https://bit.ly/2zJJ3Fh

AWS Israel Community

Sponsors

datree.io

aws

Big Data on AWS

MEET THE TEAM

Shimon Tolts

Arthur Schmunk

Tal Hibner

Niv Yungelson

Eitan Sela

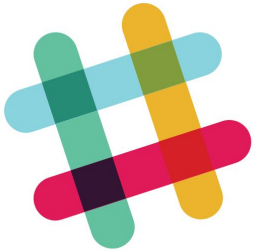
Doron Rogov

Boaz Ziniman

AWS Israel Community

- Founded - Feb **2013**
- **79** meetups with **~6000** Members
- Monthly meetups
- No Marketing, No bullshit
- Monthly Highlight (News)
- All AWS: AI, BigData, Serverless, Containers, etc

Join the Community!



https://bit.ly/2zJJ3Fh

https://www.meetup.com/AWS-IL/

https://www.meetup.com/AWS-IL/

aws.org.il

Thank you Eitan Sela!

AWS User Group Israel

Coming meetups

Coming meetups

- Big Data on AWS - 2018-07-16 18:00 @ AWS Offices.

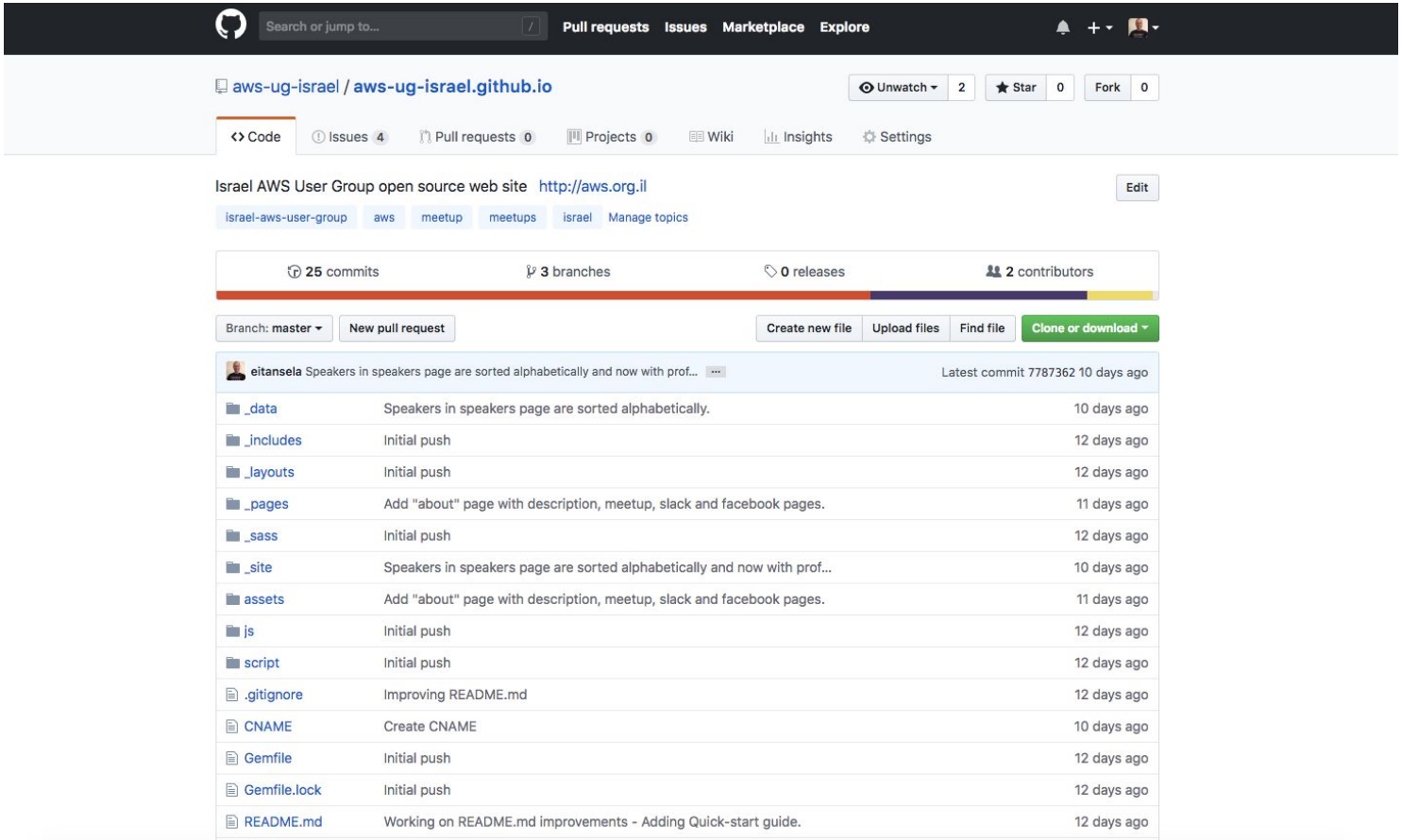
Past meetups

2018

- Guest Meetup: AWS Cloud Financial Governance Practice
- Kombinot on AWS - Running Beyond Cost Effective

aws.org.il - components

It is open source - you can contribute



AWS News

- AWS Lambda Adds Amazon Simple Queue Service to Supported Event Sources.
- Amazon SageMaker Now Supports k-Nearest-Neighbor and Object Detection Algorithms
- We Appear In The AWS User Groups Listings (Under Middle East & Africa ¯_(ツ)_/¯).

Upcoming Meetup

Production Engineering on AWS. Debugging, Blue/Green, Canary, DB consistency - August 21, 2018

Big Data on AWS

- How to squeeze ETLs performance from hours to minutes. Eitan Sela, WeissBeerger
- Squeegee - Open Source, "Serverless" AWS Cost and Usage Analysis at Scale - by Elliott Spira
- Advanced GPU operations on AWS - by Gil Bahat

How to squeeze ETLs performance from hours to minutes

Eitan Sela - Svstem Architect

WeissBeerger

A part of the ABInBev Family

eitan.sela@weissbeerger.com

$ whoami

- "Hands-On" system Architect with more than 17 years of experience with billing, banking, information security (DLP) and Cloud IoT/Big Data applications.
- Big Data specialist – Hadoop, Spark, Hive and EMR on AWS.
- Work with vast AWS services, and with serverless projects especially.
- Java development, scalability performance and stabilization expert.
- Alexa skills developer.
- Love to share my experience in lectures and meetups.

What to expect from this session

- WeissBeerger use case – Aggregating raw orders and IoT data.
- Moving to Redshift and traditional ETLs using PDI.
- Introduction to Hadoop/Hive/Spark and big data.
- Amazon EMR basics.
- Squeezing ETLs from hours to minutes.
- Our new Slack Chabot for EMR, using Amazon Lex!

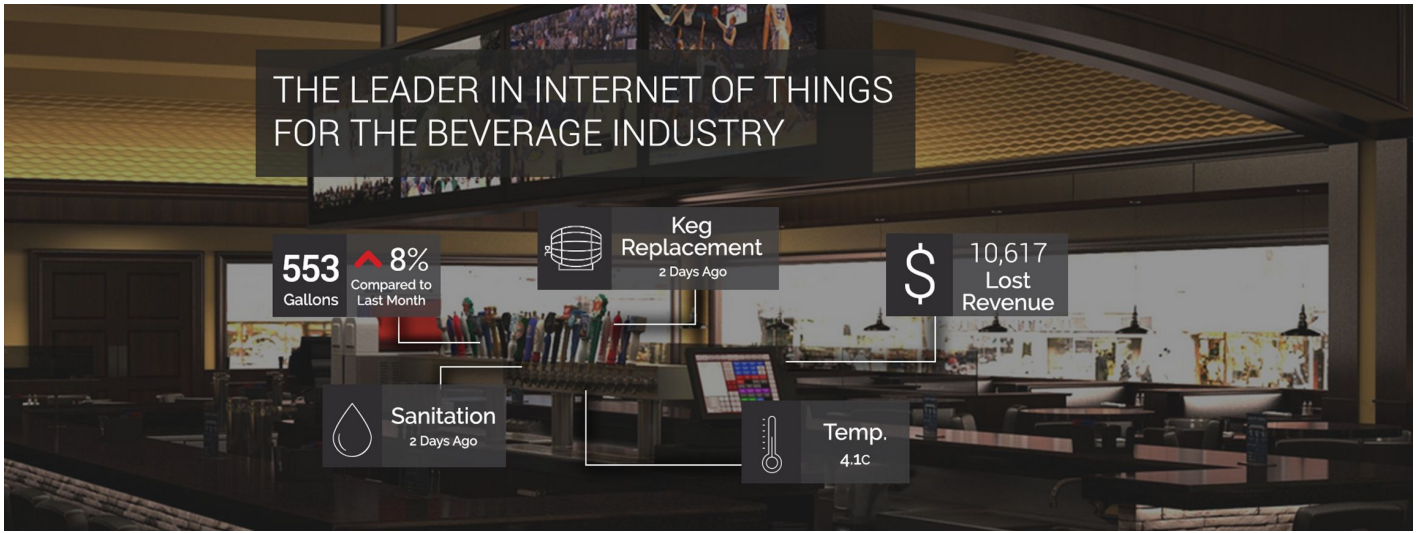
WeissBeerger use case – Aggregating raw orders and IoT data

WeissBeerger

A part of the AB InBev Family

Solution

- WeissBeerger bridges the gap between breweries, bars and customers.



Benefits for the brewery

- Consumption Analytics.
- Dynamic Promotions.
- Beer Quality.
- **Value creation.**
- **Beer penetration**.

