



Hadoop : Big Data or Big Deal

Eduard Erwee

Introduction



- ▶ Eduard Erwee
- ▶ Data Soil Ltd (www.datasoil.uk)
- ▶ Background
 - ▶ Working with Microsoft data products over 20 years
 - ▶ MCSD VB6, SQL Server 7
 - ▶ 5 years as Microsoft Certified Trainer
 - ▶ 4 years as SQL Server PFE, Reading - UK
 - ▶ Today, clean data toilets for the highest bidder
 - ▶ **No Linux / No Big Data** (until 9 months ago)

DATA SOIL LTD
GROWING YOUR BUSINESS



Agenda

- ▶ A) What is Big data?
 - ▶ i) Origins
 - ▶ ii) Technologies & Terminologies
 - ▶ iii) The Players
- ▶ B) How is Big Data Different?
 - ▶ i) Philosophies
- ▶ C) How to ride the Elephant?
 - ▶ i) All about the tools
 - ▶ ii) Sources of Inspiration
- ▶ D) BIG to the Future!
 - ▶ i) Current Common Use-cases
 - ▶ ii) Future Opportunities
- ▶ E) Summary
- ▶ F) Conclusion
- ▶ G) Q&A



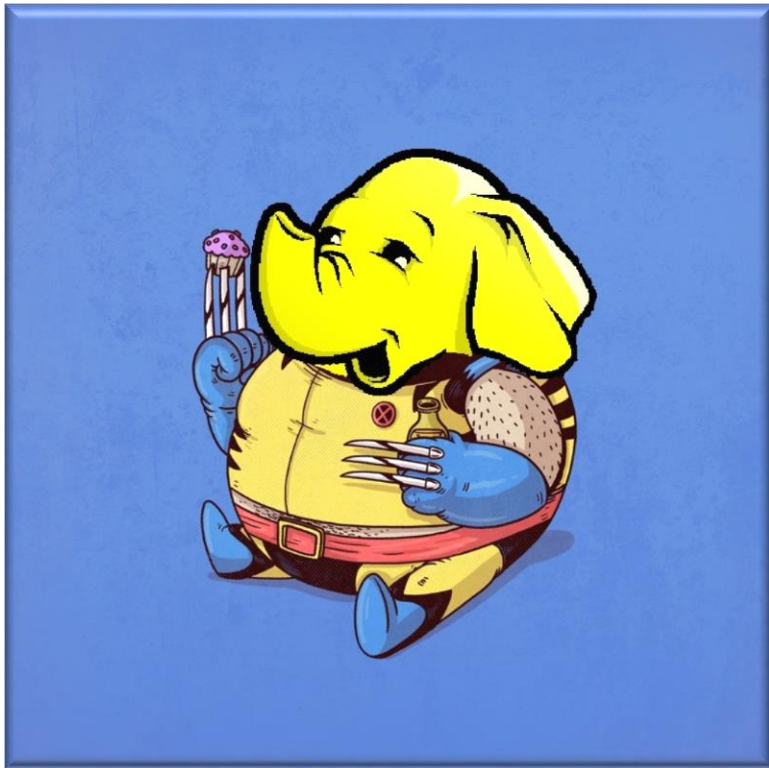
What is Big data?

- ▶ i) Origins
 - ▶ Nutch-to-Google-to-Yahoo and beyond
 - ▶ Apache Who??
- ▶ ii) Technologies & Terminologies
 - ▶ Core Hadoop
 - ▶ Hive
 - ▶ HCatalog
 - ▶ Pig
 - ▶ Sqoop
 - ▶ Oozie
 - ▶ HUE (flavours-of)
 - ▶ Mahout
 - ▶ Loads of others
 - ▶ Ha-dump!
- ▶ iii) The Players
 - ▶ The Big 3
 - ▶ One to Watch : Cascading & Lingual



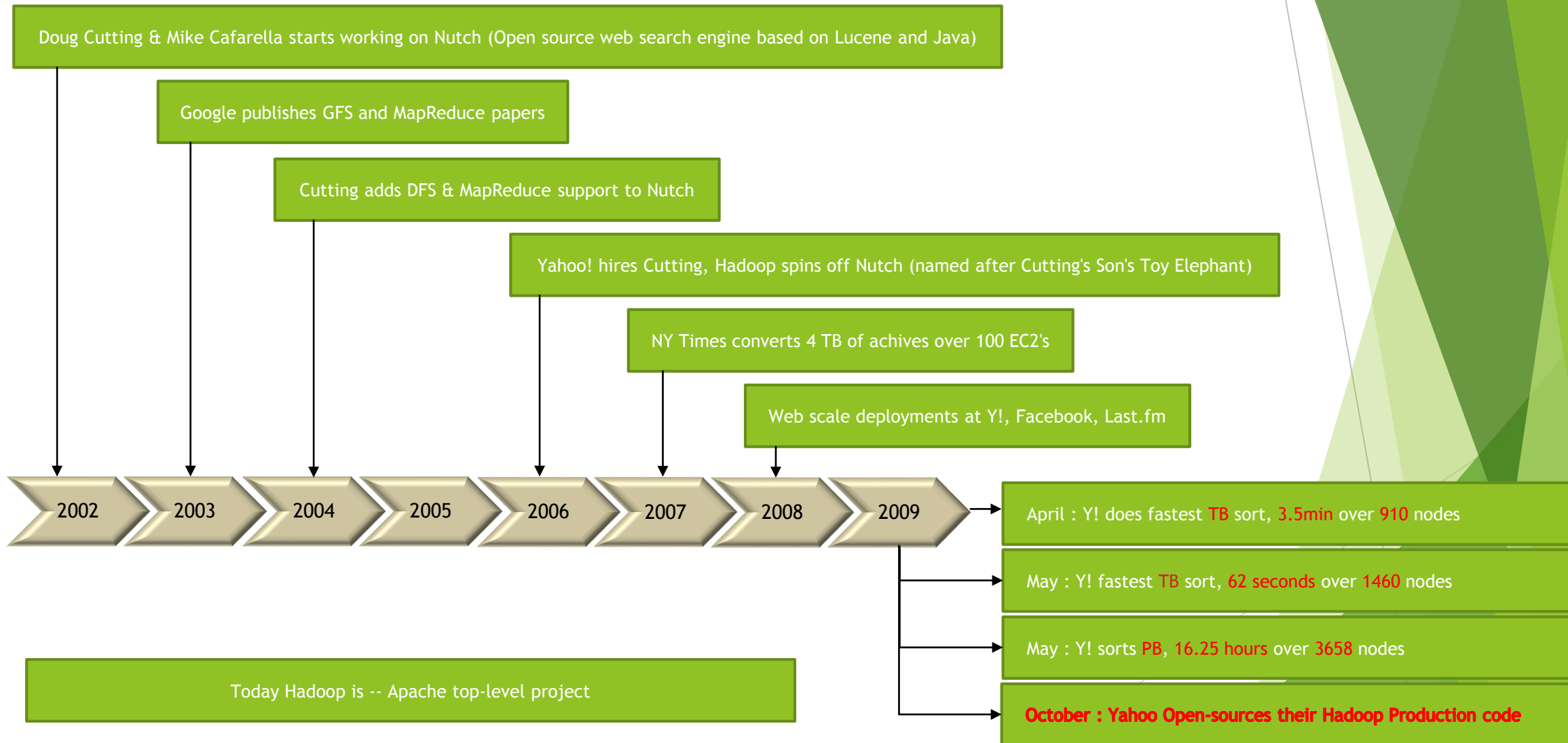
i) Origins

- ▶ Nutch-to-Google-to-Yahoo and beyond
- ▶ Apache Who??





