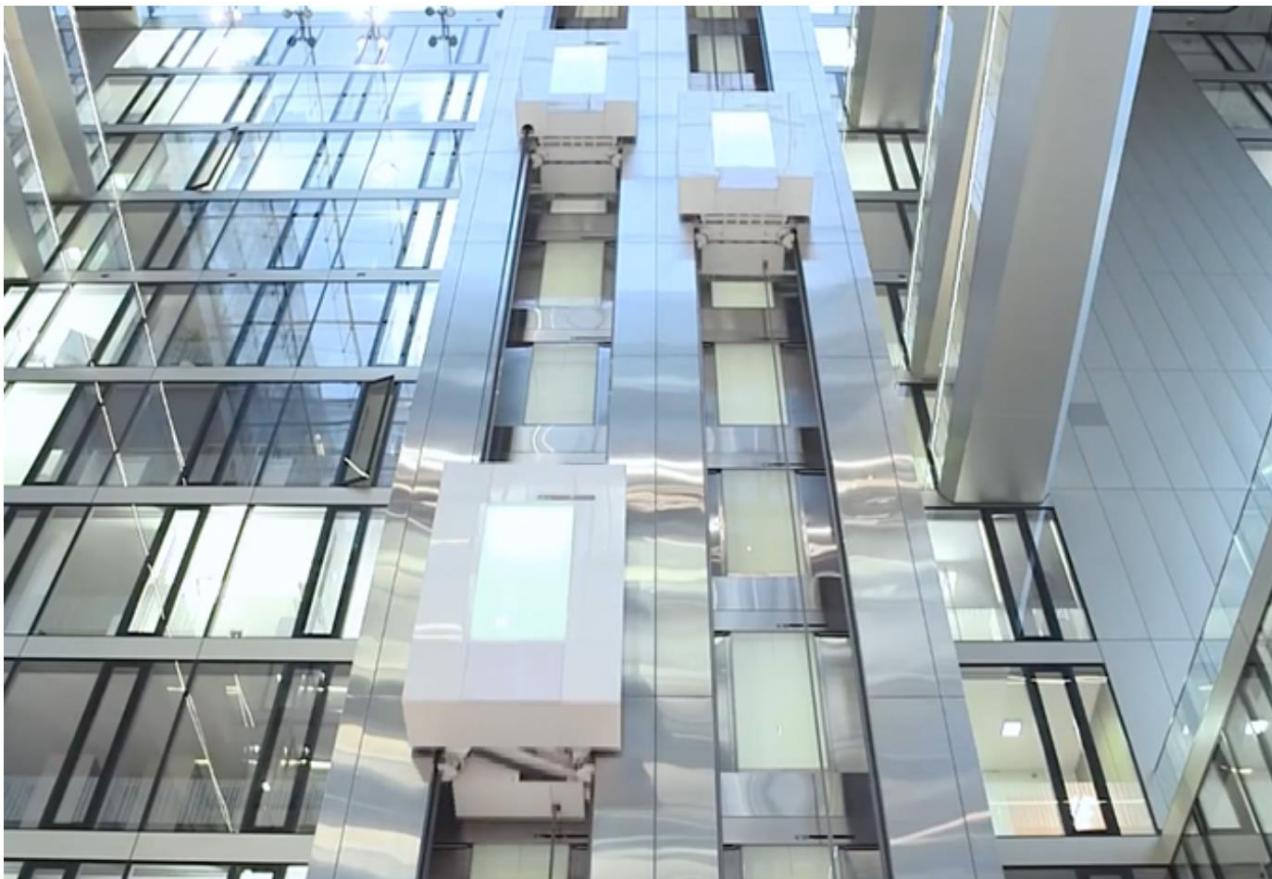
# IoT real-life use cases and scenarios

| Industry       | Use case   | Scenarios patterns   |
|----------------|--|--|
| Manufacturing  | Connected Vehicle<br>Food Traceability<br>Production Asset Mgmt<br>Manufacturing Operations<br>Connected Field Service                     | Device Connectivity & Management<br>Multi-protocol support<br>Connect legacy devices without replacing existing infrastructure |
| Consumer       | Home Security<br>Smart Appliances  | Remote Monitoring<br>Command and Control<br>Service customer equipment   |
| Government     | Automated Public Transit<br>Environmental Monitoring Detection<br>Public Infrastructure Asset Mgmt<br>Public Safety and Emergency Response | Edge computing<br>Security & Environmental Monitoring<br>Real-time analytics   |
| Healthcare     | Remote Health Monitoring<br>Clinical Care<br>Personal Wellness<br>Connected Medical Device   | Data Transformation and Routing<br>Predictive Maintenance<br>Role Based Access Control   |
| Retail         | In-Store Consumer Digital Offer<br>Personalized promotion  | Alerts and Notification<br>Data Visualization  |
| Transportation | Air traffic monitoring<br>Asset Fleet management<br>Freight monitoring   | Integration with other business services<br>Process atomic and batch data stream<br>Manage Intermittent-Connected Devices      |
| Utility        | Smart grid<br>Smart Building   | Data security from connectivity to storage<br>Geo-availability   |



# Examples from Microsoft

More on: <https://www.microsoft.com/en-us/internet-of-things/customer-stories>



## thyssenkrupp brings a new vision to elevator maintenance

Already using the predictive maintenance capabilities of [Azure IoT Suite](#) to reduce downtime and costs, thyssenkrupp is taking their IoT solution even further by enhancing their connected field service with HoloLens. Now their field technicians can access remote assistance to better identify problems and make time-saving interventions at the 1.1 million elevators the company maintains worldwide.

[Watch the video ▶](#)

[Read the full case study >](#)



## The RAC revolutionizes roadside assistance with Microsoft IoT

A 120-year-old motorist organization, the [RAC](#) provides roadside assistance, insurance, accident management and route planning to 8.6 million business and individual motorists across the U.K. by implementing [Microsoft Azure IoT Hub](#) to connect and monitor telematics devices placed in member vehicles.

[Read the full blog post >](#)



Examples from <http://iotlist.co>

# Amazon Dash Button: Smart home shopping



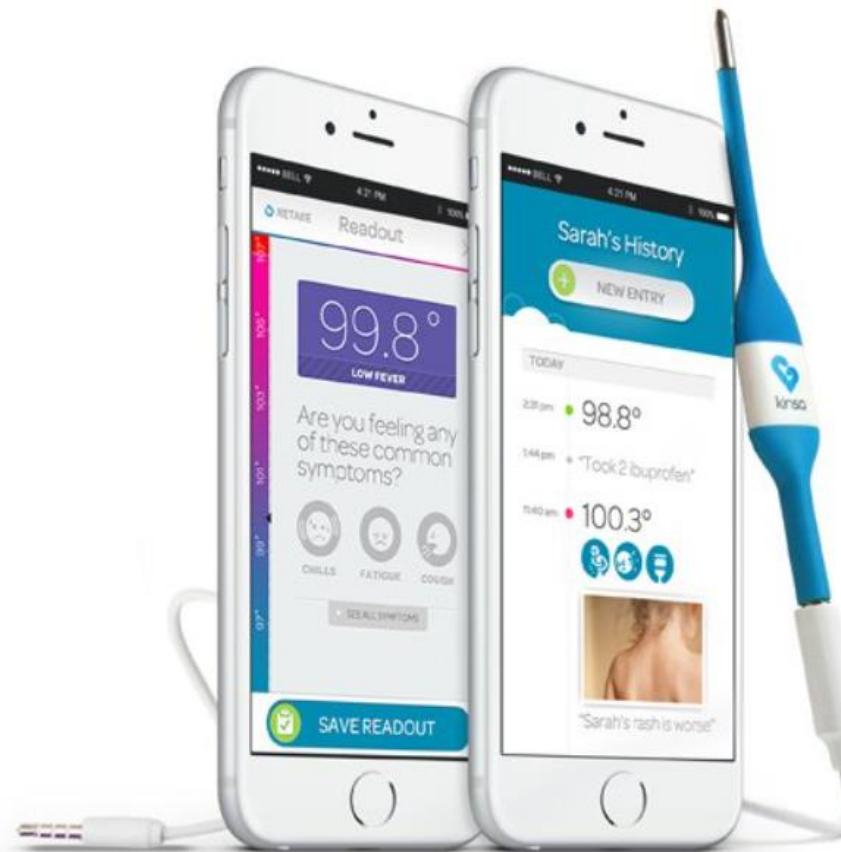
Amazon has been to great job of moving into the smart home over the past year. The Dash Button is not exception. A consumer can place a button anywhere in the home/business and simply push the button when the product needs to be refilled. This is a great smart home

# Lutron Caseta: HomeKit-enabled smart lighting kit



Homekit products continue to roll out. We tested the original Lutron Caseta system and found it to be easy to setup and use. With the latest smart bridge you can now control your lights with Siri.

# Kinsa: Smart thermometer



The Kinsa is a cool product that reinvents the way we think about taking a patient's temperature at home. It is simple to setup and makes it easy to track the health of a loved one who has the flu and help them get better. It is also a great product for new parents.

# Click & Grow: Smart herb garden



Click & Grow is a simple way to bring fresh herbs into your life. It's a great option for renters who want to bring greenery into their home and do not have a green thumb. We tested the product and liked the ease of use and the addition of fresh herbs to our office mojitos :)

# OpenSprinkler: Automate your sprinklers



Keeping your lawn and flowers beautiful doesn't have to be stressful. OpenSprinkler unchains you from your sprinkler or irrigation control box, enabling you to program, run, or stop zones at any time from anywhere.



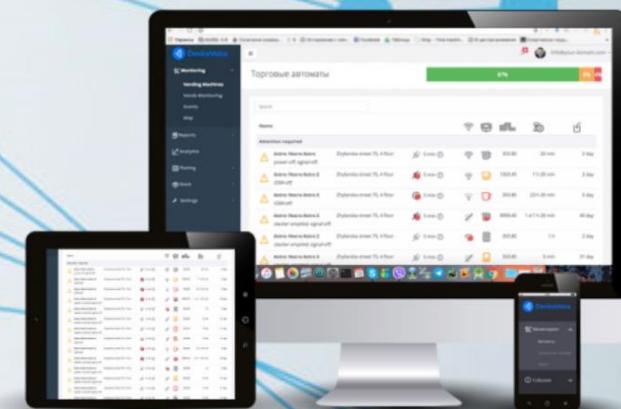
# Examples from IoT Laboratory

# FROM PRECISION TO DIGITAL AGRICULTURE



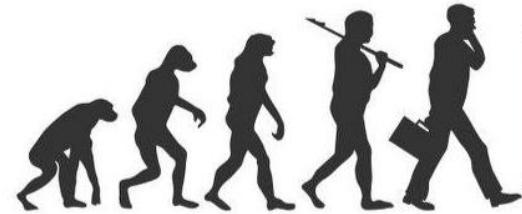
# Smart Vending Machines

24-hour machines operation monitoring. Technical conditions control and Accounting of goods and money in real time.

[Show all features](#)

# 3-COLORS ELECTRONIC Shelf Labels

# **FTS opens THE EVOLUTION of RETAIL**



**Truck №123456** is currently here:  
**GPS coordinates:**  
50.1056979, 33.779482



## PACKAGE 1:

Presence: Yes

Temperature: 18.5°C

Shock: No

Sealing: Not Activated



## PACKAGE 2:

Presence: Yes

Temperature: 32.5°C

Shock: Yes

Sealing: Not Activated



PACKAGE 3:

Presence: Yes

Temperature: Not Activated

Shock: Not Activated

Sealing: Yes



ТЕХНОЛОГІЧНІ ІННОВАЦІЇ ДЛЯ ВАШОГО ЖИТТЯ

ОБЛАДНАННЯ:

БПЛА

СПЕКТРАЛЬНА КАМЕРА

GPS-ТРЕКЕРИ

ДАТЧИКИ

СЕРВІСИ:

РОЗУМНИЙ ДІМ

СУПУТНИКОВІ ДАНІ

МЕТЕОРОЛОГІЧНИЙ СЕРВІС

КАРТОГРАФІЯ

ХМАРНЕ СХОВИЩЕ



Mr. Joe Black  
Puerto Cortes, Honduras



25,498



145,369



2,487,521



Tanya Suhodolska  
Alright got it!



Eugene Merill  
Sent



Wells Riley  
Typing...



Matt Vickers  
Not delivered

## ☰ Project **Anryze**

EXPLORE

SALES INFOGRAPHIC

WEEK

MONTH

YEAR



\$157,182  
Total Earnings

\$38,952  
Revenue

+800k  
New Customers

NEXT EVENT IN  
03 : 45 : 54  
HOUR MIN SEC

Adults  
Elders  
Kids



eNaturr

## Ідея

Проект eNaturr – це комплексне рішення з надання користувачам (B2B, B2C) інформації про стан навколошнього середовища, його безпеку та можливі еко-загрози, і все це в зрозумілій та доступній формі – прямо на смартфон

Доступні дані про стан середовища в будь-якому куточку світу

Простий та зрозумілий інтерфейс

Система “Асистент” повідомляє про важливі зміни автоматично

Можливість повідомляти про стан середовища відповідні інституції

Власні апаратні рішення

## OUR SECRET INNOVATIONS

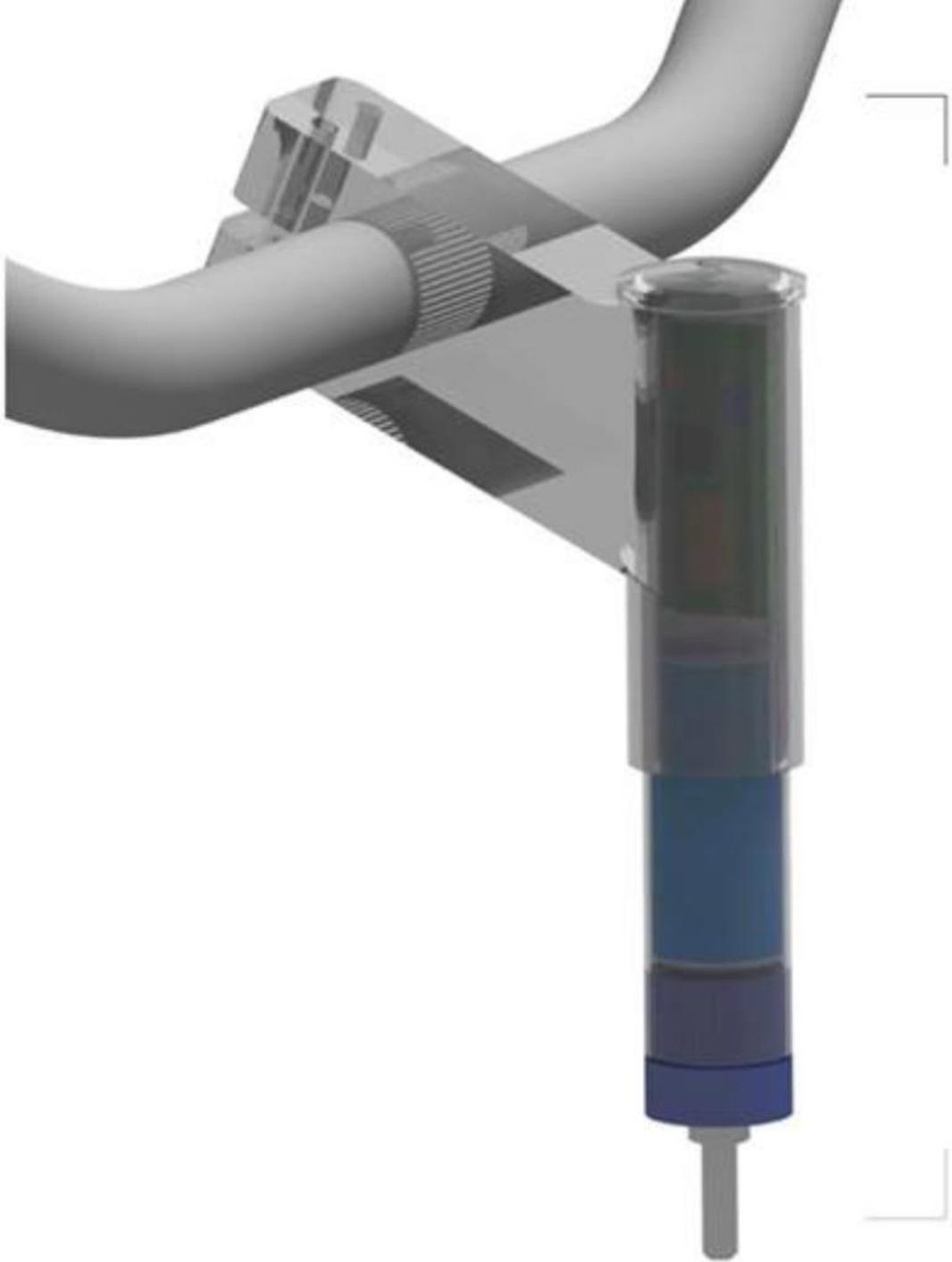
**We hid in the «Finder» to the handlebar and now no one knows what the bike is always under control**

«Finder» has GPS&GSM module, it works via Satellites and the Cellular Network for the transmission of location Data.

USB battery charging.

Everyone who use device receives a personal account with the pin tracking online.

Every device has ID and information about the owner. To use «Finder» there is not need a mobile phone or tablet. The device works independently wherever is mobile network.





# Well-known from Ukraine



# Petcube



## DISCOVER YOUR HOME POWER

Ecoisme is a home energy monitor that tells you how much energy your home appliances use and how much money you are spending on them. It eliminates grid and solar energy overuse and notifies you if something is about to break.

 Pre-order NowWatch video 

# Новое поколение беспроводных систем безопасности

В комплект Ajax StarterKit входит все необходимое для защиты помещения. Чувствительные датчики контролируют двери и активность внутри, а мощный хаб связывает их с внешним миром.

В экосистему Аях входят датчики нескольких типов, которыми вы по желанию можете дополнить защиту StarterKit.



