Microsoft

Azure IoT – The Intelligent IoT Edge-DGT0001

Digital Transformation and the Internet of Things

Francis Dogbey
DSA

Agenda

Azure IoT Edge

- Concepts
- Architecture
- Gateway Design Patterns
- Demos
    - Linux
    - Windows

Defining IoT

Things

Insights

Action

IoT projects are still complex

Security
is a **challenge**

Time-consuming
to get started

Incompatible with
existing infrastructure

Challenging
to scale

IOT & INTELLIGENT INFRASTRUCTURE

Remote Monitoring

Predictive Maintenance

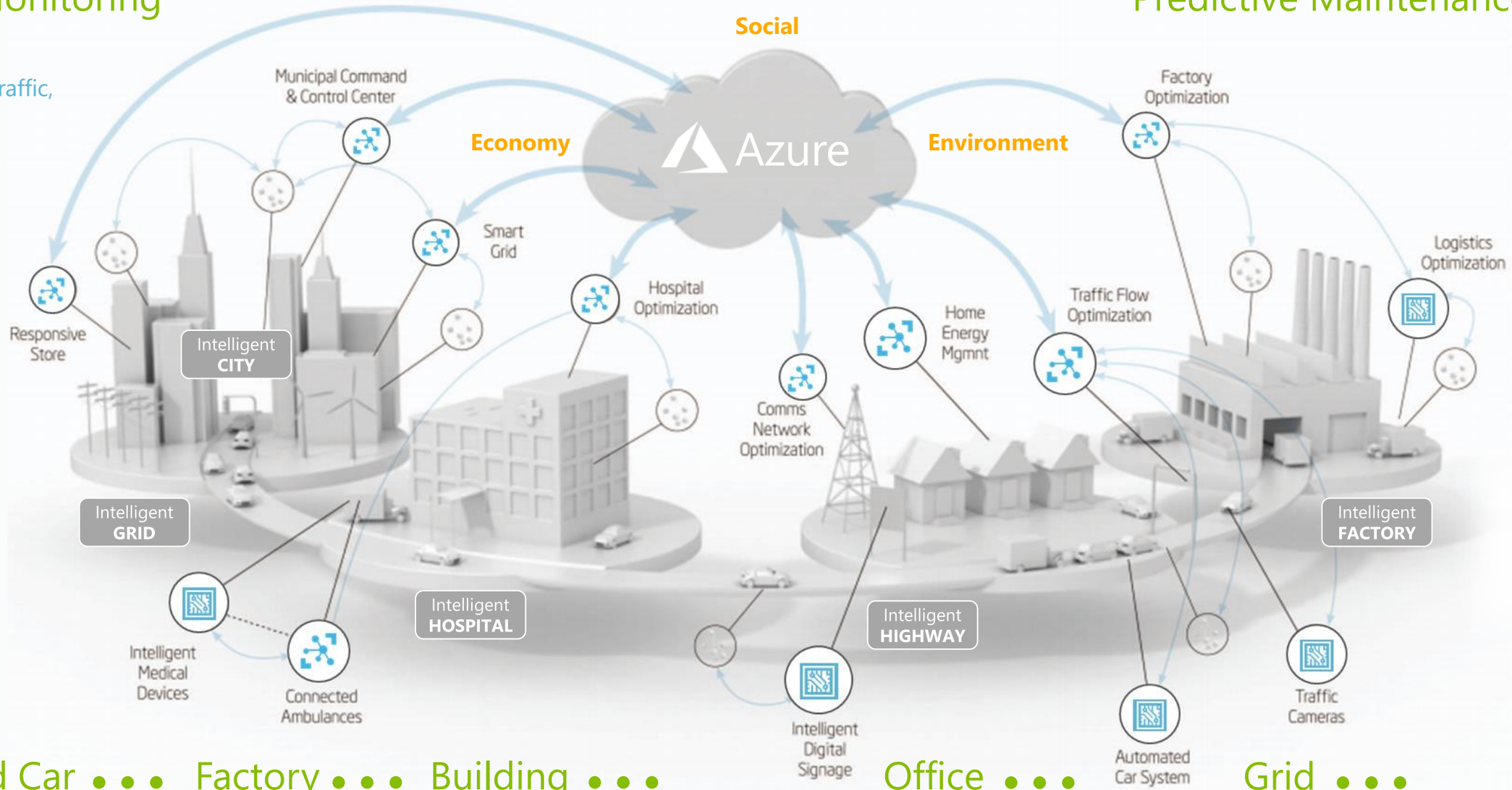
Transportation, Traffic, Parking, Vehicles

Housing, Buildings

Energy, Water & Utilities

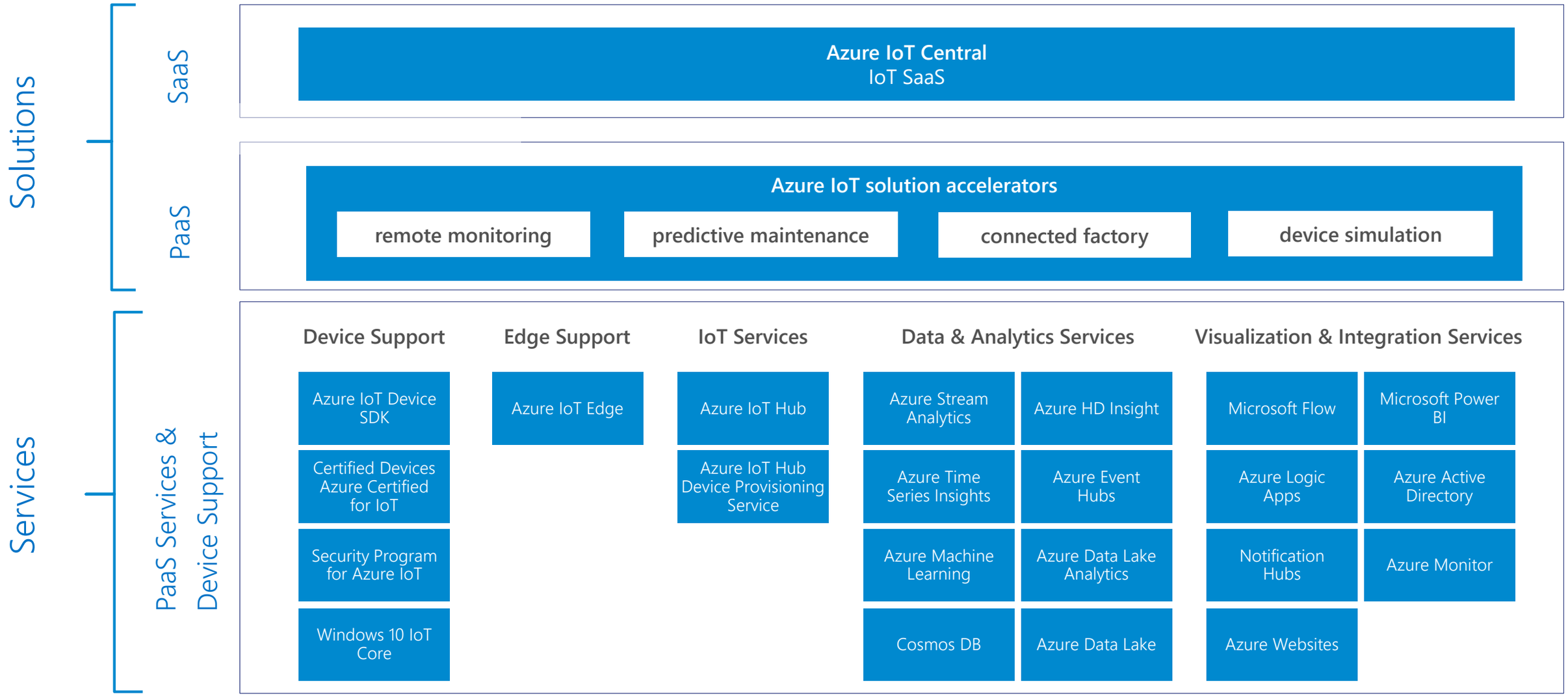
Citizen Engagement

Public Safety & Security



Connected Car • • • Factory • • • Building • • • Office • • • Grid • • •

Comprehensive Set of Capabilities for IoT Solutions



Azure IoT Hero Portfolio

Microsoft's vision is to democratize IoT by allowing everyone to access the benefits of IoT and provide the foundation for digital transformation

Azure IoT Central

Fully managed SaaS

Best used when you need to get started quickly with minimal IoT experience

Azure IoT solution accelerators

Customizable PaaS

Best used when you need a lot of control over your IoT solution

+

Azure IoT Edge

On-premises processing

Adds capacity to do local processing

Azure **IoT solution accelerators**

- End-to-end implementation
- Completely customizable
- Open-source microservices based architecture
- Device connectivity and management
- Dashboards, visualization and insights
- Workflow automation and integration
- Command and control
- Preconfigured solutions
  - Remote Monitoring
  - Connected Factory
  - Predictive Maintenance
  - DPS

