Eclipse Hono. Connect. Command. Control.

Connect and control your IoT devices

Kai Hudalla, Chief Software Architect @ Bosch Software Innovations

Paolo Patierno, Senior Software Engineer @ Red Hat

11/10/2017

Who are we ?

@ppatierno

- Senior Software Engineer @ Red Hat
  - Messaging & IoT team
- Lead/Committer @ Eclipse Foundation
  - Hono, Paho and Vert.x projects
- Microsoft MVP Azure/IoT
- Blogger and speaker about distributed systems, messaging, IoT and embedded "world"

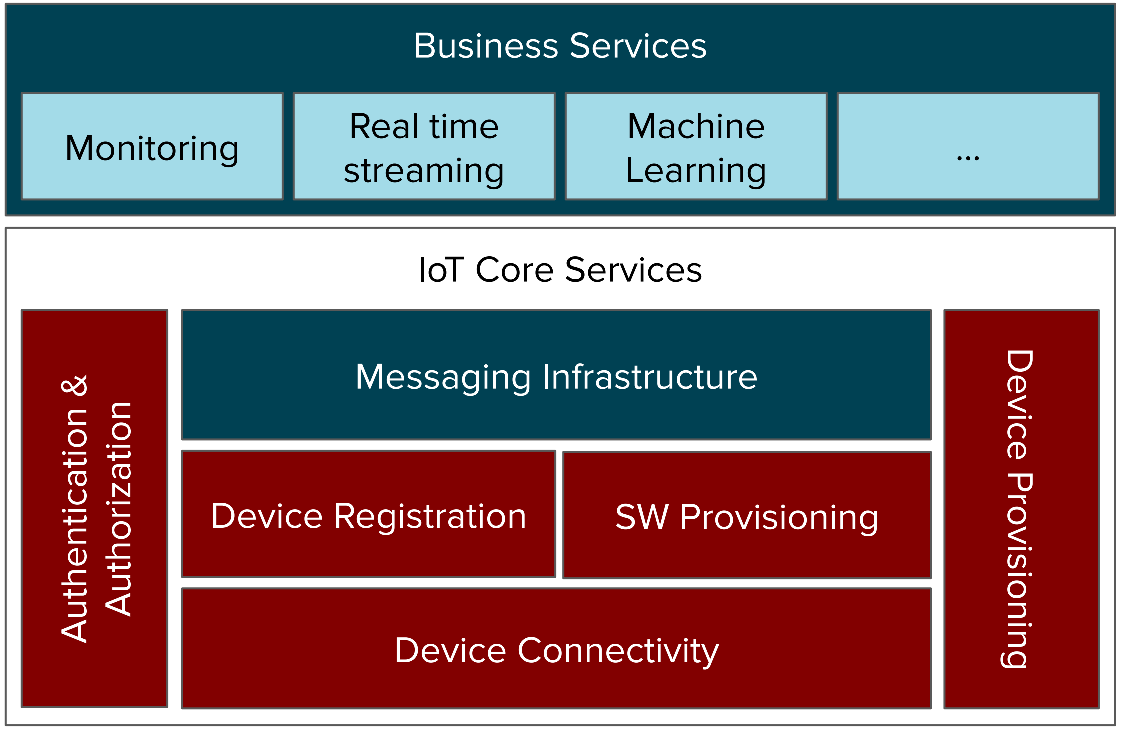
- Chief Software Architect @ Bosch SI
  - IoT Hub Team
- Lead/Committer @ Eclipse Foundation
  - Hono, Californium, Leshan projects

iot eclipse.org

Agenda

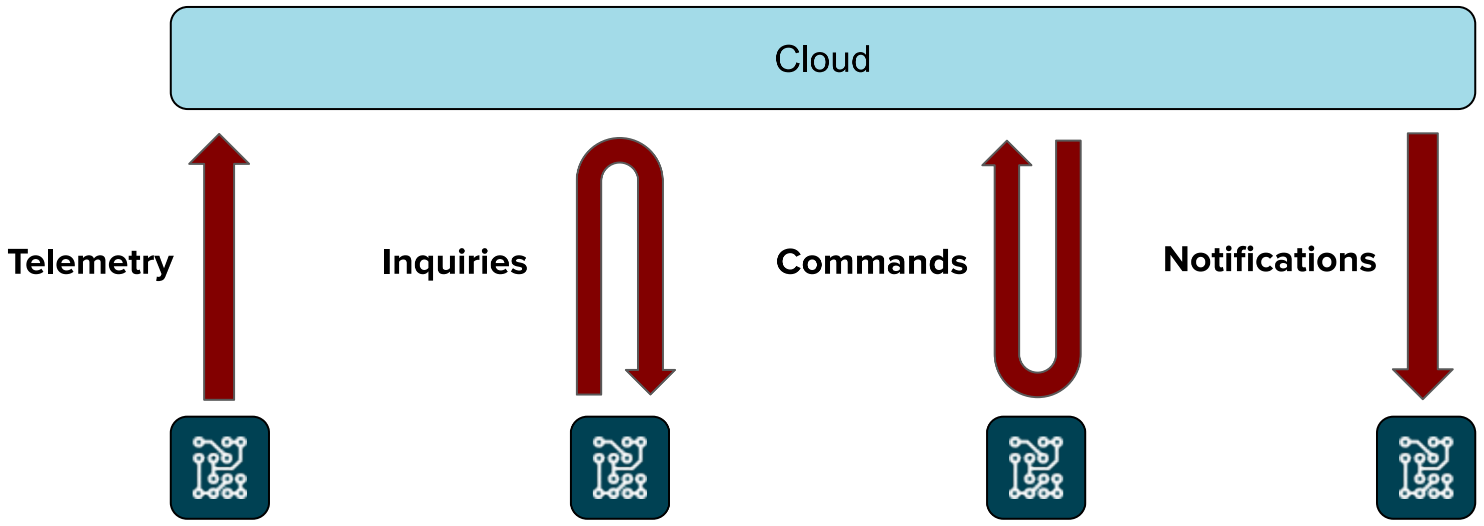
- The Internet of Things
    - The ecosystem
    - moving from closed to open source
- Please welcome to … Eclipse Hono
    - What is this ?
    - Goals & features
    - Architecture
    - Demo time !
    - Digging into the APIs
- How & where to deploy ?

