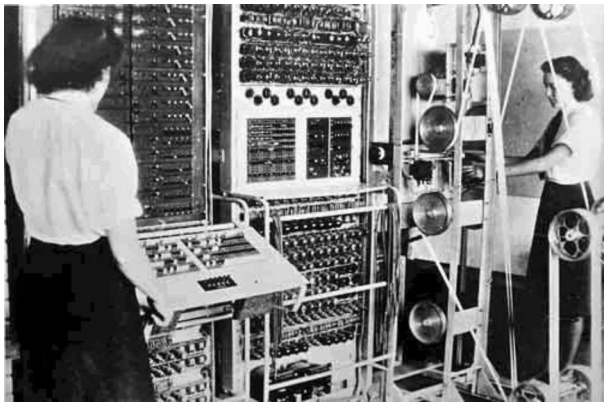
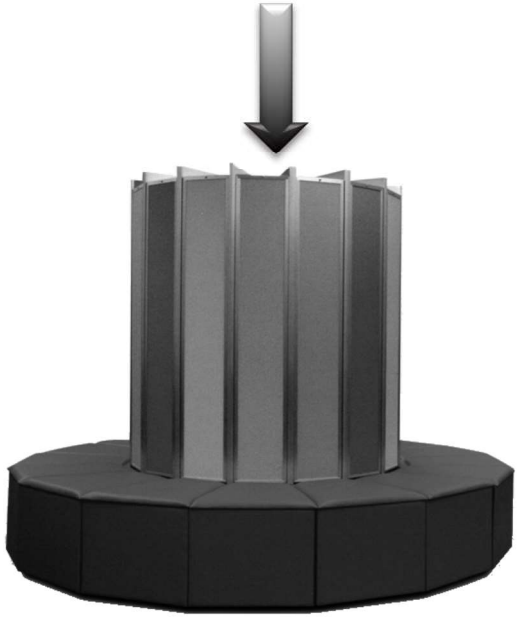
Stanford CS246H:
Mining Massive Data Sets
Hadoop Lab
Winter 2017

Traditional Large-Scale Computation

- **Traditionally, computation has been processor-bound**
  - Relatively small amounts of data
  - Lots of complex processing



- **The early solution: bigger computers**
  - Faster processor, more memory
  - But even this couldn't keep up

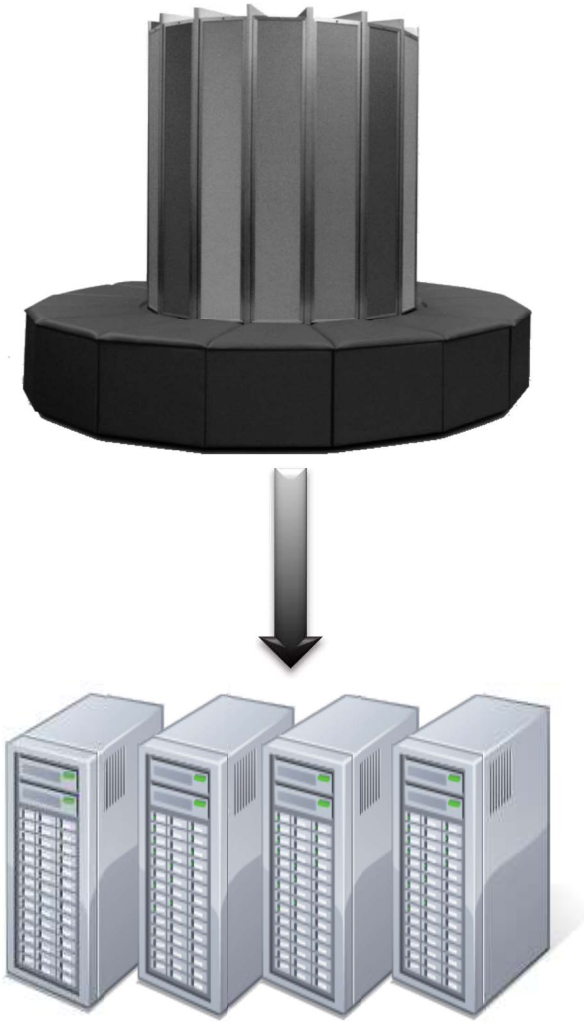


Distributed Systems

- **The better solution: more computers**
  - Distributed systems – use multiple machines for a single job

"In pioneer days they used oxen for heavy pulling, and when one ox couldn't budge a log, we didn't try to grow a larger ox. We shouldn't be trying for bigger computers, but for *more systems* of computers."

– Grace Hopper



Challenges with Distributed Systems

- **Challenges with distributed systems**
  - Programming complexity
    - Keeping data and processes in sync
  - Finite bandwidth
  - Partial failures

- **The solution?**
  - Hadoop!