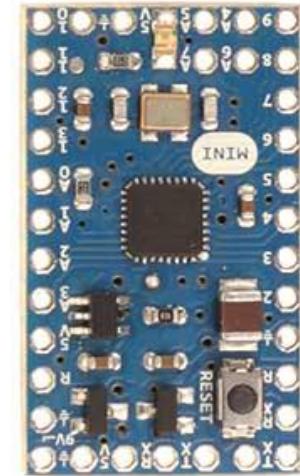
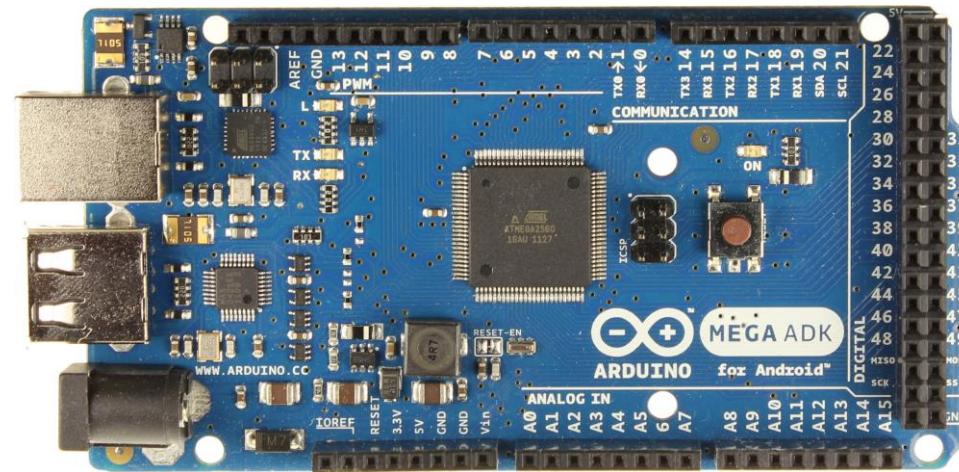
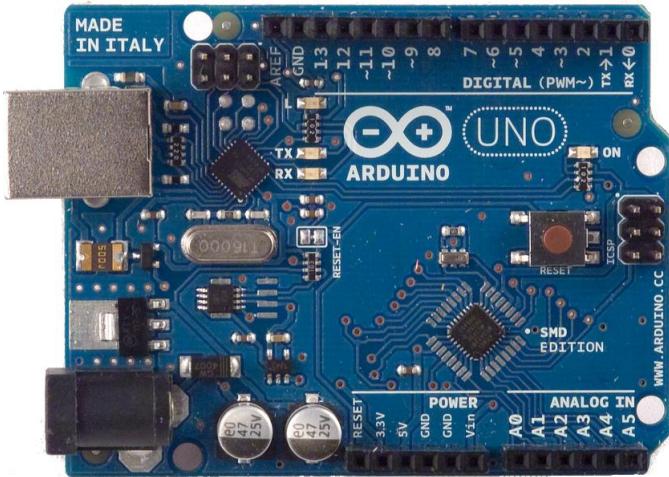


# Hardware





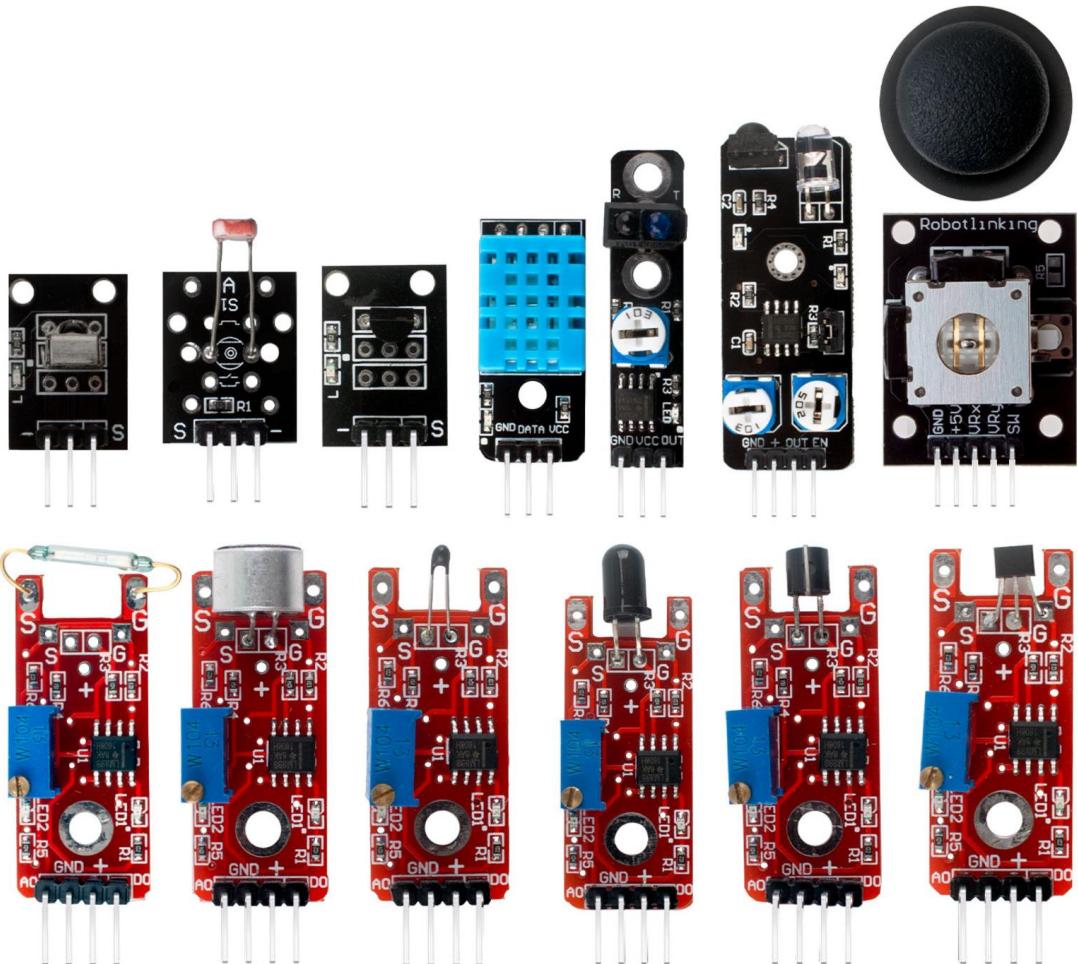
# Arduino (Atmel 8bit AVR MCU)



|                                    |                                    |                                |   |
|------------------------------------|------------------------------------|--------------------------------|---|
| <b>Microcontroller</b>             | <a href="#">ATmega328P</a>         | <b>DC Current for 3.3V Pin</b> | 50 mA   |
| <b>Operating Voltage</b>           | 5V                                 | <b>Flash Memory</b>            | 32 KB (ATmega328P) of which 0.5 KB used by bootloader |
| <b>Input Voltage (recommended)</b> | 7-12V                              | <b>SRAM</b>                    | 2 KB (ATmega328P)                                     |
| <b>Input Voltage (limit)</b>       | 6-20V                              | <b>EEPROM</b>                  | 1 KB (ATmega328P)                                     |
| <b>Digital I/O Pins</b>            | 14 (of which 6 provide PWM output) | <b>Clock Speed</b>             | 16 MHz  |
| <b>PWM Digital I/O Pins</b>        | 6                                  | <b>LED_BUILTIN</b>             | 13  |
| <b>Analog Input Pins</b>           | 6                                  | <b>Length</b>                  | 68.6 mm   |
| <b>DC Current per I/O Pin</b>      | 20 mA                              | <b>Width</b>                   | 53.4 mm   |



# Peripherals





# Code

```
#include <avr/io.h>
|
int main(void)
{
    DDRB |= (1 << PB3);          /* Set PORTB bit 4 to output. */
    // Set the ADC input to PB2/ADC1
    ADMUX |= (1 << MUX0);
    // Set the prescaler to clock/128 & enable ADC
    ADCSRA |= (1 << ADPS1) | (1 << ADPS0) | (1 << ADEN);

    while(1)
    {
        // Start the conversion
        ADCSRA |= (1 << ADSC);
        // Wait for it to finish
        while (ADCSRA & (1 << ADSC));

        if((ADCH << 8) | ADCL > 0x0200)
        {
            PORTB |= (1 << PB3);
        }
        else
        {
            PORTB &= ~(1 << PB3);
        }
    }
}

void setup()
{
    pinMode(13, OUTPUT);
}

void loop()
{
    if(analogRead(A0) > 300)
    {
        digitalWrite(13, HIGH);
    }
    else
    {
        digitalWrite(13, LOW);
    }
}
```



The screenshot shows the Arduino IDE interface. The title bar reads "sketch\_mar09a | Arduino 1....". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar contains icons for upload, download, and other functions. The main code editor window displays the following sketch:

```
void setup() {
  // put your setup code here, to run once:
}

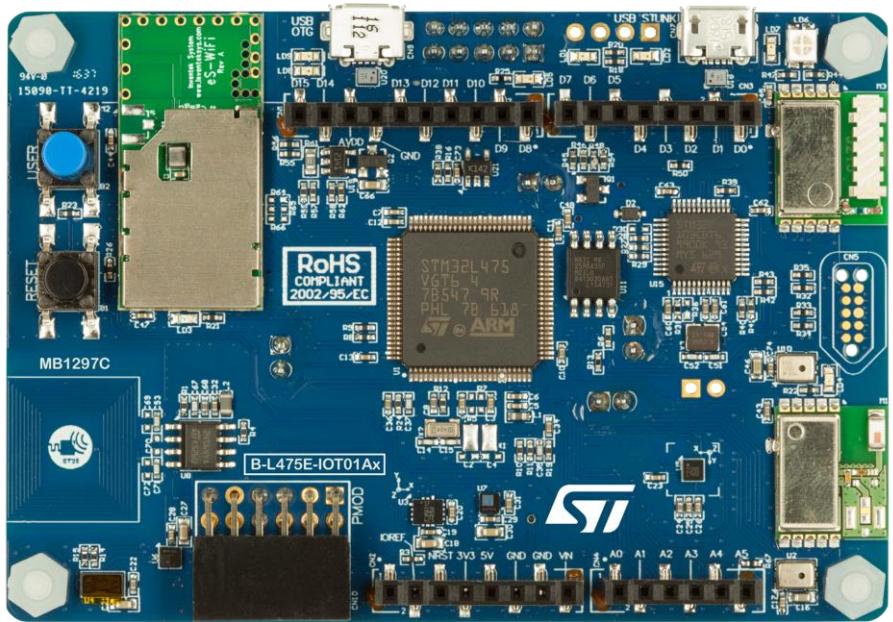
void loop() {
  // put your main code here, to run repeatedly:
}
```

The status bar at the bottom indicates "Arduino Due (Programming Port) on COM1".

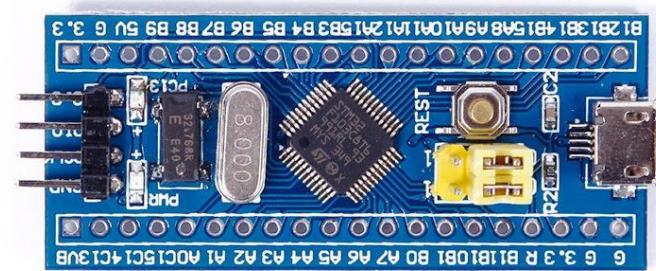
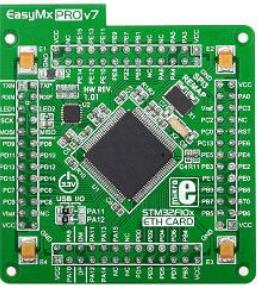
The screenshot shows the Atmel Studio 7 interface. The title bar reads "Start Page - AtmelStudio". The menu bar includes File, Edit, View, VAssistX, ASF, Project, Debug, Tools, Window, and Help. The toolbar has various development tools. The main area displays the "Discover Atmel Studio" page, which includes links for "New Project...", "New Example Project...", "Open Project...", "Recent" projects (test, test1), and "Getting started with Atmel Studio", "Getting started with AVR development", "Open Atmel Start Configurator", "Download Atmel Studio Extensions", and "Download documentation". The "Error List" panel shows "Entire Solution" with 0 Errors, 0 Warnings, and 0 Messages. The "Output" panel shows "Ready". A sidebar on the right titled "VA View" lists "[Files in Solution]" and "[Symbols in Solution]".



# STM32XXXX (ARM Cortex)

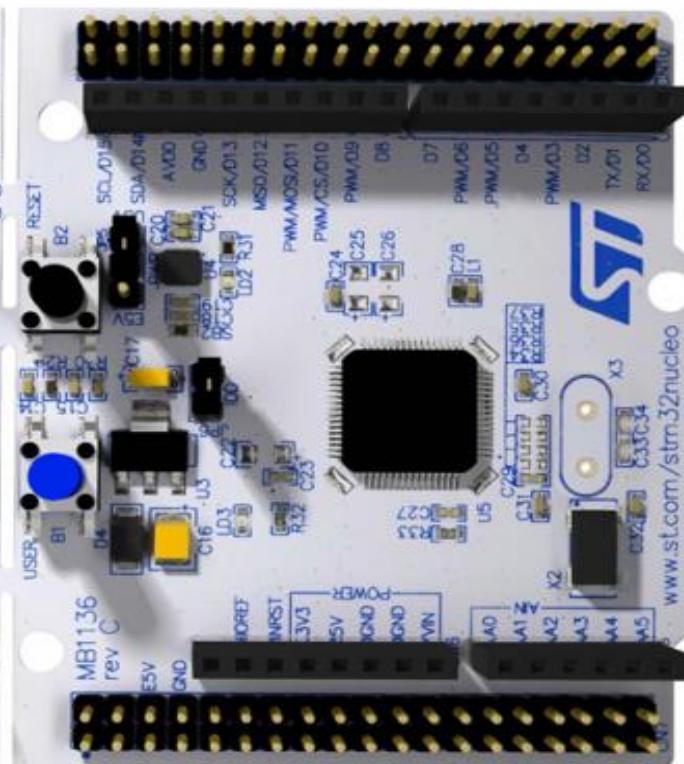


ARM  
mbed  
enabled



TouchGFX

ST  
life.augmented





# STM32F103 Features (ARM Cortex M3)

| System  | 72 MHz ARM Cortex-M3 CPU                                   | 128-Kbyte Flash memory<br>20-Kbyte SRAM<br>FSMC/SRAM/NOR/NAND CF/LCD parallel interface<br>84-byte backup data      | Analog  |
|---|--|---|---|
| Power supply<br>1.8 V internal regulator<br>POR/PDR/PVD/BOR | Nested Vector Interrupt Controller (NVIC)<br>JTAG/SW debug |   | Up to 16x12-bit ADC<br>Temperature sensor                           |
| Xtal oscillators<br>32 kHz + 3 ~25 MHz                      | 72 MHz ARM Cortex-M3 CPU                                   |   |   |
| Internal RC oscillators<br>40 kHz + 8 MHz                   |  |   |   |
| Internal multispeed ULP RC oscillator<br>64 kHz to 4 MHz    |  |   |   |
| PLL   |  |   |   |
| Clock control   |  |   |   |
| RTC/AWU   |  |   |   |
| SysTick timer   |  |   |   |
| 2x watchdogs (independent and window)                       |  |   |   |
| 26/36/51/80 I/Os  | AHB bus matrix<br>12-channel DMA                           |   |   |
| Cyclic Redundancy Check (CRC)                               |  |   |   |
|   |  | Connectivity  | Control   |
|   |  | 1x CAN<br>Up to 3x USART LIN, smartcard, IrDA, modem control<br>Up to 2x SPI<br>Up to 2x I <sup>2</sup> C<br>1x USB | 3x 16-bit timer<br>1x16-bit motor control PWM Synchronized AC timer |



# Code: C++

# ARM® mbed™

## mbed OS Overview



Minimize time-to-market



Low-power by design



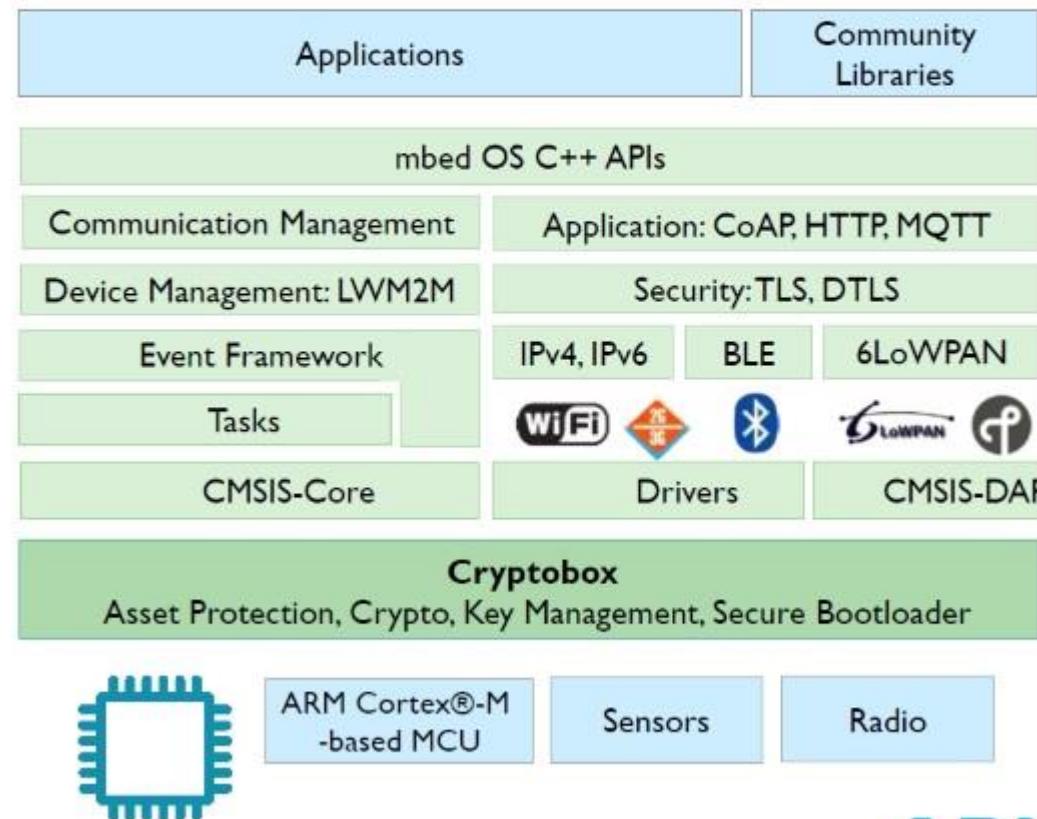
Complete security solution



Top connectivity standards



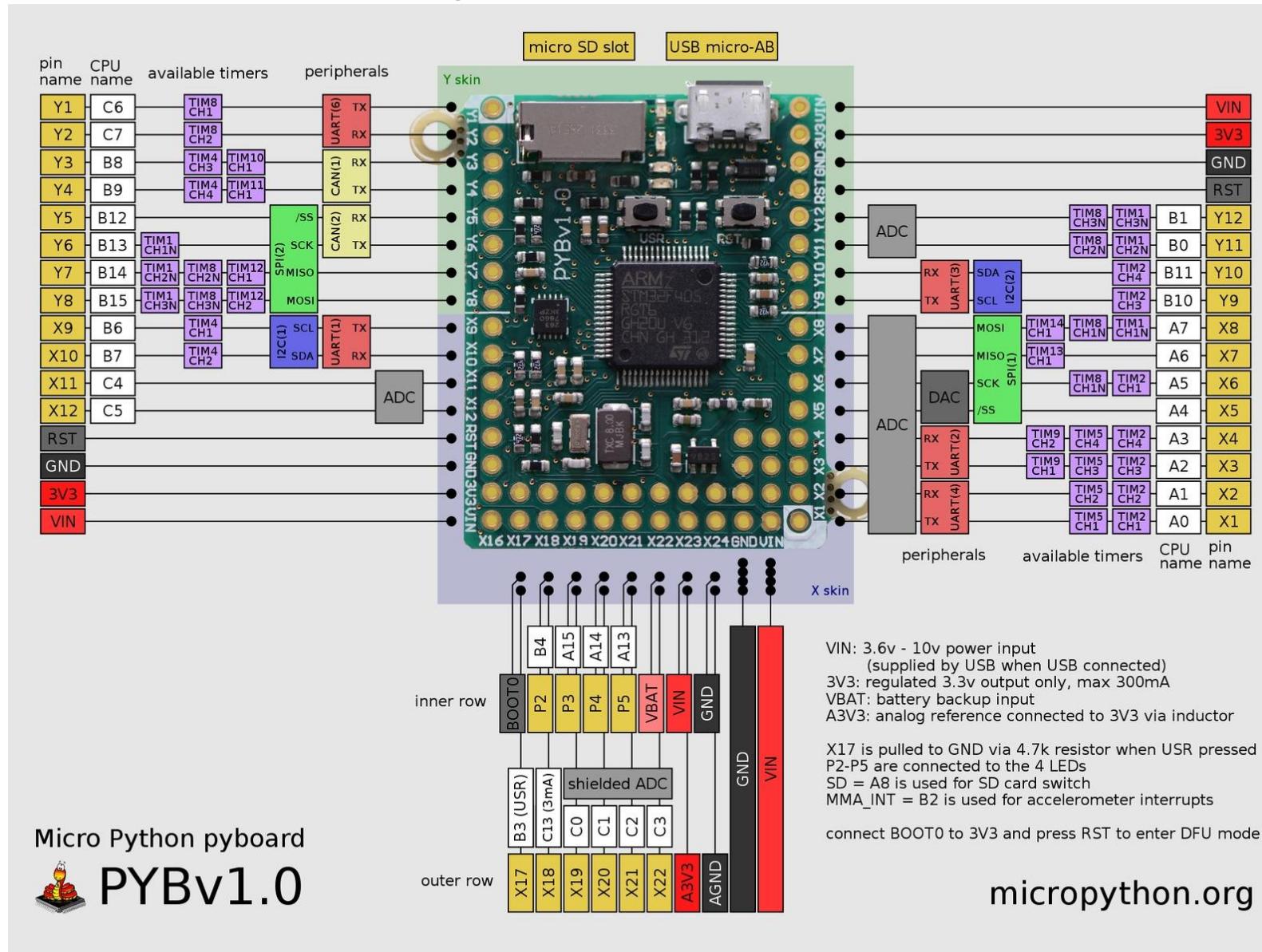
Built-in device management



ARM®



# Code: Python



```
from machine import Pin, I2C

# create an I2C bus
i2c = I2C(scl=Pin('X1'), sda=Pin('X2'))

# scan for list of attached devices
dev_list = i2c.scan()

# write to and read from a device
i2c.writeto(0x42, b'\x04')
data = i2c.readfrom(0x42, 4)

# memory transactions
i2c.writeto_mem(0x42, 0x12, b'\x04')
data = i2c.readfrom_mem(0x42, 0x12, 2)
```

```
import pyb
```

```
# turn on an LED
pyb.LED(1).on()
```

```
# print some text to the serial console
print('Hello MicroPython!')
```



# IDE: Mbed Online

ARM® mbed™

The screenshot shows the Mbed Online IDE interface. The title bar indicates the window is titled "mbed Compiler /H" and the URL is "https://developer.mbed.org/compiler/#nav:HADdemo1/main.cpp". The menu bar includes "File", "Edit", "View", "Tools", "Help", and "About". The toolbar contains icons for "New", "Import", "Save", "Save All", "Compile", "Commit", "Revisions", and "Help". A status bar at the bottom shows "Ready.", "In 18", "col 20", "37", and "INS".

