



**Stream
Native**

FLiP Into Trino

Tim Spann | Developer Advocate

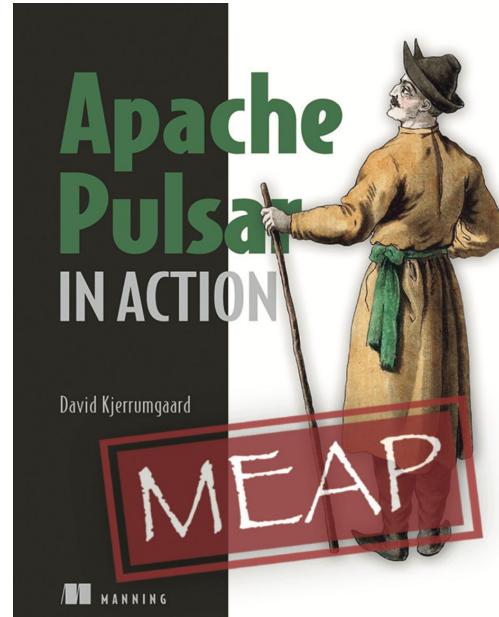
David Kjerrumgaard | Developer Advocate



Founded by the original developers of Apache Pulsar and Apache BookKeeper, StreamNative builds a cloud-native event streaming platform that enables enterprises to easily access data as real-time event streams.

Speaker Bio

David Kjerrumgaard
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>



Tim Spann, Developer Advocate

DZone Zone Leader and Big Data MVB Data DJay

FLiP(N) into Trino Stack

- Apache **Flink**
- Apache **Pulsar**
- StreamNative's Flink Connector for Pulsar
- Apache **NiFi**
- Trino

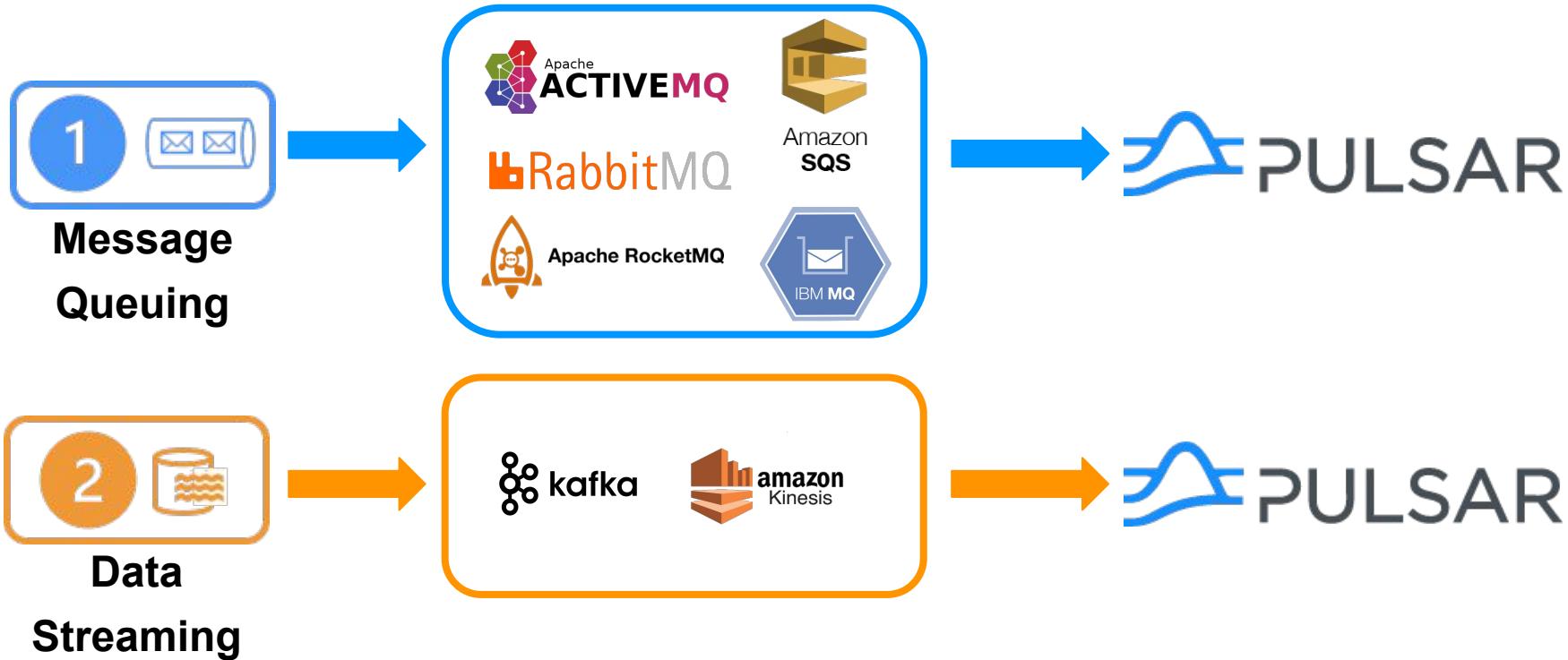


Apache Pulsar



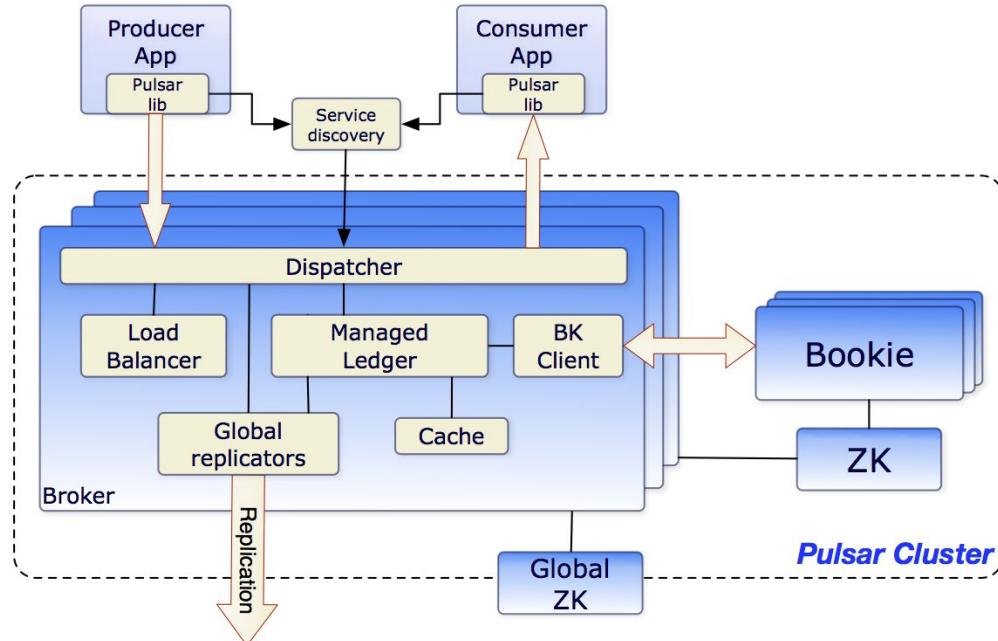
Apache  **PULSAR** is an open source, cloud-native distributed messaging and streaming platform.