Benefits for the bar

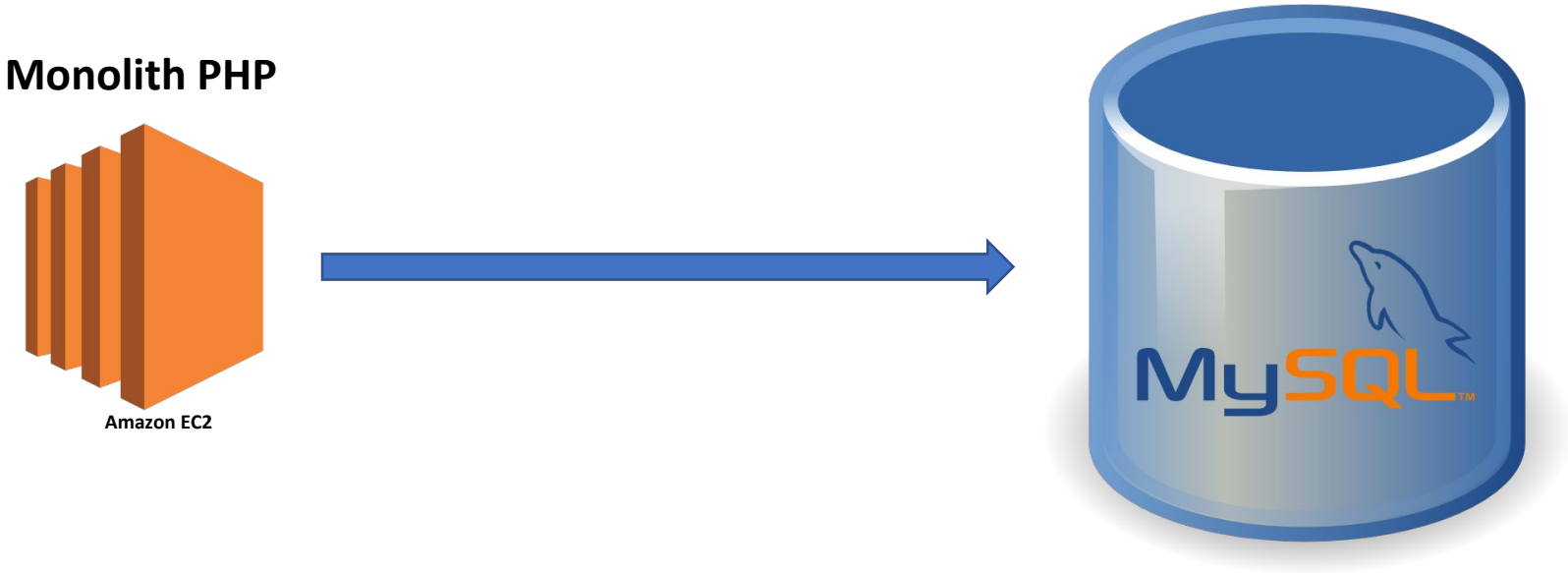
- Real Time Consumption Tracking.
- Waste Reduction.
- **Sales Growth**.

How does it work?

- IoT (Pouring) – Beverage Analytics Hub.
- Point of sales – POS Vendors, via REST API, S3, DB, etc.

Aggregating raw point of sales orders and IoT data

Web Dashboard and Mobile App – Architecture one year ago



Problem – performance issues with Web dashboard and Mobile App

```
SELECT
    t1.bar_id,
    t2.title,
    t2.order_id,
    t3.category_id,
    sum(t3.price) as total_price,
    t7.number_of_brands
FROM table_1 t1
INNER JOIN table_2 t2 ON t1.order_id = t2.id
INNER JOIN table_3 t3 ON t2.bar_id = t3.bar_id
INNER JOIN table_4 t4 ON t3.bar_id = t4.id
INNER JOIN table_5 t5 ON t4.id = t5.product_id
LEFT JOIN table_6 t6 ON t6.product_id = t5.id
LEFT JOIN table_7 t7 ON t7.product_id = t6.id
LEFT JOIN table_8 t8 ON t8.id = t7.brand_id
LEFT JOIN table_9 t9 ON t9.id = t8.beer_type_id
WHERE
  <condition 1>
AND
  <condition 2>
AND
  <condition 3>
AND
 <condition 4>
AND
 <condition 5>
GROUP BY t1.id, t1.bar_id, t2.title, t1.order_id, t3.category_id, t3.price, t7.number_of_brands
ORDER BY t1.bar_id, t1.order_id
```

Moving to Redshift and traditional ETLs using PDI

amazon
REDSHIFT

pentaho®
A Hitachi Group Company

Amazon Redshift - Fast, simple, cost-effective data warehousing

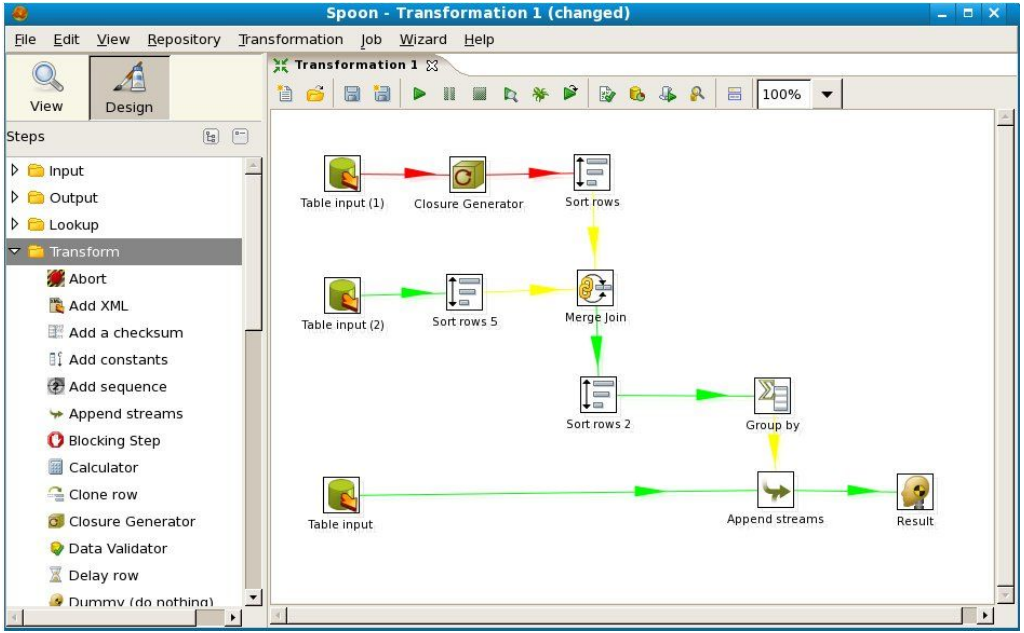
- Fully managed Platform as a Service (PaaS) solution by Amazon in the cloud.
- A column-oriented database stores data tables by column rather than by row.
- The database can more precisely access the data it needs to answer a query rather than scanning and discarding unwanted data in rows – increasing query performance for these use cases
- Postgres based, with alterations (e.g. column based, doesn't respect constraints)
- Deployed as cluster of 1 or more servers that execute queries in parallel to improve performance
- Not great at Insert - use COPY from S3

amazon
REDSHIFT

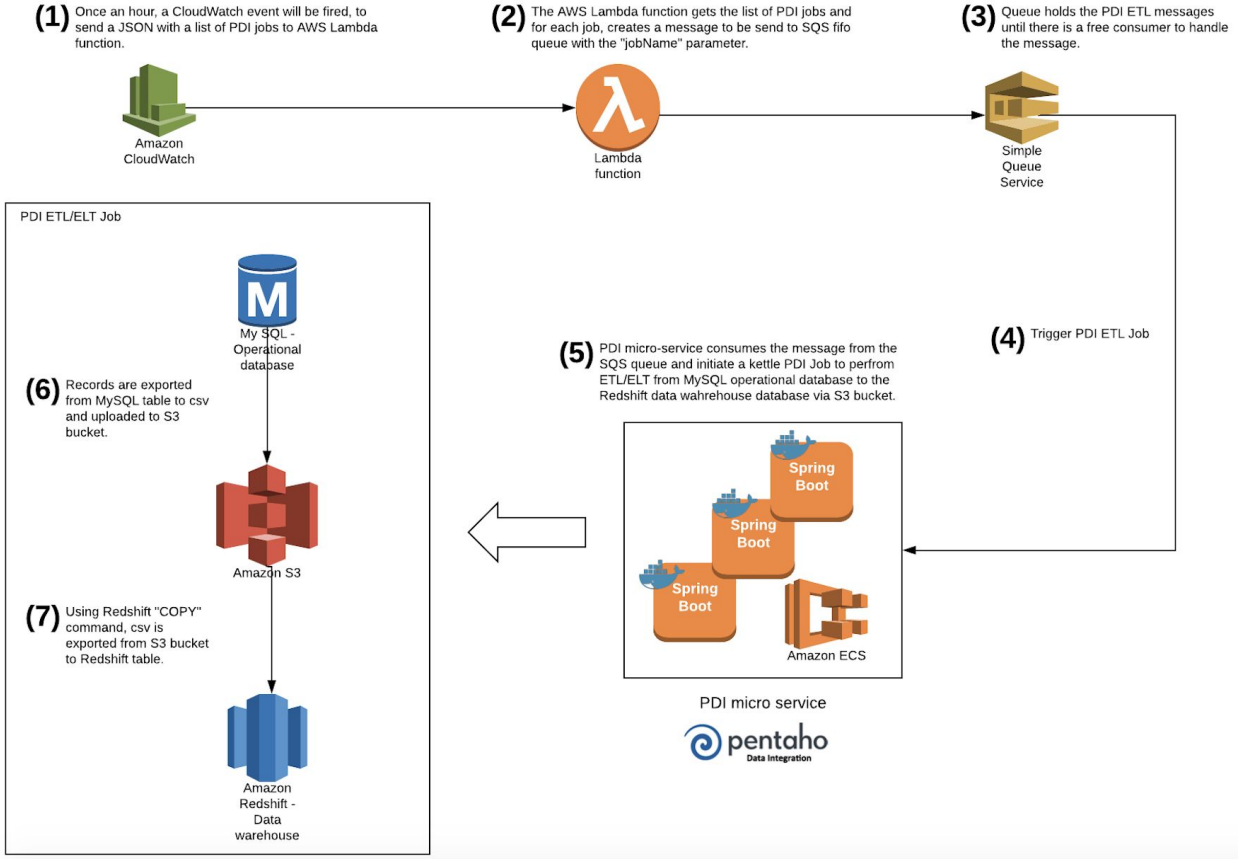
Pentaho Data Integration (PDI)

pentaho
A Hitachi Group Company

- Enable users to ingest, blend, cleanse and prepare diverse data from any source.
- With visual tools to eliminate coding and complexity, Pentaho puts the best quality data at the fingertips of IT and the business.
- Graphical ETL designer simplifies the creation of data pipelines.
- Rich library of prebuilt components help to access, prepare and blend data.
- Big Data Integration With Zero Coding Required.



Implementing ETL's using Pentaho Data Integration (PDI)

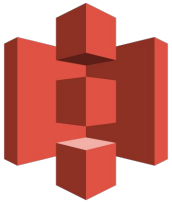


Implementing ETL's using Pentaho Data Integration (PDI) - Problems

- As the amount of data increased, the ETL's started to take more and more time, **up to 50 minutes** for the most complex ETL.

- Complex queries with joins on huge SQL tables caused heavy load on MySQL database and slave lags.