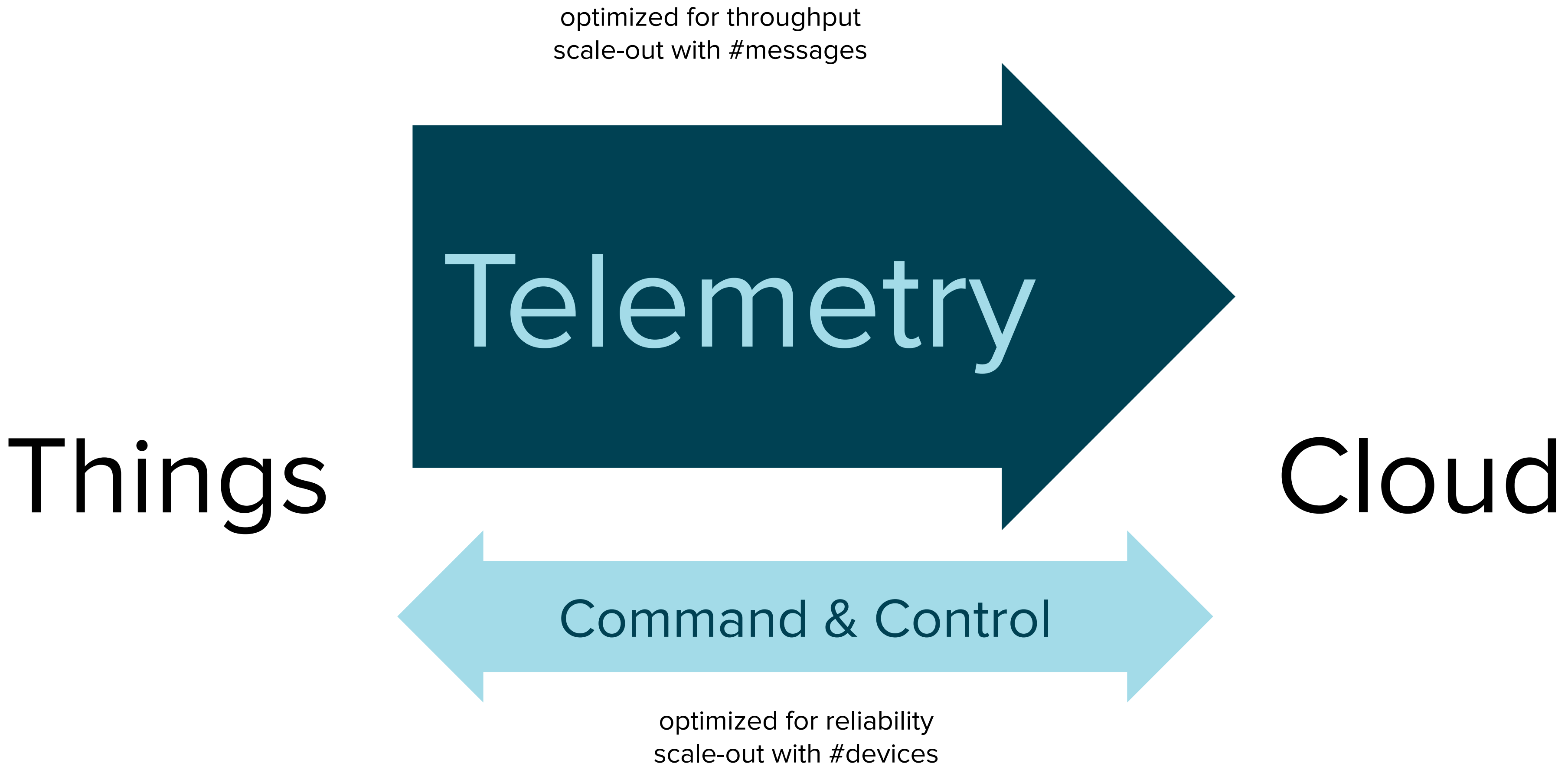
What makes an IoT platform ?