A Unified Messaging Platform



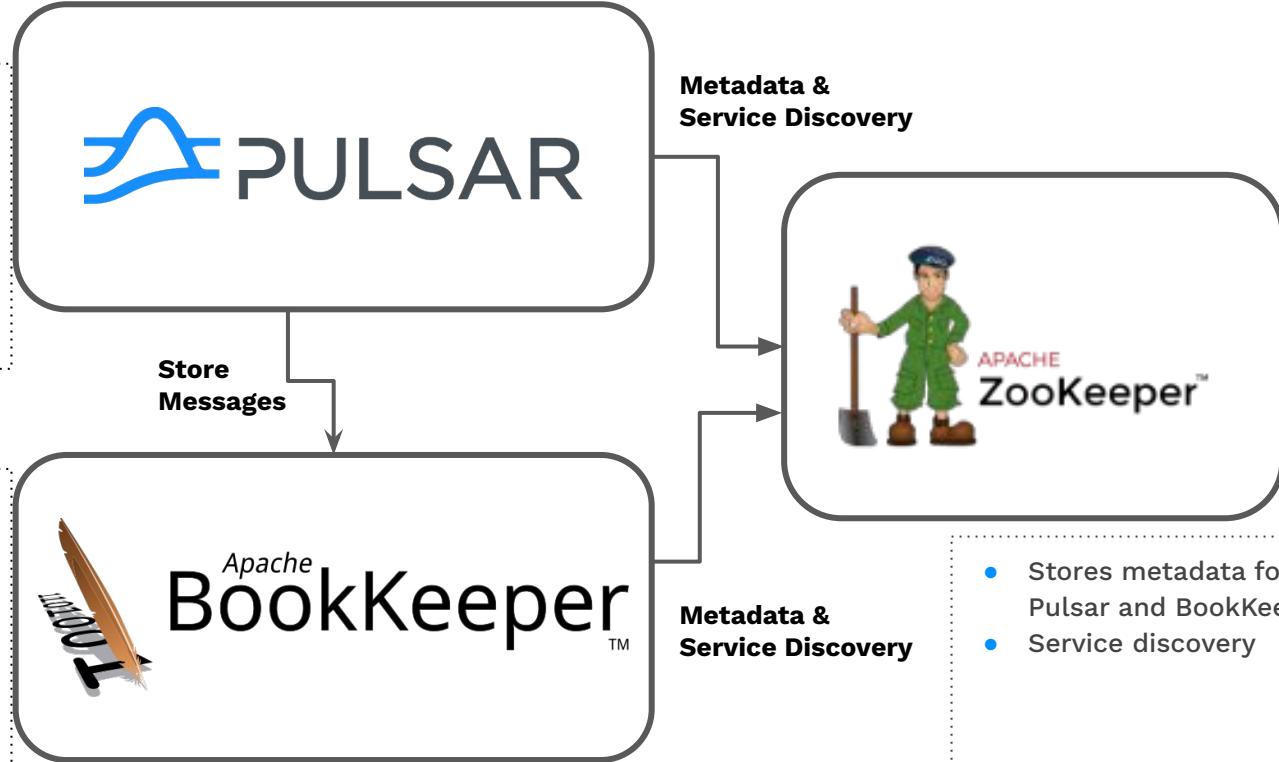
Apache Pulsar

- Pub-Sub
- Geo-Replication
- Pulsar Functions
- Horizontal Scalability
- Multi-tenancy
- Tiered Persistent Storage
- Pulsar Connectors
- REST API
- CLI
- Many clients available
- Four Different Subscription Types
- Multi-Protocol Support
 - **MQTT**
 - AMQP
 - JMS
 - **Kafka**
 - ...



Pulsar Cluster

- “Brokers”
- Handles message routing and connections
- Stateless, but with caches
- Automatic load-balancing
- Topics are composed of multiple segments



- “Bookies”
- Stores messages and cursors
- Messages are grouped in segments/ledgers
- A group of bookies form an “ensemble” to store a ledger

