Intro to AWS Data Lakes and Analytics

Immersion Day

David Bayard, Solutions Architect
Padmaja Suren, Solutions Architect
Zach Gardner, Solutions Architect

October 1, 2019

If you wish to do today's hands-on labs, please send an email to gardz@amazon.com

Agenda – Part 1

9:00 am (15 minutes)
- Introductions & Objectives

9:15 am (90 minutes)
- Cloud Data Lakes
  - What is a Cloud Data Lake?
  - Things you can do with a Cloud Data Lake
  - Building a Cloud Data Lake on AWS
- Lab: Building a Data Lake with S3 & AWS Glue & Athena

11:00 am (45 minutes)
- Cloud Data Warehouse
  - What is a Cloud Data Warehouse and how does it fit with a Cloud Data Lake?
  - Things you can do with a Cloud Data Warehouse
  - Building a Cloud Data Warehouse on AWS
- Lab: Extending the lake with Amazon Redshift

12:00 pm (30 minutes)
- Lunch

Agenda – Part 2

12:30 pm (30 minutes)
- Overview of Lake Formation
- Demonstration: Lake Formation

1:00 pm (75 minutes)
- Populating the Data Lake/Data Warehouse
  - Discussion of different complimentary approaches
    - Database Migration Service (DMS) and Schema Conversion Tool (SCT)
    - Lake Formation Blueprint and AWS Glue ETL
    - Streaming Data (Kinesis/Kafka)
    - 3rd party approaches (informatica, etc)

2:30 pm (30 minutes)
- Visualizing Data
  - Quicksight Overview
- Demonstration of Quicksight

3:00 pm (30 minutes)
- Wrap-up
- Clean-up of Lab Environments

Lab Preparation

If you wish to do today's hands-on labs, please send an email to gardz@amazon.com

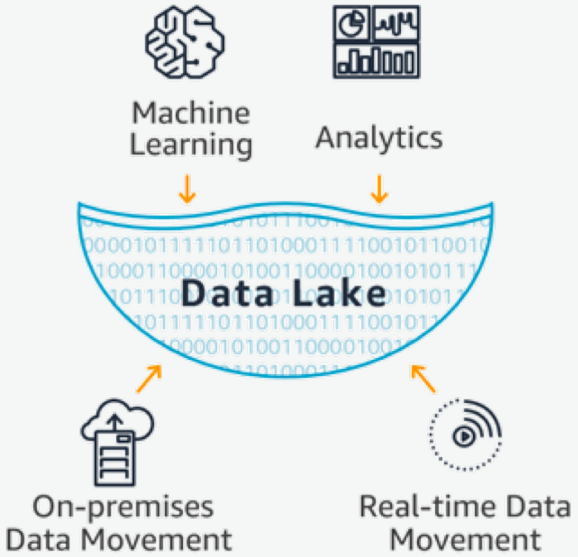
Introductions and Objectives

Q1: What is your level of experience with AWS?
Q2: What is your experience with data lakes and/or data warehousing?

Cloud Data Lakes

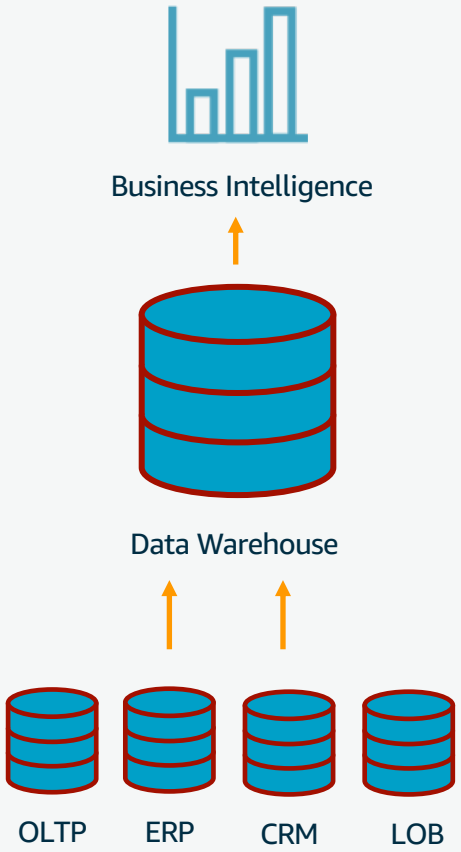
What is a Data Lake?

- A centralized repository for both structured and unstructured data
- Store data as-is in open-source file formats to enable direct analytics



aws

Traditionally, Analytics Looked Like This



Relational Data

TBs Scale

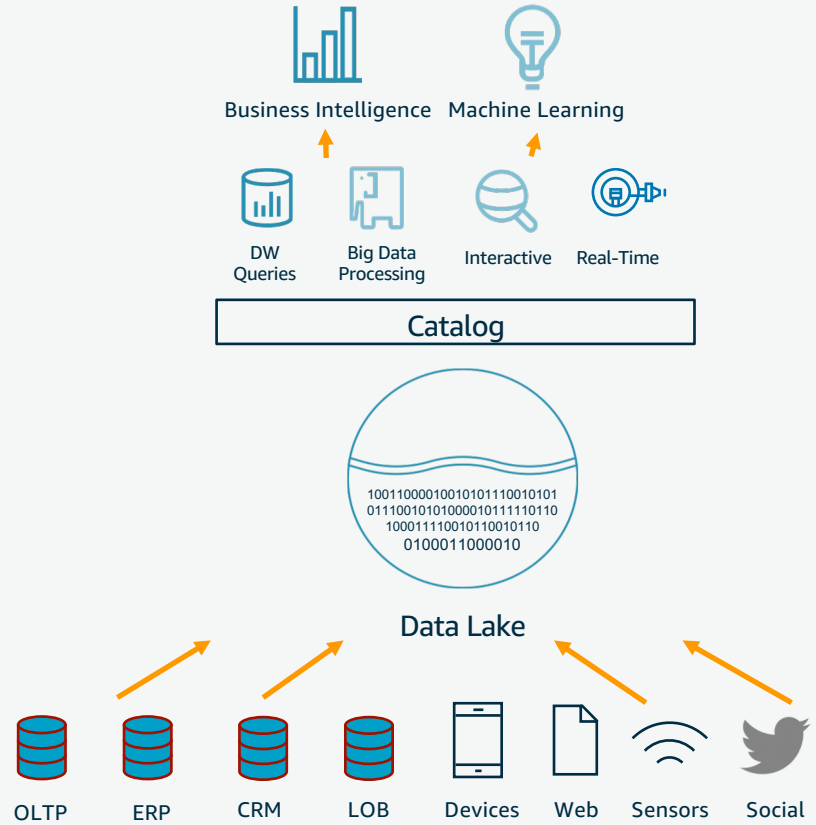
Schema Defined Prior to Data Load

Operational and Ad Hoc Reporting

Large Initial Capex + $$K / TB/ Year

aws

Data Lakes Extend the Traditional Approach

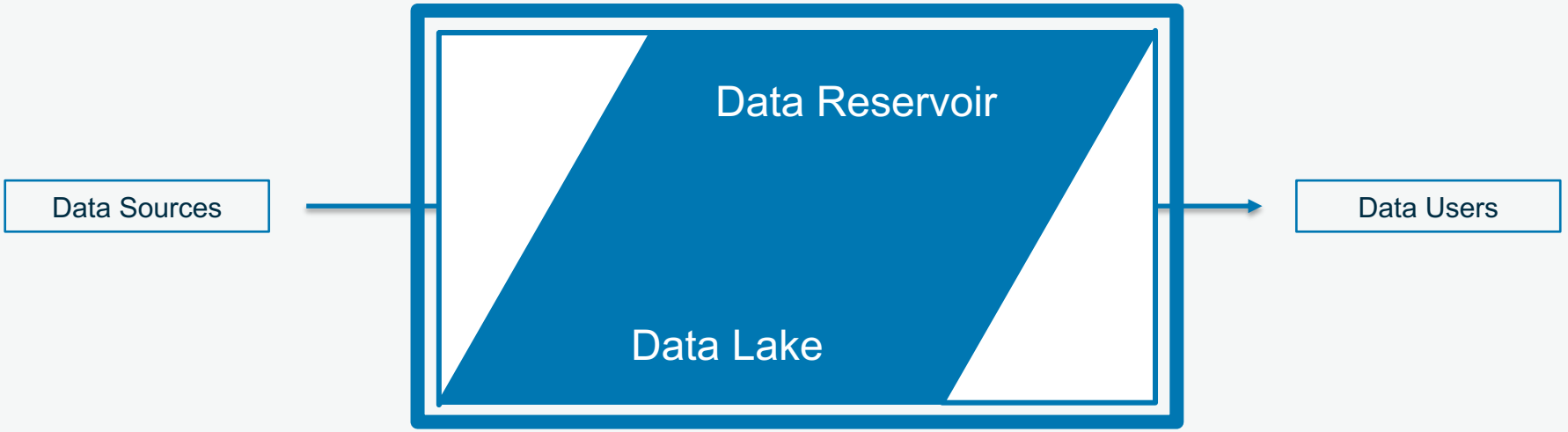


- TB-PB-EBs Scale
- All Data in one place, a Single Source of Truth
- Relational and Non-Relational Data
- Decouples (low cost) Storage and Compute
- Schema on Read
- Diverse Analytical Engines

Data Lakes and Data Reservoirs

Ingestion Triangle:
More work at the top.
Less work at the bottom.

Data Warehouse is a
kind of Data Reservoir



Data Lakes are optimized for large volumes of storage and ease of ingestion of any kind or flavor of data.

Data Warehouses are optimized for ease of retrieval of more critical and useful data.

Analytics Triangle:
Less work at the top.
More work at the bottom.

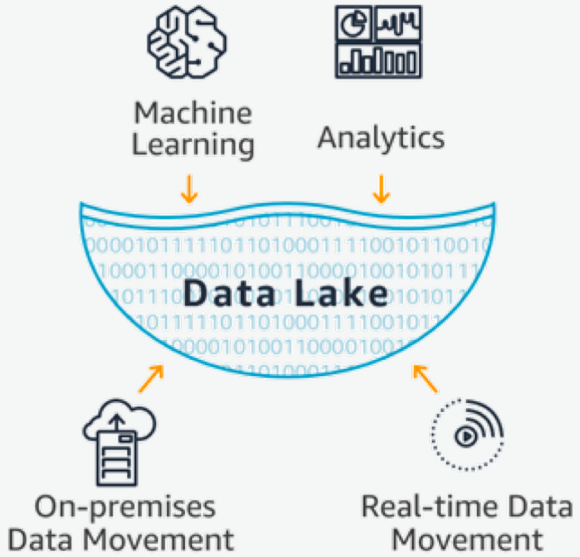
More info:
https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/

aws

Why Data Lakes?

Why a Data Lake?

- Decouple storage from compute, allowing you to scale
- Enable advanced analytics across all of your data sources
- Reduce complexity in ETL and operational overhead
- Future extensibility as new database and analytics technologies are invented



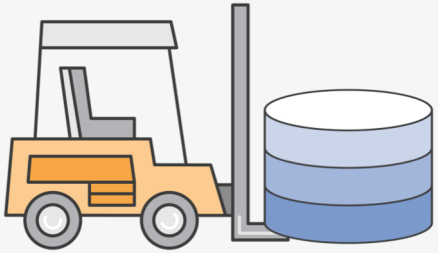
aws

Benefits of a Data Lake – All Data in One Place

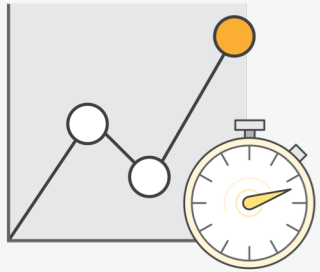
“Why is the data distributed in many locations? Where is the single source of truth ?”

Store and analyze all of your data, from all of your sources, in one centralized location.

Benefits of a Data Lake – Quick Ingest

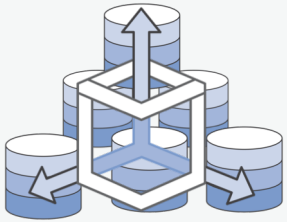


"How can I collect data quickly from various sources and store it efficiently?"

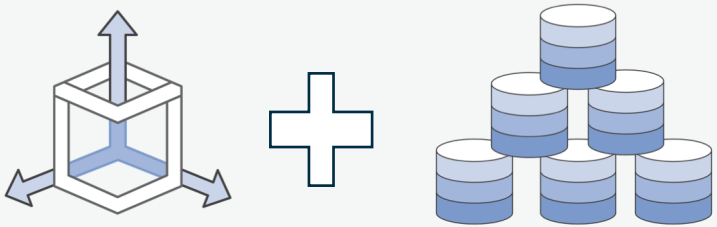


Quickly ingest data without needing to force it into a pre-defined schema.

Benefits of a Data Lake – Storage vs Compute



"How can I scale up with the volume of data being generated?"



Separating your storage and compute allows you to scale each component as required

aws

Benefits of a Data Lake – Schema on Read

“Is there a way I can apply multiple analytics and processing frameworks to the same data?”

A Data Lake enables ad-hoc analysis by applying schemas on read, not write.

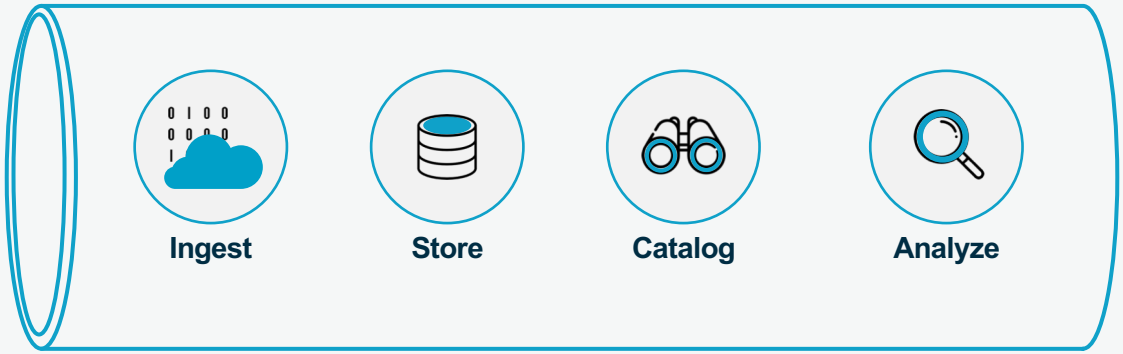
aws

Building a Data Lake on AWS

More data lakes & analytics on AWS than anywhere else

Realizing customer and operational insight from your data requires a robust Data Pipeline

Customer and Operational **Data**



Customer and Operational **Insights**

Data Lake Reference Architecture

Ingest

Kinesis
streaming data

Direct Connect
link to data center

Snowball
mail-in bulk data

Database Migration Service
import from Oracle, Netezza, and others

and many more…

Store

Amazon S3
secure, cost-effective storage

Catalog

Glue
data catalog & ETL

Analyze

Redshift
data warehouse

Athena
Ad-hoc analysis

EMR
unstructured data

QuickSight
visualization

Sagemaker
AI/Machine Learning

and many more…

amazon.com

Amazon.com's vision is to be the earth's most customer—centric company; where people can find anything they want to buy online.

Challenge:

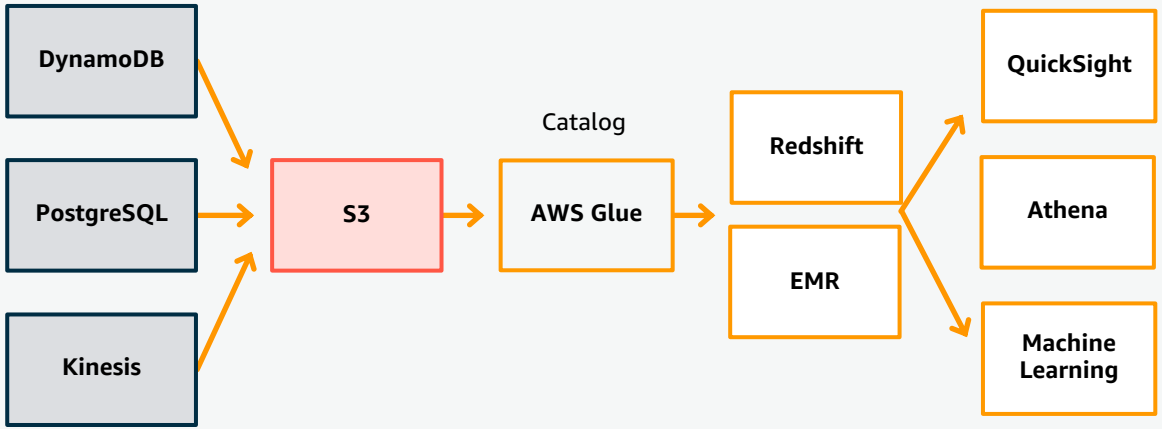
Load 500K+ transactions each day, and serve 300K+ queries/extracts each day from Amazon businsses (Amazon.com, Amazon Prime, Amazon Music, Amazon Alexa, Amazon Video, and Twitch).

Solution:

- Land data in S3 as a data lake
- Use Redshift as preferred SQL based analysis by business users, and EMR for machine learning

Amazon.com Uses AWS for Data Lakes & Analytics

amazon.com



- DynamoDB capturing all Amazon.com transactions
- Everything from DynamoDB, RDS PostgreSQL and Kinesis fed to a S3 data lake
- Glue used to catalog the data
- Redshift used for all SQL-based queries, and EMR for all machine learning and big data processing
- End-users use QuickSight for visualizations

aws

21ST CENTURY FOX

Fox Film Entertainment is one of the largest movie studios in the world.

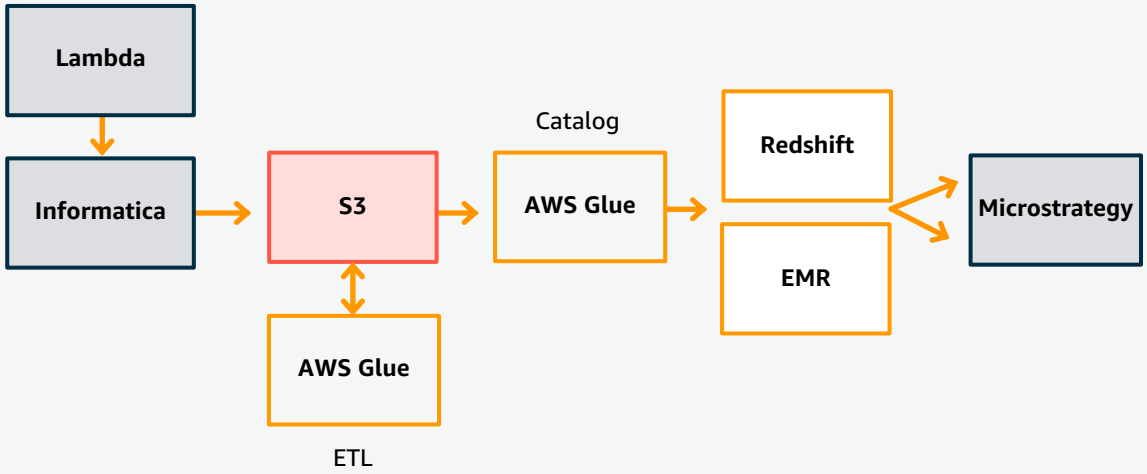
Challenge:

Processes 100+ of TB of data a day, 25k+ queries per day. They had a legacy data platform that struggled to scale, faced many outages, and had changing requirements from consumers who had many options.

Solution:

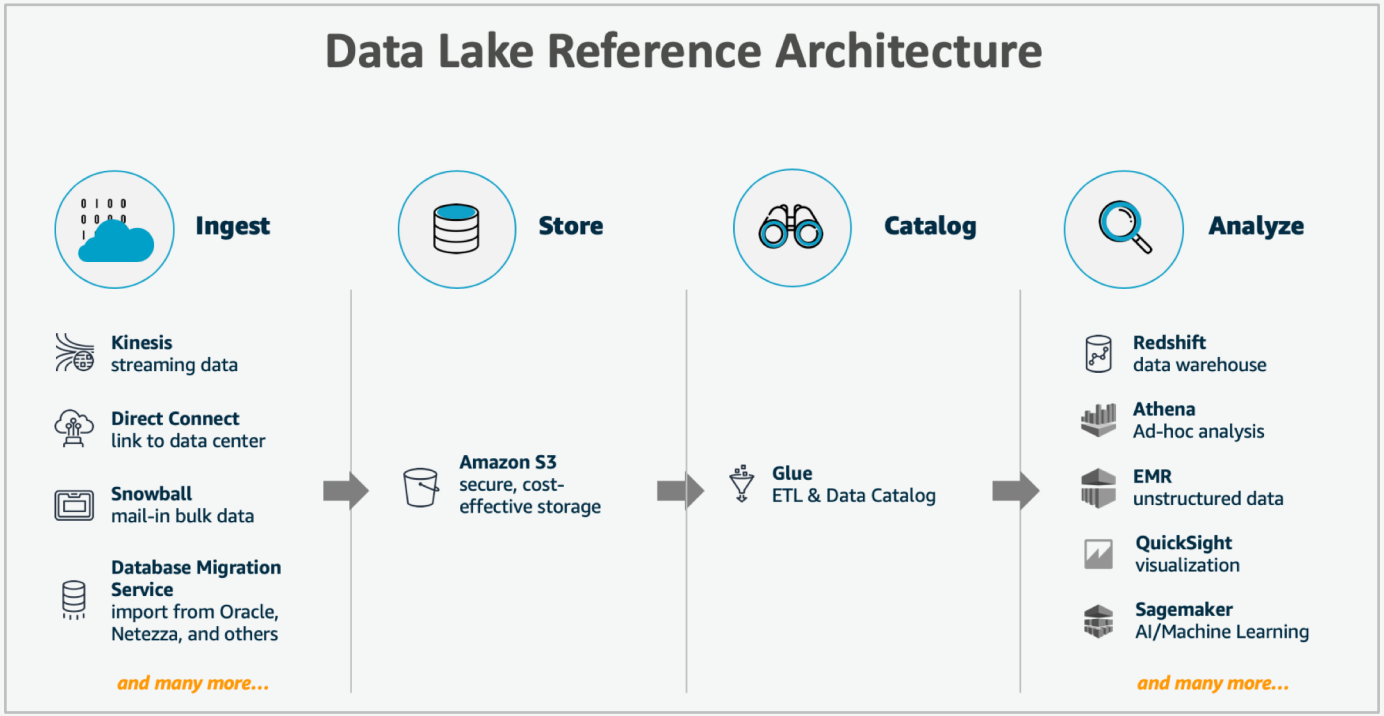
- Land data in S3 as a data lake
- Use Redshift and EMR as analytics engines to process data
- Saved 15–20% in overall costs (on-premises) with 30% of performance gain

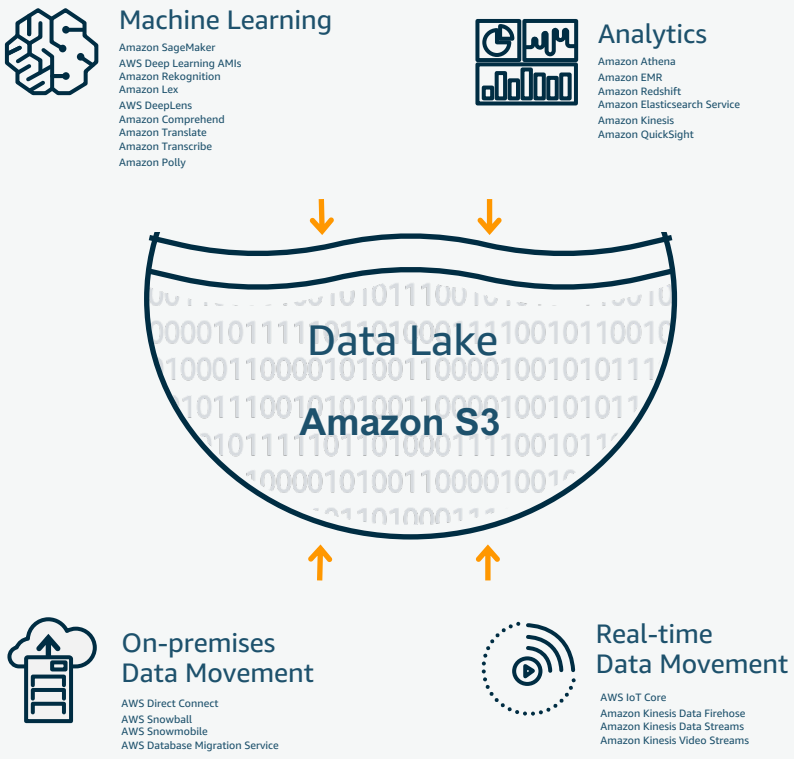
21st Century Fox Uses AWS for Data Lakes & Analytics



- Collect and ingest data with AWS Lambda for serverless scale and Informatica for data ingestion and transformation
- AWS Glue does ETL on data in S3 and provides a data catalog
- Redshift and EMR used as analytics engines
- Microstrategy used as a visualization tool

Amazon S3





Data Lakes start with Amazon S3

Why Amazon S3 for a Data Lake?

Durable

Designed for **11 9s** of durability

Available

Designed for **99.99**% availability

High performance

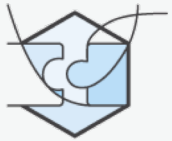
- Multiple upload
- Range GET

Easy to use

- Simple REST API
- AWS SDKs
- Read-after-create consistency
- Event notification
- Lifecycle policies

Scalable

- Store as much as you need
- Scale storage and compute independently
- No minimum usage commitments



Integrated

- Amazon EMR
- Amazon Redshift
- Amazon Athena
- AWS Glue
- Amazon DynamoDB
- More…

aws

Lab Time

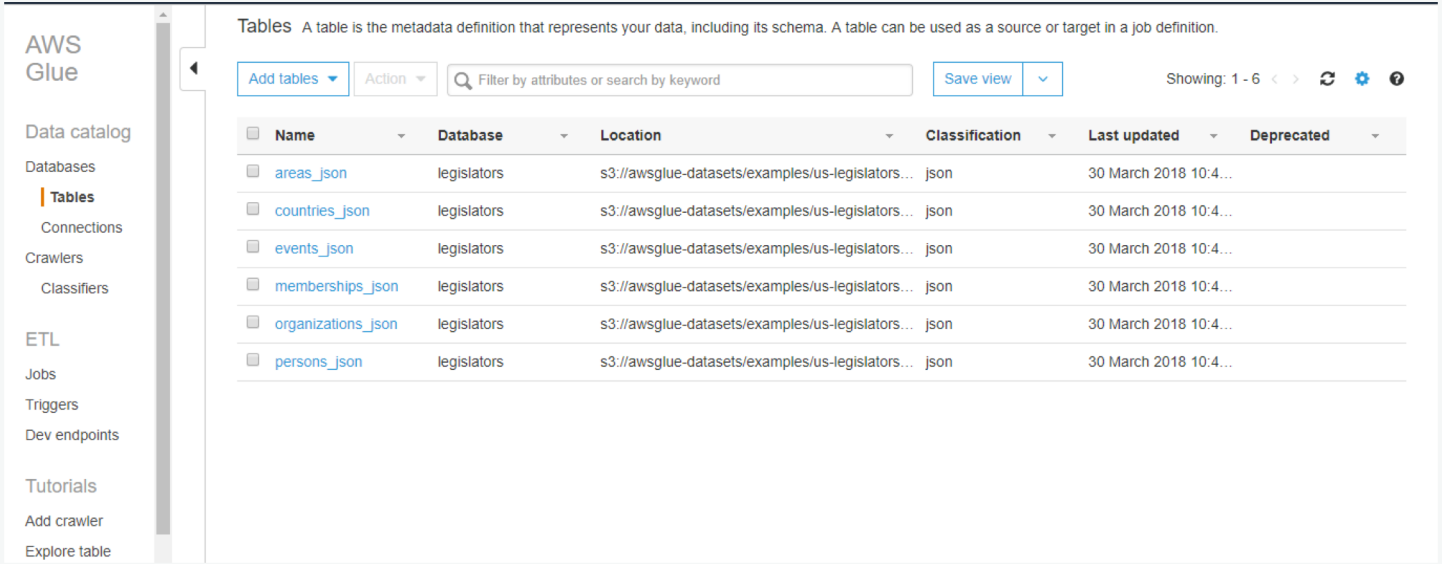
https://github.com/dbayardAWS/intro-data-lake

Please start Lab1.

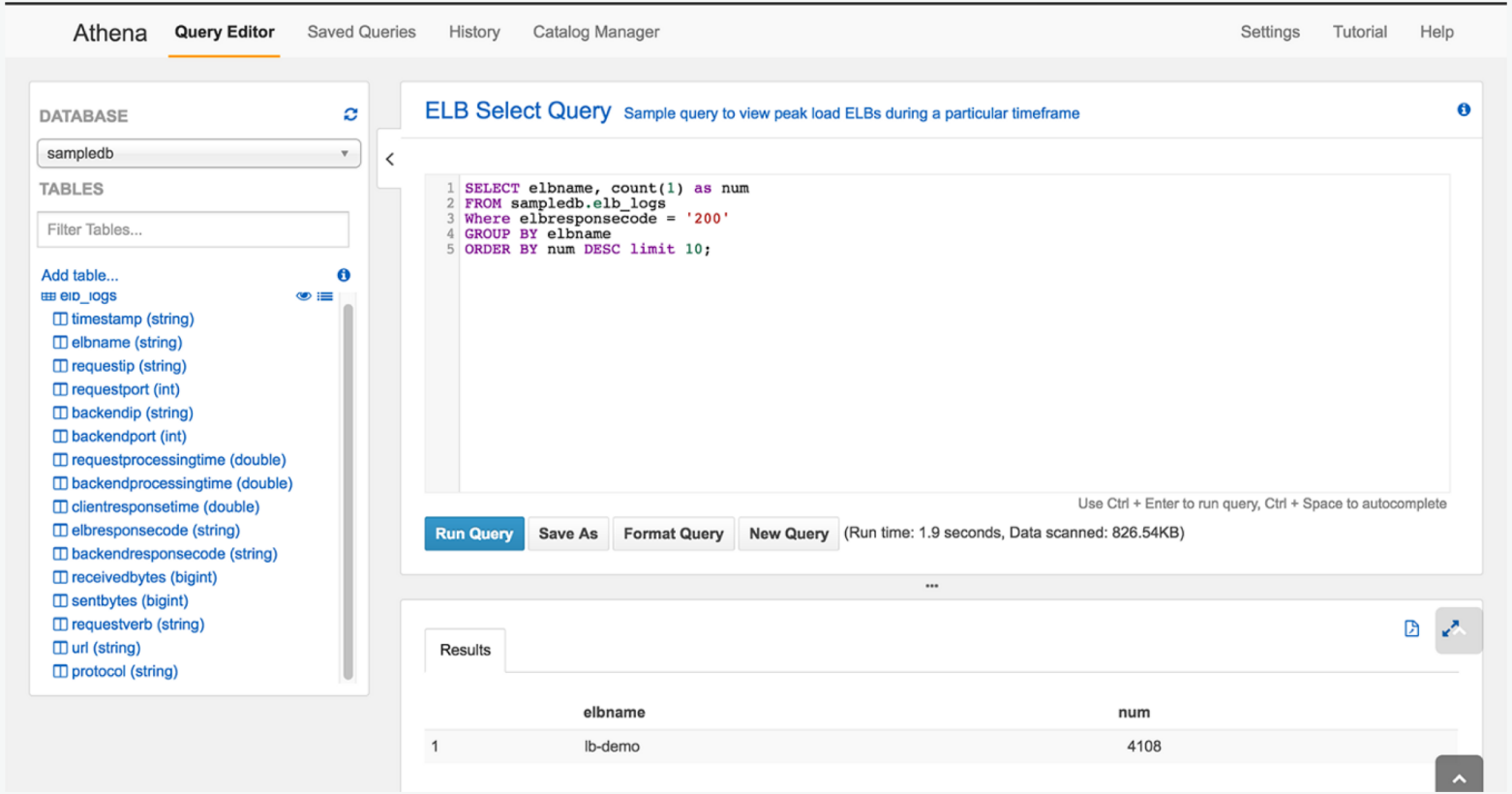
Continue thru the **Upload the sample dataset to the Data Lake** step.

What can you do with a Data Lake?

