HORTONWORKS

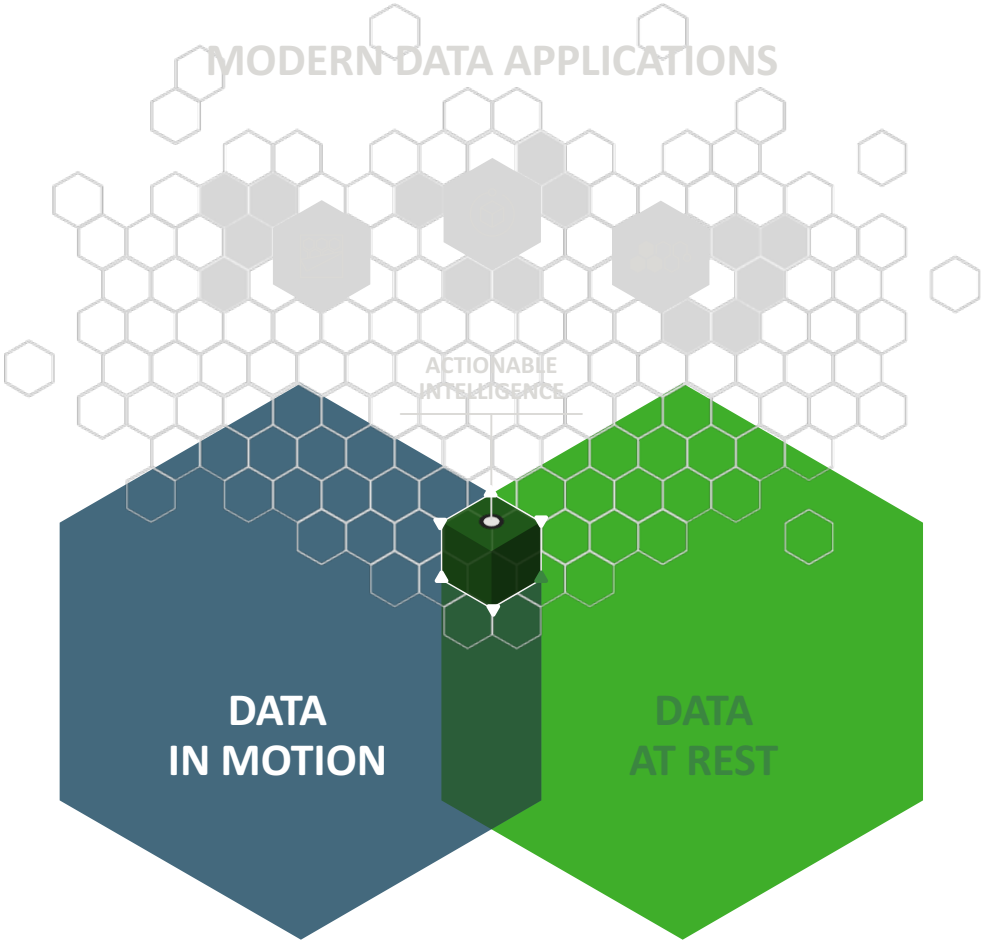
Hortonworks DataFlow 3.0

Actionable Intelligence from Connected Data Platforms

Capturing perishable insights from data in motion

Ensuring rich, historical insights on data at rest

Necessary for modern data applications



Hortonworks DataFlow

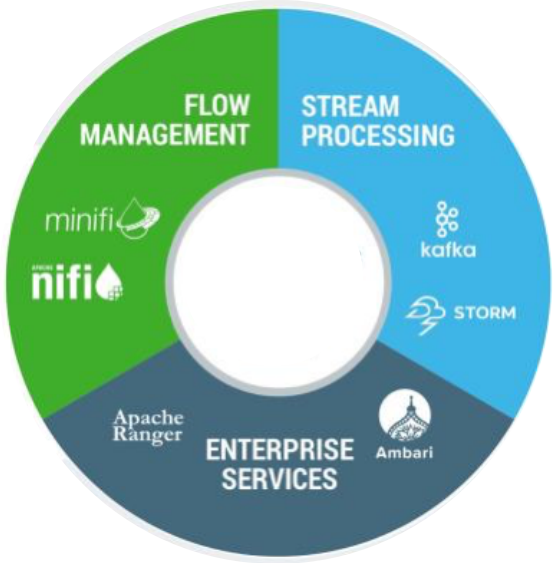
Hortonworks Data Platform

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Hortonworks DataFlow: Data-in-Motion Platform

Platform to build

DATAFLOW MANAGEMENT AND STREAMING ANALYTICS SOLUTIONS



to COLLECT, CURATE, ANALYZE **and** ACT ON

data in motion across the edge, data center and cloud

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

HORTONWORKS

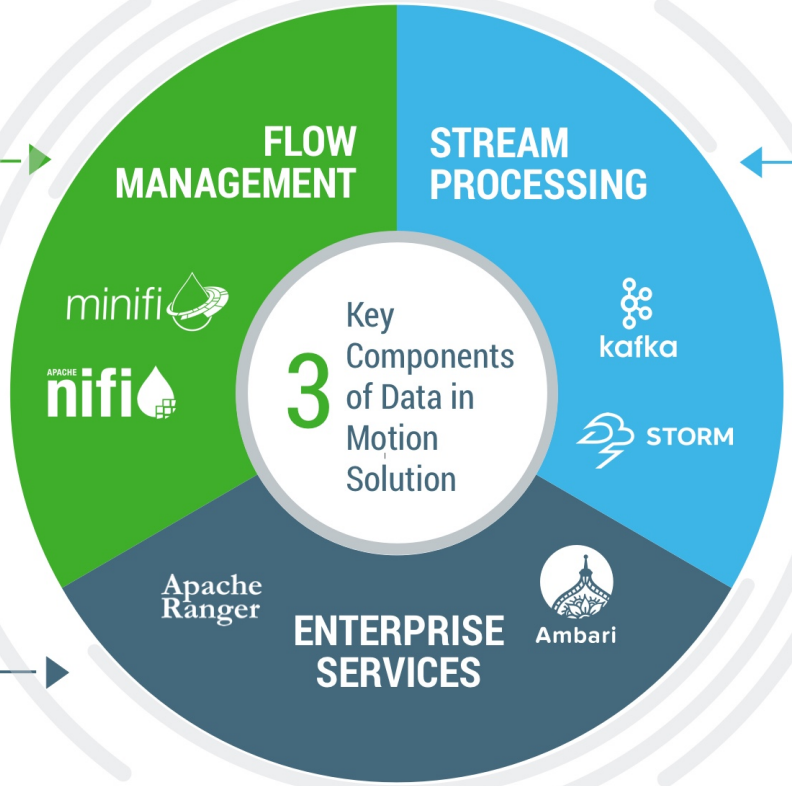
Key Components of Data in Motion Solution

EASY, SECURE, RELIABLE WAY TO GET THE DATA YOU NEED

From the edge to anywhere with intelligence

IMMEDIATE AND CONTINUOUS INSIGHTS

Because acting on perishable insights in real time maximizes value



PROVISIONING, MANAGEMENT, MONITORING, SECURITY, AUDIT, COMPLIANCE, GOVERNANCE

Because it has to work together in an enterprise environment

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

HORTONWORKS

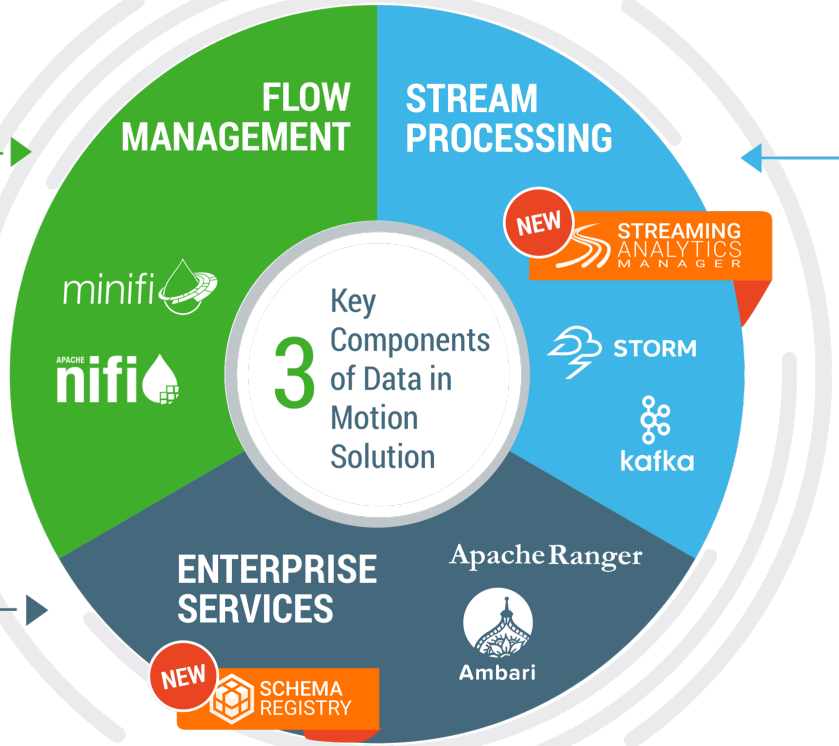
HDF Data-In-Motion Platform – with HDF 3.0 Announcement