**Program Workspace:** On the left, a tree view shows "My Programs" with various projects listed, including "HADdemo1" which is currently selected. Inside "HADdemo1", there are sub-folders "tsi\_sensor" and "mbed", and files "main.cpp" and "mbed".

**Code Editor:** The main workspace displays the content of "main.cpp". The code initializes three LEDs (green, red, blue) and reads a value from a TSI analog slider. It then uses a while loop to continuously update the blue LED based on the slider's percentage value.

```
15 int main(void) {
16     PwmOut led_g(LED_GREEN);
17     PwmOut led_r(LED_RED);
18     PwmOut led_b(LED_BLUE);
19     TSIAnalogSlider tsi(ELECO, ELEC1, 40);

20     while (true) {
21         float f=tsi.readPercentage();
22         if (f<.05)
23         {
24             led_b=0.0;
25             led_g=led_r=1.0;
26         }
27     }
}
```

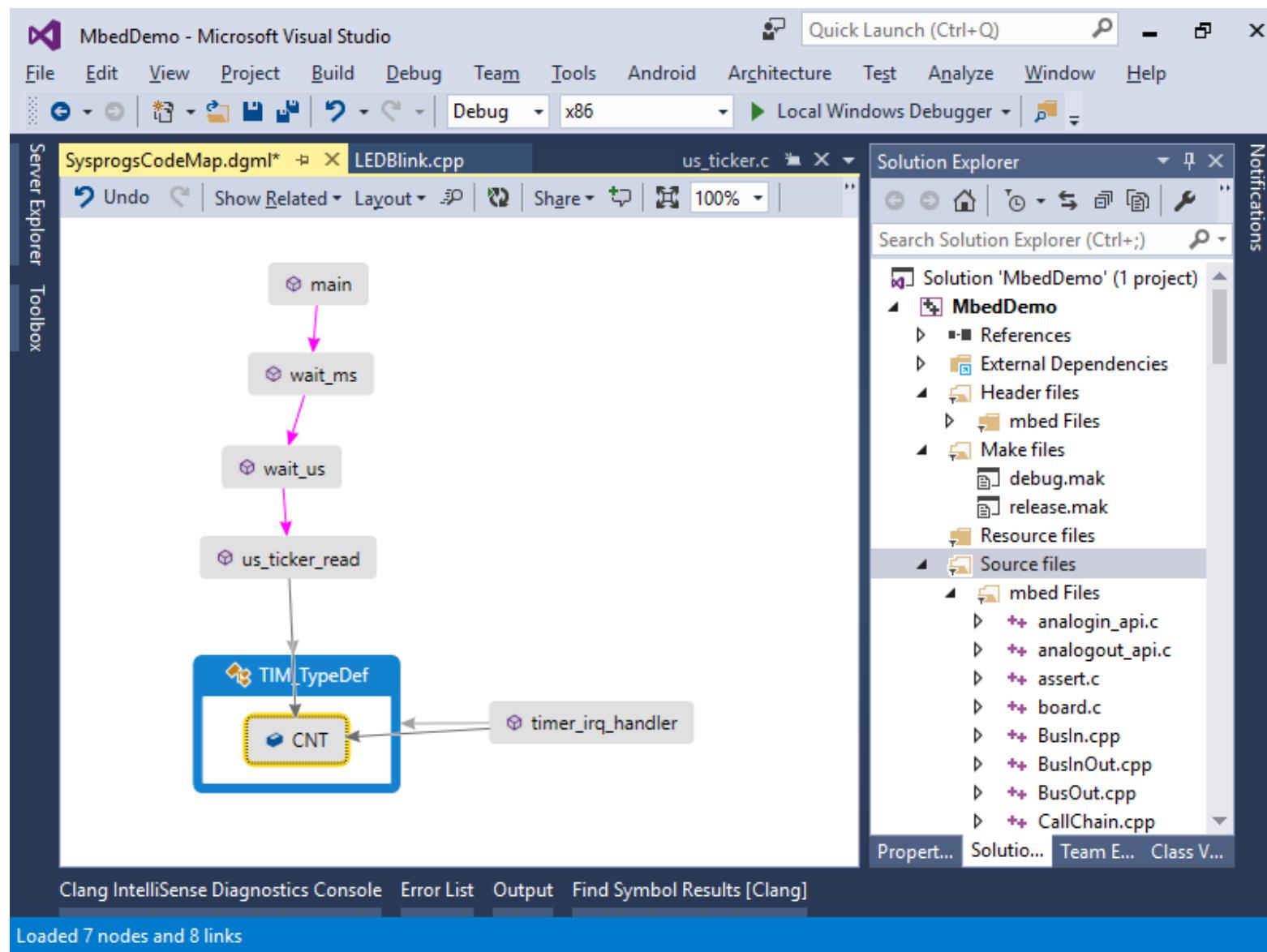
**Compiler Output:** Below the code editor, a table titled "Compile output for program: HADdemo1" shows the build results. The table has columns for "Description", "Error Number", "Resource", "In Folder", and "Location". The "Compile Output" tab is selected, showing the message "Compiling main.cpp". The status bar at the bottom right shows "85" notifications.

| Description        | Error Number | Resource | In Folder | Location |
|--------------------|--------------|----------|-----------|----------|
| Compiling main.cpp | 0            |          |           |          |

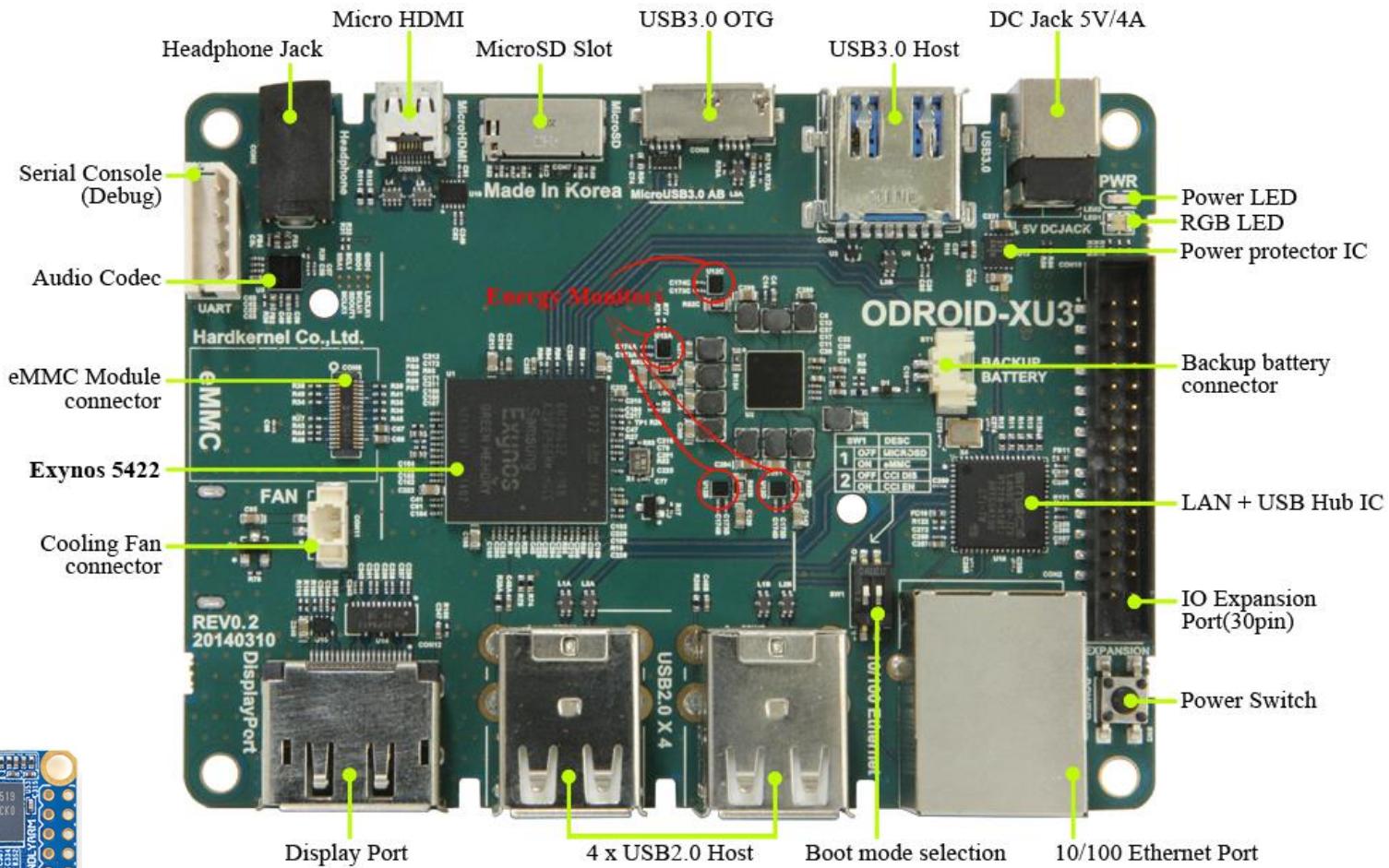
**Right Panel:** A floating panel on the right side shows two definitions for the enumerator "LED\_BLUE". The first definition is "Enum: \_\_anon612 LED\_BLUE = PTD1," and the second is "Enum: \_\_anon1172 LED\_BLUE = PC\_5," both associated with the variable "led\_b".



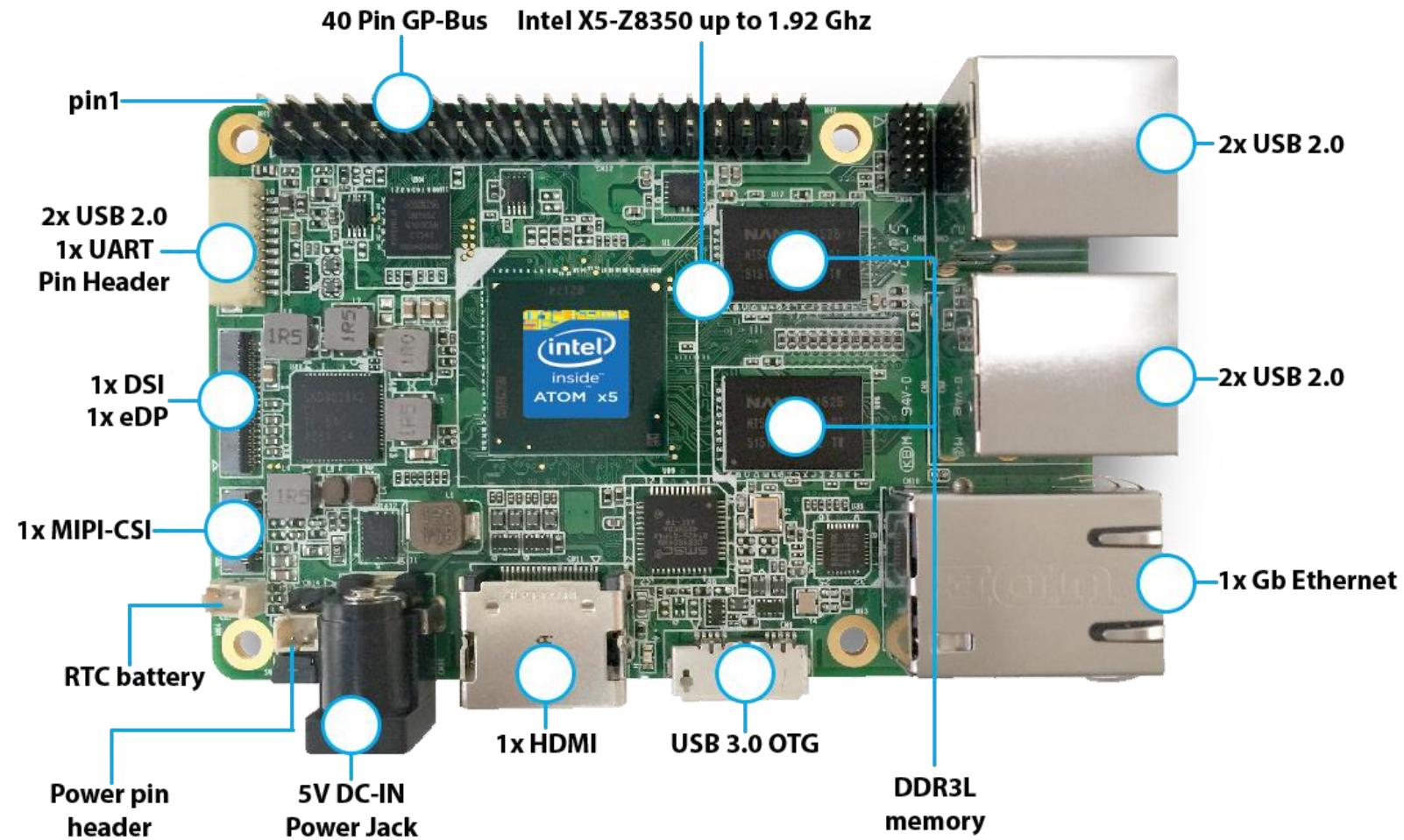
# IDE: Visual Studio



# ARM Compute Modules



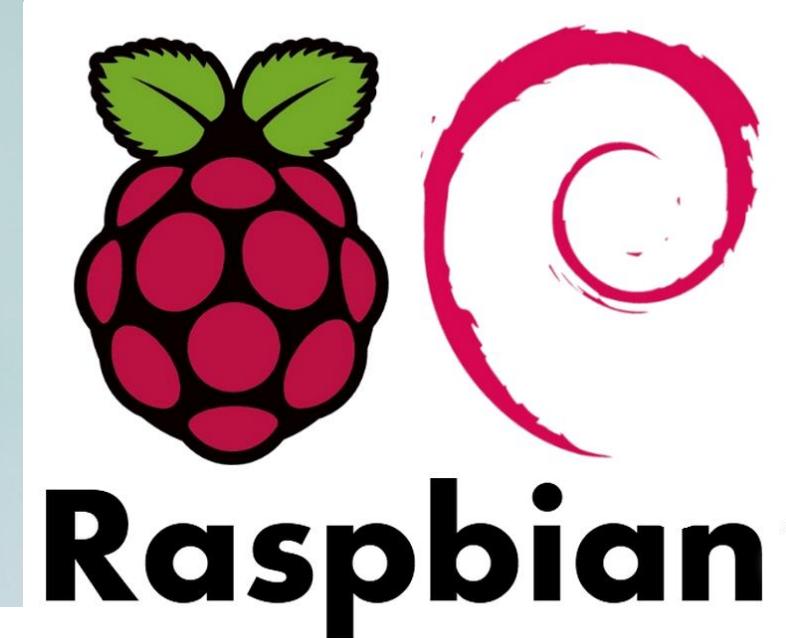
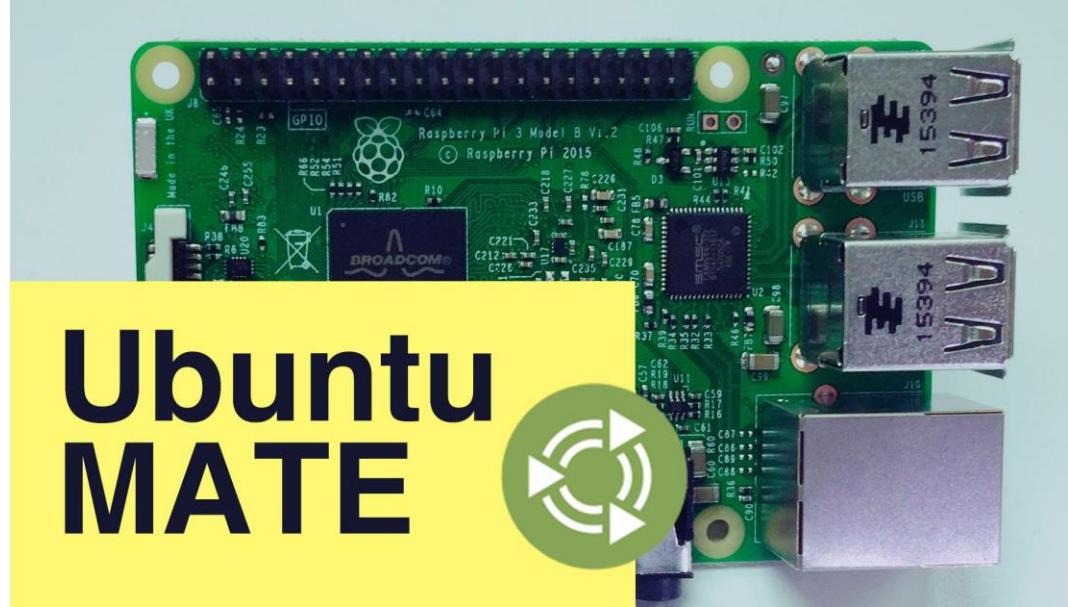
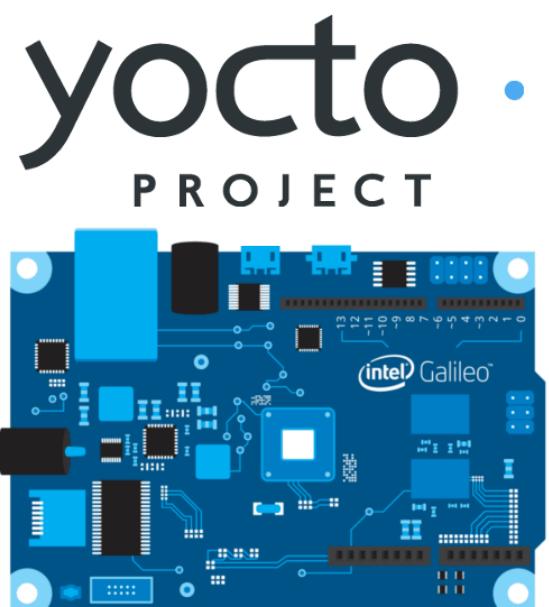
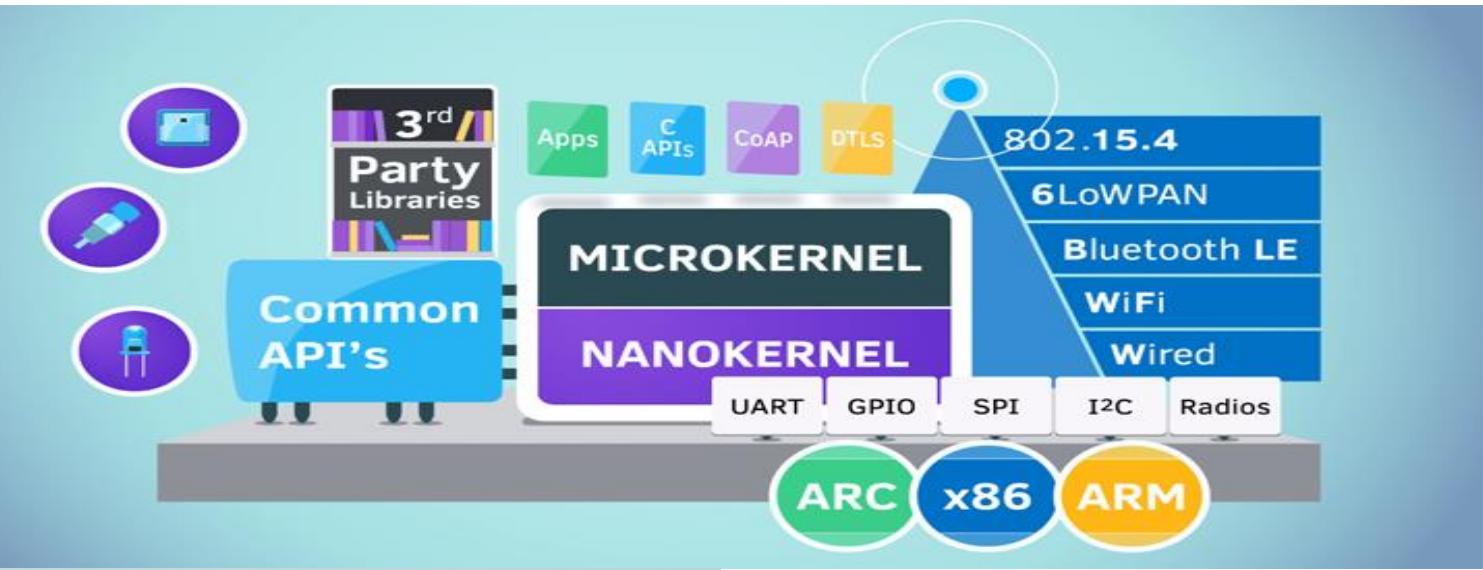
# X86 Compute Modules



