Evolution of Open Source Data Infrastructure

Past, Present, and Future

Fangjin Yang
Cofounder @ imply

Overview

Simpler times with small data

The rise of open source

Current open source landscape

Where are we headed?

Data Insights

Broadly, we care about two use cases:

- OLTP
- OLAP

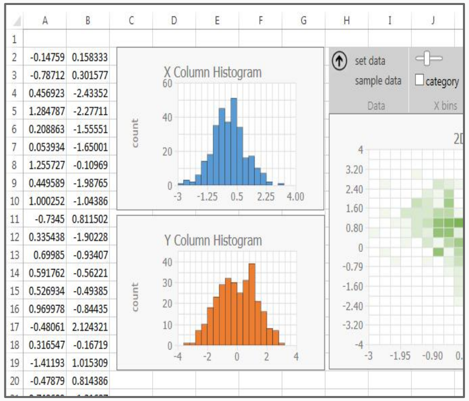
OLTP - business processing - dealing with transactions

OLAP - reporting - business intelligence

OLAP data - dimensions & measures

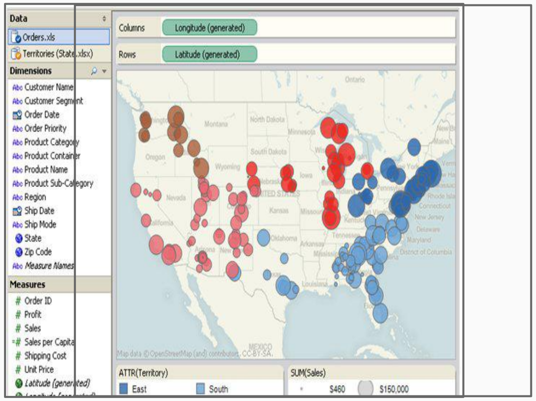
Small Data

Small Data Analytics



	A	B
1		
2	-0.14759	0.158333
3	-0.78712	0.301577
4	0.456923	-2.43352
5	1.284787	-2.27711
6	0.208863	-1.55551
7	0.053934	-1.65001
8	1.255727	-0.10969
9	0.449589	-1.98765
10	1.000252	-1.04386
11	-0.7345	0.811502
12	0.335438	-1.90228
13	0.69985	-0.93407
14	0.591762	-0.56221
15	0.526934	-0.49385
16	0.969978	-0.84435
17	-0.48061	2.124321
18	0.316547	-0.16719
19	-1.41193	1.015309
20	-0.47879	0.814386

Excel

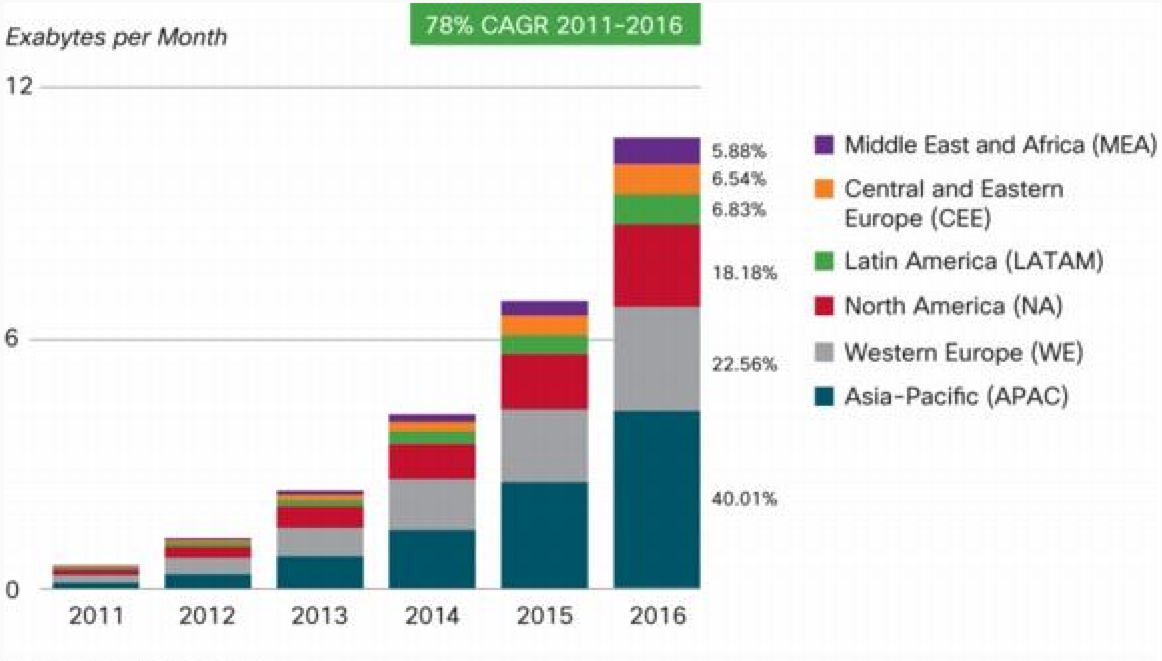


Tableau

+ MySQL®

- Solutions are very simple
- Fast and easy to extract insights
- Easy to create different custom visualizations

Data Growth



Source: Cisco VNI Mobile, 2012

MPP databases?

