



CRACKING THE NUT

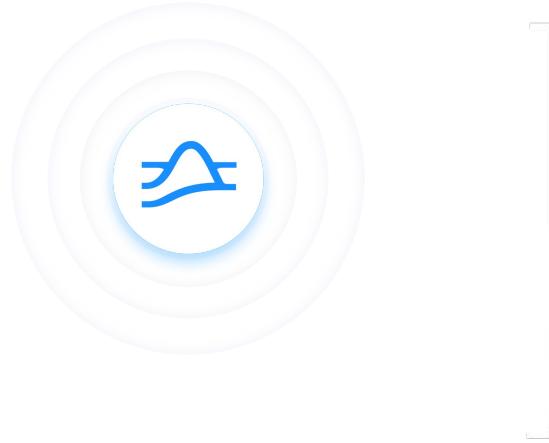
Timothy Spann & David Kjerrumgaard

Developer Advocates / StreamNative

Wednesday 22-Sept-2021



Cracking the Nut, Solving Edge AI



Agenda

Wednesday 17:10 UTC

Cracking the Nut, Solving Edge AI with Apache Tools and Frameworks

Today, data is being generated from devices and containers living at the edge of networks, clouds and data centers. We need to run business logic, analytics and deep learning at the edge before we start our real-time streaming flows. Fortunately using the all Apache FLiP Stack we can do this with ease!

Streaming AI Powered Analytics From the Edge to the Data Center is now a simple use case. With MiNiFi we can ingest the data, do data checks, cleansing, run machine learning and deep learning models and route our data in real-time to Apache NiFi and Apache Pulsar for further transformations and processing. Apache Flink will provide our advanced streaming capabilities fed real-time via Apache Kafka topics. Apache MXNet models will run both at the edge and in our data centers via Apache NiFi and MiNiFi. Our final data will be stored in various Apache datastores. Event-Driven Microservices in Apache Pulsar Functions.

Tools:

Apache Flink, Apache Pulsar, Apache NiFi, MiNiFi, Apache MXNet

My Other Talks & Apache Pulsar Talks

- Tuesday 14:10 UTC - Exclusive Producer: Using Apache Pulsar to build distributed applications by Matteo Merli
- Wednesday 15:50 UTC - Replicated Subscriptions: taking Apache Pulsar Geo-Replication to next level by Matteo Merli
- Tuesday 17:10 UTC - Apache NiFi Deep Dive 300 by Tim Spann
- Tuesday 18:00 UTC - Apache Deep Learning 302 by Tim Spann
- Wednesday 15:00 UTC - Smart Transit: Real-Time Transit Information with FLiP by David Kjerrumgaard & Tim Spann
- Wednesday 17:10 UTC - Cracking the Nut, Solving Edge AI... by David Kjerrumgaard & Tim Spann
- Thursday 14:10 UTC - Apache NiFi 101: Introduction and Best Practices - Tim Spann

Stay Connected With Us!



David Kjerrumgaard
Developer Advocate



<https://twitter.com/DavidKjerrumga1>



<https://github.com/david-streamlio>



<https://www.linkedin.com/in/davidkji/>



Tim Spann
Developer Advocate



<https://twitter.com/paasDev>



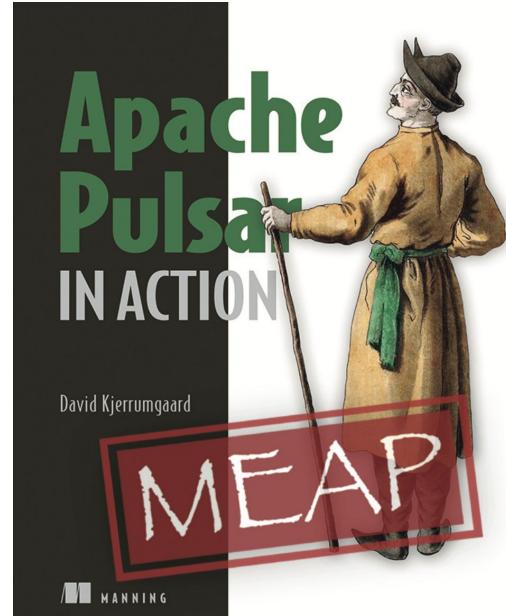
<https://github.com/tspannhw>



<https://www.linkedin.com/in/timothyspann>

Speaker Bio

Developer Advocate



<https://github.com/david-streamlio>

<https://pulsar-summit.org/en/event/virtual-conference-2020/speaker/david-kjerrumgaard>

<https://www.slideshare.net/streamnative/using-apache-pulsar-to-provide-realtime-iot-analytics-on-the-edge>

