AWS re:Invent

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

ENT207

Identifying target architectures on AWS

John Standish (pronouns: he/him)
Sr. Solutions Architect, Modernization Leader
AWS

Sibu Kutty (pronouns: he/him)
WW Tech Leader, Migration & Modernization
AWS

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

What can you expect?

There are several ways to modernize existing enterprise applications; let’s discuss the common patterns and pathways, how to decide on one versus another, and the right target state architecture components for a typical traditional enterprise n-tier application

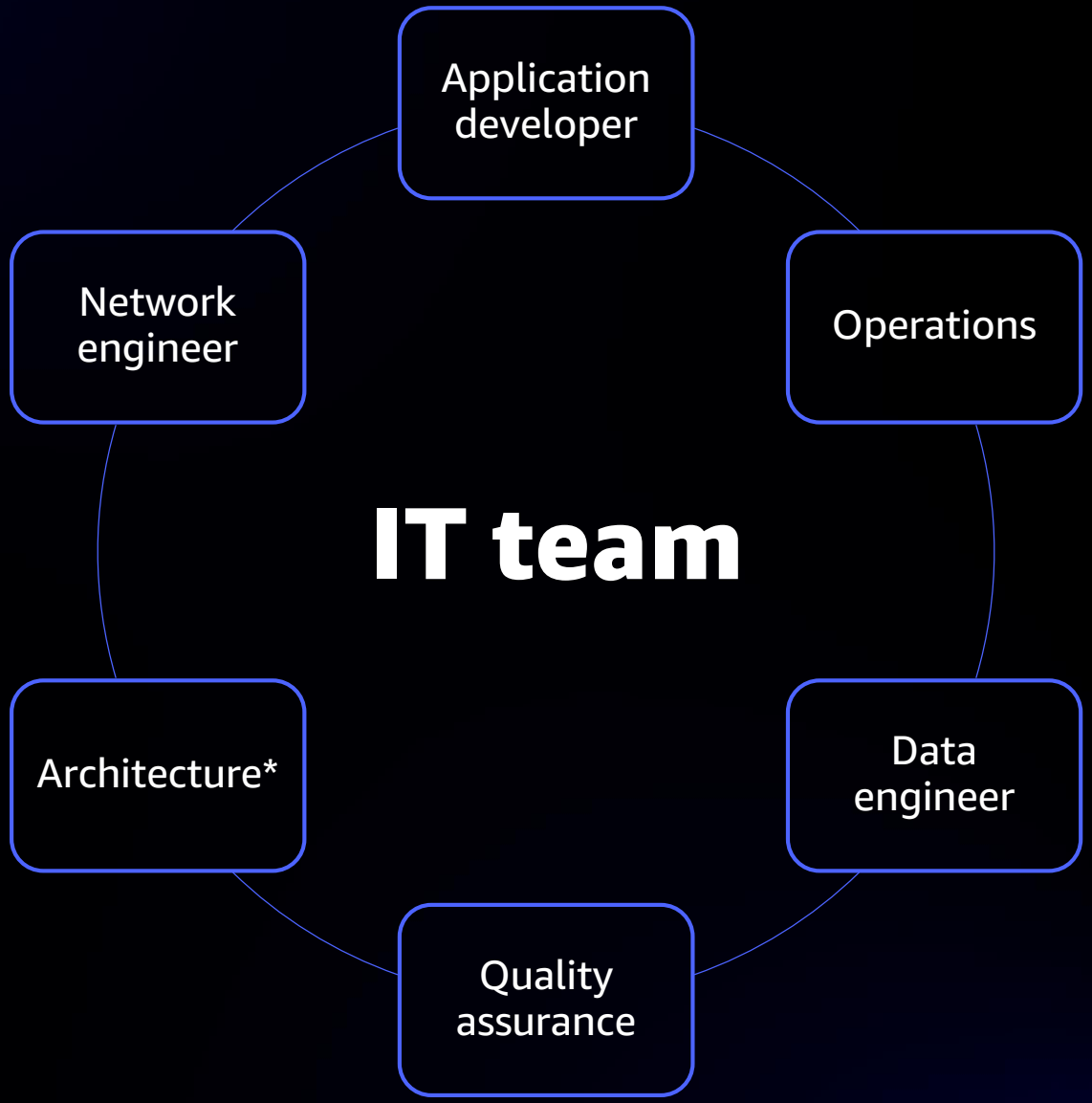
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Before we dive in, some baselining

How are we operating within an enterprise today?

