



Meetup: Mastering Data Streaming Pipelines

Tim Spann
Principal Developer Advocate

28-June-2023



Streaming Data Pipeline Development

Join us on Wednesday, June 28, 2023 | 6:00 PM ET



Ian Brooks, Ph.D.
Principal Solutions Engineer



Timothy Spann
Principal Developer Advocate



CLOUDERA



CLOUDERA



EDGE
2AI

CLOUDERA



Future of Data - Princeton + Virtual



<https://www.meetup.com/futureofdata-princeton/>

From Big Data to AI to Streaming to Containers to Cloud to Analytics to Cloud Storage to Fast Data to Machine Learning to Microservices to ...



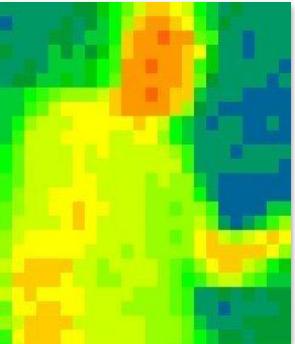
FUTURE OF DATA

AN OPEN SOURCE COMMUNITY



@PaasDev

FLaNK Stack



Tim Spann

@PaasDev // Blog: www.datainmotion.dev

Principal Developer Advocate.

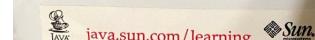
Princeton Future of Data Meetup.

ex-Pivotal, ex-Hortonworks, ex-StreamNative, ex-PwC

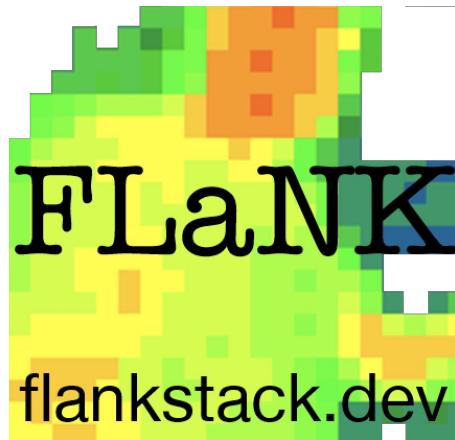
<https://medium.com/@tspann>

<https://github.com/tspannhw>

Apache NiFi x Apache Kafka x Apache Flink x Java



FLaNK Stack Weekly



<https://bit.ly/32dAJft>



This week in Apache NiFi, Apache Flink, Apache Kafka, Apache Spark, Apache Iceberg, Python, Java and Open Source friends.

CLOUDERA NOW '23

<https://www.cloudera.com/about/events/cloudera-now-cdp.html>

FREE LEARNING ENVIRONMENT

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



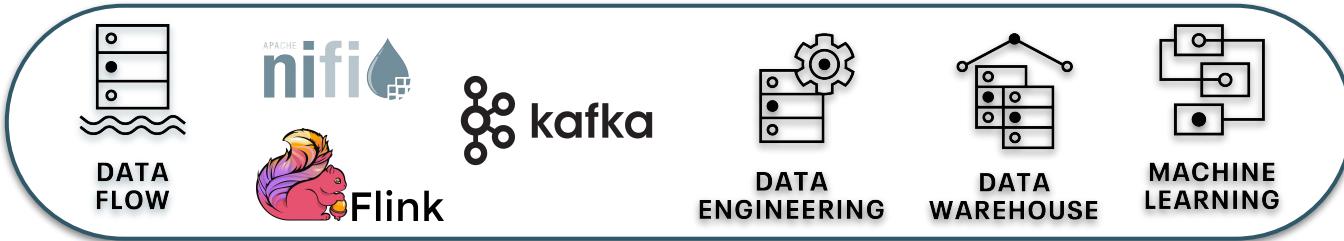
CSP Community Edition

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](#)

STREAMING

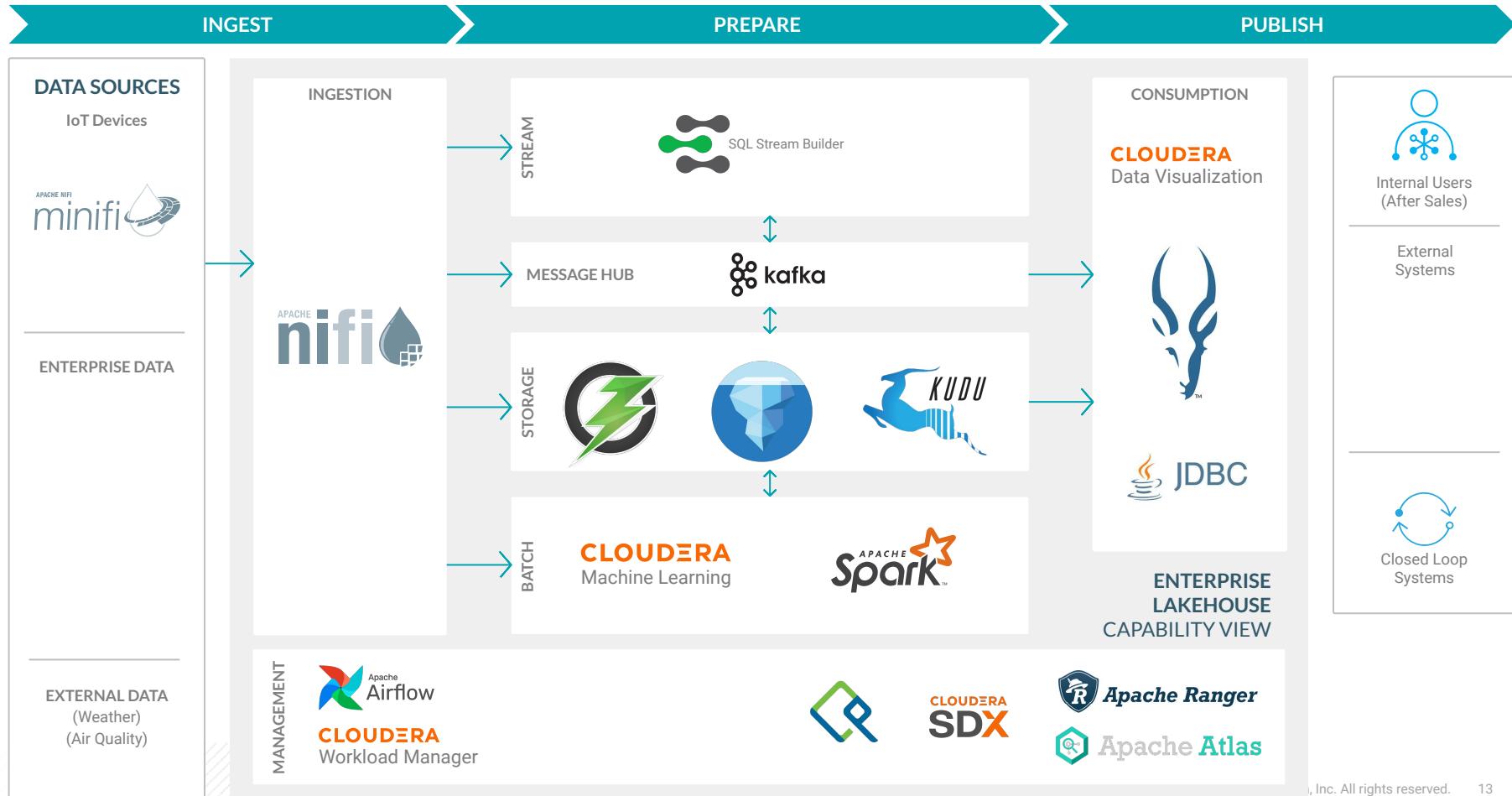
CDP: AN OPEN DATA LAKEHOUSE



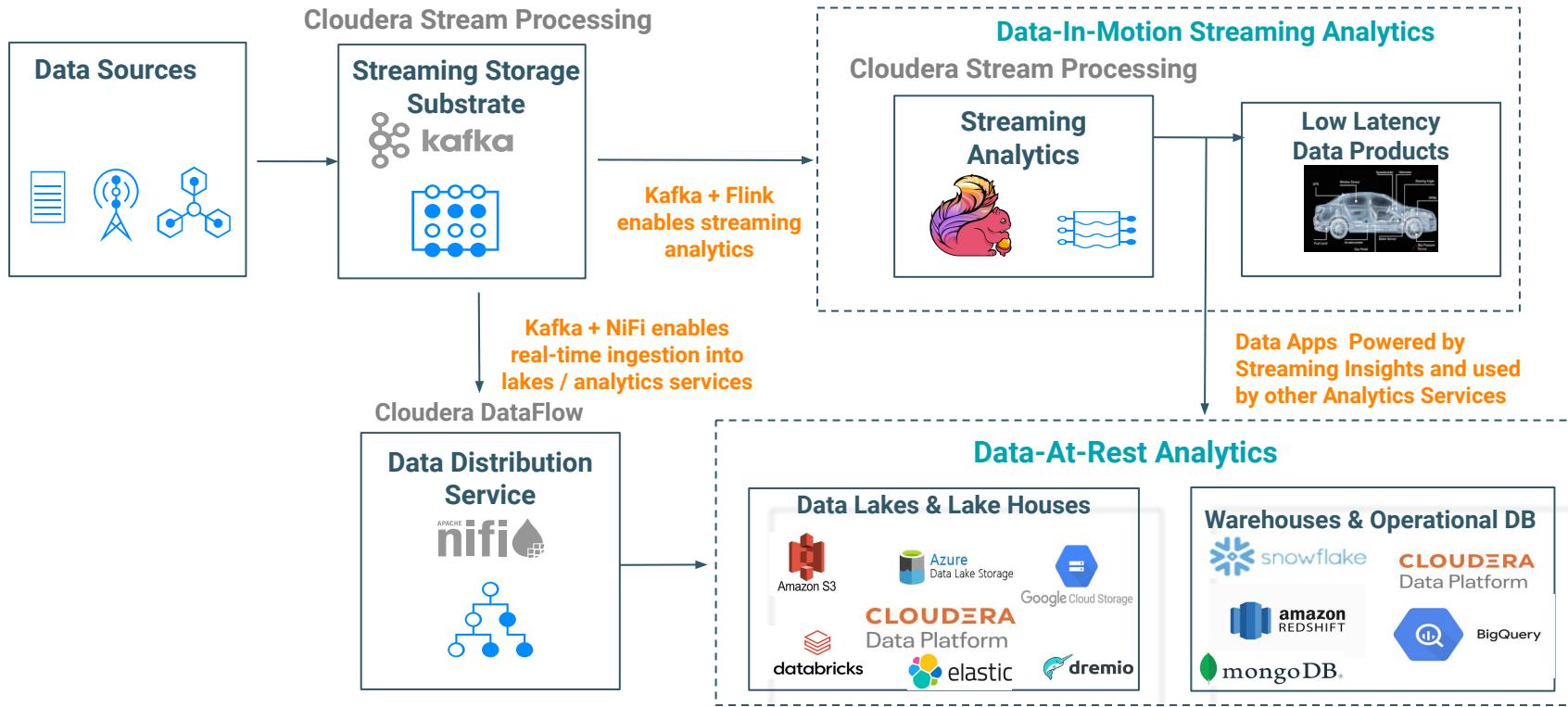
BUILDING REAL-TIME REQUIRES A TEAM



DATA INGESTION ARCHITECTURE - Using NiFi to pull the Data out of anything in near-real time



Moving Beyond Draining of Streams Into Lakes: Analytics-in-Stream





Sploot



Nano



Shaggy



Stormy



Spark, NiFi, Flink? Which engine to choose?

Already using **Spark**?

Want unified **Batch/Stream**?

Want highest **Throughput**?

Don't need **low latency**?

Large files?

Scheduled batches?

Replacing Sqoop, ETL



Need **NiFi**?

Simple JDBC queries?

Transform individual records?

Want **easy development**?

Lots of small files, events, records, rows? Want **Advanced Windowing** and **State**?

Continuous stream of rows

Support many different sources



Need **Flink**?

Need **Microservices**, **Batch** and **Stream**?

Want high **Throughput**?

Want **Low Latency**?

Happy with a **New Solution** that is best-in-class?



