Clear as Clouds

New approaches with AWS

Apeksha Jain
AWS Senior Solutions Architect
jainapek@amazon.com
24th March 2021

Walk through of next 45 minutes

- re:Cap
- Migration, Modernisation and Transformation
- Different Migration Paths
- Patterns for Decomposition
- Technology and Use Cases
    - Serverless
    - Containers
    - Infrastructure as Code
- Research and Ronin
- Thinking Beyond

re:Cap from last time

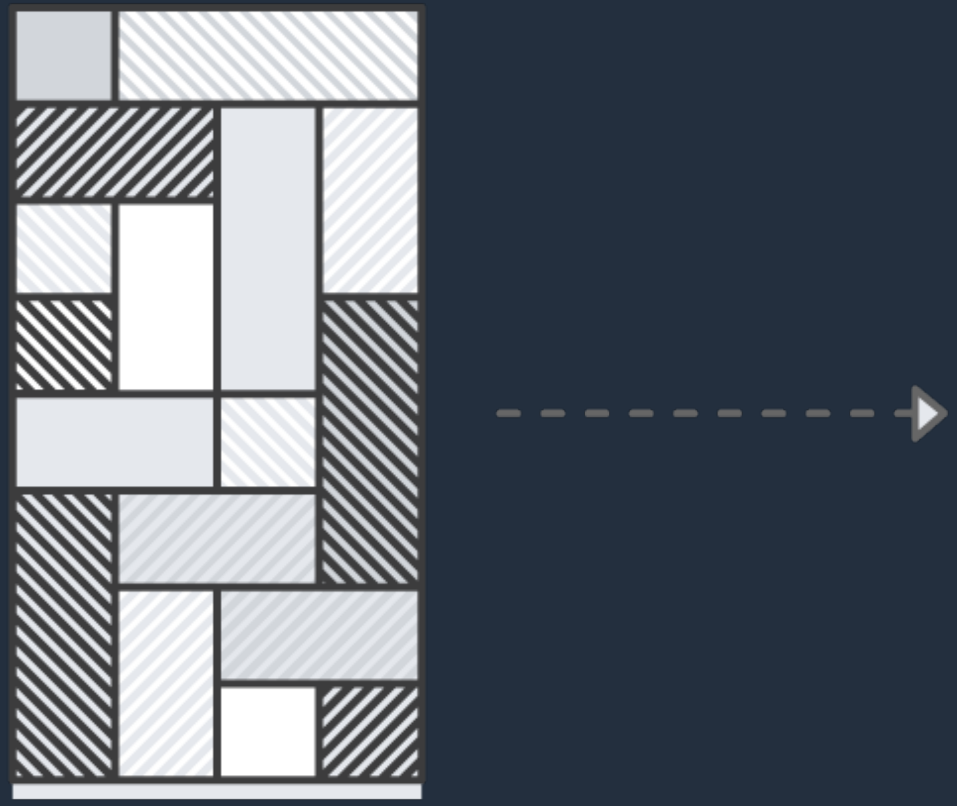
Migration, Modernisation and Transformation

What's the difference?

[Link to a podcast on the topic](#)

