



AWS GLUE

AWS GLUE

AWS Glue automates the undifferentiated heavy lifting of ETL

➤ Discover

- Automatically discover and categorize your data making it immediately searchable and queriable across data sources

➤ Develop

- Generate code to clean, enrich, and reliably move data between various data sources; you can also use their favorite tools to build ETL jobs

➤ Deploy

- Run your jobs on a serverless, fully managed, scale-out environment. No compute resources to provision or manage.

AWS GLUE



Data Catalog



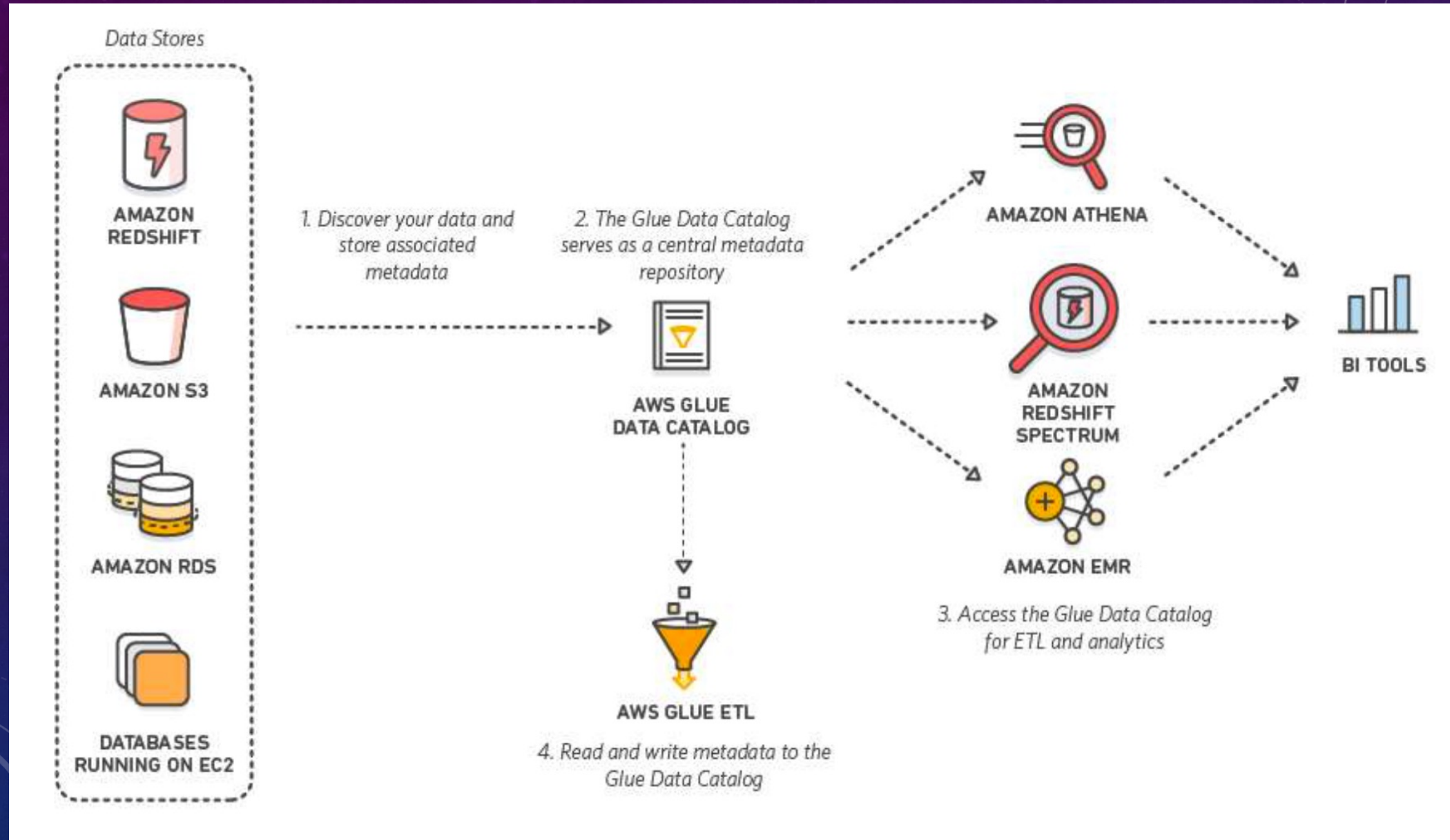
Job Authoring



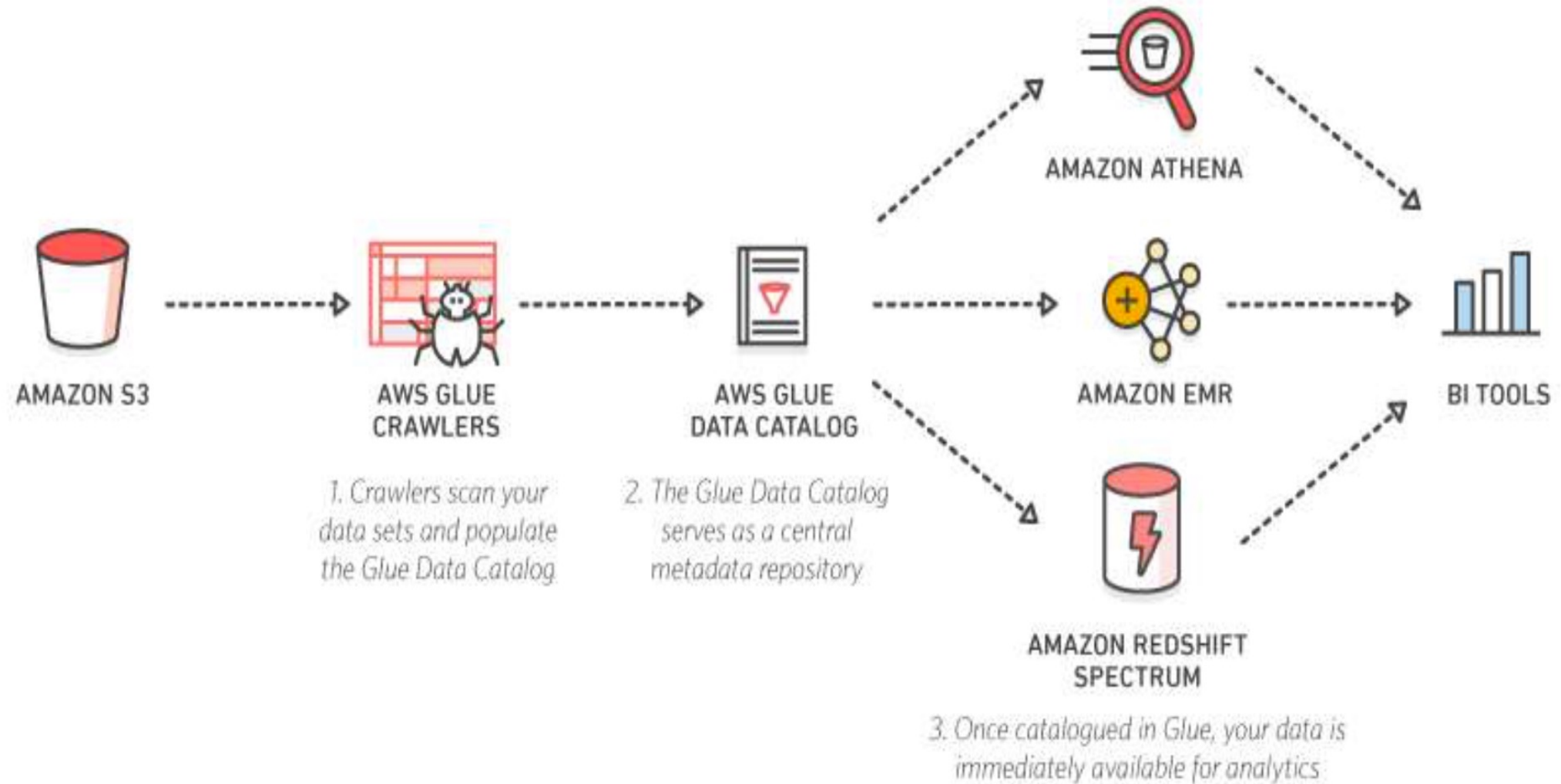
Job Execution

- Hive Metastore compatible with enhanced functionality
- Crawlers automatically extracts metadata and creates tables
- Integrated with Amazon Athena, Amazon Redshift Spectrum
- Auto-generates ETL code
- Build on open frameworks – Python and Spark
- Developer-centric – editing, debugging, sharing
- Run jobs on a serverless Spark platform
- Provides flexible scheduling
- Handles dependency resolution, monitoring and alerting