IoT Central Accelerate time to value

Start quickly for common IoT scenarios

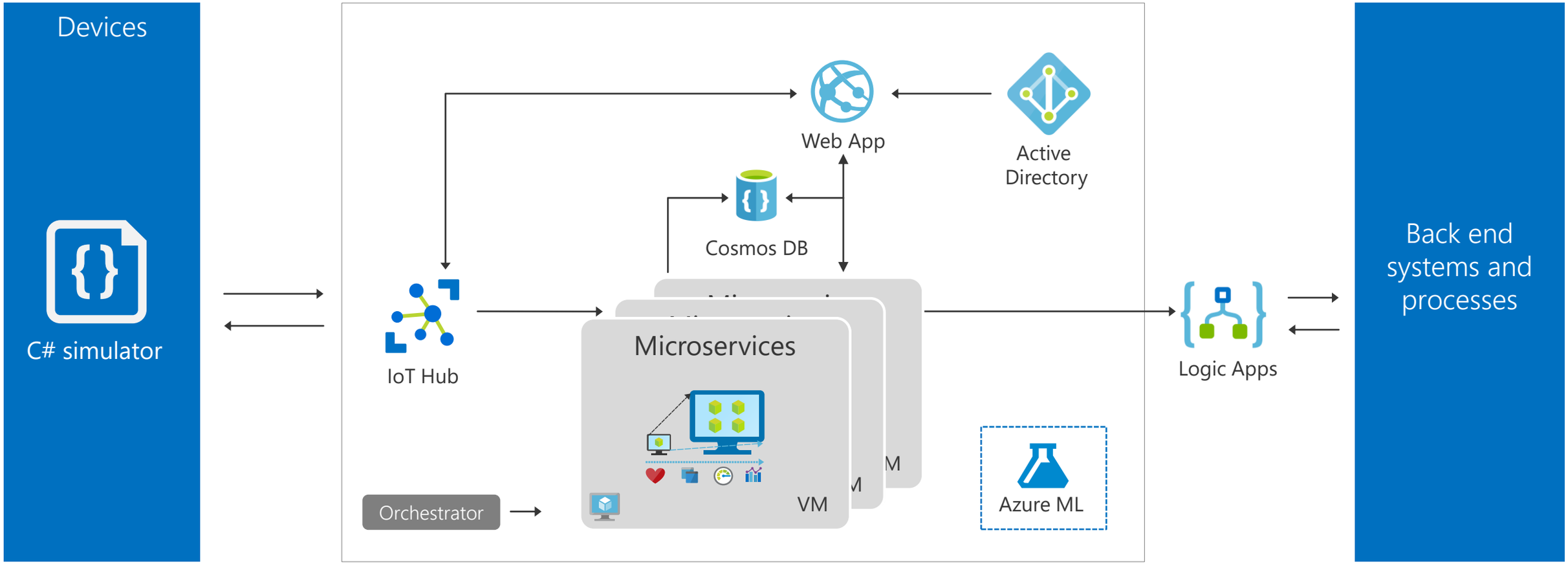
- Get started in minutes
- Modify existing rules and alerts
- Add your devices and begin tailor to your needs

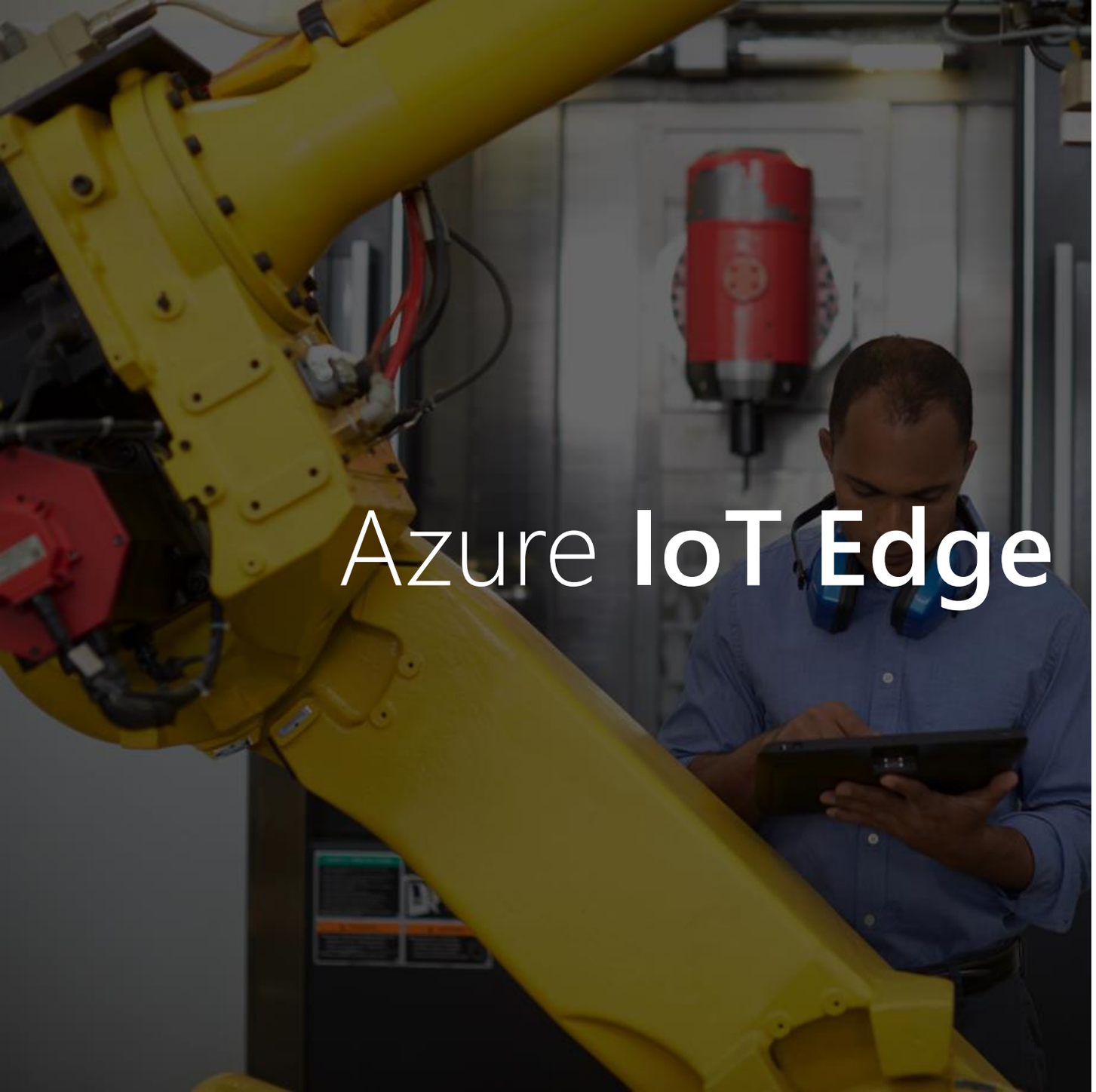
Finish with your IoT application

- Fine-tuned to specific assets and processes
- Highly visual for your real-time operational data
- Integrate with back-end systems

Components of a preconfigured solution

remote monitoring | predictive maintenance | connected factory | device simulation

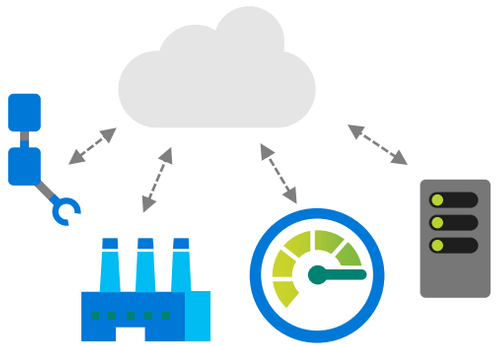


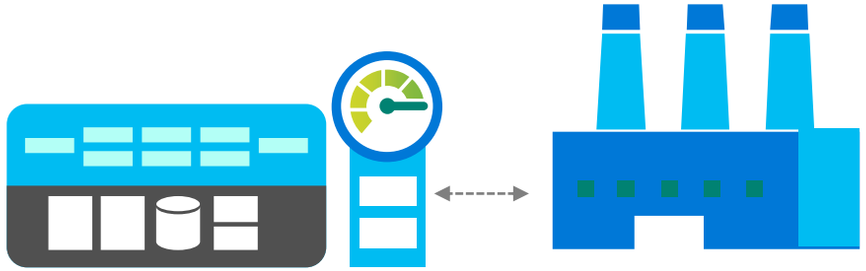


Azure **IoT Edge**

- Move cloud and custom workloads to the edge, securely
- Seamless deployment of AI and advanced analytics
- Configure, update and monitor from the cloud
- Compatible with popular operating systems
- Code symmetry between cloud and edge for easy development and testing
- Secure solution from chipset to cloud