Development Transformation at Amazon.com

1994-2001



Monolithic architecture + hierarchical organization

2002+

Decoupled services + Two-pizza teams

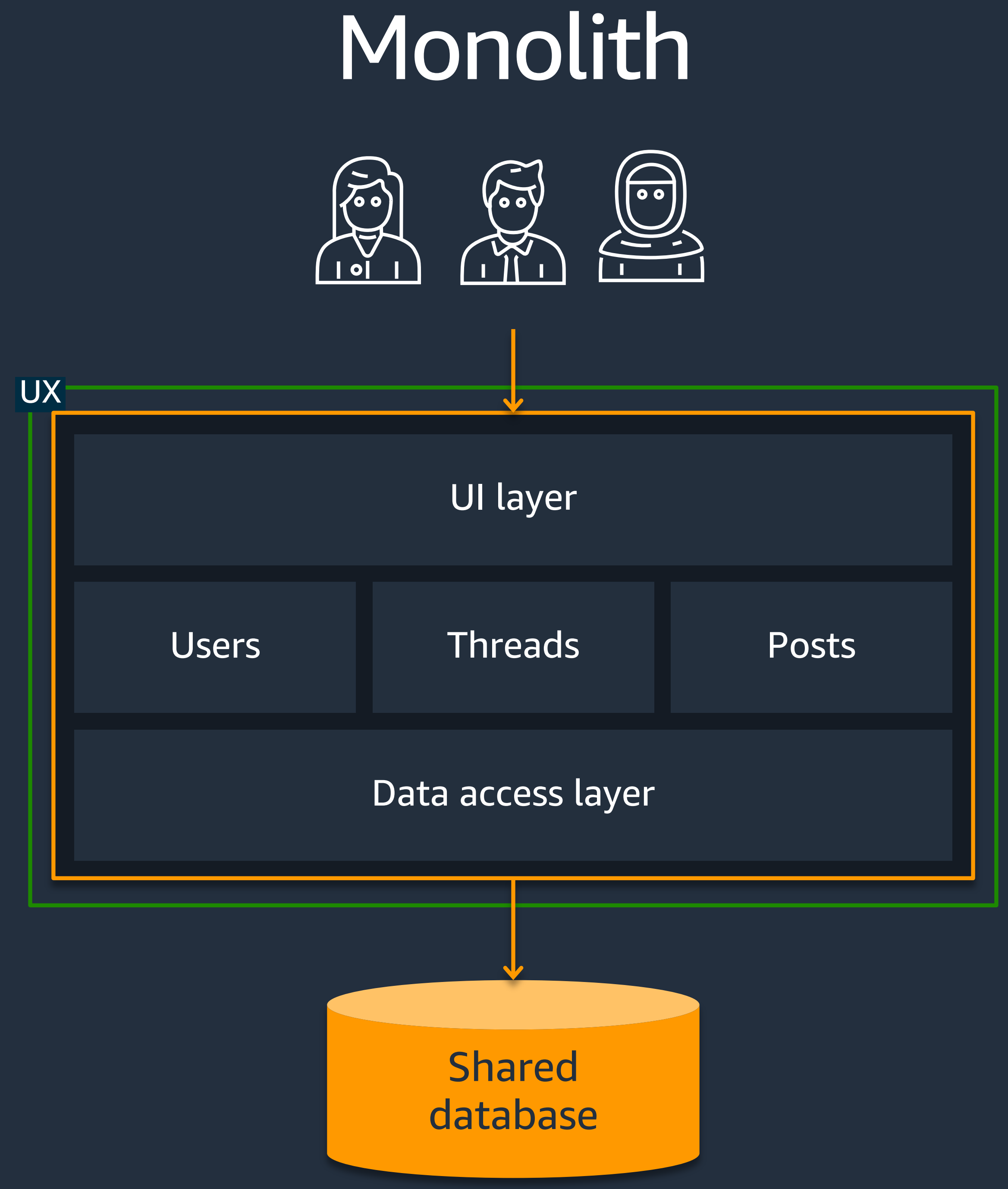
Break up the work

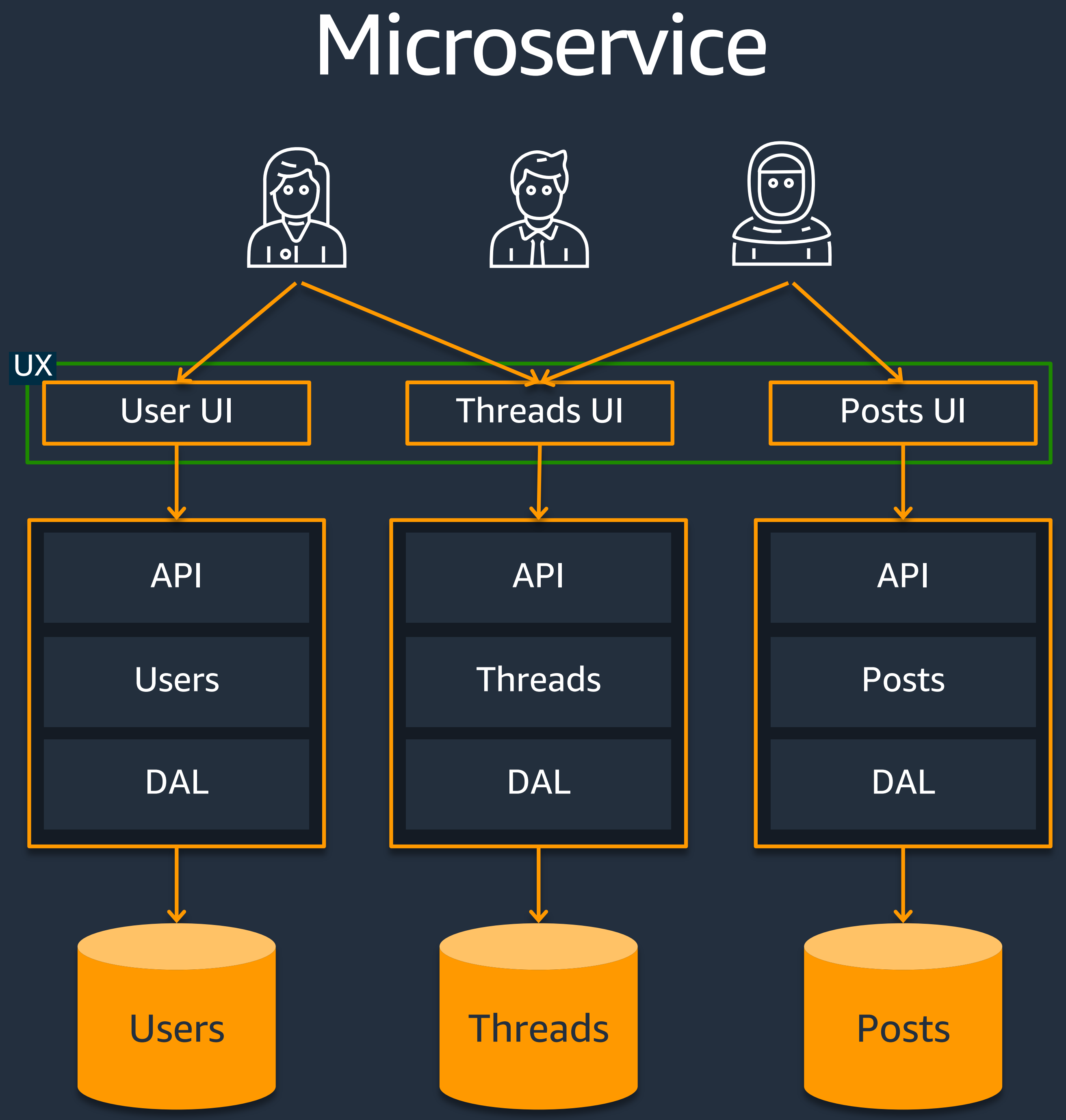
People

Smaller teams that own products and services, and organize around customer outcomes

Components

Smaller releasable components that are easier to own and operate, with well-defined boundaries, and a smaller blast radius


Monolith
UX
UI layer
Users
Threads
Posts
Data access layer
Shared database


Microservice
UX
User UI
Threads UI
Posts UI
API
Users
DAL
API
Threads
DAL
API
Posts
DAL
Users
Threads
Posts

If time was not an issue…

How to select the correct application

Business Reasons	Technical Reasons
LOB, critical to business success	Old technology – no support
Customer facing	Performance issues, inability to scale
Significant impact to revenue	Lack of extensibility
Proprietary business logic (not HR, CRM)	Lack of skill sets, rewrite and build skills
Impacts a large # of customers	Tightly coupled, single codebase
Market differentiator	Difficult to integrate
Value exceeds cost	Expensive to run (move to open source)
	Too many bugs, spaghetti code