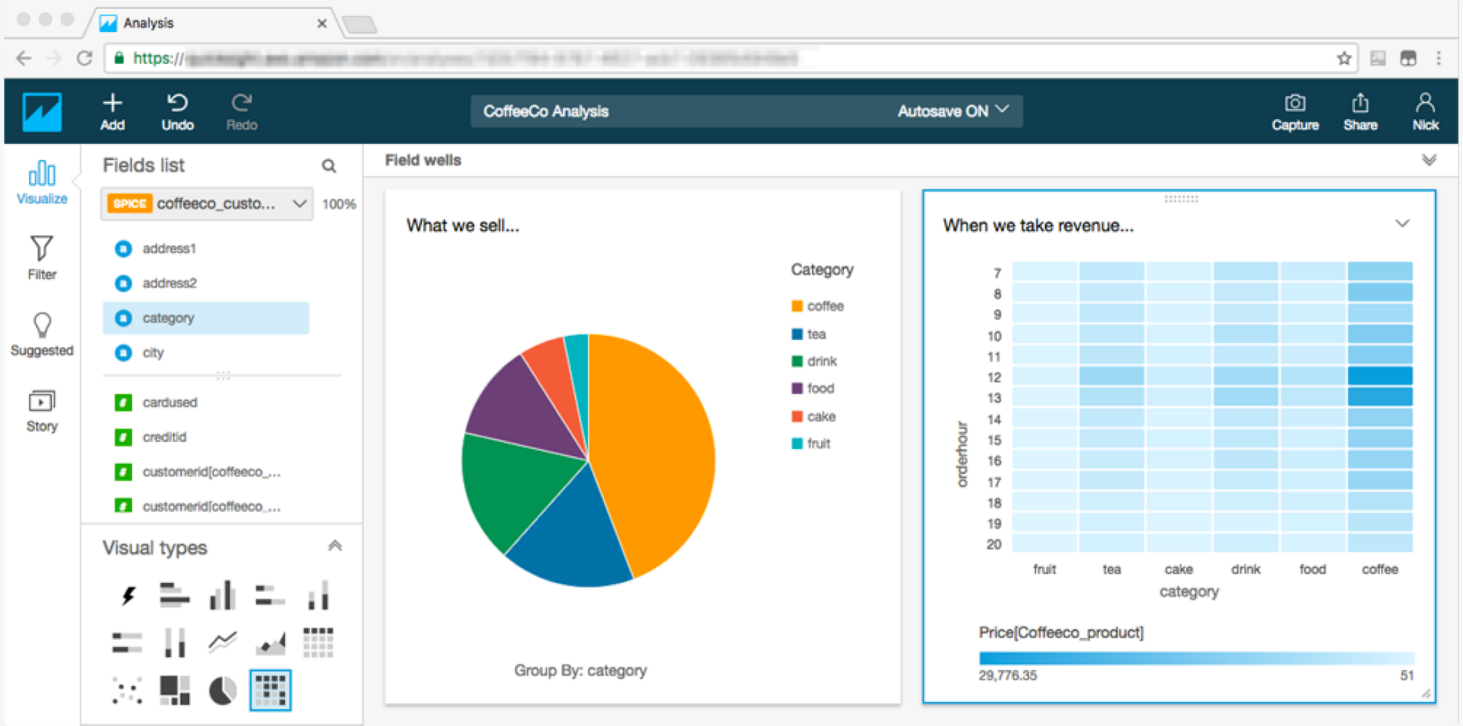
Create a Central Data Catalog with AWS Glue



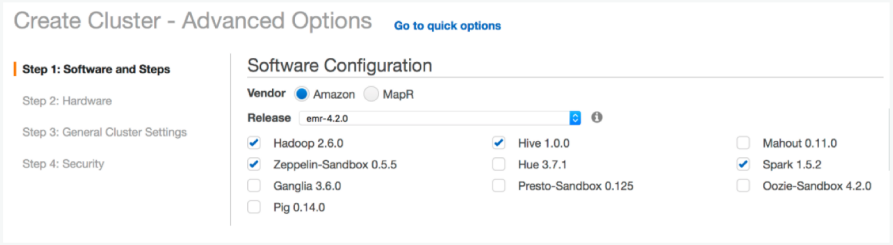
Query Directly with Amazon Athena & Amazon Redshift



Create Visualizations with Amazon QuickSight



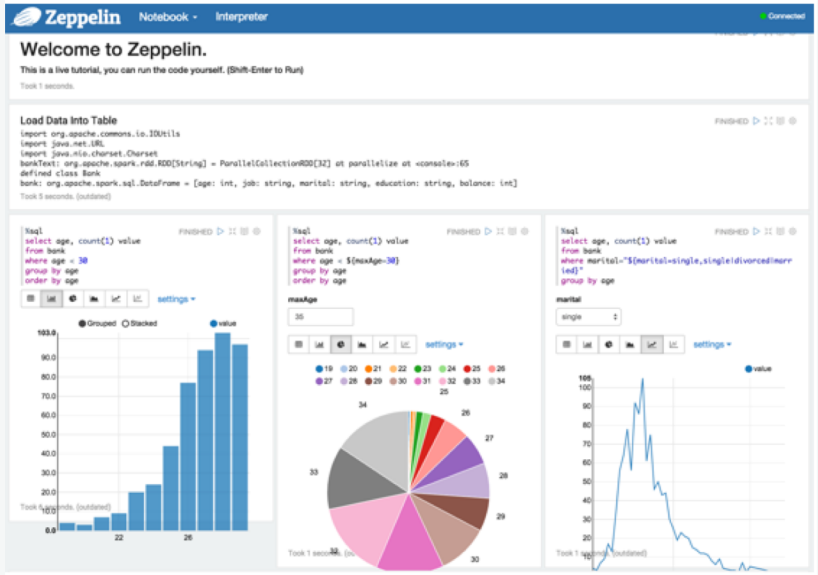
Analyze with Hadoop on Amazon EMR



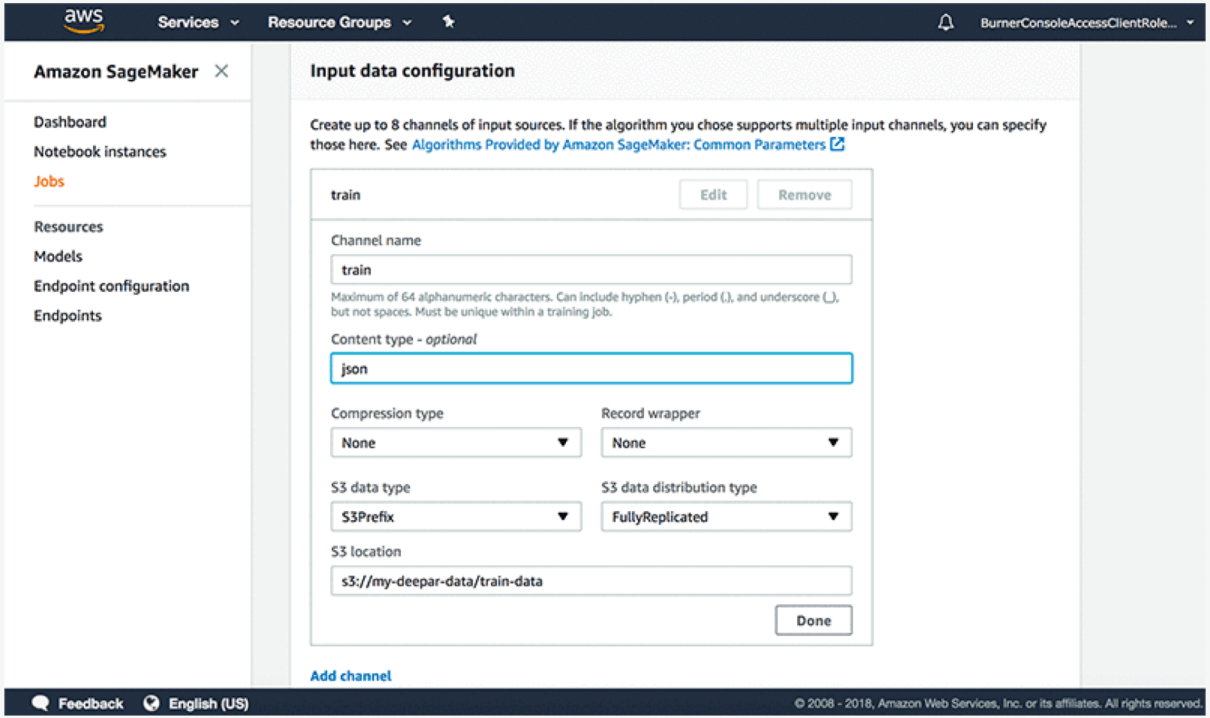

```scala
val movieLensHomeDir = "s3://emr.examples/movieLens/"

val movies = sc.textFile(movieLensHomeDir + "movies.dat").map { line =>
  val fields = line.split("::")
  // format: (movieId, movieName)
  (fields(0).toInt, fields(1))
}.collect.toMap

val ratings = sc.textFile(movieLensHomeDir + "ratings.dat").map { line =>
  val fields = line.split("::")
  // format: (timestamp % 10, Rating(userId, movieId, rating))
  (fields(3).toLong % 10, Rating(fields(0).toInt, fields(1).toInt, fields(2).toDouble))
}
```



Train ML Models with Amazon SageMaker



Load into Downstream Services

Amazon Redshift

Run complex analytic queries against petabytes of structured data

Amazon Aurora

A MySQL and PostgreSQL compatible relational database built for the cloud

Amazon DynamoDB

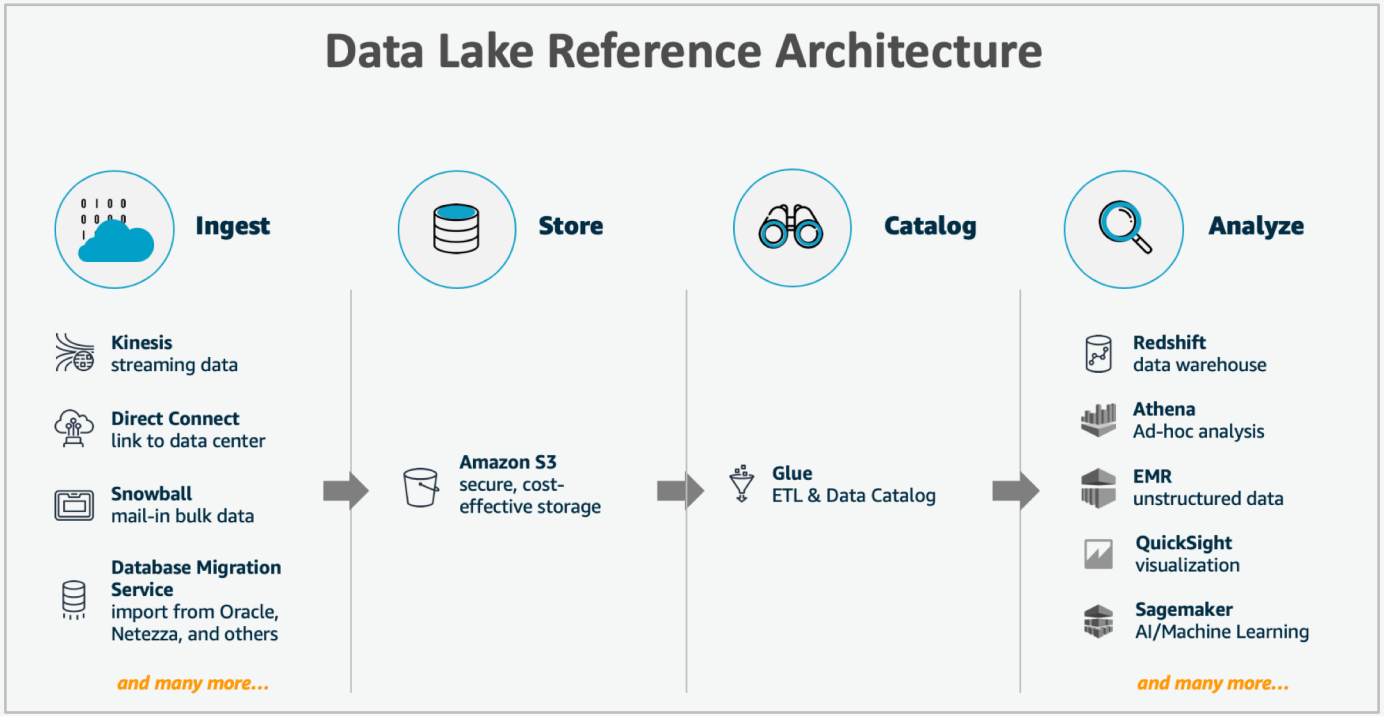
A NoSQL database service that delivers consistent, single-digit millisecond latency at any scale.

Amazon Elasticsearch

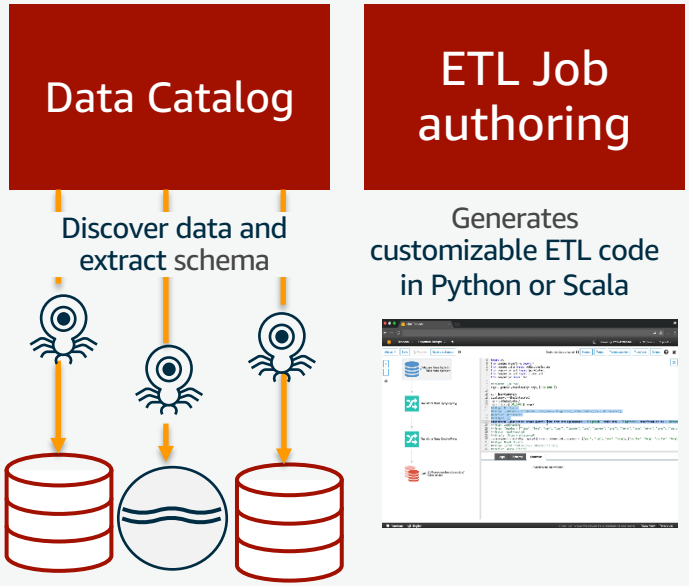
Delivers Elasticsearch's real-time analytics capabilities alongside the availability, scalability, and security that production workloads require.

aws

AWS Glue



AWS Glue: Data Catalog & ETL Service



- Automatically discovers data and stores schema
- Data is immediately searchable, and available to extract, transform, and load (ETL)
- Automatically generates customizable ETL code
- Schedules and runs your ETL jobs
- Serverless

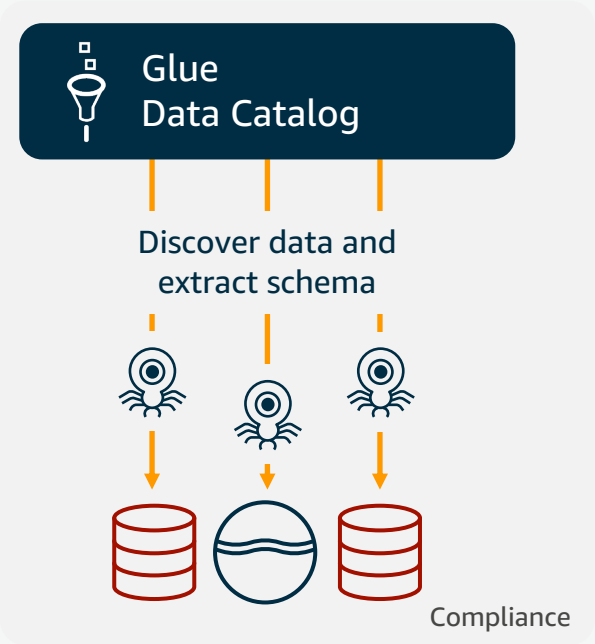
aws

Glue data catalog

Discover and organize your data sets

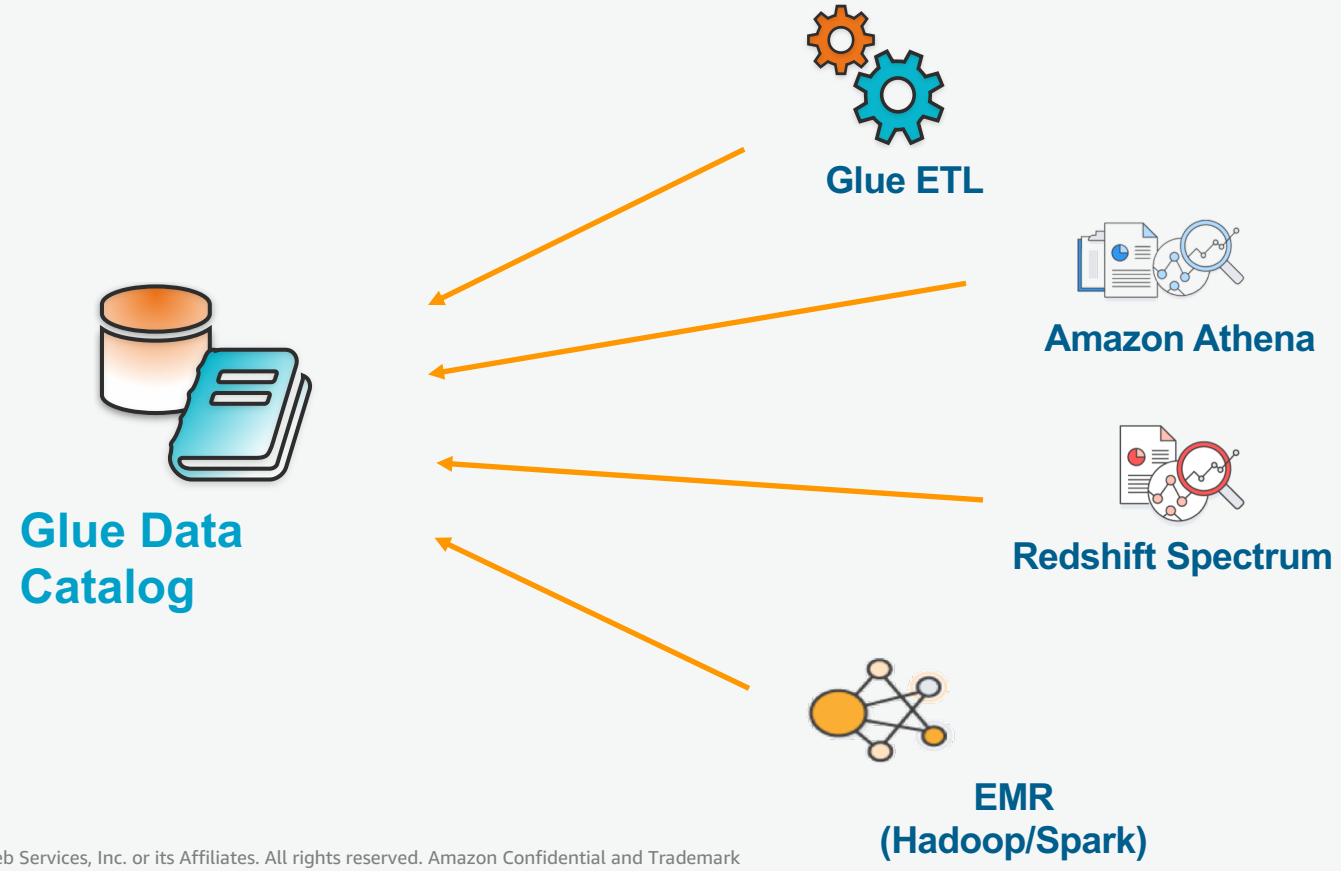
AWS Glue—Data Catalog

Make data discoverable

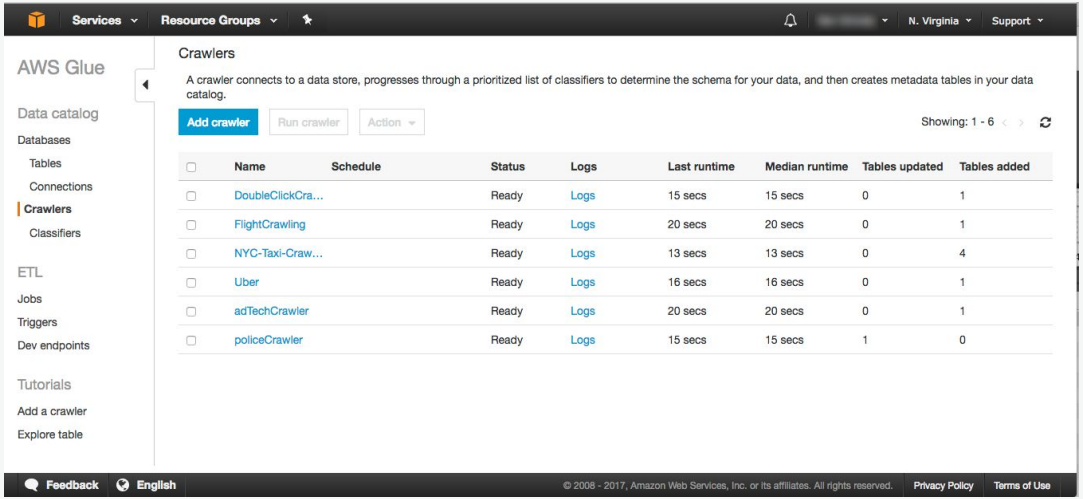


- Automatically discovers data and stores schema
- Catalog makes data searchable, and available for ETL
- Catalog contains table and job definitions
- Computes statistics to make queries efficient

Glue: Data Catalog – Queryable by Many Services



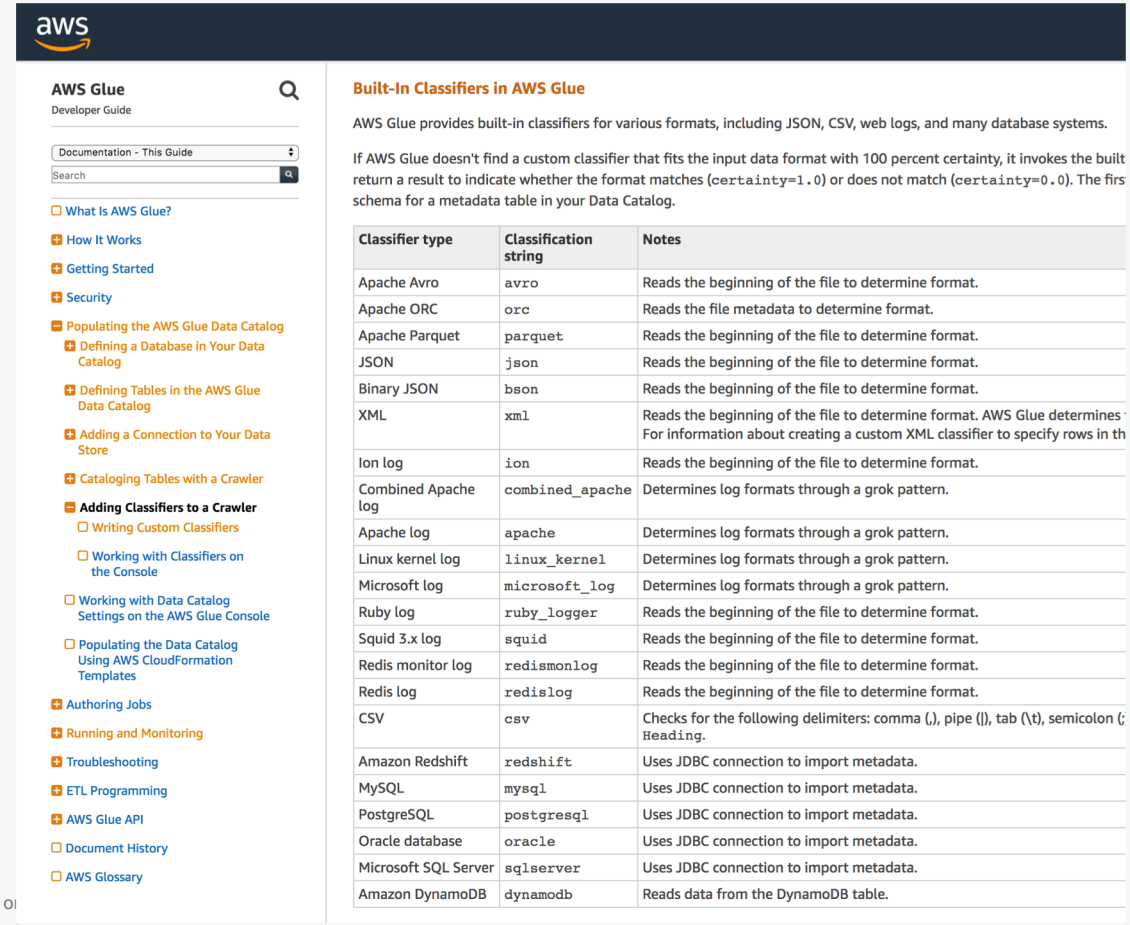
Glue: Data Catalog - Crawlers



Features Include:

- Built-in classifiers
  - **Detect file type**
  - **Extract schema**
  - **Identify partitions**
- **On-Demand** or **Scheduled** Execution
- Build-you-own classifiers
  - **Grok for ease of use**

Built-in Classifiers



aws

AWS Glue
Developer Guide

Documentation - This Guide
Search

- What Is AWS Glue?
- How It Works
- Getting Started
- Security
- Populating the AWS Glue Data Catalog
  - Defining a Database in Your Data Catalog
  - Defining Tables in the AWS Glue Data Catalog
  - Adding a Connection to Your Data Store
  - Cataloging Tables with a Crawler
  - **Adding Classifiers to a Crawler**
    - Writing Custom Classifiers
    - Working with Classifiers on the Console
  - Working with Data Catalog Settings on the AWS Glue Console
  - Populating the Data Catalog Using AWS CloudFormation Templates
- Authoring Jobs
- Running and Monitoring
- Troubleshooting
- ETL Programming
- AWS Glue API
- Document History
- AWS Glossary

Built-In Classifiers in AWS Glue

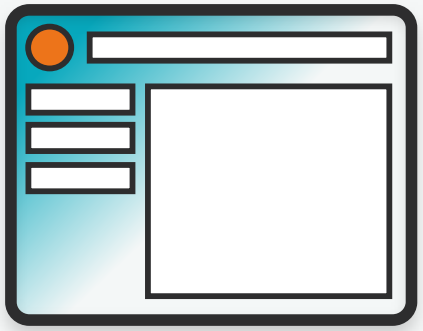
AWS Glue provides built-in classifiers for various formats, including JSON, CSV, web logs, and many database systems.

If AWS Glue doesn't find a custom classifier that fits the input data format with 100 percent certainty, it invokes the built return a result to indicate whether the format matches (`certainty=1.0`) or does not match (`certainty=0.0`). The firs schema for a metadata table in your Data Catalog.

Classifier type	Classification string	Notes
Apache Avro	`avro`	Reads the beginning of the file to determine format.
Apache ORC	`orc`	Reads the file metadata to determine format.
Apache Parquet	`parquet`	Reads the beginning of the file to determine format.
JSON	`json`	Reads the beginning of the file to determine format.
Binary JSON	`bson`	Reads the beginning of the file to determine format.
XML	`xml`	Reads the beginning of the file to determine format. AWS Glue determines For information about creating a custom XML classifier to specify rows in th
Ion log	`ion`	Reads the beginning of the file to determine format.
Combined Apache log	`combined_apache`	Determines log formats through a grok pattern.
Apache log	`apache`	Determines log formats through a grok pattern.
Linux kernel log	`linux_kernel`	Determines log formats through a grok pattern.
Microsoft log	`microsoft_log`	Determines log formats through a grok pattern.
Ruby log	`ruby_logger`	Reads the beginning of the file to determine format.
Squid 3.x log	`squid`	Reads the beginning of the file to determine format.
Redis monitor log	`redismonlog`	Reads the beginning of the file to determine format.
Redis log	`redislog`	Reads the beginning of the file to determine format.
CSV	`csv`	Checks for the following delimiters: comma (,), pipe (|), tab (\t), semicolon (; `Heading`.
Amazon Redshift	`redshift`	Uses JDBC connection to import metadata.
MySQL	`mysql`	Uses JDBC connection to import metadata.
PostgreSQL	`postgresql`	Uses JDBC connection to import metadata.
Oracle database	`oracle`	Uses JDBC connection to import metadata.
Microsoft SQL Server	`sqlserver`	Uses JDBC connection to import metadata.
Amazon DynamoDB	`dynamodb`	Reads data from the DynamoDB table.

Ref: AWS Glue Developer Guide

aws

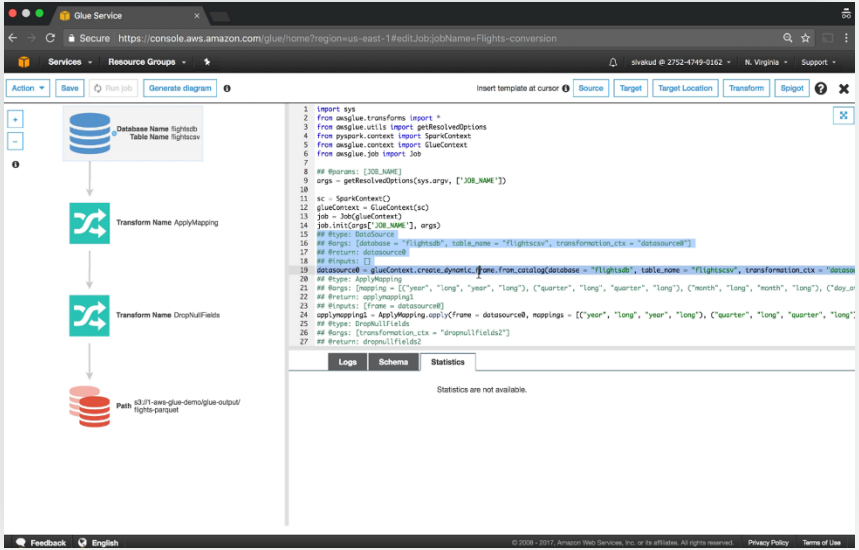


Glue ETL Jobs

Author and Deploy ETL Jobs Easily

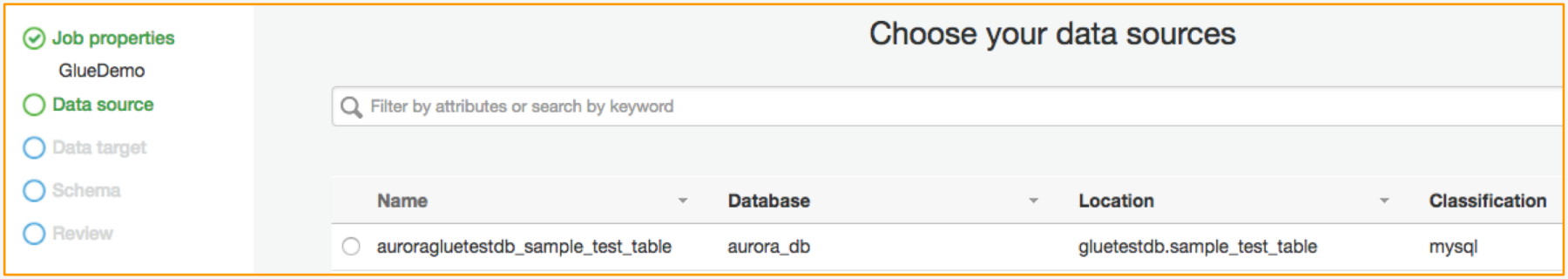
AWS Glue—ETL Service

Make ETL scripting and deployment easy



- Automatically generates ETL code
- Code is customizable with Python and Spark
- Endpoints provided to edit, debug, test code
- Jobs are scheduled or event-based
- Serverless

AWS Glue: Job Authoring - Pick a Source



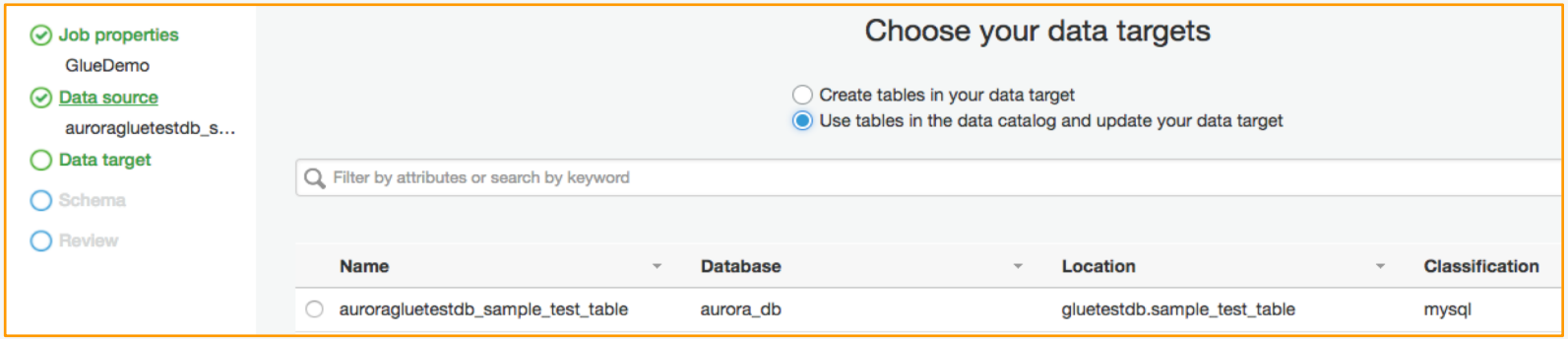
Choose from Manually-Defined or Crawler-Generated Sources:

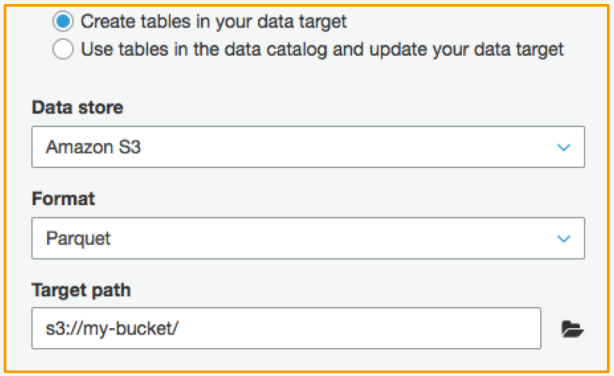
- S3 Bucket
- RDBMS
- Redshift
- DynamoDB

aws

AWS Glue: Job Authoring – Pick a Target

Write output results into an existing table.

