# Cloud Computing and Big Data

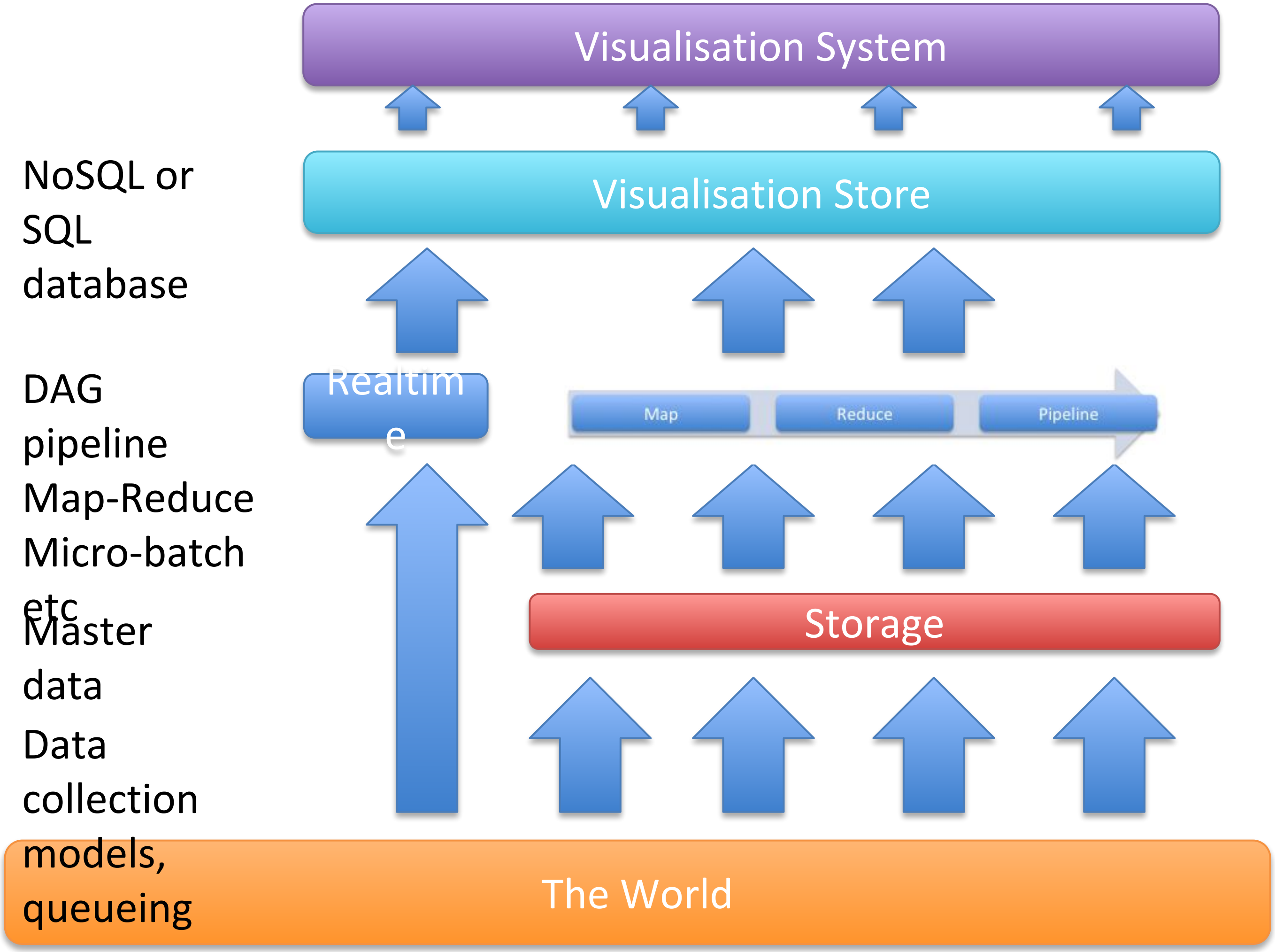
# Big Data
# Pulling it all together

Oxford University
Software Engineering
Programme
July 2019

# Contents

- Understanding the bigger picture
- What are the different components
- Message queueing and collection systems
- Map-Reduce and DAG systems
- Realtime Systems
- Fast databases for speed
- Visualisation and Dashboards