Nutch-to-Google-to-Yahoo and beyond





Apache Who??



The Apache Software Foundation

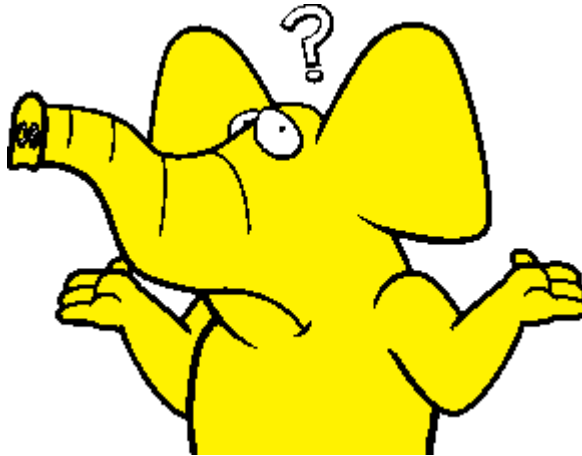
Community-led development since 1999.

- ▶ The Apache Software Foundation (<http://www.apache.org/>)
- ▶ The ASF is made up of nearly 150 Top Level Projects (Big Data and more)
 - ▶ Most of the Hadoop components we will discuss



ii) Technologies & Terminologies

- ▶ Core Hadoop
 - ▶ Hadoop Common:
 - ▶ Hadoop Distributed File System (HDFS™)
 - ▶ Hadoop MapReduce:
 - ▶ Hadoop YARN
- ▶ HUE (flavours-of)
- ▶ Hive
- ▶ HCatalog
- ▶ Pig
- ▶ Sqoop
- ▶ Oozie
- ▶ Mahout
- ▶ Loads of others
- ▶ Ha-dump!

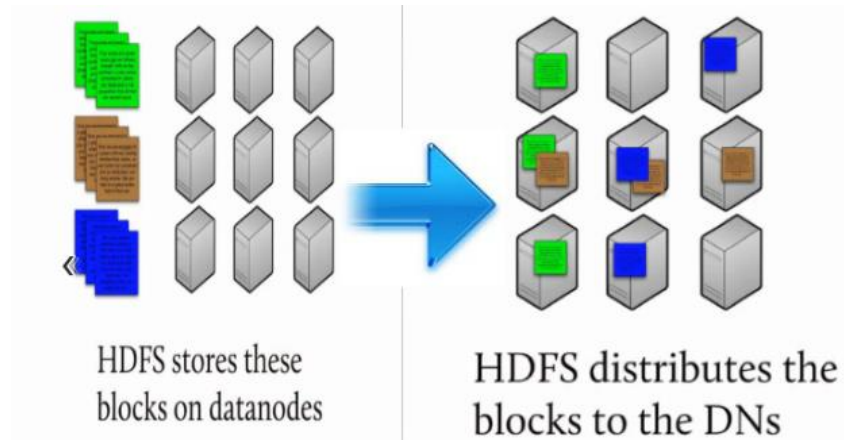
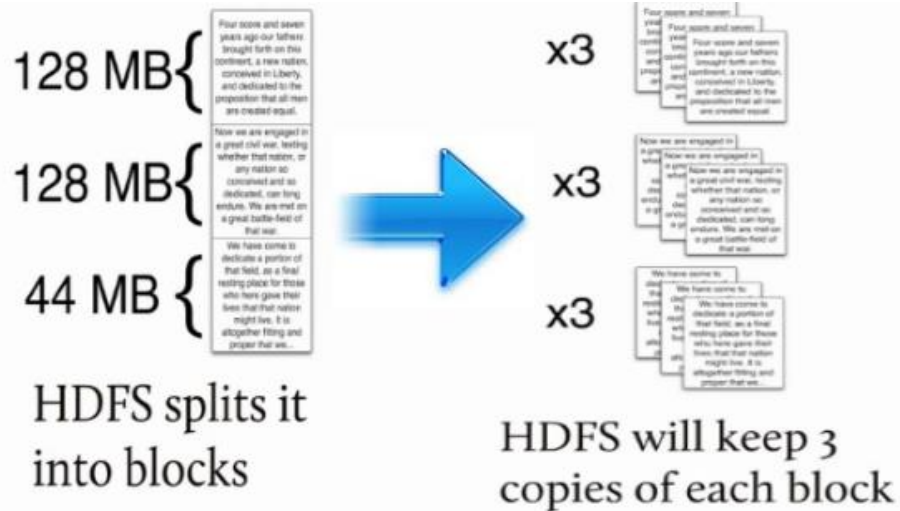




Core Hadoop



- ▶ Hadoop Common:
 - ▶ The common utilities that support the other Hadoop modules.
- ▶ Hadoop Distributed File System (HDFS™):
 - ▶ A distributed file system that provides high-throughput access to application data.



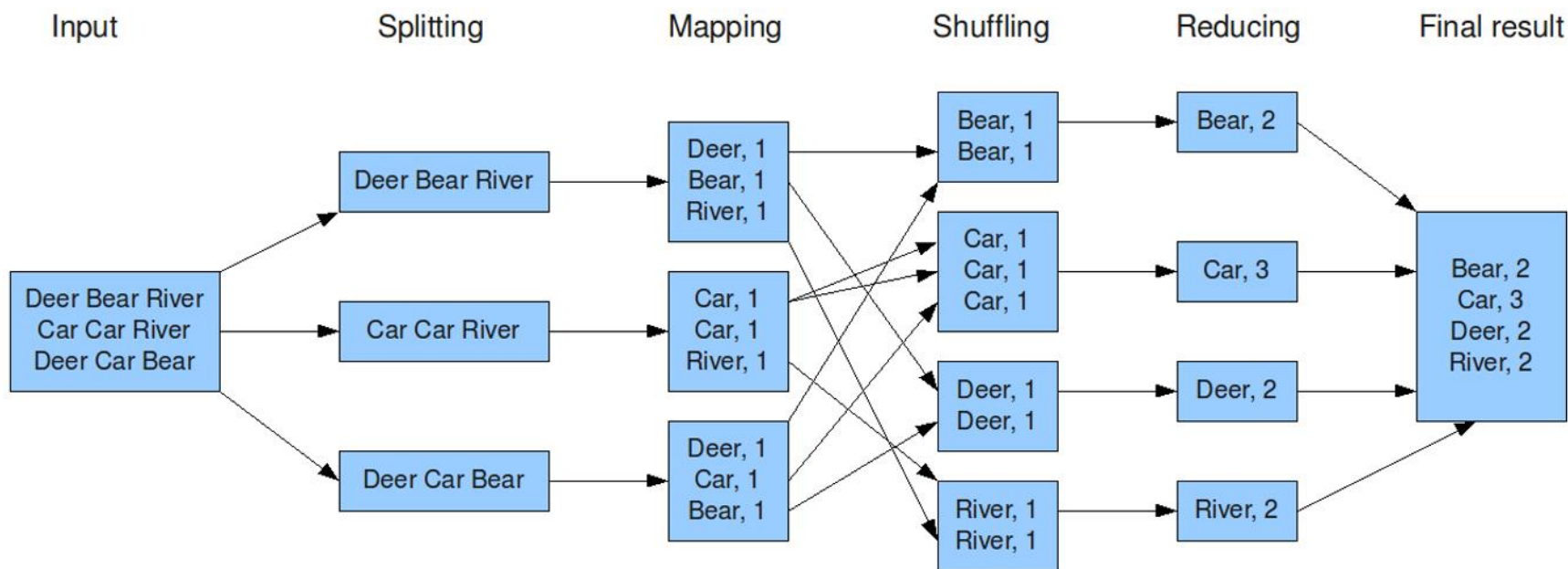


Core Hadoop

► Hadoop MapReduce

Take word counting as an example, something that Google does all of the time.

