

**CLOUDERA**

**Edge2AI**

Johannes Muselaers | Solutions Engineer Nordics

[jmuselaers@cloudera.com](mailto:jmuselaers@cloudera.com)

+46725881091

THE  
ENTERPRISE  
DATA  
CLOUD  
COMPANY

PDF of presentation:  
<https://github.com/jmuselaers/HI-IS>

# COMBINED STRENGTH

Where we come from



cloudera

DATA  
ANALYTICS

DATA  
IN MOTION

DATA  
AT REST

# NEW COMPANY

Cloudera at a glance

**2000+** **3000+** **3000+**

Customers across all  
markets

[Meet our data heroes](#)

Solution and service  
partners

[Find a partner](#)

Employees doing business  
in 28 countries

[Join us](#)

NEW COMPANY LOGO

CLOUDERA

# OPPORTUNITY



“AI, IoT and Augmented Reality are about to reinvent how industries utilize data—and could drive an era of productivity growth in our cities, farms, factories and hospitals.”

Morgan Stanley Research

# REALITY

91%

ORGANIZATIONS STRUGGLE TO REACH DATA MATURITY

Gartner

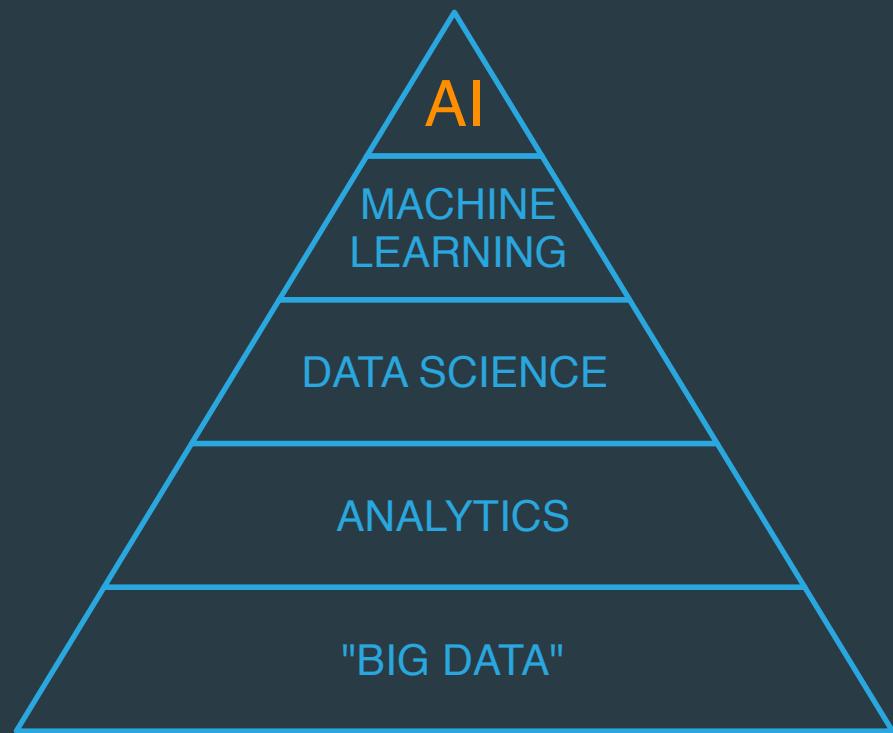
# DIGITAL TRANSFORMATION TOP PRIORITY

- Enterprises become data driven



# HIERARCHY OF NEEDS FOR THE DATA-DRIVEN ENTERPRISE

The “AI Ladder”



## DATA SCIENCE EXPERTS BEST PRACTICE

- You can't do ML without doing data science
- You can't do data science without analytics
- You can't do analytics without Big Data
- They are all dependent on each other, and must operate on the same platform

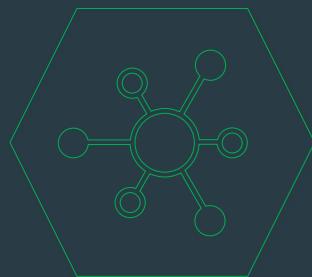
CLOUDERA

Edge2AI

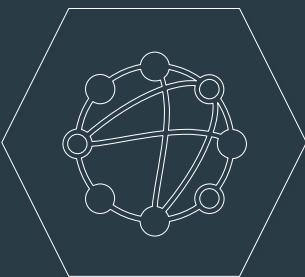
THE  
ENTERPRISE  
DATA  
CLOUD  
COMPANY

# WHERE WE ARE TODAY: DATA FROM ANYWHERE, FROM THE EDGE TO AI

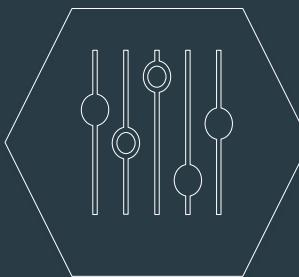
IoT, Ingest &  
Streaming



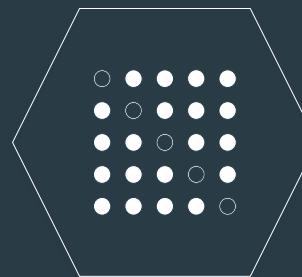
Data  
Engineering



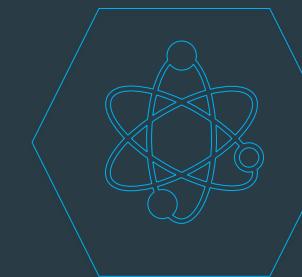
Data  
Warehouse



Operational  
Database



AI, ML &  
Data Science



Edge | Private | Public | Multi | Hybrid

# WHAT'S NEEDED FOR THE FUTURE DATA ARCHITECTURE?

## AGILE INTERACTION MODEL

- Each data application runs on its own virtual private cluster (while sharing the data)
- Strong isolation of tenants and workloads (mitigate noisy neighbors)
- Independent upgrade cycles for components associated with each application

## EFFICIENT/FLEXIBLE COMPUTE MODEL

- Run on virtualized & elastic compute infrastructure (primarily Kubernetes based)
- Scale compute separate from storage (given a solid network backbone architecture)

## UNIFIED SECURITY, MANAGEMENT & METADATA

- Single pane of glass to manage 1000s of applications with a common data substrate
- Automation to drive dozens of application patterns and lifecycles
- Common security, governance, compliance, and metadata catalog

## EASILY MOVE DATA & APPS BETWEEN ON PREMISES AND CLOUD

- Migration tools to easily move data and workloads
- Automatically synchronize metadata/security policies (and unified compliance tools)

# THE ANSWER IS AN ENTERPRISE DATA CLOUD ARCHITECTURE

- Multi-function and multi-tenant analytics
- Hybrid and multi-cloud (migration/synchronization)
- Unified security and governance across workloads and cloud deployments on the same data
- Open platform

