



APACHECON Europe, Oct 23<sup>rd</sup>, 2019


Running visual quality inspection at the edge with Apache NiFi & MiNiFi

Pierre Villard - @pvillard31





# Pierre Villard // @pvillard31

Customer Engineer @  Google Cloud

Committer and PMC member for Apache NiFi (in the community since 2015)

Twitter/Github - @pvillard31

Blog - <https://www.pierrevillard.com/>



# NiFi - a software developed 13y ago by the NSA



**2006**

NiagaraFiles (NiFi) was first incepted at the National Security Agency (NSA)



**November 2014**

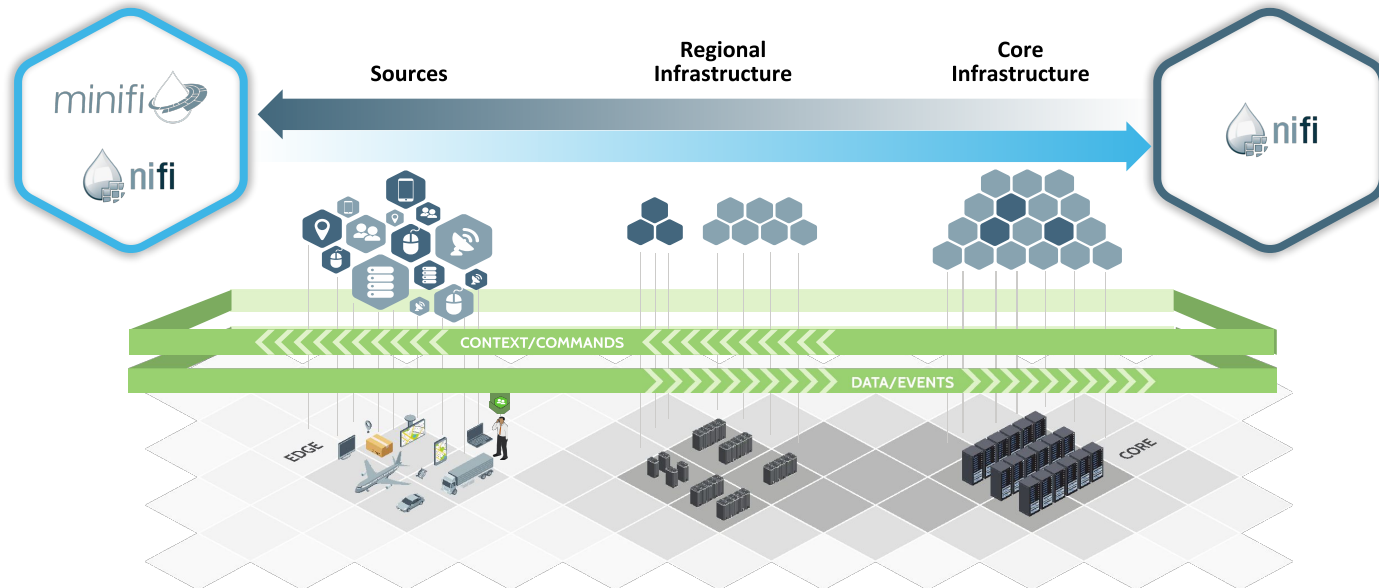
NiFi is donated to the Apache Software Foundation (ASF) through NSA's Technology Transfer Program and enters ASF's incubator.



**July 2015**

NiFi reaches ASF top-level project status

# What is NiFi used for?

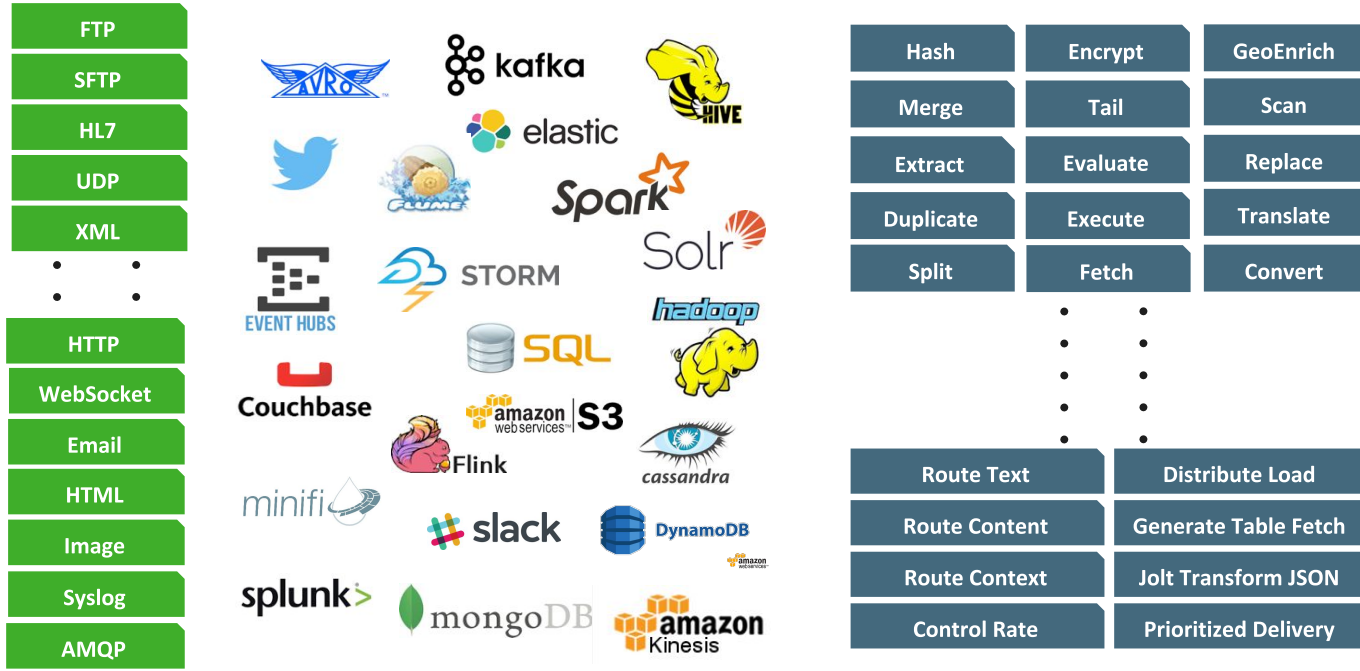




# The NiFi ecosystem

- ◆ **NiFi** - Powerful and scalable directed graphs of data routing, transformation, and system mediation logic.
- ◆ **MiNiFi (Java version)** - Complementary data collection approach that supplements the core tenets of NiFi in dataflow management, focusing on the collection of data at the source of its creation.
- ◆ **MiNiFi (C++ version)** - The C++ implementation is an additional implementation to the one in Java with the aim of an even smaller resource footprint. Perspectives of the role of MiNiFi should be from the perspective of the agent acting immediately at, or directly adjacent to, source sensors, systems, or servers.
- ◆ **NiFi Registry** - Complementary application that provides a central location for storage and management of shared resources across one or more instances of NiFi and/or MiNiFi.
- ◆ **NiFi C2 Server** - Command and control server to manage many disparate agents running on all sorts of devices, to coordinate their work and to push out revised flows/configurations.
- ◆ **NiFi Fluid Design System** - Atomic reusable platform providing consistent set of UI/UX components.