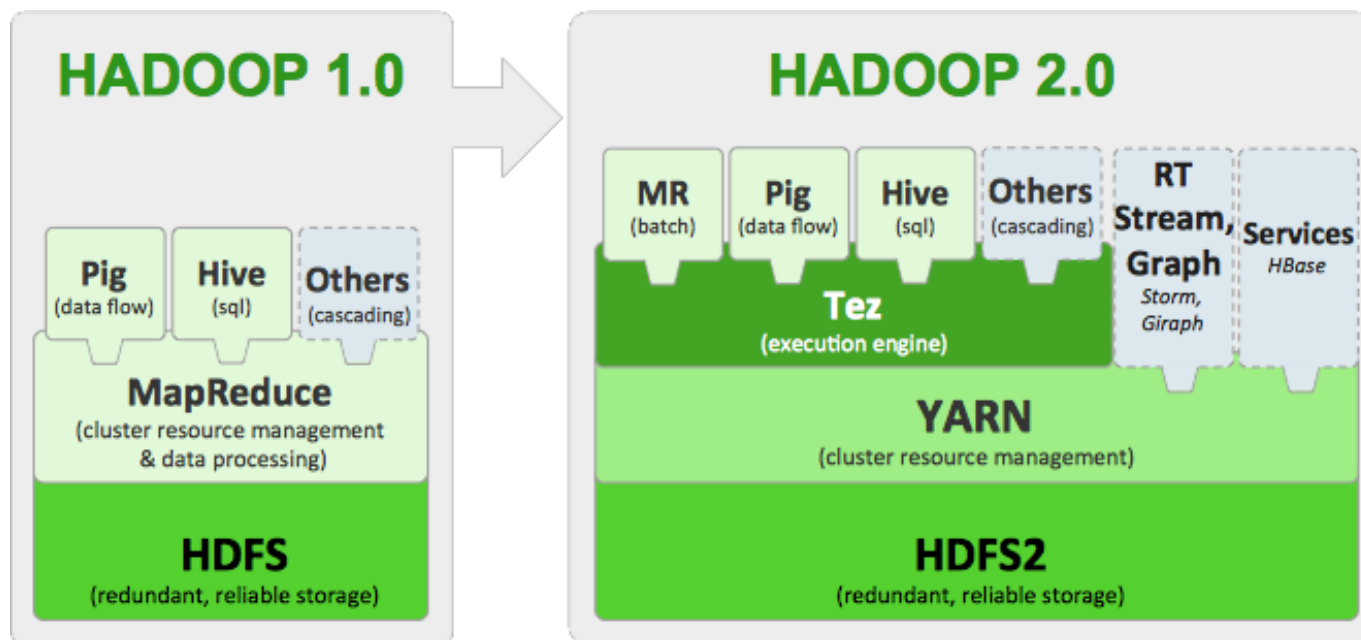
The overall MapReduce word count process





Core Hadoop

- ▶ Hadoop MapReduce (continues):
 - ▶ MapReduce-V2
 - ▶ A YARN-based system for parallel processing of large data sets.
 - ▶ Built on top of Tez



- ▶ Hadoop YARN (Yet Another Resource Negotiator):
 - ▶ A framework for job scheduling and cluster resource management.



HUE (flavours-of)



- ▶ Hue aggregates the most common Apache Hadoop components into a single UI.
- ▶ "Just use" Hadoop web based interface without worrying command line.

The screenshot displays the Hue Query Editor interface. The main window shows a SQL query in the editor:

```
1 select * from nyse_stocks
2 where nyse_stocks.stock symbol = 'IBM'
3 and date between '2001-12-01' and '2001-12-31'
4 order by date asc
```

Below the query editor, there are several settings and execution options:

- SETTINGS:** A key-value pair is set to `ABORT_ON_ERROR` with a value of `1`.
- PARAMETERIZATION:** The checkbox `Enable Parameterization` is checked.
- EMAIL NOTIFICATION:** The checkbox `Email me on completion` is unchecked.

At the bottom of the editor, there are buttons for `Execute`, `Save`, `Save as...`, `Explain`, and `or create a`.

On the right side, a larger view of the Query Editor shows a different SQL query:

```
1 SELECT s07.description, s07.total_emp, s08.total_emp, s07.salary
2 FROM
3   sample_07 s07 JOINzzzz
4   sample_08 s08
5 ON ( s07.code = s08.code )
6 WHERE
7 ( s07.total emp > s08.total_emp
8 AND s07.salary > 100000 )
9 SORT BY s07.salary DESC
10
11
12
13
14
15
16
17
18
19
```

Below this query, there are buttons for `Execute`, `Download`, `Save as...`, `or create a`, and `New query`.



Hive



- ▶ Managing large datasets residing HDFS.
- ▶ Mechanism to query the data using a SQL-like language called HiveQL.
- ▶ Runs in HUE

The screenshot shows the HUE web interface. The browser address bar displays the URL `192.168.63.159:8000/beeswax/execute/8`. The interface includes a navigation bar with tabs for 'Query Editor', 'My Queries', 'Saved Queries', 'History', 'Databases', 'Tables', and 'Settings'. The main content area contains a HiveQL query:

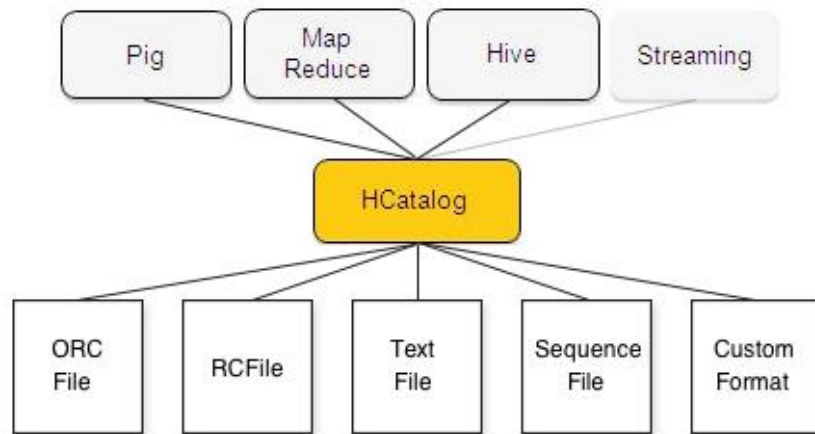
```
1 select * from nyse_stocks
2 where nyse_stocks.stock_symbol = 'IBM'
3 and date between '2001-12-01' and '2001-12-31'
4 order by date asc
```

Below the query editor, there are sections for 'USER-DEFINED FUNCTIONS' (with an 'Add' button), 'PARAMETERIZATION' (with a checked 'Enable' checkbox and a 'Parameterization' link), and 'EMAIL NOTIFICATION' (with an unchecked 'Email me on completion' checkbox). At the bottom, there are buttons for 'Execute', 'Save', 'Save as...', 'Explain', and 'New query'.



