

# TECHNOLOGY



## Designing Applications and Architectures in AWS

(Aligned with AWS Solution Architect Associate Certification)

## Serverless and Application Services





# A Day in the Life of a Cloud Architect

John is working as a cloud architect in an organization and has been asked to launch the lambda function and perform the trigger operation on the lambda.

Highly available architecture on AWS has business-transforming capabilities. As a solution architect, he must replicate the contents of a bucket to other buckets deployed in different regions, thereby making the resources highly available.

To achieve these, he will be learning a few concepts in this lesson that will help him find a solution for the given scenario.



# Learning Objectives

By the end of this lesson, you will be able to:

- Describe the concept of lambda serverless computing to make changes to data in S3 buckets
- Configure event notification for AWS resources using SNS to provide real-time event handling
- Analyze EventBridge, Event Bus, and SaaS Partner Bus to integrate various AWS services
- Differentiate between SNS and SQS
- Implement real-time processing of streaming big data to gain immediate insights from data as it arrives



Lambda

# AWS Serverless Services

AWS serverless services empower users to develop and operate applications without the need to concern themselves with server provisioning, maintenance, or management.



# Usage of AWS Serverless Services

AWS serverless services remove the need to manage the following infrastructure tasks:

Provisioning servers or clusters

Maintaining the operating system

Patching software and hardware

Allocating compute capacity



# Benefits of AWS Serverless Services

AWS serverless services offer the following benefits:



- No need for server management
- Dynamic scalability
- Payment based on usage
- Enhanced availability





# What Is AWS Lambda?

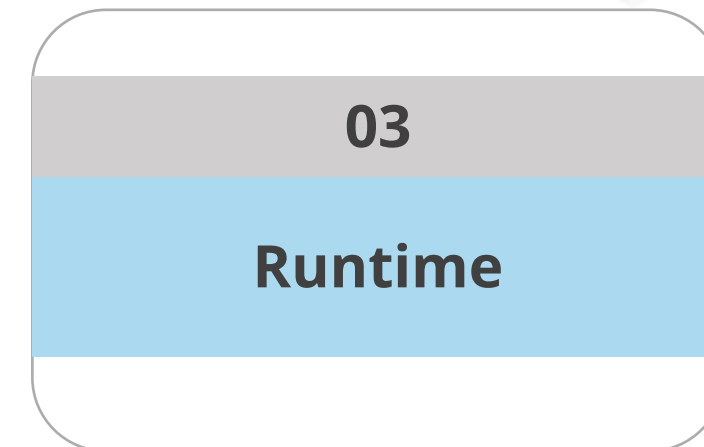
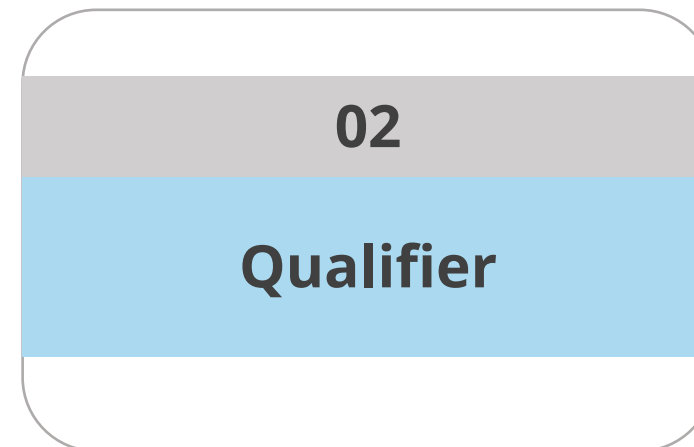
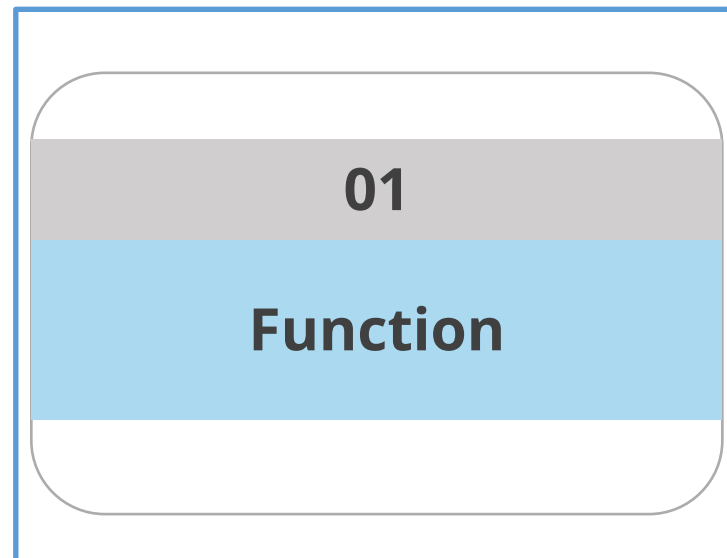
It is a serverless computing service that enables users to execute code without provisioning or managing servers.



It executes code upon demand, automatically scaling from a few requests per day to handling thousands of requests per second.

# Terminologies in AWS Lambda

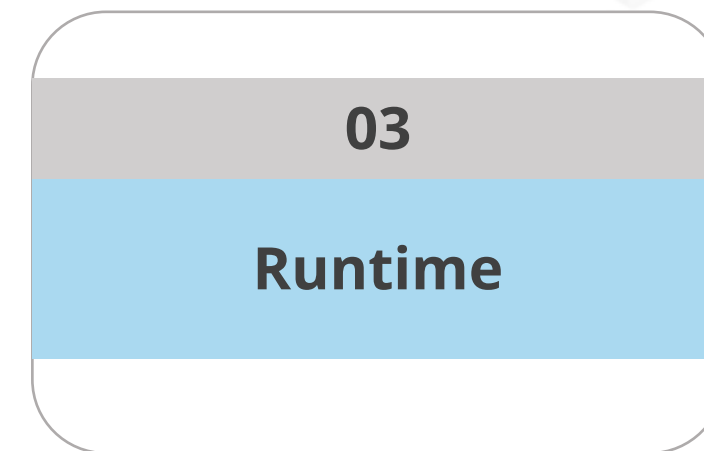
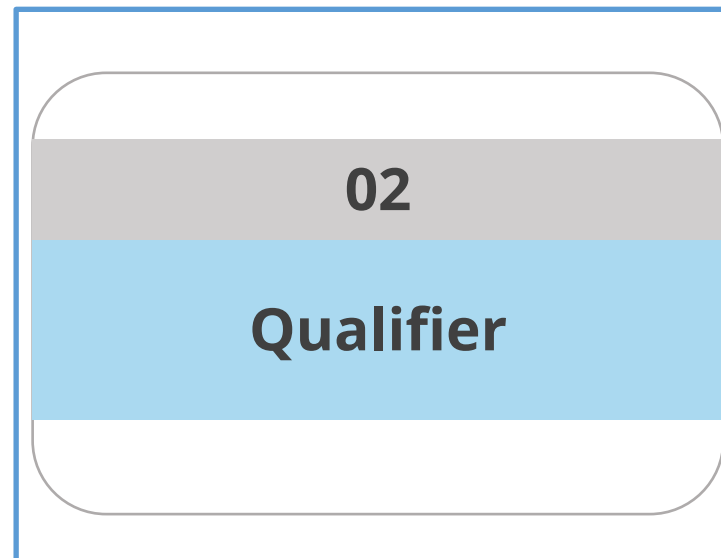
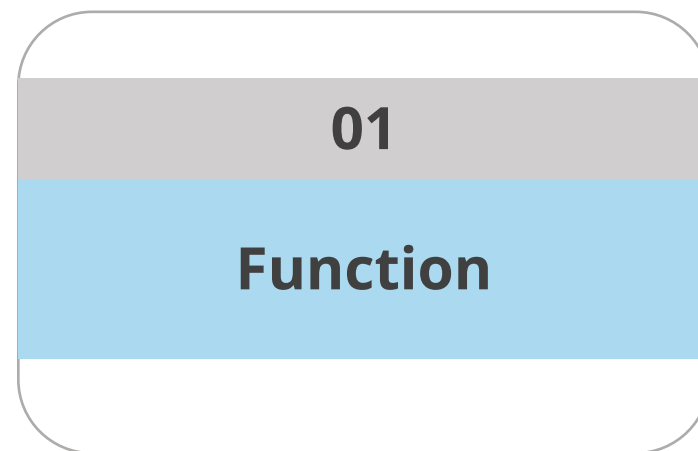
The following terms are essential in the context of AWS Lambda:



A function is a resource that holds code for event processing and runtime for forwarding requests between Lambda and the function code.

# Terminologies in AWS Lambda

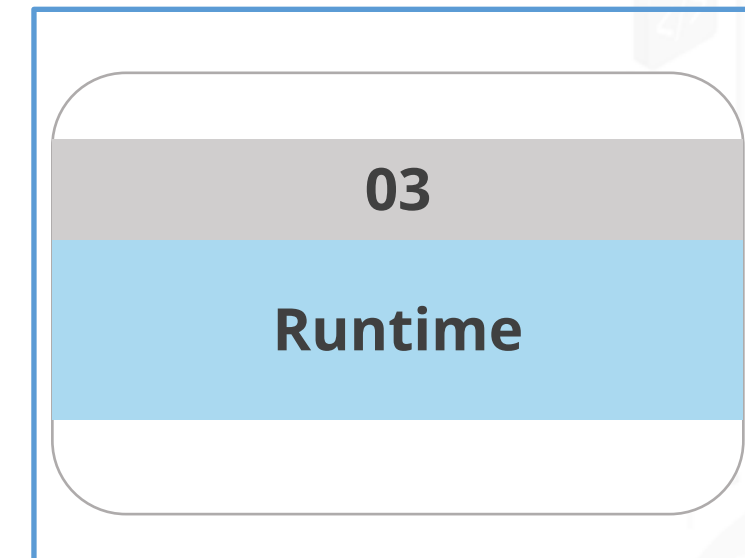
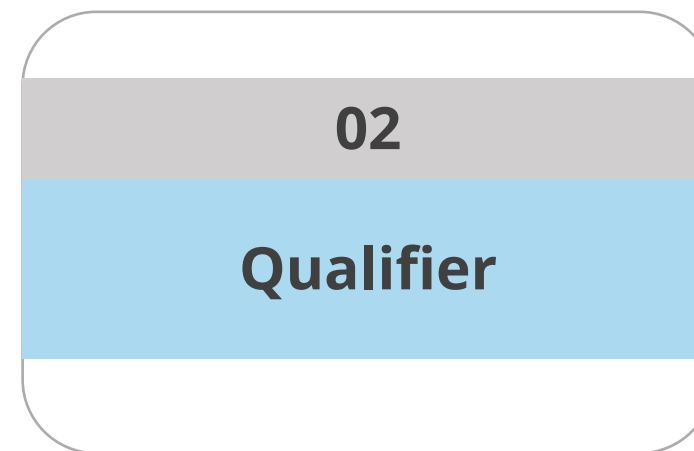
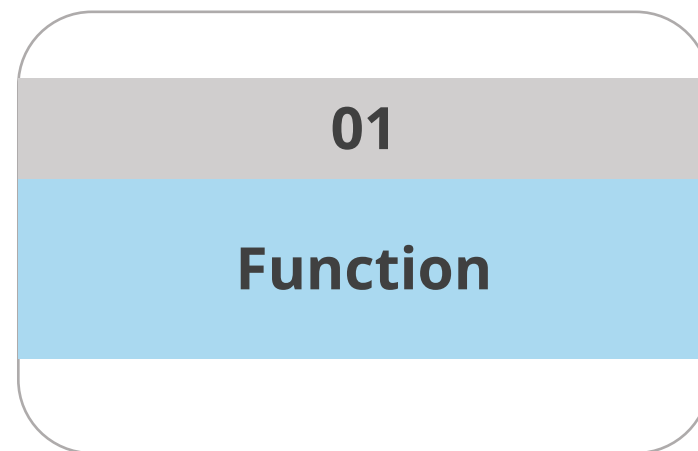
The following terms are essential in the context of AWS Lambda:



The qualifier is employed to indicate a version or an alias for a Lambda function.

# Terminologies in AWS Lambda

The following terms are essential in the context of AWS Lambda:

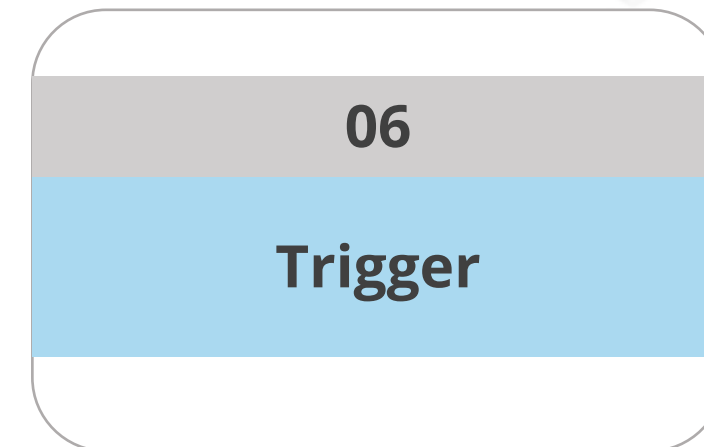
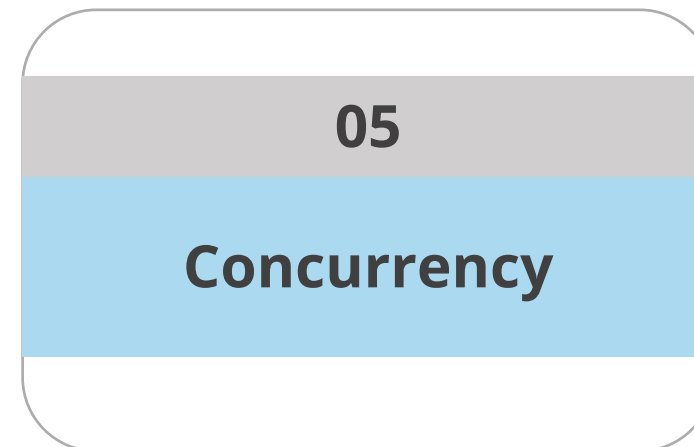
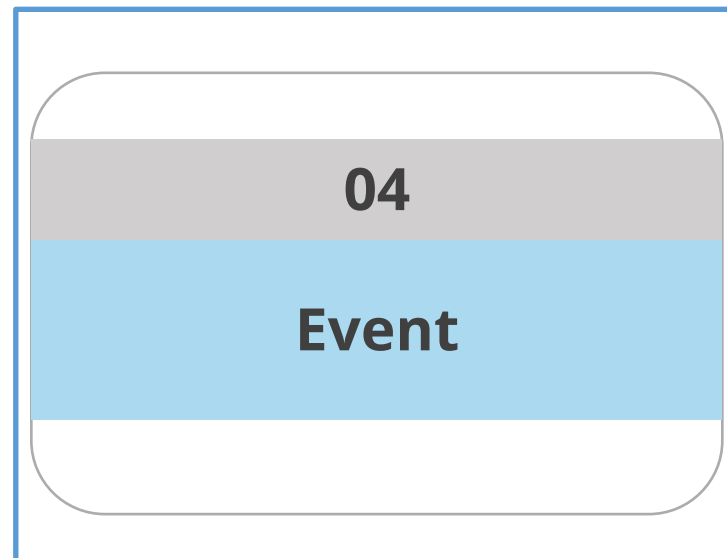


Runtimes enable diverse code languages to run in the same environment. Users select a runtime that matches their code's language.