- Your team structure
- Different roles and personas
- Project delivery models and persona based metrics

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



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

Roles and personas

Application developer

System admin

Application architect

DevOps architect

DBA

Project manager

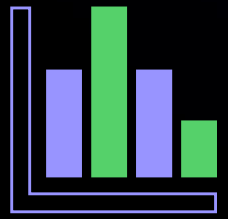
Business analyst

Enterprise architect

Cloud operations

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

Delivery metrics



Velocity



Sprints



KPIs



Success metrics

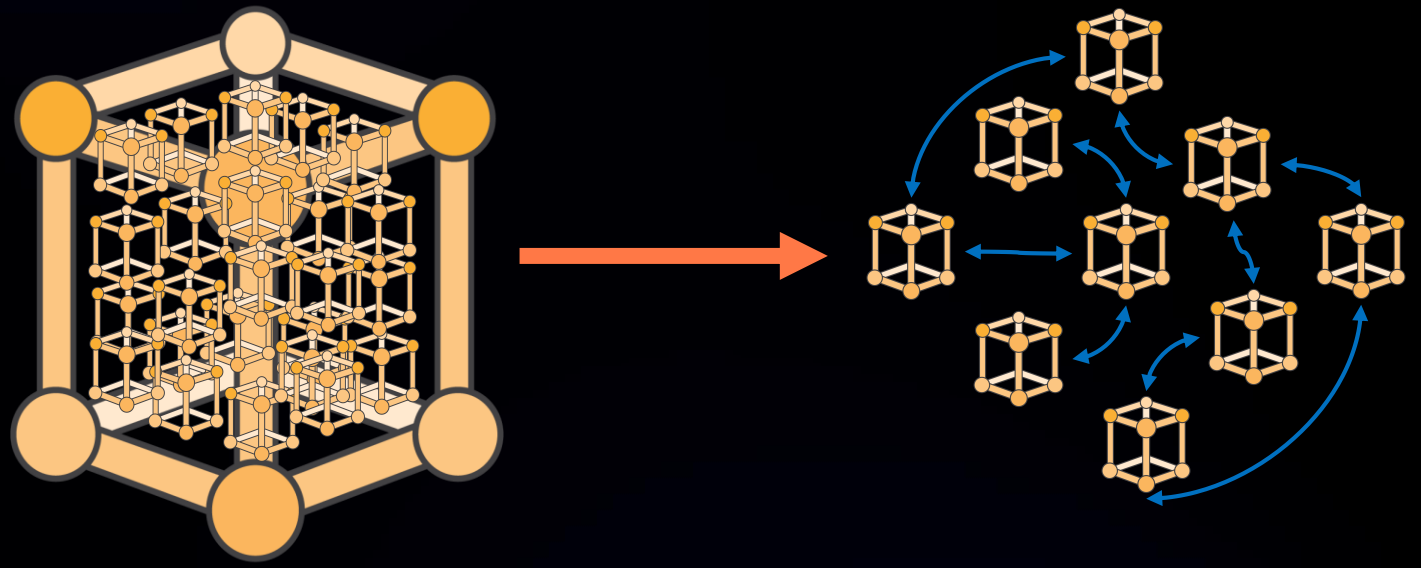
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How do you think about modernization?

