



# **Eventbridge, best new serverless solution since lambda?**

**Overview of new features**

**Jacob Verhoeks**

**Awsug.nl 24 May 2022**



# Jacob Verhoeks

Schuberg Philis

AWS Architect / Data Engineer

<https://www.linkedin.com/in/jacobverhoeks/>

<https://github.com/jverhoeks>

<https://jacob.verhoeks.org/>

**AWS**

community  
**builders**

SCHUBERG  
PHILIS



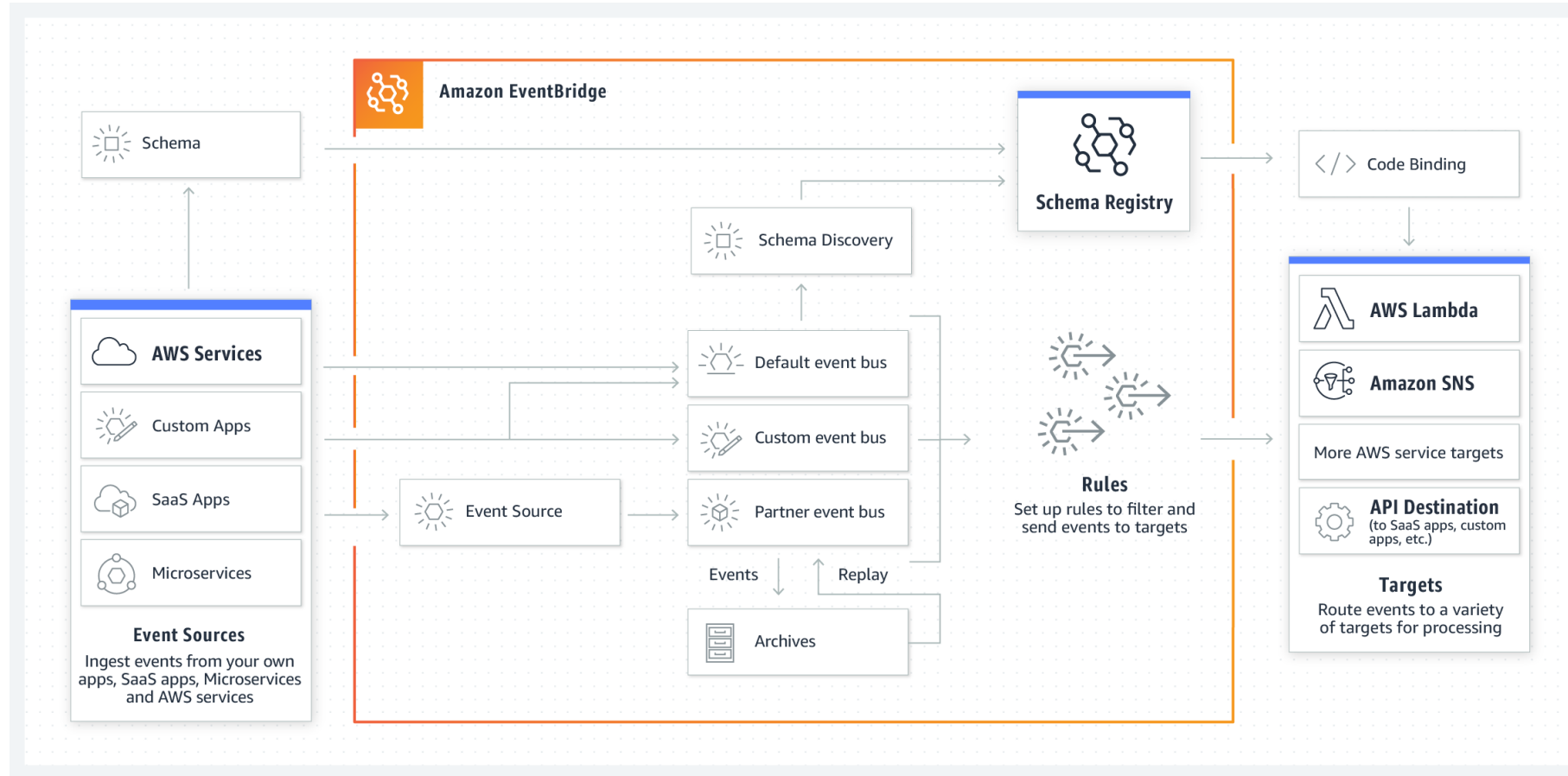


# EventBridge Overview releases Handy tools

For today



# July 2019 Announcement



<https://aws.amazon.com/blogs/aws/amazon-eventbridge-event-driven-aws-integration-for-your-saas-applications/>

