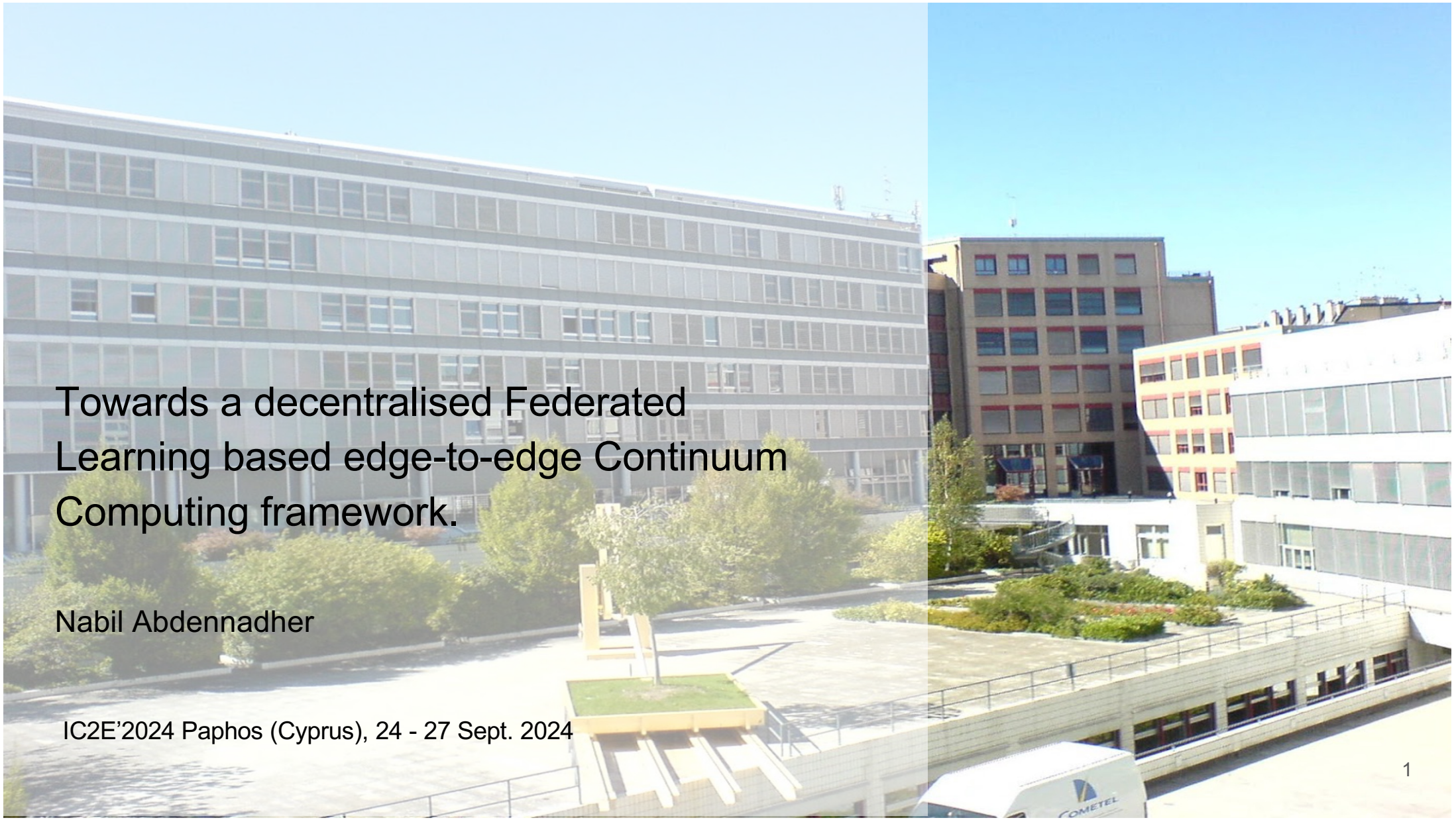


Towards a decentralised Federated Learning based edge-to-edge Continuum Computing framework.

Nabil Abdennadher

IC2E'2024 Paphos (Cyprus), 24 - 27 Sept. 2024

Raoul Dupuis

Francisco Mendonça

Mohamad Moussa

Nabil Abdennadher

Before starting …

https://lsds.hesge.ch/tutorial/

Before starting …

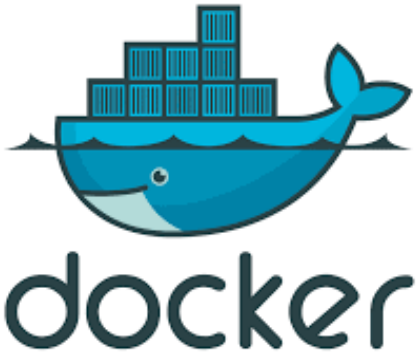
Towards a decentralised Federated Learning based edge-to-edge **Cloud Continuum** framework

Today, I'll have more questions than answers !!

Before starting …

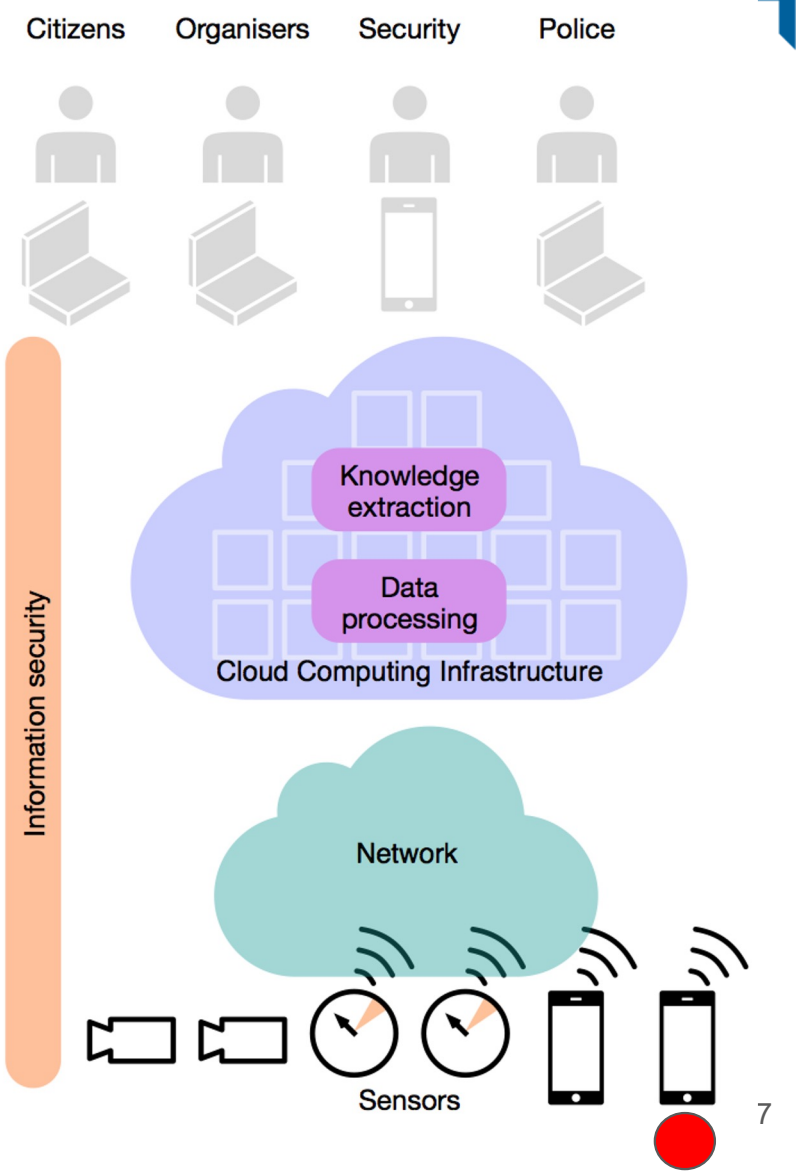
Do not hesitate to interact, comment, correct and ask questions !

Before starting …



The context

- Thousands of connected IoT sensors deployed at a large scale.
- Several applications are emerging.
  - Data sensitive, Compute intensive
  - Self-adaptive/Context-aware
- Centralised IoT platforms are ill equipped to cope with the huge quantity of collected data.
- … And here where the Edge-to-Cloud and Cloud Continuum come in …

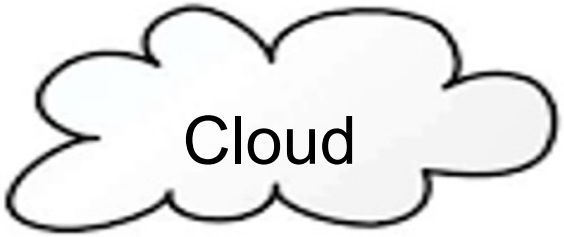


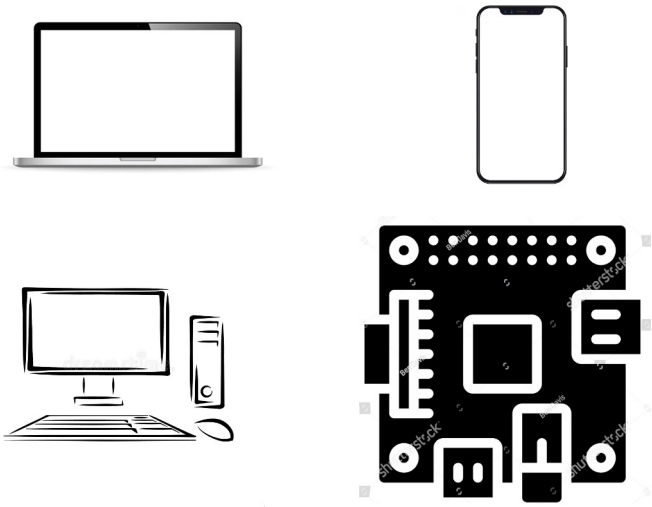
What is the pain?

And here where the Cloud Continuum comes in …

- The Edge is the solution ... but no one is perfect …
- Because of edge computing, we need now a new approach to:
  - design/develop,
  - deploy, monitor, control executions of …

… large scale distributed applications on HPC, Cloud, mobiles, edges, IoT devices, etc..

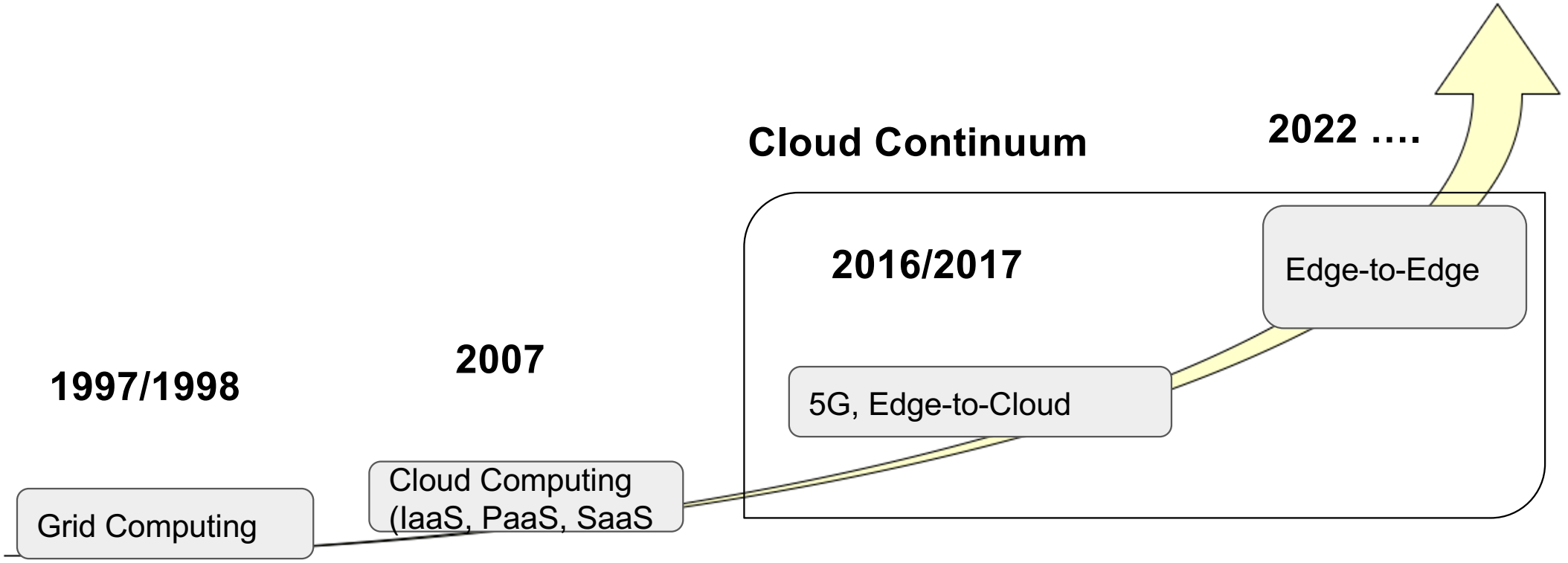




Plan

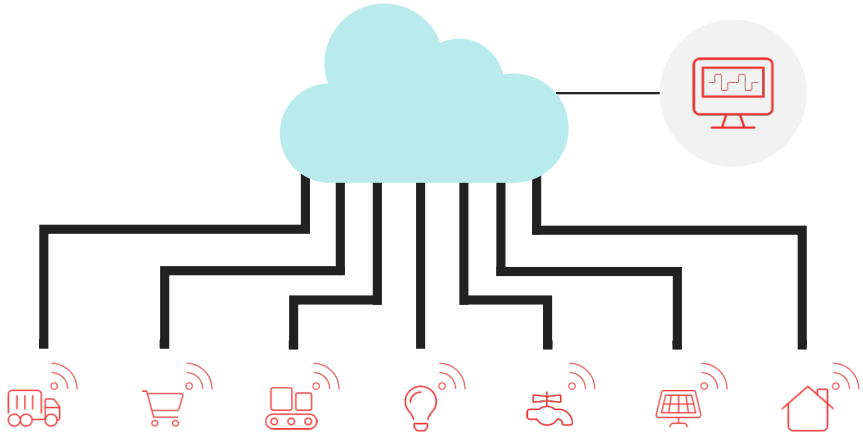
1. Why/What is Edge Computing and Cloud Continuum ?
2. Targeted applications (self-adaptive/context-aware)
3. Use-cases
4. Cloud Continuum (Edge-to-Cloud) solutions
5. Towards a decentralised Federated Learning based edge-to-edge Cloud Continuum framework.
6. The Smart Grid Application

A bit of history …

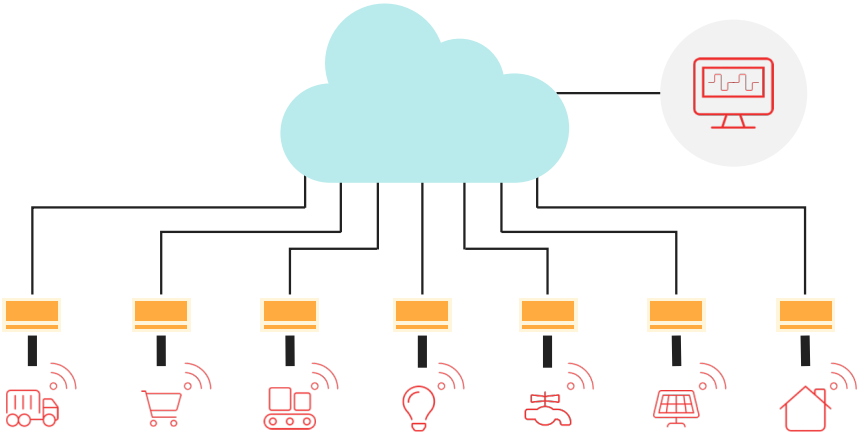


Why the edge is the solution?

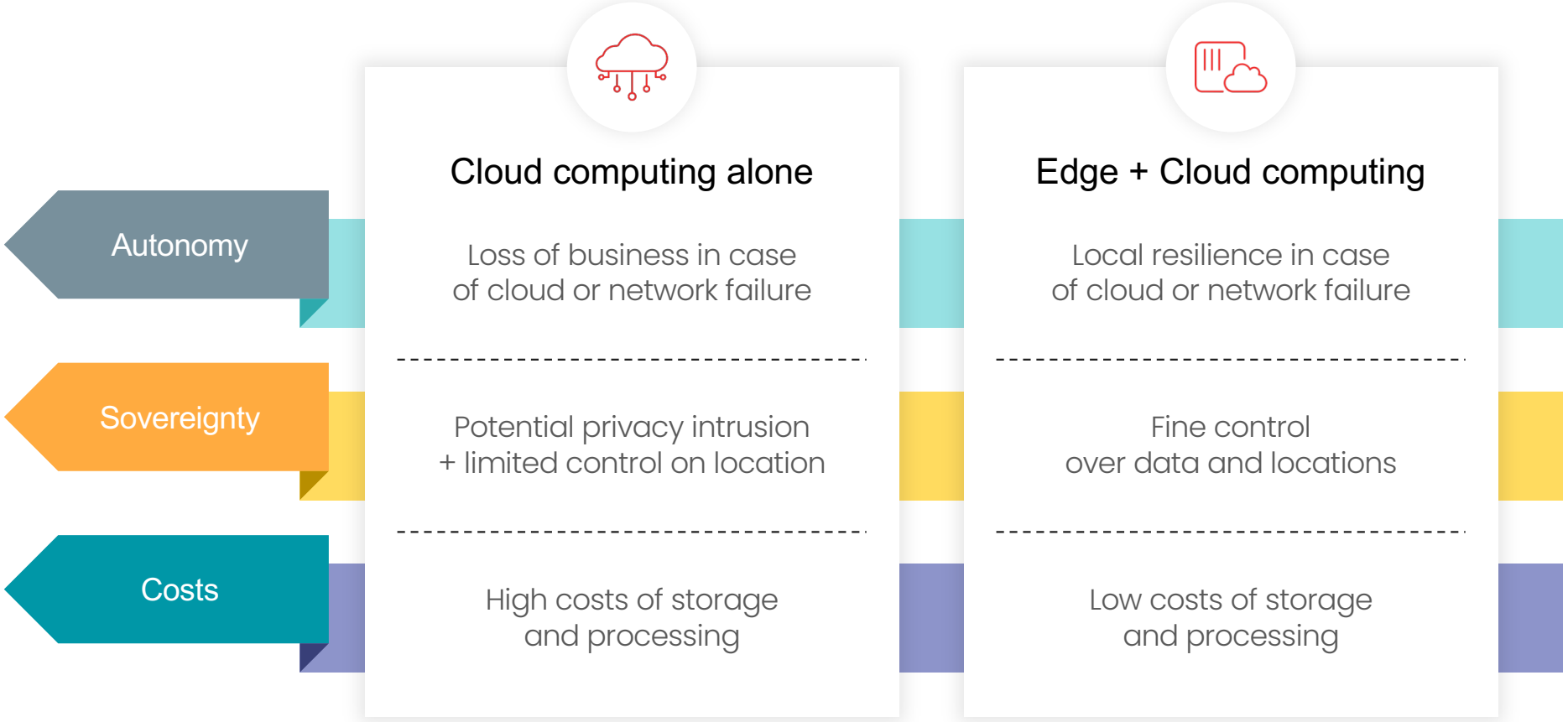
IoT straight to Cloud... **A bad idea**	Potential privacy instrusion
High cloud costs	High carbon impact
High latency	At scale... This will fail !!



With the edge	No privacy instrusion
Low cloud costs	Lower carbon impact
Low latency	Autonomy at scale... This rocks !!



Edge + cloud are the solution for three challenges.



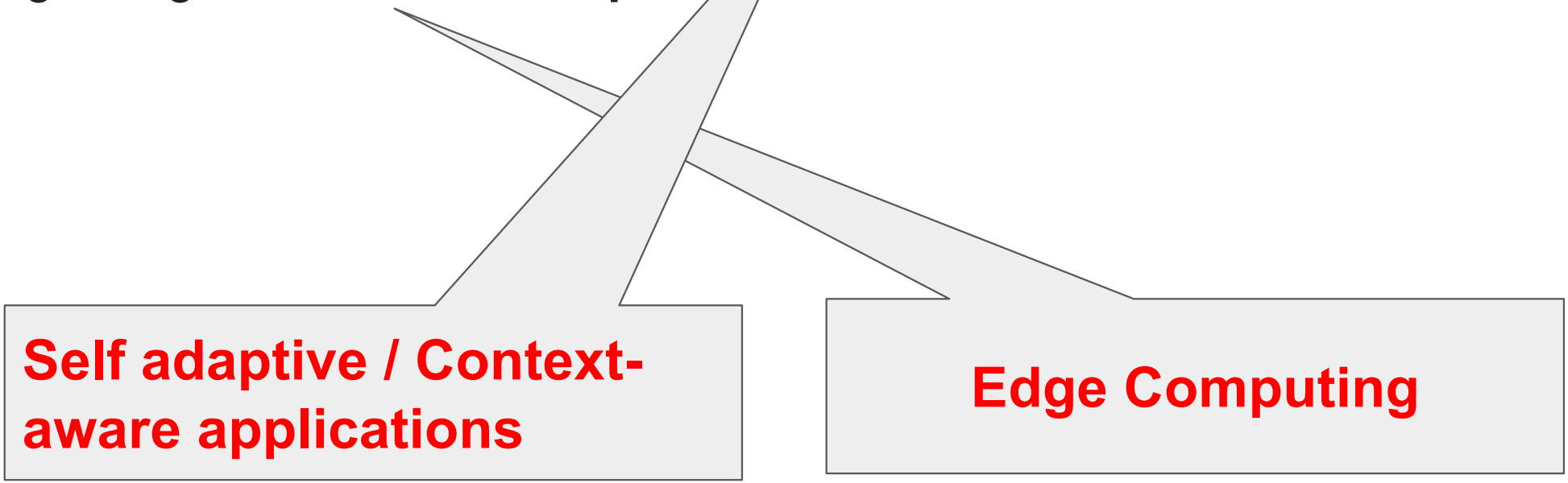
Why Edge Computing ?

If the answer is **yes** to one or more of these questions, then you need an **edge computing**:

- Is there a need for **near-real-time action** on data generated by sensors?
- Is the **data** generated **too big to transfer** to the cloud?
- Is the **internet link** between the sensors/actuators and the cloud **unreliable** ?
- Is there a **privacy/security issue** with transferring or processing the data in the cloud ?

