

GOTO x AWS

EDA DAY **WARSAW 2024**

Building next-generation applications with event-driven architecture

Who am I?

- Principal DA – serverless
- Developer for 30 years
- Solutions architect for 10 years
- IaC/automation fan
- Father of 5
- Drummer
- Foodie



Agenda

Coupling

Integration patterns

What is an event?

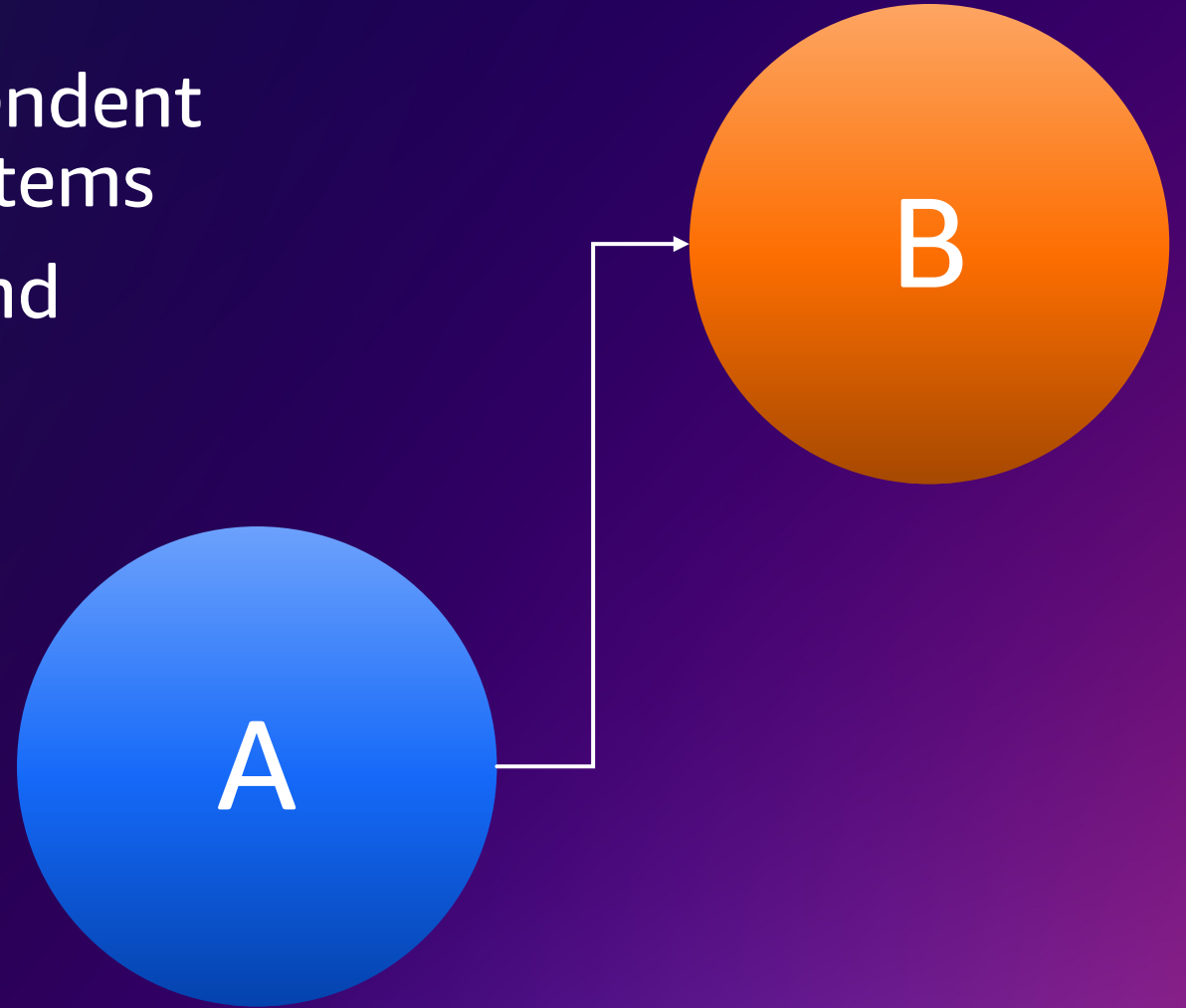
Event-driven architectures

Reducing event duplication
with idempotency

Coupling

Coupling – Integration's magic word

- Coupling is the measure of independent variability between connected systems
- Decoupling has a cost at design and runtime
- Coupling isn't binary
- Coupling isn't one-dimensional



The many facets of coupling

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

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

”

Gregor Hohpe

Co-author of “Enterprise Integration Patterns”

Integration patterns

Synchronous request-response model



Synchronous request-response model

Advantages

- Low latency
- Simple
- Fail fast



Synchronous request-response model

Advantages

- Low latency
- Simple
- Fail fast

Disadvantages

- Receiver failure



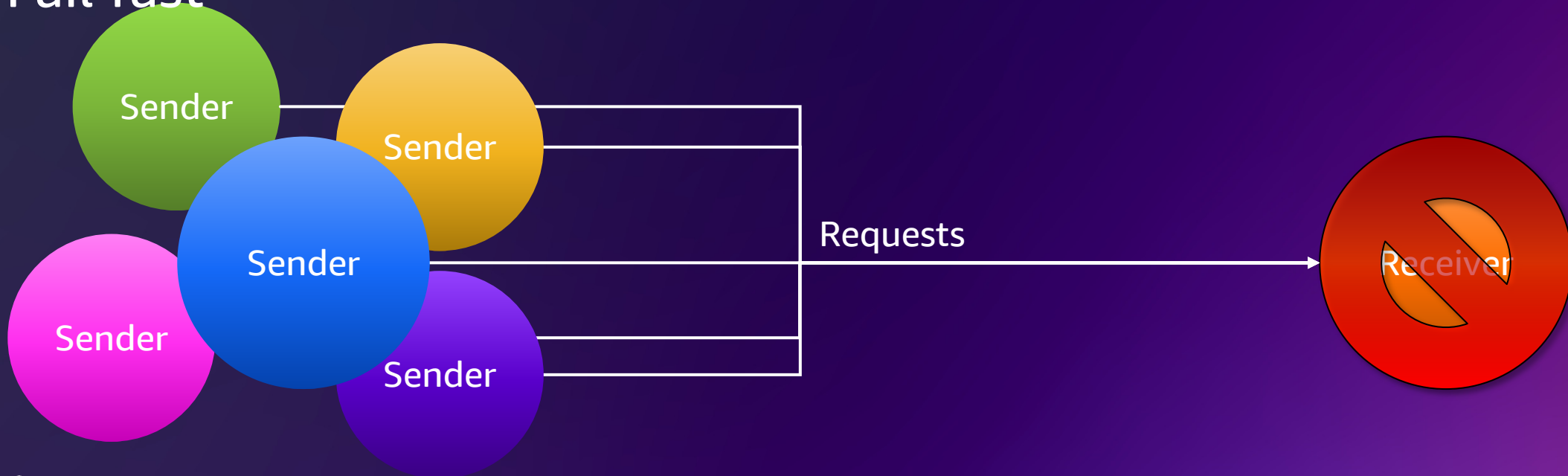
Synchronous request-response model

Advantages

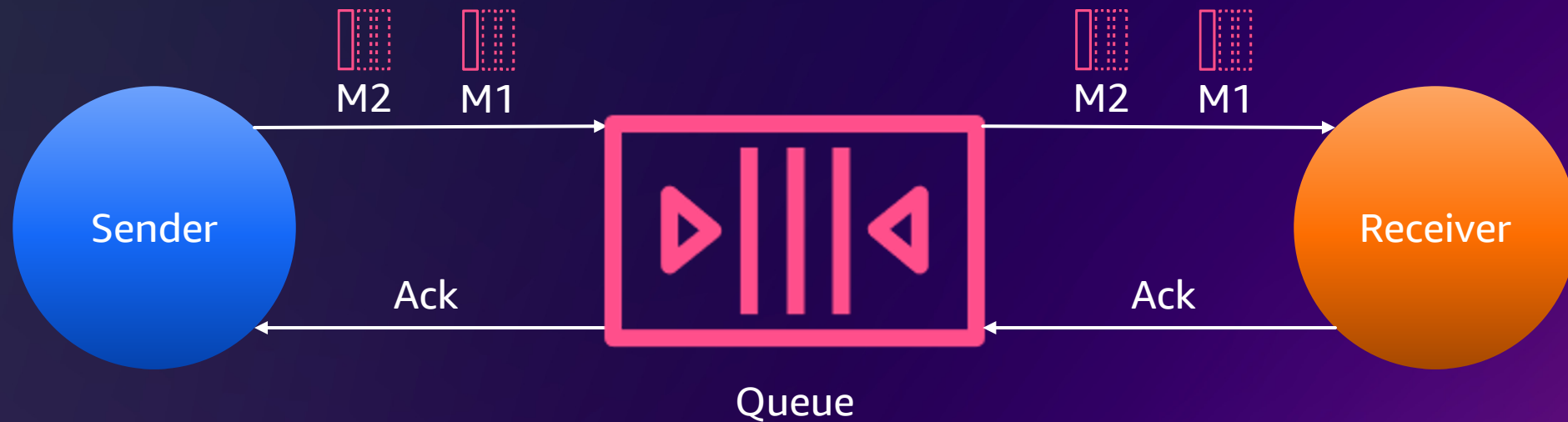
- Low latency
- Simple
- Fail fast

Disadvantages

- Receiver failure
- Receiver throttled



Asynchronous point-to-point model (queue)



Asynchronous point-to-point model (queue)

Advantages

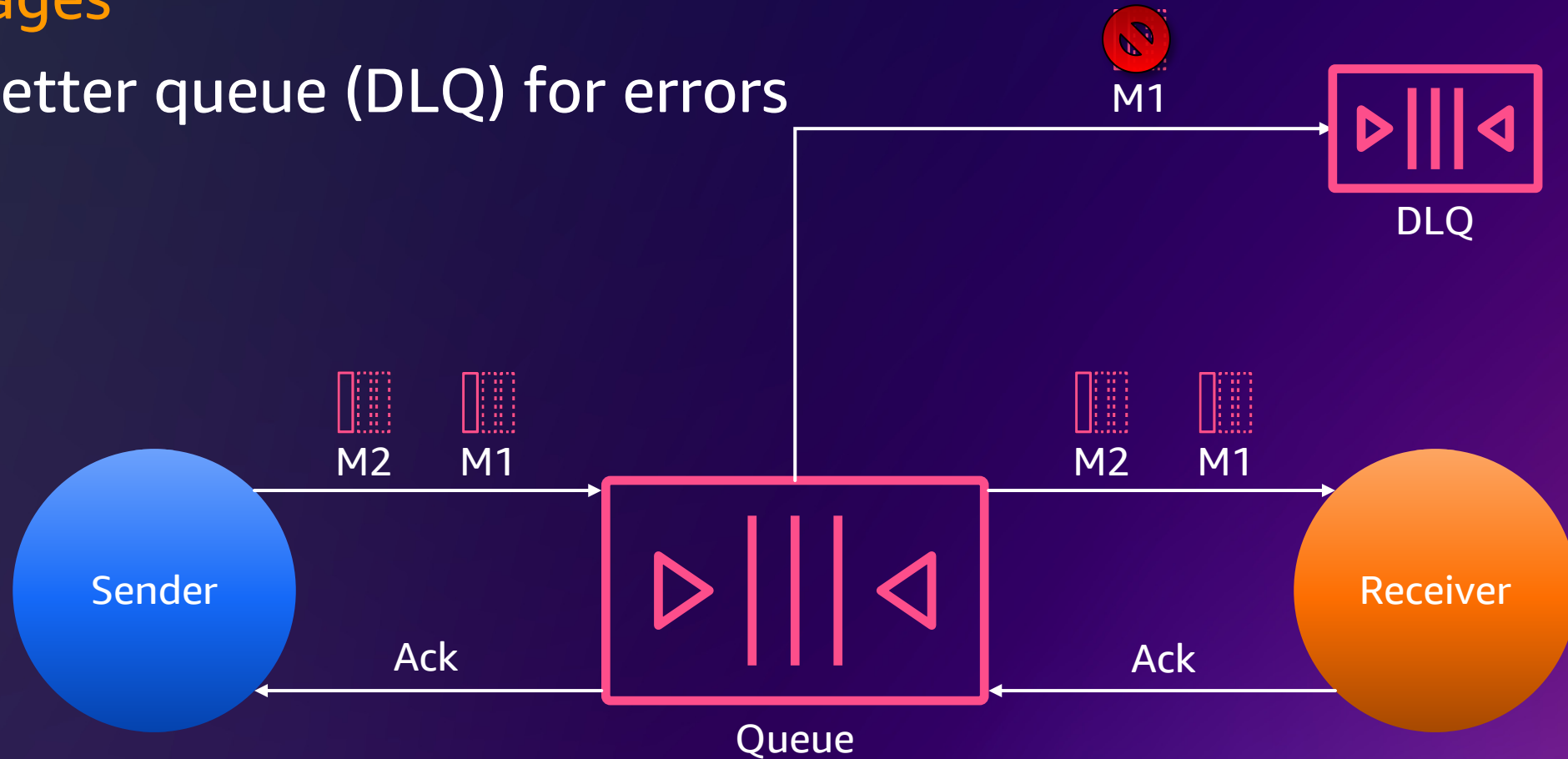
- Decreases **temporal coupling**
- Resilient to receiver failure
- Receiver controls consumption rate



Asynchronous point-to-point model (queue)

Advantages

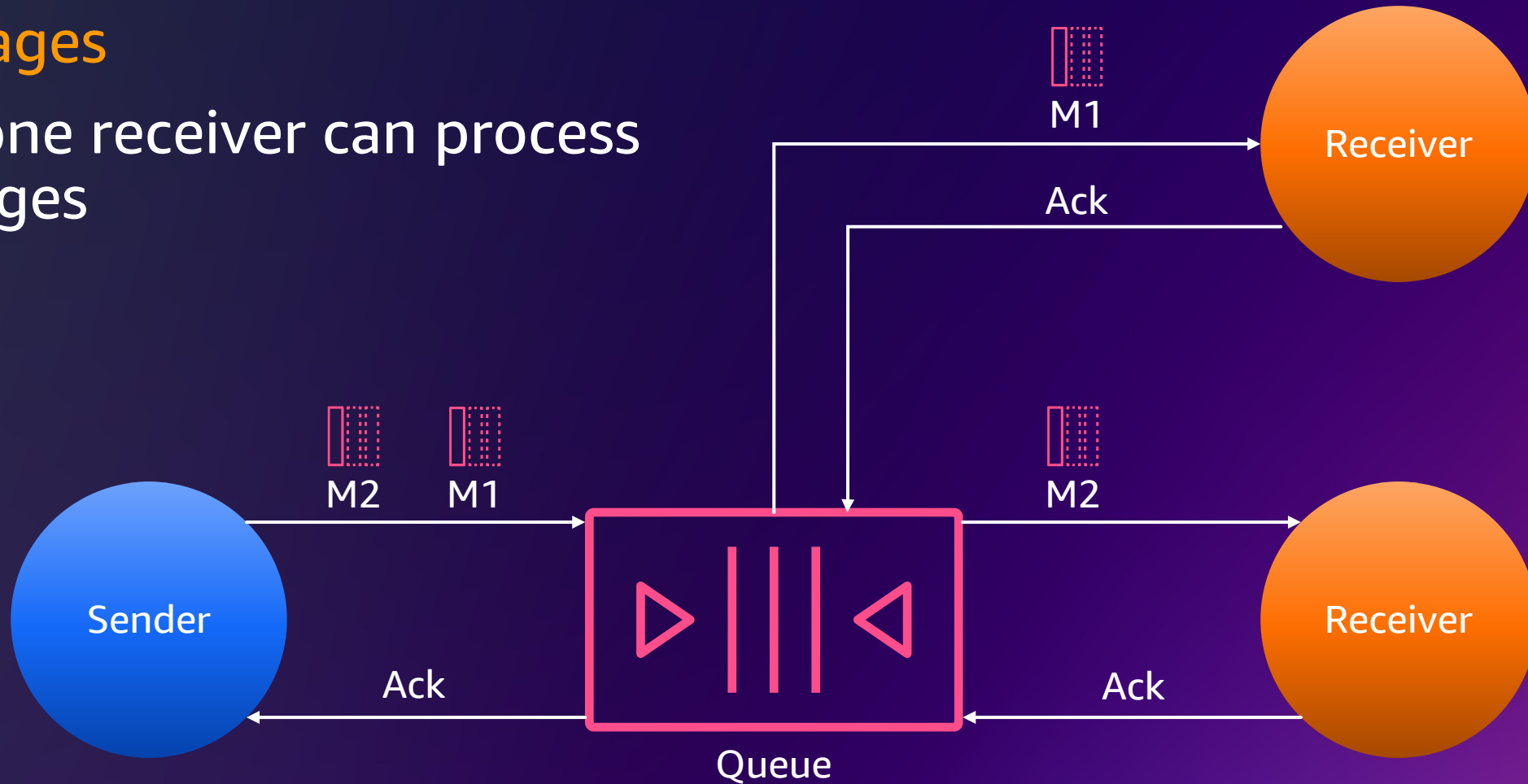
- Dead letter queue (DLQ) for errors



Asynchronous point-to-point model (queue)

Advantages

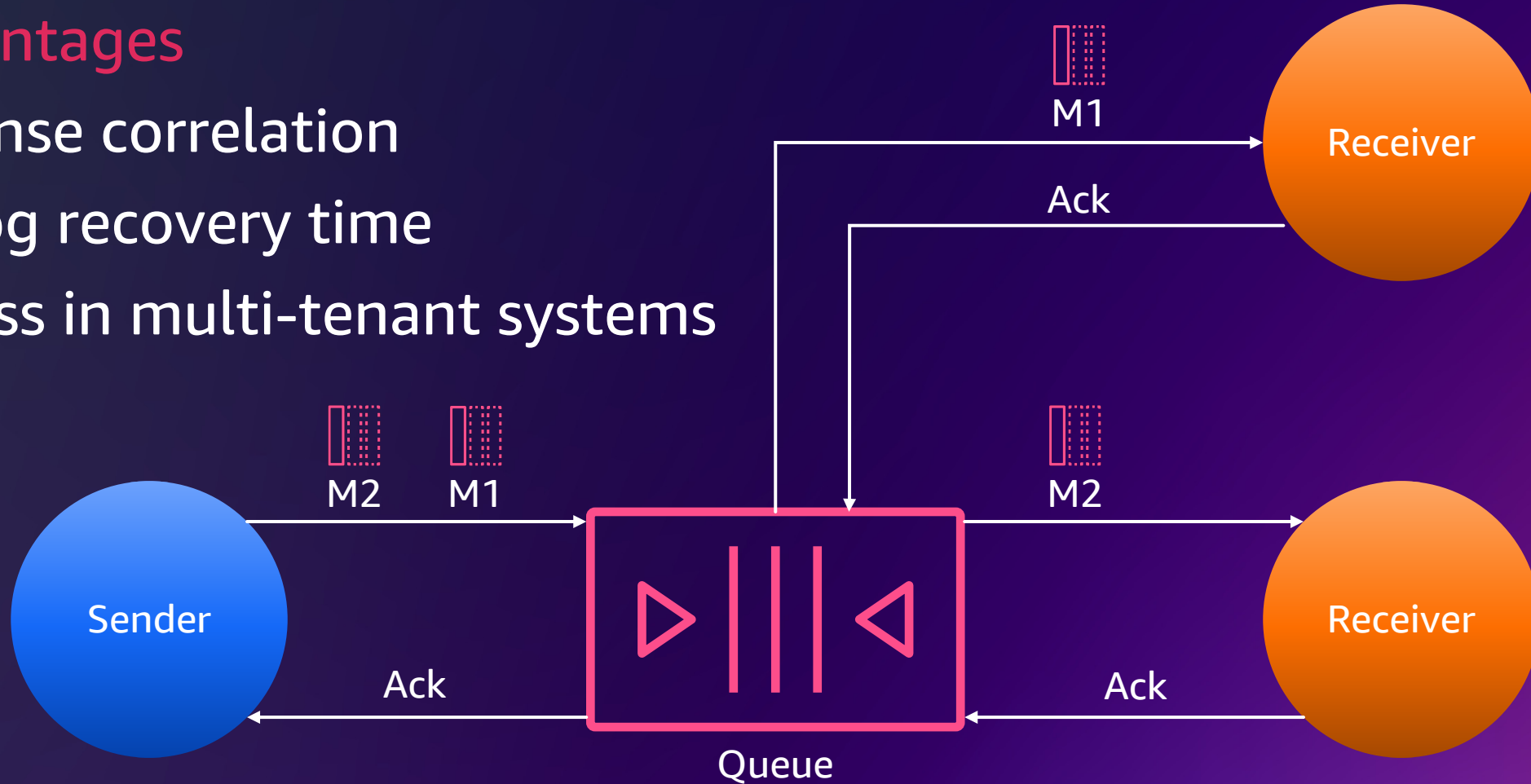
- Only one receiver can process messages



