CLOUDERA

Continuous SQL with SQL Stream Builder

Kenny Gorman - Product Owner
Timothy Spann - Principal DataFlow Field Engineer
John Kuchmek - Senior Solutions Engineer

06-May-2021

https://www.meetup.com/futureofdata-newyork/

@PaasDev

Welcome to Future of Data – Virtual

FUTURE OF DATA

[Princeton Future of Data Meetup]
[New York Future of Data Meetup]
[Philadelphia Future of Data Meetup]

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 ...

AGENDA

- Introductions with Kenny, John and Tim
- Flink Quick Overview
- SQL Stream Builder Overview
- Q&A
- Demos
- Q&A - Interactive Panel Session
- Next Meetups
- Raffle

Cloudera DataFlow Use Cases

Data Movement
Optimize resource utilization by moving data between data centers or between on-premises and cloud infrastructures
e.g. intercontinental data exchange

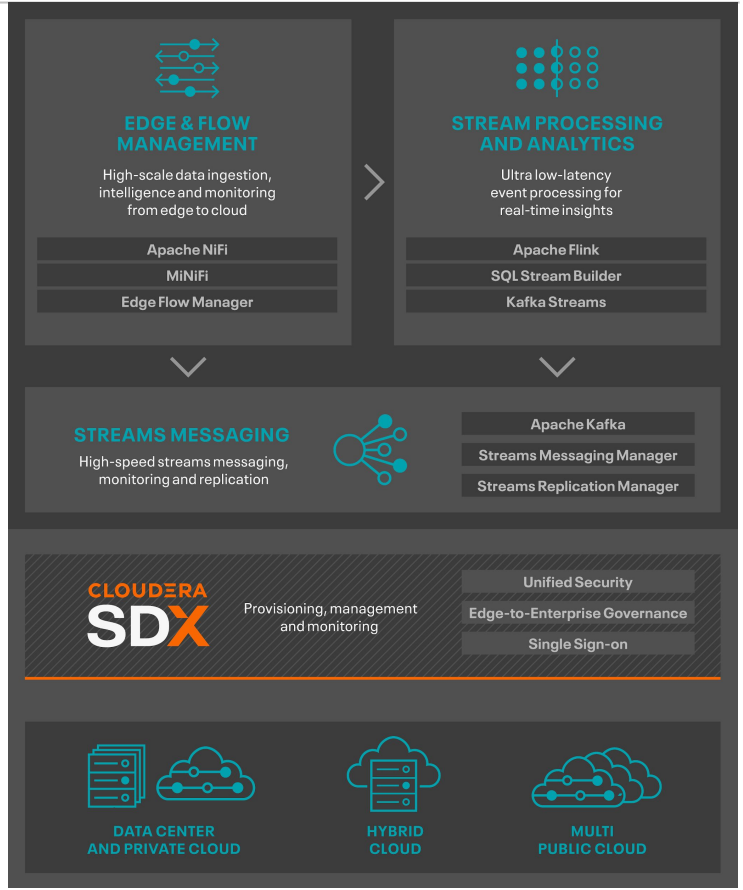
Logging Modernization
Optimize log analytics solutions by with CDF in simplifying log ingestion from the edge, reducing costs and gaining key analytics
e.g. Splunk / Logstash offload

Streaming analytics insights
Make key business decisions by analyzing streaming data for complex patterns, gaining actionable intelligence etc.
e.g. Fraud detection, Network threat analysis, app monitoring, Clickstream analysis

360° view of customer
Ingest, transform and combine customer data from multiple sources into a single data view / lake
e.g. Real-time customer offers, Loan approvals

IoT & Edge use cases
e.g. Predictive Maintenance, Asset Tracking / Monitoring, Patient Monitoring, Quality Processes, Fleet Management, Connected Cars and more