Introduction to Hadoop/Hive/Spark and big data



Amazon
S3

Dimensions of Big Data

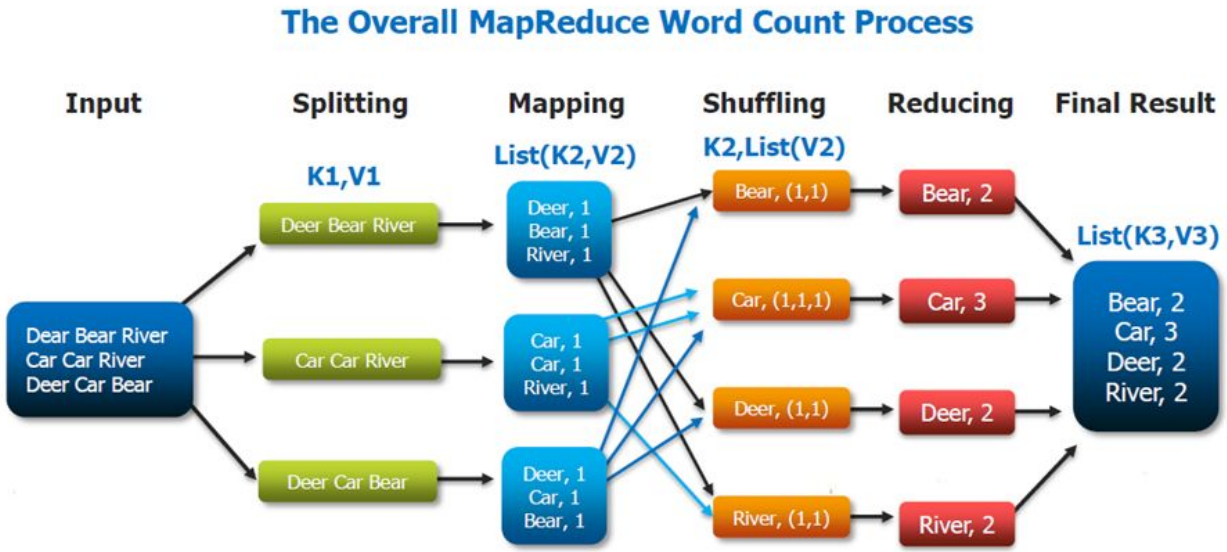
- **Volume** – The amount of data generated in the world is increasing in exponential rate.

- **Variety** - There was a time when only structured data was meant to be processed. Today we analyze every sort of data – XML, JSON, CSV, S3, Avro, Parquet.

- **Velocity** – Data is not only increasing in size, but also in the rate which it is arriving. Ability to analyze data in near time to generate real value.

What exactly is Map Reduce?

- “MapReduce is a programming model and an associated implementation for processing and generating big data sets with a parallel, distributed algorithm on a cluster”



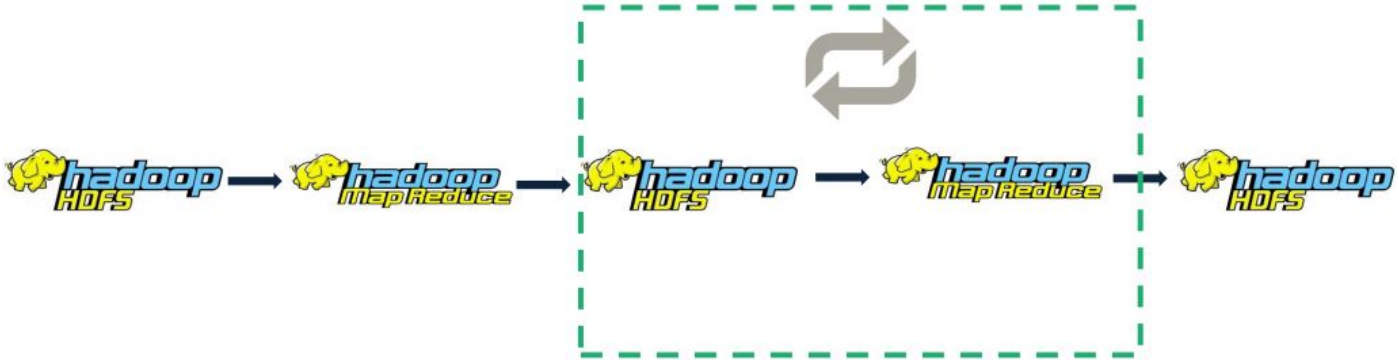
What Is Apache Hadoop

- A collection of open-source software utilities that facilitate using a network of many computers to solve problems involving massive amounts of data and computation.
- It provides a software framework for distributed storage and processing of big data using the MapReduce programming model.
- It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.

Hadoop Map Reduce and I/O overhead

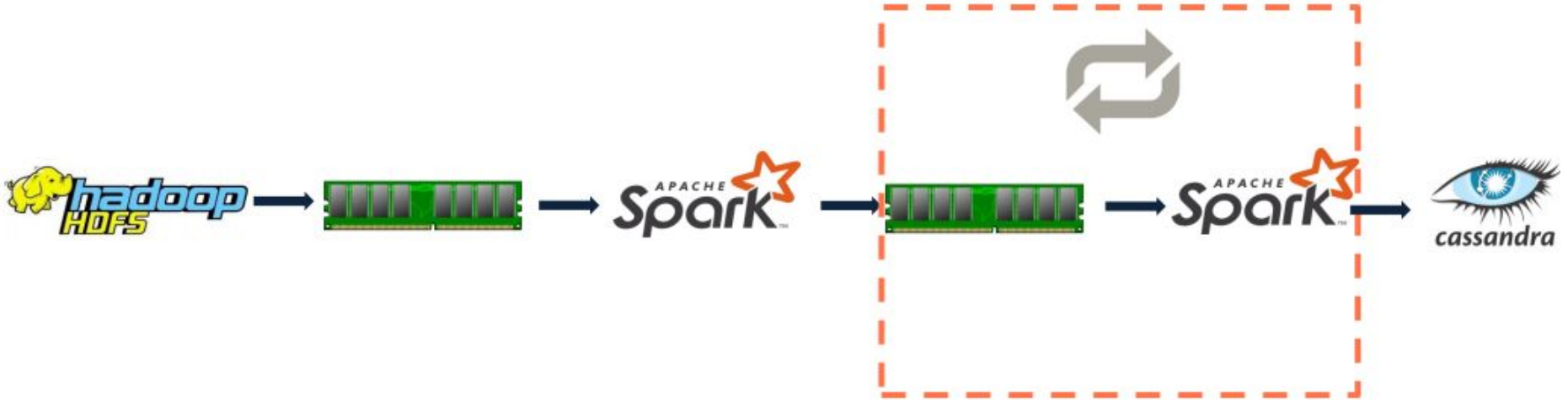
- Read from disk ("*hdfs://123.23.12.4344:9000/wc/words.txt*") -> process -> write back to disk ("*hdfs://123.23.12.4344:9000/wc/wordsCount.txt*")

- Iterative processing e.g. Machine learning Repeatedly accessing the



Apache Spark – In Memory computation

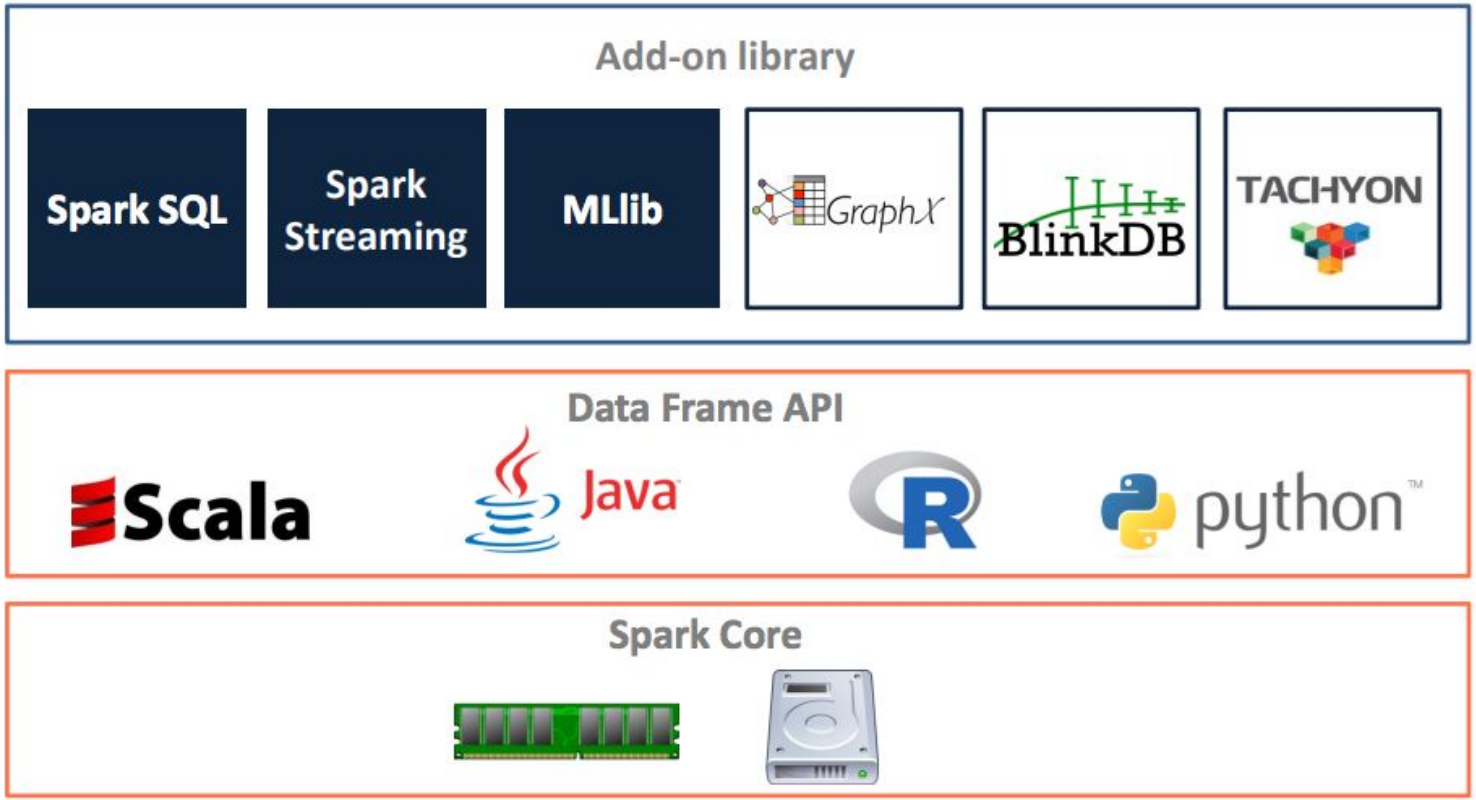
- In case of Apache Spark, it keeps the output of your previous stage in memory for that in next iteration, so it can be retrieved from memory which is quite faster than Disk IO.



