AWS re:Invent

NOV. 29 – DEC. 3, 2021 | LAS VEGAS, NV

ENT401

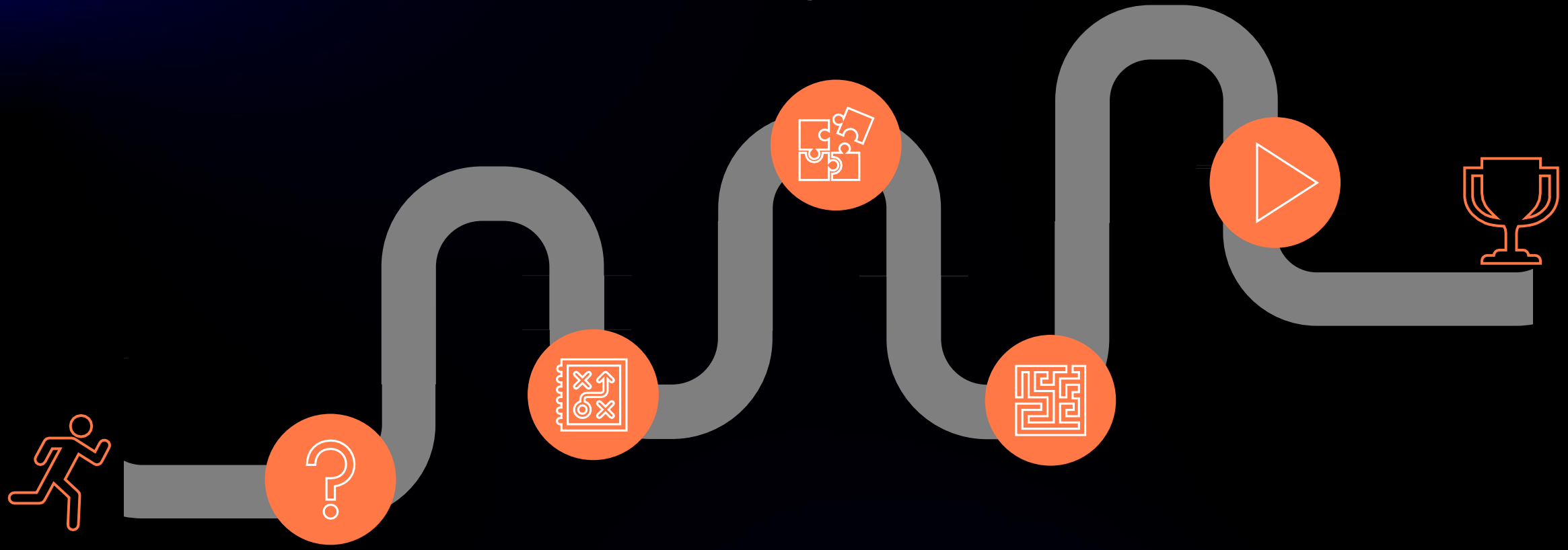
Modernize faster using the Strangler Fig pattern on AWS

Melash Moodley (he/him)
Solution Architect
AWS

Carlos Tagle (he/him)
Head of Business Development,
Control Services
AWS

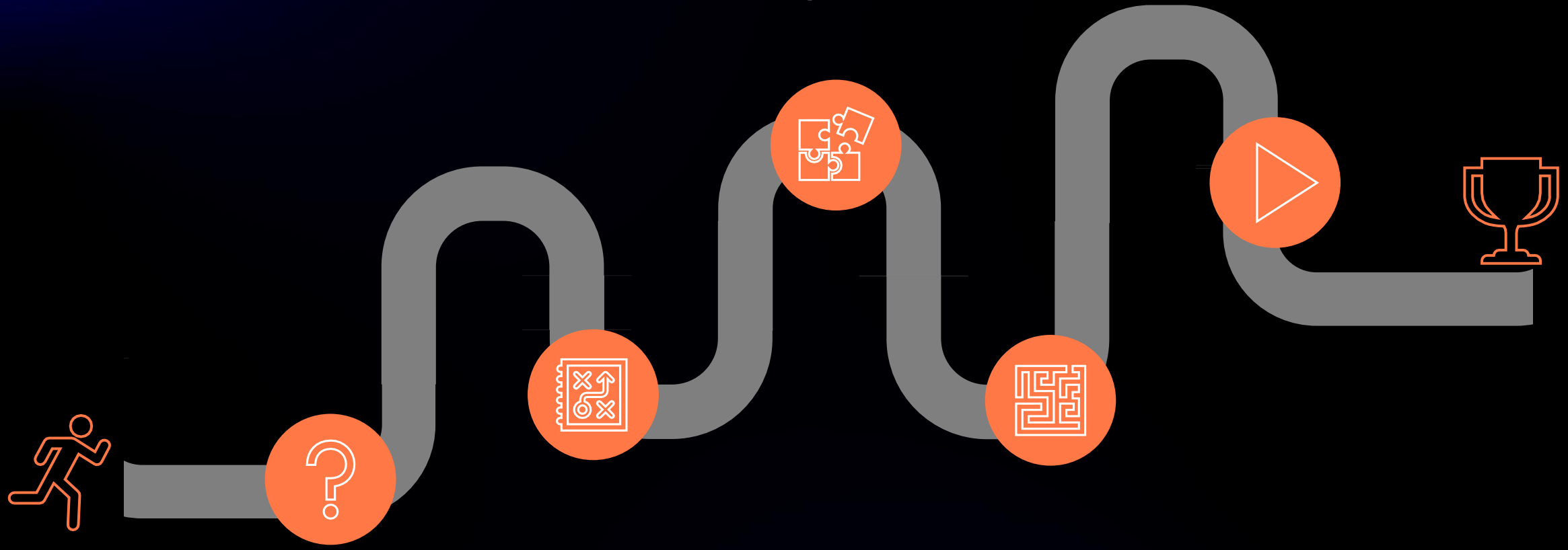
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we will cover today



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we will cover today

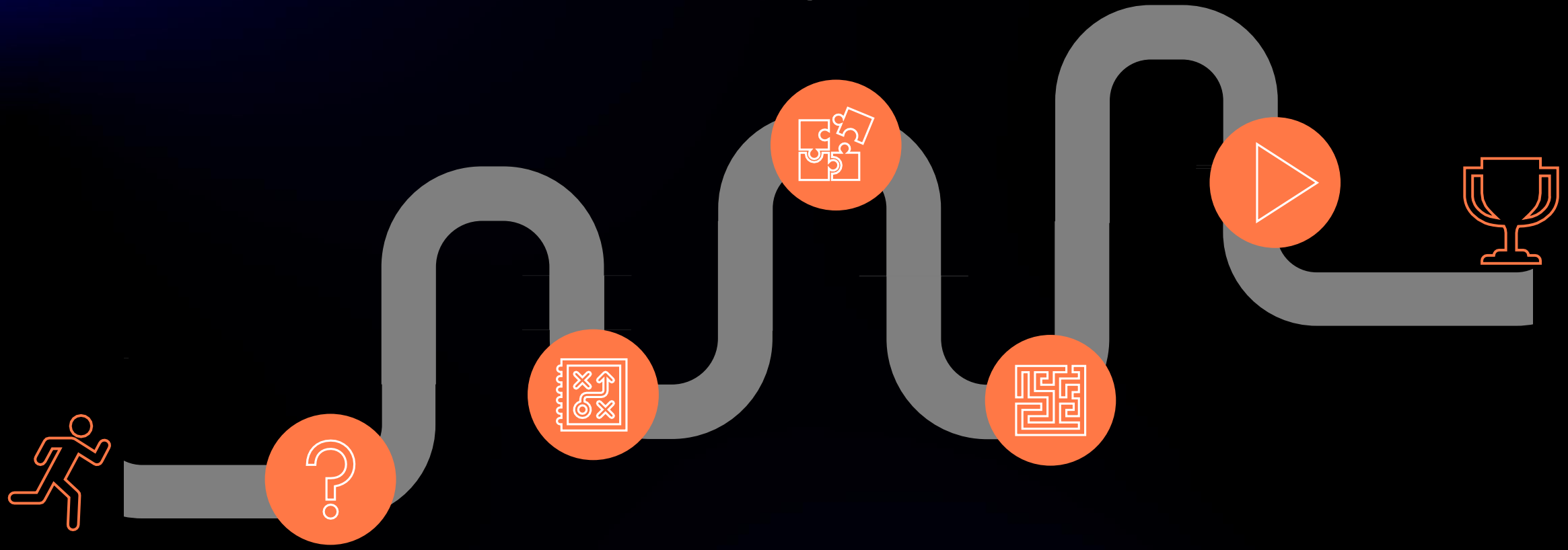


Challenges of refactoring

How to de-risk

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we will cover today



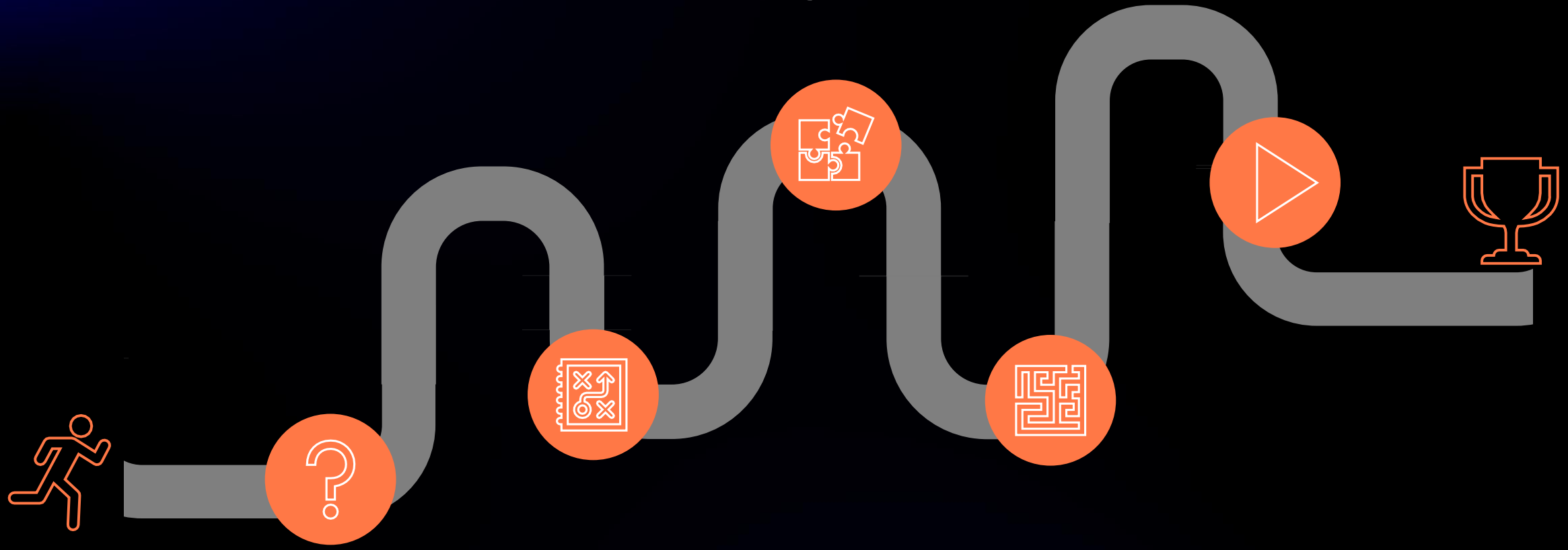
Challenges of refactoring

How to de-risk

How to decompose

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we will cover today



Challenges of refactoring

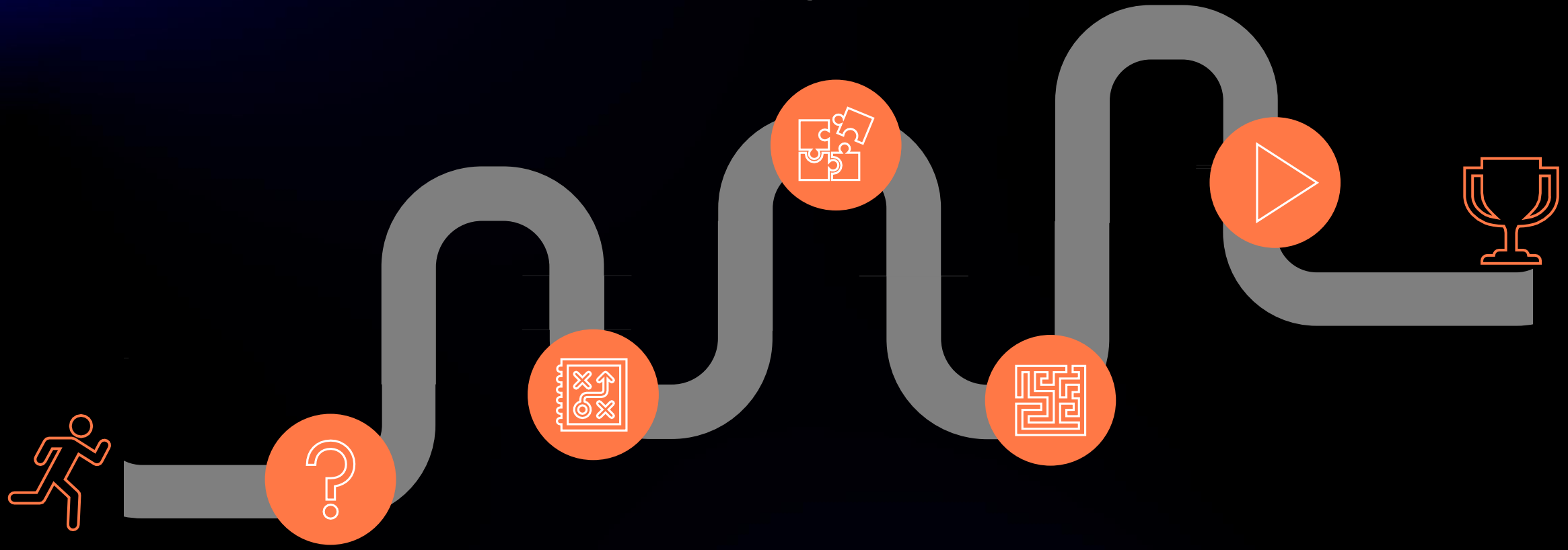
How to de-risk

How to decompose

Managing challenges of multiple accounts

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What we will cover today



Challenges of refactoring

How to de-risk

How to decompose

Managing challenges of multiple accounts

Demo

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Challenge - Refactoring monolithic applications



