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

API311-R

Building next-gen applications with event-driven architectures

Eric Johnson

Principal Developer Advocate
AWS

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

Who am I?

- Principal Developer Advocate – AWS
- Father of 5
- Brand-new grandpa
- Drummer
- Foodie



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

Agenda

Enterprise integration patterns

Event-driven architecture

Handling event duplication using idempotency

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

Enterprise integration patterns

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

Coupling – Integration's magic word



Coupling is a measure of independent variability between connected systems

Decoupling has a cost, both at design and runtime

Coupling isn't binary

Coupling isn't one-dimensional

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

Source: EnterpriseIntegrationPatterns.com

The many facets of coupling

- Technology dependency: Java vs. C++
- Location dependency: IP addresses, DNS
- Data format dependency: Binary, XML, JSON, ProtoBuf, Avro
- Data type dependency: int16, int32, string, UTF-8, null, empty
- Semantic dependency: Name, middle name, ZIP
- Temporal dependency: sync, async
- Interaction style dependency: messaging, RPC, query-style (GraphQL)
- Conversation dependency: pagination, caching, retries

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

Source: EnterpriseIntegrationPatterns.com

“The appropriate level of coupling depends on the level of control you have over the endpoints.”

Gregor Hohpe

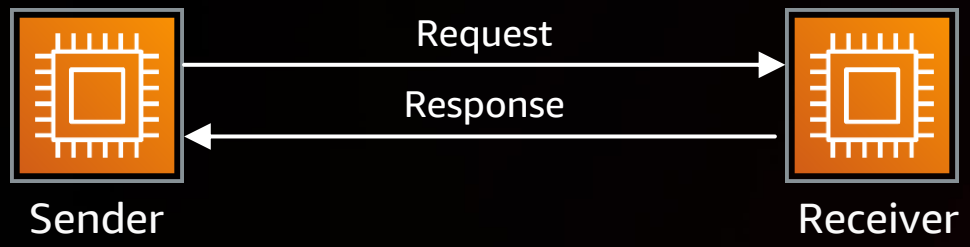
Co-author of “Enterprise Integration Patterns”

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

Synchronous request-response model

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

Synchronous request-response model

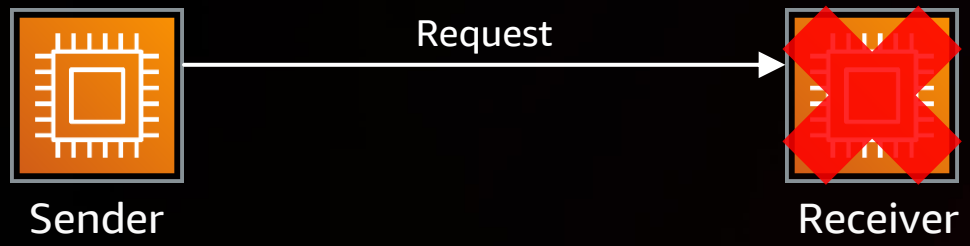


Advantages

- Low latency
- Simple
- Fail fast

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

Synchronous request-response model



Advantages

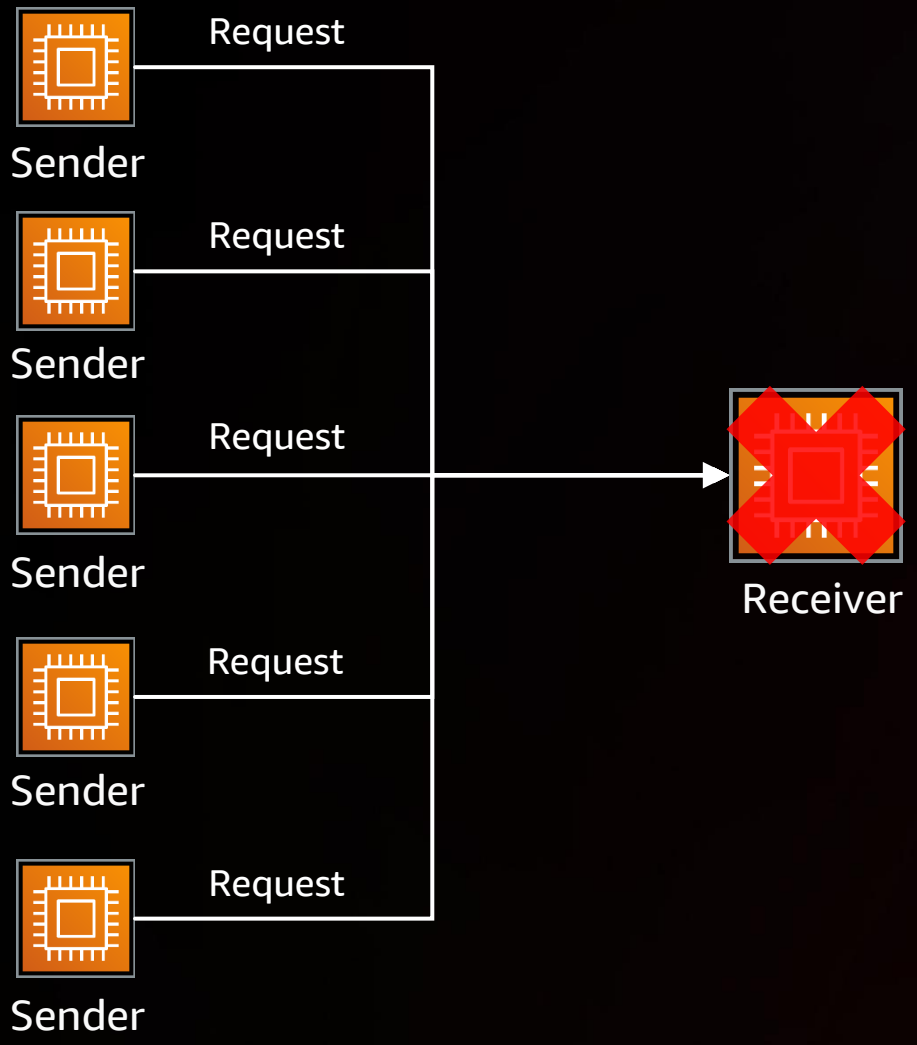
- Low latency
- Simple
- Fail fast

Disadvantages

- Receiver failure

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

Synchronous request-response model



Advantages

- Low latency
- Simple
- Fail fast

Disadvantages

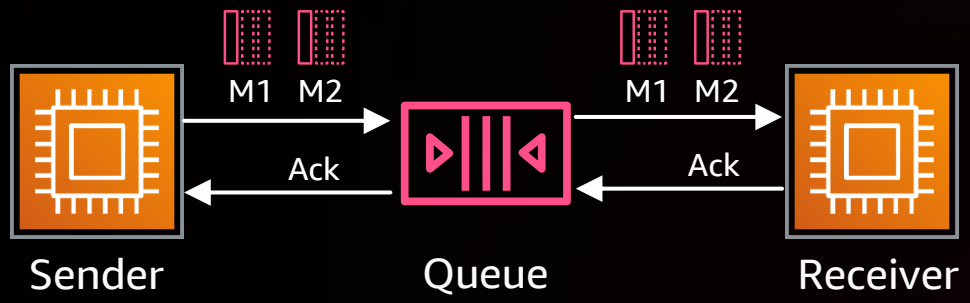
- Receiver failure
- Receiver throttled

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

Asynchronous point-to-point model (queue)

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

Asynchronous point-to-point model (queue)

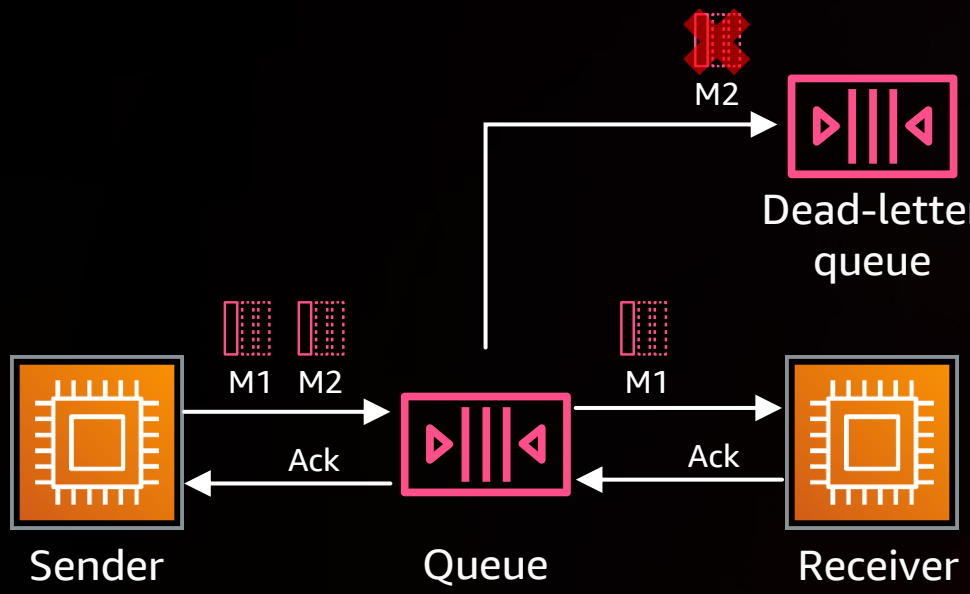


Advantages

- Decreases **temporal coupling**
- Resilient to receiver failure
- Receiver controls consumption rate

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

Asynchronous point-to-point model (queue)

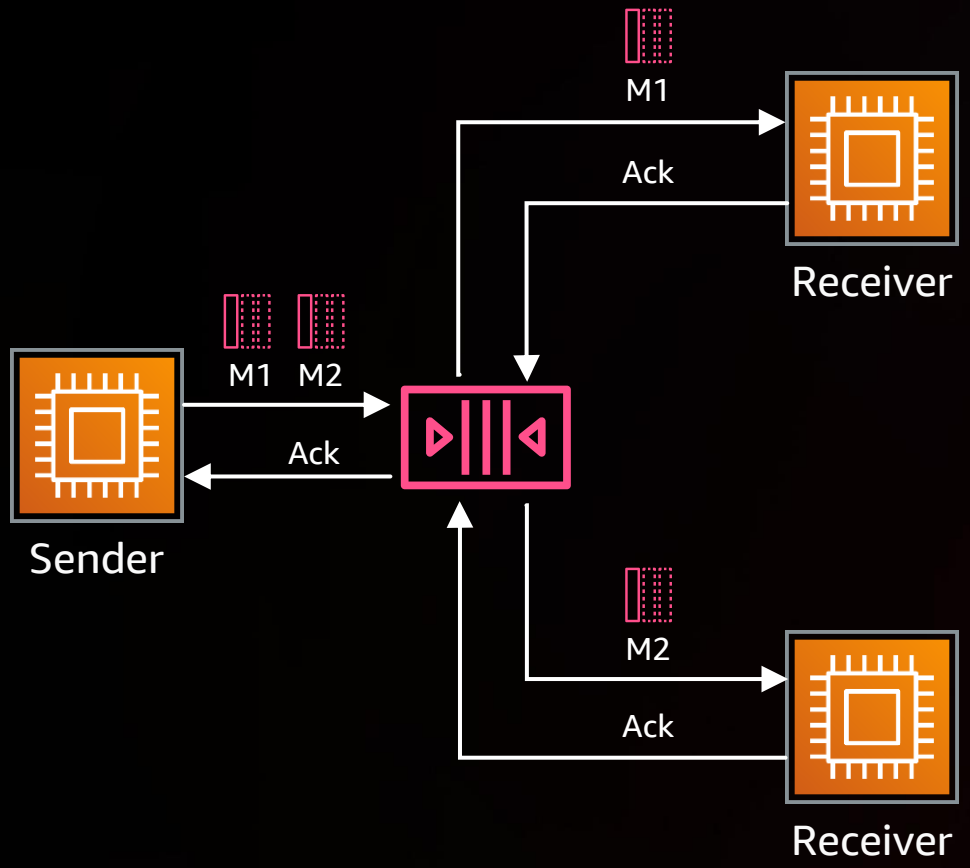


Advantages

- …
- Dead-letter queue (DLQ) for errors

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

Asynchronous point-to-point model (queue)

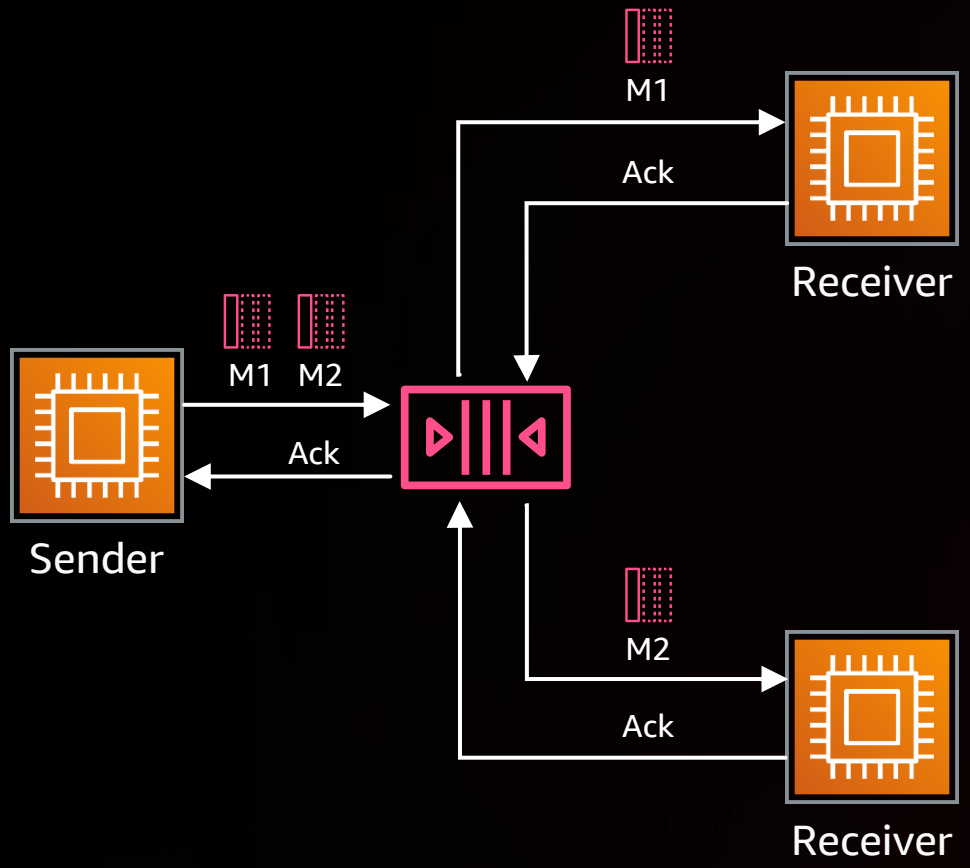


Advantages

- …
- Only one receiver can consume each message

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

Asynchronous point-to-point model (queue)



Advantages

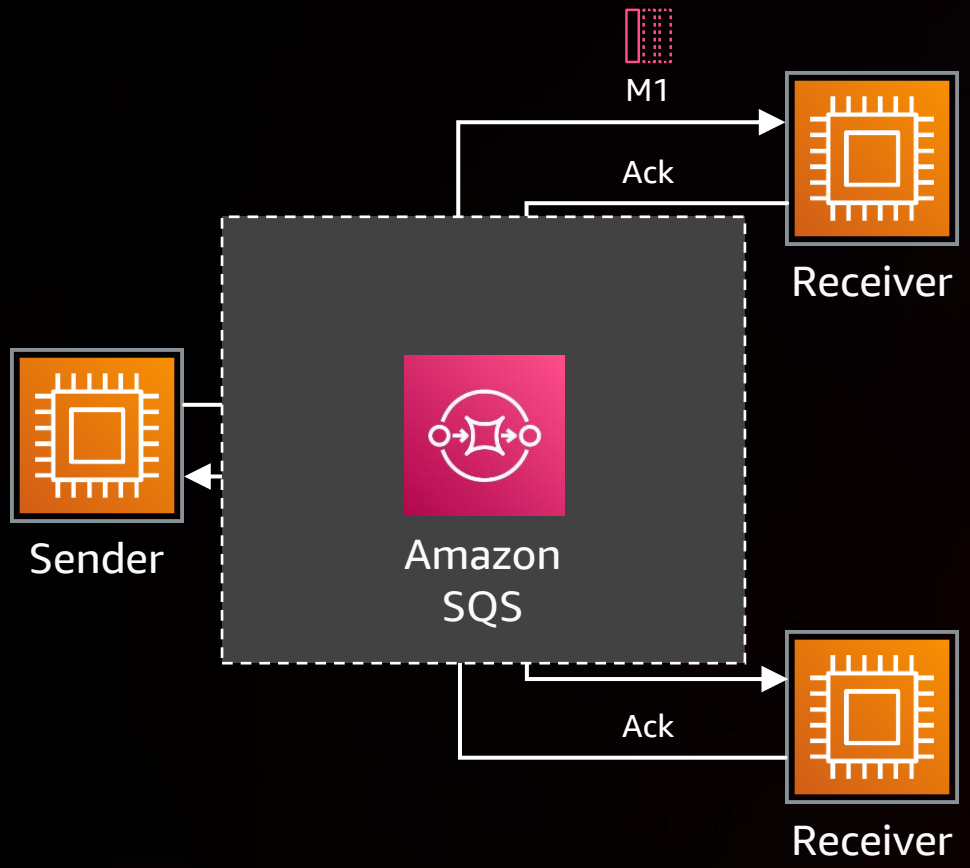
- …
- Only one receiver can consume each message

Disadvantages

- Response correlation
- Backlog recovery time
- Fairness in multi-tenant systems

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

Asynchronous point-to-point model (queue)



Amazon Simple Queue Service (Amazon SQS)

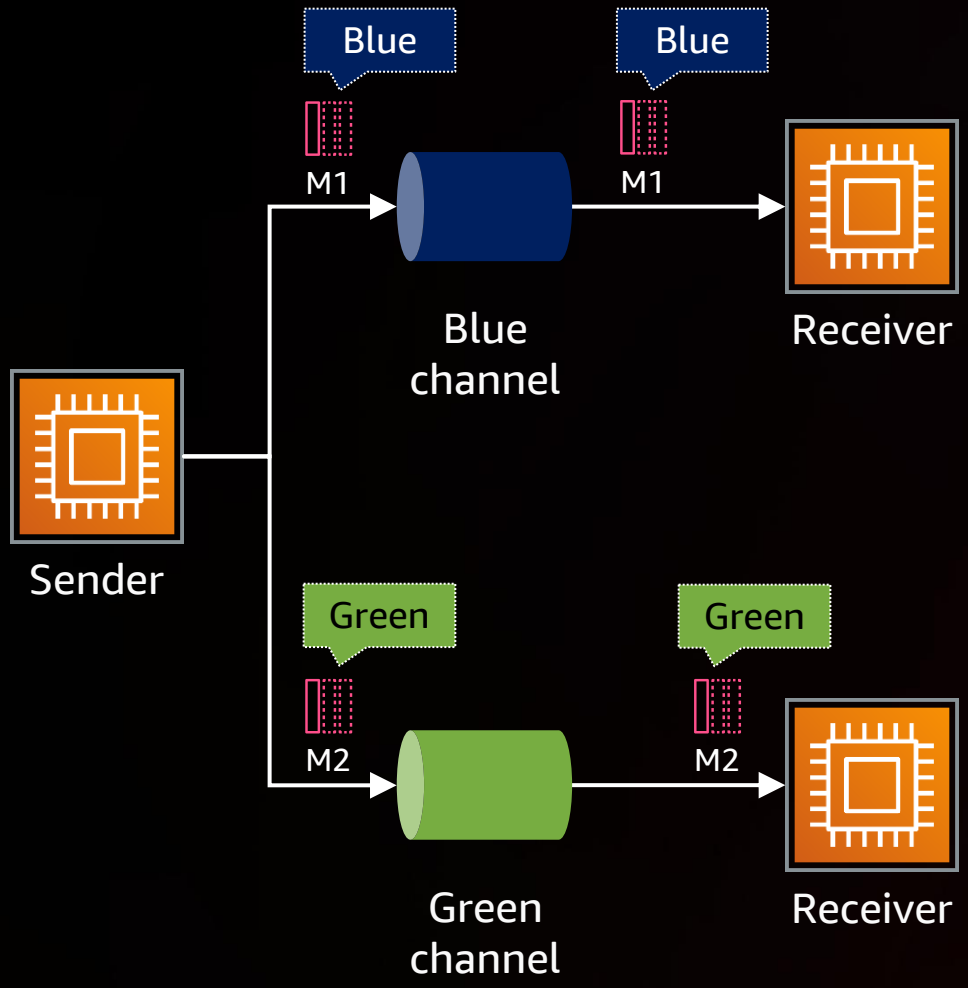
- Fully managed message queue
- Scales almost infinitely
- Simple, easy-to-use API
- DLQ support
- Standard and FIFO options

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

Asynchronous point-to-point model (router)