 StreamNative

Speaker Bio

Developer Advocate

DZone Zone Leader and Big Data MVB;
@PaasDev

<https://github.com/tspannhw>

<https://www.datainmotion.dev/>

<https://github.com/tspannhw/SpeakerProfile>

<https://dev.to/tspannhw>

<https://sessionize.com/tspann/>

<https://www.slideshare.net/bunkertor>



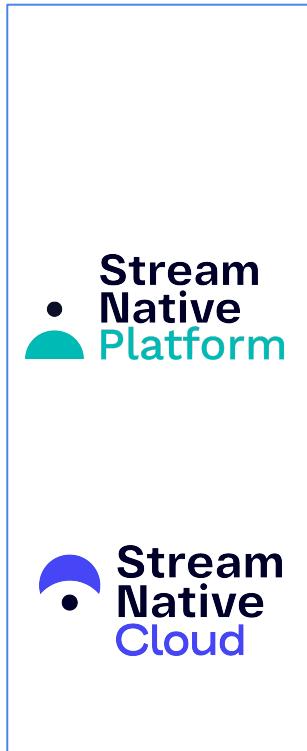
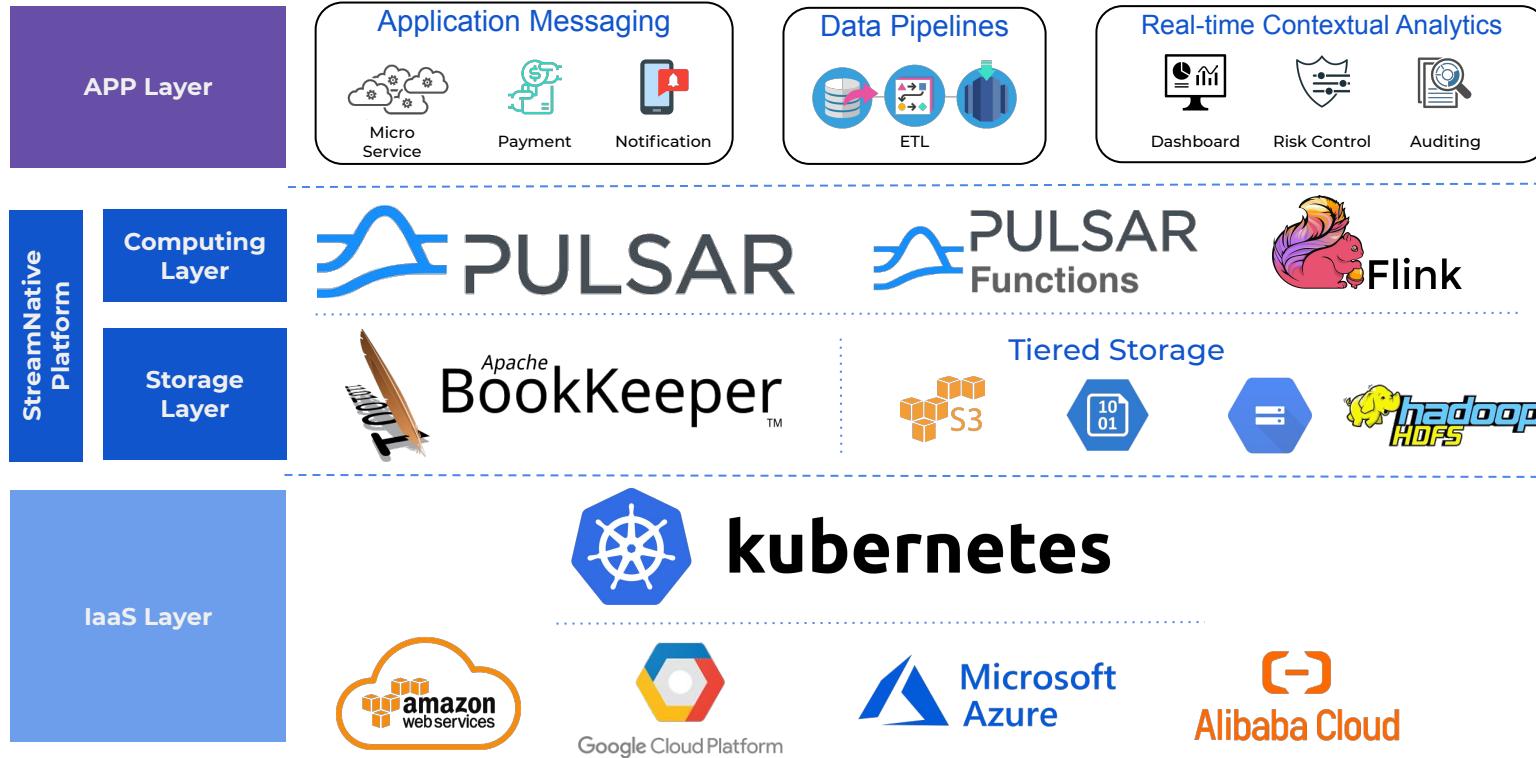
FLaNK and FLiP Stacks

- Apache Flink
- Apache NiFi
- Apache Kafka
- Apache Flink
- Apache Pulsar
- StreamNative's Flink Connector for Pulsar
- Apache ++

Apache projects are the way for all streaming use cases.



