

AWS  
re:Invent

# Scalable serverless event-driven applications using Amazon SQS & Lambda

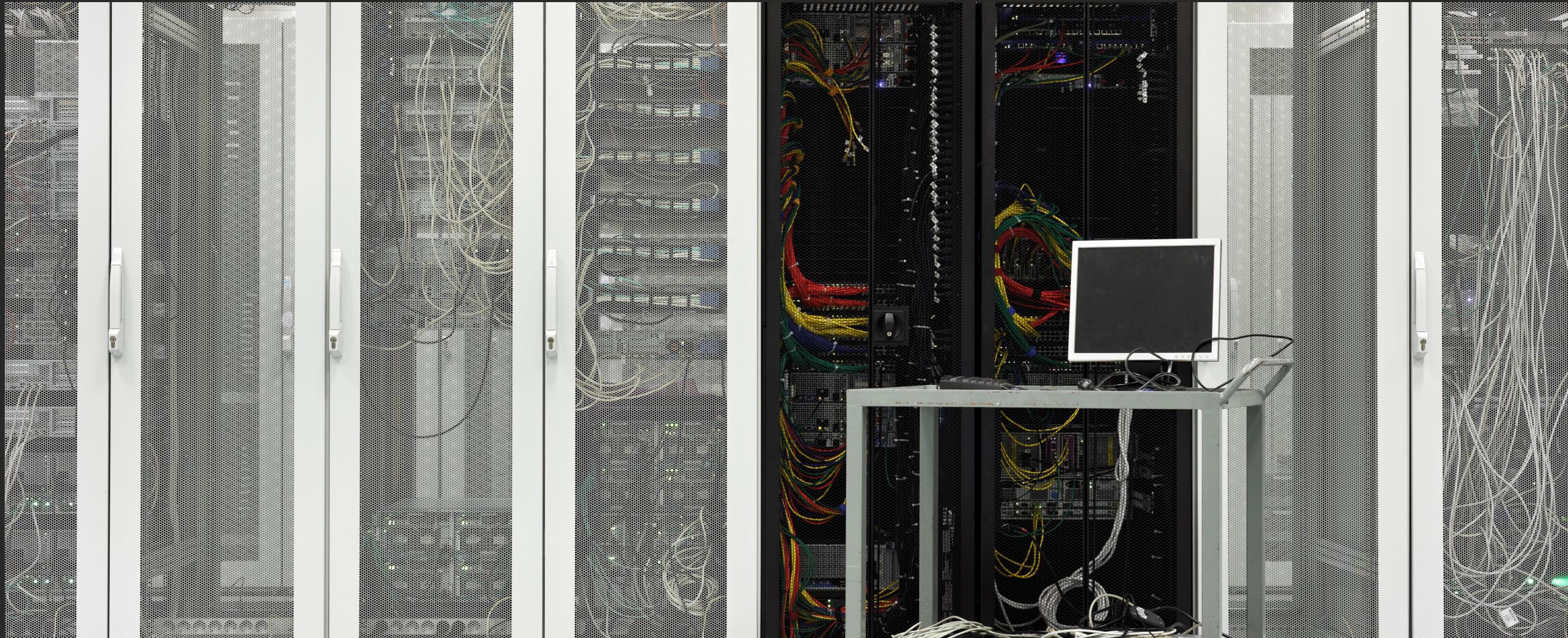
**Luay Kawasme**

Sr. Manager, Head of Engineering  
Amazon Simple Queue Service (SQS)  
Amazon Web Services

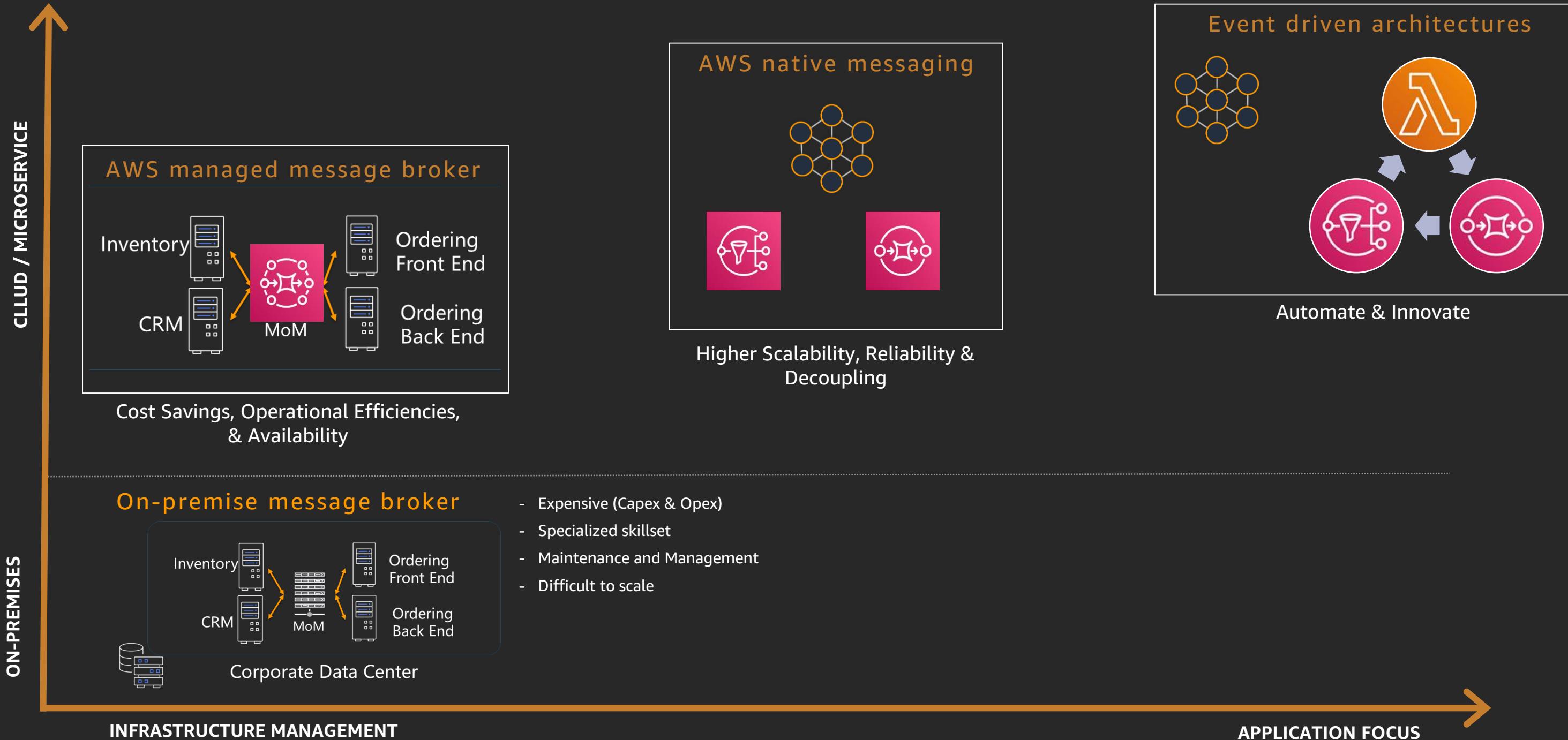
**Rory Richardson**

Head of Business Development,  
Serverless  
Amazon Web Services

# A short history of the messaging broker...



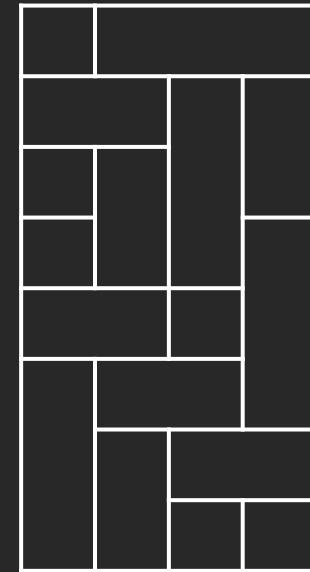
# Modern messaging architectures



# Development transformation at Amazon: 2001–2002

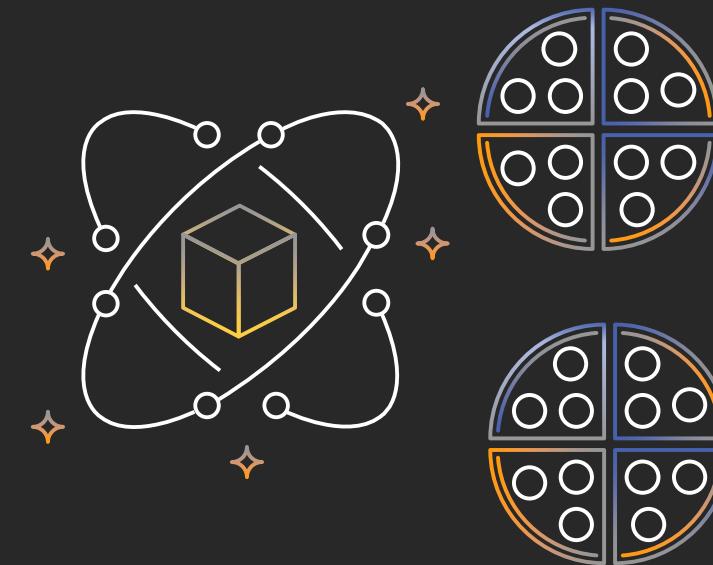
Lesson learned: decompose for agility

2001



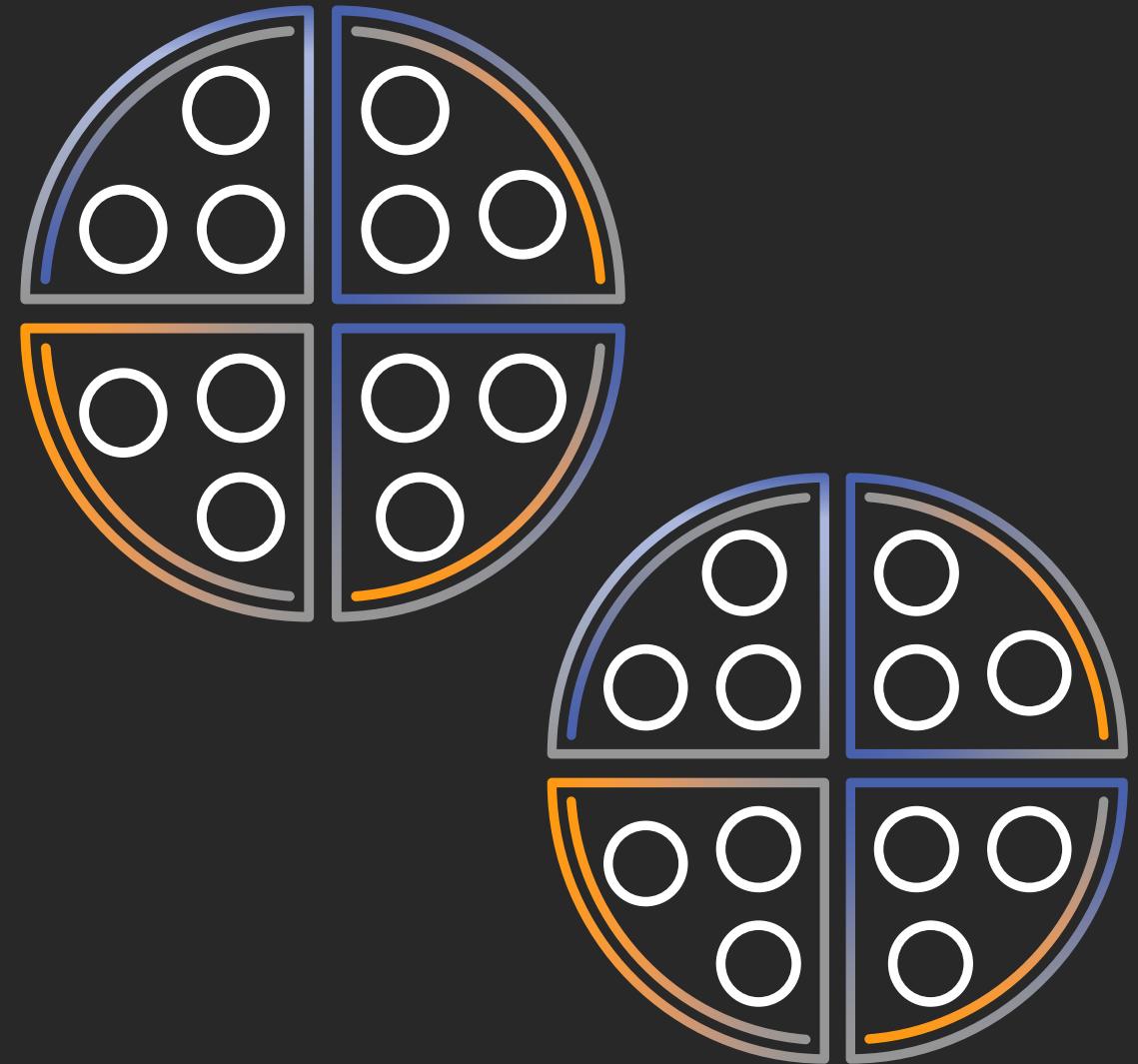
monolithic application  
+ teams

2002



microservices  
+ 2 pizza teams

# Two-pizza teams are fast & agile



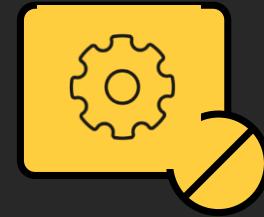
Full ownership & autonomy

You build it, you run it

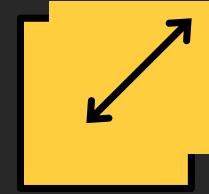
DevOps – small, nimble teams

Focused innovation

# Build microservices with serverless technologies



No infrastructure to provision or manage



Scales automatically by unit of consumption



Pay for value



Built-in security,  
highly available compute

# Why Amazon SQS as an Event Source for AWS Lambda

- Asynchronous programing
- SQS is built for ultimate scale and low latency
- Messaging is the neural network of modern architectures
- SQS is the #1 event trigger for Lambda