- Metadata & Service Discovery

- Stores metadata for both Pulsar and BookKeeper
- Service discovery

Apache Pulsar - Example Sinks



- mongoDB



- AWS Lambda



- redis



- AWS S3

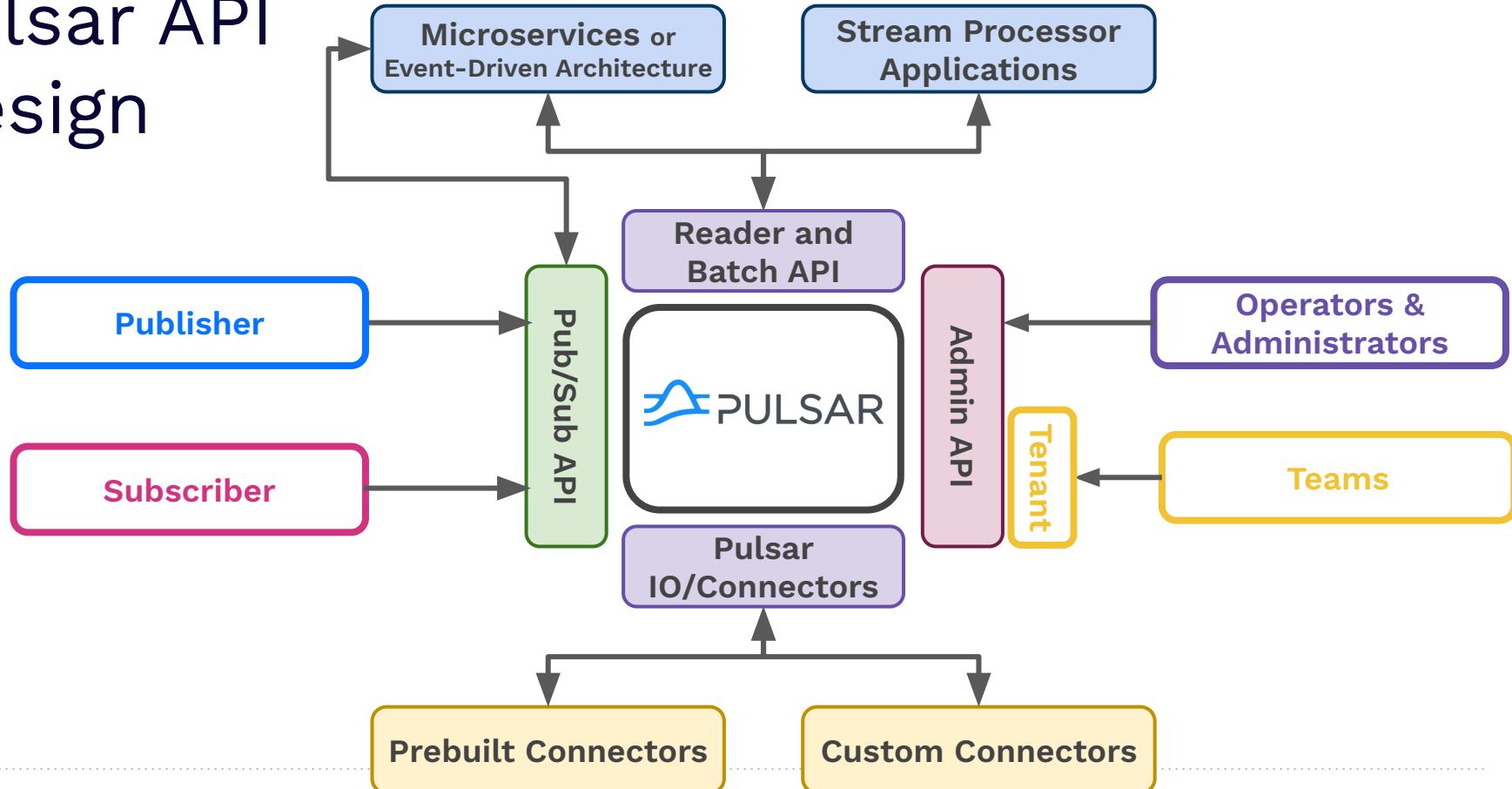


- GCS



<https://hub.streamnative.io/connectors/cloud-storage-sink/2.5.1/>

Pulsar API Design



Subscription Modes

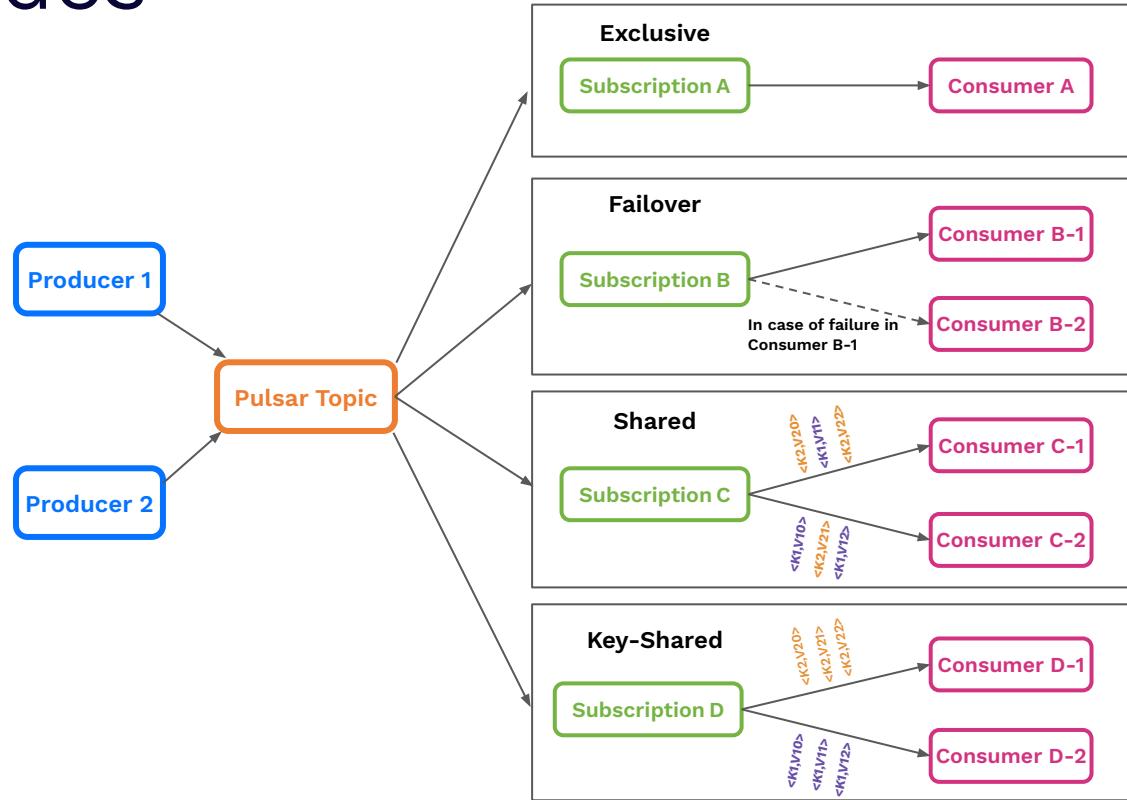
Different subscription modes have different semantics:

Exclusive/Failover -

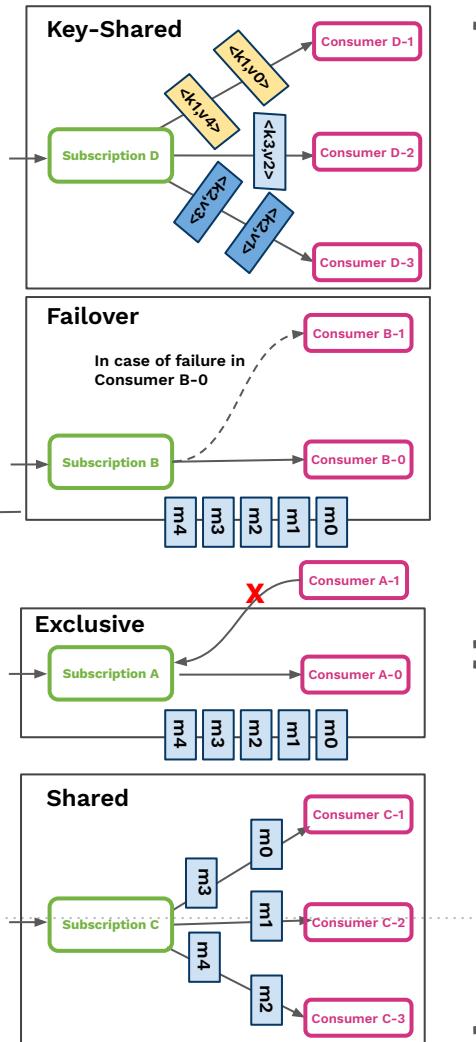
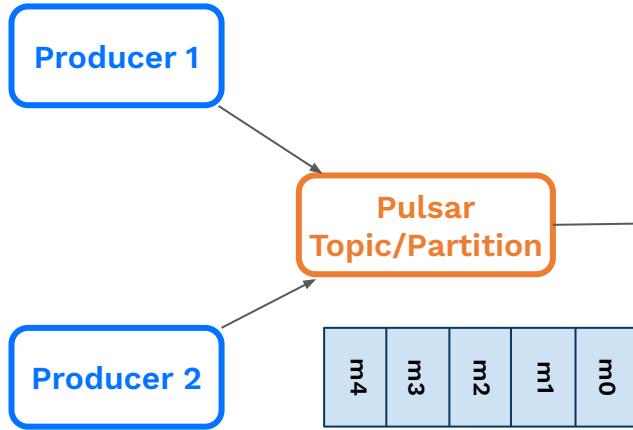
guaranteed order, single active consumer

Shared - multiple active consumers, no order

Key_Shared - multiple active consumers, order for given key



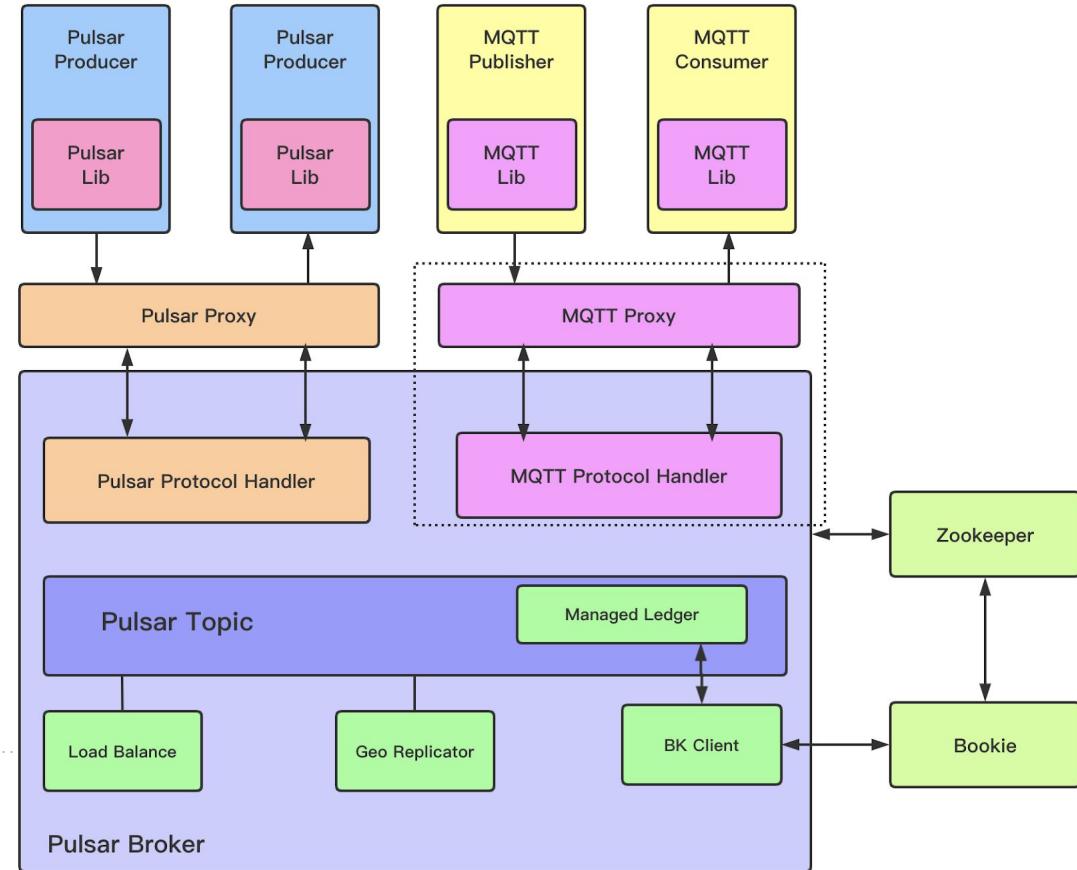
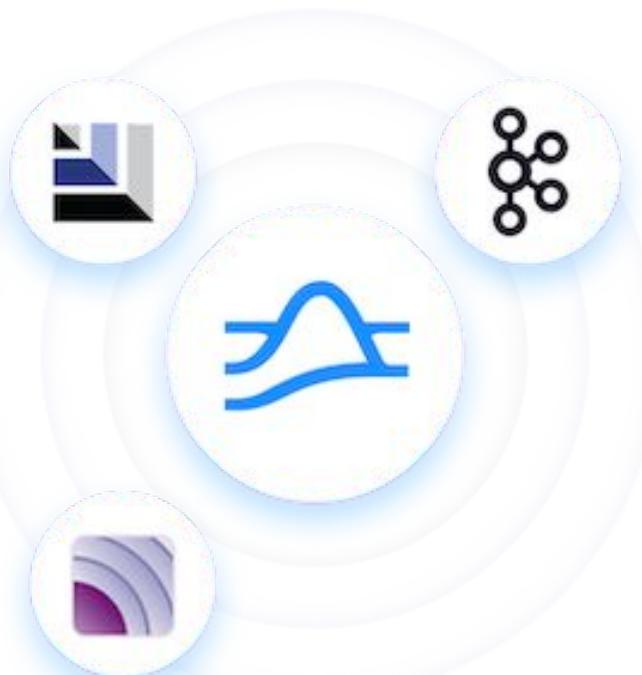
Unified Messaging Model



