Internet of Things at the Edge
Azure IoT Edge and Azure Sphere

Sonu Jose
Software Engineer, Valorem

Agenda

Internet of Things with Microsoft Azure
Recap – Azure IoT Hub, Docker
Why Edge Computing
Azure IoT Edge Concepts
IoT Edge Concepts – Modules,
IoT Edge Concepts – Edge Runtime
Module Routing
Introduction - Azure Sphere

Any-things

Connected buildings
Connected cars
Connected Fridges

....

Microsoft's Perspective on IoT

"Empower each and every **device** in this planet to achieve more"

"Massive Opportunity for Developers"

"Microsoft is investing 5 Million dollars in IoT over the next 4 years"

Easier to build secure, scalable solutions from device to cloud

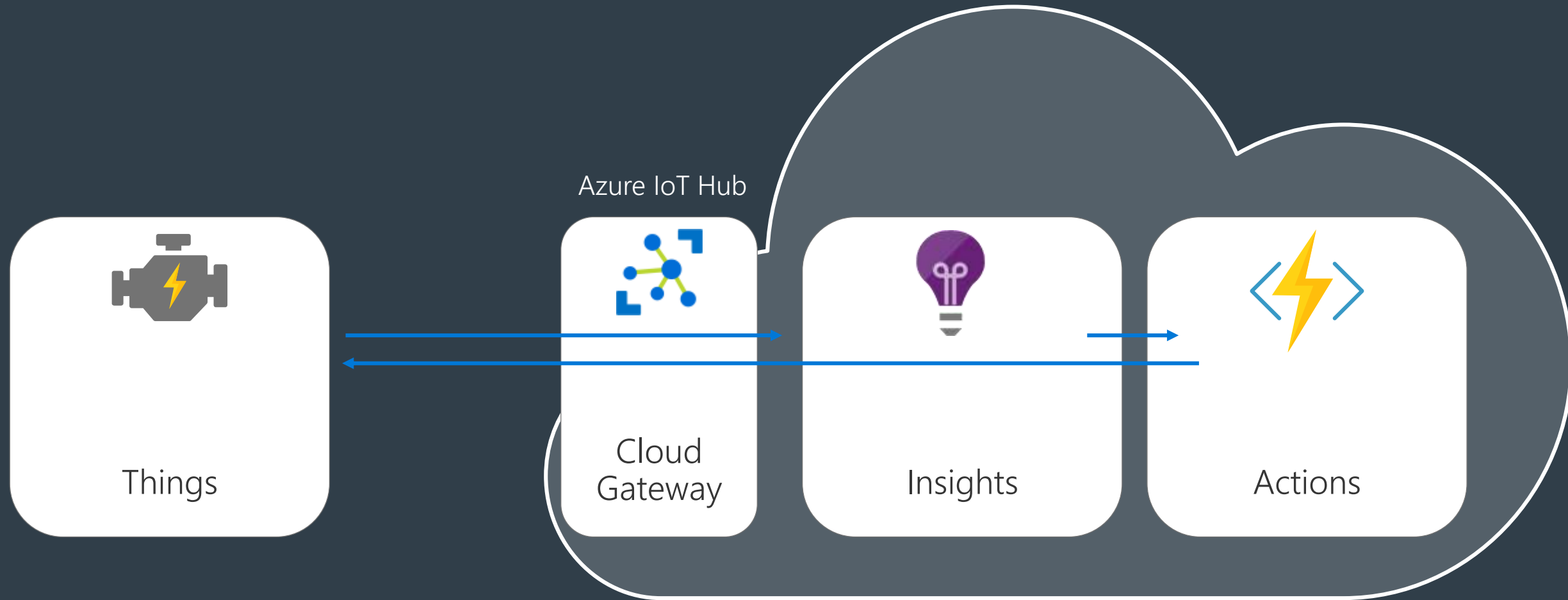# Microsoft Azure Offer for IoT

Azure IoT Hub

Azure IoT Edge

IoT Suite

Azure IoT Central

- Azure IoT Hub Device Provisioning service
- Azure Time Series Insights
- Azure Maps

IoT Pattern

Azure IoT Hub

Things

Cloud Gateway

Insights

Actions

Azure IoT Overview

Microsoft Azure

Management app
Device registry/Management
Monitoring
Send commands

Client SDK

Applications
- Companion apps
- Dashboards
- Controllers
- ...

Device

Client SDK

AMQP, MQTT, HTTPS

IoT Hub

Device

Field
Gateway

AMQP

IoT Edge

Device

Management app
Device registry/Management
Monitoring
Send commands

Client SDK

Azure IoT Hub Overview

Device

Field GW / Cloud GW

IoT Hub

- D2C send endpoint
- C2D queue endpoint
- Twin endpoint
- Methods endpoint
- Device id
- Device ...
- Device...
- Device ...
- IoT Hub management

- D2C receive endpoint
- Custom endpoints
- C2D send and feedback endpoints
- Twins endpoint
- Device identity management

Event processing (hot path)

Event processing (hot and cold path)

Device management, device business logic, Connectivity monitoring

Device provisioning and authorization

IoT in the Cloud and on the Edge
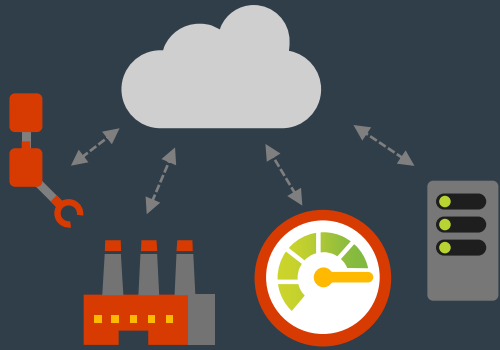
IoT in the Cloud

