CLOUDERA

SF 23 Real-Time Analytics SUMMIT

Building a Real-Time IoT Application

Tim Spann
Principal Developer Advocate

26-April-2023

RTA Summit

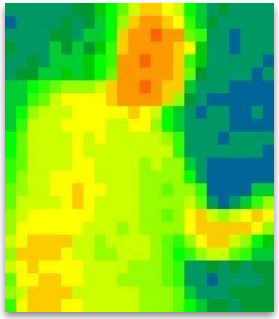
Notes

We will walk step-by-step with live code and demos on how to build a real-time IoT application with Pinot + Pulsar. First, we stream sensor data from an edge device monitoring location conditions to Pulsar via a Python application. We have our Apache Pinot “realtime” table connected to Pulsar via the pinot-pulsar stream ingestion connector. Our data streams into the stream, and we visualize it with Superset.

https://medium.com/@tspann/building-a-real-time-iot-application-with-apache-pulsar-and-apache-pinot-1e3baf8c1824

https://github.com/tspannhw/pulsar-thermal-pinot

FLiPN-FLaNK Stack



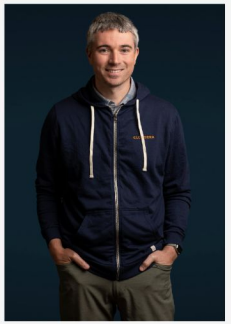
Tim Spann
@PaasDev // Blog: www.datainmotion.dev
Principal Developer Advocate, Cloudera
Princeton Future of Data Meetup.
ex-Pivotal, ex-Hortonworks, ex-StreamNative, ex-PwC
https://github.com/tspannhw/EverythingApacheNiFi
https://medium.com/@tspann
Apache NiFi x Apache Kafka x Apache Flink x Java

FLaNK
flankstack.dev

Hosts

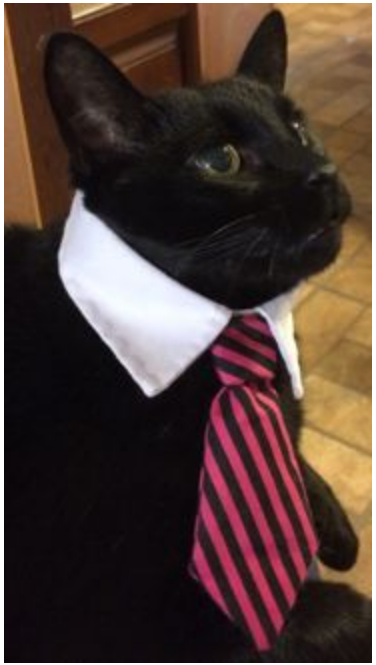
Joe Witt, VP of Engineering, Cloudera

Mark Payne, Principal Engineer, Cloudera

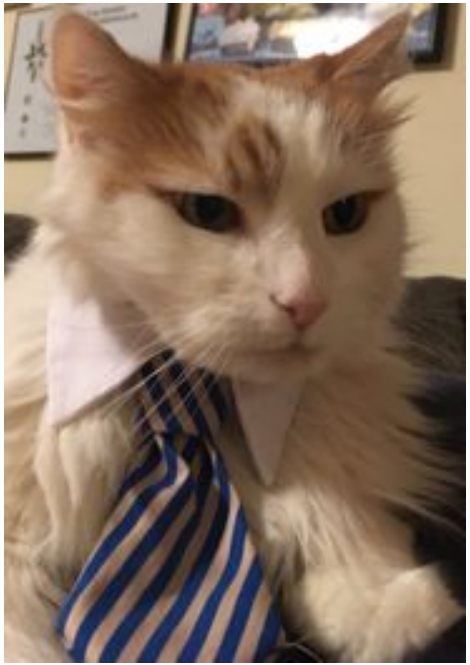


Matt Gilman, Principal Engineer, Cloudera

https://attend.cloudera.com/nificommitters0503



- Introduction to Pinot
- Introduction to Apache Pulsar
- NiFi to Pulsar to Pinot (FLiPN)
- NiFi to Kafka to Pinot (P-FLaNK)
- FLaNK Ingest
- Demos