Streaming

Messaging

MQTT on Pulsar (MoP)



MQTT on Pulsar (MoP) Configuration

```
messagingProtocols= mqtt
```

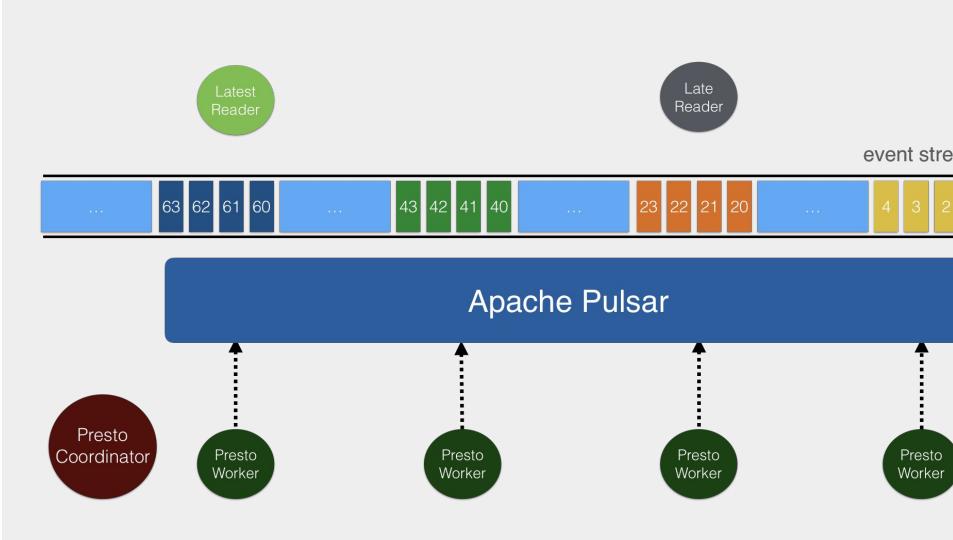
```
# directory  
protocolHandlerDirectory=./protocols
```

```
#mqtt 3.1.1 - port / ip  
mqttListeners=mqtt://127.0.0.1:1883  
advertisedAddress=127.0.0.1
```

Pulsar SQL

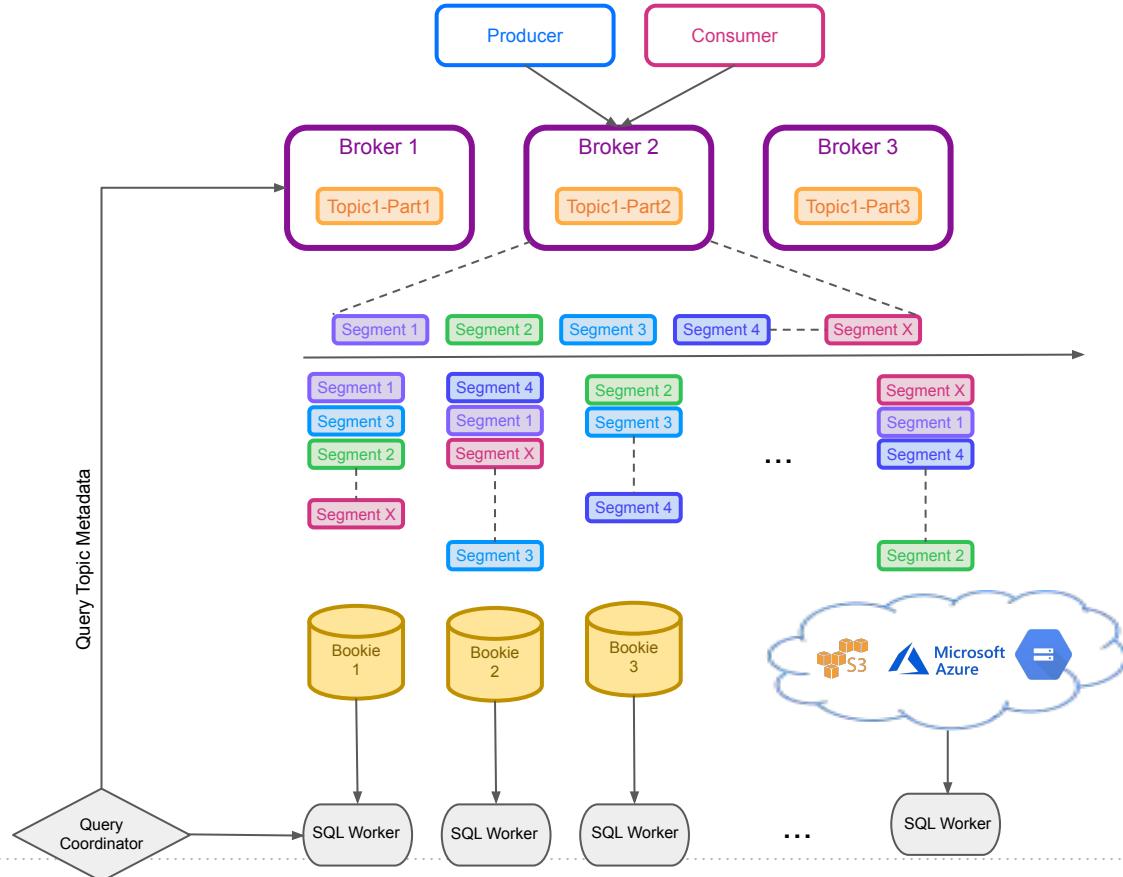
Using Trino for querying Pulsar topic data.