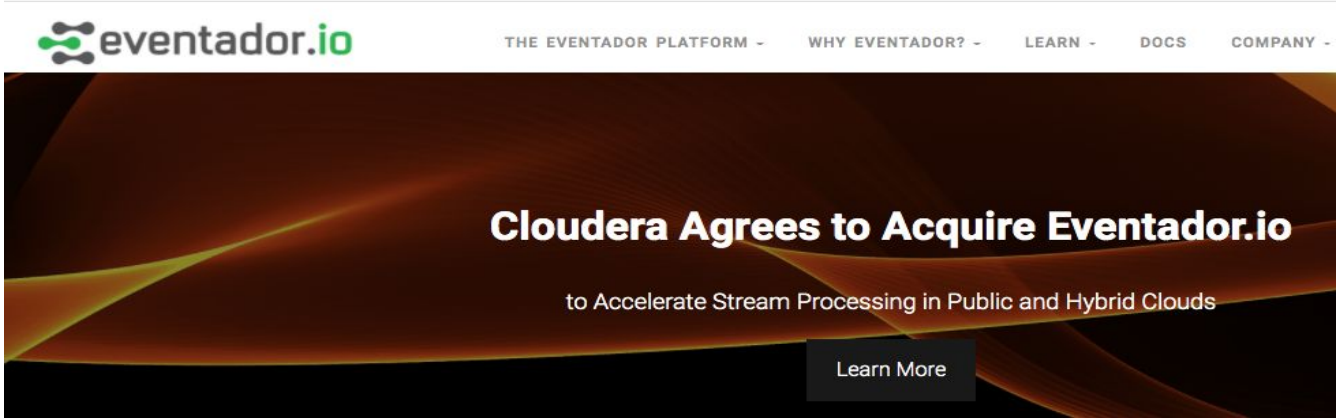
Enterprise data management
Managing massive volumes of high-velocity data to/from legacy systems, ETL tools and other data stores
e.g. Flume offload, ETL replacement, payment data processing, integration with Oracle



Simplifying the User Experience

October 12, 2020

Cloudera acquires Eventador to accelerate Stream Processing in Public & Hybrid Clouds



While the acquisition may be seen as incremental or small in nature as Eventador isn't yet a household name, the capabilities and improvements it has the potential to make for Cloudera shouldn't be understated. A solid move by Cloudera to continue its momentum forward.

APACHE FLINK

3B+

3B+ data points daily streaming in from 25 million customers running real time machine learning prediction

USE CASE

Streaming real-time data pipelines that need to handle complex stream or batch data event processing, analytics, and/or support event-driven applications

TECHNOLOGY

Flink performs compute at in-memory speed at any scale

Flink parses SQL using Apache Calcite, which supports standard ANSI SQL

Flink runs standalone, on YARN, and has a K8s Operator

APPLICATION

Comcast a global media uses Flink for operationalizing machine learning models and near-real-time event stream processing

Flink helps deliver a personalized, contextual interaction reducing time to support resolutions saving millions of dollars per year