- Oracle, Teradata, IBM, Microsoft, etc.

Proprietary databases are expensive!

The Rise of Hadoop

Hadoop

Google GFS paper published in 2003

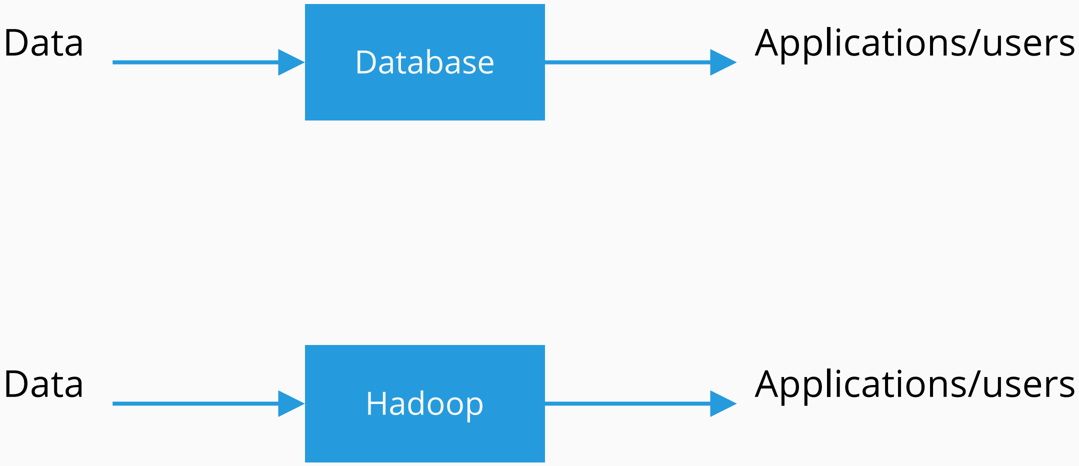
Google MapReduce paper published in 2004

Nutch project started in 2005 at Yahoo

Nutch became Hadoop and was open sourced in 2006

Community quickly grew

Early Open Source Stacks


Data

Database

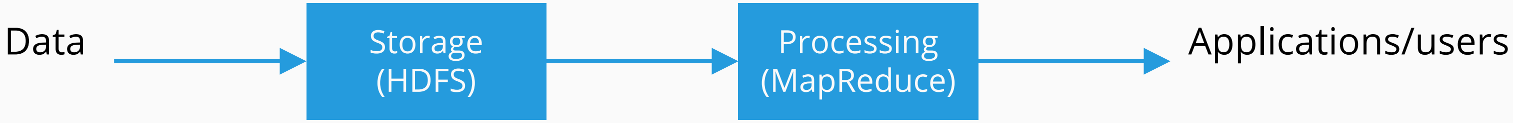
Applications/users

Data

Hadoop

Applications/users

Hadoop



Data

Storage (HDFS)

Processing (MapReduce)

Applications/users

Hadoop

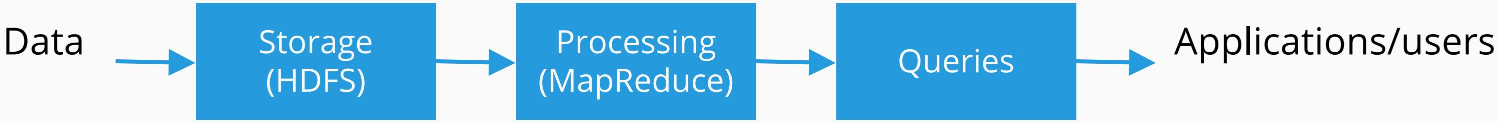
When one technology becomes very adopted, its limitations also become more well known

Hadoop is a very flexible solution

Most commonly used for data processing

Not optimized for many things - many inefficiencies!

Hadoop


Data

Storage (HDFS)

Processing (MapReduce)

Queries

Applications/users

Rise of Open Source Data Infrastructure

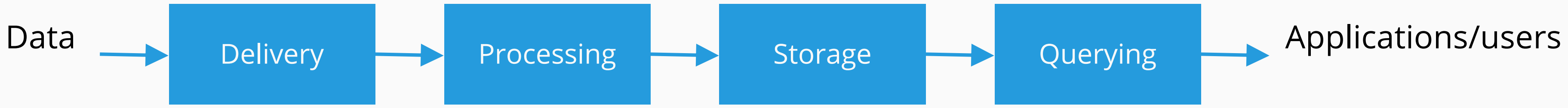
Things Hadoop isn’t good at:

- Fast queries
- Deliver (streams of) events
- Stream processing
- In-memory computation

