Big Data

Dilip Krishna
Partner, North-East Financial Services
Teradata
dilip.krishna@teradata.com

TERADATA
Raising Intelligence

Agenda

- Advances in Relational Database Technology
- MapReduce and Hadoop
- Applications of MapReduce
- MapReduce and RDBMS - Comparisons
- Coexistence between MapReduce and RDBMS

Brief Bio of Dilip Krishna

"Financial Data and Technology Guy"

Engagement Partner, North-East Financial Services, Teradata

> Former Director of N.A Enterprise Risk Management practice

> Consulted on ERM and Basel II initiatives - U.S and Canadian banks

Long experience in technology and business consulting in the financial industry

Large-scale projects including Basel II implementations.

Authored numerous articles about risk management and data architecture

Engineering degrees from the Ohio State University and the Indian Institute of Technology

CFA and FRM designations.

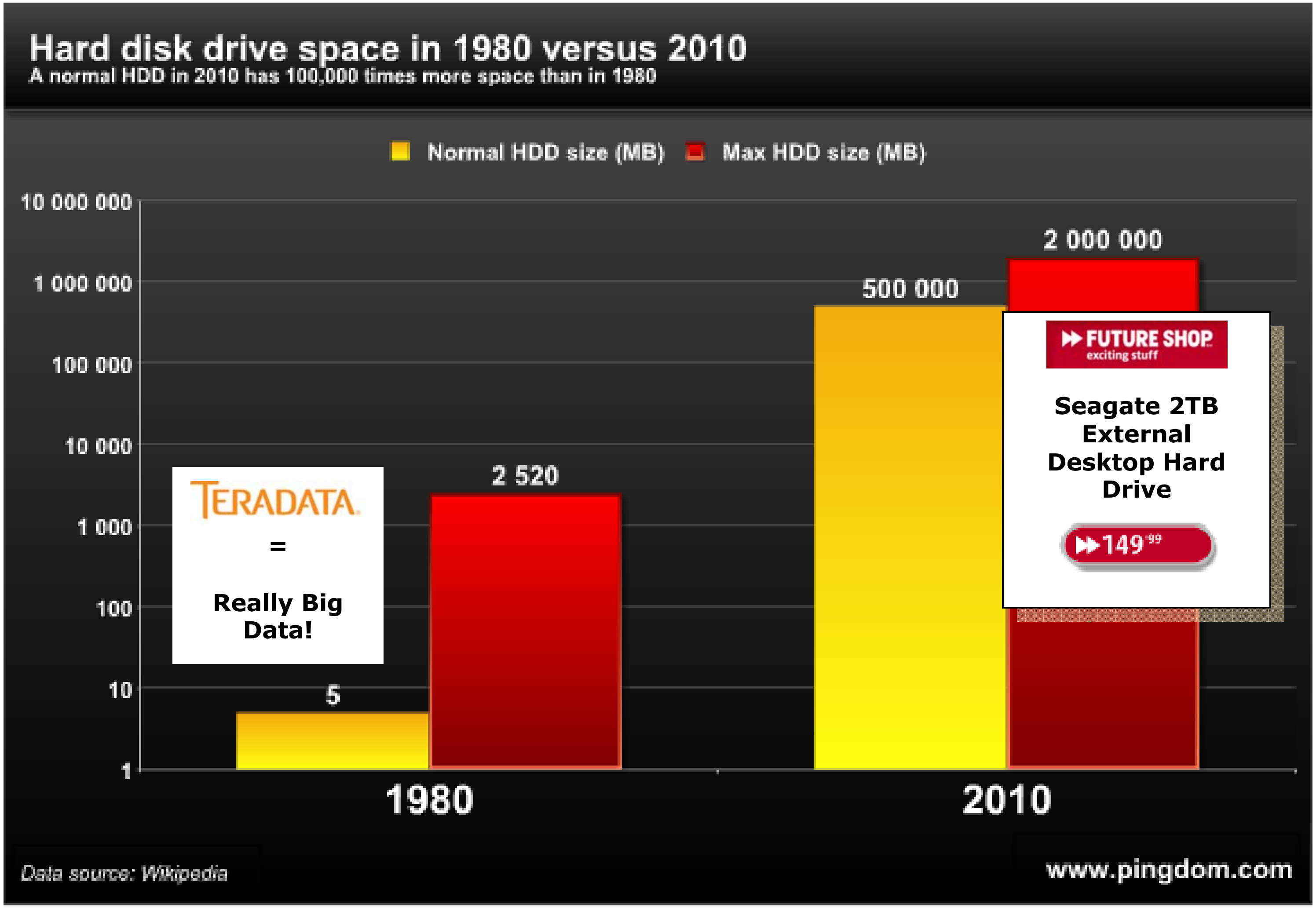
Disclaimer

1. These ideas do not necessarily reflect positions of Teradata
   > *I take full responsibility for any errors or omissions*

2. This presentation is meant as a review of current Big Data technologies, but the examples of RDBMS will solely use Teradata by necessity

What is this?

Big Data has evolved



What is Big Data?

Big data is a **collection of digital information** whose size is beyond the ability of most software tools and people to capture, manage, and process the data.

Current examples of Big Data include weblogs, RFID readers, sensor networks, social networks, Internet text and documents, Internet search indexing, call detail records, genomics, astronomy, biological research, military surveillance, photography archives, video archives, and large scale eCommerce.

Big Data **requires scalable technologies to efficiently process large quantities of data within tolerable elapsed times**. Big Data scalable technologies include MPP databases, the Apache Hadoop Framework, the Internet, and archival storage systems.

Currently Big Data ranges from a few dozen terabytes to several petabytes and is constantly increasing at both the low and high end size.

Source: Teradata working definition, Sep 2010

TERADATA Raising Intelligence

Big Data can work on lots of technologies but…

Some very, very, very large data warehouses

By *CurtMonash*
Created *May 12 2009 - 12:18pm*

In the course of my research, I'm running across some VERY large data warehouses. Several of them, especially in the web log/network event area, are in the multi-petabyte range. Perhaps most surprisingly, they're run on a broad range of data management software -- not just Teradata, but also Greenplum, Hadoop/Hive (which isn't even a DBMS!), Greenplum, and others.

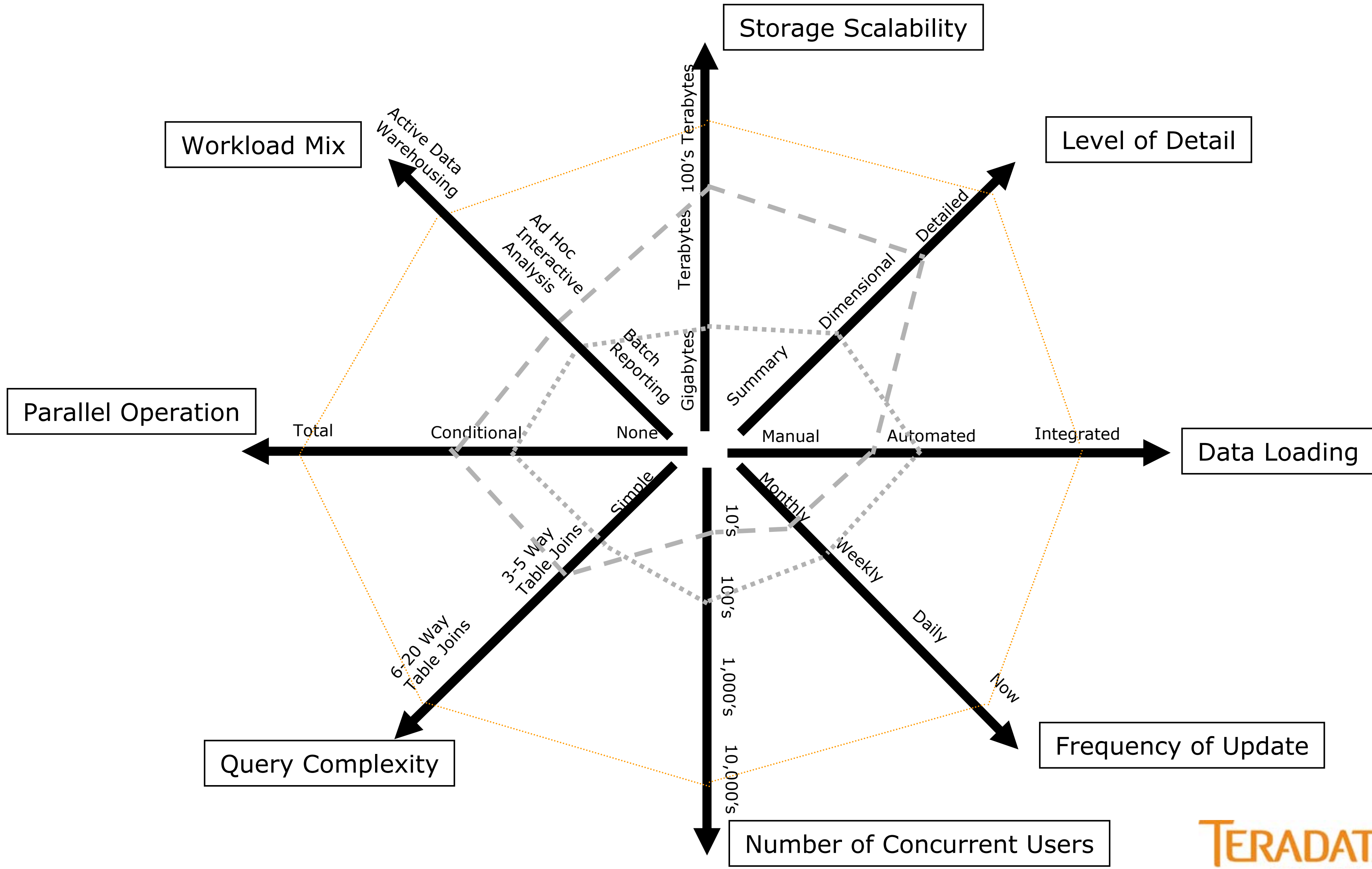
My current golly-gee-that's-really-big list goes something like this:

- eBay [1] has a 6 1/2 petabyte database running on Greenplum and a 2 1/2 petabyte enterprise data warehouse running on Teradata.
- Facebook [2] has a 2 1/2 petabyte datawarehouse runnin on Hadoop/Hive.
- Wal-Mart, Bank of America, another financial services company, and Dell also have very large Teradata databases [3].
- Yahoo's web/network events database, running on proprietary software, sounded

* Source: Curt Monash, http://www.networkworld.com/community/node/41777, downloaded on 11-Sep-2010

TERADATA
Raising Intelligence

It's not the size of data, but what you do with it!