Business

- Slow product innovation
- Decreased time to market
- Drive unclear team ownership boundaries
- Changes have unforeseen side effects



Technical

- High coupling and low cohesion
- Difficult and infrequent deployments
- Apps are brittle and difficult to understand
- Rapid experimentation has the potential of high scale collateral impact

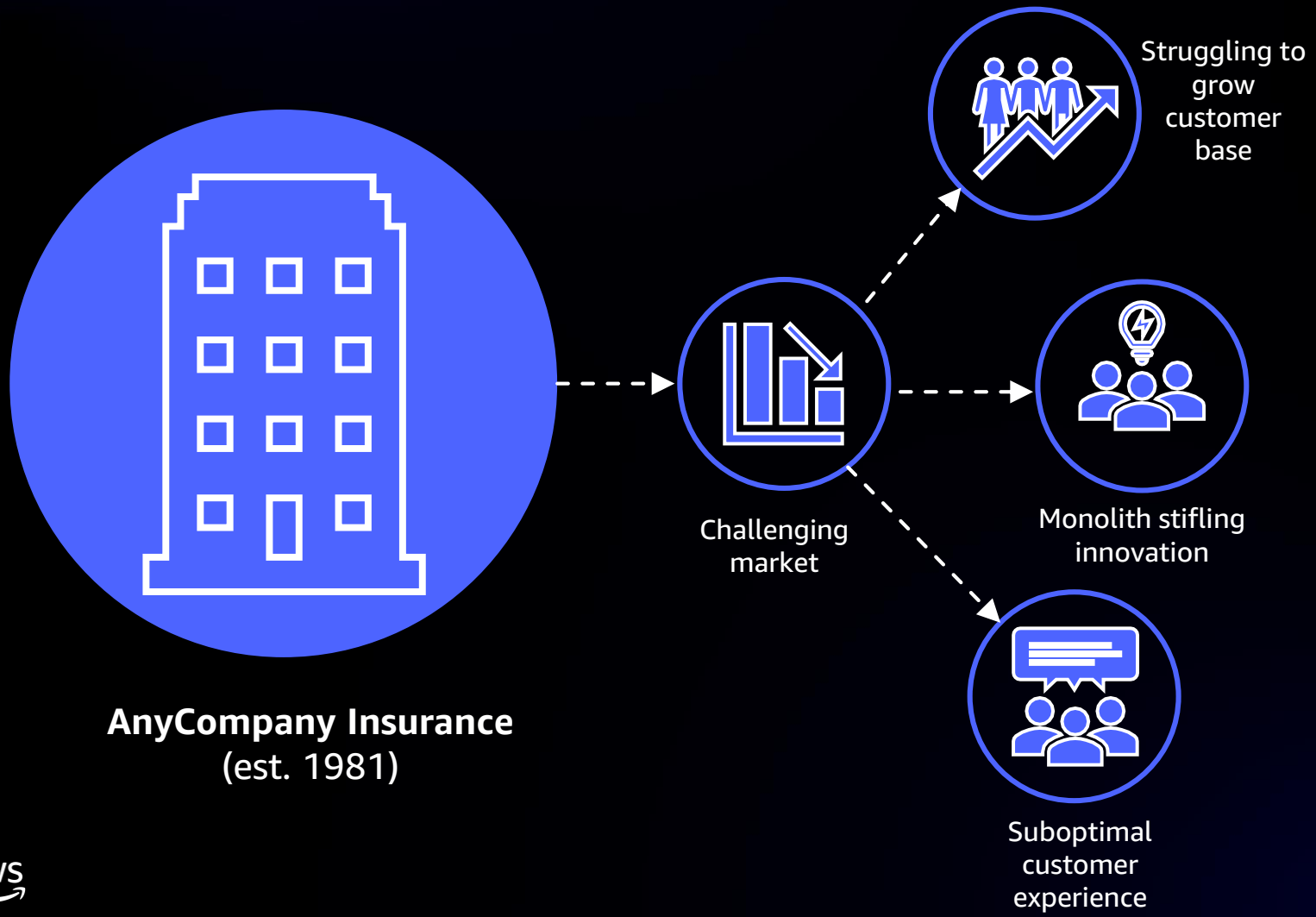
7 Rs

- Refactor
- Re-platform
- Repurchase
- Rehost
- Relocate
- Retain
- Retire

Business Value

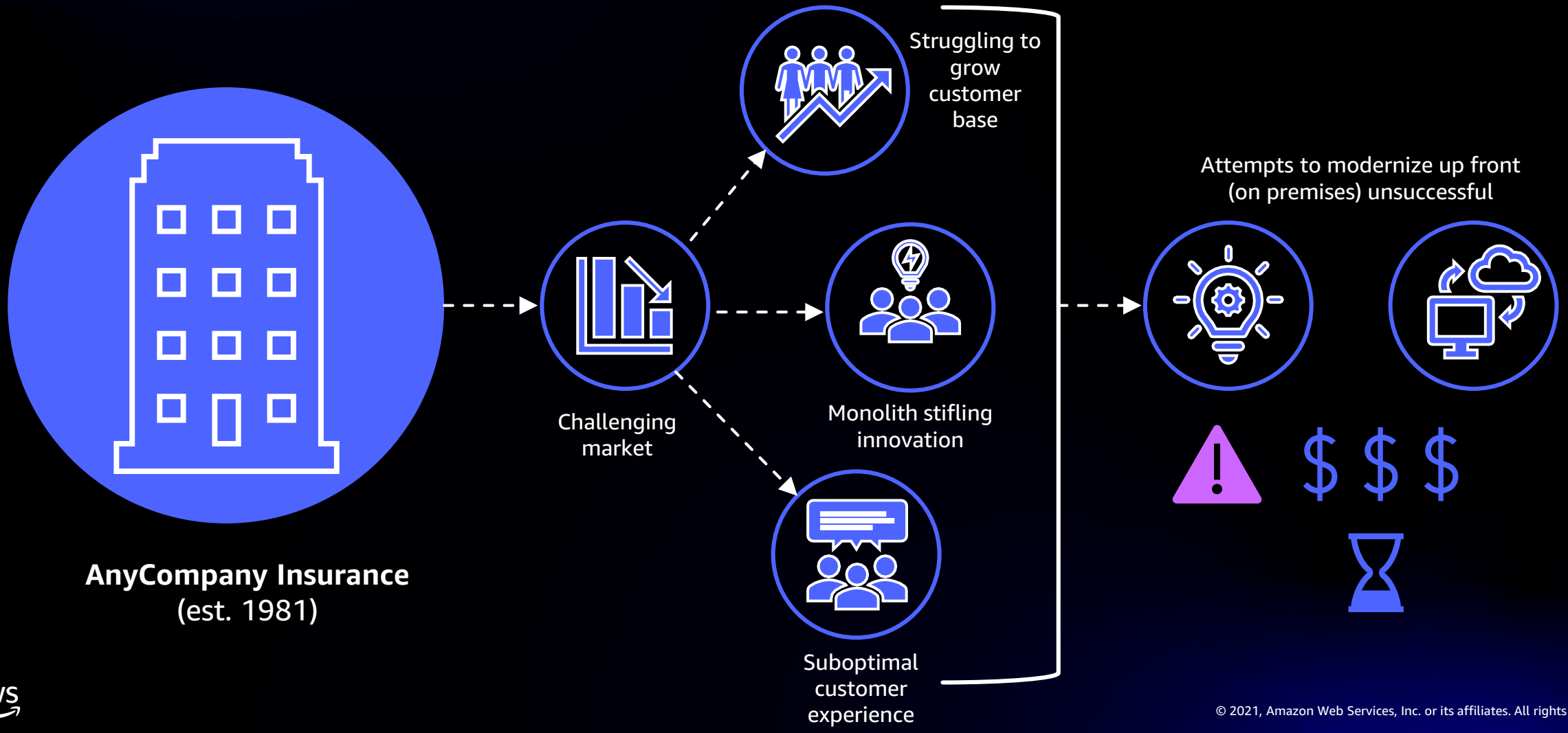
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

A story about AnyCompany's monolith



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

A story about AnyCompany's monolith



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AnyCompany's 2nd attempt to modernize



Phase one: lift and shift

Quickly take advantage of AWS

- Cost savings
- Productivity
- Resilience
- Agility

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AnyCompany's 2nd attempt to modernize