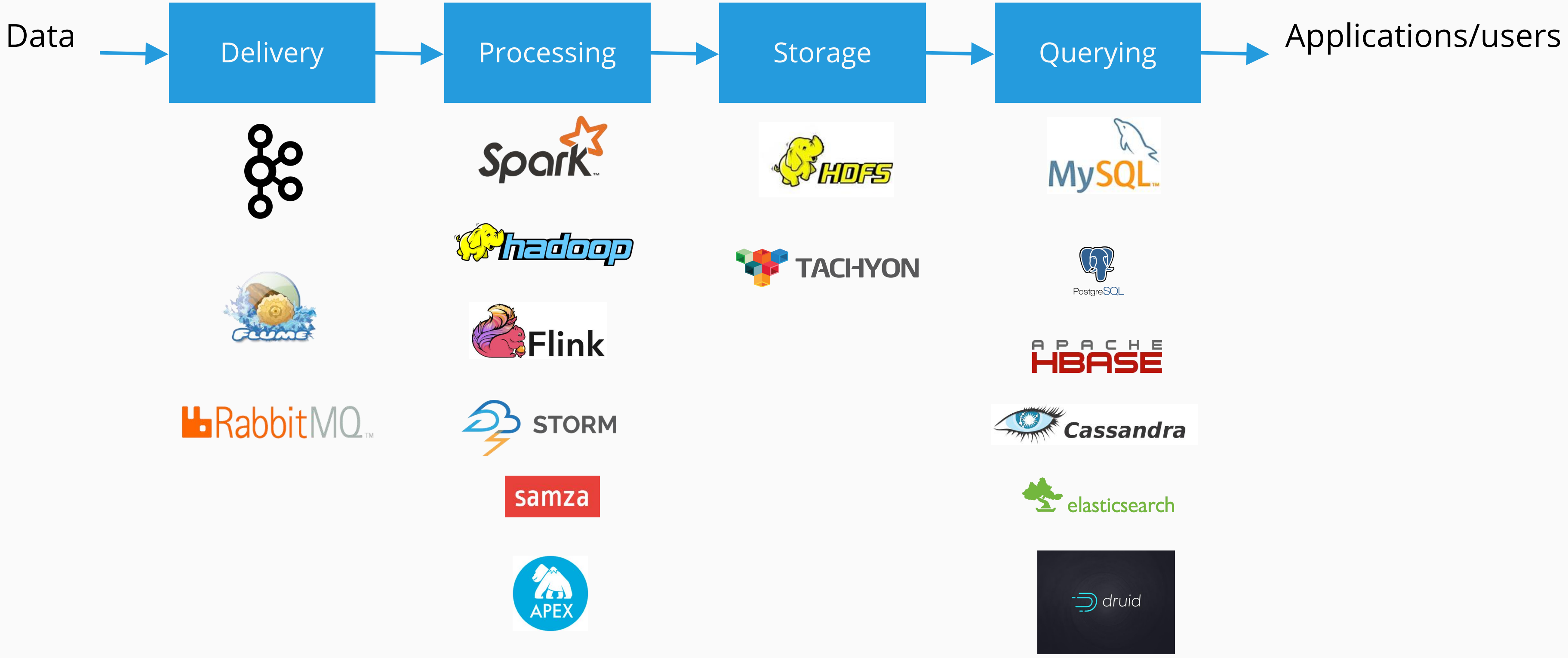
These limitations led to new technologies to be created

Data Infrastructure Space Today

Modern Open Source Stacks

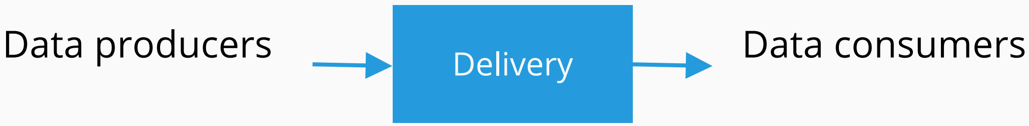


Modern Open Source Stacks



Data → Delivery → Processing → Storage → Querying → Applications/users

Data Delivery



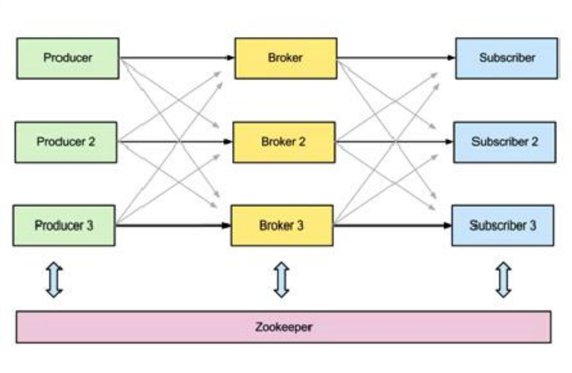
Data Delivery

Focus is storing data for a limited time and delivering it elsewhere

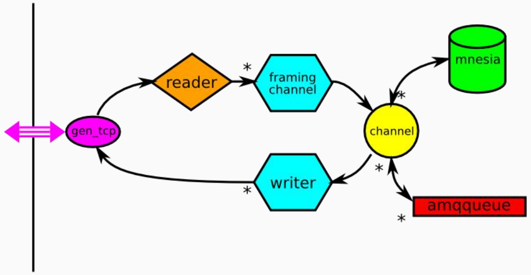
Three different approaches

- Apache Kafka - publish/subscribe, transaction queues
- RabbitMQ - publish/subscribe, distributed queues
- Apache Flume - push-based event delivery