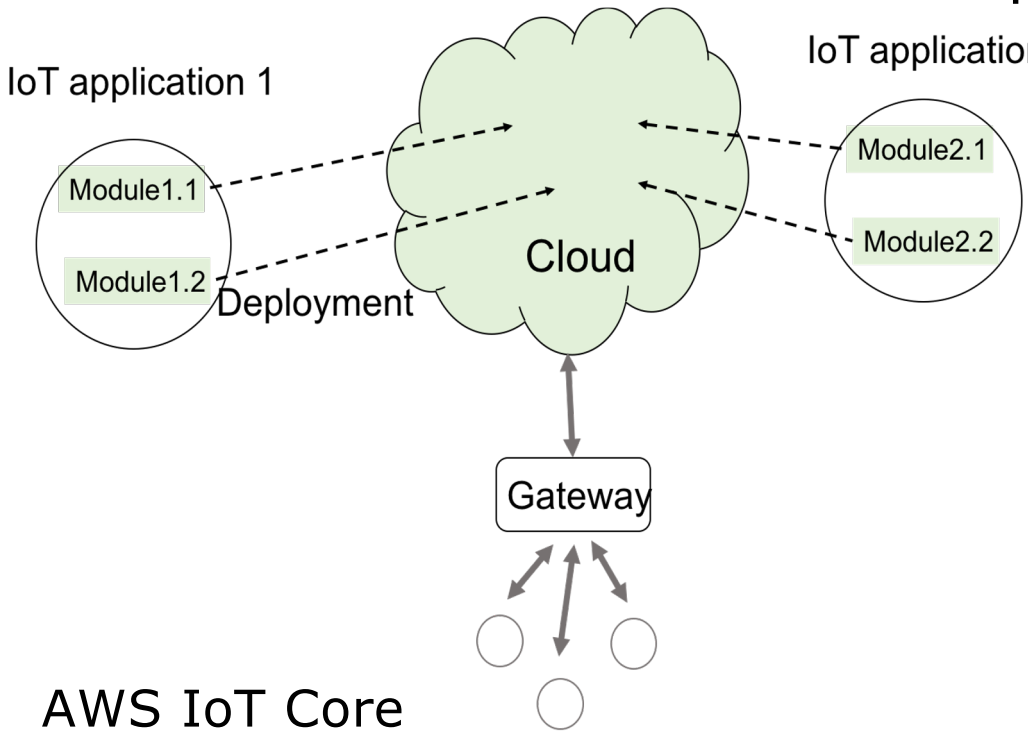
What is Cloud Continuum? Attempted definition

It is the current trend of developing, deploying and running **highly distributed, computing intense and data-sensitive applications** on a set of IT resources ranging from high density compute and storage to lightweight **embedded computer**.



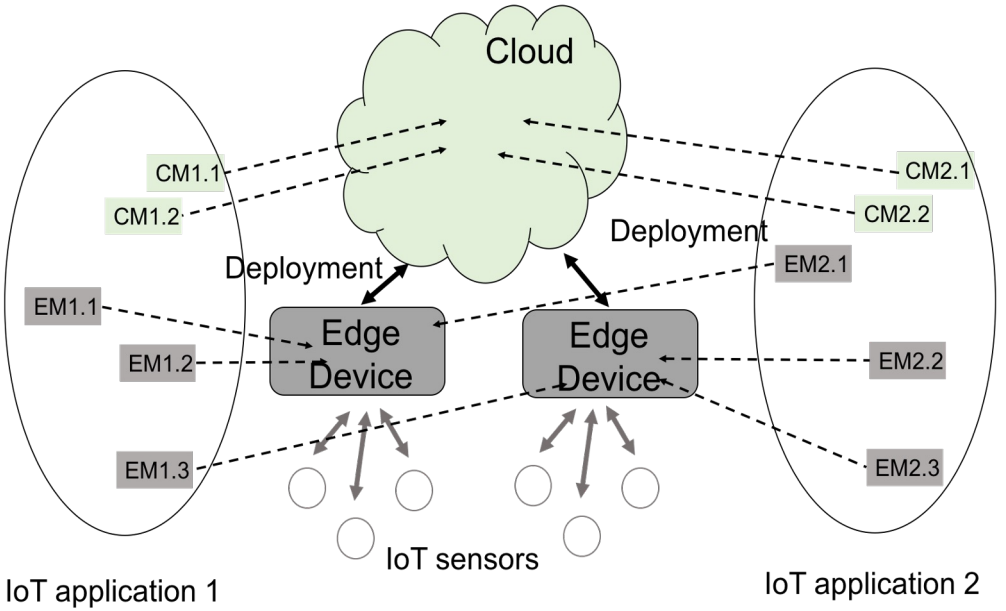
Cloud Continuum is also about software engineering

From a centralised to a decentralised paradigm



- AWS IoT Core
- Azure IoT hub
- Cloud IoT Core (Google)

IoT infrastructure



Edge-to-Cloud Computing

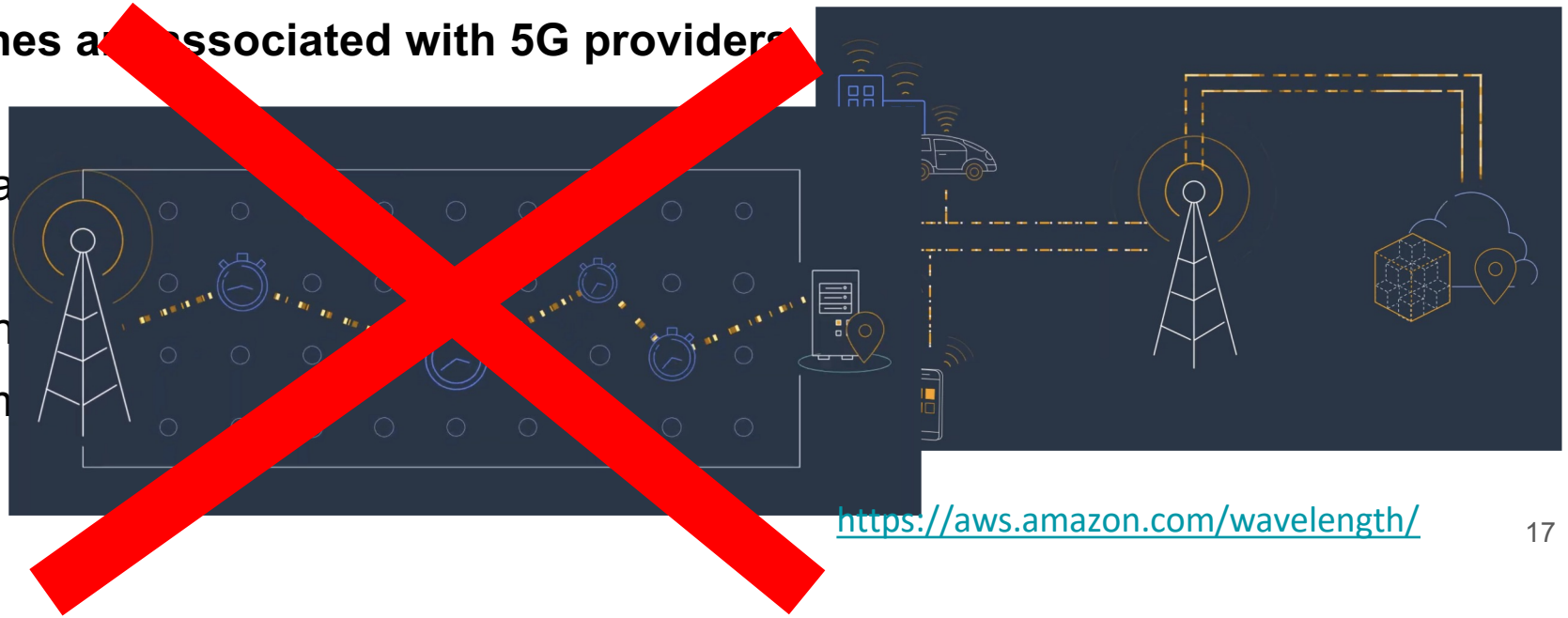
“Side effects” of Edge Computing

Strike the balance between:

- “keeping data at the edge” and “bringing it into a central cloud”
- “sophisticated algorithms in the cloud” and “lightweight analytical processes” in the edge

AWS wavelength zones

- Infrastructure offering optimised for mobile edge computing applications.
- Application traffic from 5G devices can reach application servers running in Wavelength Zones without leaving the local telecom network
- Take full advantage of the latency and bandwidth benefits of modern 5G networks.
- **Wavelength zones a[illegible]ssociated with 5G provider[illegible]**
- Use-cases
  - Connected ca[illegible]
  - AR/VR
  - ML assisted h[illegible]
  - Real time gan[illegible]
  - …

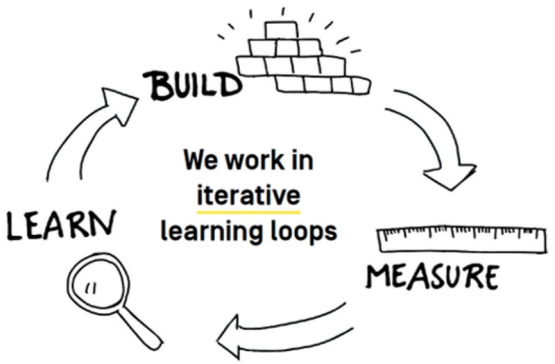


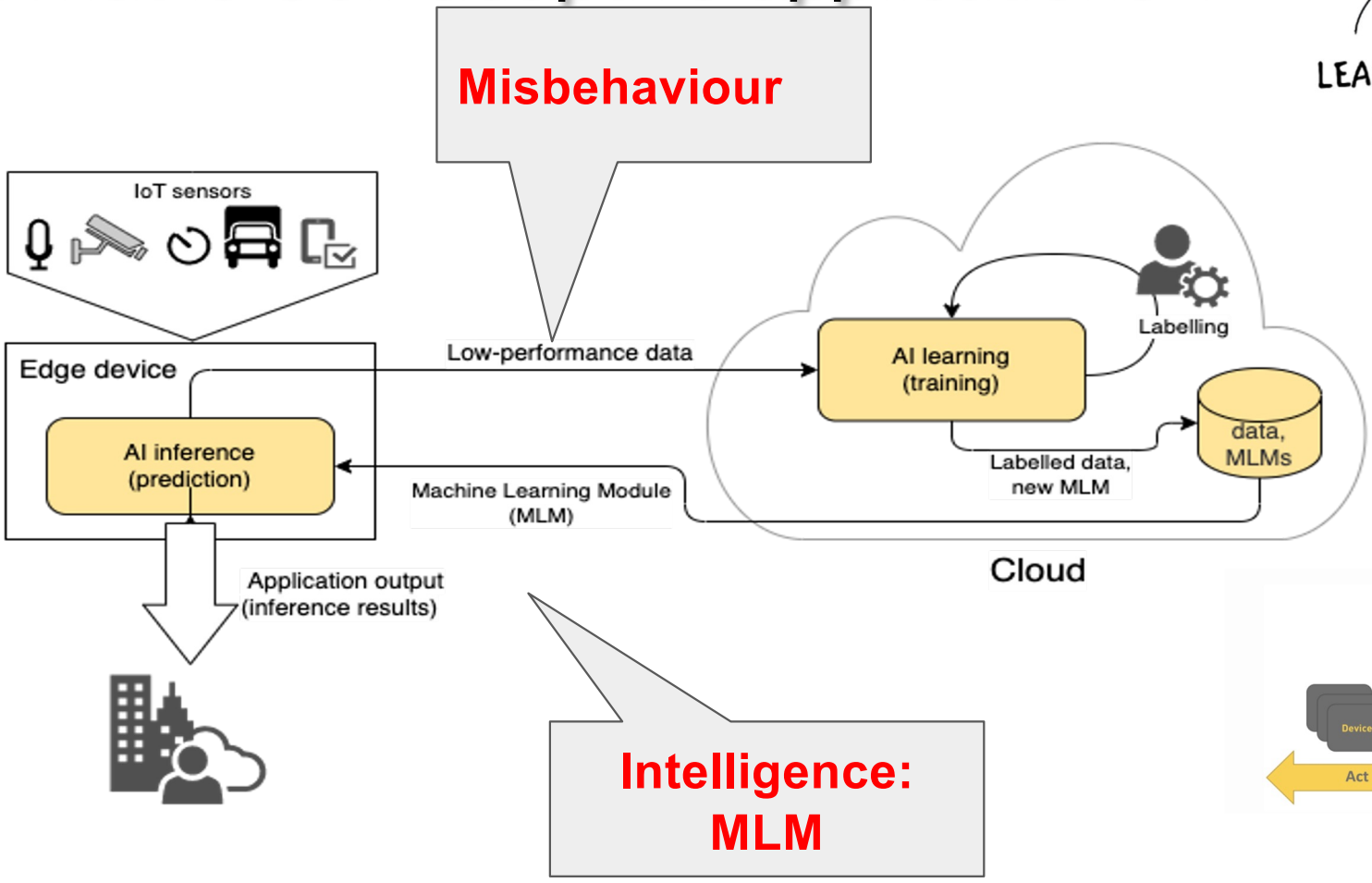
https://aws.amazon.com/wavelength/

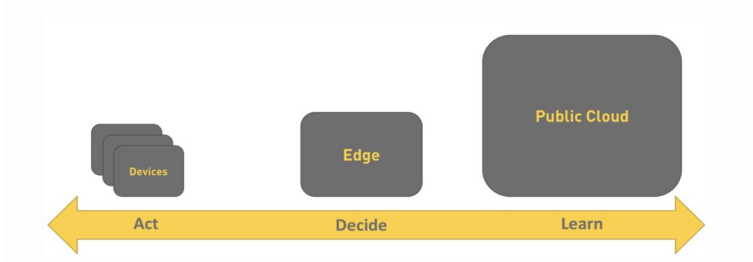
Plan

1. Why/What is Edge Computing and Cloud Continuum ?
2. **Targeted applications (self-adaptive/context-aware)**
3. Use-cases
4. Cloud Continuum (Edge-to-Cloud) solutions
5. Towards a decentralised Federated Learning based edge-to-edge Cloud Continuum framework.
6. The Smart Grid Application

What is Self-adaptive applications ?





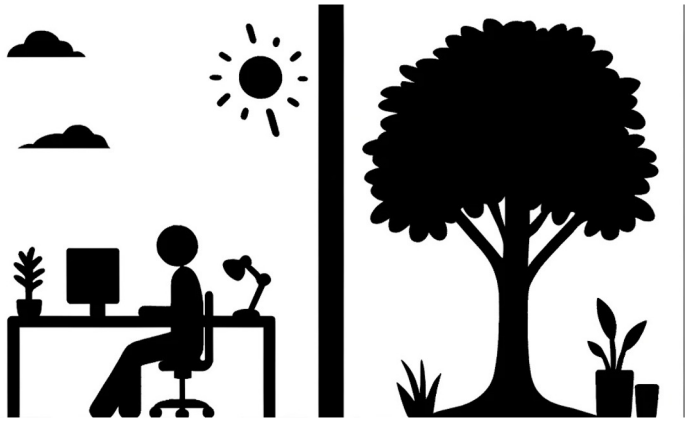


What is a “context aware” application ?

- A self-adaptive application that distinguishes between the following two cases:

A mistake (a misbehaviour)

A context shift



To summarise: “context aware” applications deal with sensors …

- … deployed at large scale and exposed to different and varying conditions that cannot be tackled with a unique inference (MLM) configuration.
- … that are split into groups. Each group is belong to a specific spatio-temporal *context.*
- Each context needs its own MLM configuration (NNs weights)

Self-adaptive/Context-aware applications & Cloud Continuum

In our case, Cloud Continuum refers to the deployment and execution of self-adaptive/context-aware IoT applications.

Four challenges:

1. Is the system smart enough to know when it misbehaves (low performance data) : How to recognise a bad “decision” ?
2. What data the edge must forward to the cloud ?
3. When to start a new learning step ?
4. Is there any human interaction? (Human in the Loop: HITL)

Plan

1. Why/What is Edge Computing and Cloud Continuum ?
2. Targeted applications (self-adaptive/context-aware)
3. **Use-cases**
4. Cloud Continuum (Edge-to-Cloud) solutions
5. Towards a decentralised Federated Learning based edge-to-edge Continuum Computing framework.
6. The Smart Grid Application

Use-case 1: Smart public lighting

Innovation project supported by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Innosuisse – Swiss Innovation Agency

- sMart EDge fabric for Iot Applications (MEDiNA)
  - Reduce energy consumption
  - Reduce light pollution

Use-case 1: Smart public lighting

Low light

Low traffic

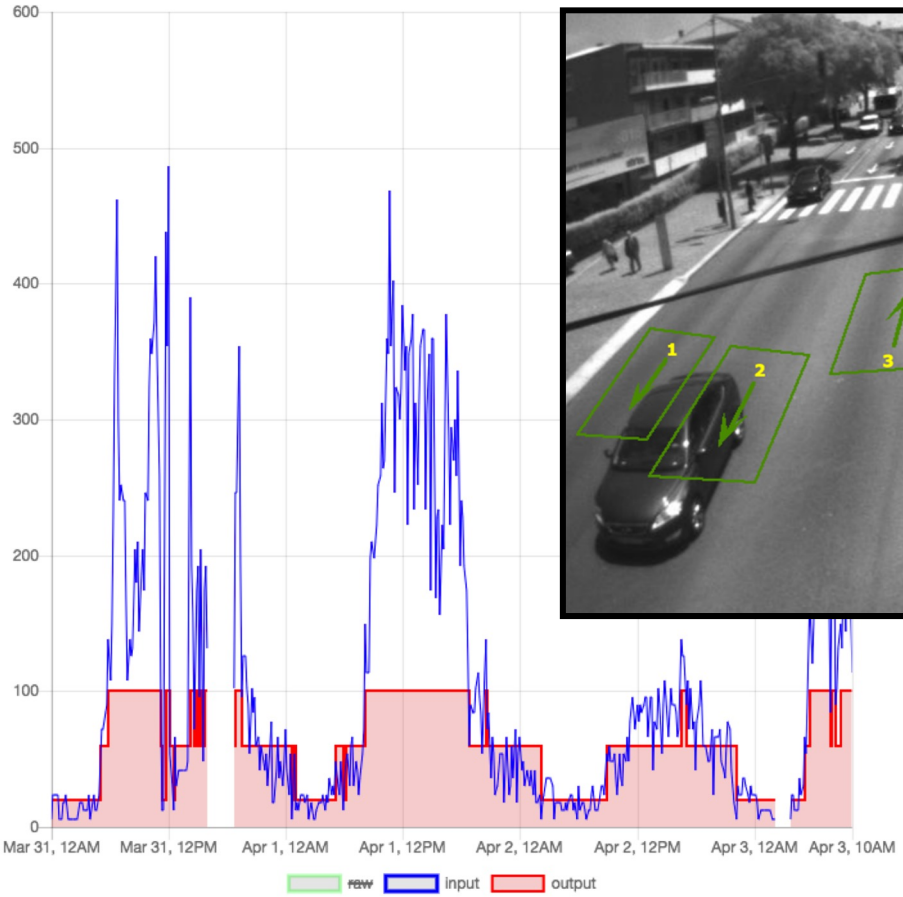
Medium light

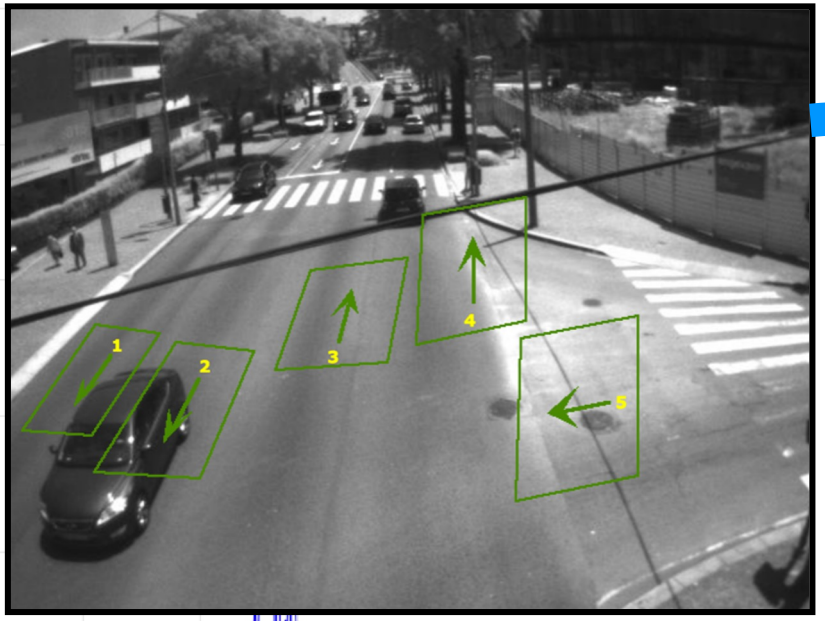
Medium traffic

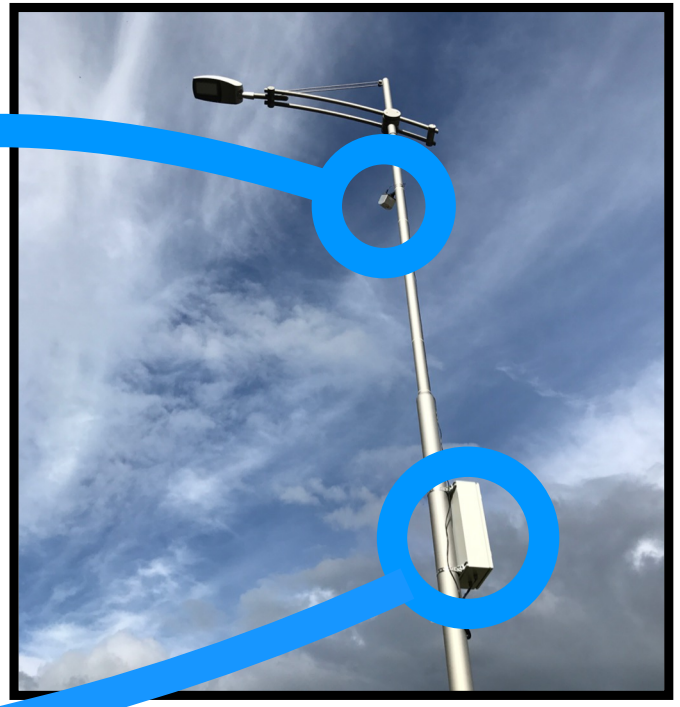
Full light

High traffic

Use-case 1: Smart public lighting





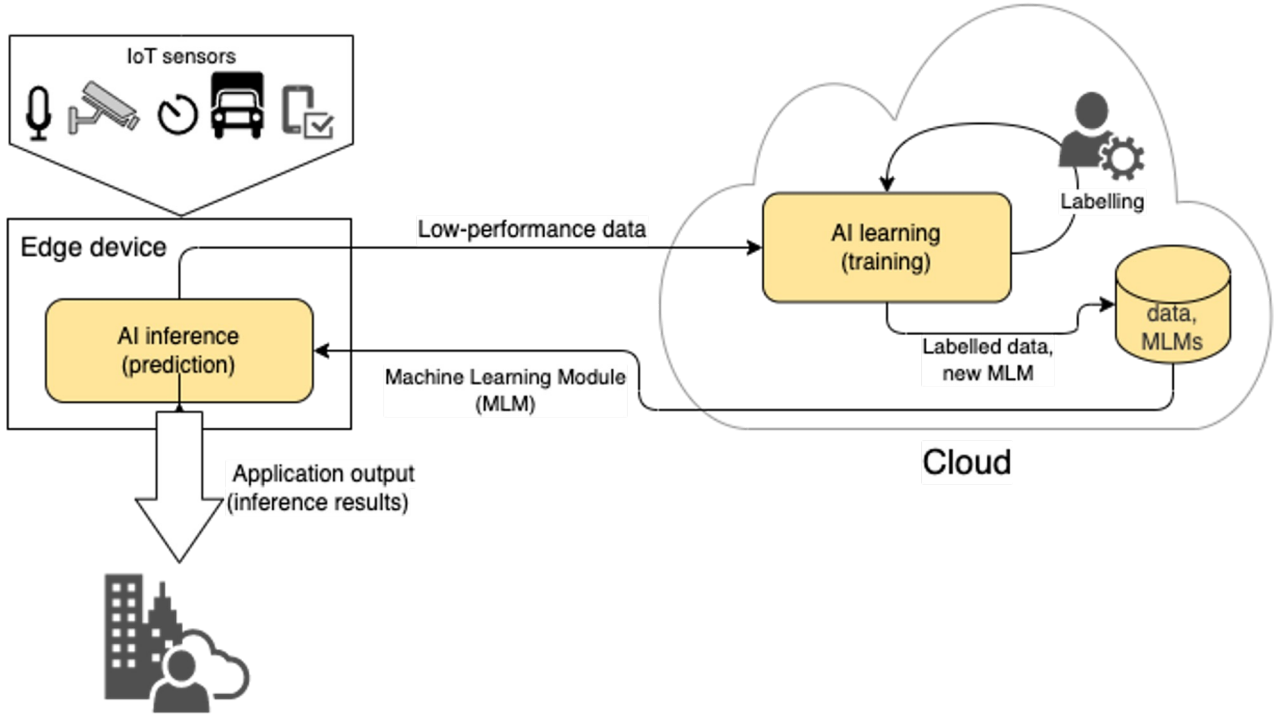


Use-case 1: Smart public lighting

Use-case 1: Smart public lighting

Two questions:

- How to recognise a bad “decision” ?
- What information the edge must forward to the cloud ?