SOURCES AND SINKS



APACHE ICEBERG

A Flexible, Performant & Scalable Table Format

- Donated by **Netflix** to the Apache Foundation in 2018
- Flexibility
 - Hidden partitioning
 - Full schema evolution
- Data Warehouse Operations
 - Atomic Consistent Isolated Durable (ACID) Transactions
 - Time travel and rollback
- Supports best in class SQL performance
 - High performance at Petabyte scale





AMQP



AWS Lambda



Airtable

Amazon
API Gateway

Amazon CloudWatch



Amazon DynamoDB



Amazon Kinesis Data Firehose



Amazon Kinesis Data Streams



Amazon SQS

Amazon Simple Notification Services
(SNS)

Amazon Simple Storage Service (S3)



Apache Accumulo



Apache Cassandra



Apache HBase



Apache Hive



Apache Iceberg



Apache Ignite



Apache Kafka



Apache Kudu



Apache Solr

COMMON USE CASES

BE READY TO DETECT, RESPOND AND COMPLY

Real-time alerting & Longer retention = Enhanced Readiness

1

Detect

Collect event data at high events per second
Automatically identify high fidelity alerts
Analytics for proactive threat hunting

2

Respond

Respond automatically to real-time alerts
Full context for complete forensics and incident response

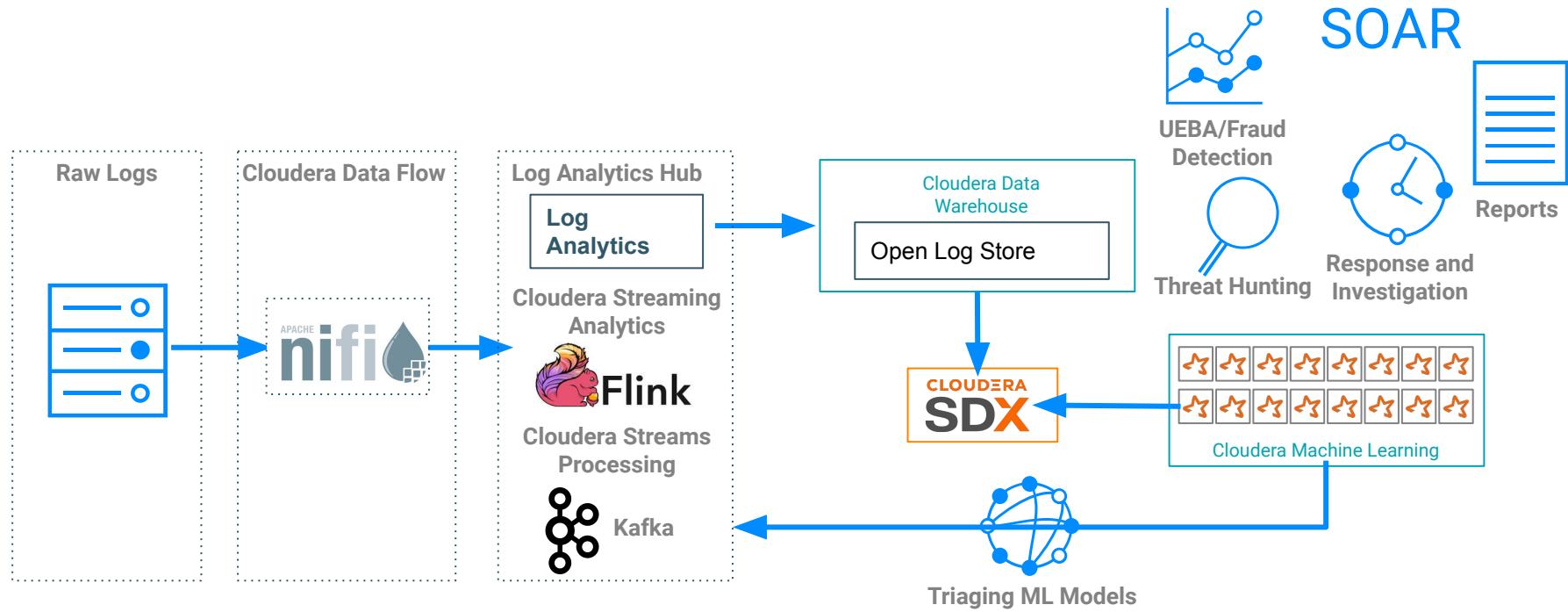
3

Comply

Retain data as required
Automated compliance assessments

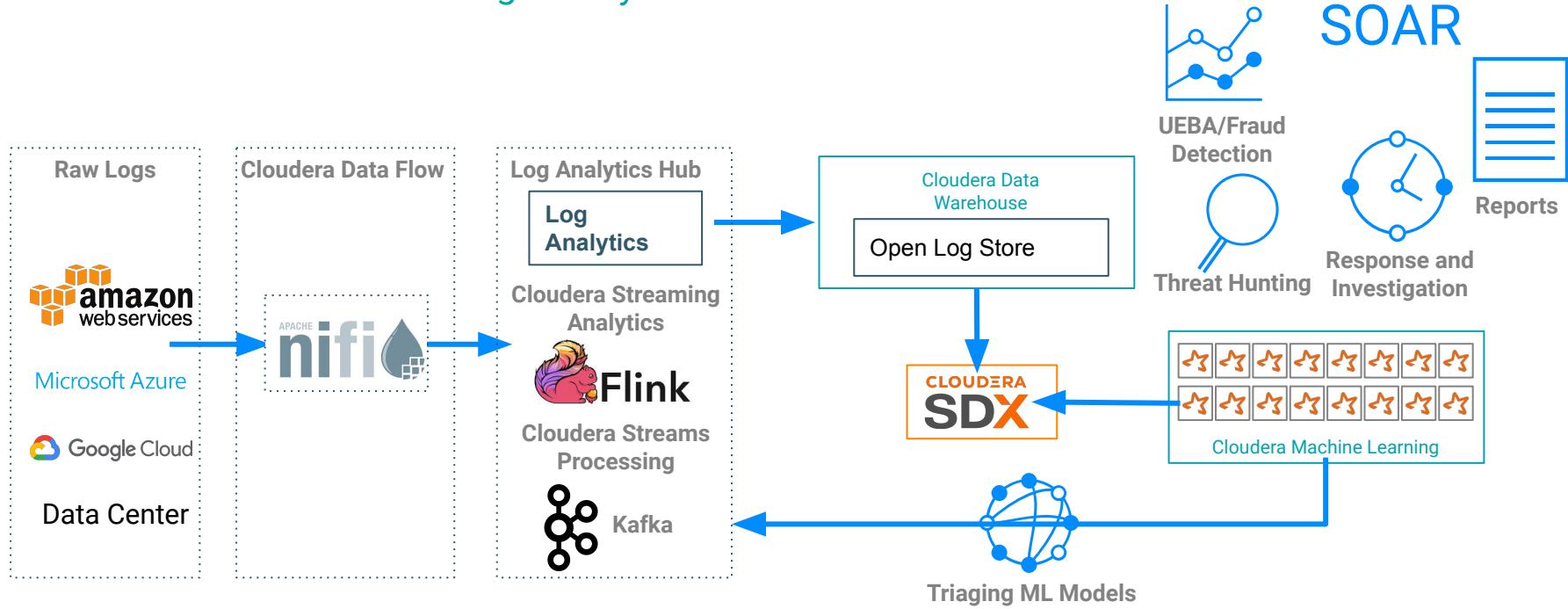
STREAMING LOG ANALYTICS REFERENCE ARCHITECTURE

Scalable and Secure Real-Time Log Analytics



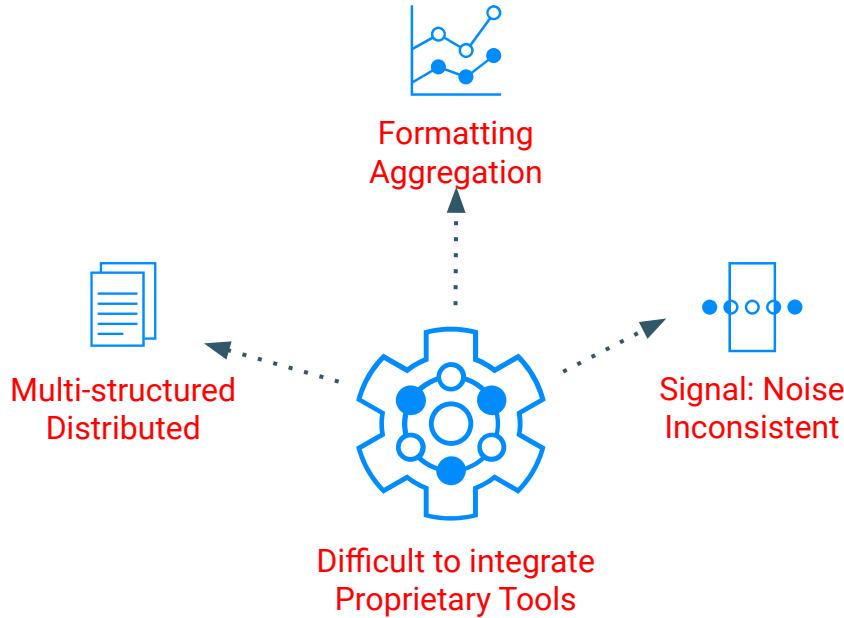
CONSOLIDATE ACROSS CLOUDS

Run collection and streaming on any cloud



THE PROBLEM

Making Observable Events Actionable can be Challenging



Resulting in....

- Inability to **distribute** event data
- Data Silos
- Inefficient data movement
- **Situational awareness**



- License costs
- Consumption costs
- Development overhead
- **Resource utilization**



CLOUDERA DATA-IN-MOTION SOLUTION

Enable Real-Time Observability

- Dramatically reduce time to insight
- Monitor proactively for critical events
- **Distribute broadly and process in-stream real-time via stream**



Create Useful Information

- Process raw data
- Integrate with enterprise data
- **Build, test, deploy quickly**



01

02

03

04

Actionability

Filter and Distribute

- Reduce noise, boost signal
- Reduce load on other system
- **Filter, compress, route anywhere**



Capture Observable Data

- Capture events from any system anywhere
- Any data format, any protocol
- **Open- Not limited to proprietary tools**



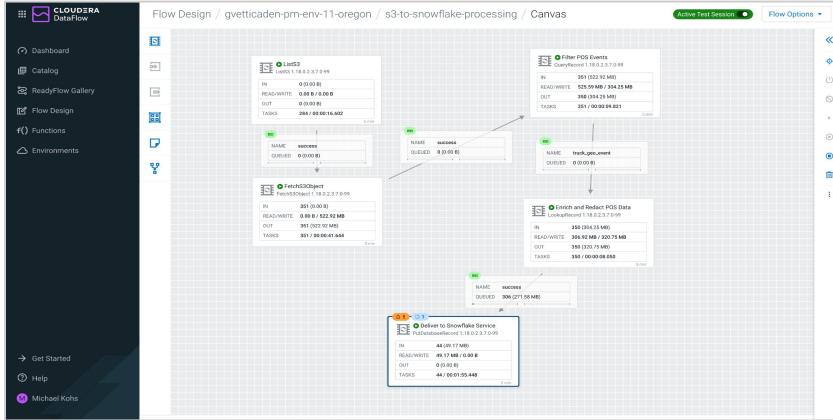
Universal Connectivity
450+ Connectors

Hybrid Deployment-Ready
Author once, deploy anywhere

Self-Service across full lifecycle
Integrated no code UX

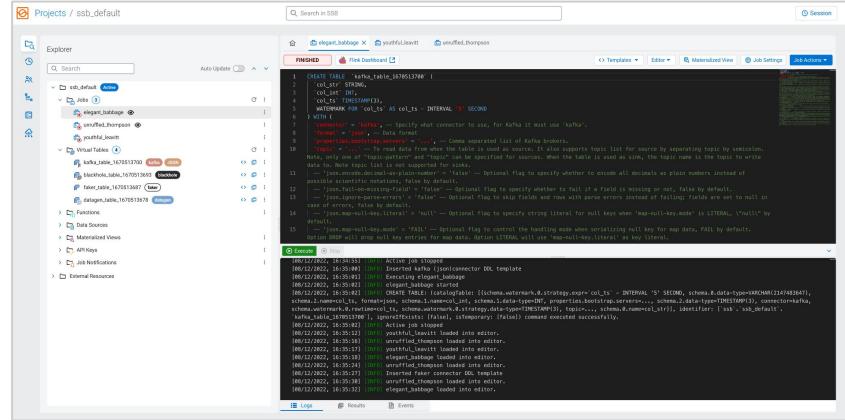
NO-CODE/LOW-CODE USER EXPERIENCE

DataFlow Designer



visual canvas + automatic provisioning =
Self-service pipeline development

SQL Stream Builder



Instant data access + unified SQL processing =
Self-service streaming event processing

KEY USE CASES AND OUTCOMES

CYBER SECURITY

"Cut log data by 60% resulting in >40% reduction in Splunk costs.

Delivered cyber logs to other teams analyzing logs resulting in a **decrease of MTTD by 90%.**

Fortune 100 Energy Company

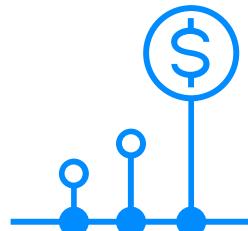
"Improved threat hunting by **eliminating 25% of false positive anomalies** and speeding up investigations"

Global Telco

CUSTOMER INSIGHT

"Improved Marketing Campaign effectiveness 15% YoY. This allowed us to redirect funds for failing campaigns and allocate to new ones."

Global Telco



INTELLIGENT OPS & SYSTEM RELIABILITY

"Cut our data volume in half that translated to a 30%-40% reduction in public cloud cost. With saving, expanded footprint & coverage of monitoring & visibility without a significant \$\$\$ impact. This drove **improved situational awareness of app infrastructure availability & performance**"

Large US Airline

"Faster resolution of incidents, reducing exponentially the stops of trains in full operation."