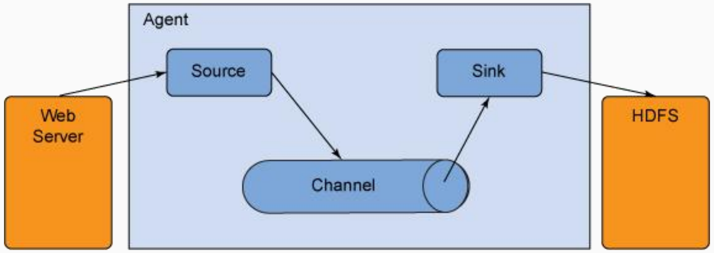
Data Delivery



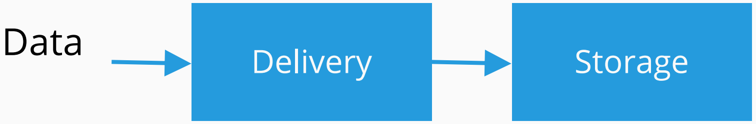
RabbitMQ™



FLUME



Storage



Storage

Distributed file systems

Store data indefinitely

Standard: HDFS

Can overlap with delivery systems (e.g. Kafka)

HDFS

ALLUXIO

Processing

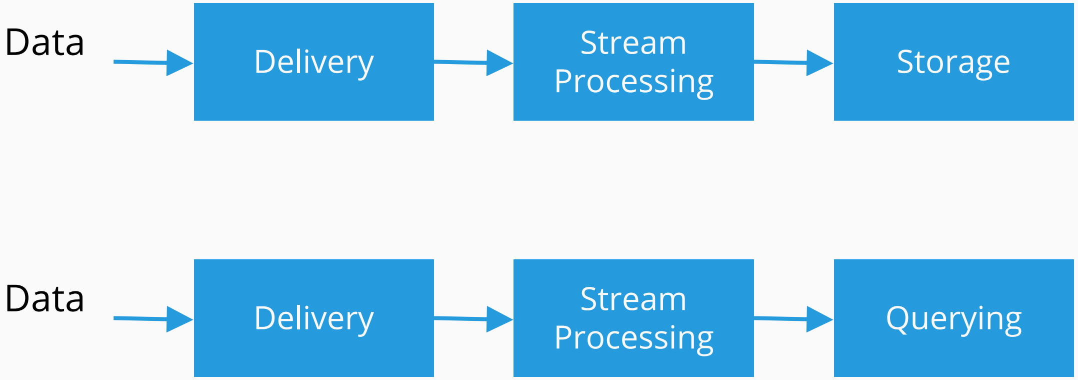
Processing systems are designed to transform data

Has overlap with querying systems

- Query systems: output set smaller than input set
- Processing systems: output set same size as input set
- Having separation is more standard nowadays

samza Spark hadoop Flink STORM APEX

Stream Processing



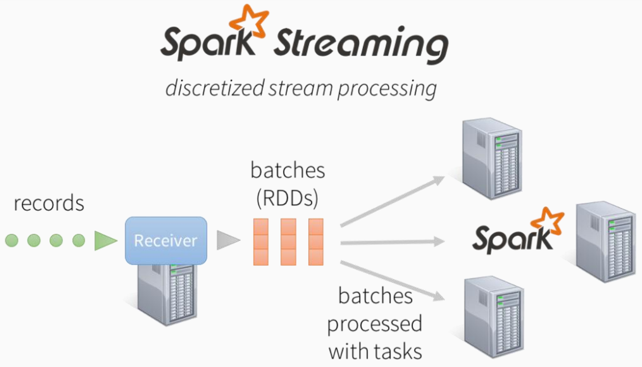
Stream Processing

Systems deal with unbounded messages/events

Different approaches

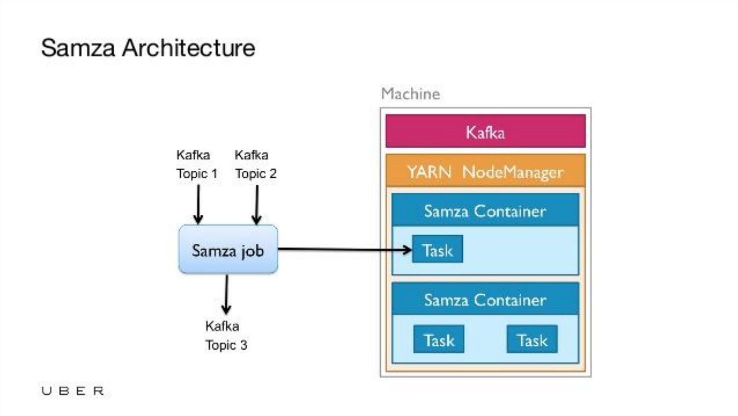
- Spark Streaming
- Storm
- Samza
- Flink
- Kafka Streams
- etc.