CONSIDERATION

Data Freshness SLAs

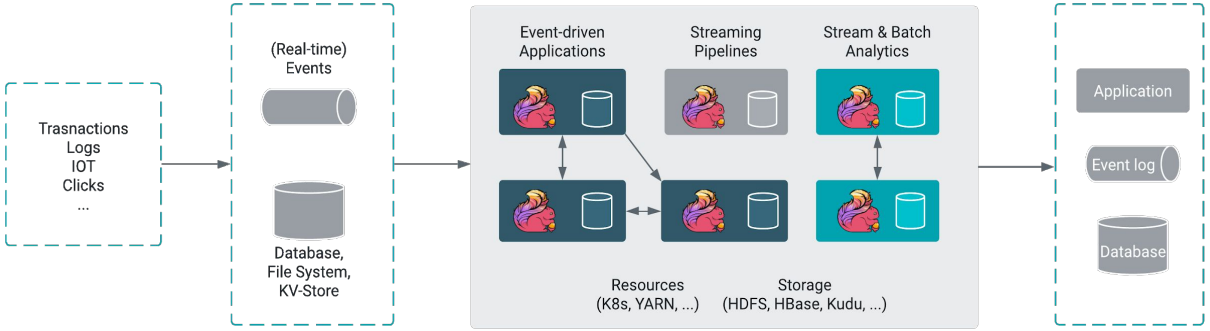
Flink can read and write from Hive data

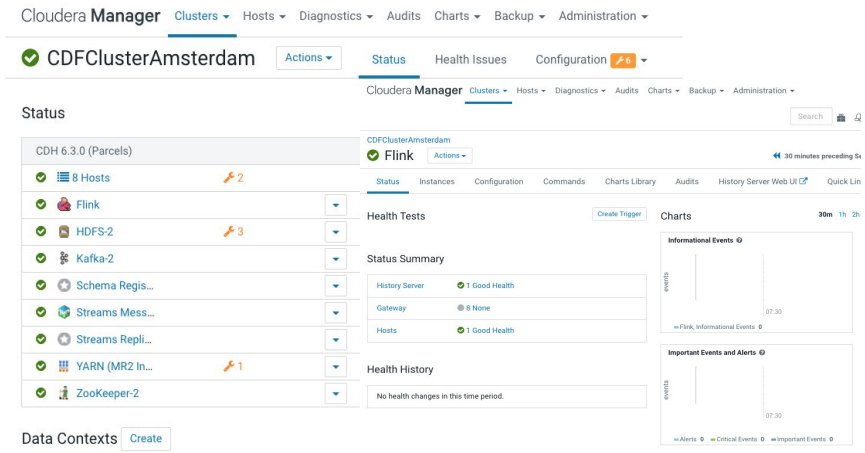
Review requirements for fault tolerance, resilience, and HA

Other technologies play in this space like Hive storage handler to connect to Kafka

FLINK FEATURES

- Distributed processing engine for stateful computations
- 10s TBs of managed state
- Flexible and expressive APIs
- Guaranteed correctness & Exactly-once state consistency
- Event-time semantics
- Flexible deployment & large ecosystem (K8s, YARN, S3, HDFS..)
- Support for Flink SQL API

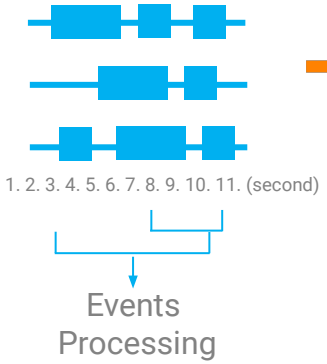




DELIVERING STREAMING ANALYTICS

Capture Events that Matter

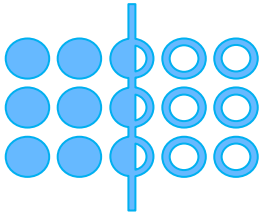
Low-latency analytics use cases



+

Parsing and Blending Data

Both offline and streaming data



+

Data Analysts Can Write Queries

Across the Line of Businesses

SQL

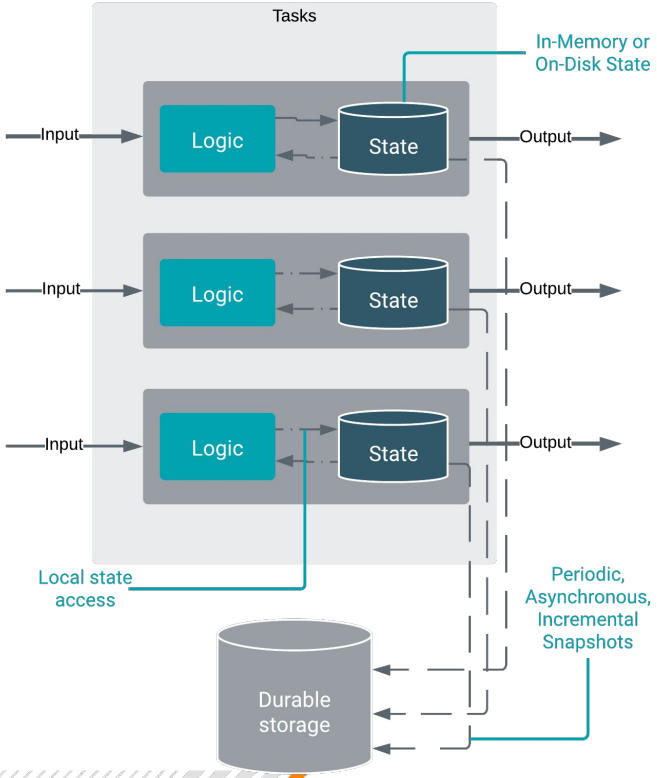
=

Streaming Analytics

MAINTAINS & CHECKPOINTS STATE

- Flink maintains state locally per task (in-mem / on-disk)
  - Fast access!