EASY, SECURE, RELIABLE WAY TO GET THE DATA YOU NEED

From the edge to anywhere with intelligence

PROVISIONING, MANAGEMENT, MONITORING, SECURITY, AUDIT, COMPLIANCE, GOVERNANCE

Because it has to work together in an enterprise environment



IMMEDIATE AND CONTINUOUS INSIGHTS

Because acting on perishable insights in real time maximizes value

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

HORTONWORKS

Flow Management Powered by Apache NiFi/MiniFi

© Hortonworks Inc. 2011 – 2017. All Rights Reserved

Flow Management with Apache NiFi and Apache MiniFi

Enable easy ingestion, routing, management and delivery of any data anywhere (Edge, cloud, data center) to any downstream system with built in end-to-end security and provenance

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

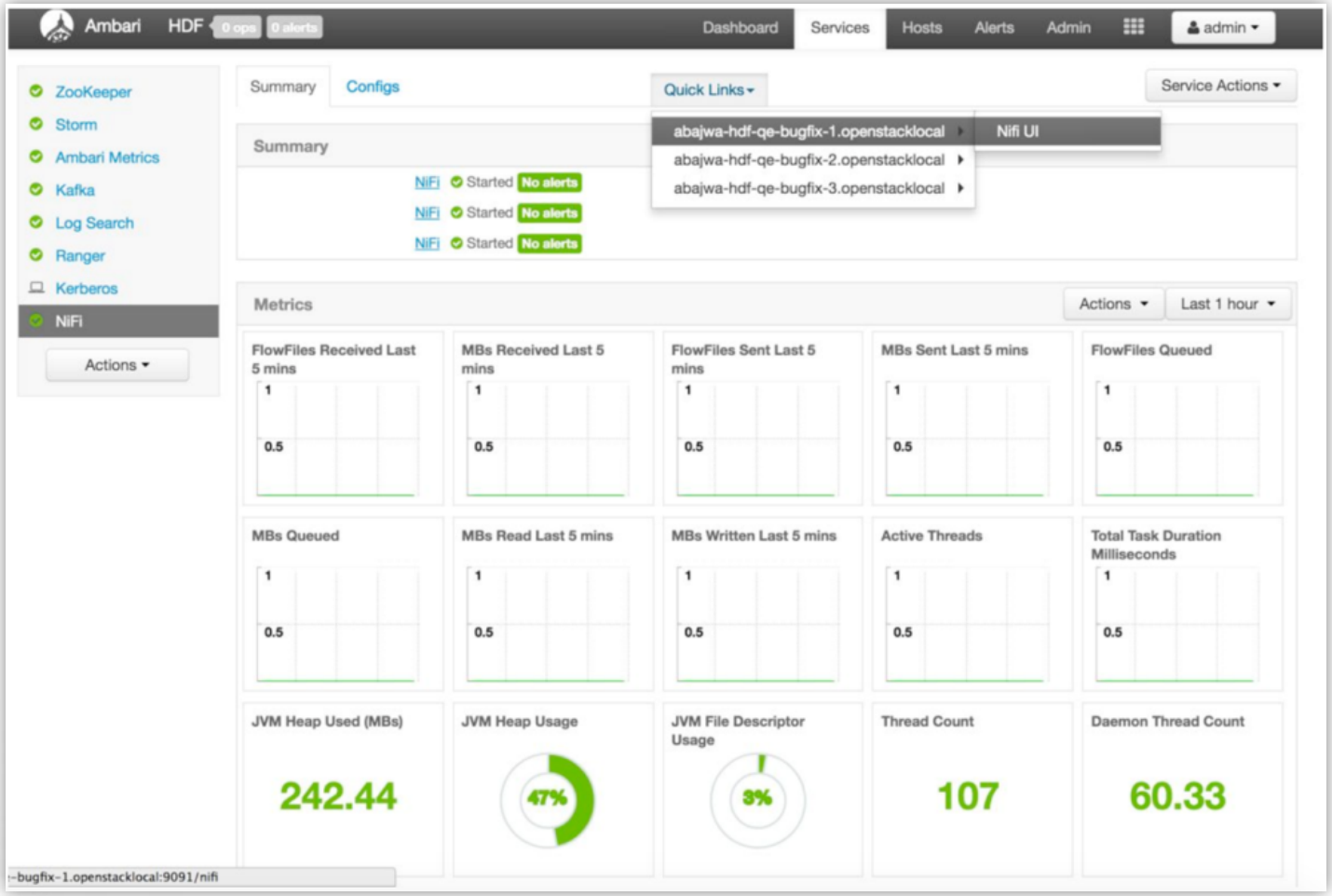
HORTONWORKS

Single Ambari/Ranger Managing HDF and HDP Services

- **The Benefits:**
  - Make NiFi available as an add-on service to HDP stack
  - Single Ambari for cluster management, single Ranger for policy management
  - Benefit customers that have both HDF and HDP
  - Optimized manageability and reduced operational overhead

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Ambari Integration



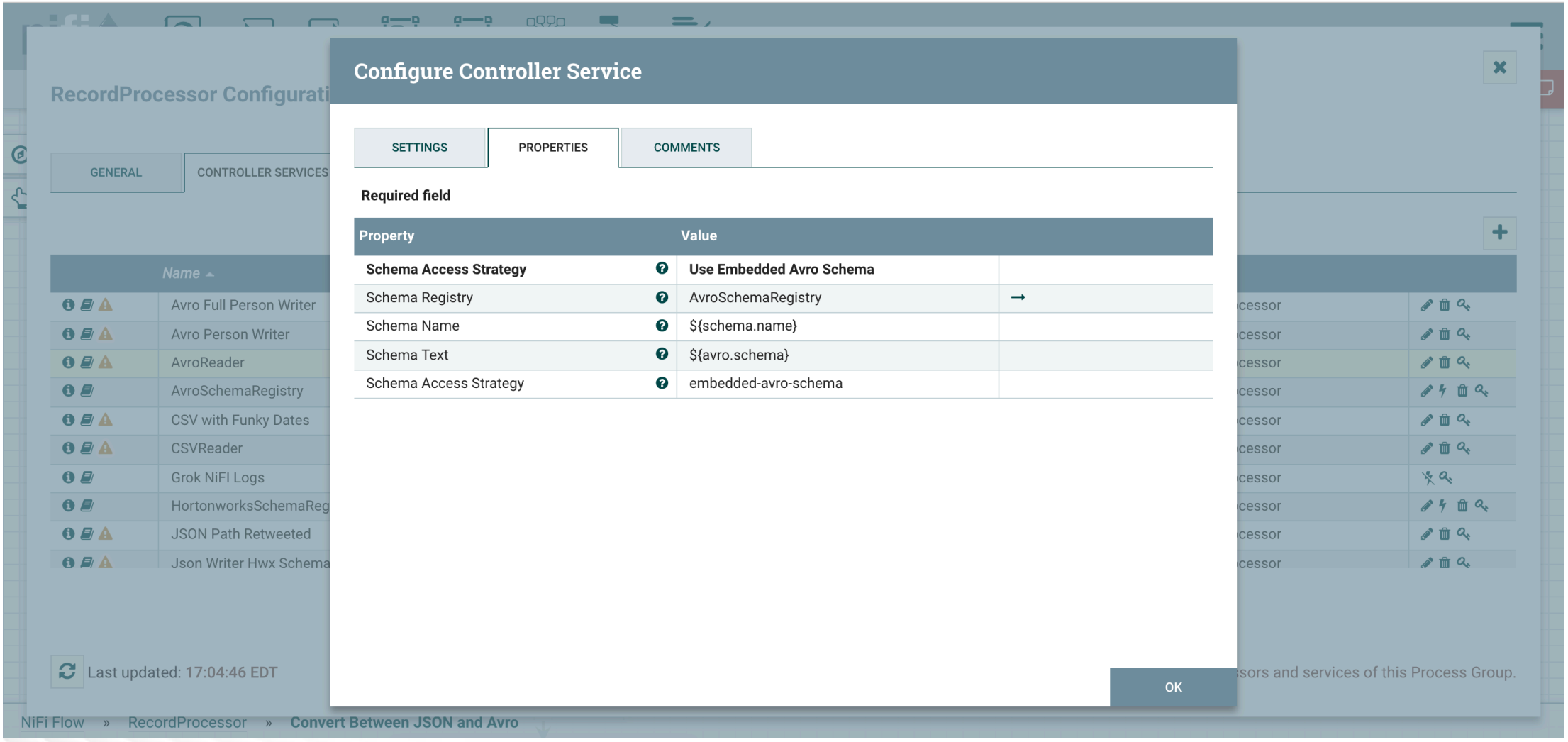
© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Format and Schema Aware Efficient Flow Management