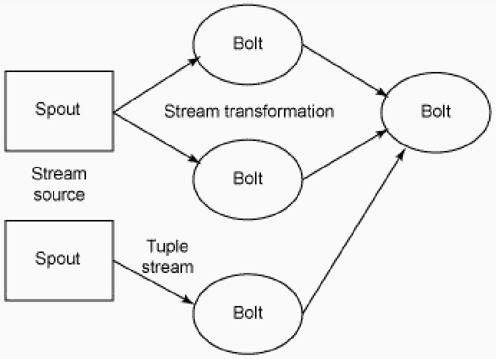
Stream Processing



records processed in batches with short tasks
each batch is a RDD (partitioned dataset)

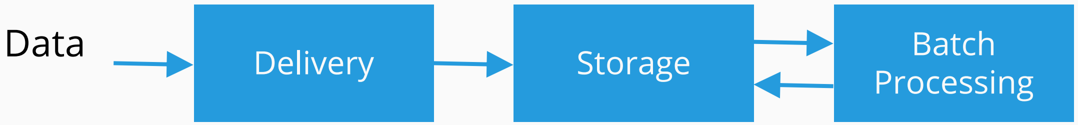
samza





STORM

Batch Processing



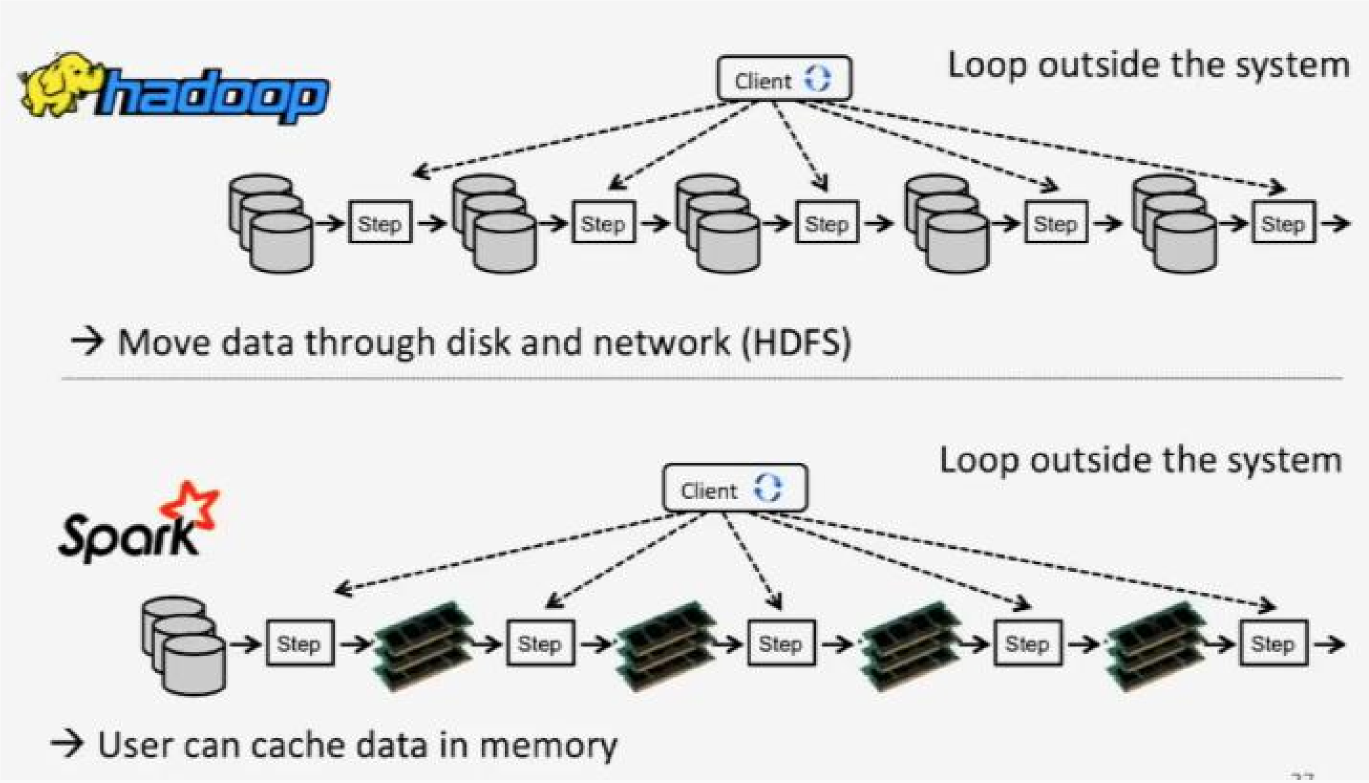
Batch Processing

Manipulate (large) static sets of data

Different approaches

- Spark
- Hadoop

Batch Processing



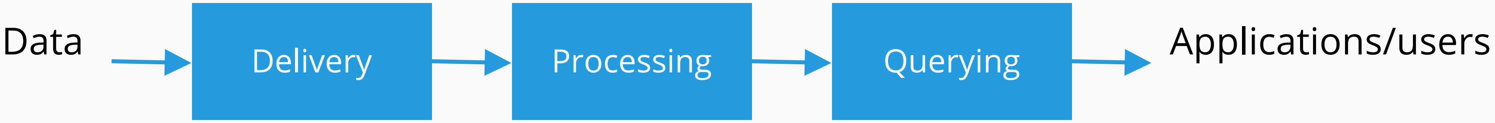
Loop outside the system

→ Move data through disk and network (HDFS)

Loop outside the system

→ User can cache data in memory

Querying

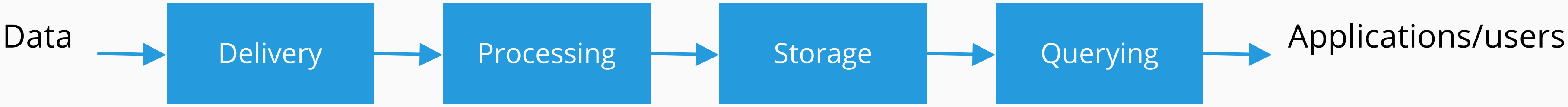

Data

Delivery

Processing

Querying

Applications/users


