DATA ANALYTICS MINISERIES: INGESTION TO INSIGHTS

Data Analytics Miniseries

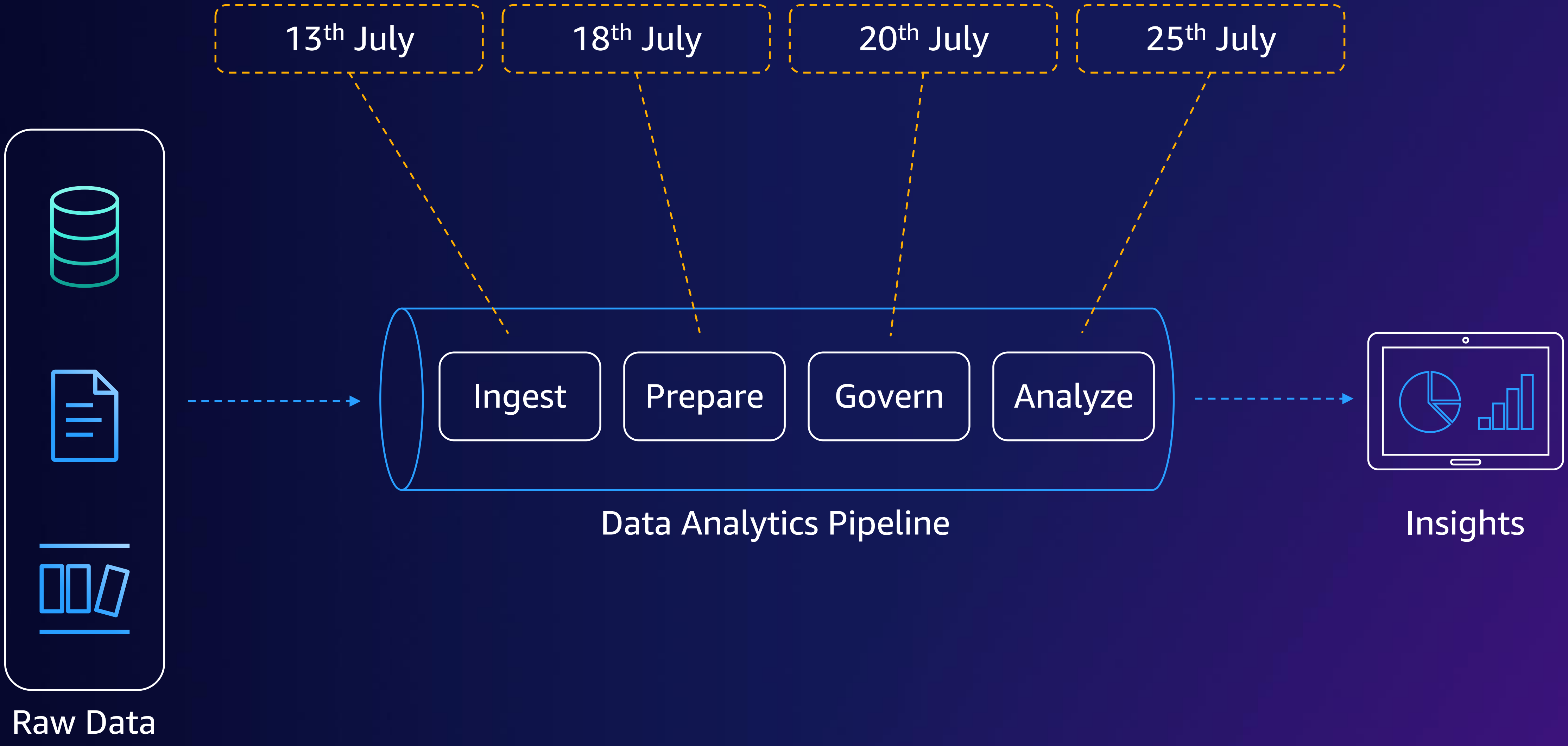
Episode 3: Data Preparation

Rhiannon Liebowitz
liebowi@amazon.com

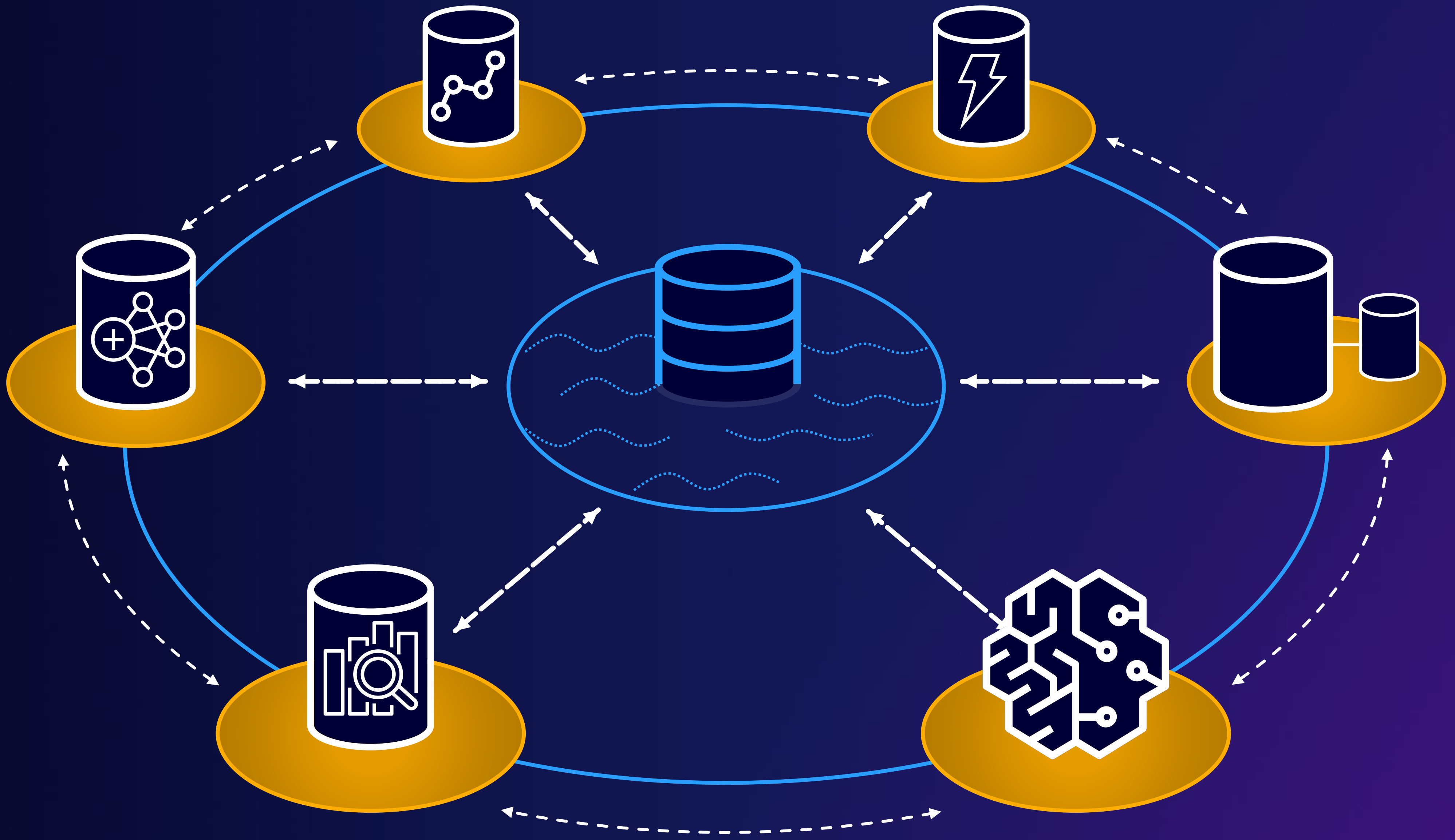
Pavan Kumar Vadupu
Lakshman Manikya
pavanvlm@amazon.com