Remote monitoring and management

Merging remote data from multiple IoT devices

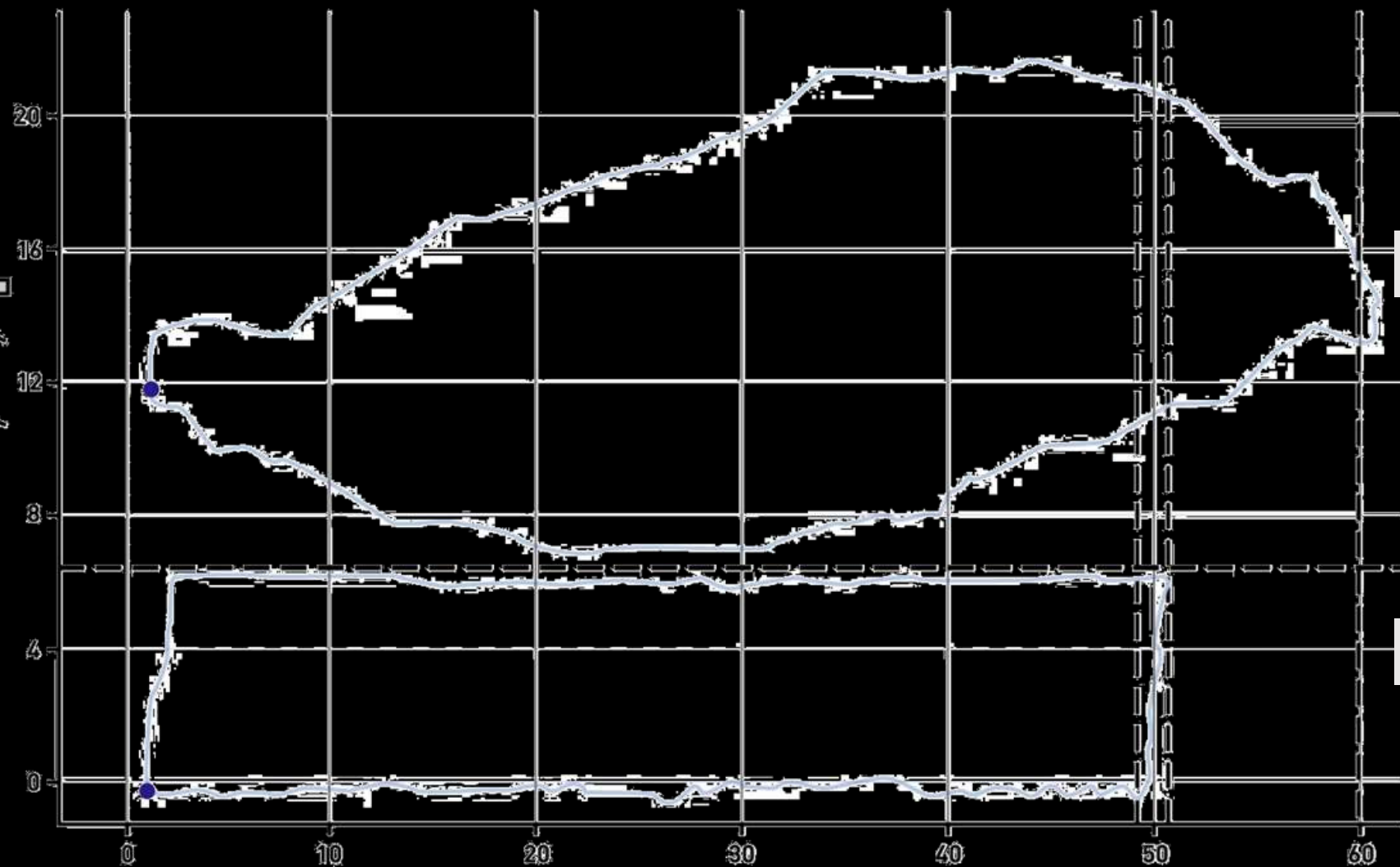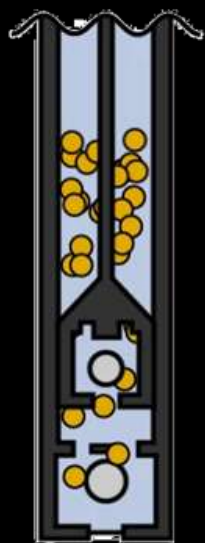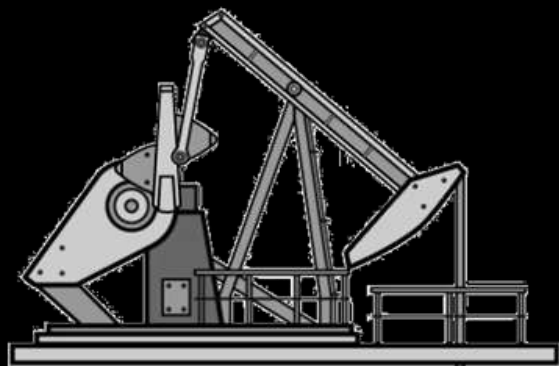Infinite compute and storage to train machine learning and other advanced AI tools

IoT on the Edge

Low latency tight control loops require near real-time response

Protocol translation & data normalization

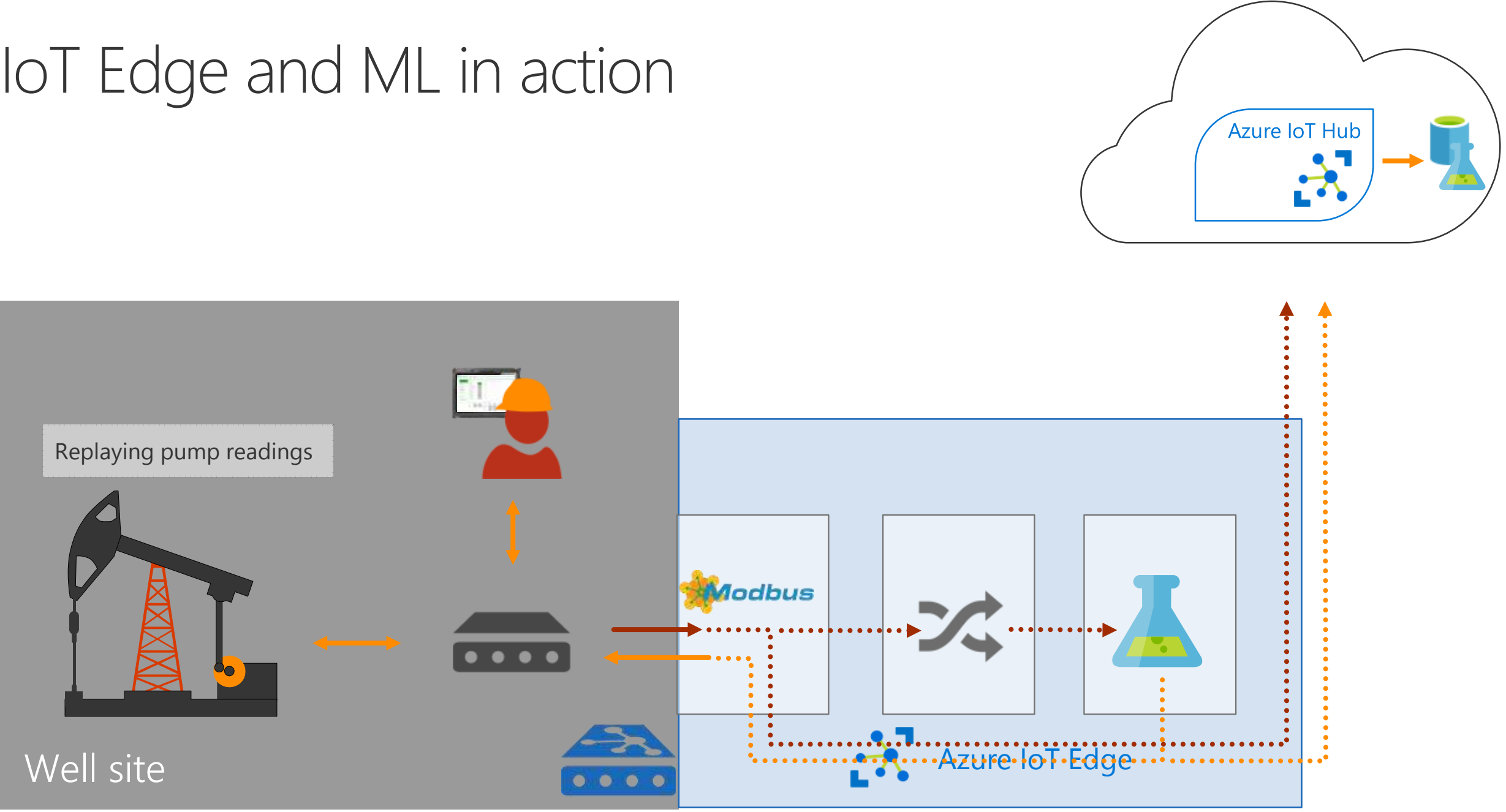Privacy of data and protection of IP
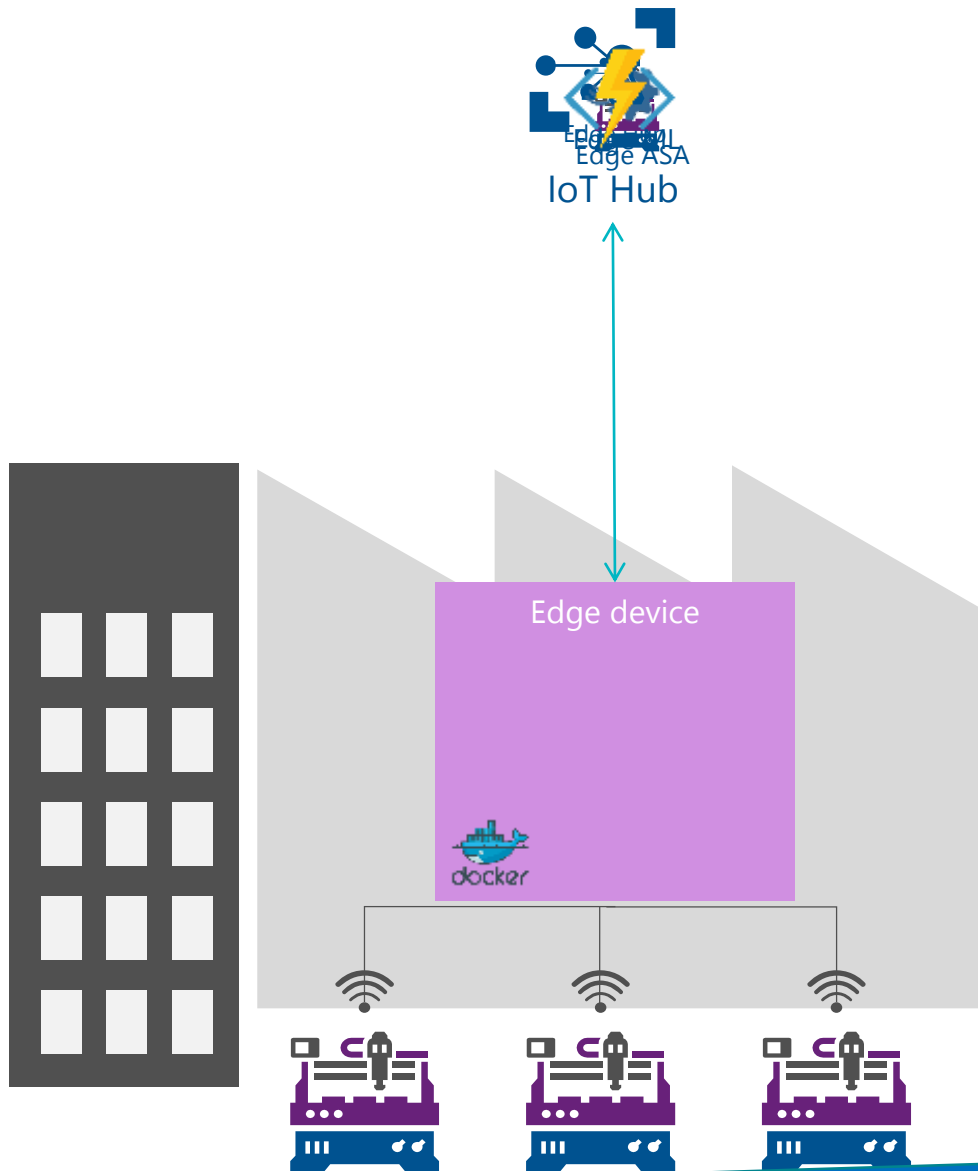
Azure IoT Edge

