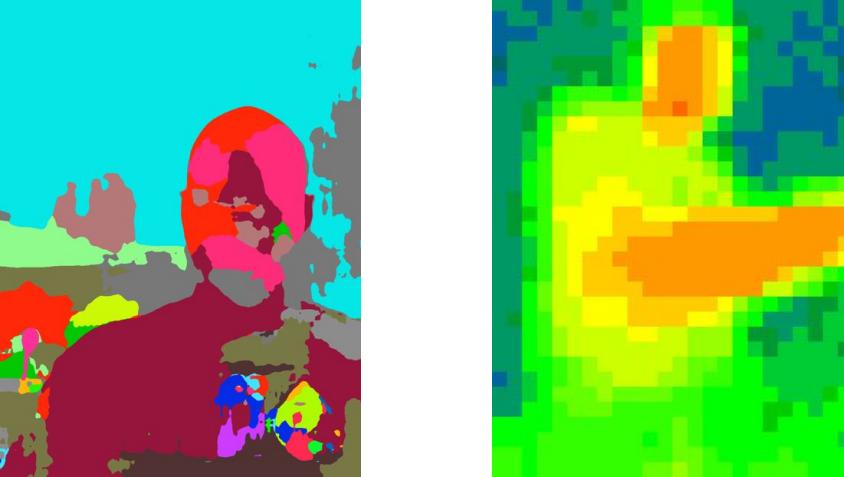


Real-Time Streaming Processing Round Table

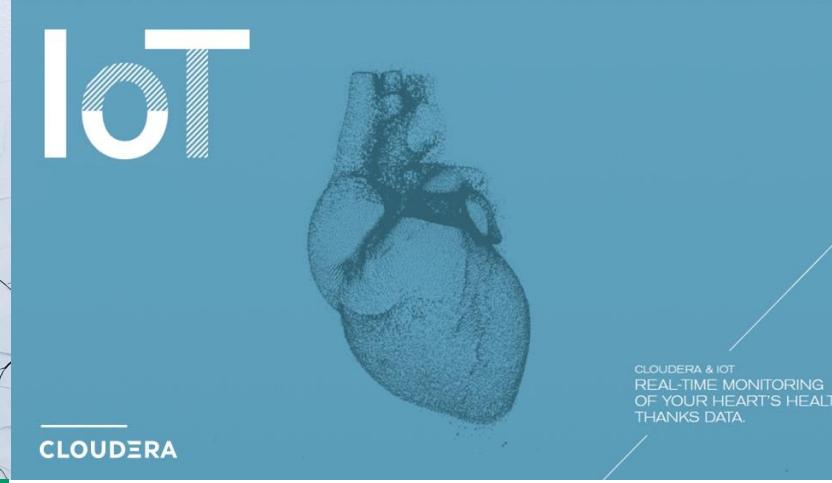
Timothy Spann
Principal Developer Advocate

CLOUDERA



ENTERPRISE DATA CLOUD

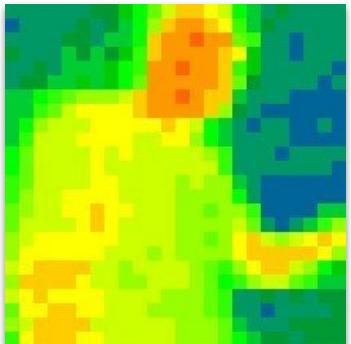
CLOUDERA



CLOUDERA



FLiPN-FLaNK Stack



Tim Spann

@PaasDev // Blog: www.datainmotion.dev

Principal Developer Advocate. Princeton Future of Data Meetup.

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

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

Apache NiFi x Apache Kafka x Apache Flink



Round Table Discussion

- Current trends of real-time stream processing in 2023
- Current challenges of real-time stream processing and proposed solutions
- Performance / Benchmarking real-time stream processing
- The future of real-time stream processing

Round Table Discussion

Current trends of real-time stream processing in 2023

- Lightweight serverless
- Hazelcast
- SQL
- Flink
- Kafka or Pulsar as Messaging Hub
- Java 17+
- Managed Clusters, Containers and Environments
- Real-Time Analytics
- Fast Storage Options

Round Table Discussion

Current challenges of real-time stream processing and proposed solutions

- Deployment, Automation and Scaling
- Choosing right project/sizing for use case
 - Simple Event Processing vs Complex Event Processing
 - Leveraging existing applications
- Developer Skills
- Self management and monitoring
- Cost issues -> autoscaling, optimizing, performance, hybrid deployment

Round Table Discussion

Performance / Benchmarking real-time stream processing

- Kafka/Pulsar: <https://openmessaging.cloud/docs/benchmarks/>
- NiFi: <https://blog.cloudera.com/benchmarking-nifi-performance-and-scalability/>
- Flink: <https://github.com/ververica/flink-sql-benchmark>
- Hazelcast:
<https://hazelcast.com/press-release/hazelcast-demonstrates-cloud-efficiency-real-time-stream-processing-of-one-billion-events-per-second/>

Round Table Discussion

The future of real-time stream processing

- WASM (Web Assembly)
- Petabyte, 5000 Node Clusters, Smart Hyper Scaling
- Multi-language support (Python, Rust, Kotlin, Golang, Carbon, JVM)
- Machine Learning, Deep Learning, AI and Advanced Math
- Low Code Development like Apache NiFi, DataFlow Designer, SQL
- Dynamic Hybrid Deployment
- Citizen Stream Engineer
- Edge Streaming and Hybrid Edge Streaming
- Java 20, 21; Java Loom Virtual Threading
- Ultra low latency, trillions of events per second, massive RAM/network



Round Table Resources

- <https://dzone.com/articles/real-time-stream-processing-with-hazelcast-and-streamnative>
- <https://flipstackweekly.com/>
- <https://www.datainmotion.dev/>
- <https://www.flankstack.dev/>
- <https://github.com/tspannhw>
- <https://medium.com/@tspann>
- <https://medium.com/@tspann/predictions-for-streaming-in-2023-ad4d7395d714>
- https://www.apachecon.com/acna2022/slides/04_Spann_Tim_Citizen_Streaming_Engineer.pdf

Tim **SPANN**

<https://github.com/tspannhw>

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



CONNECTED DEVICES ARE EVERYWHERE

EDGE



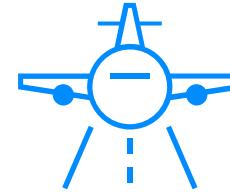
Miles driven



Wear-out
tires



Wearing of
the doors

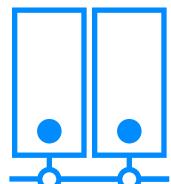


Engine wear

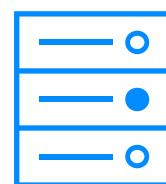


Capture data from all
these sources and
scale with a data
streaming platform in a
hybrid architecture

DATA
CENTER



Data capacity



Compute speed



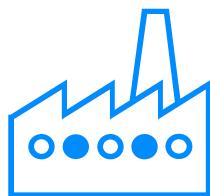
Rising temperature

TODAY'S NEEDS FOR DATA STREAMING

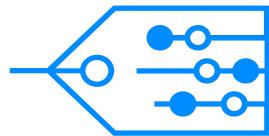
Gain Competitive Advantage

"Many leading enterprises realize that real-time analytics – the analytics of the present – is an incredible **competitive advantage** because they can act now to **serve fickle customers, fix operational problems, power internet-of-things (IoT) apps, and respond decisively to competitors.**"