HCatalog

- ▶ Built on top of the Hive metastore and incorporates Hive's DDL
- ▶ HCatalog's table abstraction - presents relational view - of data in (HDFS)
- ▶ Removes worry about format their data is stored



- ▶ For me - Very similar to a set of views in SQL Server over staging feeds
- ▶ Exposed to Pig / Map Reduce / Hive
- ▶ Runs in HUE



HCatalog - Sample

File options

Input File

Encoding

Delimiter

Replace delimiter with

Single line comment

Read column headers Import data

Autodetect delimiter Ignore whitespaces

Java-style comments Ignore tabs

Table preview

Column name	Column name	Column name	Column name	Column name	Column name
<input type="text" value="exchange"/>	<input type="text" value="stock_symbol"/>	<input type="text" value="date"/>	<input type="text" value="stock_price_open"/>	<input type="text" value="stock_price_high"/>	<input type="text" value="stock_price_low"/>
Column type	Column type	Column type	Column type	Column type	Column type
<input type="text" value="string"/>	<input type="text" value="string"/>	<input type="text" value="string"/>	<input type="text" value="double"/>	<input type="text" value="double"/>	<input type="text" value="double"/>
Row #1 NYSE	ASP	array	12.55	12.8	12.42
Row #2 NYSE	ASP	bigint	12.5	12.55	12.42
Row #3 NYSE	ASP	binary	12.59	12.59	12.5
Row #4 NYSE	ASP	boolean	12.45	12.6	12.45
Row #5 NYSE	ASP	decimal	12.61	12.61	12.61
Row #6 NYSE	ASP	double	12.4	12.78	12.4
Row #7 NYSE	ASP	float	12.35	12.58	12.35
Row #8 NYSE	ASP	int	2001-12-19	12.42	12.6
Row #9 NYSE	ASP	map	2001-12-18	12.37	12.5
		smallint			
		string			
		timestamp			
		tinyint			



Pig

- ▶ Pig is a high-level platform used for creating MapReduce.
- ▶ The programming language is called Pig Latin
- ▶ Optimizer turns Pig into optimized Java Mapreduce.



```
1 a = LOAD 'nyse_stocks' using org.apache.hcatalog.pig.HCatLoader ();
2 b = filter a by stock_symbol == 'IBM';
3 c = group b all;
4 d = foreach c generate AVG(b.stock_price_open) , AVG(b.stock price close);
5 dump d;
```

- ▶ Similar to M in Power Query
- ▶ It's the VB.net Vs C++ debate all over again.
- ▶ Structure
 - ▶ Hive require data to be more structured
 - ▶ Pig allows you to work with unstructured data.
- ▶ Compatible with Hcatalog
- ▶ Runs in Hue



The Job job_1404726436893_0007 has been started successfully.
You can always go back to Query History for results after the run.

```
(109.02535999999999,109.08895999999983)
```




Sqoop

- ▶ Apache Sqoop(TM) is a tool designed for efficiently transferring bulk data between Apache Hadoop and structured datastores such as relational databases.
- ▶ Runs in Hue





Oozie



- ▶ Workflow scheduler system to manage Apache Hadoop jobs.
- ▶ Oozie Coordinator jobs
 - ▶ Recurrent Oozie Workflow
 - ▶ Jobs triggered
 - ▶ by time (frequency)
 - ▶ data availability.
- ▶ Integrated with the rest of the Hadoop stack
- ▶ Scalable, reliable and extensible system.
- ▶ Available in HUE



Mahout



- ▶ Goal : scalable machine learning library.
- ▶ Examples of Mahout use cases:
 - ▶ Recommendation mining
 - ▶ takes users' behaviour and from that tries to find items users might like. (Netflix)
 - ▶ Clustering
 - ▶ Group documents, web pages and articles based on
 - ▶ contained topics
 - ▶ their related documents.
 - ▶ Most common use of this is search engines, which cluster pages based on keywords, page links, etc.
 - ▶ Classification
 - ▶ Based on prior categorization of documents
 - ▶ Evaluates new documents and determine best categories.
 - ▶ Filter new mail into INBOX
 - ▶ Auto-organize new content
 - ▶ flag potential spam comments.



Loads of others



Apache Ambari
<http://incubator.apache.org/ambari>



Apache Kafka

A high-throughput distributed messaging system.

KNOX



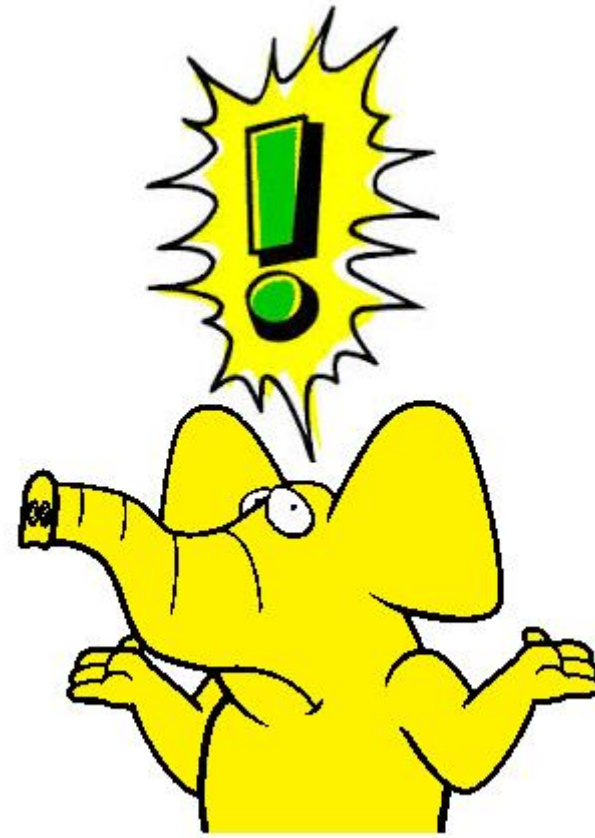


Ha-dump!



Steaming pile
of Data

Store



Inside the Elephant !?



iii) The Players



- ▶ The Big 3
- ▶ One to Watch : Cascading & Lingual



The Big 3

- ▶ Hortonworks claims to be the only fully open source distribution.
- ▶ Cloudera is close on their heels with everything based on open source but has some additional maintenance and installation functionality that is proprietary
- ▶ MAP-R on the other hand re-wrote the storage engine from scratch to improve performance at the cost of being vendor specific
- ▶ My Opinion ?
- ▶ Benchmarking -- Altoros



Altoros did some significant benchmarking between the 3, and can be found here:
http://www.althoros.com/hadoop_benchmark.html



One To Watch : Cascading & Lingual



- ▶ Developed by Chris Wensel & Team from Concurrent:
 - ▶ <http://www.concurrentinc.com/>
- ▶ Cascading is a development platform for building data applications on Hadoop
 - ▶ Developed on top of Cascading:
 - ▶ Lingual
 - ▶ Simplifies systems integration -- ANSI SQL compatibility -- JDBC driver
 - ▶ Pattern
 - ▶ Machine learning scoring algorithms through PMML compatibility
 - ▶ Scalding
 - ▶ Enables development with Scala, a powerful language for solving functional problems
 - ▶ Cascalog
 - ▶ Enables development with Clojure, a Lisp dialect
 - ▶ Driven
 - ▶ Understand data usage + accelerate Cascading application development and management