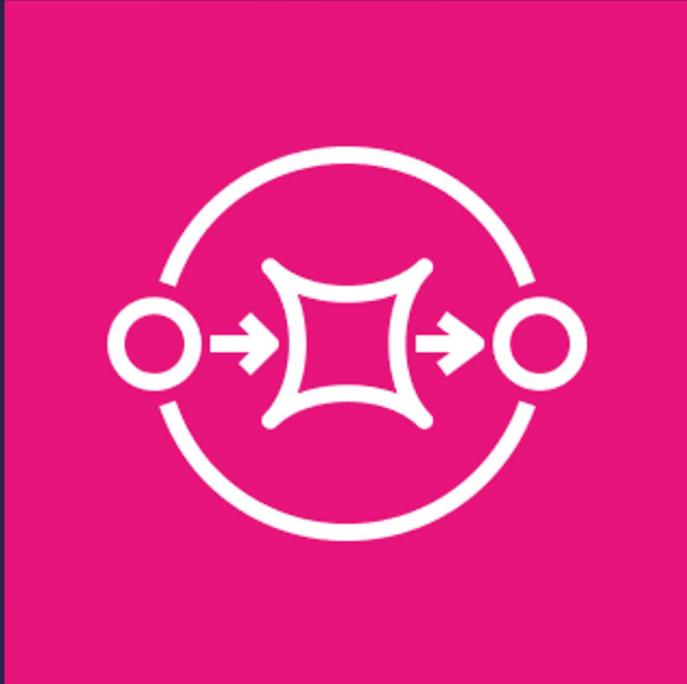
Asynchronous point-to-point model (queue)

Disadvantages

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



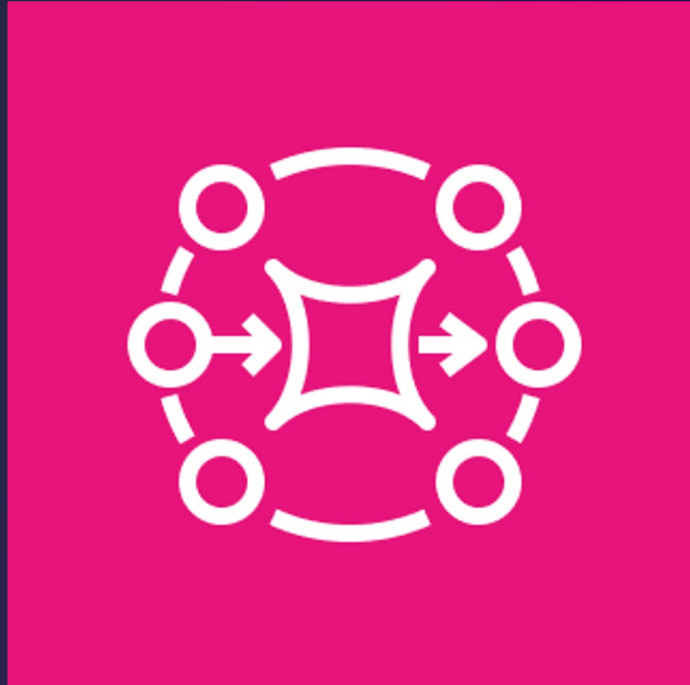
Amazon SQS



Amazon Simple Queue Service
(Amazon SQS)

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

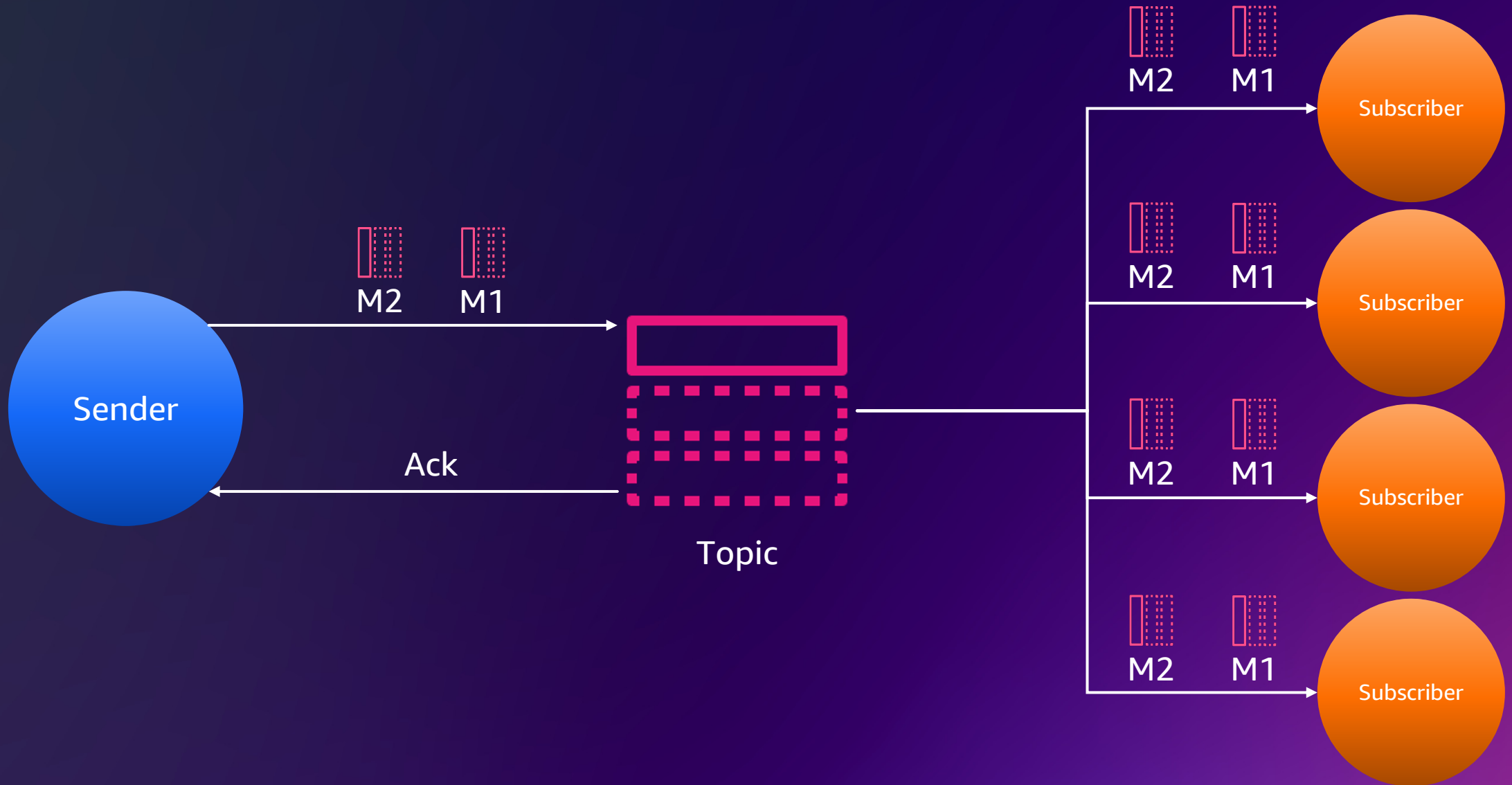
Amazon MQ



Amazon MQ

- ActiveMQ and RabbitMQ broker engine options
- AWS manages the provisioning, setup, and maintenance of message brokers
- Connects to current applications with industry-standard APIs and protocols

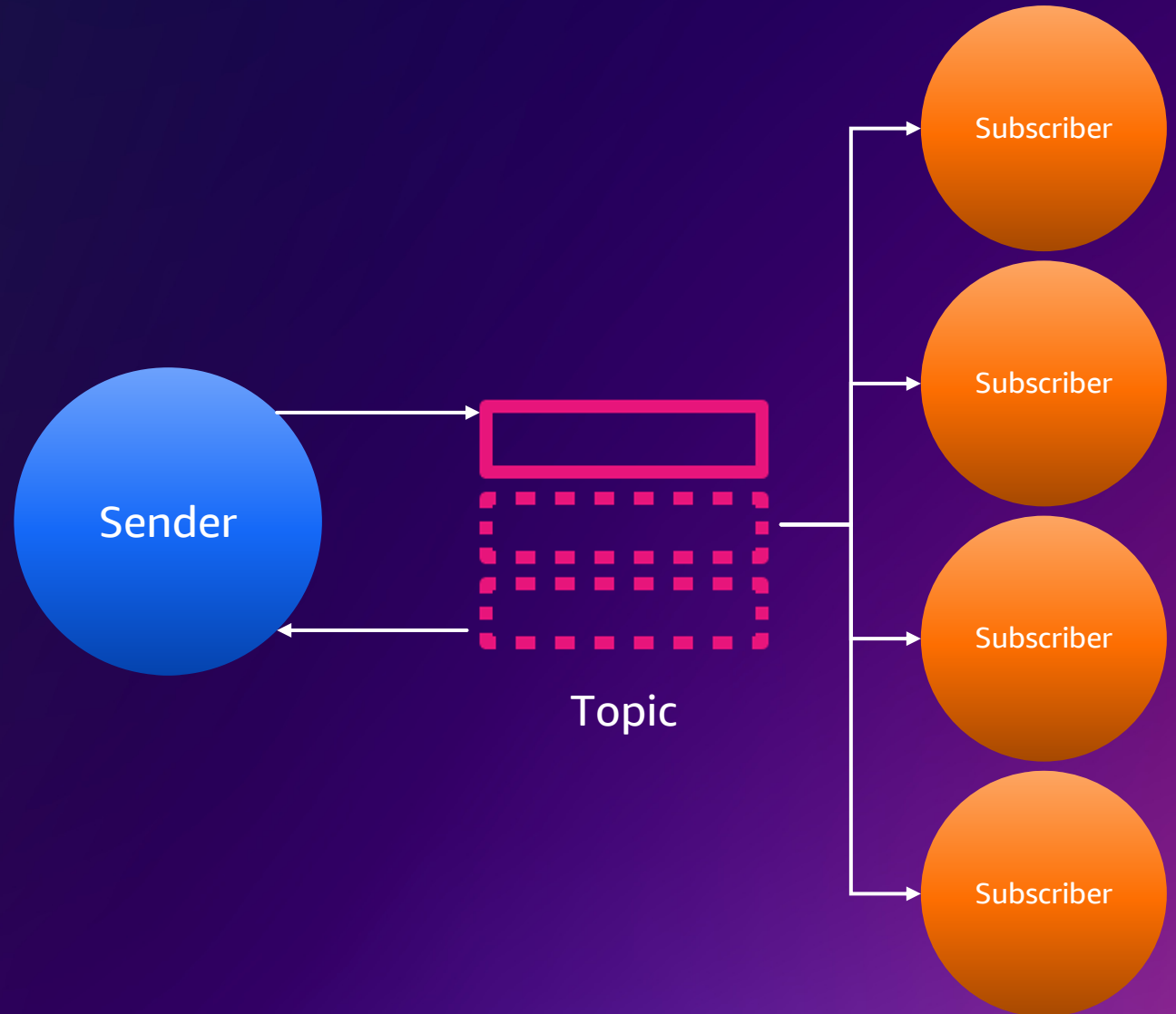
Asynchronous broadcast model (topic)



Asynchronous broadcast model (topic)

Advantages

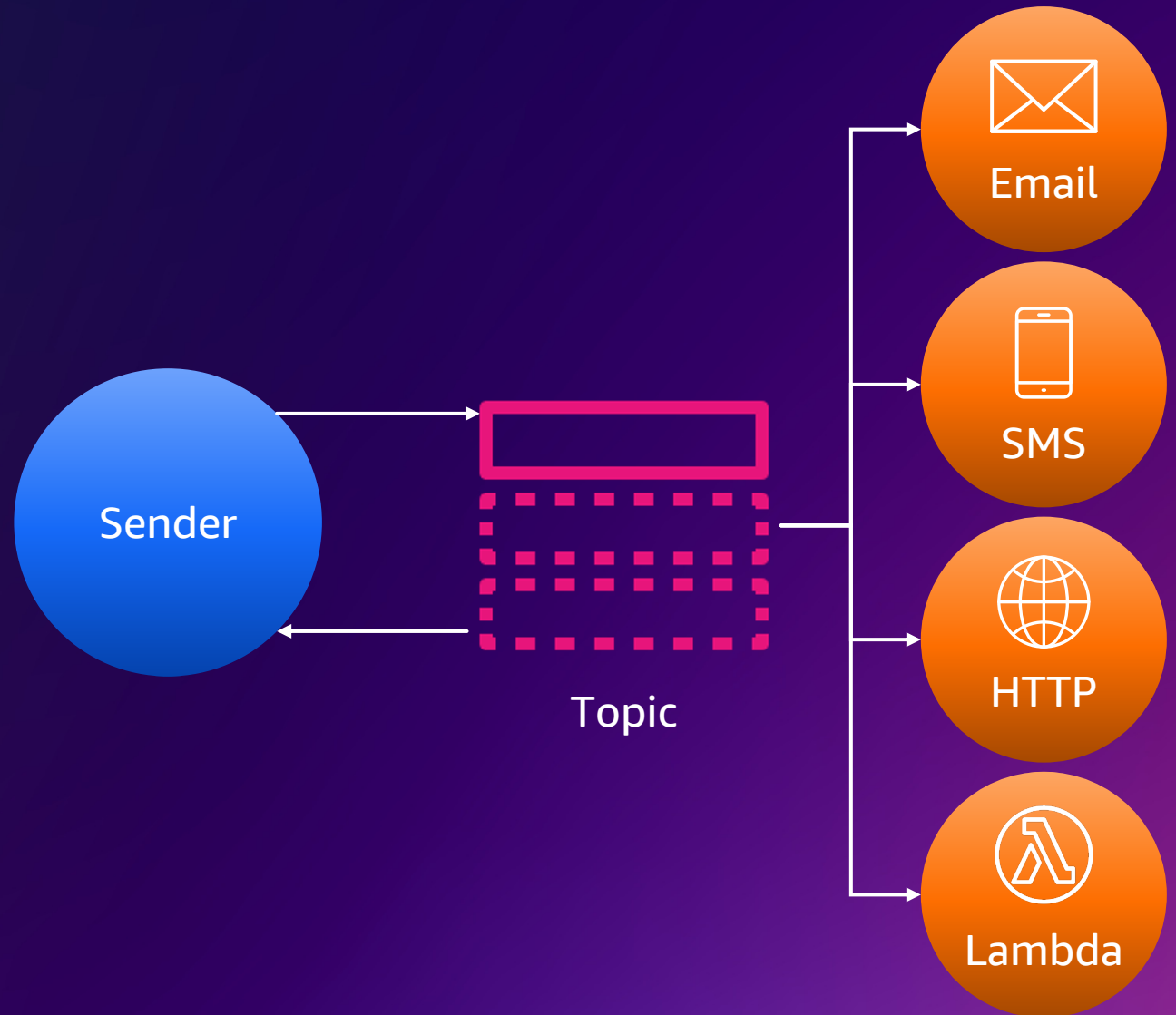
- Decreases **temporal coupling**
- Resilient to receiver failure
- Subscriber controls notification
- Each subscriber is notified



Asynchronous broadcast model (topic)

Advantages

- Decreases **temporal coupling**
- Resilient to receiver failure
- Subscriber controls notification
- Each subscriber is notified
- Multiple protocols are supported



