

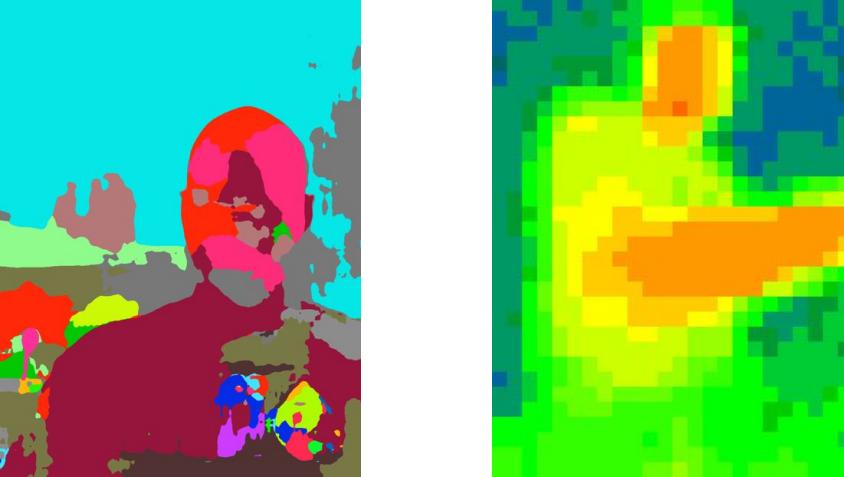


# Apache NiFi 101: Introduction and Best Practices

Timothy Spann  
Principal Developer Advocate

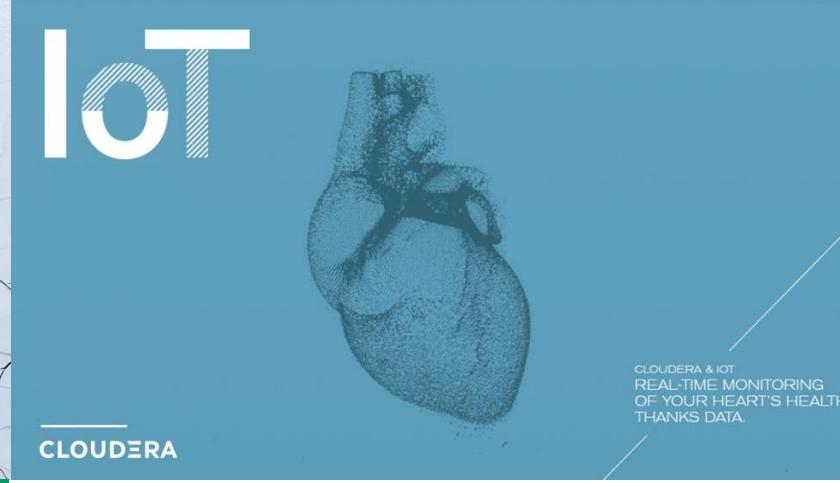
A large, stylized graphic element consisting of three diagonal stripes. The top stripe is light gray, the middle is dark teal, and the bottom is orange. The text "CLOUDERA" is partially visible at the bottom right corner of this graphic.

CLOUDERA



# ENTERPRISE DATA CLOUD

CLOUDERA



## CLOUDERA



# Tim Spann

Twitter: @PaasDev // Blog: [datainmotion.dev](http://datainmotion.dev)

Principal Developer Advocate.

Princeton Future of Data Meetup.

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

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

<https://github.com/tspannhw>



DZone REFCARDS TREND REPORTS EI  
Top IoT Experts  
Tim Spann  
Principal Developer Advocate, Cloudera  
<https://github.com/tspannhw/SpeakerProfile/>  
Tim Spann is a Principal Developer Advocate in Data In Motion for Cloudera. He works with Apache NiFi, Apache Pulsar, Apache...



# Future of Data - New York + Princeton + Virtual



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

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

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 ...



