Incrementally Streaming RDBMS Data To Your Data Lake Automagically

APACHECON @HOME
Spt, 29th – Oct. 1st

2020

John Kuchmek

Senior Solutions Engineer

APACHECON NA
Spt, 28th – Oct. 2nd
2020


Tim Spann
Principal Field Engineer
APACHECON NA
Spt, 28th – Oct. 2nd
2020

Incrementally Streaming RDBMS Data

- Loading Oracle Data
- Loading to Apache HDFS
- Upserting to Apache Kudu
- Merging to Apache Hive

APACHECON
@Home
Sep 29 - Oct 1, 2020

Future of Data - Princeton

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

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

@PaasDev

Speakers

John Kuchmek

Senior Solutions Engineer

Speakers

Tim Spann

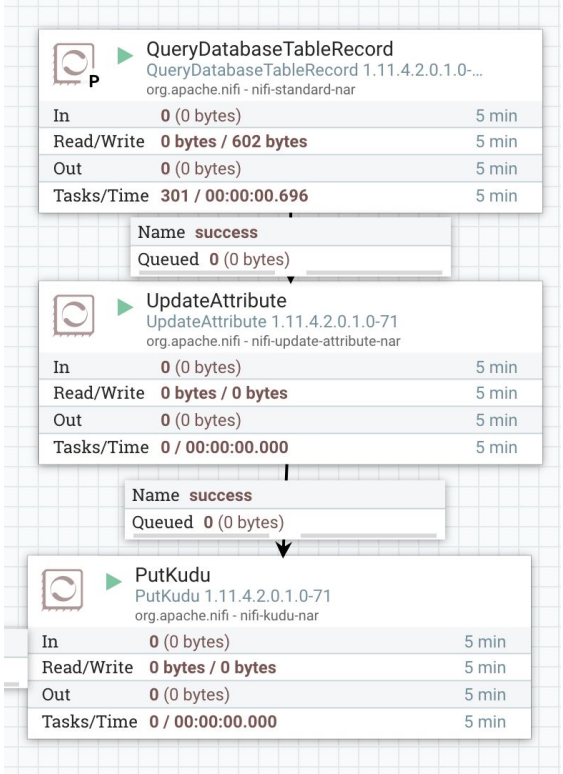
Principal DataFlow Field Engineer

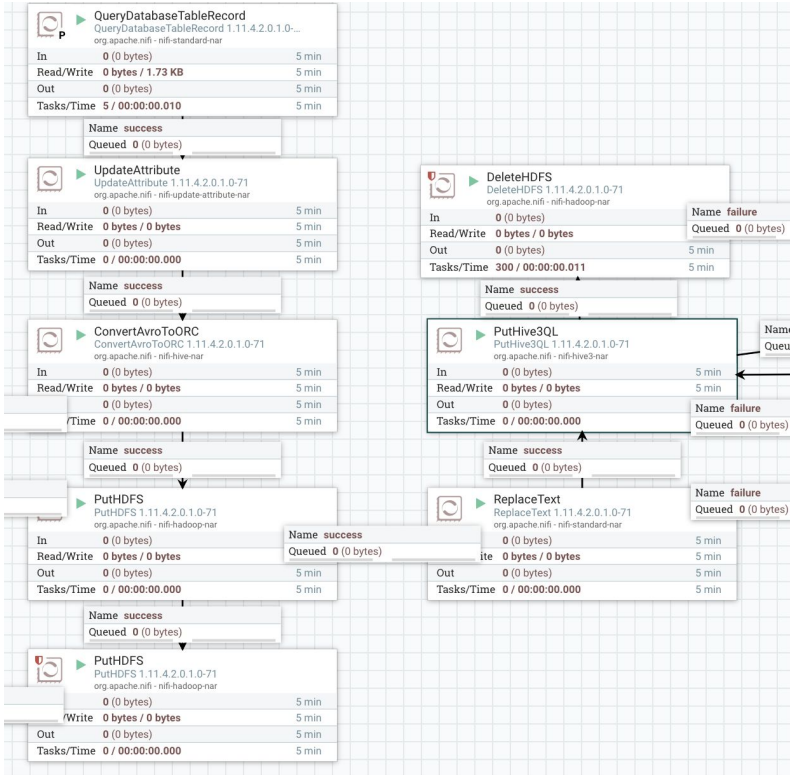
DZone Zone Leader and Big Data MVB
Princeton NJ Future of Data Meetup