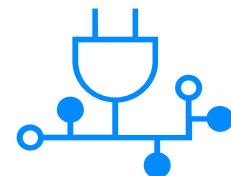
Forrester



Supply chain
impacts
manufacturing



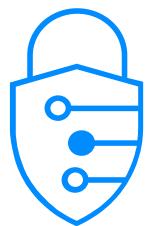
Predict
customer
buying pattern



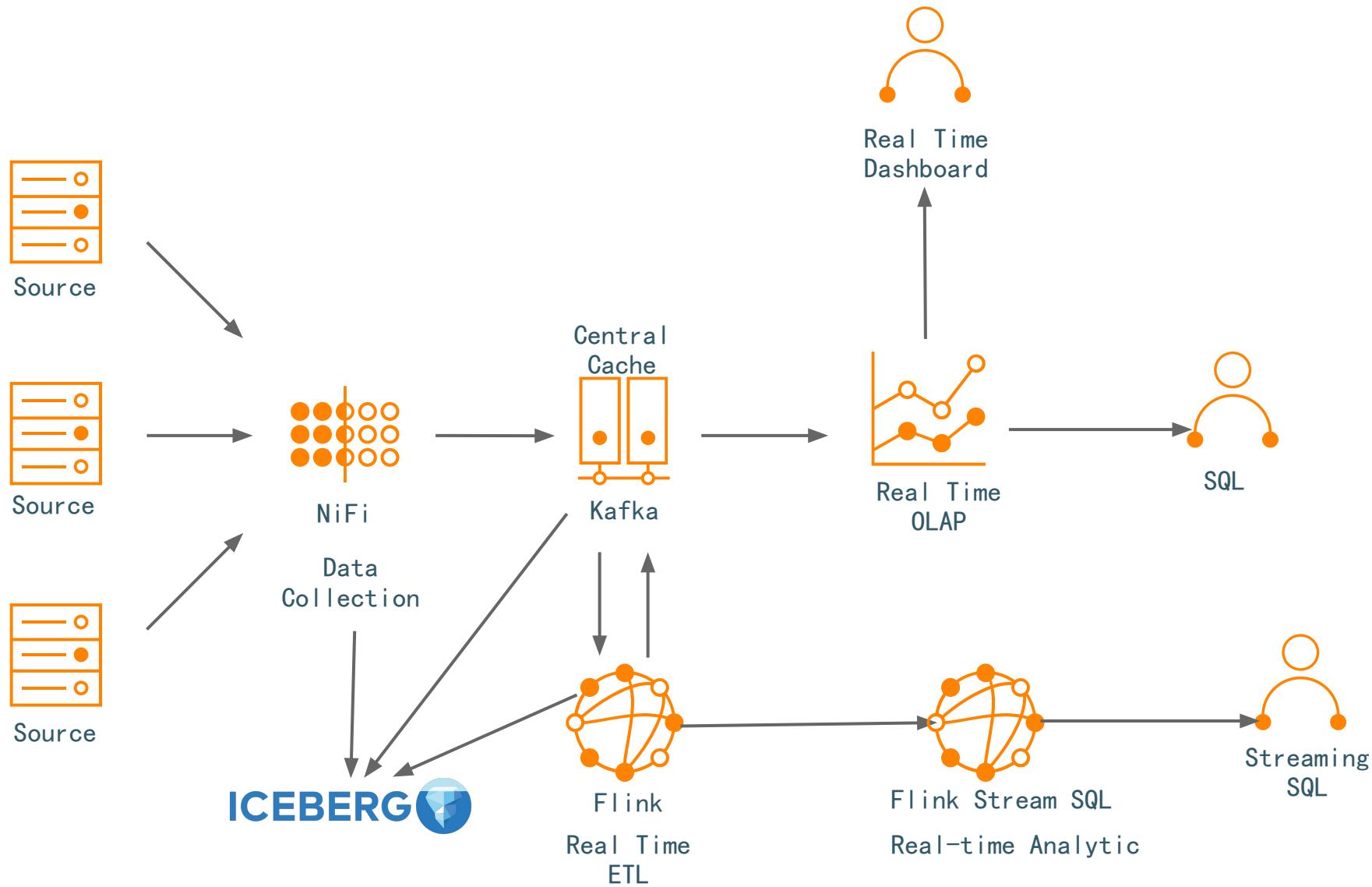
Utilities
prevent power
outage

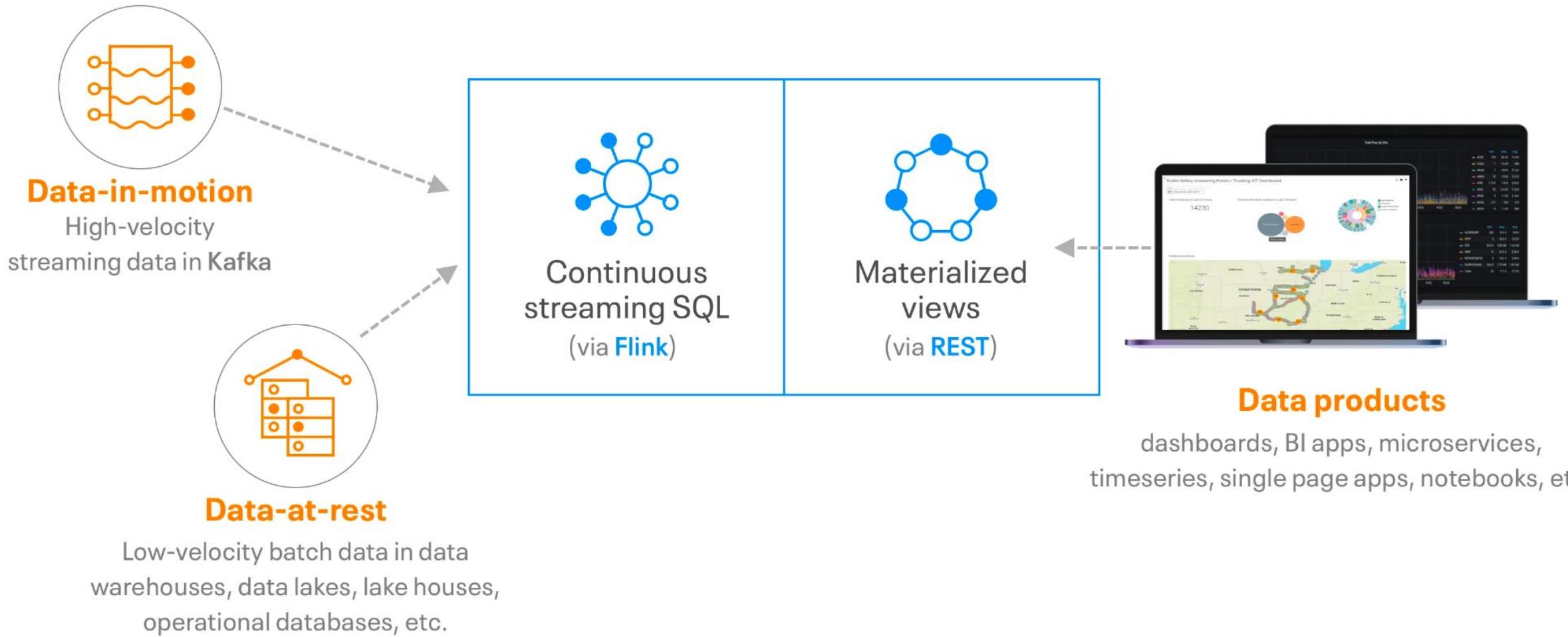