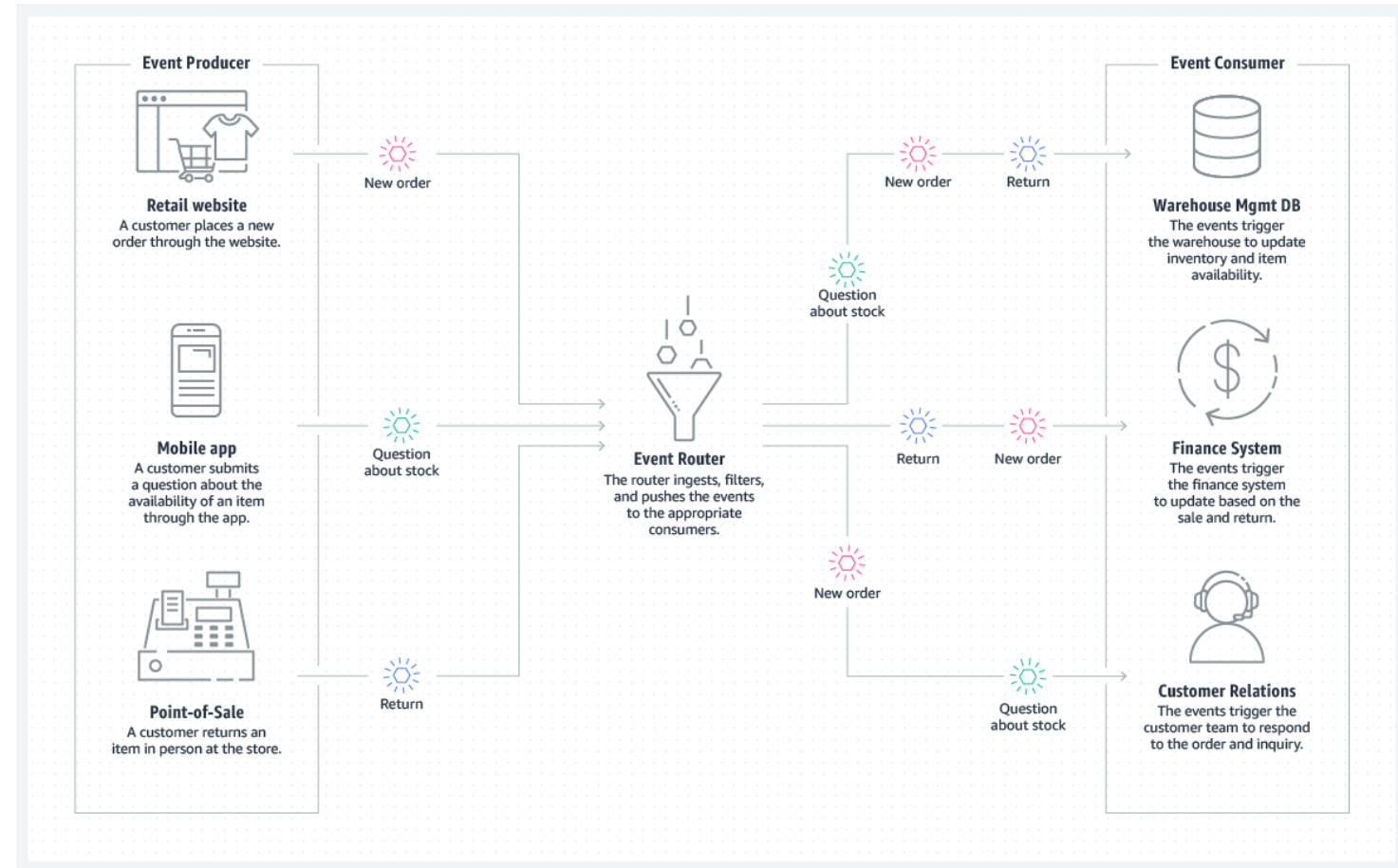
# Event Driven Design

## Async messaging

AWS Cloudtrail  
(except: list, get, describe)

AWS Services

SaaS



<https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-service-event.html>

## Oct 2019: Cloudformation & CDK

```
Type: AWS::Events::EventBus
Properties:
  EventSourceName: String
  Name: String
  Tags:
    - TagEntry
```

```
const bus = new events.EventBus(this, 'bus', {
  EventBusName: 'MyCustomEventBus'
});
```

Terraform start Sep 2019, finally released March **2022**

```
resource "aws_cloudwatch_event_bus" "messenger" {
  name = "chat-messages"
}
```



# Re:invent 2019/May 2020 Schema Registry

Amazon EventBridge > Schemas > aws.signin@AWSConsoleSignInViaCloudTrail

## aws.signin@AWSConsoleSignInViaCloudTrail

**Schema details**

Schema name	Last modified	Schema ARN	
aws.signin@AWSConsoleSignInViaCloudTrail	Feb 8, 2021, 06:04 PM GMT+1	-	
Description	Schema registry	Number of versions	Schema type
Schema for event type AWSConsoleSignInViaCloudTrail, published by AWS service aws.signin	aws.events	1	OpenAPI 3.0

**Version 1** Created on Feb 8, 2021, 06:04 PM GMT+1

Version 1 ▼ Action ▼ [Download code bindings](#)

```
1 {
2   "openapi": "3.0.0",
3   "info": {
4     "version": "1.0.0",
5     "title": "AWSConsoleSignInViaCloudTrail"
6   },
7   "paths": {},
8   "components": {
9     "schemas": {
10      "AWSEvent": {
11        "type": "object",
12        "required": ["detail-type", "resources", "id", "source", "time", "detail", "region", "version", "account"],
13        "x-amazon-events-detail-type": "AWS Console Sign In via CloudTrail",
14        "x-amazon-events-source": "aws.signin",
15        "properties": {
16          "detail": {
17            "$ref": "#/components/schemas/AWSConsoleSignInViaCloudTrail"
18          }
19        }
20      }
21    }
22  }
```

Lookup AWS Events  
Register Application Events  
Auto Discovery

OpenAPI 3 /  
JSONSchema Draft4

Codebindings to create Events  
(Java/Typescript/Python/GO)

# Feb 2020 Content Filtering

```
{
  "detail": {
    "state": [ { "anything-but": "initializing" } ]
  }
}
```

Filter messages in a rule to have only a subnet pass through.

```
{
  "detail": {
    "sourceIPAddress": [ { "cidr": "10.0.0.0/24" } ]
  }
}
```