# Terminologies in AWS Lambda

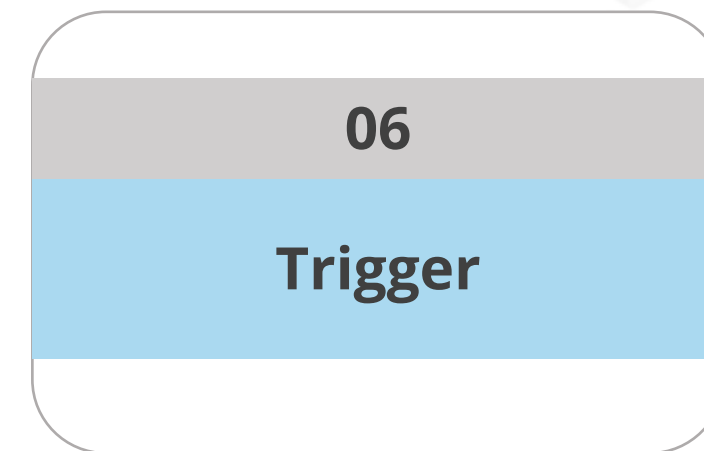
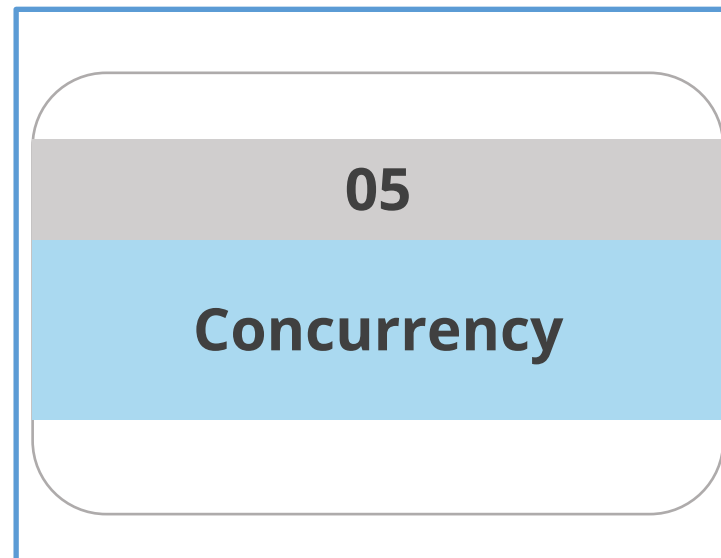
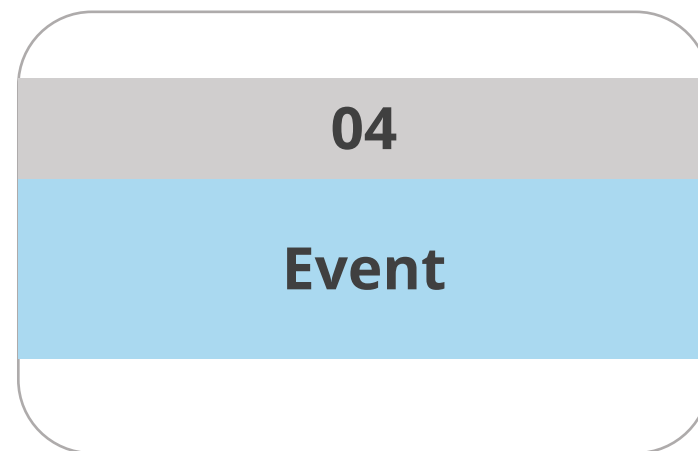
The following terms are essential in the context of AWS Lambda:



An event constitutes a JSON-formatted document with data for function processing. This document is transformed into an object and then conveyed to the function code.

# Terminologies in AWS Lambda

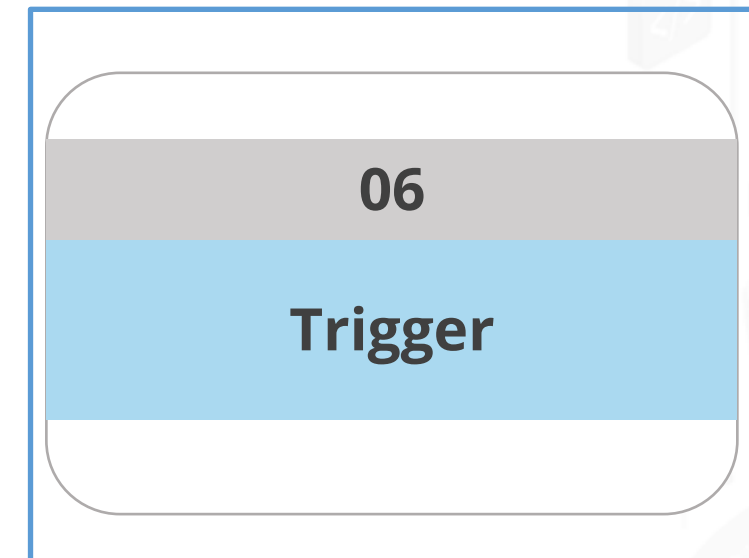
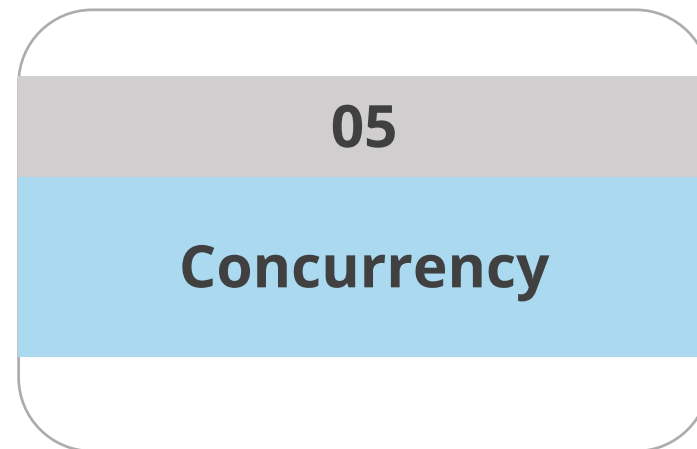
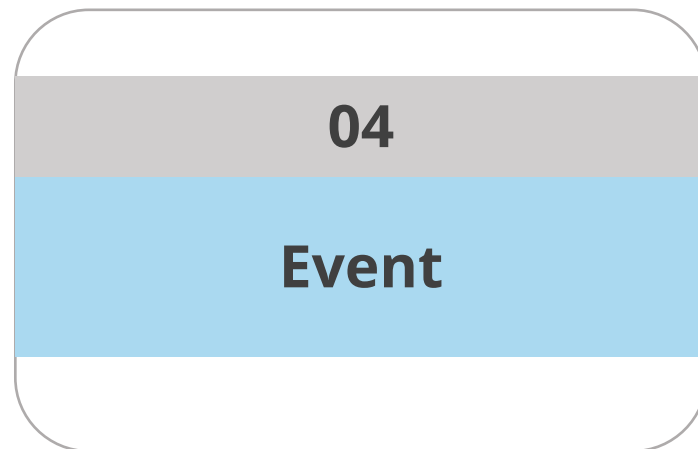
The following terms are essential in the context of AWS Lambda:



Concurrency denotes the number of ongoing requests a function handles. Users have the option to set concurrency limits for their functions.

# Terminologies in AWS Lambda

The following terms are essential in the context of AWS Lambda:



A trigger serves as a mechanism to activate a Lambda function. It can take the form of an AWS service, an application, or an event.

# Amazon RDS Proxy

Amazon RDS Proxy is a fully managed and highly available database proxy designed for AWS RDS.

Amazon RDS Proxy makes applications more:

01

Scalable

02

Resilient to database failures

03

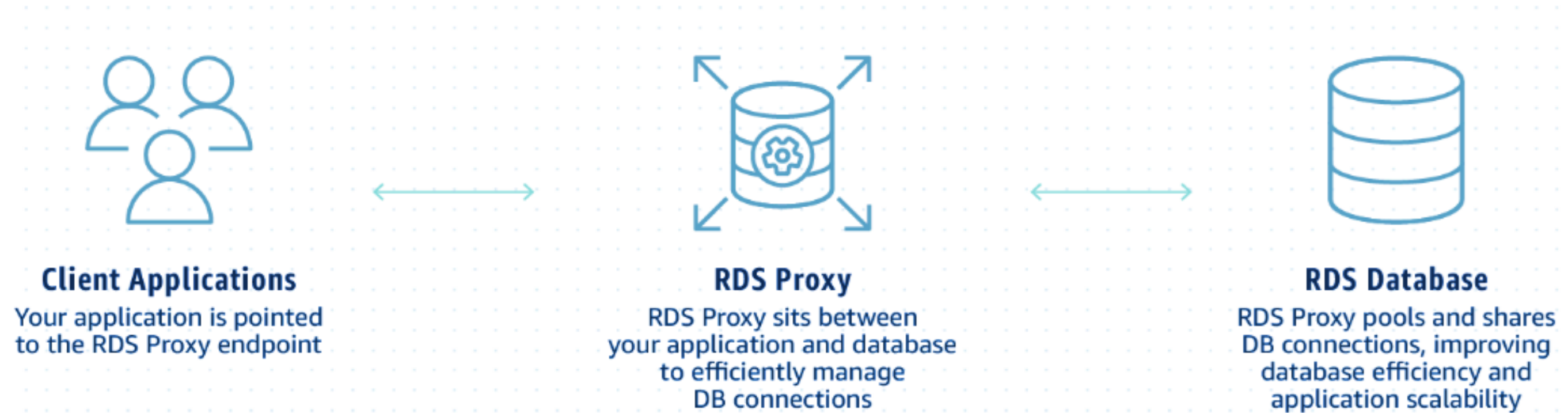
Secure





# Amazon RDS Proxy

It serves as a pivotal element for efficient connection management and enhanced application scalability.



# Networking and VPC Configurations

The Networking and VPC configurations for Lambda functions are as follows:

- Lambda functions consistently operate within VPCs owned by the Lambda service.
- These VPCs are not accessible to customers.
- Configurations are automatically managed, and monitoring is overseen by the service.
- The Lambda service employs a Network Function Virtualization platform to deliver NAT capabilities from the Lambda VPC to customer VPCs.

# Security

Security practices for Lambda functions encompass the following aspects:

- Lambda function execution environments are always isolated, ensuring no sharing across functions.
- The Lambda service employs multiple mechanisms to safeguard customer data.
- The principles of least privilege are applied, encompassing permissions and function scoping, to secure user accounts adequately.
- Workloads are protected with public endpoints that have authentication and authorization protocols in place.
- Robust data encryption measures are implemented within Lambda-based applications.

# Synchronous Invocations

Synchronous invocations are particularly well-suited for Lambda functions with short lifespans.





# Synchronous Invocations

Below are the examples of synchronous invocations:



Amazon API Gateway

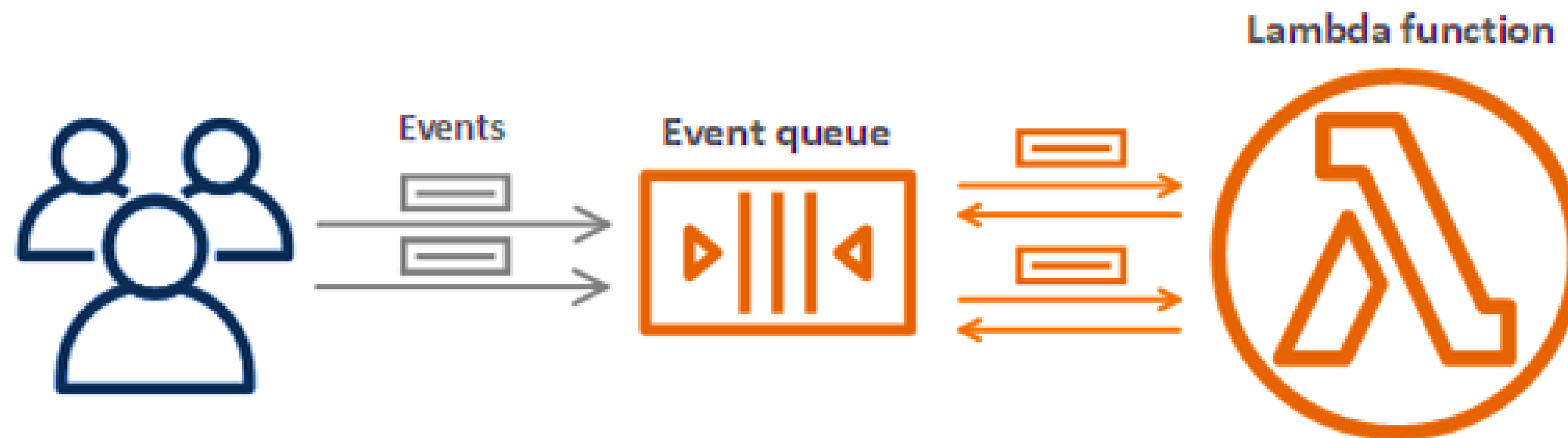


Amazon DynamoDB Streams



# Asynchronous Invocations

These invocations are particularly suitable for Lambda functions with longer lifespans.