Joel Farvault
jfarvau@amazon.com

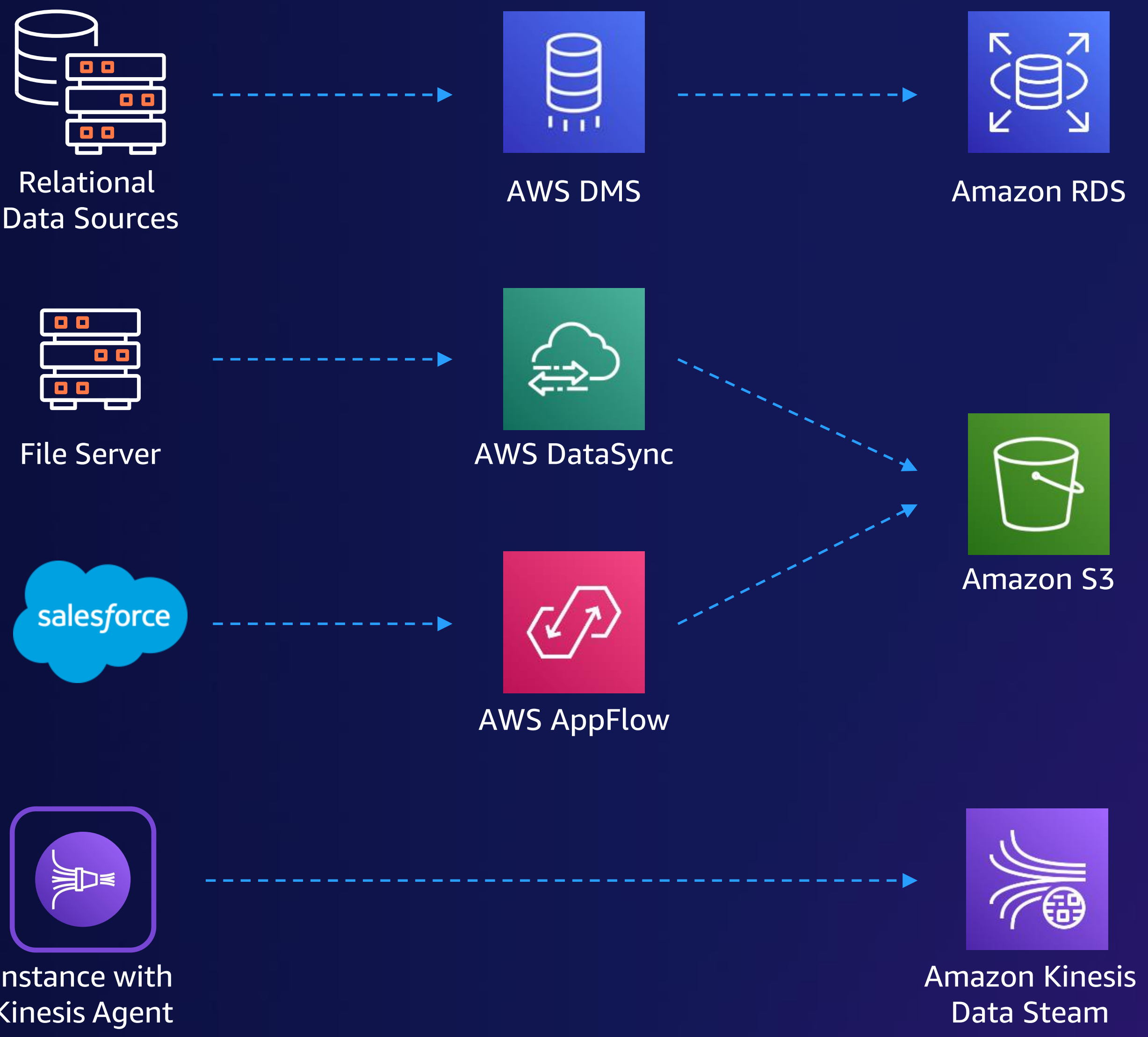
Series Overview



Our Data is in AWS, Now What?

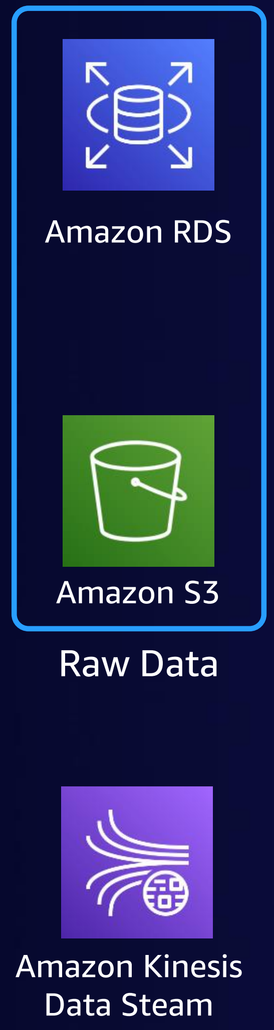


Our Data Journey



Our Data Journey

RAW DATA



- A mix of structured (Relational) & unstructured (JSON files) data
- Comprise multiple unknown formats
- Requires enrichment by an Extract Transform Load (ETL) processing
- Requires format changes or compression
- Is not accessible by AWS analytical services
- Unknown / questionable data quality

We have our raw data in AWS, but how do we get it ready for processing?

AWS Glue

COMPONENTS

- Data Catalog
- Schema Registry
- Crawlers
- Jobs
- Studio
- DataBrew/Quality

AWS Glue

BENEFITS

Fully Managed Serverless

No Infrastructure to maintain.

Scaling

Elastically scales to meet your workload requirements.

Cost Effective

Only pay for the resources you use.

No Lock In

ETL in open source frameworks: SparkSQL, PySpark or Scala

How does Glue help us make sense of the raw data we have in AWS?

How do we expose our data to AWS analytical services?

AWS Glue Data Catalog

WHAT IT IS

“The AWS Glue Data Catalog is a central repository to store structural and operational metadata for all your data assets. For a given data set, you can store its table definition, physical location, add business relevant attributes, as well as track how this data has changed over time.”

Central repository of metadata for your data sources

Stores schemas for structured and semi-structured data sources

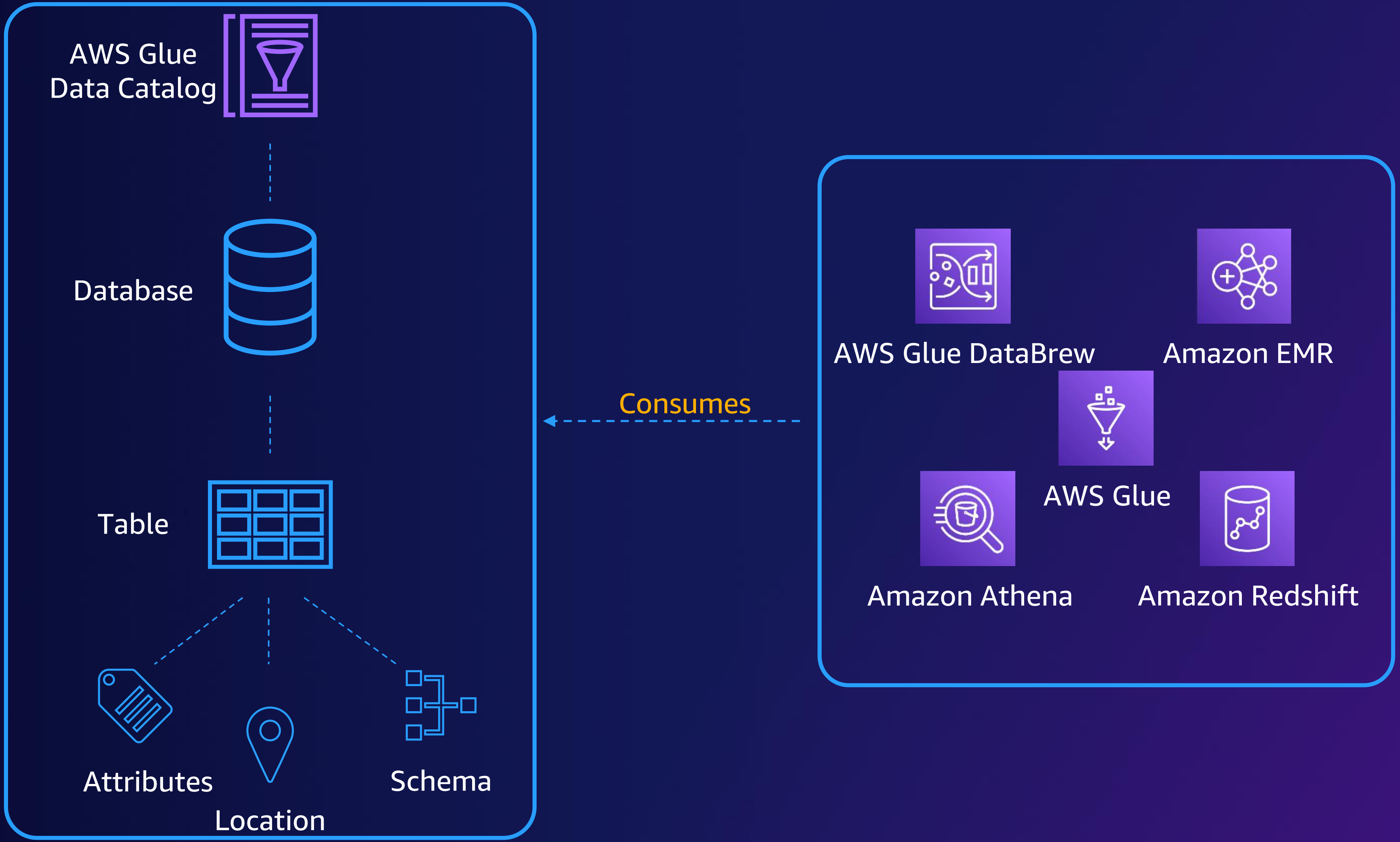
Stores physical location

Stores business metadata / attributes

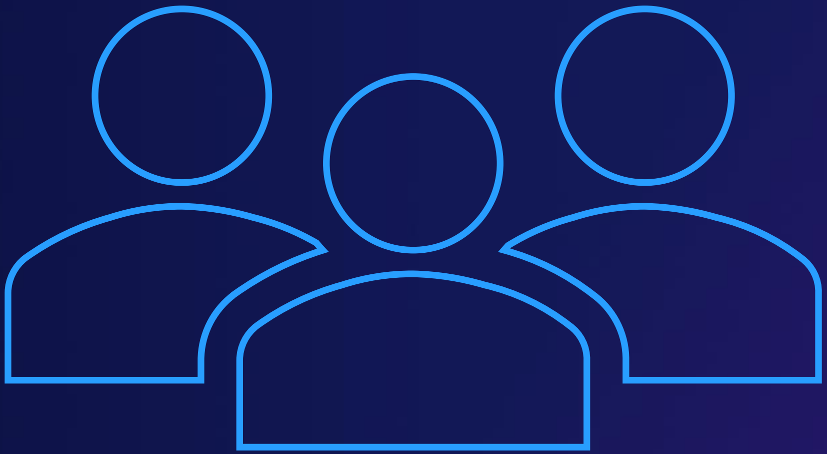
Tracks schema changes over time

AWS Glue Data Catalog

HOW IT WORKS



Manually capturing our data catalog won’t scale! How can we discover our data?