# The big picture



NoSQL or SQL database

DAG pipeline Map-Reduce Micro-batch etc

Master data

Data collection models, queueing

# The big picture

- You have *immutable* master data
- You create a set of processes to:
  - Collect that data
  - Store master data
  - Process data
  - Visualise and present
- Some of those processes act on batch and others on real-time data

# How to choose the components?

- Two main approaches:
  - Best of breed
    - Choose the best available component in each space
  - Stack
    - Choose a curated stack that a team or organization is providing/selling/supporting

# Approach

- Minimise the pain
  - Choose what you need when you need it
  - Don’t over engineer

# How do I ingest data?

- File transfer
- Live stream
  - Sockets
  - Syslog
  - Messaging system
- From existing databases

# How do I store data?

- HDFS
- Cassandra File System
- NoSQL database only
  - Mongo / HBase / Cassandra
- zFS / GlusterFS / NFS etc

# How do I process data?

- Simple Map Reduce
- Hive / Pig
- DAG
- Pipeline
- etc

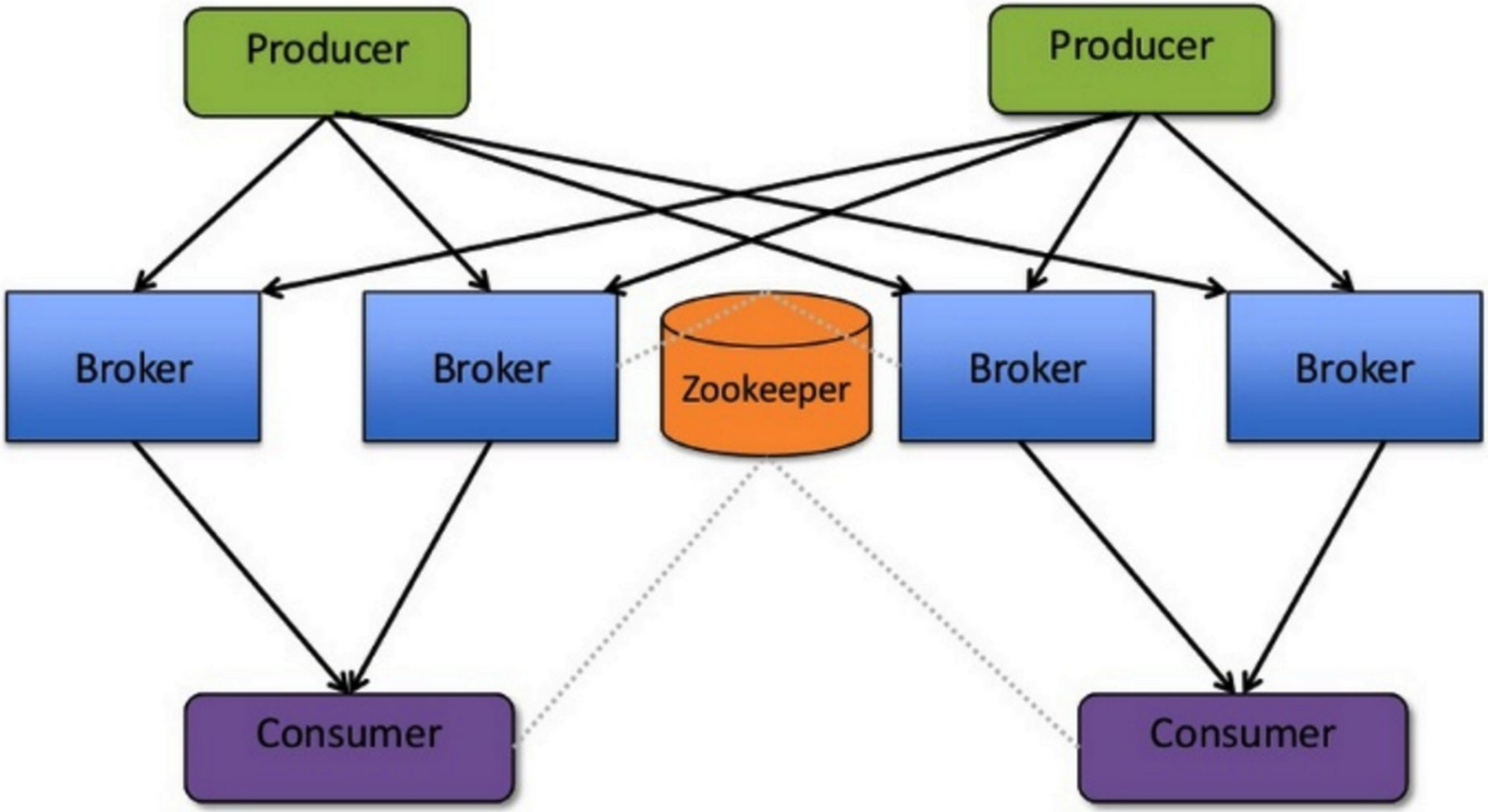
# How do I visualise data

- From a SQL database?
- From a NoSQL database?
- Generate charts in Python Spark?
- Etc?

# Collection / Queuing systems

- Two ways of making the choice
  - The protocol
  - The middleware
- Protocols
  - ZeroMQ, MQTT, AMQP, STOMP, Kafka Protocol, Rendevouz, etc
- Middleware
  - Kafka, Apollo, Mosquitto, QPid, MQ, Tibco, WSO2, etc