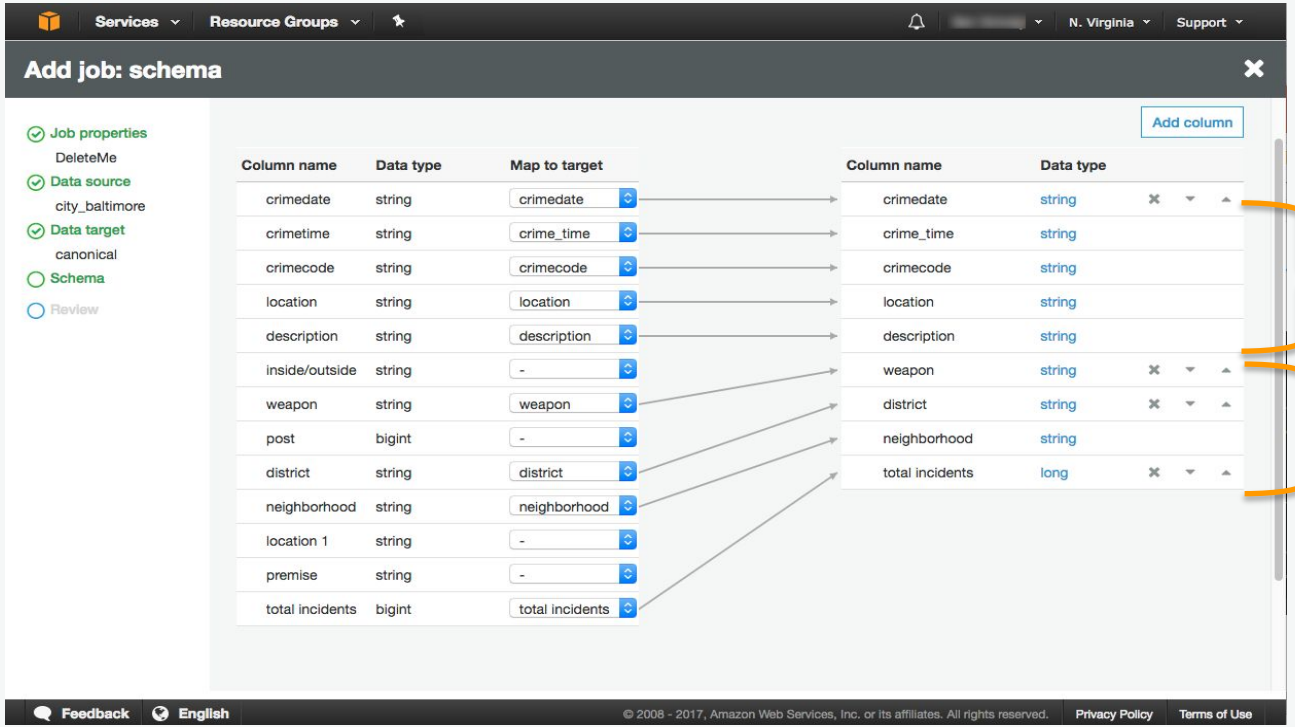




Crawler-Defined or Manually-Created:

- S3 Bucket
- RDBMS
- Redshift

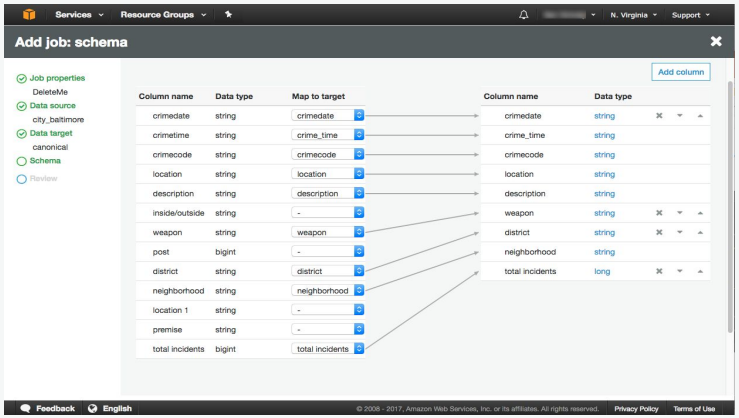
AWS Glue: Job Authoring – Code Generation

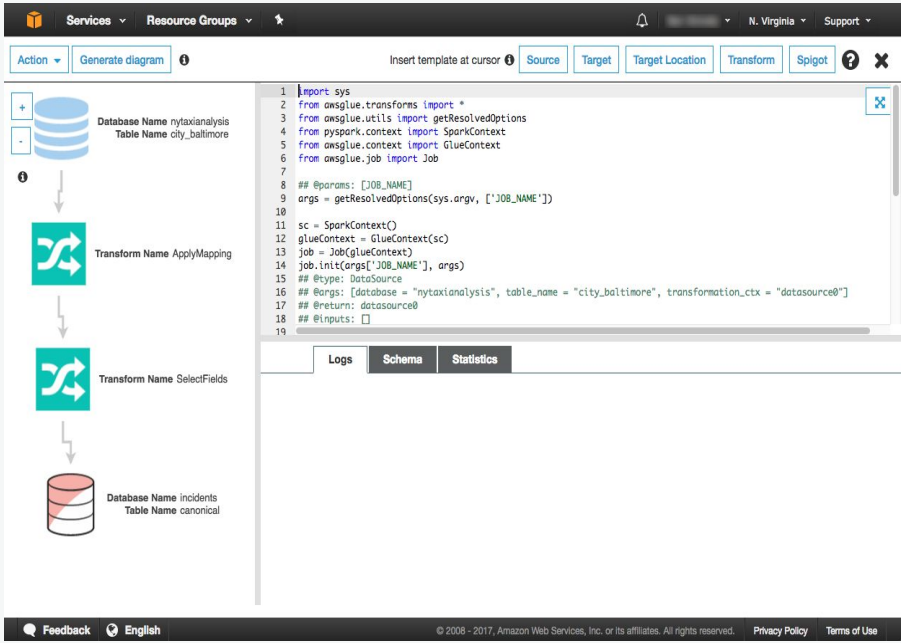


Existing columns in target

Can extend/add new columns to target

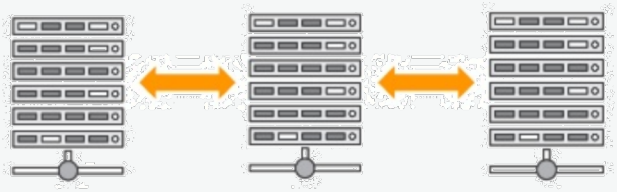
AWS Glue: Job Authoring – Code Generation

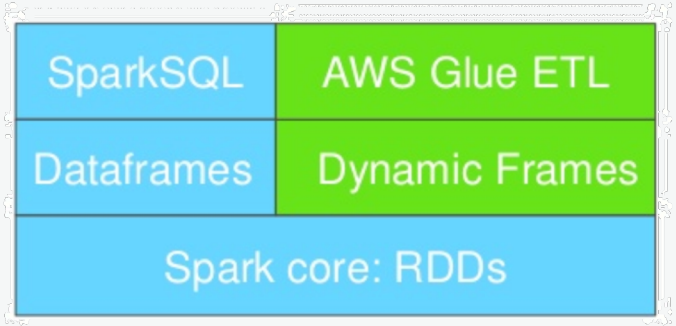




Glue generates transformation graph and either *Python* or *Scala* Spark code

Apache Spark & AWS Glue ETL





What is Apache Spark?

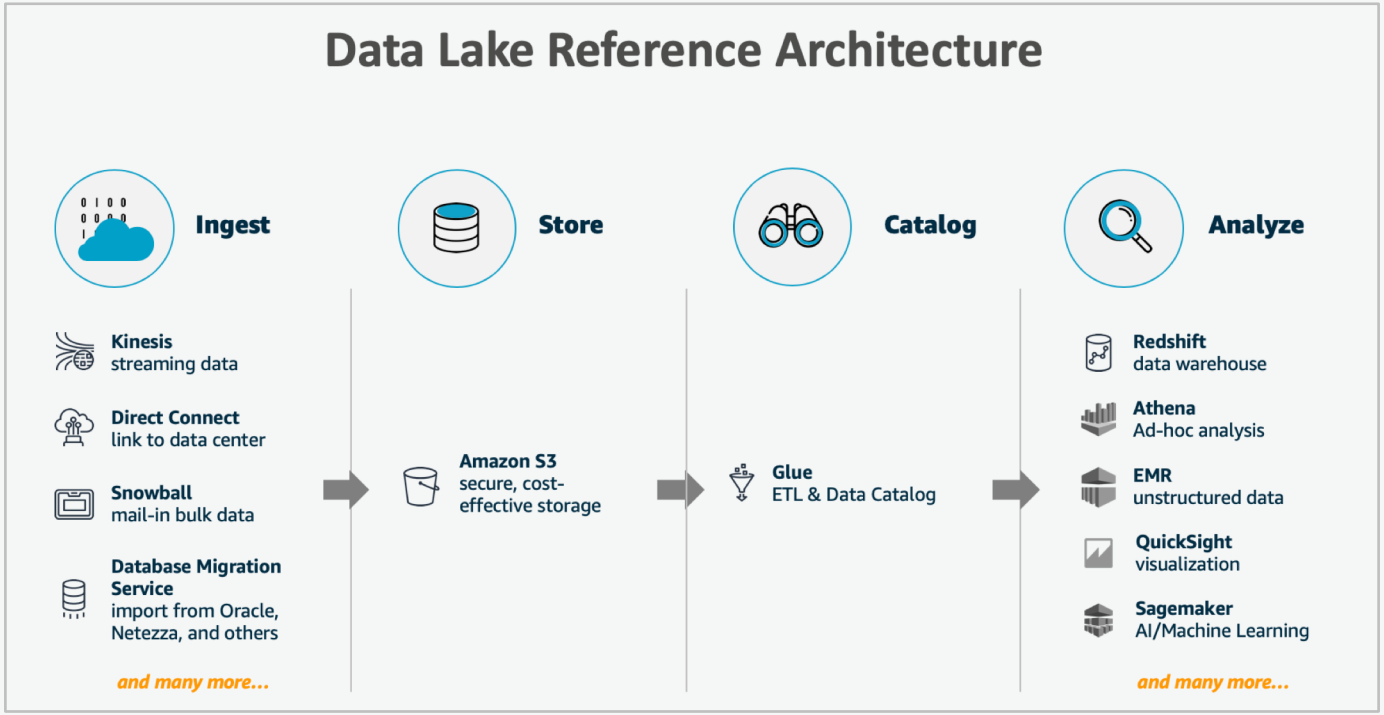
- Parallel, scale-out data processing engine
- Fault-tolerance built-in
- Flexible interface: Python scripting, SQL
- Rich eco-system: ML, Graph, analytics, …

AWS Glue ETL libraries

- Integration: Data Catalog, job orchestration, code-generation, job bookmarks, S3, RDS
- ETL transforms, more connectors & formats
- New data structure: Dynamic frames

aws

Amazon Athena



Amazon Athena—Interactive Analysis

Interactive query service to analyze data in Amazon S3 using standard SQL

No infrastructure to set up or manage and no data to load

Query Instantly

Zero setup cost; just point to S3 and start querying

Pay per query

Pay only for queries run; save 30–90% on per-query costs through compression

Open

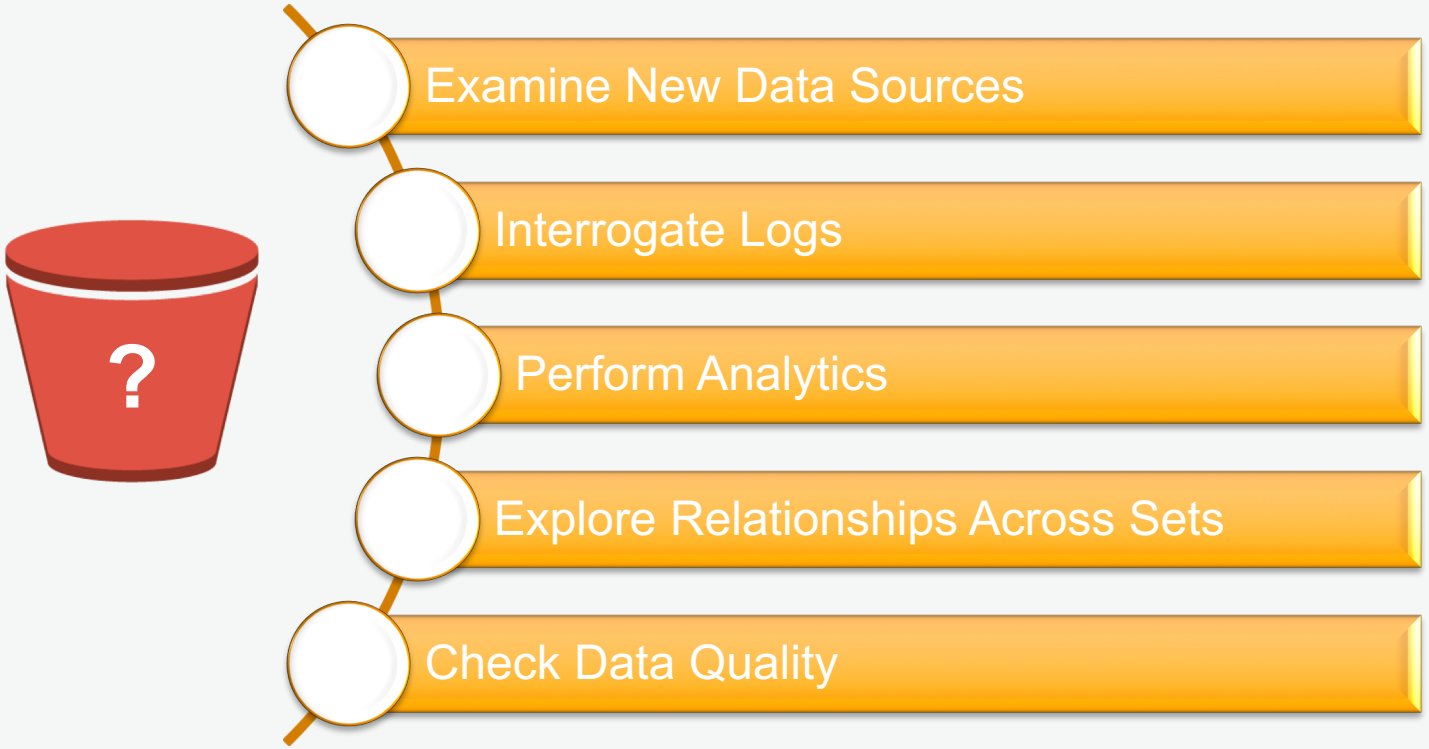
ANSI SQL interface, JDBC/ODBC drivers, multiple formats, compression types, and complex joins and data types

Easy

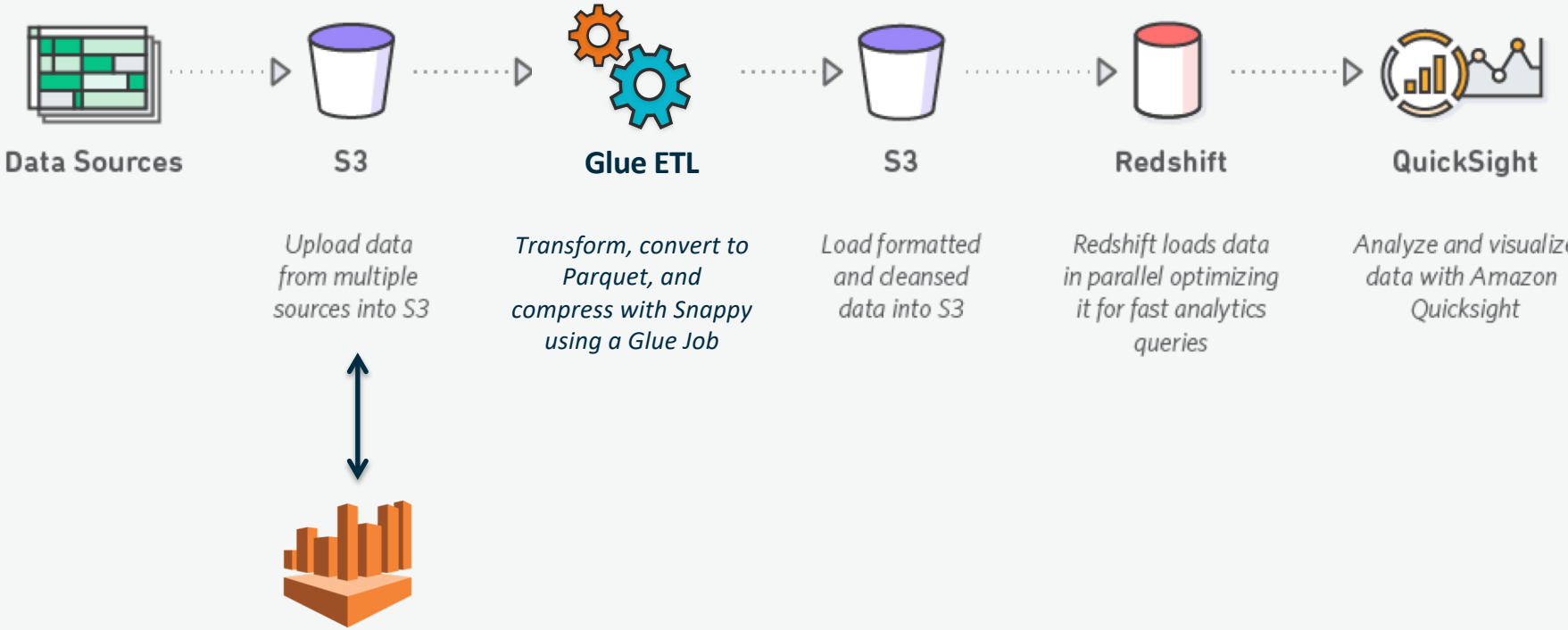
Serverless: zero infrastructure, zero administration Integrated with QuickSight

aws

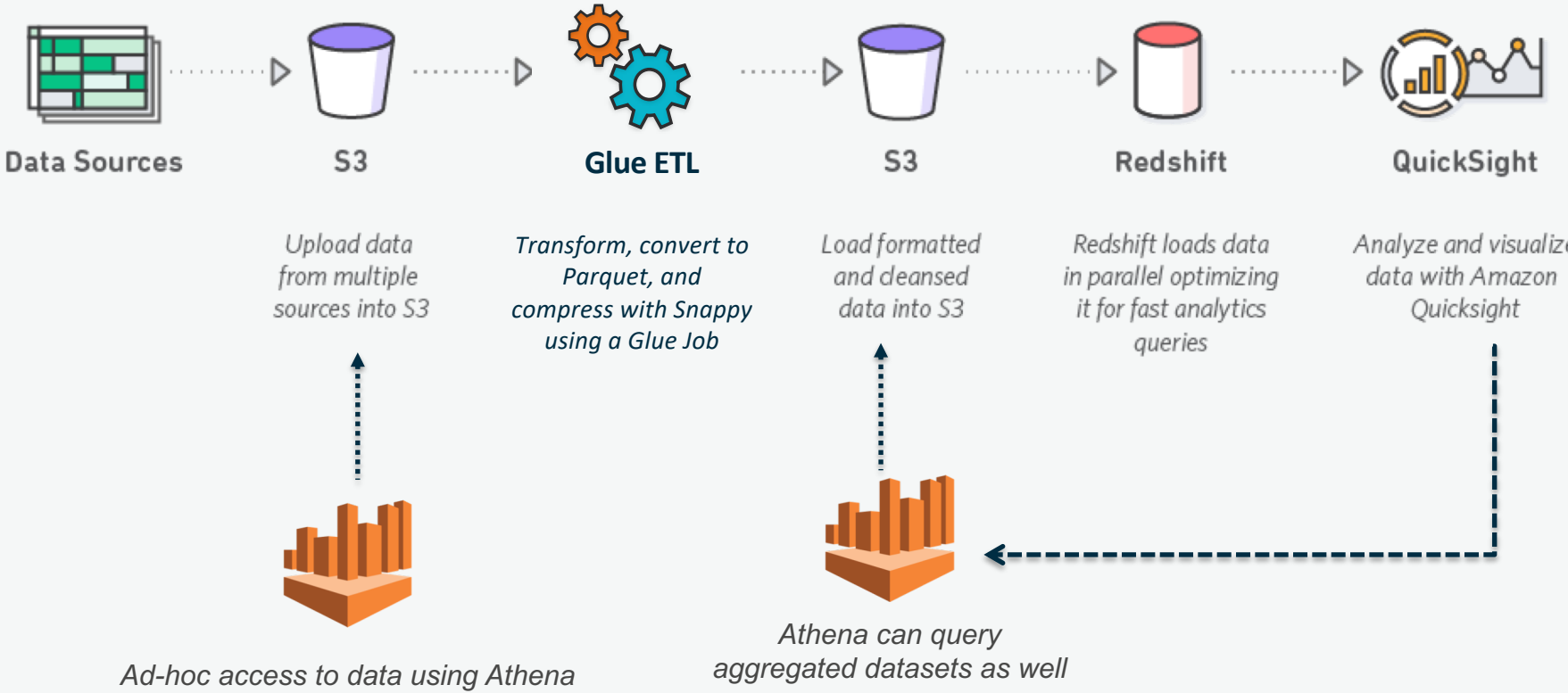
Examples of when you would query the Data Lake…



A Sample Pipeline

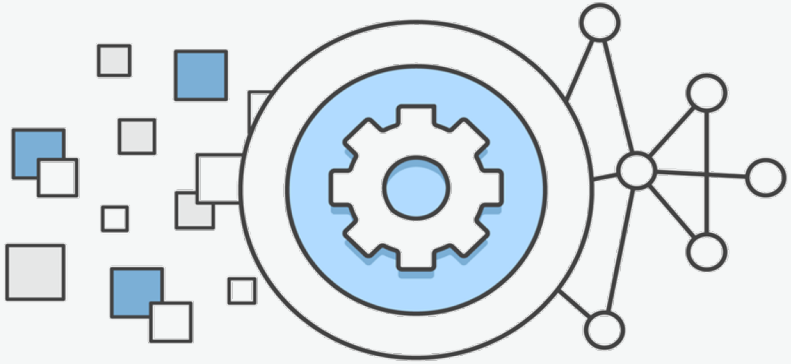


A Sample Pipeline



Athena is Serverless

- No Infrastructure or administration
- Zero Spin up time
- Transparent upgrades



Amazon Athena is Cost Effective

- Pay per query
- $5 per TB scanned from S3
- DDL Queries and failed queries are free
- Save by using compression, columnar formats, partitions

Recap: The core of a Data Lake in AWS

Versatile Compute Layers


Athena


Amazon EMR


AWS Glue ETL


Amazon Redshift

Data Lake

Data & Metadata


Amazon S3


AWS Glue Data Catalog

aws

Lab Time

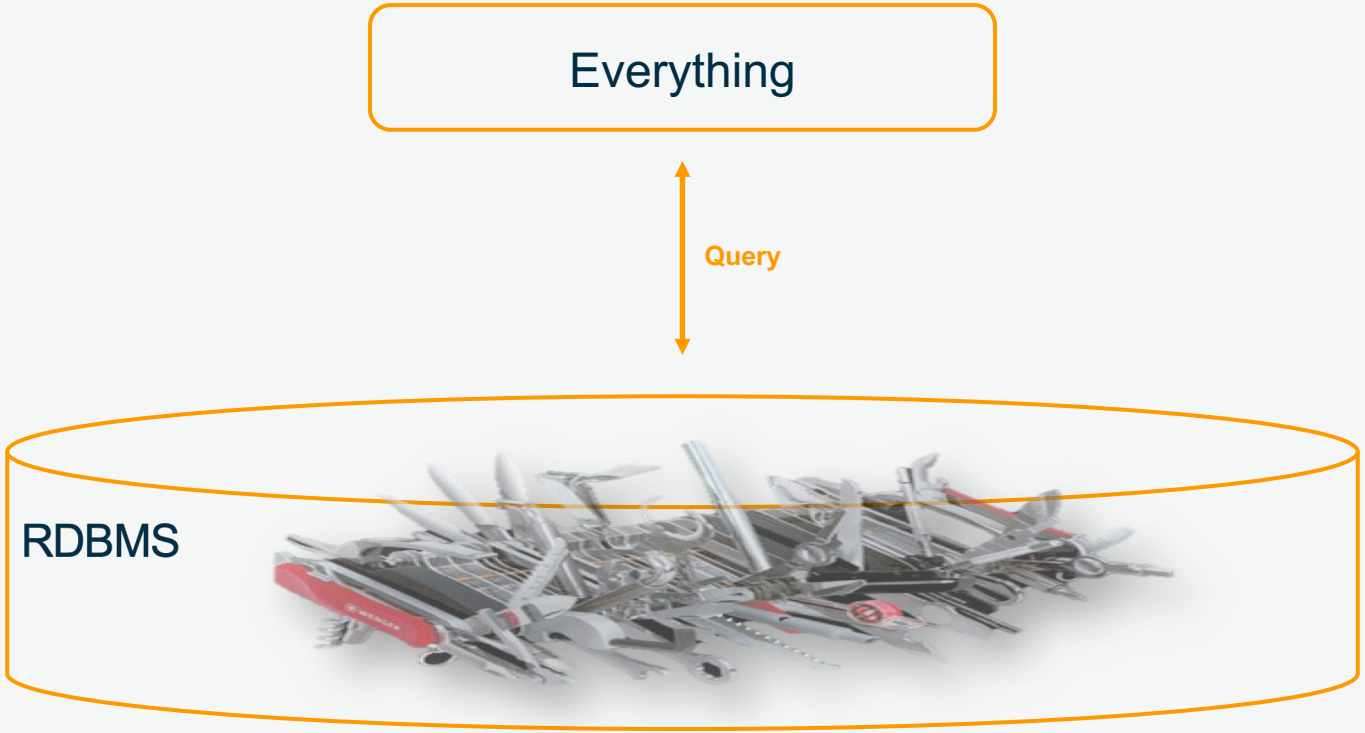
https://github.com/dbayardAWS/intro-data-lake

Please resume Lab1.
Your next step should be **Catalog our new dataset.**
Finish Lab1 (the last exercise is optional).

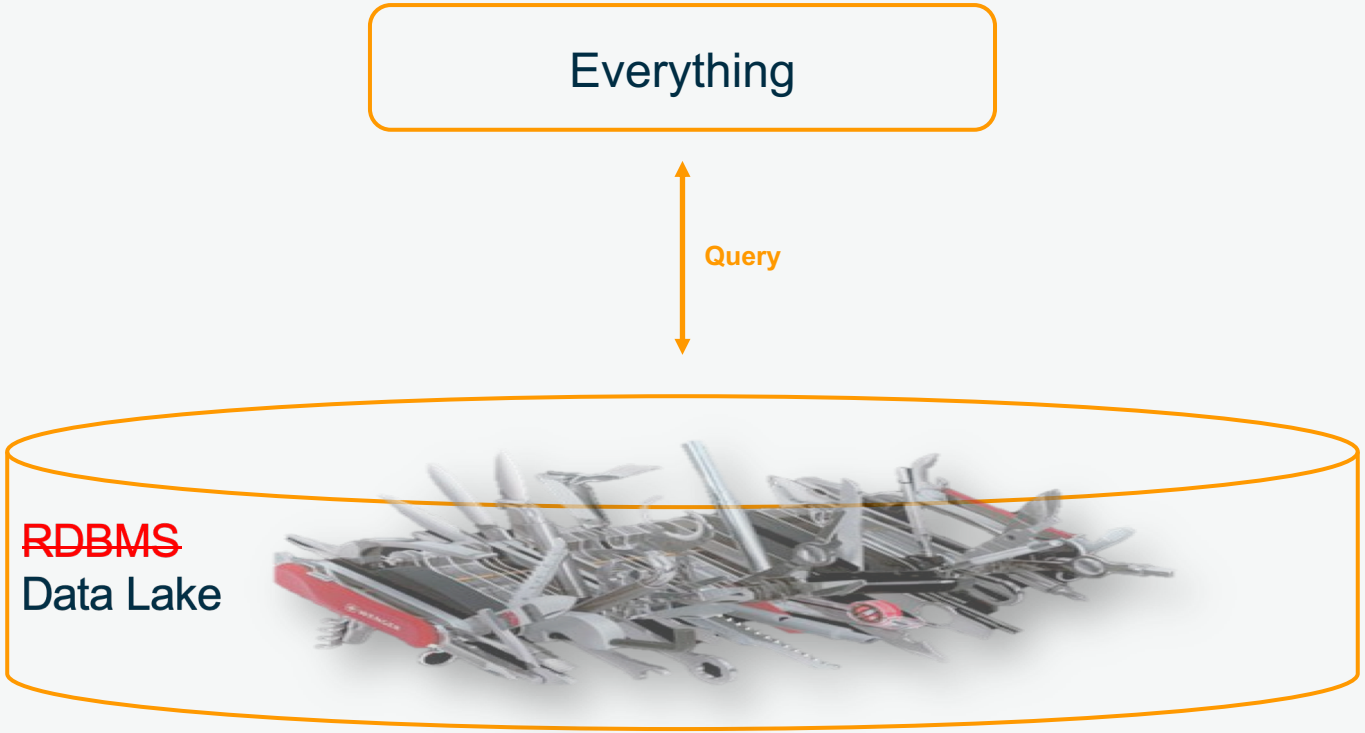
Cloud Data Warehouse

aws

Anti-Pattern



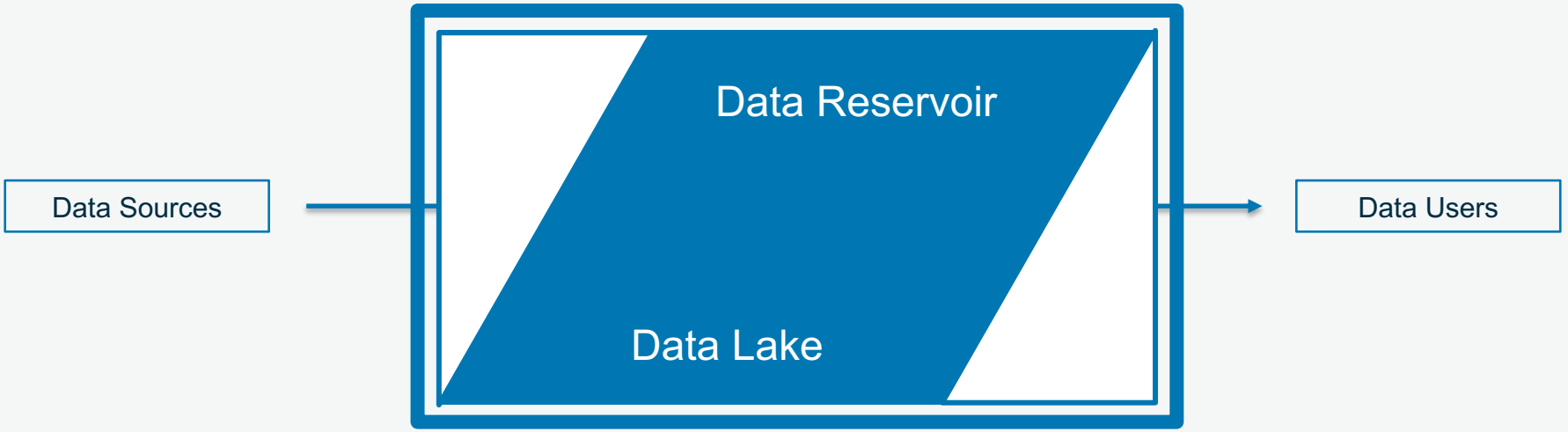
Also an Anti-Pattern



Data Lakes and Data Reservoirs

Ingestion Triangle:
More work at the top.
Less work at the bottom.

Data Warehouse is a
kind of Data Reservoir



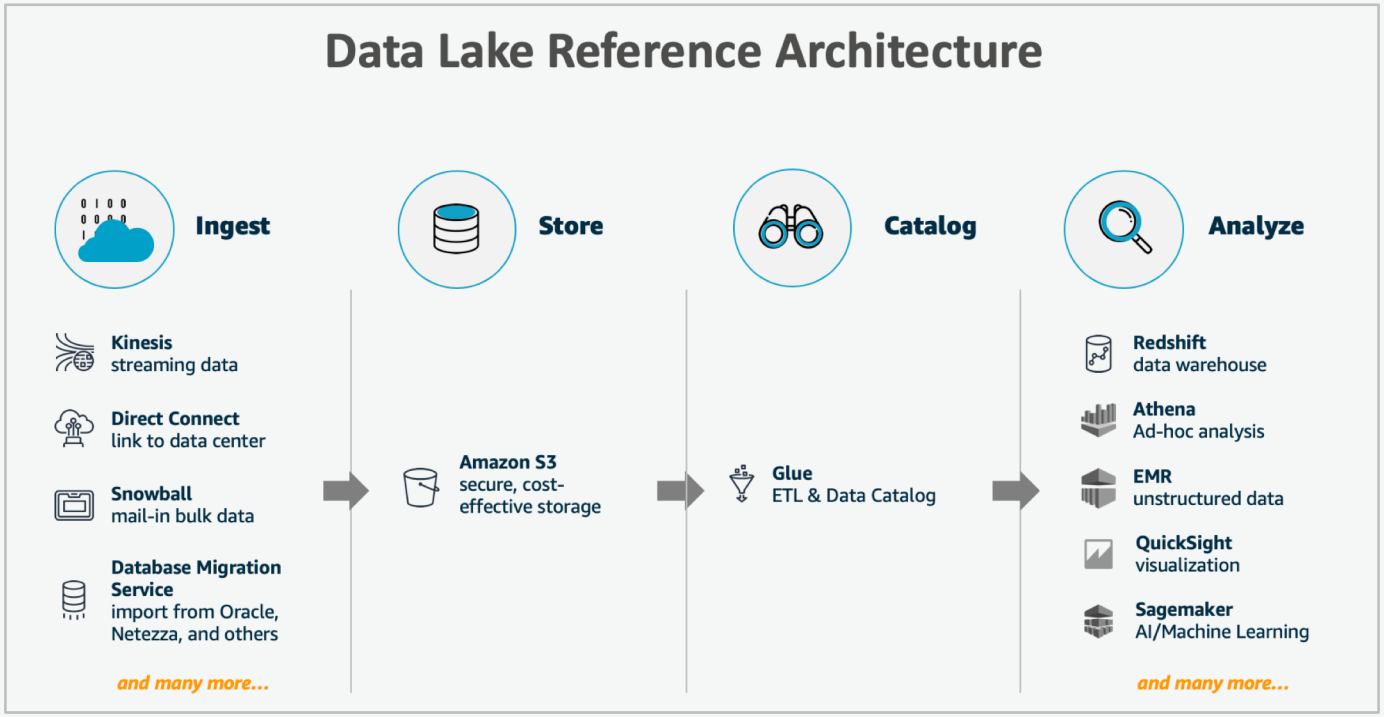
Data Lakes are optimized for large volumes of storage and ease of ingestion of any kind or flavor of data.

