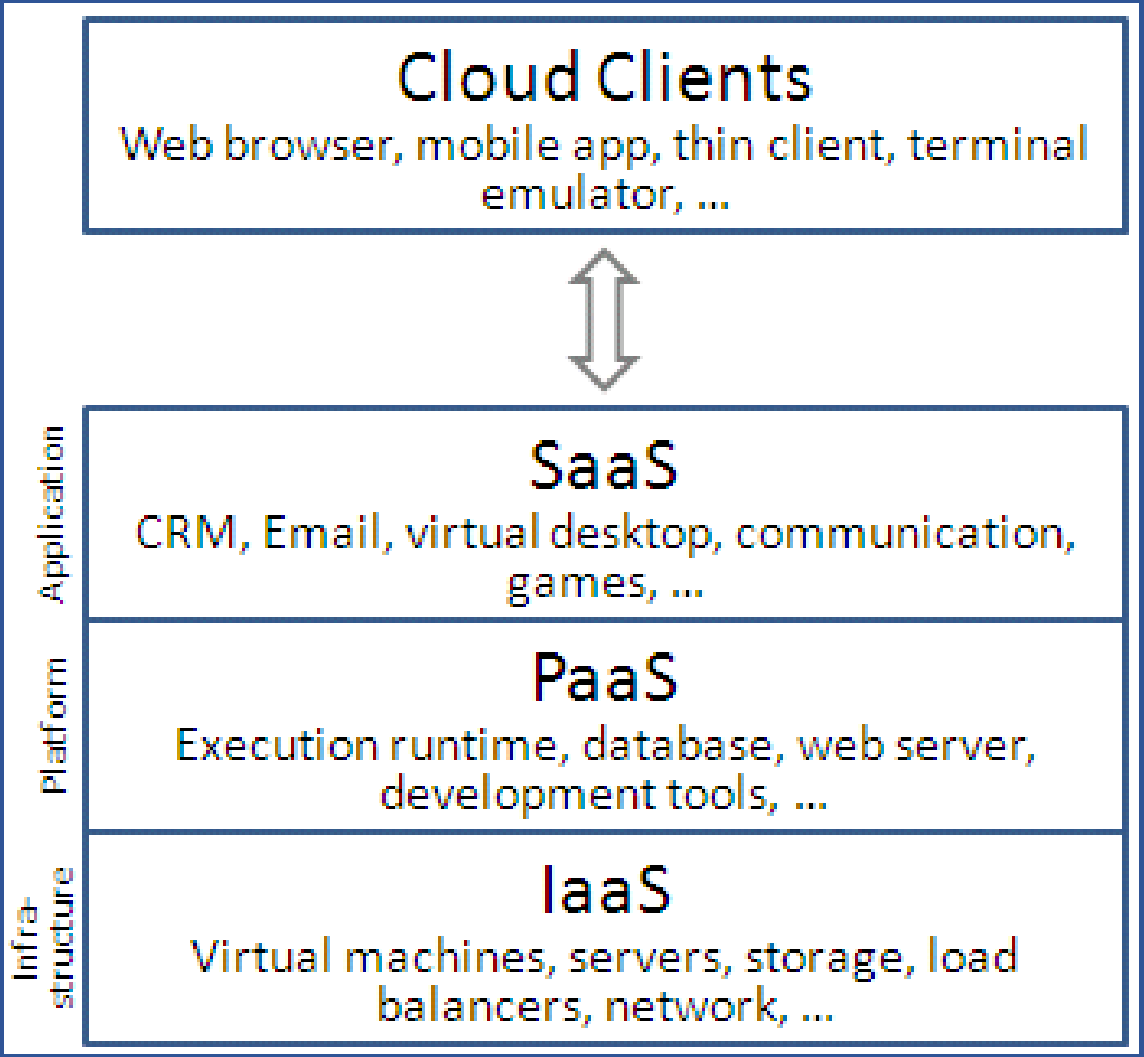
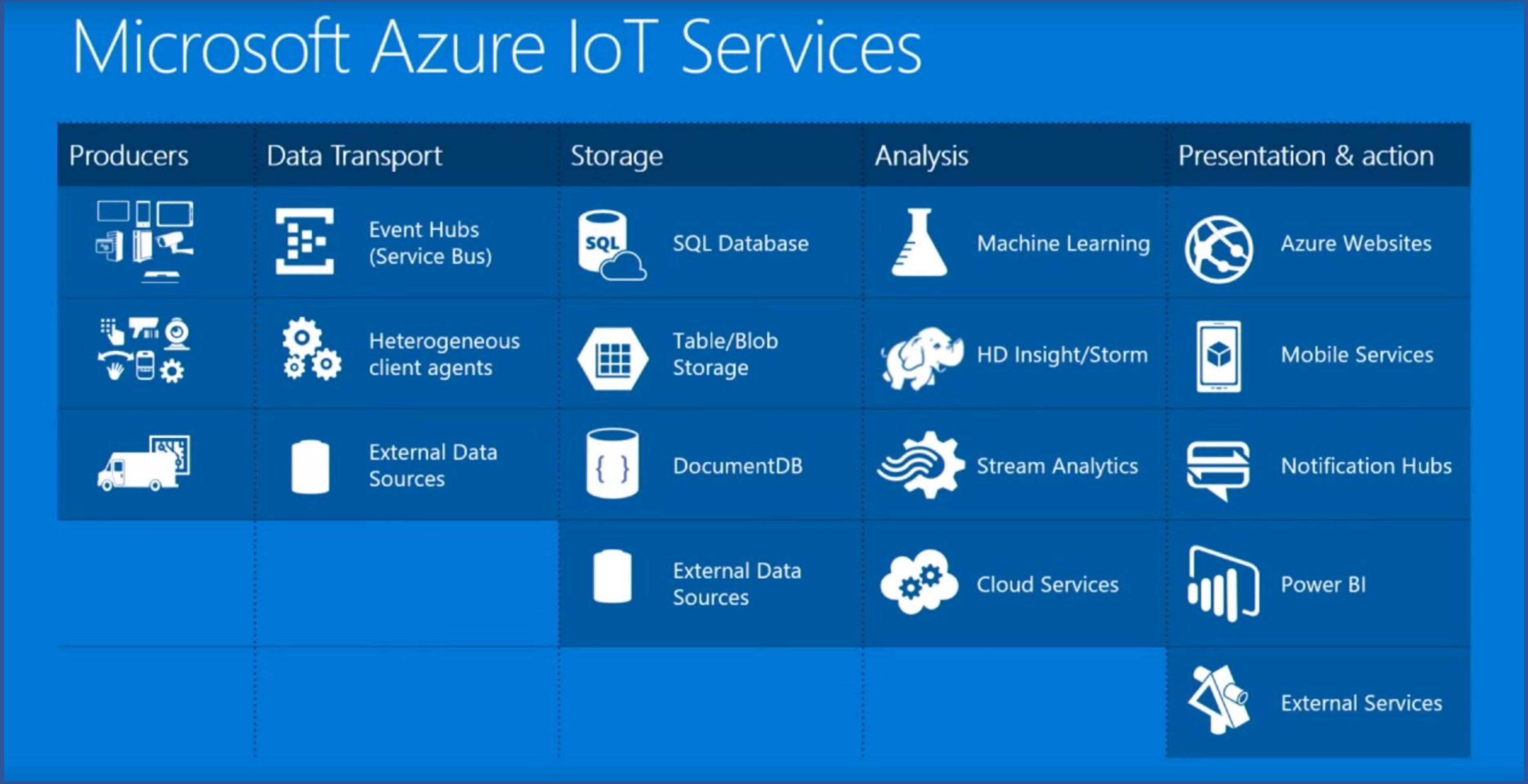
Getting started with Microsoft Azure IoT suite