How do we automatically track changes to the structure of our data over time?

AWS Glue Crawlers

WHAT IT IS

“An AWS Glue crawler connects to a data store, progresses through a prioritized list of classifiers to extract the schema of your data and other statistics, and then populates the Glue Data Catalog with this metadata. Crawlers can run periodically to detect the availability of new data as well as changes to existing data, including table definition changes.”

Connect to varied data sources

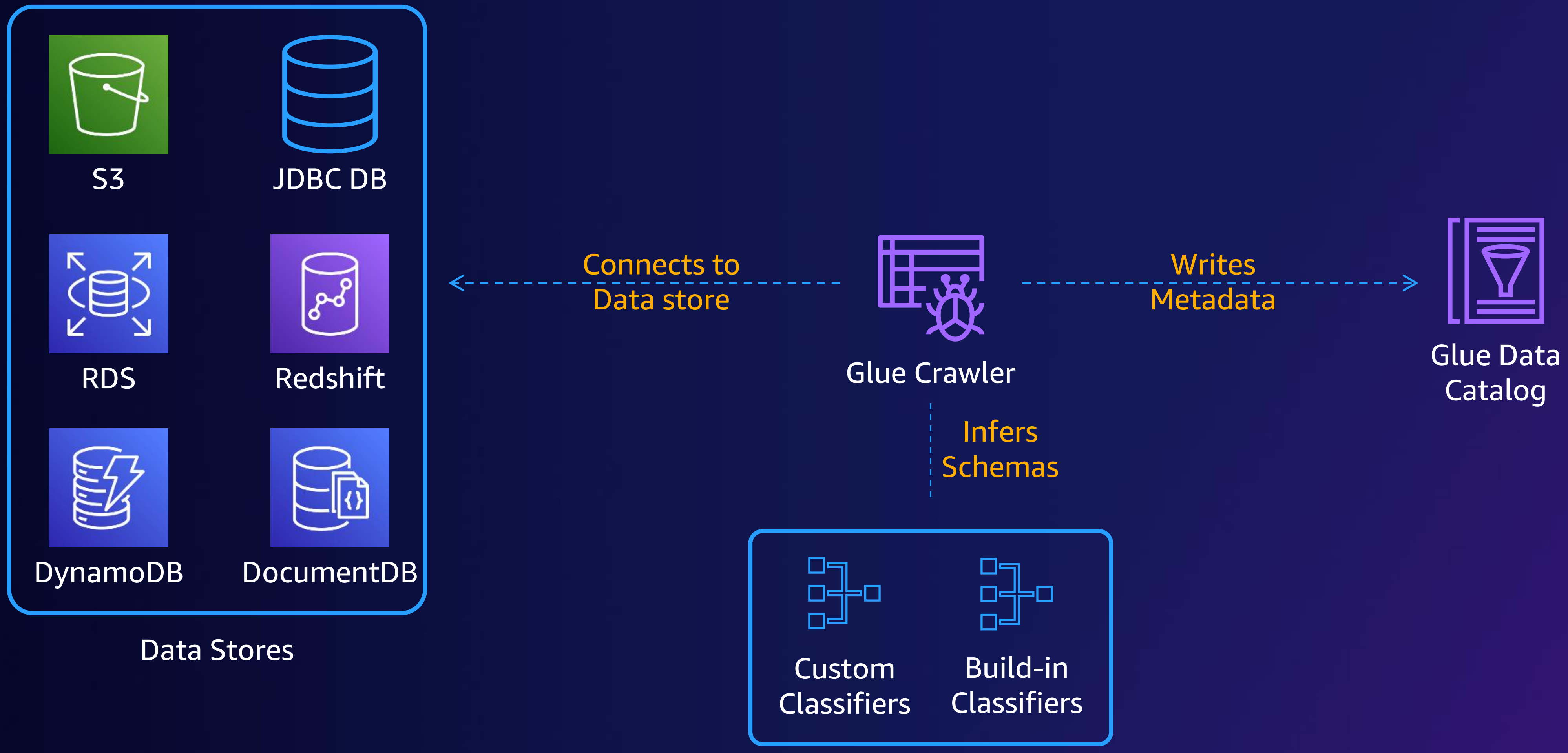
Extract schemas at scale by applying classifiers to your data.

Track schema changes over time

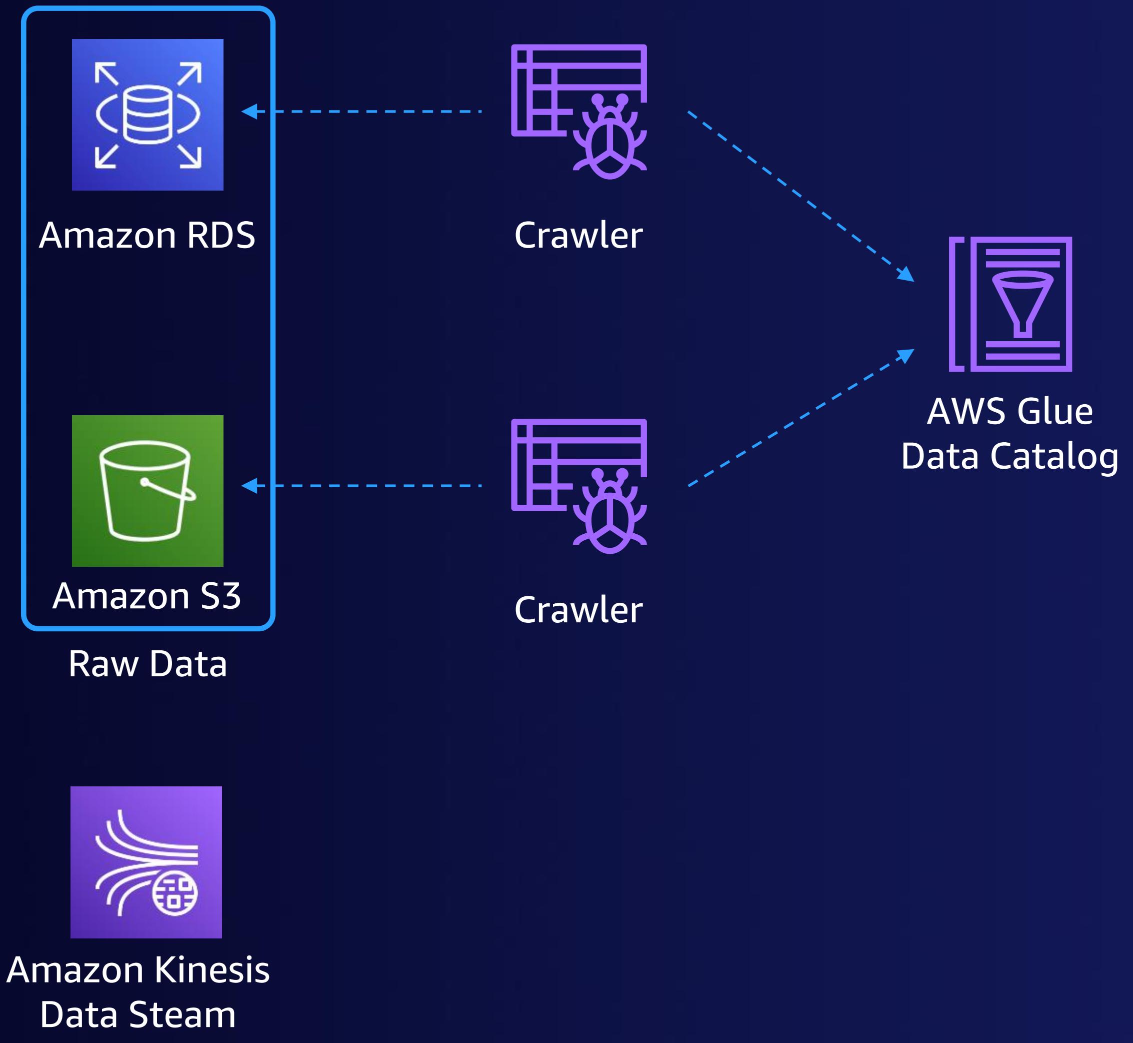
Crawlers can be either be triggered or scheduled

AWS Glue Crawlers

HOW IT WORKS



Our Data Journey



How do we transform or enrich our data?

AWS Glue Jobs

WHAT IT IS

“An AWS Glue enables customers to execute fully managed and scalable Extract Transform & Load (ETL) jobs on their data. The ETL engine that can automatically generate Scala or Python code and incorporates a flexible scheduler that handles dependency resolution, job monitoring, and retries.”

