Microsoft Azure

Azure IoT Academy

Transforming your business

Marianela Ramsdell
Sr. Technical Specialist - Global Black Belt
Microsoft

Manisha Kumari
Sr. Technical Specialist – Global Black Belt
Microsoft

Agenda

- Introduction to IoT Central
- Azure IoT Central use cases
- Pricing
- Customer evidence
- Embedded C SDK
- Resources


Azure IoT Central
Microsoft

Developing an IoT solution can be challenging

Complexity

IoT requires numerous skill sets.

Time

Many IoT solutions require months to set up.

Cost

Heavy up-front investments can be cost-prohibitive.

Security

IoT poses unique security, privacy, and compliance challenges.

Use | Solutions

Build your own | Platform

SaaS

Managed Solutions

IoT Central

Connected Field Service

Solution Accelerators

Remote monitoring

Predictive maintenance

Connected factory

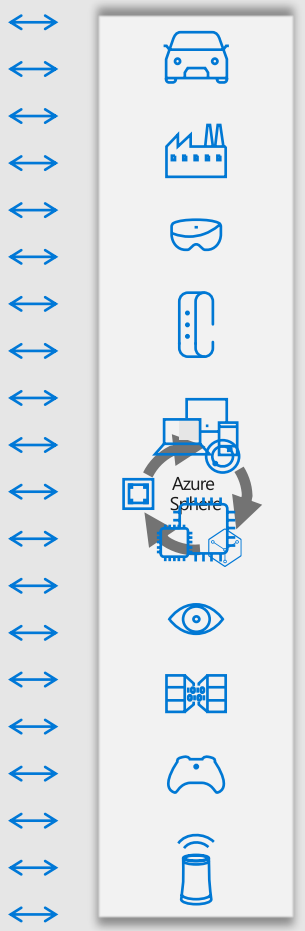
...

Device Support

Data & Analytics

Visualization & Integration

IoT Edge



Partner Solutions

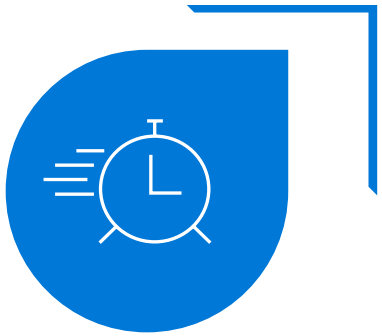
Experience the simplicity of SaaS for IoT with Azure IoT Central

Get results fast by leveraging a fully-managed SaaS solution with built-in analytics

Grow and scale with ease, securely with a solution that integrates with your business

Enterprise grade solution built on proven services and decades of IoT and SaaS experience

Azure IoT Central empowers you to…

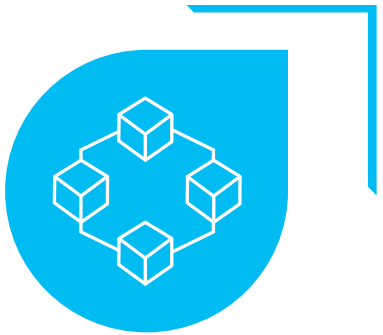


Get results fast

- Build production-grade applications in hours
- Remove the management burden, operational cost, and overhead
- Understand the total cost of ownership (TCO) with transparent and predictable per-device pricing

Grow and scale with ease, securely

- Implement IoT projects from small to large
- Integrate with your existing business systems to execute workflows, such as field service support and alerts
- Leverage industry-leading security standards and data protection features to keep you in control of your data



Enterprise grade solution

- Utilize hyperscale and enterprise-grade services provided by Azure and IoT
- Access the latest advances when you need them
- Bring your connected product strategy to life by keeping your most critical data secure

Azure IoT Central

- Fully hosted and managed by Microsoft
- No cloud development expertise required
- Device connectivity and management
- Monitoring rules and triggered actions
- Extensibilities (Flow, Dynamics, Webhooks, etc.)
- Analytics, dashboards and visualization
- Risk-free trial with simplified pricing

Introduce: What are the Azure IoT Central use cases?

