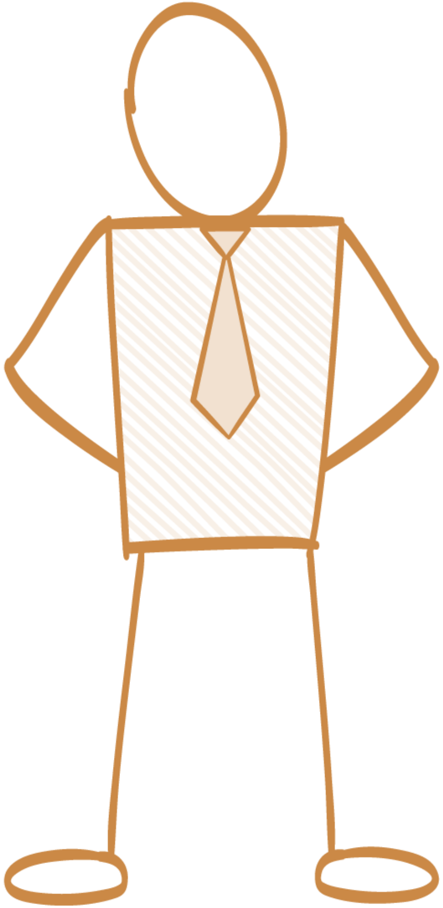
Training and Certification

Architecting on AWS Student Guide

Version 3.1

100-ARC-31-EN-SG



Module 9: Overview of Application Services

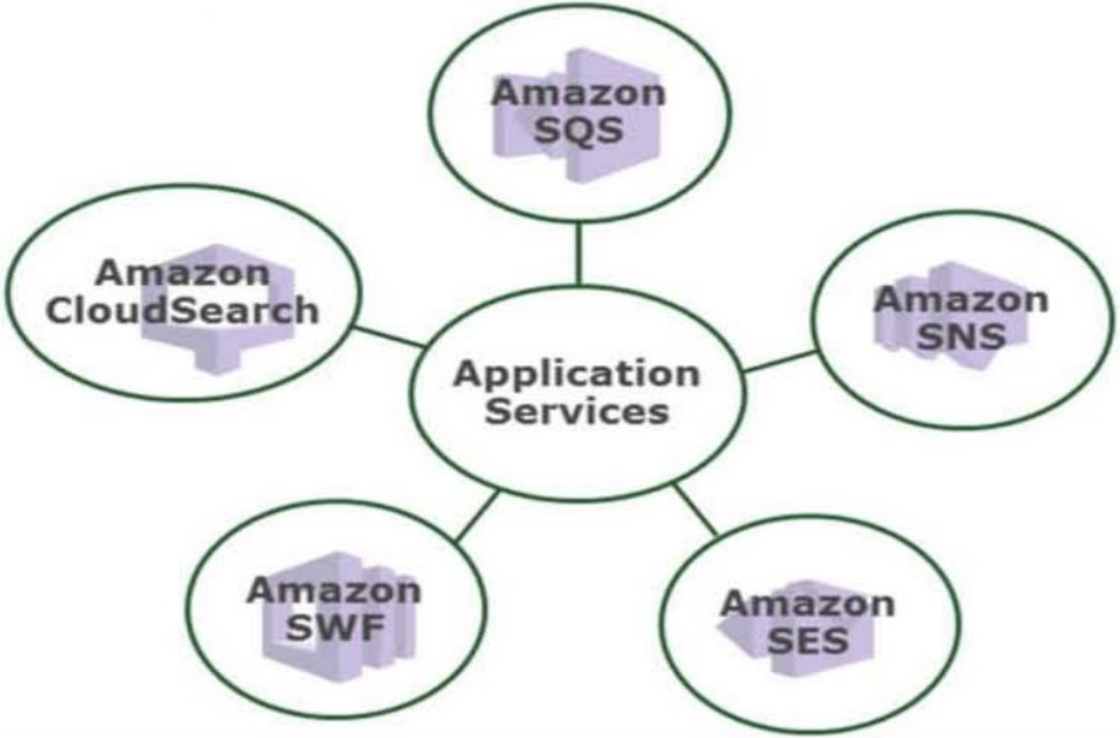
Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES)
  - Amazon CloudSearch

Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES)
  - Amazon CloudSearch

AWS Application Services



Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES)
  - Amazon CloudSearch

Amazon Simple Queue Service (SQS)

- Hosted queue for storing messages as they travel between computers or processes
- Move data between distributed components of their applications

AWS Solutions Architect Associate

Simple Queue Service

Simple Queue Service Introduction

Simple Queue Service (SQS)

Fully managed queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications

What is a Queveing System?

What is a Messaging System?
Used to provide asynchronous communication and decouple processes via messages / events
From a sender and receiver (producer and consumer)

Queueing

Generally will delete messages once they are consumed. Simple communication. **Not Real-time**
Have to pull. Not reactive.

VS

Streaming

Multiple consumers can **react** to events (messages), Event live in the stream for long periods of time, so complex operations can be applied. **Real-time**

SQS

Sidekiq

RabbitMQ

Kinesis

APACHE kafka

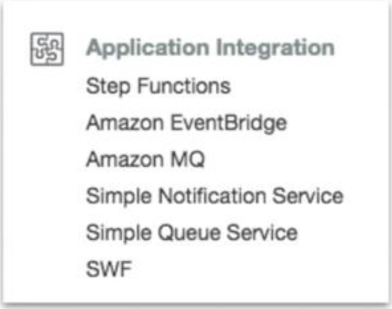
NATS

Simplified example because there are lots of ifs and buts

Introduction to SQS

SQS is for **Application Integration.**

AWS SQS is a solution for the distributed **queuing** of messages generated by your application. It connects isolate applications together by passing along message to one another.



Application Integration

Step Functions

Amazon EventBridge

Amazon MQ

Simple Notification Service

Simple Queue Service

SWF

A **queue** is a temporary repository for messages that are awaiting processing.

Using the **AWS SDK** you write code which publishes messages onto the queue or you pull the queue for messages.

SQS is pull based

Not push based!

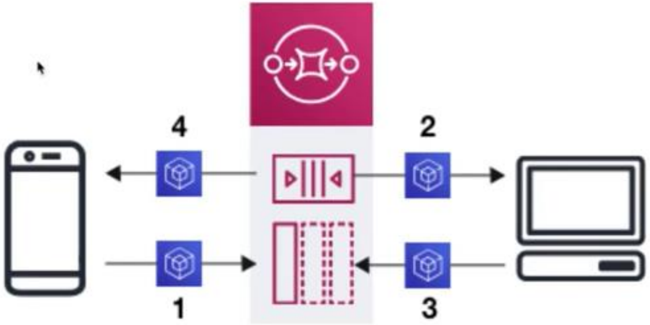
AWS Solutions Architect Associate

Simple Queue Service

SQS Use Case

SQS – Use Case

1. App publishes messages to the queue
2. Other app pulls the queue and find the message and does something
3. Other app reports that they completed their task and marks the message for completion
4. Original app pulls the queue and sees the message is no longer in the queue.



Both apps are using the **AWS SDK** to push messages and pull the queue.

