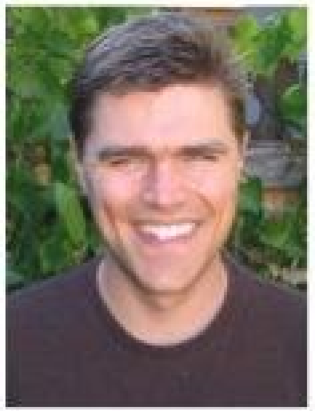
Hortonworks

Discover HDP 2.1

Apache Storm for Stream Data Processing in Hadoop

Hortonworks. We do Hadoop.

Speakers



Justin Sears

Hortonworks Product Marketing Manager

Himanshu Bari

Hortonworks Senior Product Manager & PM for Apache Storm & Apache Falcon in Hortonworks Data Platform

Taylor Goetz

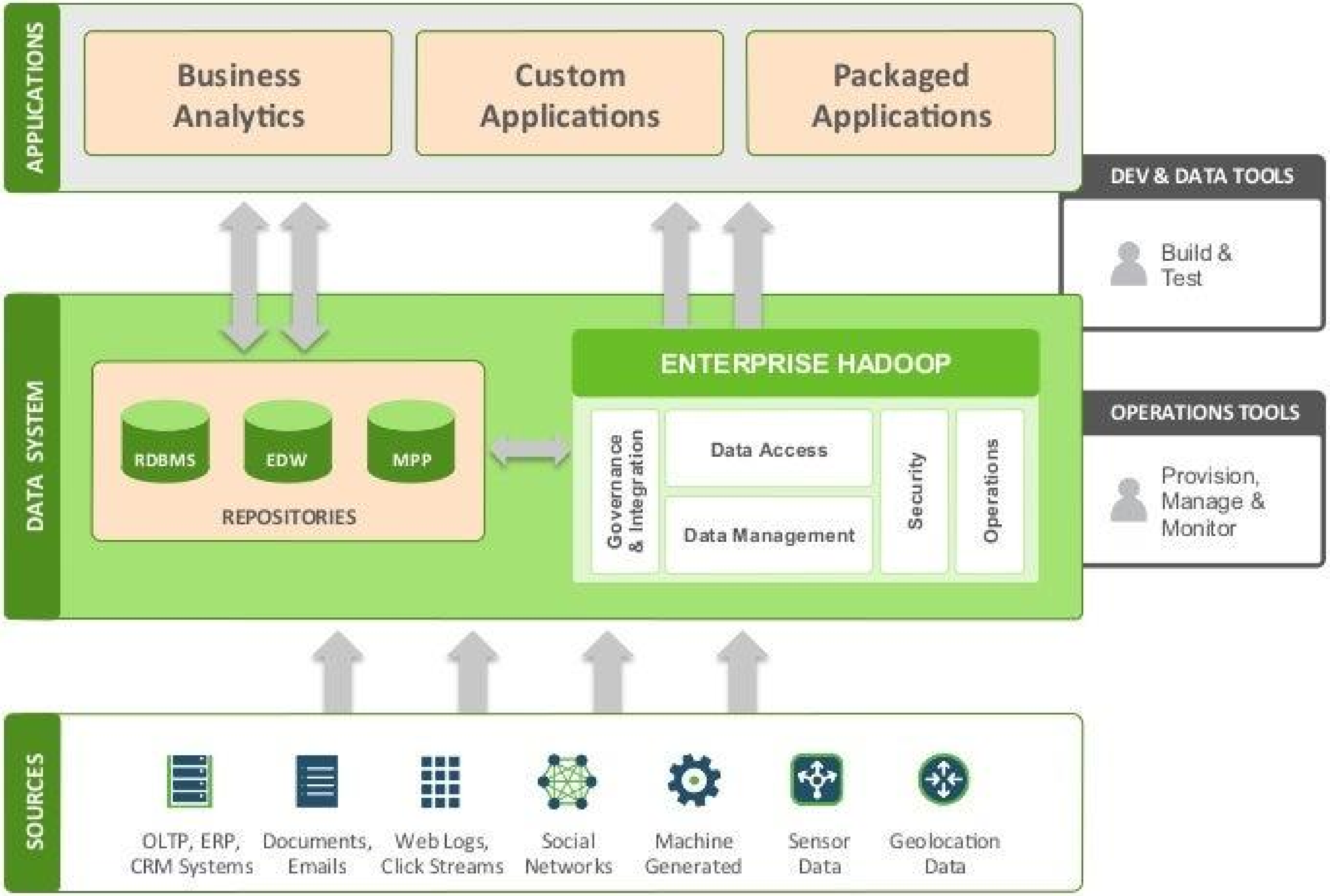
Hortonworks Engineer & Committer for Apache Storm, with deep expertise in master data management