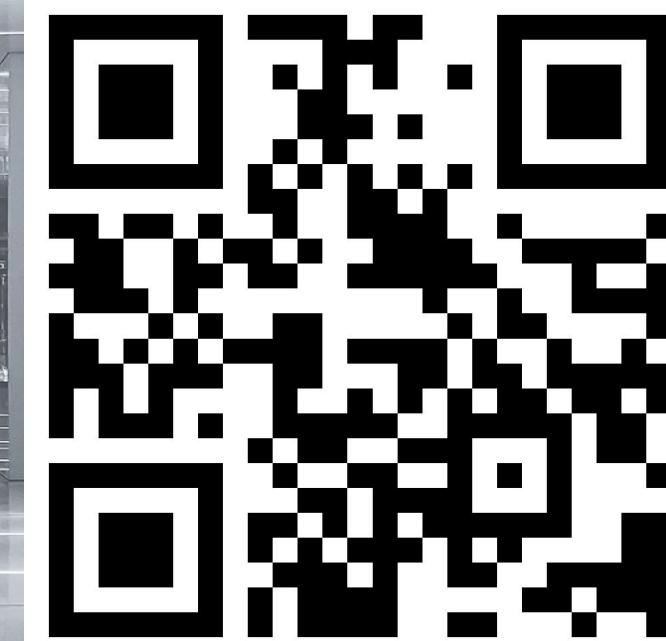
AN OPEN SOURCE COMMUNITY

@PaasDev

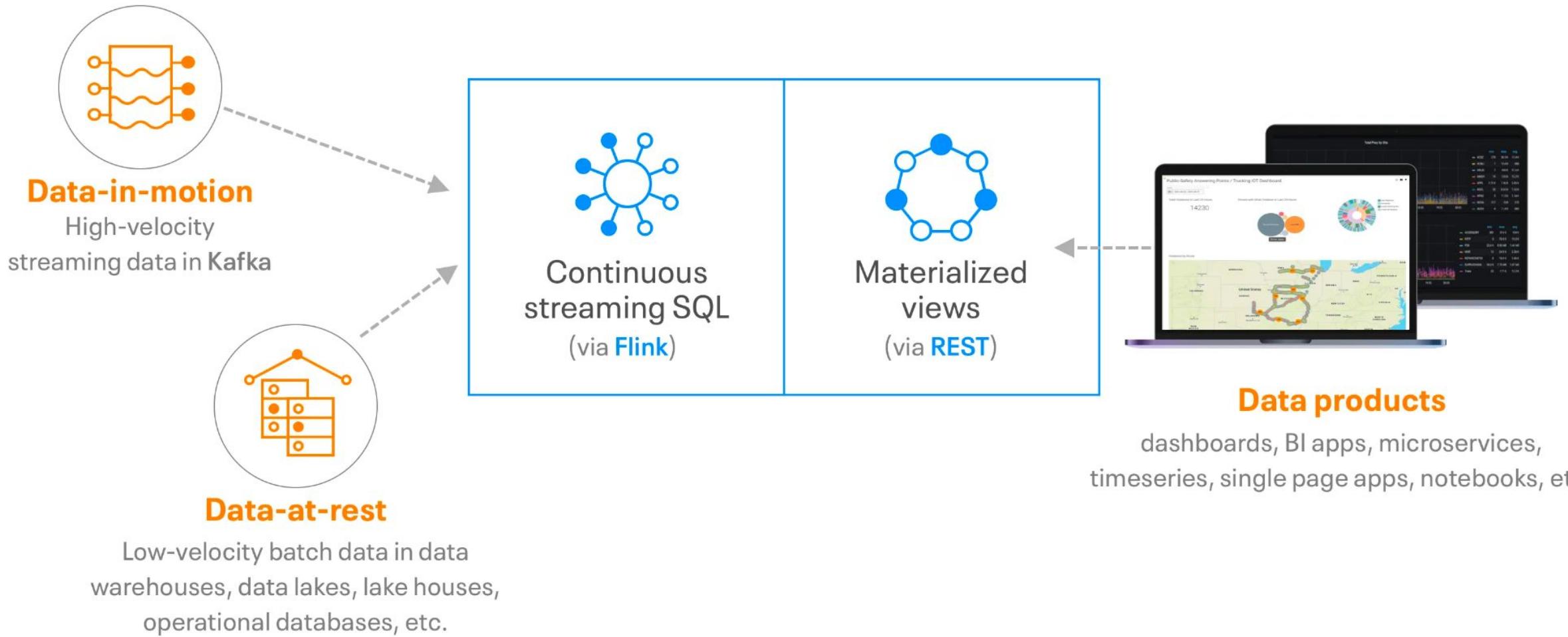
# FLaNK Stack Weekly



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



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



# Apache NiFi Resources

<https://www.datainmotion.dev/2020/10/top-25-use-cases-of-cloudera-flow.html>

<https://www.datainmotion.dev/2020/12/basic-understanding-of-cloudera-flow.html>

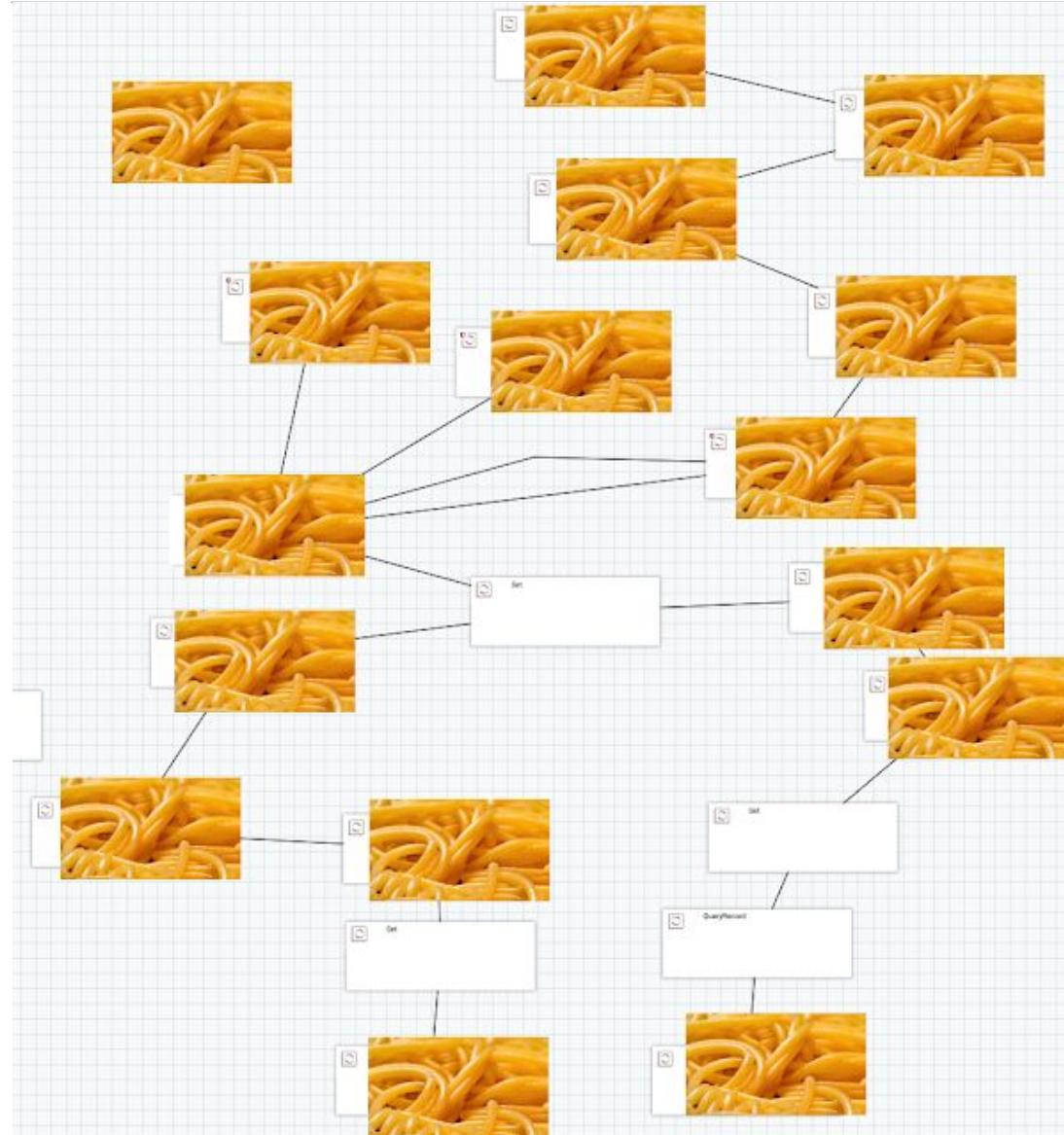
<https://www.datainmotion.dev/2020/06/no-more-spaghetti-flows.html>