Cloud services

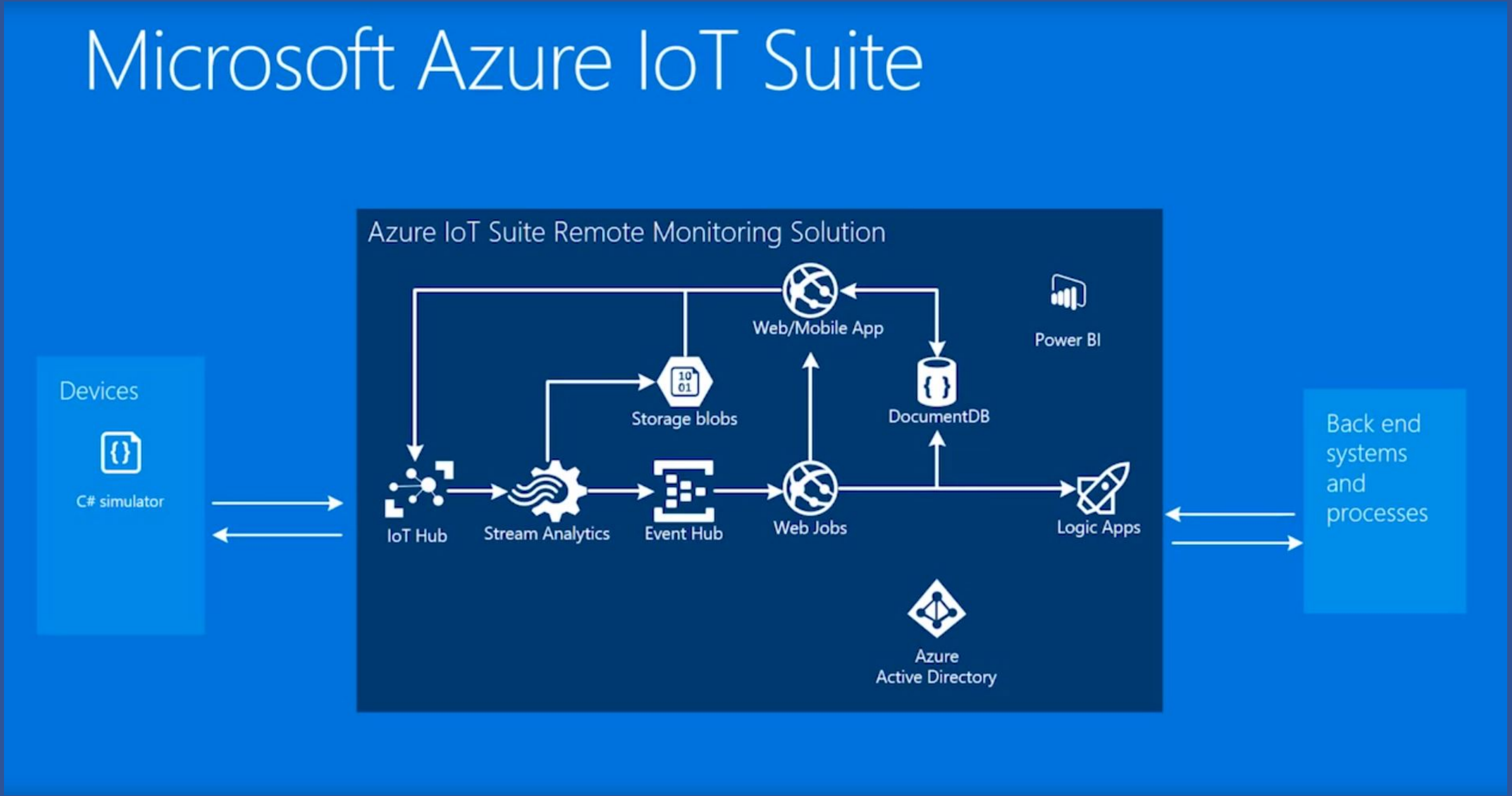


Cloud Clients
Web browser, mobile app, thin client, terminal emulator, ...
Application
SaaS
CRM, Email, virtual desktop, communication, games, ...
Platform
PaaS
Execution runtime, database, web server, development tools, ...
Infra-structure
IaaS
Virtual machines, servers, storage, load balancers, network, ...



Microsoft Azure IoT Services

Producers	Data Transport	Storage	Analysis	Presentation & action
	Event Hubs (Service Bus)	SQL Database	Machine Learning	Azure Websites
	Heterogeneous client agents	Table/Blob Storage	HD Insight/Storm	Mobile Services
	External Data Sources	DocumentDB	Stream Analytics	Notification Hubs
		External Data Sources	Cloud Services	Power BI
				External Services