Assets

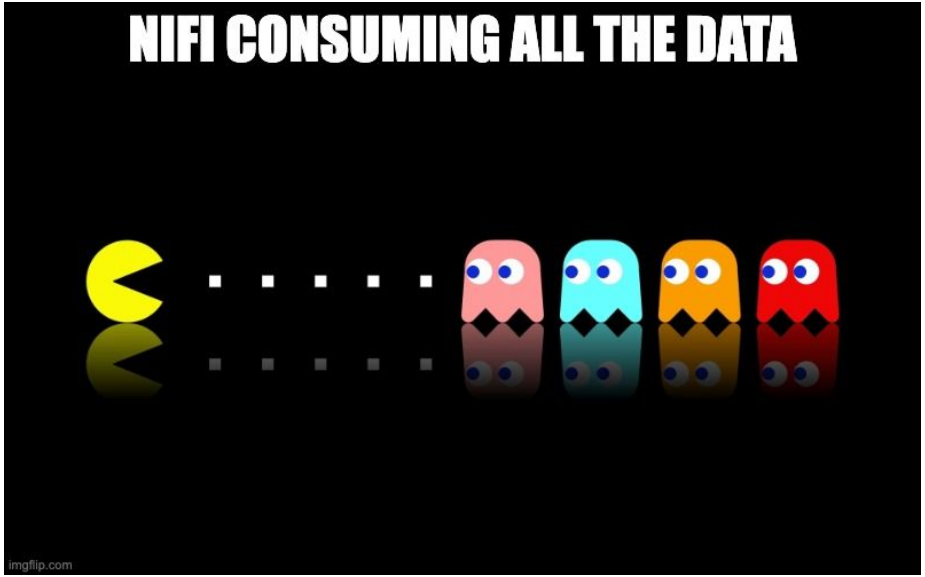
Apache NiFi: Flows

Apache Pinot: Real-Time Tables

Apache Kafka: Topics

Apache Pulsar: Topics

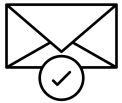
Apache Flink SQL: Virtual Tables



APACHE PULSAR

PULSAR 101

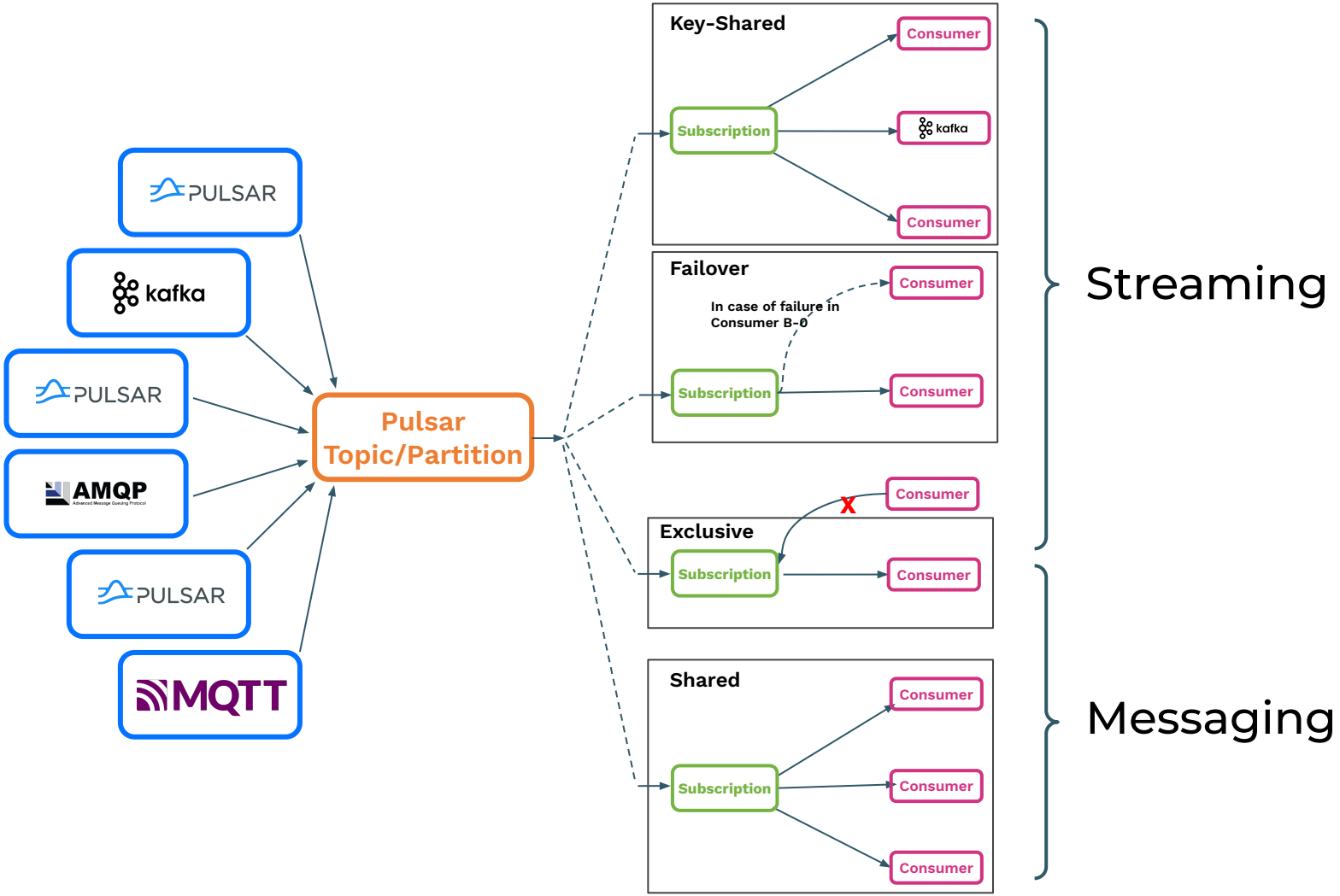
Unified Messaging Platform



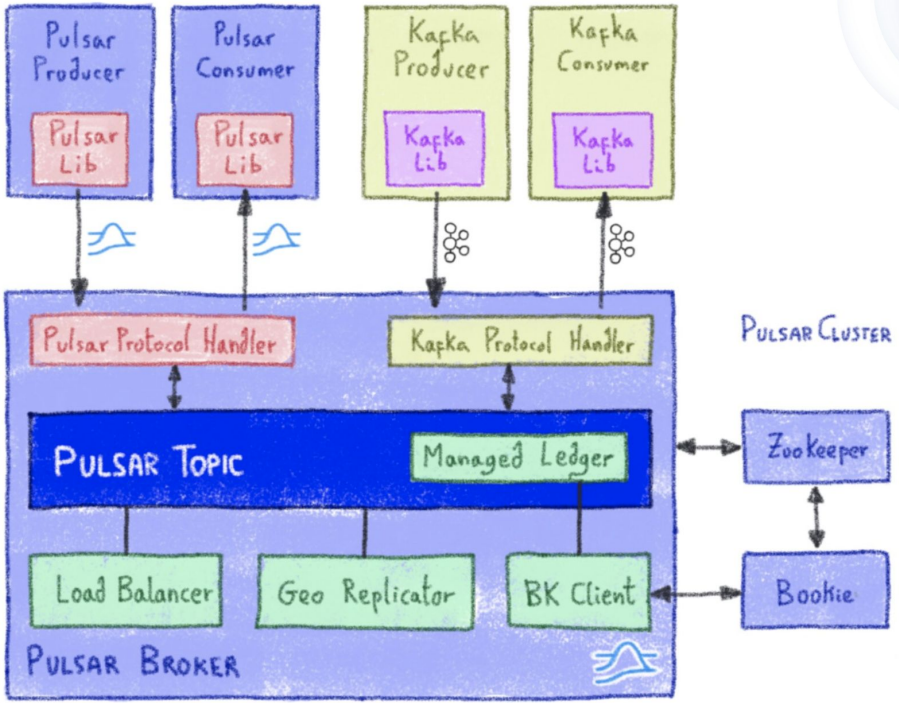
Guaranteed Message Delivery

Resiliency

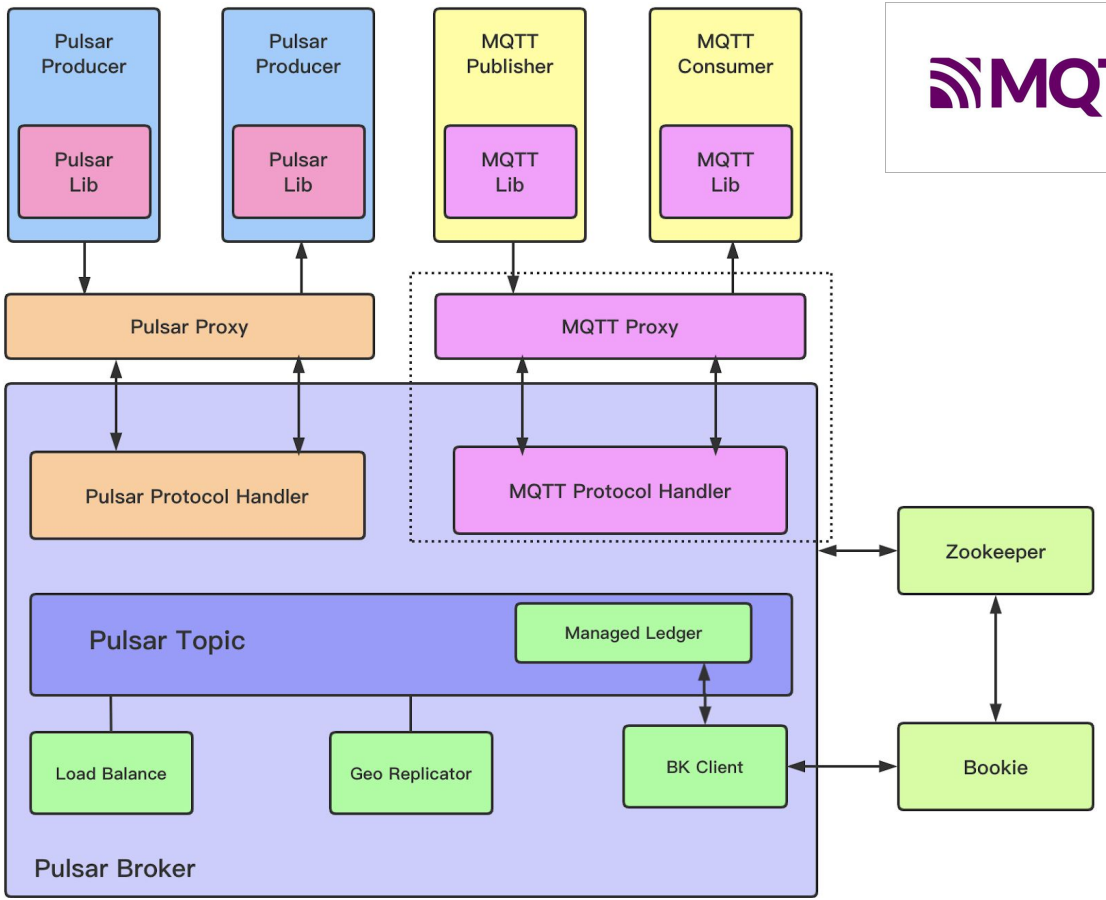
Infinite Scalability


PULSAR
kafka
PULSAR
AMQP
PULSAR
MQTT
Pulsar Topic/Partition
Key-Shared
Subscription
Consumer
kafka
Consumer
Failover
In case of failure in Consumer B-0
Subscription
Consumer
Consumer
Exclusive
Consumer
X
Subscription
Consumer
Shared
Subscription
Consumer
Consumer
Consumer
Streaming
Messaging

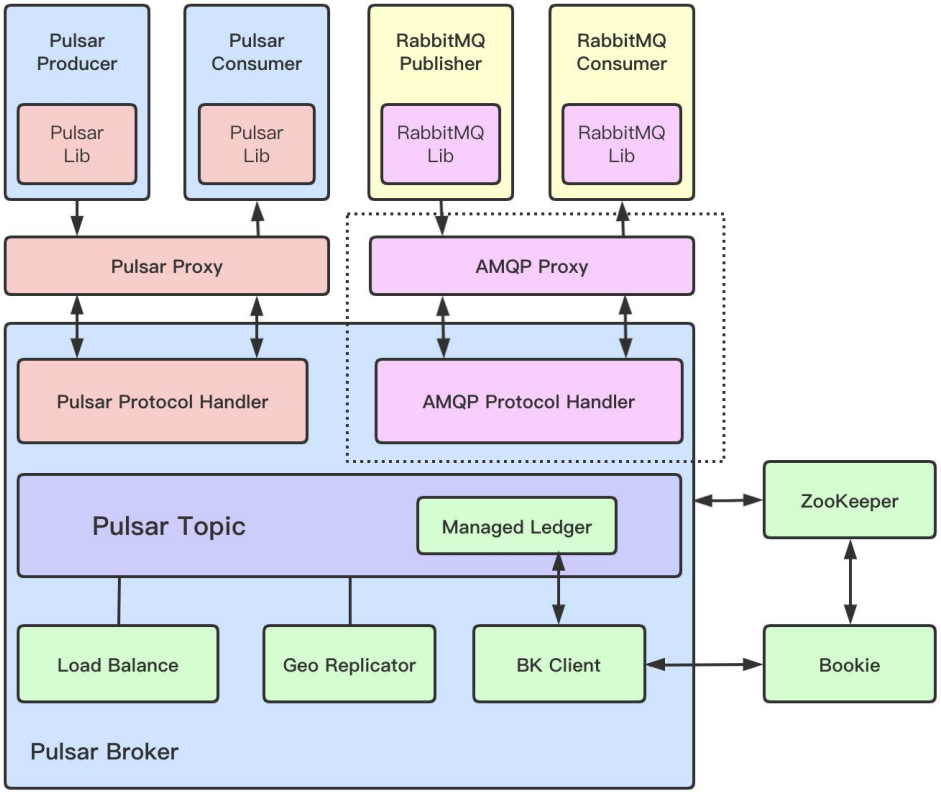
Kafka On Pulsar (KoP)



MQTT On Pulsar (MoP)



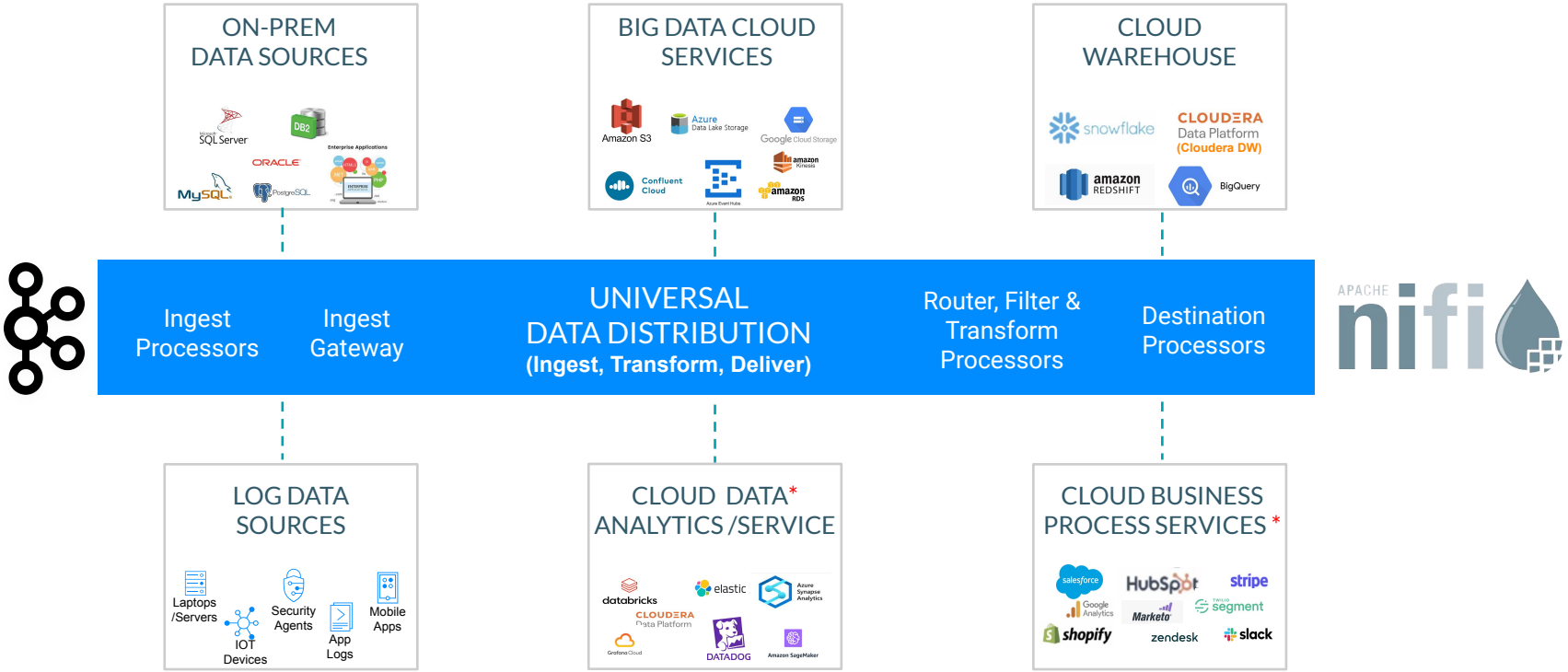
AMQP On Pulsar (AoP)



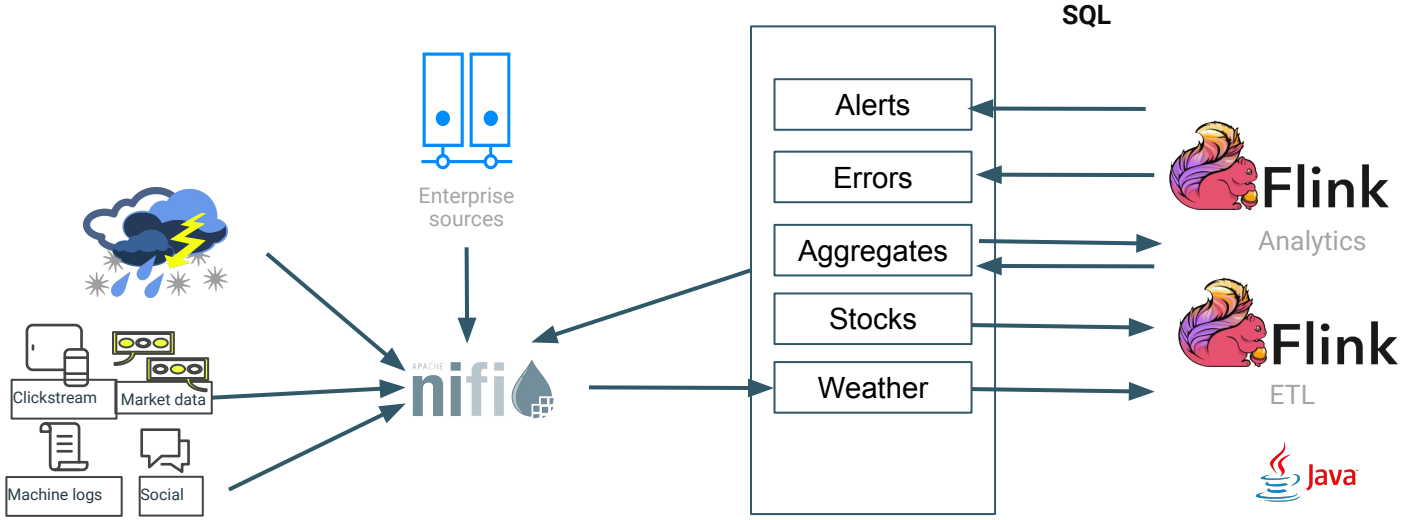
STREAMING

STREAMING FROM … TO .. WHILE ..