Driven -- Visualize Development of Flows

- ▶ Like SSMS Execution Plans
- ▶ Breaks up Query
- ▶ Shows Data flow
 - ▶ Drill down

The screenshot displays the DRIVEN application interface. At the top, there's a navigation bar with 'All Applications' and a 'Running' status for 'TPCDS_Q7'. Below this, a summary bar shows 'App: tpcds_q7' with details like 'Owner: jposner', 'Jar Info: load-hadoop-20140502.jar', 'Platform: Apache.Hadoop.1.1.2', and 'Run Time: 11m 4s'. It also shows 'Tuples read: 7.77 M', 'Tuples written: 4.01 M', 'Bytes read: 2.01 GB', and 'Bytes written: 1.81 GB'. The main area features a flow diagram with nodes like 'FilterDateDim', 'FilterCustomerDemographics', 'FilterStore', 'JoinsAgainstStoreSales', 'RemoveExtraFields', 'CalculateAverageCouponAmount', 'CalculateAverageListPrice', 'CalculateAverageSalePrice', 'CalculateAverageQuantity', 'GenerateReport', and 'FormatReport'. Below the diagram is a 'Flows' table with columns for Status, Name, Default Duration, Type, Bytes Read, Bytes Written, and Timeline.

Status	Name	Default Duration	Type	Bytes Read	Bytes Written	Timeline
Summary:	tpcds_q7	11m 20s	app	2.01 GB	1.81 GB	
✓	FilterDateDim	21s	flow	9.84 MB	168.45 KB	
✓	FilterCustomerDemographics	30s	flow	76.93 MB	1.12 MB	
✓	FilterStore	25s	flow	5.15 KB	121.33 KB	
✓	JoinsAgainstStoreSales	10m 17s	flow	1.91 GB	1.81 GB	
✓	RemoveExtraFields	15s	flow	6.08 MB	413.01 KB	
⏸	CalculateAverageCouponAmount	0s	flow	0	0	
⏸	CalculateAverageListPrice	15s	flow	146.8 KB	82.53 KB	
⏸	CalculateAverageQuantity	0s	flow	0	0	



Driven -- Application Insights

- ▶ Drill down into steps
 - ▶ Execution Time
 - ▶ Bottle-necks
 - ▶ Resource usage

The screenshot displays the DRIVEN application monitoring interface. At the top, the status is 'LOAD Successful'. Below this, key statistics are provided: Owner: hadoop, Version: 20140302, Jar Info: load-20140302.jar, App Tags: (undefined), Platform: Apache.Hadoop.1.0.3, Frameworks: (undefined), Tuples trapped: 13.47 K, Slice retrys: 0, Run Time: 9m 22s, Progress: 100%, Tuples read: 8.36 M, Tuples written: 6.55 M, Bytes read: 3.6 GB, Bytes written: 3.27 GB.

A flow diagram shows the execution path: DateDimFilter and ItemFilter feed into InnerJoins, which then feeds into OuterJoin, and finally into FinalReport.

The 'InnerJoinToCatalogReturn' window is open, showing the following details:

- Name: InnerJoinToCatalogReturn
- Type: HashJoin
- Used at: cascading.load.tpc.ds.query.Q40.java:186

Join Keys:

CatalogReturn	InnerJoinResults
cr_item_sk	cs_item_sk
cr_order_number	cs_order_number

Output:




```
cr_returned_date_sk
cr_returned_time_sk
cr_item_sk
cr_refunded_customer_sk
cr_refunded_cdemo_sk
cr_refunded_hdemo_sk
cr_refunded_addr_sk
cr_returning_customer_sk
cr_returning_cdemo_sk
cr_returning_hdemo_sk
cr_returning_addr_sk
cr_call_center_sk
cr_catalog_page_sk
```

Flows table:

Status	Name	Total Duration	Tuples Trapped	Timeline
Summary:	load	10m 23s	13,472	
✓	DateDimFilter	1m 42s	0	
✓	ItemFilter	1m 22s	0	
✓	InnerJoins	7m 16s	0	
✓	OuterJoin	8m 32s	0	
✓	FinalReport	10m 0s	13,472	



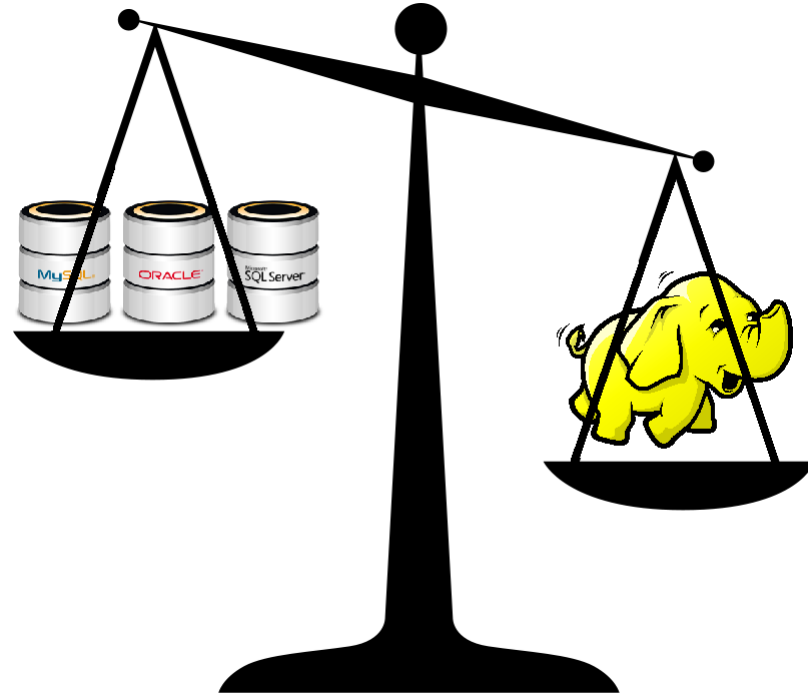
Why Watch : Cascading & Lingual ?

- ▶ All 3 Big data platform vendors mentioned before
 - ▶ supports Cascading integration
 - ▶ investing in ensuring continued support for Cascading on their own platforms
- ▶ Used by   
- ▶ Single platform to develop code on that evolves with changing big data landscape.
- ▶ Single JAR deployment.
- ▶ Ansi-92 interface via JDBC for moving data between systems / platforms
- ▶ All Open-Source (no vendor lock-in)
- ▶ Data Soil is contributing to develop the SQL Server Plug-in for Cascading & Lingual.
 - ▶ (see our blogs for getting into Cascading using Microsoft Technologies)



B) How is Big Data Different?

- ▶ Philosophies
 - ▶ Current Architecture vs Schema-On-Read
 - ▶ S-O-R : Advantages & Disadvantages
 - ▶ Integration with SQL Server & Windows





Current Architecture vs Schema-On-Read

Current BI Architecture	Big Data BI Architecture
Get Business Requirements and prioritize	Get Business Requirements and prioritize
Find / Collect all relevant data sources	All Data is already in the Ha-dump
Normalize / copy to staging / create structures / schemas / ETL	Create schema for question 1 / ETL
Create Warehouse / Cube	Send processing instructions to data
Start answering questions 1 / 2 / 3 / 4 / 5	Answer question 1 {& Repeat}



S-O-R : Advantages & Disadvantages

▶ Advantages

- ▶ Store first, ask questions later
 - ▶ Storage is cheap compare to high availability SAN
 - ▶ Format agnostic as not pre-normalization / conversion required
- ▶ All data is available in a central place
- ▶ High degree of parallel processing → speeds up large batch processing
- ▶ Possible to start answering business questions quicker

▶ Disadvantages

- ▶ New skillsets & training required
- ▶ Company may not support new software stack
- ▶ Creating new schemas for proprietary data can be difficult



Integration with SQL Server & Windows

▶ ODBC

- ▶ Hortonworks / Cloudera / MAPR all have supported ODBC drivers
- ▶ Create Linked Servers directly from SQL Server
- ▶ SSIS integration
- ▶ Pull Data directly into Excel (see Hortonworks Sandbox)

▶ JDBC & Other

- ▶ Tableau / squirrel-sql / Revolution R / Business Objects ext.