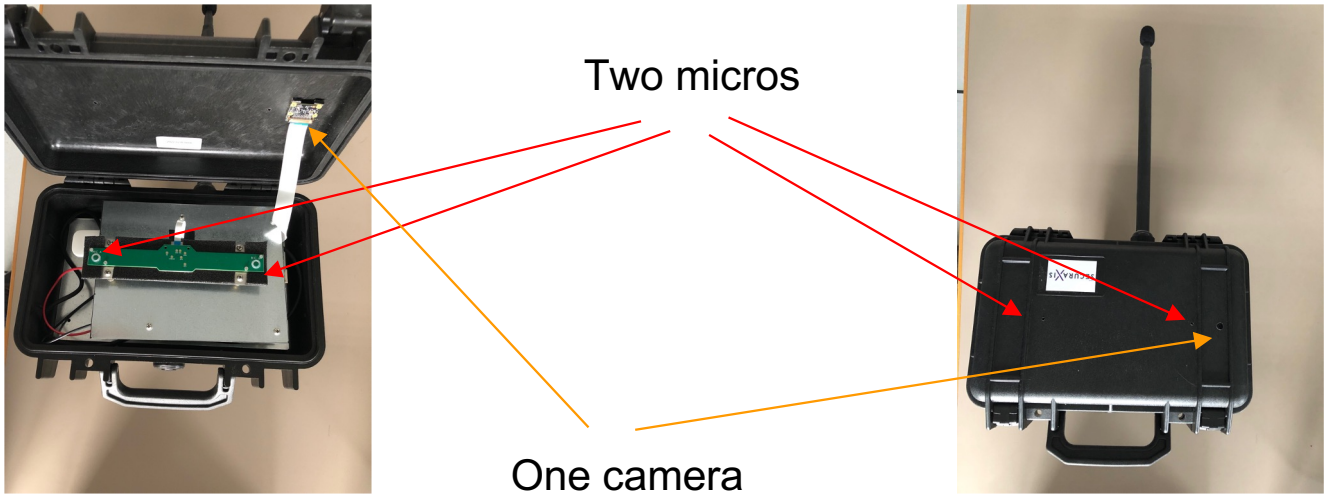


Use-case 1: Smart public lighting

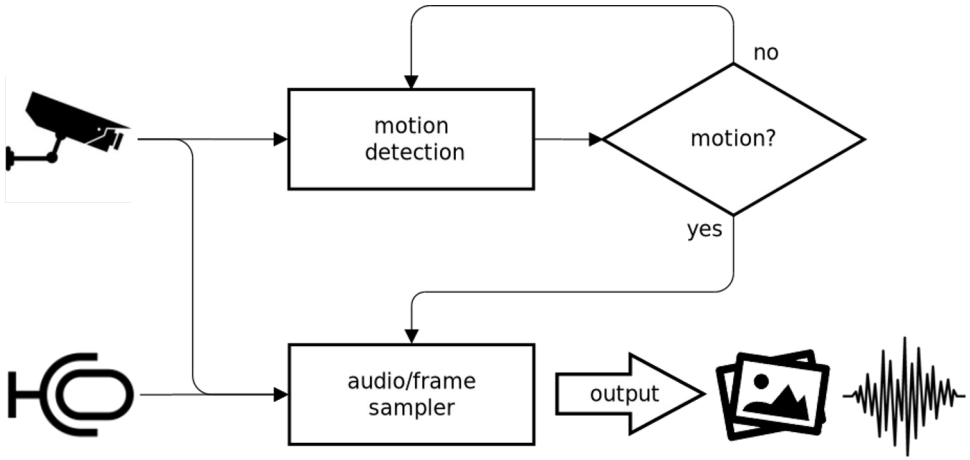
The MEDInA video

Use-case 2: Traffic noise monitoring

- A noise **“blind”** detector based on acoustic and AI technologies
  - No camera → No privacy problems
- Classifying vehicles (cars, truck, buses, motorcycles, electric cars)

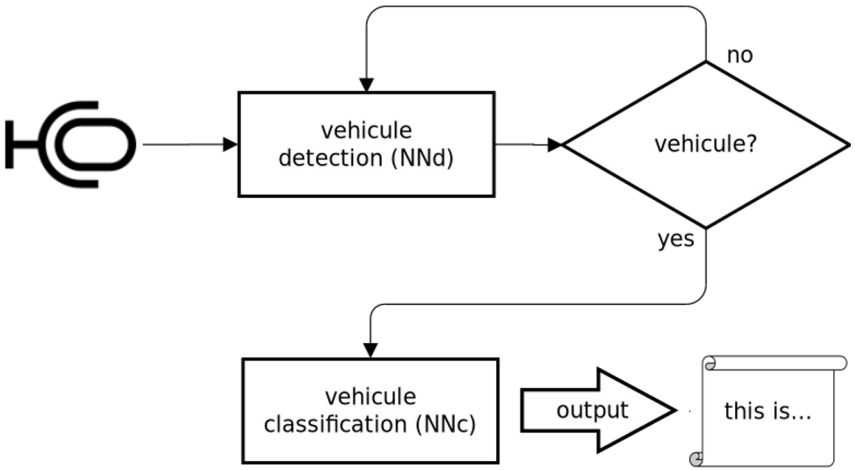


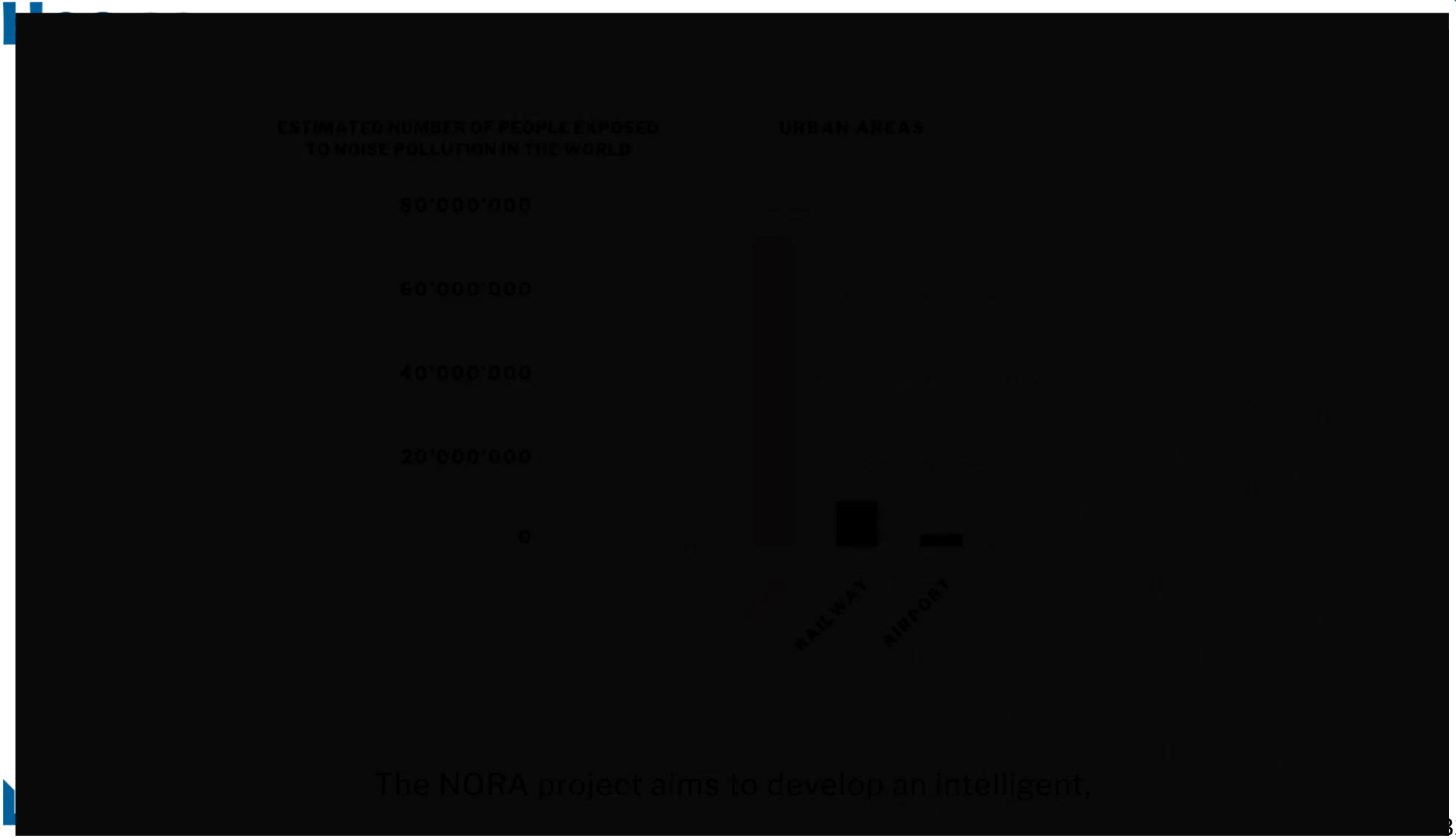
Use-case 2: Learning/Inference



Learning

Inference





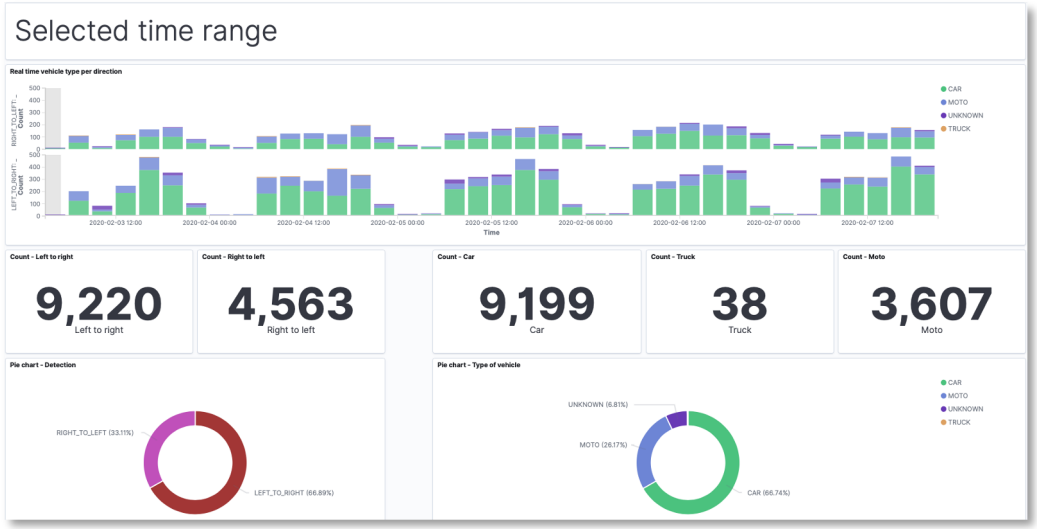
The NORA project aims to develop an intelligent,

Use-case 2: Traffic noise monitoring

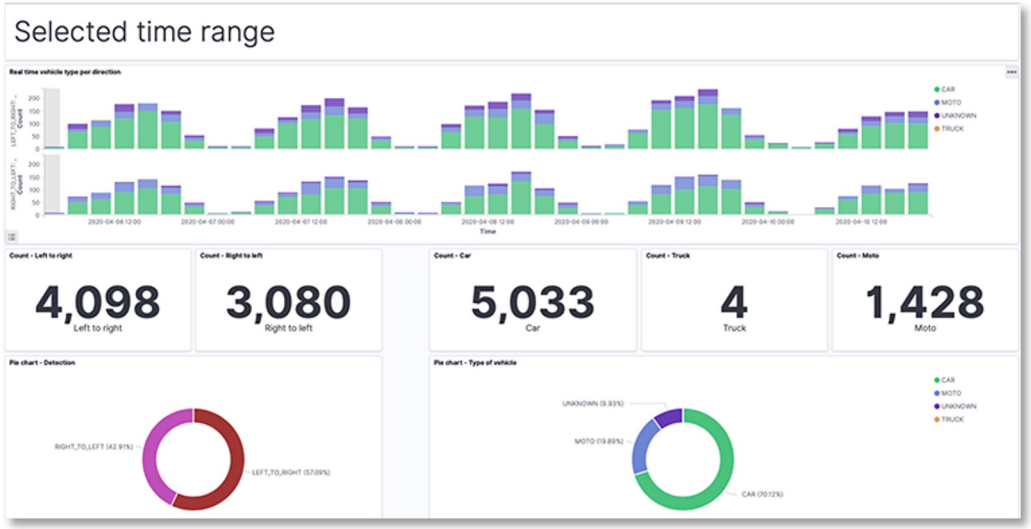
2019

Use-case 2: Traffic noise monitoring

Feb. 3 to Feb. 7 (2020): 13’783 vehicles



From April. 6 to Friday 10 April (2020) : 7’178 vehicles



Use-case 2: Traffic noise monitoring

- March 2021: ***Noise Radar (NORA)***
  Innosuisse project: ,
  - FPGA based edge
  - **Context-awareness** intelligence (30 km zone, 1 lane, 2 lanes, 4 lanes, weather, snowing, rainy day, etc.)
  - “Remotely” control the intelligence in case of (1) misbehaving sensors or (2) switching from one context to another

Innovation project supported by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

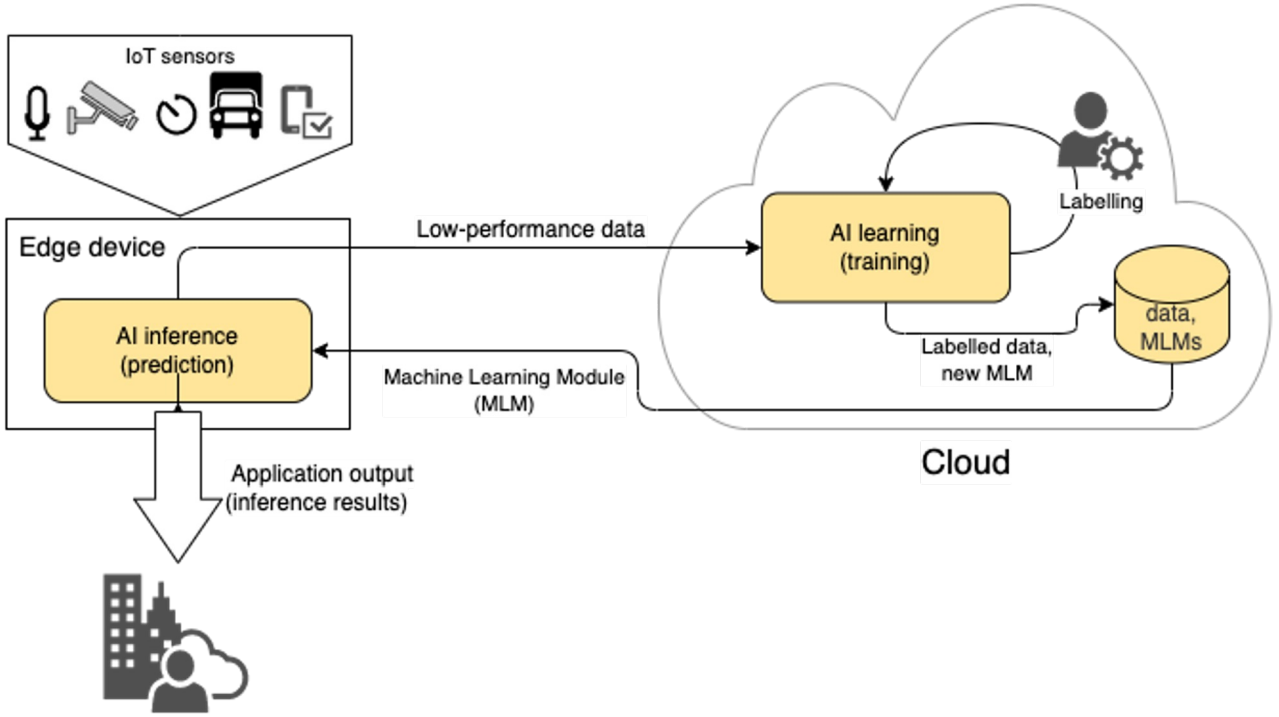
Innosuisse – Swiss Innovation Agency

SECURAXIS

Use-case 2: Traffic noise monitoring

Two questions:

- How to recognise a bad “decision” ?
- What kind of information the edge must forward to the cloud ?

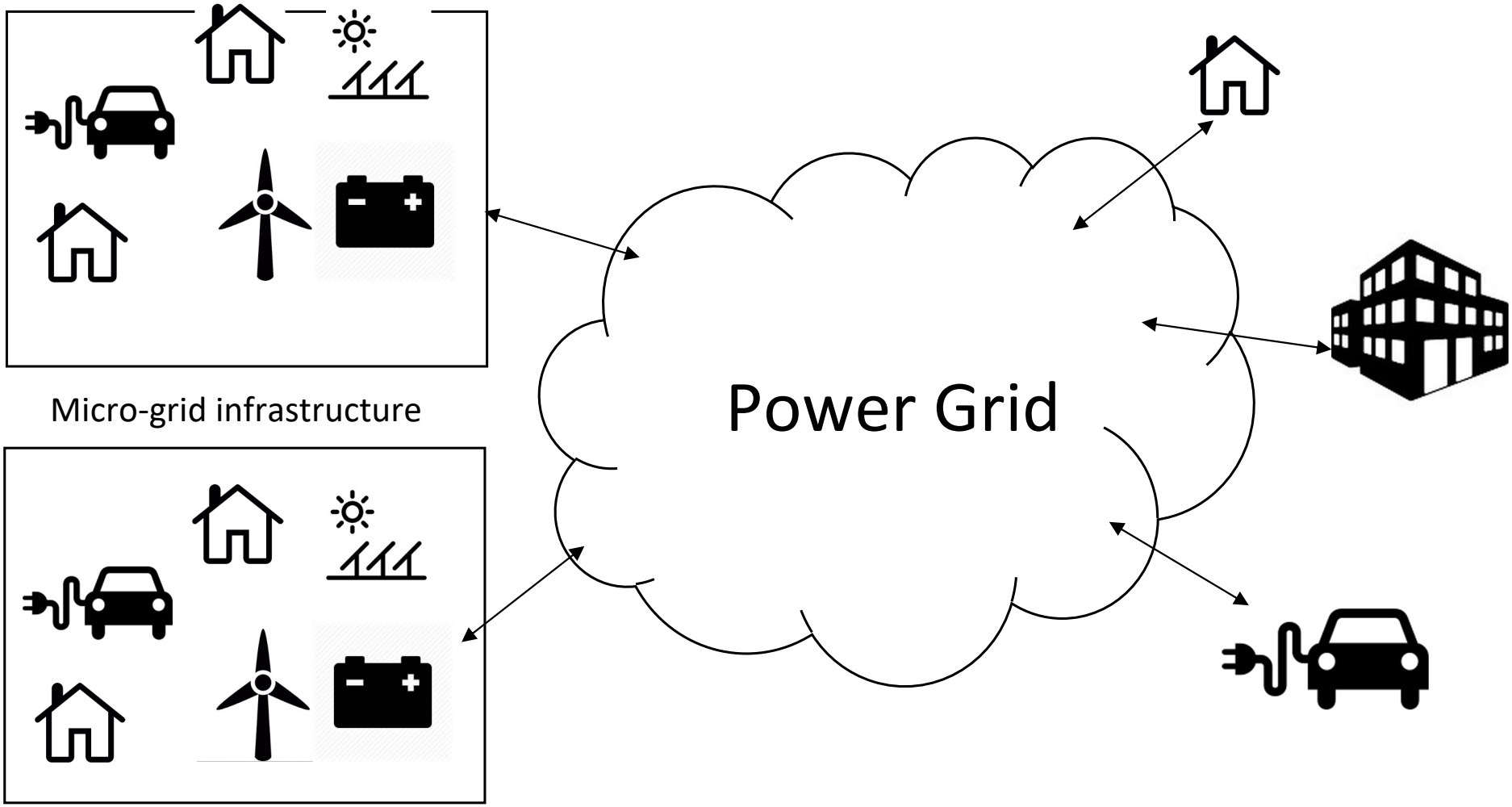


Use-case 3: Monitoring heating systems

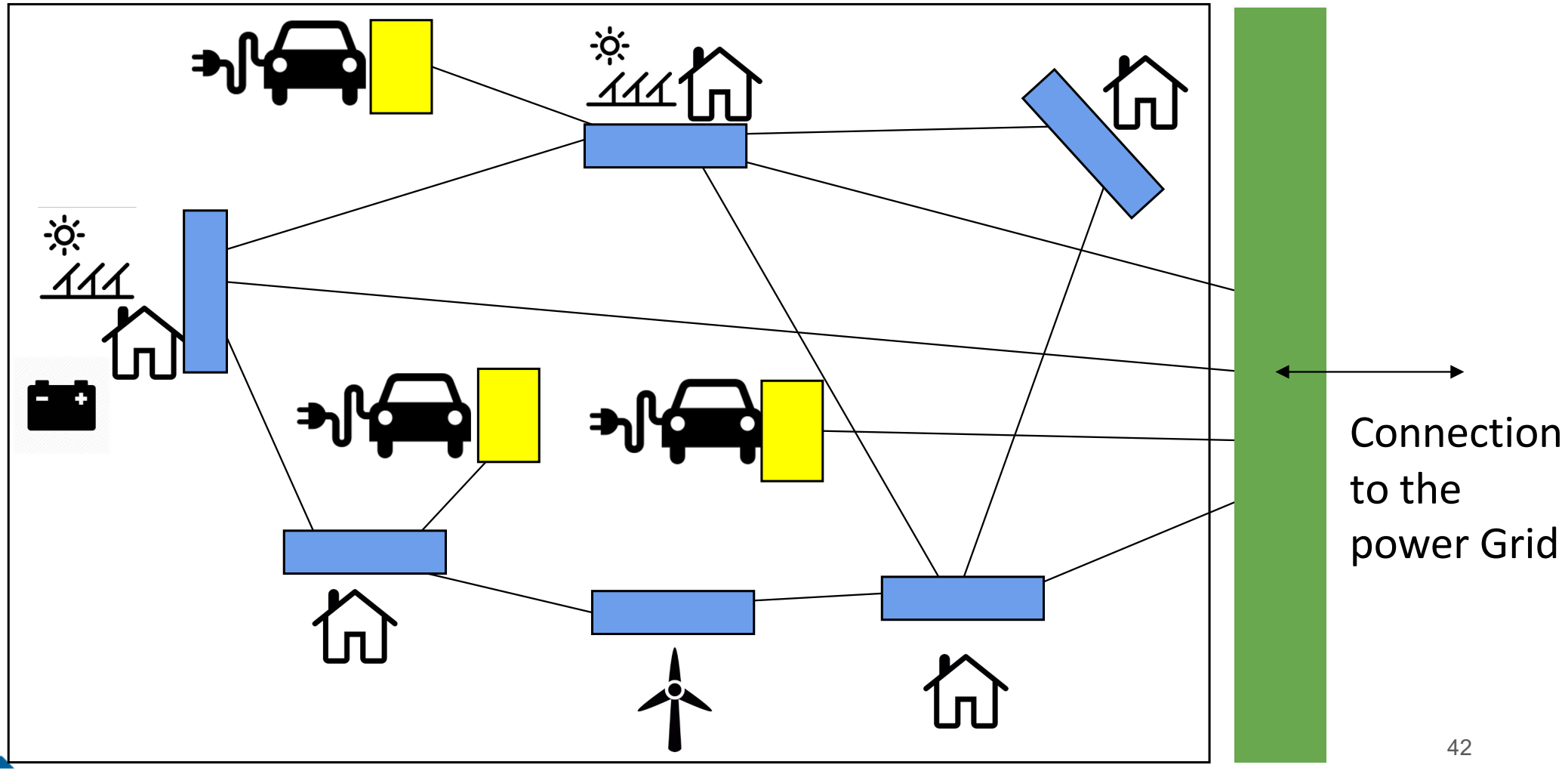
- SG-Box (BACnet, ModBus, LoRa, etc.)
- Siemens (LPB, Wifi)
- The goals are to:
  - collect data and take real time decisions
  - decrease the frequency and duration of breakdowns
  - decrease the on-site technical visits
  - Forecast, in order to control/optimise