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

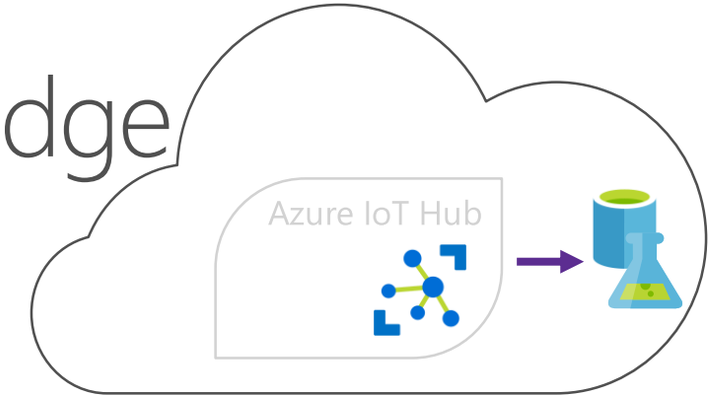
Low latency tight control loops require near real-time response

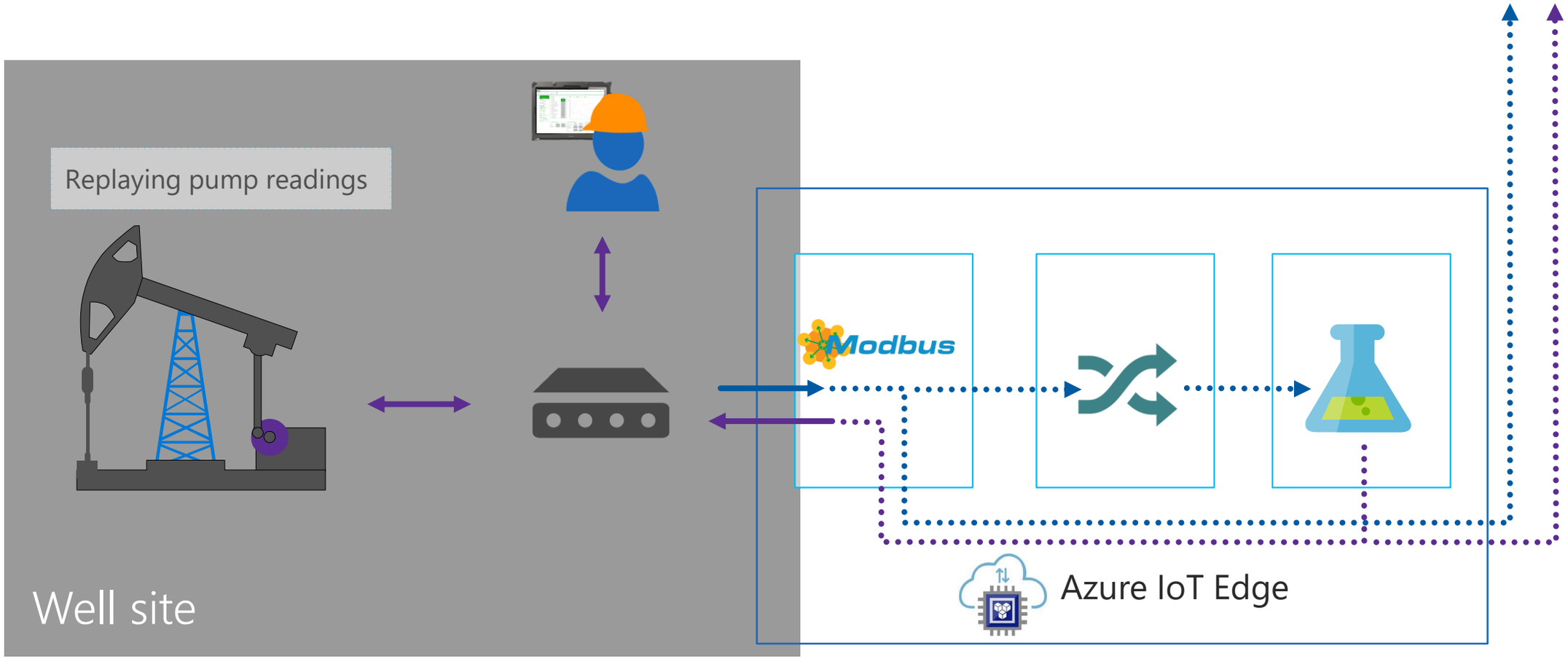
Protocol translation & data normalization

Privacy of data and protection of IP

Symmetry

Schneider – Machine Learning on the Edge Model On IoT Edge Example





Design

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 provide 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, Linux, ARM, x86 Commodity HW

Portable

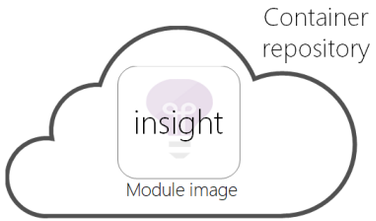
Enables Dev/Test of edge workloads in the cloud with later deployment to the edge as part of a continuous integration / continuous deployment pipeline

Extensible

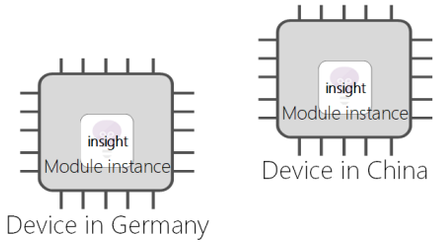
Enables seamless deployment of advanced capabilities such as AI from Microsoft, and any third party, today and tomorrow

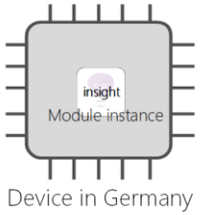
Concept – Module

Module image lives in the cloud



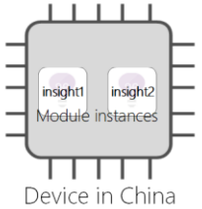
Module instances run on-premises





Device /devices/Hannover01

Modules /devices/Hannover01/modules/insight

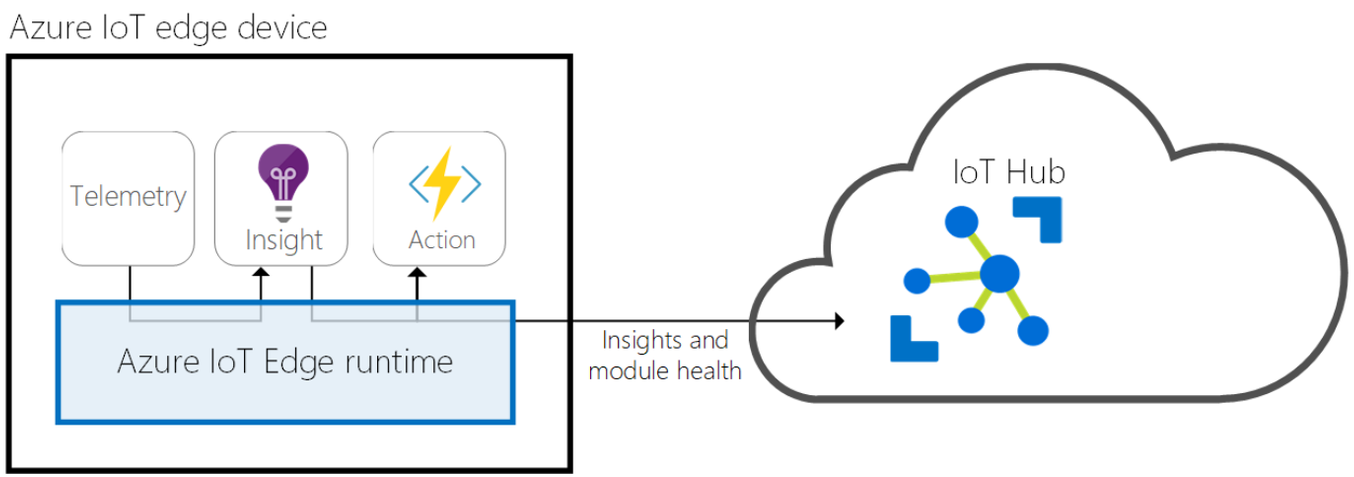


Device /devices/Shenzhen01

Modules /devices/Shenzhen01/modules/insight1
/devices/Shenzhen01/modules/insight2

- 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.