StreamNative Solution



Apache Tools and Frameworks Used



MQTT on Pulsar (MoP)

FLiP Stack for Cloud Data Engineers - Edge AI

Multiple users, frameworks, languages, clouds, data sources & clusters



CLOUD DATA ENGINEER

- Experience in ETL/ELT
- Coding skills in Python or Java
- Knowledge of database query languages such as SQL
- Experience with Streaming
- Knowledge of Cloud Tools



CAT

- Expert in ETL (Eating, Ties and Laziness)
- Edge Camera Interaction
- Typical User
- No Coding Skills
- Can use NiFi
- Questions your cloud spend



AI / Deep Learning / ML / DS

- Can run in Apache NiFi
- Can run in Pulsar Functions
- Can run in Apache Flink



Function Mesh

<https://functionmesh.io/>

Apache Tools and Frameworks Used



MQTT on Pulsar (MoP)

Apache MXNet Native Processor through DJL.AI for Apache NiFi



#workshop

Deep Learning Class Label: person

File: cc0a469f-c108-42c7-95c6-10e5fda95006.person.png

Probability: 0.96

UUID: 32ef65a3-0650-42cd-965c-ba25597eb1ad

Rank: 1

Bounding Box (Height/Width, X,Y)

0.74 / 0.69

0.27, 0.25

Image (Height/Width, X,Y)

480 / 640

0, 0

tspann 11:30 AM 371bdb8f-35bc-4a2a-919c-bdeb609b726c.person.png



```
=====
    return Paths.get(url).toAbsolutePath().getParent().toString();
}

private void runMockUserDeploy() {
    testRunner.setValidateExpressionsUsage(false);
    testRunner.run();
    testRunner.assertValid();
}

testRunner.assertAllFlowFilesTransferred(DeepLearningProcessor.REL_SUCCESS);
List<MockFlowFile> successfuls = testRunner.getFlowFilesForRelationship(DeepLearningProcessor.REL_SUCCESS);

for (MockFlowFile mockFile : successfuls) {
    assertEquals("car", mockFile.getAttribute("category"));
    assertEquals("1.0", mockFile.getAttribute("probability"));

    System.out.println("MockString, String- attributes = " + mockFile.getAttribute("boundingbox_x_min"));
    System.out.println("MockString, String- attributes = " + mockFile.getAttribute("boundingbox_y_min"));
    System.out.println("MockString, String- attributes = " + mockFile.getAttribute("boundingbox_x_max"));
    System.out.println("MockString, String- attributes = " + mockFile.getAttribute("boundingbox_y_max"));

    for (String attribute : attributes.keySet()) {
        System.out.println("Attribute:" + attribute);
    }
}

@test
public void testProcessor() throws Exception {
    java.io.File resourcesDirectory = new java.io.File("resources");
    System.out.println(resourcesDirectory.getAbsolutePath());
    testRunner.setProperty(DeepLearningProcessor.MACROS);
    testRunner.setProperty(DeepLearningProcessor.DATASET);
    DeepLearningProcessorTest testProcessor = new DeepLearningProcessorTest();
    testProcessor.start();
    testProcessor.onTrigger(testProcessor);
    testProcessor.stop();
}

Run: DeepLearningProcessorTest.testProcessor
=====
  Tests passed: 1 of 1 test - 4 s 0.08 ms
Size:17632B
Attribute:boundingbox_x_min = 0.25
Attribute:boundingbox_y_min = 0.0
Attribute:boundingbox_x_max = 1.0
Attribute:boundingbox_y_max = 0.69
Attribute:category = car
Attribute:rnd_1 = 1
Attribute:uuid = e9993c52-f5ab-4849-8876-a25796714984
Attribute:boundingbox_width_1 = 0.24