IOT, INGEST &  
STREAMING

DATA  
WAREHOUSING

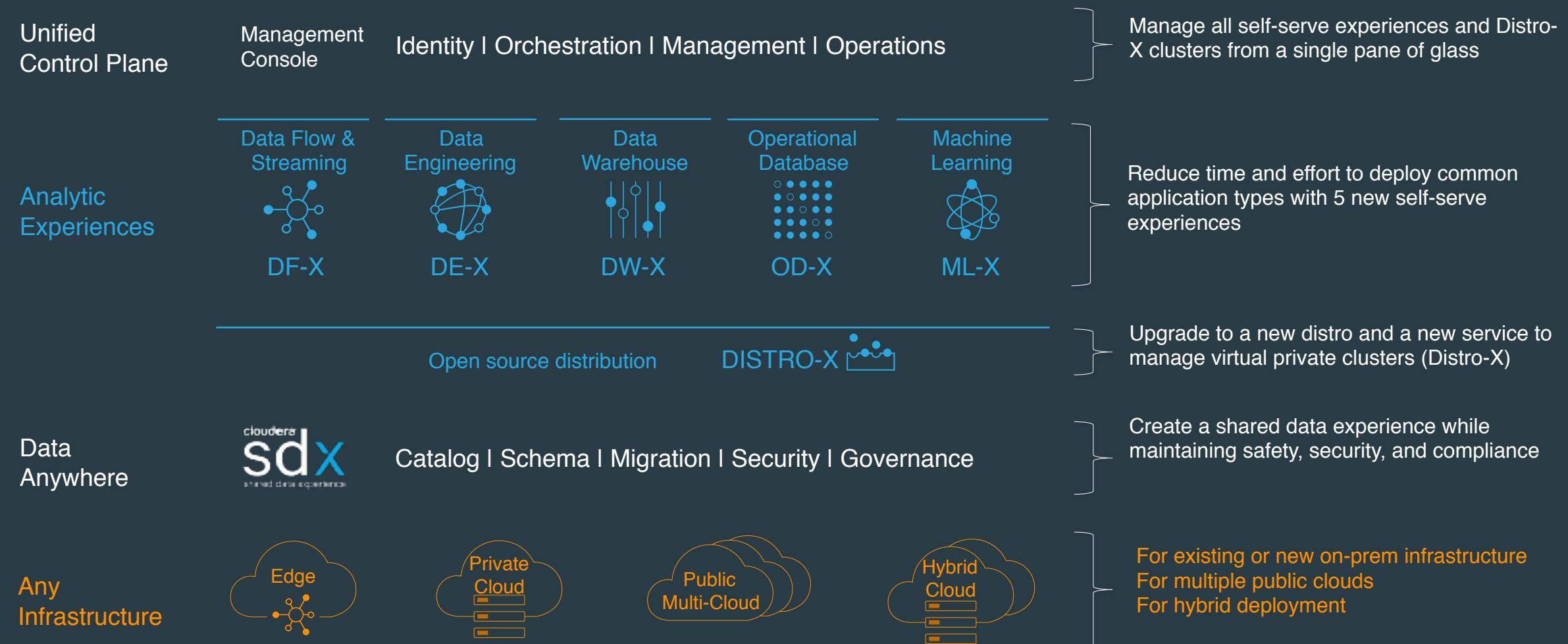
AI/ ML  
DATA SCIENCE

SECURITY & GOVERNANCE

PUBLIC CLOUDS  
compute & storage

DATA CENTER  
compute & storage

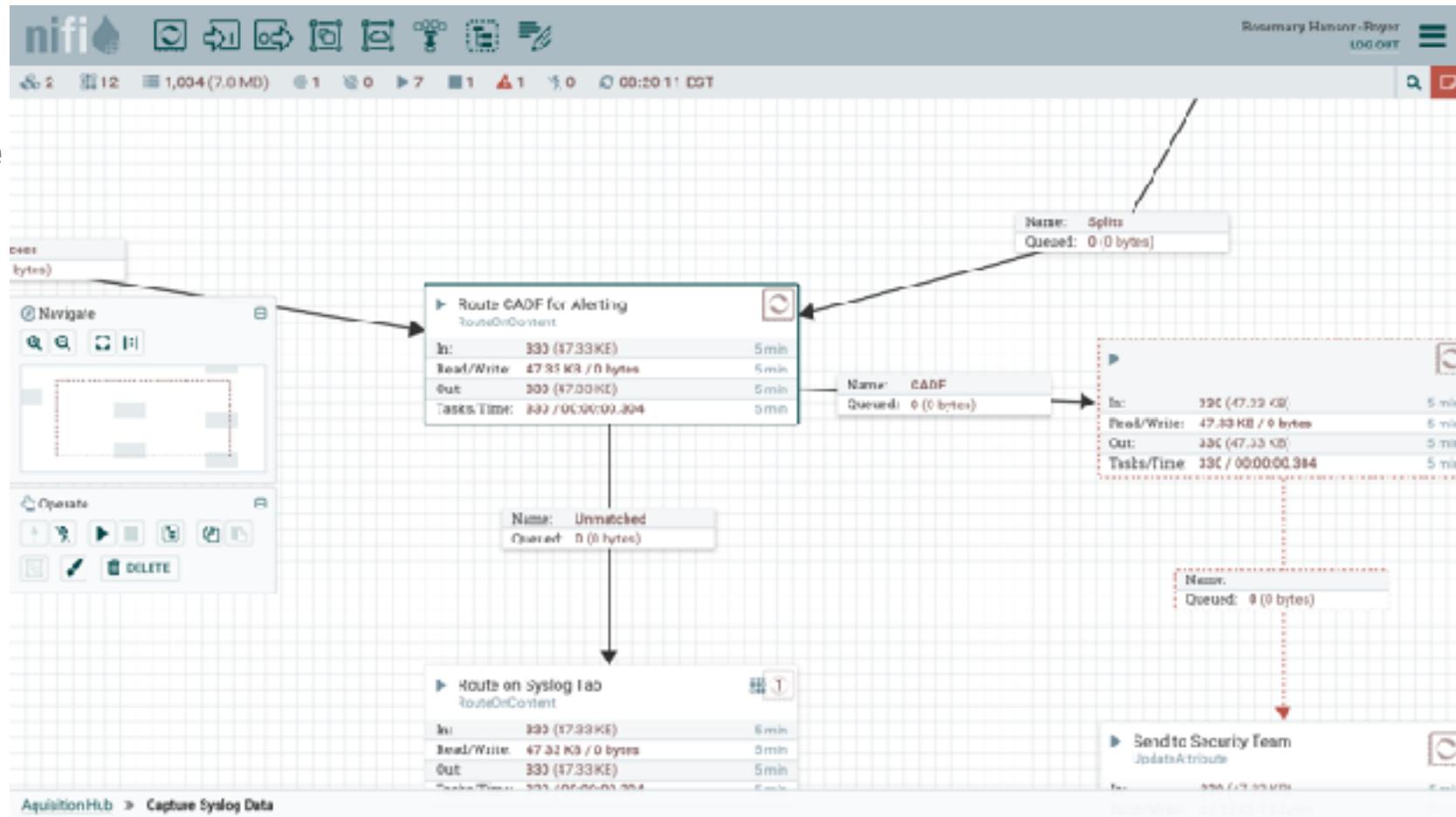
# OUR ANSWER: THE CLOUDERA DATA PLATFORM (CDP)





# FLOW MANAGEMENT

- Web-based user interface
- Highly configurable
- Data provenance
- Designed for extensibility
- Secure
- NiFi Registry
  - DevOps support
  - FDLC
  - Versioning
  - Deployment



# CLOUDERA DATA SCIENCE WORKBENCH

Accelerate Machine Learning from Research to Production



## For data scientists

- **Experiment faster**  
Use R, Python, or Scala with on-demand compute and secure CDH data access
- **Work together**  
Share reproducible research with your whole team
- **Deploy with confidence**  
Get to production repeatably and without recoding

## For IT professionals

- **Bring data science to the data**  
Give your data science team more freedom while reducing the risk and cost of silos
- **Secure by default**  
Leverage common security and governance across workloads
- **Run anywhere**  
On-premises or in the cloud

CLOUDERA

Data Science Workbench  
CDSW

THE  
ENTERPRISE  
DATA  
CLOUD  
COMPANY

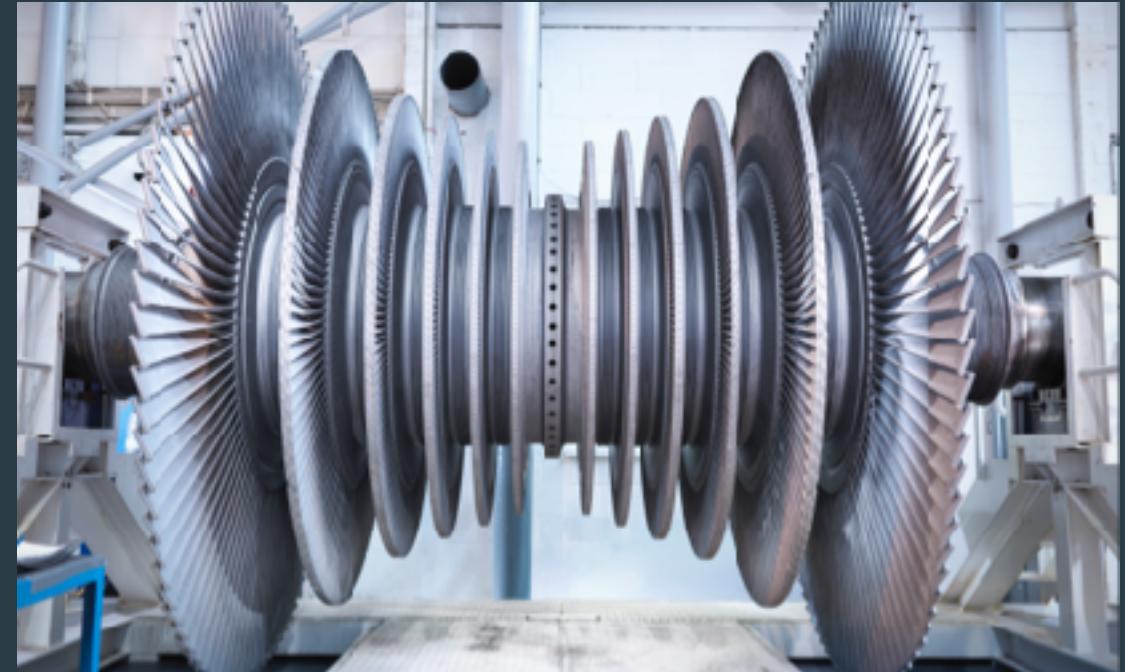
# MACHINE LEARNING AT CLOUDERA

## Our philosophy

We **empower** our customers to run their business on data with an **open platform**:

- Your data
- Open algorithms
- Running anywhere

We accelerate **enterprise** data science.



# What is Cloudera Data Science Workbench (CDSW)

Supports your workloads in an enterprise secure way

Accelerate data science from exploration to production using R, Python, Spark and more

For data scientists



**Open data science, your way.**

Use R, Python, or Scala with your favorite libraries and frameworks



**No need to sample.**

Directly access data in secure Hadoop clusters through Apache Spark and Apache Impala



**Reproducible, collaborative research.**

Share insights with your whole team

For IT professionals



**Bring analysis to the data.**

Give your data science team the freedom to work how they want, when they want



**Secure by default.**

Stay compliant with out-of-the-box support for full Hadoop security



**Flexible deployment.**

Run on-premises or in the cloud

# THE CHALLENGE

Balance these needs

## DATA SCIENTIST

- Access to granular data
- Flexibility
  - Preferred open source tools
- Elastic provisioning
  - Compute
  - Storage
- Reproducible research
- Path to production

VS.



## DevOps/IT

- Security
- Governance
- Standards
- Low maintenance
- Low cost
- Self-service access

---

# THE TYPICAL SOLUTION

“If I can’t use my favorite tools, I’ll...”

- Copy data to my laptop
- Copy data to a data science appliance
- Copy data to a cloud service

Why this is a problem:

- Complicates security
- Breaks data governance
- Adds latency to process
- Makes collaboration more difficult
- Complicates model management and deployment
- Creates infrastructure silos

# CLOUDERA DATA SCIENCE WORKBENCH

Accelerate Machine Learning from Research to Production



## For data scientists

- Experiment faster**  
Use R, Python, or Scala with on-demand compute and secure CDH data access
- Work together**  
Share reproducible research with your whole team
- Deploy with confidence**  
Get to production repeatably and without recoding

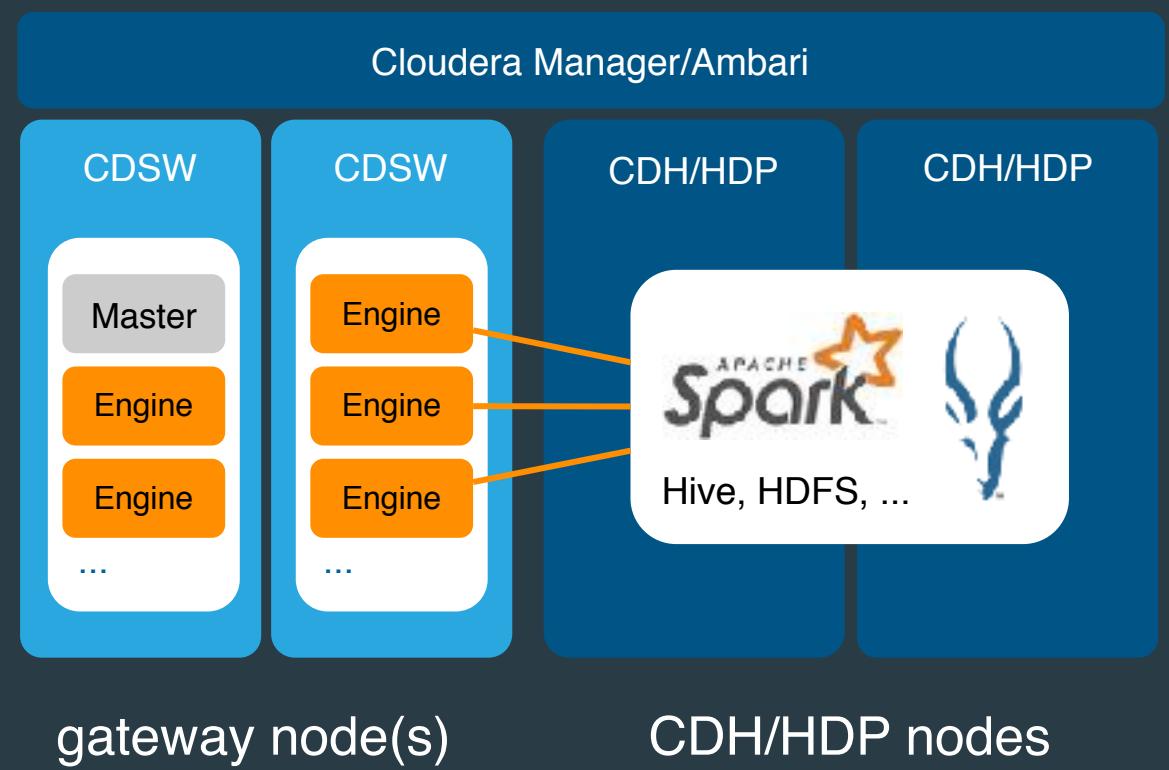
## For IT professionals

- Bring data science to the data**  
Give your data science team more freedom while reducing the risk and cost of silos
- Secure by default**  
Leverage common security and governance across workloads
- Run anywhere**  
On-premises or in the cloud

# A MODERN DATA SCIENCE ARCHITECTURE

## Containerized environments with scalable, on-demand compute

- Built with Docker and Kubernetes
  - Isolated, reproducible user environments
- Supports both big and small data
  - Local Python, R, Scala runtimes
  - Schedule & share GPU resources
  - Run Spark, Impala, and other CDH services
- Secure and governed by default
  - Easy, audited access to Kerberized clusters
  - Leverages SDX platform services
- Deployed with Cloudera Manager/Ambari