https://github.com/tspannhw
https://www.datainmotion.dev/

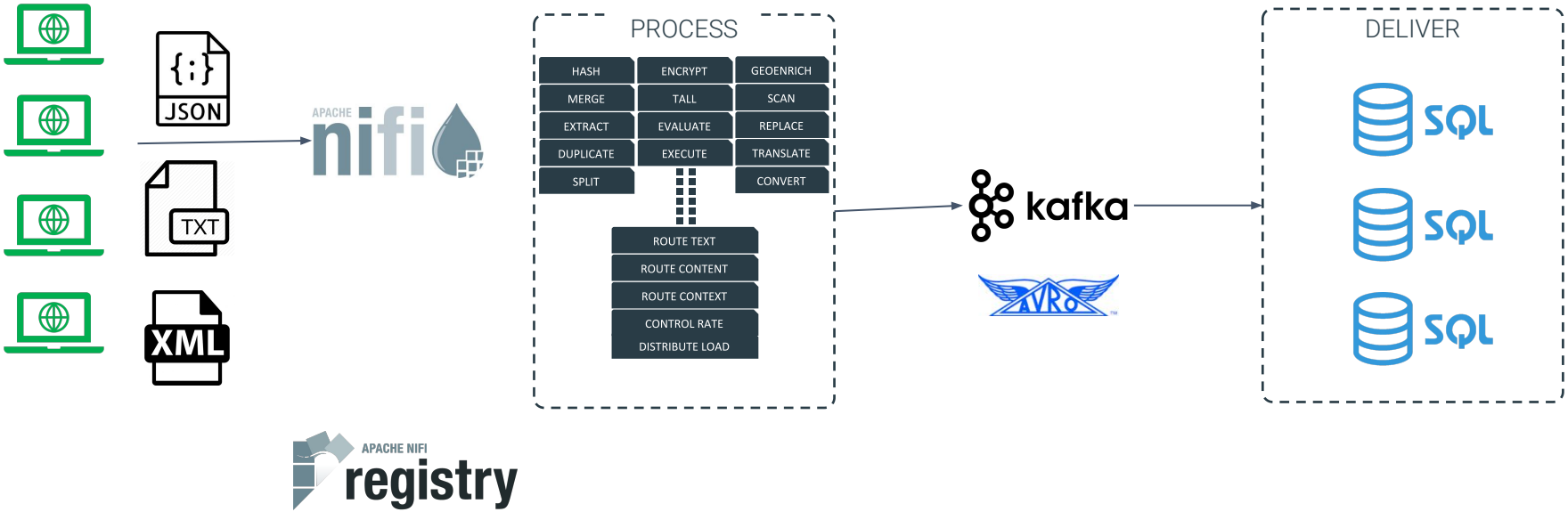
@PaasDev

JDBC Database to Apache Kudu / JDBC Database to HDFS and Hive

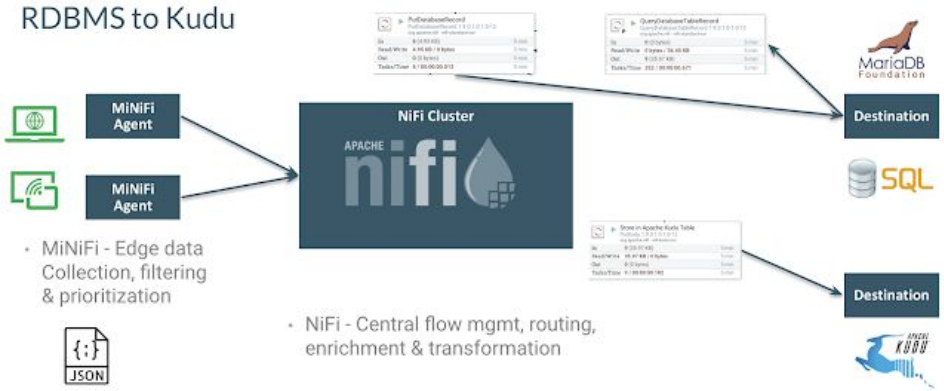




Trillions of Messages to SQL Databases and Data Warehouses



https://www.datainmotion.dev/2019/10/migrating-apache-flume-flows-to-apache_15.html