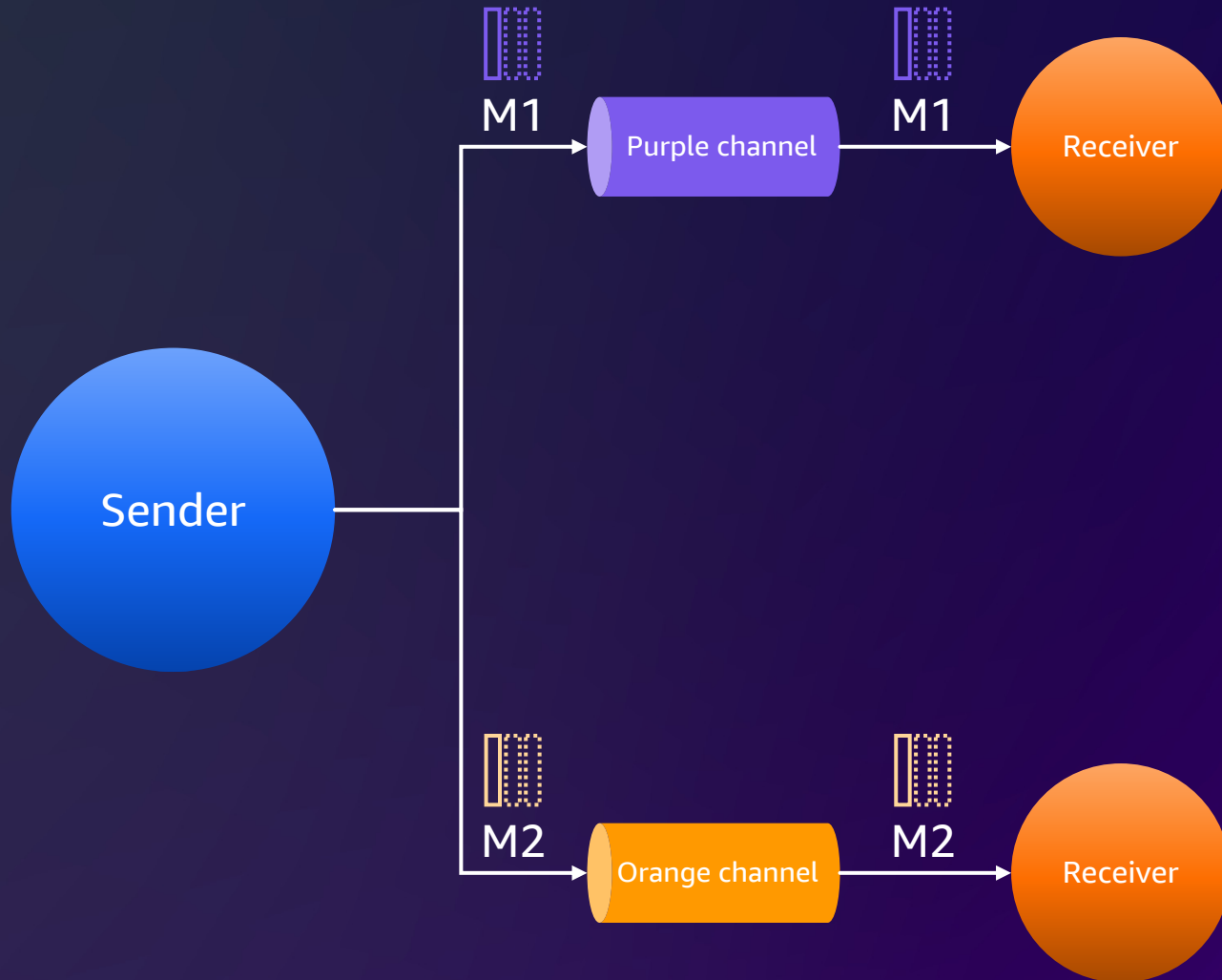
Amazon SNS



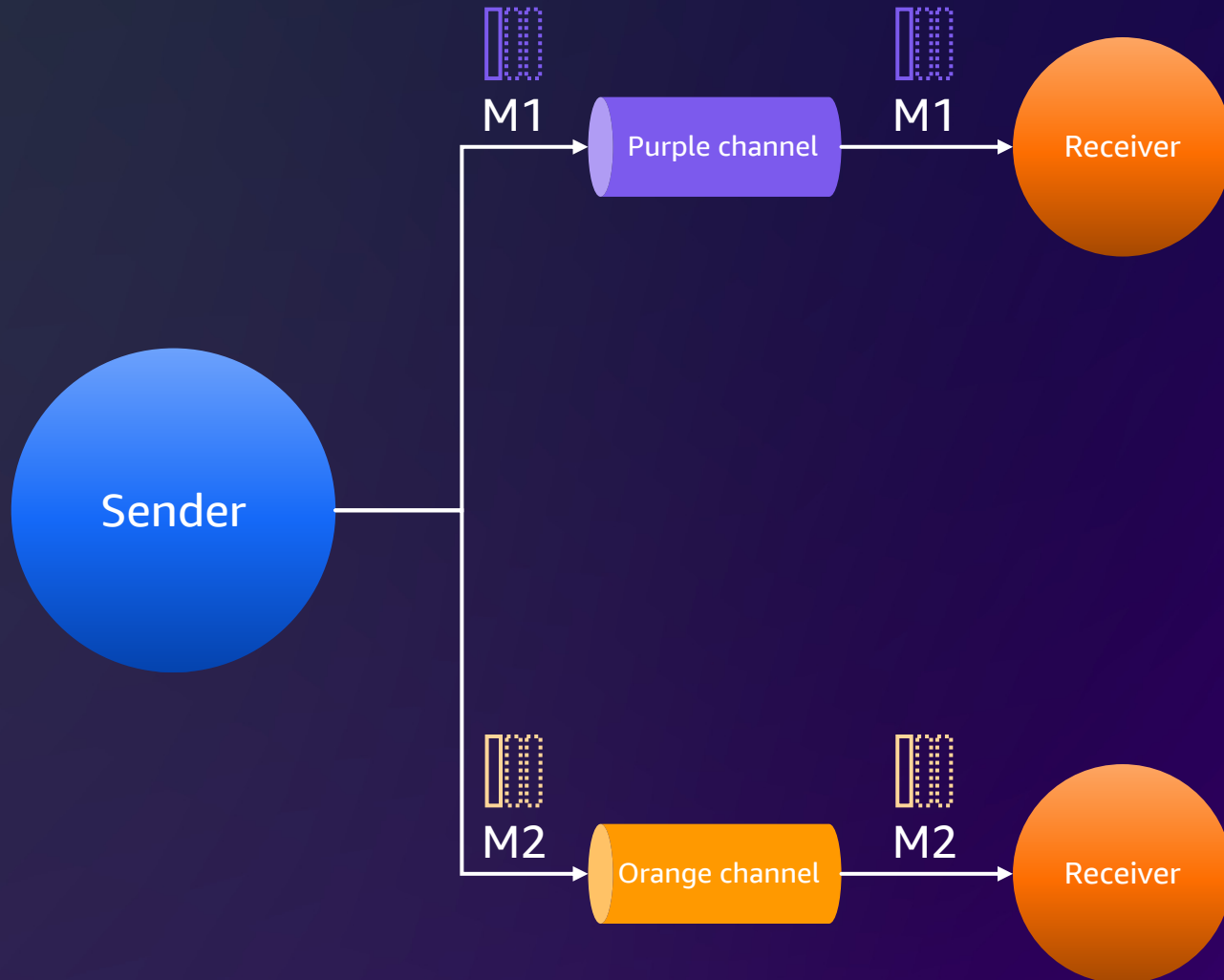
Amazon Simple Notification Service
(Amazon SNS)

- Fully managed pub/sub service for A2A and A2P messaging
- Standard topics support a nearly unlimited number of messages per second
- Each topic supports up to 12.5 million subscriptions
- FIFO topics when strict ordering is required

Asynchronous point-to-point model (router)



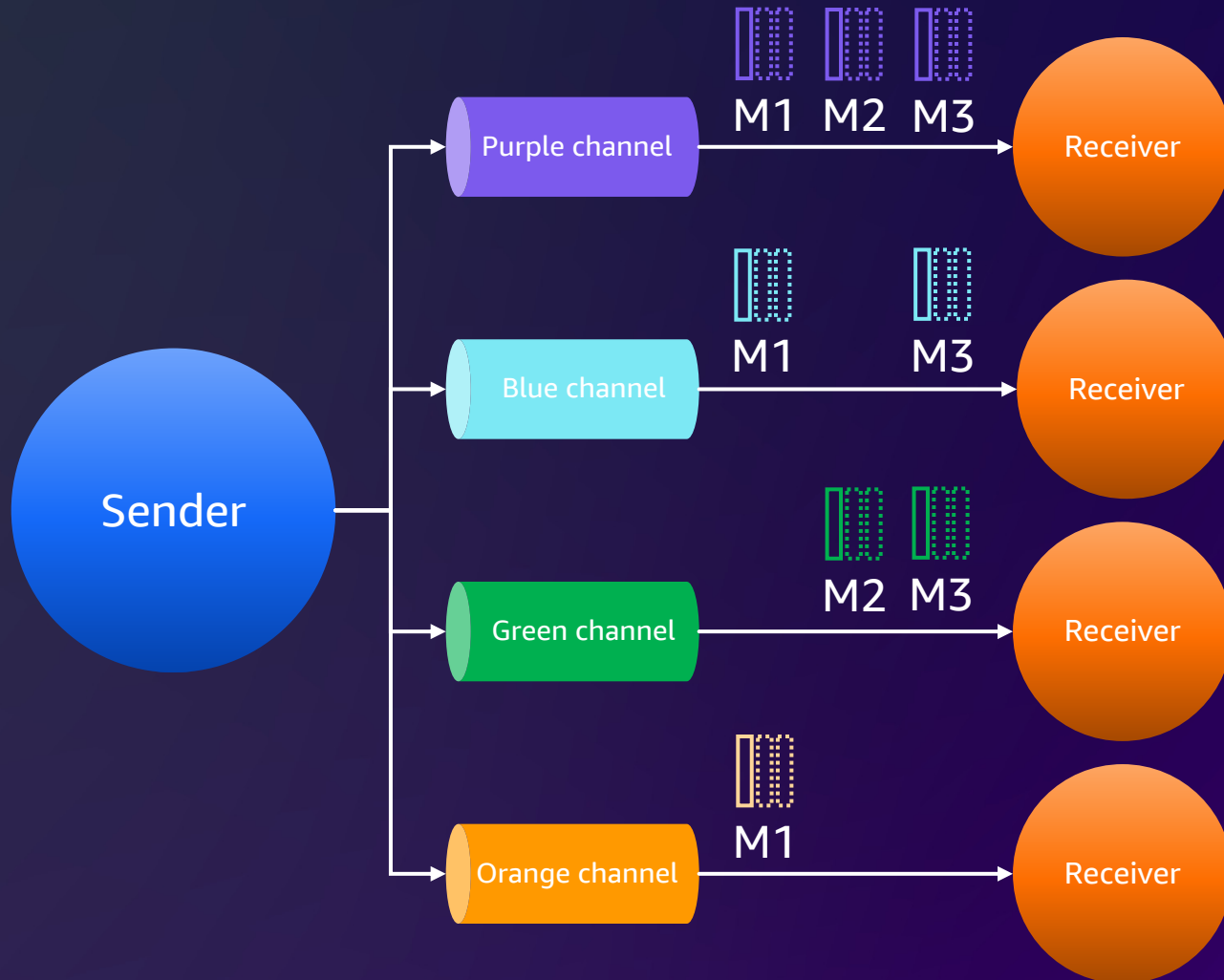
Asynchronous point-to-point model (router)



Disadvantages

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

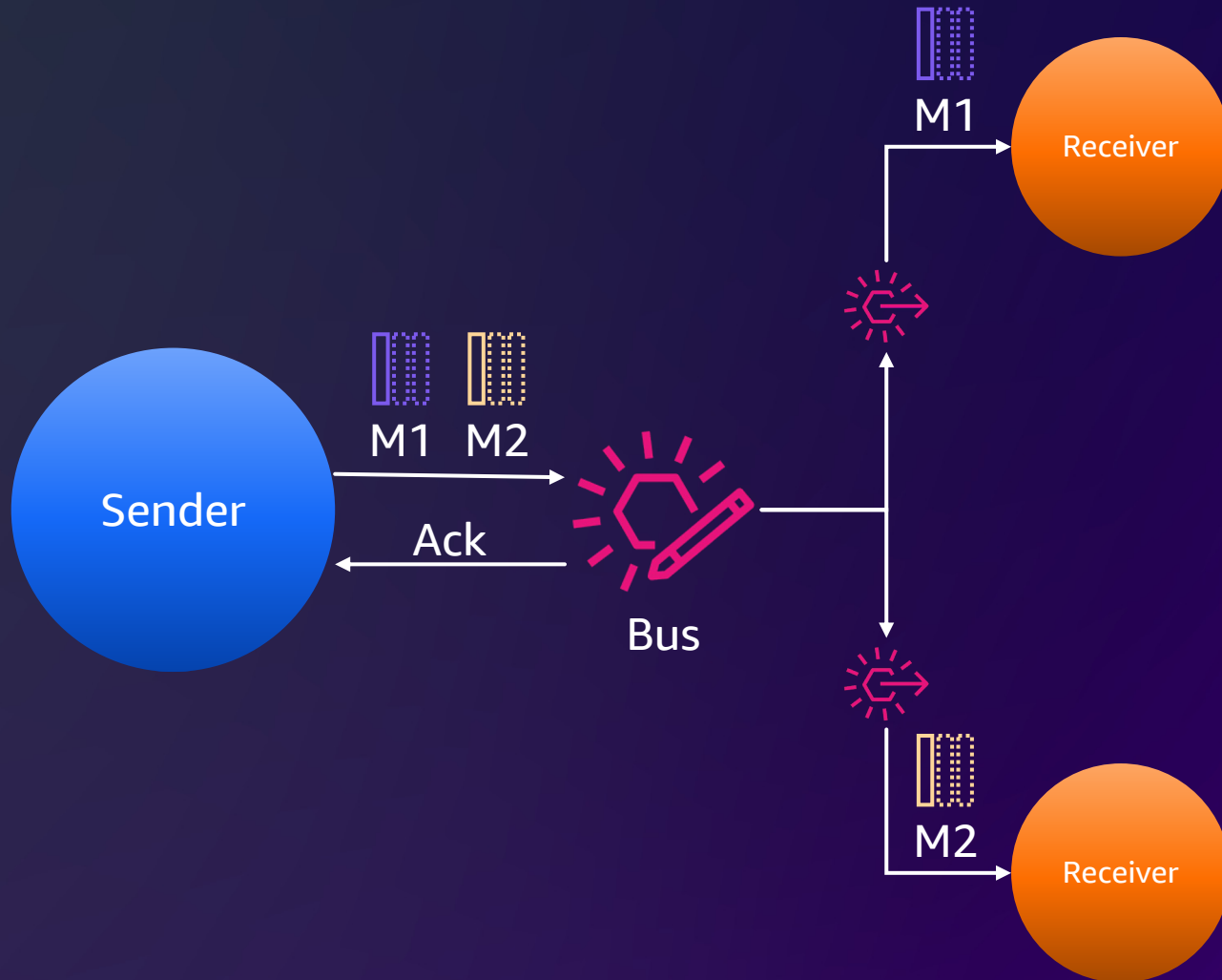
Asynchronous point-to-point model (router)



Disadvantages

- Increases **location coupling**
- Sender maintains routing logic
- Increased complexity over time

Asynchronous message-router model (bus)



