Integrating Event Streams and File Data with Apache Flume and Apache NiFi

Joey Echeverria | April 13, 2015

SCALINGDATA

Data integration

Data integration

- Multiple data source

Data integration

- Multiple data source
- Questions

Challenges

Challenges

- Unique sources

Challenges

- Unique sources
  - Format

Challenges

- Unique sources
  - Format
  - Schema

Challenges

- Unique sources
  - Format
  - Schema
  - Protocol

Challenges

- Unique sources
  - Format
  - Schema
  - Protocol
  - Batchiness

Challenges

- Unique sources
  - Format
  - Schema
  - Protocol
  - Batchiness
- Big data

Traditional (Hadoop) approach

Traditional (Hadoop) approach

- In so far as anything with Apache Hadoop can be called “traditional”

Traditional (Hadoop) approach

- Identify source class

Traditional (Hadoop) approach

- Identify source class
  - Event streams

Traditional (Hadoop) approach

- Identify source class
  - Event streams
  - Database tables

Traditional (Hadoop) approach

- Identify source class
  - Event streams
  - Database tables
  - Files

Traditional (Hadoop) approach

- Map class to system

Traditional (Hadoop) approach

- Map class to system
  - Event streams ➔ Apache Flume

Traditional (Hadoop) approach

- Map class to system
  - Event streams ➔ Apache Flume
  - Database tables ➔ Apache Sqoop

Traditional (Hadoop) approach

- Map class to system
  - Event streams ➔ Apache Flume
  - Database tables ➔ Apache Sqoop
  - Files ➔ `hdfs dfs -put`?

Integrate in the repository

Integrate in the repository

- Ingest raw data

Integrate in the repository

- Ingest raw data
  - Raw database tables?

Integrate in the repository

- Ingest raw data
  - Raw database tables?
  - Raw events?

Integrate in the repository

- Ingest raw data
  - Raw database tables?
  - Raw events?
- MapReduce jobs for ETL

Use case

Use case

- Completely contrived for this presentation, but maybe you really want to do this

Use case

- Data sources

Use case

- Data sources
  - Twitter fire hose

Use case

- Data sources
  - Twitter fire hose*

*1%

Use case

- Data sources
  - Twitter fire hose*
  - My tweet archive

*1%

Use case

- Data sources
  - Twitter fire hose*
  - My tweet archive
- Goal

*1%

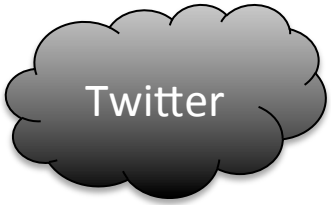
Use case

- Data sources
  - Twitter fire hose*
  - My tweet archive
- Goal
  - Identify the user most similar to me