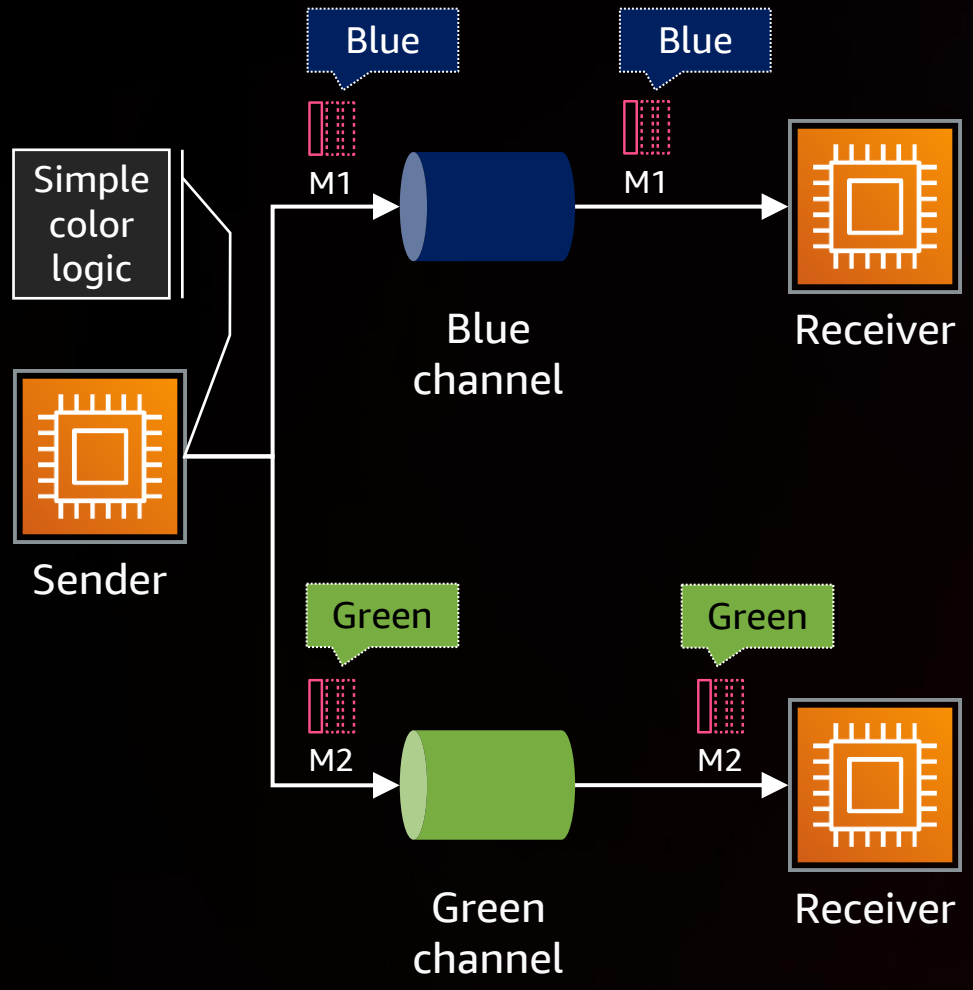
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Asynchronous point-to-point model (router)



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

Asynchronous point-to-point model (router)

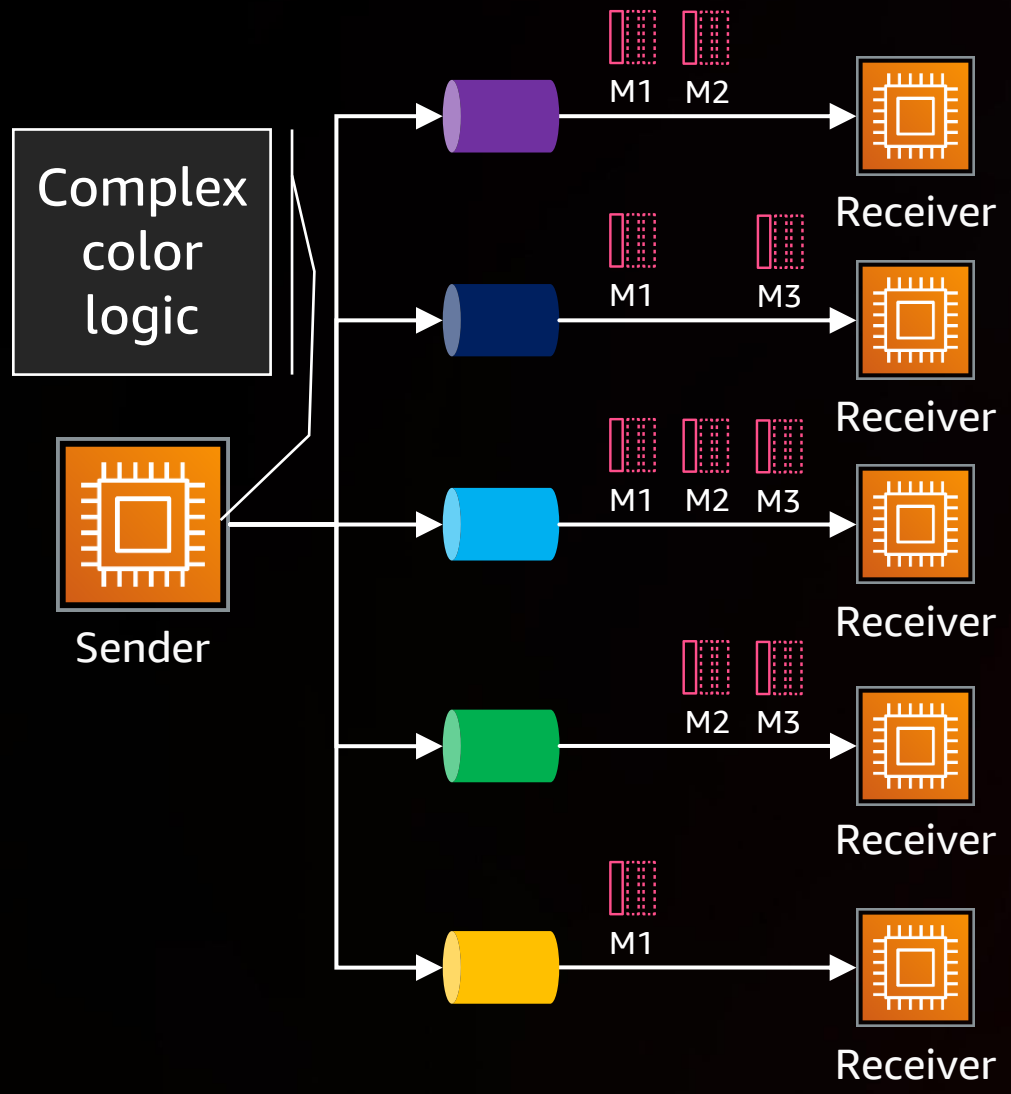


Disadvantages

- Increases **location coupling**
- Sender maintains routing logic

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

Asynchronous point-to-point model (router)



Disadvantages

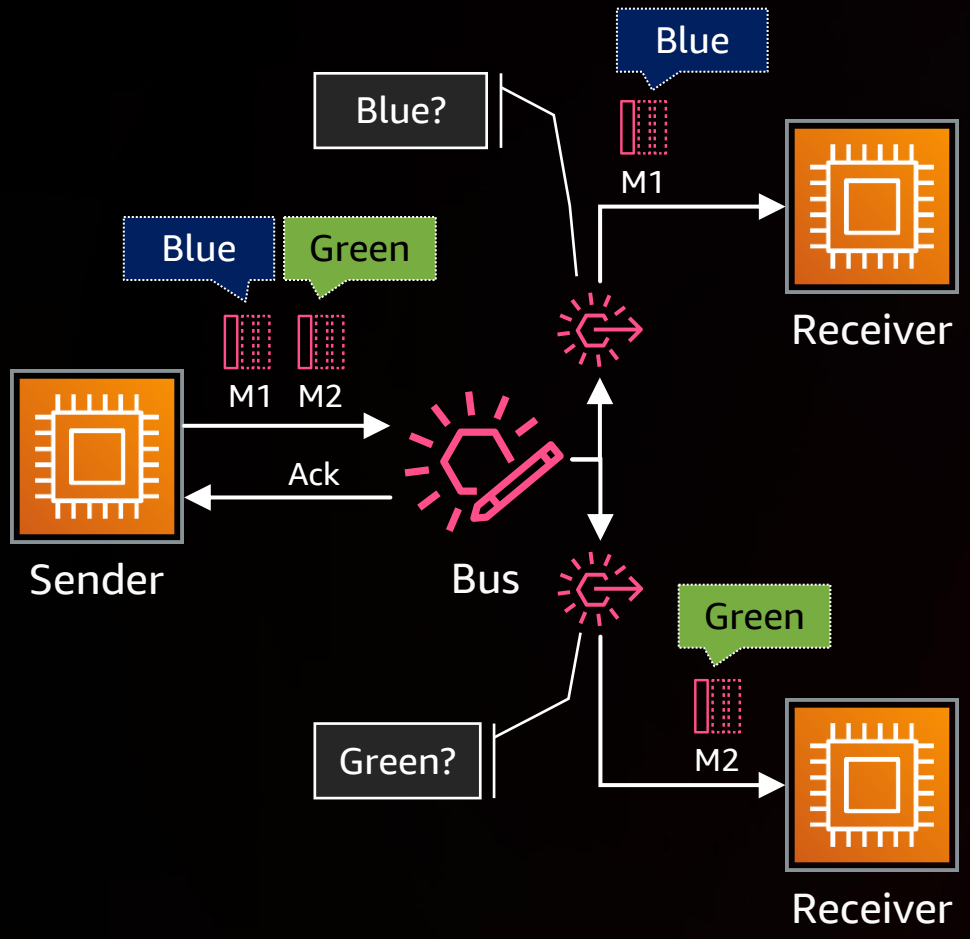
- Increases **location coupling**
- Sender maintains routing logic
- Sender complexity increases with time

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

Asynchronous message-router model (bus)

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

Asynchronous message-router (bus)

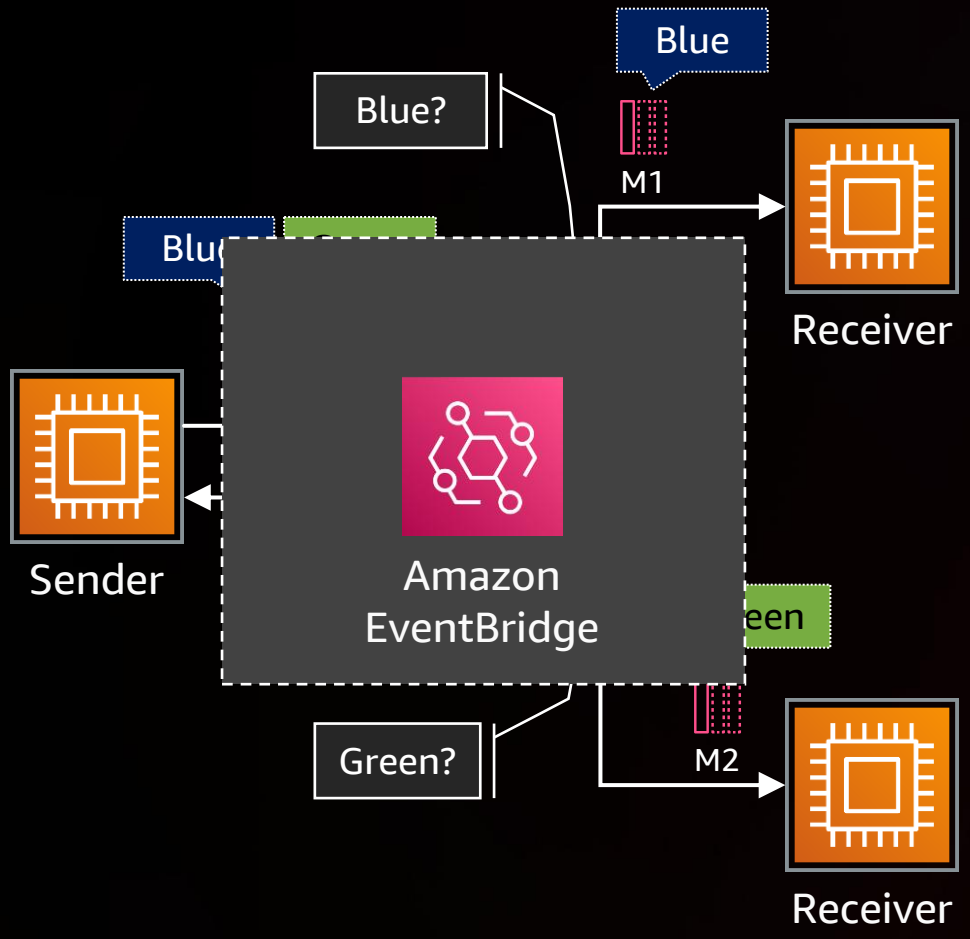


Advantages

- Reduces **location coupling**
- Efficient for senders and receivers

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

Asynchronous message-router (bus)



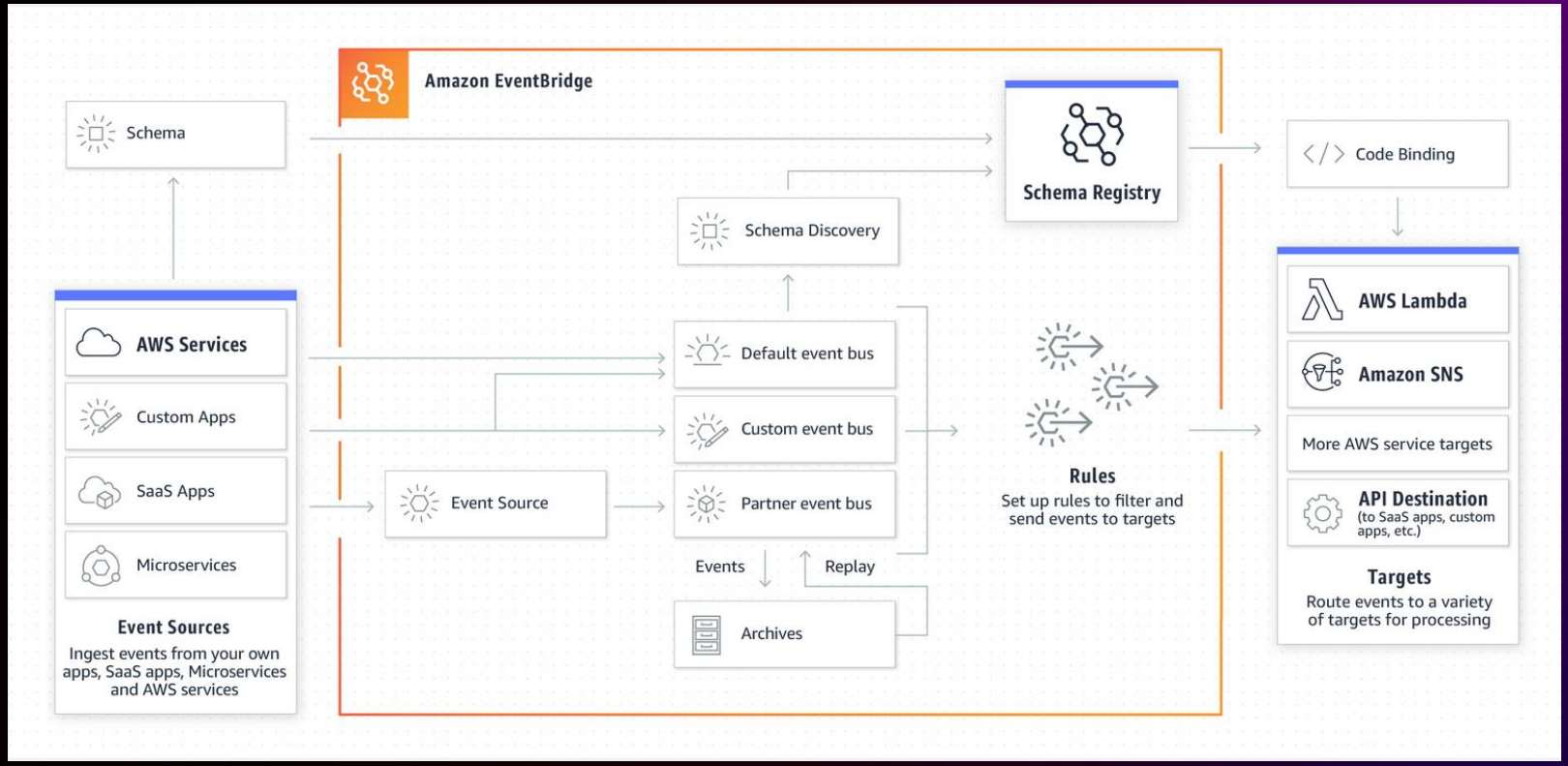
Advantages

- Reduces **location coupling**
- Efficient for senders and receivers

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

EventBridge

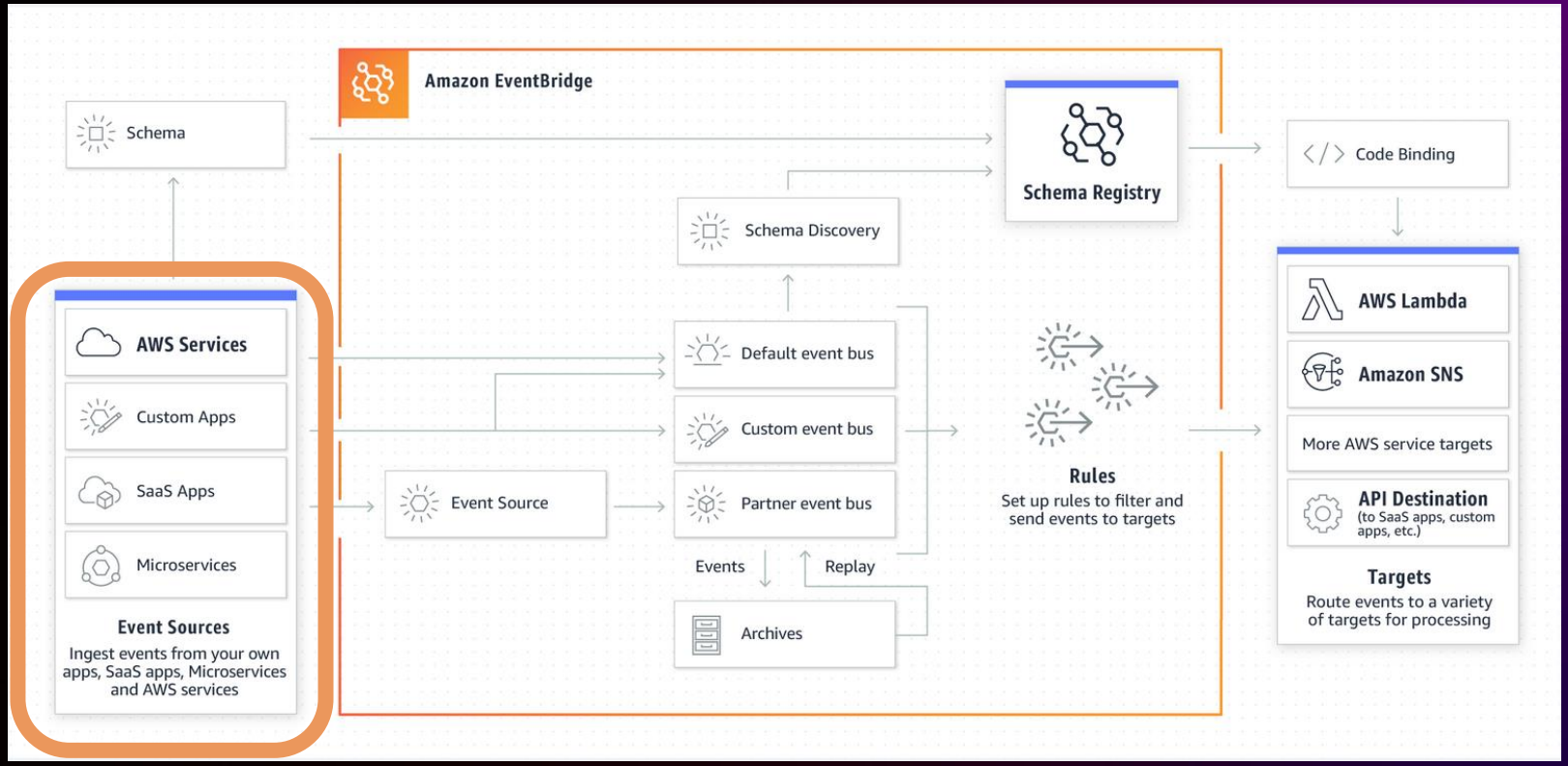
EventBridge is a simple, flexible, fully managed, pay-as-you-go **event bus service** that makes it easy to ingest and process data from **AWS services, your own applications**, and **SaaS applications**



