

# Application Integration

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampor.co/clf-c01](http://www.exampor.co/clf-c01)



## What is Application Integration?

Application Integration is the process of letting two independent applications to communicate and work with each other, commonly facilitated by an intermediate system.



Cloud workloads encourage systems and services to be loosely coupled and so AWS has many service for the specific purpose of application integration.

The common systems or design patterns utilized for Application Integration generally are:

- Queueing
- Streaming
- Pub/Sub
- API Gateways
- State Machine
- Event Bus

# Queueing

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampor.co/clf-c01](http://www.exampor.co/clf-c01)

## What is a Messaging System?

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

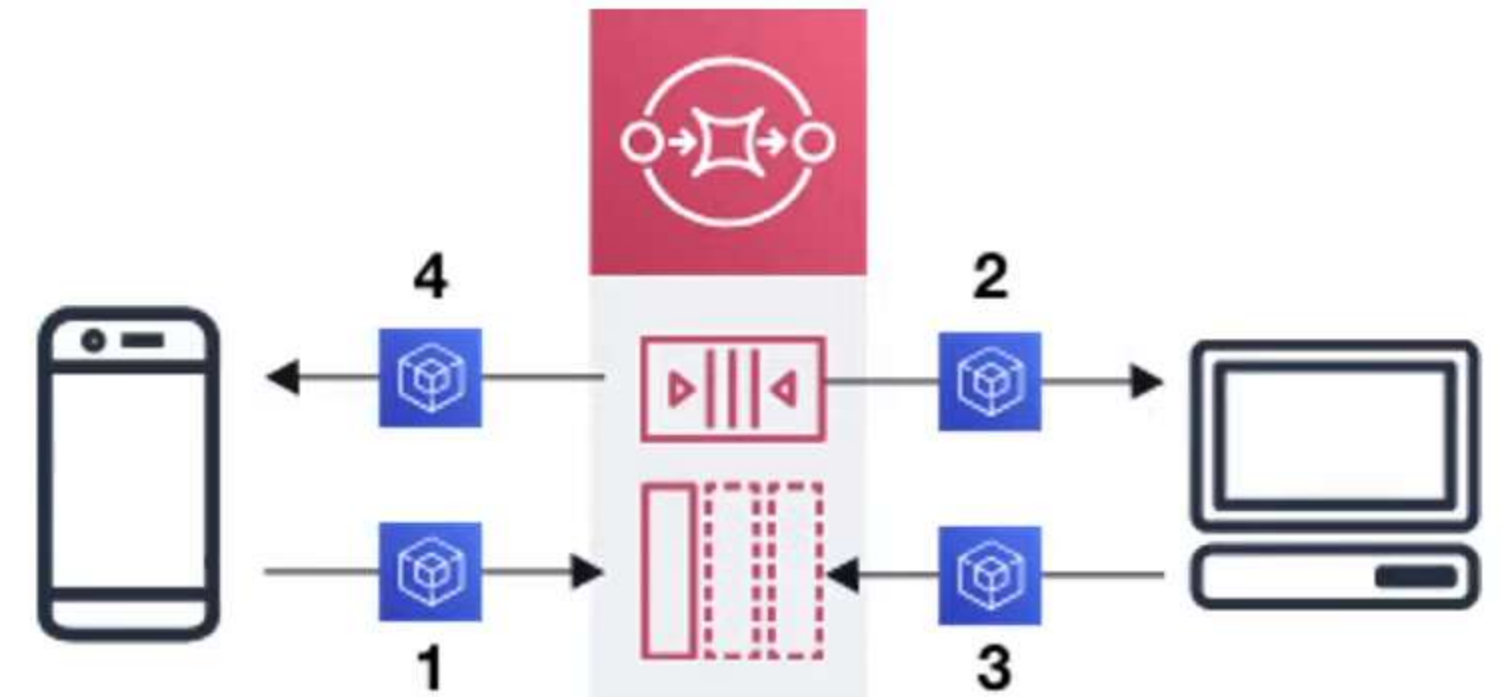
## What is a Queueing System?

A Queueing system is a messaging system that generally will delete messages once they are consumed.  
Simple communication. **Not Real-time**. Have to pull. Not reactive.



### Simple Queueing Service (SQS)

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



Use Case: You need to queue up transaction emails to be sent e.g. Signup, Reset Password.



