Scalable ETL with Talend and Hadoop

Talend, Global Leader in Open Source Integration Solutions

Cédric Carbone – Talend CTO
Twitter : @carbone
ccarbone@talend.com
Why speaking about ETL with Hadoop

Hadoop is complex
BI consultants don’t have the skill to manipulate Hadoop
→ Biggest issue in Hadoop project is to find skilled Hadoop Engineers

→ ETL tool like Talend can help to democratize Hadoop
Trying to get from this...
Talend generates code that is executed within map reduce. This open approach removes the limitation of a proprietary “engine” to provide a truly unique and powerful set of tools for big data.
Big Data Management

Big Data Production

- RDBMS
- Analytical DB
- NoSQL DB
- ERP/CRM
- SaaS
- Social Media
- Web Analytics
- Log Files
- RFID
- Call Data Records
- Sensors
- Machine-Generated

Big Data Management

- Big Data Integration
- Big Data Quality

Big Data Consumption

- Mining
- Analytics
- Search
- Enrichment

Turn Big Data into actionable information
Two methods for inserting data quality into a big data job

1. Pipelining: as part of the load process

2. Load the cluster than implement and execute a data quality map reduce job
E-T-L
Extract – Transform - Load
E-DQ-L

Extract – Improve/Cleanse - Load
Pipelining: data quality with big data

- Use traditional data quality tools
- Once and done
Big data alternative: Load and improve within the cluster

- Load first, improve later
- Complex matching cannot be done outside
One key DQ rules: Match

→ Find duplicates within Hadoop
→ Today’s matching algorithms are processor-intensive
→ Tomorrow’s matching algorithms could be more precise, more intensive
What is Hadoop?
What’s hadoop

- The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

- Java framework for storage and running data transformation on large cluster of commodity hardware.

- Licensed under the Apache v2 license.

- Created from Google's MapReduce, BigTable and Google File System (GFS) papers.
Hadoop ecosystem
Talend for Big Data: Hadoop Story

4.0: [April 2010] Put or get data into Hadoop through HDFS connectors

4.1: [Oct 2010] Hadoop Query (Hive) Bulk load & fast export to Hadoop (Sqoop)

4.2: [May 2011] Transformation (Pig)

5.0: [Nov 2011] Hbase NoSQL. Extend our tPig

5.1: [May 2012] Metadata (Hcatalog) Deployment & Scheduling (Oozie) Embeded into HDP


5.3: [June 2013] Visual Pig mapping Machine Learning (Mahout) Native MapReduce Code Gen
Democratizing Integration with Data Integration tools for Big Data
WordCount

WordCount Example

WordCount example reads text files and counts how often words occur. The input is text files and the output is text files, each line of which contains a word count:

```
• Comes with Hadoop
• “First” demo that everyone tries!
```

How-to in Talend Big Data

```
• Simple read, count, load results
• No coding, just drag-n-drop
• Runs remotely
```
Map Reduce

Input Files

- Data Node 1
  - Apple
  - Orange
  - Mango
  - Grapes
  - Plum

- Data Node 2
  - Apple
  - Plum
  - Mango
  - Apple

Final Output

- Apple, 4
- Grapes, 1
- Mango, 2
- Orange, 2
- Plum, 3
Map Reduce

Input Files

Apple Orange Mango
Orange Grapes Plum

Apple Plum Mango
Apple Apple Plum

Each line passed to individual mapper instances

Map Key Value Splitting

Apple,1
Orange,1
Mango,1

Orange,1
Grapes,1
Plum,1

Apple,1
Plum,1
Mango,1

Reduce Key Value Pairs

Apple,4
Grapes,1
Mango,2
Grapes,1
Mango,1
Orange,2
Orange,1
Plum,3
Plum,1
Plum,1
Plum,1
Map Reduce