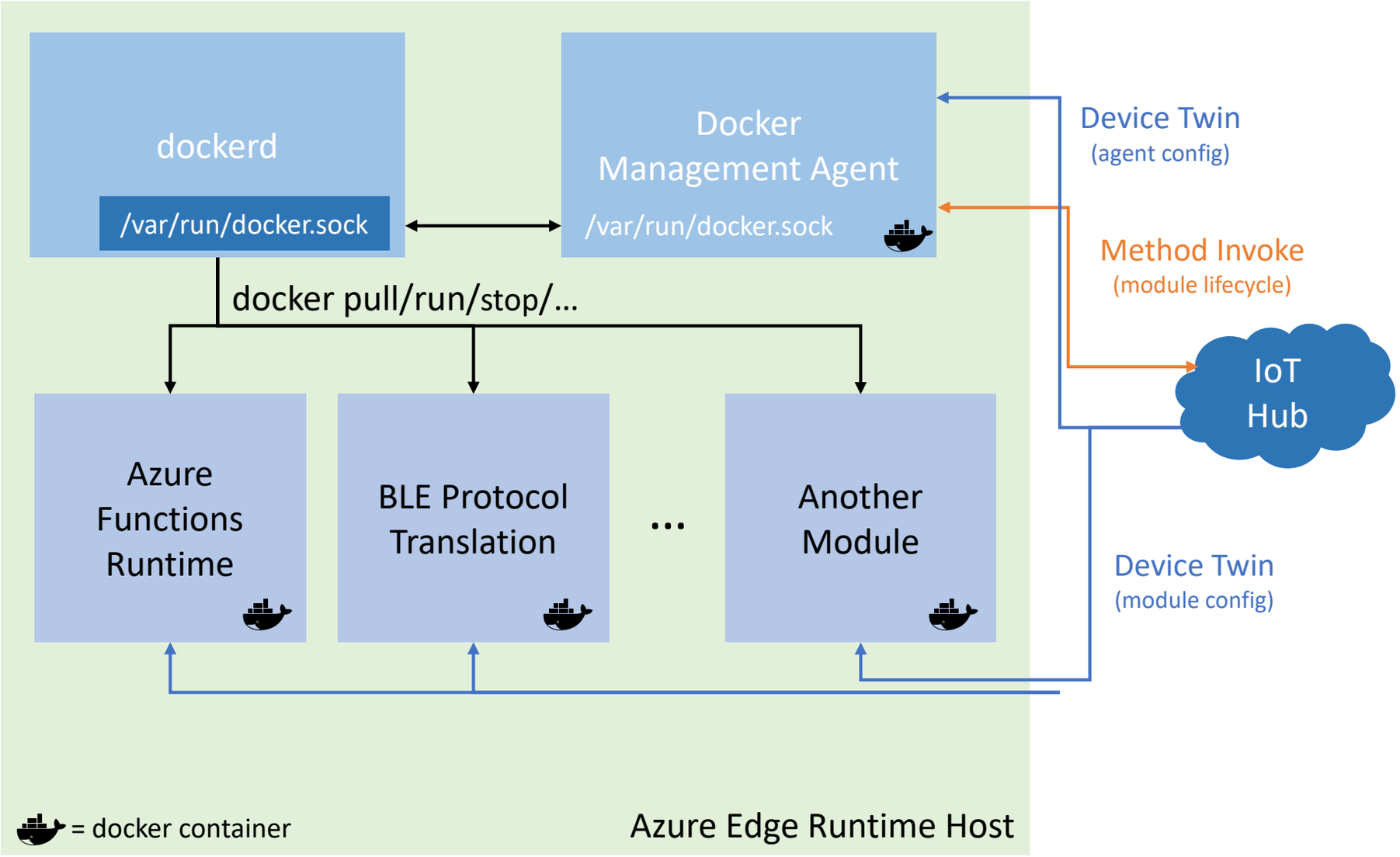
Concept – Azure IoT Edge Runtime



- 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

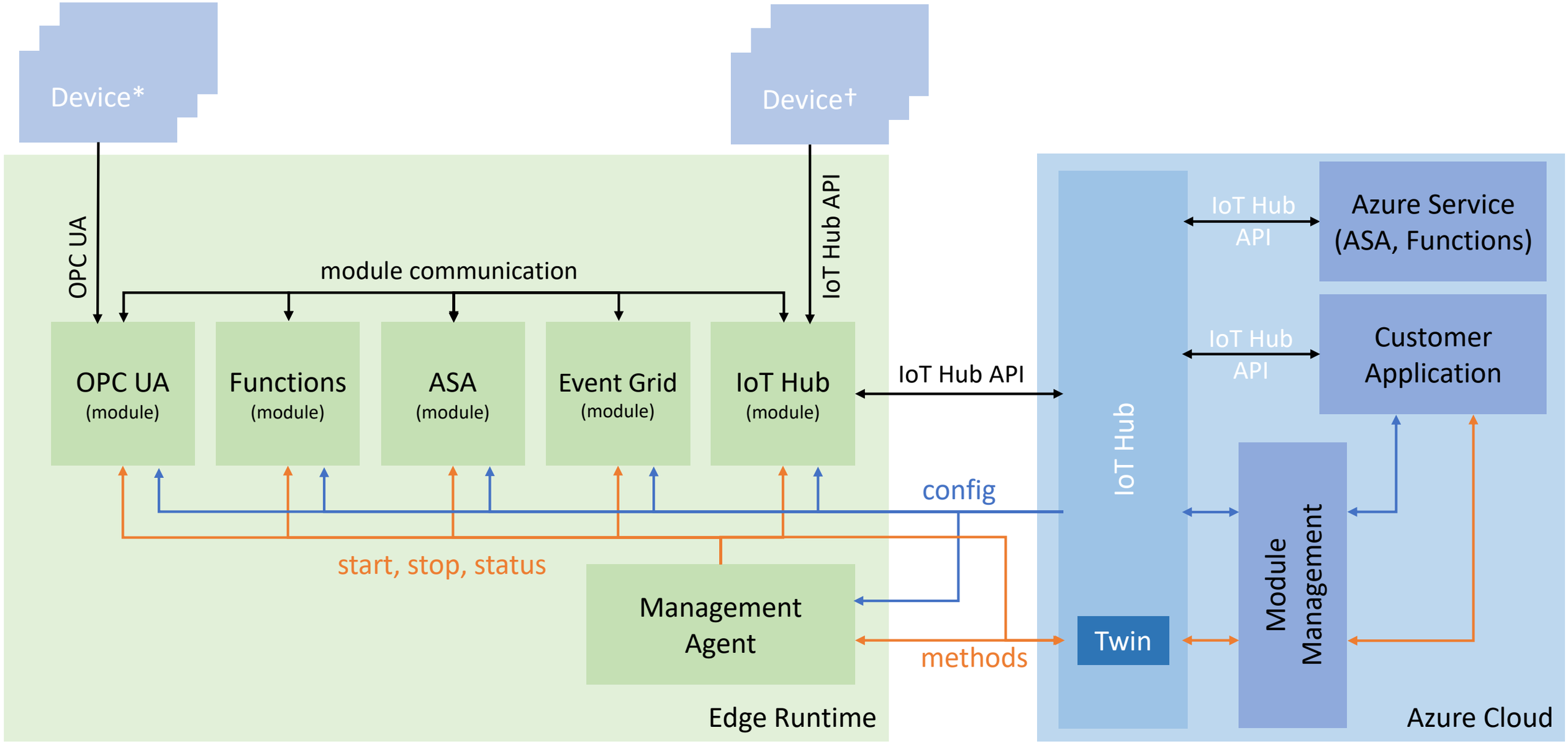
Edge Runtime

Using Docker Containers



Note: containers allow modules to be developed in language/framework of choice

docker command
config (device twin)
control (methods)

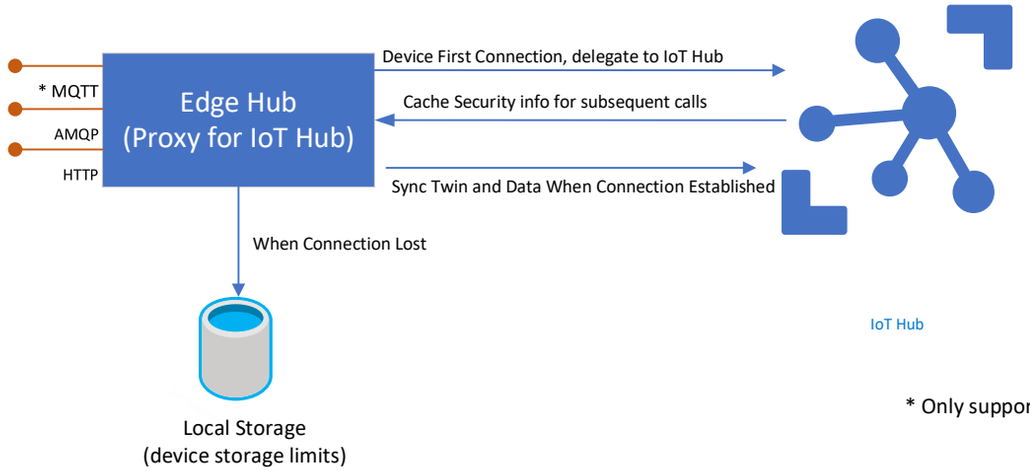


data
config (device twin)
control (methods)

* Devices requiring module for protocol translation
† Devices capable of using IoT Hub SDK