# Streaming

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampopro.co/clf-c01](http://www.exampopro.co/clf-c01)

## What is streaming?

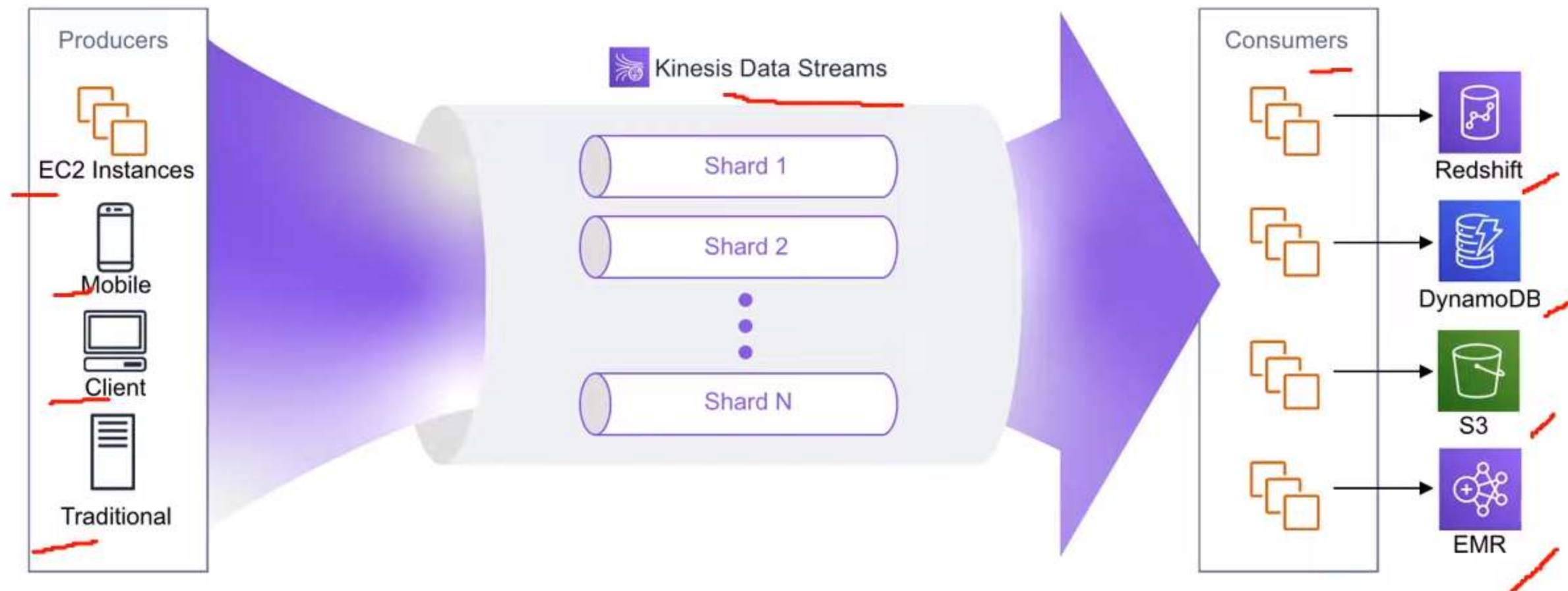
Multiple consumers can **react** to events (messages)

Events live in the stream for long periods of time, so complex operations can be applied. **Real-time**



## Amazon Kinesis

Amazon Kinesis is the AWS fully managed solution for collecting, processing, and analyzing streaming data in the cloud.



# Pub/Sub

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampor.co/clf-c01](http://www.exampor.co/clf-c01)