- State is periodically checkpointed to durable storage
  - A checkpoint is a consistent snapshot of the state of all tasks



Integrated Governance

Unified Governance & Lineage

Flow Management

Reports Entity and Lineage information about NiFi Flows

Connects with existing Lineage information

Streams Messaging

Topic access centrally managed supporting granular CRUD operations

Manage permissions on dedicated clusters or manage multiple clusters at once

Manage schemas centrally and make them available to consumers/producers

Stream Processing

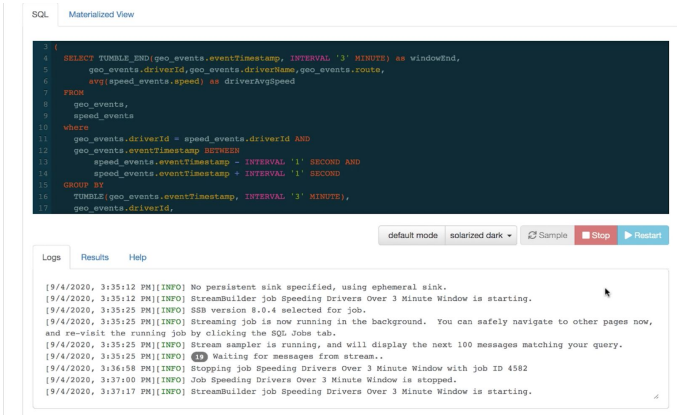
Reports Flink Apps as an operation

Lineage through integration with existing Lineage information like Kafka topics, HBase tables etc.

Integrated SQL and materialized view engine via SQL Stream Builder.