Data distribution as a first class citizen



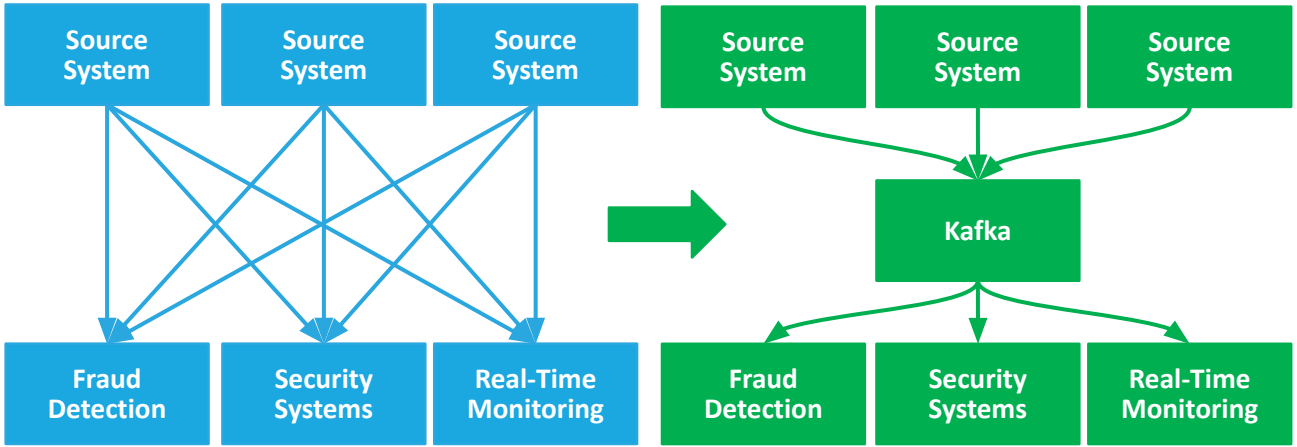
End to End Streaming Pipeline Example



APACHE KAFKA

Apache Kafka

- Highly reliable distributed messaging system
- Decouple applications, enables many-to-many patterns
- Publish-Subscribe semantics
- Horizontal scalability
- Efficient implementation to operate at speed with big data volumes
- Organized by topic to support several use cases



STREAM TEAM

CSP Community Edition

- Kafka, KConnect, SMM, SR, Flink, and SSB in Docker
- Runs in Docker
- Try new features quickly
- Develop applications locally

- Docker compose file of CSP to run from command line w/o any dependencies, including Flink, SQL Stream Builder, Kafka, Kafka Connect, Streams Messaging Manager and Schema Registry
  - `$> docker compose up`
- Licensed under the Cloudera Community License
- **Unsupported**
- Community Group Hub for CSP
- Find it on docs.cloudera.com under Applications

CSP Community Edition

A readily available, dockerized deployment of Apache Kafka and Apache Flink that allows you to test the features and capabilities of Cloudera Stream Processing.

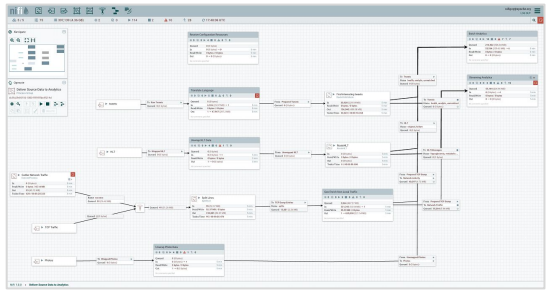
Learn More

DATAFLOW APACHE NIFI

Cloudera Flow and Edge Management

Enable easy ingestion, routing, management and delivery of any data anywhere *(Edge, cloud, data center)* to any downstream system with built in end-to-end security and provenance

minifi APACHE nifi



APACHE NIFI registry

Advanced tooling to industrialize flow development *(Flow Development Life Cycle)*

ACQUIRE

kafka, HIVE, Spark, Storm, Solr, SQL, Flink, DynamoDB, cassandra, amazon KINESIS, elastic, S3, mongoDB, slack

FTP
SFTP
HL7
UDP
XML
HTTP
EMAIL
HTML
IMAGE
SYSLOG

PROCESS

HASH	EVALUATE
MERGE	EXECUTE
EXTRACT	GEOENRICH
DUPLICATE	SCAN
SPLIT	REPLACE
ENCRYPT	TRANSLATE
TALL	CONVERT
ROUTE TEXT	
ROUTE CONTENT	
ROUTE CONTEXT	
ROUTE RATE	
DISTRIBUTE LOAD	

DELIVER

FTP
SFTP
HL7
UDP
XML
HTTP
EMAIL
HTML
IMAGE
SYSLOG

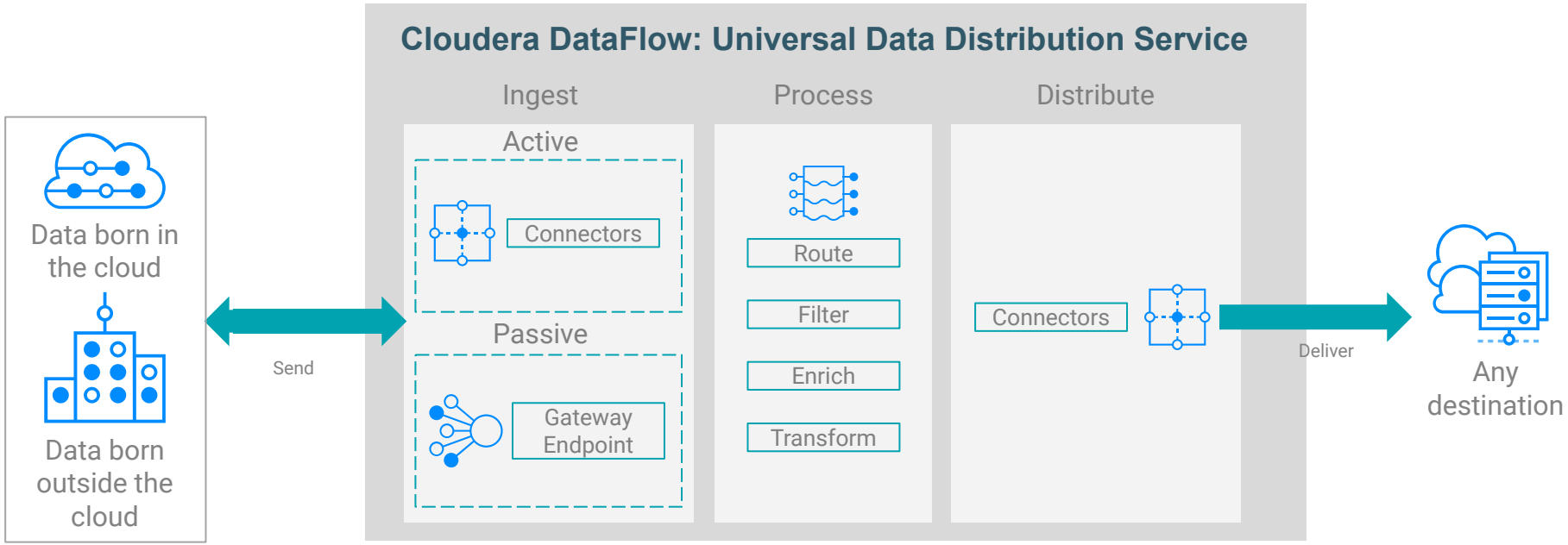
kafka, HIVE, Spark, Storm, Solr, SQL, Flink, DynamoDB, cassandra, amazon KINESIS, elastic, S3, mongoDB, slack

- Over 300 Prebuilt Processors
- Easy to build your own
- Parse, Enrich & Apply Schema
- Filter, Split, Merger & Route
- Throttle & Backpressure

- Guaranteed Delivery
- Full data provenance from acquisition to delivery
- Diverse, Non-Traditional Sources
- Eco-system integration

Universal Data Distribution

Connect to Any Data Source Anywhere then Process and Deliver to Any Destination

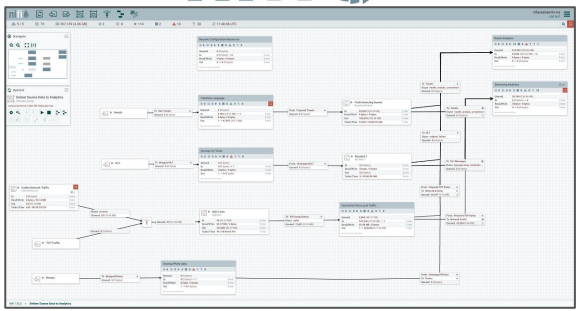


What is Apache NiFi?

Apache NiFi is a scalable, real-time streaming data platform that collects, curates, and analyzes data so customers gain key insights for immediate actionable intelligence.