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

EventBridge

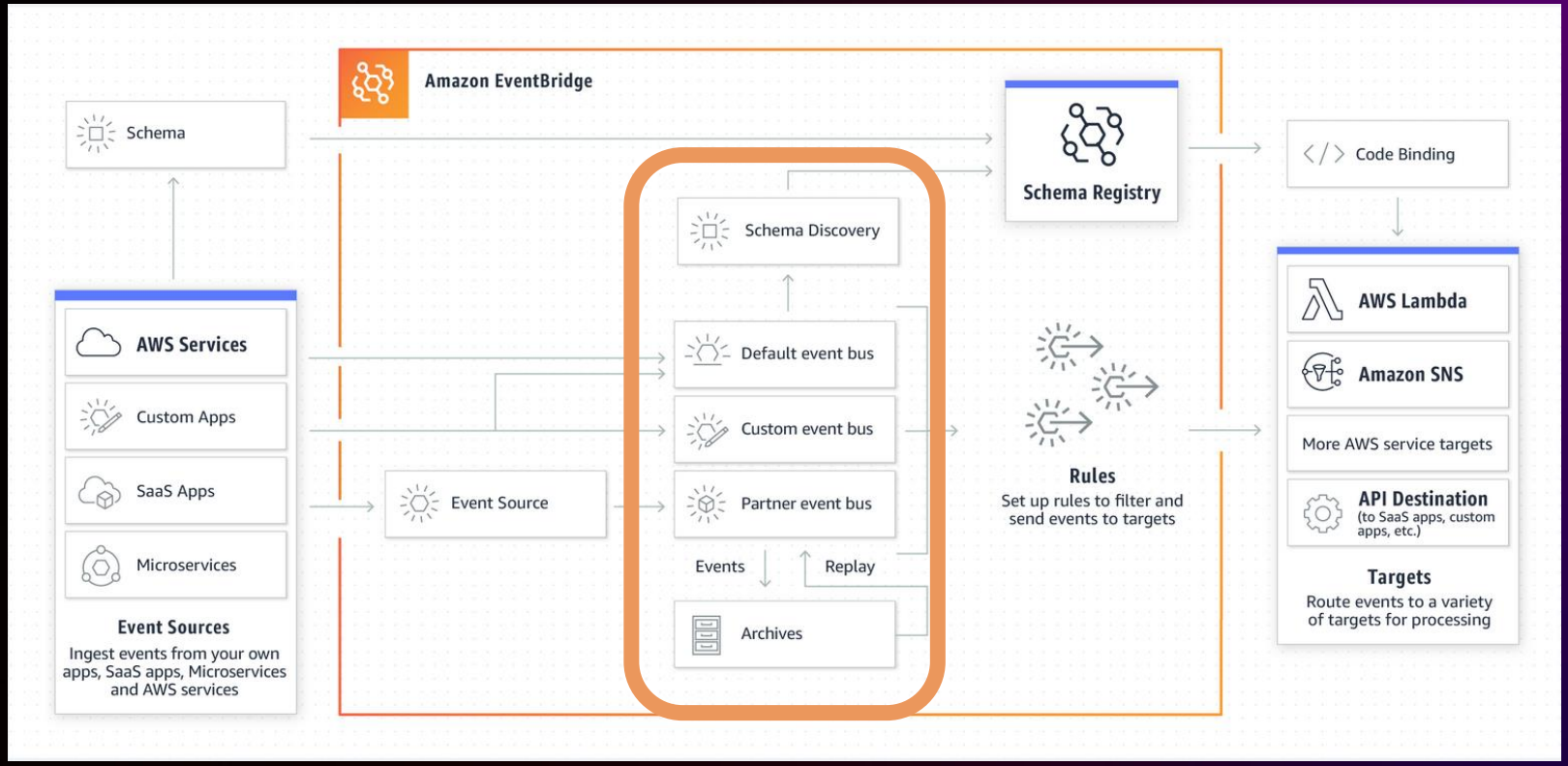
EventBridge is a simple, flexible, fully managed, pay-as-you-go **event bus service** that makes it easy to ingest and process data from **AWS services, your own applications**, and **SaaS applications**



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

EventBridge

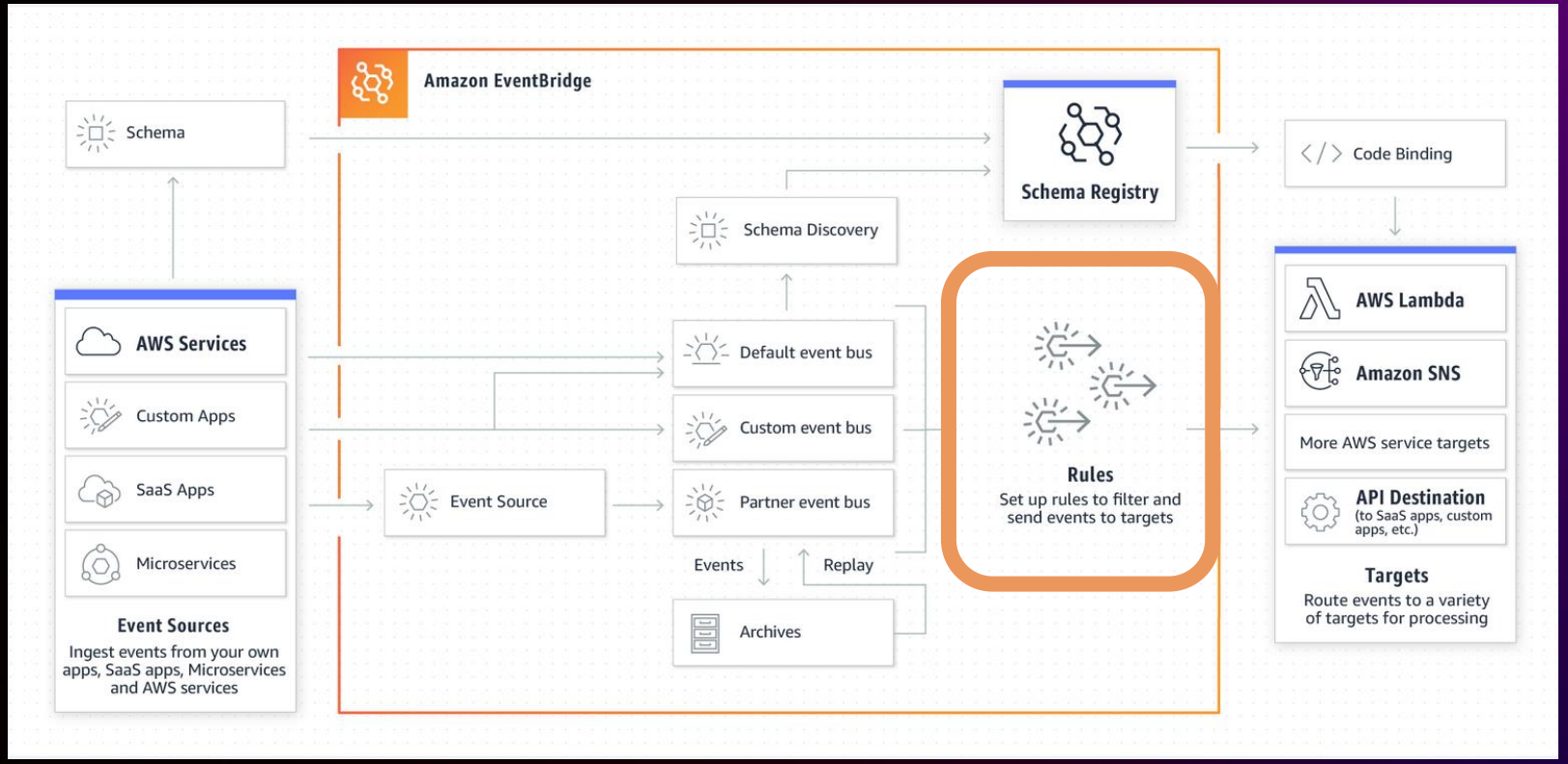
EventBridge is a simple, flexible, fully managed, pay-as-you-go **event bus service** that makes it easy to ingest and process data from **AWS services, your own applications**, and **SaaS applications**



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

EventBridge

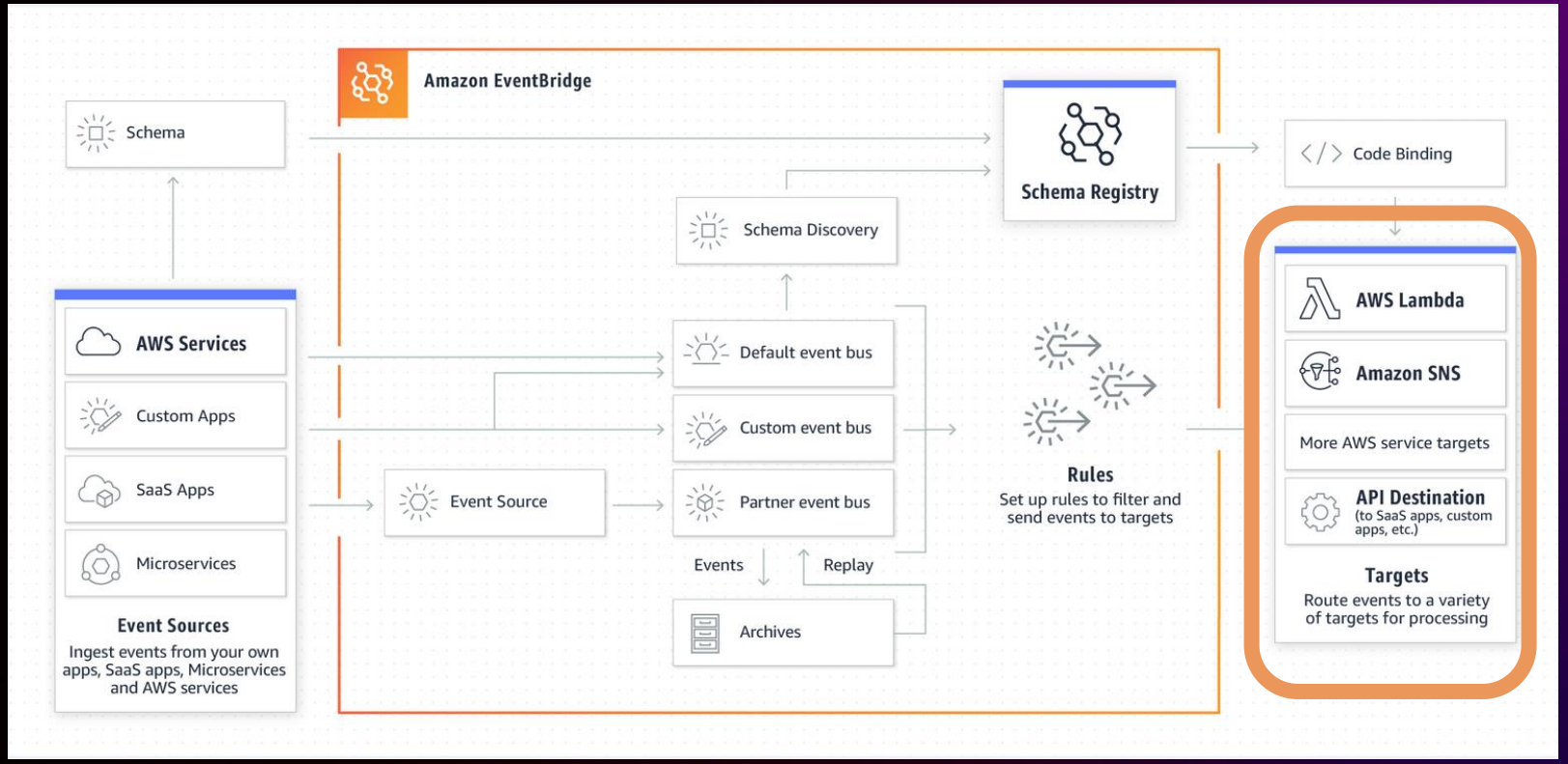
EventBridge is a simple, flexible, fully managed, pay-as-you-go **event bus service** that makes it easy to ingest and process data from **AWS services, your own applications**, and **SaaS applications**



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

EventBridge

EventBridge is a simple, flexible, fully managed, pay-as-you-go **event bus service** that makes it easy to ingest and process data from **AWS services, your own applications**, and **SaaS applications**



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

EventBridge content-based routing rules

EventBridge example event

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
    },
    "data": {
      "order-id": "1073459984",
      "created-at": "2021-11-26T16:05:09-04:00",
      "price": 24.62
      "currency": "AU",
    }
  }
}
```

aws

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

EventBridge content-based routing rules

EventBridge example event

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
    },
    "data": {
      "order-id": "1073459984",
      "created-at": "2021-11-26T16:05:09-04:00",
      "price": 24.62
      "currency": "AU",
    }
  }
}
```

EventBridge example rule

```
{
  "detail": {
    "data":  {
      "currency": ["AU", "NZ"]
    }
  }
}
```

aws

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

EventBridge content-based routing rules

EventBridge example event

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
    },
    "data": {
      "order-id": "1073459984",
      "created-at": "2021-11-26T16:05:09-04:00",
      "price": 24.62
      "currency": "AU",
    }
  }
}
```

EventBridge example rule

```
{
  "detail": {
    "data":  {
      "currency": ["AU", "NZ"]
    }
  }
}
```

aws

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

EventBridge content-based routing rules

EventBridge example event

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
    },
    "data": {
      "order-id": "1073459984",
      "created-at": "2021-11-26T16:05:09-04:00",
      "price": 24.62
      "currency": "AU",
    }
  }
}
```

EventBridge example rule

```
{
  "detail": {
    "data":  {
      "currency": ["AU", "NZ"]
    }
  }
}
```



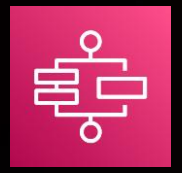
Match!

aws

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

Target 20+ AWS services and API destinations


AWS Lambda


AWS Step Functions


Amazon Kinesis Data Streams


Amazon CloudWatch


AWS CodePipeline


Amazon ECS


Amazon SNS


Amazon Kinesis Data Firehose


AWS Systems Manager


AWS CodeBuild


Amazon EC2

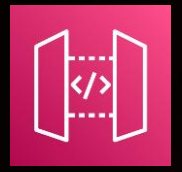

Amazon SQS


AWS Glue


Incident Manager, a capability of AWS Systems Manager


Amazon SageMaker


AWS Batch


Amazon API Gateway


Amazon Redshift


Amazon Inspector


API destinations

aws

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

Event-driven architecture

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

Properties of events

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "c1b95b88"
    },
    "data": {
      "order-id": "1073459984"
    }
}
```

- Events are signals that a system's state has changed
- Events occur in the past (e.g., OrderCreated)
- Events cannot be changed (immutable)
- Decrease coupling by restricting information to key data

aws

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

Sparse events vs. full state descriptions

Order 123 was created at 10:47 AM by customer 456



Sparse events

Order 123 was created at 10:47 AM by customer 456. The current status is Open, the total was $237.51, …



Full state description

aws

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

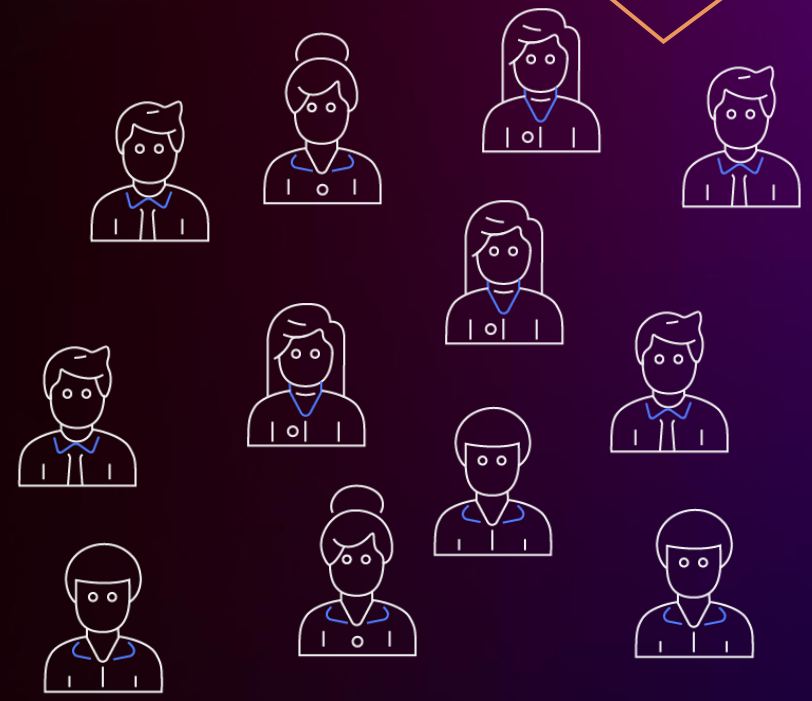
Considerations with sparse events

Order 123 was created at 10:47 AM by customer 456



Events

What are the details for order 123?



aws

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

Considerations with full state descriptions

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "c1b95b88"
    },
    "data": {
      "order-id": "1073459984",
      "status": "Open",
      "total": "237.51"
    }
}
```

- Event schemas should be backward compatible

aws

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

Considerations with full state descriptions

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "c1b95b88"
    },
    "data": {
      "order-id": "1073459984",
      "status": "Open",
      "total": "237.51"
    }
}
```

- Event schemas should be backward compatible
- Cost to calculate values can increase over time



aws

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

Considerations with full state descriptions