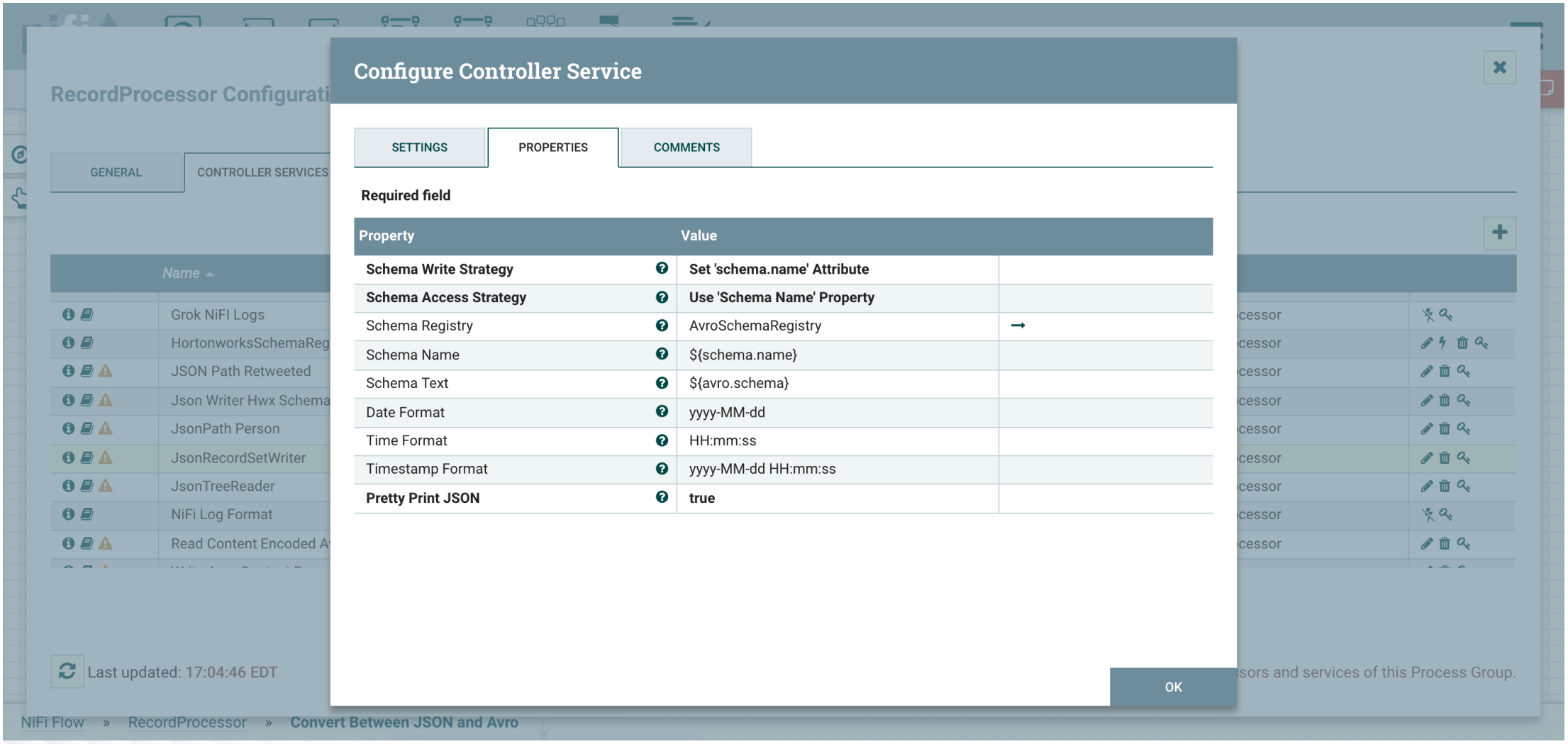
- Provide schema aware record based processors for common processing patterns
  - Split, Enrich, Partition, Convert, Query (SQL queries powered by Apache Calcite)
  - Put/Get records between NiFi and Kafka, ElasticSearch, RDMBS (more soon)
  - Easy bridging to/from Columnar data formats like ORC or Parquet
- Separate format/schema specific logic into extensible record readers and writers
  - Developers can write new readers/writers
  - Users can create new readers/writers with scripting live in production!
- So what?
  - Integration with Hortonworks Schema Registry and extensible for others
  - Format and schema aware processing *with* generic reusable components
  - Accelerate application development lifecycle
  - Easily integrate schema changes.