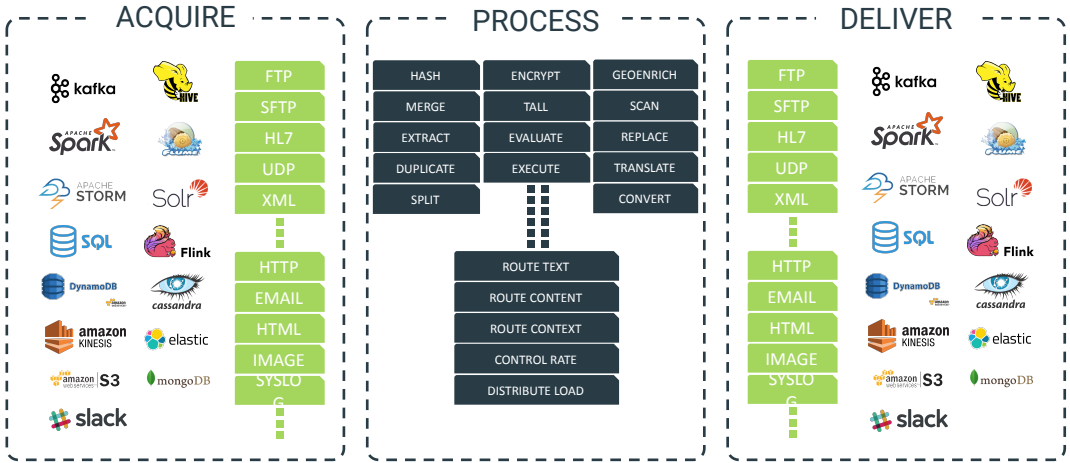
Apache NiFi

Enable easy ingestion, routing, management and delivery of any data anywhere (Edge, cloud, data center) to any downstream system with built in end-to-end security and provenance



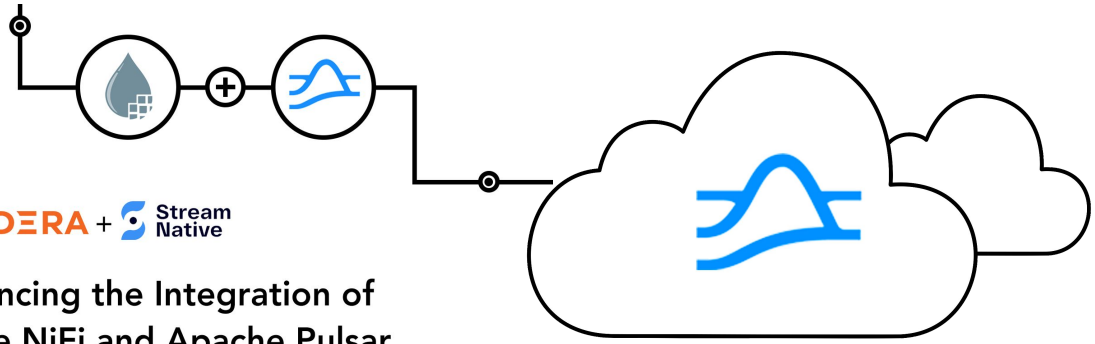
APACHE NIFI registry

Advanced tooling to industrialize flow development (Flow Development Life Cycle)



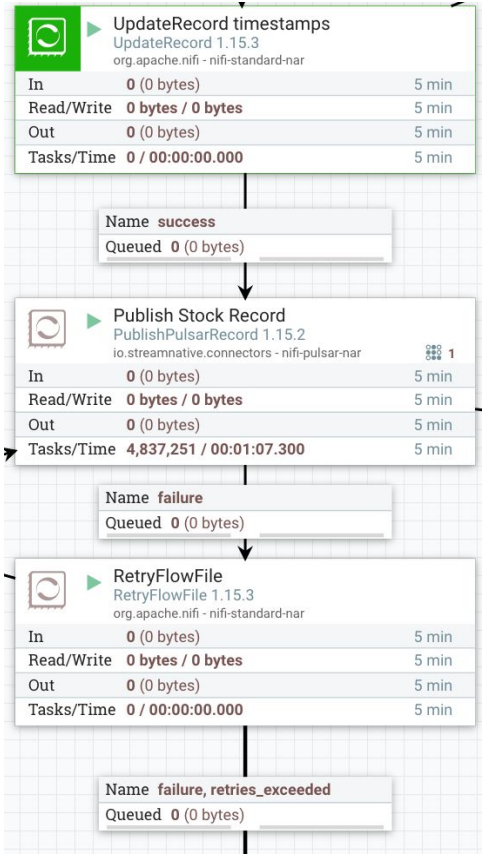
- Over 300 Prebuilt Processors
- Easy to build your own
- Parse, Enrich & Apply Schema
- Filter, Split, Merger & Route
- Throttle & Backpressure
- Guaranteed Delivery
- Full data provenance from acquisition to delivery
- Diverse, Non-Traditional Sources
- Eco-system integration

Apache NiFi Pulsar Connector



CLOUDERA + Stream Native

Announcing the Integration of Apache NiFi and Apache Pulsar

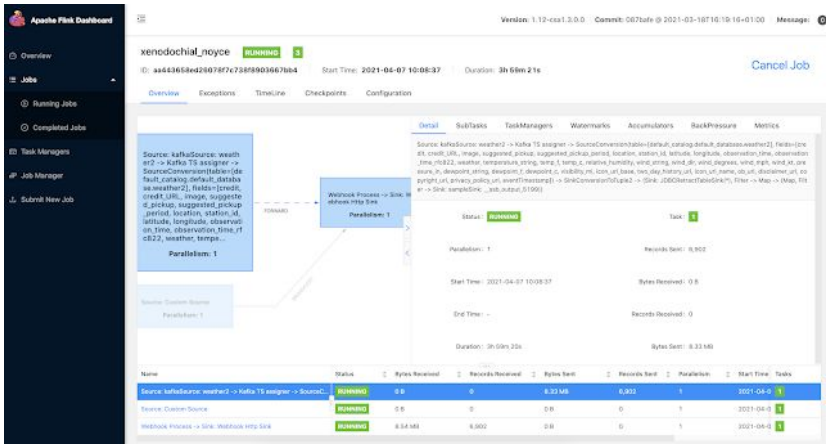


https://streamnative.io/apache-nifi-connector/

APACHE FLINK

Flink SQL

- Streaming Analytics
- Continuous SQL
- Continuous ETL
- Complex Event Processing
- Standard SQL Powered by Apache Calcite



https://www.datainmotion.dev/2021/04/cloudera-sql-stream-builder-ssb-updated.html

Flink SQL

Key Takeaway: Rich SQL grammar with advanced time and aggregation tools

```
-- specify Kafka partition key on output
SELECT foo AS _eventKey FROM sensors

-- use event time timestamp from kafka
-- exactly once compatible
SELECT eventTimestamp FROM sensors

-- nested structures access
SELECT foo.'bar' FROM table; -- must quote nested
column

-- timestamps
SELECT * FROM payments
WHERE eventTimestamp > CURRENT_TIMESTAMP-interval
'10' second;

-- unnest
SELECT b.*, u.*
FROM bgp_avro b,
UNNEST(b.path) AS u(pathitem)
```

```
-- aggregations and windows
SELECT card,
MAX(amount) as theamount,
TUMBLE_END(eventTimestamp, interval '5' minute) as
ts
FROM payments
WHERE lat IS NOT NULL
AND lon IS NOT NULL
GROUP BY card,
TUMBLE(eventTimestamp, interval '5' minute)
HAVING COUNT(*) > 4  -- >4==fraud

-- try to do this ksql!
SELECT us_west.user_score+ap_south.user_score
FROM kafka_in_zone_us_west us_west
FULL OUTER JOIN kafka_in_zone_ap_south ap_south
ON us_west.user_id = ap_south.user_id;
```

SQL Stream Builder (SSB)

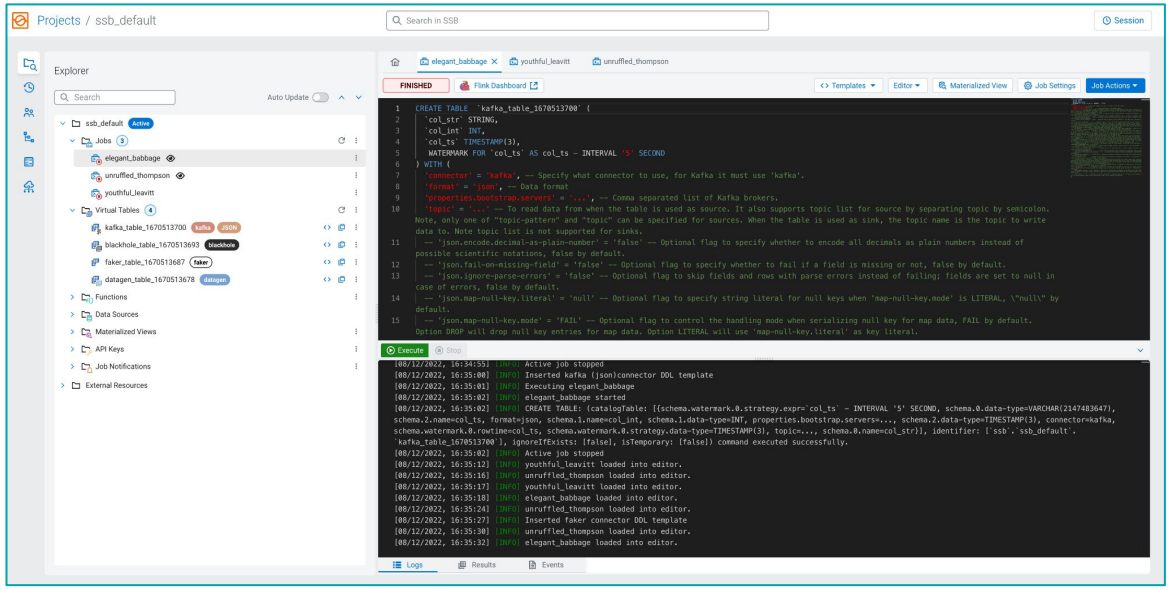
Democratize access to real-time data with just SQL

SQL STREAM BUILDER allows developers, analysts, and data scientists to **write streaming applications** with industry standard **SQL**.

No Java or Scala code development required.

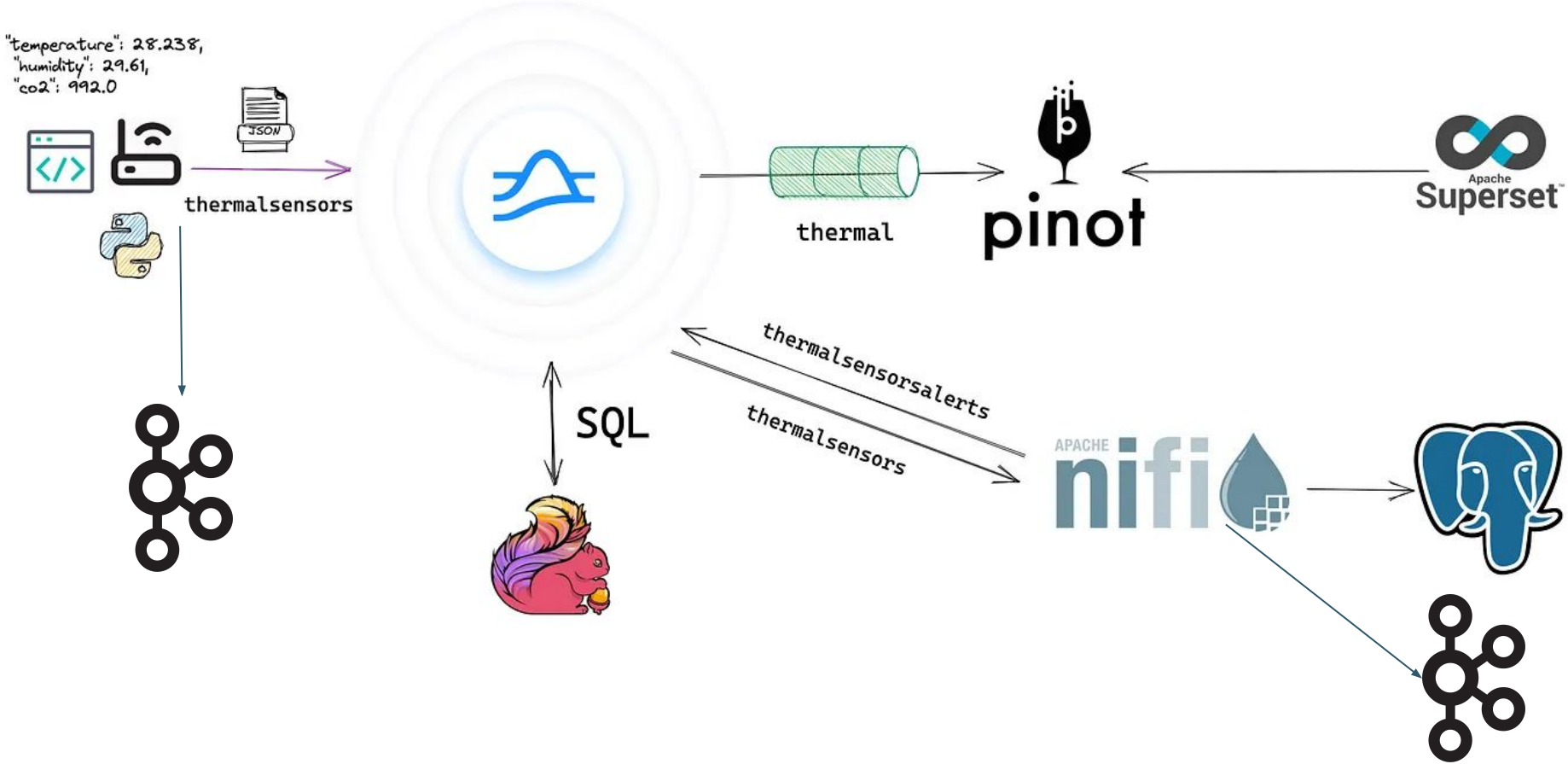
Simplifies access to data in Kafka & Flink. Connectors to batch data in HDFS, Kudu, Hive, S3, JDBC, CDC and more