Phase one: lift and shift

Quickly take advantage of AWS

- Cost savings
- Productivity
- Resilience
- Agility

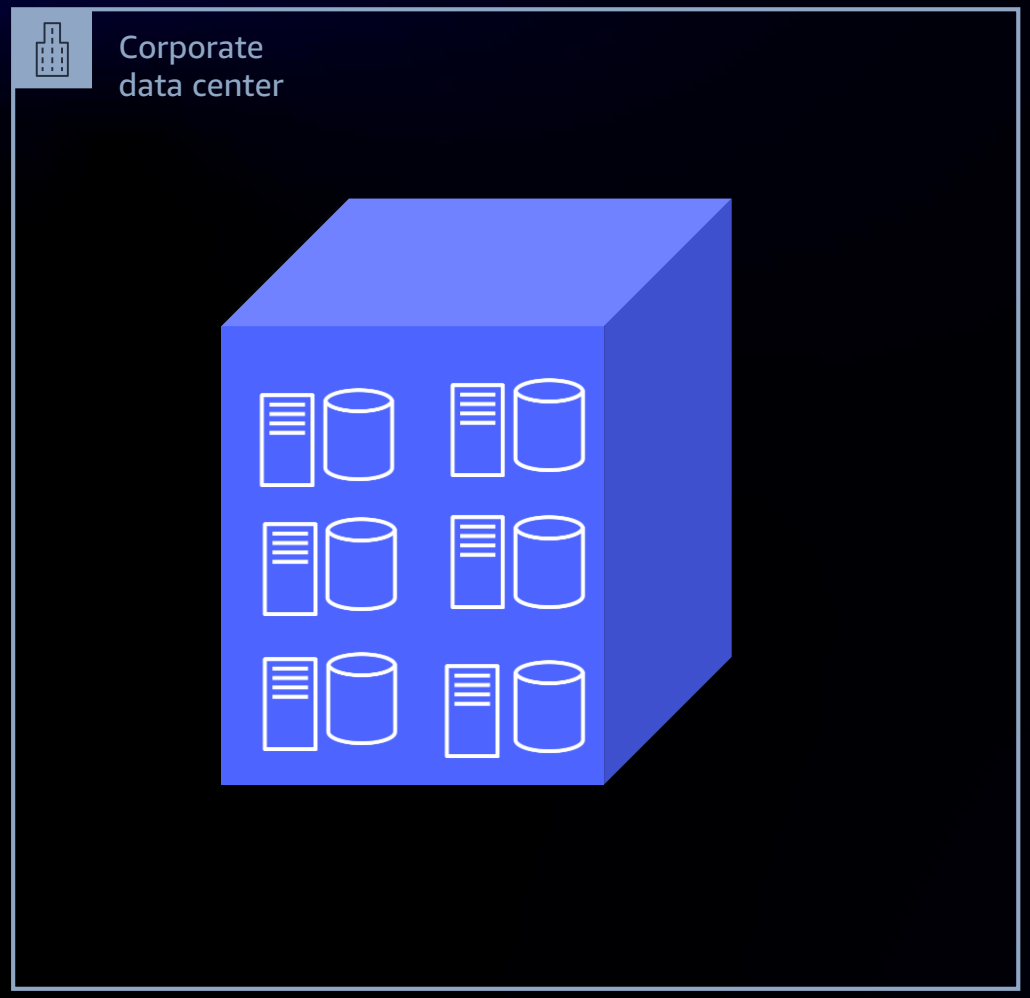


Phase two: modernize

After applications are on AWS, it's easier to refactor

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

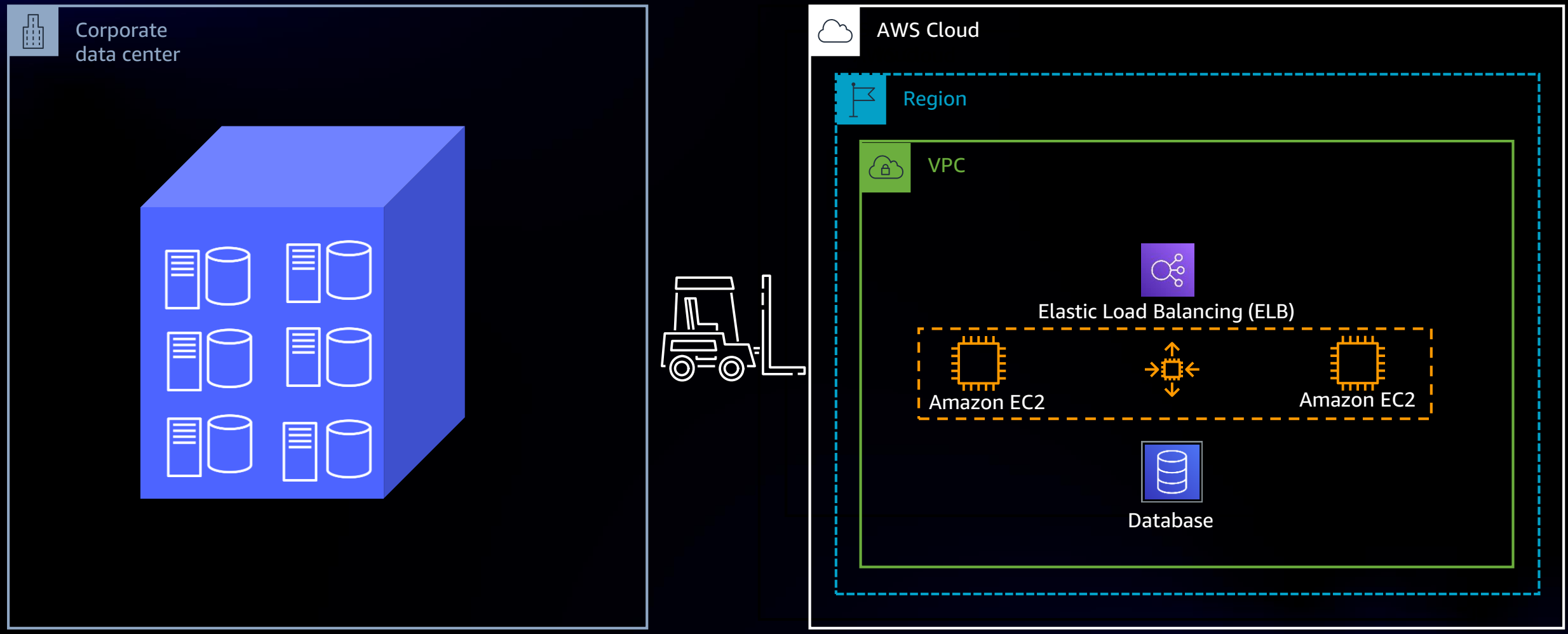
Phase one: Lift and shift



Monolith

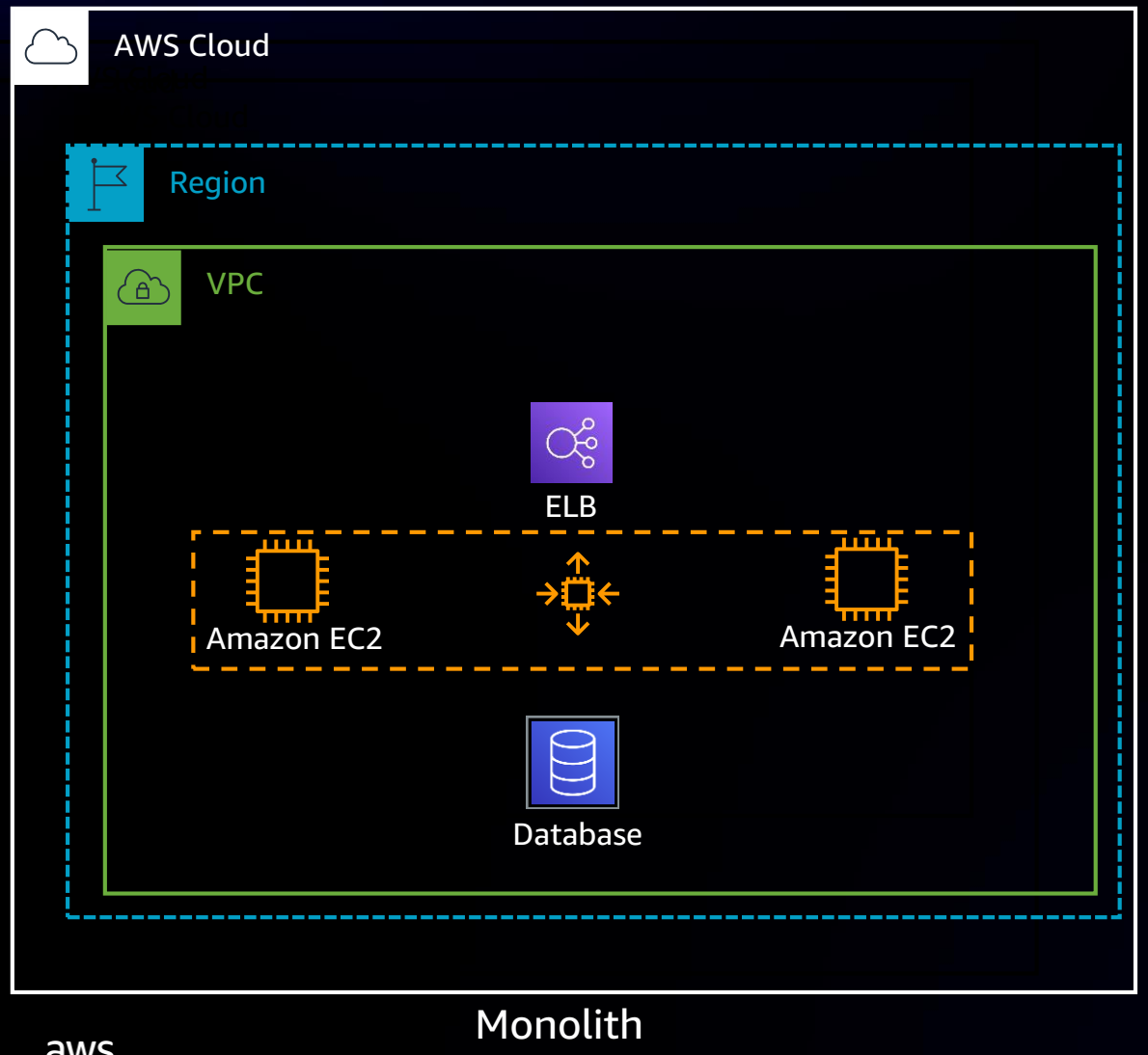
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Phase one: Lift and shift



