Big Data Processing

Anusuriya Devaraju

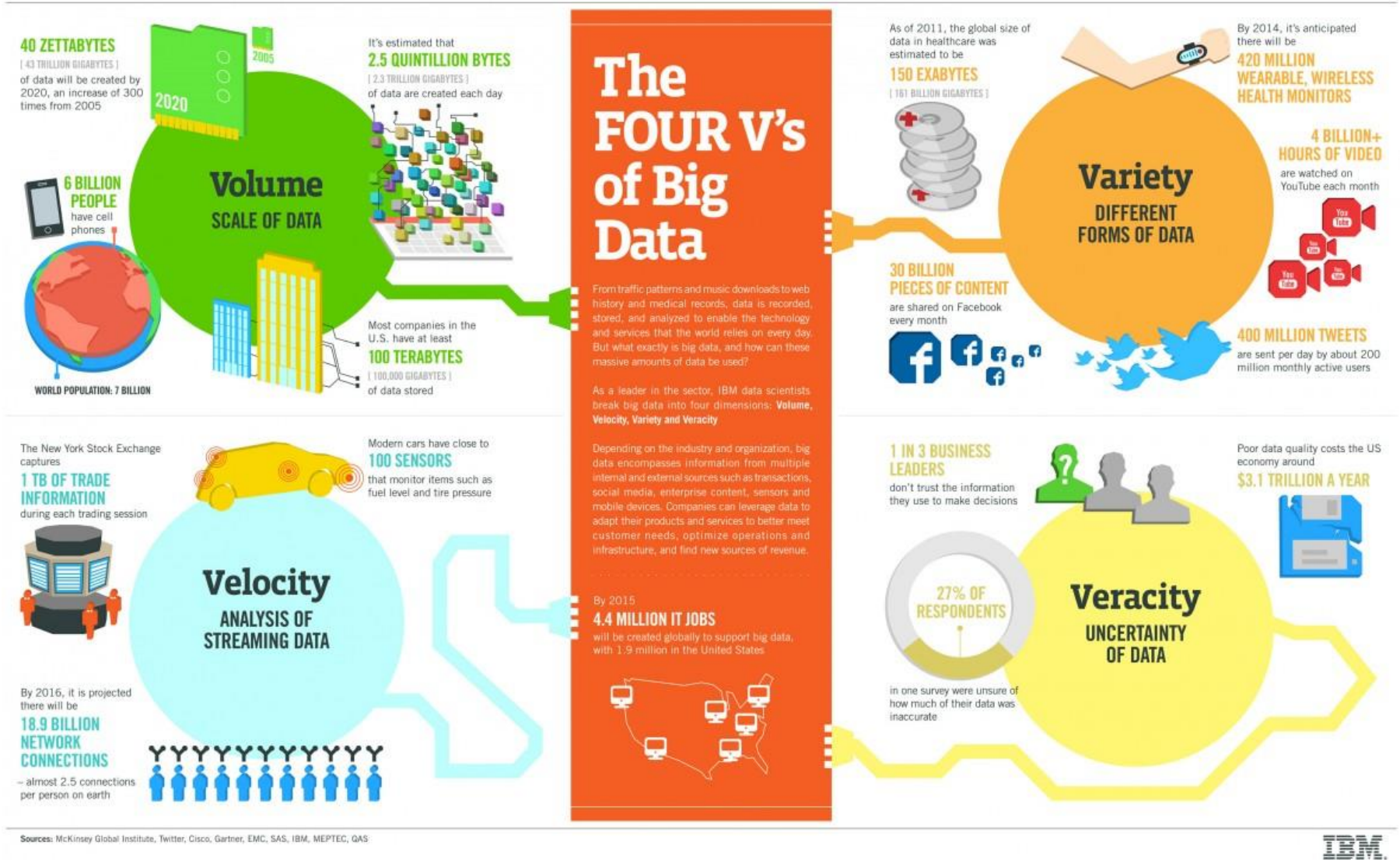
01.03.2017

Outline

- Basic Concepts
- Hands-on & Review Questions
  - Lab 1 - Using Amazon Athena to Analyze Log Data
  - Lab 2 - Processing Server Logs with Hive on Amazon EMR
  - Lab 3 - Apache Spark on Amazon Web Services EMR

Reference: Some of the following slides were modified from AWS. Big Data on AWS 3.2 (EN): Student Guide. AWS/Gilmore. VitalBook file.