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

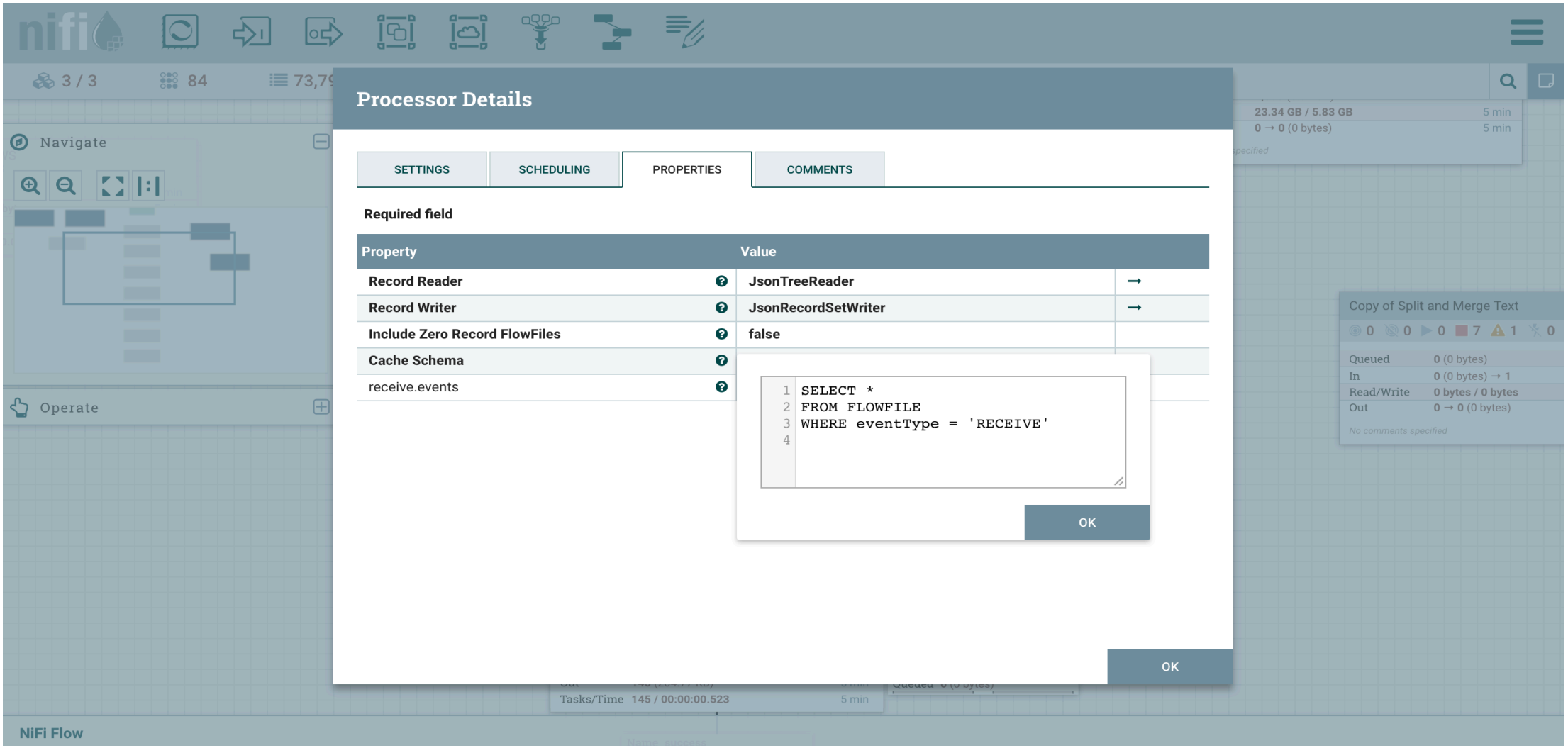
Record Reader CS



Record Writer CS



‘QueryRecord’ Processor – Treat streaming records as tables

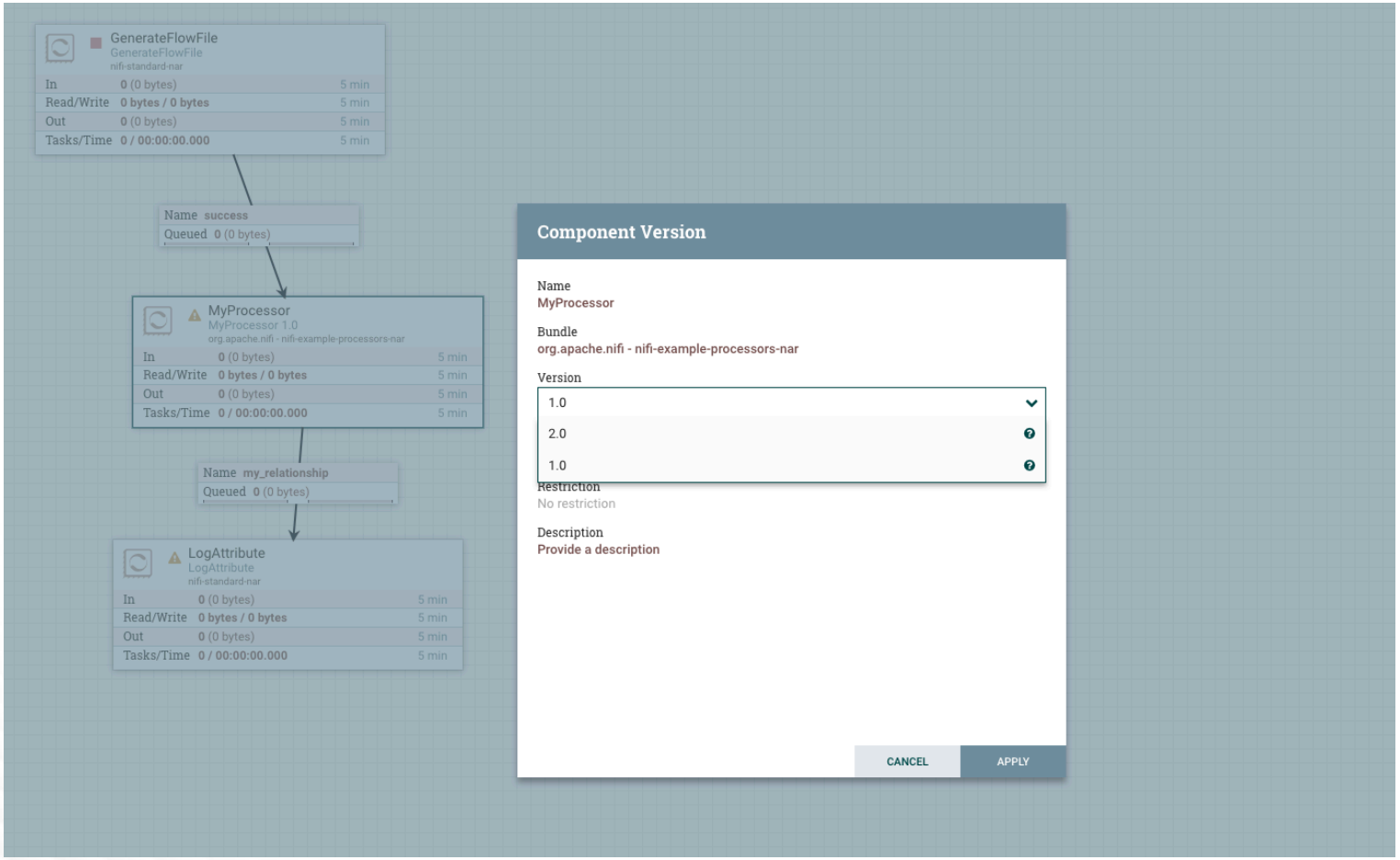


Component Versioning

- **What?**
  - Support multiple versions of the same component in a single NiFi instance
  - Enable direct mapping to pre-configured/tested flows into production
  - Enable smooth transition from old configurations to new configurations (and code)
  - Rollback to an older version or upgrade to a new version with the click of a button
  - Lead to live code uptake at runtime and integration with extension registry

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Component Versioning



© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Stream Processing

© Hortonworks Inc. 2011 – 2017. All Rights Reserved

Stream Processing – Introducing Streaming Analytics Manager (SAM)

A brand new product module in the HDF stack to design, develop, deploy and manage streaming analytics app with a drag-and-drop user experience

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Key Challenges to Build Streaming Analytics Applications

- Need specialized skillset and considerable amount of low level programming, testing and tuning.
- Extensive time it takes to design, develop, test, deploy into production/
- Traditional mature streaming vendors (IBM Streams, Tibco, SAS, SAP) are cost prohibitive and closed source. No true open source tools today.

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

What does SAM offer ?

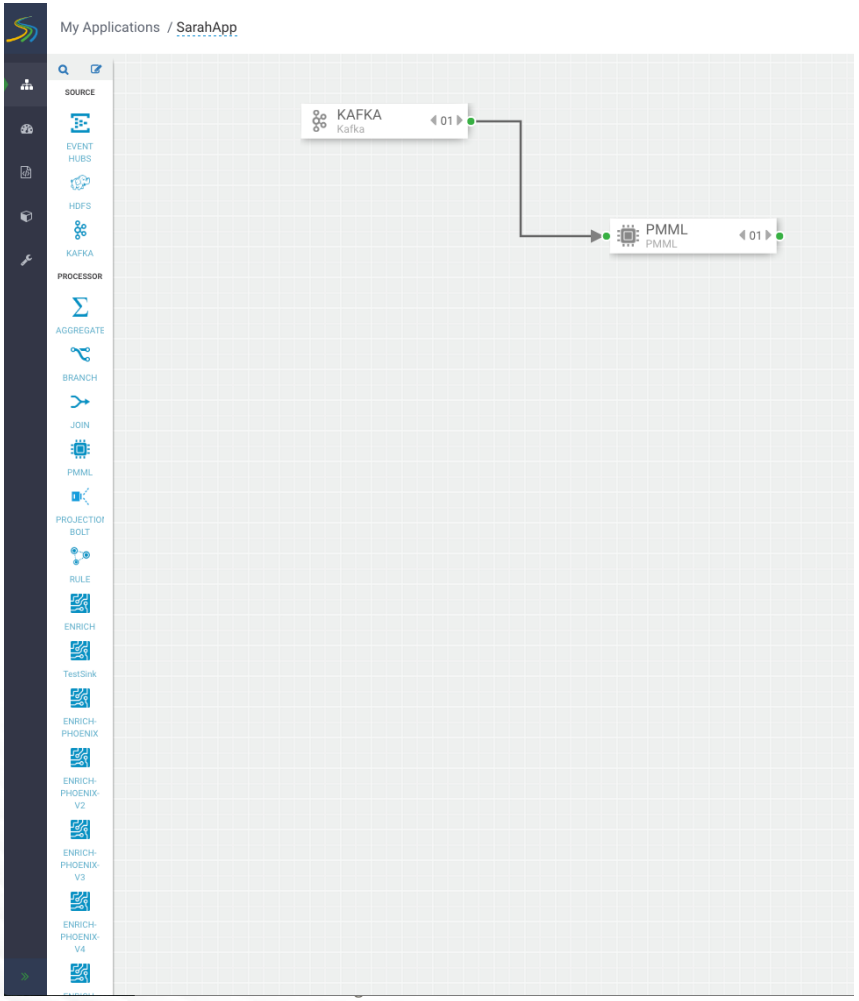
- **Build and deploy complex stream applications without writing any code**
- **Only open source tool in the market** with graphical programming paradigm
- Speed time-to-market to build complex **streaming analytics applications**
- Build streaming analytics applications **without specialized skillsets**.
- Decouple data format from the streaming application itself while being schema aware
- Support multiple underlining streaming engines

