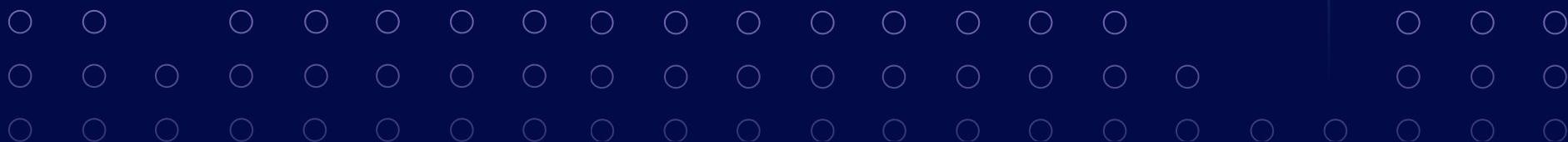


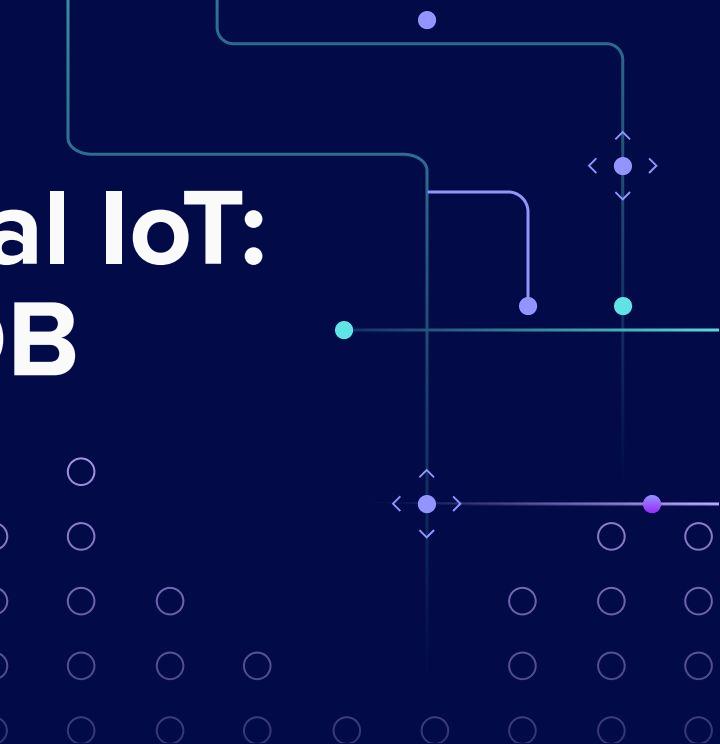


# Physical AI for Industrial IoT: Edge Impulse + InfluxDB

October 21, 2025



6.4830624082  
6.468081871635  
8.484777408835.  
8.484435921435.14  
8.488697114 35.1669  
8.485292383495.163143





**Marc Pous**  
Developer Relations,  
Edge Impulse



**Suyash Joshi**  
Senior Developer Advocate,  
InfluxDB

# Agenda

1. Introduction
2. Physical AI for Industrial IoT
3. Edge Impulse Platform
4. InfluxDB 3 for Physical AI
5. Demo
6. Q&A

# Definitions

## Edge AI

AI deployed locally on devices  
*(IoT, sensors, robots)*  
for real-time processing.

## Physical AI

AI integrated into physical systems  
*(robots, machines)*  
to interact with the real world in real-time.