Data Warehouses are optimized for ease of retrieval of more critical and useful data.

Analytics Triangle:
Less work at the top.
More work at the bottom.

aws

Amazon Redshift



Amazon Redshift is a cloud-native data warehouse that's...

Most popular

More than 15K customers use Redshift and process more than 2 EB of data per day

Fastest

2-3x faster than other cloud DW solutions

Most cost-effective

25% less than other solutions using on-demand pricing and 75% with reserved instances (RIs).

Integrated with the data lake

Query exabytes of data directly in open formats with no loading required

aws

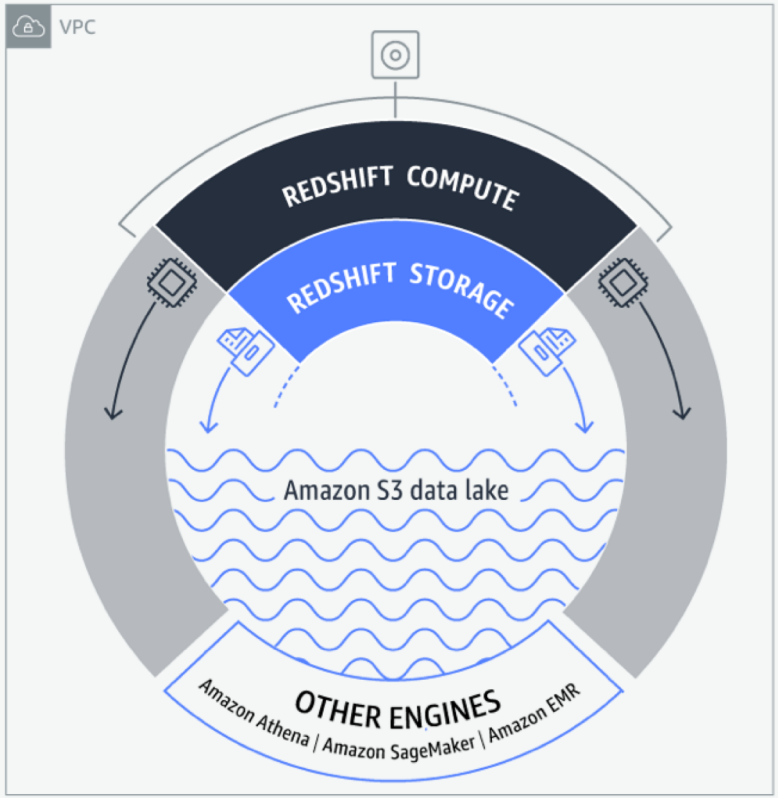
Lab Time

https://github.com/dbayardAWS/intro-data-lake

Please start Lab2.

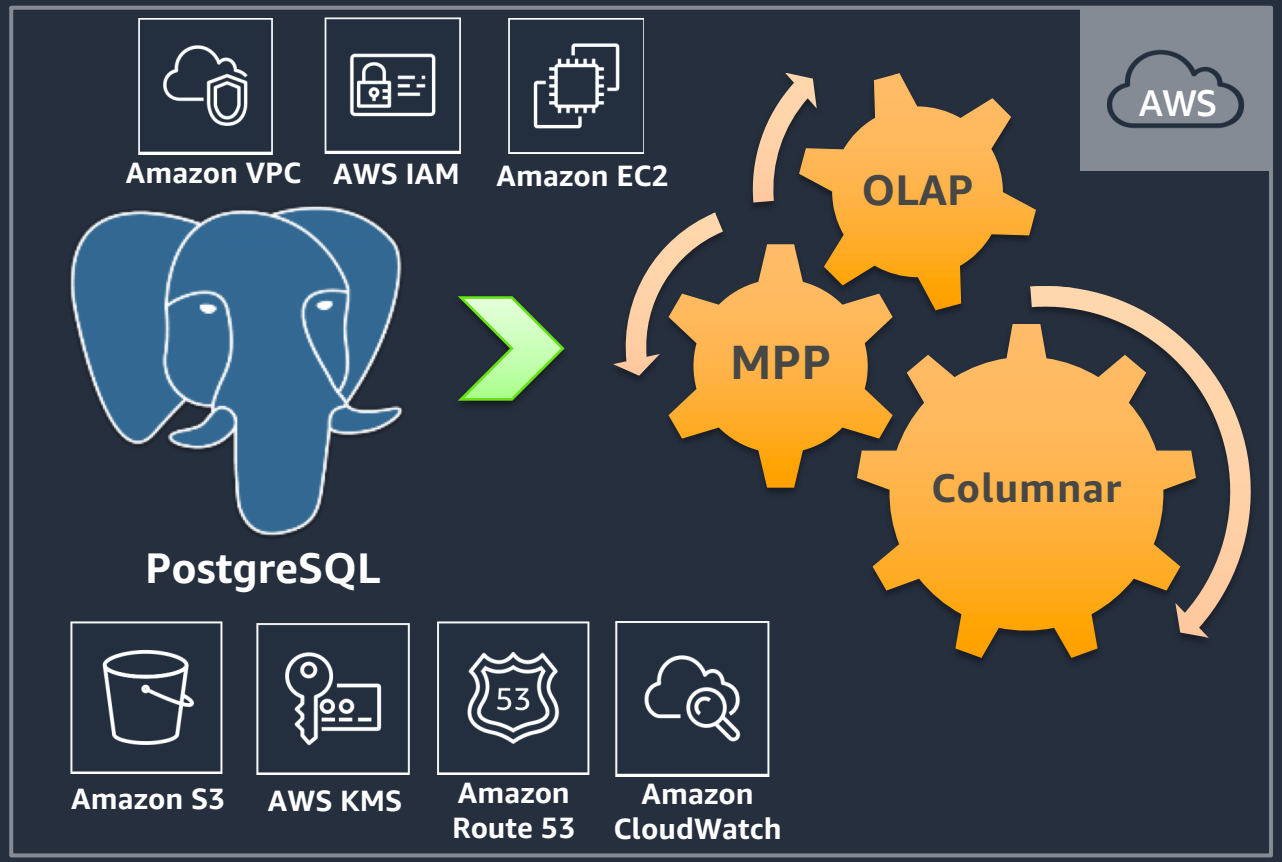
Continue thru the **Provision a new Redshift Cluster** step.

Redshift Conceptual Architecture



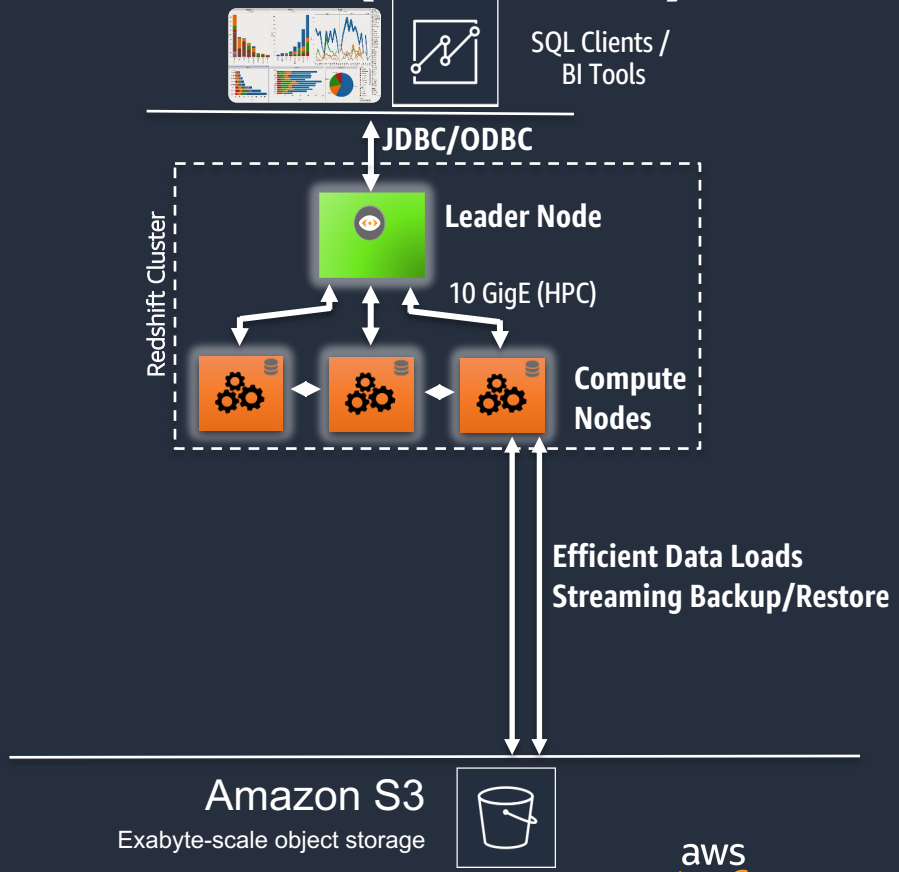
- Leader node creates query execution plans using machine learning and caches query results
- Compute scales to handle high concurrency and query the data lake directly
- Storage scales to exabytes of data in the Amazon S3 data lake in Parquet, ORC, and other open data formats

Redshift: What's Under the Hood?



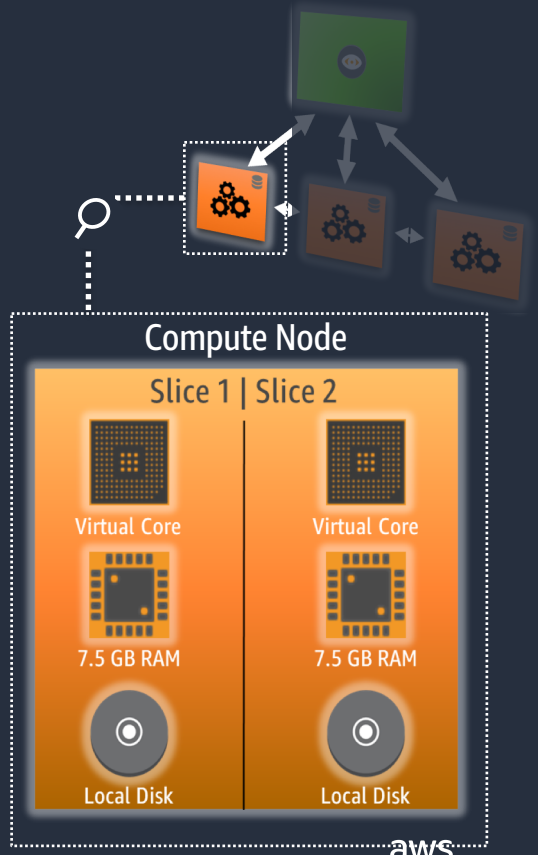
Redshift Cluster Architecture (Classic)

- Massively parallel, shared nothing architecture
- Streaming Backup/Restore from S3
- Leader node
  - SQL endpoint
  - Stores metadata
  - Coordinates parallel SQL processing; ML optimizations
  - Customers are not charged for the leader node for any cluster with two or more nodes
- Compute nodes
  - Local, columnar storage
  - Executes queries in parallel
  - Load, backup, restore



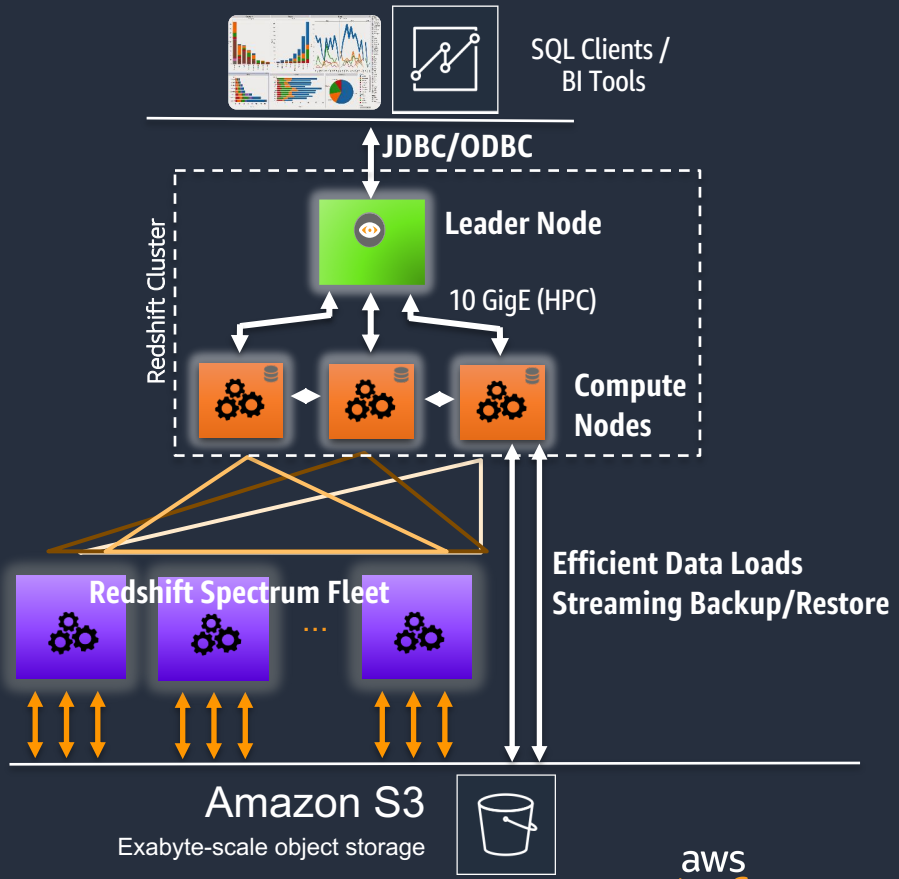
Compute Node: Under the Hood (Slices)

- A compute node is partitioned into either 2 or 16 *slices*; a slice can be thought of as a "virtual compute node"
- Each slice is allocated a portion of the compute node's memory and disk space, where it processes a portion of the workload assigned to the compute node by the leader node
- The leader node manages distributing data to the slices and apportions the workload for any queries or other database operations to the slices
- Slices work in parallel to complete operations



Redshift Cluster Architecture

- Massively parallel, shared nothing architecture
- Streaming Backup/Restore from S3
- Leader node
  - SQL endpoint
  - Stores metadata
  - Coordinates parallel SQL processing; ML optimizations
  - Customers are not charged for the leader node for any cluster with two or more nodes
- Compute nodes
  - Local, columnar storage
  - Executes queries in parallel
  - Load, backup, restore
- Amazon Redshift Spectrum nodes
  Execute queries directly against data lake

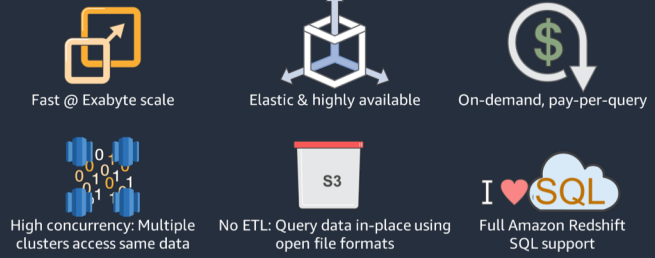


Redshift Spectrum Overview

- Redshift Spectrum is a feature of Amazon Redshift that allows SQL queries to reference external data on Amazon S3 as they would any other table in Amazon Redshift
- Allows for querying of potentially Exabytes of data in an S3 data lake from within Amazon Redshift
- Data is queried in-place, so no loading of data into your Redshift cluster is required
- Powered by a fleet of Amazon Redshift Spectrum nodes

  There is a soft limit/allowance of 10 spectrum nodes per Redshift cluster slice. So, a cluster with 20 slices will be allowed to leverage up to 200 Spectrum nodes

Run SQL queries directly against data in S3 using thousands of nodes



Lab Time

https://github.com/dbayardAWS/intro-data-lake

Please finish Lab2.

Lunch

We will start again at 12:30pm

Restart

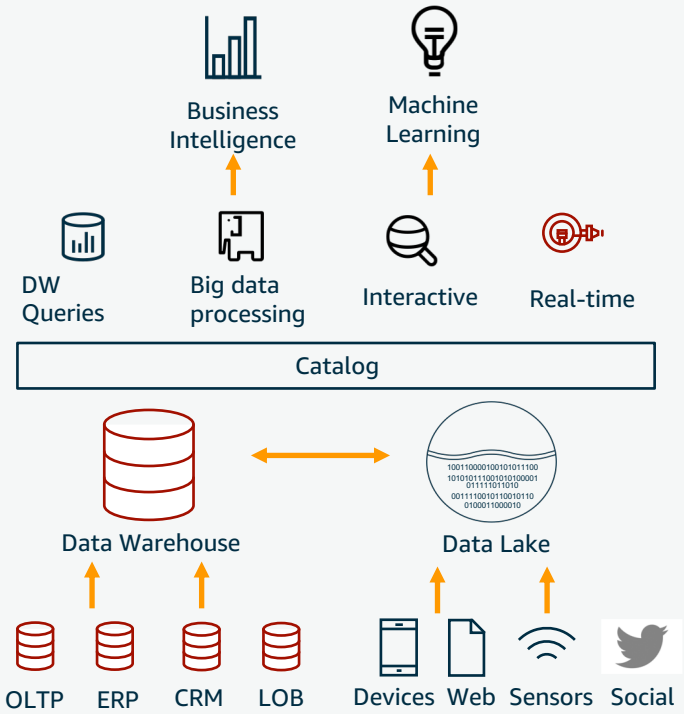
What did you think of the labs?
Any questions about the content so far?

AWS Lake Formation

Agenda

- Data Lake Process Steps
- AWS Lake Formation Overview
- AWS Lake Formation Demonstration

Why data lakes?



Data Lakes provide:

Relational and non-relational data

Scale-out to Exabytes

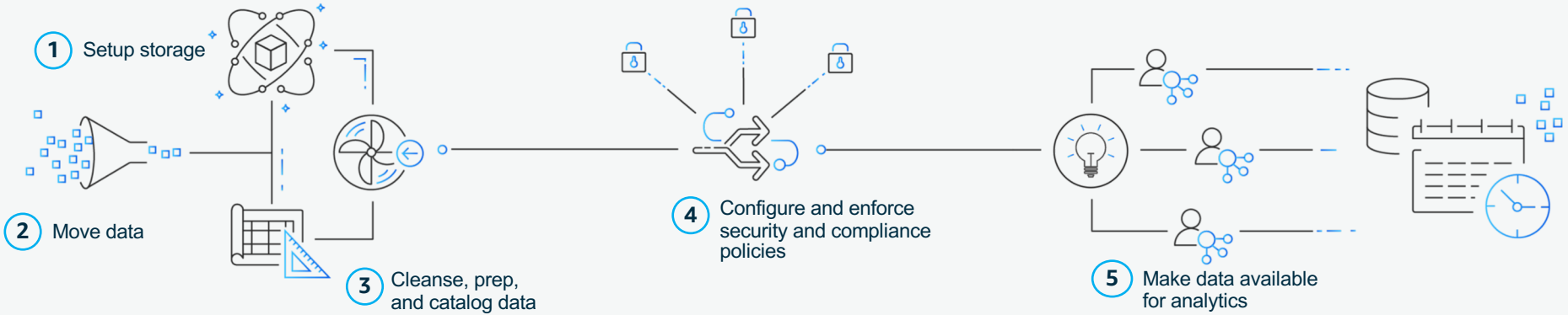
Diverse set of analytics and machine learning tools

Work on data without any data movement

Designed for low cost storage and analytics

aws

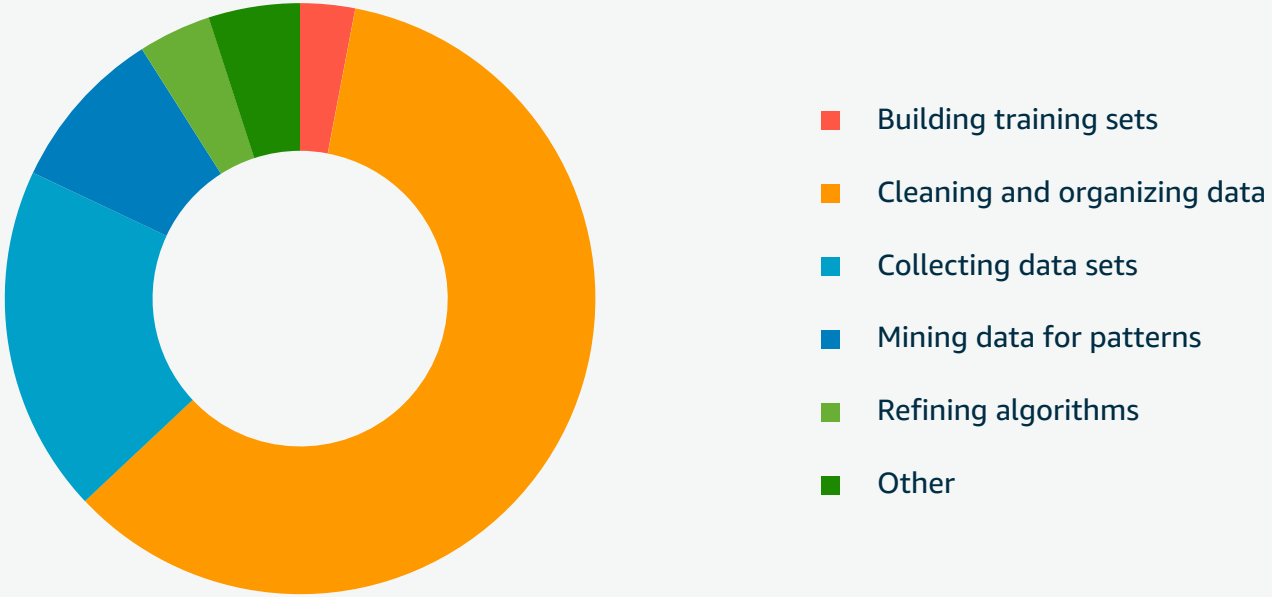
Typical steps of building a data lake



aws

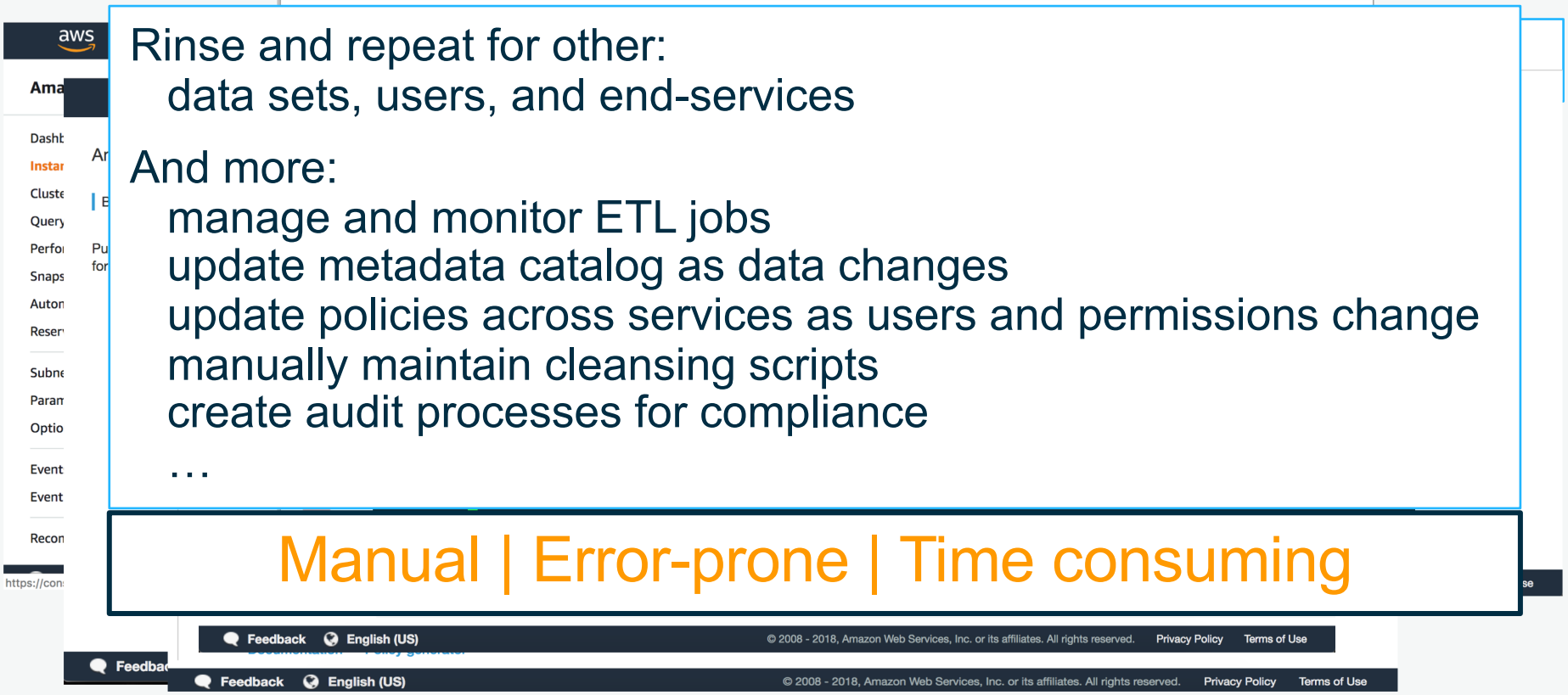
Data Preparation Accounts for ~80% of the Work

Building data lakes can still take months



Sample of steps required

Configure access from analytics services



Rinse and repeat for other:

- data sets, users, and end-services

And more:

- manage and monitor ETL jobs
- update metadata catalog as data changes
- update policies across services as users and permissions change
- manually maintain cleansing scripts
- create audit processes for compliance
- …

Manual | Error-prone | Time consuming

AWS Lake Formation

Build a secure data lake in days

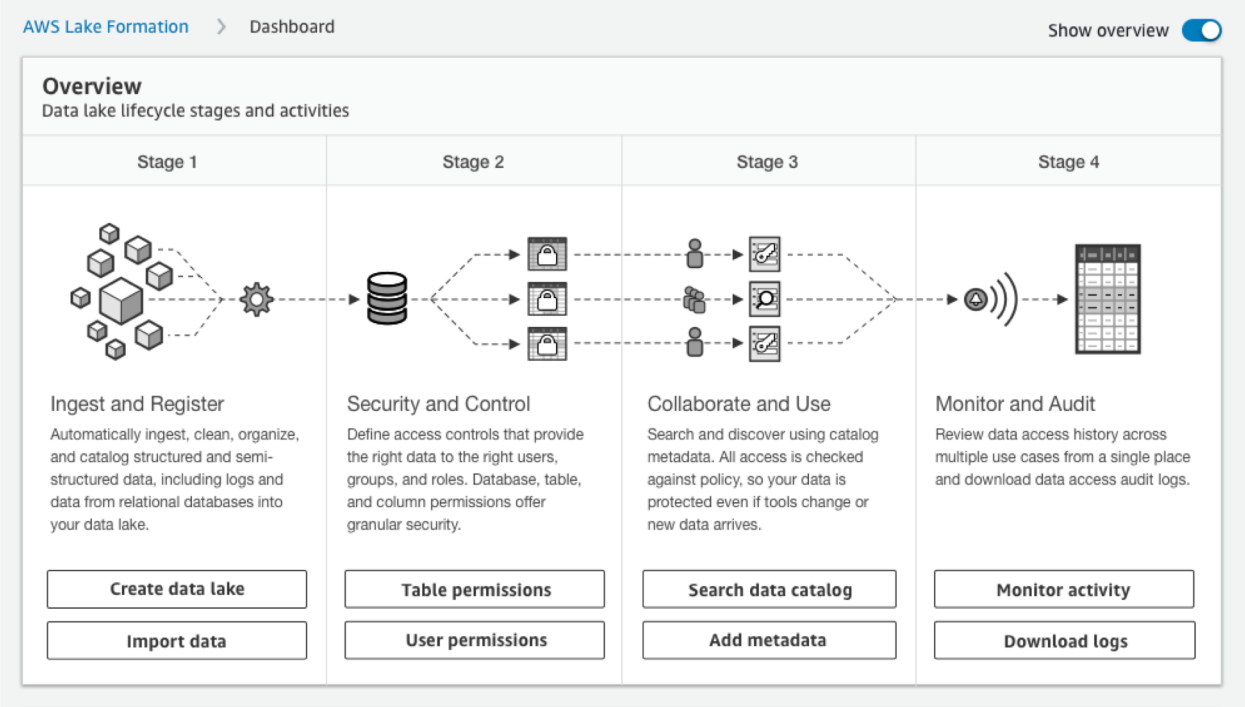
Identify, ingest, clean, and transform data