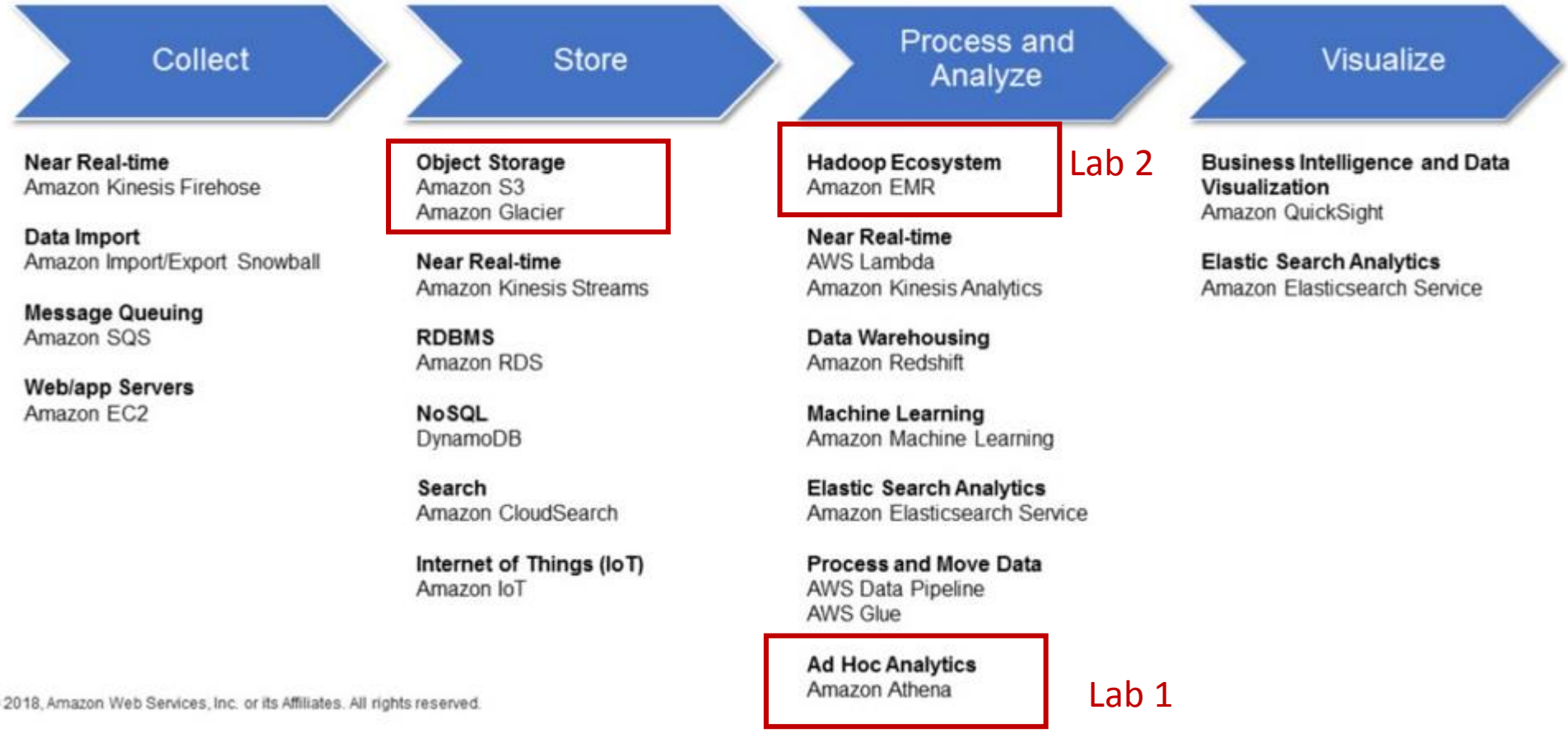
What is Big Data?