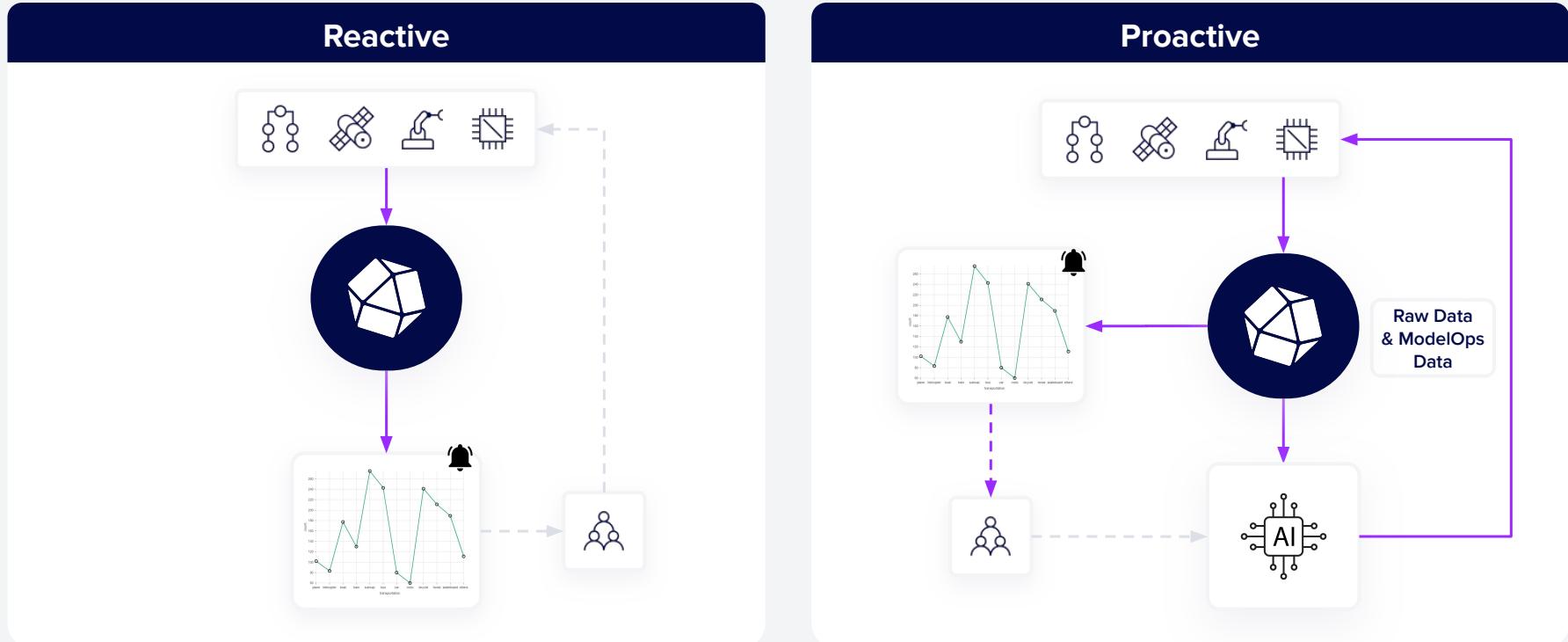
## Embodied AI

AI integrated into physical systems  
*(robots, drones)*  
that learn through environment interaction.

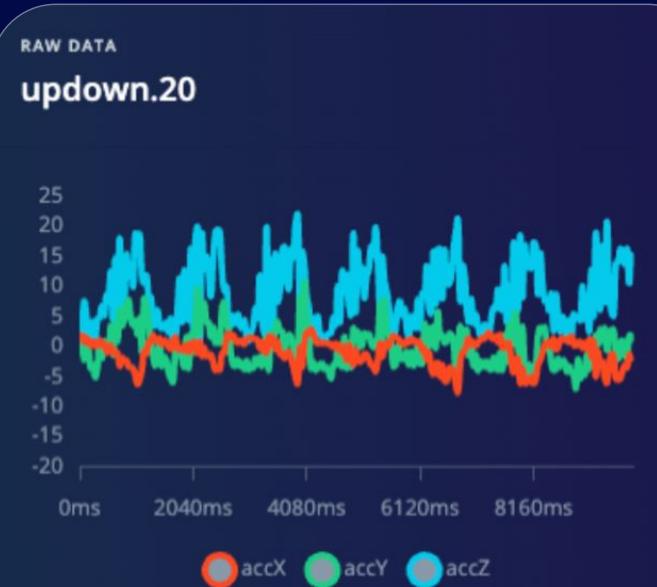
## Time Series Data

A sequence of data points  
index in time order / timestamp  
*(IoT, DevOps etc)*

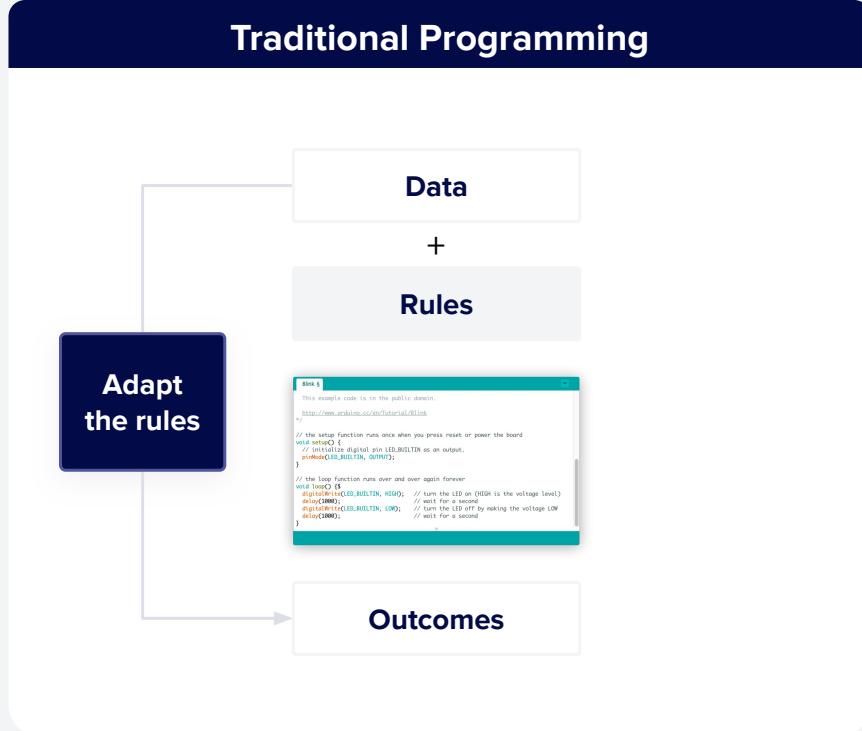
# Physical AI: From Observation to Action