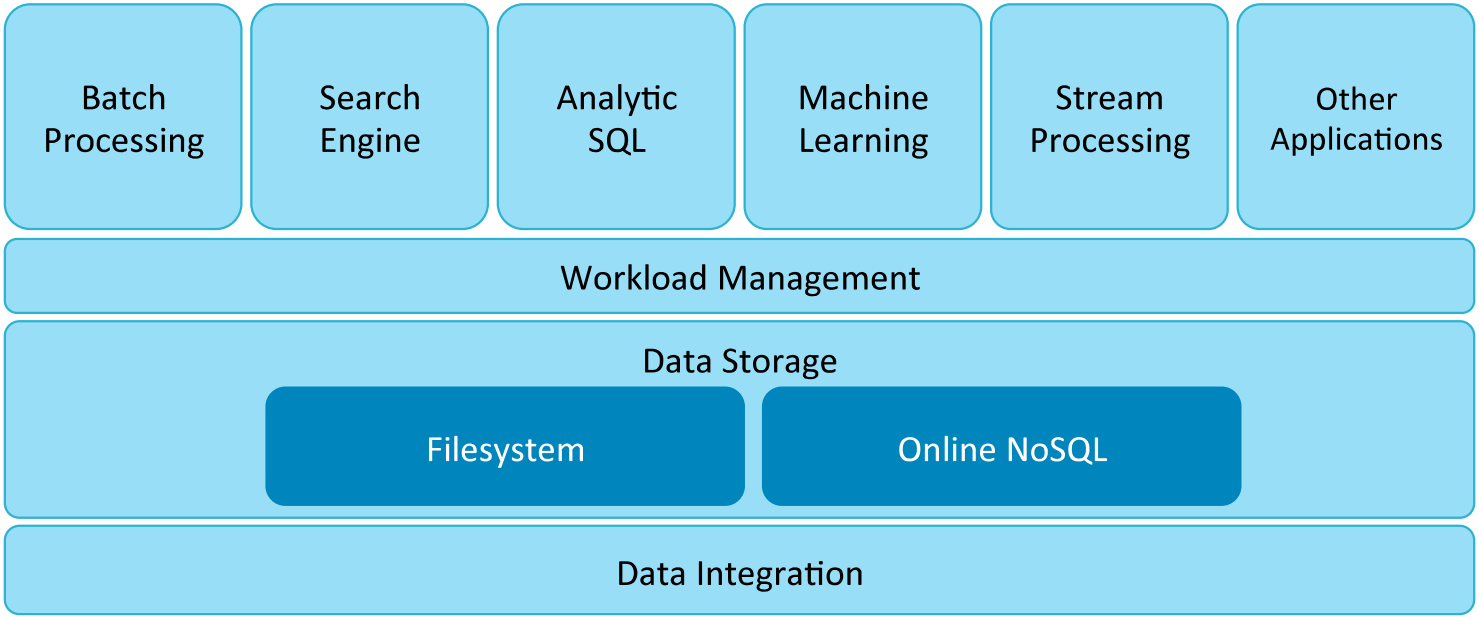
The Real Reason You're Here

http://www.edureka.co/blog/5-reasons-to-learn-hadoop

Introduction to Hadoop and the Hadoop Ecosystem

What is Apache Hadoop?

- **Scalable and economical data storage, processing and analysis**
  - Distributed and fault-tolerant
  - Harnesses the power of industry standard hardware
- **Heavily inspired by technical documents published by Google**



Common Hadoop Use Cases

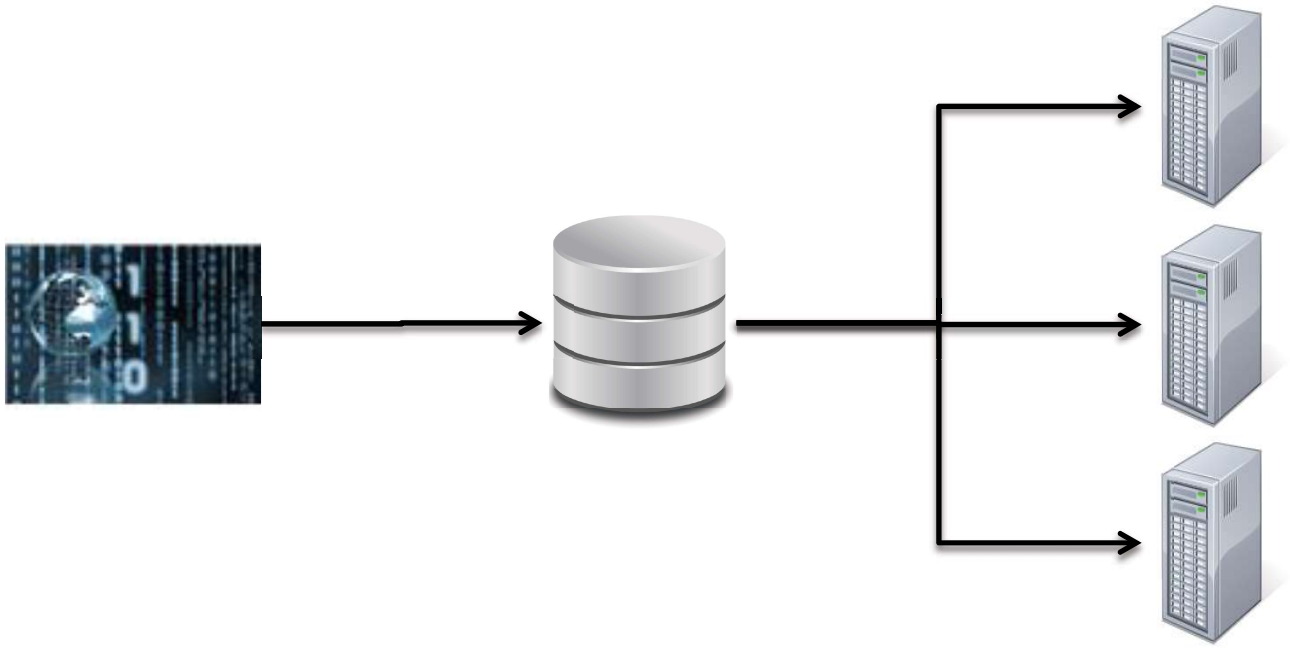
- **Extract/Transform/Load (ETL)**
- **Text mining**
- **Index building**
- **Graph creation and analysis**
- **Pattern recognition**
- **Collaborative filtering**
- **Prediction models**
- **Sentiment analysis**
- **Risk assessment**