AWS Solutions Architect Associate

Simple Queue Service

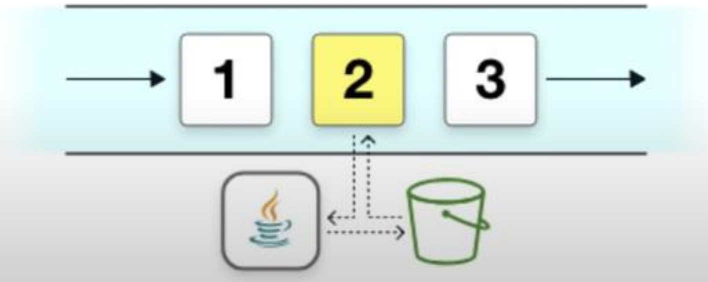
SQS Limits & Retention

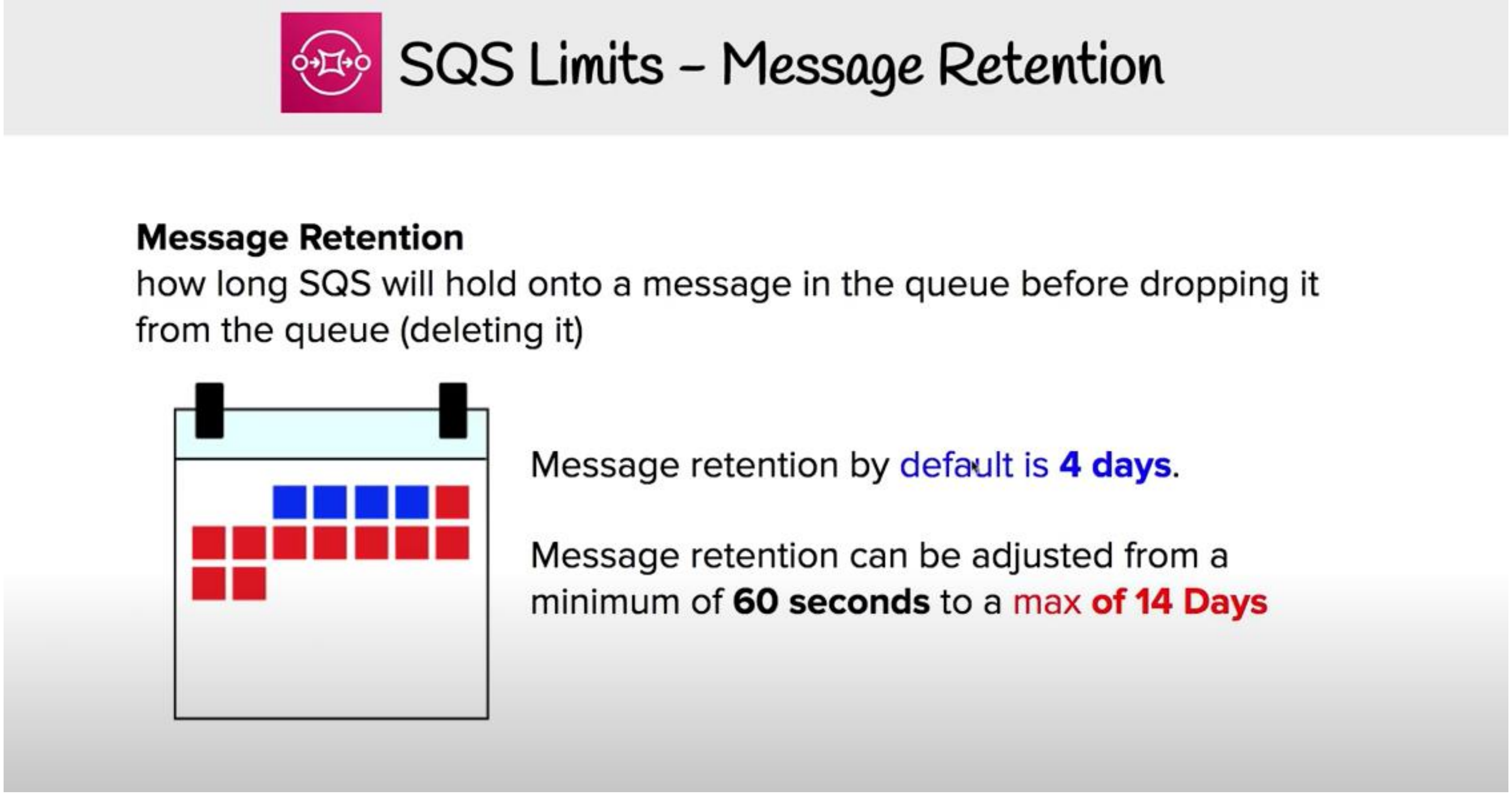
SQS Limits – Message Size

Message Size

The size of a message can be between **1 byte and 256 KB**

Amazon SQS Extended Client Library for Java lets you send messages **256KB to 2GB** in size. The message will be stored in S3 and library will reference the S3 object.





SQS Limits – Message Retention

Message Retention
how long SQS will hold onto a message in the queue before dropping it from the queue (deleting it)

Message retention by default is **4 days**.

Message retention can be adjusted from a minimum of **60 seconds** to a max **of 14 Days**

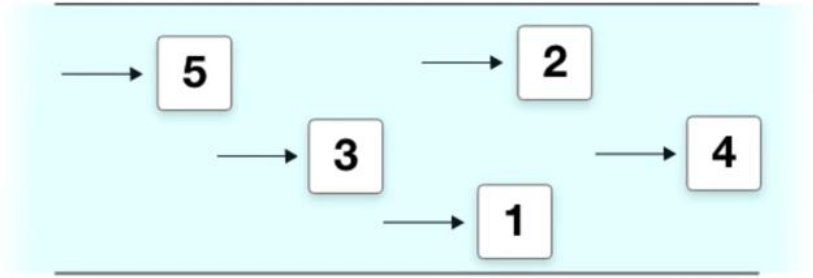
AWS Solutions Architect Associate

Simple Queue Service

SQS Queue Types

amazon web services

SQS – Standard Queues



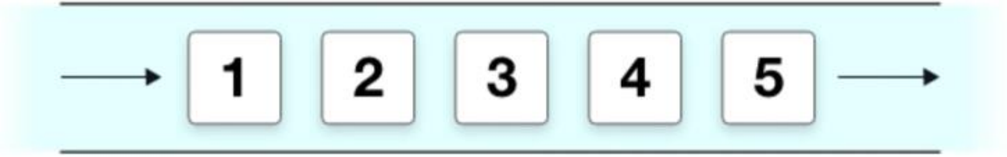
AWS SQS **Standard Queues** allow you a **nearly-unlimited** number of transactions per second.

Guarantees that a message will be delivered **AT LEAST once**.

More than one copy of a message could be potentially delivered **out of order**.

Provides **best-effort ordering** that helps ensure a message is generally delivered in the same order that it was sent.

SQS – FIFO Queues



AWS SQS **First-In-First-Out** queues support multiple ordered message groups within a single queue.

Limited to 300 transactions per second.

SWS FIFO queues have all the same capabilities of a Standard Queue

AWS Solutions Architect Associate

Simple Queue Service

SQS Visibility Timeout

SQS – Visibility Timeout

How do we prevent another app from reading a message while another one Is busy with that message? (Avoid someone doing the same task)

A **visibility time-out** is the period of time that messages are **invisible in the SQS queue**, after a reader picks up that message.