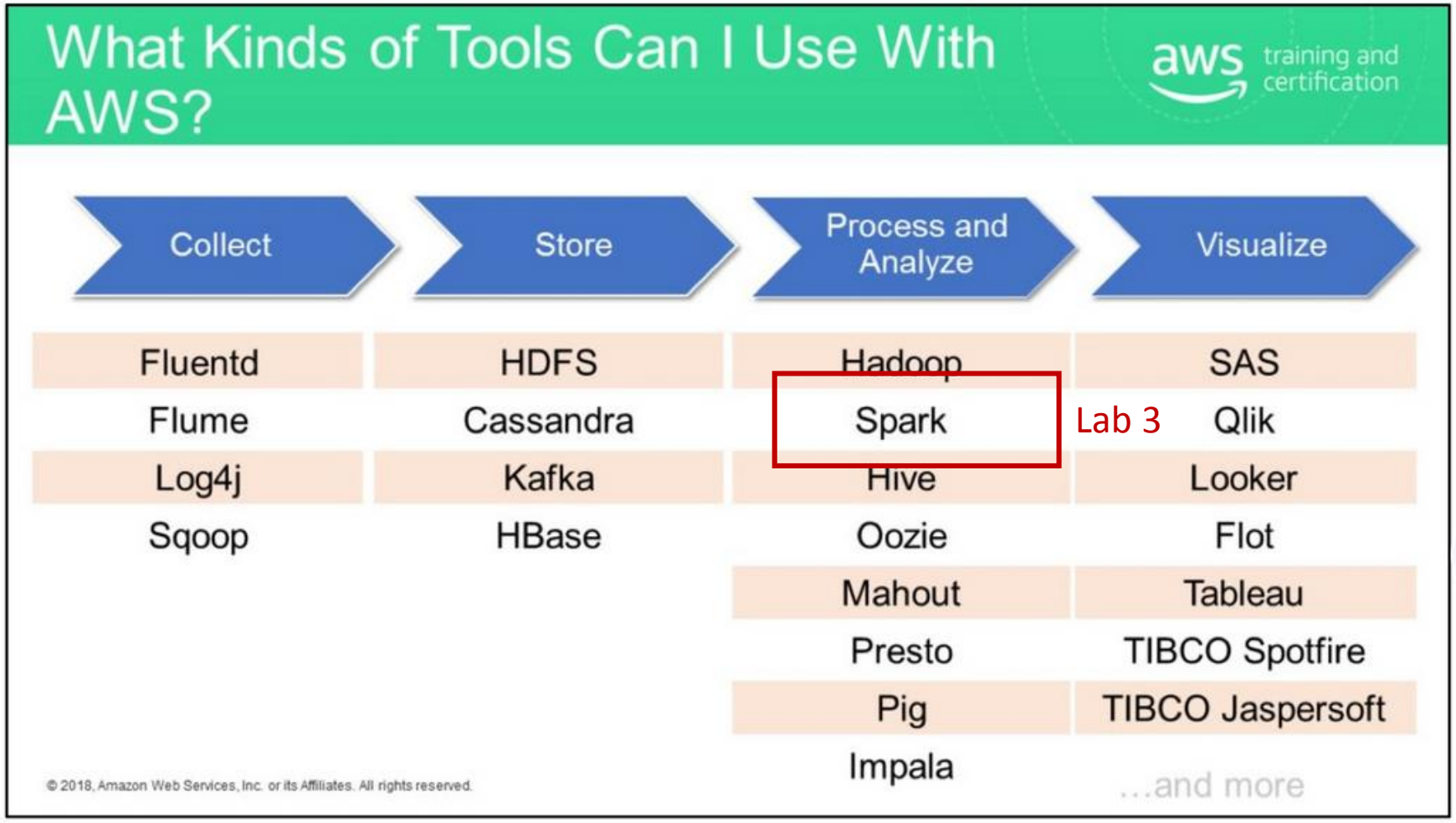
The Big Data Pipeline

Source: 2018, Amazon Web Services, Inc.