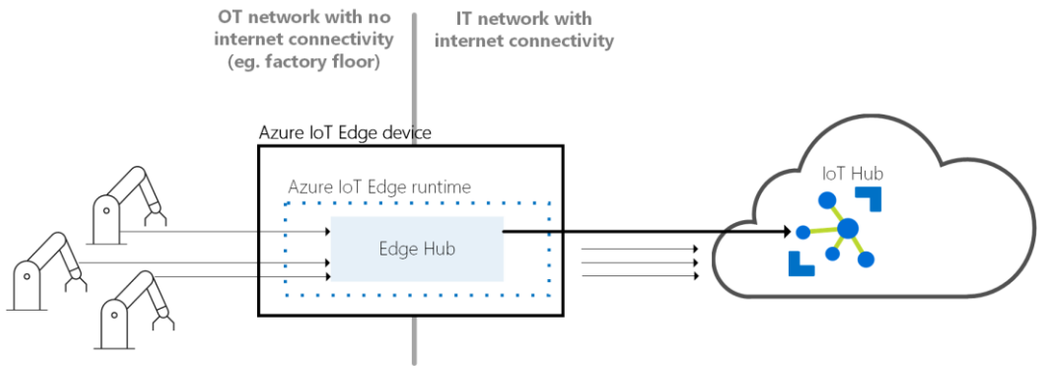
Azure IoT Edge Runtime Modules – Edge Hub

Edge Hub

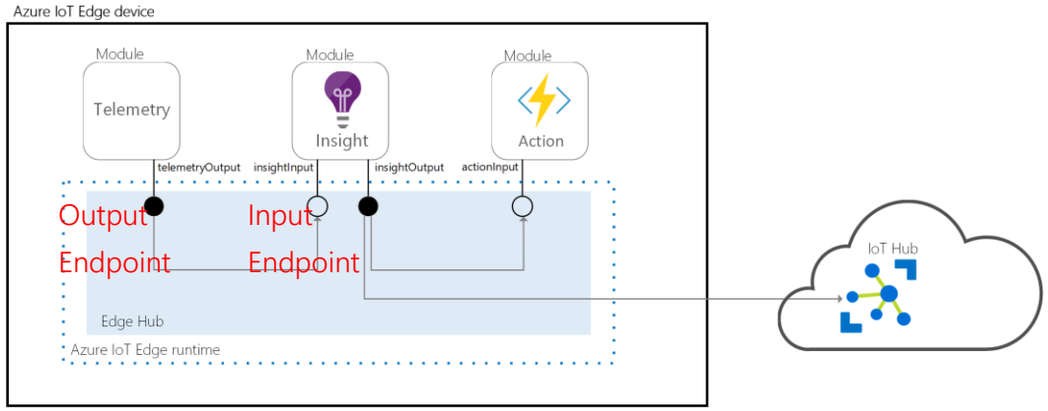


* Only support in public preview

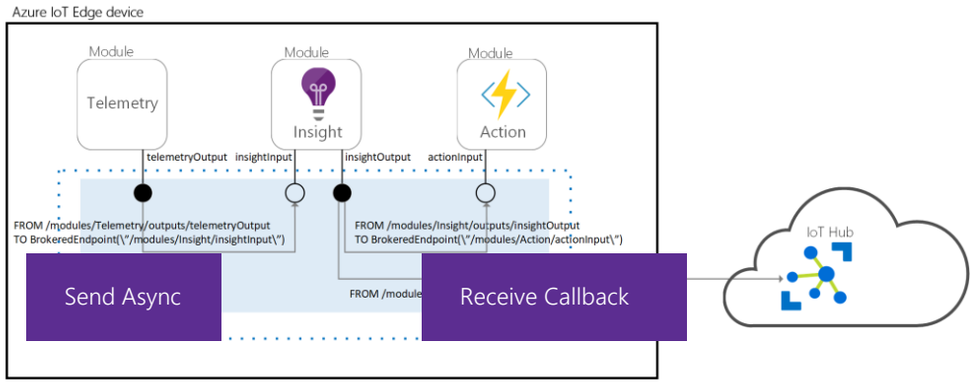
Supports Multiplexing

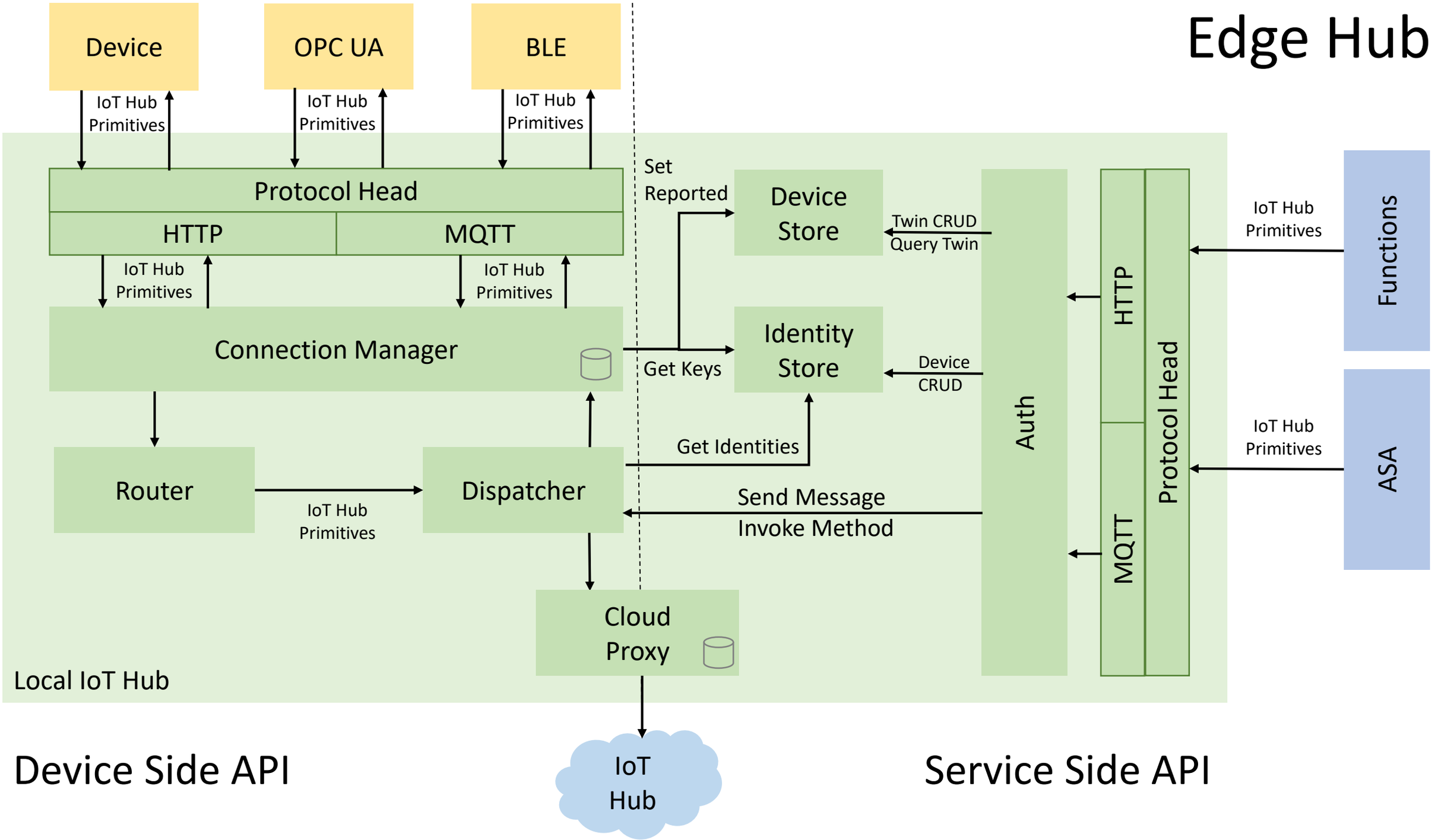


Module Communication



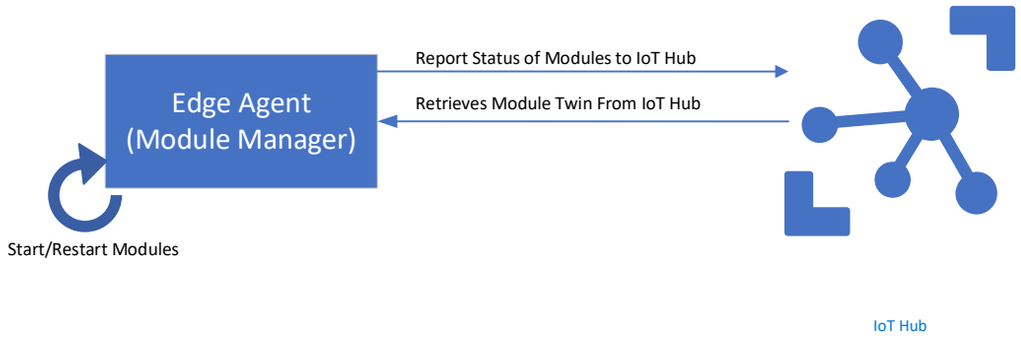
Send/Receive Data




Edge Hub
Device
OPC UA
BLE
IoT Hub Primitives
IoT Hub Primitives
IoT Hub Primitives
Protocol Head
HTTP
MQTT
IoT Hub Primitives
IoT Hub Primitives
Connection Manager
Router
IoT Hub Primitives
Dispatcher
Cloud Proxy
Local IoT Hub
Set Reported
Device Store
Twin CRUD
Query Twin
Identity Store
Get Keys
Device CRUD
Get Identities
Send Message
Invoke Method
Auth
HTTP
MQTT
Protocol Head
IoT Hub Primitives
Functions
IoT Hub Primitives
ASA
Device Side API
IoT Hub
Service Side API