Apache Spark - Data Source API

- Read and write with variety of formats

Apache Spark – Eco System



Word count – Hadoop Map Reduce

```
1.  package org.myorg;
2.
3.  import java.io.IOException;
4.  import java.util.*;
5.
6.  import org.apache.hadoop.fs.Path;
7.  import org.apache.hadoop.conf.*;
8.  import org.apache.hadoop.io.*;
9.  import org.apache.hadoop.mapred.*;
10. import org.apache.hadoop.util.*;
11.
12. public class WordCount {
13.
14.  public static class Map extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {
15.   private final static IntWritable one = new IntWritable(1);
16.   private Text word = new Text();
17.
18.   public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {
19.    String line = value.toString();
20.    StringTokenizer tokenizer = new StringTokenizer(line);
21.    while (tokenizer.hasMoreTokens()) {
22.     word.set(tokenizer.nextToken());
23.     output.collect(word, one);
24.    }
25.   }
26.  }
27.
28.  public static class Reduce extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {
29.   public void reduce(Text key, Iterator<IntWritable> values, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {
30.    int sum = 0;
31.    while (values.hasNext()) {
32.     sum += values.next().get();
33.    }
34.    output.collect(key, new IntWritable(sum));
35.   }
36.  }
37.
38.  public static void main(String[] args) throws Exception {
39.   JobConf conf = new JobConf(WordCount.class);
40.   conf.setJobName("wordcount");
41.
42.   conf.setOutputKeyClass(Text.class);
43.   conf.setOutputValueClass(IntWritable.class);
44.
45.   conf.setMapperClass(Map.class);
46.   conf.setCombinerClass(Reduce.class);
47.   conf.setReducerClass(Reduce.class);
48.
49.   conf.setInputFormat(TextInputFormat.class);
50.   conf.setOutputFormat(TextOutputFormat.class);
51.
52.   FileInputFormat.setInputPaths(conf, new Path(args[0]));
53.   FileOutputFormat.setOutputPath(conf, new Path(args[1]));
54.
55.   JobClient.runJob(conf);
56.  }
57. }
```

Word count - Spark Dataframe

```
import org.apache.spark.sql.Row

val dfsFilename = "/input/humpty.txt"
val readFileDF = spark.sparkContext.textFile(dfsFilename)
val wordsDF = readFileDF.flatMap(_.split(" ")).toDF
val wcounts3 = wordsDF.filter(r => (r(0) =="Humpty") || (r(0) == "Dumpty"))
                      .groupBy("Value")
                      .count()
wcounts3.collect.foreach(println)
```

What Is Hive

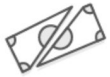
- Apache Hive is a data warehouse (initially developed by Facebook) software project built on top of Apache Hadoop for providing data summarization, query and analysis.
- Hive gives an SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop.
- It provides a SQL interface to query data stored in Hadoop distributed file system (HDFS) or Amazon S3 (an AWS implementation) through an HDFS-like abstraction layer called EMRFS (Elastic MapReduce File System).

Amazon EMR basics

Amazon EMR - Easily Run and Scale Apache Hadoop, Spark, HBase, Presto, Hive, and other Big Data Frameworks

Easy to Use

You can launch an Amazon EMR cluster in minutes. You don't need to worry about node provisioning, cluster setup, Hadoop configuration, or cluster tuning. Amazon EMR takes care of these tasks so you can focus on analysis.



Low Cost

Amazon EMR pricing is simple and predictable: You pay a per-second rate for every second used, with a one-minute minimum charge. You can launch a 10-node Hadoop cluster for as little as $0.15 per hour. Because Amazon EMR has native support for Amazon EC2 Spot and Reserved Instances, you can also save 50-80% on the cost of the underlying instances.

Elastic

With Amazon EMR, you can provision one, hundreds, or thousands of compute instances to process data at any scale. You can easily increase or decrease the number of instances manually or with Auto Scaling, and you only pay for what you use.

Reliable

You can spend less time tuning and monitoring your cluster. Amazon EMR has tuned Hadoop for the cloud; it also monitors your cluster —retrying failed tasks and automatically replacing poorly performing instances.

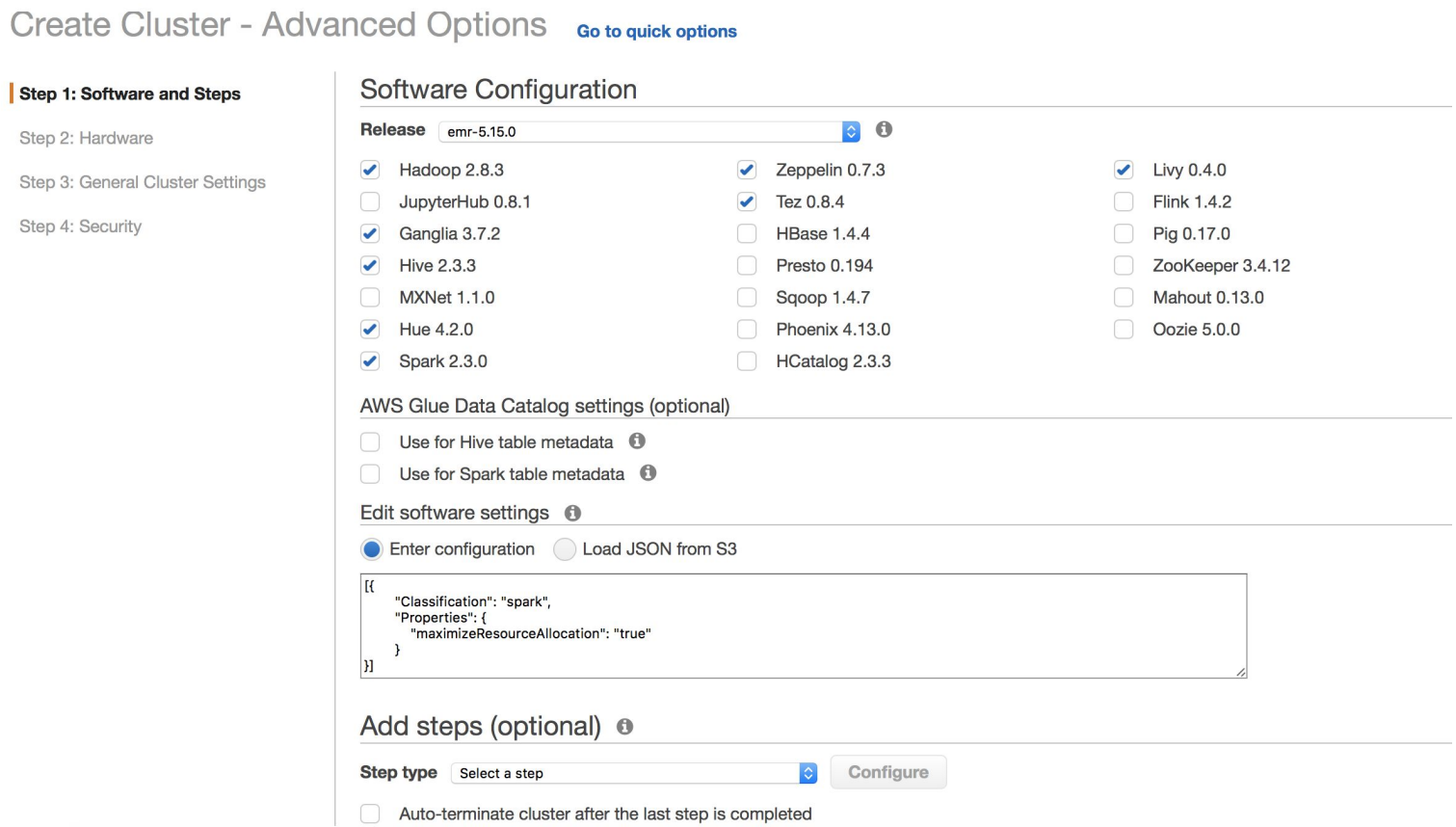
Secure

Amazon EMR automatically configures Amazon EC2 firewall settings that control network access to instances, and you can launch clusters in an Amazon Virtual Private Cloud (VPC), a logically isolated network you define. For objects stored in Amazon S3, you can use Amazon S3 server-side encryption or Amazon S3 client-side encryption with EMRFS, with AWS Key Management Service or customer-managed keys. You can also easily enable other encryption

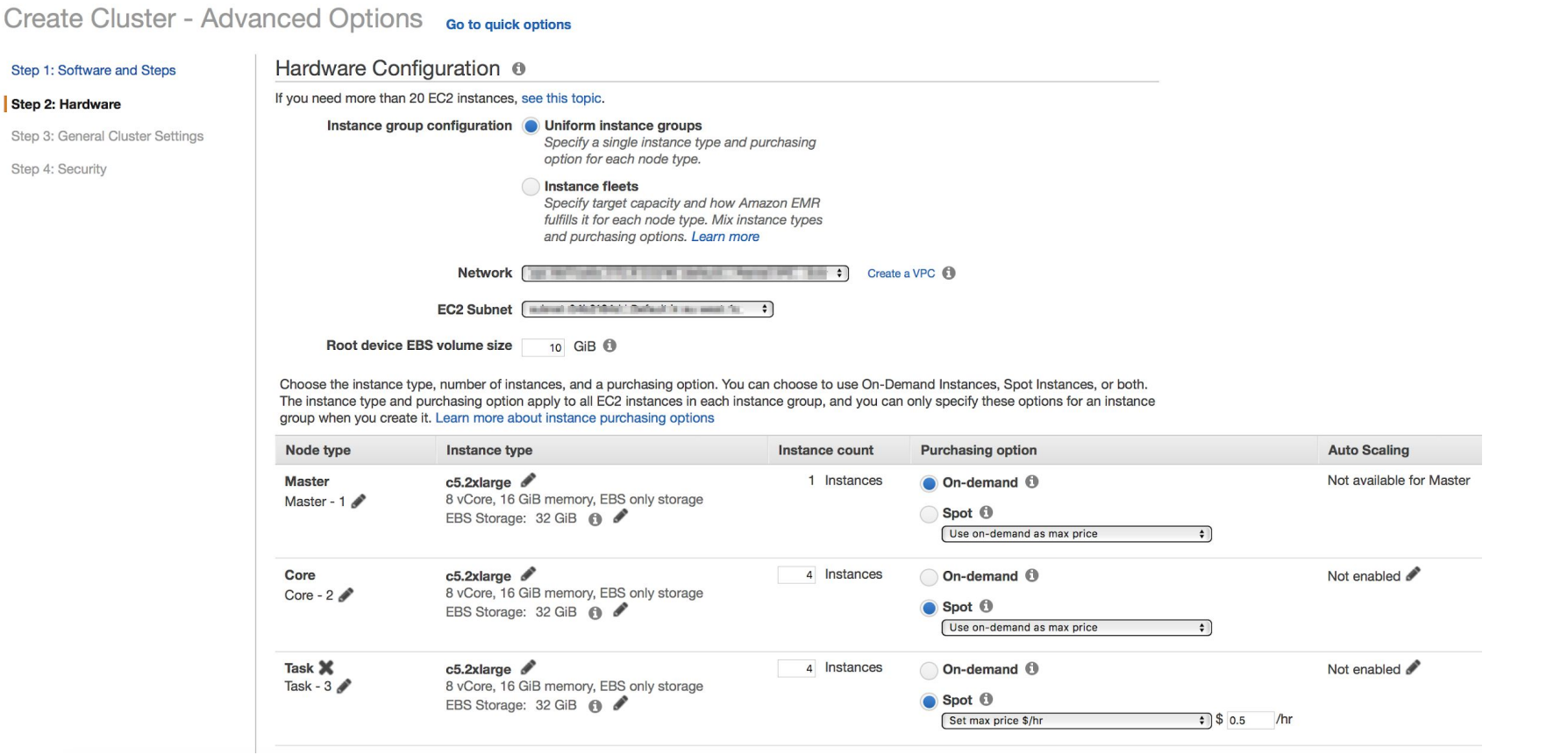
Flexible

You have complete control over your cluster. You have root access to every instance, you can easily install additional applications, and you can customize every cluster with bootstrap actions. You can also launch Amazon EMR clusters with custom Amazon Linux AMIs.