From AWS Solutions to the Big Data Pipeline



Other Tools That Can be Used With AWS



Outline

- Basic Concepts
- Hands-on & Review Questions
  - **Lab 1 - Using Amazon Athena to Analyze Log Data**
  - Lab 2 - Processing Server Logs with Hive on Amazon EMR
  - Lab 3 - Apache Spark on Amazon Web Services EMR

Lab 1 - Using Amazon Athena to Analyze Log Data

- The following log data is provided in an Amazon S3 bucket stored in the US East (North Virginia) Region.

```
2015-01-01T00:00:00.022719Z elb_demo_005 244.218.91.244:2255 172.36.231.239:443
0.000878 0.000803 0.000891 200 200 0 1886 "GET https://www.example.com/jobs/376
HTTP/1.1" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/47.0.2526.111 Safari/537.36" DHE-RSA-AES128-SHA TLSv1.2
```

- Raw data stored in plain text
  a) Raw data stored in plain text
  b) Compressed data in gzip format
  c) Compressed and Partitioned data split into sub-directories
  d) Columnar data, stored in Parquet format

Lab 1 - Results

TASKS	RUNTIME & DATA SCANNED
Task 1: Raw data stored in plain text	
Task 2: Compressed data in gzip format	
Task 3: Compressed and Partitioned data split into sub-directories	
Task 4: Columnar data (Parquet with Snappy compression)	

Lab 1 – Conclusions and Q&A

Q1. Name few benefits of Amazon Athena

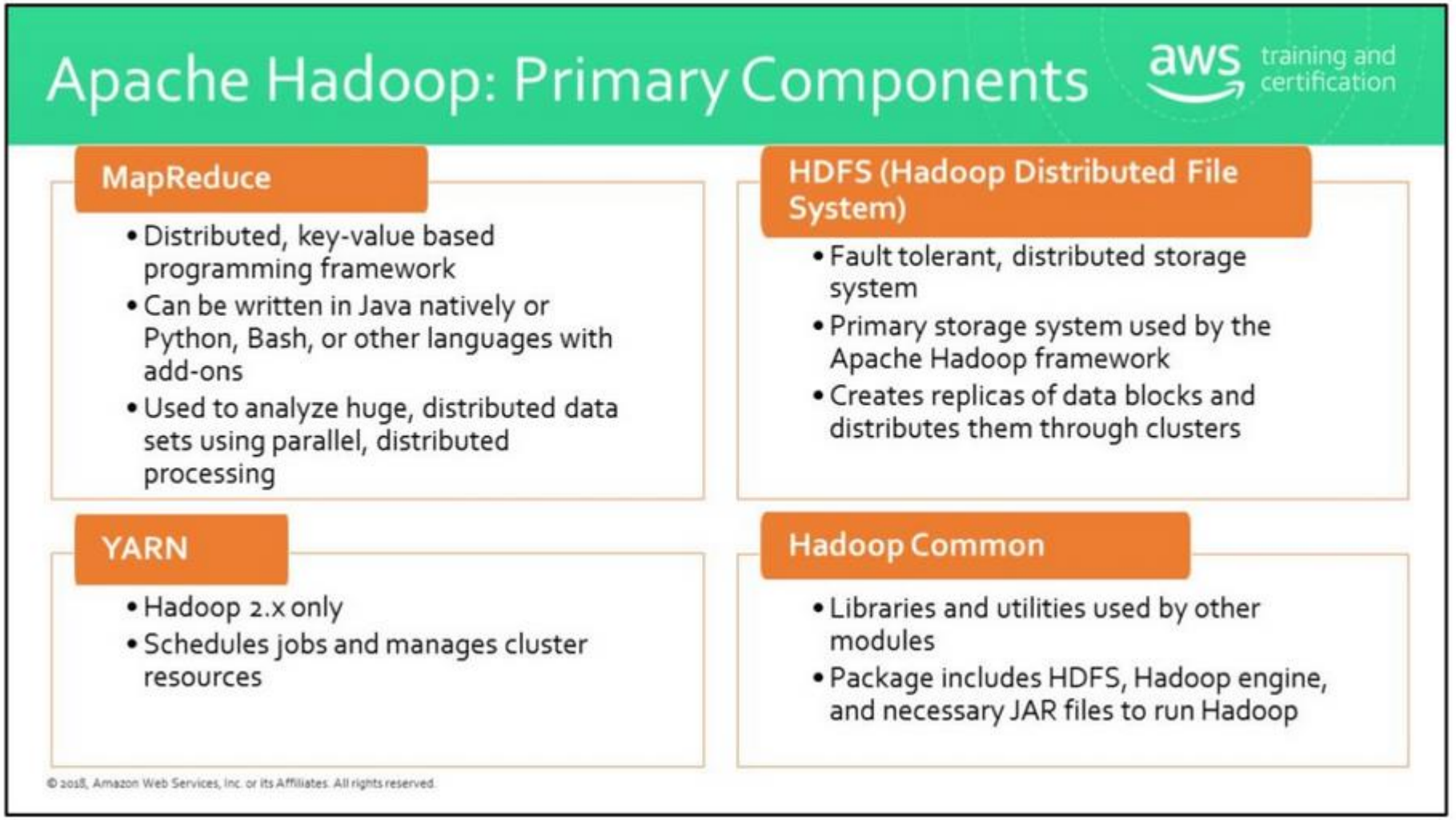
Q2. In your Lab 1, which data format improve the performance and save cost in Amazon Athena, and why?