# Amazon SQS statistics

25,000,000,000  
messages per hour



# Better together



# Lambda invocation integrations

Amazon S3

Amazon DynamoDB

Amazon Kinesis Data Streams

Amazon SNS

Amazon SES

Amazon SQS

Amazon Cognito

AWS CloudFormation

Amazon CloudWatch Logs

Amazon EventBridge

AWS CodeCommit

AWS Config

Amazon Alexa

Amazon Lex

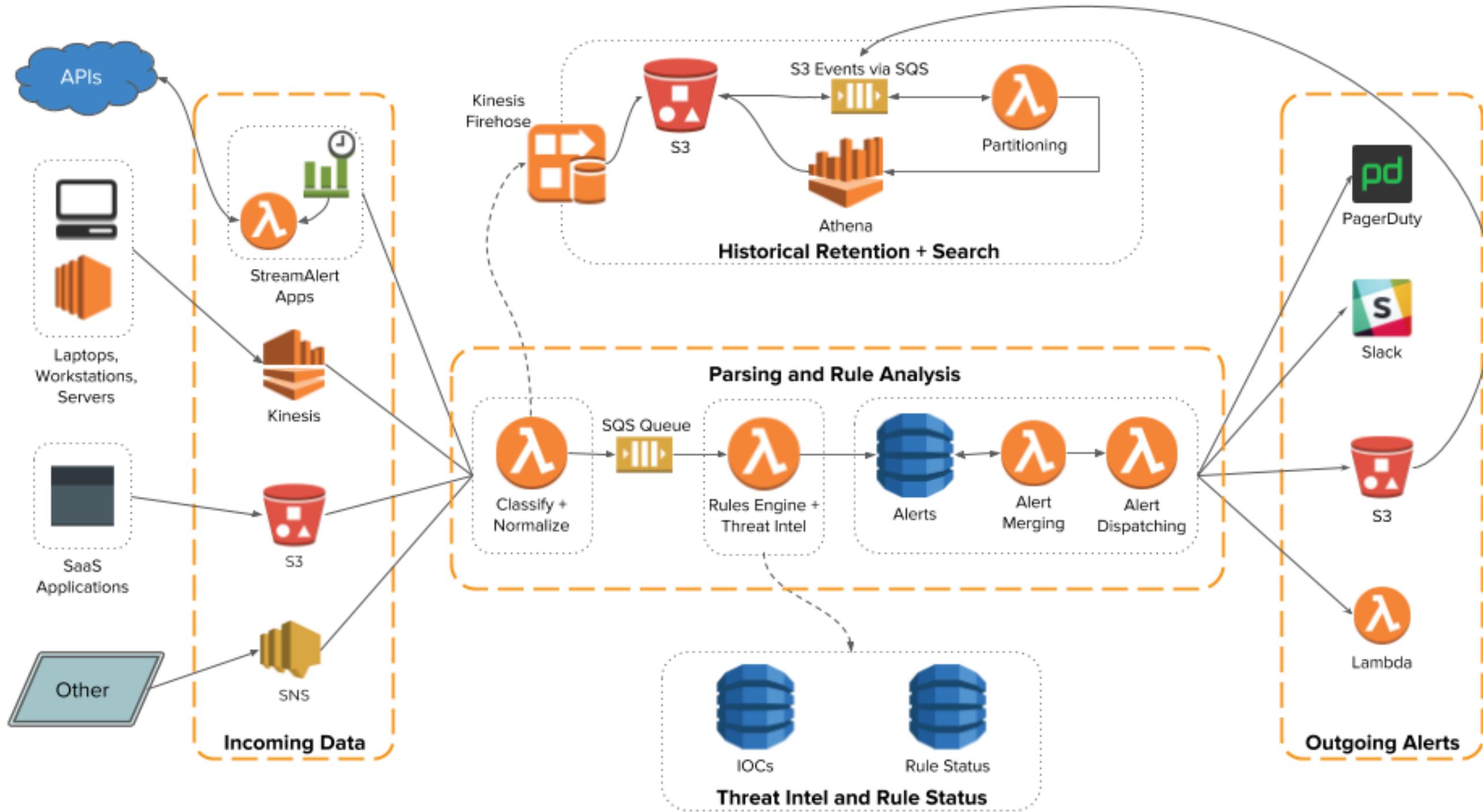
Amazon API Gateway

AWS IoT Button

and more . . .



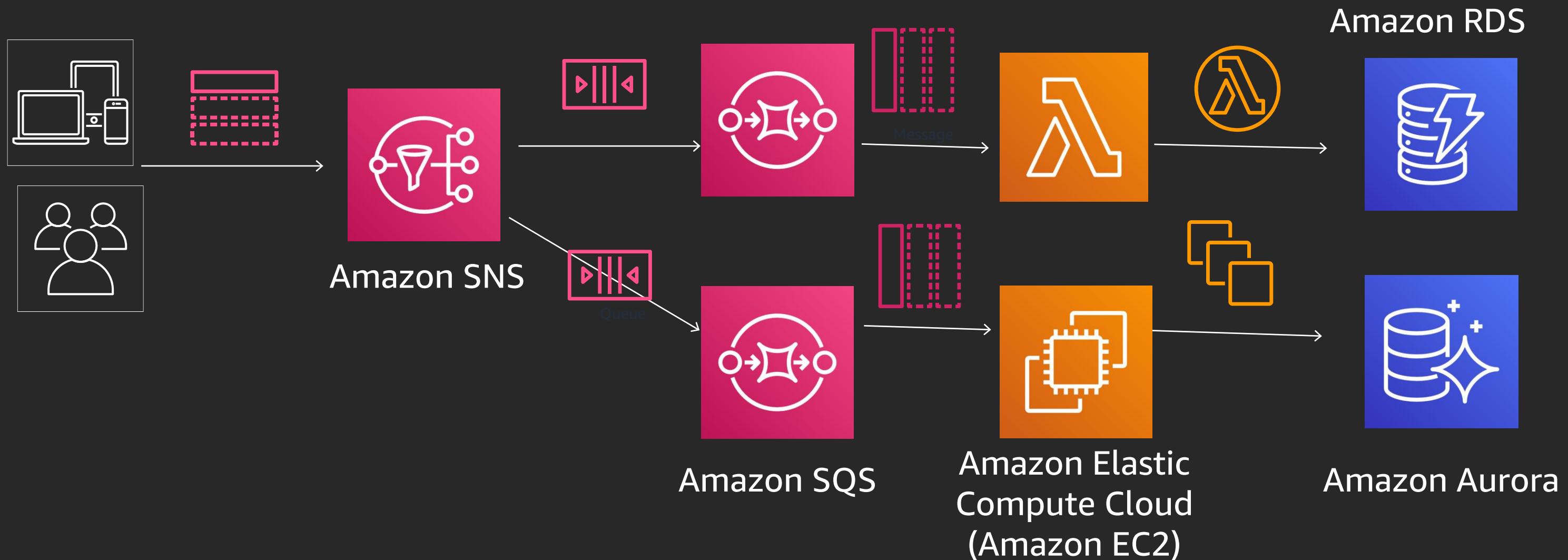
# Amazon SQS + AWS Lambda



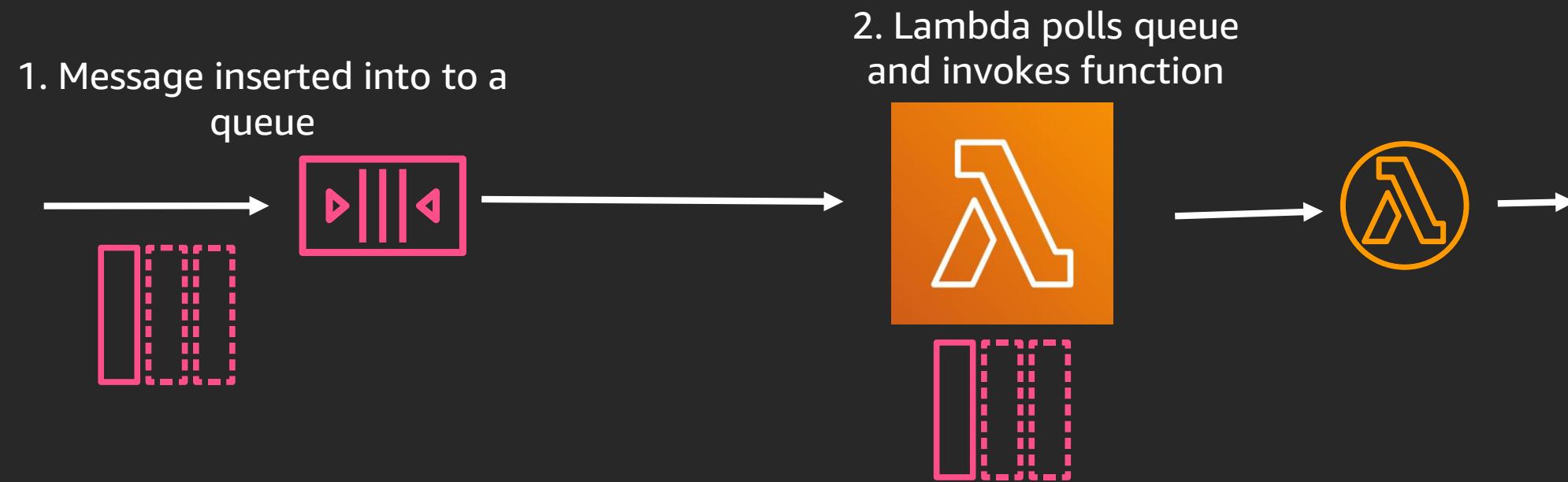
# Serverless message processing

# Enterprise integration pattern - Pipes and Filters

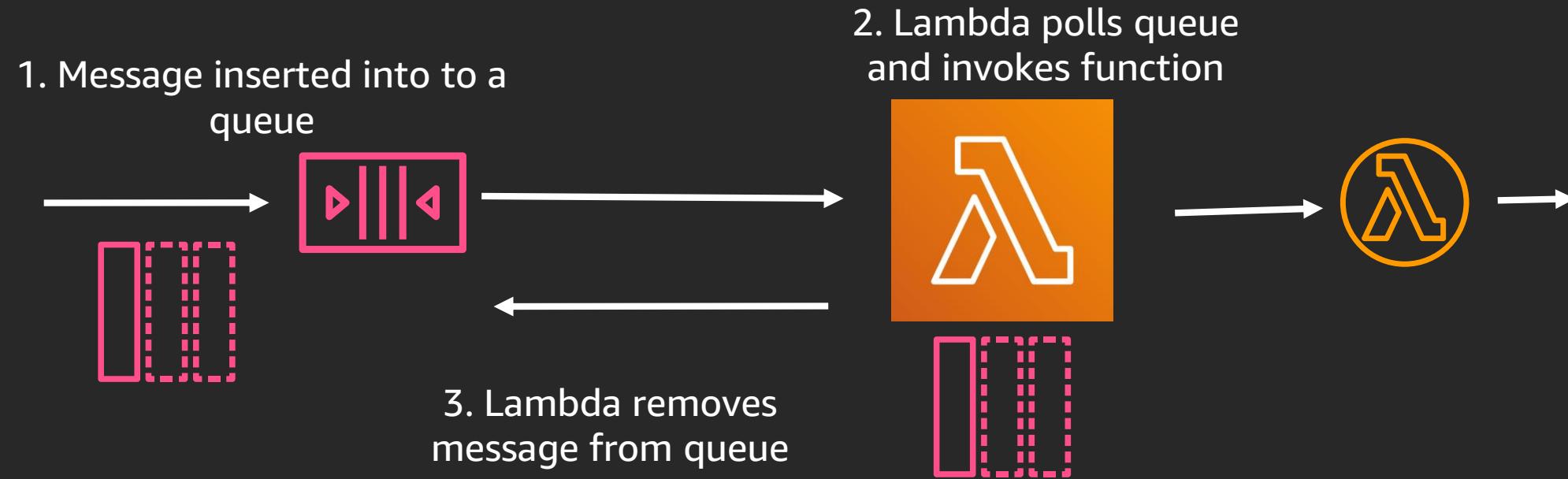
A single event triggers a sequence of processing steps. Typically, an event traverses a pipeline and goes through several transformations (e.g., filtering, enrichment).