Use-case 3: Monitoring heating systems

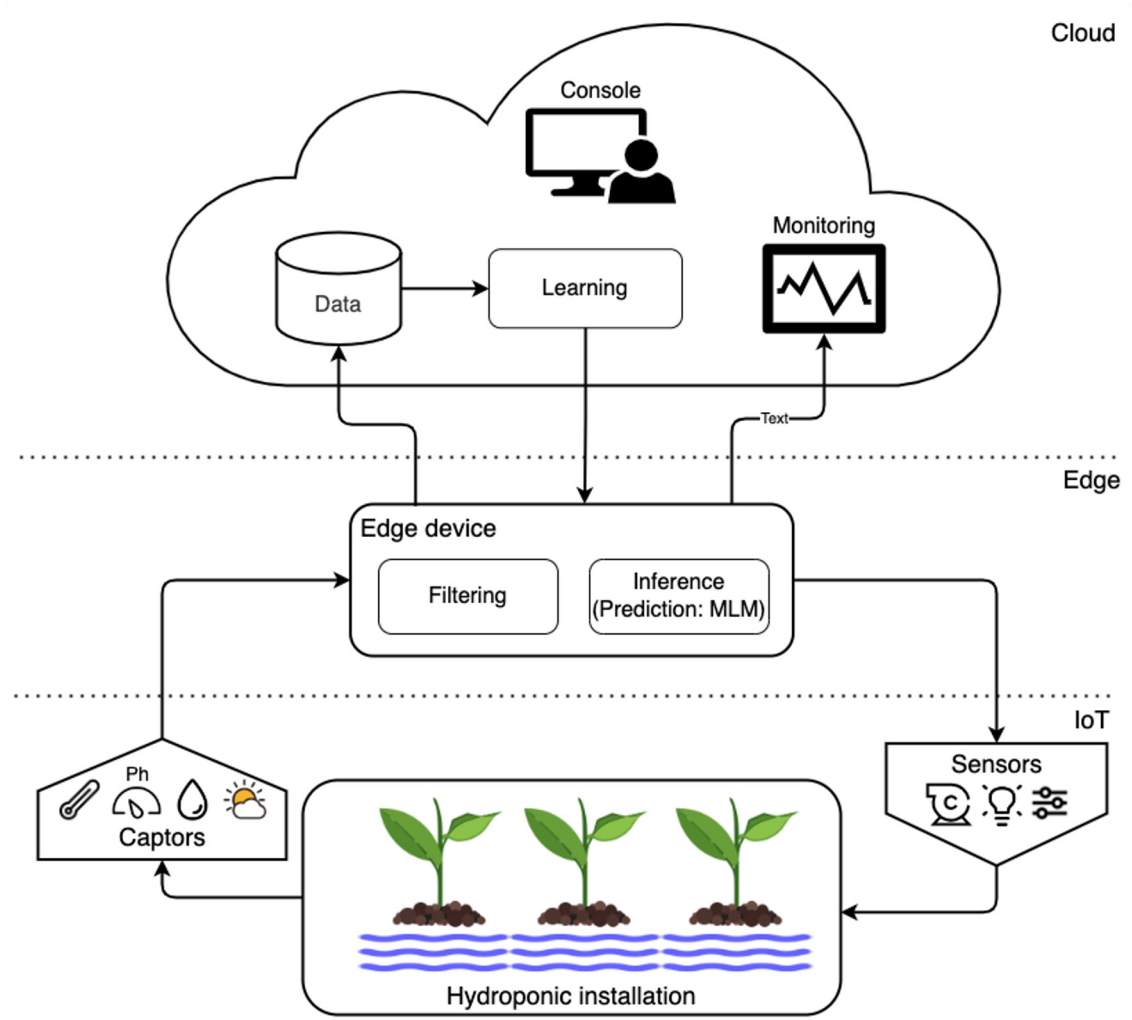
Use-case 4: Smart energy



Use-case 4: Smart Energy



Use-case 5: Indoor Vertical farming (IVF)



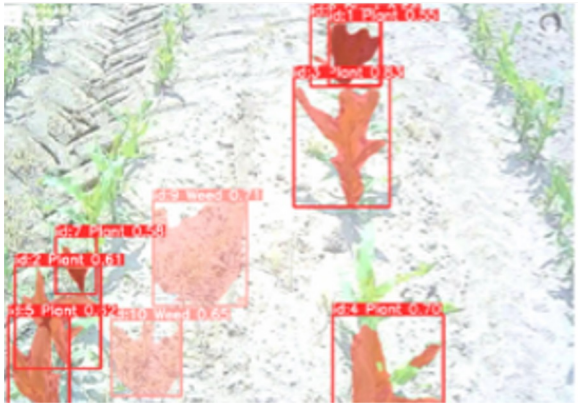
Two objectives

- Control the nutrient level
- Optimise energy consumption

Use-case 6: Agriculture Operational Robotic Platform

Objectives:

1. controlling the level of growth
2. Spot weeds.
3. minimise the use of pesticides

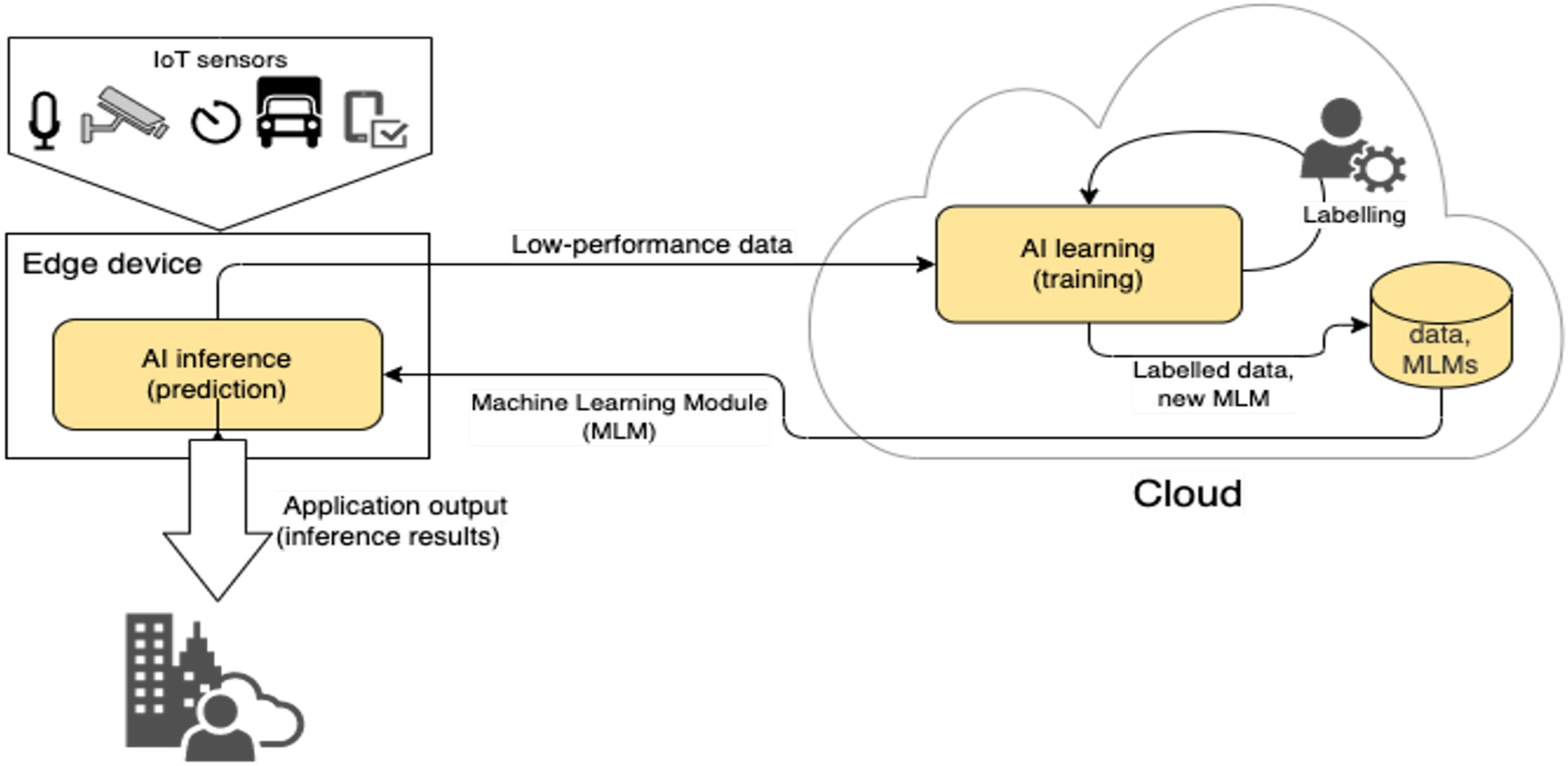


Segmentation, tracking and counting

Plan

1. Why/What is Edge Computing and Cloud Continuum ?
2. Targeted applications (self-adaptive/context-aware)
3. Use-cases
4. **Cloud Continuum (Edge-to-Cloud) solutions**
5. Towards a decentralised Federated Learning based edge-to-edge Continuum Computing framework.
6. The Smart Grid Application

Cloud Continuum: the closed loop



Five edge-to-Cloud solutions, three comparative criteria

- AWS Amazon
- Azure
- Google
- Nuvla/NuvlaEdge
- Balena

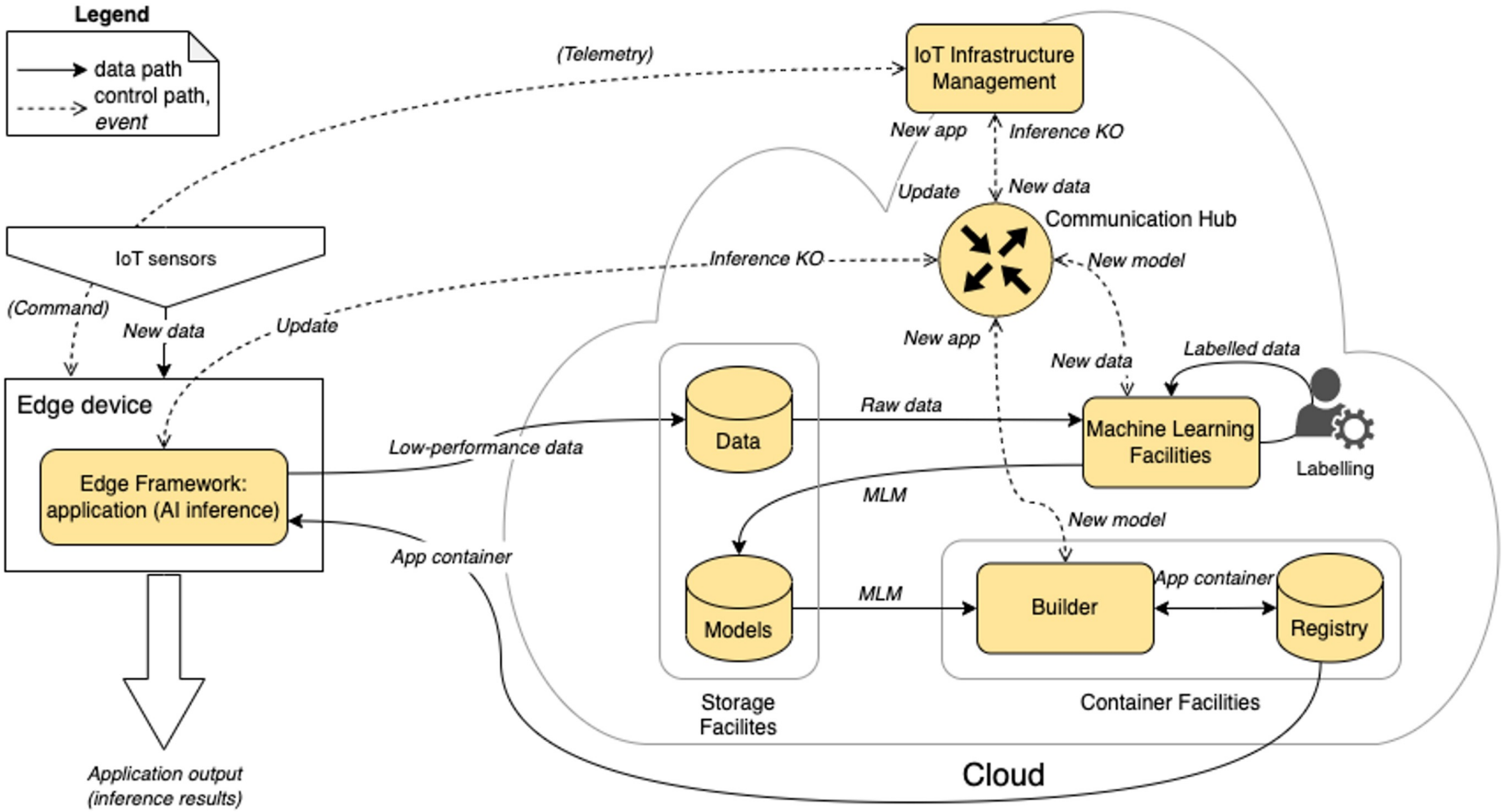
- Service level
- Application level
- Operating Cost

Marco Emilio Poleggi

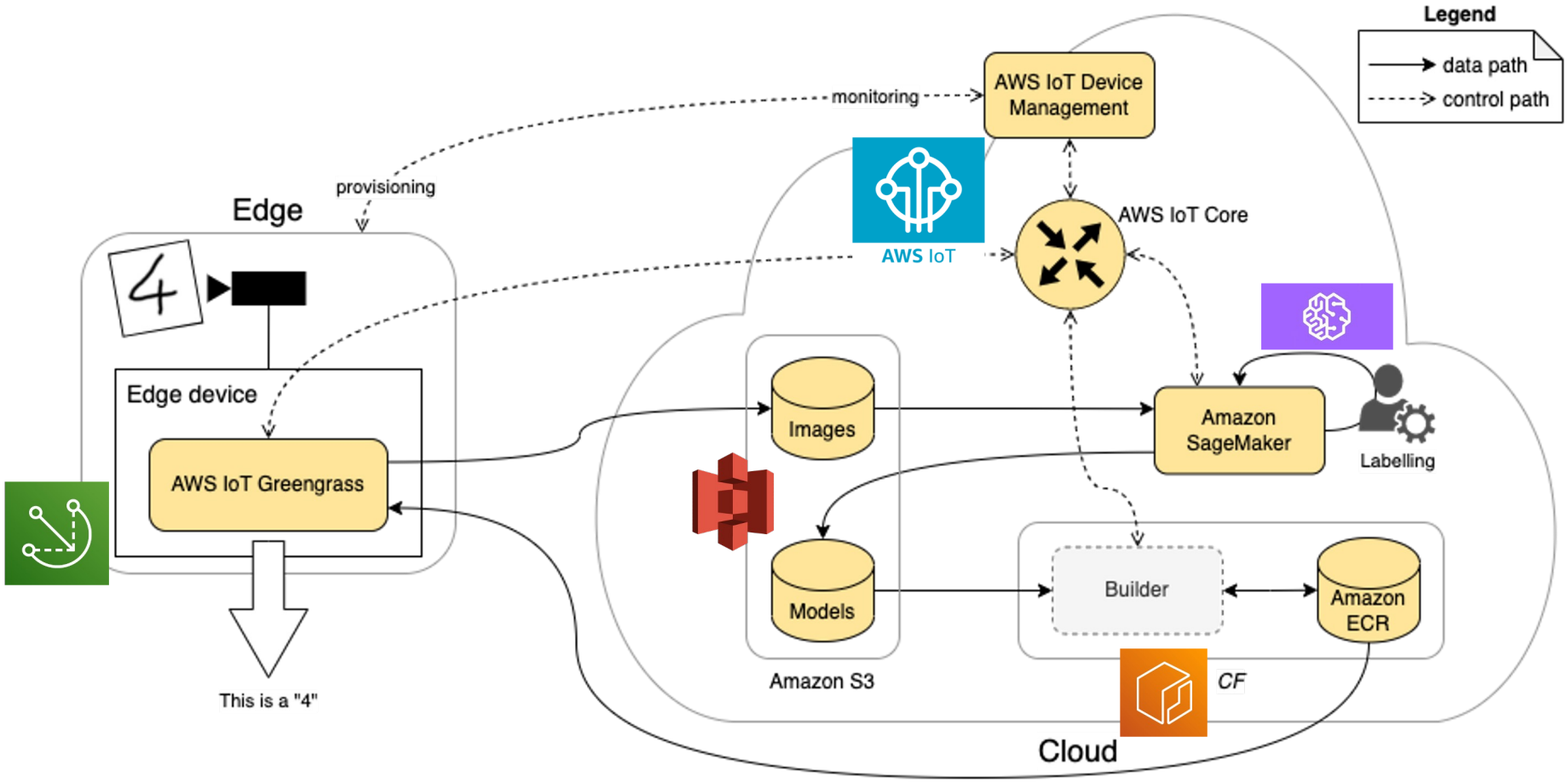
Anatomy of Edge-To-Cloud solutions

- **Edge Framework**: enable edge application modules programming and execution.
- **Container Facilities**: build a *container* (such as Docker) with a trained ML Models (MLM)
- **Communication Hub**: create a messaging infrastructure for networked components
- **Storage Facilities**: store training data (labeled images in the MNIST case) and several MLM versions in (possibly different) Cloud data warehouses.
- **Machine Learning Facilities**: build and train a MLM in the Cloud.
- **IoT Infrastructure Management**: An orchestration service that composes, provisions (configures and deploys) and monitors Edge/IoT networks.

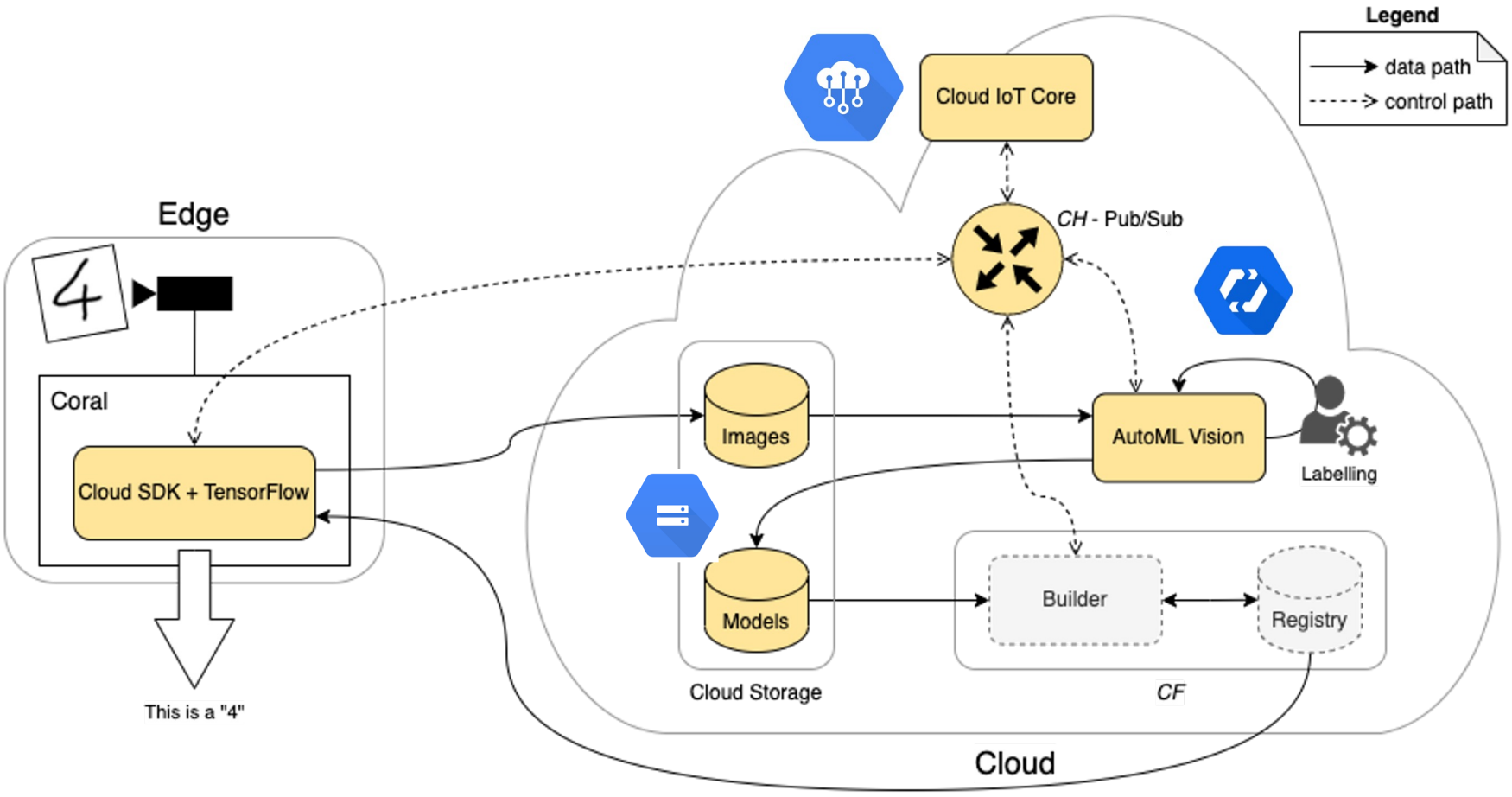
Anatomy of Cloud Continuum solutions



AWS Amazon solution



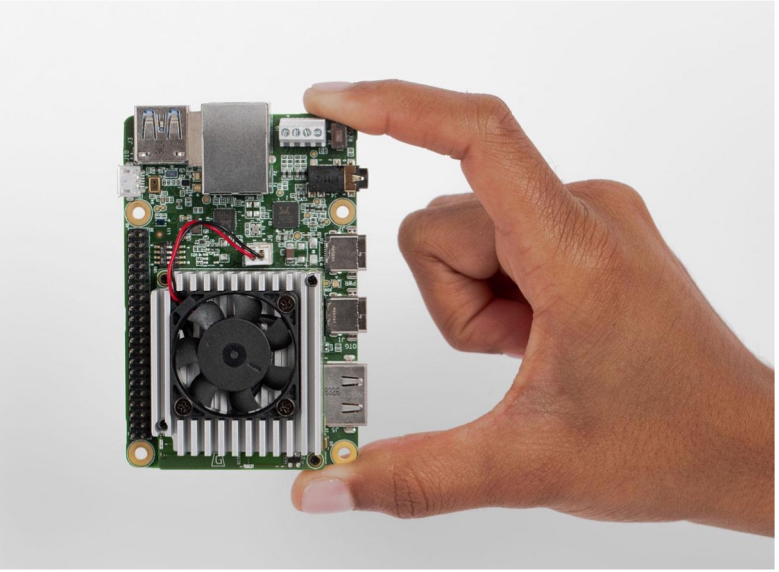
Google solution



Google Coral Dev Board

- A single-board computer that contains an Edge TPU* coprocessor.
- Ideal for projects that demand fast on-device machine learning models.