# 300+ processors for deeper ecosystem integration





# The Apache way: community over code

Subscribe to the mailing lists:

[https://nifi.apache.org/mailling\\_lists.html](https://nifi.apache.org/mailling_lists.html)  
users@nifi.apache.org & dev@nifi.apache.org

Open and comment JIRAs:

<https://issues.apache.org/jira/projects/NIFI>

Contribute code:

<https://nifi.apache.org/developer-guide.html>  
<https://cwiki.apache.org/confluence/display/NIFI/Contributor+Guide>  
<https://issues.apache.org/jira/projects/NIFI/issues/>

Get involved in the code review process:

<https://github.com/apache/nifi>  
<https://github.com/apache/nifi-registry>  
<https://github.com/apache/nifi-minifi>  
<https://github.com/apache/nifi-minifi-cpp>  
<https://github.com/apache/nifi-fds>



# The Apache NiFi community in few numbers

**535+ members** on the Slack channel

**260+ contributors** on Github across the repositories

**45 committers** in the Apache NiFi community

**Apache NiFi 1.10.0** to be released soon (RC vote in progress!)

**1M+ docker pulls** of the Apache NiFi image





# **Visual quality inspection: Detect broken cookies**

# The ML spectrum in GCP



The TensorFlow logo is at the top left. Below it is a complex diagram showing a neural network architecture with various layers and nodes, representing the use of an open-source SDK.

The CloudML logo is at the top left. Below it is a photograph of a Google Cloud TPU (Tensor Processing Unit) hardware card. To the right of the hardware, the text reads: "Scale, No-ops Infrastructure" and "TPU - 7 years ahead of GPU in terms of price/performance".

### Google Trained Models

Four icons representing Google Trained Models: Cloud Translate (with Chinese characters), Cloud Vision, Cloud Natural Language, and Cloud Speech.  
Two images: a beehive with bees and a collection of donuts, representing the types of models available.