A typical scenario

Surface

Downhole

Pump position

IoT Edge and ML in action

Azure IoT Hub

Well site

Replaying pump readings

Modbus

Azure IoT Edge

Overview

Edge runtime using Docker for distribution and management

Runs a local version of IoT Hub

Deploy a device sensor module/container

Deploy edge stream analytics module

Deploy anomaly detection ML module

Deploy machine shutdown Azure Function module

Deploy an updated Azure Function module

Edge ML
Edge ASA
IoT Hub

Edge device

docker

Quick Recap – Containers and Docker

Containers wraps an application's software into an invisible box with everything the application needs to run. That includes the operating system, application code, runtime, system tools, system libraries, and etc.

Docker is an open-source project based on Linux containers. It uses Linux Kernel features like namespaces and control groups to create containers on top of an operating system.
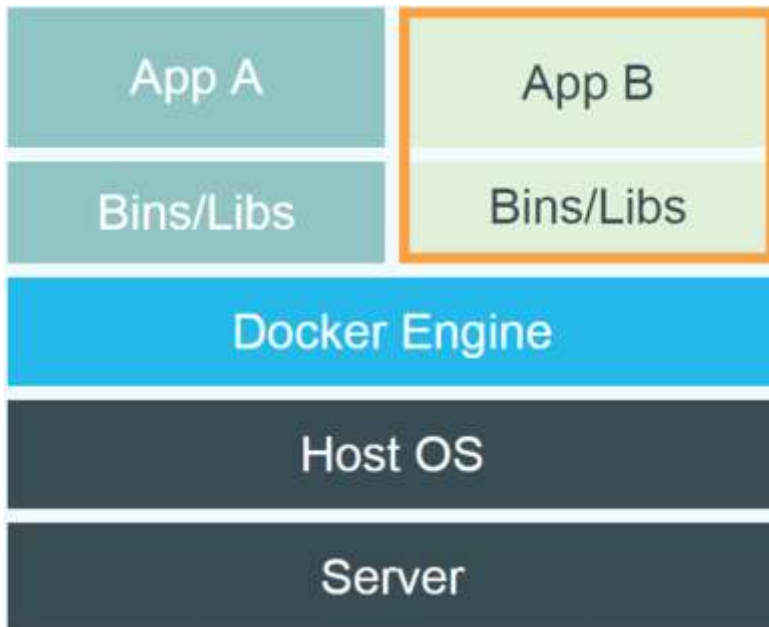
Images

How you store your application

Docker File

where you write the instructions to build a Docker image

Container

How you run your application

App A	App B
Bins/Libs	Bins/Libs

Docker Engine

Host OS

Server

You can build Docker images that hold your applications
You can create Docker containers from those Docker images to run your applications.
You can share those Docker images via Docker Hub or your own registry

Azure IoT Edge

- Deploy **Azure services** to IoT Edge devices

- Deploy your **own code in language of your choice**

- **Manage** IoT Edge and downstream devices

- Do all of this **securely**, in a **scalable fashion** from the Azure IoT Hub

Azure IoT Edge
Design Principles

Secure

Provides a secure connection to the Azure IoT Edge, update software/firmware/configuration remotely, collect state and telemetry and monitor security of the device

Cloud managed

Enables rich management of Azure IoT Edge from Azure; provides a complete solution instead of just an SDK

Cross-platform

Enables Azure IoT Edge to target the most popular edge operating systems, such as Windows and Linux
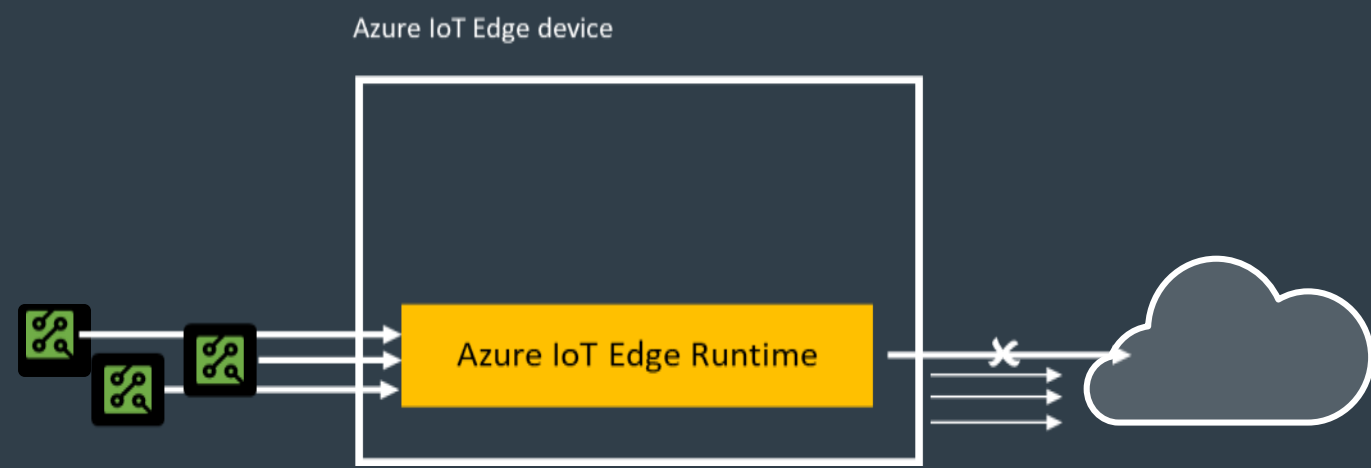
Portable

Enables dev/test of edge workloads in the cloud with later deployment to the edge as part of a continuous integration/continuous deployment pipeline

Extensible

Enables seamless deployment of advanced capabilities such as AI from Microsoft, and any third party, today and tomorrow

Concepts – Edge Runtime

Azure IoT Edge device

Azure IoT Edge Runtime

- Installs and updates workloads on the device.