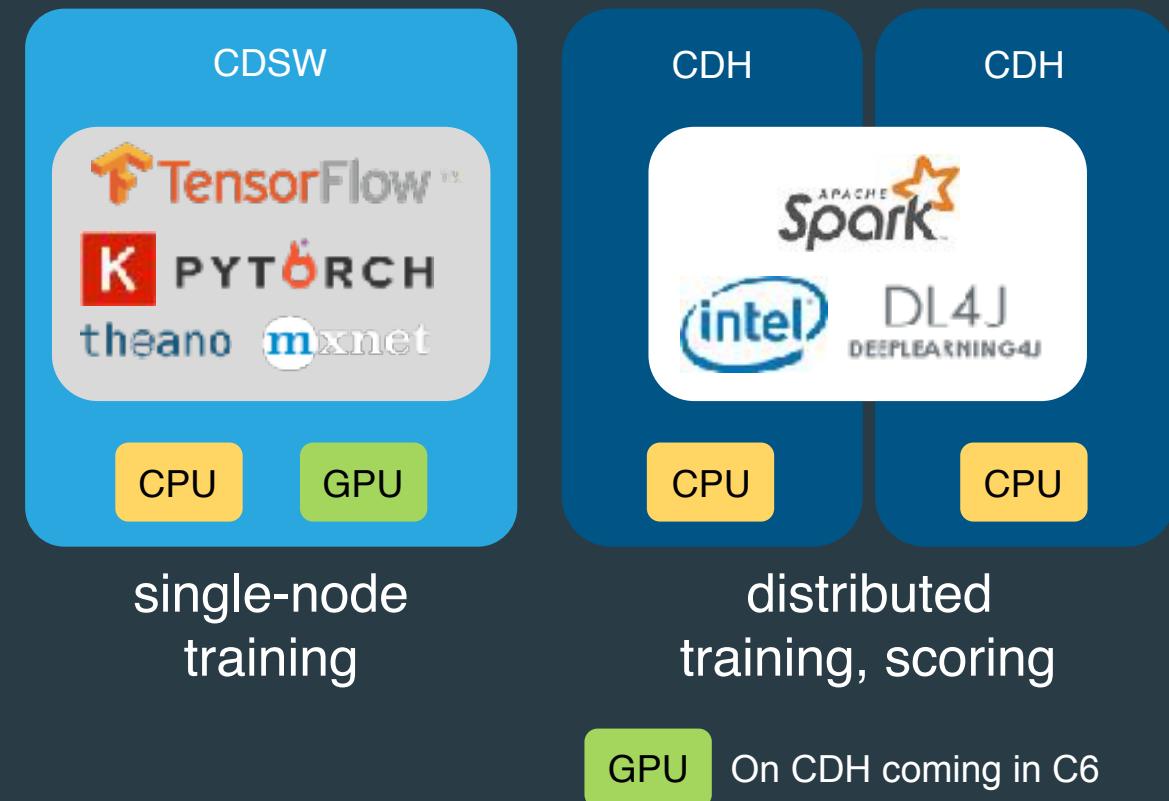


# ACCELERATED DEEP LEARNING WITH GPUS

Multi-tenant GPU support on-premises or cloud

*“Our data scientists want GPUs, but we need multi-tenancy. If they go to the cloud on their own, it’s expensive and we lose governance.”*

- Extend CDSW to deep learning
- Schedule & share GPU resources
- Train on GPUs, deploy on CPUs
- Works **on-premises** or cloud



# WHAT DATA SCIENCE TEAMS DO

---

## PREPARE DATA

---

Ingest data at scale.

Store and secure data.

Clean and transform data  
for analysis.

## BUILD MODELS

---

Explore data and build  
predictive models, offline.

Evaluate and tune models.  
Develop and deliver a  
modeling pipeline.

## DEPLOY MODELS

---

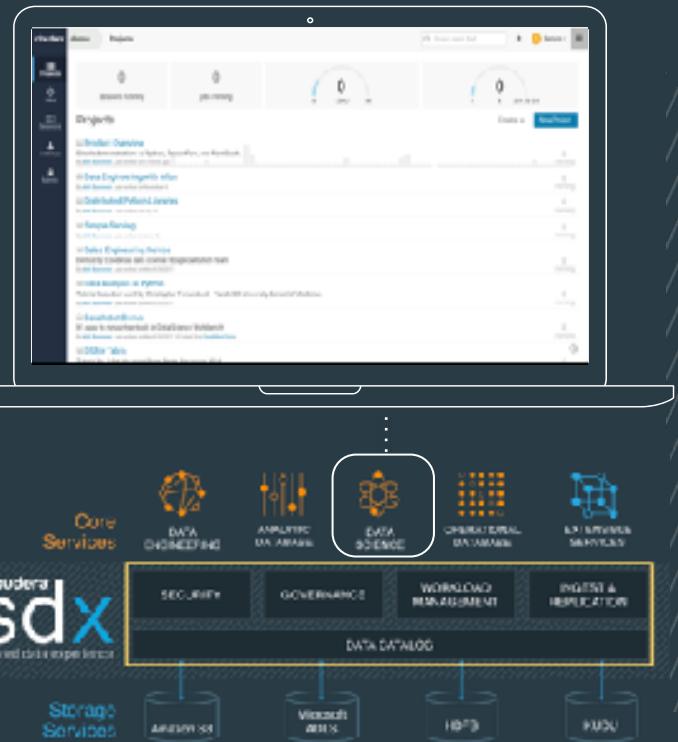
Test, verify, and approve  
model for deployment.

Create and maintain batch/  
stream pipelines,  
embedded models, APIs.

Update models in  
production.

# CLOUDERA DATA SCIENCE WORKBENCH

Accelerate and simplify machine learning from research to production



## ANALYZE DATA

- Explore data securely and share **insights** with the team



## TRAIN MODELS

- Run, track, and compare reproducible experiments



## DEPLOY APIs

- Deploy and monitor models as APIs to serve **predictions**

## MANAGE SHARED RESOURCES

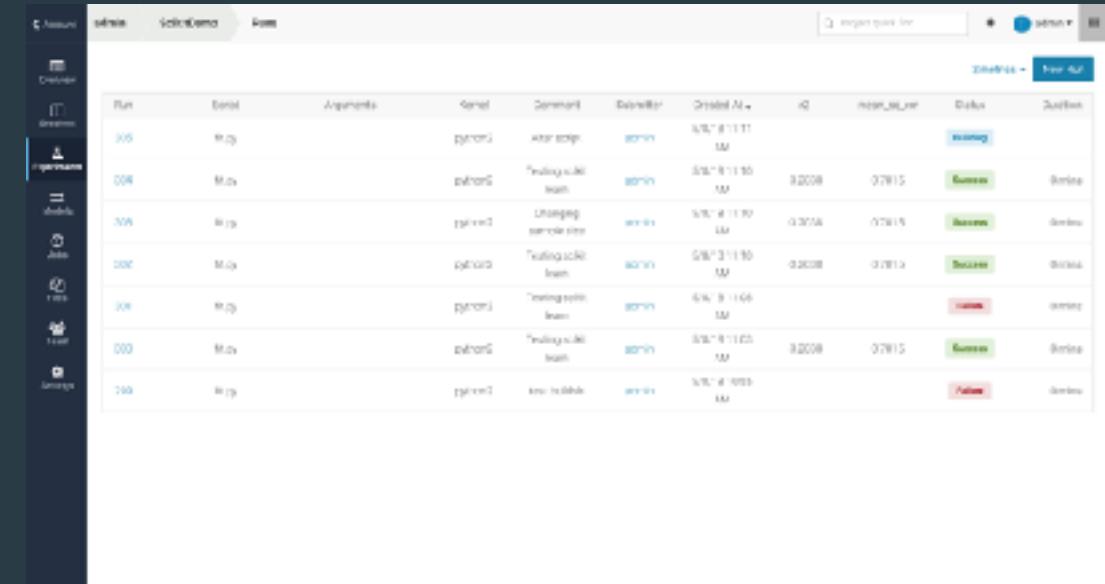
- Provide a secure, collaborative, self-service **platform** for your data science teams

# INTRODUCING EXPERIMENTS

Versioned model training runs for evaluation and reproducibility

Data scientists can now...

- Create a snapshot of model code, dependencies, and configuration necessary to train the model
- Build and execute the training run in an isolated container
- Track specified model metrics, performance, and model artifacts
- Inspect, compare, or deploy prior models



The screenshot shows the Cloudera Machine Learning interface with the 'Experiments' tab selected. The main area displays a table of training runs. The columns include Run ID, Serial, Arguments, Kernel, Environment, Estimated AI, AI ID, model\_file, Status, and Duration. There are seven rows of data, each representing a different experiment run. The 'Status' column includes entries like 'Running', 'Success', 'Warning', and 'Failure'. The 'Duration' column shows times such as '0m 0s', '0m 1s', and '0m 10s'.

Run	Serial	Arguments	Kernel	Environment	Estimated AI	AI ID	model_file	Status	Duration
205	May		PyTorch	Adam SGD	2020-07-11T11:59:59Z	12	model_12.ser	Success	0m 0s
206	May		PyTorch	Testing job 01 batch	2020-07-11T10:59:59Z	12	model_12.ser	Success	0m 0s
207	May		PyTorch	Testing job 02 batch	2020-07-11T10:59:59Z	12	model_12.ser	Success	0m 0s
208	May		PyTorch	Testing job 03 batch	2020-07-11T10:59:59Z	12	model_12.ser	Success	0m 0s
209	May		PyTorch	Testing job 04 batch	2020-07-11T10:59:59Z	12	model_12.ser	Success	0m 0s
210	May		PyTorch	Testing job 05 batch	2020-07-11T10:59:59Z	12	model_12.ser	Success	0m 0s

# INTRODUCING MODELS

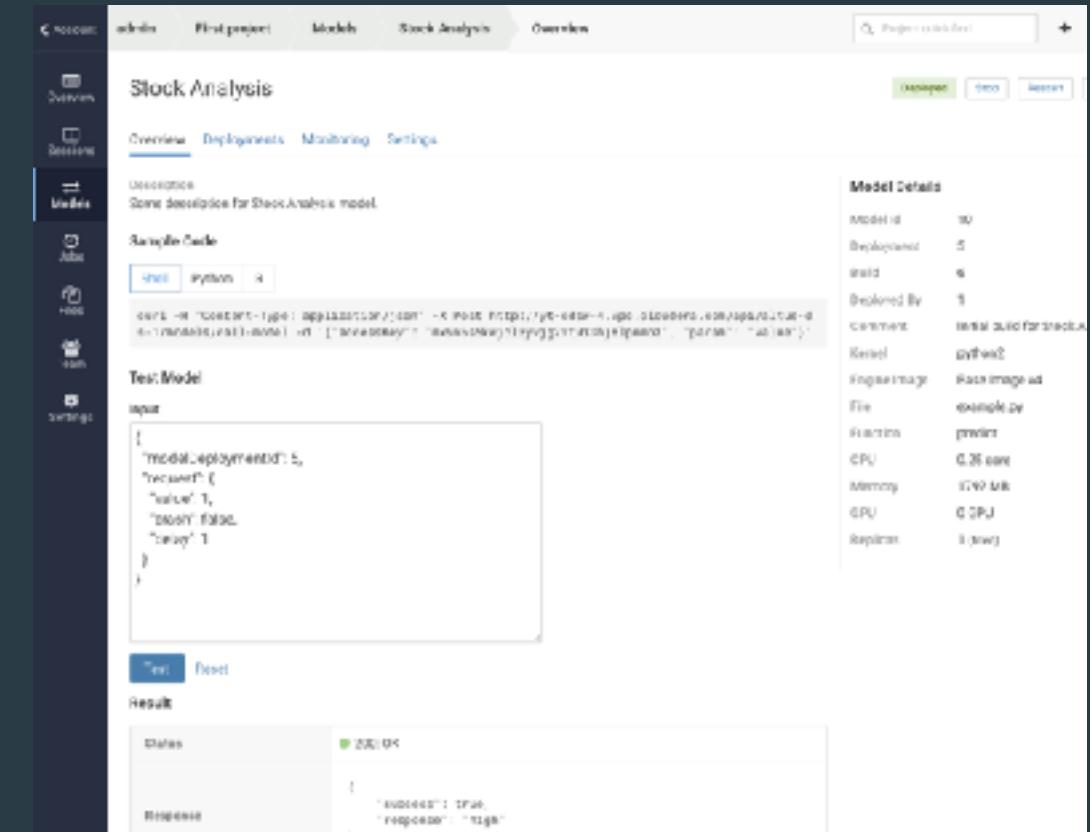
Machine learning models as one-click microservices (REST APIs)

1. Choose file, e.g. score.py
2. Choose function, e.g. forecast

```
f = open('model.pk', 'rb')  
model = pickle.load(f)  
  
def forecast(data):  
    return model.predict(data)
```

3. Choose resources
4. Deploy!

*Running model containers also have access to CDH for data lookups.*



# MODEL MANAGEMENT

View, test, monitor, and update models by team or project

The image displays two side-by-side screenshots of a web-based model management application. Both screenshots show the 'Stock Analysis' project under the 'First project' tab.