Advantages

- Reduces **location coupling**
- Efficient for sender and receiver

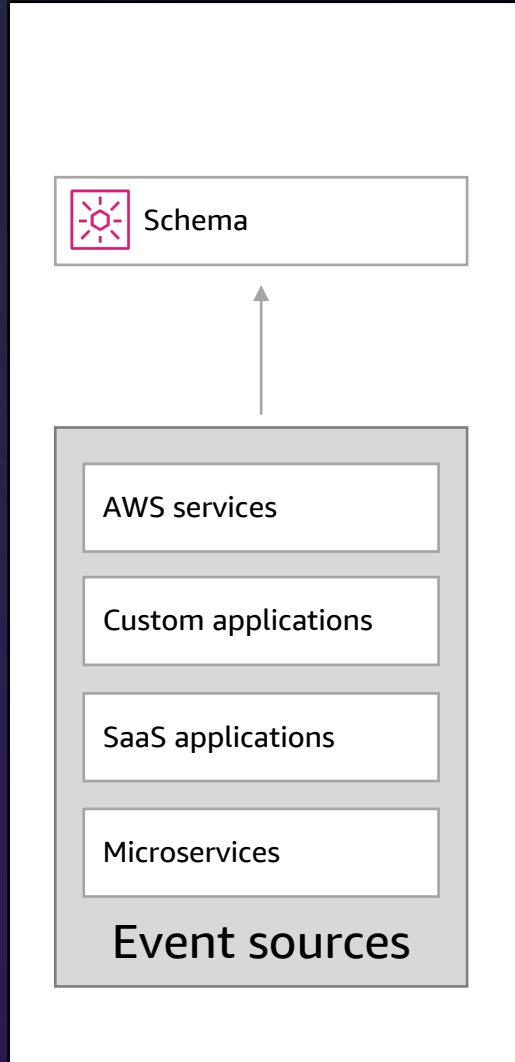
Amazon EventBridge



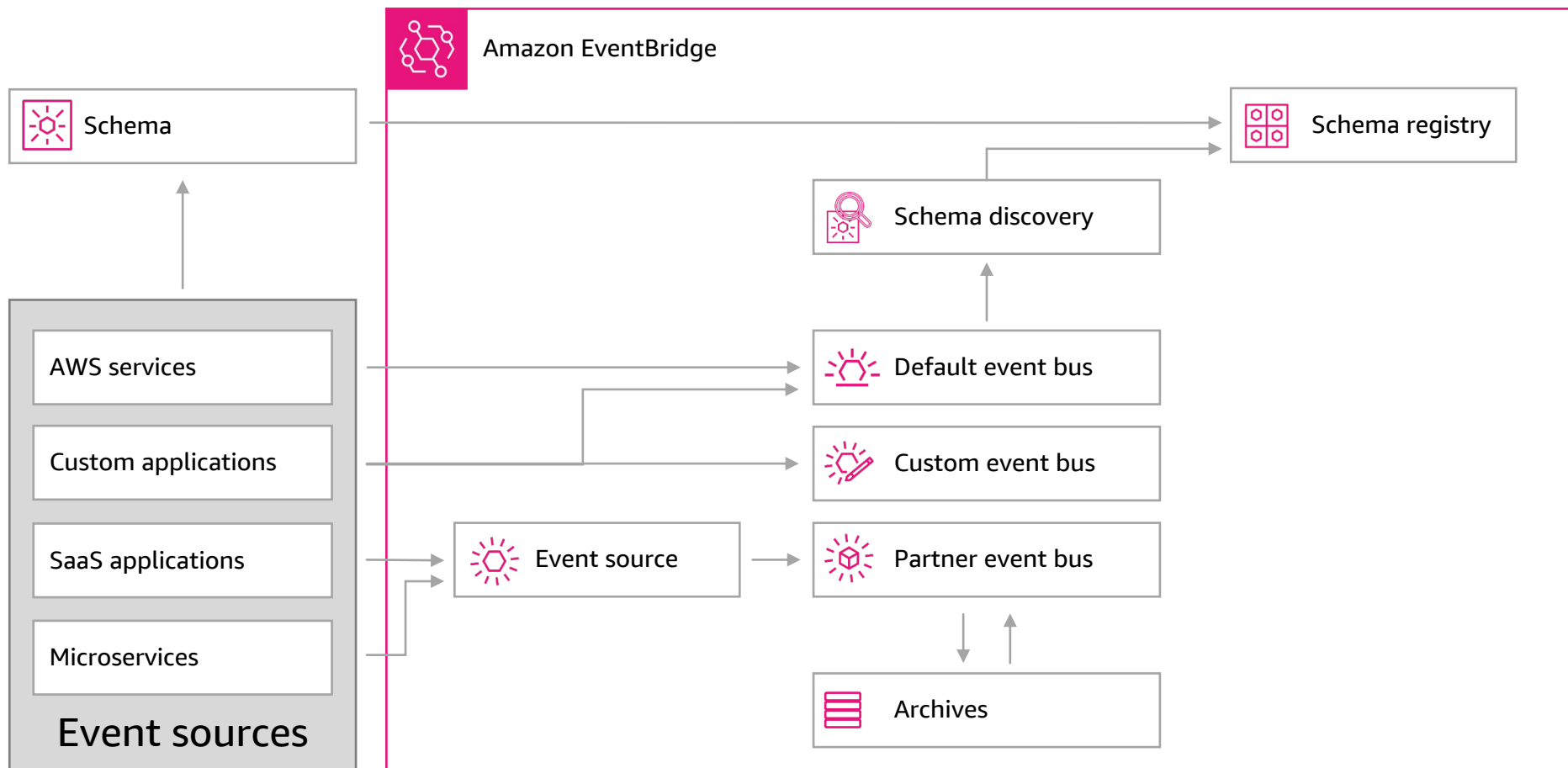
Amazon EventBridge

Amazon 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**

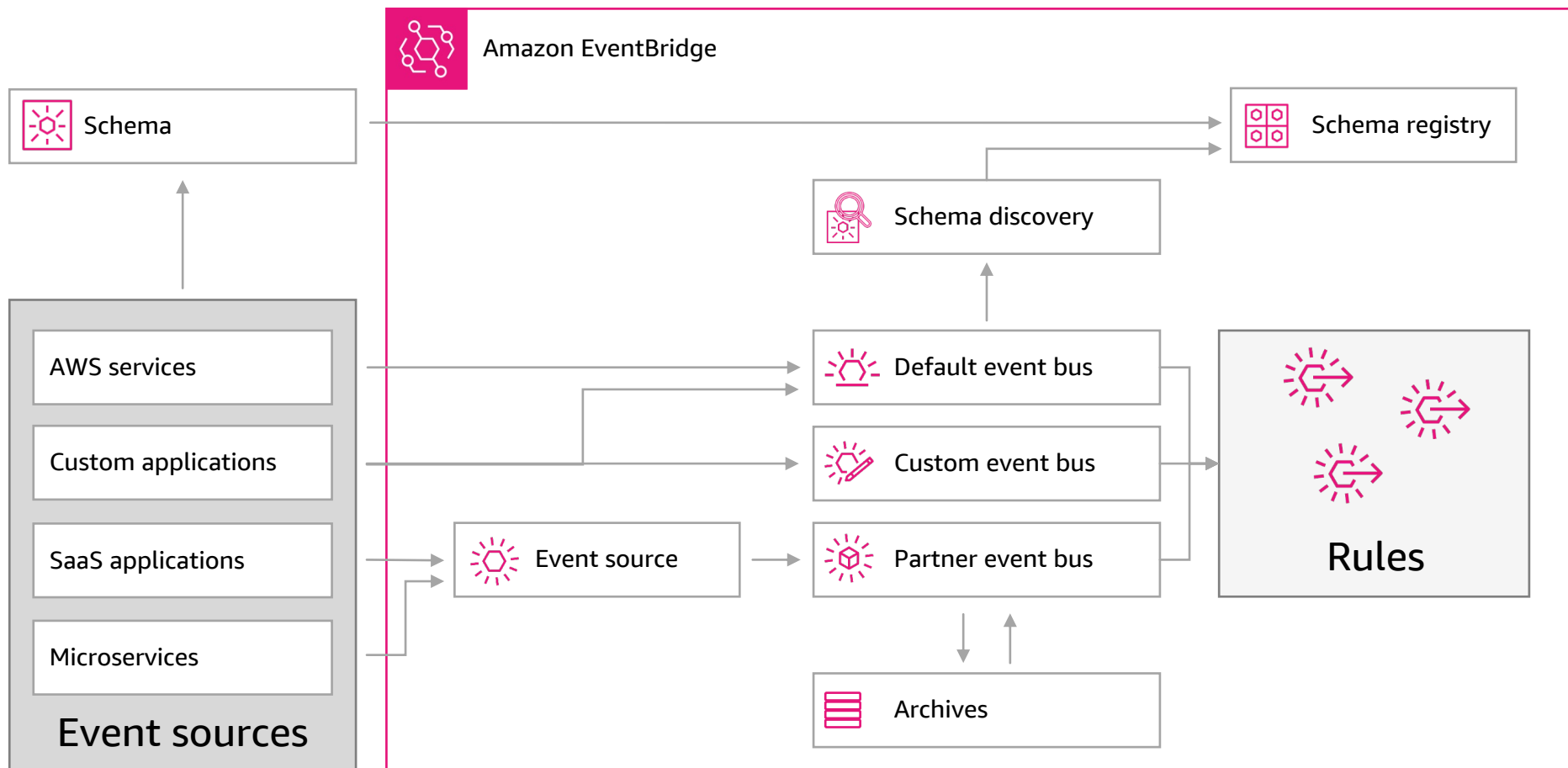
Amazon EventBridge



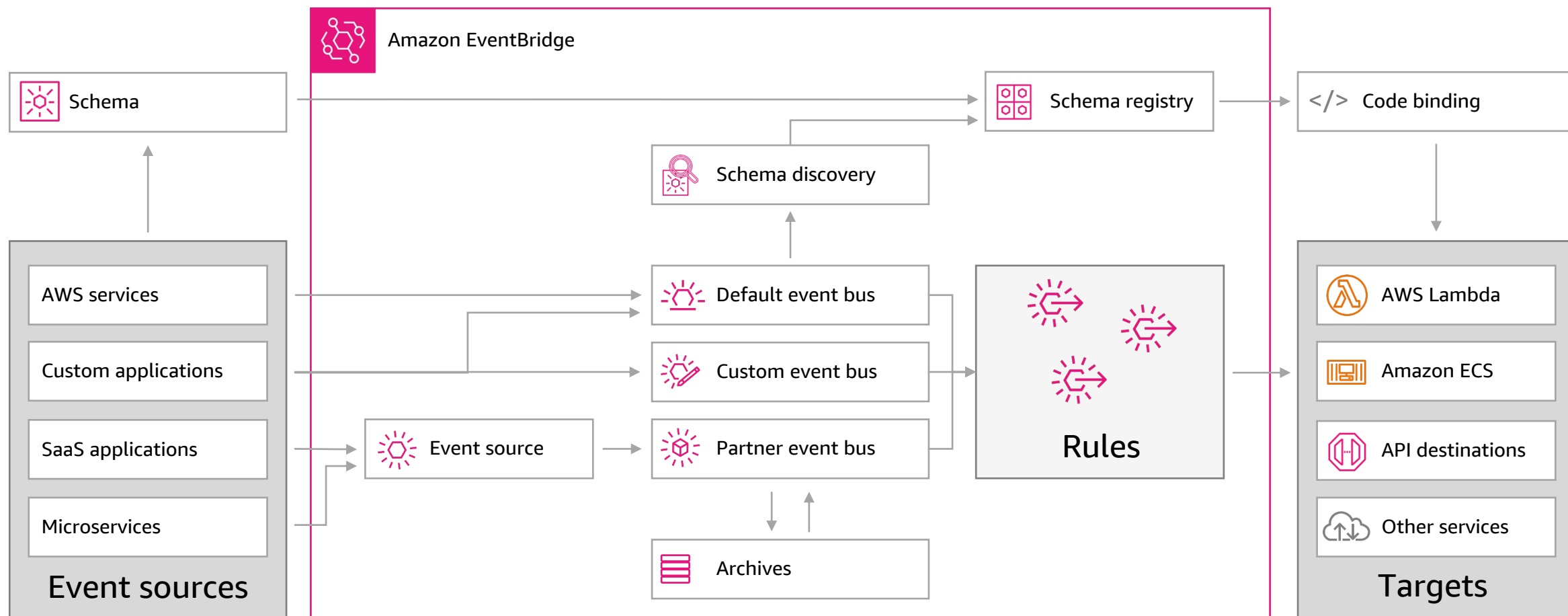
Amazon EventBridge



Amazon EventBridge



Amazon EventBridge



EventBridge content-based routing rules

EventBridge example event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

EventBridge content-based routing rules

EventBridge event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

EventBridge rule

```
{
  "source": ["com.flix"]
  "detail": {
    "data": {
      "region": ["AU", "NZ"]
    }
  }
}
```

EventBridge content-based routing rules

EventBridge event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

EventBridge rule

```
{
  "source": ["com.flix"]
  "detail": {
    "data": {
      "region": ["AU", "NZ"]
    }
  }
}
```

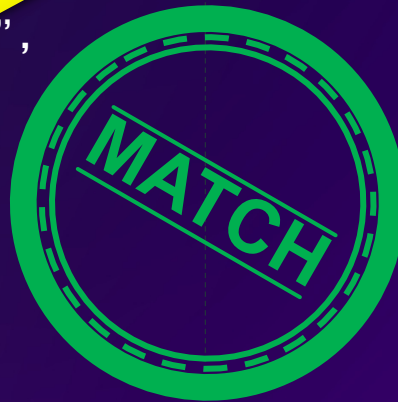
EventBridge content-based routing rules

EventBridge event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

EventBridge rule

```
{
  "source": ["com.flix"]
  "detail": {
    "data": {
      "region": ["AU", "NZ"]
    }
  }
}
```



Target over 30 services



AWS Lambda



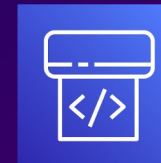
AWS Step Functions



Amazon Kinesis
Data Streams



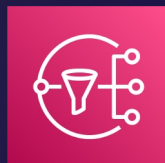
Amazon CloudWatch



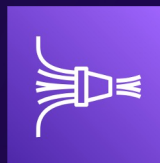
AWS CodePipeline



Amazon ECS



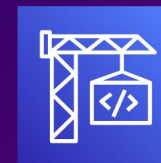
Amazon SNS



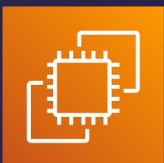
Amazon Kinesis
Data Firehose



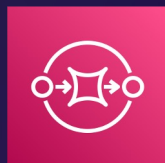
AWS Systems Manager



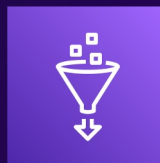
AWS CodeBuild



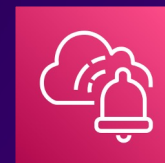
Amazon EC2



Amazon SQS



AWS Glue



Incident Manager



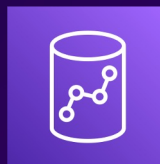
Amazon SageMaker



AWS Batch



Amazon API Gateway



Amazon Redshift



Amazon Inspector



API destinations

What is an event?

Properties of an event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

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

Sparse events vs. full-state events

Sparse event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
    }
  }
}
```

Full-state event

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

Considerations with sparse events

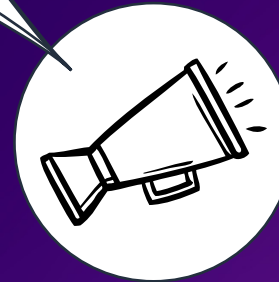
```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
    }
  }
}
```

What Region
are they in?



Video streaming
service

What are the
channel tags?



Video manager
service

What's the name
of the channel?



Publisher
service

Considerations with full-state description

