SPA Conference 2016

Real-World Big Data in Action

Nick Rozanski
Eoin Woods
Chris Cooper-Bland

Number 2 in an occasional series

Agenda

Agenda

- Introduction to Big Data (15 minutes)
- Exercise 1: Hadoop (60 minutes)
- Exercise 2: Spark (30 minutes)
- Exercise 3: Hive (45 minutes)

Real-World Big Data in Action

Introduction to Big Data

History and Background

The Origins of Big Data

The Three V's

- First proposed by META Group analyst Doug Laney in 2001 (https://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf)
- Driven by the (then) looming growth in e-commerce and the strain this would impose on computer systems

Volume

- The increase in depth and breadth of data

Velocity

- The speed at which data needs to be made available for use

Variety

- Problems caused by incompatible data formats, non-aligned data structures and inconsistent data semantics

- Subsequent analysts have added additional V's such as **Variability** and **Veracity** (data quality)

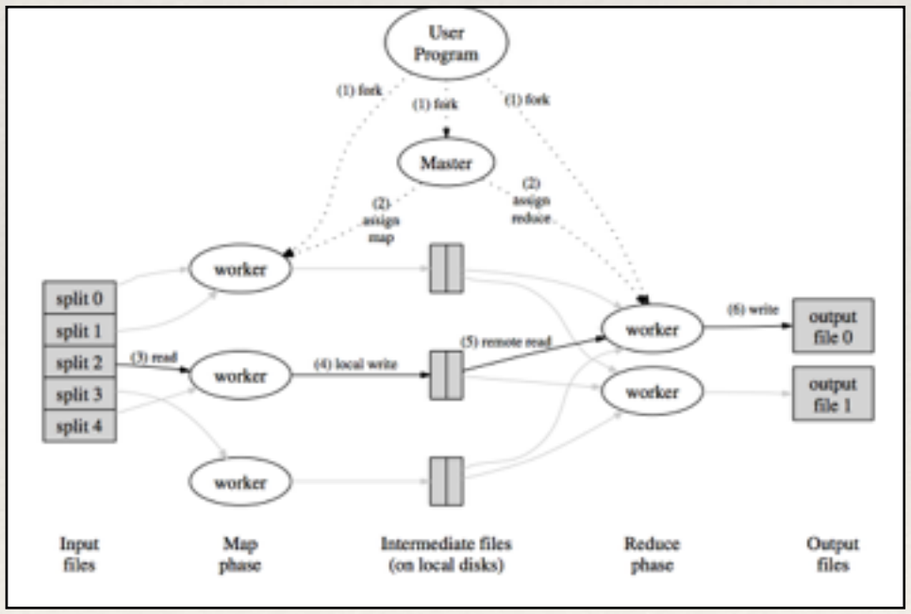
The MapReduce Algorithm

MapReduce

- Paper published by Google in 2004 (http://research.google.com/archive/mapreduce.html)
- Describes how they rewrote their production indexing system using MapReduce
- Provides automatic parallelization and distribution, fault-tolerance, I/O scheduling and status monitoring