Hortonworks

Agenda

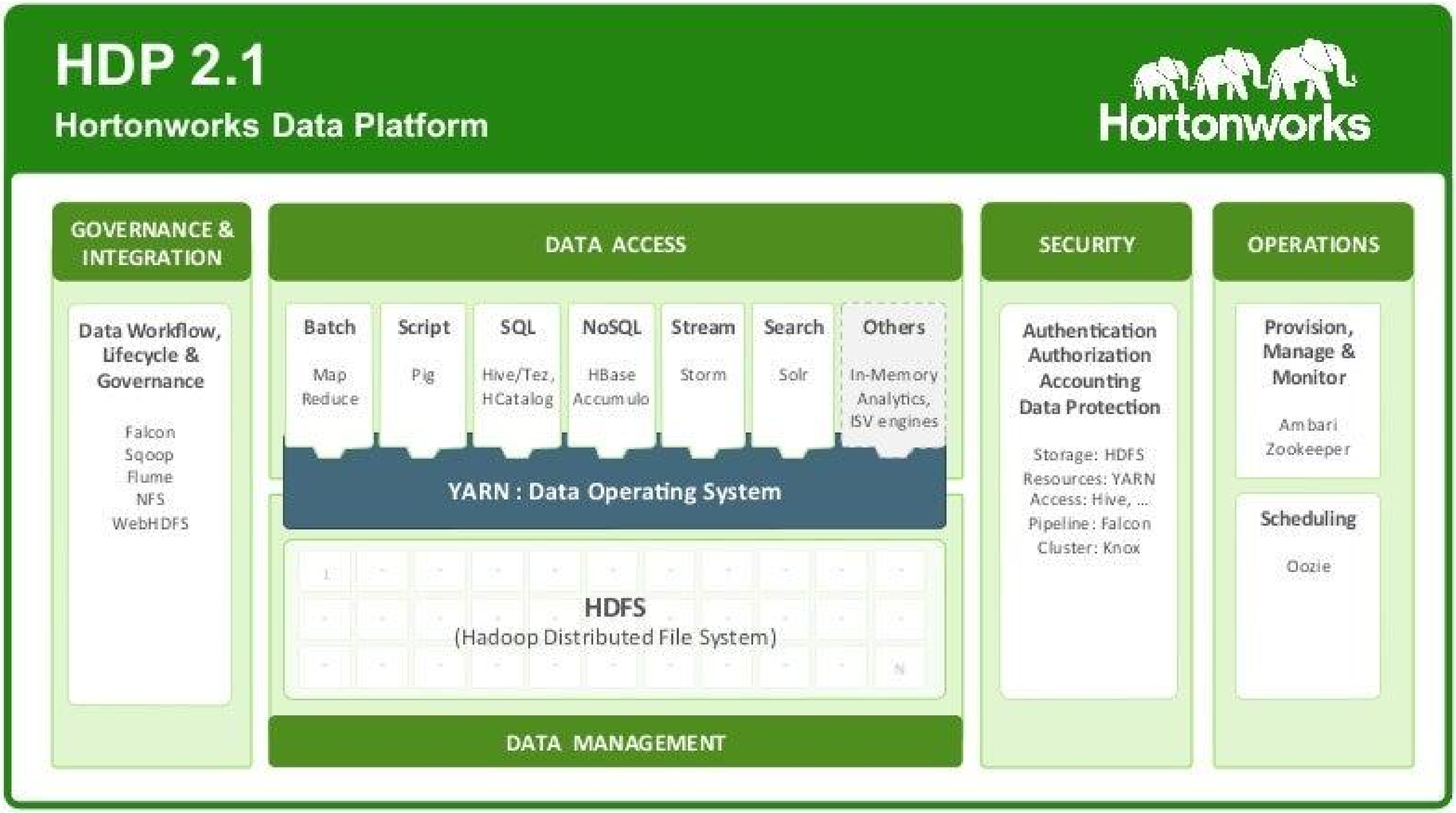
- **Why Stream Processing?**
- **Overview of Apache Storm**
- **Q & A**

A Modern Data Architecture



Hortonworks

HDP 2.1: Enterprise Hadoop



HDP 2.1

Hortonworks Data Platform

Hortonworks

GOVERNANCE & INTEGRATION

Data Workflow, Lifecycle & Governance

Falcon
Sqoop
Flume
NFS
WebHDFS

DATA ACCESS

Batch: Map Reduce

Script: Pig

SQL: Hive/Tez, HCatalog

NoSQL: HBase Accumulo

Stream: Storm

Search: Solr

Others: In-Memory Analytics, ISV engines

YARN : Data Operating System

HDFS (Hadoop Distributed File System)