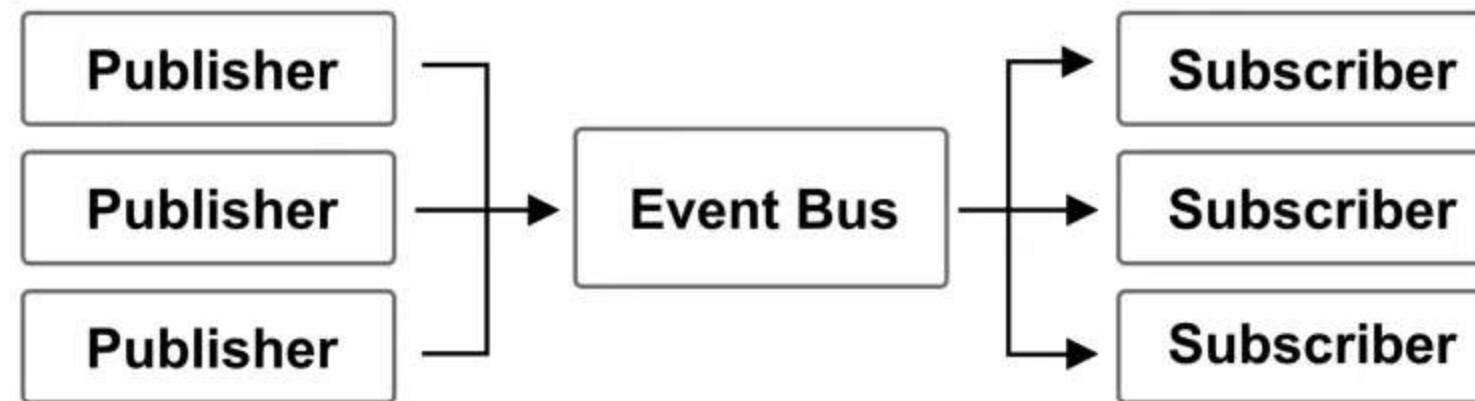
## What is Pub/Sub?

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.



Use Case: a real-time chat system. A web-hook system

- 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

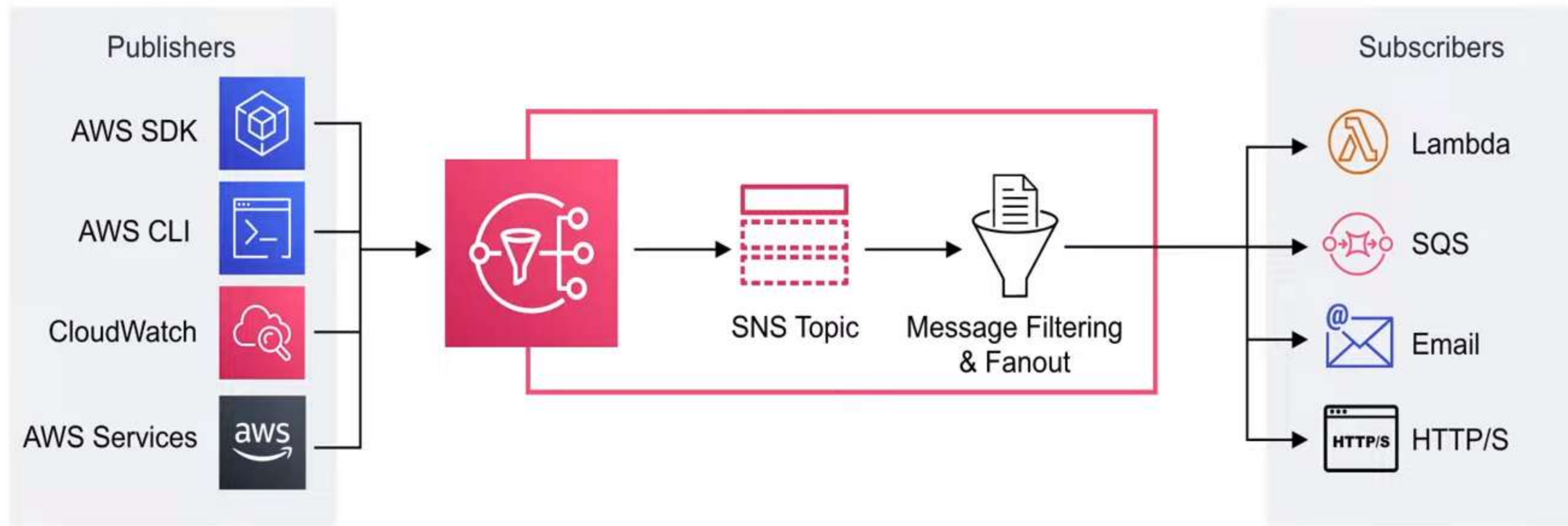


# Pub/Sub

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampopro.co/clf-c01](http://www.exampopro.co/clf-c01)



**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.



# API Gateway

Cheat sheets, Practice Exams and Flash cards  [www.examprompro.co/clf-c01](http://www.examprompro.co/clf-c01)

