

HadoopDB

Need 4 Speed in Analytics

Azza Abouzeid, Kamil Bajda-Pawlikowski, Justin Borgman
Daniel J. Abadi, Avi Silberschatz

Yale University
<http://www.hadoopdb.net>

May 4, 2010

Major Trends

- ① Data explosion
 - Automation of business processes, proliferation of digital devices.
 - Facebook grew from 200GB/day in March 2008 to 15TB/day in Jan 2010. A growth factor of 75!!
 - eBay has a 6.5 PB warehouse. Yahoo! Everest has 10 PB.
- ② Analysis over raw data
 - Analysis over pre-aggregated data not enough
 - Complex analytics

The Ideal Data Analysis Solution

Key Features:

1 Performance

- *“With my current workload, I want to get information faster”*

2 Scalability

- *“As data and processing demands increase, I want to scale linearly by throwing in more nodes”*
- *“As I grow to hundreds of cheap nodes, it should tolerate failure”*

3 Flexibility

- *“I want to express complex analytical tasks as naturally as possible: SQL or MapReduce”*

4 Price

- *“It has to be affordable”*
- *“It needs to work on the cloud”*

MapReduce (Hadoop) & Parallel DBMS

MapReduce is a programming model which specifies:

- A **map** function that processes a key/value pair to generate a set of intermediate key/value pairs,
- A **reduce** function that merges all intermediate values associated with the same intermediate key.

Hadoop is the open source MapReduce implementation

Parallel DBMS

- Process partitioned, structured, relational data sets.
- SQL query transformed into relational operator pipelines: filter, **project**, **join**, **aggregate**, ...

Differences between Parallel Databases and Hadoop

Property	Parallel DBMS	MapReduce	Affects
Data model	Relational	Unstructured key-value	Performance
Checkpointing	Pipelines tuples between operators	Materializes intermediate results	Performance, Scalability
Fault-tolerance scheme	Entire query re-executed	Re-executes failed tasks only not entire job	Scalability
Scheduling	Pre-planned execution	Runtime Scheduler	Cloud-suitability
Query interface	SQL and UDF support	MR programs in a variety of languages and SQL support with Hive	Flexibility

To summarize

	Scalability*	High Performance**	Flexibility	Cloud Suitability	Price
MapReduce	✓	✗	~	✓	✓
Parallel Databases	✗	✓	~	~	✗
What we need	✓	✓	✓	✓	✓