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

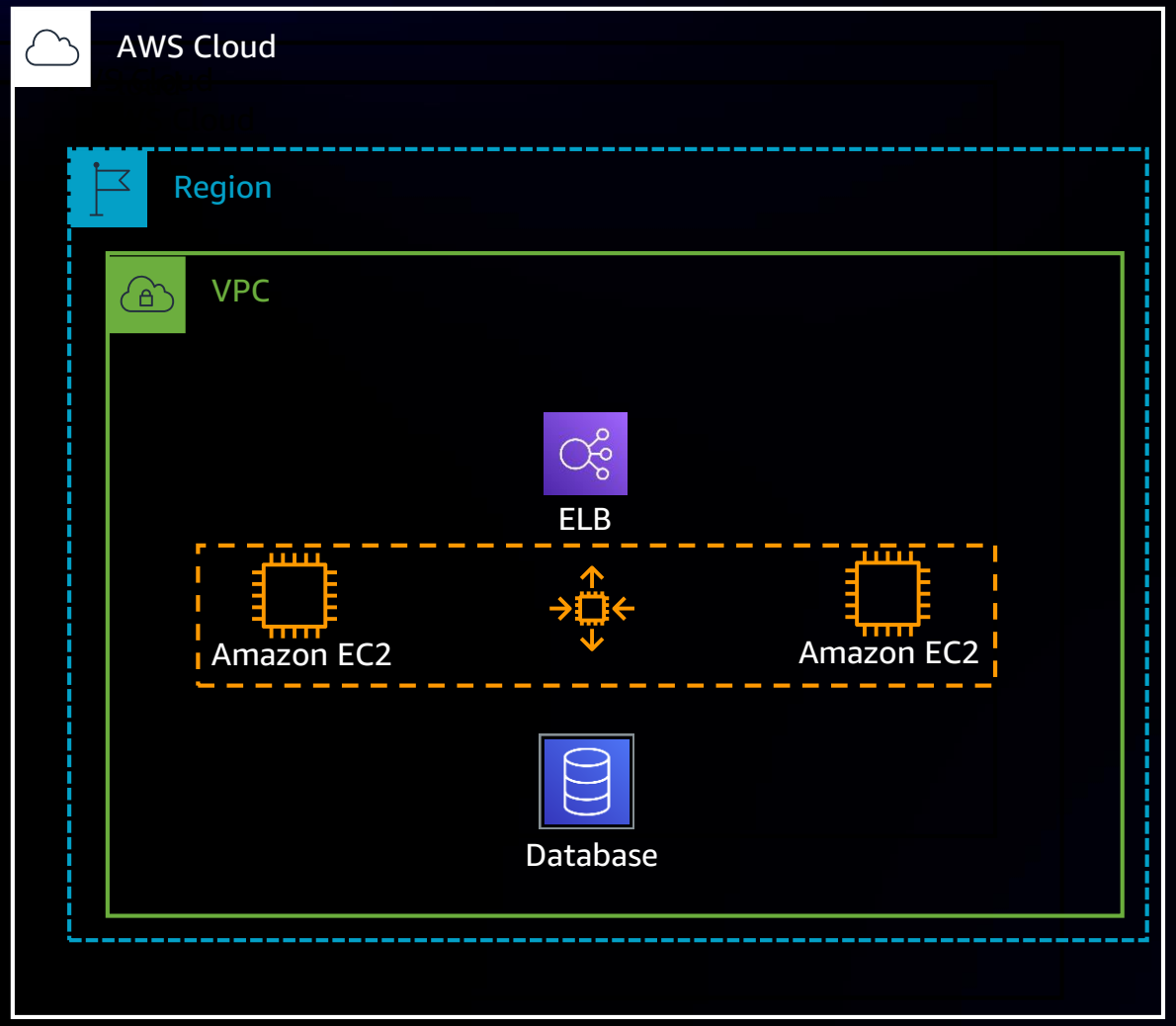
Phase two: Modernize



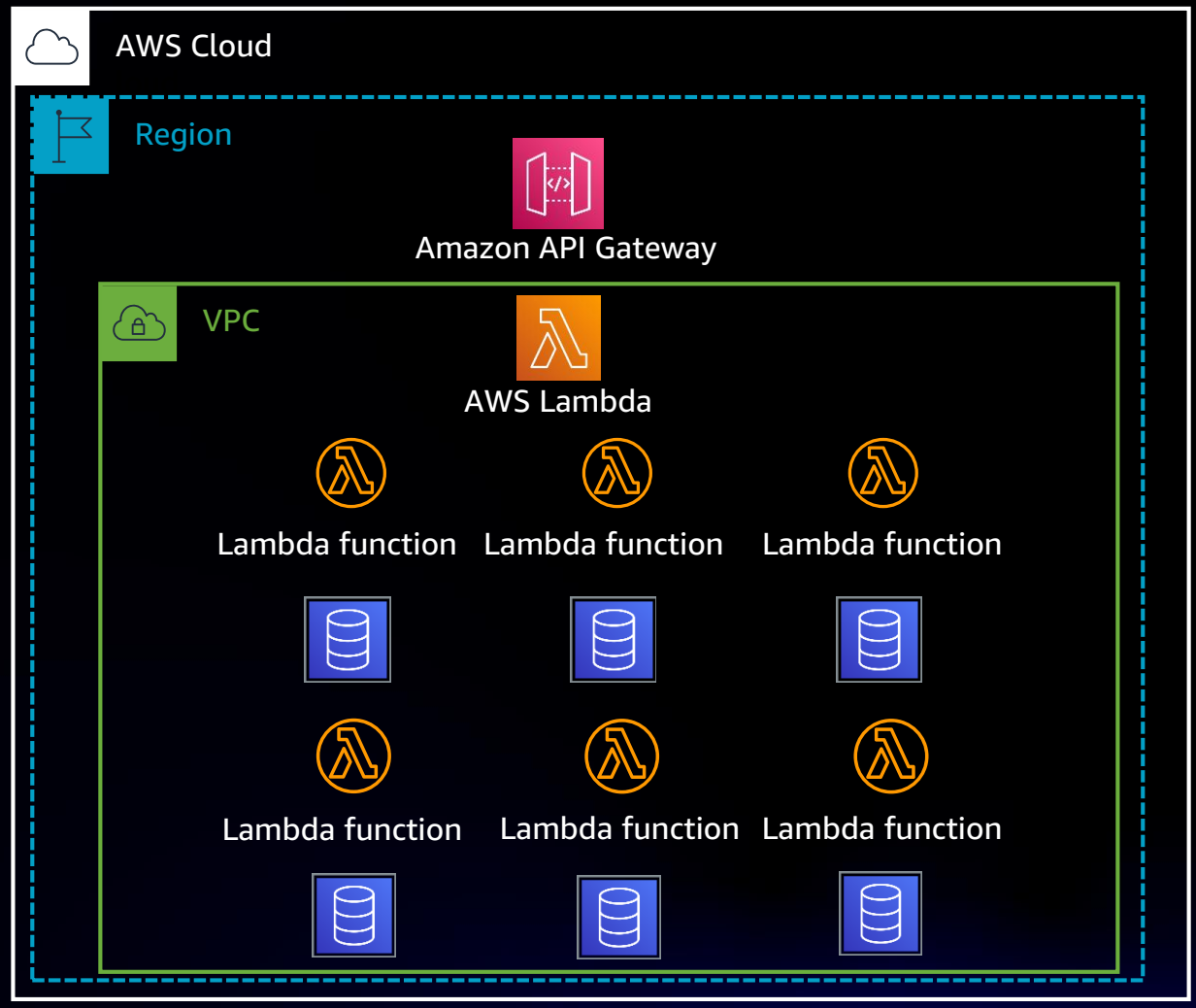
Monolith

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Phase two: Modernize



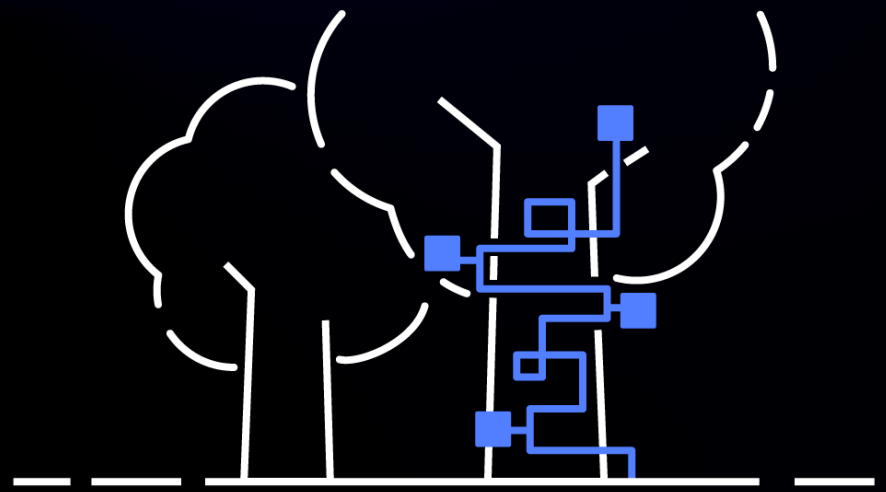
Monolith



Refactor

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Modernize with the Strangler Fig pattern



"Gradually create a new system around the edges of the old, letting it grow slowly over several years until the old system is strangled."

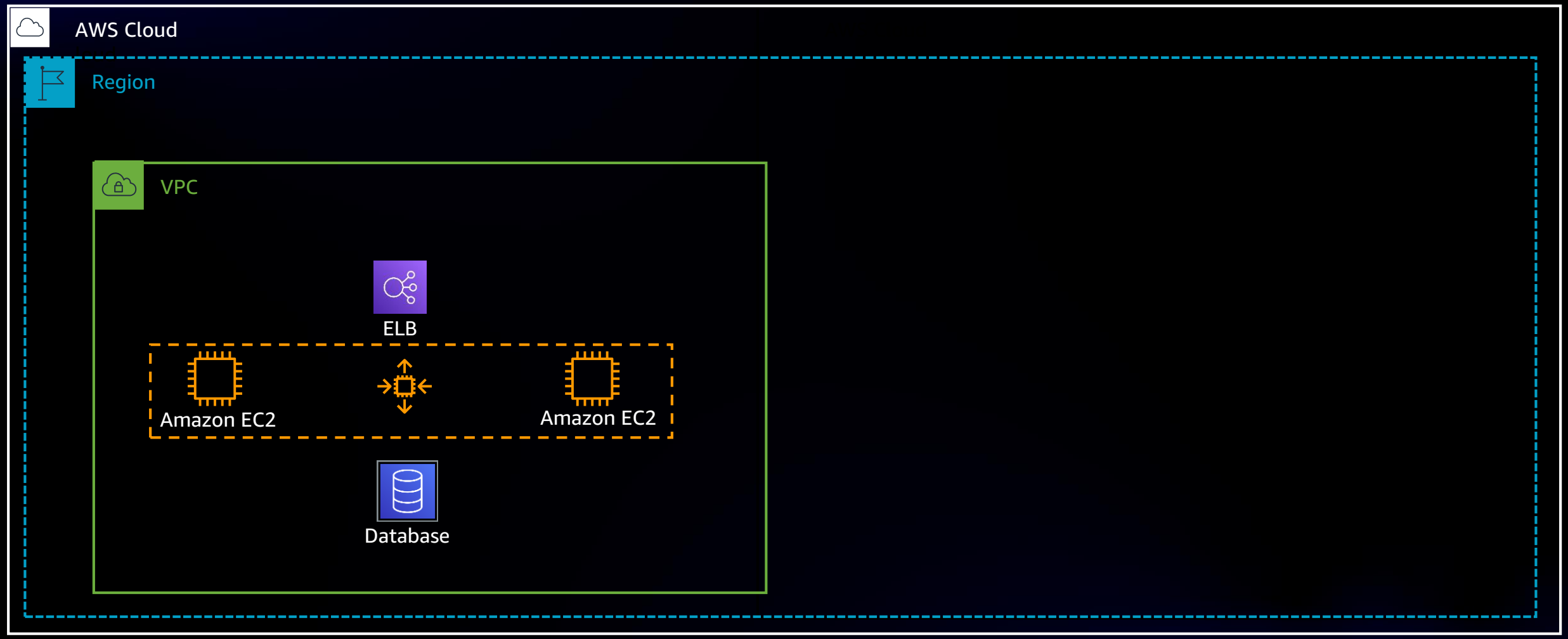
Martin Fowler, Thoughtworks

Benefits

- Agility
- Frequent releases
- Lowers risk
- Immediate benefits
- Easier rollback

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

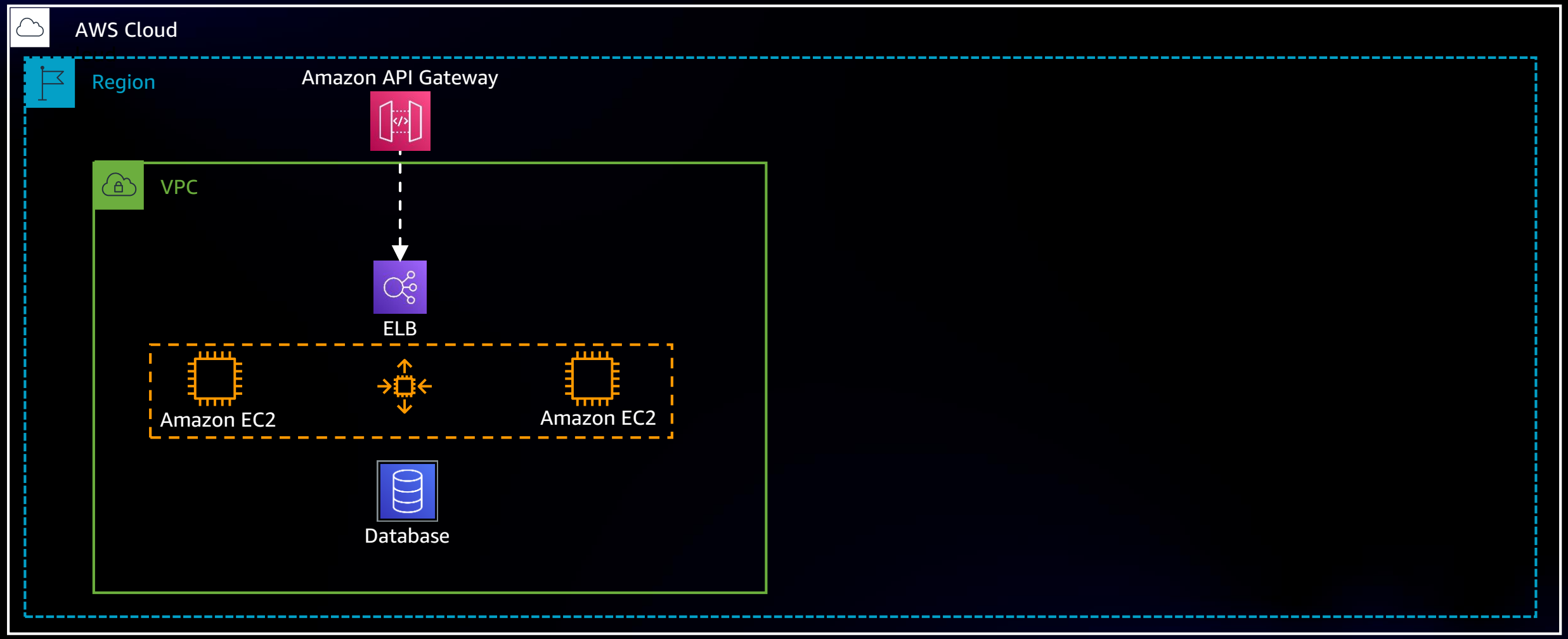
Strangler Fig pattern with API Gateway



A single application

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

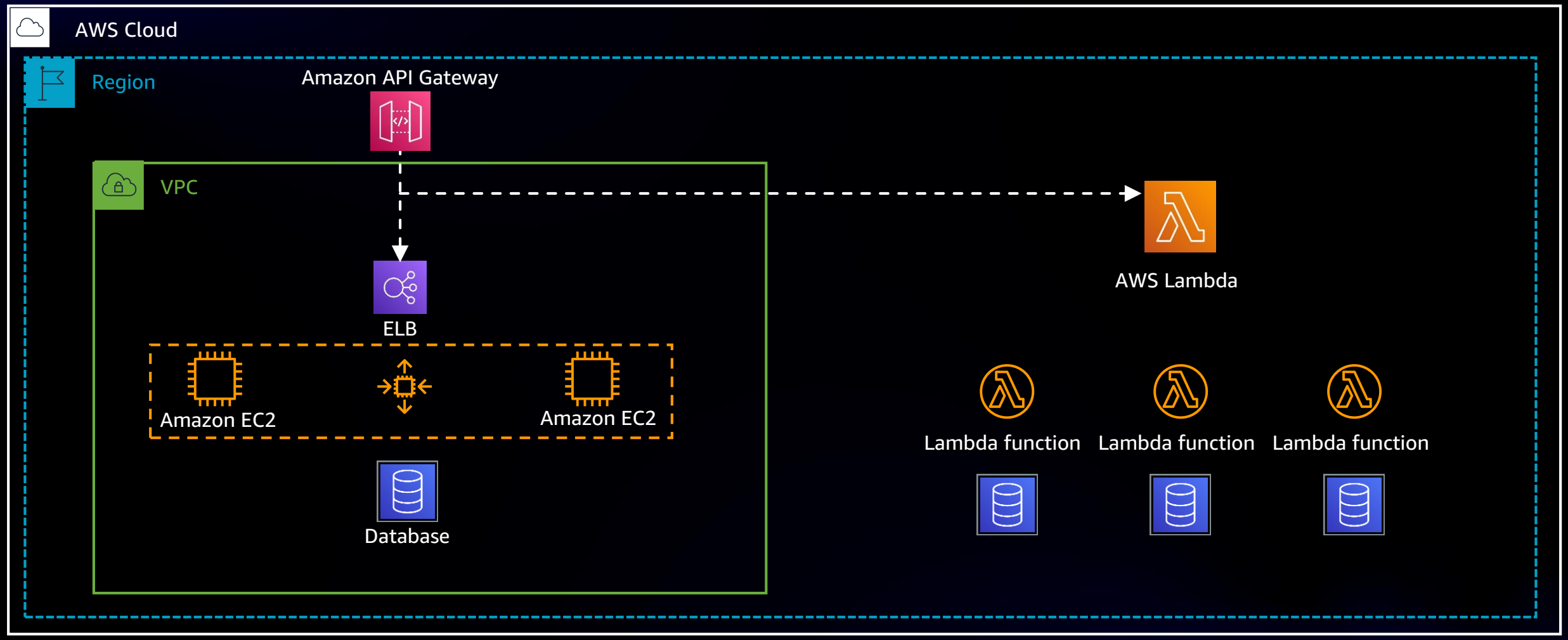
Strangler Fig pattern with API Gateway



A single application

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Strangler Fig pattern with API Gateway



A single application

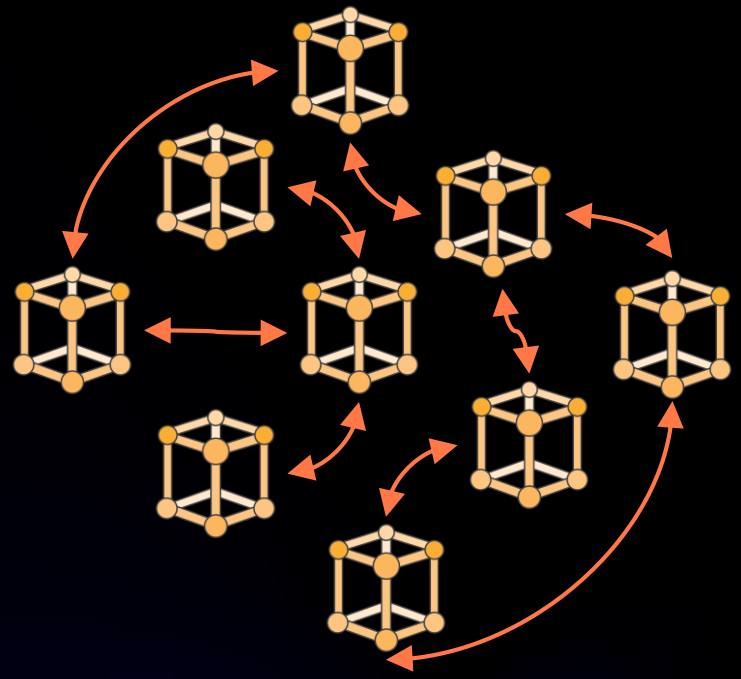
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Diving into decomposition patterns