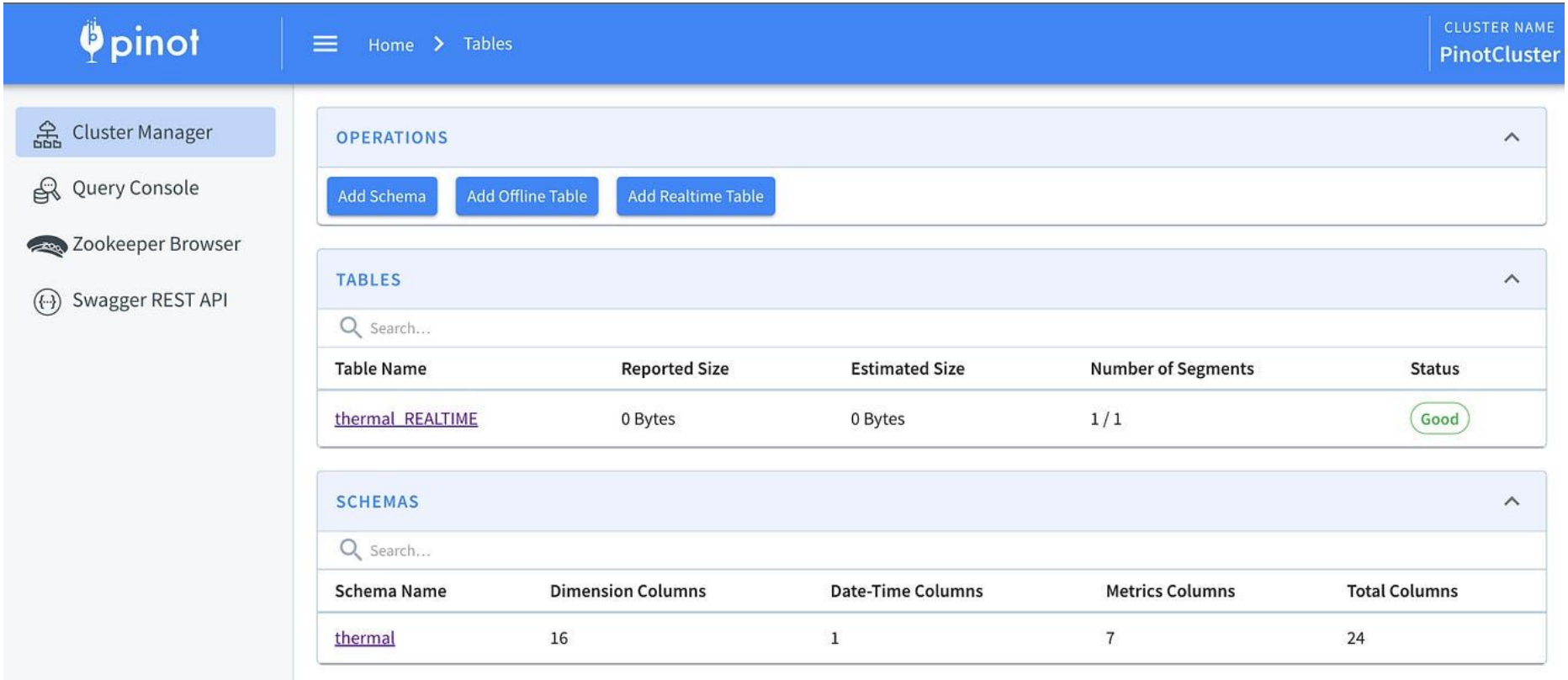
Enrich streaming data with batch data in a single tool