**Screenshot 1 (Left): Overview Page**

- Header:** Account, admin, First project, Models, Stock Analysis, Overview.
- Project Details:** Stock Analysis, Model ID 10, Deployments 5, Build 6, Deployed by 1, Current L 1, Kernel python2, Engine Image v4, File example.py, Function predict, CPU 0.29 core, Memory 1792 MB, GPU 0 GPU, Replicas 1 (wait).
- Test Model:** Input JSON: 

```
{"modelDeploymentId": 1, "request": {"value": 1}, "output": false, "deploy": true}
```

 Test button.
- Result:** Status 200 OK, Response: {"success": true, "message": "high"}

**Screenshot 2 (Right): Deployments Page**

- Header:** Account, admin, First project, Models, Stock Analysis, Deployments.
- Table:** Shows deployment details: 5 Deployments, 6 Builds, 1 Deployed, 26 minutes ago, Status Never, Deployed By 1.
- Model Details:** Model ID 10, Name Stock Analysis, Description Some description for Stock Analysis model, LUID hoensdhi7bbcf91, File example.py, Function predict, Kernel python2, Engine Image v4.
- Deployment Details:** ID 5, Deployed At 26 minutes ago, Last Result ID never, Last Updated just now, CPU Intensity 0.25 vCPUs / 0.35 tasks, replicas 1 (used).

0

sessions running

0

jobs running

0

vCPU

0

B

251.53 GB

## Projects

Creator ▾

New Project

### Product Overview

Simple demonstrations of Python, TensorFlow, and R on Spark.

By Matt Brandwein. Last worked on 4 hours ago.

0  
running

### Data Engineering with Altus

By Matt Brandwein. Last worked on November 4.

0  
running

### Distributed Python Libraries

By Matt Brandwein. Last worked on July 19.

0  
running

### Simple Serving

By Matt Brandwein. Last worked on June 10.

0  
running

### Sales Engineering Demos

Demos by Cloudera's data science SE specialization team.

0  
running

By Matt Brandwein. Last worked on March 29, 2012.

### Data Analysis in Python

Tutorial based on work by Christopher Fonnesbeck - Vanderbilt University School of Medicine.

0  
running

By Matt Brandwein. Last worked on March 20, 2012.

### DataRobot Demo

It's easy to run partner tools in Data Science Workbench.

0  
running

By Matt Brandwein. Last worked on March 20, 2012. Forked from [DataRobot Demo](#)

## Product Overview

Simple demonstrations of Python, TensorFlow, and R on Spark.

2 Fork

Open Workbench

### Jobs

Creator ▾

Name	Runs / Failures	Duration	Status	Latest Run	Actions
Nightly Analysis	383 / 1	00:29	Success	4 hours ago	<button>Run</button>

### Files

[Download](#) [New](#) [Upload](#)

	Name	Size	Last Modified
	data	-	April 19
	R	-	December 8
	slides	-	June 1
	1_python.py	2.95 kB	June 20
	2_pyspark.py	1.63 kB	May 8
	3_tensorflow.py	3.38 kB	April 24
	4_sparklyr.R	2.25 kB	April 25
	4a.R	355 B	April 2, 2017
	hello	0 B	June 20

2\_pyspark.py

1\_python.py

4\_sparklyr.R

Product Overview

1\_python.py

2\_pyspark.py

3\_tensorflow.py

4\_sparklyr.R

4.R

▼ data

GoogleTrendsData.csv

kmeans\_data.txt

▶ MINIST

hello

▶ R

README.md

▶ slides

utils.py

utils.pyc

```

1 # Google Stock Analytics
2 # =====
3 #
4 # This notebook implements a strategy that uses Google
5 # trade the Dow Jones Industrial Average.
6
7 import pandas as pd
8 import matplotlib.pyplot as plt
9 import matplotlib as mpl
10 from pandas_highcharts.display import display_charts
11 import seaborn
12 mpl.rcParams['font.family'] = 'Source Sans Pro'
13 mpl.rcParams['axes.labelsize'] = '15'
14
15 # Import Data
16 # =====
17 #
18 # Load data from Google Trends.
19
20 data = pd.read_csv('data/GoogleTrendsData.csv', index_
21 data.head()
22
23 # Show DJIA vs. debt related query volume.
24 display_charts(data, chart_type="stock", title="DJIA v_
25 seaborn.lmplot("debt", "djia", data=data, size=7)
26
27 # Detect if search volume is increasing or decreasing :
28 # any given week by forming a moving average and testin
29 # crosses the moving average of the past 3 weeks.
30 #
31 # Let's first compute the moving average.
32
33 data['debt_mavg'] = data.debt.rolling(window=3, center_
34 data.head()
35
36 # Since we want to see if the current value is above th
37 # *preceding* weeks, we have to shift the moving avera
38
39 data['debt_mavg'] = data.debt_mavg.shift(1)
40 data.head()
41
42 # Generate Predictions

```

## Start New Session

### Engine Image - [Configure](#)

Base Image v1 - docker.repository.cloudera.com/cdsweb/engine:1

### Select Engine Kernel

- Python 2
- Python 3
- Scala
- R

### Select Engine Profile

1 vCPU / 2 GiB Memory

**Launch Session**

File Edit View Navigate Run ▶ 1\_python.p...

2\_pyspark.py  
1\_python.py  
4\_sparklyr.R

Product Overview ↗  
1\_python.py  
2\_pyspark.py  
3\_tensorflow.py  
4\_sparklyr.R  
40.R

▼ data  
  GoogleTrendsData.csv  
  kmeans\_data.txt  
  ▶ MINIST  
  hello  
▶ R  
  README.md  
▶ slides  
  utils.py  
  utils.pyc

```

1 # Google Stock Analytics
2 # =====
3 #
4 # This notebook implements a strategy that uses Google
5 # to trade the Dow Jones Industrial Average.
6
7 import pandas as pd
8 import matplotlib.pyplot as plt
9 import matplotlib as mpl
10 from pandas_highcharts.display import display_charts
11 import seaborn
12 mpl.rcParams['font.family'] = 'Source Sans Pro'
13 mpl.rcParams['axes.labelsize'] = 16
14
15 # Import Data
16 # =====
17 #
18 # Load data from Google Trends.
19
20 data = pd.read_csv('data/GoogleTrendsData.csv', index_col='Date')
21 data.head()
22
23 # Show DJIA vs. debt related query volume.
24 display_charts(data, chart_type="stock", title="DJIA v
25 seaborn.lmplot("debt", "djia", data=data, size=7)
26
27 # Detect if search volume is increasing or decreasing
28 # any given week by forming a moving average and testing
29 # crosses the moving average of the past 3 weeks.
30
31 # Let's first compute the moving average.
32
33 data['debt_mavg'] = data.debt.rolling(window=3, center=True).mean()
34 data.head()
35
36 # Since we want to see if the current value is above the
37 # proceeding 3 weeks, we have to shift the moving average
38
39 data['debt_mavg'] = data.debt_mavg.shift(1)
40 data.head()
41
42 # Generate Orders
43 # =====

```

Project Terminal access Clear Interrupt Stop Sessions ▾

## My Python Session

By Matt Brandwein — Python 2 Session — 1 vCPU / 2 GiB Memory — just now

Cloudera Data Science Workbench Terminal

⑥ 1qp7dhoexb4sn8frcdsw.edh.cloudera.com/fky3o8qgcyi0n49p/

### Google Trends Data

Welcome to Cloudera Data Science Workbench

This notebook Kernel: python2

Industrial Average Project workspace: /home/cdsw

> import pandas  
> import numpy  
> import math  
> import matplotlib.pyplot as plt  
> import matplotlib as mpl  
> import seaborn  
> mpl.rcParams['font.family'] = 'Source Sans Pro'  
> mpl.rcParams['axes.labelsize'] = 16  
> # Import Data  
> # =====  
> # Load data from Google Trends.  
>  
> data = pd.read\_csv('data/GoogleTrendsData.csv', index\_col='Date')  
> data.head()  
>  
> # Show DJIA vs. debt related query volume.  
> display\_charts(data, chart\_type="stock", title="DJIA v  
> seaborn.lmplot("debt", "djia", data=data, size=7)  
>  
> # Detect if search volume is increasing or decreasing  
> # any given week by forming a moving average and testing  
> # crosses the moving average of the past 3 weeks.  
>  
> # Let's first compute the moving average.  
>  
> data['debt\_mavg'] = data.debt.rolling(window=3, center=True).mean()  
> data.head()  
>  
> # Since we want to see if the current value is above the  
> # proceeding 3 weeks, we have to shift the moving average  
>  
> data['debt\_mavg'] = data.debt\_mavg.shift(1)  
> data.head()  
>  
> # Generate Orders  
> # =====

Runtimes:

R: R version 3.3.0 (2016-05-03) — "Supposedly Educational"  
Python 2: Python 2.7.11  
Python 3: Python 3.6.1  
Java: java version "1.8.0\_111"

Git origin: http://github.mtv.cloudera.com/mbrandwein/cdsw-demo-sh

Import Data

cdsw@1qp7dhoexb4sn8frcdsw.edh.cloudera.com:~\$ ls -al  
total 96  
drwxr-xr-x 14 cdsw cdsw 4096 Jul 14 2017 .