Q3. Where does Amazon Athena store information and schemas about database?

Outline

- Basic Concepts
- Hands-on & Review Questions
  - Lab 1 - Using Amazon Athena to Analyze Log Data
  - **Lab 2 - Processing Server Logs with Hive on Amazon EMR**
  - Lab 3 - Apache Spark on Amazon Web Services EMR

Apache Hadoop



Amazon EMR

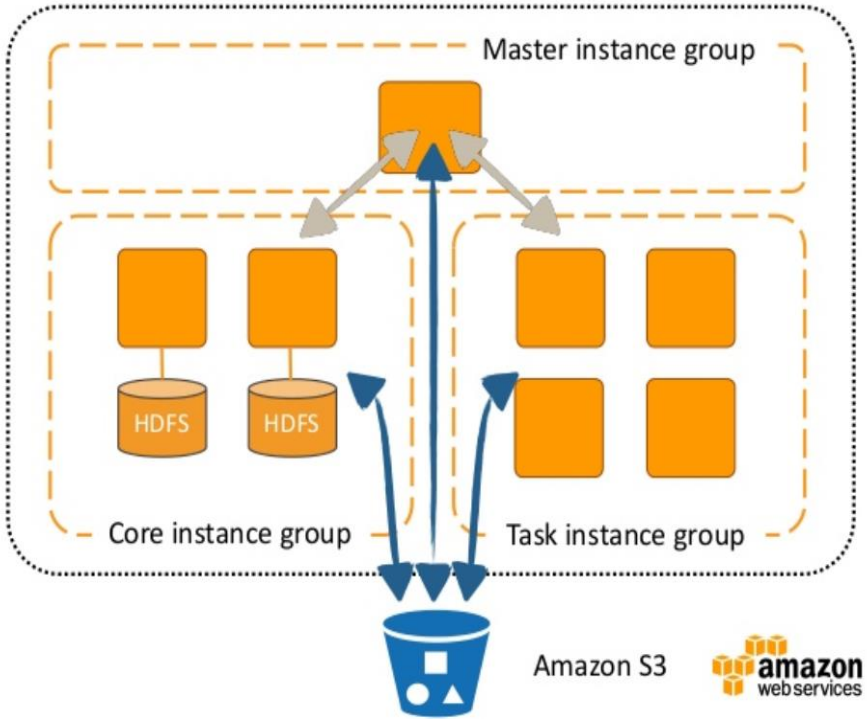
- Managed cluster platform
- Run Hadoop, Spark and other applications
- Easy – launch cluster in minutes!
- Elastic – change your cluster dynamically to response to increased workloads.
- Use HDFS and S3 file systems