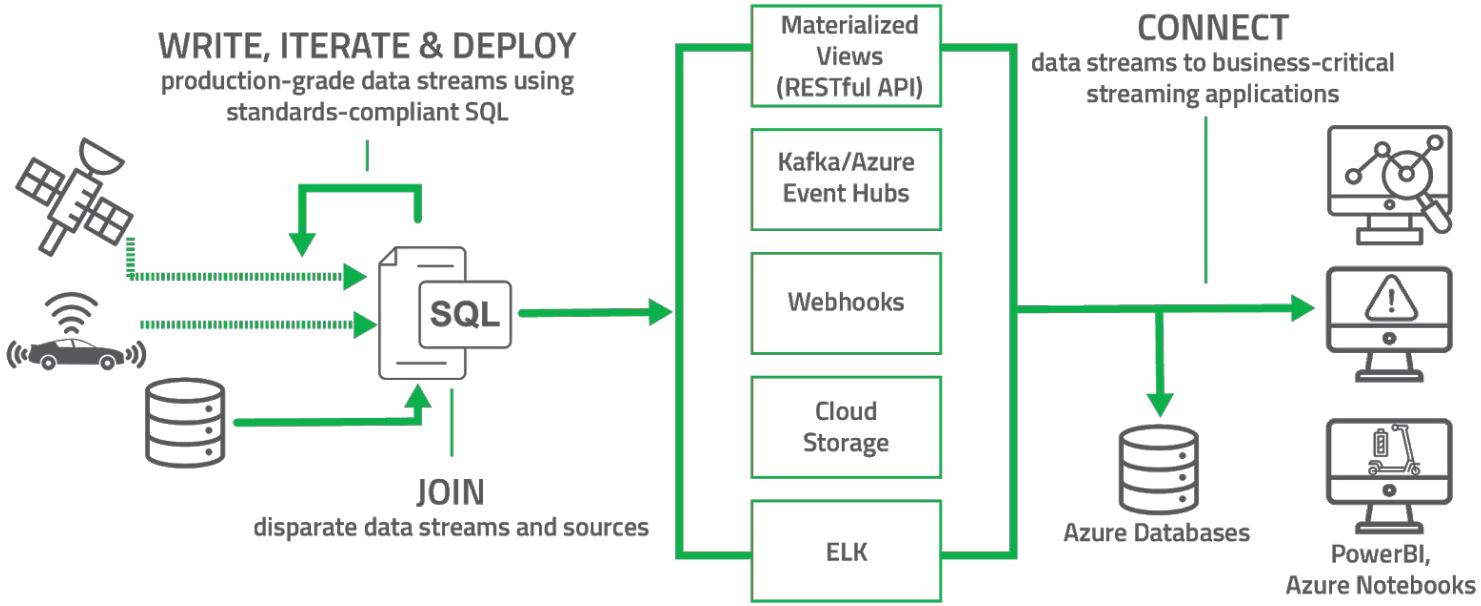
SQL Stream Builder

- Democratize data access across enterprise - anyone who knows SQL can create powerful stream processors.
- Iterative interface - Just like SQL on databases, run queries and reason about the data with an interactive UI.
- Leverages Apache Flink for running of SQL jobs - production grade, scalable and high performance
- Deep integration and features above and beyond just UI features - UDF's, input transforms, Kafka key and time integration, CEP framework and more.
- Create Materialized Views to integrate with downstream components like notebooks, visualizations and applications.



Streaming SQL

Democratizing access to streams of data via structured query language



Download these assets today


CLOUDERA

WHITE PAPER

CLOUDERA DELIVERS THE BEST KAFKA ECOSYSTEM TODAY

Serving Hundreds of Customers Globally

CLOUDERA

WHITE PAPER

The Complete Edge-to-Cloud Streaming Data Platform

This paper has focused primarily on the streams messaging aspects of the Kafka ecosystem with regard to how to secure, monitor, balance, and replicate large scale Kafka environments across on-premises, hybrid, private, and public cloud environments. The Data-in-Motion reference architecture diagram in Figure 7 below, puts this all in perspective.

A DATA-IN-MOTION REFERENCE ARCHITECTURE

Figure 7

As a result of supporting our customers through their data journeys, we've learned that it is not enough to have the best messaging solution at the heart of your end-to-end streaming architecture. As represented in the diagram above, flow management, along with stream processing and analytics, are two additional tenets that need to be unified with streams messaging capabilities. These three tenets, if properly integrated, will ensure a sustainable, scalable and adaptable end-to-end streaming architecture and is the basis to our data-in-motion philosophy.


CLOUDERA

SOLUTION BRIEF

DATA-IN-MOTION PHILOSOPHY

A Blueprint for Enterprise-wide Streaming Data Architecture

Real Life Data-in-Motion

Below describes how the data-in-motion capabilities described in this solution brief have been successfully applied by a Cloudera customer, end-to-end.

A global medical device manufacturer successfully modernized their messaging architecture to support a new line of implantable medical devices that generate more data, more often, and at a higher resolution than previous products.

- **Flow management**—Due to the private nature of medical data, the data flow was complex, requiring in motion and at rest encryption. NiFi's no code user interface enabled the initiative to be 100% business driven, only engaging the technology teams as needed.
- **Streams messaging**—Messaging volume jumped from quarterly reporting of device status to real-time health monitoring. Kafka enabled the business to scale that volume across multiple on-premises and cloud environments.
- **Stream processing and analytics**—The company had to transition from batch to real-time data processing. Flink handles both models along with complex event processing that is planned for the near future. The company, therefore, only needs to adopt and support one type of stream processing and analytics engine.

Don't Let Your End-to-End Streaming Architecture Fall Short

It's not enough to have the best messaging solution at the heart of your end-to-end streaming architecture. We've learned that as a result of supporting our customers through their data journeys. Flow management, along with stream processing and analytics, are two additional tenets that need to be unified with streams messaging capabilities. These three tenets, if properly integrated will ensure a sustainable, scalable, and adaptable end-to-end streaming architecture. Like a three legged stool, one weak tenet can make the entire structure fall short.

This solution brief describes Cloudera's data-in-motion philosophy and is meant as a blueprint to help business and technology decision makers evaluate and simplify their approach to streaming data across their enterprise.

Streaming Architecture in Context

Below we have listed the three tenets that together provide a unified end-to-end streaming architecture.