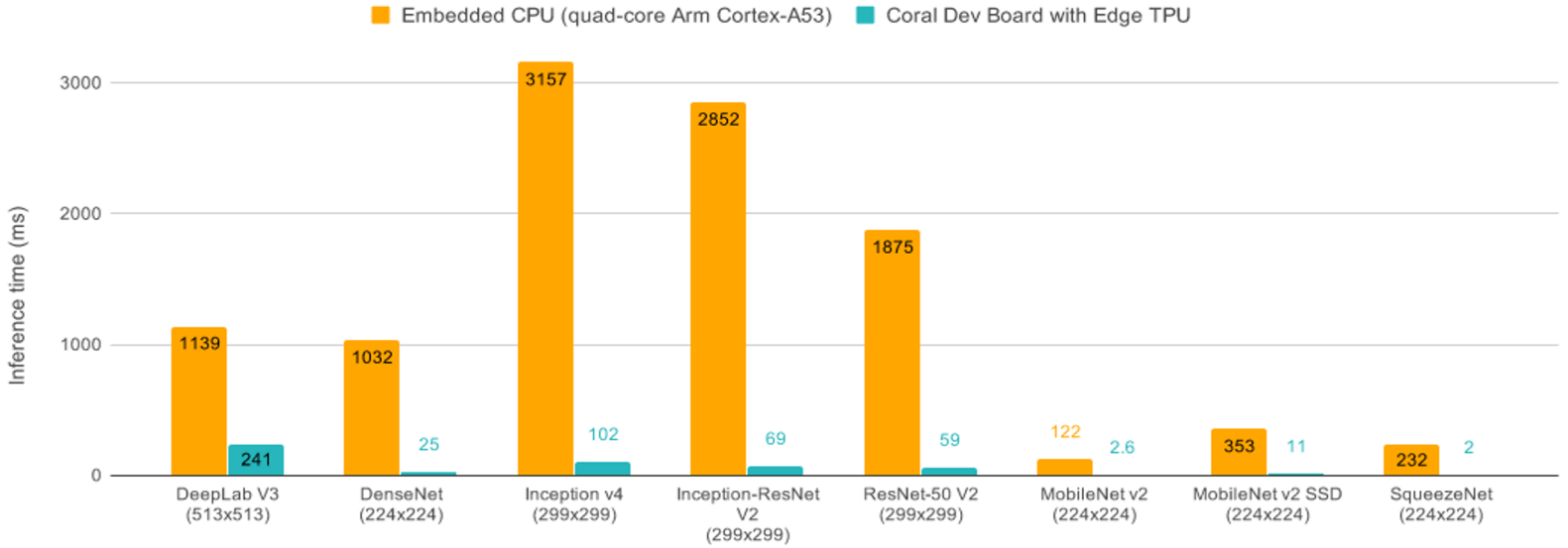
*: Tensor Processor Unit



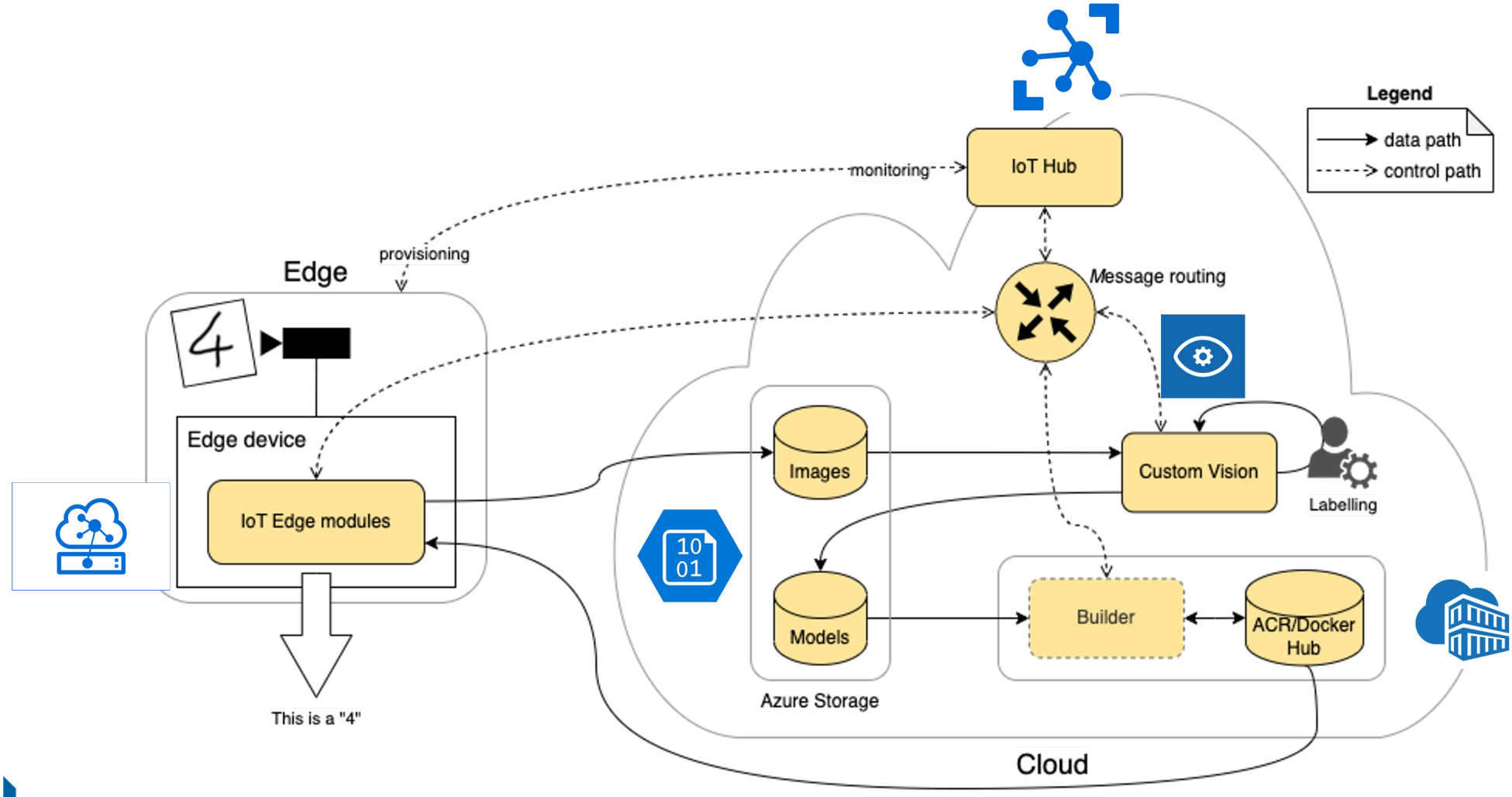
Coral device

Edge TPU coprocessor (ASIC)

TensorFlow Lite API



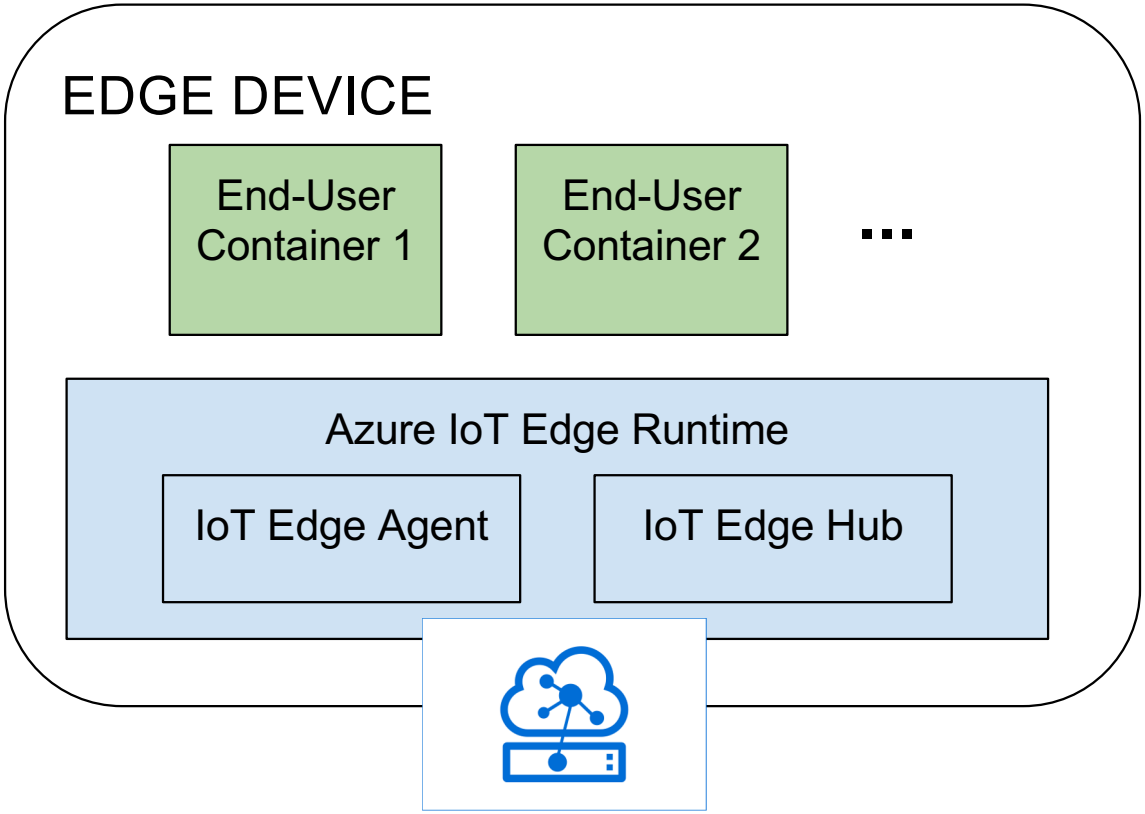
Azure solution



Azure IoT Edge runtime

Two containers :

- IoT Edge Agent
- IoT Edge Hub



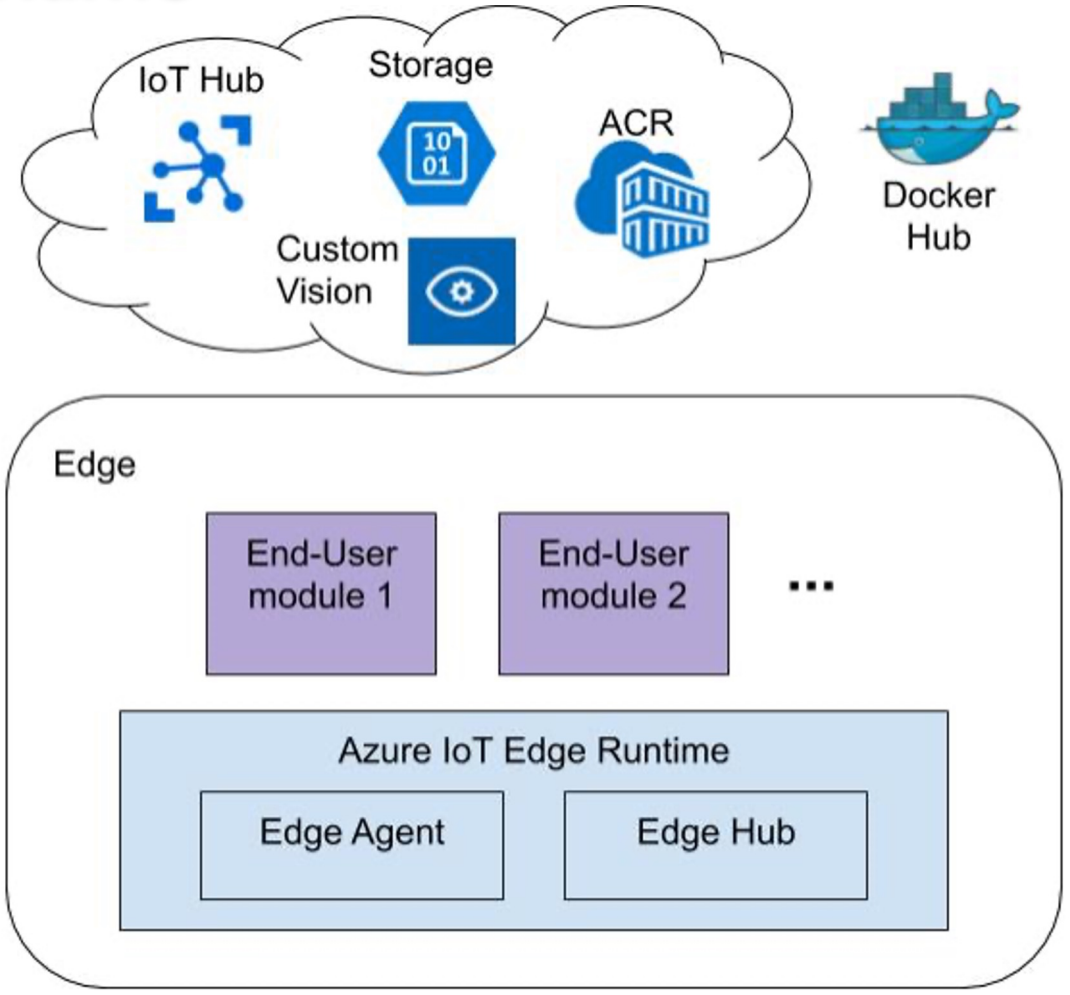
Azure IoT Edge runtime

Composed of two containers:

- IoT Edge Agent
- IoT Edge Hub

End-user containers:

- Module 1
- Module 2
- ...



IoT Edge Hub

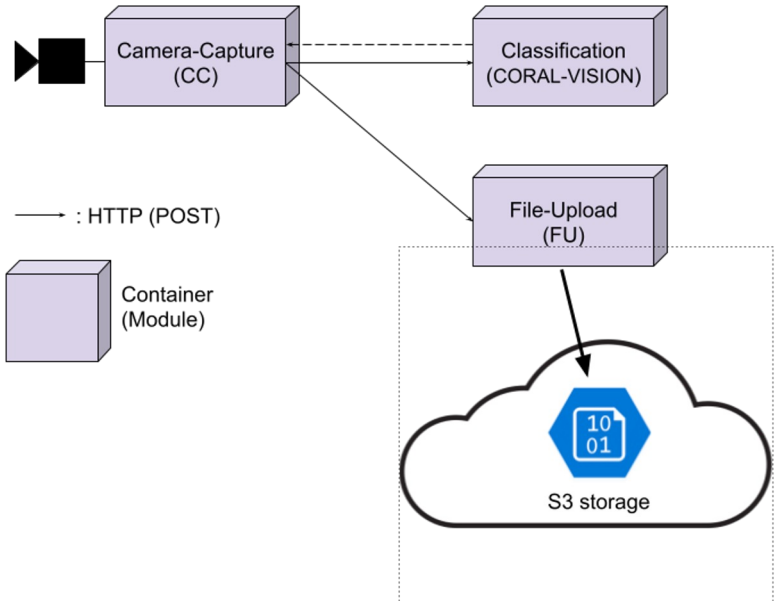
- Acts as a local proxy of IoT Hub
  - The clients can connect to the IoT Edge Hub just as they would to IoT Hub.
- Mimics Azure IoT Hub
  - It is for IoT devices what cloud is for Edge devices.
- Functionalities:
  - Authentication
  - Reducing bandwidth
  - Working offline
  - Module communication

IoT Edge Agent

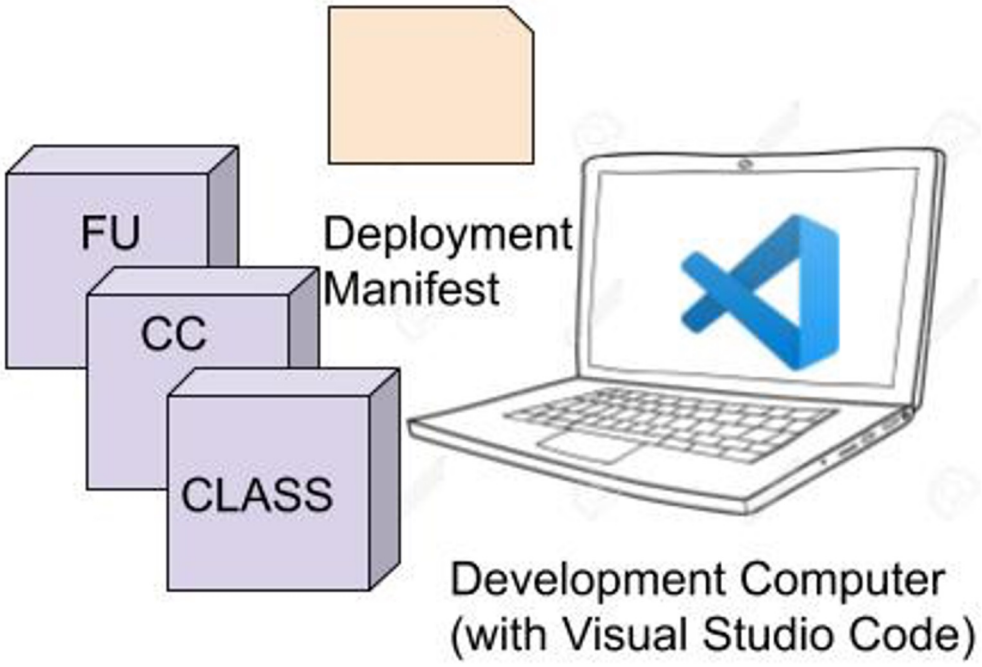
- Instantiates modules
- Ensures that they continue to work
- Reports the status of the module back to the IoT Hub (Cloud)
- Retrieves the information from the deployment manifest

Application overview

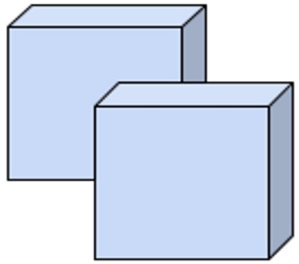
- **CC** takes a frame and pre-process it
- **CC** sends to **CLASS** the frame, **CLASS** answers with an inference result
- In case of low performance data, **CC** sends the frame to **FU**
- **FU** sends the frame to the cloud Storage



Development & Deployment manifest

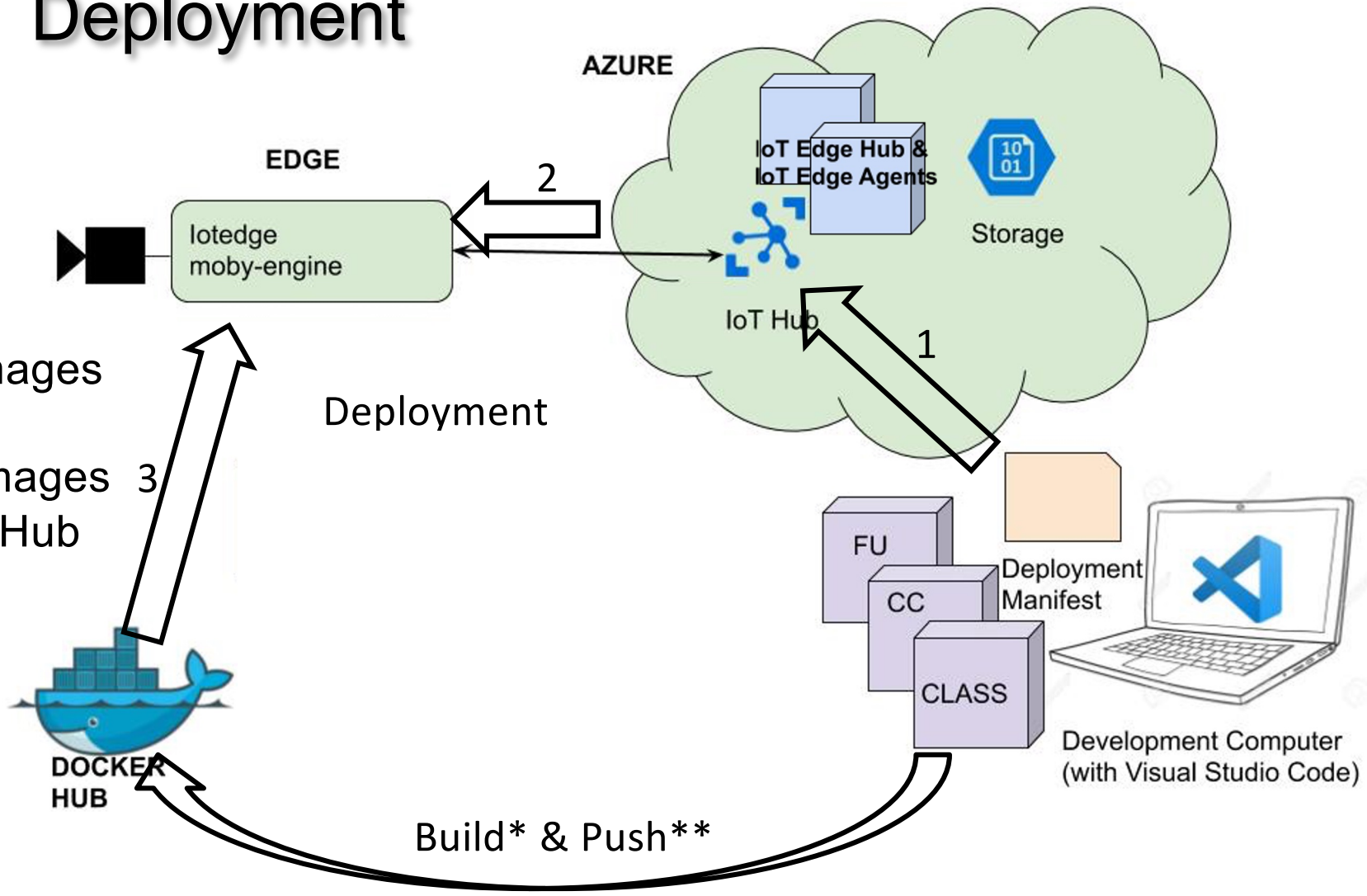


Azure Container Registry (ACR)