- **What do these workloads have in common? Nature of the data…**
  - Volume
  - Velocity
  - Variety

Distributed Systems: The Data Bottleneck (1)

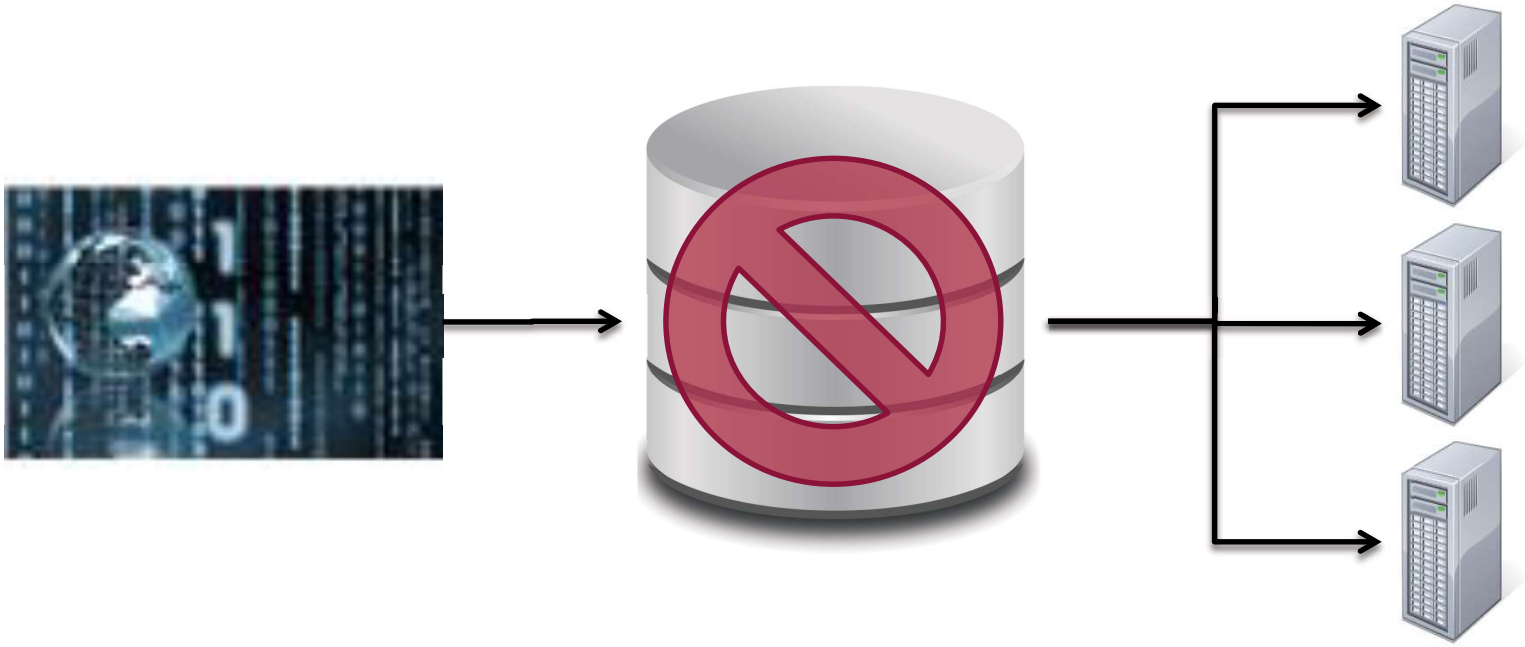
- **Traditionally, data is stored in a central location**
- **Data is copied to processors at runtime**
- **Fine for limited amounts of data**

CPU bound vs. I/O bound



Distributed Systems: The Data Bottleneck (2)

