
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

BOA309

Build a managed analytics platform for your ecommerce business

Rohini Gaonkar

Sr. Developer Advocate
Amazon Web Services

Suman Debnath

Principal Developer Advocate
Amazon Web Services

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Shopping, shopping, and more shopping!

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Shopping, shopping, and more shopping!

END USER STANDPOINT



Good deals



Quality products

Useful recommendations

Competitive price

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Shopping, shopping, and more shopping!

ORGANIZATION STANDPOINT

prime day



amazon

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Analytics platform for an ecommerce store

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Analytics platform for an ecommerce store

Brand's popularity

Know in which brands you need to invest more marketing



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Analytics platform for an ecommerce store

Brand’s popularity

Know in which brands you need to invest more marketing

User’s journey

Why do users add products to their carts after 11:00 PM but not buy them?



aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Analytics platform for an ecommerce store

Brand's popularity

Know in which brands you need to invest more marketing

User's journey

Why do users add products to their carts after 11:00 PM but not buy them?

Know your customers

Unique visitors per day
Top categories per hour or weekday



aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

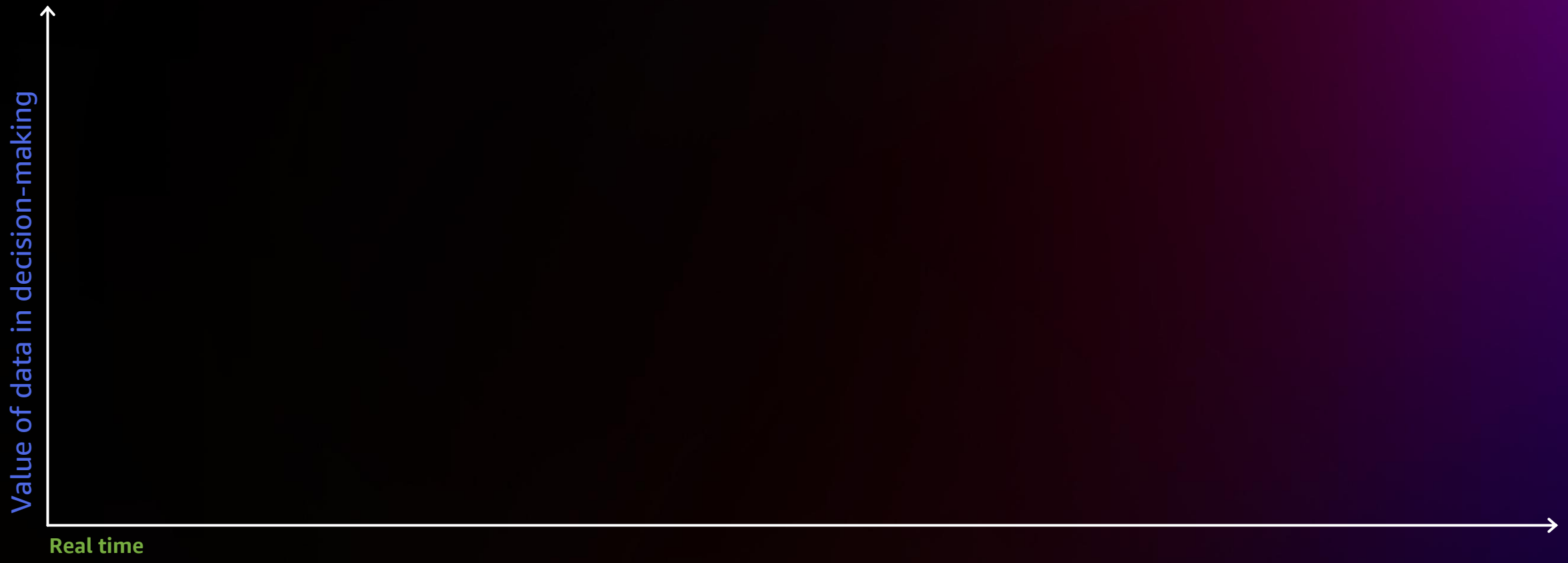
It’s all about time!

DATA LOSES VALUE QUICKLY OVER TIME

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

It’s all about time!

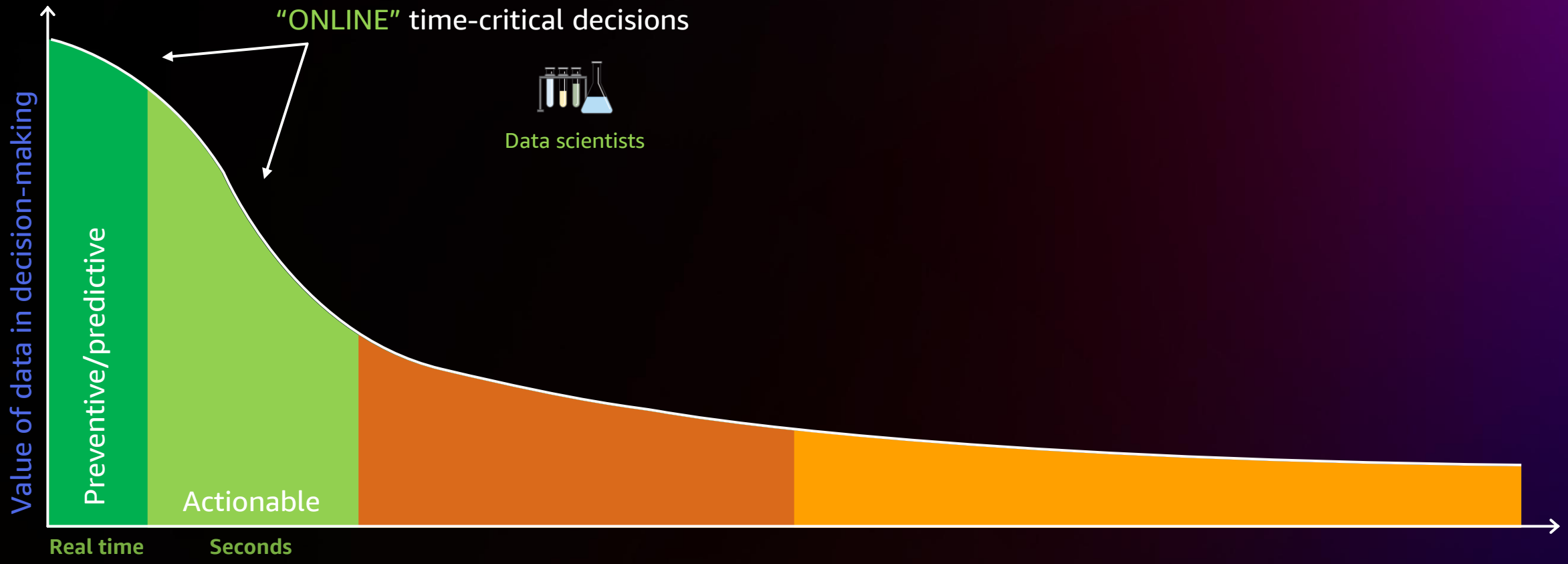
DATA LOSES VALUE QUICKLY OVER TIME



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

It’s all about time!