Event storming as a way of understanding a business domain

Main types of decomposition patterns

- Auxiliary service
- Business capability
- Business transaction
- Service per team
- Business subdomain



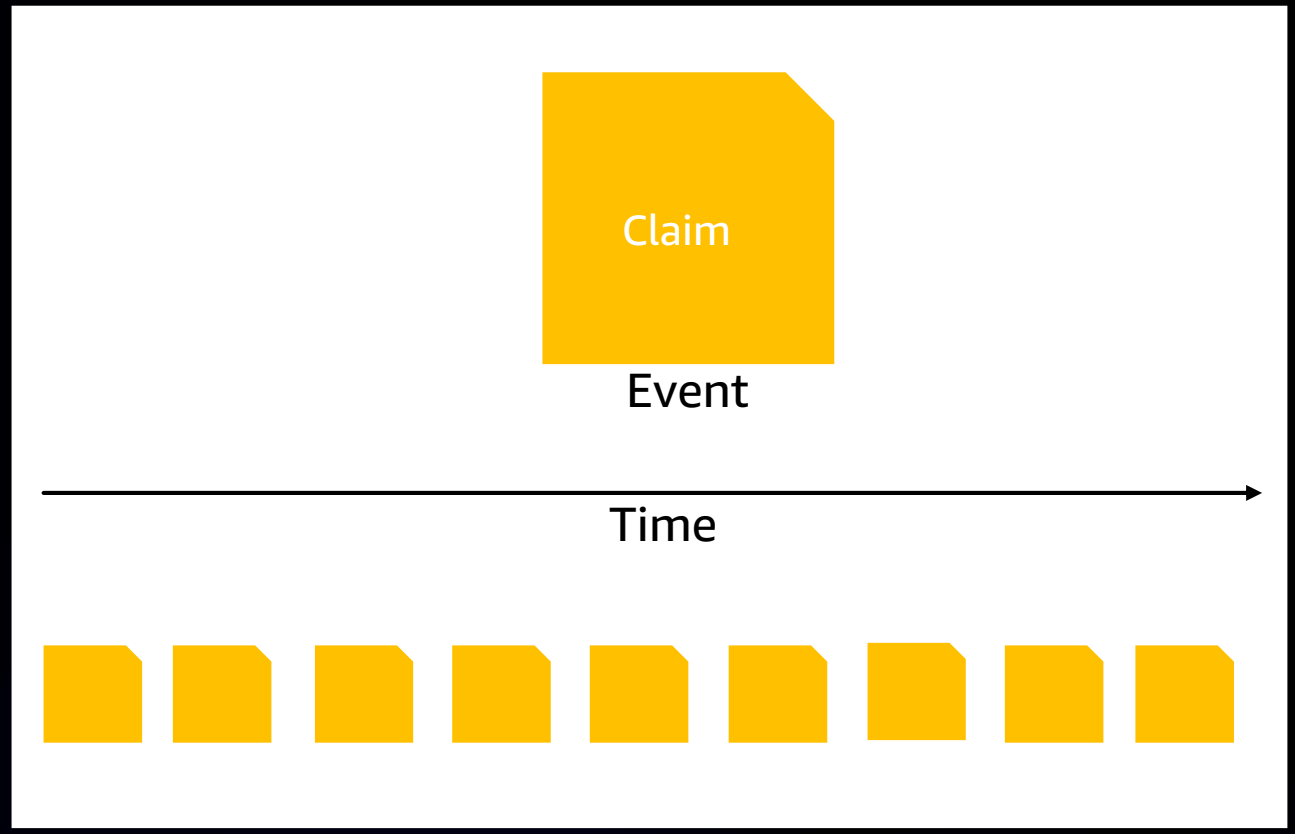
https://docs.aws.amazon.com/prescriptive-guidance/latest/modernization-decomposing-monoliths/welcome.html

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Event storming: Overview

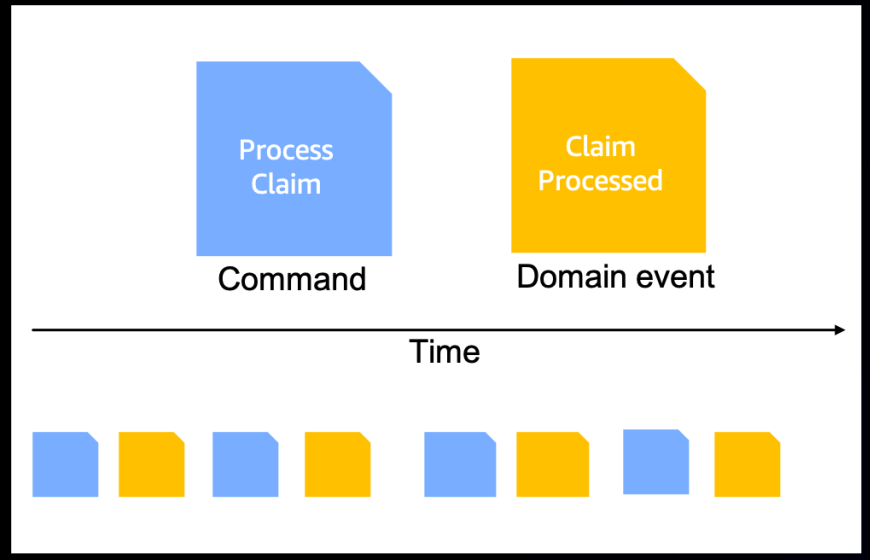
- A simple technique to map out complexity of the business process
- Brings together key stakeholders across key functions in usually a day-long workshop
- Also useful when target architecture is defined but not business rules



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Event storming: The process

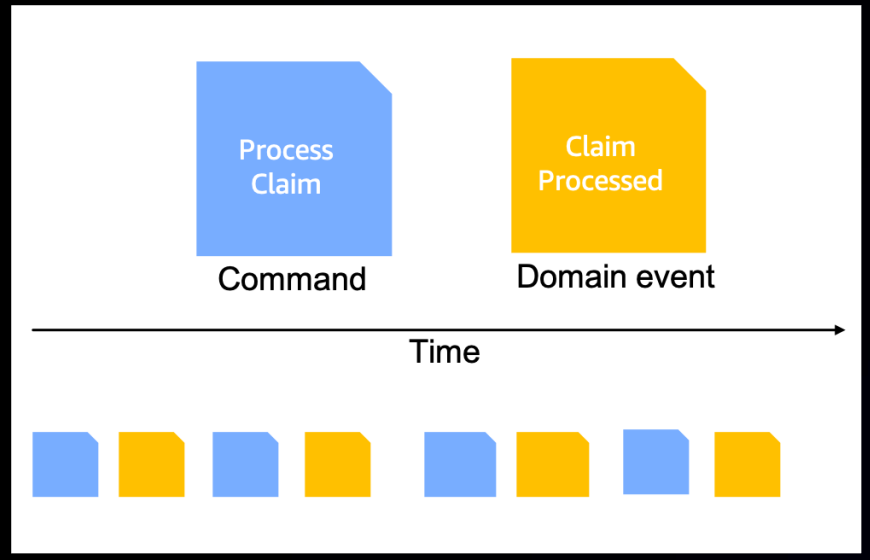
1. Work through domain events in a process
2. Discover what triggers these domain events and note these as commands next to the domain events

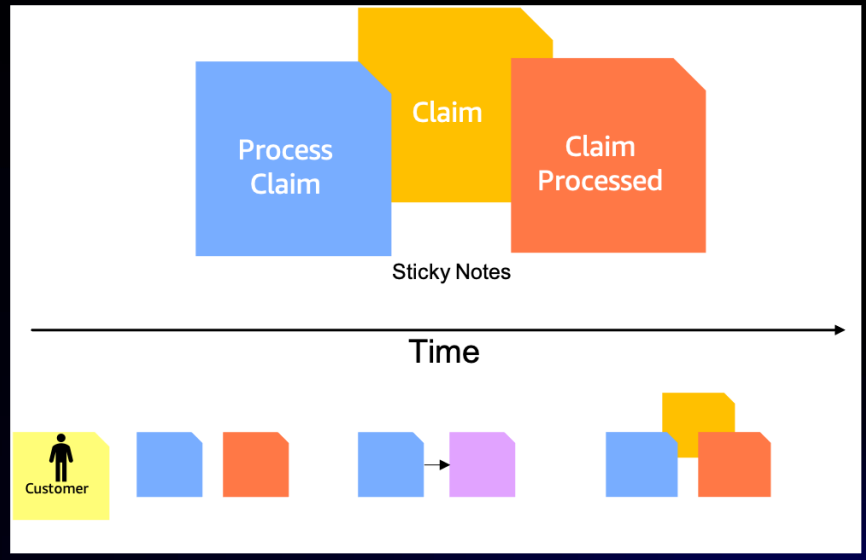


© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Event storming: The process

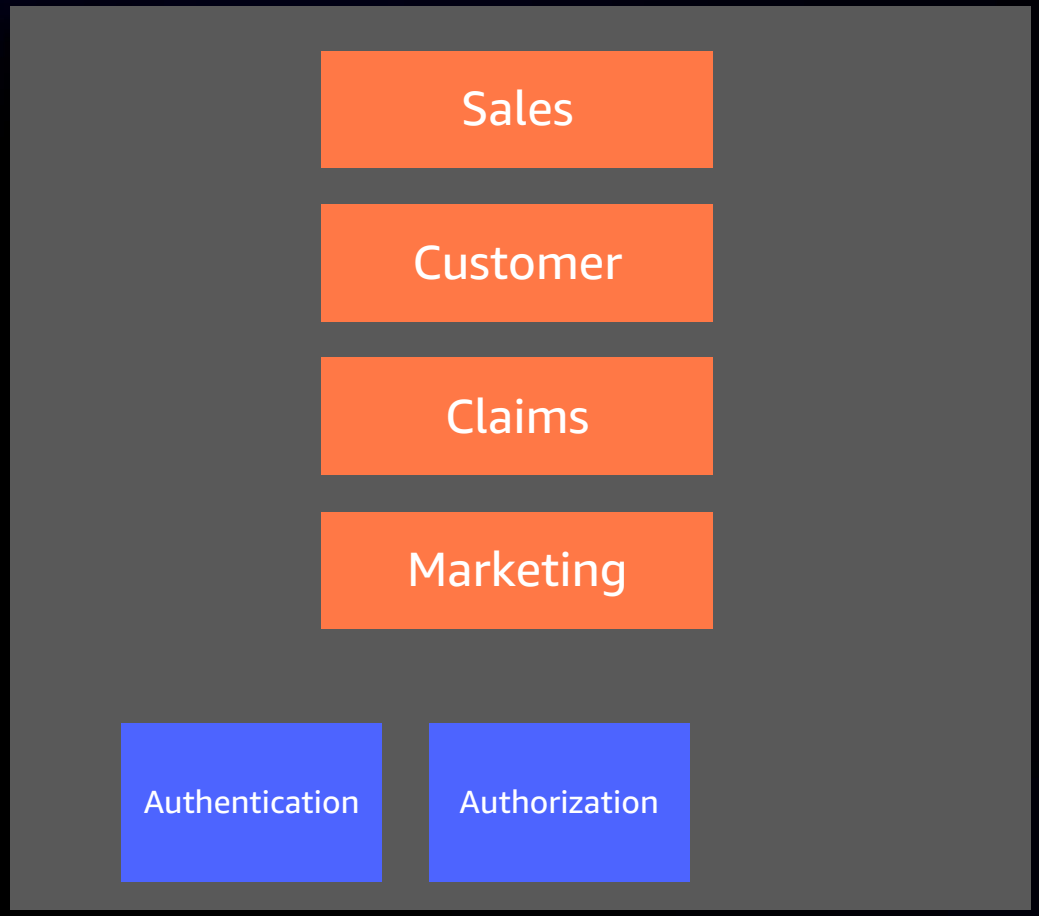
1. Work through domain events in a process
2. Discover what triggers these domain events and note these as commands next to the domain events
3. As you continue, you will discover the entity responsible for emitting the domain event
4. These entities will enforce the business rules applicable in the particular use cases
5. Use another sticky note to indicate what entity will handle a command and eventually emit one or more domain events





© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

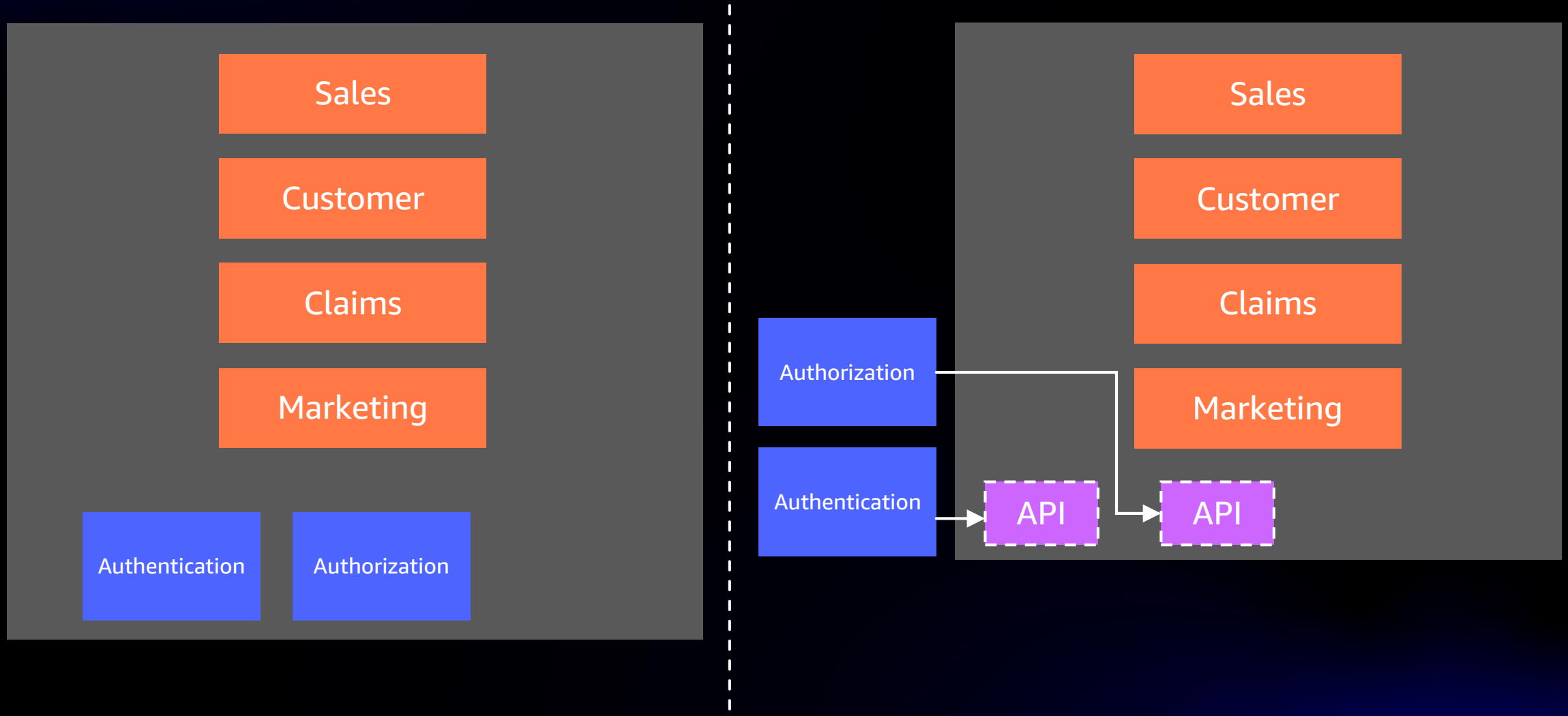
Decompose by auxiliary service



- A good starting point for decomposing
- Requires in-depth understanding of auxiliary and infrastructure services

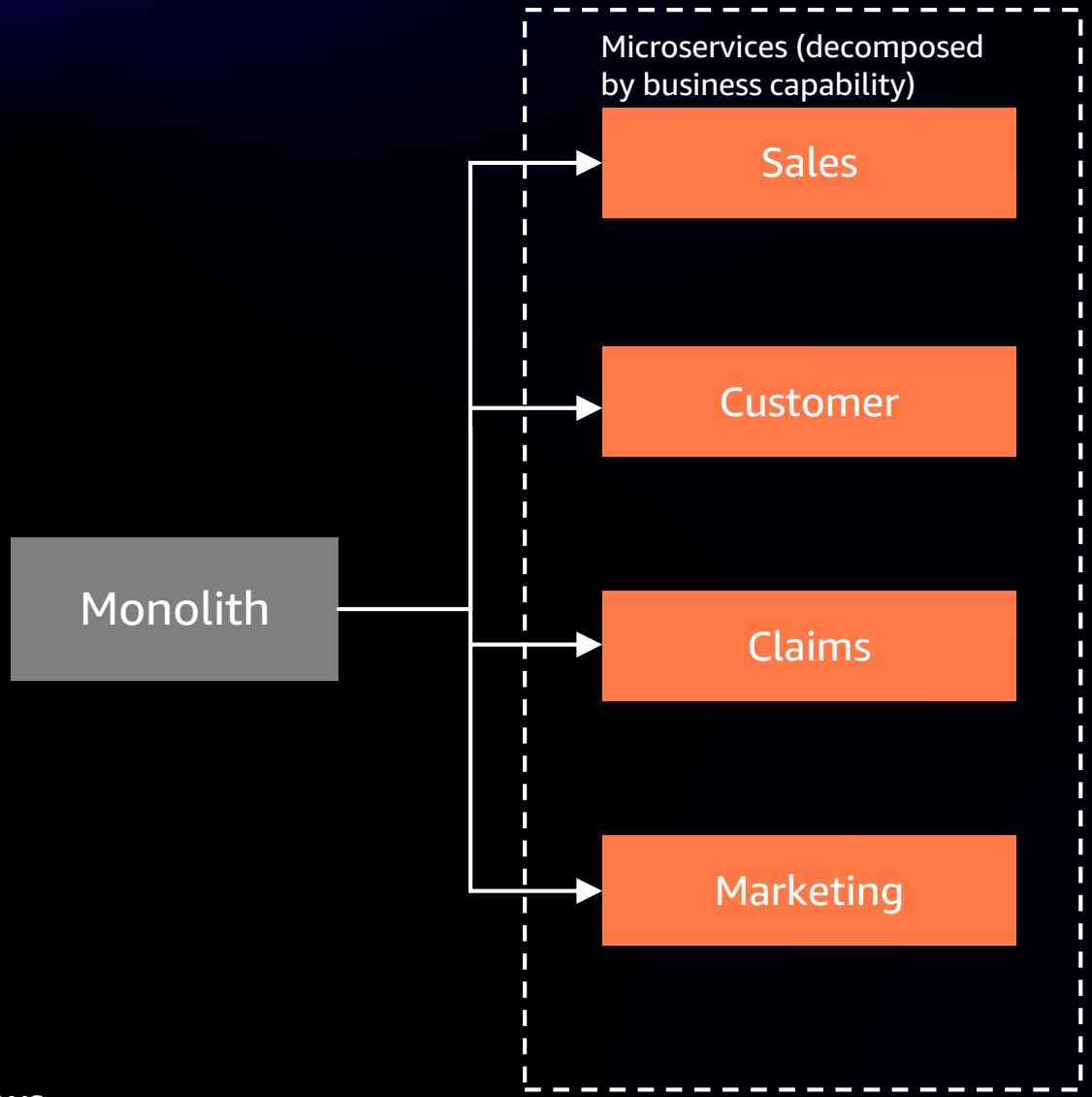
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Decompose by auxiliary service



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Decompose by business capability

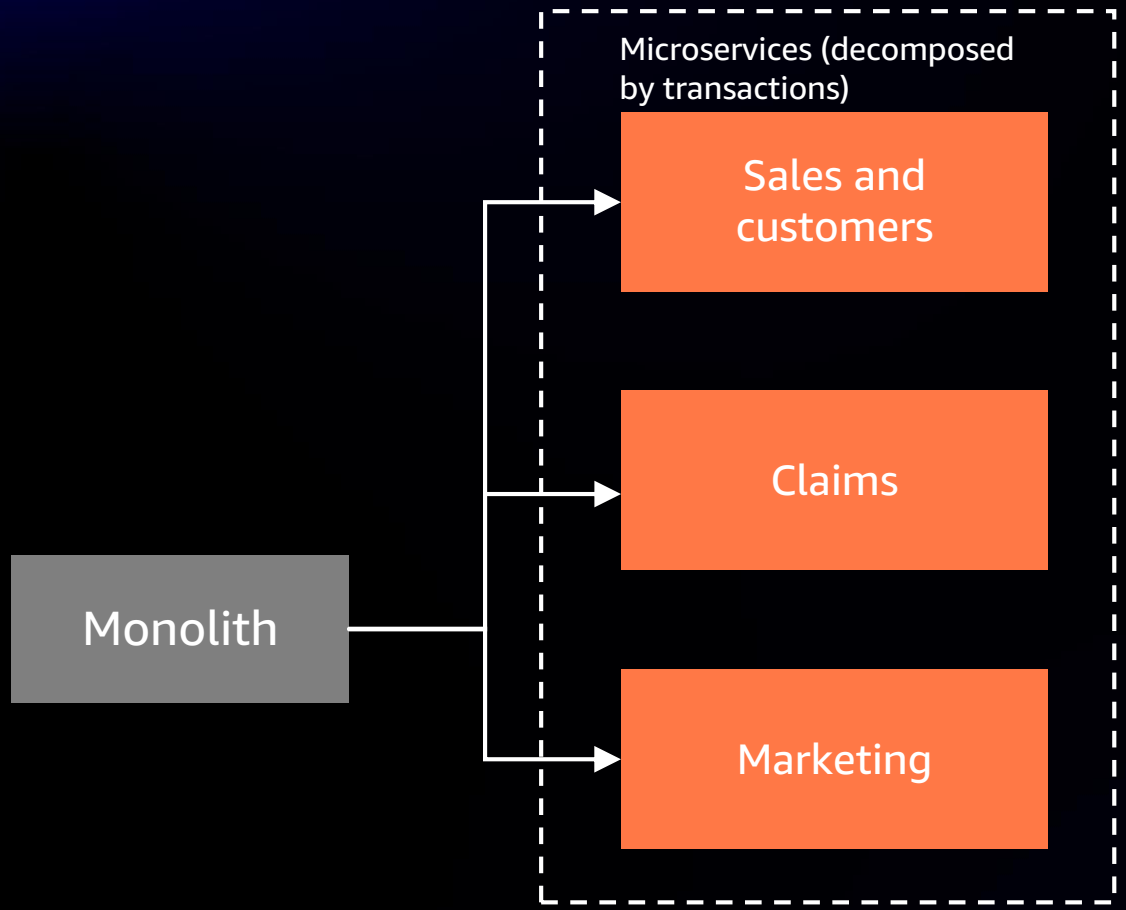


- Loosely coupled services
- Teams organized around delivering business value

- Requires in-depth understanding of the business

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Decompose by business transaction

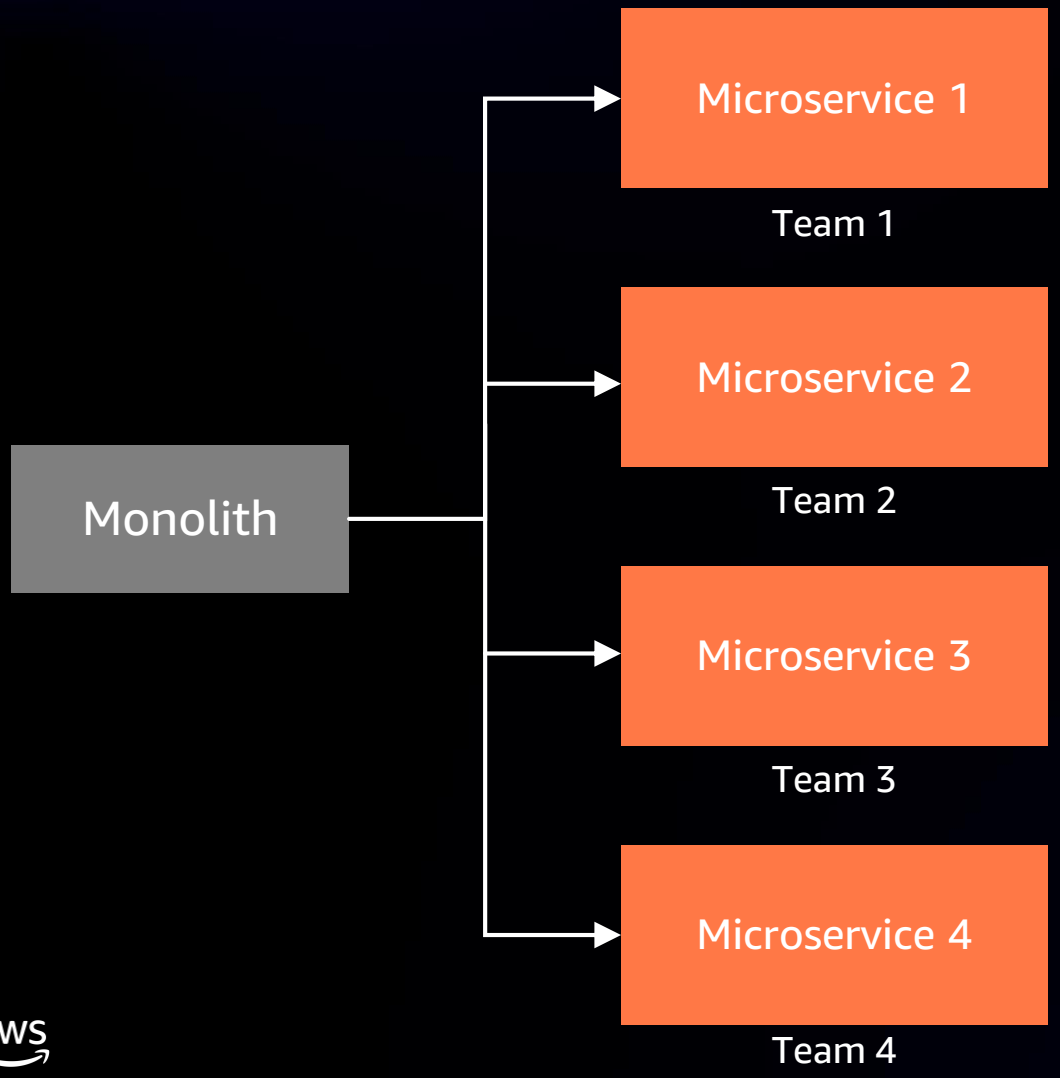


- Faster response times
- No need to worry about data consistency
- Improved availability