# IoT Gateways



# OS: Linux



# OS: Windows 10

## Windows 10 IoT Enterprise

Desktop Shell, Universal Windows Apps, Classic Windows Applications

1 GB RAM, 16 GB Storage

X86



## Windows 10 IoT Mobile Enterprise

Modern Shell, Universal Windows Apps

512 MB RAM, 4 GB storage

ARM



## Windows 10 IoT Core

No Shell, Universal Windows Apps

256MB RAM, 2GB storage

X86 or ARM





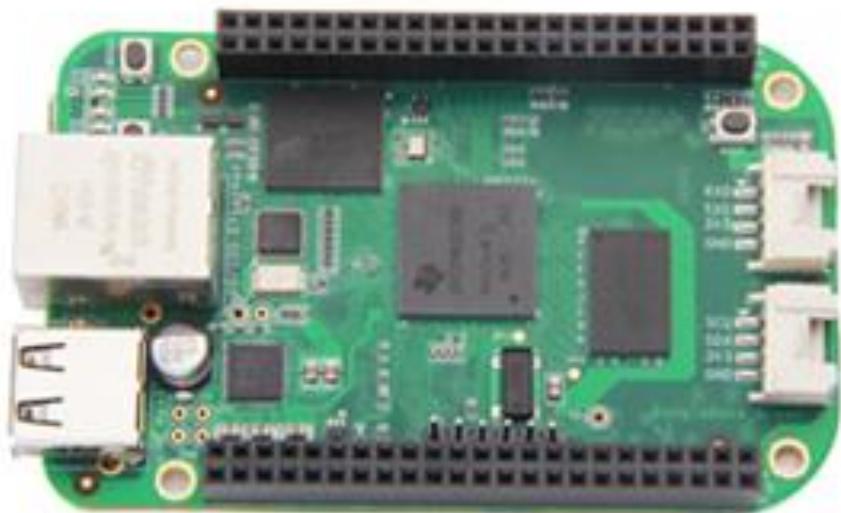
# Azure certified kits



# Azure Certified



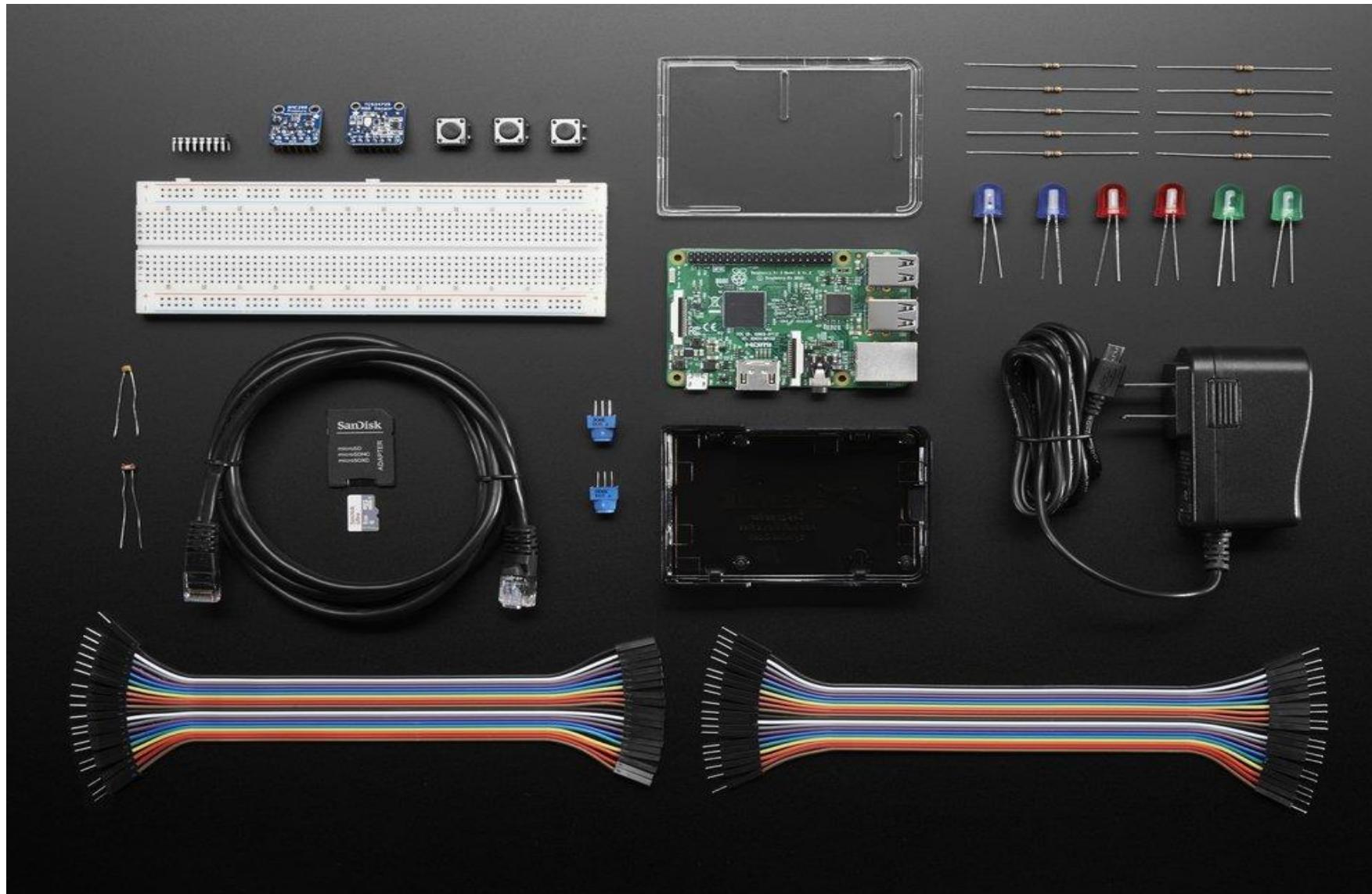
**BeagleBone  
Green** Manufactured  
by Seeedstudio



# Azure Certified



# Azure Certified

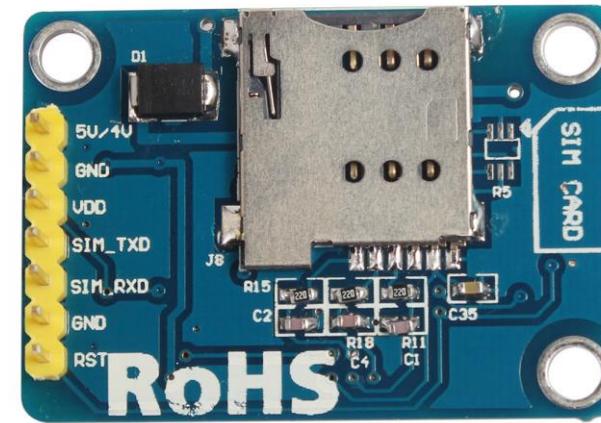




# Connectivity

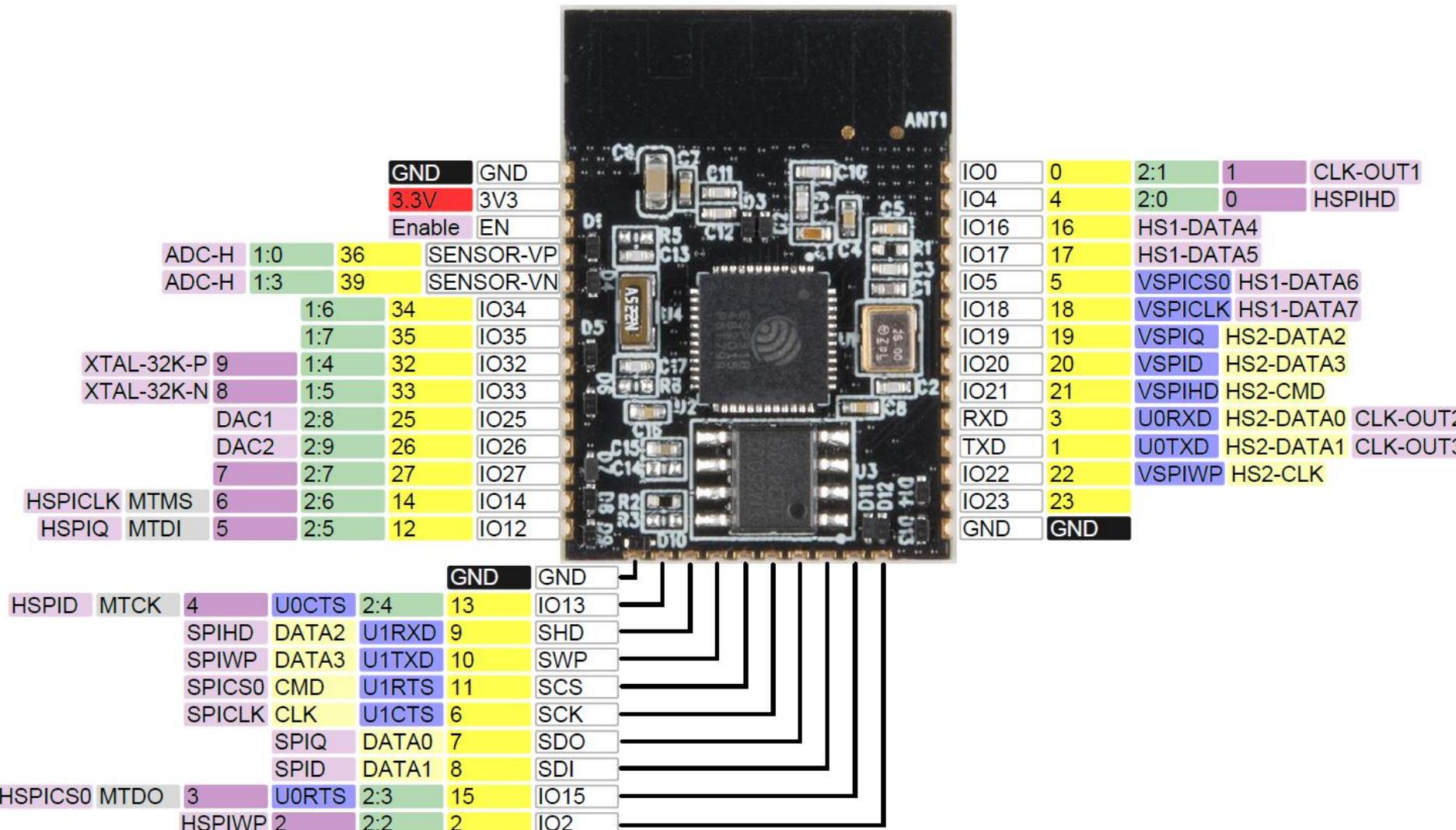


# Connectivity: GPRS\3G



# WiFi\Bt + Compute: ESP32

Name Power GND GPIO ADC Serial Touch SDIO JTAG Misc

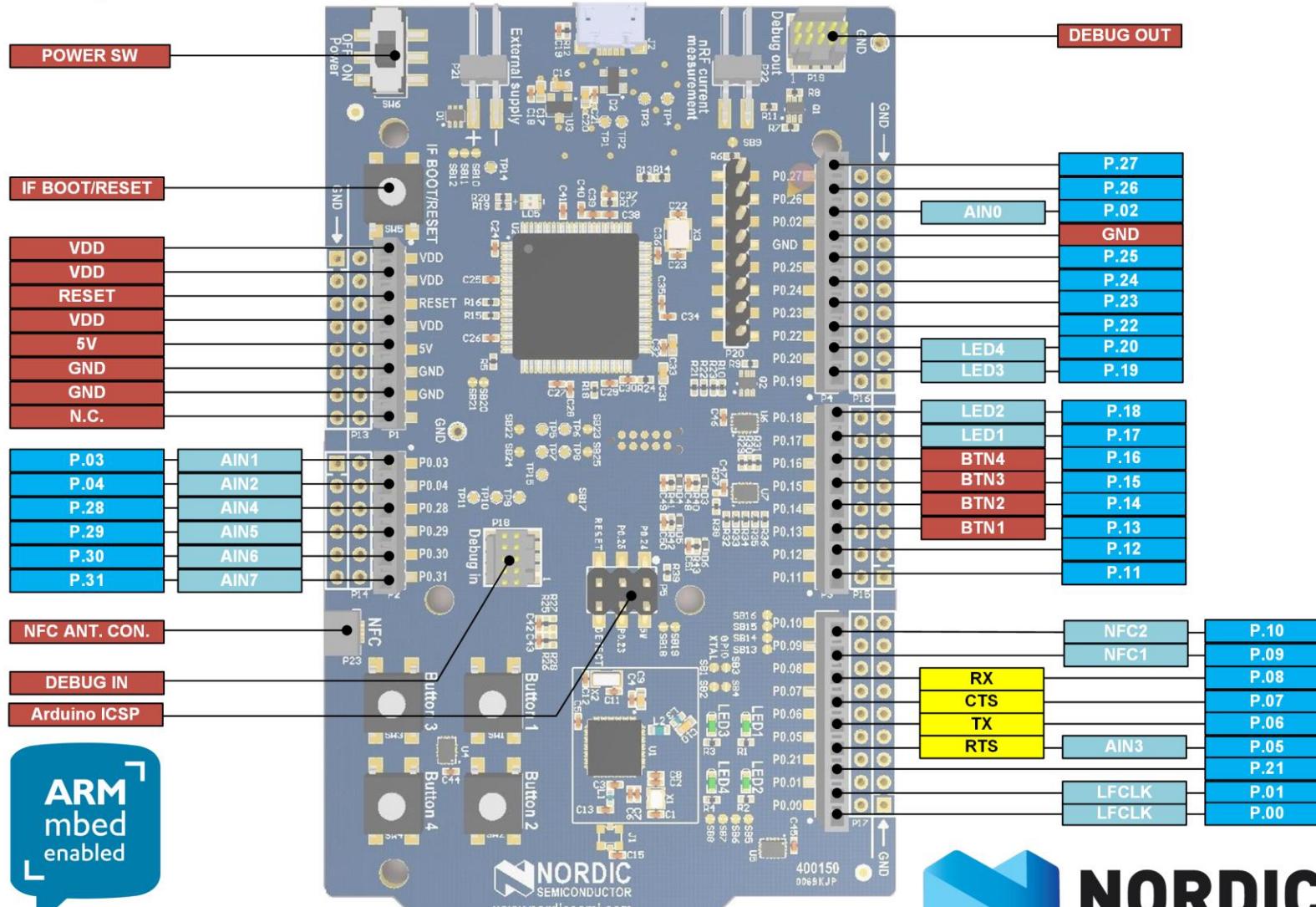


Motor PWM: Any GPIO - Four channels of 16-bit timers to generate PWM waveforms.

LED PWM: Any GPIO - 8 independent channels running at 80MHz clock

# Bluetooth (LE) 5 + Compute: nRF52840

## nRF52-DK



**NORDIC**  
SEMICONDUCTOR

Smarter Things



VS



# Lora vs SigFox

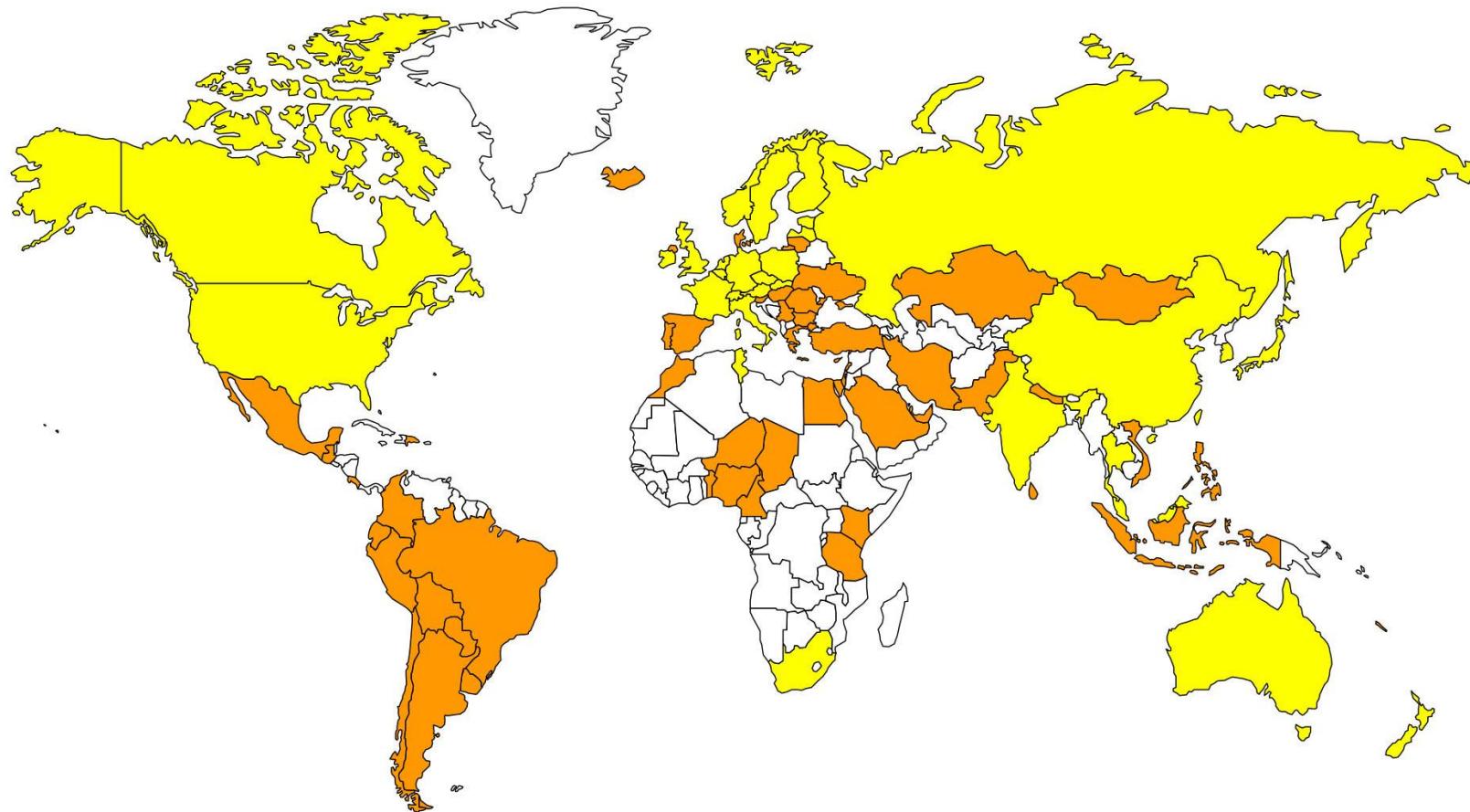
## SigFox

The SigFox business model takes a top-down approach. The company owns all of its technology—from the backend data and cloud server to the endpoints software. But the differentiator is that SigFox is essentially an open market for the endpoints. SigFox gives away its endpoint technology to whatever silicon manufacturer or vendor wants it so long as certain business terms are agreed upon. Large manufacturers like [STMicroelectronics](#), [Atmel](#), and [Texas Instruments](#) make SigFox radios. SigFox thinks that allowing the application to be really inexpensive is the way to drive people to its market.

## LoRa

The LoRa Alliance has a different strategy. They would say they're more open than SigFox, strictly because the specification that governs how the network is managed is relatively open. You can download the specifications and join the LoRa Alliance, and any hardware or gateway manufacturer can build a module or gateway that conforms with the LoRa specifications. The catch is that the only company that makes the radio for LoRa is Semtech. (They've announced licensing to other silicon manufacturers in the future, but Semtech is the only option right now.) So while the ecosystem itself is open, it does have a closed element.

