11/16/2011, Stanford EE380 Computer Systems Colloquium

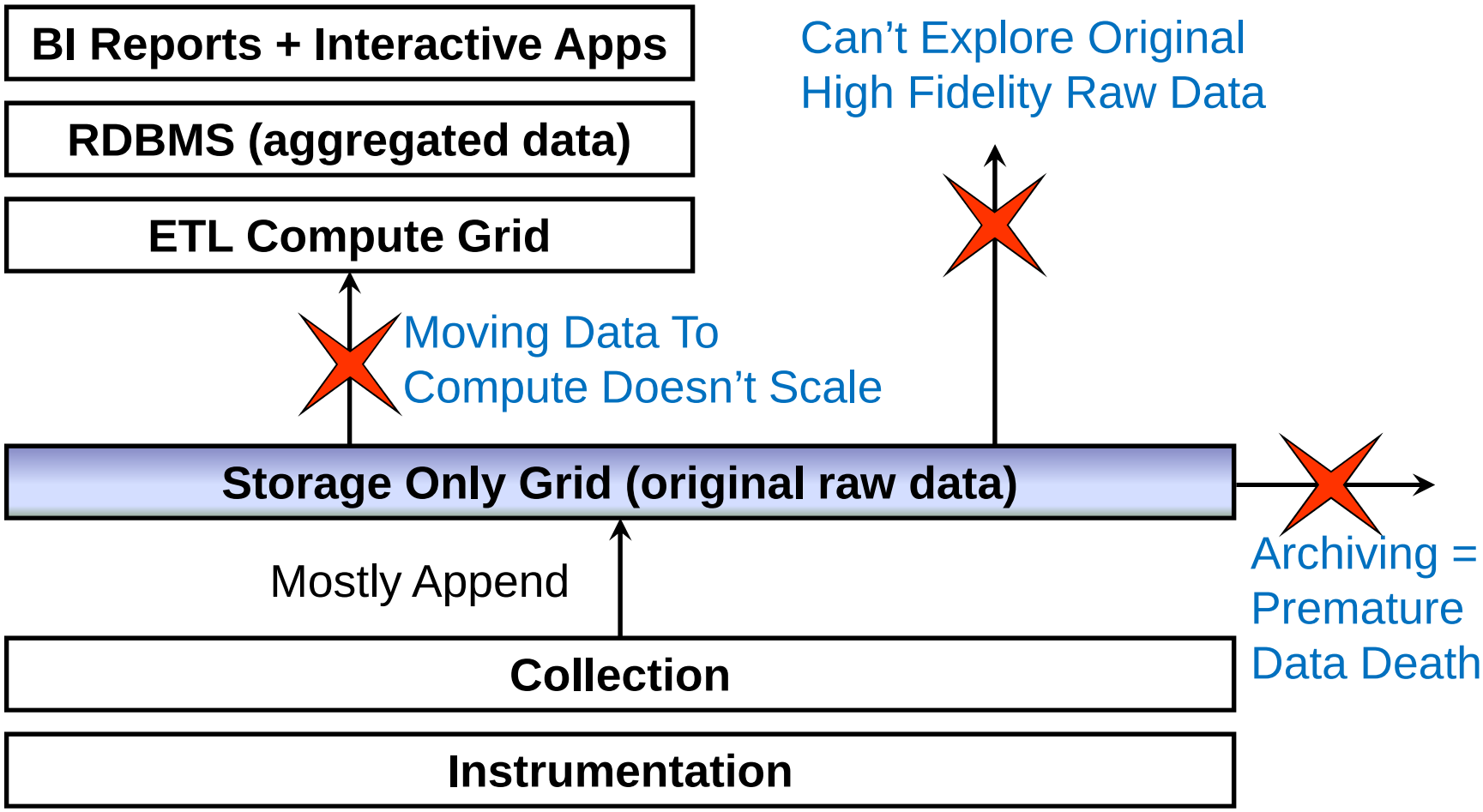
Introducing Apache Hadoop: The Modern Data Operating System

Dr. Amr Awadallah | Founder, CTO, VP of Engineering

aaa@cloudera.com, twitter: @awadallah

cloudera

Limitations of Existing Data Analytics Architecture



cloudera

So What is Apache  ?

- A scalable fault-tolerant distributed system for data storage and processing (open source under the Apache license).

- Core Hadoop has two main systems:
  - **Hadoop Distributed File System**: self-healing high-bandwidth clustered storage.
  - **MapReduce**: distributed fault-tolerant resource management and scheduling coupled with a scalable data programming abstraction.

The Key Benefit: Agility/Flexibility

Schema-on-Write (RDBMS):

- Schema must be created before any data can be loaded.
- An explicit load operation has to take place which transforms data to DB internal structure.
- New columns must be added explicitly before new data for such columns can be loaded into the database.

Schema-on-Read (Hadoop):

- Data is simply copied to the file store, no transformation is needed.
- A SerDe (Serializer/Deserlizer) is applied during read time to extract the required columns (*late binding*)
- New data can start flowing anytime and will appear retroactively once the SerDe is updated to parse it.

- Read is Fast
- Standards/Governance

- Load is Fast
- Flexibility/Agility

cloudera

Innovation: Explore Original Raw Data

Data Committee

Data Scientist

cloudera

Flexibility: Complex Data Processing

1. **Java MapReduce:** Most flexibility and performance, but tedious development cycle (the *assembly language* of Hadoop).
2. **Streaming MapReduce (**aka **Pipes):** Allows you to develop in any programming language of your choice, but slightly lower performance and less flexibility than native Java MapReduce.
3. **Crunch:** A library for multi-stage MapReduce pipelines in Java (modeled After Google's FlumeJava)
4. **Pig Latin:** A high-level language out of Yahoo, suitable for batch data flow workloads.
5. **Hive:** A SQL interpreter out of Facebook, also includes a meta-store mapping files to their schemas and associated SerDes.
6. **Oozie:** A PDL XML workflow engine that enables creating a workflow of jobs composed of any of the above.

Scalability: Scalable Software Development

Grows without requiring developers to re-architect their algorithms/application.