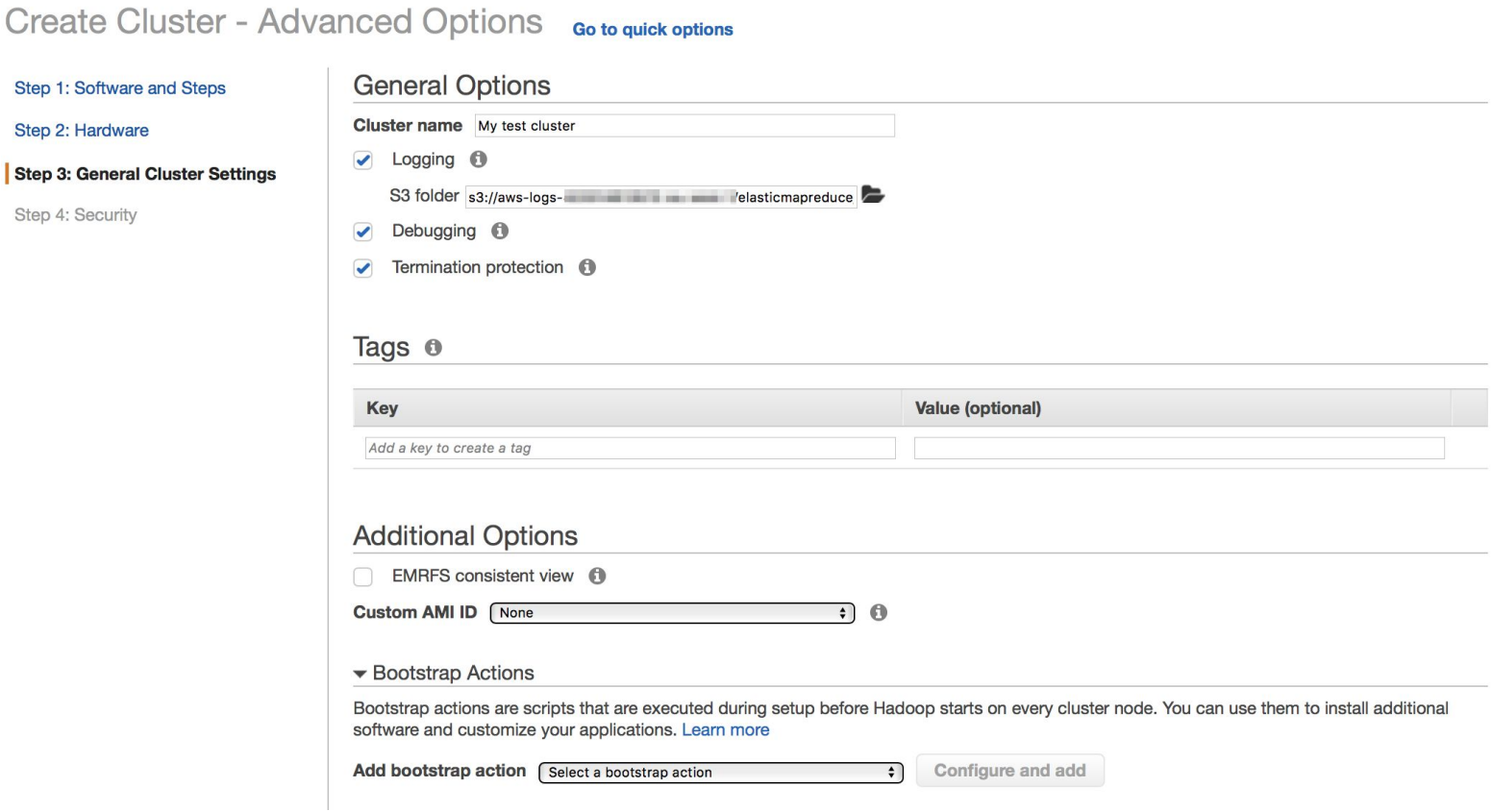
Amazon EMR – Create Cluster – Software and Steps



Amazon EMR – Create Cluster – Hardware



Amazon EMR – General Cluster Settings



Amazon EMR – Security

Create Cluster - Advanced Options

Go to quick options

- Step 1: Software and Steps
- Step 2: Hardware
- Step 3: General Cluster Settings
- **Step 4: Security**

Security Options

EC2 key pair [redacted]

Cluster visible to all IAM users in account

Permissions

Default Custom

Use default IAM roles. If roles are not present, they will be automatically created for you with managed policies for automatic policy updates.

EMR role EMR_DefaultRole

EC2 instance profile EMR_EC2_DefaultRole

Auto Scaling role EMR_AutoScaling_DefaultRole

Authentication and encryption

Security configuration [redacted]

EC2 security groups

An EC2 security group acts as a virtual firewall for your cluster nodes to control inbound and outbound traffic. There are two types of security groups you can configure, EMR managed security groups and additional security groups. EMR will automatically update the rules in the EMR managed security groups in order to launch a cluster. Learn more.

Type	EMR managed security groups EMR will automatically update the selected group	Additional security groups EMR will not modify the selected groups
Master	Create ElasticMapReduce-master	*No security groups selected*
Core & Task	Create ElasticMapReduce-slave	*No security groups selected*

Create a security group

Squeezing ETLs from hours to minutes

Implementing all ETL's with PySpark

```
from pyspark.sql import SparkSession
from datetime import datetime

def process_etl(from_date, to_date, spark):

    df = spark.sql("SELECT\
                    nd,\
                    bar_id,\
                    #... huge query 70 lines ...
                    FROM
                    #... multiple Hive on S3 tables ...
                    WHERE\
                        o.nd BETWEEN '"+from_date+"' AND '"+to_date+"'\
                    GROUP BY\
                        oi.id, o.nd, o.bar_id")


    df.write \
        .format("com.databricks.spark.redshift") \
        .option("url", redshift_url) \
        .option("dbtable", "public.[redacted]") \
        .option("tempdir", redshift_temp_dir) \
        .option("aws_iam_role", redshift_aws_iam_role) \
        .mode("append") \
        .save()
```

Using Zeppelin for ETL development with PySpark

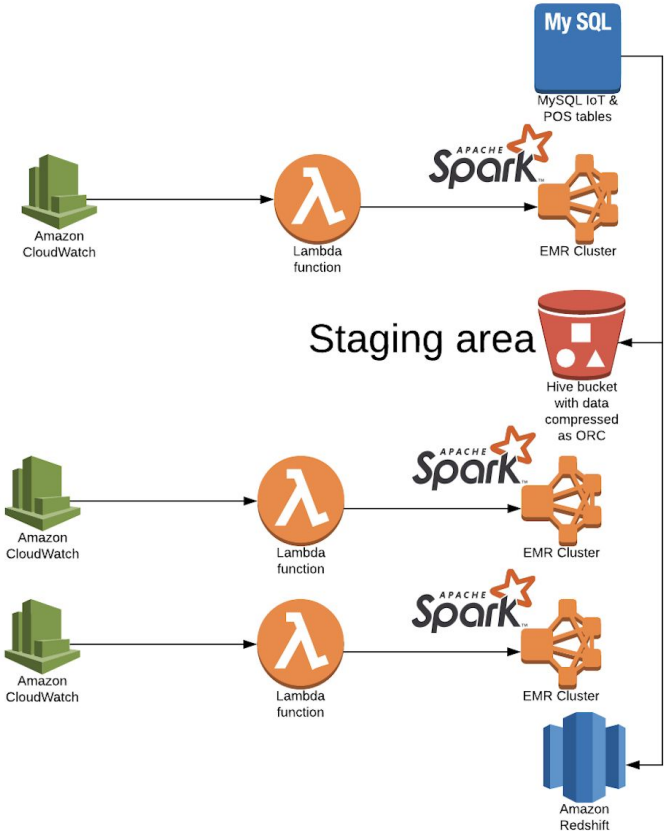
Zeppelin Notebook Job Search your Notes anonymous

bars_table_mysql_to_stg Head default

```
    spark.sql("INSERT OVERWRITE TABLE stg.bars SELECT id, [illegible] from mysql_bars")

    bars_redshift_df = bars_df.select("id", [illegible])

    print("%s - bars_table_mysql_to_stg - inserting to redshift" % datetime.now())

    bars_redshift_df.write \
        .format("com.databricks.spark.redshift") \
        .option("url", redshift_url) \
        .option("dbtable", "public.bars") \
        .option("tempdir", redshift_temp_dir) \
        .option("aws_iam_role", redshift_aws_iam_role) \
        .mode("overwrite") \
        .save()

    print("%s - bars_table_mysql_to_stg - ended" % datetime.now())

spark = SparkSession \
        .builder \
        .appName("Python Spark SQL to execute dimension tables mysql to stg job") \
        .enableHiveSupport() \
        .getOrCreate()
process_etl(spark)
```

```
2018-06-21 05:19:26.586611 - bars_table_mysql_to_stg - started
Records Count - bars_df: [illegible]
2018-06-21 05:19:39.403798 - bars_table_mysql_to_stg - inserting to redshift
```