Use/extend OSS SDK

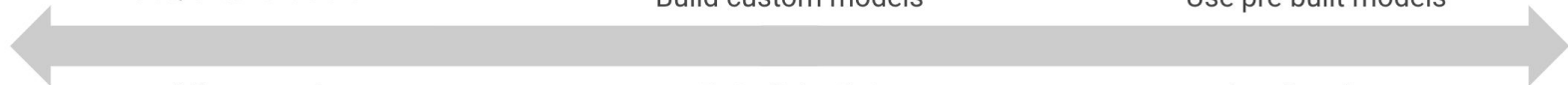
Build custom models

Use pre-built models

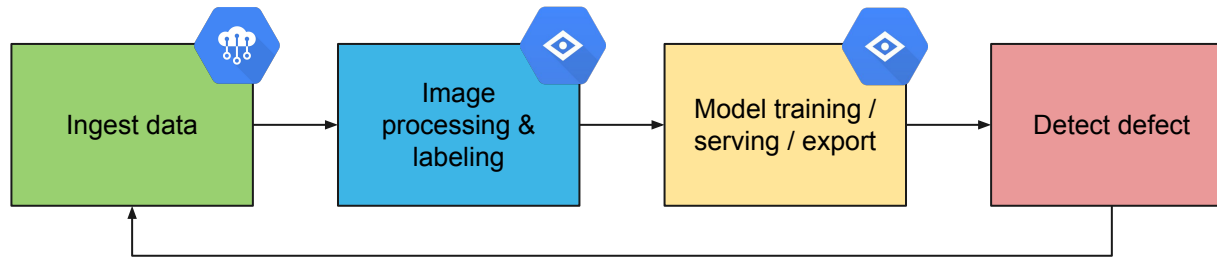
ML researcher

Data Scientist

App Developer

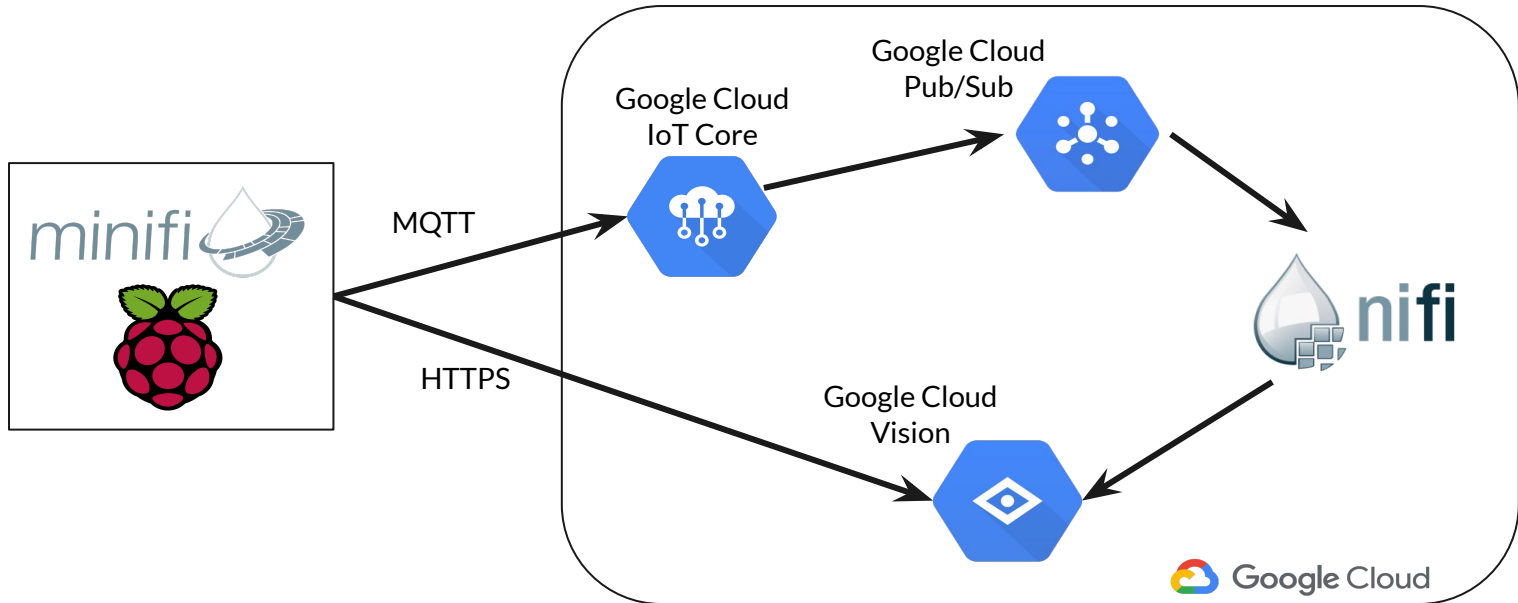


# Continuous model retrain

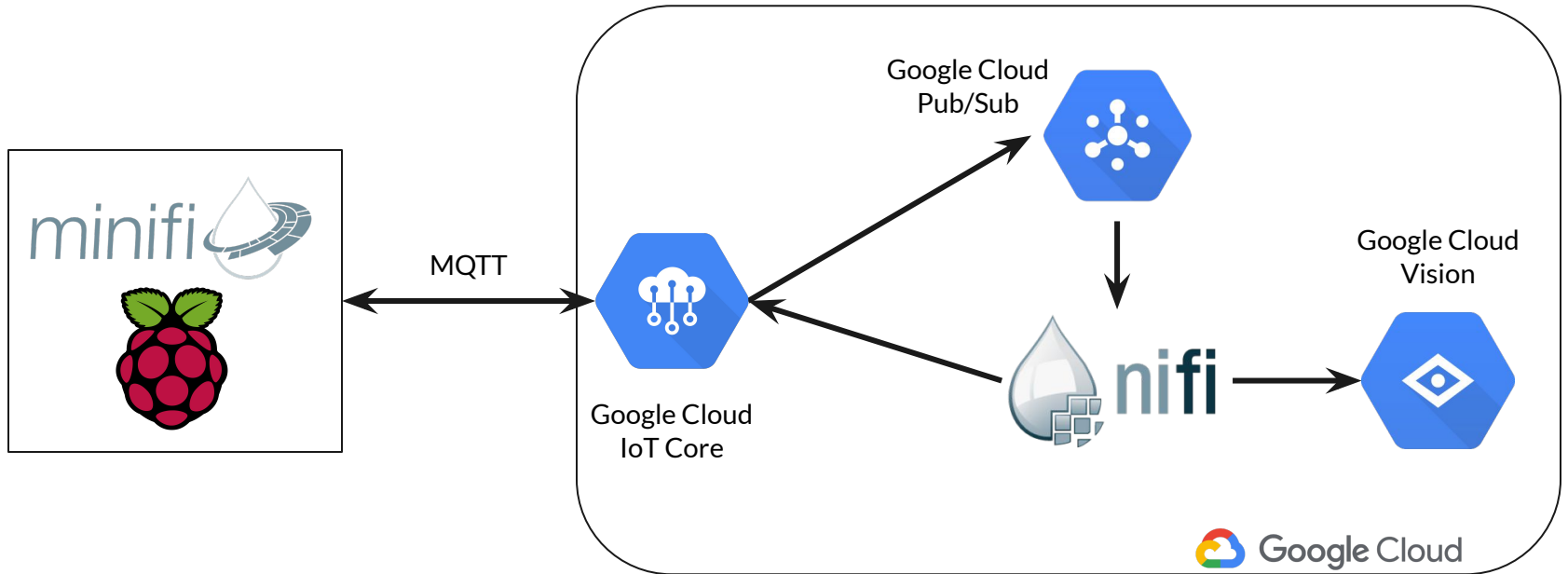


- Automatically train customized ML models in the cloud
- Efficiently acquire images, label images, deploy model and run inference
- Continuously refresh models using fresh data from the production lines

## Architecture #1 - training & inference in the cloud



## Architecture #2 - training in the cloud & inference at the edge





**Collect & label data to initialize a dataset**

# Register my device in Google Cloud IoT Core



Google Cloud Platform aceu19

IoT Core

Devices [+ CREATE A DEVICE](#) [DELETE](#)

Registry ID: aceu19registry  
europe-west1

Devices are things that connect to the Internet directly or through a gateway. [Learn more](#)


Enter exact device ID

<input type="checkbox"/>	Device ID	Communication	Last seen	Stackdriver Logging
<input type="checkbox"/>	my-device	✔ Allowed	19 Oct 2019, 15:26:58	Registry default

[Cloud IoT Core documentation](#)

# Take pictures and send over MQTT

MiNiFi

	<b>Take pictures</b> ExecuteProcess 1.10.0-SNAPSHOT org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

```
pi@raspberrypi:~/aceu19/raspberry $ cat takePicture.py
from picamera import PiCamera
from gpiozero import LED

imagePath = '/home/pi/aceu19/raspberry/pictures/picture.jpg'

amber = LED(27)

amber.on()

camera = PiCamera()
camera.rotation = 180
camera.capture(imagePath, quality = 10)

amber.off()
```

	<b>ListFile</b> ListFile 1.10.0-SNAPSHOT org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

	<b>FetchFile</b> FetchFile 1.10.0-SNAPSHOT org.apache.nifi - nifi-standard-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

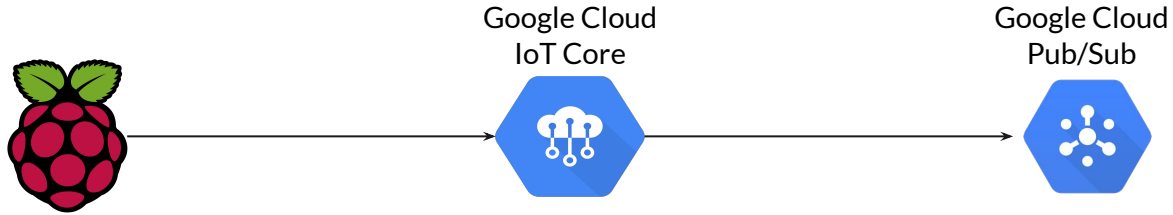
	<b>IoTDeviceMQTT</b> IoTDeviceMQTT 1.10.0-SNAPSHOT org.apache.nifi - nifi-mqtt-nar	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Name success  
Queued 0 (0 bytes)

Name success  
Queued 0 (0 bytes)



# Ingest images into the dataset



**RECEIVE IMAGES AND INGEST IMAGES INTO DATASET**

The screenshot shows a NiFi workflow with the following components and data:

- ConsumeGCPubSub** (org.apache.nifi - nifi-gcp-nar):
  - In: 0 (0 bytes)
  - Read/Write: 0 bytes / 689.05 KB
  - Out: 8 (689.05 KB)
  - Tasks/Time: 8 / 00:04:48.428
- PutGCSObject** (org.apache.nifi - nifi-gcp-nar):
  - In: 8 (689.05 KB)
  - Read/Write: 689.05 KB / 0 bytes
  - Out: 8 (689.05 KB)
  - Tasks/Time: 8 / 00:00:04.057
- To input**:
  - Name success
  - Queued 0 (0 bytes)
- Ingest data into Vision Dataset**:
  - Queued: 45 (2.86 KB)
  - In: 8 (689.05 KB) → 1
  - Read/Write: 0 bytes / 520 bytes
  - Out: 0 → 0 (0 bytes)

Additional workflow details:

- Name failure**: Queued 0 (0 bytes)
- Name success**: Queued 0 (0 bytes)

**NiFi in GCP**



















- Vision
- Dashboard
- Datasets
- Models

Filter images

All images	564
Labelled	517
Unlabelled	47
Filter labels	
NOK	135
OK	382

ADD NEW LABEL

Filter images

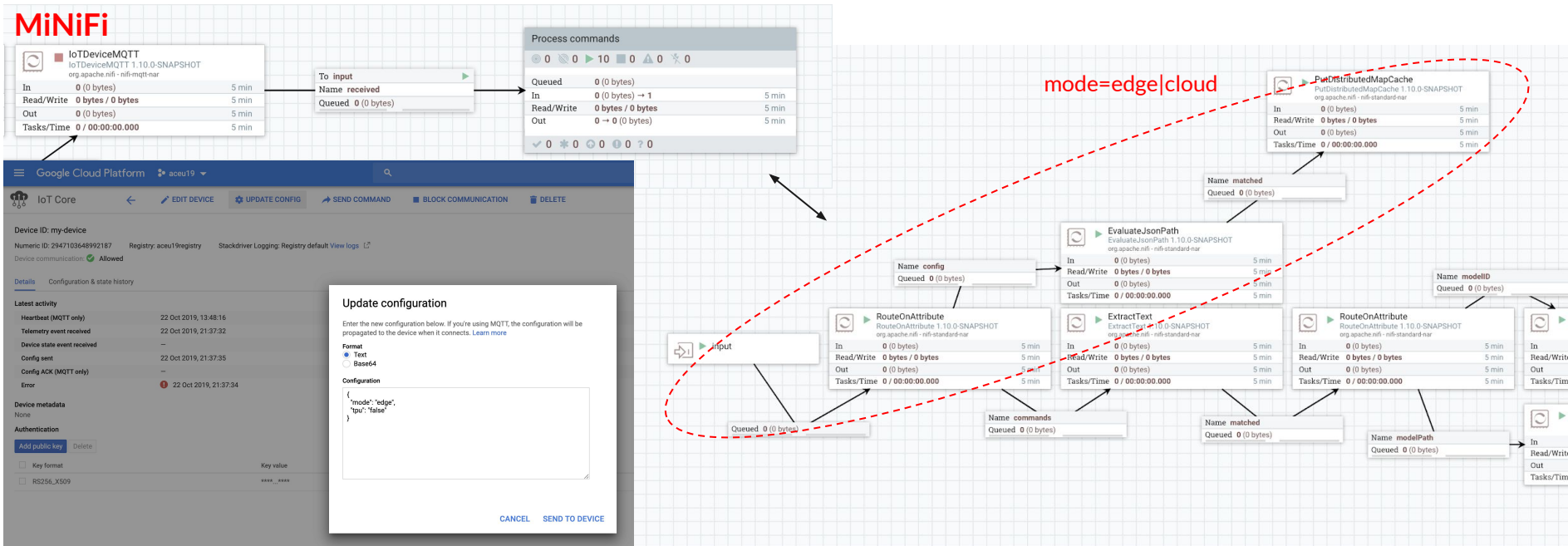
					
OK(1)	OK(1)	OK(1)	OK(1)		NOK(1)
					
