Storm

Induction

1. Storm Overview
2. Storm Cluster Configuration;
3. Exercise 1
4. Topology Overview;
5. Topology Insights.
6. Exercise 2

Storm

1. What’s Storm;
2. Concepts;
3. Features;
4. Who uses it;
5. Input Platform.

What is Storm

A distributed real-time computation engine for processing large volumes of data near realtime.

It makes it easy to reliably process unbounded streams of data, doing for realtime processing what Hadoop did for batch processing.

What is Storm

Uses Cases:

- Realtime Analytics;
- Machine Learning;
- Continuous Computation;
- Distributed RPC;
- ETL;

What is Storm

Throughput (tuples/sec)

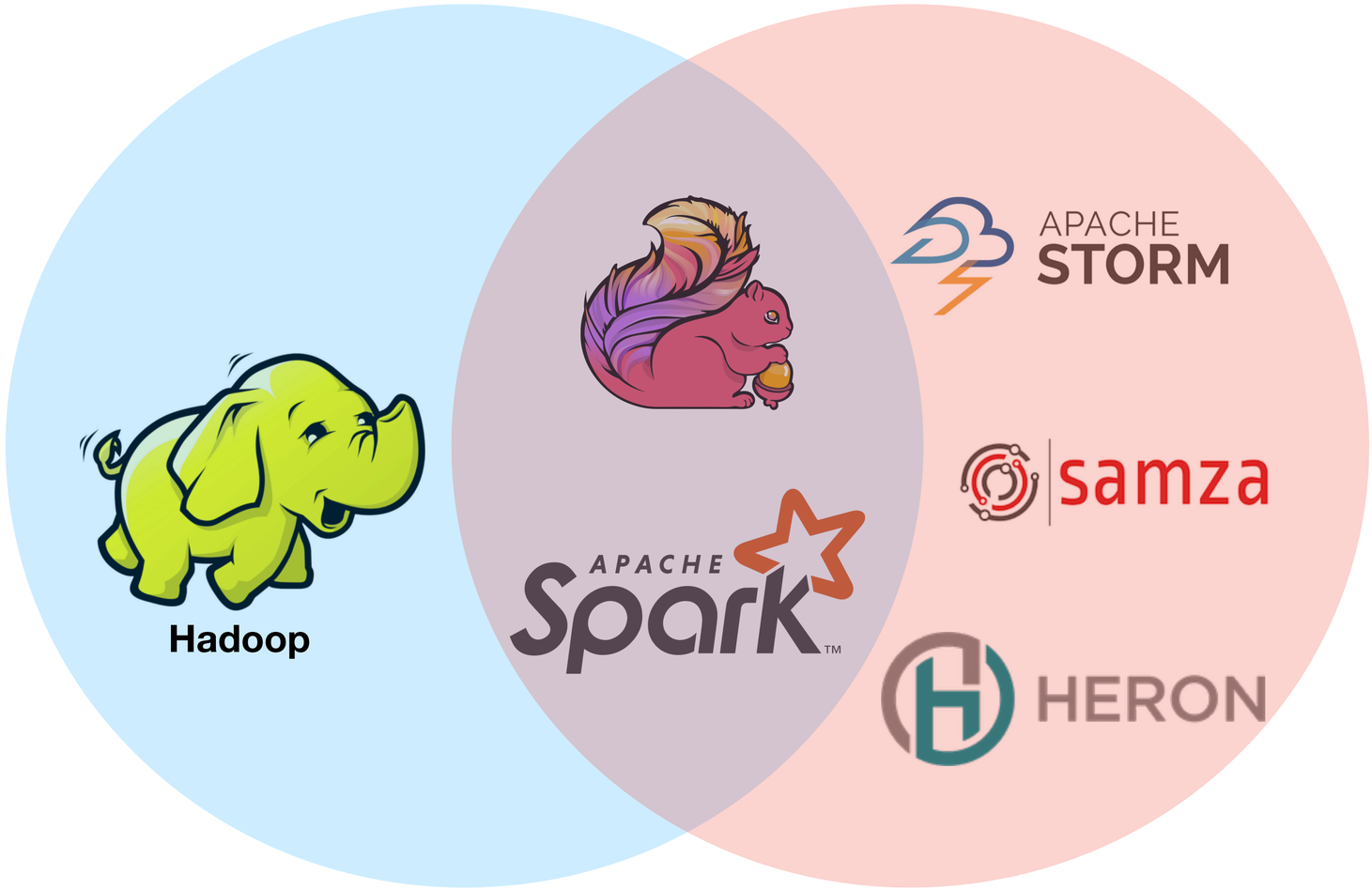
Storm version	Spout Emit	MSGing IntraWorker	MSGing InterWorker 1	MSGing InterWorker 2	MSGing InterHost 1	MSGing InterHost 2
v0.9.0.1	108,000	87,000	48,000	43,000	48,000	50,000
v1.0.1	3,200,000	233,000	287,000	292,000	316,000	303,000