- Maintains Azure IoT Edge security standards on the device.

- Ensures that IoT Edge modules are always running.

- Reports module health to the cloud for remote monitoring.

- Facilitates communication between downstream leaf devices and the IoT Edge device.

- Facilitates communication between modules on the IoT Edge device.

- Facilitates communication between the IoT Edge device and the cloud

Concept – Module
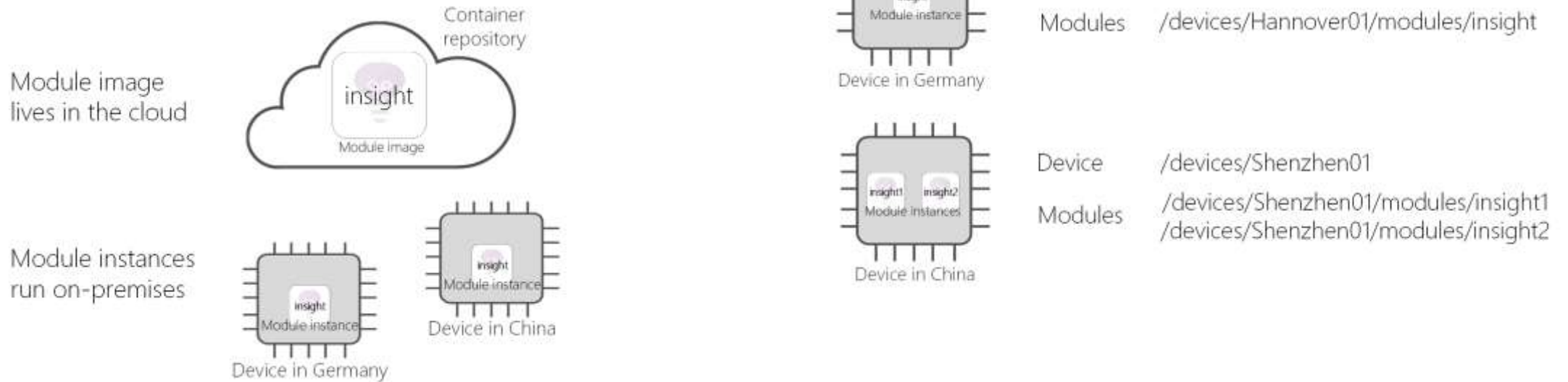
- A **module image** is a package containing the software that defines a module.

- A **module instance** is the specific unit of computation running the module image on an IoT Edge device. The module instance is started by the IoT Edge runtime.

- A **module identity** is a piece of information (including security credentials) stored in IoT Hub, that is associated to each module instance.

- A **module twin** is a JSON document stored in IoT Hub, that contains state information for a module instance, including metadata, configurations, and conditions.

- SDKs to develop custom modules in multiple languages (C#, C, Python, Java, Node.JS)

Concepts – Modules

Edge Runtime manages modules
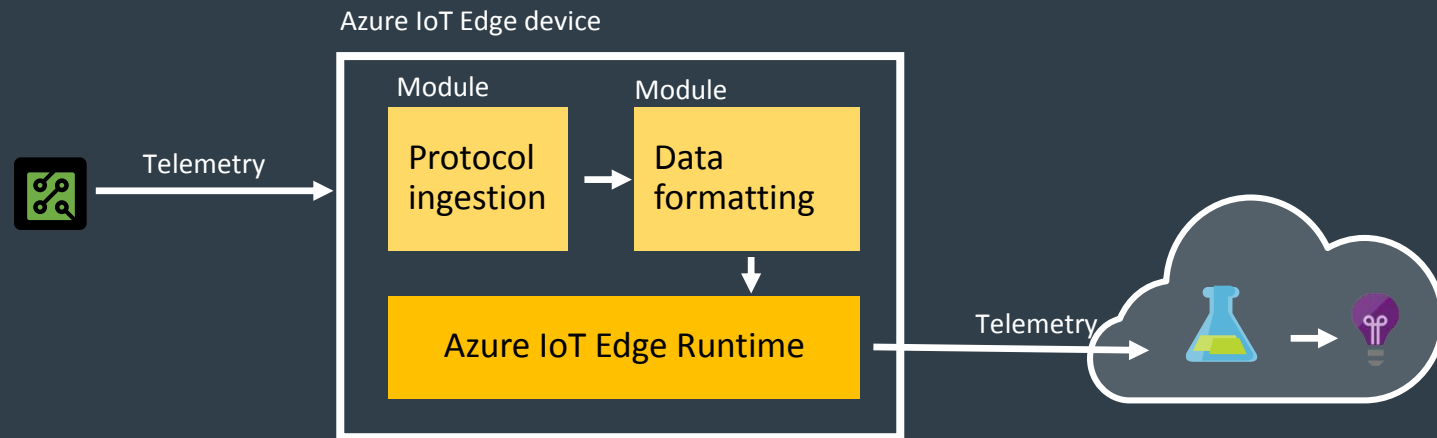
Modules add capabilities to the runtime

Each module performs an action

Chain of modules can be thought of as a data processing pipeline, solving an end to end scenario

Modules are Docker containers

Custom modules can be written in the language of your choice

Scenario: Find insights in the cloud from telemetry sent by a device that does not speak an internet ready protocol.

Azure IoT Edge device

Module — Protocol ingestion

Module — Data formatting

Telemetry

Azure IoT Edge Runtime

Telemetry

Concept - Routing

FROM \<source> WHERE \<condition> INTO \<sink>

Sources – source of messages
    /messages/modules/{mid}/outputs/{out1}

Condition – expression on messages properties/body
    sensorType = "temp" and alert = true