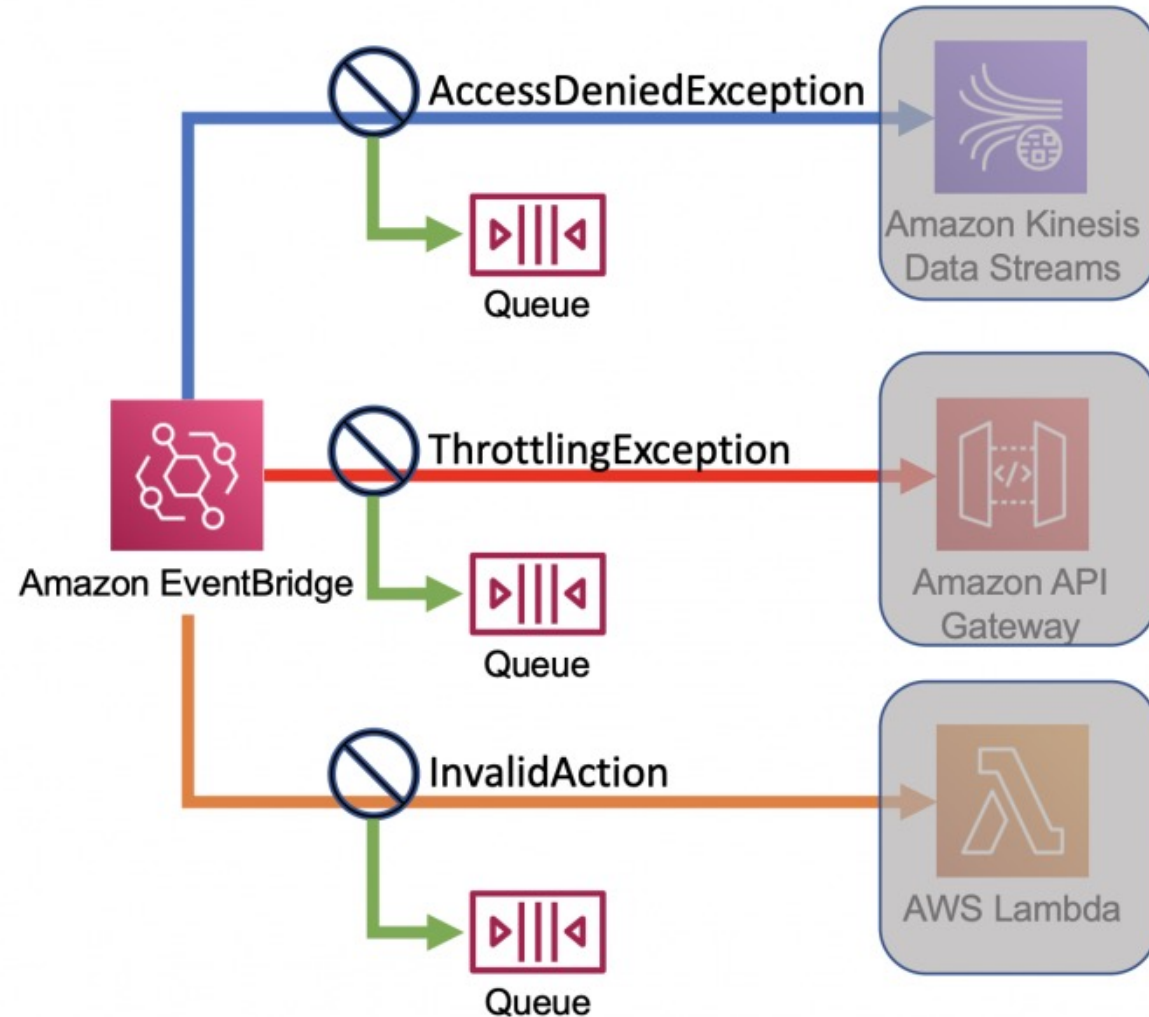
Instead of having to this in a lambda



# Oct 2020 Dead letter Queues

DLQ = SQS Queue

Default Retry Policy  
24 hours  
185 times



# Re:invent 2020 Event Archive & Replay

## Define archive details

Create an archive by defining the source and the retention period. Events are continuously saved in archives, and individual events will be deleted after the retention period. An archive will persist until you manually delete it.

Amazon EventBridge charges apply to archives. Please refer to [Amazon EventBridge Pricing](#) for details.

### Archive detail

#### Name

*Enter archive name*

Maximum of 48 characters consisting of numbers, lower/upper case letters, -, \_, . You can't change the name of the archive after it is created.

#### Description

*This is a description*

Maximum of 512 characters.

#### Source

The source, .e.g. an event bus, from which the events will be archived.

default

You can't change the source of the archive after it is created.

#### Retention period

The number of days the archive of events will be retained. 0 is equivalent to Indefinite. The maximum is 2 billion days.

Indefinite

days

### Specify rules(s) - optional

You can select one or more rules on the destination event bus. Events will be replayed to all rules or specific rules.

- All rules  
 Specify rules

Select a rule

### Replay time frame

Specify a time frame for replaying the events.

#### Start time

The ingestion time of the first event to be replayed.

##### Date

YYYY/MM/DD

YYYY/MM/DD

##### Time

hh:mm:ss

Use 24-hour format timestamp (hh:mm:ss)

##### Time zone

Local time zone

#### End time

The ingestion time of the last event to be replayed.