Relational Databases

Not your father's Relational Database…

- Concepts date back to 1970 (Codd and Date)
- Evolution of full-fledged RDBMS systems – Oracle, Sybase, DB2 etc.
- Performance Considerations
  - Massively-Parallel Processing (MPP) platforms – pioneered by Teradata
  - Column-oriented databases – Sybase IQ, Vertica etc
- "Data Warehouse Appliances"
- Extremely large data-sets are routinely being processed
  - Many companies in "petabyte-club"

An explosion in maturity in RDBMS'
An Example

	Data Mart Appliance	Extreme Data Appliance	Data Warehouse Appliance	Extreme Performance Appliance	Active Enterprise Data Warehouse
Purpose	Test/ Development or Smaller Data Marts	Analytics on Extreme Data Volumes	Entry-level EDW or Departmental Data Marts	Extreme Performance for Operational Analytics	EDW/ADW for both Strategic and Operational Intelligence
Scalability	Up to 6TB	Up to 88PB	Up to 275TB	Up to 17TB	Up to 86PB
	Powered by the Teradata Database				

Big Data

Benefits of RDBMS and DW Appliances

- Declarative Programming Style of SQL
  - Ease-of-use (“no programming required”)
- Separation of code and data using Schemas
- Rich set of analysis tools
  - Report writers
  - Business intelligence tools
  - Data mining tools
  - Database design tools
- Custom coding possible via UDFs (but not easy)

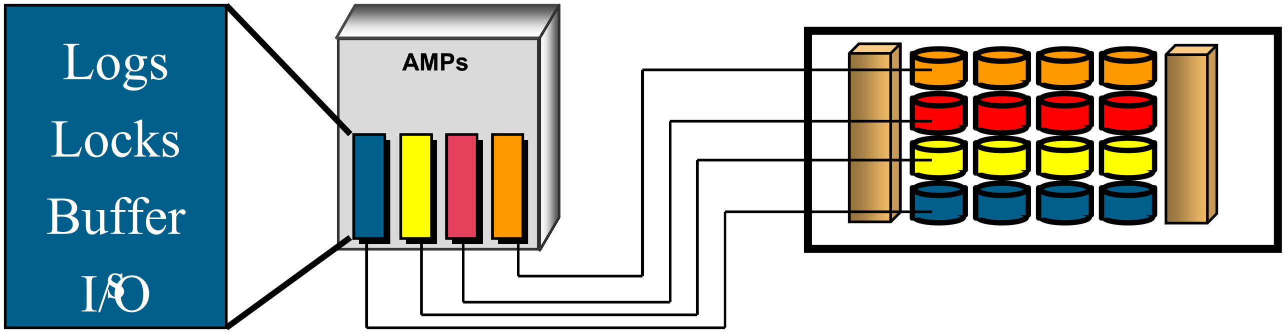
- Codebase Maturity
  - 30+ years of development
- Manageability
  - Out of the box Performance
  - One number to call
  - TCO Reduction
  - Reduced administration
  - Built-in high availability
  - Scalability
  - Rapid time-to-value

Low “Cost per Query”

TERADATA
Raising Intelligence

Scalable Shared Nothing Software
Virtual Processors

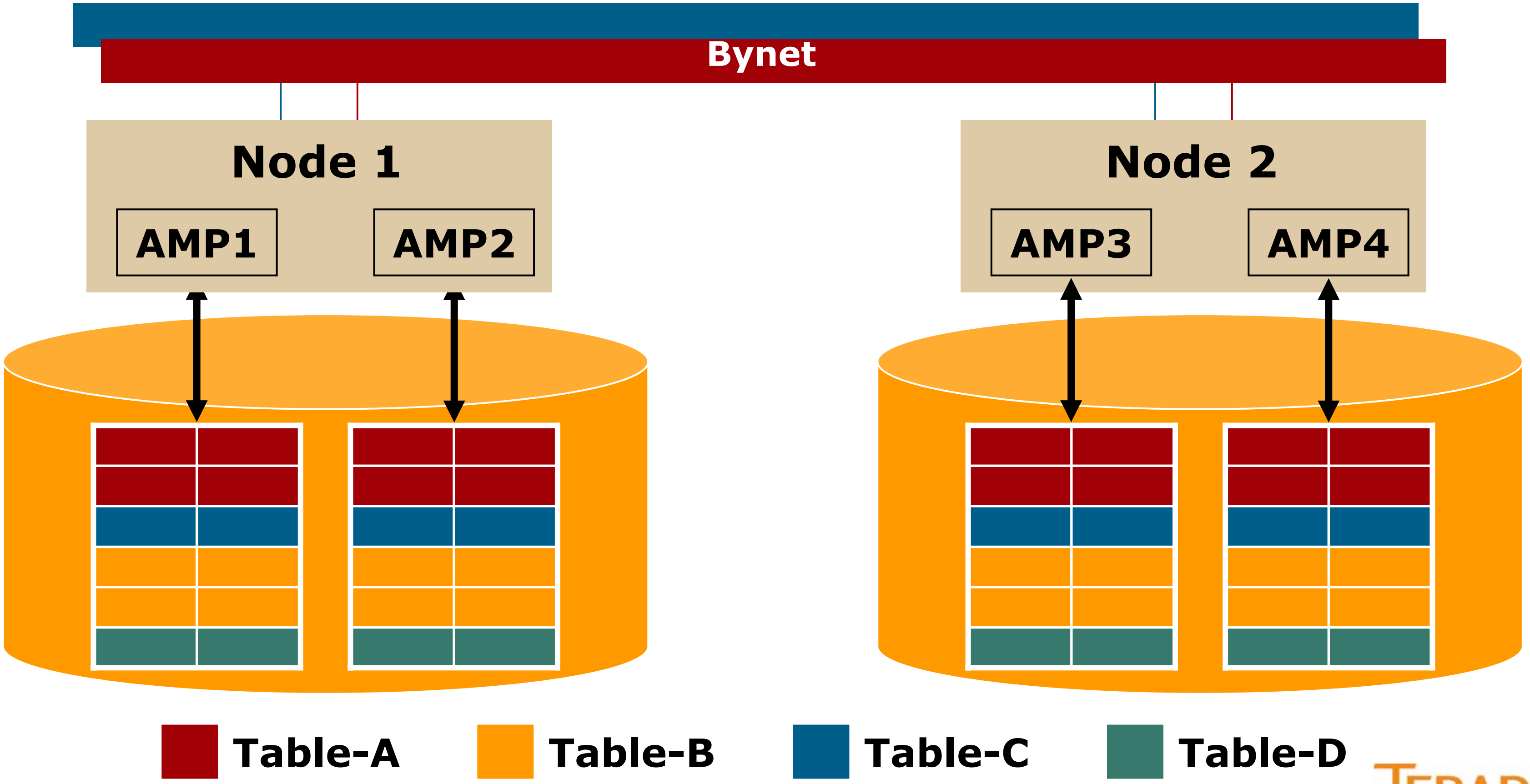
- Divide the data, workload, and system resources evenly among many parallel, processing units (AMPs)
- No single point of control for any operation
  - > I/O, Buffers, Locking, Logging, Dictionary
  - > Nothing centralized
  - > Nothing in the way of linear scalability



