Towards a Methodology for Benchmarking Edge Processing Frameworks

Pedro Silva, Alexandru Costan, Gabriel Antoniu
Inria Kerdata, IRISA

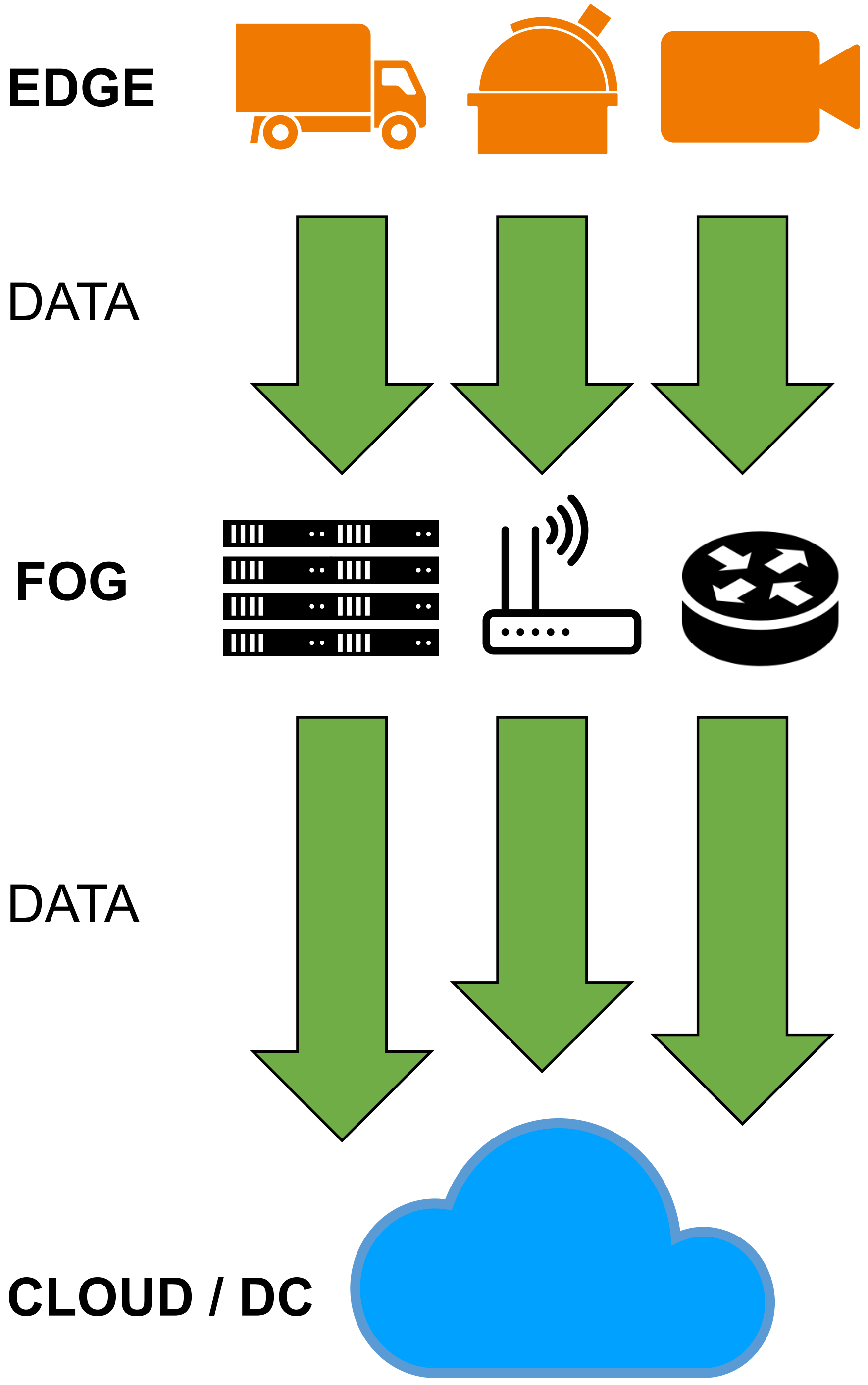
TESLA

JOHN DEERE

Edge processing / computing

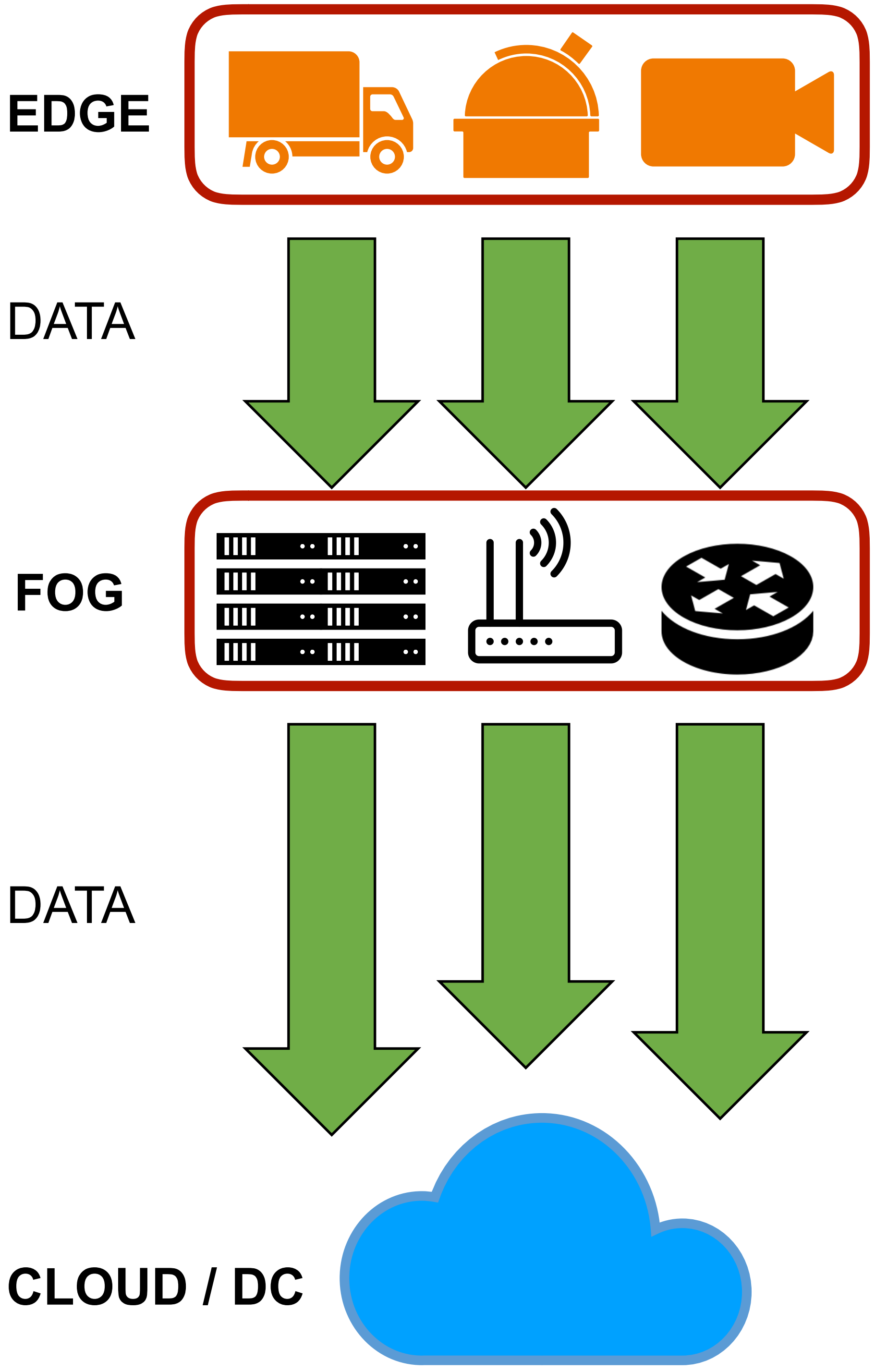
Edge computing advantages:

- easier access to data
- bandwidth saving
- “privacy”
- potential high parallelism

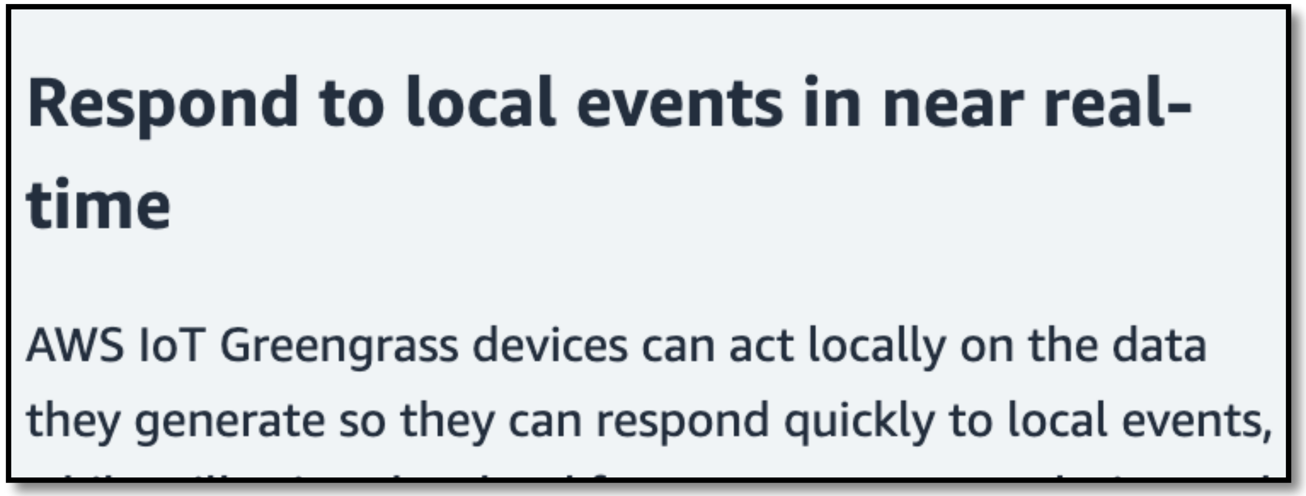


Edge processing tools

- Custom software
- Apache Edgent
- Amazon Greengrass
- Azure Stream Analytics
- IBM Watson IoT
- Intel IoT
- Oracle Edge Analytics
- …