## Countries — LoRaWAN™ Networks



- 42 Publicly Announced Operators
- 30 Alliance Member Operators
- 250+ on-going trials & city deployments
- 480+ members in the Alliance

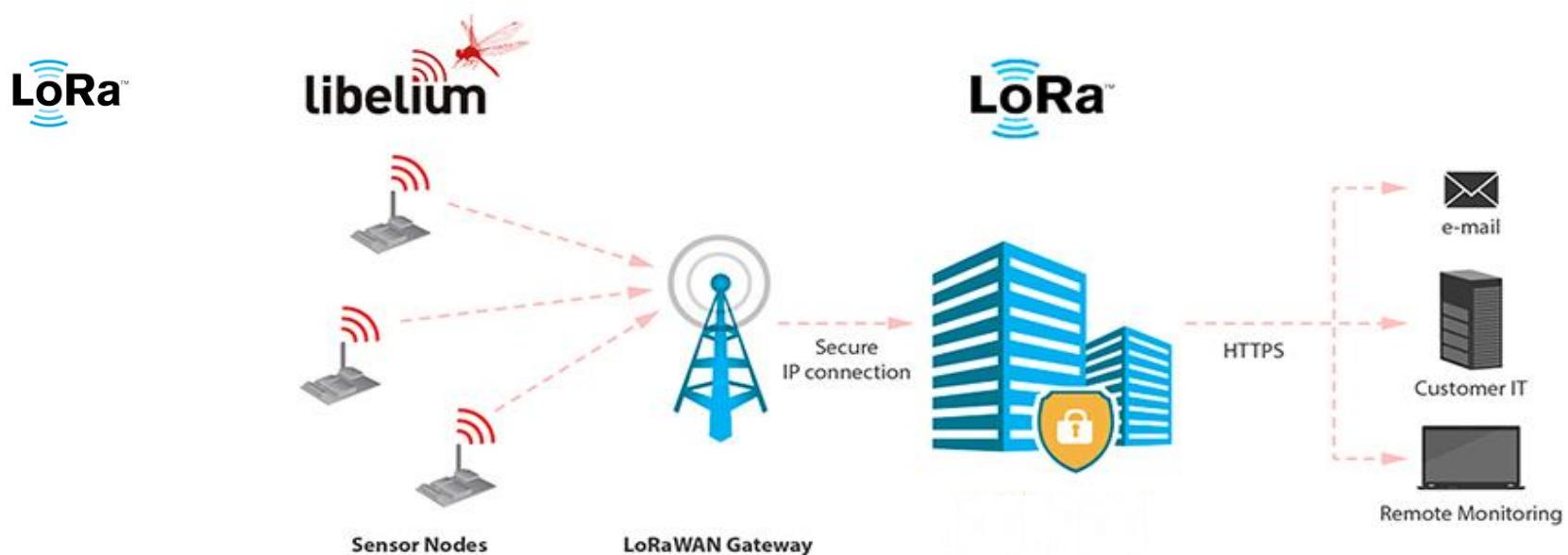
Legend:

- (Yellow circle) Publicly Announced
- (Orange circle) Other Deployments

# SigFox



# LoRaWan



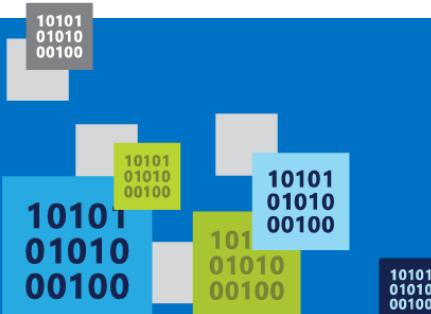
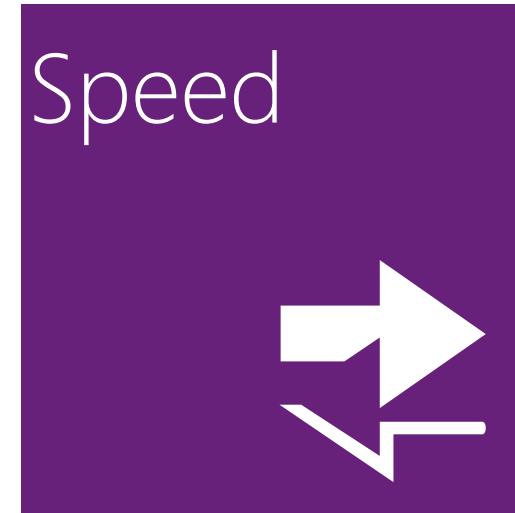


# IoT Cloud



- Rapidly setup environments to drive business priorities
- Scale to meet peak demands
- Increase daily activities, efficiency and reduced cost.

# Why the cloud?





DIFFERENTIATION  
AGILITY  
COST



32 DC Generally Available, 4 DC coming soon

>80%  
of Fortune 500 use  
the Microsoft Cloud



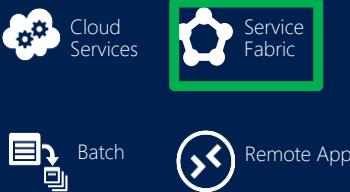
# Azure

## Azure Platform Services

### Security & Management



### Compute



### App Services



### Developer Services



### Hybrid Operations



### Integration



### Analytics & IoT



### Data



### Media & CDN



## Azure Infrastructure Services

### Virtual Machines



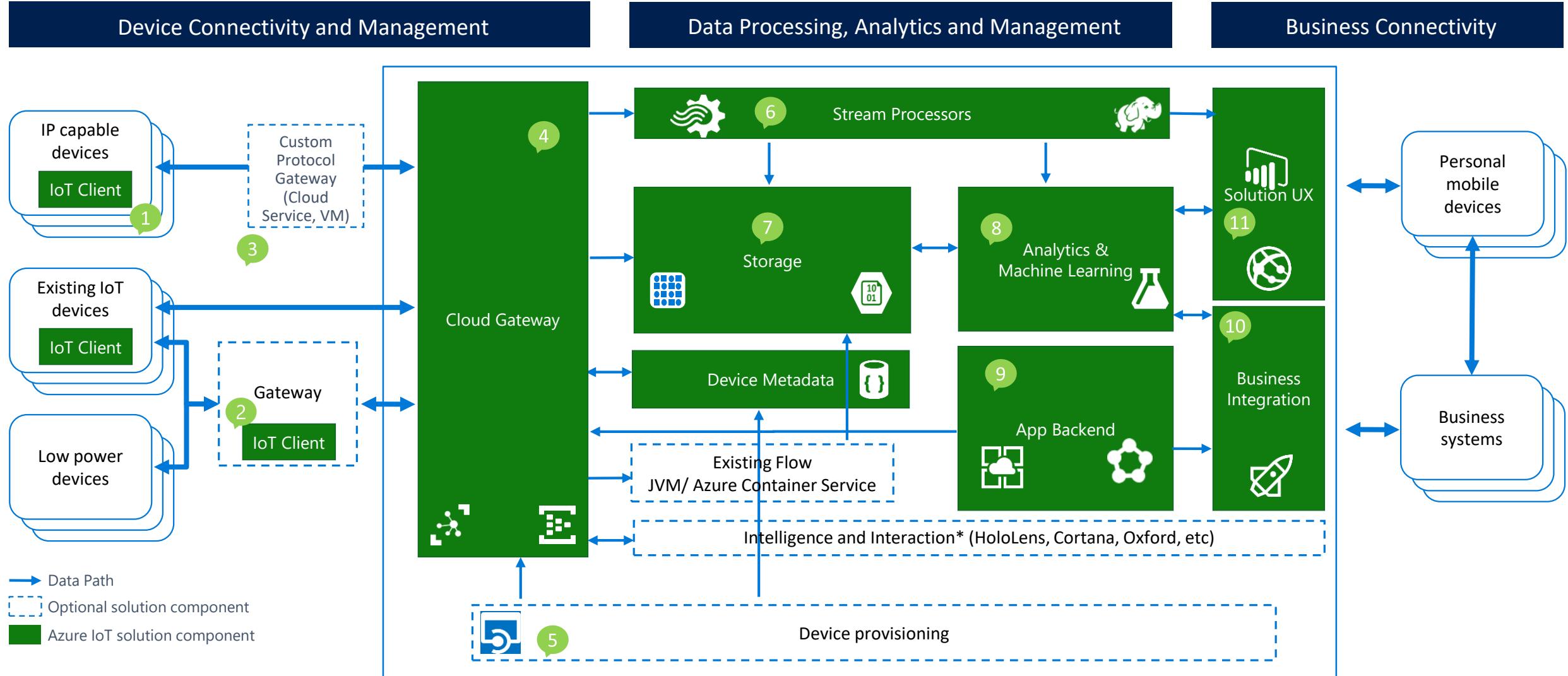
### Storage



### Networking



# Azure IoT reference architecture



# Many aspects of connectivity

## Functionality

Device-to-cloud telemetry,  
Cloud-to-device commands and notifications,  
File uploads/downloads

## Security

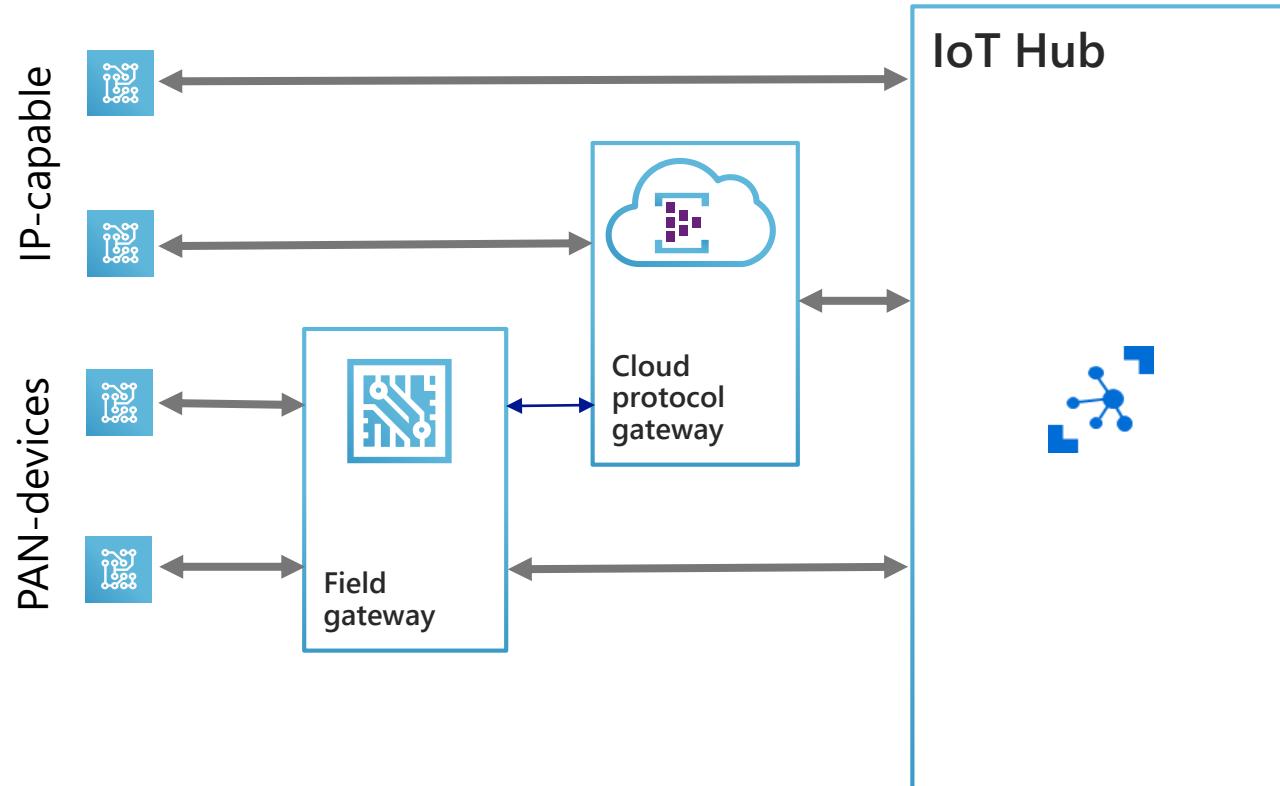
Device security,  
Cloud security,  
Channel security, ...

## Monitoring

Identify malfunctioning devices when they  
cannot be reached directly

## Reach and customization

RTOS/Linux/Windows/non-IP capable,  
Network/application protocols,  
Authentication schemes



# Device identity in IoT Hub

## Identity registry

Contains per-device security materials

Individual device blocking

No queries → keep a device registry to query by app-data

## Authenticated comms

Label incoming telemetry

Ensure identity of devices receiving messages

## Custom

AAD integration

“Master” key for gateway scenarios

## Device provisioning orchestration

IoT Hub  
(identity registry)

Device registry  
(e.g. DocDB, SQL)

Other  
(e.g. ERP)

# Device-to-cloud messages

## Interface

AMQP and HTTPS device-side endpoint

AMQP service-side endpoint

Device and service SDKs

## Compatible with Event Hubs

Partitioned receiver, client check-pointing

Integrations with Azure Stream Analytics, Storm, ...

100% compatible with Event Hubs receivers

## IoT Hub services for D2C

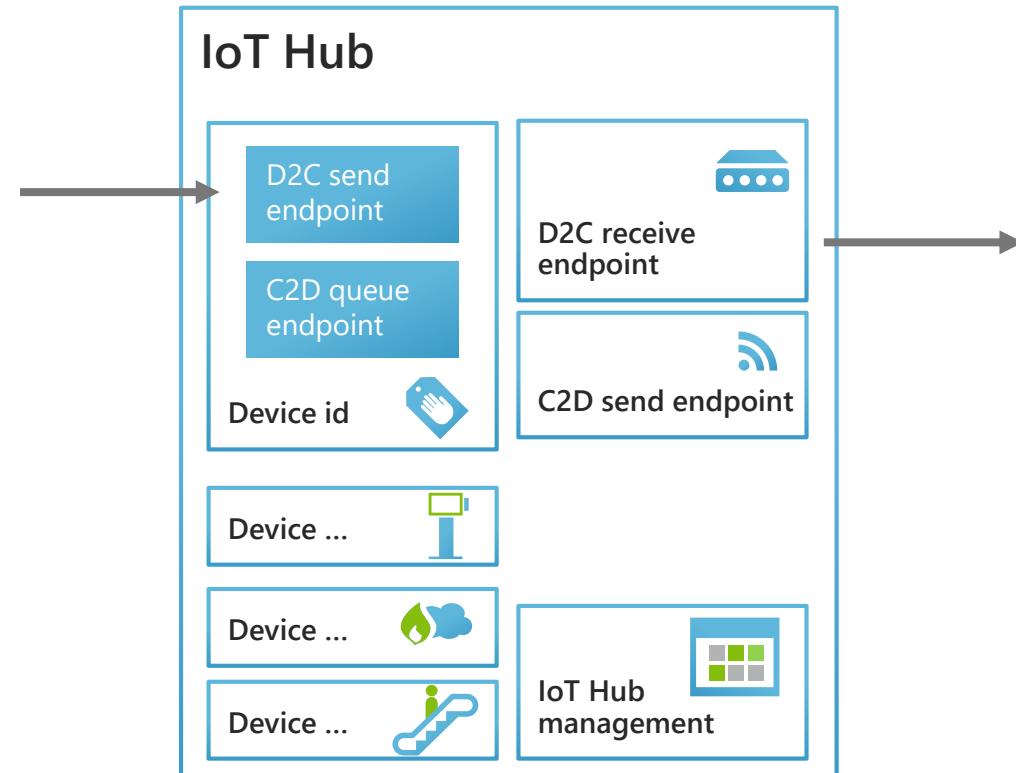
Millions of simultaneously connected devices

Per-device authentication

Connection-multiplexing:

C2D and D2C traffic

Across multiple devices for gateway scenarios



# Cloud-to-device messages

## Interface

AMQP and HTTPS device-side endpoint  
AMQP service-side endpoint

## At-least-once semantics

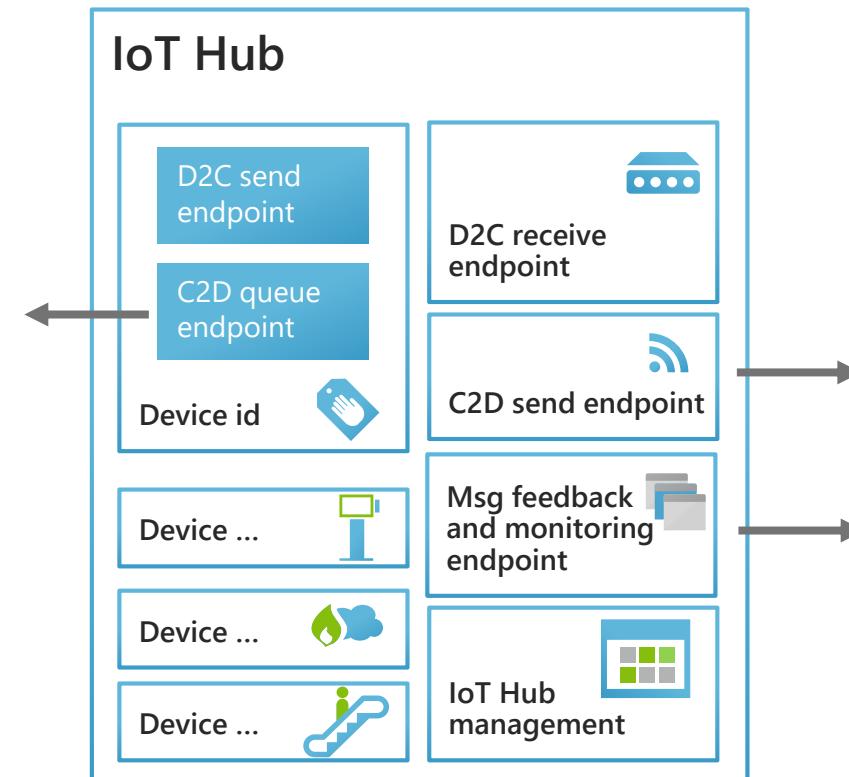
Durable messages  
Device acknowledges receipt  
(Send - Receive - Abandon OR Complete)

## TTL and receipts

Per-message TTL  
Per-message positive and negative receipts

## Command lifecycle pattern

Use correlated D2C for responses  
Use feedback information to retry  
Store command state in command registry



# Pattern: file uploads

App creates link to new blob

Appends temporary token

Device uploads file

Directly to storage

Using block blobs:

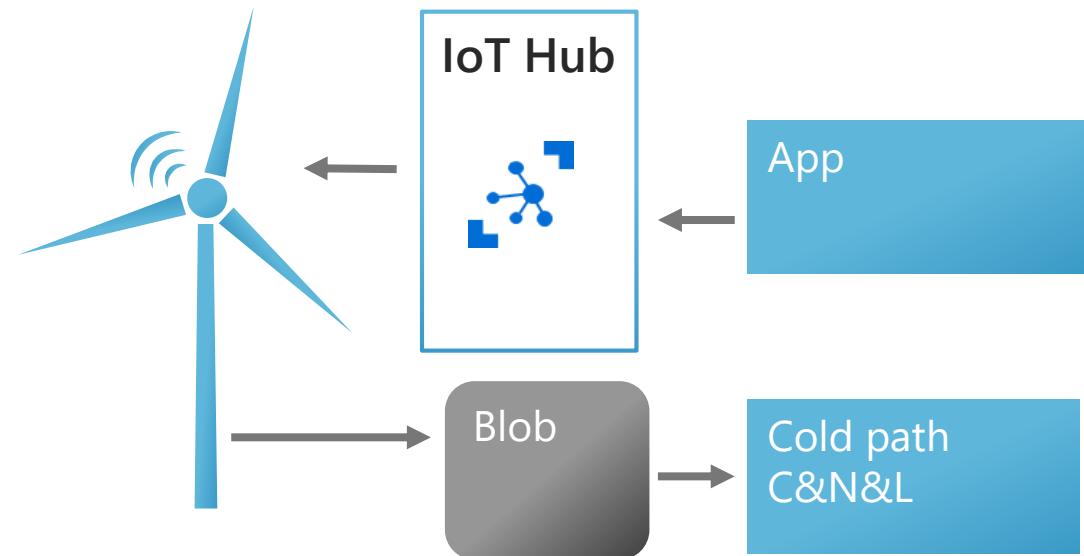
Use small block sizes

Minimizes failure rates on slow connections

Enable resumes

(Optional) send D2C “alert”

Notification of completion



# Pattern: publish file

App creates link to blob

Appends temporary token

Device download file

Directly from storage

Using block blobs:

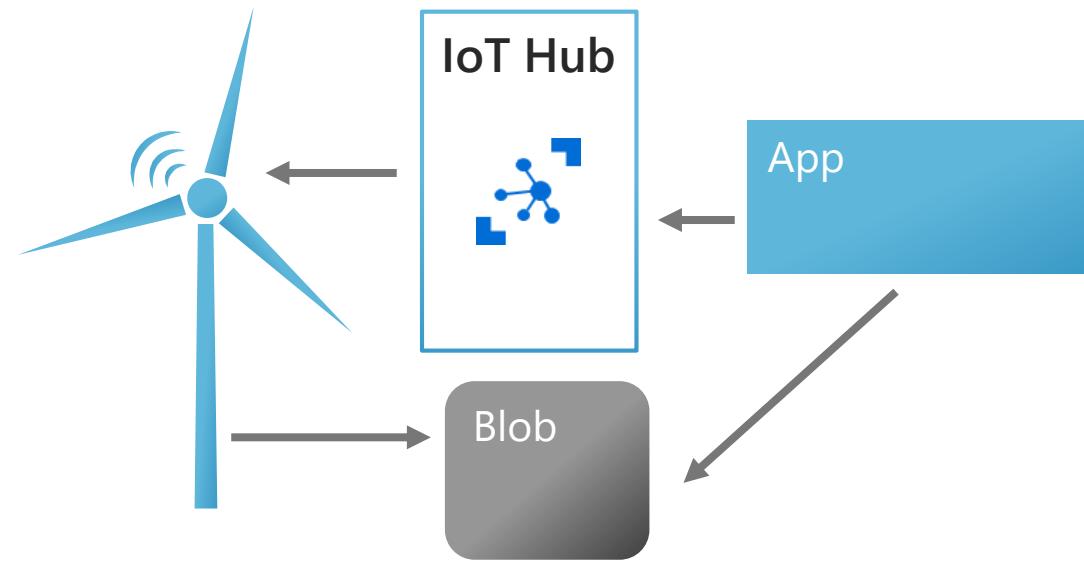
Use small block sizes

Minimizes failure rates on slow connections

Enable resumes

(optional) verify integrity

Very important for software/config updates.