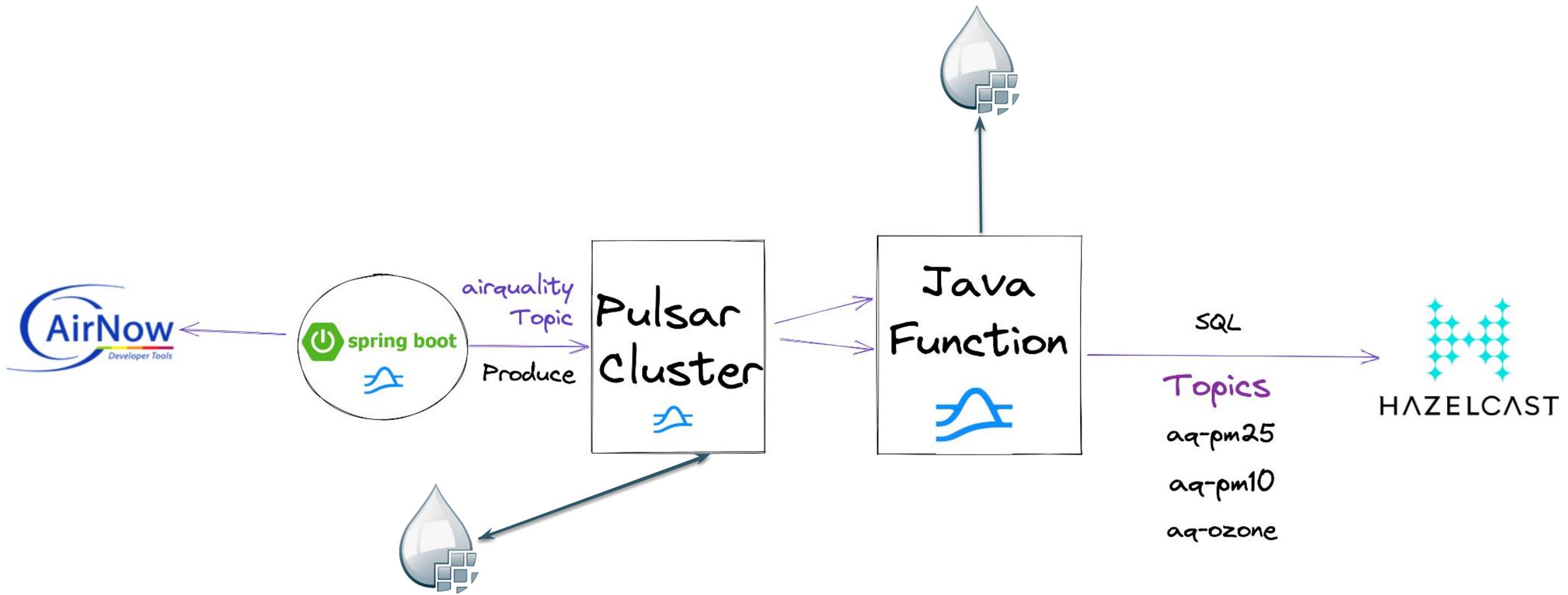
Telecoms deliver
continuous
QoS



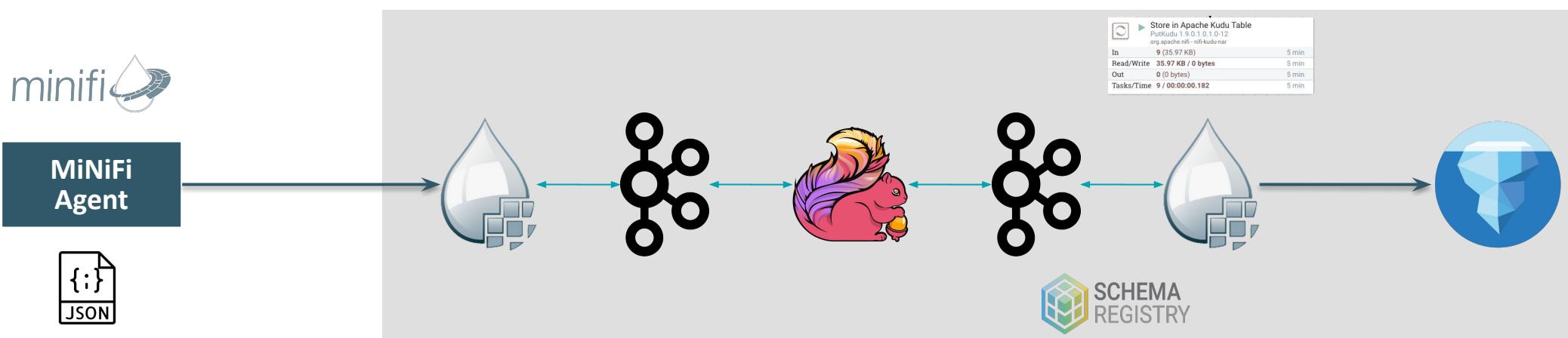
Reduce
cyber
threats





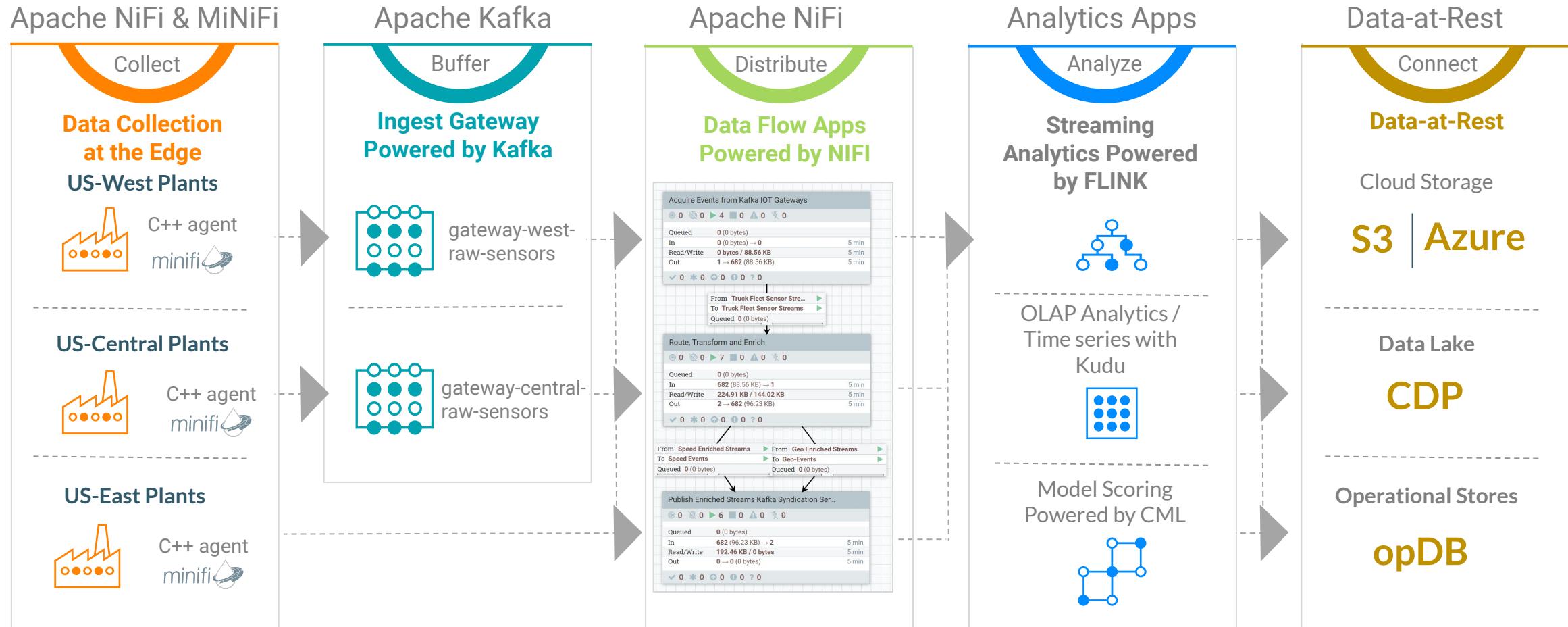


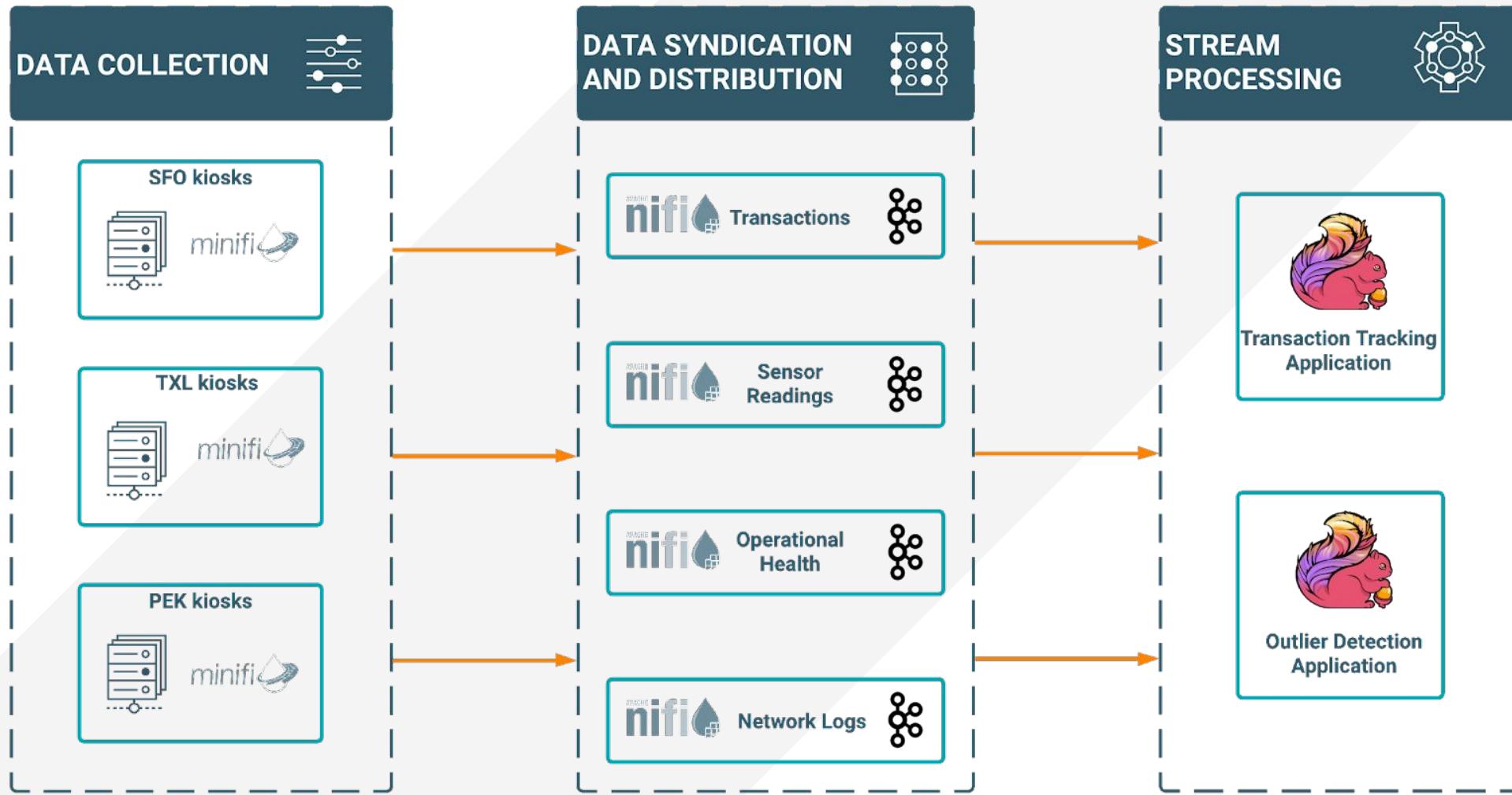
FLaNK Stack