Latency (milliseconds)

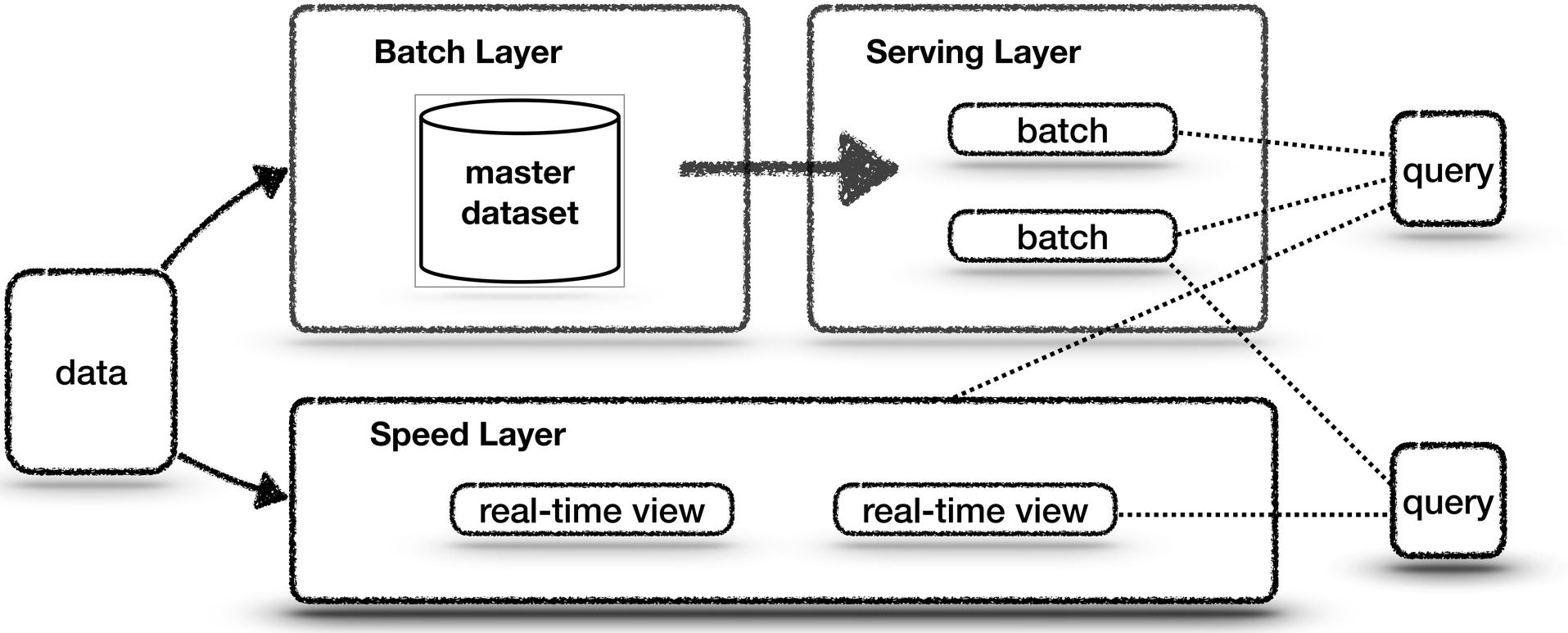
Storm ver	MSGing IntraWorker	MSGing InterWorker 1	MSGing InterWorker 2	MSGing InterHost 1	MSGing InterHost 2
v1.0.1	3	8	9	13	7
v0.9.0.1	16	170	116	845	1,700

"Microbenchmarking Apache Storm 1.0 Performance."