Azure IoT Edge Runtime Modules – Edge Agent

Edge Agent



Module Dictionary

- Image Settings
- Create Options (Sent to Docker)
- Status
- Restart Policy

Supported Statuses

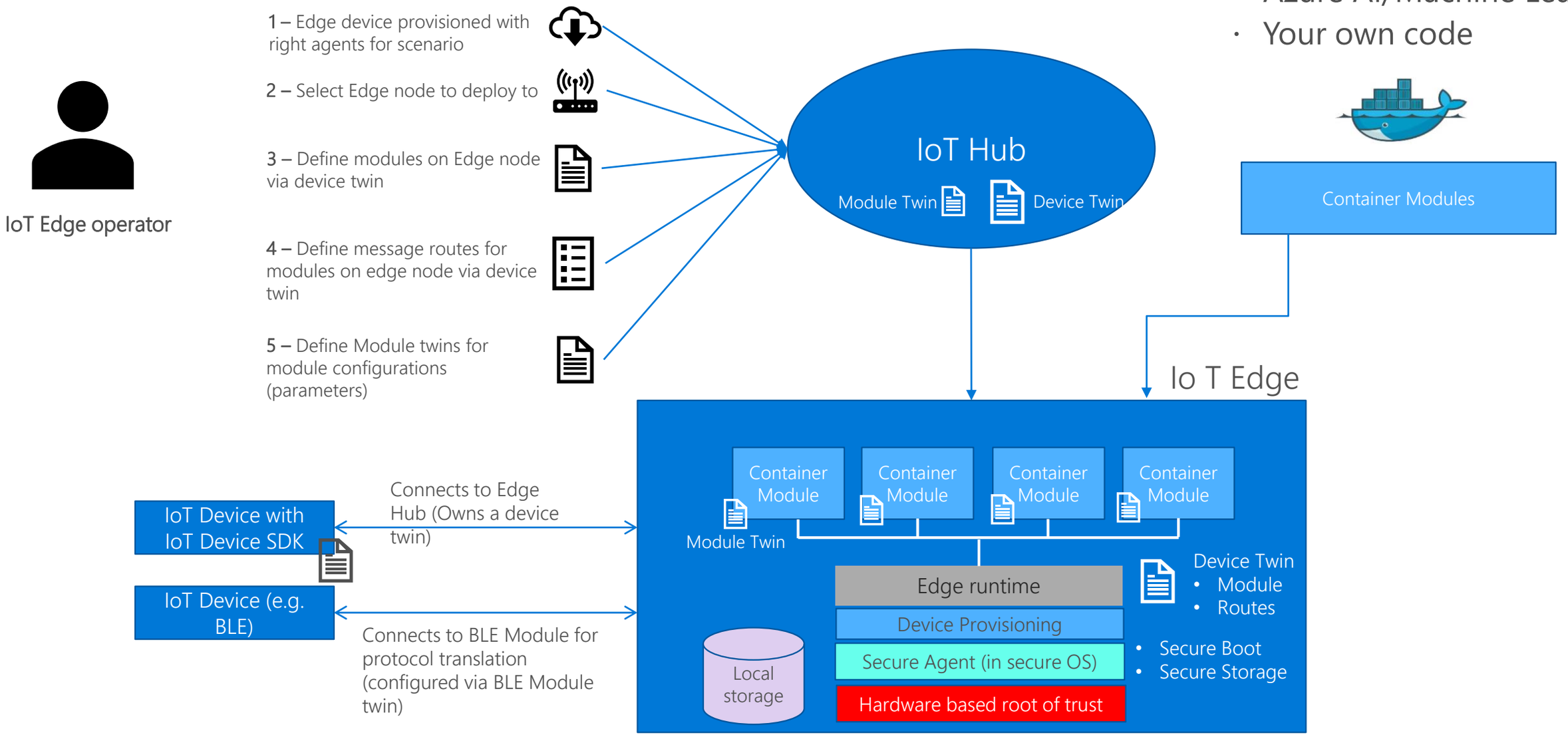
- Downloading
- Running
- Unhealthy
- Failed
- Stopped

Restart Policy

- Never
- On Failure
- Unhealthy
- Always

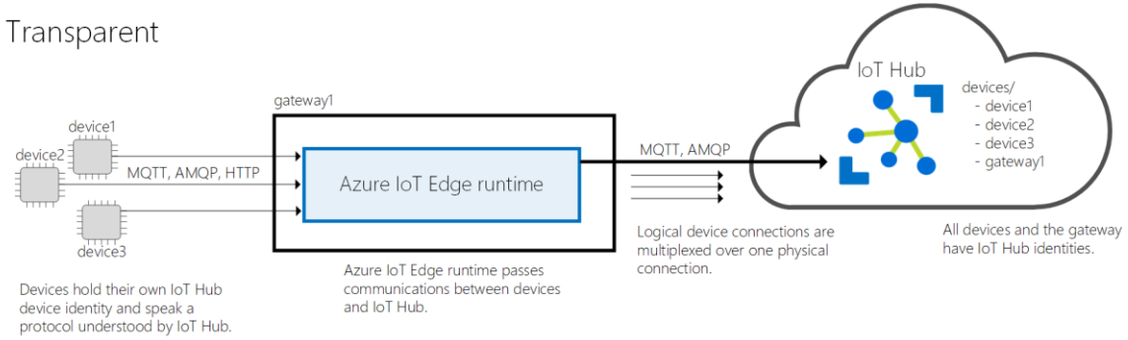
IoT Edge in action

- Container based workloads
- Azure Functions
- Azure Stream Analytics
- Azure AI/Machine Learning
- Your own code

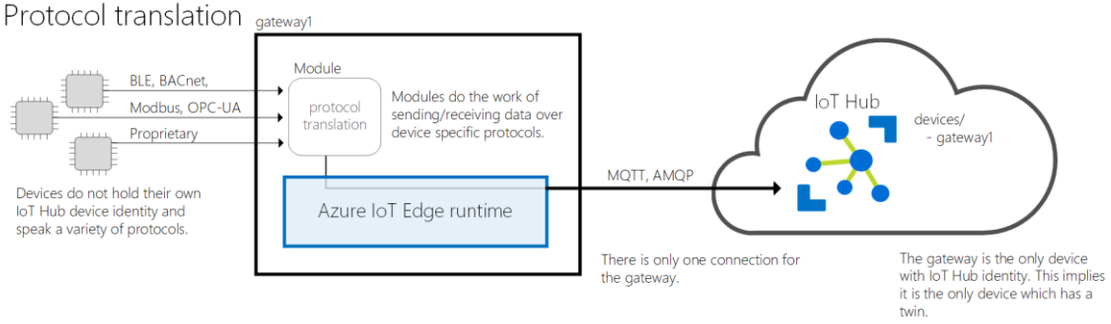


Azure IoT Edge as a Gateway (Patterns)

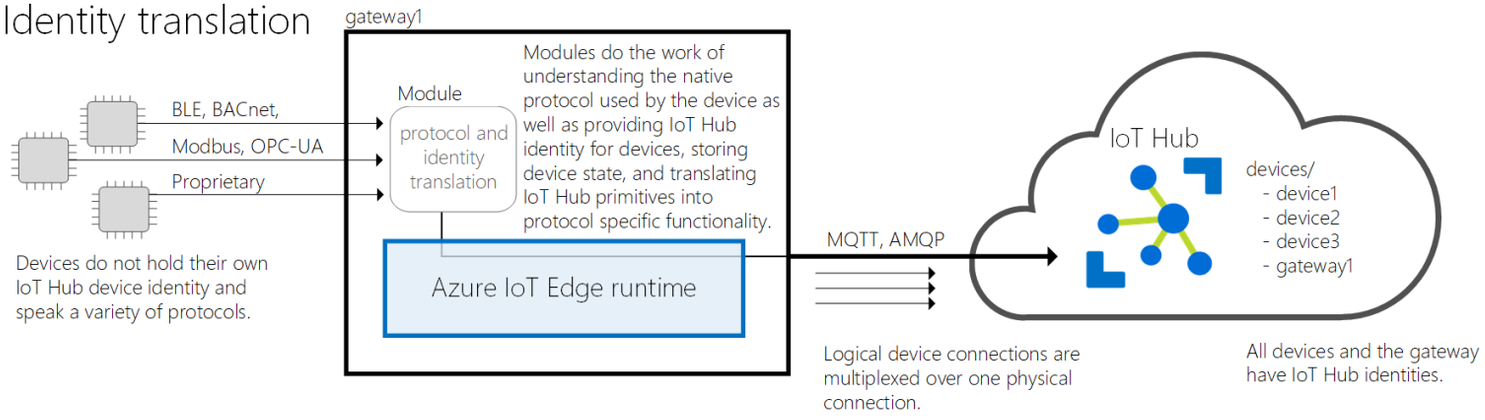
Transparent



Protocol translation



Identity translation



Azure IoT Edge Gateway Use Cases

Edge Analytics

(ML & AI Services)