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

Most sought-after modernized state

REFACTOR TO CLOUD NATIVE



Monolith to microservices

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

Modernizing an existing application

Refactor to cloud native

- Rewrite to serverless/containers, 12-factor app, managed services
- Fits “core” mission-critical apps

Relocate with containers

- No change to functionality/source code
- Gain operational efficiency
- Fits vendor apps and apps within the enterprise

Replatform components

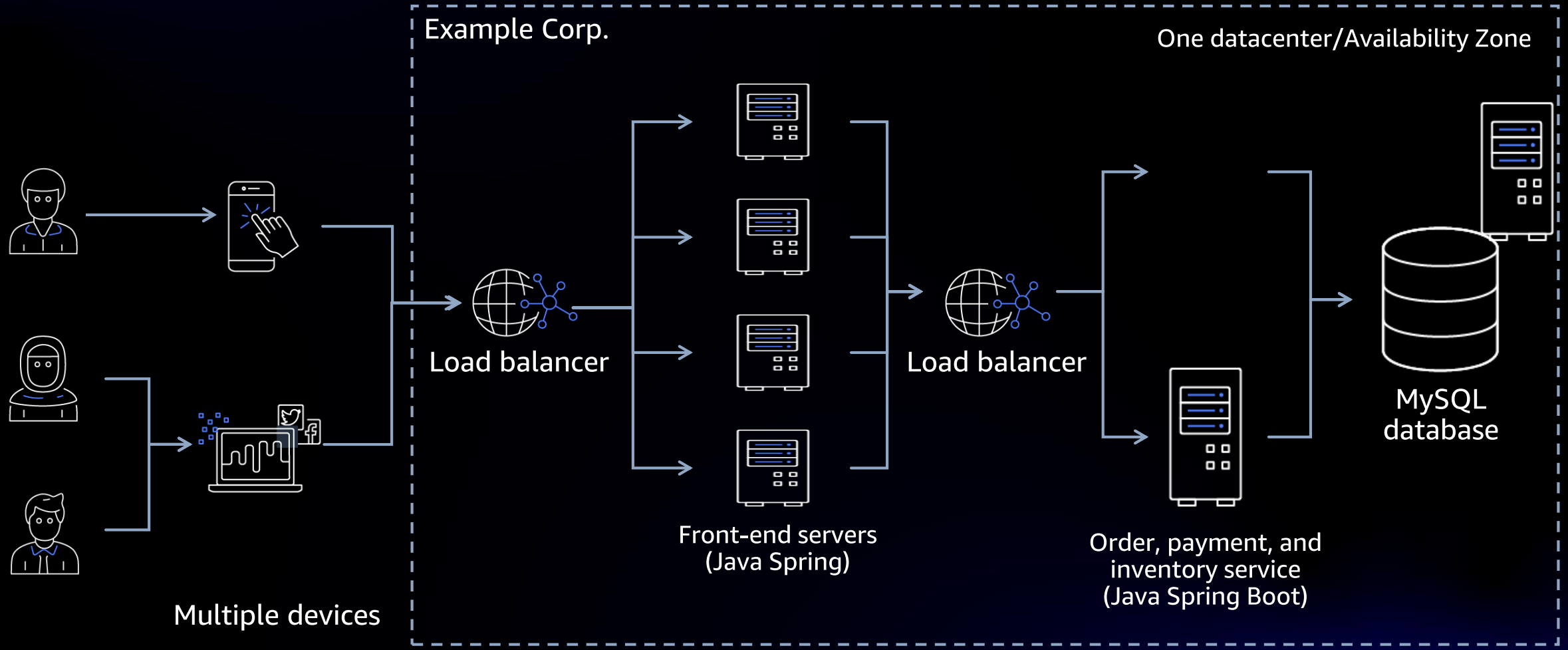
- Move to cloud-centric, open source, lightweight, and cloud ready
- Fits apps where a full refactor may not be feasible