No Infrastructure to maintain

Elastically scales to meet your workload requirements

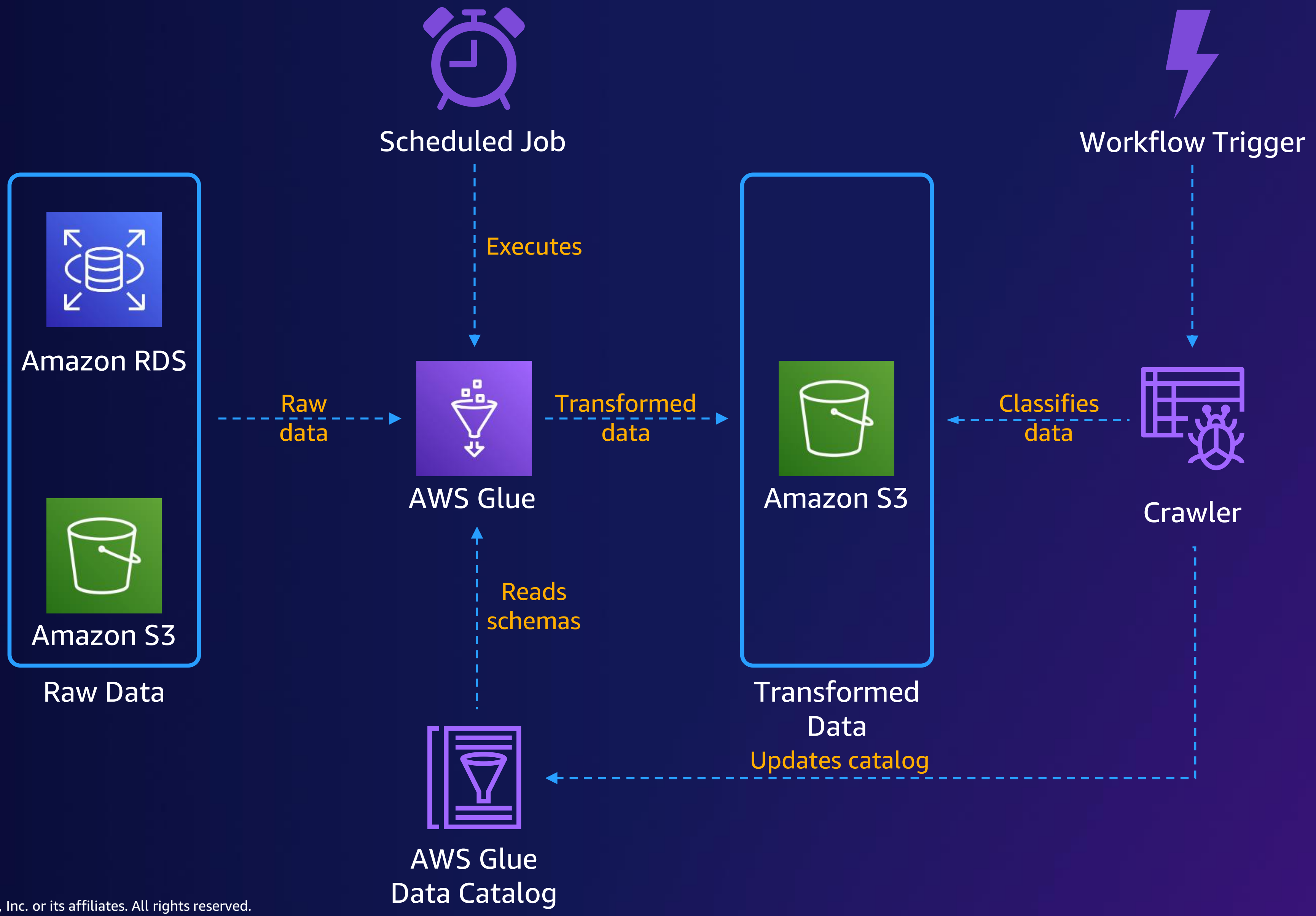
Handles scheduling, monitoring, dependencies and retries

ETL in open source frameworks: SparkSQL, PySpark or Scala

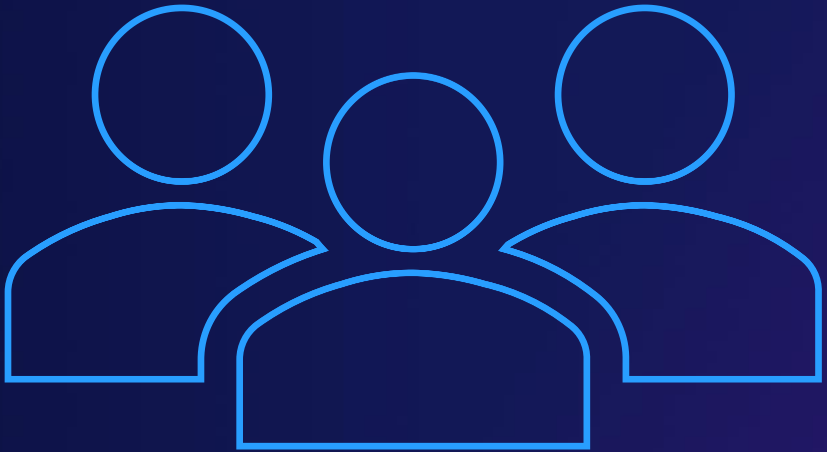
Save up to 35% with the Flexible Execution Class for non urgent workloads

AWS Glue Jobs

HOW IT WORKS



What if we don’t have developers and want to make use of an IDE?



AWS Glue Studio

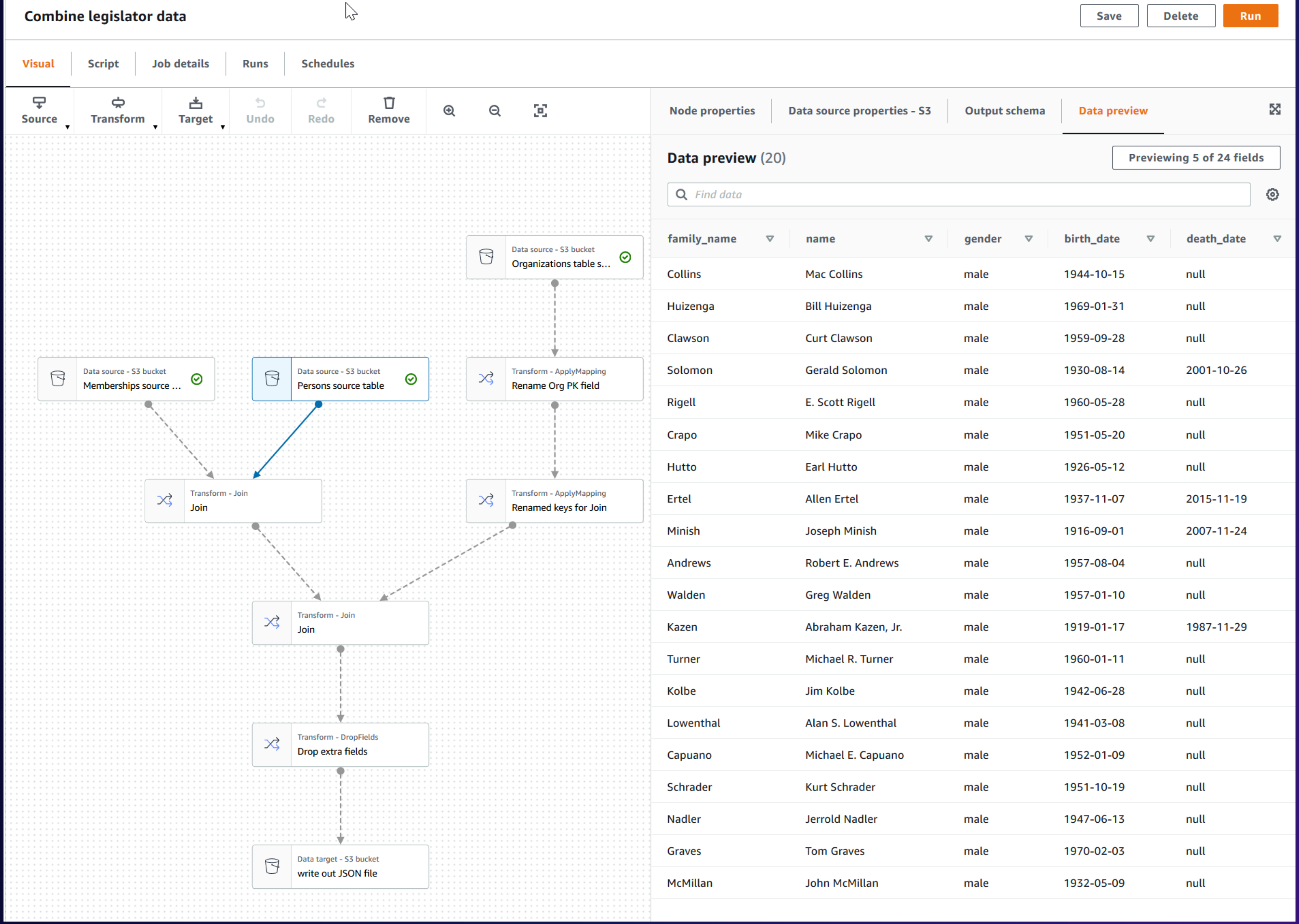
WHAT IT IS

“AWS Glue Studio is a new graphical interface that makes it easy to create, run, and monitor extract, transform, and load (ETL) jobs in AWS Glue. You can visually compose data transformation workflows and seamlessly run them on AWS Glue’s Apache Spark-based serverless ETL engine. You can inspect the schema and data results in each step of the job.”