DATA MANAGEMENT

SECURITY

Authentication
Authorization
Accounting
Data Protection

Storage: HDFS
Resources: YARN
Access: Hive, ...
Pipeline: Falcon
Cluster: Knox

OPERATIONS

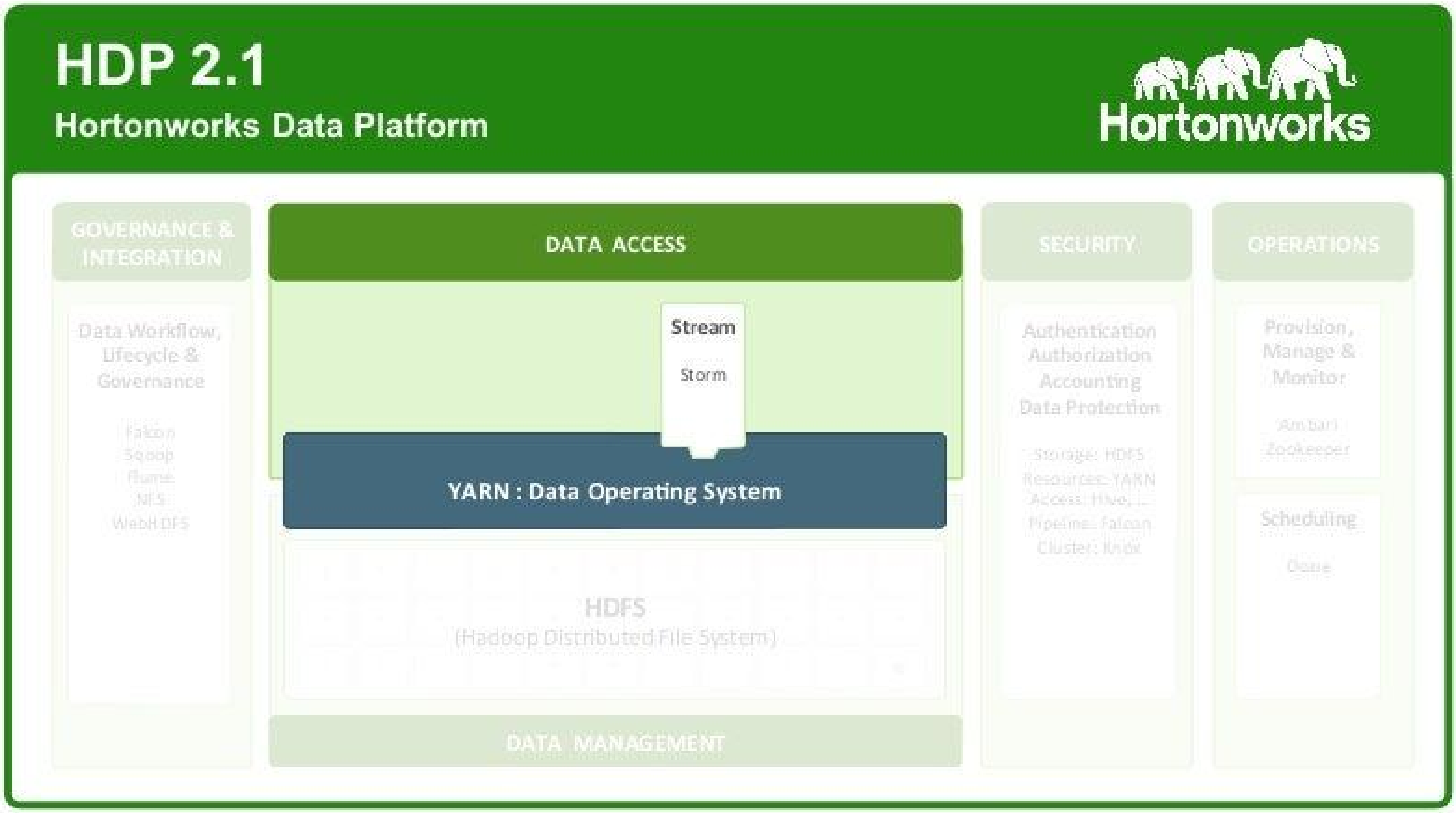
Provision, Manage & Monitor

Ambari
Zookeeper

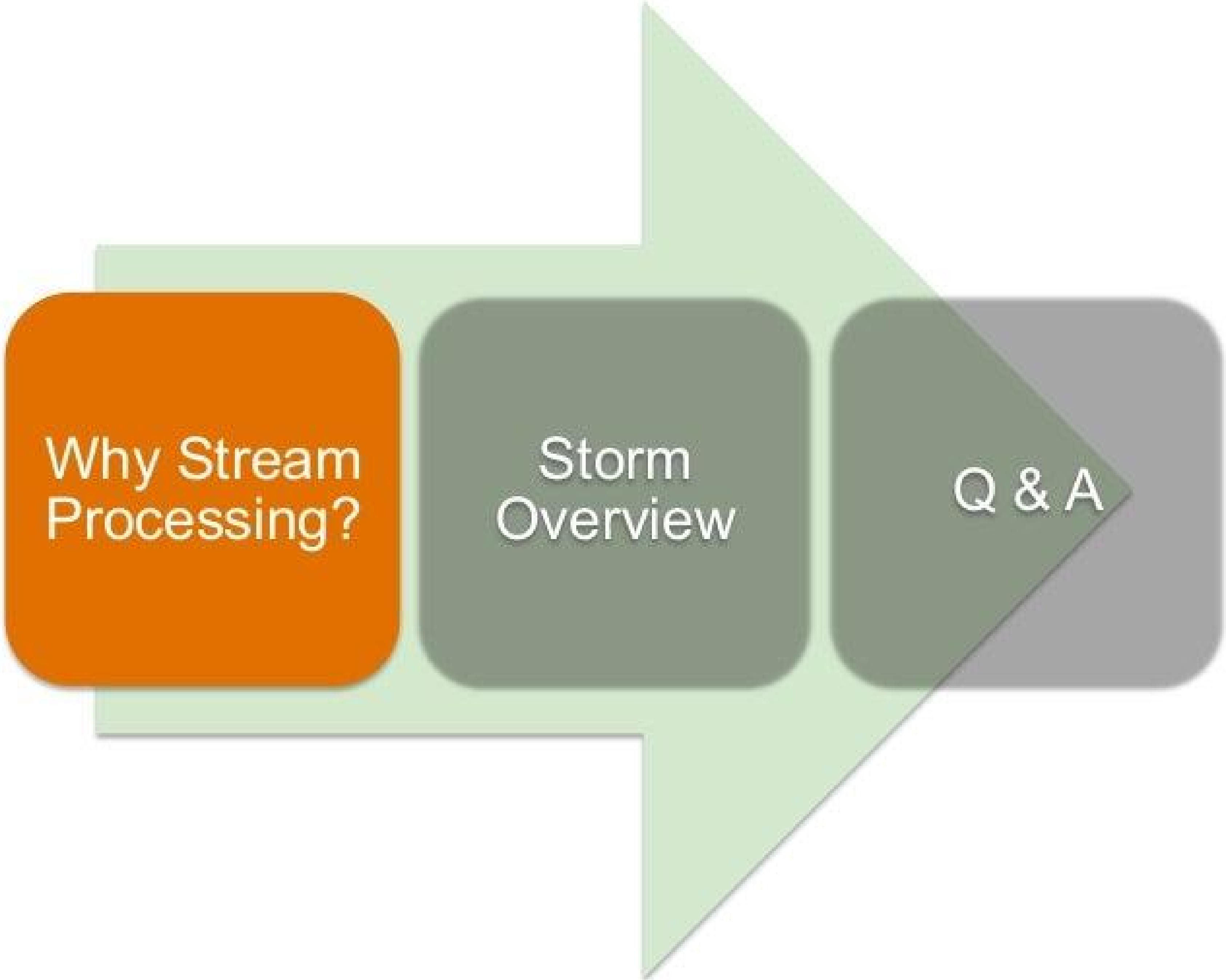
Scheduling

Oozie

HDP 2.1: Enterprise Hadoop



Agenda



Why Stream Processing?

Storm Overview

Q & A

Why Stream Processing IN Hadoop?