Input Files

Apple Orange Mango
Orange Grapes Plum

Apple Plum Mango
Apple Apple Plum

Each line passed to individual mapper instances

Map Key Value Splitting

Apple, 1
Orange, 1
Mango, 1

Orange, 1
Grapes, 1
Plum, 1

Apple, 1
Plum, 1
Mango, 1

Map Key Value Pairs

Apple, 4

Orange, 2

Plum, 1
Plum, 1
Plum, 1

Reduce Key Value Pairs

Grapes, 1

Mango, 2

Orange, 2

Plum, 3
```java
package org.myorg;
import java.io.IOException;
import java.util.*;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class WordCount {

    public static class Map extends Mapper<LongWritable, Text, Text, IntWritable> {
        
        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

        public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException {
            String line = value.toString();
            StringTokenizer tokenizer = new StringTokenizer(line);
            while (tokenizer.hasMoreTokens()) {
                word.set(tokenizer.nextToken());
                context.write(word, one);
            }
        }
    }

    public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {

        public void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOException, InterruptedException {
            int sum = 0;
            for (IntWritable val : values) {
                sum += val.get();
            }
            context.write(key, new IntWritable(sum));
        }
    }

    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();
        Job job = new Job(conf, "wordcount");
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);
        job.setMapperClass(Map.class);
        job.setReducerClass(Reduce.class);
        job.setInputFormatClass(TextInputFormat.class);
        job.setOutputFormatClass(TextOutputFormat.class);
        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));
        job.waitForCompletion(true);
    }
```

WordCount: How to with a Graphical ETL

The wordcount example reads text files and counts how often words occur.
Thank You!
Let us show you...
Talend Open Studio
Generate Pure Map Reduce
FOREACH tPigMap_1_out1_RESULT GENERATE $4 AS Revenu, $6 AS Label
HiveQL generation

```
SELECT
customer.id, customer.lastname, customer.revenue, states.StatesLabel
FROM
customer INNER JOIN states ON (states.PostalCode = customer.postalCode)
```
HDFS Management and Sqoop

HDFS
- tHDFSCheckpoint
- tHDFSCheckpointConfig
- tHDFSCheckpointDelete
- tHDFSCheckpointSize
- tHDFSCheckpointStatus
- tHDFSCheckpointStart
- tHDFSCheckpointEnd
- tHDFSCheckpointResume
- tHDFSCheckpointComplete
- tHDFSCheckpointRollback
- tHDFSCheckpointVerify
- tHDFSCheckpointCompact
- tHDFSCheckpointMove
- tHDFSCheckpointCopy
- tHDFSCheckpointDelete
- tHDFSCheckpointExist
- tHDFSCheckpointGet
- tHDFSCheckpointInput
- tHDFSCheckpointList
- tHDFSCheckpointOutput
- tHDFSCheckpointProperties
- tHDFSPut
- tHDFSPutProperties
- tHDFSRename
- tHDFSRowCount

Sqoop
- tSqoopExport
- tSqoopImport
- tSqoopImportAllTables
Apache Mahout

- Big Data can also be a blob of data to an organization
- Apache Mahout provides algorithms to understanding data – data mining
- “You don’t know what you don’t know.” and mahout will tell you.
Centralize Metadata repository for Hadoop Cluster, HDFS, Hive...

- Versioning
- Impact Analysis and Data Lineage

→ HCatalog accros

- HDFS
- Hive
- Pig
Thank You!
Choose your Hadoop distro

- Widely adopted
- Management tooling is not OSS
- Fully OpenSource
- Strong Developer ecosystem
- More proprietary
- GTM partner with AWS
- A lot of more are coming
Choose your Hadoop distro

Provide tooling:
→ For installation
→ For server monitoring

But
→ No GUI for parsing, transforming, easily loading. No data management
Parse and Standardize

→ Big Data is not always structured
→ Correct big data so that data conforms to the same rules
Profiling & Monitor DQ
DATA

Challenge: Information explosion increases complexity of integration and requires governance to maintain data quality

Requirement: Information processing must scale
Implications for Integration

**APPLICATION**

**Challenge:** Brittle, point-to-point connections cannot adapt to evolving business requirements, new channels, and quickly changing topologies