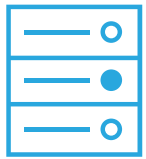


Property	Value
Database Connection Pooling Service	**MariaDB Connection Pool**
Database Type	**MySQL**
Table Name	**iot**
Columns to Return	No value set
Additional WHERE clause	No value set
Custom Query	No value set
Record Writer	**INFER JsonRecordSetWriter**
Maximum-value Columns	systemtime
Max Wait Time	**30 seconds**
Fetch Size	**0**
Max Rows Per Flow File	**0**
Output Batch Size	**0**
Maximum Number of Fragments	**0**
Normalize Table/Column Names	**true**

Schema Write Strategy	**Do Not Write Schema**
Schema Cache	No value set
Schema Access Strategy	**Inherit Record Schema**
Schema Registry	No value set
Schema Name	${schema.name}
Schema Version	No value set
Schema Branch	No value set
Schema Text	${avro.schema}
Date Format	No value set
Time Format	No value set
Timestamp Format	No value set
Pretty Print JSON	**false**
Suppress Null Values	**Never Suppress**
Output Grouping	**Array**

Reference Architecture

Files to RDBMS



Source Files

Read Files

Cloudera DataFlow Pipeline

PutDatabaseRecord

Messages to Databases

Controller Service Details

SETTINGS | PROPERTIES | COMMENTS

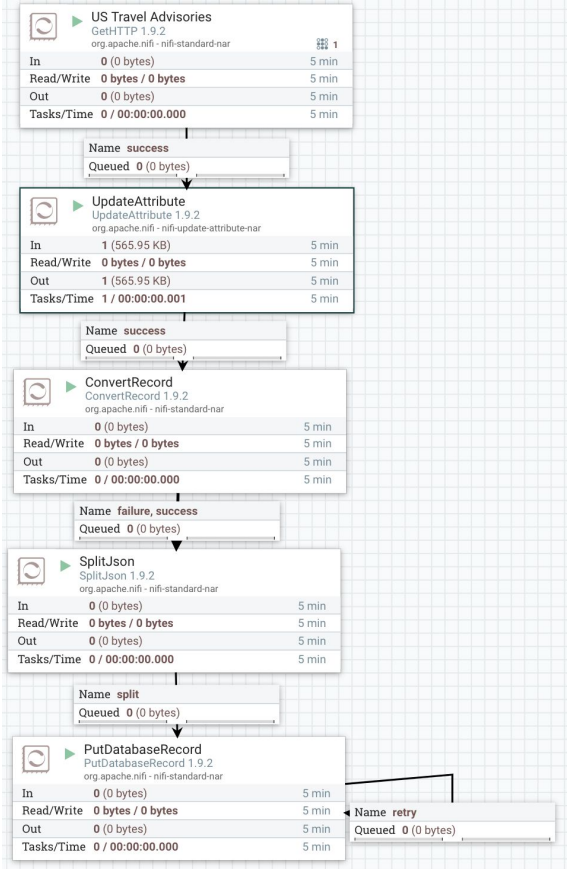
Required field

Property	Value
Database Connection URL	**jdbc:mysql://**
Database Driver Class Name	**com.mysql.j**
Database Driver Location(s)	/Users/tspar
Kerberos Credentials Service	No value set
Database User	usdba
Password	Sensitive val
Max Wait Time	**500 millis**
Max Total Connections	**3**
Validation query	No value set
Minimum Idle Connections	0
Max Idle Connections	8
Max Connection Lifetime	-1
Time Between Eviction Runs	-1
Minimum Evictable Idle Time	30 mins