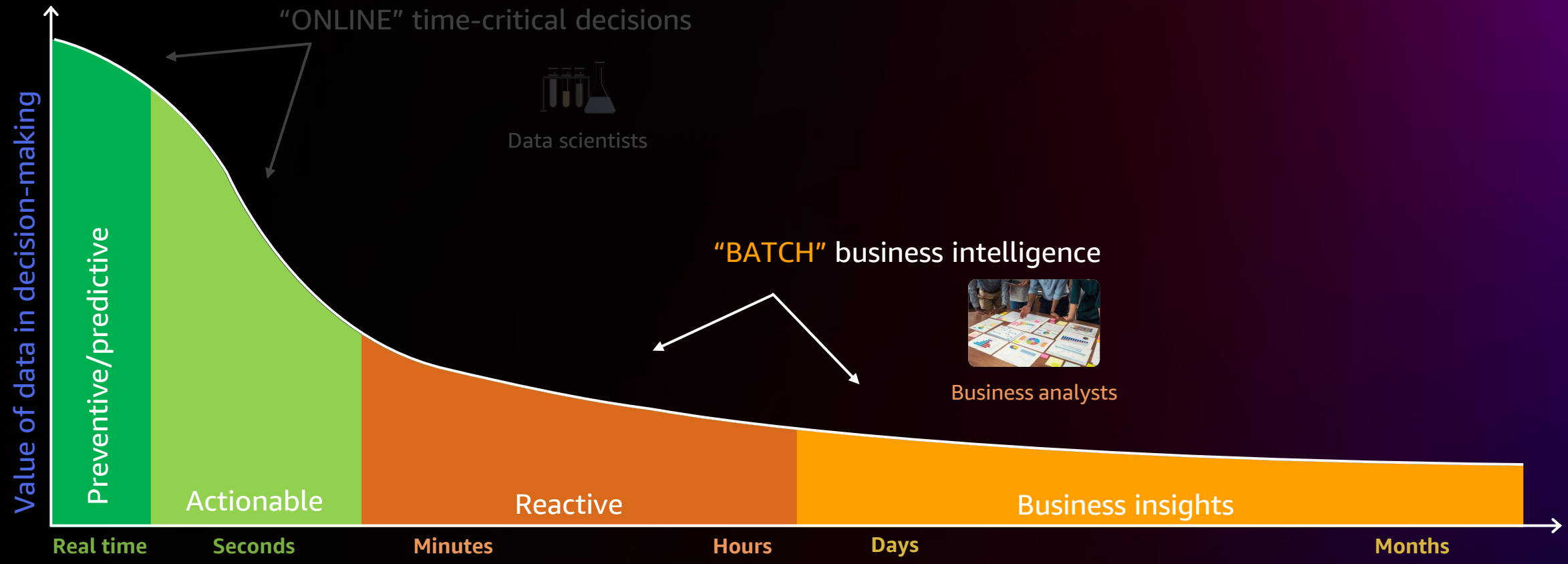
DATA LOSES VALUE QUICKLY OVER TIME



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

It’s all about time!

DATA LOSES VALUE QUICKLY OVER TIME



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What are we building in the next 40 minutes?

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What are we building in the next 40 minutes?

- End-to-end data engineering pipeline
  - Simulate the ecommerce workload
  - Open source dataset from REES46





aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What are we building in the next 40 minutes?

- End-to-end data engineering pipeline
  - Simulate the ecommerce workload
  - Open source dataset from REES46
- Online processing
  - DDoS attack (fraudulent transactions)
  - System alerting and notification









aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What are we building in the next 40 minutes?

- End-to-end data engineering pipeline
  - Simulate the ecommerce workload
  - Open source dataset from REES46
- Online processing
  - DDoS attack (fraudulent transactions)
  - System alerting and notification
- Batch processing
  - Top/bottom selling products
  - Dashboarding for business insights
  - Identify new product category











aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



Amazon Kinesis
Data Streams

ONLINE processing

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture





Amazon Kinesis
Data Streams



Amazon Kinesis
Data Analytics

ONLINE processing

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture





Apache Flink

Amazon Kinesis Data Streams

Amazon Kinesis Data Analytics

Amazon Kinesis Data Streams

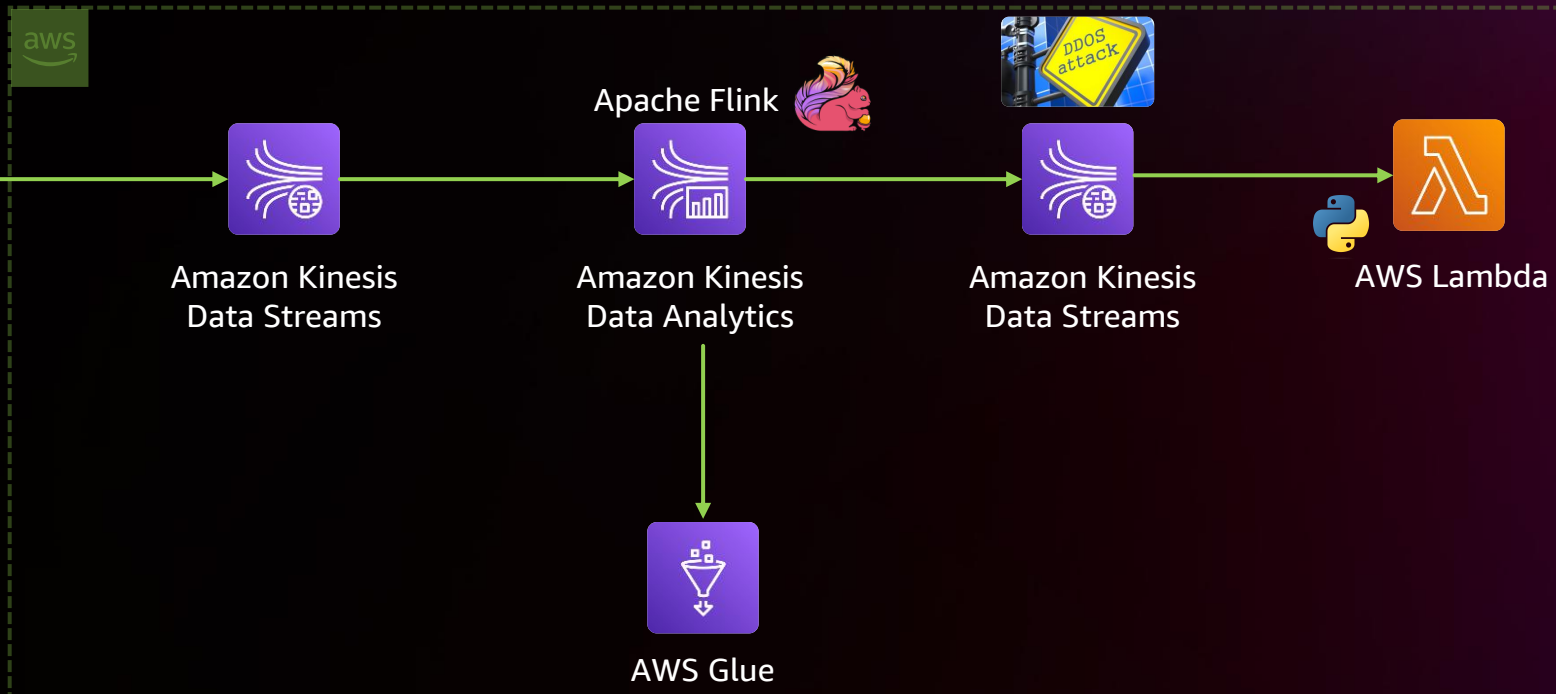
AWS Glue

ONLINE processing

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



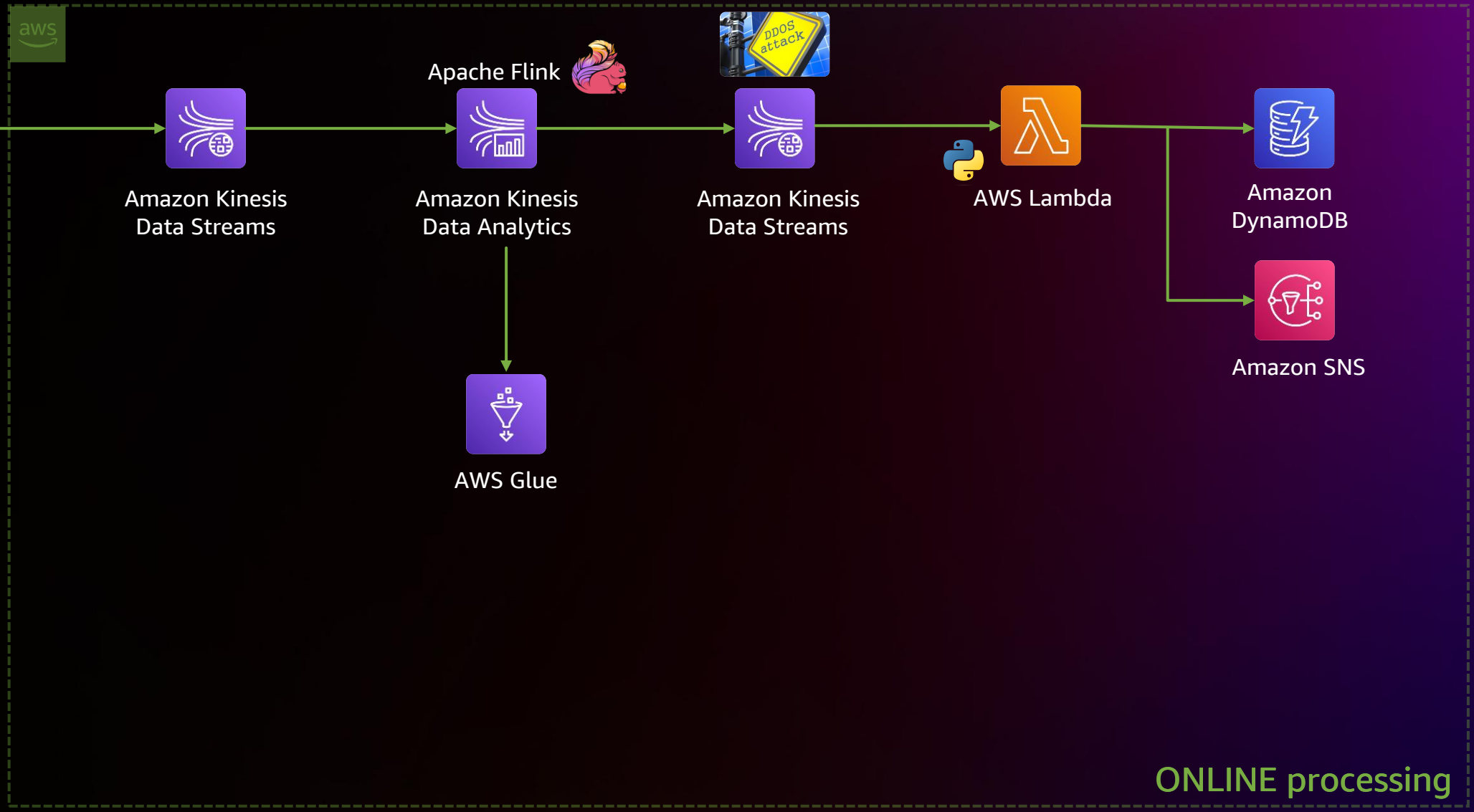


ONLINE processing

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

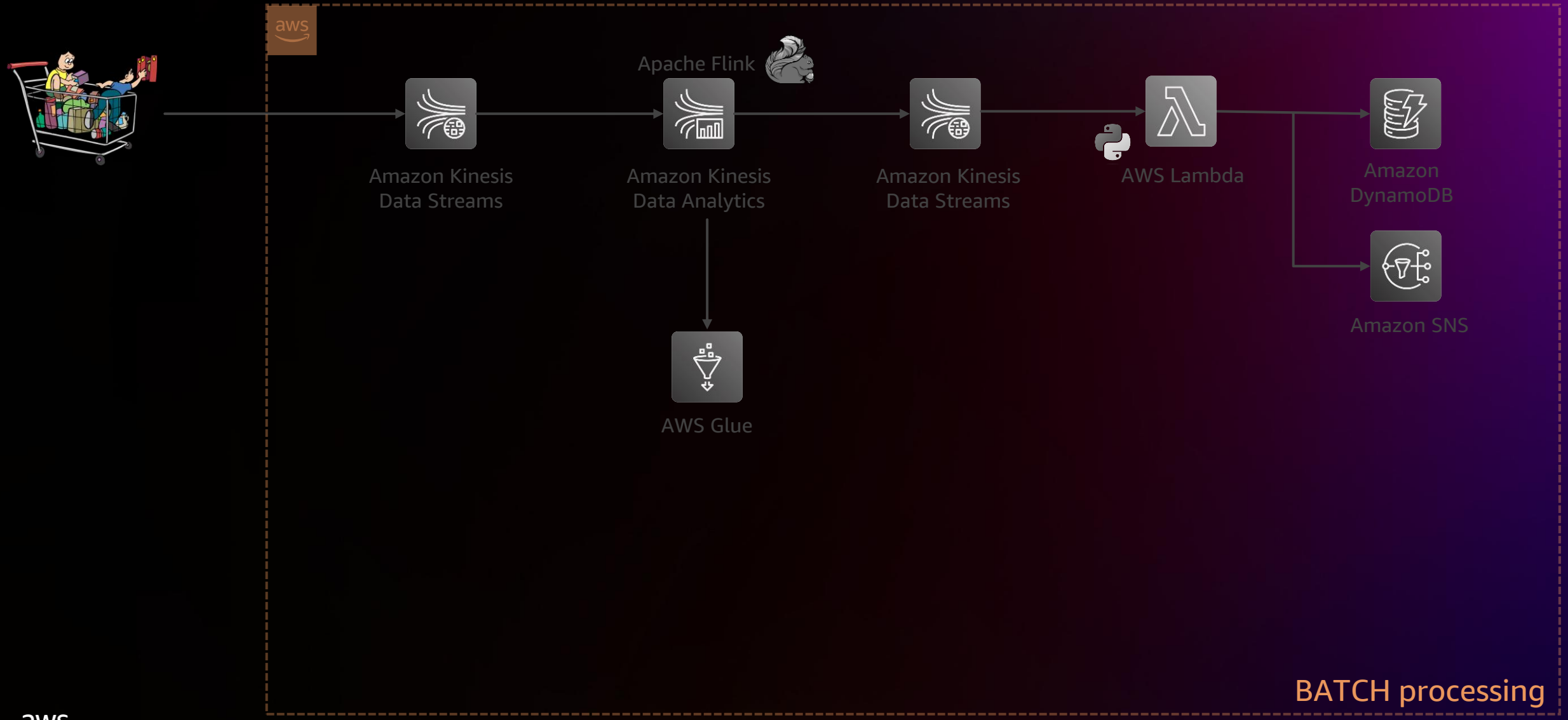
Data engineering architecture





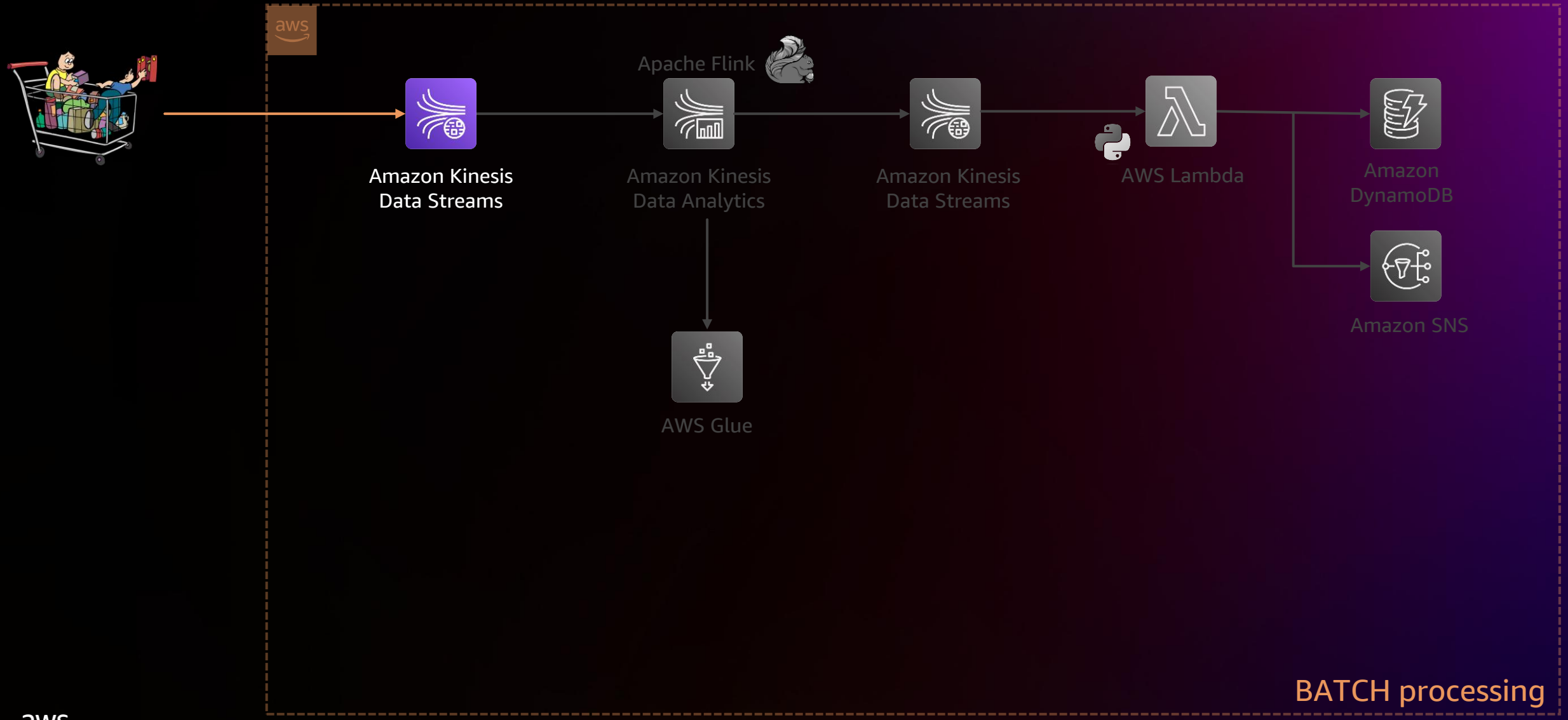
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