<https://pulsar.apache.org/docs/en/sql-overview/>

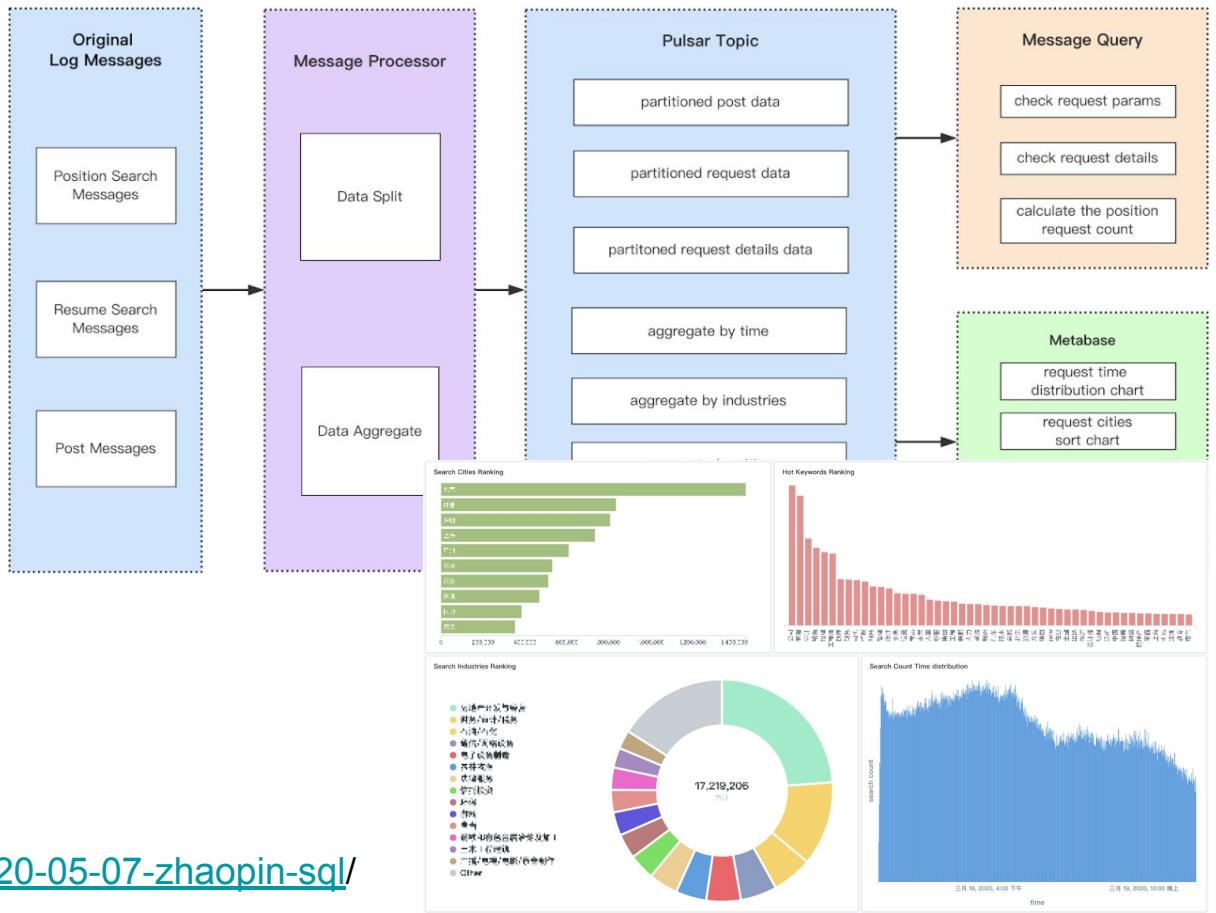


Pulsar SQL

Presto/Trino workers can read segments directly from bookies (or offloaded storage) in parallel.



Pulsar SQL



<https://streamnative.io/blog/case/2020-05-07-zhaopin-sql/>

Query Your Topics with Pulsar SQL (Trino)

```
presto> select camera, cpu, cputempf, gputempf, memory, top1, top1pct, uuid, __publish_time__, __message_id__, __key__ from pulsar."public/default".iotjetsonjson;
      camera |   cpu | cputempf | gputempf |   memory |      top1 |    top1pct |        uuid | __publish_time__ | __message_id__ | __key__
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
/dev/video0 |  8.7 |   82 |  82 |  33.5 | microphone, mike | 18.85986328125 | xav_uuid_video0_lgl_20211001183019 | 2021-10-01 14:30:30.657 | (564,3,0) |
/dev/video0 |  8.7 |   82 |  82 |  33.6 | microphone, mike | 19.22607421875 | xav_uuid_video0_kpt_20211001183033 | 2021-10-01 14:30:44.380 | (564,4,0) |
/dev/video0 | 12.0 |   80 |  81 |  33.5 | microphone, mike | 12.53662109375 | xav_uuid_video0_gzd_20211001182930 | 2021-10-01 14:29:48.756 | (564,0,0) |
/dev/video0 |  8.5 |   82 |  82 |  33.6 | microphone, mike | 14.0625 | xav_uuid_video0_wlw_20211001182951 | 2021-10-01 14:30:02.919 | (564,1,0) |
/dev/video0 |  8.5 |   82 |  82 |  33.5 | microphone, mike | 29.8828125 | xav_uuid_video0_ulq_20211001183005 | 2021-10-01 14:30:16.787 | (564,2,0)
5 rows
[END]
```

```
presto> show tables in pulsar."public/default";
      Table
-----
generator_test
iotjetsonjson
mqtt-2
(3 rows)

Query 20211001_054538_00008_s8x23, FINISHED, 1 node
Splits: 19 total, 19 done (100.00%)
0:00 [3 rows, 105B] [14 rows/s, 493B/s]
```

Running a Pulsar SQL Query

- To run a query, you need to start Pulsar SQL with:
 \$ pulsar sql
- All queries must:
 - Be terminated with a ;
 - Use single quotes (') for strings
 - If you run a query with many results, Pulsar SQL will show a list
- Exit out by typing q
 - Scroll through results with the up and down arrows or page up and page down keys
- Queries can be run using Presto's/Trino's REST API
 - Query results are returned as JSON

Viewing Topics with Pulsar SQL

- Show available namespaces

```
SHOW schemas IN pulsar;
```

- Show topics in a namespace

```
SHOW tables IN pulsar."public/default";
```

- Show schema in a topic

```
SHOW columns IN pulsar."public/default".mytopic;
```

Supported SQL Syntax

```
SELECT card, suit FROM cards;
```

```
SELECT * FROM cards WHERE suit = "Spade";
```

```
SELECT * FROM cards WHERE card LIKE "1%";
```

```
SELECT * FROM cards WHERE suit = "Spade" AND card = "1";
```

```
SELECT * FROM cards LIMIT 10;
```

```
SELECT * FROM cards WHERE suit = "Spade" LIMIT 10;
```

```
SELECT suit, COUNT(card) FROM cards GROUP BY suit;
```

```
SELECT suit, card FROM cards ORDER BY suit, card;
```

Defining Schemas

To execute a query, Pulsar SQL needs to know the schema.

- Schemas are accessible from the Broker and stored in BookKeeper.
- Pulsar SQL needs to know:
 - Name of the column
 - Type of the column
 - Nullability of the column
- Pulsar SQL currently supports Avro and JSON for automatic schema detection.

Use cases for Pulsar SQL

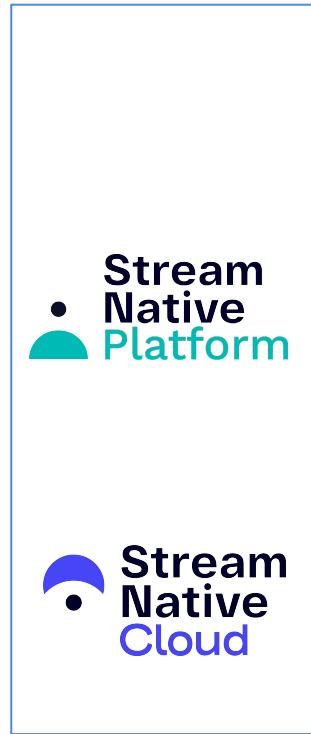
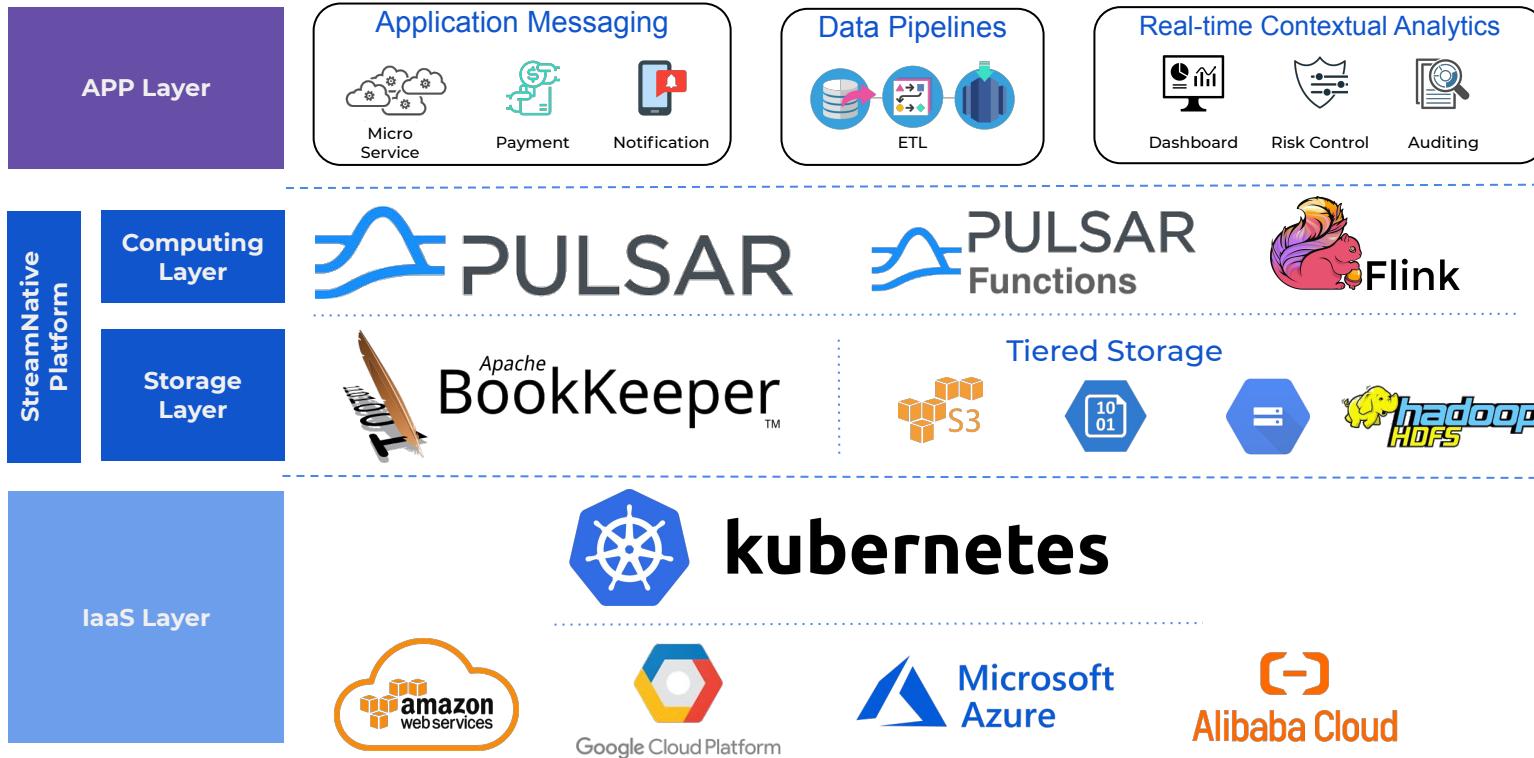
- Pulsar SQL is a useful tool for answering questions about data in your streams, such as basic analytics or searching for specific data.
- Pulsar SQL is not intended for high throughput queries or for running “continuous” queries that update as new records are added.