```
{
  "source": "com.orders",
  "detail-type": "OrderCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "c1b95b88"
    },
    "data": {
      "order-id": "1073459984",
      "status": "Open",
      "total": "237.51"
    }
}
```

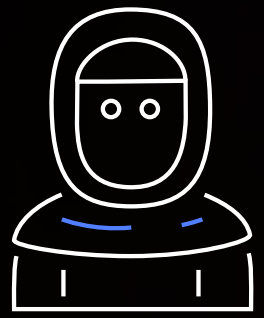
- Event schemas should be backward compatible
- Cost to calculate values can increase over time



aws

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

Choreograph events between domains using subscriptions



Retail



Fulfillment

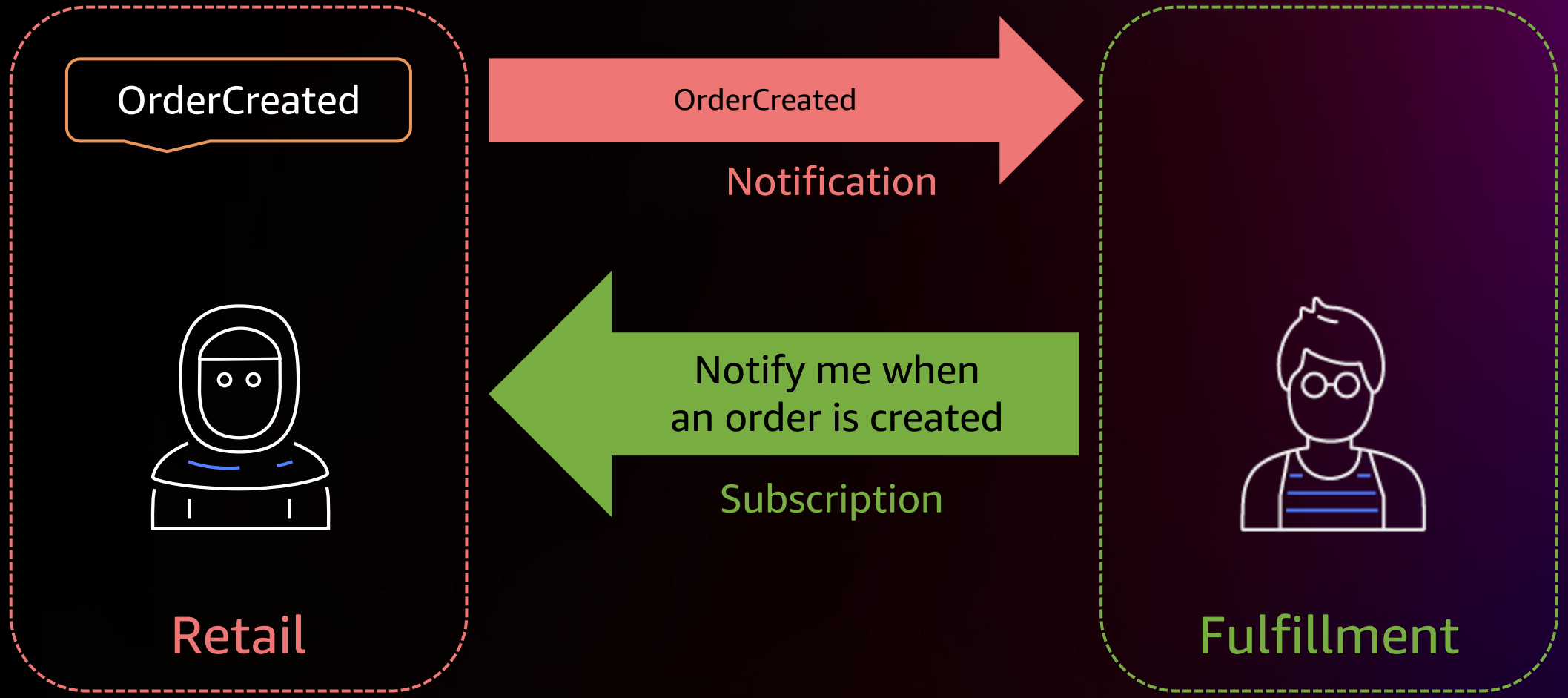
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Choreograph events between domains using subscriptions



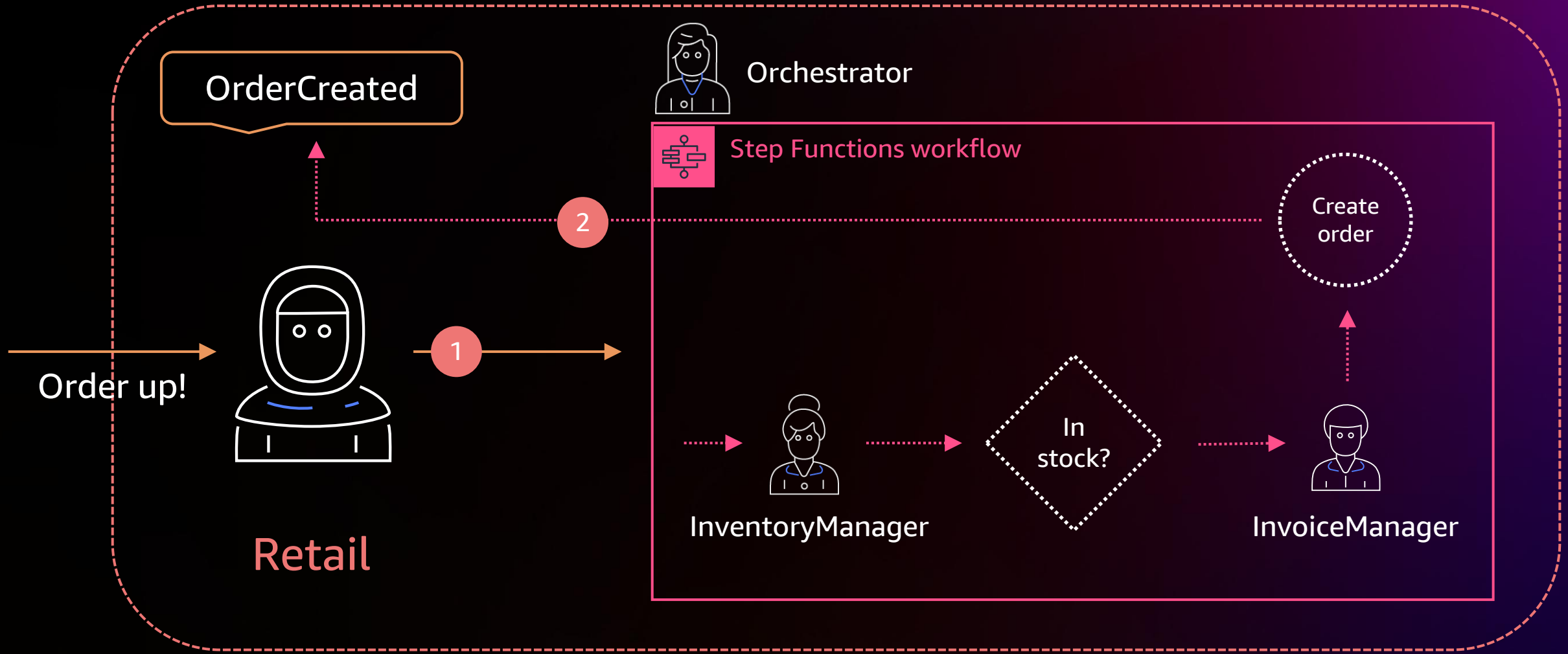
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Choreograph events between domains using subscriptions



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

Orchestrate a business process within a domain, resulting in a published event



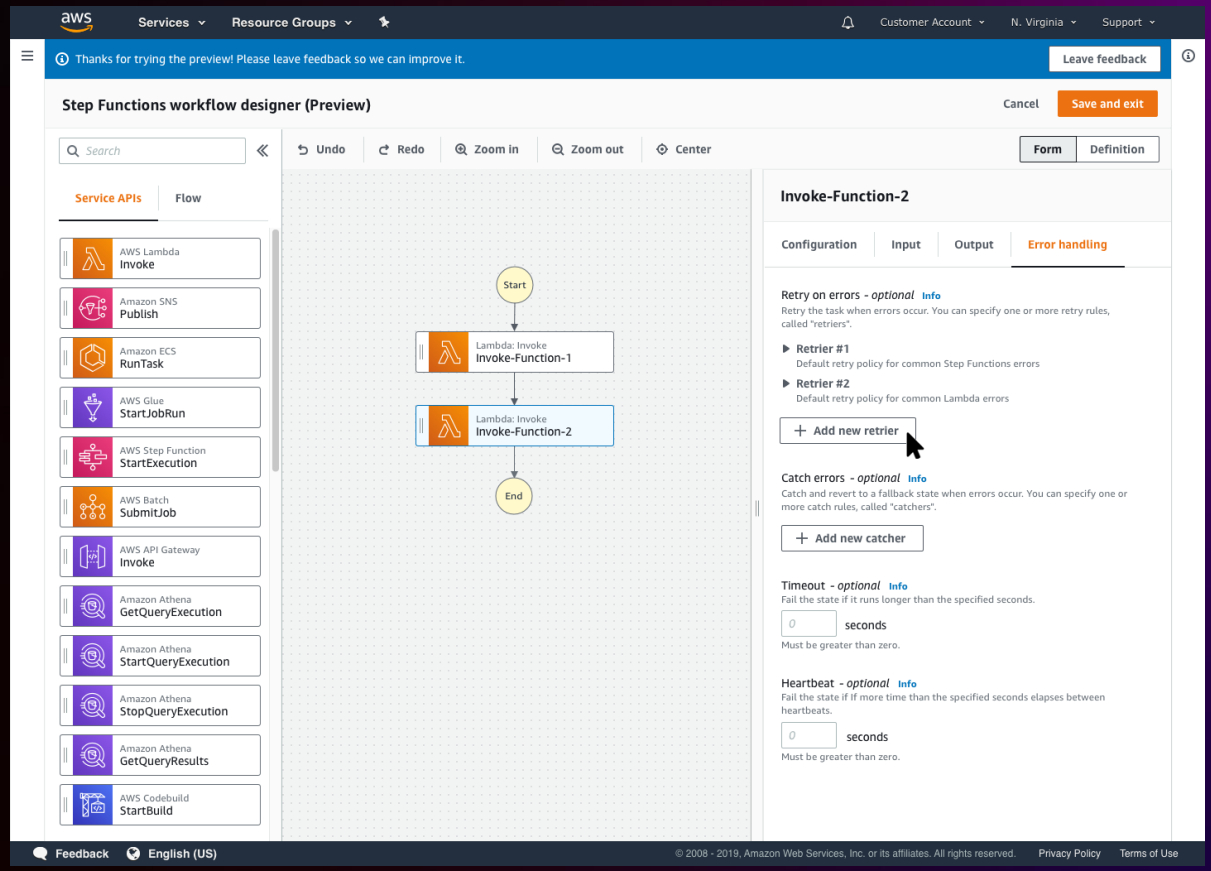
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Step Functions

The **workflows** you build with Step Functions are called **state machines**, and **each step** of your workflow is called a **state**

When you execute your state machine, **each move** from one state to the next is called a **state transition**

You can **reuse components**, easily edit the sequence of steps, or swap out the code called by task states as your needs change



Step Functions Workflow Studio

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

Visual workflows

Define

JSON – Amazon States Language

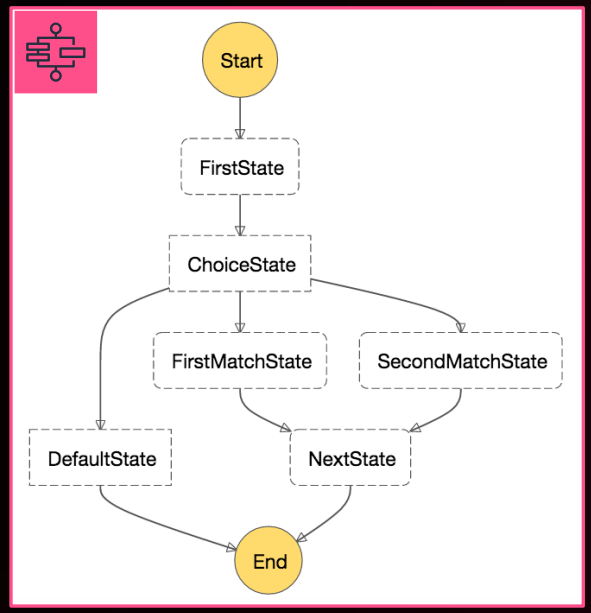
```
Code
{
  "Comment": "An AWL example using a choice state.",
  "StartAt": "FirstState",
  "States": {
    "FirstState": {
      "Type": "Task",
      "Resource": "arn:aws:lambda:REGION:ACCOUNT_ID:function:FUNCTION_NAME",
      "Next": "ChoiceState"
    },
    "ChoiceState": {
      "Type" : "Choice",
      "Choices": [
        {
```

CDK TypeScript, JavaScript, Python, Java, C#

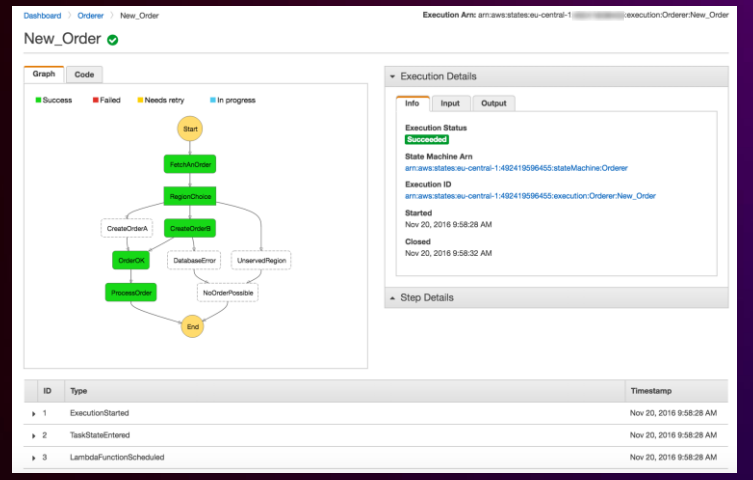
Data science SDK

Python

Visualize



Execute and monitor





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

Step Functions integration types



Optimized integrations

- Customized to simplify the usage of 17 AWS services

Supported integration patterns

- Request Response
- Wait for a Callback (.waitForTaskToken)
- Run a Job (.sync)

AWS SDK integrations

- Call 200+ AWS services directly (10,000+ API actions)

Supported integration patterns