TERADATA
Raising Intelligence

Automatic Data Distribution

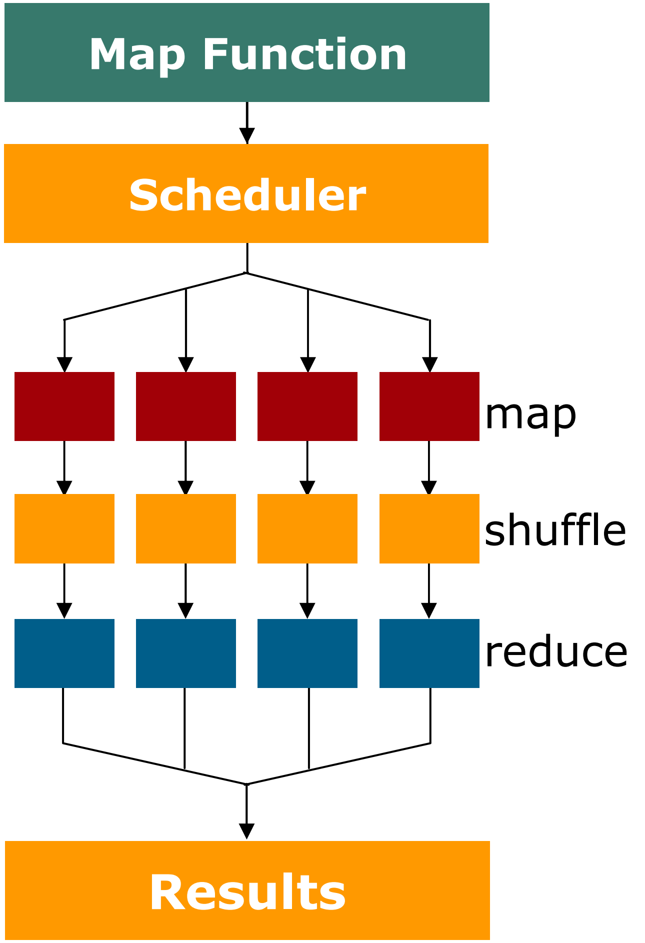
- Each node has rows equally distributed from every table
- Each node works independently, in parallel, on its rows



MapReduce and Hadoop

What is MapReduce?

- **A parallel programming framework**
  - > Originally from Google
    - – Generate search indexes
    - – Web scoring algorithms
  - > C++, Java, Python, etc.
  - > Harness 1000s of CPUs
- MapReduce provides
  - > Automatic parallelization
  - > Fault tolerance
  - > Monitoring & status updates
- Hadoop
  - > Open source MapReduce



What is Hadoop?

- Open source version of MapReduce
  - > Top level Apache project
  - > Contributors: Yahoo!, IBM, Google
- Hadoop Core includes
  - > Distributed File System
  - > Map/Reduce
  - > Job Tracker/Task Tracker
- Written in Java
  - > Linux, Mac OS/X, Windows, and Solaris
- Growing list of supporting tools

http://hadoop.apache.org

Hadoop is named for inventor’s child's stuffed elephant

MapReduce has arrived

Google	Clusters of 100s/1000s of commodity Intel machines Implementation is a C++ library linked into user programs
YAHOO!	15,000+ computers running Hadoop Research for Ad systems and web search
amazon.com	Product search indexes Analytics from user sessions
facebook	Log analysis for reporting and analytics and machine learning
FOX INTERACTIVE MEDIA	Log analysis, data mining, and machine learning
The New York Times ON THE WEB	Large scale image conversion
SLAC NATIONAL ACCELERATOR LABORATORY	High energy physics, genomics, Digital Sky Survey

> 60 Listed users of Hadoop on http://wiki.apache.org/hadoop/PoweredBy

Yahoo Hadoop clusters

- Has ~38,000 machines running Hadoop
- Largest Hadoop clusters are *currently* 2000 nodes
- Several petabytes of user data (compressed, unreplicated)
- Run hundreds of thousands of jobs every month

How does MapReduce work?
An Example

2 Input Files

File1 => `Hello World Bye World`

File2 => `Hello Hadoop Bye Hadoop`

Map Reduce Class

```
public class WordCount {
…
public void map( …)

public static class Reduce ()

     public static void main(…)
     {
          setMapperClass(Map.class);
          setReducerClass(Reduce.class);
     }
…
}
```

Intermediate Files (Output of Map)

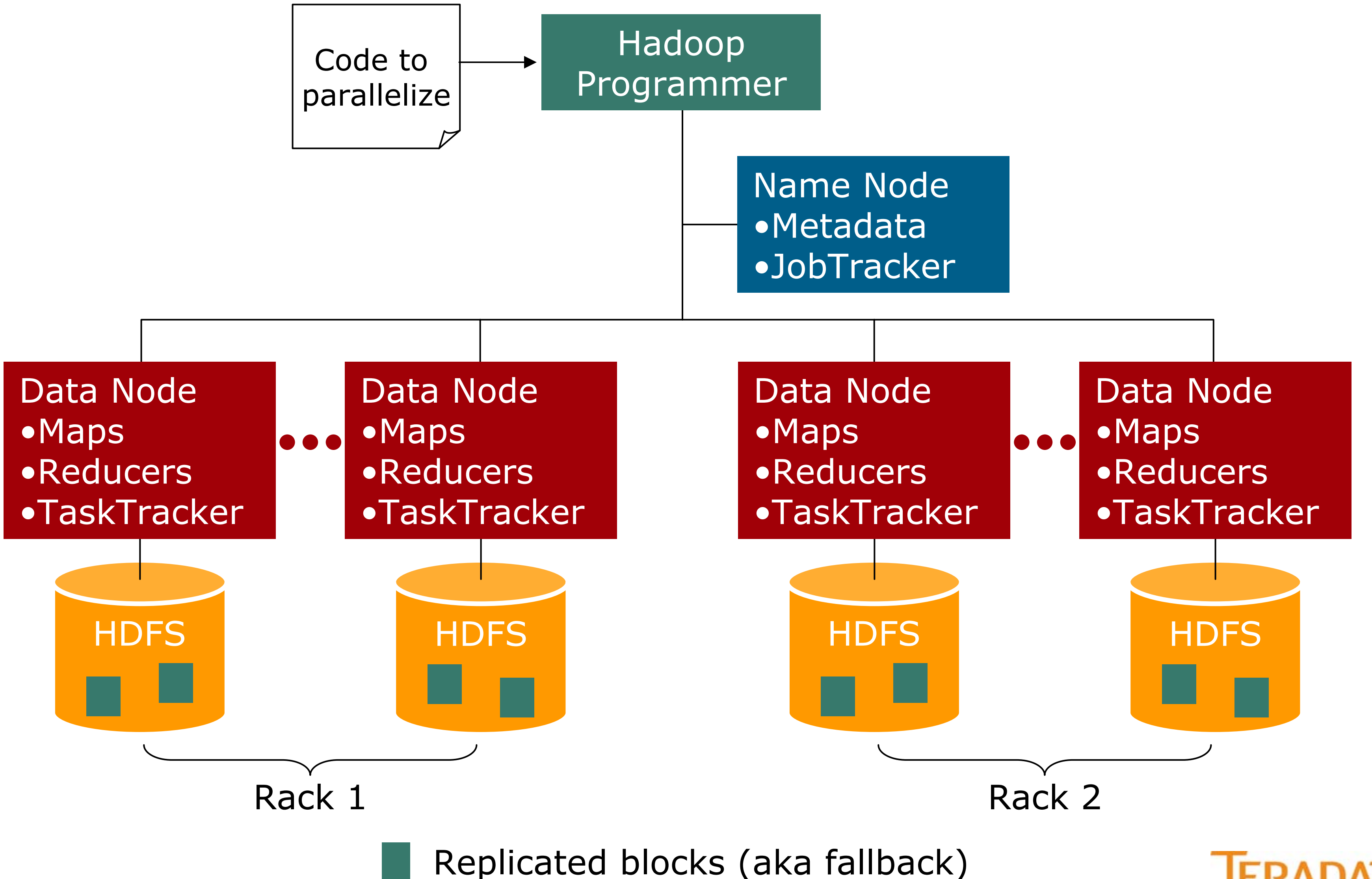
< Hello, 1> < Hadoop, 1> < Goodbye, 1> < Hadoop, 1>	< Hello, 1> < Hadoop, 1> < Goodbye, 1> < Hadoop, 1>

Output File (Output of Reduce)

Bye 1
Goodbye 1
Hadoop 2
Hello 2
World 2

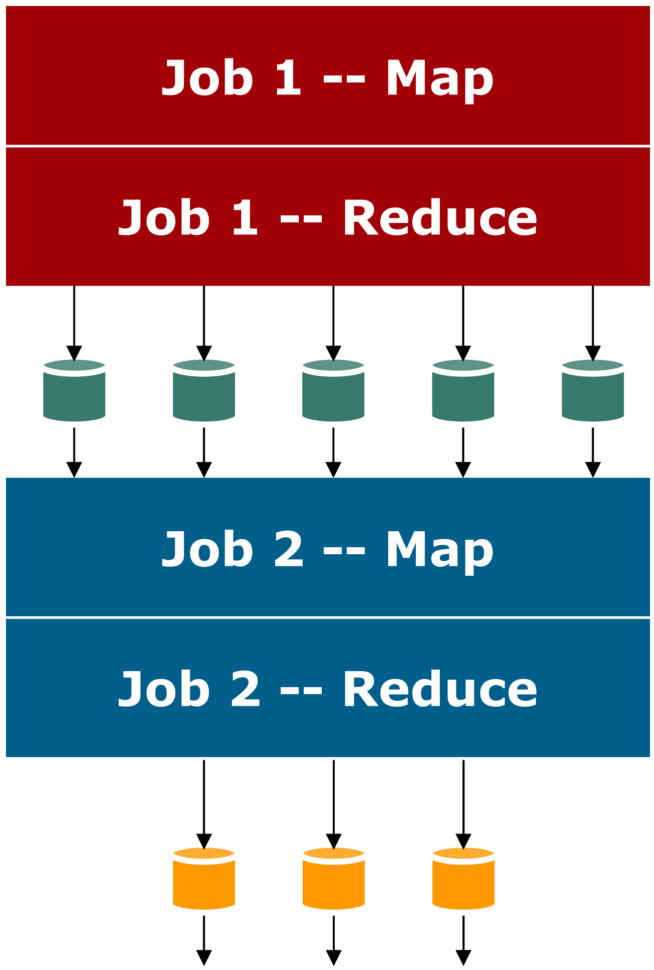
Raising Intelligence

Simplified Hadoop Architecture



Running Successive MapReduce Jobs

- MR is batch processing
- Run one MR job
  - > Output files on reduce node
- Run next MR job
  - > Input files are prior reducer output
  - > Creates new reducer files
- Keep running MapReduce batch jobs until solved