Sinks – destination for messages (endpoints)
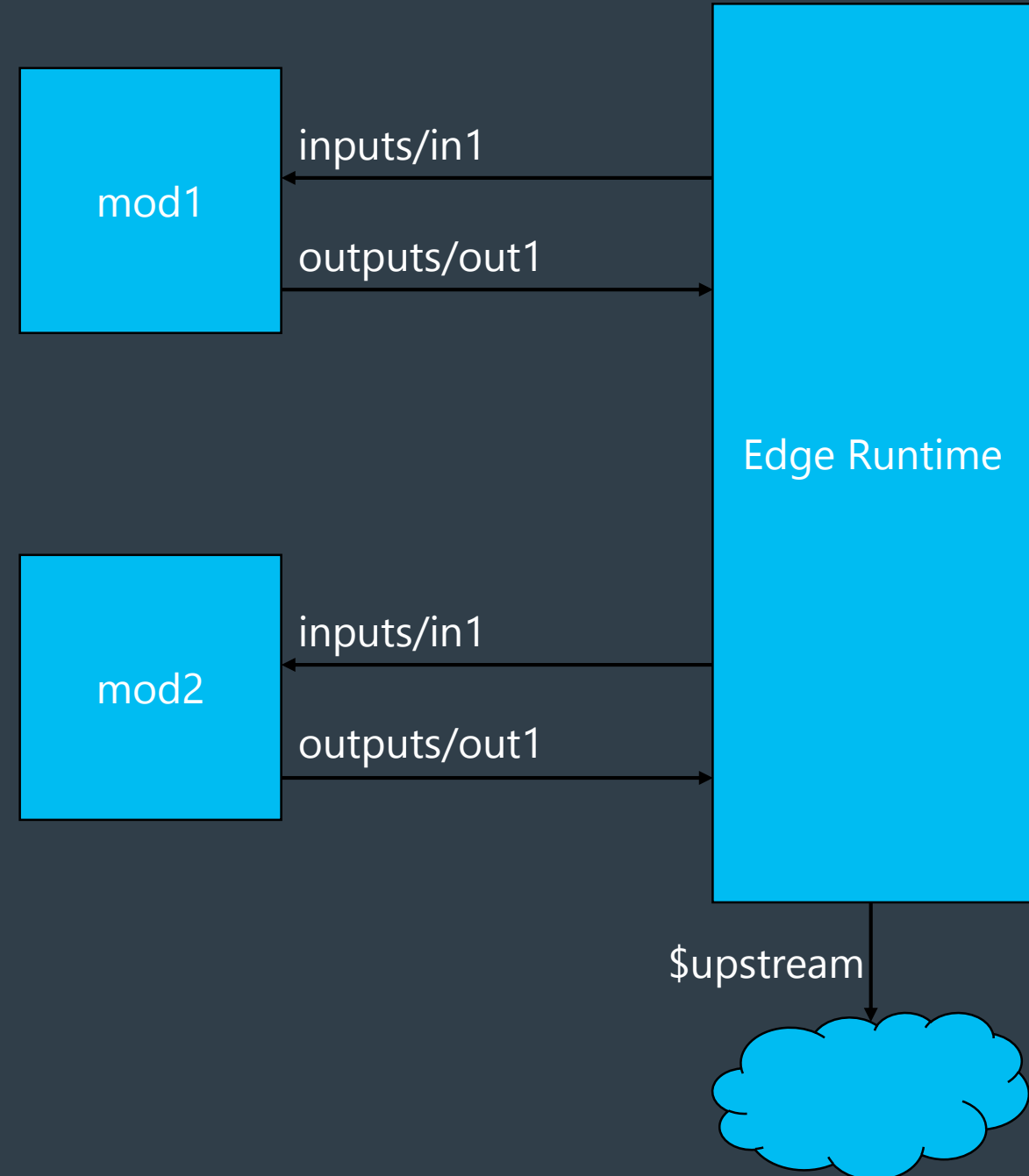 $upstream
    brokeredEndpoint("/modules/{mid}/inputs/{in1}")
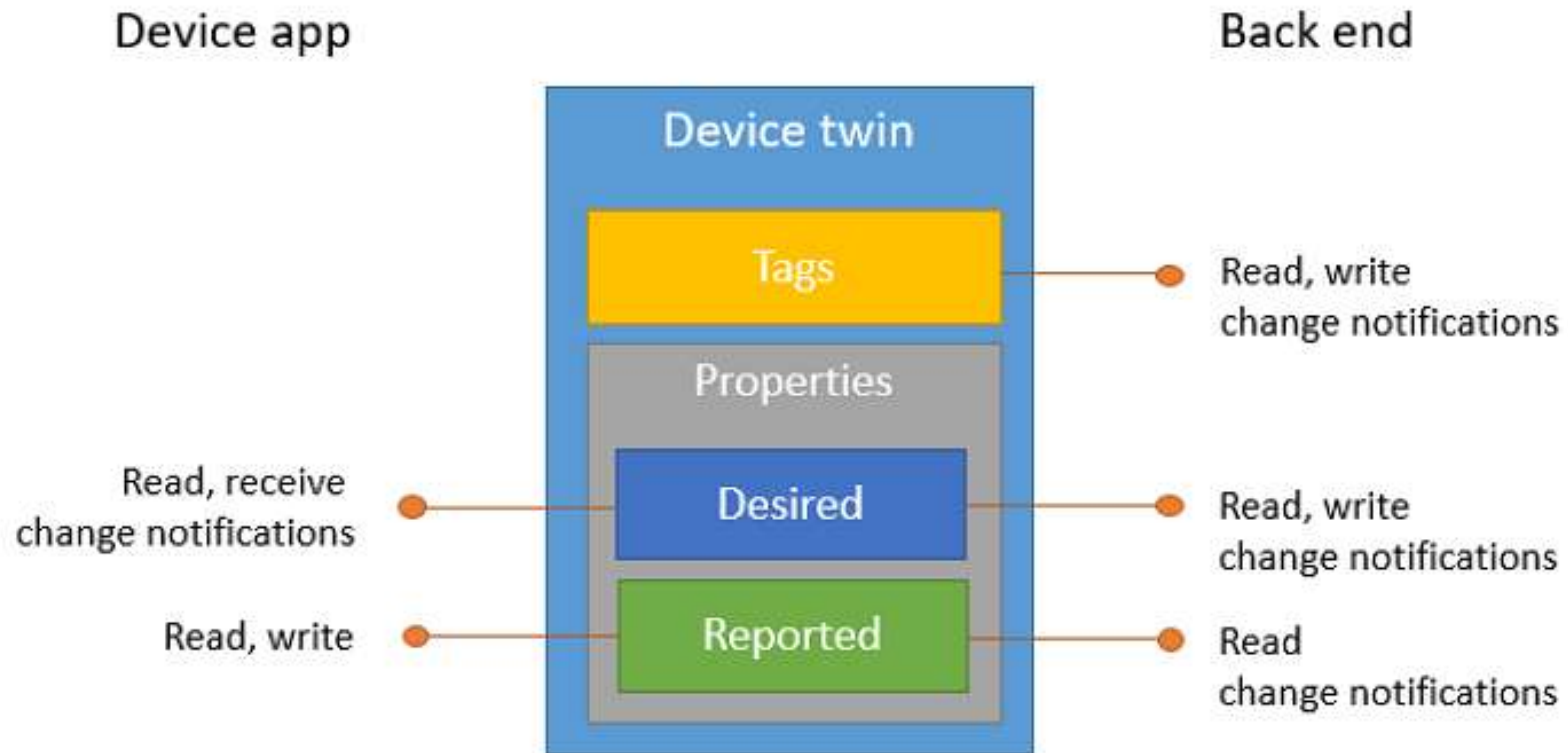
For example:
    FROM /messages/modules/mod1/outputs/*
    WHERE sensorType = "temp"
    INTO brokeredEndpoint("/modules/mod2/inputs/in1")
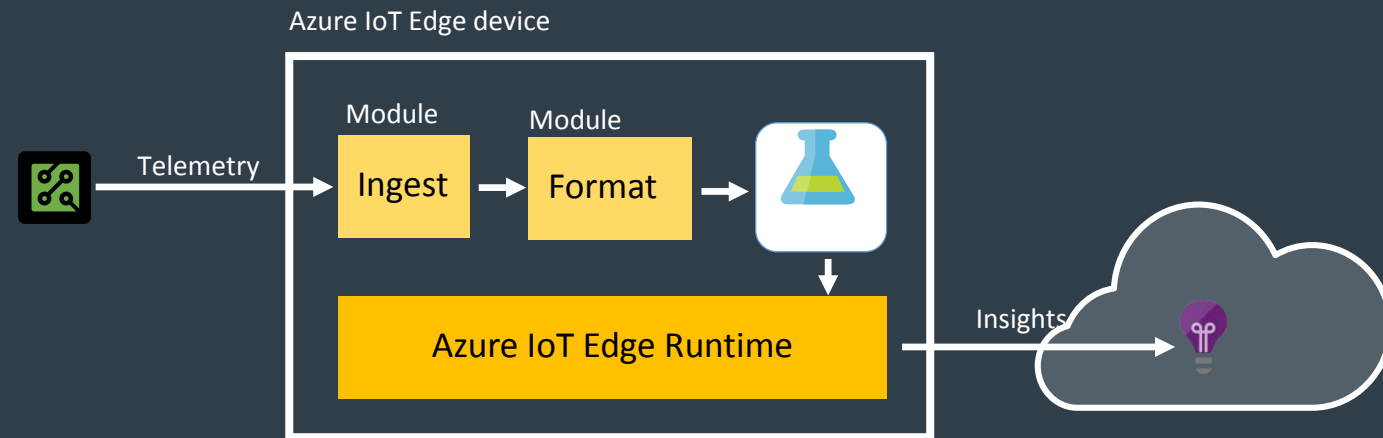
Query Language