# Cloud and field gateways

## Use cases

PAN-IP access

Protocol translation

## IoT Hub capabilities

Connection-multiplexing for multiple devices

Individual device identities through gateway

Extensible MQTT cloud gateway

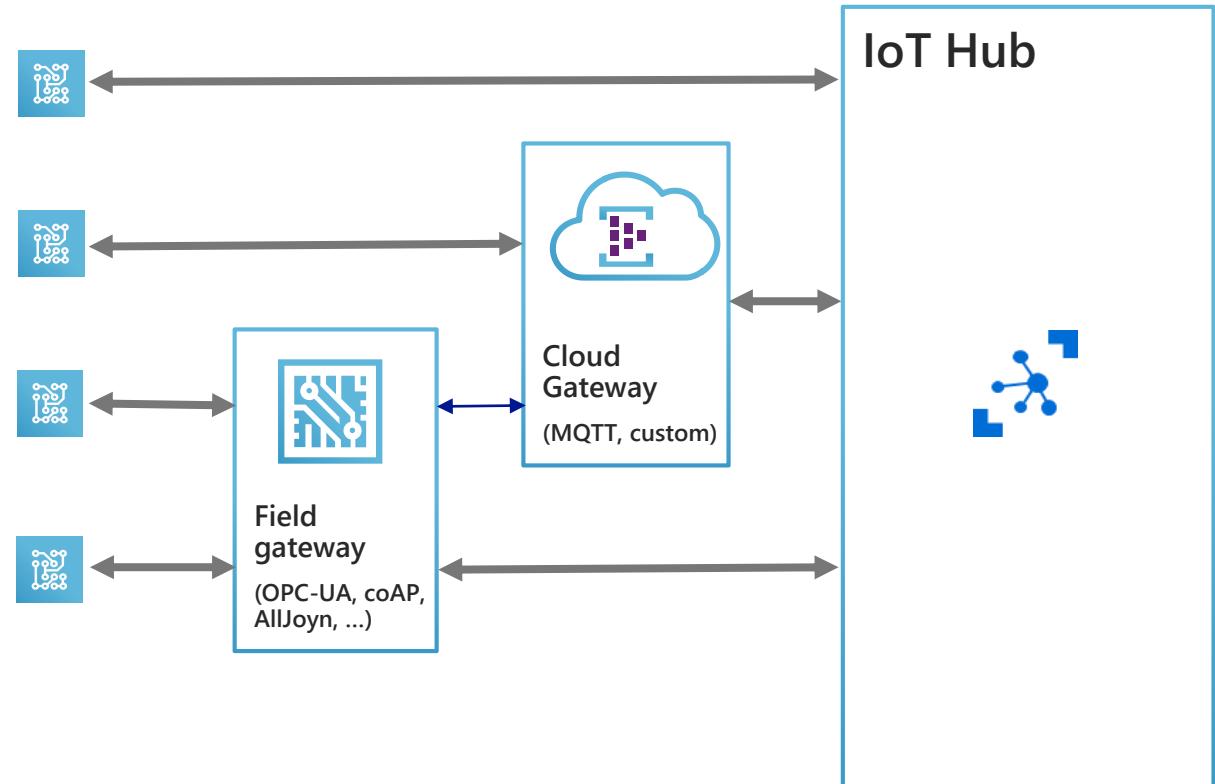
## Patterns

### Transparent vs opaque

Individual identities known to hub or not

### Pull vs push

Individual devices acts as servers or maintain persistent connection



# Pattern: token service

## Custom authentication

Do not put a gateway in front of IoT Hub!

## Device requests token

Token service perform custom auth

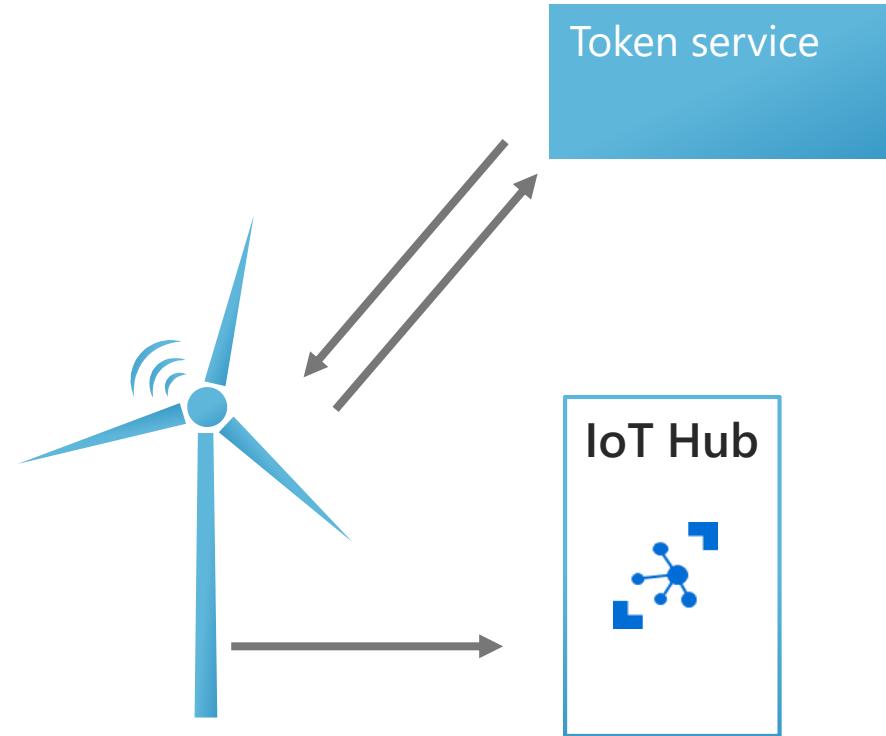
E.g. using custom IDP

Returns IoTHub-specific token

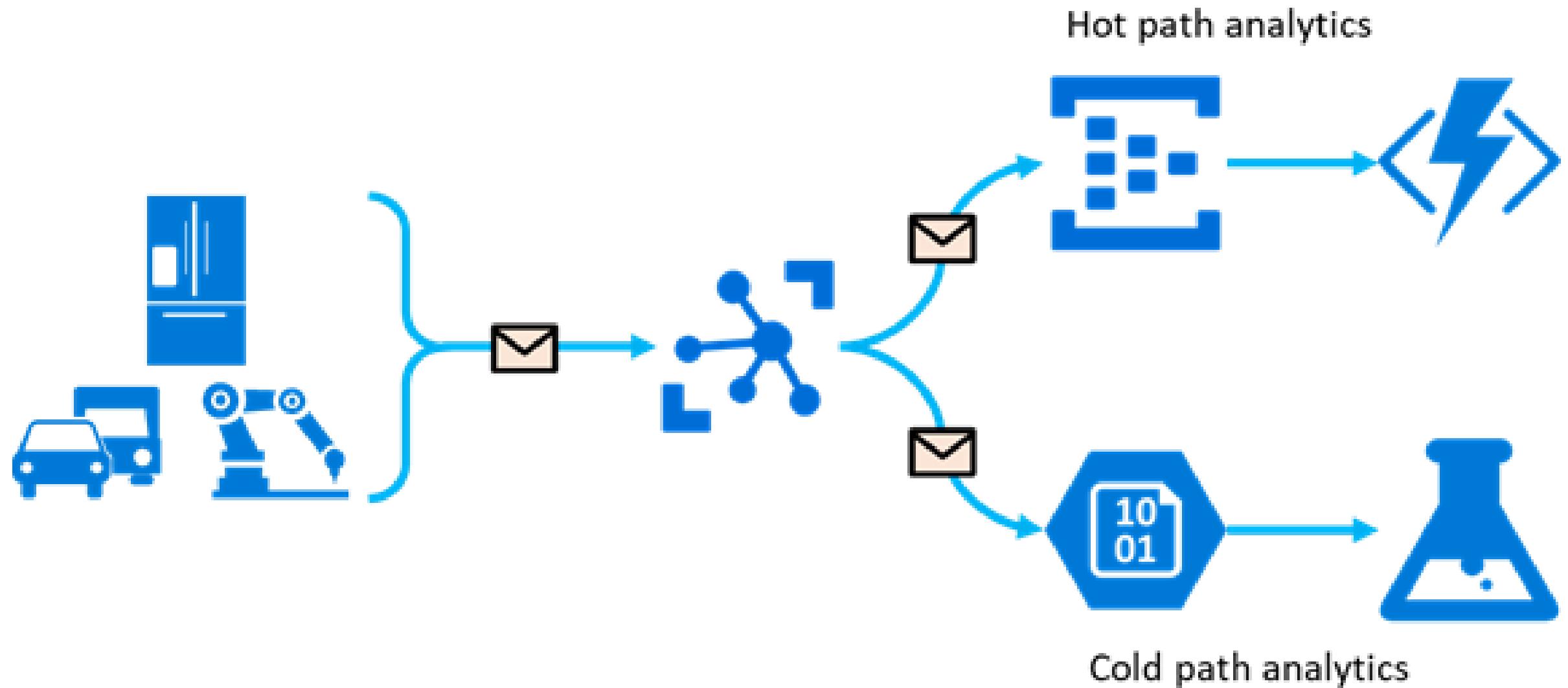
Using IoTHub key with "DeviceConnect" rights

## Device uses token with IoTHub

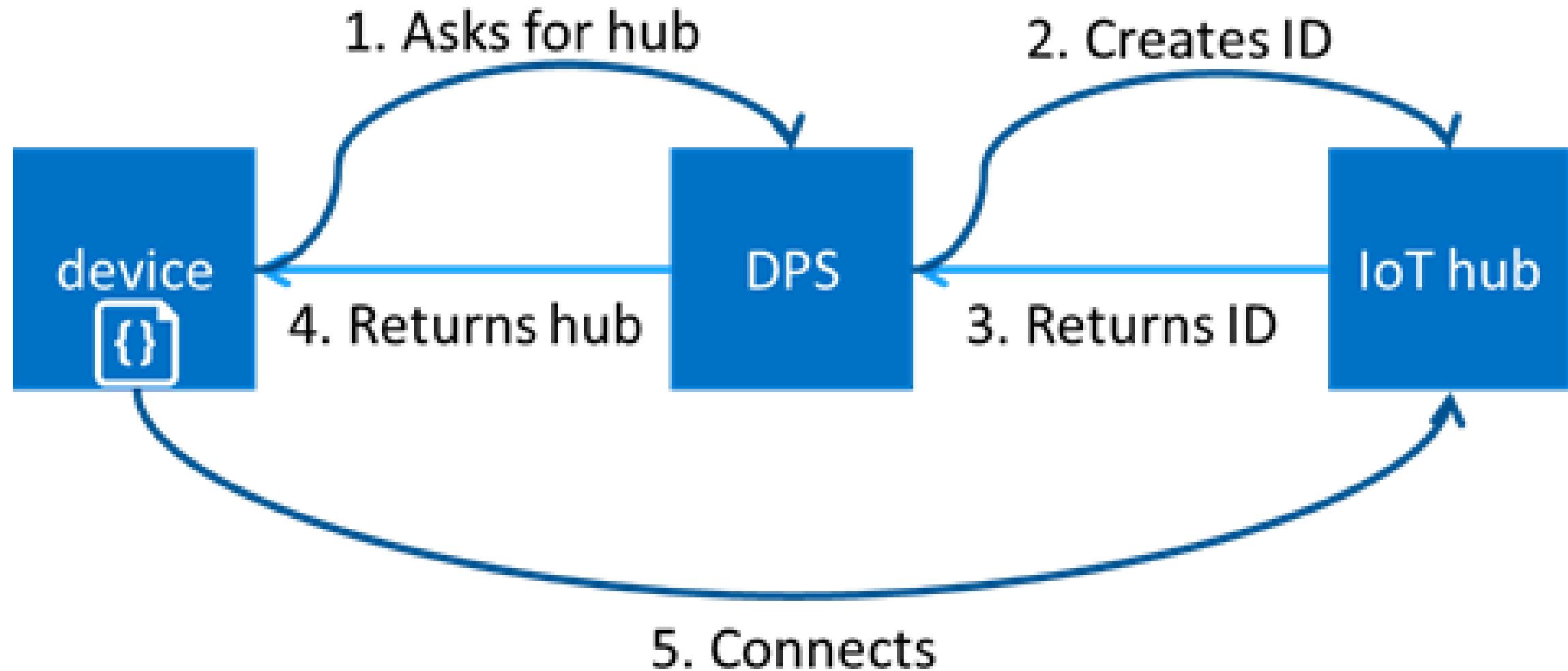
Requests a new token from service if disconnected, or connection is rejected



# NEW: Routing to Azure Storage



# NEW: Device Provisioning



# NEW: Message routing on Body

- Simple body reference
  - \$body.Weather.Temperature = 50
  - \$body.Weather.IsEnabled
  - \$body.message.Weather.Location.State = 'WA'
- Body array reference
  - \$body.Weather.HistoricalData[0].Month = 'Feb'
- Multiple body references
  - \$body.Weather.Temperature >= \$body.Weather.PrevTemperatures[0] + \$body.Weather.PrevTemperatures[1]
  - \$body.Weather.Temperature = 50 AND \$body.message.Weather.IsEnabled
- Combined with built-in functions
  - length(\$body.Weather.Location.State) = 2
  - lower(\$body.Weather.Location.State) = 'wa'
- Combination with message header
  - \$body.Weather.Temperature = 50 AND Status = 'Active'

# NEW: Azure Functions with IoT Hub

*One quick note: if you want to trigger an Azure Function on every message sent to IoT Hub, you can do that already! Just use the Event Hubs trigger and specify IoT Hub's built-in Event Hub-compatible endpoint as the trigger in the function. You can get the IoT Hub built-in endpoint information in the portal under Endpoints -> Events:*

The screenshot shows the 'dochub - Endpoints' blade in the Azure portal. On the left, a sidebar lists 'Operations monitoring', 'IP Filter', 'Properties', 'Locks', and 'Automation script'. Under 'MESSAGING', 'File upload' is listed, and 'Endpoints' is highlighted with a blue background. The main area displays 'Built-in endpoints' with a table:

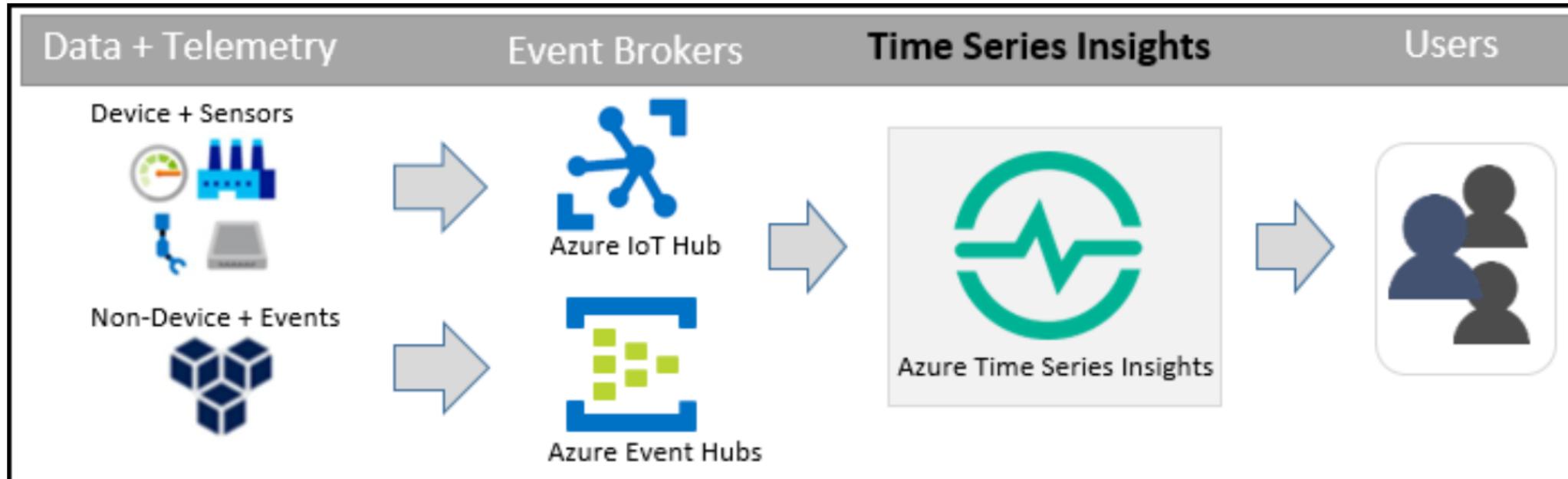
| NAME                     | ENDPOINT                       |
|--------------------------|--------------------------------|
| Cloud to device feedback | messages/servicebound/feedback |
| Events                   | messages/events                |

Below this is a section for 'Custom endpoints' with a note: 'You may have up to 10 endpoints on an IoT hub.' A modal window titled 'Properties' is open over the table, showing the configuration for the 'Events' endpoint:

- Device-to-cloud settings:
  - Partitions: 4
- Event Hub-compatible name: dochub
- Event Hub-compatible endpoint: sb://iothub-ns-dochub-93417-b13d7679
- Retention time: 1 days

*Here's where you enter that information when setting up your Function:*

# NEW: Time Series Insights



## Primary scenarios

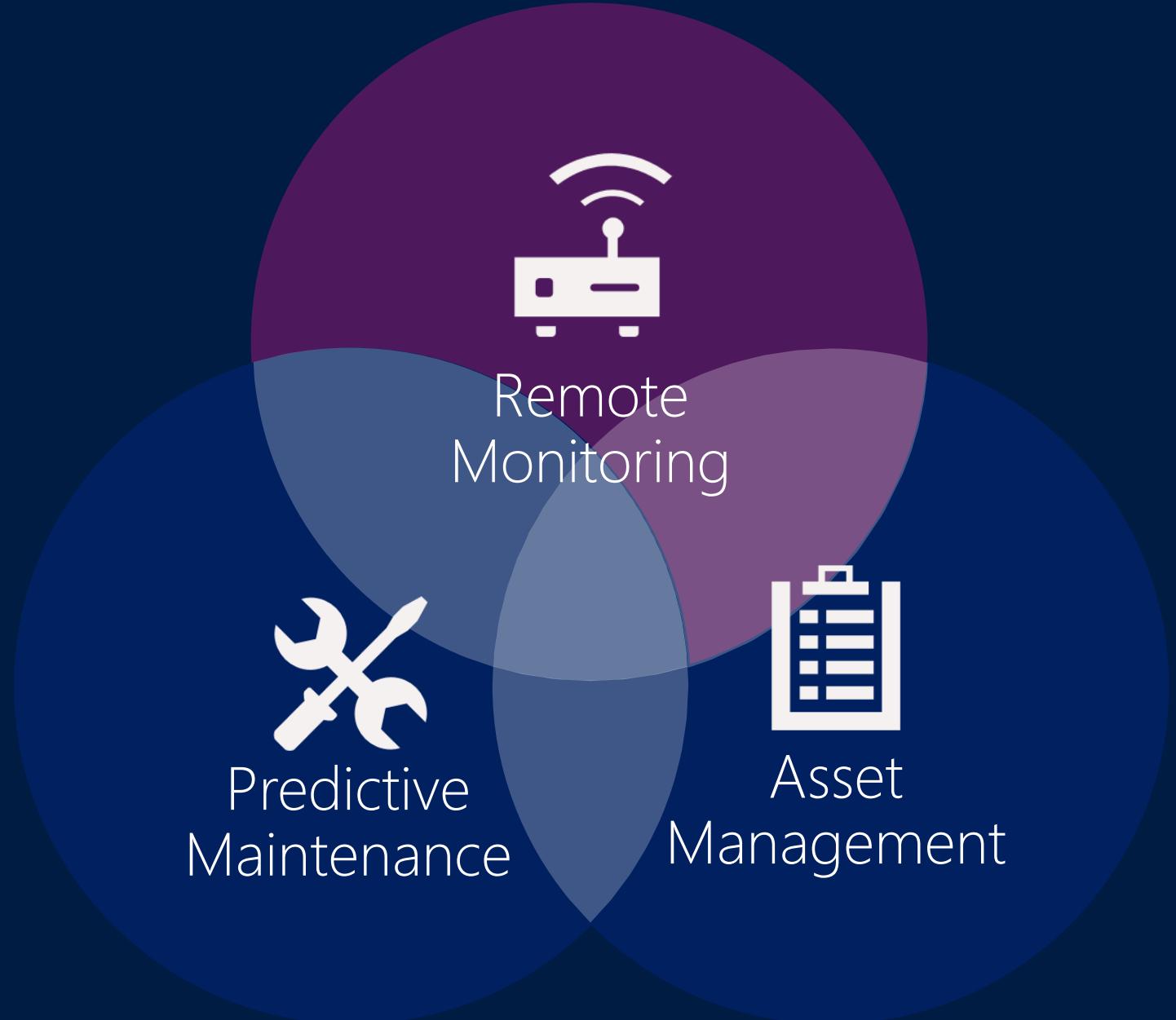
- Monitor and validate IoT solutions in minutes.
- Visualize and analyze IoT data at scale.
- Expedite root-cause analysis and anomaly detection.
- Create a global view of multiple devices, plants, and data.