Benefits of MapReduce / Hadoop

- Support for 100s to 1000s of server nodes
  - Extreme scalability
  - Commodity hardware = low costs
- Simplicity of programming model
  - Attractive to Programmers (programming vs. SQL)
- Easy integration of developer tools
  - Java, grep, python, etc.
- Batch programming
  - Complex multi-step processing
- Absence of rigid schema
  - "No ETL"
- Flexibility
- Some fault-tolerance
- Easy setup

TERADATA
Raising Intelligence

MapReduce / Hadoop Challenges

- Performance lags MPP databases
- Quality of Service
  - > Big jobs can hog the cluster (No workload management)
  - > JobTracker memory as limited resource
  - > No “Fair Scheduler”
  - > Hard to troubleshoot performance bottlenecks
- Skills
  - > Ongoing code maintenance depends on open-source process
  - > Skilled resources are still few-and-far-between

Addressing Hadoop drawbacks
Related Open Source Projects

- Pig (Pig Latin)
  - Process language
  - Data set handling, transformations
- Hive
  - SQL interface to Hadoop
- HBase
  - Column oriented database
- Many others
  - Zookeeper
  - Eclipse Hadoop plug-in
  - Thrift
  - Cascading
  - Scope
  - Dryad/Linq
  - etc.

MapReduce Applications

Common Hadoop Applications

- Reporting
  - Web logs
- Web analytics
  - Click streams
  - Session analysis
  - Social media
- Data mining
  - Personalization
  - Clustering, k-means, bayesian
- Text analysis
  - Search indexes
  - Sentiment analysis