# What does the real world look like?



# A paradigm shift



# A paradigm shift

## Traditional Programming

Data

+

Rules

Adapt  
the rules

```
Sketch: 1
This example code is in the public domain.
http://www.arduino.cc/en/Tutorial/Blink

/*
 * the setup function runs once when you press reset or power the board
 * initializes the digital pin LED_BUILTIN as an output.
 */
pinMode(LED_BUILTIN, OUTPUT);

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

Outcomes

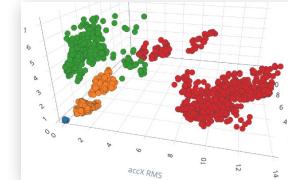
## Machine Learning

Data

+

Outcomes

Collect  
more data



Rules

# Accelerating industrial productivity use cases

Inspect production lines, predict failures and forecast maintenance cycles

Images

Vibration

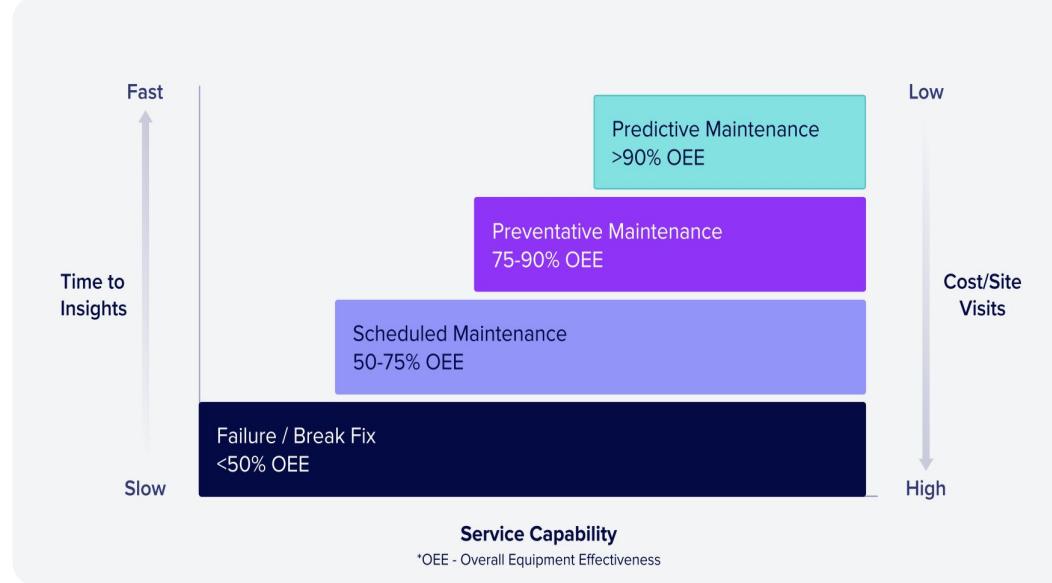
Sound

Temperature

Current



# Use Case: Predictive Maintenance

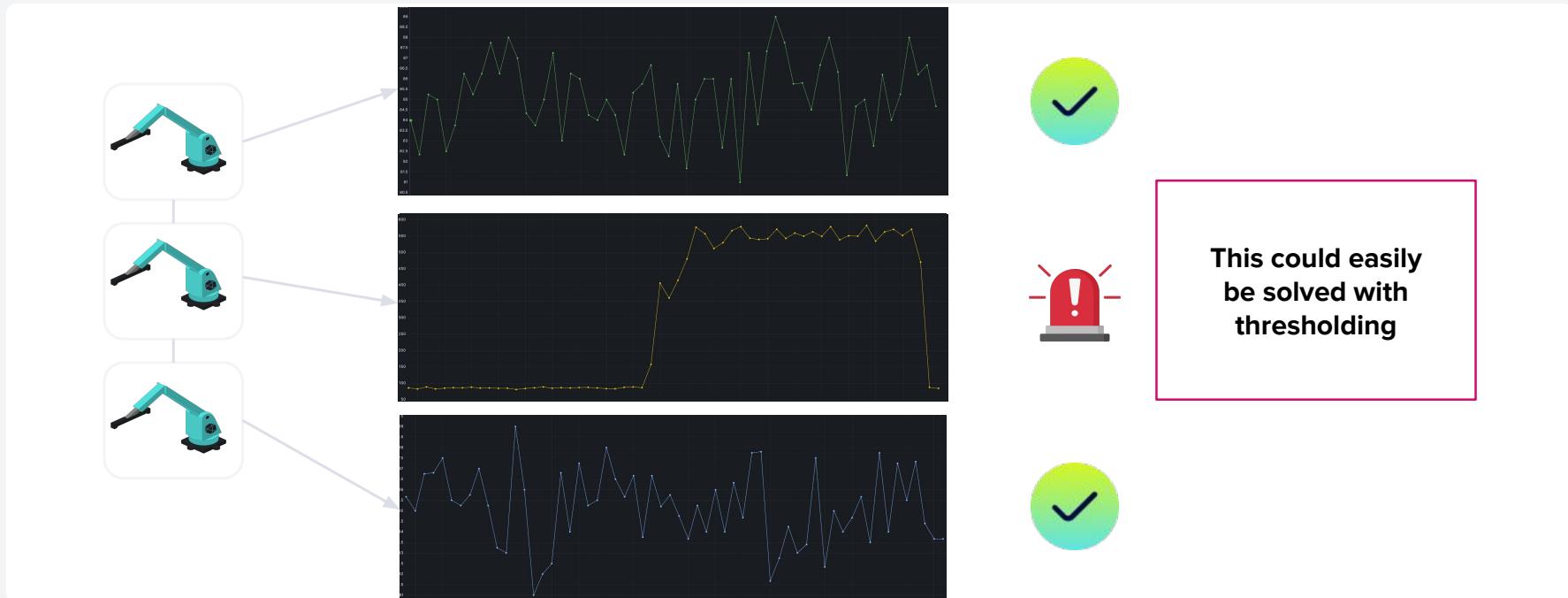


## Machine Learning for Time Series Data:

- Forecasting
- Anomaly Detection

# Predictive Maintenance

## Naive Approach



# Predictive Maintenance

ML based Approach



What do we do when  
our result becomes  
unpredictable by  
conventional means?

# Challenges Implementing Physical AI

**99% of sensor & device data is discarded\***

Accelerate next gen IoT products and solutions turning data from any edge device into actionable insights