## What is an API Gateway?

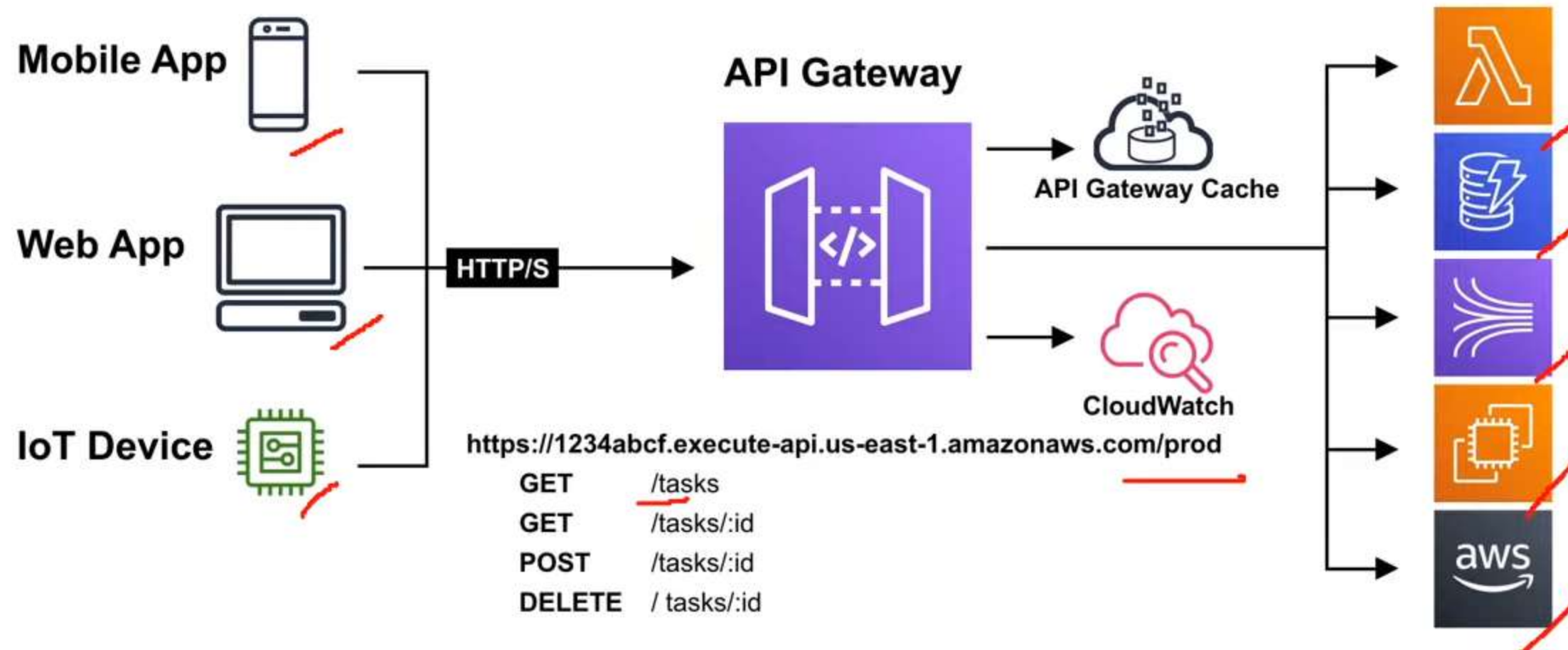
An API Gateway is a program that sits between a single-entry point and multiple backends.

API Gateway allows for throttling, logging, routing logic or formatting of the request and response



**Amazon API Gateway** is a solution for **creating secure APIs** in your cloud environment at **any scale**.

Create APIs that act as a front door for applications to access data, business logic, or functionality from back-end services.





# State Machines

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampopro.co/clf-c01](http://www.exampopro.co/clf-c01)