# Apache Kafka



Source: http://www.slideshare.net/charmalloc/

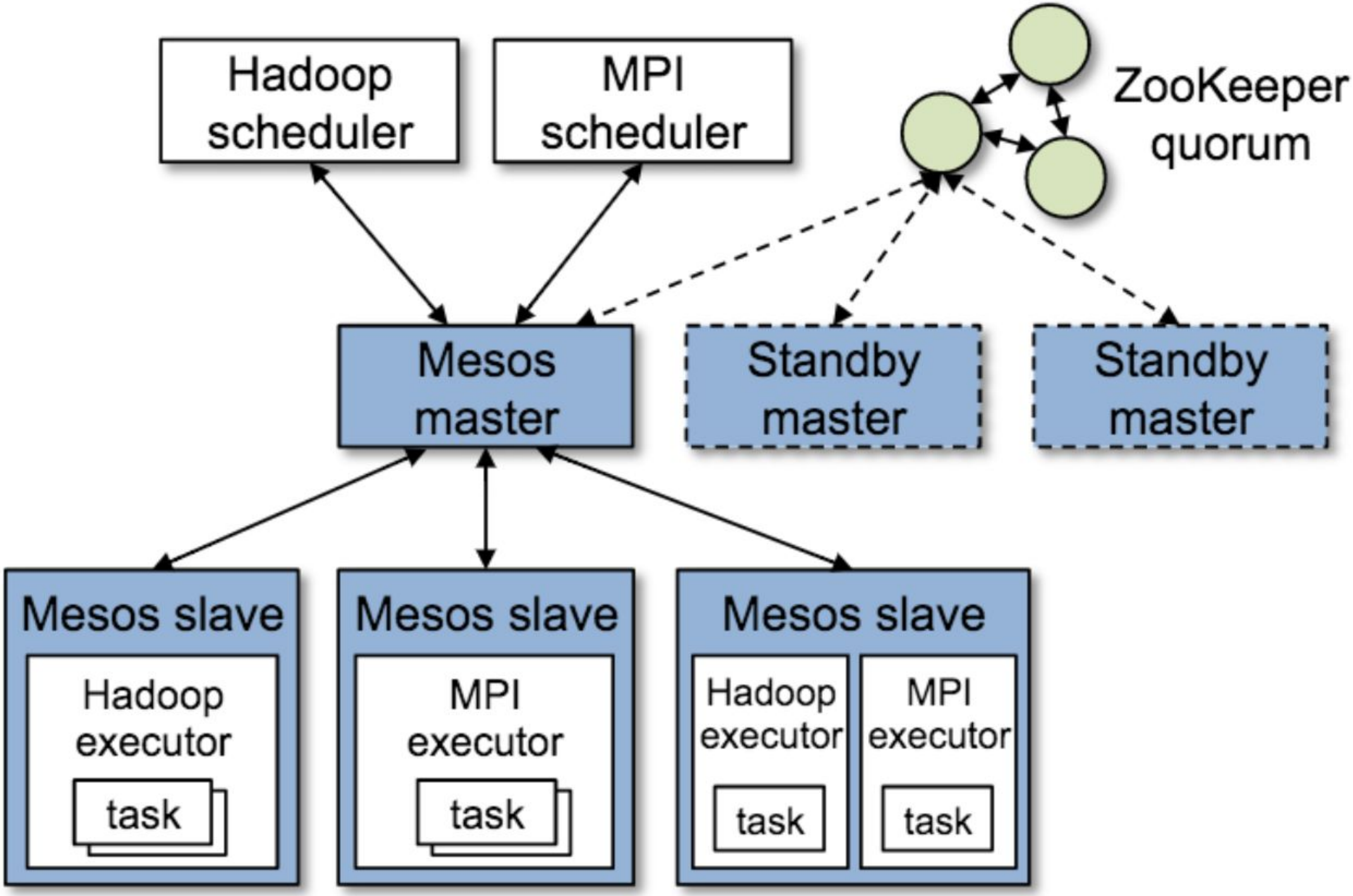
# Processing approaches

- Covered in detail already
- Hadoop
- Spark
- Tez
- etc

# Cluster management systems for Big Data

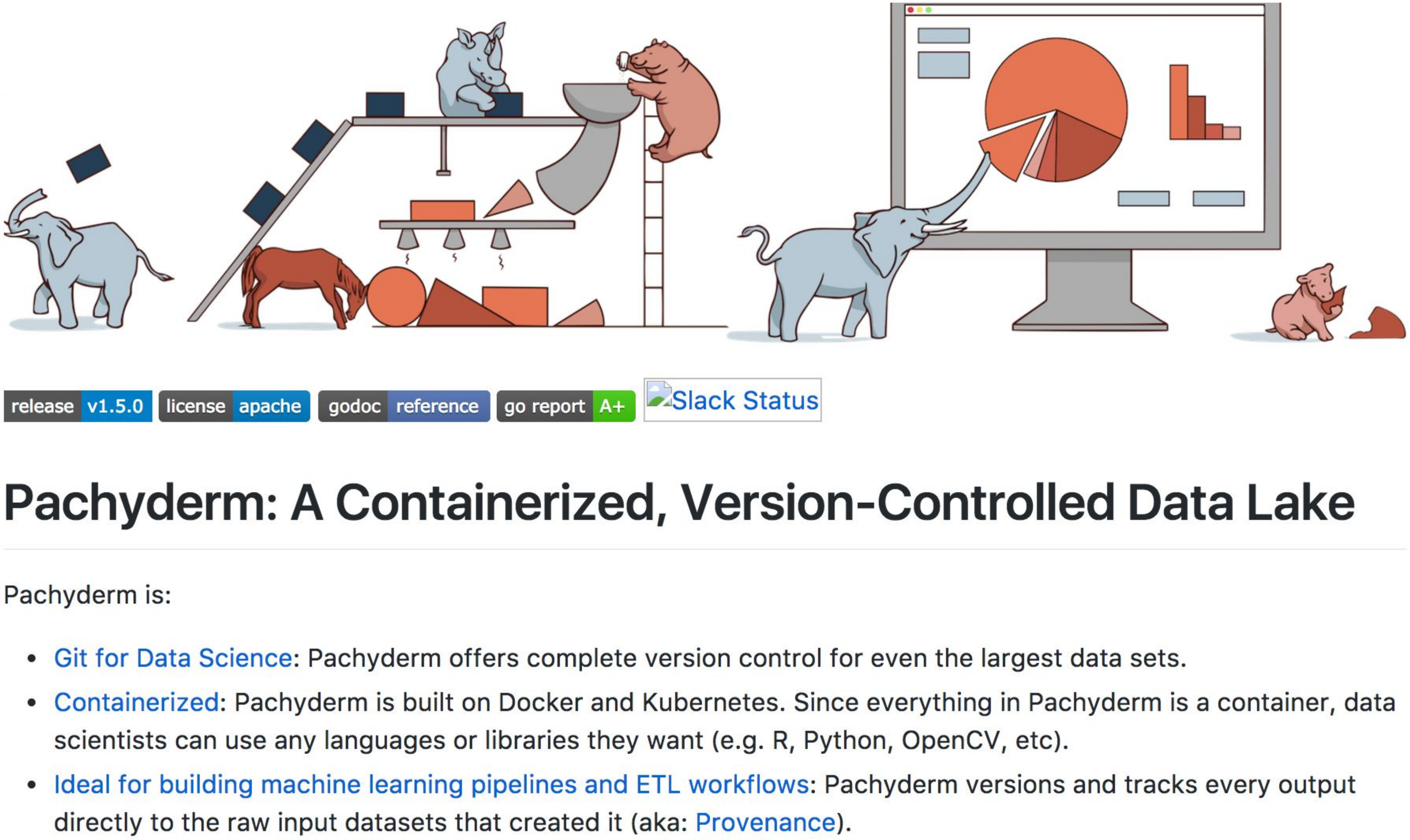
- YARN
  - Part of Hadoop but significantly rebuilt since Hadoop 1
- Mesos
  - Popular Apache project
  - Built to be a resource manager for a complete datacenter
    - Supports many workloads (e.g. Docker as well as Spark)

