Avaro

What is Athena?

- Interactive query service for S3 (SQL)
  - No need to load data, it stays in S3
- Presto under the hood
- Serverless!
- Supports many data formats
  - CSV (human readable)
  - JSON (human readable)
  - ORC (columnar, splittable)
  - Parquet (columnar, splittable)
  - Avro (splittable)
- Unstructured, semi-structured, or structured

Amazon Athena

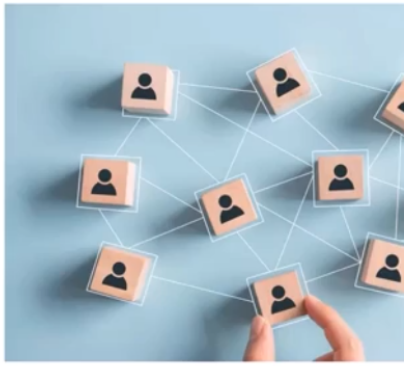
Some examples

- Ad-hoc queries of web logs
- Querying staging data before loading to Redshift
- Analyze CloudTrail / CloudFront / VPC / ELB etc logs in S3
- Integration with Jupyter, Zeppelin, RStudio notebooks
- Integration with QuickSight
- Integration via ODBC / JDBC with other visualization tools

Athena + Glue

Athena Workgroups

- Can organize users / teams / apps / workloads into Workgroups
- Can control query access and track costs by Workgroup
- Integrates with IAM, CloudWatch, SNS
- Each workgroup can have its own:
  - Query history
  - Data limits (*you can limit how much data queries may scan by workgroup*)
  - IAM policies
  - Encryption settings



Athena cost model

- Pay-as-you-go
  - $5 per TB scanned
  - Successful or cancelled queries count, failed queries do not.
  - No charge for DDL (CREATE/ALTER/DROP etc.)
- Save LOTS of money by using columnar formats
  - ORC, Parquet
  - Save 30-90%, and get better performance
- Glue and S3 have their own charges

Athena Security

- Access control
  - IAM, ACLs, S3 bucket policies
  - AmazonAthenaFullAccess / AWSQuicksightAthenaAccess
- Encrypt results at rest in S3 staging directory
  - Server-side encryption with S3-managed key (SSE-S3)
  - Server-side encryption with KMS key (SSE-KMS)
  - Client-side encryption with KMS key (CSE-KMS)
- Cross-account access in S3 bucket policy possible
- Transport Layer Security (TLS) encrypts in-transit (between Athena and S3)

Athena anti-patterns

- Highly formatted reports / visualization
  - That's what QuickSight is for
- ETL
  - Use Glue instead

Athena: Optimizing performance

- Use columnar data (ORC, Parquet)
- Small number of large files performs better than large number of small files
- Use partitions
  - If adding partitions after the fact, use MSCK REPAIR TABLE command

Athena ACID transactions

- Powered by Apache Iceberg
  - Just add ‘table_type’ = ‘ICEBERG’ in your CREATE TABLE command
- Concurrent users can safely make row-level modifications
- Compatible with EMR, Spark, anything that supports Iceberg table format.
- Removes need for custom record locking
- Time travel operations
  - Recover data recently deleted with a SELECT statement
- Remember governed tables in Lake Formation? This is another way of getting ACID features in Athena.
- Benefits from periodic compaction to preserve performance

```
OPTIMIZE table REWRITE DATA
  USING BIN_PACK
  WHERE catalog = 'c1'
```