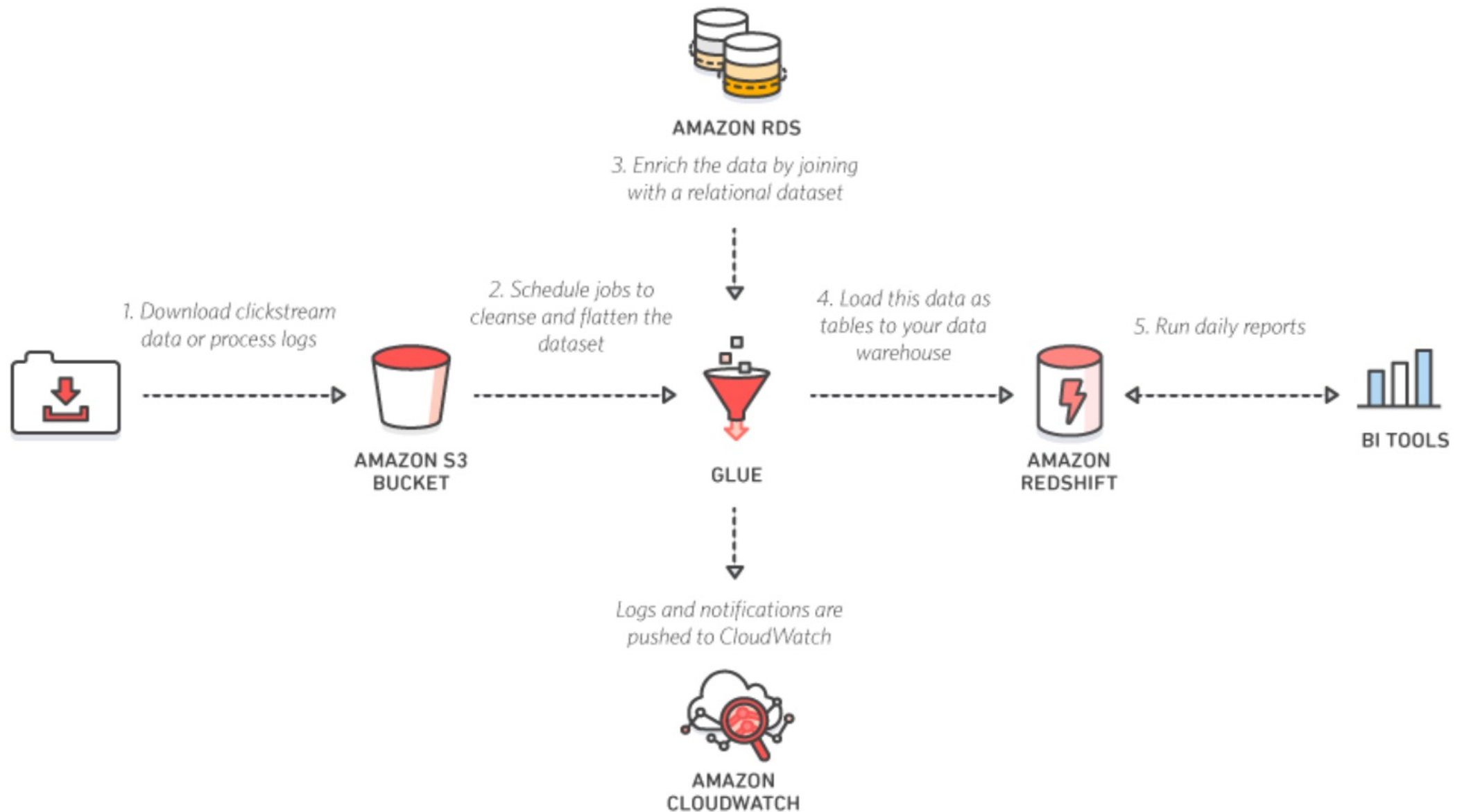
USE CASE 1: UNDERSTAND YOUR DATA ASSETS



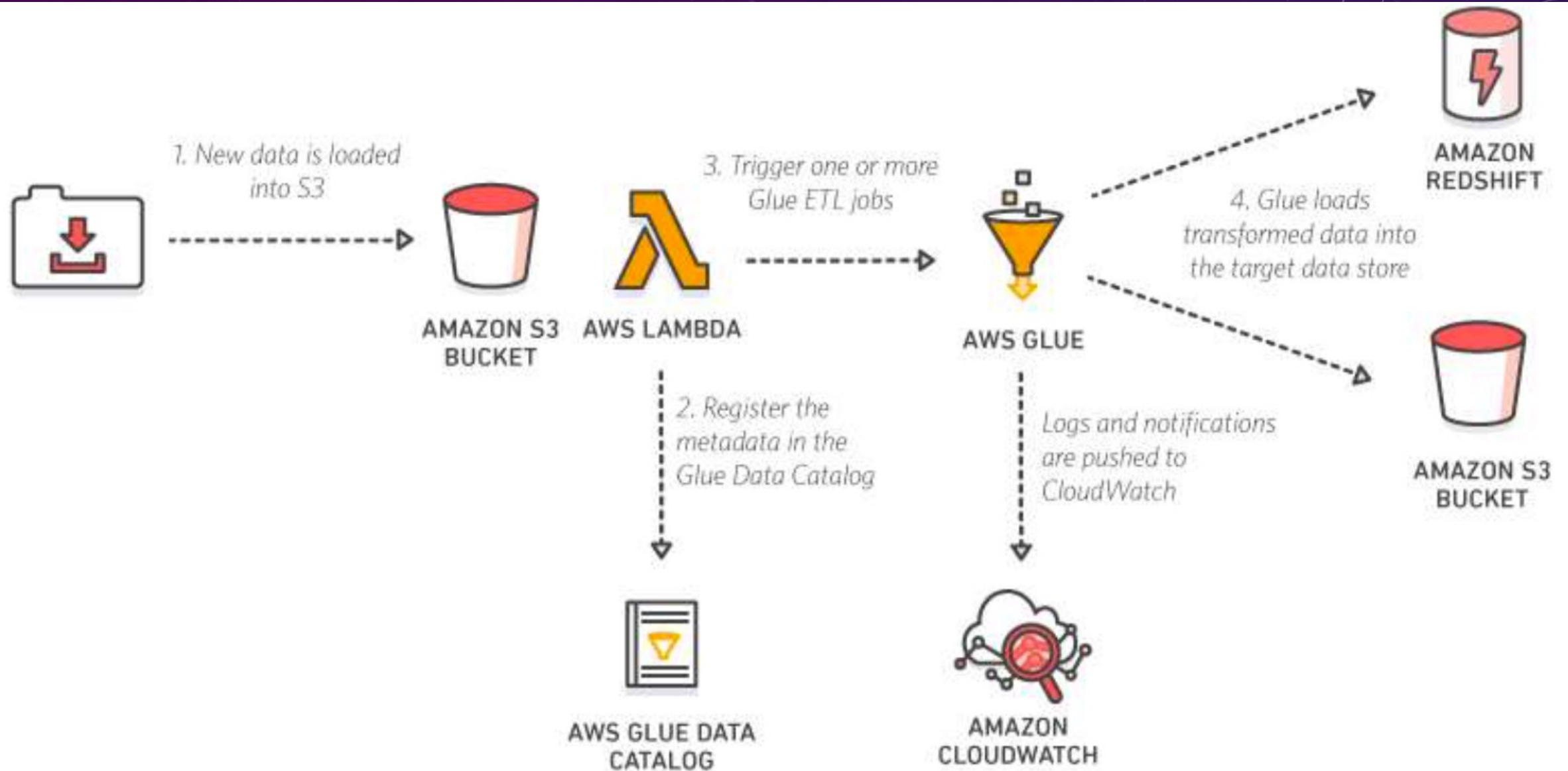
USE CASE 2: INSTANTLY QUERY YOUR DATA LAKE ON AMAZON S3



USE CASE 3: ETL DATA INTO YOUR DATA WAREHOUSE



USE CASE 4: BUILD EVENT-DRIVEN ETL PIPELINES