# Serverless message processing



# Serverless message processing



# Amazon SQS to Lambda

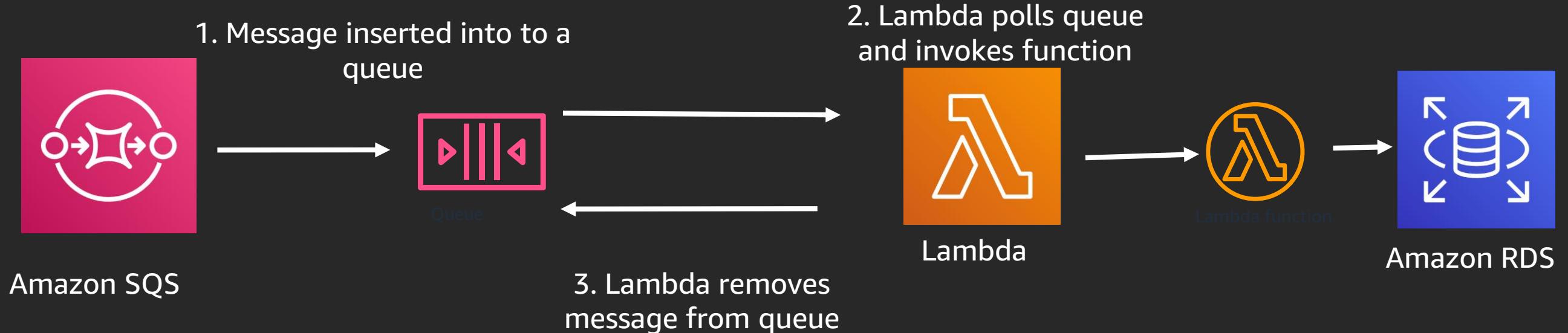
## Message processing

Messages are stored durably

Queue accepts messages at any rate

No need to provision capacity

Visibility timeout for failure processing



## Serverless processing

Lambda polls queue for messages

Processed in batches

# Lambda async push vs. Amazon SQS

## Lambda async invoke

- Direct integration
- Automatic scaling
- No batching
- Simple DLQ policy
- No backlog visibility
- No extra queue features
- No extra cost

## Amazon SQS + Lambda

- Two resources
- Automatic scaling
- Supports batching
- Advanced DLQ policy
- Amazon CloudWatch metrics
- All SQS features
- Cost of requests to SQS

# Advanced Amazon SQS features

Advanced DLQ policy

Batching

Delayed messages

Server-side encryption with customer managed key (CMK)

Retention period

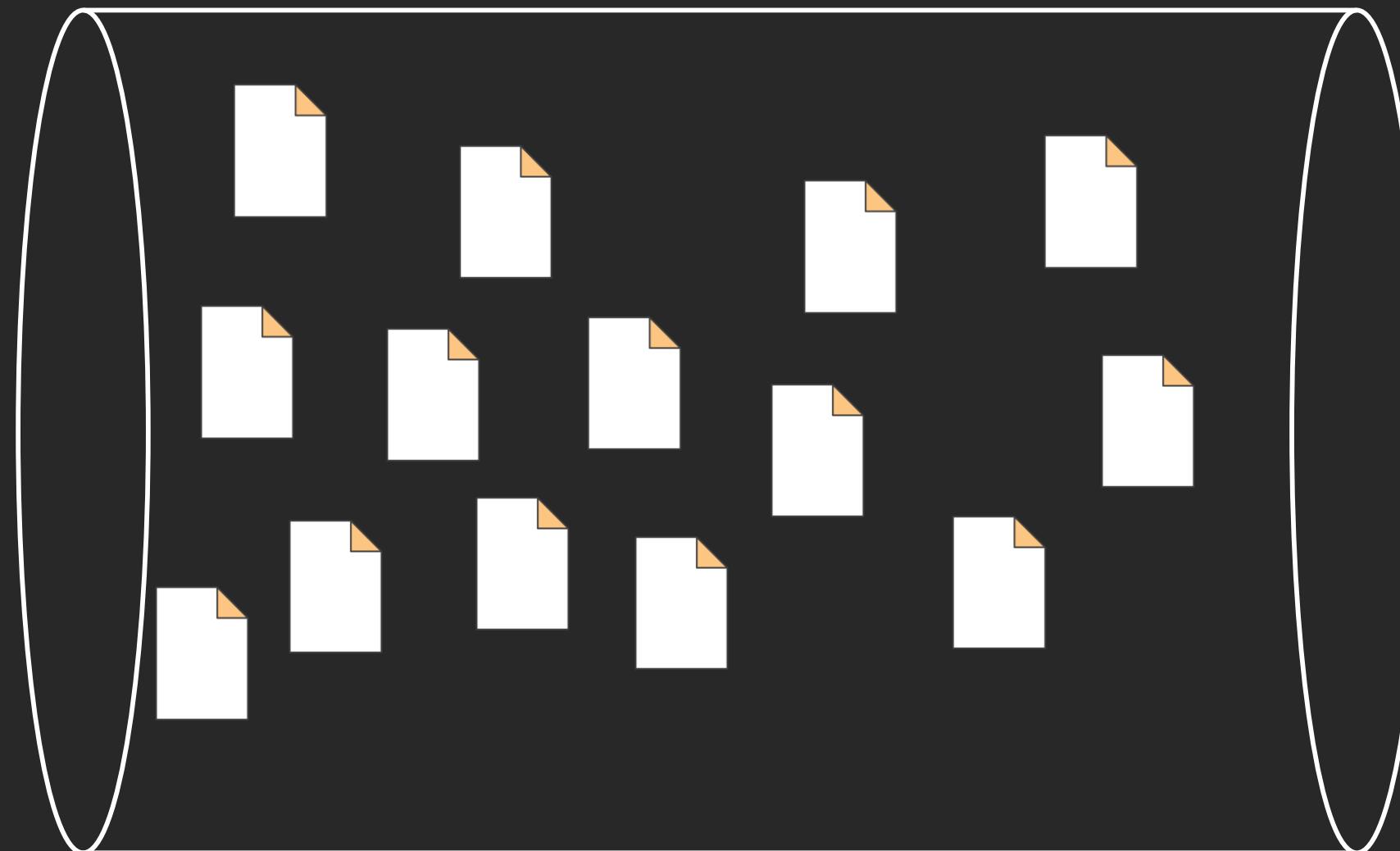
Message attributes

CloudWatch metrics

Purge operation

# Inside Amazon SQS and Lambda

# Consuming from Amazon SQS

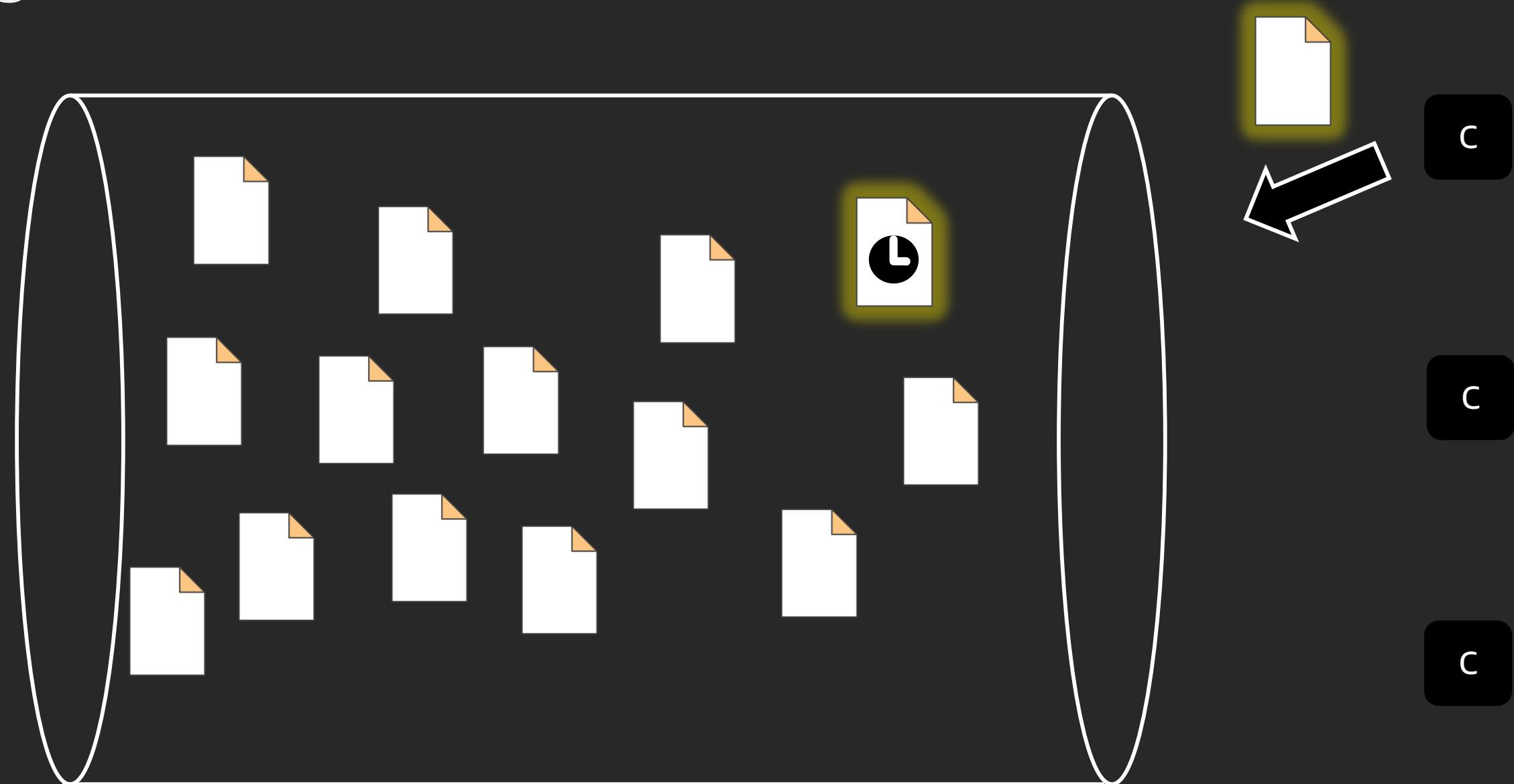


C

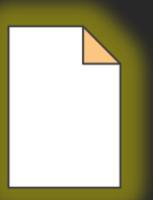
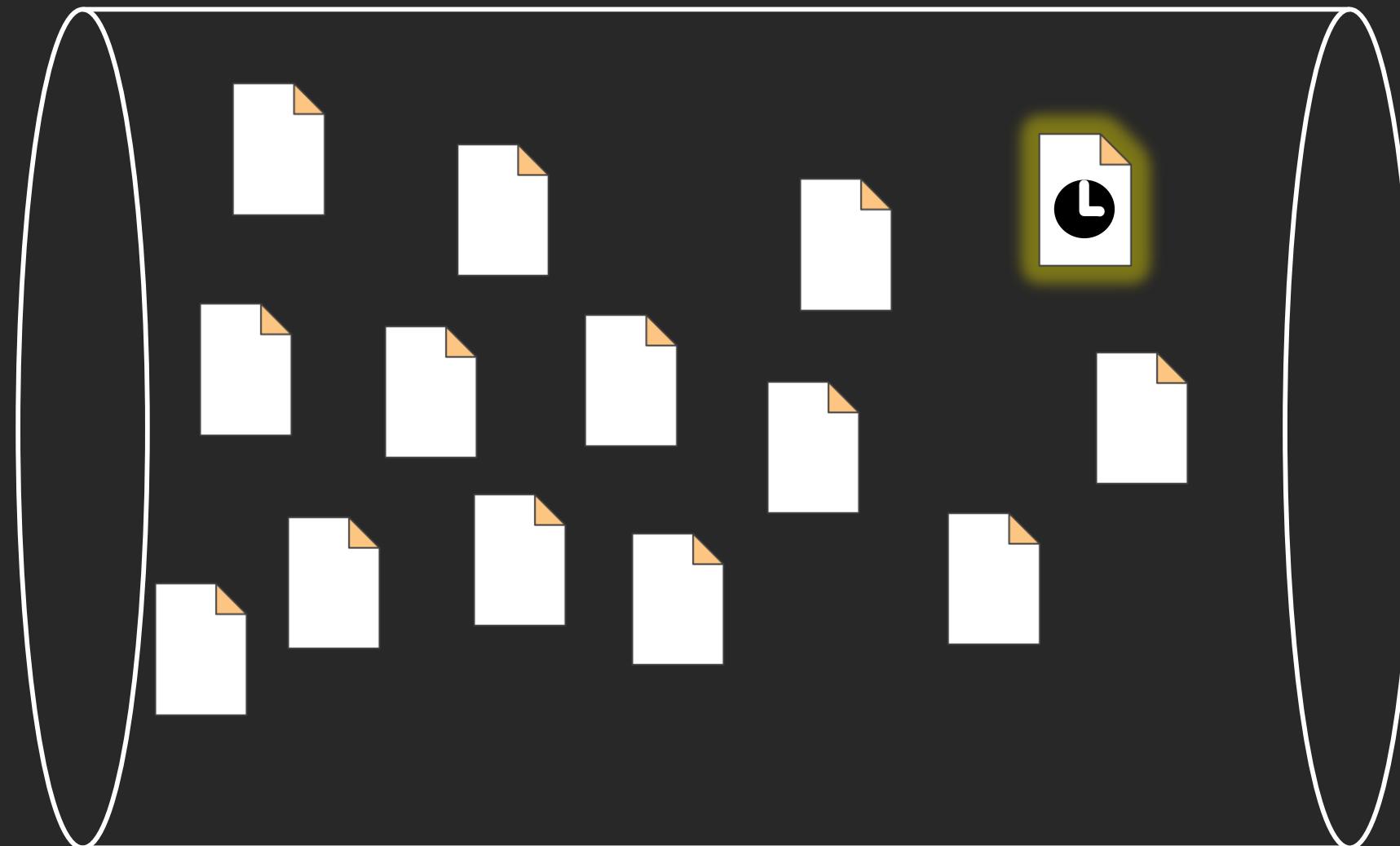
C