Enforce security policies across multiple services

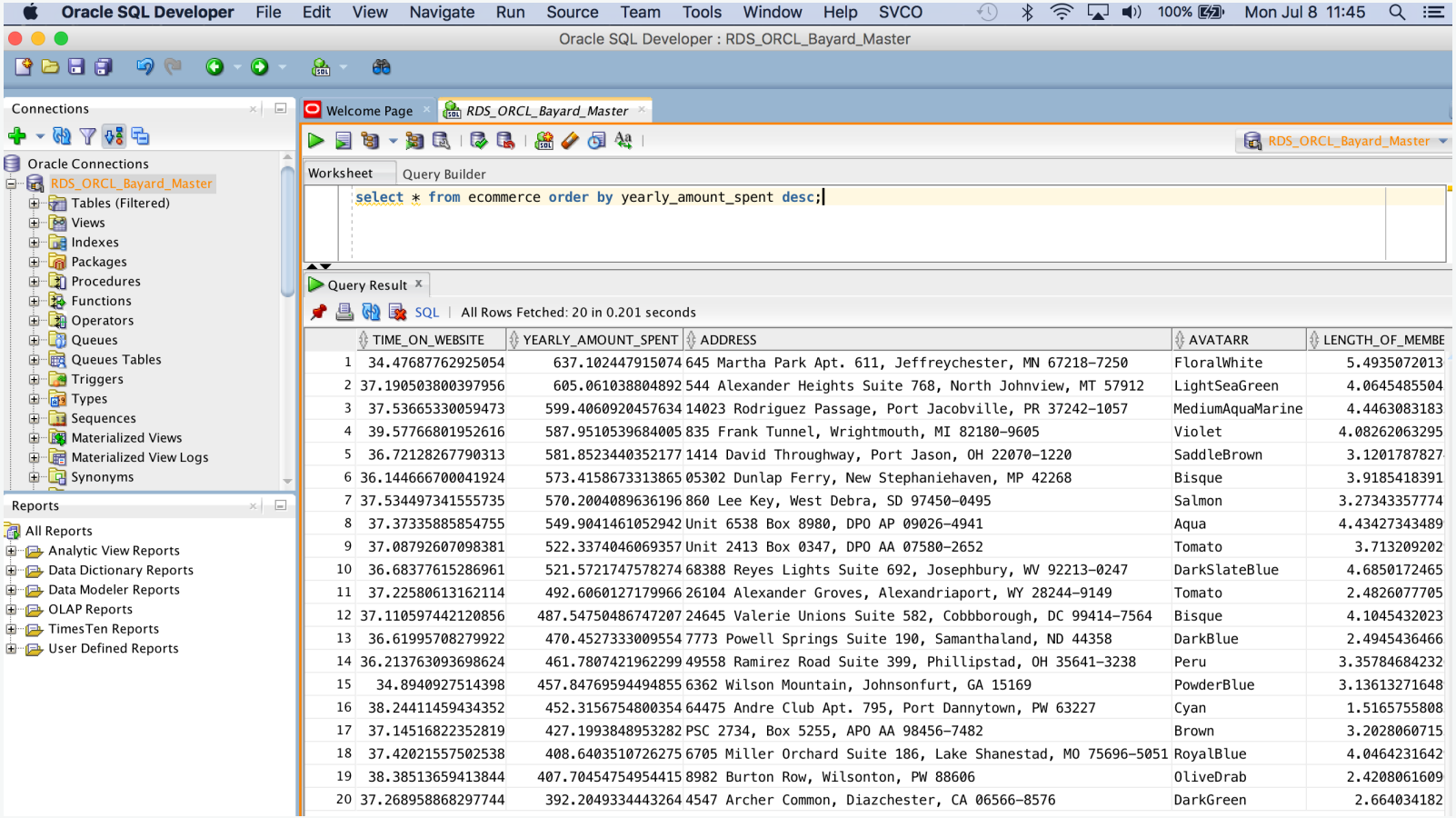
Gain and manage new insights

aws

How it works

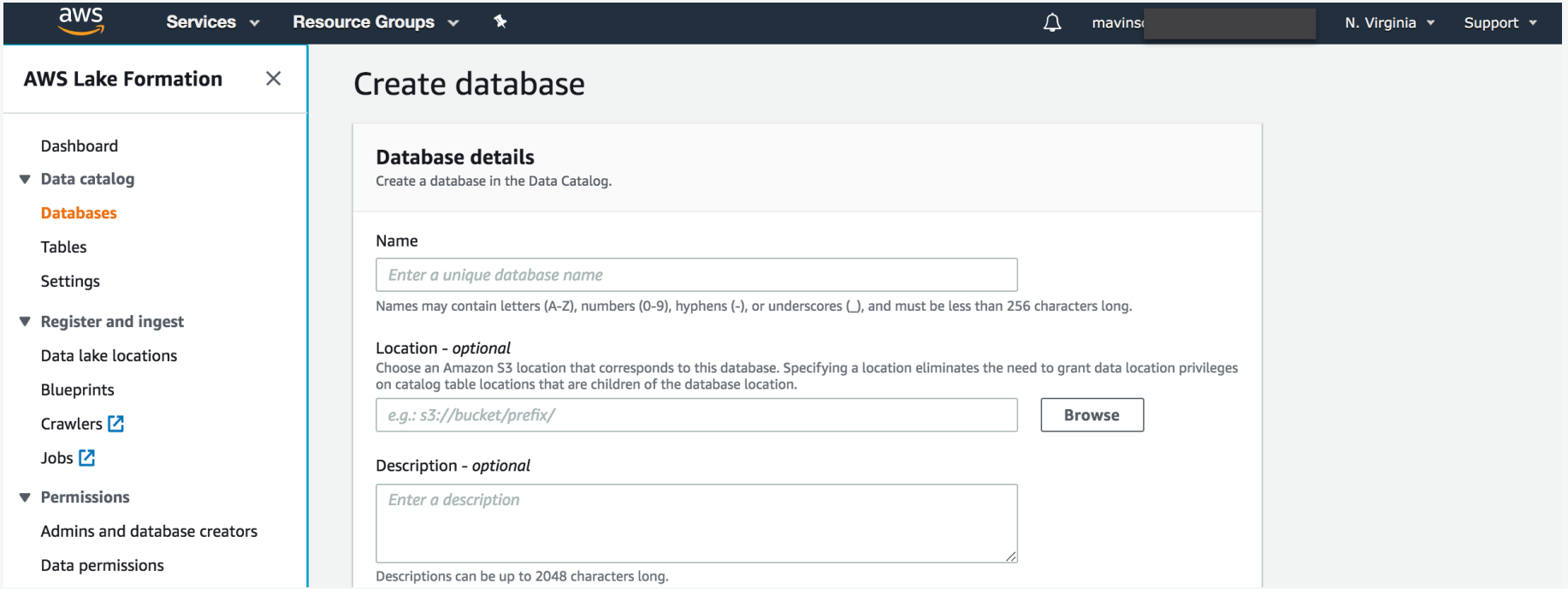


Demonstration Data Set

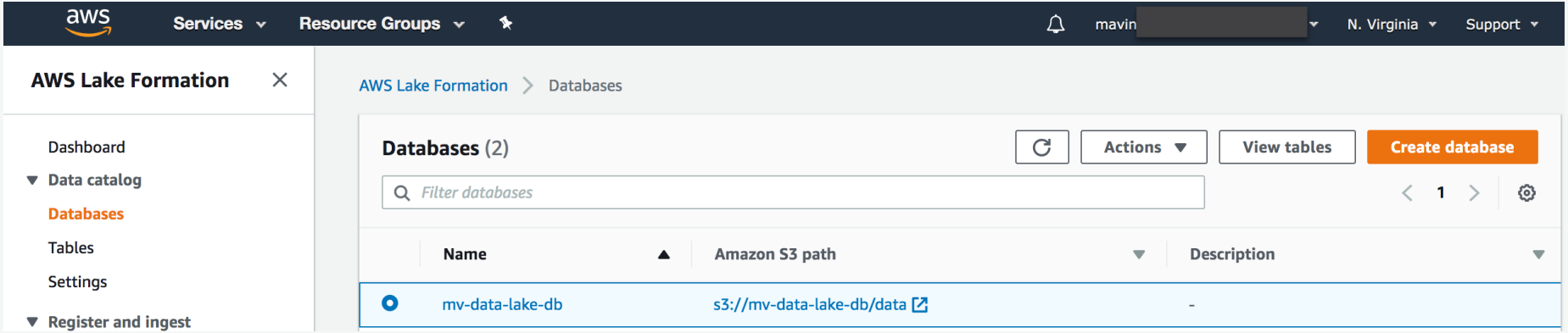


	TIME_ON_WEBSITE	YEARLY_AMOUNT_SPENT	ADDRESS	AVATARR	LENGTH_OF_MEMBE
1	34.47687762925054	637.102447915074	645 Martha Park Apt. 611, Jeffreychester, MN 67218-7250	FloralWhite	5.4935072013
2	37.190503800397956	605.061038804892	544 Alexander Heights Suite 768, North Johnview, MT 57912	LightSeaGreen	4.0645485504
3	37.53665330059473	599.4060920457634	14023 Rodriguez Passage, Port Jacobville, PR 37242-1057	MediumAquaMarine	4.4463083183
4	39.57766801952616	587.9510539684005	835 Frank Tunnel, Wrightmouth, MI 82180-9605	Violet	4.08262063295
5	36.72128267790313	581.8523440352177	1414 David Throughway, Port Jason, OH 22070-1220	SaddleBrown	3.1201787827
6	36.144666700041924	573.4158673313865	05302 Dunlap Ferry, New Stephaniehaven, MP 42268	Bisque	3.9185418391
7	37.534497341555735	570.2004089636196	860 Lee Key, West Debra, SD 97450-0495	Salmon	3.27343357774
8	37.37335885854755	549.9041461052942	Unit 6538 Box 8980, DPO AP 09026-4941	Aqua	4.43427343489
9	37.08792607098381	522.3374046069357	Unit 2413 Box 0347, DPO AA 07580-2652	Tomato	3.713209202
10	36.68377615286961	521.5721747578274	68388 Reyes Lights Suite 692, Josephbury, WV 92213-0247	DarkSlateBlue	4.6850172465
11	37.22580613162114	492.6060127179966	26104 Alexander Groves, Alexandriaport, WY 28244-9149	Tomato	2.4826077705
12	37.110597442120856	487.54750486747207	24645 Valerie Unions Suite 582, Cobbborough, DC 99414-7564	Bisque	4.1045432023
13	36.61995708279922	470.4527333009554	7773 Powell Springs Suite 190, Samanthaland, ND 44358	DarkBlue	2.4945436466
14	36.213763093698624	461.7807421962299	49558 Ramirez Road Suite 399, Phillipstad, OH 35641-3238	Peru	3.35784684232
15	34.8940927514398	457.84769594494855	6362 Wilson Mountain, Johnsonfurt, GA 15169	PowderBlue	3.13613271648
16	38.24411459434352	452.3156754800354	64475 Andre Club Apt. 795, Port Dannytown, PW 63227	Cyan	1.5165755808
17	37.14516822352819	427.1993848953282	PSC 2734, Box 5255, APO AA 98456-7482	Brown	3.2028060715
18	37.42021557502538	408.6403510726275	6705 Miller Orchard Suite 186, Lake Shanestad, MO 75696-5051	RoyalBlue	4.0464231642
19	38.38513659413844	407.70454754954415	8982 Burton Row, Wilsonton, PW 88606	OliveDrab	2.4208061609
20	37.268958868297744	392.2049334443264	4547 Archer Common, Diazchester, CA 06566-8576	DarkGreen	2.664034182

Step 1: Create a Lake Formation Database



Step 1: Create a Lake Formation Database



Step 2: Grant Privileges [Data Formation Database]

Grant permissions

Grant access permissions to specific users and roles.

IAM users and roles

Add one or more IAM users or roles.

Choose IAM principals to add

Database

Add one or more databases.

Choose databases

Database permissions

Choose the specific access permissions to grant.

Select all | Create table | Alter | Drop

Grant all

Enabling this permission grants full access to the specified resources while still logging access requests based on the individual permission settings above. It is typically used for debugging. Specific individual permissions set above will take effect when Grant all is later removed.

Grantable permissions

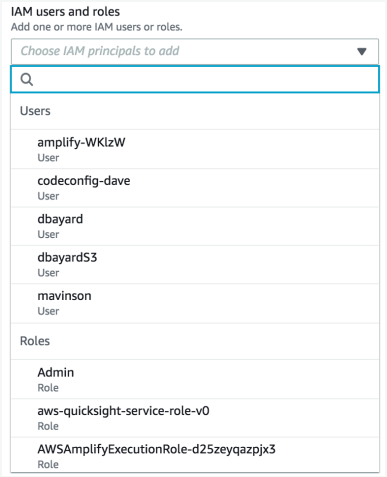
Choose the specific permissions that may be granted to others.

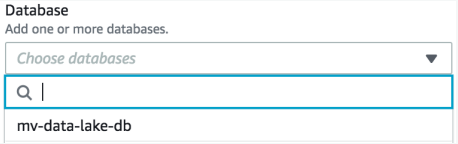
Select all | Create table | Alter | Drop

Grant all

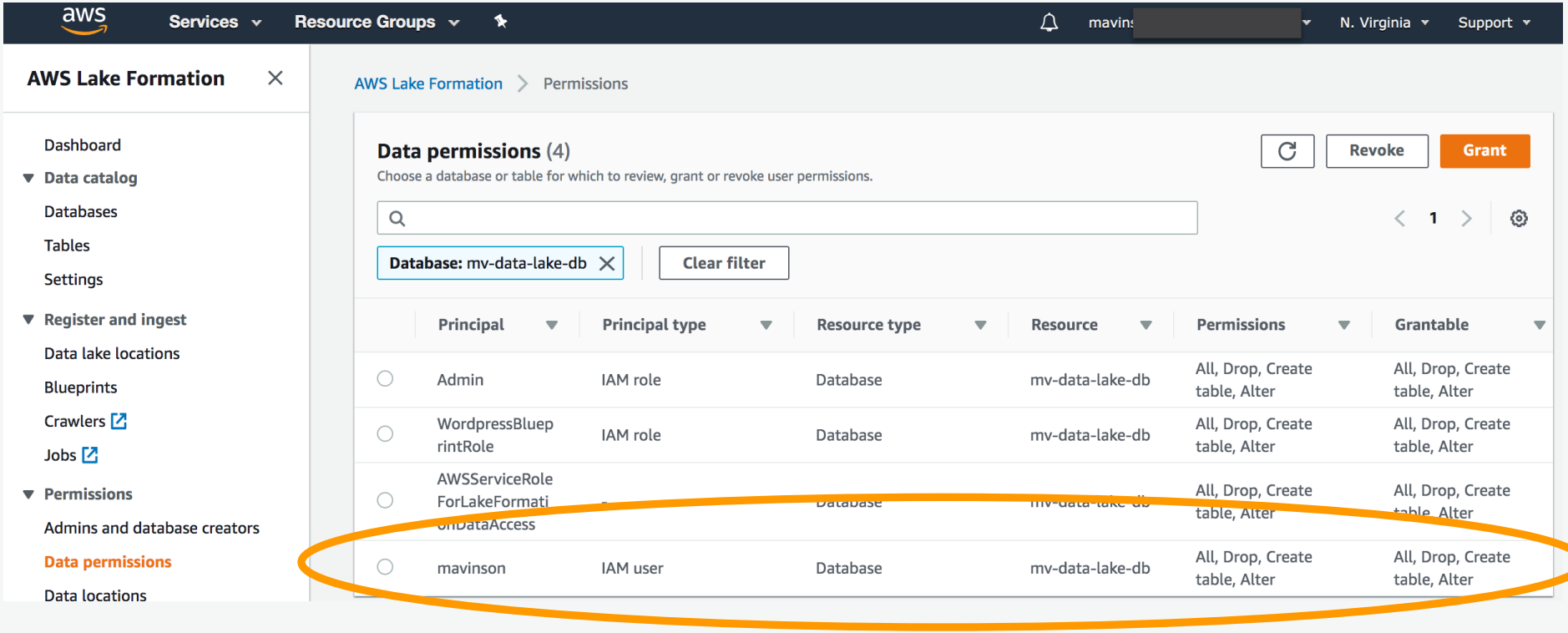
Enabling this permission grants full access to the specified resources while still logging access requests based on the individual permission settings above. It is typically used for debugging. Specific individual permissions set above will take effect when Grant all is later removed.

Cancel | Grant



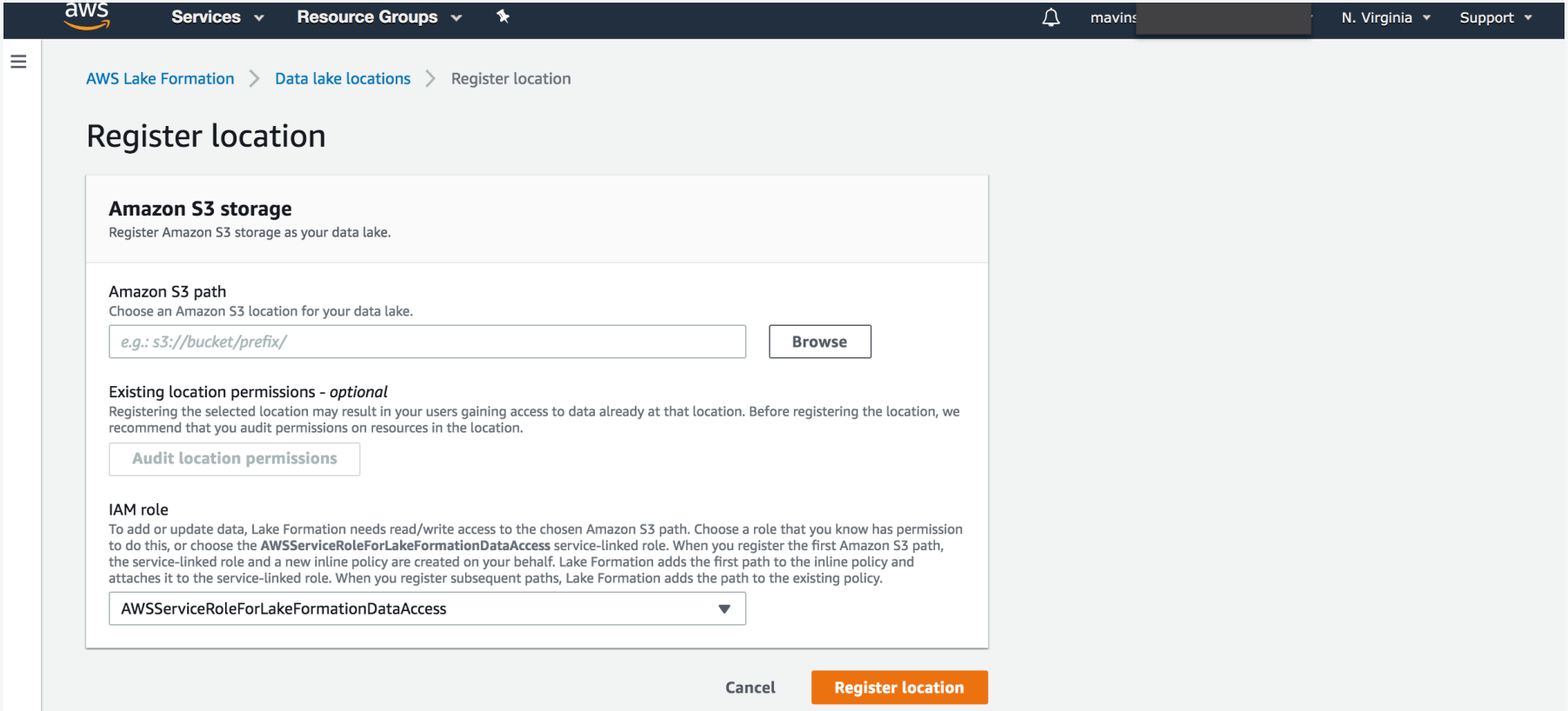


Step 2: Grant Privileges [Data Formation Database]

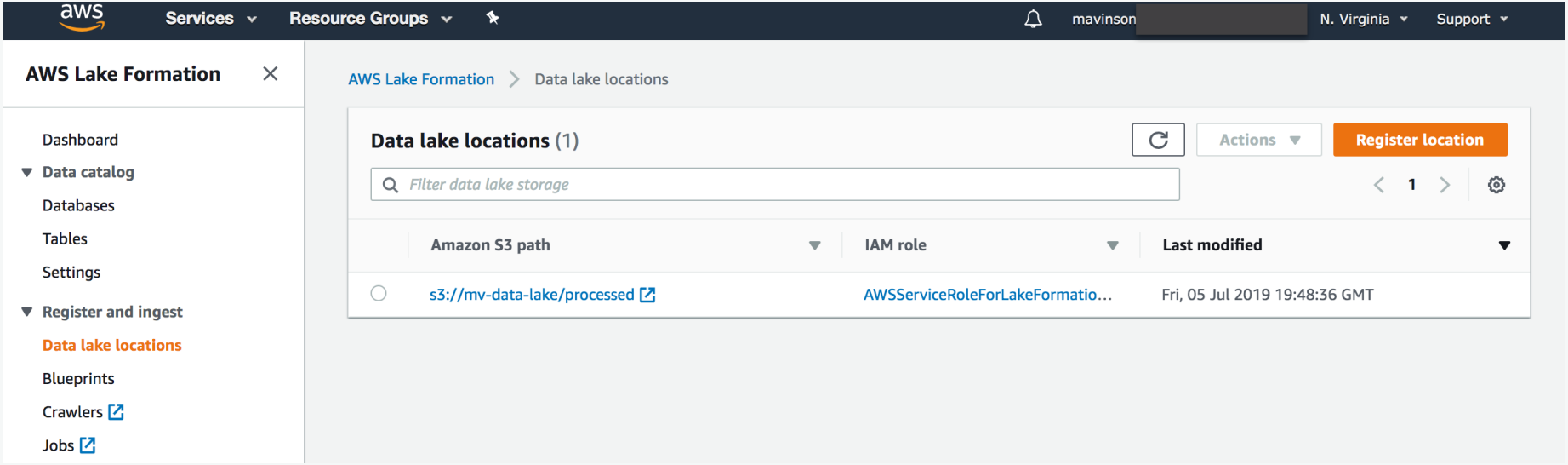


aws

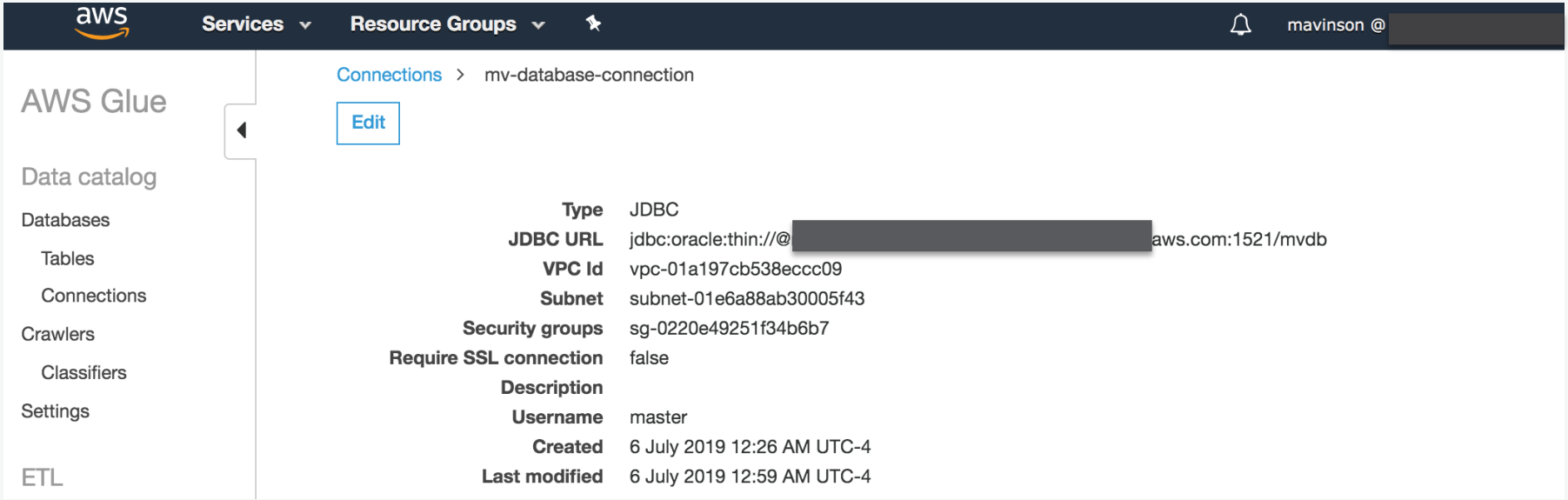
Step 3: Create a Data Lake Location



Step 3: Create a Data Lake Location



Step 4: Create a Glue Database Connection



Step 5: Create a Blueprint

You

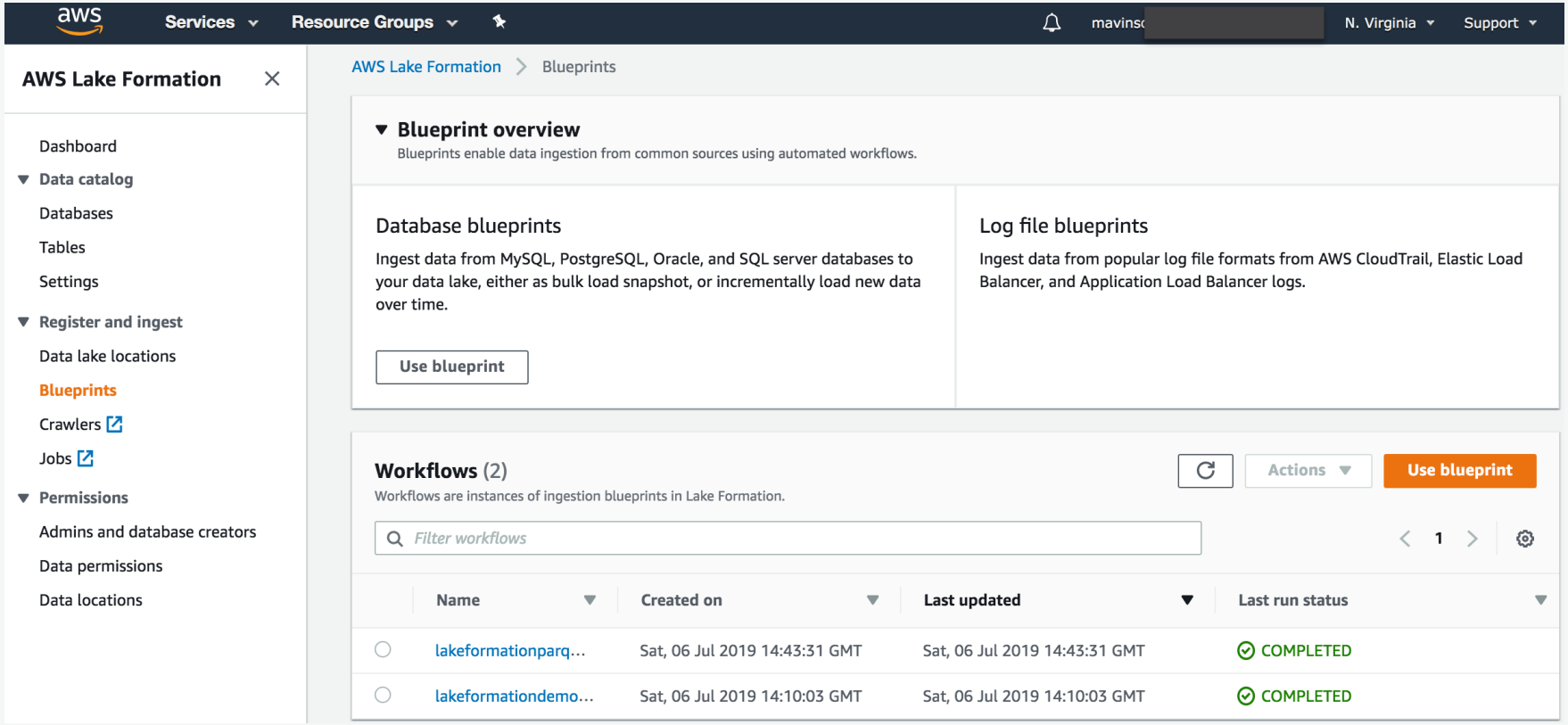
1. Point us to the source
2. Tell us the location to load to in your data lake
3. Specify how often you want to load the data

Blueprints

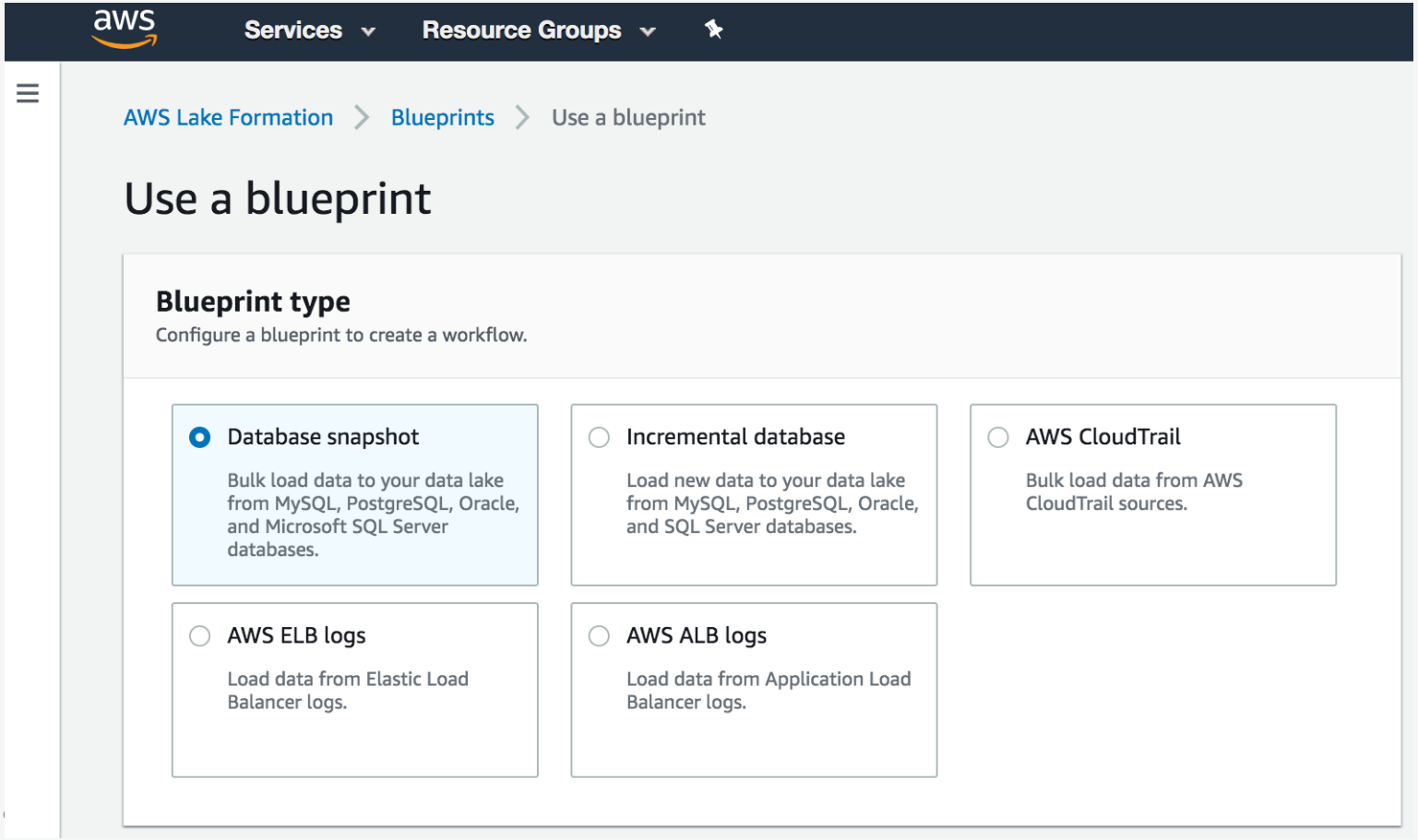
1. Discover the source table(s) schema
2. Automatically convert to the target data format
3. Automatically partition the data based on the partitioning schema
4. Keep track of data that was already processed
5. You can customize any of the above

aws

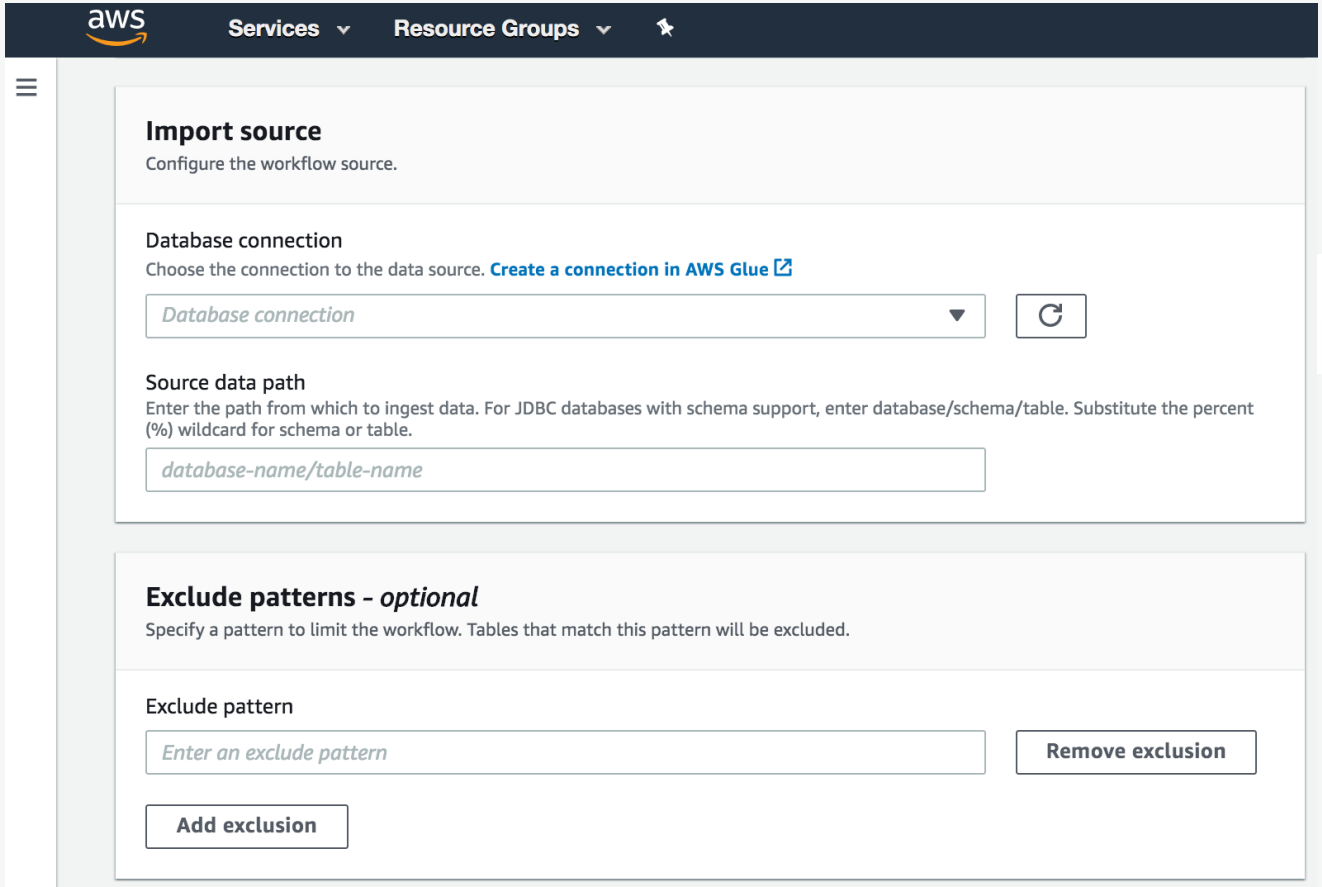
Step 5: Create a Blueprint

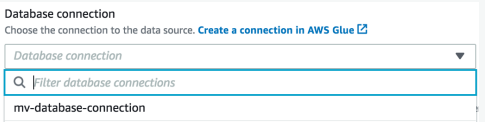


Step 5: Create a Blueprint (continued)



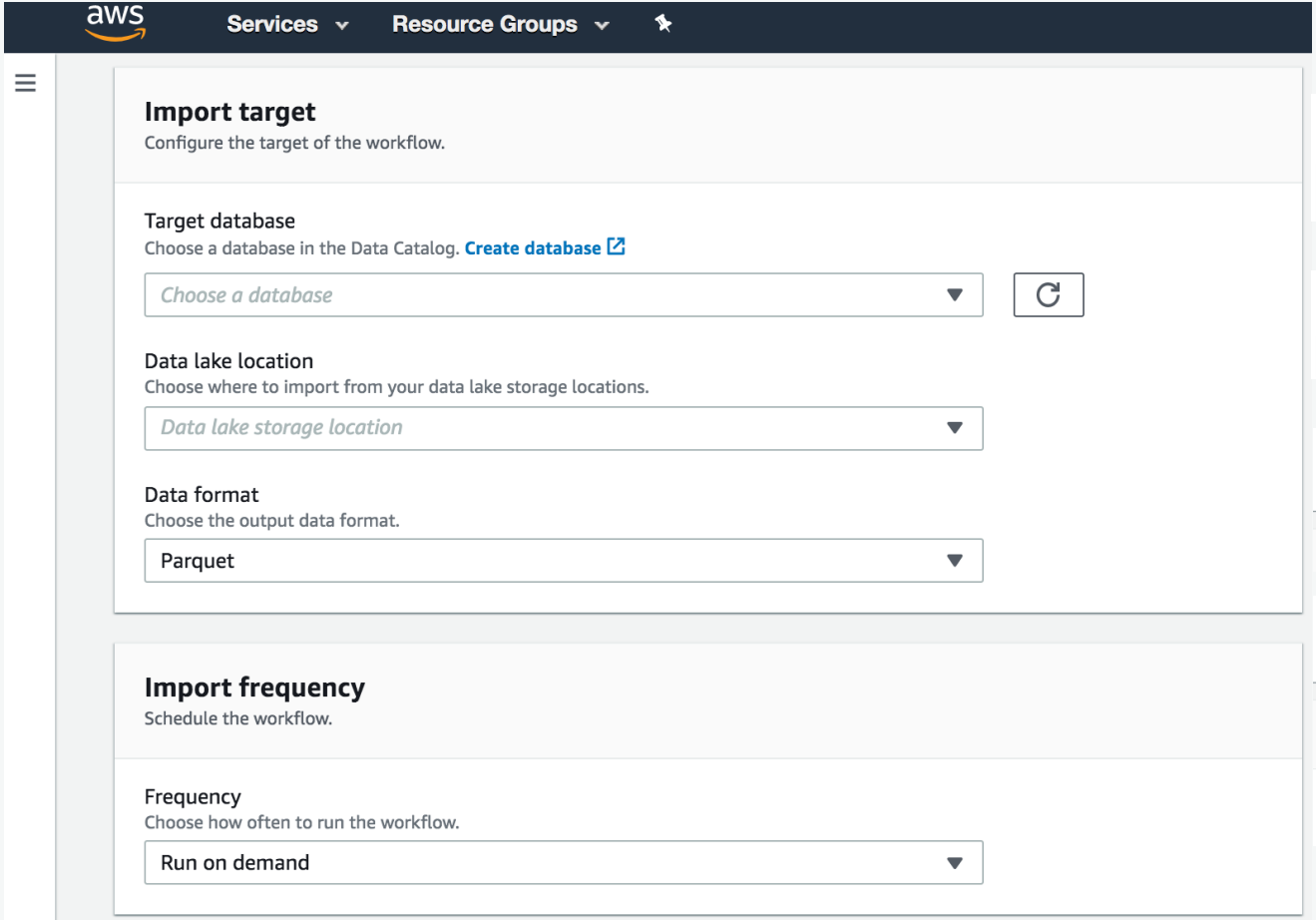
Step 5: Create a Blueprint (continued)