Web Analytics / Web Server Log Analysis

Multivariate testing

Usability testing

Site error monitoring

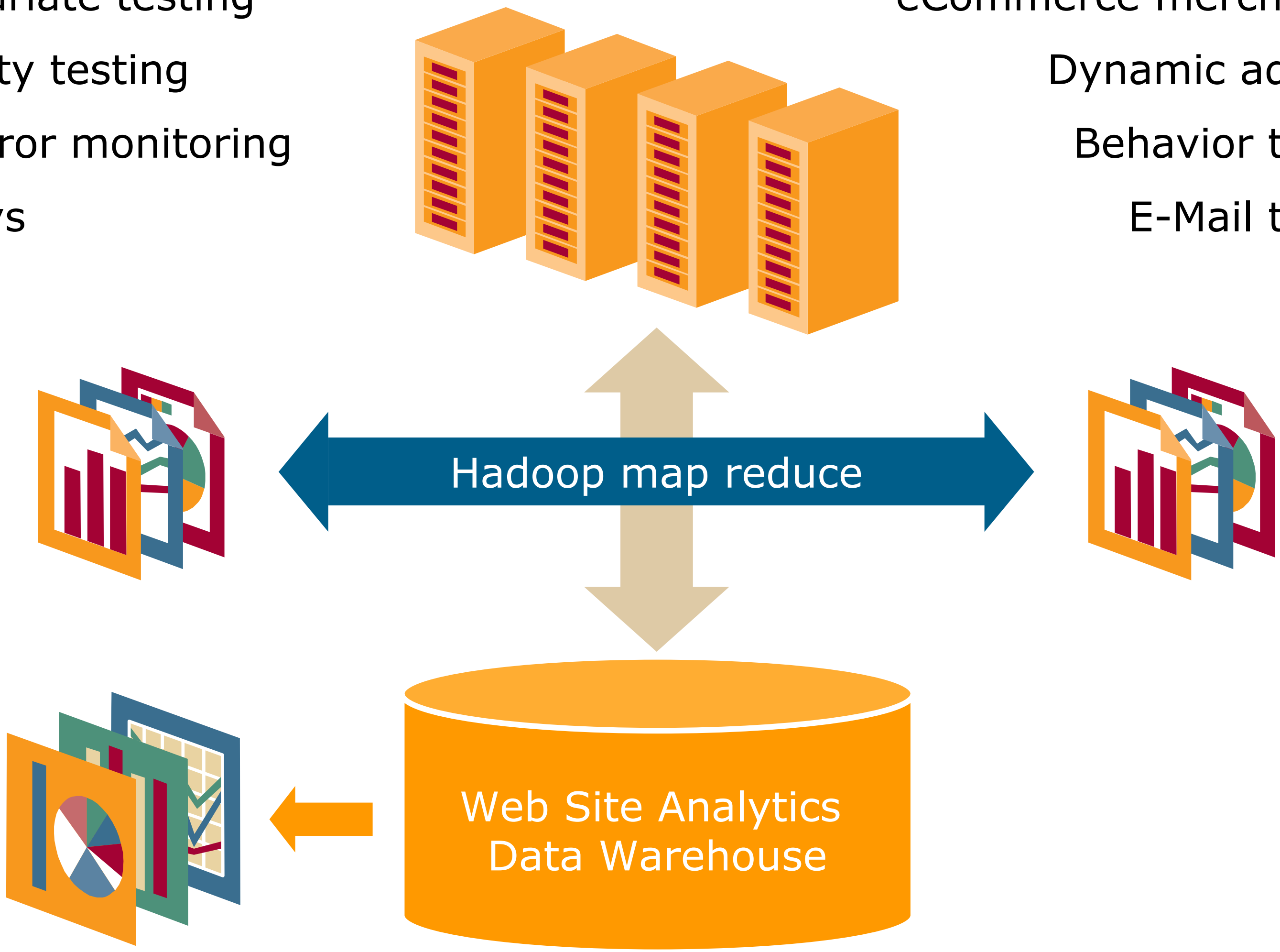
Surveys

eCommerce merchandising

Dynamic ad serving

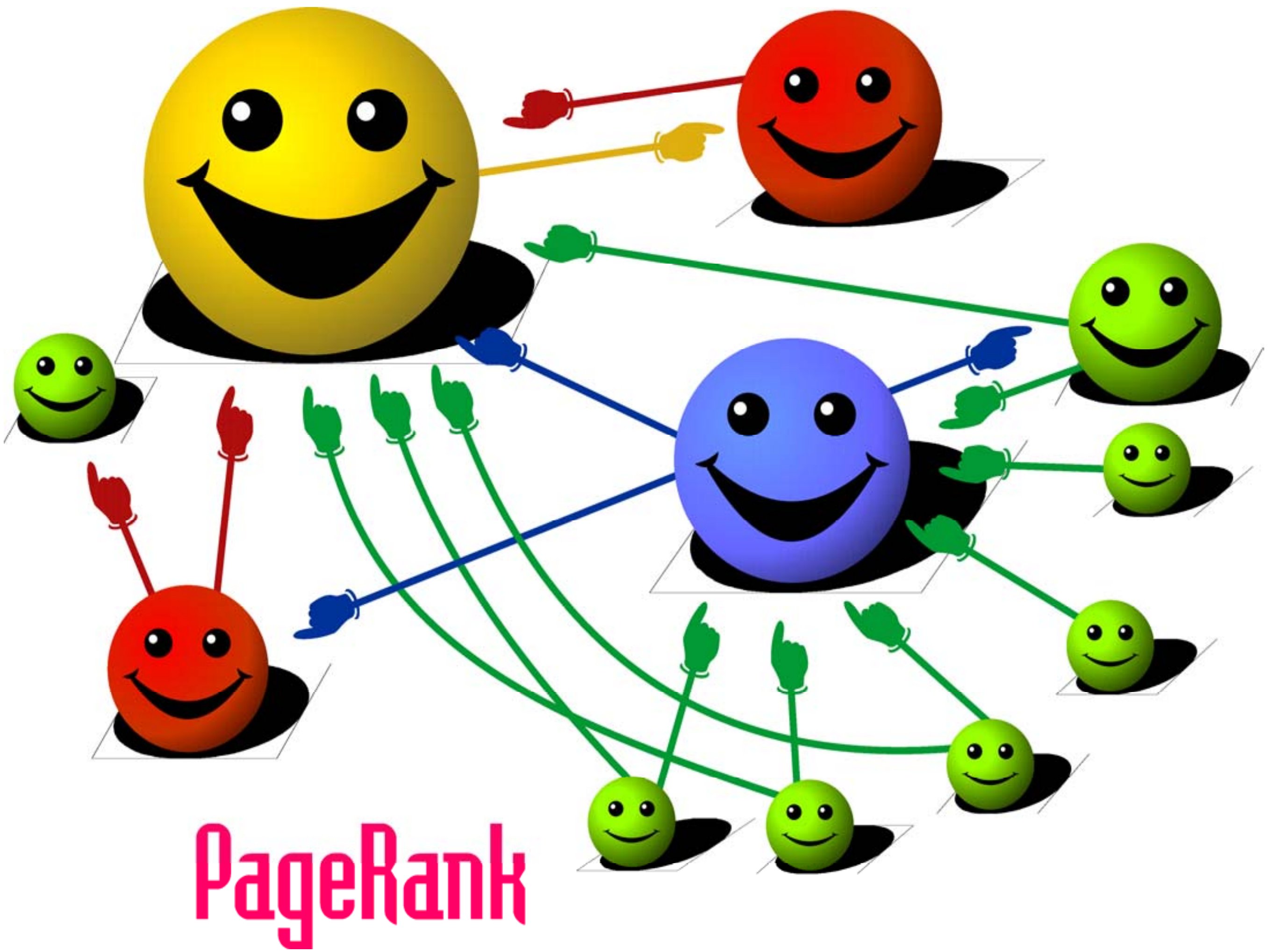
Behavior targeting

E-Mail targeting



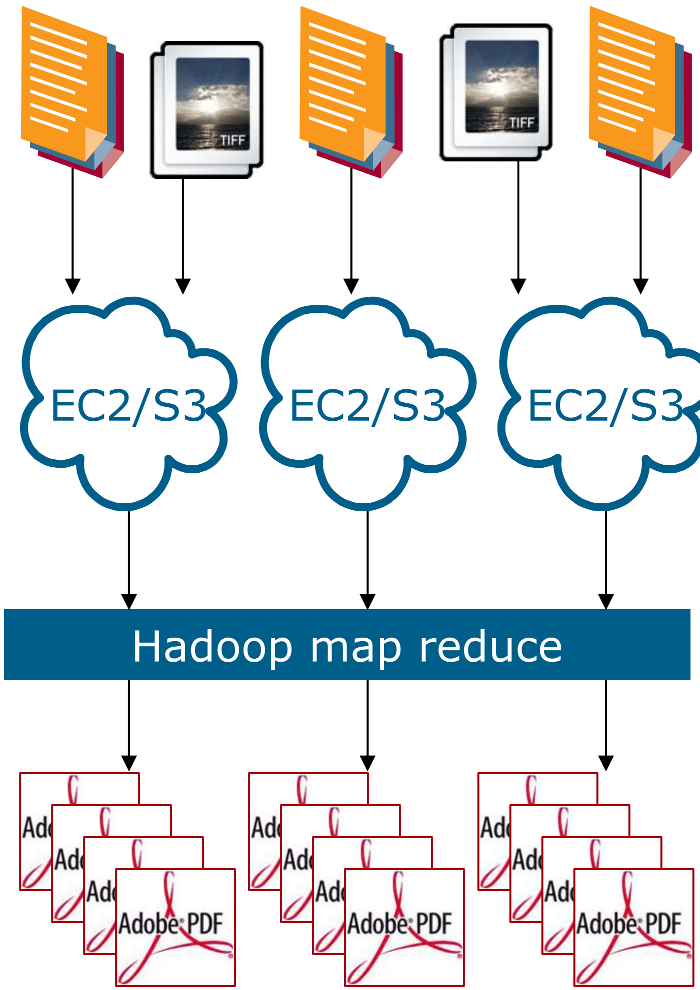
Cross-channel, trends,
forecasts, anomalies

Search Index Page Ranking

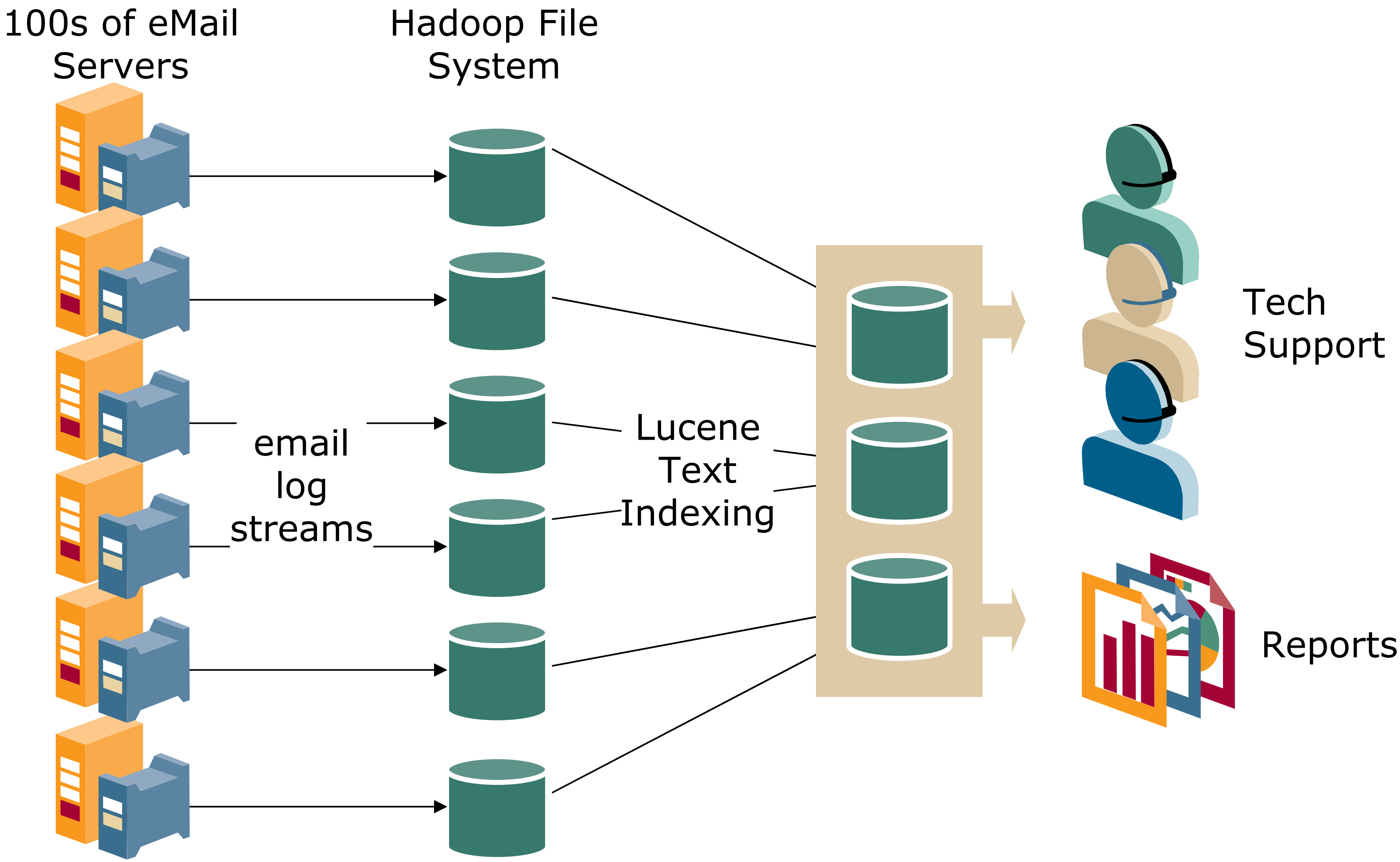


Large Scale Data Transformation

- Millions of files or objects
- Find key object and transform it
  - > Convert BMPs to JPGs
  - > Convert DOCs to PDFs
  - > Change NYC to New York City
  - > Etc.
- ETL but data stays on node
- New York Times
  - > Convert 11M articles to PDFs
  - > Convert TIFFs in clouds
  - > Use Amazon EC2 and S3 clouds
  - > 4TB of articles → 1.5TB of PDFs



RackSpace eMail Logs Tech Support



http://highscalability.com/how-rackspace-now-uses-mapreduce-and-hadoop-query-terabytes-data

http://blog.racklabs.com/?p=66

MapReduce and RDBMS Comparisons

Lots of controversy and flame-wars

DBMS2

April 14, 2009

eBay thinks MPP DBMS clobber Ma

I talked with Oliver Ratzesberger and his team at eBay already knew to be MapReduce non-fans. This time I ac

Oliver believes that, on the whole, MapReduce is 6-8X s functionality in an MPP DBMS, and hence should only be This view is based on part on simulations eBay ran of tl benchmark. On 72 Teradata nodes or 96 lower-powered nodes running another (currently unnamed, as per yet another of my PR fire drills) MPP DBMS, a simulation of Terasort executed in 78 and 120 secs respectively, which is very comparable to the times Google and Yahoo got on 1000 nodes or more.

And by the way, if you use many fewer nodes, you also consume much less floor space or electric power.

MapReduce: A major step backwards

on Jan 17 in Database architecture, Database history, Database innovation posted by DeWitt

[Note: Although the system attributes this post to a single author, it was written by David J. DeWitt and Michael Stonebraker]

On January 8, a Database Column reader asked for our views on new distributed database research efforts, and we'll begin here with our views on MapReduce. This is a good time to discuss it, since the recent trade press has been filled with news of the revolution of so-called "cloud computing." This paradigm entails harnessing large numbers of (low-end) processors working in parallel to solve a computing problem. In effect, this suggests constructing a data center by lining up a large number of "jelly beans" rather than utilizing a much smaller number of high-end servers.

This article, coming as it does from such eminent folks from academia, just goes on to show how dogmatic some of the academic community has become. Or is it just that the earlier world / status-quo has been shattered so much that you can't bear the success of a newer system? This is truly what academia should NOT be.