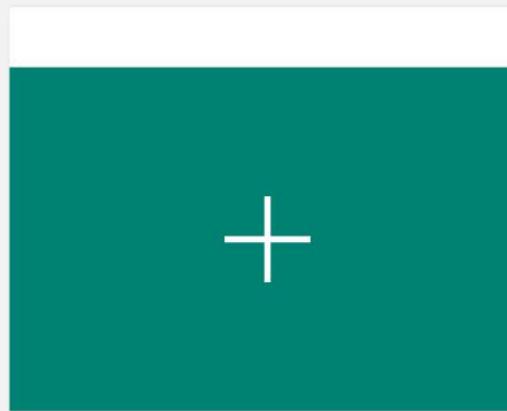
# IoT Suite



# Preconfigured Solutions: Remote Monitoring



# Provisioned solutions



## Create a new solution

Create your own fully integrated provisioning solution

Provisioning...



A worker wearing a blue hard hat and an orange safety vest with reflective stripes is standing in front of a complex industrial control panel. The panel features several cylindrical tanks labeled "MIGA JASCH", "INCA JASCH", and "DELA JASCH" with digital displays showing values like "202-T". Numerous pipes and valves are visible, with green arrows indicating flow paths. The worker is holding a walkie-talkie and looking towards the camera.

**RMsolution**

Monitor events and conditions from your devices in the field.

[Details](#)

## RMsolution



Provisioning your **Remote monitoring** solution, in **East US** region.



DASHBOARD



DEVICES



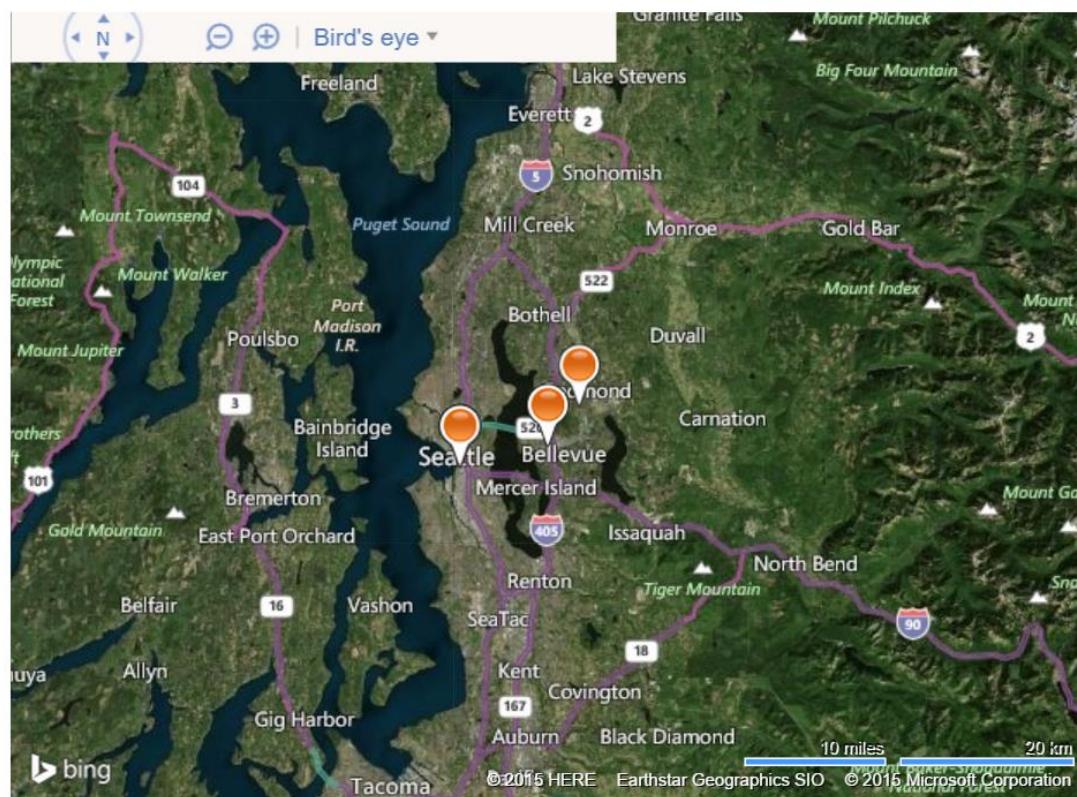
RULES



ACTIONS



ADD A DEVICE



## Alarm History

| TIME                     | DEVICE ID           | RULE OUTPUT   | VALUE  |
|--------------------------|---------------------|---------------|--------|
| 09/29/2015<br>9:22:25 AM | SampleDevice001_363 | AlarmTemp     | 43.817 |
| 09/29/2015<br>9:22:25 AM | SampleDevice001_363 | AlarmHumidity | 22.588 |
| 09/29/2015<br>9:22:23 AM | SampleDevice001_363 | AlarmTemp     | 42.933 |
| 09/29/2015<br>9:22:23 AM | SampleDevice001_363 | AlarmHumidity | 25.135 |

Device to View: SampleDevice001\_363

## Telemetry History

Humidity Temperature

45

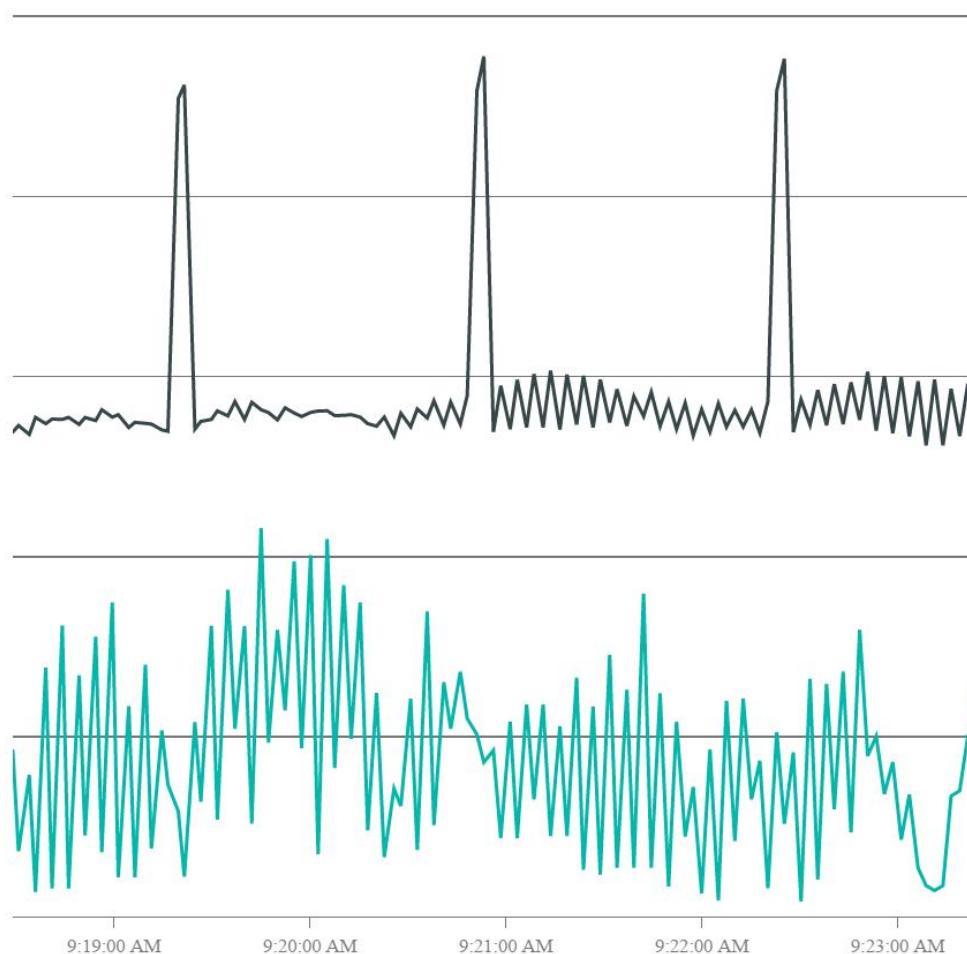
40

35

30

25

20



Max of device humidity



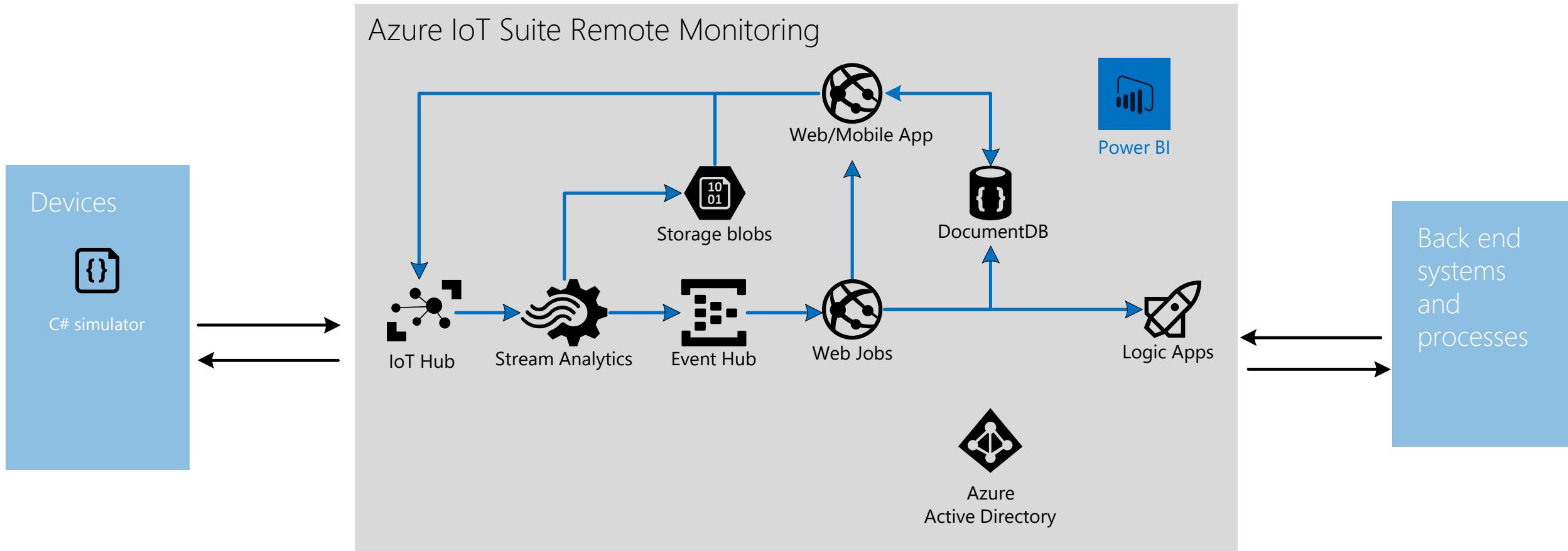
Min of device humidity



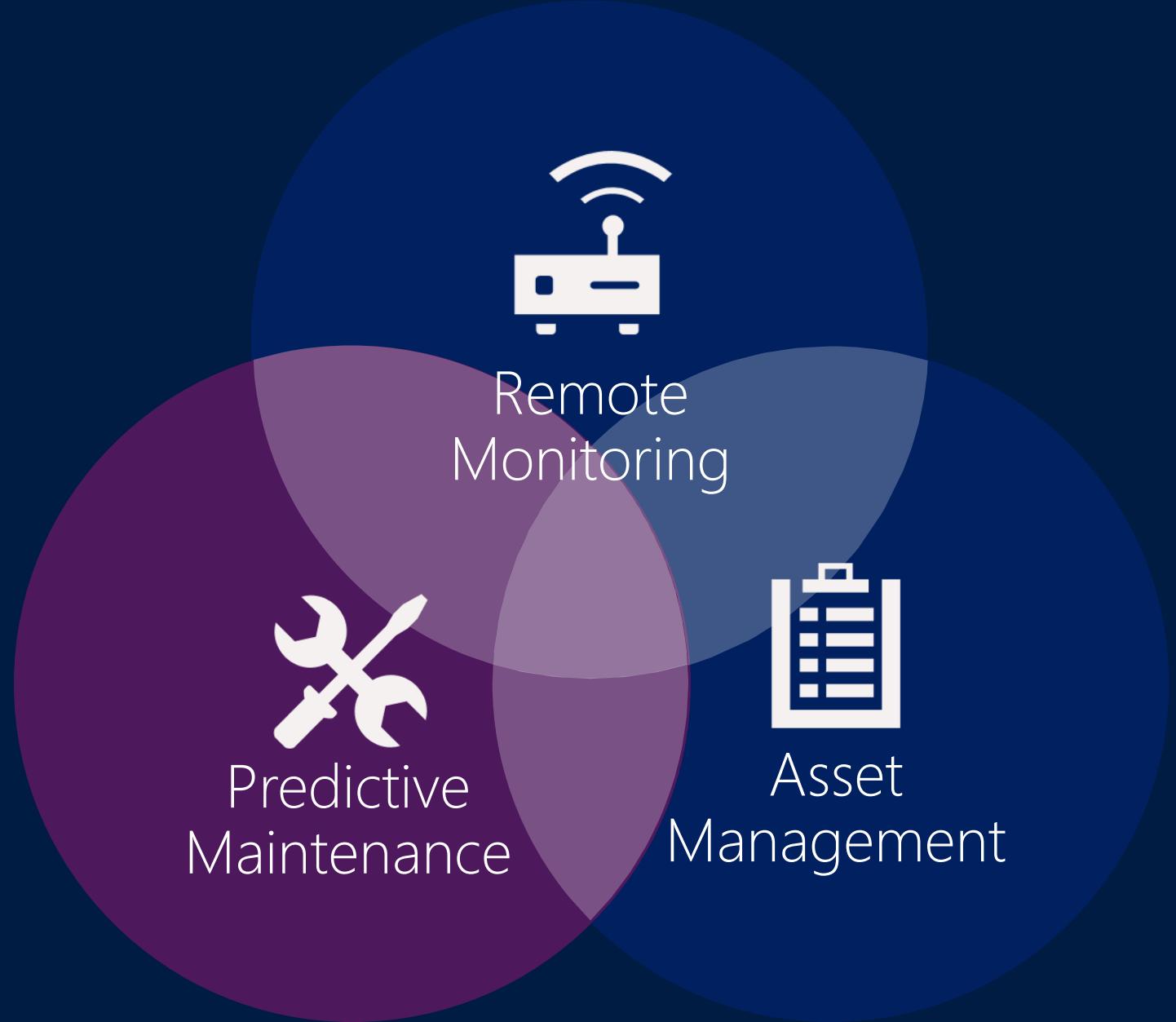
Average of device humidity



# What you get with remote monitoring preconfigured solution



# Preconfigured Solutions: Connected Factory



Connected to OPC server : opc.tcp://scada2194.munich0.corp.contoso:51210/UA/Munich/ProductionLine0/AssemblyStation

Root

Server

StationInstance

StationProduct

StationTelemetry

OverallRunningTime

✓ FaultyTime

Status

✓ EnergyConsumption

✓ Pressure

✓ IdealCycleTime

✓ ActualCycleTime

StationCommands

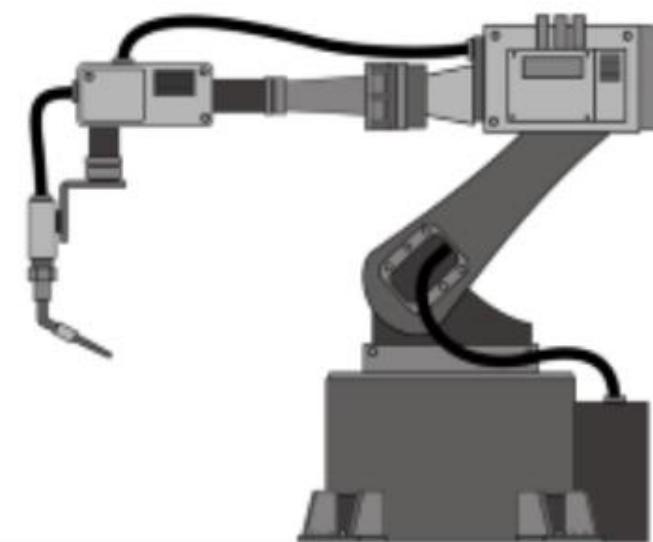
Execute

Reset

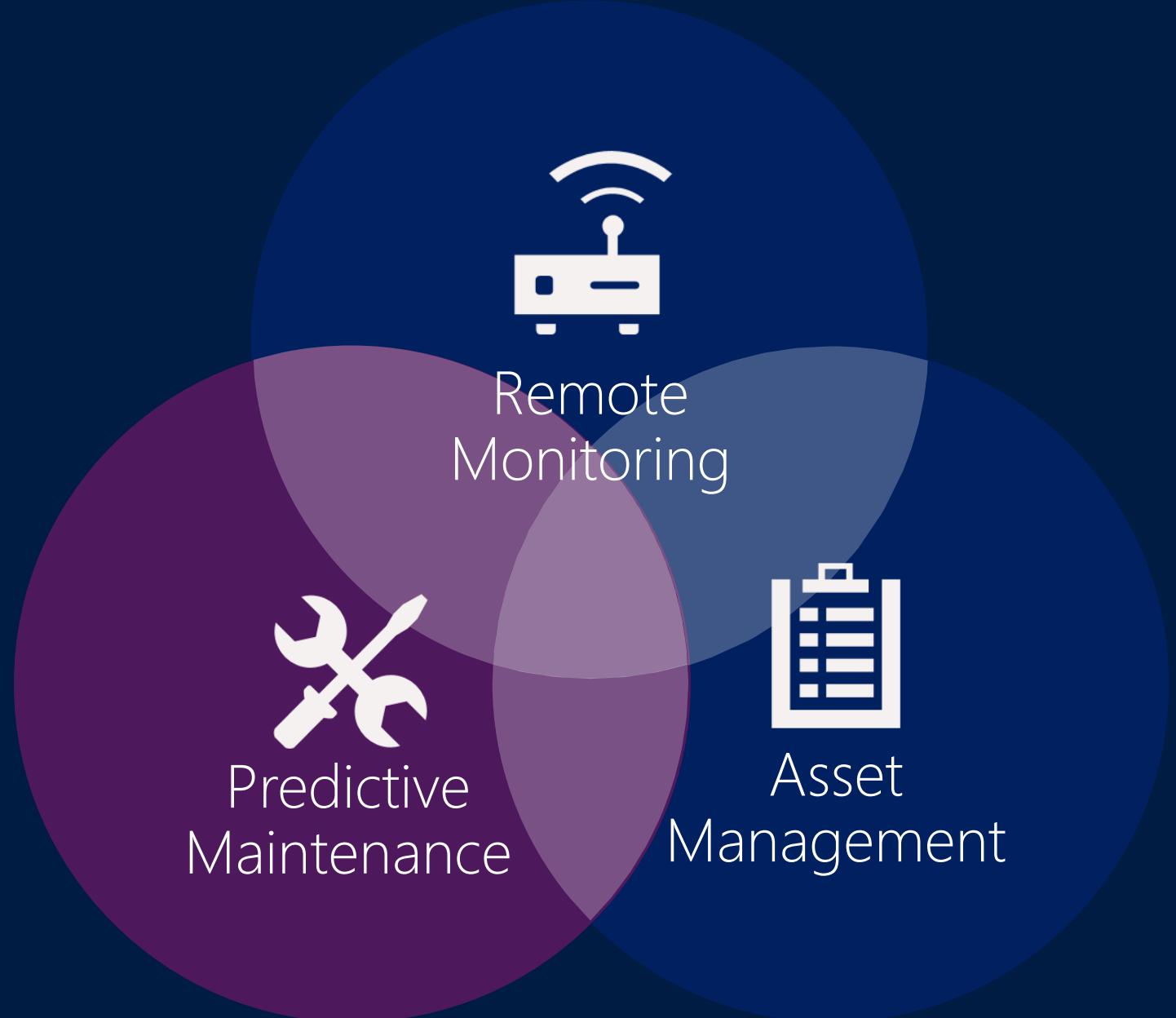
OpenPressureReleaseValve

Call...