Visually compose ETL Glue Jobs through an IDE

Launch & monitor jobs through the IDE

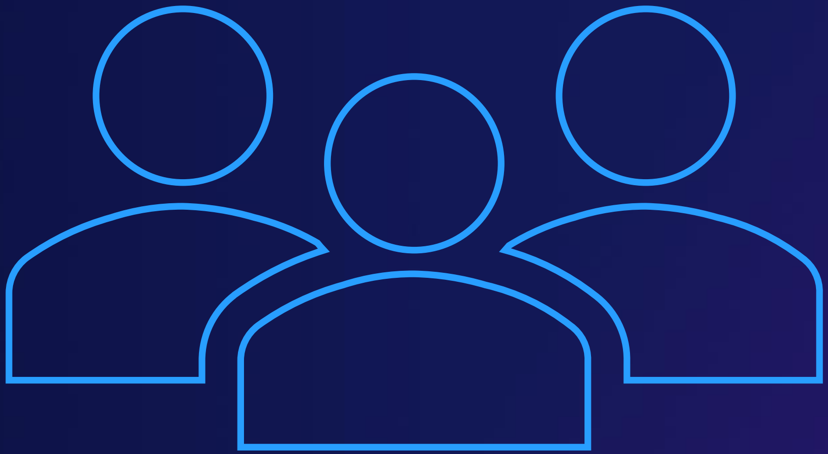
Export generated code



Data preview (20)

family_name	name	gender	birth_date	death_date
Collins	Mac Collins	male	1944-10-15	null
Huizenga	Bill Huizenga	male	1969-01-31	null
Clawson	Curt Clawson	male	1959-09-28	null
Solomon	Gerald Solomon	male	1930-08-14	2001-10-26
Rigell	E. Scott Rigell	male	1960-05-28	null
Crapo	Mike Crapo	male	1951-05-20	null
Hutto	Earl Hutto	male	1926-05-12	null
Ertel	Allen Ertel	male	1937-11-07	2015-11-19
Minish	Joseph Minish	male	1916-09-01	2007-11-24
Andrews	Robert E. Andrews	male	1957-08-04	null
Walden	Greg Walden	male	1957-01-10	null
Kazen	Abraham Kazen, Jr.	male	1919-01-17	1987-11-29
Turner	Michael R. Turner	male	1960-01-11	null
Kolbe	Jim Kolbe	male	1942-06-28	null
Lowenthal	Alan S. Lowenthal	male	1941-03-08	null
Capuano	Michael E. Capuano	male	1952-01-09	null
Schrader	Kurt Schrader	male	1951-10-19	null
Nadler	Jerrold Nadler	male	1947-06-13	null
Graves	Tom Graves	male	1970-02-03	null
McMillan	John McMillan	male	1932-05-09	null

How do we understand and profile our data?



AWS Glue DataBrew

WHAT IT IS

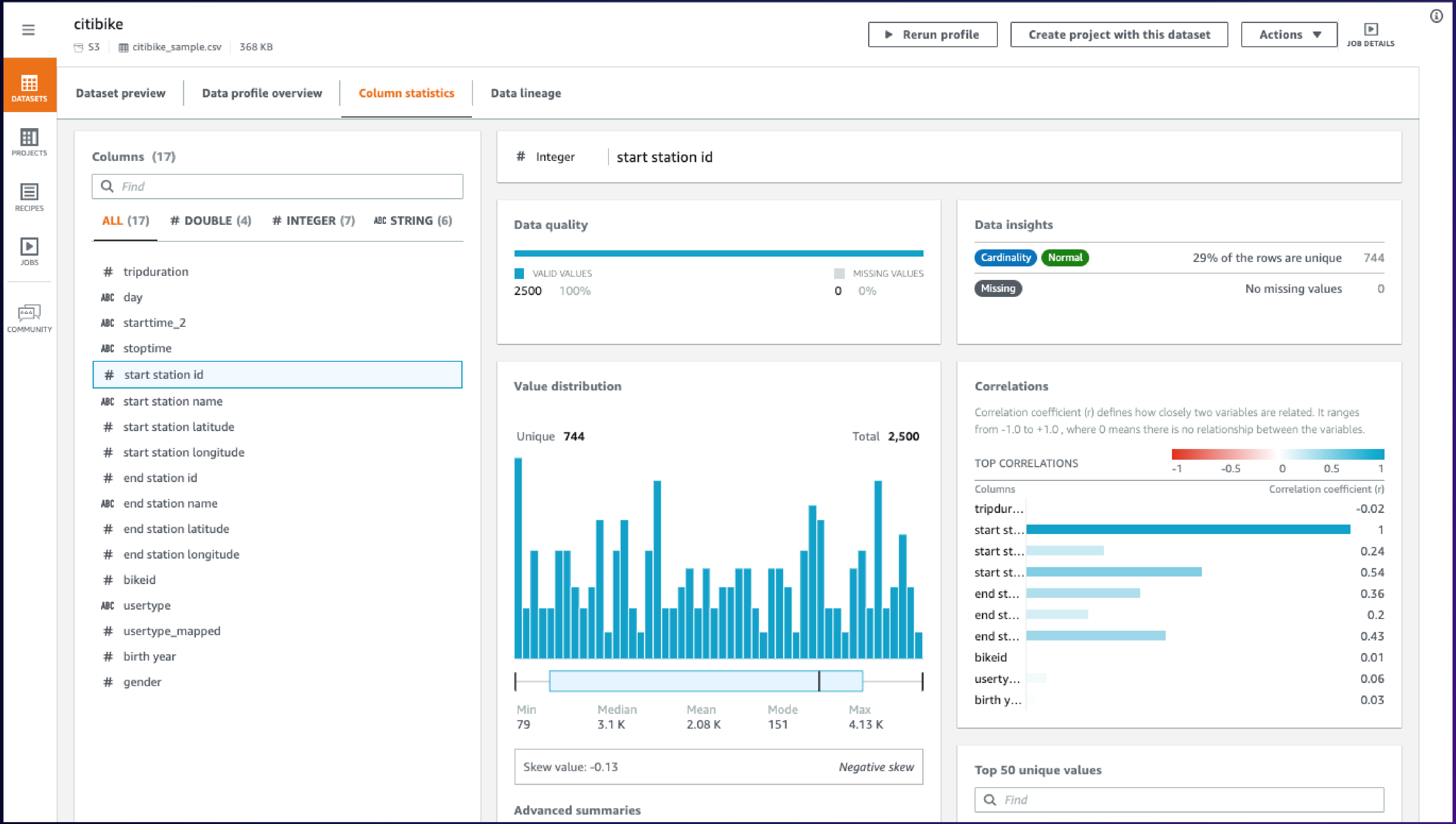
“AWS Glue DataBrew is a visual data preparation tool that makes it easy for data analysts and data scientists to prepare data with an interactive, point-and-click visual interface without writing code. With Glue DataBrew, you can easily visualize, clean, and normalize terabytes, and even petabytes of data directly from your data lake, data warehouses, and databases”

Easily visually prepare data through an IDE without code

Clean and normalize petabytes of data using over 250 built in transformations

Visually map data lineage

Save transformations as a recipe for reuse on new incoming data



citibike

S3 | citibike_sample.csv | 368 KB

Dataset preview | Data profile overview | **Column statistics** | Data lineage

Columns (17)

Find

ALL (17) | # DOUBLE (4) | # INTEGER (7) | ABC STRING (6)

- # tripduration
- ABC day
- ABC starttime_2
- ABC stoptime
- # start station id
- ABC start station name
- # start station latitude
- # start station longitude
- # end station id
- ABC end station name
- # end station latitude
- # end station longitude
- # bikeid
- ABC usertype
- # usertype_mapped
- # birth year
- # gender

Integer | start station id

Data quality

VALID VALUES		MISSING VALUES	
2500	100%	0	0%

Data insights

Cardinality Normal	29% of the rows are unique	744
Missing	No missing values	0

Value distribution

Unique 744 | Total 2,500

Min	Median	Mean	Mode	Max
79	3.1 K	2.08 K	151	4.13 K

Skew value: -0.13 *Negative skew*

Advanced summaries

Correlations

Correlation coefficient (r) defines how closely two variables are related. It ranges from -1.0 to +1.0 , where 0 means there is no relationship between the variables.

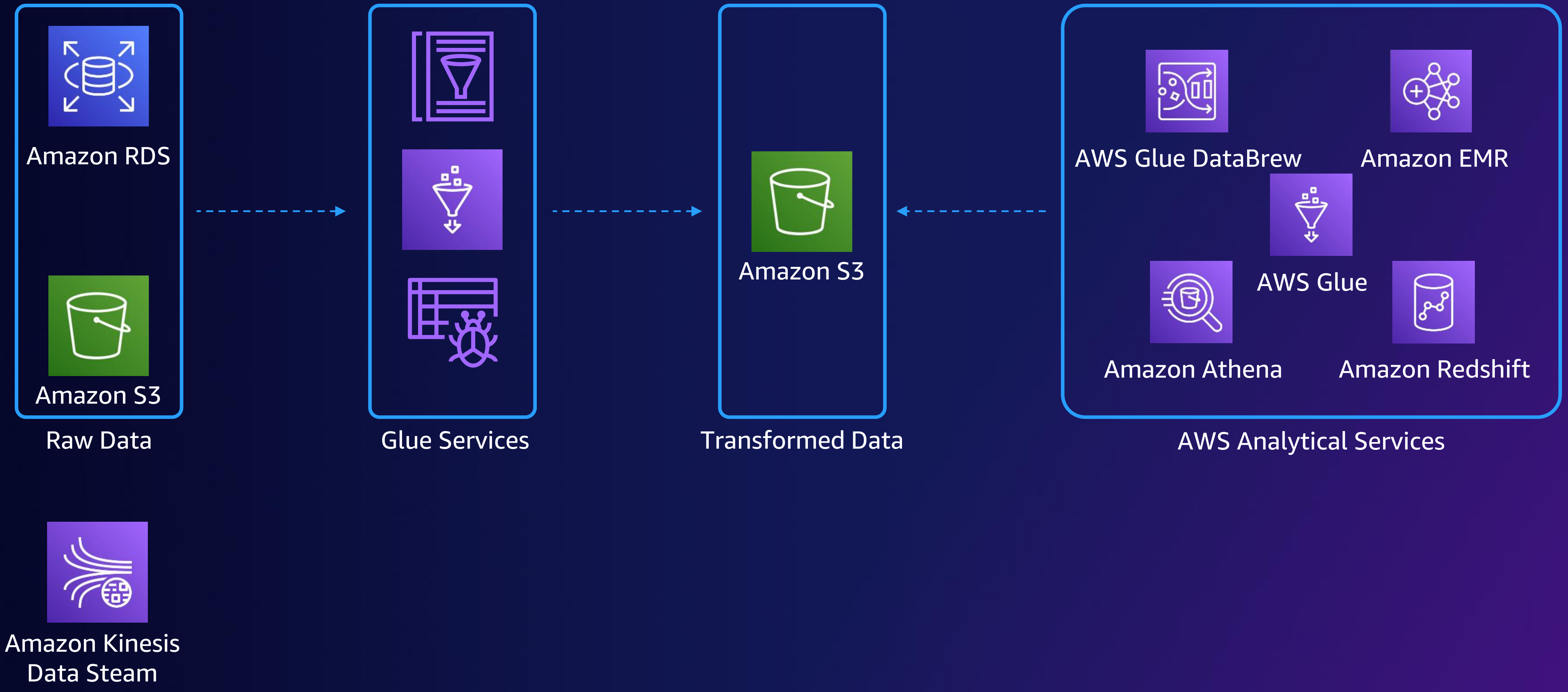
TOP CORRELATIONS

Columns	Correlation coefficient (r)
tripdur…	-0.02
start st…	1
start st…	0.24
start st…	0.54
end st…	0.36
end st…	0.2
end st…	0.43
bikeid	0.01
userty…	0.06
birth y…	0.03