Load data from

> data = pd.read\_csv('data/GoogleTrendsData.csv', index\_col='Date', parse\_dates=True)  
> data.head()

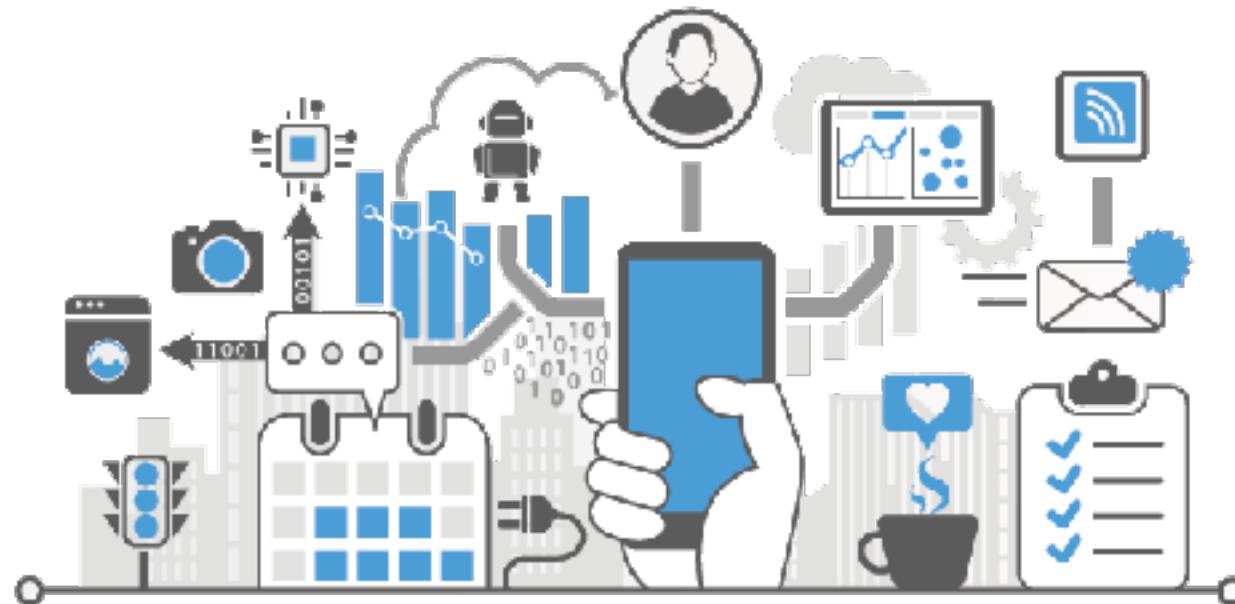
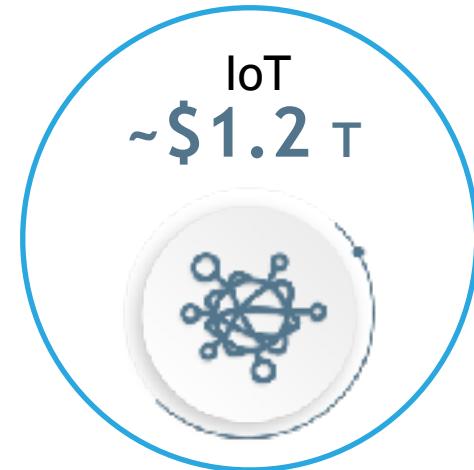
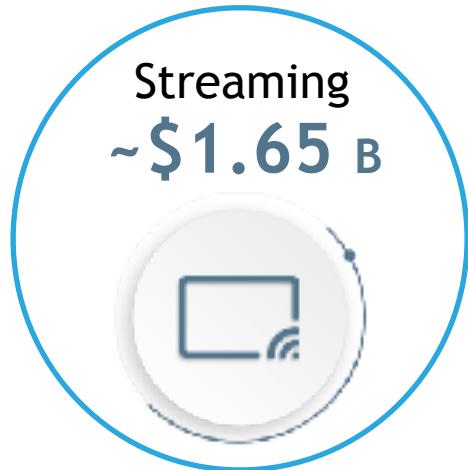
	djia	debt
Date		
2004-01-14	10485.18	0.210000
2004-01-22	10528.66	0.210000

**CLOUDERA**

Cloudera Data Flow  
CDF

THE  
ENTERPRISE  
DATA  
CLOUD  
COMPANY

# MARKET OPPORTUNITIES



# IOT MARKET

24.9B

**By 2024 more than 24.9 Billion IoT connections will be established**

\$70B

**An estimated \$70 billion will be spent by global manufacturers on IoT solutions in 2020**

646M

**An estimated 646 million healthcare devices (excluding fitness trackers and wearable devices) will be connected by 2020**

78%

**An estimated 78% of cars shipped globally will be built with hardware that connects to the internet by 2020**

50%

**50% of decision-makers in IT, services, utilities, and manufacturing have either deployed IoT, or will deploy it in the next 12-24 months**

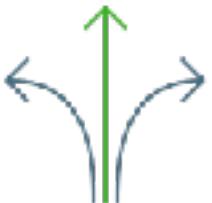
# PROBLEMS IN THE MARKET – PAIN THE CUSTOMER EXPERIENCES



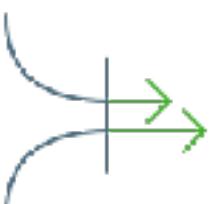
Data movement



Continuous data ingestion



Streaming ETL



Streaming analytics

# COMMON USE CASES

## Data Movement

Optimize resource utilization by moving data between data centers or between on-premises infrastructure and cloud infrastructure

## Optimize Log Collection & Analysis

Optimize log analytics solutions by using CDF as a single platform to collect and deliver multiple data sources

## Gain key insights with Streaming Analytics

Accelerate big data ROI by analyzing streaming data for patterns, comparing with ML models and delivering actionable intelligence

## Single view / 360° view of customer

Ingest, transform and combine customer data from multiple sources into a single data view / lake

## Stream Processing

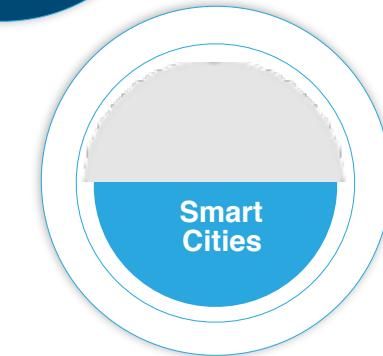
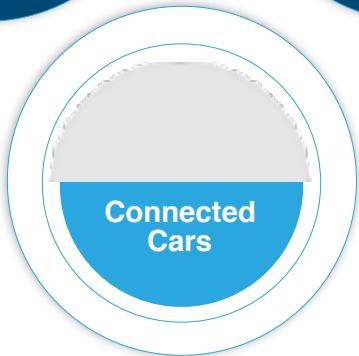
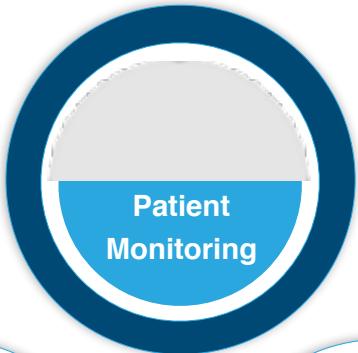
Combine multiple streams of data in real-time, enrich the data and route it to different end points based on rules

## Capture and Analyze IoT Data

Ingest sensor data from IoT devices and stream it for further processing and comprehensive analysis

# COMMON IOT USE CASES BY INDUSTRY

Top 5  
Use cases



Public Sector

Transportation

Utilities

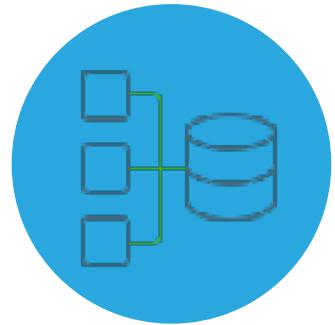
Healthcare

Manufacturing

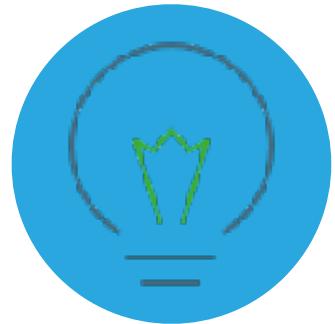
Retail

- IoT is a \$1.13T market opportunity in 2021.
- Americas - \$329B IoT spending. Manufacturing and Transportation are top industries, accounting for 26% of total spending.
- APAC - \$500B IoT spending. Manufacturing, Utilities and Transportation are top industries.
- EMEA - \$264B IoT spending. Manufacturing is top industry, powered by Industry 4.0 initiatives.
- Worldwide IoT Analytics and Information Management Market = \$573M

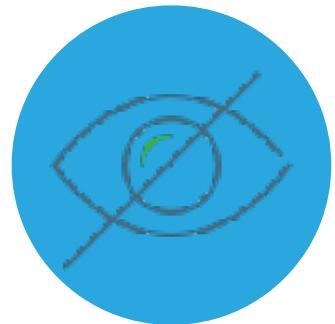
# KEY CUSTOMER CHALLENGES



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



**Real-time Insights:** Analyzing continuous and rapid inflow (velocity) of streaming data at high volumes creates major challenges for gaining real-time insights



**Visibility:** Lack visibility of end-to-end streaming data flows, inability to troubleshoot bottlenecks, consumption patterns etc.

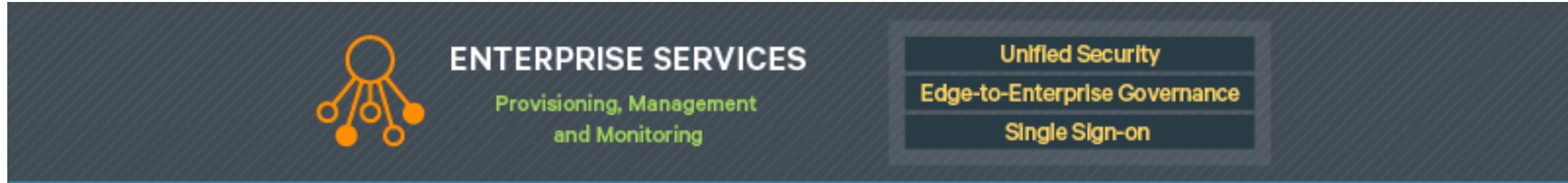
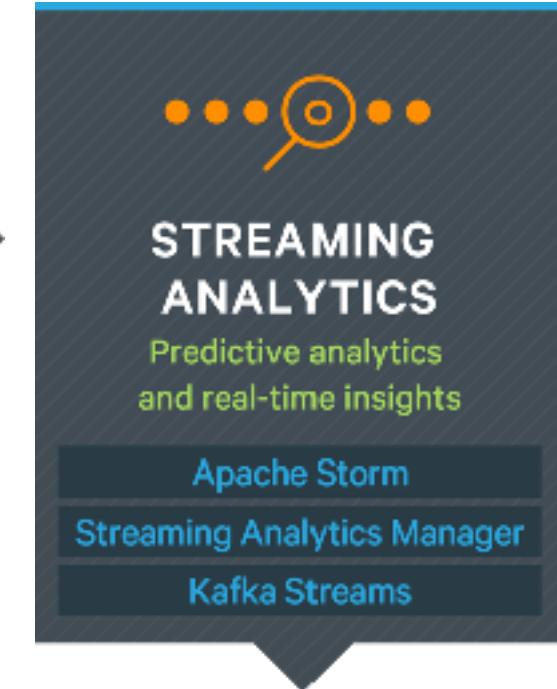
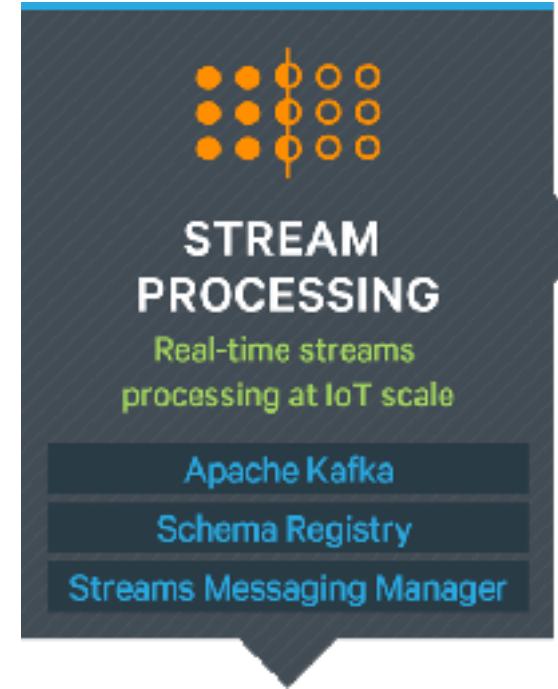
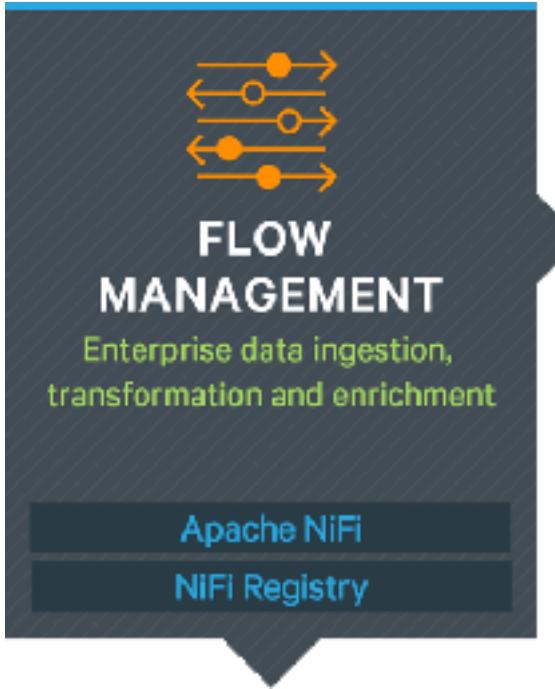
---

# PRODUCT OVERVIEW

# WHAT IS CLOUDERA DATAFLOW (CDF)?

**Cloudera DataFlow (CDF)** is a scalable, real-time streaming data platform that collects, curates, and analyzes data so customers gain key insights for immediate actionable intelligence.