iot eclipse.org

IoT Communication Patterns




optimized for throughput
scale-out with #messages
Telemetry
Things
Cloud
Command & Control
optimized for reliability
scale-out with #devices

IoT : Communication Patterns

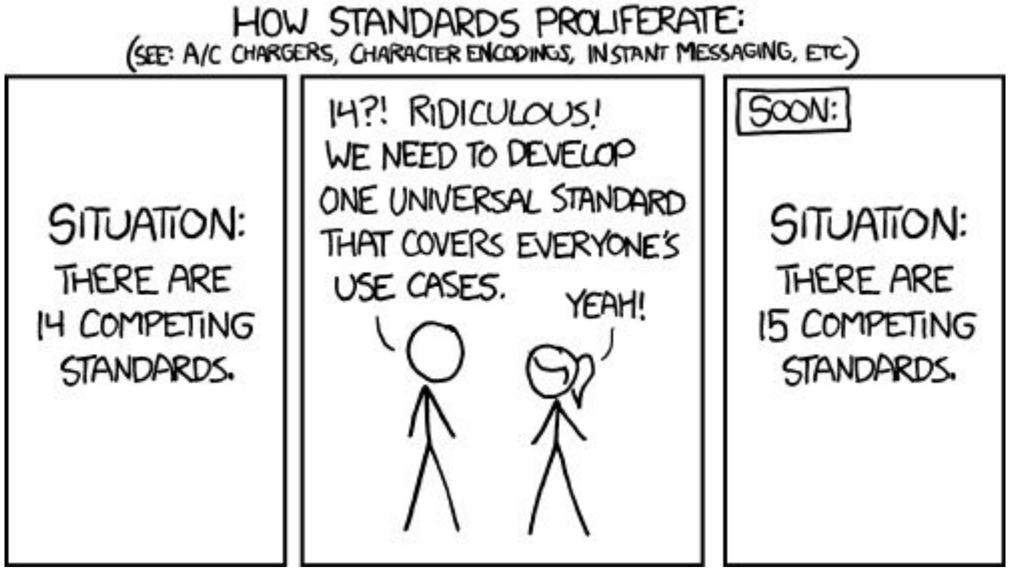
Messaging patterns & protocols

- **Telemetry** & **Notifications** are about …
  - …. messaging **publish/subscribe**
- **Commands** & **Inquiries** are about …
  - … messaging **request/response**
- Different protocols (AMQP, MQTT, HTTP, …) implement them in different way
  - As built-in support …
  - … or on top of it at application level
  - Read more on *"Strengths And Weaknesses Of IoT Communication Patterns"* *

* DZone IoT Guide : https://dzone.com/guides/iot-applications-protocols-and-best-practices

IoT : Interoperability

Open standards


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

Here are some of the 14 ...

AMQP 1.0

HTTP

STOMP

MQTT

XMPP

CoAP

IoT in the Cloud

Existing Offerings

- Microsoft Azure
  - IoT Hub
- Amazon Web Services
  - AWS IoT
- Google
  - IoT Core
- IBM
  - Watson IoT

Cloud provider limitations

- They are not open source !
- Freedom of choice
    - On-premise or in the cloud
    - Ability to choose which cloud
    - Open Standards protocols allows users to choose client freely
- Migrating from one to the other can be complex

Eclipse Hono

Connect. Command. Control.

- An Eclipse Foundation IoT project ...
    - Bosch and Red Hat as main contributors
- https://www.eclipse.org/hono/

Eclipse Hono

Connect. Command. Control.

- Open source IoT connectivity platform running on ...
  - Kubernetes
  - OpenShift
  - Docker Swarm
- On-premise & in the cloud
- Provided as a set of Docker images

Eclipse Hono

Goals

- Interact with devices regardless of communication protocol
- Support large number of devices (10^6)
- Don't trade in throughput for reliability
- Support arbitrary device protocols (MQTT, HTTP, CoAP, AMQP 1.0, etc)
- Integrate with existing infrastructure