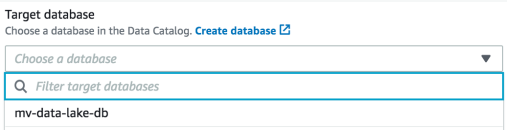


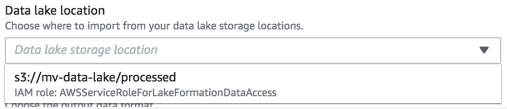


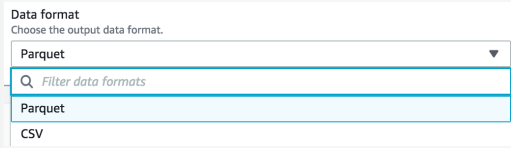
aws

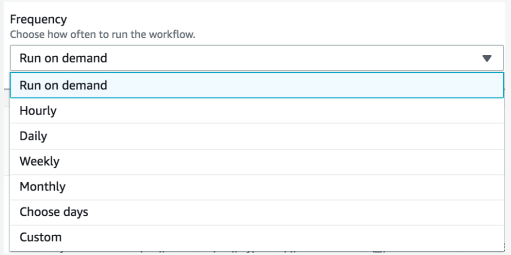
Step 5: Create a Blueprint (continued)



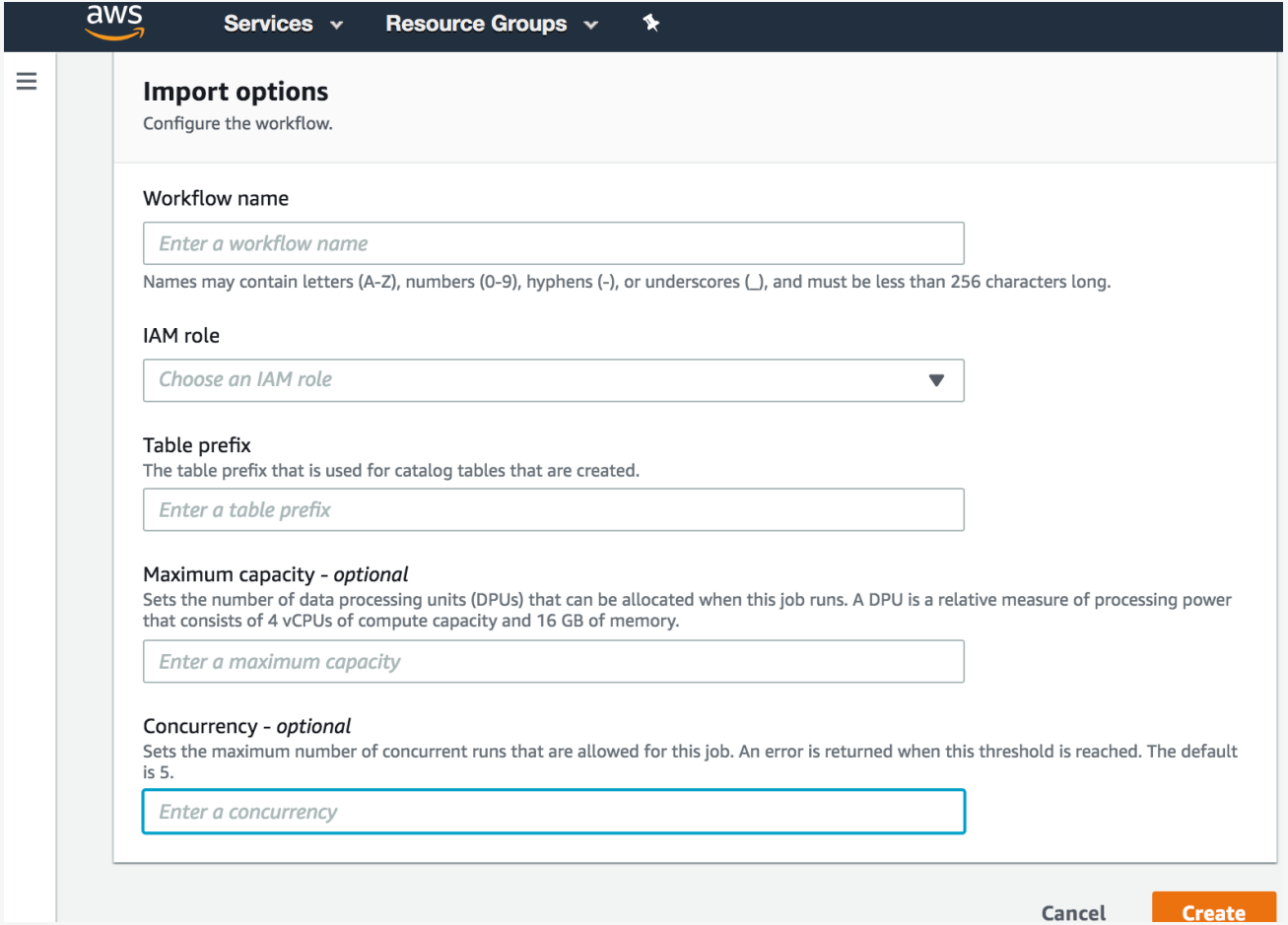




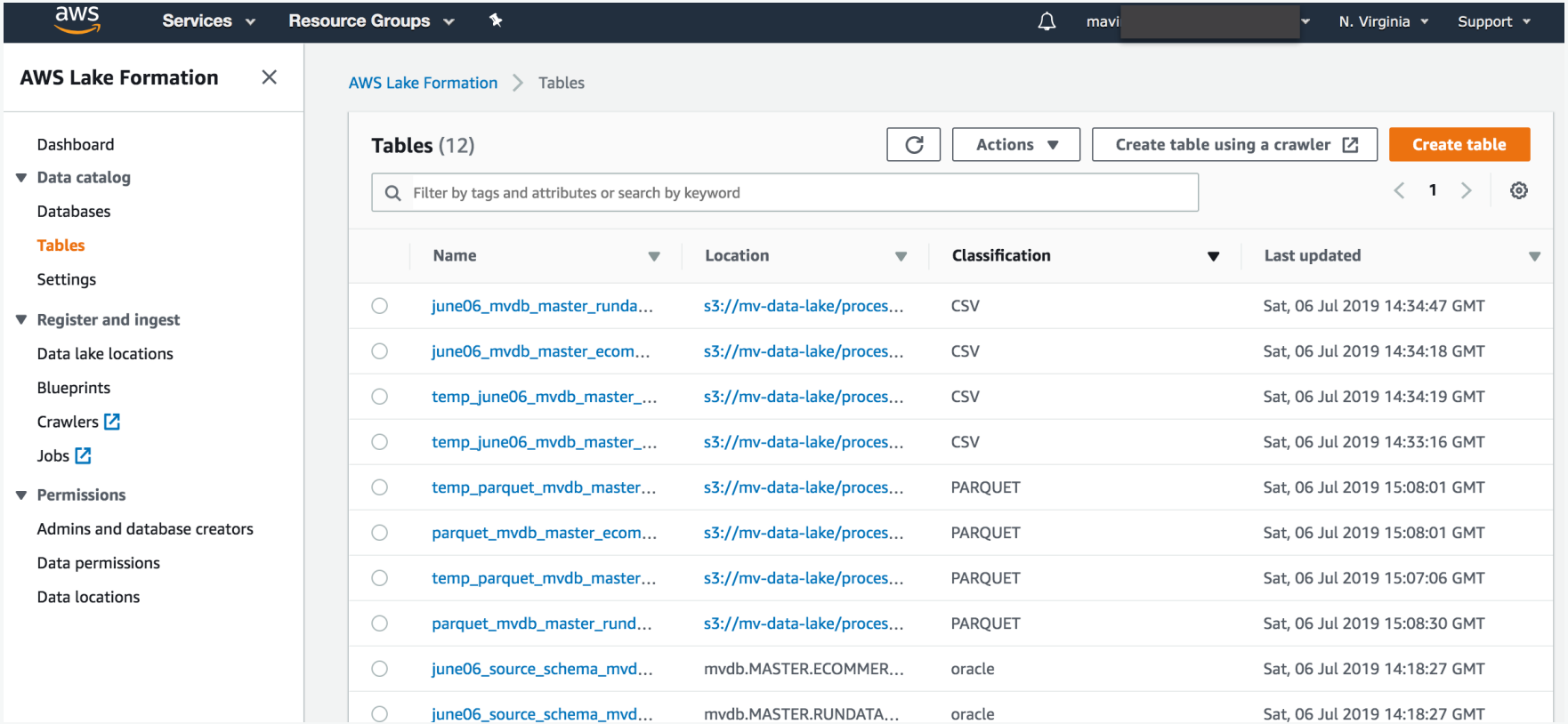




Step 5: Create a Blueprint (continued)

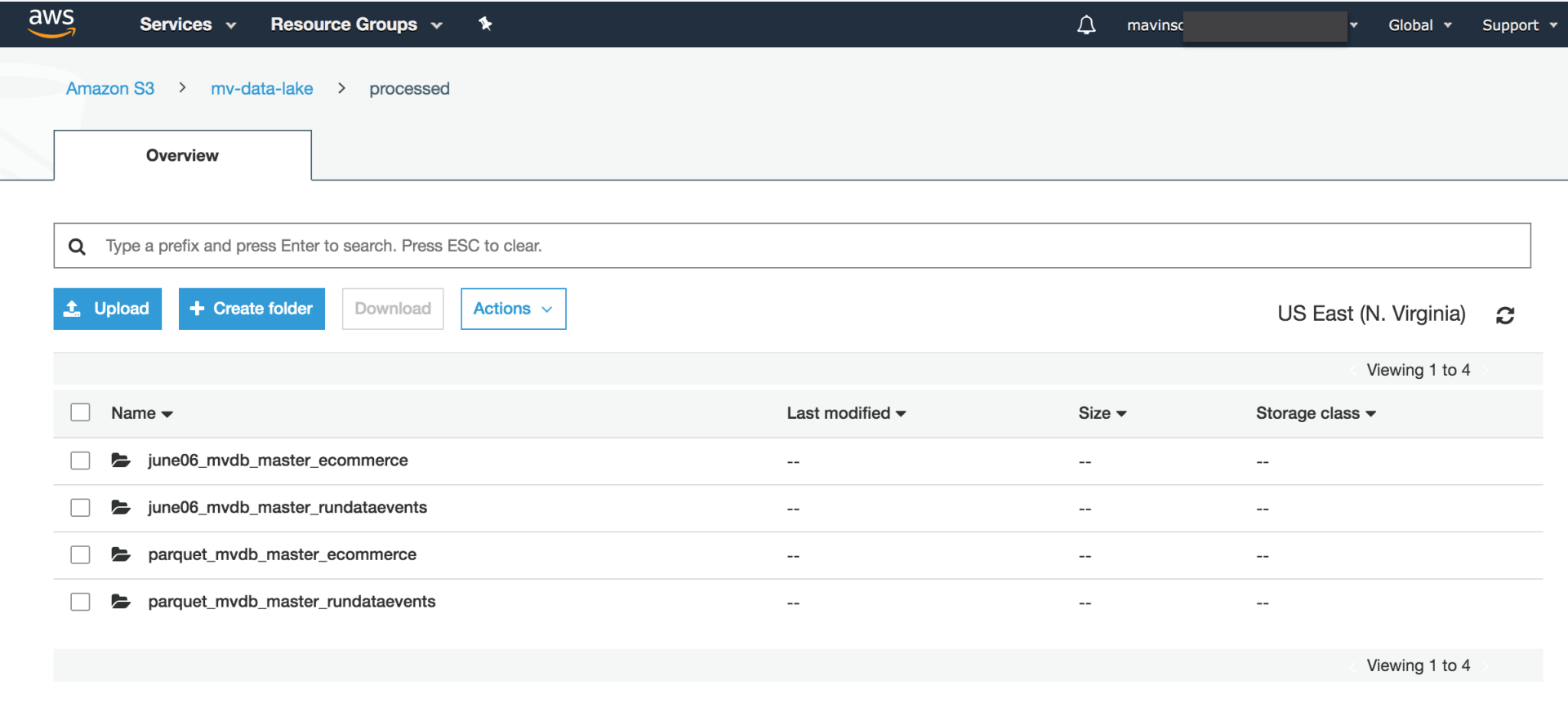


Lake Formation Setup and Ingestion is COMPLETE

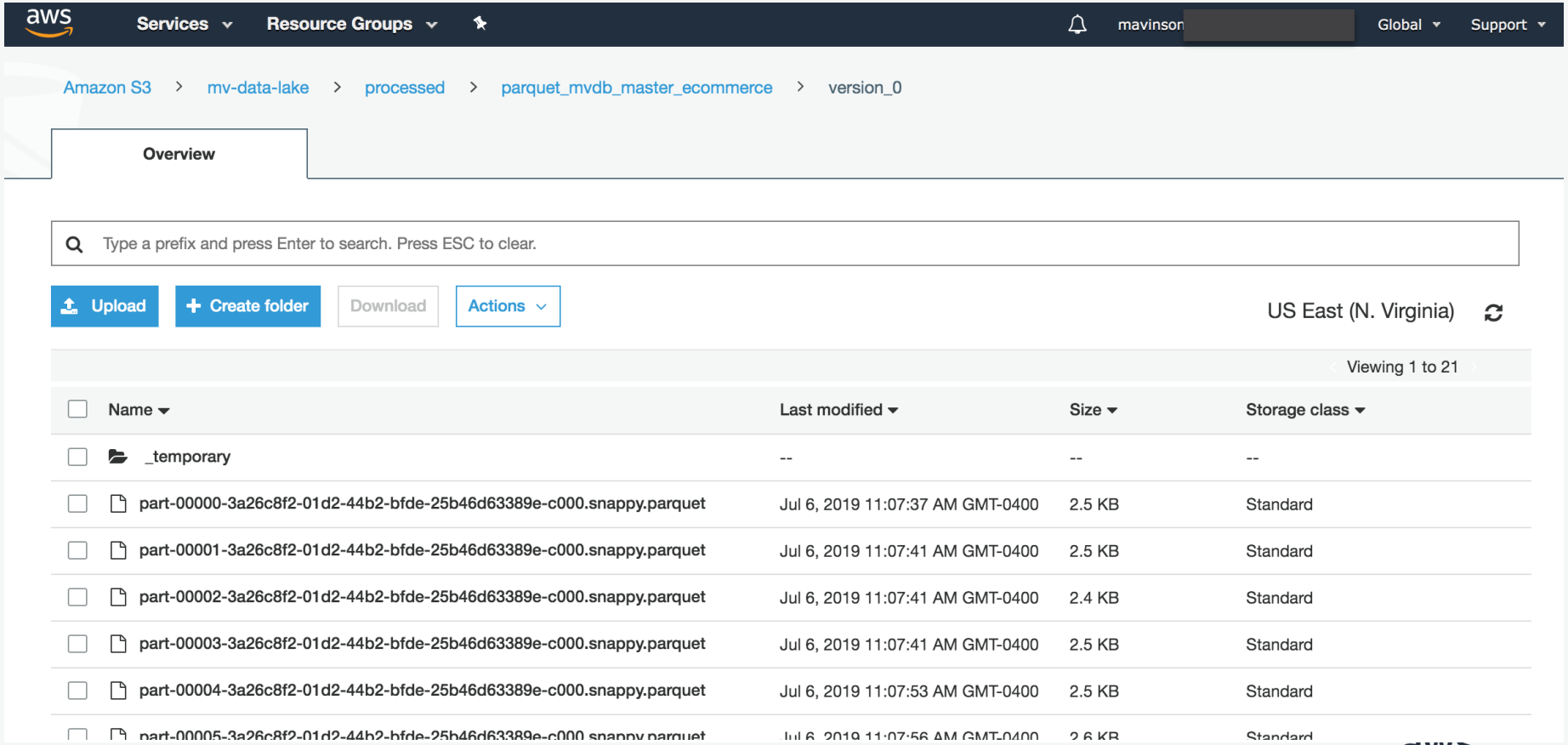


aws

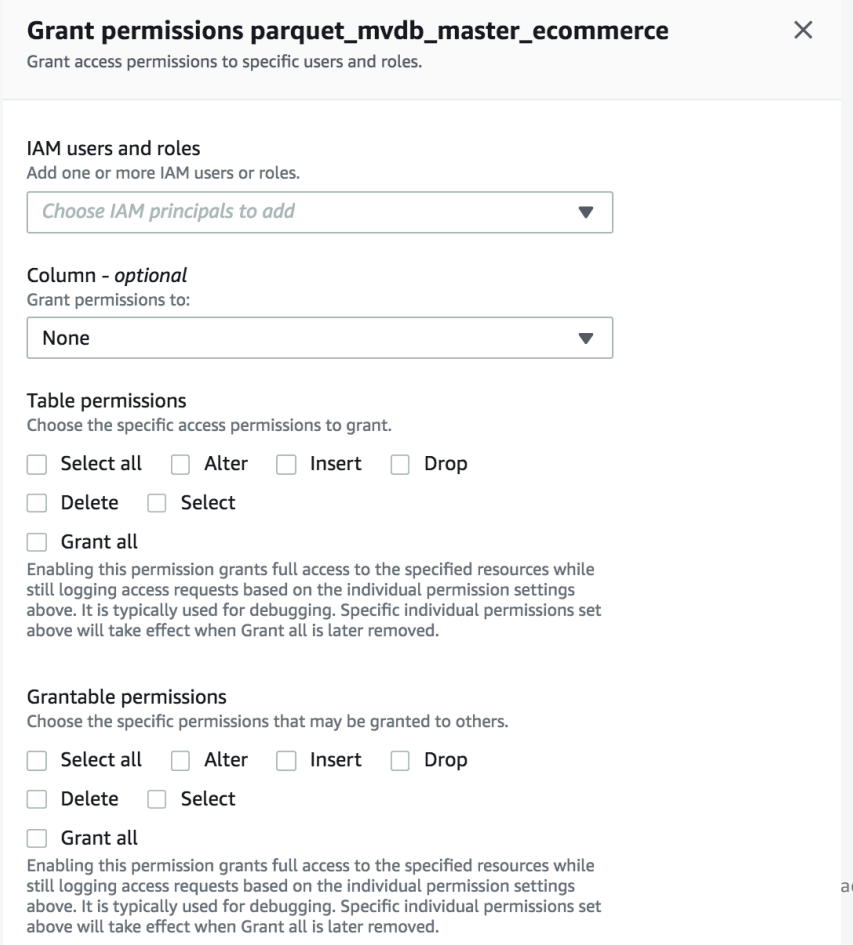
Lake Formation Setup and Ingestion is COMPLETE



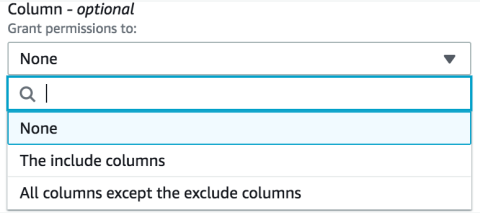
Lake Formation Setup and Ingestion is COMPLETE



Grant Privileges to Lake Formation Tables

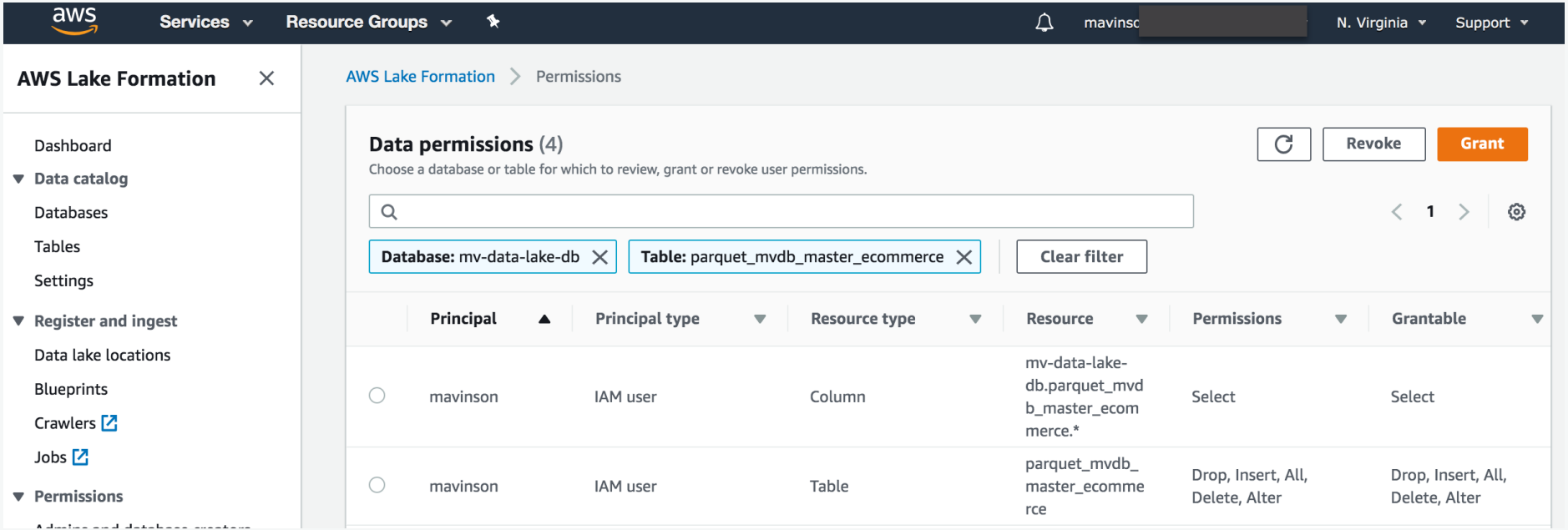


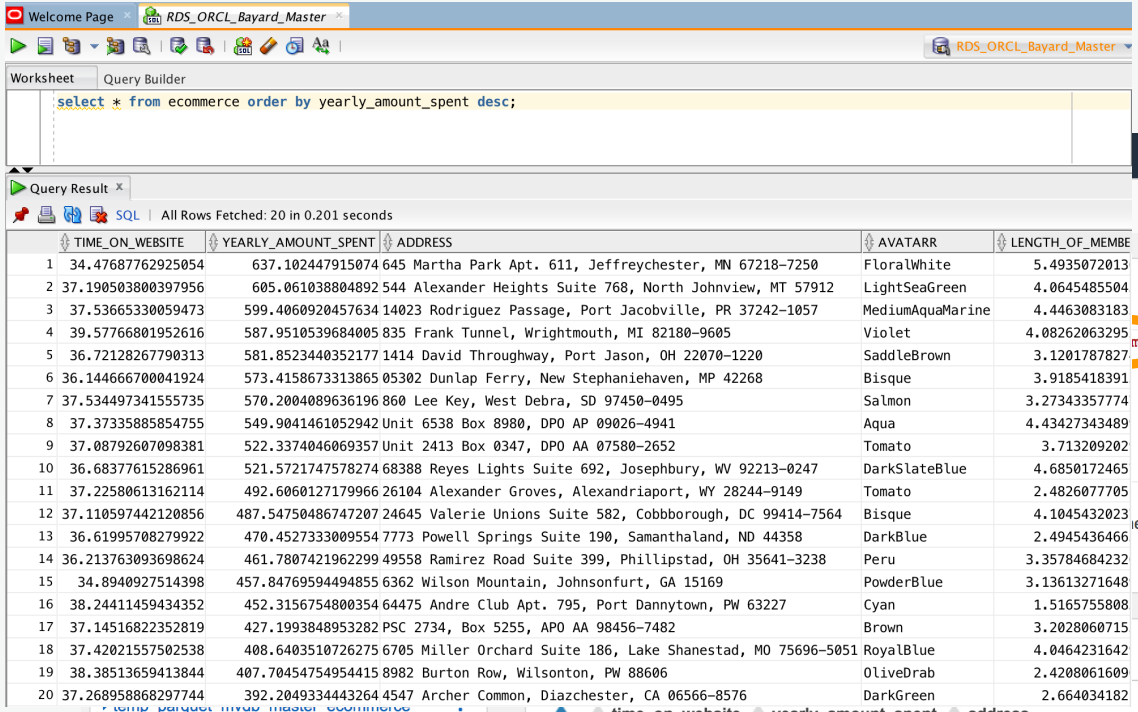
Grant table and column-level permissions



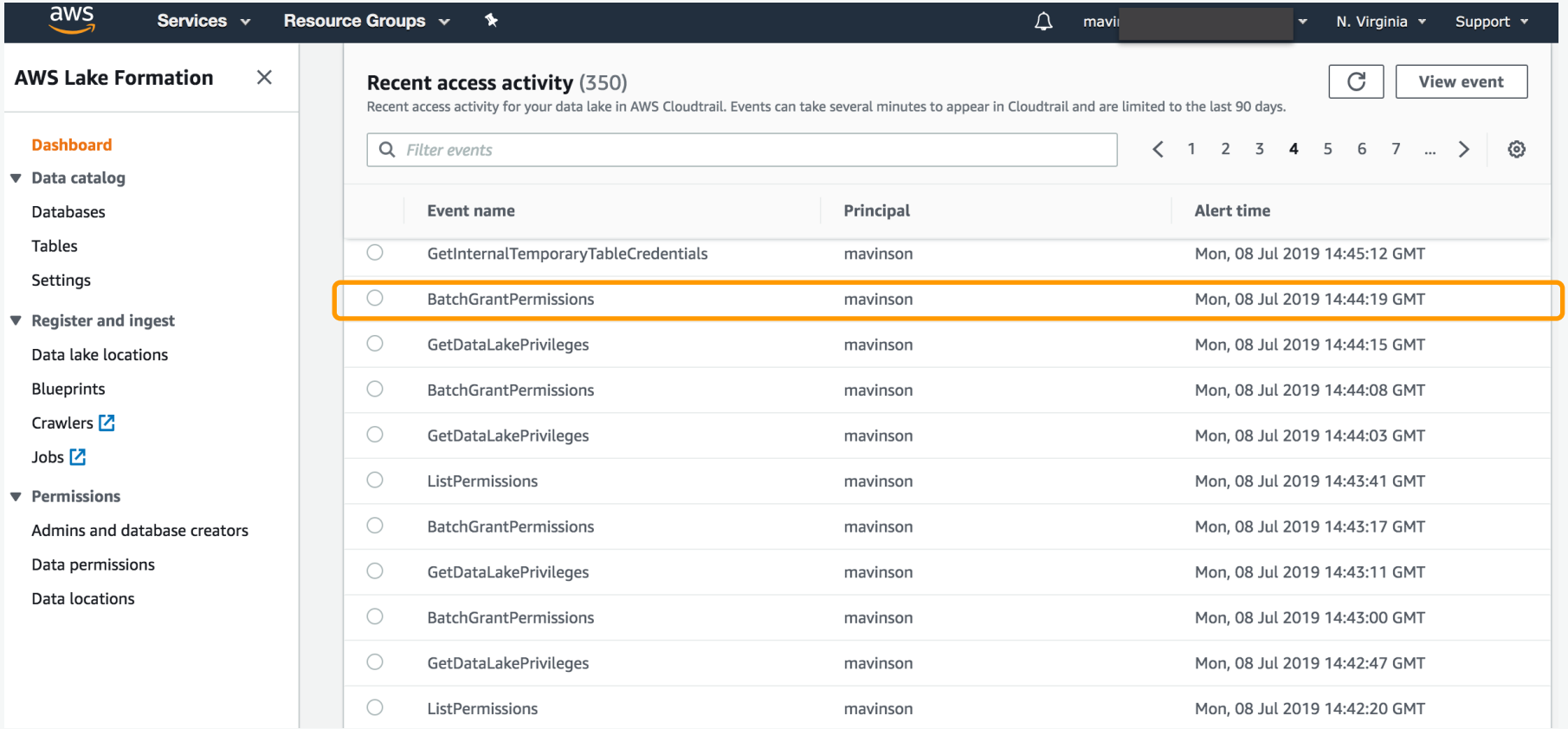
aws

Grant Privileges to Lake Formation Tables





Audit and Monitor in Real Time



aws

Audit and Monitor in Real Time

Event details


```
"eventTime": "2019-07-08T14:44:19Z",
"eventSource": "lakeformation.amazonaws.com",
"eventName": "BatchGrantPermissions",
"awsRegion": "us-east-1",
"sourceIPAddress": "72.21.196.66",
"userAgent": "signin.amazonaws.com",
"requestParameters": {
    "catalogId": "535356700373/datalake",
    "entries": [
        {
            "permissions": [
                "INSERT",
                "ALTER",
                "DROP",
                "DELETE"
            ],
            "permissionsWithGrantOption": [
                "INSERT",
                "ALTER",
                "DROP",
                "DELETE",
                "ALL"
            ],
            "principal": {
                "dataLakePrincipalIdentifier": "arn:aws:iam::535356700373:user/mavinson"
            },
            "resource": {
                "table": {
                    "name": "june06_mvdb_master_ecommerce",
                    "databaseName": "mv-data-lake-db"
```

Customer interest

"We are very excited about the launch of AWS Lake Formation, which provides a **central point of control** to easily load, clean, secure, and catalog data from thousands of clients to our AWS-based data lake, **dramatically reducing our operational load**. … Additionally, AWS Lake Formation will be **HIPAA compliant** from day one …"

Aaron Symanski, CTO, Change Healthcare

"I can't wait for my team to get our hands on AWS Lake Formation. With an enterprise-ready option like Lake Formation, we will be able to **spend more time deriving value from our data** rather than doing the heavy lifting involved in manually setting up and managing our data lake."

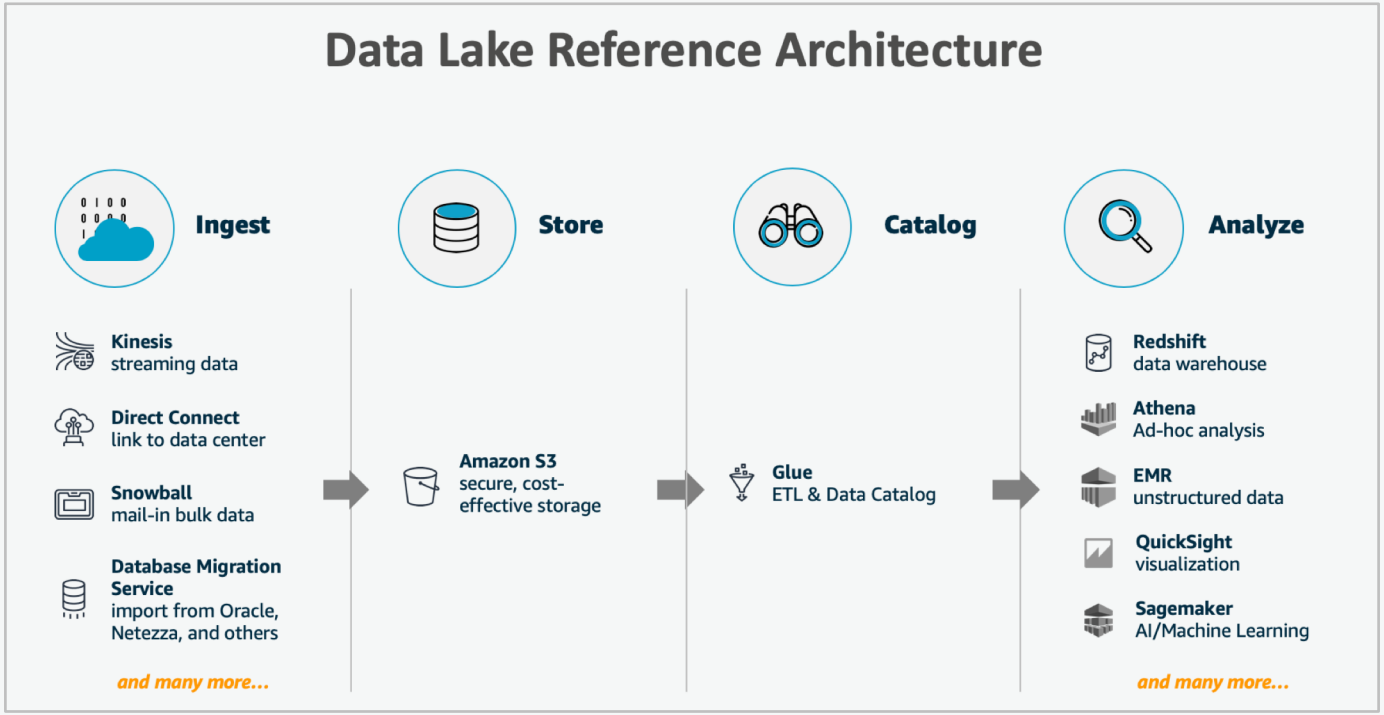
Joshua Couch, VP Engineering, Fender Digital

aws

Thank you!