Migration Paths

The 6 Rs

Know your application portfolio and *understand your options*

Reduce the size of your estate*

- Retire
- SaaS

Move to AWS

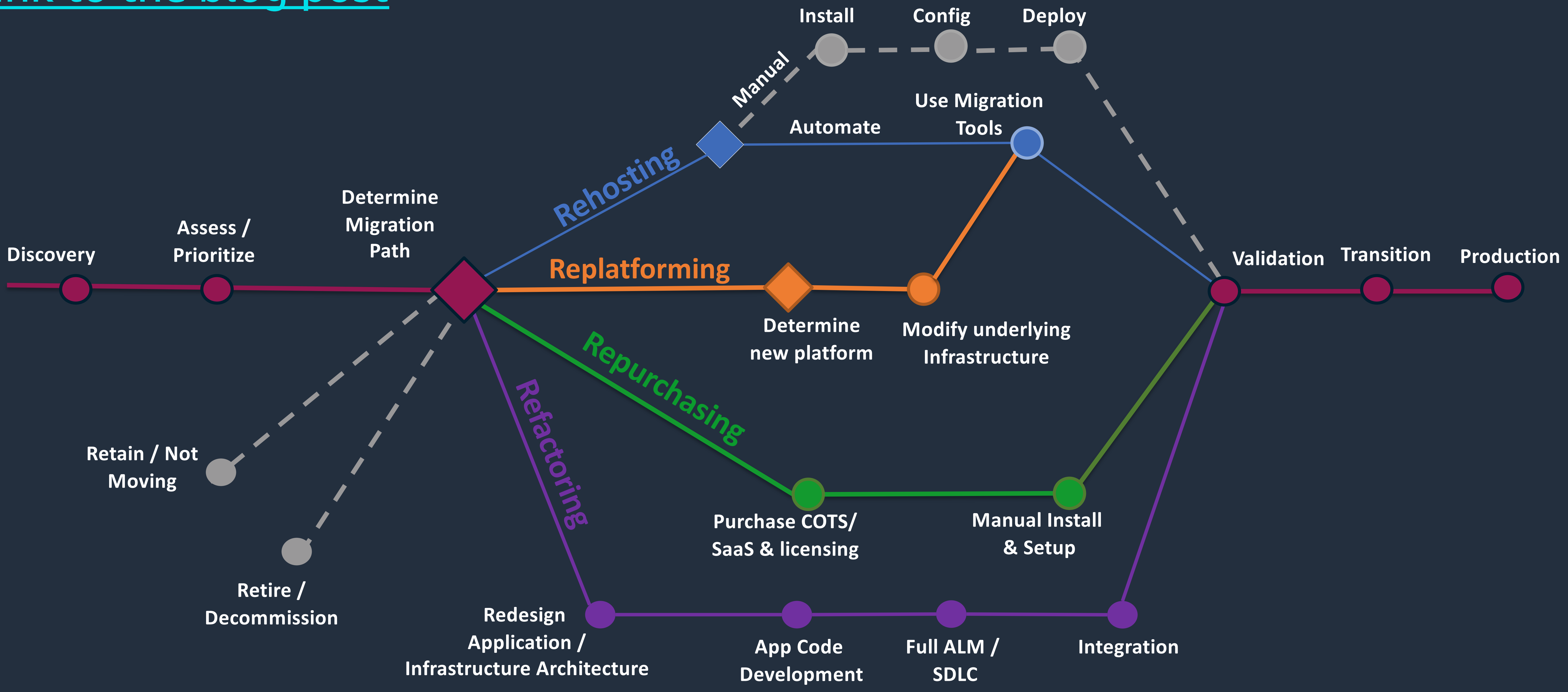
- Lift and shift

Modernize on AWS

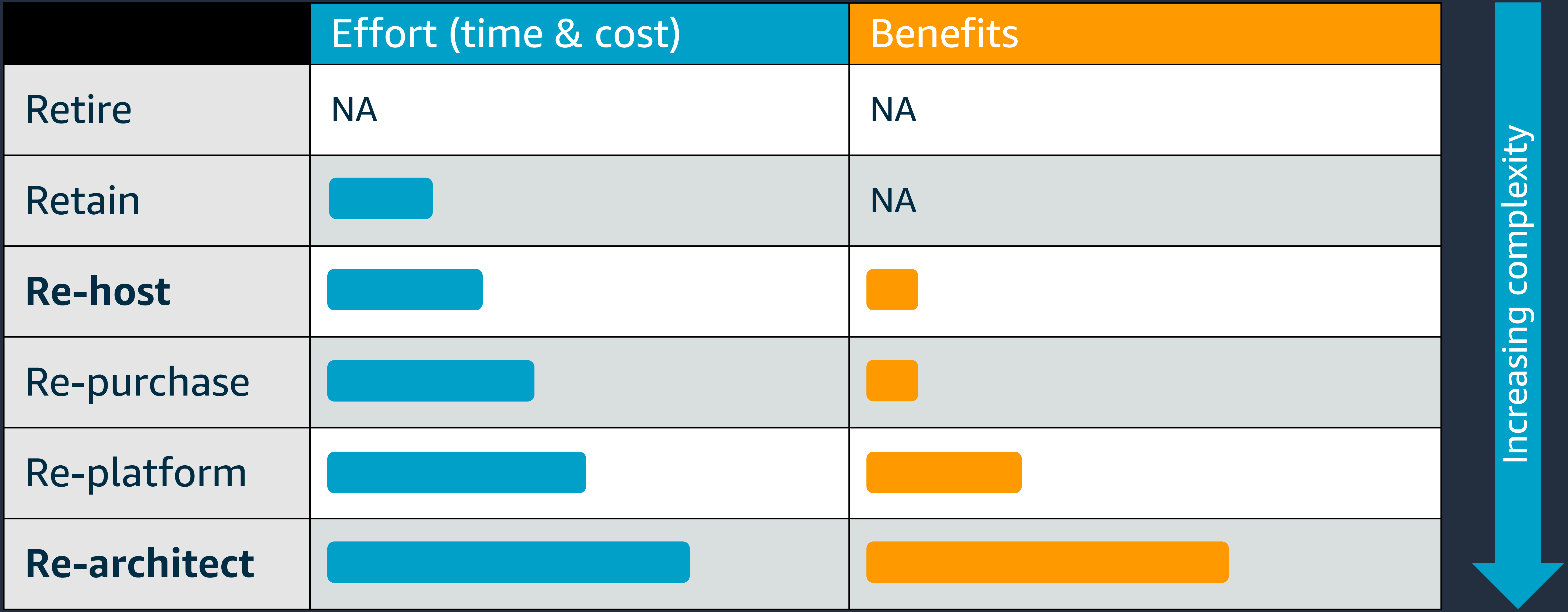
- Refactor
- Re-platform

Application migration strategies

[Link to the blog post]



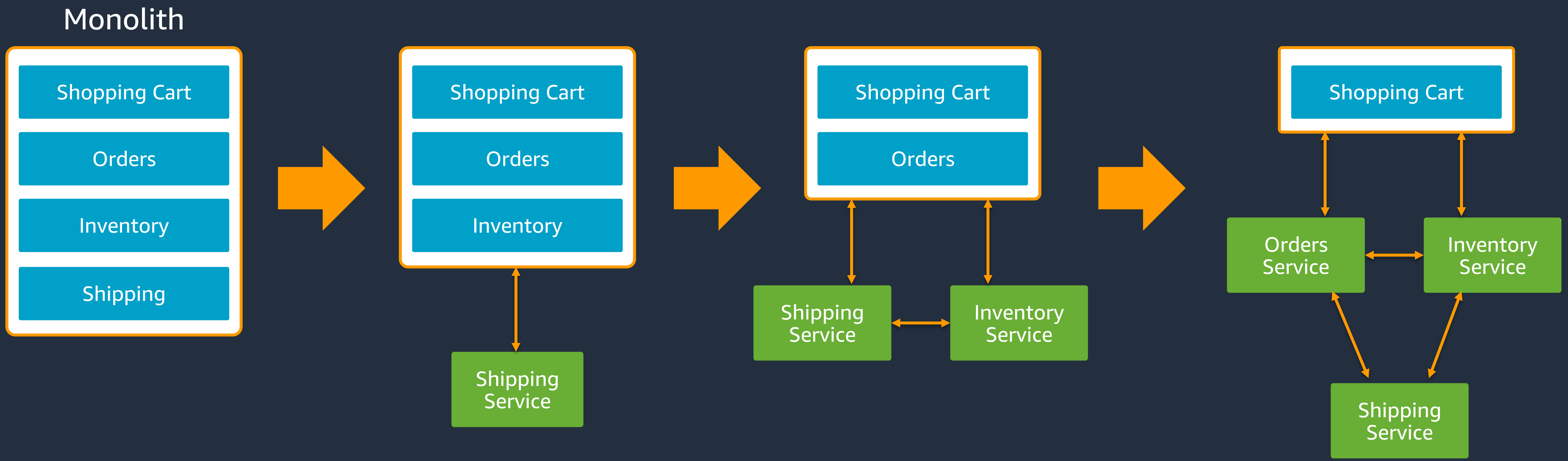
Comparison of Migration Strategies



	Effort (time & cost)	Benefits
Retire	NA	NA
Retain	▬▬	NA
Re-host	▬▬▬	▬
Re-purchase	▬▬▬▬	▬
Re-platform	▬▬▬▬▬	▬▬▬
Re-architect	▬▬▬▬▬▬▬	▬▬▬▬▬▬▬