Downstream Device Isolation

(Shield Downstream Devices from Internet)

Connection Multiplexing

(All Devices use same underlying connection)

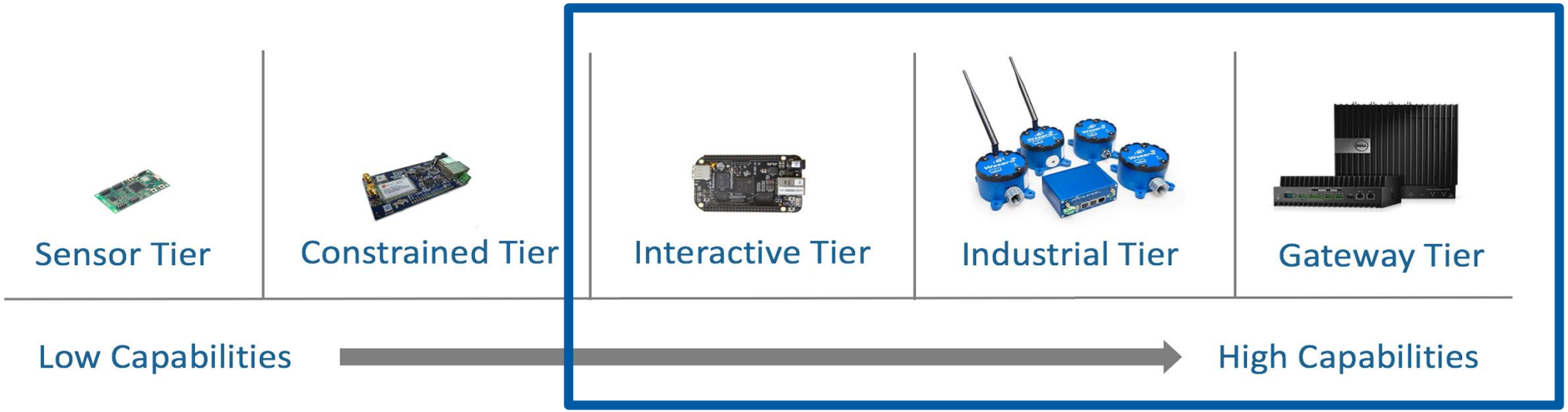
Traffic Smoothing

(Back-off in case of IoT Hub throttling and use local storage)

Limited Offline Support

(Store messages and twins locally when IoT Hub connectivity lost)

Hardware for Azure IoT Edge



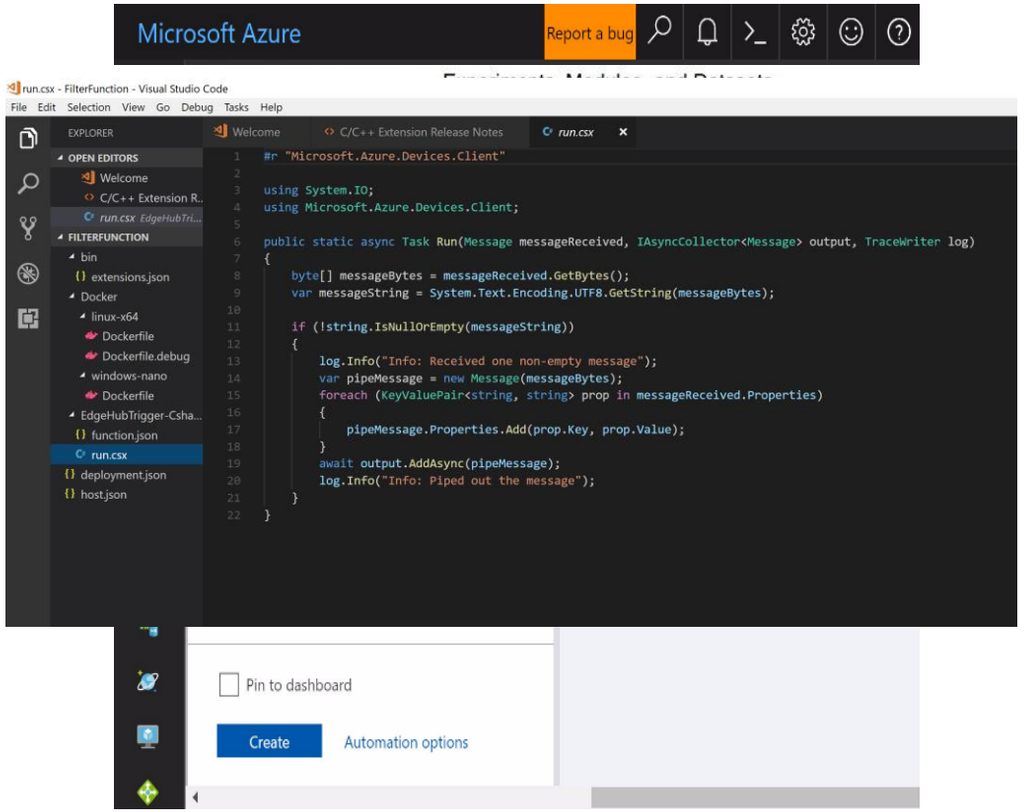
Linux and Windows supported on x64 and ARM (support for containers required)

Hardware sizing dependent on workloads

Internal tests on devices as small as Raspberry Pi 3

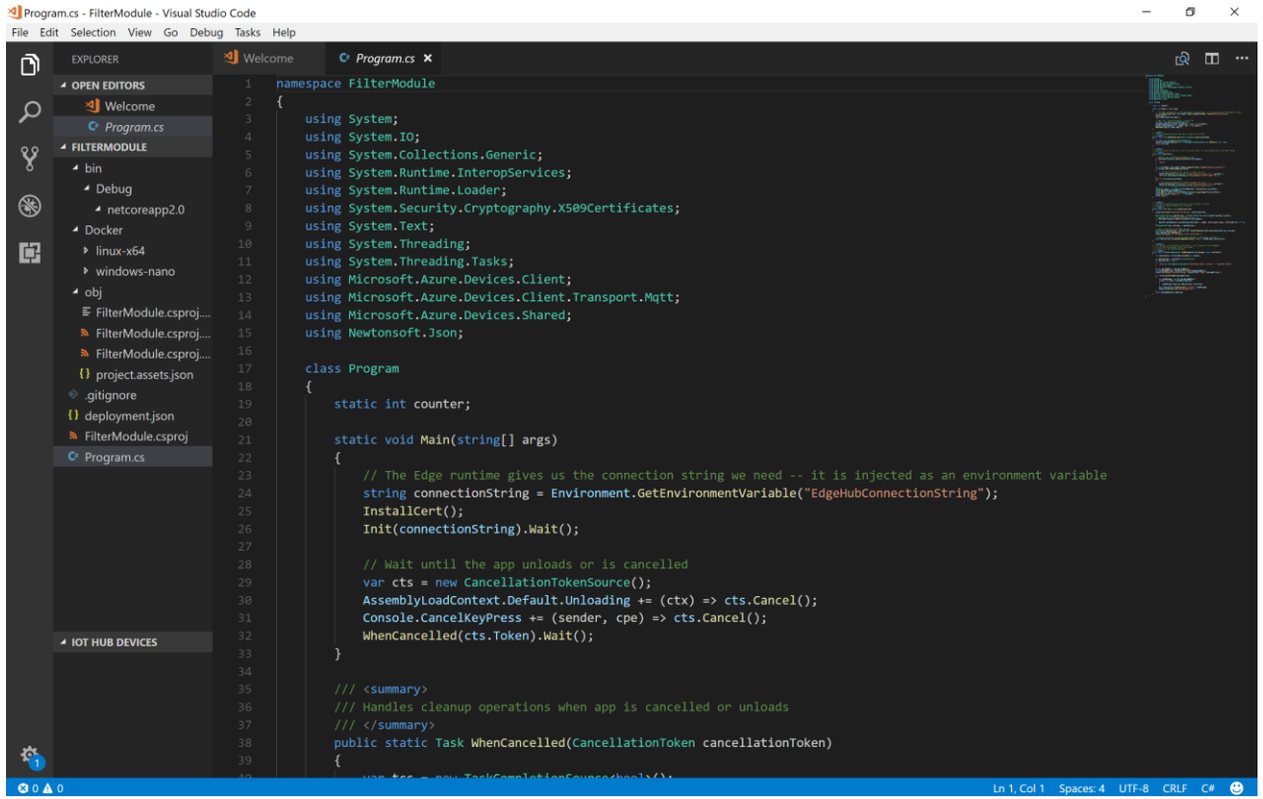
Azure IoT Edge – Package services in containers

- **Azure Stream Analytics –** In line experience in the ASA web portal
- **Azure Functions –** Use VSCode to develop Function for your scenario and package as a container
- **AI and Azure Machine Learning –** Package AI and ML model as a module in a container after training using Azure ML Python SDK, ML Studio, Custom Vision AI. Deploy packaged ML modules to the IoT Edge