Top 50 unique values

Find

Our Data Journey



Amazon RDS

Amazon S3

Raw Data

Glue Services

Amazon S3

Transformed Data

AWS Glue DataBrew

Amazon EMR

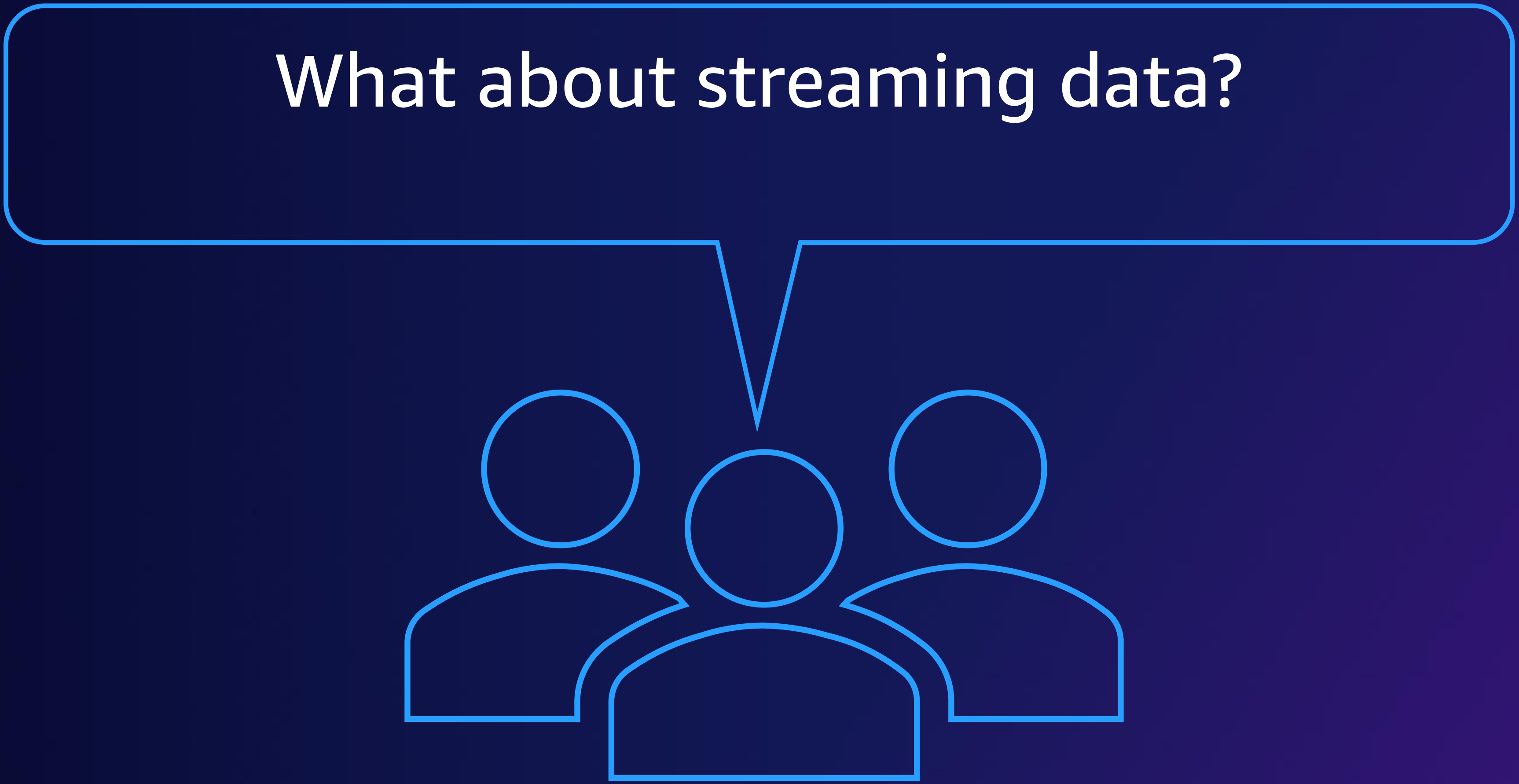
AWS Glue

Amazon Athena

Amazon Redshift

AWS Analytical Services

Amazon Kinesis
Data Steam



What about streaming data?

AWS Glue Schema Registry

WHAT IT IS

“AWS Glue Schema Registry, a serverless feature of AWS Glue, enables you to validate and control the evolution of streaming data using schemas registered in Apache Avro and JSON Schema data formats, at no additional charge. Through Apache-licensed serializers and deserializers, the Schema Registry integrates with data streaming applications developed for Apache Kafka, Amazon Managed Streaming for Apache Kafka (MSK), Amazon Kinesis Data Streams, Apache Flink, Amazon Kinesis Data Analytics for Apache Flink, and AWS Lambda.”

Supports multiple streaming platforms including: MSK, Amazon Kinesis, Apache Flink and Lambda

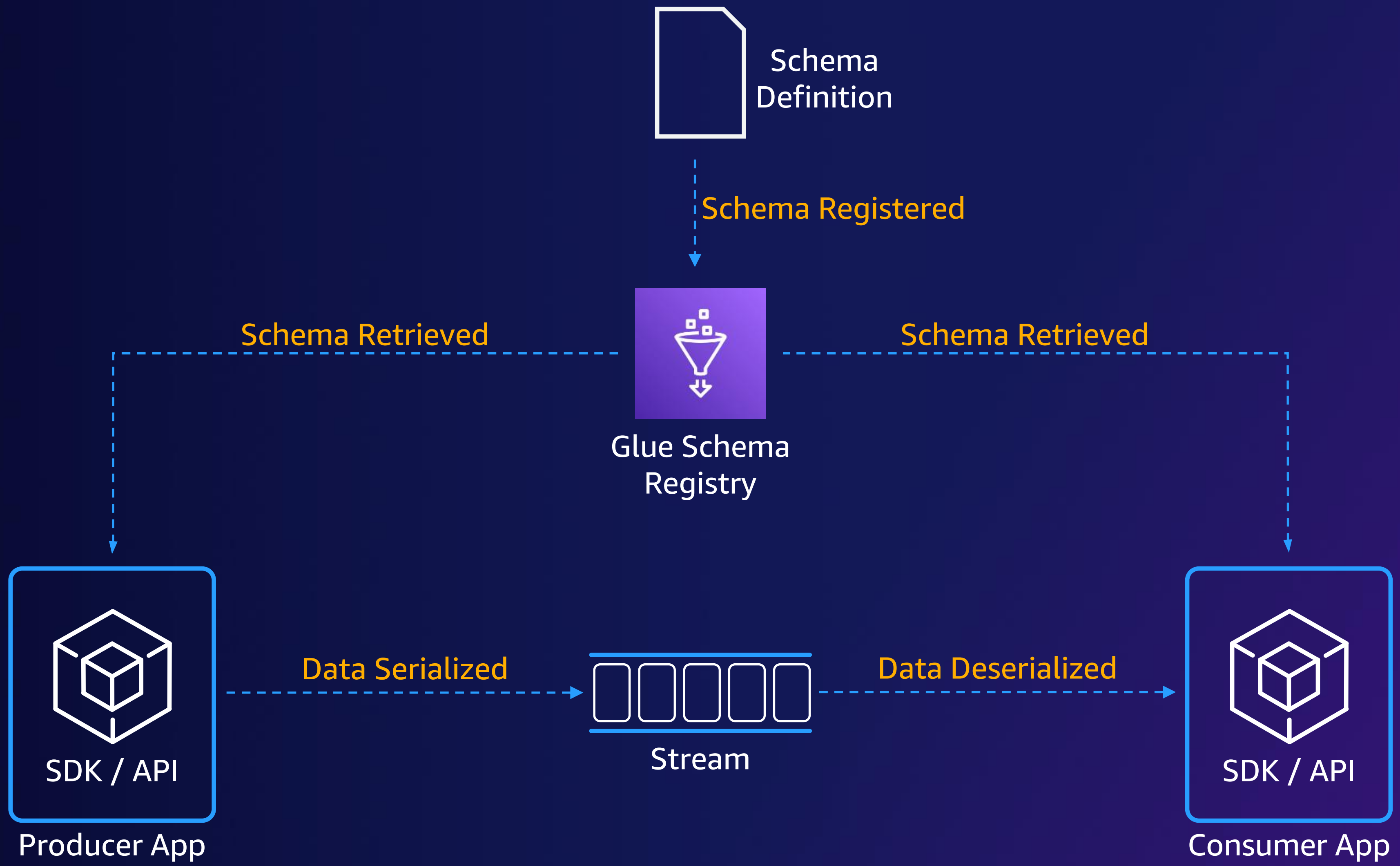
Improves data quality by performing schema validation

Store, validate and control the evolution of schemas

Serializers convert data into a binary format and can compress it before it is delivered, reducing data transfer and storage costs.