Primary use cases

EQUIPMENT MONITORING

CONNECTED FIELD SERVICE

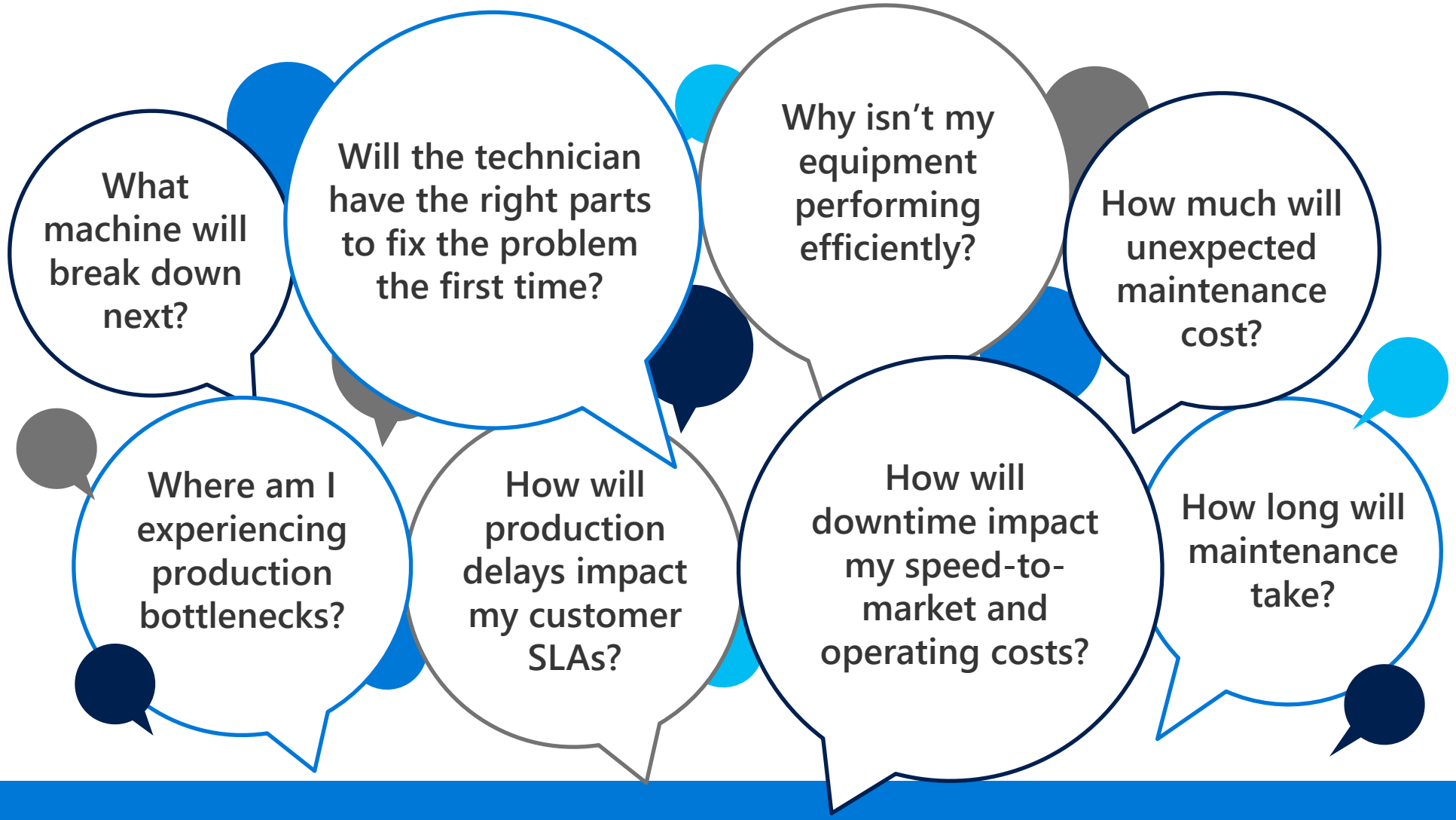
PRODUCT-AS-A-SERVICE

Other use cases

- Asset tracking
- Predictive maintenance
- Product design/R&D

Equipment Monitoring

You have a lot of unknowns when it comes to equipment



70% of companies have a lack of awareness about when their assets need to be maintained.[1]

IoT data delivers the visibility you need for better insights and faster resolution

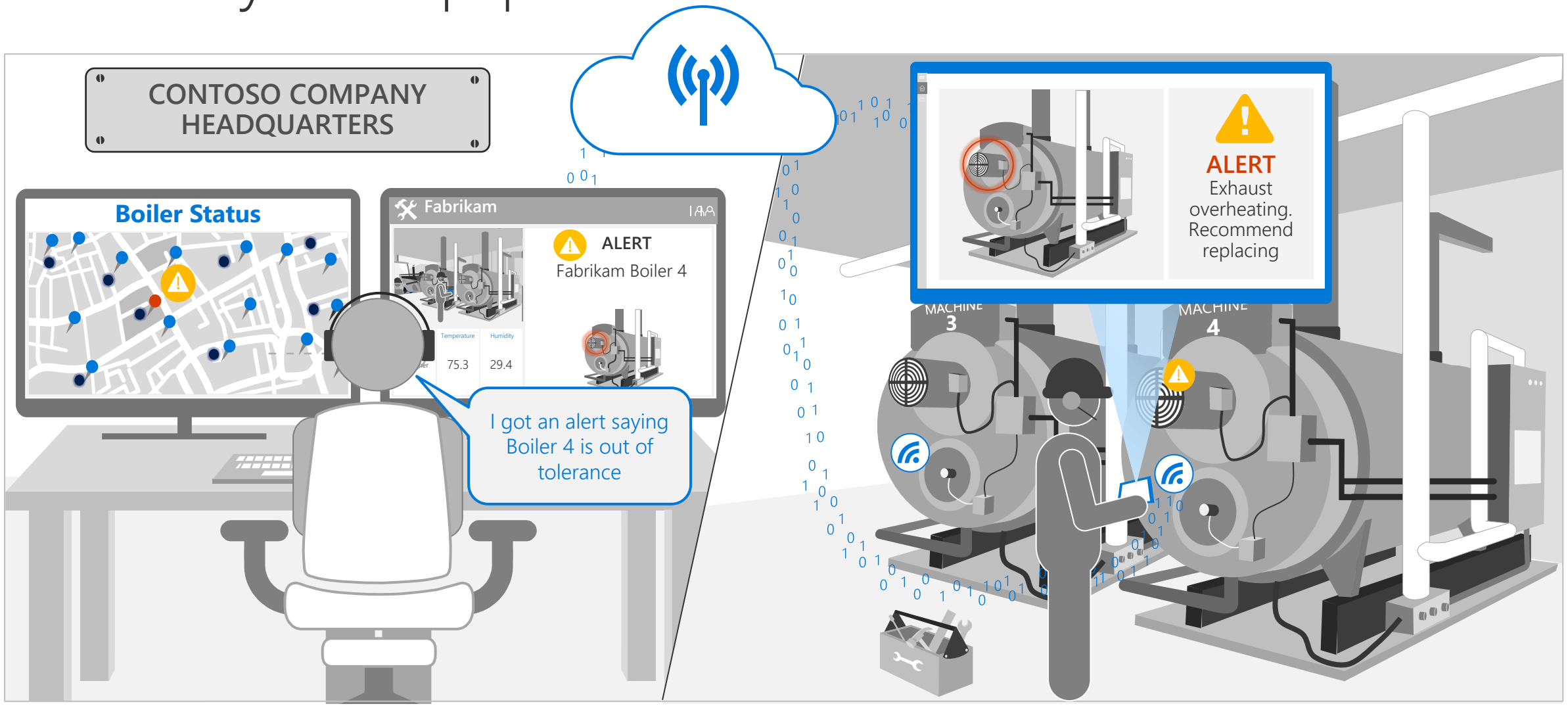
Helps technicians identify and repair malfunctioning assets before damage occurs

Reduces the need for service calls by enabling technicians to remotely diagnose equipment issues

Arms technicians with the diagnostic information they need to improve first-time fix rates

Enables service organizations to analyze equipment failure patterns to improve maintenance strategies

Monitor your equipment with IoT



Improve visibility and efficiency across your equipment

Make targeted improvements to business processes

Proactively anticipate and prevent equipment issues

Introduce: Equipment Monitoring

What is the value proposition?

Gather data from connected assets and products using a secure and cost-effective cloud platform that eliminates the need to invest in additional infrastructure

What is it?

- Equipment Monitoring is an Azure IoT Central use case that enables organizations to collect data from and monitor the health of connected assets

What does it do?

- Ingests data from connected assets and equipment
- Alerts technicians when asset performance deviates from predefined settings
- Provides technicians with secure access to data from connected assets
- Enables technicians to remotely update and control assets and equipment both in a facility or out in the field

How does it add value?