# Apache Mesos



# Pachyderm

https://github.com/pachyderm/pachyderm



release v1.5.0 license apache godoc reference go report A+ Slack Status

## Pachyderm: A Containerized, Version-Controlled Data Lake

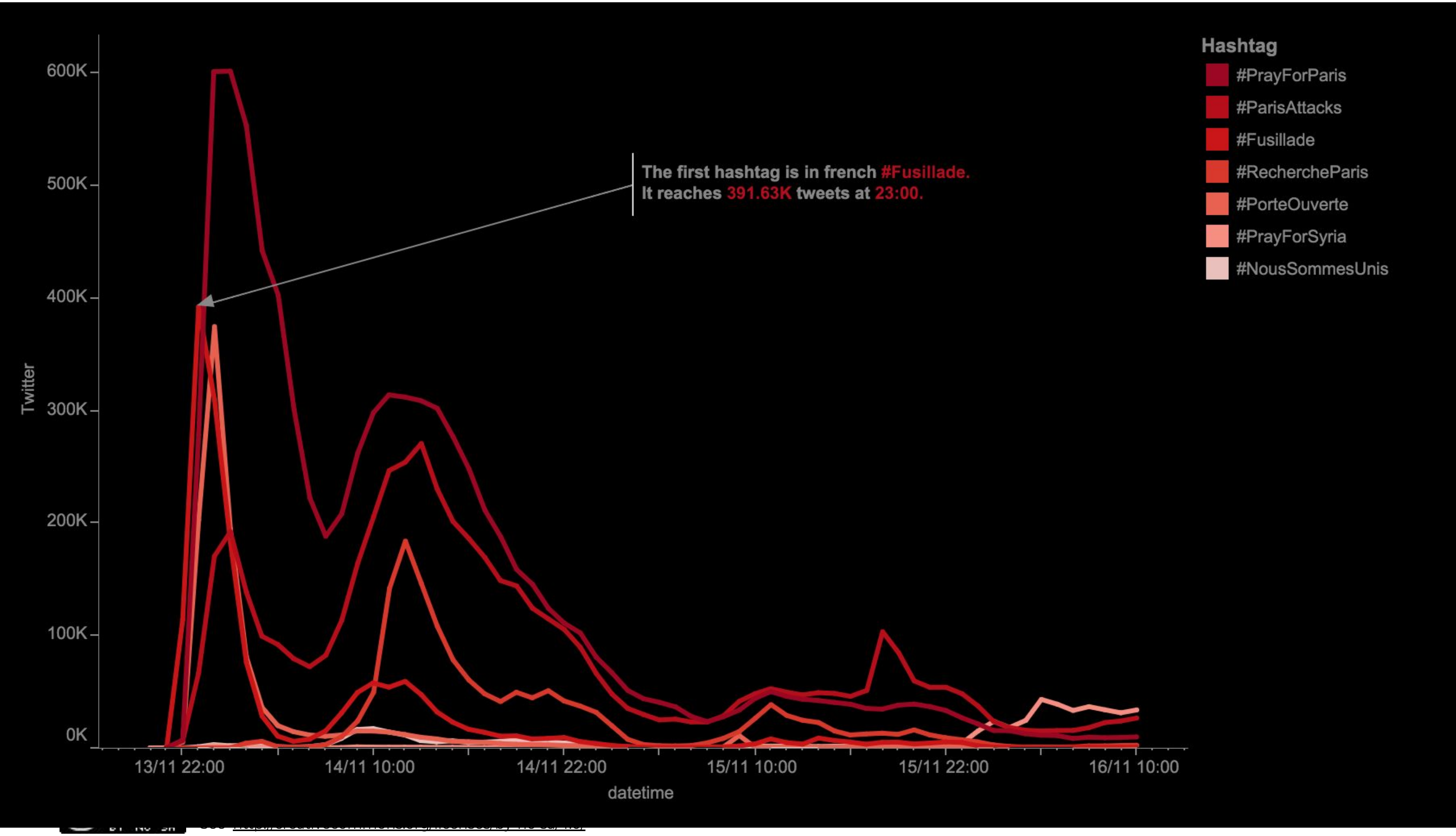
Pachyderm is:

- Git for Data Science: Pachyderm offers complete version control for even the largest data sets.
- Containerized: Pachyderm is built on Docker and Kubernetes. Since everything in Pachyderm is a container, data scientists can use any languages or libraries they want (e.g. R, Python, OpenCV, etc).
- Ideal for building machine learning pipelines and ETL workflows: Pachyderm versions and tracks every output directly to the raw input datasets that created it (aka: Provenance).

# Realtime

- Apache Storm
  - – Highly flexible model
  - – Supports pure streaming and micro-batch
  - – Lots of plugins
- Apache Spark
  - – Micro-batch only
  - – Integrates cleanly into Spark (fewer components)
  - – Some plugins and more being developed

# Visualisation



The first hashtag is in french #Fusillade.
It reaches 391.63K tweets at 23:00.

# Visualisation approaches

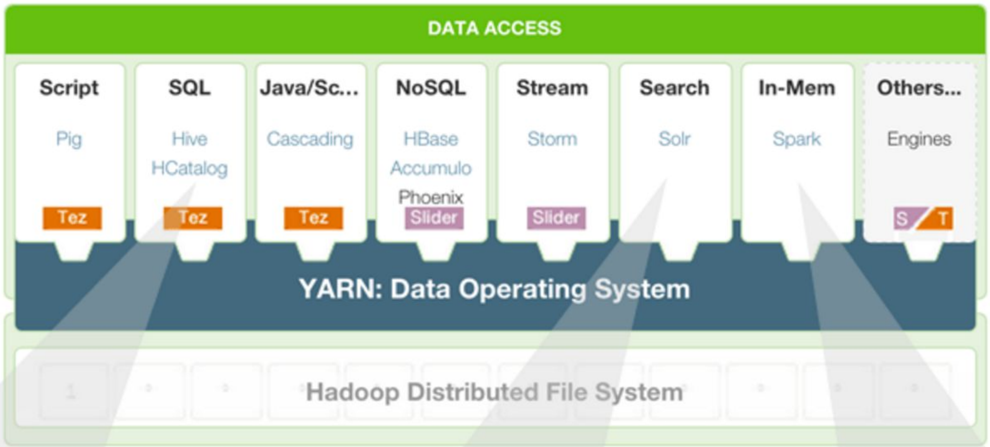
- Full products
  - Tableau, Qlik, SAS, GoodData
- Web-based systems
  - Tableau Public, Datawrapper, Raw, Plotly
- Developer oriented
  - D3.js, dygraphs, Python charting, Leaflet, Fusion Charts, Google Charts, etc

# Fortune top 10 big data companies

fortune.com/2014/06/13/these-big-data-companies-are-ones-to-watch/

- MapR – Apache Hadoop
- MemSQL
- Databricks – Apache Spark
- Platfora – Apache Hadoop
- Splunk
- Teradata – Apache Hadoop
- Palantir – Hadoop, Cassandra, Lucene
- Premise
- Datameer – Apache Hadoop
- Cloudera – Apache Hadoop
- Hortonworks – Apache Hadoop
- MongoDB – MongoDB
- Trifacta – Apache Hadoop

# Hortonworks



## Enhanced SQL Semantics in Apache Hive

Hive adds time intervals and UNION semantics, 2.5x performance improvements and improved query scheduling, along with a more streamlined user interface for Hive within Ambari.