Microsoft Azure IoT Suite

Devices

C# simulator

Azure IoT Suite Remote Monitoring Solution

Web/Mobile App

Power BI

Storage blobs

DocumentDB

IoT Hub

Stream Analytics

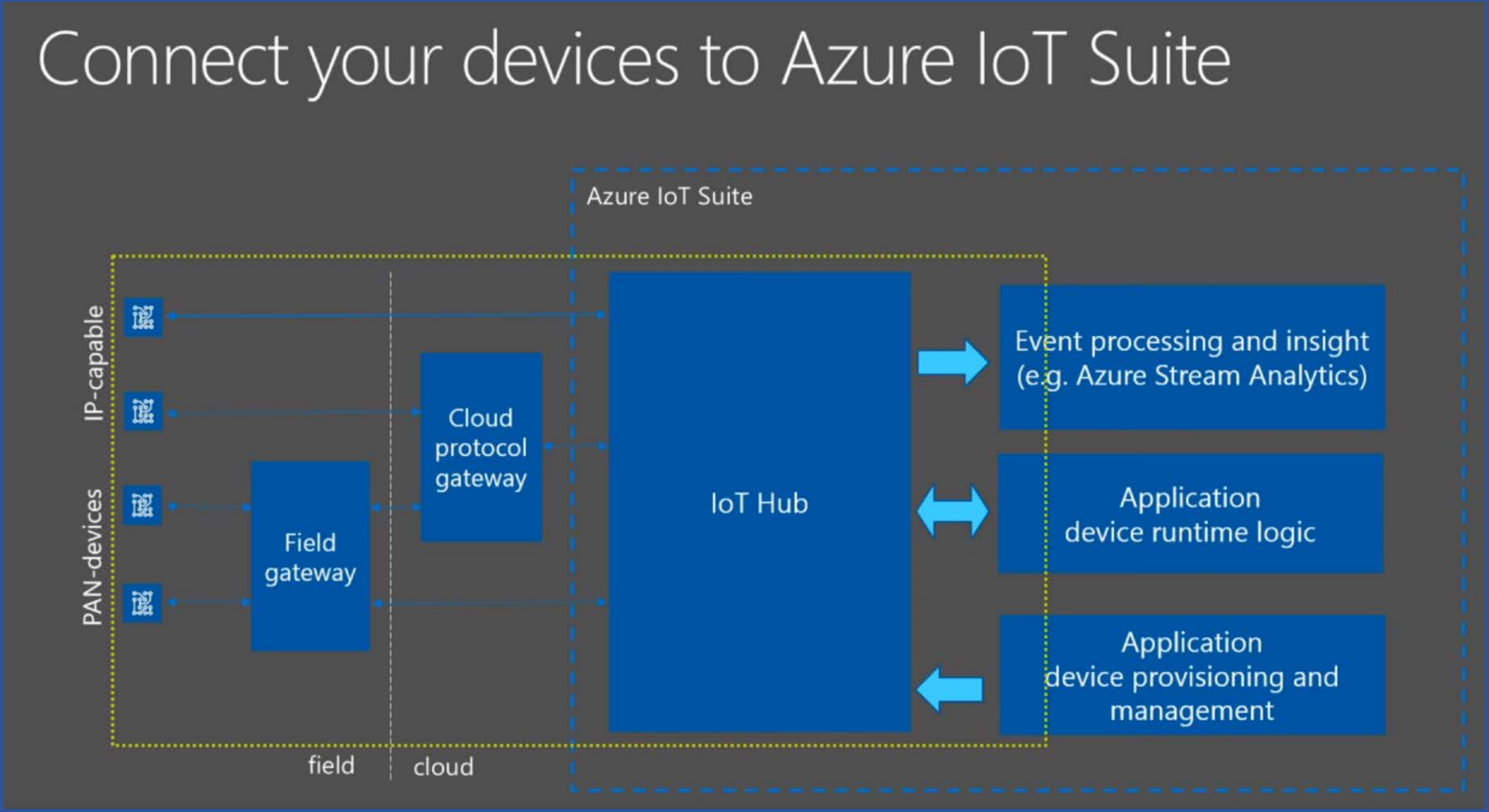
Event Hub

Web Jobs

Logic Apps

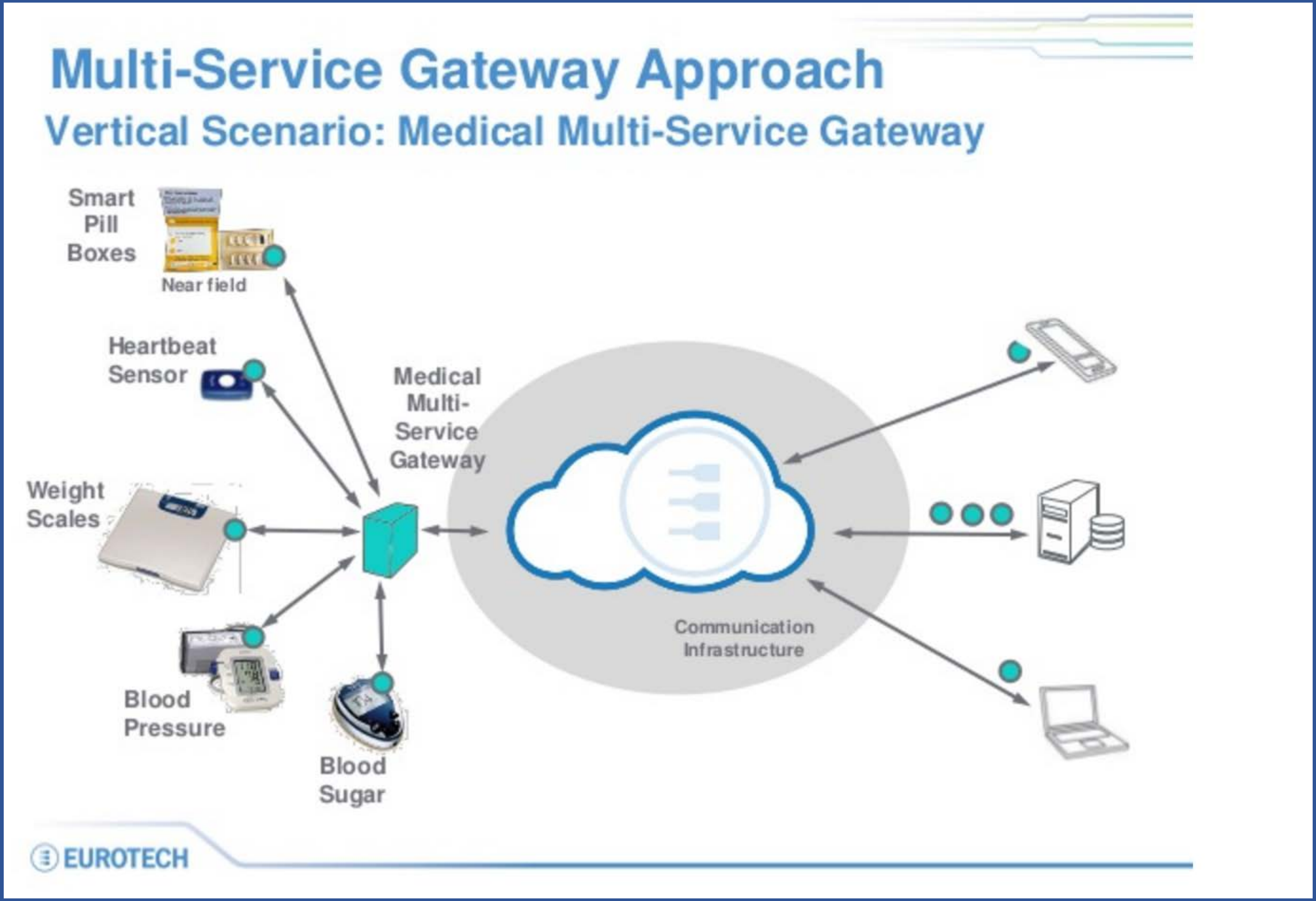
Azure
Active Directory

Back end
systems
and
processes


Connect your devices to Azure IoT Suite
Azure IoT Suite
IP-capable
PAN-devices
Field gateway
Cloud protocol gateway
IoT Hub
Event processing and insight (e.g. Azure Stream Analytics)
Application device runtime logic
Application device provisioning and management
field
cloud

Field Gateway

- Sits between your devices and your IoT hub.
- Located close to your devices.
- Your devices communicate directly with the field gateway by using a protocol supported by the devices.
- The field gateway communicates with IoT Hub using a protocol that is supported by IoT Hub.
- A field gateway can be highly specialized hardware or a low power computer