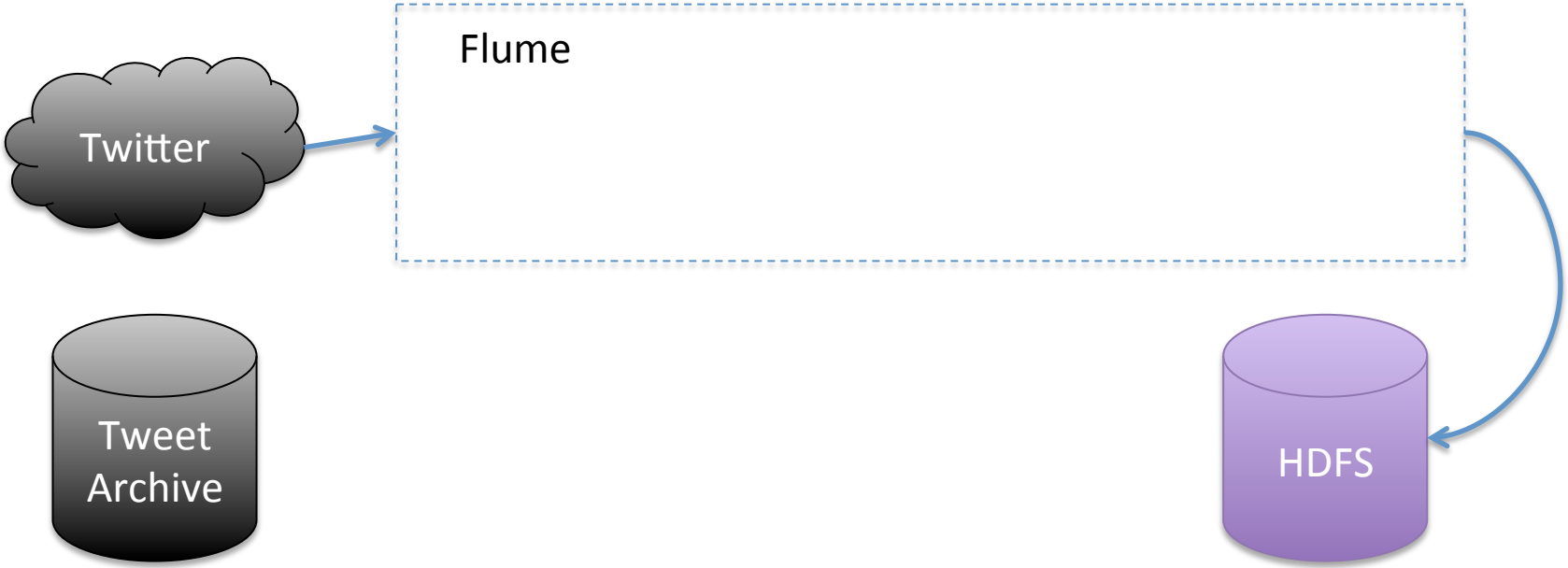
*1%

(Mostly) traditional solution

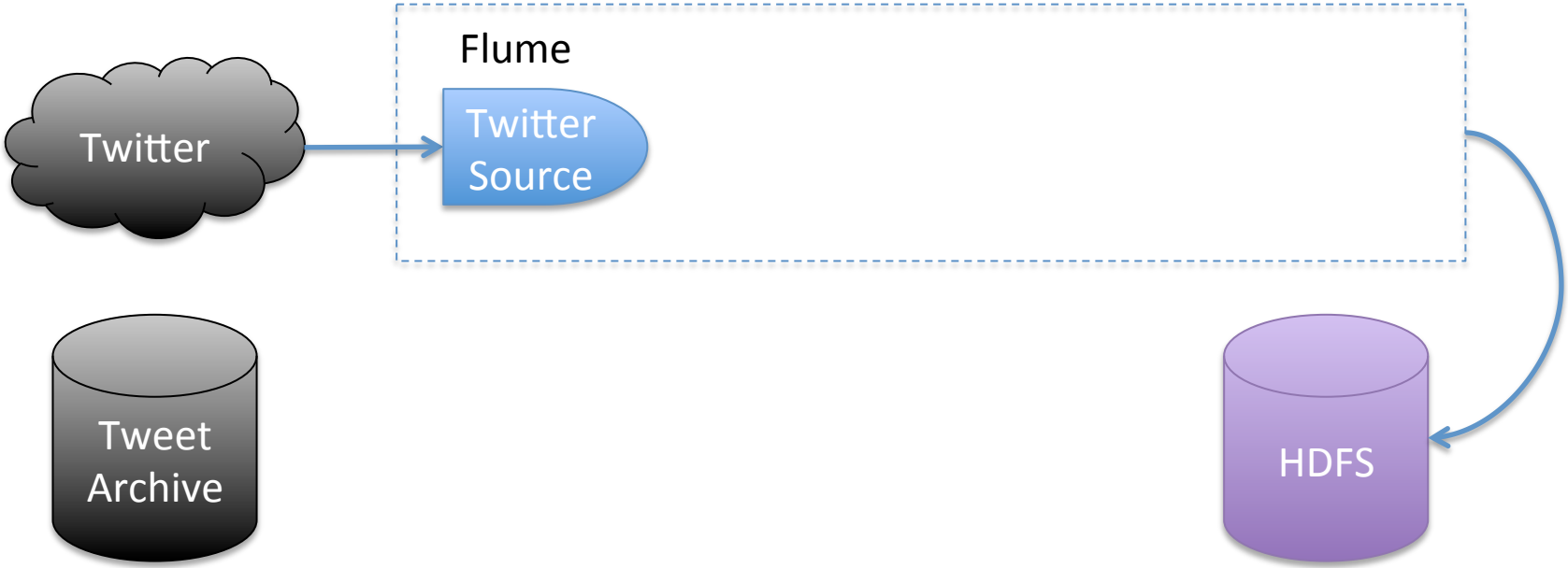
(Mostly) traditional solution



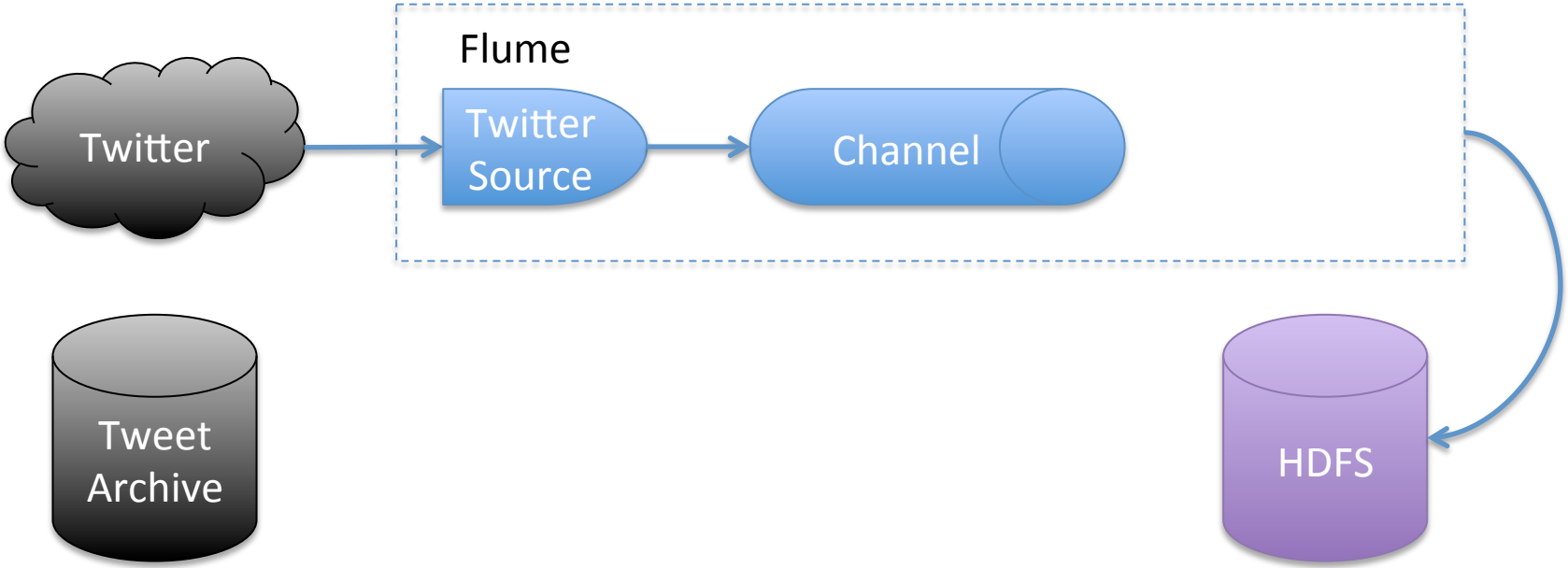
(Mostly) traditional solution



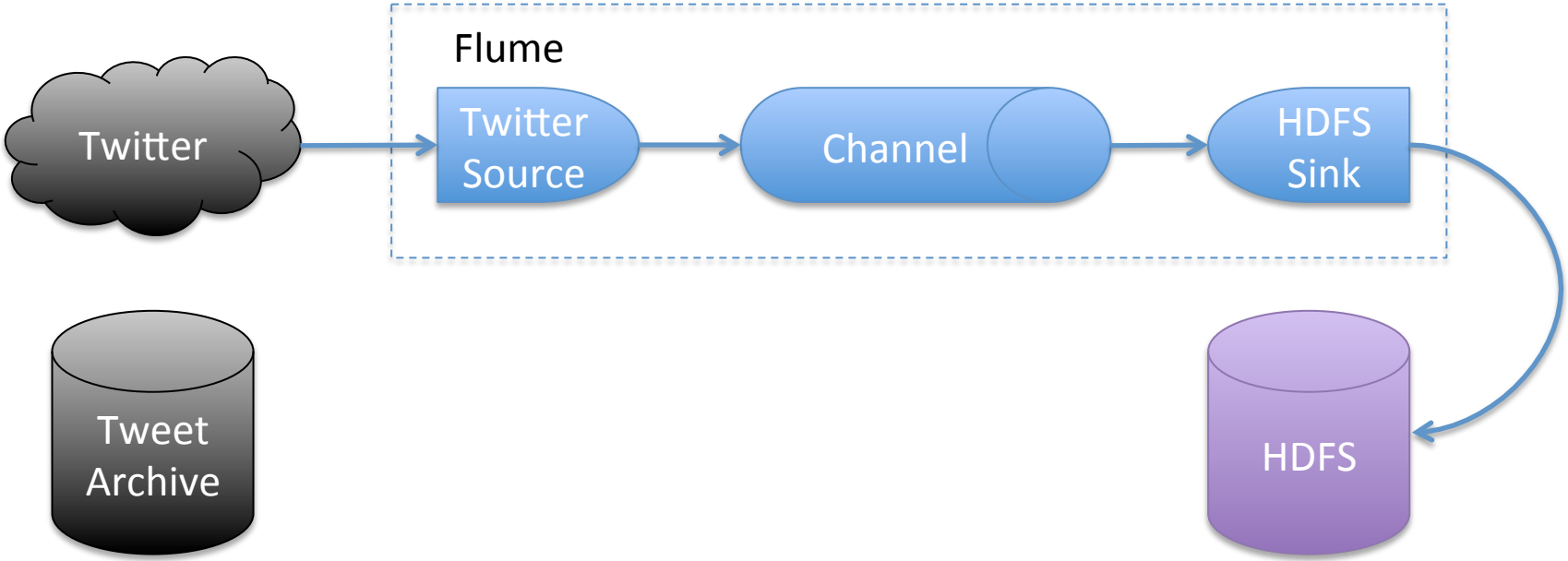
(Mostly) traditional solution



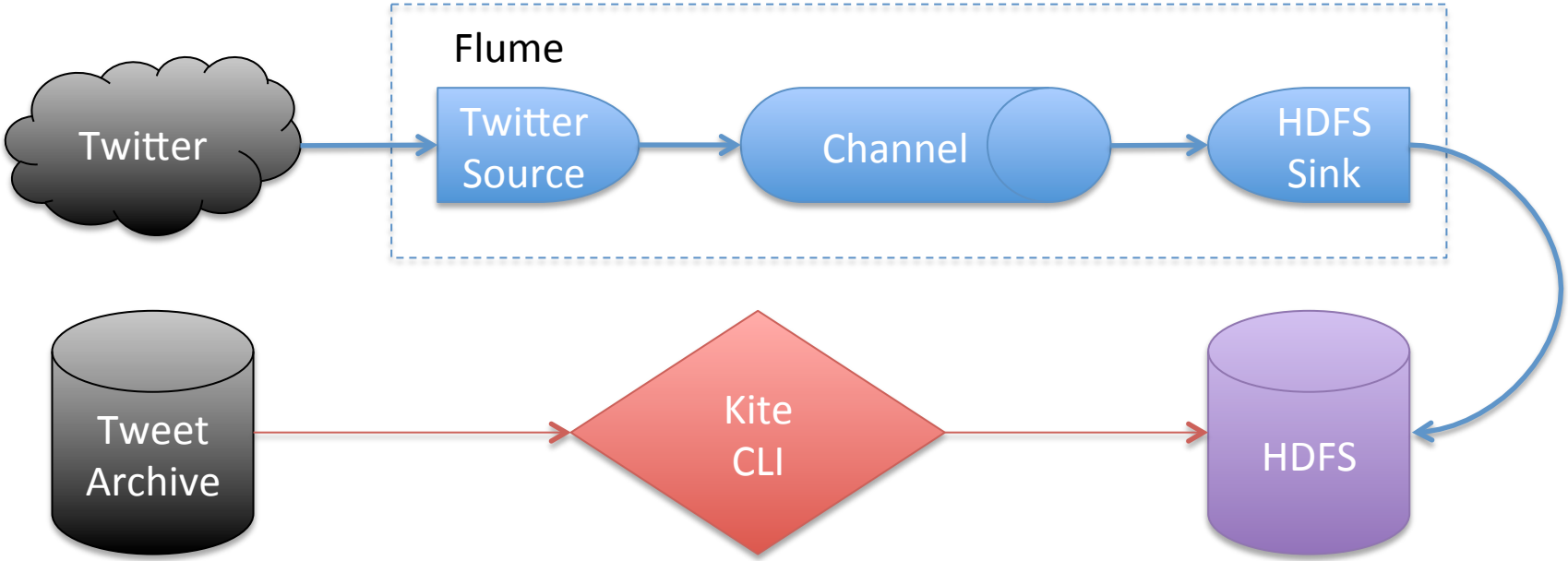
(Mostly) traditional solution



(Mostly) traditional solution



(Mostly) traditional solution



Demo

Drawbacks

Drawbacks