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

SAM's 3 Components
Builder – Ops - Insight

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Stream Builder Module for App Developers



- Builder components, shown on the canvas palette, are the building blocks used by the app developer to build streaming applications
- Drag and drop to build a working streaming application without writing a single line of code
- 4 Types of Components: Sources, Processors, Sinks and Custom

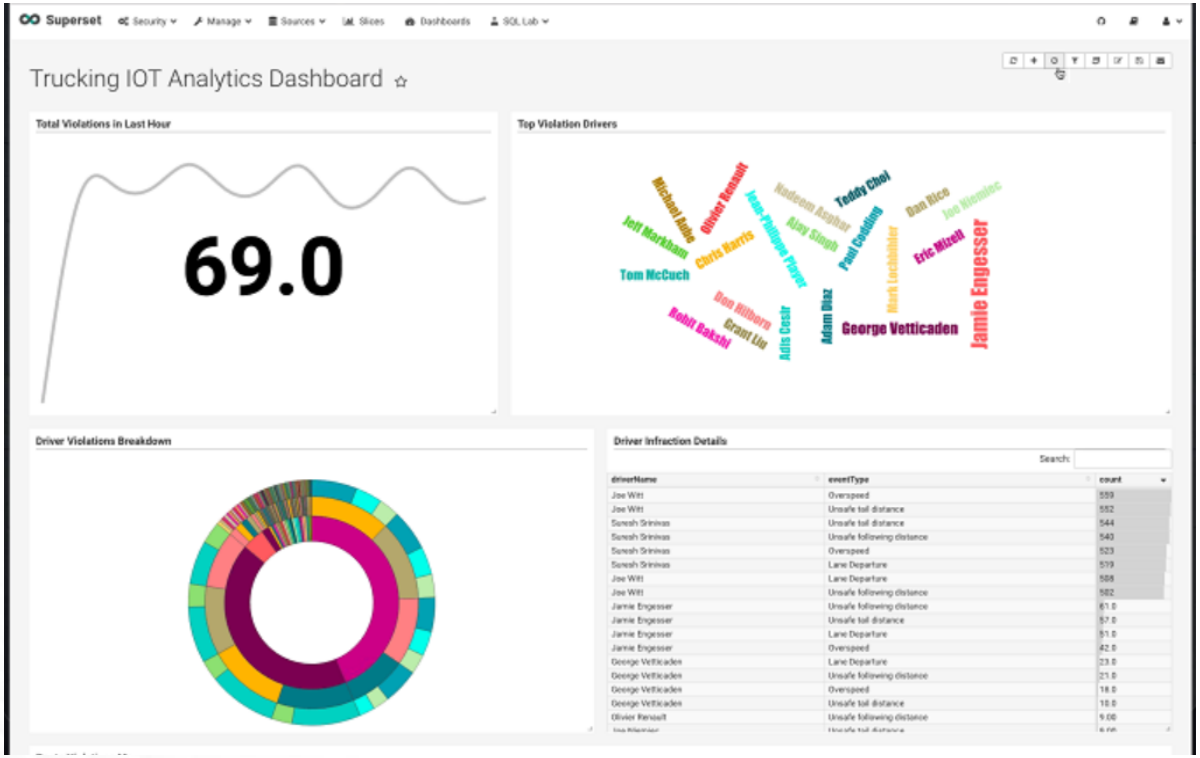
Extensibility with StreamLine SDK

- Custom Components
  - Most users will want to build custom components to meet certain requirements.
  - StreamLine provides the ability to add custom component using the StreamLine SDK.

- Custom aggregate functions for Windowing
  - SDK can be used to create custom UDF functions for windowed aggregations

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

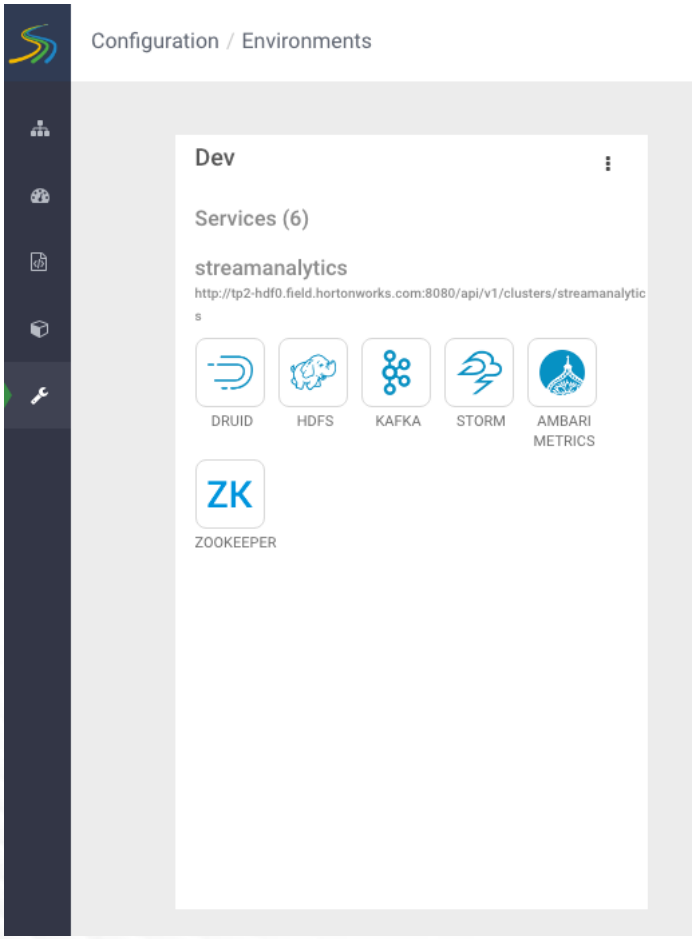
Stream Insight Module for Business Analysts



- A tool to create real-time analytics dashboards, charts and graphs
- 30+ visualization charts out of the box with customization capability
- Druid is the Analytics Engine that powers the Stream Insight Module.