OK(1)	OK(1)	OK(1)	NOK(1)	OK(1)	
					
NOK(1)	OK(1)	OK(1)	OK(1)	NOK(1)	NOK(1)



# Configure the device

# Set device mode: edge/cloud

MINIFI





# Configure NiFi running on GCP

# Variables

### Variables

Process Group  
NiFi Flow

Scope	Name	Value	
NiFi Flow	bucket	images-input-aceu19	
NiFi Flow	datasetID	ICN4695798657952251904	
NiFi Flow	deviceID	my-device	
NiFi Flow	mode	edge	
NiFi Flow	modelGcsPath	gs://aceu19-edge-models/	
NiFi Flow	projectID	aceu19	
NiFi Flow	region	europa-west1	
NiFi Flow	registryID	aceu19registry	
NiFi Flow	subscription	projects/aceu19/subscrip...	

Variables  
mode

Referencing Processors

- RouteOnAttribute
- RouteOnAttribute

Referencing Controller Services

None

Unauthorized Referencing Components

None

Variables do not support sensitive values and will be included when versioning a Process Group.

CANCEL APPLY

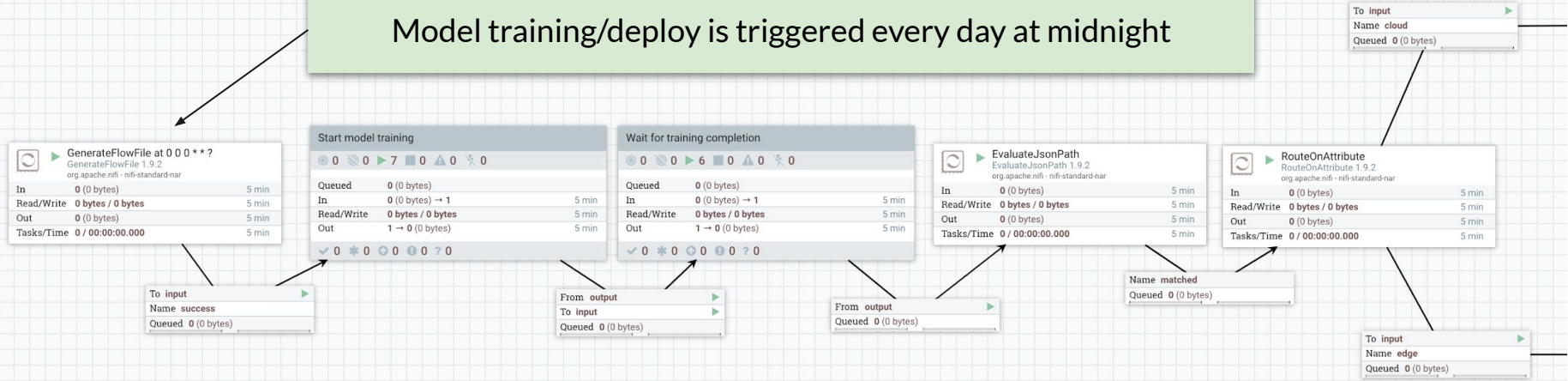


# Model training

# Automate model training

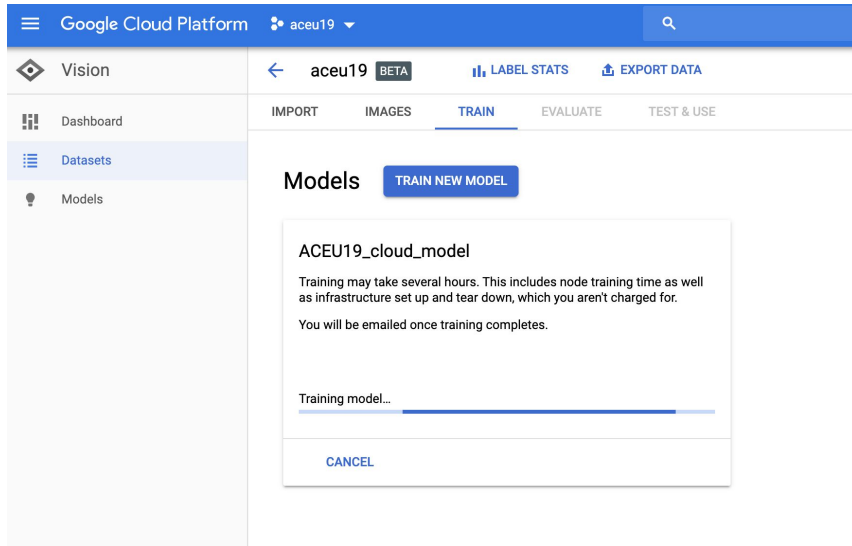
NiFi in GCP

Model training/deploy is triggered every day at midnight

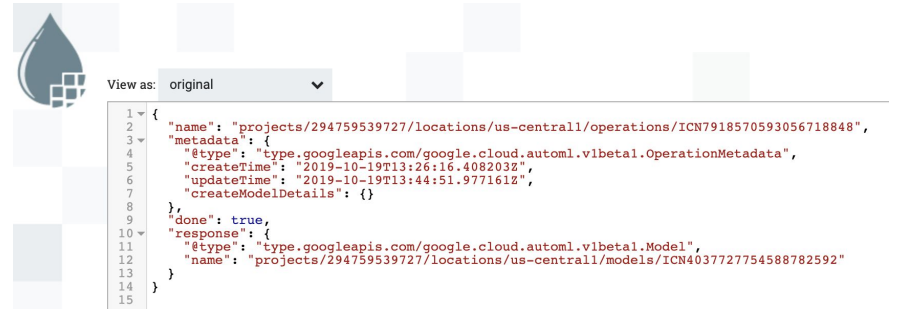




# Automate model training



The screenshot shows the Google Cloud Platform Vision API console. The left sidebar contains navigation options: Vision, Dashboard, Datasets, and Models. The main content area is titled 'Models' and shows a progress bar for 'ACEU19\_cloud\_model'. The progress bar is approximately 75% full. Below the progress bar is a 'CANCEL' button. The text above the progress bar states: 'Training may take several hours. This includes node training time as well as infrastructure set up and tear down, which you aren't charged for. You will be emailed once training completes.'