▶ Other ETL Tools

- ▶ Talend (to be discussed later)

▶ Local Install

- ▶ Hortonworks Data Platform (HDP)
- ▶ HDInsight Emulator



C) How to ride the Elephant?

- ▶ i) All about the tools
 - ▶ Local VM platform providers
 - ▶ Online platform providers
 - ▶ Vagrant
 - ▶ Talend
 - ▶ Reuse of old machines
- ▶ ii) Sources of Inspiration
 - ▶ Sandbox's
 - ▶ The Apache Software Foundation
 - ▶ Github





i) All about the tools

- ▶ Local VM platform providers
- ▶ Online platform providers
- ▶ Vagrant
- ▶ Talend
- ▶ Pet Project : Reuse of old machines



Local VM platform providers

- ▶ Hyper-V (Microsoft)
 - ▶ Windows Server
 - ▶ Windows 8.1
- ▶ VMWARE
 - ▶ VMWARE Server Products
 - ▶ Workstation - On Windows
 - ▶ Personally, I absolutely LOVE Workstation 10.0
 - ▶ Fusion - On Mac
- ▶ Virtual Box (Oracle)
 - ▶ Runs on EVERYTHING
 - ▶ Close second favourite
 - ▶ Integrates extremely well with Vagrant (to be discussed)



Microsoft
Hyper-V



vmware®





Online platform providers

- ▶ Azure & Big Data

- ▶ HD-Insight (Based on Hortonworks HDP platform)

- ▶ Real World Big Data (SQL-Bits Session)

- ▶ Adam Jorgensen / John Welch

- ▶ Restored my confidence in MS Big Data Cloud Solutions

- ▶ Amazon Cloud (AWS)

- ▶ EC2

- ▶ Host of supporting services





Vagrant



- ▶ Vagrant provides
 - ▶ easy to configure,
 - ▶ reproducible,
 - ▶ and portable work environments built on industry standards.
- ▶ Spins up / Hibernates / Destroys complex development environments with one line of code
- ▶ Supports Virtualbox / VMWARE / Docker / Hyper-V / Custom Providers
- ▶ Ability to spin up environments locally or directly to Amazon EC2



Talend

talend*
*open data solutions

- ▶ Enterprise grade development environment for creating data integration across just about anything.

Talend Open Studio for Big Data BASIC - Free

Eclipse-Based Tooling

Hadoop 2.0 and YARN Support

Big Data ETL and ELT

HDFS, HBase, HCatalog, Hive, Pig, Sqoop Components

Job Designer

Apache License 2.0

Broadest NoSQL Support

Fully Open Source

<http://www.talend.com/download>



Talend (i)

Talend Open Studio (3.2.0.M1_r26328) | TALENDEMOSJAVA (Connection: Local)

File Edit View Window Help

100%

Repositor Selezione

Business Models
Job Designs
t01_Compo
CustomCode
Databases
Bulk
InOut
SCD
step1CreateTable
step2ModifyData
tMysqlSCD 0.1
SP
ComponentRow 0.1
Connection 0.1
DataQuality
ELT
File
Internet
LogError

Job step2ModifyData 0.1

To retrieve the idstate max
tMysqlInput_3 → row3 (Main) → tSetGlobalVar_1

OnSubJobOk

Insert one line where idstate= idstate max
tRowGenerator_1 → row1 (Main) → tMysqlOutput_1
INSERT

OnSubJobOk

Updates all the lines (puts in upcase all the labelstates)
tMysqlInput_1 → FromStates (Main) → tMap_1 → ToStates (Main) → tMysqlOutput_2
UPDATE

Designer Code

Job(step2) Contexts(J) Component tSetGlobalVar_1 Run (Job st) Problems Modules Talend Exc Scheduler Job Hierarc

Basic settings
Advanced settings
Dynamic settings
View
Documentation

Key	Value
"idStatesMax"	row3.max+1

Outline
Code View

- tMap_1
- tMysqlInput_1
- tMysqlInput_2
- tMysqlInput_3
- tMysqlOutput_1 (tMysqlOutput_1

- tMysqlOutput_2 (tMysqlOutput_2

- tMysqlOutput_3 (tMysqlOutput_3

- tRowGenerator_1
- tSetGlobalVar_1

Palette
Find component... OK

- Business
- Business Intelligence
- Custom Code
- Data Quality
- Databases
Interbase
- JavaDB
- LDAP
- MS SQL Server
- MaxDB
- MySQL
- Netezza
- Oracle
tOracleBulkExec
tOracleCommit
tOracleConnection
tOracleInput
tOracleOutput
- ELT
- File
- Internet
- Logs & Errors
- Misc
- MultiSchema
- Orchestration
- Processing
- System
- XML

Start Talend Open Studio... data-integration ftp.robertomarchet... Spoon - kettle_loadi... talend_vs_kettle_p... Talend Open Studi... 10.32



Talend (ii)

The screenshot displays the Talend Open Studio interface with a job design titled "Discovery job". The job flow is as follows:

- File Fetch over HTTP** (tFileFetch_1) connects to **Orders - delimited file** (tFileInputDelimited_1).
- OnSubJobOk** (tOnSubJobOk) triggers the start of the main flow.
- row1 (Main)** (tMap_1) performs a **Mapping** operation, pulling data from **Customers - lookup MySQL database** (tMysqlInput_2).
- The flow splits into three paths:
 - validOrders (Main order:1)** (tFilterRow) leads to **Valid Orders - Oracle Bulk** (tOracleOutputBulkExec_1).
 - RejectedOrders (Main order:2)** (tFilterRow) leads to **Rejected Orders XML File** (tFileOutputXML_1).
 - row3 (Main)** (tLogCatcher_1) leads to **Log table MSSQL** (tMssqlOutput_1).

The interface includes a left-hand **Repository** pane with categories like Business Models, Job Designs, and Code. A right-hand **Palette** lists various components such as tExtractPositionalFields, tAggregateRow, and tSortRow. At the bottom, the **Job DataDiscovery** configuration pane shows a **Debug** section with a **Traces Debug** button and a **Line limit** of 100.



Talend

Supported Database & Data Source Connectivity

Amazon RDS	HIVE	Oracle
Amazon Redshift	HSQLDB	ParAccel
Amazon S3	Informix	PostgresSQL
AS400	Ingres	PostgresPlus
DB2	InterBase	SAS
Derby DB	JavaDB	SQLite
Exasol	JDBC	Sybase
eXist-db	MaxDB	Teradata
Firebird	Microsoft OLE-DB	VectorWise
Google Storage	Microsoft SQL Server	Vertica
Greenplum	MySQL	Windows Azure Blob Storage
H2	Netezza	



Pet project : Reuse of old machines

- ▶ Challenge your manager
- ▶ If you can build a cluster from your old desktops that will outperform his current development server, he has to give you a raise!