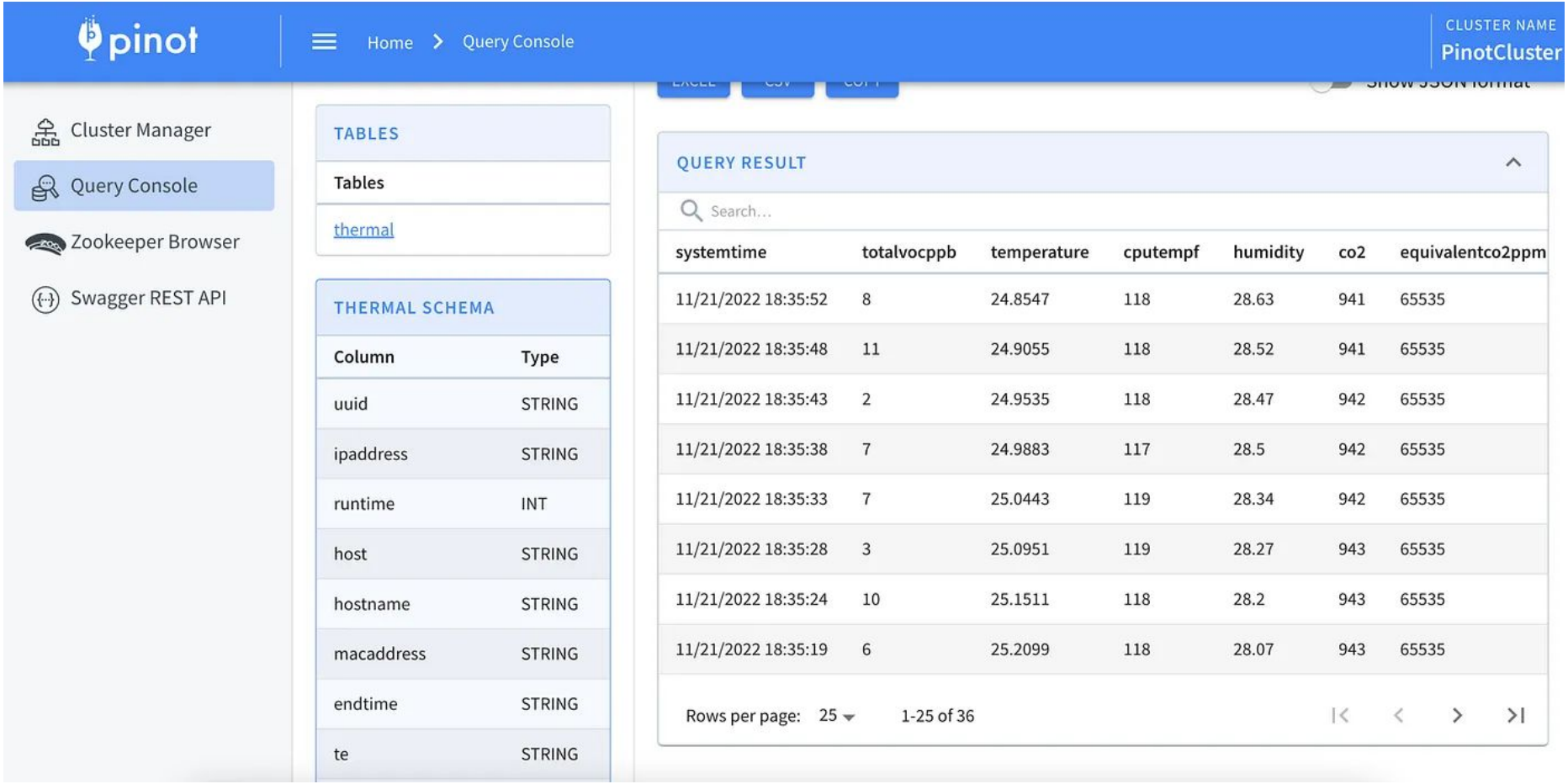


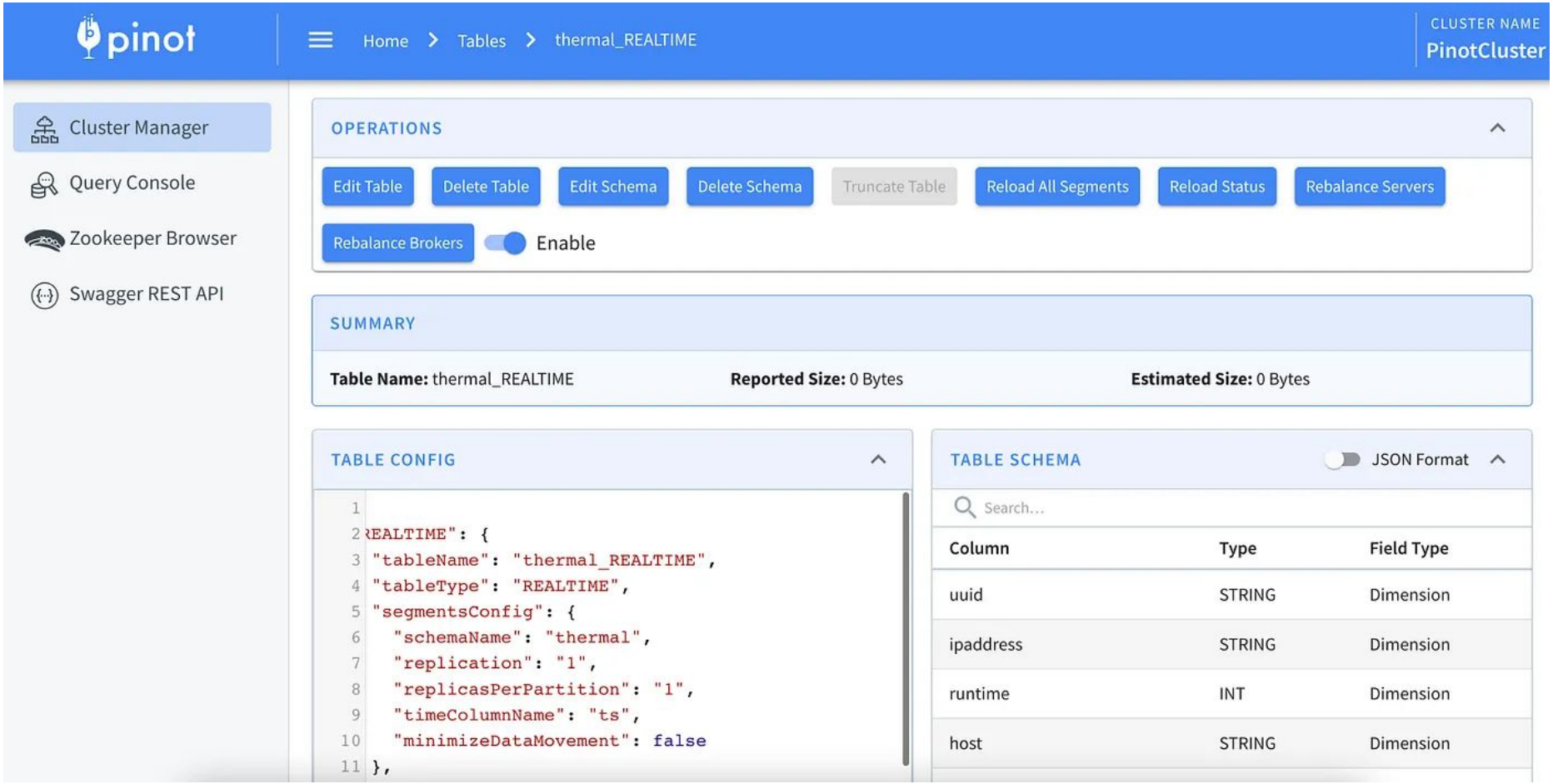
DEMO

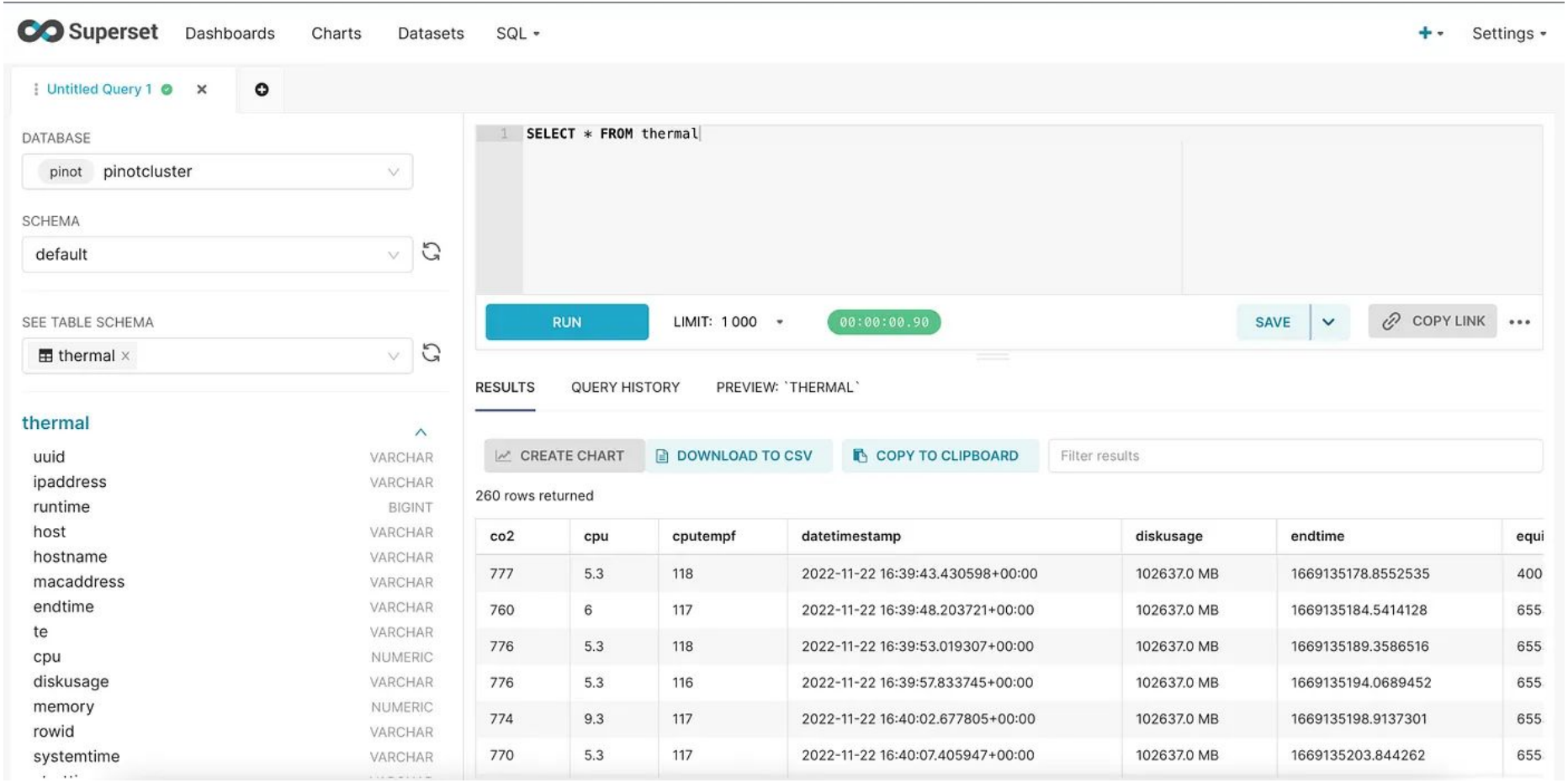
Device -> Pulsar -> Pinot


"temperature": 28.238,
"humidity": 29.61,
"co2": 992.0
JSON
thermalsensors
thermal
pinot
Apache
Superset
SQL
thermalsensorsalerts
thermalsensors
APACHE
nifi


pinot
Home
Tables
CLUSTER NAME
PinotCluster
Cluster Manager
Query Console
Zookeeper Browser
Swagger REST API
OPERATIONS
Add Schema
Add Offline Table
Add Realtime Table
TABLES
Search...
Table Name
Reported Size
Estimated Size
Number of Segments
Status
thermal_REALTIME
0 Bytes
0 Bytes
1 / 1
Good
SCHEMAS
Search...
Schema Name
Dimension Columns
Date-Time Columns
Metrics Columns
Total Columns
thermal
16
1
7
24


pinot
Home
Query Console
CLUSTER NAME
PinotCluster
Cluster Manager
Query Console
Zookeeper Browser
Swagger REST API
TABLES
Tables
thermal
THERMAL SCHEMA
Column
Type
uuid
STRING
ipaddress
STRING
runtime
INT
host
STRING
hostname
STRING
macaddress
STRING
endtime
STRING
te
STRING
QUERY RESULT
Search...
systemtime
totalvocppb
temperature
cputempf
humidity
co2
equivalentco2ppm
11/21/2022 18:35:52
8
24.8547
118
28.63
941
65535
11/21/2022 18:35:48
11
24.9055
118
28.52
941
65535
11/21/2022 18:35:43
2
24.9535
118
28.47
942
65535
11/21/2022 18:35:38
7
24.9883
117
28.5
942
65535
11/21/2022 18:35:33
7
25.0443
119
28.34
942
65535
11/21/2022 18:35:28
3
25.0951
119
28.27
943
65535
11/21/2022 18:35:24
10
25.1511
118
28.2
943
65535
11/21/2022 18:35:19
6
25.2099
118
28.07
943
65535
Rows per page:
25
1-25 of 36


pinot
Home > Tables > thermal_REALTIME
CLUSTER NAME
PinotCluster
Cluster Manager
Query Console
Zookeeper Browser
Swagger REST API
OPERATIONS
Edit Table
Delete Table
Edit Schema
Delete Schema
Truncate Table
Reload All Segments
Reload Status
Rebalance Servers
Rebalance Brokers
Enable
SUMMARY
Table Name: thermal_REALTIME
Reported Size: 0 Bytes
Estimated Size: 0 Bytes
TABLE CONFIG
1
2 REALTIME": {
3 "tableName": "thermal_REALTIME",
4 "tableType": "REALTIME",
5 "segmentsConfig": {
6 "schemaName": "thermal",
7 "replication": "1",
8 "replicasPerPartition": "1",
9 "timeColumnName": "ts",
10 "minimizeDataMovement": false
11 },
TABLE SCHEMA
JSON Format
Search...
Column
Type
Field Type
uuid
STRING
Dimension
ipaddress
STRING
Dimension
runtime
INT
Dimension
host
STRING
Dimension


Superset
Dashboards
Charts
Datasets
SQL
Settings
Untitled Query 1
DATABASE
pinot
pinotcluster
SCHEMA
default
SEE TABLE SCHEMA
thermal
thermal
uuid VARCHAR
ipaddress VARCHAR
runtime BIGINT
host VARCHAR
hostname VARCHAR
macaddress VARCHAR
endtime VARCHAR
te VARCHAR
cpu NUMERIC
diskusage VARCHAR
memory NUMERIC
rowid VARCHAR
systemtime VARCHAR
1 SELECT * FROM thermal
RUN
LIMIT: 1 000
00:00:00.90
SAVE
COPY LINK
RESULTS
QUERY HISTORY
PREVIEW: `THERMAL`
CREATE CHART
DOWNLOAD TO CSV
COPY TO CLIPBOARD
Filter results
260 rows returned
co2 cpu cputempf datetimestamp diskusage endtime equi
777 5.3 118 2022-11-22 16:39:43.430598+00:00 102637.0 MB 1669135178.8552535 400
760 6 117 2022-11-22 16:39:48.203721+00:00 102637.0 MB 1669135184.5414128 655
776 5.3 118 2022-11-22 16:39:53.019307+00:00 102637.0 MB 1669135189.3586516 655
776 5.3 116 2022-11-22 16:39:57.833745+00:00 102637.0 MB 1669135194.0689452 655
774 9.3 117 2022-11-22 16:40:02.677805+00:00 102637.0 MB 1669135198.9137301 655
770 5.3 117 2022-11-22 16:40:07.405947+00:00 102637.0 MB 1669135203.844262 655

Superset Dashboards Charts Datasets SQL

Create a new chart

Choose a dataset

thermal

thermal

Add a dataset or view instructions.

All charts

Search all charts

Recommended tags

Popular

ECharts

Advanced-Analytics

215
+7.0% WoW
Big Number

80.7M
Big Number

Table

Pivot Table v2

Time-series

Time-series

Time-series

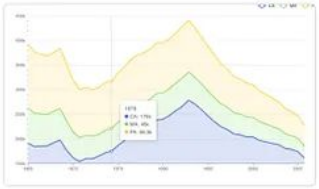
Time-series

Time-series Area Chart

ECharts Predictive Advanced-Analytics Aesthetic
Time Line Transformable Stacked Popular

Time-series Area chart are similar to line chart in that they represent variables with the same scale, but area charts stack the metrics on top of each other. An area chart in Superset can be stream, stack, or

Examples



CREATE NEW CHART

Thermal Sensor Data ★ Draft

EDIT DASHBOARD ⋯

thermaltable

totalvocppb	humidity	temperature	cputempf	datetimestamp
0	22.45	29.4129	118	2022-11-22
0	22.9	29.0337	117	2022-11-22
0	23.27	28.7801	118	2022-11-22
0	23.59	28.5504	116	2022-11-22

thermaltablets

datetimestamp	host	macaddress	endtime
2022-11-22	thermal	e4:5f:01:7c:3f:34	1669148339.271487
2022-11-22	thermal	e4:5f:01:7c:3f:34	1669148344.9127774
2022-11-22	thermal	e4:5f:01:7c:3f:34	1669148349.722541
2022-11-22	thermal	e4:5f:01:7c:3f:34	1669148354.5408292
2022-11-22	thermal	e4:5f:01:7c:3f:34	1669148359.410982

avgEquivalentCo2PPM

53.3k

thermalchart4

Time	cpu	rowid	datet
2022-11-22 20:19:04	4.8	20221122201859_ed73f6c9-c497-4e40-9828-e1a6319e3c15	2022
2022-11-22 16:39:44	5.3	20221122163938_d3981346-97f9-448b-b43d-b916e6cf718e	2022
2022-11-22 20:19:09	10.3	20221122201904_68f81865-8ba5-4249-88e4-cb98b60dc423	2022
2022-11-22 16:39:49	6	20221122163944_c4aedd91-cca1-4ecf-b56a-c782abb1e74c	2022
2022-11-21 23:20:14	6.8	20221121232009_87f9e59c-5100-41aa-b4ef-25b2f2070f16	2022