aws

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

Lets take a sample n-tier web application

CURRENT STATE



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

Solving the problem in groups

Let’s split into teams

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

Application target state goals

Business expectations

1. Reduce overall cost
2. Increase per-order reliability
3. Deliver changes as soon as completed without downtime

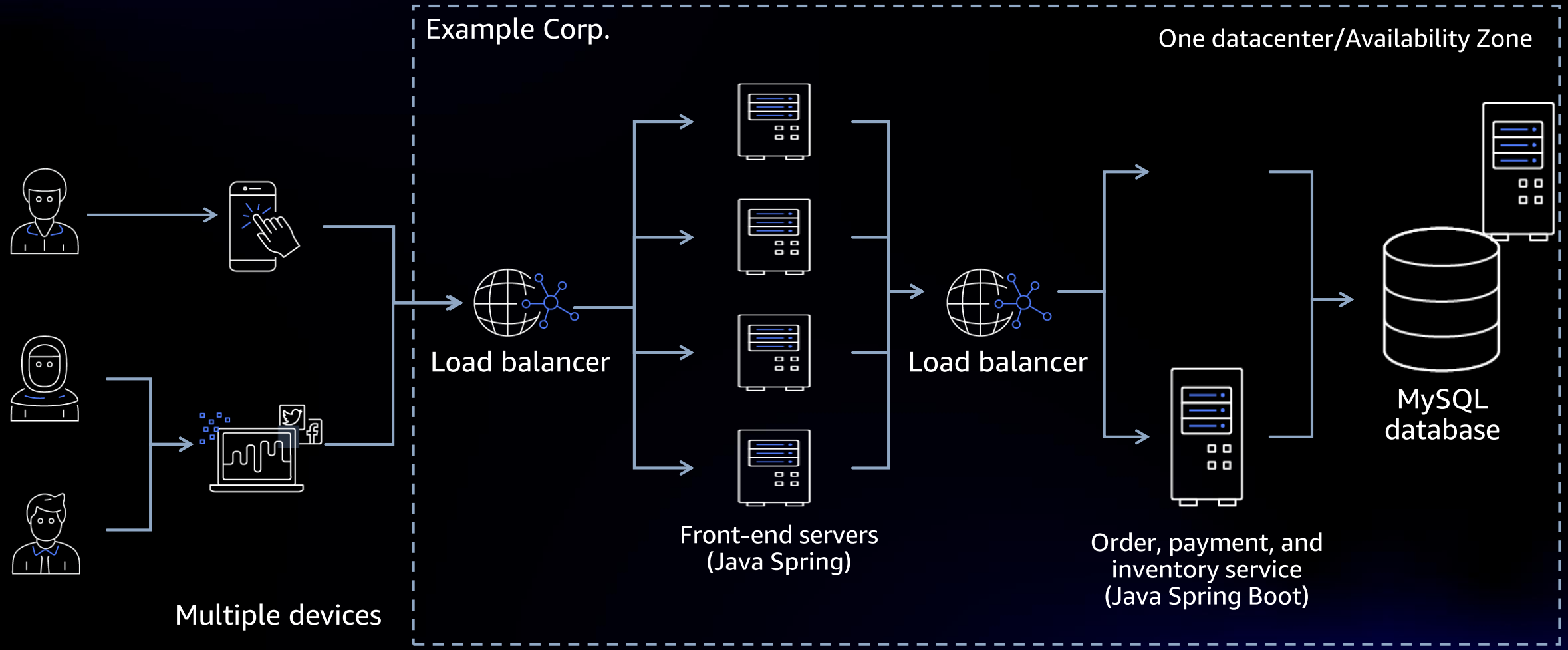
Goals

1. Improve peak holiday load capacity
2. Reduce time to delivery
3. Add new features: user activity tracking and personalized incentives
4. Increase reliability of payments

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

Architecture

CURRENT STATE



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

Break into teams (15 minutes)

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

What approach applies to these goals?

Refactor to cloud native

- Rewrite to serverless/containers, 12-factor app, managed services
- Fits “core” mission-critical apps

Relocate with containers

- No change to functionality/source code
- Gain operational efficiency
- Fits vendor apps and apps within the enterprise

Replatform components

- Move to cloud-centric, open source, lightweight, and cloud ready
- Fits apps where a full refactor may not be feasible

aws

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

What approach applies to these goals?