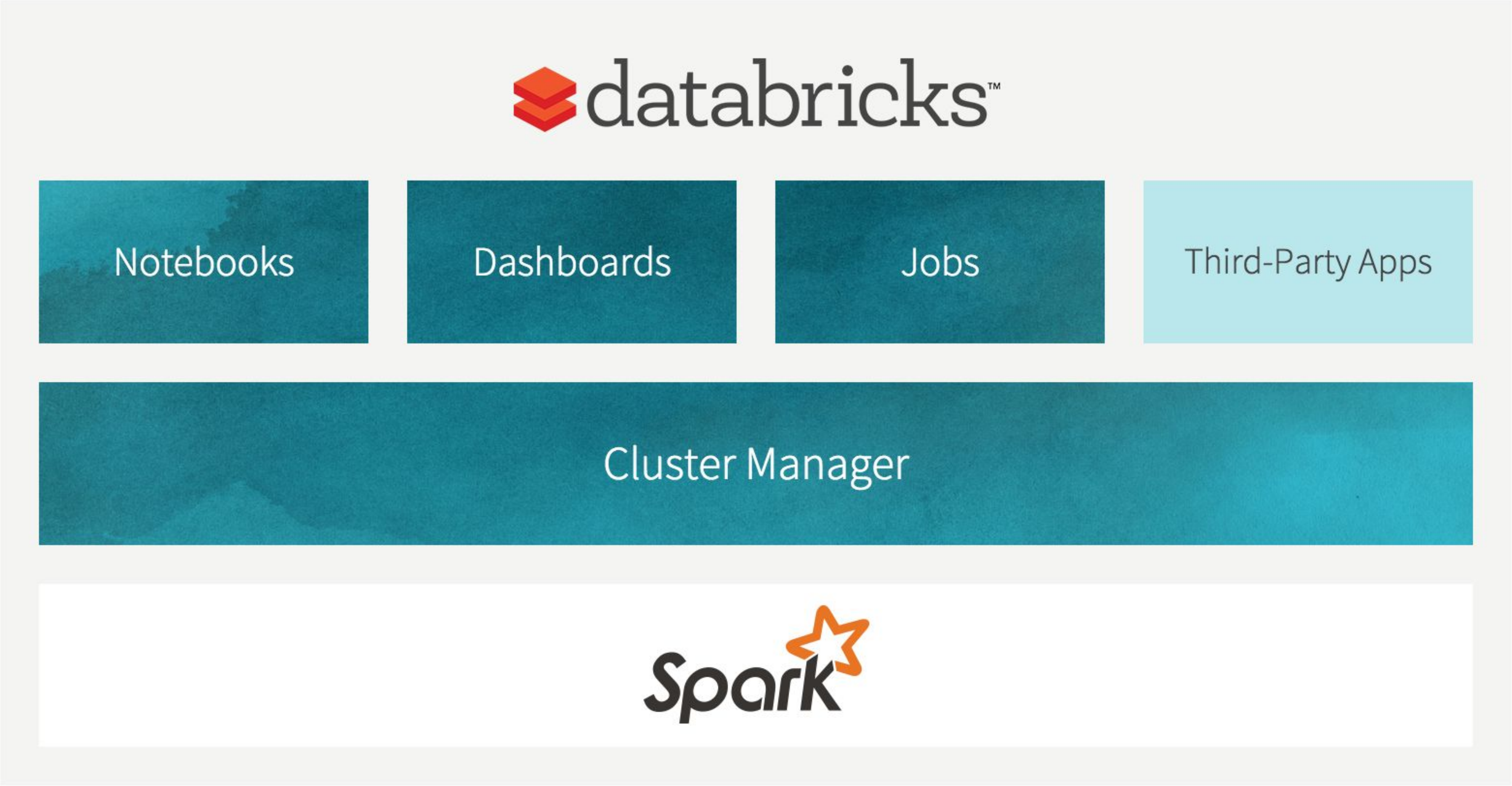
## Solr on YARN

The Solr search engine is being built to run on YARN and is now in technical preview. This critical advancement allows customers to reduce their total cost of ownership by deploying Solr within the same cluster as other workloads – eliminating the need for a “side cluster” dedicated to indexing data and delivering search results.