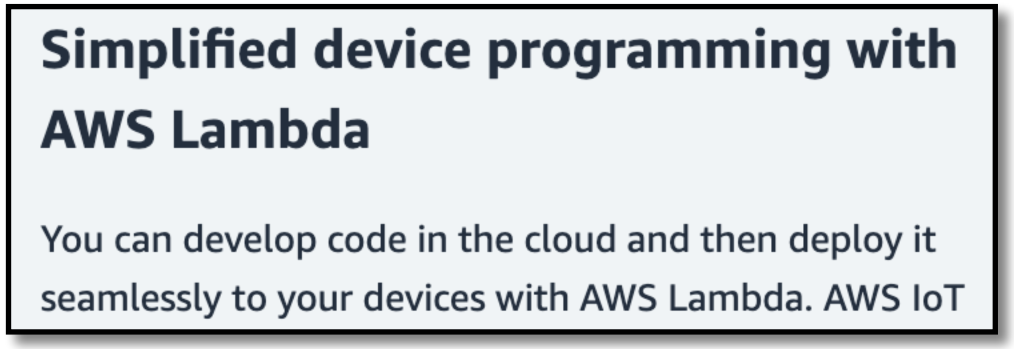


Edge processing tools



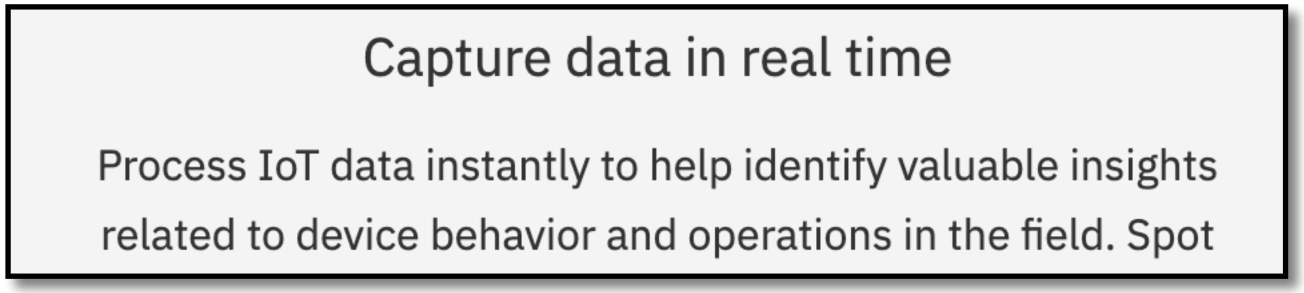
Respond to local events in near real-time

AWS IoT Greengrass devices can act locally on the data they generate so they can respond quickly to local events,



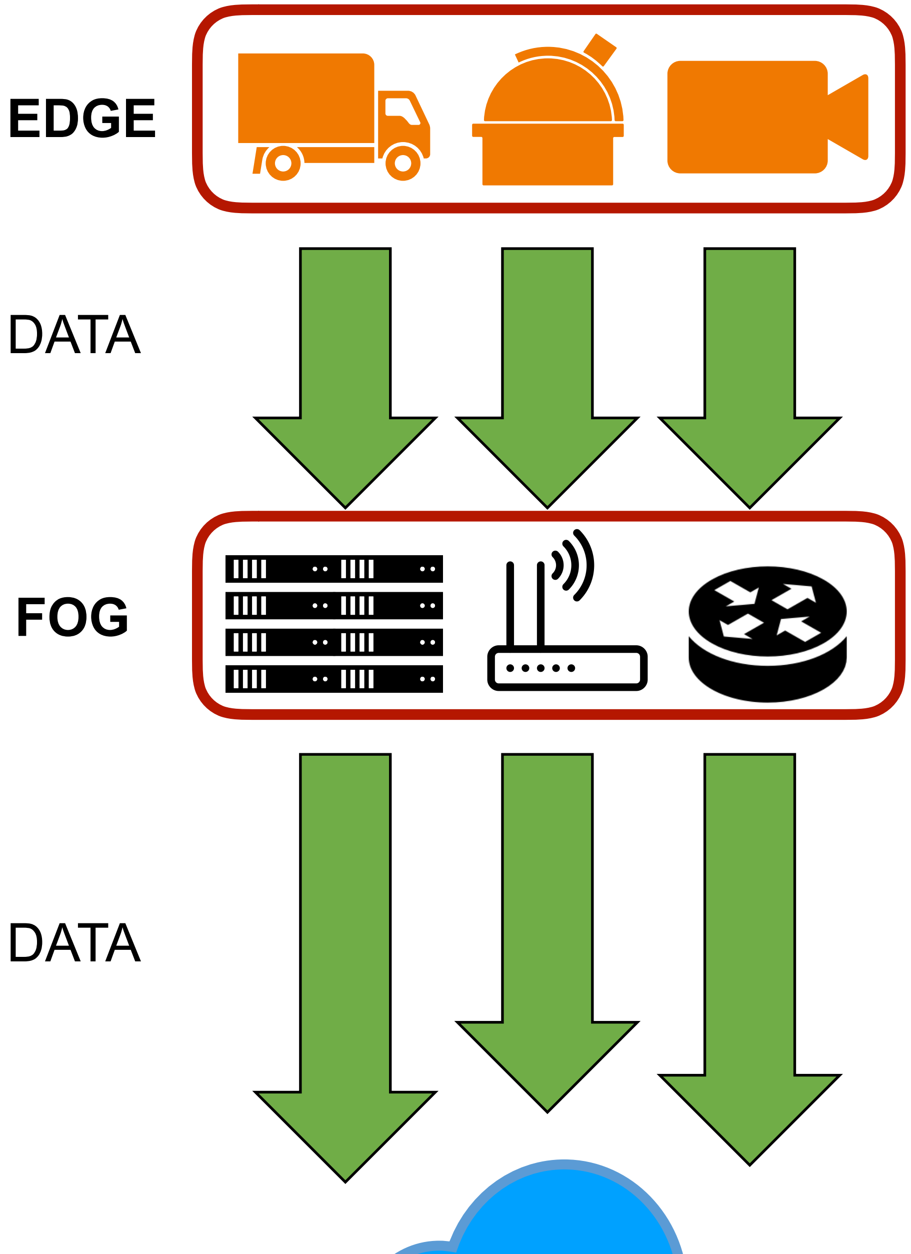
Simplified device programming with AWS Lambda

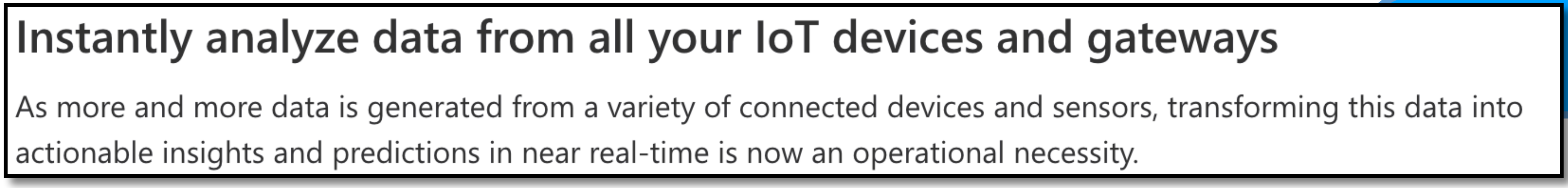
You can develop code in the cloud and then deploy it seamlessly to your devices with AWS Lambda. AWS IoT



Capture data in real time

Process IoT data instantly to help identify valuable insights related to device behavior and operations in the field. Spot

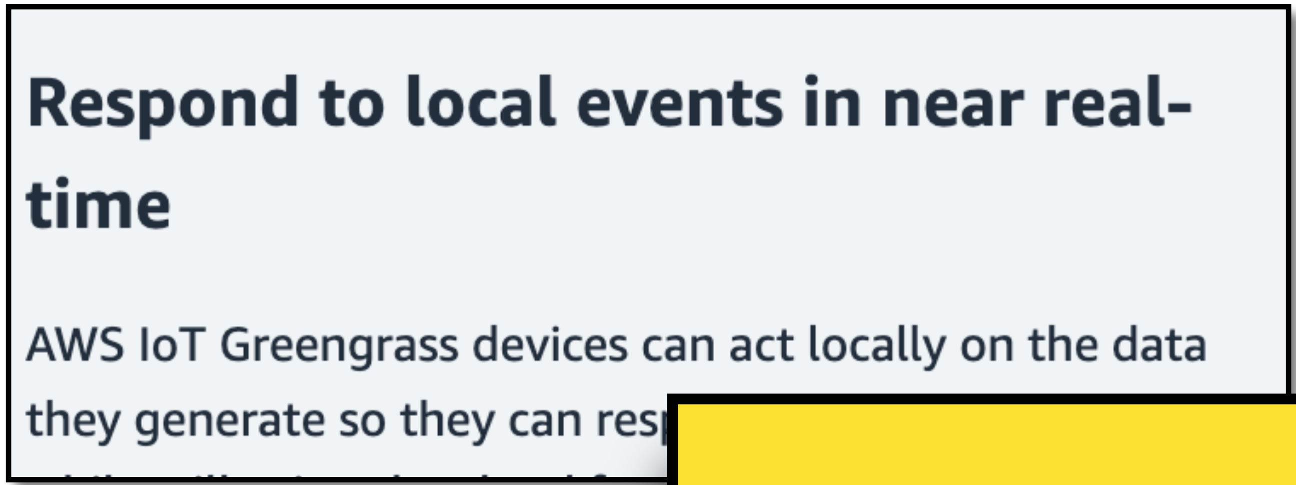




Instantly analyze data from all your IoT devices and gateways

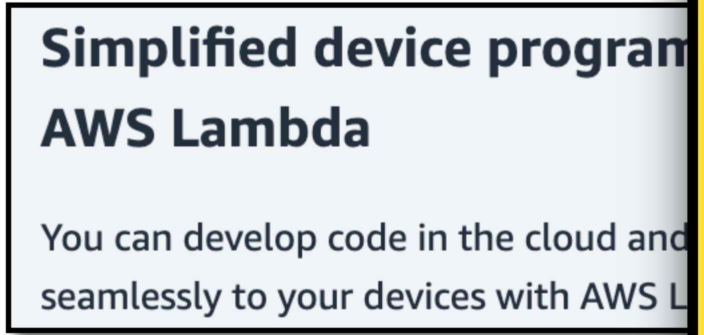
As more and more data is generated from a variety of connected devices and sensors, transforming this data into actionable insights and predictions in near real-time is now an operational necessity.