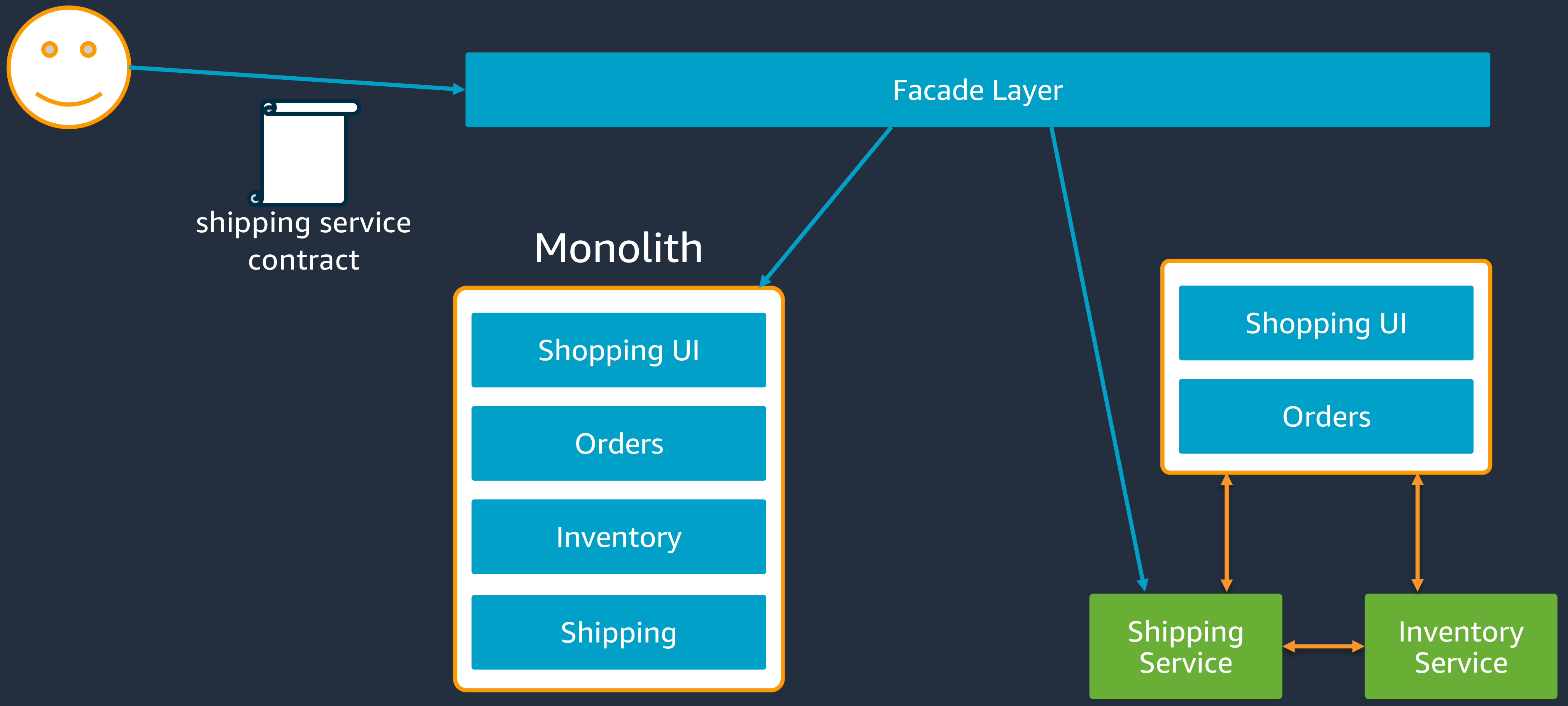
Patterns for Decomposition

Strangler Pattern example



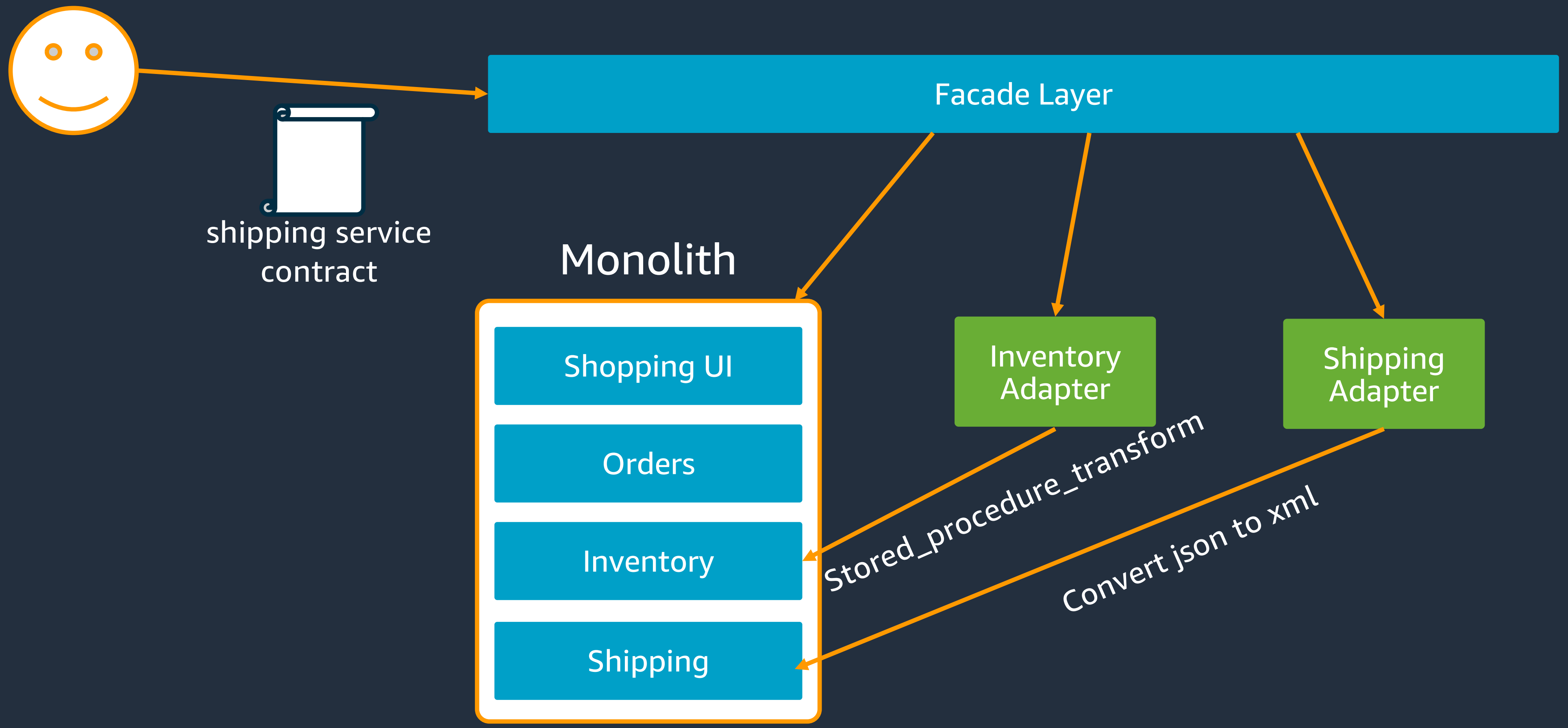
Facade Pattern

Analogous to a **facade** in architecture, a **facade** is an object that serves as a front-facing interface masking more complex underlying or structural code.



Adapter Pattern

An adapter is a wrapper that allows an interface to communicate with another interface without modifying the source code.



Technologies to help

What is serverless?