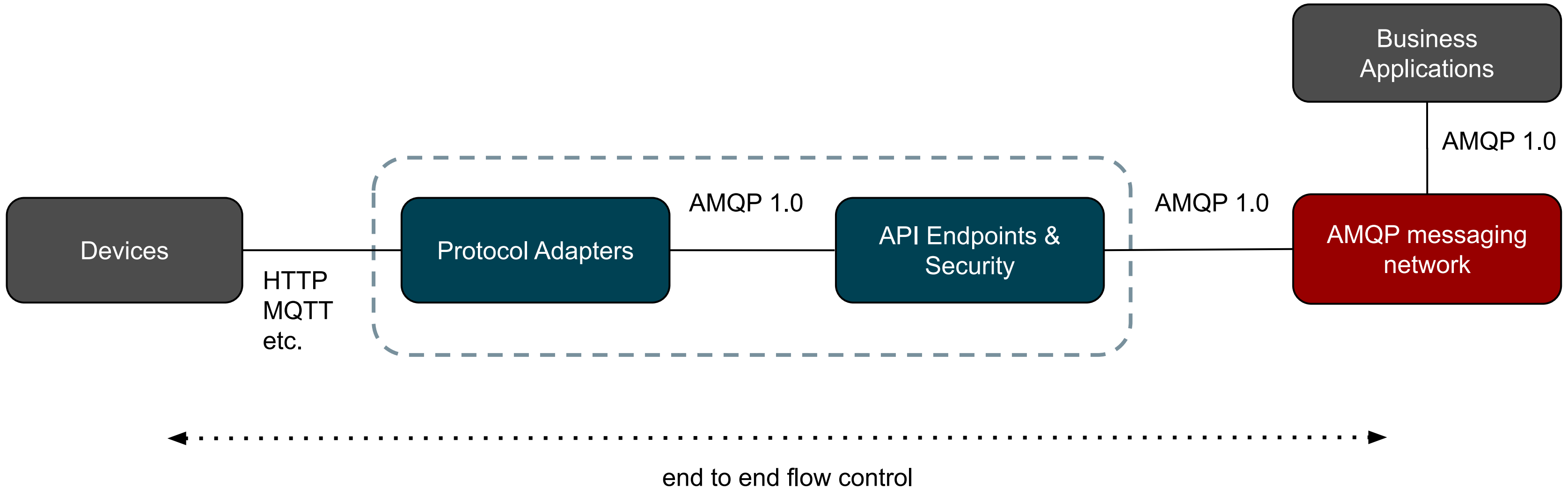
Eclipse Hono

Features

- Uniform APIs for interacting with devices (regardless of protocol)
- Out-of-the-Box Connectivity for Devices supporting MQTT or HTTP
  - Additional protocols by implementing *custom* Protocol Adapters
- Device-level Authentication
- Tenant based Security Model
- Support for arbitrary messaging infrastructure (AMQP 1.0 based)
- Horizontal Scalability
- End-to-End Flow Control