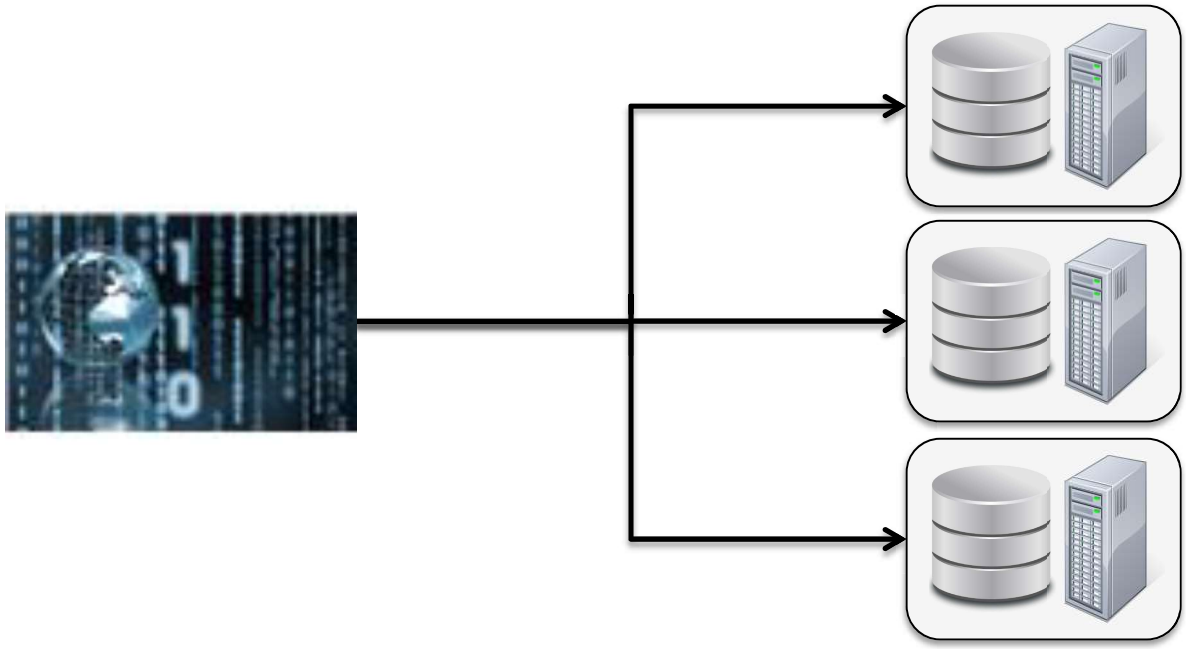
- **Modern systems have much more data**
  - terabytes+ a day
  - petabytes+ total
- **We need a new approach…**