- Can result in a distributed monolith

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Service per team

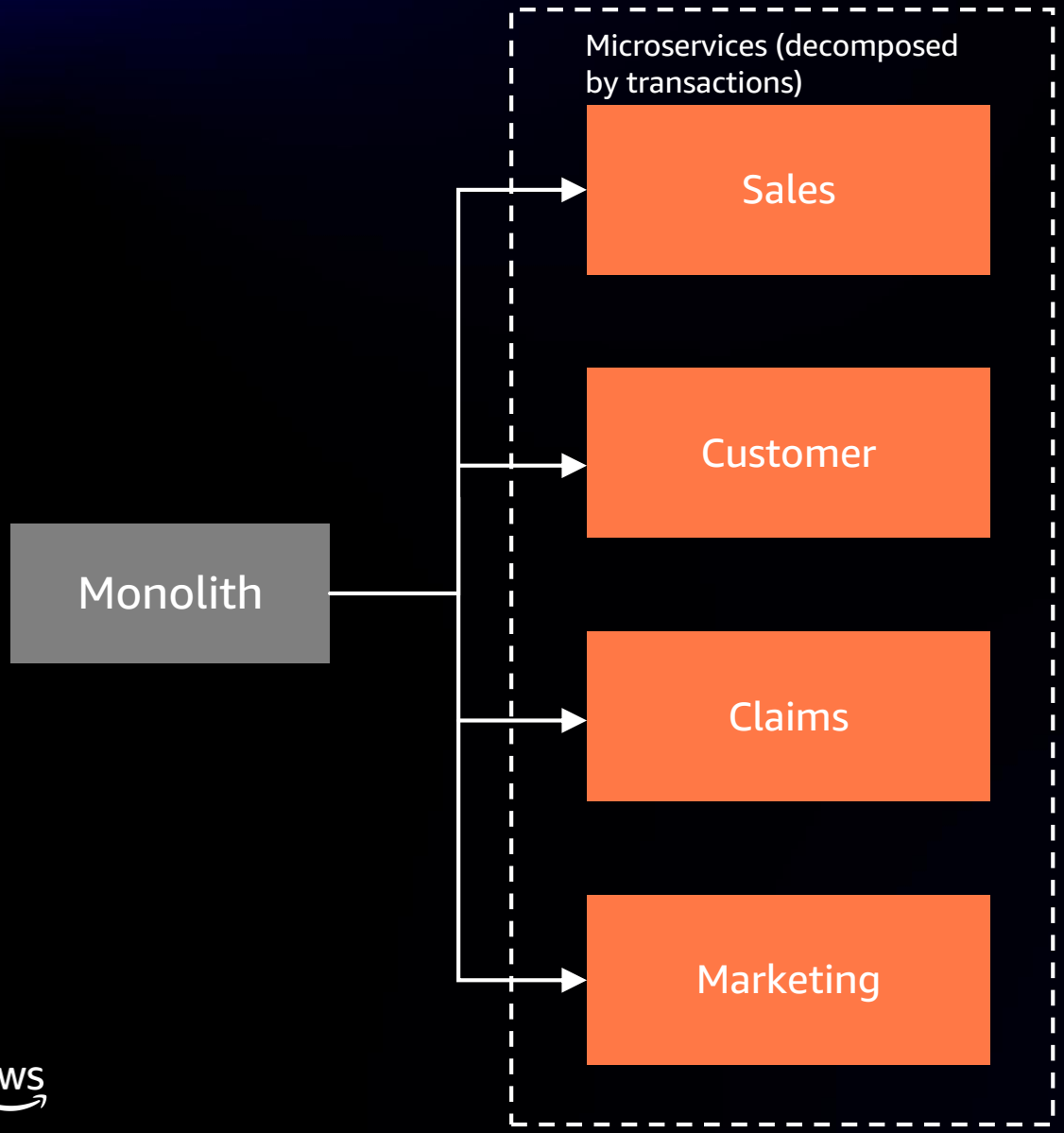


- Different teams can use different technologies, frameworks, and programming languages

- Harder to align teams to business capability
- Could result in bottlenecks if a team is aligned to multiple services

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Decompose by business subdomain

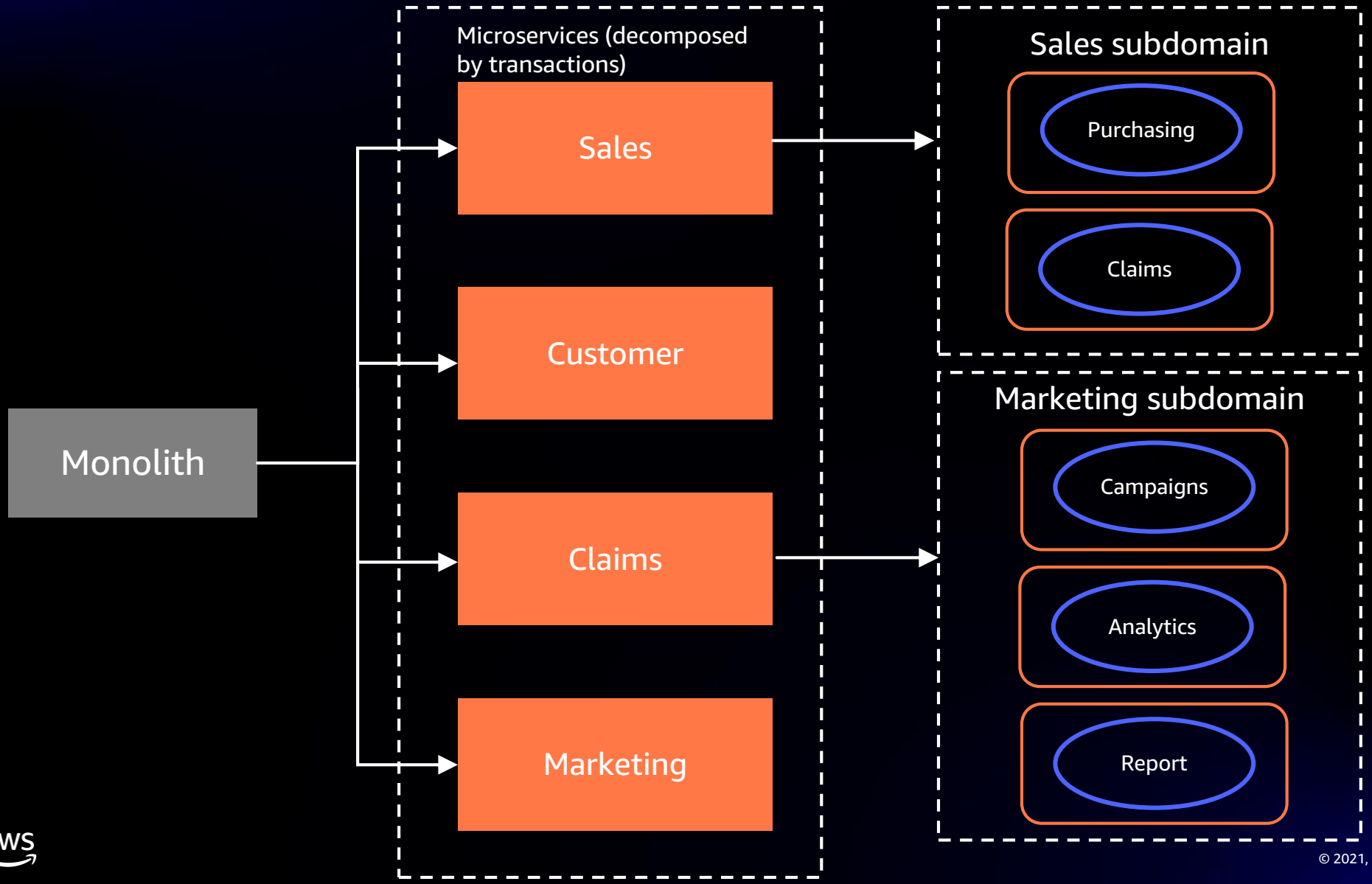


- Loosely coupled architecture
- System is scalable and predictable
- Technology and team independence
- Modeled around the business

- Requires in-depth understanding of the overall business

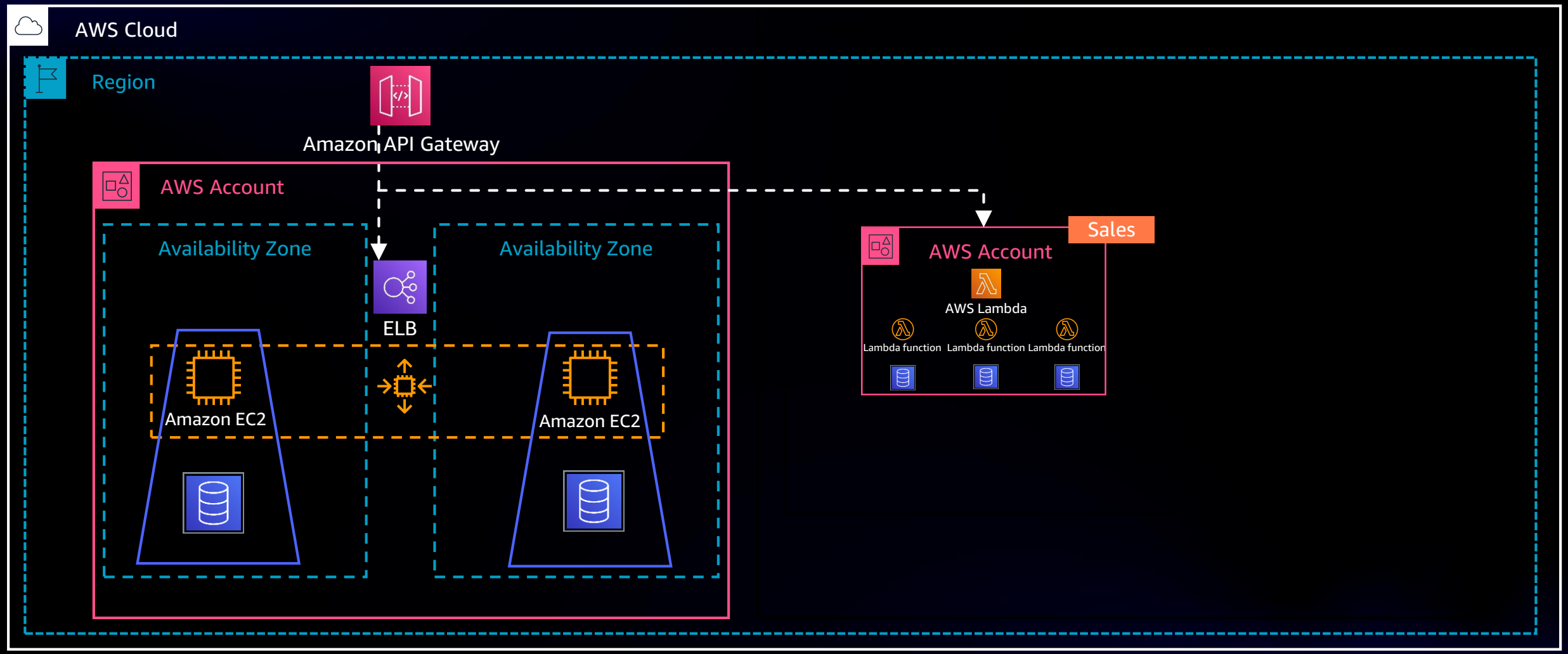
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Decompose by business subdomain