IoT Edge Hub & IoT Edge Agent containers

Deployment



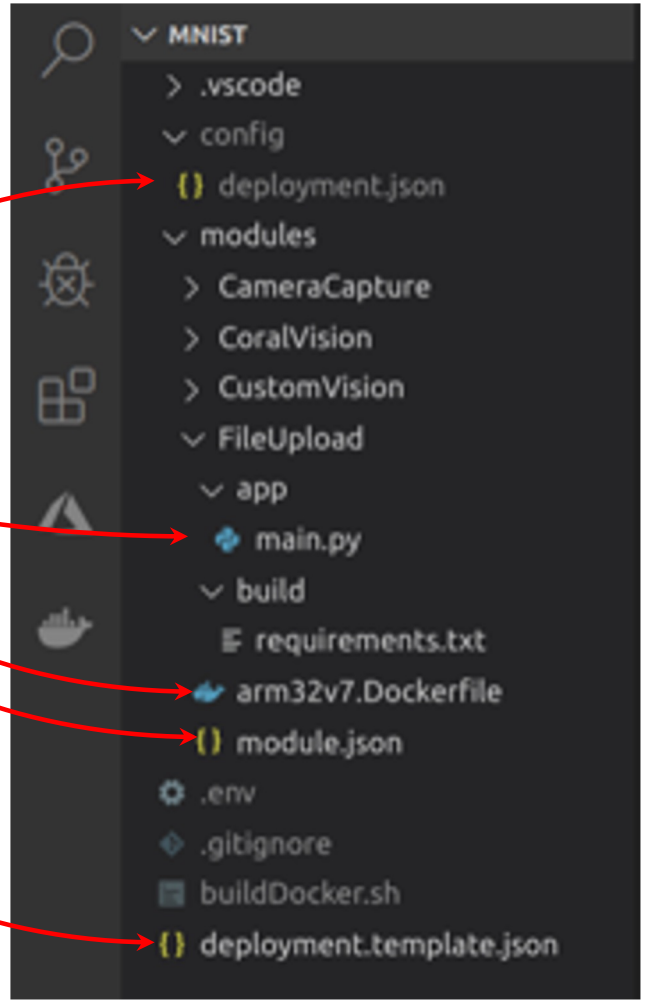
*: Store the images locally

**: push the images to the Docker Hub

Development (Visual Studio)

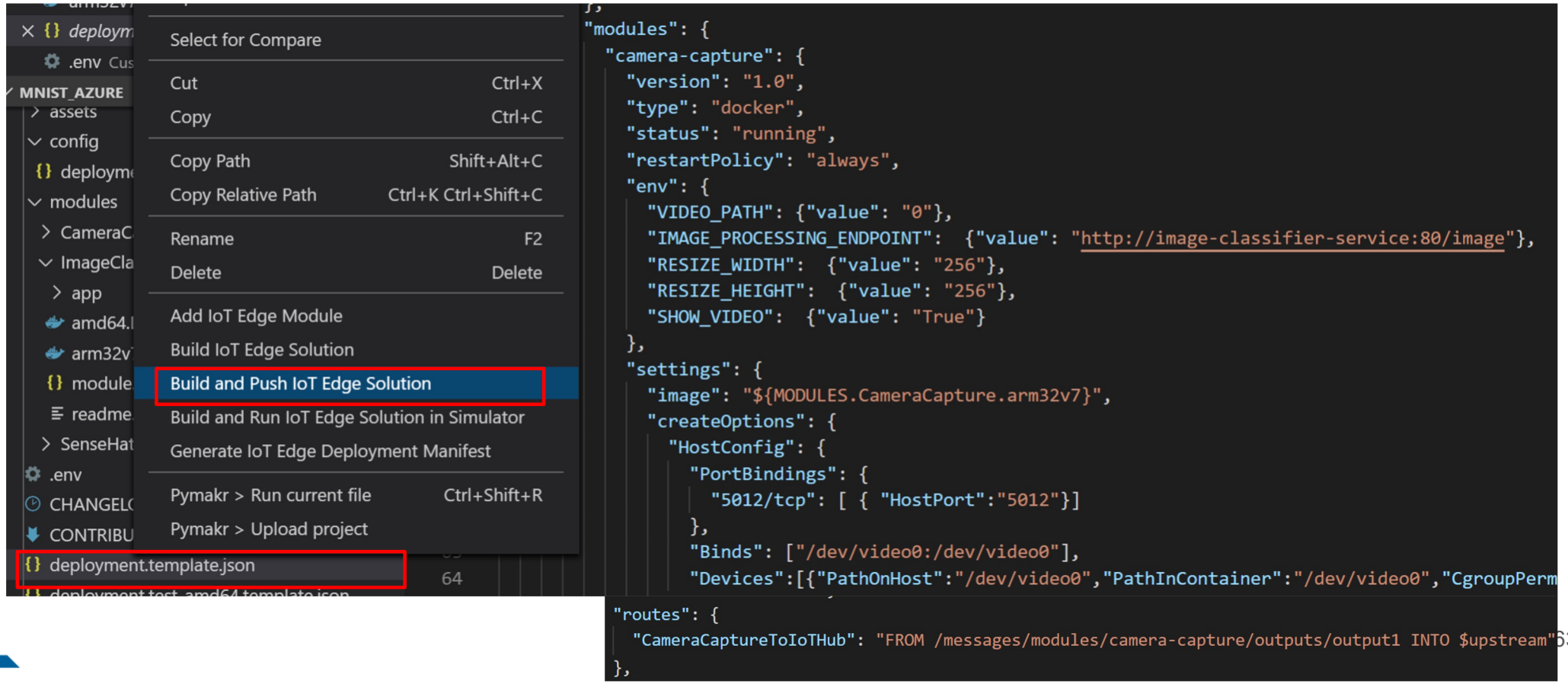
Files for IoT Edge Solution :

- Program code
- Dockerfile with cross-compile
- module.json
  repository URL, dockerfile, version ...
- deployment.template.json
  Build and Push IoT Edge Solution
- deployment.json
  generated by previous step
  Create Deployment



Deployment (Visual Studio)

Build and Push to Registry

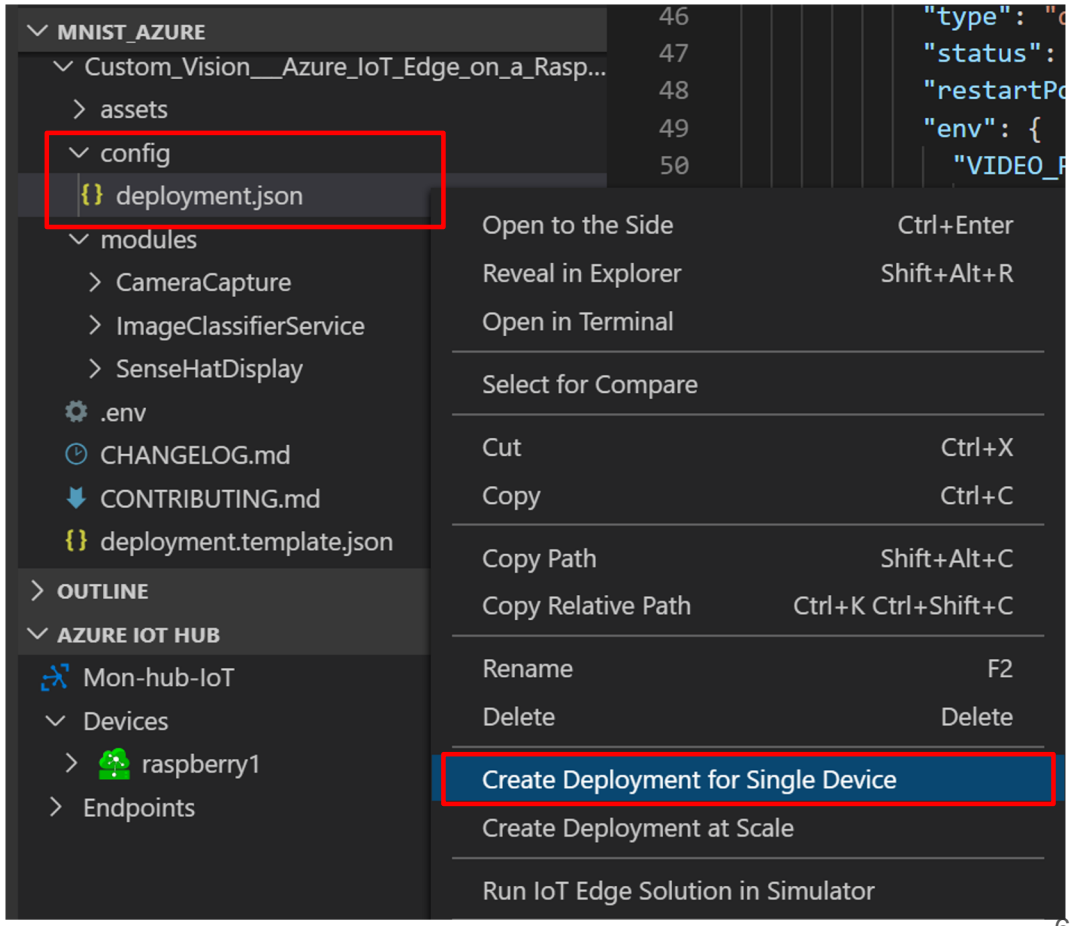


Deployment (Visual Studio)

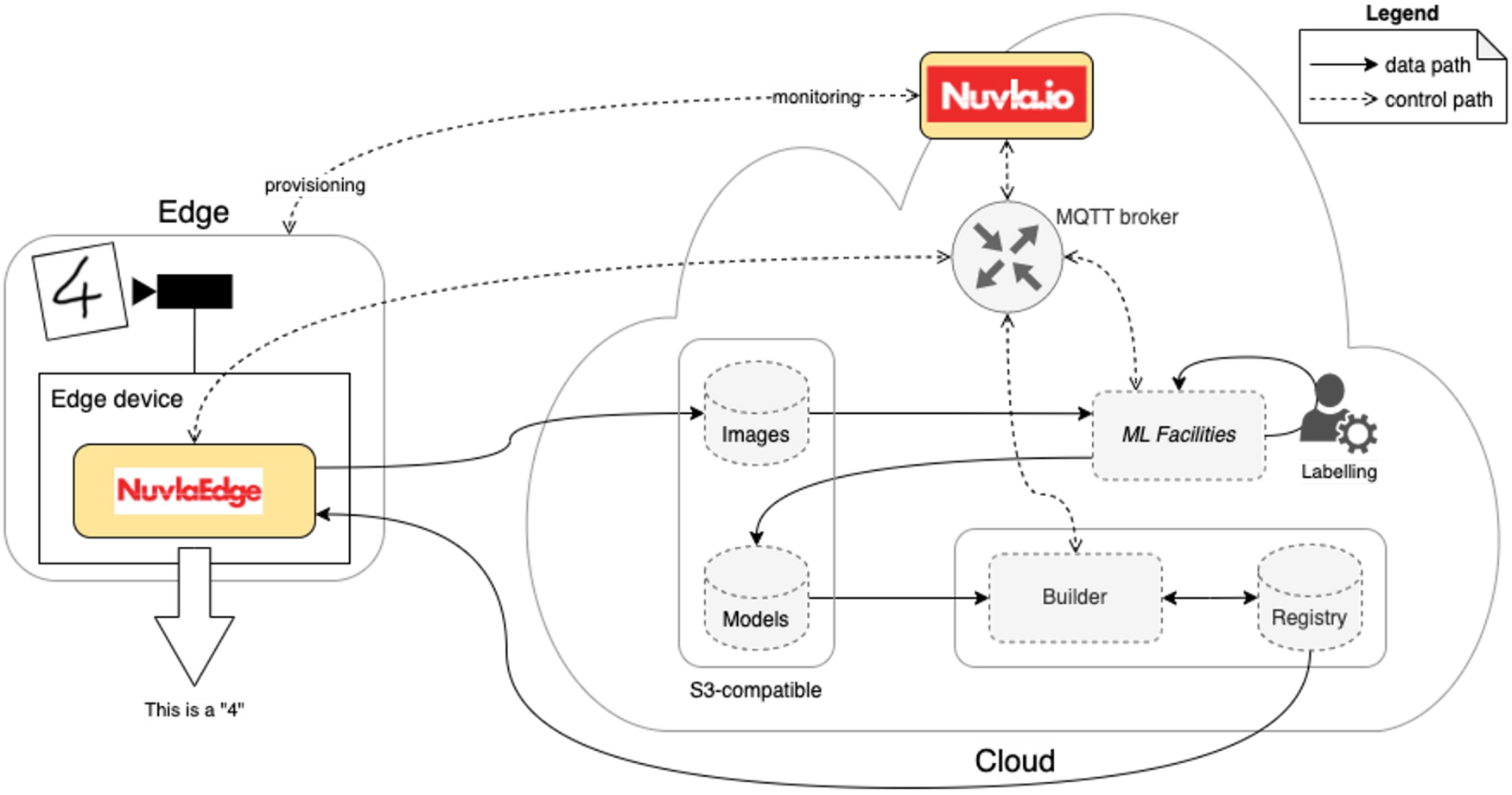
Deployment

File “config/deployment.json” generated by previous step

Send JSON file to edge device



Nuvla/NuvlaEdge solution



Hes·so

sixsq

Any hardware

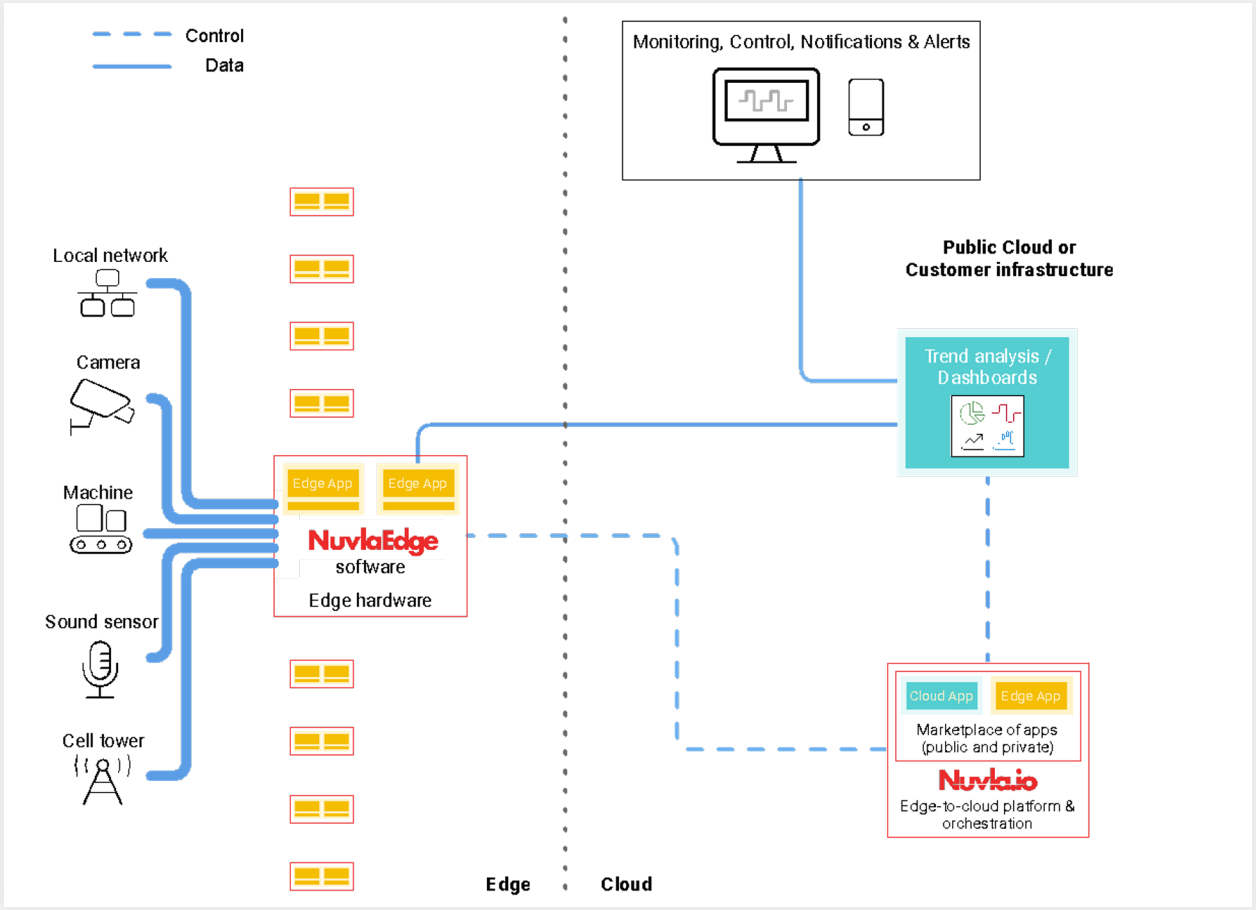
NuvlaEdge

- Turns any computer into a smart edge device
- Hardware agnostic
- Connects securely to Nuvla.io for control

SaaS B2B platform

Nuvla.io

- Application centric
- Cloud neutral
- Container native
- Open, secure & agile



Secure

Access control, integrity checks, zero-touch provisioning and multi-tenancy.

Scalable

Manage fleets of 10,000s of devices.

All cloud & hardware

All major cloud vendors. Certified for both ARM and x86 architectures.

Slide belonging to SixSq

Nuvla

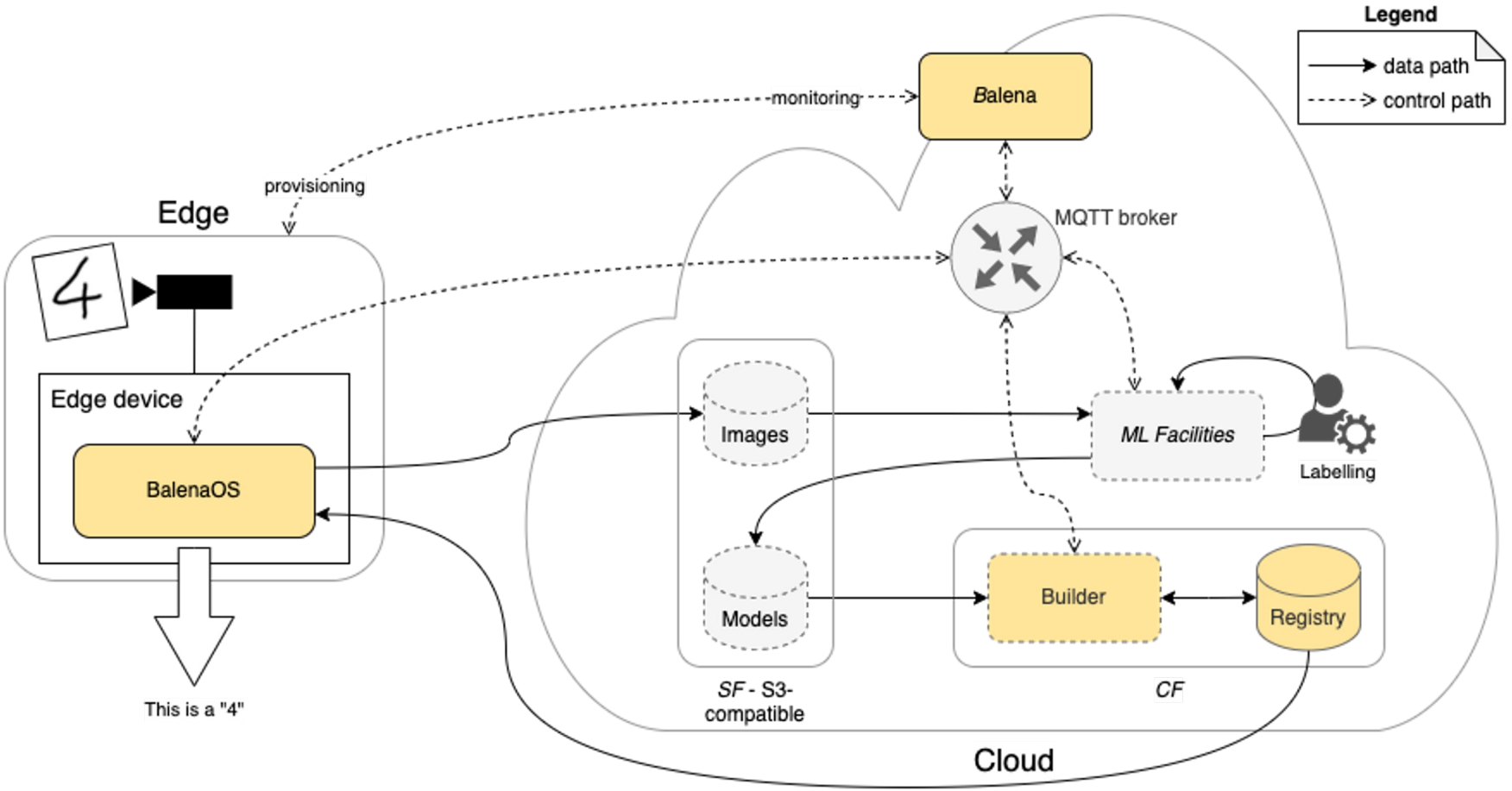
- An edge and a container management platform built upon open-source software and open standards
- Available as either a stand-alone software stack for installation on premises or as a PaaS via *Nuvla.io*.
- Enables users to manage their edge devices and deploy applications that combine edge and cloud modules.
- All applications are packaged as containers images and stored in a registry.
- Edge devices and Cloud Computing instances which support Containers Orchestration Engines (COE) can be onboarded and used to provision applications.

Hes·so NuvlaEdge

NuvlaEdge

- An Edge framework runtime software composed of a set of microservices, used to transform any device into an “NuvlaEdge” Edge device.
- Allows the user (through Nuvla) to connect to and monitor each edge device individually.
- NuvlaEdge microservices are containers deployed under the control of the Nuvla cloud service.
- Microservices provide facilities for: VPN-based networking, MQTT-based internal messaging, application monitoring, security and discovery of attached HW components

Balena solution



Anatomy of a Cloud Continuum solution

	IoT Infrastr. Manag.	Edge Framework	Container Facilities	Comm. Hub	Storage facilities	ML facilities
AWS Amazon	AWS IoT Device Manag.	GreenGrass	Elastic Container Registry (ECR)	AWS IoT Core	S3	Sagemaker
Azure	IoT Hub	IoT Edge Runtime	Azure Container Registry (ACR)	IoT Hub	Azure Storage	Azure AI
Google	Cloud IoT Core	Coral Accelerator (HW)	Google Container Engine (GKE)	Pub/Sub	Cloud Storage (S3)	AutoML Vertex AI (Jan 24)
NuvlaEdge SixSq)	Nuvla	NuvlaEdge				
Balena	BalenaOS	balenaCloud				

To summarise …

- *Amazon AWS* and *MS Azure* are the most complete platforms, in terms of privately-integrated potential.
  - But this comes at the price of reduced flexibility
- *Nuvla* and *Balena* have their strength in the possibility of deploying *anything* on *any* (foreign) cloud platform.
  - They offer a flexible cloud-neutral “container as a service” (CaaS) orchestration platform
  - In a freedom-of-choice perspective, ***less* might be *more***.