```

Attribute Values

boundingbox_height_1

0.99

No value set

boundingbox_width_1

0.90

No value set

boundingbox_x_1

0.09

No value set

boundingbox_y_1

0.01

No value set

class_1

tmonitor

No value set

filename

2020-08-26_1330.jpg.tmonitor.png

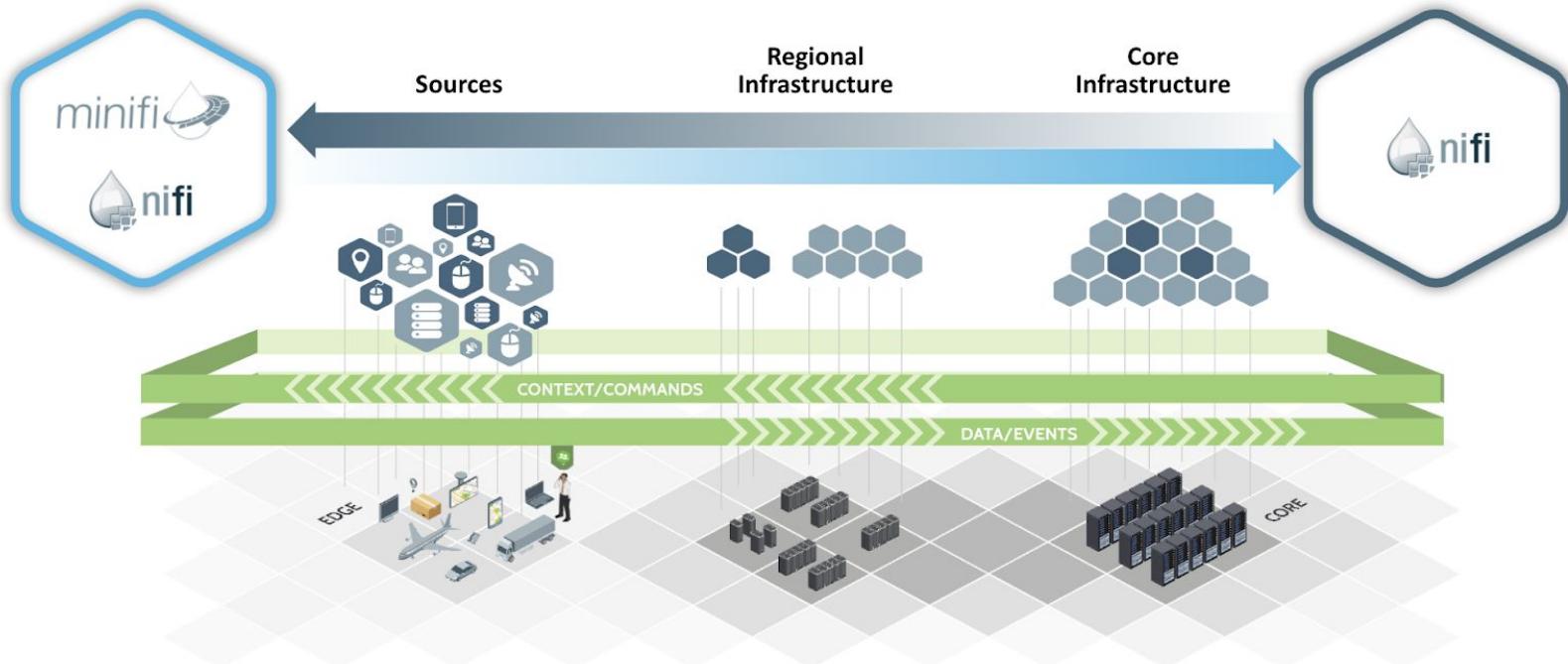