- Request Response
- Wait for a Callback (.waitForTaskToken)

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

Request Response

SERVICE INTEGRATION PATTERN

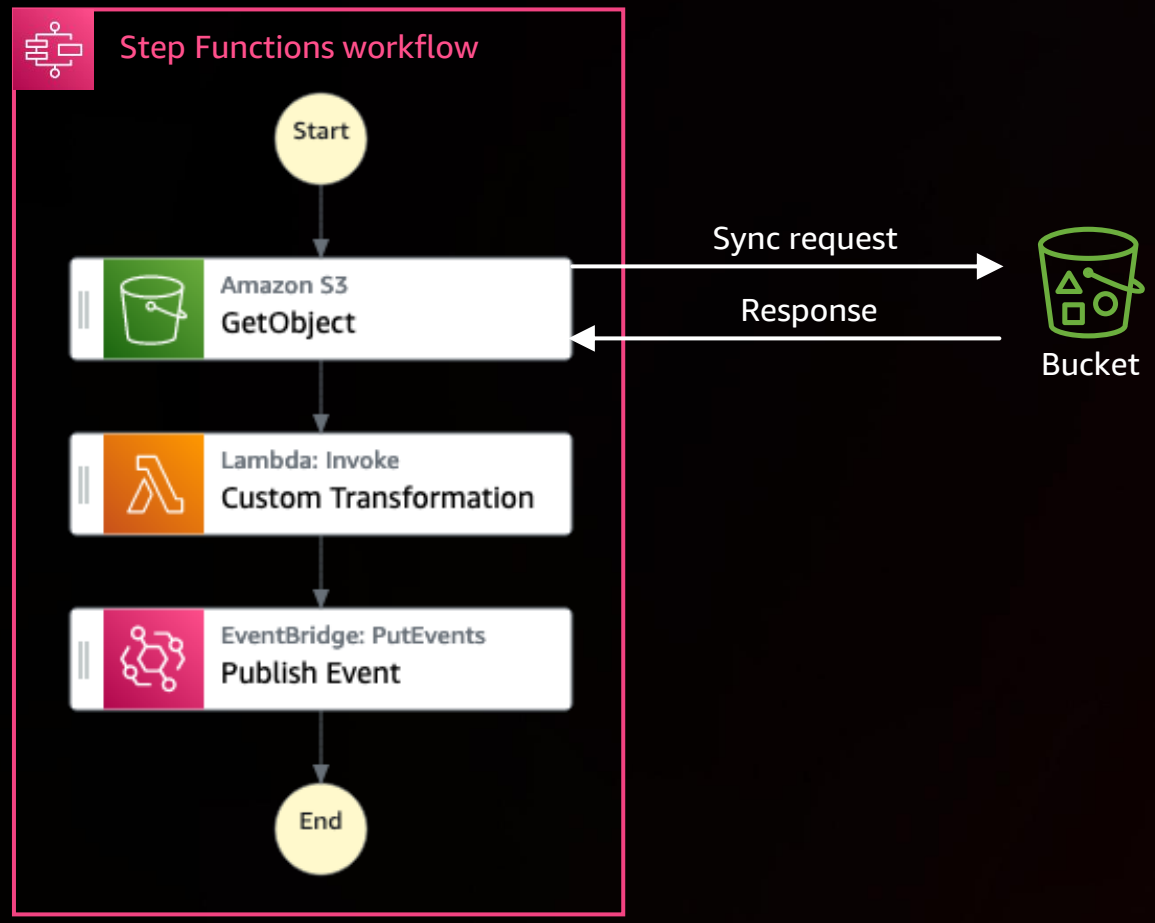


- Amazon EMR
- Lambda
- Amazon SQS
- AWS Glue
- Amazon ECS/ AWS Fargate
- Amazon SNS
- Amazon Athena
- Amazon EKS
- Step Functions
- SageMaker
- AWS Batch
- EventBridge
- Amazon DynamoDB
- AWS SDK
- API Gateway

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

Request Response

SERVICE INTEGRATION PATTERN

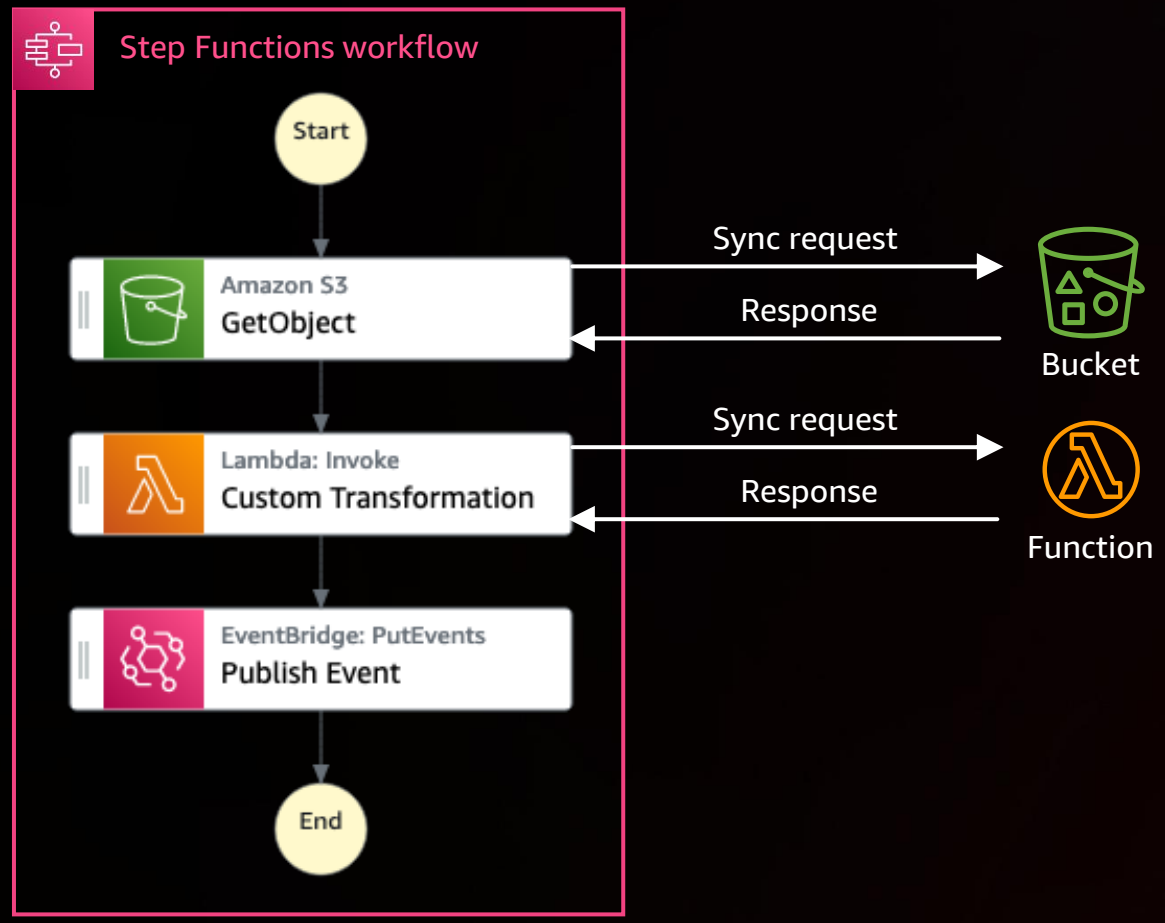


- Amazon EMR
- Lambda
- Amazon SQS
- AWS Glue
- Amazon ECS/ AWS Fargate
- Amazon SNS
- Amazon Athena
- Amazon EKS
- Step Functions
- SageMaker
- AWS Batch
- EventBridge
- Amazon DynamoDB
- AWS SDK
- API Gateway

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

Request Response

SERVICE INTEGRATION PATTERN

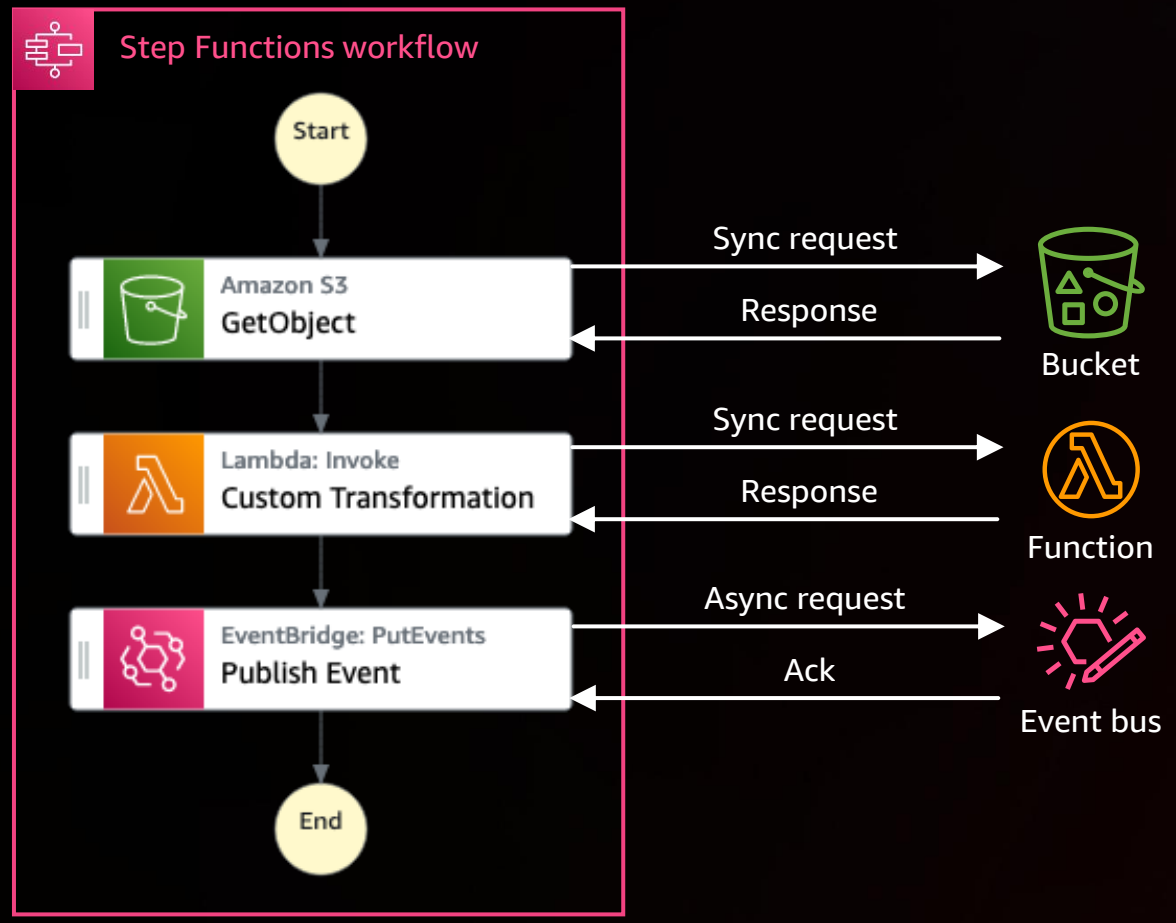


- Amazon EMR
- Lambda
- Amazon SQS
- AWS Glue
- Amazon ECS/ AWS Fargate
- Amazon SNS
- Amazon Athena
- Amazon EKS
- Step Functions
- SageMaker
- AWS Batch
- EventBridge
- Amazon DynamoDB
- AWS SDK
- API Gateway

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

Request Response

SERVICE INTEGRATION PATTERN

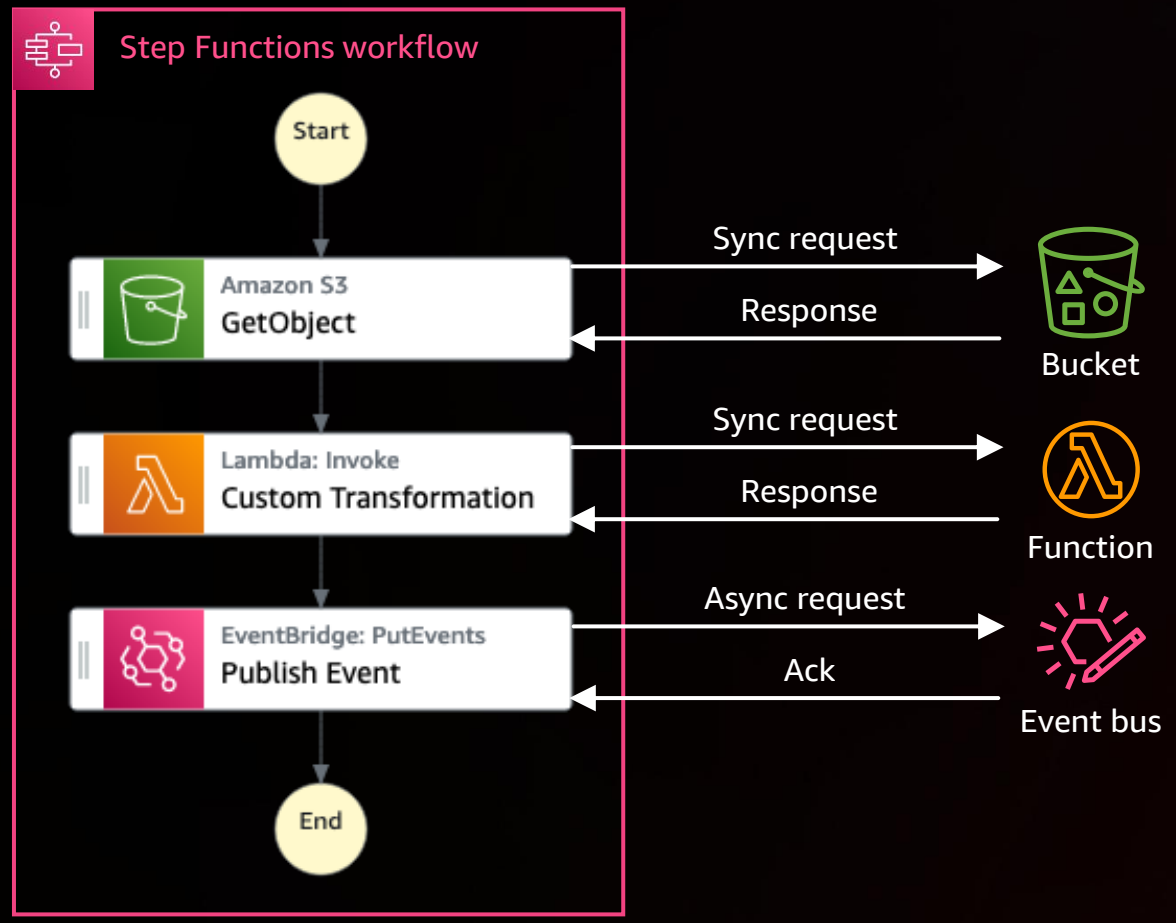


- Amazon EMR
- Lambda
- Amazon SQS
- AWS Glue
- Amazon ECS/ AWS Fargate
- Amazon SNS
- Amazon Athena
- Amazon EKS
- Step Functions
- SageMaker
- AWS Batch
- EventBridge
- Amazon DynamoDB
- AWS SDK
- API Gateway

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

Request Response

SERVICE INTEGRATION PATTERN

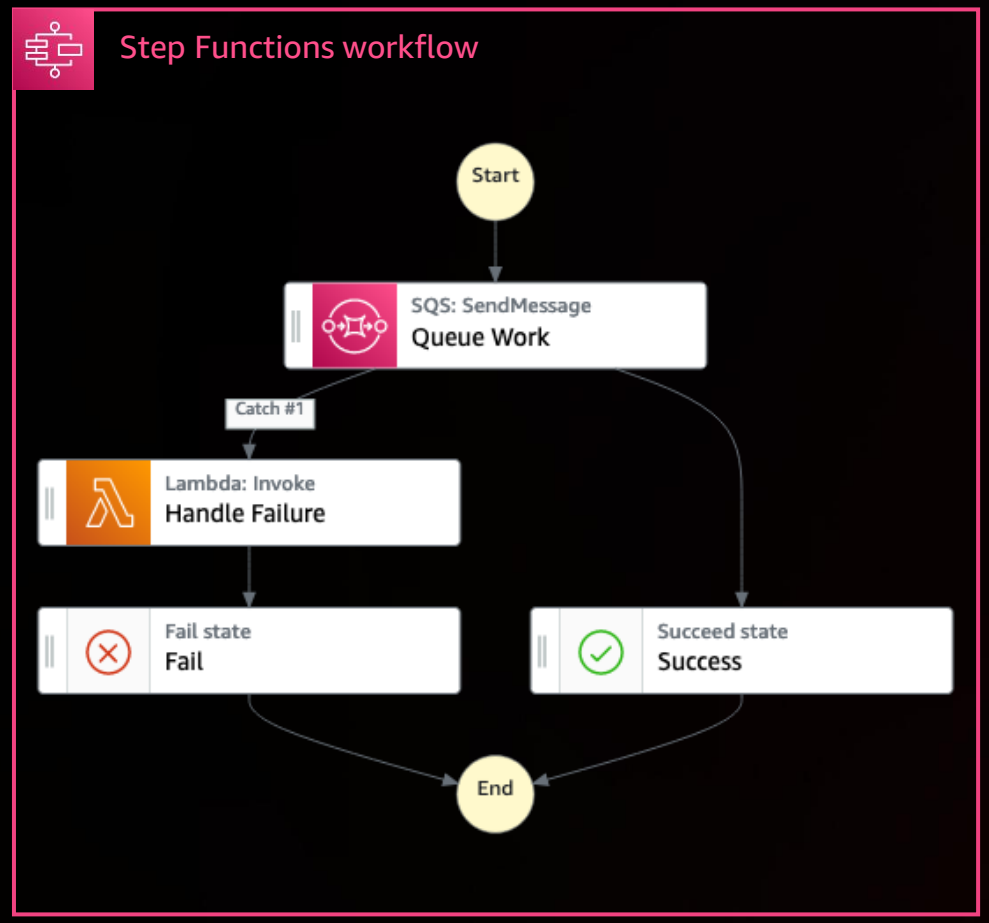




© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Wait for a Callback (.waitForTaskToken)

SERVICE INTEGRATION PATTERN

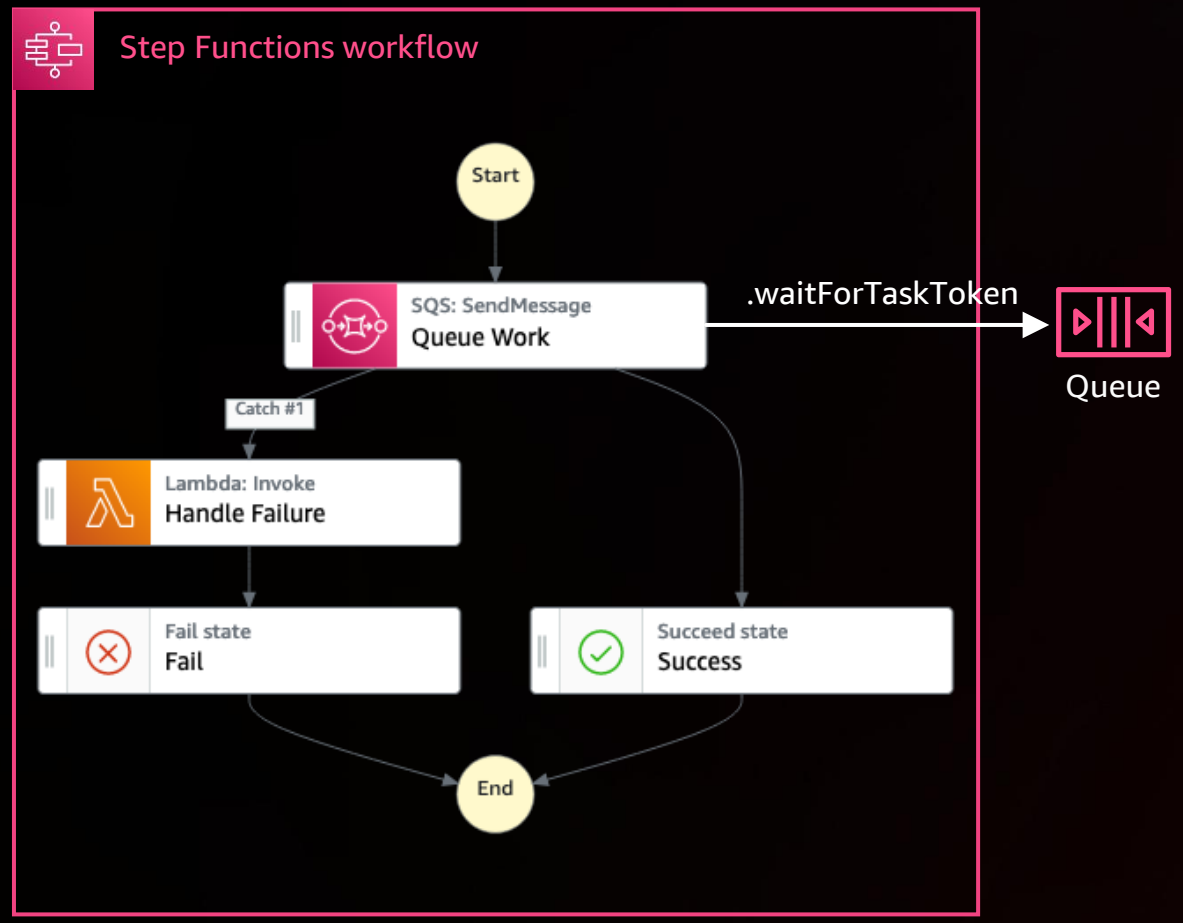


- Lambda
- Amazon SQS
- Amazon ECS/ AWS Fargate
- Amazon SNS
- Step Functions
- EventBridge
- AWS SDK
- API Gateway

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

Wait for a Callback (.waitForTaskToken)

SERVICE INTEGRATION PATTERN



Lambda

Amazon SQS

Amazon ECS/
AWS Fargate

Amazon SNS

Step Functions

EventBridge

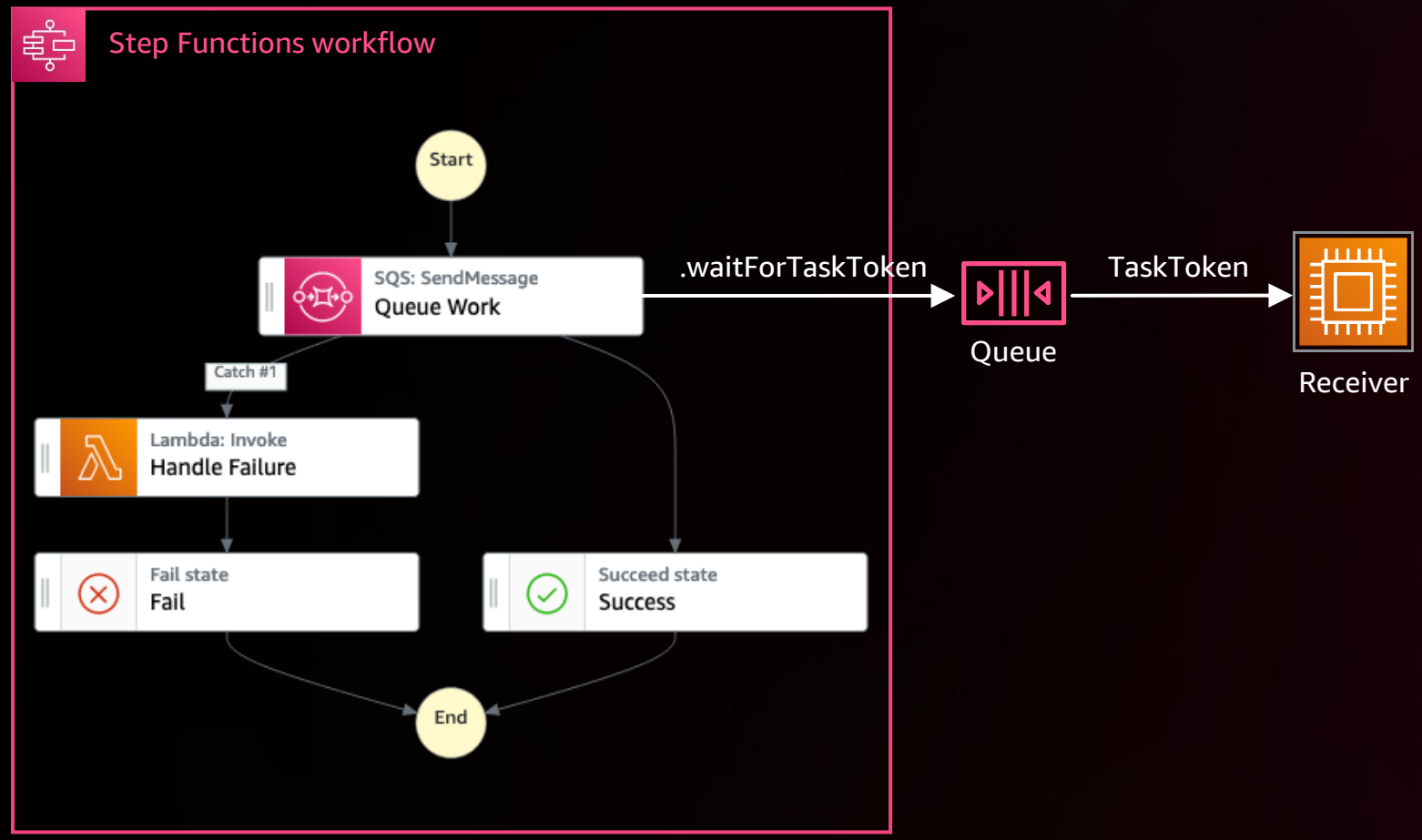
AWS SDK

API Gateway

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

Wait for a Callback (.waitForTaskToken)

SERVICE INTEGRATION PATTERN



.waitForTaskToken

Queue

TaskToken

Receiver

Lambda

Amazon SQS

Amazon ECS/
AWS Fargate

Amazon SNS

Step Functions

EventBridge

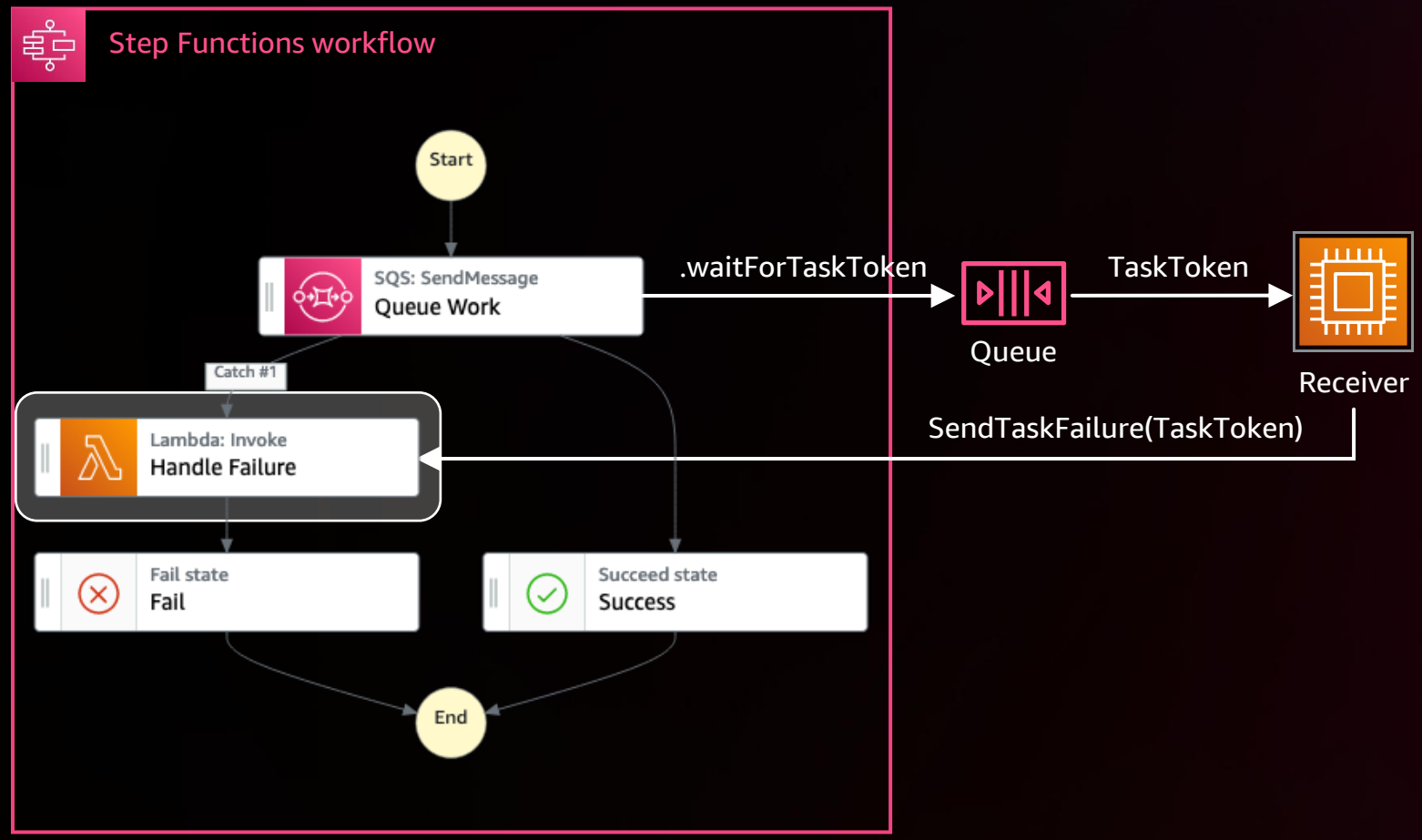