Messages will be **deleted** from the queue **after a job has processed**. (before the visibility timeout expires)

If a job is **NOT** processed before the visibility time-out period, the message will **become visible again** and another reader will process it.

This can result in the same message being delivered **twice**!

30 seconds (default)
0 seconds minimum
12 hours maximum

AWS Solutions Architect Associate

Simple Queue Service

Short vs Long Polling

SQS – Short vs Long Polling

Polling is the method in which we retrieve messages from the queues.

Short polling (default) returns messages immediately, even if the message queue being polled is empty.

When you need a message **right away**. shorting polling is what you want to use.

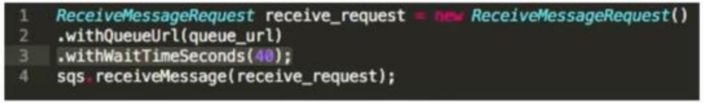
Long polling waits until message **arrives in the queue**, or the **long poll timeout expires**.

Long polling makes it **inexpensive to retrieve messages** from your queue as soon as the messages are available.

Using long polling will reduce the cost because you can **reduce the number of empty receives**.

Most use-cases you want to use Long Polling

You can enable long polling when receiving a message by setting the wait time in seconds on the **ReceiveMessageRequest**



```
ReceiveMessageRequest receive_request = new ReceiveMessageRequest()
.withQueueUrl(queue_url)
.withWaitTimeSeconds(40);
sqs.receiveMessage(receive_request);
```

AWS Solutions Architect Associate

Simple Queue Service

SQS Cheat Sheet

amazon web services

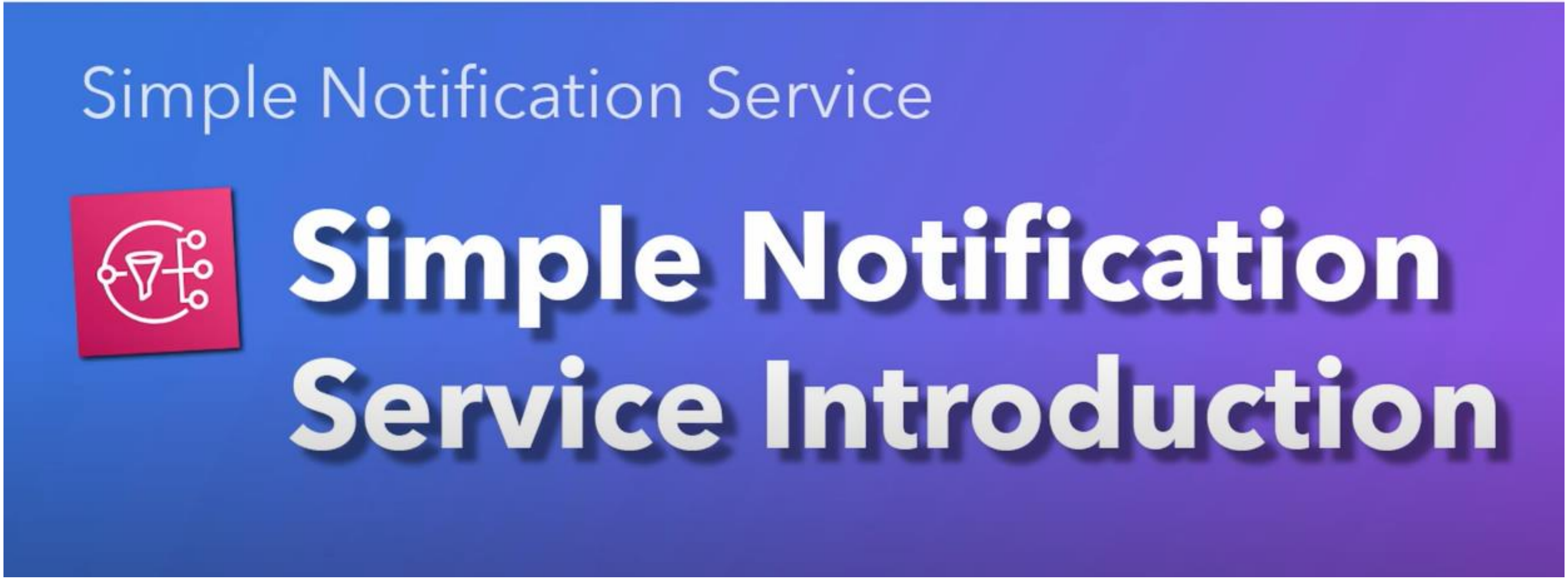
Simple Queue Service (SQS) *CheatSheet*

- SQS is a queuing service using messages with a queue. Think Sidekiq or RabbitMQ
- SQS is used for Application Integration, it lets decoupled services and apps to talk to each other
- To read SQS use need to **pull** the queue using the AWS SDK. SQS is **not pushed-based**
- SQS supports both Standard and First-In-First-Out (FIFO) queues
- Standard allows nearly unlimited messages per second, does not guarantee order of delivery, always delivers at least once, you must protect again duplicate messages being processed
- FIFO maintain the order of messages with a 300 limit
- There are two kinds of polling Short (Default) and Long Polling
- Short polling returns messages immediately, even if the message queue being polled is empty.
- Long polling waits until message arrives in the queue, or the long poll timeout expires.
- In majority of cases Long polling is preferred over short polling.
- **Visibility time-out** is the period of time that messages are invisible in the SQS queue
- Messages will be deleted from queue after a job has processed. (before visibility timeout expires)
- If Visibility timeout expires than a job will become visible to the queue
- The default Visibility time-out is 30 seconds. Timeout can be **0 seconds** to a maximum of 12 hours.
- SQS can retain messages from 60 seconds to 14 days and by default is 4 days
- Message size between 1 byte to 256 kb, Extended Client Library for Java can increase to 2GB

Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES) – beta
  - Amazon CloudSearch - beta

AWS Solutions Architect Associate



Simple Notification Service

Simple Notification Service Introduction

Simple Notification Service (SNS)

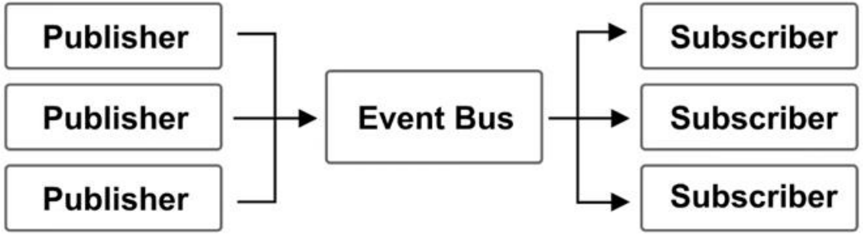
Subscribe and **send notifications** via
text message email, webhooks, lambdas, SQS and mobile notifications

Introduction to SNS

What is Pub/Sub?

Publish–subscribe pattern commonly implemented in **messaging systems**. In a pub/sub system the sender of messages (**publishers**) do not send their messages directly to receivers. They instead send their messages to an **event bus**. The event bus categorizes their messages into groups. Then receivers of messages (**subscribers**) subscribe to these groups. Whenever new messages appear within their subscription the messages are immediately delivered to them.