Schemas

Build a schema

https://github.com/startreedata/pinot-recipes/tree/main/recipes/infer-schema-json-data

Development Resources

https://docs.pinot.apache.org/basics/data-import/pinot-stream-ingestion/apache-pulsar

https://dev.startree.ai/docs/pinot/recipes/pulsar

https://github.com/startreedata/pinot-recipes/tree/main/recipes/pulsar

Easy Docker Demo

```
docker exec -it pinot-controller /bin/bash

docker exec -it pinot-controller bin/pinot-admin.sh JsonToPinotSchema \
  -timeColumnName ts \
  -metrics "temperature,humidity,co2,totalvocppb,equivalentco2ppm,pressure,temperatureicp,cputempf"\
  -dimensions "host,ipaddress" \
  -pinotSchemaName=thermal \
  -jsonFile=/data/thermal.json \
  -outputDir=/config

docker exec -it pinot-controller bin/pinot-admin.sh AddSchema   \
 -schemaFile /config/thermalschema.json \
 -exec
```

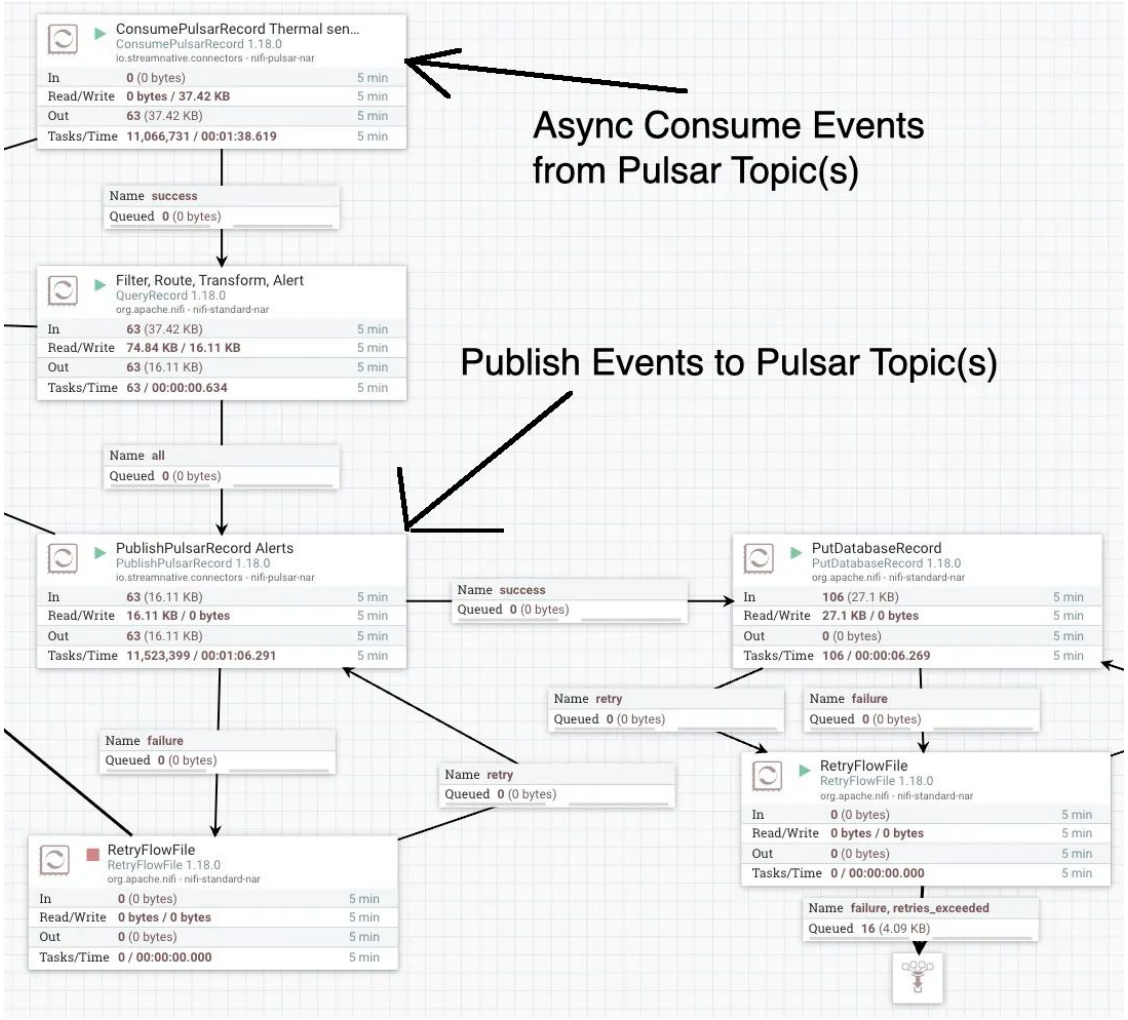
Local Apache Pinot Admin

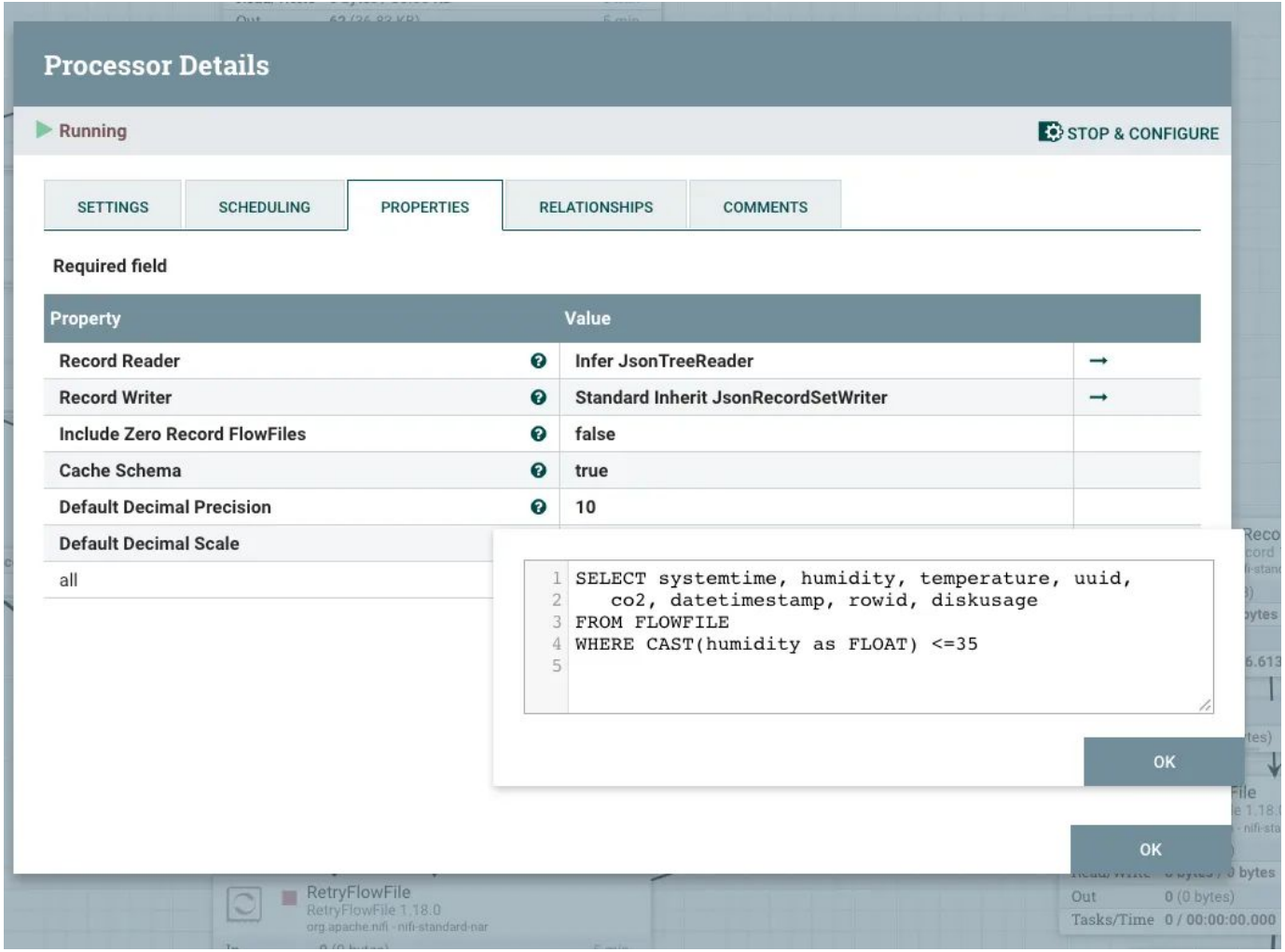
```
curl -X DELETE "http://localhost:9000/tables/thermal?type=realtime" -H "accept: application/json"
```

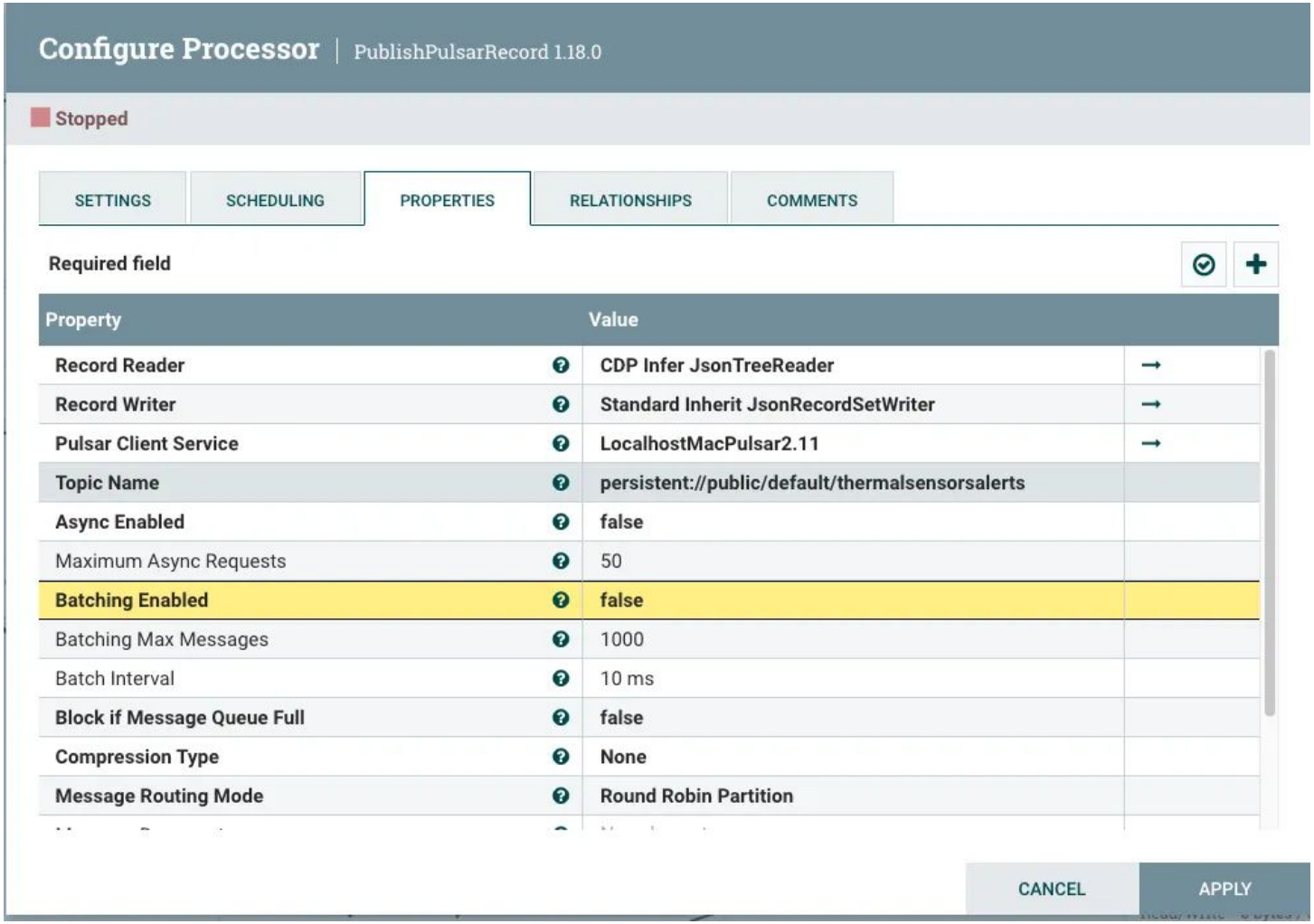
```
curl -X DELETE "http://localhost:9000/schemas/thermal" -H "accept: application/json"
```

```
docker exec -it pinot-controller bin/pinot-admin.sh AddSchema   \
 -schemaFile /config/thermalschema.json \
 -exec
```

```
curl -X POST "http://localhost:9000/tables" -H "accept: application/json" -H " ….
```