AWS GLUE – DATA CATALOG

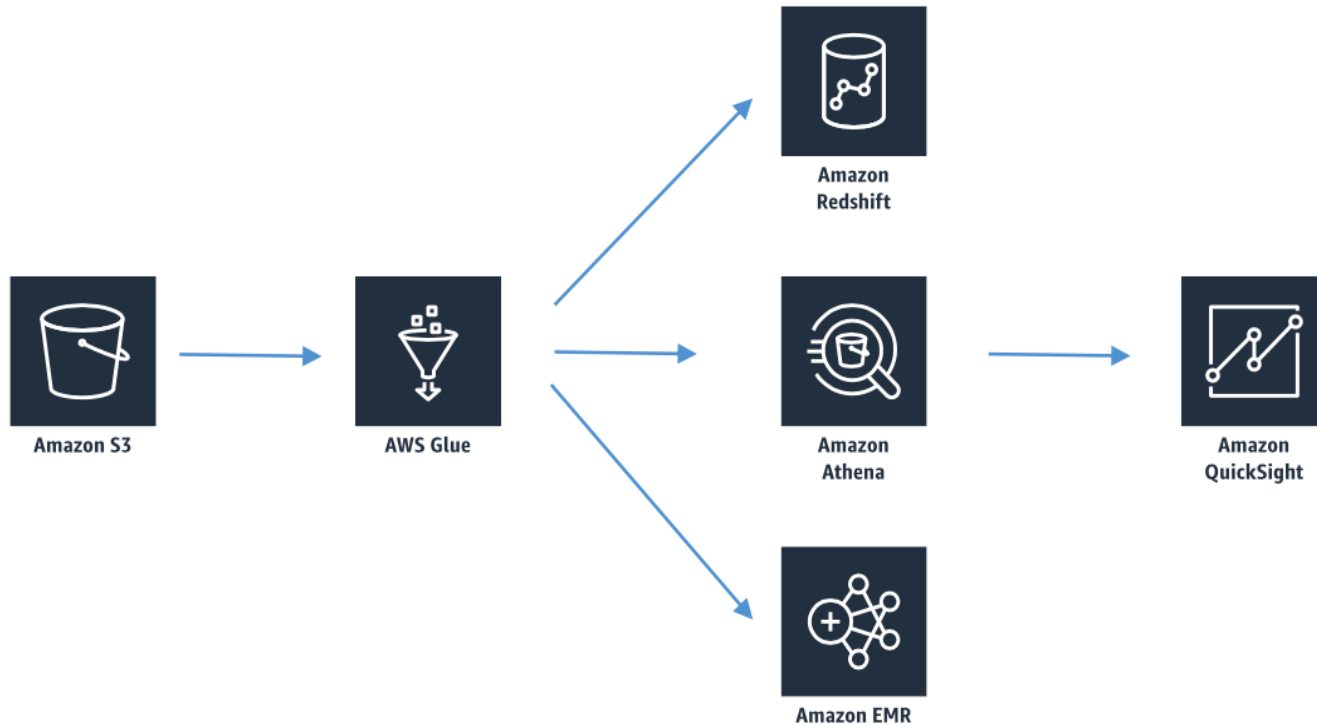
- Manage table metadata through a Hive meta-store API or Hive SQL. Supported by tools like Hive, Presto, Spark etc.
- Added a few extensions:
 - **Search** over metadata for data discovery
 - **Connection info** – JDBC URLs, credentials
 - **Classification** for identifying and parsing files
 - **Versioning** of table metadata as schemas evolve and other metadata are updated
- Populate using Hive DDL, bulk import, or automatically through **Crawlers**.