New data pipeline using EMR PySpark jobs – ELT rather than ETL

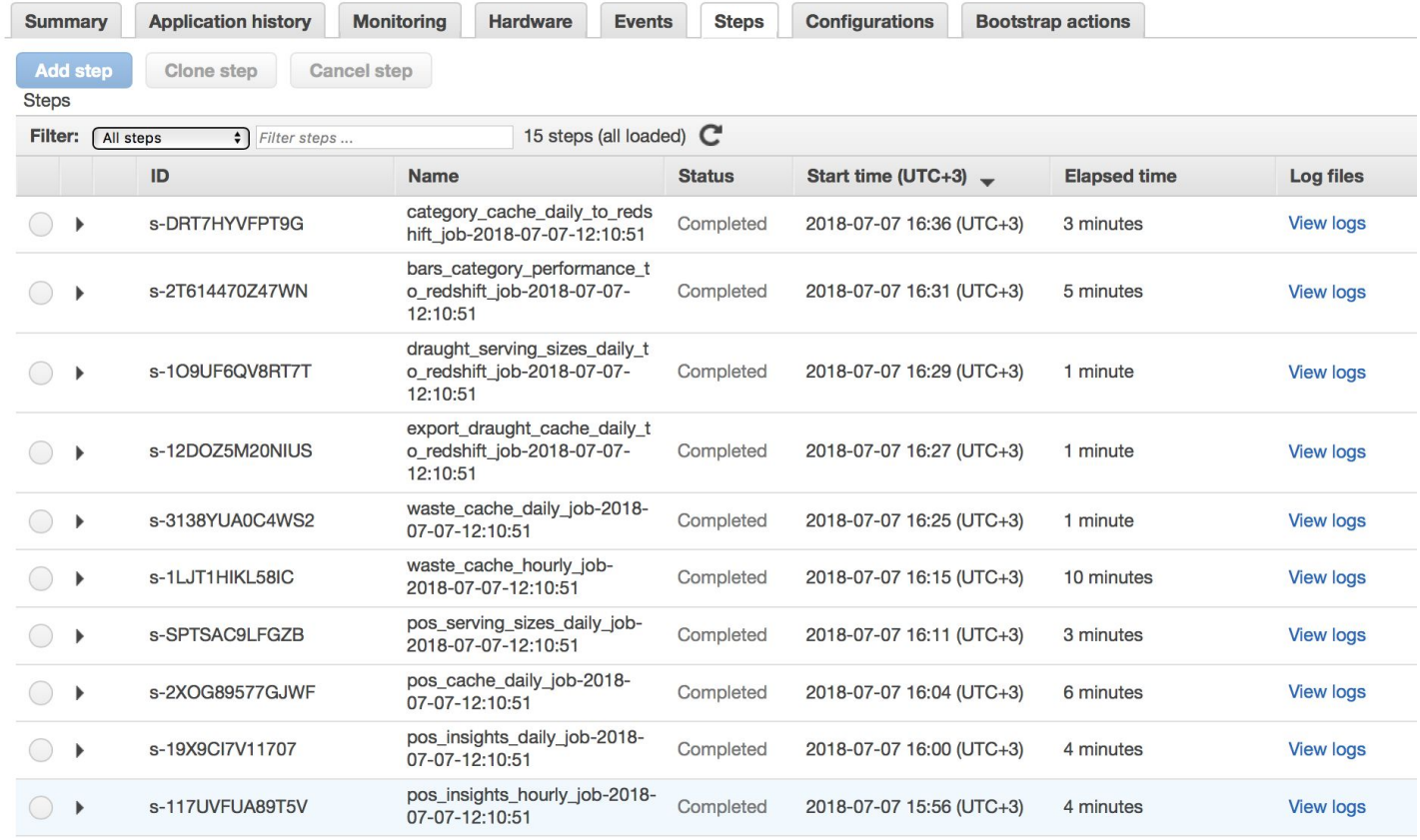


(1) On scheduled time - EMR transient cluster created with jobs to import tables from MySQL to Hive on S3, partitioned by nd (date).
The Hive tables are saved as ORC, ZLIB compressed.

(2) On scheduled time - EMR transient cluster created with jobs to perform IoT draught ETLs from Hive tables on S3 to Redhsift.

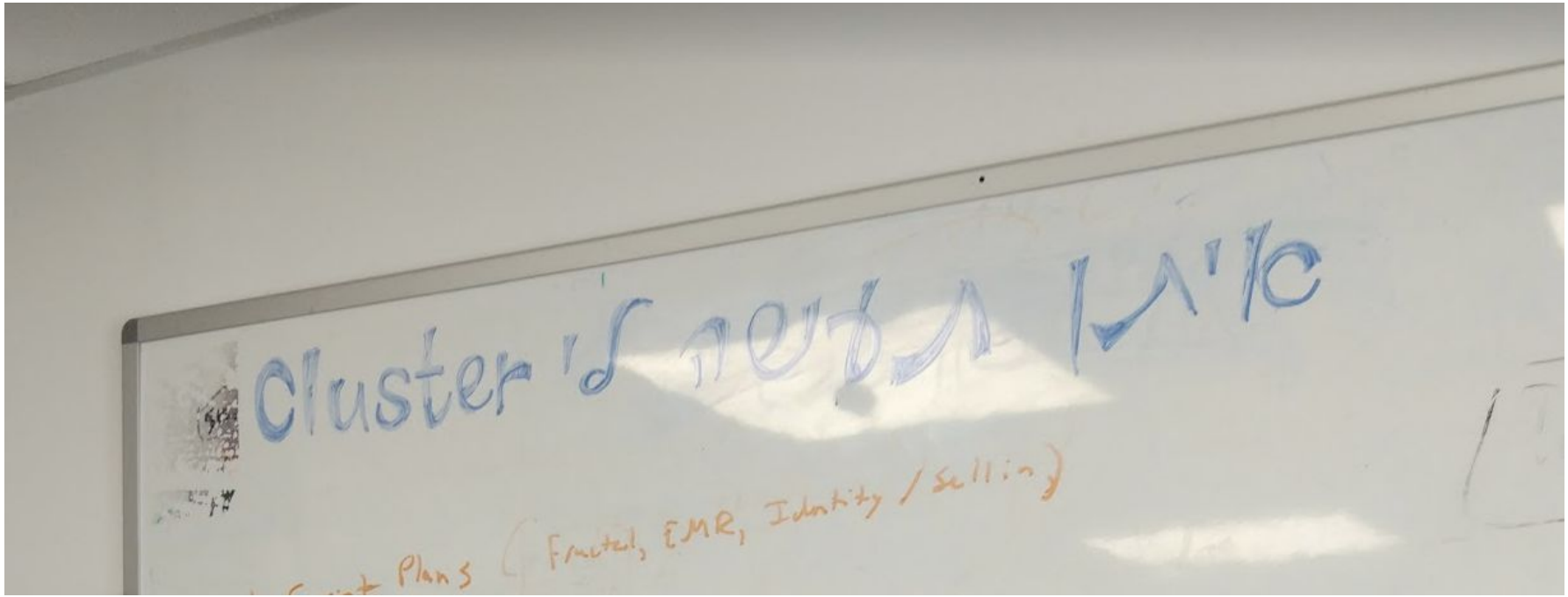
(3) On scheduled time - EMR transient cluster created with jobs to perform point of sale ETLs from Hive tables on S3 to Redhsift.

EMR Cluster – Spark ETL jobs complete in minutes

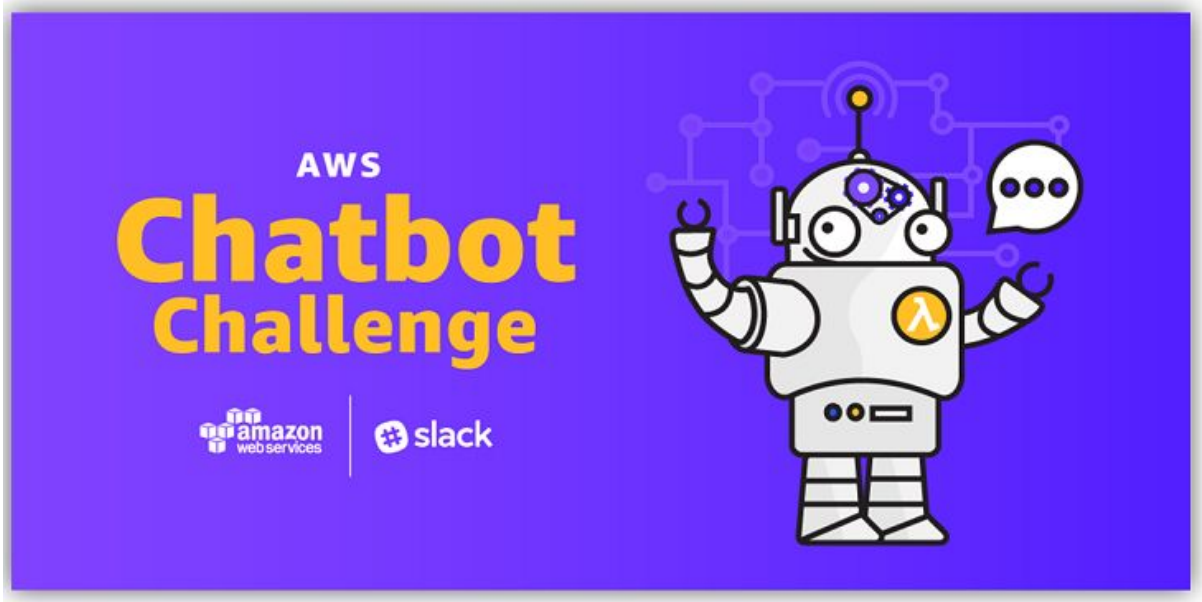


ID	Name	Status	Start time (UTC+3)	Elapsed time	Log files
s-DRT7HYVFPT9G	category_cache_daily_to_redshift_job-2018-07-07-12:10:51	Completed	2018-07-07 16:36 (UTC+3)	3 minutes	View logs
s-2T614470Z47WN	bars_category_performance_to_redshift_job-2018-07-07-12:10:51	Completed	2018-07-07 16:31 (UTC+3)	5 minutes	View logs
s-1O9UF6QV8RT7T	draught_serving_sizes_daily_to_redshift_job-2018-07-07-12:10:51	Completed	2018-07-07 16:29 (UTC+3)	1 minute	View logs
s-12DOZ5M20NIUS	export_draught_cache_daily_to_redshift_job-2018-07-07-12:10:51	Completed	2018-07-07 16:27 (UTC+3)	1 minute	View logs
s-3138YUA0C4WS2	waste_cache_daily_job-2018-07-07-12:10:51	Completed	2018-07-07 16:25 (UTC+3)	1 minute	View logs
s-1LJT1HIKL58IC	waste_cache_hourly_job-2018-07-07-12:10:51	Completed	2018-07-07 16:15 (UTC+3)	10 minutes	View logs
s-SPTSAC9LFGZB	pos_serving_sizes_daily_job-2018-07-07-12:10:51	Completed	2018-07-07 16:11 (UTC+3)	3 minutes	View logs
s-2XOG89577GJWF	pos_cache_daily_job-2018-07-07-12:10:51	Completed	2018-07-07 16:04 (UTC+3)	6 minutes	View logs
s-19X9CI7V11707	pos_insights_daily_job-2018-07-07-12:10:51	Completed	2018-07-07 16:00 (UTC+3)	4 minutes	View logs
s-117UVFUA89T5V	pos_insights_hourly_job-2018-07-07-12:10:51	Completed	2018-07-07 15:56 (UTC+3)	4 minutes	View logs