* 1000s of nodes

** Queries on structured data

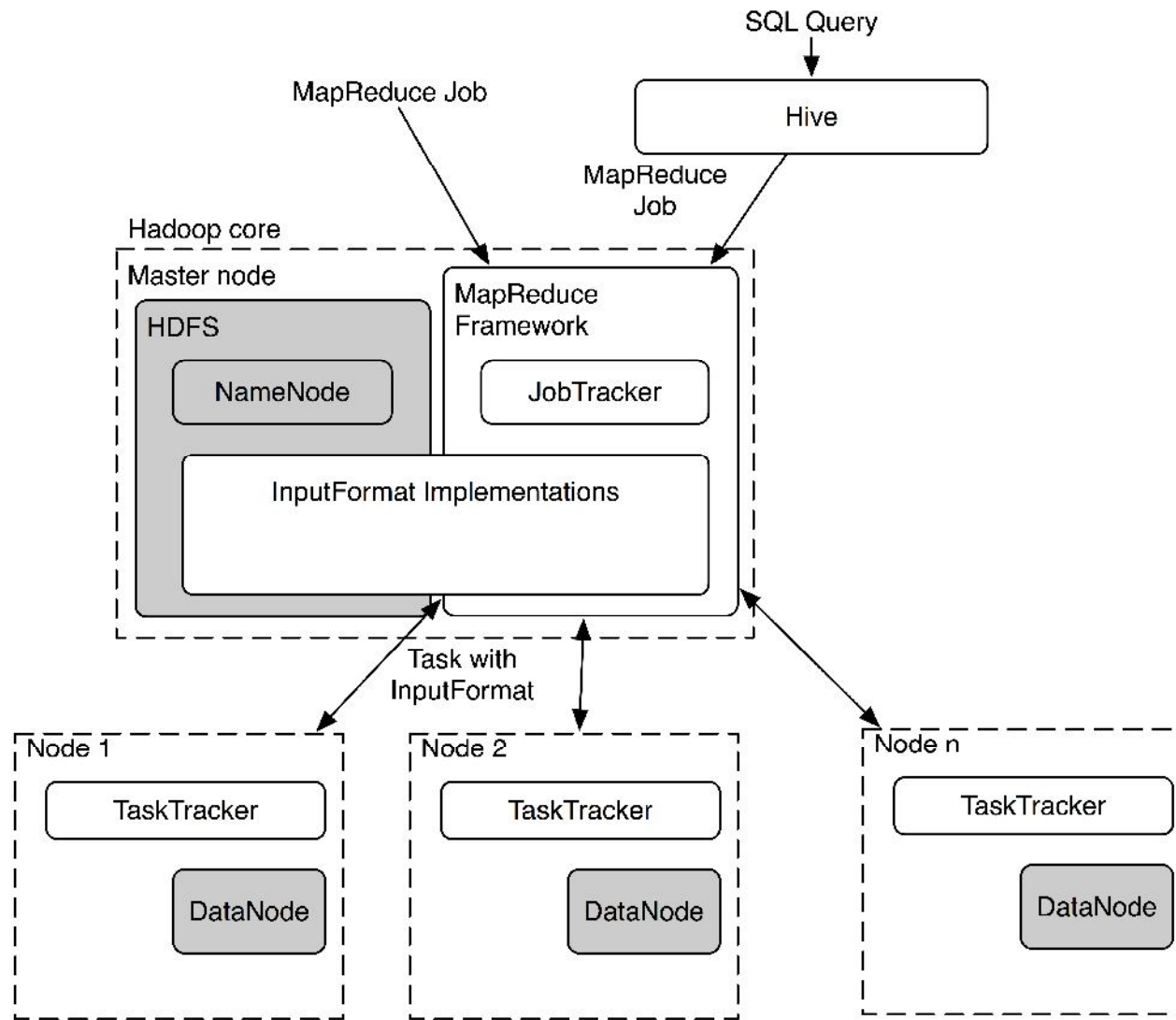
The Best of Both Worlds