- Helps technicians identify and repair malfunctioning assets before damage occurs
- Reduces the need for service calls by enabling technicians to remotely diagnose equipment issues
- Arms technicians with the diagnostic information they need to improve first time fix rates
- Enables service organizations to analyze equipment failure patterns to improve maintenance strategies

Target decision makers:

- Product leadership
- Sales/Services leadership
- Operations leadership

Target industry:

- Manufacturing
- Transportation
- Retail
- Government
- Healthcare

Equipment monitoring examples:

- Fleet management
- Engines
- Onshore and offshore drilling rigs
- Elevators
- Power generators
- Industrial equipment
- Digital signage
- Cold chain
- Product display racks

Connected Field Service

Field service can make or break the customer experience

The average first-time fix rate is 74%, meaning that more than a quarter of service calls require return visits[1]

Common reasons

Technician lacked the right tool or part

Technician didn't have the right information about the issue

Technician didn't have the expertise to solve the issue

Consequences

Dissatisfied customers

Broken SLAs

Frustrated technicians

Wasted time and resources

1 The Service Council

IoT supports better field service outcomes

Gain visibility into performance by connecting and remotely monitoring your devices

Increase first time fix rates by equipping your technicians with information, augmented reality tools and remote guidance

Minimize downtime by creating alerts and automated workflows that enable predictive maintenance

Accelerate time to value through cloud deployment

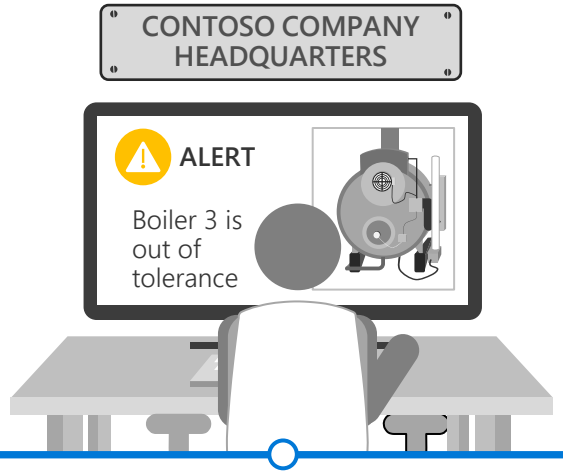
Reduce maintenance costs and truck rolls by optimizing schedules and skill profiles to dispatch the right technician only when needed

Transform your products and services by gaining insights from repair, performance, and utilization data

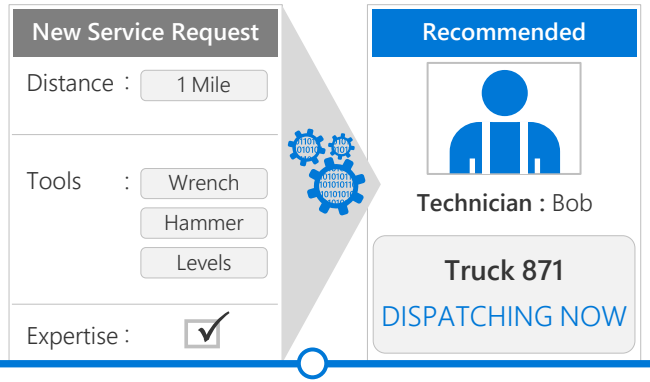
Connected Field Service increased technician productivity by an average of 25%[1]

An effective preventative maintenance program delivers average savings of over 12%[2]

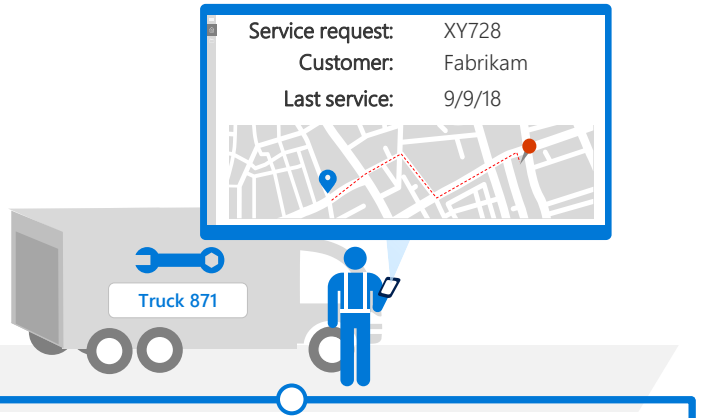
Transform field service with IoT



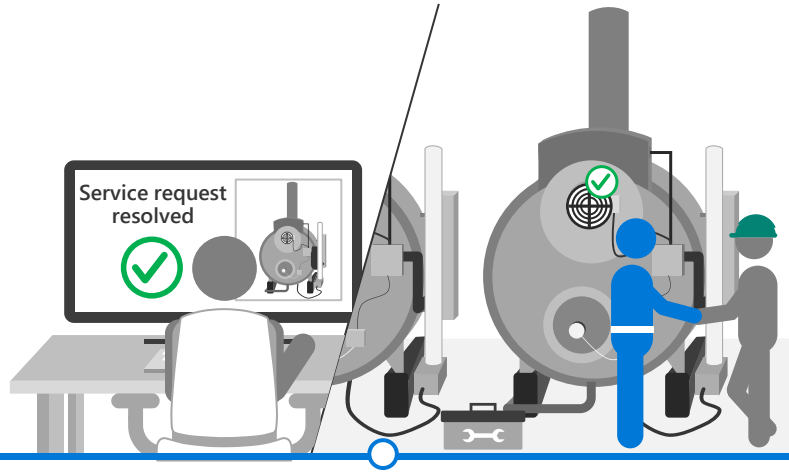
Automatically trigger work orders



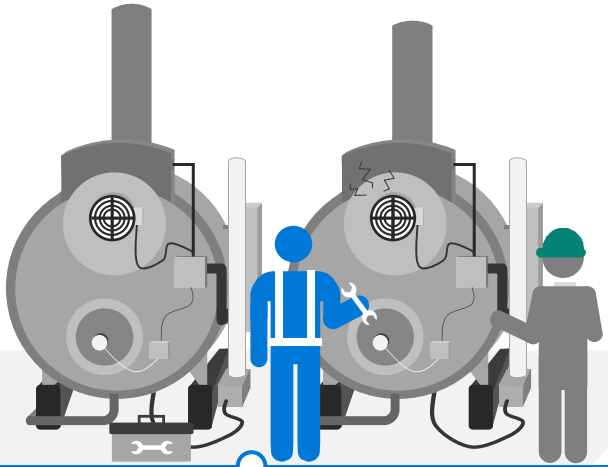
Dispatch the right technician with the right skills



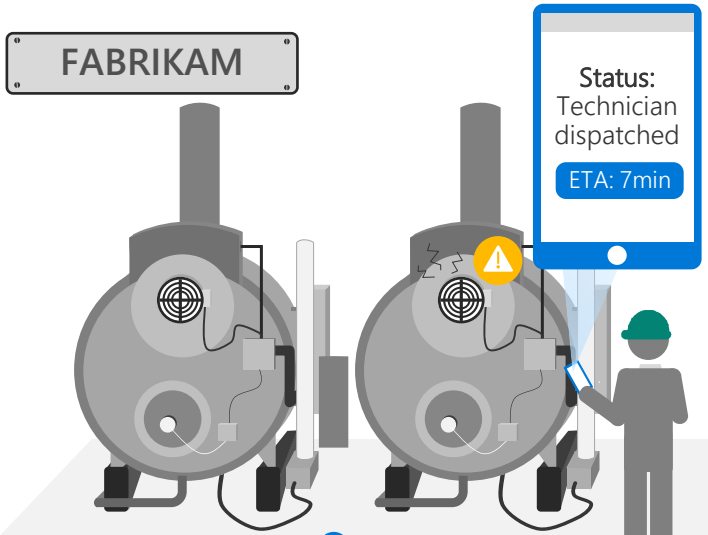
Look up case history and information on mobile devices



Update and view cases in real time



Ensure a first time fix and collect customer information on mobile devices



View technician status, schedules, and arrival times

Introduce: Connected Field Service

What is the value proposition?