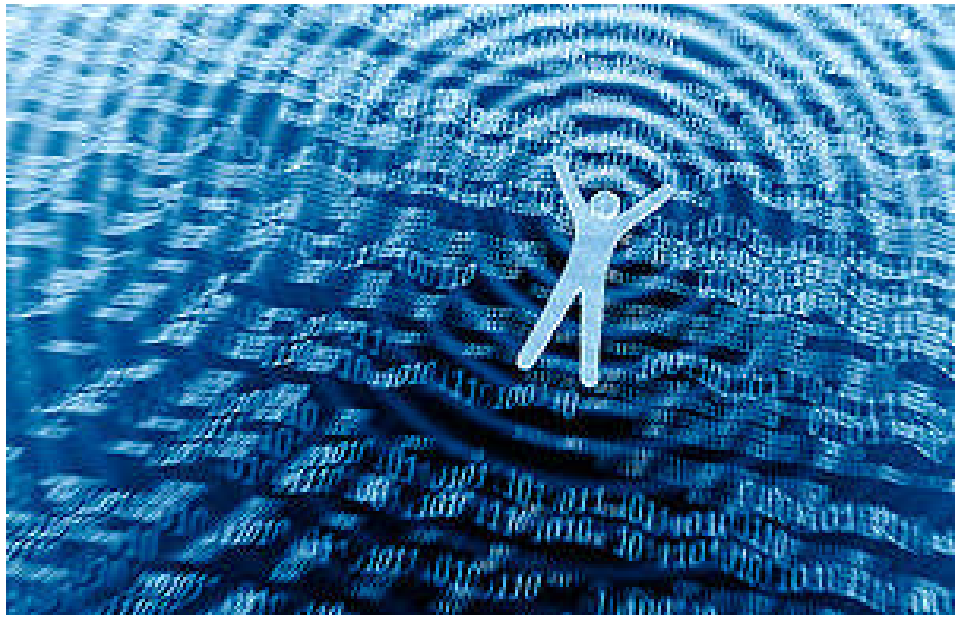
AUTO SCALE

cloudera

Scalability: Data Beats Algorithm

Smarter Algos

More Data



A. Halevy et al, “The Unreasonable Effectiveness of Data”, IEEE Intelligent Systems, March 2009

cloudera

Scalability: Keep All Data Alive Forever

Archive to Tape and Never See It Again

Extract Value From All Your Data

cloudera

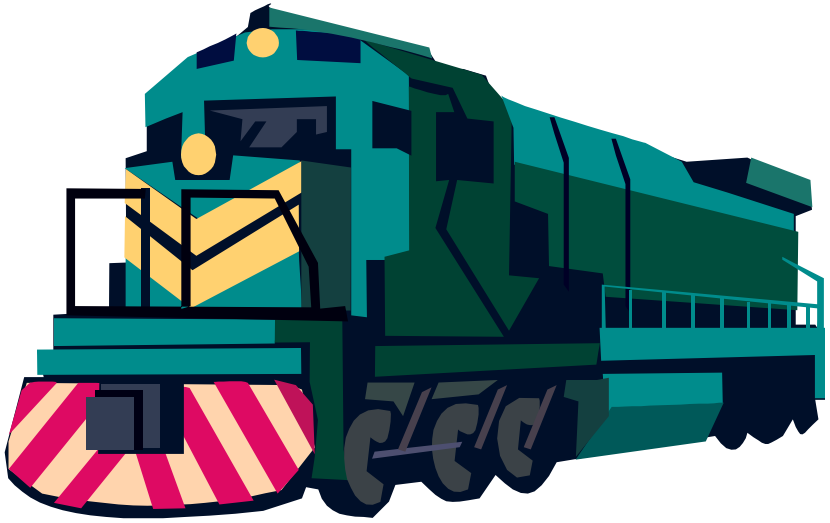
Use The Right Tool For The Right Job

Relational Databases:

Use when:

- **Interactive OLAP Analytics (<1sec)**
- **Multistep ACID Transactions**
- **100% SQL Compliance**

Hadoop:



Use when:

- **Structured or Not (Flexibility)**
- **Scalability of Storage/Compute**
- **Complex Data Processing**

cloudera

HDFS: Hadoop Distributed File System

A given file is broken down into blocks (default=64MB), then blocks are replicated across cluster (default=3).

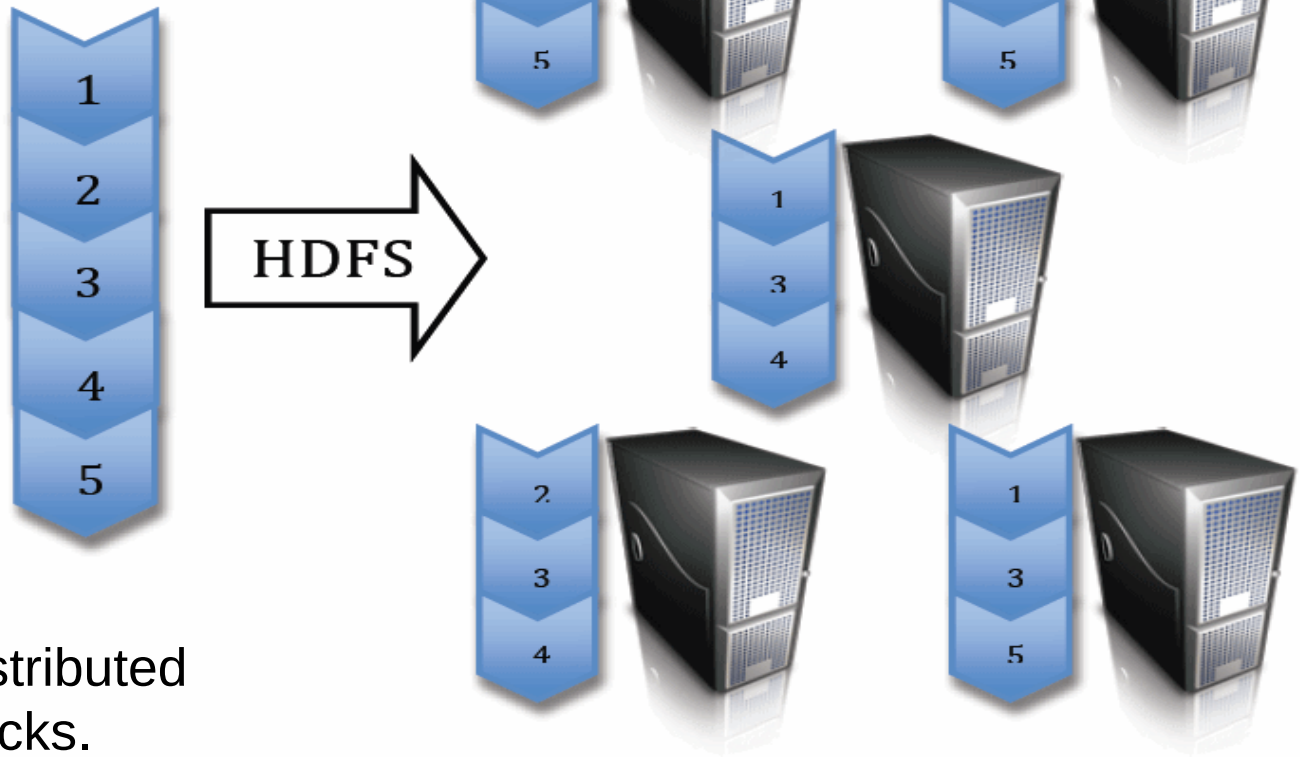
Optimized for:

- Throughput
- Put/Get/Delete
- Appends

Block Replication for:

- Durability
- Availability
- Throughput

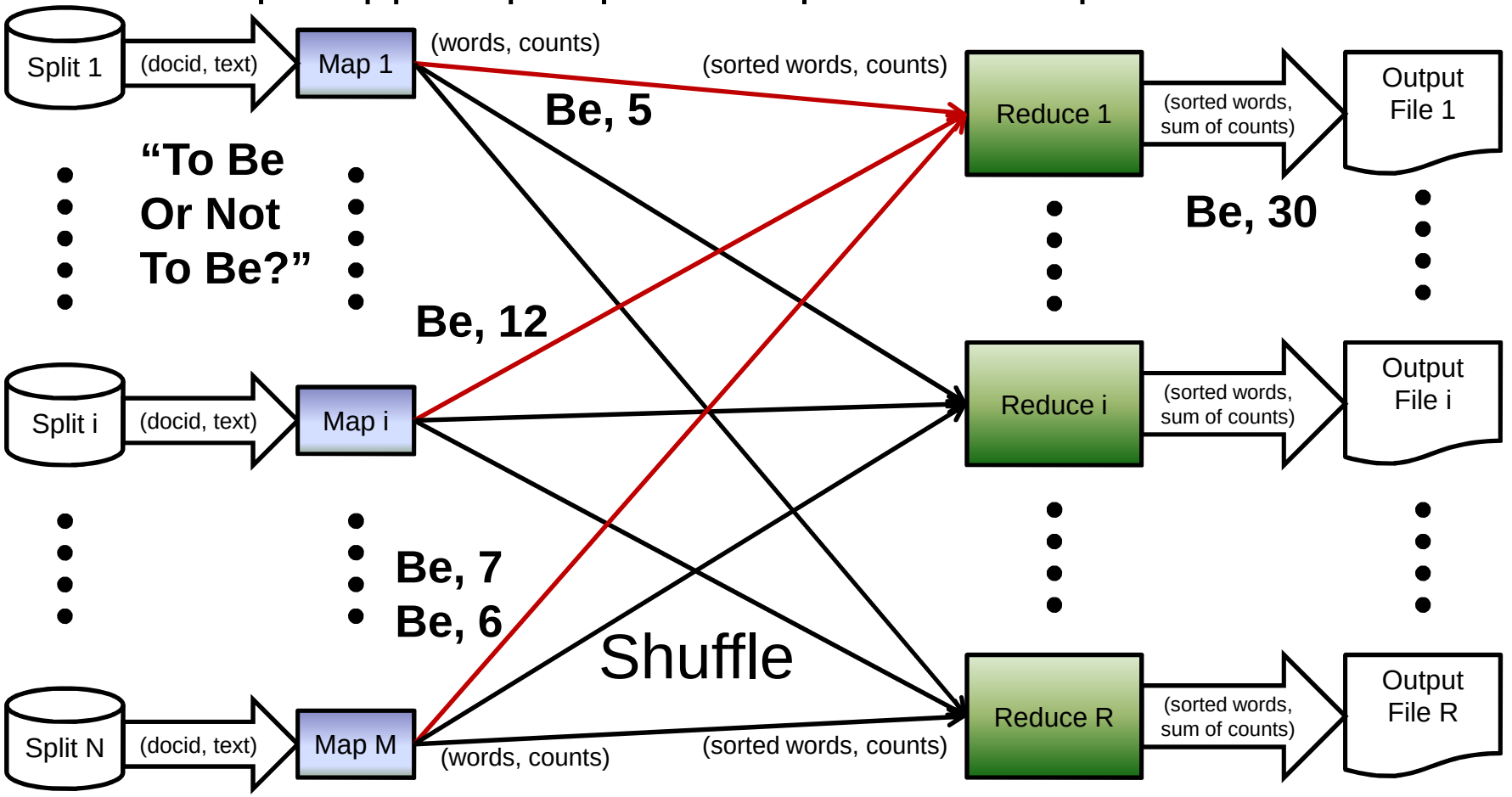
Block Replicas are distributed across servers and racks.



cloudera

MapReduce: Computational Framework

```
cat *.txt | mapper.pl | sort | reducer.pl > out.txt
```



Map(in_key, in_value) => list of (out_key, intermediate_value)

Reduce(out_key, list of intermediate_values) => out_value(s)

cloudera

MapReduce: Resource Manager / Scheduler

A given job is broken down into tasks, then tasks are scheduled to be as close to data as possible.

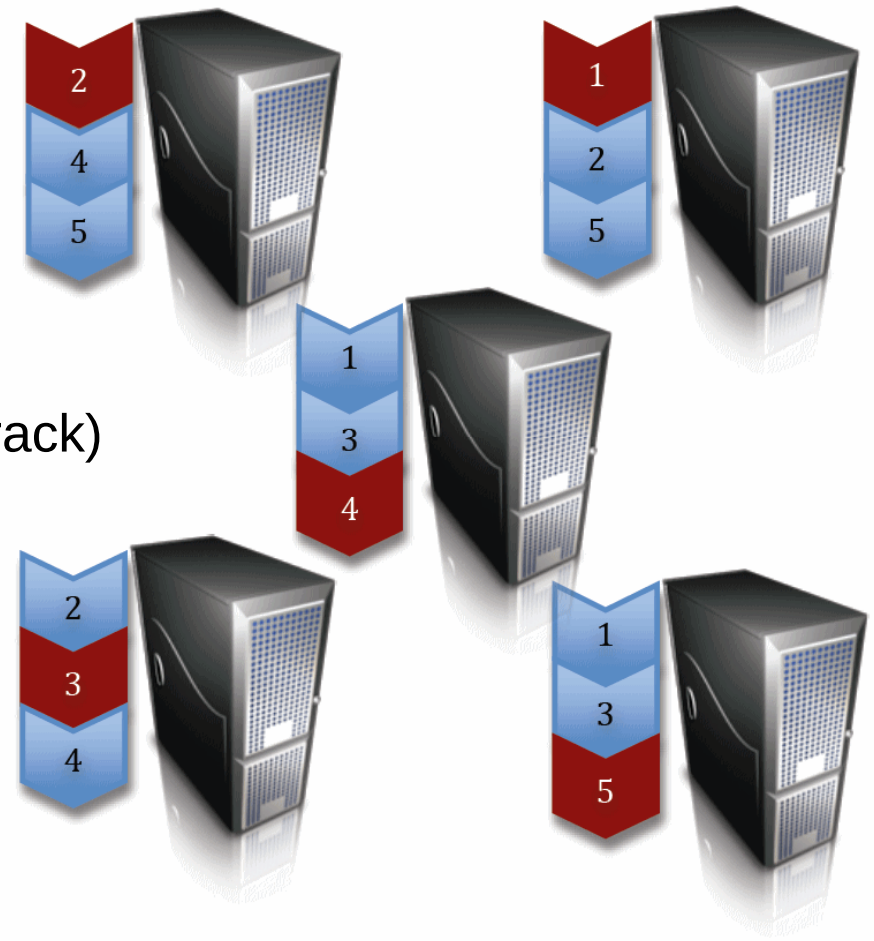
Three levels of data locality:

- Same server as data (local disk)
- Same rack as data (rack/leaf switch)
- Wherever there is a free slot (cross rack)

Optimized for:

- Batch Processing
- Failure Recovery

System detects *laggard* tasks and speculatively executes parallel tasks on the same slice of data.



