



# Event-driven architectures on AWS

Emily Shea

WW Head of Application Integration Go-To-Market  
Amazon Web Services

# About me

- From Seattle
- Living in London since 2021
- 5 years in AWS serverless & integration
- Currently lead global go-to-market strategy for Application Integration



@em\_shea



© 2023, Amazon Web Services, Inc. or its affiliates.



A man with a shaved head and a beard, wearing a dark t-shirt with an orange logo, stands on a stage. He is gesturing with his hands while speaking. Behind him is a large screen displaying a blue-toned image of a landscape with mountains and clouds. Overlaid on the screen is white text that reads "Event-driven architectures enable global scale".

Event-driven architectures  
enable global scale

AWS re:Invent 2022 keynote

# An **event** is a change in state or an update.



## Retail

*New order placed  
(or donation placed)*



## Media & Entertainment

*Video uploaded for processing  
(or content uploaded / updated)*



## Financial Services

*Financial analysis requested  
(or payment received)*



## Health Care

*Patient medical data updated  
(user record updated)*



## Publication

*Document uploaded for translation*



## Education

*New student enrolled*



## Manufacturing

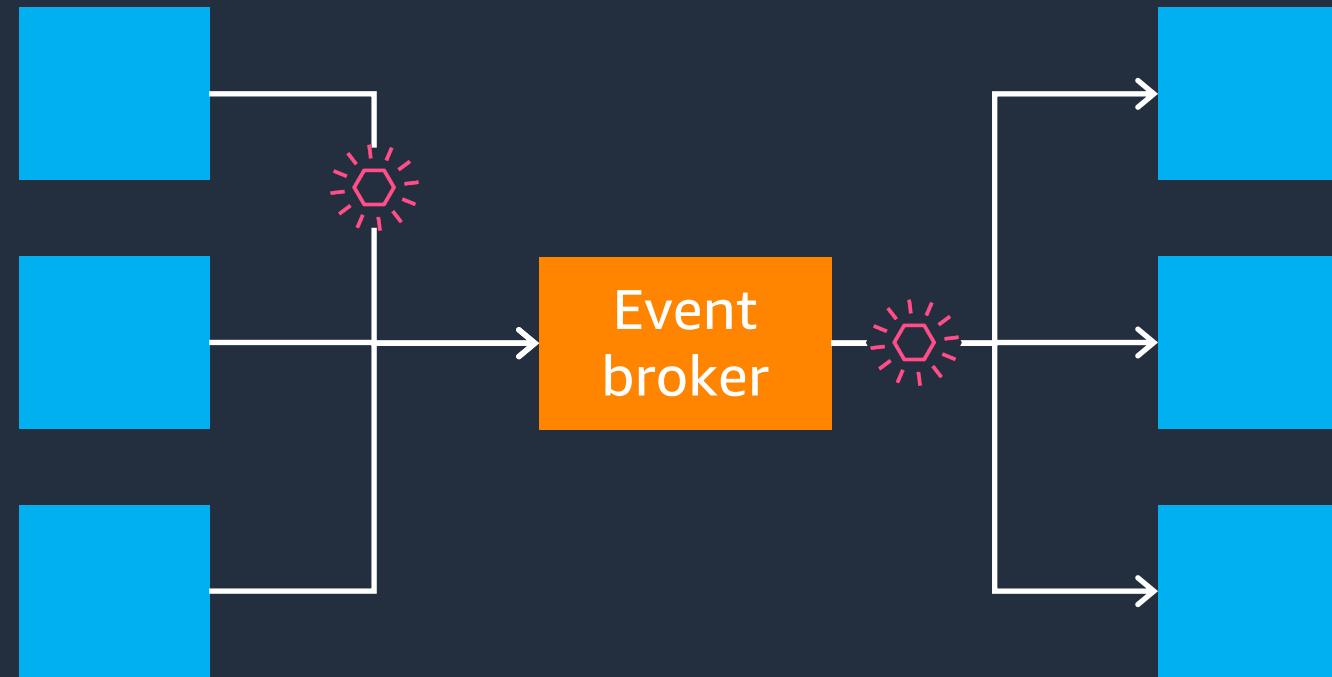
*Issue detected from IoT sensor*



## Government

*New application submitted*

# In an event-driven architecture, systems and development teams **communicate** via events



# Why build event-driven?

# Event driven architectures can give you:



Greater developer  