# Asynchronous Invocations

Below are the examples of asynchronous invocations:



# Lambda@Edge



Lambda@Edge serves as an extension or potential replacement for the origin infrastructure.

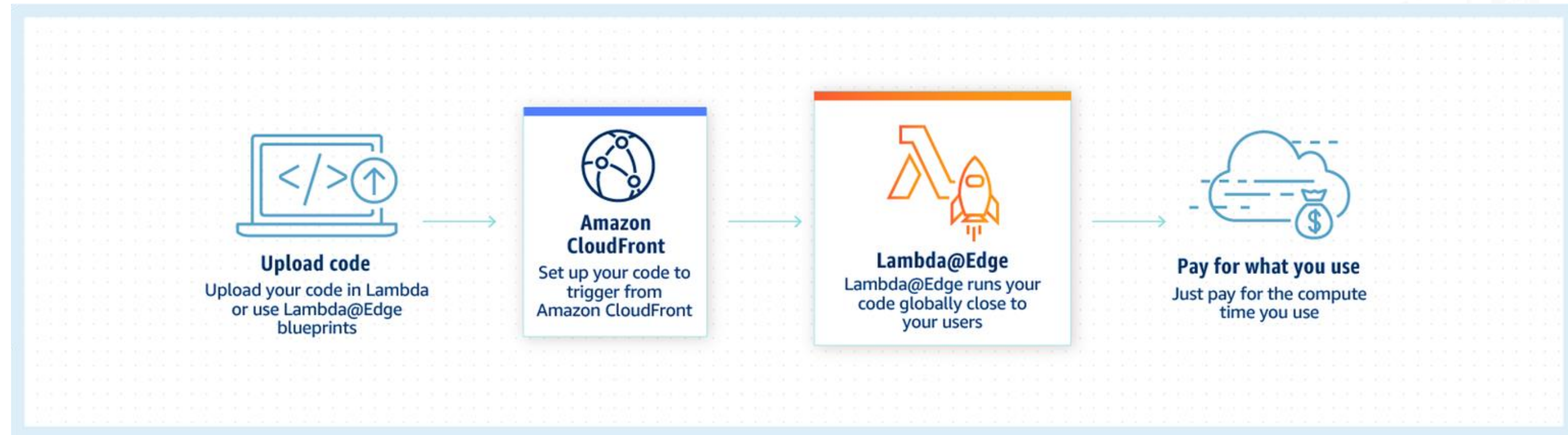


It empowers users to perform tasks ranging from basic HTTP requests and response processing at the edge to more advanced functionalities.



# Lambda@Edge

Lambda@Edge simplifies and streamlines origin infrastructure.



# Lambda Function and CloudWatch



**Duration:10 min**

## **Problem Statement:**

You have been assigned a task to create an AWS Lambda function, deploy code to it, test the function, and view its CloudWatch logs.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

## Steps to be followed are:

1. Create a Lambda service
2. Deploy code to Lambda function
3. Test the Lambda function
4. View CloudWatch logs



# Configuring a Lambda Layer



Duration:10 min

## Problem Statement:

You have been assigned a task to create and configure a Lambda function along with a Lambda layer.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

## Steps to be followed are:

1. Create a Lambda function
2. Create a Lambda layer



# Creating a Serverless Web App



**Duration:10 min**

## **Problem Statement:**

You have been assigned a task to create a serverless web application on the Amazon Web Services (AWS) platform.

ASSISTED PRACTICE



# Assisted Practice: Guidelines

---

**Steps to be followed are:**

1. Develop a serverless web application



## API Gateway

# What Is an API?

API, which stands for Application Programming Interface, facilitates communication between two applications. It is designed to enable apps to access data, logic, and more.



Application Programming Interface



# Amazon API Gateway

Amazon API Gateway is a fully managed and scalable API management service. It empowers users to create, publish, maintain, monitor, and secure the APIs effectively.

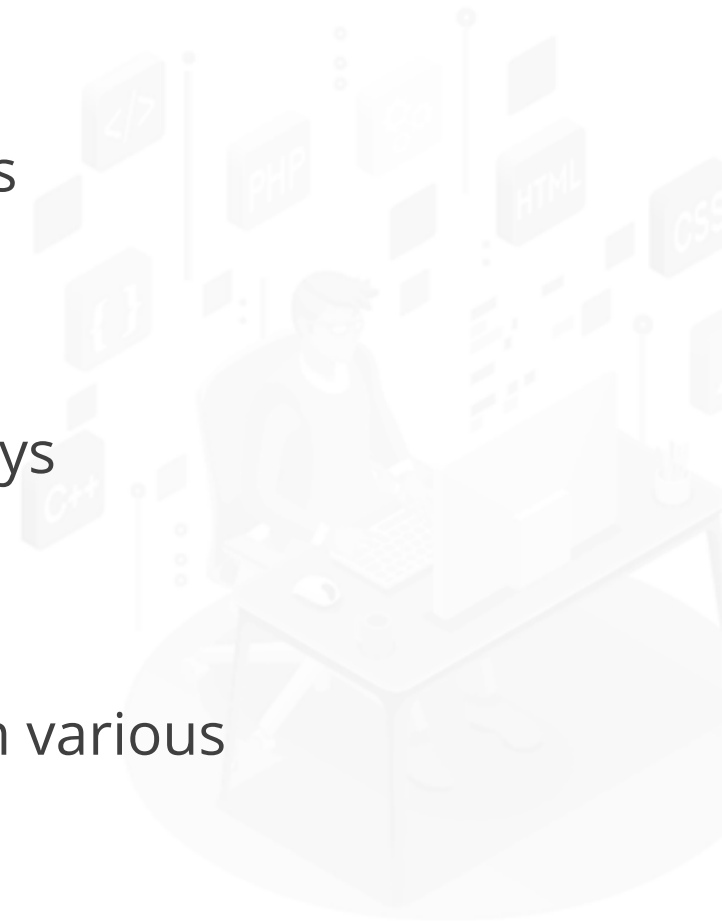


# API Gateway: Features

API Gateway offers several key features, including:



- Storing responses for common HTTP requests
- Automatic scalability
- Cost-effectiveness compared to other gateways
- Request throttling for attack prevention
- CORS support for serving HTTP requests from various domains

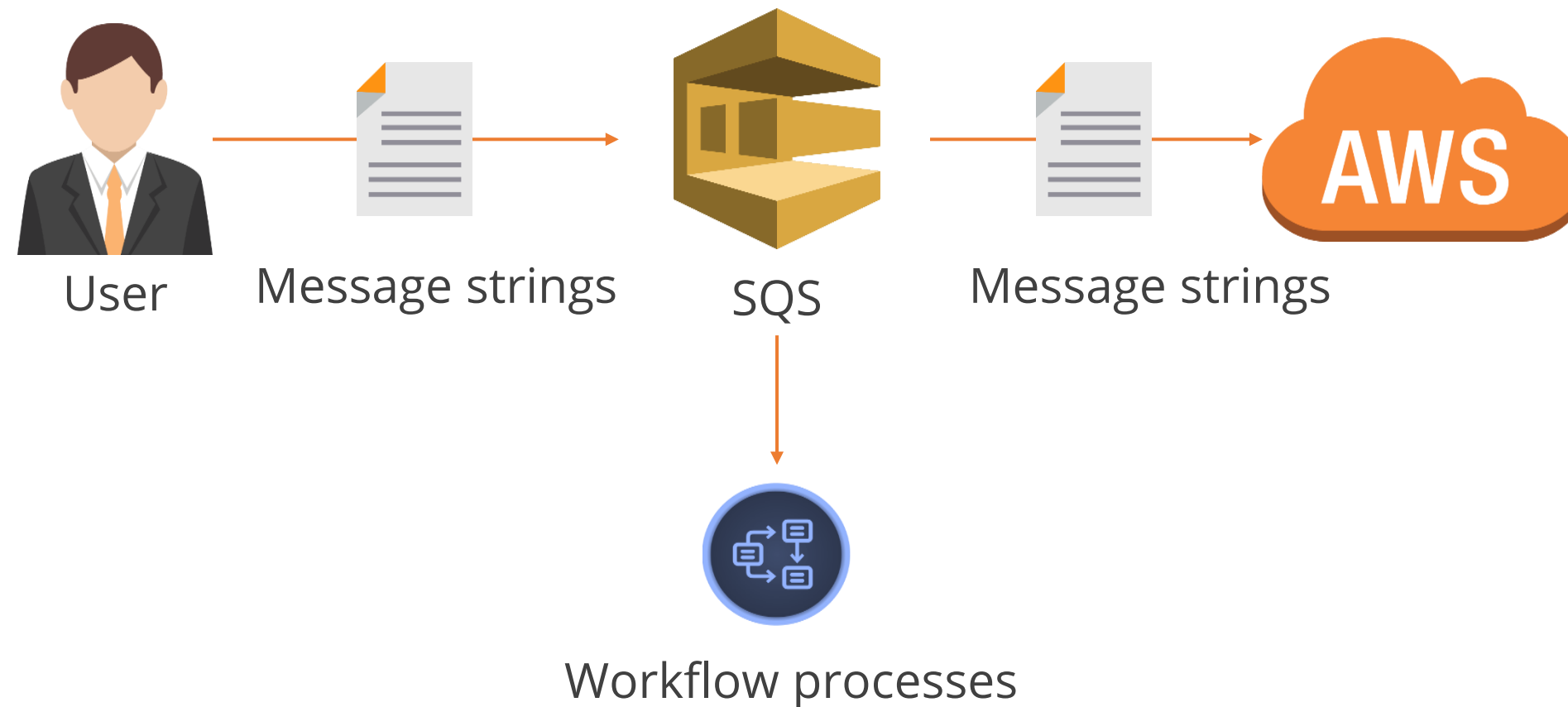


## SQS Introduction



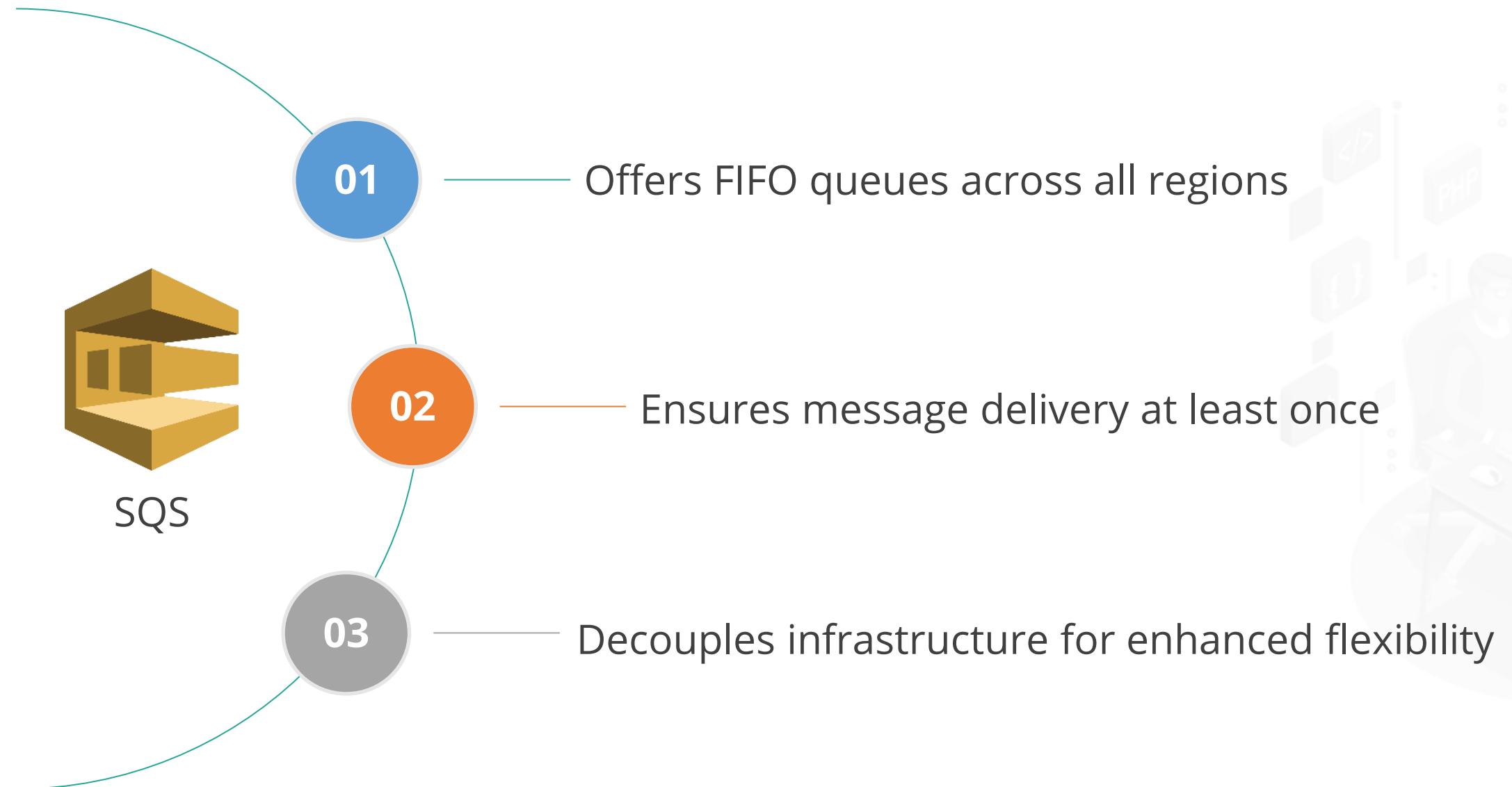
# SQS Standard

Amazon Simple Queue Service (SQS) is a fully managed queue service. It receives, stores, and sends message strings containing job descriptions across application components and AWS services.