AWS SDK

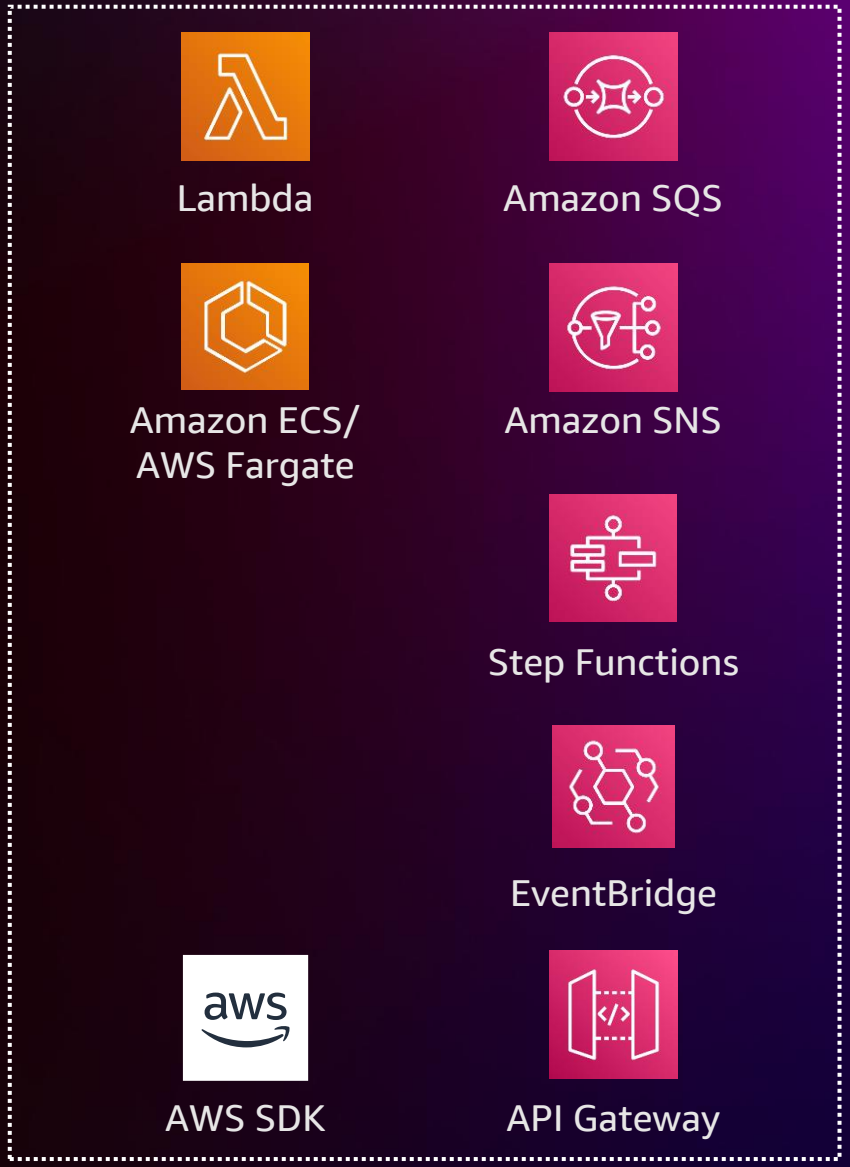
API Gateway

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

Wait for a Callback (.waitForTaskToken)

SERVICE INTEGRATION PATTERN

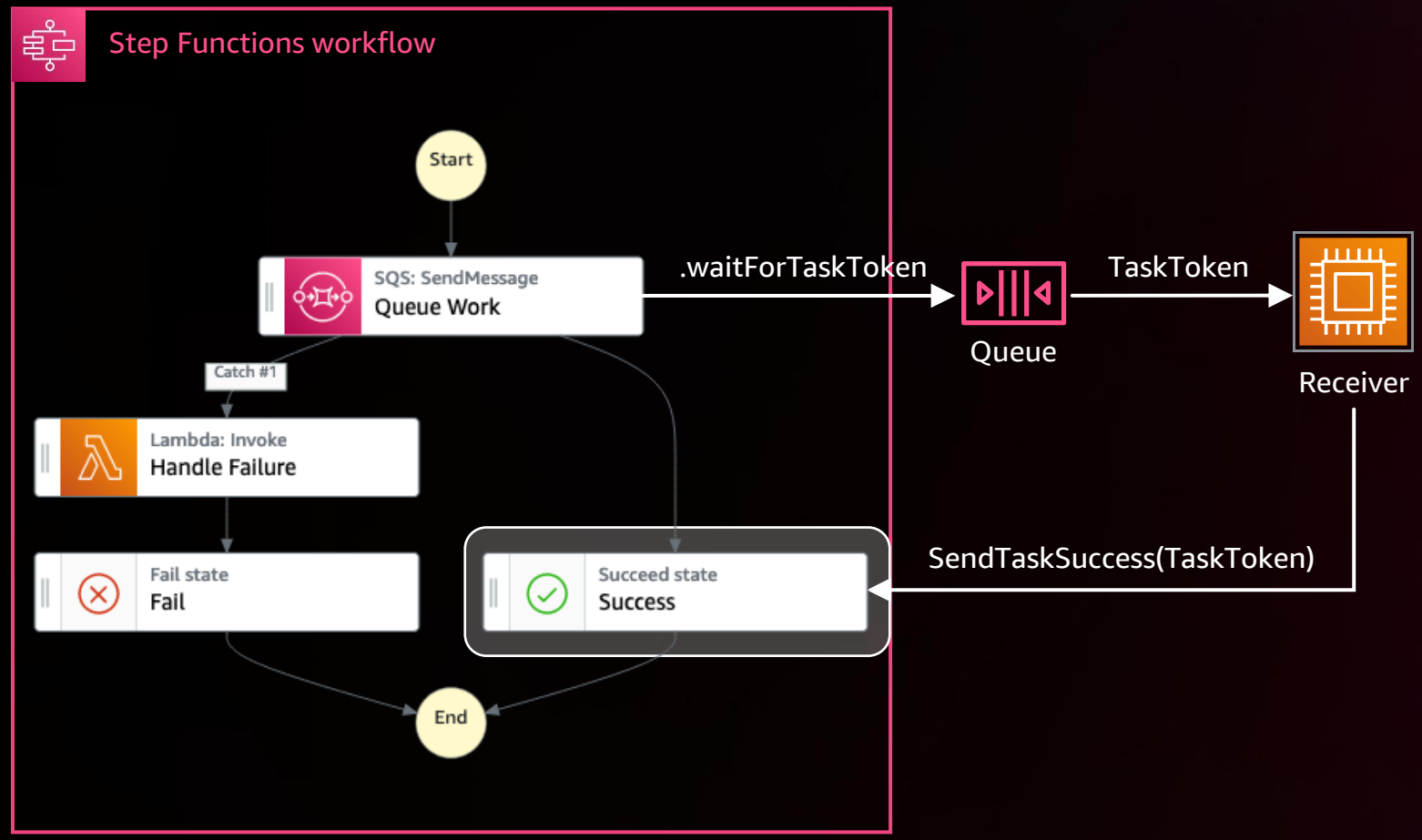


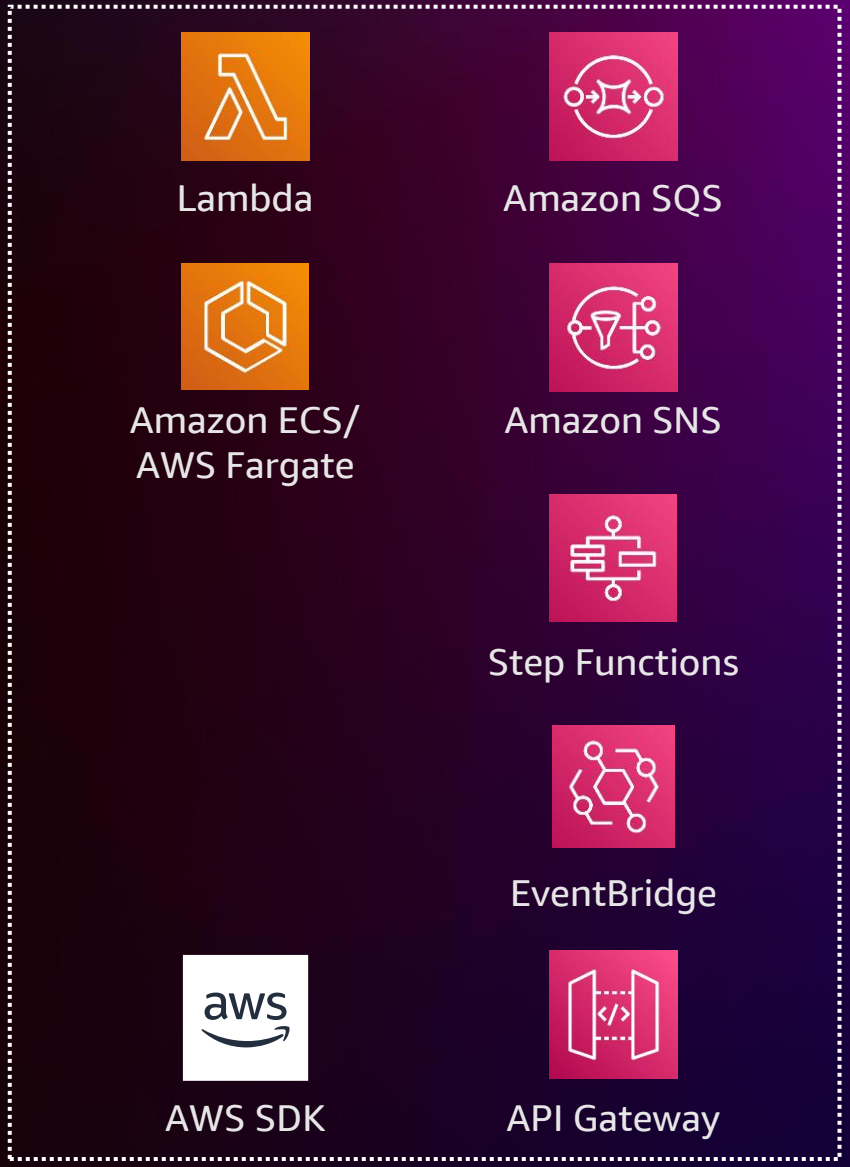


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Wait for a Callback (.waitForTaskToken)

SERVICE INTEGRATION PATTERN

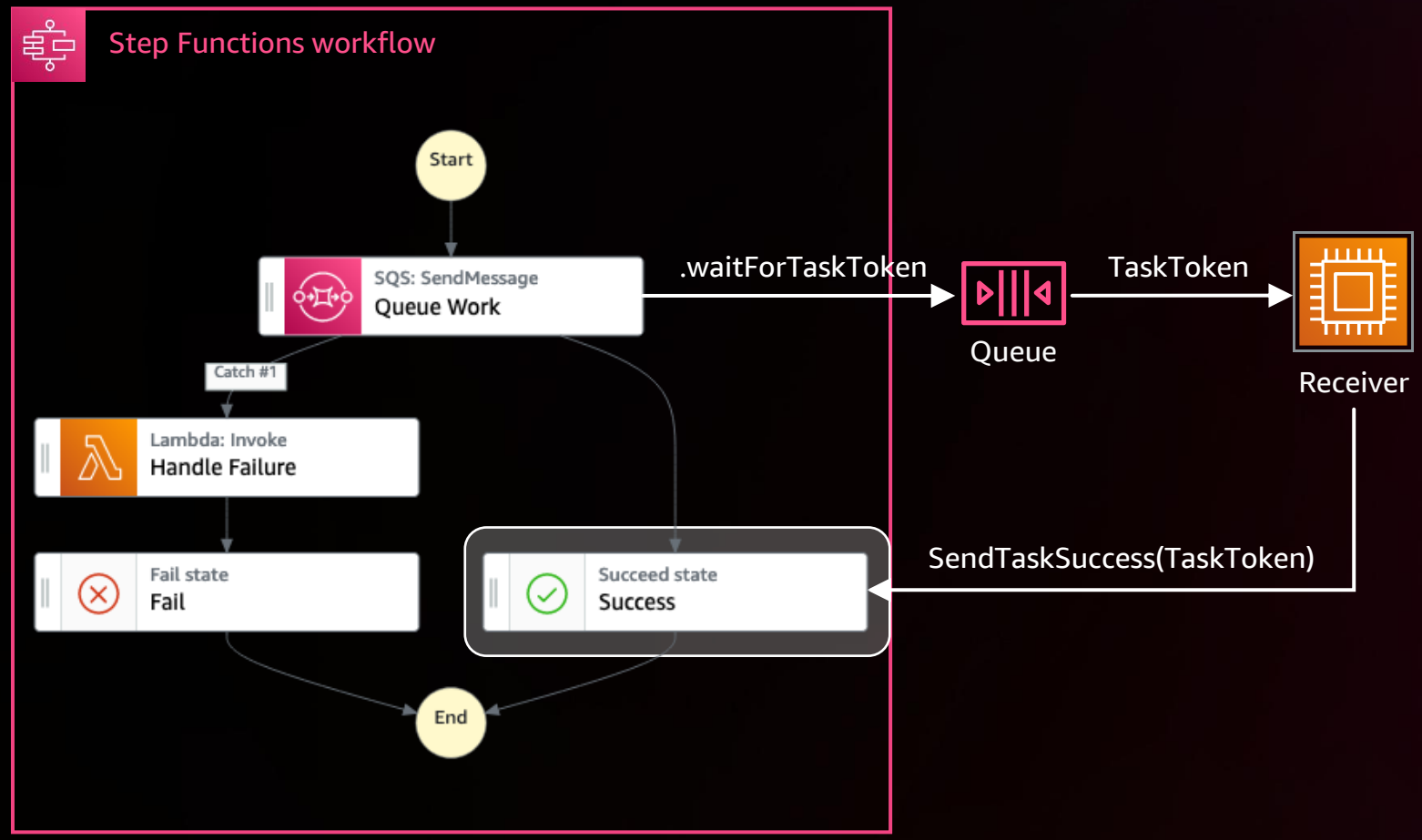


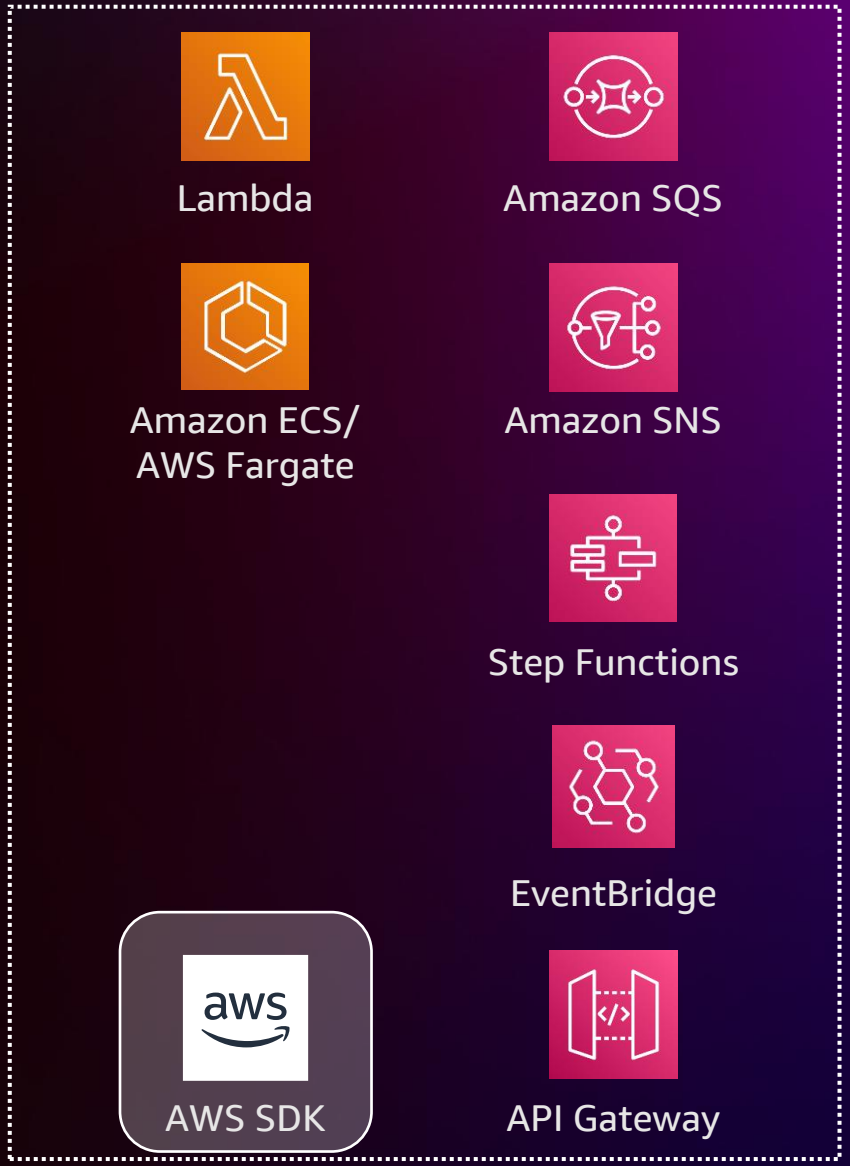


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Wait for a Callback (.waitForTaskToken)

SERVICE INTEGRATION PATTERN

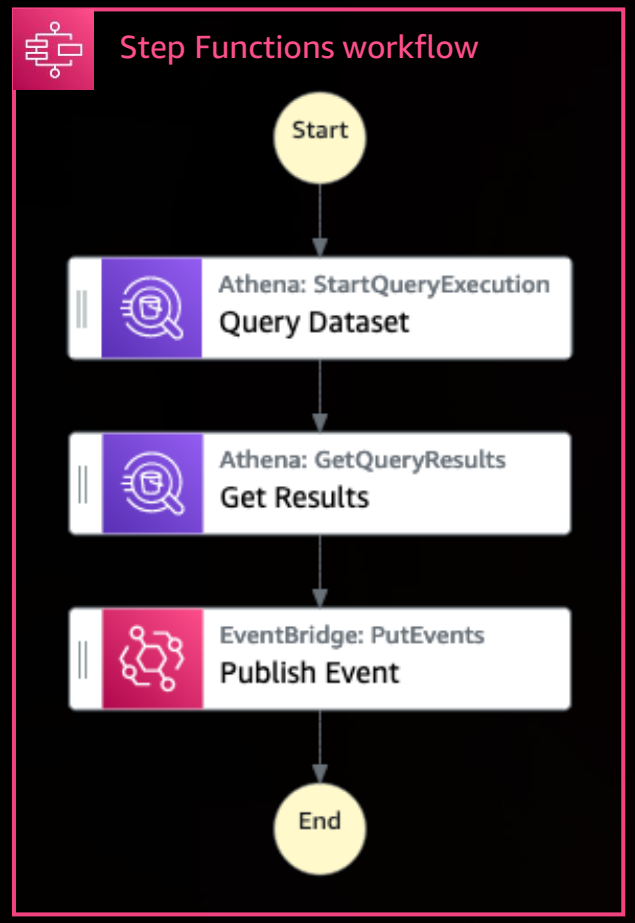




© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Run a Job (.sync)

SERVICE INTEGRATION PATTERN

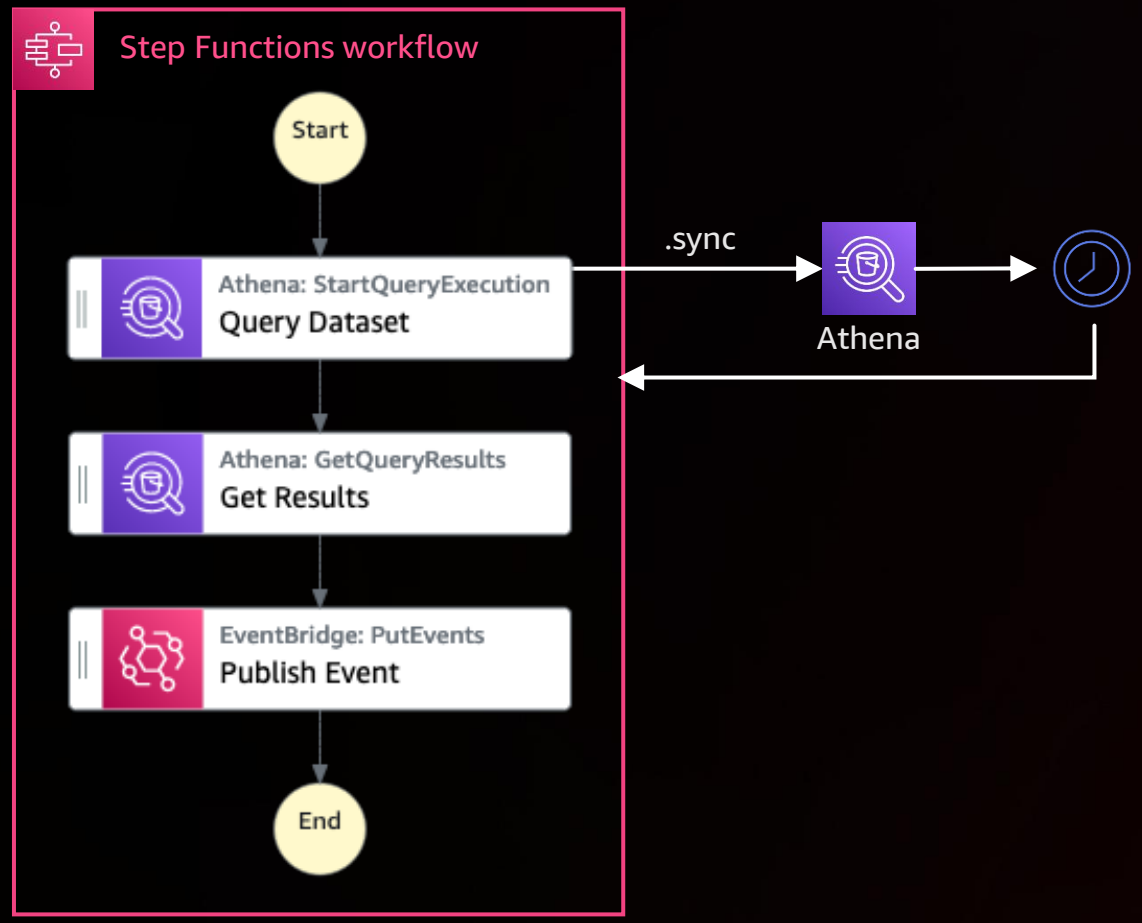


- Amazon EMR
- AWS Glue
- Amazon ECS/ AWS Fargate
- Amazon Athena
- Amazon EKS
- Step Functions
- SageMaker
- AWS Batch
- CodeBuild
- API Gateway

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

Run a Job (.sync)

SERVICE INTEGRATION PATTERN

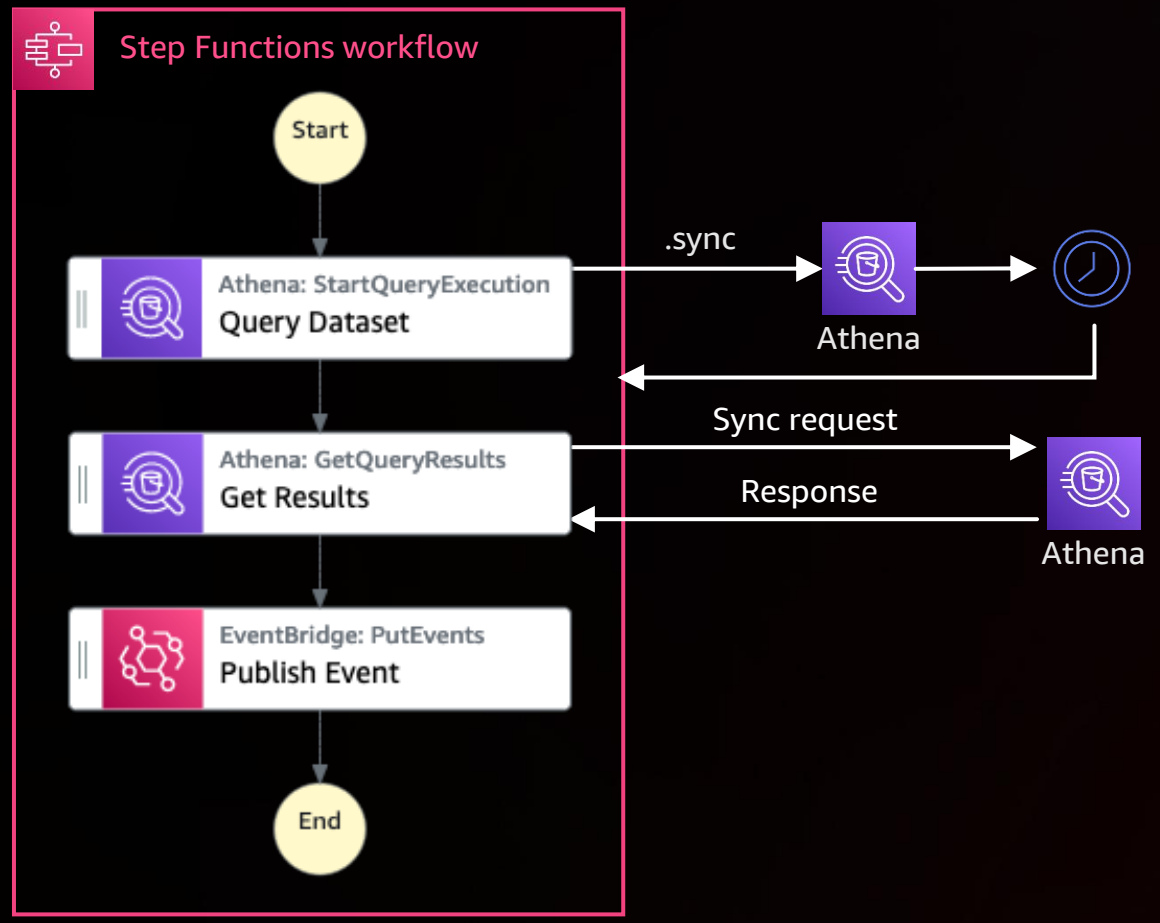


- Amazon EMR
- AWS Glue
- Amazon ECS/ AWS Fargate
- Amazon Athena
- Amazon EKS
- Step Functions
- SageMaker
- AWS Batch
- CodeBuild
- API Gateway

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

Run a Job (.sync)

SERVICE INTEGRATION PATTERN



Amazon EMR

AWS Glue

Amazon ECS/
AWS Fargate

Amazon Athena

Amazon EKS

Step Functions

SageMaker

AWS Batch

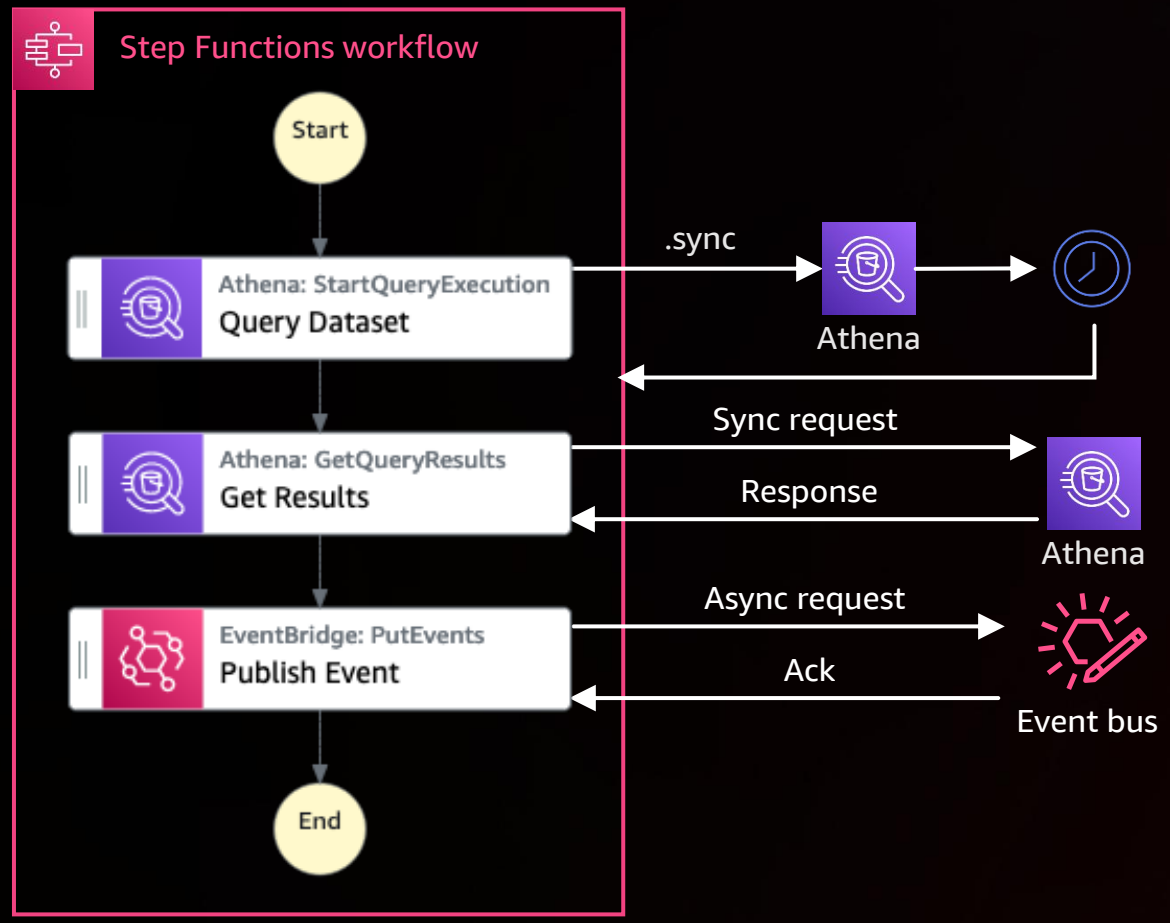
CodeBuild

API Gateway

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

Run a Job (.sync)

SERVICE INTEGRATION PATTERN



Amazon EMR

AWS Glue

Amazon ECS/
AWS Fargate

Amazon Athena

Amazon EKS

Step Functions

SageMaker

AWS Batch

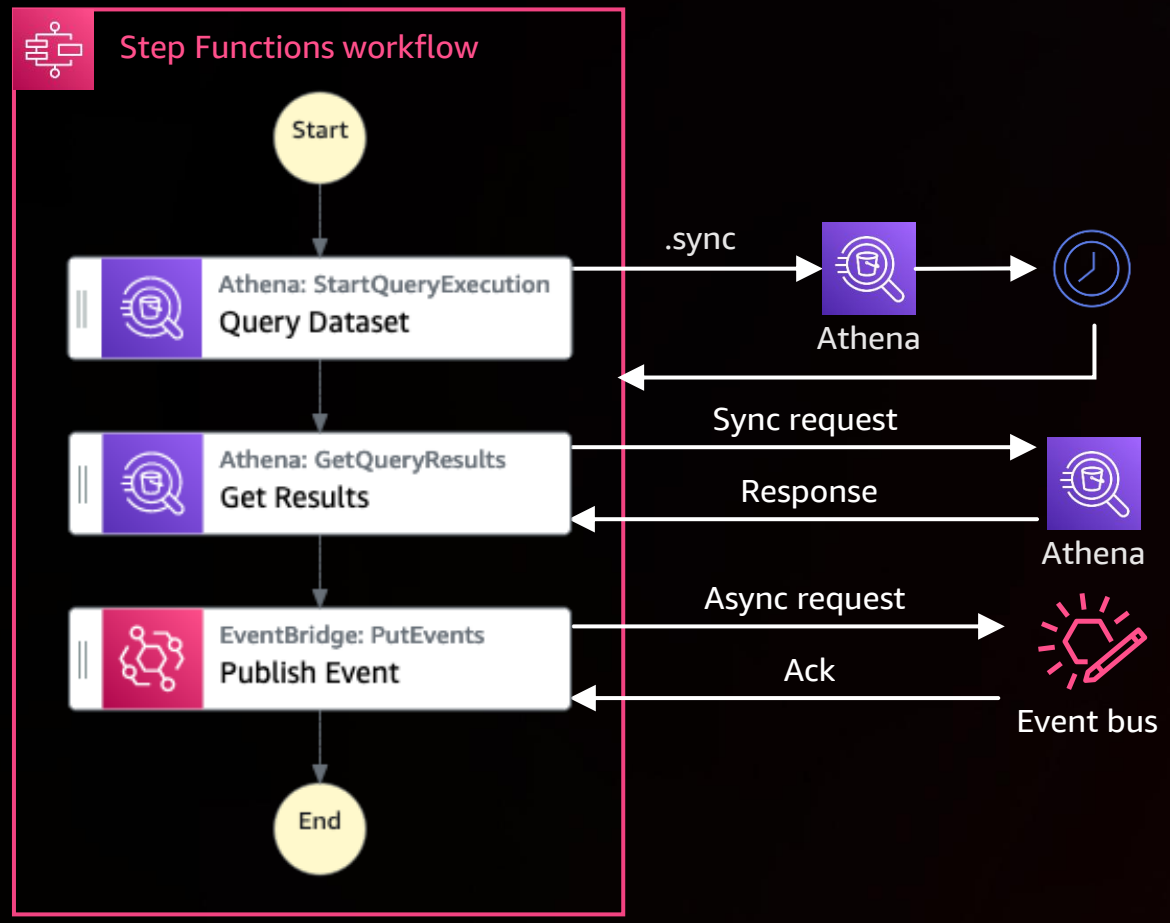
CodeBuild

API Gateway

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

Run a Job (.sync)

SERVICE INTEGRATION PATTERN



Amazon EMR

AWS Glue

Amazon ECS/
AWS Fargate

Amazon Athena

Amazon EKS

Step Functions

SageMaker

AWS Batch

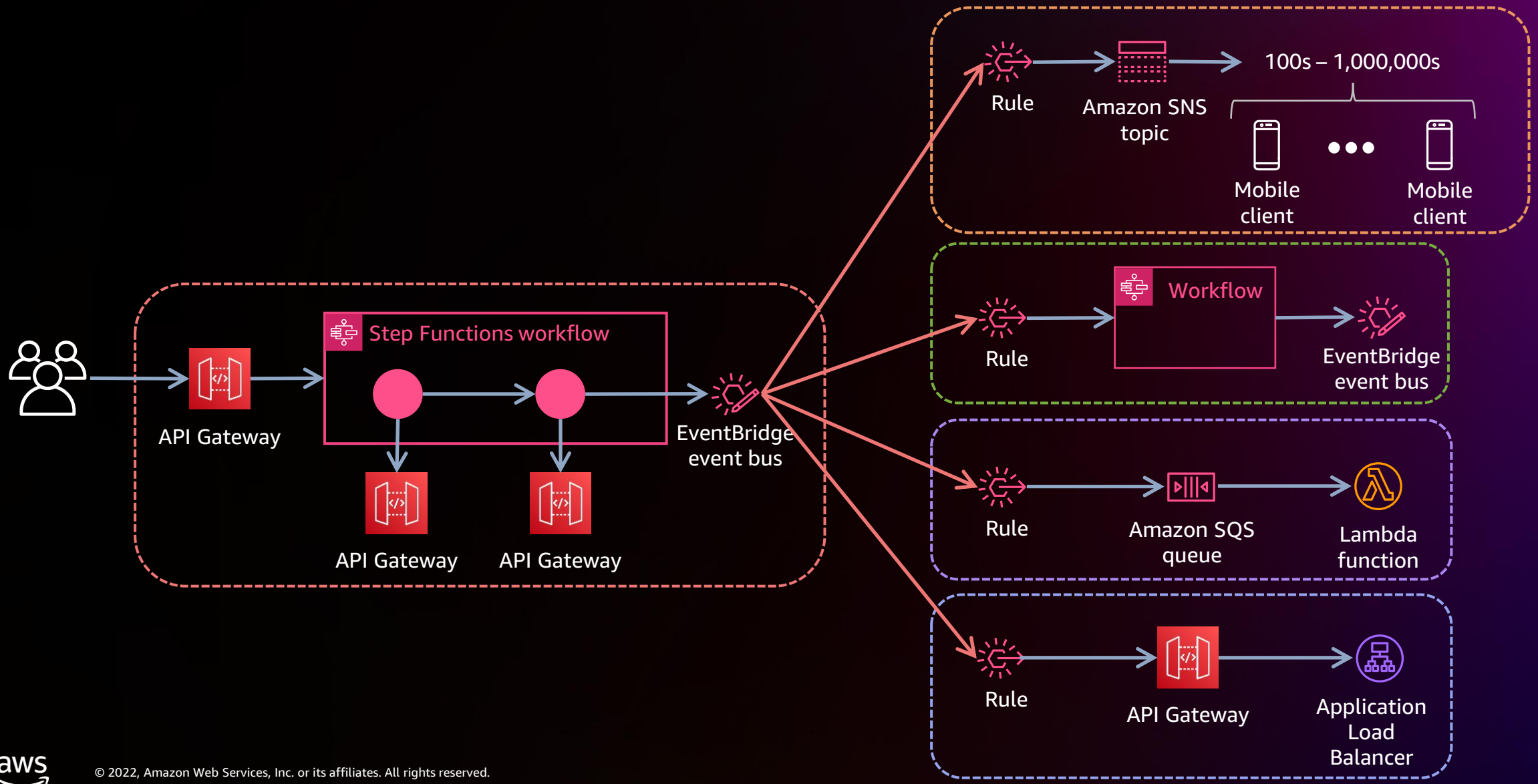
CodeBuild

AWS SDK

API Gateway

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

Better together: Orchestration + Choreography



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

Handling event duplication using idempotency

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

What is idempotence, idempotency, idempotent?

"[Mathematical operations] that can be applied multiple times without changing the result." – Wikipedia