## What is a state machine?

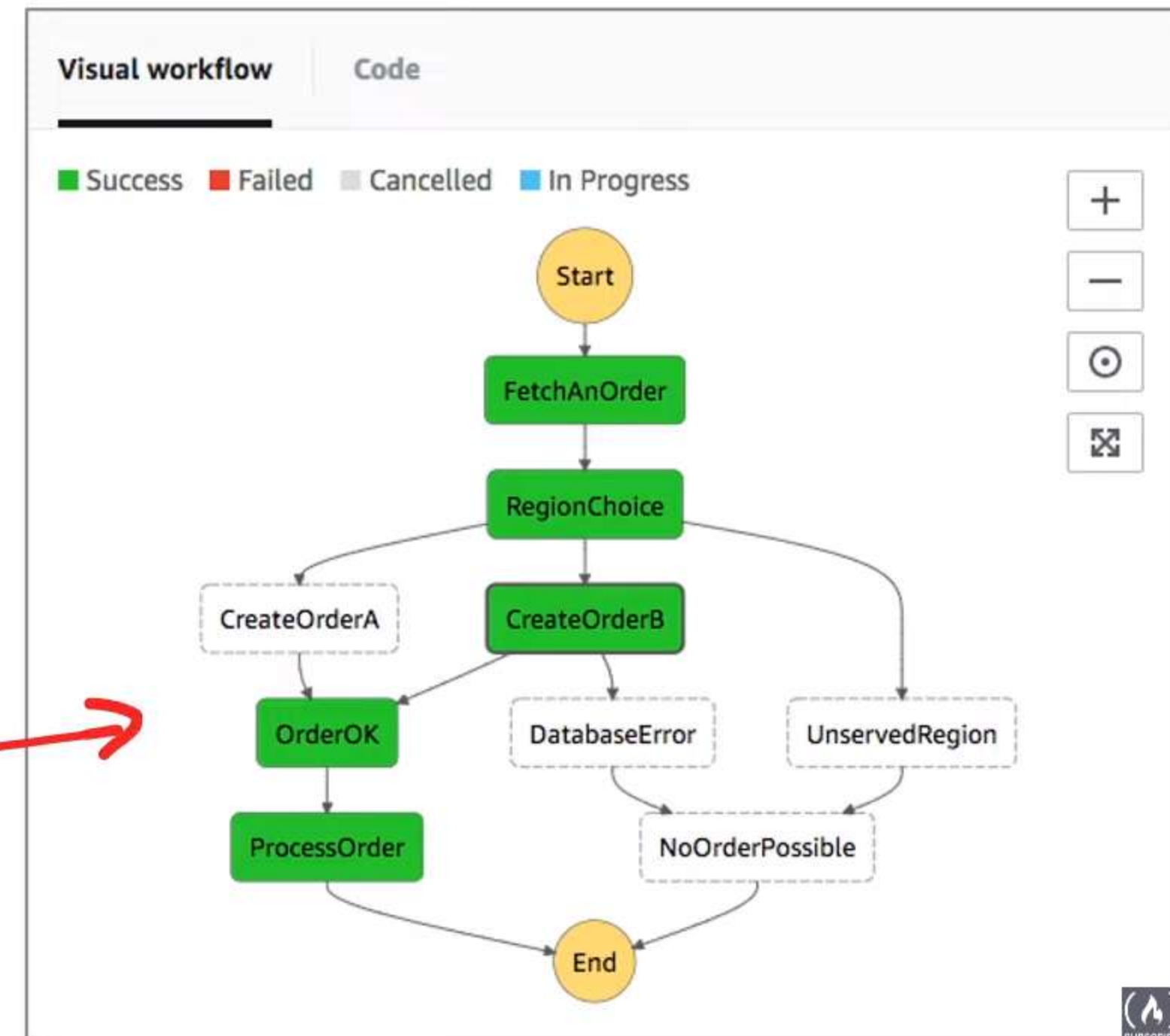
A state machine is an abstract model which decides how one state moves to another based on a series of conditions. **Think of a state machine like a flow chart.**



## What is AWS Step Functions?

- Coordinate multiple AWS Services into a serverless workflow
- A graphical console to visualize the components of your application as a series of steps.
- Automatically triggers and tracks each step, and retries when there are errors, so your application executes in order and as expected, every time
- logs the state of each step, so when things go wrong, you can diagnose and debug problems quickly