C

# Consuming from Amazon SQS



# Consuming from Amazon SQS

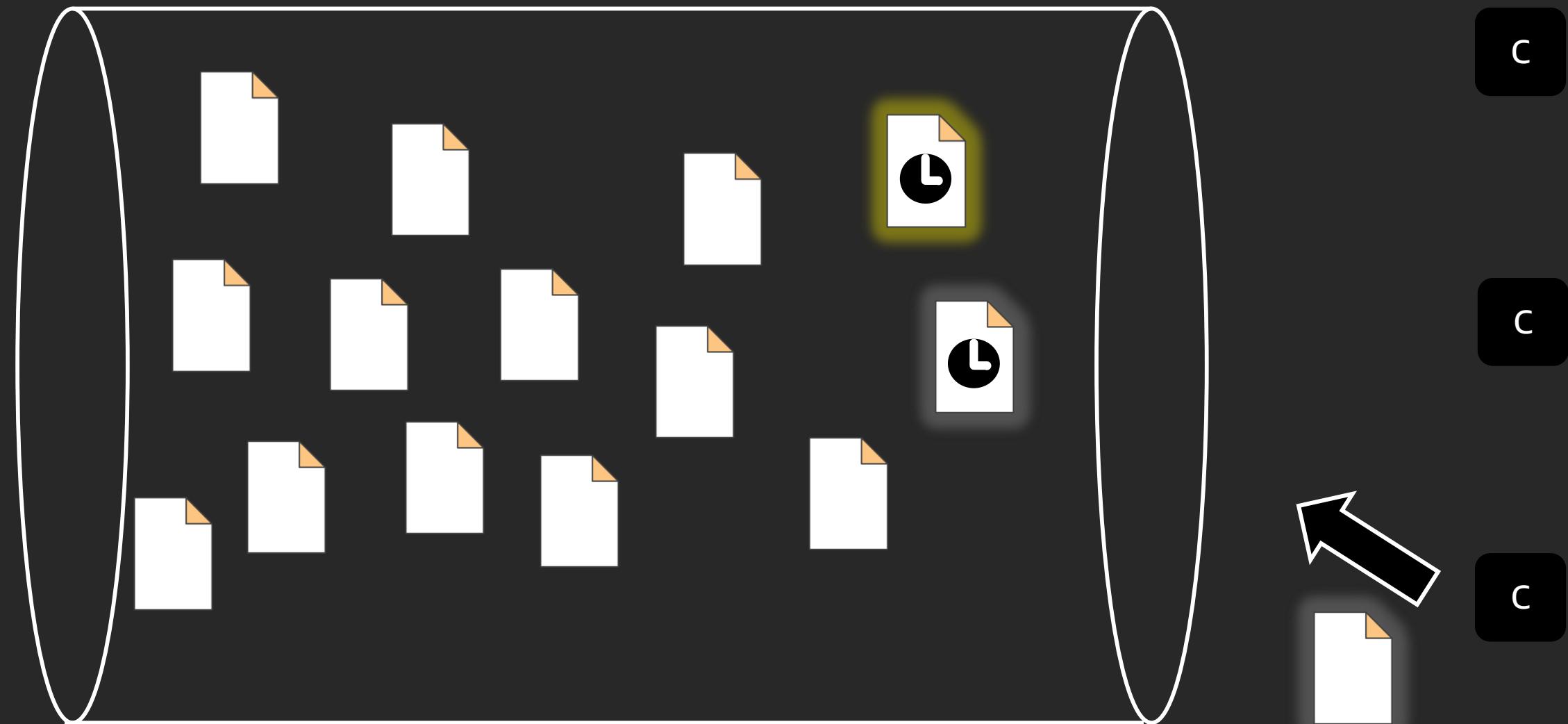


c

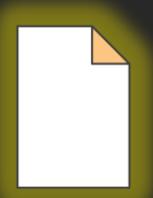
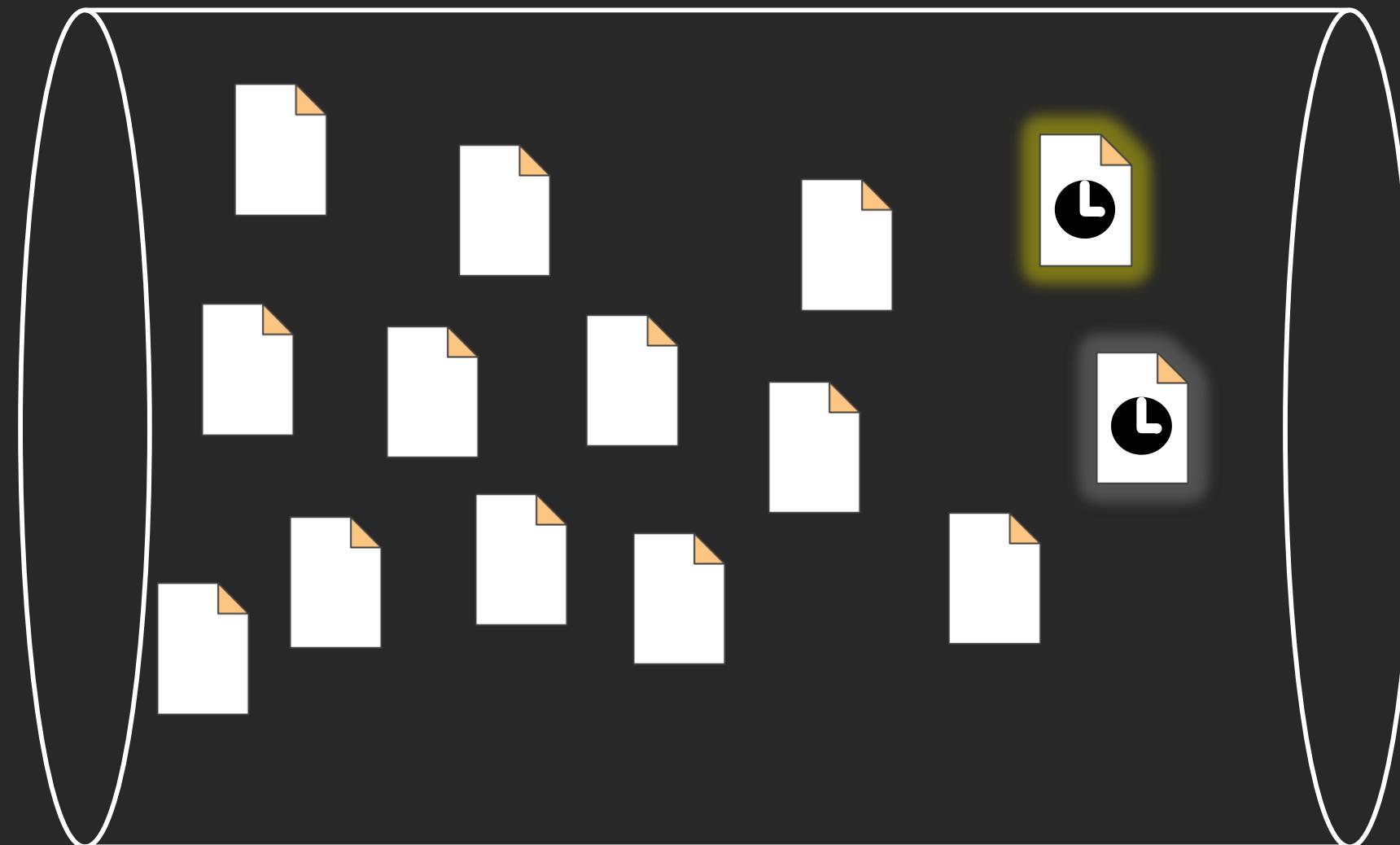
c

c

# Consuming from Amazon SQS



# Consuming from Amazon SQS



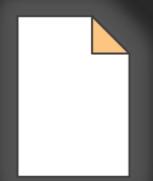
C



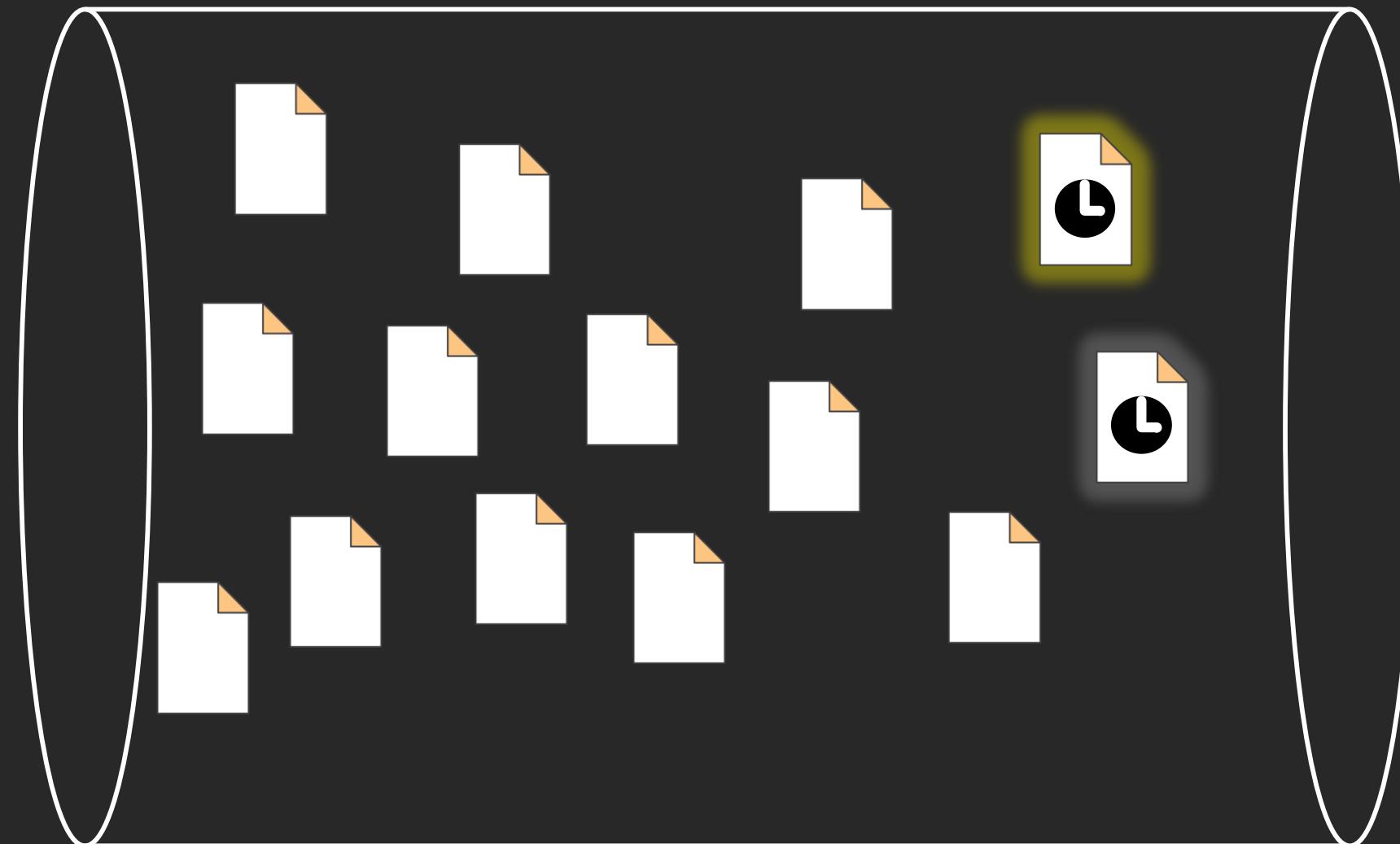
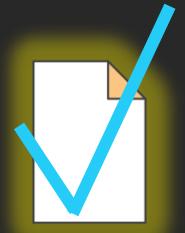
C



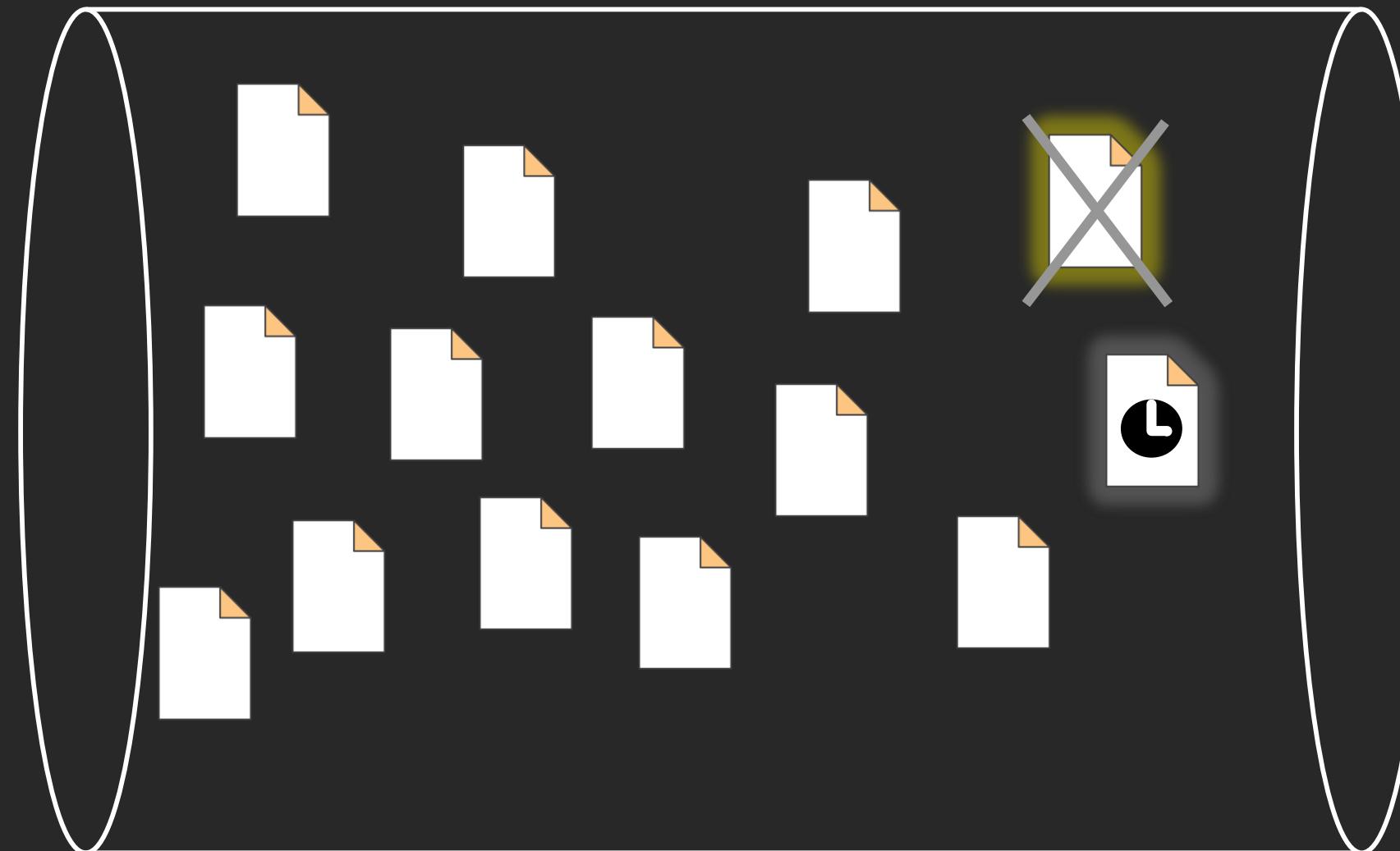
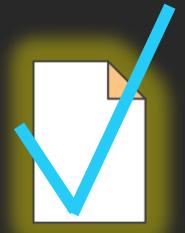
C