Data

Delivery

Processing

Storage

Querying

Applications/users

Querying

Largest and most complex (broad range of use cases)

Let's focus on the most common use case:

- Business intelligence/data warehousing/OLAP

Significant overlap with storage

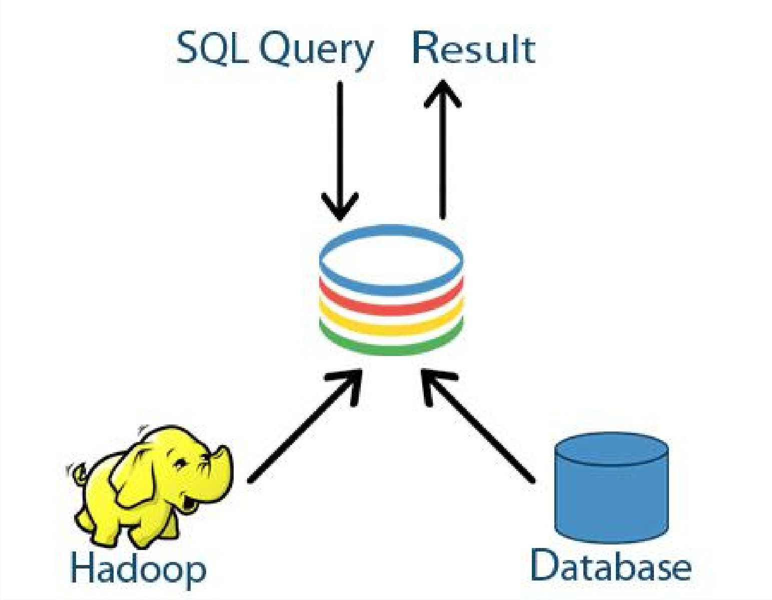
- Separation is becoming more common

SQL-on-Hadoop

Enable ad-hoc queries on different input formats

Examples: Impala, Hive, Spark SQL, Drill, Presto

SQL-on-Hadoop



SQL-on-Hadoop

Advantages:

- Flexible /w full SQL support

Disadvantages:

- Slow - serialization/deserialization can have overhead

Many new storage formats

- Apache Parquet, Apache Kudu, Apache Arrow, etc.