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

# Basic Understanding of Cloudera Flow Management - Apache NiFi

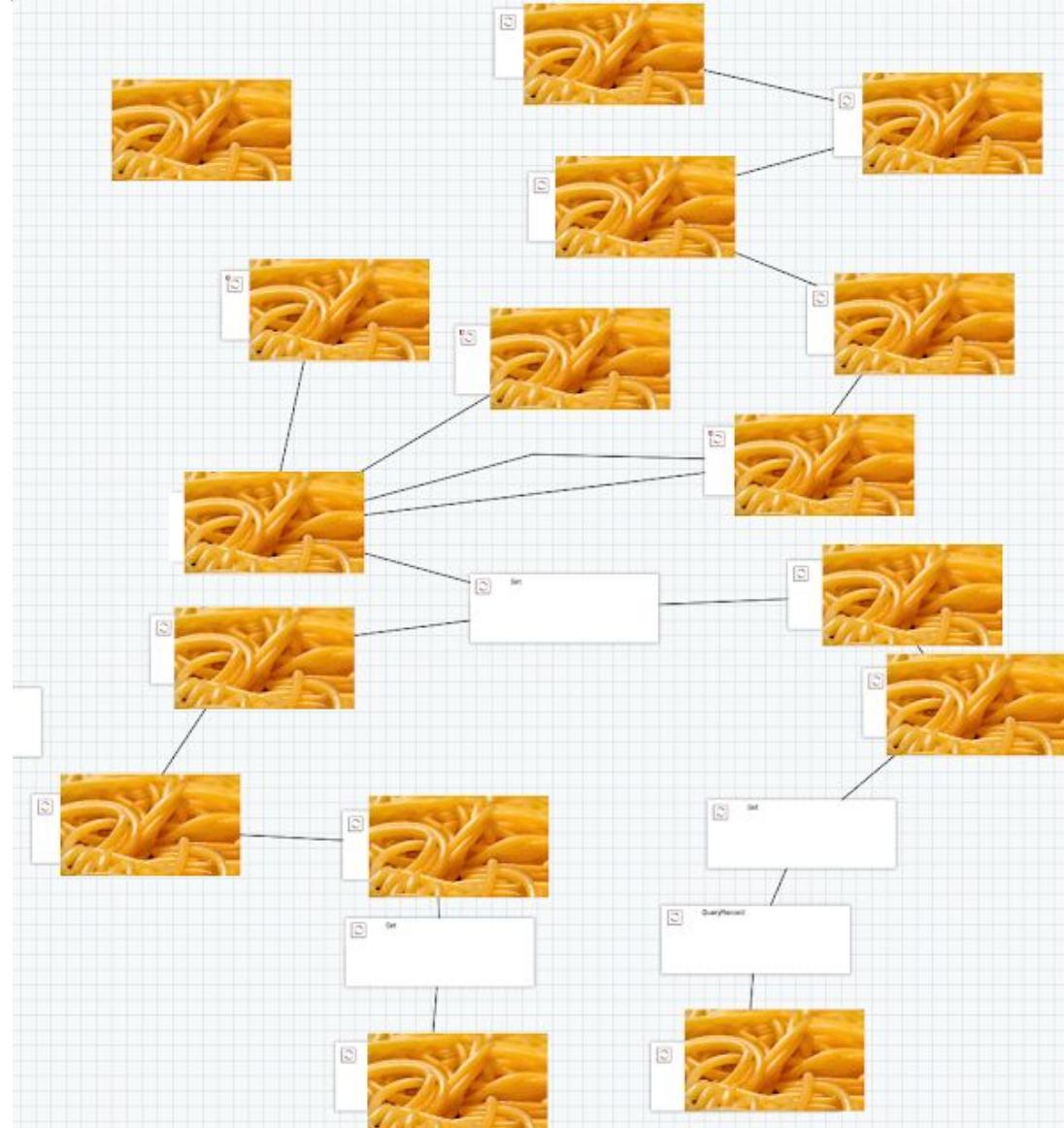
---

- NiFi Cluster Architecture
- Content Repository
- Provenance Repository
- FlowFile Repository
- FlowFile, Attributes, Process Groups, Connections, Flow Controllers
- Controller Services
- Common Attributes (uuid, filename, path, file size, ...)
- Expression Language
- Relationships
- Bulletins
- Input Port
- Output Port
- Empty Queues
- Setting Warning Levels
- Funnels
- RecordPath
- Using Record Processors (Readers/Writers)
- NiFi REST API
- Handling Errors
- Parameter Context / Parameters
- Summary / Cluster / Bulletins
- Reporting Tasks
- Back pressure
- Prioritized Queues
- Load Balancing Strategies
- Prioritization
- Using Search
- Using Documentation
- Site-to-Site Communication / Remote Process Groups
- Extensions
- Scheduling
- Tailing Files
- Reading sFTP/FTP Files
- Wait and Notify
- RetryFlowFile Pattern
- NiFi Calcite SQL
- Using Jolt
- Using JsonPath
- Making REST Calls
- Receiving REST Calls
- LookupRecord
- Working with Caches
- Restarting Flows
- Pass by Reference
- Using Regular Expressions
- Funnels



### Do Not:

- Do not Put 1,000 Flows on one workspace.
- If your flow has hundreds of steps, this is a Flow Smell. Investigate why.
- Do not Use ExecuteProcess, ExecuteScripts or a lot of Groovy scripts as a default, look for existing processors
- Do not Use Random Custom Processors you find that have no documentation or are unknown.
- Do not forget to upgrade, if you are running anything before Apache NiFi 1.10, upgrade now!
- Do not run on default 512M RAM.
- Do not run one node and think you have a highly available cluster.
- Do not split a file with millions of records to individual records in one shot without checking available space/memory and back pressure.
- Use Split processors only as an **absolute last resort**. Many processors are designed to work on FlowFiles that contain many records or many lines of text. Keeping the FlowFiles together instead of splitting them apart can often yield performance that is improved by 1-2 orders of magnitude.

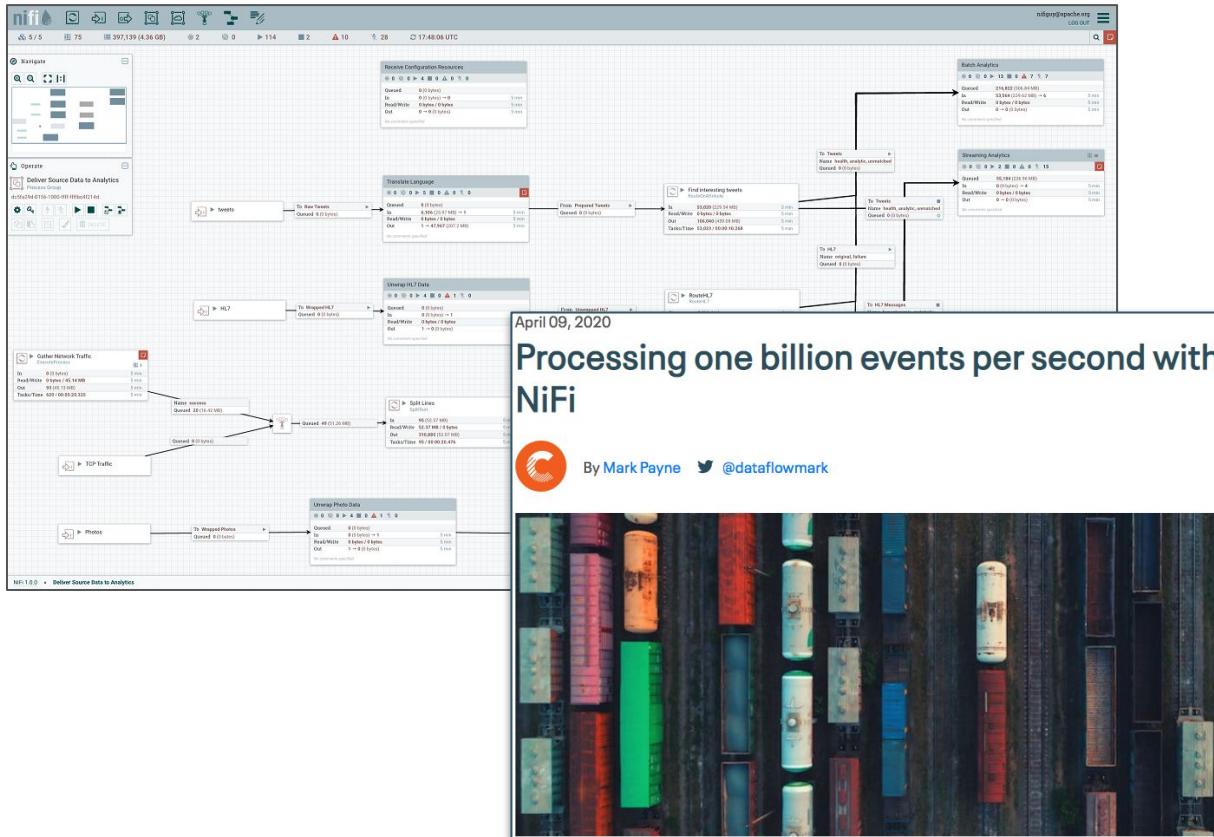


**Do:**

- Reduce, Reuse, Recycle. Use Parameters to reuse common modules.
- Put flows, reusable chunks (write to Slack, Database, Kafka) into separate Process Groups.
- Write custom processors if you need new or specialized features
- Use RecordProcessors everywhere
- Read the Docs!
- Use the NiFi Registry for version control.
- Use NiFi CLI and DevOps for Migrations.
- Run a CDP NiFi Datahub or CFM managed 3 or more node cluster.
- Walk through your flow and make sure you understand every step and it's easy to read and follow. Is every processor used? Are there dead ends?
- Do run Zookeeper on different nodes from Apache NiFi.
- For Cloud Hosted Apache NiFi - go with the "high cpu" instances, such as 8 cores, 7 GB ram.
- same flow 'templatized' and deployed many many times with different params in the same instance
- Use routing based on content and attributes to allow one flow to handle multiple nearly identical flows is better than deploying the same flow many times with tweaks to parameters in same cluster.
- Use the correct driver for your database. There's usually a couple different JDBC drivers.