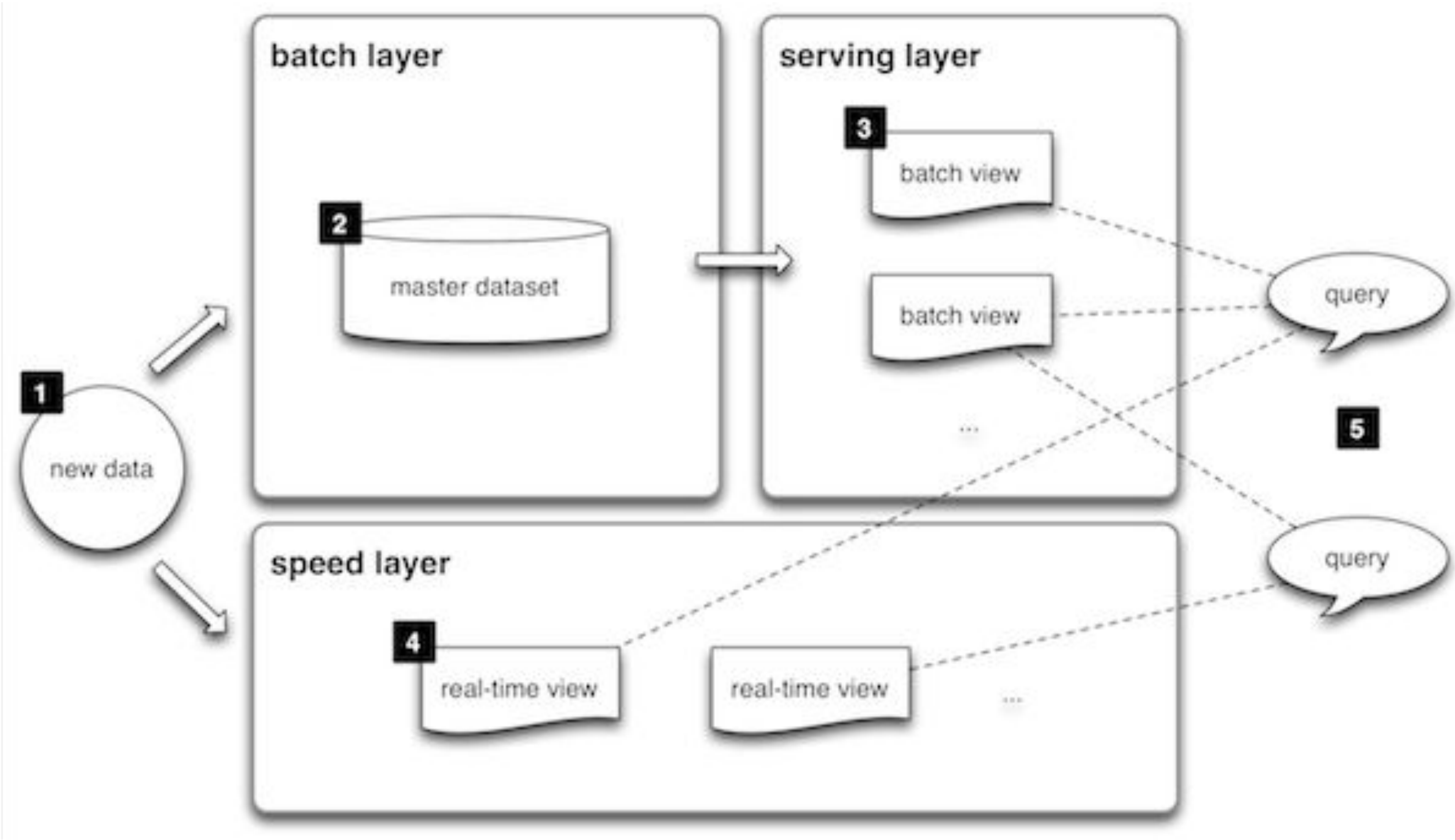
## New capabilities for feature-rich Spark applications

Apache Spark on YARN is enhanced with the new DataFrame API, machine learning algorithms such as clustering, frequent pattern-mining algorithms and a technology preview of SparkSQL.

# Databricks


databricks™
Notebooks
Dashboards
Jobs
Third-Party Apps
Cluster Manager
Spark

# Lambda


batch layer
serving layer
3
batch view
2
master dataset
batch view
query
1
new data
...
5
speed layer
query
4
real-time view
real-time view
...

# The real answer

You are on the bleeding edge

- Expect to have some pain

# Questions?