Comment by Ashwin — January 19, 2008 @ 12:27 am

- Mark Callaghan on April 15th, 2009 1:09 am

  @UnHolyGuy – stop it with all of the facts. I prefer to read that MapReduce is bad because a big Teradata customer says so.

Our friend the Typical Programmer points out why this article shows a complete mis-understanding of what MapReduce is or is for. Amongst other things, that it's not intended, in any way, to be a system for storing and managing structured data, and thus **not a database.**

http://typicalprogrammer.com/programming/mapreduce/

http://www.dbms2.com/2009/04/14/ebay-thinks-mpp-dbms-clobber-mapreduce/

TERADATA
Raising Intelligence

"MapReduce: A major step backwards"
DeWitt and Stonebraker

- Not a database
  - No schema, indexes, optimizer
- Not high performance
- Not a data warehouse
  - No integrated data, no history
  - Simple reports, single fact table
  - Severe data skew
- Not mature technology
  - Many single points of failure
  - No ISV tools
  - No performance tools
- Not for business users
  - No pivot, drill down, self service
- **But does it matter??**

TERADATA
Raising Intelligence

Architectural Comparisons

	RDBMS	MapReduce
Repetitive Record Parsing	• Parse once during ETL phase	• Parse (at least) once per query
Compression	• Compression yields performance (and space) benefits	• Compression yields poorer performance (currently)
Pipelining	• Optimized Query plan allows data "push" • Intermediate data not written to disk	• Data "pull" via intermediate files • Improved fault-tolerance ("checkpointing") • Reduced performance
Schema Support	• Native	• Externally imposed
Indexing	• Native	• Programmed
Programming Model	• Declarative	• Programmatic • Attractive to programmers
Flexibility	• Limited flexibility	• Significant flexibility
Time-to-value	• High cost-of-acquisition • Difficult to setup	• Easy to setup and use • Low cost of acquisition
Cost of Operation	• Lower costs/query	• High labor cost/query (labor + operating costs)

* MapReduce and Parallel RDBMSs: Friends or Foes?

TERADATA
Raising Intelligence

Performance Comparisons
Load Times better in Hadoop

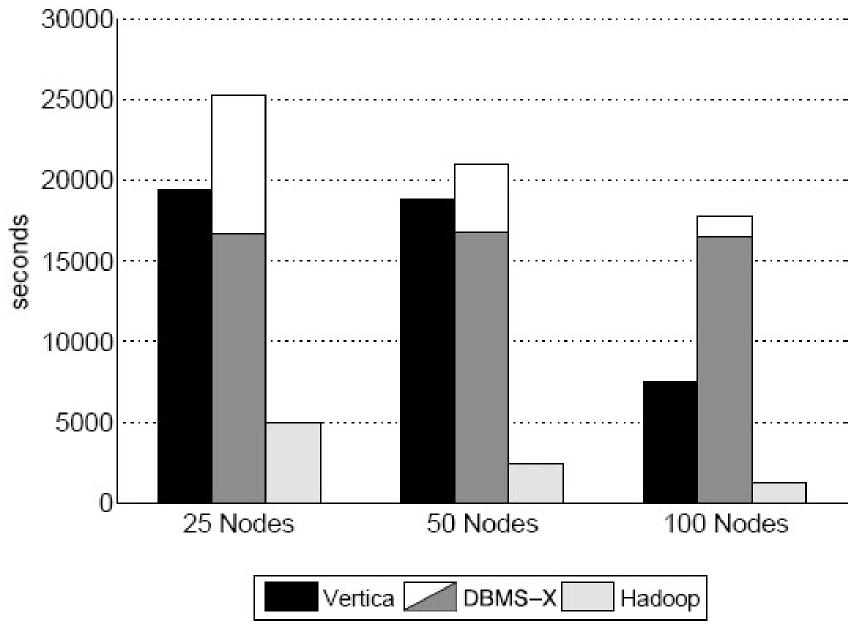


Figure 2: Load Times – Grep Task Data Set (1TB/cluster)

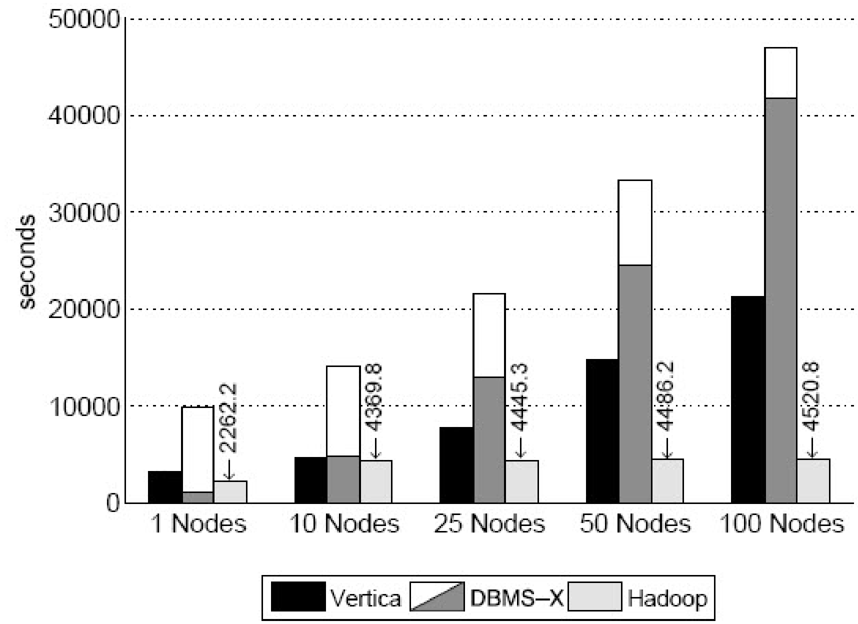


Figure 3: Load Times – UserVisits Data Set (20GB/node)

Source: Stonebreaker et al, A Comparison of Approaches to Large-Scale Data Analysis, Apr 13,2009

TERADATA
Raising Intelligence

Performance Comparisons

grep and select times worse in Hadoop

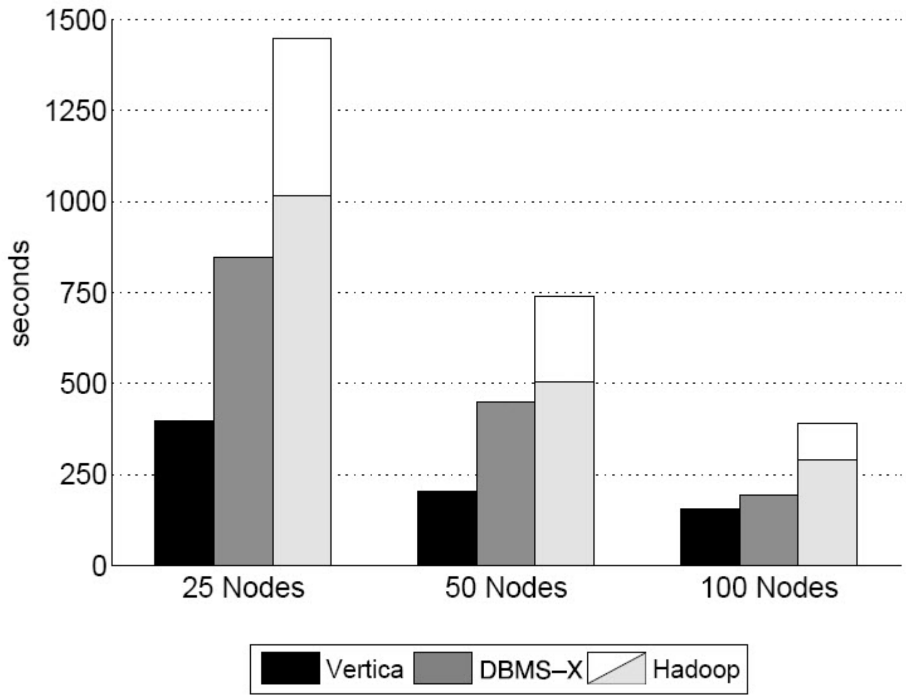


Figure 5: Grep Task Results – 1TB/cluster Data Set

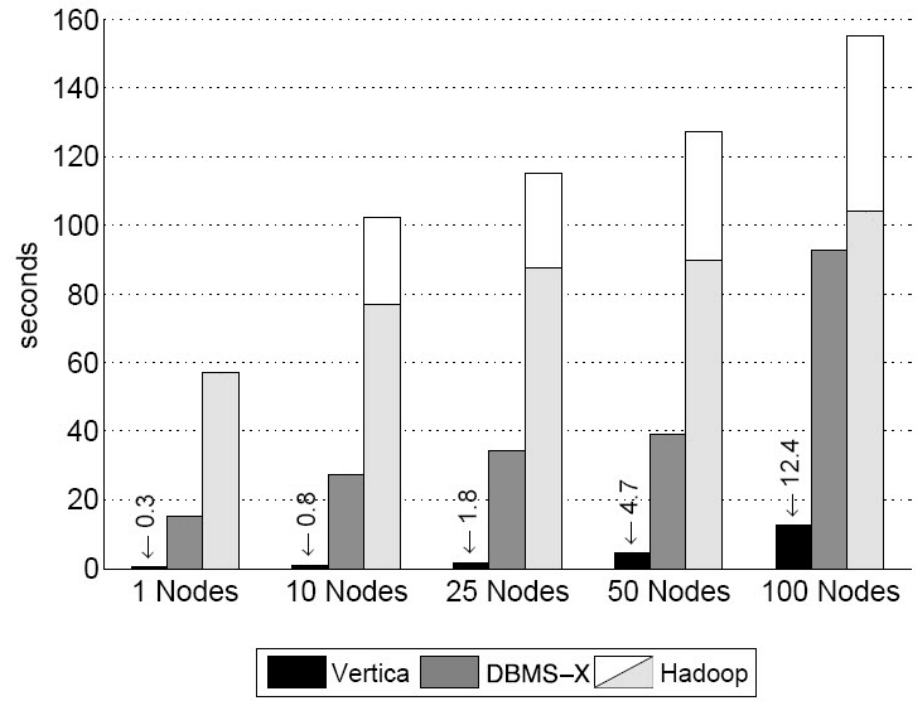


Figure 6: Selection Task Results

Source: Stonebreaker et al, A Comparison of Approaches to Large-Scale Data Analysis, Apr 13,2009