**Current ML tools are not designed for edge**

Enable developers enterprise-wide to rapidly build ML solutions natively for edge devices, amplifying your ML experts

**87% of AI/ML projects fail due to infrastructure complexity**

MLOps infrastructure from data collection to edge deployment, for enterprise data science and ML teams, with leading integrations

**Difficulties in building smart product experiences**

Build innovative experiences with ML with ease, leading to growth and competitive differentiation

# What does the real world look like?

Edge AI development platform helping your teams to collect **data**, develop **any model** and deploy to **any target**.



Build real-world  
datasets at scale



Develop custom ML  
solutions fast



Deploy intelligent edge  
products



# Active learning with Edge Impulse & InfluxDB



# Active learning with Edge Impulse & InfluxDB

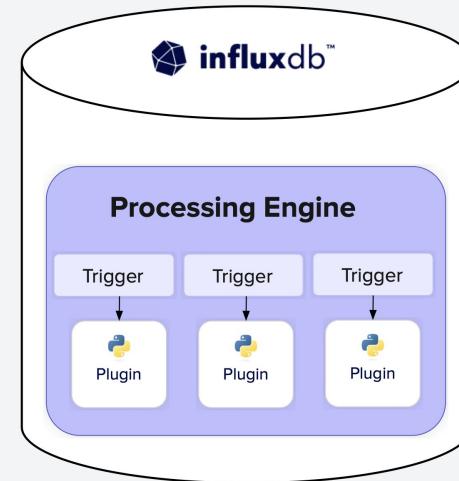


# InfluxDB 3 | The Processing Engine

A fully modern embedded virtual **python environment directly inside the database** for efficiency and ease of use.

Access to Python's **ecosystem** of libraries and tools.

Ideal for **data processing, transformation, alerting, monitoring, anomaly detection, forecasting** and more.

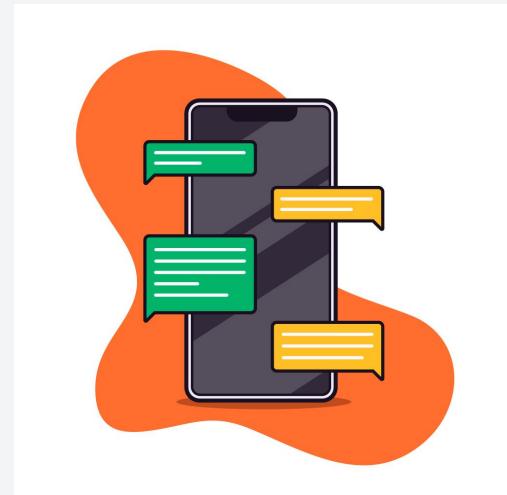
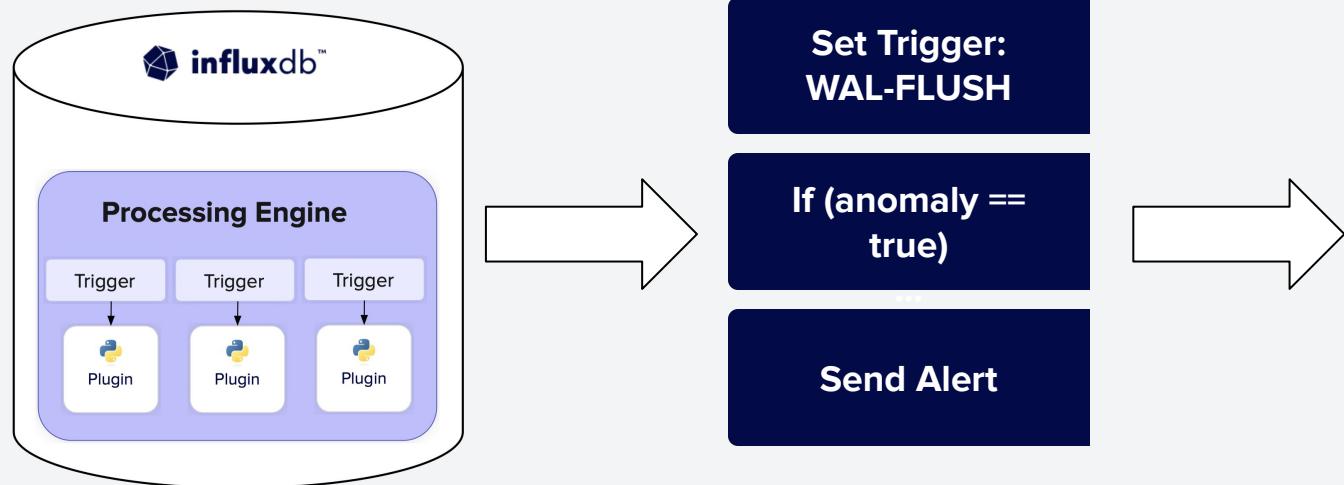