```
f(x) = x + 0 = x
f(x) = x * 1 = x
```

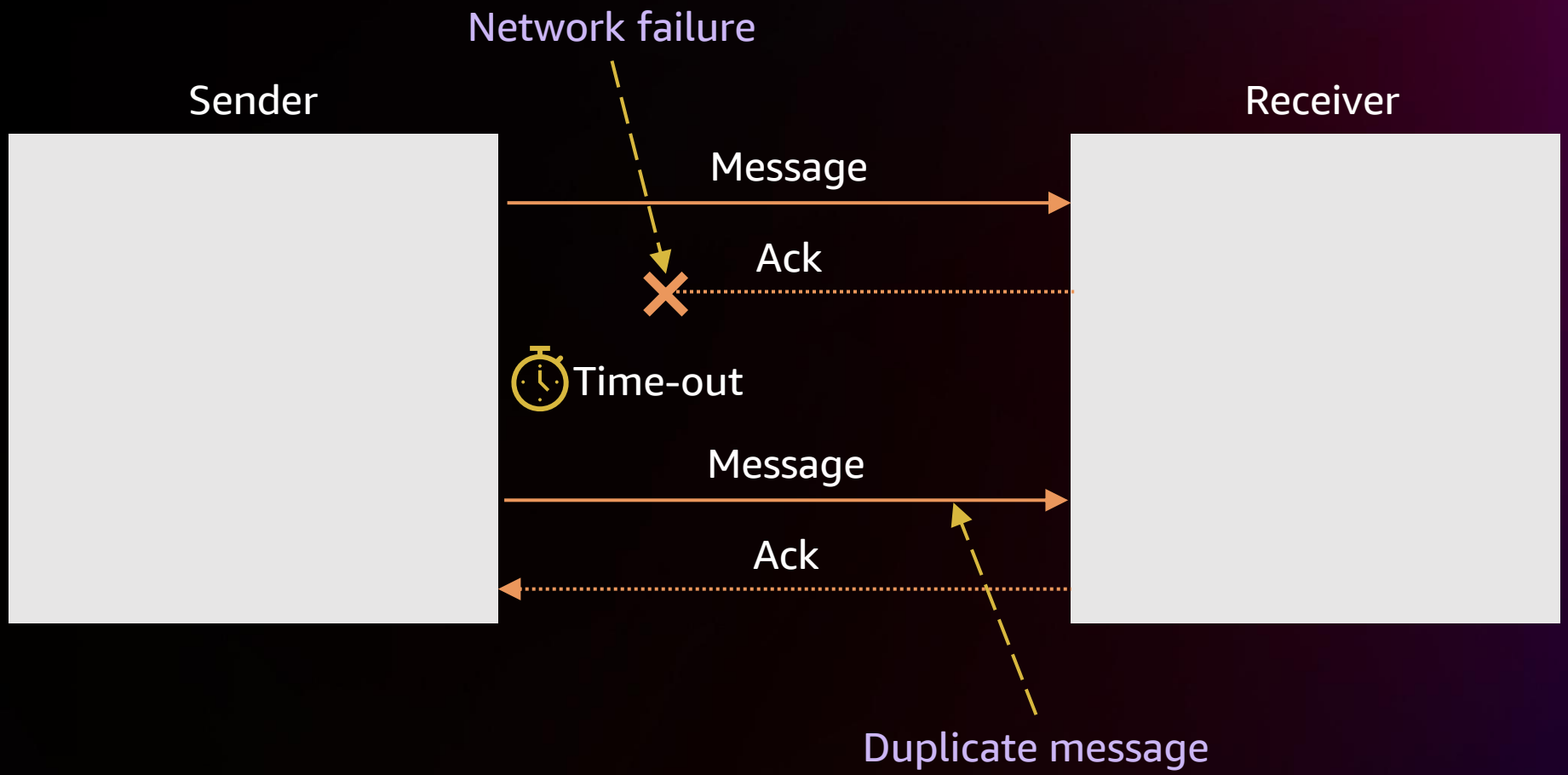
"A message that has the same effect whether it is received once or multiple times." – "Enterprise Integration Patterns" (Hohpe, Woolf)

"Did the internet charge my credit card twice?" – Text message (Mom)

aws

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

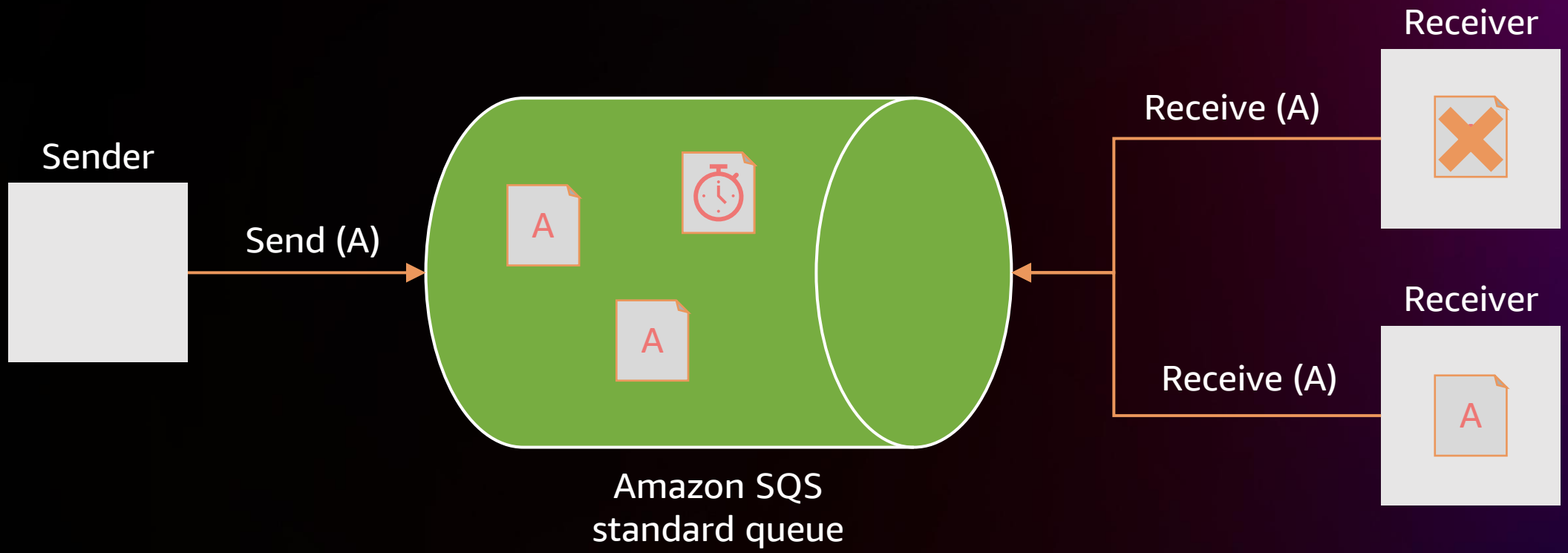
Duplication: Problem sending acknowledgement



"Enterprise Integration Patterns" (Hohpe, Woolf)

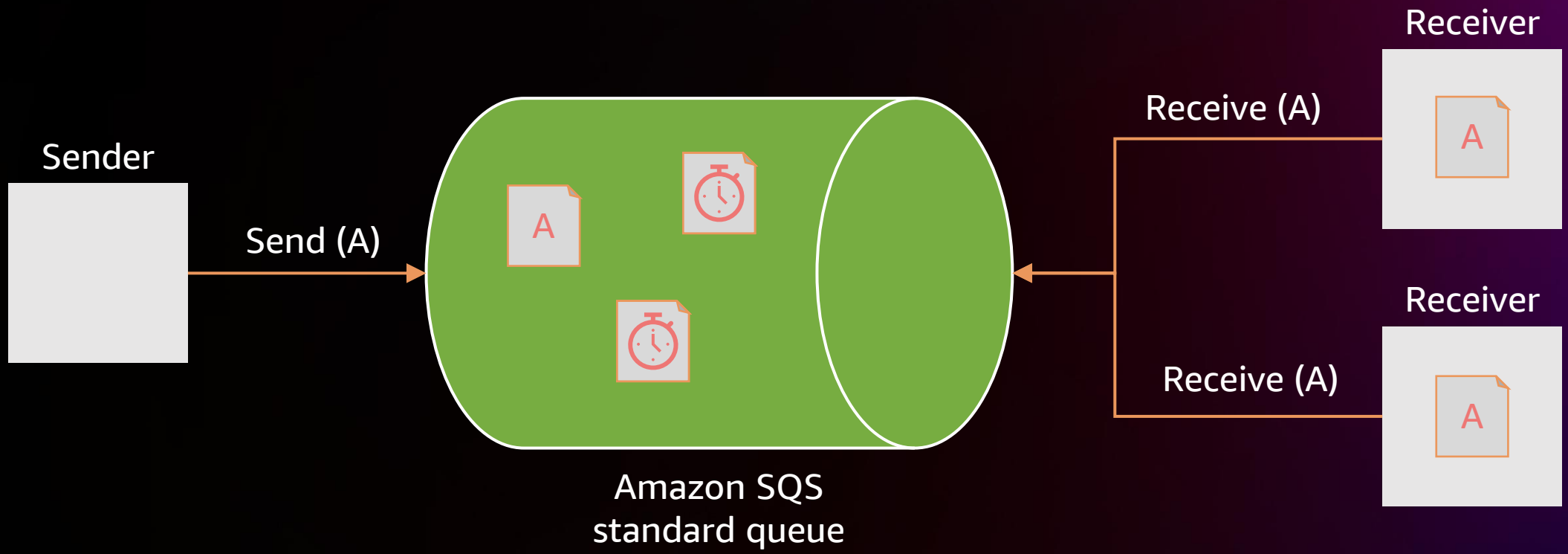
aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Duplication: Problem processing message



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

Duplication: At-least-once delivery



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

Idempotency tokens

Idempotency tokens uniquely identify messages to enable receivers to avoid duplicate side effects

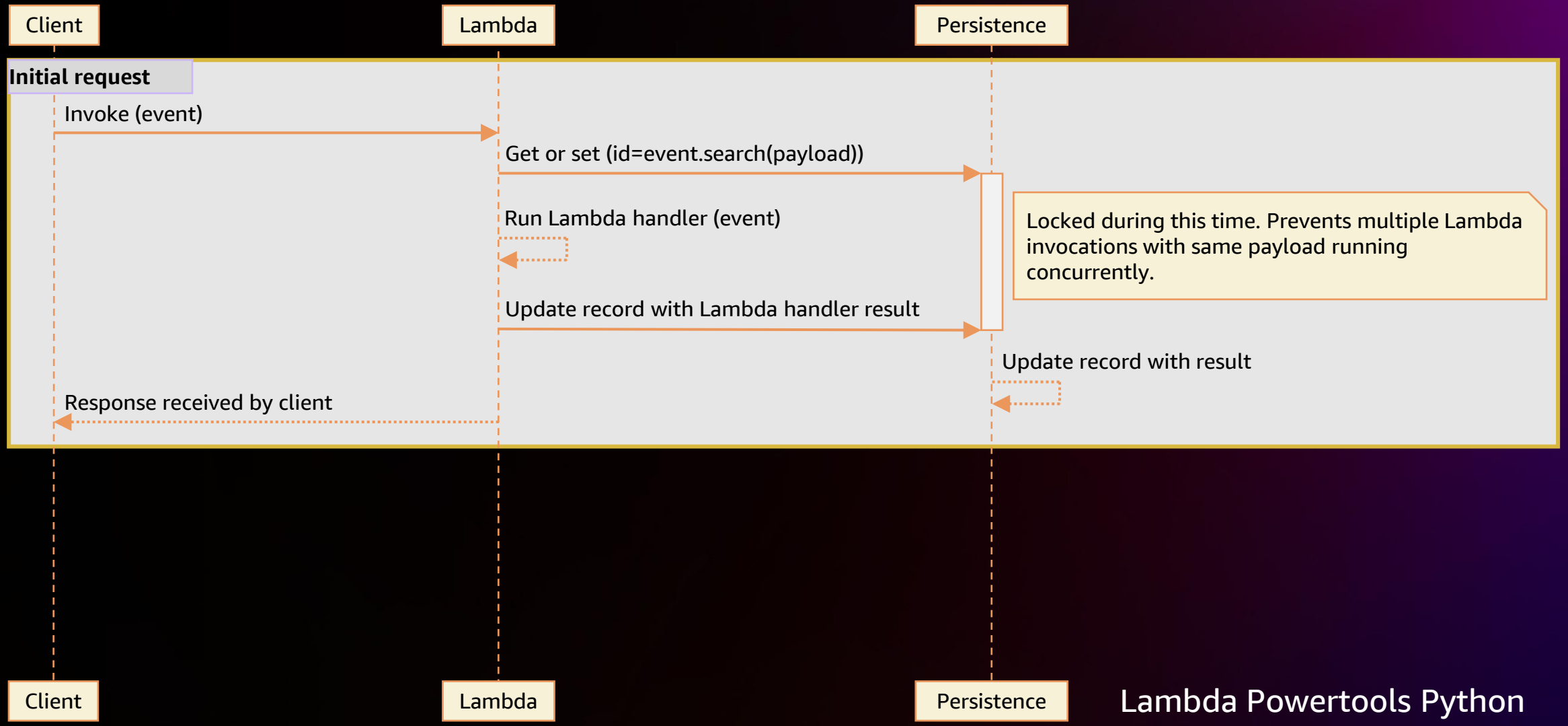
Well-behaved idempotency tokens

1. Are **generated by the client**
2. Are ***regenerated* by the client** on retry
3. Are unique per message (**UUID** is common)
4. Use a **dedicated field**, separate from message content

aws

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

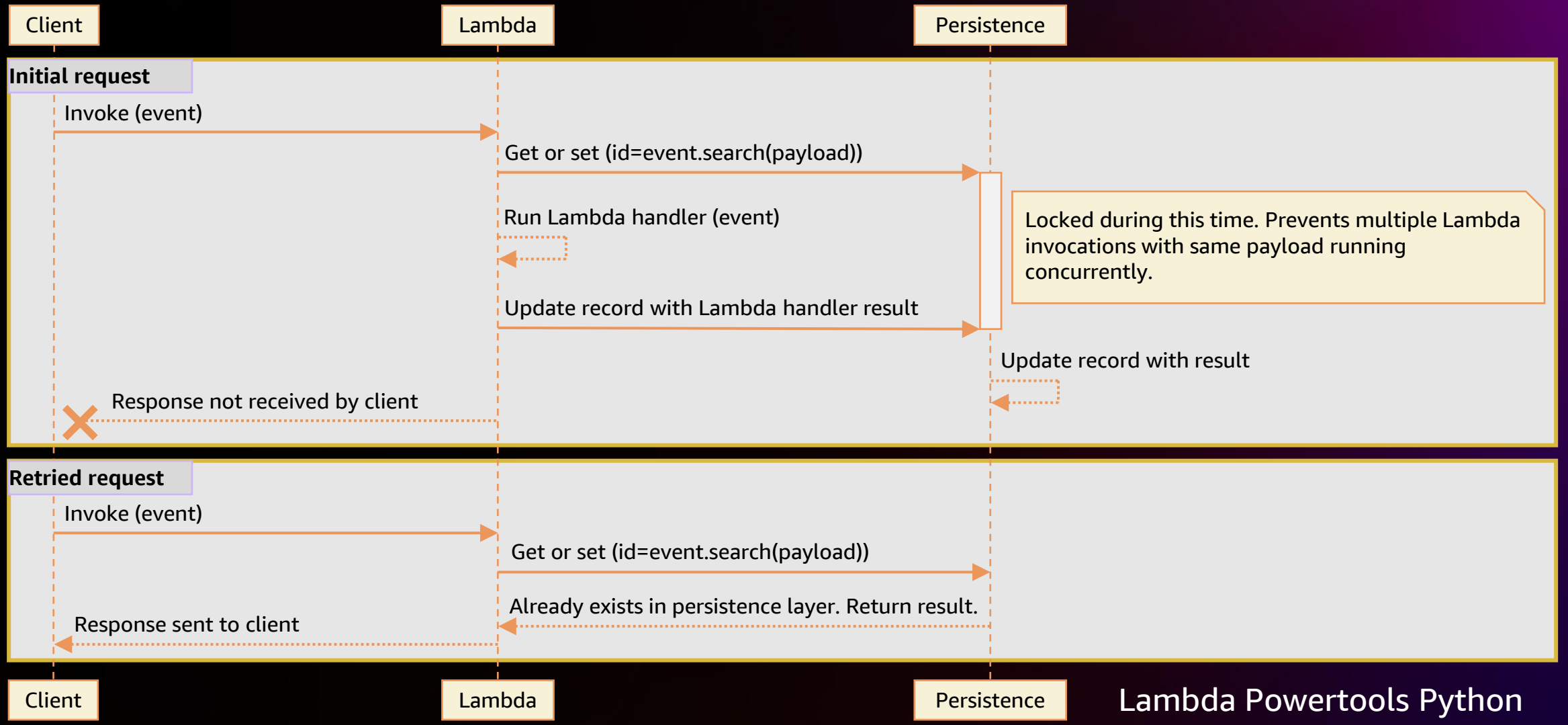
Successful idempotent request sequence



Lambda Powertools Python
bit.ly/37UURGF

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

Retried idempotent request sequence



Lambda Powertools Python
bit.ly/37UURGF

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

Idempotency duration

Provider	Idempotency duration
Lambda Powertools idempotency utilities (default)	1 hour
Amazon SQS (FIFO) SendMessage	5 minutes
Amazon SNS (FIFO) Publish	5 minutes
Step Functions StartExecution	90 days

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

Example: SQS.FIFO SendMessage API

5 minutes*