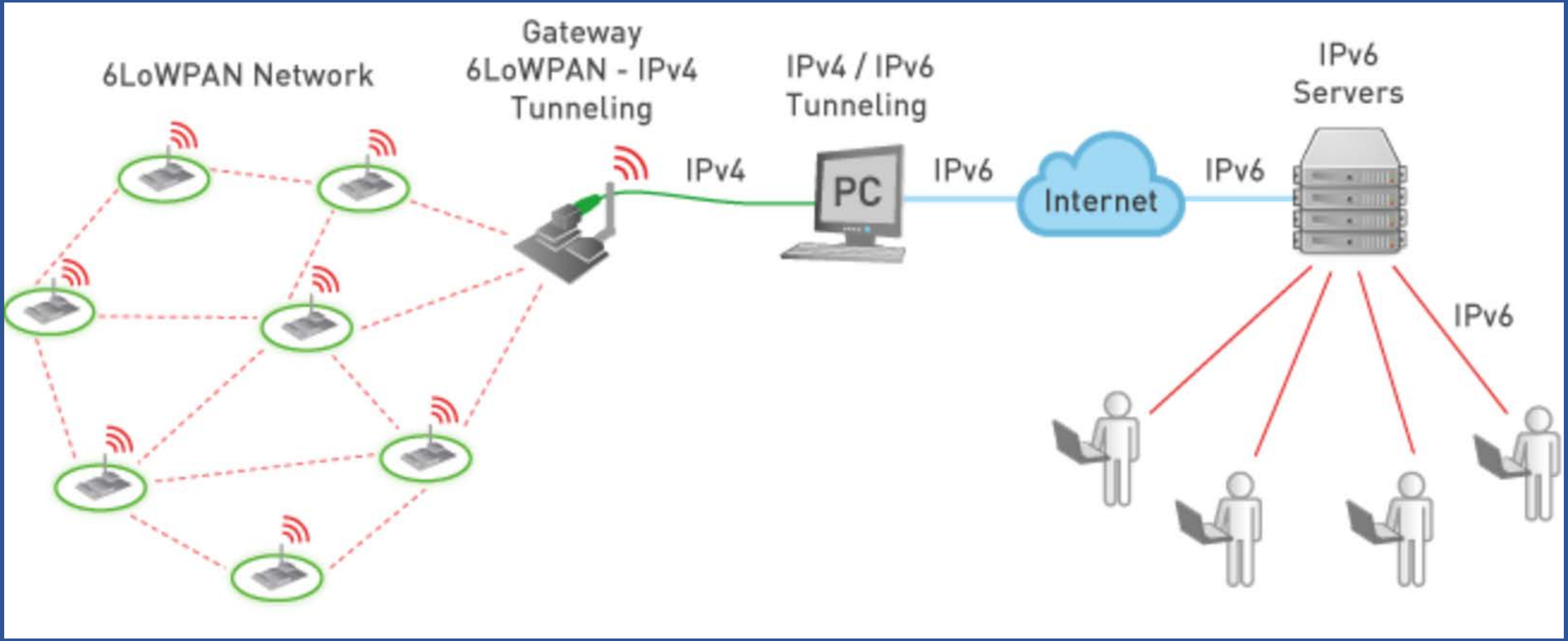


Multi-Service Gateway Approach

Vertical Scenario: Medical Multi-Service Gateway

6LoPAN Network Filed Gateway

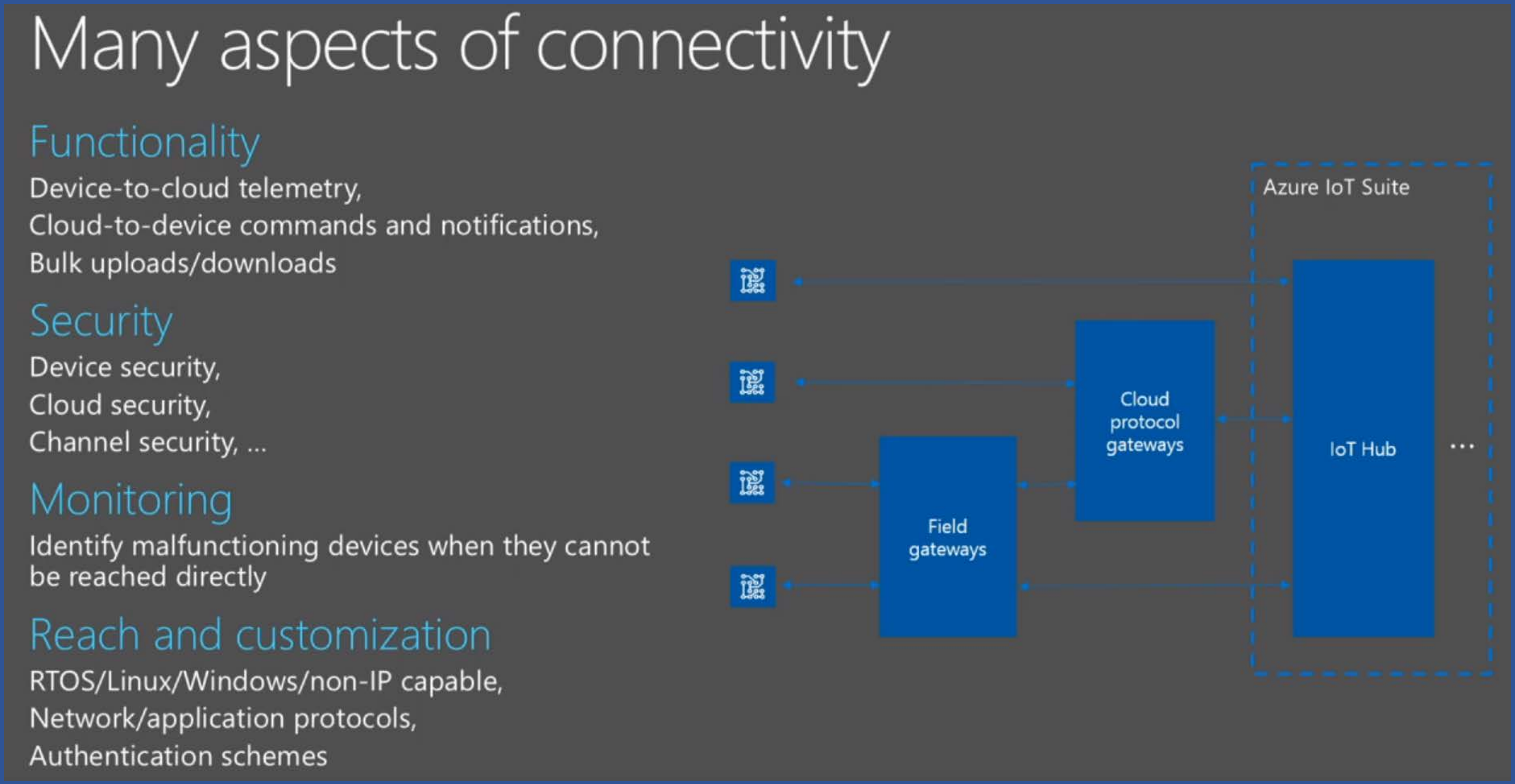
6LoWPAN is an acronym of IPv6 over Low power Wireless Personal Area Networks.



Cloud Protocol gateway

- is a framework for protocol adaptation that is designed for high-scale, bidirectional device communication with IoT Hub.
- is a pass-through component that accepts device connections over a specific protocol; AMQP and MQTT