Map Reduce Steps

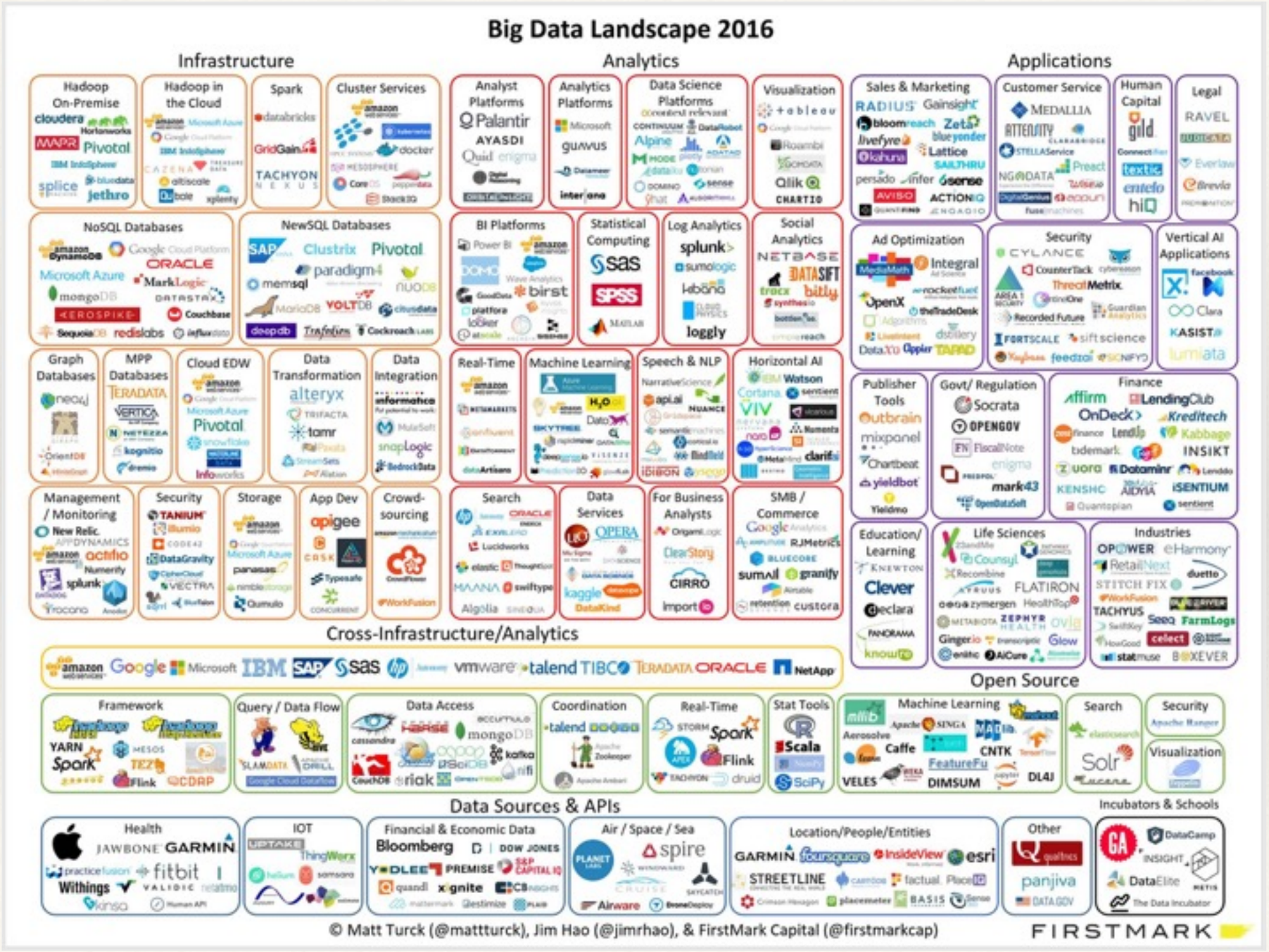
- *Map step*: master breaks up query and distributes portions across a massive number of computers
- *Reduce step*: results collated and returned to requestor



Benchmark

- scan 10 billion 100-byte records to extract records matching a rare pattern (92K matching records); once started up 1800 machines read 1 TB of data at peak of ~31 GB/s