Eclipse Hono

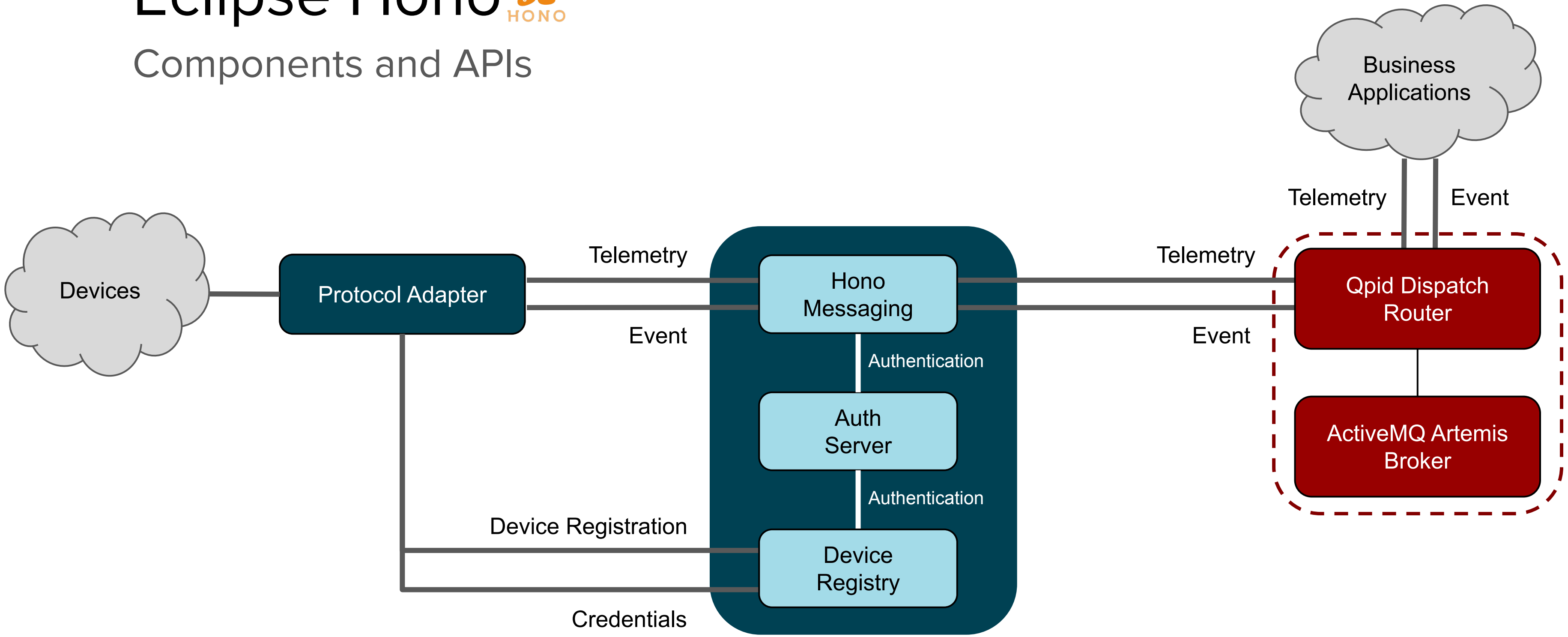
Building Blocks & Scope



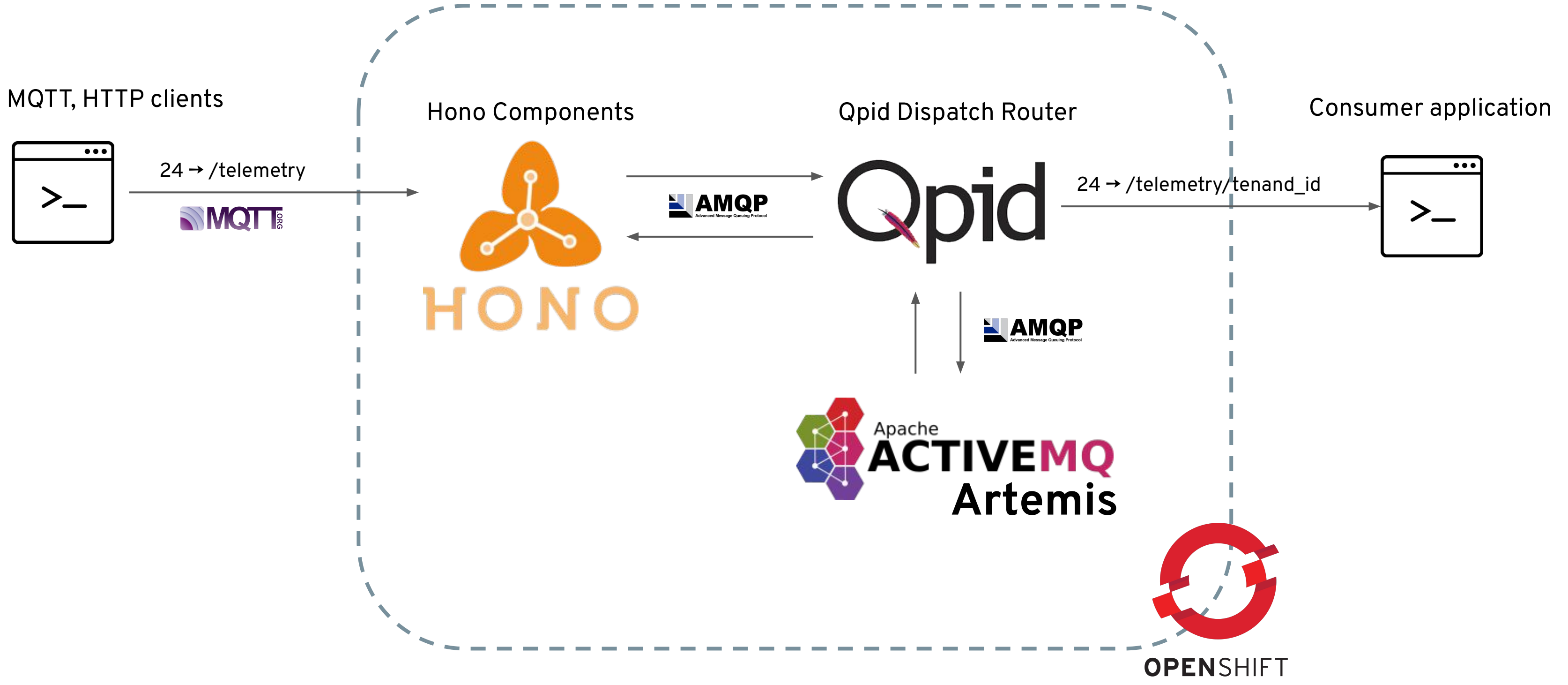
- - - - - Hono

Eclipse Hono

Components and APIs



DEMO



iot eclipse.org

Eclipse Hono

APIs

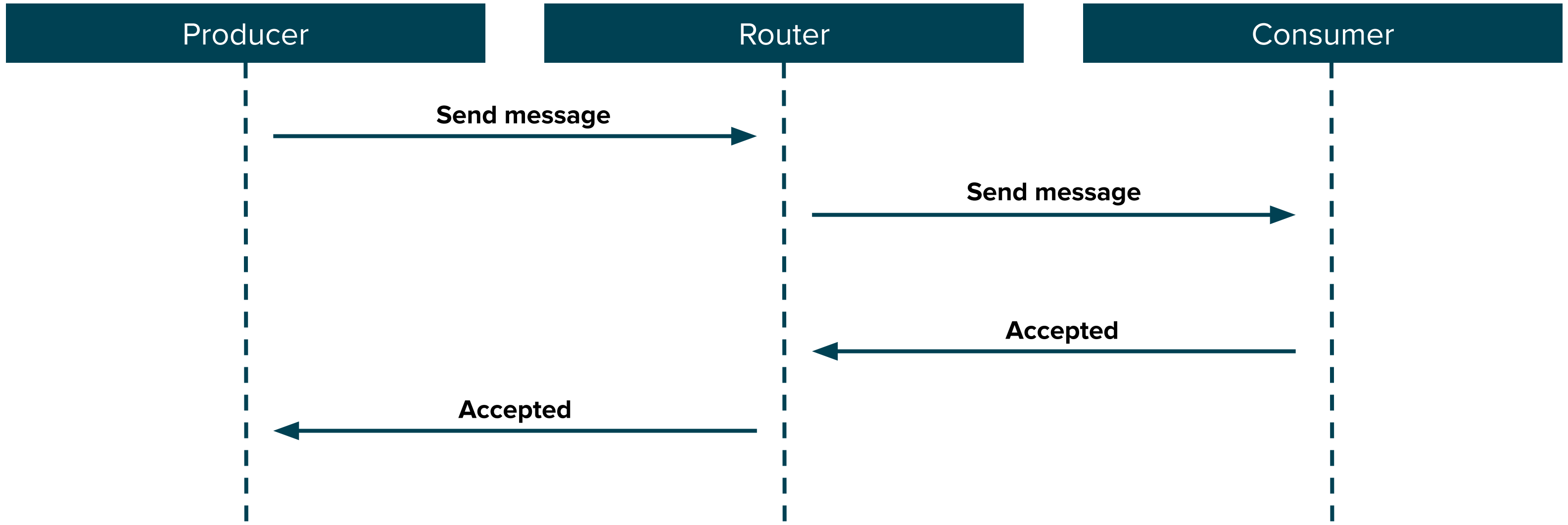
- Telemetry & Event
- Device Registration
- Credentials
- Authentication
- Command & Control (not available in 0.5 release)

Eclipse Hono