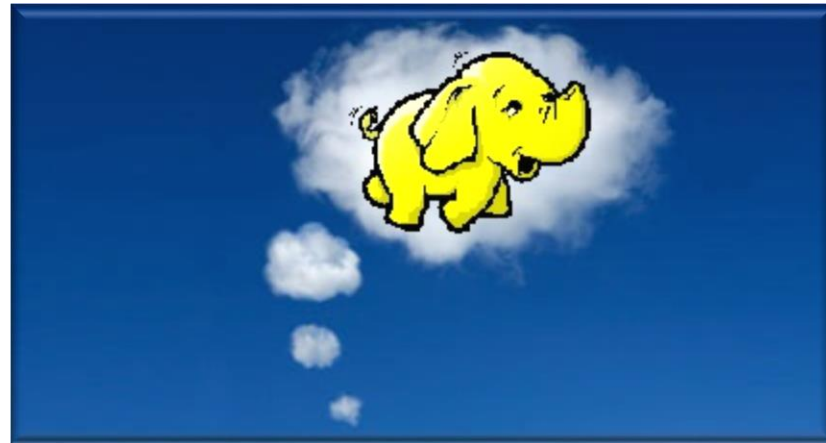


- ▶ You'd be surprised what you can do with a pile of these!



ii) Sources of Inspiration

- ▶ Sandbox's
- ▶ The Apache Software Foundation
- ▶ Github





Sandbox's

- ▶ All three the Big Data Players have their pre-built Sandbox's you can download and experiment with
- ▶ Hortonworks
 - ▶ Current Version 2.1
 - ▶ Supports: VirtualBox / VMWare / Hyper-V
- ▶ Cloudera
 - ▶ Current Version CDH 5.0.x
 - ▶ Cloudera Live online (beta)
 - ▶ Supports: VirtualBox / Vmware / Linux KVM (Kernel-based Virtual Machine)
- ▶ MAPR
 - ▶ Supports: VirtualBox / Vmware
- ▶ Cascading & Lingual
 - ▶ Vagrant Image that spins up 4 Node Cluster via GitHub
 - ▶ Supports: VirtualBox



The Apache Software Foundation

- ▶ Want to know about BIG future technologies
- ▶ Apache Incubator - (<http://incubator.apache.org/>)
 - ▶ Tez → Speed up MapReduce
 - ▶ Storm → high-performance realtime computation system
 - ▶ Optiq → SQL interface & advanced query optimization - non-RDBMS systems
 - ▶ Falcon → quickly onboard their data, associated processing & management tasks on Hadoop clusters



GitHub



- ▶ GitHub is a web-based hosting service based on Git.
- ▶ Git a distributed revision control and source code management (SCM) system initially designed and developed by Linus Torvalds for Linux kernel development
- ▶ Great source of Vagrant-Based VM's
 - ▶ Cascading & Lingual Cluster (Get Vagrant & Virtual Box)
 - ▶ <https://github.com/Cascading/vagrant-cascading-hadoop-cluster>



D) BIG to the Future!

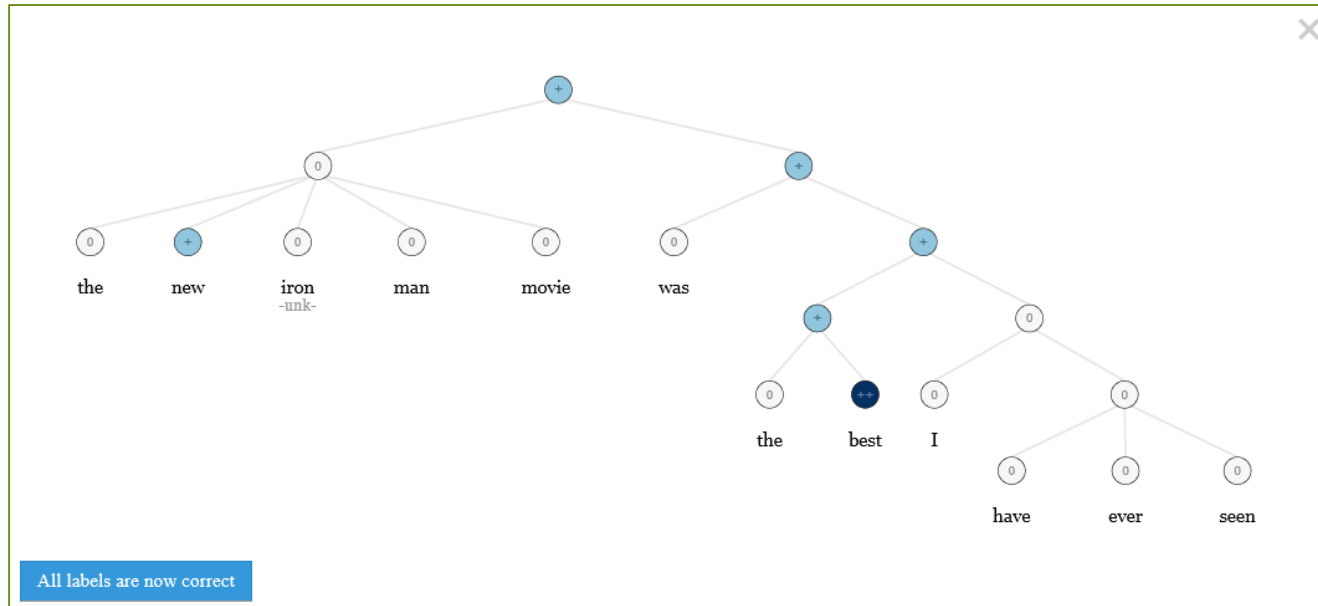
- ▶ i) Current Common Use-cases
- ▶ ii) Future Opportunities





i) Current Common Use-cases

- ▶ Sentiment (twitter feeds / wordpress scrapes / facebook likes)
 - ▶ Natural Language Processing : Stanford (<http://nlp.stanford.edu:8080/sentiment/rntnDemo.html>)



- ▶ Recommendation Engines using Mahout / Other (Netflix)
- ▶ Anti Money Laundering ??
 - ▶ Live Transaction monitoring - not that big for some reason
 - ▶ Graph Databases seems to be doing better here.



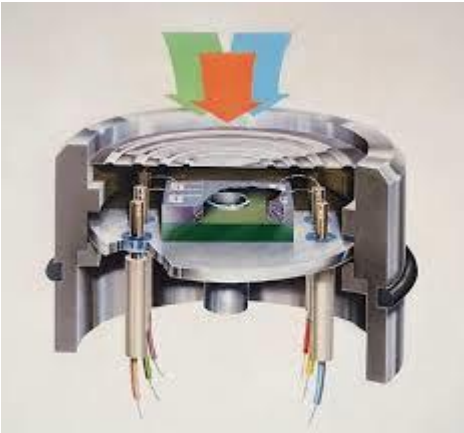
ii) Future Opportunities

- ▶ Sensors
- ▶ Self-Contained Clusters
- ▶ Combination ?



Sensors

- ▶ These days, sensors can be installed everywhere to monitor all aspects of life / business
 - ▶ Temperature Sensors
 - ▶ Pressure Sensors
 - ▶ Gas Sensors
 - ▶ Smoke Sensors
- ▶ A better understanding of day to day happenings can save money and lives.