Amazon Kinesis Data Streams

Apache Flink

Amazon Kinesis Data Analytics

Amazon Kinesis Data Streams

AWS Lambda

Amazon DynamoDB

Amazon SNS

AWS Glue

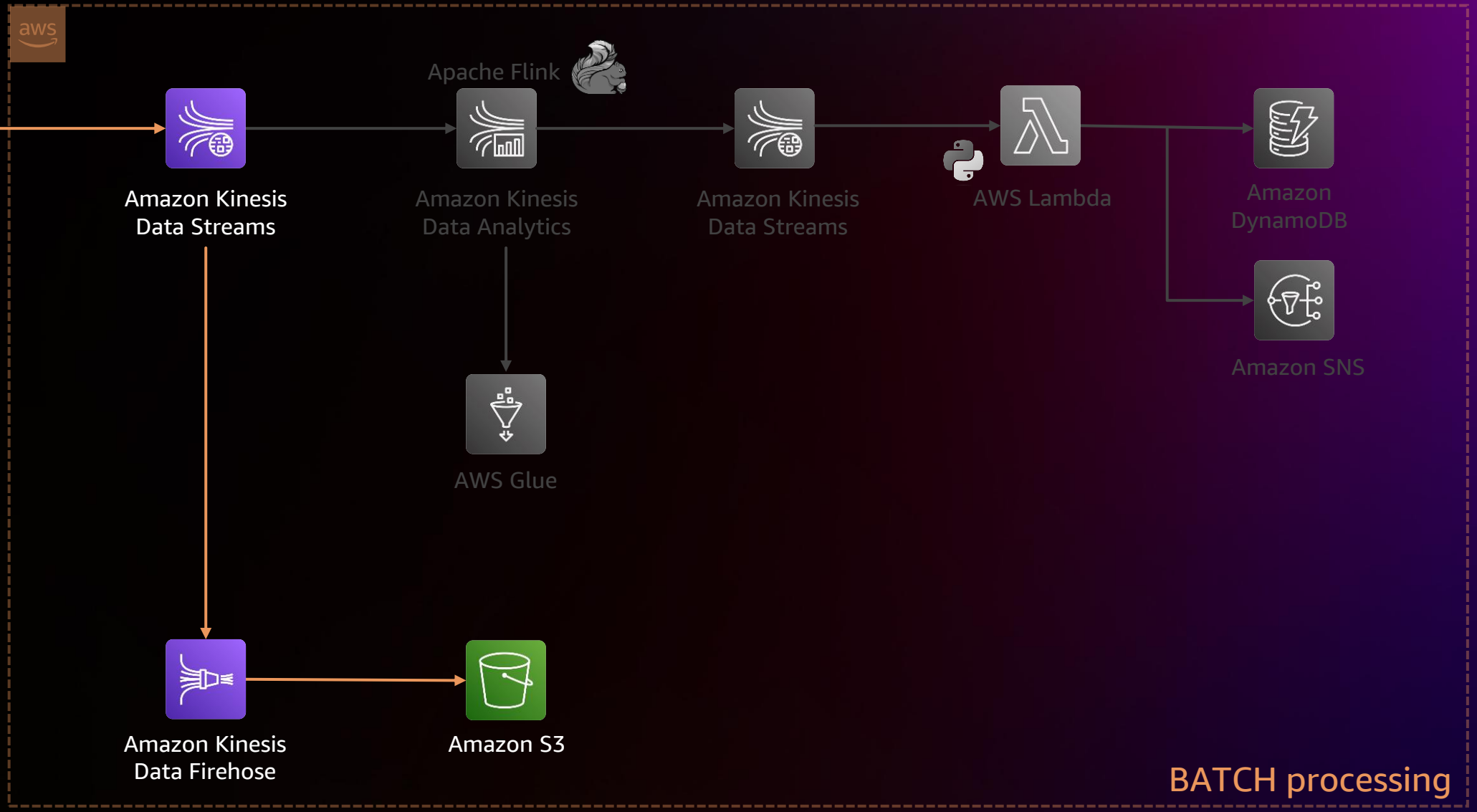
Amazon Kinesis Data Firehose

BATCH processing

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture

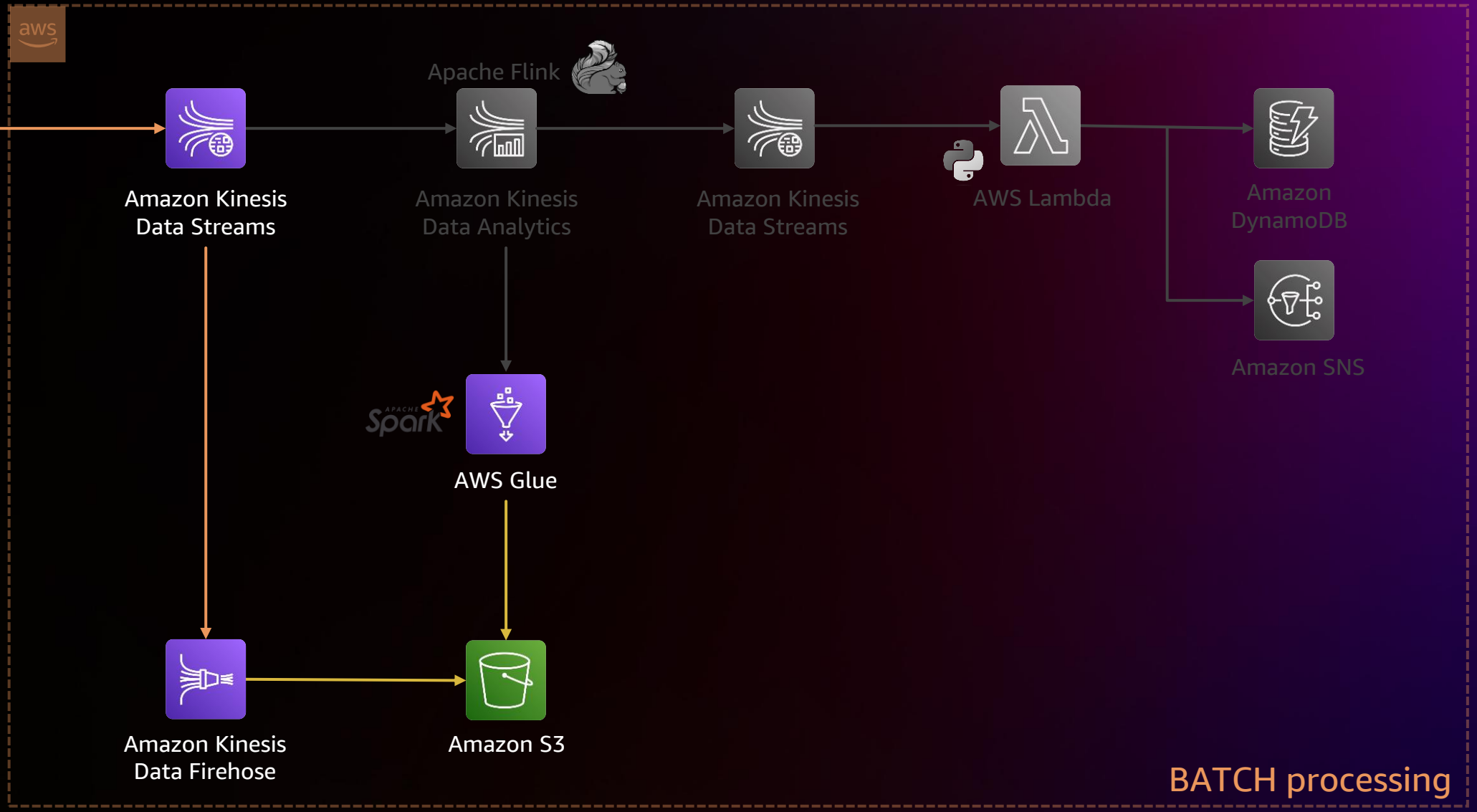




© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture





© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



Amazon Kinesis Data Streams

Apache Flink

Amazon Kinesis Data Analytics

Amazon Kinesis Data Streams

AWS Lambda

Amazon DynamoDB

Amazon SNS

Apache Spark

AWS Glue

Amazon Athena

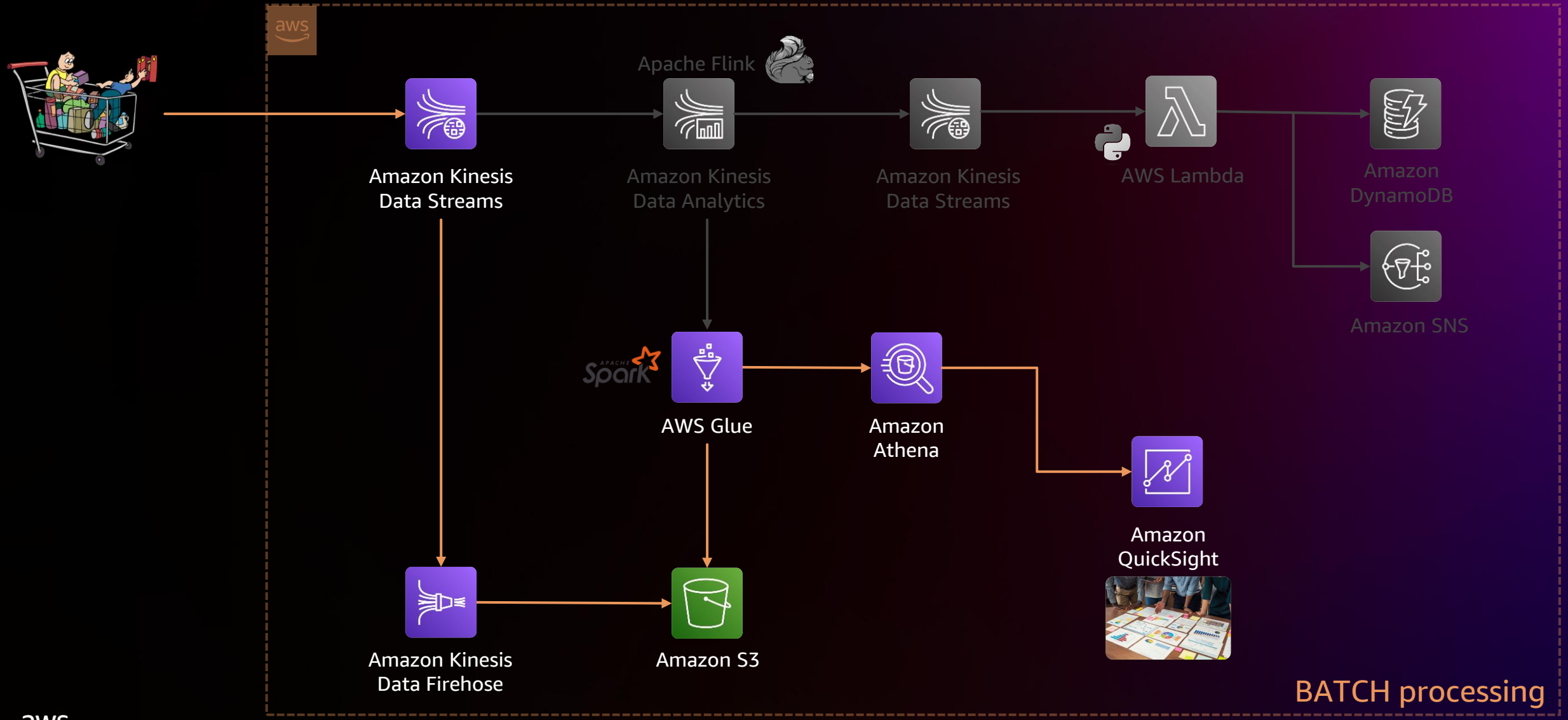
Amazon Kinesis Data Firehose

Amazon S3

BATCH processing

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ecommerce workload simulation



Amazon Kinesis Data Streams

Apache Flink

Amazon Kinesis Data Analytics

Amazon Kinesis Data Streams

AWS Lambda

Amazon DynamoDB

Amazon SNS

Apache Spark

AWS Glue

Amazon Athena

Amazon QuickSight



Amazon Kinesis Data Firehose

Amazon S3

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ecommerce workload simulation



```python
def stream_data_simulator(input_s3_bucket, input_s3_key):
    s3_bucket = input_s3_bucket
    s3_key = input_s3_key

    # Read CSV Lines and split the file into lines
    csv_file = s3_resource.Object(s3_bucket, s3_key)
    s3_response = csv_file.get()
    lines = s3_response['Body'].read().decode('utf-8').split('\n')

    for row in csv.DictReader(lines):
        try:
            # Convert to JSON, to make it easier to work in Kinesis Analytics
            line_json = json.dumps(row)
            json_load = json.loads(line_json)

            # Adding fake txn ts:
            json_load['txn_timestamp'] = datetime.now().isoformat()
            # print(json_load)

            # Write to Kinesis Streams:
            response = kinesis_client.put_record(StreamName=kinesis_stream_name, Data=json.dumps(json_load, indent=4),
                                                 PartitionKey=str(json_load[streaming_partition_key]))
            # response['category_code'] = json_load['category_code']
            print('HttpStatusCode:', response['ResponseMetadata']['HTTPStatusCode'], ', ', json_load['category_code'])
            # print(response)

            # Adding a temporary pause, for demo-purposes:
            sleep(0.5)

        except Exception as e:
            print('Error: {}'.format(e))

```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ecommerce workload simulation



AWS Cloud9





Amazon S3

```python
def stream_data_simulator(input_s3_bucket, input_s3_key):
    s3_bucket = input_s3_bucket
    s3_key = input_s3_key

    # Read CSV Lines and split the file into lines
    csv_file = s3_resource.Object(s3_bucket, s3_key)
    s3_response = csv_file.get()
    lines = s3_response['Body'].read().decode('utf-8').split('\n')

    for row in csv.DictReader(lines):
        try:
            # Convert to JSON, to make it easier to work in Kinesis Analytics
            line_json = json.dumps(row)
            json_load = json.loads(line_json)

            # Adding fake txn ts:
            json_load['txn_timestamp'] = datetime.now().isoformat()
            # print(json_load)

            # Write to Kinesis Streams:
            response = kinesis_client.put_record(StreamName=kinesis_stream_name, Data=json.dumps(json_load, indent=4),