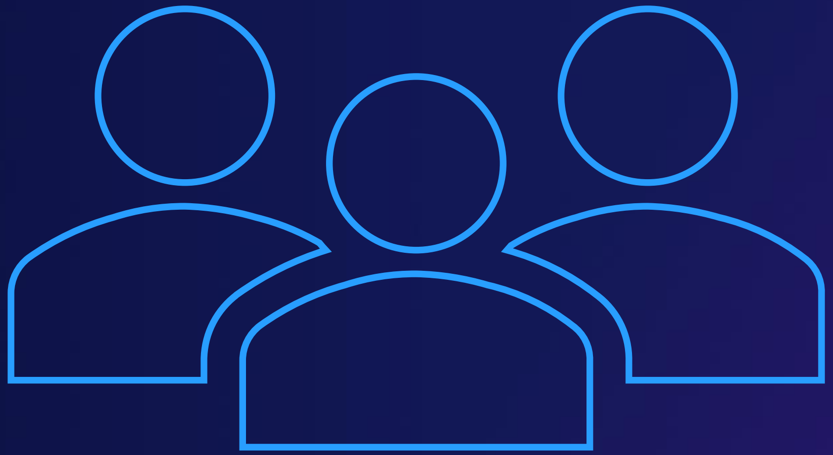
AWS Glue Schema Registry

HOW IT WORKS



AWS Glue

How do I perform ETL on Streaming Data?



Streaming ETL

AWS Lambda

Is a serverless, event-driven compute service that lets you run code for virtually any type of without provisioning or managing servers.

Kinesis Data Firehose

Is an ETL service that reliably captures, transforms, and delivers streaming data to data lakes, data stores, and analytics services.

Kinesis Family

USE CASES

AWS Lambda

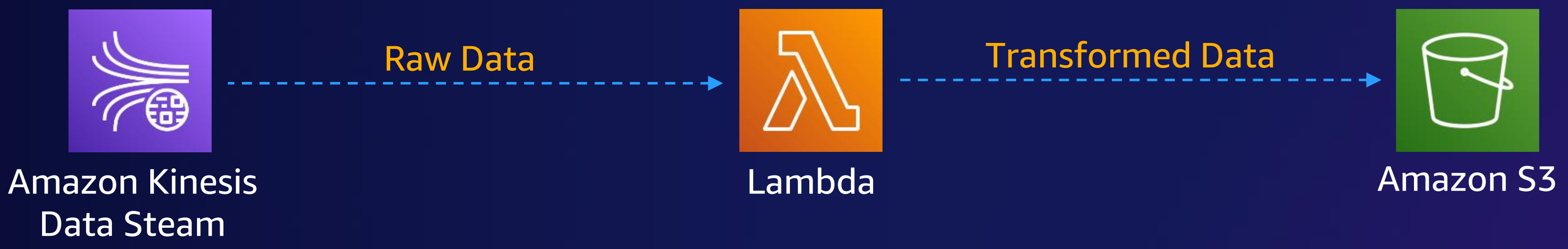
- Capture streaming data for real-time downstream processing.
- Allows the execution of custom code and business logic.
- Cannot run for more than 15 minutes per execution.
- Can consume from multiple data sources and events.

Kinesis Data Firehose

- Buffers records in a stream into a single output for more efficient storage.
- Time Based Buffer 1 to 15 mins.
- Volume Based Buffer 1 to 128 MB.
- Automatically flushing of buffer to S3, OpenSearch or other downstream destination.
- Outputs in near real-time.
- Supports custom Lambda transformations and data format conversion to Parquet / ORC.

AWS Lambda

HOW IT WORKS

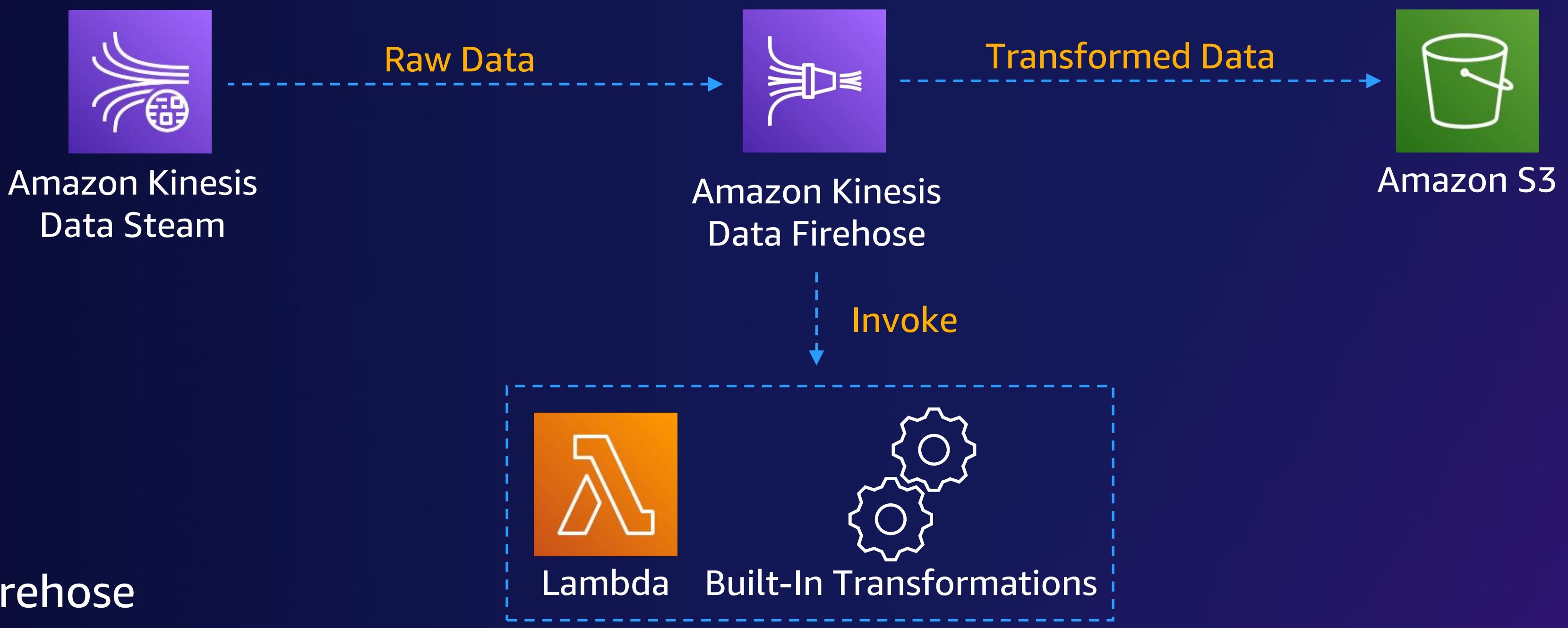


Lambda is a Swiss Army Knife of services, it could run any logic required:

- Interact with all other AWS services
- Enrich from other data sources
- Include custom libraries (for serialization)
- Call external services / applications
- In real time