Refactor to cloud native

- Rewrite to serverless/containers, 12-factor app, managed services
- Fits “core” mission-critical apps

Relocate with containers

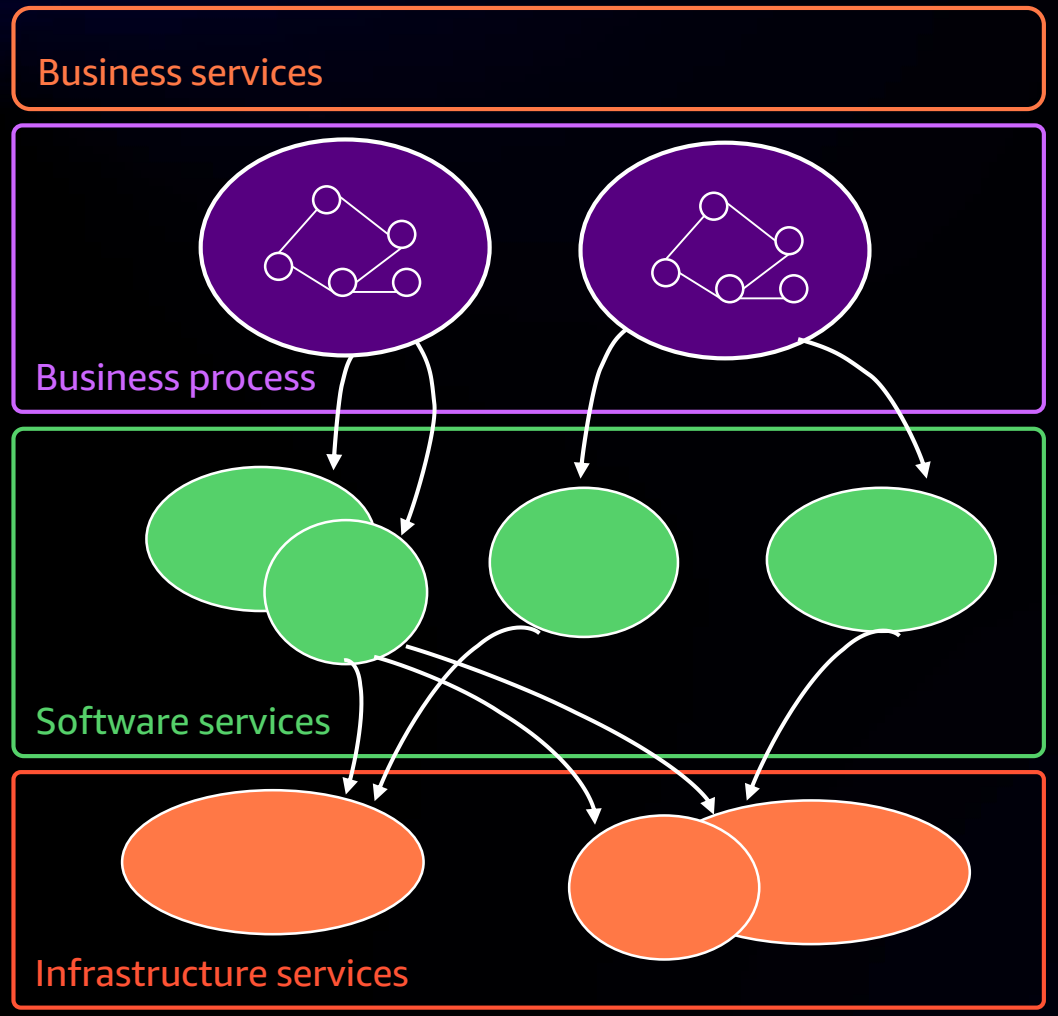
- No change to functionality/source code
- Gain operational efficiency
- Fits vendor apps and apps within the enterprise

Replatform components

- Move to cloud-centric, open source, lightweight, and cloud ready
- Fits apps where a full refactor may not be feasible

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

Identifying the right microservices



- Domain-driven design
- Process decomposition
- Goal modeling
- Business-layer division
- Vertical slicing
- Component analysis

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

Identifying the right microservices



What if we can look at user behaviors?