**WAL Flush**

**Scheduled Task**

**On Request**

# Alerting | InfluxDB 3 Processing Engine



# How InfluxDB turns raw signals into intelligent action

**Automate and Predict:** Trigger real-time actions through in-database Python processing. Open standards & Iceberg export make data available to advanced analytics & machine learning systems.

**Ingest:** Stream and ingest millions of multi-source data points per second across edge, on-prem, and cloud deployments.

**Store:** Make decades of high-fidelity data affordable to retain via low-cost object storage and flexible retention policies.

**Query:** Get immediate responses at scale, even with high-cardinality data, via columnar engine and SQL.

Built on an enterprise-grade foundation

✓ High availability

✓ Horizontal scaling

✓ Open interoperability

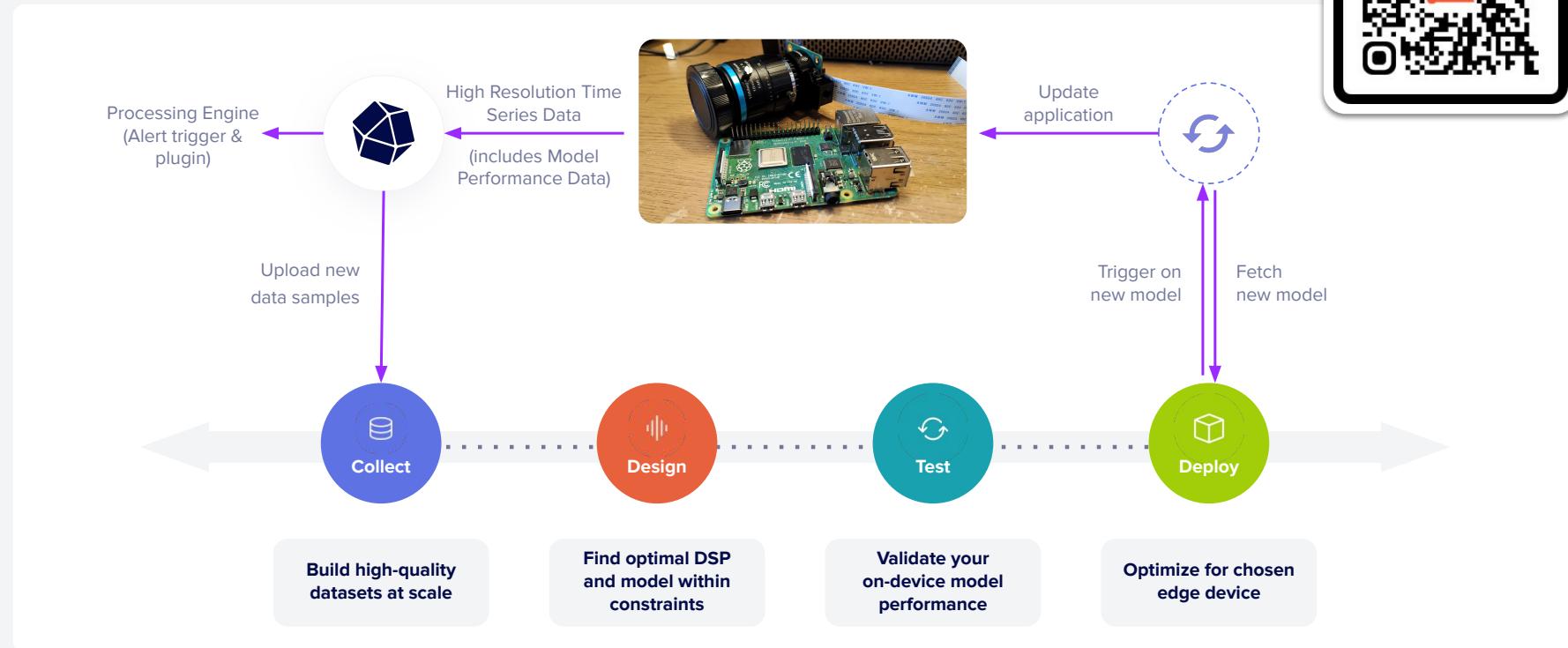
✓ Enterprise security

✓ Enterprise support

# Demo



# Demo App Architecture



# Using Arduino UNO Q + Edge Impulse + InfluxDB



# Using Arduino UNO Q + Edge Impulse + InfluxDB

The screenshot shows a dual-pane interface. On the left, a terminal window displays a long JSON string of bounding box detections. On the right, a video feed from a camera shows several small components (red, yellow, green) scattered on a surface. Blue boxes with labels indicate the detected objects and their confidence scores.

Terminal Output (JSON Bounding Boxes):