The screenshot shows a JSON response from the Google Cloud Platform Vision API. The response is displayed in a code editor with a 'View as: original' dropdown menu. The JSON object contains metadata and response details for a model training operation.

```
1 {
2   "name": "projects/294759539727/locations/us-central1/operations/ICN7918570593056718848",
3   "metadata": {
4     "@type": "type.googleapis.com/google.cloud.automl.v1beta1.OperationMetadata",
5     "createTime": "2019-10-19T13:26:16.408203Z",
6     "updateTime": "2019-10-19T13:44:51.977161Z",
7     "createModelDetails": {}
8   },
9   "done": true,
10  "response": {
11    "@type": "type.googleapis.com/google.cloud.automl.v1beta1.Model",
12    "name": "projects/294759539727/locations/us-central1/models/ICN4037727754588782592"
13  }
14 }
15 }
```

- Vision
- Dashboard
- Datasets
- Models

Model: ACEU19\_cloud\_model

Confidence threshold: 0.85

- Filter labels
- All labels
  - NOK**
  - OK

### NOK

Total images	497
Test items	0
Precision	100%
Recall	100%

Use the slider to see which confidence threshold works best for your model on the precision-recall tradeoff curve.  
[Learn more about these metrics and graphs.](#)

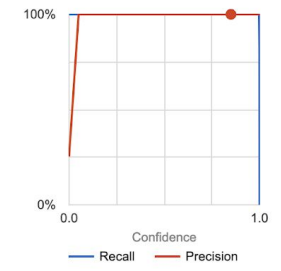
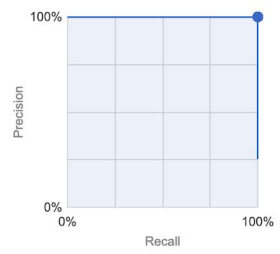
All test images are evaluated at the time of training. If you modify your dataset after training, these results will not be accurate.

### True positives

Your model correctly predicted **NOK** on these images



Score: 0.9975636    Score: 0.9988317    Score: 0.9988655    Score: 0.99907696    Score: 0.99919194    Score: 0.99943775