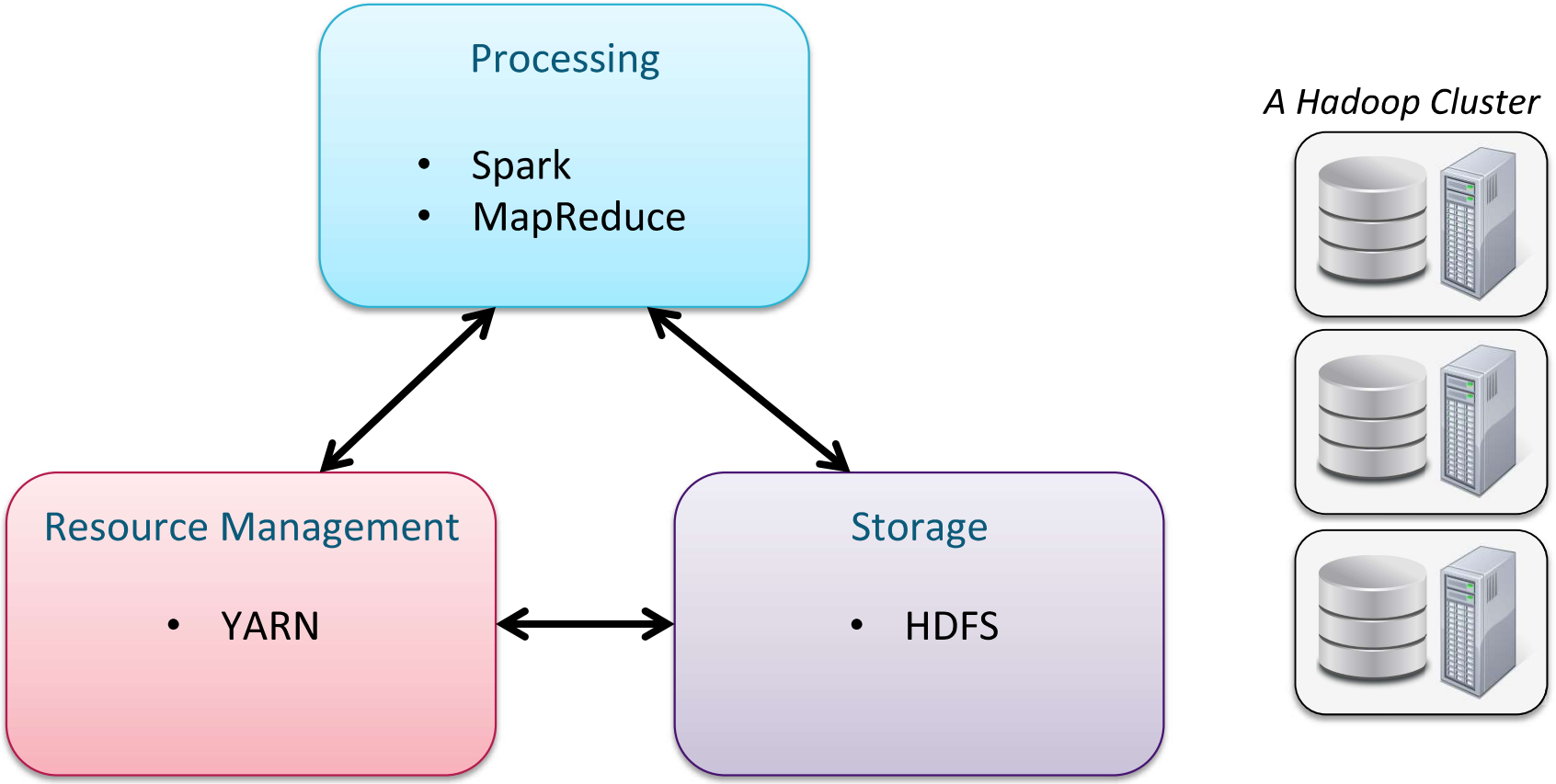
Big Data Processing with Hadoop

- **Hadoop introduced a radical new approach:**
  - Bring the program to the data rather than the data to the program
- **Based on two key concepts**
  - Distribute data when the data is stored
  - Run computation where the data resides