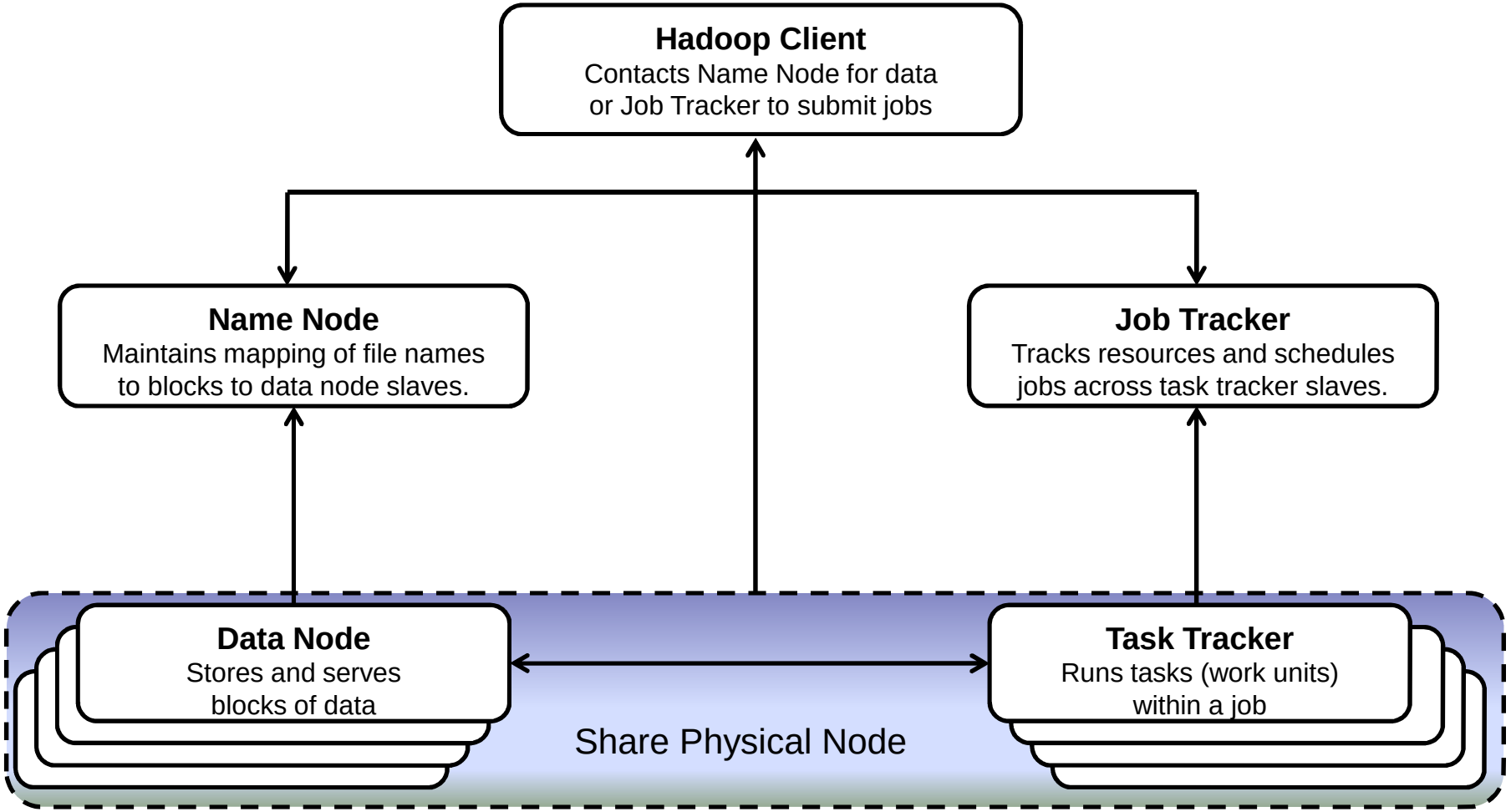
cloudera

But Networks Are Faster Than Disks!

Yes, however, core and disk density per server are going up very quickly:

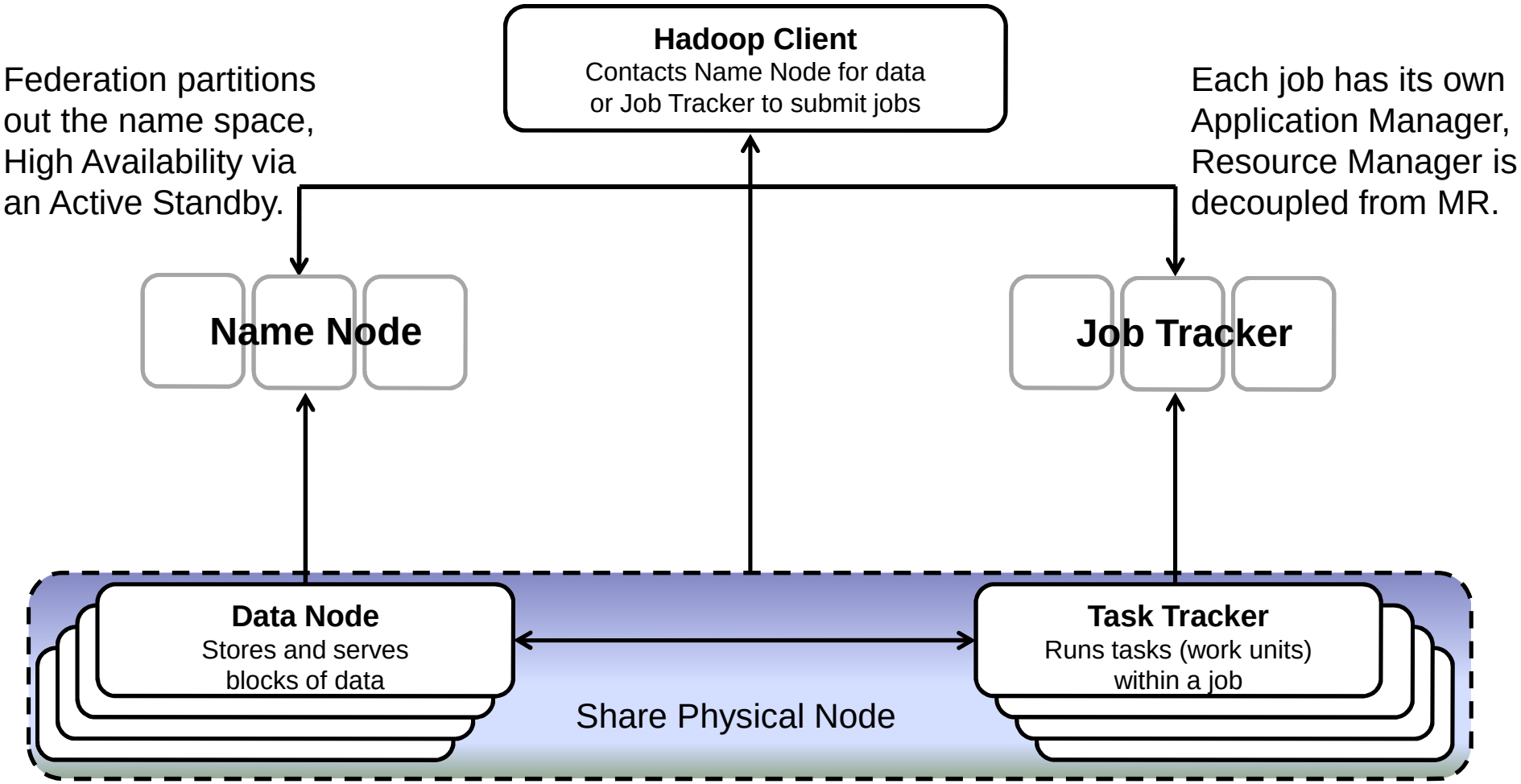
- 1 Hard Disk = 100MB/sec (~1Gbps)
- Server = 12 Hard Disks = 1.2GB/sec (~12Gbps)
- Rack = 20 Servers = 24GB/sec (~240Gbps)
- Avg. Cluster = 6 Racks = 144GB/sec (~1.4Tbps)
- Large Cluster = 200 Racks = 4.8TB/sec (~48Tbps)
- Scanning 4.8TB at 100MB/sec takes 13 hours.