AWS GLUE – DATA CATALOG - CRAWLERS

Crawlers automatically build your Data Catalog and keep it in sync

- Automatically discover new data, extracts schema definitions
 - Detect schema changes and version tables
 - Detect Hive style partitions on Amazon S3
- Built-in classifiers for popular types; custom classifiers
- Run ad hoc or on a schedule; serverless – only pay when crawler runs

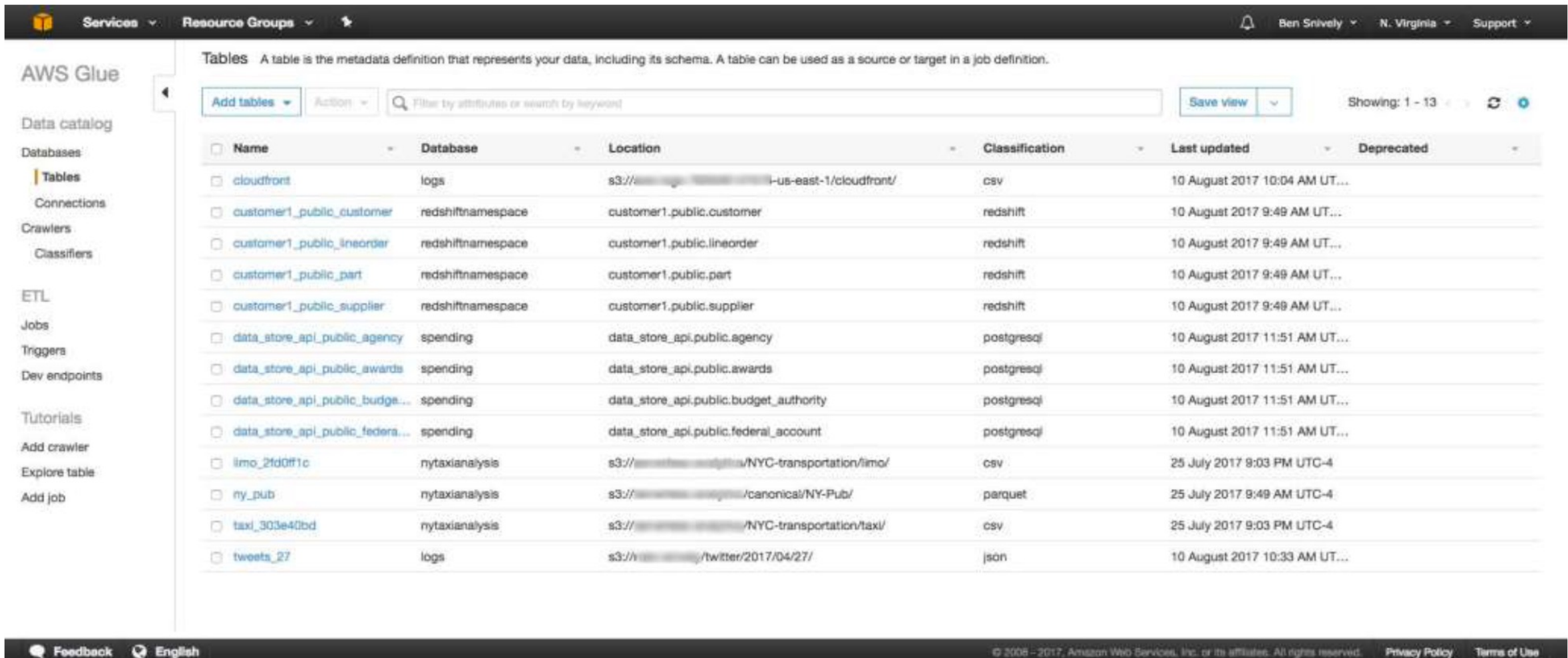
AWS GLUE – DATA CATALOG/CRAWLER



- Glue crawler scans data in S3, creates schema
- Can run periodically
- Populates the Glue Data Catalog
 - Stores only table definition
 - Original data stays in S3
- Once cataloged, you can treat your unstructured data like it's structured
 - Redshift Spectrum
 - Athena
 - EMR
 - Quicksight

AWS GLUE – DATA CATALOG

- Bring in metadata from a variety of data sources (Amazon S3, Amazon Redshift, etc.) into a single categorized list that is searchable



The screenshot displays the AWS Glue Data Catalog console. The left sidebar contains navigation links for AWS Glue, Data catalog, Databases, Tables (selected), Connections, Crawlers, Classifiers, ETL, Jobs, Triggers, Dev endpoints, Tutorials, Add crawler, Explore table, and Add job. The main content area shows a list of tables with columns: Name, Database, Location, Classification, Last updated, and Deprecated. A search bar and 'Add tables' button are at the top. The table list includes entries like 'cloudfront', 'customer1_public_customer', 'customer1_public_lineorder', 'customer1_public_part', 'customer1_public_supplier', 'data_store_api_public_agency', 'data_store_api_public_awards', 'data_store_api_public_budge...', 'data_store_api_public_federa...', 'limo_2fd0ff1c', 'ny_pub', 'taxi_303e40bd', and 'tweets_27'.