Edge processing tools



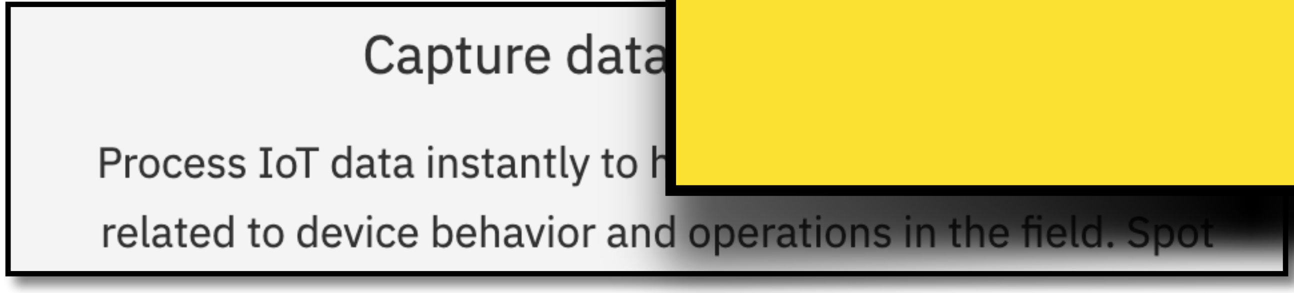
Respond to local events in near real-time

AWS IoT Greengrass devices can act locally on the data they generate so they can resp



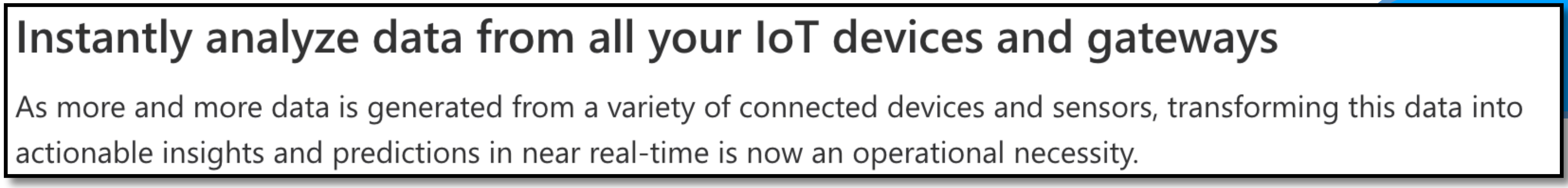
Simplified device program AWS Lambda

You can develop code in the cloud and seamlessly to your devices with AWS L



Capture data

Process IoT data instantly to h related to device behavior and operations in the field. Spot



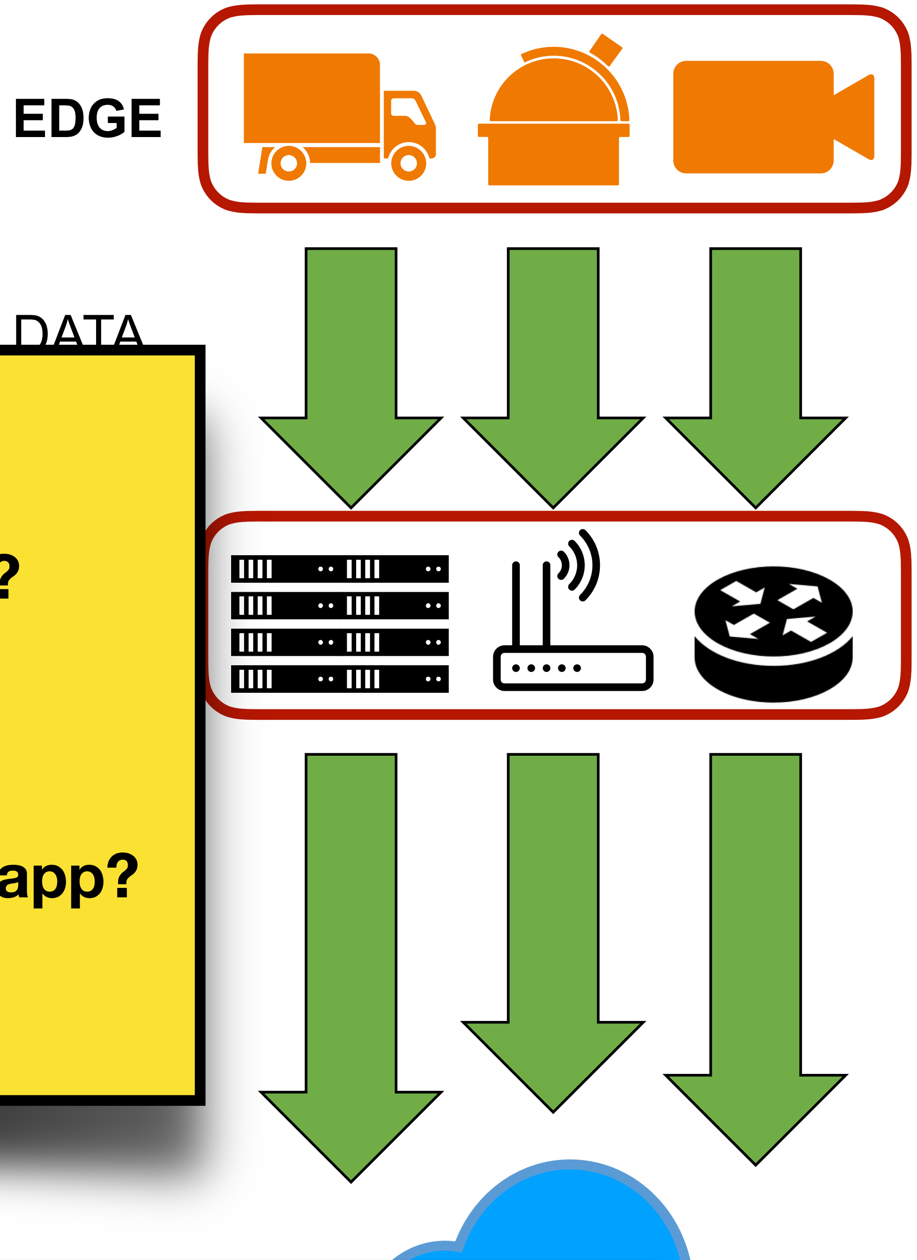
Instantly analyze data from all your IoT devices and gateways

As more and more data is generated from a variety of connected devices and sensors, transforming this data into actionable insights and predictions in near real-time is now an operational necessity.

What's their performances?

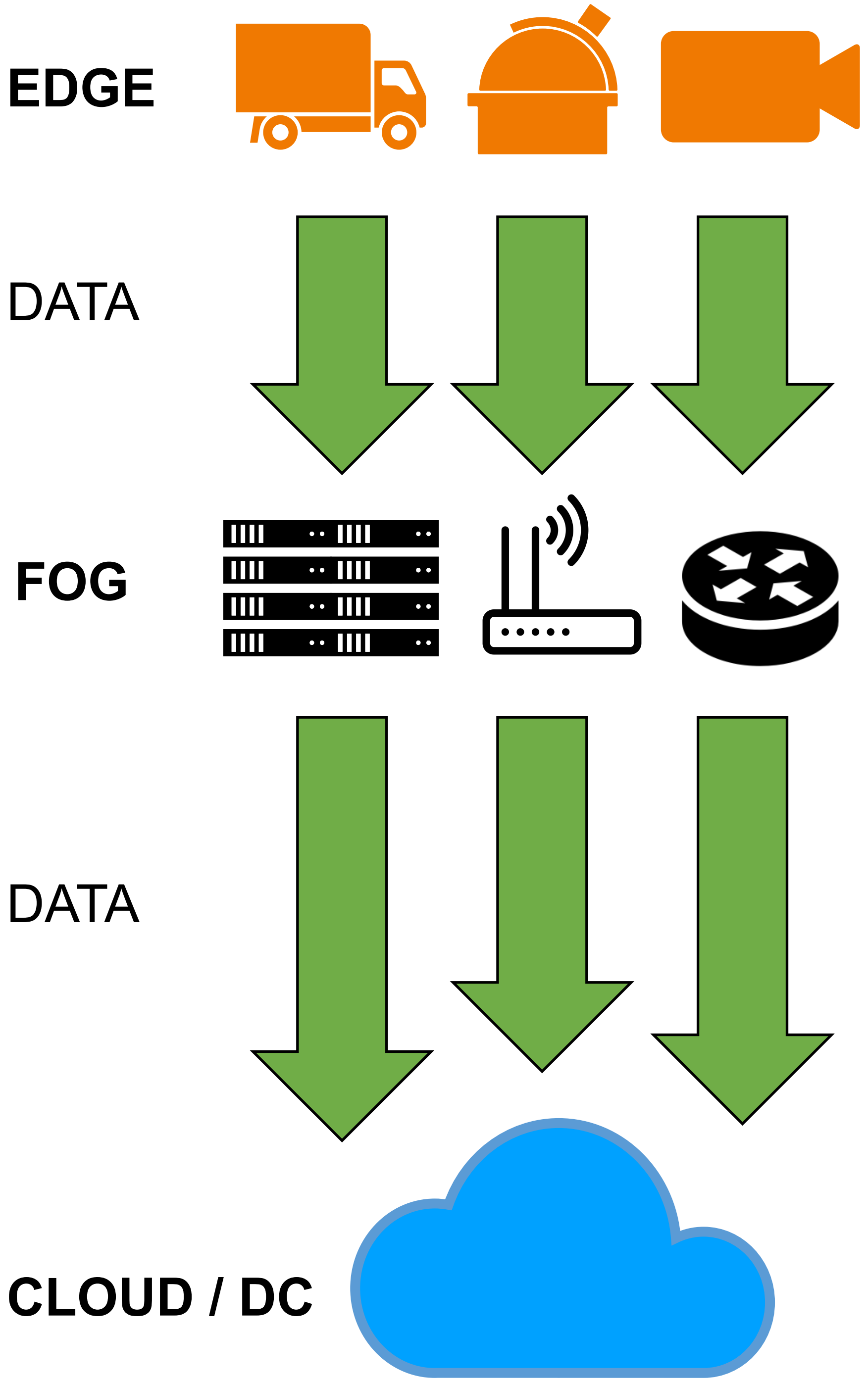
Under which conditions?

Do they integrate well with my app?