2020-08-26_1330.jpg (previous)

This processor uses the DJL.AI Java Interface

<https://github.com/tspannhw/nifi-djl-processor>

<https://dev.to/tspannhw/easy-deep-learning-in-apache-nifi-with-djl-2d79>

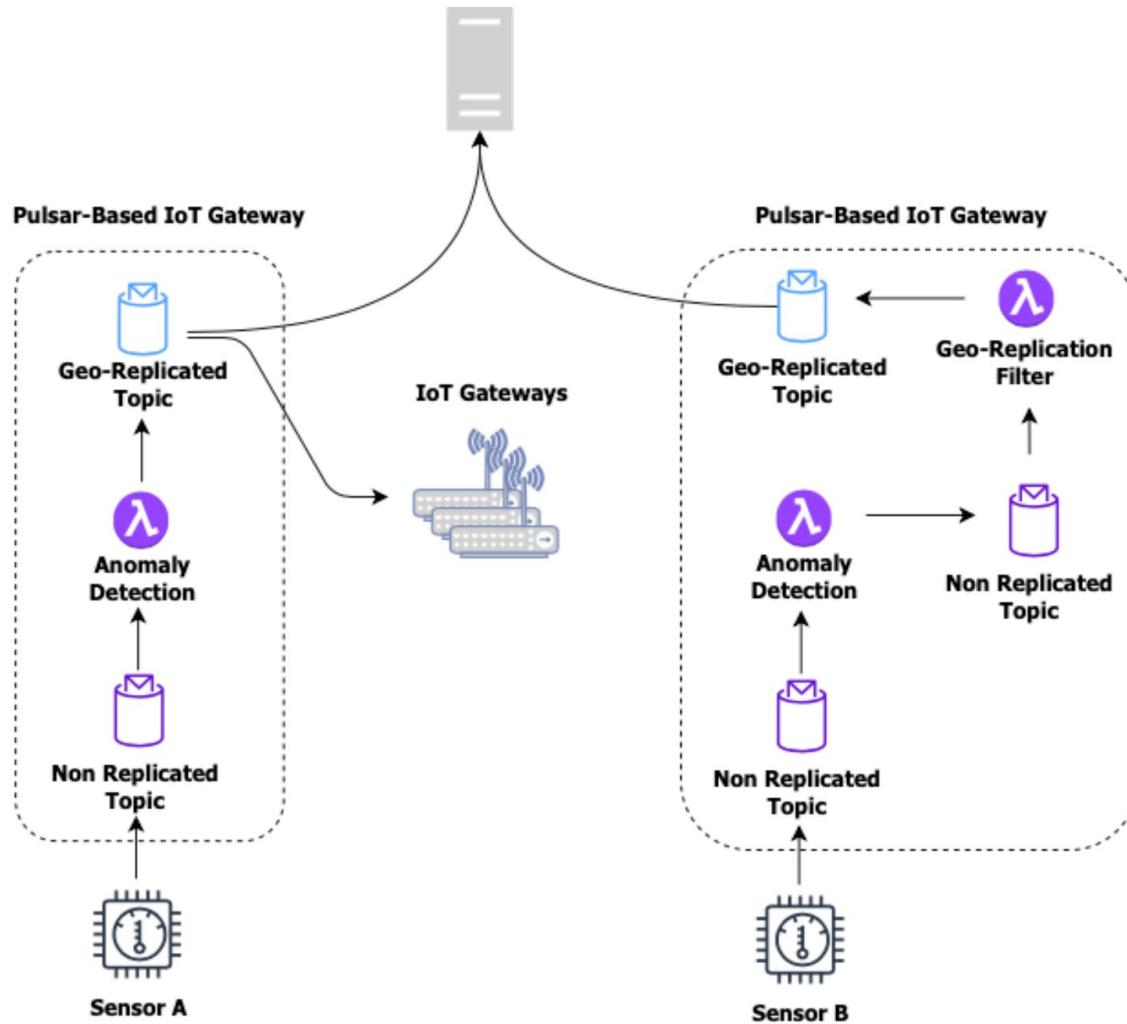
What is Apache NiFi and MiNiFi used for?