Three comparative criteria

- Service level
- **Application level**
- Operating Cost

Application level

- Licensing model.
- Hosting model.
- Hardware support.
- Documentation quality:
- Integration
  - Interoperability.
  - Components on premises.
  - Orchestration.
  - Monitoring, logging and telemetry.
- Security
  - End-to-end encryption.
  - Identity and access management (IAM)
  - Integrity enforcement.
  - Controlled commissioning.
- Risks
  - Vendor lock-in
  - Security breaches
  - Fault tolerance and resilience:

Three comparative criteria

- Service level
- Application level
- **Operating Cost**

Cost model's parameters

Parameter	Description
`event_rate`	Rate at which the MLM **(inference)** is triggered
`raw_data_size`	Size of a raw data item fed to the MLM
`app_output_size`	Size of an output item produced by the MLM
`ml_error_rate`	Inference error rate: fraction of events which the MLM is unable to classify/predict over a given period of time
`ml_model_size`	Size of the MLM
`ml_point_size`	Size of a data point used to train the MLM: roughly equivalent to `raw_data_size` + (negligible) label metadata

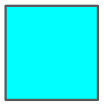
Cost model's parameters

Parameter	Description
`ml_train_size`	Number of data points (of size ml_point_size) used to train the MLM
`ml_train_time`	Computing time needed for training the MLM on 1 CPU core with all the data set in RAM
`ml_train_rate`	Rate of MLM training rounds in the Cloud
`ml_underperf`	Fraction (i.e., the "underperforming") of all MLMs that must be retrained at each round
`edge_img_size`	Size of the MLM (including OS, containers, libraries) deployed to any edge device
`daily_connect_time`	Number of minutes per day during which an edge device is connected to the Cloud

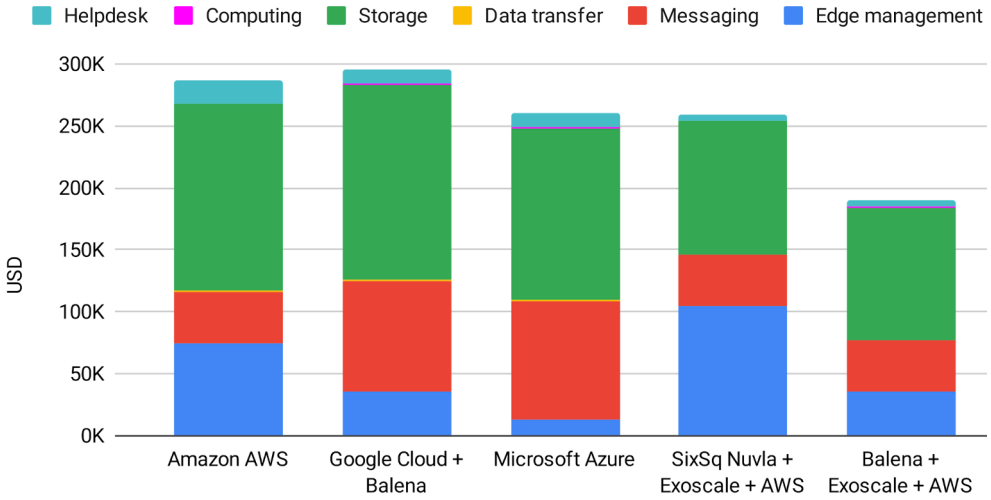
Cost model's parameters

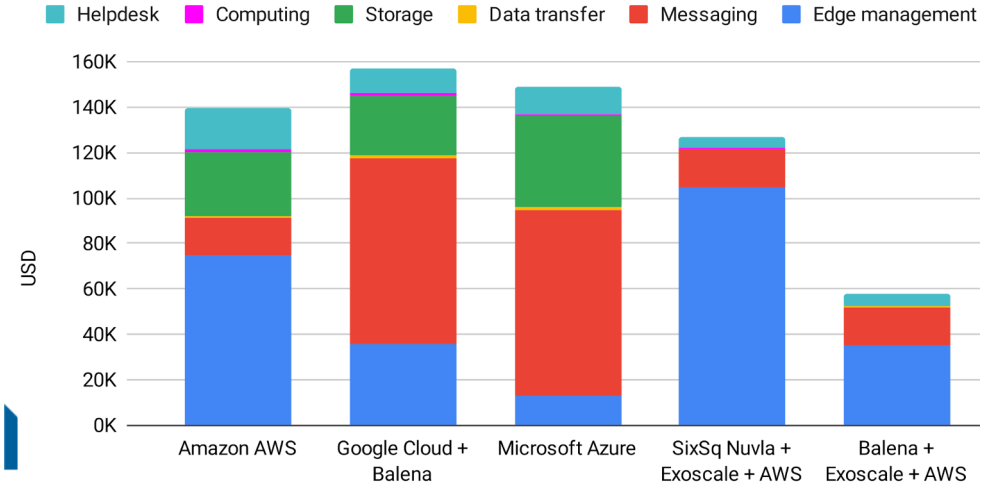
Parameter	Description
`deployment_size`	Number of edge devices deployed
`tmetry_metrics`	Number of different telemetry metrics collected at the edge devices
`tmetry_msg_rate`	Rate at which telemetry messages are sent form the edge application to the Cloud
`tmetry_msg_size`	Size of a telemetry message

Service costs

Edge device management: yearly subscription, device registration fees, connectivity charges	
Messaging: telemetry & “end-user” applications	
Data transfer: cloud-to-edge, edge-to-cloud and intra-cloud	
Storage: space (standard “hot” S3 tier), read and write operations	
Computing: VM, GPU, etc.	
Helpdesk: business/professional tier for 1 user.	

Operating cost comparison: results – 1K/1Y





Road traffic management

- Computing and data transfer ~negligible: why?
  - Network bandwidth is cheap
  - No zone replication (no site redundancy)
  - Optimistic computing model

Smart grid

- Computing and data transfer ~negligible (see above)
- Storage: small footprint + Exoscale (no fees for operations) ⇒ advantage for Balena and SixSq

Plan

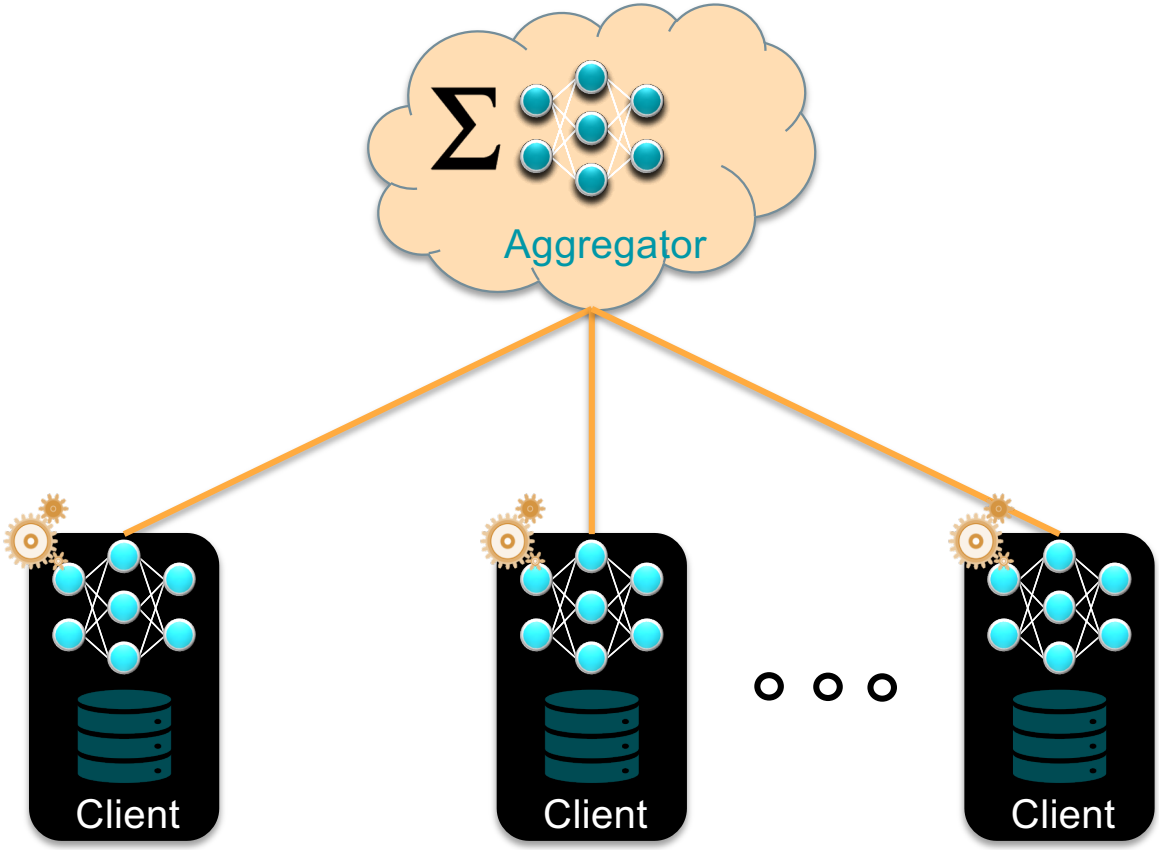
1. Why/What is Edge Computing and Cloud Continuum ?
2. Targeted applications (self-adaptive/context-aware)
3. Use-cases
4. Cloud Continuum (Edge-to-Cloud) solutions
5. **Towards a decentralised Federated Learning based edge-to-edge Continuum Computing framework.**
6. The Smart Grid Application

Why Edge-To-Edge ?

1. Unload the cloud.
2. Limit the traffic between edge devices and the cloud
3. Keep intelligence improvement as close as possible to the Edge devices
4. Improve the intelligence of edges by getting them communicate with each other

… And here where the Federated Learning comes in.

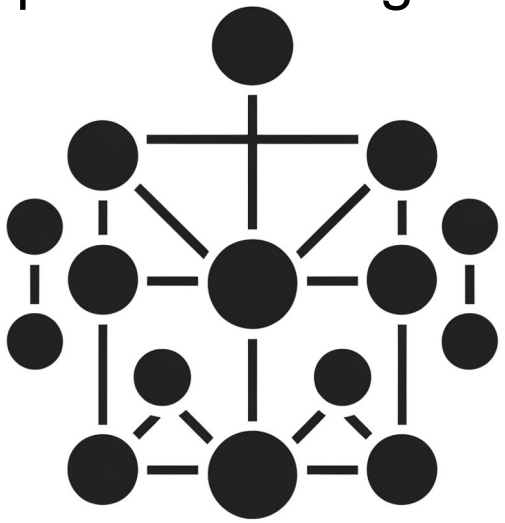
Federated Learning



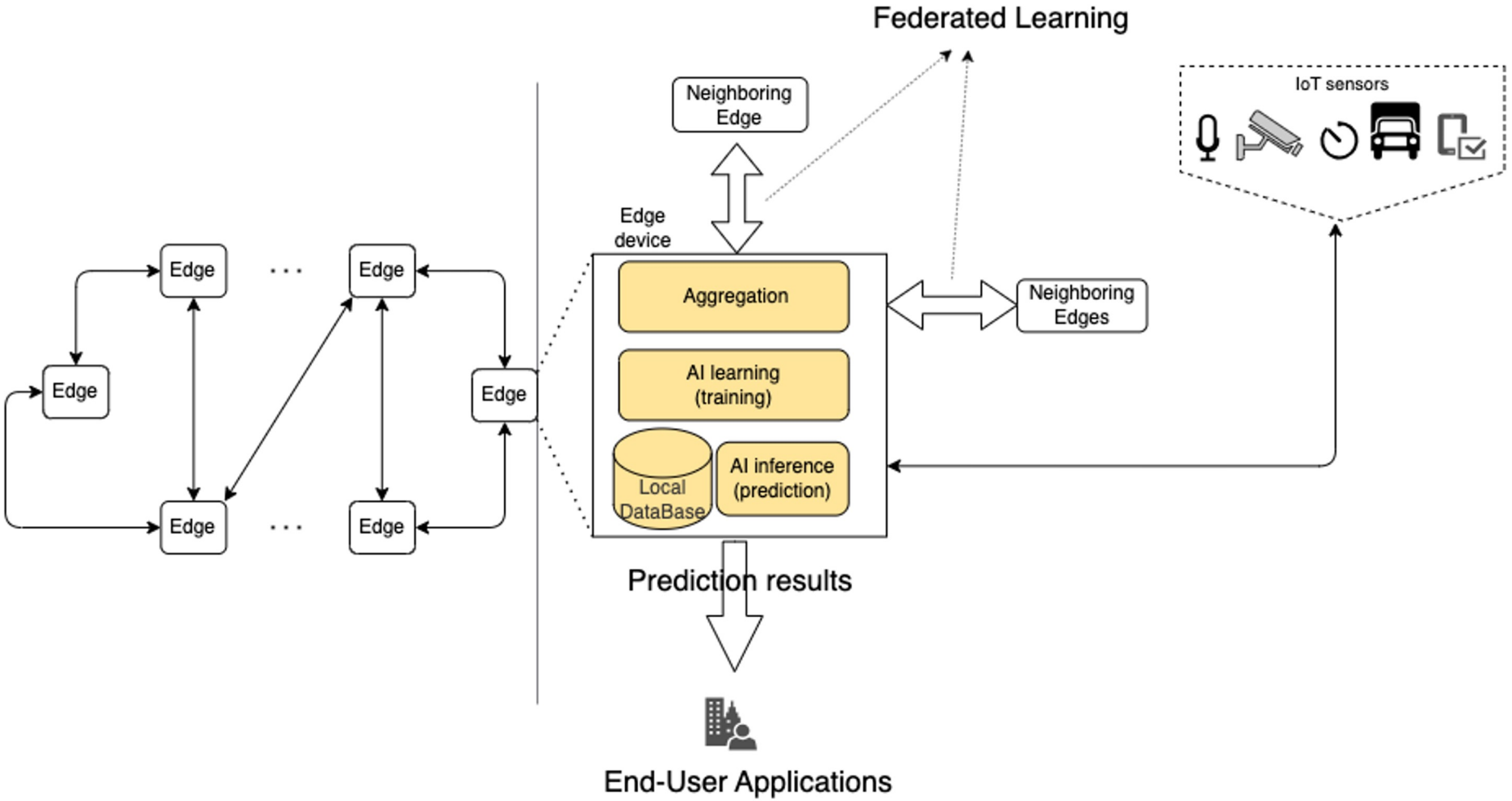
However …

1. Scalability problem: Centralised solutions are not efficient when the number of edges increases
2. Aggregating with all nodes (having different contexts) makes no sense.
   a. → Aggregation must be done between “similar” nodes (neighbours)
   b. → Need to define a neighbourhood relationship between edges

… And here where the Distributed Federated Learning comes in.



Distributed Federated Learning



Distributed Federated Learning (A first draft algorithm)

Initiate (or not) neighbours

While *true*

1. Collect data/measures
2. Prediction/Classification
3. Update (or not) the list of neighbours
4. Communicate (or not) with neighbours and retrieve their weights
5. Aggregation (how?). Update (or not) local NN configuration (weights)
6. Start (or not) a new training to generate a new local MLM

End While

Who are our neighbours ?

How and when to update them?

When should we communicate with them?

How to aggregate ?

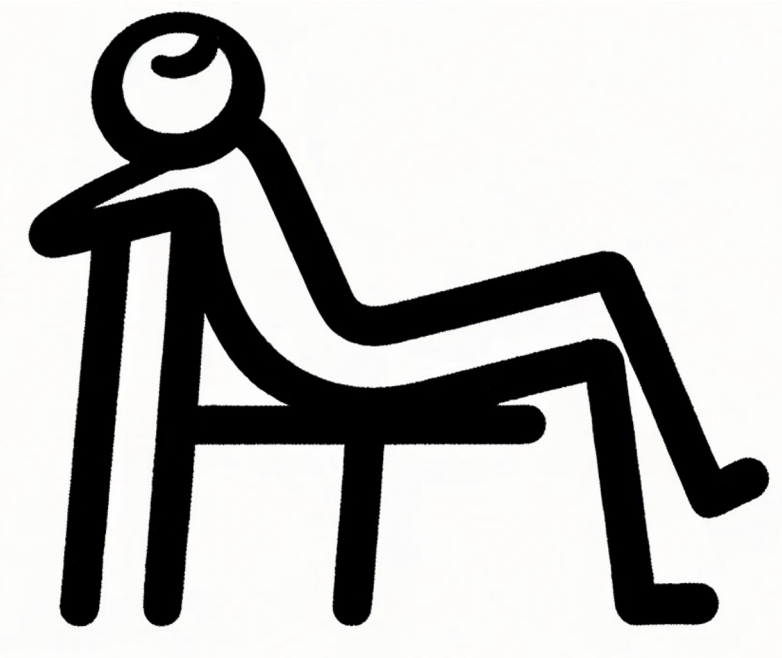
When starting a new training?

When updating the NN?

Open questions

1. Who are our neighbours ?
2. How to update them?
3. **When should we communicate with our neighbours?**
4. How to calculate aggregation?
5. When should we aggregate (update the NN configuration: weight)?
6. When should we launch a new training?

When should we communicate with our neighbours?



I feel good !!

I feel bad !!

The good, the bad and the ugly …

Clint Eastwood Lee Van Cleef Eli Wallach

The “Good” scenario algorithm

What is a good performance?

Initiate neighbours

While *true*

1. Collect data/measure
2. Result = Prediction (collected data)
3. If “Good performance (Result)” then send ***My_Weights*** to my neighbours
4. When receiving a “***My_Weights”*** message from a given neighbour then (1) Make aggregation, (2) initiate learning and (3) update the local Neural Network

End While

Should we receive from all neighbours before executing (1), (2) and (3)?

The “bad” scenario algorithm

Ask for a help!!

Initiate neighbours

While *true*

1. Collect data/measure
2. Result = Prediction (collected data)
3. If “low performance (Result)” then send ***Weights_Request*** to my neighbours
4. When receiving a message from a given neighbour
   a. if type (message) == ***Weights_Request*** then send ***My_Weights*** to the neighbour
   b. If type (message) == ***My_Weights*** then make aggregation, initiate learning and update the local Neural Network

Plan

1. Why/What is Edge Computing and Cloud Continuum ?
2. Targeted applications (self-adaptive/context-aware)
3. Use-cases
4. Cloud Continuum (Edge-to-Cloud) solutions
5. Towards a decentralised Federated Learning based edge-to-edge Continuum Computing framework.
6. **The Smart Grid Application**

SWARM Project

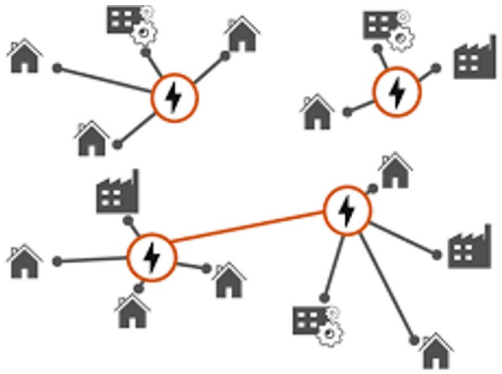
Smart and **W**idely-distributed **A**ppliances for **R**enewable energy **M**anagement: Market driven project

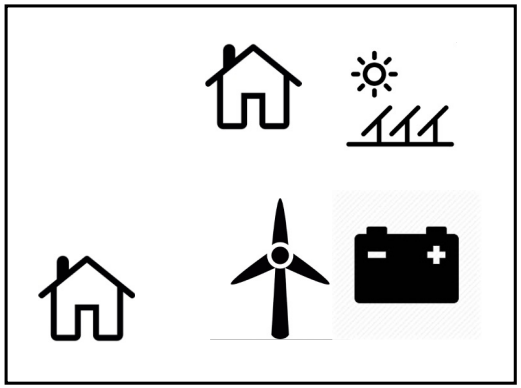
LASAGNE Project

digita**L** fr**A**mework for **S**m**A**rt **G**rid and

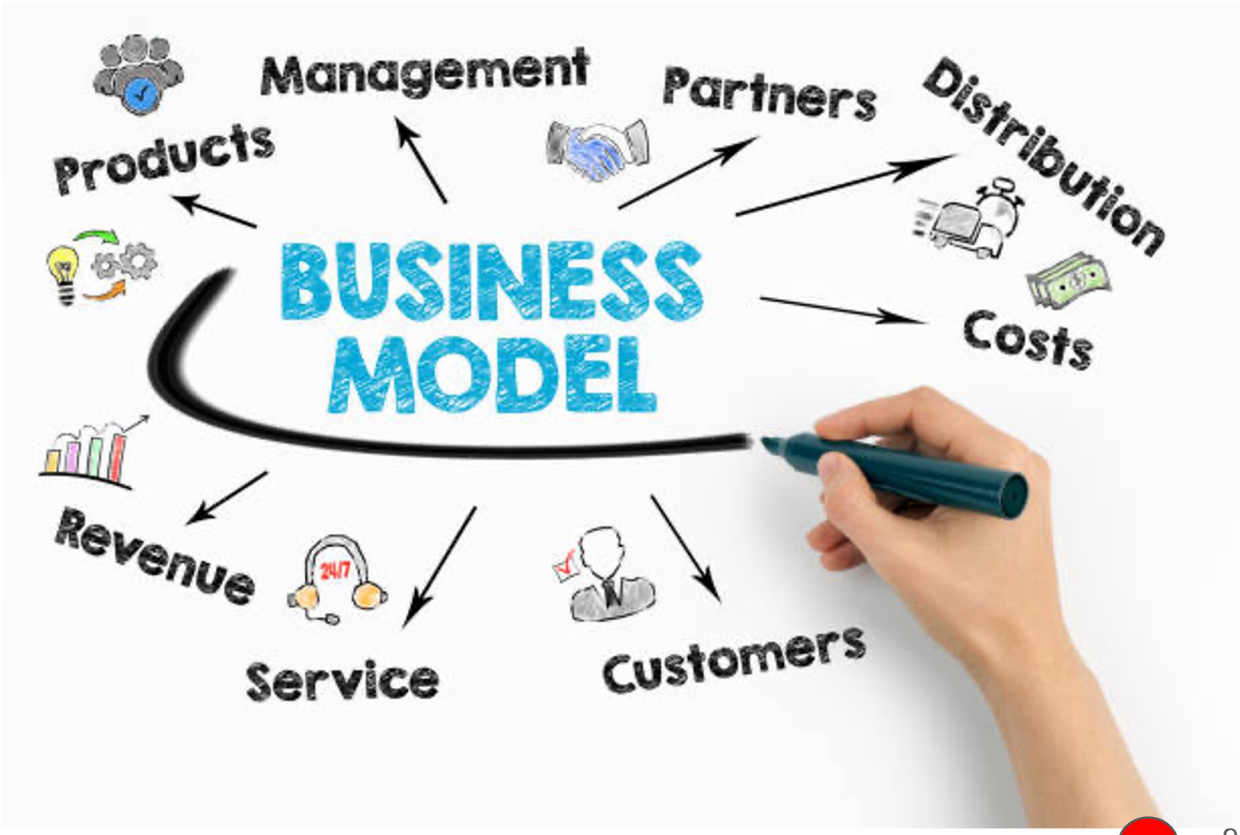
re**N**ewable **E**nergy: research project