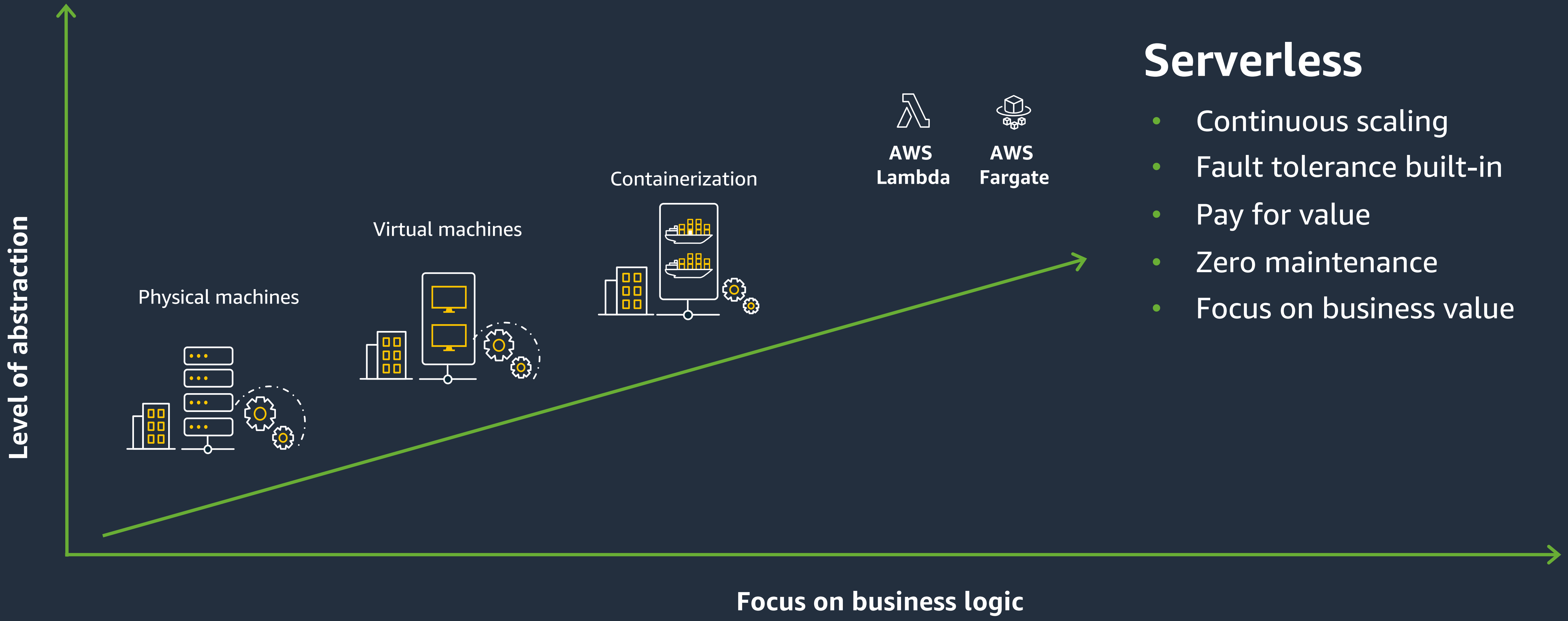
No infrastructure provisioning, no management

Automatic scaling

Pay for value

Highly available and secure

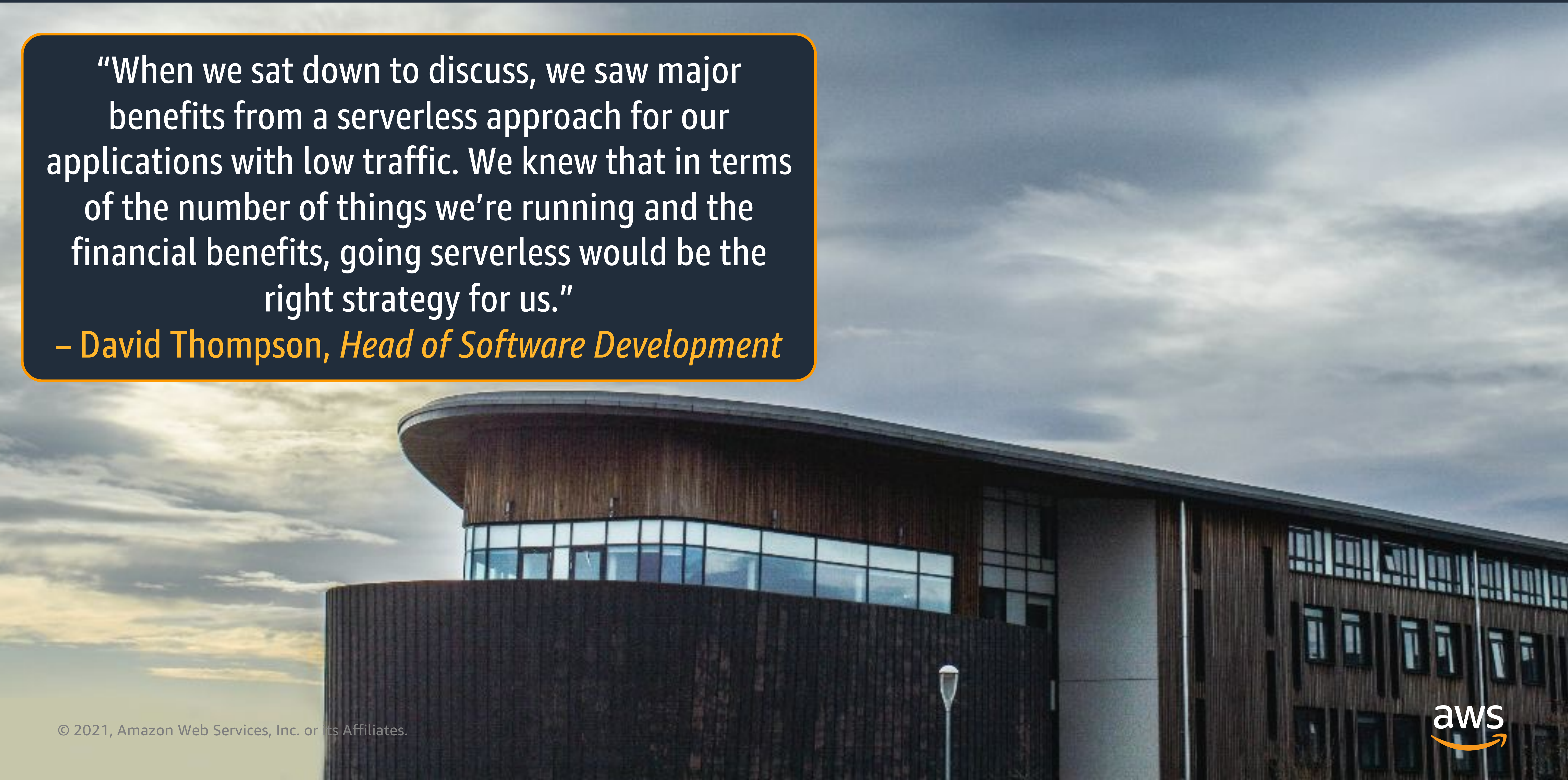
Move up the stack = less work for you



Serverless

- Continuous scaling
- Fault tolerance built-in
- Pay for value
- Zero maintenance
- Focus on business value

University of York Goes Serverless



"When we sat down to discuss, we saw major benefits from a serverless approach for our applications with low traffic. We knew that in terms of the number of things we're running and the financial benefits, going serverless would be the right strategy for us."

– David Thompson, *Head of Software Development*

Containers: Amazon EKS and Amazon ECS

ECS

Powerful simplicity

EKS

Open flexibility

Run your containers anywhere based on your workload needs

Serverless

AWS Fargate

EC2 options

Amazon EC2

Spot instance

Edge and 5G

AWS Local Zones

AWS Wavelength

On-premises

AWS Outposts

EKS Anywhere
ECS Anywhere

Infrastructure as Code

The code template describes the intended state of your resources

CloudFormation translates the intention to API calls

AWS CloudFormation/Terraform

The code template describes the intended state of your resources

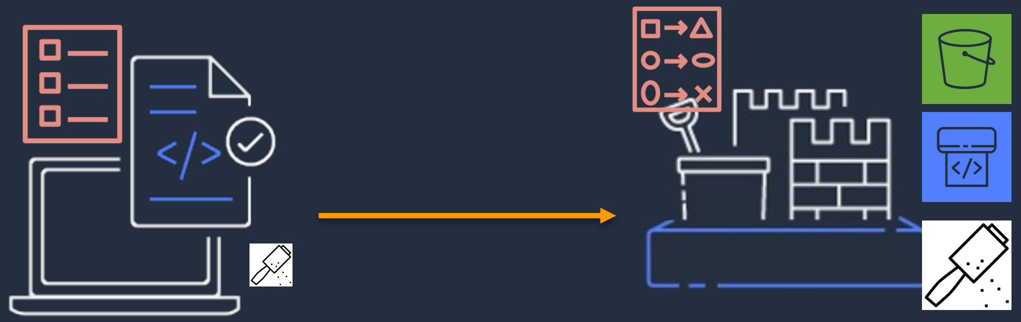
CloudFormation translates the intention to API calls