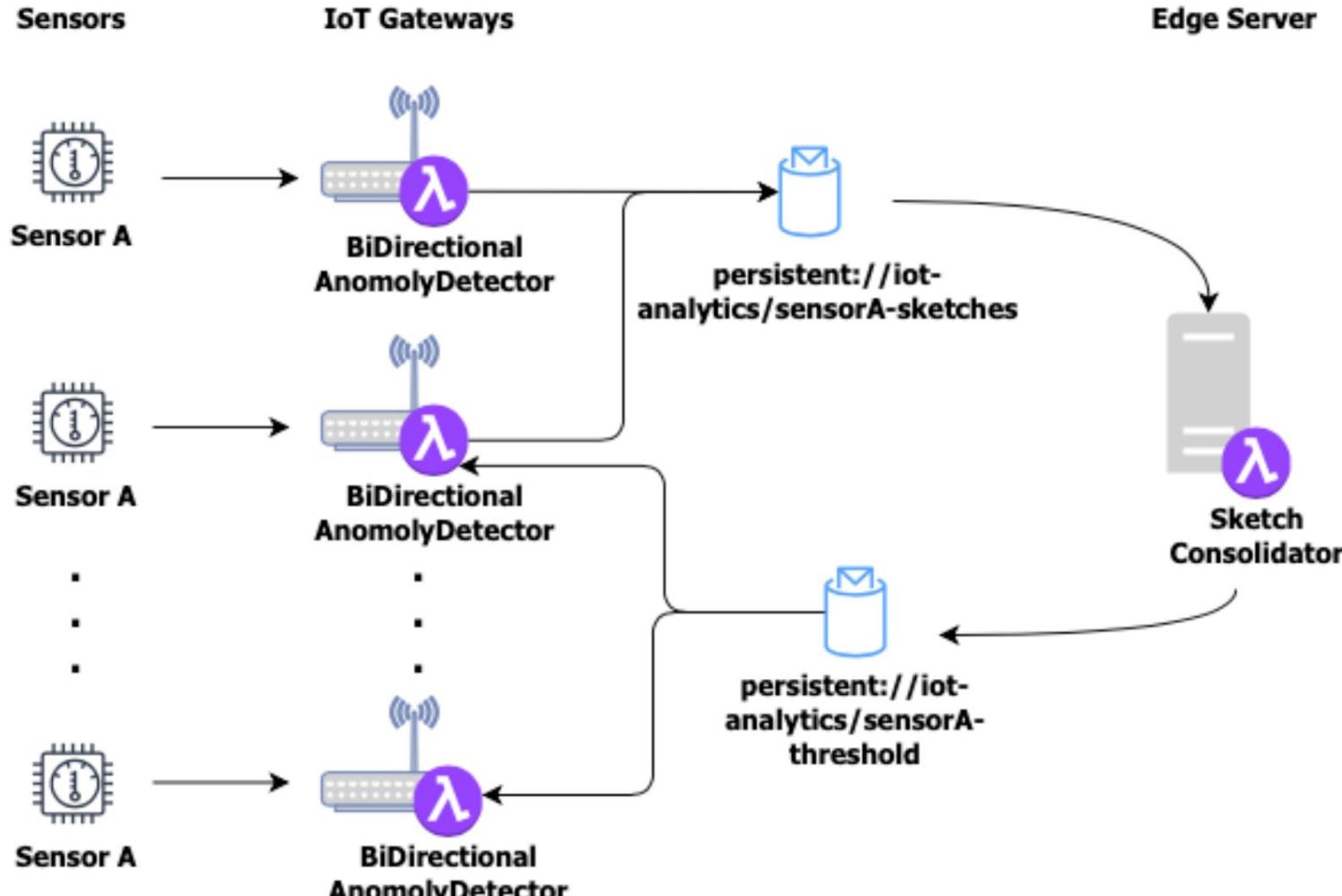


A Unified Messaging Platform



Demo





	ConsumePulsar	ConsumePulsar 1.11.0 org.apache.nifi - nifi-pulsar-nar		1
In	0 (0 bytes)	5 min		
Read/Write	0 bytes / 0 bytes	5 min		
Out	0 (0 bytes)	5 min		
Tasks/Time	6,795,105 / 00:03:13.342	5 min		

Name success
Queued 0 (0 bytes)

	Queries on Humidity	QueryRecord 1.14.0 org.apache.nifi - nifi-standard-nar		1
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 all
Queued 0 (0 bytes)

	Send to Cloud Postgresql	PutDatabaseRecord 1.14.0 org.apache.nifi - nifi-standard-nar		1
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 failure
Queued 0 (0 bytes)

	RetryFlowFile	RetryFlowfile 1.14.0 org.apache.nifi - nifi-standard-nar		1
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		

	ConsumeKafka_2_6	ConsumeKafka_2_6 1.14.0 org.apache.nifi - nifi-kafka-2-6-nar		1
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)

	Identify Data Origin	UpdateAttribute 1.14.0 org.apache.nifi - nifi-update-attribute-nar		1
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)



	ConsumeMQTT	ConsumeMQTT 1.14.0 org.apache.nifi - nifi-mqtt-nar		1
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 Message
Queued 0 (0 bytes)

	Set Reader Name and Destination	UpdateAttribute 1.14.0 org.apache.nifi - nifi-update-attribute-nar		1
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)