HadoopDB's Design Concept

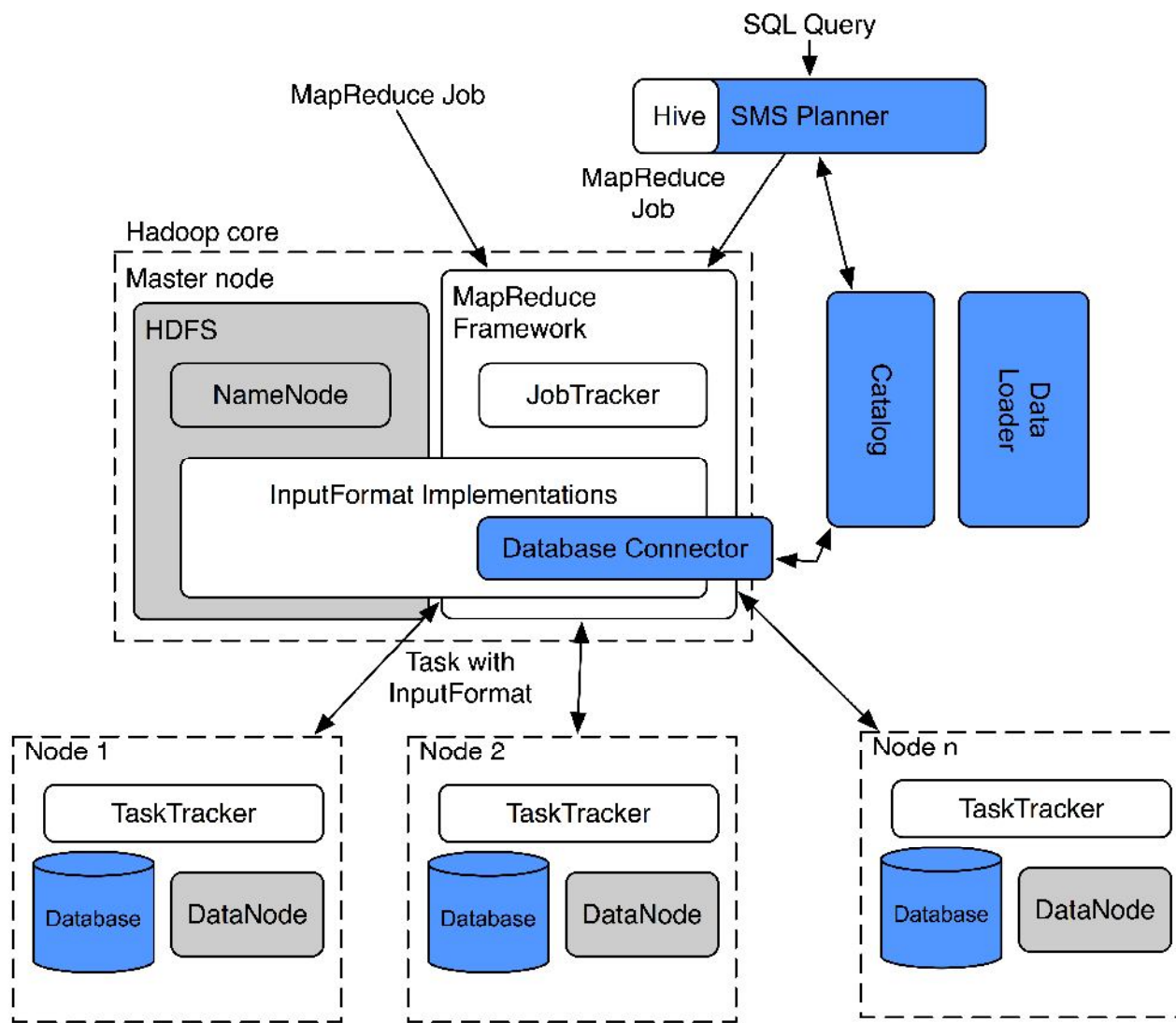
Multiple, independent, single node databases coordinated by Hadoop.