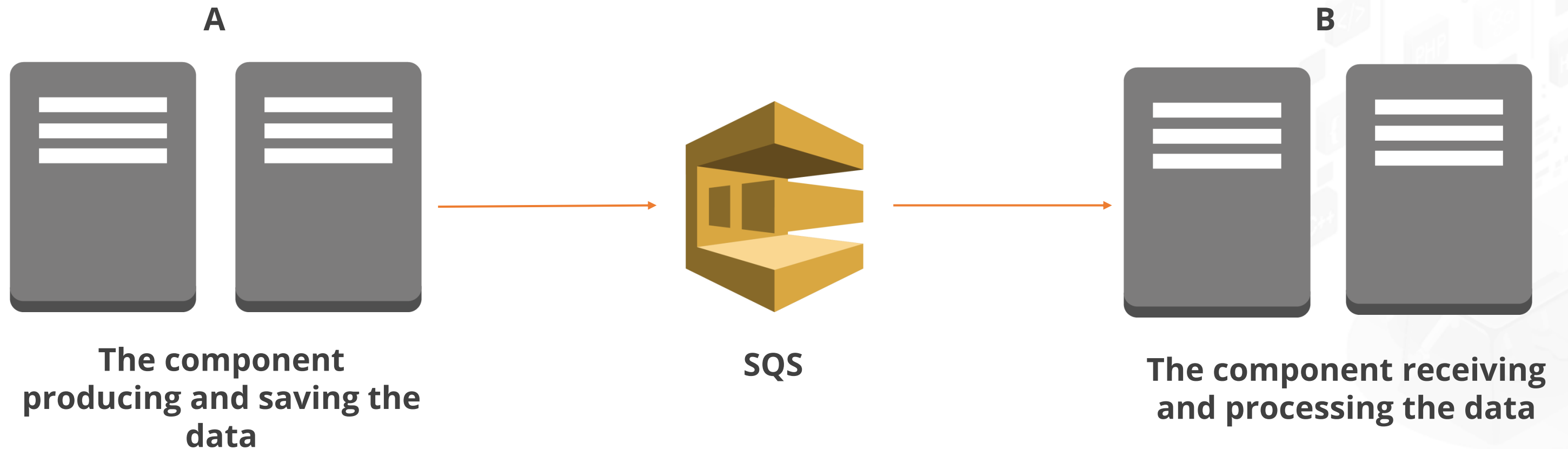
# SQS: Features

The following are the key features of SQS:



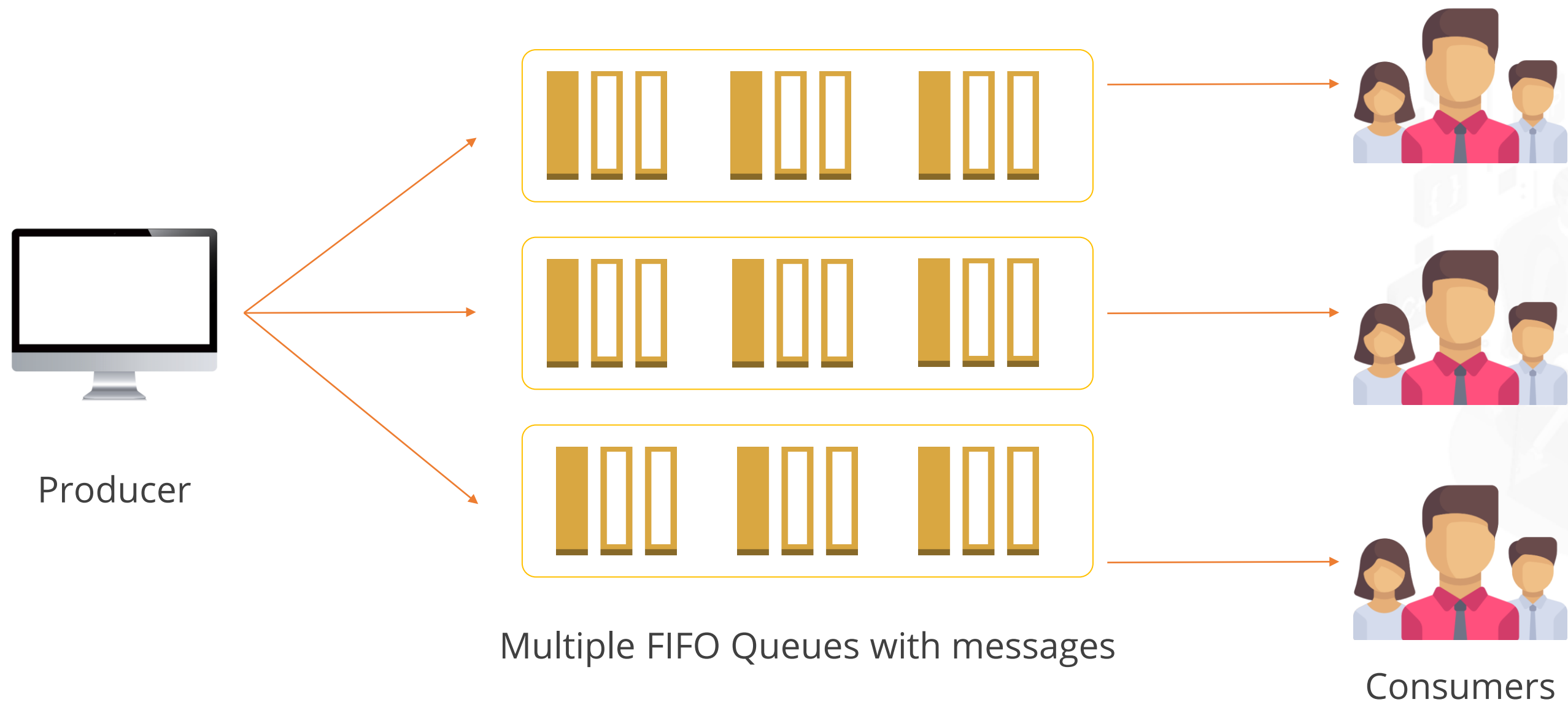
# SQS: Workflow

SQS enables the decoupling of components within a cloud application. This aligns with important AWS best practices for architecting cloud-based architectures.



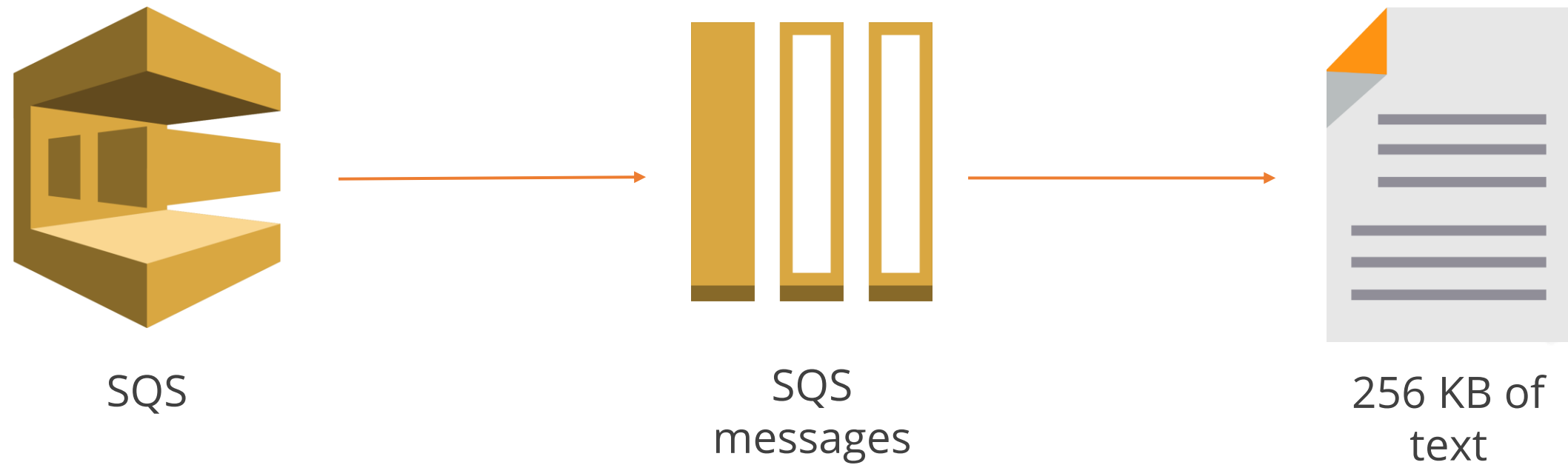
# FIFO Queues

FIFO Queues handle the temporary storage of messages and text strings. They store these messages until they are ready for delivery to the intended consumer, where further processing takes place.



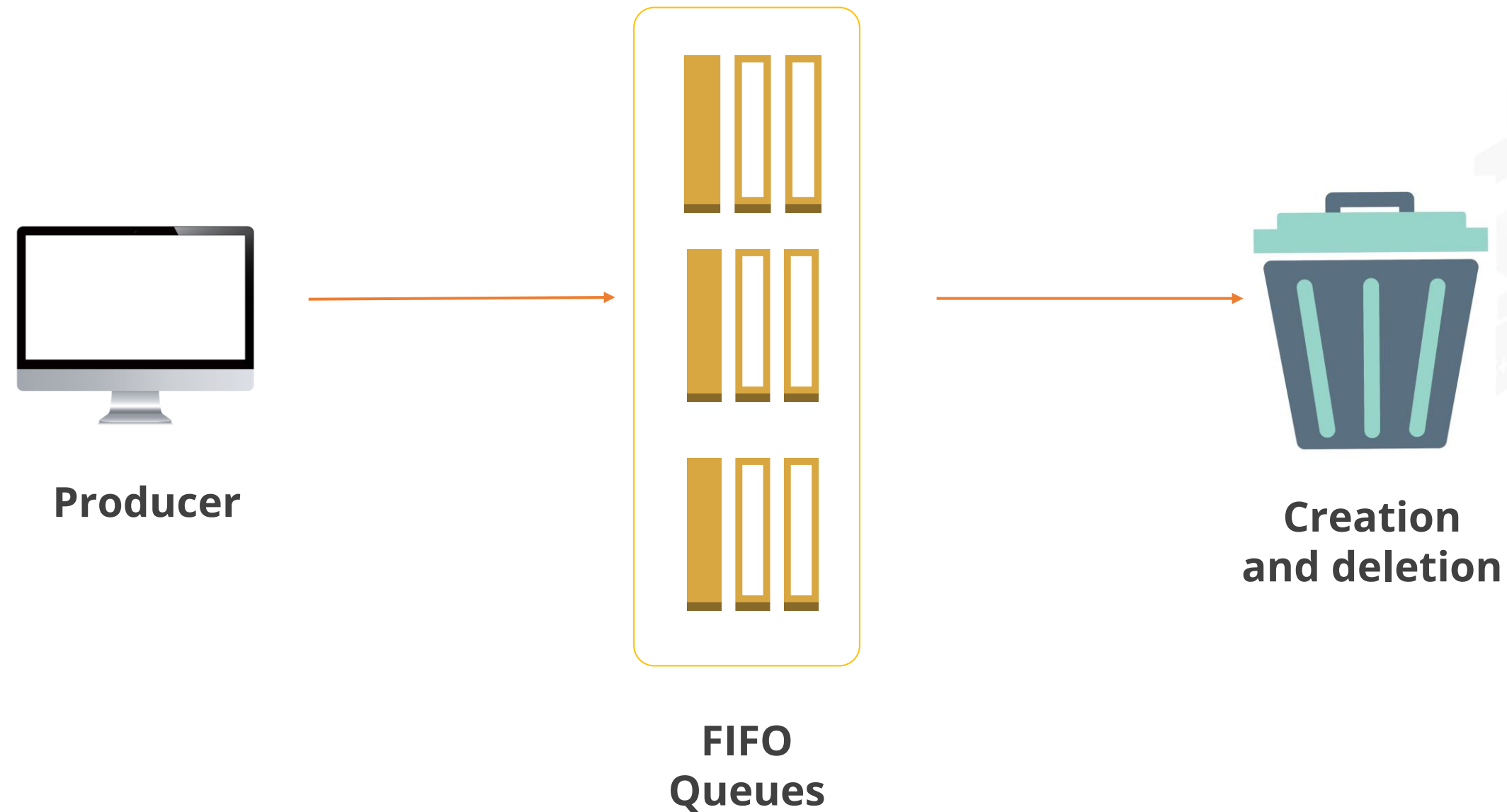
# SQS Messages

SQS messages can hold up to 256 KB of text and are billed in increments of 64 KB of data.



# Messages Lifecycle

It refers to the various stages that a message goes through, starting from its creation and continuing until its eventual deletion.