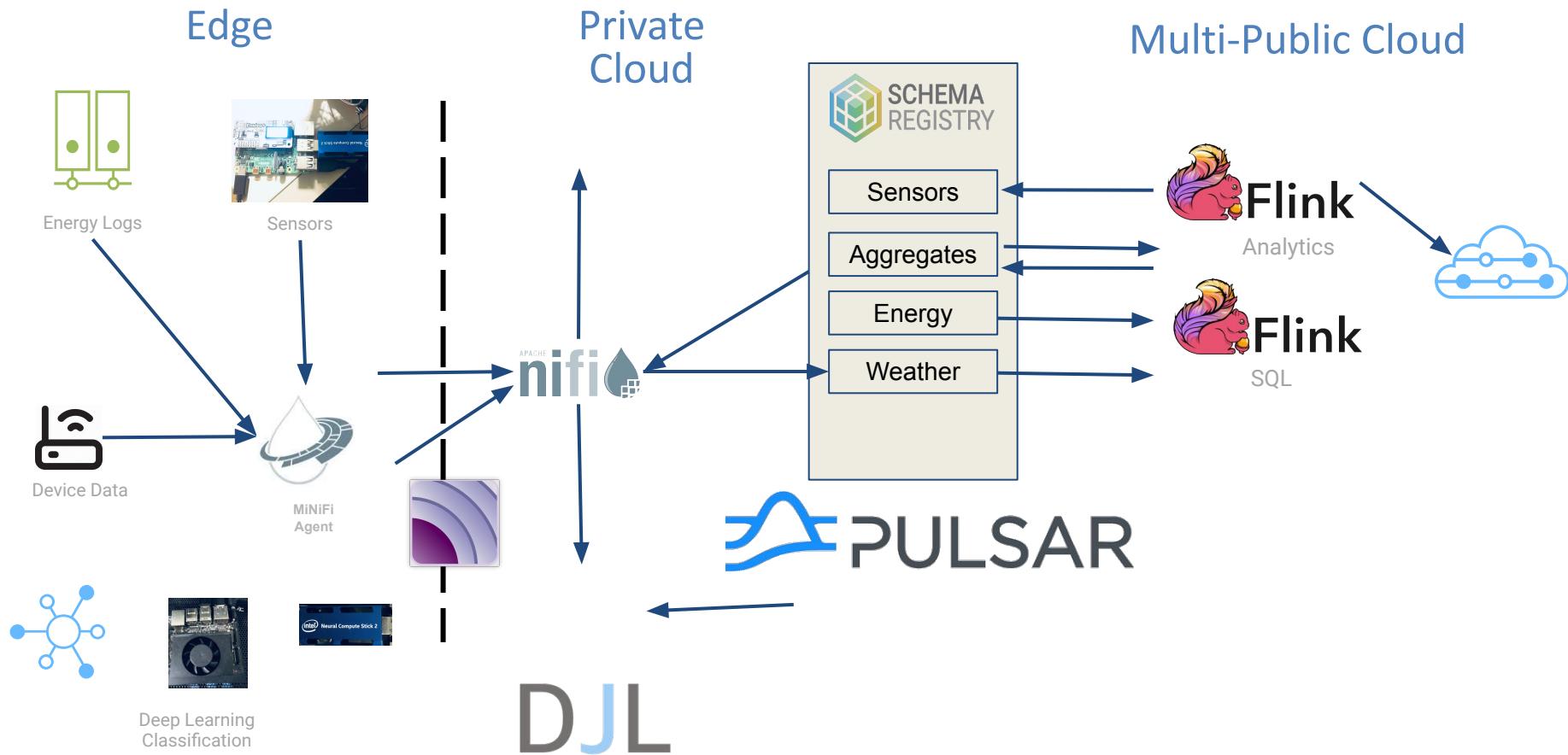


Name retry
Queued 0 (0 bytes)

Name retry
Queued 0 (0 bytes)

Name failure
Queued 0 (0 bytes)

Edge AI to Cloud Streaming Pipeline



SHOW ME THE DATA

```
{"uuid": "rpi4_uuid_jfx_20200826203733", "amplitude100": 1.2, "amplitude500": 0.6, "amplitude1000": 0.3, "lownoise": 0.6, "midnoise": 0.2, "highnoise": 0.2, "amps": 0.3, "ipaddress": "192.168.1.76", "host": "rp4", "host_name": "rp4", "macaddress": "6e:37:12:08:63:e1", "systemtime": "08/26/2020 16:37:34", "endtime": "1598474254.75", "runtime": "28179.03", "starttime": "08/26/2020 08:47:54", "cpu": 48.3, "cpu_temp": "72.0", "diskusage": "40219.3 MB", "memory": 24.3, "id": "20200826203733_28ce9520-6832-4f80-b17d-f36c21fd8fc9", "temperature": "47.2", "adjtemp": "35.8", "adjtempf": "76.4", "temperatureref": "97.0", "pressure": 1010.0, "humidity": 8.3, "lux": 67.4, "proximity": 0, "oxidising": 77.9, "reducing": 184.6, "nh3": 144.7, "gasKO": "Oxidising: 77913.04 Ohms\nReducing: 184625.00 Ohms\nNH3: 144651.47 Ohms"}
```



WHERE DID THAT DATA COME FROM?

BME280 - temperature, pressure, humidity sensor

LTR-559 - light and proximity sensor

MICS6814 - analog gas sensor

ADS1015 ADC

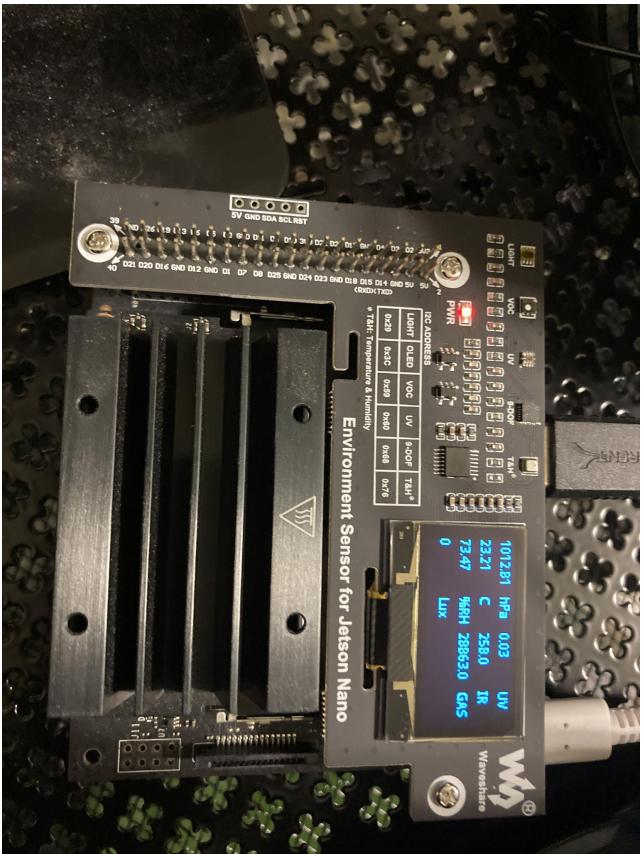
MEMS - microphone

0.96-inch, 160 x 80 color LCD

MANY DEVICES



NVIDIA JETSON WITH ENVIRONMENTAL SENSOR



Using NVIDIA Jetson Devices With Pulsar

<https://dev.to/tspannhw/unboxing-the-most-amazing-edge-ai-device-part-1-of-3-nvidia-jetson-xavier-nx-595k>