# CLOUDERA FLOW MANAGEMENT - POWERED BY APACHE NiFi

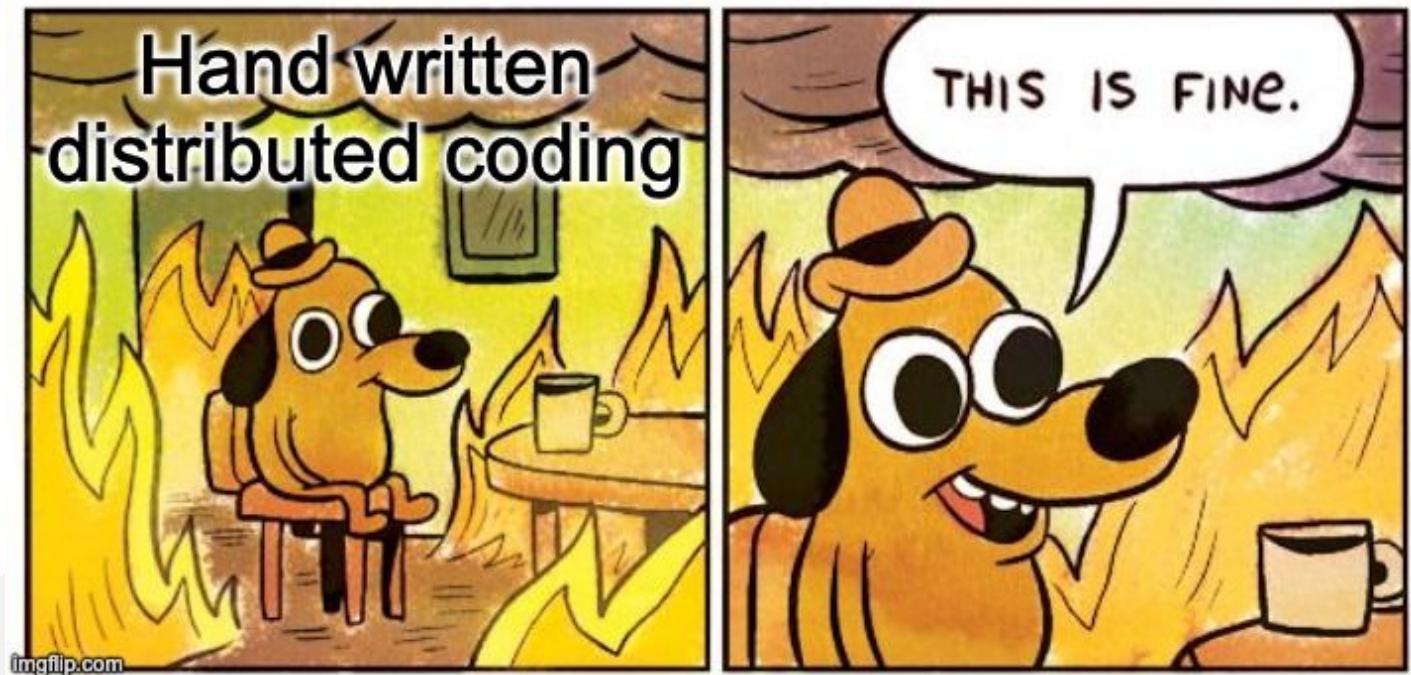
Ingest and manage data from edge-to-cloud using a no-code interface



- #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

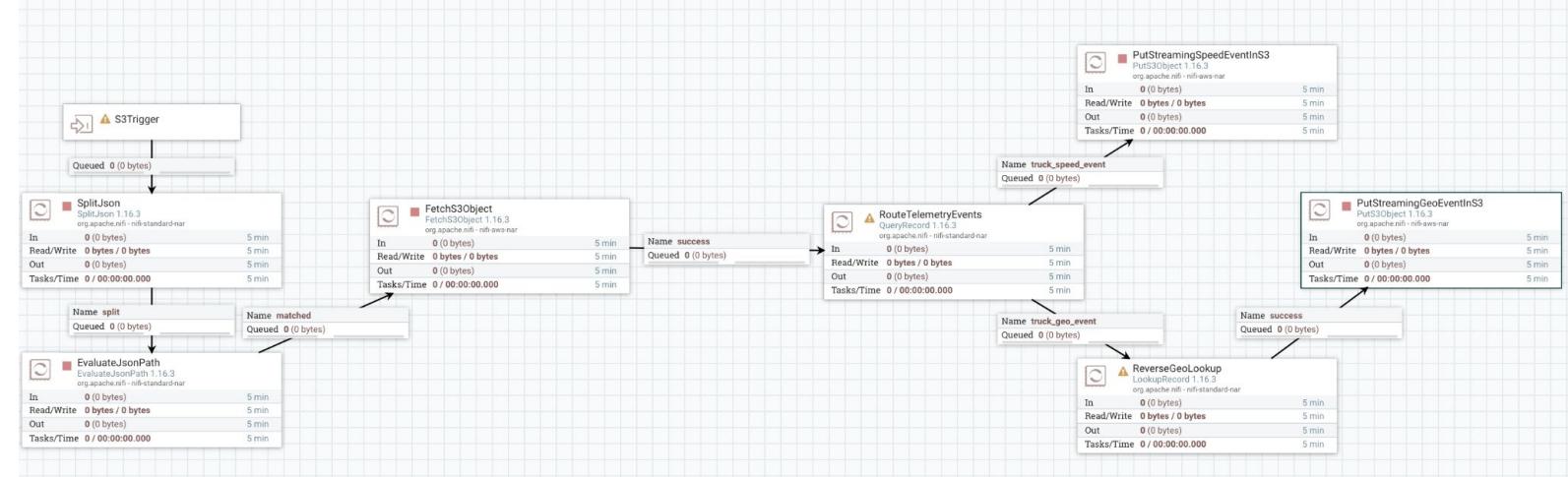
# CLOUD

---



# 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

# 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

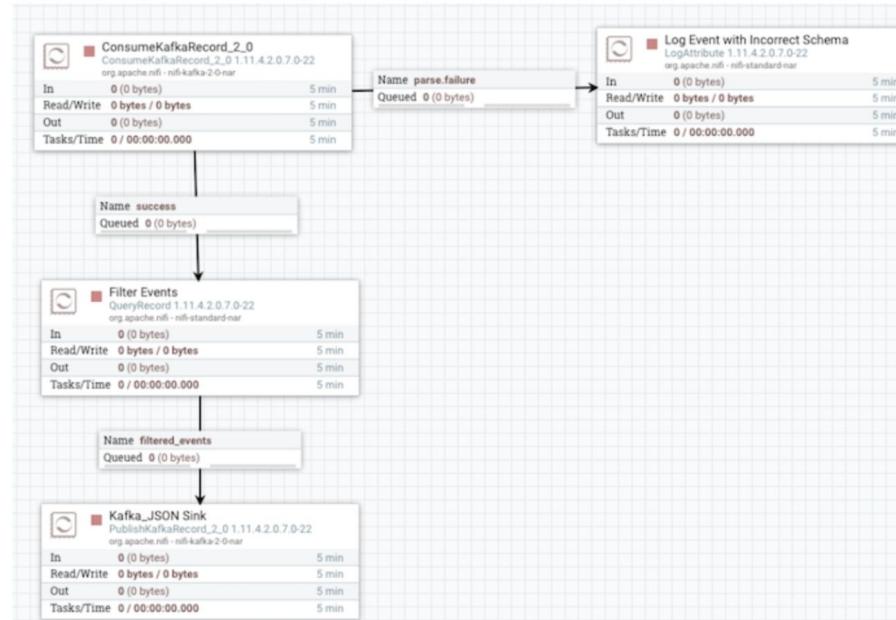
Managing a DataFlow Environment

## 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.



```
graph TD; A[ConsumeKafkaRecord_2_0] --> B{Name parse.failure  
Queued 0 (0 bytes)}; B --> C[Name success  
Queued 0 (0 bytes)]; C --> D[Filter Events]; D --> E{Name filtered_events  
Queued 0 (0 bytes)}; E --> F[Kafka_JSON Sink]
```

Detailed description of the NiFi flow:

- Source:** ConsumeKafkaRecord\_2\_0 (Consuming from Kafka topic)
- Decision:** Name parse.failure (Queued 0 (0 bytes))
- Decision:** Name success (Queued 0 (0 bytes))
- Processor:** Filter Events (QueryRecord)
- Decision:** Name filtered\_events (Queued 0 (0 bytes))
- Sink:** Kafka\_JSON Sink (Publishing to Kafka topic)

# 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

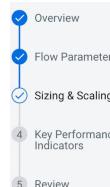
Enter parameter values.