- Publisher have no knowledge of who their subscribers are.
- Subscribers do **not pull** for messages.
- Messages are instead automatically and immediately **pushed** to subscribers.
- Messages and events are interchangeable terms in pub/sub

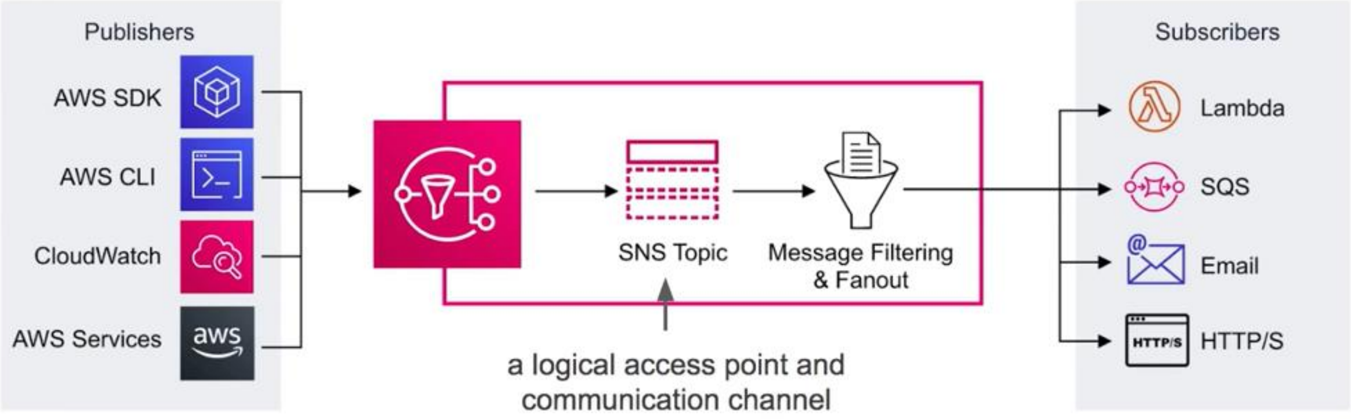


Introduction to SNS

Simple Notification Service (SNS) is a highly available, durable, secure, fully managed **pub/sub messaging** service that enables you to **decouple** microservices, distributed systems, and serverless applications.

`Application Integration!`

- Publishers **push** events to an SNS Topic
- Subscribers subscribe to SNS Topic to have events **pushed** to them



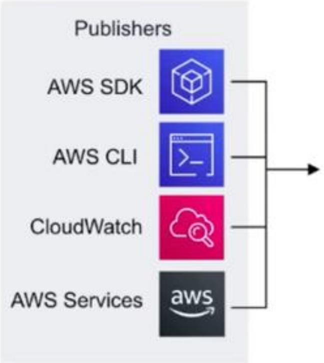
AWS Solutions Architect Associate

Simple Notification Service

SNS Topics

SNS – Topics

Publishers don't care about the subscribers protocol

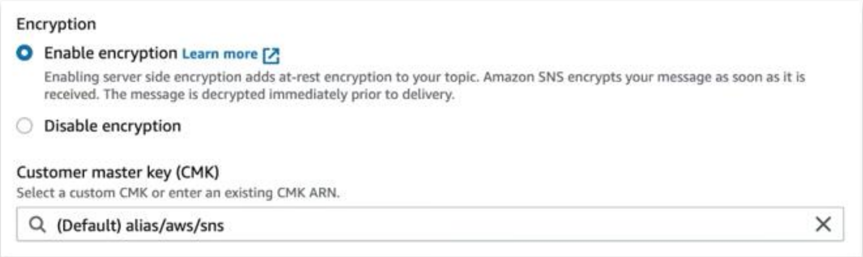


Topics allow you to group multiple subscriptions together.

A topic is able to deliver to multiple protocols at once eg. email, text message, http/s

When topics deliver messages to subscribers it will automatically format your message according to the subscriber's chosen protocol

You can encrypt Topics via **KMS**



Encryption

● Enable encryption Learn more

Enabling server side encryption adds at-rest encryption to your topic. Amazon SNS encrypts your message as soon as it is received. The message is decrypted immediately prior to delivery.

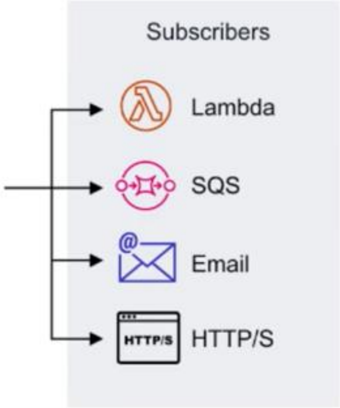
○ Disable encryption

Customer master key (CMK)

Select a custom CMK or enter an existing CMK ARN.

(Default) alias/aws/sns

Subscribers listen for incoming messages

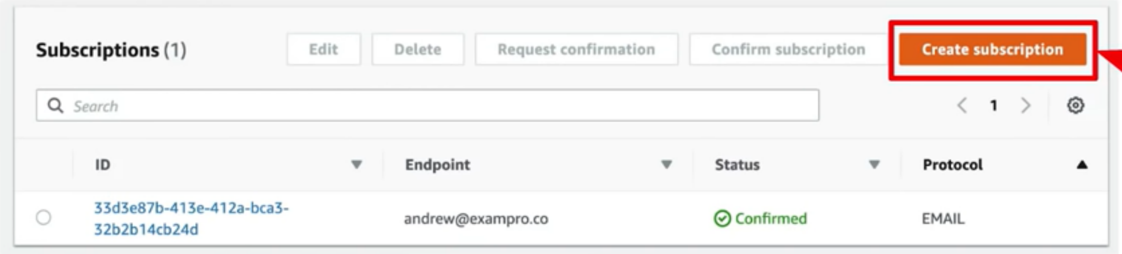


AWS Solutions Architect Associate

Simple Notification Service

SNS Subscriptions

SNS – Subscriptions



Subscriptions (1)

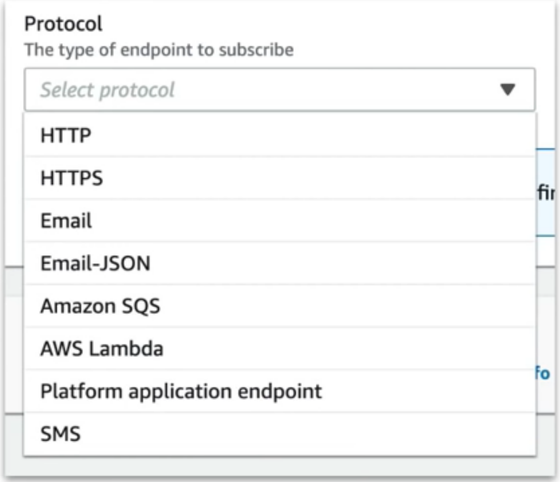
	ID	Endpoint	Status	Protocol
○	33d3e87b-413e-412a-bca3-32b2b14cb24d	andrew@exampro.co	✓ Confirmed	EMAIL

To receive messages from a topic you need to create a Subscription.

A subscription can only subscribe to one protocol and one topic.