Amazon EMR Architecture

Basic EMR Architecture

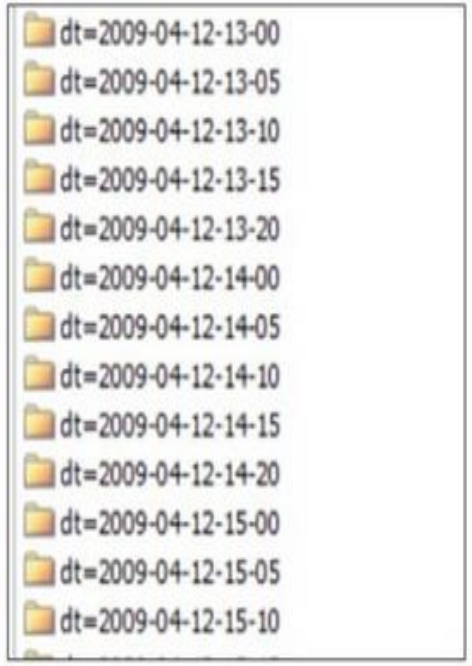
aws training and certification

- **Master** node:
  - Manages resources of cluster
  - Schedules jobs on worker instances
- **Core** nodes:
  - CPU
  - HDFS storage
  - RAM
- **Task** nodes:
  - CPU
  - RAM



Lab 2 - Processing Server Logs with Hive on Amazon EMR

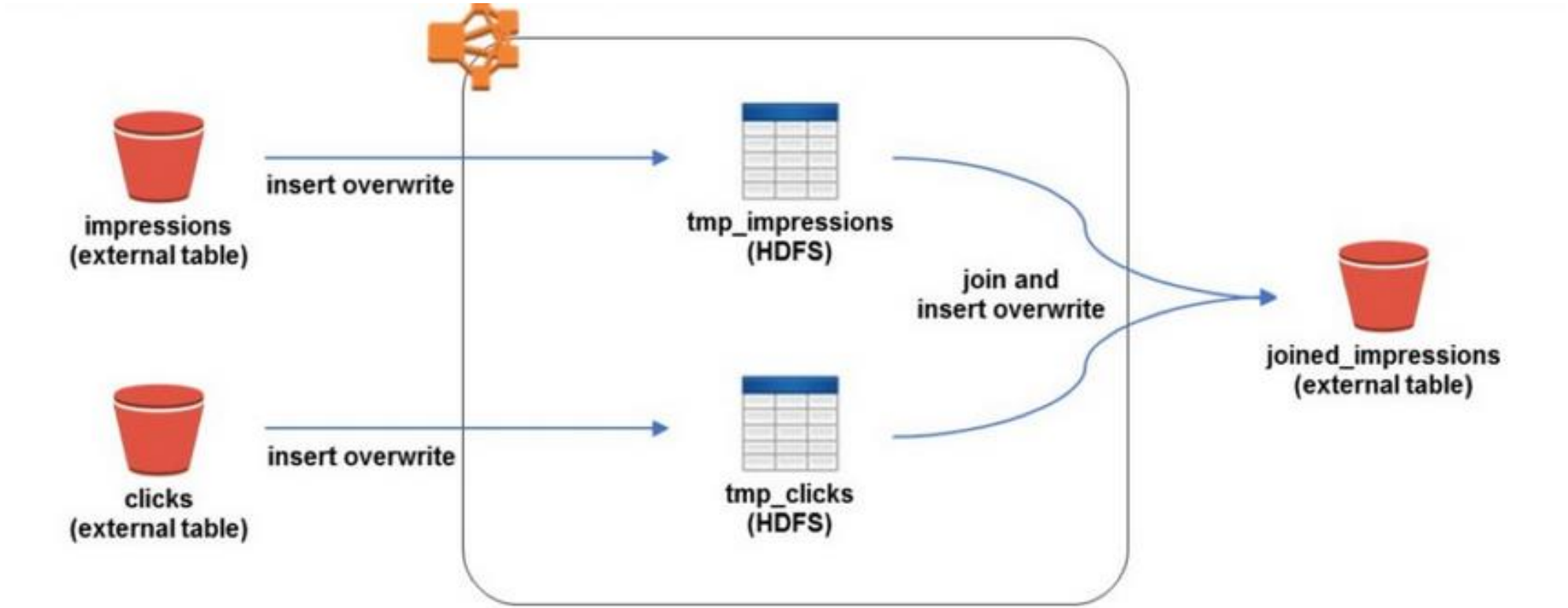
- Launch an Amazon EMR cluster with Apache Hive installed
- Use Hive commands to create external relation database tables from log data stored in Amazon S3
- Use Hive to query the tables you create and persist in Amazon S3.