Metro de Madrid



APACHE NiFi - MiNiFi Agents

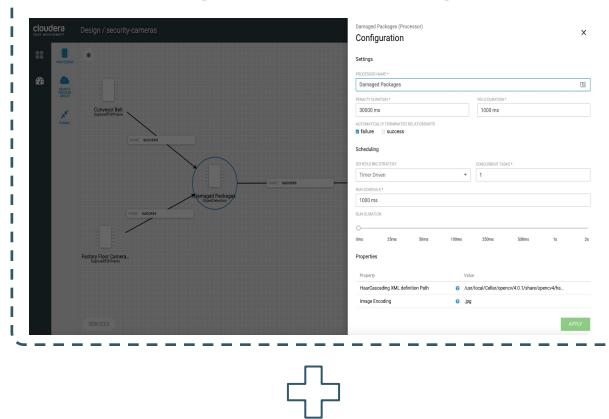


NIFI For Ants

Cloudera Edge Management

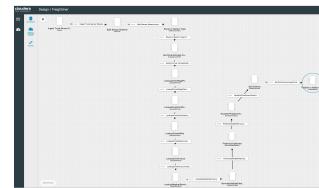
Edge device data collection and processing with easy to use central command and control

Edge Flow Manager

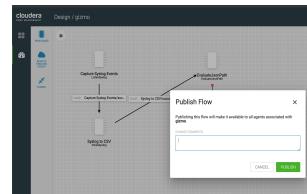


A lightweight edge agent that implements the core features of Apache NiFi, focusing on data collection and processing at the edge

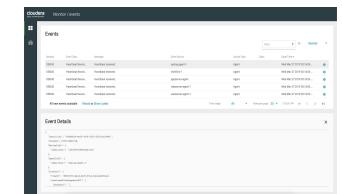
Flow Authorship



Flow Deployment



Flow Monitoring

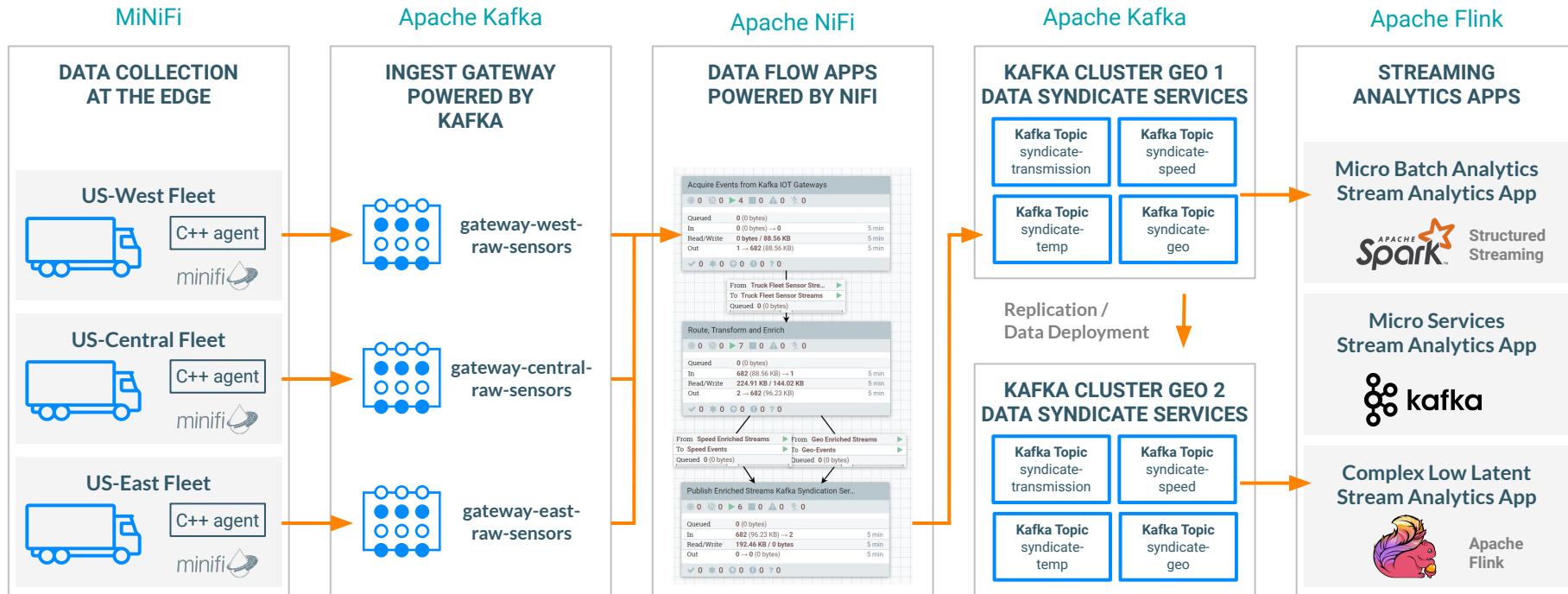


- Small footprint agent with MiNiFi
- Java and C++ agents
- Rich edge processors (edge collection & processing)
- End to end lineage and security
- Kubernetes support
- Central Command and Control (C2)
- Design and deploy to millions of agents
- Edge Applications lifecycle management
- Multitenancy with Agent classes
- Native integration with other CDF services

APACHE KAFKA



Apache Kafka



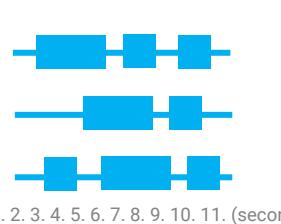
APACHE FLINK



DELIVERING STREAMING ANALYTICS

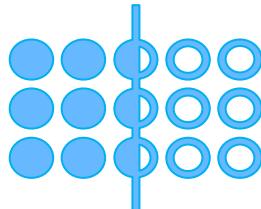
Capture Events that Matter

Low-latency analytics use cases



Parsing and Blending Data

Both offline and streaming data



Data Analysts Can Write Queries

Across the Lines of Business



Streaming Analytics

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.

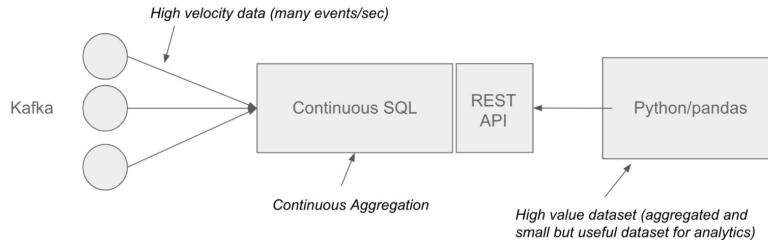
Simplifies access to data in Kafka & Flink. Connectors to batch data in HDFS, Kudu, Hive, S3, JDBC, CDC and more

Enrich streaming data with batch data in a single tool

```
1 CREATE TABLE `kafka_table_1670513700` (
2   `col_str` STRING,
3   `col_int` INT,
4   `col_ts` TIMESTAMP(3),
5   WATERMARK FOR `col_ts` AS col_ts - INTERVAL '5' SECOND
6 ) WITH (
7   'connector' = 'kafka', -- Specify what connector to use, for Kafka it must use 'kafka'.
8   'format' = 'json', -- Topic name to read from.
9   'topic' = 'elegant_babbage', -- Comma separated list of Kafka brokers.
10  'properties.bootstrap.servers' = '...', -- Optional flag to specify whether to encode all decimals as plain numbers instead of
Note, only one of 'topic-pattern' and 'topic' can be specified for sources. When the table is used as sink, the topic name is the topic to write
11  'properties.ignoreDecimalsAsPlainNumbers' = 'false' -- Optional flag to specify whether to fail if a field is missing or not, false by default.
12  'parseAsString' = 'true' -- Optional flag to parse strings by default.
13  'json.fail-on-missing-field' = 'false' -- Optional flag to skip fields and rows with parse errors instead of failing; fields are set to null in
case of errors, false by default.
14  'json.ignore-errors' = 'false' -- Optional flag to skip fields and rows with parse errors instead of failing; fields are set to null in
case of errors, false by default.
15  'json.map-null-key.literal' = 'null' -- Optional flag to specify string literal for null keys when 'map-null-key.mode' is LITERAL, '\"null\"' by
default.
16  'map-null-key.mode' = 'FAIL' -- Optional flag to control the handling mode when serializing null key for map data, FAIL by default.
Option DROP will drop null key entries for map data. Option LITERAL will use 'map-null-key.literal' as key literal.
```

SSB MATERIALIZED VIEWS

Key Takeaway; MV's allow data scientist, analyst and developers consume data from the firehose

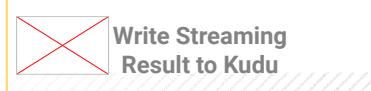
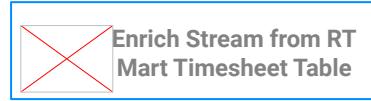
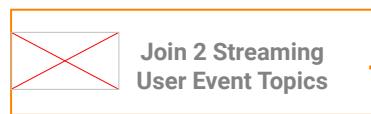


```
SELECT userid,
       max(amount) as max_amount,
       sum(amount) as sum_amount,
       count(*) as thecount,
       tumble_end(eventTimestamp, interval '5' second) as ts
  FROM authorizations
 GROUP BY userid, tumble(eventTimestamp, interval '5' second)
 HAVING count(*) > 1
```



```
[90]: import pandas as pd
[91]: mv = "https://xxxxxxxxxx"
[92]: df = pd.read_json(mv)
[93]: len(df.keys())
[93]: 5
[95]: df['ts'] = pd.to_datetime(df['ts'])
[97]: df.dtypes
[97]: max_amount          int64
       sum_amount          int64
       thecount            int64
       ts                  datetime64[ns]
       userid              int64
       dtype: object
[98]: df.set_index('userid').sort_values(by=['thecount'], ascending=False).head()
[98]:
      max_amount  sum_amount  thecount      ts
userid
    787      34911     57304     10 2020-06-16 19:52:15
    744      77407     95407      9 2020-06-16 19:52:15
    78      88761     330397      9 2020-06-16 19:52:15
    541      78762     282682      8 2020-06-16 19:52:15
    926      85636     129728      8 2020-06-16 19:52:15
```

Streaming ETL Data Pipeline Made Simple with SQL StreamBuilder



```
1 SELECT
2     geo_event.eventTimestamp, geo_event.driverId, geo_event.eventTime, geo_event.eventSource,
3     geo_event.truckId, geo_event.driverName, geo_event.routeId, geo_event.route, geo_event.eventType,
4     geo_event.latitude, geo_event.longitude, geo_event.correlationId, geo_event.geoAddress,
5     speed_event.speed,
6     driver.certified, driver.wage_plan,
7     timesheet.hours_logged, timesheet.miles_logged
8
9 FROM
10    geo_events_json AS geo_event
11    JOIN speed_events_json AS speed_event
12    ON (geo_event.driverId = speed_event.driverId)
13    LEFT JOIN CDP_Hive_Catalog.employees_hr_hive_db.driver
14    FOR SYSTEM_TIME AS OF PROCTIME() driver
15    ON driver.driverid = geo_event.driverId
16    LEFT JOIN `CDP_Kudu_Catalog`.`default_database`.`impala::employees_hr_kudu_impala_db.timesheet`
17    FOR SYSTEM_TIME AS OF PROCTIME() timesheet
18    ON (timesheet.driverid = geo_event.driverId AND timesheet.week = 1)
19    WHERE
20        geo_event.eventTimestamp BETWEEN
21            speed_event.eventTimestamp - INTERVAL '1' SECOND AND
22            speed_event.eventTimestamp + INTERVAL '1' SECOND
23        AND geo_event.eventType != 'Normal'
24        AND driver.wage_plan = 'hours'
25        AND timesheet.hours_logged > .45
```

Infer Kafka Tables from JSON, AVRO, ... Data

Kafka Table

Table Name *

Kafka Cluster *

Data Format *

Topic Name *

Schema Definition

```
1 {  
2   "type": "record",  
3   "name": "inferredSchema",  
4   "fields": [  
5     {  
6       "name": "bssid",  
7       "type": "string",  
8       "doc": "Type inferred from '\\"\\\"'"  
9     },  
10    {  
11      "name": "channel",  
12      "type": "string",  
13      "doc": "Type inferred from '\"52\\\"'"  
14    },  
15    {  
16      "name": "channel_band",  
17      "type": "string",  
18      "doc": "Type inferred from '\"5\\\"'"  
19    },  
20    {  
21      "name": "channel_width",  
22      "type": "string",  
23      "doc": "Type inferred from '\"80\\\"'"  
24    },  
25    {  
26      "name": "country_code",  
27      "type": "string",  
28      "doc": "Type inferred from '\\"\\\"'"  
29    },  
30    {  
31      "name": "interface",  
32      "type": "string",  
33      "doc": "Type inferred from '\"en0\\\"'"  
34    },  
35  }  
}
```

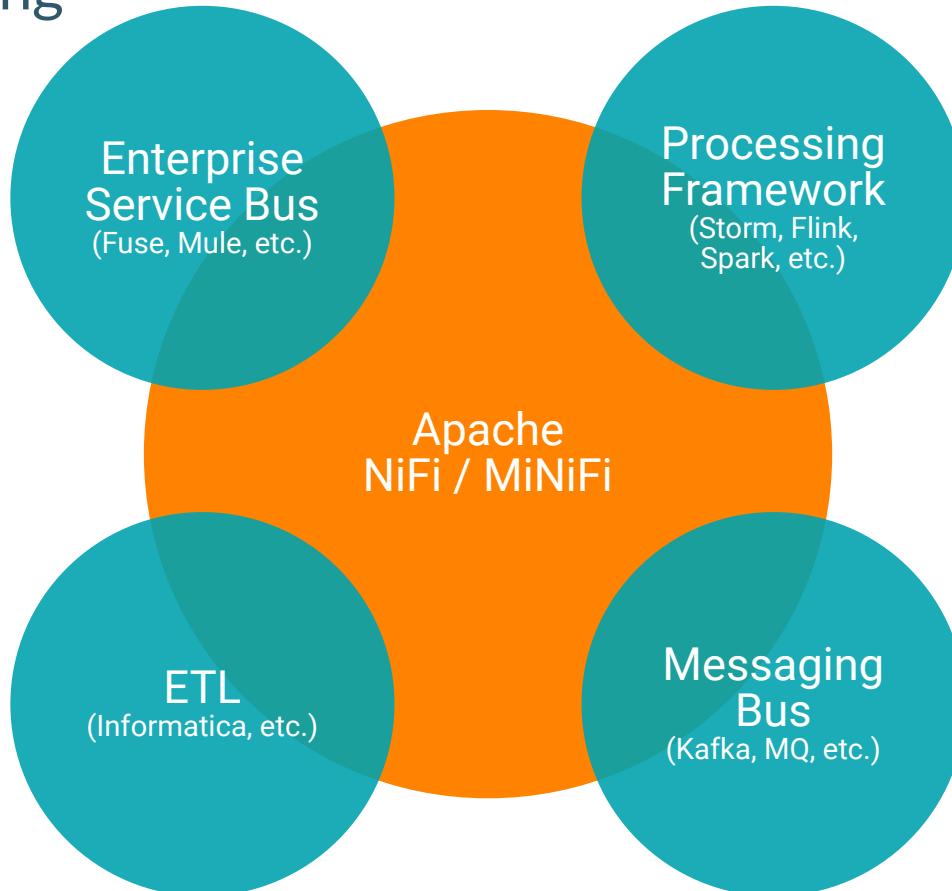
Event Time Data Transformation Properties Deserialization