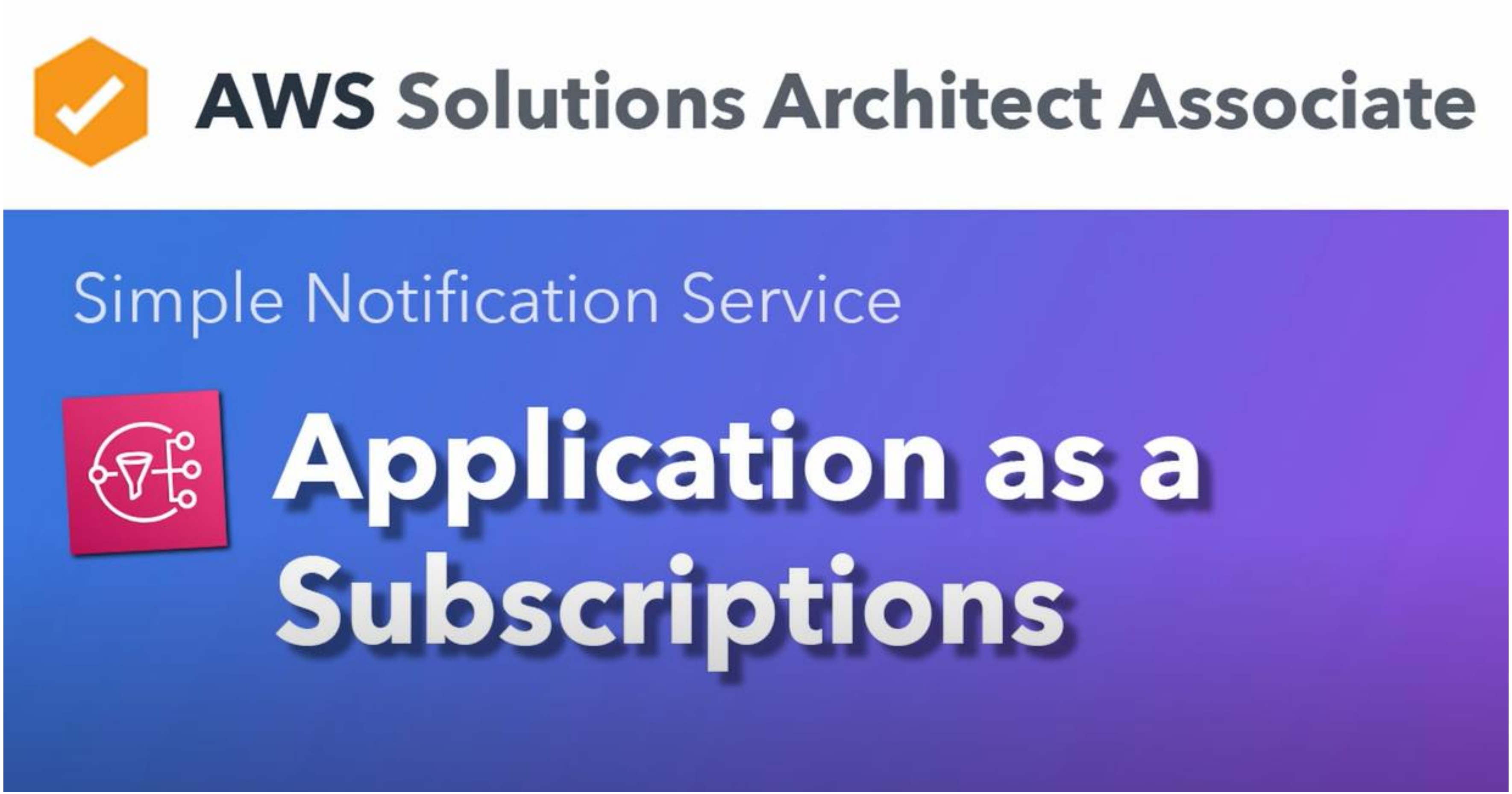
The following protocols:

HTTPs and HTTPs create webhooks into your web-application
Email good for internal email notifications (only supports **plain text**)
Email-JSON sends you json via email
Amazon SQS place SNS message into SQS queue
AWS Lambda triggers a lambda function
SMS send a text message
Platform application endpoints Mobile Push



Protocol
The type of endpoint to subscribe

Select protocol ▼



AWS Solutions Architect Associate

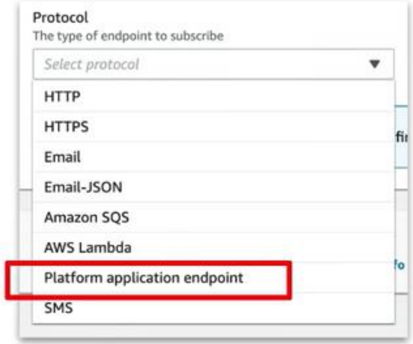
Simple Notification Service

Application as a Subscriptions

Application As Subscriber

Send push notification messages directly to apps on **mobile devices**.

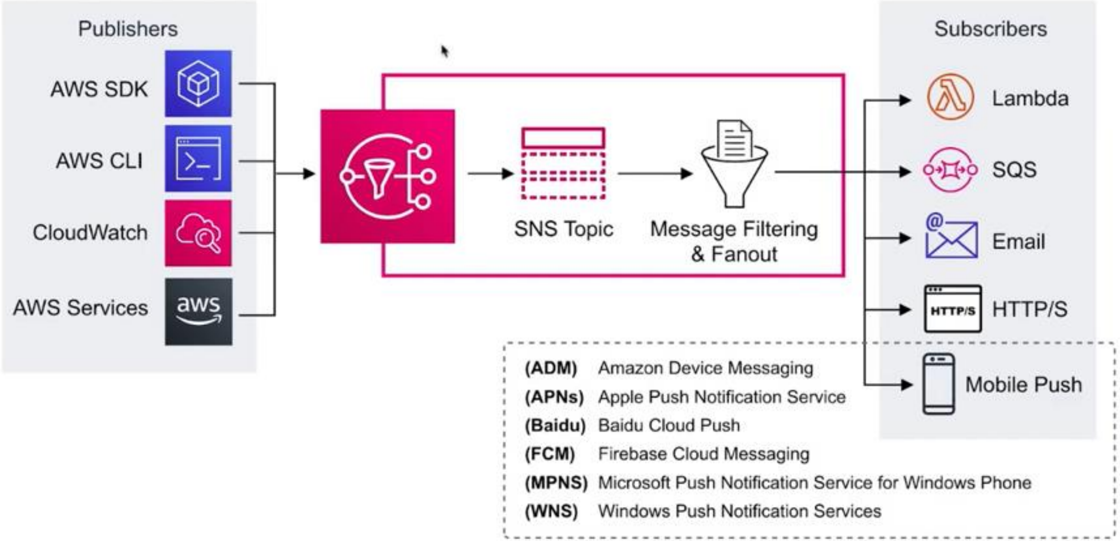
Push notification messages sent to a mobile endpoint can appear in the mobile app as message alerts, badge updates, or even sound alerts.