Populating the Data Lake/Warehouse

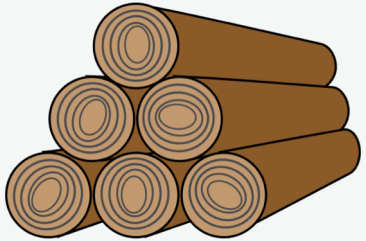
Data Ingestion



Data Sources

Databases

Streams



Logs

Files

aws

Data Sources - Files

Files

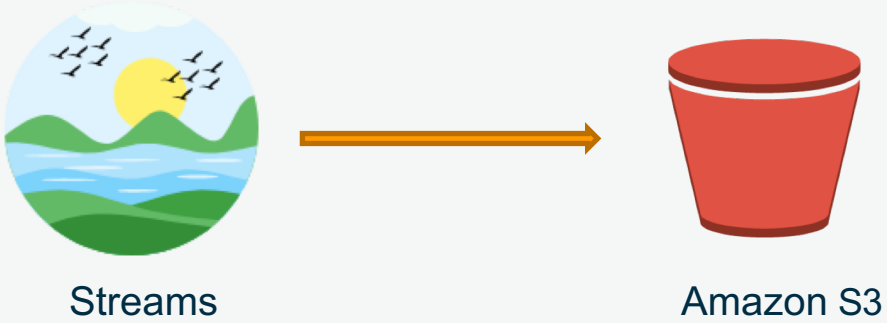
Optimizing Transfers

- S3 Multi-Part Upload
- S3 Transfer Acceleration
- AWS Direct Connect

Available Services

- AWS DataSync
- AWS Transfer - SFTP
- AWS Snowball/Snowmobile

Data Sources - Streams

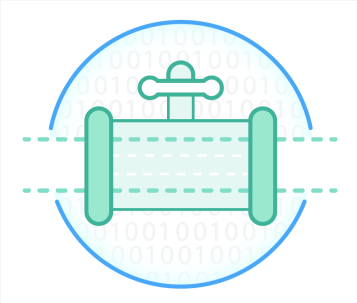


Streams

Collecting and Analyzing

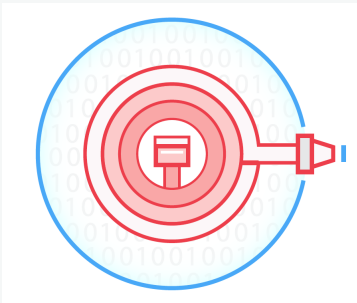
- Amazon Kinesis
- Amazon Managed Streaming for Kafka (MSK)
- Example: Clickstream Analytics

Amazon Kinesis - Stream Processing on AWS



Streams

- Capture streaming data for downstream processing
- Allow multiple processors to read streams at their own rate



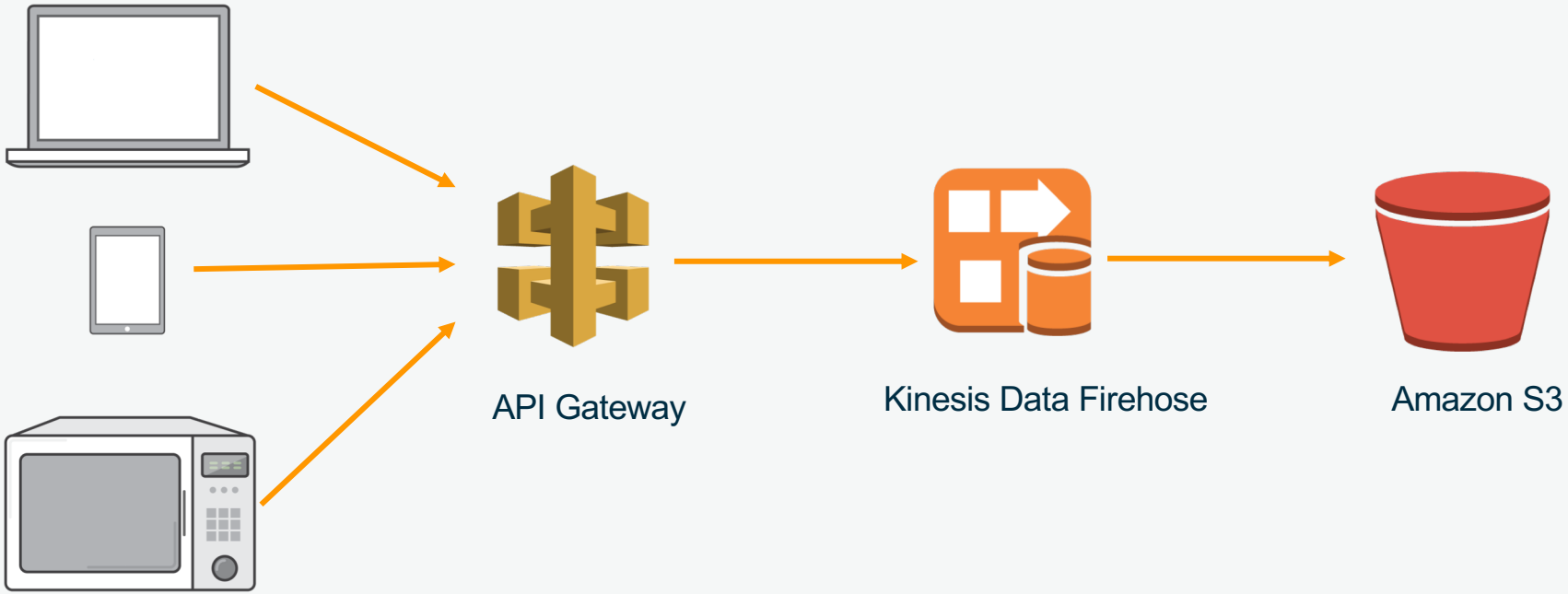
Firehose

- Buffer records in a stream into a single output for more efficient storage
- Automatic flushing of buffer to S3, ElasticSearch, Redshift, or Splunk

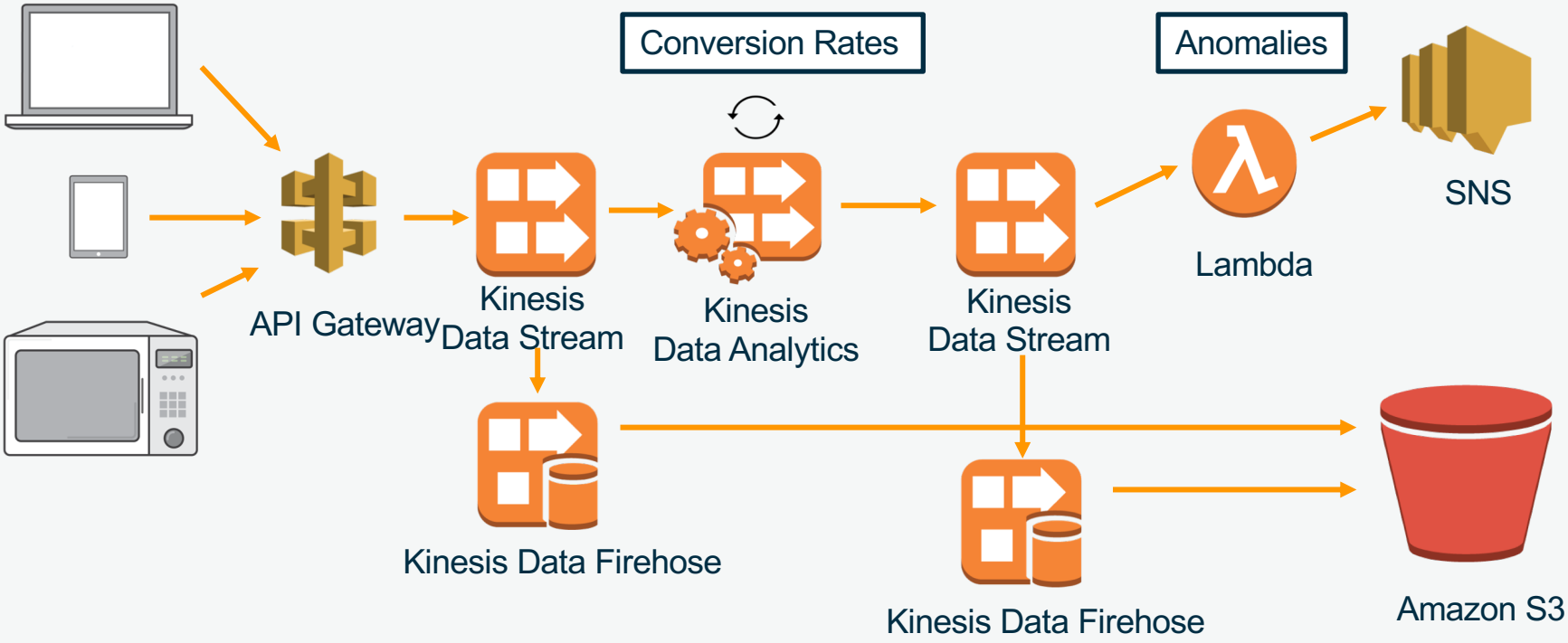
Analytics

- Create time windows over streams and perform aggregate operations using SQL
- Join together multiple streams and output to new streams

Clickstream

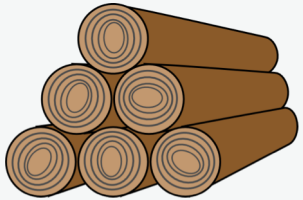


Clickstream with Real-Time Analytics



Data Sources - Logs

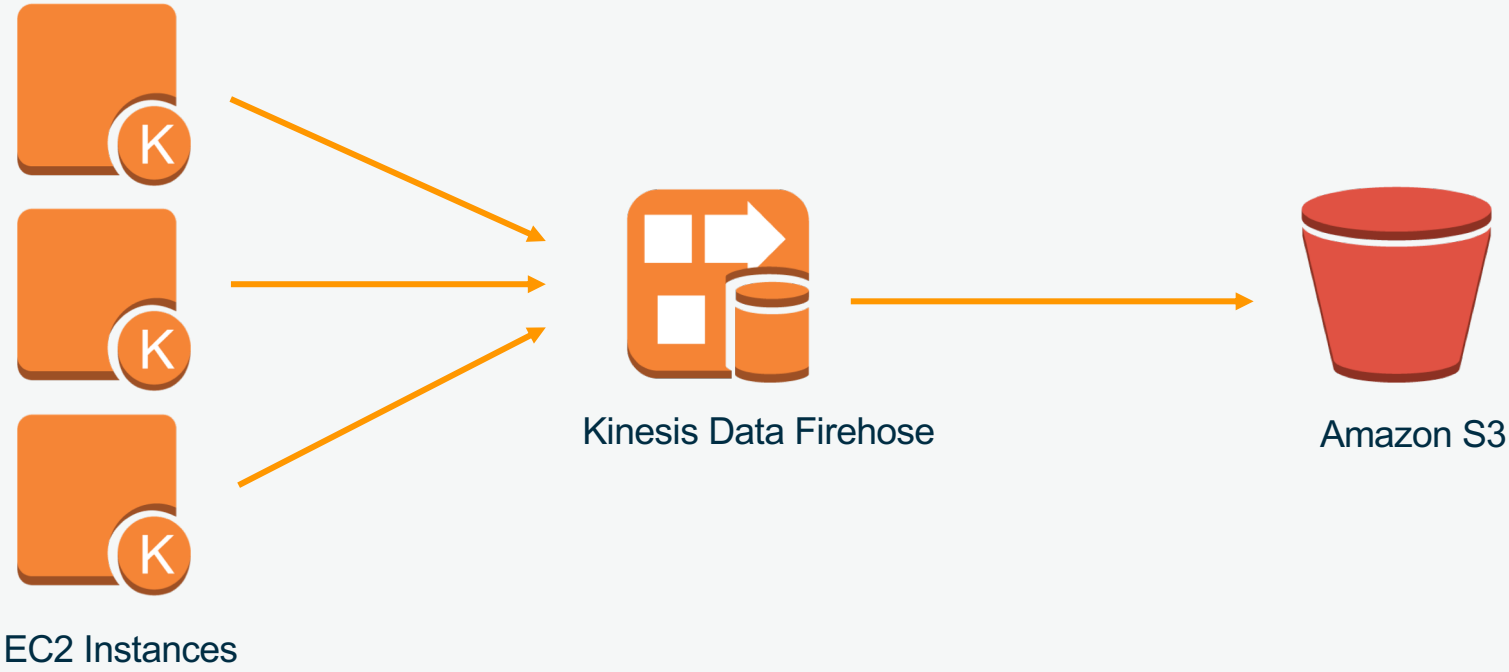
Logs



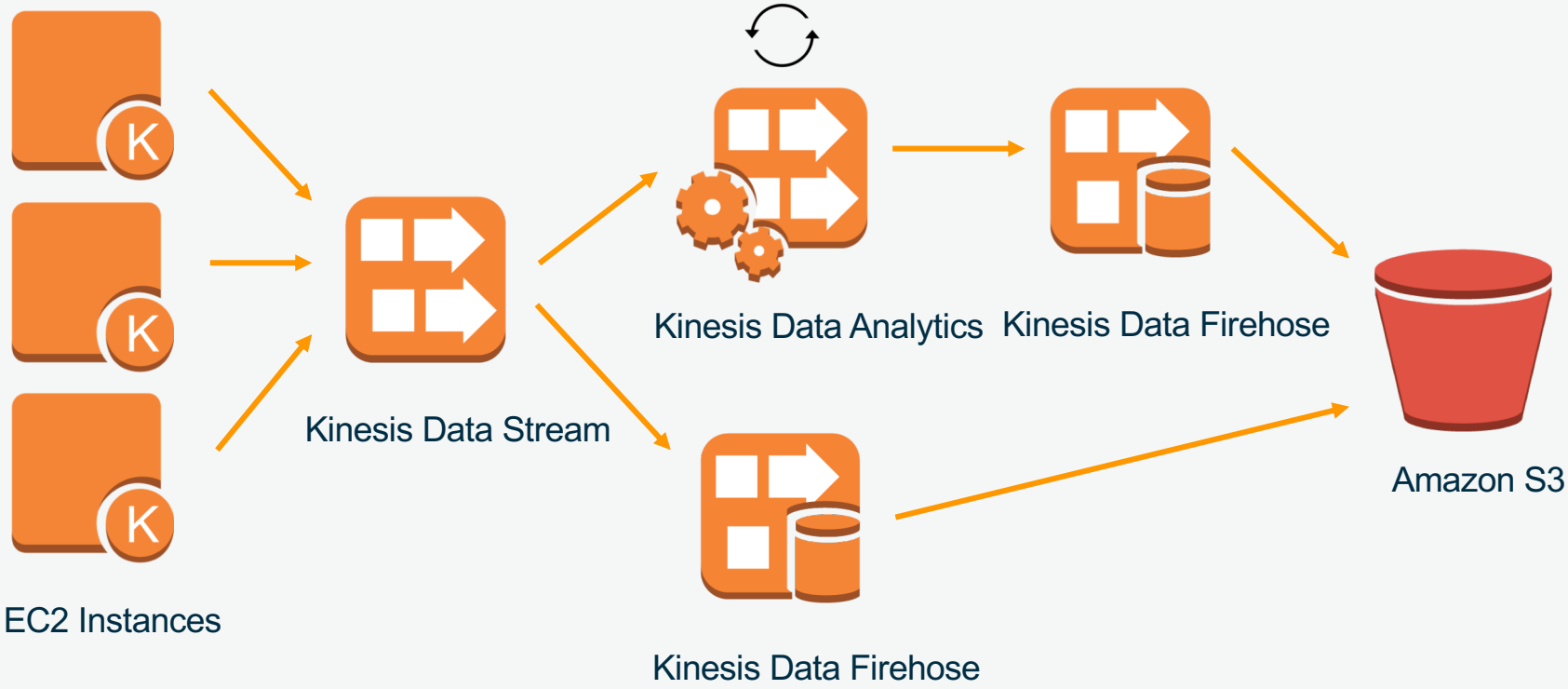
Collecting and Analyzing

- Amazon CloudWatch
- Amazon Kinesis
- Other Options

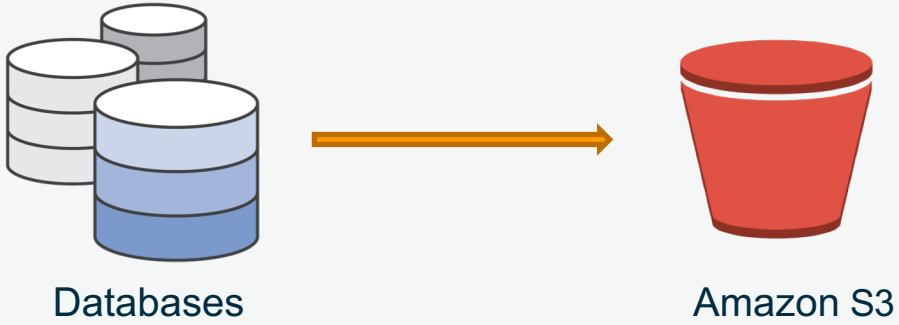
Logs – Kinesis Agent



Logs – Kinesis Agent (with Analytics)

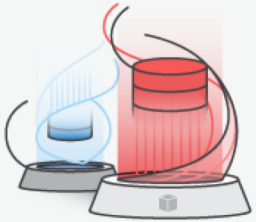


Data Sources - Databases



Database Migration Service

AWS Database Migration Service (AWS DMS) easily and securely migrate **and/or replicate your databases** *and* data warehouses to AWS



AWS Schema Conversion Tool (AWS SCT) convert your commercial database and data warehouse schemas to open-source engines **or AWS-native services**, such as Amazon Aurora and Redshift

aws

When to use DMS and SCT?

Modernize

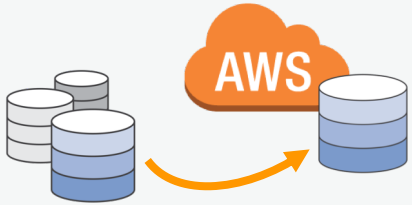
Modernize your database tier –

- Commercial to open-source
- Commercial to Amazon Aurora

Modernize your Data Warehouse –

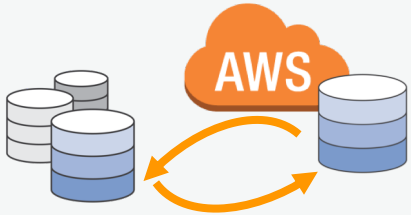
- Commercial to Redshift

Migrate



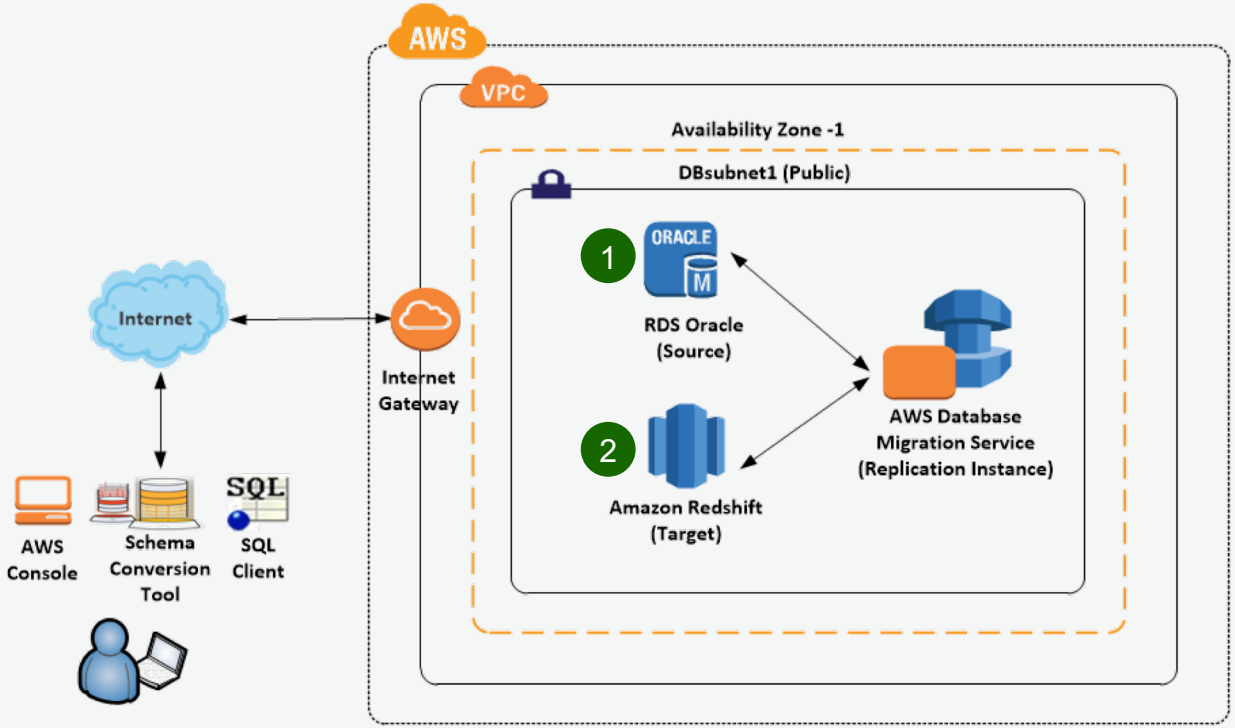
- Migrate business-critical applications
- Migrate from Classic to VPC
- Migrate data warehouse to Redshift
- Upgrade to a minor version

Replicate



- Create cross-regions Read Replicas
- **Run your analytics in the cloud**
- Keep your dev/test and production environment sync

SCT/DMS Demonstration Architecture



DMS Data Lake Use-Case: S3 as a Target

Bulk File

```
s3://mybucket/schemaName/tableName
s3://mybucket/hr/employee

/schemaName/tableName/LOAD001.csv
/schemaName/tableName/LOAD002.csv
/schemaName/tableName/LOAD003.csv
...
101,Smith,Bob,4-Jun-14,New York
102,Smith,Bob,8-Oct-15,Los Angeles
103,Smith,Bob,13-Mar-17,Dallas
104,Smith,Bob,13-Mar-17,Dallas
```

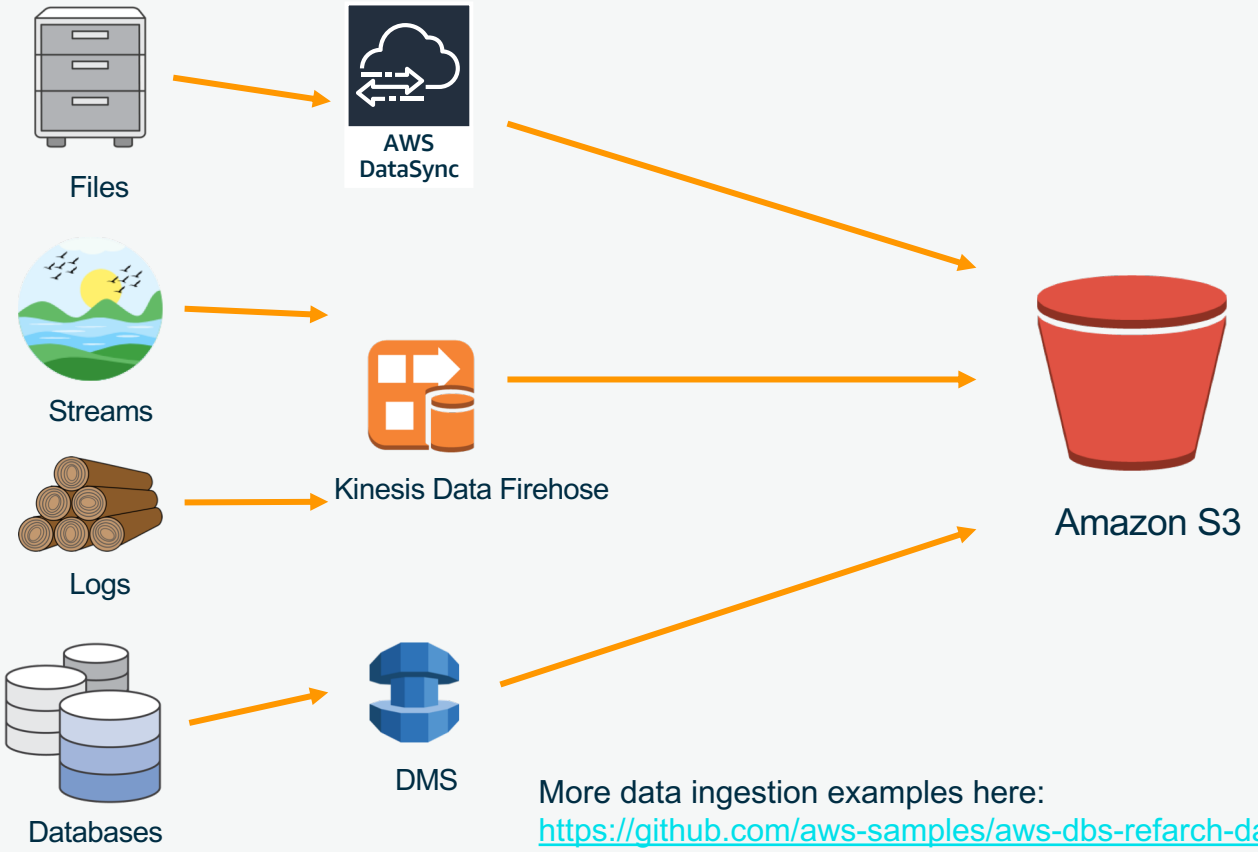
Ongoing CDC Files

```
s3://mybucket/schemaName/tableName

<time-stamp>.csv
<time-stamp>.csv
<time-stamp>.csv
...

I,101,Smith,Bob,4-Jun-14,New York
U,101,Smith,Bob,8-Oct-15,Los Angeles
U,101,Smith,Bob,13-Mar-17,Dallas
D,101,Smith,Bob,13-Mar-17,Dallas
```

Summary - Ingestion



s3://datalake/

- /vendorfeeds
  - /vendorA
  - /vendorB
- /clickstream
  - /orders
  - /vendors
  - /customers
- /app_logs
  - /instance1
  - /instance2
- /syslogs
  - /instance1
  - /instance2
- /databases
  - /customers
  - /orders
  - /vendors

More data ingestion examples here:
https://github.com/aws-samples/aws-dbs-refarch-datalake

Architecture Examples

Nasdaq

Nasdaq operates financial exchanges around the world, and processes large volumes of data.

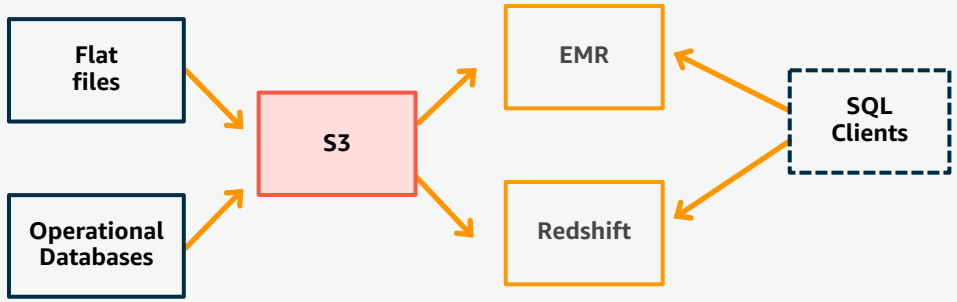
Challenge:

Nasdaq wanted to make their large historical data footprint available to analyze as a single dataset.

Solution:

- Use Amazon Redshift for interactive querying
- Use Amazon S3 as a Data Lake, and Presto on EMR to process historical data

Nasdaq Uses AWS to Build a Data Lake



Data from all 7 exchanges operated by Nasdaq
(orders, quotes, trade executions)

- Migrate legacy on-premises warehouse to Amazon Redshift
- 4.8B rows inserted per trading day (orders, trades, quotes)
- Ingest data from multiple sources, validates, and stages in S3
- Redshift reads data out of S3 for fast queries
- Presto on EMR and S3 used for analysis of massive historical data set

Good things come from Sysco

Sysco is the leader in selling, marketing, and distributing food.

Challenge:

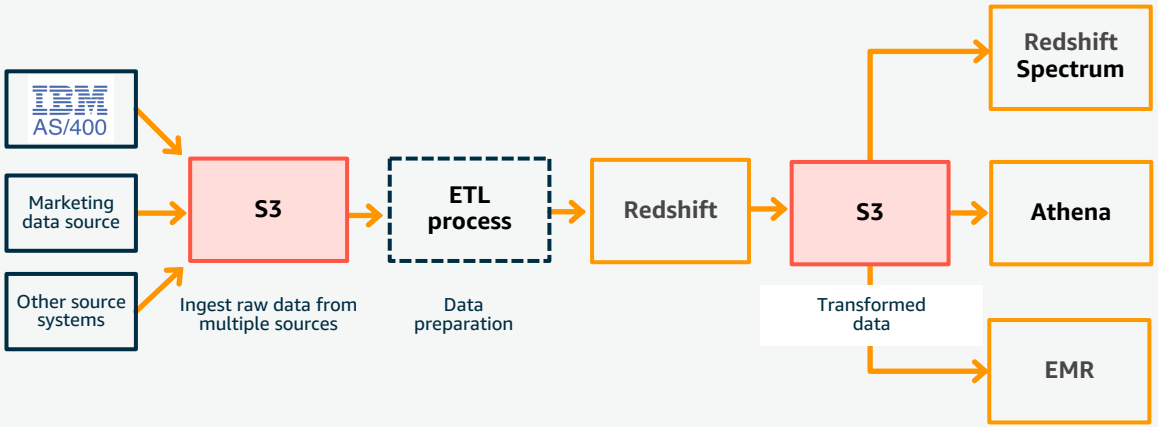
Large volumes of data in multiple systems. Also, high costs from maintaining on-premises EDW deployment.

Solution:

- Migrated their on-premises solution to the cloud with Redshift, S3, EMR, and Athena

Analytics on the Data Lake

Good things come from Sysco



- Sysco is the leader in selling, marketing, & distributing food
- Challenge: large volumes of data in multiple systems
- Consolidated data into a single S3 data lake
- Data scientists use EMR notebooks, Athena & Amazon Redshift Spectrum used by business users for reporting

aws

21ST CENTURY FOX

Fox Film Entertainment is one of the largest movie studios in the world.

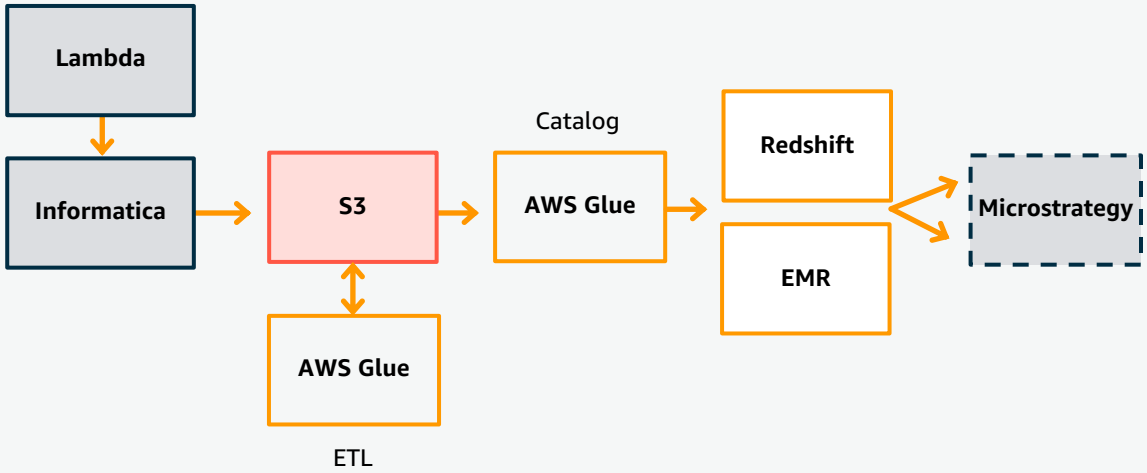
Challenge:

Processes 100+ of TB of data a day, 25k+ queries per day. They had a legacy data platform that struggled to scale, faced many outages, and had changing requirements from consumers who had many options.

Solution:

- Land data in S3 as a data lake
- Use Redshift and EMR as analytics engines to process data
- Saved 15–20% in overall costs (on-premises) with 30% of performance gain

21st Century Fox Uses AWS for Data Lakes & Analytics



- Collect and ingest data with AWS Lambda for serverless scale and Informatica for data ingestion and transformation
- AWS Glue does ETL on data in S3 and provides a data catalog
- Redshift and EMR used as analytics engines
- Microstrategy used as a visualization tool

amazon.com

Amazon.com's vision is to be the earth's most customer—centric company; where people can find anything they want to buy online.