Benchmarking Edge tools

- Understanding a tool's performance through benchmarking

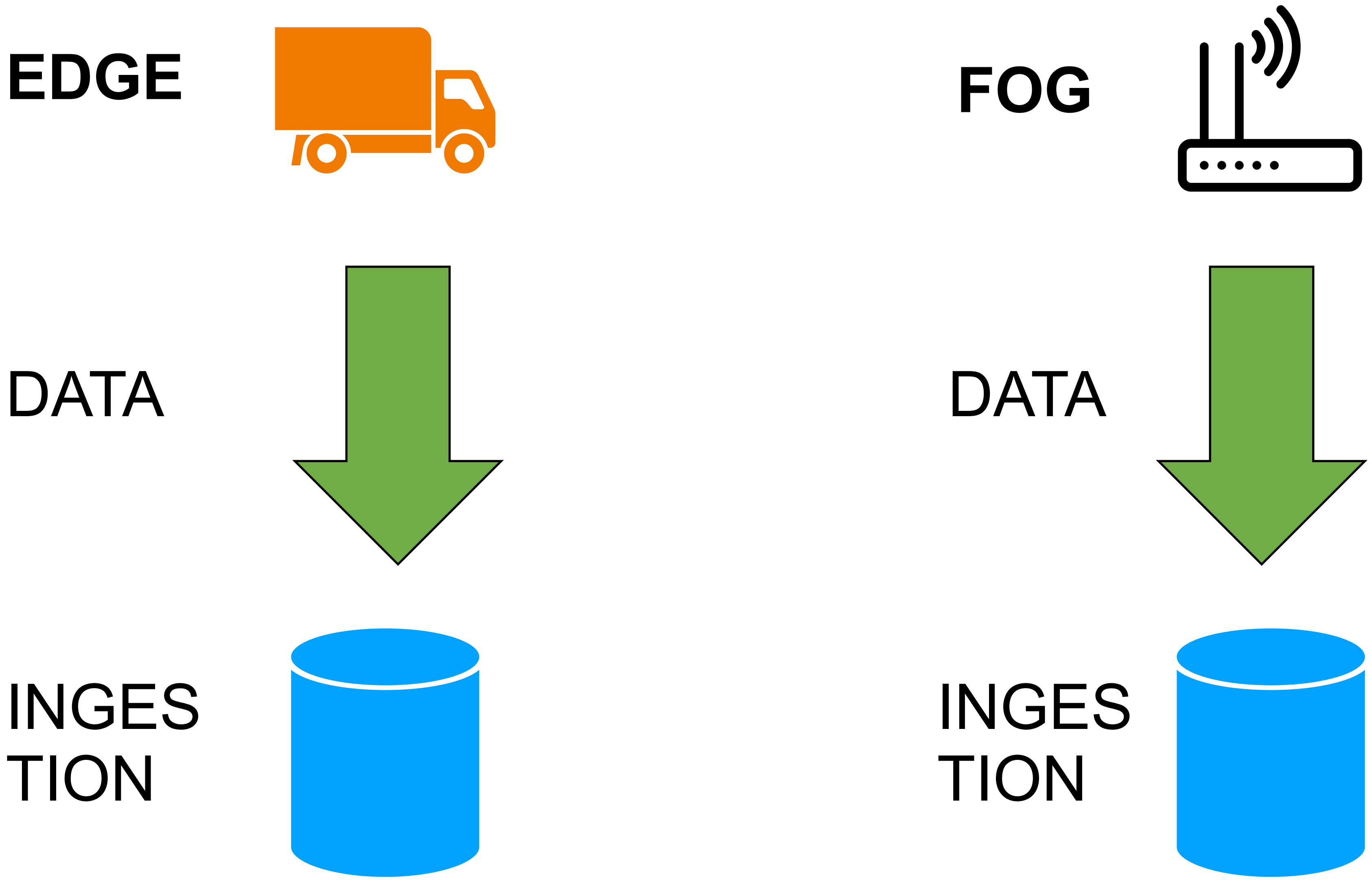


Related work

- **TPCx-IoT:**
  - Created for hardware benchmarking
  - Fog oriented

- **Academic benchmarks:**
  - Irreproducible
  - Just a few commercial tools
  - Lack a clear methodology (metrics, workloads, parameters)
  - Not focused on the tools

Benchmarking Edge tools



EDGE

DATA

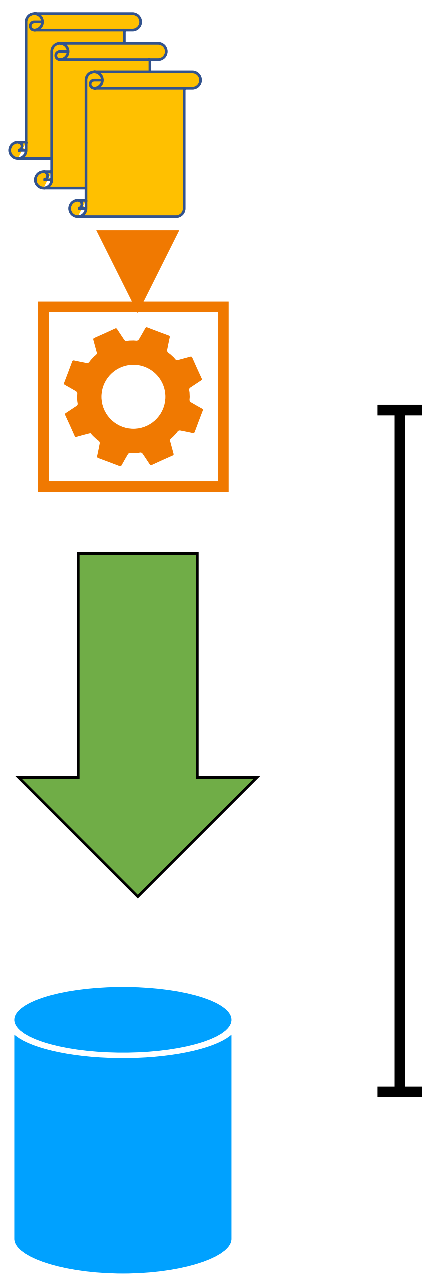
INGES
TION

FOG

DATA

INGES
TION

General view



Workload

Deployed Tools

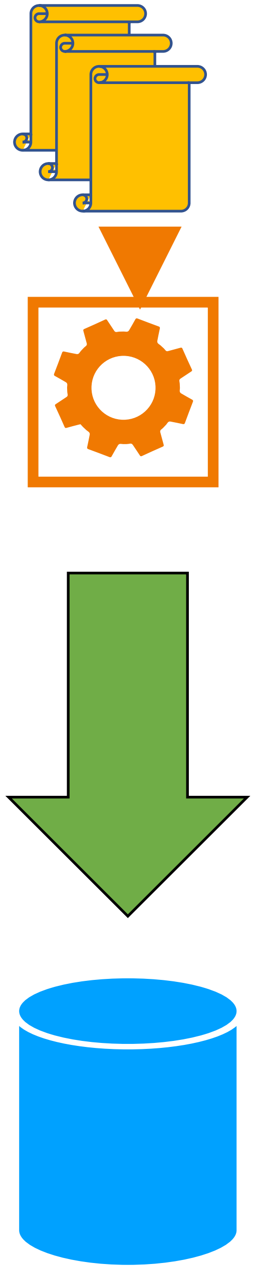
Data

Ingestion system

- Latency
- Throughput
- Resource usage

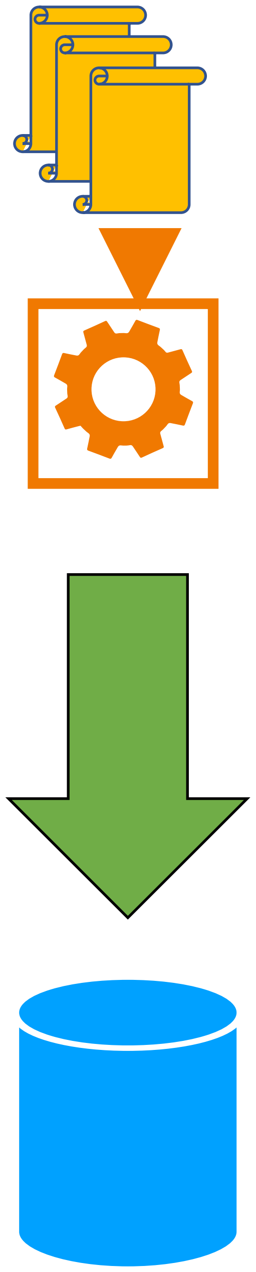
Benchmark objectives

- Processing performance
- Supported programming languages
- Connectivity
- Development easiness



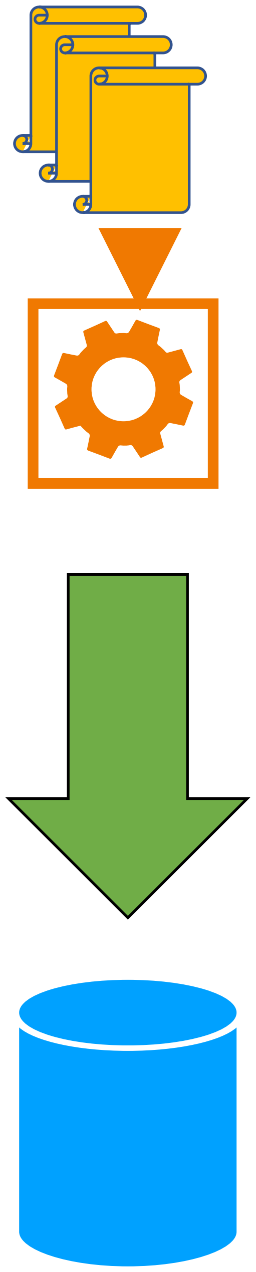
Benchmark parameters

- Edge processing frameworks
- Edge infrastructure
- Scenarios / Workload
- Input data throughput



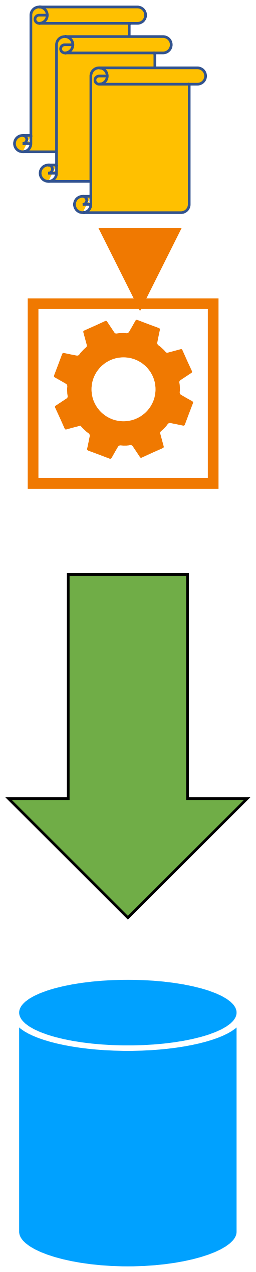
Edge processing frameworks

- Apache Edgent
- Amazon Greengrass
- Azure Stream Analytics
- IBM Watson IoT
- Intel IoT
- Oracle Edge Analytics
- Baselines (C++, Java)