```
[{"bel": "Yellow", "value": 0.63164625453949, "width": 16, "x": 56, "y": 64}, {"height": 8, "label": "Red", "value": 0.581828678501709, "width": 8, "x": 24, "y": 72}], [{"height": 8, "label": "Red", "value": 0.629278140658054, "width": 8, "x": 24, "y": 8}, {"height": 8, "label": "Yellow", "value": 0.532643719863892, "width": 8, "x": 56, "y": 32}, {"height": 8, "label": "Yellow", "value": 0.52798694770813, "width": 8, "x": 24, "y": 0}, {"height": 8, "label": "Green", "value": 0.6820173976074, "width": 8, "x": 16, "y": 16}, {"height": 16, "label": "Green", "value": 0.76806924128418, "width": 8, "x": 24, "y": 24}, {"height": 8, "label": "Green", "value": 0.581197262289568, "width": 16, "x": 48, "y": 48}, {"height": 8, "label": "Yellow", "value": 0.6669759222668, "width": 16, "x": 48, "y": 24}, {"height": 8, "label": "Yellow", "value": 0.62021778318833, "width": 8, "x": 48, "y": 32}, {"height": 8, "label": "Yellow", "value": 0.62021778318833, "width": 8, "x": 80, "y": 32}, {"height": 8, "label": "Green", "value": 0.6820173976074, "width": 8, "x": 48, "y": 40}, {"height": 16, "label": "Green", "value": 0.631941396980286, "width": 8, "x": 16, "y": 40}, {"height": 16, "label": "Green", "value": 0.89104616641998276, "width": 16, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.631941396980286, "width": 8, "x": 16, "y": 40}, {"height": 8, "label": "Green", "value": 0.531630158423774, "width": 8, "x": 24, "y": 48}, {"height": 16, "label": "Yellow", "value": 0.68160359859666, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Yellow", "value": 0.6463371515274048, "width": 8, "x": 8, "y": 72}, {"height": 8, "label": "Green", "value": 0.6282328963279724, "width": 8, "x": 48, "y": 72}], [{"height": 8, "label": "Yellow", "value": 0.589341163632539, "width": 16, "x": 16, "y": 0}, {"height": 8, "label": "Red", "value": 0.5676714181900824, "width": 8, "x": 24, "y": 0}, {"height": 8, "label": "Green", "value": 0.52798694770813, "width": 8, "x": 24, "y": 8}, {"height": 8, "label": "Yellow", "value": 0.532643719863892, "width": 8, "x": 56, "y": 32}, {"height": 8, "label": "Yellow", "value": 0.52798694770813, "width": 8, "x": 24, "y": 0}, {"height": 8, "label": "Green", "value": 0.6820173976074, "width": 8, "x": 16, "y": 16}, {"height": 16, "label": "Green", "value": 0.76806924128418, "width": 8, "x": 24, "y": 24}, {"height": 8, "label": "Green", "value": 0.581197262289568, "width": 16, "x": 48, "y": 48}, {"height": 8, "label": "Yellow", "value": 0.6669759222668, "width": 16, "x": 48, "y": 24}, {"height": 8, "label": "Yellow", "value": 0.62021778318833, "width": 8, "x": 48, "y": 32}, {"height": 8, "label": "Yellow", "value": 0.62021778318833, "width": 8, "x": 80, "y": 32}, {"height": 8, "label": "Green", "value": 0.6820173976074, "width": 8, "x": 48, "y": 40}, {"height": 16, "label": "Green", "value": 0.631941396980286, "width": 8, "x": 16, "y": 40}, {"height": 16, "label": "Green", "value": 0.89104616641998276, "width": 16, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.631941396980286, "width": 8, "x": 16, "y": 40}, {"height": 8, "label": "Green", "value": 0.531630158423774, "width": 8, "x": 24, "y": 48}, {"height": 16, "label": "Yellow", "value": 0.681603598596666, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Yellow", "value": 0.6463371515274048, "width": 8, "x": 8, "y": 72}, {"height": 8, "label": "Green", "value": 0.6282328963279724, "width": 8, "x": 48, "y": 72}], [{"height": 8, "label": "Yellow", "value": 0.76806924128418, "width": 16, "x": 0, "y": 0}, {"height": 8, "label": "Green", "value": 0.532643719863892, "width": 8, "x": 0, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.77870388401835, "width": 8, "x": 8, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.77228780212402, "width": 8, "x": 24, "y": 0}, {"height": 