Increase profitability and ensure a first time fix by dispatching the right people, parts, and equipment for every maintenance visit

What is it?

- Connected Field Service helps companies move from a costly break-fix model to a proactive and predictive service model

What does it do?

- Automatically generates targeted work orders using data from connected assets
- Optimizes scheduling based on service need, technician skill, and customer location
- Tracks parts inventory down to the service truck level
- Catalogues customer preferences, equipment health and performance, and service history details
- Enables equipment to automatically request proactive service and customers to track technicians online

How does it add value?

- Reduces downtime by enabling service organizations to rapidly dispatch technicians
- Helps service teams improve their first time fix rate by selecting the right technicians and parts for each call
- Increases service efficiency by optimizing service call assignments, routes, and scheduling
- Increases customer satisfaction by ensuring technicians are aware of service and equipment history
- Reduces inventory costs by utilizing predictive inventory forecasting

Target decision makers:

- Service leadership
- Sales leadership
- Finance leadership
- Operations leadership

Target industry:

- Manufacturing
- Government
- Retail

Connected Field Service examples:

- Water towers
- Espresso machines
- Cold chain
- Packaging machinery
- Engines
- Industrial equipment
- Bridge infrastructure

Introduce: Product-as-a-Service

What is the value proposition?

Provide services to customers which generate value, create new revenue streams, and differentiate product offering

What is it?

- Product-as-a-service enables companies to create new service-oriented business models, better respond to customer needs, and monetize data via connected devices

What does it do?

- Offers asset management services by remotely monitoring performance and controlling devices
- Removes the burden of asset management from customers to improve customer experience
- Enables faster iteration of the next generation of smart products by collecting insights instantly and continuously

How does it add value?

- Build new competitive advantages by adding more value for customers
- Enables data and insights from connected products to be converted to service revenue opportunities
- Decreases the time and resources required to maintain equipment and devices
- Increases customer satisfaction by delivering new insights into equipment health and performance

Target decision makers:

- Product/engineering leadership
- Sales leadership
- Finance leadership
- Operations leadership
- Service leadership

Target industry:

- Manufacturing

Product-as-a-Service examples:

- Engines
- Air compressors
- Connected chillers
- Power generators
- Industrial equipment
- Retail equipment

Introduce: Additional use cases

Use case	What is it?	What value does it add?
Asset tracking	Track physical assets and their information including location, status, and usage	Secure assets that are critical to business operations
Predictive maintenance	Identify equipment failure patterns using sensor and service data and provide device-specific maintenance recommendations	Reduce downtime, increase efficiency, and utilize insights from service data
Product design/R&D	Collect data and gain insight into device usage, location, temperature, and other valuable information	Improve the next generation of products to better meet customer needs

Start Free

Free forever
for up to 5 devices

Pay-As-You-Go

Per device, per month
with no hidden cost

Pricing

Per Device Pricing (Each Device comes with 50,000 included messages per month)	0 - 5 6 - 1,000 1,001 - 10,000 10,001 - 100,000 100,001+	Free €1.7 per device/month €1.3 per device/month €0.9 per device/month €0.5 per device/month
Additional Message Pricing[1]	per 1M messages	€4.217

[1]The maximum message size is 1 KB. For example, if the device sends a 4.5 KB message it will be billed as 5 messages.

https://azure.microsoft.com/en-us/pricing/details/iot-central/

Next steps

- Explore the solution with a free trial at www.AzureIoTCentral.com
- Skill up at IoT School
- Select a partner
- Contact us
- Start a Proof of Concept quickly and easily

Customer evidence

Johnson Controls

Connected chillers come back online **9x faster** than unconnected equipment, avoiding more than **$300,000** in hourly downtime costs

Rolls-Royce

Rolls Royce provides maximize aircraft availability by employing “power by the hour” model. Cutting fuel usage by 1 percent could save **$250,000** per plane per year

DUNAV NET

Keeping farmers informed about when to irrigate, how to control diseases, and fighting pests, has led to **increased yields of 30%, and a 20% reduction in water use**.

Rockwell Automation

Improves **access** to production and supply chain **data** worldwide, reducing downtime costs by as much as **$300,000 per day**

HERSHEY'S

Licorice extruders on Twizzler’s production line are performing at peak optimization, **saving over $500K/year** on materials alone

Tetra Pak

Tetra Pak’s has cut down-time cut down for each packaging line by up to 48 hours, **saving €30,000 for customers**.

RAC

RAC has reduced its accident rate by 25% and fuel usage by 20%, reporting **annual savings of $1.8 million**

thyssenkrupp

Gathers data from sensors and systems to create valuable business intelligence and **reduce downtime by 50%**

Tetra Pak · FUJITSU · Schneider Electric · ABB · LIEBHERR · THE BOLD LOOK OF KOHLER · SANDVIK Coromant · ECOLAB · JABIL · FINNING

Microsoft


Azure IoT
Embedded C SDK
IoT for Tiny Devices
Dane Walton
Ewerton Scaboro
Wellington Duraes

Microsoft

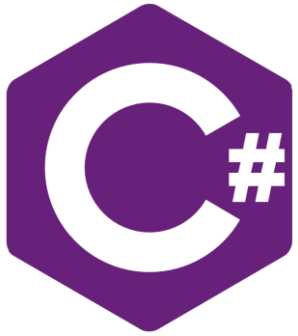
Microsoft

Thank you

Azure IoT Embedded C SDK

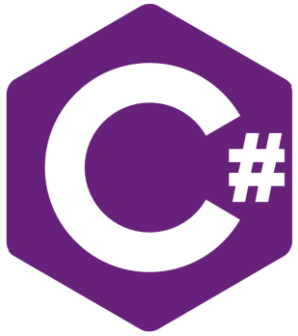
IoT for Tiny Devices

Azure IoT SDKs – Device SDK



- Device to Cloud – D2C
- Cloud to Device – C2D
- Direct Methods
- Device Twin
- Device Provisioning

Azure IoT SDKs – Device SDK

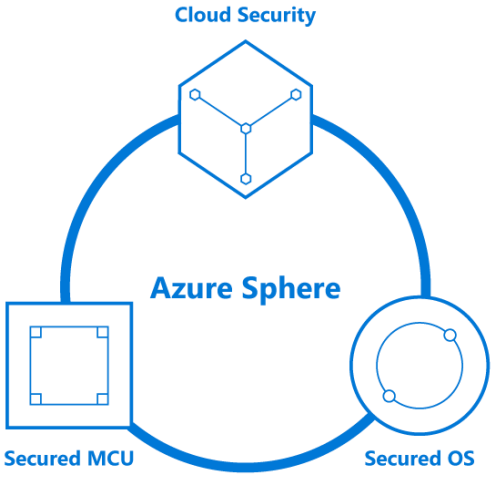


Embedded

1. Optimize for size
2. Give customer choice
3. No dynamic memory allocation
4. Fully embraces MQTT
5. BYO Network Stack
6. Everywhere SDK