The context

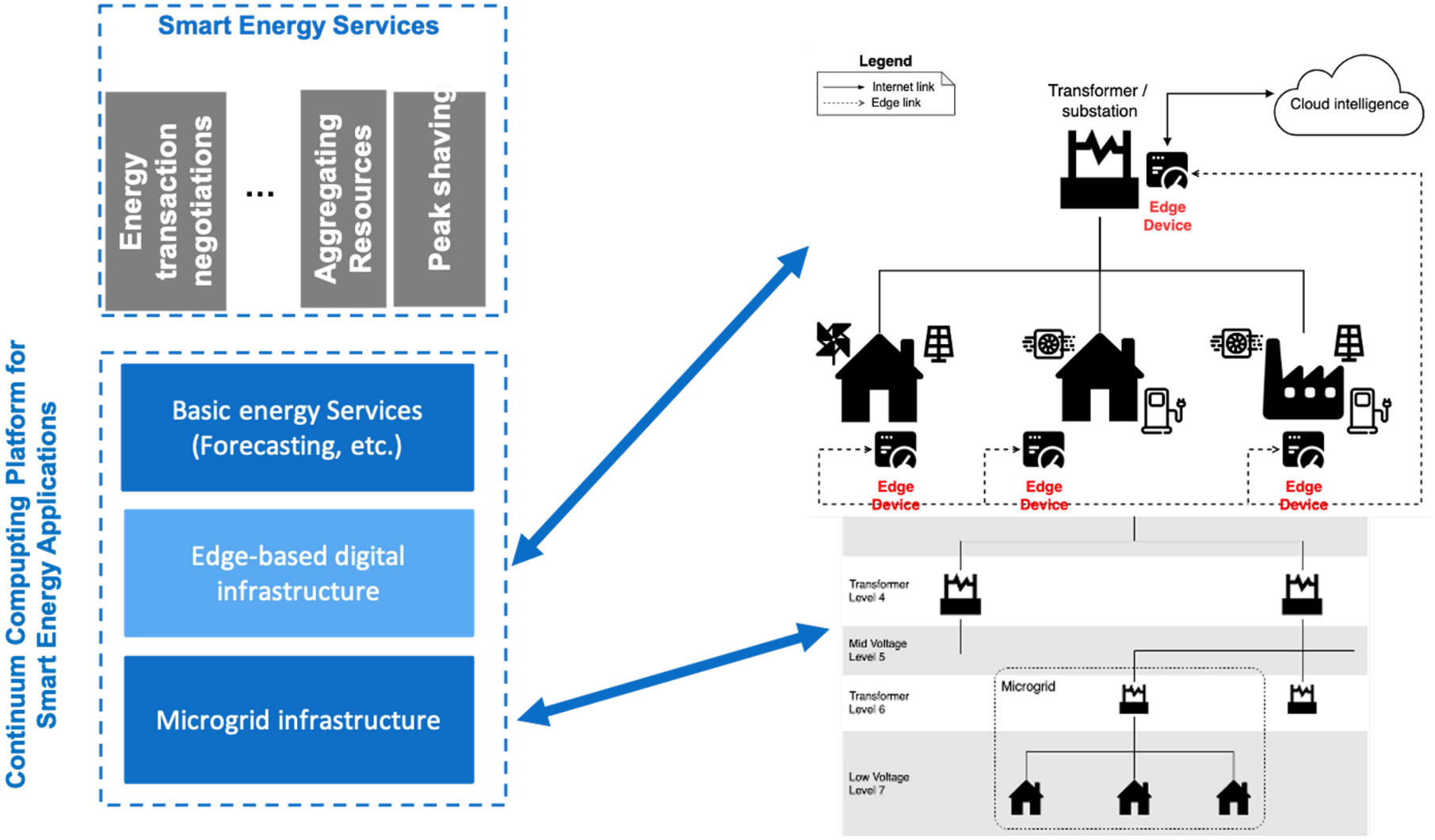




Smart Energy services



Edge based smart energy Services

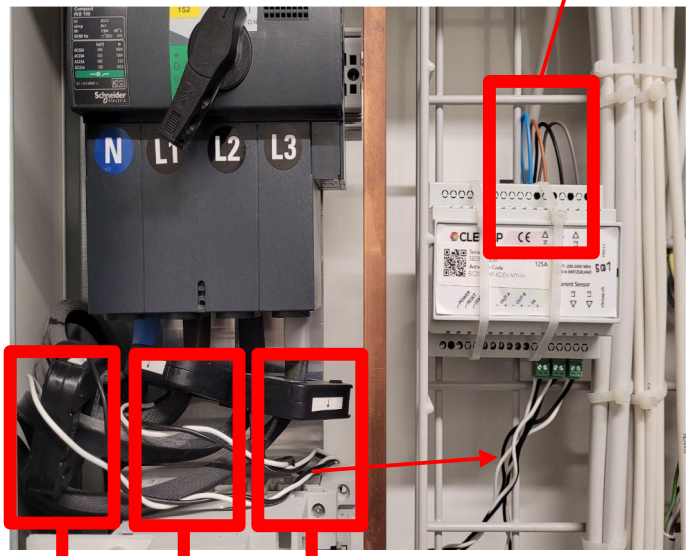


CLEMAP as edge device

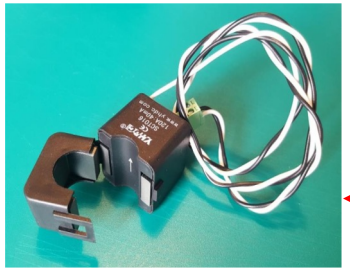
Linux (Raspberry pi 3)

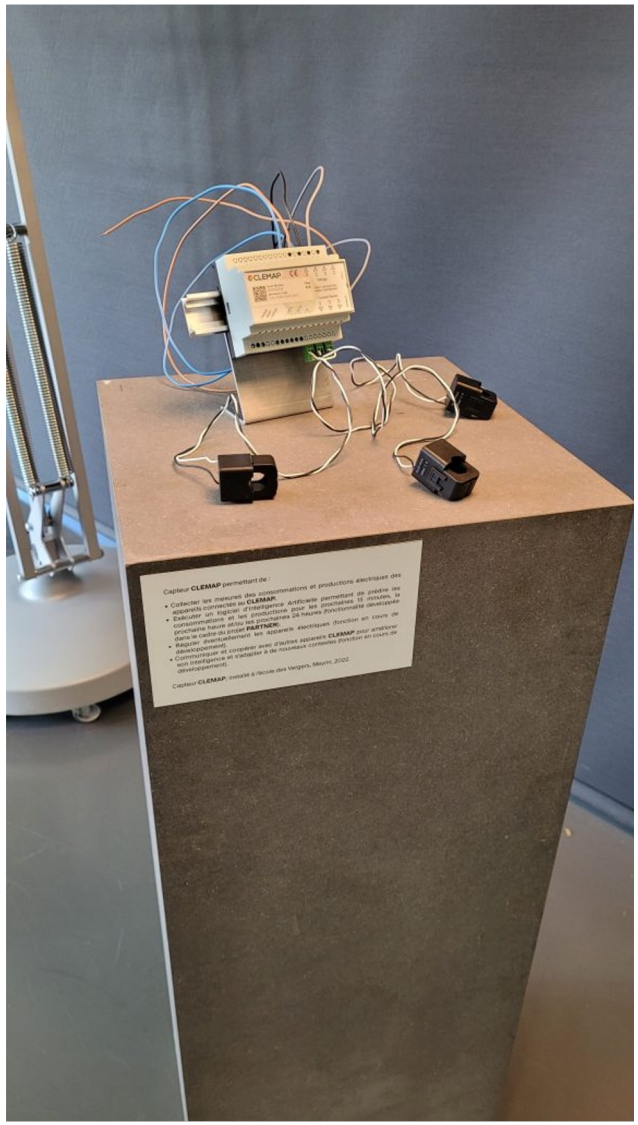
Sensor : Voltage, Current (3-phases)

Voltage sensors

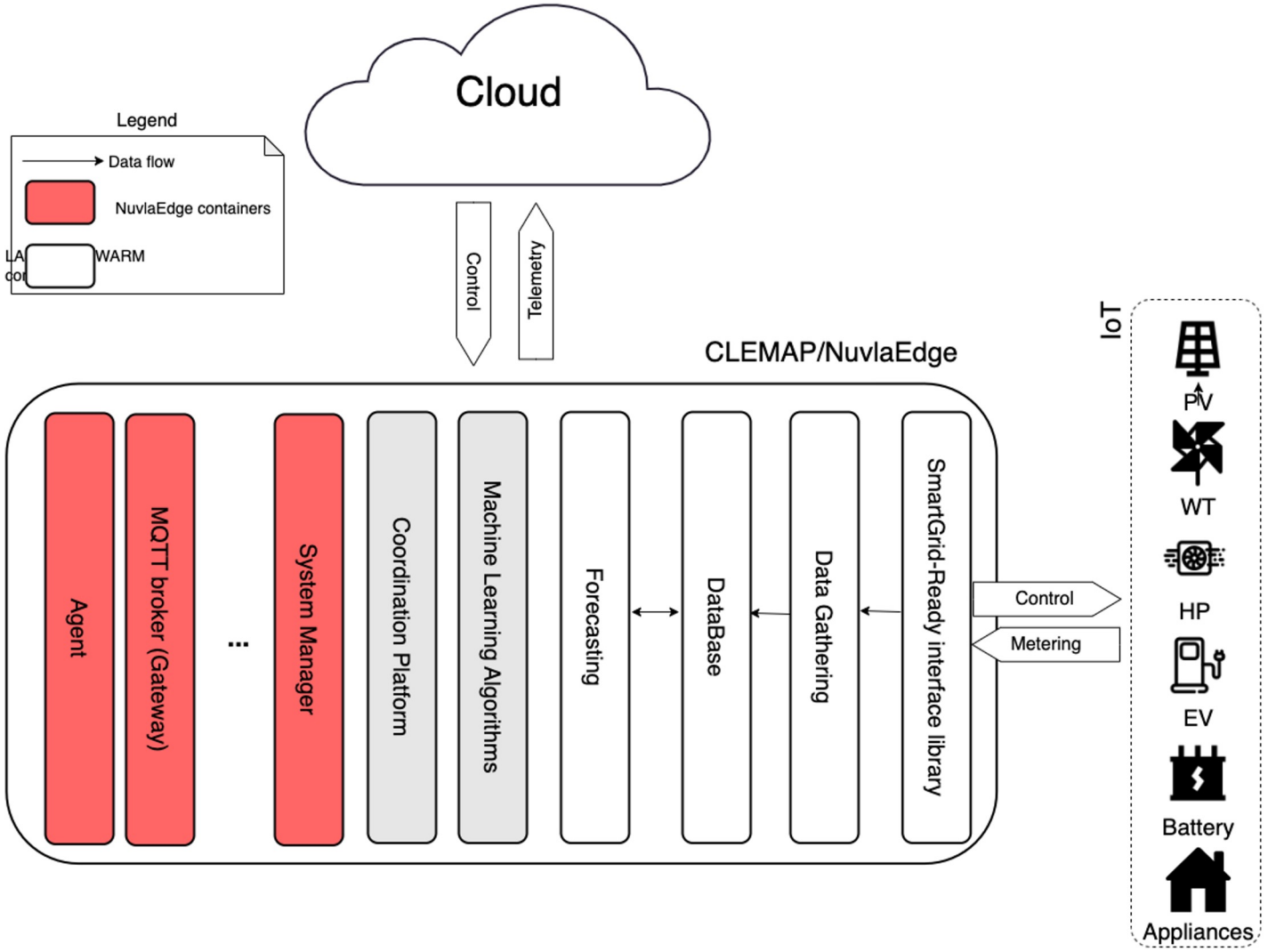


Current sensors

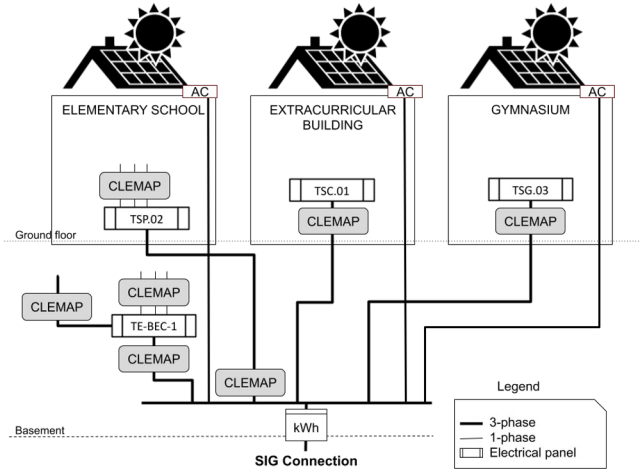


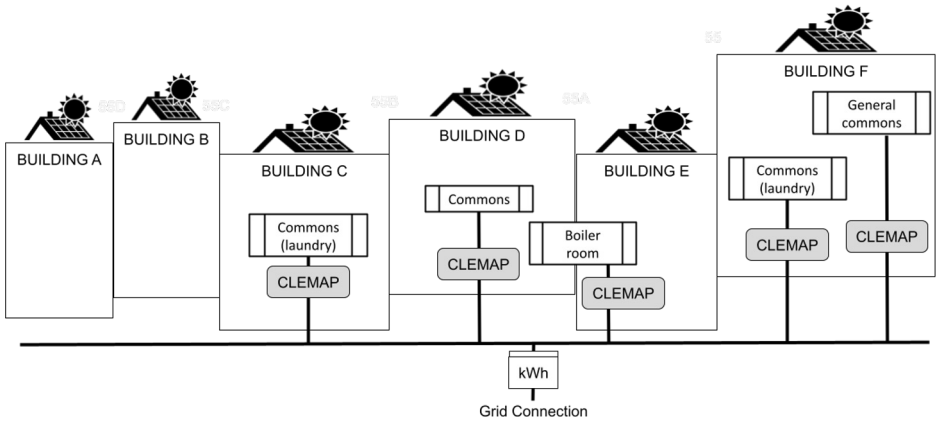


The edge device: What is on board?



Deployments (Vergers School, Chêne-Bougeries)





Polygones: A local energy community

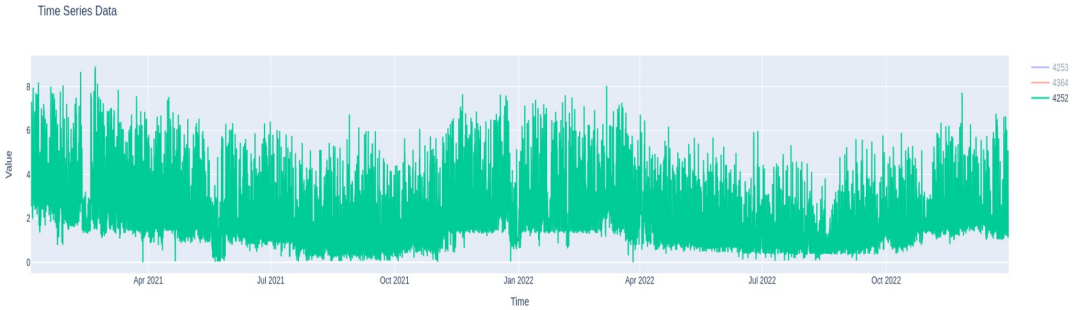
- 5-floor building with around 28 apartments
- Planned in Sept - Dec. 2024

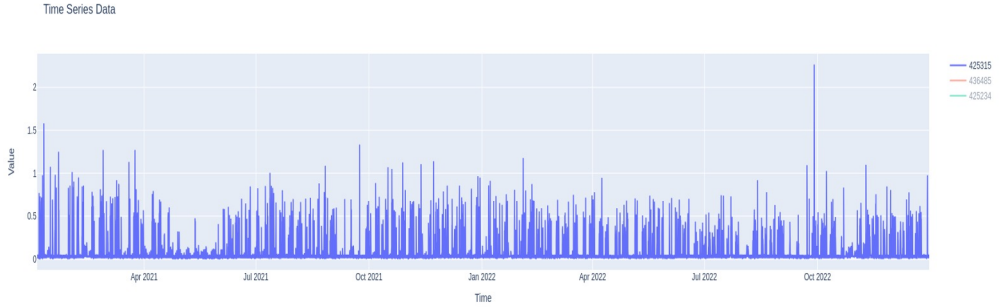
ML algorithms

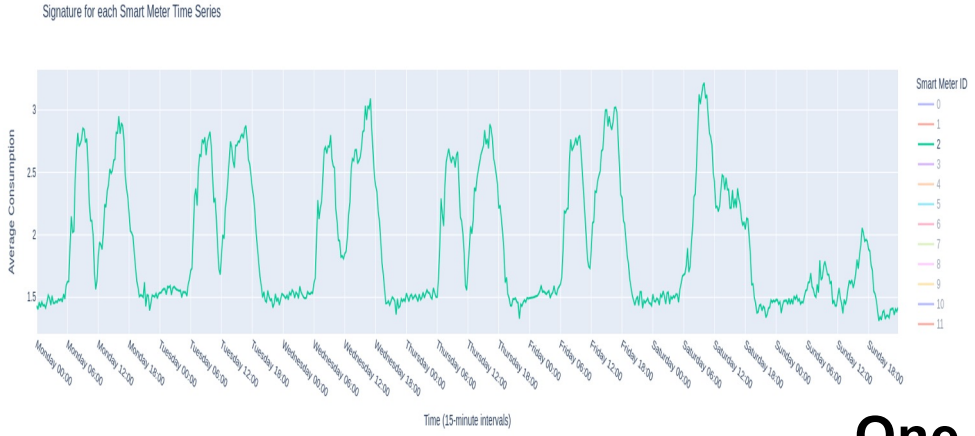
1. XGBoost
2. **LSTM**
3. **CNN-LSTM**
4. **CNN-LSTM with attention mechanism**
5. Transformer
6. Prophet
7. DeepAR

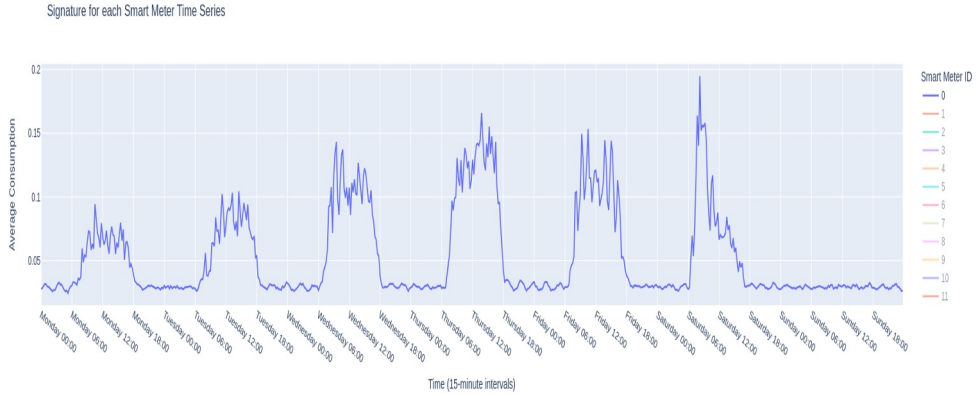
Signatures, Neighbourhood

Raw data (1.5 year :April 2021 - October 2022)



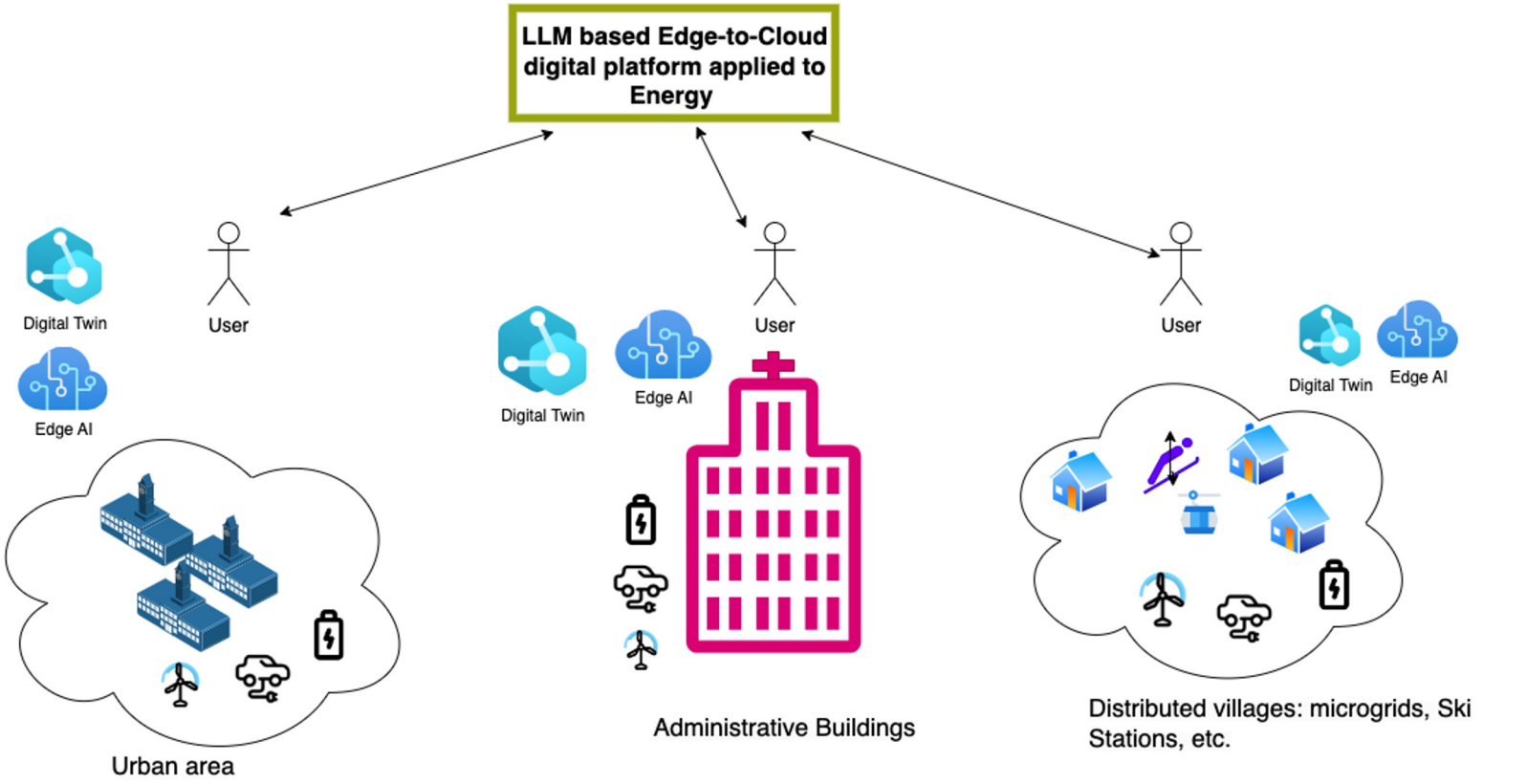






One week

The future …


LLM based Edge-to-Cloud digital platform applied to Energy
Digital Twin
Edge AI
User
Digital Twin
Edge AI
User
User
Digital Twin
Edge AI
Urban area
Administrative Buildings
Distributed villages: microgrids, Ski Stations, etc.
Cloud Continuum Platform for Smart Energy Services
Cloud Continuum Platform for Smart Energy Services
Cloud Continuum Platform for Smart Energy Services

Thank you !