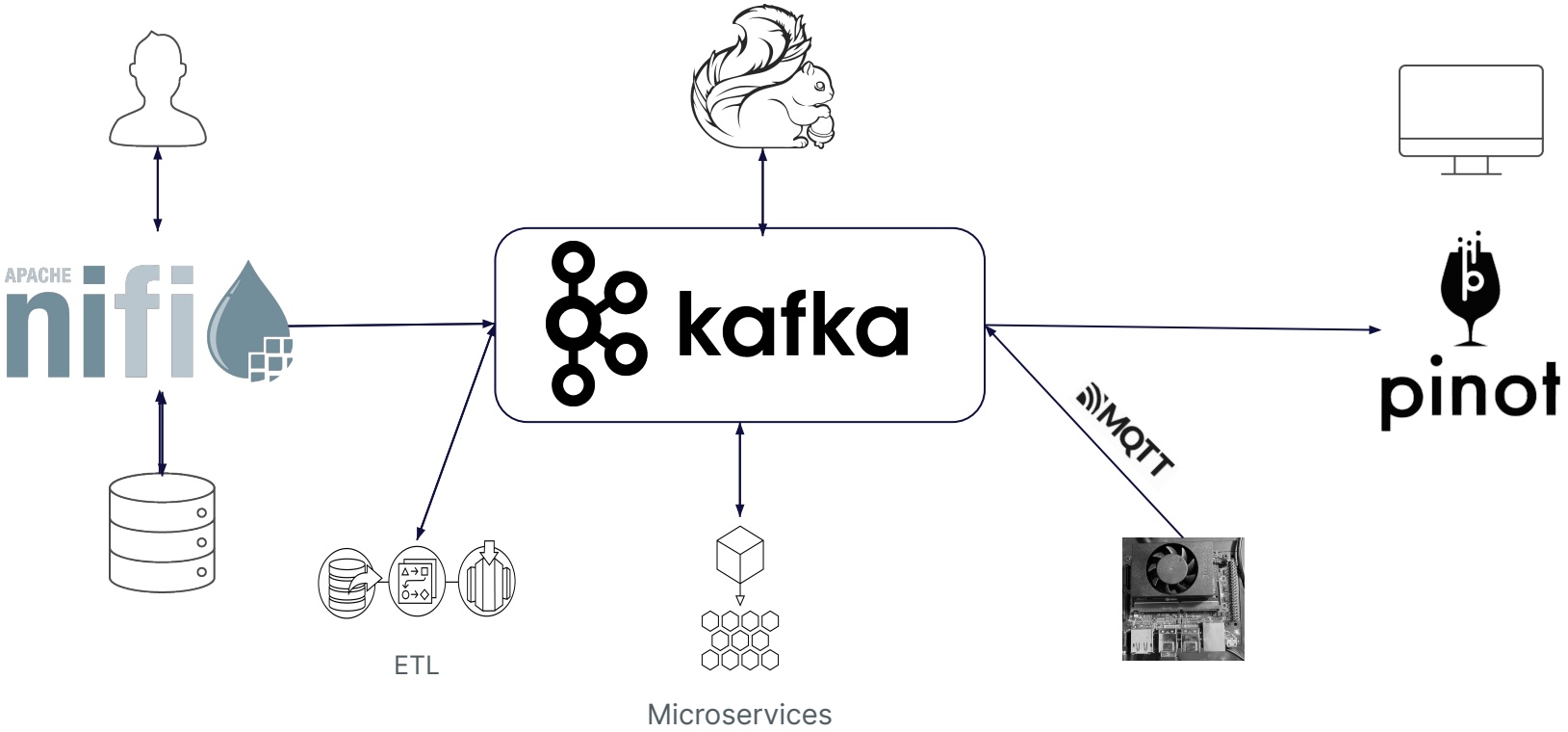

ConsumePulsarRecord Thermal sen...
ConsumePulsarRecord 1.18.0
io.streamnative.connectors - nifi-pulsar-nar
In 0 (0 bytes) 5 min
Read/Write 0 bytes / 37.42 KB 5 min
Out 63 (37.42 KB) 5 min
Tasks/Time 11,066,731 / 00:01:38.619 5 min
Async Consume Events from Pulsar Topic(s)
Name success
Queued 0 (0 bytes)
Filter, Route, Transform, Alert
QueryRecord 1.18.0
org.apache.nifi - nifi-standard-nar
In 63 (37.42 KB) 5 min
Read/Write 74.84 KB / 16.11 KB 5 min
Out 63 (16.11 KB) 5 min
Tasks/Time 63 / 00:00:00.634 5 min
Publish Events to Pulsar Topic(s)
Name all
Queued 0 (0 bytes)
PublishPulsarRecord Alerts
PublishPulsarRecord 1.18.0
io.streamnative.connectors - nifi-pulsar-nar
In 63 (16.11 KB) 5 min
Read/Write 16.11 KB / 0 bytes 5 min
Out 63 (16.11 KB) 5 min
Tasks/Time 11,523,399 / 00:01:06.291 5 min
Name success
Queued 0 (0 bytes)
PutDatabaseRecord
PutDatabaseRecord 1.18.0
org.apache.nifi - nifi-standard-nar
In 106 (27.1 KB) 5 min
Read/Write 27.1 KB / 0 bytes 5 min
Out 0 (0 bytes) 5 min
Tasks/Time 106 / 00:00:06.269 5 min
Name retry
Queued 0 (0 bytes)
Name failure
Queued 0 (0 bytes)
Name failure
Queued 0 (0 bytes)
Name retry
Queued 0 (0 bytes)
RetryFlowFile
RetryFlowFile 1.18.0
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
RetryFlowFile
RetryFlowFile 1.18.0
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
Name failure, retries_exceeded
Queued 16 (4.09 KB)


Processor Details
Running
STOP & CONFIGURE
SETTINGS
SCHEDULING
PROPERTIES
RELATIONSHIPS
COMMENTS
Required field
Property
Value
Record Reader
Infer JsonTreeReader
Record Writer
Standard Inherit JsonRecordSetWriter
Include Zero Record FlowFiles
false
Cache Schema
true
Default Decimal Precision
10
Default Decimal Scale
all
SELECT systemtime, humidity, temperature, uuid,
co2, datetimestamp, rowid, diskusage
FROM FLOWFILE
WHERE CAST(humidity as FLOAT) <=35
OK
OK
RetryFlowFile
RetryFlowFile 1.18.0
org.apache.nifi - nifi-standard-nar
Out
0 (0 bytes)
Tasks/Time
0 / 00:00:00.000


Configure Processor | PublishPulsarRecord 1.18.0
Stopped
SETTINGS
SCHEDULING
PROPERTIES
RELATIONSHIPS
COMMENTS
Required field
Property
Value
Record Reader
CDP Infer JsonTreeReader
Record Writer
Standard Inherit JsonRecordSetWriter
Pulsar Client Service
LocalhostMacPulsar2.11
Topic Name
persistent://public/default/thermalsensorsalerts
Async Enabled
false
Maximum Async Requests
50
Batching Enabled
false
Batching Max Messages
1000
Batch Interval
10 ms
Block if Message Queue Full
false
Compression Type
None
Message Routing Mode
Round Robin Partition
CANCEL
APPLY

NiFi -> Kafka -> Pinot

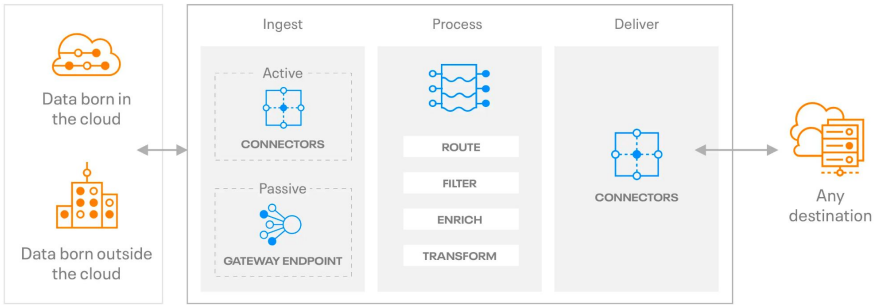
Reference Architecture

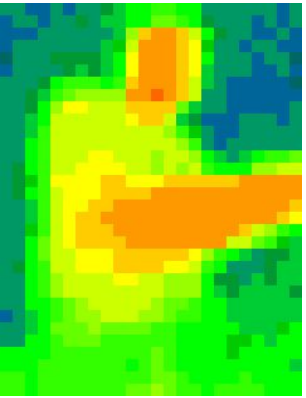


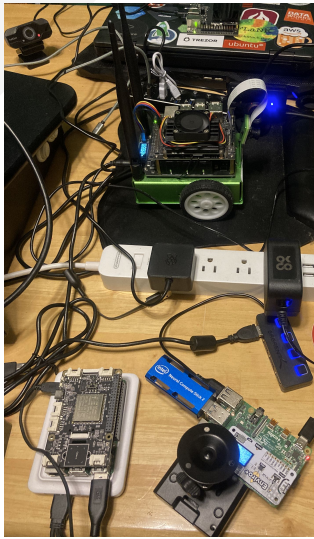
RESOURCES AND WRAP-UP

Resources


DATA AND CAT
imgflip.com


Data born in the cloud
Data born outside the cloud
Ingest
Process
Deliver
Active
CONNECTORS
Passive
GATEWAY ENDPOINT
ROUTE
FILTER
ENRICH
TRANSFORM
CONNECTORS
Any destination





THANK YOU