# Messages Lifecycle

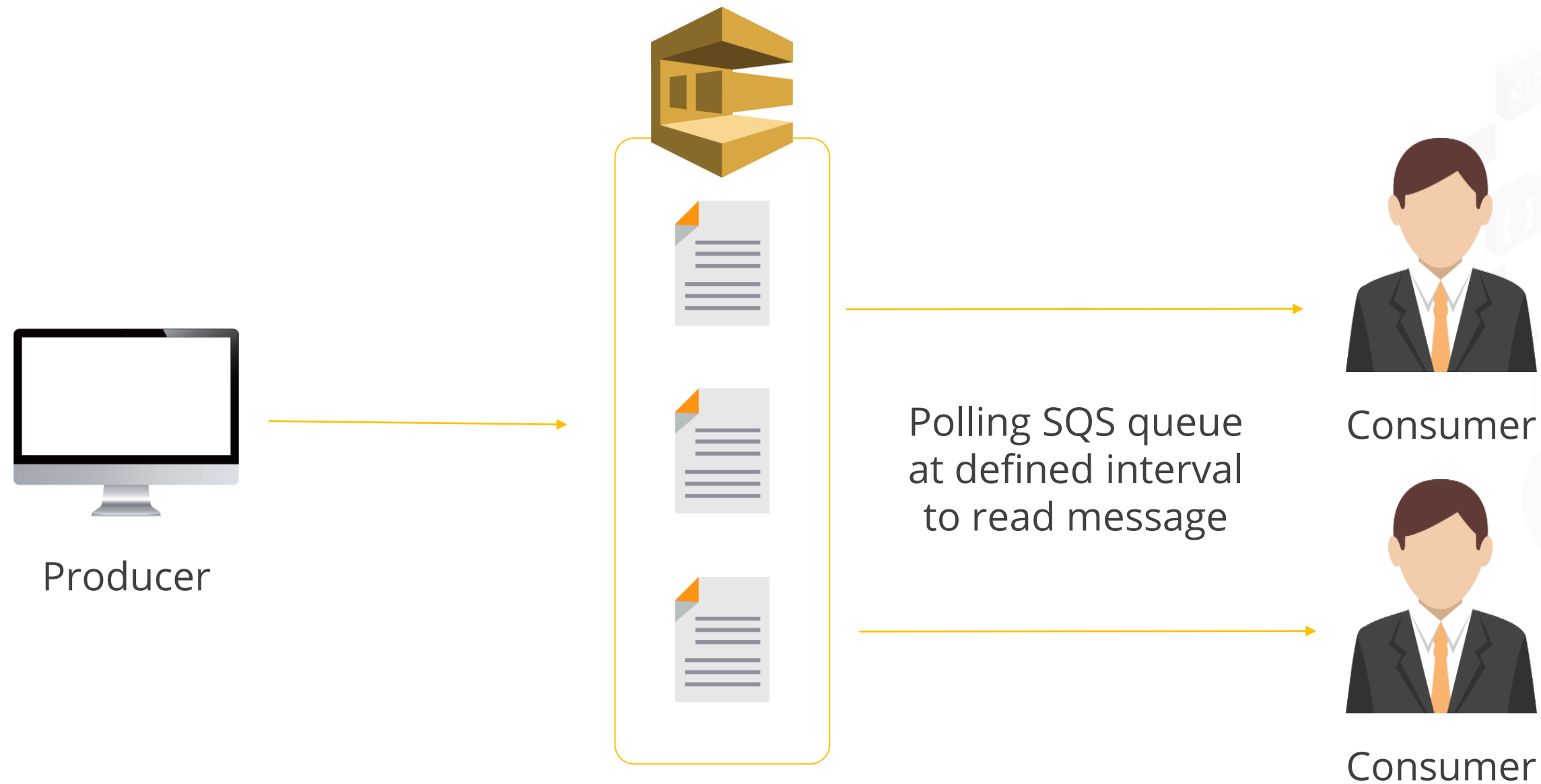
The sent messages are processed by users and automatically deleted by the system.



If users do not receive messages, an error message is sent to the producer. Both users and producers can delete received messages.

# Visibility Timeout

It is the duration during which SQS hides sent messages from other users, preventing them from being read and processed.



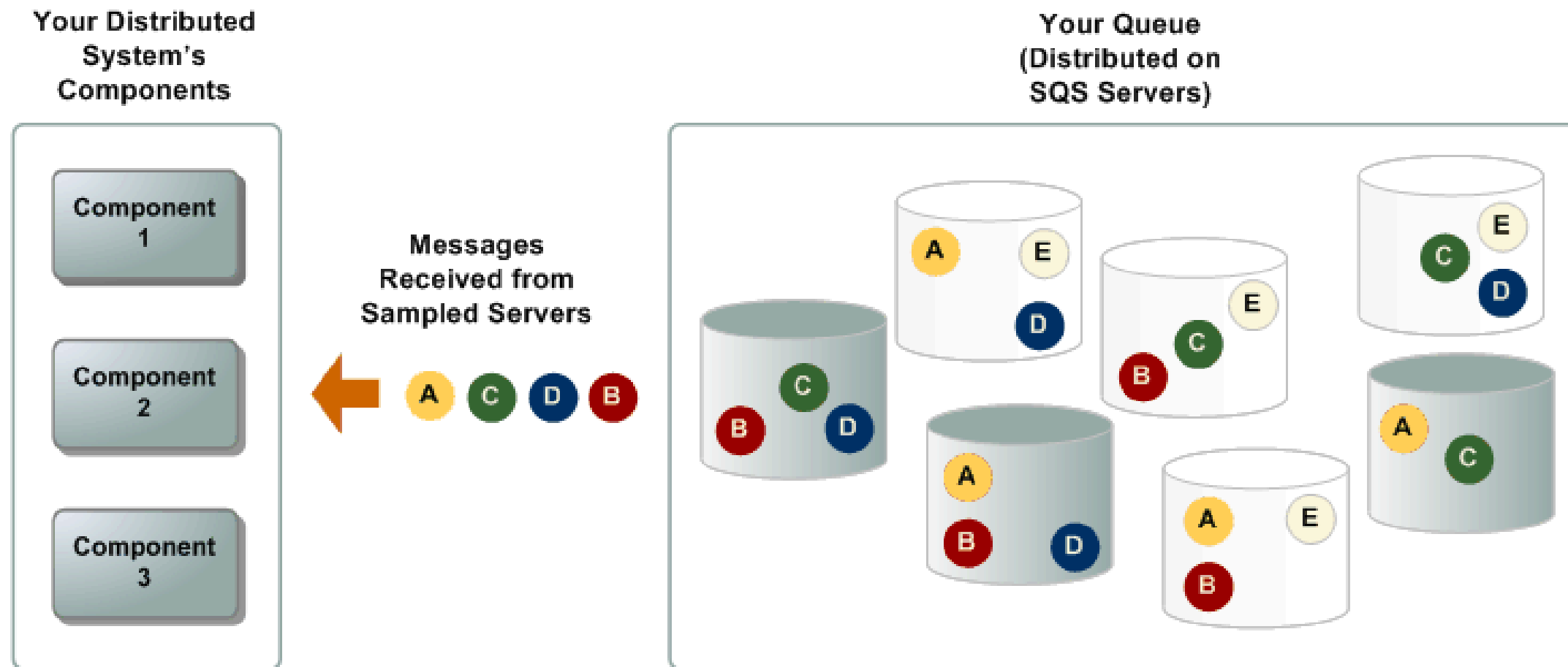
# Visibility Timeout: Overview



- The default value for a visibility timeout is 30 seconds.
- The minimum timeout value is 0 seconds, and the maximum is 12 hours.
- Visibility Timeout safeguards the message's integrity.

# Amazon Long Polling

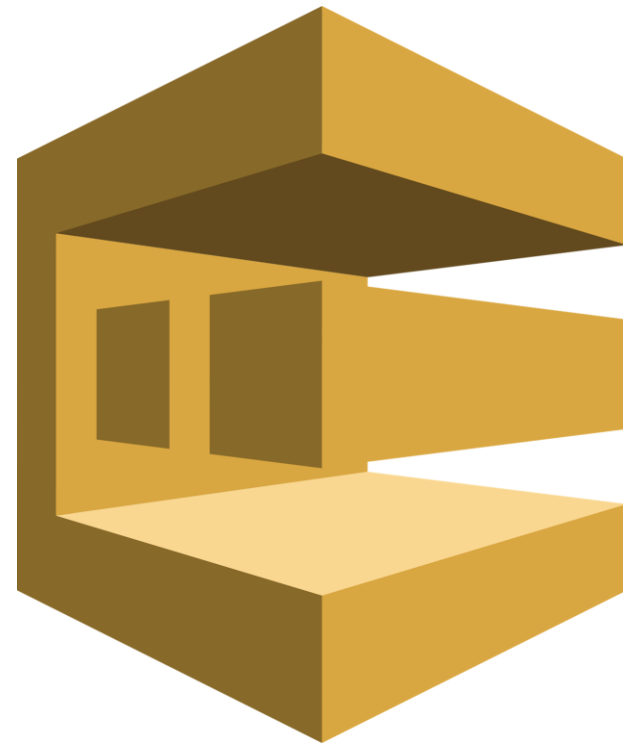
Long polling configures the message retrieval waiting time to a value exceeding 0, with a maximum limit of 20 seconds.



This approach reduces the number of requests, lowers cost, and eliminates false empty responses.

# SQS Batching

SQS batching offers a solution for managing partial failures during the processing of message batches from SQS.

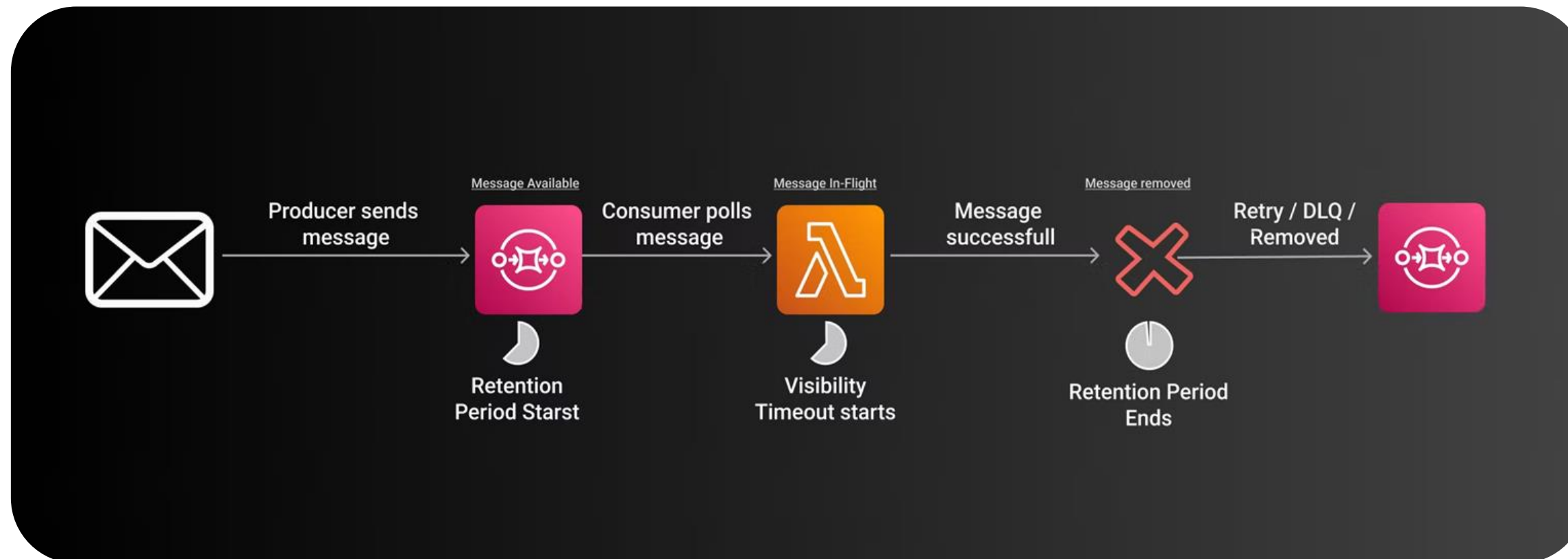


It prevents successfully processed messages from being returned to SQS.



# SQS Retention

Users can configure the Amazon SQS message retention period, setting it anywhere from one minute to fourteen days.

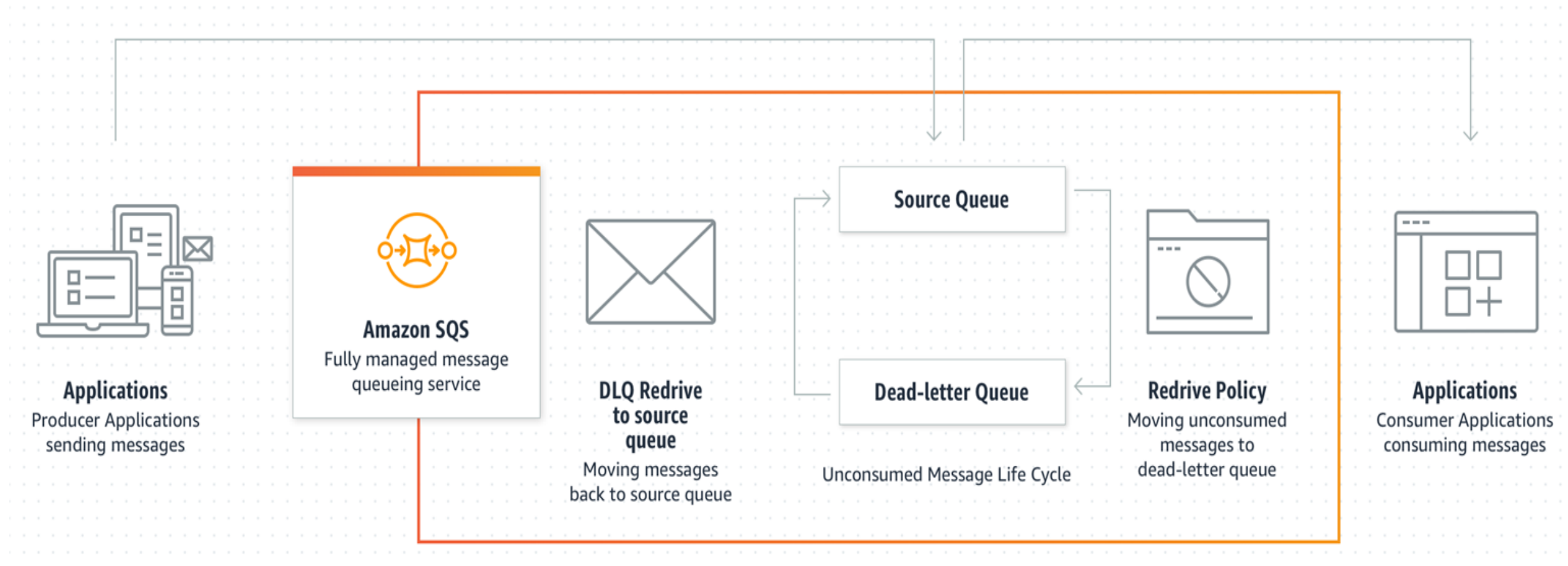


This also includes the automatic deletion of messages once the message retention period elapses.



# SQS Dead Letter Queue (DLQ)

It is a separate SQS queue that receives messages from one or many source queues when those messages cannot be successfully processed or consumed.



It allows applications to isolate and debug messages, especially those that cannot be processed correctly.

# SQS Encryption

SQS employs server-side encryption utilizing the 256-bit Advanced Encryption Standard.



Integration with the AWS Key Management Service (KMS) centrally manages the keys responsible for safeguarding SQS messages.



# Creating SQS Standard Queue, Visibility Timeout, and DLQ



Duration:15 min

## Problem Statement:

You have been assigned a task to create and manage an Amazon SQS Standard Queue, emphasizing the concepts of visibility timeout and Dead Letter Queue (DLQ).