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

The screenshot shows the 'KPIs' tab of the 'Covid Ingest' deployment details. At the top, it says 'aws dataflow-demo' and has a 'Manage Deployment' link. Below are tabs for 'KPIs', 'System Metrics', and 'Alerts'. The 'KPIs' tab is active, showing 'Deployment Information' for 'CovidIDBroker V.2' deployed by Michael Kohs. It shows 'NODE COUNT 1', 'AUTO SCALING Up to 3 nodes', 'CREATED ON 2021-03-29 14:33 PDT', and 'LAST UPDATED 2021-03-29 14:36 PDT'. The 'Data In' section displays metrics for 'Entire Flow':

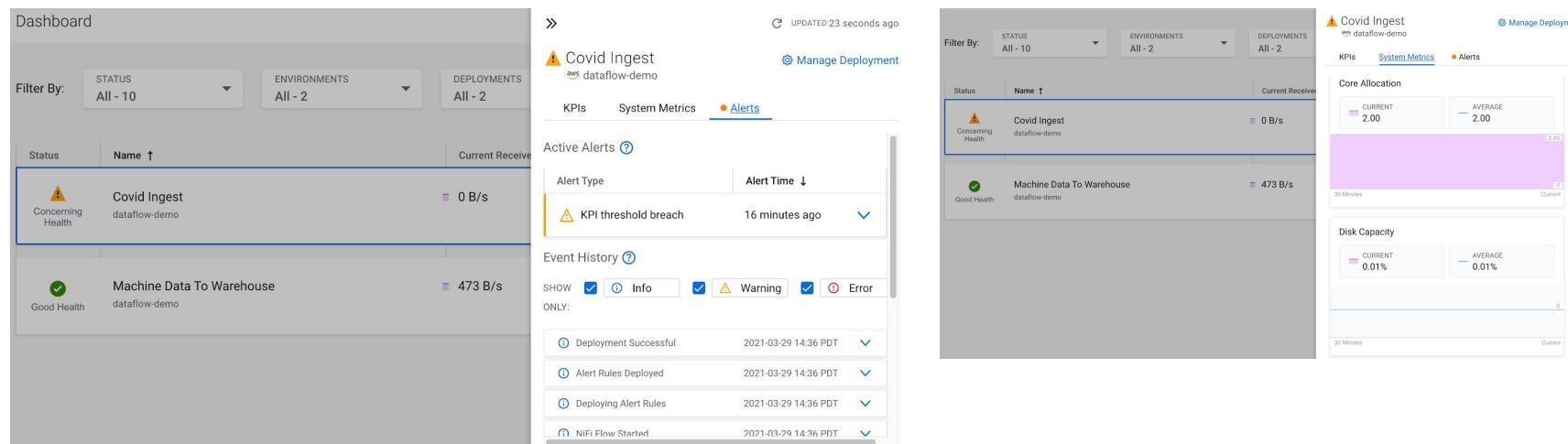
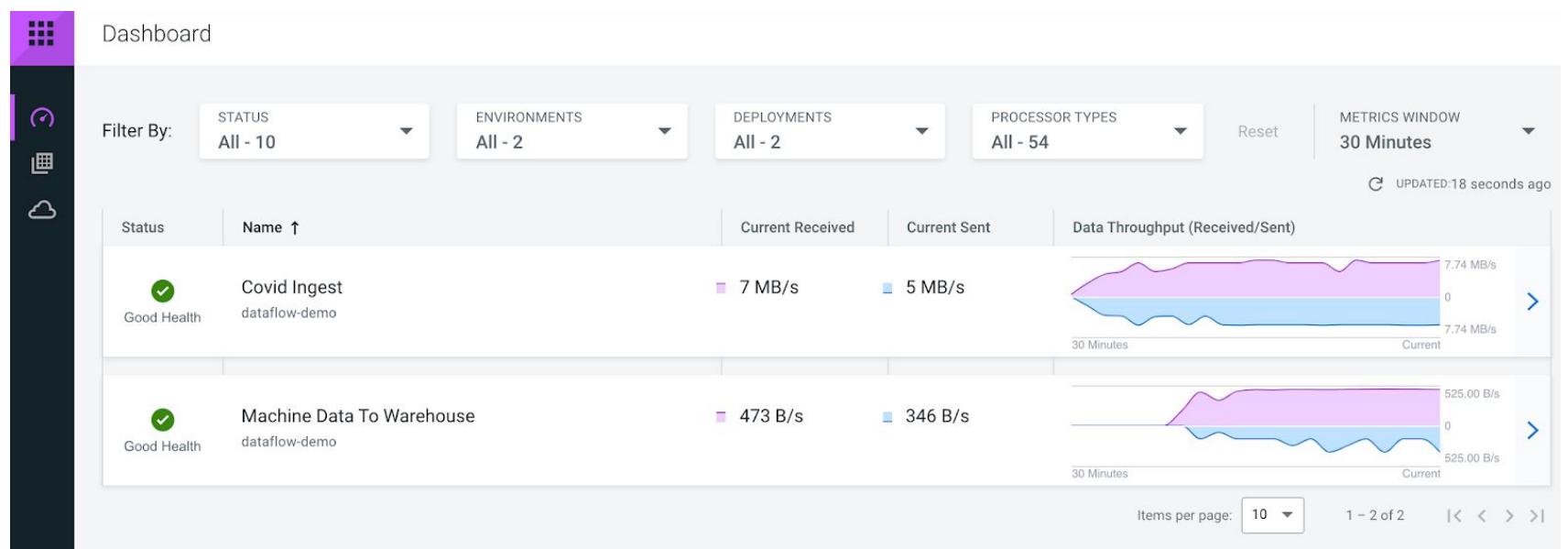
CURRENT 2.68 MB/s	AVERAGE 2.43 MB/s	BOUNDARY < 150.00 KB/s
-------------------	-------------------	------------------------

Below this is a chart showing data in over 30 minutes. The 'Bytes Sent' section displays metrics for 'Write to S3 using HDFS proc':