##### Date

YYYY/MM/DD

YYYY/MM/DD

##### Time

hh:mm:ss

Use 24-hour format timestamp (hh:mm:ss)

##### Time zone

Local time zone

<https://aws.amazon.com/blogs/aws/new-archive-and-replay-events-with-amazon-eventbridge/>

# Mar 2021 X-Ray + HTTP Api Destinations



## API Destination

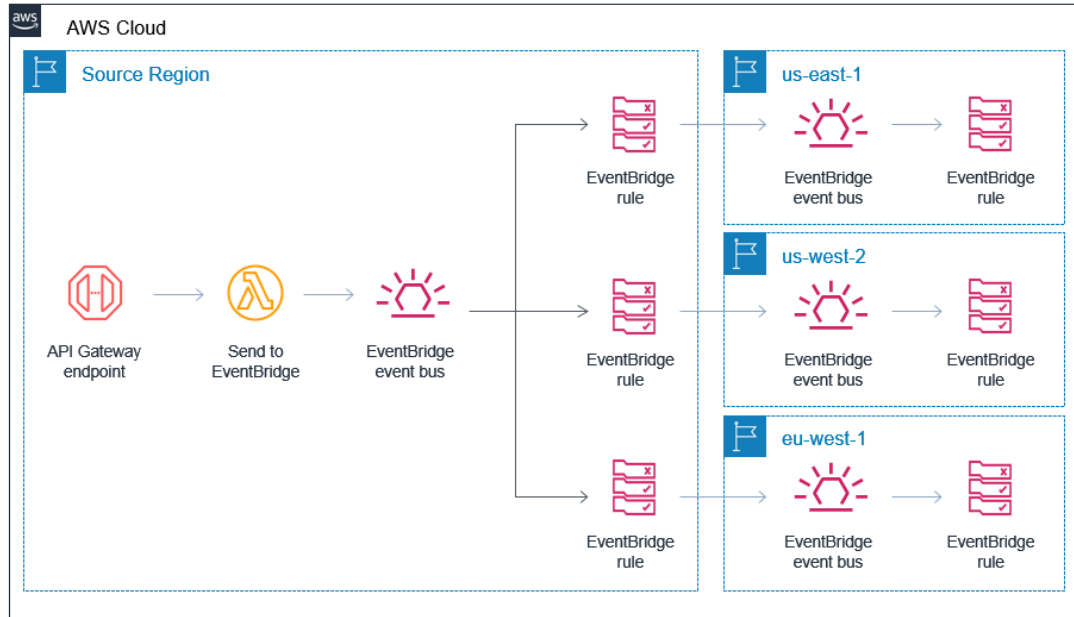
Authentication: Basic, OAuth, API key

JSON, with transformation

Datadog, NewRelic, Slack, etc, or your own



# April 2021 Cross Region Eventbus & Chatbot



**i** ECS Task State Change | us-east-2 | Account: [REDACTED]

An ECS task running on cluster 'Default' transitioned to PENDING status.

**Task** arn:aws:ecs:us-east-2:[REDACTED]

**Task Definition** arn:aws:ecs:us-east-2:[REDACTED]:task-definition/first-run-task-definition:1

**Cluster** arn:aws:ecs:us-east-2:[REDACTED]:cluster/Default

**Last Status** PENDING

**Desired Status** RUNNING

**Updated at** Wed, 14 Apr 2021 00:14:12 GMT

<https://aws.amazon.com/blogs/mt/monitor-amazon-eventbridge-events-in-your-slack-channels-with-aws-chatbot/>

# July 2021 Glue workflows

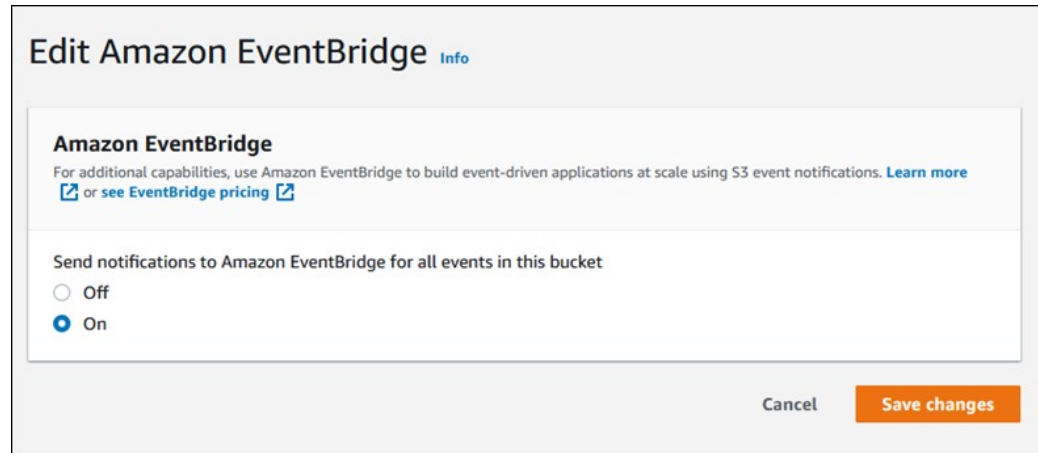
## **Event Sources**

Crawler State Change  
Data Catalog Database State Change  
Data Catalog Table State Change  
Job Run Status  
Job State Change

## **Event Targets**

Initiate Glue Workflow with event rule

# Re:invent 2021 S3 Event Notification (1/3)



Edit Amazon EventBridge [Info](#)

**Amazon EventBridge**  
For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. [Learn more](#)  
[or see EventBridge pricing](#)

Send notifications to Amazon EventBridge for all events in this bucket

Off  
 On

Cancel [Save changes](#)

Alternative for Event Notifications

Advanced Filtering  
Multiple destinations  
Fast, reliable (at-least once)

<https://aws.amazon.com/blogs/aws/new-use-amazon-s3-event-notifications-with-amazon-eventbridge/>



# Re:invent 2021 S3 Event Notification (2/3)

## Sample event - optional

You don't have to select or enter a sample event, but it's recommended so you can reference it when writing or filter criteria.

