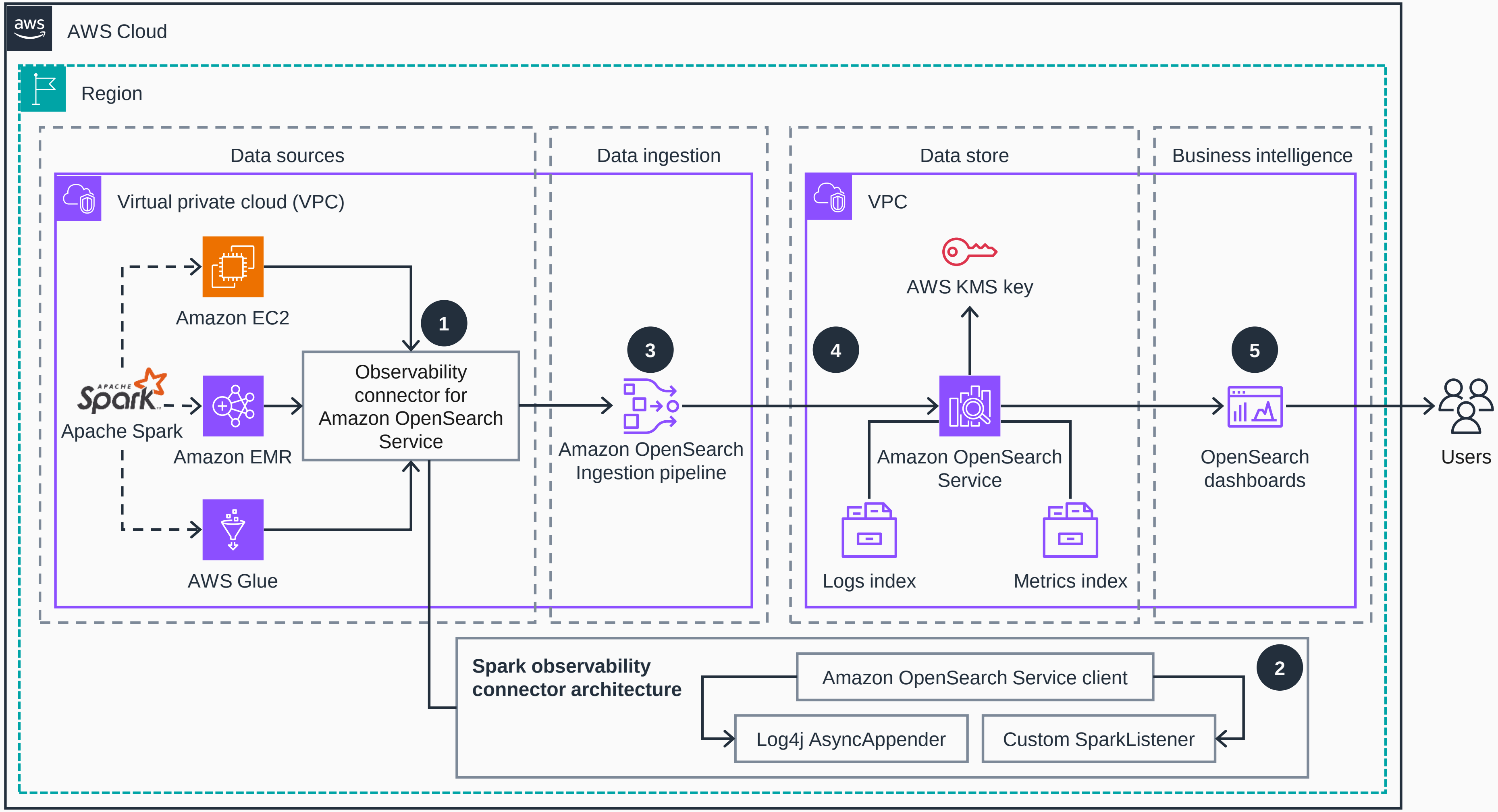
Guidance for Analytics Observability on AWS

This architecture diagram shows how to enhance data pipeline observability on Apache Spark, aggregating metrics and logs from AWS services that perform extract, transform, and load (ETL) processes. It aids in identifying optimization opportunities and reducing monitoring overhead.



1. The observability connector for **Amazon OpenSearch Service** is packaged into Apache Spark applications running through **Amazon EMR** or **AWS Glue** or self-hosted on **Amazon Elastic Compute Cloud (Amazon EC2)**. The connector is a Java Archive (JAR) file to put on the driver and executor classpaths.

2. The observability connector includes a custom log appender (Log4j AsyncAppender) and a custom SparkListener. They collect logs and metrics from the application and push the data out through the **OpenSearch Service** client.

3. The observability connector pushes the data into an Amazon OpenSearch Ingestion pipeline. The pipeline applies data transformation and also acts as an ingestion buffer into **OpenSearch Service**.

4. Ingestion-related logs and metrics are stored in **OpenSearch Service** indexes: one for each data type. The data delivery frequency is defined as part of the **OpenSearch Service** pipeline configuration. Log and metric data are encrypted using an **AWS Key Management Service (AWS KMS)** key.

5. Prebuilt **OpenSearch** Dashboards is a tool that offers authenticated users insights into their data pipelines using aggregated views of performance metrics and logs at various levels of granularity, such as Spark application, job run, stage, and partition. The dashboard also provides performance scores calculated based on the collected metrics to enable easier analysis.

Reviewed for technical accuracy April 26, 2024

AWS Reference Architecture