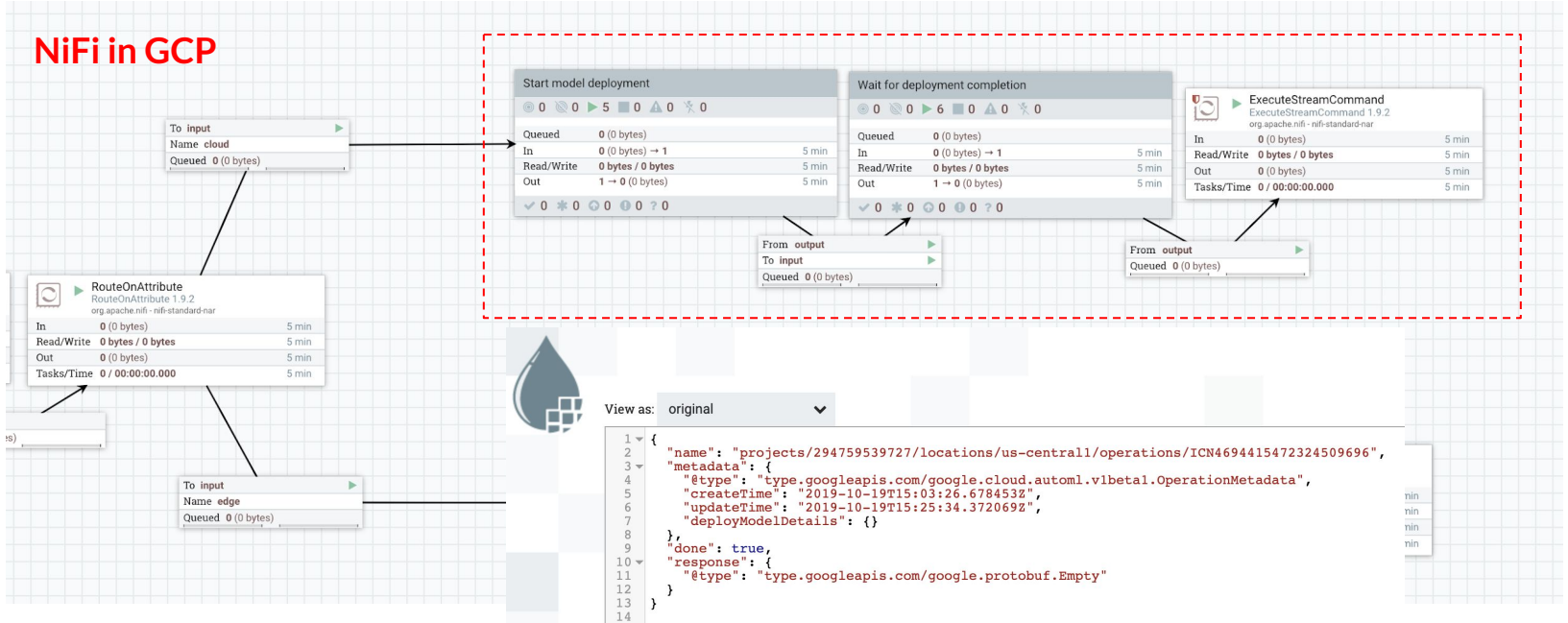


# Architecture #1 - Model deployment

**TL;DR - the model is running in the cloud**

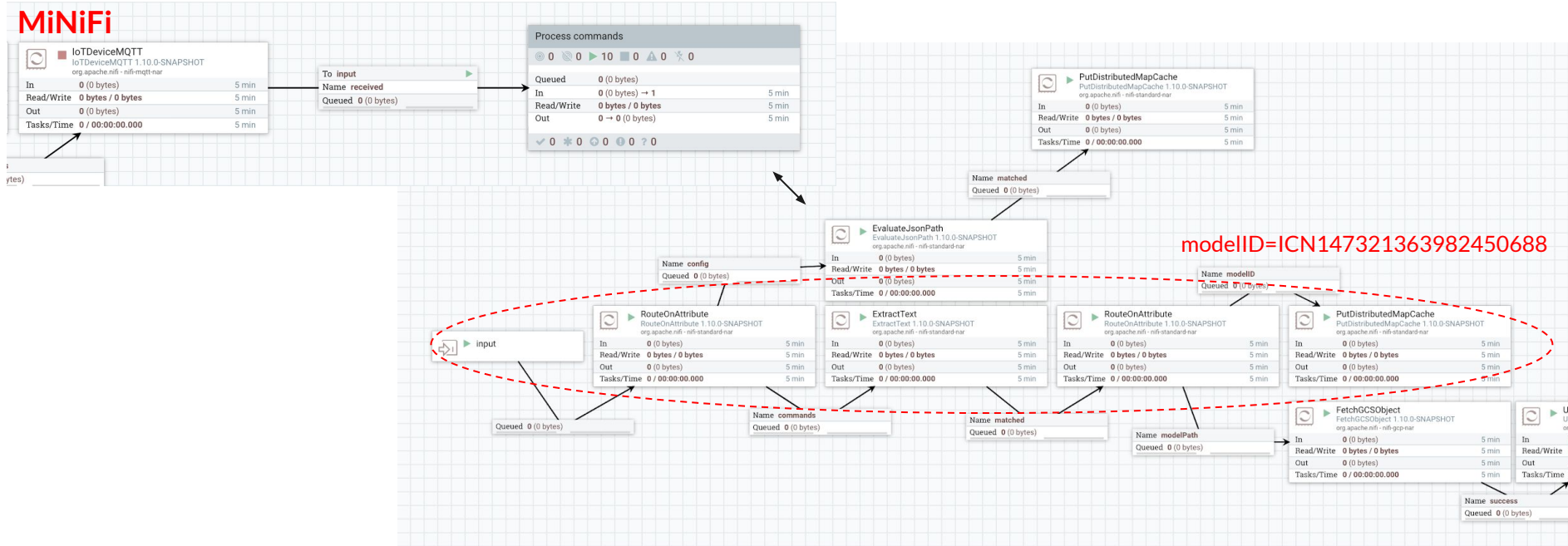
# Cloud model deployment

## NiFi in GCP



# Update on the device

MINIFI



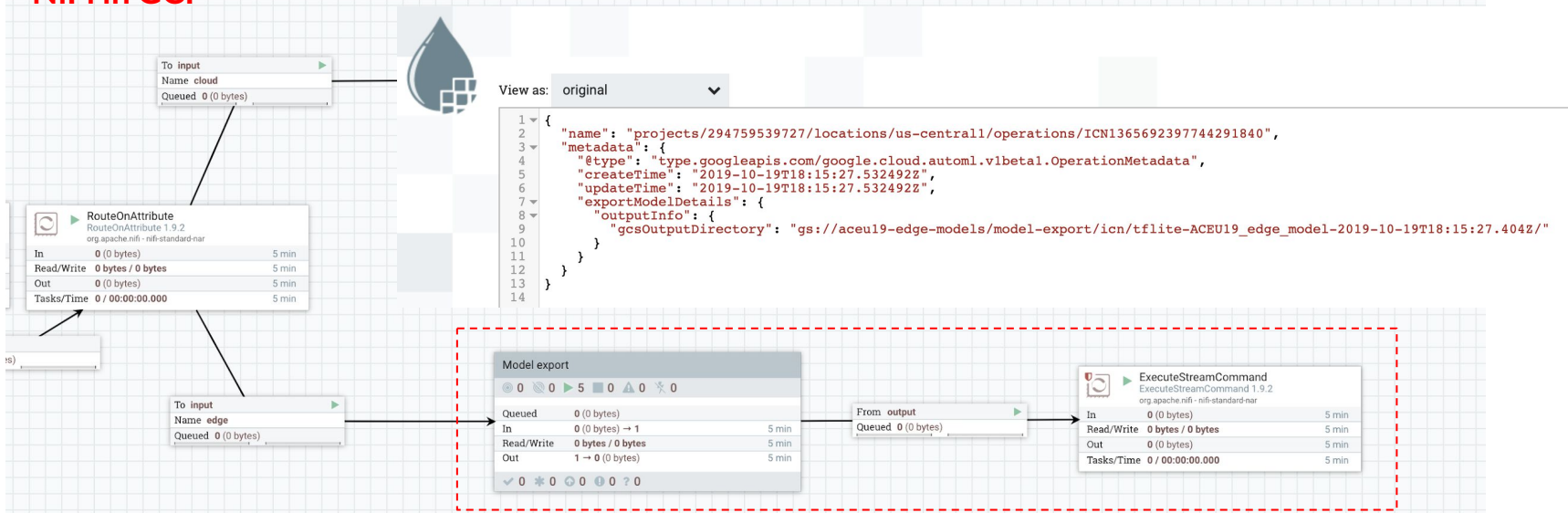


## **Architecture #2 - Model export**

**TL;DR - the model is running on the edge**

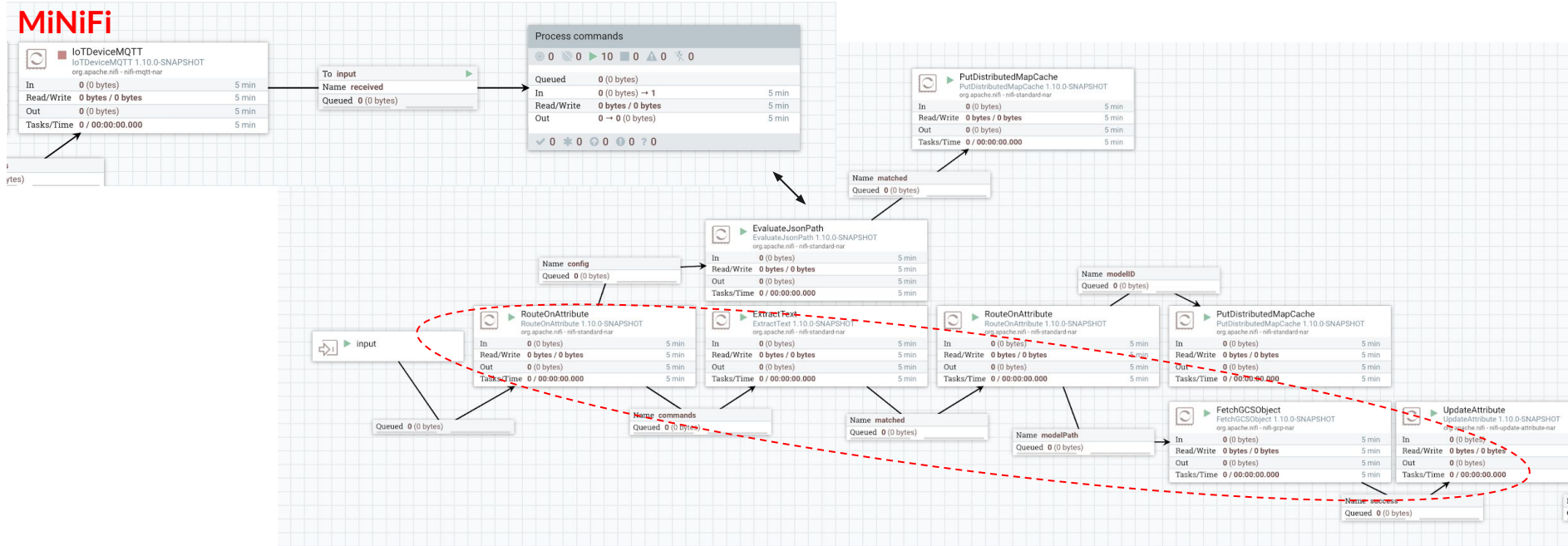
# Edge model export into Google Cloud Storage