You can reference the sample event when write the event pattern, or use the sample event to event pattern. Find a sample event, enter your own, or edit a sample event below.

Learn more about sample/test event in [Test Event Pattern](#).

AWS events

EventBridge partner events

Ente

## Sample events

Filter by event source and type or by keyword.

Object Created

```
1 {
2   "version": "0",
3   "id": "17793124-05d4-b198-2fde-7ededc63b103",
4   "detail-type": "Object Created",
5   "source": "aws.s3",
6   "account": "123456789012",
7   "time": "2021-11-12T00:00:00Z",
8   "region": "ca-central-1",
9   "resources": ["arn:aws:s3:::example-bucket"],
10  "detail": {
11    "version": "0",
12    "bucket": {
13      "name": "example-bucket"
14    },
15    "object": {
16      "key": "example-key",
17      "size": 5,
18      "etag": "b1946ac92492d2347c6235b4d2611184",
19      "version-id": "TVV3n45RT0ac8kiHa1hou5d51a_Mno8a"
```

## Event pattern Info

Event pattern form

Custom pattern

✔ Sample event matched the event pattern.

## Event pattern

Write an event pattern in JSON. You can test the event pattern against the sample event. You can also go to pre-defi

Select matching pattern

Insert

Content-based filter synt

```
1 {
2   "source": ["aws.s3"],
3   "detail-type": ["Object Created"],
4   "detail": {
5     "bucket": {
6       "name": [
7         "example-bucket"
8       ]
9     },
10    "object": {
11      "size": [{
12        "numeric": ["<=", 1048576]
13      ]
14    }
15  }
16 }
```