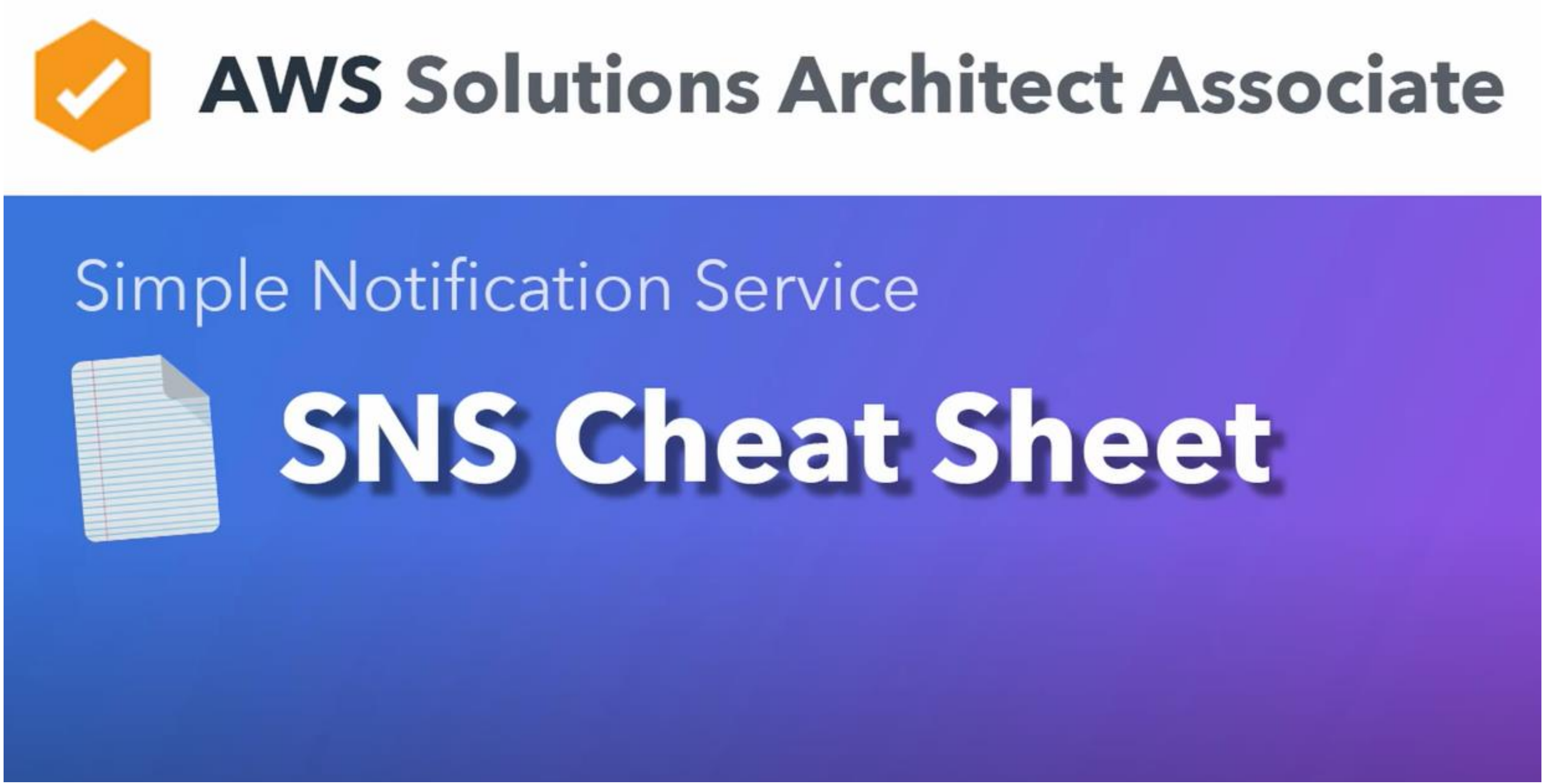
Protocol
The type of endpoint to subscribe

Select protocol ▼

- HTTP
- HTTPS
- Email
- Email-JSON
- Amazon SQS
- AWS Lambda
- Platform application endpoint
- SMS



(ADM)	Amazon Device Messaging
(APNs)	Apple Push Notification Service
(Baidu)	Baidu Cloud Push
(FCM)	Firebase Cloud Messaging
(MPNS)	Microsoft Push Notification Service for Windows Phone
(WNS)	Windows Push Notification Services



AWS Solutions Architect Associate

Simple Notification Service

SNS Cheat Sheet

SNS CheatSheet

- **Simple Notification Service (SNS)** is a fully managed pub/sub messaging service
- SNS is for **Application Integration**. It allows decoupled services and apps to communicate with each other
- **Topic** a logical access point and communication channel.
- A topic is able to deliver to multiple protocols
- You can encrypt topics via KMS
- **Publishers** use the AWS API via AWS CLI or SDK to push messages to a topic. Many AWS services integrate with SNS and act as publishers.
- **Subscriptions** subscribe to topics. When a topic receives a message it automatically and immediately pushes messages to subscribers.
- All messages published to SNS are stored redundantly across multiple Availability Zones (AZ)
- The following protocols:
    - **HTTPs and HTTPs** create webhooks into your web-application
    - **Email** good for internal email notifications (only supports **plain text**)
    - **Email-JSON** sends you json via email
    - **Amazon SQS** place SNS message into SQS queue
    - **AWS Lambda** triggers a lambda function
    - **SMS** send a text message
    - **Platform application endpoints** Mobile Push eg. Apple, Google, Microsoft Baidu notification systems

Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES) – beta
  - Amazon CloudSearch - beta

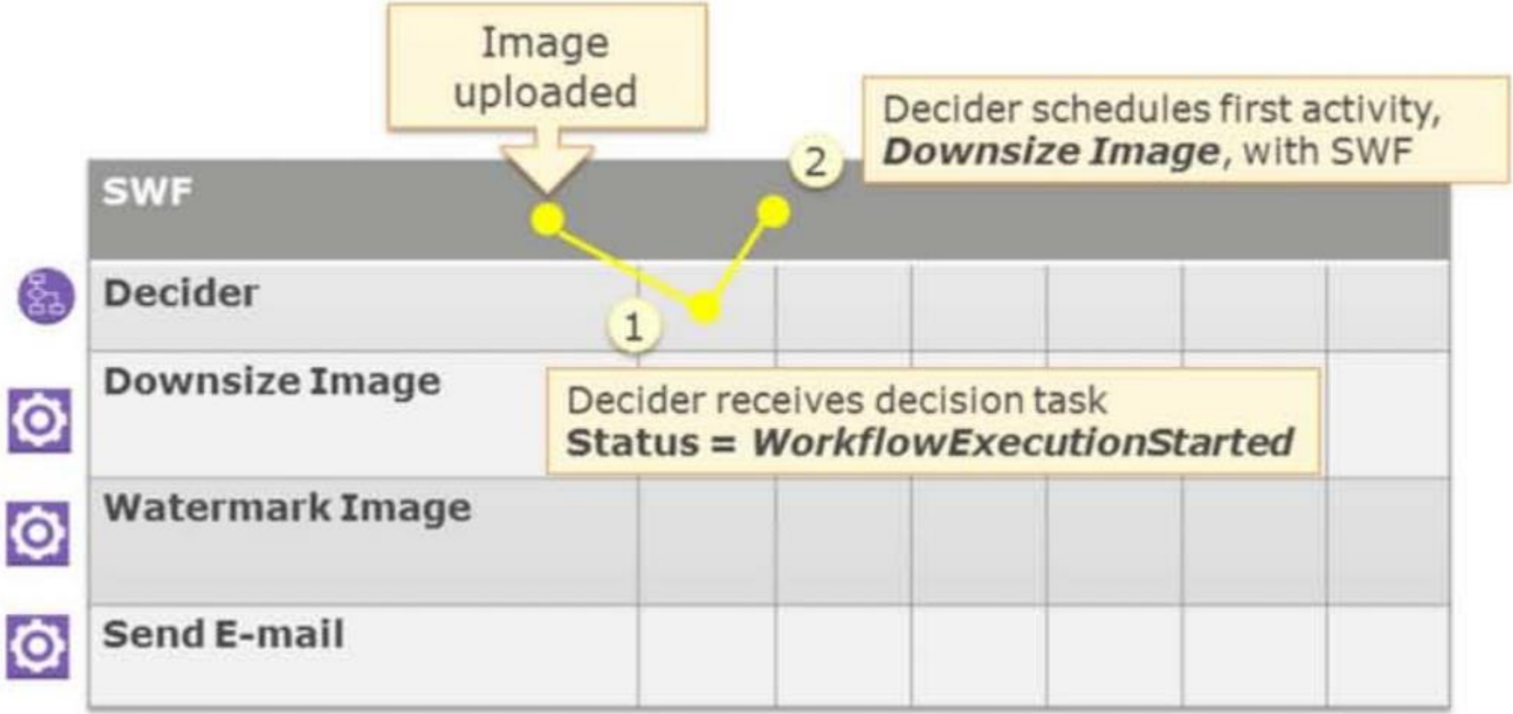
Amazon Simple Workflow Service (SWF)