**Requirement:** Application architecture must scale
Implications for Integration

PROCESS

Challenge: Competitive market forces drive frequent process changes and increased process complexity

Requirement: Business processes must scale
Implications for Integration

**Challenge:** Interdependencies across data, applications and processes require more resources and budget

**Requirement:** Resources and skillsets must scale
Integration at Any Scale

True scalability for
- Any integration challenge
- Any data volume
- Any project size

Enables integration convergence
Technology that Scales

STANDARDS-BASED
Easy to learn, flexible to adopt, reduces vendor lock-in

CODE GENERATOR
No black-box engine means faster maintenance and deployment with improved quality

The “engine” for Big Data is “Hadoop”, making it uniquely run at infinite scale.
Technology Continuum...

ELT (SQL CodeGen)
- Terradata
- Netezza
- Vertica

ETL
- Java Code
- Partitionning
- Parallelisation

NoSQL
- MongoDB
- Neo4J
- Cassandra
- Hbase
- Amazon Redshift

Google Big Query
- Visual Wizard
- Hive/Pig/MR CodeGen
- HDFS, Sqoop, Oozie…

Hadoop
Talend Overview

At a glance

- Founded in 2005
- Offers highly scalable integration solutions addressing Data Integration, Data Quality, MDM, ESB and BPM
- Provides:
  - Subscriptions including 24/7 support and indemnification;
  - Worldwide training and services
- Recognized as the open source leader in each of its market categories

Talend today

- 400 employees in 7 countries with dual HQ in Los Altos, CA and Paris, France
- Over 4,000 paying customers across different industry verticals and company sizes
- Backed by Silver Lake Sumeru, Balderton Capital and Idinvest Partners

High growth through a proven model

- Brand Awareness: 20 million Downloads
- Market Momentum: +50 New Customers / Month
- Adoption: 1,000,000 Users
- Monetization: 4,000 Customers
Talend’s Unique Integration Solution

- Reduce costs
- Eliminate risk
- Reuse skills
- Economies of scale
- Incremental adoption

Recognized as the open source leader in each of its market category by all industry analysts
Solutions that Scale

UNIFIED PLATFORM
A shared foundation and toolset increases resource reuse

CONVERGED INTEGRATION
Use for any data, application and process project
The 6 Dimensions of **BIG** Data

**Primary challenges**
- Volume
- Velocity
- Variety

**And also**
- Complexity
- Validation
- Lineage
"Big data" is information of extreme size, diversity, complexity and need for rapid processing.

Ted Friedman - Information Infrastructure and Big Data Projects Key Initiative Overview - July 2011

Gartner®

3,500 tweets per second (June 2011)

1,000,000 transactions per day at Walmart

200 billion intelligent devices

275 exabytes of data flowing over the Internet each day

2015

2020
What is Big Data?
How to define Big data is....

Hans Rosling – uses big data to analyze world health trends

Key Takeaway #1

**volume, variety, velocity**
Traditional Data Flows

- Scheduled—daily or weekly, sometimes more frequently.
- Volumes rarely exceed terabytes
The new world of big data

CRM

ERP

Finance

Big Data

Social Networking
The new world of big data

CRM
ERP
Finance

Social Networking
Mobile Devices

Big Data
The new world of big data
Key Takeaway #2

Forces us to think differently
Data driven business

Information provides value to the business

If you can't rely on your information then the result can be missed opportunities, or higher costs.

Information provides value to the business

If you can't rely on your information then the result can be missed opportunities, or higher costs.

How is big data integration being used?

Use Cases
• Recommendation Engine
• Sentiment Analysis
• Risk Modeling
• Fraud Detection
• Behavior Analysis
• Marketing Campaign Analysis
• Customer Churn Analysis
• Social Graph Analysis
• Customer Experience Analytics
• Network Monitoring

BUT: to what level is DQ required for your use case?
Key Takeaway #3

Define your use case