Stored log structure in Amazon S3

Log data (sample)

Lab 2 - Processing Server Logs with Hive on Amazon EMR



Lab 2 - Review

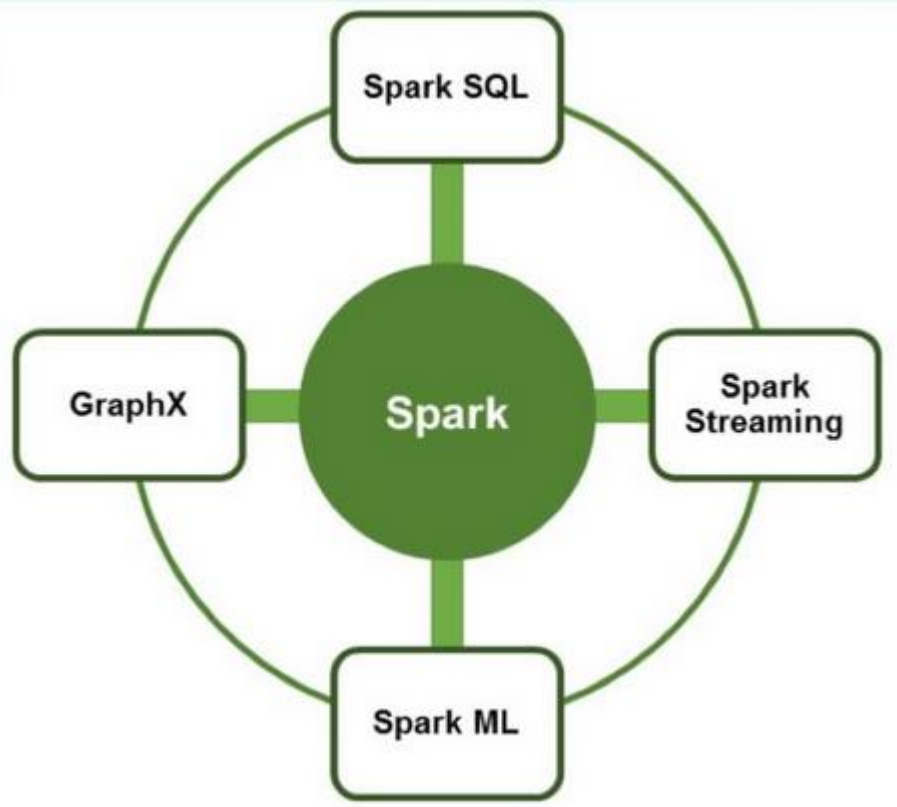
- What are the primary Apache Hadoop components?
- List applications for which Apache Hadoop is not suitable.
- What is the difference between a transient and a long-running cluster?
- List advantages of running Hive and Amazon EMR