Connectivity



Many aspects of connectivity

Functionality

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

Security

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

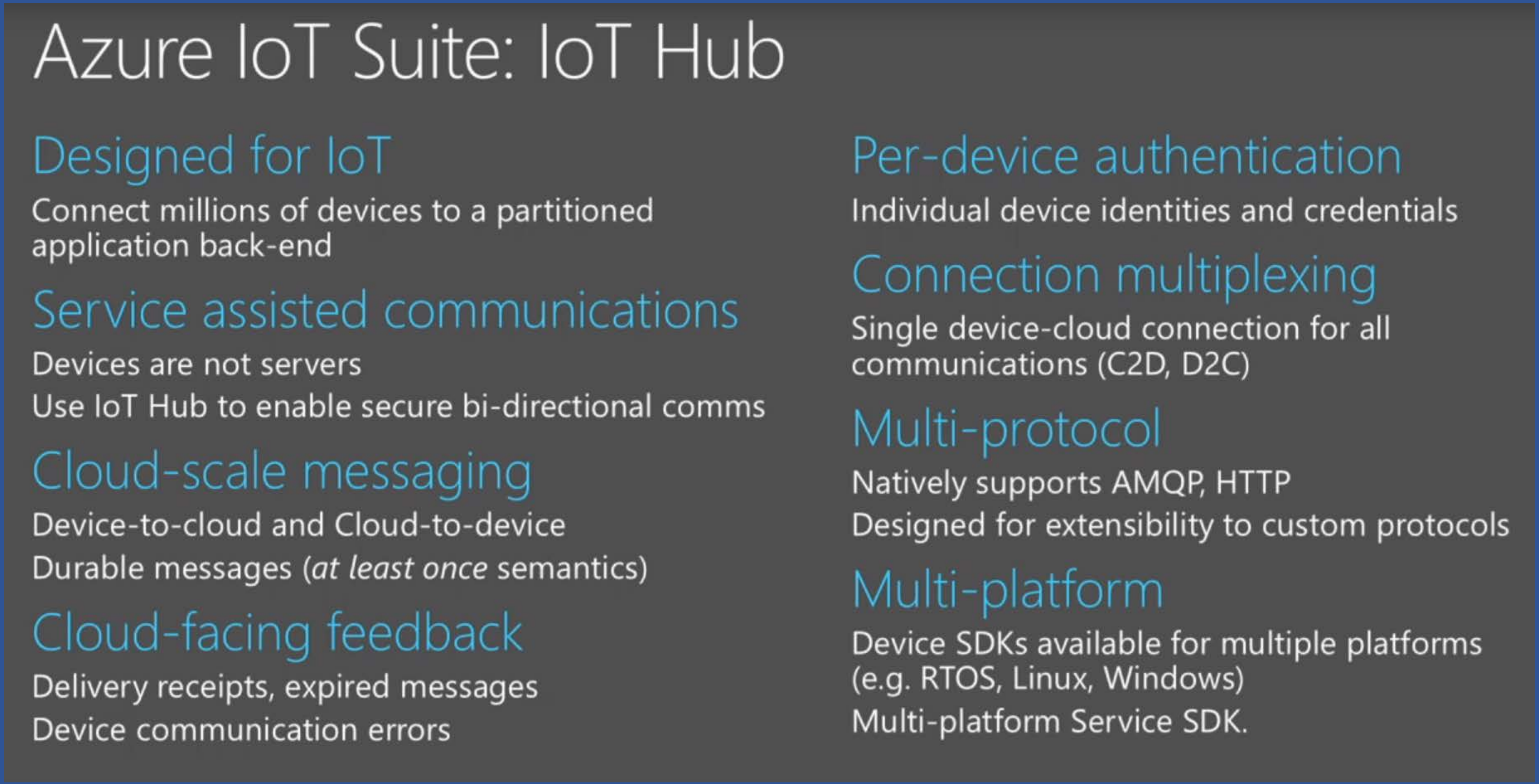
Monitoring

Identify malfunctioning devices when they cannot be reached directly

Reach and customization

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

IoT Hub



Azure IoT Suite: IoT Hub

Designed for IoT

Connect millions of devices to a partitioned application back-end

Service assisted communications

Devices are not servers

Use IoT Hub to enable secure bi-directional comms

Cloud-scale messaging

Device-to-cloud and Cloud-to-device

Durable messages (*at least once* semantics)

Cloud-facing feedback

Delivery receipts, expired messages

Device communication errors

Per-device authentication

Individual device identities and credentials

Connection multiplexing

Single device-cloud connection for all communications (C2D, D2C)

Multi-protocol

Natively supports AMQP, HTTP

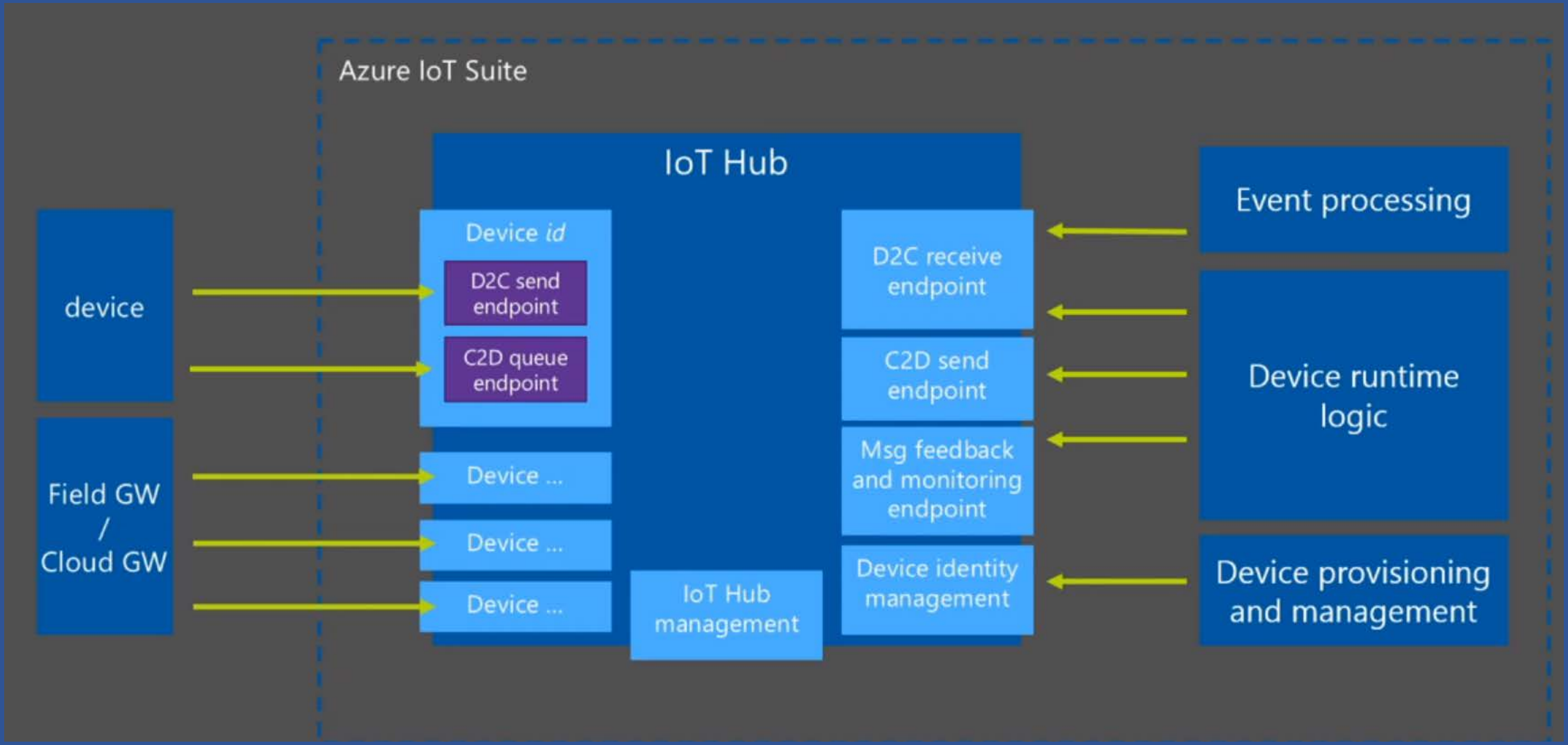
Designed for extensibility to custom protocols