TERADATA
Raising Intelligence

Performance Comparisons

Aggregate and Join Comparison – Much worse in Hadoop

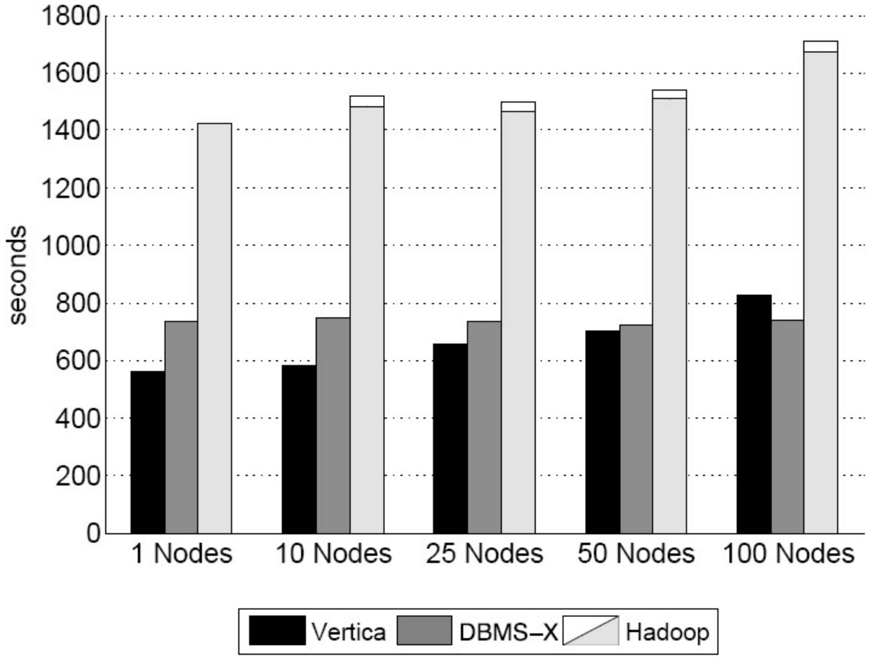


Figure 7: Aggregation Task Results (2.5 million Groups)

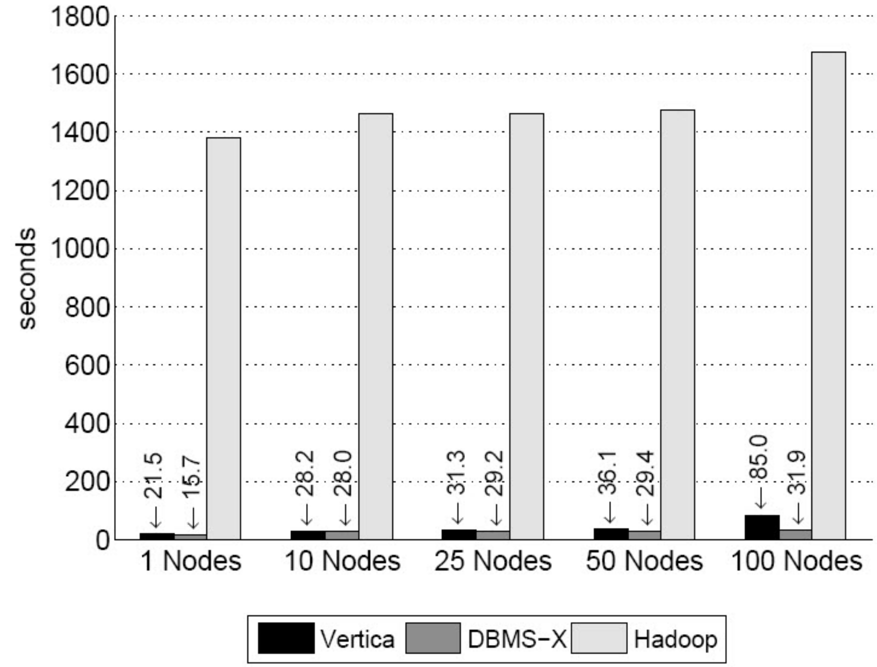


Figure 9: Join Task Results

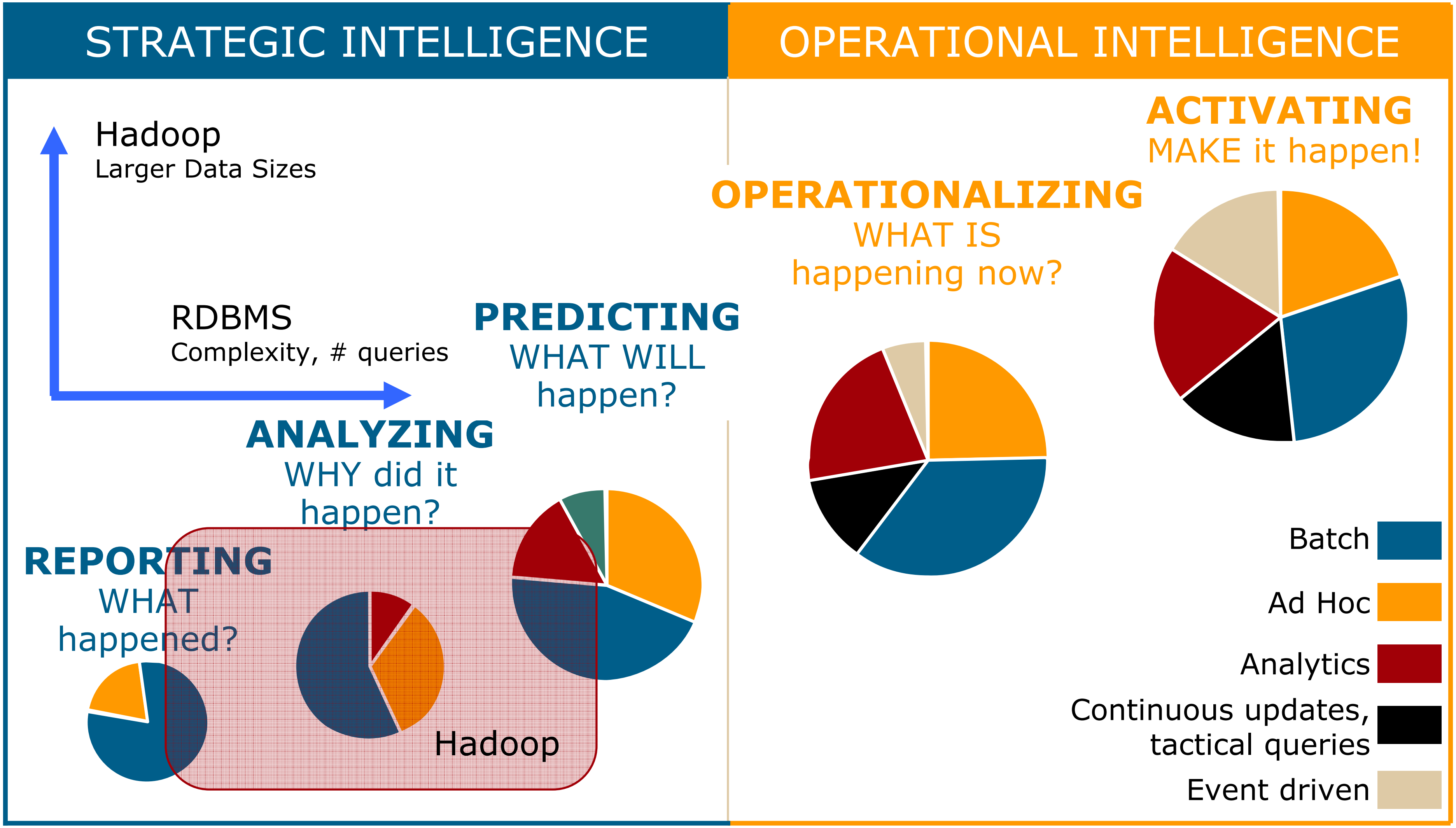
Source: Stonebreaker et al, A Comparison of Approaches to Large-Scale Data Analysis, Apr 13,2009

TERADATA
Raising Intelligence

MapReduce and RDBMS

Complementary, not Competitive!