8, "label": "Yellow", "value": 0.6669516858100891, "width": 8, "x": 48, "y": 16}, {"height": 8, "label": "Green", "value": 0.4386685551719666, "width": 8, "x": 48, "y": 40}, {"height": 16, "label": "Green", "value": 0.909573078155176, "width": 16, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.896406421246338, "width": 8, "x": 48, "y": 40}, {"height": 8, "label": "Red", "value": 0.5880738228859904, "width": 8, "x": 0, "y": 40}, {"height": 16, "label": "Yellow", "value": 0.7222641801858106, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.5418087704658580, "width": 8, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.809196985174255, "width": 24, "x": 16, "y": 0}, {"height": 8, "label": "Green", "value": 0.66032165642804, "width": 8, "x": 8, "y": 16}, {"height": 8, "label": "Green", "value": 0.739825937211182, "width": 8, "x": 24, "y": 16}, {"height": 24, "label": "Yellow", "value": 0.76558203869505, "width": 8, "x": 56, "y": 32}, {"height": 16, "label": "Green", "value": 0.936019833221436, "width": 8, "x": 40, "y": 40}, {"height": 8, "label": "Red", "value": 0.56438668562164387, "width": 8, "x": 0, "y": 40}, {"height": 16, "label": "Yellow", "value": 0.7260077395710754, "width": 8, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.6818941631588745, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.53490757594219971, "width": 8, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.782993078218115, "width": 8, "x": 0, "y": 0}, {"height": 8, "label": "Yellow", "value": 0.78078115310669, "width": 16, "x": 0, "y": 16}, {"height": 8, "label": "Green", "value": 0.5055442033783, "width": 8, "x": 48, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.76558203869505, "width": 8, "x": 56, "y": 32}, {"height": 16, "label": "Green", "value": 0.936019833221436, "width": 8, "x": 40, "y": 40}, {"height": 8, "label": "Red", "value": 0.56438668562164387, "width": 8, "x": 0, "y": 40}, {"height": 16, "label": "Yellow", "value": 0.7260077395710754, "width": 8, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.6818941631588745, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.53490757594219971, "width": 8, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.7283225620005, "width": 8, "x": 24, "y": 0}, {"height": 8, "label": "Yellow", "value": 0.79005402293843, "width": 16, "x": 8, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.5055310889722888, "width": 8, "x": 48, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.74192698403842, "width": 8, "x": 64, "y": 32}, {"height": 16, "label": "Green", "value": 0.9045254588127136, "width": 8, "x": 40, "y": 40}, {"height": 8, "label": "Red", "value": 0.558822131156921, "width": 8, "x": 0, "y": 40}, {"height": 16, "label": "Yellow", "value": 0.722912073135376, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.53490757594219971, "width": 8, "x": 48, "y": 40}, {"height": 8, "label": "Yellow", "value": 0.7283225620005, "width": 8, "x": 24, "y": 0}, {"height": 8, "label": "Yellow", "value": 0.79005402293843, "width": 16, "x": 8, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.5055310889722888, "width": 8, "x": 48, "y": 16}, {"height": 8, "label": "Yellow", "value": 0.74192698403842, "width": 8, "x": 64, "y": 32}, {"height": 16, "label": "Green", "value": 0.9045254588127136, "width": 8, "x": 40, "y": 40}, {"height": 8, "label": "Red", "value": 0.558822131156921, "width": 8, "x": 0, "y": 40}, {"height": 16, "label": "Yellow", "value": 0.722912073135376, "width": 16, "x": 32, "y": 64}, {"height": 8, "label": "Green", "value": 0.53490757594219971, "width": 8, "x": 48, "y": 40}], [{"x": 375, "y": 925, "text": "Settings"}, {"x": 685, "y": 925, "text": "Data Collection"}, {"x": 515, "y": 905, "text": "Powered by Edge Impulse"}]
```

# Using Arduino UNO Q + Edge Impulse + InfluxDB