1 Code your template

AWS CloudFormation/Terraform

The code template describes the intended state of your resources

CloudFormation translates the intention to API calls



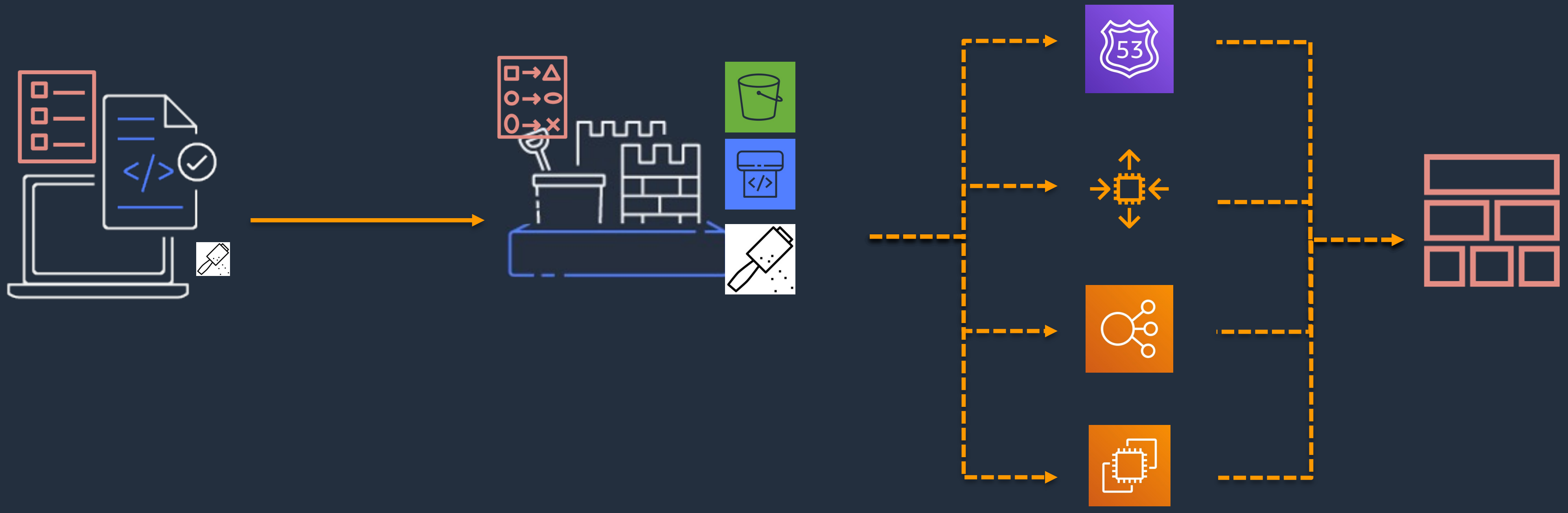
1 Code your template

2 Upload, test, and review changes

AWS CloudFormation/Terraform

The code template describes the intended state of your resources

CloudFormation translates the intention to API calls



1 Code your template

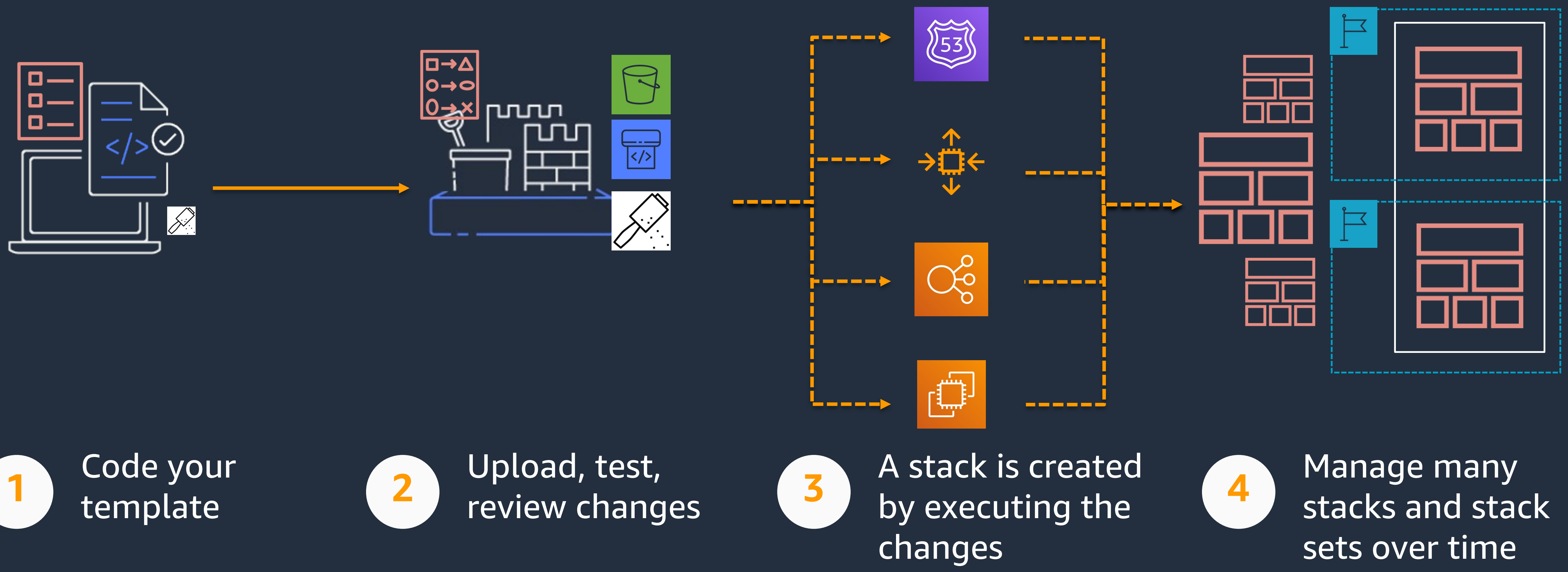
2 Upload, test, and review changes