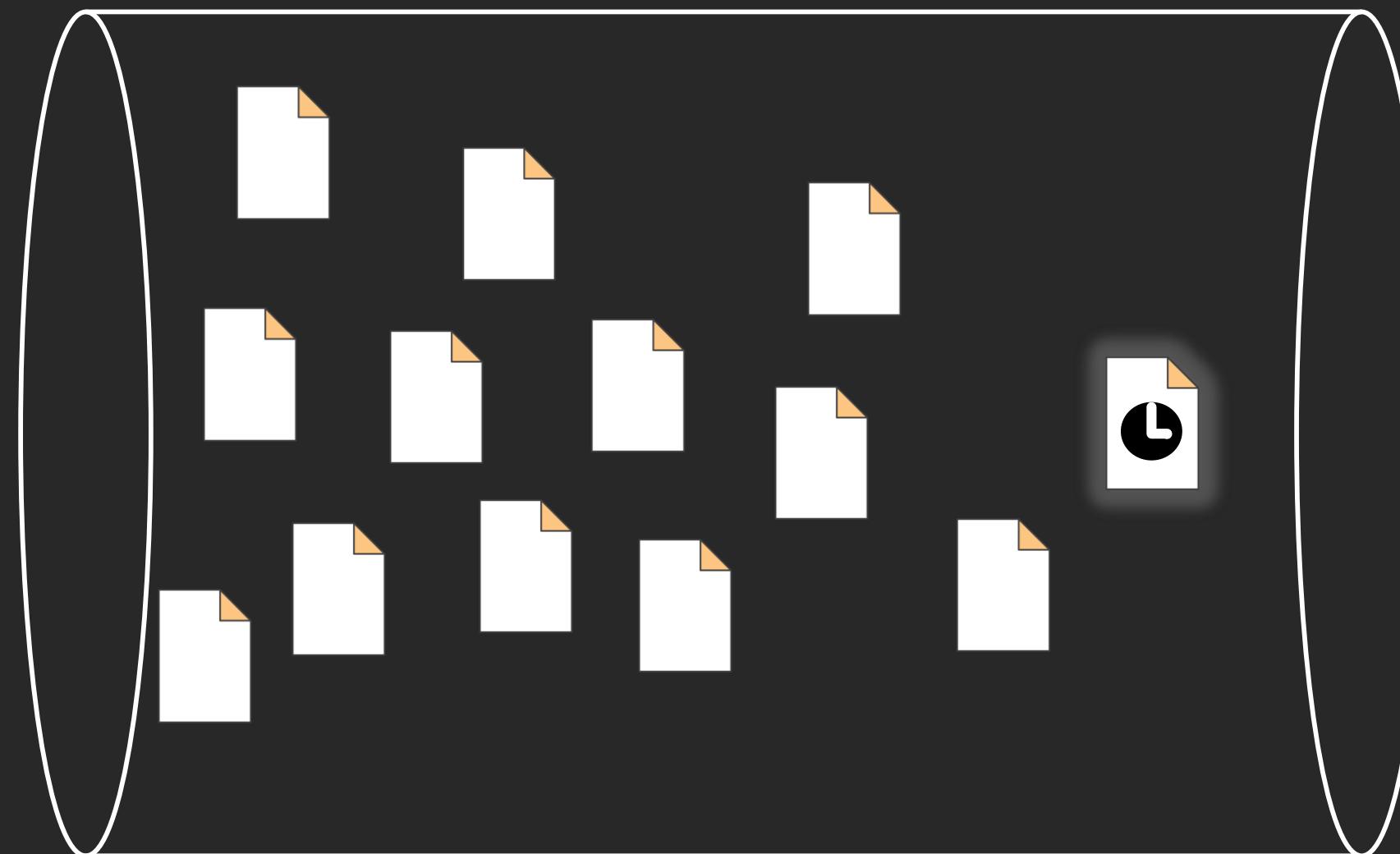
# Consuming from Amazon SQS



# Consuming from Amazon SQS



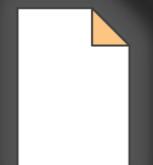
# Consuming from Amazon SQS



C

C

C



# Consuming from Amazon SQS



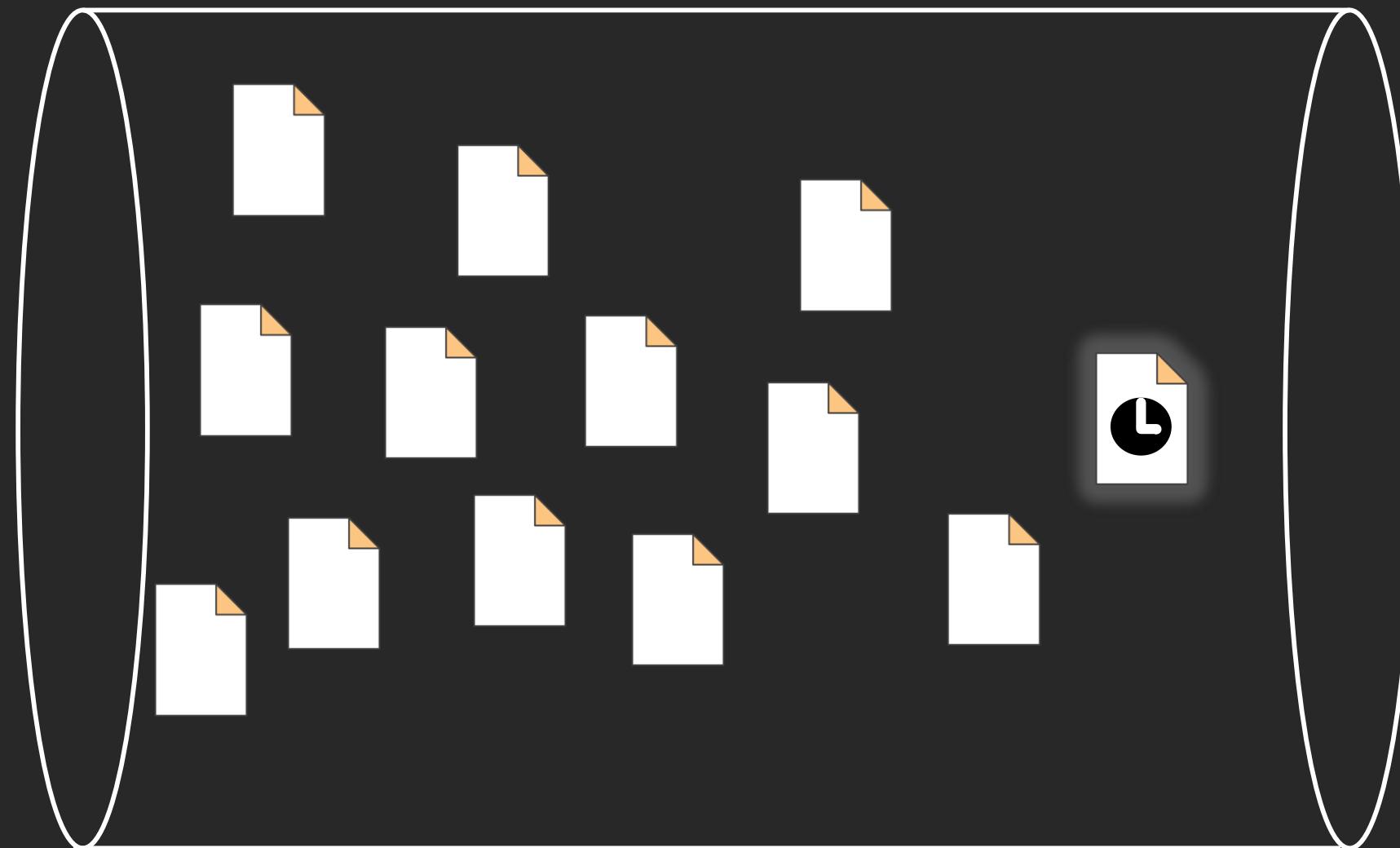
C

C

C