**Any one of these steps could be using an AWS Service**

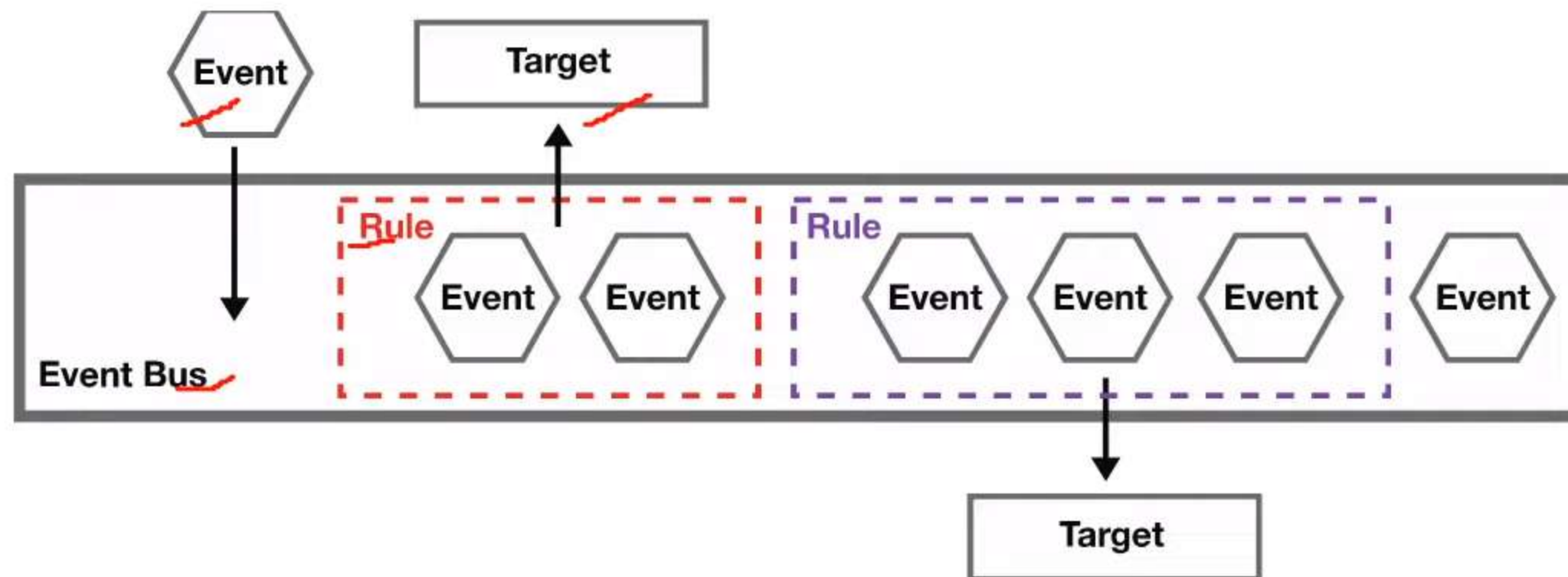


# Event Bus

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.examprompro.co/clf-c01](http://www.examprompro.co/clf-c01)

## What is an Event Bus?

An event bus **receives events** from a **source** and **routes events** to a **target** based on **rules**



**EventBridge** is a **serverless** event bus service that is used for application integration by **streaming real-time** data to your applications

**EventBridge** was *formerly* called **Amazon CloudWatch Events**.



# Amazon Event Bridge

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.exampopro.co/clf-c01](http://www.exampopro.co/clf-c01)

## Event Bus

Holds event data, define rules on an event bus to react to events.

**Default Event Bus** — An AWS account has a default event bus

**Custom Event Bus** — Scoped to multiple accounts or other AWS accounts

**SaaS Event Bus** — Scoped to with Third party SaaS Providers

## Rules

Determines what events to capture and pass to targets. (100 Rules per bus)