agility and extensibility



Increased scalability  
and fault tolerance



Lower total cost of  
ownership

# Event driven architectures can give you:



Greater developer  
agility and extensibility

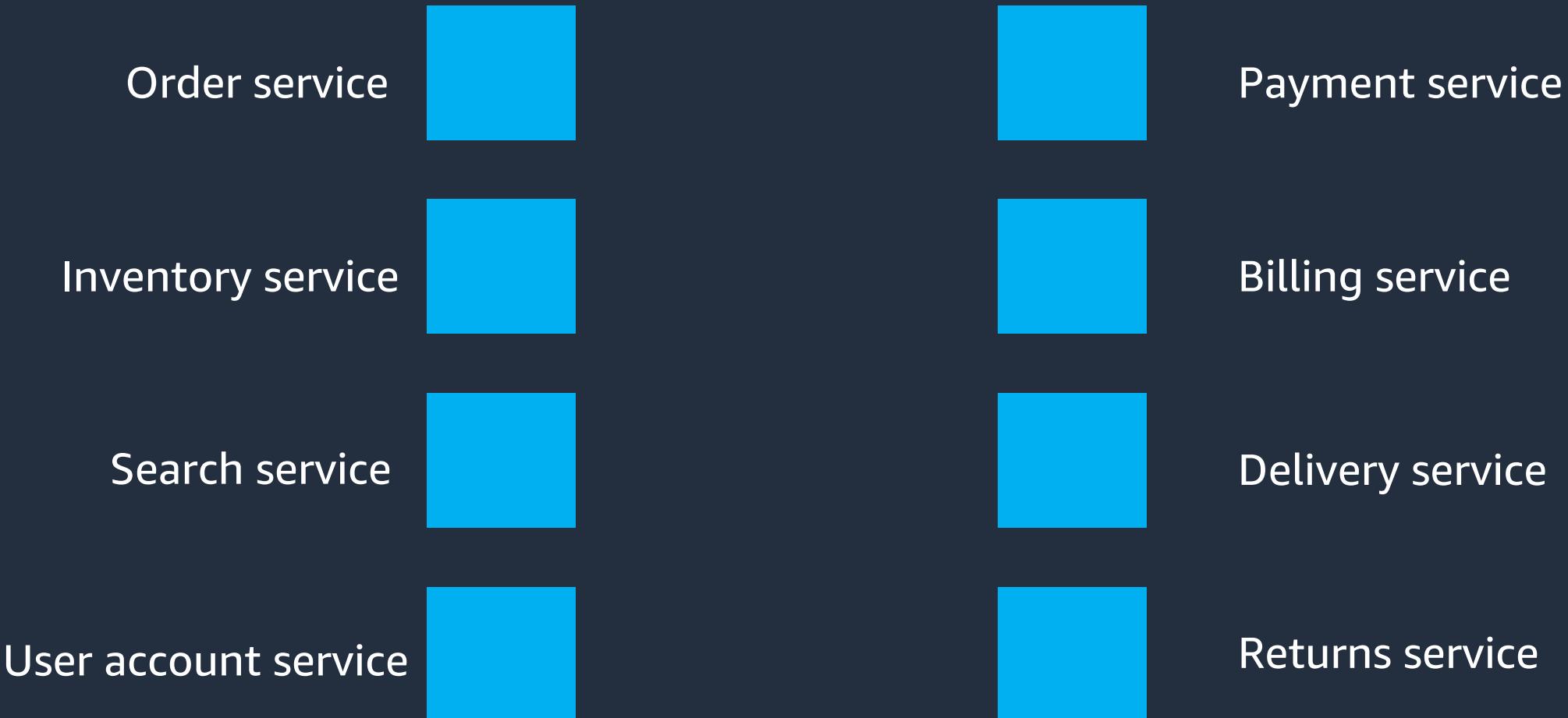


Increased scalability  
and fault tolerance

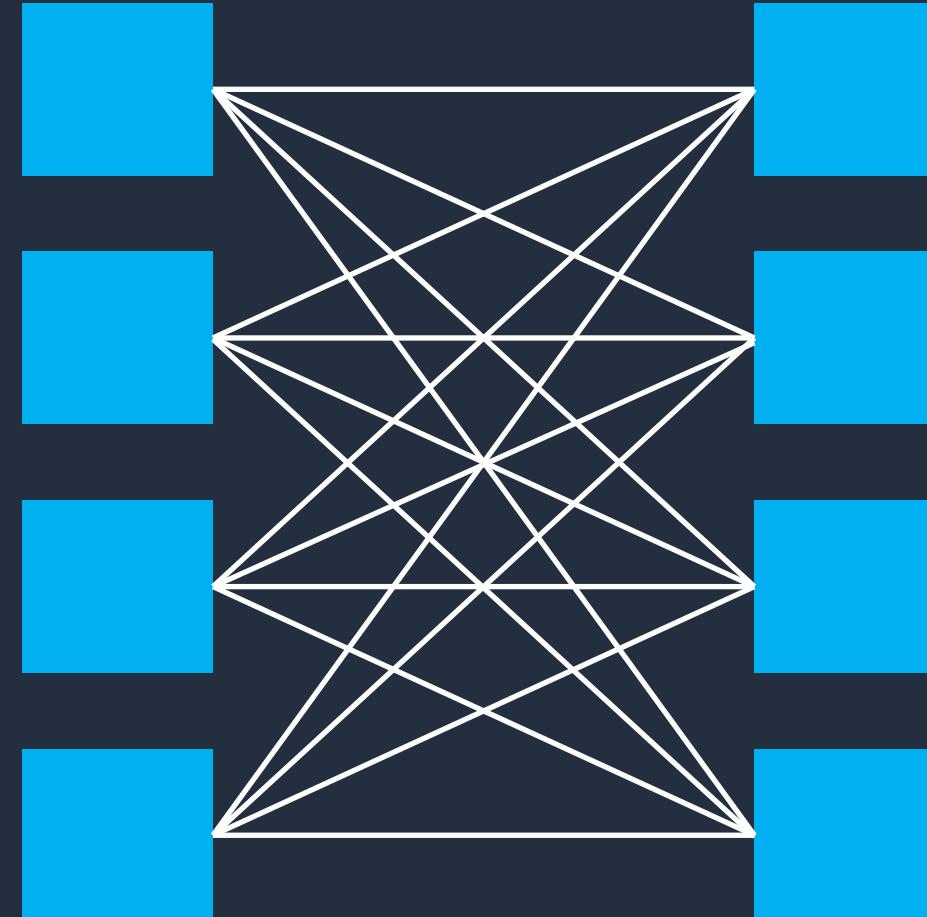


Lower total cost of  
ownership

# A microservice application



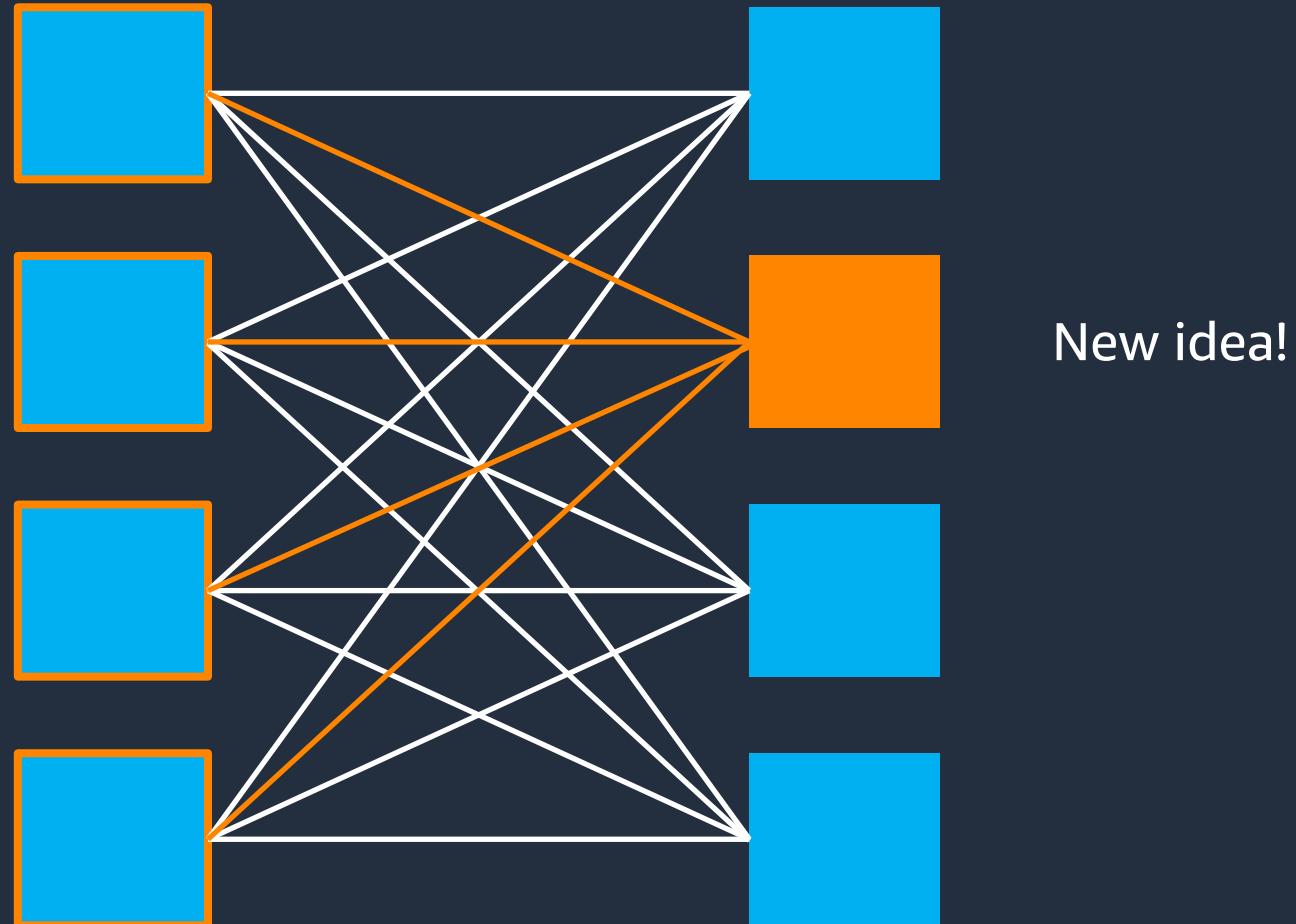
# Interdependent teams



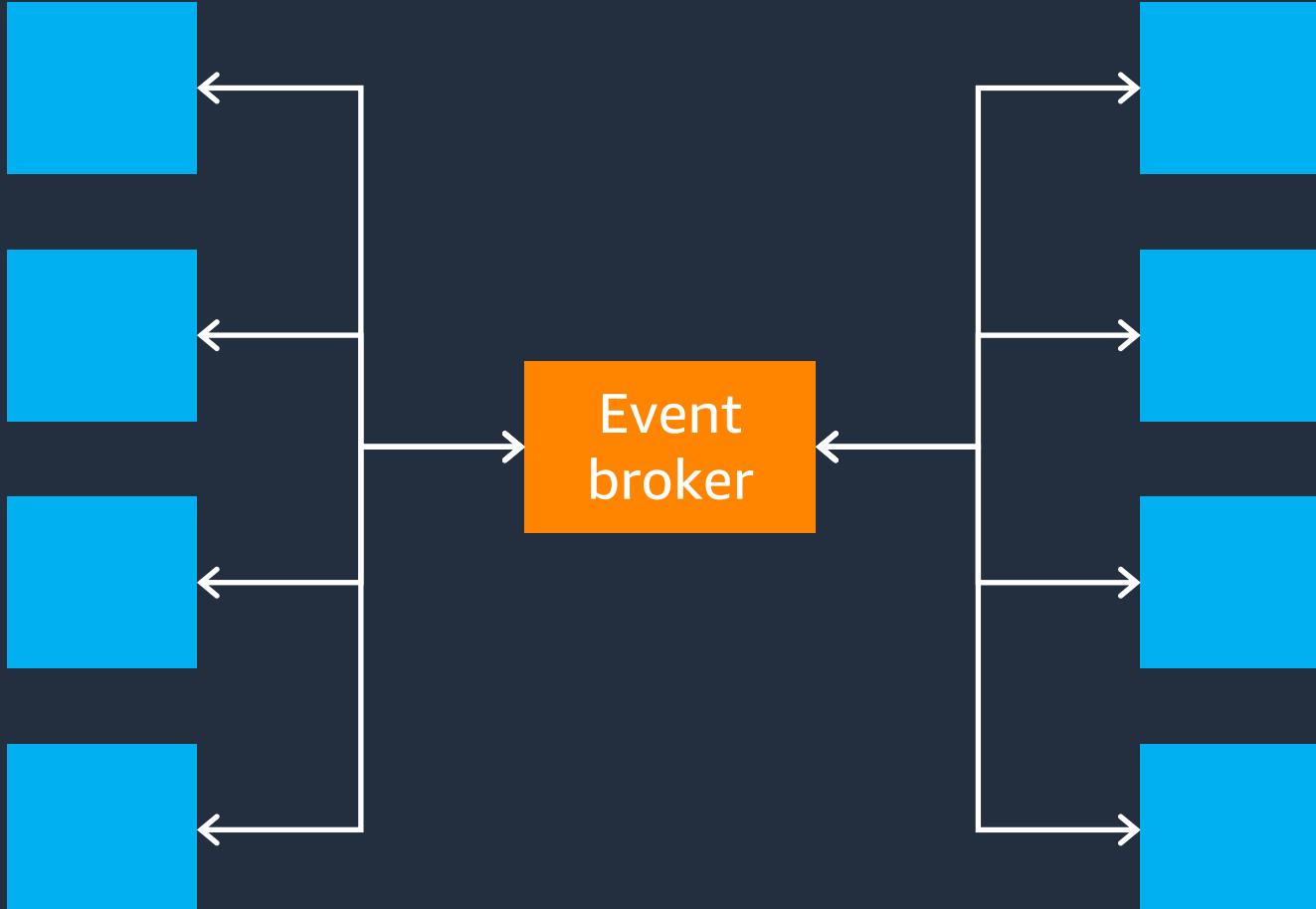
# Interdependent teams



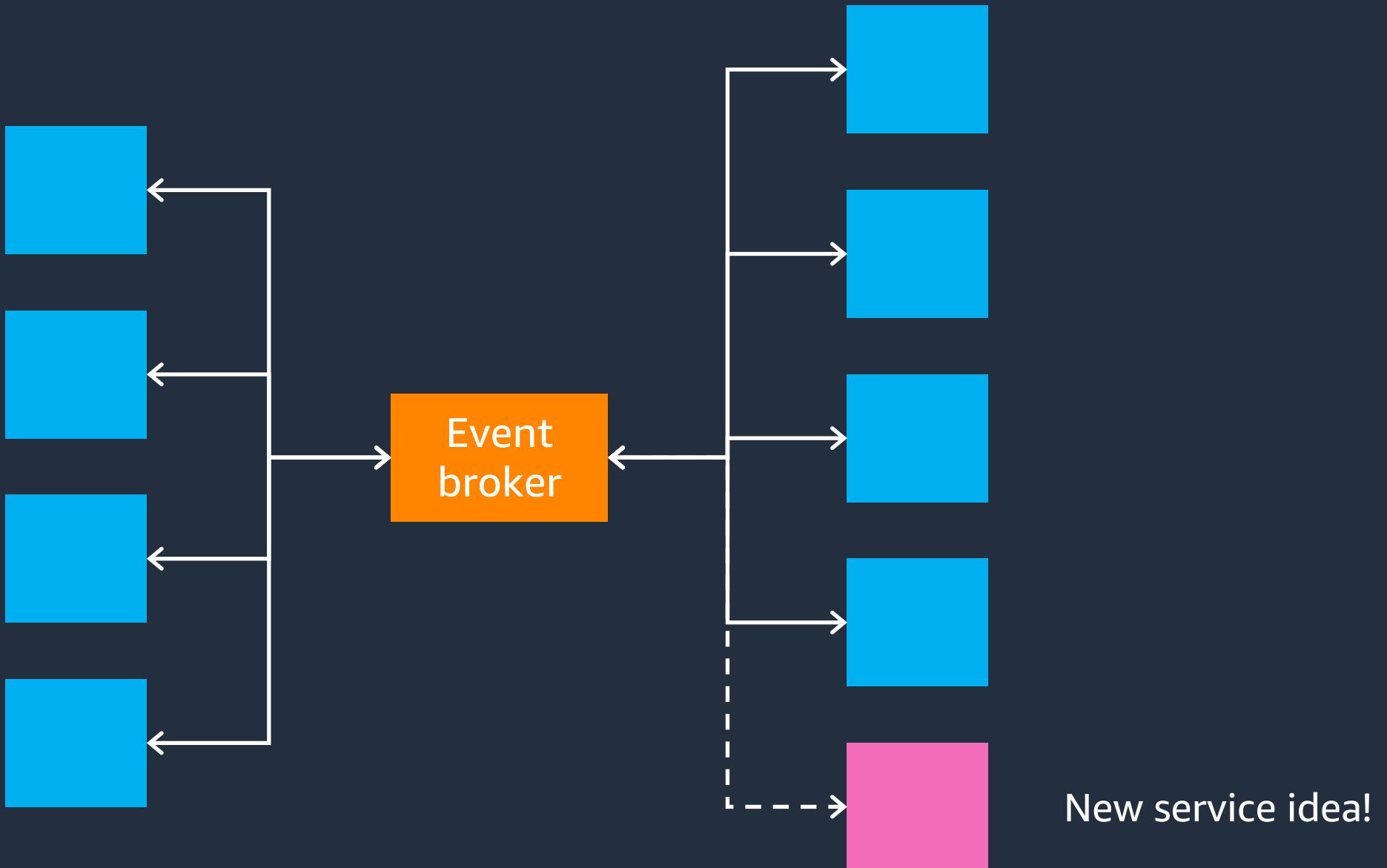
# Interdependent teams... a slower pace of development



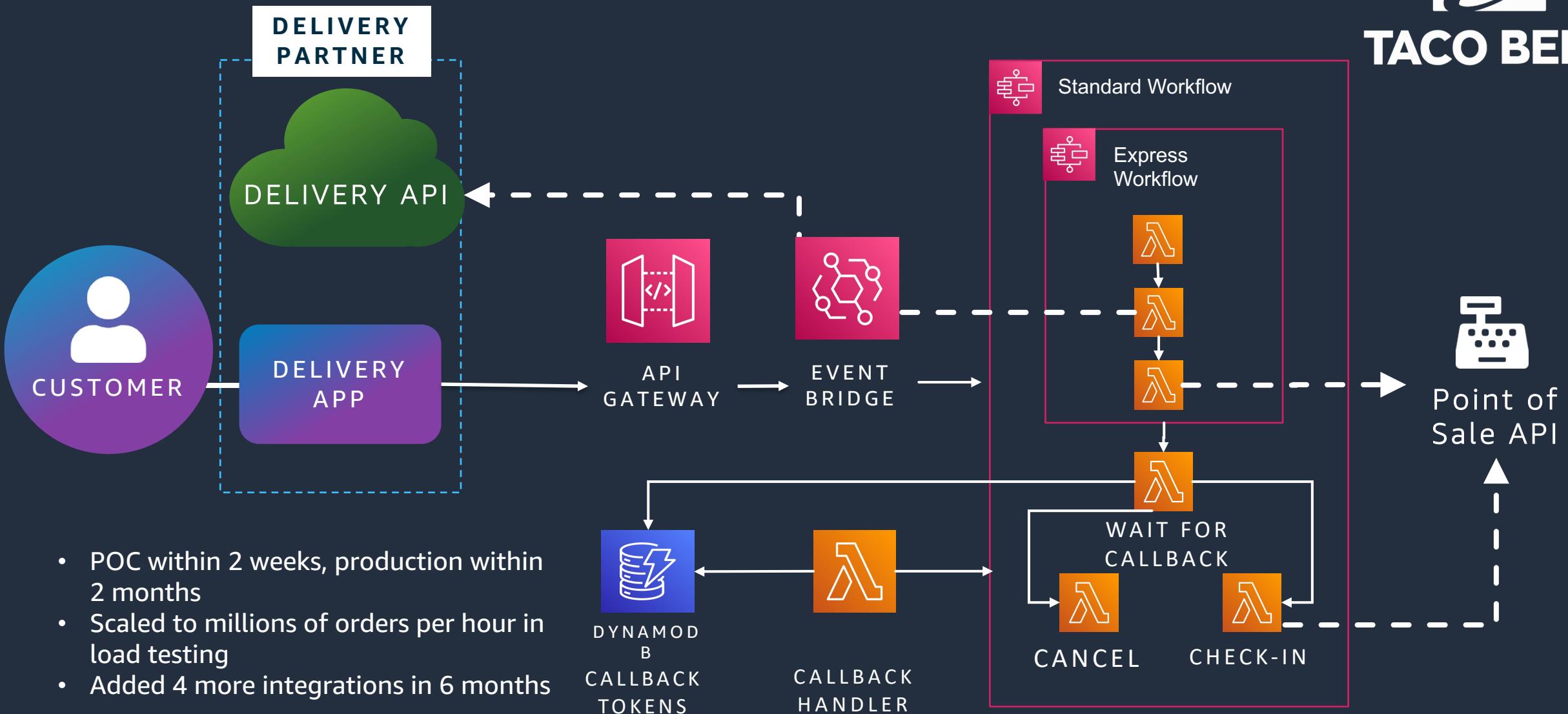
# A central event broker reduces the need for tight coordination between teams