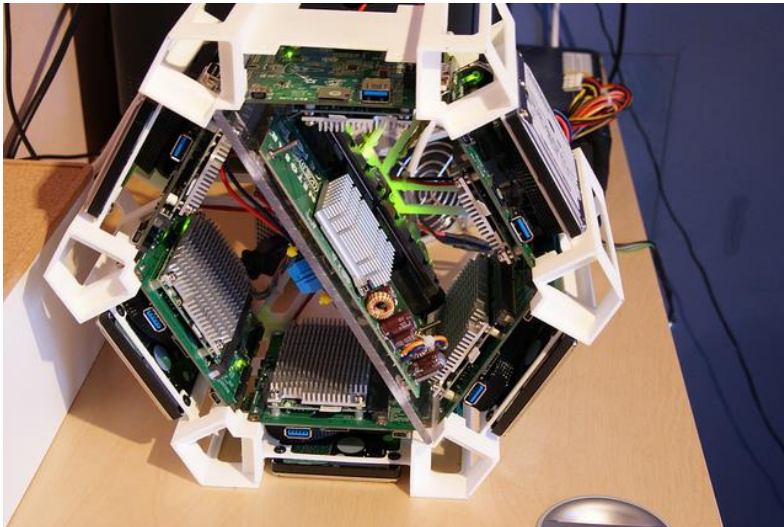




Self-Contained Clusters

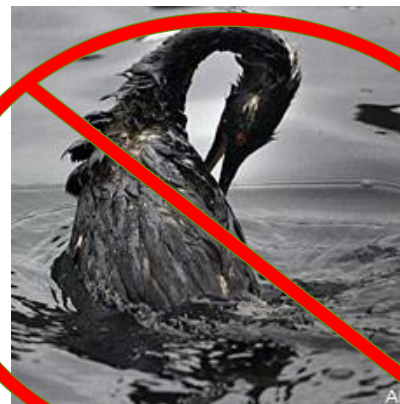
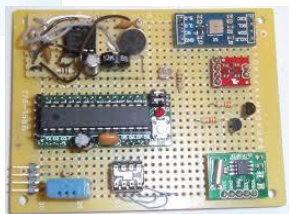
BIGBOARDS

- ▶ Met these guys at the Hadoop Summit in Amsterdam 2014 (<http://bigboards.io/>)
- ▶ 5 data processing nodes
20 CPU cores and 5TB of raw storage
1GB ethernet to interlink everything
1 management console with technology and data library



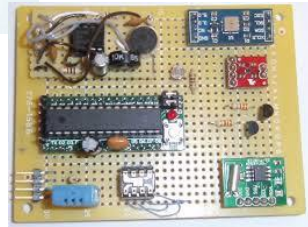


Self-Contained Clusters + Sensors



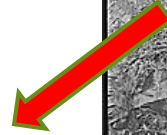
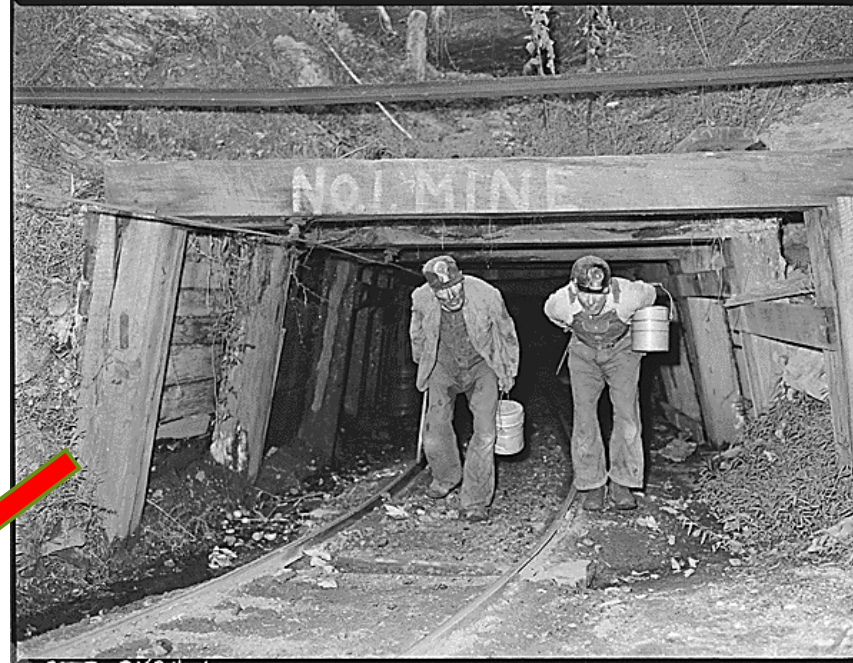
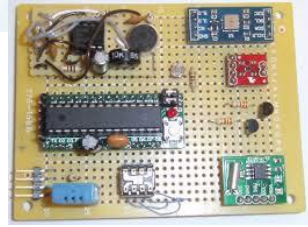


Self-Contained Clusters + Sensors





Self-Contained Clusters + Sensors





E) Summary

- ▶ Big data does not replace random read and reporting capabilities of SQL Server.
- ▶ Big Data is not close to replacing our
 - ▶ trusted
 - ▶ high volume
 - ▶ transaction safe
 - ▶ OLTP frameworks we built.
- ▶ Big data opens up opportunities for storing and processing data at a larger scale than we could never have dreamed of before.



F) Conclusion

- ▶ THE FUTURE is not going to be won by one OR the other ...



...but by a combination of BOTH!



Tools To Play With

- ▶ Hortonworks Sandbox
 - ▶ <http://hortonworks.com/products/ Hortonworks-sandbox/>
- ▶ Cloudera Sandbox
 - ▶ <http://www.cloudera.com/content/support/en/downloads.html>
- ▶ MAPR Sandbox
 - ▶ <http://www.mapr.com/products/mapr-sandbox-hadoop>
- ▶ Cascading & Lingual Cluster (Get Vagrant & Virtual Box)
 - ▶ <https://github.com/Cascading/vagrant-cascading-hadoop-cluster>
- ▶ Vagrant
 - ▶ <http://www.vagrantup.com/>
- ▶ Virtual Box
 - ▶ <https://www.virtualbox.org/>
- ▶ Talend
 - ▶ <http://www.talend.com/download>
- ▶ VMWARE Workstation 10
 - ▶ https://my.vmware.com/web/vmware/info/slug/desktop_end_user_computing/vmware_workstation/10_0
- ▶ HDInsight Emulator
 - ▶ <http://azure.microsoft.com/en-us/documentation/articles/hdinsight-get-started-emulator/#install>



Appendix : References

- ▶ **1) Hadoop : Distributed Data Processing [Amr Awadallah]
 - ▶ <http://www.slideshare.net/cloudera/hadoop-distributed-data-processing>
- ▶ **2) Hadoop [K Subrahmanyam]
 - ▶ <http://www.authorstream.com/Presentation/aSGuest129127-1356869-techseminar-on-hadoop-ppt/>
- ▶ **3) An Introduction to Apache Hadoop MapReduce [Mike Frampton]
 - ▶ http://www.powershow.com/view/3fdd1b-MGRkZ/An_Introduction_to_Apache_Hadoop_MapReduce_powerpoint_ppt_presentation
- ▶ **4) Mahout Explained in 5 Minutes or Less [Josh Gertzen]
 - ▶ <http://blog.credera.com/technology-insights/java/mahout-explained-5-minutes-less/>
- ▶ **5) What is Apache Tez? [Roopesh Shenoy]
 - ▶ <http://www.infoq.com/articles/apache-tez-saha-murthy>