Kinesis Data Firehose

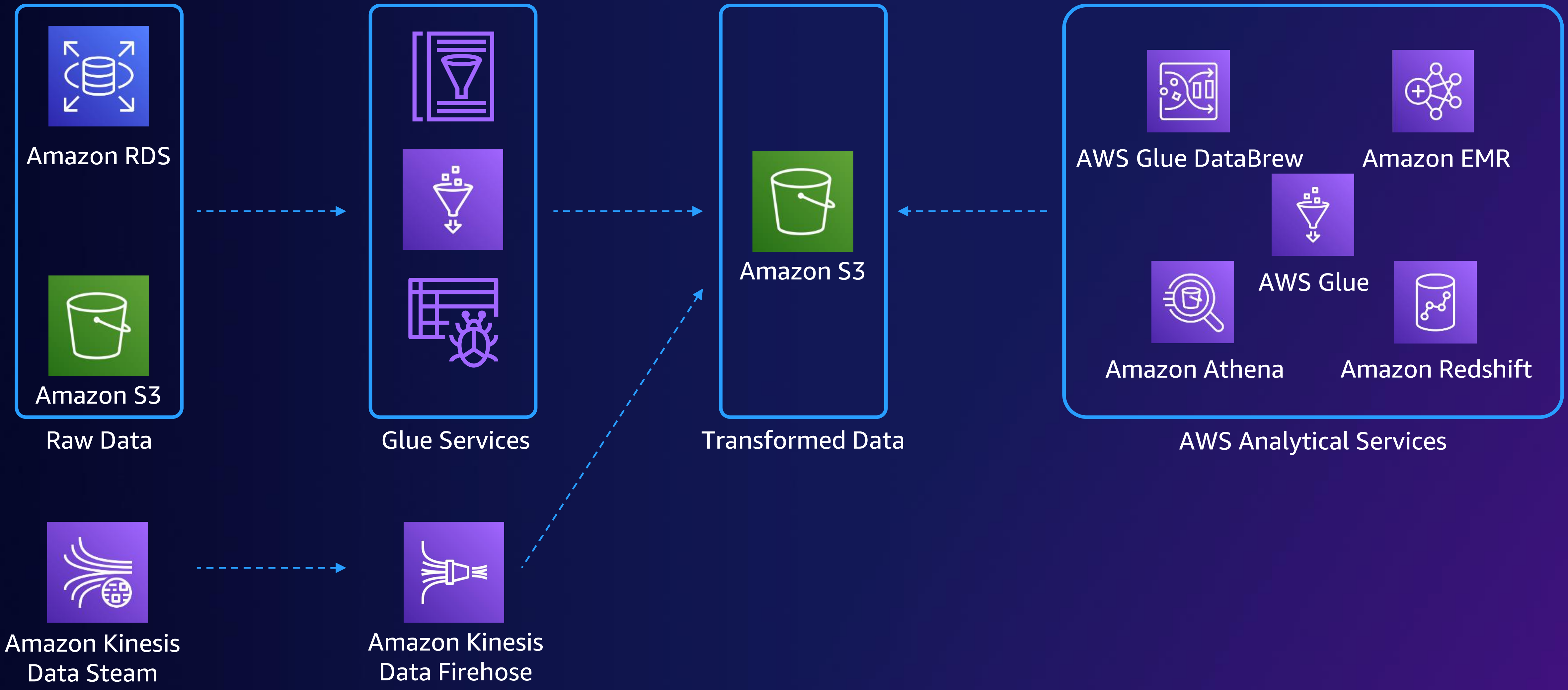
HOW IT WORKS



Kinesis Data Firehose

- Will read multiple messages
- Optionally invoke a Lambda to transform each message
- Built-in transformations can convert to ORC, Parquet and write to partitions
- Flush multiple transformed messages into a single output file
- In near real time

Our Data Journey



What if I want to perform quality checks on my data?

AWS Glue Data Quality

WHAT IT IS

“AWS Glue Data Quality automatically measures and monitors the quality of data in data lakes and pipelines. In data lakes, it then automatically recommends data quality rules. You can modify these rules, add additional rules from built-in rule types, and configure actions to alert teams when quality issues occur. Rules can also be included in AWS Glue data pipelines and scheduled to run periodically.

Rule Recommendations

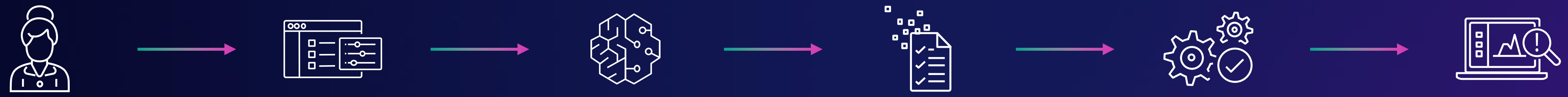
Data Quality Definition Language

Incorporated into data catalog and glue jobs

Serverless, scalable and high-performing

AWS Glue Data Quality

HOW IT WORKS WITH THE GLUE DATA CATALOG



Data steward

Selects a dataset in the AWS Glue Data Catalog

AWS Glue Data Quality analyzes data and recommends rules

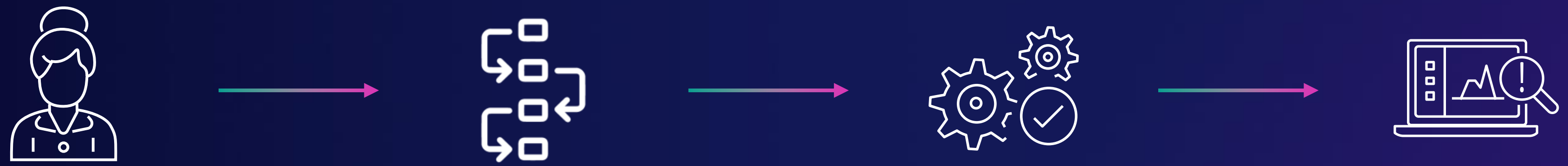
Data steward refines rules to create finalized rules

AWS Glue Data Quality evaluates rules

Data steward reviews results and alerts and takes appropriate action

AWS Glue Data Quality

HOW IT WORKS WITH GLUE ETL PIPELINES



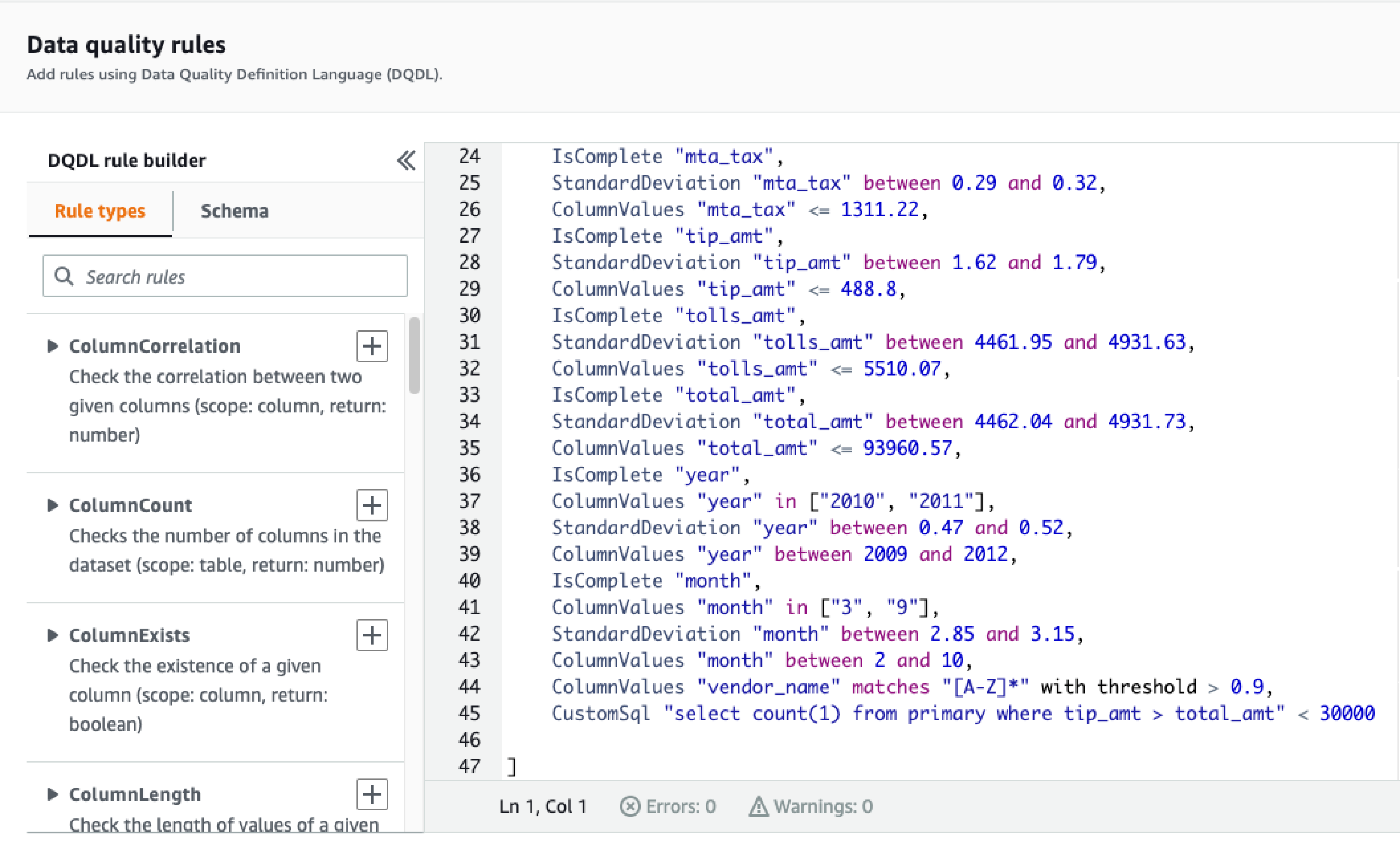
Data engineer

Selects an ETL job and adds AWS Glue Data Quality rules and actions

AWS Glue Data Quality evaluates rules

Engineer reviews results and alerts and takes appropriate action

AWS Glue Data Quality



Data quality rules

Add rules using Data Quality Definition Language (DQDL).

DQDL rule builder

ColumnCorrelation Check the correlation between two given columns (scope: column, return: number)

ColumnCount Checks the number of columns in the dataset (scope: table, return: number)

ColumnExists Check the existence of a given column (scope: column, return: boolean)

ColumnLength Check the length of values of a given

```
    IsComplete "mta_tax",
    StandardDeviation "mta_tax" between 0.29 and 0.32,
    ColumnValues "mta_tax" <= 1311.22,
    IsComplete "tip_amt",
    StandardDeviation "tip_amt" between 1.62 and 1.79,
    ColumnValues "tip_amt" <= 488.8,
    IsComplete "tolls_amt",
    StandardDeviation "tolls_amt" between 4461.95 and 4931.63,
    ColumnValues "tolls_amt" <= 5510.07,
    IsComplete "total_amt",
    StandardDeviation "total_amt" between 4462.04 and 4931.73,
    ColumnValues "total_amt" <= 93960.57,
    IsComplete "year",
    ColumnValues "year" in ["2010", "2011"],
    StandardDeviation "year" between 0.47 and 0.52,
    ColumnValues "year" between 2009 and 2012,
    IsComplete "month",
    ColumnValues "month" in ["3", "9"],
    StandardDeviation "month" between 2.85 and 3.15,
    ColumnValues "month" between 2 and 10,
    ColumnValues "vendor_name" matches "[A-Z]*" with threshold > 0.9,
    CustomSql "select count(1) from primary where tip_amt > total_amt" < 30000

]
```

Ln 1, Col 1 Errors: 0 Warnings: 0

- Easier to author
- Intuitive
- Complex rule support
- Reusable and easier to deploy
- Author SQL-based rules

Demo