- 1 Get the performance of databases on structured data
- 2 Inherit the fault-tolerance and scalability properties of MapReduce
- 3 Provide a flexible querying interface - SQL or MR
- 4 Offer it for free as an Open Source system

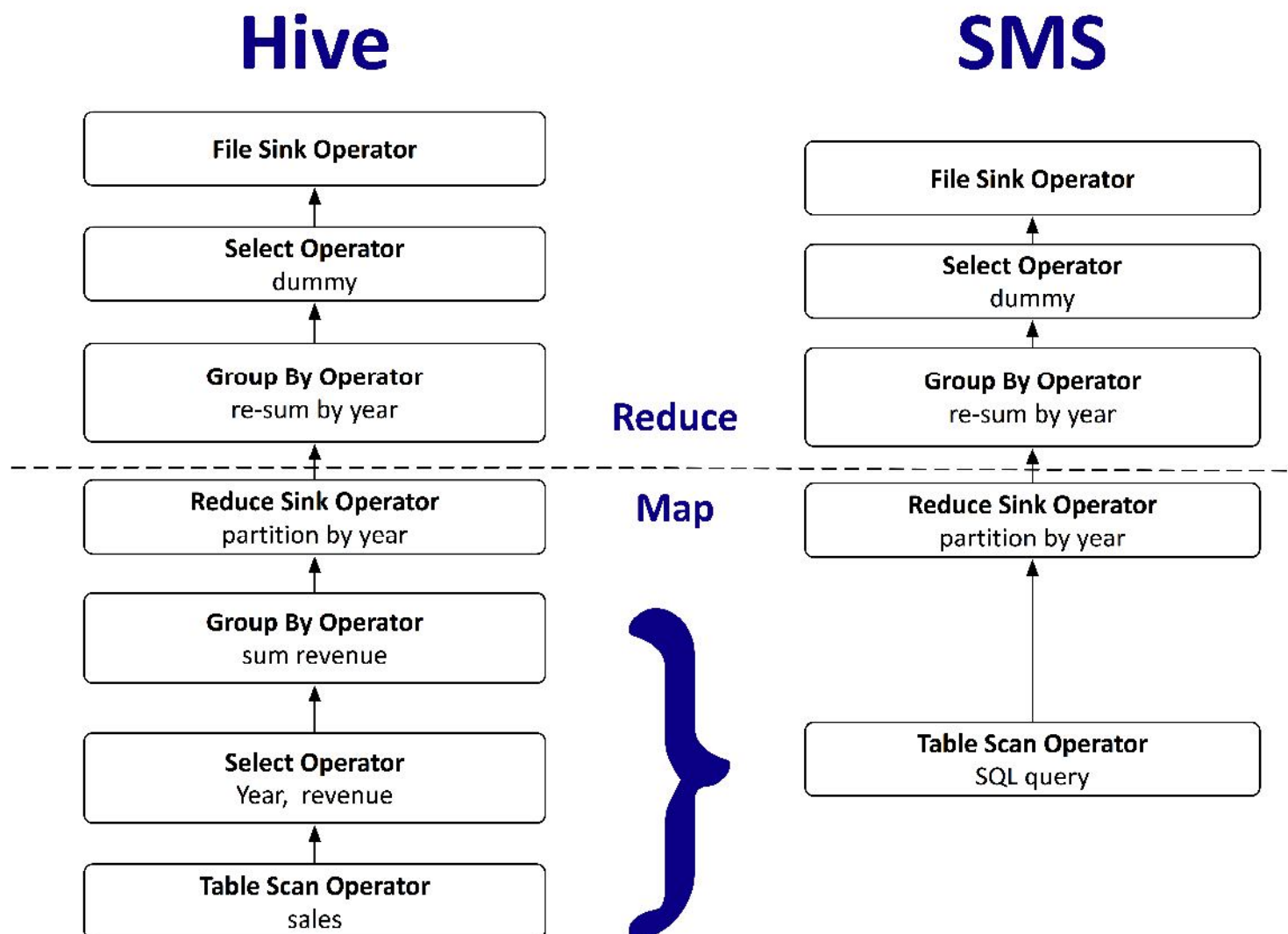
Hadoop Basics



Architecture



SQL-MR-SQL



```
SELECT YEAR(saleDate), SUM(revenue) FROM sales GROUP BY YEAR(saleDate);
```

Evaluating HadoopDB

Compare HadoopDB to

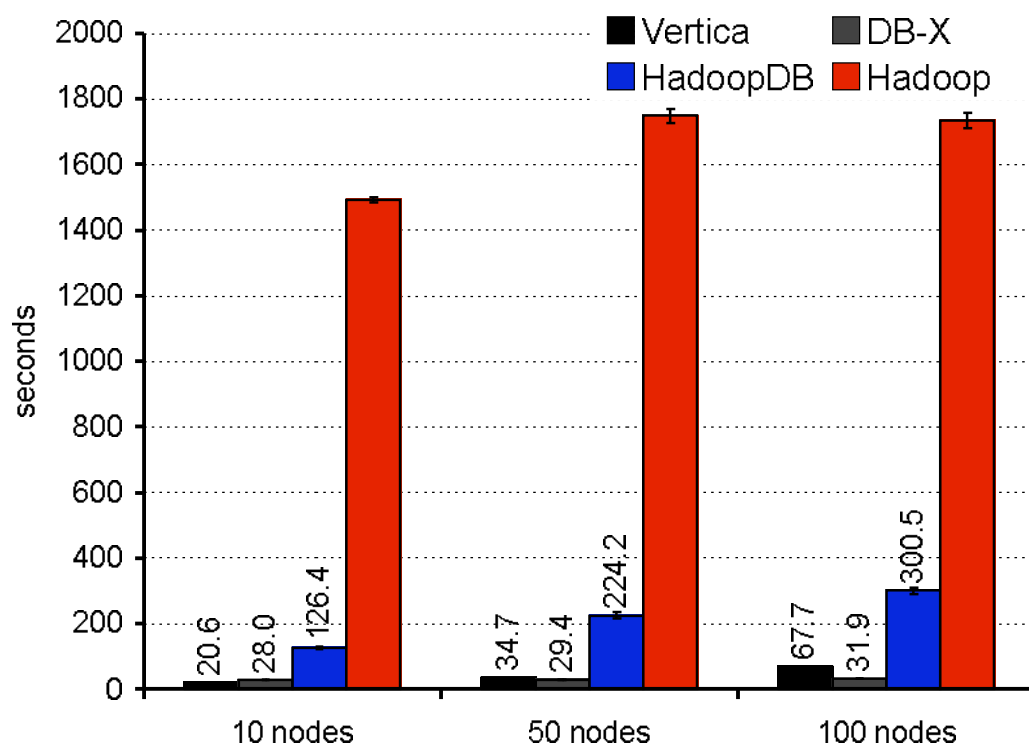
- 1 Hadoop
- 2 Parallel databases (Vertica, DBMS-X)

Features:

- 1 Performance:
 - *We expected HadoopDB to approach the performance of parallel databases*
- 2 Scalability:
 - *We expected HadoopDB to scale as well as Hadoop*

We ran the Pavlo et al. SIGMOD'09 benchmark on Amazon EC2 clusters of 10, 50, 100 nodes.