Our story for Embedded

Azure Sphere	Azure RTOS	Embedded C SDK



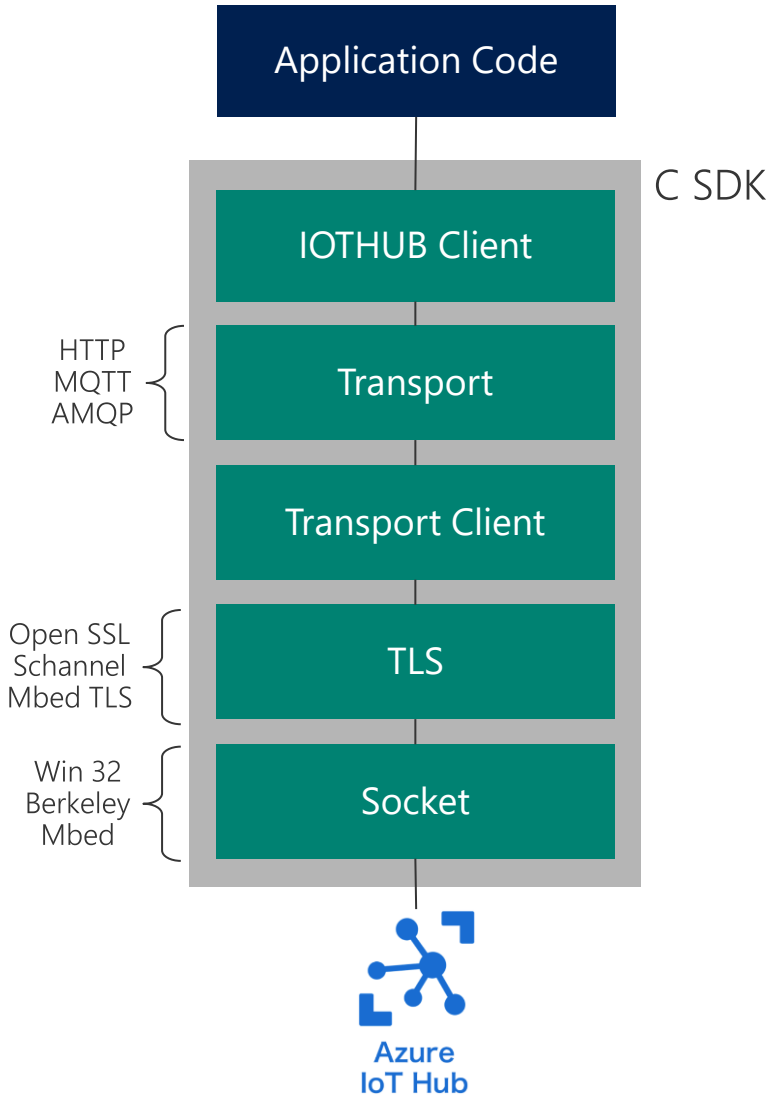
Embedded

We meet our customers where they are.

Our story for Embedded

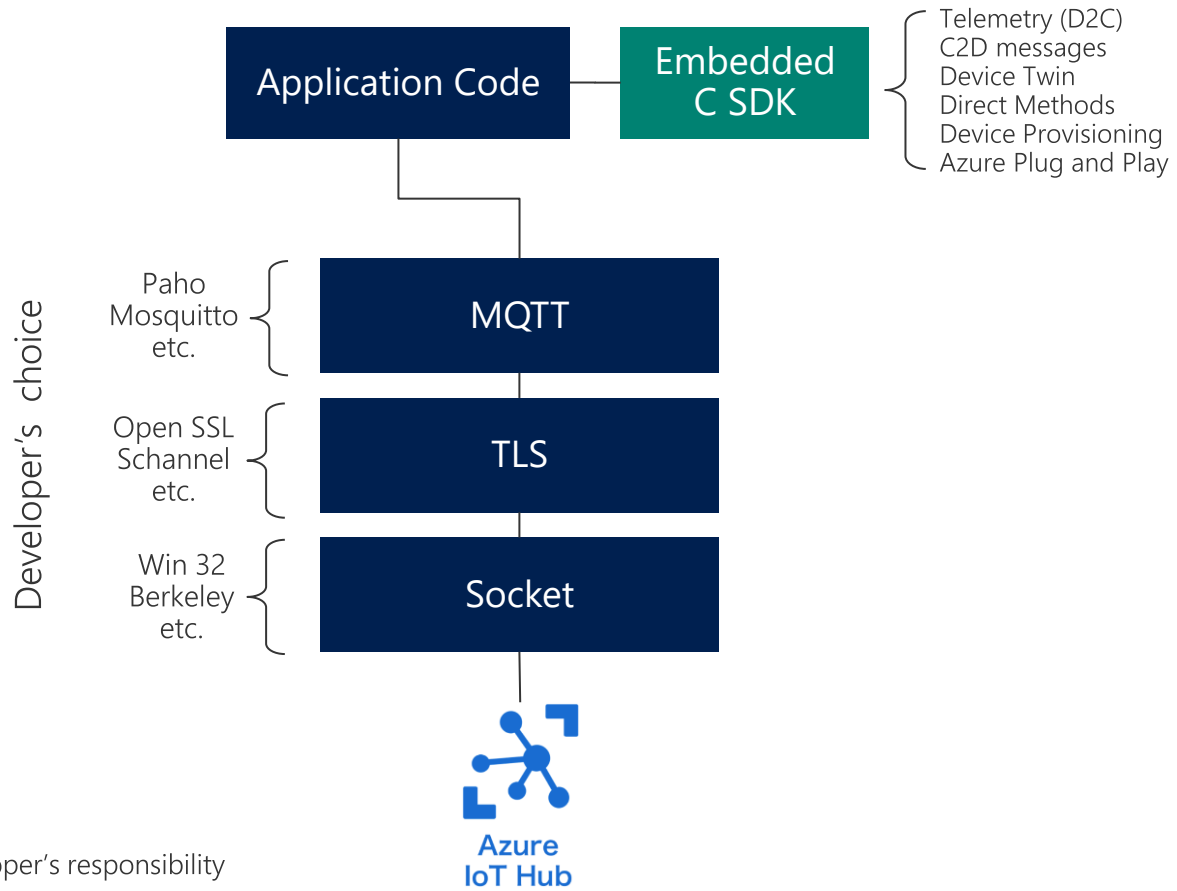
Scenario 1 – C SDK

(Linux – Windows)



Scenario 2 – Embedded C SDK

(Constrained devices)