- Manage workflows, including state, decisions, executions, tasks and logging
- Coordinate processing steps across distributed systems
- Ensure tasks are executed reliably, in order, and without duplication
- HTTP API: executed from code written in any language
- Use from any app that supports HTTP, from EC2 to mobile device in a warehouse

Amazon SWF: Decider and Activity Worker

- Decider
  - Decision logic (i.e., flow control)
  - Decoupled from application logic
  - Makes decision based on events assigned by SWF
- Activity Worker
  - Does work assigned by SWF
  - Work scheduled by Decider
  - Run anywhere

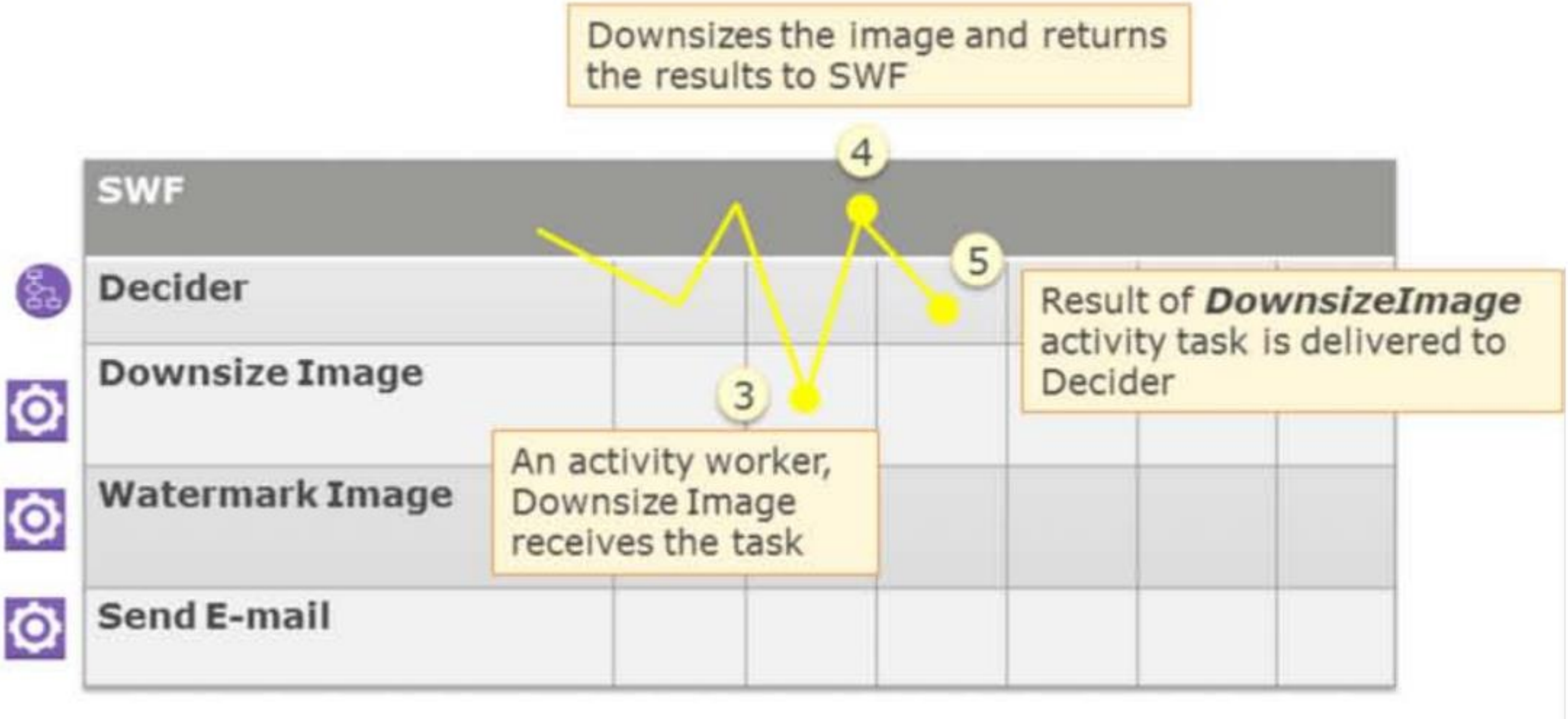
Watermarking application example with SWF (1 of 4)



Decider schedules first activity, ***Downsize Image***, with SWF

Decider receives decision task
Status = *WorkflowExecutionStarted*

Watermarking application example with SWF (2 of 4)