Hadoop High-Level Architecture



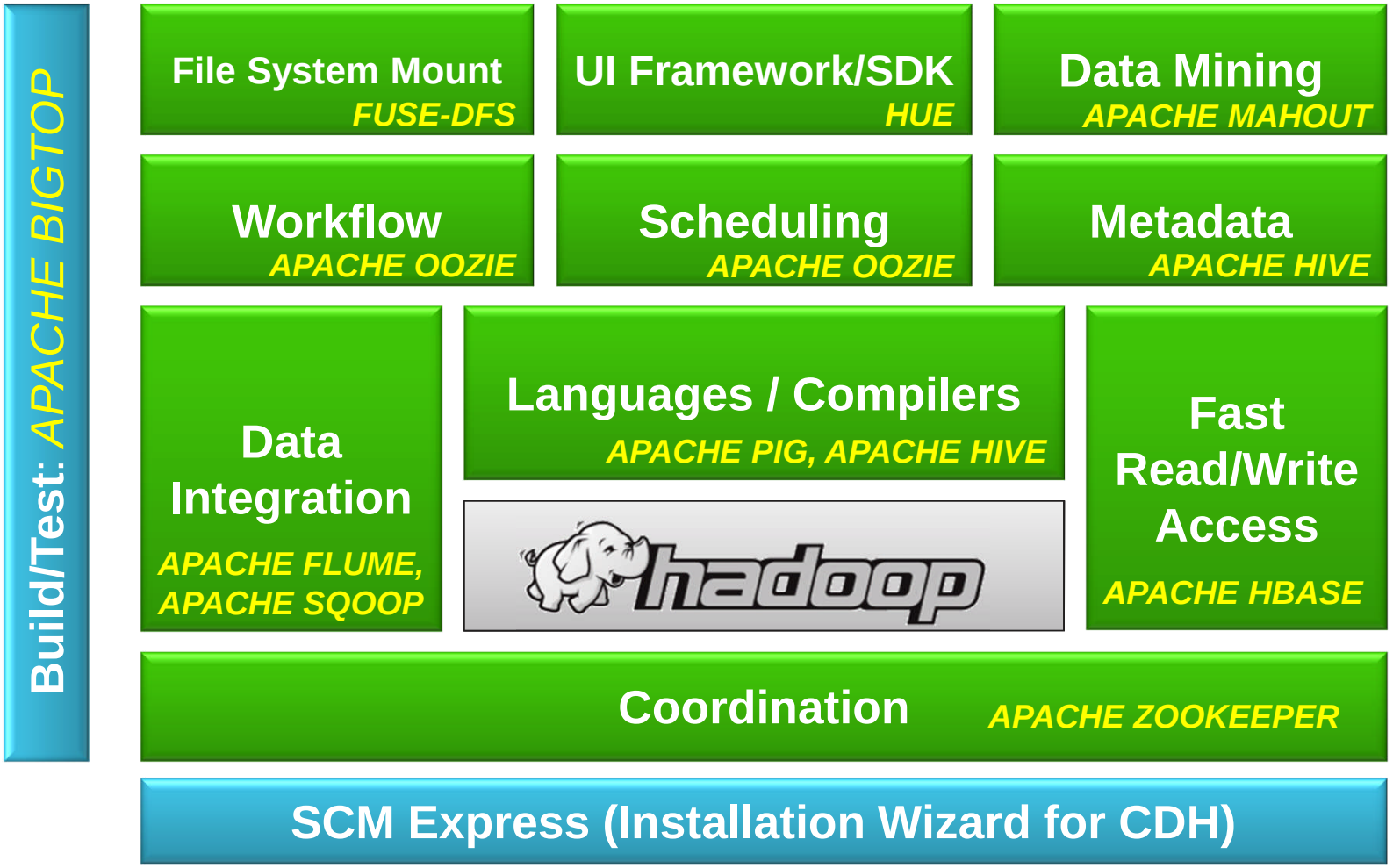
cloudera

Changes for Better Availability/Scalability



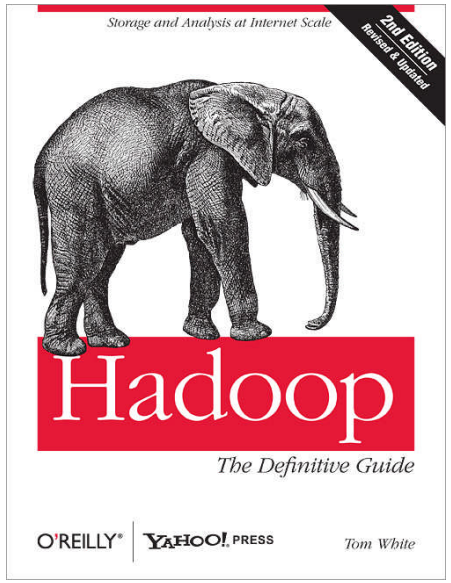
cloudera

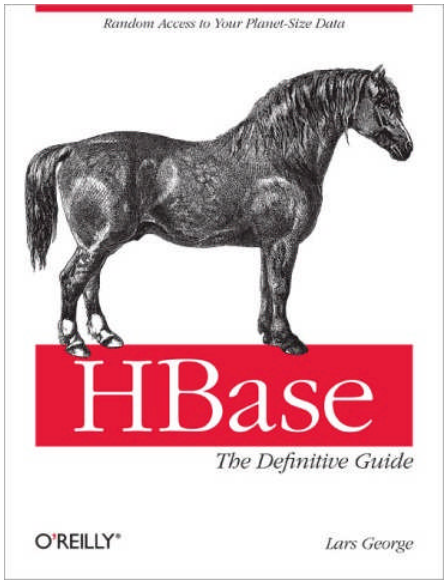
CDH: Cloudera's Distribution Including Apache Hadoop