CURRENT 0.00 B	AVERAGE 0.00 B
----------------	----------------

# 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)

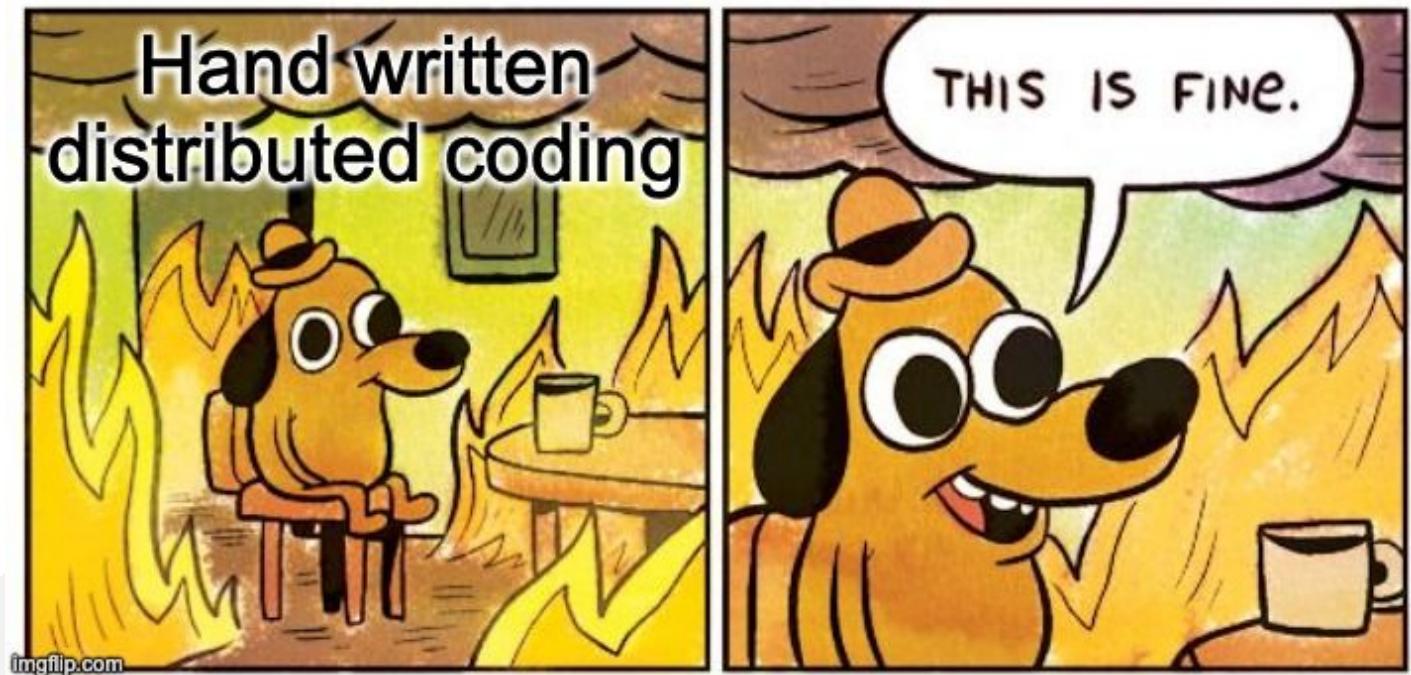
TASKS 29/00:00:00.123

5 min

The screenshot shows the Cloudera Data Flow interface. On the left is a dark sidebar with navigation links: Dashboard, Catalog, ReadyFlow Gallery, Flow Design (which is selected and highlighted in purple), Functions, Environments, Get Started, Help, and Stephen Hawking (version v2.0.0). The main area is titled 'Flow Design / [RootProcessGroupName]' and shows a visual canvas with a single processor node. A tooltip for the processor displays its details: [ProcessorName], [ProcessorType], [Version#]. The configuration pane on the right lists various parameters with their current values and edit controls. At the bottom are 'Apply Changes' and 'Discard Changes' buttons.