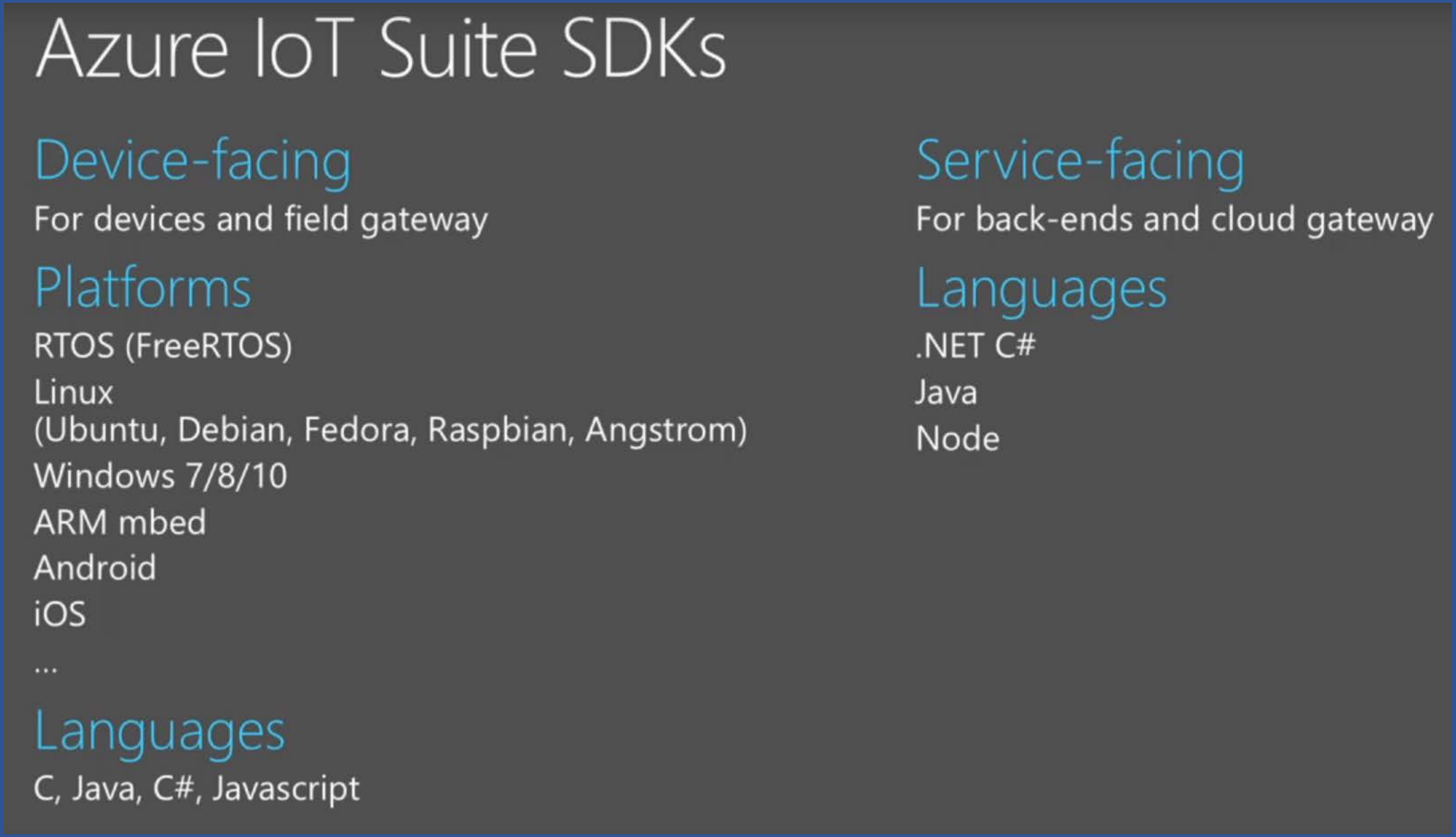
Multi-platform

Device SDKs available for multiple platforms (e.g. RTOS, Linux, Windows)

Multi-platform Service SDK.

IoT Hub Endpoint





Azure IoT Suite SDKs

Device-facing

For devices and field gateway

Platforms

RTOS (FreeRTOS)

Linux
(Ubuntu, Debian, Fedora, Raspbian, Angstrom)

Windows 7/8/10

ARM mbed

Android

iOS

...

Languages

C, Java, C#, Javascript

Service-facing

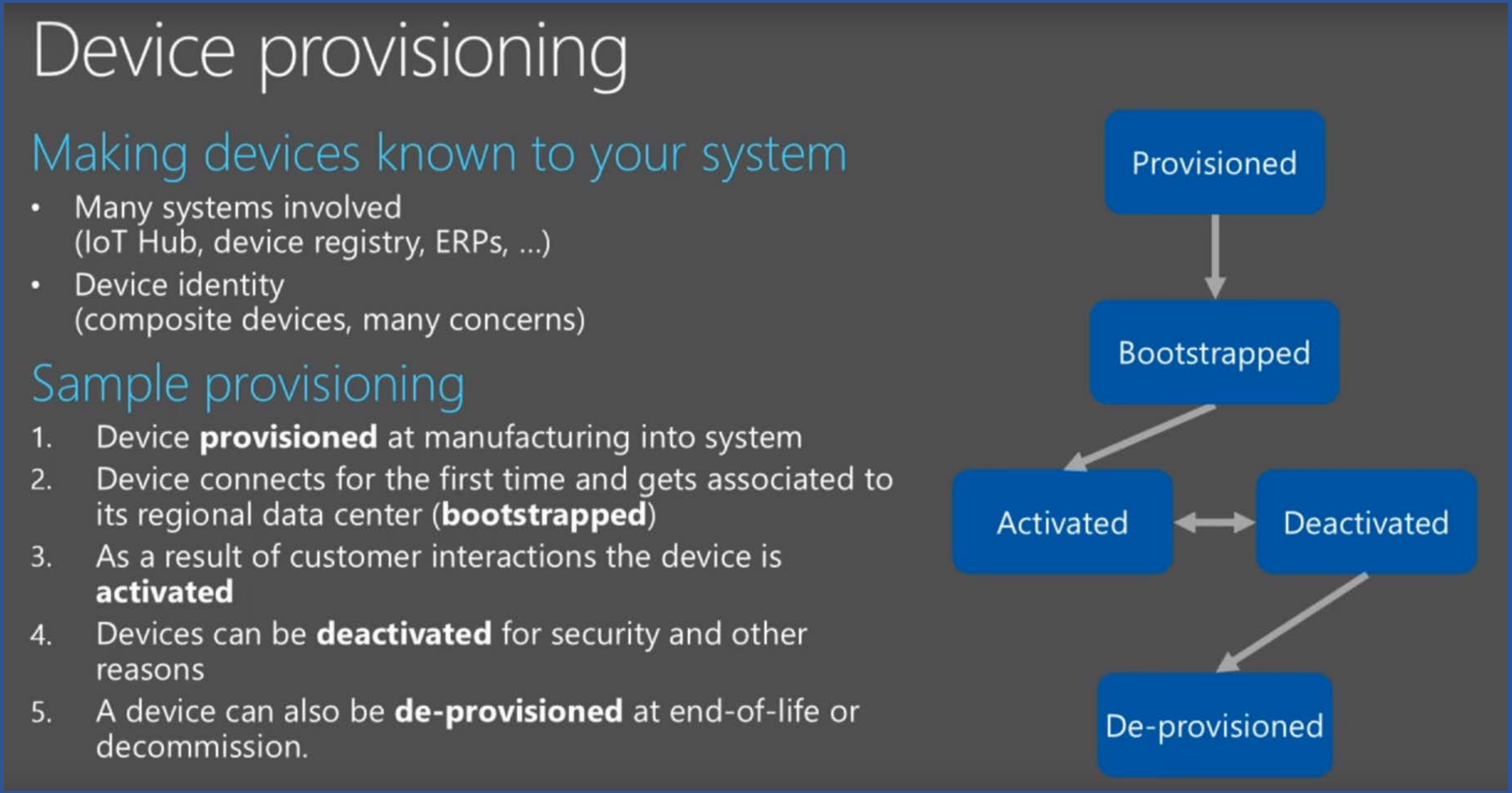
For back-ends and cloud gateway

Languages

.NET C#

Java

Node



Device provisioning

Making devices known to your system

- Many systems involved
  (IoT Hub, device registry, ERPs, ...)
- Device identity
  (composite devices, many concerns)

Sample provisioning

1. Device **provisioned** at manufacturing into system
2. Device connects for the first time and gets associated to its regional data center (**bootstrapped**)
3. As a result of customer interactions the device is **activated**
4. Devices can be **deactivated** for security and other reasons
5. A device can also be **de-provisioned** at end-of-life or decommission.