```
% TOKEN=$(uuidgen) && echo $TOKEN
2DAA31B6-3ED6-4A9A-A4EE-825C88AA8A67

% aws sqs send-message --queue-url "https://sqs.us-east-
1.amazonaws.com/068896461592/idempotence.fifo" --message-deduplication-id "$TOKEN" --
message-body "Hello, World" --message-group-id foo
{
    "MD5OfMessageBody": "82bb413746aee42f89dea2b59614f9ef",
    "MessageId": "9ce79aca-2bbd-4a29-a761-0cc2dc3cd1b9",
    "SequenceNumber": "18869557808186863616"
}

% aws sqs send-message --queue-url "https://sqs.us-east-
1.amazonaws.com/068896461592/idempotence.fifo" --message-deduplication-id "$TOKEN" --
message-body "Hello, World" --message-group-id foo
{
    "MD5OfMessageBody": "82bb413746aee42f89dea2b59614f9ef",
    "MessageId": "9ce79aca-2bbd-4a29-a761-0cc2dc3cd1b9",
    "SequenceNumber": "18869557808186863616"
}
```

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

*Amazon SQS Developer Guide: https://go.aws/38K2QpY

Example: Step Functions StartExecution API

90 days*

```
% TOKEN=$(uuidgen) && echo $TOKEN
AF8074B2-3C23-415B-B465-71A849C63452

% aws stepfunctions start-execution --state-machine-arn arn:aws:states:us-east-
1:068896461592:stateMachine:StandardWait60 --name "$TOKEN"
{
  "executionArn": "arn:aws:states:us-east-
1:068896461592:execution:StandardWait60:AF8074B2-3C23-415B-B465-71A849C63452",
  "startDate": "2022-05-01T17:50:48.073000-04:00"
}

% aws stepfunctions start-execution --state-machine-arn arn:aws:states:us-east-
1:068896461592:stateMachine:StandardWait60 --name "$TOKEN"
{
  "executionArn": "arn:aws:states:us-east-
1:068896461592:execution:StandardWait60:AF8074B2-3C23-415B-B465-71A849C63452",
  "startDate": "2022-05-01T17:50:48.073000-04:00"
}
```

uuidgen(1): https://bit.ly/37cxKqx

*AWS Step Functions API Reference: go.aws/37aSajl

aws

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

Example: EventBridge PutEvents API

NOT IDEMPOTENT!

```
% aws events put-events --entries '[{"Source": "foo", "DetailType": "bar", "Detail": "{}"}]'
{
    "Entries": [
        {
            "EventId": "24a2939e-8d31-677f-992d-ff99e4e0eafd"
        }
    ]
}

% aws events put-events --entries '[{"Source": "foo", "DetailType": "bar", "Detail": "{}"}]'
{
    "Entries": [
        {
            "EventId": "61a15356-f8d3-4b6e-7da9-5bfccde8016d"
        }
    ]
}
```

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

Example: EventBridge idempotency identifier

```
{
  "version": "0",
  "id": "61a15356-f8d3-4b6e-7da9-5bfccde8016d",
  "detail-type": "OrderCreated",
  "source": "com.orders",
  "account": "068896461592",
  "time": "2022-05-01T22:15:20Z",
  "region": "us-east-1",
  "detail": {
    "metadata": {
      "idempotency-key": "AF8074B2-3C23-415B-B465-71A849C63452"
    },
    "data": {
      "order-id": "1073459984"
    }
}
```

!! Event.id is not an idempotency ID !!

Client-provided idempotency ID

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

Example: EventBridge PutEvents API

```
% TOKEN=$(uuidgen) && echo $TOKEN
AF8074B2-3C23-415B-B465-71A849C63452

% aws events put-events \
  --entries '[{"Source": "foo", "DetailType": "bar", "Detail": "{\"metadata\": {\"idempotency-
key\": \"$TOKEN\"}}"}]'
{
    "Entries": [
        {
            "EventId": "a6765cc7-e0f8-c78d-4a06-64a5f45d4e3b"
        }
    ]
}
```

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

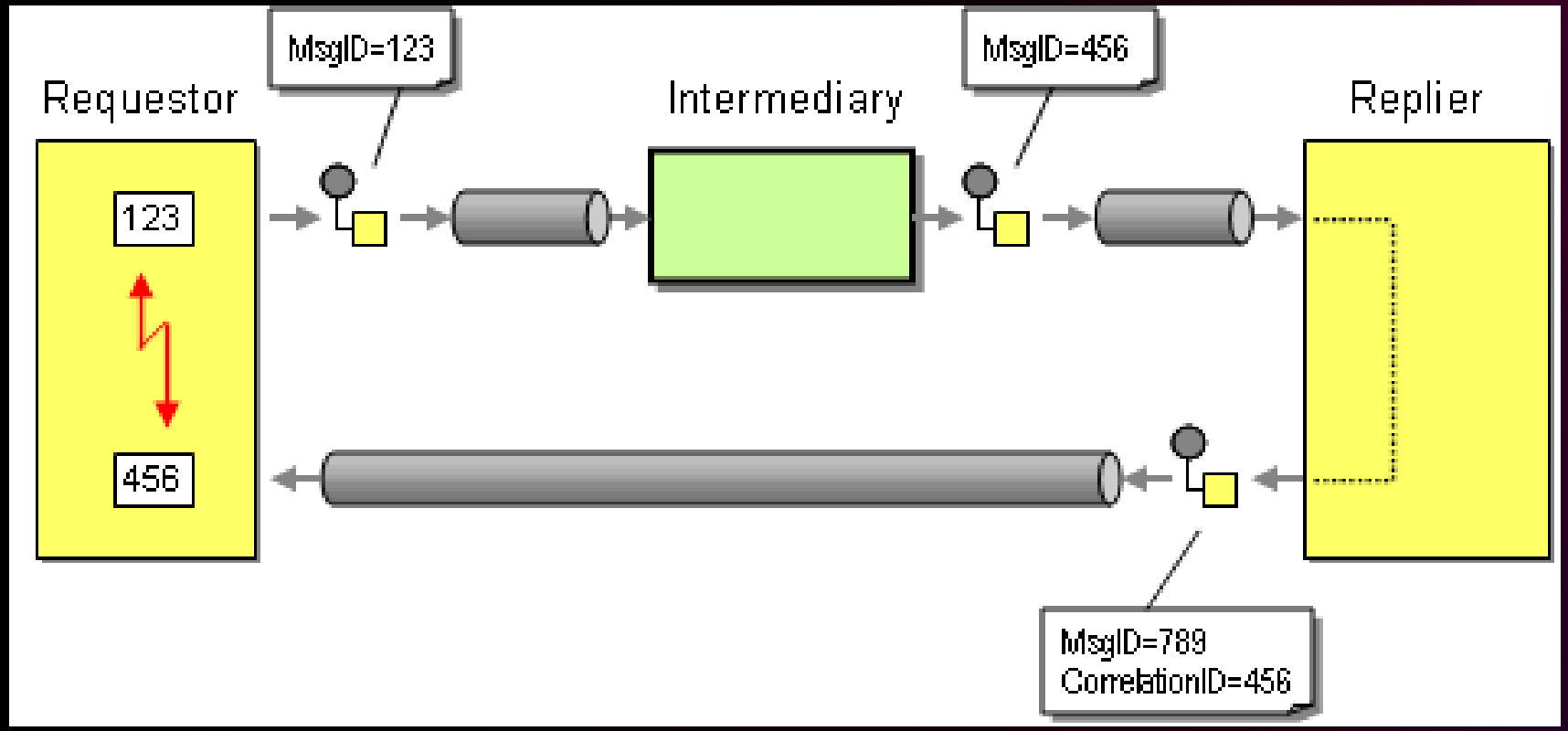
Idempotency by any other name would smell as sweet

Provider	Idempotency identifier
Step Functions StartExecution bit.ly/3NgMHaj	name
Amazon SQS (FIFO) SendMessage go.aws/3LcgGyX	message-deduplication-id
Amazon EC2 RunInstances go.aws/3FKbQYC	client-token
Stripe bit.ly/3a1ci99	Idempotency-Key
PayPal bit.ly/3FGNjDP	PayPal-Request-Id
Square squ.re/3wcjqrH	idempotency_key
IETF	Idempotency-Key
Enterprise Integration Patterns bit.ly/3Me9L9T	Message ID

aws

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

Idempotency tokens flow through intermediaries



Source: Enterprise Integration Patterns, Correlation and Conversations (Hohpe)
bit.ly/3LJiPmV

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

Resources

Don’t stop here, learn more about event-driven architectures



https://s12d.com/api311-22

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

New Blogs Videos Learn Events Patterns About Search Search

Welcome to **Serverless** Land

This site brings together all the latest blogs, videos, and training for AWS Serverless. Learn to use and build apps that scale automatically on low-cost, fully-managed serverless architecture.

Learn More

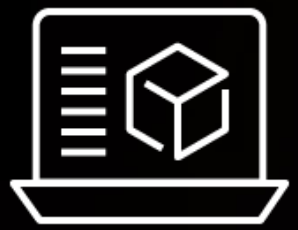


For more serverless learning resources, visit:
https://serverlessland.com

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

Continue your AWS Serverless learning

Learn at your own pace



Expand your serverless skills with our learning plan on **AWS Skill Builder**

Increase your knowledge



Use our **Ramp-Up Guides** to build your serverless knowledge

Earn an AWS Serverless badge



Demonstrate your knowledge by achieving **digital badges**

https://s12d.com/serverless-learning

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

Thank you!

Eric Johnson
@edjgeek



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

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