# CLOUDERA DATAFLOW



# CLOUDERA DATAFLOW



# WHAT IS CLOUDERA EDGE MANAGEMENT (CEM)?

**Cloudera Edge Management (CEM)** is an edge management solution made up of edge agents and an edge management hub. It manages, controls and monitors edge agents to collect data from edge devices and push intelligence back to the edge. CEM allows you to develop, deploy, run and monitor edge flow apps on thousands of edge devices.

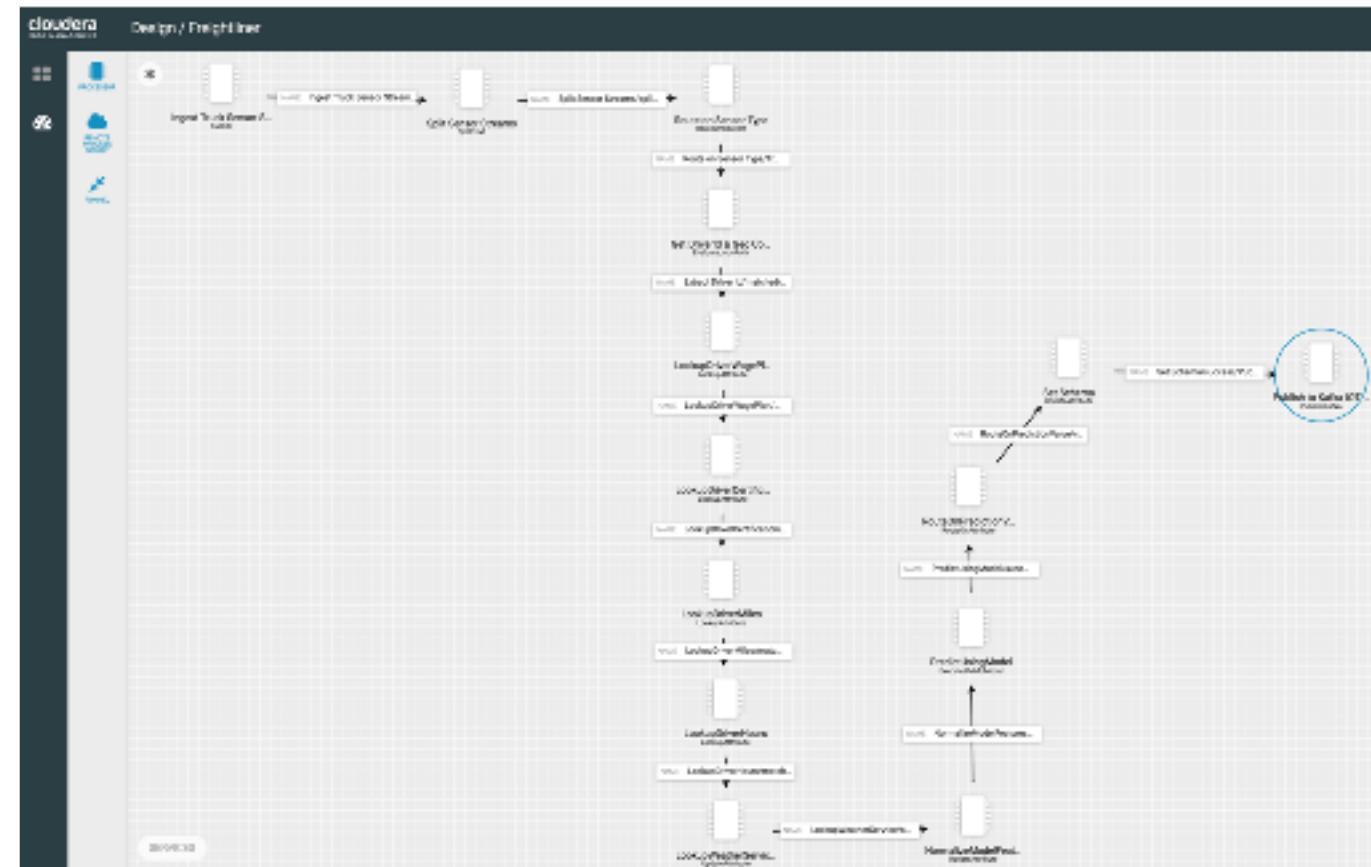
# EDGE DATA MANAGEMENT

- Edge data collection powered by Apache MiNiFi
- MiNiFi – smaller footprint than NiFi
  - Guaranteed delivery
  - Data buffering
  - Prioritized queuing
  - Flow-specific QoS
  - Data provenance
  - Designed for extension
  - C++ / Java agents
  - TensorFlow support
- Designed for IoT

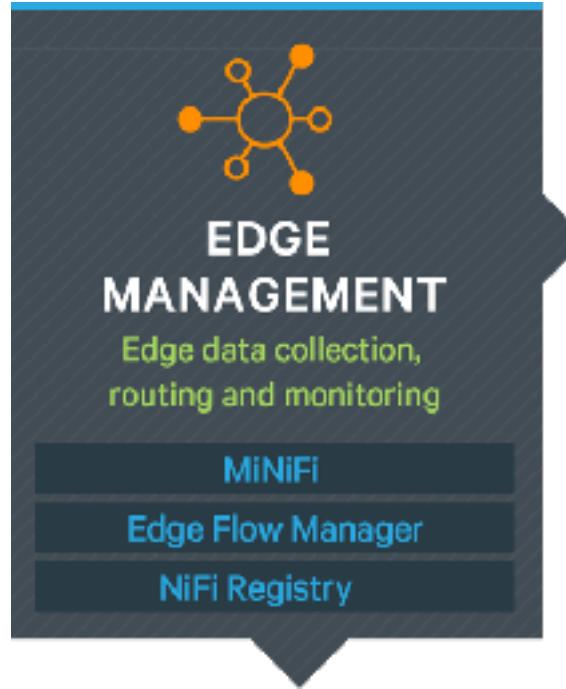


# EDGE FLOW MANAGER

- Edge management hub
  - NiFi-like user interface to develop and deploy flow files to the edge
  - Update and deploy ML model files to the edge agents
  - Monitor thousands of edge agents
  - Integration with NiFi Registry



# CLOUDERA DATAFLOW

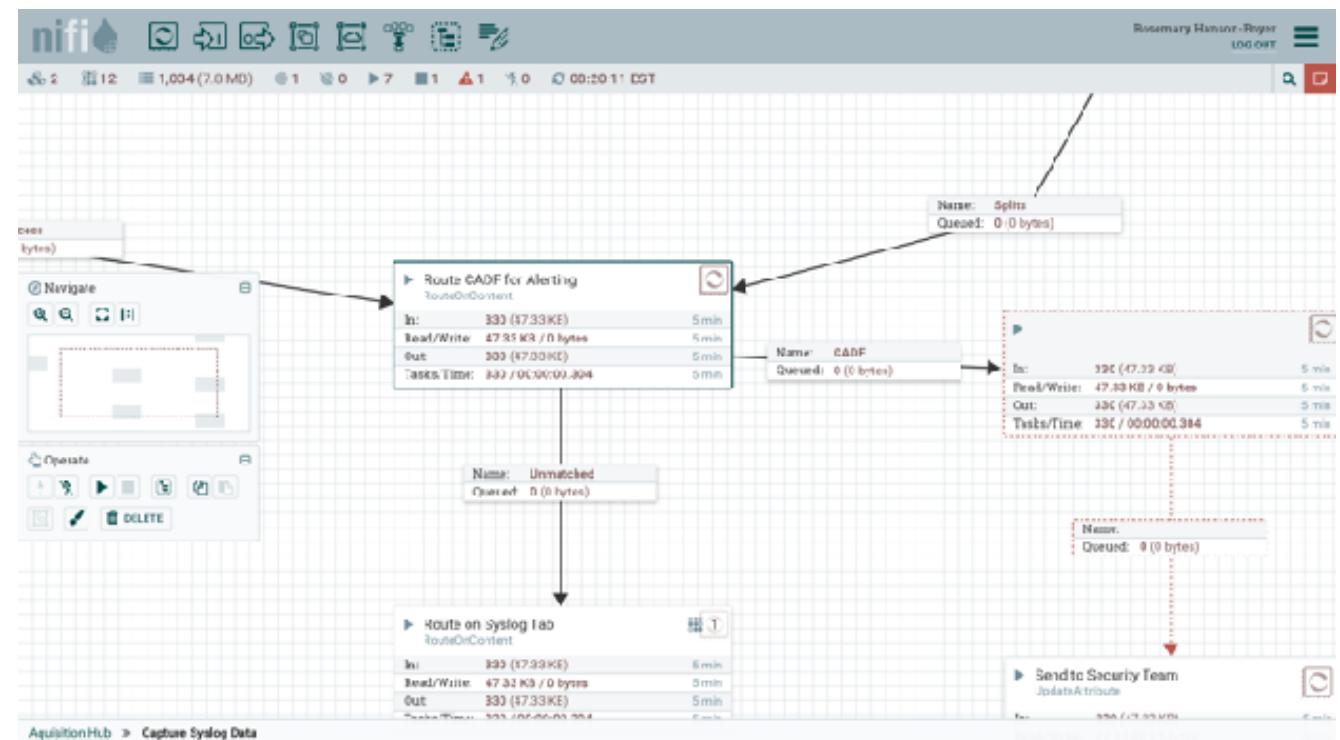


# WHAT IS CLOUDERA FLOW MANAGEMENT (CFM)?

**Cloudera Flow Management (CFM)** is a no-code data ingestion and management solution powered by Apache NiFi. With NiFi's intuitive graphical interface and 300+ processors, CFM delivers highly scalable data movement, transformation and management capabilities to the enterprise. CFM also enables DevOps type development and deployment with its support for NiFi Registry.