                                                 PartitionKey=str(json_load[streaming_partition_key]))
            # response['category_code'] = json_load['category_code']
            print('HttpStatusCode:', response['ResponseMetadata']['HTTPStatusCode'], ', ', json_load['category_code'])
            # print(response)

            # Adding a temporary pause, for demo-purposes:
            sleep(0.5)

        except Exception as e:
            print('Error: {}'.format(e))

```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ecommerce workload simulation



AWS Cloud9





Amazon S3

```python
def stream_data_simulator(input_s3_bucket, input_s3_key):
    s3_bucket = input_s3_bucket
    s3_key = input_s3_key

    # Read CSV Lines and split the file into lines
    csv_file = s3_resource.Object(s3_bucket, s3_key)
    s3_response = csv_file.get()
    lines = s3_response['Body'].read().decode('utf-8').split('\n')

    for row in csv.DictReader(lines):
        try:
            # Convert to JSON, to make it easier to work in Kinesis Analytics
            line_json = json.dumps(row)
            json_load = json.loads(line_json)

            # Adding fake txn ts:
            json_load['txn_timestamp'] = datetime.now().isoformat()
            # print(json_load)

            # Write to Kinesis Streams:
            response = kinesis_client.put_record(StreamName=kinesis_stream_name, Data=json.dumps(json_load, indent=4),
                                                 PartitionKey=str(json_load[streaming_partition_key]))
            # response['category_code'] = json_load['category_code']
            print('HttpStatusCode:', response['ResponseMetadata']['HTTPStatusCode'], ', ', json_load['category_code'])
            # print(response)

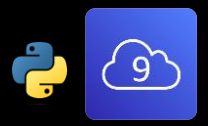
            # Adding a temporary pause, for demo-purposes:
            sleep(0.5)

        except Exception as e:
            print('Error: {}'.format(e))

```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ecommerce workload simulation



AWS Cloud9



Amazon S3

```python
def stream_data_simulator(input_s3_bucket, input_s3_key):
    s3_bucket = input_s3_bucket
    s3_key = input_s3_key

    # Read CSV Lines and split the file into lines
    csv_file = s3_resource.Object(s3_bucket, s3_key)
    s3_response = csv_file.get()
    lines = s3_response['Body'].read().decode('utf-8').split('\n')

    for row in csv.DictReader(lines):
        try:
            # Convert to JSON, to make it easier to work in Kinesis Analytics
            line_json = json.dumps(row)
            json_load = json.loads(line_json)

            # Adding fake txn ts:
            json_load['txn_timestamp'] = datetime.now().isoformat()
            # print(json_load)

            # Write to Kinesis Streams:
            response = kinesis_client.put_record(StreamName=kinesis_stream_name, Data=json.dumps(json_load, indent=4),
                                                 PartitionKey=str(json_load[streaming_partition_key]))
            # response['category_code'] = json_load['category_code']
            print('HttpStatusCode:', response['ResponseMetadata']['HTTPStatusCode'], ', ', json_load['category_code'])
            # print(response)

            # Adding a temporary pause, for demo-purposes:
            sleep(0.5)

        except Exception as e:
            print('Error: {}'.format(e))
```