Device Management

Concepts – Cloud off load

Modular architecture for cloud offload
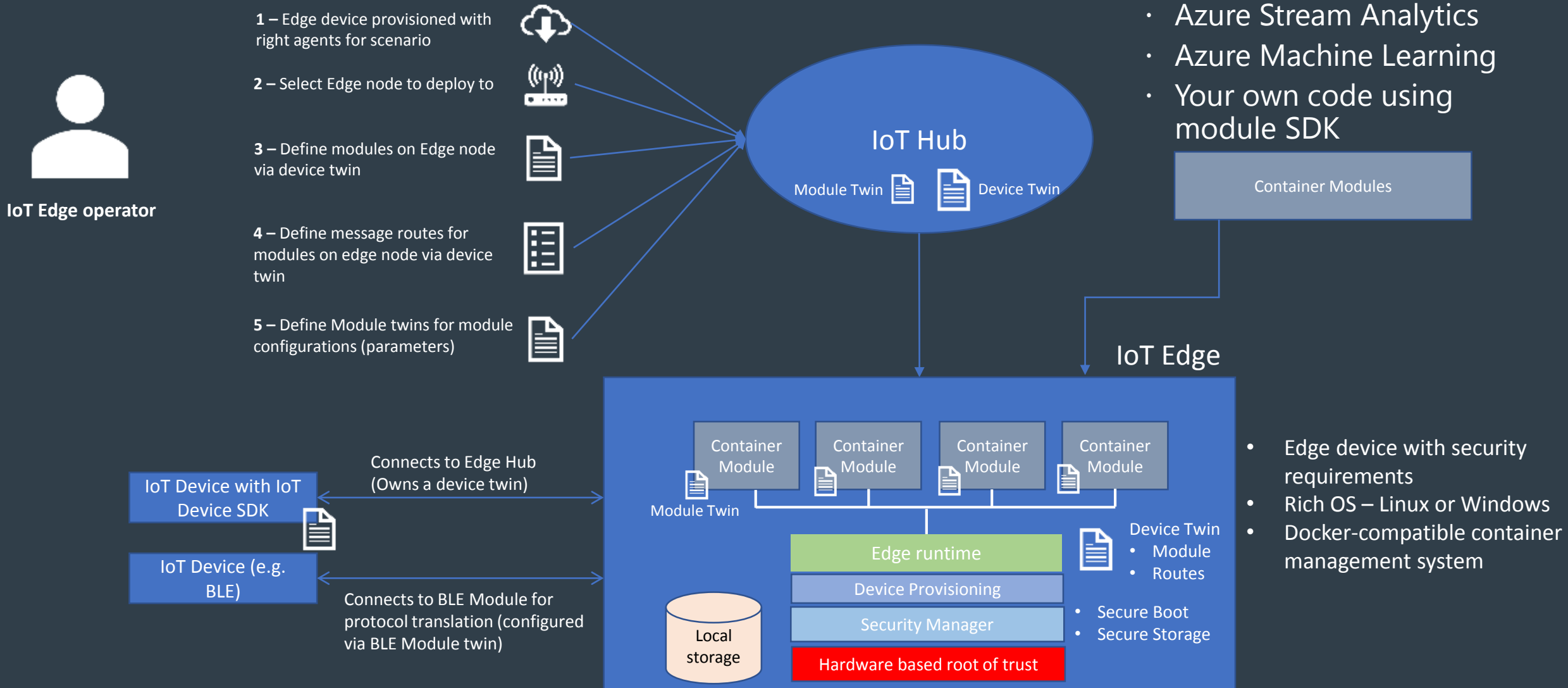
Azure services provide AI modules provide true edge analytics

Ecosystem for 3rd party edge services

Scenario: Find insights locally from telemetry data and only send insights to the cloud.

Azure IoT Edge device

Module

Module

Telemetry

Ingest

Format

Azure IoT Edge Runtime

Insights

IoT Edge in action

IoT Edge operator

- Container based workloads
- AI Services
- Azure Functions
- Azure Stream Analytics
- Azure Machine Learning
- Your own code using module SDK

1 – Edge device provisioned with right agents for scenario

2 – Select Edge node to deploy to

3 – Define modules on Edge node via device twin

4 – Define message routes for modules on edge node via device twin

5 – Define Module twins for module configurations (parameters)

IoT Hub

Module Twin   Device Twin

Container Modules

IoT Edge

IoT Device with IoT Device SDK

Connects to Edge Hub (Owns a device twin)

IoT Device (e.g. BLE)

Connects to BLE Module for protocol translation (configured via BLE Module twin)

Container Module

Container Module

Container Module

Container Module

Module Twin

Edge runtime

Device Provisioning

Security Manager

Hardware based root of trust

Local storage

Device Twin
- Module
- Routes

- Secure Boot
- Secure Storage

- Edge device with security requirements
- Rich OS – Linux or Windows
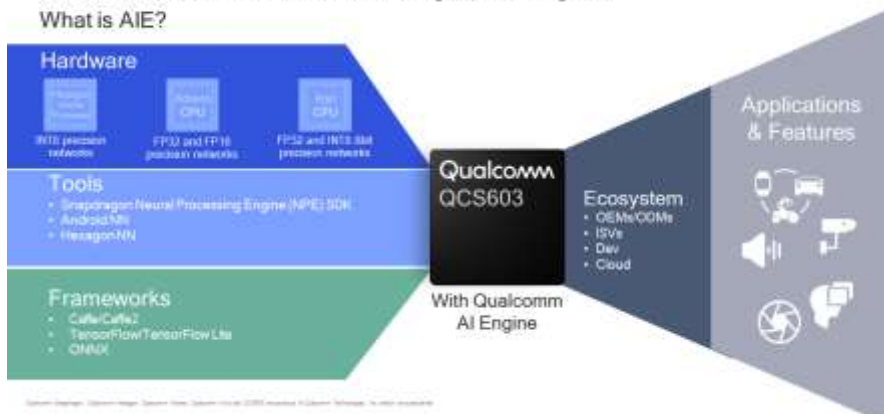- Docker-compatible container management system