# Consuming from Amazon SQS

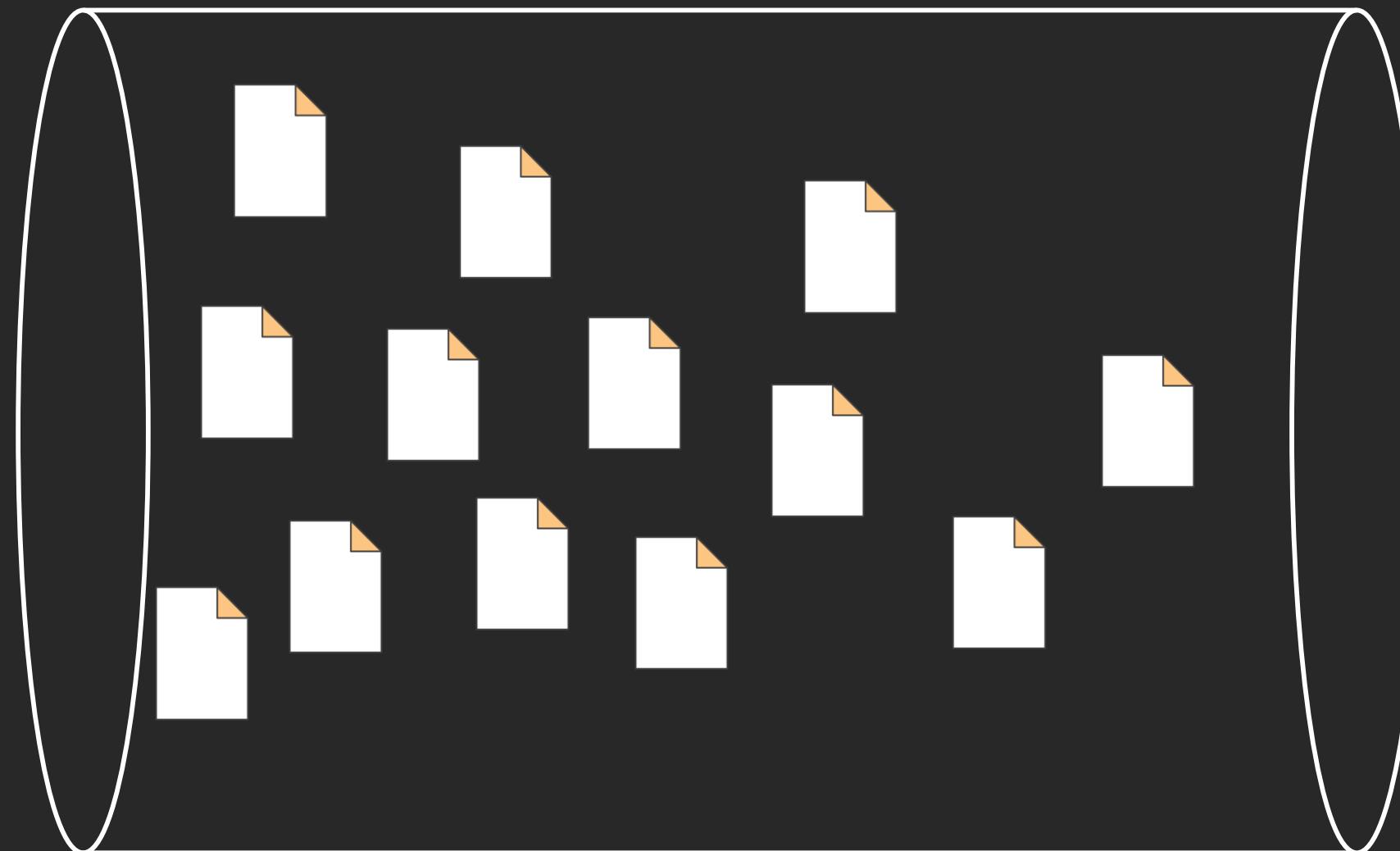


C

C

C

# Consuming from Amazon SQS

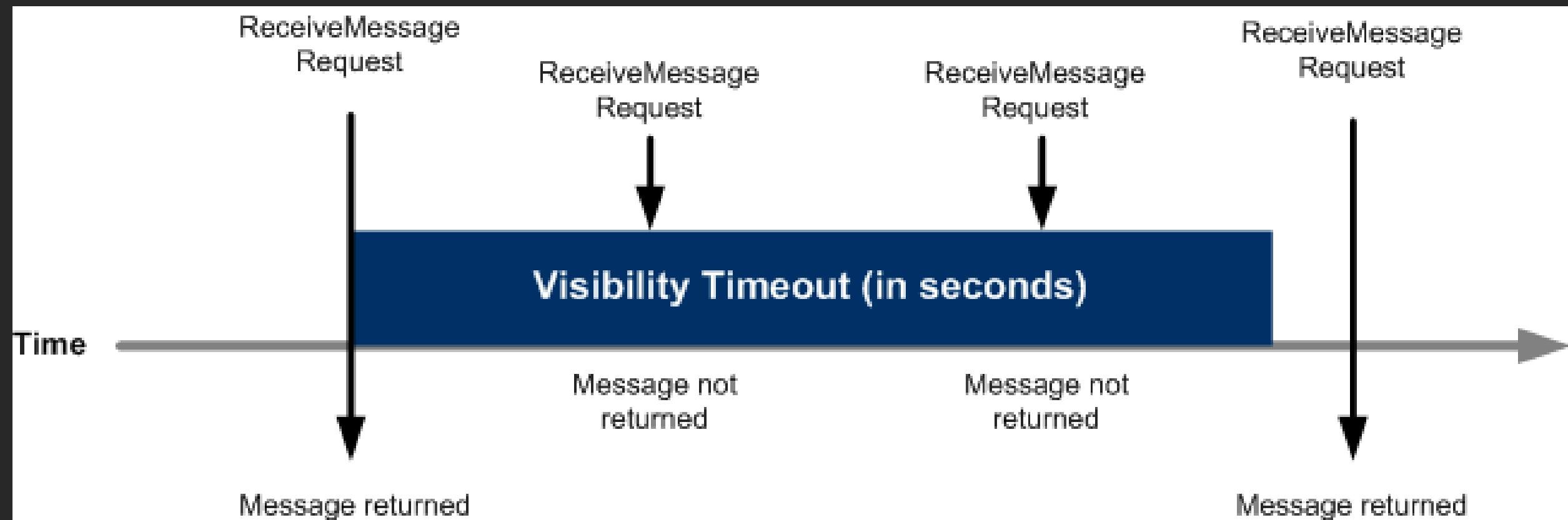


C

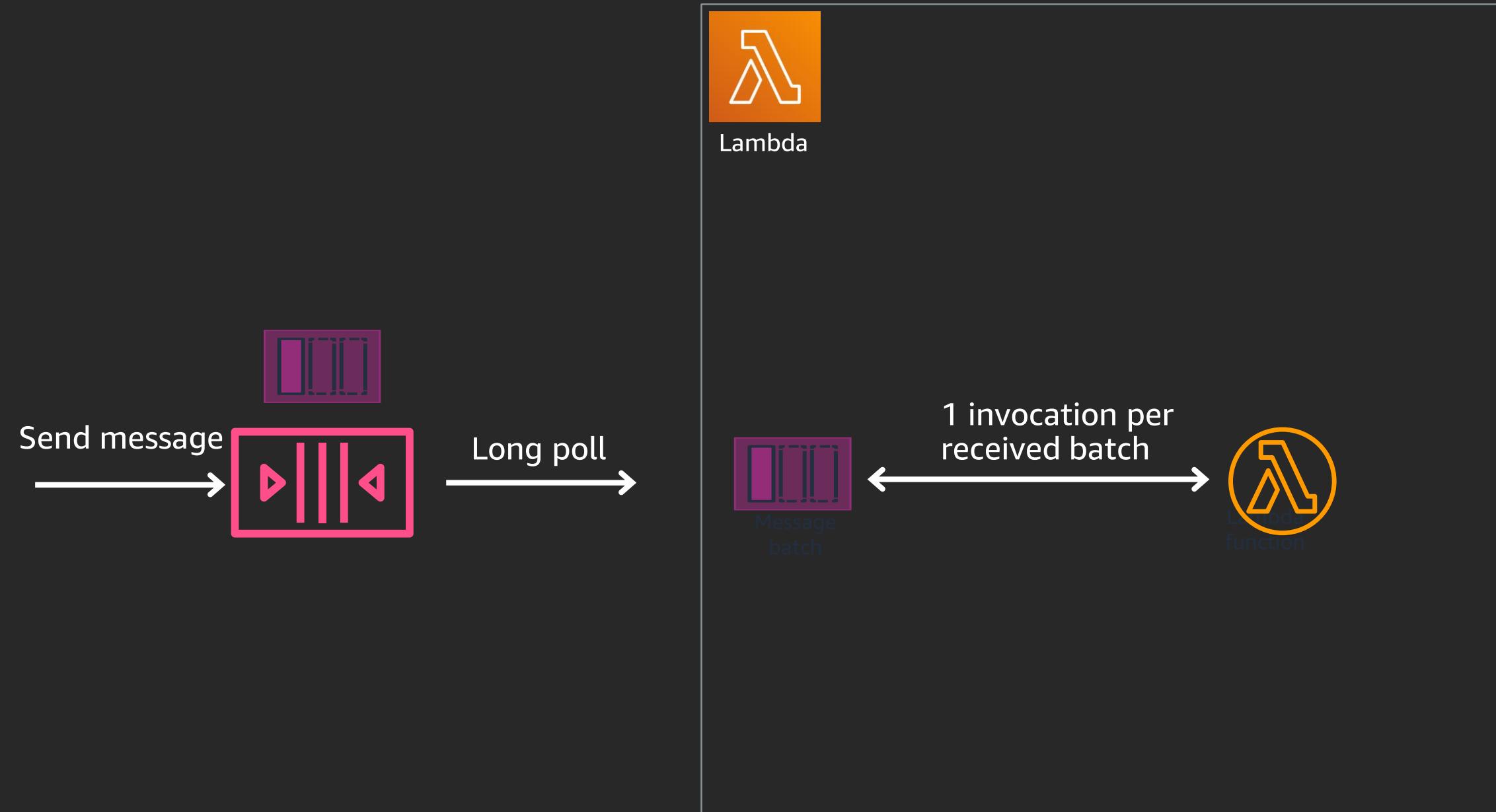
C

C

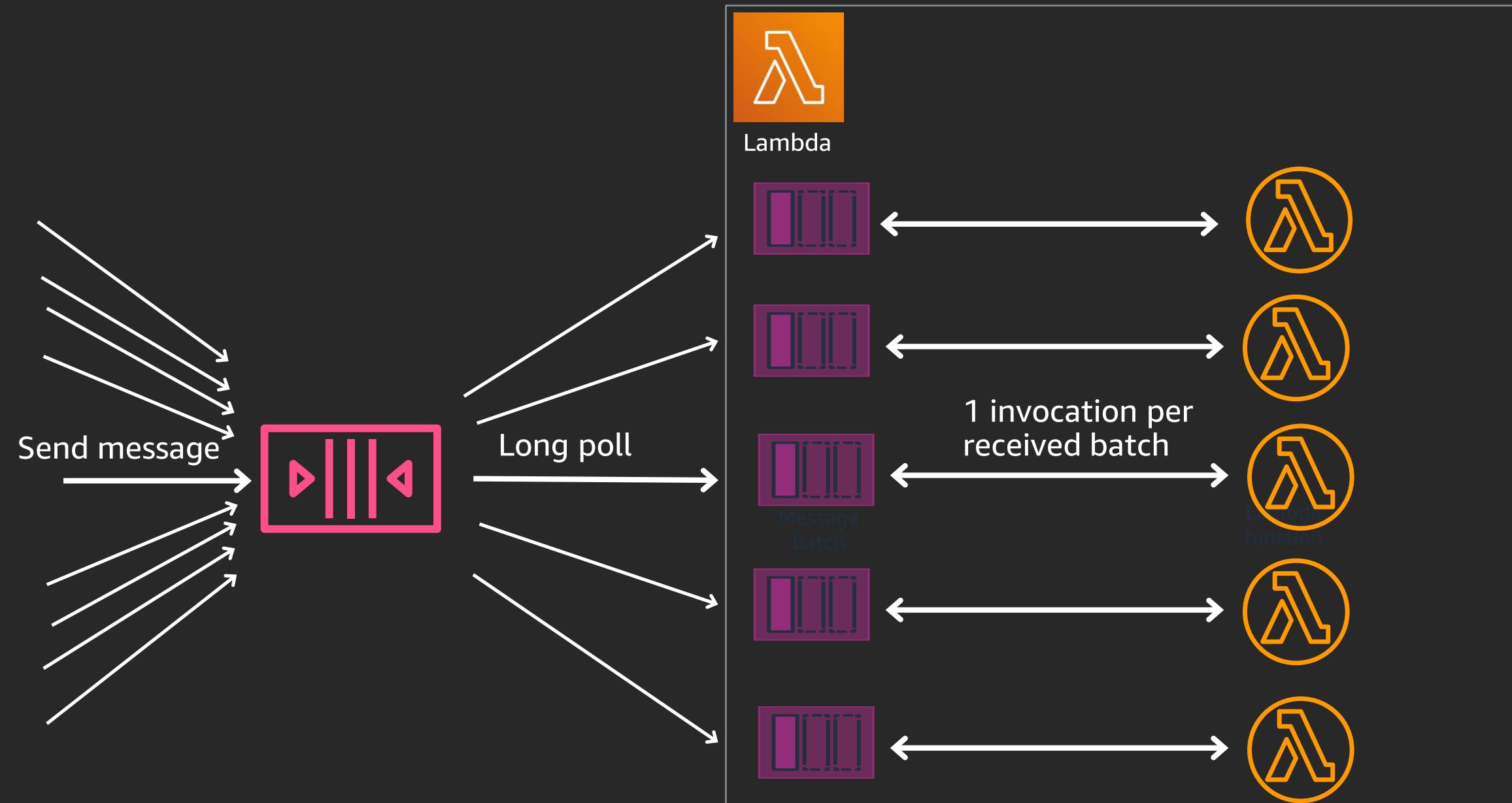
# Inside Amazon SQS: Visibility timeouts