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

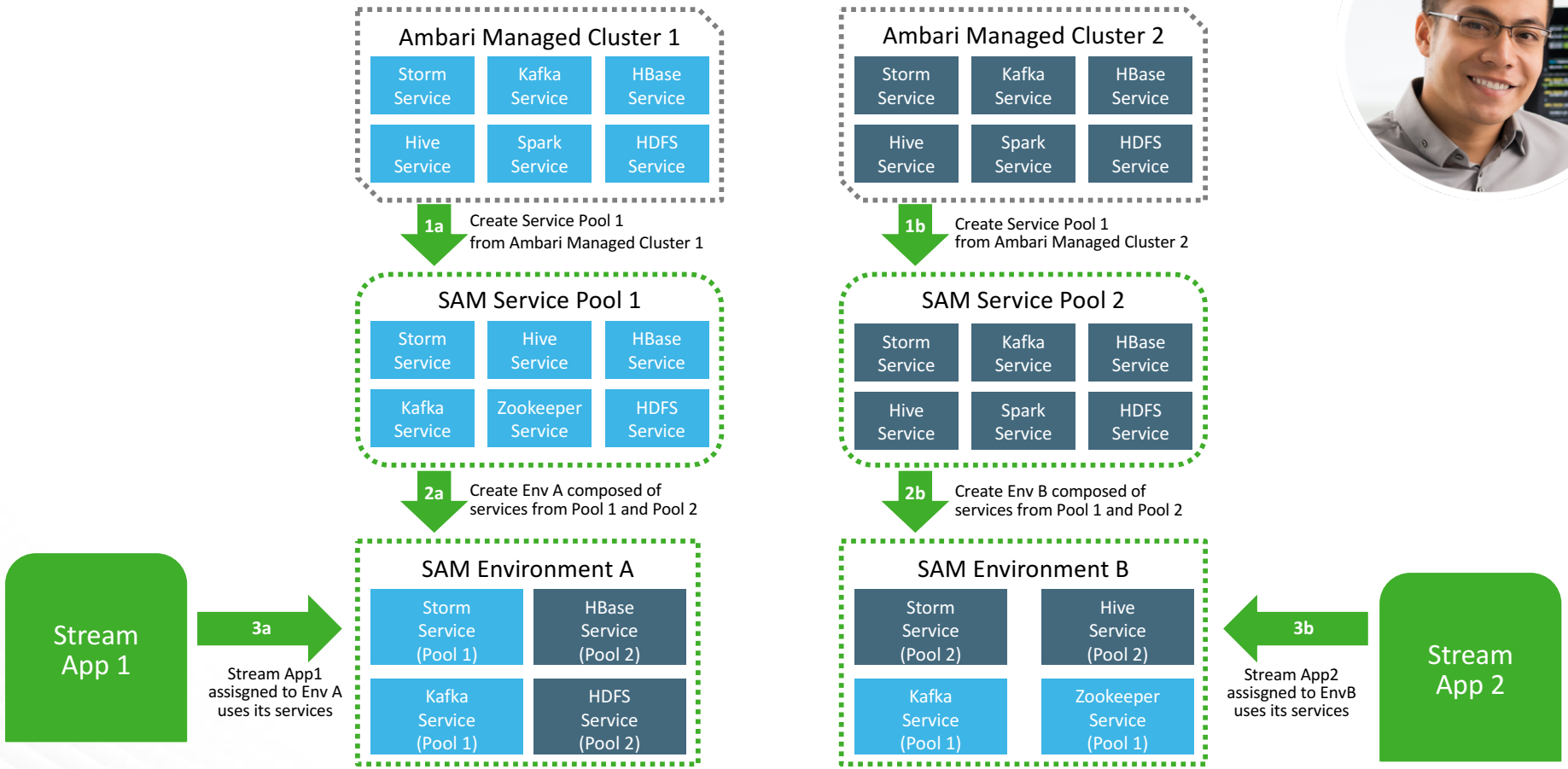
Stream Ops Module for IT Operations



- Create and manage different environments in which individual streaming applications will be built
- Environments consists of services such as HDFS, Kafka, Storm from different service pools
- Save time and reduce operational overhead with same drag and drop paradigm as the stream build module

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

SAM's Service Pools and Environments



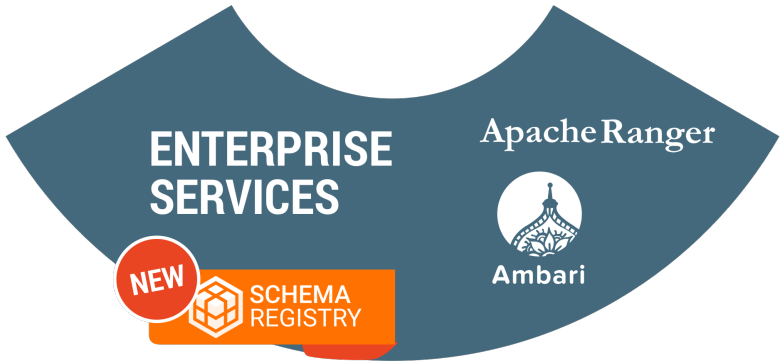
© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Enterprise Services

© Hortonworks Inc. 2011 – 2017. All Rights Reserved

Enterprise Services – Introducing Schema Registry

A shared repository of schemas that allows applications to save, retrieve and reuse schemas and flexibly interact with each other



© Hortonworks Inc. 2011 – 2017 All Rights Reserved

What Value Does Schema Registry Provide?

- **Data Governance**
  - Centralized registry to provide reusable schemas
  - Version management to define relationship between schemas
  - Validation to enable generic format conversion and generic routing

- **Operational Efficiency**
  - Centralized registry to avoid attaching schema to every piece of data
  - Version management to enable consumers and producers to evolve at different rates
  - Validation to ensure data quality

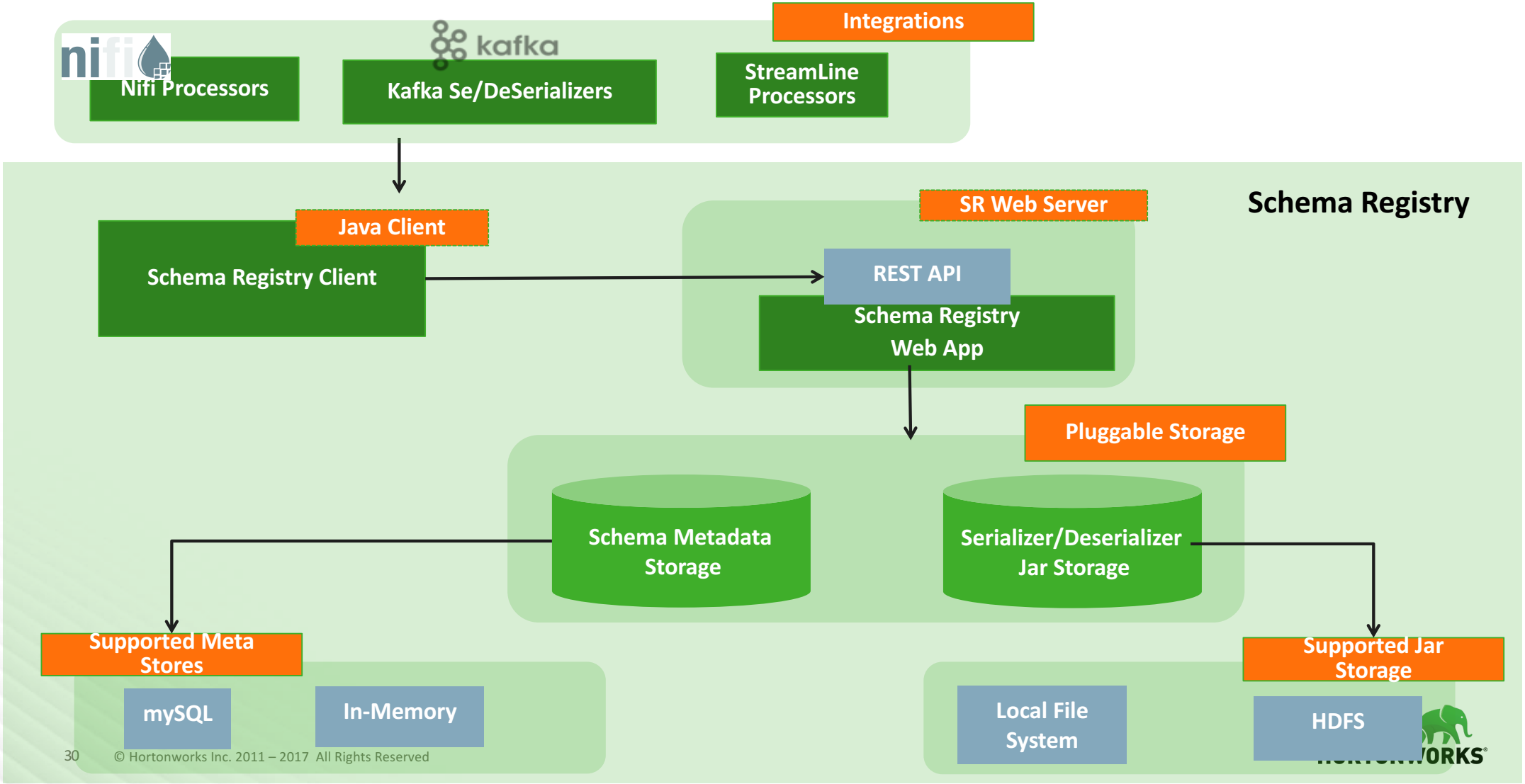
© Hortonworks Inc. 2011 – 2017 All Rights Reserved

How Schema Registry Work with the Rest of the Platform

- **NiFi** Processors for Schema Registry
  - Schema aware flow management (record readers/writers)
  - Live schema reference / automated schema conversion
- **Streaming Analytics Manager** processors for Schema Registry
  - For example: Lookup a schema of a Kafka Topic
  - Context/schema aware user experience eases time-to-market of building stream apps
- **Atlas** integration with Schema Registry (Future release)
  - Just like Atlas pull schema info from Hive MetaStore, Atlas will be able to capture schema, format and semantic metadata from events in HDF via the registry.