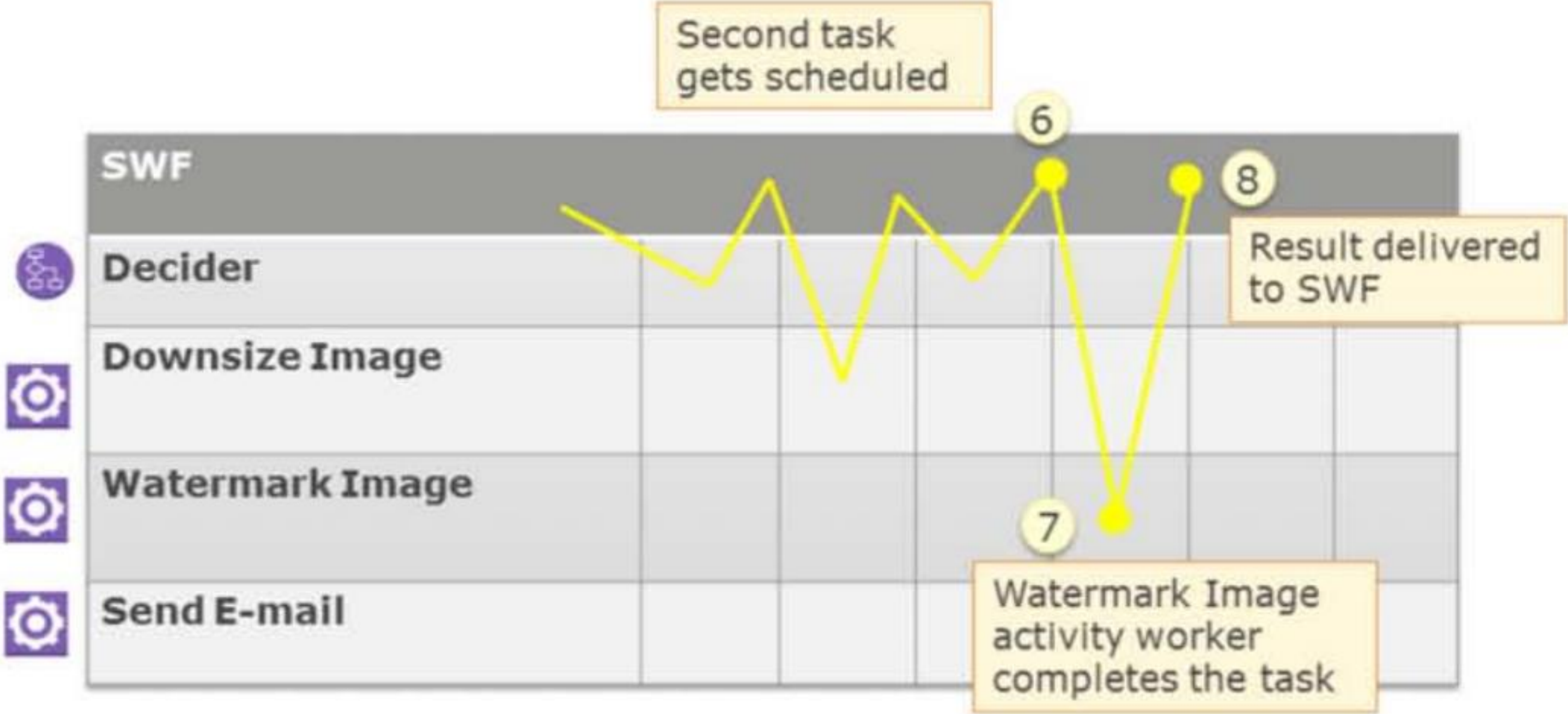


Downsizes the image and returns the results to SWF

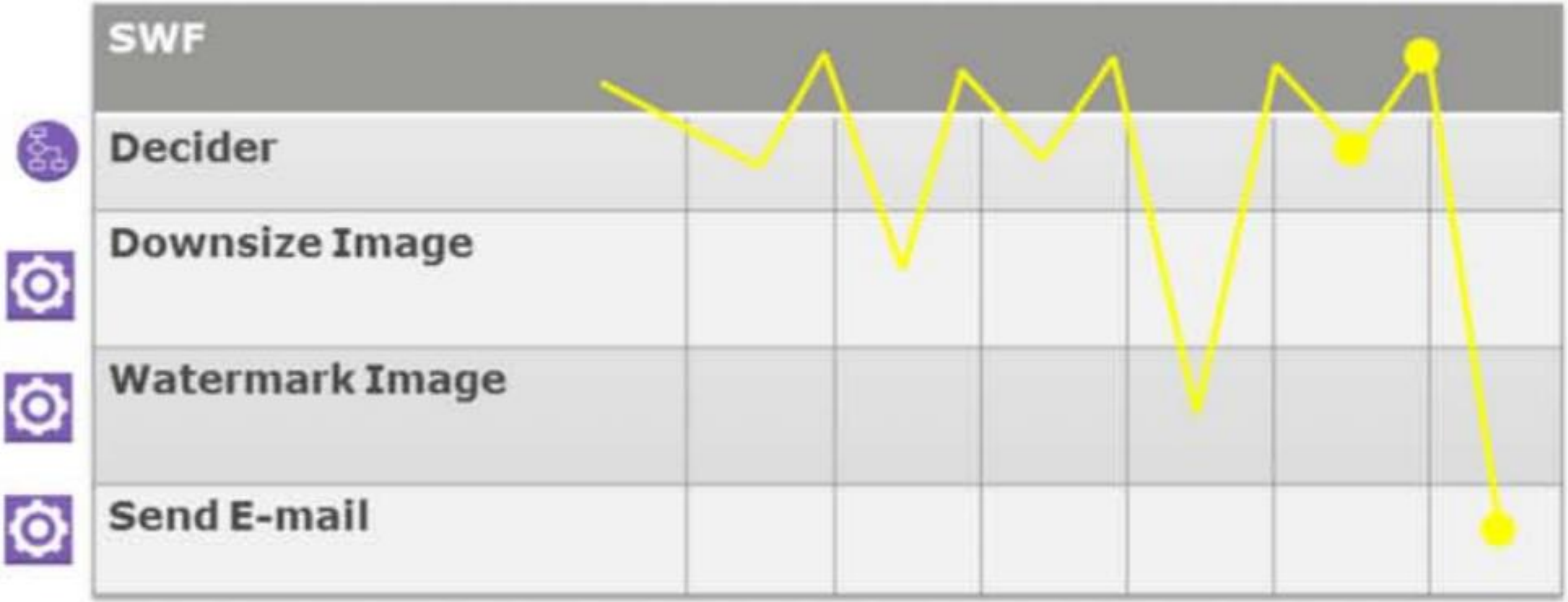
Result of ***DownsizeImage*** activity task is delivered to Decider

An activity worker, Downsize Image receives the task

Watermarking application example with SWF (3 of 4)


Second task gets scheduled
6
SWF
8
Result delivered to SWF
Decider
Downsize Image
Watermark Image
Send E-mail
7
Watermark Image activity worker completes the task

Watermarking application example with SWF (4 of 4)



SWF

Decider

Downsize Image

Watermark Image

Send E-mail

Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES)
  - Amazon CloudSearch

Amazon Simple Email Service (SES)

- Bulk and transactional email-sending service
- Eliminates the hassle of email server management, network configuration, and meeting rigorous Internet Service Provider (ISP) standards
- Provides a built-in feedback loop, which includes notifications of bounce backs, failed and successful delivery attempts, and spam complaints

Topics

- AWS application services
- Roles of application services in AWS architecture
  - Amazon Simple Queue Service (SQS)
  - Amazon Simple Notification Service (SNS)
  - Amazon Simple Workflow Service (SWF)
  - Amazon Simple Email Service (SES)
  - Amazon CloudSearch

Amazon CloudSearch

- Fully-managed search service
- Integrate fast and highly scalable search functionality into applications
- Scales automatically: with increases in searchable data or as query rate changes
- You can assign weights to selected fields for customized relevance ranking
- AWS manages hardware provisioning, data partitioning, and software patches
- Supports 33 languages and popular search features such as highlighting, autocomplete, and geospatial search

Module review

- List the five main AWS application services
- What role does SQS play in designing loosely-coupled systems?
- What subscriber types does SNS support?
- What are the two components of SWF?