# FLOW MANAGEMENT

- Web-based user interface
- Highly configurable
- Out-of-the-box data provenance
- Designed for extensibility
- Secure
- NiFi Registry
  - DevOps support
  - FDLC
  - Versioning
  - Deployment



# 300+ PROCESSORS FOR DEEPER ECOSYSTEM INTEGRATION

FTP
SFTP
HL7
UDP
XML
⋮
HTTP
WebSocket
Email
HTML
Image
Syslog
AMQP

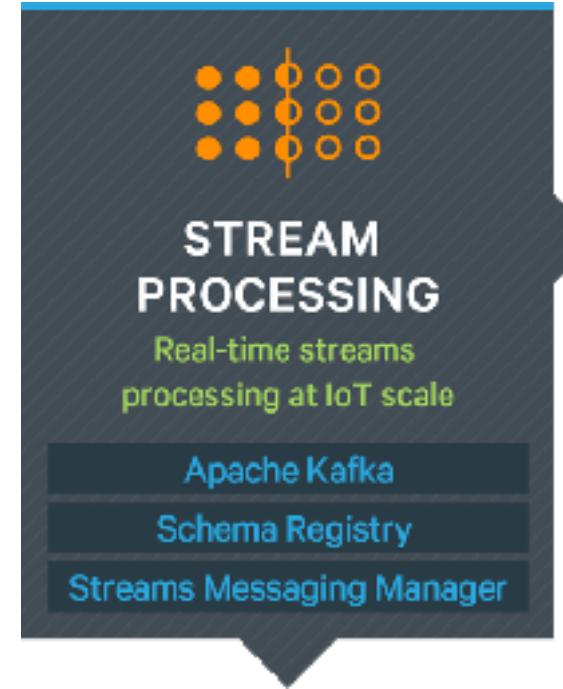
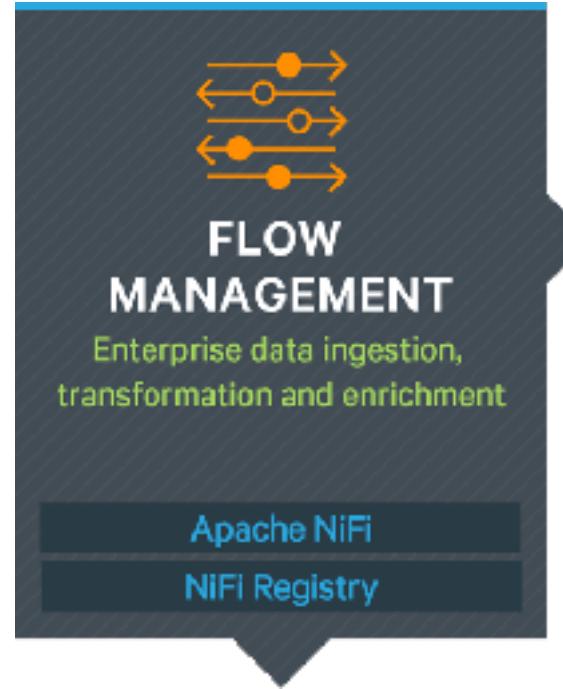
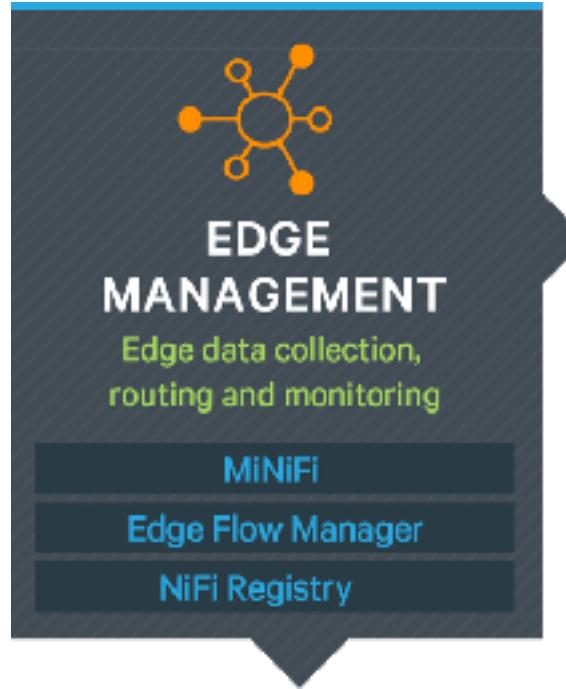


Hash	Encrypt	GeoEnrich
Merge	Tail	Scan
Extract	Evaluate	Replace
Duplicate	Execute	Translate
Split	Fetch	Convert

Route Text	Distribute Load
Route Content	Generate Table Fetch
Route Context	Jolt Transform JSON
Control Rate	Prioritized Delivery

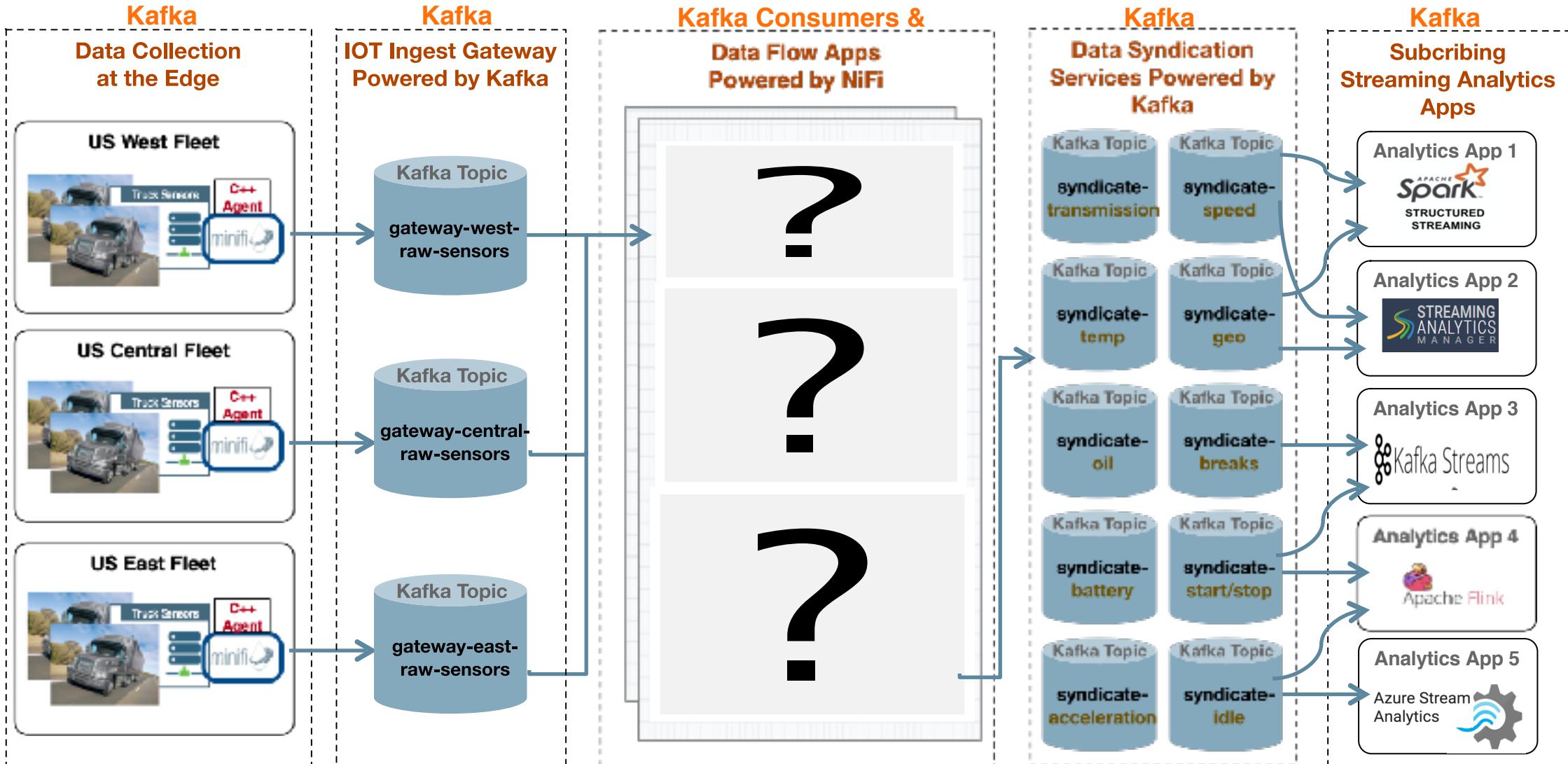
All Apache project logos are trademarks of the ASF and the respective projects.

# CLOUDERA DATAFLOW



# Streaming Analytics Reference Architecture

*Kafka is Everywhere. Critical Component of Streaming*

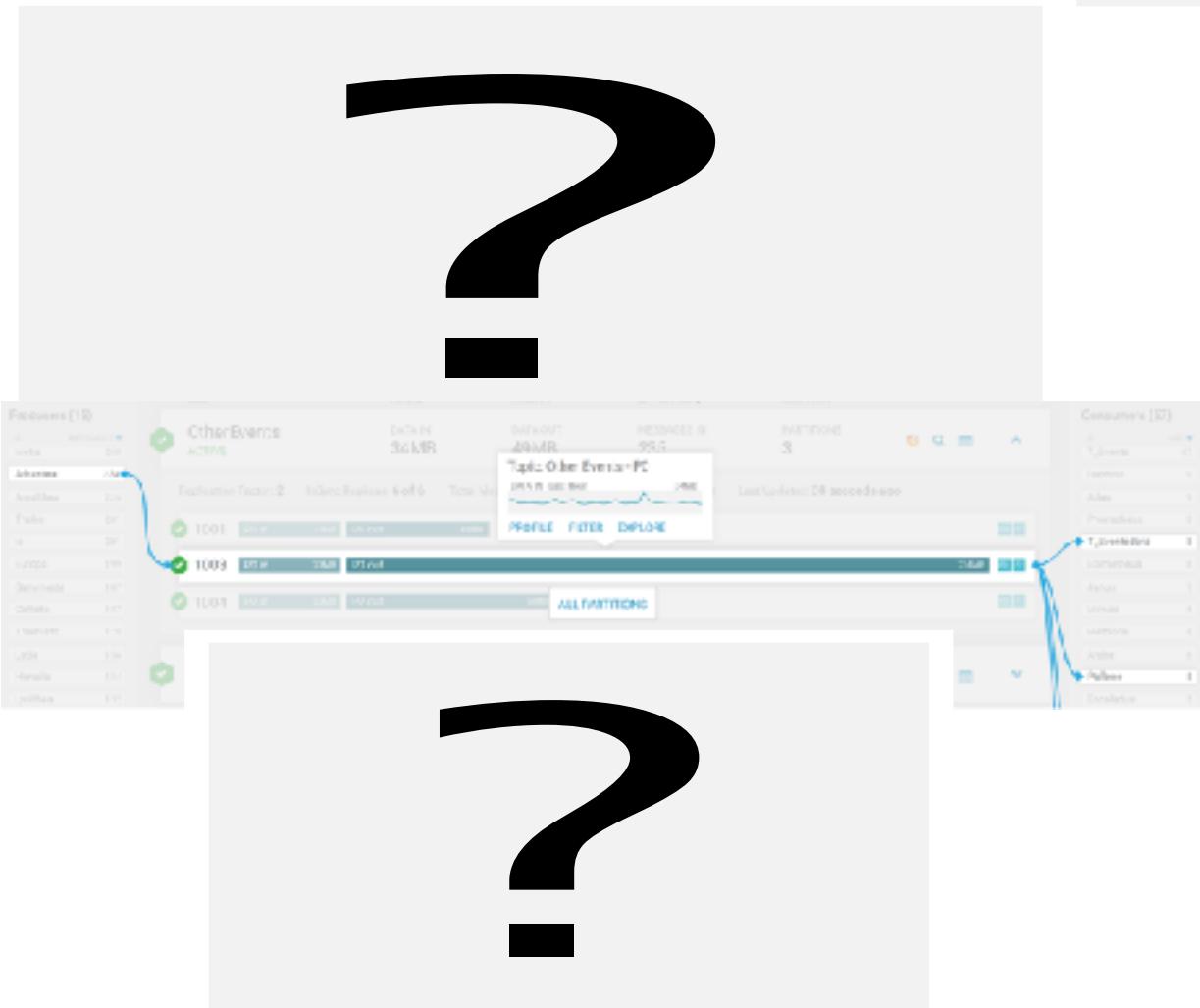


# Cloudera Streams Messaging Manager (SMM)



## What is SMM?

- ◆ Kafka Management and Monitoring tool
- ◆ Cure the “Kafka Blindness”
- ◆ Single Monitoring Dashboard for all your Kafka Clusters across 4 entities
  - Broker
  - Producer
  - Topic
  - Consumer
- ◆ REST as a First Class Citizen