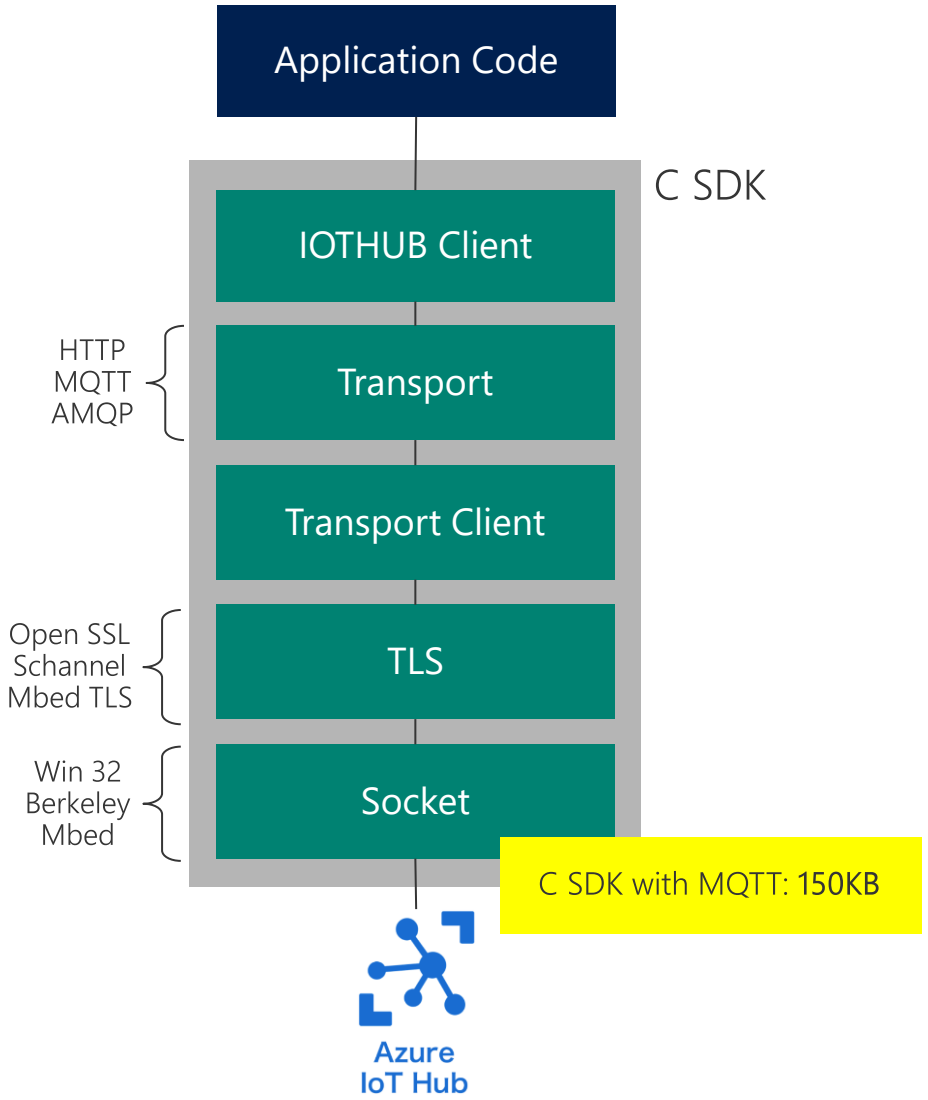
IoT Developer's responsibility

Microsoft's responsibility

Our story for Embedded

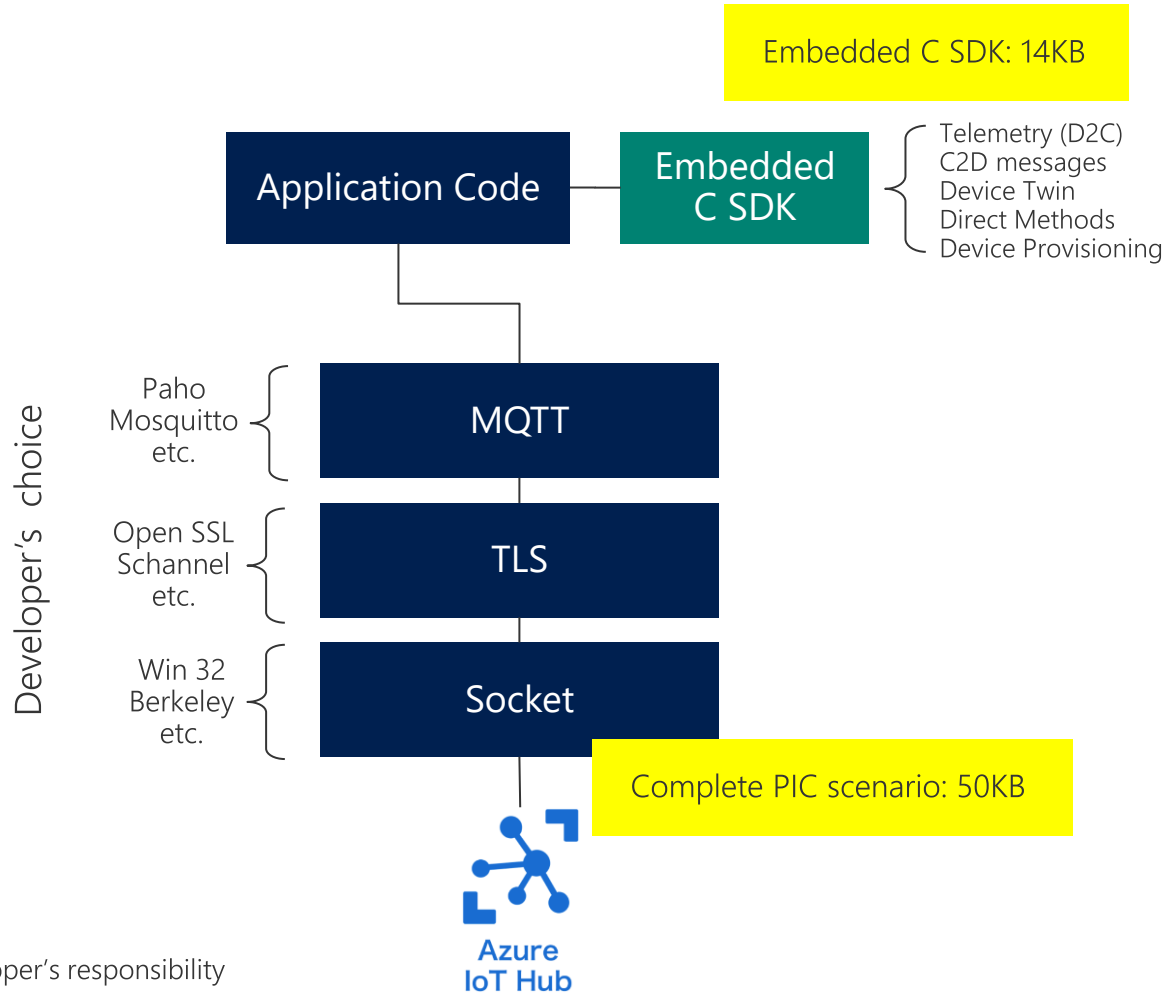
Scenario 1 – C SDK

(Linux – Windows)