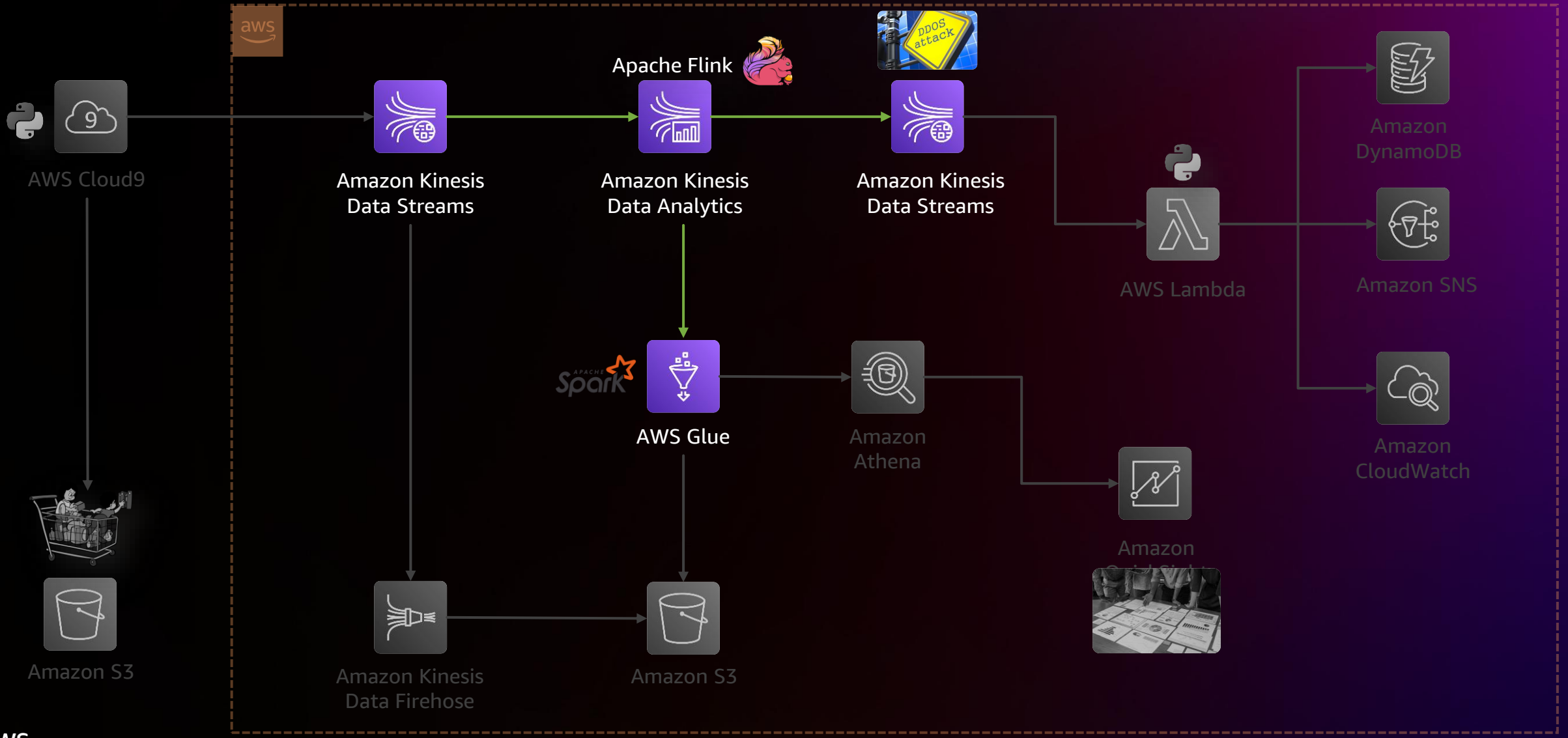


Amazon Kinesis Data Streams

aws

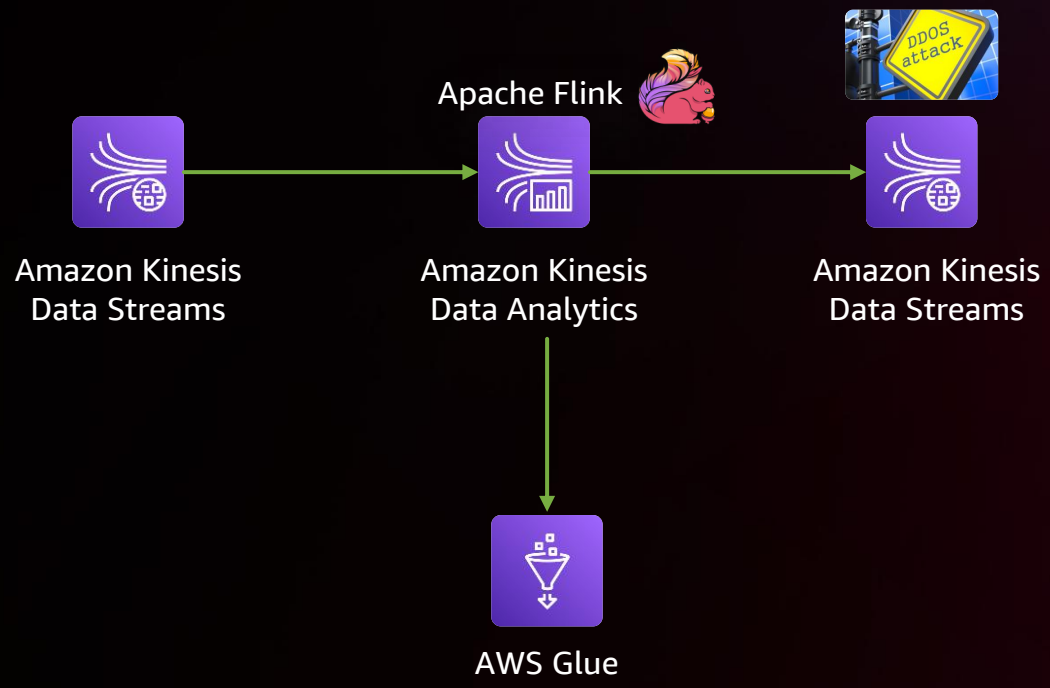
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Streaming application



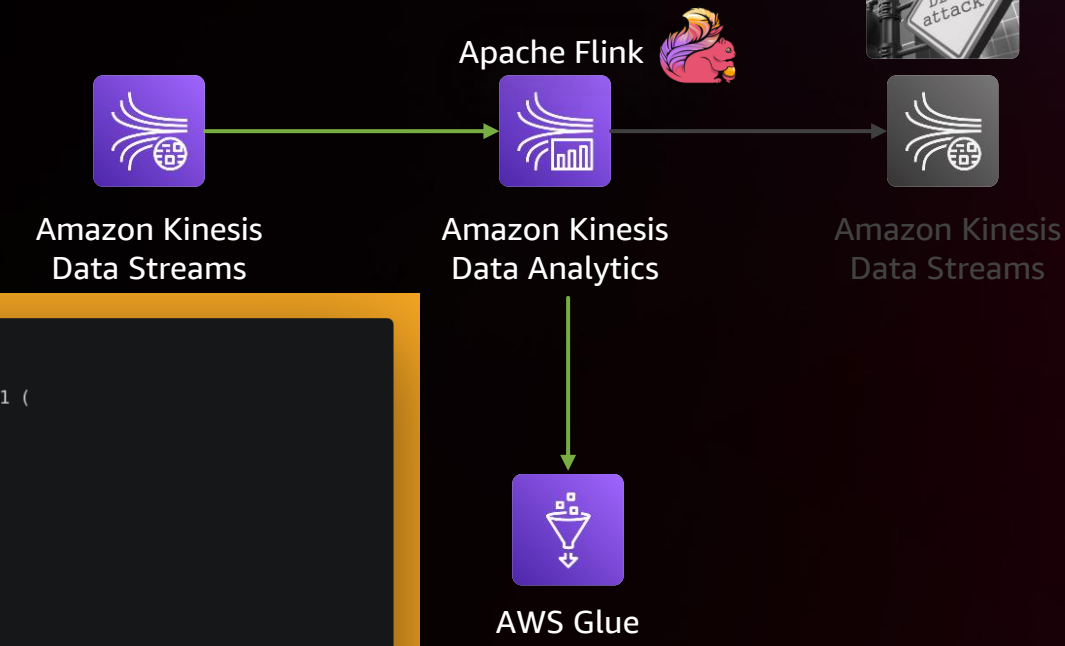
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Streaming application