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

## Steps to be followed are:

1. Create a standard SQS Queue
2. Send and receive messages
3. Create message polling
4. Edit visibility timeout
5. Create DLQ



# Creating SQS Queue, Deduplication ID, and Message Group



Duration:15 min

## Problem Statement:

You have been assigned a task to provide a practical understanding of creating and utilizing a FIFO (First-In, First-Out) queue in the Amazon Simple Queue Service (SQS).

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

## Steps to be followed are:

1. Create a Queue
2. Send a message with Group ID and Duplication ID

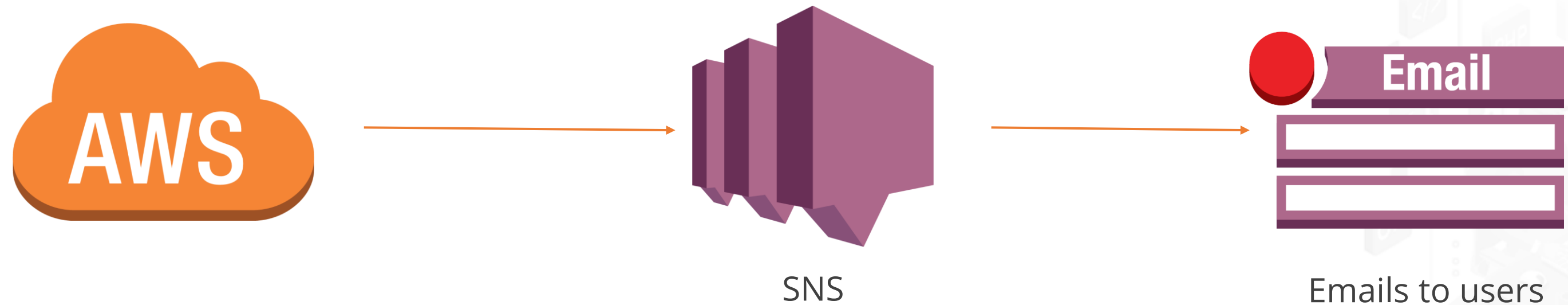


## Amazon Simple Notification Services (SNS)



# Amazon Simple Notification Service (SNS)

Amazon SNS is a fully managed messaging service based on the publication-subscription model.



It is designed to facilitate the sending of push notifications, emails, and SMS messages.

# Amazon SNS: Features

Amazon SNS offers several key features that enhance its functionality, such as:

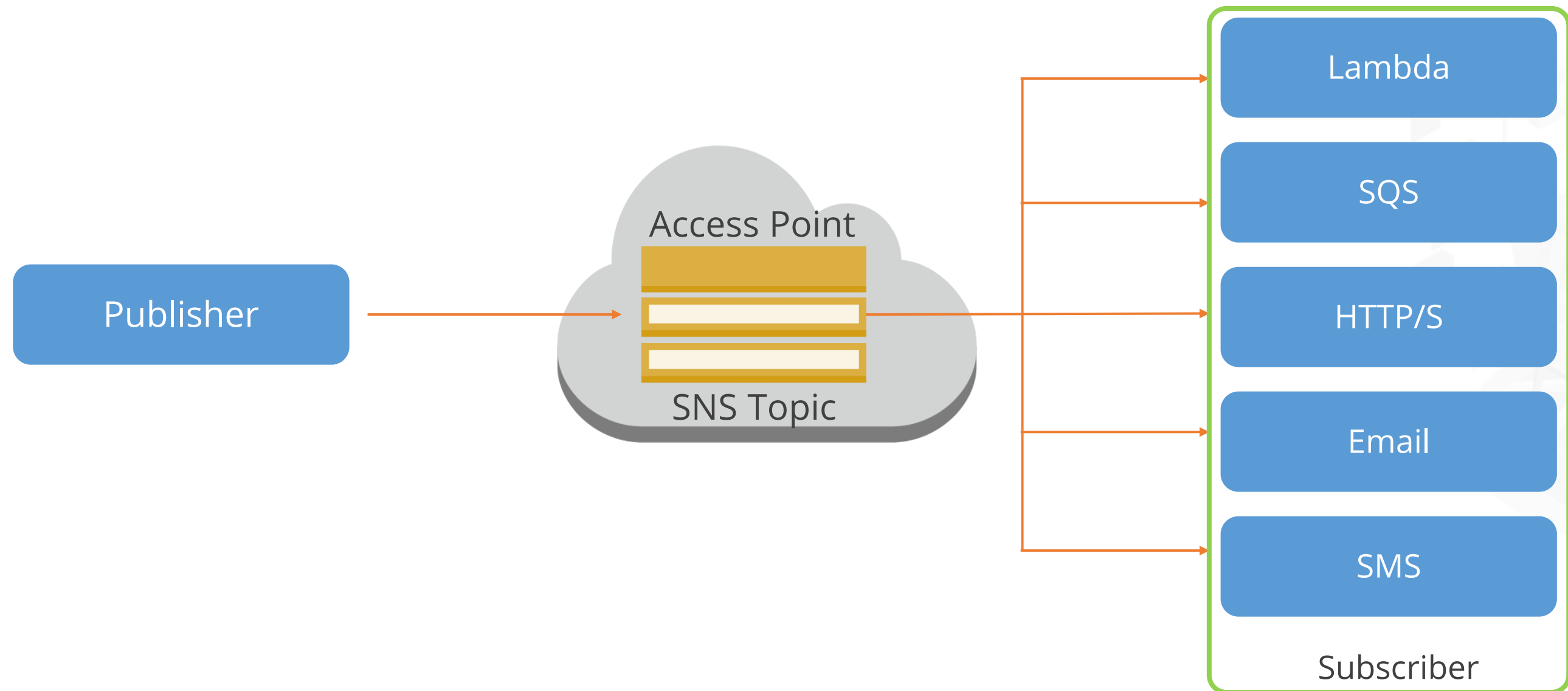


- Instantaneous push-based delivery
- Support for multiple transfer protocols
- Flexible pay-as-you-go model
- User-friendly web-based interface
- Ensured message durability



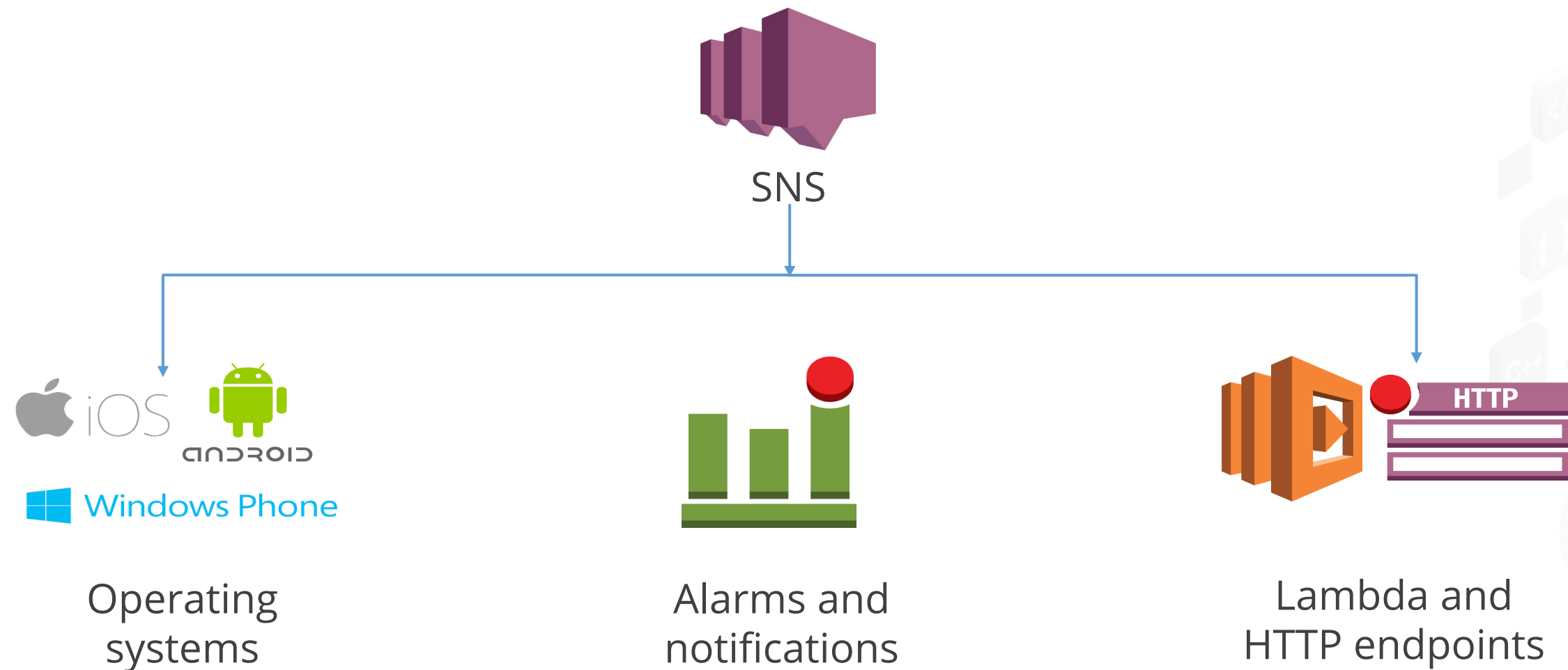
# Amazon SNS Topic

An Amazon SNS topic serves as a communication channel that facilitates message transmission and subscription to notifications.



# Push Model

In the push model, Amazon SNS sends notifications to various operating systems running on mobile devices, including iOS, Android, Windows, and more.



These notifications can be in the form of emails or SMS messages.

# Message Filtering

Message filtering enables subscriber applications to craft filter policies. This capability empowers applications to receive only those notifications that align with their specific interests.



This approach stands in contrast to the conventional practice of receiving every single message published on the topic.



# Logging Amazon SNS API

Amazon SNS seamlessly integrates with AWS CloudTrail, a service designed to maintain a comprehensive record of actions taken by users, roles, or Amazon services.



Log files are generated within CloudTrail, and each contains one or more entries that provide details about specific actions and events.



## Event Bridge



# Amazon Event Bridge

Amazon Event Bridge stands as a serverless event bus service uniquely designed to enable users to seamlessly connect their applications to a diverse range of data sources.

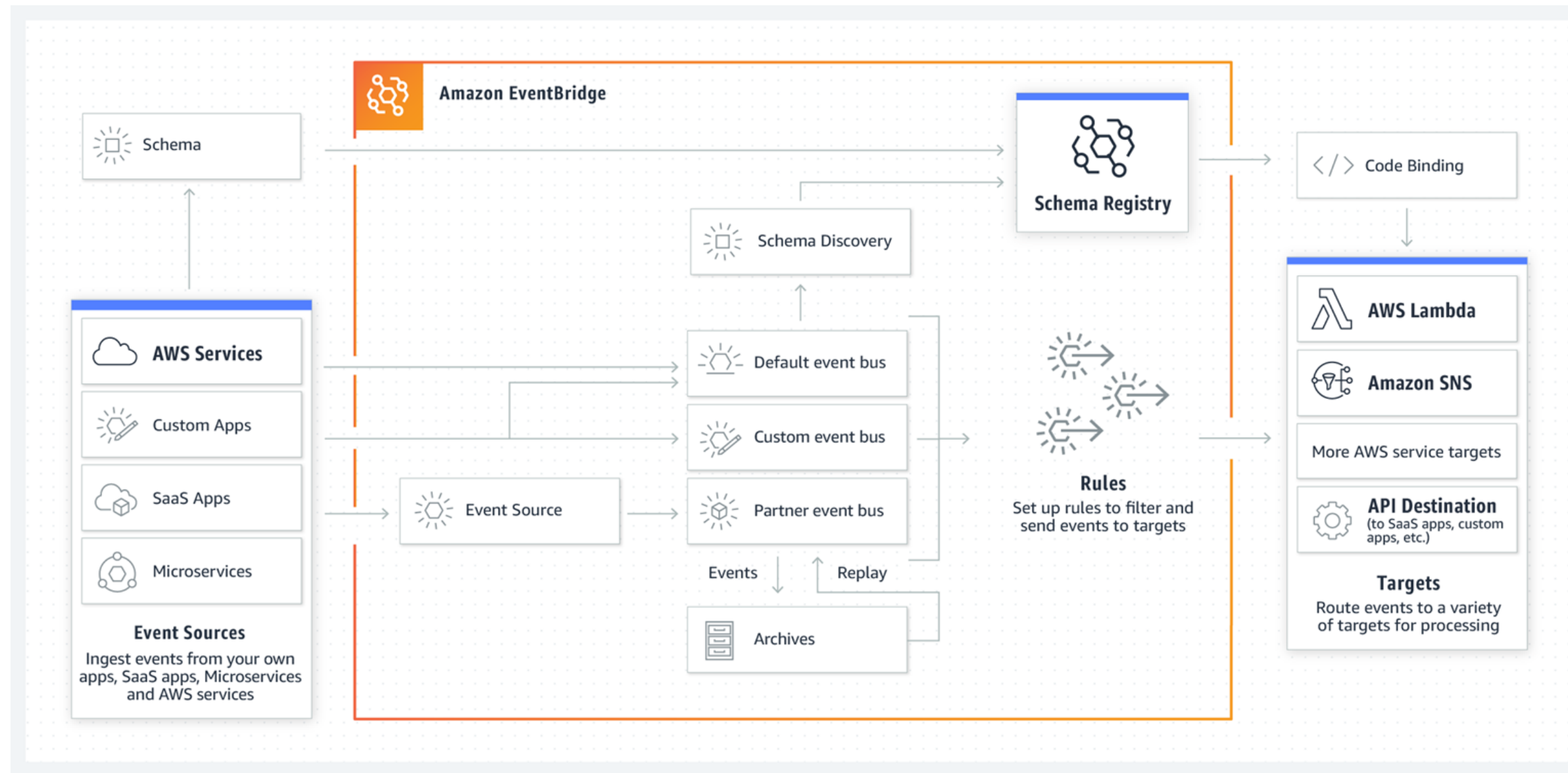


This service acts as a conduit for delivering real-time data from various origins, including a user's proprietary applications, Software as a Service (SaaS) applications, and an array of AWS services.