```
{
  "source": "com.flix",
  "detail-type": "ChannelCreated",
  "detail": {
    "metadata": {
      "idempotency-key": "837uy4erje"
    },
    "data": {
      "channel-id": "983u4ejrhewio9039oi4kerj",
      "created-at": "2021-11-26T16:05:09-04:00",
      "name": "noPineappleOnPizza",
      "description": "All about real pizza",
      "Tags": ["Food", "Proper Pizza"],
      "city": "Melbourne",
      "region": "AU",
    }
  }
}
```

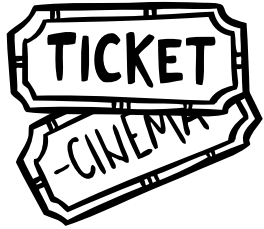
- Event schemas should be backward compatible
- Cost to calculate values can increase over time

Event-driven architectures

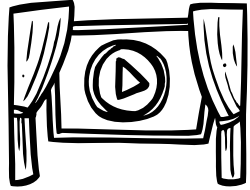
Choreograph events between domains

Choreograph events between domains

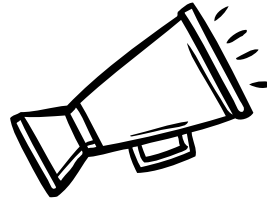
NON-EVENT-DRIVEN PATTERN



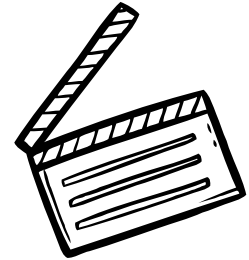
Channel
service



Video streaming
service



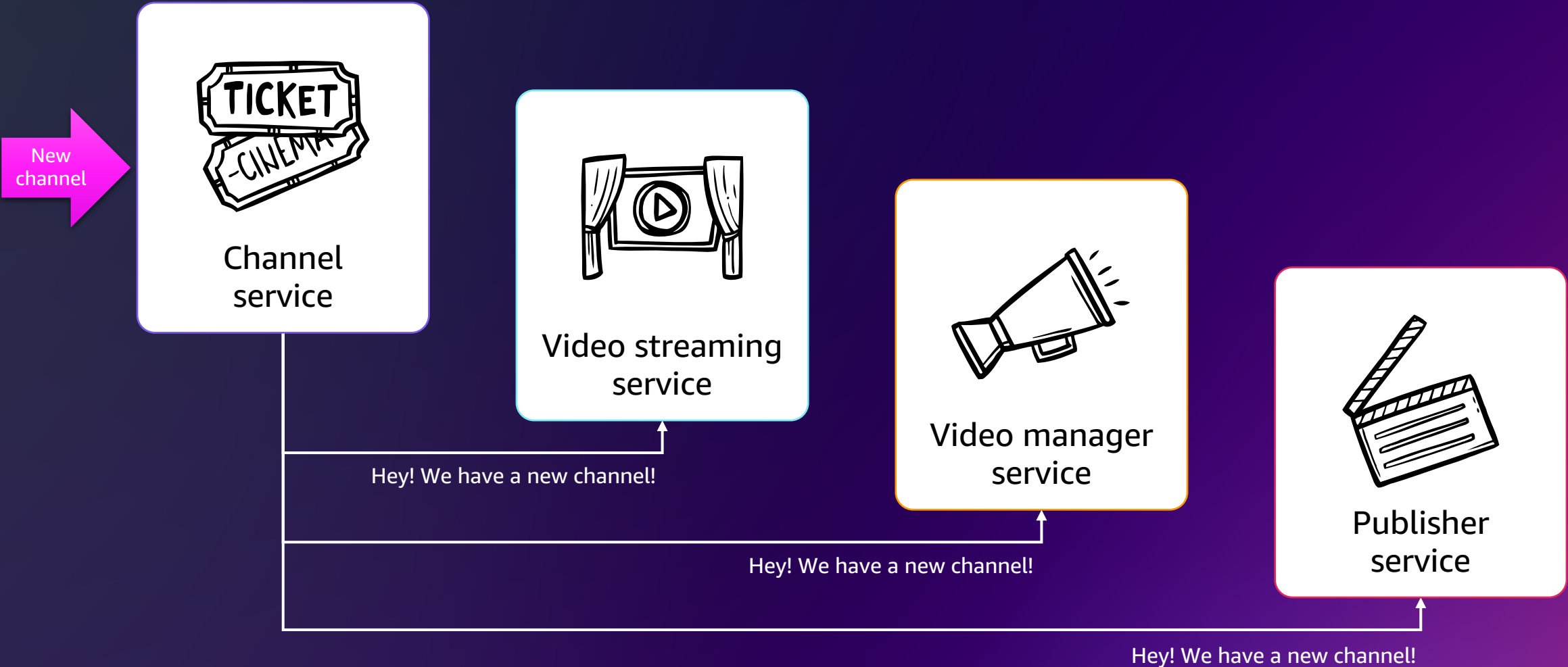
Video manager
service



Publisher
service

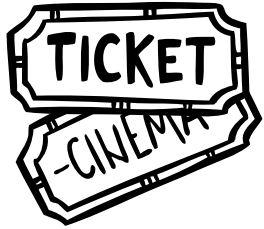
Choreograph events between domains

NON-EVENT-DRIVEN PATTERN

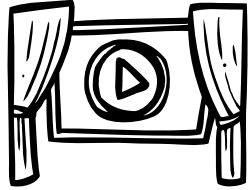


Choreograph events between domains

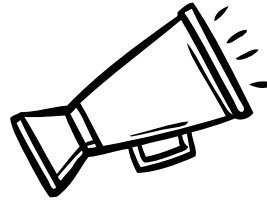
ASYNCHRONOUS BROADCAST MODEL (TOPIC)



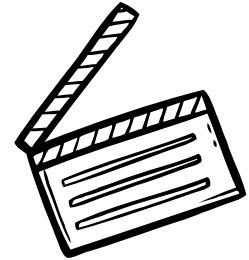
Channel
service



Video streaming
service



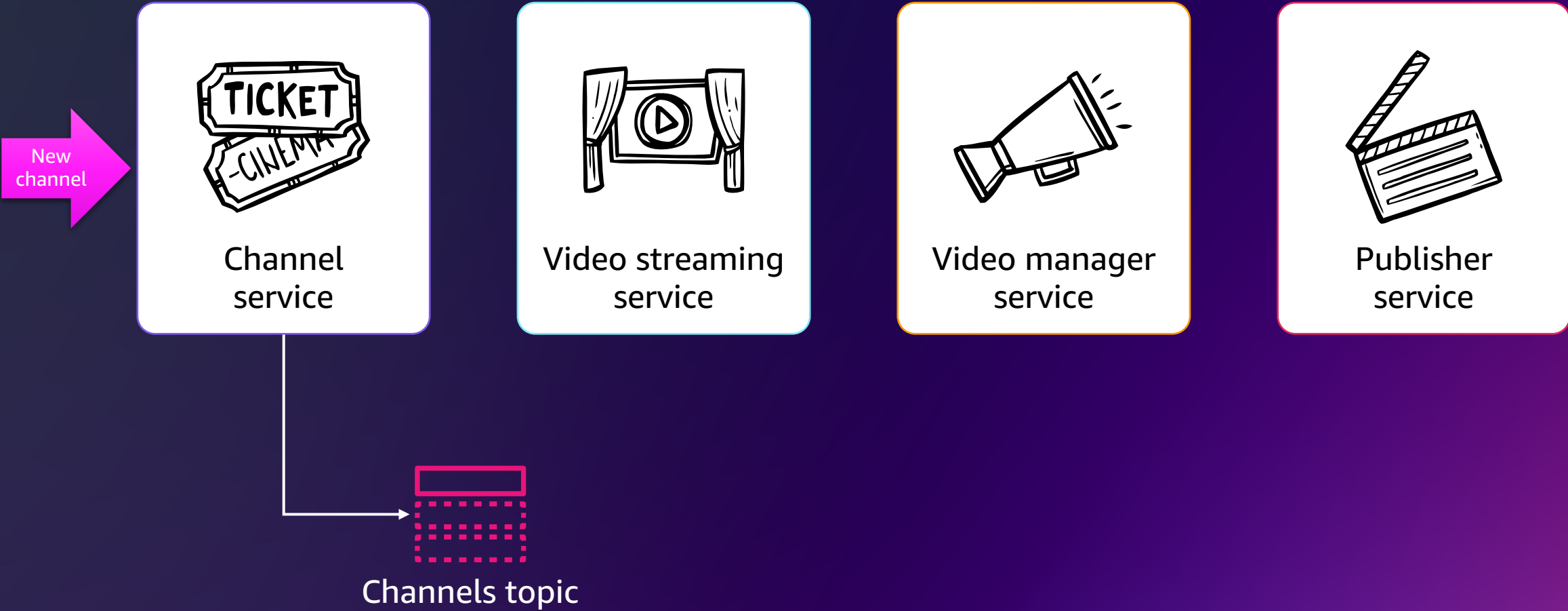
Video manager
service



Publisher
service

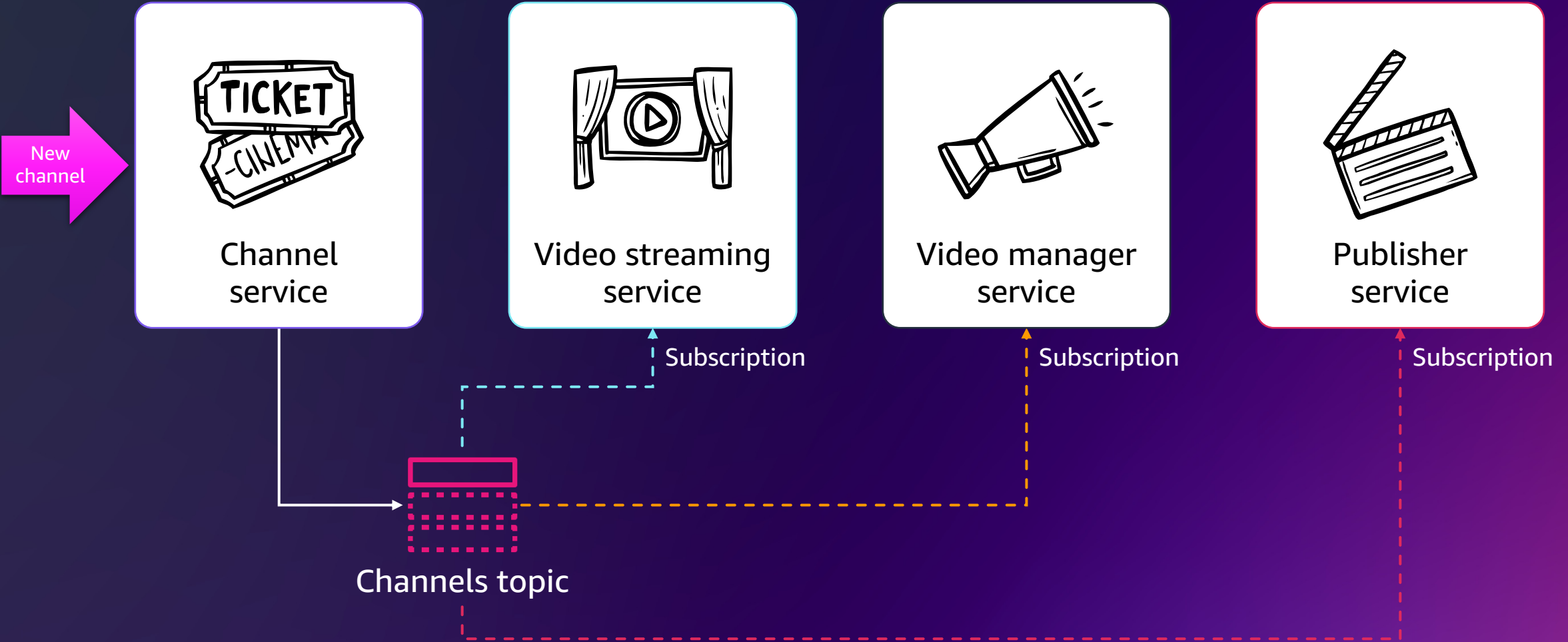
Choreograph events between domains

ASYNCHRONOUS BROADCAST MODEL (TOPIC)



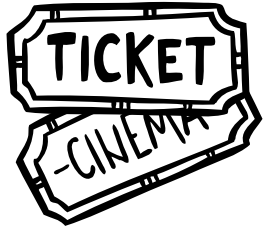
Choreograph events between domains

ASYNCHRONOUS BROADCAST MODEL (TOPIC)

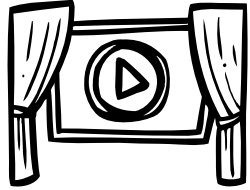


Choreograph events between domains

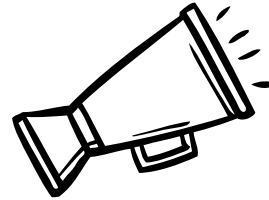
ASYNCHRONOUS MESSAGE-ROUTER MODEL (BUS)



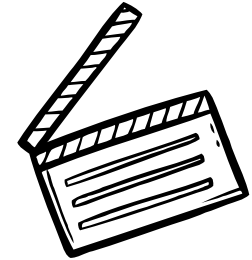
Channel
service



Video streaming
service



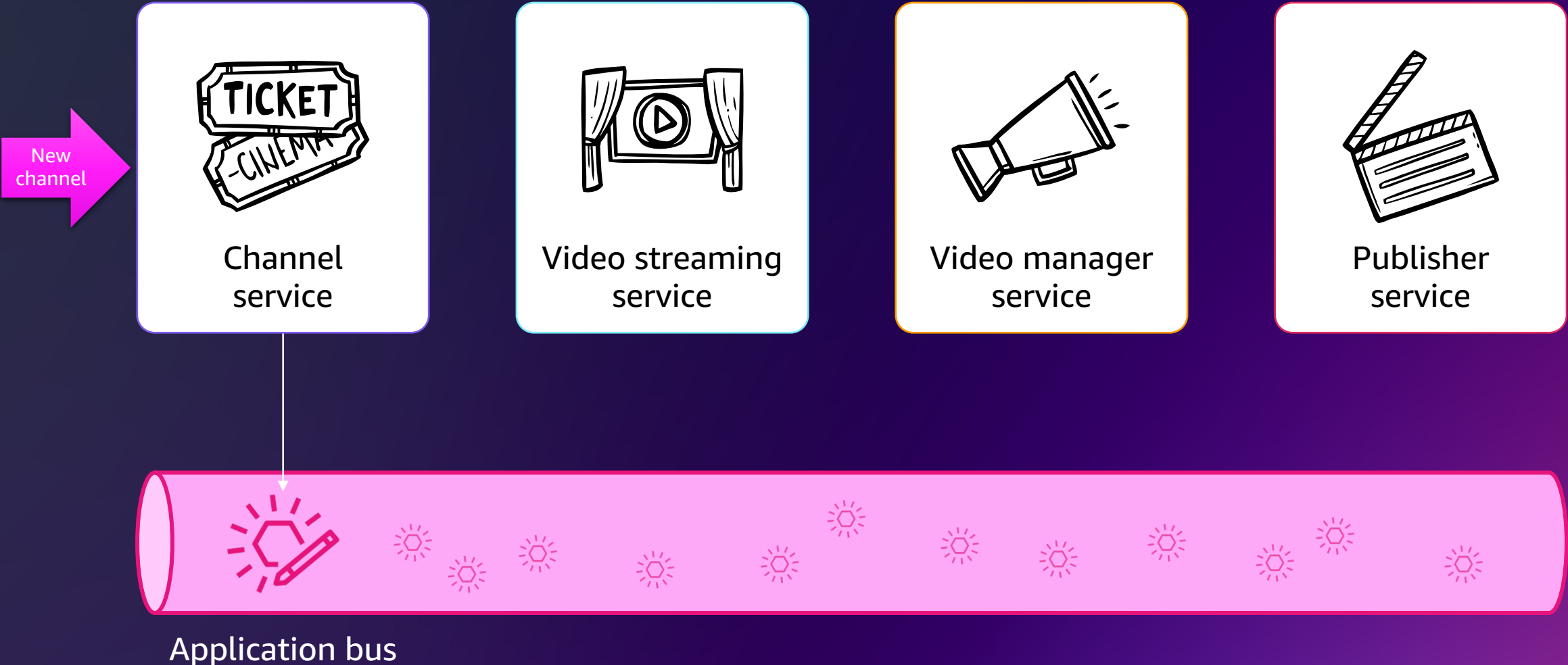
Video manager
service



Publisher
service

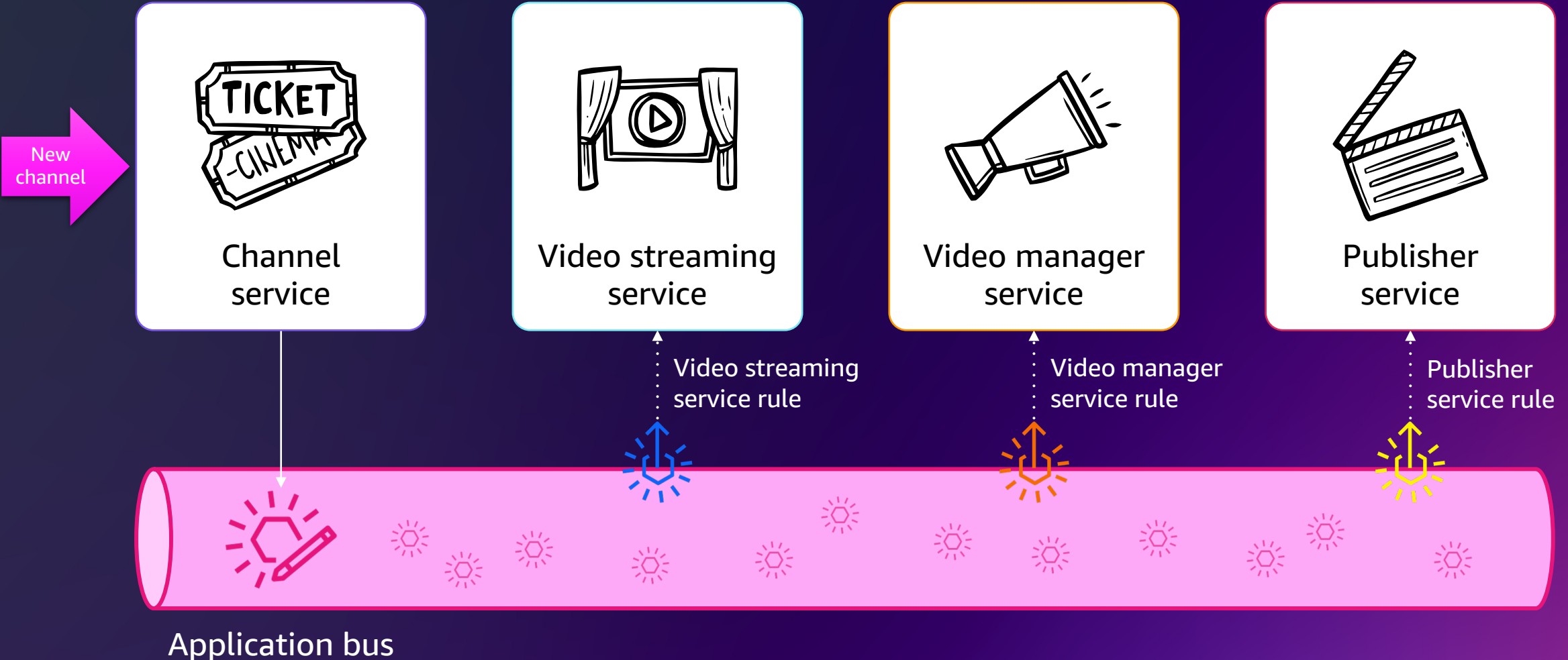
Choreograph events between domains

ASYNCHRONOUS MESSAGE-ROUTER MODEL (BUS)



Choreograph events between domains

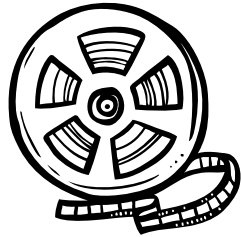
ASYNCHRONOUS MESSAGE-ROUTER MODEL (BUS)