Analytics Processes
Comparing MapReduce and RDBMS

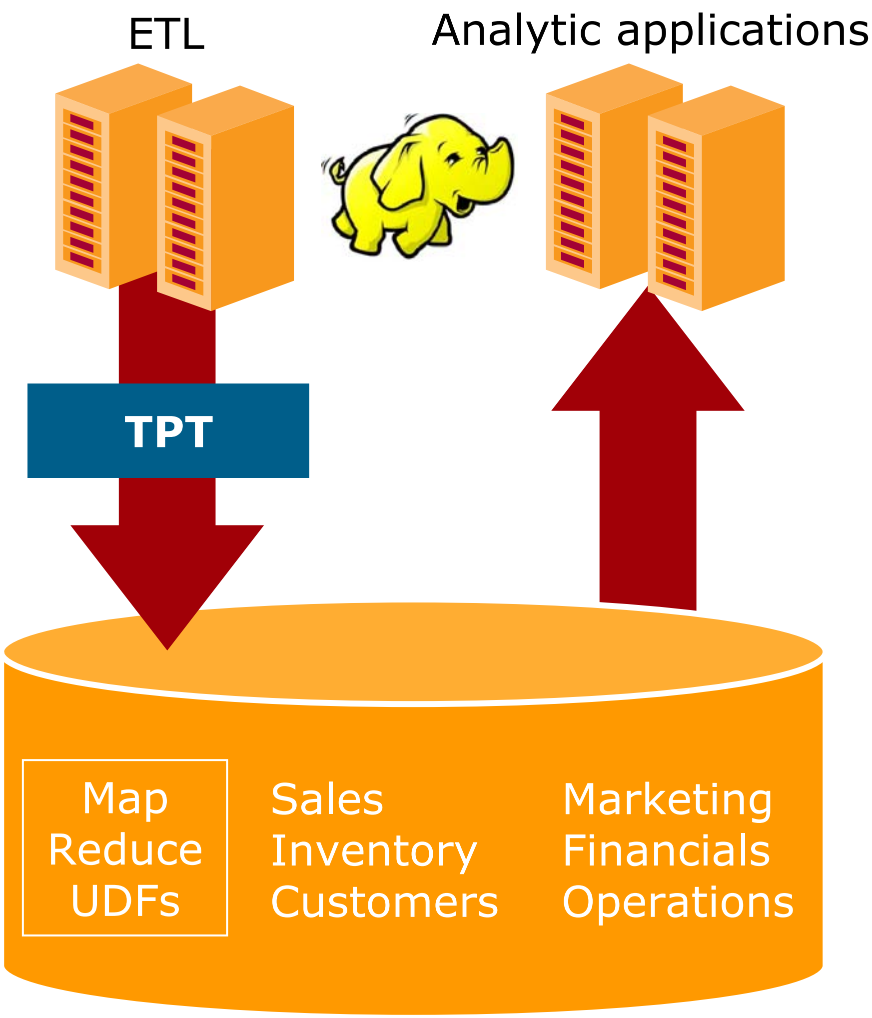


TERADATA
Raising Intelligence

Teradata and Hadoop

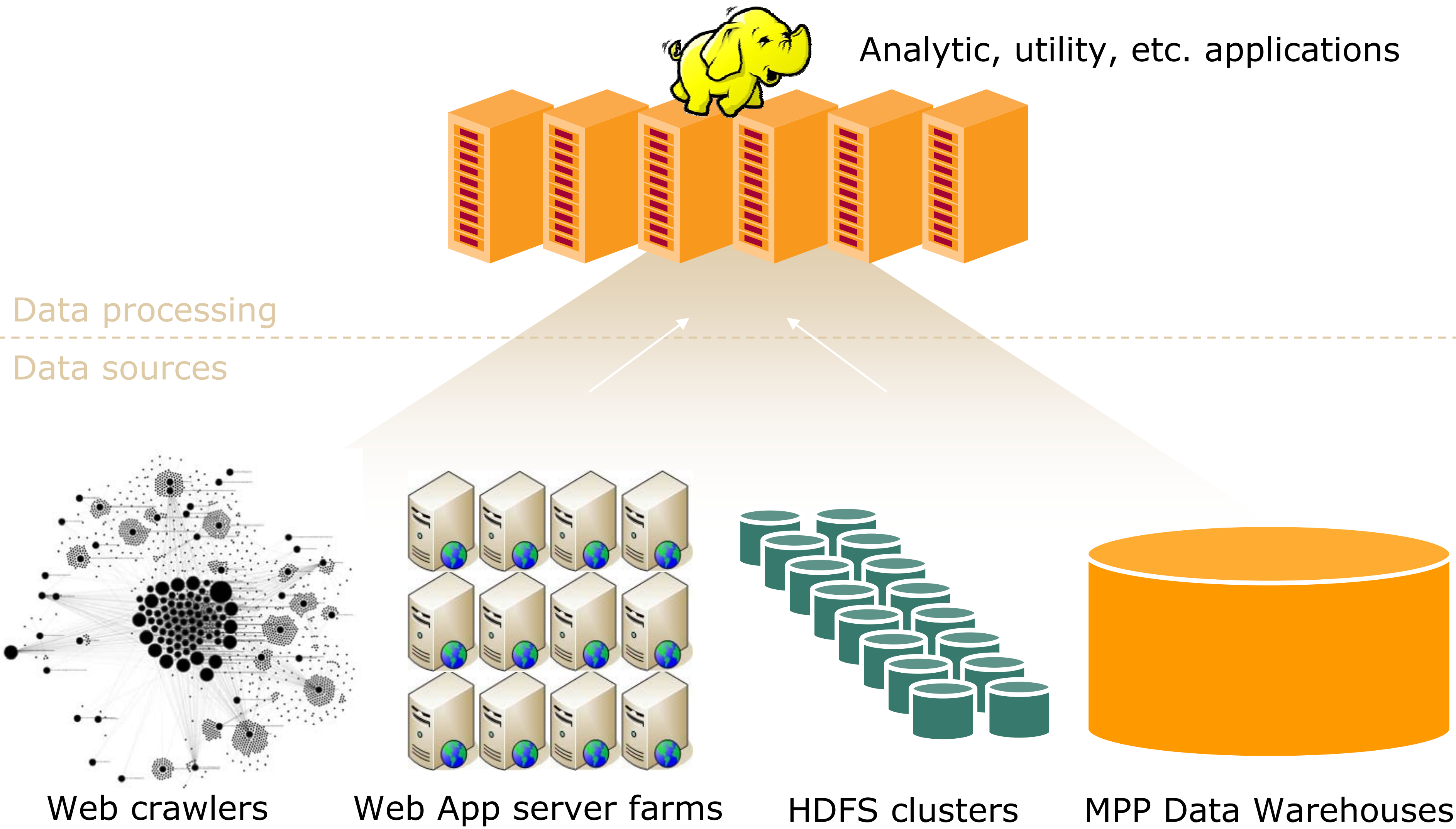
- Use cases
  - > ETL on steroids
  - > Hadoop as a BI Tool
  - > Hadoop as a BI application

- Teradata Developer Exchange
  - > UDF examples



http://developer.teradata.com/extensibility/articles/hadoop-dfs-to-teradata
http://developer.teradata.com/extensibility/articles/sessionization-map-reduce-support-in-teradata

Hadoop Parallel Processing



TERADATA
Raising Intelligence

When to Use Which Infrastructure

MapReduce

Complex processes
Multi-step processes
1,000+ nodes required
Can't move the data
Extensive text parsing
Householding analysis

"It depends"

Data mining
Simple reports
Data cleansing

RDBMS

Iterative discovery
Drill down, OLAP
Dashboards
Business user querying
Integrated subject areas
Multidimensional views
Trends over time
Visualization tools
Text analysis tools

Conclusion

Keep a close eye on MapReduce, but *don't give up your RDBMS just yet!*

- MapReduce vs RDBMS
  - Surface similarities, but deep differences
  - RDBMS performance far superior to Hadoop
  - Still immature but expect "Deficiencies" to be solved over time
- <u>Complementary</u> rather than Competitive technologies
  - Unstructured datasets
  - One/few-time processing (e.g. web-logs)
- Key Criterion - $/Query
  - Large # queries on dataset – consider RDBMS
  - Quick time-to-value for few queries – consider MapReduce
  - Expect MapReduce high cost/query to drop over time
- Consider MapReduce for
  - ETL processing
  - Parallel programming

Acknowledgements and Research material

- "MapReduce: Simplified Data Processing on Large Clusters", Dean and Ghemawat, Google
- Internal paper – "MapReduce and Hadoop Technical Overview", Dan Graham and Rick Burns, Teradata
- "Tutorial: What is MapReduce and Hadoop", Dan Graham and Rick Burns, Teradata
- "MapReduce, Hadoop, and Data Warehousing Coexistence" ", Dan Graham and Rick Burns, Teradata
- "Pig Latin: A Not-So-Foreign Language for Data Processing", Olston et al
- "Hive A Warehousing Solution Over a MapReduce Framework", Thusoo et al
- "MapReduce and Parallel DBMSs: Friends or Foes?", Stonebraker et al, Communications Of The ACM | January 2010
- "MapReduce: A major step backwards", Dewitt and Stonebraker,
- "A Comparison of Approaches to Large-Scale Data Analysis", Pavlo et al, SIGMOD'09, June 29–July 2, 2009

Special thanks to Dan Graham, Teradata Corporation

TERADATA
Raising Intelligence