Lets see some code…

The vision AI developer kit

Qualcomm® Vision Intelligence Platform with Qualcomm Artificial Intelligence Engine
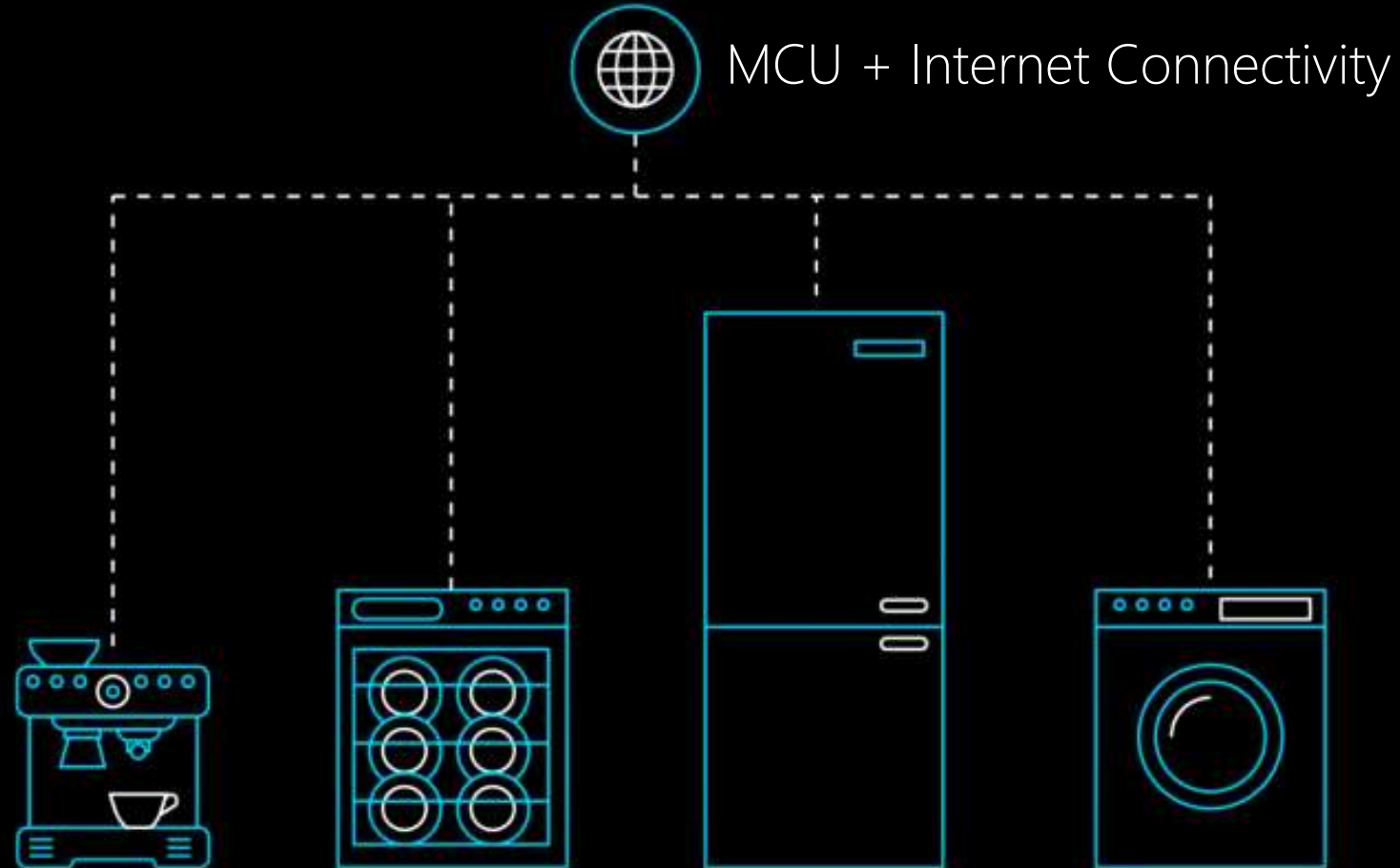What is AIE?

- Qualcomm® Technologies, Inc. and Microsoft collaboration

- Run AI models on the edge without additional computers or web connection or leverage the cloud

- Create, deploy and manage all your models in the cloud and the edge with Azure ML and Azure IoT Edge

Azure Sphere

Prepare for the 2nd wave of digital transformation...

MCU + Internet Connectivity
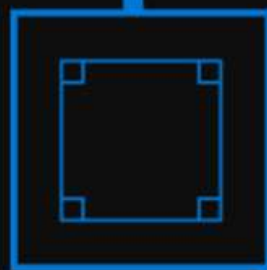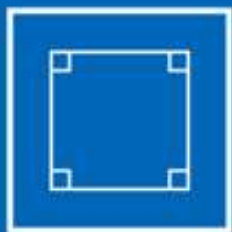
Meet the newest class of microcontrollers, plus the OS and cloud technology that secure them

Secured MCU

Secured from the silicon up. Our new crossover class of MCUs combines for the first time both real-time and application processors with built-in Microsoft security technology and connectivity.

Secured OS

An OS purpose built for security and agility to create a trustworthy platform for new IoT experiences. Our secured OS builds security innovations pioneered in Windows into a HLOS small enough for MCUs.

Cloud Security

Protect devices with a cloud built for IoT security. The Azure Sphere Security Service renews device security, identifies emerging threats, and brokers trust between device, cloud, and other endpoints.

The seven Properties of Highly Secure Device: The new standard for securing MCU powered IoT experiences

Hardware Root of Trust

Defense in Depth

Small Trusted Computing Base

Dynamic Compartments

Certificate-Based Authentication

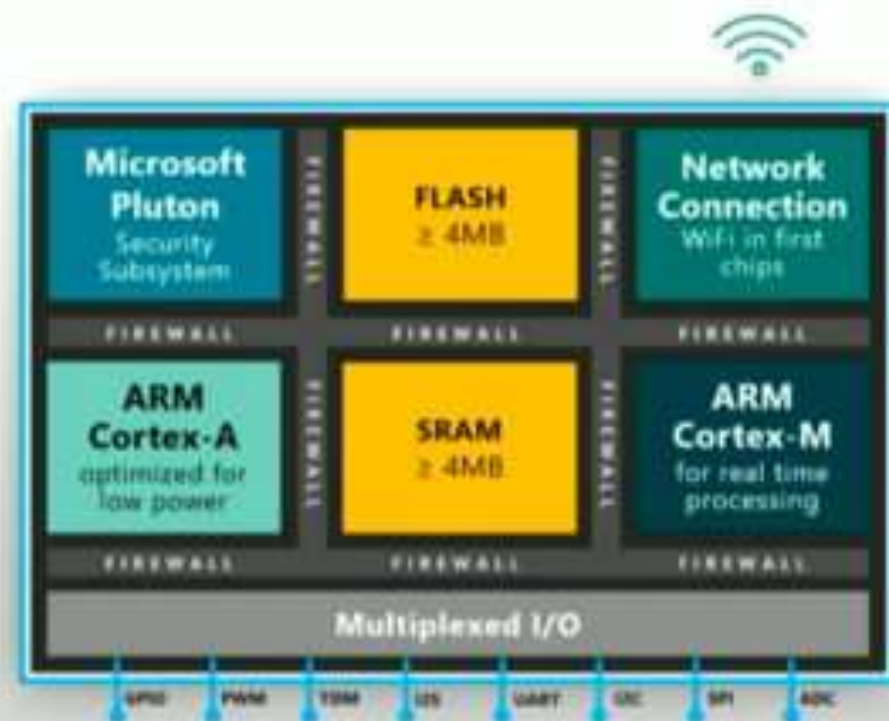**Failure Reporting**

Renewable Security

https://aka.ms/7properties

Engineering security at the root

Connected with built-in networking

Secured with built-in silicon technology pioneered by Xbox

Crossover capabilities enabling ambient intelligence in an MCU for first time

Microsoft is Fundamentally committed for making world more secure

Security

Azure Sphere provides security that starts in the hardware and extends to the cloud, delivering holistic security that protects, detects, and responds to threats – so they're always prepared.

Productivity

Azure Sphere's software delivery model and Visual Studio development tools deliver productivity and dramatically optimize the process of developing and maintaining apps on their devices.

Opportunity

The real magic begins when device manufacturers start imagining the possibilities that open with Azure Sphere. The built-in connectivity and additional headroom included in Azure Sphere certified MCUs changes everything. It's been incredible to watch them design next generation experiences with Azure Sphere.