Disconnect



# Preconfigured Solutions: Predictive Maintenance



## Solution types



### Predictive maintenance

Anticipate maintenance needs and avoid unscheduled downtime by connecting and monitoring your devices for predictive maintenance.

Select



### Remote monitoring

Connect and monitor your devices to analyze untapped data and improve business outcomes by automating processes.

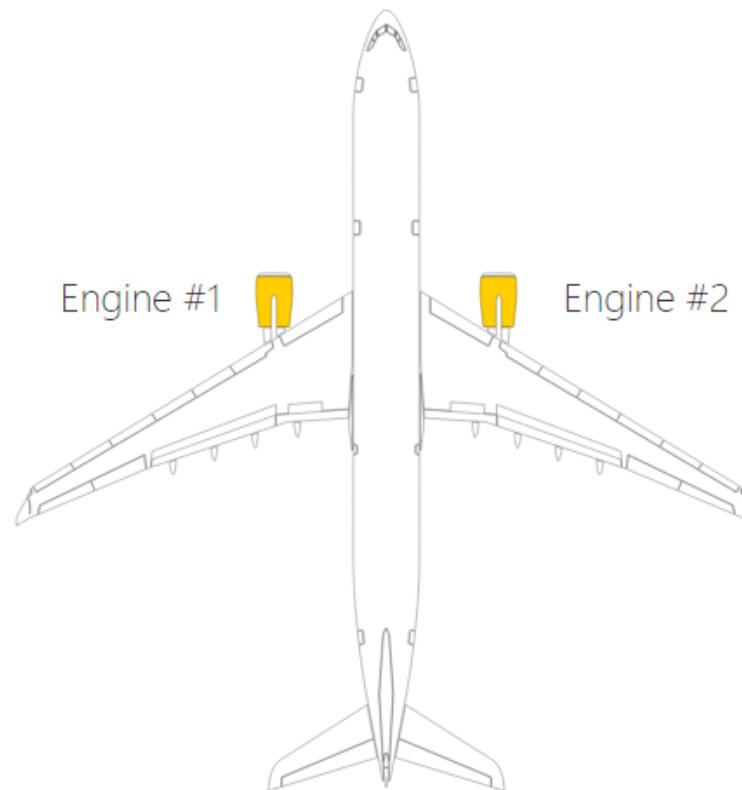
Select



DASHBOARD

## Aircraft map

Simulation in progress

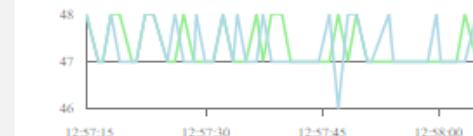


Stop simulation

## Sensor history

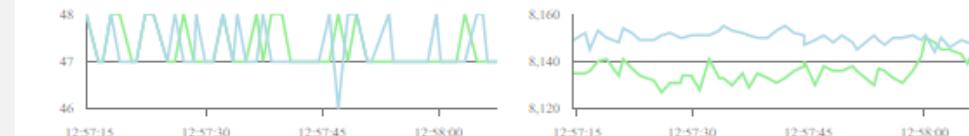
Sensor 9

Engine 1 Engine 2



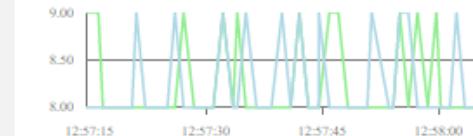
Sensor 11

Engine 1 Engine 2



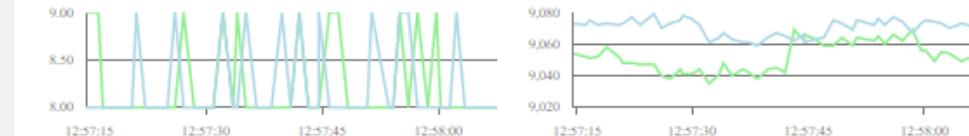
Sensor 14

Engine 1 Engine 2



Sensor 15

Engine 1 Engine 2

Remaining Useful Life (RUL)  
IN DAYS

159 !

ENGINE #1

139 !

ENGINE #2

Cycles  
#

72

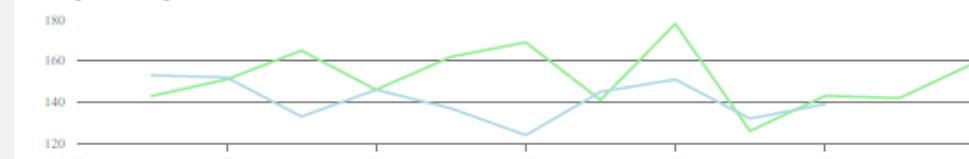
ENGINE #1

73

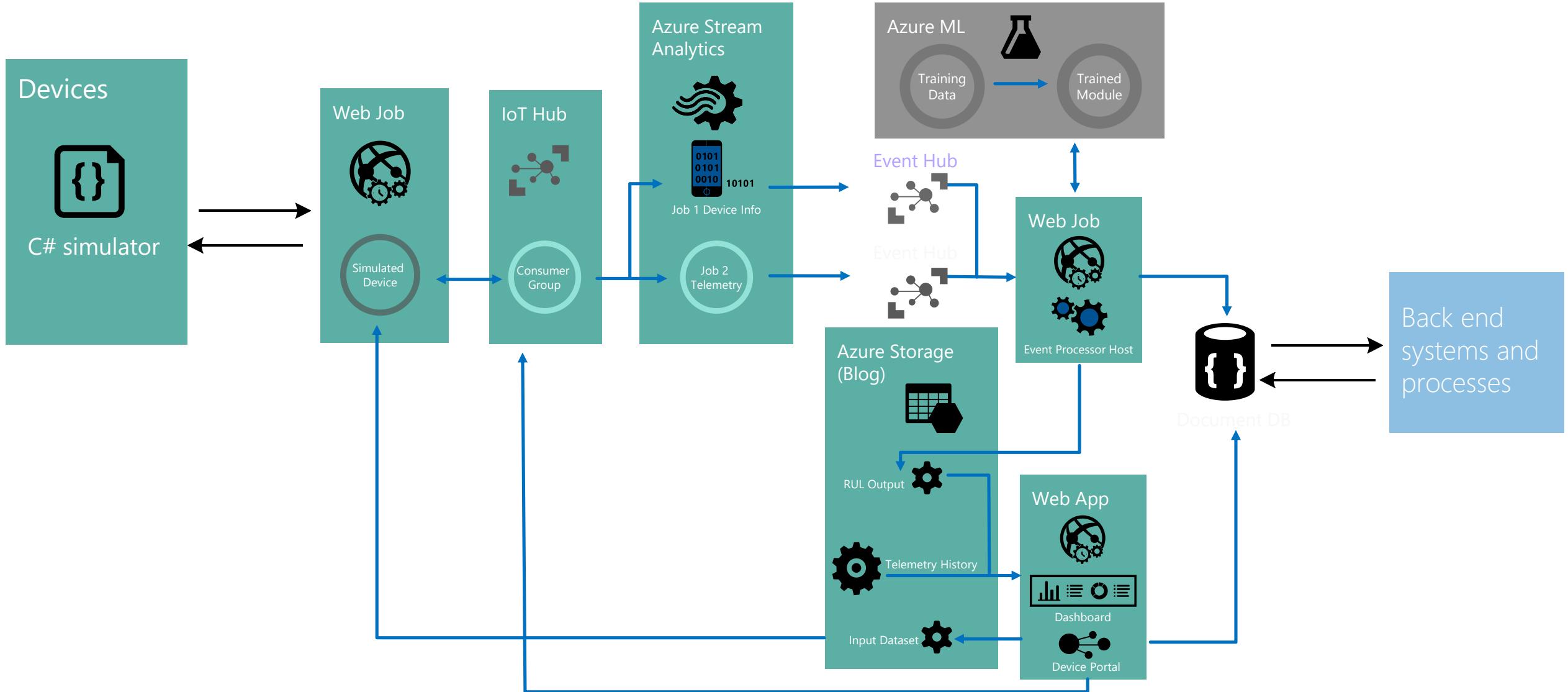
ENGINE #2

Remaining Useful Life (RUL) history  
IN DAYS

Engine 1 Engine 2



# What you get with predictive maintenance solution

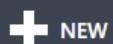
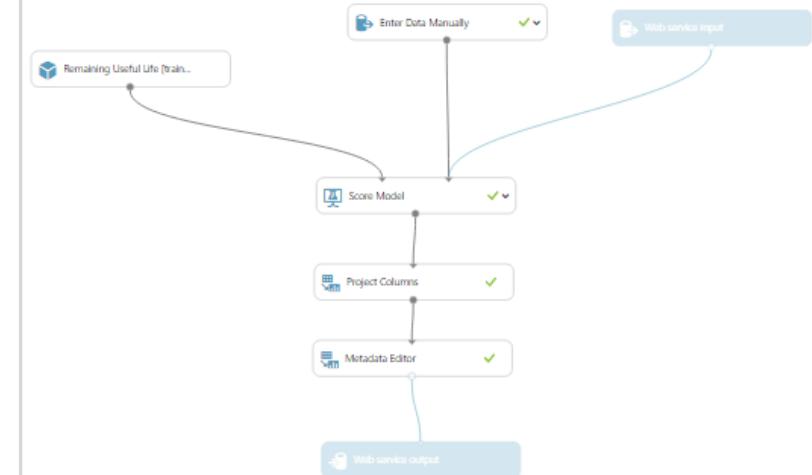




## experiments

MY EXPERIMENTS SAMPLES

|  | NAME                                | AUTHOR                | STATUS   | LAST EDITED          |  |
|--|-------------------------------------|-----------------------|----------|----------------------|--|
|  | Remaining Useful Life [Predictiv... | Microsoft Corporation | Finished | 12/1/2015 4:08:19 AM |  |
|  | Remaining Useful Life [Predictiv... | Microsoft Corporation | Draft    | 12/1/2015 4:08:11 AM |  |
|  | Remaining Useful Life Engines       | Microsoft Corporation | Draft    | 12/1/2015 4:08:03 AM |  |



# Push & Broadcast Notifications Notification Hubs



# Push Notifications

## Register device handle at app launch

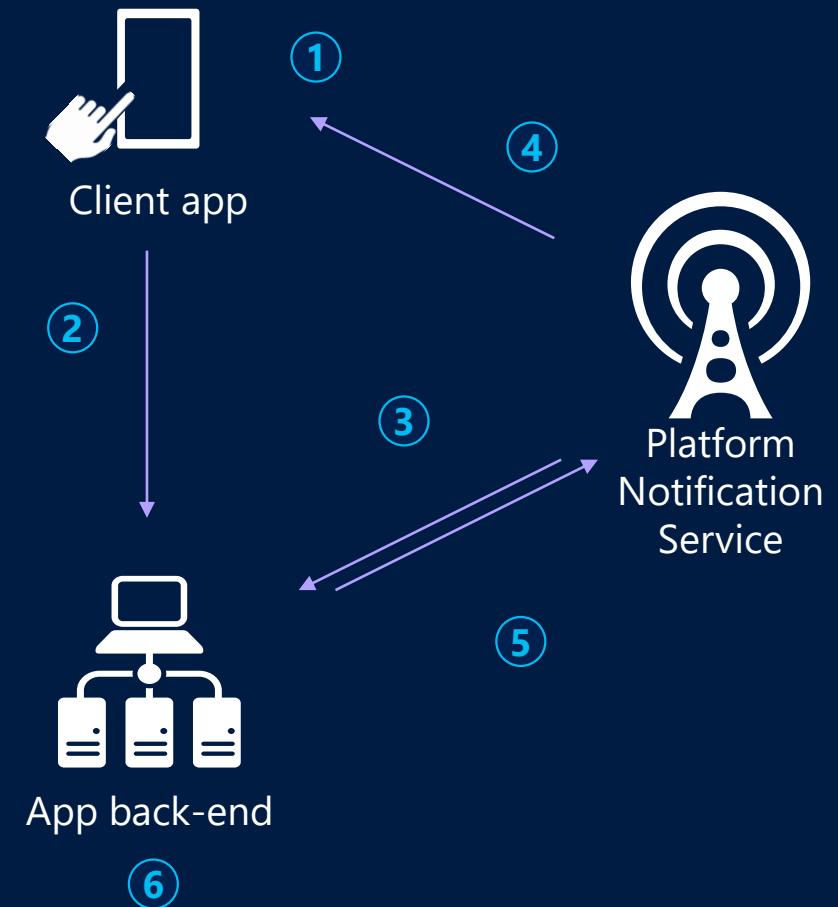
- ① Client app retrieves handle from Platform Notification Service (PNS)
- ② App passes the handle to the back-end

## Send Notification

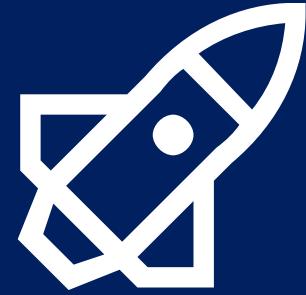
- ③ App back-end sends notification to Platform Notification Service (PNS), uses PNS credentials to authenticate
- ④ PNS pushes the notification to the app on the device

## Maintain device handles

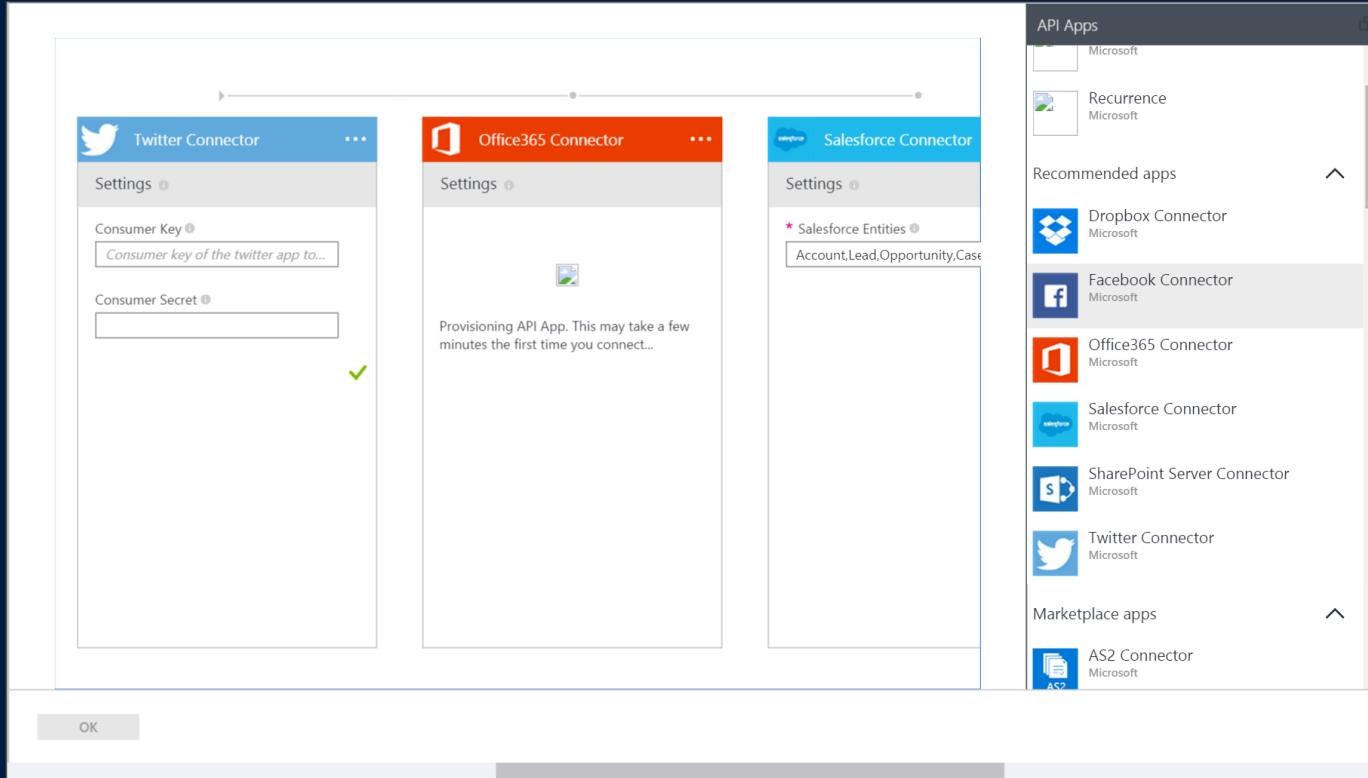
- ⑤ Backend deletes expired handles when PNS rejects them
- ⑥ Maintain mapping between logical users/groups and device handles



# Workflow Integration Logic Apps



# Drag and drop connector integration

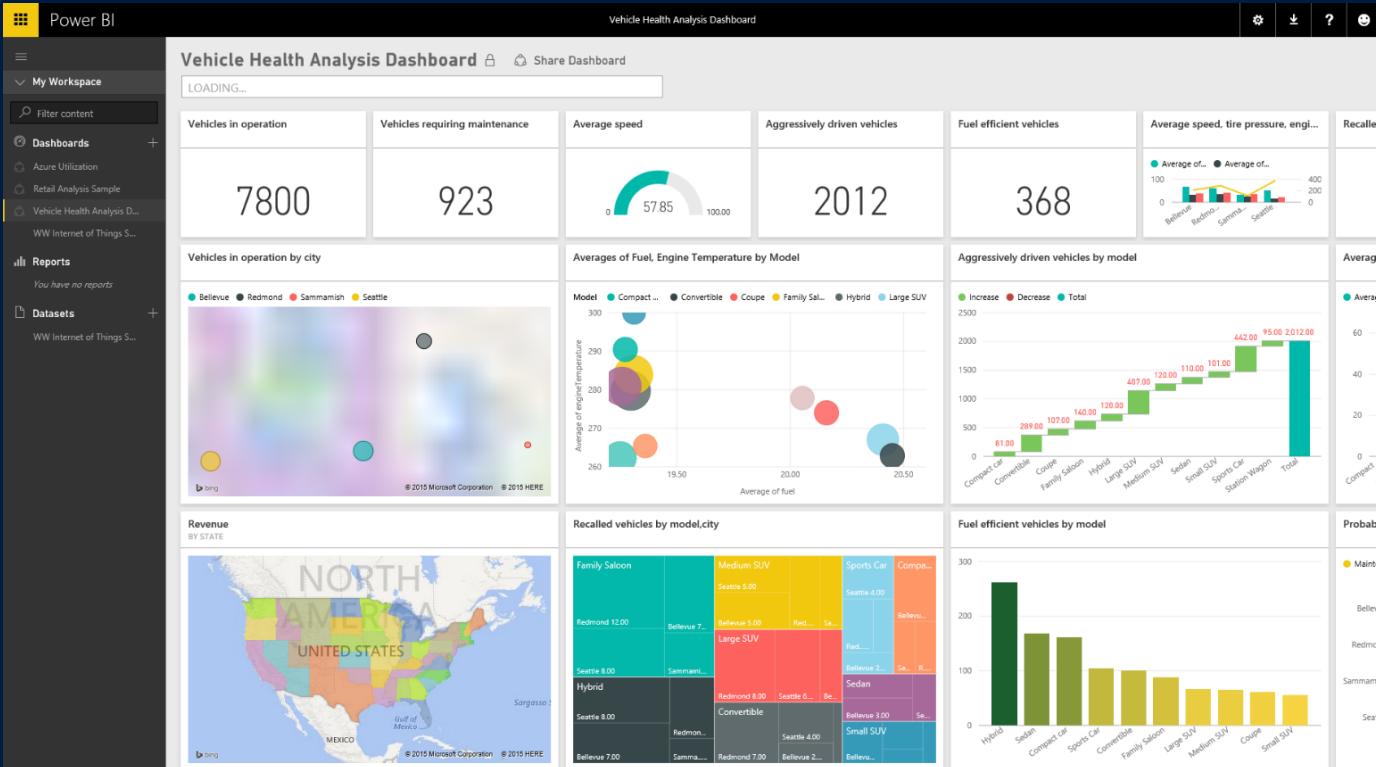


Drag and drop UI  
Simple connector integration  
Authentication  
Enter product keys and authentication data

# Data Visualization Power BI



# Data visualization with PowerBI



Rich visuals

Standard and custom graphing options

Custom dashboards

Build heat maps and visually track data

External data

Integrate external data feeds to add value to device data, or pull in external information such as weather or market information.



# Security







# Mass Production



# Mass Production



OR





# Q&A





Microsoft