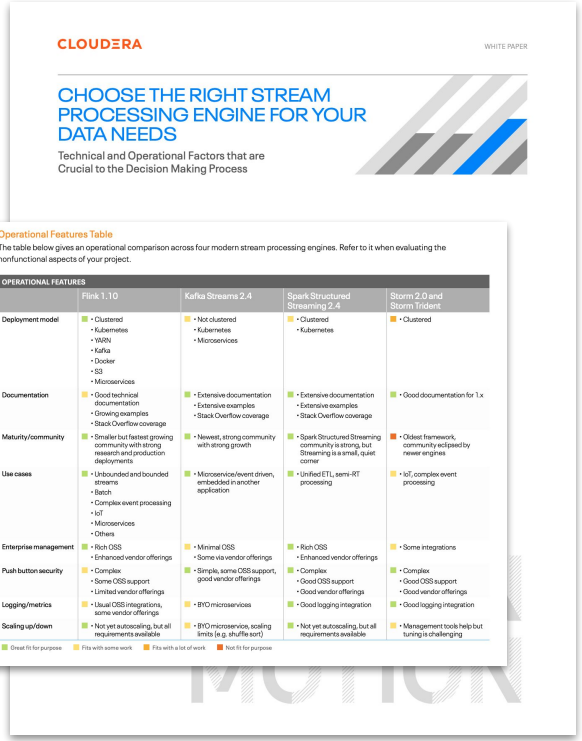
- **Flow management**, broadly speaking, refers to the collection, distribution, and transformation of data across multiple points of producers and consumers
- **Streams messaging** is the provisioning and distribution of messages between producers and consumers
- **Stream processing and analytics** is how you generate real-time analytical insights from the data streaming between producers and consumers

KEY TENETS OF A STREAMING ARCHITECTURE

nifi FLOW MANAGEMENT + kafka STREAMS MESSAGING + Flink STREAM PROCESSING AND ANALYTICS

DATA PIPELINES

Cloudera's data-in-motion philosophy is rooted in the complementary powers that are brought to the table by Apache NiFi for flow management, Apache Kafka for streams messaging, and Apache Flink for stream processing and analytics.


CLOUDERA

WHITE PAPER

CHOOSE THE RIGHT STREAM PROCESSING ENGINE FOR YOUR DATA NEEDS

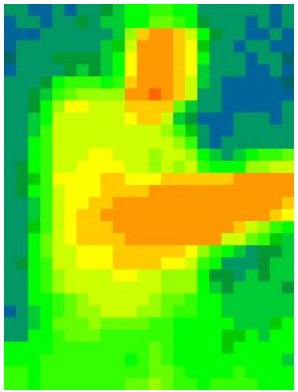
Technical and Operational Factors that are Crucial to the Decision Making Process

Operational Features Table

The table below gives an operational comparison across four modern stream processing engines. Refer to it when evaluating the nonfunctional aspects of your project.

OPERATIONAL FEATURES	Flink 1.10	Kafka Streams 2.4	Spark Structured Streaming 2.4	Storm 2.0 and Storm Trident
Deployment model	• Clustered • Kubernetes • YARN • Kafka • Docker • S3 • Microservices	• Not clustered • Kubernetes • Microservices	• Clustered • Kubernetes	• Clustered
Documentation	• Good technical documentation • Growing examples • Stack Overflow coverage	• Extensive documentation • Extensive examples • Stack Overflow coverage	• Extensive documentation • Extensive examples • Stack Overflow coverage	• Good documentation for 1.x
Maturity/community	• Smaller but fastest growing community with strong research and production deployments	• Newest, strong community with strong growth	• Spark Structured Streaming community is strong, but Streaming is a small, quiet corner	• Oldest framework, community eclipsed by newer engines
Use cases	• Unbounded and bounded streams • Batch • Complex event processing • IoT • Microservices • Others	• Microservice/event driven, embedded in another application	• Unified ETL, semi-RT processing	• IoT, complex event processing
Enterprise management	• Rich OSS • Enhanced vendor offerings	• Minimal OSS • Some via vendor offerings	• Rich OSS • Enhanced vendor offerings	• Some integrations
Push button security	• Complex • Some OSS support • Limited vendor offerings	• Simple, some OSS support, good vendor offerings	• Complex • Good OSS support • Good vendor offerings	• Complex • Good OSS support • Good vendor offerings
Logging/metrics	• Usual OSS integrations, some vendor offerings	• BYO microservices	• Good logging integration	• Good logging integration
Scaling up/down	• Not yet autoscaling, but all requirements available	• BYO microservice, scaling limits (e.g. shuffle sort)	• Not yet autoscaling, but all requirements available	• Management tools help but tuning is challenging

Great fit for purpose | Fits with some work | Fits with a lot of work | Not fit for purpose



THANK YOU