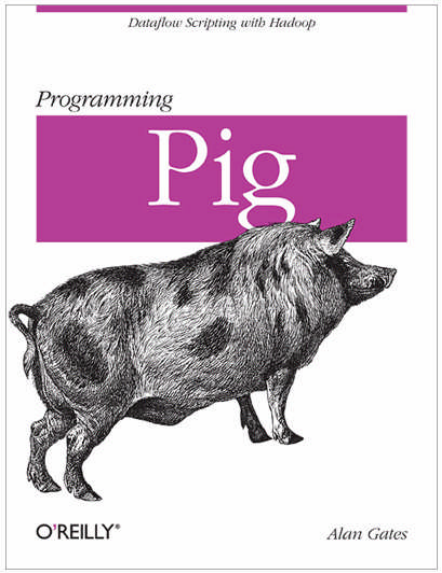


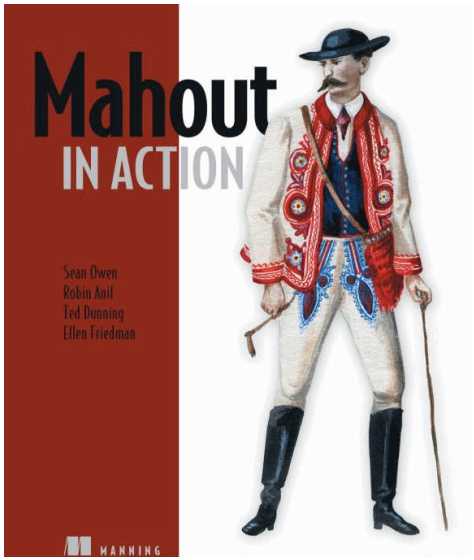
cloudera

Books









cloudera

Conclusion

- **The Key Benefits of Apache Hadoop:**
  - **Agility/Flexibility** (Quickest Time to Insight).
  - **Complex Data Processing** (Any Language, Any Problem).
  - **Scalability of Storage/Compute** (Freedom to Grow).
  - **Economical Storage** (Keep All Your Data Alive Forever).

- **The Key Systems for Apache Hadoop are:**
  - **Hadoop Distributed File System**: self-healing high-bandwidth clustered storage.
  - **MapReduce**: distributed fault-tolerant resource management coupled with scalable data processing.

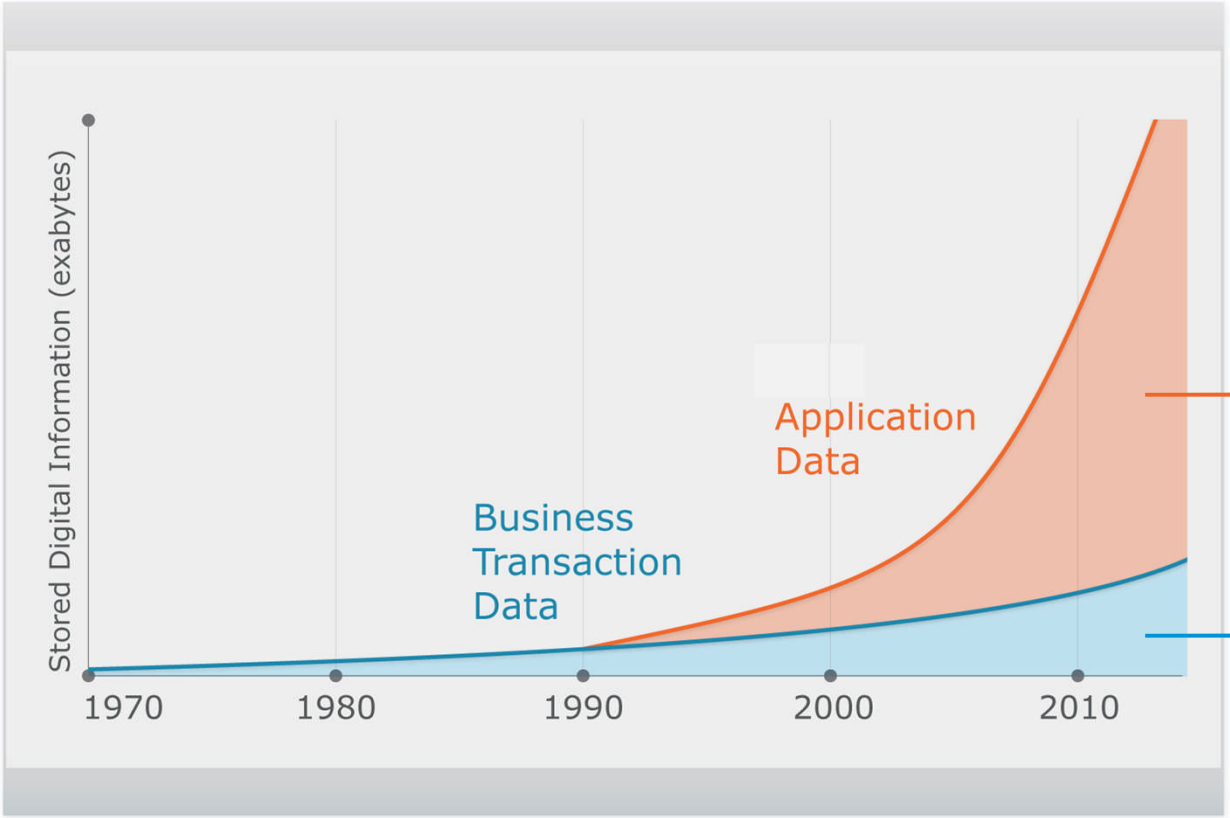
cloudera

Appendix

BACKUP SLIDES

cloudera

Unstructured Data is Exploding



- 2,500 exabytes of new information in 2012 with Internet as primary driver
- Digital universe grew by 62% last year to 800K petabytes and will grow to 1.2 "zettabytes" this year

Source: IDC White Paper - sponsored by EMC.
As the Economy Contracts, the Digital Universe Expands. May 2009.