Schema is valid

DATAFLOW APACHE NIFI

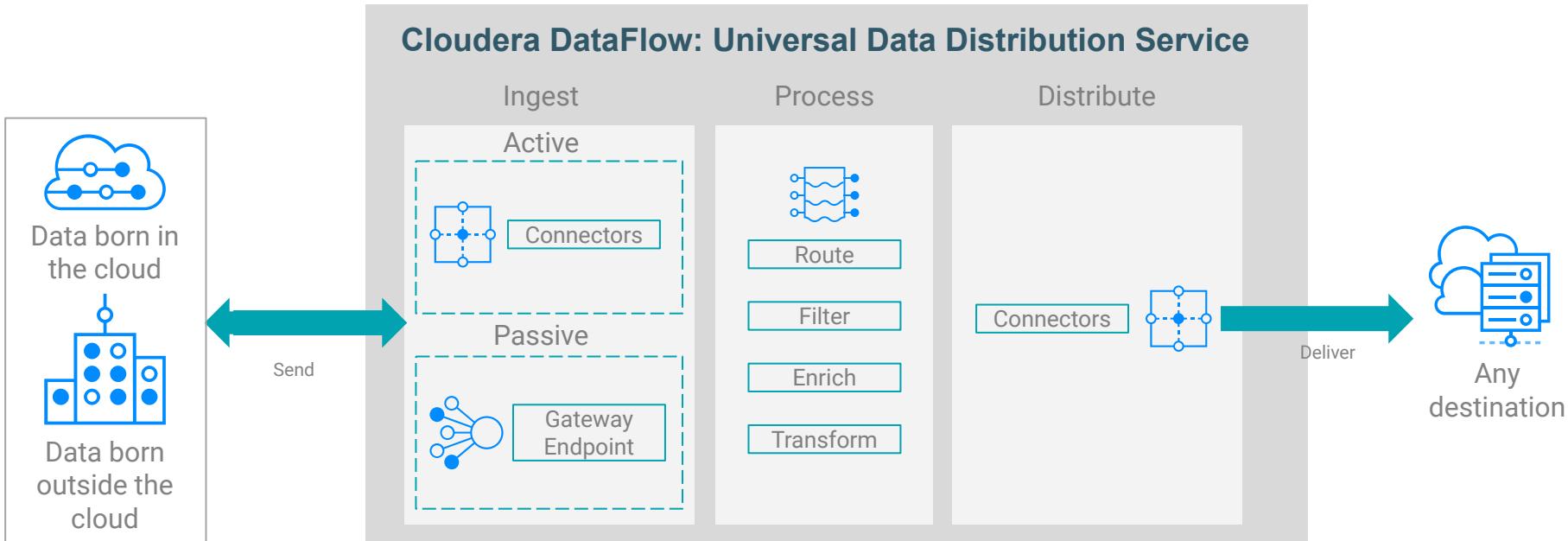


NiFi Positioning



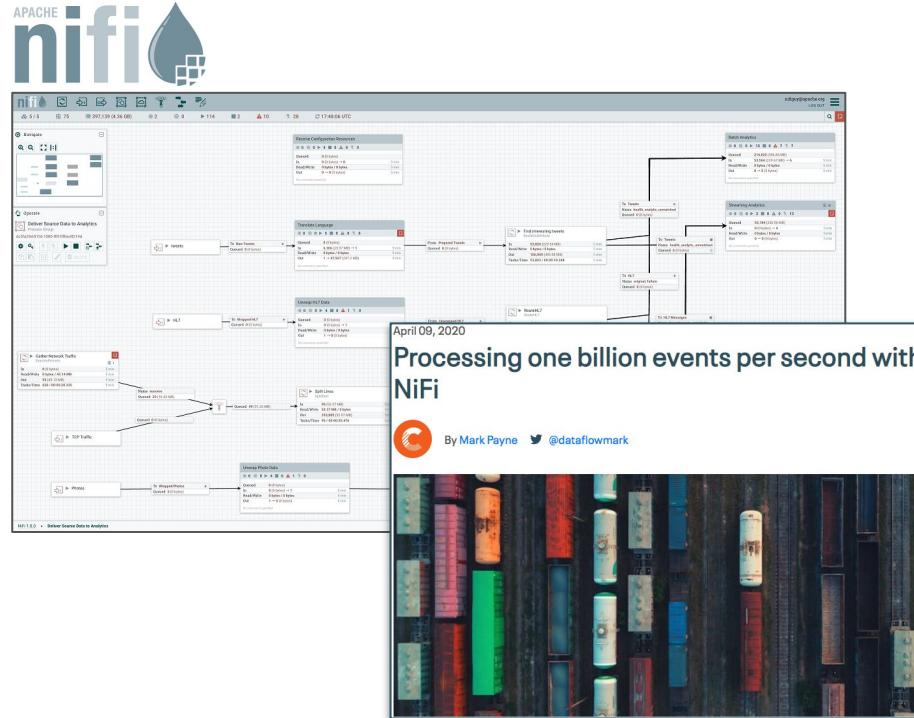
UNIVERSAL DATA DISTRIBUTION WITH CLOUDERA DATAFLOW (CDF)

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



CLOUDERA DATAFLOW - POWERED BY APACHE NiFi

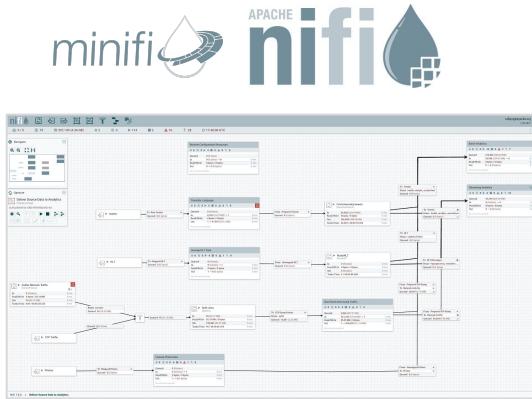
Ingest and manage data from edge-to-cloud using a no-code interface



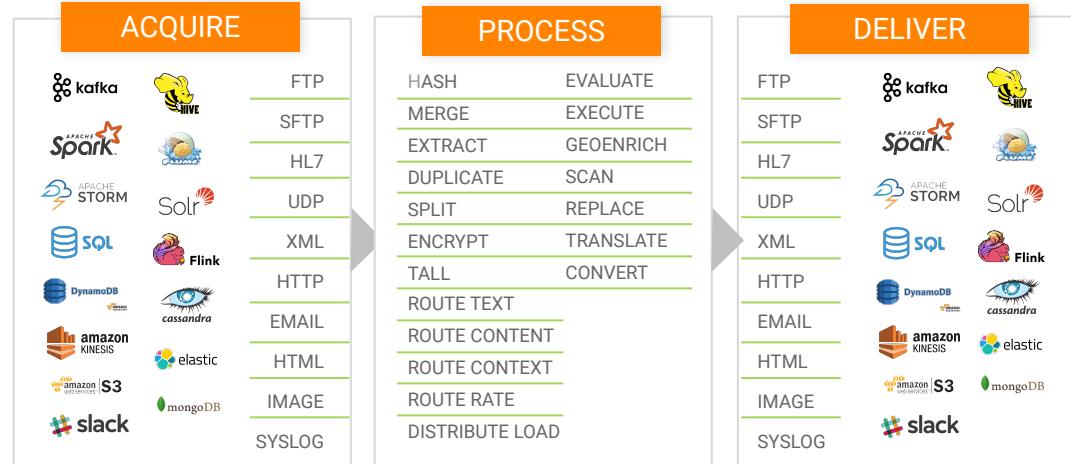
- #1 data ingestion/movement engine
 - Strong community
 - Product maturity over 11 years
 - Deploy on-premises or in the cloud
 - Over 400+ pre-built processors
 - Built-in data provenance
 - Guaranteed delivery
 - Throttling and Back pressure

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



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



PROVENANCE

Displaying 13 of 104
Oldest event available: 11/15/2016 13:34:50 EST

Showing the most recent events.

ConsumeKafka by component name

Date/Time	Type	FlowFile Uuid	Size	Component Name	Component Type
11/15/2016 13:35:03.8...	RECEIVE	379fc4f6-60e0-4151-9743-28...	44 bytes	ConsumeKafka	ConsumeKafka
11/15/2016 13:35:02.7...	RECEIVE	78f8c38b-89fc-4d00-a8d8-51...	44 bytes	ConsumeKafka	ConsumeKafka
11/15/2016 13:35:01.6...	RECEIVE	2bcd5124-bb78-489f-ad8a-7...	44 bytes	ConsumeKafka	ConsumeKafka

• Tracks data at each point as it flows through the system

• Records, indexes, and makes events available for display

• Handles fan-in/fan-out, i.e. merging and splitting data

• View attributes and content at given points in time

The diagram illustrates a data flow process. It starts with a red circle labeled "RECEIVE", which has an arrow pointing down to a grey circle labeled "JOIN". From the "JOIN" circle, an arrow points down to a grey circle labeled "DROP". A green arrow originates from the "RECEIVE" circle and points to a "Provenance Event" panel on the right. The "Provenance Event" panel contains three tabs: DETAILS, ATTRIBUTES, and CONTENT. The ATTRIBUTES tab is selected, showing the following attribute values:

Attribute	Value	Previously Set
filename	328717796819631	No value previously set
kafka.offset	44815	No value previously set
kafka.partition	6	No value previously set
kafka.topic	nifi-testing	No value previously set
path	/	No value previously set
uuid	328717796819631-0000-0000-0000-000000000000	No value previously set

EXTENSIBILITY

- Built from the ground up with extensions in mind
- Service-loader pattern for...
 - Processors
 - Controller Services
 - Reporting Tasks
 - Prioritizers
- Extensions packaged as NiFi Archives (NARs)
 - Deploy NiFi lib directory and restart
 - Same model as standard components

The screenshot shows the IntelliJ IDEA IDE interface. The top navigation bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help, and a battery icon. The title bar indicates the project is 'nifi-mxnetinference-processor'.

The left sidebar displays the project structure:

- Project
- nifi-mxnetinference-processor (Volumes/TS...)
- .idea
- data
- nifi-mxnetinference-nar
- target
- pom.xml
- nifi-mxnetinference-processors
- src
- main
- java
- com.dataflowdeveloper.mxnet
- InferenceProcessor 100% me
- Result 73% methods, 42% lir
- SSDClassifierService 100% m
- resources
- test
- java
- com.dataflowdeveloper.mxnet
- InferenceProcessorTest
- resources
- target
- test.jpg

The right side shows three processor configurations in a flow:

- InferenceProcessor** (LinkProcessor)
 - In: 0 (bytes)
 - Read/Write: 0 bytes / 31.45 KB
 - Out: 2 (31.45 KB)
 - Tasks/Time: 2 / 0:00:04.808
- UpdateAttribute** (UpdateAttribute)
 - In: 2 (31.45 KB)
 - Read/Write: 0 bytes / 0 bytes
 - Out: 2 (31.45 KB)
 - Tasks/Time: 2 / 0:00:00.005
- PutHDFS** (PutHDFS)
 - In: 2 (31.45 KB)
 - Read/Write: 31.45 KB / 0 bytes
 - Out: 0 (0 bytes)
 - Tasks/Time: 2 / 0:00:00.603

Annotations in the flow diagram include:

- Name success
- Queued 0 (0 bytes)

NiFi Load Balancing

- Improve NiFi cluster throughput
- Defined at connection level
- Configurable balancing strategies
- Critical for scale up paradigm in Kubernetes
- Alleviates S2S balancing “hack” customers use

The screenshot shows the NiFi interface with a flow editor. At the top, there's a 'DETAILS' tab and a 'SETTINGS' tab. Under 'DETAILS', you can see the flowfile's Name, Id (88cbd631-0166-1000-0000-00002af80f96), FlowFile Expiration (0 sec), Back Pressure Object Threshold (10000), and Size Threshold (1 GB). The 'Available Prioritizers' section lists 'FirstInFirstOutPrioritizer', 'NewestFlowFileFirstPrioritizer', 'OldestFlowFileFirstPrioritizer', and 'PriorityAttributePrioritizer'. The 'Selected Prioritizers' section is currently empty. Below these settings is a 'Load Balance Strategy' dropdown menu with options: 'Do not load balance', 'Partition by attribute', 'Round robin', and 'Single node'.