# New services and features can be built on top of existing events



# Taco Bell's event-driven delivery order solution



# Event driven architectures can give you:



Greater developer  
agility and extensibility



Increased scalability  
and fault tolerance

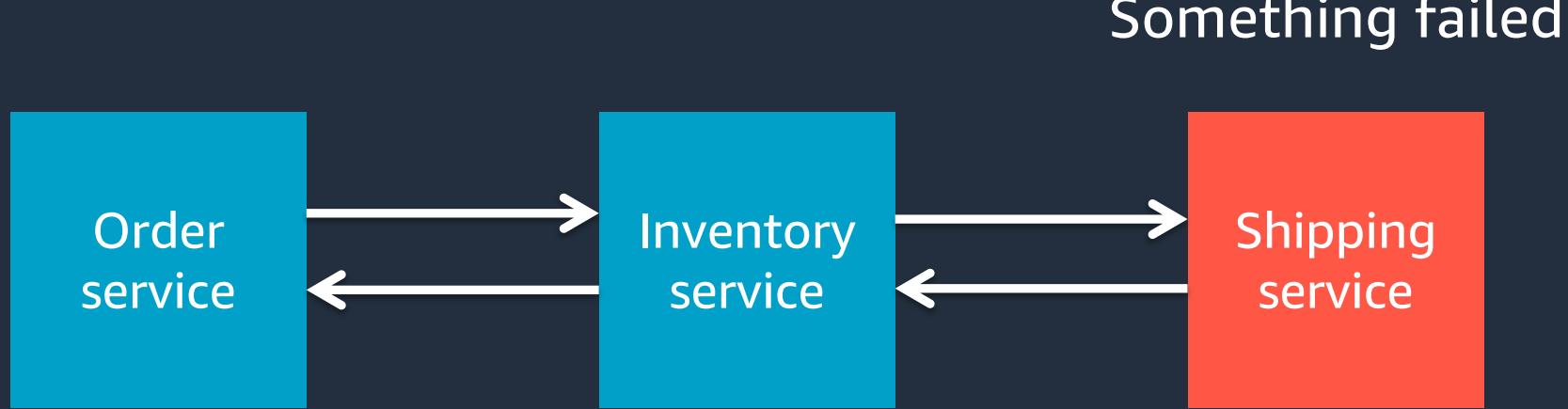


Lower total cost of  
ownership

# Synchronous calls to downstream services



# Synchronous calls to downstream services

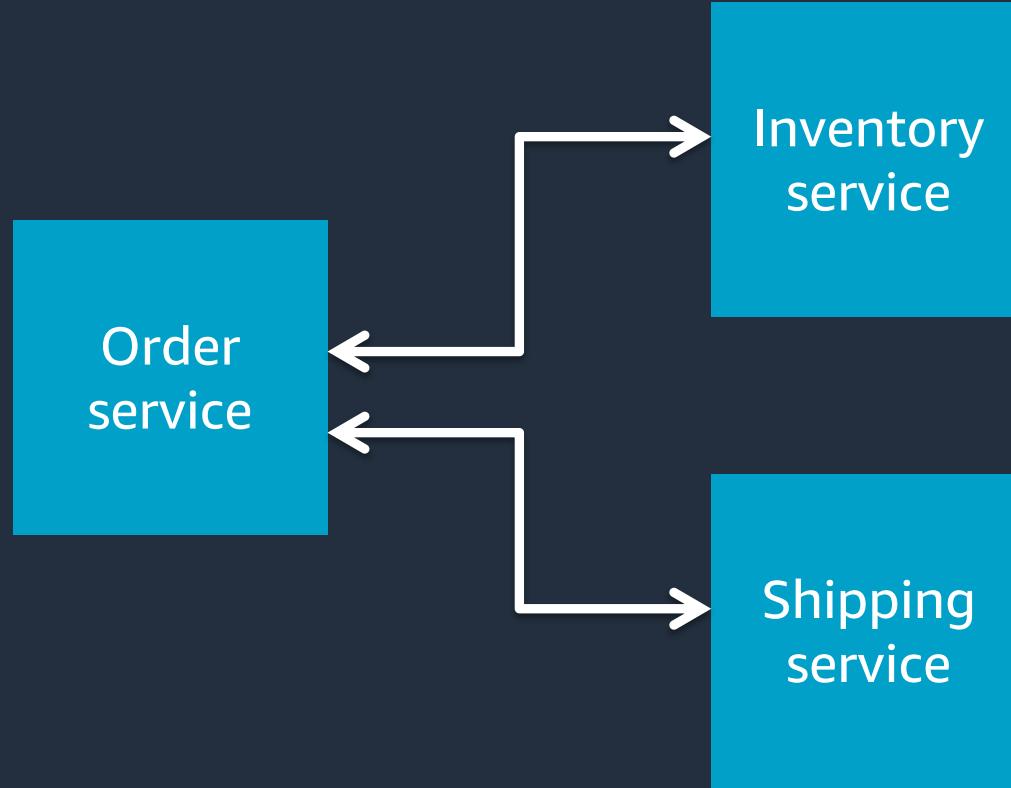


# Synchronous calls to downstream services

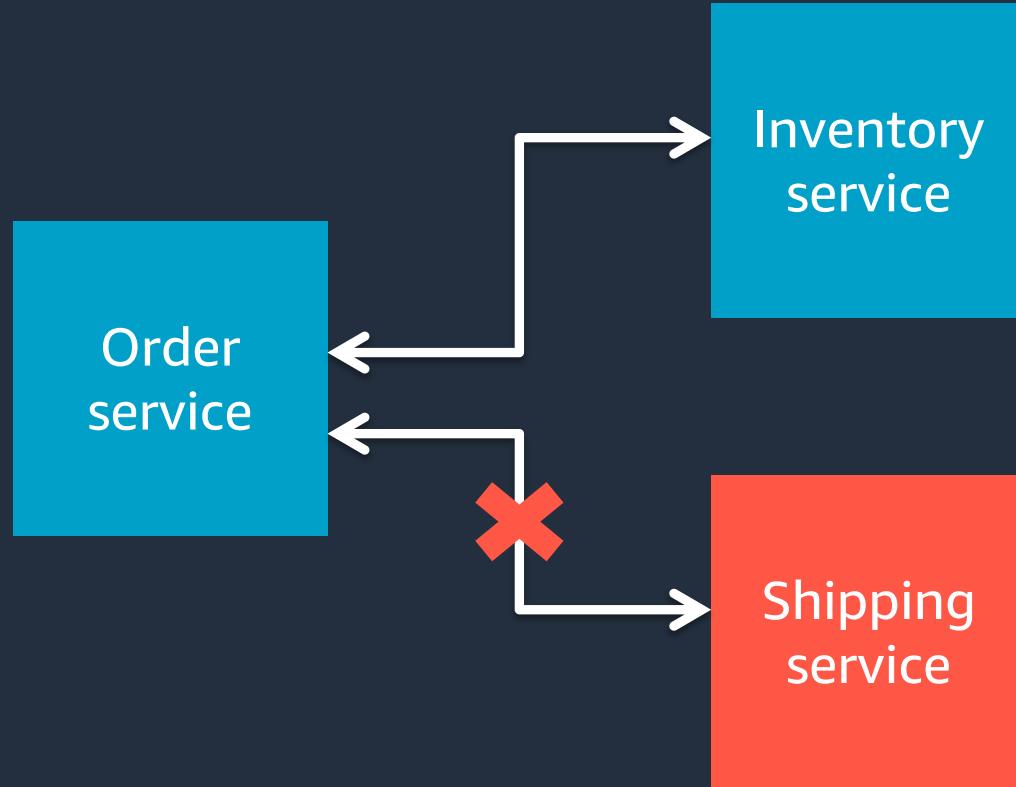
Everything failed



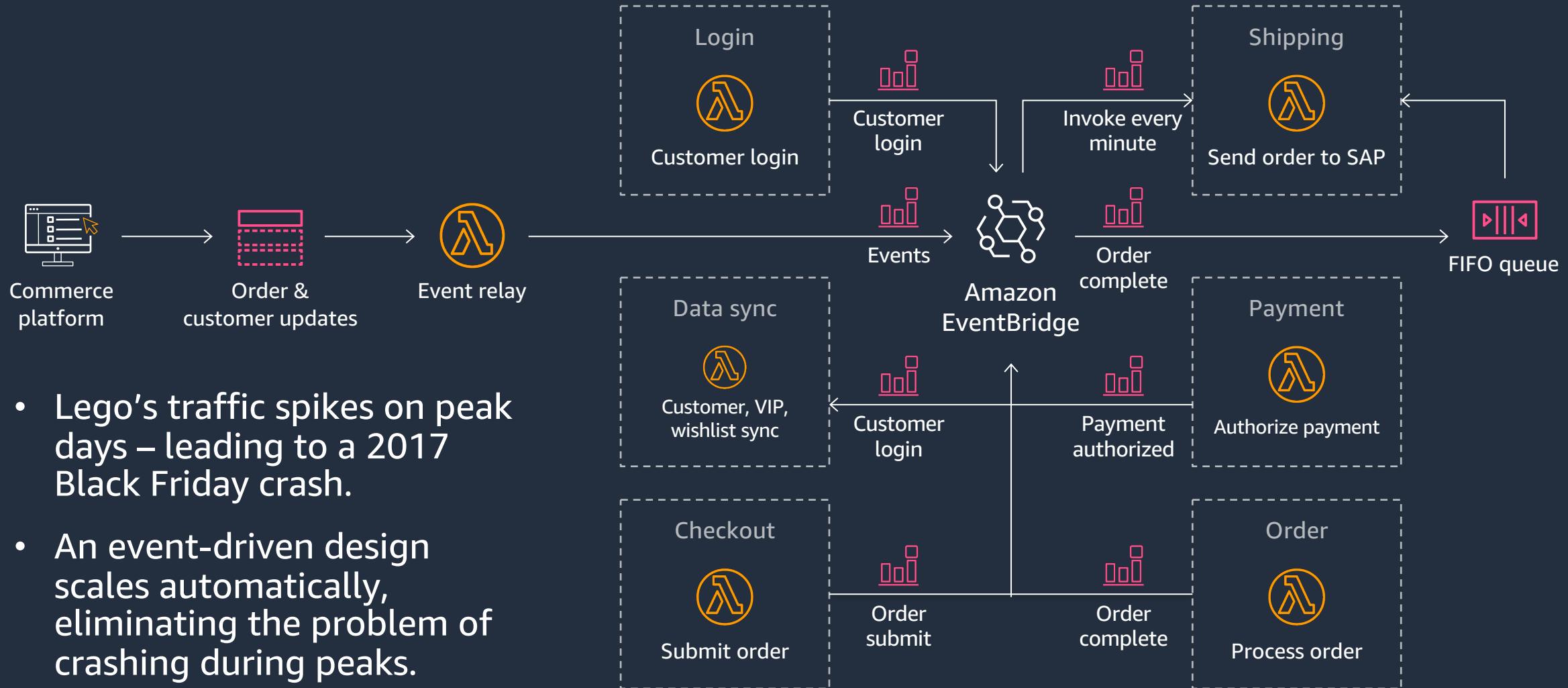
# Asynchronous calls to downstream services



# Asynchronous calls to downstream services



# Lego uses an event-driven design for scalability



# Event driven architectures can give you:



Greater developer  
agility and extensibility



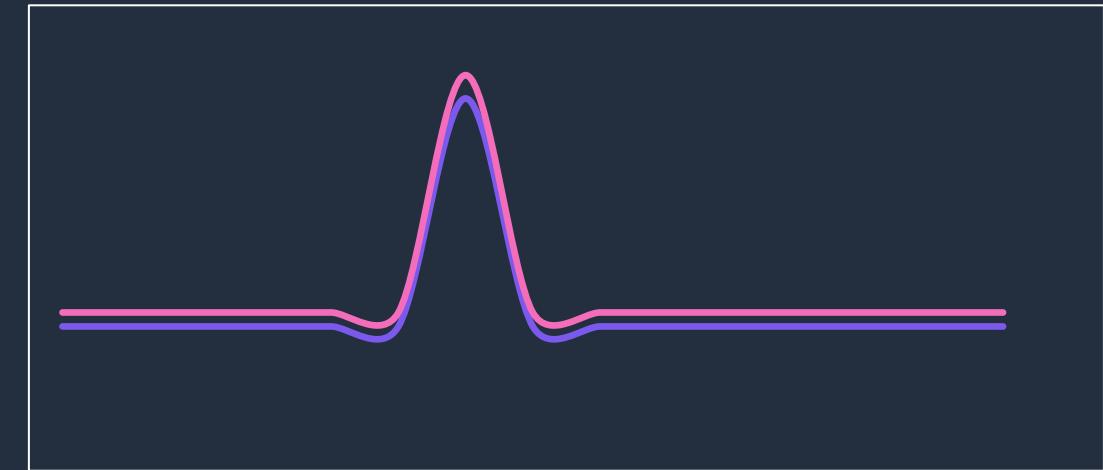
Increased scalability  
and fault tolerance



Lower total cost of  
ownership

# Serverless was designed for the event-driven model

- Only run when invoked by events
- Built-in features for working with events
- Focus on business logic, not infrastructure management



Playvox saw a **5x cost reduction** at scale per invocation.