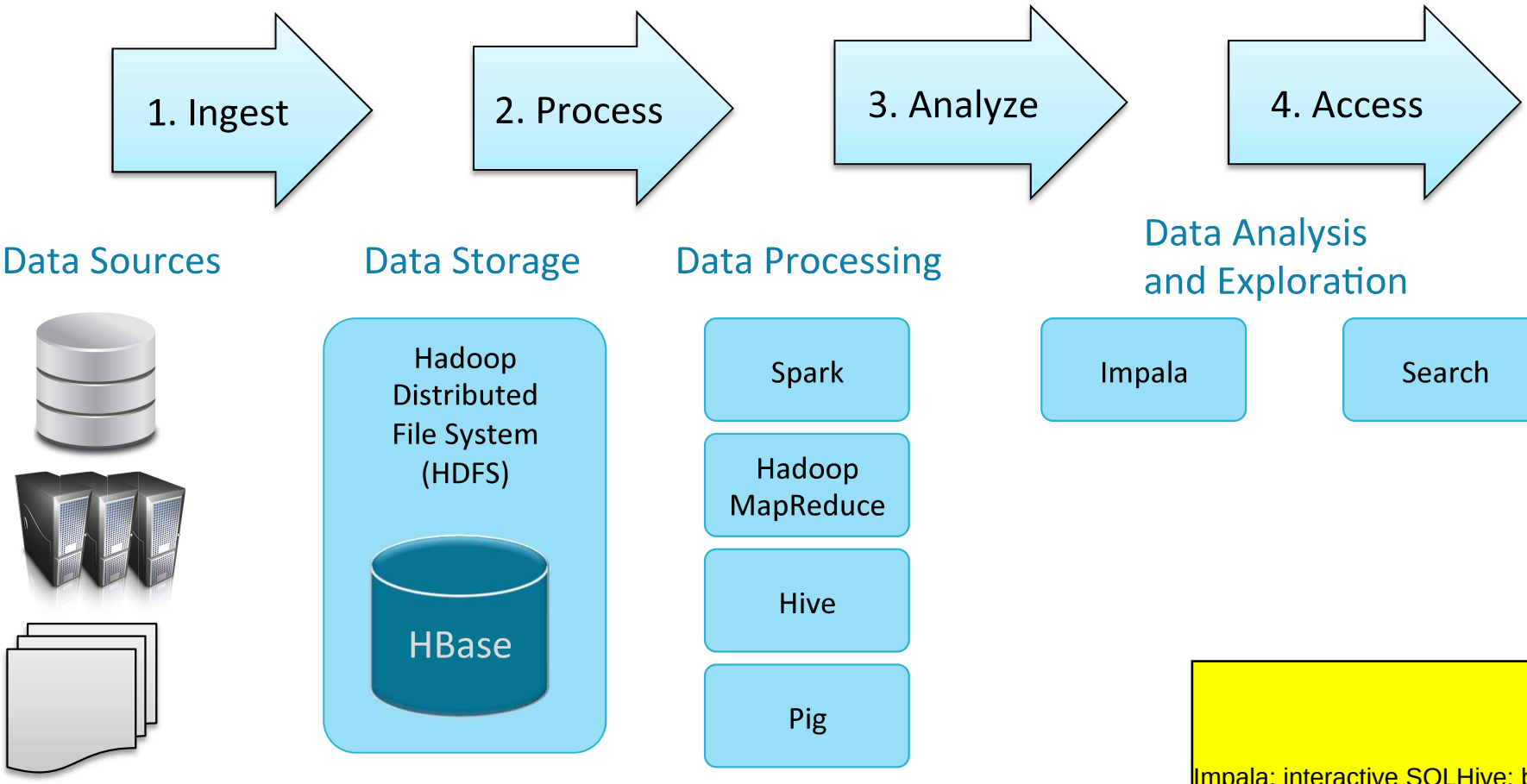
A Hadoop Cluster



Core Hadoop

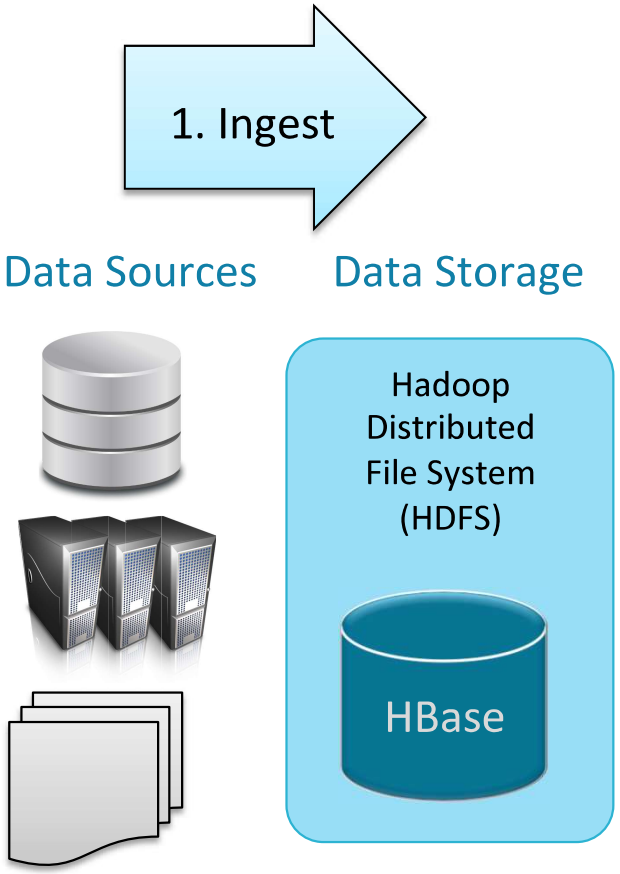


Big Data Processing



Data Ingest and Storage

- **Hadoop typically ingests data from many sources and in many formats**
  - Traditional data management systems, e.g. databases
  - Logs and other machine generated data (event data)
  - Imported files

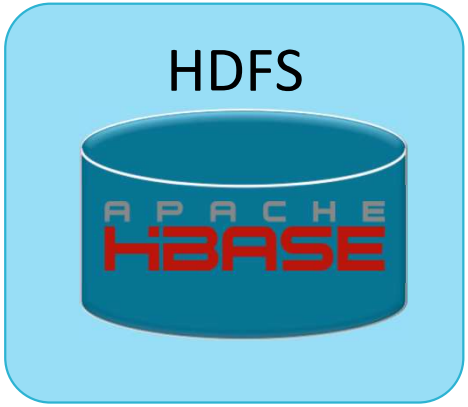


Data Storage

- **Hadoop Distributed File System (HDFS)**
  - HDFS is the storage layer for Hadoop
  - Provides inexpensive reliable storage for massive amounts of data on industry-standard hardware
  - Data is distributed when stored
  - Covered later in this course

HDFS is inspired by Google

- **Apache HBase: The Hadoop Database**
  - A NoSQL distributed database built on HDFS
  - Scales to support very large amounts of data and high throughput
  - A table can have thousands of columns
  - Covered in depth in *Cloudera Training for Apache HBase*

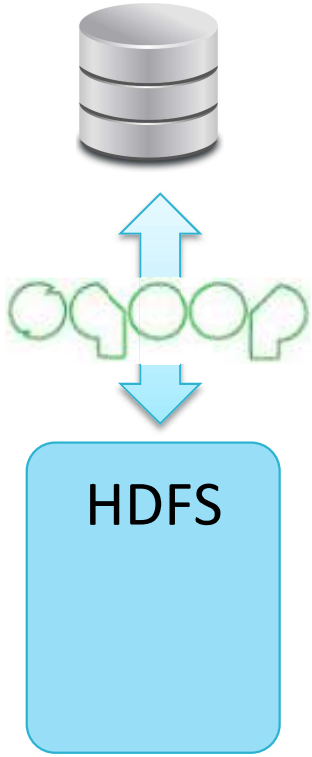


HBase is inspired by Goo

Data Ingest Tools (1)