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

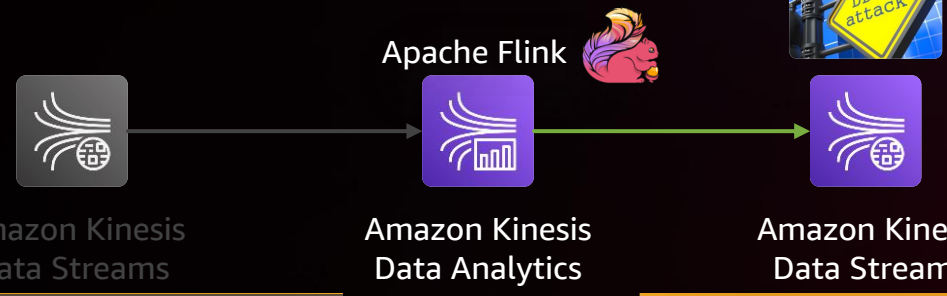
Streaming application




```sql
CREATE TABLE ecomm_user_activity_stream_1 (
  `event_time` VARCHAR(30),
  `event_type` VARCHAR(30),
  `product_id` BIGINT,
  `category_id` BIGINT,
  `category_code` VARCHAR(30),
  `brand` VARCHAR(30),
  `price` DOUBLE,
  `user_id` BIGINT,
  `user_session` VARCHAR(30),
  `txn_timestamp` TIMESTAMP(3),
  WATERMARK FOR txn_timestamp as txn_timestamp - INTERVAL '10' SECOND
)
PARTITIONED BY (category_id)
WITH (
  'connector' = 'kinesis',
  'stream' = 'ecommerce-raw-user-activity-stream-1',
  'aws.region' = 'us-east-1',
  'scan.stream.initpos' = 'LATEST',
  'format' = 'json',
  'json.timestamp-format.standard' = 'ISO-8601'
);
```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Streaming application

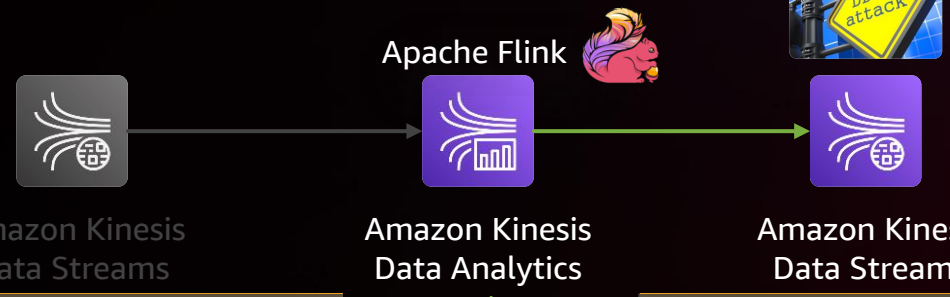



```
CREATE TABLE ecomm_user_activity_stream_1 (
  `event_time` VARCHAR(30),
  `event_type` VARCHAR(30),
  `product_id` BIGINT,
  `category_id` BIGINT,
  `category_code` VARCHAR(30),
  `brand` VARCHAR(30),
  `price` DOUBLE,
  `user_id` BIGINT,
  `user_session` VARCHAR(30),
  `txn_timestamp` TIMESTAMP(3),
  WATERMARK FOR txn_timestamp as txn_timestamp - INTERVAL '10' SECOND
)
PARTITIONED BY (category_id)
WITH (
  'connector' = 'kinesis',
  'stream' = 'ecommerce-raw-user-activity-stream-1',
  'aws.region' = 'us-east-1',
  'scan.stream.initpos' = 'LATEST',
  'format' = 'json',
  'json.timestamp-format.standard' = 'ISO-8601'
);
```

```
CREATE TABLE ecomm_user_activity_stream_2 (
  `user_id` BIGINT,
  `num_actions_per_watermark` BIGINT
)
WITH (
  'connector' = 'kinesis',
  'stream' = 'ecommerce-raw-user-activity-stream-2',
  'aws.region' = 'us-east-1',
  'format' = 'json',
  'json.timestamp-format.standard' = 'ISO-8601'
);
```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Streaming application




```
CREATE TABLE ecomm_user_activity_stream_1 (
  `event_time` VARCHAR(30),
  `event_type` VARCHAR(30),
  `product_id` BIGINT,
  `category_id` BIGINT,
  `category_code` VARCHAR(30),
  `brand` VARCHAR(30),
  `price` DOUBLE,
  `user_id` BIGINT,
  `user_session` VARCHAR(30),
  `txn_timestamp` TIMESTAMP(3),
  WATERMARK FOR txn_timestamp as txn_timestamp - INTERVAL '10' SECOND
)
PARTITIONED BY (category_id)
WITH (
  'connector' = 'kinesis',
  'stream' = 'ecommerce-raw-user-activity-stream-1',
  'aws.region' = 'us-east-1',
  'scan.stream.initpos' = 'LATEST',
  'format' = 'json',
  'json.timestamp-format.standard' = 'ISO-8601'
);
```

```
CREATE TABLE ecomm_user_activity_stream_2 (
  `user_id` BIGINT,
  `num_actions_per_watermark` BIGINT
)
WITH (
  'connector' = 'kinesis',
  'stream' = 'ecommerce-raw-user-activity-stream-2',
  'aws.region' = 'us-east-1',
  'format' = 'json',
  'json.timestamp-format.standard' = 'ISO-8601'
);
```

```
insert into ecomm_user_activity_stream_2
select  user_id, count(1) as num_actions_per_watermark
from ecomm_user_activity_stream_1
group by tumble(txn_timestamp, INTERVAL '10' SECOND), user_id
having count(1) > 10;
```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data engineering architecture



Amazon Kinesis Data Streams

Apache Flink

Amazon Kinesis Data Analytics

Amazon Kinesis Data Streams

AWS Lambda

Amazon DynamoDB

Amazon SNS

Apache Spark

AWS Glue

Amazon Athena

Amazon QuickSight



Amazon Kinesis Data Firehose

Amazon S3

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What's next

- **ANT223**
  - Simplify and accelerate data integration and ETL modernization with AWS Glue
- **ANT333**
  - Build a serverless data streaming workload with Amazon Kinesis
- **Hands-on workshops**
  - https://workshops.aws/card/analytics
    - Analyze Toll Data using Serverless Analytics on AWS
    - Streaming Analytics Workshop



https://github.com/debnsuma/boa309

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Rohini Gaonkar

in/rohinigaonkar/

@rohini_gaonkar

Suman Debnath

in/suman-d/

@_sumand



Please complete the session survey in the **mobile app**

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.