cloudera

Hadoop Creation History

Doug Cutting adds DFS & MapReduce support to Nutch

Doug Cutting & Mike Cafarella started working on Nutch

Google publishes GFS & MapReduce papers

Yahoo! hires Cutting, Hadoop spins out of Nutch

NY Times converts 4TB of image archives over 100 EC2s

Facebooks launches Hive: SQL Support for Hadoop

Fastest sort of a TB, 3.5mins over 910 nodes

cloudera Founded

- Fastest sort of a TB, 62secs over 1,460 nodes
- Sorted a PB in 16.25hours over 3,658 nodes

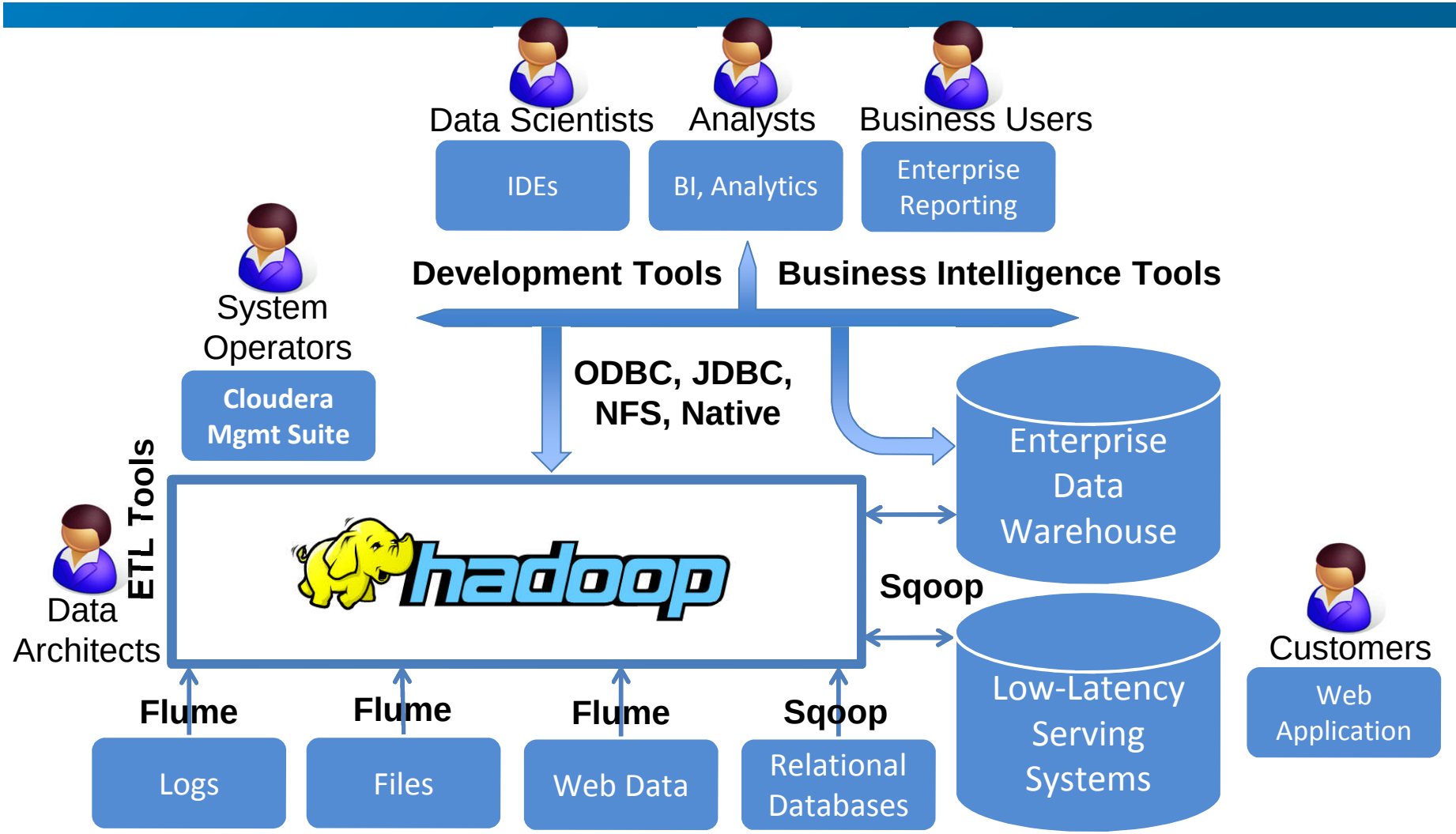
Hadoop Summit 2009, 750 attendees

Doug Cutting joins Cloudera

2002 2003 2004 2005 2006 2007 2008 2009

cloudera

Hadoop in the Enterprise Data Stack



cloudera

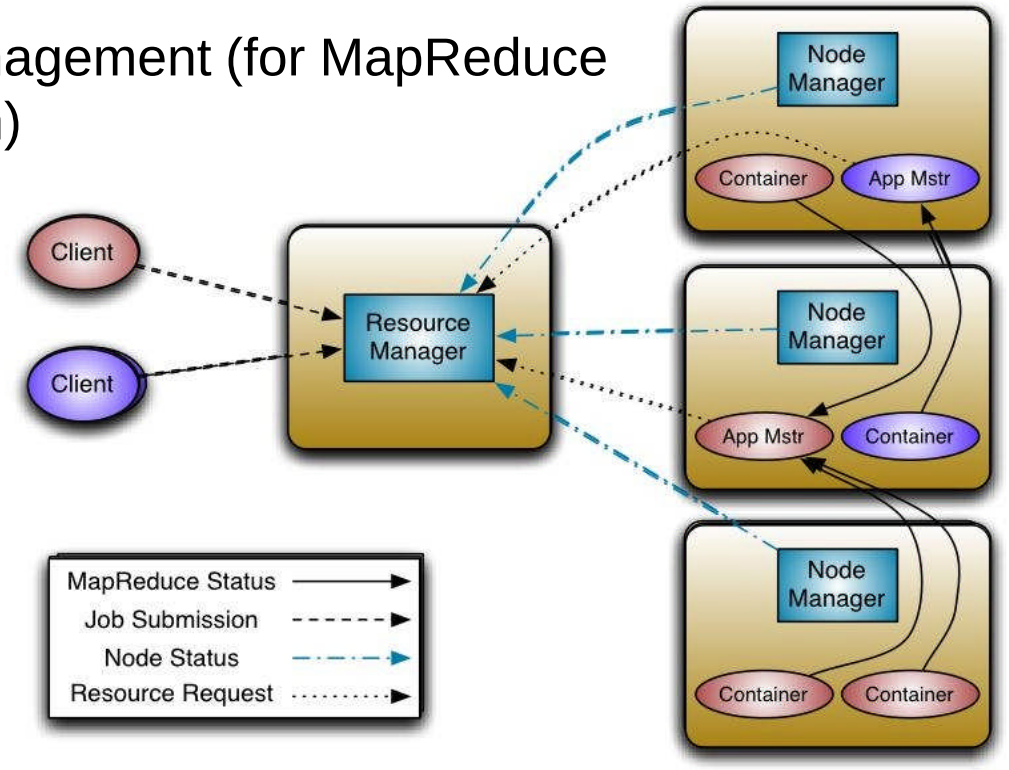
MapReduce Next Gen

Main idea is to split up the JobTracker functions:

- Cluster resource management (for tracking and allocating nodes)
- Application life-cycle management (for MapReduce scheduling and execution)

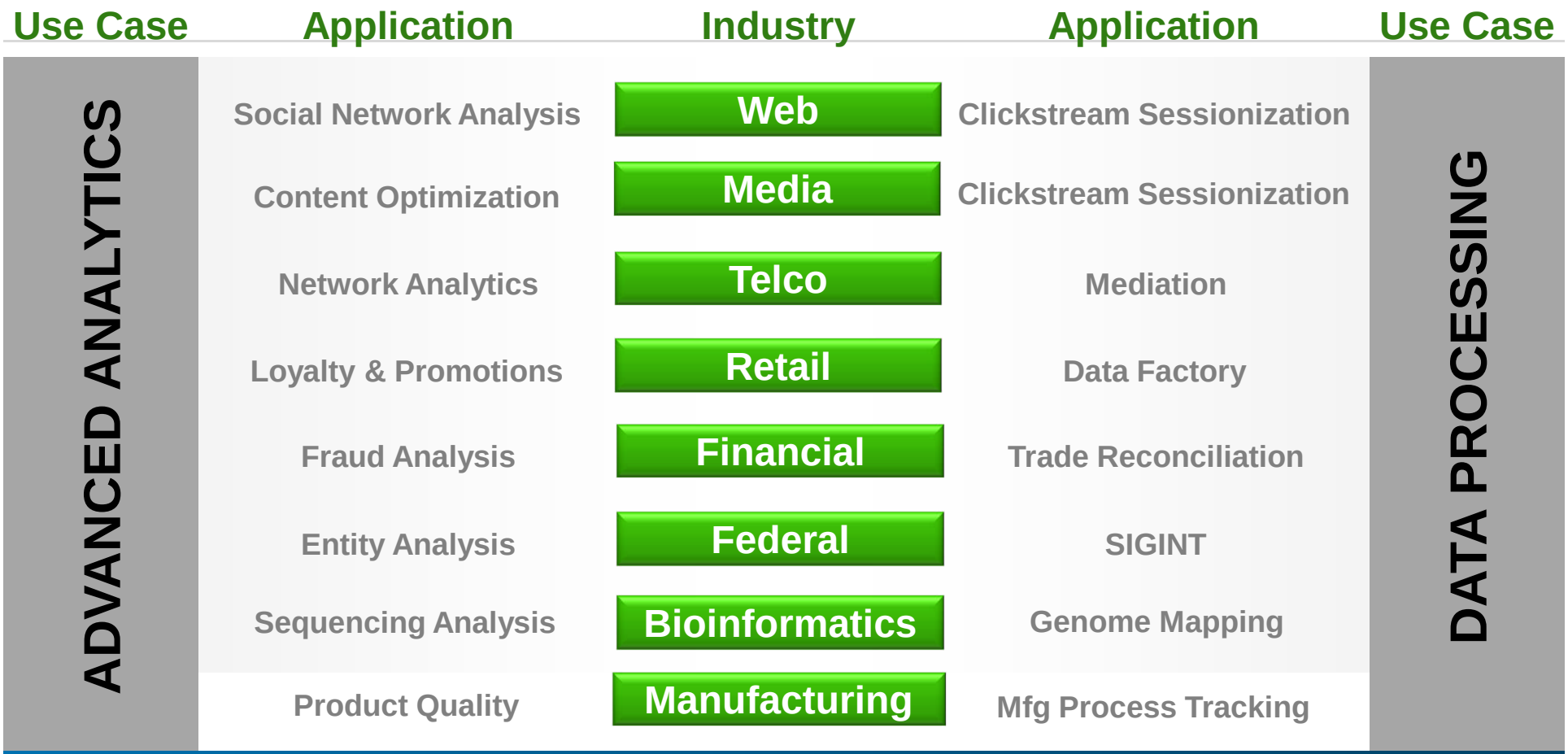
Enables:

- High Availability
- Better Scalability
- Efficient Slot Allocation
- Rolling Upgrades
- Non-MapReduce Apps



cloudera

Two Core Use Cases Common Across Many Industries



cloudera

What is Cloudera Enterprise?

Cloudera Enterprise makes open source Apache Hadoop enterprise-easy

- ✓ Simplify and Accelerate Hadoop Deployment
- ✓ Reduce Adoption Costs and Risks
- ✓ Lower the Cost of Administration
- ✓ Increase the Transparency & Control of Hadoop
- ✓ Leverage the Experience of Our Experts

CLOUDERA ENTERPRISE COMPONENTS

Cloudera Management Suite

Comprehensive Toolset for Hadoop Administration

+

Production-Level Support

Our Team of Experts On-Call to Help You Meet Your SLAs

3 of the top 5 telecommunications, mobile services, defense & intelligence, banking, media and retail organizations depend on Cloudera

EFFECTIVENESS

Ensuring Repeatable Value from Apache Hadoop Deployments

EFFICIENCY

Enabling Apache Hadoop to be Affordably Run in Production

cloudera

Hive vs Pig Latin (count distinct values > 0)

- Hive Syntax:

```
SELECT COUNT(DISTINCT col1)
FROM mytable
WHERE col1 > 0;
```

- Pig Latin Syntax:

```
mytable = LOAD 'myfile' AS (col1, col2, col3);
mytable = FOREACH mytable GENERATE col1;
mytable = FILTER mytable BY col1 > 0;
mytable = DISTINCT col1;
mytable = GROUP mytable BY col1;
mytable = FOREACH mytable GENERATE COUNT(mytable);
DUMP mytable;
```


cloudera

Apache Hive Key Features

- A subset of SQL covering the most common statements
- JDBC/ODBC support
- Agile data types: Array, Map, Struct, and JSON objects
- Pluggable SerDe system to work on unstructured files directly
- User Defined Functions and Aggregates
- Regular Expression support
- MapReduce support
- Partitions and Buckets (for performance optimization)
- Microstrategy/Tableau Compatibility (through ODBC)
- In The Works: Indices, Columnar Storage, Views, Explode/Collect
- More details: http://wiki.apache.org/hadoop/Hive

Hive Agile Data Types

- STRUCTS:
  - **`SELECT mytable.mycolumn.myfield FROM …`**
- MAPS (Hashes):
  - **`SELECT mytable.mycolumn[mykey] FROM …`**
- ARRAYS:
  - **`SELECT mytable.mycolumn[5] FROM …`**
- JSON:
  - **`SELECT get_json_object(mycolumn, objpath) FROM …`**

cloudera


cloudera