✔ JSON is valid

Copy

Prettify JSON

Event pattern form

Test pattern

# Re:invent 2021 S3 Event Notification (3/3)

## Target 1

### Target types

Select an EventBridge event bus, EventBr

- EventBridge event bus
- EventBridge API destination
- AWS service

API Gateway

Batch job queue

CloudWatch log group

CodeBuild project

CodePipeline

EBS Create Snapshot

EC2 ImageBuilder

EC2 RebootInstances API call

EC2 StopInstances API call

EC2 TerminateInstances API call

ECS task

Firehose delivery stream

Glue workflow

Incident Manager response plan

Inspector assessment template

Kinesis stream

Lambda function

Redshift cluster

SageMaker Pipeline

SNS topic

SQS queue

Step Functions state machine

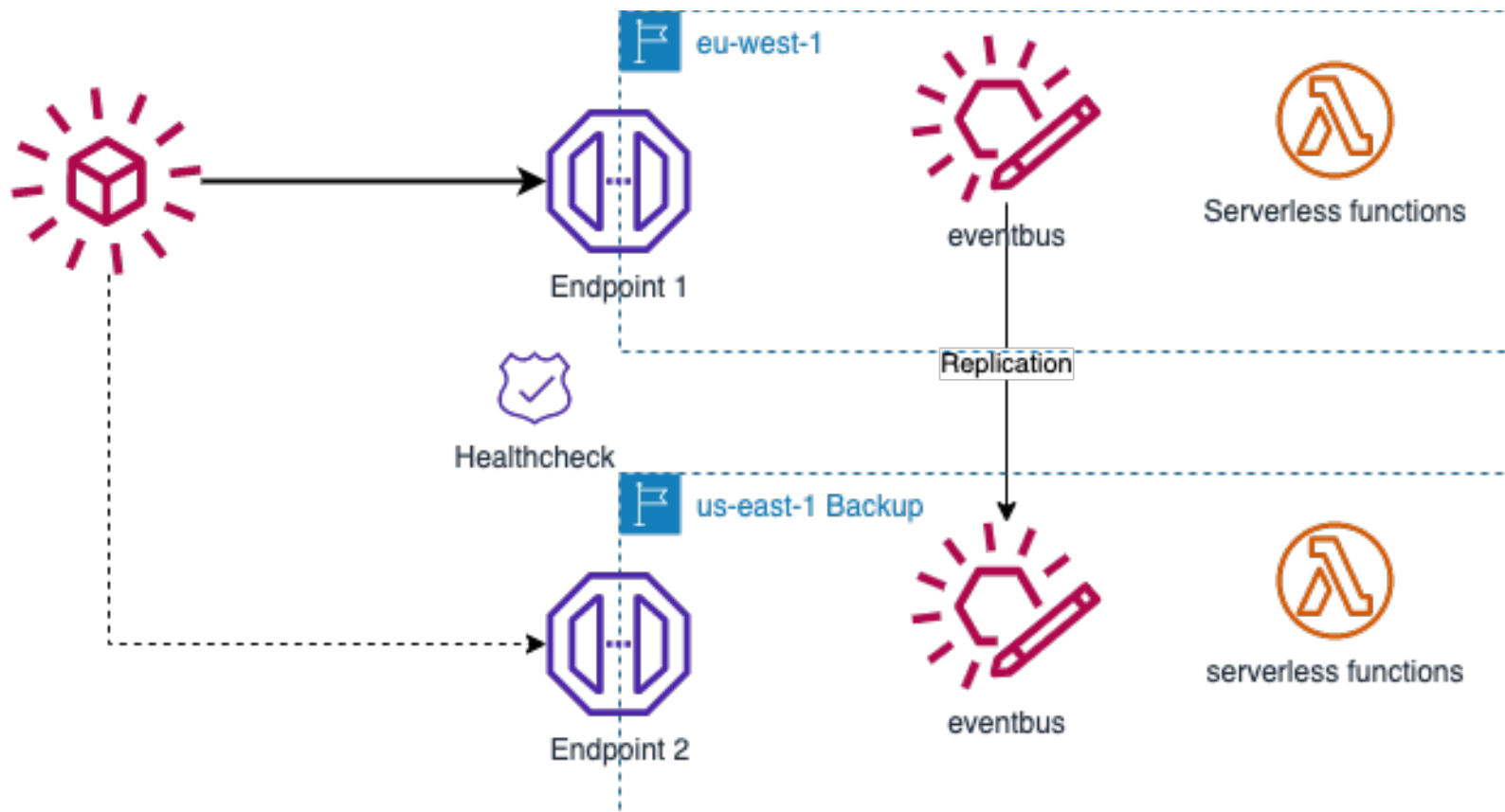
System Manager Automation

System Manager OpsItem

System Manager Run Command

SNS topic

# April 2022 Global Endpoints







DAY  
21  
9

22

22

AUGUST  
WEDNESDAY

MARCH  
TUESDAY

JANUARY  
THURSDAY

MARCH  
DAY  
APRIL  
WEDNESDAY

23

24

10  
1

WEDNESDAY

3

29

DAY

TUESDAY

SEPTEMBER  
FRIDAY

16

AUGUST  
SUNDAY

30

6

5

7

NOVEMBER  
MONDAY

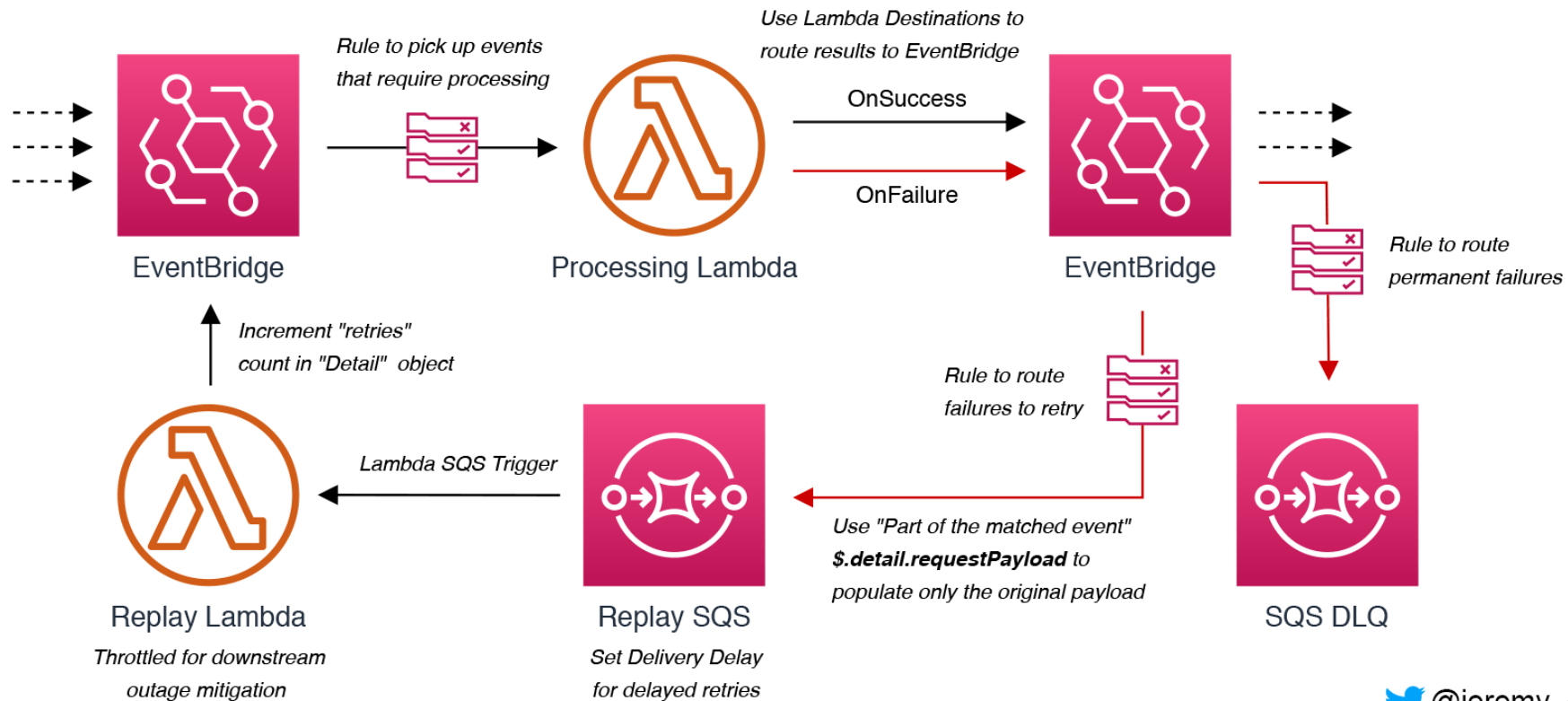
JUNE  
MONDAY

DECEMBER  
TUESDAY

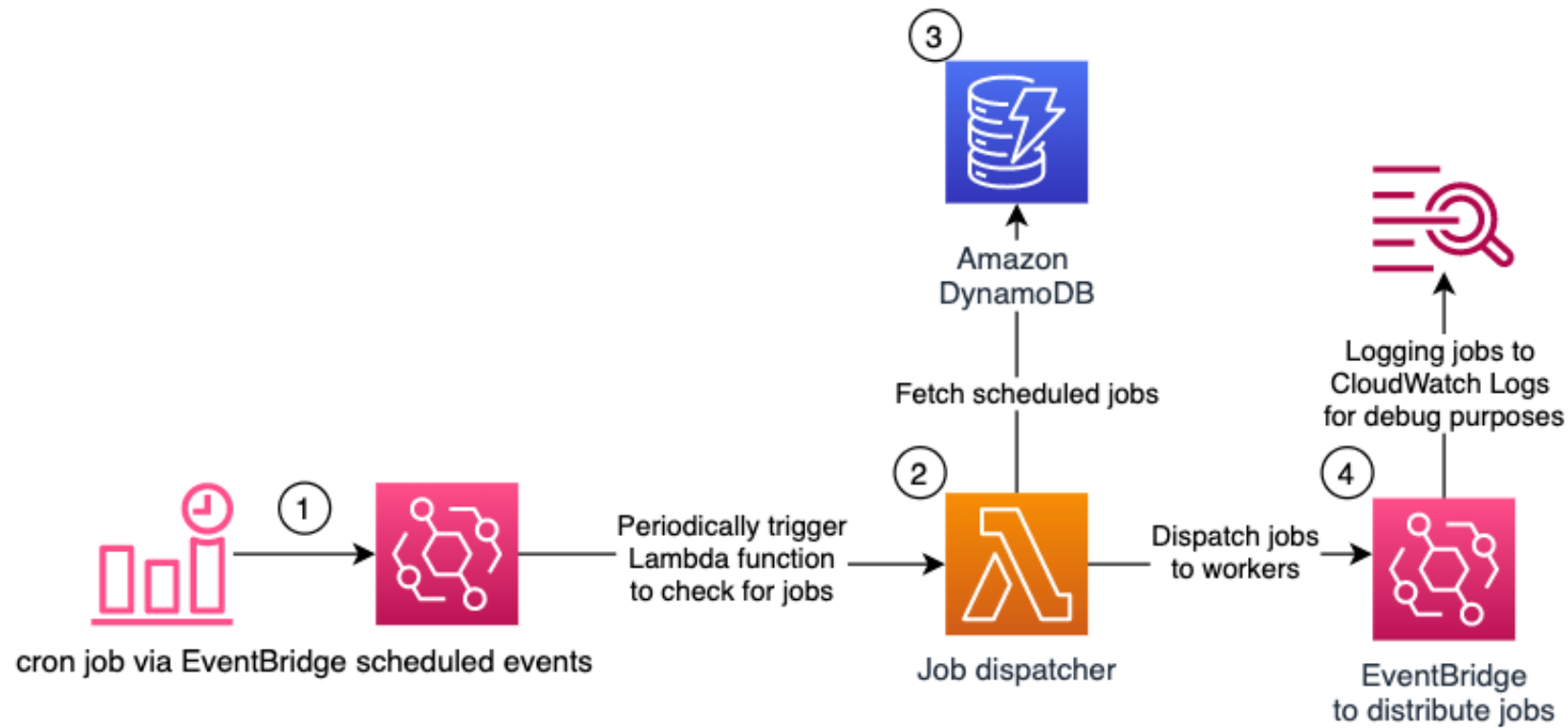
9