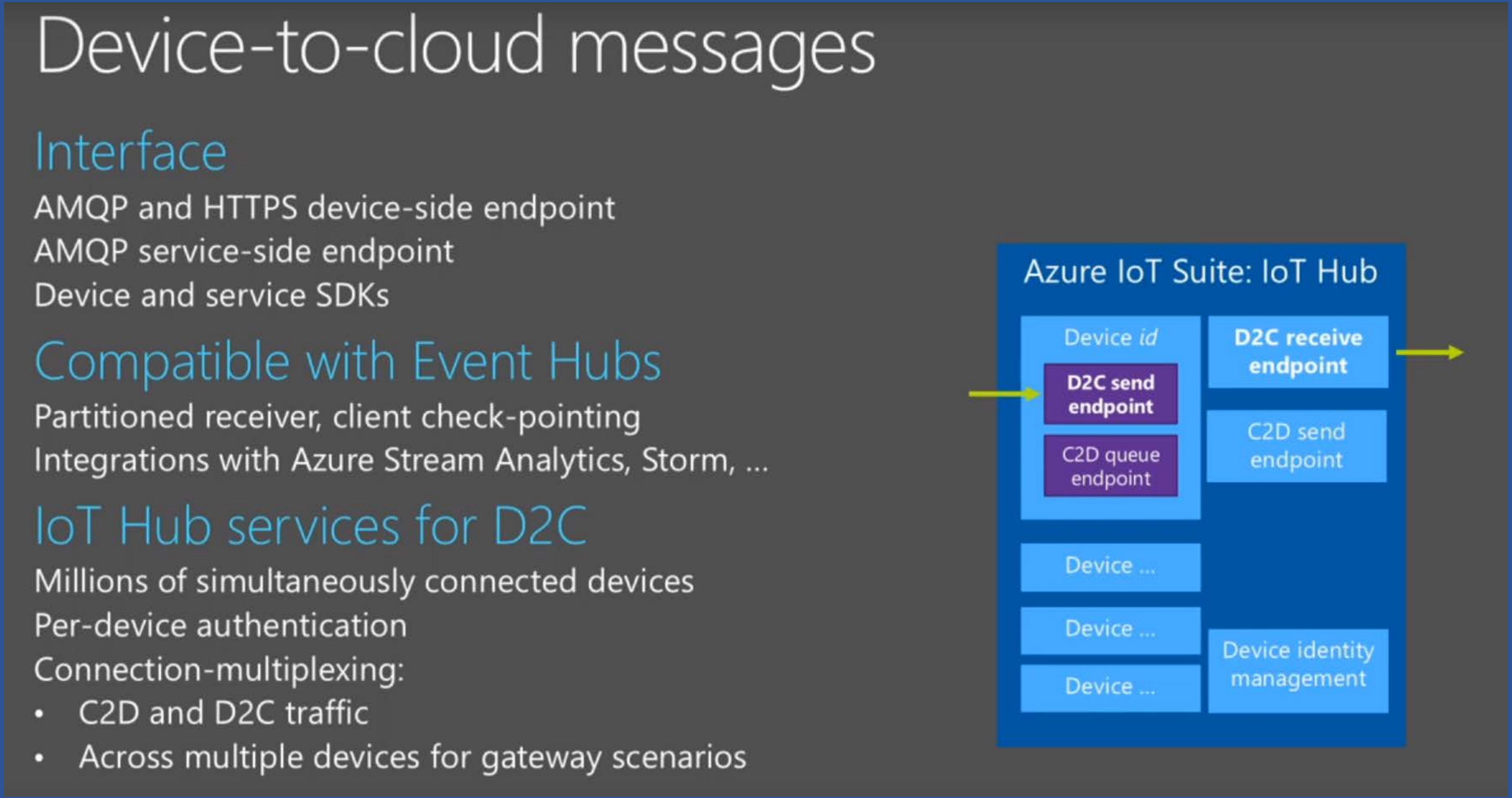
Provisioned

Bootstrapped

Activated

Deactivated

De-provisioned



Device-to-cloud messages

Interface

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

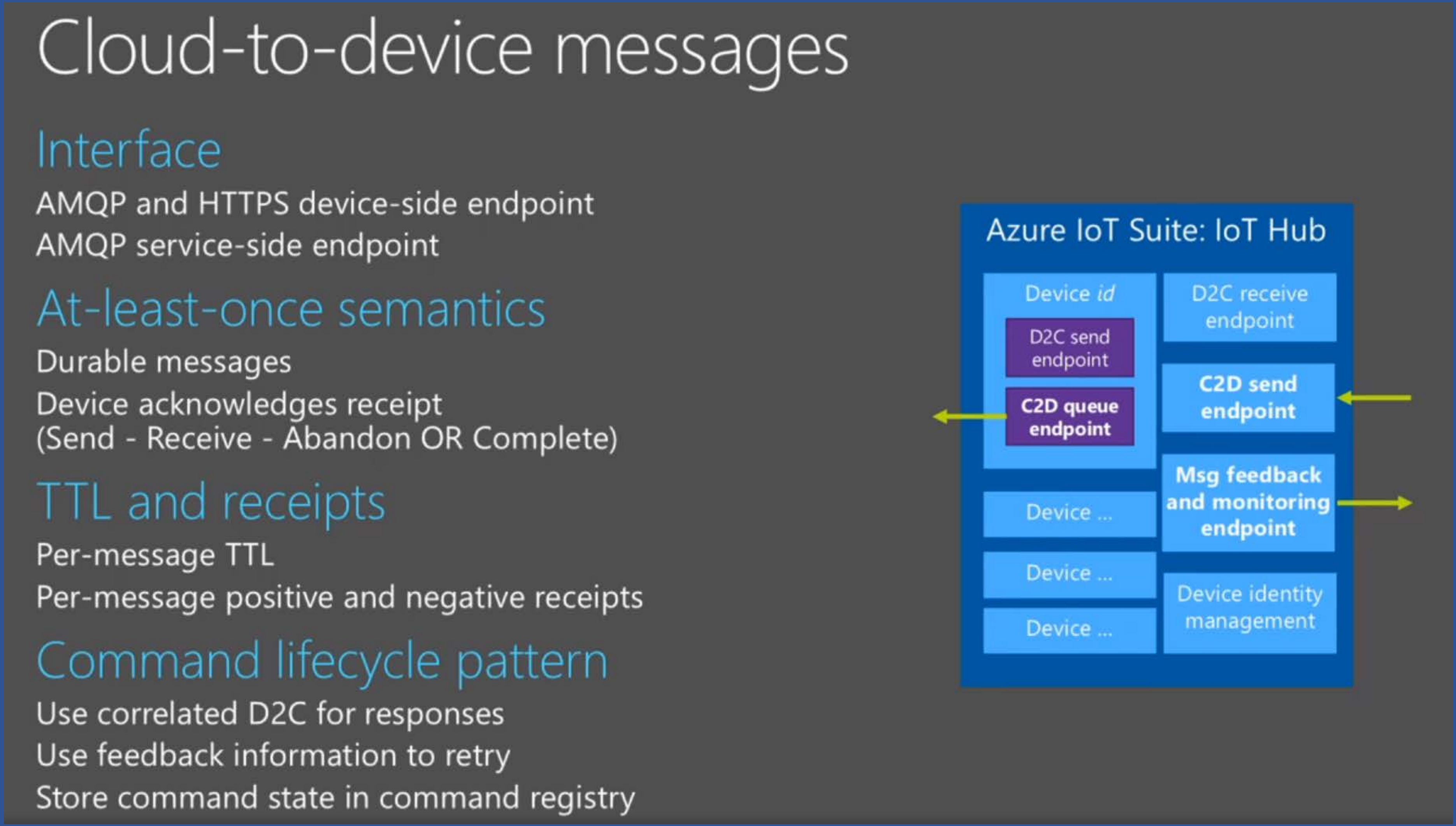
Compatible with Event Hubs

Partitioned receiver, client check-pointing
Integrations with Azure Stream Analytics, Storm, …

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

Azure IoT Suite: IoT Hub

Device *id*

D2C send endpoint

C2D queue endpoint

Device ...

Device ...