Cinch reduced cost of their e-commerce infrastructure by **66%**.

Liberty Mutual processes **one million transactions** for just \$60.



# Which AWS service should I choose to build EDAs?



Amazon EventBridge



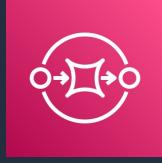
Amazon SNS



AWS Lambda



AWS Step Functions



Amazon SQS



Amazon Kinesis  
Data Streams

# Which AWS service should I choose to build EDAs?

Pub/sub messaging

Event bus



Amazon EventBridge



Amazon SNS



Event-driven compute

Workflows



AWS Step Functions



Amazon SQS



Event streams

Queues

# ~~Which AWS service should I choose to build EDAs?~~



Amazon EventBridge



Amazon SNS



AWS Lambda



AWS Step Functions



Amazon SQS

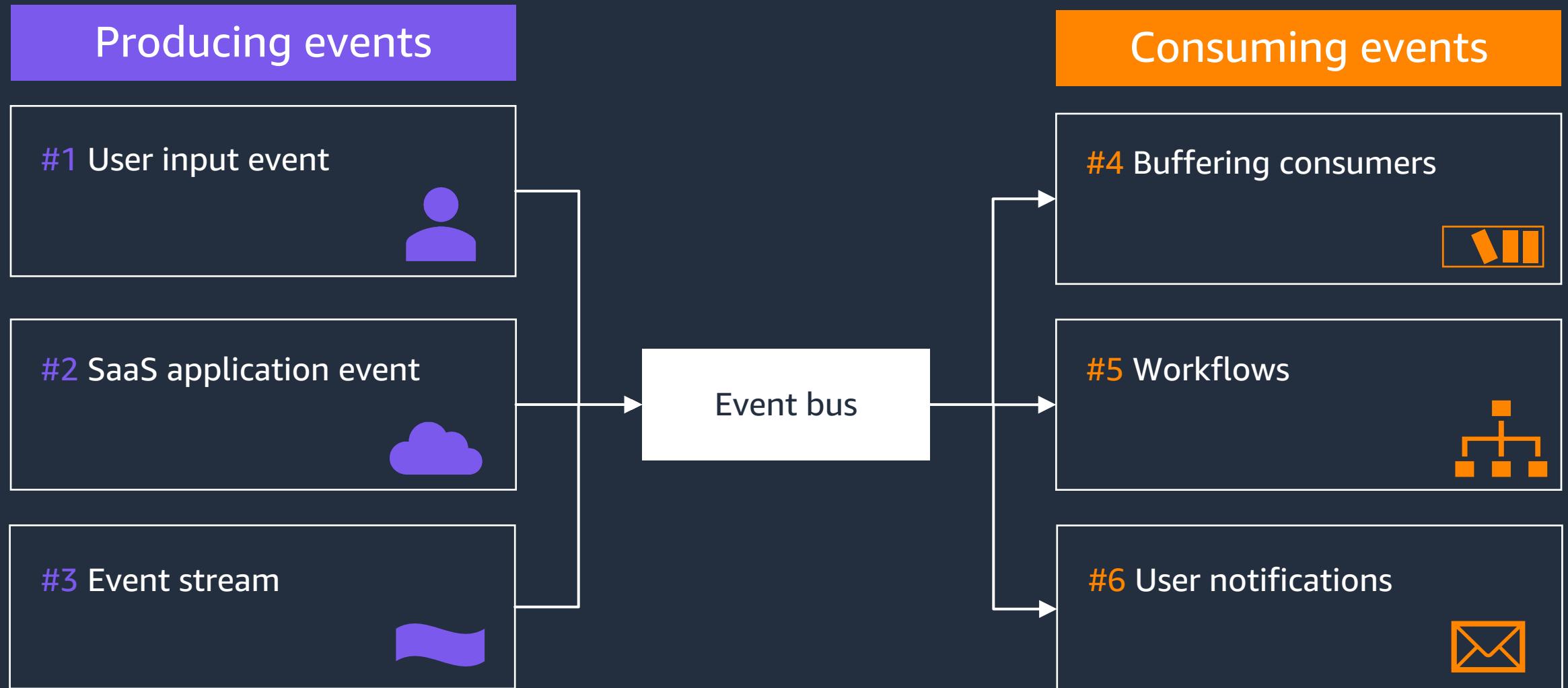


Amazon Kinesis  
Data Streams

## Which EDA patterns fit my use case?

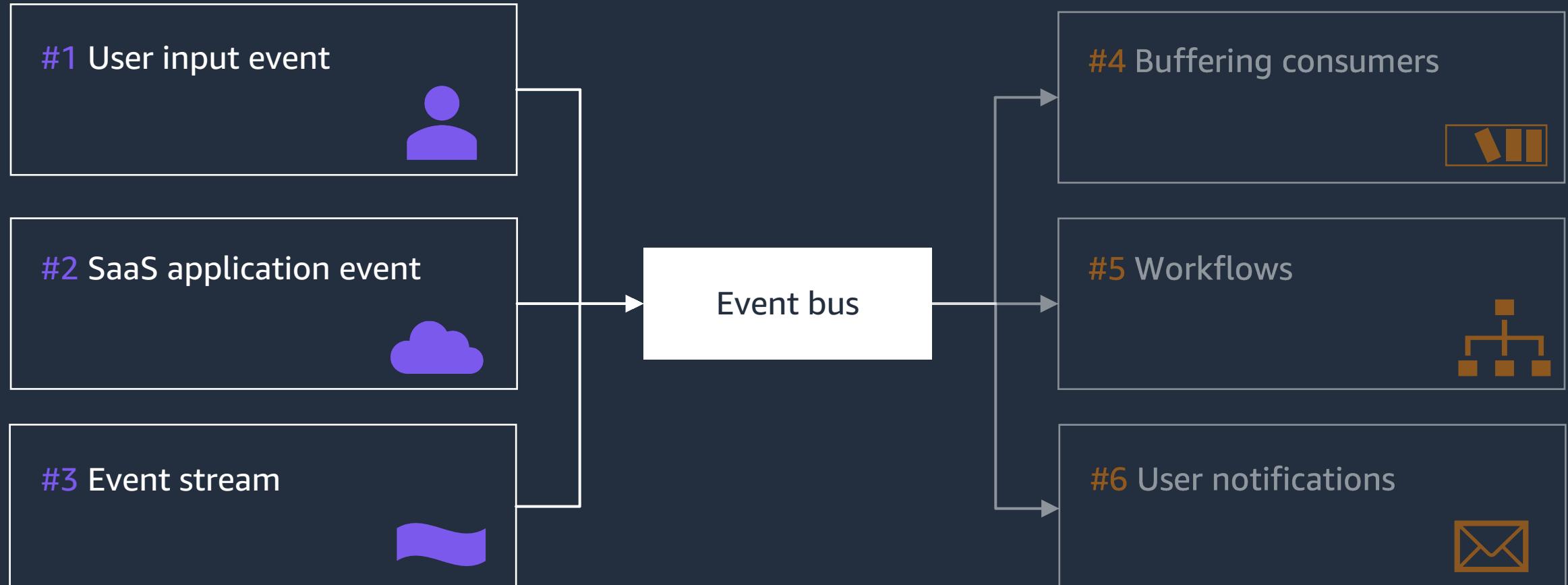
# Event-driven on AWS: 6 patterns to know

# 6 patterns to know



# 6 patterns to know

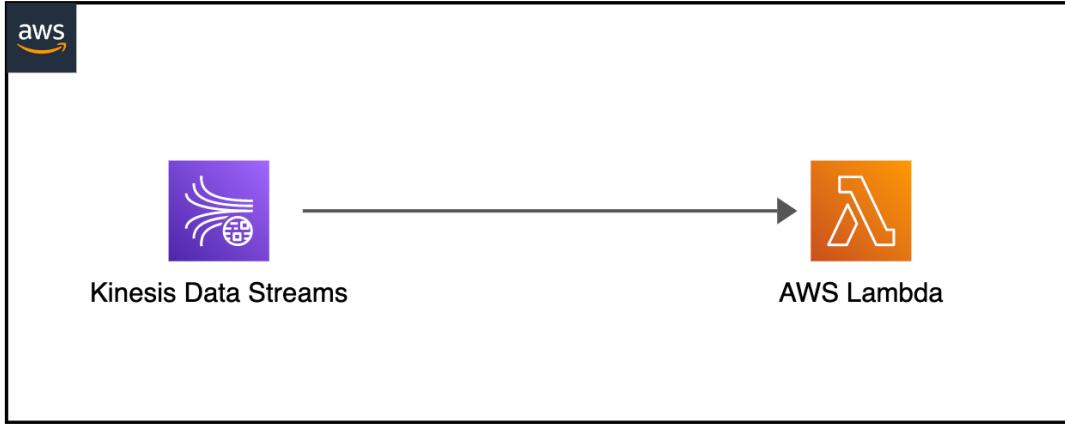
## Producing events



# Serverless Land Patterns – 560+ deployable patterns

 **Serverless Land**

Content ▾ Learn ▾ Code ▾ EDA ▾  Search



```
graph LR; KDS[Kinesis Data Streams] --> AWSLambda[AWS Lambda]
```

**New Serverless testing content!**

Guides and serverless testing patterns to help you understand how you can test your serverless applications.

[Start learning →](#)

[< Back to patterns](#)

**Create an AWS Kinesis Data Stream that invokes a Lambda function**

The SAM template deploys a Kinesis Data Stream, a Lambda function and the IAM resources required to run the application. As data enters the stream, the Lambda function is invoked and decodes each record in the batch and logs the record to AWS CloudWatch.