# Eventbridge Patterns

## Event Processing Conditional Retry Pattern



# Eventbridge Patterns: cron



<https://aws.amazon.com/blogs/architecture/serverless-scheduling-with-amazon-eventbridge-aws-lambda-and-amazon-dynamodb/>



# Interesting links

<https://serverlessland.com/whats-new?services=amazon-eventBridge>

<https://serverlessland.com/patterns?services=eventbridge>

David Boyney (Joined AWS as developer advocate serverless)

<https://github.com/boyney123/awesome-eventbridge>

<https://www.boyney.io/projects>

<https://www.eventbridgebook.com/>



Create an AppSync API with an EventBridge event bus

This project contains a sample pattern for using an Amazon EventBridge event bus as a HTTP data source with an AWS AppSync API.

```
import { Stack, StackProps, CfnOutput } from 'aws-cdk-lib';
import { Construct } from 'constructs';
import { GraphQLApi, Schema, FieldLogLevel, AuthorizationType } from 'aws-cdk-lib/aws-appsync';
import { Role, ServicePrincipal, PolicyStatement } from 'aws-cdk-lib/aws-iam';
import { EventBus } from 'aws-cdk-lib/aws-events';
import { join } from 'path';
```

# EventBridge Transformer

## Amazon EventBridge Transformer

Quickly generate input transforms for Amazon EventBridge.

Enter your event, select properties to map, profit.