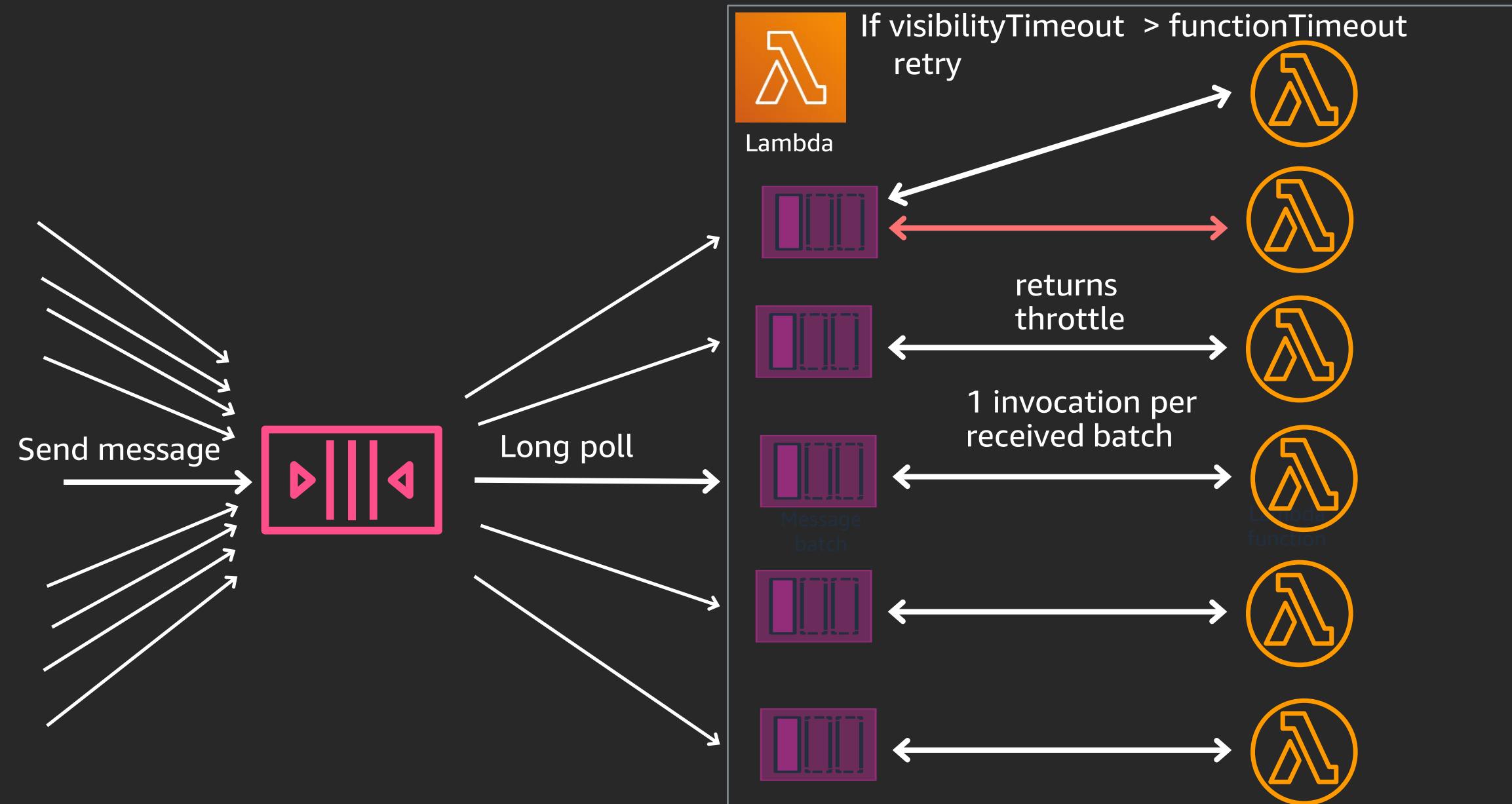
# Amazon SQS to Lambda: Details



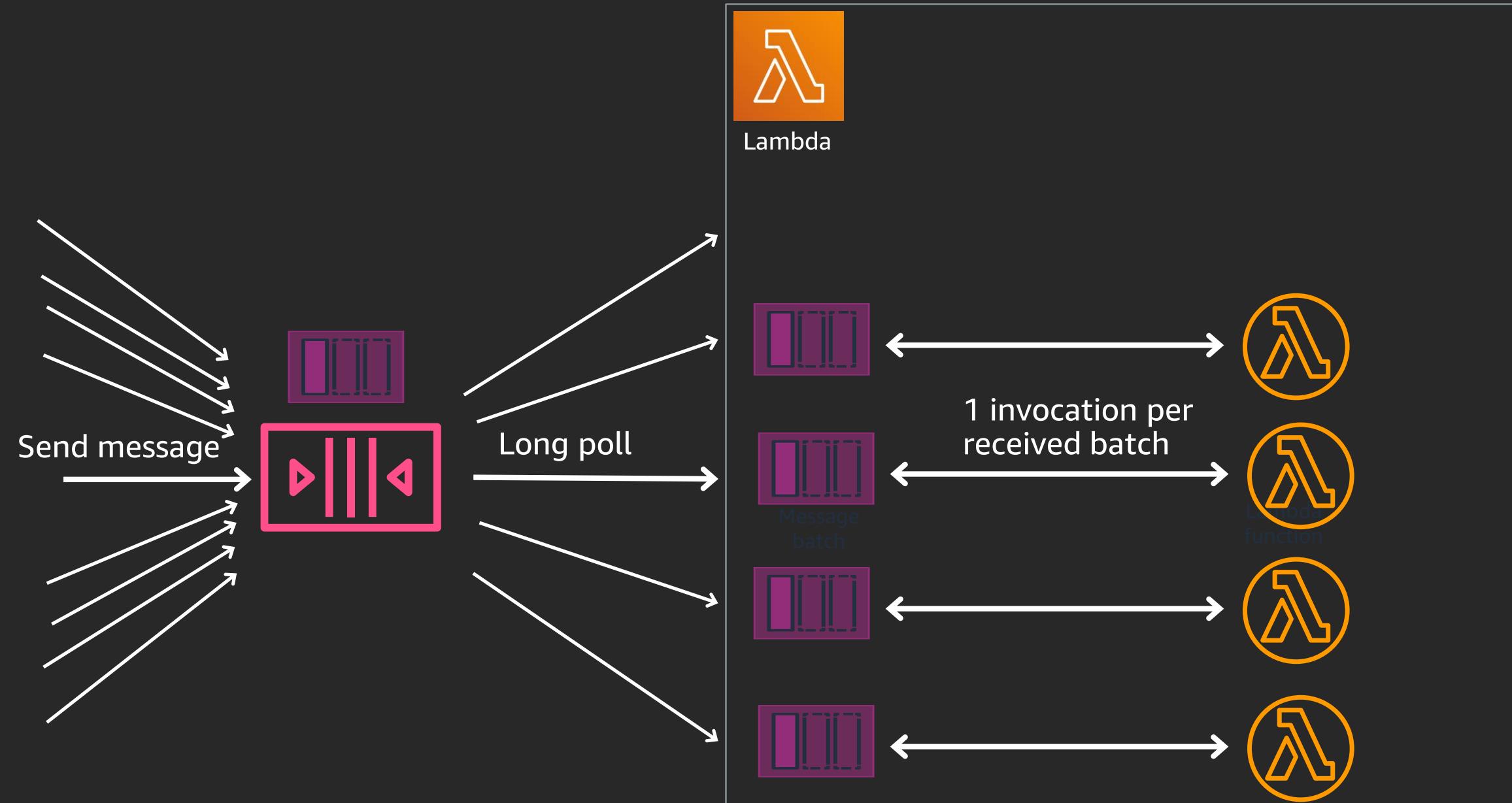
# Amazon SQS to Lambda: Details



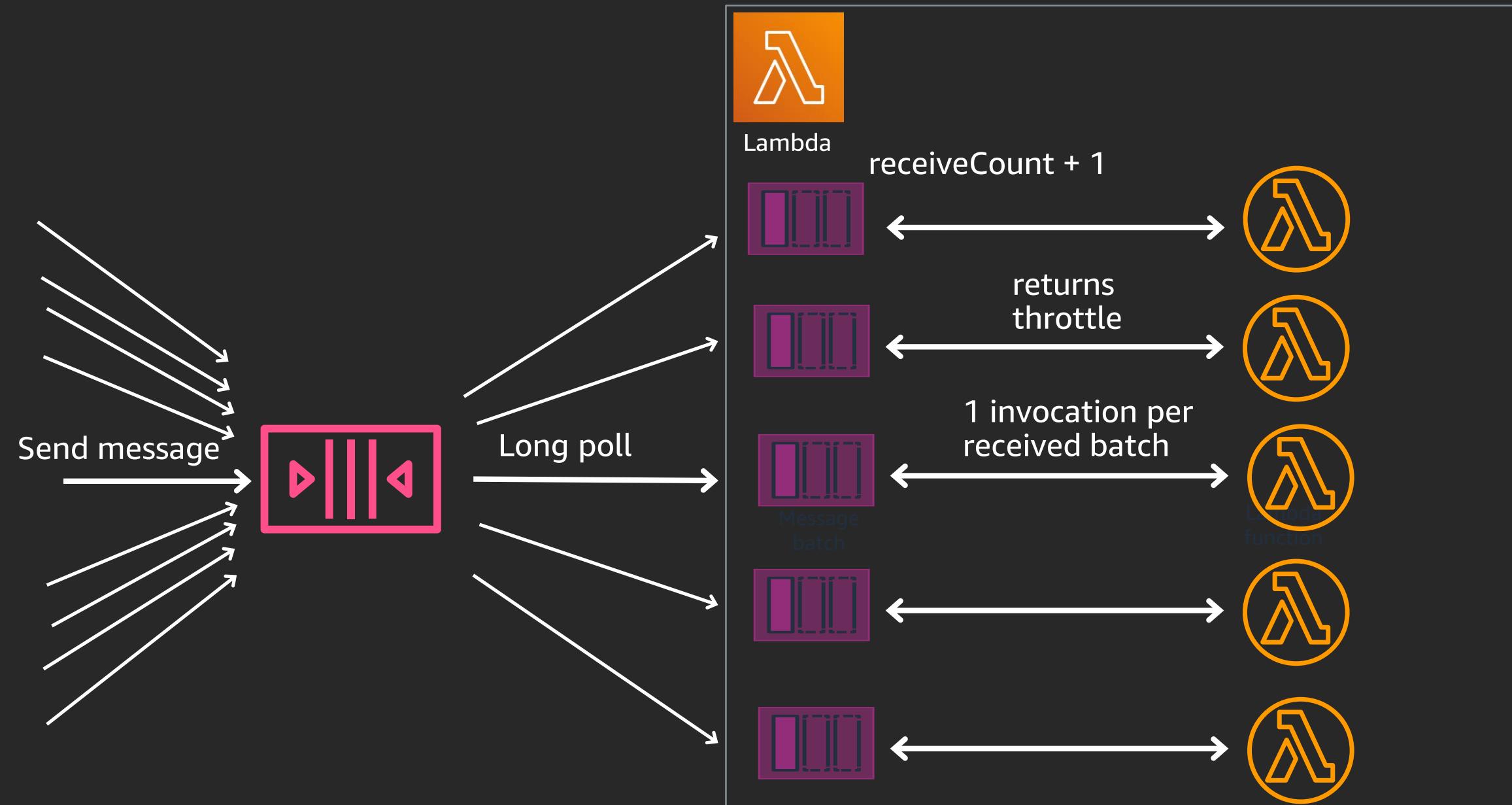
# Amazon SQS to Lambda: Details



# Amazon SQS to Lambda: Details



# Amazon SQS to Lambda: Details



# Demo

SQS Management Console    Lambda Management Console    Using AWS Lambda with Amaz...    +

us-east-2.console.aws.amazon.com/sqs/home?region=us-east-2#queue-browser:prefix=

Apps Bookmarks Alpine-https://cw... SQS/OpsPortal -... WeeklyOps (SQS...) SQS/Dashboards... Automation and M... SQS/InternalDocs... IAM Other Bookmarks

Services Resource Groups

Luay Kawasme Ohio Support

# Simple Queue Service

Amazon Simple Queue Service (SQS) is a reliable, scalable, fully-managed message queuing service.

[Get Started Now](#)

Learn more about Amazon SQS

## Ensure high availability

Amazon SQS uses a distributed architecture within Amazon's high-availability data centers, so queues will be available whenever applications need them. To prevent messages from being lost, all messages are stored redundantly across multiple

## Scale with your business

Amazon SQS enables an unlimited number of services to read and write an unlimited number of messages at any time. Amazon SQS is used by some of the most highly-scaled applications in the world, such as [Netflix](#).

## Reduce your cost

Amazon SQS is a fully-managed service, with no up-front costs or fixed expenses. Pay only for what you use, with a small charge for each API request and data transfer.

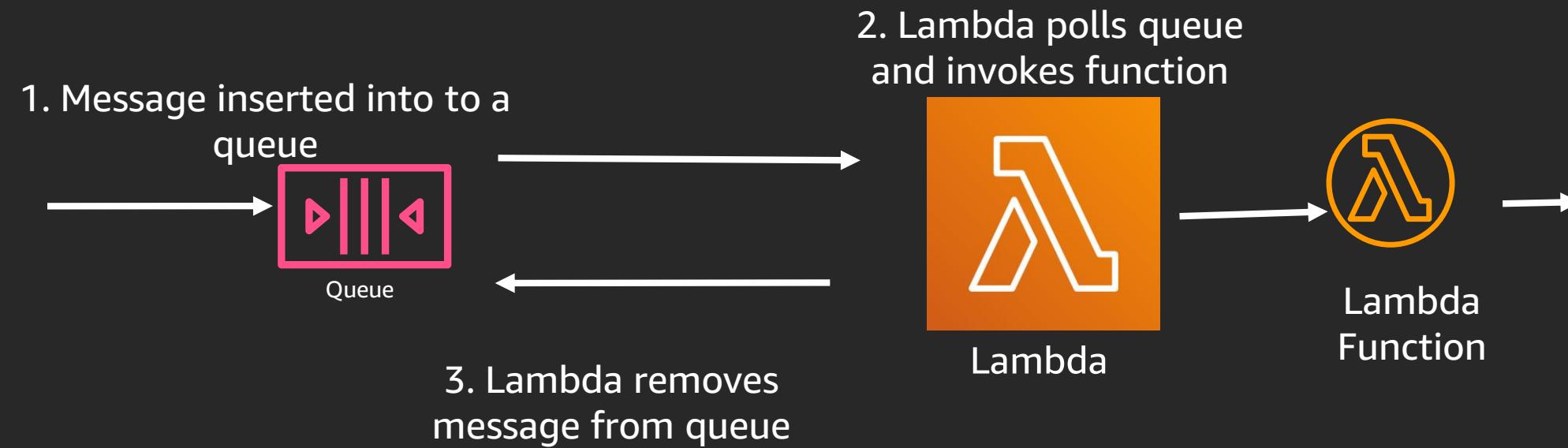
  

Feedback English (US)

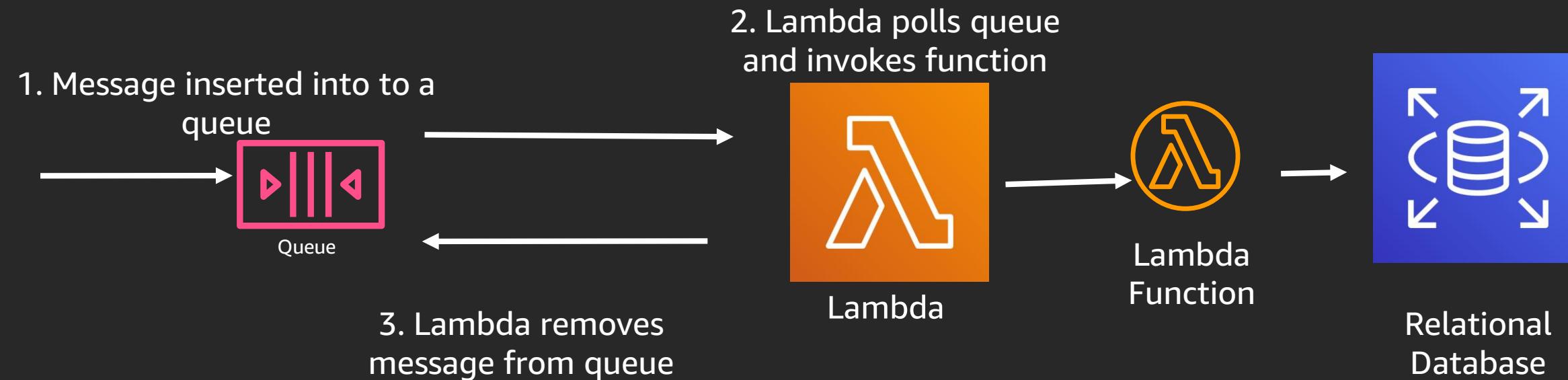
© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# Limiting concurrency