- Two ingest systems

Drawbacks

- Two ingest systems
  - Separate monitoring

Drawbacks

- Two ingest systems
  - Separate monitoring
  - Separate failure modes

Drawbacks

- Two ingest systems
  - Distinct monitoring
  - Distinct failure modes
  - Distinct debugging

Drawbacks

- Two ingest systems
  - Distinct monitoring
  - Distinct failure modes
  - Distinct debugging
- Manual integration

Drawbacks

- Two ingest systems
  - Distinct monitoring
  - Distinct failure modes
  - Distinct debugging
- Manual integration
  - Kite CLI with cron

Enter Apache NiFi

Enter Apache NiFi

Bounded context

Bounded context

- You control all the parts

Bounded context

- You control all the parts
  - Protocols

Bounded context

- You control all the parts
  - Protocols
  - Schemas

Bounded context

- You control all the parts
  - Protocols
  - Schemas
  - Formats

Bounded context

- You control all the parts
  - Protocols
  - Schemas
  - Formats
  - Changes

NiFi strengths

NiFi strengths

- Generic data flow

NiFi strengths

- Generic data flow
- Built-in editor/monitor

NiFi strengths

- Generic data flow
- Built-in editor/monitor
- Varying object size

NiFi strengths

- Generic data flow
- Built-in editor/monitor
- Varying object size
- Traditional sources

NiFi strengths

- Generic data flow
- Built-in editor/monitor
- Varying object size
- Traditional sources
  - Files, FTP, SFTP, HTTP, etc.