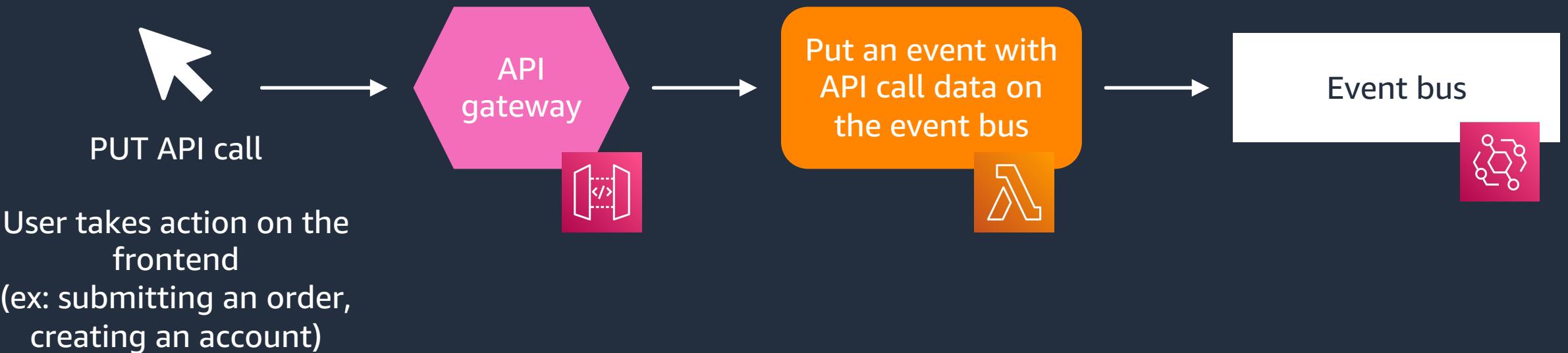
**Download**

```
> git clone https://github.com/awslabs/serverless-patterns/kinesis-data-stream-invokes-lambda
```

**Deploy**

```
> sam deploy --guided
```

# #1 User input event



[serverlessland.com/patterns/apigw-lambda-eventbridge-sam-java](https://serverlessland.com/patterns/apigw-lambda-eventbridge-sam-java)

## #2 SaaS application event

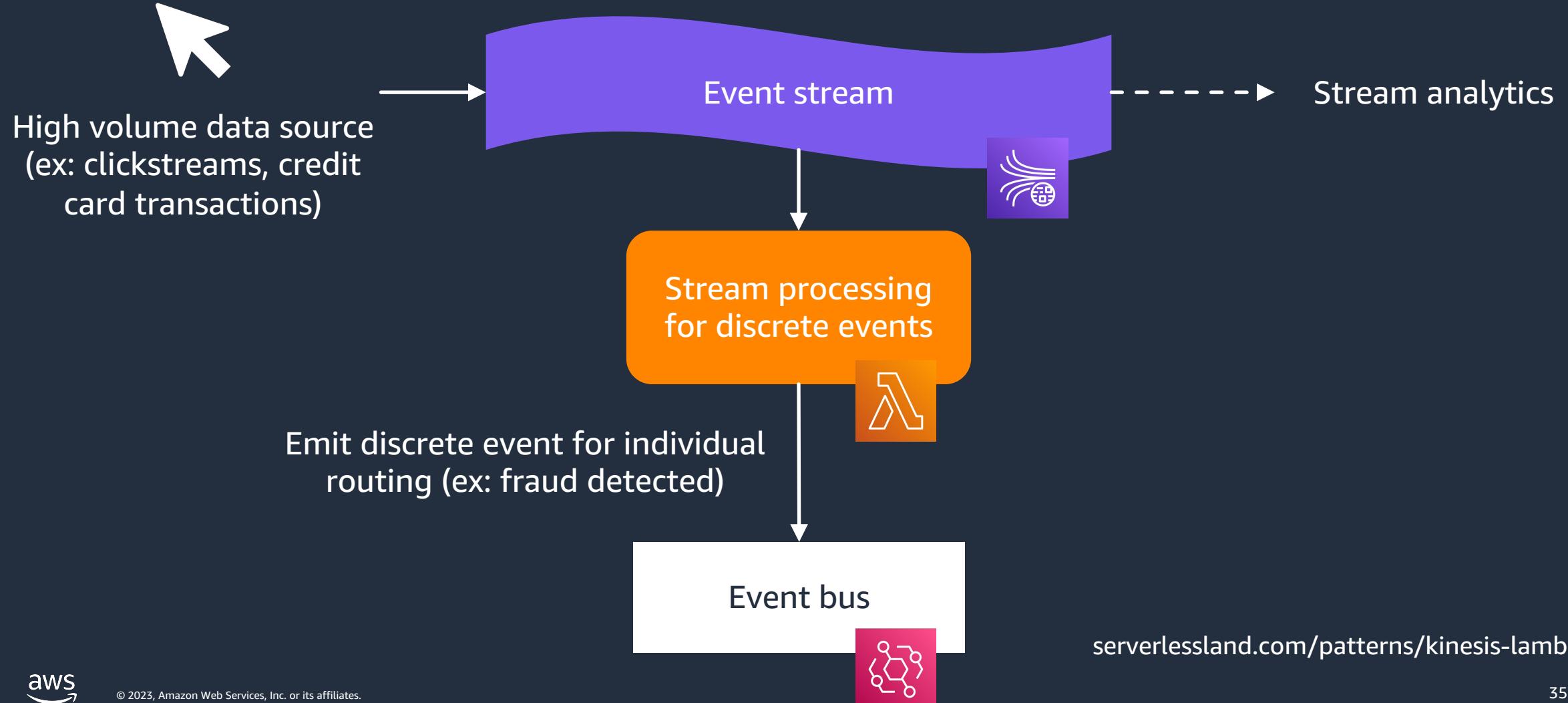


SaaS application emits an event  
(ex: contact created, code committed, payment received)



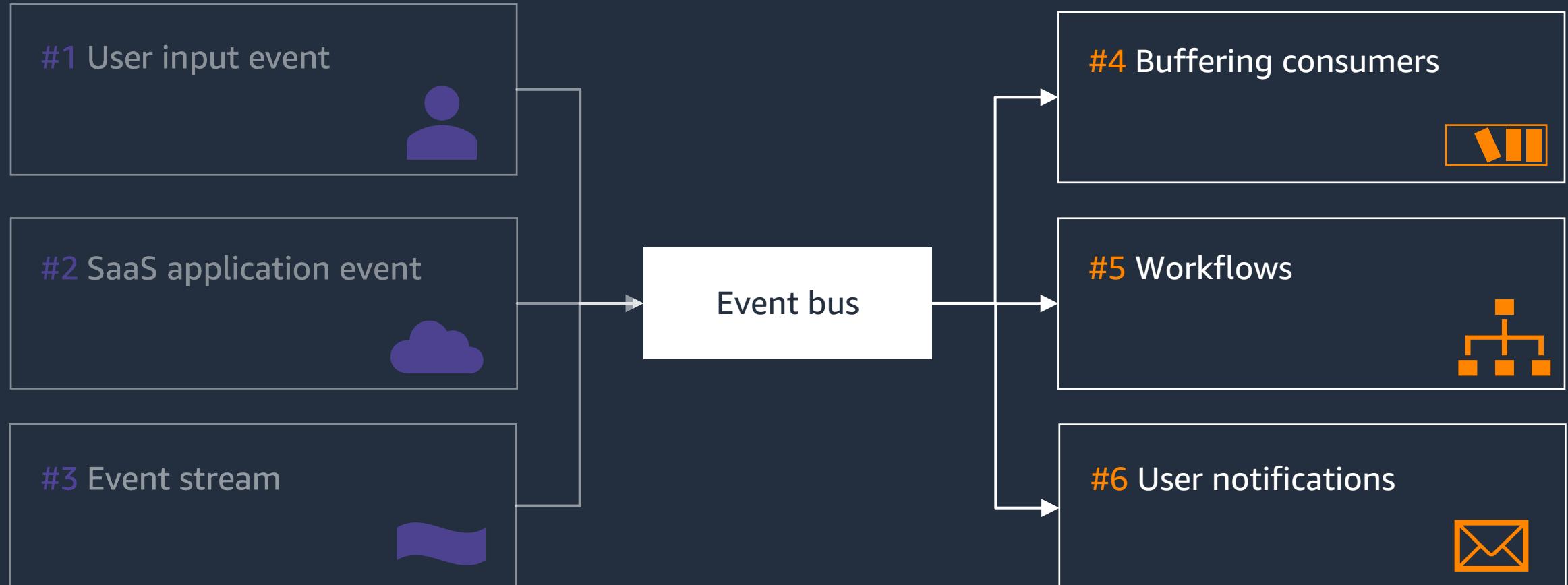
[s12d.com/salesforce-eventbridge](https://s12d.com/salesforce-eventbridge)