As more developers are developing PySpark Jobs…



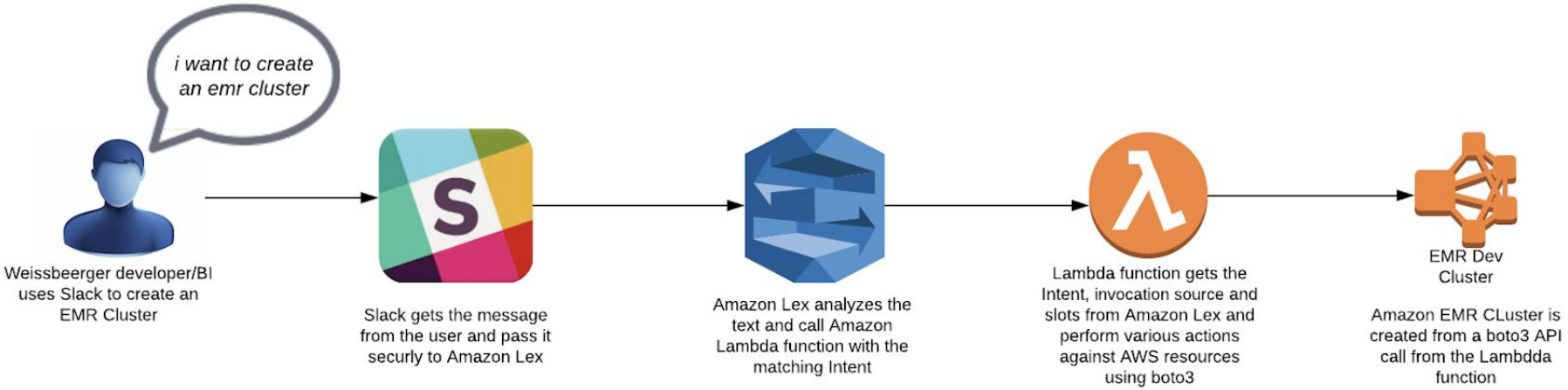
Our new Slack Chabot for EMR, using Amazon Lex



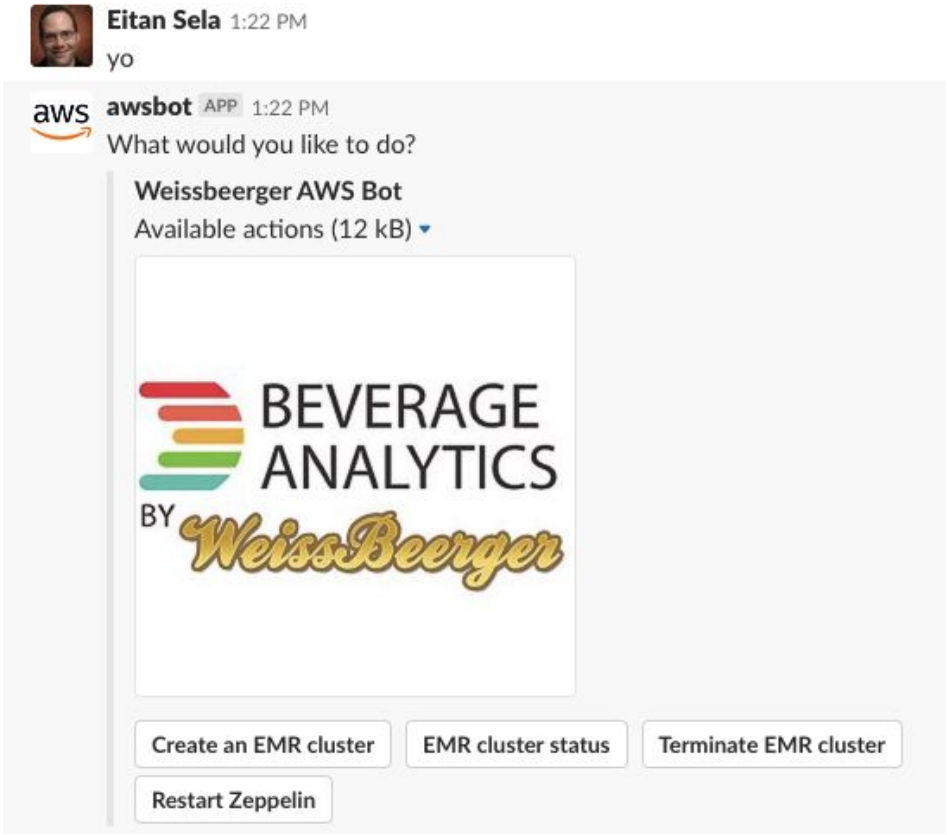
Amazon Lex

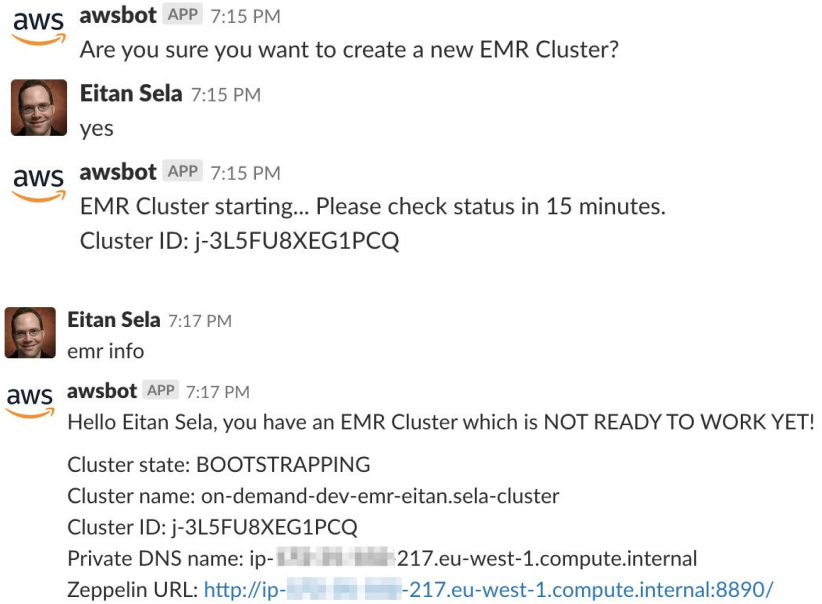
- Conversational interfaces for your applications.
- Powered by the same deep learning technologies as Alexa.
- Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU).

awsbot - the chatbot that help you manage AWS resources

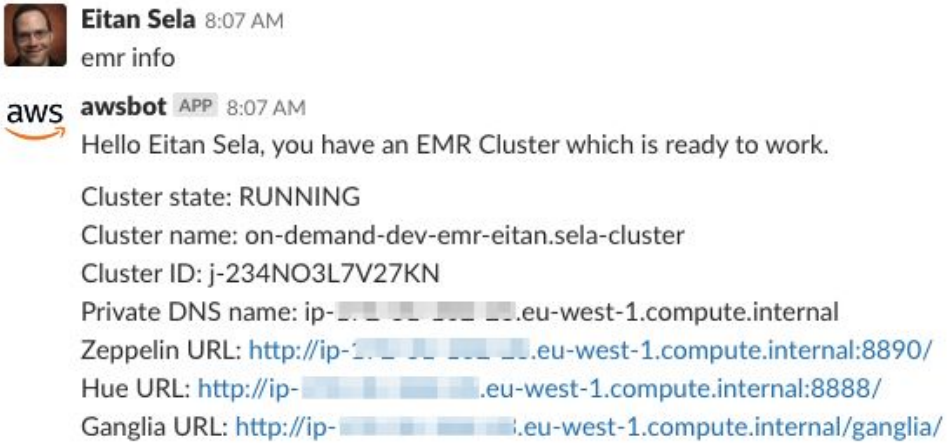


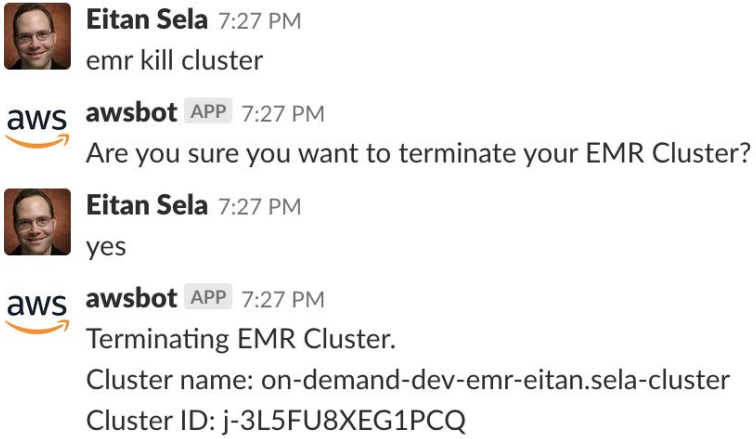
awsbot - Demo


Eitan Sela 1:22 PM
yo
awsbot APP 1:22 PM
What would you like to do?
Weissbeerger AWS Bot
Available actions (12 kB)
BEVERAGE ANALYTICS BY WeissBeerger
Create an EMR cluster
EMR cluster status
Terminate EMR cluster
Restart Zeppelin


awsbot APP 7:15 PM
Are you sure you want to create a new EMR Cluster?
Eitan Sela 7:15 PM
yes
awsbot APP 7:15 PM
EMR Cluster starting... Please check status in 15 minutes.
Cluster ID: j-3L5FU8XEG1PCQ
Eitan Sela 7:17 PM
emr info
awsbot APP 7:17 PM
Hello Eitan Sela, you have an EMR Cluster which is NOT READY TO WORK YET!
Cluster state: BOOTSTRAPPING
Cluster name: on-demand-dev-emr-eitan.sela-cluster
Cluster ID: j-3L5FU8XEG1PCQ
Private DNS name: ip-[redacted]217.eu-west-1.compute.internal
Zeppelin URL: http://ip-[redacted]-217.eu-west-1.compute.internal:8890/

awsbot - Demo - EMR Cluster is ready





Q & A

We Are Hiring!

Senior Data Scientist
Senior Designer (UI/UX)
Senior Full Stack Developer
Java Developer
Senior Manual QA
Director of Ops
BI Analyst
Data Management Analyst
Customer Success Manager
Senior BI Analyst

Squeegee

Open Source, "Serverless" AWS Cost and Usage Analysis at Scale.

whois

@ElliottSpira

CEO @ GorillaStack

GorillaStack

In Israel on a NSW Government trade mission

What's squeegee?