<https://pulsar.apache.org/docs/en/sql-rest-api/>

StreamNative Cloud



StreamNative Solution





A cloud-native, real-time messaging and streaming platform to support multi-cloud and hybrid cloud strategies.

**Powered
by Pulsar**



**Cloud
Native**



**Built for
Containers**



Flink SQL



The unified messaging and streaming platform made by the creators of Apache Pulsar.

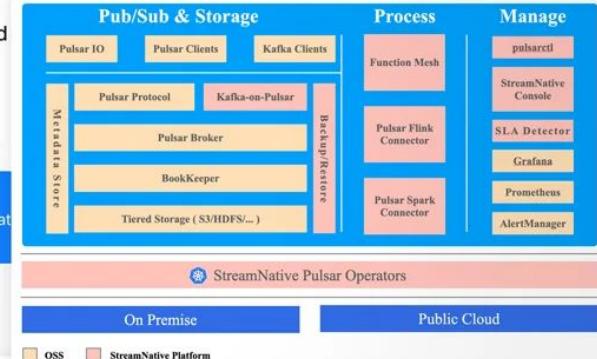
Built for Kubernetes. Made for the cloud. Enables multi-cloud and hybrid

Take A Tour

Contact Sales

Product Update

StreamNative Cloud on AWS Marketplace. Leverage Pulsar on the largest cloud provider with StreamNative



StreamNative, Powered by Apache Pulsar

StreamNative
Cloud

Apache Pulsar as a service, StreamNative Cloud delivers a resilient and scalable messaging and event streaming service deployable in minutes.

StreamNative
Platform

StreamNative Platform is a cloud-native messaging and event streaming platform built by the original creators for Apache Pulsar.

StreamNative
Pro Services

Accelerate your messaging and streaming platform development and drive business results with help from StreamNative's Pulsar experts.

InfoWorld

BOSSIE
2021 AWARDS



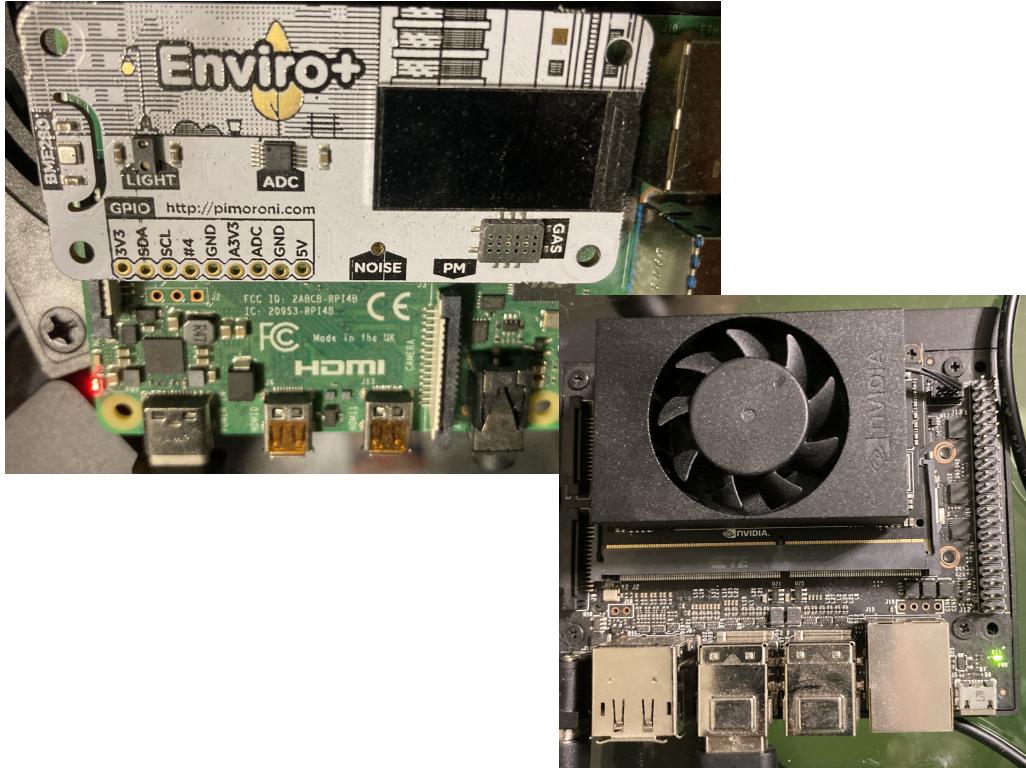
StreamNative

Use Cases

USE CASE

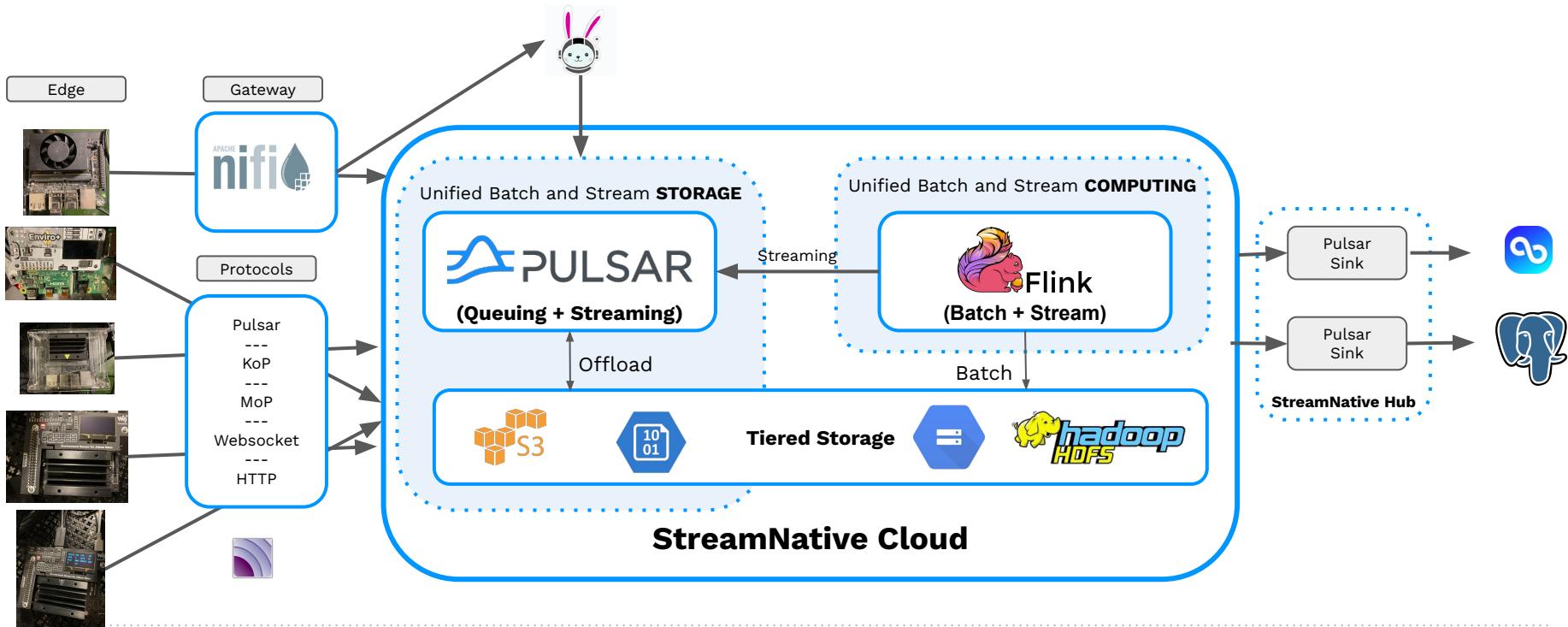
IoT Ingestion: High-volume streaming sources, sensors, multiple message formats, diverse protocols and multi-vendor devices creates data ingestion challenges.