Outline

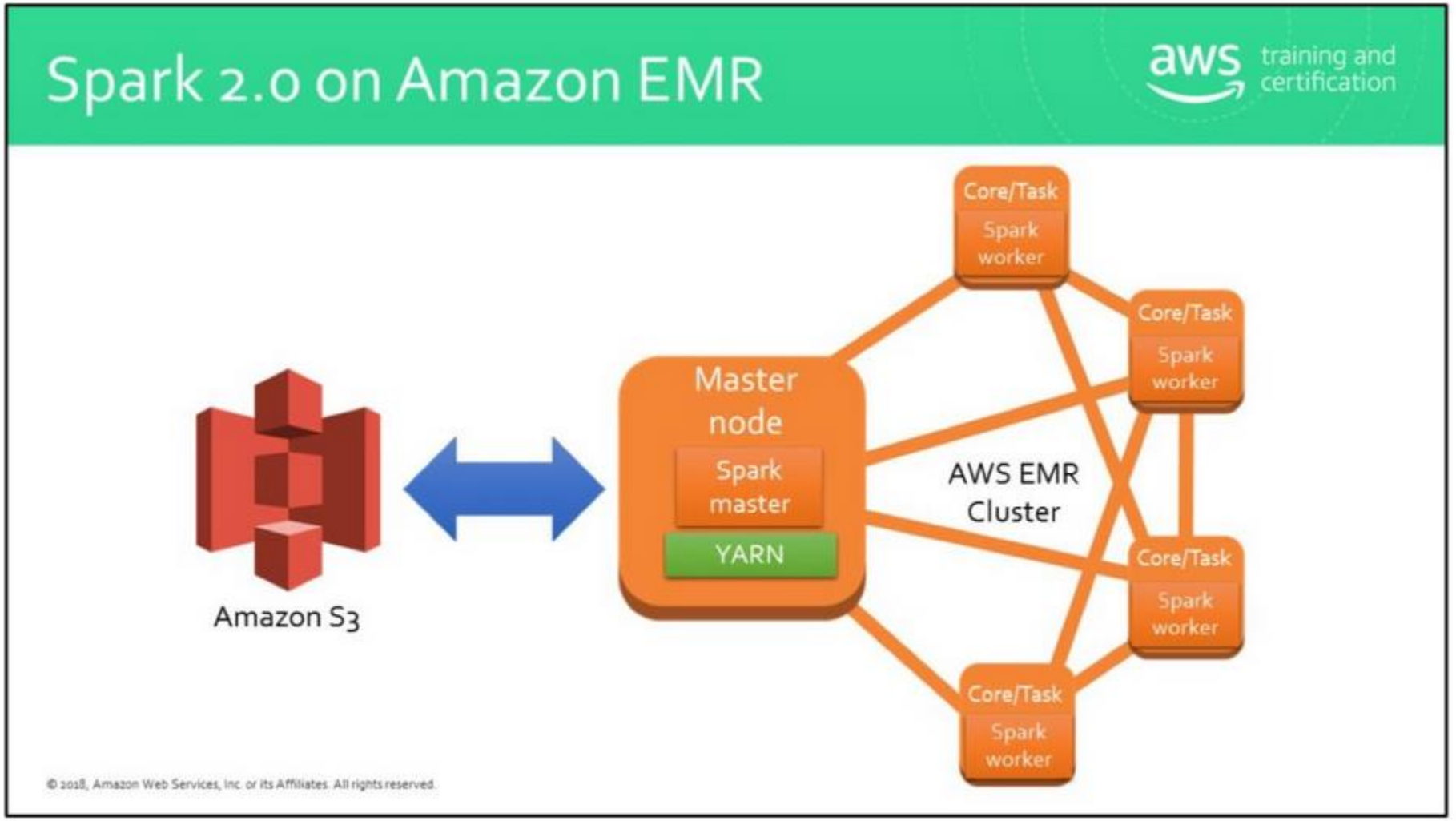
- Basic Concepts
- Hands-on & Review Questions
  - Lab 1 - Using Amazon Athena to Analyze Log Data
  - Lab 2 - Processing Server Logs with Hive on Amazon EMR
  - **Lab 3 - Apache Spark on Amazon Web Services EMR**

Apache Sparks

- Allows in-memory data mining and querying big datasets at fast speeds
- Resilient distributed datasets (RDD)
- Faster than MapReduce!
- Supports batch, interactive and streaming applications
- Can read and store data in HDFS, Amazon S3, and other databases.
- JDBC/ODBC connector compatible



Spark 2.0 on Amazon EMR



Lab 3 - Apache Spark on Amazon Web Services EMR

1. Create an AWS S3 storage bucket to hold data inputs, data outputs and logs
2. Create, Configure and Launch an Amazon EMR Cluster
3. Log in to the cluster master node
4. Run Spark-SQL Command Line Interface and Issue commands to create tables and run queries
5. Shut down cluster and remove any temporary resources

Lab 3 - Review

- SparkQL Exercise: Find the total trading volume (trade_size) for each stock before 12:00 noon
- What are the alternatives to load data to S3 from your local PC?

Data61 Research Big Data Cluster