## NiFi in GCP



# Update on the device

MINIFI

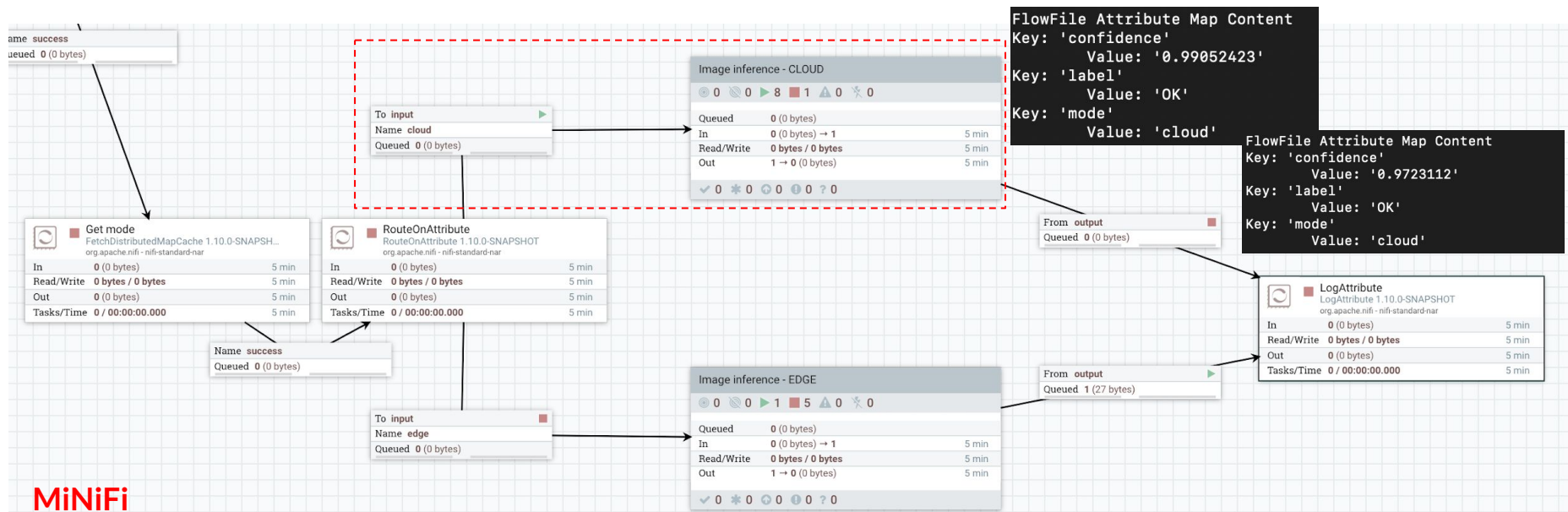




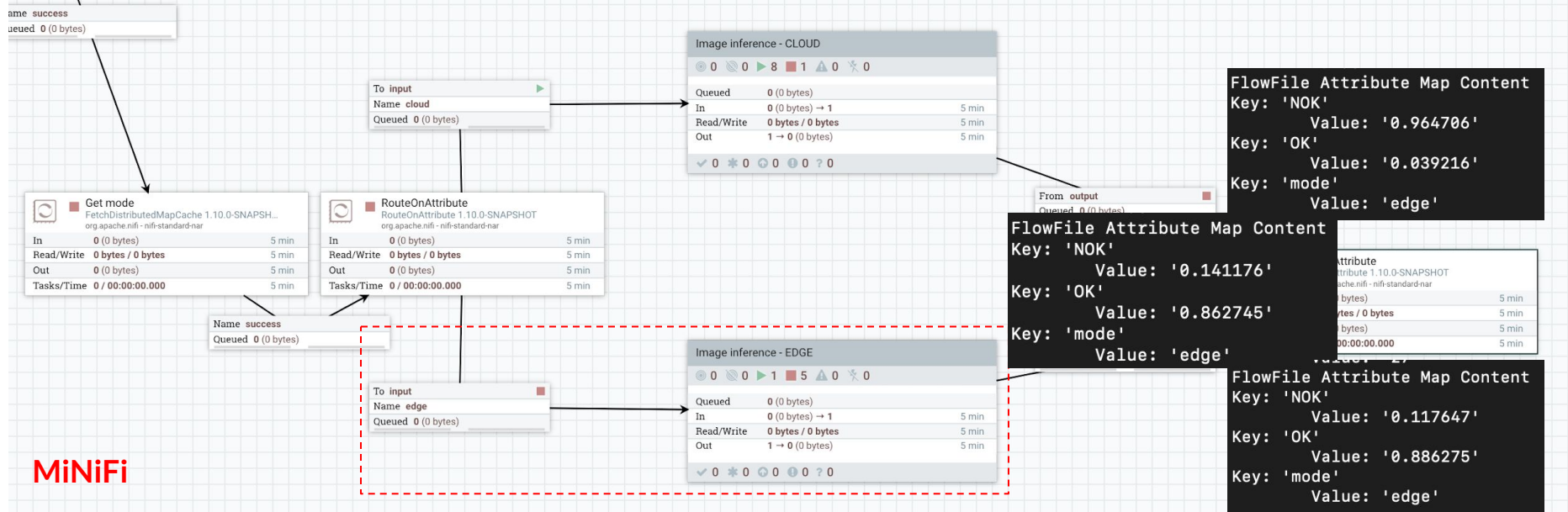


# Inference

# Inference with model in the cloud



# Inference with model on the edge



---

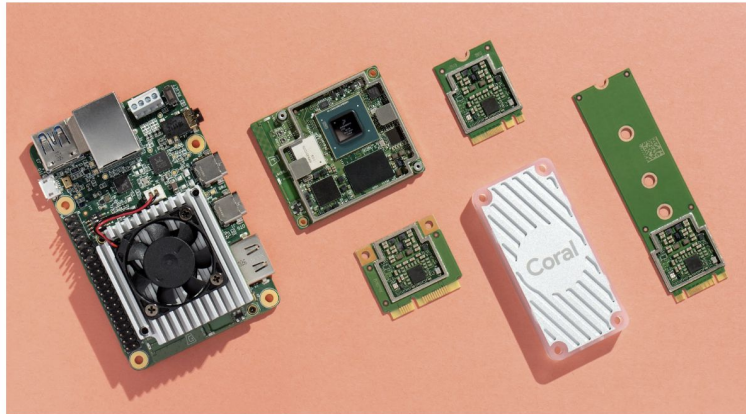
# Boosting your edge device with Google's TPU

Coral moves out of beta

Tuesday, October 22, 2019

<https://developers.googleblog.com/2019/10/coral-moves-out-of-beta.html>

*Posted by Vikram Tank (Product Manager), Coral Team*



 Google Cloud

<https://coral.ai/>



# Processing & Inference time

Architecture	Processing time	Inference time
Model in the cloud	about 6 seconds	about 2.5 seconds
Model on the edge	about 750 milliseconds	about 127 milliseconds
Model on the edge + Coral USB accelerator	about 500 milliseconds	about 9 milliseconds



# Model optimized for Edge TPU

```
pi@raspberrypi:~/aceu19/raspberry/inference $ time ./inference/bin/python3 classify_image.py --model model.tflite --label labels.txt --input image.jpg
INFO: Initialized TensorFlow Lite runtime.
----INFERENCE TIME----
Note: The first inference on Edge TPU is slow because it includes loading the model into Edge TPU memory.
136.8ms
127.4ms
127.4ms
127.7ms
127.3ms
-----RESULTS-----
NOK: 0.66797
```

Model non optimized for Edge TPU

```
pi@raspberrypi:~/aceu19/raspberry/inference $ time ./inference/bin/python3 classify_image.py --model edgetpu_model.tflite --label labels.txt --input image.jpg
INFO: Initialized TensorFlow Lite runtime.
----INFERENCE TIME----
Note: The first inference on Edge TPU is slow because it includes loading the model into Edge TPU memory.
32.9ms
9.2ms
9.2ms
9.0ms
9.1ms
-----RESULTS-----
NOK: 0.67969
```

Model optimized for Edge TPU

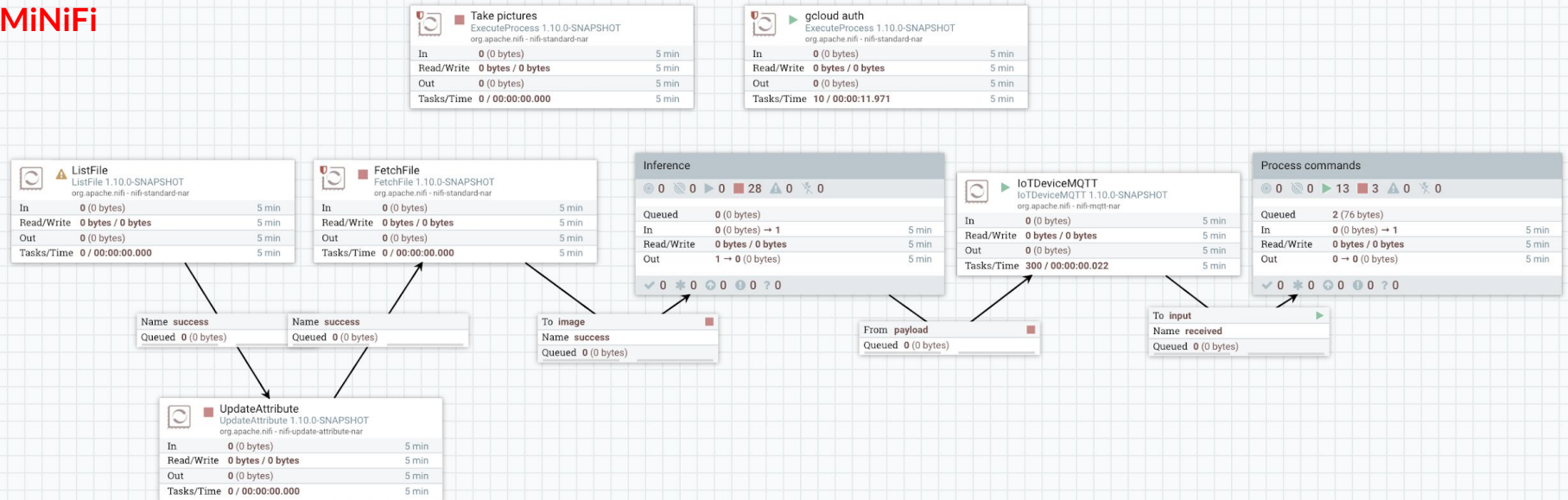


## Next steps

- Auto-labeling when confidence is over a given threshold (example: 0.90)
  - Will drastically reduce human effort to label newly captured data
- Send inference results along with pictures
  - Allow performance monitoring over time, detect outliers and inference performance