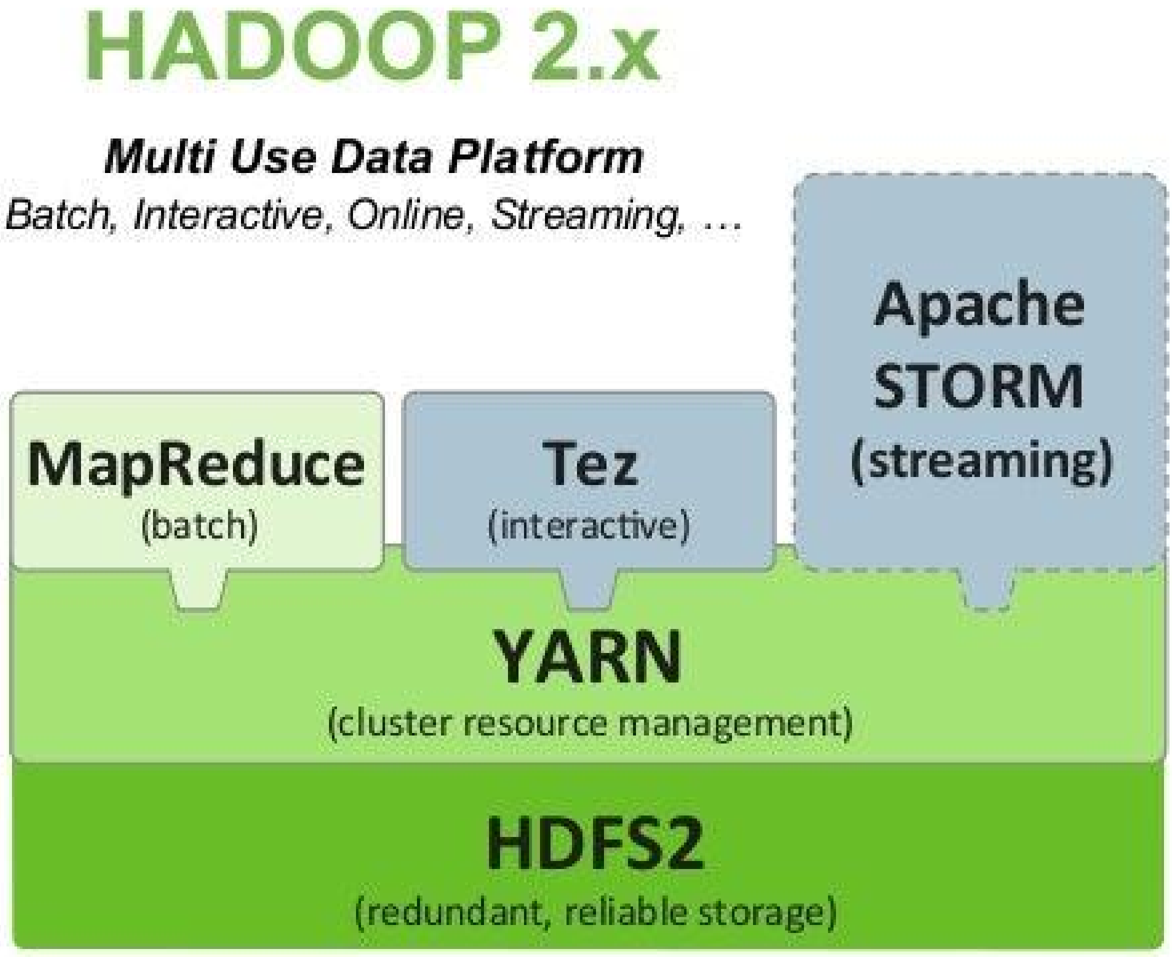
Stream processing has emerged as a key use case

What is the need?

- Exponential rise in real-time data
- Ability to process real-time data opens new business opportunities

Why Now?

- Economics of Open source software & commodity hardware
- YARN allows multiple computing paradigms to co-exist in the data lake


HADOOP 2.x
Multi Use Data Platform
Batch, Interactive, Online, Streaming, …
MapReduce
(batch)
Tez
(interactive)
Apache STORM
(streaming)
YARN
(cluster resource management)
HDFS2
(redundant, reliable storage)

Hortonworks

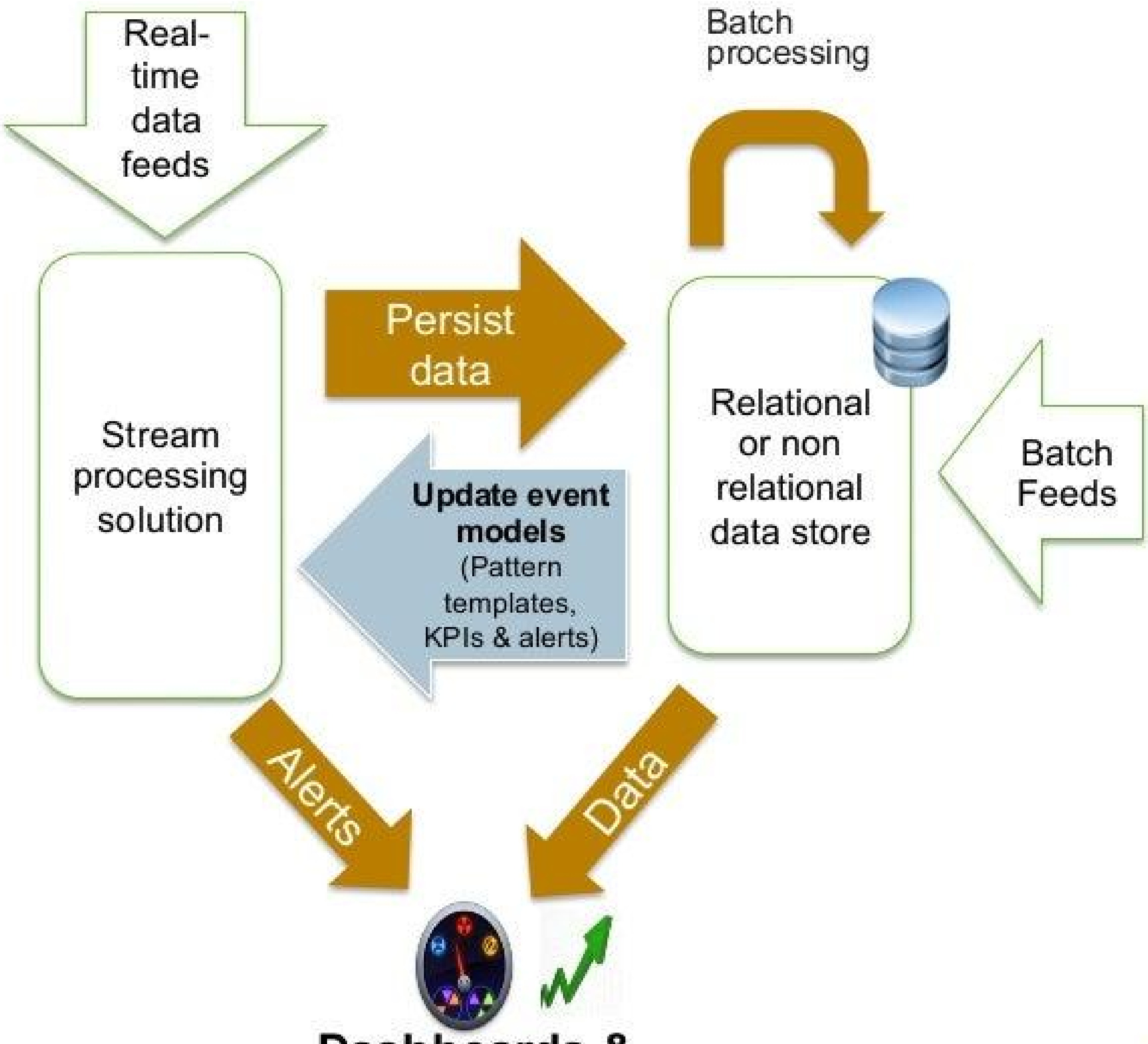
Why Apache Storm?