- ➔ OSS
- ➔ Mike Fuller @ Atlassian
- ➔ Serverless ETL pipeline for unstructured AWS Cost and Usage report data

Amazon Glue

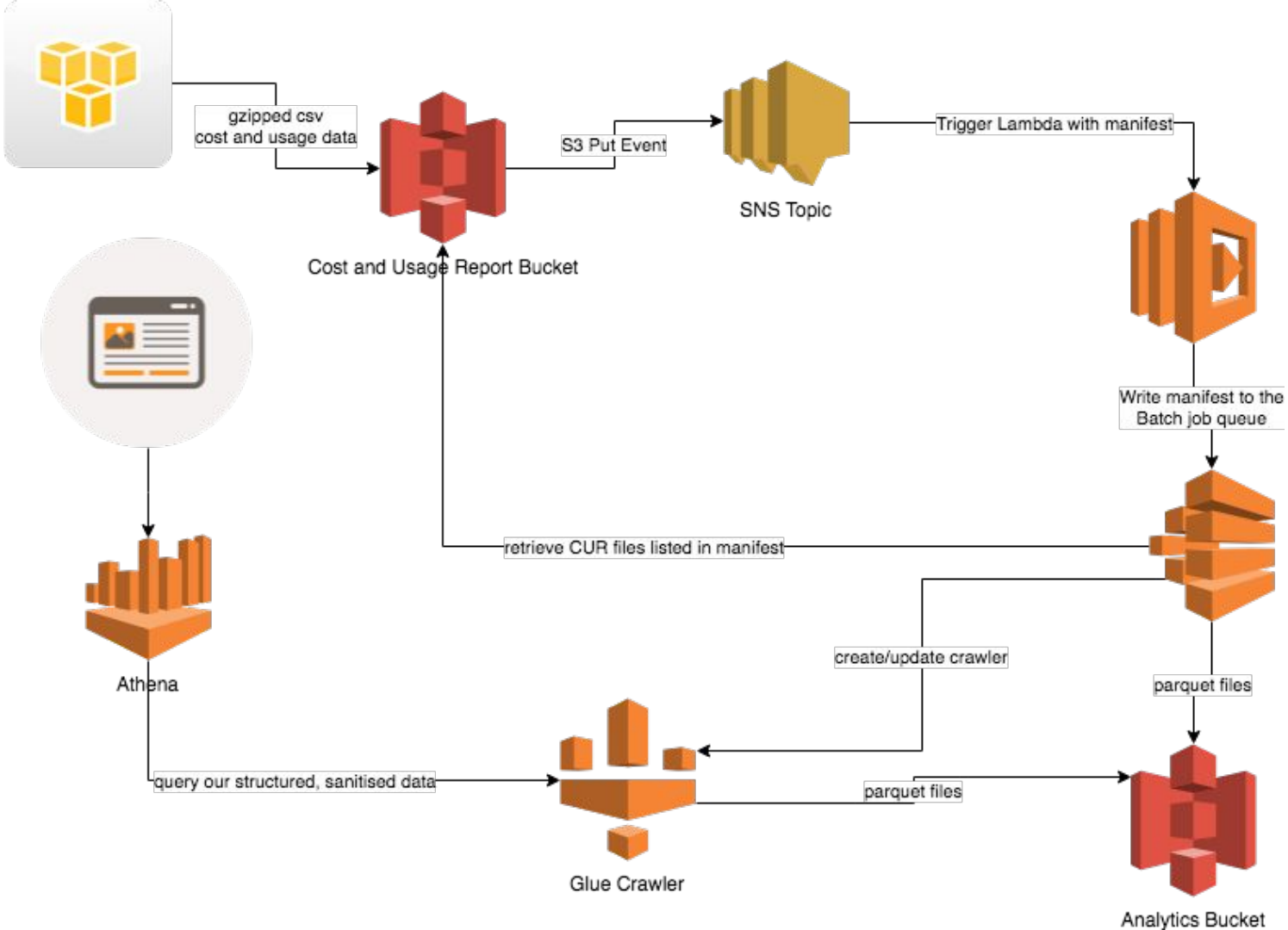
- ➔ A fully managed, serverless, ETL (extract, transform, load) service
- ➔ Source data from S3, Redshift, RDS and SQL DBs running on EC2 instances
- ➔ Once data is in an AWS Glue Data Catalog, it is available for analysis with Athena, EMR and Redshift Spectrum

Amazon Athena

- ➔ Interactive query service that makes it easy to analyze data in Amazon S3 using standard (ANSI) SQL

- ➔ Serverless

- ➔ Works well with Glue Data Catalog (inferring schemas and partitions)

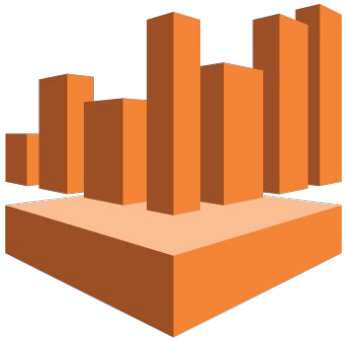

gzipped csv
cost and usage data
S3 Put Event
SNS Topic
Trigger Lambda with manifest
Cost and Usage Report Bucket
Write manifest to the
Batch job queue
retrieve CUR files listed in manifest
Athena
create/update crawler
parquet files
query our structured, sanitised data
parquet files
Glue Crawler
Analytics Bucket

Costs of our solution components

- ➔ Runs from $5-$10 for SME through $100 for enterprise
- ➔ S3: < 10GB Parquet files for small business, > 30GB for enterprise
- ➔ SNS: Should be nothing - only used to trigger Lambda
- ➔ Lambda: Only used to write Manifest file to Batch Job queue (negligible)
- ➔ CloudWatch Logs: Lambda writes some logs, should be < 5GB
- ➔ Glue: Glue crawler “$0.44 per DPU-Hour, 10 minute minimum crawler run, charged per second”. For twice daily CUR delivery, should be $4.55 a month
- ➔ Batch: Running spot instances for compute. Max vCPUs for squeegee is 64. Should generally be < $50/mo for most businesses

Demo

Thank you!



Come say hi after the presentation :)

Advanced GPU operations on AWS

When regular GPU hassles just don’t cut it anymore

A little bit about me

Independent DevOps consultant

Like to solve hard problems (do you have one?)

Social links:

www.linkedin.com/in/gilbahat

www.twitter.com/GilBahat

Knowing the terrain

A very very brief primer about GPU computing

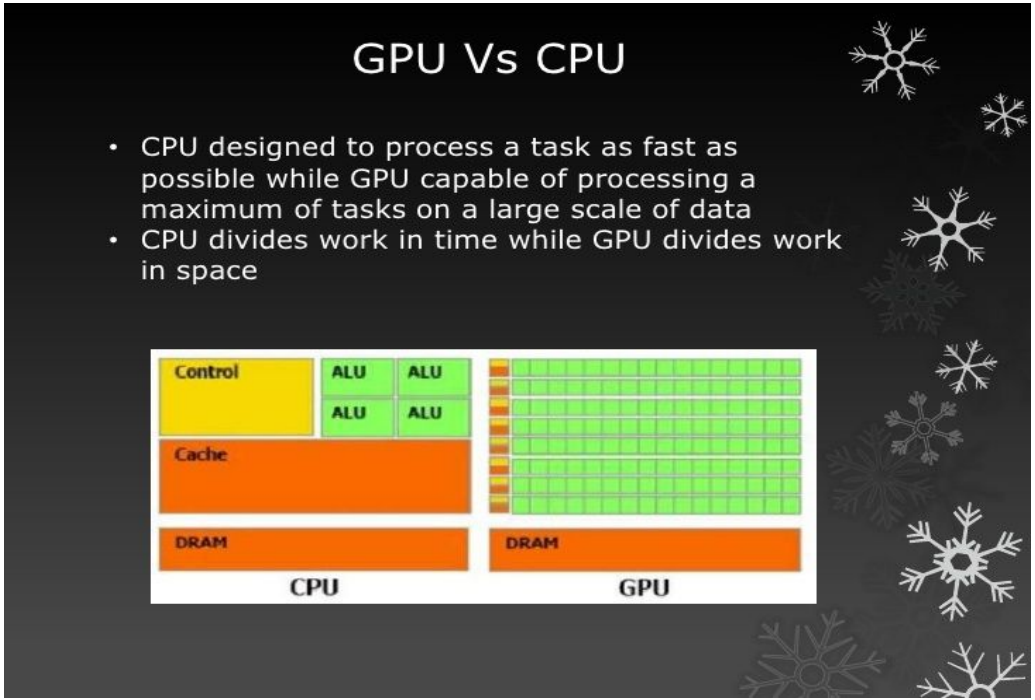
Runs on the parallel compute units of the GPU.

Doesn’t run on the CPU (but CPU needed for some iterations)

Runs on GPU VRAM (but may need main RAM for some iterations or prep work)

Runs via repeating ‘code kernels’ which are loaded into the GPU.

A very very brief primer about GPU computing



Source: Rajiv Kumar @ Slideshare

Another moment on our payload

Doesn’t share resources as much as CPU workloads do

(no context switching, avoids branching if possible, etc)

This is kinda not surprising given how GPUs work (vertex operations, shaders)

More often than not, highly tuned to requisite requirements

Even the programming platforms are not common (Cuda vs OpenCL vs RenderScript…)

Let’s make it more complex - spot market

We want to be cost efficient. This is more complex on GPUs due to lesser interchangeability.

Smart spot buying - going where it’s cheap!

GPUs - not as much as selection, needs to support multiple GPUs

Amazon GPU instance selection

Instance Family	GPU
cg1 (mostly obsolete)	Nvidia M2050
g2 (2xlarge-8xlarge)	Nvidia GRID K520
g3 (4xlarge-16xlarge)	Nvidia M60
p2 (xlarge-16xlarge)	Nvidia K80
p3 (2xlarge-16xlarge)	Nvidia V100

To docker or not to docker?

Docker is designed for ease of development by environment containment and for hyperconvergence

These goals are harder to achieve with GPU/AI oriented tasks

YMMV

(at the last time we surveyed it, we thought it wasn’t worth it)

Querying our GPU

We need to support more than one GPU type

We can hardcode, but we prefer to autodetect

The solution for that is called Nvidia-ML (management library)

The underpinnings of nvidia-smi

To ML or not to ML

Sometimes nvidia-ml just doesn’t cut it correctly.

Many bugs and weird nuances.

(ECC example)

The solution: mix it with alternate CUDA APIs.

CUDA capabilities APIs

PTX bytecode levels

Compute architecture levels

(we'll get to these back again!)

Multiple architecture builds

These builds are long, but maybe they pay off.

Tweak the build flags for dev, test and prod differently

Use single payload! it’s tempting to have multiple of them. Don’t.

AI Acceleration libraries

CudNN

CuBLAS

TensorRT

NCCL

https://developer.nvidia.com/gpu-accelerated-libraries

No good ability to know whether they are used or not

And now for GFX

AI and video

A lot of the AI work relates to video

Need to do video decoding

Onboard SIP Core

AI and GL

Need to do GL manipulations

Cuda/OpenGL interop - complex!

Going standalone and mutexing

Xorg required

Old beast

Driver issues

Not operations friendly

Not friendly to anyone...

Driver interop issues

NVRM: Xid (PCI:0000:00:03): 13, Graphics Exception: ChID 0006, Class 00009097, Offset 000023a8, Data 00000000

NVRM: Xid (PCI:0000:00:03): 8, Channel 00000009

NVRM: os_schedule: Attempted to yield the CPU while in atomic or interrupt context

Keeping devs in check

Free discussion (if we have time)

(About working as Ops with AI researchers and AI devs)

Questions?

Questions?

Ask in Slack #bigdata_on_aws

http://bit.ly/2ErwJa5