## Targets

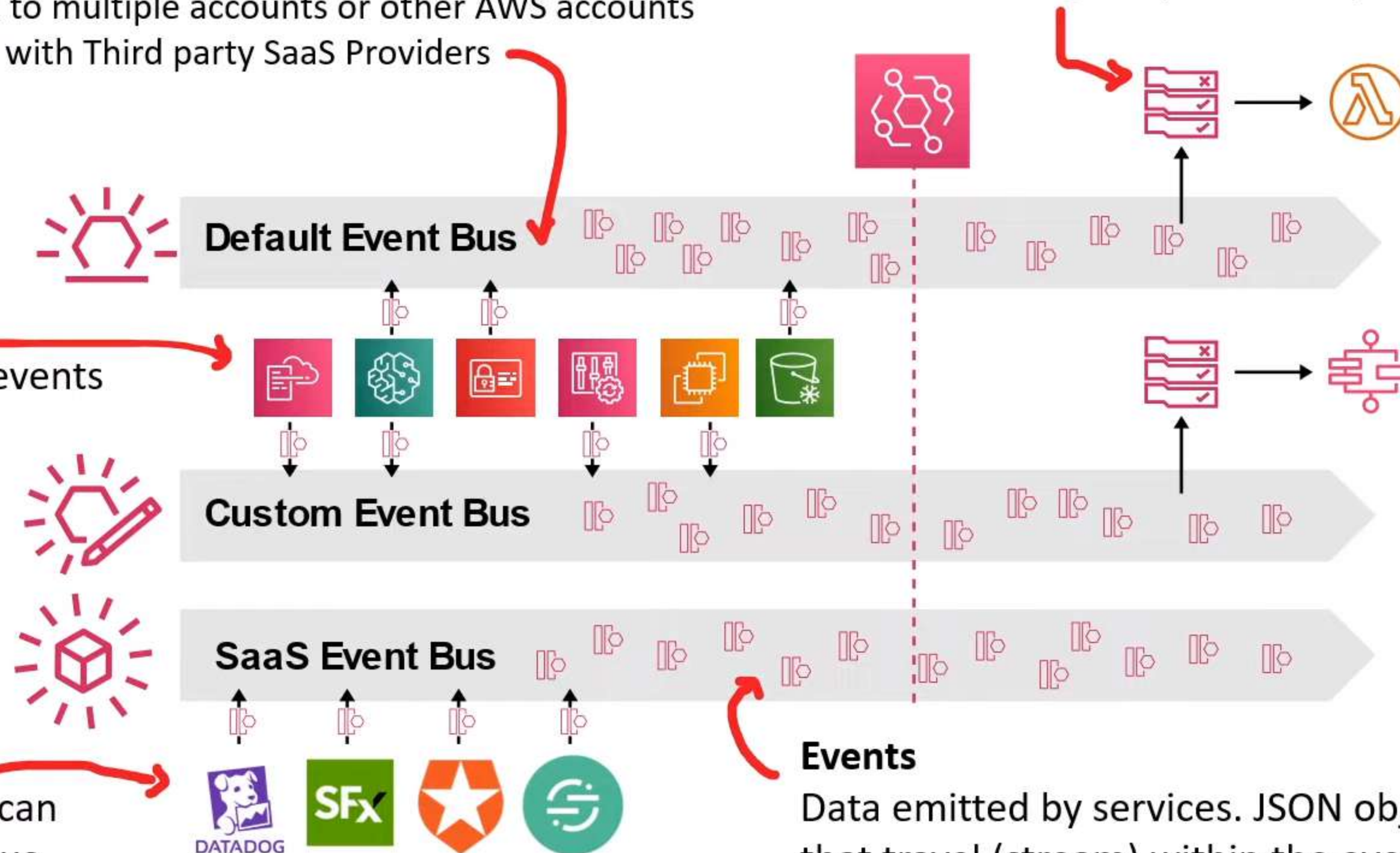
AWS Services that consume events  
(5 targets per rule)

## Producers

AWS Services that emit events

## Partner Sources

Are third-party apps that can emit events to an event bus



## Events

Data emitted by services. JSON objects that travel (stream) within the event bus.



# Application Integration Services

Cheat sheets, Practice Exams and Flash cards 🖱️ [www.examprompro.co/clf-c01](http://www.examprompro.co/clf-c01)



**Simple Notification Service (SNS)** - a **pub-sub messaging system**. Sends notifications via various formats such as Plain-text **Email**, HTTP/s (**webhooks**) SMS (**text messages**), **SQS** and **Lambda**. Push messages which then are sent to subscribers



**Simple Queue Service (SQS)** is a **queueing messaging service**. Send events to a queue. Other applications pull the queue for messages. Commonly used for background jobs.



**Step Functions** is a **state machine service**. It coordinate multiple AWS services into serverless workflows. Easily share data among Lambdas. Have a group of lambdas wait for each other. Create logical steps. Also works with Fargate Tasks.



**EventBridge (CloudWatch Events)** is a **serverless event bus** that makes it easy to connect applications together from your own application, third-party services and AWS services.



**Kinesis** is a **real-time streaming data service**. Create **Producers** which send data to a stream. **Multiple Consumers** can consume data within a stream. Use for real-time analytics, click streams, ingesting data from a fleet of IOT Devices



**Amazon MQ** is a **managed message broker service** that uses **Apache ActiveMQ** 



**Managed Kafka Service (MSK)** a **fully managed Apache Kafka service**. Kafka is an open-source platform for building real-time streaming data pipelines and applications. Similar to Kinesis but more robust



**API Gateway** is a fully-managed service for developers to create, publish, maintain, monitor, and secure APIs. You can create API endpoints and route them to AWS services.



**AppSync** is a **fully managed GraphQL service**. GraphQL is an open-source agnostic query adaptor that allows you to query data from many different data sources.