Key/Value Stores

Very fast writes

Very fast lookups

Timeseries databases often have K/V storage engines

APACHE HBASE

cassandra

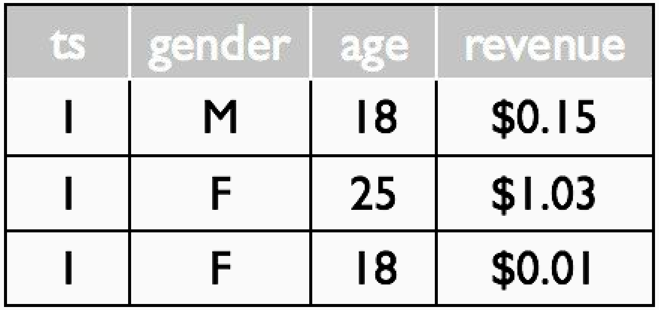
OPENTSDB

InfluxDB

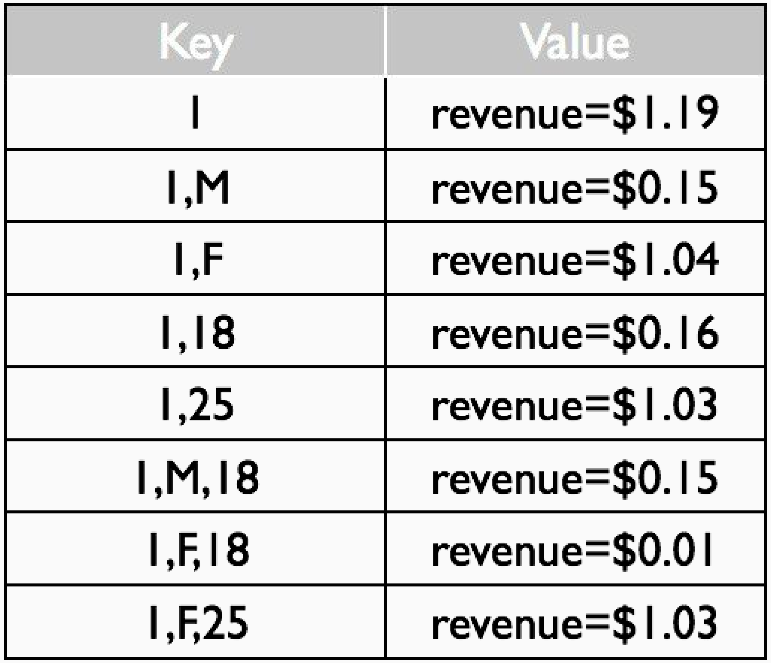
Key/Value Stores

Pre-computation

- Pre-compute every possible query
- Pre-compute a set of queries
- Exponential scaling costs



ts	gender	age	revenue
1	M	18	$0.15
1	F	25	$1.03
1	F	18	$0.01

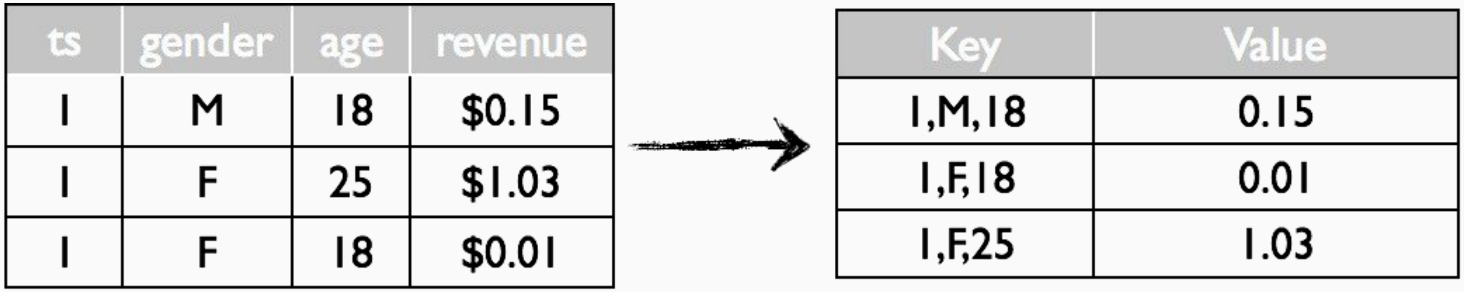


Key	Value
1	revenue=$1.19
1,M	revenue=$0.15
1,F	revenue=$1.04
1,18	revenue=$0.16
1,25	revenue=$1.03
1,M,18	revenue=$0.15
1,F,18	revenue=$0.01
1,F,25	revenue=$1.03

Key/Value Stores

Range scans

- Primary key: dimensions/attributes
- Value: measures/metrics (things to aggregate)
- Still too slow!