<https://flankstack.dev/>

A DATA-IN-MOTION REFERENCE ARCHITECTURE



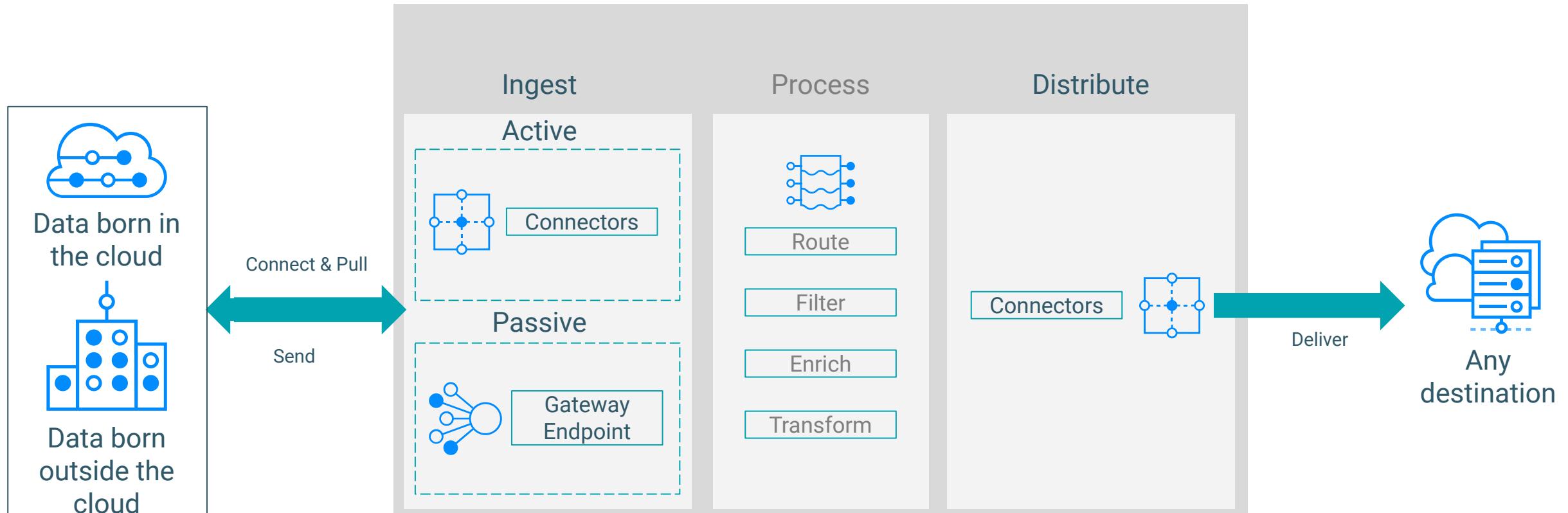


CLOUDERA DATAFLOW DATA-IN-MOTION PLATFORM



CLOUDERA DATAFLOW

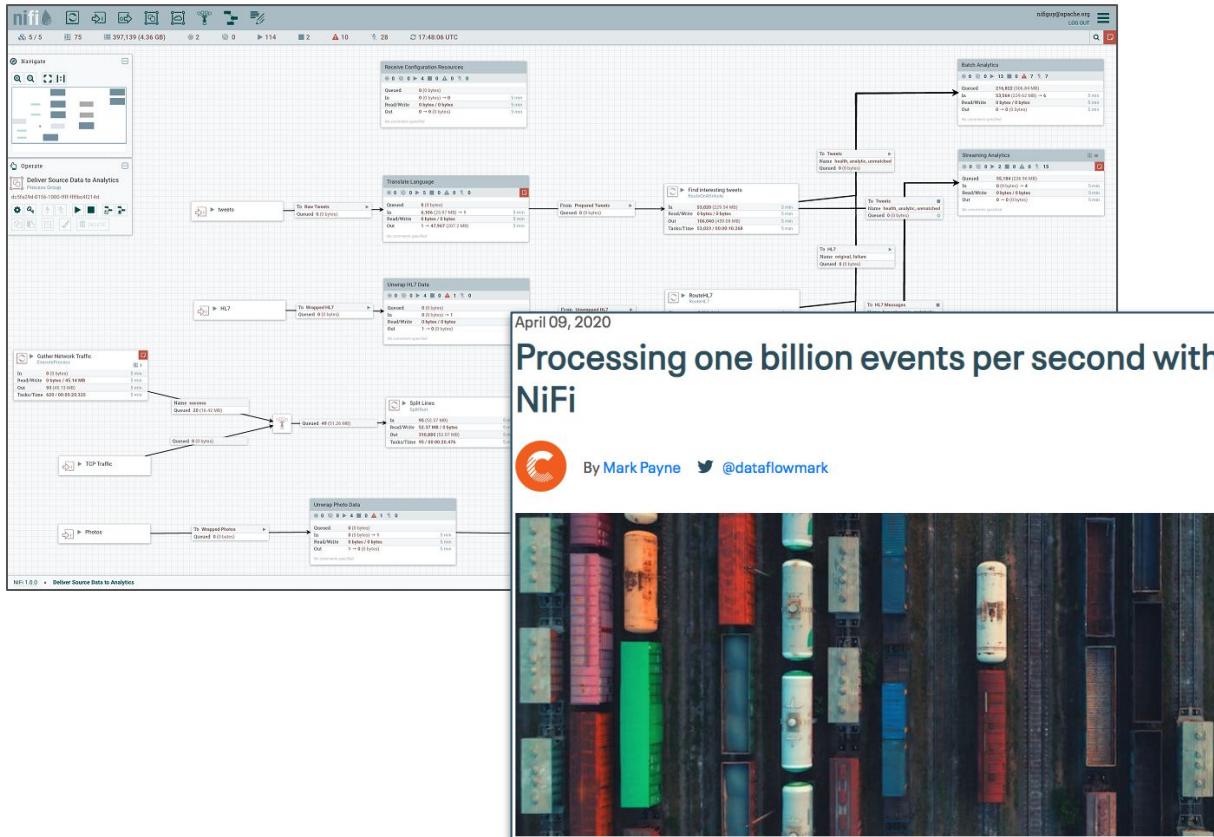
Solve the First Mile Data Collection Problem