Show Example Output

### Event Payload →

Click on the properties you want to map.

```
{
  "version": "0",
  "id": "0d079340-135a-c8c6-95c2-41fb8f",
  "detail-type": "OrderCreated",
  "source": "myapp.orders",
  "account": "123451235123",
  "time": "2022-02-01T18:41:53Z",
  "region": "us-west-1",
  "detail": {
    "metadata": {
      "correlation_id": "dddd9340-1",
      "service": "my-service",
      "domain": "SHOP"
    },
    "data": {
      "amount": "19.99",
      "quantity": "2",
      "orderId": "0d079340-535a-c8c",
      "userId": "9e223efa-e318-4e06"
    }
  }
}
```

PROBLEMS 0 VALID

### Input Path →

Define variables as key value pairs.

```
{
  "account": "$.account",
  "service": "$.detail.metadata.service",
  "amount": "$.detail.data.amount"
}
```

PROBLEMS 0 VALID

### Input Template →

Define variables as key value pairs.

```
{
  "account": "<account>",
  "service": "<service>",
  "amount": "<amount>"
}
```

PROBLEMS 0 VALID

### Example Output

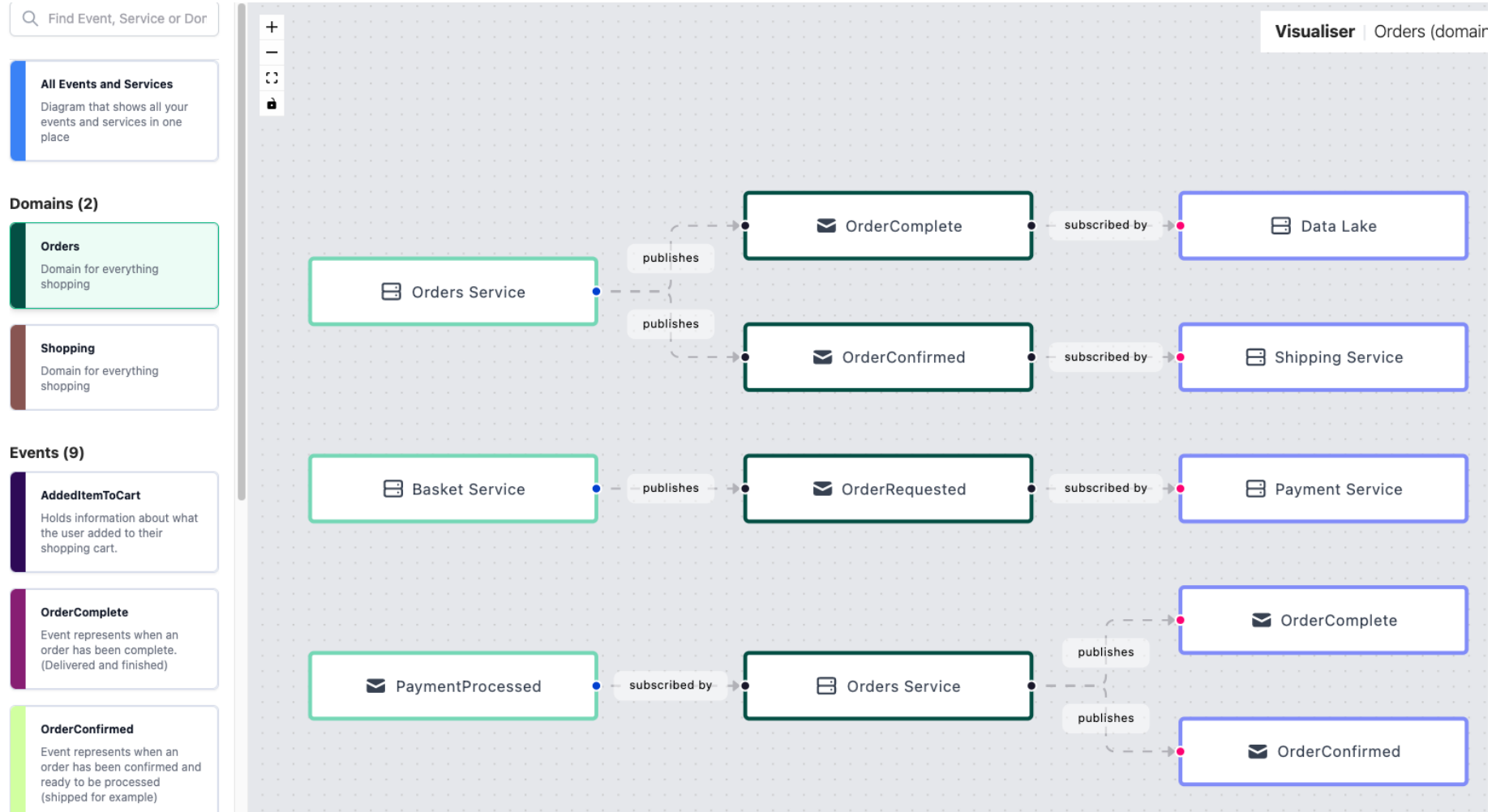
Example output based on transform rules.

```
{
  "account": "123451235123",
  "service": "my-service",
  "amount": "19.99"
}
```

PROBLEMS 0 VALID

<https://eventbridge-transformer.vercel.app/>

# EventCatalog (1/2)



<https://www.eventcatalog.dev/>

# EventCatalog (2/2)

## Payment Service

Edit

Event based application that integrates with Stripe.

The payment service is our own internal payment service that listens to events from stripe and handles them within our own domain.

We use Stripe to handle services and use this Payment service to enrich events for internal use.

### Payment service API 0.0.1 OAS3

A Payment Service API

Servers

http://payment.example.com/v1 - Optional server description, e.g. Main (production) server

#### PaymentServices

GET /getSinglePaymentURL Generate a Payment endpoint URL

+  
-  
⌂  
🔒

OrderRequested

Payment Service

PaymentProcessed

PaymentProcessed

Subscribes to Events (1)

OrderRequested

Domain Owners (1)

David Boyne

View in Visualiser

**Powerful tool in serverless architecture**

**Async vs stepfunction sync**

**AWS change events via Cloudtrail**

**SaaS / External events input**

**Output via HTTP api's**

**Move logic out of lambda's**

Conclusions







# Questions



# Next Topics?

- New services
- Deep dives
- Explain Architecture and lessons learned
- Development or Infrastructure