AWS Glue

Data catalog

Databases

Tables

Connections

Crawlers

Classifiers

ETL

Jobs

Triggers

Dev endpoints

Tutorials

Add crawler

Explore table

Add job

Tables A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

[Add tables](#) [Action](#) [Save view](#) Showing: 1 - 13

<input type="checkbox"/> Name	Database	Location	Classification	Last updated	Deprecated
<input type="checkbox"/> cloudfront	logs	s3://[redacted]-us-east-1/cloudfront/	csv	10 August 2017 10:04 AM UT...	
<input type="checkbox"/> customer1_public_customer	redshiftnamespace	customer1.public.customer	redshift	10 August 2017 9:49 AM UT...	
<input type="checkbox"/> customer1_public_lineorder	redshiftnamespace	customer1.public.lineorder	redshift	10 August 2017 9:49 AM UT...	
<input type="checkbox"/> customer1_public_part	redshiftnamespace	customer1.public.part	redshift	10 August 2017 9:49 AM UT...	
<input type="checkbox"/> customer1_public_supplier	redshiftnamespace	customer1.public.supplier	redshift	10 August 2017 9:49 AM UT...	
<input type="checkbox"/> data_store_api_public_agency	spending	data_store_api.public.agency	postgresql	10 August 2017 11:51 AM UT...	
<input type="checkbox"/> data_store_api_public_awards	spending	data_store_api.public.awards	postgresql	10 August 2017 11:51 AM UT...	
<input type="checkbox"/> data_store_api_public_budge...	spending	data_store_api.public.budget_authority	postgresql	10 August 2017 11:51 AM UT...	
<input type="checkbox"/> data_store_api_public_federa...	spending	data_store_api.public.federal_account	postgresql	10 August 2017 11:51 AM UT...	
<input type="checkbox"/> limo_2fd0ff1c	nytaxianalysis	s3://[redacted]/NYC-transportation/limo/	csv	25 July 2017 8:03 PM UTC-4	
<input type="checkbox"/> ny_pub	nytaxianalysis	s3://[redacted]/canonical/NY-Pub/	parquet	25 July 2017 9:49 AM UTC-4	
<input type="checkbox"/> taxi_303e40bd	nytaxianalysis	s3://[redacted]/NYC-transportation/taxi/	csv	25 July 2017 9:03 PM UTC-4	
<input type="checkbox"/> tweets_27	logs	s3://[redacted]/twitter/2017/04/27/	json	10 August 2017 10:33 AM UT...	

[Feedback](#) [English](#) © 2006 – 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

AWS GLUE – JOB AUTHORIZING/GLUE ETL

➤ **Job Authoring Choices**

- Python code generated by AWS Glue
- Connect a notebook or IDE to AWS Glue
- Existing code brought into AWS Glue
- Automatic code generation in Scala or Python
- Server-side (at rest) and SSL (in transit)
- Can be event-driven
- Can provision additional “DPU’s” (data processing units) to increase performance of underlying Spark jobs
- Enabling job metrics can help you understand the maximum capacity in DPU’s you need
- Errors reported to CloudWatch ,Could also tie into SNS for notification

AWS GLUE – JOB AUTHORIZING/GLUE ETL

- Transform data, Clean Data, Enrich Data (before doing analysis)
 - Generate ETL code in Python or Scala, you can modify the code
 - Can provide your own Spark or PySpark scripts
 - Target can be S3, JDBC (RDS, Redshift), or in Glue Data Catalog
- Fully managed, cost effective, pay only for the resources consumed
- Jobs are run on a serverless Spark platform
- Glue Scheduler to schedule the jobs
- Glue Triggers to automate job runs based on “events”

AWS GLUE – JOB AUTHORIZING/GLUE ETL - TRANSFORMATIONS

➤ Bundled Transformations:

- DropFields, DropNullFields – remove (null) fields
- Filter – specify a function to filter records
- Join – to enrich data
- Map - add fields, delete fields, perform external lookups

➤ Machine Learning Transformations:

- **FindMatches ML:** identify duplicate or matching records in your dataset, even when the records do not have a common unique identifier and no fields match exactly.