NiFi limitations

NiFi limitations

- Streaming sources

NiFi limitations

- Streaming sources
  - ListenHttp

NiFi limitations

- Streaming sources
  - ListenHttp
  - ListenUdp

NiFi limitations

- Streaming sources
  - ListenHttp
  - ListenUdp
  - GetKafka

Enter Apache Flume

Enter Apache Flume

- Streaming from the start

Enter Apache Flume

- Streaming from the start
- Rich set of sources/sinks

Enter Apache Flume

- Streaming from the start
- Rich set of sources/sinks
  - Apache Avro, Apache Thrift, Twitter, NetCat, Syslog

Enter Apache Flume

- Streaming from the start
- Rich set of sources/sinks
  - Apache Avro, Apache Thrift, Twitter, NetCat, Syslog
  - HDFS, IRC, Hbase, Kite

Cake

Cake

- NiFi combines ingest contexts

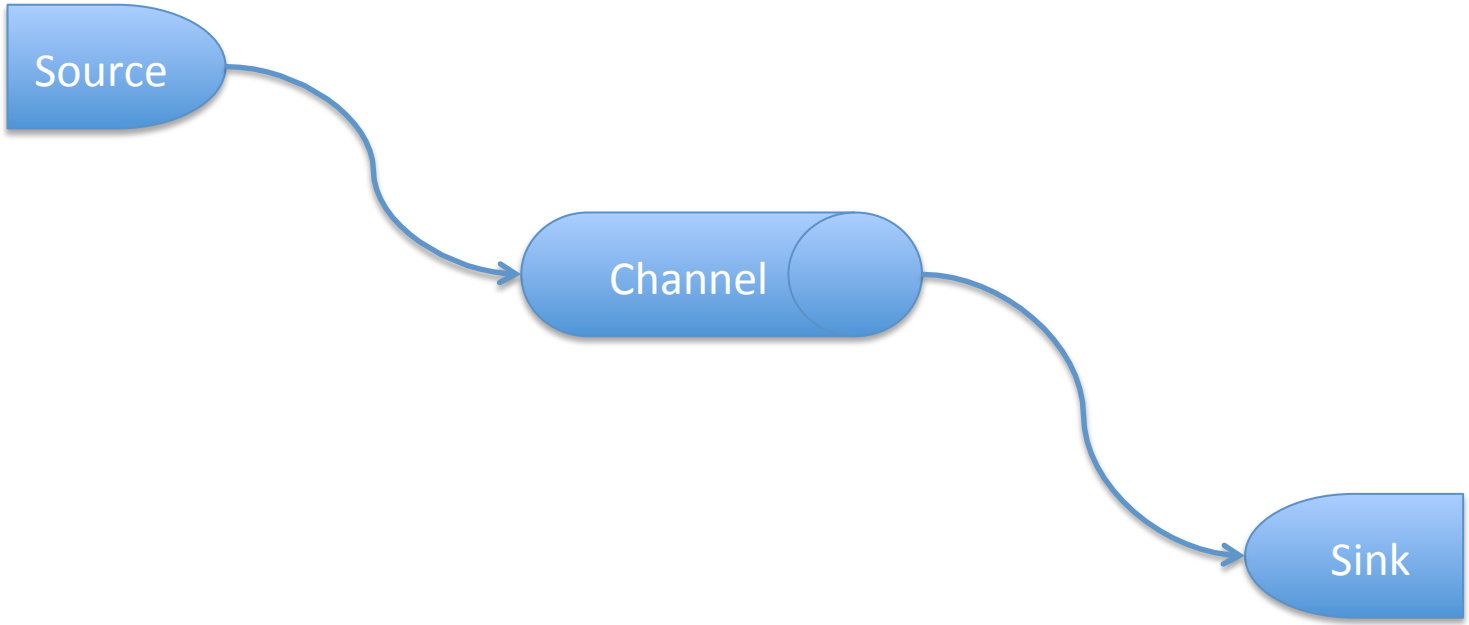
Cake

- NiFi combines ingest contexts
- Flume requires static stream configuration

Cake

- NiFi combines ingest contexts
- Flume requires static stream configuration
- I want both