# CLOUDERA DATAFLOW



## EDGE MANAGEMENT

Edge data collection,  
routing and monitoring

MiNiFi

Edge Flow Manager

NiFi Registry



## FLOW MANAGEMENT

Enterprise data ingestion,  
transformation and enrichment

Apache NiFi

NiFi Registry



## STREAM PROCESSING

Real-time streams  
processing at IoT scale

Apache Kafka

Schema Registry

Streams Messaging Manager



## STREAMING ANALYTICS

Predictive analytics  
and real-time insights

Apache Storm

Streaming Analytics Manager

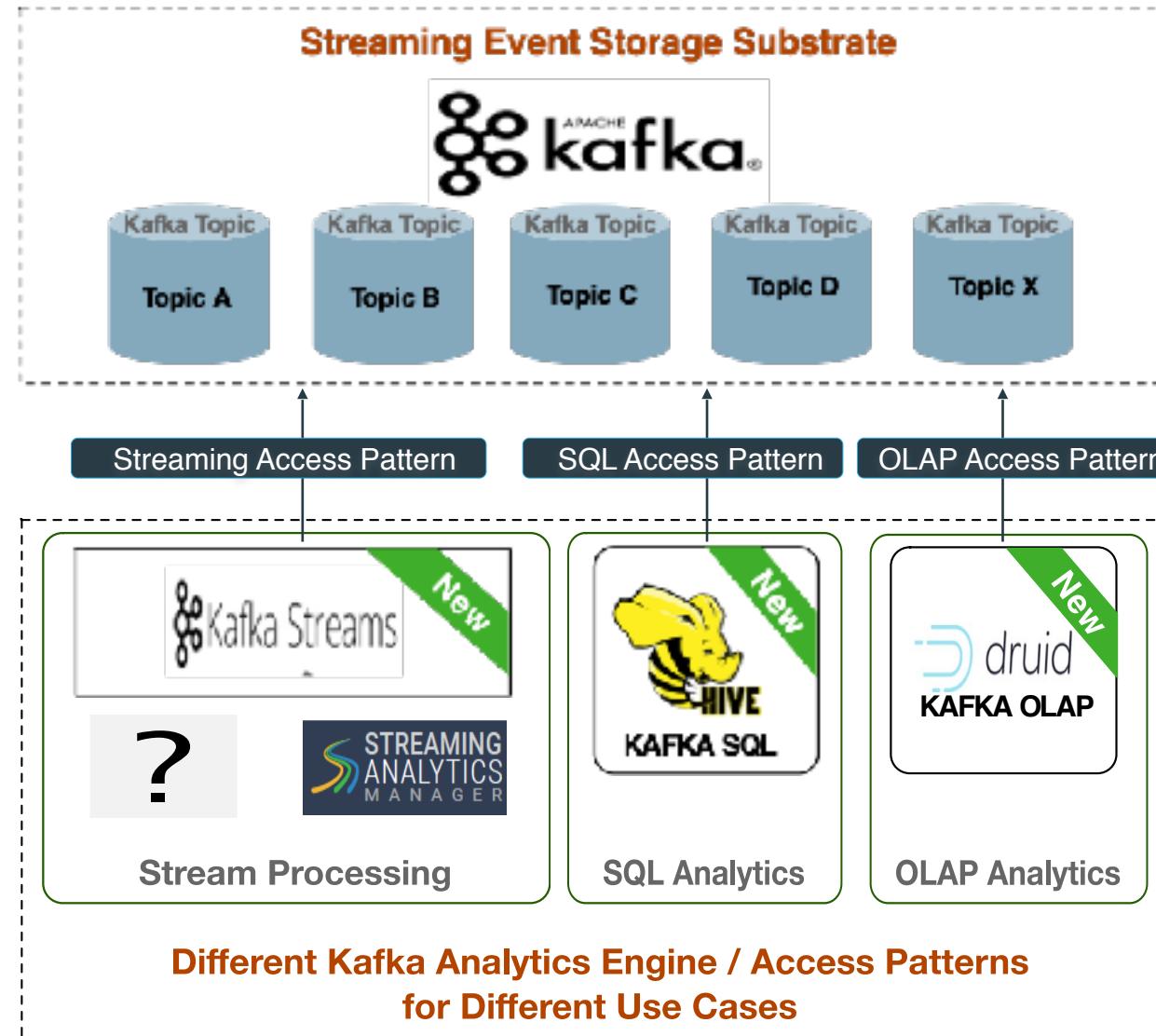
Kafka Streams

# STREAMING ANALYTICS

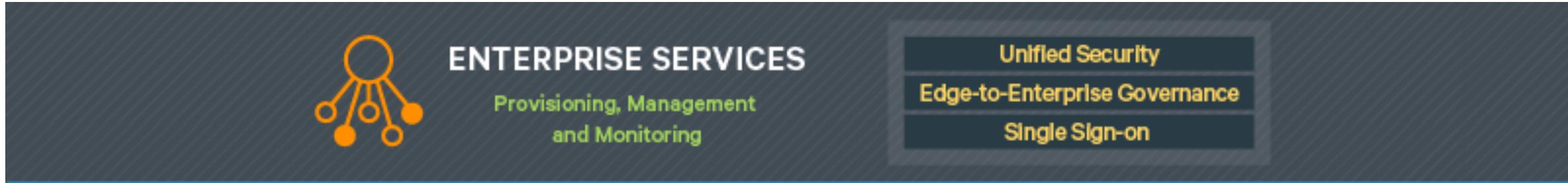
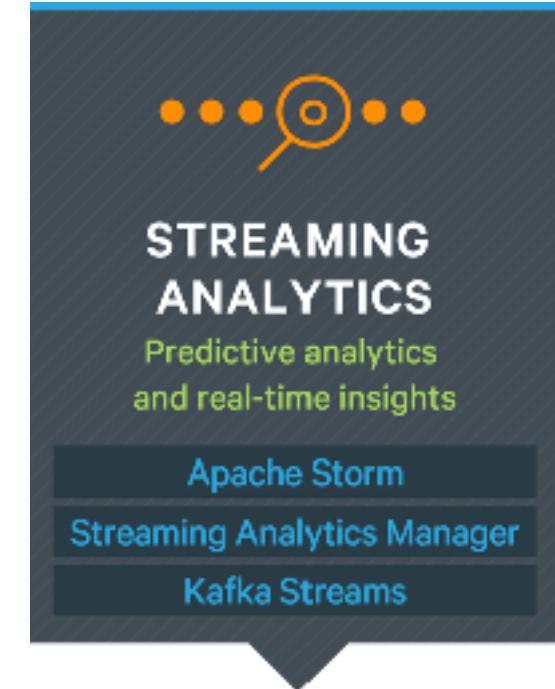
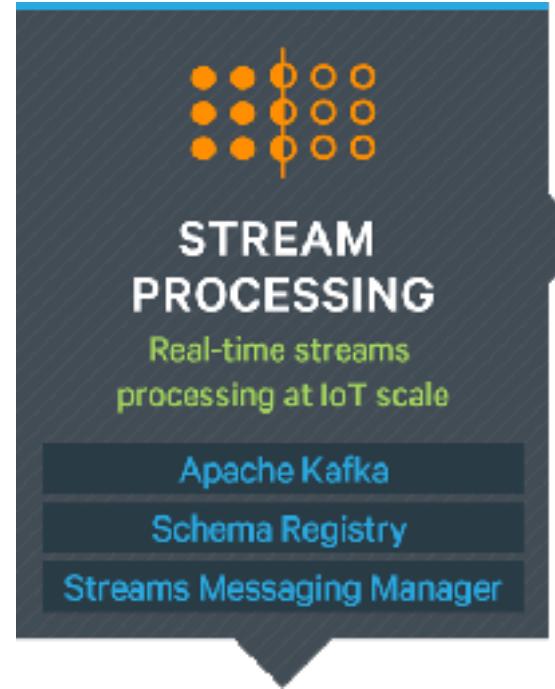
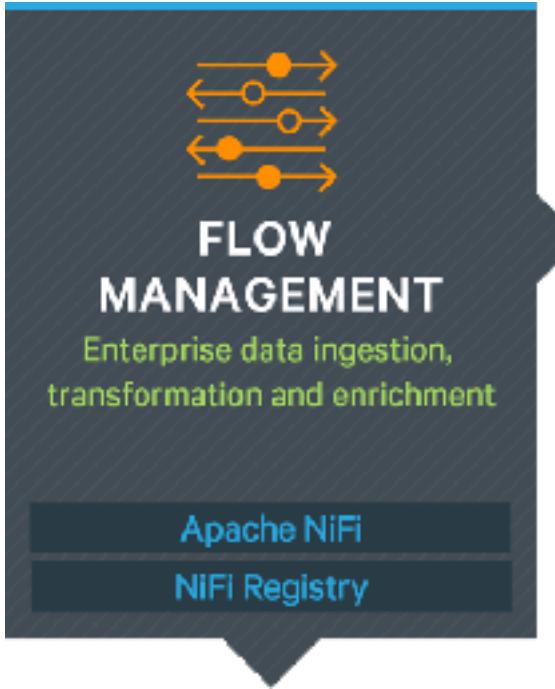
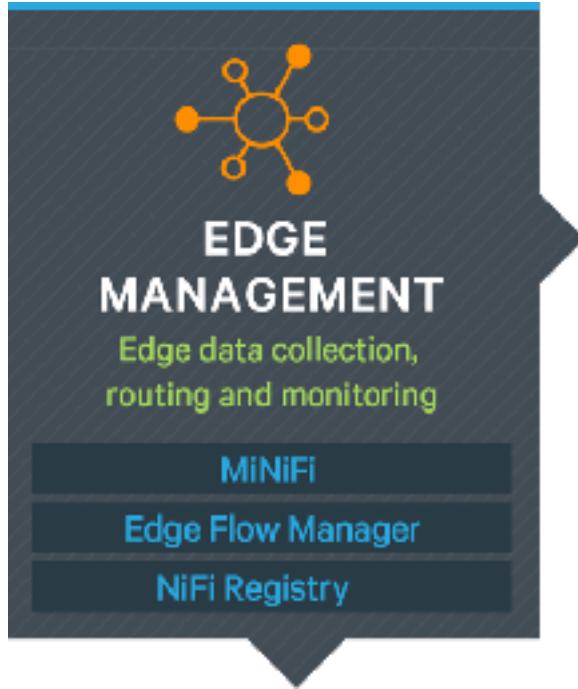
- Pattern matching
- Predictive and Prescriptive Analytics
- Complex Event Processing
- Continuous & Real-time Insights



# 3 Kafka Analytics Access Patterns



# CLOUDERA DATAFLOW



---

# ENTERPRISE SERVICES

- Provisioning
- Management
- Monitoring
- Unified Security
- Single Sign-on
- Audit
- Compliance
- Edge-to-Enterprise Governance



# KEY DIFFERENTIATORS

**100% open source technology** – Only vendor with this strategy; prevents vendor lock-in



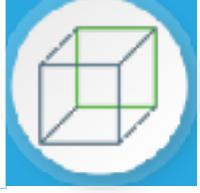
**300+ pre-built processors** – Only product to offer such comprehensive connectivity from edge to enterprise



**3 Streaming analytics engines** – Only vendor to offer a choice of three streaming analytics engines to customers for all their streaming architecture needs



**Built-in data provenance** – Only product in the market to offer out-of-the-box data provenance on data-in-motion



**Comprehensive streaming platform** – Only big data vendor to offer a comprehensive streaming platform from real-time data ingestion, transformation, routing to descriptive, prescriptive and predictive analytics.



# THANK YOU

cloudera