Challenge:

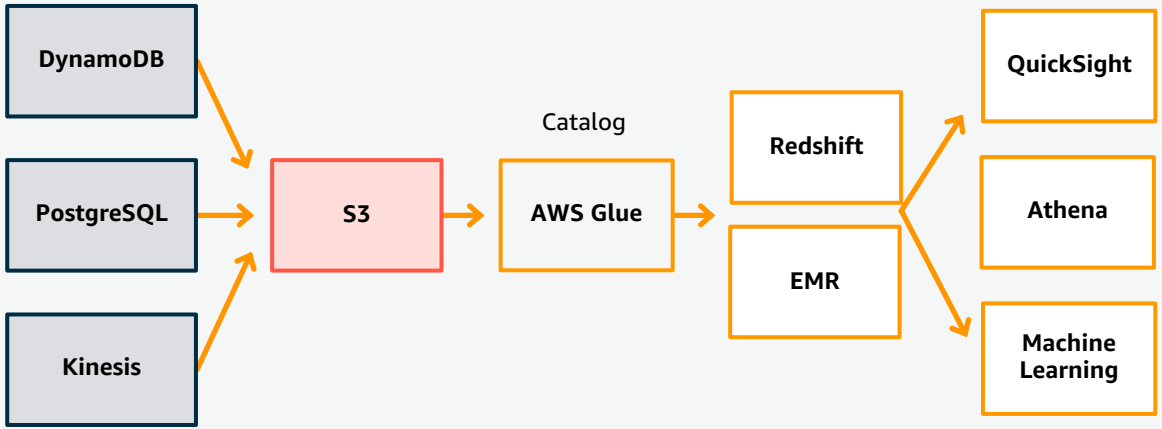
Load 500K+ transactions each day, and serve 300K+ queries/extracts each day from Amazon businsses (Amazon.com, Amazon Prime, Amazon Music, Amazon Alexa, Amazon Video, and Twitch).

Solution:

- Land data in S3 as a data lake
- Use Redshift as preferred SQL based analysis by business users, and EMR for machine learning

Amazon.com Uses AWS for Data Lakes & Analytics

amazon.com

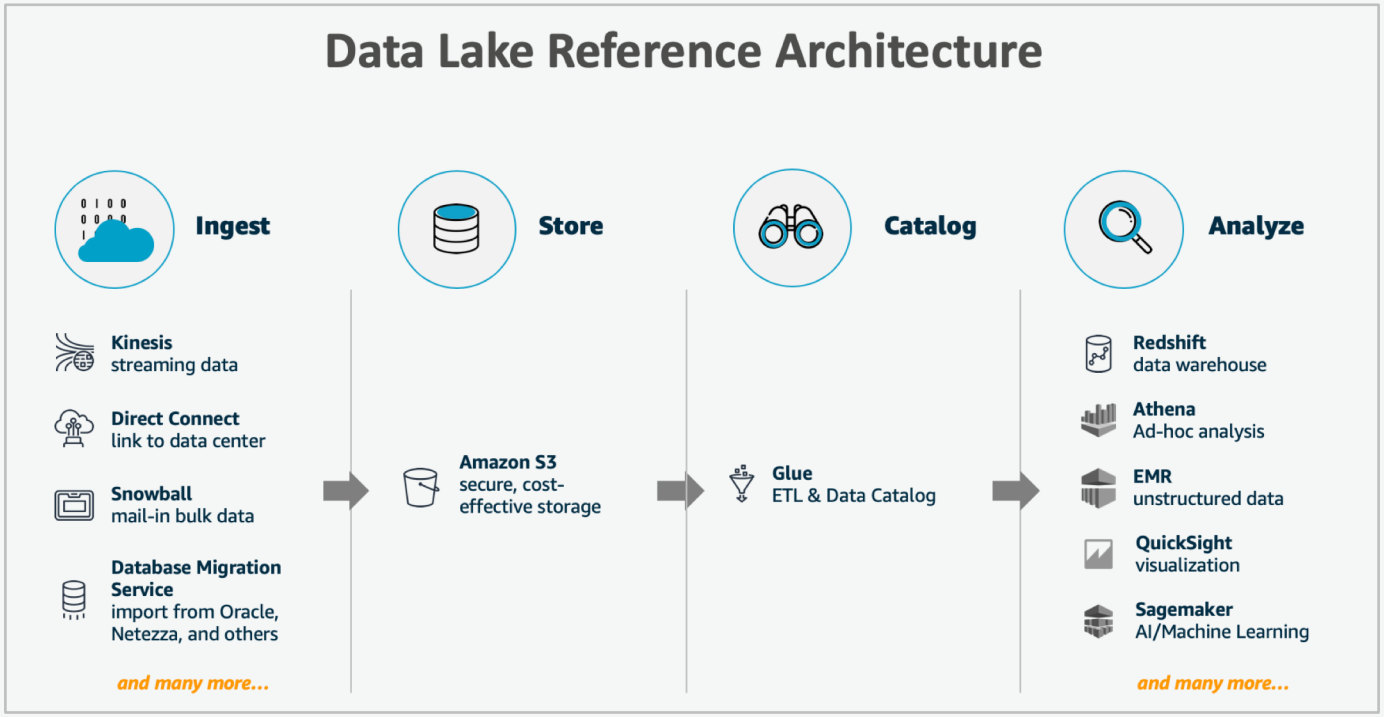


- DynamoDB capturing all Amazon.com transactions
- Everything from DynamoDB, RDS PostgreSQL and Kinesis fed to a S3 data lake
- Glue used to catalog the data
- Redshift used for all SQL-based queries, and EMR for all machine learning and big data processing
- End-users use QuickSight for visualizations

aws

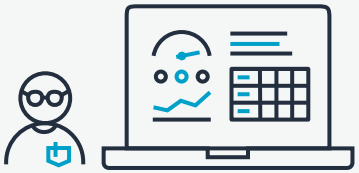
Visualizing Data

Amazon QuickSight




QuickSight
Amazon QuickSight is a fully managed, serverless, cloud business intelligence system

Who Is QuickSight For?



Enterprise BI

Easily share interactive dashboards and insights across your organization at scale.

Developers/ISVs

Embed rich interactive analytics into your Own applications, websites and portals.

Analysts

Allow self-serve power users to easily connect to and explore their data.

Why QuickSight?

No Servers to Manage

QuickSight is a fully managed cloud service. There is no infrastructure to maintain or upgrade and no upfront costs.

Pay For What You Use

Instead of buying costly licenses for all of your users, QuickSight allows you to share dashboards and reports and only pay when users access them.

Scalable

From 10 users to 10,000, QuickSight seamlessly grows with you with no need for additional servers or infrastructure.

Fully integrated

QuickSight integrates with your other AWS services and data sources giving you everything you need to build an end-to-end cloud analytics solution.

Connect to your data, wherever it is

QuickSight allows you to connect to AWS data sources, Private VPC subnets, on-premises and hosted databases and third party business applications.

On-premises

Securely connect to on-premise databases and flat files like Excel and CSV

- Excel
- CSV
- Teradata
- MySQL
- SQL Server
- PostgreSQL

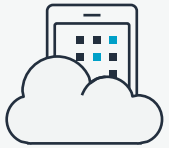
In the cloud

Connect to hosted database, big data formats, and secure VPCs

- Redshift
- RDS
- S3
- Athena
- Aurora
- Teradata
- MySQL
- Presto
- Spark
- SQL Server
- Postgre SQL
- MariaDB
- Snowflake
- IoT Analytics

Applications

Connect directly to third party business applications

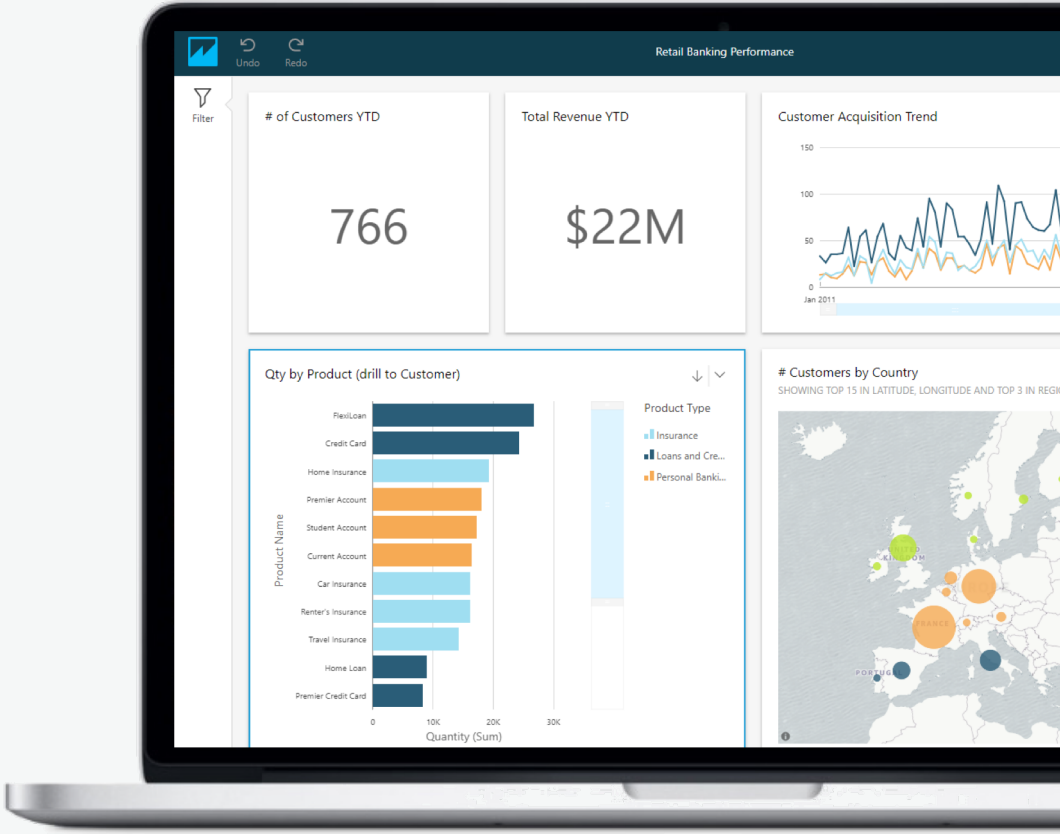


- Salesforce
- Square
- Adobe Analytics
- Jira
- ServiceNow
- Twitter
- Github

Demonstration

Create beautiful, interactive dashboards.

- Add rich interactivity like filters, drill downs, zooming, and more
- Blazing fast navigation
- Accessible on any device
- Data Refresh
- Publish to everyone with a click

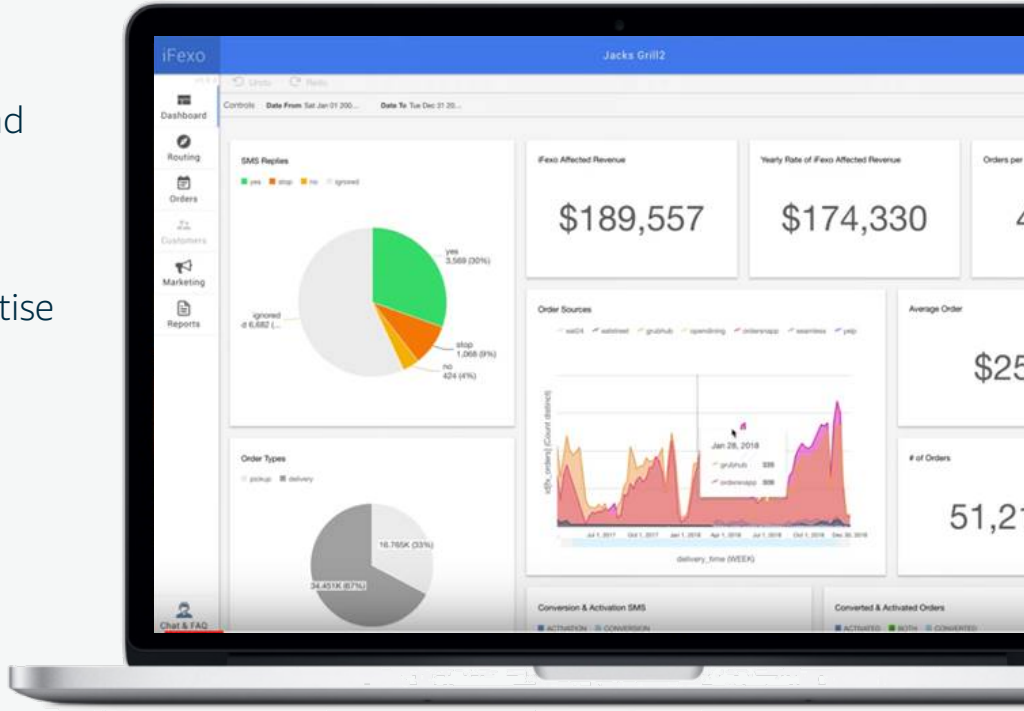


aws

Embedding Dashboards In Your Application

QuickSight allows you to seamlessly integrate interactive dashboards and analytics into your own applications

- Enhance your applications with rich analytics and dashboards
- Easy maintenance, no servers to manage
- Fast! No Custom development or domain expertise needed
- Leverage new features as we add them
- Utilizes Pay-per-Session Pricing.

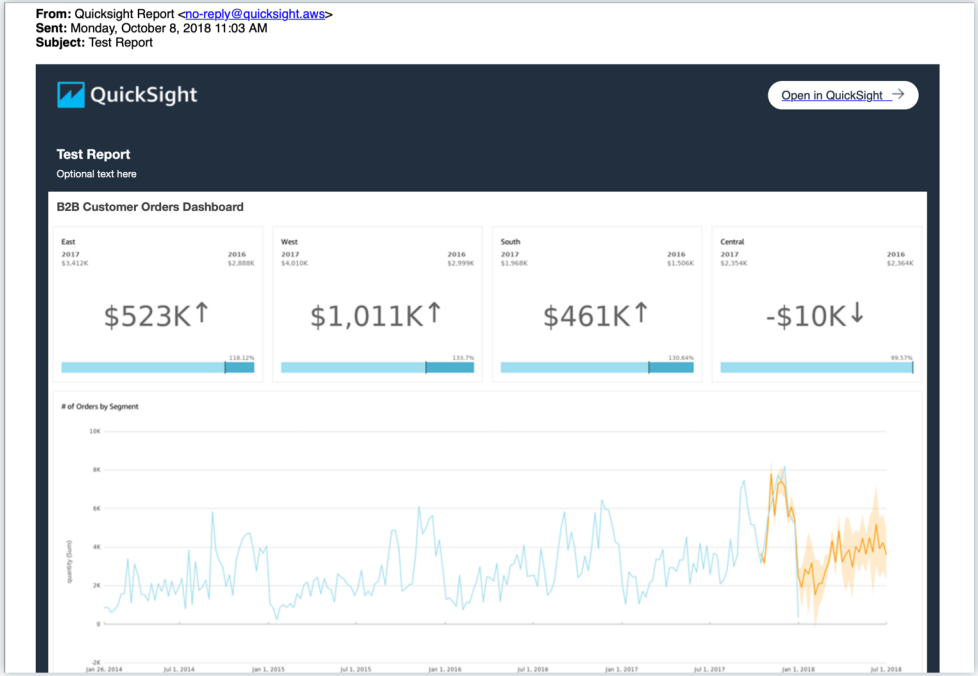


aws

Insights Delivered to Your Inbox

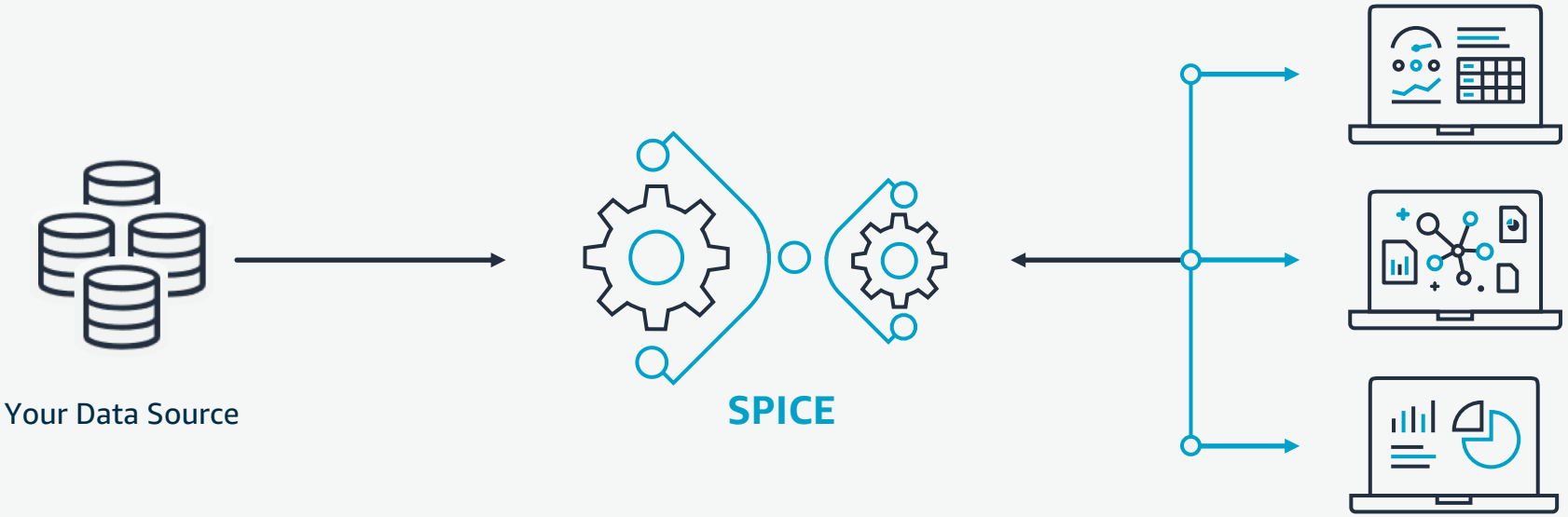
QuickSight lets you send report snapshots directly to your users inbox.

- Schedule email reports on a daily, weekly, or monthly basis
- Sent to individual users or groups
- Users can opt out of any report so they can focus on what's important.
- Uses Pay-per-Session Pricing



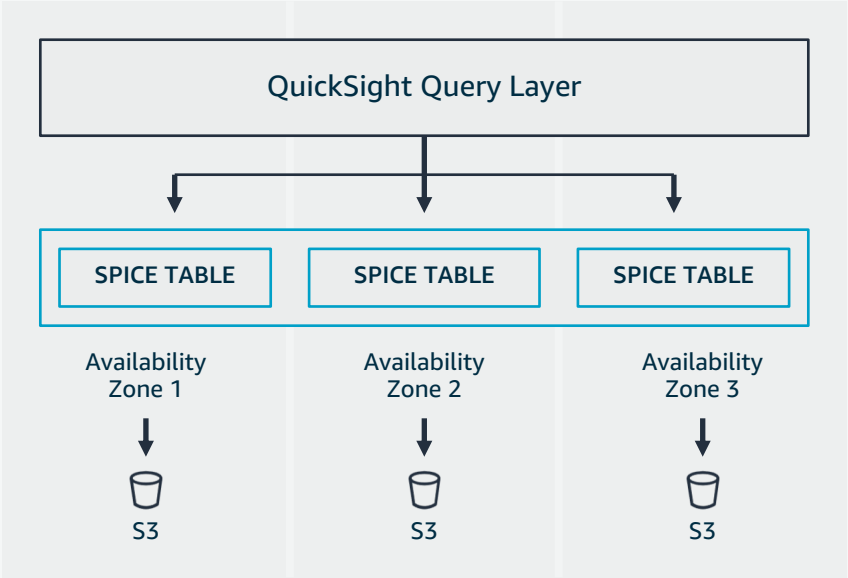
SPICE

QuickSight is powered by SPICE, a super-fast in-memory calculation engine that delivers performance and scale, regardless of how many users are active.



More About SPICE

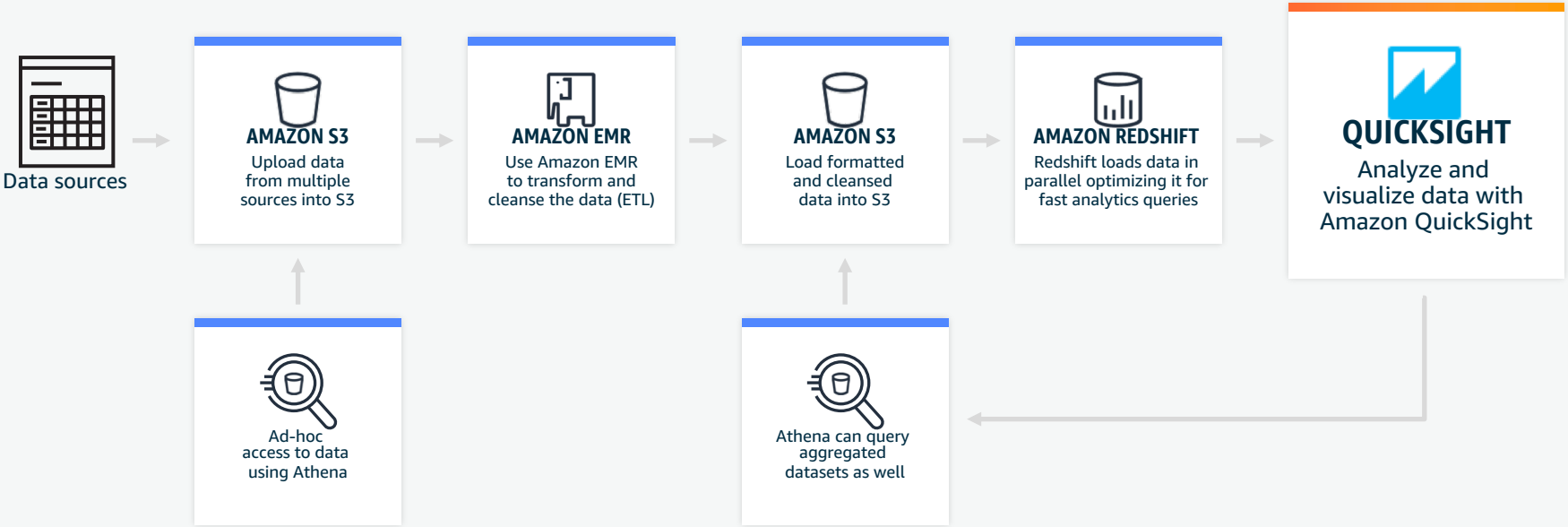
Storing your data in SPICE protects your underlying data sources from end user activity, guaranteeing performance and saving you money.



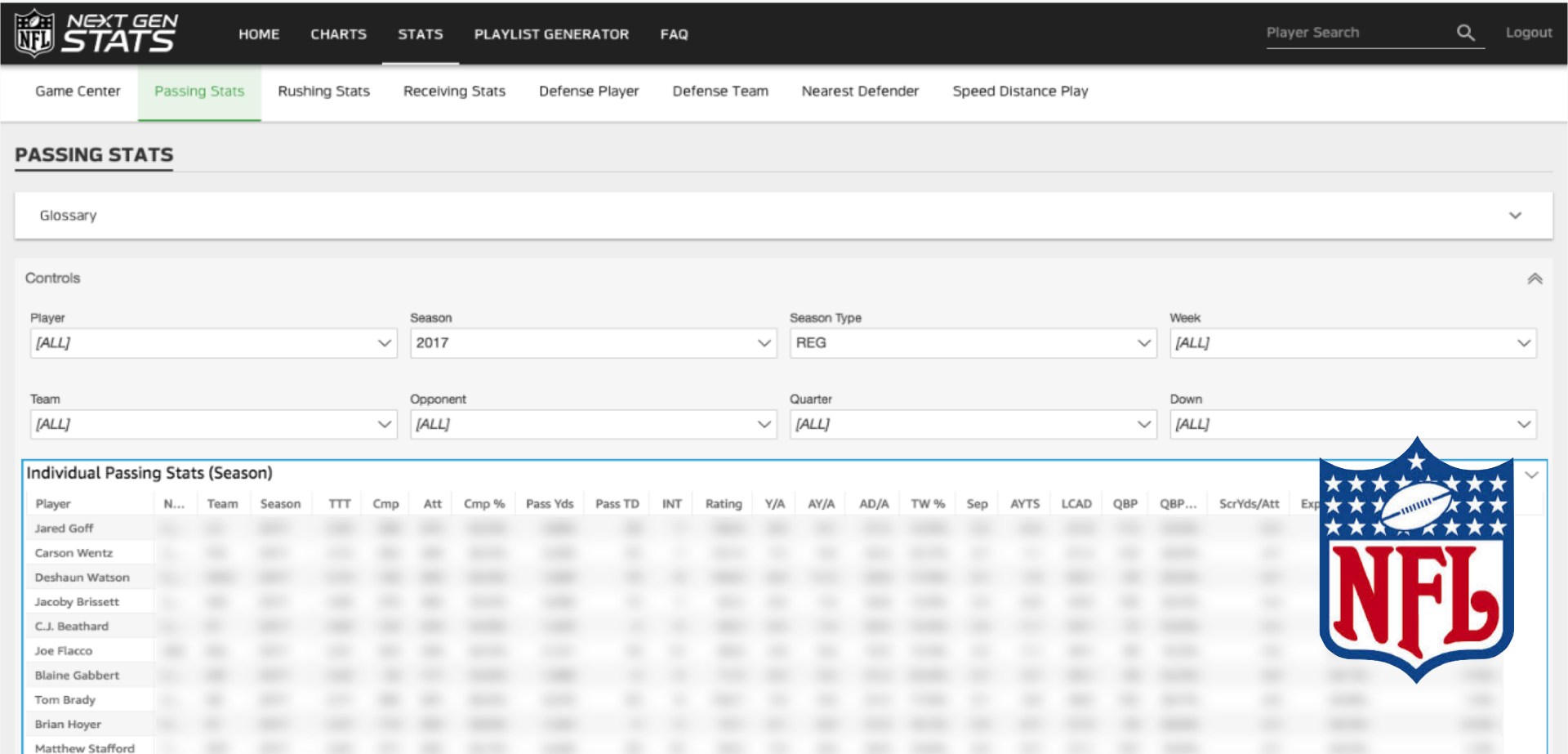
- Up to 10X faster (millisecond latency)
- Fault-tolerant, self-healing
- Support for high concurrency
- Backed up in S3 (Write Ahead Log)
- Instant failover with zero impact

aws

ETL, Data Warehousing and Analytics with AWS



Use Case: NFL Next Gen Stats



Customers

AutoTrader

Customer name:
Auto Trader Group plc.

Company description:
Automotive marketplace

Data source:
S3/Athena

Use case:
50+ users and growing

Previous tools:
Spreadsheets and static graphs

Auth:
SAML-based SSO

Use case:

- Provide business dashboards to multiple teams from S3 data lake
- Equip analysts with easy, self-service dashboard building capability. Eliminate need to manually curate, process and build spreadsheets, saving 10% of an analyst's monthly efforts.

"Timely access to relevant data is the key to success in our business. Amazon QuickSight has enabled our analysts to move from static graphs to fast, interactive dashboards automatically updated with the latest data. QuickSight's native integration with Amazon Athena has allowed faster time to analysis and saved us effort involved in additional data curation. With the new QuickSight readers, we can extend interactive dashboards to the entire team, and Pay-per-Session pricing assures us we only pay when we use the product. The serverless nature of QuickSight aligns with our vision for the data platform, with no infrastructure management needed to scale deployments across the company! We look forward to enabling more use cases in QuickSight."

Jim Stamp, Head of Product – Data Engineering

aws

QuickSight

As a native cloud service, QuickSight combines the speed, scalability, and security that our customers have come to depend on with the value and cost effectiveness you expect from AWS

Native AWS service

No server licensing or maintenance costs

Pay-as-you-go

Scalable, fast, easy

No deployment time

Try it free Today @

Quicksight.AWS

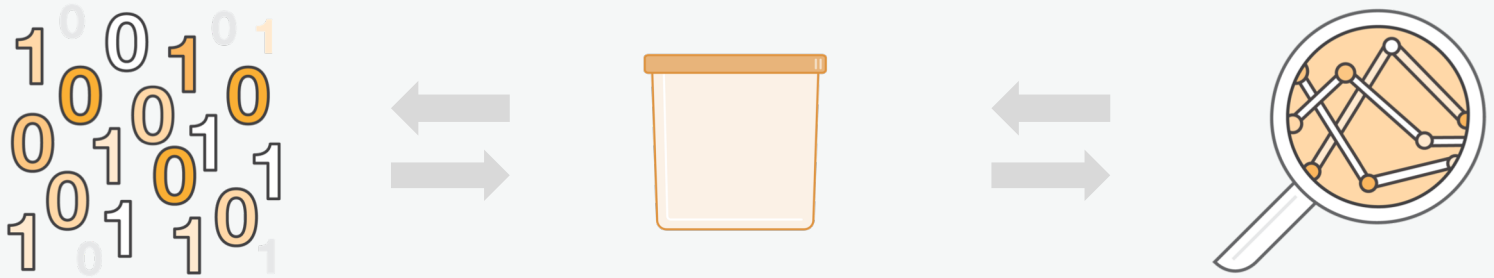
aws

Wrap-up

Feedback survey: http://bit.ly/2mjYBcO

Summary

AWS helps unlock Data in three ways



Future-proofed for new data types

Ingest **new data for differentiated experiences:** image/video, social media, IoT sensors, and more

Single source of truth (aka “data lake”)

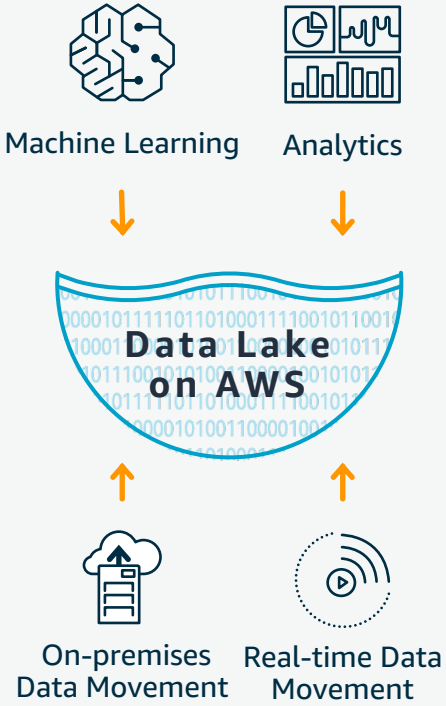
Secure and scalable backbone for all incoming data and all analytics – **no more data siloes**

The right analytics tool for every job

New analytics possibilities – machine learning, natural language processing, Hadoop-as-a-service, and more

Summary

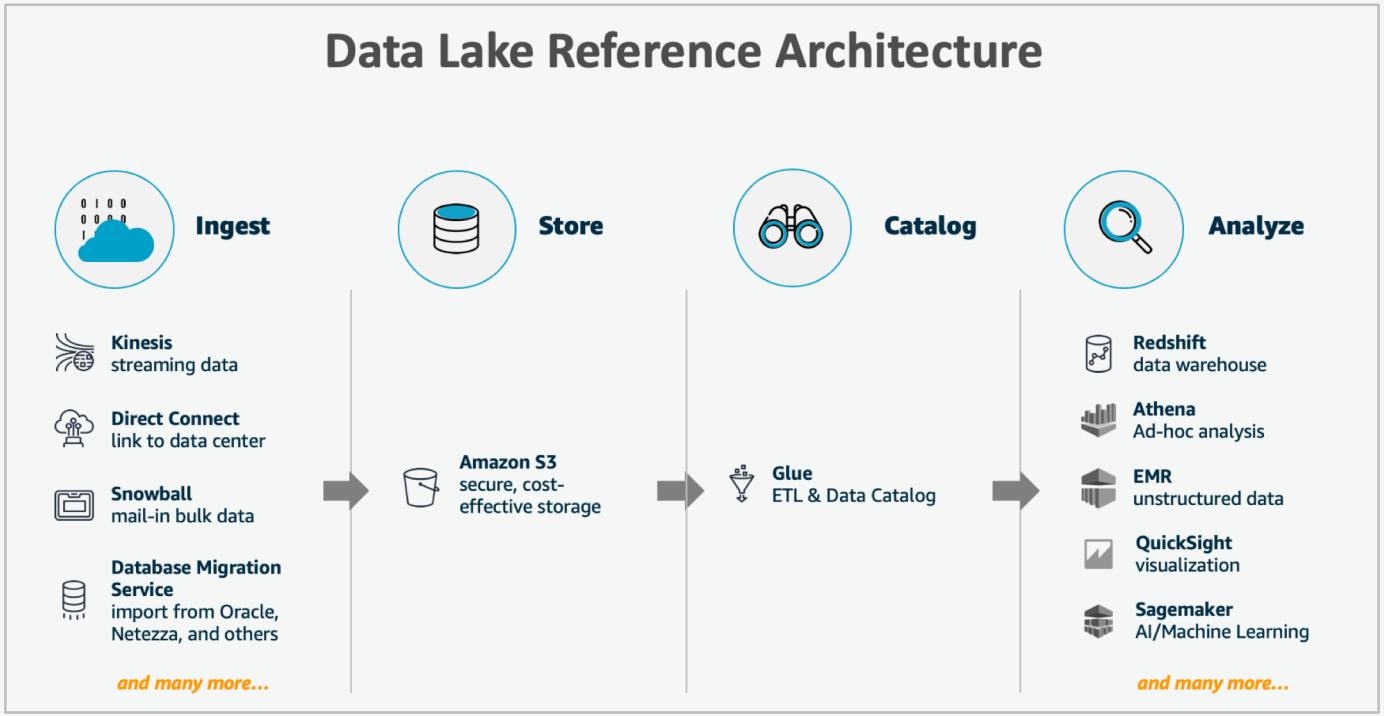
Benefits of Data Lakes from AWS



- Open and comprehensive
- Secure
- Scalable and durable
- Lowest cost

More Data Lakes & Analytics on AWS than Anywhere Else

Thank you!



Thank you!

Survey: http://bit.ly/2mjYBcO