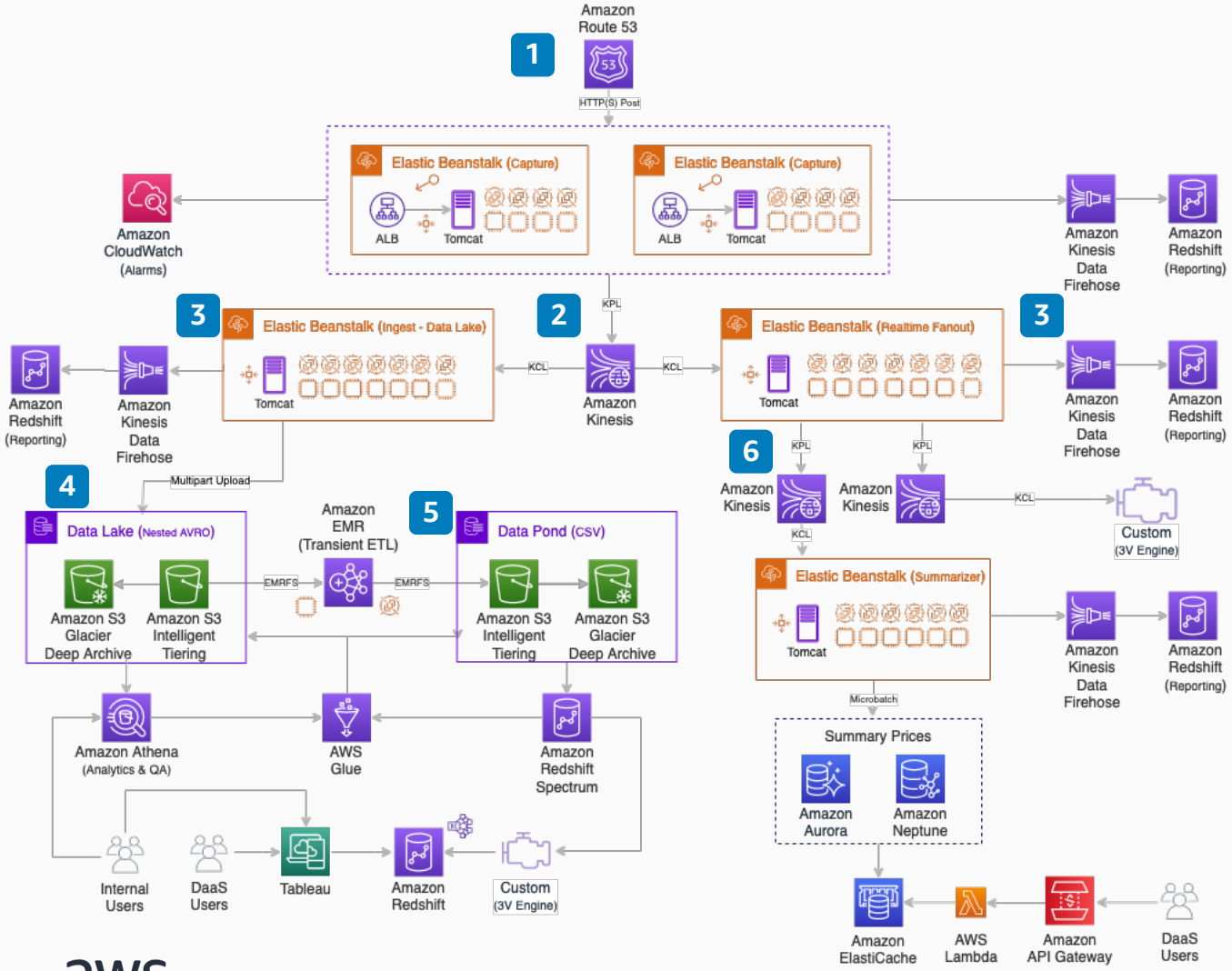
Streaming Airline Ticket Shopping Insights

Predictive Analytics on Timeseries Airline Shopping Data

3Victors implemented an AWS cloud-based architecture to capture and durably store over 10Tb of daily streamed air shopping data messages into a Data Lake. Dozens of ETL jobs run at regular intervals to populate use case specific Data Ponds. Simultaneously, the implementation provides an extensible, real-time predictive analytics pipeline for demand forecasting and deal classification.



1. **Amazon Route 53** directs per stream vendor content to source specific **AWS Elastic Beanstalk** application orchestration 20k/sec HTTPS POST transactions autoscaling for peaks and valleys using **Amazon EC2** spot instances when possible and monitored with **Amazon CloudWatch** while logging application specific statistics to an **Amazon Redshift** columnar data warehouse via **Amazon Kinesis Data Firehose**.

2. Captured pricing data is lightly transformed into a common format, serialized and placed on a durable 24-hour **Amazon Kinesis** streaming buffer balanced across 100+ shards supporting checkpoint time range replay for downstream issue remediation.

3. Two concurrent pipelines fan out from the streaming buffer (cost optimized using the 2x read vs. write stream buffer capability) feeding **AWS Elastic Beanstalk** compute clusters that read and deserialize the data in real time.

4. A Data Lake pipeline durably persists in a cost-optimized **Amazon S3 Intelligent Tiering** bucket via multipart upload with directory paths suitable for partitioning and cross account accessibility. Schema on read Data lake mappings are updated in **AWS Glue** metastore supporting **Amazon Athena** analytics. Bucket-defined life cycle policies for archiving to **Amazon S3 Glacier** Deep Archive optimize cost.

5. Periodic ETL jobs based on customer SLA deposit results into cross account accessible Data Ponds using the **Amazon EMR** transient massively parallel compute framework, cost optimized with spot instance fleets. Schema on read Data Pond mappings updated in **AWS Glue** metastore facilitating **Amazon Redshift** Spectrum access integrated with **Amazon Redshift** whereby customer jobs are run that join and filter ponds into columnar tables for BI Tool access.

6. A real-time analytics pipeline fans out to unlimited **Amazon Kinesis** stream buffers that support an extensible micro-batching framework. A default use case populates **Amazon Aurora** (row) and **Amazon Neptune** (graph) databases for customer API access via serverless **Amazon API Gateway** and **AWS Lambda**. Database I/O optimized using **Amazon ElastiCache** cache.

Reviewed for technical accuracy March 10, 2021

AWS Reference Architecture