---

NEW





0 51,434 / 61.69 MB 0 0 193 914 555 166 0 0 0 0 0 21:02:42 EDT



# Records

New ExcelRecord Reader

AmazonGlueSchemaRegistry

<https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12353320>

0 51,434 / 61.69 MB 0 0 193 914 555 166 0 0 0 0 0 21:02:42 EDT



# New to 2023 Processors

GenerateRecord

GetAsanaObject

PutSalesforceObject

QuerySalesforceObject

PutIoTDBRecord

QueryIoTDBRecord

ListGoogleDrive

FetchGoogleDrive

PutGoogleDrive

PutBoxFile

ListBoxFile

FetchBoxFile

PutDropbox

DecryptContent

DecryptContentCompatibility

<https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12353320>

0 51,434 / 61.69 MB 0 0 193 914 555 166 0 0 0 0 0 21:02:42 EDT



# New to 2023 Processors

ExtractRecordSchema

RemoveRecordField

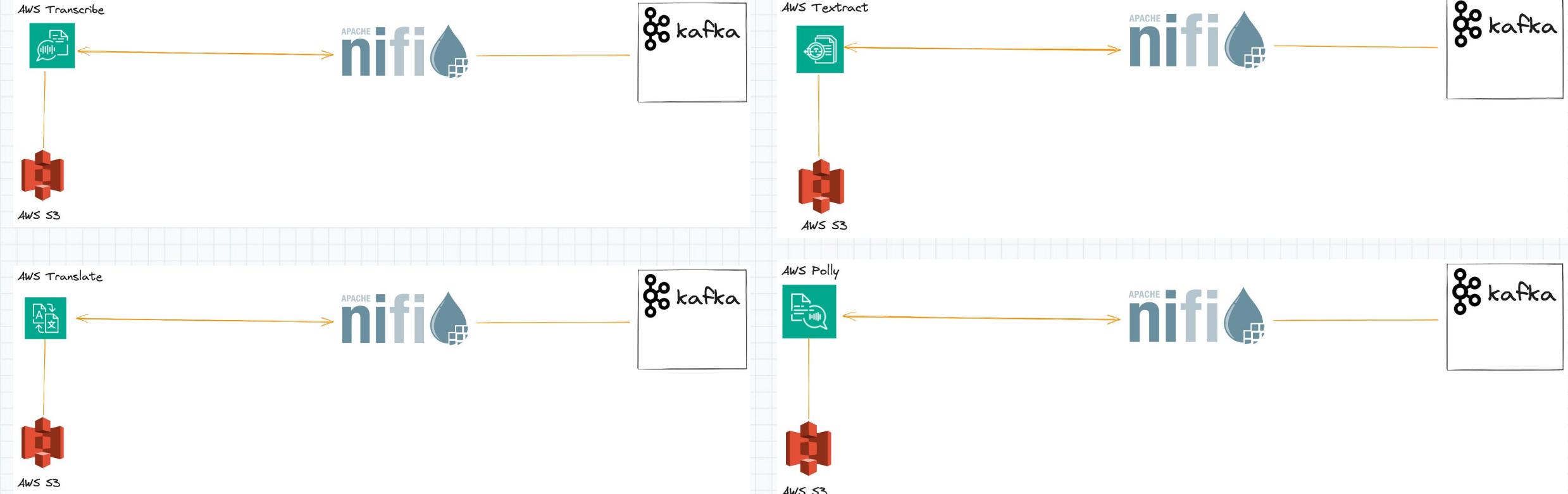
VerifyContentMAC

TriggerHiveMetaStoreEvent

“count” function added to RecordPath

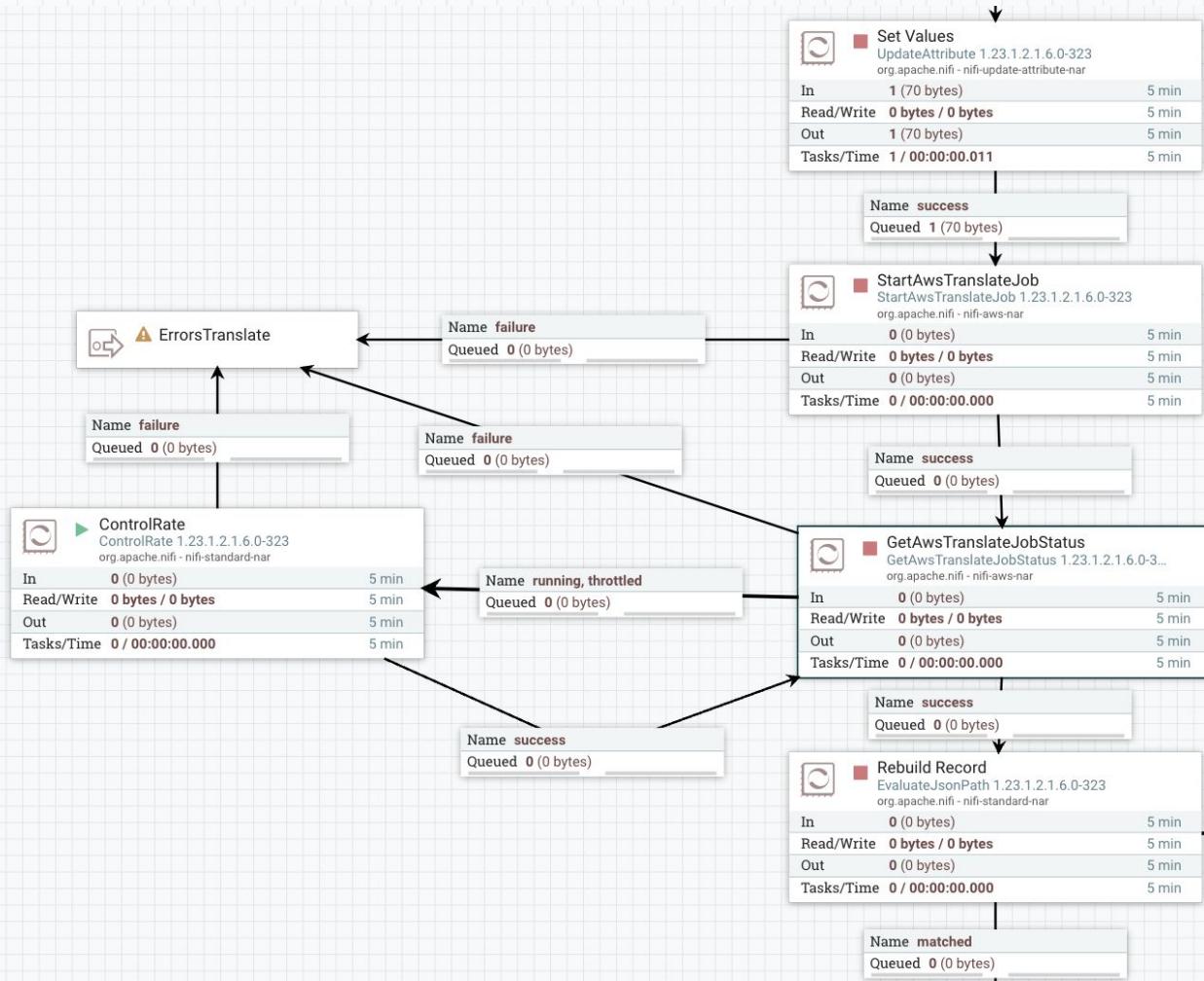


# AWS ML Service Processors



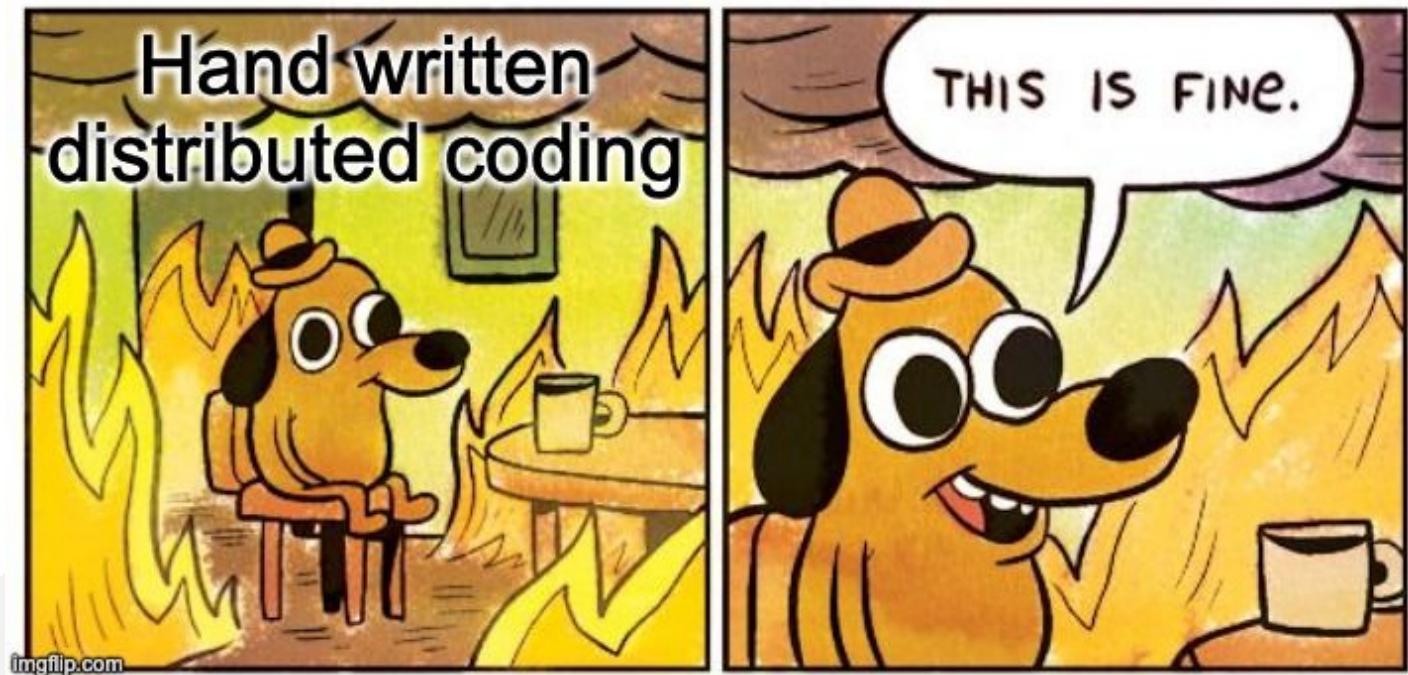
<https://github.com/tspannhw/FLaNK-AWSML>

# AWS Translate



---

2.0!