Orchestrate the business process within domains and produce one or more events

Orchestrate the business process within domains

Introducing two more ServerlessVideo services



Video post-
processing service

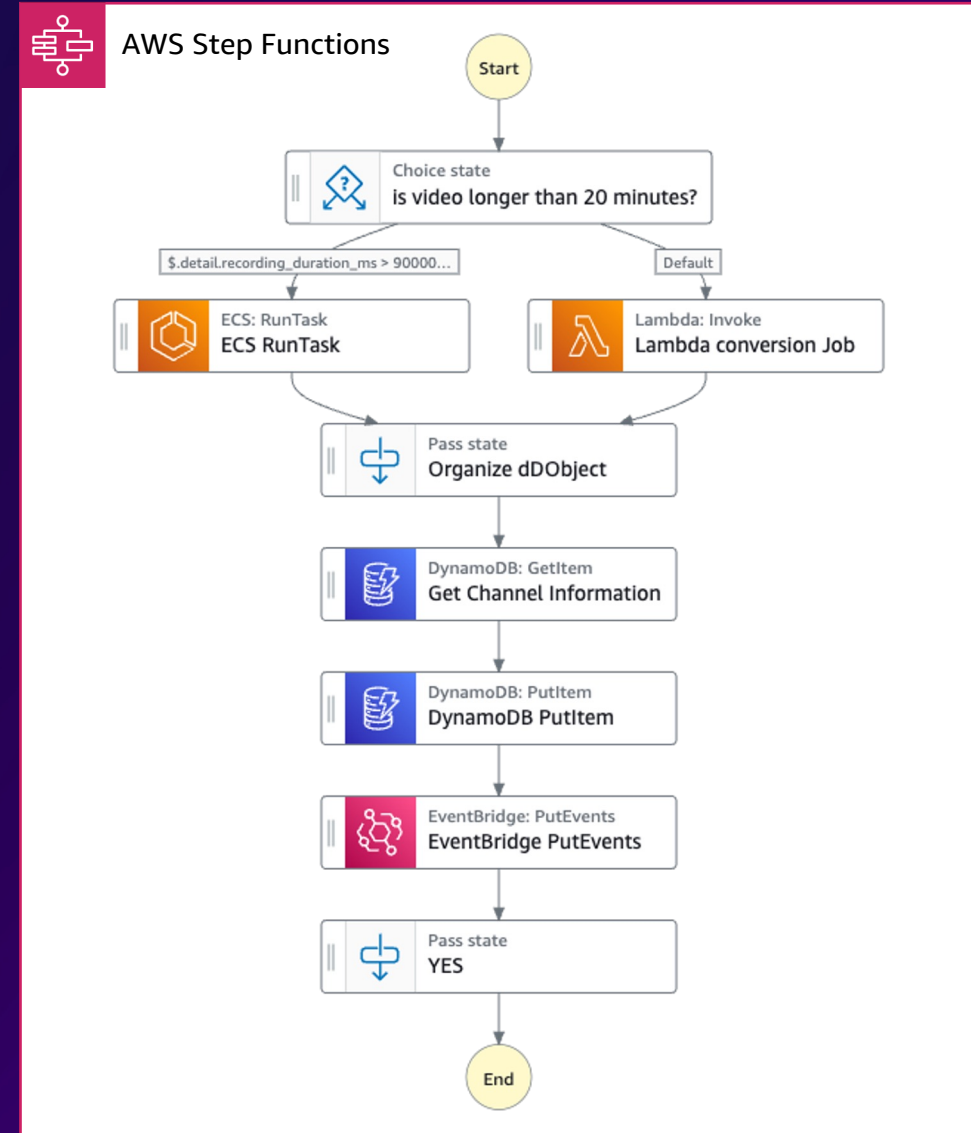


Plugin manager
service

Orchestrate the business process within domains

VIDEO POST-PROCESSING SERVICE

- Which compute to use
- Direct integration when possible

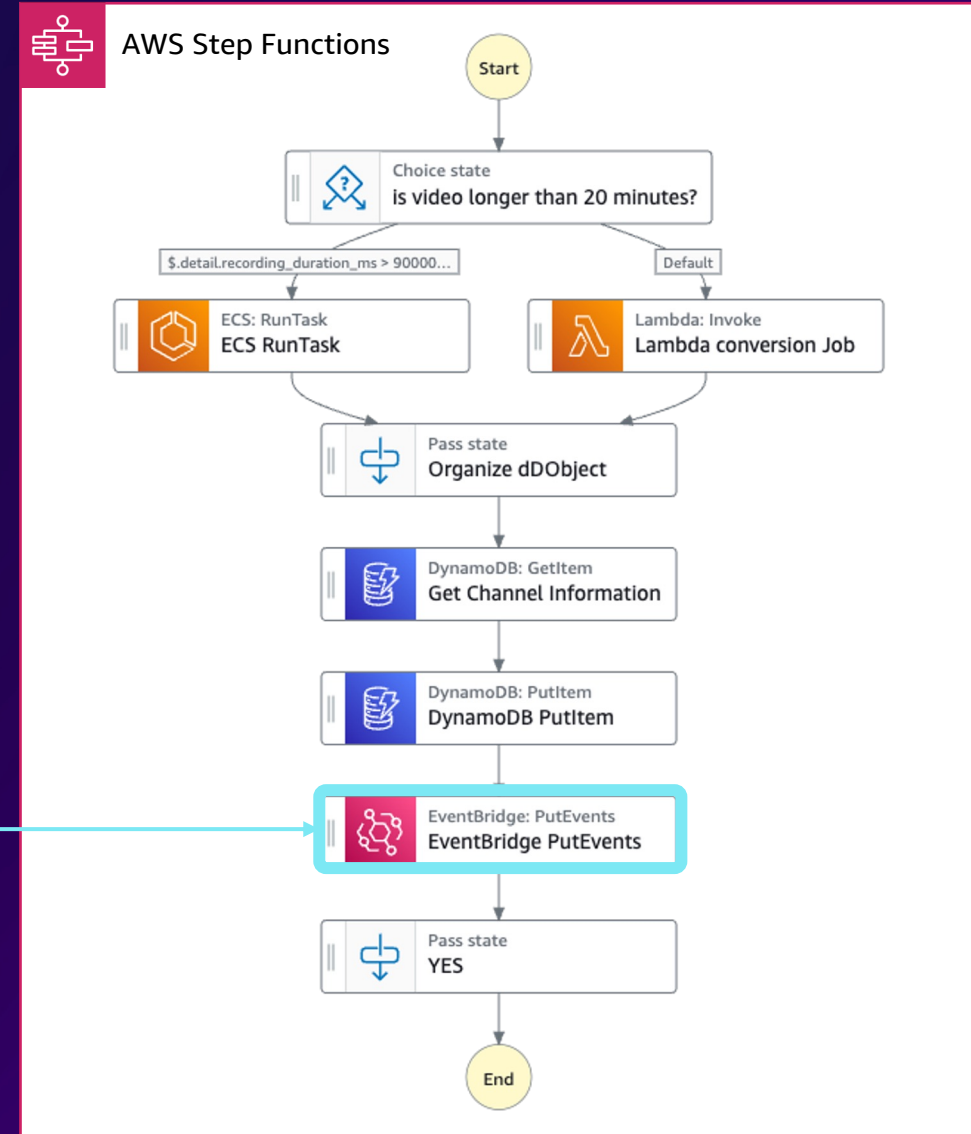


Orchestrate the business process within domains

VIDEO POST-PROCESSING SERVICE

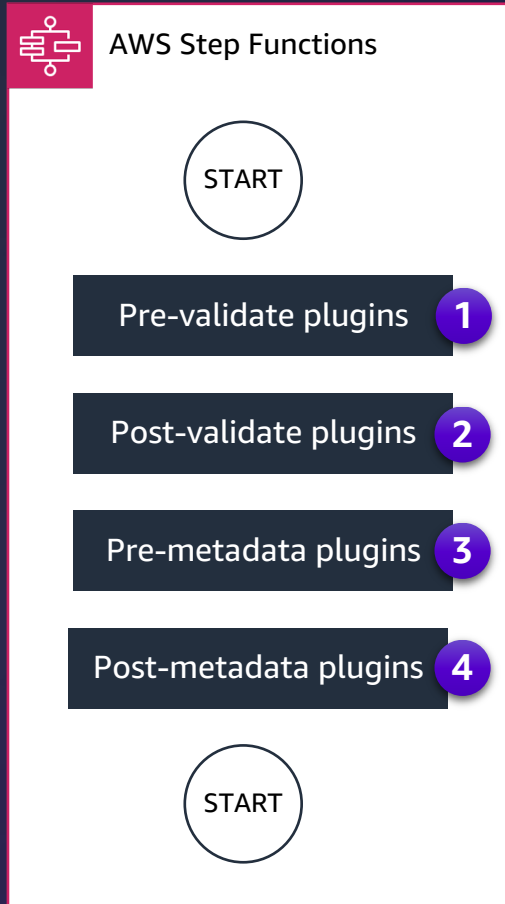
- Which compute to use
- Direct integration when possible

... and produce one or more events



Orchestrate the business process within domains

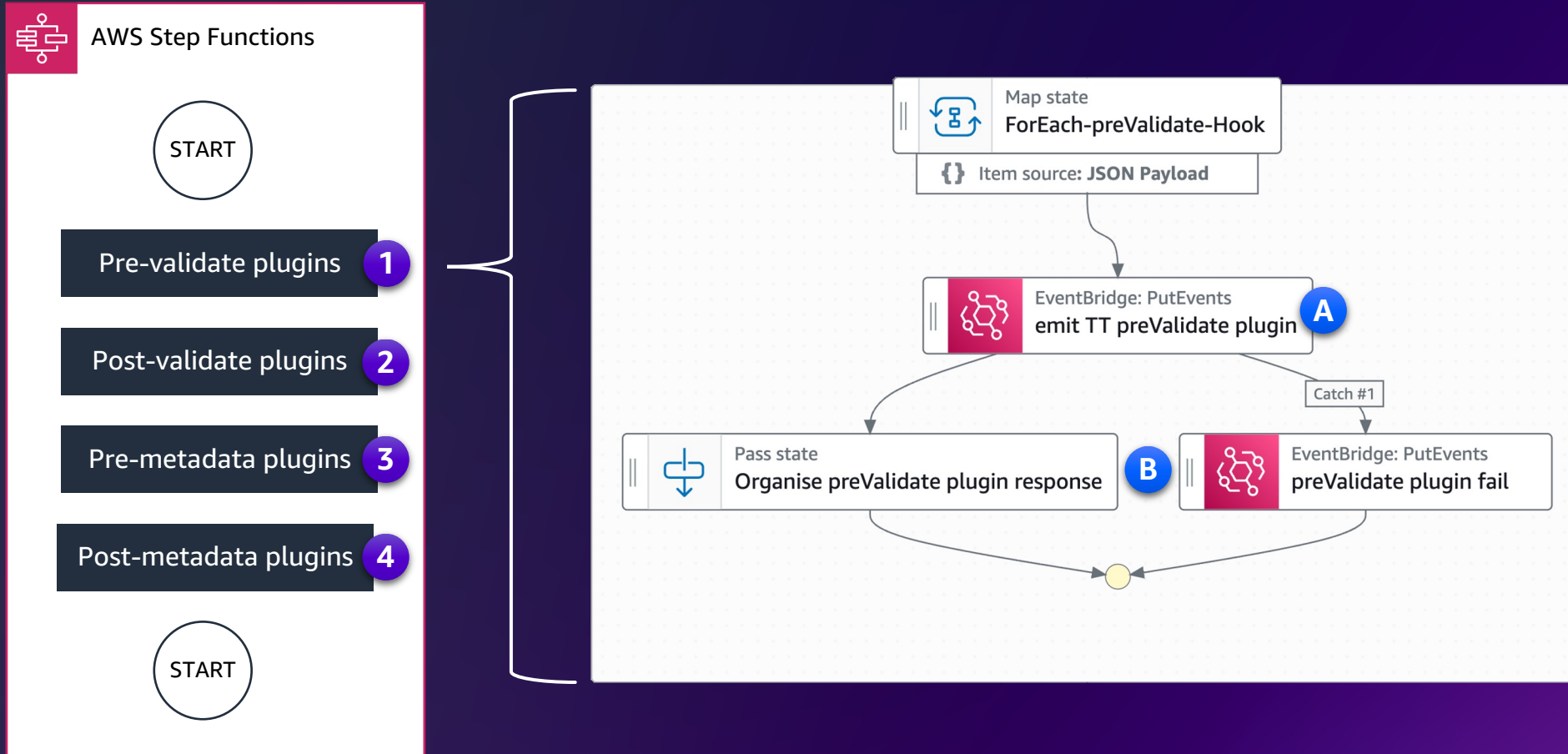
PLUGIN MANAGER



Each step must
run in order

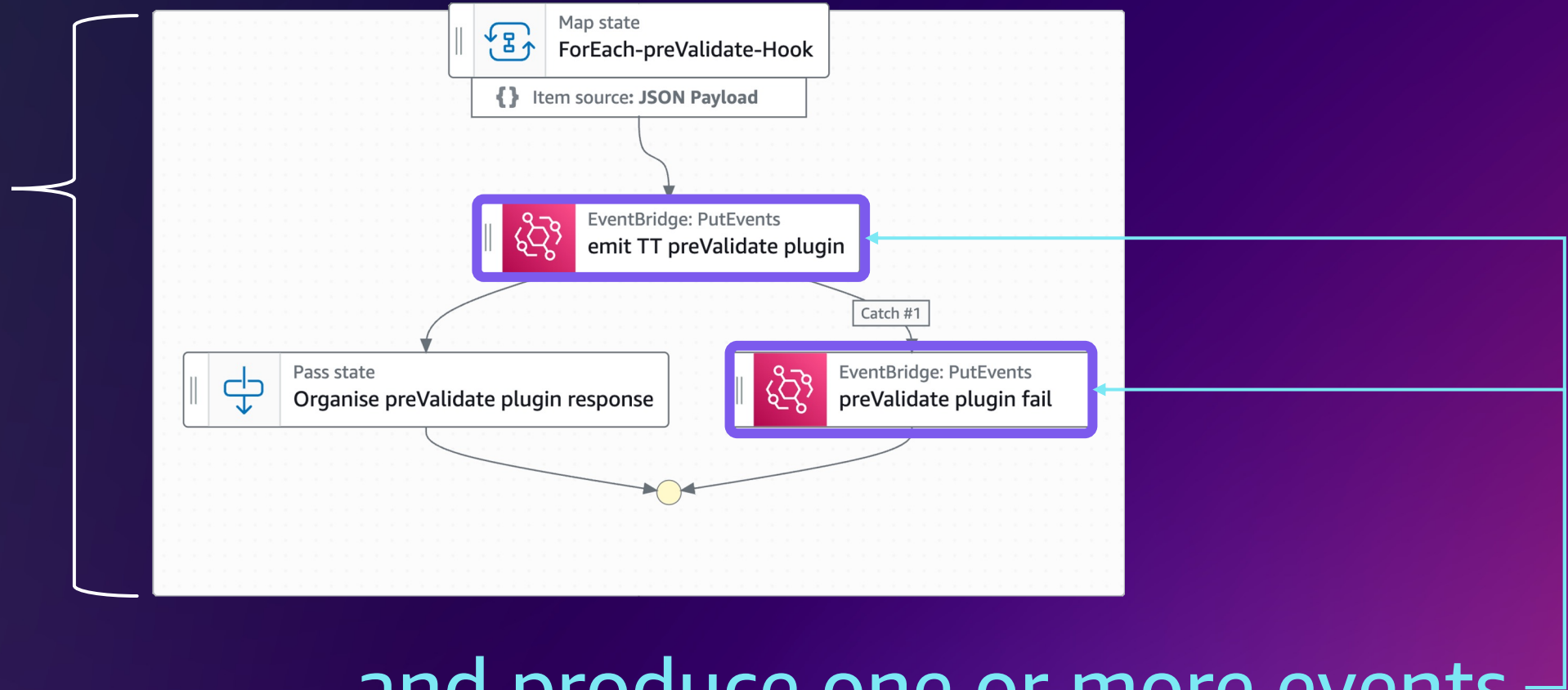
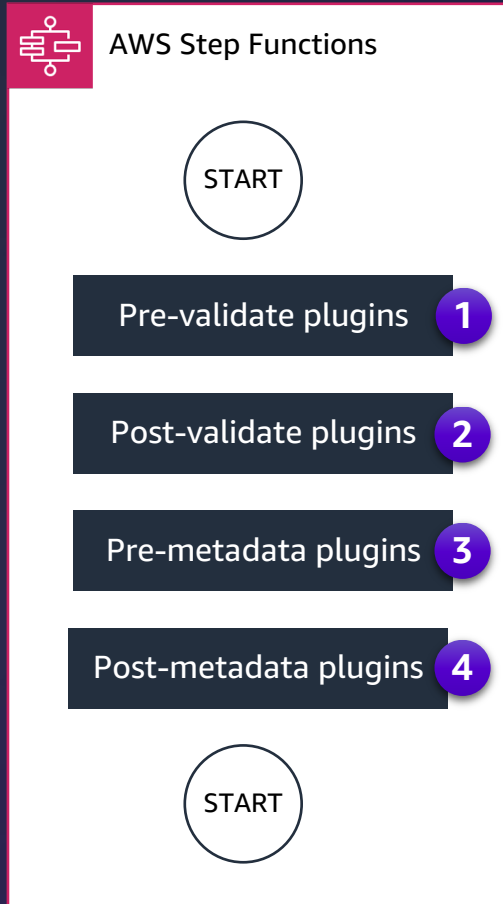
Orchestrate the business process within domains

PLUGIN MANAGER



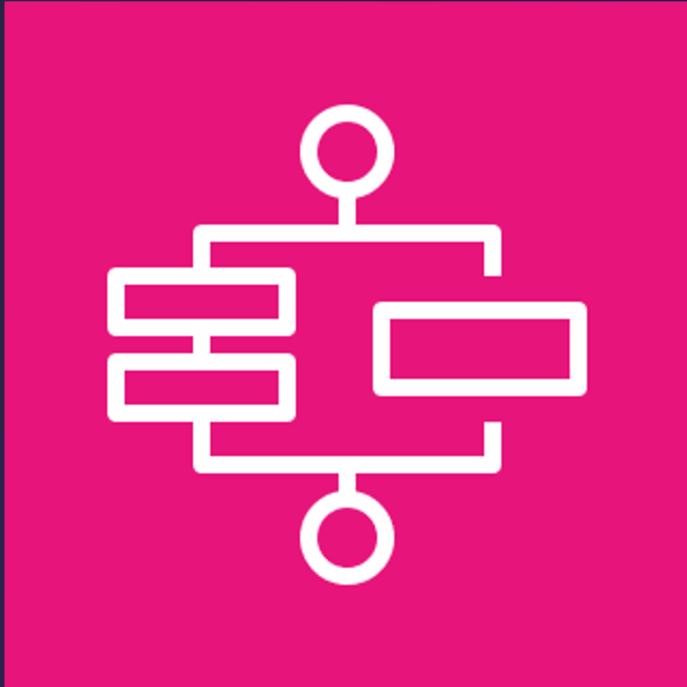
Orchestrate the business process within domains

PLUGIN MANAGER



... and produce one or more events

AWS Step Functions



AWS Step Functions

- Visual workflows for distributed applications
- Built-in error handling, timeouts, and parallel processing flow for distributed applications
- Direct integration with 200+ AWS services and over 10,000 APIs
- Built-in state management and workflow progress tracking

Choreography and orchestration working together



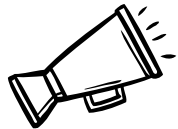
Channel
service



Video streaming
service



Publisher
service



Video manager
service



Video post-
processing service



Plugin manager
service

Choreography and orchestration working together

Orchestrate
business
processes
within the
domains

Orchestrate
business
processes
within the
domains

Orchestrate
business
processes
within the
domains

Choreograph events between domains to build loosely coupled,
highly resilient event-driven applications

Orchestrate
business
processes
within the
domains

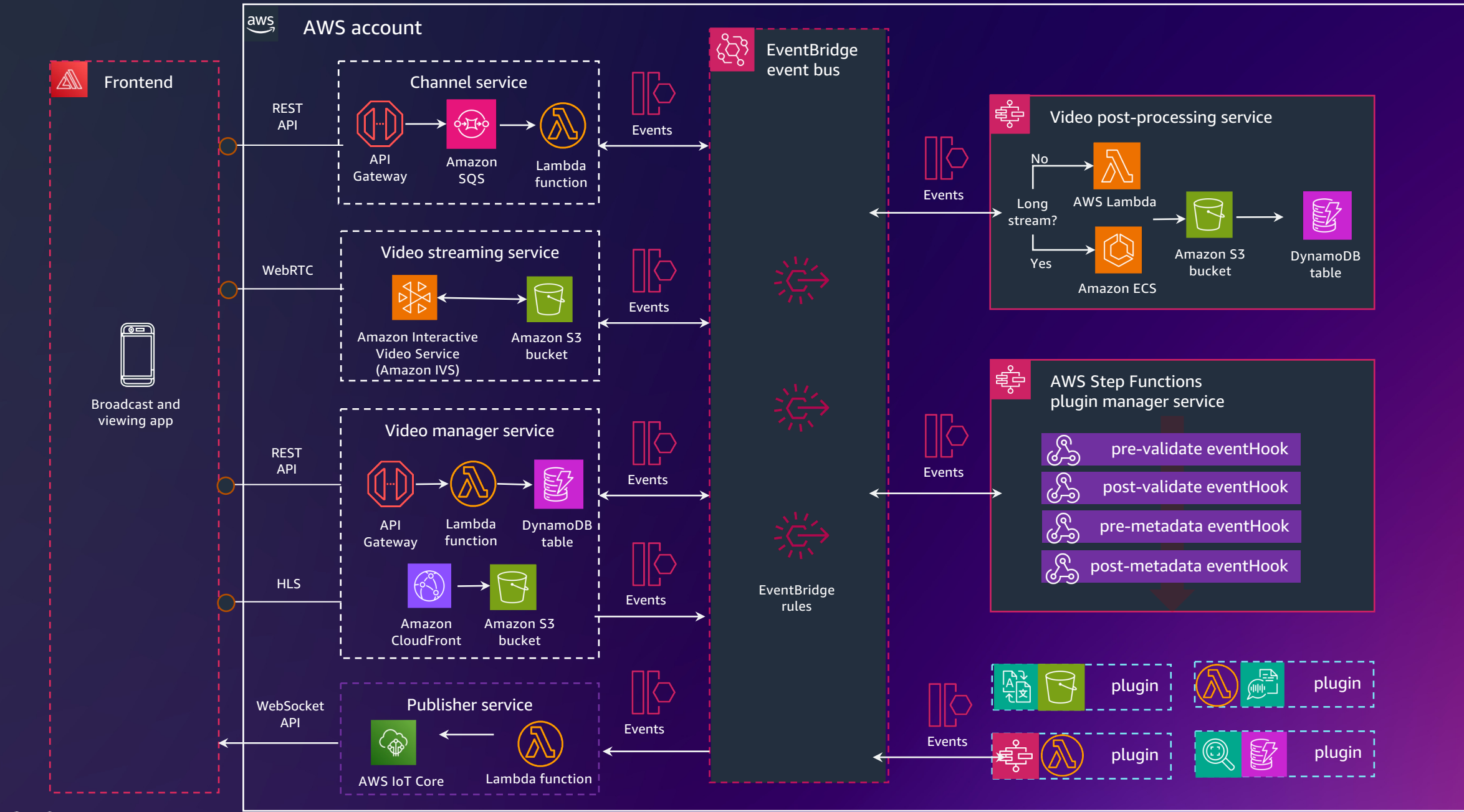
Orchestrate
business
processes
within the
domains

Orchestrate
business
processes
within the
domains