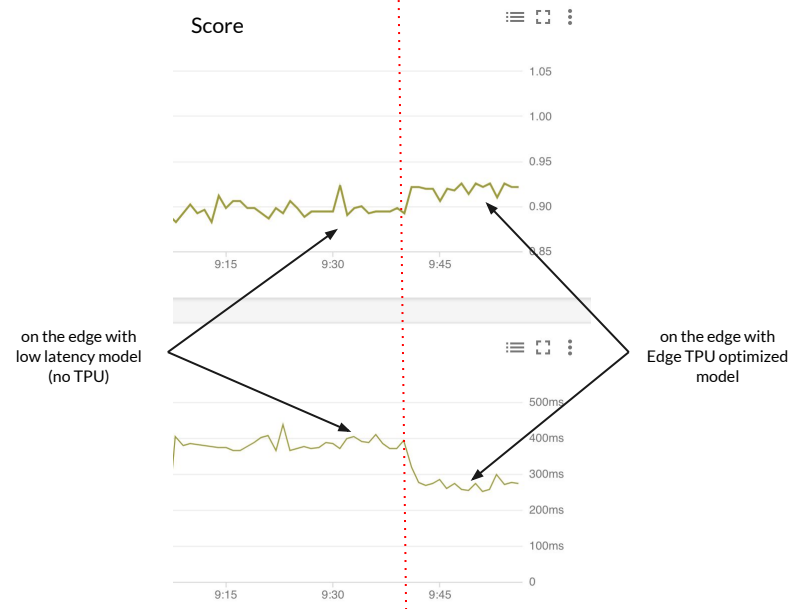
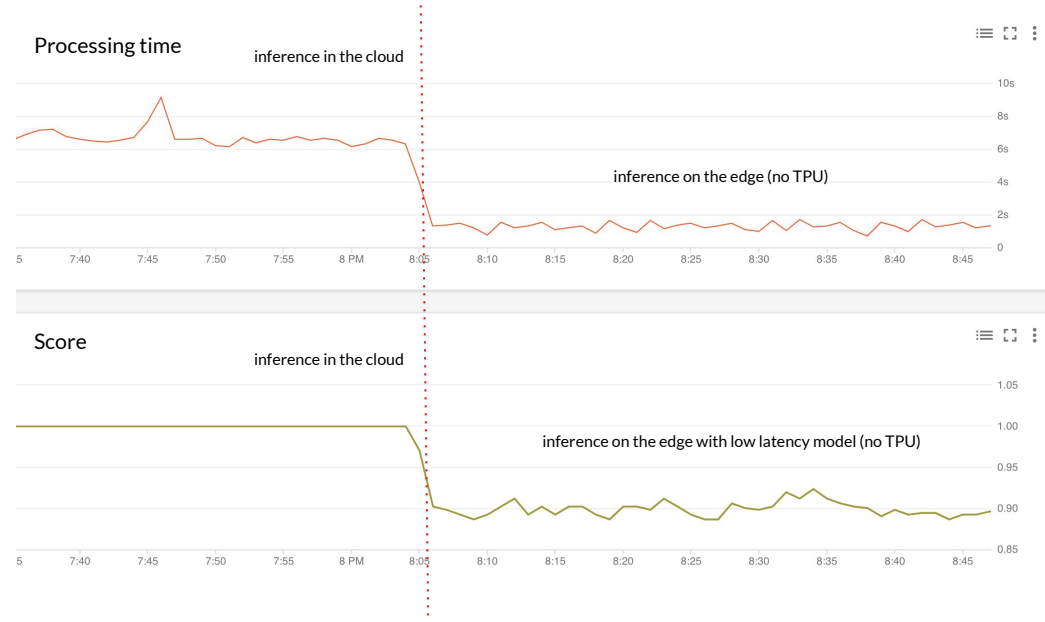
# Sending inference results along with pictures

MINiFi





# Monitoring dashboards in Stackdriver

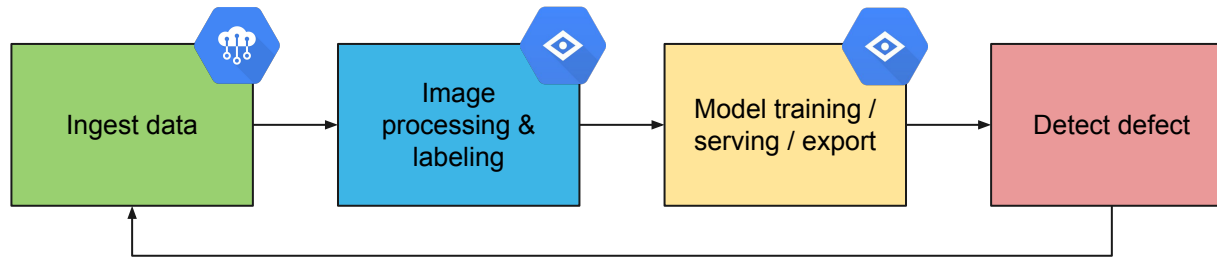


# Analytics in BigQuery

The screenshot displays the Google Cloud Platform BigQuery interface. At the top, the navigation bar shows 'Google Cloud Platform' and the user 'aceu19'. Below this, the 'BigQuery' header includes links for 'FEATURES & INFO' and 'SHORTCUTS'. The left sidebar contains a navigation menu with categories like 'Query history', 'Saved queries', 'Job history', 'Transfers', 'Scheduled queries', 'BI Engine', and 'Resources'. Under 'Resources', a search bar is present, and a tree view shows the project 'aceu19' with sub-items 'monitoring' and 'inference'. The main area is split into a 'Query editor' and a 'Query results' section. The query editor contains a SQL query: `1 SELECT mode, tpu, AVG(duration_ms) as duration_ms, AVG(inference_s)*1000 as inference_ms, AVG(score) as score  
2 FROM monitoring.inference  
3 GROUP BY mode, tpu`. Below the editor are buttons for 'Run', 'Save query', 'Save view', 'Schedule query', and 'More'. The 'Query results' section shows a status message: 'Query complete (1.4 sec elapsed, 24.3 KB processed)'. It includes tabs for 'Job information', 'Results', 'JSON', and 'Execution details'. A table displays the results with columns: 'Row', 'mode', 'tpu', 'duration\_ms', 'inference\_ms', and 'score'.

Row	mode	tpu	duration_ms	inference_ms	score
1	edge	false	1267.0166666666667	385.0352170760269	0.8984375555555554
2	edge	true	1202.5081967213114	273.518443081841	0.9245022950819674

## Conclusion - thanks to NiFi and GCP:



- Codeless deployment of customized ML models on the edge
- Feedback loop and continuously updated models
- Processing on the edge and optimization with TF Lite et Coral Edge TPU

THANKS!

# Running visual quality inspection at the edge with Apache NiFi & MiNiFi

---

Pierre Villard - @pvillard31



<https://nifi.apache.org>

<https://github.com/pvillard31/aceu19>

APACHE



APACHECON

EUROPE Oct. 22<sup>nd</sup> - 24<sup>th</sup>

# 2019