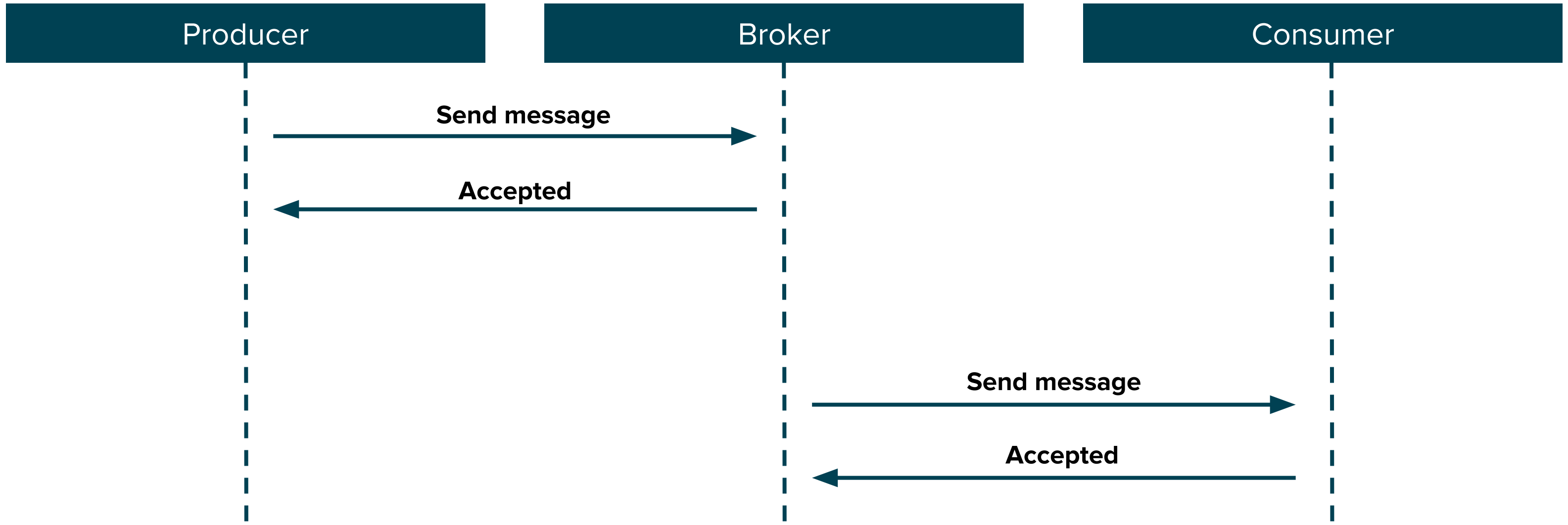
Telemetry & Event

- used by devices to **send data/events downstream**
- leverages on **“direct messaging”** …
    - Telemetry
    - Devices can send data only if consumers are online
    - No broker involved
- … **“store and forward”**
    - Event
    - Broker for storing event with a “ttl” eventually
- consumers receive data published by devices belonging to a particular tenant