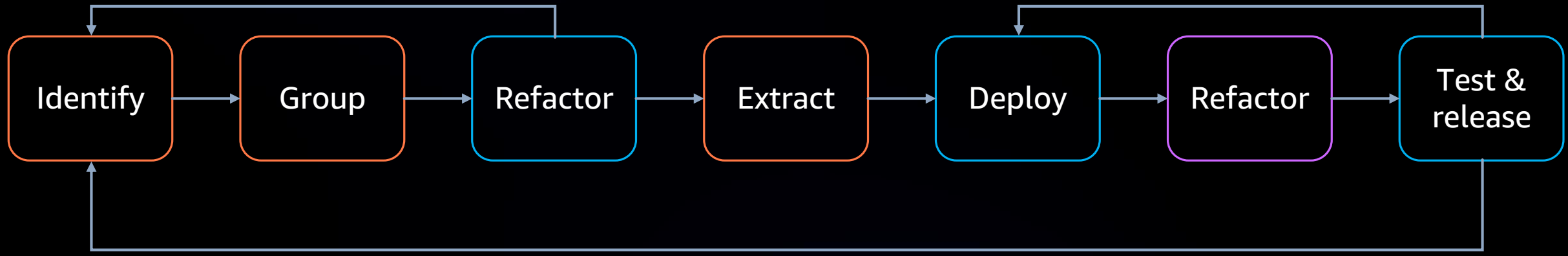
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Microservice Extractor for .NET



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

AWS Microservice Extractor for .NET refactoring



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

Supported environment

- .NET Framework 4.7 – 4.8 with C#
- ASP .NET web applications
- Excludes non-code files (HTML, CSHYML, XML, etc.)
- Excludes databases

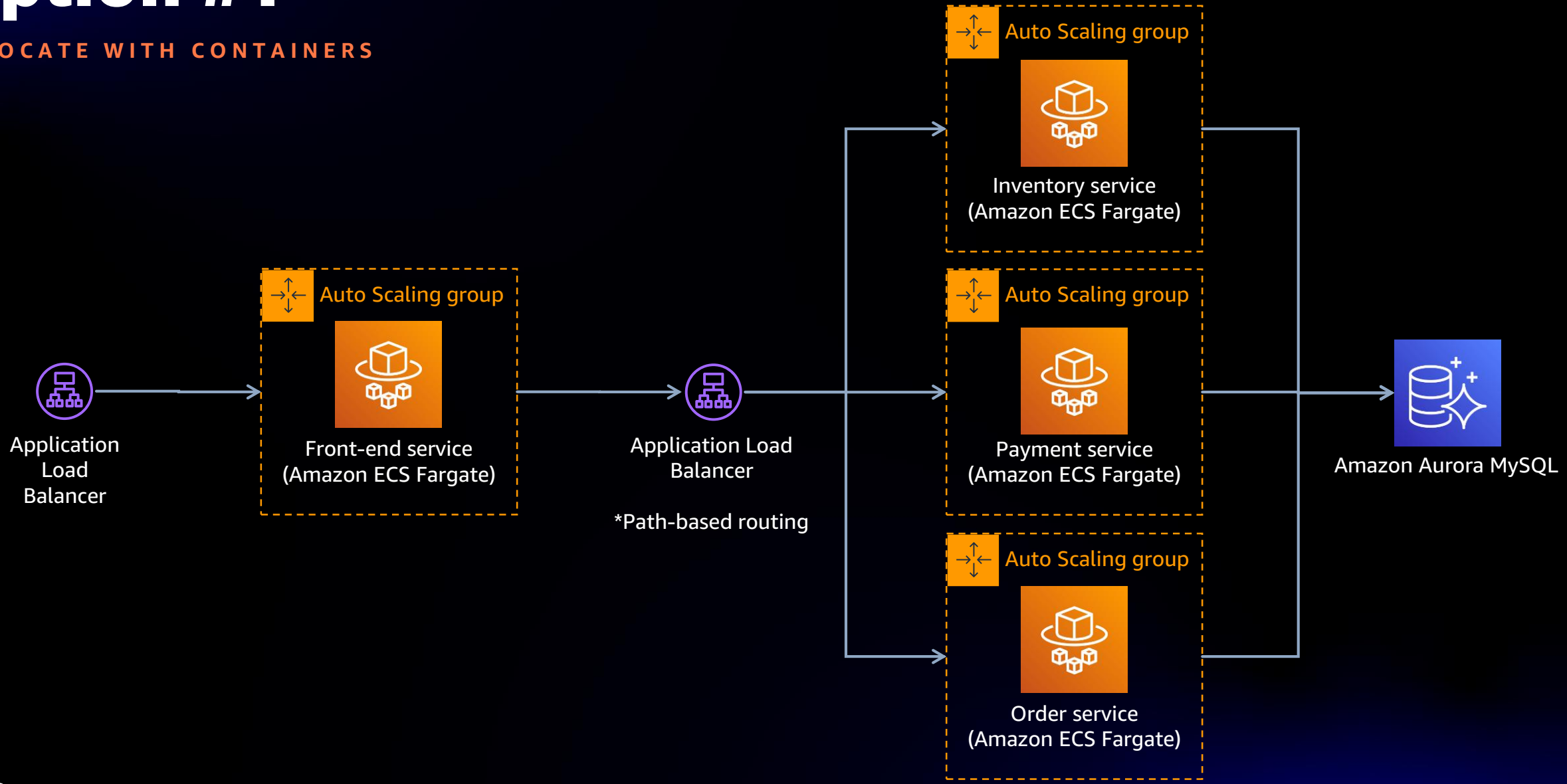
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Approaches