<https://github.com/tspannhw/minifi-xaviernx/>

<https://github.com/tspannhw/minifi-jetson-nano>

<https://github.com/tspannhw/Flip-iot>

<https://www.datainmotion.dev/2020/10/flank-streaming-edgeai-on-new-nvidia.html>



Demo Walkthrough



```
Flink SQL> use catalog pulsarcatalog;
[INFO] Execute statement succeed.

Flink SQL> show tables;
+-----+-----+
| table name | 
+-----+-----+
# click_events hello_world
kafka-1
kafka-2
kafka-3
kafka-4
kafka-5
mqtt-1
mqtt-2
mqtt-3
mqtt-4
mqtt-5
mqtt-go
mqtt-mac
mqtt-nifi
mqtt-nvidia
mqtt-python
mqtt-rp4
my-topic
nvidia-kafka-1
rp4-kafka-1
rwar
+-----+-----+
23 rows in set

Flink SQL> |
```

```
{"entriesAddedCounter":1,"numberOfEntries":1,"totalSize":651,"currentLedgerEntries":1,"currentLedgerSize":651,"lastLedgerCreatedTimestamp":"2021-09-13T16:13:06.6-04:00","waiting CursorsCount":0,"pendingAddEntriesCount":0,"lastConfirmedEntry":"7076:0","state":"LedgerOpened","ledgers":[{"ledgerId":7076,"entries":0,"size":0,"offloaded":false,"underReplicated":false}],"cursors":{},"schemaLedgers":[],"compactedLedge r":{"ledgerId":-1,"entries":-1,"size":-1,"offloaded":false,"underReplicated":false}}
```

NVIDIA JETSON WITH ENVIRONMENTAL SENSOR

```
NVIDIA Jetson Nano (Developer Kit Version) - Jetpack 4.4.1 [L4T 32.4.4]
CPU1 [|||||] Schedutil - 10%] 1.5GHz
CPU2 [|||||] Schedutil - 10%] 1.5GHz
CPU3 [|||] Schedutil - 6%] 1.5GHz
CPU4 [|||] Schedutil - 9%] 1.5GHz

Mem [|||||||||||||||||] 0.4G/2.0GB] (lfb 1x2MB)
Imm [ 0.0k/252.0kB] (lfb 252kB)
Swp [|||] 0.261GB/5.1GB] (cached 27MB)
EMC [ 0%] 1.6GHz

GPU [ 0%] 921MHz
Dsk [#####] 24.0GB/58.4GB]

[info] ————— [Sensor] ————— [Temp]
UpT: 0 days 23:36:43          A0      24.50C
FAN: [ 0%] Ta= 0%             CPU     17.50C
Jetson Clocks: running        GPU     18.00C
NV Power[0]: MAXN            PLL     15.00C
                                     thermal 17.75C

[HW engines]
APE: 25MHz
NVENC: [OFF]                  NVDEC: [OFF]
NVJPEG: [OFF]
```

NVIDIA JETSON WITH ENVIRONMENTAL SENSOR

```
key:[null], properties:[], content:{"ir": "258.0", "id":  
"20210914142805_cfc3c4ef-6639-46cb-bb16-1fa7ee6424d9", "end": "1631629865.51", "uuid":  
"nano_uuid_xmm_20210914142805", "lux": "0", "gputemp": "16.5", "cputemp": "16.5", "te": "179.529592991",  
"systemtime": "09/14/2021 10:31:05", "hum": "63.9", "memory": 21.8, "gas": "29843.0", "pressure": "1015.63",  
"host": "nano2gb-desktop", "diskusage": "32617.2 MB", "ipaddress": "192.168.1.217", "macaddress":  
"1c:bf:ce:1a:7f:a0", "temp": "16.39", "uv": "0.02", "gputempf": "62.0", "host_name": "nano2gb-desktop",  
"runtime": "180.0", "cpu": 0.0, "cputempf": "62.0"}
```

DEMO SOURCE CODE

- <https://github.com/tspannhw/minifi-xaviernx>
- <https://github.com/tspannhw/minifi-jetson-nano>
- <https://github.com/tspannhw/minifi-enviroplus>
- <https://github.com/tspannhw/EverythingApacheNiFi>
- <https://github.com/tspannhw/CloudDemo2021>
- <https://www.datainmotion.dev/2019/07/powering-edge-ai-with-powerful-jetson.html>
- <https://www.datainmotion.dev/2021/07/upcoming-apache-pulsar-and-apache-flink.html>

Deeper Content

- <https://github.com/streamnative/pulsar-flink>
- <https://www.linkedin.com/pulse/2021-schedule-tim-spann/>
- https://github.com/tspannhw/SpeakerProfile/blob/main/2021/talks/20210729_HailHydrate!FromStreamtoLake_TimSpann.pdf
- <https://streamnative.io/en/blog/release/2021-04-20-flink-sql-on-streamnative-cloud>
- <https://docs.streamnative.io/cloud/stable/compute/flink-sql>
- <https://dzone.com/articles/real-time-transit-feed-data-processing>



@PaasDev



timothyspann

<https://www.pulsardeveloper.com/>



Pulsar Summit Europe

October 6, 2021

Pulsar Summit Asia

November 20-21, 2021

Contact us at partners@pulsar-summit.org to become a sponsor or partner

Announcing

Flink SQL on StreamNative Cloud



Stream
Native
Cloud