➤ Format conversions: CSV, JSON, Avro, Parquet, ORC, XML

➤ Apache Spark transformations (example: K-Means)

- Can convert between Spark DataFrame and Glue DynamicFrame

AWS GLUE – RUNNING GLUE JOBS

- Time-based schedules (cron style)
 - Job bookmarks
 - Persists state from the job run
 - Prevents reprocessing of old data
 - Allows you to process new data only when re-running on a schedule
 - Works with S3 sources in a variety of formats
 - Works with relational databases via JDBC (if PK's are in sequential order)
 - Only handles new rows, not updated rows
- CloudWatch Events
 - Fire off a Lambda function or SNS notification when ETL succeeds or fails
 - Invoke EC2 run, send event to Kinesis, activate a Step Function

AWS GLUE – RUNNING GLUE JOBS

There is no need to provision, configure, or manage servers

- Auto-configure VPC and role-based access
- Customers can specify the capacity that gets allocated to each job
- Automatically scale resources (on post-GA roadmap)
- You pay only for the resources you consume while consuming them

AWS GLUE – COST MODEL

- Billed by the minute for crawler and ETL jobs
- First million objects stored and accesses are free for the Glue Data Catalog
- Development endpoints for developing ETL code charged by the minute

AWS GLUE – ANTI-PATTERNS

- Multiple ETL engines
 - Glue ETL is based on Spark
 - If you want to use other engines (Hive, Pig, etc) Data Pipeline EMR would be a better fit.
- No longer an anti-pattern: streaming
- As of April 2020, Glue ETL supports serverless streaming ETL
 - Consumes from Kinesis or Kafka
 - Clean & transform in-flight
 - Store results into S3 or other data stores
- Runs on Apache Spark Structured Streaming

AWS GLUE – GLUE STUDIO

- Visual interface for ETL workflows
 - Visual job editor
 - Create DAG's for complex workflows
 - Sources include S3, Kinesis, Kafka, JDBC
 - Transform / sample / join data
 - Target to S3 or Glue Data Catalog
 - Support partitioning
- Visual job dashboard
 - Overviews, status, run times

The screenshot displays the AWS Glue Studio web interface. The top navigation bar includes the AWS logo, a search bar, and user information. The main workspace shows a DAG (Directed Acyclic Graph) with the following components:

- Data source - S3 bucket**: The starting point of the workflow.
- Transform - ApplyMapping**: A central transformation node.
- Transform - Filter**: A filter node that receives input from the 'ApplyMapping' node.
- Data target - S3 bucket**: A target node that receives input from the 'Filter' node.
- Data target - S3 bucket**: Another target node that receives input from the 'ApplyMapping' node.

On the right side, the **Apply mapping** configuration panel is visible, showing a table of source and target keys with their respective data types and drop checkboxes.

Source key	Target key	Data type	Drop
invciceno	invciceno	string	<input type="checkbox"/>
stockcode	stockcode	string	<input type="checkbox"/>
description			<input checked="" type="checkbox"/>
quantity	quantity	long	<input type="checkbox"/>
invoicedate	invoicedate	string	<input type="checkbox"/>
unitprice	unitprice	double	<input type="checkbox"/>
customerid	customerid	long	<input type="checkbox"/>
country	country	string	<input type="checkbox"/>
year	year	int	<input type="checkbox"/>
month	month	string	<input type="checkbox"/>
day	day	int	<input type="checkbox"/>
hour	hour	string	<input type="checkbox"/>

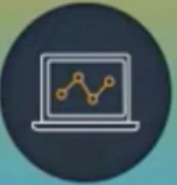
GLUE STUDIO BENEFITS

Visual ETL in AWS Glue



Rapid development

Author ETL jobs faster and with less debugging



Job Monitoring

See all your job runs through a single pane of glass



Streaming and batch ETL

Use one service to process data from all your sources



Complex Transformations

Use visual transforms or write code snippets to clean and prepare your data for analysis



Structured & semi structured data handling

Use Glue studio to handle structured and semi structured data like IOT logs