The flow editor displays two parallel connections. The first connection starts with a 'GenerateFlowFile' processor. Its configuration shows 'GenerateFlowFile 1.8.0.3.3.0.0-165 org.apache.nifi - nifi-standard-nar'. It has an 'In' port with 0 bytes and an 'Out' port with 42 (42 KB). The second connection starts with a 'LogAttribute' processor. Its configuration shows 'LogAttribute 1.8.0.3.3.0.0-165 org.apache.nifi - nifi-standard-nar'. It has an 'In' port with 41 (41 KB) and an 'Out' port with 0 (0 bytes).

Below the processors, a table lists flowfiles with their IDs, sizes, and destinations. The last three rows in this table are highlighted with a red box:

FlowFile	In	Out	Destination
0733ad94-3c80-44d7-9fc2-480caa...	0 (0 bytes)	1,024 bytes	LogAttribute
2bc7b5c1-c164-40fb-9e7e-da57884...	0 bytes / 42 KB	1,024 bytes	LogAttribute
80dece7a-15c8-4eb7-80ad-176bfe9...	42 (42 KB)	1,024 bytes	LogAttribute
98d9f9c4-bb47-4fe7-9786-964d027...	0 bytes / 0 bytes	1,024 bytes	LogAttribute
26c165ca-2f6d-4714-8c0c-e1de6e2...	0 bytes / 0 bytes	1,024 bytes	LogAttribute
8bfff920b-97a3-4b64-998d-046324a...	0 bytes / 0 bytes	1,024 bytes	LogAttribute
6345a326-4843-442e-b77d-480d20...	0 bytes / 0 bytes	1,024 bytes	LogAttribute
5fc30a5a-641e-4aa0-9c67-3b1d438...	0 bytes / 0 bytes	1,024 bytes	LogAttribute
1e90e7ee-92fe-47b3-9aa4-4094fa8a...	0 bytes / 0 bytes	1,024 bytes	LogAttribute

QUEUE CONFIGURATION

- **FlowFile Expiration** - Data that cannot be processed in a timely fashion can be automatically removed from the flow.
- **Back Pressure Thresholds** - Thresholds indicate how much data should be allowed to exist in the queue before the component that is the source of the Connection is no longer scheduled to run. This allows the system to avoid being overrun with data.
- **Load Balance Strategy** – Strategy to distribute the data in a flow across the nodes in the cluster. When enabled, compression can be configured on FlowFile contents and attributes.
- **Prioritization** – Determines the order in which flow files are processed.

Generate Syslog RFC5424	
ExecuteScript 1.13.2.2.2.0-127 org.apache.nifi - nifi-scripting-nar	
In	0 (0 bytes)
Read/Write	0 bytes / 0 bytes
Out	0 (0 bytes)
Tasks/Time	0 / 00:00:00.000
	5 min

Configure Connection

DETAILS SETTINGS

Name	success_Generate-FilterEvents	Available Prioritizers	FirstInFirstOutPrioritizer NewestFlowFileFirstPrioritizer OldestFlowFileFirstPrioritizer PriorityAttributePrioritizer
Id	64146cca-d197-3c27-9c47-015dd7b7a6c6	Selected Prioritizers	
FlowFile Expiration	0 sec	Load Balance Strategy	Round robin
Back Pressure Object Threshold	10000	Size Threshold	1 GB
Load Balance Compression	Do not compress		

RECORD-ORIENTED DATA WITH NIFI

- **Record Readers** - Avro, CSV, Grok, IPFIX, JSON1, JSON, Parquet, Scripted, Syslog5424, Syslog, WindowsEvent, XML
- **Record Writers** - Avro, CSV, FreeFromText, Json, Parquet, Scripted, XML
- Record Reader and Writer support referencing a schema registry for retrieving schemas when necessary.
- Enable processors that accept any data format without having to worry about the parsing and serialization logic.
- Allows us to keep FlowFiles larger, each consisting of multiple records, which results in far better performance.

Filter Events		
QueryRecord 1.13.2.2.2.2.0-127 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

Configure Processor

SETTINGS	SCHEDULING	PROPERTIES	COMMENTS
Required field			
Property	Value		
Record Reader	CSVReader	→	
Record Writer	JsonRecordSetWriter	→	

RUNNING SQL ON FLOWFILES

- Evaluates one or more SQL queries against the contents of a FlowFile.
- This can be used, for example, for field-specific filtering, transformation, and row-level filtering.
- Columns can be renamed, simple calculations and aggregations performed.
- The SQL statement must be valid ANSI SQL and is powered by Apache Calcite.

Filter Events		
QueryRecord 1.13.2.2.2.2.0-127 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

Configure Processor | QueryRecord 1.13.2.2.2.2.0-127

Stopped

SETTINGS SCHEDULING PROPERTIES COMMENTS

Required field

Property	Value
Record Reader	Syslog_5424_Reader
Record Writer	JSON_Syslog_5424_Writer
Include Zero Record FlowFiles	false
Cache Schema	false
Default Decimal Precision	10
Default Decimal Scale	0
filtered_events	#(Filter Rule)

Apache NiFi with Python Custom Processors

Python as a 1st class citizen

```
import cv2
import numpy as np
import json
from nifiapi.properties import PropertyDescriptor
from nifiapi.properties import ResourceDefinition
from nifiapi.flowfiletransform import FlowfiletransformResult

SCALE_FACTOR = 0.00392
NMS_THRESHOLD = 0.4 # non-maximum suppression threshold
CONFIDENCE_THRESHOLD = 0.5

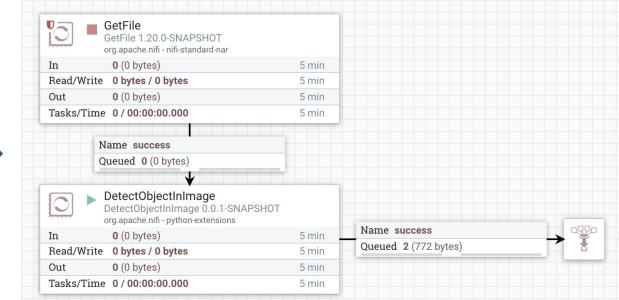
class DetectObjectInImage:
    class Java:
        implements = ['org.apache.nifi.python.processor.FlowfileTransform']
        class ProcessorDetails:
            version = '0.0.1-SNAPSHOT'
            dependencies = ['numpy >= 1.23.5', 'opencv-python >= 4.6']

    def __init__(self, jvm=None, **kwargs):
        self.jvm = jvm

        # Build Property Descriptors
        self.model_file = PropertyDescriptor(
            name = 'Model File',
            description = 'The binary file containing the trained Deep Neural Network weights. Supports Caffe (*.caffemodel), TensorFlow (*.pb), Torch (*.t7, *.net), Darknet (*.weights), ' +
                        'OLDY (*.bin), and ONNX (*.onnx)',
            required = True,
            resource_definition = ResourceDefinition(allow_file = True)
        )
        self.config_file = PropertyDescriptor(
            name = 'Network Config File',
            description = 'The text file containing the Network configuration. Supports Caffe (*.prototxt), TensorFlow (*.pbtxt), Darknet (*.cfg), and DLDT (*.xml)',
            required = False,
            resource_definition = ResourceDefinition(allow_file = True)
        )
        self.class_name_file = PropertyDescriptor(
            name = 'Class Names File',
            description = 'A text file containing the names of the classes that may be detected by the model. Expected format is one class name per line, new-line terminated.',
            required = True,
            resource_definition = ResourceDefinition(allow_file = True)
        )
        self.descriptors = [self.model_file, self.config_file, self.class_name_file]

    def getPropertyDescriptors(self):
        return self.descriptors

    def onScheduled(self, context):
        # read class names from text file
        class_name_file = context.getProperty(self.class_name_file.name).getValue()
        if class_name_file is None:
```



https://www.youtube.com/watch?v=9Oj_6nFmbPg

<https://github.com/apache/nifi/blob/614947e4ac6798ad80817e82514c39349d5faacb/nifi-docs/src/main/asciidoc/python-developer-guide.adoc>

Add Processor

Source

Displaying 13 of 333

beta

all groups

amazon attributes
aws azure cloud
database fetch get
google hadoop ingest
json listen logs
message microsoft
put python record
restricted source
storage test text
update

Type ▾	Version	Tags
ConvertCsvToExcel	0.0.1-SNAPSHOT	excel, python, test, csv, beta
DetectObjectInImage	0.0.1-SNAPSHOT	python, test, beta
GenerateRecord	0.0.1-SNAPSHOT	python, test, beta
LogContents	0.0.1-SNAPSHOT	python, test, beta
LookupAddress	0.0.1-SNAPSHOT	python, test, beta
PopulateRecord	0.0.1-SNAPSHOT	python, test, beta
PrettyPrintJson	0.0.1-SNAPSHOT	python, test, beta
SetRecordField	0.0.1-SNAPSHOT	python, test, beta
WriteMessage	0.0.1-SNAPSHOT	python, test, beta
WriteMessage	0.0.2-SNAPSHOT	python, test, beta
WriteNumber	0.0.1-SNAPSHOT	python, test, beta
WriteNumpyVersion	0.0.1-SNAPSHOT	python, test, beta

ConvertCsvToExcel 0.0.1-SNAPSHOT org.apache.nifi - python-extensions

Converts a CSV file into a Microsoft Excel file

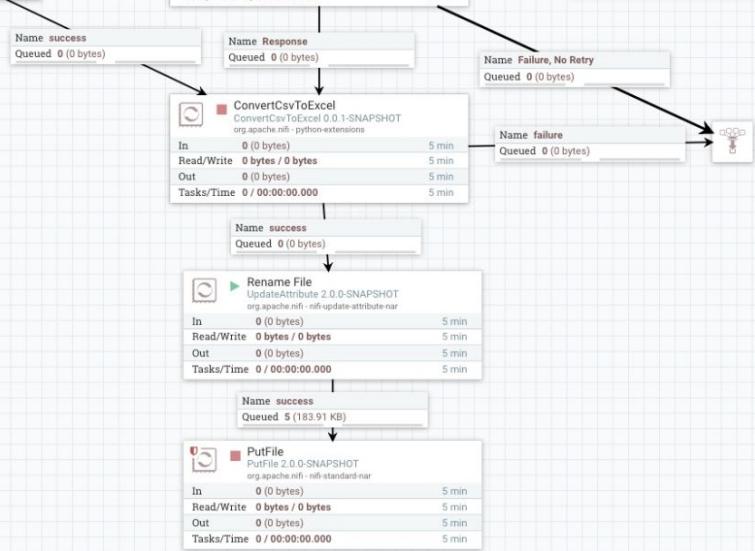
CANCEL

ADD



Generate CSV Record
GenerateRecord 2.0.0-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



GenerateFlowFile
GenerateFlowFile 2.0.0-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

GenerateRecord
GenerateRecord 2.0.0-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

GenerateRecord
GenerateRecord 2.0.0-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

About Apache NiFi



2.0.0-SNAPSHOT
02/09/2023 22:16:12 EST
Tagged nifi-1.15.0-RC3

Apache NiFi is a framework to support highly scalable and flexible dataflows. It can be run on laptops up through clusters of enterprise class servers. Instead of dictating a particular dataflow or behavior it empowers you to design your own optimal dataflow tailored to your specific environment.



ReadyFlow Gallery

Iceberg X

Added

Kafka to Iceberg

Version 1

Consumes JSON, CSV or Avro events from Kafka and writes them as Parquet files to a destination Iceberg table.

[View Added Flow Definition](#) [Create New Draft](#)

READYFLOW GALLERY

- Cloudera provided flow definitions
- Cover most common data flow use cases
- Optimized to work with CDP sources/destinations
- Can be deployed and adjusted as needed

ReadyFlow Gallery

Search by name

Added

Kafka filter to Kafka Version 1

Consumes JSON, CSV or Avro events from Kafka, filters them before writing them back to Kafka as JSON, CSV or Avro.

[View Added Flow Definition](#)

Added

Kafka to Cloudera Operational Database Version 1

Consumes JSON, CSV or Avro events from Kafka and ingests them into Cloudera Operational Database (COD).

[View Added Flow Definition](#)

Kafka to Kafka Version 1

Consumes events from Kafka and writes them to another Kafka topic.

[Add To Catalog](#)

Kafka to Kudu Version 1

Consumes JSON, CSV or Avro events from Kafka and ingests them into Kudu.

[Add To Catalog](#)

Kafka to S3 Avro Version 1

Consumes JSON, CSV or Avro events from Kafka and writes Avro files to S3.

[View Added Flow Definition](#)

S3 to S3 Avro Version 1

Consumes JSON, CSV or Avro files from source S3 location and writes Avro files to a destination S3 location.

[Add To Catalog](#)

FLOW CATALOG

- Central repository for flow definitions
- Import existing NiFi flows
- Manage flow definitions
- Initiate flow deployments

The screenshot shows the Cloudera DataFlow interface with the 'Catalog' tab selected. The main area is titled 'Flow Catalog' and contains a table of flow definitions. A search bar at the top allows users to 'Search by name'. A blue button labeled 'Import Flow Definition' is located in the top right corner. The table has columns for Name, Type, Versions, and Last Updated. Each row in the table includes a blue 'View' icon (represented by a right-pointing arrow) for further details. The table lists ten flow definitions, all of which are 'Custom Flow Definition' type. The last updated column shows various dates from 'a day ago' to '21 days ago'. At the bottom of the catalog table, there are pagination controls for 'Items per page' (set to 10), a page number '1 - 10 of 10', and navigation icons for previous and next pages.