# Amazon Event Bridge

The diagram below illustrates the operational mechanism of an Event Bridge:



# Amazon Event Bridge: Features

The features of the event bridge are as follows:

1

Advanced event  
rules filtering

2

Content-based  
event filtering

3

Schema registry

4

Message  
transformation

5

Custom events

6

Archive and  
replay events

7

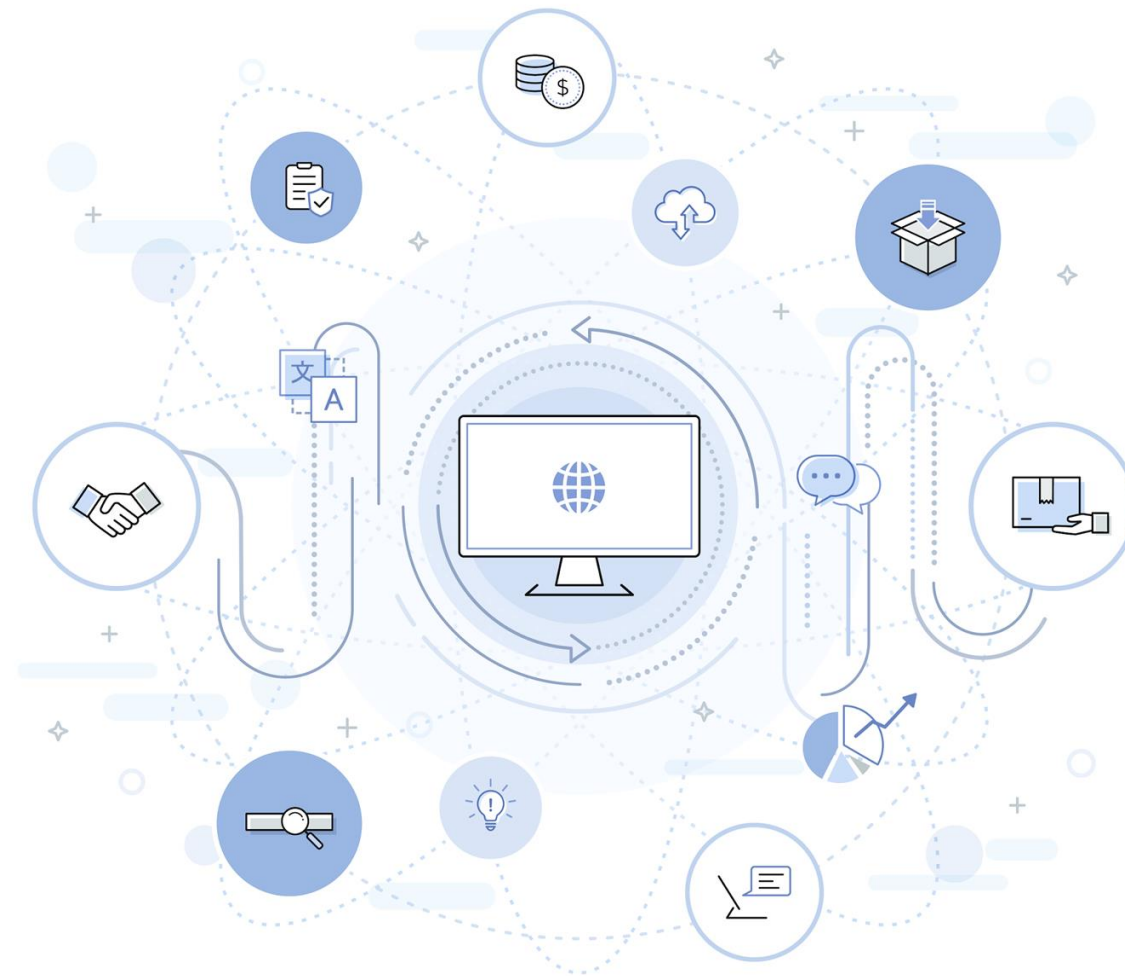
SaaS Apps  
Integration

8

API destination

## Event Bus

An event bus receives events from a source and directs them to rules associated with that event bus. A custom event bus can receive events from custom applications and services.



# Creating SNS Topic, Fanout, and S3 Event Notification



**Duration:15 min**

## **Problem Statement:**

You have been assigned a task to create an Amazon SNS topic, subscribe to it, and set up an S3 event notification to highlight the fanout capability of SNS.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

## Steps to be followed are:

1. Create an SNS topic
2. Subscribe to the SNS topic



## Kinesis



# Amazon Kinesis

Amazon Kinesis stands as a fully managed and remarkably scalable service engineered to facilitate the real-time collection, processing, and analysis of streaming data.



Amazon Kinesis



# Amazon Kinesis Capabilities



**Kinesis Data Streams:** It builds custom applications to process data in real time.

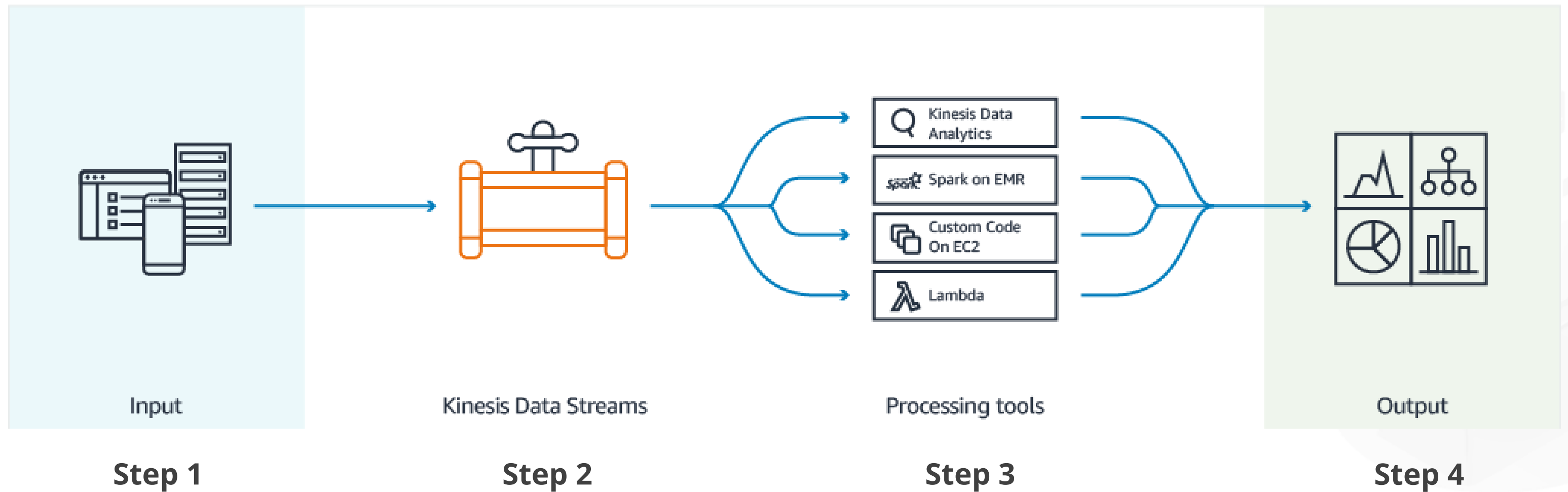
**Kinesis Videos Streams:** It securely streams videos from connected devices to AWS for processing.

**Kinesis Data Firehose:** It captures, transforms, and loads data into AWS data stores.

**Kinesis Data Analytics:** It runs queries against the data in real time.

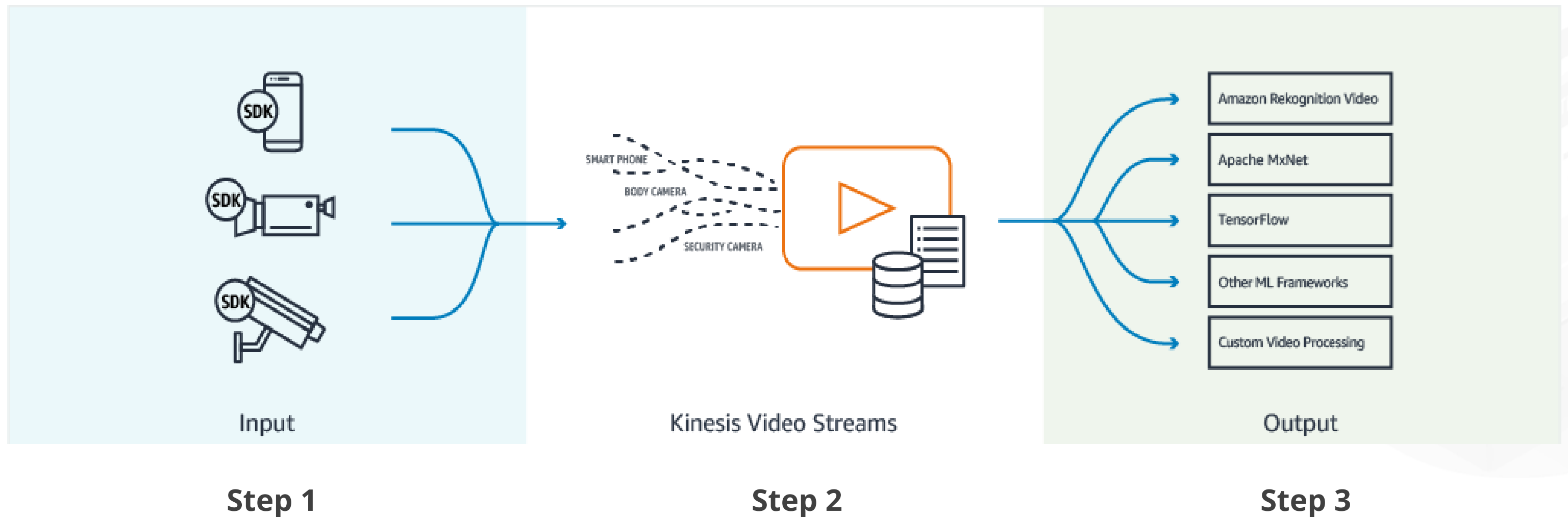
# Amazon Kinesis Data Streams

The following diagram shows the working of Amazon Kinesis Data Streams:



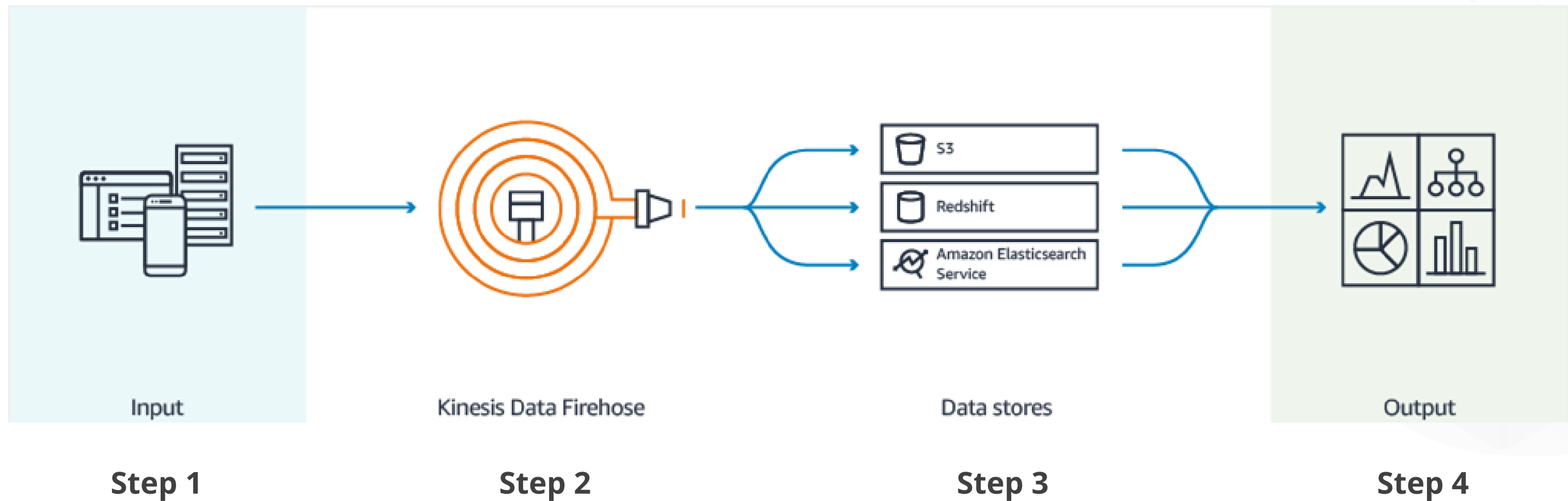
# Amazon Kinesis Video Streams

The following diagram shows the working of Amazon Kinesis Video Streams:



# Amazon Kinesis Data Firehose

The following diagram shows the working of the Amazon Kinesis Data Firehose:



# Amazon Kinesis Data Analytics

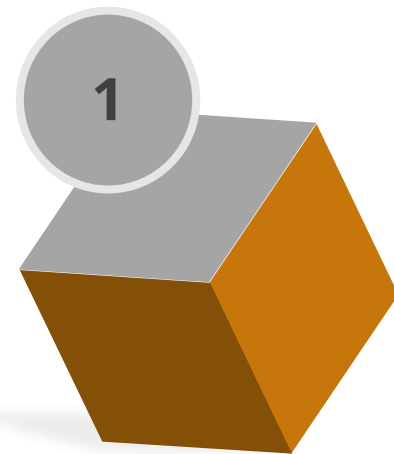
The following diagram shows the working of Amazon Kinesis Data Analytics:



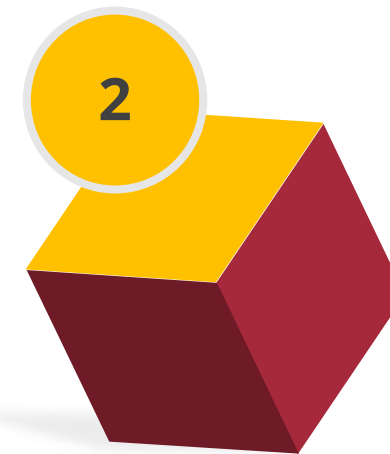
# Kinesis Shards

A Kinesis shard is a fundamental unit within a data stream comprising a sequence of data records. It serves as the foundational throughput component for a Kinesis data stream.