Today's Big Data Landscape



All Big Data tools are required by EU Law to have ridiculous names

- Sqoop
- Oozie
- Pig
- Impala
- Flume
- Parquet

A Simple Big Data Stack

Hadoop

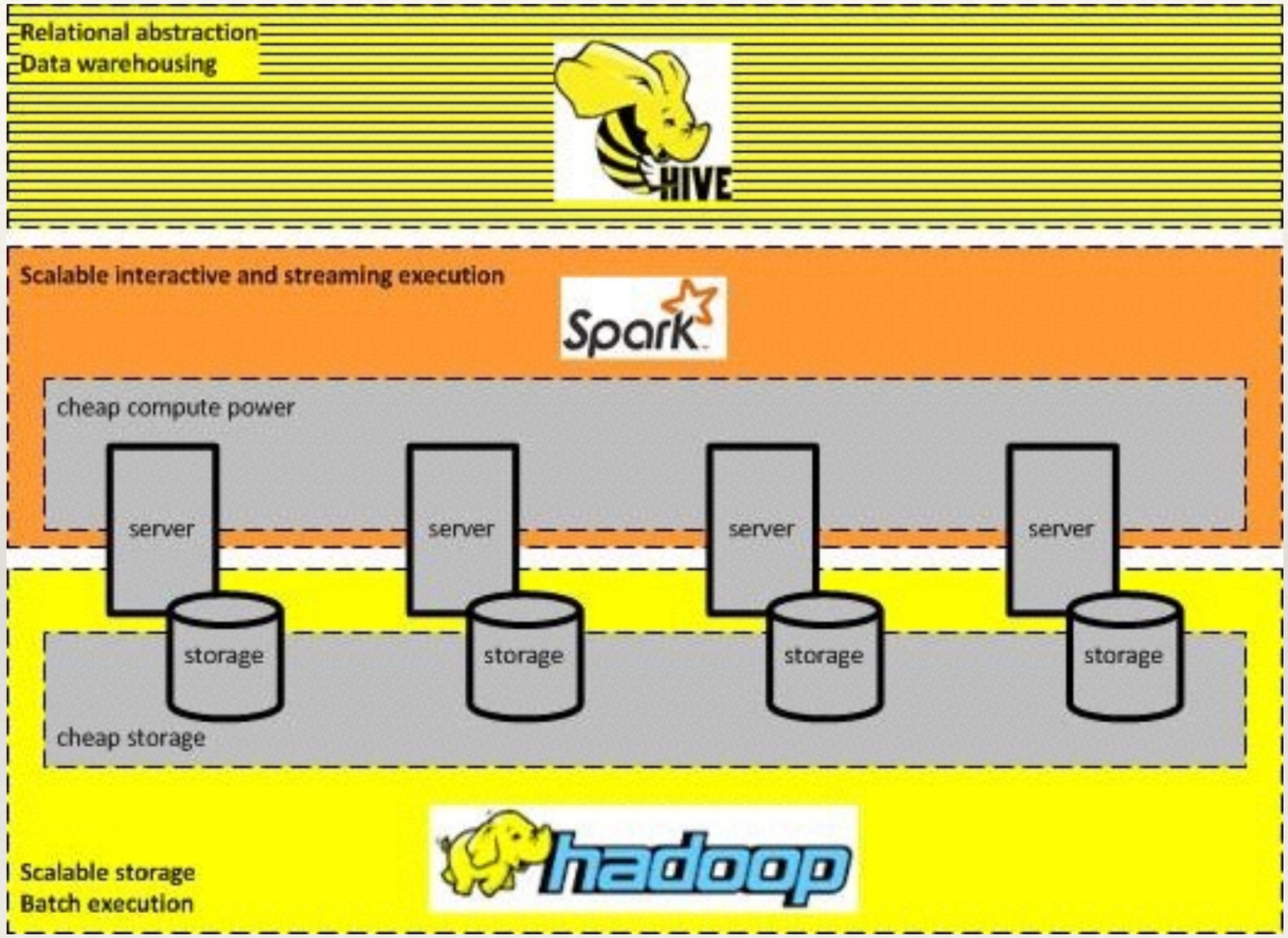
- scalable file storage
- batch execution

Spark

- scaleable interactive and streaming execution

Hive

- relational abstraction
- data warehousing



Hadoop, Spark and Hive

Hadoop

- Doug Cutting and others started working on a web crawler called Nutch in 2002
- This eventually morphed into Hadoop and was adopted by Yahoo! in 2006
- It became a top-level Apache project and was adopted by Last.fm, Facebook and others
- The Yahoo web map comprised 100 billion nodes and 1 trillion edges by 2009
- Hadoop continued to break records for volume and velocity: in 2014, a team from Databricks sorted 100TB of data in 1,406 seconds on 207 nodes (4.27TB per min)
- Hadoop is a made-up name (the name of Doug Cutting's daughter's yellow toy elephant)

Spark

- Started at UC Berkeley's AMPLab in 2009 and open sourced in 2010
- A Top-Level Apache Project since 2014

Hive

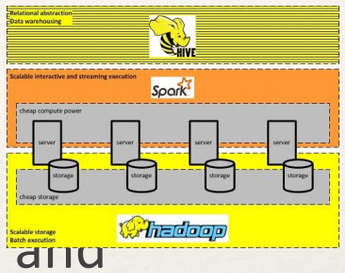
- Originally developed at Facebook around 2007-8
- https://www.facebook.com/notes/facebook-engineering/hive-a-petabyte-scale-data-warehouse-using-hadoop/89508453919/
- Now used at Netflix, FINRA (UK regulator), and part of Amazon Elastic MapReduce