Connect to any data source anywhere, process, and deliver to any destination

CLOUDERA FLOW MANAGEMENT - POWERED BY APACHE NiFi

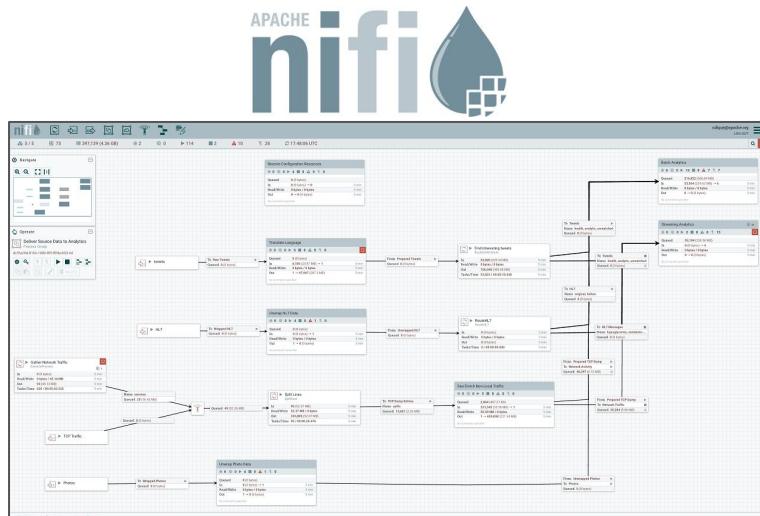
Ingest and manage data from edge-to-cloud using a no-code interface



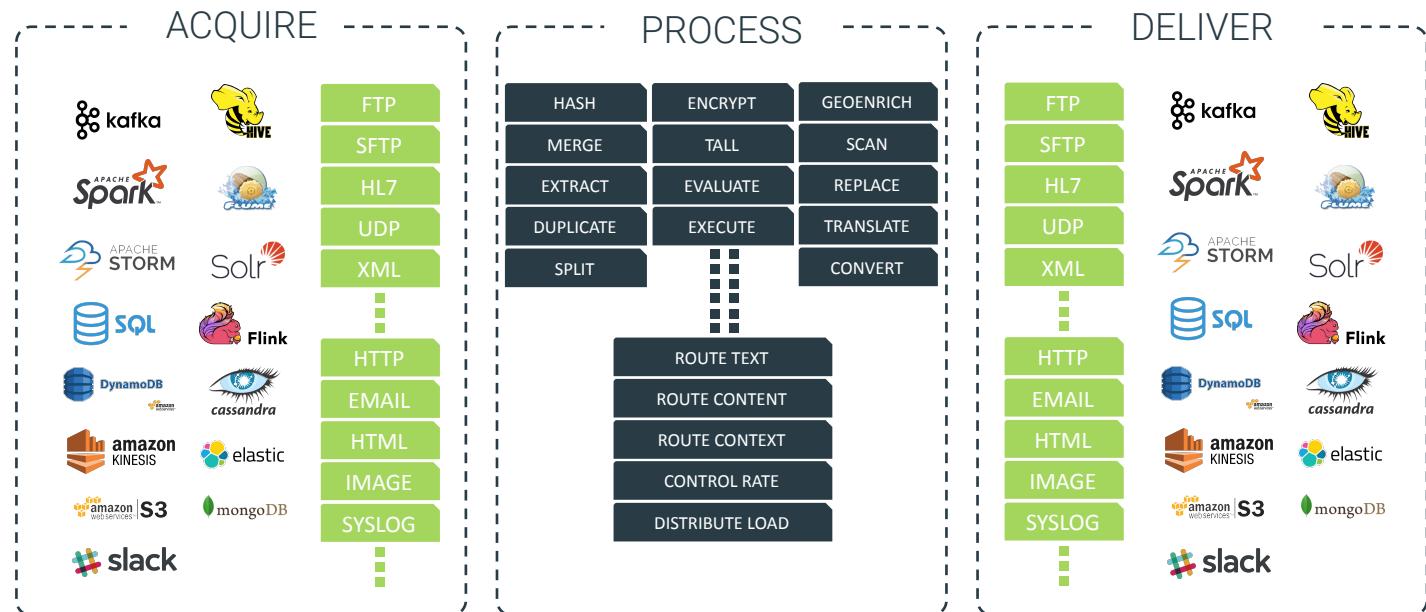
- #1 data ingestion/movement engine
- Strong community
- Product maturity over 11 years
- Deploy on-premises or in the cloud
- Over 400+ pre-built processors
- Built-in data provenance
- Guaranteed delivery
- Throttling and Back pressure

Cloudera Flow Management

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



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

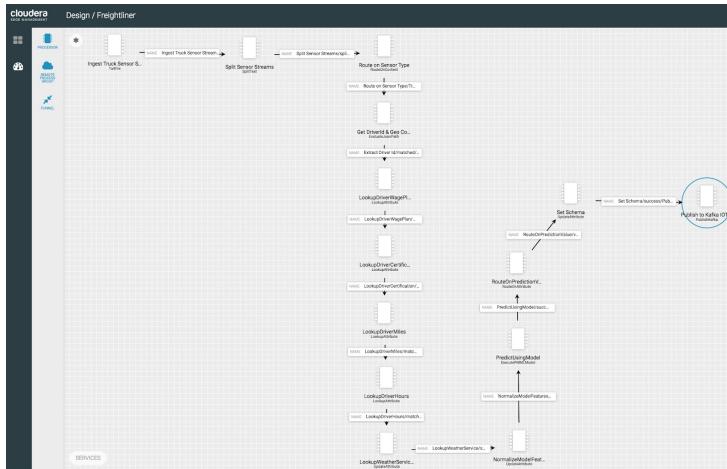


- Over 300 pre-built processors
- Easy to build your own processors
- Parse, enrich & apply schema
- Filter, Split, Merge & Route
- Throttle & Backpressure
- Guaranteed delivery
- Full data provenance
- Eco-system integration

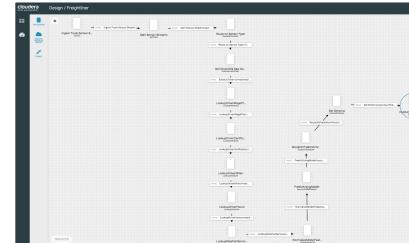
Cloudera Edge Management

Manage, control, and monitor the edge for all your streaming and IoT initiatives

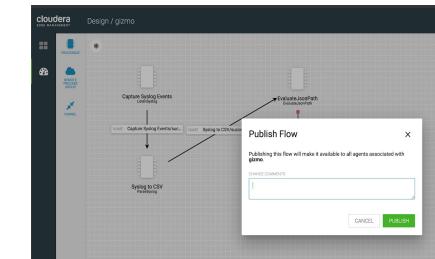
Edge Flow Manager



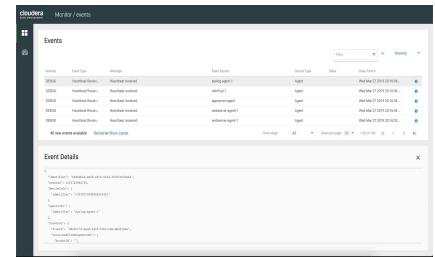
Flow Authorship



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

- Central Command and Control
- Design and deploy to thousands of agents
- Edge Applications lifecycle management
- Multitenancy with Agent classes
- Native integration with other CDF services