Other Sources: Transit data, news, twitter, status feeds, REST data, stock data and more.



End-to-End Streaming FLiPN Edge AI Application

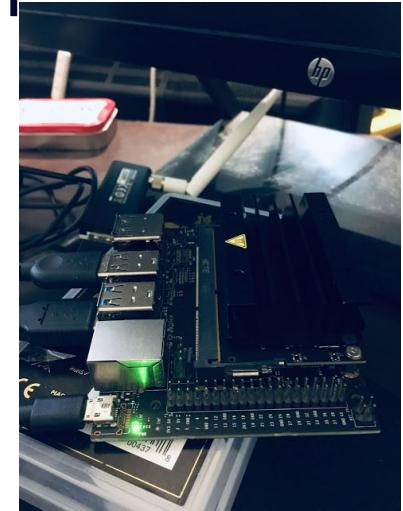
Apache Flink - Apache Pulsar - Apache NiFi <-> Trino



Using NVIDIA Jetson Devices With Pulsar

DEMO TIME

<https://github.com/tspannhw/minifi-xaviernx/>
<https://github.com/tspannhw/minifi-jetson-nano>
<https://github.com/tspannhw/Flip-iot>
<https://github.com/tspannhw/FLiP-EdgeAI>
<https://github.com/tspannhw/FLiP-CloudIngest>
<https://github.com/tspannhw/FLiP-Transit>
<https://github.com/tspannhw/FLiP-Jetson>
<https://www.datainmotion.dev/2020/10/flank-streaming-edgeai-on-new-nvidia.html>



Show Me More Data

The screenshot shows a StreamNative Cloud interface with a sidebar containing navigation links such as International, sndev, sgconnector, Tenants, Namespaces, Topics, SQL, Clients, Connector, Manage, Service Accounts, Flink Clusters, Pulsar Clusters, and Instance Setting. The main area displays a log stream with five data rows, each with a timestamp and a detailed log entry.

Timestamp	Log Entry (JSON)
46:20	{"uuid": "xav_uuid_video0_uul_20200921211433", "camera": "/dev/video0", "ipaddress": "192.168.1.78", "networktime": "25.09616088671875, "top1pc": 18.29833984375, "top1": "desk", "cputemp": "30.5", "gputemp": "31.0", "gputempf": "88", "cputempf": "87", "runtime": "6", "host": "nvidia-desktop", "filename": "/home/nvidia/nvme/images/out_video0_x5s_20200921211433.jpg", "imageinput": "/home/nvidia/nvme/images/img_video0_nzj_20200921211433.jpg", "host_name": "nvidia-desktop", "macaddress": "70:66:55:15:b4:a5", "end": "1600722879.891937", "te": "6.490077018737793", "systemtime": "09/21/2020 17:14:39", "cpu": "48.0", "diskusage": "42145.1 MB", "memory": 45.9, "id": "video0_20200921211433_ff3757d5-52eb-4a8a-8b40-5da2cf3edca"}, {"uuid": "xav_uuid_video0_cg2_20200921211450", "camera": "/dev/video0", "ipaddress": "192.168.1.78", "networktime": "29.682527542114258, "top1pc": 33.203125, "top1": "ski", "cputemp": "31.0", "gputemp": "31.0", "gputempf": "88", "cputempf": "88", "runtime": "8", "host": "nvidia-desktop", "filename": "/home/nvidia/nvme/images/out_video0_zsz_20200921211450.jpg", "imageinput": "/home/nvidia/nvme/images/img_video0_mgi_20200921211450.jpg", "host_name": "nvidia-desktop", "macaddress": "70:66:55:15:b4:a5", "end": "1600722898.9325902", "te": "8.286669492721558", "systemtime": "09/21/2020 17:14:58", "cpu": "45.3", "diskusage": "42145.1 MB", "memory": 46.0, "id": "video0_20200921211450_b5c396ad-82f4-8aa-acb9-9caeaf42855"}, {"uuid": "xav_uuid_video0_prb_20200921211601", "camera": "/dev/video0", "ipaddress": "192.168.1.78", "networktime": "6.29107891784668, "top1pc": 16.02783023125, "top1": "desk", "cputemp": "32.0", "gputemp": "32.5", "gputempf": "90", "cputempf": "90", "runtime": "5", "host": "nvidia-desktop", "filename": "/home/nvidia/nvme/images/out_video0_give_20200921211601.jpg", "imageinput": "/home/nvidia/nvme/images/img_video0_oon_20200921211601.jpg", "host_name": "nvidia-desktop", "macaddress": "70:66:55:15:b4:a5", "end": "1600722966.4952142", "te": "5.116875648498535", "systemtime": "09/21/2020 17:16:06", "cpu": "44.5", "diskusage": "42145.1 MB", "memory": 46.3, "id": "video0_20200921211601_935b5e90-4bcd-47dc-84eb-f85846fae190"}, {"uuid": "xav_uuid_video0_zwh_20210804181811", "camera": "/dev/video0", "ipaddress": "192.168.1.214", "networktime": "48.33692932128906, "top1pc": 40.91796875, "top1": "Model T", "cputemp": "41.5", "gputemp": "42.0", "gputempf": "108", "cputempf": "107", "runtime": "11", "host": "nvidia-desktop", "filename": "/home/nvidia/nvme/images/out_video0_tes_20210804181811.jpg", "imageinput": "/home/nvidia/nvme/images/img_video0_prs_20210804181811.jpg", "host_name": "nvidia-desktop", "macaddress": "70:66:55:15:b4:a5", "end": "1628101102.2976809", "te": "10.955556154251099", "systemtime": "08/04/2021 14:18:22", "cpu": "12.0", "diskusage": "33277.7 MB", "memory": 33.5, "id": "video0_20210804181811_67e85abc-9085-41c3-b27f-eb7f7aa36daa"}, {"uuid": "xav_uuid_video0_zwh_20210804181811", "camera": "/dev/video0", "ipaddress": "192.168.1.214", "networktime": "48.33692932128906, "top1pc": 40.91796875, "top1": "Model T", "cputemp": "41.5", "gputemp": "42.0", "gputempf": "108", "cputempf": "107", "runtime": "11", "host": "nvidia-desktop", "filename": "/home/nvidia/nvme/images/out_video0_tes_20210804181811.jpg", "imageinput": "/home/nvidia/nvme/images/img_video0_prs_20210804181811.jpg", "host_name": "nvidia-desktop", "macaddress": "70:66:55:15:b4:a5", "end": "1628101102.2976809", "te": "10.955556154251099", "systemtime": "08/04/2021 14:18:22", "cpu": "12.0", "diskusage": "33277.7 MB", "memory": 33.5, "id": "video0_20210804181811_67e85abc-9085-41c3-b27f-eb7f7aa36daa"}]