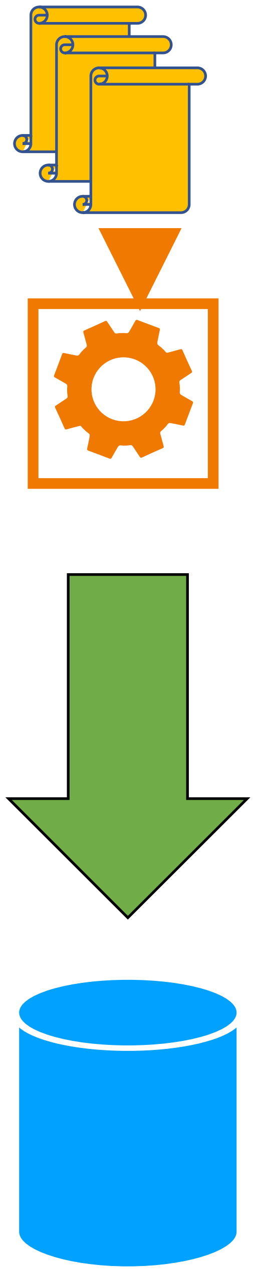
Infrastructure

- **Virtual machines and bare metal**
  - **nano** (1 core, 256MB)
  - **mini** (1 core, 1GB)
  - **Raspberry PI2** (4 cores, 1GB)
  - **medium** (4 cores, 4GB)
  - **large** (8 cores, 8GB)
  - Dell **PowerEdge** R630 (16 cores, 128GB)



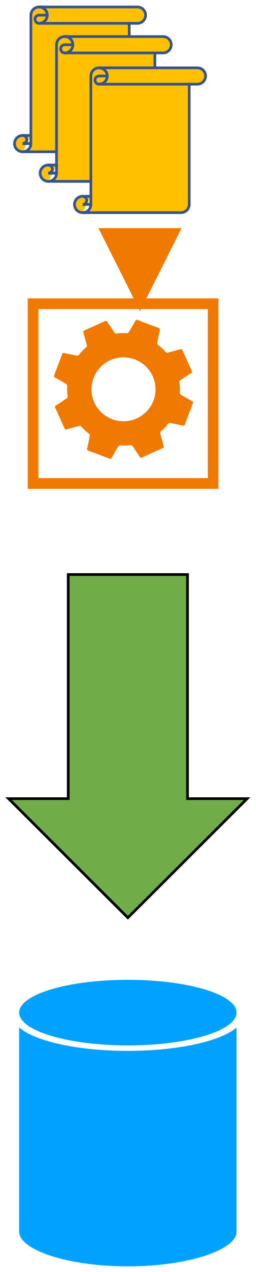
Scenarios / Workload

- **New York City Taxi and Limousine Commission**
  - Busiest driver in the last hour minutes every 5 minutes
- **CCTV footage from Univ. of California San Diego**
  - Busiest places in the last hour every 5 minutes



Evaluation metrics

- Message processing throughput
- Processing latency
- Number of supported programming languages
- Framework connections
- Lines of code



Inflection: earthquake early warning

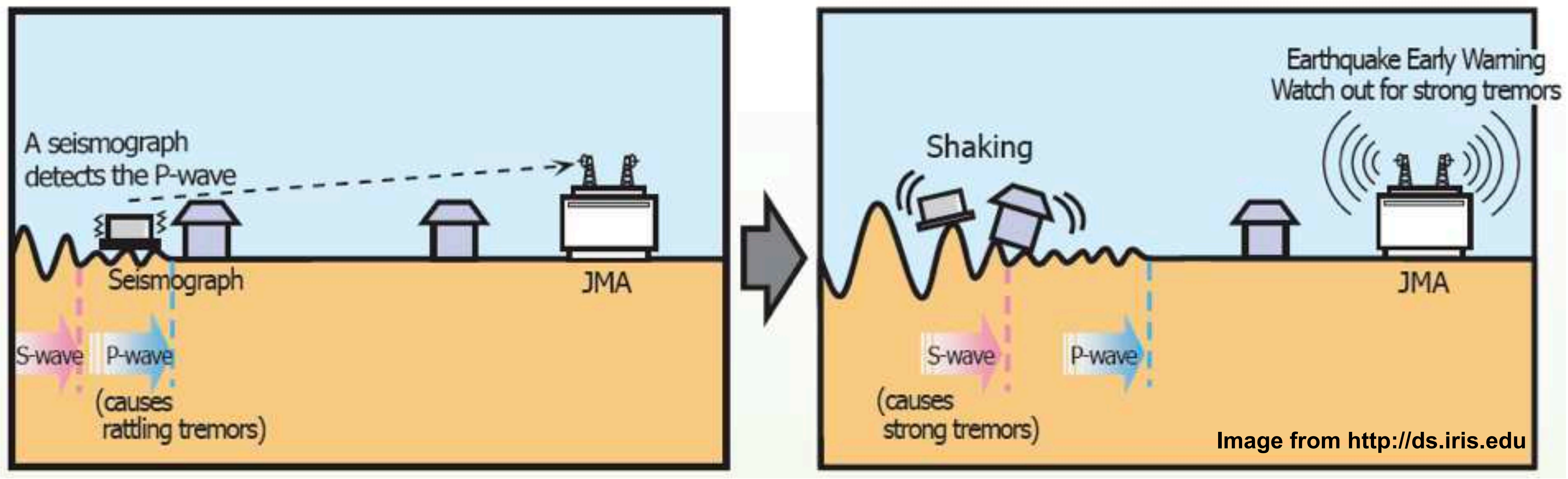
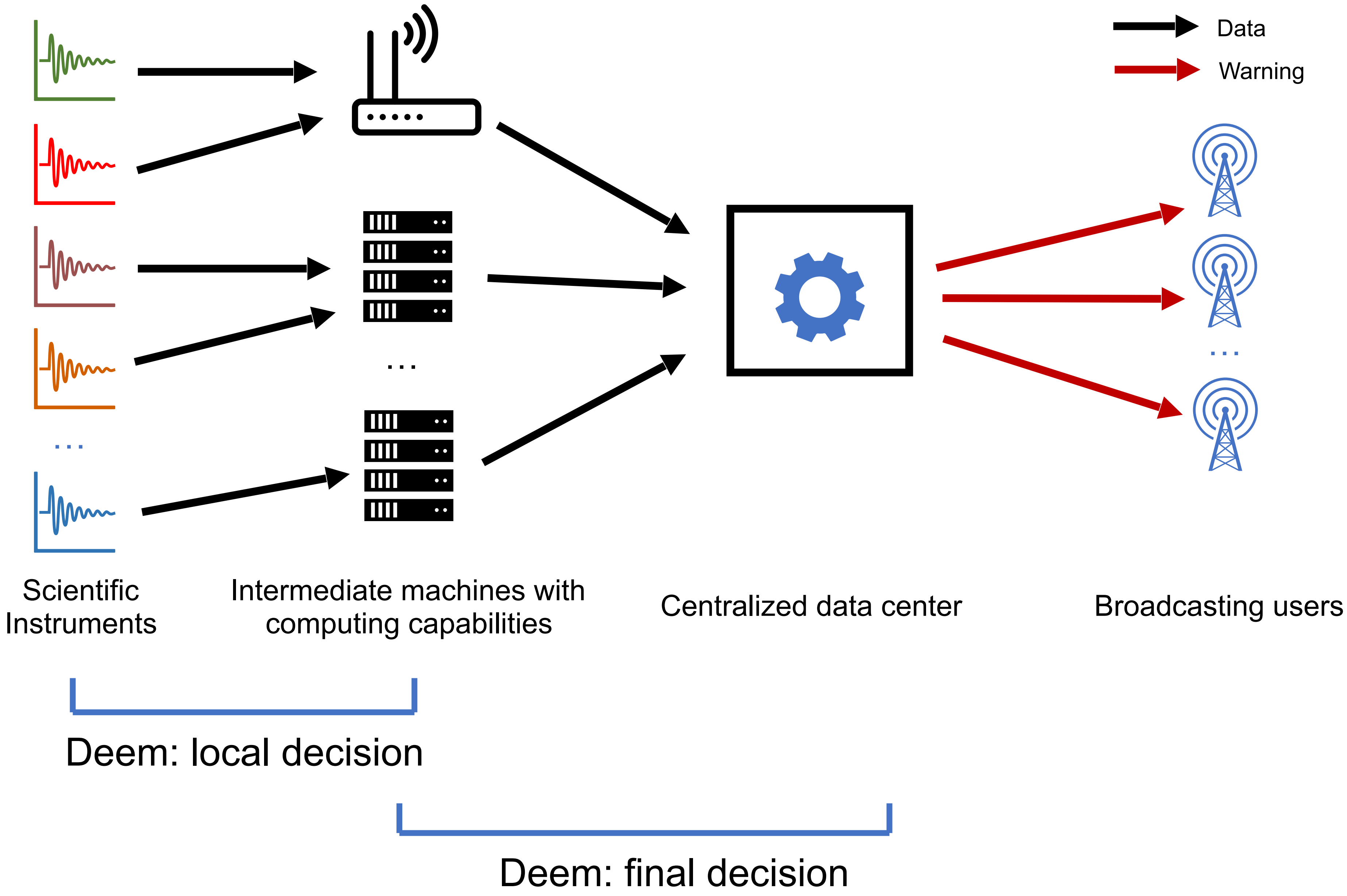


Image from http://ds.iris.edu

- ❑ **Objective:** process P-waves (time series) in order to characterize earthquakes before they start.
  - ❑ **DEEM**: real time distributed hierarchical ML algorithm for earthquake magnitude measurement.