Performance: Join Task



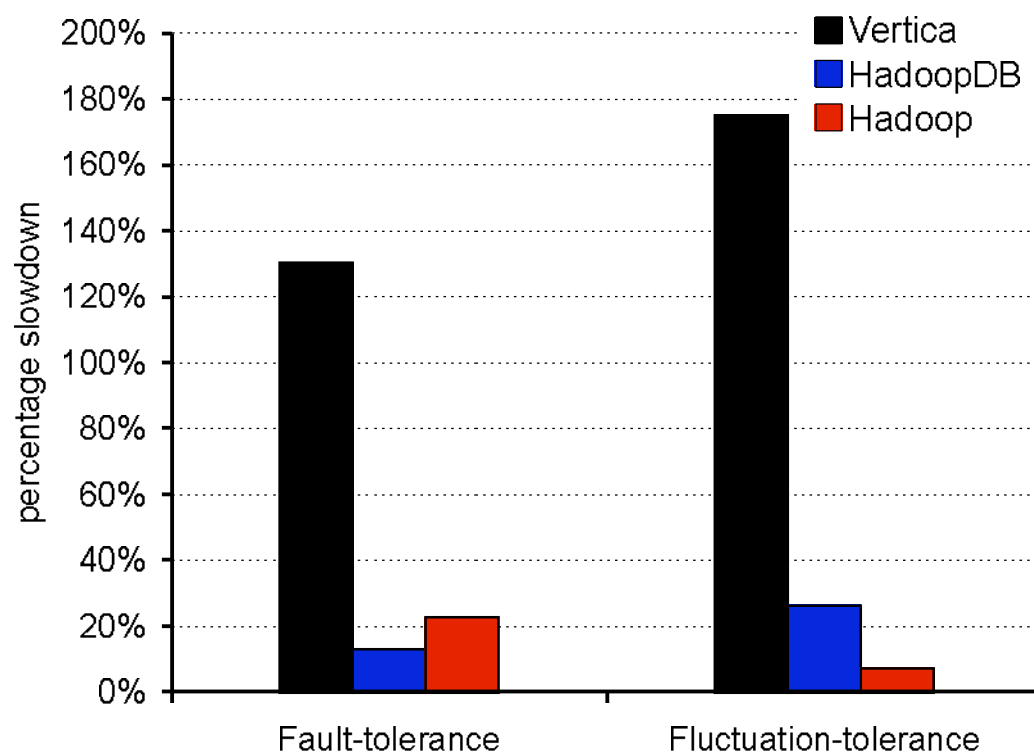
- 1 No full table scan due to clustered indexing
- 2 Hash partitioning and efficient join algorithm

```

SELECT sourceIP, AVG(pageRank), SUM(adRevenue)
FROM rankings, uservisits
WHERE pageURL=destURL
AND visitDate BETWEEN 2000-1-15 AND 2000-1-22
GROUP BY sourceIP
ORDER BY SUM(adRevenue) DESC LIMIT 1;

```

Scalability: Results



- 1 HadoopDB and Hadoop take advantage of runtime scheduling by splitting data into chunks or blocks
- 2 Parallel databases restart entire query on node failure or wait for the slowest node

To summarize

HadoopDB ...

- 1 Hybrid of Parallel DBMS and MapReduce
- 2 Scalable
- 3 Fault-tolerant
- 4 High-performing
- 5 Free and Open-source

<http://www.hadoopdb.net>

Roadmap

Moving from the research prototype to a full-fledged system:

- ① Full SQL support in SMS Planner
- ② Speed up and automate loading/replication
- ③ Transparent integration with DBMS layer
- ④ Easier deployment and management
- ⑤ Looking for developers

We appreciate your feedback and help!