Name ↑	Type	Versions	Last Updated	
cc_fraud_template_int101run	Custom Flow Definition	2	a day ago	>
cc_fraud_template_int101run2	Custom Flow Definition	1	9 days ago	>
JSON_Kafka_To_Avro_S3	Custom Flow Definition	2	a day ago	>
Kafka filter to Kafka	ReadyFlow	1	2 days ago	>
Kafka to Cloudera Operational Database	ReadyFlow	1	2 days ago	>
Kafka to S3 Avro	ReadyFlow	1	14 hours ago	>
nifi_flows	Custom Flow Definition	1	2 months ago	>
Weather Data Flow	Custom Flow Definition	1	a day ago	>
Weather_Data	Custom Flow Definition	1	15 days ago	>
Weather_JSON_Kafka_To_Avro_S3	Custom Flow Definition	1	21 days ago	>

DEPLOYMENT WIZARD

- Turns flow definitions into flow deployments
- Guides users through providing required configuration
- Choose NiFi runtime version
- Pick from pre-defined NiFi node sizes
- Define KPIs for the deployment

Start Deployment Wizard

dataflow-demo-new / New Flow Deployment

Overview

Selected Flow Definition

NAME	Machine Data To Warehouse	VERSION	3
------	---------------------------	---------	---

Target Environment

NAME	dataflow-demo-new
------	-------------------

NiFi Runtime Version

CURRENT VERSION	Latest Version (1.14.0.2.3.1.0-3)	Change Version
-----------------	-----------------------------------	----------------

Deployment Name

Provide Parameters

Flow Parameters

Data entered here never leaves the environment in your cloud account. Provide parameter values directly in the text input or upload a file for parameters that expect a file.

MachineData

AWS Credential File

Select File

Drop file or browse

CDP Truststore

Select File

Drop file or browse

CDPSchemaRegistry

<https://dataflow-streams-master0.dataflow.xcu2-8y8x.dev.cldr.work:7790/api/v1>

Configure Sizing & Scaling

Overview

Flow Parameters

Sizing & Scaling

Key Performance Indicators

Review

Sizing & Scaling

Select the NiFi node size and the number of nodes provisioned for your flow.

NiFi Node Sizing

<input checked="" type="radio"/> Extra Small	<input type="radio"/> Small	<input type="radio"/> Medium	<input type="radio"/> Large
2 vCores Per Node 4 GB Per Node	4 vCores Per Node 8 GB Per Node	8 vCores Per Node 16 GB Per Node	16 vCores Per Node 32 GB Per Node

Number of NiFi Nodes

Auto Scaling

Enabled

Min. Nodes: 1

Max. Nodes: 3

Define KPIs

Key Performance Indicators

Set up KPIs to track specific performance metrics of a deployed flow. Click and drag to reorder how they are displayed.

Entire Flow

METRIC TO TRACK: Data In

ALERT SET: Notify if less than 150 KB/sec, for at least 30 seconds.

Processor: Write to S3 using HDFS proc

METRIC TO TRACK: Bytes Sent

ALERT SET: No alert set

Add New KPI

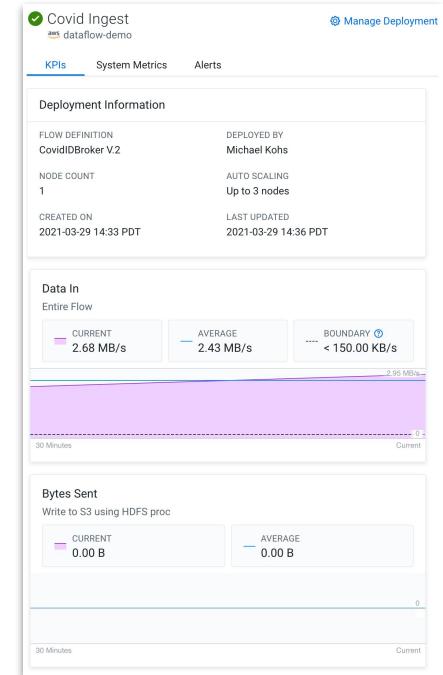
KEY PERFORMANCE INDICATORS

- Visibility into flow deployments
- Track high level flow performance
- Track in-depth NiFi component metrics
- Defined in Deployment Wizard
- Monitoring & Alerts in Deployment Details

KPI Definition in Deployment Wizard

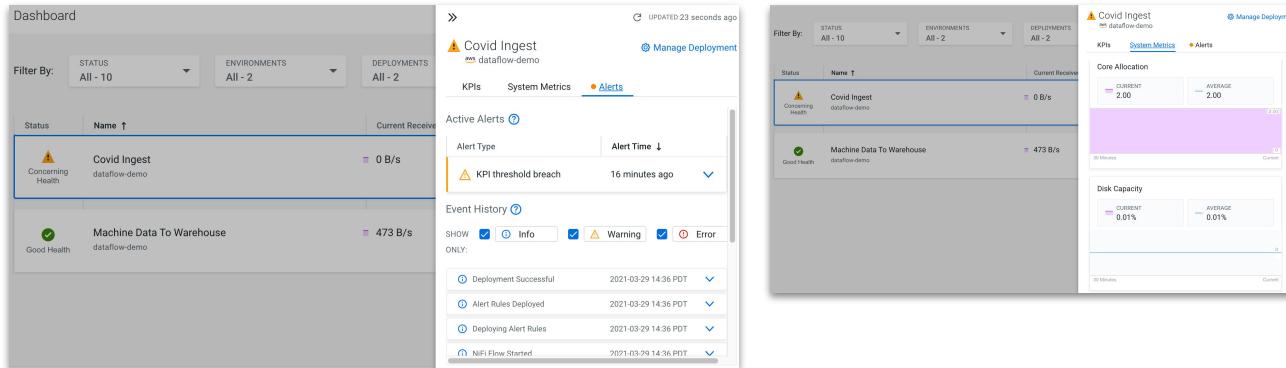
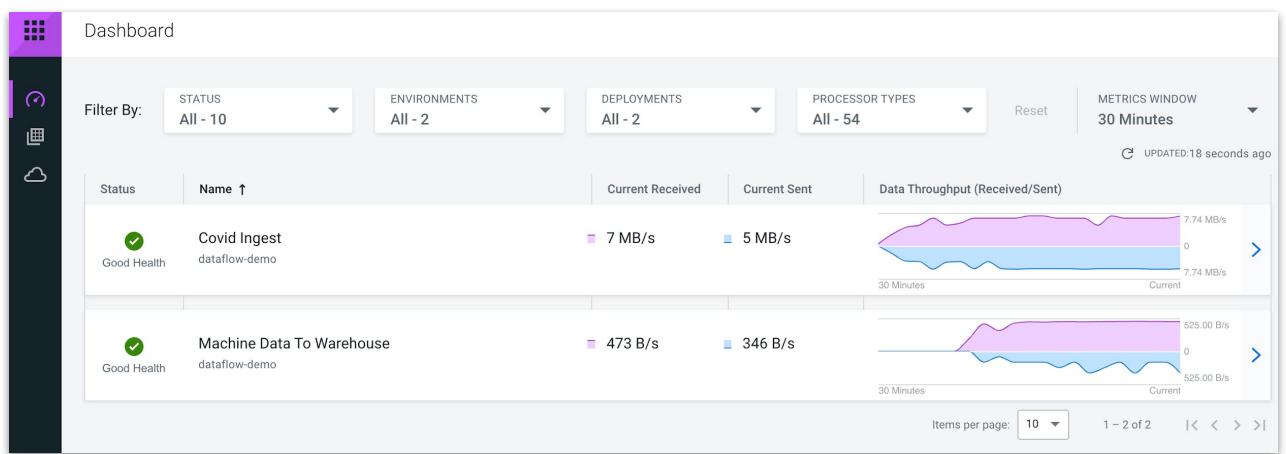
The screenshot shows the 'New Deployment' step of the deployment wizard. On the left, a sidebar lists steps: Overview, NiFi Configuration, Parameters, Sizing & Scaling, Key Performance Indicators (selected), and Review. The main area is titled 'Key Performance Indicators' with the sub-section 'Entire Flow'. It shows 'METRIC TO TRACK' set to 'Flow Files Queued', 'ALERT SET' as 'No alert set', and a note: 'Notify if outside the range of 999 MB/sec - 1 MB/sec, for at least 5 minutes.' A button at the bottom right says '(+) Add New KPI'.

KPI Monitoring



DASHBOARD

- Central Monitoring View
- Monitors flow deployments across CDP environments
- Monitors flow deployment health & performance
- Drill into flow deployment to monitor system metrics and deployment events



DEPLOYMENT MANAGER

- Manage flow deployment lifecycle
(Suspend/Start/Terminate)
- Add/Edit KPIs
- Change sizing configuration
- Update parameters
- Change NiFi version of the deployment
- Gateway to NiFi canvas

Dashboard / dataflow-demo-new / Kafka to COD

REFRESHED 12 seconds ago

Actions ▾

Deployment Manager

Status: Good Health

Deployment Name: Kafka to COD

Flow Definition: Kafka to Cloudera Operational Database V1

Deployed By: Michael Kohs

Node Count: 1

Auto Scaling: Disabled

Created On: 2021-07-26 17:05 PDT

Last Updated: 2021-07-26 17:07 PDT

Environment: dataflow-demo-new

Region: US West (Oregon)

NIFI Runtime Version: 1.14.0.2.3.0.0-89

Deployment Settings

KPIs and Alerts Sizing and Scaling Parameters

Parameters

Running Processors that are affected by the Parameter changes will automatically be restarted.

Data entered here never leaves the environment in your cloud account. Provide parameter values directly in the text input or upload a file for parameters that expect a file.

The selected flow definition references an external Default NiFi SSL Context Service. Hence, DataFlow will automatically create a matching SSL Context Service with a keystore and truststore generated from the target environment's FreeIPA certificate.

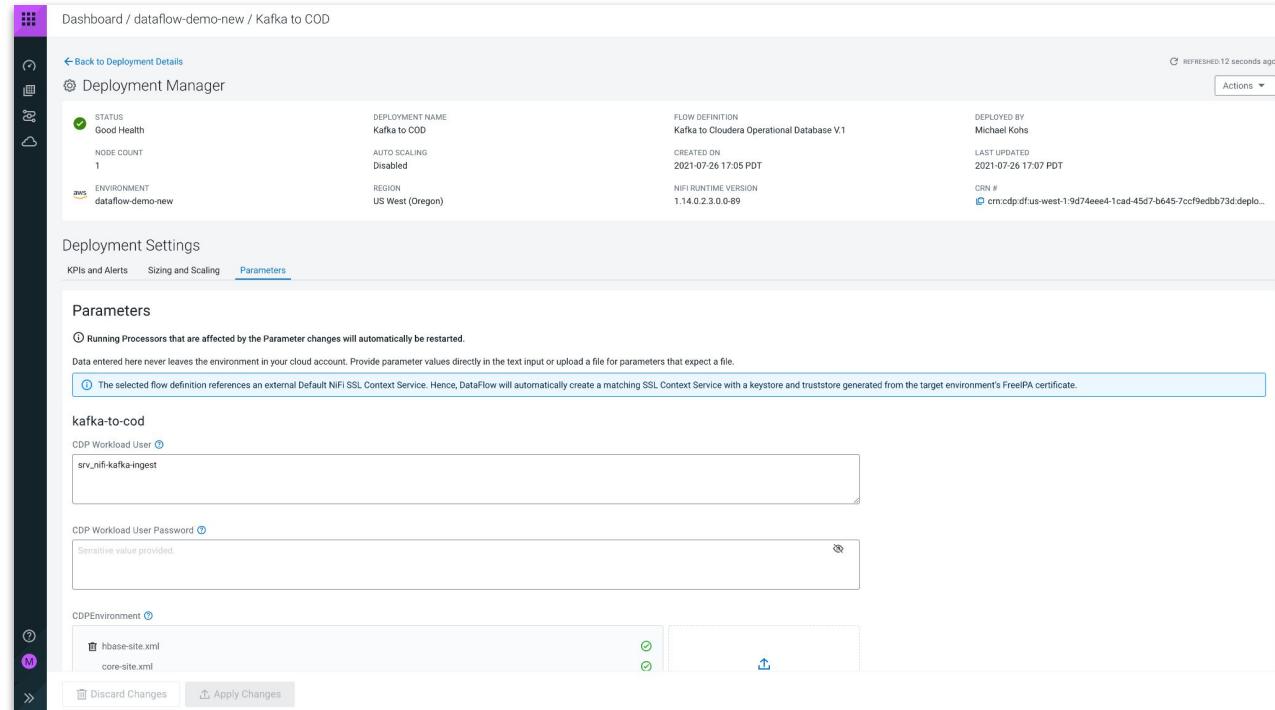
kafka-to-cod

CDP Workload User:

CDP Workload User Password:

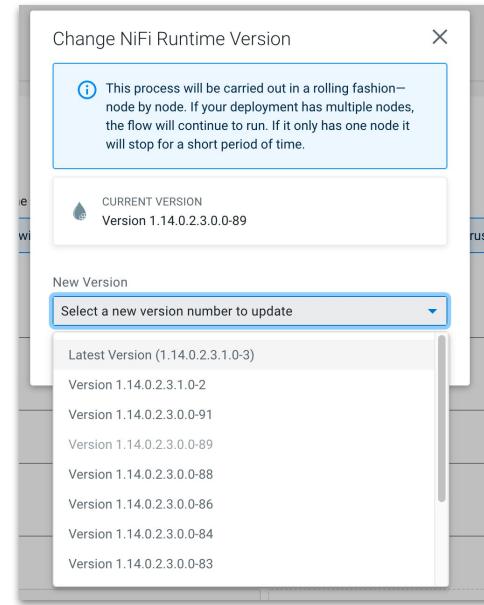
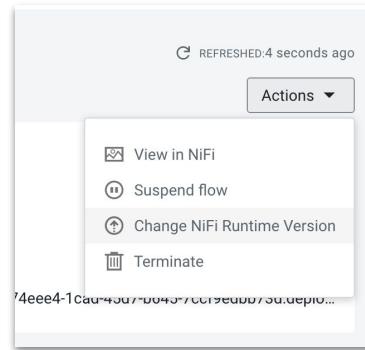
CDP Environment: hbase-site.xml core-site.xml

Discard Changes Apply Changes



NIFI VERSION UPGRADES

- Pick up NiFi hotfixes easily
- Upgrade (or downgrade) the hotfix version of existing deployments
- Rolling upgrade (if the deployment has >1 NiFi nodes)



Apache NiFi - Release History Review

- Soon: NiFi 2.0
- Apr 23: 1.21
- Feb 23: 1.20
 - Non-stop security improvements, vulnerability resolution, library updates. “Security by default”. Secure Zookeeper.
- Dec 22: 1.19
- Oct 22: 1.18
- Aug 22: 1.17
- Mar 22: 1.16
 - Key capabilities: all repository on-disk encryption, classloader isolation for native libs, empty all queues, prometheus, Azure AD authz, ‘Run Once’, Framework level retries, Stateless NiFi in NiFi, Parameter Contexts with inheritance and sensitive values, interactive component validation, generate fake records, scripted record transforms
- Dec 21: 1.15
- Jul 21: 1.14
- Mar 21: 1.13
 - New connectors for: IBM MQ, GCP, AWS, Azure, AMQP, MQTT, CDC/MySQL, Kudu, CEF, Geo Hashing, SNMP Traps, SMB, Airtable, Dropbox, Box, Hubspot, Workday, Zendesk, Iceberg, Snowflake, Salesforce, Google Drive
- Aug 20: 1.12
- Jan 20: 1.11

NiFi 2.0 - Quick Look

- Moves to Java 11 and Java 17 as the base
- JSON for flow serialization
 - XML and Templates concepts gone
- Parameter Contexts for easier SDLC/Deployment
 - Variables removed.
- Technical Debt Reduction
 - Delete a bunch of legacy components.
- Focus on timer and cron based scheduling
 - Event driven scheduling removed / unnecessary.

Key / Common Themes Of Change beyond the framework

Stateless NiFi engine

- Faster execution (no disk)
- End to end transactional
- Run in FaaS or NiFi or command line

Record Oriented Processors

- The proven best practice pattern for handling any 'record' pattern
- Same proven processors - just bring your format/schema

Scripting / Python

- Data Engineers and Data Science user base growing
- Existing script processors always improving - heavy use.

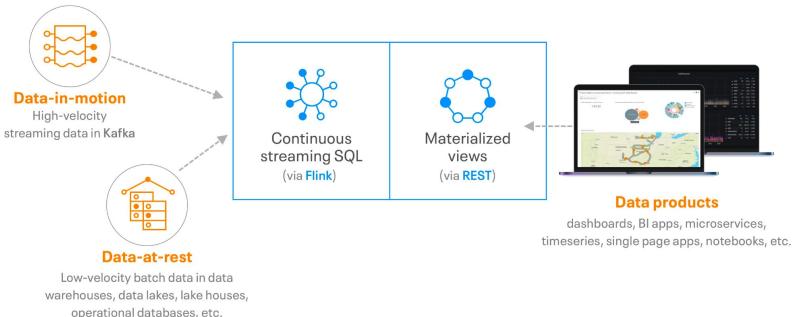
All the connectors!

- Apache NiFi includes **362** processors by default with more available in Maven
- 132 controller services
- 18 Reporting Tasks

BEST PRACTICES

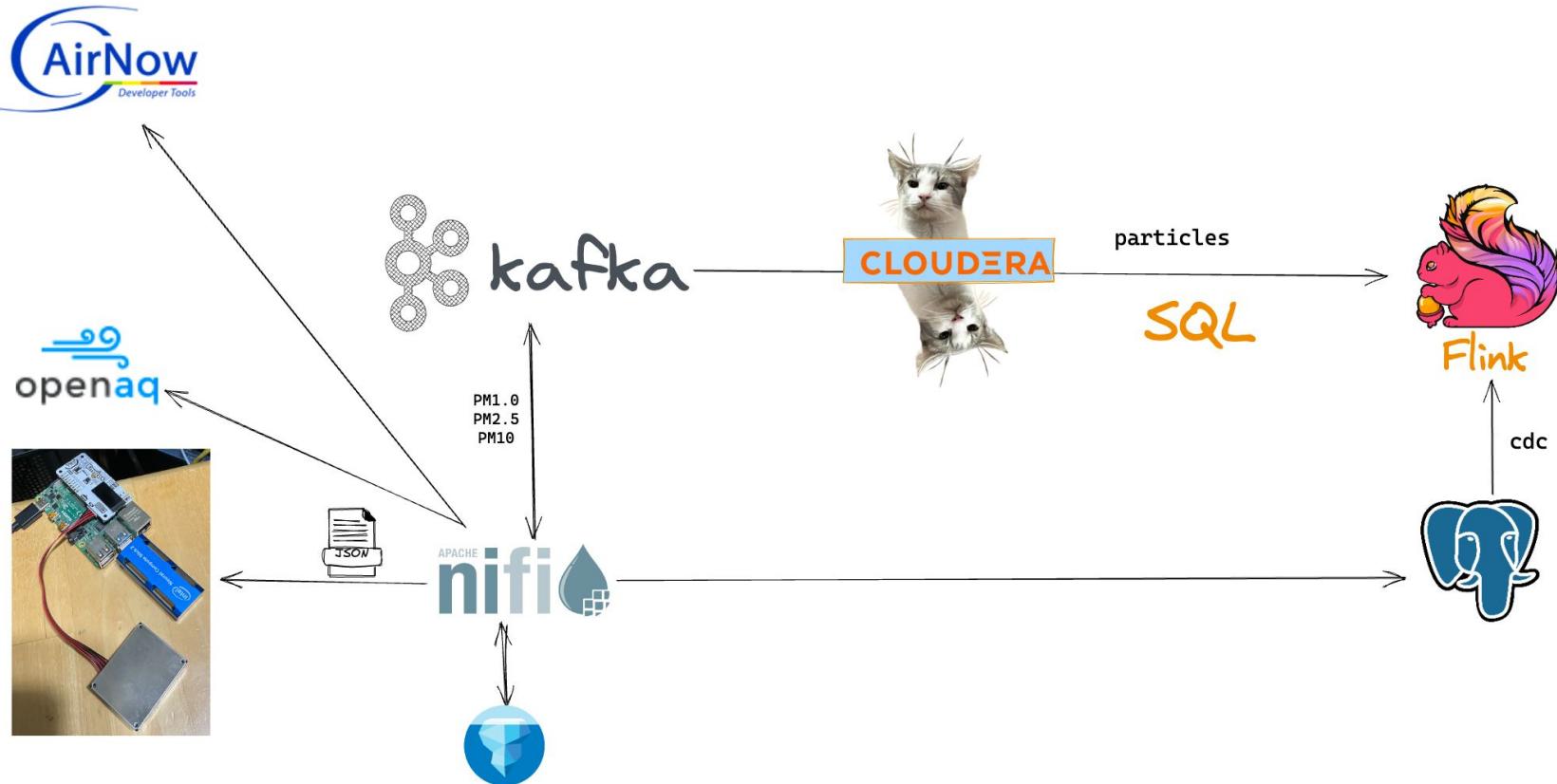
STREAMING TECH DEBT TIPS

- Version Control All Assets
- Managed Public Cloud like Cloudera
- Use DevOps and APIs
- Latest Java and Python
- Stream Sizing (NiFi, Kafka, Flink)



EXAMPLES

Particulate Matter Sensor Pipeline

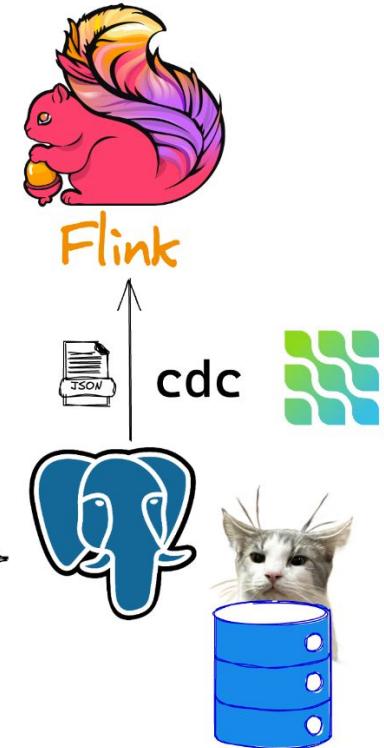


CDC with Debezium and Flink

SQL Stream Builder with Flink SQL



PutDatabaseRecord

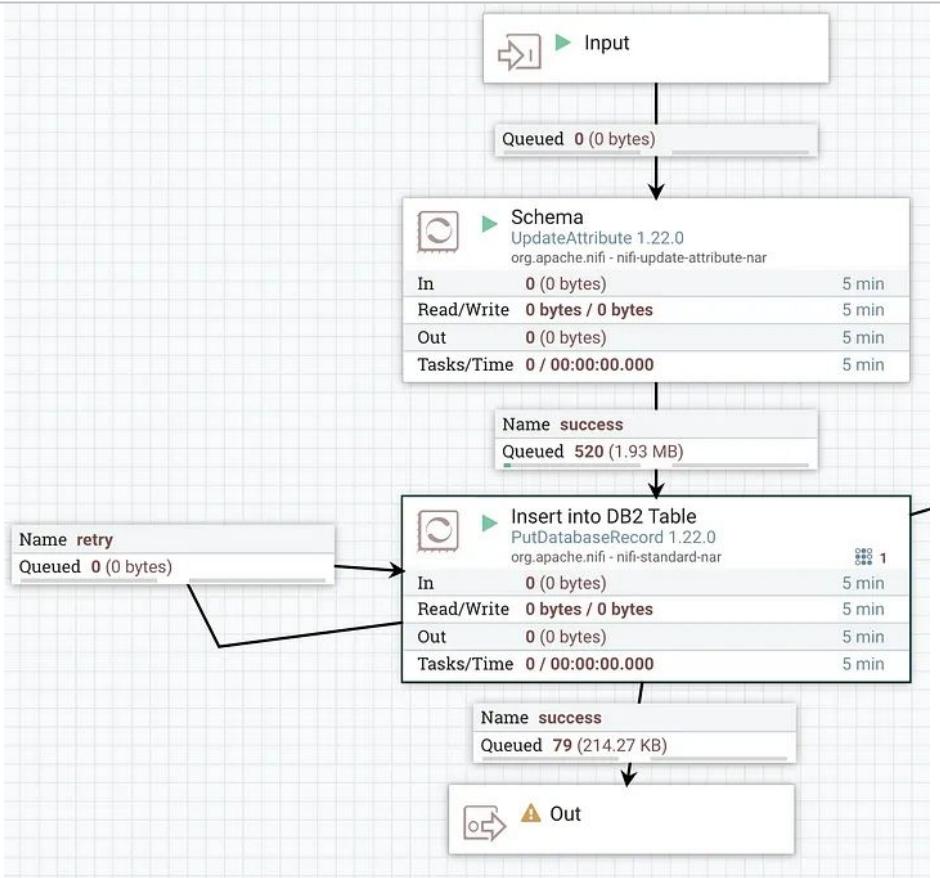
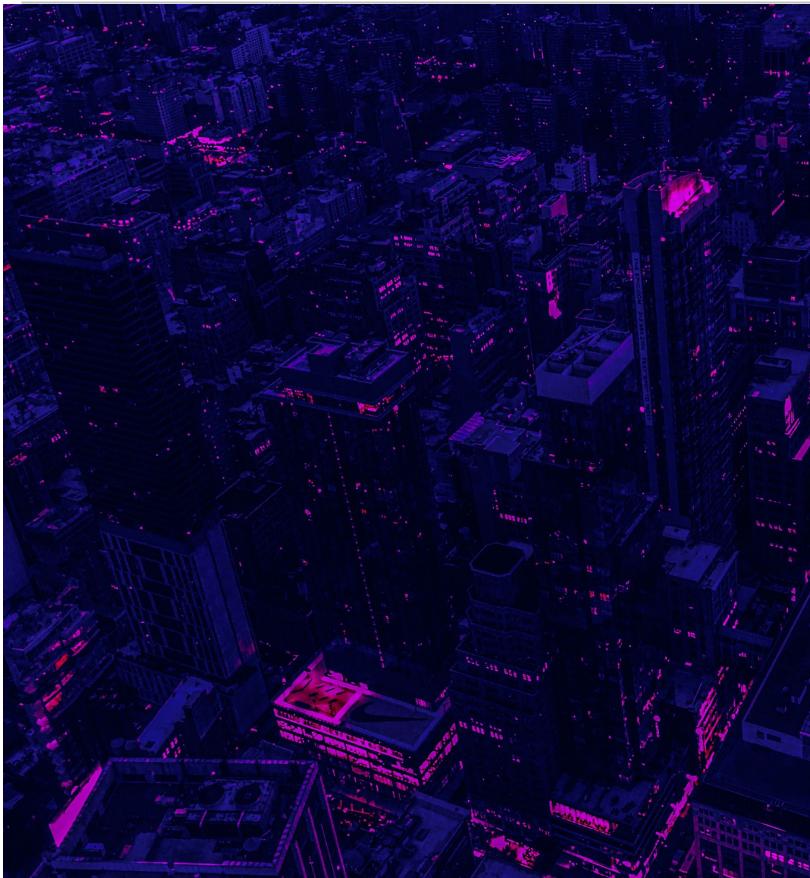