❑ Kevin Fauvel, Daniel Balouek-Thomert, Diego Melgar, Pedro Silva, Anthony Simonet, Gabriel Antoniu, Alexandru Costan, Manish Parashar, and Ivan Rodero. Towards a decentralized multi-sensor machine learning approach for Earthquake Early Warning. Soumission à ECML PKDD 2019

Inflection: earthquake early warning



Data

Warning

Scientific Instruments

Intermediate machines with computing capabilities

Centralized data center

Broadcasting users

Deem: local decision

Deem: final decision

❑ **Deem:** distributed hierarchical ML algorithm

❑ Allows for heterogeneous sensors

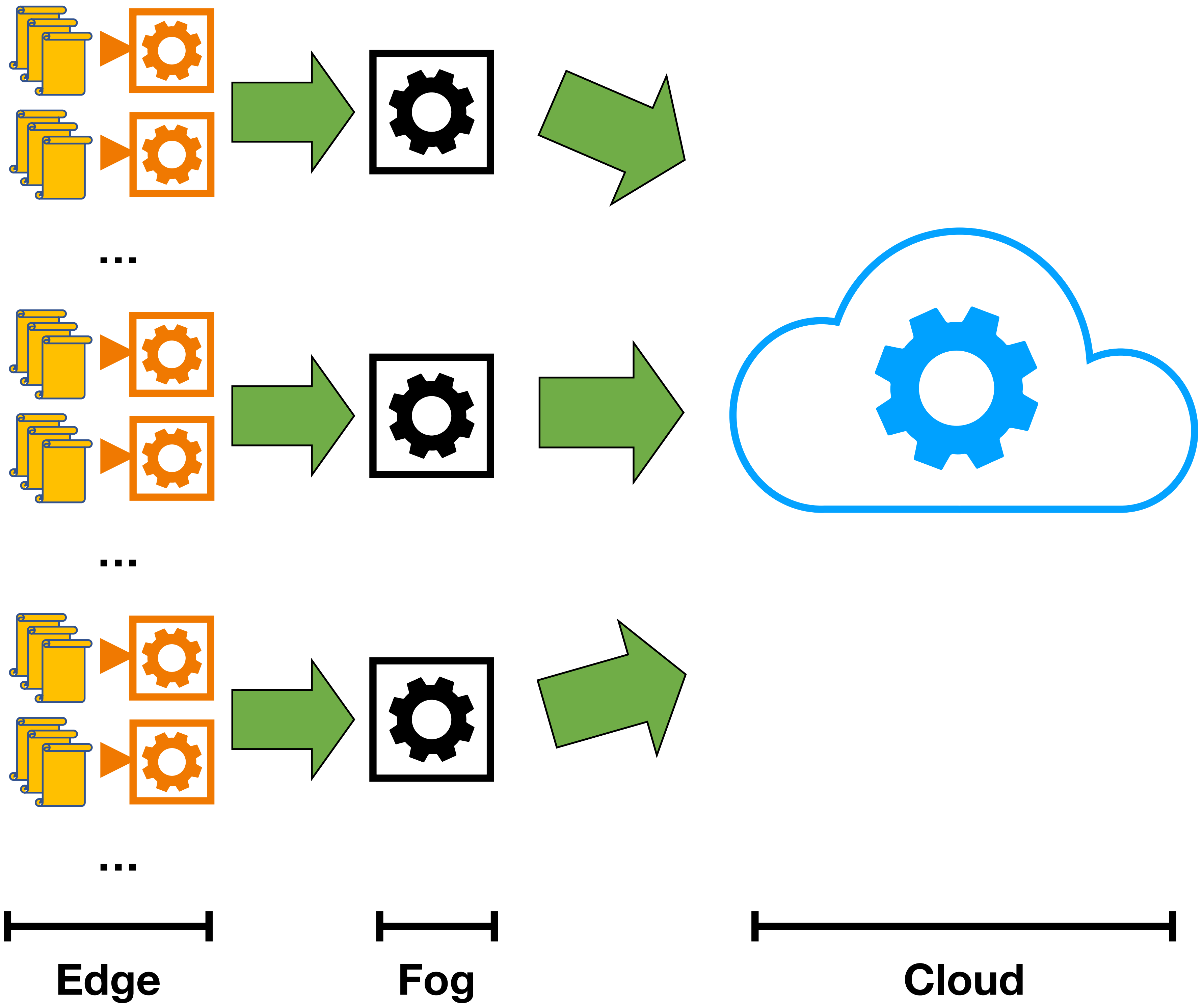
❑ Can be used on low quality networks

❑ Allow for local decision making

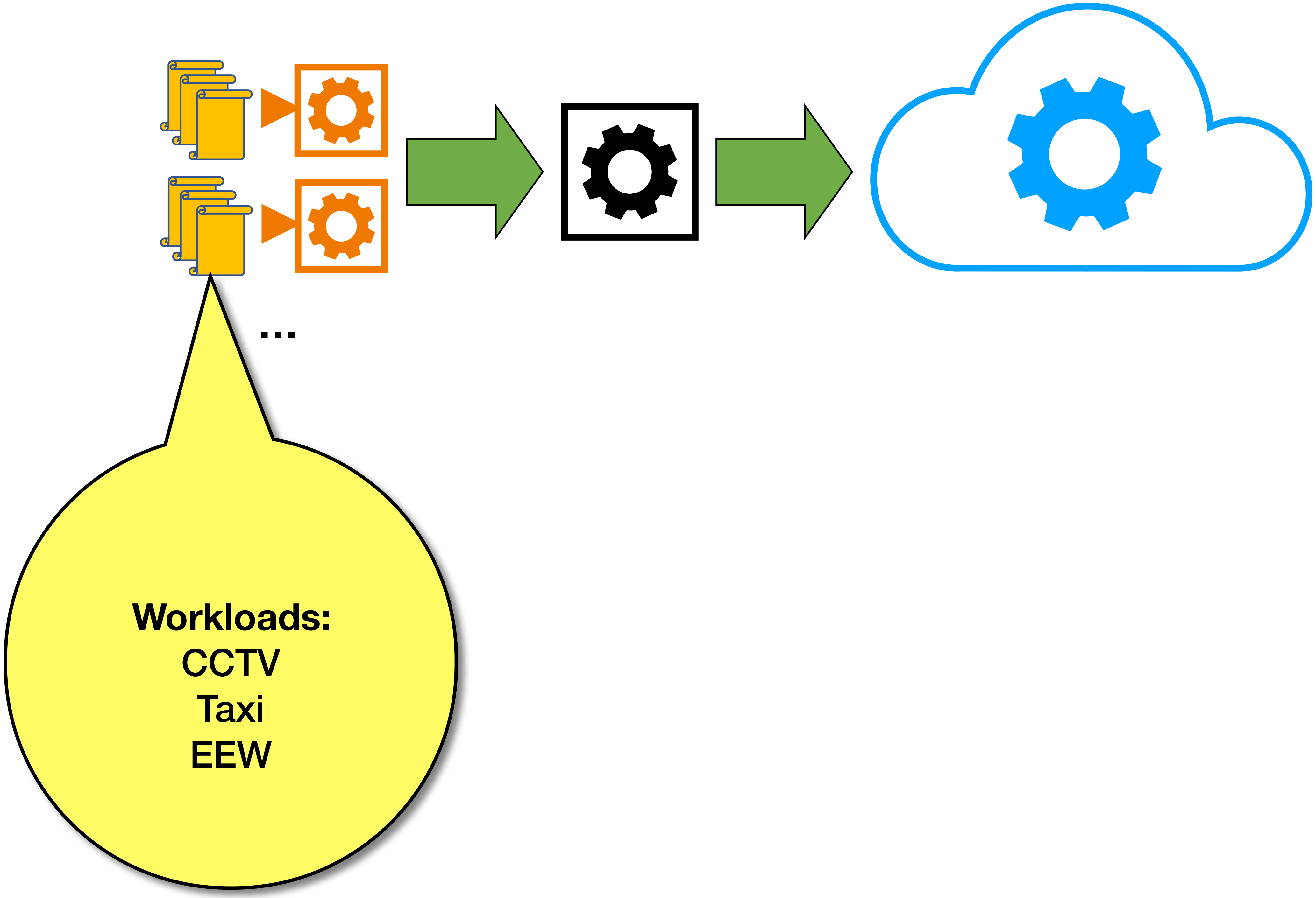
New requirements

- Benchmark a complete scenario
- Control network characteristics
- Control frameworks' configuration parameters
- Control Edge, Fog and Cloud infrastructures