Comprehensive Streams Messaging

Extend streams messaging services for Schema Mgmt, Replication & Monitoring

Streams Messaging Manager
Management & Monitoring Service
for all of your Kafka clusters

Streams Replication Manager
Kafka Replication Service powered by
MirrorMaker2

The screenshot shows the Schema Registry interface with the title 'All Schemas'. It displays a schema named 'syndicate-speed-event-avro' with the following details:

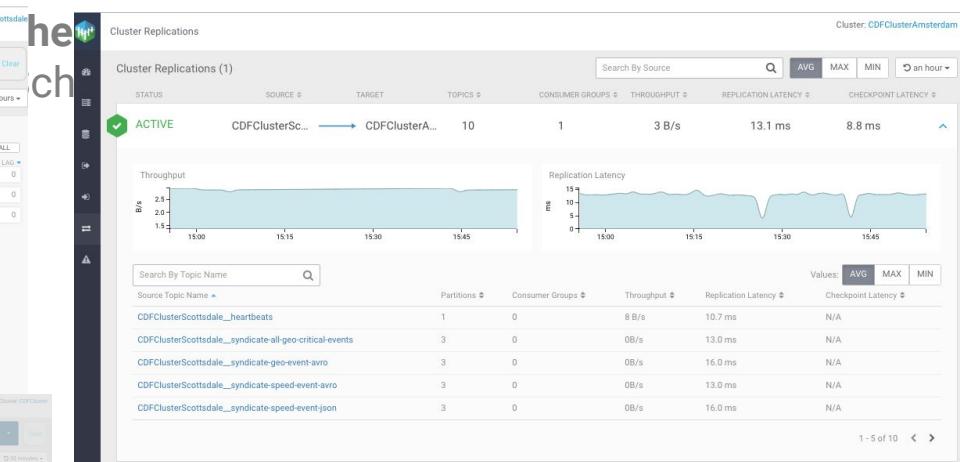
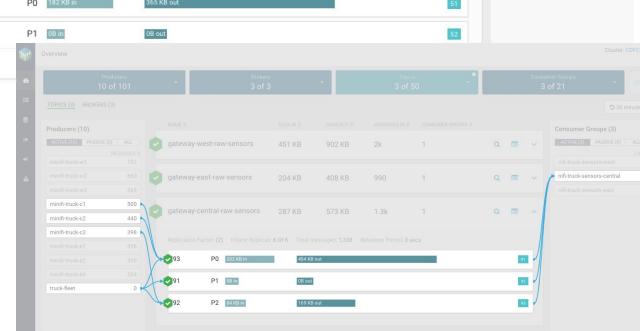
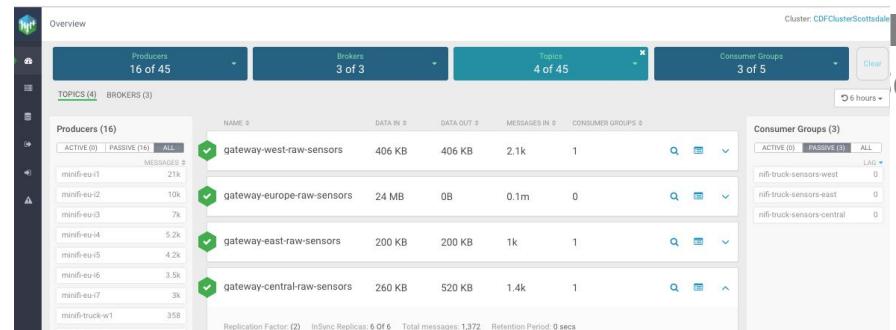
- Type: avro
- Group: truck...
- Branch: 1 ↗
- Serializer & Deserializer: 0

Branch Description: 'MASTER' branch for schema metadata 'syndicate-speed-event-avro'.

Version Description: Enriched Speed Events from trucks in Kafka Topic.

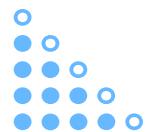
The schema code is as follows:

```
1 {  
2   "type": "record",  
3   "namespace": "cloudera.cdf.csp.schema.refapp",  
4   "name": "TruckSpeedEventEnriched",  
5   "fields": [  
6     {"name": "eventTime",  
7      "type": "string"},  
8     {"name": "eventTimeLong",  
9        "type": "long",  
10       "default": 0},  
11     {"name": "speed",  
12       "type": "float"},  
13     {"name": "lat",  
14       "type": "float"},  
15     {"name": "lon",  
16       "type": "float"}]
```



Next Generation Stream Processing & Analytics

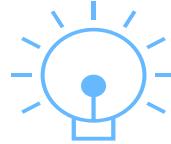
Low latency stateful stream processing



Low
Latency



Event
Processing



Real-Time
Insights

- Advanced features - late arriving data, checkpointing, event time processing, Exactly Once Processing
- We support Apache Flink along with other stream processing engines like Kafka Streams and Spark Structured Streaming.

The screenshot displays the Cloudera Manager interface. At the top, it shows the status of the CDFClusterAmsterdam cluster, which is running. Below this, there are two main sections: 'Status' and 'Charts'. The 'Status' section lists various components: CDH 6.3.0 (Parcels), Flink, HDFS-2, Kafka-2, Schema Registry, Streams Metrics, Streams Repl..., YARN (MR2 In...), and ZooKeeper-2. The 'Charts' section contains two graphs: 'Cluster CPU' and 'Cluster Network IO'. The 'Cluster CPU' graph shows CPU usage over time, with a notable peak around 07:15. The 'Cluster Network IO' graph shows network traffic in bytes/second. Below these are sections for 'Data Contexts' and 'Flink' (under the Clusters tab). The 'Flink' section includes tabs for Status, Instances, Configuration, Commands, Charts Library, Audits, and History Server Web. It also includes 'Health Tests' and 'Status Summary' sections. On the right side of the interface, there is a detailed view of a 'Trucking Streaming Analytics Flink App' (ID: 72c0982cf7fbcb8027e2872926070558) which is running. This view includes tabs for Overview, Exceptions, TimeLine, Checkpoints, and Configuration. The Overview tab shows the job's ID, version, commit, start time, duration, and a 'Cancel Job' button. The configuration tab shows the Flink job code, which includes operations like Window Tumbling Event Time Windows(180000), EventTimeTrigger, Map, Reduce Aggregate Function, Pass Through Window Function, Interval Join, Timestamps/Watermarks, Filter, and Sink. A table at the bottom lists the tasks for each component, showing their status, bytes received, records received, bytes sent, and tasks.

Name	Status	Bytes Received	Records Received	Bytes Sent	Tasks
Source: Kafka SpeedGeoStream	RUNNING	0 B	0	3.16 MB	6
Source: Kafka TruckGeoStream	RUNNING	0 B	0	4.09 MB	6
Stream Join using Interval Join -> Timestamps/Watermarks	RUNNING	7.48 MB	9,980	4.19 MB	6
WindowTumblingEventTimeWindows(180000), EventTimeTrigg...	RUNNING	4.58 MB	4,994	0 B	6

KEY DIFFERENTIATORS

Stream to Cloud – Extend the same on-premises streaming capabilities to the cloud with full support for multi-cloud and hybrid cloud models



400+ pre-built processors – Only product to offer such comprehensive connectivity to a wide range of data sources from edge to cloud



Democratize access to real-time data – Enable data analysts and other personas to quickly build streaming applications with just SQL



Enterprise-Grade Security & Governance – Deploy your streaming applications with confidence and trust with Cloudera SDX offering unified security and governance across the entire platform