- **HDFS**
  - Direct file transfer
- **Apache Sqoop**
  - High speed import to HDFS from Relationship Database (and vice versa)
  - Supports many data storage systems
    - e.g. Netezza, Mongo, MySQL, Teradata, Oracle
  - Covered later in this course

from relational databases

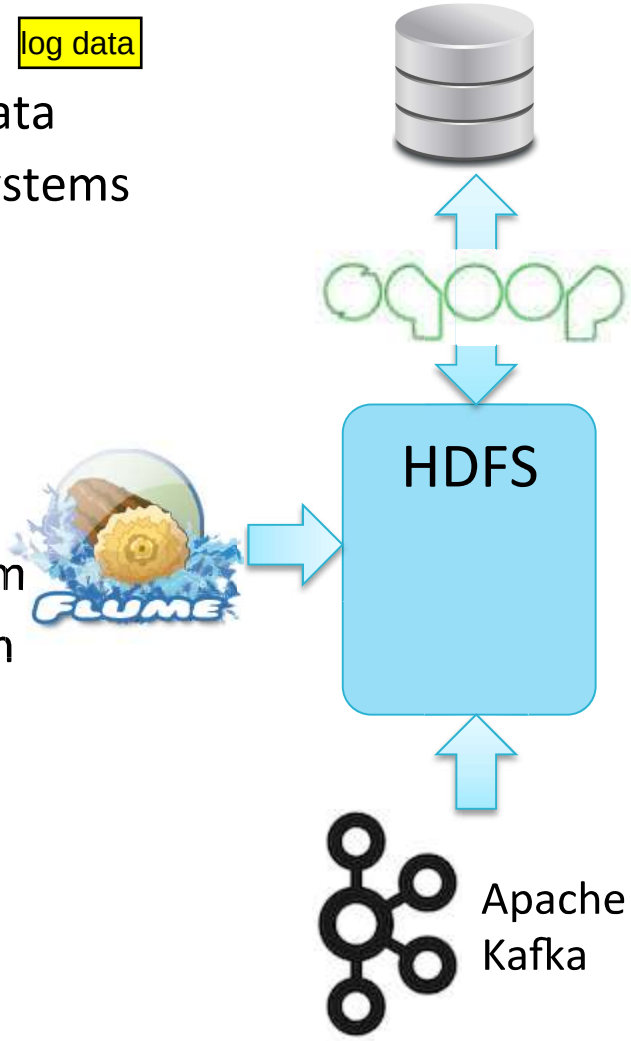


Note: Apache has retired Sqoop

Data Ingest Tools (2)

- **Apache Flume**
  - Distributed service for ingesting streaming data
  - Ideally suited for event data from multiple systems
    - For example, log files
  - Covered later in this course

- **Kafka**
  - A high throughput, scalable messaging system
  - Distributed, reliable publish-subscribe system
  - Integrates with Flume and Spark Streaming



Apache Spark: An Engine For Large-scale Data Processing

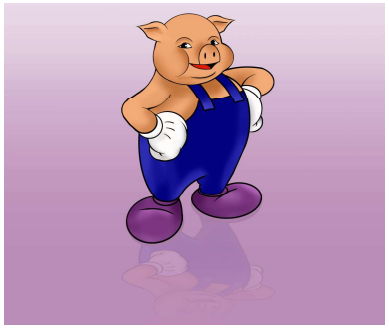
- **Spark is large-scale data processing engine**
  - General purpose
  - Runs on Hadoop clusters and data in HDFS
- **Supports a wide range of workloads**
  - Machine learning
  - Business intelligence
  - Streaming
  - Batch Processing
- **This course uses Spark for data processing**

Hadoop MapReduce: The Original Hadoop Processing Engine

- **Hadoop MapReduce is the original Hadoop framework**
  - Primarily Java based
- **Based on the MapReduce programming model**
- **The core Hadoop processing engine before Spark was introduced**
- **Still the dominant technology**
  - But losing ground to Spark fast
- **Many existing tools are still built using MapReduce code**
- **Has extensive and mature fault tolerance built into the framework**

Apache Pig: Scripting for MapReduce

- **Apache Pig builds on Hadoop to offer high-level data processing**
  - This is an alternative to writing low-level MapReduce code
  - Pig is especially good at joining and transforming data
- **The Pig interpreter runs on the client machine**
  - Turns Pig Latin scripts into MapReduce or Spark jobs
  - Submits those jobs to a Hadoop cluster
  - Covered in Cloudera *Data Analyst Training*



```
people = LOAD '/user/training/customers' AS (cust_id, name);
orders = LOAD '/user/training/orders' AS (ord_id, cust_id, cost);
groups = GROUP orders BY cust_id;
totals = FOREACH groups GENERATE group, SUM(orders.cost) AS t;
result = JOIN totals BY group, people BY cust_id;
DUMP result;
```

Cloudera Impala: High Performance SQL

- **Impala is a high-performance SQL engine**
  - Runs on Hadoop clusters
  - Data stored in HDFS files
  - Inspired by Google's Dremel project
  - Very low latency – measured in milliseconds
  - Ideal for interactive analysis
- **Impala supports a dialect of SQL (Impala SQL)**
  - Data in HDFS modeled as database tables
- **Impala was developed by Cloudera**
  - 100% open source, released under the Apache software license
- **Impala is used for data analysis in this course**