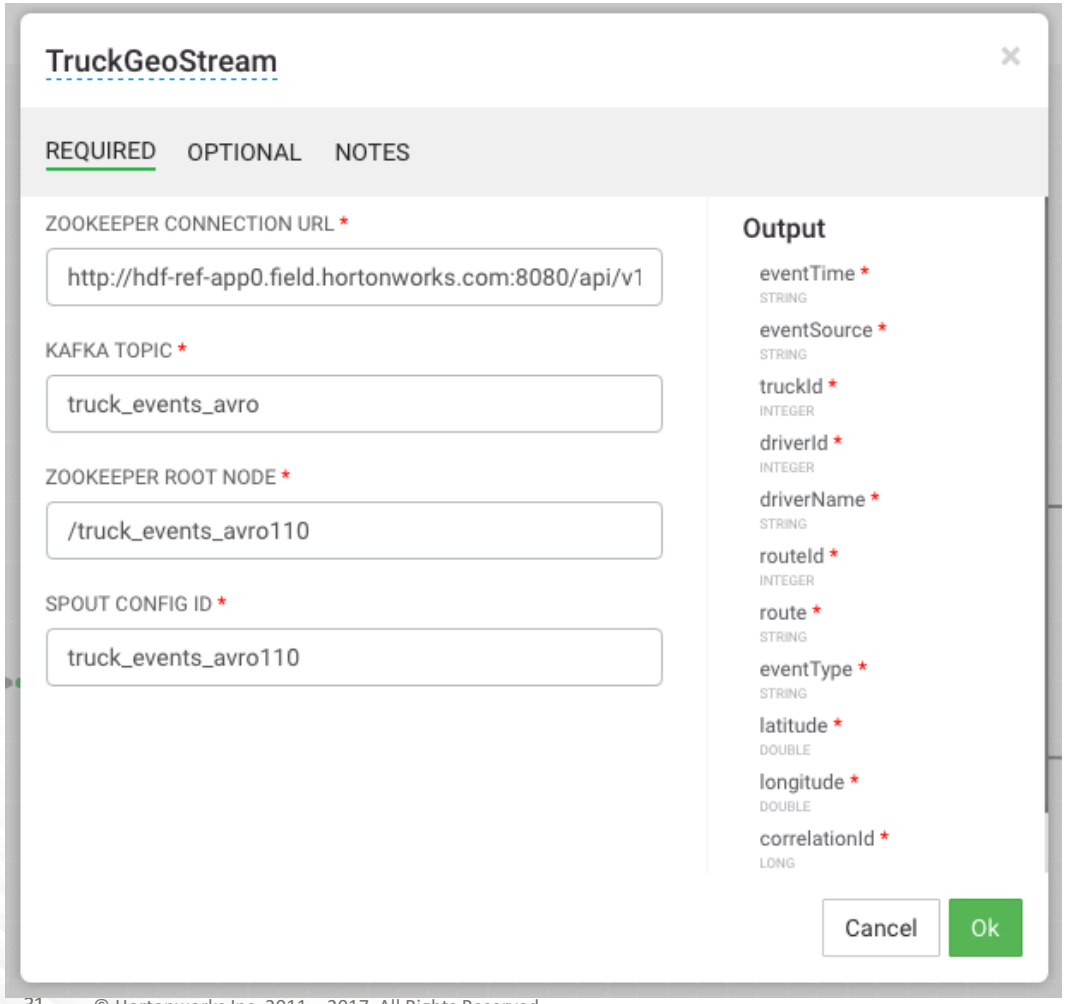
© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Schema Registry Component Architecture



© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Example – How SAM and Schema Registry are Integrated



- SAM's Kafka Source Component integrated with Schema Registry
  - Every SAM builder component requires a schema to function.
  - When you configure a Kafka source and supply kafka topic, SAM calls the Schema Registry.
  - Using the Kafka topic as the key, SAM will retrieve the schema.
  - This schema is then displayed on the tile component, and is passed to downstream components.

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

1 New Controller Service & 7 New Nifi Processors Introduced Integrated with Schema Registry

Type	Name	Description
Nifi Controller Service	**SchemaRegistryService**	• Exposes operations to serialize and deserialize the contents of the FlowFile as well as the operations to query the actual schema. NOTE: at the moment only AVRO schema type is supported.
Nifi Processor	**UpdateAttributeViaSchemaRegistry**	• Retrieve the schema for a given schema name and version (optional) and inserts textual representation of the retrieved schema 'schema.text' attribute of the outgoing FlowFile. • A typical application of such processor may be to integrate with legacy flows where a processor requiring schema text (i.e., ConvertAvroToJson) may now retrieve such text from FlowFile attribute
Nifi Processor	**TransformCSVToAvroViaSchemaRegistry**	• Transforms the CSV contents of the FlowFile to Avro format using Schema and Serializer provided by the Schema Registry.
Nifi Processor	**TransformAvroToCSVViaSchemaRegistry**	• Transforms the Avro contents of the FlowFile to CSV format using Schema and Deserializer provided by the Schema Registry.
Nifi Processor	**TransformJsonToAvroViaSchemaRegistry**	• Transforms the Json contents of the FlowFile to Avro format using Schema and Deserializer provided by the Schema Registry.
Nifi Processor	**TransformAvroToJsonViaSchemaRegistry**	• Transforms the Avro contents of the FlowFile to Json format using Schema and Deserializer provided by the Schema Registry.
Nifi Processor	**TransformCSVToJsonViaSchemaRegistry**	• Transforms the CSV contents of the FlowFile to Json format using Schema and Deserializer provided by the Schema Registry.
Nifi Processor	**TransformJSONToCSVViaSchemaRegistry**	• Transforms the JSON contents of the FlowFile to CSV format using Schema and Deserializer provided by the Schema Registry.
Nifi Processor	**ExtractAvroFieldsViaSchemaRegistry**	• Similar to ExtractText, this processor extracts Avro fields from the incoming Avro record and sets their values as FlowFile attributes via Dynamic Properties feature of NiFi.

Demo

© Hortonworks Inc. 2011 – 2017. All Rights Reserved

MiniFi

© Hortonworks Inc. 2011 – 2017. All Rights Reserved

MiNiFi differentiation

- Let me get the key parts of NiFi close to where data begins and provide bidirectional communication
- NiFi lives close to the data center. Give it an enterprise server or a cluster of them.
- MiNiFi lives at the edge to where data is born and is a guest on that device or system

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

HORTONWORKS

Apache ~~NiFi~~ MiNiFi

Key Features

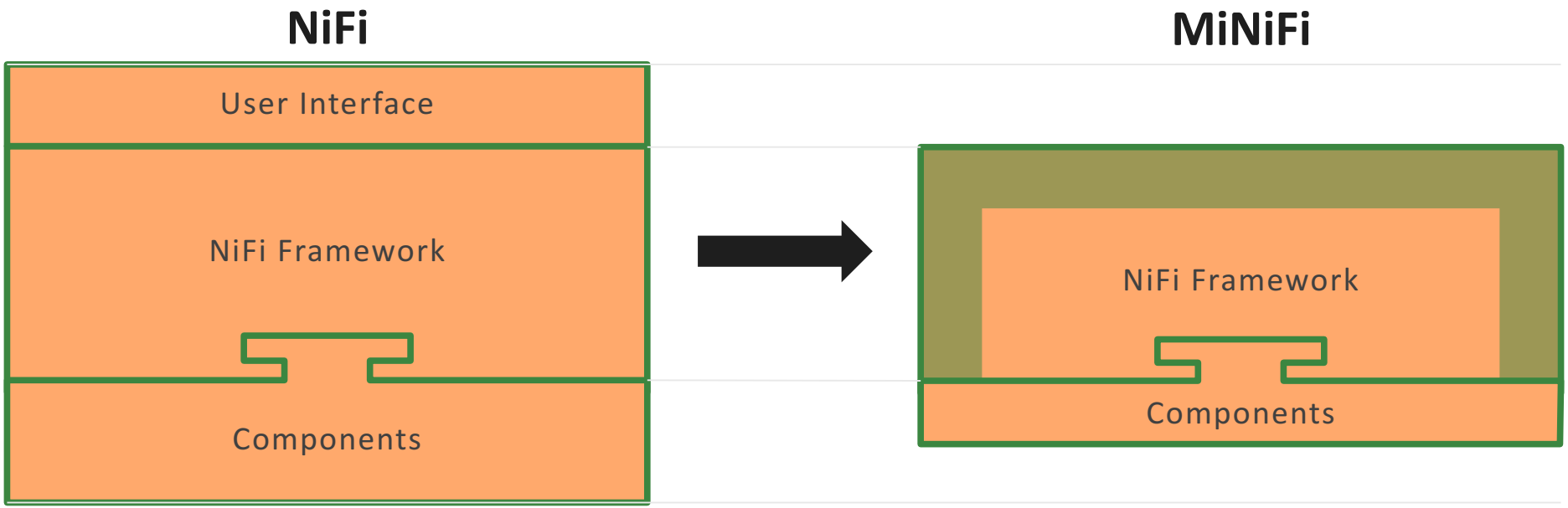
- Guaranteed delivery
- Data buffering
  - Backpressure
  - Pressure release
- Prioritized queuing
- Flow specific QoS
  - Latency vs. throughput
  - Loss tolerance
- Data provenance
- Recovery/recording a rolling log of fine-grained history
- Designed for extension
- Design and Deploy
- Warm re-deploys