Real-World Big Data in Action

Hadoop

A Big Data Virtual Filesystem

Hadoop Overview



MORE WORK ON THIS

- HDFS – scalable, fault-tolerant, distributed file system
- distributes storage and computation across many servers, so can grow linearly and economically with demand
- Hadoop moves compute processes to the data on HDFS and not the other way around
- Health diagnosis, management and data rebalancing with minimal operator intervention
- https://www.quora.com/What-are-some-of-the-largest-Hadoop-clusters-to-date
- Offers a subset of POSIX file management capabilities
- Data is stored in datanodes, with data replicated across them for performance and reliability
- NameNode keeps the directory tree and tracks where across the cluster data is kept
- Clients talk to the NameNodewhen they wish to locate a file
- NameNode is a SPOF – if it is offline, HDFS is offline
- SecondaryNameNode provides some resilience but…
- YARN for cluster management
- https://wiki.apache.org/hadoop/FrontPage

Exercise 1: Hadoop

Goals of This Exercise

- install the Big Data software
- configure the Big Data software
- start Hadoop and check it is running
- format the Hadoop filesystem
- load a file into Hadoop
- browse the Hadoop filesystem

Real-World Big Data in Action

Spark

A Big Data Processing Engine

Spark Overview

MORE WORK ON THIS

- Resilient Distributed Dataset – a fault-tolerant collection of elements that can be operated on in parallel
- RDDs often reference a dataset in an external storage system such as Hadoop HDFS
- Once created, can call operations on the RDD (count, sum, average etc, plus more sophisticated analysis like graphing and machine learning) which execute in parallel
- Spark cuts the dataset into a number of partitions and runs a task for each
- It processes data in-memory so can be much faster than traditional map / reduce
- Has an extensive set of APIs for Java, Scala and Python
- DataFrames (since 2015) – “a distributed collection of data organized into named columns”
- conceptually equivalent to a table in a relational database, in particular queries can be optimised (like in an RDBMS)
- can then access programmatically, or using Spark SQL or Hive SQL (Hive seems to offer more features, esp. around security)
- you are sacrificing performance for convenience
- we will be mapping fields in CVS files to DataFrames, but can also use JSON, Parquet, RDBMS tables etc)

Exercise 2: Spark

Goals of This Exercise

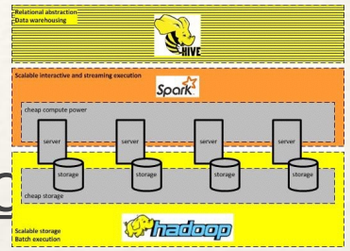
- start Hadoop and Spark and check they are running
- start the Spark client
- load the Hadoop file into Spark
- do some data science!

Real-World Big Data in Action

Hive

A Big Data data warehousing infrastructure

Hive Overview



NOT REALLY STARTED YET

- Metastore http://www.cloudera.com/documentation/archive/cdh/4-x/4-2-0/C Installation-Guide/cdh4ig_topic_18_4.html
- Uses Apache Derby – small-footprint embedded RDBMS implemented in Java

Exercise 3: Hive

Goals of This Exercise

- start Hadoop, Spark and Hive and check they are running
- start the Hive client
- create a Hive table from the Hadoop file
- do some more data science!

Next Steps

Bigger Big Data
Other Big Data Tools

Next Steps

- Multiple slaves
- YARN
- Cloudera VM

TO DO

Appendix

Further Information

Useful Links

Hadoop FAQ

- http://wiki.apache.org/hadoop/FAQ

Hadoop Filesystem commands Reference

- http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html

Set Up Hadoop Cluster

- https://hadoop.apache.org/docs/r2.7.2/hadoop-project-dist/hadoop-common/SingleCluster.html#Prepare_to_Start_the_Hadoop_Cluster

Pyspark sqlContext Reference

- https://spark.apache.org/docs/latest/api/python/pyspark.sql.html#pyspark.sql.DataFrame

Hive SQL Reference

- https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DDL
- https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DML