What is Storm

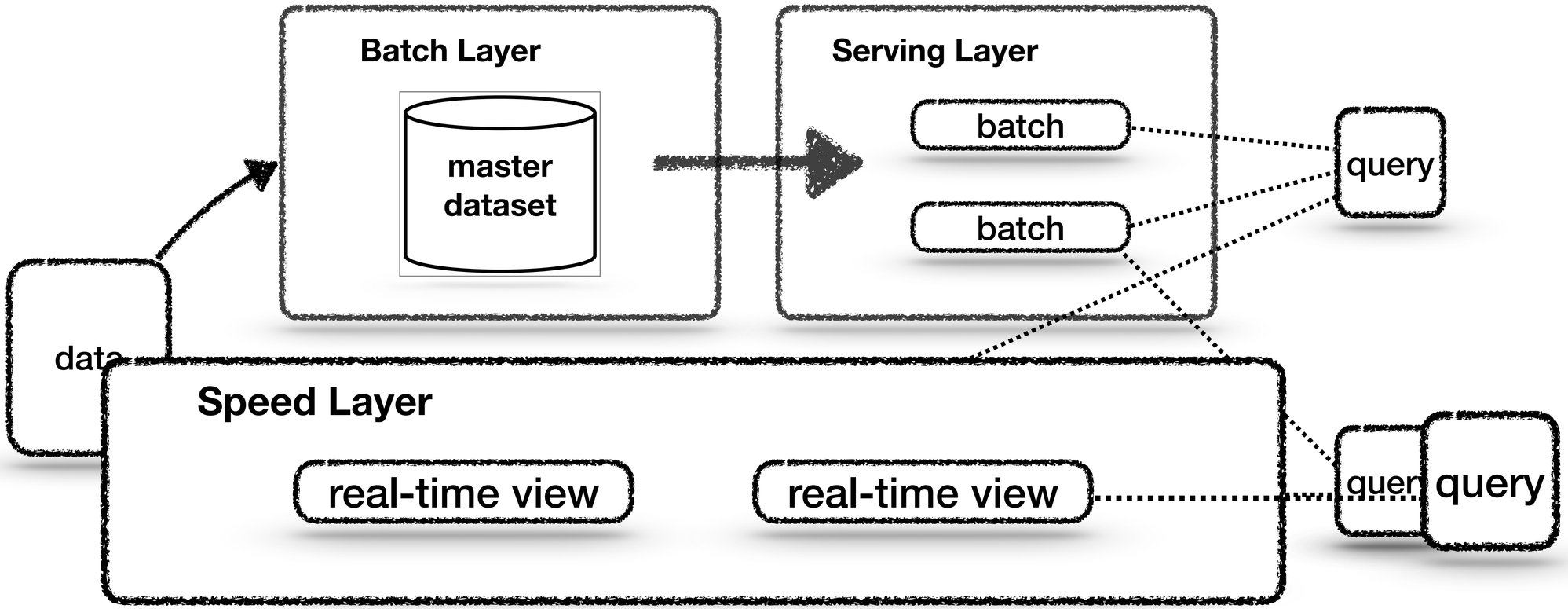


What is Storm



Lambda Architecture

What is Storm



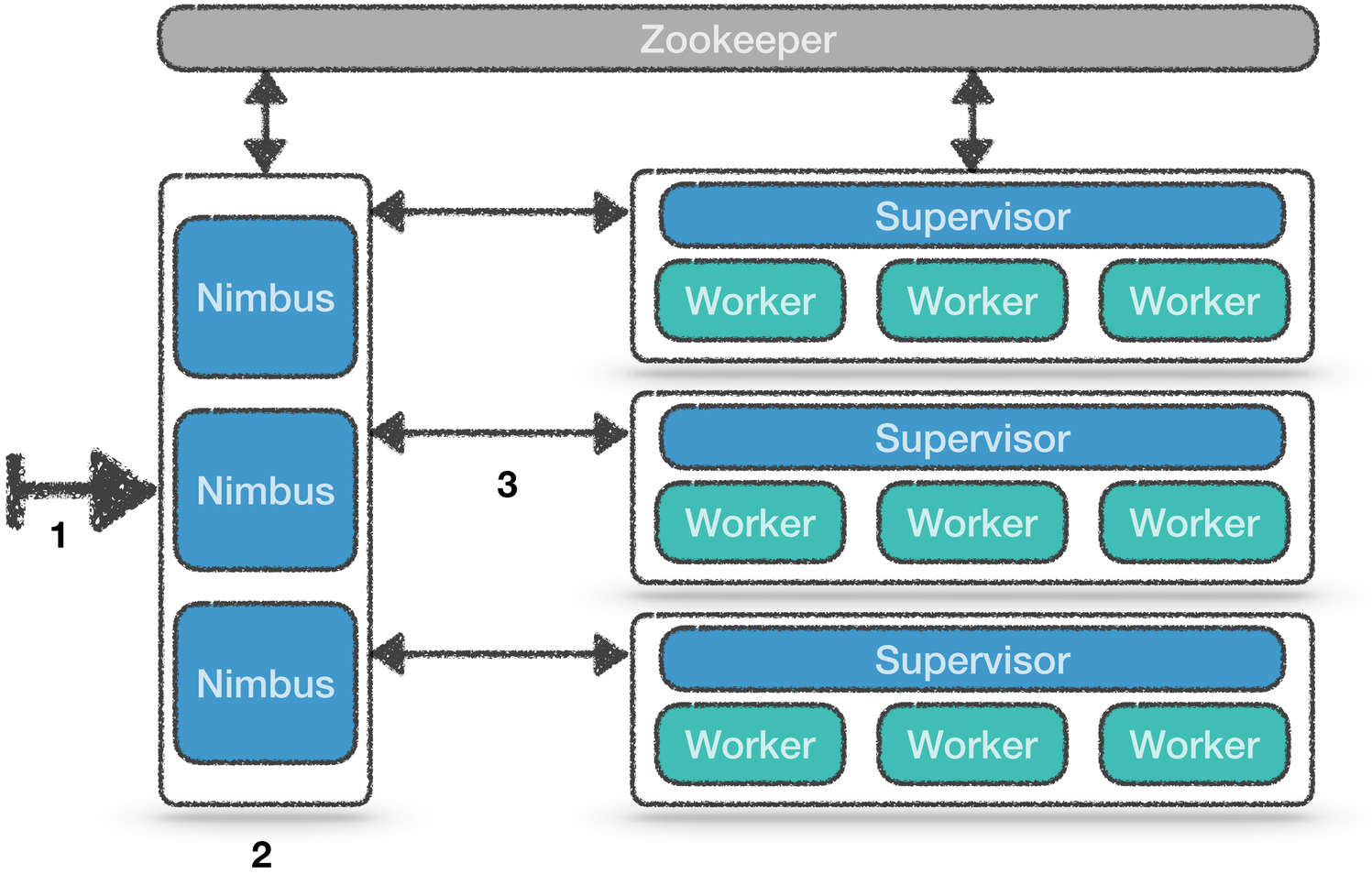
Lambda Architecture

Concepts

Key Concepts:

- Nimbus - **Distributes** apps across supervisor nodes, **monitors**, and **restarts** them whenever issues arise;
- Supervisor - **Manages** the life cycle of **worker** processes assigned to a single **node**;
- Worker - A JVM that executes a subset of all the **tasks** tat compose a **topology**;
- Topology - A computation **graph** that **defines** how **tasks** will be executed inside the **cluster**;
- Bolt - Processing **tasks** used for filtering, functions, aggregations, joins, accessing to DBs, etc.
- Zookeeper - Distributed **coordinator** for the entire **cluster**;

Concepts



Storm Cluster

1. The topology's Jar and Config files are uploaded to Nimbus;
2. Nimbus builds the topology and serializes it;
3. The serialized topology is sent to the available workers;

Features

- Easy to Use;
- Programming Language Agnostic;
- Highly Scalable;
- Resilient;
- Reliable;
- Message Driven;

Features

How Resilient:

- When all Nimbus Fail?

Topologies continue to work, but reassignment feature is lost;

- When Supervisor Fails?

Topologies continue to work, but assignment feature is lost;

- When a worker fails?

Tasks are reassigned to other workers;

Features

Guarantee Message Processing:

- **At-most-once** - When its guaranteed that no single tuple ever gets processed more than once;
- **At-least-once** - When its guaranteed that every single tuple must be processed successfully at least once;
- **Exactly-once** - When its guarantee that every tuple is processed successfully;

Features

Pitfalls:

- **Exactly-once** processing guarantee can only be achieved by using Trident abstraction on top of Storm. It's an afterthought.

Features

Message Driven:

- Uses internally message queues backed by LMAX Disruptor to send data between tasks;
- Has integration with Apache Kafka (distributed messaging system that provides a strong durability and fault tolerance guarantees);
- This creates a boundary between systems that ensures loose coupling and isolation.

Who uses it


WebMD


Flipboard


淘宝网
Taobao.com


Alibaba.com


yelp


KLOUT


loggly


infochimps
A CSC BIG DATA BUSINESS


parc
A Xerox Company


YAHOO!
JAPAN


rubicon
PROJECT


The
Weather
Channel


GROUPON


OOYALA


Spotify


Cerner


Baidu 百度


FullContact


YAHOO!


twitter

Who uses it

Flutter™

“The integration of different feed data providers drove PPB to create a streams processing platform capable of ingesting, transforming, and pushing high volumes of data into the PPB ecosystem.

Today, Product Catalog Input Platform_is one of the key pieces on the event and market management flow, enabling Flutter to increase the number of offering events to their Exchange and Sportsbook customers.”

Catalog Input Platform