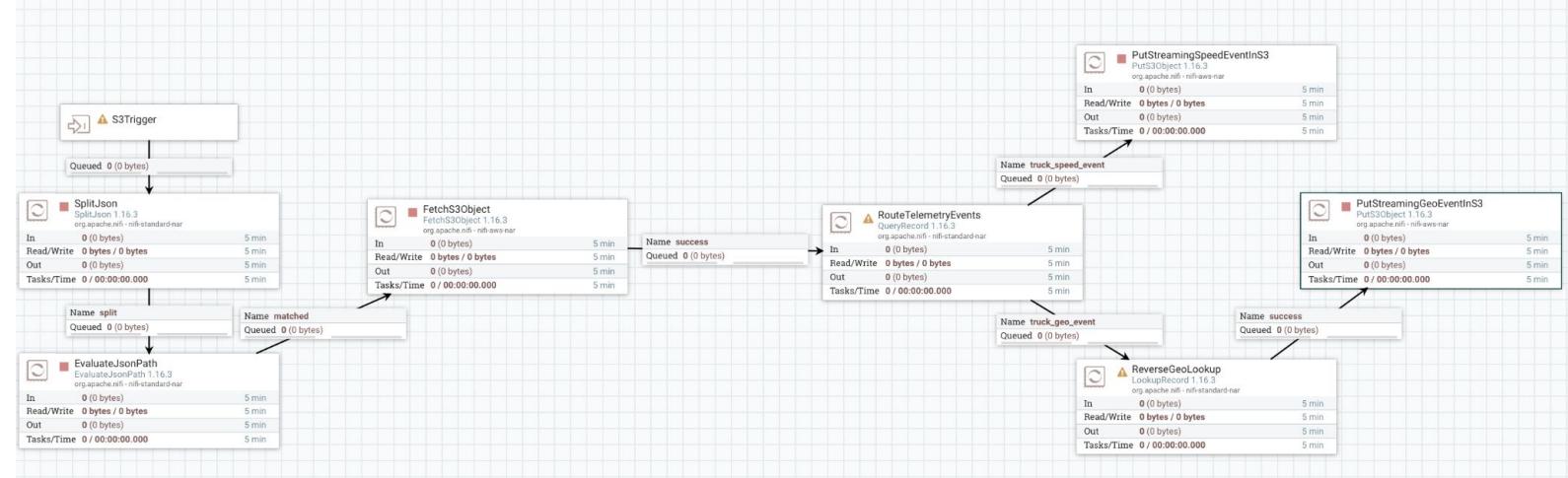


Comprehensive streaming platform – Only vendor to offer a open and comprehensive streaming platform for real-time data ingestion and processing to produce prescriptive and predictive analytics



Development & Runtime of DataFlow Functions

Step1. Develop functions on local workstation or in CDP Public Cloud using **no-code**, UI designer



Step 2. Run functions on serverless compute services in AWS, Azure & GCP



AWS Lambda



Azure Functions



Google Cloud Functions

DataFlow Functions Use Cases

Trigger Based, Batch, Scheduled and Microservice Use Cases

Serverless Trigger-Based File Processing Pipeline

Develop & run data processing pipelines when files are created or updated in any of the cloud object stores

Example: When a photo is uploaded to object storage, a data flow is triggered which runs image resizing code and delivers resized image to different locations.

Serverless Workflows / Orchestration

Chain different low-code functions to build complex workflows

Example: Automate the handling of support tickets in a call center or orchestrate data movement across different cloud services.

Serverless Scheduled Tasks

Develop and run scheduled tasks without any code on pre-defined timed intervals

Example: Offload an external database running on-premises into the cloud once a day every morning at 4:00 a.m.

Serverless Microservices

Build and deploy serverless independent modules that power your applications microservices architecture

Example: Event-driven functions for easy communication between thousands of decoupled services that power a ride-sharing application.

Serverless Web APIs

Easily build endpoints for your web applications with HTTP APIs without any code using DFF and any of the cloud providers' function triggers

Example: Build high performant, scalable web applications across multiple data centers.

Serverless Customized Triggers

With the DFF State feature, build flows to create customized triggers allowing access to on-premises or external services

Example: Near real time offloading of files from a remote SFTP server.

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 displays a list of flow definitions:

Name
Covid Data Stream
CovidIDBroker
drew_kafka-hdfs-querydb-kudu
drew_kafka_to_hdfs
Employees Data
Empty Dev Flow
Generate Flow File Log

To the right, a detailed view of the 'Covid Data Stream' flow is shown:

- Actions** button
- » Covid Data Stream**: Updated 2 months ago by Michael Kohs
- FLOW DESCRIPTION**: This flow reads covid data from several sources and writes it to CDP
- Only show deployed versions
- Version** table:
 - Version 13, Deployments 0
 - Version 12, Deployments 0
 - Version 11, Deployments 0
- Deploy New Flow →** button
- LAST UPDATE**: 2021-02-02 14:25 PST by Michael Kohs
"This version includes the latest fixes"

ReadyFlows

- Cloudera provided flow definitions
- Cover most common data flow use cases
- Can be deployed and adjusted as needed
- Made available through docs during Tech Preview

Cloudera Docs / DataFlow master ▾ (test • Technical Preview) Search Document

Cloudera DataFlow

Release Notes

Release Notes

Concepts

Overview

Planning

AWS Resource Planning

NiFi Flow Limitations

Getting Started

Quick Start

Out of Box Flow Definitions

Import a flow definition

Flow definition for ingesting data into a Kafka topic

Flow definition for ingesting data into Amazon S3 Buckets

How To: Environments

Enabling a DataFlow Environment

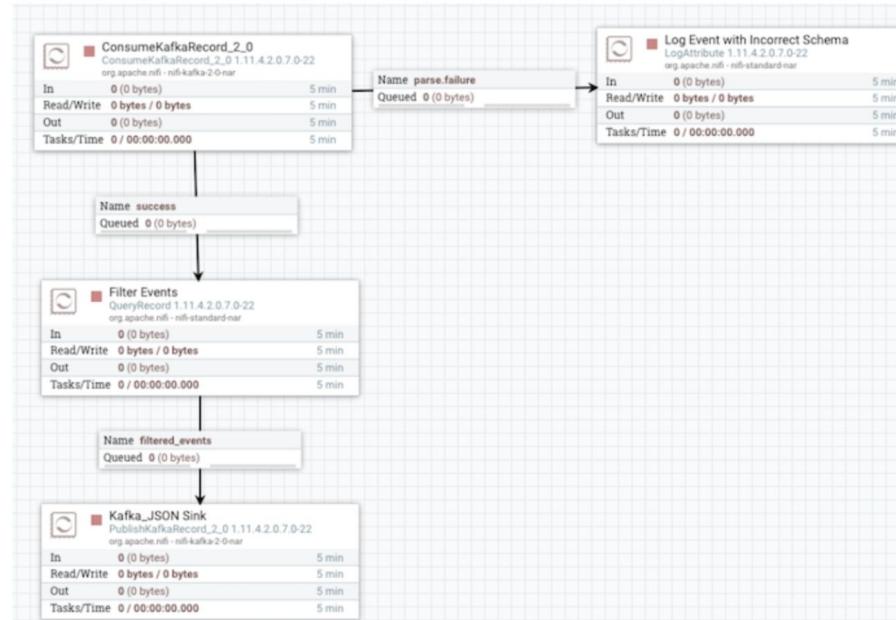
Managing a DataFlow Environment

OUT OF BOX FLOW DEFINITIONS

Flow definition for ingesting data into a Kafka topic

Example

The resulting flow will look similar to the following, on your NiFi canvas.



Deployment Wizard

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

Start Deployment Wizard

New Deployment

Select the target environment

ⓘ Sensitive data never leaves the environment. Changing the environment after this step requires restarting the deployment process.

Selected Flow Definition

NAME
 Machine Data To Warehouse

VERSION
 2

Target Environment

aws dataflow-demo 60% (3 of 5)

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

Enter parameter values.

Select File
Drop file or browse

CDP Truststore

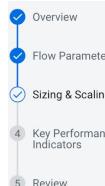
Enter parameter values.