Scenario 2 – new Embedded C SDK

(Constrained devices)

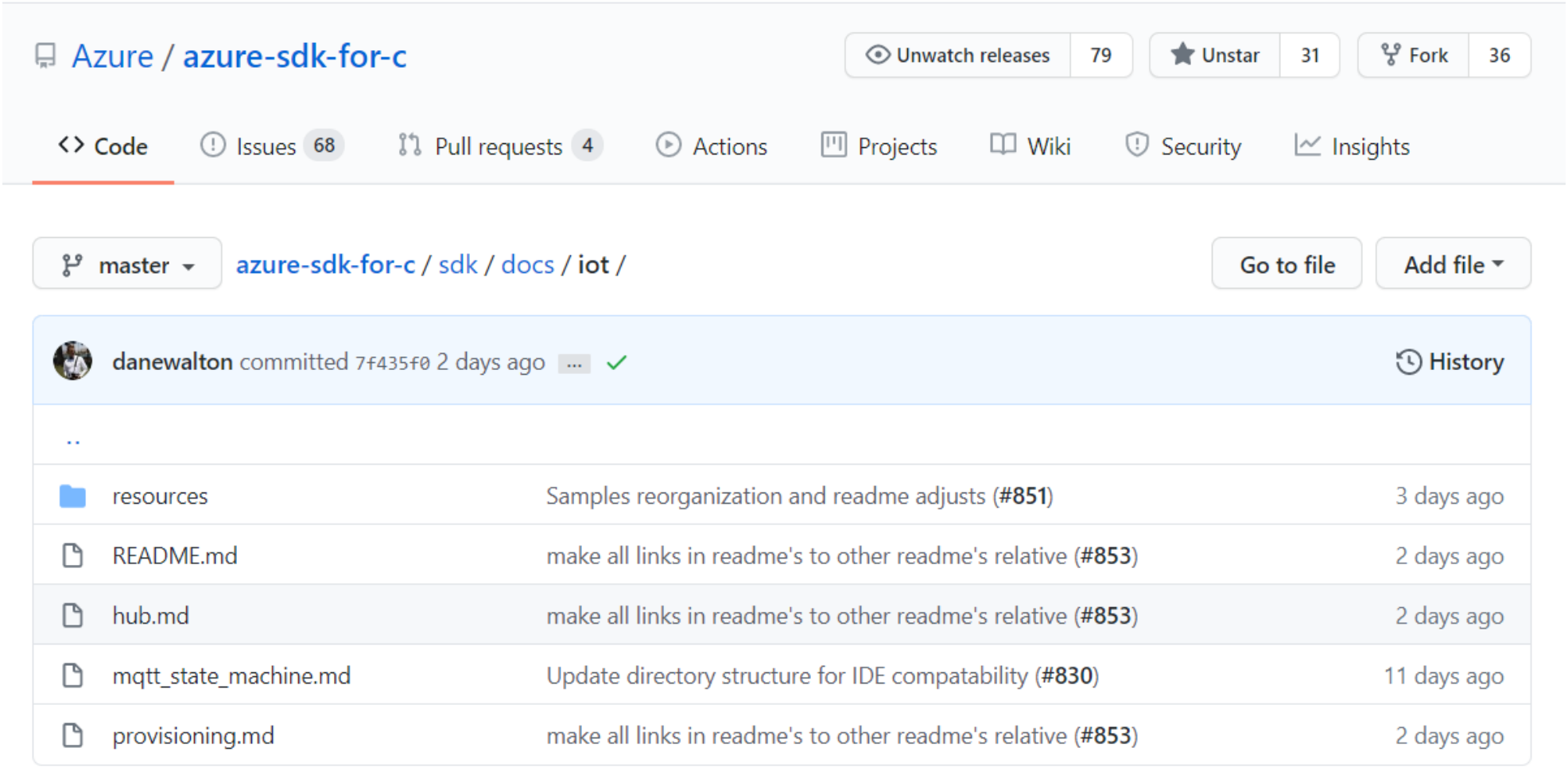


IoT Developer's responsibility

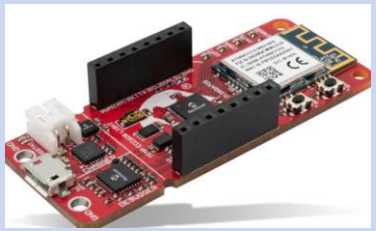
Microsoft's responsibility

GitHub Walkthrough https://aka.ms/embeddedcsdk

Current release: 1.0.0-preview.2 → preview.3 available mid-July



Samples availability Plan

Windows	Linux	Espressif	Microchip	ST
Windows		 ESP8266	 AT SAMD21	 Discovery Kit STM32L5
Sample Available	Sample Available	Sample Available	In progress	In progress

Other partners:

- Qualcomm – 9205 modem
- Renesas – TBD
- NxP – TBD
- Espressif - ESP32

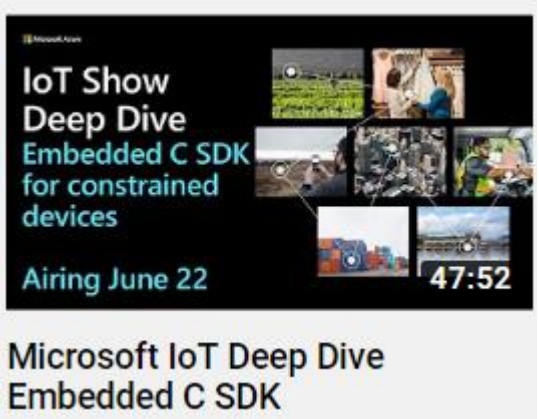
Resources

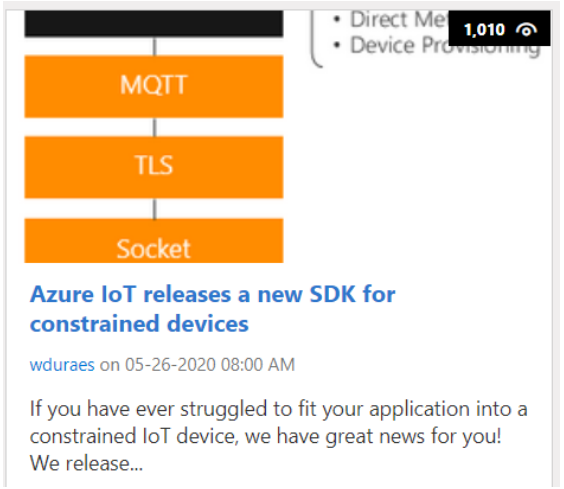
Resources to get started with Embedded C SDK:

GitHub repo https://aka.ms/EmbeddedDSDK (point to the latest official release)

State machine diagrams to learn more about Embedded C SDK: https://github.com/Azure/azure-sdk-for-c/blob/master/sdk/docs/iot/mqtt_state_machine.md