Apache Hive: SQL on MapReduce

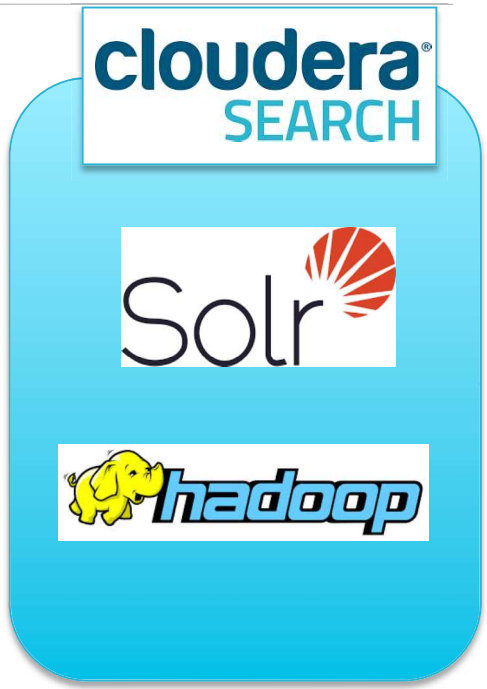
- **Hive is an abstraction layer on top of Hadoop**
  - Hive uses a SQL-like language called HiveQL
  - Similar to Impala SQL
  - Useful for data processing and ETL
    - Impala is preferred for ad hoc analytics
- **Hive executes queries using MapReduce**
  - Hive on Spark is available for early adopters; not yet recommended for production
- **Hive can optionally be used for data analysis in this course**

Now, Spark/Tez is the recommende

Cloudera Search: A Platform For Data Exploration

Solr

- **Interactive full-text search for data in a Hadoop cluster**
- **Allows non-technical users to access your data**
  - Nearly everyone can use a search engine
- **Cloudera Search enhances Apache Solr**
  - Integrates Solr with HDFS, MapReduce, HBase, and Flume
  - Supports file formats widely used with Hadoop
  - Dynamic Web-based dashboard interface with Hue
  - Apache Sentry based security
- **Cloudera Search is 100% open source**

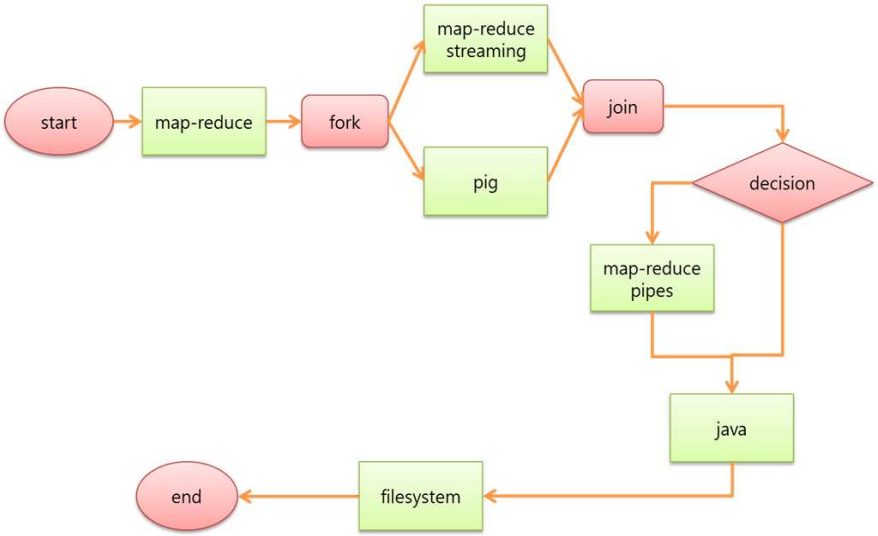


Hue: The UI for Hadoop

- **Hue = Hadoop User Experience**
- **Hue provides a Web front-end to a Hadoop**
  - Upload and browse data
  - Query tables in Impala and Hive
  - Run Spark and Pig jobs and workflows
  - Search
  - And much more
- **Makes Hadoop easier to use**
- **Hue is 100% open-source**
- **Created by Cloudera**
  - Open source, released under Apache license
- **Hue is used throughout this course**

Apache Oozie: Workflow Management

- **Oozie**
  - Workflow engine for Hadoop jobs
  - Defines dependencies between jobs
- **The Oozie server submits the jobs to the server in the correct sequence**



Oozie is a workflow scheduler, schedules

Apache Sentry: Hadoop Security

- **Sentry provides fine-grained access control (authorization) to various Hadoop ecosystem components**
  - Impala
  - Hive
  - Cloudera Search
  - HDFS
- **In conjunction with Kerberos authentication, Sentry authorization provides a complete cluster security solution**
- **Created by Cloudera**
  - Now an open-source Apache project

Bibliography

The following offer more information on topics discussed in this chapter

- ***Hadoop: The Definitive Guide*** **(published by O'Reilly)**
  - `http://tiny.cloudera.com/hadooptdg`
- ***Cloudera Essentials for Apache Hadoop*** **– free online training**
  - `http://tiny.cloudera.com/esscourse`