Open source real-time event stream processing platform that provides fixed, continuous & low latency processing for very high frequency streaming data

Highly scalable	• Horizontally scalable like Hadoop • Eg: 10 node cluster can process 1M tuples per second per node
Fault-tolerant	• Automatically reassigns tasks on failed nodes
Guarantees processing	• Supports at least once & exactly once processing semantics
Language agnostic	• Processing logic can be defined in any language
Apache project	• Brand, governance & a large active community

Hortonworks

Typical Stream Processing Flow


Real-time data feeds
Batch processing
Stream processing solution
Persist data
Update event models
(Pattern templates, KPIs & alerts)
Relational or non relational data store
Batch Feeds
Alerts
Data

Dashboards & Applications

Who is Using Storm today?

E-COMMERCE

淘宝网
Taobao.com

支付宝
Alipay.com

QUICKLIZARD
Real Time Pricing

wego

Healthcare

Cerner

IDEXX
LABORATORIES

Source: Storm-project.net

TELCO

Aeris
COMMUNICATIONS

2lemetry

FINANCE

PREMISE

AD- TECH

OOYALA

rubicon
PROJECT

rocketfuel
Artificial intelligence. Real results.

SOCIAL MEDIA

UMENG

KLOUT

AND MANY OTHERS…

TIME WARNER
CABLE

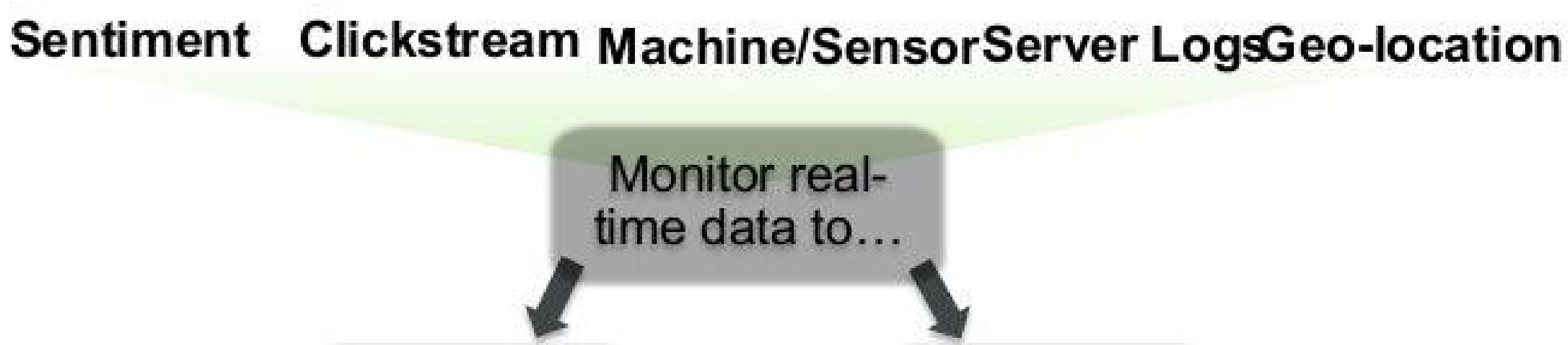
spider.io + Google

NaviSite

Y!