3 A stack is created by executing the changes

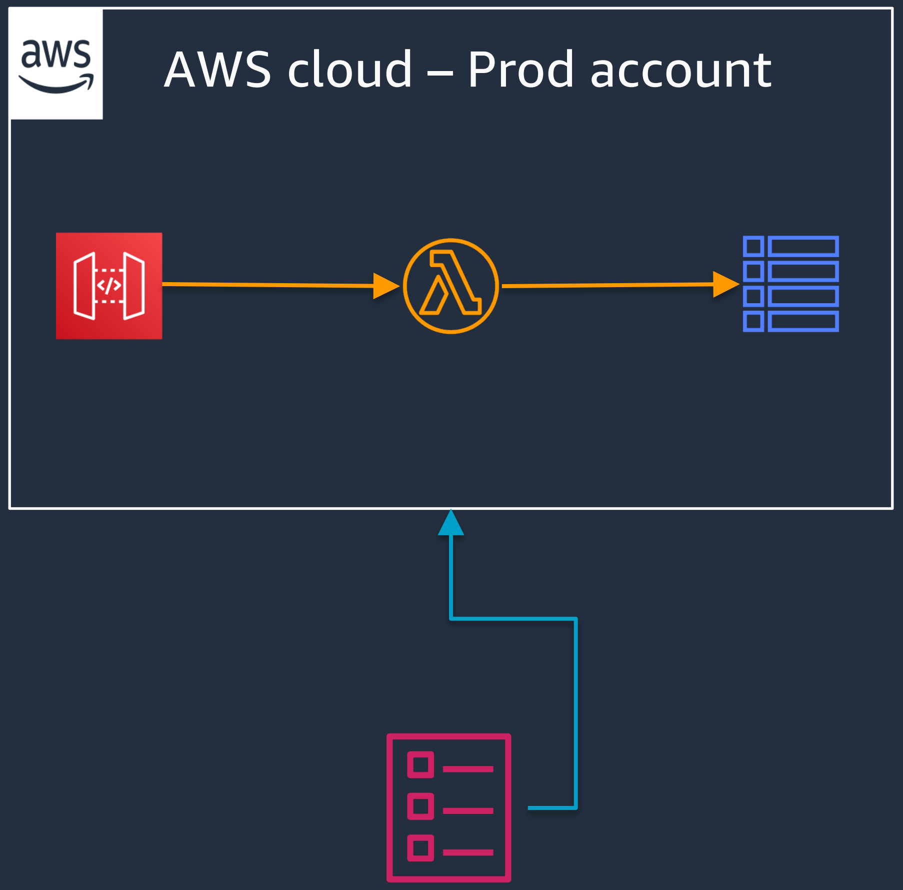
AWS CloudFormation/Terraform

The code template describes the intended state of your resources

CloudFormation translates the intention to API calls

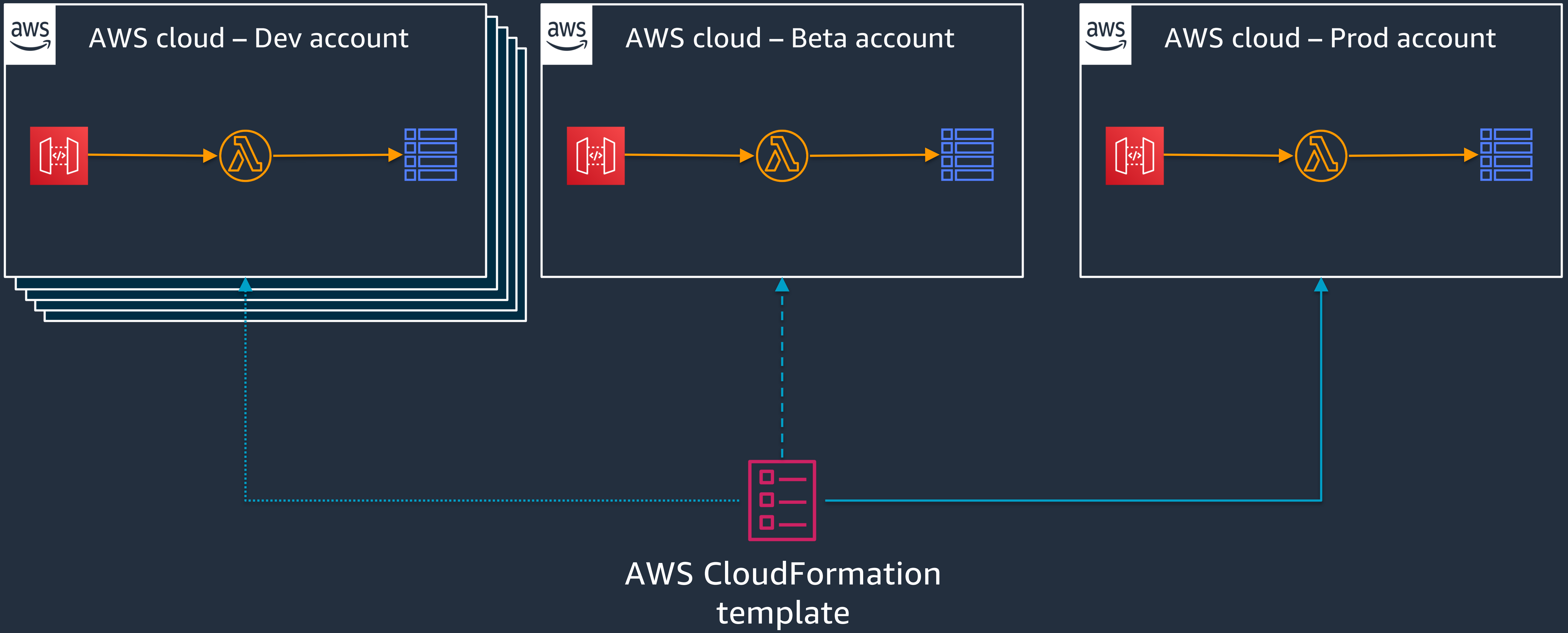

53
1
Code your template
2
Upload, test, review changes
3
A stack is created by executing the changes
4
Manage many stacks and stack sets over time

Why reusable templates?

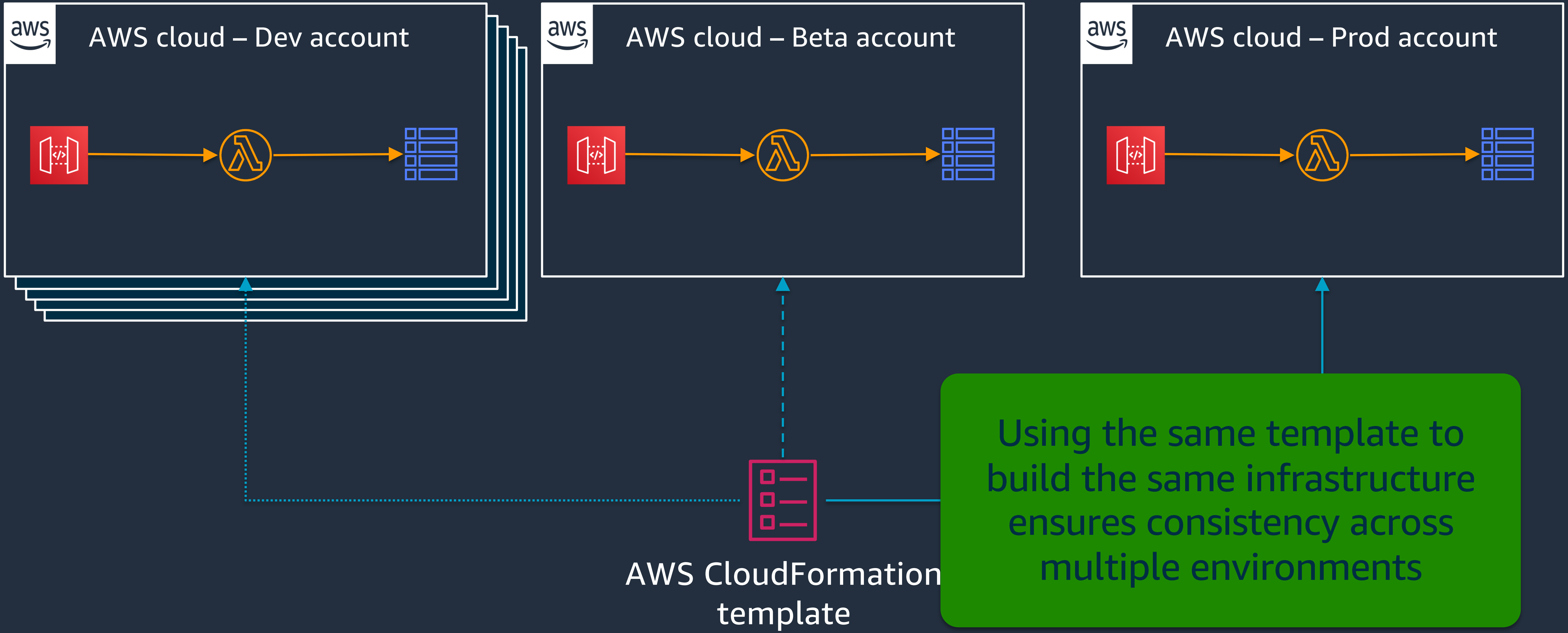


AWS CloudFormation template

Why reusable templates?