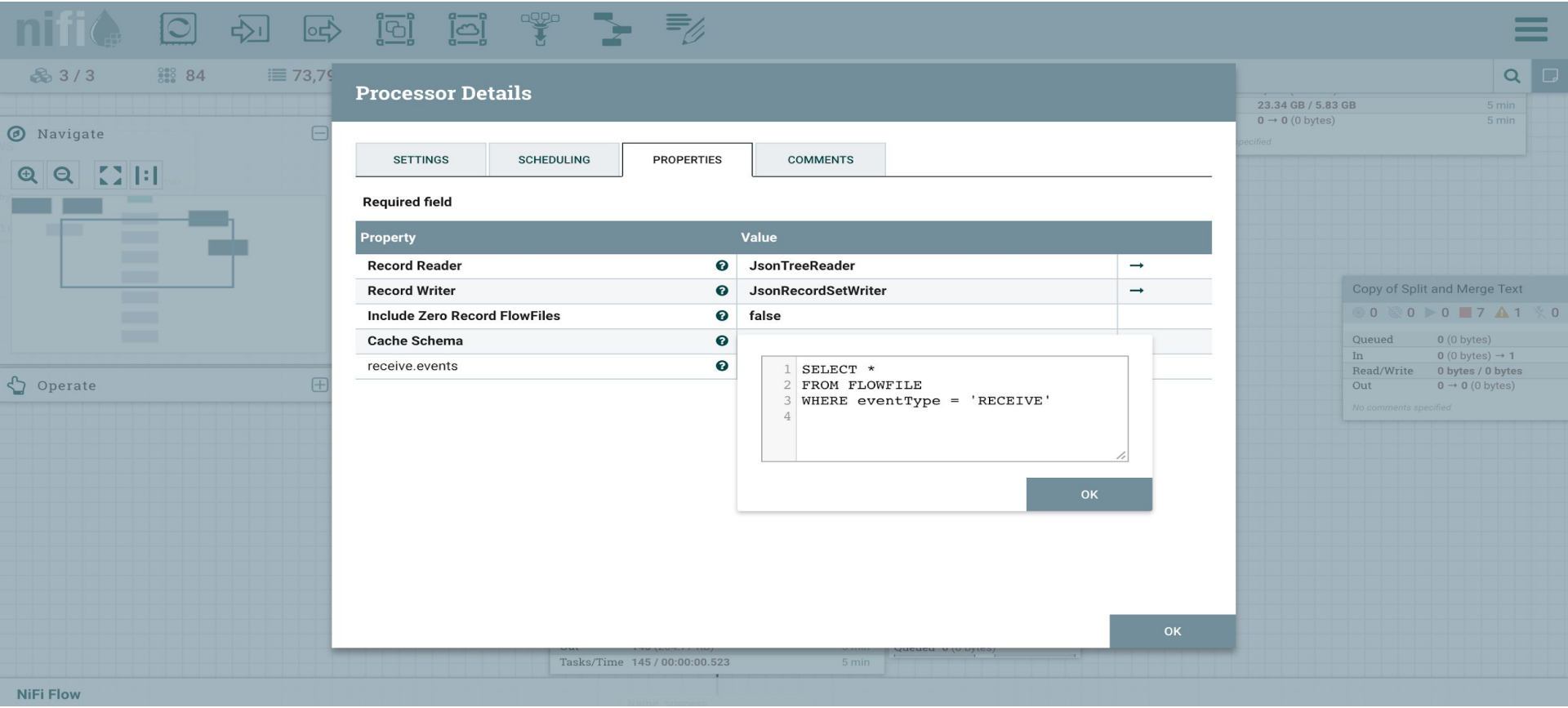
SETTINGS | SCHEDULING | PROPERTIES | COMMENTS

Required field

Property	Value
Record Reader	**JsonTreeReader**
Statement Type	**INSERT**
Database Connection Pooling Service	**DBCPConnectionPool**
Catalog Name	No value set
Schema Name	usdata
Table Name	**travelwarning**
Translate Field Names	true
Unmatched Field Behavior	Ignore Unmatched Fields
Unmatched Column Behavior	Ignore Unmatched Columns
Update Keys	No value set
Field Containing SQL	No value set
Quote Column Identifiers	false
Quote Table Identifiers	false
Max Wait Time	**0 seconds**



‘QueryRecord’ Processor



https://medium.com/@abdelkrim.hadjidj/democratizing-nifi-record-processors-with-automatic-schemas-inference-4f2b2794c427

ADVANCED XML PROCESSING

Property	Value
Schema Access Strategy	**Infer Schema**
Schema Registry	AvroSchemaRegistry
Schema Name	${schema.name}
Schema Version	No value set
Schema Branch	No value set
Schema Text	${avro.schema}
Schema Inference Cache	No value set
Expect Records as Array	**false**
Attribute Prefix	No value set
Field Name for Content	No value set
Date Format	No value set
Time Format	No value set
Timestamp Format	No value set