The Ladders

Patterns Driving Most Streaming Use Cases

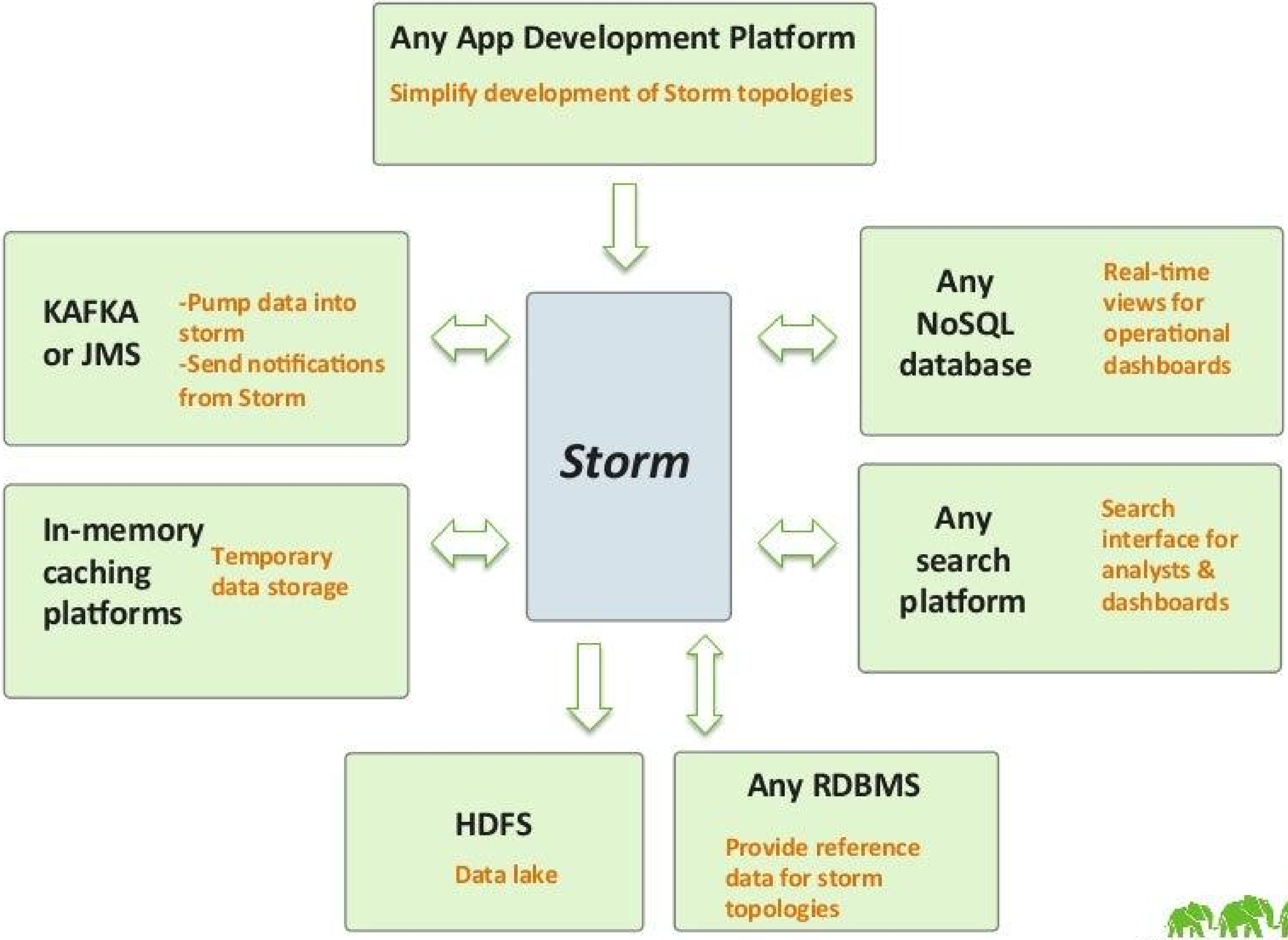


Sentiment Clickstream Machine/Sensor Server Logs Geo-location

Monitor real-time data to…

	Prevent	Optimize
Finance	- Securities Fraud - Compliance violations	- Order routing - Pricing
Telco	- Security breaches - Network Outages	- Bandwidth allocation - Customer service
Retail	----	- Offers - Pricing
Manufacturing	- Machine failures	- Supply chain
Transportation	- Driver & fleet issues	- Routes - Pricing
Web	- Application failures - Operational issues	- Site content