```
# Send to InfluxDB3
# Convert to line protocol
lines = []
for bbox in res["result"]["bounding_boxes"]:
    line = (
        f"table-name,label={bbox['label']} "
        f"height={bbox['height']},width={bbox['width']},x={bbox['x']},y={bbox['y']},value={bbox['value']}"
    )
    lines.append(line)
data = "\n".join(lines)

# Send to InfluxDB 3 Core
response = requests.post(
    "http://localhost:8181/api/v3/write_lp?db=database&precision=auto",
    headers={
        "Authorization": f"Bearer {token}",
        "Content-Type": "text/plain; charset=utf-8"
    },
    data=data
)

print(response.status_code, response.text)
```

# Continue Learning & Sharing



Try InfluxDB 3: <https://www.influxdata.com/downloads>



GitHub (Lab): <https://github.com/InfluxCommunity/Processing-Engine-Training>



Community Forum: <https://community.influxdata.com>



InfluxDB University: <https://university.influxdata.com>



Try Edge Impulse: <https://edgeimpulse.com>

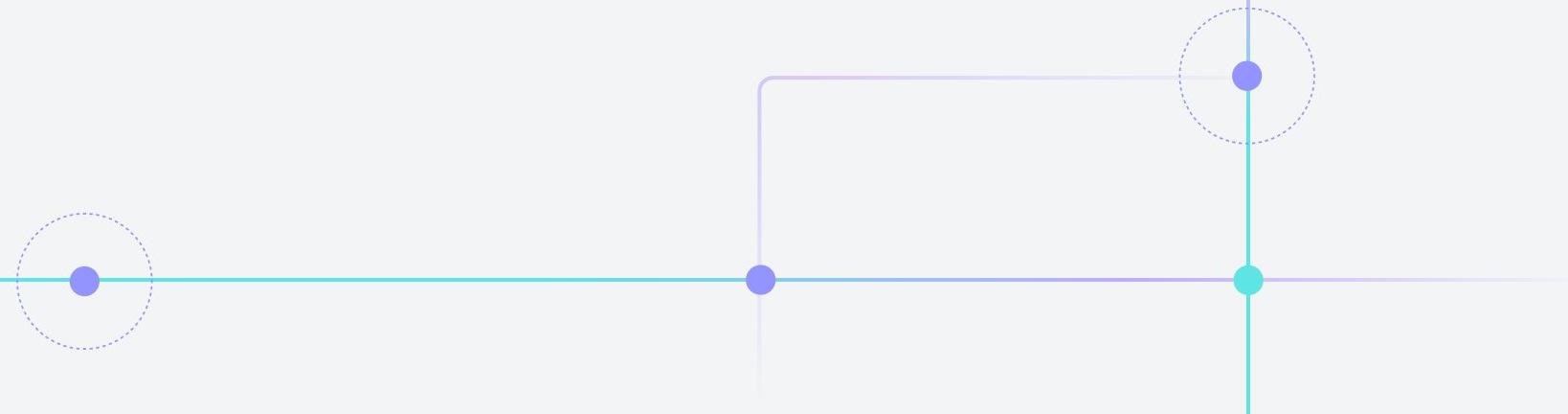


Edge Impulse Forum: <https://forum.edgeimpulse.com>



Edge Impulse Discord: <https://discord.gg/edgeimpulse>

# Questions?





# Thank you

[www.influxdata.com](http://www.influxdata.com)

[www.edgeimpulse.com](http://www.edgeimpulse.com)