ts	gender	age	revenue
1	M	18	$0.15
1	F	25	$1.03
1	F	18	$0.01

Key	Value
1,M,18	0.15
1,F,18	0.01
1,F,25	1.03

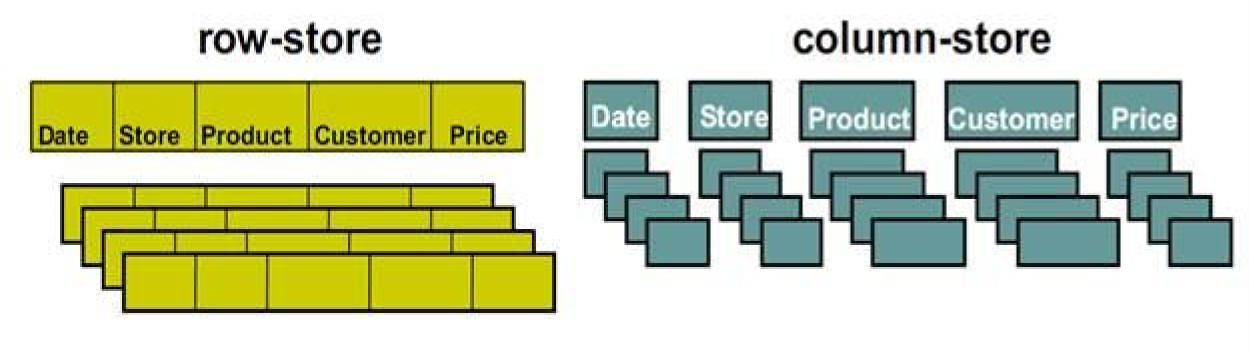
Column stores

Load/scan exactly what you need for a query

Different compression algorithms for different columns

- Encoding for string columns
- Compression for measure columns

Different indexes for different columns



Druid

Targeted for extremely low latency queries - powering user-facing analytic applications

Custom column format optimized for event data and BI queries

Supports lots of concurrent reads

Streaming data ingestion

So many choices!

Does the project solve your use case?

Is it stable? Cheap? Fast?

Is there an active and growing community?

10x faster or 10x cheaper -- upgrade!

The Next Few Years

General Trends

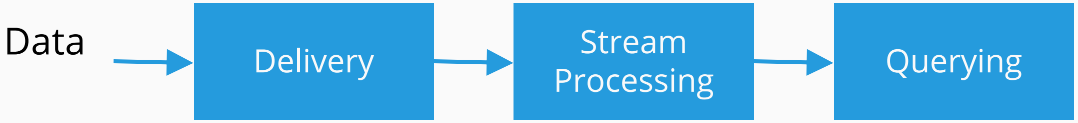
Number of projects reaching saturation point

Streaming computation

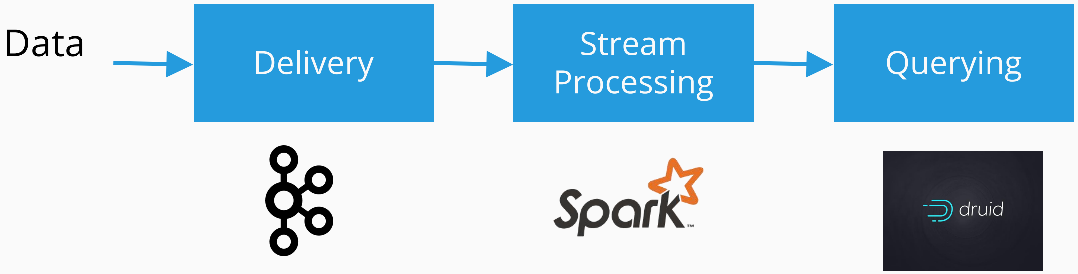
In-memory computation

Standards are slowly emerging

Future Open Source Stack?



Future Open Source Stack?



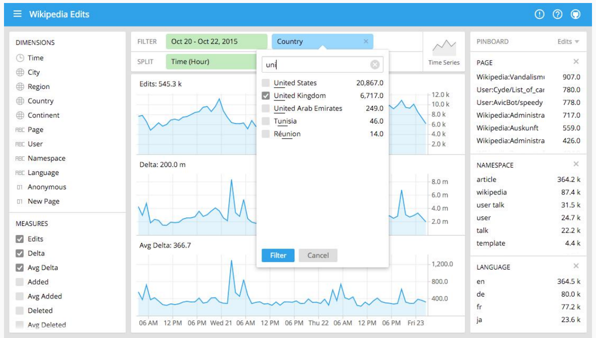
Data

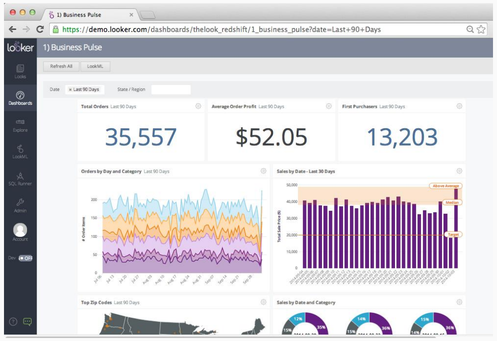
Delivery

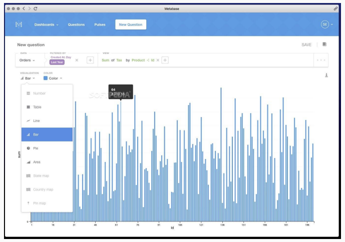
Stream Processing

Querying

Applications




1) Business Pulse
https://demo.looker.com/dashboards/thelook_redshift/1_business_pulse?date=Last+90+Days
looker
Total Orders
35,557
Average Order Profit
$52.05
First Purchasers
13,203
Orders by Day and Category
Sales by Date - Last 30 Days
Top Zip Codes
Sales by Date and Category



TensorFlow

DL4J Deep Learning for Java

scikit learn

Thanks!

imply.io
druid.io