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

Option #1

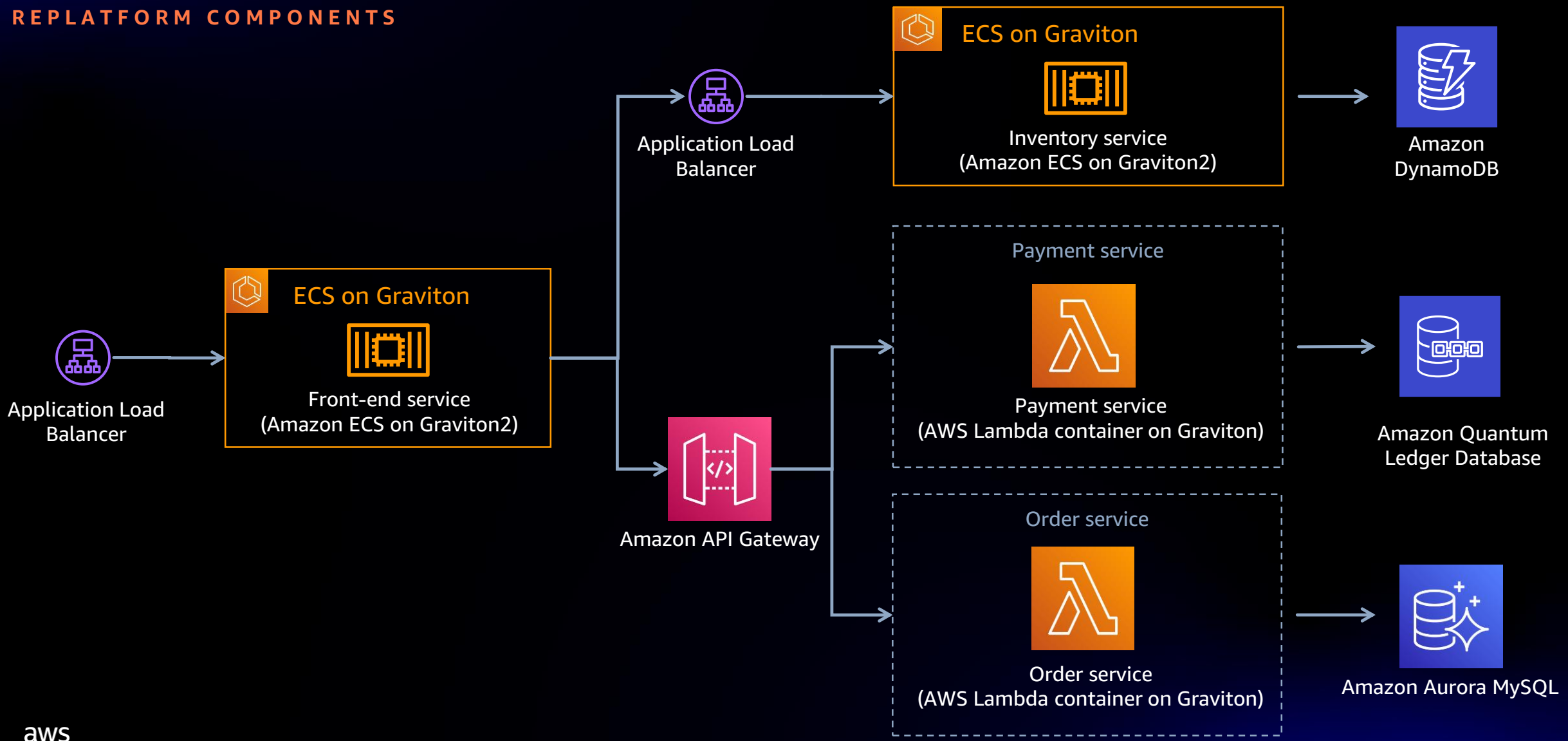
RELOCATE WITH CONTAINERS



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

Option #2

REPLATFORM COMPONENTS



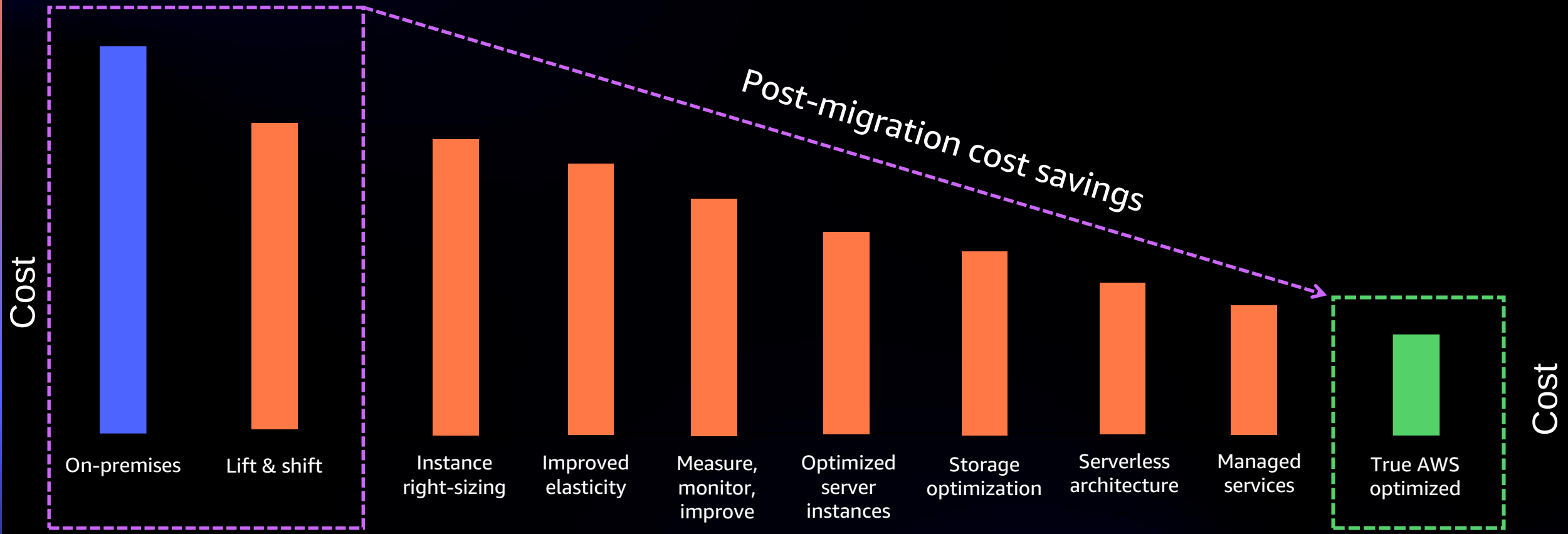
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Measuring modernization success

Refactor	Relocate	Replatform
Key business goals Flexibility Agility	MTTR, RTO, RPO Level of automation (CloudOps)	TCO

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

Continuous improvement and optimizations



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

Thank you!

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