Embedded C SDK Samples: https://github.com/Azure/azure-sdk-for-c/tree/master/sdk/samples/iot





Feature Comparison

Feature	C SDK	Embedded C SDK
Send device to cloud (D2C) messages	[supported]	[supported]
Receive could to device (C2D) messages	[supported]	[supported]
Direct Methods	[supported]	[supported]
Device Twins	[supported]	[supported]
Device Provisioning Services (DPS)	[supported]	[supported]
Protocol	MQTT/AMQP/HTTPS	MQTT
Retry Policies	[supported with automatic retries]	[supported, provides guidelines for retries, but actual retries should be handled by the application]
Devices Multiplexing over single connection	[supported with AMQP]	[not supported]
Plug and Play (PnP)	[supported – public preview]	[not supported yet]*
Upload to Blob	[supported]	[not supported]**

* Feature will be available until the end of 2020
** To be supported in the future by an additional Azure SDK

Customer Scenarios

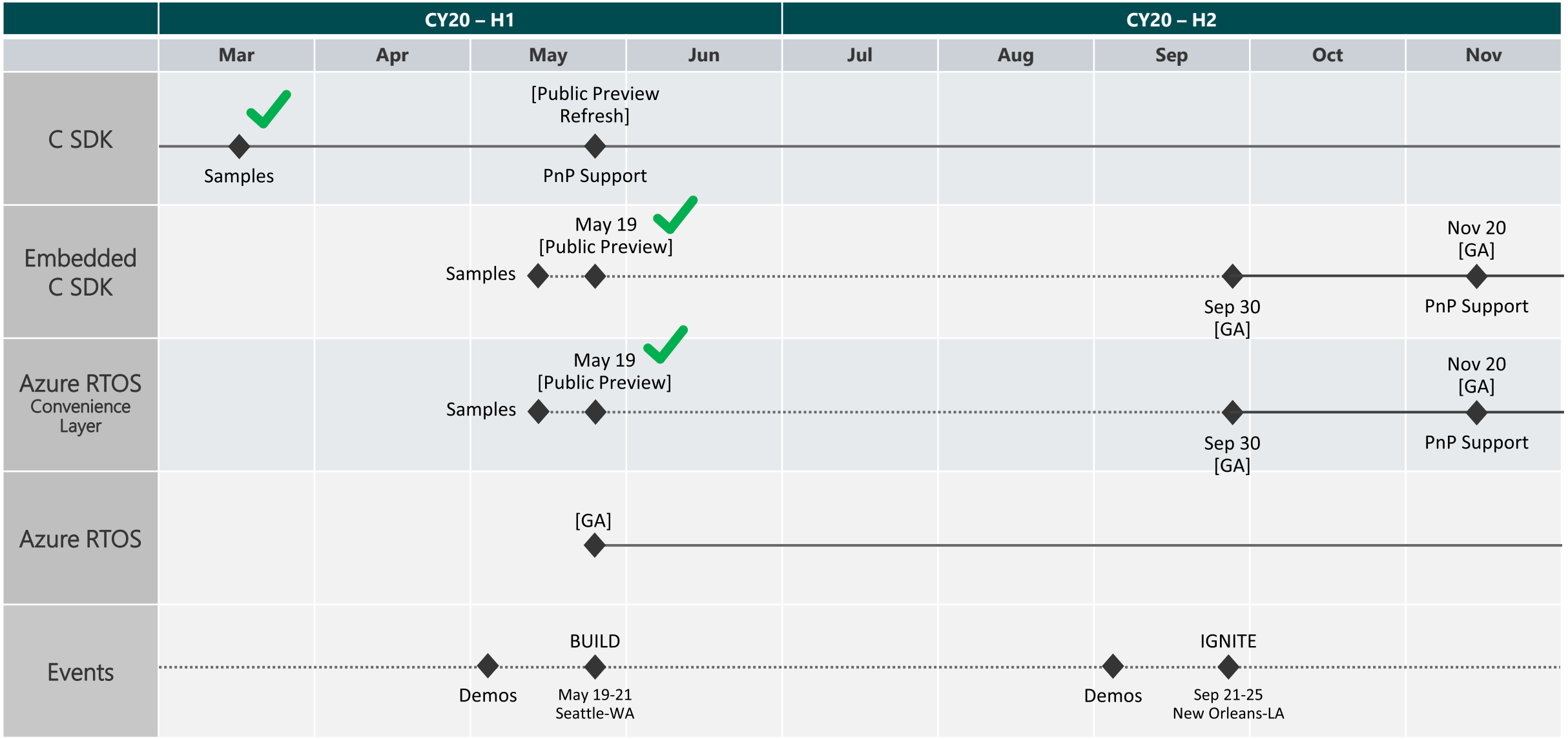
Recommendation for the C SDK varies according to the customer scenarios:

Scenario	Platform	Recommendation
MPU based device	Windows	C SDK
MPU based device	Linux	C SDK
MCU / Constrained device	No RTOS / Bare Metal	Embedded C SDK
MCU / Constrained device	RTOS	Azure RTOS (includes Embedded C SDK)
MCU / Constrained device	Mbed	C SDK
Device Multiplexing	[any]	C SDK

Roadmap

Backlog

1. Free RTOS convenience layer
2. Linux convenience layer
3. C SDK LTS



Internal Roadshow – FAQ

How is the memory allocation for Scenarios 2?
Unlike the C SDK, embedded C SDK won't allocate memory dynamically. The application will be responsible for memory allocation and pass it along with the API calls.

What is the strategy to add new features to the C SDK?
New features will be brought into the C SDK, such as PnP planned for GA this year. We'll continue with this approach until we reach feature parity.

To which platforms can I port the Embedded C SDK?
Just like C SDK, the Embedded C SDK is written in ANSI C (C99) so portable to any supported platform.

Would bug and security fixes on C SDK get the same attention and the ones in the Embedded C SDK?
Absolutely. If there is any bug we'll fix them on both the C SDK and the Embedded C SDK.

Is the code available? Is it open source?
Embedded C SDK is an open source project. Code is available here: https://aka.ms/embeddedcsdk

Why would a customer move from the existing C SDK and move to the other scenarios?
For any scenario which requires a smaller footprint, or scenarios in which leveraging an RTOS would be better for the application.

Will the new SDK support MQTT 5?
Not yet, as the IoT Hub doesn't support it yet.
Some customers can't use MQTT 3.1 due to lack of the negative ACK (it only guarantees transport), and gateway scenarios which leverage device multiplexing. For these customers, AMQP in the existing C SDK will still be a good choice.

What is the incremental value of the new SDK?
The value is in the flexibility it provides, and the smallest SDK footprint

Does Embedded C SDK require threads?
Embedded C SDK does not require threads.
The Azure RTOS in the scenario 3 is the one that would have threads.

Is the Embedded C SDK replacing C SDK?
No, it complements the offer and enables a set of constrained devices.