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

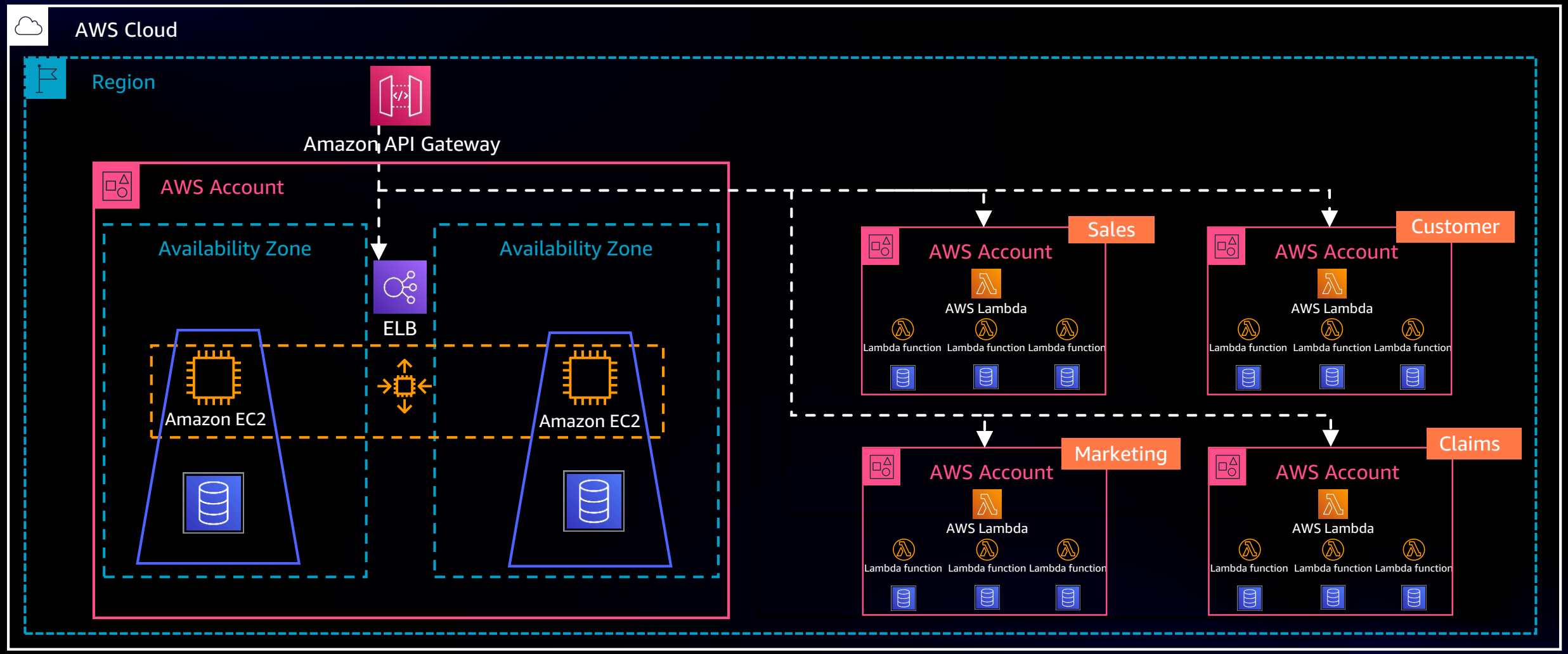
Complexity of multiple accounts



Single application

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

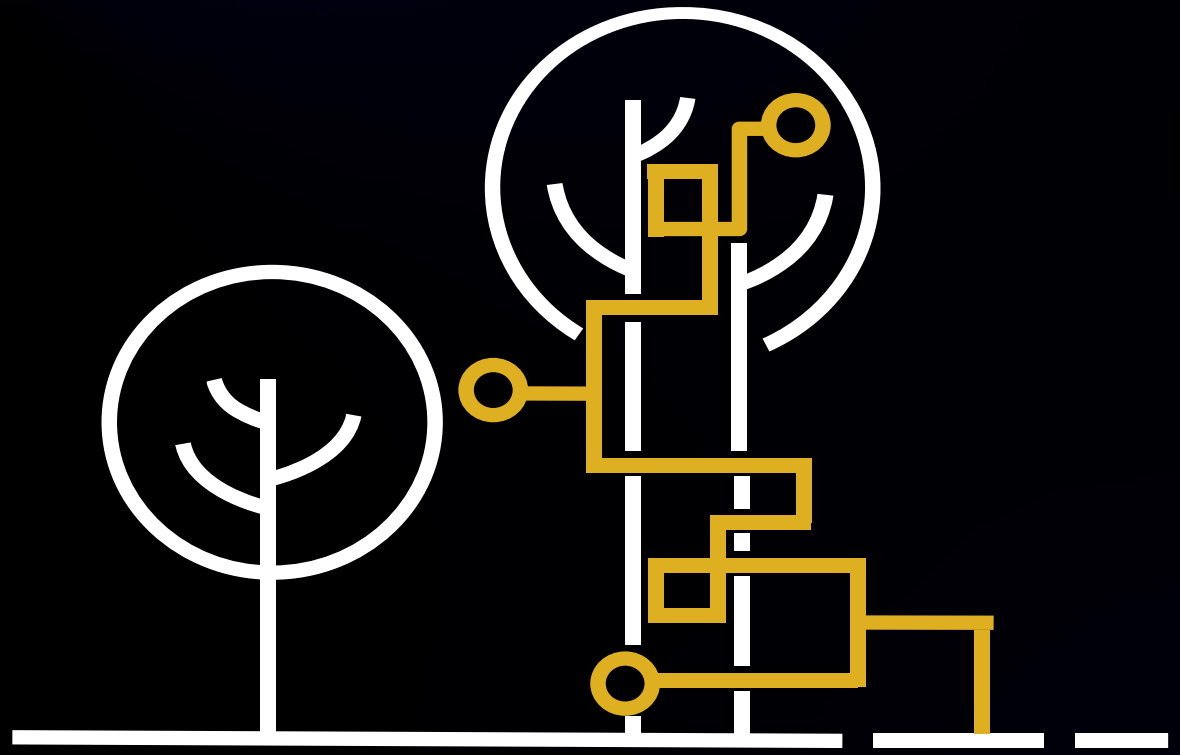
Complexity of multiple accounts



Single application

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Migration Hub Refactor Spaces
Transform your applications faster



Start refactoring applications in days instead of months

Combine your legacy app and microservices into a single application

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Introducing AWS Migration Hub Refactor Spaces



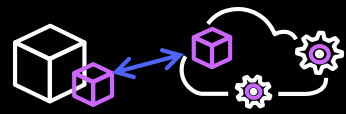
AWS Migration Hub
Refactor Spaces



Reduce the time to setup and manage a refactor environment



Shield application consumers from infrastructure changes



Re-route traffic from old to new across multiple AWS accounts

Start **refactoring** applications in **days** instead of months

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Migration Hub Refactor Spaces: An application transformation feature

What does it solve?



- Setting up an infrastructure for refactoring, rewriting, and rearchitecting apps
- Operating apps at scale while they are changing

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Migration Hub Refactor Spaces: An application transformation feature

What does it solve?



- Setting up an infrastructure for refactoring, rewriting, and rearchitecting apps
- Operating apps at scale while they are changing

What does it orchestrate?



- AWS Transit Gateway and VPCs to bridge networking across accounts to simplify old and new communication services
- API Gateway and VPC link for external API access to transparently add new services and incrementally route traffic from old to new

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How does Migration Hub Refactor Spaces do it?

Environment

Multi-account home for an application

- Provides unified networking and services across accounts
- Account-level isolation of services to support dev process and team alignment

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How does Migration Hub Refactor Spaces do it?

Environment

Multi-account home for an application

- Provides unified networking and services across accounts
- Account-level isolation of services to support dev process and team alignment

Application

Contains services and routes to them

- Provides configurable request routing to legacy and refactored microservices
- Underlying architecture changes transparent to app consumers
- Used to provide proxy for Strangler Fig pattern

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How does Migration Hub Refactor Spaces do it?

Environment

Multi-account home for an application

- Provides unified networking and services across accounts
- Account-level isolation of services to support dev process and team alignment

Application

Contains services and routes to them

- Provides configurable request routing to legacy and refactored microservices
- Underlying architecture changes transparent to app consumers
- Used to provide proxy for Strangler Fig pattern

Service

Business capabilities reachable via unique endpoints

- Exist inside a Refactor Spaces environment and application
- Communicate across accounts within an environment
- Accessed via configurable routes for inbound requests

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How does Migration Hub Refactor Spaces do it?

Environment

Multi-account home for an application

- Provides unified networking and services across accounts
- Account-level isolation of services to support dev process and team alignment

Application

Contains services and routes to them

- Provides configurable request routing to legacy and refactored microservices
- Underlying architecture changes transparent to app consumers
- Used to provide proxy for Strangler Fig pattern

Service

Business capabilities reachable via unique endpoints

- Exist inside a Refactor Spaces environment and application
- Communicate across accounts within an environment
- Accessed via configurable routes for inbound requests

Route

Configurable resources contained within an application

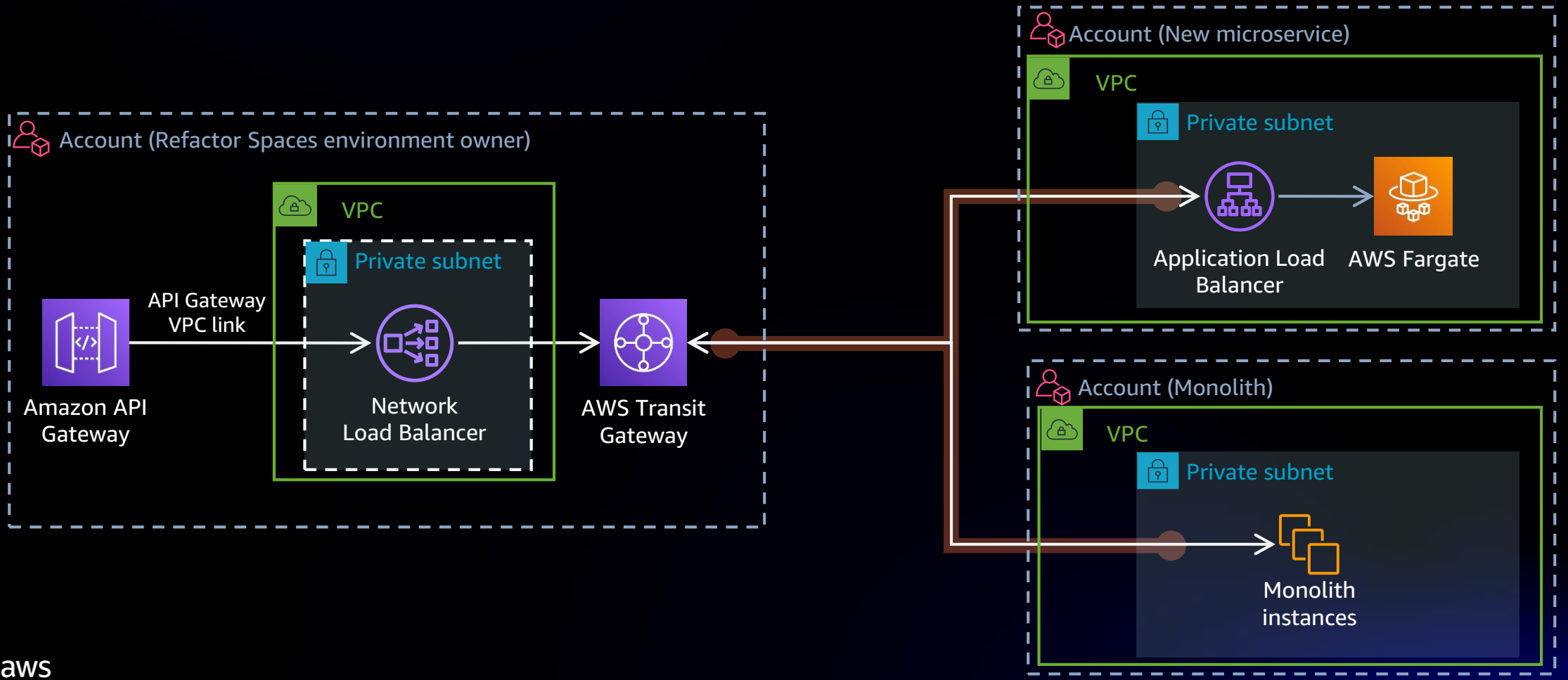
- Request routing from external HTTP gateway to targeted application services
- Route targets include protocol, DNS name, port, URI path

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Migration Hub Refactor Spaces

Create environment → Add accounts → Create application → Create services with VPC → Add routes



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Learn more about refactoring with Migration Hub

Sessions

- ENT321 – Incremental application refactoring to increase business agility
- ENT402 – Stay decoupled when refactoring using multiple AWS accounts

Workshop

- ENT302 – Incremental refactoring: Evolve existing apps with microservices

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Wrap up

How to de-risk modernization



- Take a two-phase approach: migrate then modernize
- Adopt an incremental approach to refactoring using the Strangler Fig pattern on AWS
- Use the Amazon API Gateway as the required proxy layer

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Wrap up

How to de-risk modernization



- Take a two-phase approach: migrate then modernize
- Adopt an incremental approach to refactoring using the Strangler Fig pattern on AWS
- Use the Amazon API Gateway as the required proxy layer

How to decompose the monolith



- Use event storming to model the microservices around your business
- Start decomposing slowly by first decomposing by auxiliary service
- Business objectives should guide the selection of the decomposition pattern

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Wrap up

How to de-risk modernization



- Take a two-phase approach: migrate then modernize
- Adopt an incremental approach to refactoring using the Strangler Fig pattern on AWS
- Use the Amazon API Gateway as the required proxy layer

How to decompose the monolith



- Use event storming to model the microservices around your business
- Start decomposing slowly by first decomposing by auxiliary service
- Business objectives should guide the selection of the decomposition pattern

How to manage the complexity of a multi-account infrastructure



- AWS Migration Hub Refactor Spaces automates the provisioning of resources required for implementing the Strangler Fig pattern in a multi-account environment

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Get started with Refactor Spaces today

01 Quickly setup an infrastructure for refactoring, rewriting, and rearchitecting apps

02 Operate apps at scale while they are changing

03 Focus on delivering value from refactoring without having to create the environment

Join the Migration Hub Refactor Spaces preview now

Visit Migration Hub home page, Migration Hub console, User Guide, contact us: aws-flux-bd@amazon.com

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Melash Moodley
mmelash@amazon.com

Carlos Tagle
taglec@amazon.com

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.