Ingesting IoT Data via Java Pulsar

```
UUID uuidKey = UUID.randomUUID();
String pulsarKey = uuidKey.toString();
String OS = System.getProperty("os.name").toLowerCase();
String message = "" + jct.message;
IoTMessage iotMessage = parseMessage("") + jct.message);
String topic = DEFAULT_TOPIC;
if ( jct.topic != null && jct.topic.trim().length()>0) {
    topic = jct.topic.trim();
}
ProducerBuilder<IoTMessage> producerBuilder = client.newProducer(JSONSchema.of(IoTMessage.class))
    .topic(topic)
    .producerName("jetson")
    .sendTimeout(5, TimeUnit.SECONDS);

Producer<IoTMessage> producer = producerBuilder.create();

MessageId msgID = producer.newMessage()
    .key(iotMessage.getUuid())
    .value(iotMessage)
    .send();
```

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

Ingesting IoT Data via Java Pulsar

```
private static IoTMessage parseMessage(String message) {  
    IoTMessage iotMessage = null;  
  
    try {  
        if ( message != null && message.trim().length() > 0 ) {  
            ObjectMapper mapper = new ObjectMapper();  
            iotMessage = mapper.readValue(message, IoTMessage.class);  
            mapper = null;  
        }  
    }  
    catch(Throwable t) {  
        t.printStackTrace();  
    }  
  
    if (iotMessage == null) {  
        iotMessage = new IoTMessage();  
    }  
    return iotMessage;  
}
```

FLiP Into Trino with Apache NiFi

Apache NiFi to JDBC Sink

<https://docs.starburst.io/data-consumer/clients/jdbc.html>

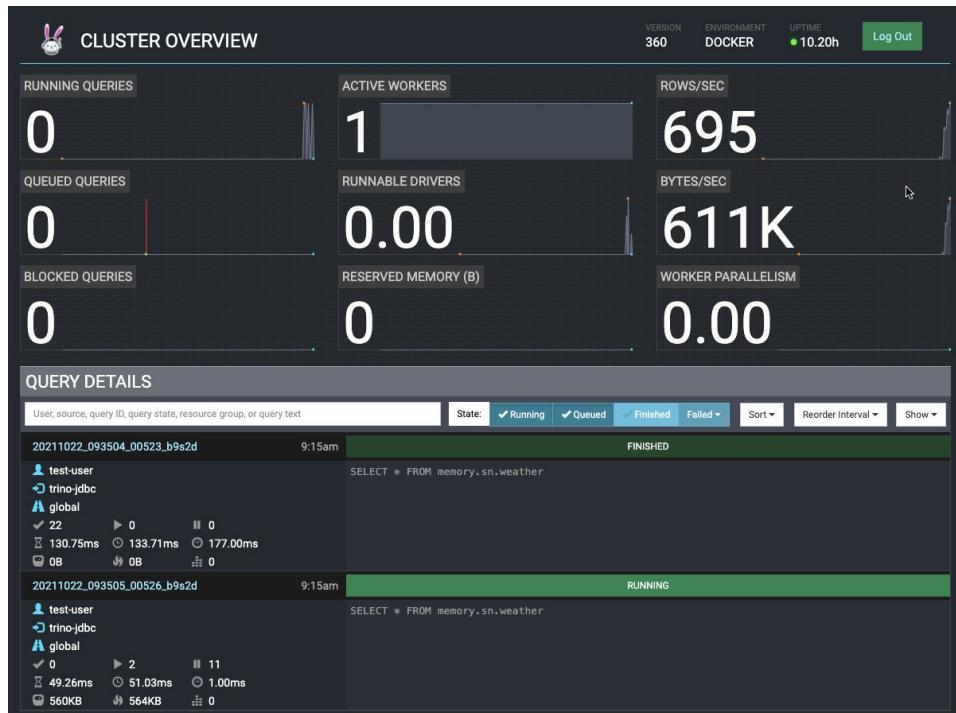
<https://docs.starburst.io/data-consumer/clients/dbeaver.html>

<https://hub.docker.com/r/trinodb/trino>

<https://trino.io/download.html>

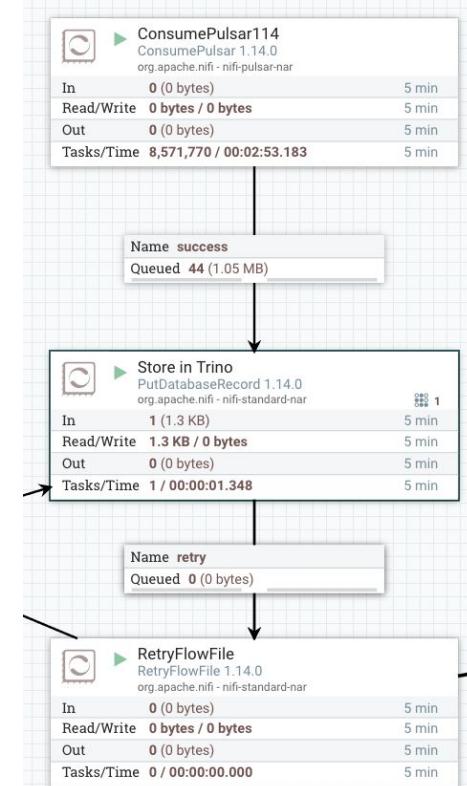
```
docker run -p 8080:8080 --name trino trinodb/trino
```

FLiP Into Trino with Apache NiFi



FLiP Into Trino with Apache NiFi

Record Reader	CDP Infer JsonTreeReader
Database Type	Generic
Statement Type	INSERT
Data Record Path	No value set
Database Connection Pooling Service	Trino JDBC
Catalog Name	memory
Schema Name	sn
Table Name	weather
Translate Field Names	true
Database Connection URL	jdbc:trino://timothys-mbp:8080
Database Driver Class Name	io.trino.jdbc.TrinoDriver
Database Driver Location(s)	/home/nvidia/nvme/trino-jdbc-363.jar
Kerberos Credentials Service	No value set
Kerberos Principal	No value set
Kerberos Password	No value set
Database User	test-user



Deeper Content

- <https://www.datainmotion.dev/2020/10/running-flink-sql-against-kafka-using.html>
- <https://www.datainmotion.dev/2020/10/top-25-use-cases-of-cloudera-flow.html>
- <https://github.com/tspannhw/EverythingApacheNiFi>
- <https://github.com/tspannhw/CloudDemo2021>
- <https://github.com/tspannhw/FLiP-Into-Trino>
- <https://github.com/tspannhw/StreamingSQLExamples>
- <https://www.linkedin.com/pulse/2021-schedule-tim-spann/>
- <https://github.com/tspannhw/StreamingSQLExamples/blob/8d02e62260e82b027b43abb911b5c366a3081927/README.md>
- <https://www.pulsardeveloper.com/>
- <https://streamnative.io/success-story/zhaopin-pulsar-sql/>

Let's Keep in Touch!



Tim Spann

Developer Advocate



@[PassDev](#)



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



<https://github.com/tspannhw>

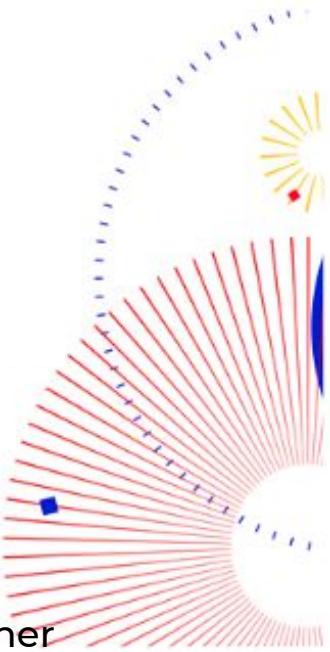
Connect with the Community & Stay Up-To-Date

- Join the Pulsar Slack channel -
Apache-Pulsar.slack.com
- Follow [@streamnativeio](https://twitter.com/streamnativeio) and [@apache_pulsar](https://twitter.com/apache_pulsar) on Twitter
- [Subscribe](#) to Monthly Pulsar Newsletter for major news, events, project updates, and resources in the Pulsar community



Pulsar Summit Asia

November 20-21, 2021



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

Interested In Learning More?



Resources

[Flink SQL Cookbook](#)

[The Github Source for Flink SQL Demo](#)

[The GitHub Source for Demo](#)



Free eBooks

[Manning's Apache Pulsar in Action](#)

[O'Reilly Book](#)



Upcoming Events

[\[10/21\] Trino Summit](#)

Questions