<https://docs.cloudera.com/csa/1.10.0/how-to-ssb/topics/csa-ssb-cdc-connectors.html>

CDC with Debezium and Kafka

Kafka Connect





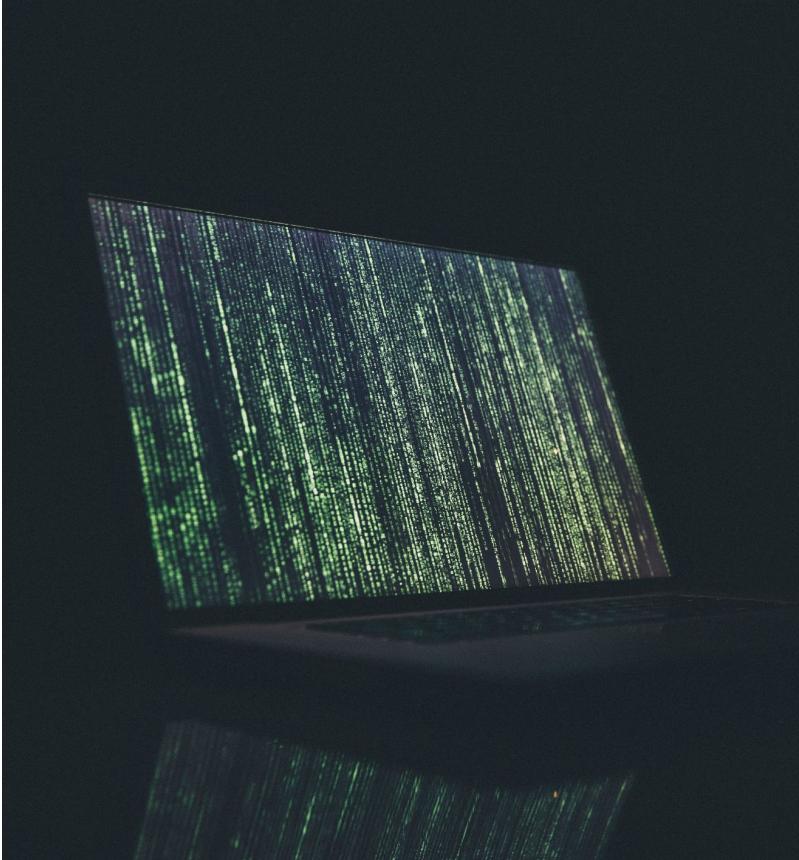
Configure Processor | PutDatabaseRecord 1.22.0

■ Stopped

SETTINGS SCHEDULING PROPERTIES RELATIONSHIPS COMMENTS

Required field

Property	Value
Record Reader	JsonTreeReaderSchema
Database Type	Generic
Statement Type	INSERT
Data Record Path	No value set
Database Connection Pooling Service	ibmdb2cluster
Catalog Name	No value set
Schema Name	DB2INST1
Table Name	TRAVELADVISORY
Translate Field Names	true
Unmatched Field Behavior	Ignore Unmatched Fields
Unmatched Column Behavior	Ignore Unmatched Columns
Quote Column Identifiers	false





Controller Service Details | DBCPConnectionPool 1.22.0

SETTINGS PROPERTIES COMMENTS

Required field

Property	Value
Database Connection URL	jdbc:db2://192.168.1.157:50000/SAMPLE
Database Driver Class Name	com.ibm.db2.jcc.DB2Driver
Database Driver Location(s)	/Users/tspann/Downloads/db2/db2jcc4.jar
Kerberos User Service	No value set
Kerberos Credentials Service	No value set
Kerberos Principal	No value set
Kerberos Password	No value set
Database User	db2inst1
Password	Sensitive value set
Max Wait Time	500 millis
Max Total Connections	8
Validation query	No value set
Minimum Idle Connections	0
Max Idle Connections	8



Processor Details

▶ Running STOP & CONFIGURE

SETTINGS SCHEDULING PROPERTIES RELATIONSHIPS COMMENTS

Required field

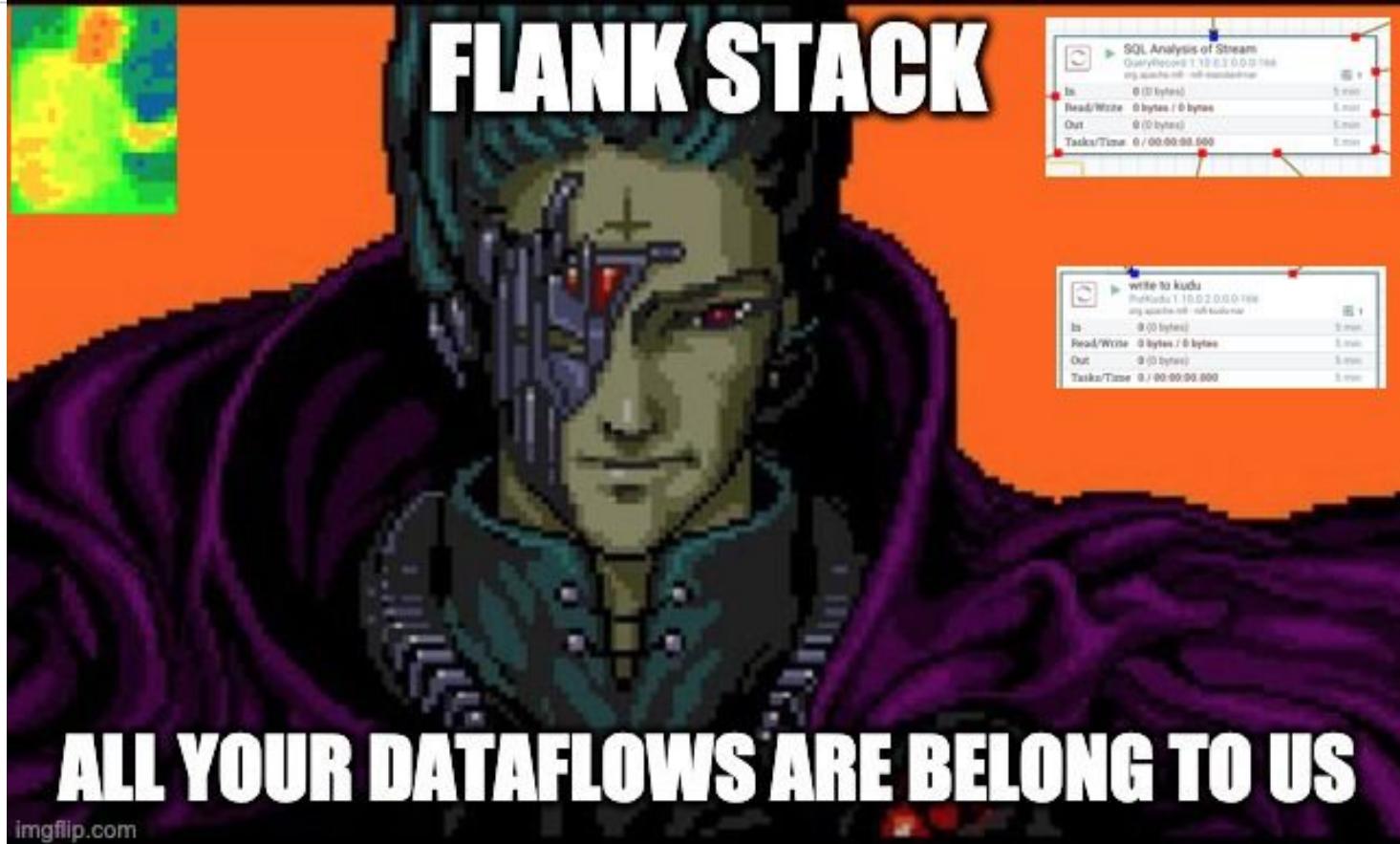
Property	Value
Record Reader	CDP Infer JsonTreeReader →
Database Type	Oracle 12+
Statement Type	INSERT
Data Record Path	No value set
Database Connection Pooling Service	Oracle12DBCPConnectionPool →
Catalog Name	No value set
Schema Name	TSPANN
Table Name	NEWJERSEYBUS
Translate Field Names	true
Unmatched Field Behavior	Ignore Unmatched Fields
Unmatched Column Behavior	Ignore Unmatched Columns
Quote Column Identifiers	false

OK

RESOURCES AND WRAP-UP

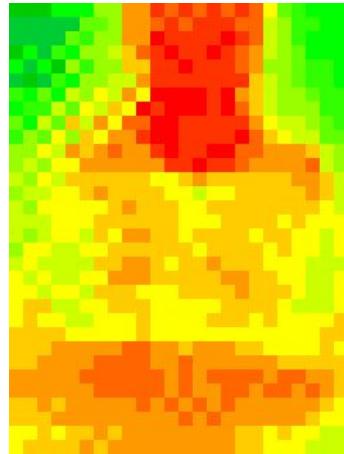
Resources



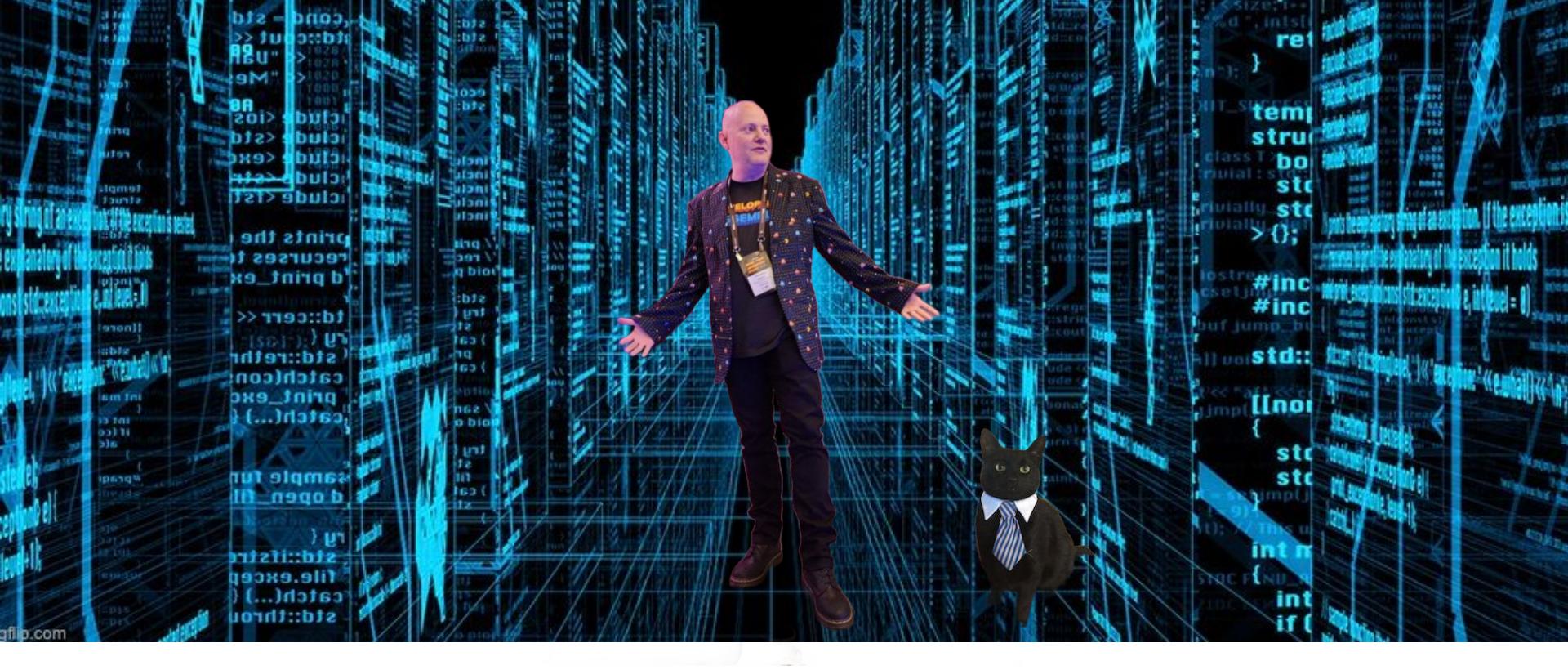


Upcoming Events

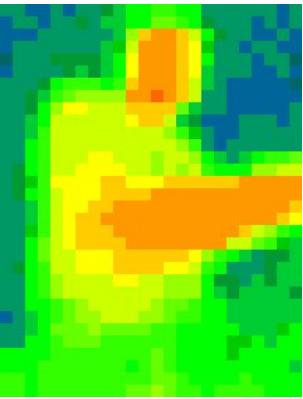
Nov 22



<https://events.pinetool.ai/3079/#sessions/101077>



<https://medium.com/@tspann/cdc-not-cat-data-capture-e43713879c03>



TH^ON^G Y^OU[★]