Property	Value
Record Reader	**XMLReader**
Record Writer	**JsonRecordSetWriter**
Include Zero Record FlowFiles	**false**
Cache Schema	**true**
query1	SELECT * FROM FLOWFILE

https://pierrevillard.com/2018/06/28/nifi-1-7-xml-reader-writer-and-forkrecord-processor/

https://www.datainmotion.dev/2019/03/advanced-xml-processing-with-apache.html

Pull From External Sources

- *Example*
  - Flat files on an FTP server named by date
    - Downloads file
  - HTTP REST API endpoint
    - Invokes API and downloads data
  - Legacy/Remote DB
    - Performs SQL queries

DBCP Connection Pool to remote SQL Server

Property	Value
Database Connection URL	**jdbc:sqlserver://example.database.windows.net:1433;data...**
Database Driver Class Name	**com.microsoft.sqlserver.jdbc.SQLServerDriver**
Database Driver Location(s)	/Users/alopresto/Workspace/scratch/
Kerberos Credentials Service	No value set
Database User	No value set
Password	No value set
Max Wait Time	**500 millis**
Max Total Connections	**8**
Validation query	No value set
Minimum Idle Connections	0
Max Idle Connections	8

ExecuteSQLRecord processor

SQL Pre-Query	No value set
SQL select query	SELECT * FROM dbo.patient;
SQL Post-Query	No value set
Max Wait Time	**0 seconds**
Record Writer	**AvroRecordSetWriter** →
Normalize Table/Column Names	**false**
Use Avro Logical Types	**true**
Max Rows Per Flow File	**0**
Output Batch Size	**0**

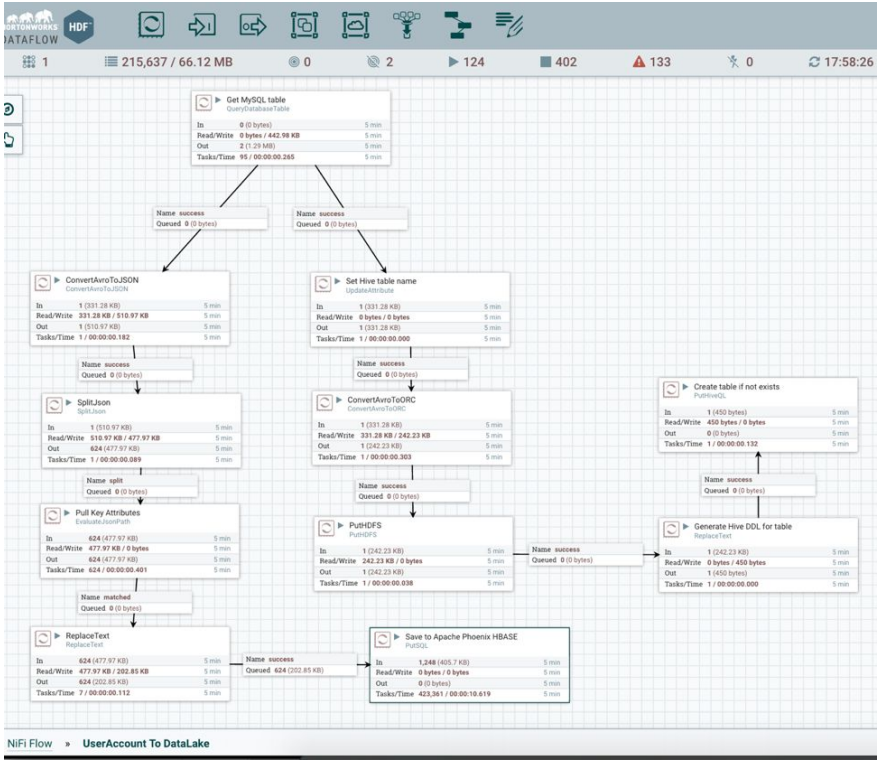
REST, Kafka, AVRO

Property	Value
Record Reader	**AvroReader**
Record Writer	**HDFS AvroRecordSetWriter**
Include Zero Record FlowFil...	**false**
Cache Schema	**true**
lowPriceAlert	SELECT * FROM FLOWFILE WHERE latestPrice <= week52Low
regularPrice	SELECT * FROM FLOWFILE WHERE latestPrice > week52Low

- We can convert JSON / AVRO /CSV / XML to AVRO in a Query/ConvertRecord. Then add an additional step to convert to Parquet or ORC for HDFS storage.