# Thanks to Pierre!



## Pierre Villard

Apache NiFi Committer & PMC member |  
Working @Cloudera - ex-@Google | Twitter &  
Github — @pvillard31 | Blog @  
[www.pierrevillard.com](http://www.pierrevillard.com)





# NiFi 2.0 Coming

- Python Integration
- Parameters
- JDK 17, maybe JDK 21+
- JSON Flow Serialization
- Rules Engine for Development Assistance
- Run Process Group as Stateless
- flow.json.gz

<https://cwiki.apache.org/confluence/display/NIFI/NiFi+2.0+Release+Goals>

<https://medium.com/cloudera-inc/getting-ready-for-apache-nifi-2-0-5a5e6a67f450>



# Deprecating for Removal

Deprecate Lua and Ruby Script Engines

Deprecate ECMAScript Script Engine

Deprecate the Ambari Reporting Task

Deprecate Kafka 1.x components and 2.0 components

XML Templates

Variables

See:

<https://cwiki.apache.org/confluence/display/NIFI/Deprecated+Components+and+Features>



0 51,434 / 61.69 MB 0 0 193 914 555 166 0 0 0 0 0 21:02:42 EDT



# Start Using

ExecuteStateless -> run your stateless flows right in a regular NiFi cluster

Parameters

JSON Flow Serialization

Records everywhere

# Python as First Class (NIFI-11241)

## Graphical UI with custom Python based extensions

```
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

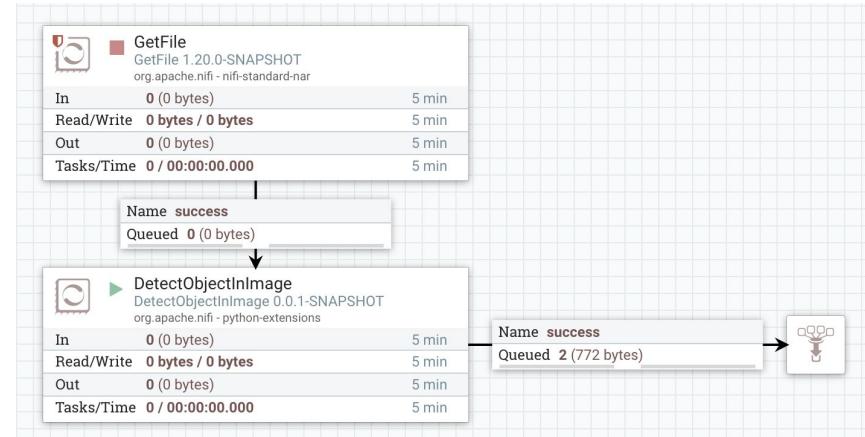
class DetectObjectInImage:
    class Java:
        implements = ['org.apache.nifi.python.processor.FlowFileTransform']
    class ProcessorDetails:
        version = '0.0.1-SNAPSHOT'
        dependencies = ['numpy >= 1.23.5', 'opencv-python >= 4.6']

    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), ' +
                      'DLDT (*.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:
```



## Apache NiFi in a few numbers

A very active project with a dynamic community & comparison with ACEU 2019

**2800+ members on the Slack channel (535+ - 4 years ago)**

**475+ contributors on Github across the repositories  
(260+ - 4 years ago)**

**65 committers in the Apache NiFi community (45 - 4 years ago)**

**Apache NiFi 1.23.2 is the latest release, NiFi 2.0 coming soon (NiFi 1.10 - 4 years ago)**

**14M+ docker pulls of the Apache NiFi image (1M+ - 4 years ago)**

**MiNiFi C++**  
*(small footprint)*

**MiNiFi Java**  
*(headless version of NiFi)*

**NiFi Registry**

**Stateless NiFi**

**Cloudera Edge Flow Manager**  
*(Command & Control of MiNiFi Agents)*

**Cloudera NiFi for Kafka Connect**

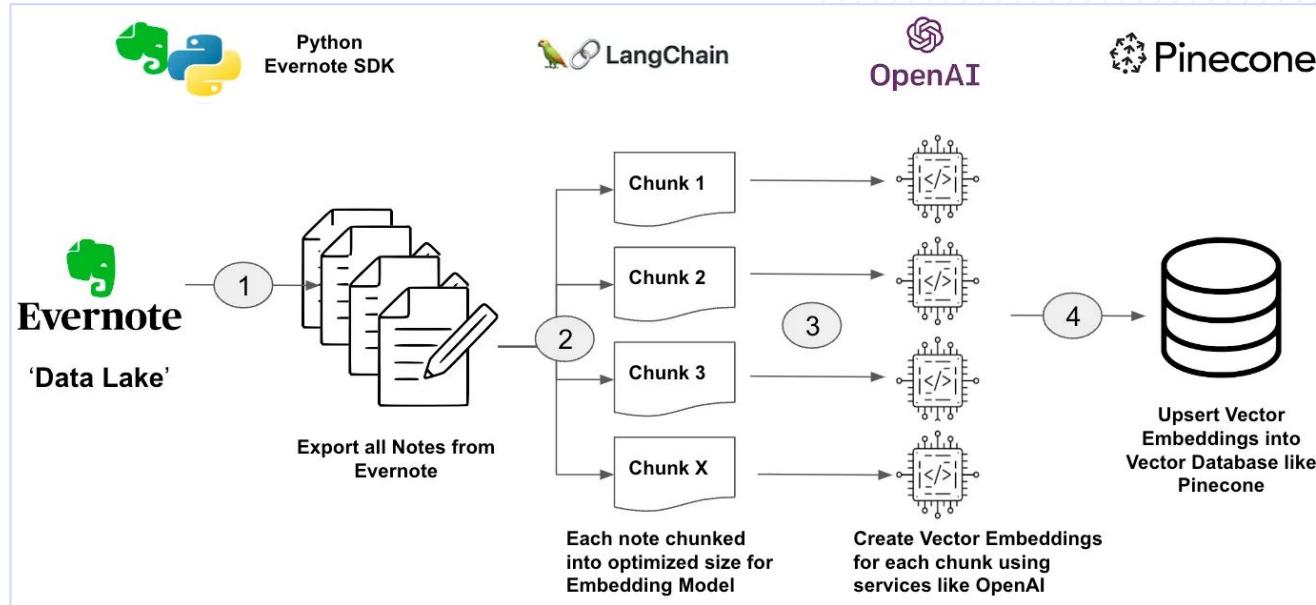
**NiFi in Cloudera DataFlow Functions**

**Cloudera DataFlow**

## NiFi Deploy Options from Open Source to Managed

# NiFi 2.0 is coming... <https://medium.com/cloudera-inc/getting-ready-for-apache-nifi-2-0-5a5e6a67f450>

- First-class citizen Python API
- Rules Engine
- NiFi Stateless at Process Group level
- Java 21 (virtual threads, perf improvements, etc)



Closing the gap between data engineers and data scientists...

- Export documentation (Sharepoint, OCR) to build the knowledge base powering your chatbot
- Scrape the internet (Sitemap) to build the knowledge base powering your chatbot
- Real-time streaming ingest of Slack to build the knowledge base powering your chatbot

## Evernote AI Chatbot Powered by Apache NiFi using OpenAI, Pinecone & Langchain

When did my right elbow pain occur?

The right elbow pain occurred on January 11, 2023.

Evernote Source: Note: Right Elbow Pain - 01-11-23 (Notebook: Family\_Aju\_Health),

What doctor did I see for it?

You saw Dr. Patel for your right elbow pain.

Evernote Source: Note: Appointment with Dr. Patel on 6\_20 for Right Elbow Pain (Notebook: Family\_Aju\_Health),

What was the cause for the right elbow pain and what did the doctor suggest?

The cause for the right elbow pain is tennis elbow. The doctor suggested a cortisone shot in the right elbow and taking Diclofenac for prescription pain relief.

Evernote Source: Note: Appointment with Dr. Patel on 6\_20 for Right Elbow Pain (Notebook: Family\_Aju\_Health), Note: Right Elbow Pain - 01-11-23 (Notebook: Family\_Aju\_Health),

What was the dosage for Diclofenac?

The dosage for Diclofenac is 50 mg, 3 times a week.

Evernote Source: Note: Medications & Prescriptions That I Take (Notebook: Family\_Aju\_Health),

You:

<https://medium.com/@george.vetticaden/accelerating-ai-data-pipelines-building-an-evernote-chatbot-with-apache-nifi-2-0-and-generative-ai-9d977466ff4c>

---

# WALKTHRU





**EVOLVE**  
**NYC**

NOV 2, 2023  
THE GLASSHOUSE  
660 12TH AVE, NEW YORK, NY 10019

presented by

CLOUDERA

IBM

intel.

aws

TH<sup>AIR</sup>N YOU  