Updated workflow

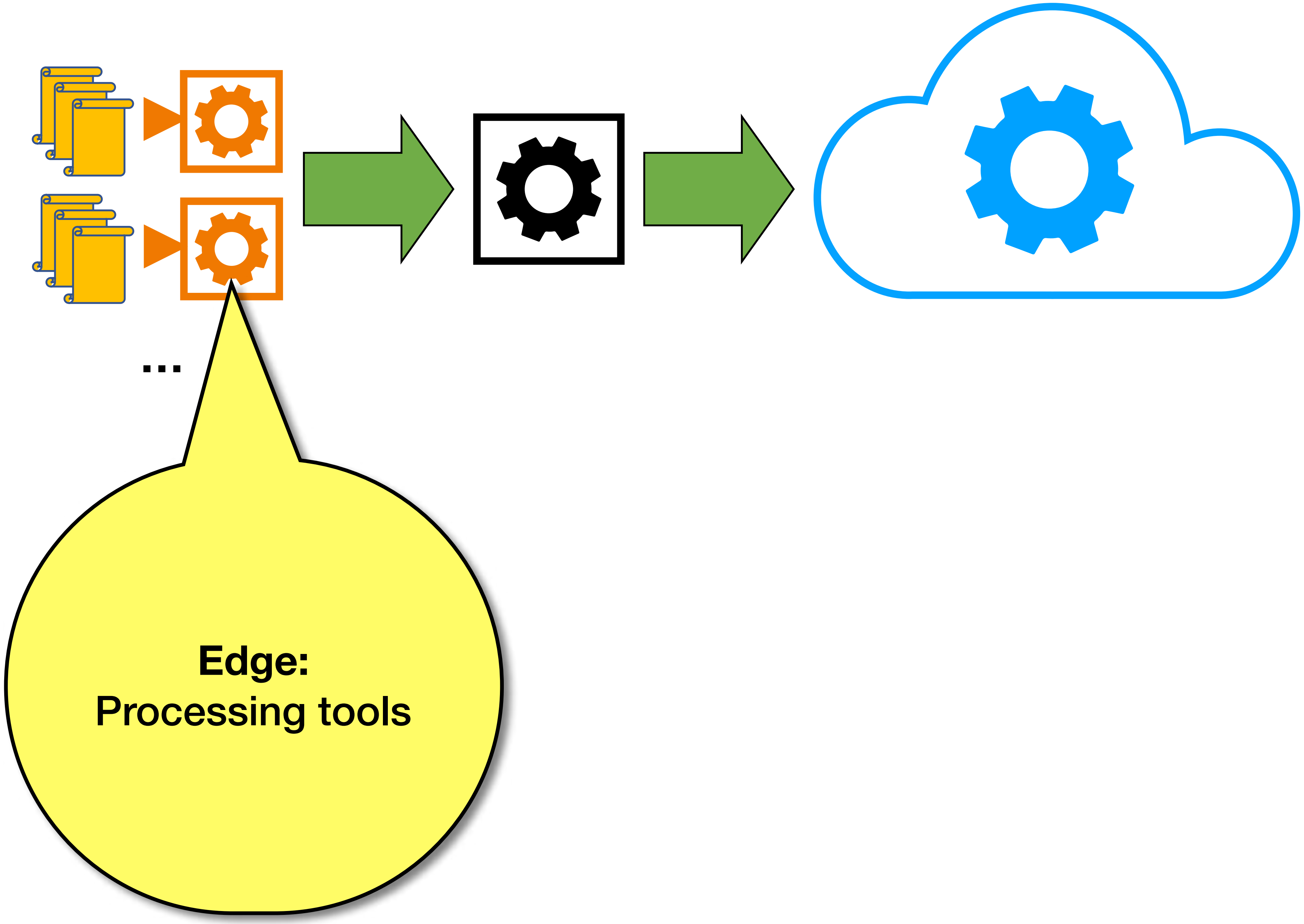


Updated workflow



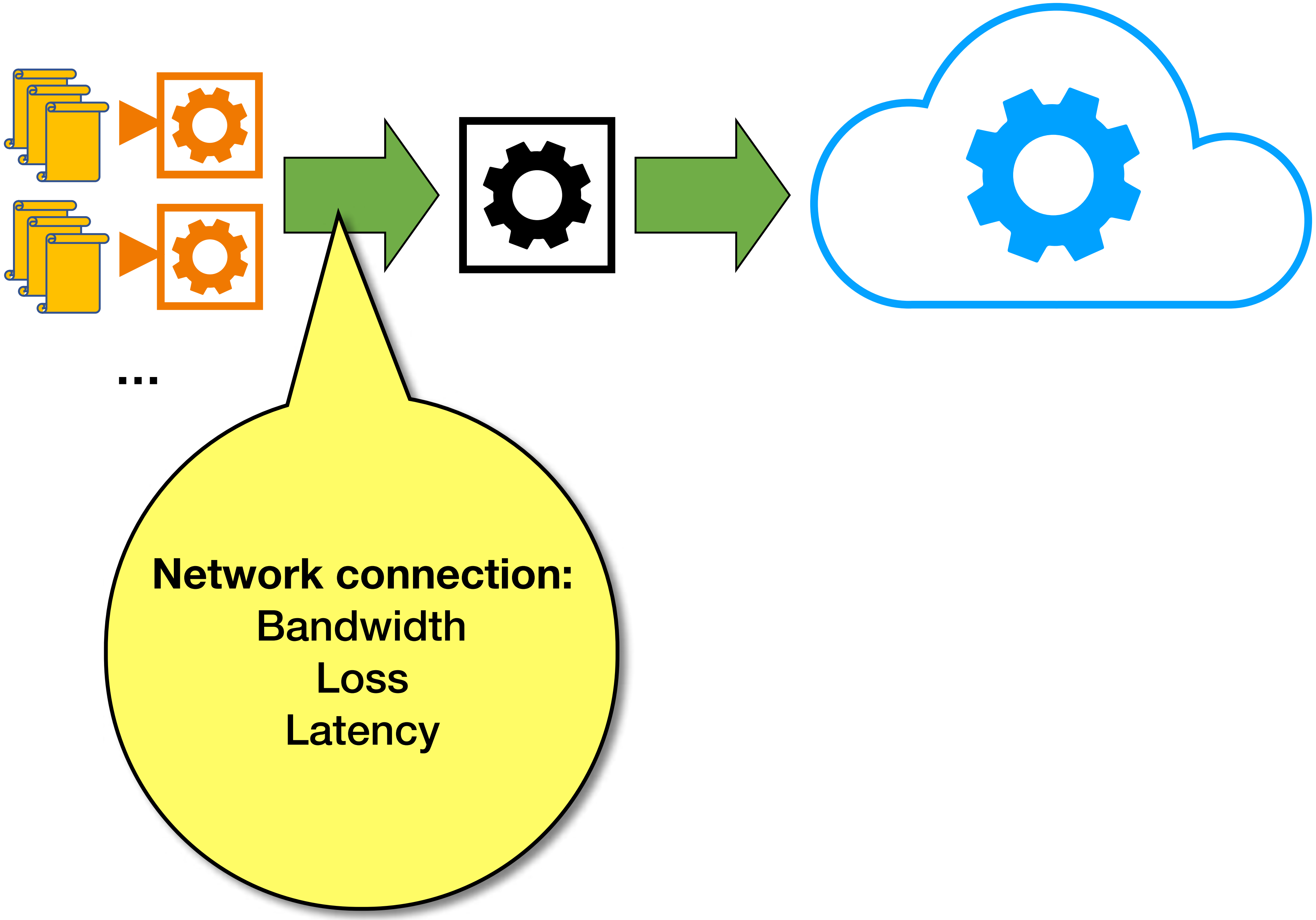
Workloads:
CCTV
Taxi
EEW

Updated workflow



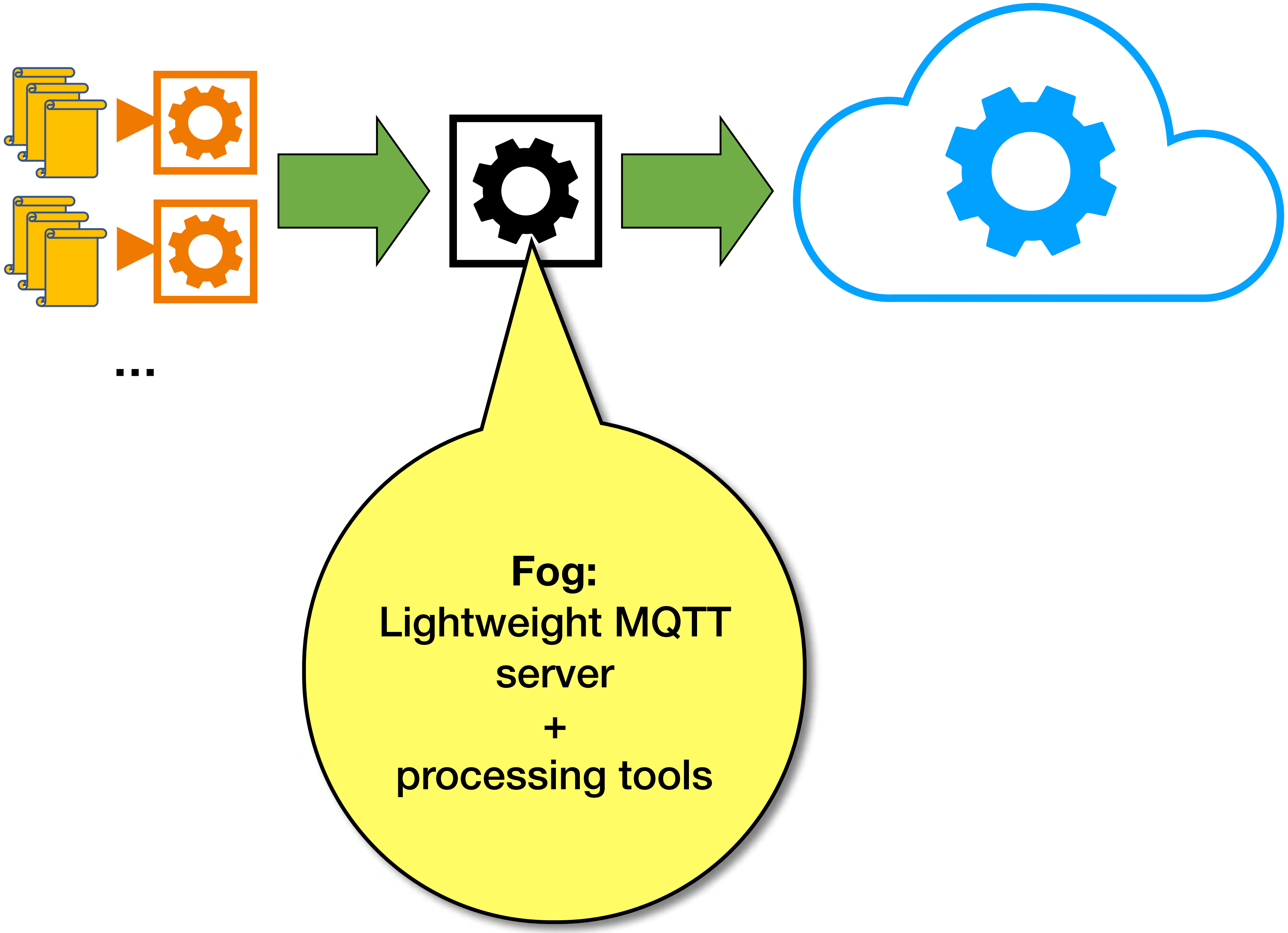
Edge:
Processing tools

Updated workflow



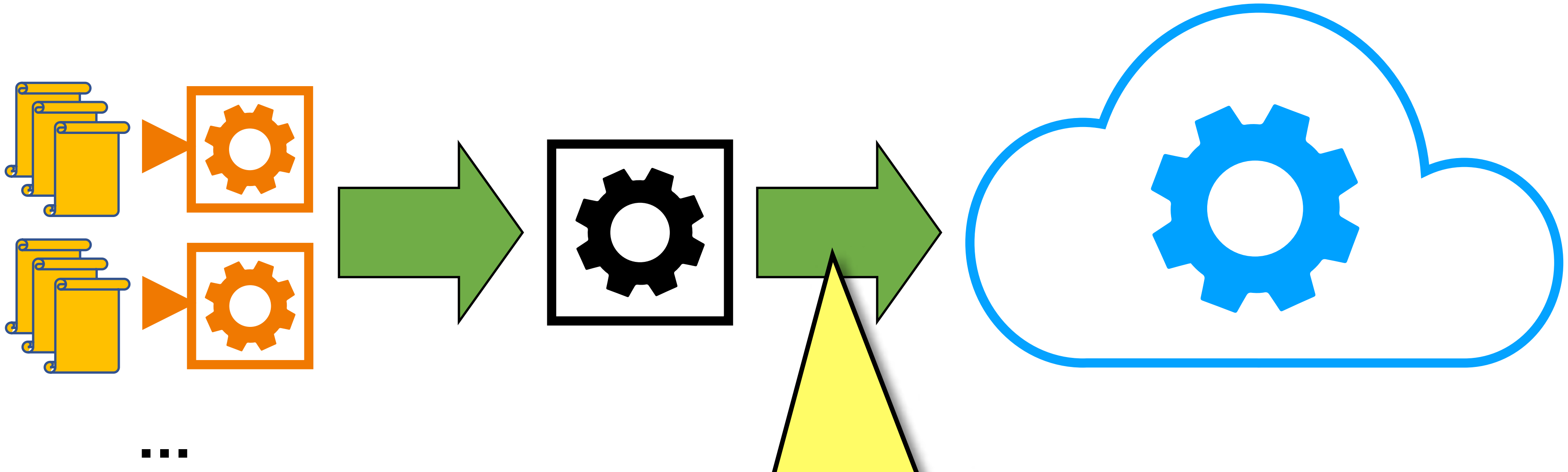
Network connection:
Bandwidth
Loss
Latency