Why reusable templates?



Using the same template to build the same infrastructure ensures consistency across multiple environments

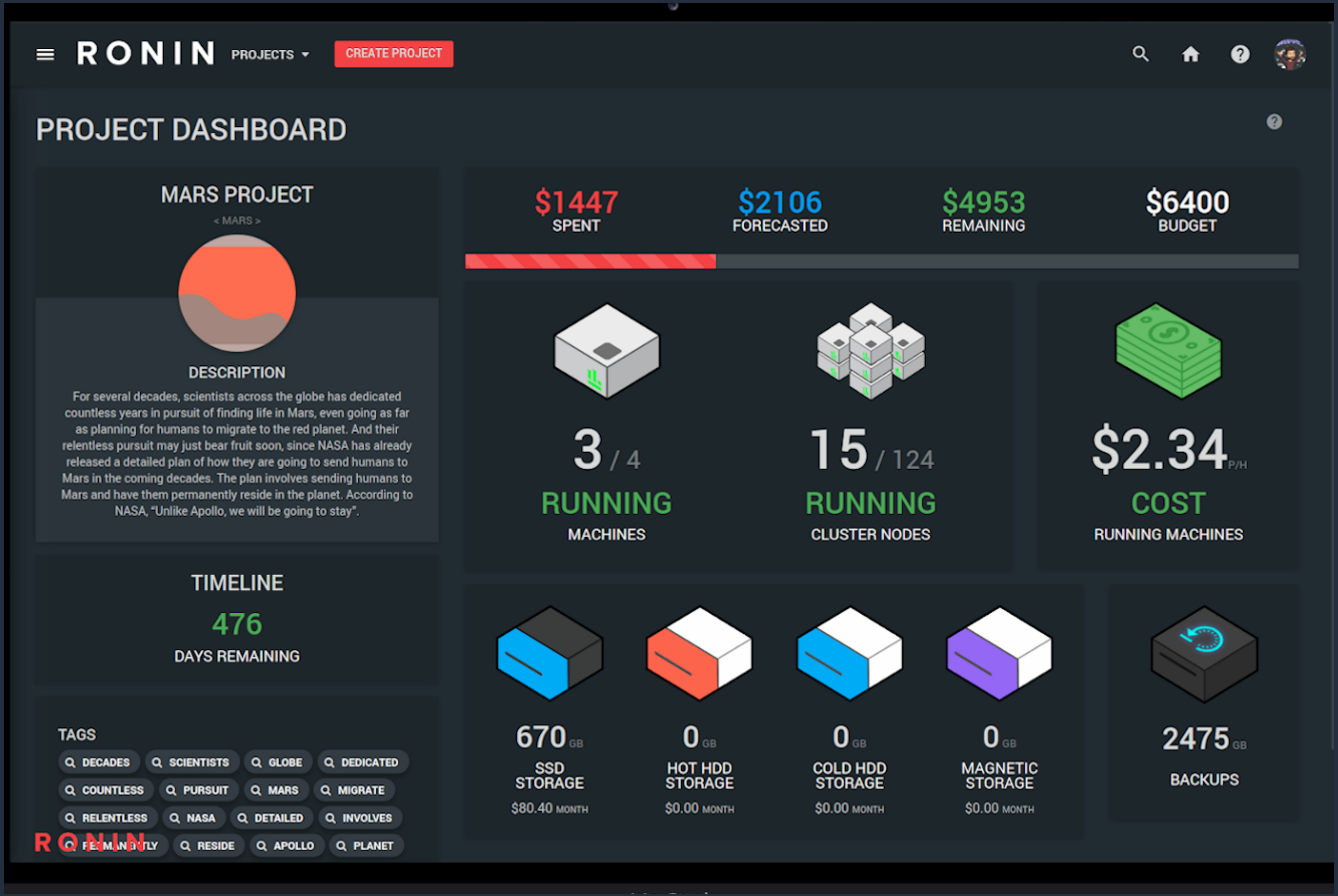
University of Arizona

"Once you've adopted infrastructure as code, your ability to spin up environments that use a new feature or test something, can now be done in minutes, whereas before it would take us weeks. With AWS, that is no longer a bounding factor."
– Derek Masseth, *Chief Technology Officer*

Courtesy of University of Arizona

Transforming Research

RONIN



PROJECT DASHBOARD

MARS PROJECT

< MARS >

DESCRIPTION

For several decades, scientists across the globe has dedicated countless years in pursuit of finding life in Mars, even going as far as planning for humans to migrate to the red planet. And their relentless pursuit may just bear fruit soon, since NASA has already released a detailed plan of how they are going to send humans to Mars in the coming decades. The plan involves sending humans to Mars and have them permanently reside in the planet. According to NASA, "Unlike Apollo, we will be going to stay".

TIMELINE

476

DAYS REMAINING

TAGS

DECADES · SCIENTISTS · GLOBE · DEDICATED · COUNTLESS · PURSUIT · MARS · MIGRATE · RELENTLESS · NASA · DETAILED · INVOLVES · PERMANENTLY · RESIDE · APOLLO · PLANET

$1447 SPENT · $2106 FORECASTED · $4953 REMAINING · $6400 BUDGET

3 / 4 RUNNING MACHINES

15 / 124 RUNNING CLUSTER NODES

$2.34 P/H COST RUNNING MACHINES

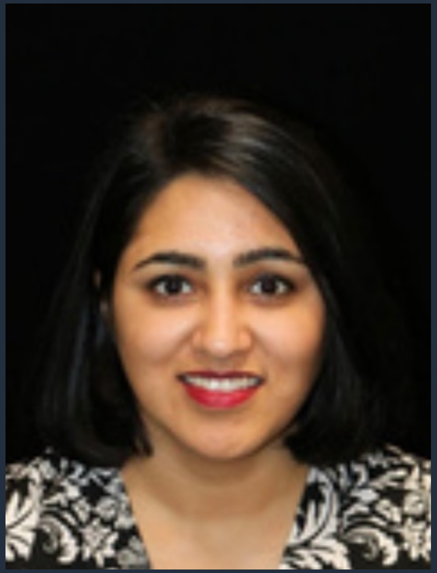
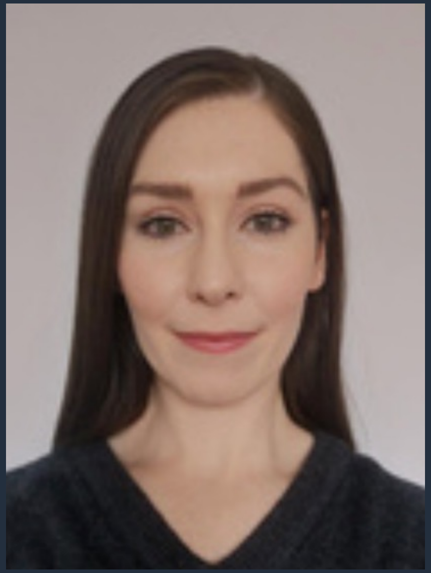
670 GB	0 GB	0 GB	0 GB	2475 GB
SSD STORAGE	HOT HDD STORAGE	COLD HDD STORAGE	MAGNETIC STORAGE	BACKUPS
$80.40 MONTH	$0.00 MONTH	$0.00 MONTH	$0.00 MONTH	

Thinking Beyond..

The Oxford X-Reality Hub Ed Tech project

Thank you!

We are here to help you on your cloud journey!

Apeksha Jain
Solutions Architect
jainapek@amazon.com

Courtney Waugh
Solutions Architect
cmwaugh@amazon.com

Matt Burridge
Account Manager
mburridg@amazon.com