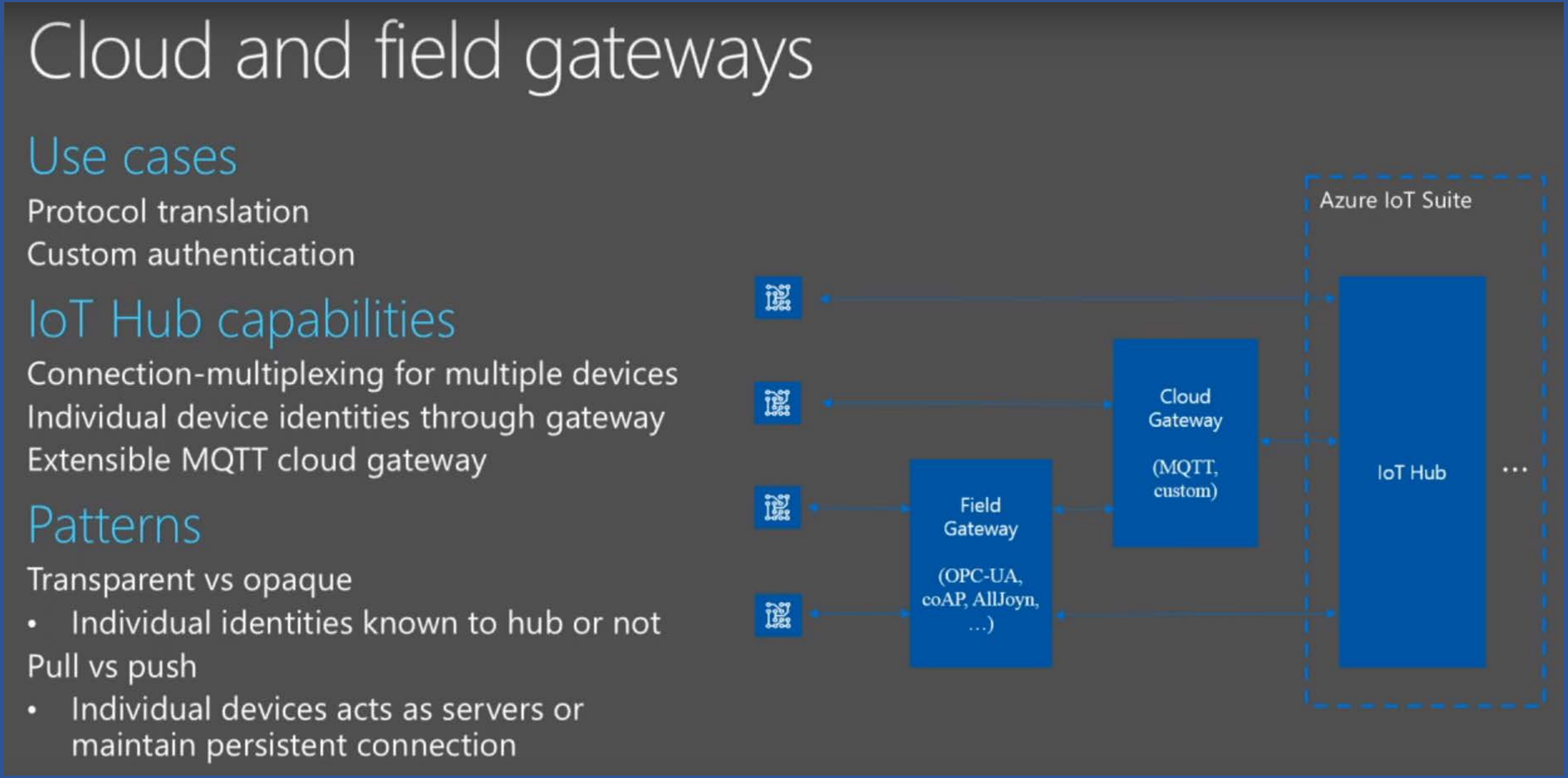
Device ...

D2C receive endpoint

C2D send endpoint

Msg feedback and monitoring endpoint

Device identity management



Cloud and field gateways

Use cases

Protocol translation

Custom authentication

IoT Hub capabilities

Connection-multiplexing for multiple devices

Individual device identities through gateway

Extensible MQTT cloud gateway

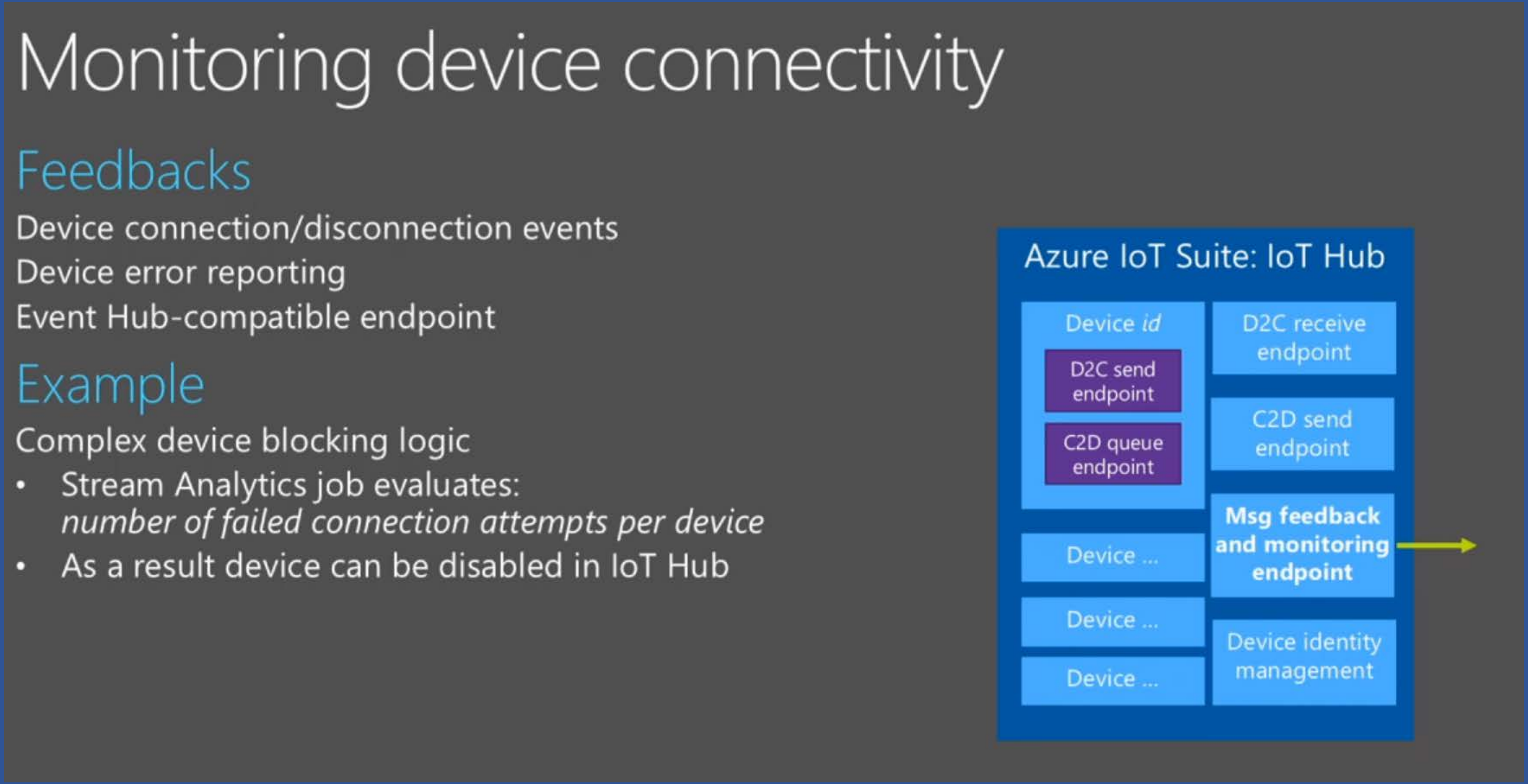
Patterns

Transparent vs opaque

- Individual identities known to hub or not

Pull vs push

- Individual devices acts as servers or maintain persistent connection



Monitoring device connectivity

Feedbacks

Device connection/disconnection events

Device error reporting

Event Hub-compatible endpoint

Example

Complex device blocking logic

- Stream Analytics job evaluates:
  number of failed connection attempts per device
- As a result device can be disabled in IoT Hub