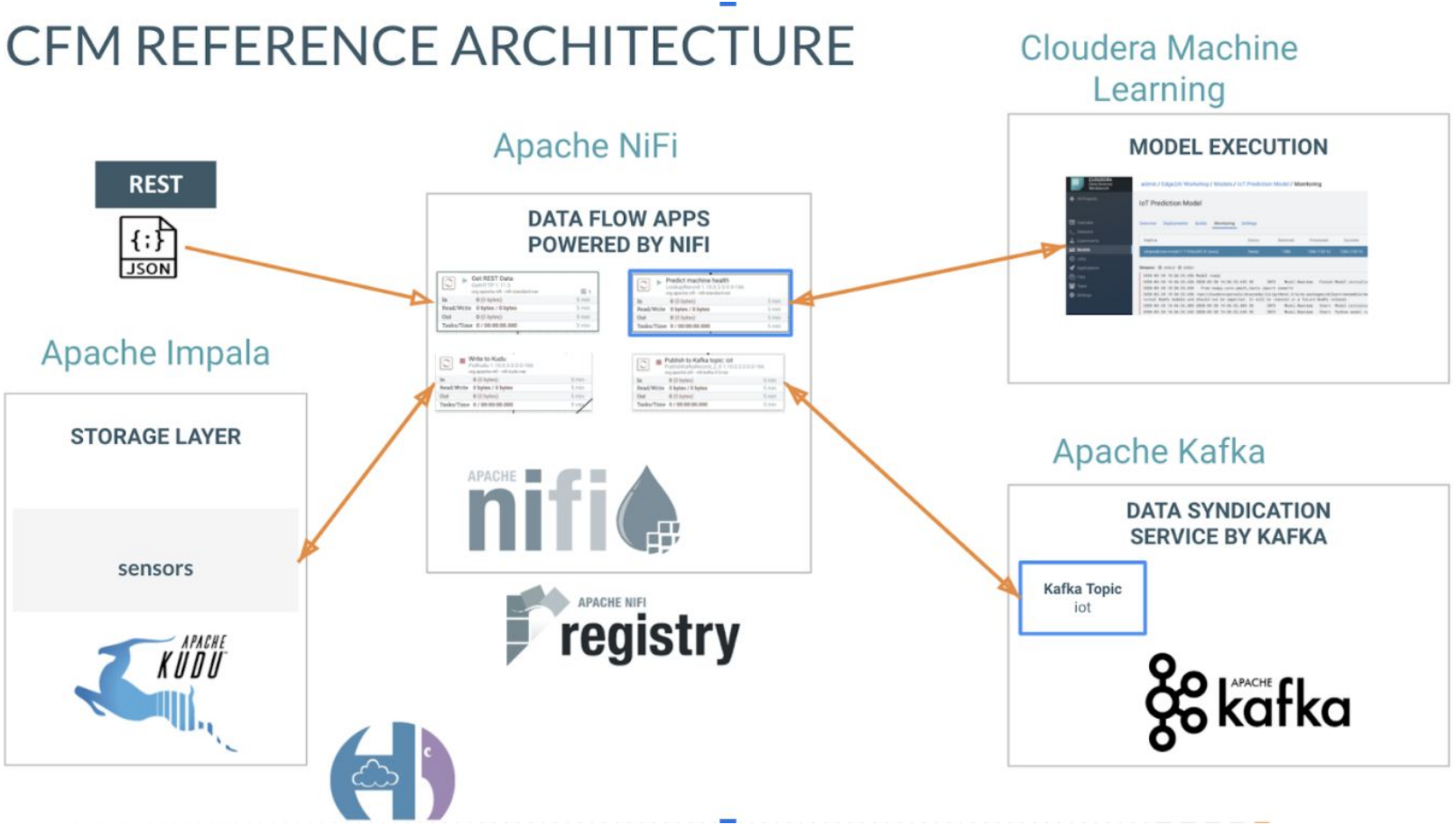
https://www.datainmotion.dev/2019/03/implementing-streaming-use-case-from.html

INGEST RDBMS TABLES



https://community.cloudera.com/t5/Community-Articles/Incrementally-Streaming-RDBMS-Data-to-Your-Hadoop-DataLake/ta-p/247927

CFM REFERENCE ARCHITECTURE



https://dzone.com/articles/lets-build-a-simple-ingest-to-cloud-data-warehouse