Routing Telemetry Data

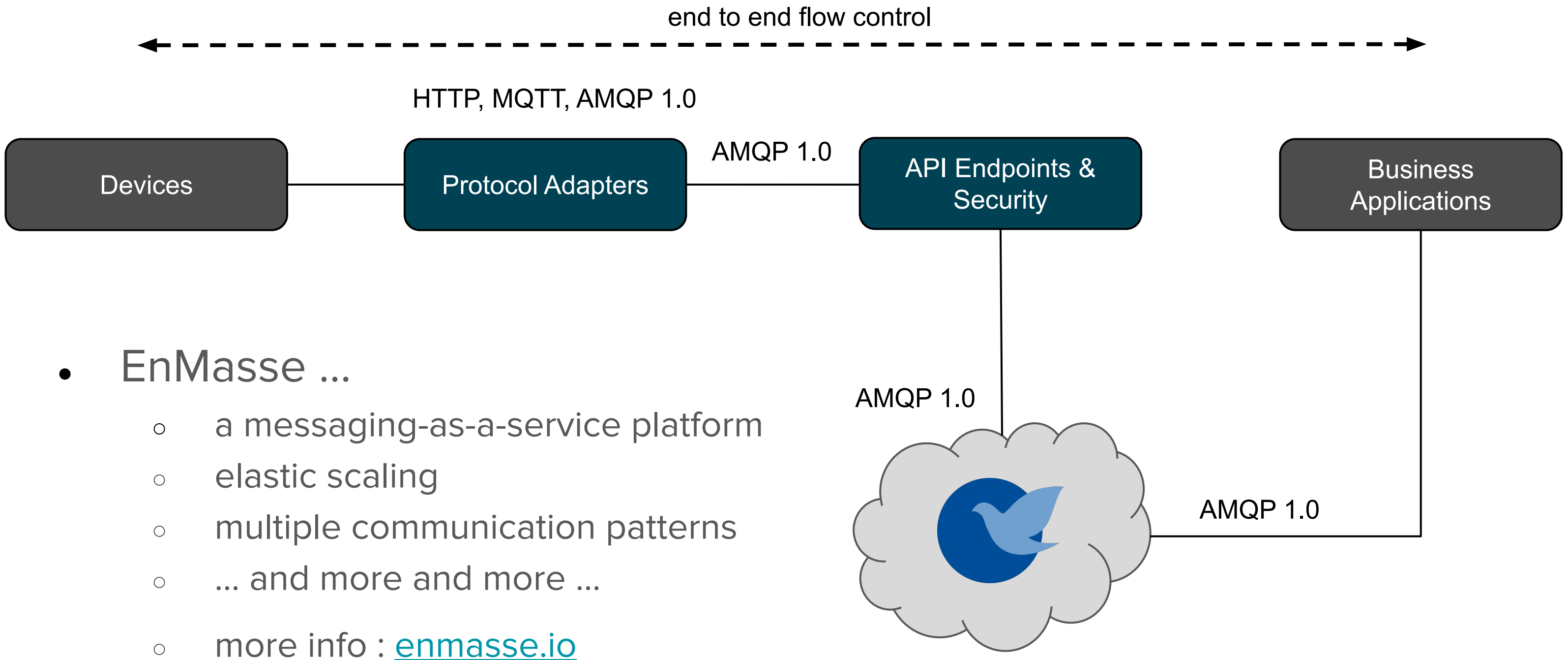


Brokering Events



Scaling out the messaging network

Connect. Command. Control



- EnMasse ...
  - a messaging-as-a-service platform
  - elastic scaling
  - multiple communication patterns
  - ... and more and more ...
  - more info : enmasse.io