Meet Azure Sphere

Azure Sphere MT3620 Development Kit

MT3620 Development Board for Azure Sphere is a development board with built-in Wi-Fi communication (IEEE 802.11 a/b/g/n compliant) and security features for internet-connected devices.
Each development board includes the entire Azure Sphere solution:

• **An Azure Sphere compliant MCU**, the MT3620 from MediaTek, which combines for the first time both real-time and application processors with built-in Microsoft security technology and connectivity.
• **The Azure Sphere OS**. A highly-secured OS from Microsoft creates a trustworthy, defense in depth platform for new IoT experiences.
• **The Azure Sphere Security Service**. A turnkey security service that guards every Azure Sphere device; renewing security, identifying emerging threats, and brokering trust between device, cloud, and other endpoints.

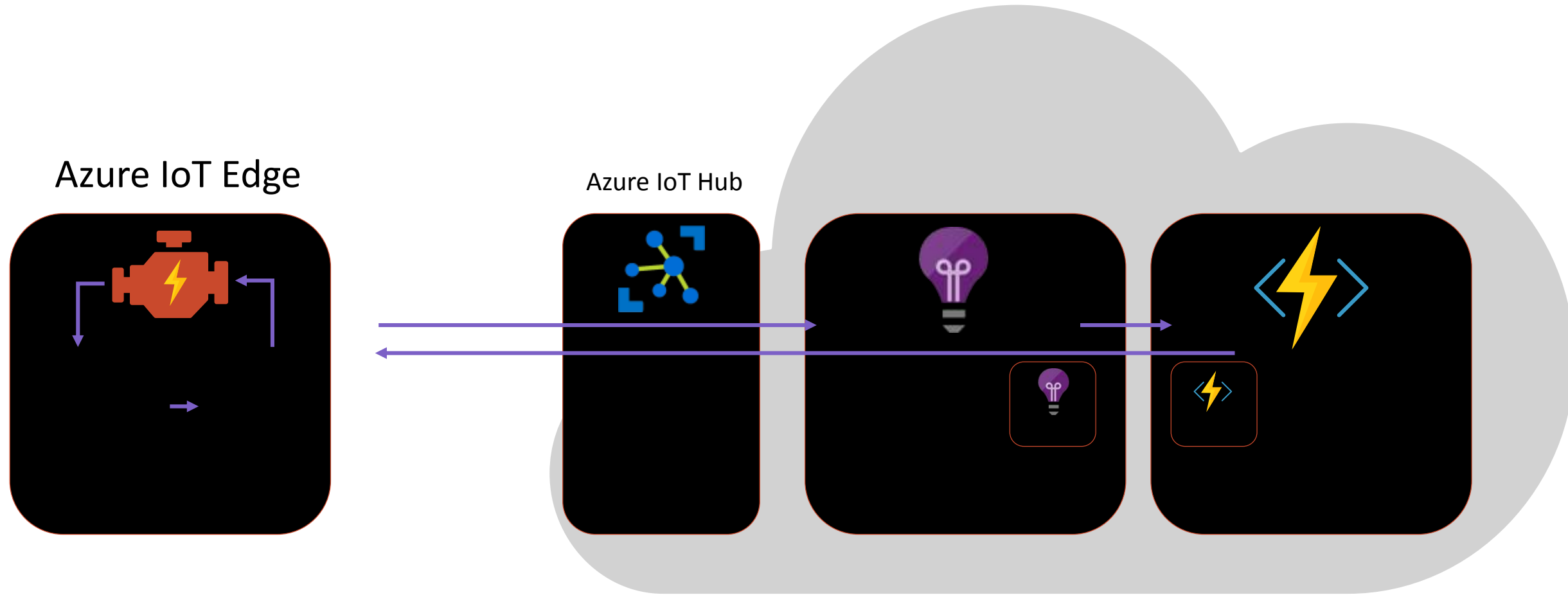The MT3620 development board enables software and hardware development in the Azure Sphere ecosystem.
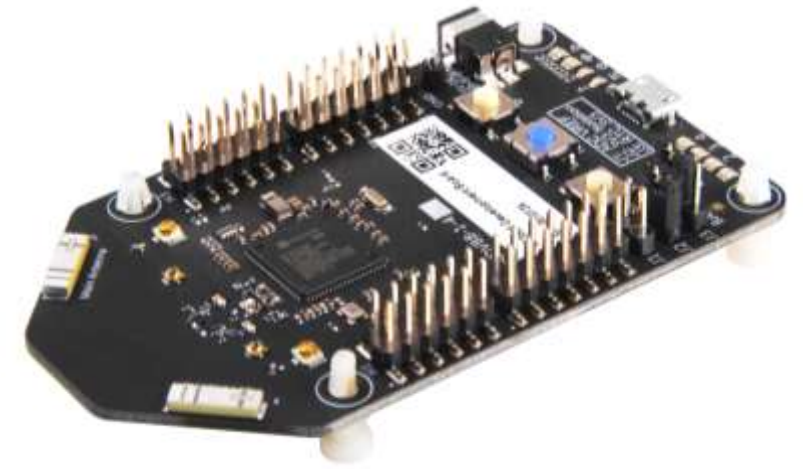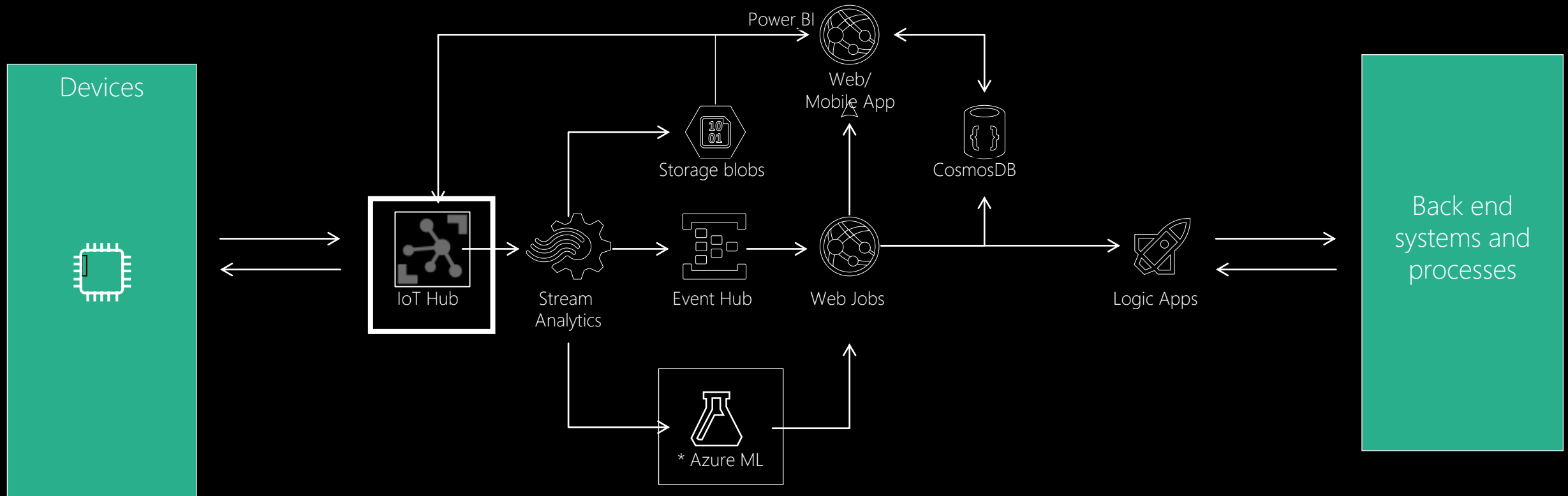
Thank you

Sonu Jose

Software Engineer, Valorem

IoT Pattern + Edge

Azure IoT Edge

Azure IoT Hub

Example of an Azure IoT Solution

Edge Runtime provides fundamental services

Security

Multiplexing

Store and forward (Offline)

Management for devices otherwise isolated from internet