Select File
Drop file or browse

CDPSchemaRegistry

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

Configure Sizing & Scaling



Sizing & Scaling

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

NiFi Node Sizing



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

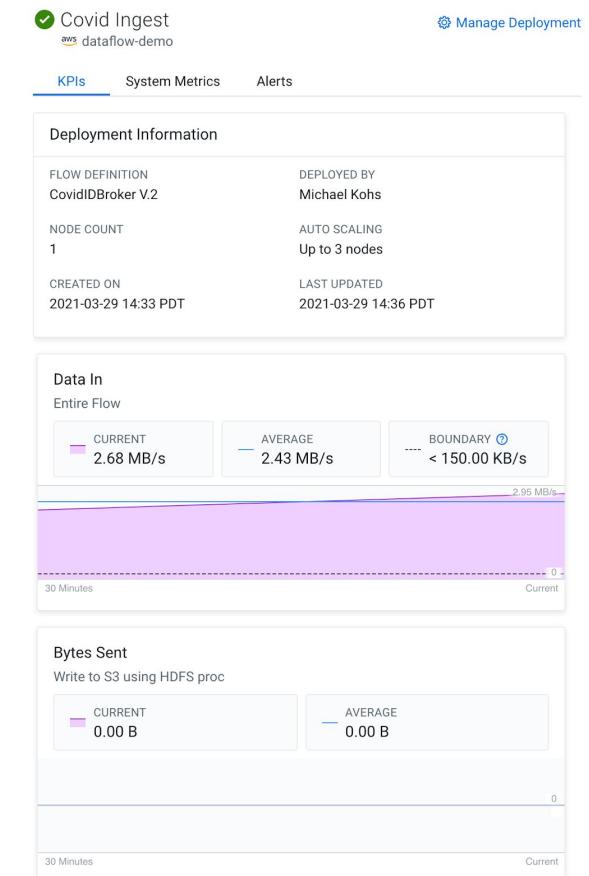
The screenshot shows the 'KPIs' tab of the Deployment Wizard. On the left, a vertical navigation bar lists steps: Overview, Flow Parameters, Sizing & Scaling, Key Performance Indicators (which is the current step), and Review. The main area is titled 'Key Performance Indicators' with the sub-instruction: 'Set up KPIs to track specific performance metrics of a deployed flow. Click and drag to reorder how they are displayed.' Below this, two KPI definitions are listed in a grid:

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

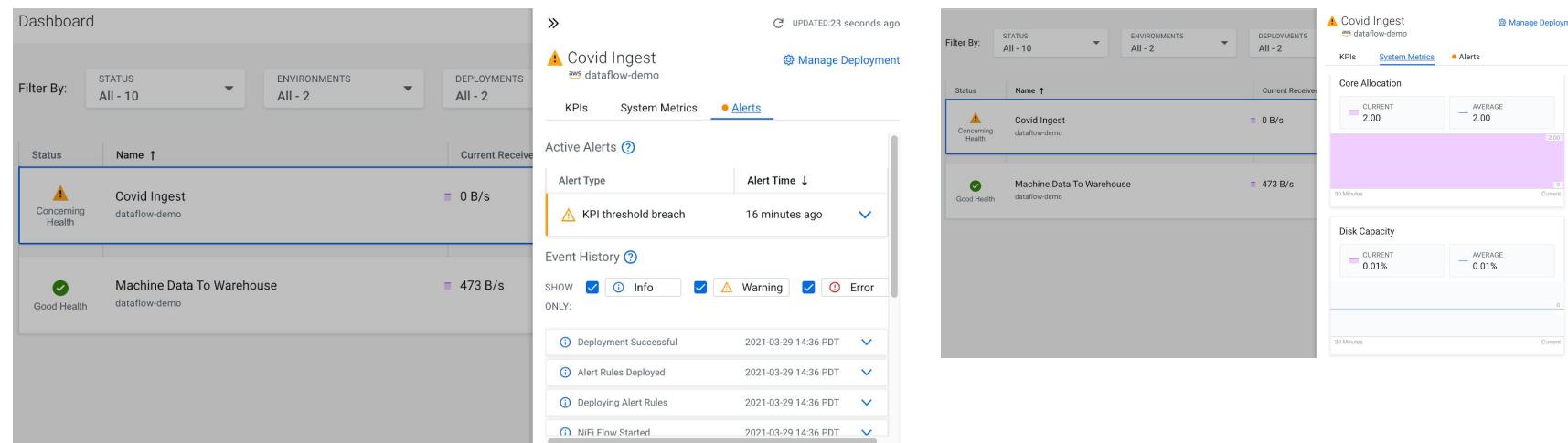
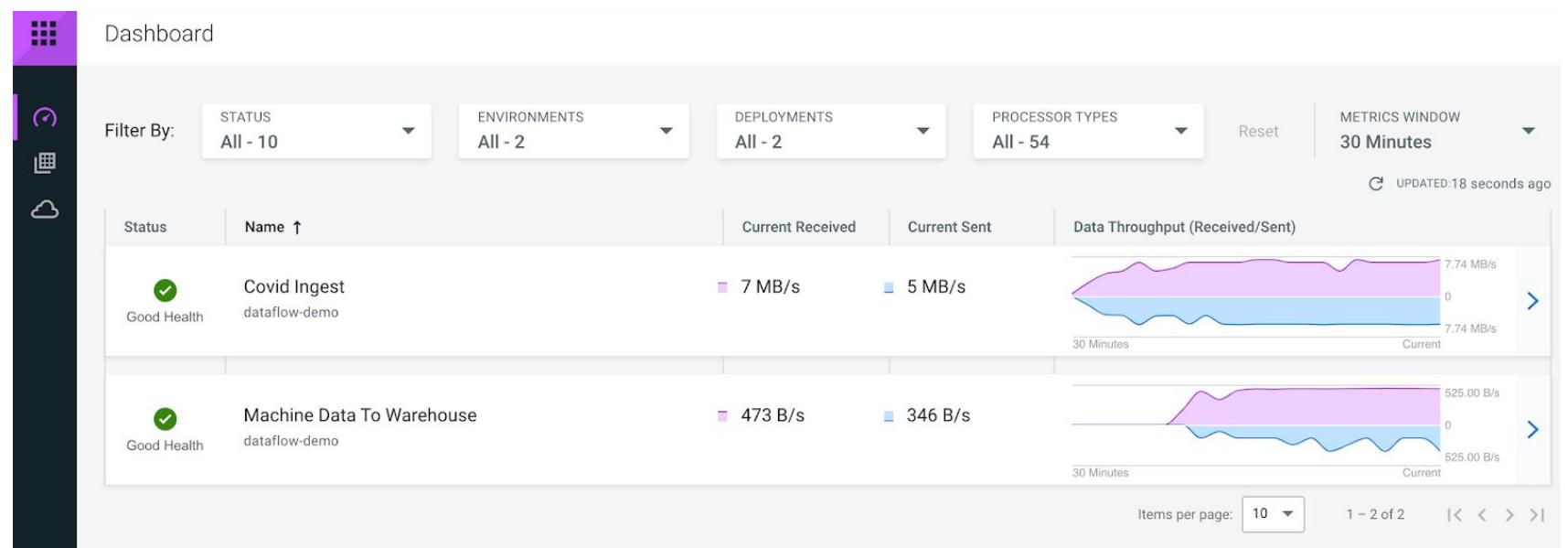
A dashed box highlights the 'Add New KPI' button at the bottom right of the grid.

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



DATA FLOW DESIGN FOR EVERYONE

- Cloud-native data flow development
- Developers get their own sandbox
- Start developing flows without installing NiFi
- Redesigned visual canvas
- Optimized interaction patterns
- Integration into CDF-PC Catalog for versioning

Flow Design / [WorkspaceName] / [FlowDefinitionName]

Active Test Session Flow Options

» Configuration Metrics

* Region ::

Access Key ID ::

Secret Access Key ::

Record Writer ::

* Minimum Object Age ::

Listing Batch Size ::

* Write Object Tags ::

* Write User Metadata ::

Credentials File ::

AWS Credentials Provider... ::

* Communications Timeout ::

SSL Context Service ::

Endpoint Override URL ::

- +

Apply Changes Discard Changes

Flow Design / [RootProcessGroupName]

Processor Details

[ProcessorName] [ProcessorType] [Version#]

IN 19 (14.16 MB)

READ/WRITE 4.88 MB/4.88 MB

OUT 0 (0 bytes)

TASKS 29/00:00:00.123

5 min