# #3 Event stream



# 6 patterns to know

## Producing events



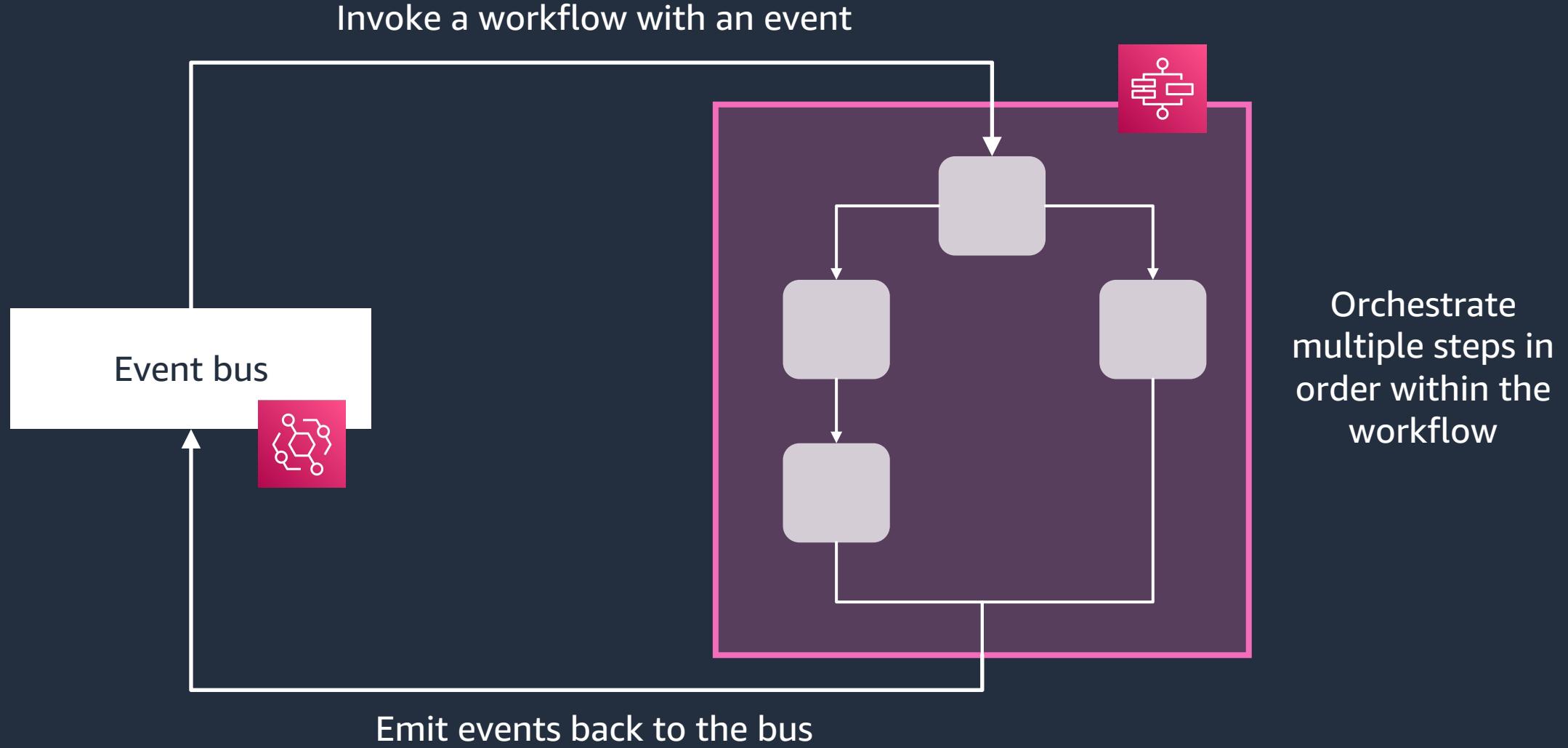
# #4 Buffering consumers



Buffer events in a queue to  
avoid overwhelming  
downstream dependencies

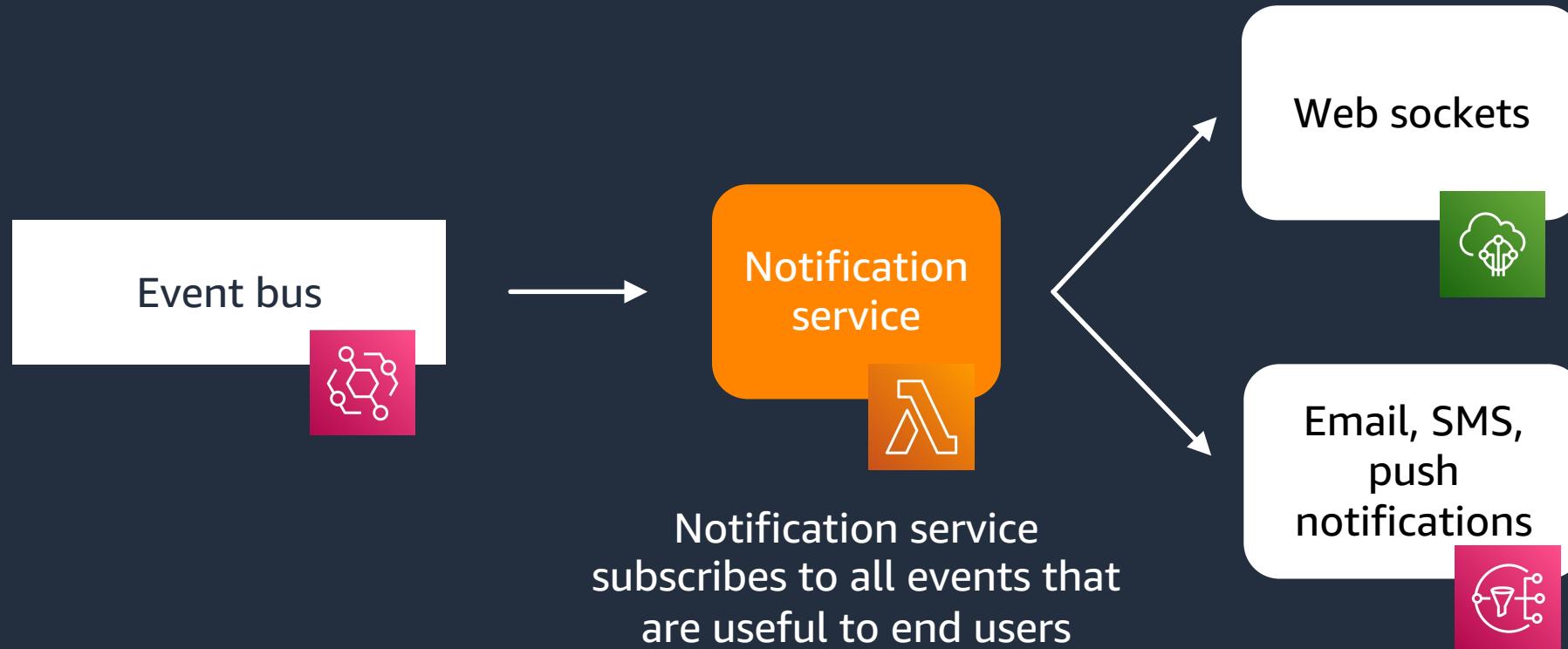
[serverlessland.com/patterns/eventbridge-sqs-ecs-cdk](https://serverlessland.com/patterns/eventbridge-sqs-ecs-cdk)

# #5 Workflows



[serverlessland.com/patterns/eventbridge-sfn](https://serverlessland.com/patterns/eventbridge-sfn)

# #6 User notifications



<https://serverlessland.com/patterns/lambda-sns-sms>

# Bringing it all together: Insurance claims processing

Allow users to sign up and submit documents and claim information

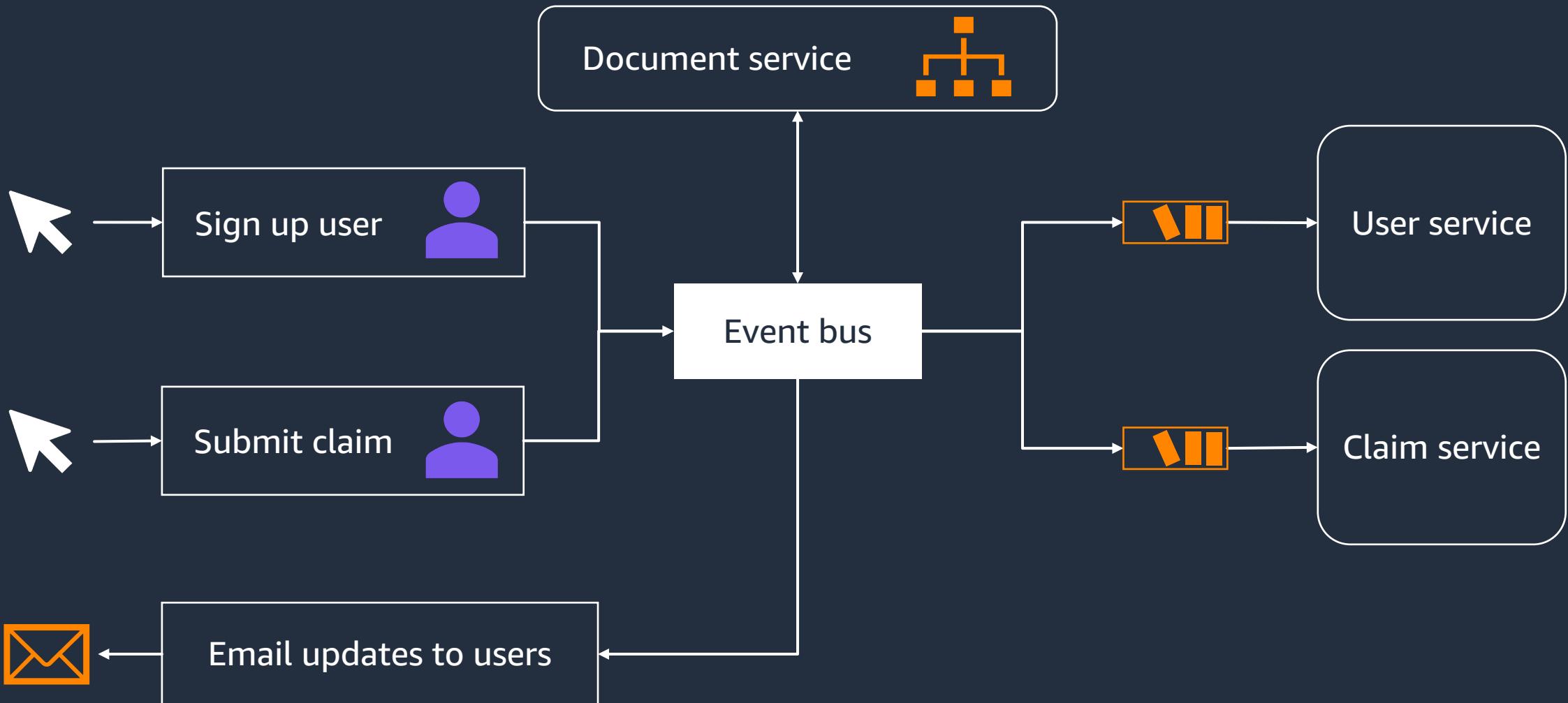
Send notifications to users on status of claim