A Key Storm Benefit: Flexibility



Any App Development Platform

Simplify development of Storm topologies

KAFKA or JMS

- -Pump data into storm
- -Send notifications from Storm

In-memory caching platforms

Temporary data storage

Storm

Any NoSQL database

Real-time views for operational dashboards

Any search platform

Search interface for analysts & dashboards

HDFS

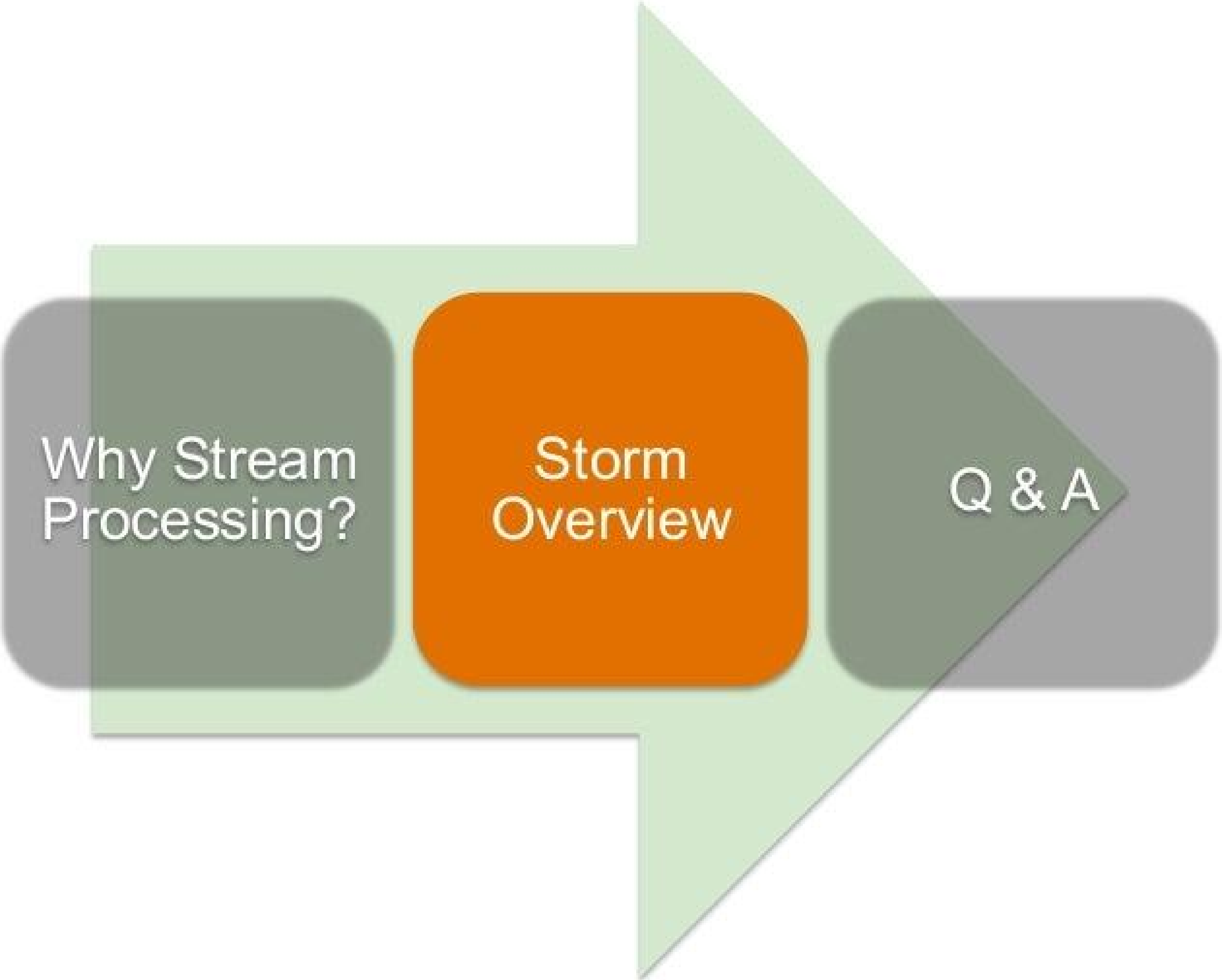
Data lake

Any RDBMS

Provide reference data for storm topologies

Hortonworks

Agenda

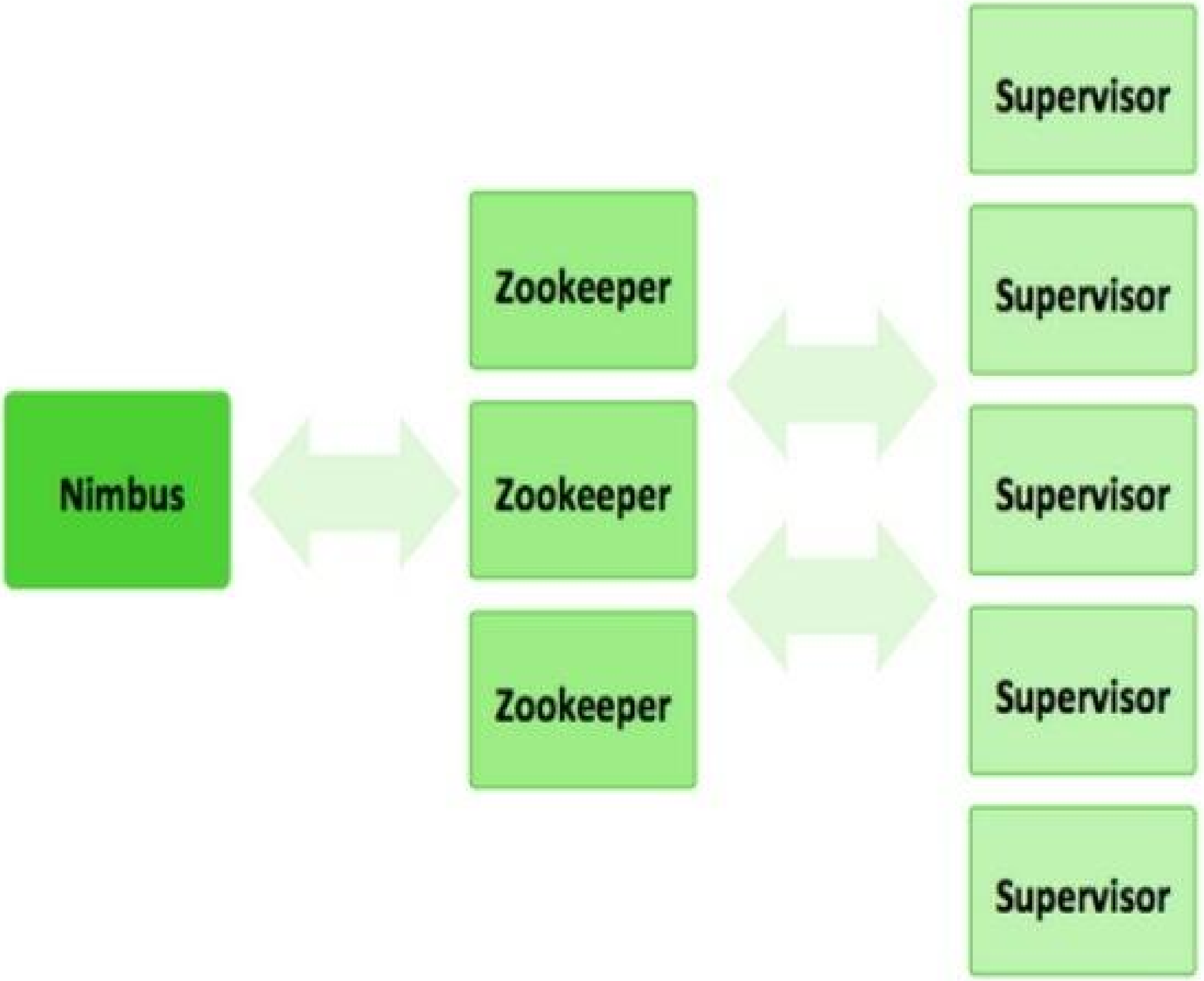


Why Stream Processing?

Storm Overview

Q & A

Storm Architecture



Nimbus(Management server)

- Similar to job tracker
- Distributes code around cluster
- Assigns tasks
- Handles failures