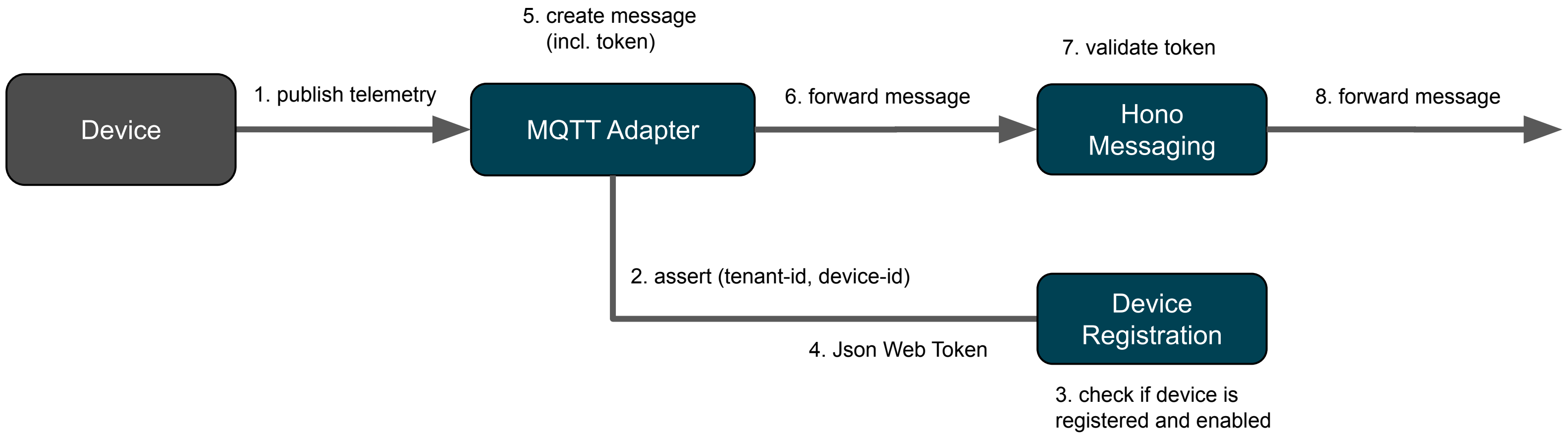
Eclipse Hono

Device Registration

- used to make Hono **aware of devices** that can/will connect to the service
- solutions/consumers may use the API to get information about devices
- operations
    - **assert status** (mandatory)
    - register, deregister, get information (optional)
- where a system managing device identities is already in place
    - ...implement this API as a *facade* to the existing system

Eclipse Hono

Registration Assertion Flow



Assertions are *cached* by the adapter per device connection

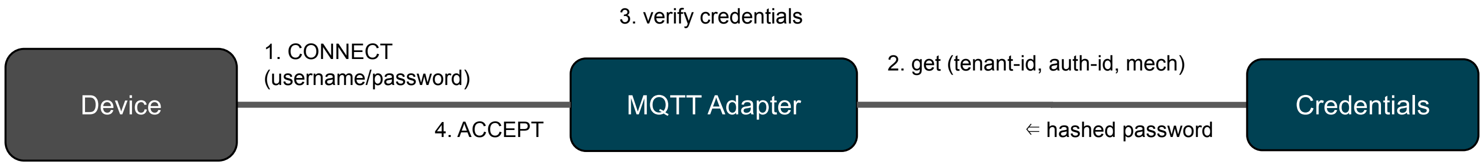
Eclipse Hono

Credentials

- supports the process of authenticating devices
- used by **protocol adapters** to retrieve credentials used to authenticate **devices** connecting to the adapter (MQTT, HTTP, …)
- different types of credentials
  - psk, hashed password, public key, …
- operations
  - **get** (mandatory)
  - add, update, remove (optional)
- where a system for doing this is already in place …
  - implement this API as a *facade* to the existing system

Eclipse Hono

Device Authentication Flow



Verification of credentials is *always* responsibility of Protocol Adapter

Eclipse Hono

Authentication

- handle authentication between components (Protocol Adapters, Hono Messaging, …)
- used by clients/components for getting a **token** asserting …
  - subject's identity
  - granted authorities
- services use the token to make authorization decisions on a client's request to read or write from/to a resource or to invoke a certain operation
  - i.e. Hono Messaging checks if an Adapter may write telemetry data
- Where an **identity management system** is already in place (e.g. Keycloak) …
  - implement this API as a *facade* to the existing system

Eclipse Hono

Command & Control

- used by applications to **send commands to devices**
- command execution can be “just in time” or “deferred”
  - **just in time** : command already executed, the response from device contains the result
  - **deferred** : command not executed yet, the response from device specifies it’s accepted; for long running operations the result will be provided later

IoT : how to deploy ?

- “On premise” …
  - … maybe for a not so big solution
  - … ingesting few data and handling few devices
- “Cloud” …
  - … needs for more scalability
  - … don’t want to manage the infrastructure
- “Hybrid” …
  - … needs for processing at the edge
  - … needs for not making sensible data public

Azure Container Service

- A containers hosting solution
- Scale and orchestrate using …
  - Kubernetes
  - Docker Swarm
  - DC/OS
- Deploying a cluster using Azure CLI / portal
  - Resource group with VMs, load balancer, …
- Managing directly your preferred “orchestrator”
  - ACS provides you “only” the infrastructure

Azure & OpenShift

- OpenShift Origin
  - the upstream open source project
- OpenShift Container Platform
  - the Red Hat productized version
  - enterprise grade container platform

Amazon EC2

- Spinning up virtual machines …
  - for making a cluster
- Providing …
  - Docker and …
  - … Kubernetes or OpenShift …
  - … or just Docker using Swarm mode

+

Resources

- **Eclipse Hono** : https://www.eclipse.org/hono/
- **Eclipse IoT** : https://iot.eclipse.org/
- **Qpid Dispatch Router** : http://qpid.apache.org/components/dispatch-router/
- **ActiveMQ Artemis** : https://activemq.apache.org/artemis/
- **EnMasse** : http://enmasse.io/
- **Azure Container Service** : https://azure.microsoft.com/en-us/services/container-service/
- **OpenShift on Azure** : http://aka.ms/openshift

Thank you ! Questions ?