Process documents (ex: driver's licenses, damage images)

Process new user sign up

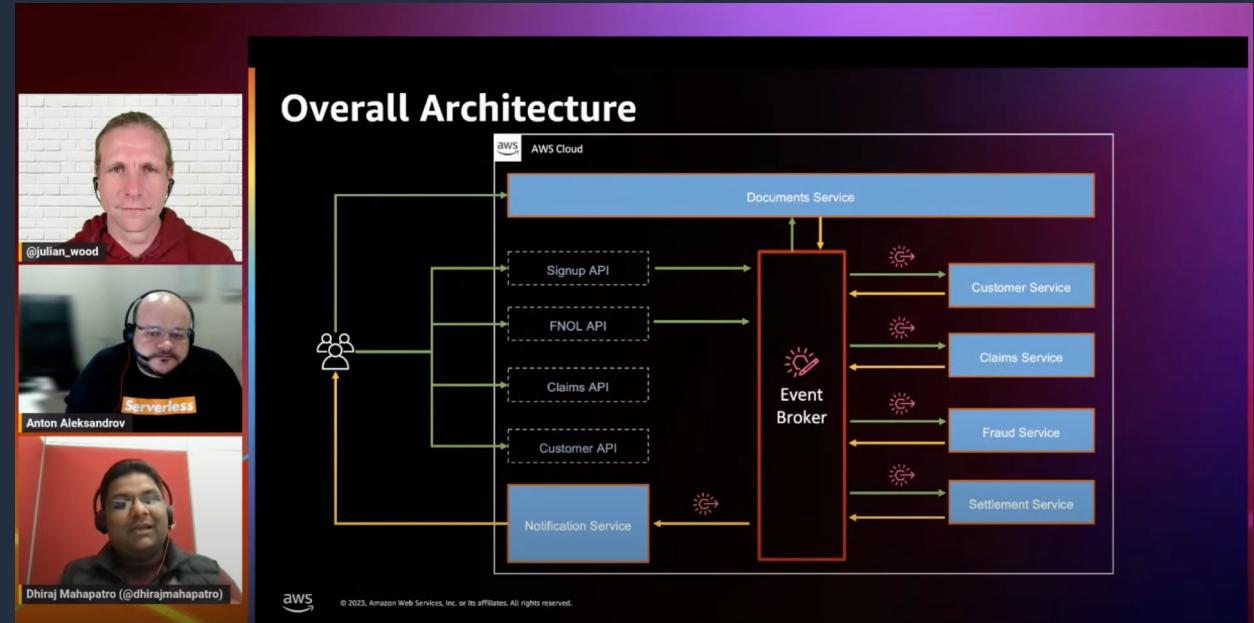
Process new claim

# Bringing it all together: Insurance claims processing



# Explore the insurance claims processing app

- Blog series
- GitHub repo
- Serverless office hours demo



## Building a modern, event-driven application for insurance claims processing – Part 2

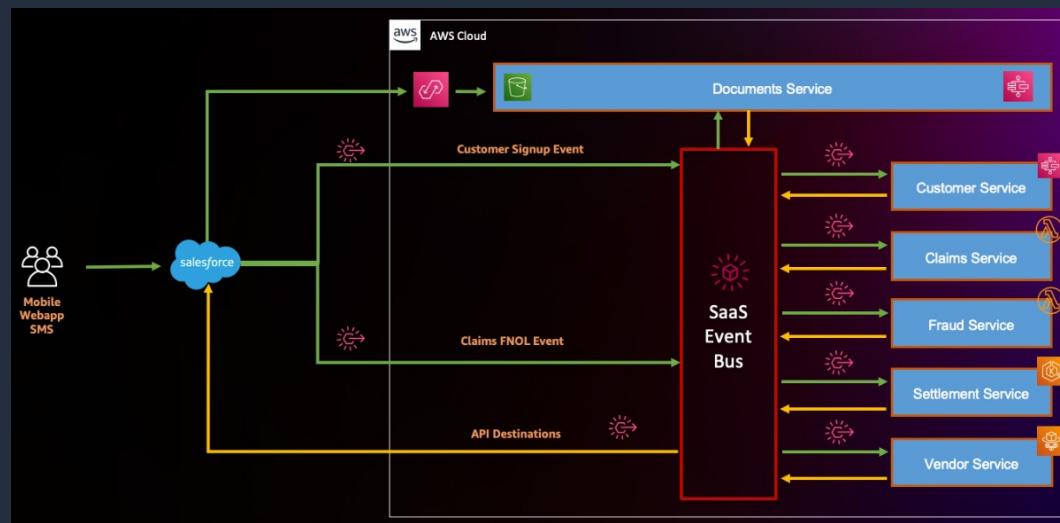
by Emily Shea, Vaibhav Jain, and Dhiraj Mahapatro | on 14 MAR 2023 | in [Amazon Simple Queue Service \(SQS\)](#), [Application Services](#), [AWS Lambda](#), [AWS Step Functions](#), [Compute](#), [Financial Services](#), [Industries](#), [Messaging](#) | [Permalink](#) | [Share](#)

In [Part 1](#) of this series, you learned how insurance claims processing makes a good industry use case for event-driven architectures. In Part 2, you dive deeper into the application architecture and learn how each component or domain of the insurance claims processing system uses asynchronous events to coordinate communication. You learn why serverless services are well-suited for event-driven architectures. Services such as [AWS Lambda](#), [Amazon EventBridge](#), [AWS Step Functions](#), [Amazon Simple Queue Service \(Amazon SQS\)](#), [Amazon DynamoDB](#), [Amazon API Gateway](#), and [Amazon Simple Storage Service \(Amazon S3\)](#) build a scalable, fault tolerant, and extensible insurance claims processing solution. You can explore the [sample application](#) code repository and deploy the application in your AWS account.

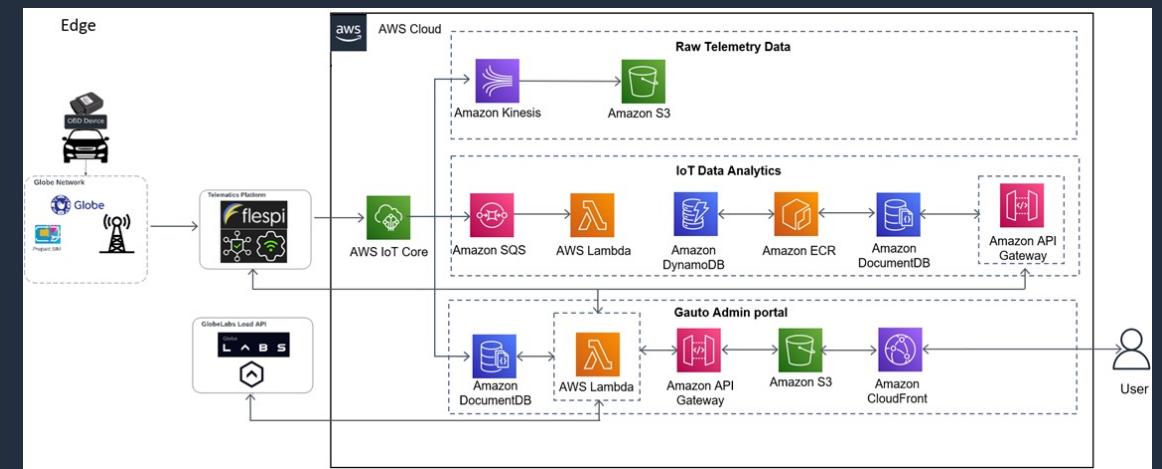
# Not every application needs every pattern

But there are use cases for every pattern

## Example SaaS integration

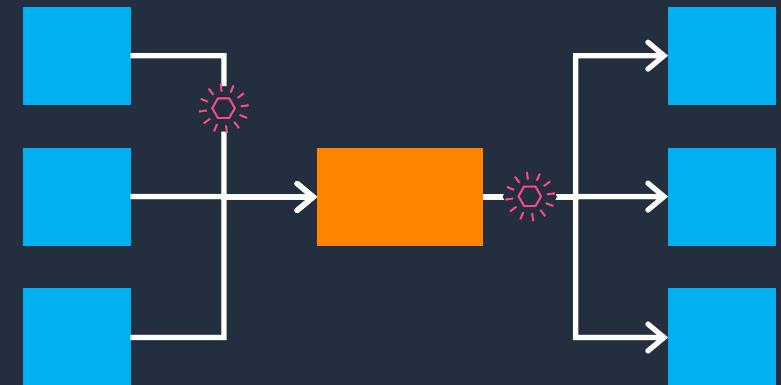


## Example with event streams for usage-based insurance



# Takeaways

- In an event-driven architecture, systems and development teams communicate via events.
- EDAs can give you:
  - Developer agility and extensibility
  - Increased scalability and fault tolerance
  - Lower TCO
- Serverless services are uniquely compatible with EDA
- Combine multiple purpose-built services into EDA patterns



## Learn more

- EDA on Serverless Land
- Serverless Skill Builder
- All patterns and resources linked here:



[github.com/em-shea/eda-patterns-resources](https://github.com/em-shea/eda-patterns-resources)