Updated workflow



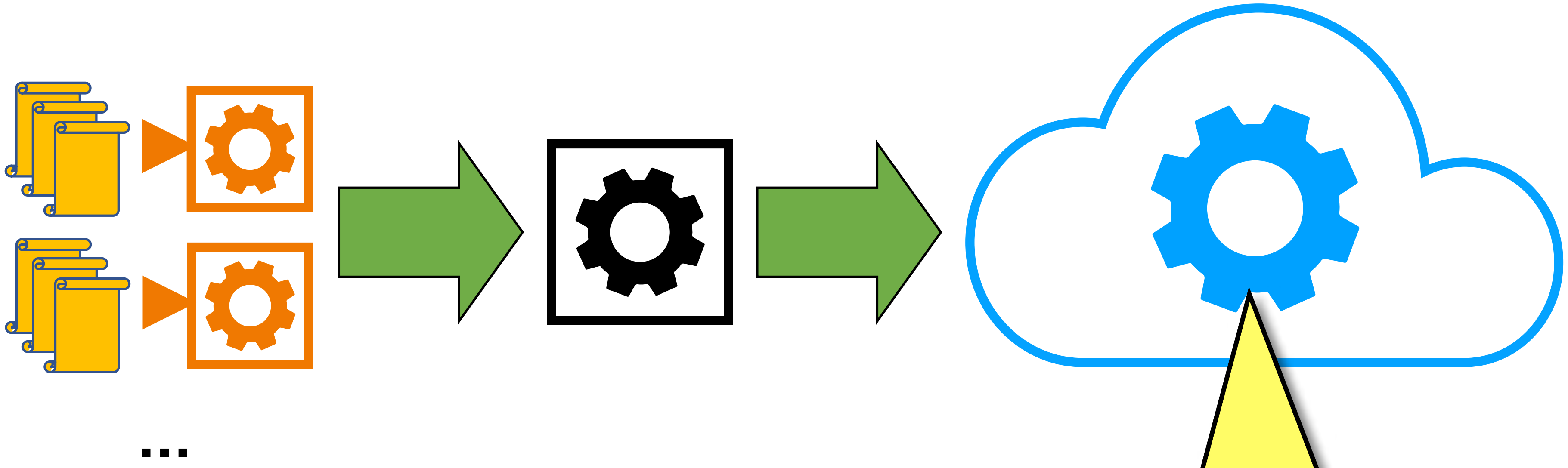
Fog:
Lightweight MQTT server
+
processing tools

Updated workflow



Network connection:
Bandwidth
Loss
Latency

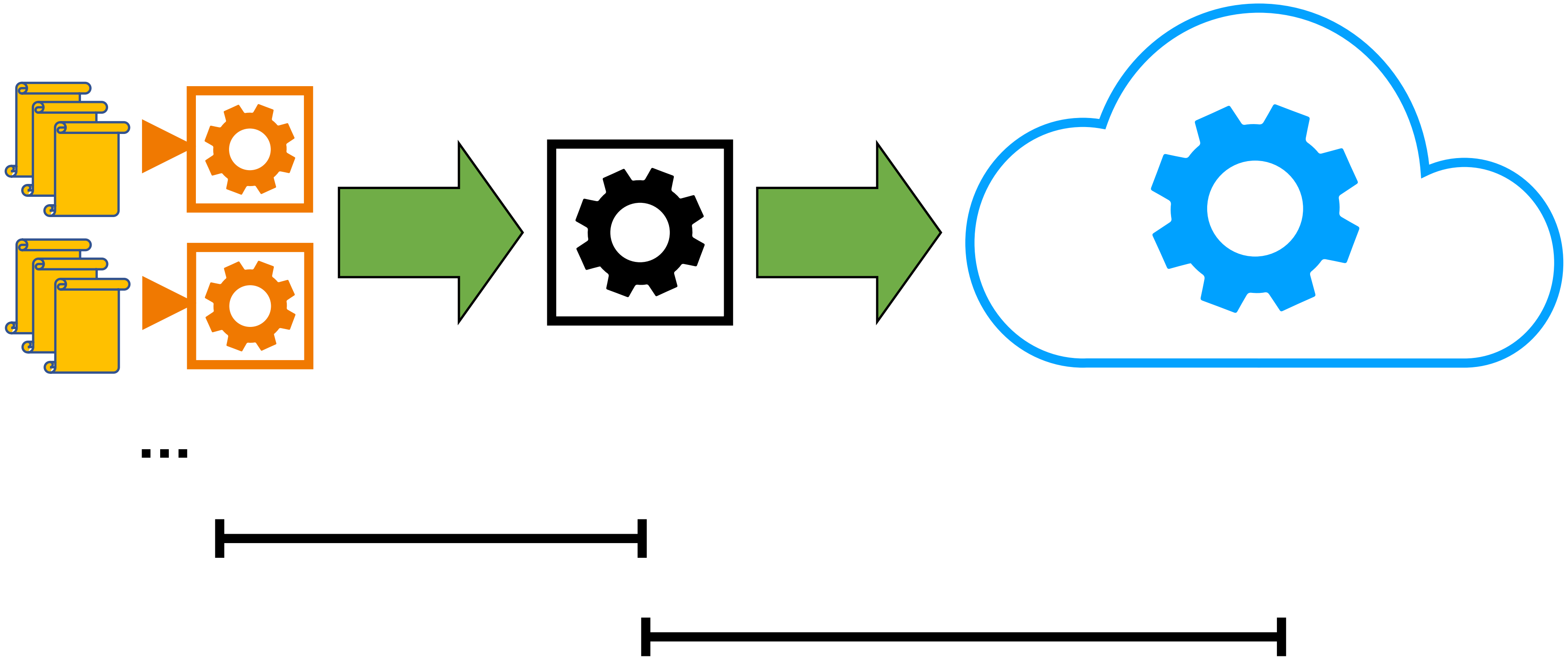
Updated workflow



Stream processing:
Kafka brokers
Zookeeper server
Flink Cluster

- **There is a selection of Kafka, Zookeeper and Flink parameters that can be set**

Updated workflow

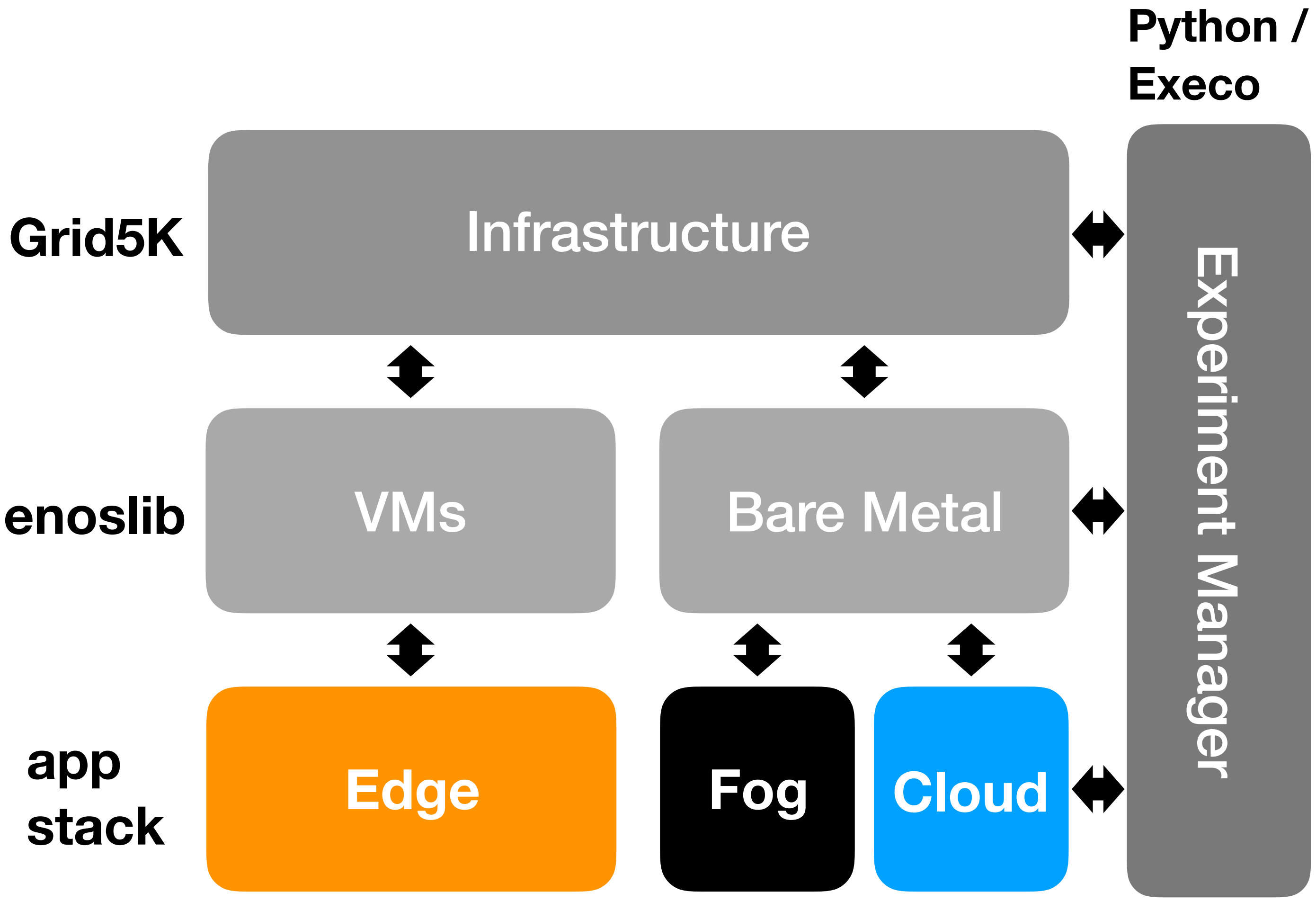


- Latency
- Throughput
- Resource usage

Glimpse on the implementation

- **Experiment manager:**
  - Configures the infrastructure
  - Deploys frameworks/tools
  - Deploys applications and manages their executions
  - Monitors resource usage
  - Gathers metrics and logs

- **Edge+Fog+Cloud processing management:**
  - Wrappers / interfaces (metric generation, configuration, connection)



Future work

- Finish the benchmark prototype
- Finish paper with EEW use case
- Integrate a DL based use case