The key features of Kinesis shards include:



Support up to five transactions per second for reads



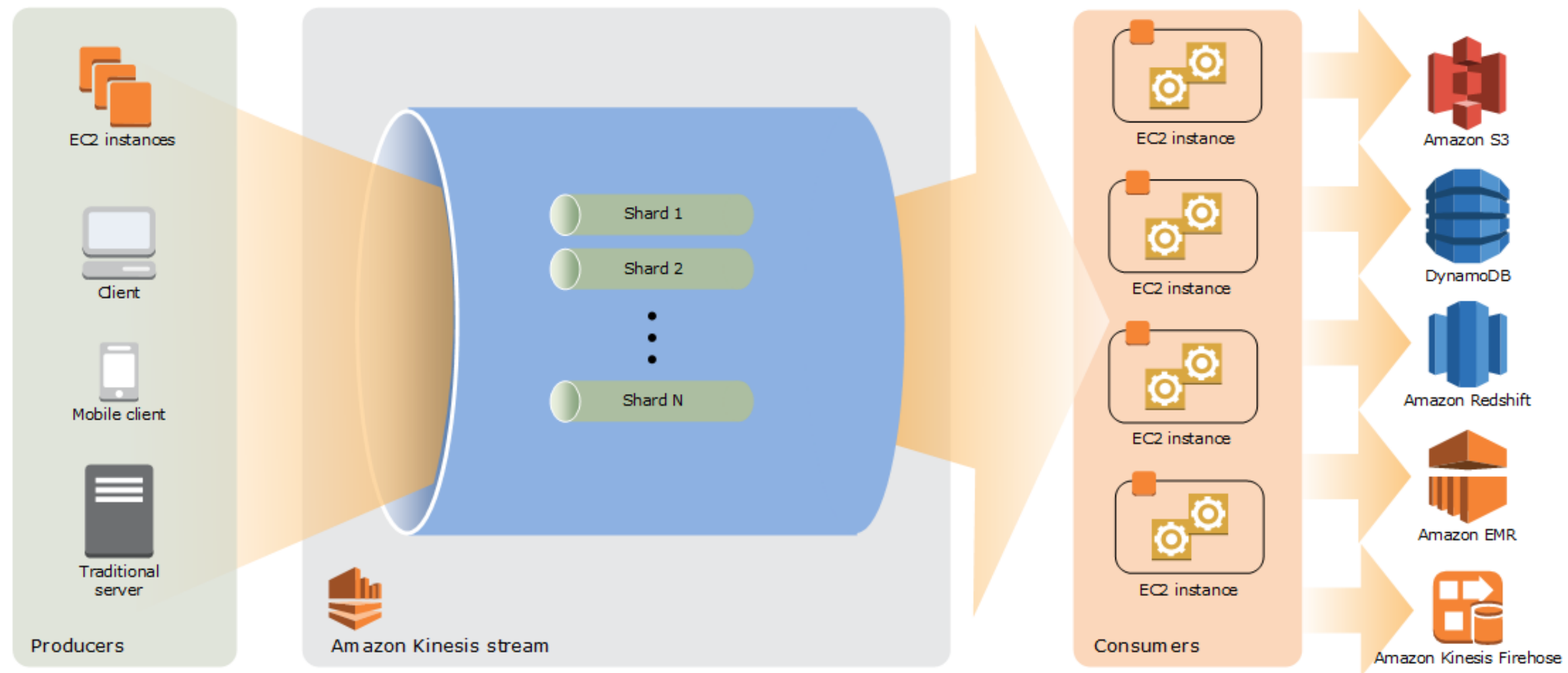
Support for a maximum total data read rate of 2MB per second





# Kinesis Shards

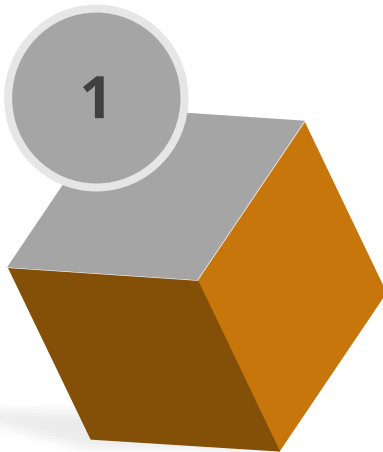
The following diagram depicts the workflow of kinesis shards:



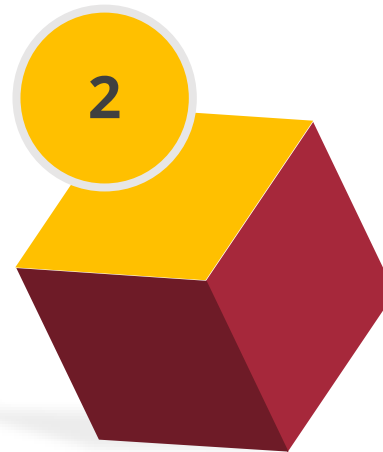
# Resharding

Resharding provides users with the capability to adjust the number of shards within a data stream, facilitating seamless adaptation to fluctuations in the rate of data flow.

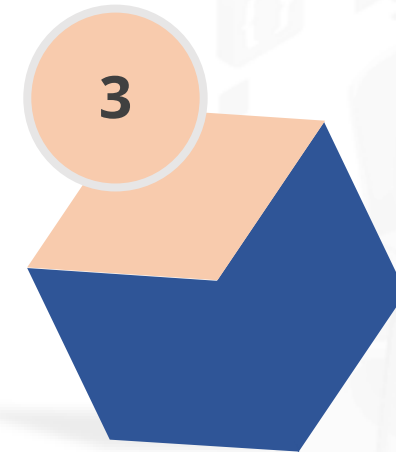
The key features of resharding include:



Dynamic scaling



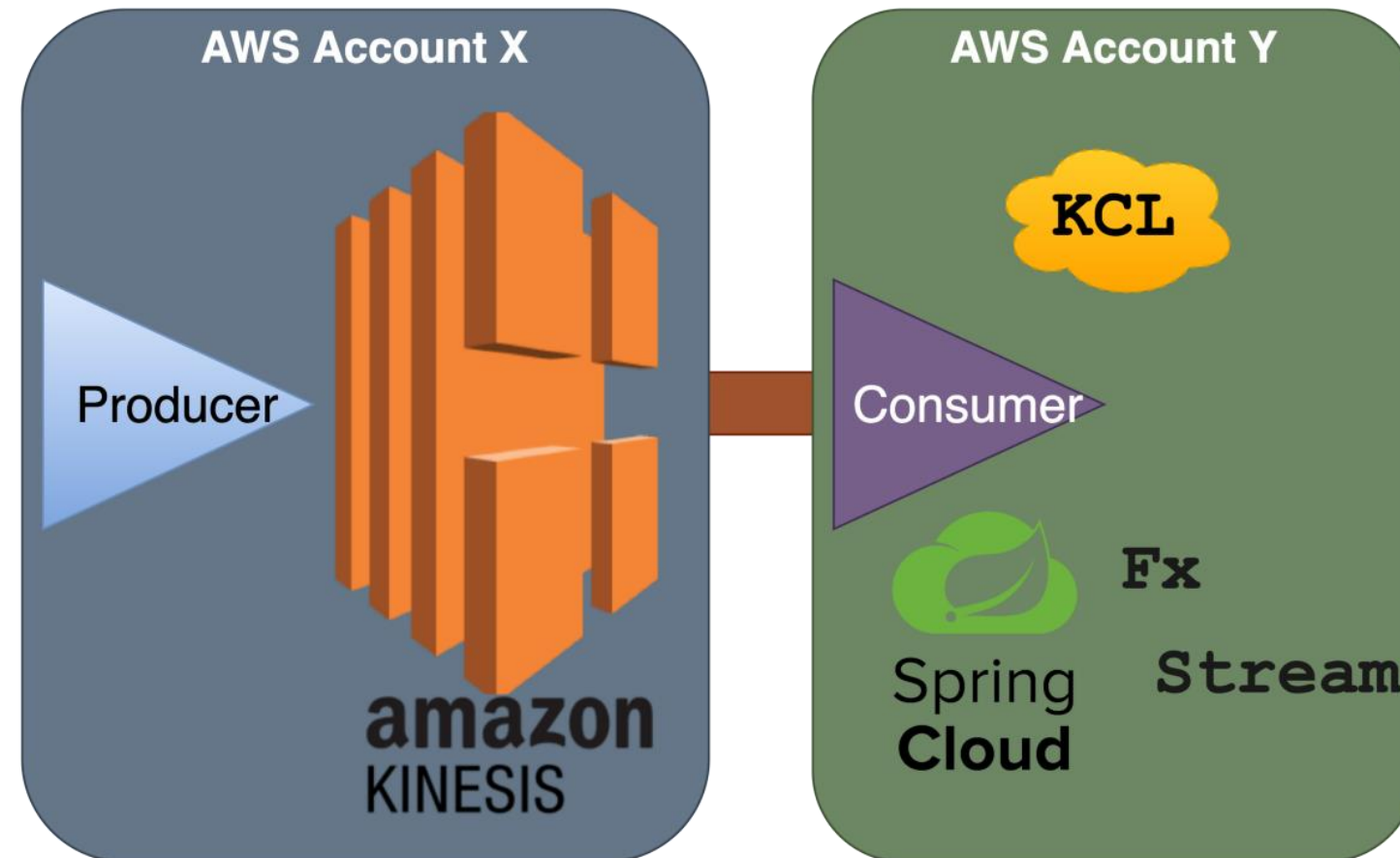
Pairwise operation



Seamless adaptation

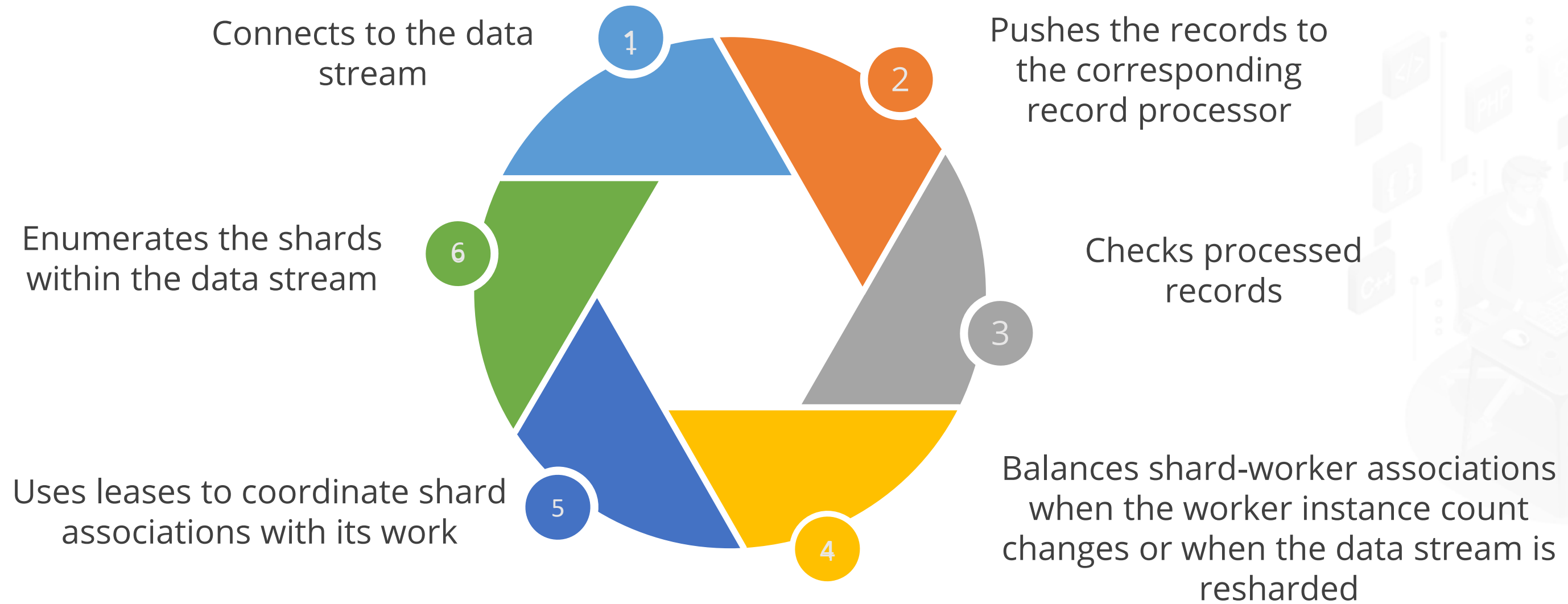
# Kinesis Client Library (KCL)

The Kinesis Client Library (KCL) plays a pivotal role in the development of custom consumer applications designed to process data originating from Kinesis Data Streams (KDS).



# Kinesis Client Library: Features

The features of KCL are as follows:



# Creating a Kinesis Data Stream



**Duration:10 min**

## **Problem Statement:**

You have been assigned a task to create a Kinesis Data Stream in Amazon Kinesis to set up a data stream, configure its capacity, and understand the concept of shards as a fundamental resource in Kinesis.

ASSISTED PRACTICE

# Assisted Practice: Guidelines

---

**Steps to be followed are:**

1. Configure a Kinesis Data Stream



# Creating a Kinesis Data Firehose



**Duration:10 min**

## **Problem Statement:**

You have been assigned a task to create a Kinesis Data Firehose, set up an Amazon S3 bucket as the destination, and test the data delivery stream.

ASSISTED PRACTICE



# Assisted Practice: Guidelines

---

## Steps to be followed are:

1. Configure a Kinesis Data Firehose
2. Create an S3 bucket



## Key Takeaways

- Amazon SQS is a fast, reliable, scalable, and fully managed message queuing service. It can be used to control workflow processes.
- Amazon SNS is a fully managed publication-subscription-based messaging service that sends push notifications, emails, and SMS messages.
- Amazon Lambda is a computing service that runs code in response to events and automatically manages the computing resources required by that code.
- Amazon Kinesis is a fully managed and scalable service that allows real-time collection, processing, and analysis of streaming data.



# Initiating a Lambda Function to Copy Content in S3 Buckets

Duration: 30 mins



**Project agenda:** To launch the Lambda function and perform the trigger operation on Lambda

**Description:** As a solution architect, you must replicate the contents of a bucket to other buckets deployed in different regions, thereby making the resources available.

## Perform the following:

1. Create an S3 Bucket
2. Create a Lambda Function
3. Test the Lambda Function
4. Create a trigger
5. Add a destination

# TECHNOLOGY

**Thank You**