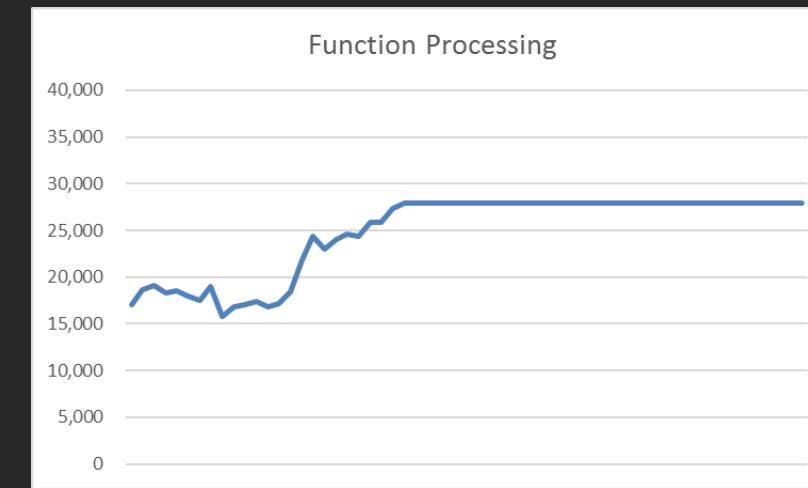
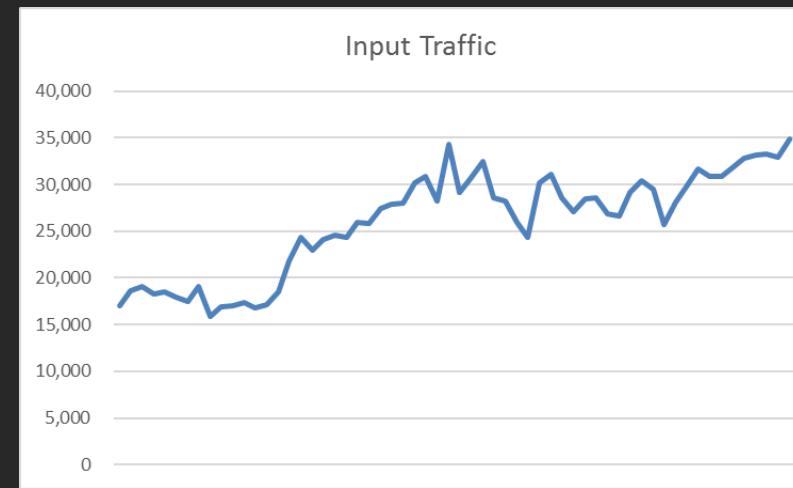
# Highly scalable processing



# Downstream dependency with a limit processing



# Limiting concurrency



# Demo

Create New Queue

Queue Actions

Refresh

Settings

Filter by Prefix:  Enter Text...

X

1 to 1 of 1 items

Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
------	------------	-----------------------------	--------------------	--------------------	---------

reinvent-2019-queue-demo	Standard	N/A	71	540	2019-11-30 21:37:15 GMT-08:00
--------------------------	----------	-----	----	-----	-------------------------------

1 SQS Queue selected

Details

Permissions

Redrive Policy

Monitoring

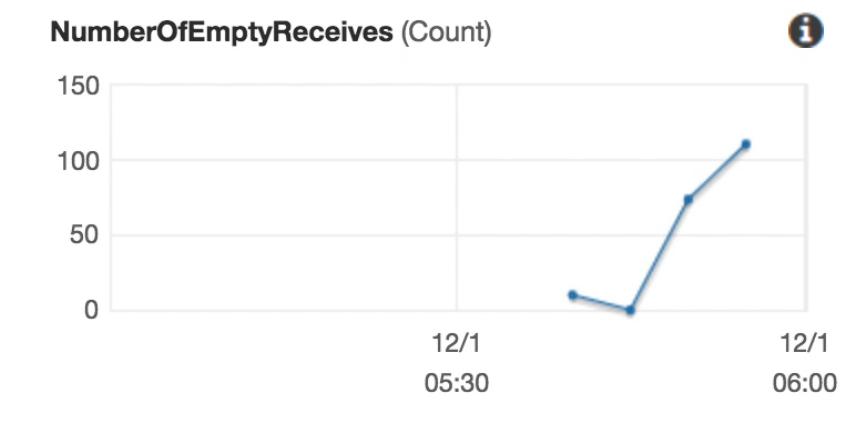
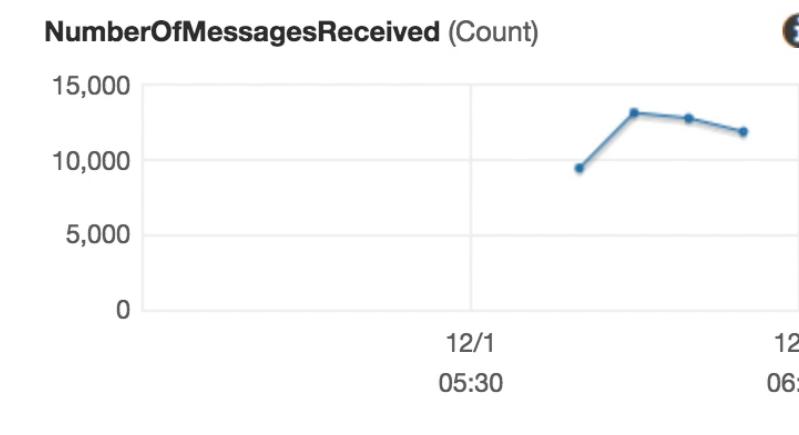
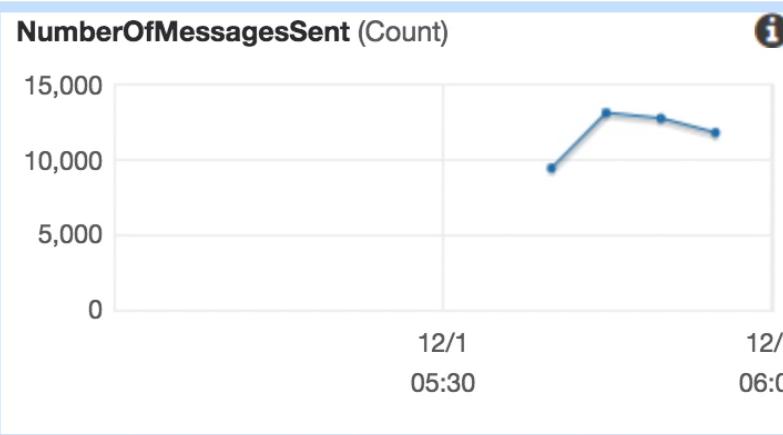
Tags

Encryption

Lambda Triggers

Time Range: Last Hour

Refresh

Below are your CloudWatch metrics for the selected resources (maximum of 10). Click on a graph to see an expanded view. All times shown are in UTC. [View all CloudWatch metrics](#)Legend: █ reinvent-2019-queue-demo**NumberOfMessagesDeleted (Count)**

i

**ApproximateNumberOfMessagesVisible (Count)**

i

**ApproximateNumberOfMessagesNotVisible (Count)**

i

Create New Queue

Queue Actions

Filter by Prefix:  Enter Text...

◀ ▶ 1 to 1 of 1 items ▶ ▶

<input type="checkbox"/>	Name	Queue Type	Content-Based Deduplication	Messages Available	Messages in Flight	Created
<input type="checkbox"/>	reinvent-2019-queue-demo	Standard	N/A	3,493	90	2019-11-30 21:37:15 GMT-08:00

1 SQS Queue selected



Details

Permissions

Redrive Policy

Monitoring

Tags

Encryption

Lambda Triggers

SQS metrics Time Range: Last HourBelow are your CloudWatch metrics for the selected resources (maximum of 10). Click on a graph to see an expanded view. All times shown are in UTC. [View all CloudWatch metrics](#)Legend:  reinvent-2019-queue-demo

NumberOfMessagesSent (Count)



i

NumberOfMessagesReceived (Count)



i

NumberOfEmptyReceives (Count)



i

NumberOfMessagesDeleted (Count)

i

ApproximateNumberOfMessagesVisible (Count)

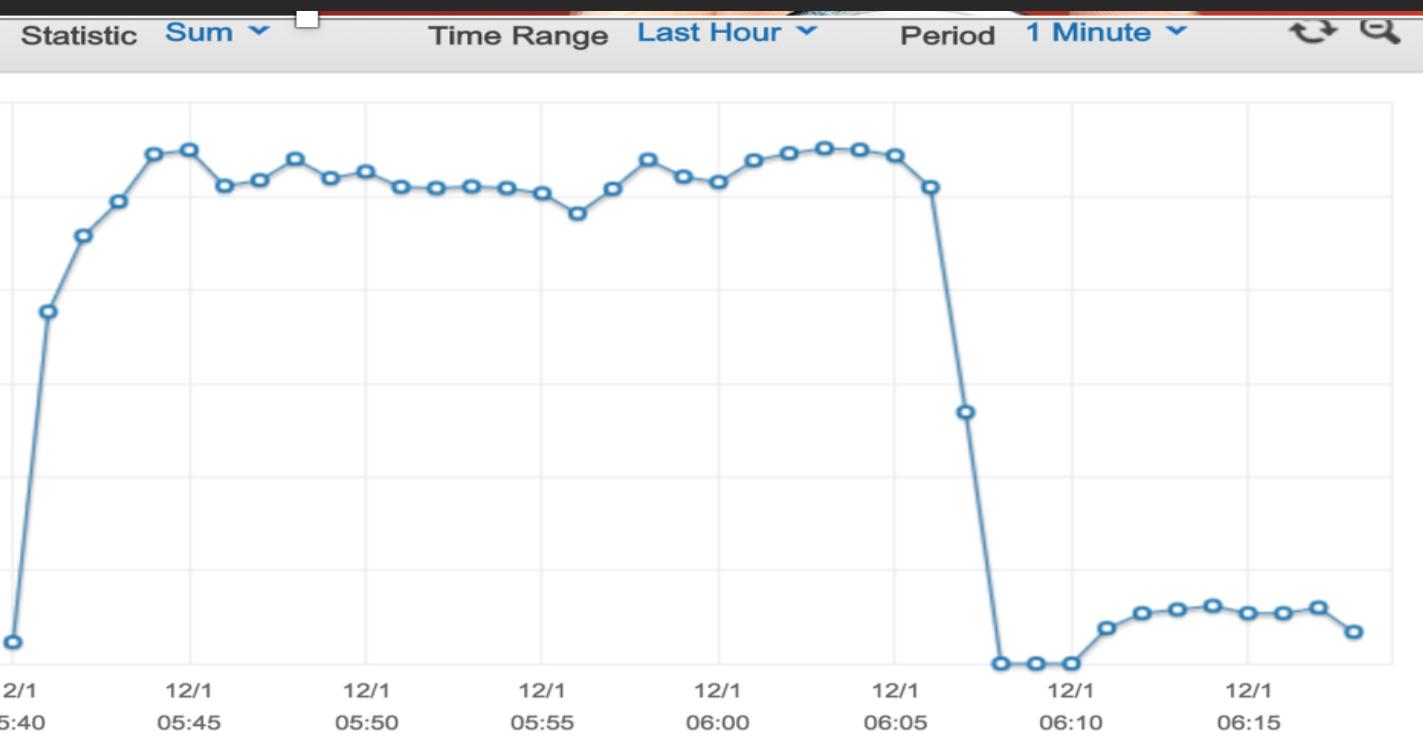
i

ApproximateNumberOfMessagesNotVisible (Count)

i

# One minute Amazon SQS metrics

- Coming soon to all regions
- Now available in Ohio, Ireland, Stockholm, and Tokyo regions



# Tips

# Amazon SQS FIFO triggers to Lambda



# Amazon SNS or Amazon SQS as an event source?

- Amazon SNS for Lambda is simple to use, with lower latency. Use SNS for Lambda if your downstream dependencies can keep up with incoming message rates.
- Amazon SQS for Lambda has more control and is better for decoupling systems and rate matching.

Comparison	SNS as an event source for Lambda	SQS as an event source for Lambda
Scale/ concurrency	Automatic scaling, per function concurrency	Same
Durability	Durable storage across multiple AZs	Same
Consumption model	Messages pushed to Lambda; Amazon SNS can “fanout” the same message	Lambda polls for messages and invokes function; each message is consumed and deleted by Lambda
Resilience	Once delivered, message is deleted, even if Lambda function does not complete processing	Lambda function is responsible for deleting message after processing; ability to monitor and control queue
Persistence/ retries	Delivery retry logic that extends up through potentially 13 hours	Message retention for up to 14 days

# Amazon SQS to Lambda: Tips

- For low-concurrency, high-duration functions, set visibility timeout to 5x function duration. This will avoid `receiveCount` retries and DLQ.
- Check that visibility timeout is always greater than function duration to avoid duplicate invokes
- Even a fraction of errors will induce backpressure and interfere with ramping up to full concurrency
- The number of invocations does not equal throughput. Maximum batching is not guaranteed.

# Thank you!

**Luay Kawasme**

@luay\_ca

**Rory Richardson**



Please complete the session  
survey in the mobile app.

# Lambda invocation integrations

Amazon S3

DynamoDB

Amazon Kinesis Data Streams

Amazon SNS

Amazon SES

Amazon SQS

Amazon Cognito

AWS CloudFormation

Amazon CloudWatch Logs

Amazon CloudWatch Events

AWS CodeCommit

AWS Config

Amazon Alexa

Amazon Lex

Amazon API Gateway

AWS IoT Button

and more . . .