ServerlessVideo

The full ServerlessVideo event-driven architecture



Reducing event duplication with idempotency

What is idempotence, idempotency, idempotent?

THE MATHEMATICAL DEFINITION

$$f(x) = x + 0 = x$$

or

$$f(x) = x * 1 = x$$

Operations that can be applied multiple times
without changing the result

“A message that has the same effect whether it is received once or multiple times.”

Enterprise Integration Patterns

Gregor Hohpe and Bobby Woolf

**“Was my credit card
charged twice?! ”**

My mom
Just last week

Where can duplicates occur?

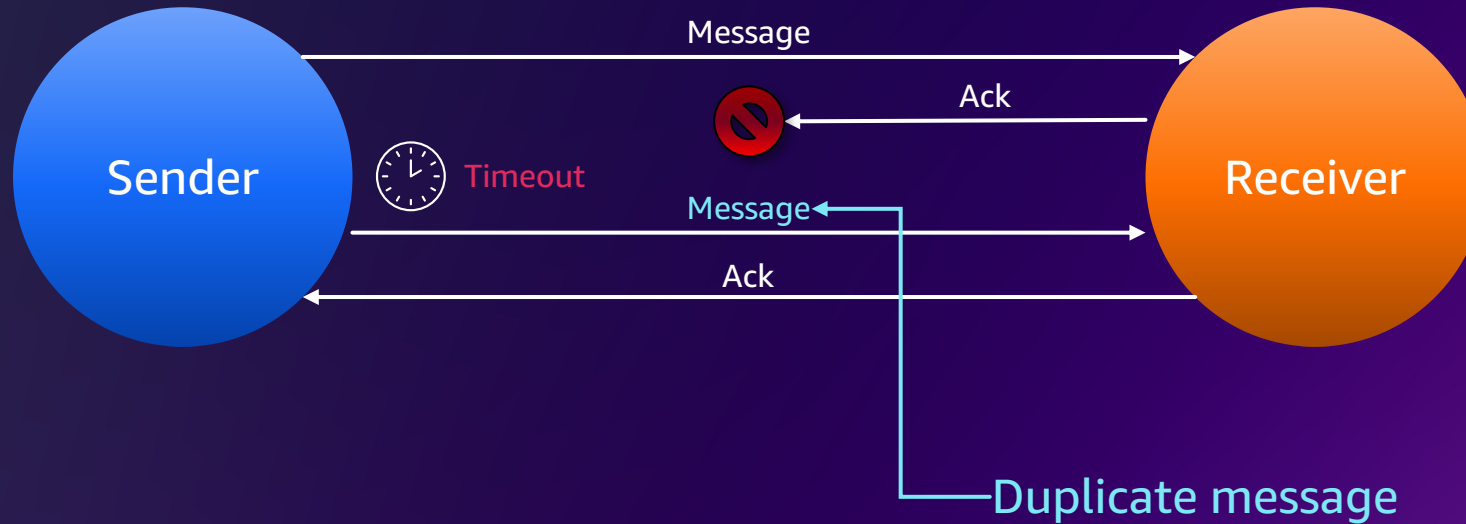
Where can duplicates occur?

TRANSMISSION ISSUES



Where can duplicates occur?

TRANSMISSION ISSUES



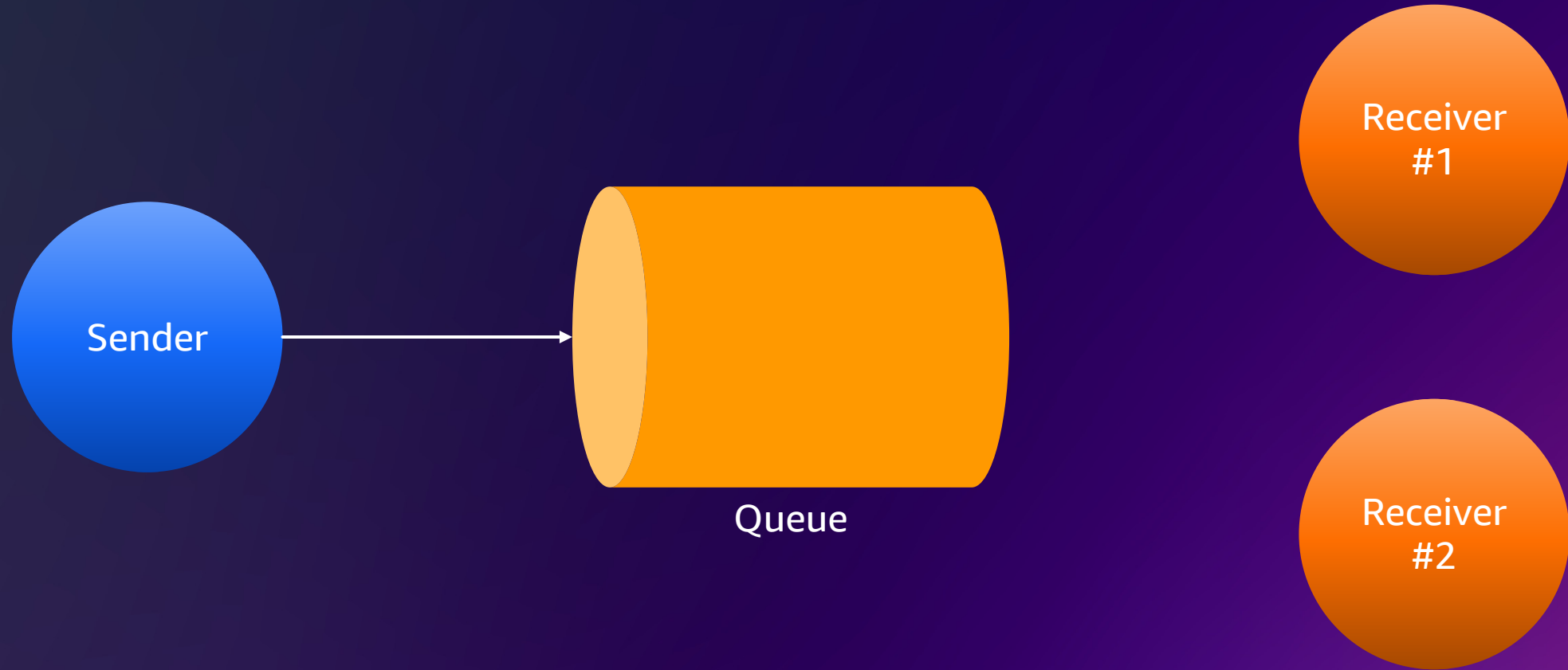
Where can duplicates occur?

RECEIVER ISSUES



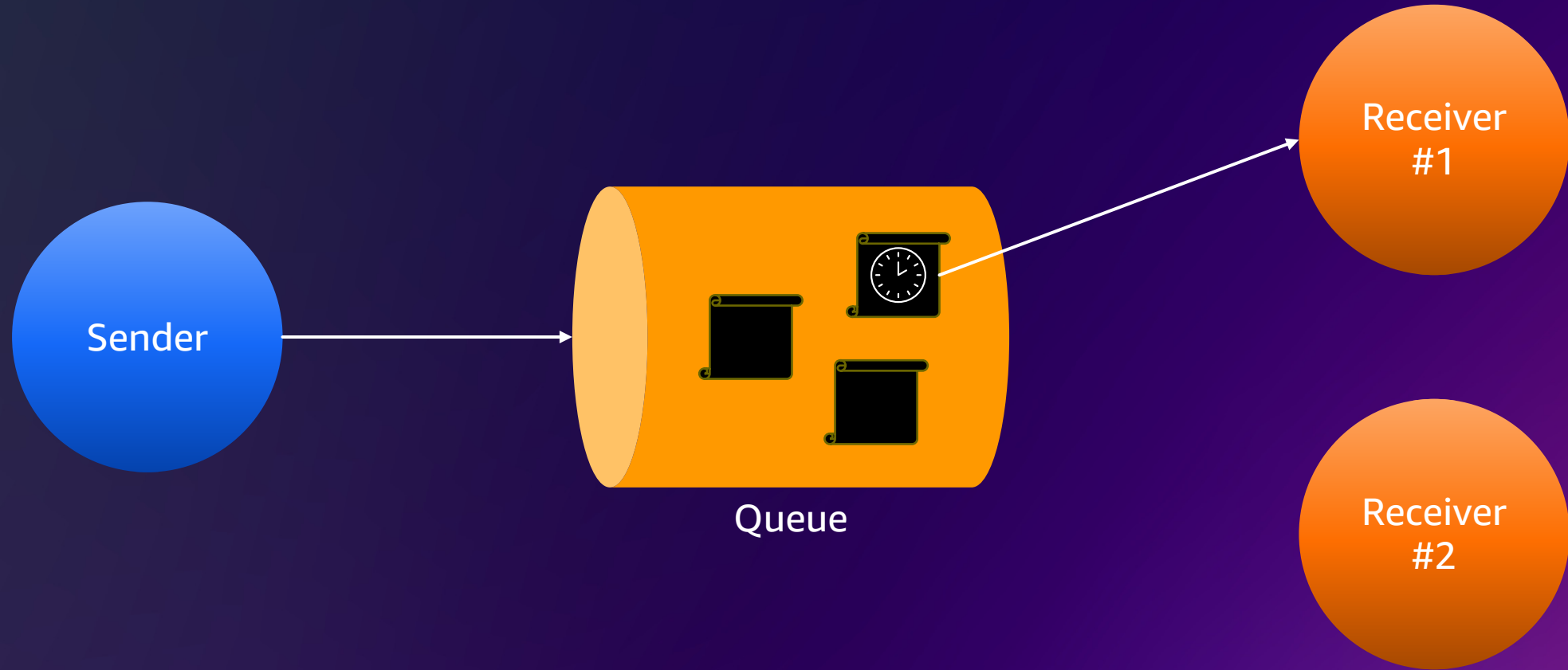
Where can duplicates occur?

RECEIVER ISSUES



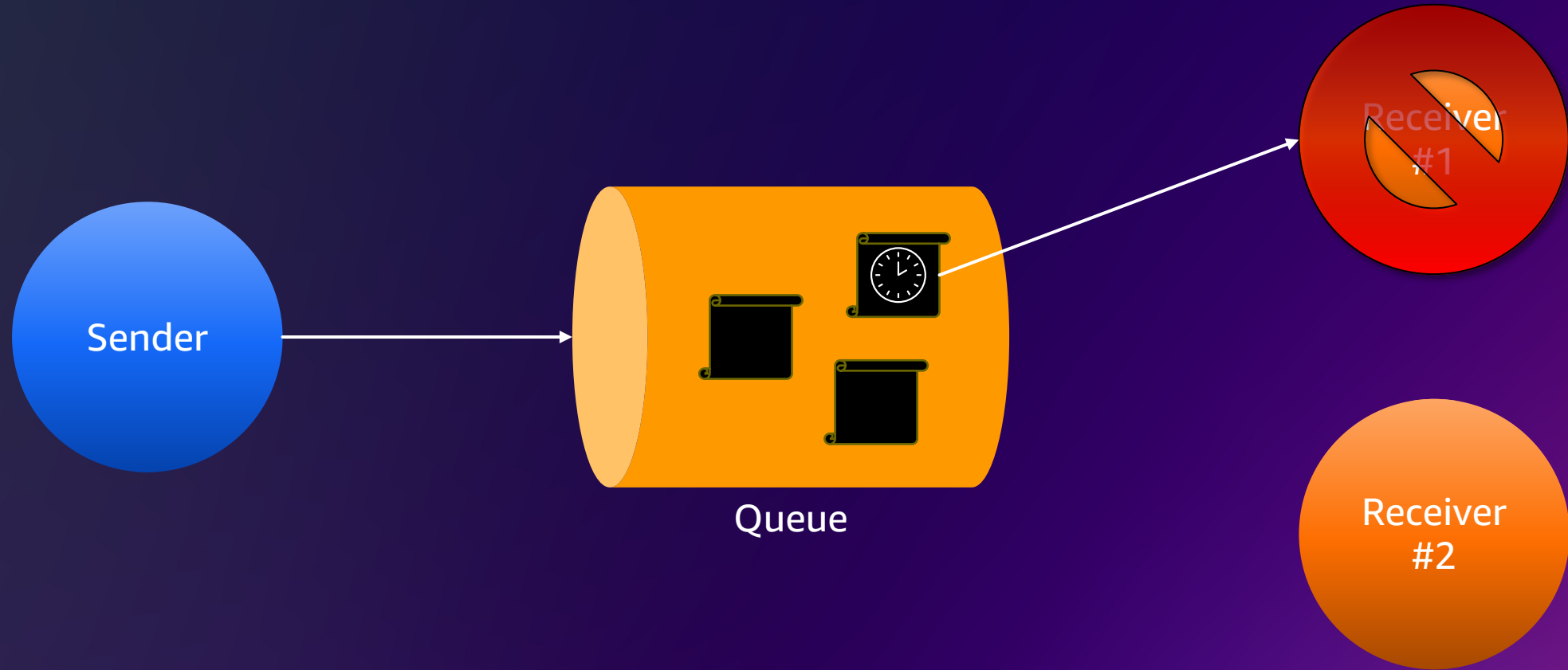
Where can duplicates occur?

RECEIVER ISSUES



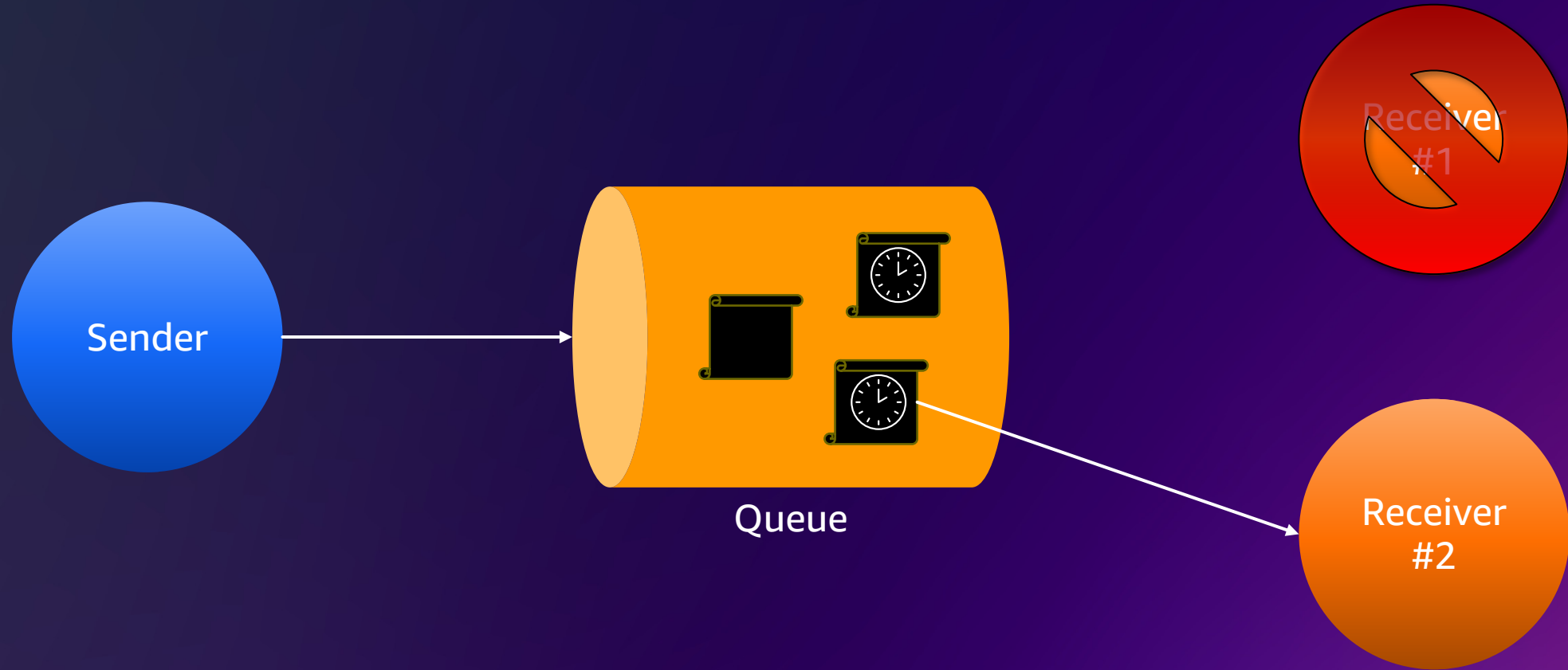
Where can duplicates occur?

RECEIVER ISSUES



Where can duplicates occur?

RECEIVER ISSUES



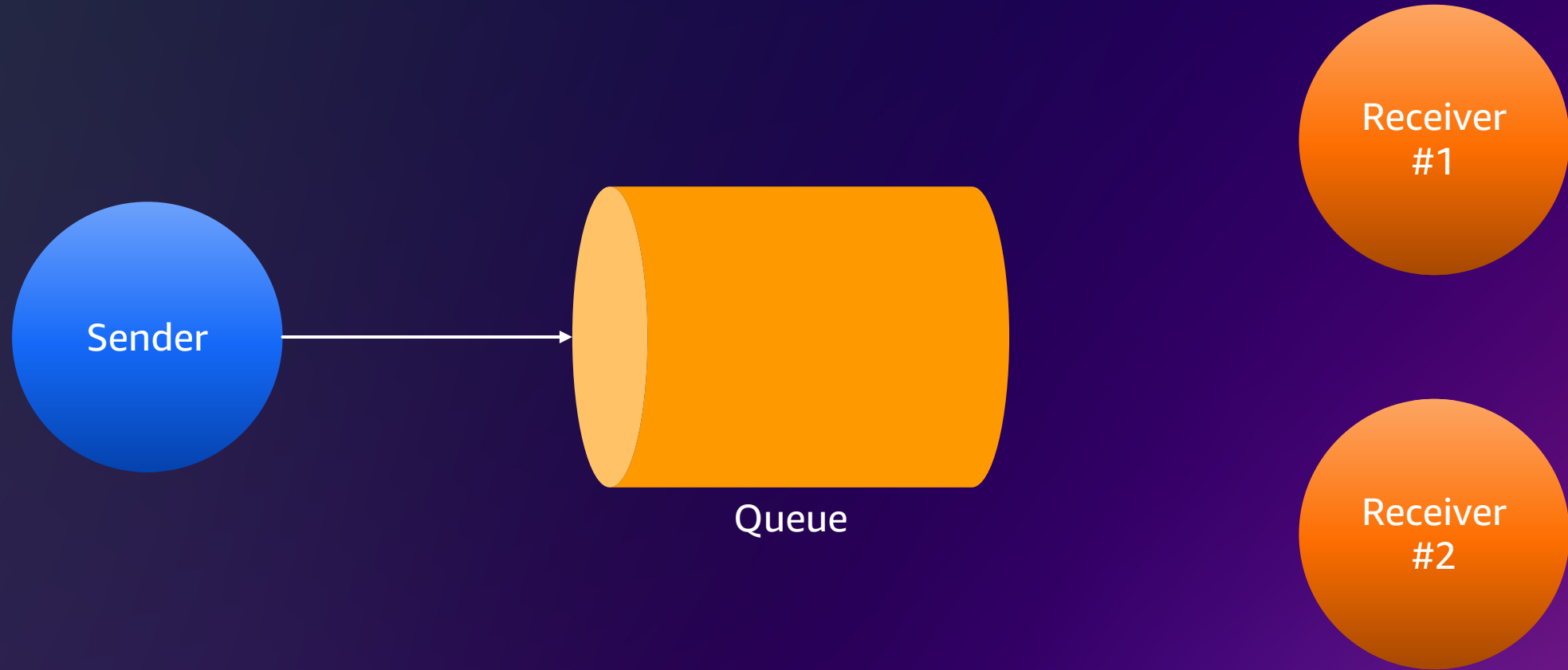
Where can duplicates occur?

SERVICE ISSUES



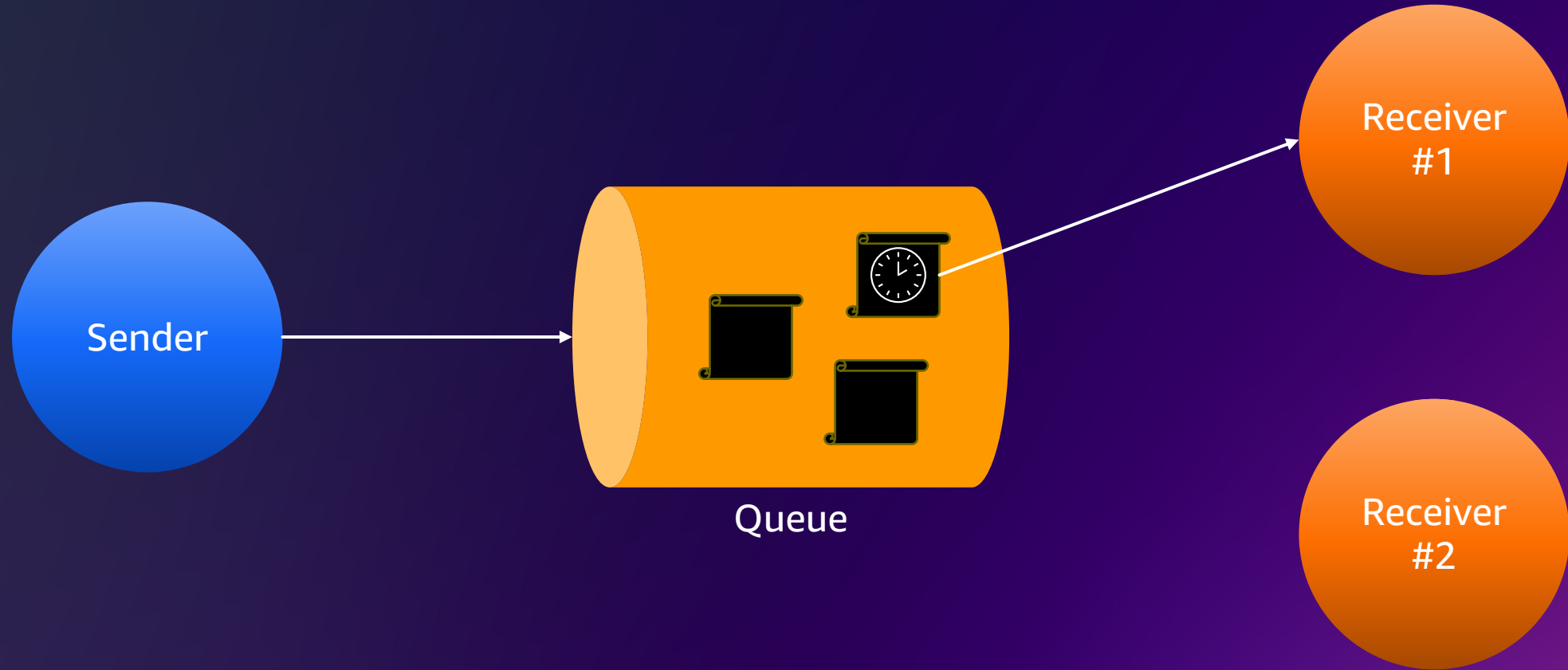
Where can duplicates occur?

SERVICE ISSUES



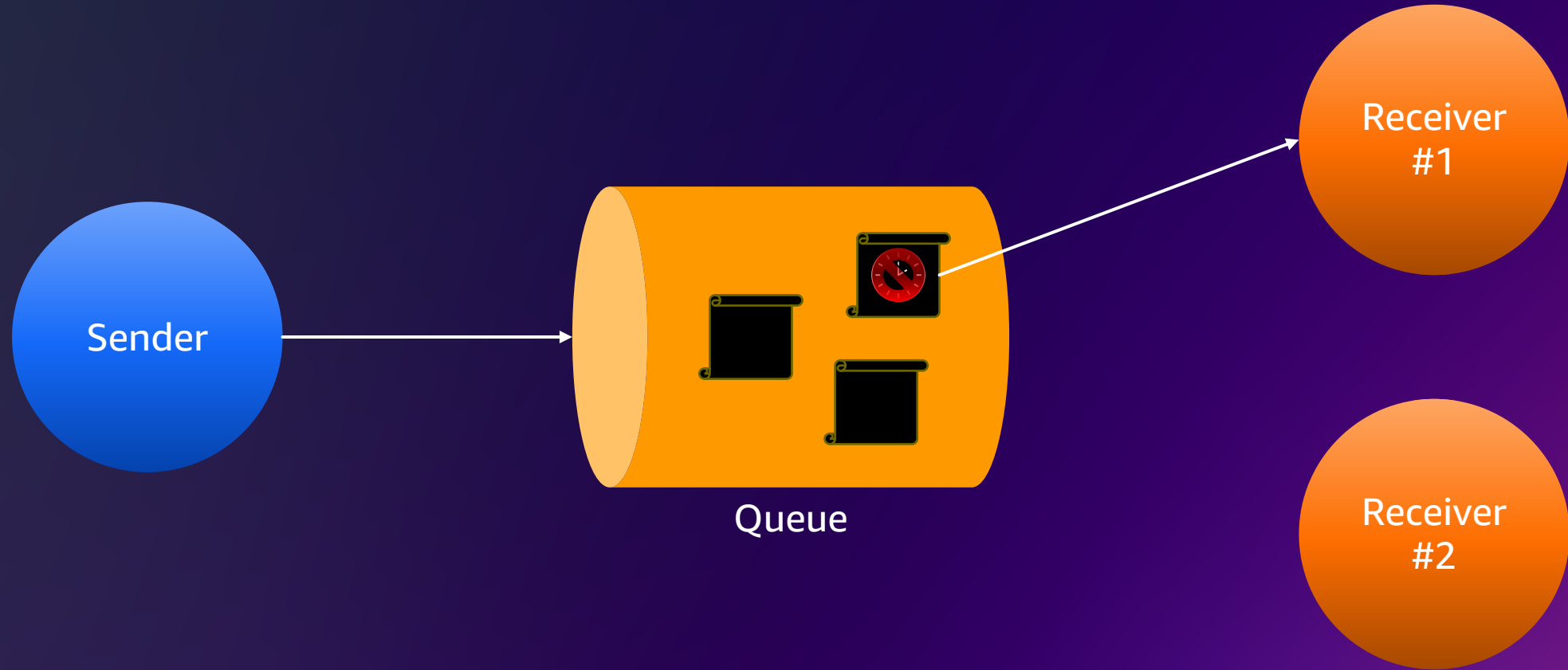
Where can duplicates occur?

SERVICE ISSUES



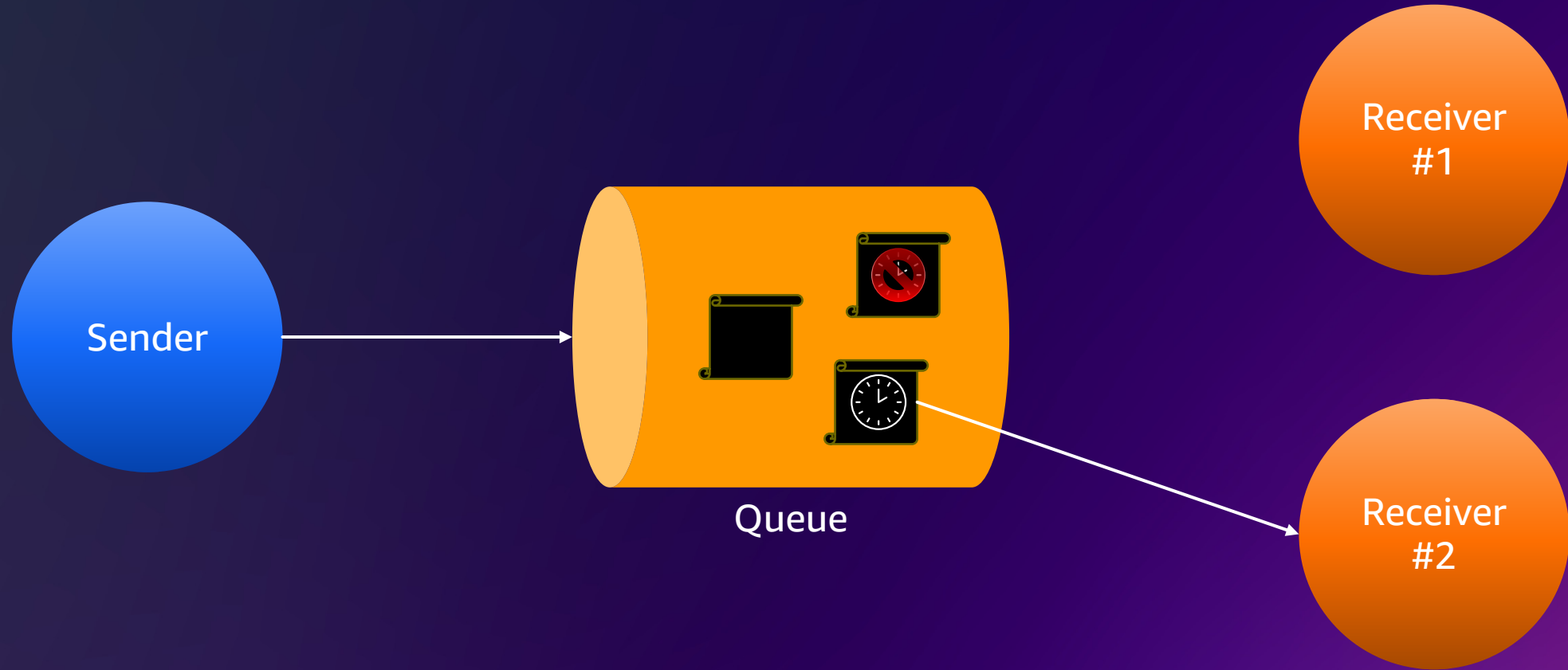
Where can duplicates occur?

SERVICE ISSUES



Where can duplicates occur?

SERVICE ISSUES



Idempotency tokens

Idempotency tokens

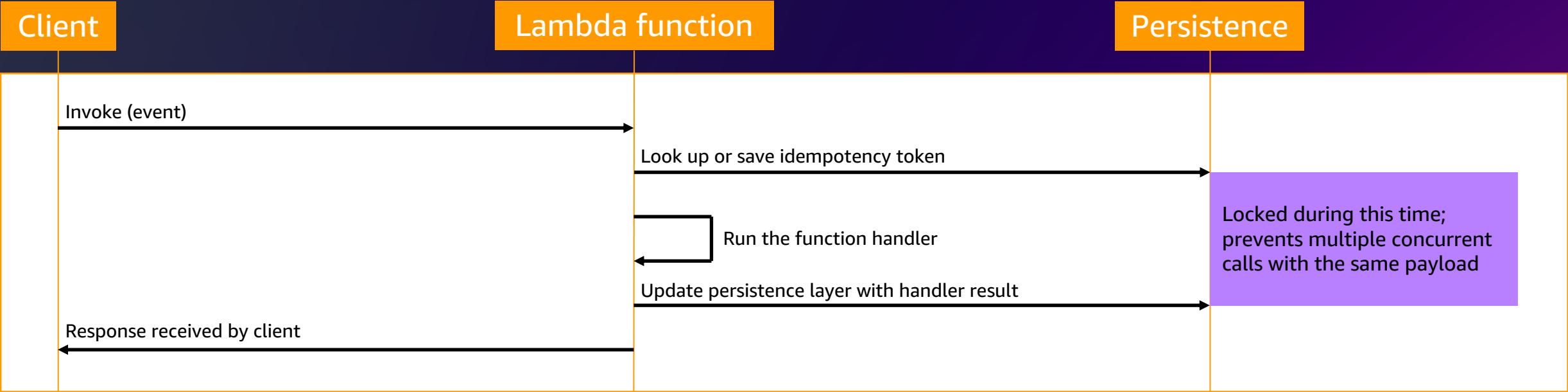
Idempotency tokens uniquely identify messages to enable receivers to avoid duplicate side effects

Well-behaved idempotency tokens

- Are **generated by the client**
- Are **re-sent by the client** on retry
- Are unique **per message** (UUID is common)
- Use a **dedicated field** separate from message content

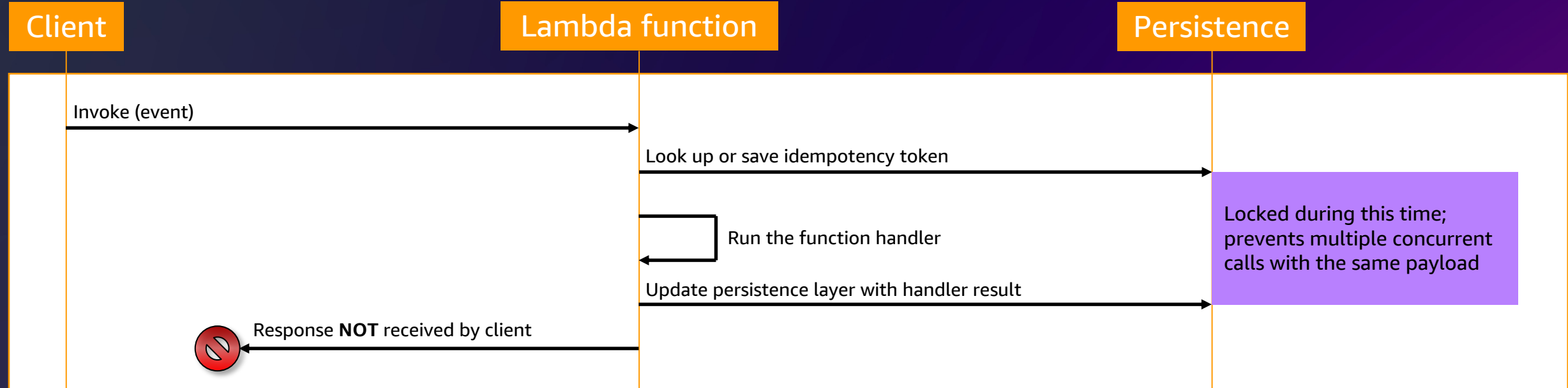
Idempotency tokens

IDEMPOTENT REQUEST SEQUENCE



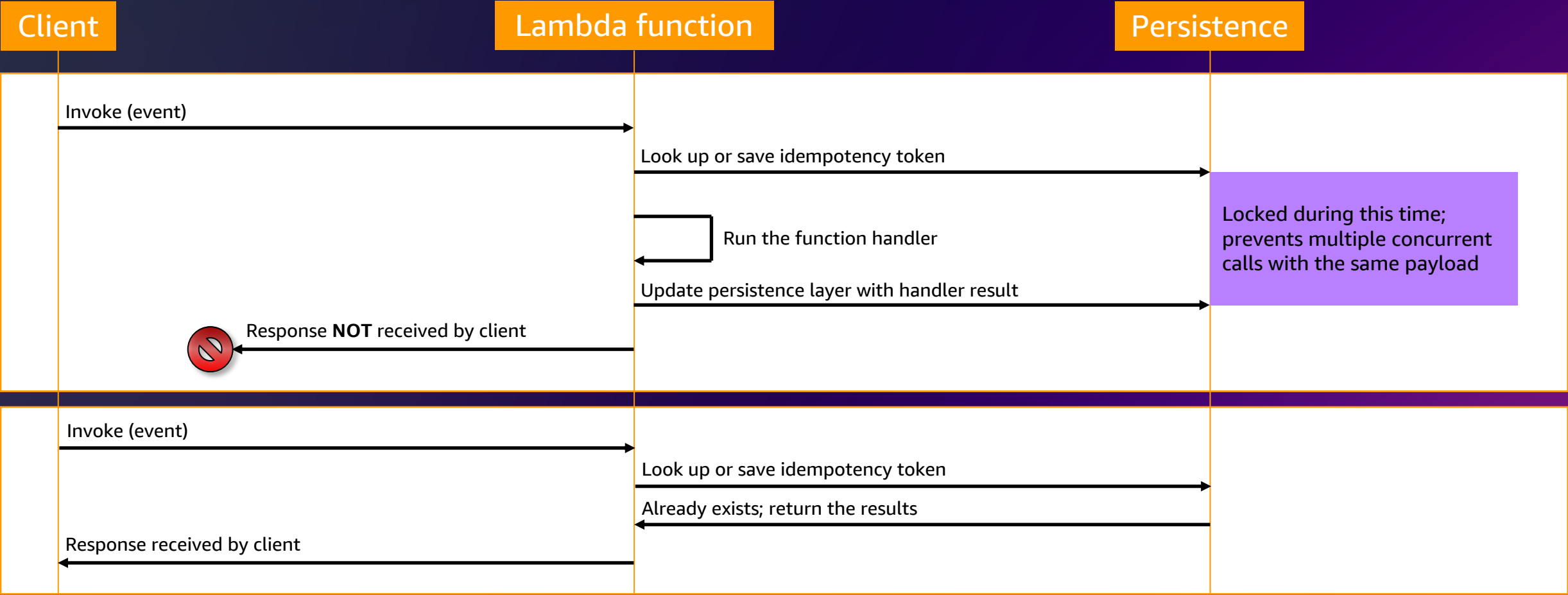
Idempotency tokens

IDEMPOTENT FAILED REQUEST SEQUENCE



Idempotency tokens

RETRIED IDEMPOTENT SEQUENCE



Idempotent services

Idempotent services

SQS.FIFO