Flume architecture



Flume ➔ NiFi

Flume ➔ NiFi

- Source/Sink

Flume ➔ NiFi

- Source/Sink
- Event

Flume ➔ NiFi

- Source/Sink
- Event
- Channel

Flume ➔ NiFi

- Source/Sink ➔ Processor

Flume ➔ NiFi

- Source/Sink ➔ Processor
- Event ➔ FlowFile

Flume ➔ NiFi

- Source/Sink ➔ Processor
- Event ➔ FlowFile
- Channel ➔ FlowFile Queue/Connection

Solution

Solution

- NiFi processors to run Flume sources/sinks

Solution

- NiFi processors to run Flume sources/sinks
- Prototype

Solution

- NiFi processors to run Flume sources/sinks
- Prototype
- http://bit.ly/flume-processors

Demo

Summary

Summary

- Integrating data is challenging

Summary

- Integrating data is challenging
- Managing multiple systems adds complexity

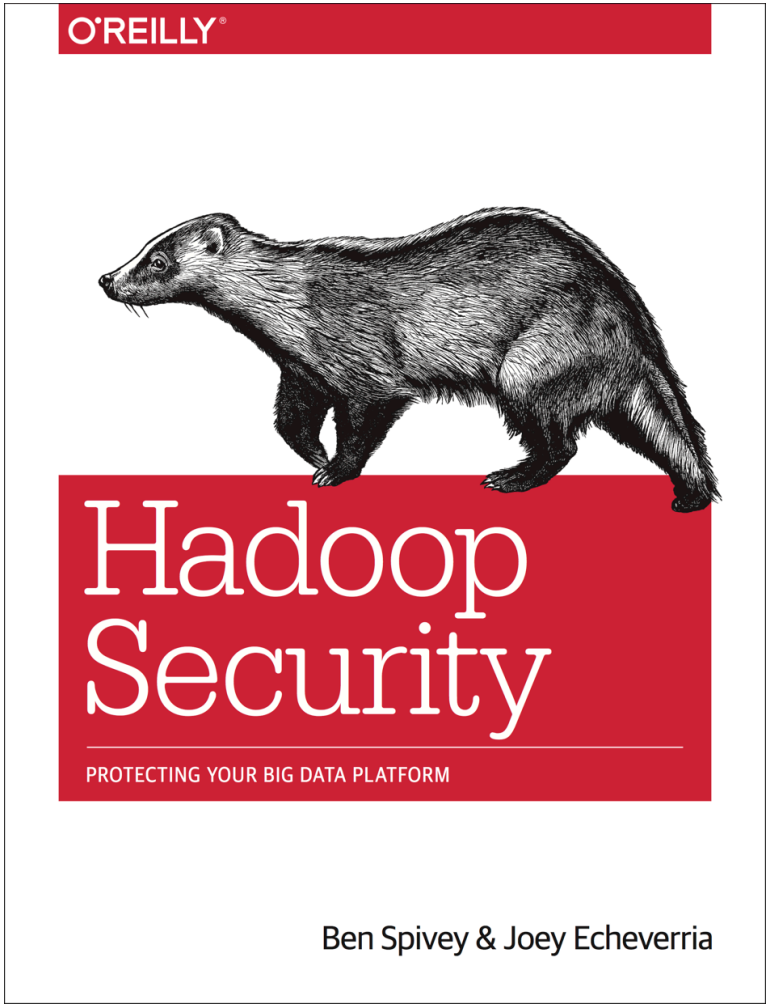
Summary

- Integrating data is challenging
- Managing multiple systems adds complexity
- NiFi supports generic data flow

Summary

- Integrating data is challenging
- Managing multiple systems adds complexity
- NiFi supports generic data flow
- NiFi can be extended to solve new use cases

Joey Echeverria
joey@scalingdata.com
@fwiffo



Big Data Meets IT Ops

SCALINGDATA