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

MiNiFi 0.0.1-Java

Hortonworks

NiFi vs MiNiFi Java Processes



© Hortonworks Inc. 2011 – 2017 All Rights Reserved

MiNiFi 0.0.1-Java

- Declarative configuration of processing flows through a YAML configuration file
- Exporting of provenance events to another NiFi instance via a Reporting Task over Site to Site
- Flow change configuration watcher implementations that provide reloading a NiFi instance when receiving an updated flow over REST or changes on a file system
- Providing a mechanism to query an instance's status
- <40mb binary distribution

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

MiNiFi Toolkit

CLI to facilitate config.yml building

- Convert NiFi templates to config.yml
- Validate config.yml files

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

minifi.sh flowStatus

Get flow status at the command line

- Components
- Instance
- System Diagnostics

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

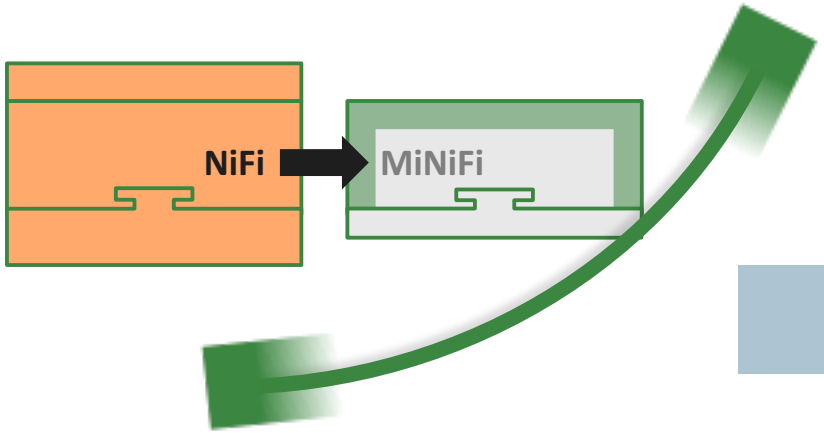
Same Extensible framework (nars)

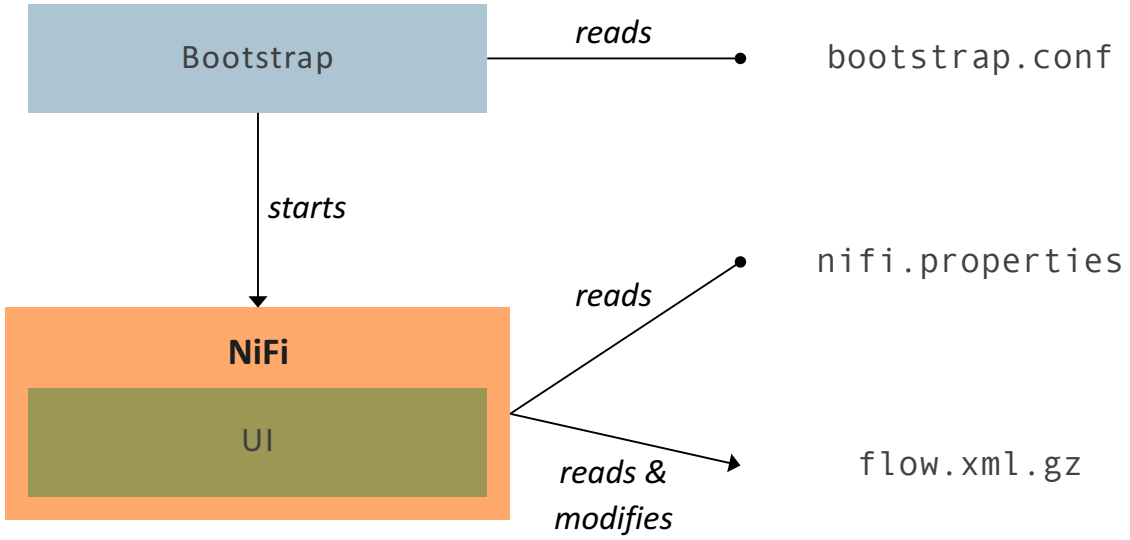
Allows MiNiFi to use any NiFi processors

- By default the 0.6.1 standard processors are included (~20mb)
  - Tailing a Log
  - UpdateAttribute
  - Routing by content or attributes
  - PutEmail

© Hortonworks Inc. 2011 – 2017 All Rights Reserved

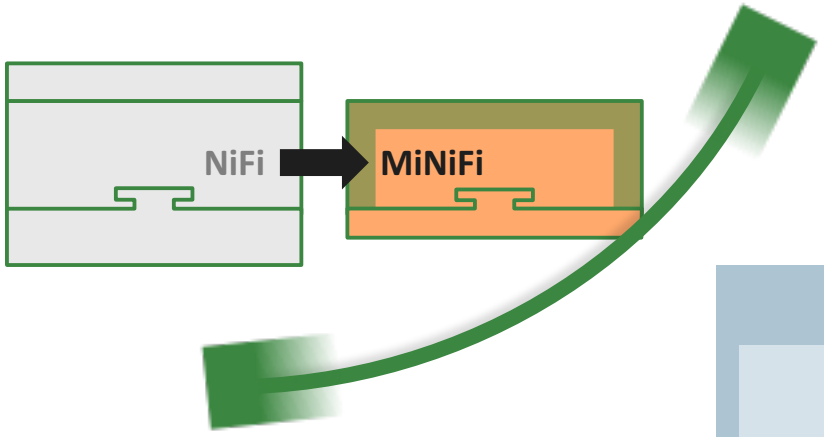
NiFi Java Processes

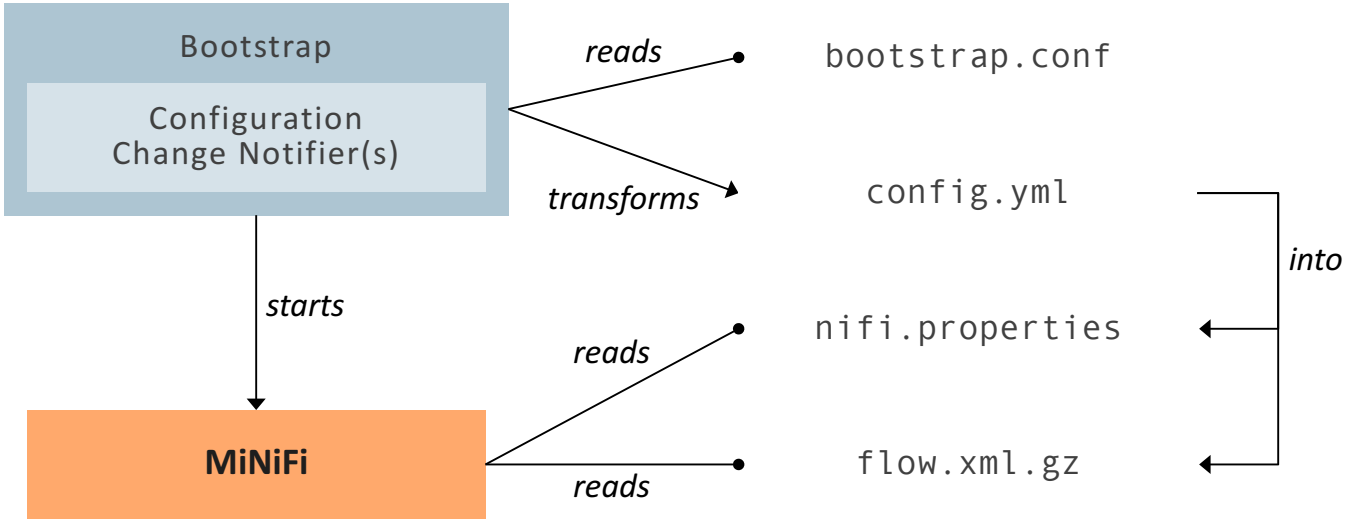




© Hortonworks Inc. 2011 – 2017 All Rights Reserved

MiNiFi Java Processes





© Hortonworks Inc. 2011 – 2017 All Rights Reserved

Thank You

© Hortonworks Inc. 2011 – 2017. All Rights Reserved