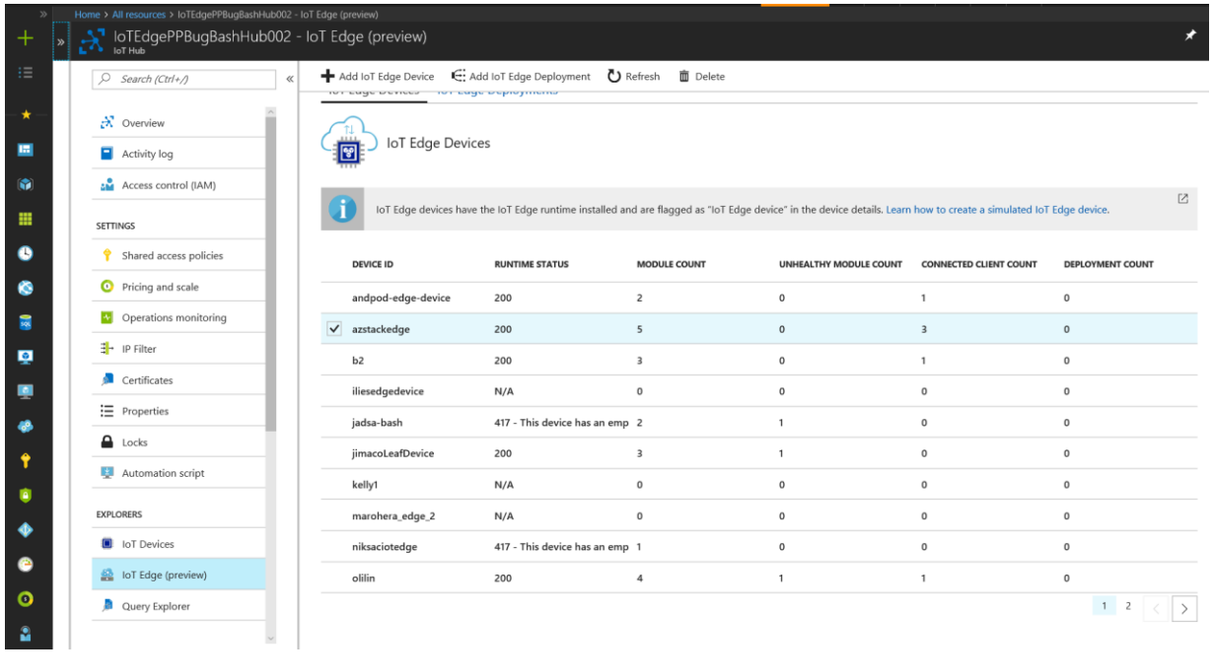


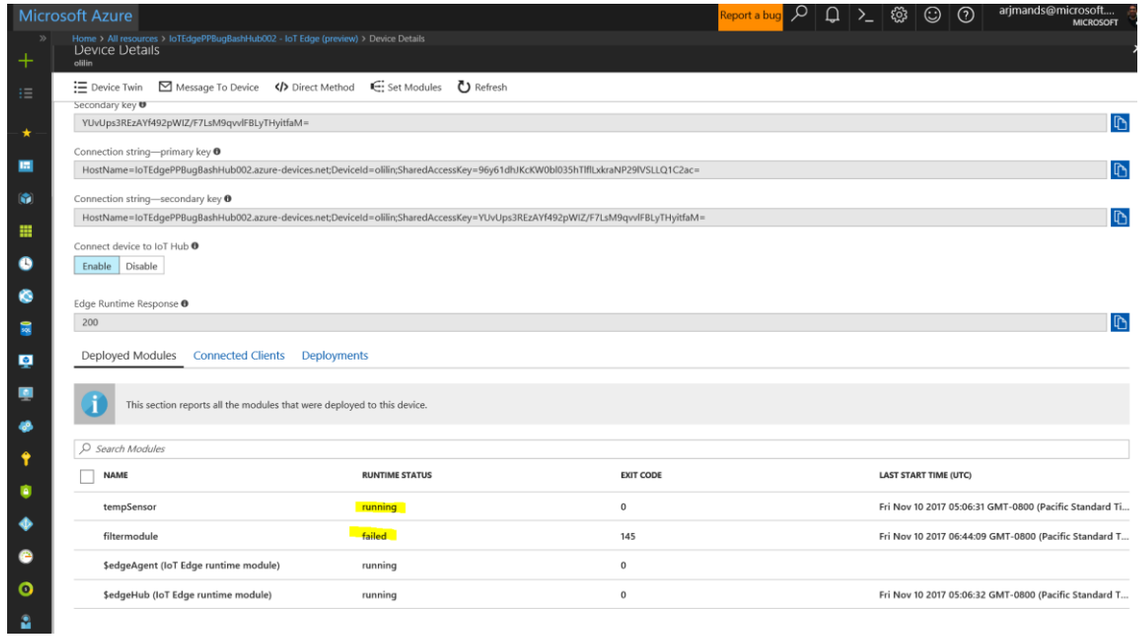
Azure IoT Edge – Developer experience

- Azure IoT SDK for developing modules, which provide:
  - Protocol and messaging support
  - Security for module (identification and authentication)
  - Module twin support
- Develop and debug in your favorite language (C# released, C, Python, Java and Node.JS coming soon)
- Build container with your code and host in a container repo (e.g. Docker Hub or Azure Container Registry)



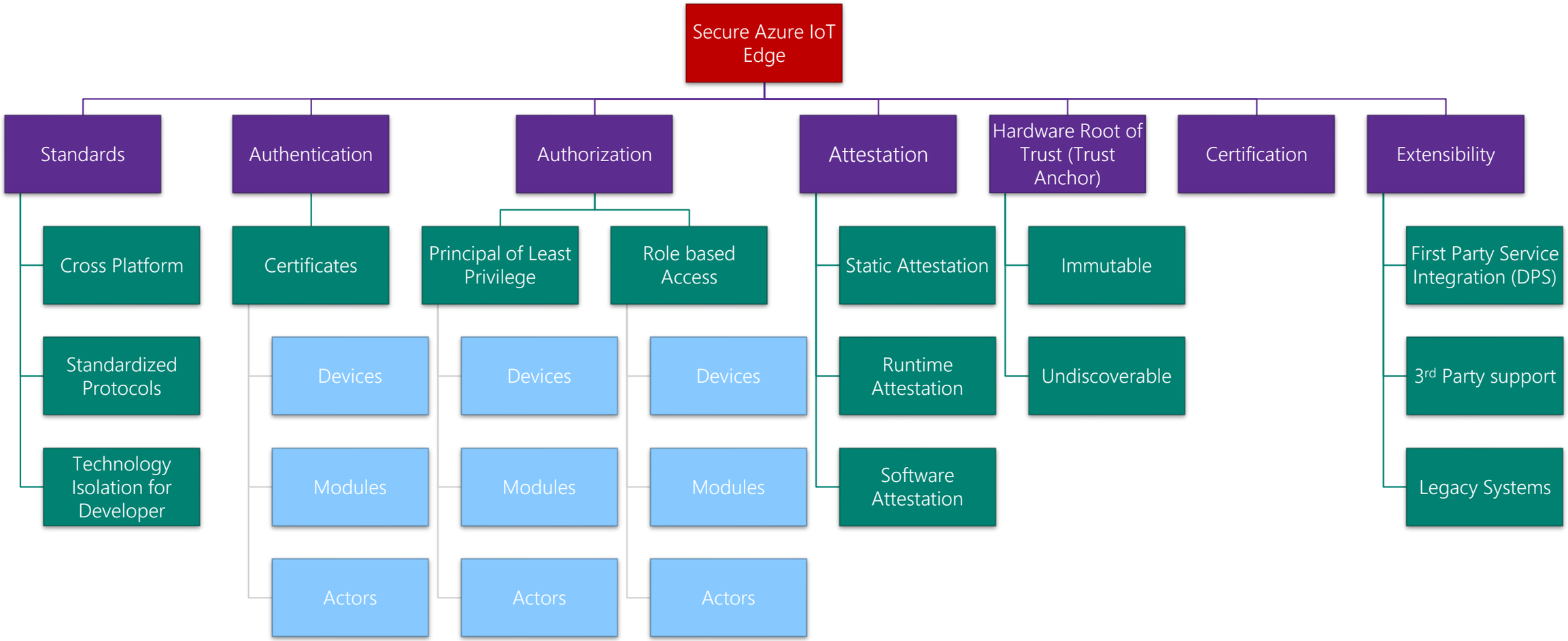
Azure IoT Edge - Deployment and management





Portal demo

Securing Azure IoT Edge



Enable edge intelligence with Azure IoT Edge

Extending the power of the cloud to the edge

Learn more: aka.ms/azure-iot-edge

Get started: aka.ms/azure-iot-edge-get-started

Demo

Azure IoT Edge

- Deploy *Azure services* to IoT Edge devices
- *Manage* IoT Edge and downstream devices
- Deploy your *own code in language of your choice*
- Do all of this *securely*, in a *scalable fashion* from the Azure IoT Hub

Microsoft