```
% 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"  
}
```

Idempotent services

SQS.FIFO

```
% 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"  
}
```



Idempotent services

STEP FUNCTIONS

```
% 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"  
}
```

Idempotent services

STEP FUNCTIONS

```
% 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"  
}
```



Idempotent services

EVENTBRIDGE

```
% 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"
    }
  ]
}
```

Idempotent services

EVENTBRIDGE

```
% 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"
    }
  ]
}
```



Idempotent services

EVENTBRIDGE WITH 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

Idempotent services

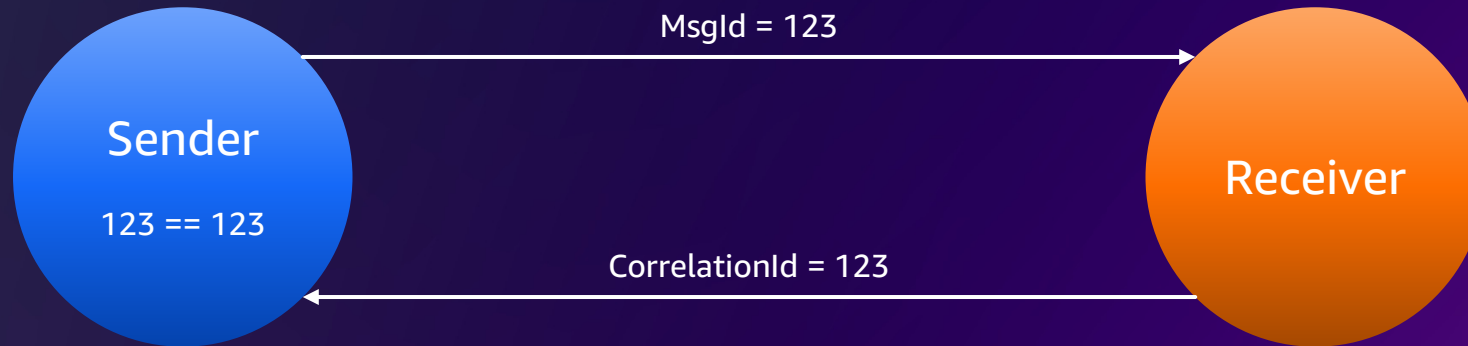
EVENTBRIDGE WITH IDEMPOTENCY IDENTIFIER

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

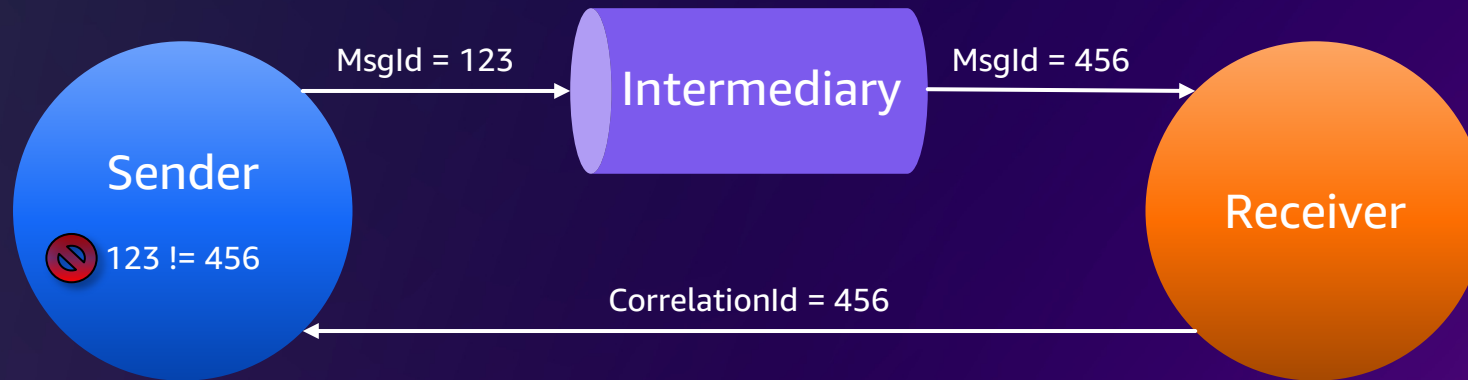
```
% aws events put-events \  
--entries '[{"Source": "foo", "DetailType": "bar", "Detail": "{\"metadata\": {\"idempotency-key\": \"$TOKEN\"}}"]'  
{  
  "Entries": [  
    {  
      "EventId": "a6765cc7-e0f8-c78d-4a06-64a5f45d4e3b"  
    }  
  ]  
}
```



Idempotency tokens



Idempotency tokens



GOTOEDADay Warsaw

You're Invited!

Come learn more about
EDA at GOTOEDADay
Warsaw tomorrow!

Code: **edjguest**



s12d.com/eda-warsaw

Serverlessland.com

- Code snippets
- Architectural patterns
- Workshops
- Visualizations
- Training videos
- ... and more

The screenshot displays the 'Serverless Land' website's 'Serverless Patterns Collection' page. The header includes the 'Serverless Land' logo, navigation links for 'Content', 'Learn', 'Code', and 'EDA', the 're:Invent 2023' event, and a search bar. The main heading is 'Serverless Patterns Collection'. Below this, a 'Submit a pattern' button is on the left. The central text explains that serverless patterns are used to build integrations with AWS SAM and CDK templates, and can be filtered by pattern and copied into applications. A 'Filters (669 templates)' sidebar on the left allows filtering by 'Type' (CDK, Pulumi, SAM, Serverless Framework, Terraform, Terraform (with modules)) and 'Language' (.NET, Go, Java). The main content area shows two pattern cards. The first card, titled 'ALB, Amazon Cognito and Lambda integration', shows a diagram of Application Load Balancer, Amazon Cognito, and AWS Lambda, and offers a 'View pattern' link for SAM/Python. The second card, titled 'Application Load Balancer with Lambda as a target', shows a diagram of Application Load Balancer and AWS Lambda, and offers a 'View pattern' link for CDK/TypeScript.

Serverless Land

Content ▾ Learn Code ▾ EDA ▾ re:Invent 2023

Search

Serverless Patterns Collection

[Submit a pattern](#)

Use serverless patterns to quickly build integrations using AWS SAM and CDK templates. Filter by pattern and copy the template directly into your application.

Filters (669 templates) [Reset](#)

Type

- ☐ CDK
- ☐ Pulumi
- ☐ SAM
- ☐ Serverless Framework
- ☐ Terraform
- ☐ Terraform (with modules)

Language

- ☐ .NET
- ☐ Go
- ☐ Java

Pattern 1: ALB, Amazon Cognito and Lambda integration

Application Load Balancer → Amazon Cognito → AWS Lambda

Create an ALB, Cognito and Lambda integration to fetch regional metrics

[SAM](#) [Python](#) [View pattern](#)

Pattern 2: Application Load Balancer with Lambda as a target

Application Load Balancer → AWS Lambda

Create an Application Load Balancer with Lambda as target using CDK

[CDK](#) [TypeScript](#) [View pattern](#)

Thank you!

Eric Johnson

 @edjgeek
 singledigit



© 2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Don't forget to
vote for this session
in the **GOTO Guide app**