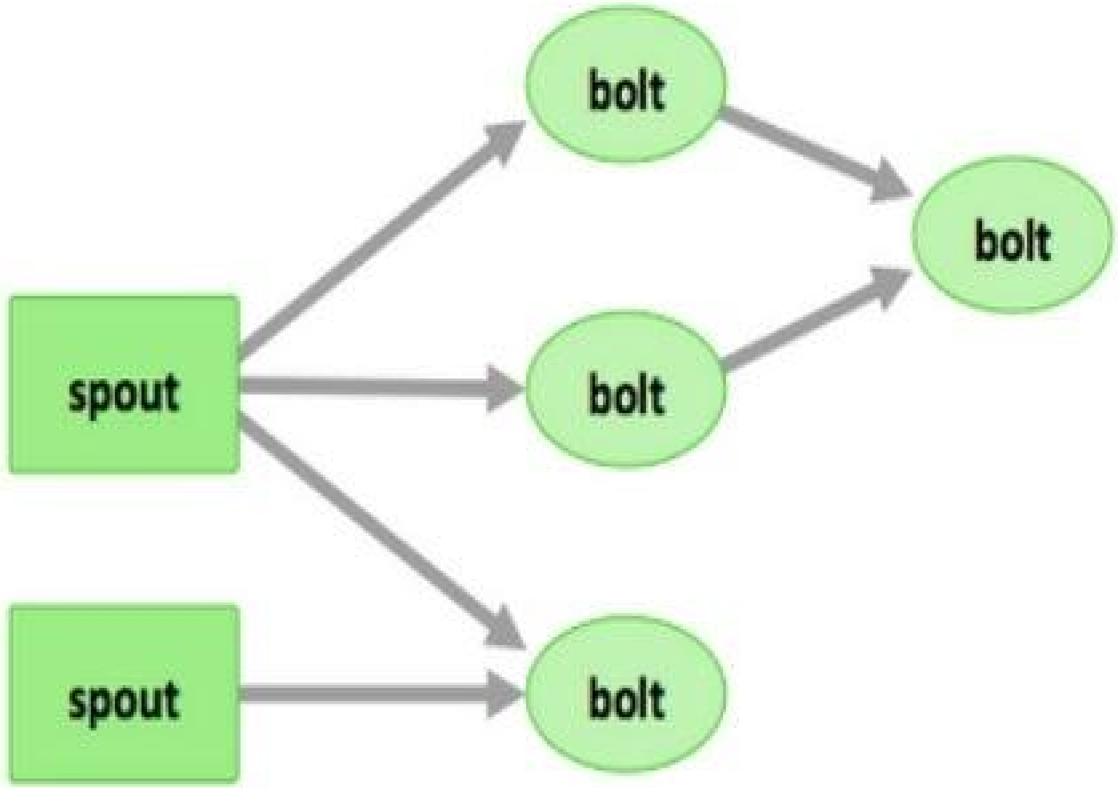
Supervisor(Worker nodes):

- Similar to task tracker
- Run bolts and spouts as ‘tasks’

Zookeper:

- Cluster co-ordination
- Nimbus HA
- Stores cluster metrics
- Consumption related metadata for Trident topologies

Basic Storm Concepts



Tuple: Most fundamental data structure and is a named list of values that can be of any datatype

Streams: Groups of tuples

Spouts: Generate streams.

Bolts: Contain data processing, persistence and alerting logic. Can also emit tuples for downstream bolts

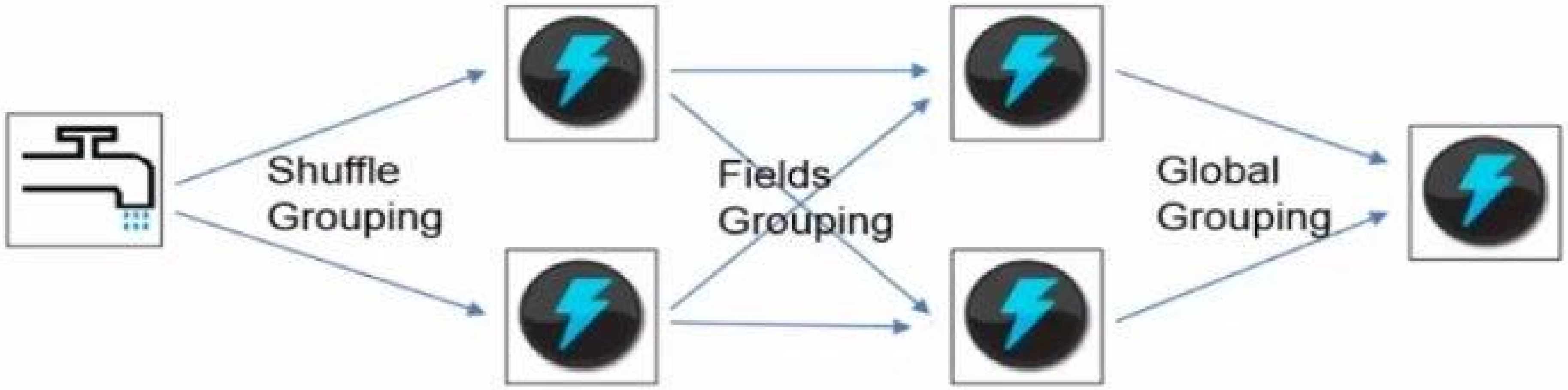
Tuple Tree: First tuple and all the tuples that were emitted by the bolts that processed it

Topology: Group of spouts and bolts wired together into a workflow

Hortonworks

Storm Topology

Get Tweet → Find Hashtags → Count Hashtags → Report Findings



Kafka Spout
"reader"

Bolt
"normalizer"

HashTagNormalizer.java

Removes non-alphanumeric characters, extracts hashtag values and emits them.

Bolt
"enumerator"

HashTagEnumerator.java

Keeps track of how many instances of each hashtag have occurred.

Bolt
"reporter"

ResultsReporter.java

Regularly creates a report and uploads it to Amazon S3.

What is Trident?

Provides exactly once processing semantics in Storm using real-time batch processing

Core concept: process a group of tuples as a ‘batch’ rather than process tuple at a time like core Storm

Provides a ‘higher level abstraction’ for Storm operations like what cascading does for MapReduce

All Trident topologies are automatically converted into core Storm concepts (Spouts & Bolts)

Key Trident Concepts

Spouts and Tuples

- Remain the same as core Storm topologies

Transactions

- Way of tagging tuples together so they can be processed with exactly once semantics

Batches

- All tuples tied to the same transactionID form a batch

Partitions

- Segments of a batch that are guaranteed to process their tuples in order.
- Multiple partitions in a given batch can/will be processed in parallel

Streams

- Series of batches form a stream (just like series of tuples form a stream in core Storm)

Operations

- The higher level abstraction for processing tuples are called 'operations'
- Multiple inbuilt operations available for joins, grouping, aggregations & filtering

Apache Storm and Apache Ambari

Apache Ambari is now integrated with Apache Storm

- Install Storm with Ambari
- Monitor Storm services with Ambari