Scale and operate microservices with AWS Proton

Mai Nishitani

Solutions Architect
Amazon Web Services

aws

Agenda

- Complexity of microservices

- AWS Proton overview

- Operator and developer experience

- What's next?

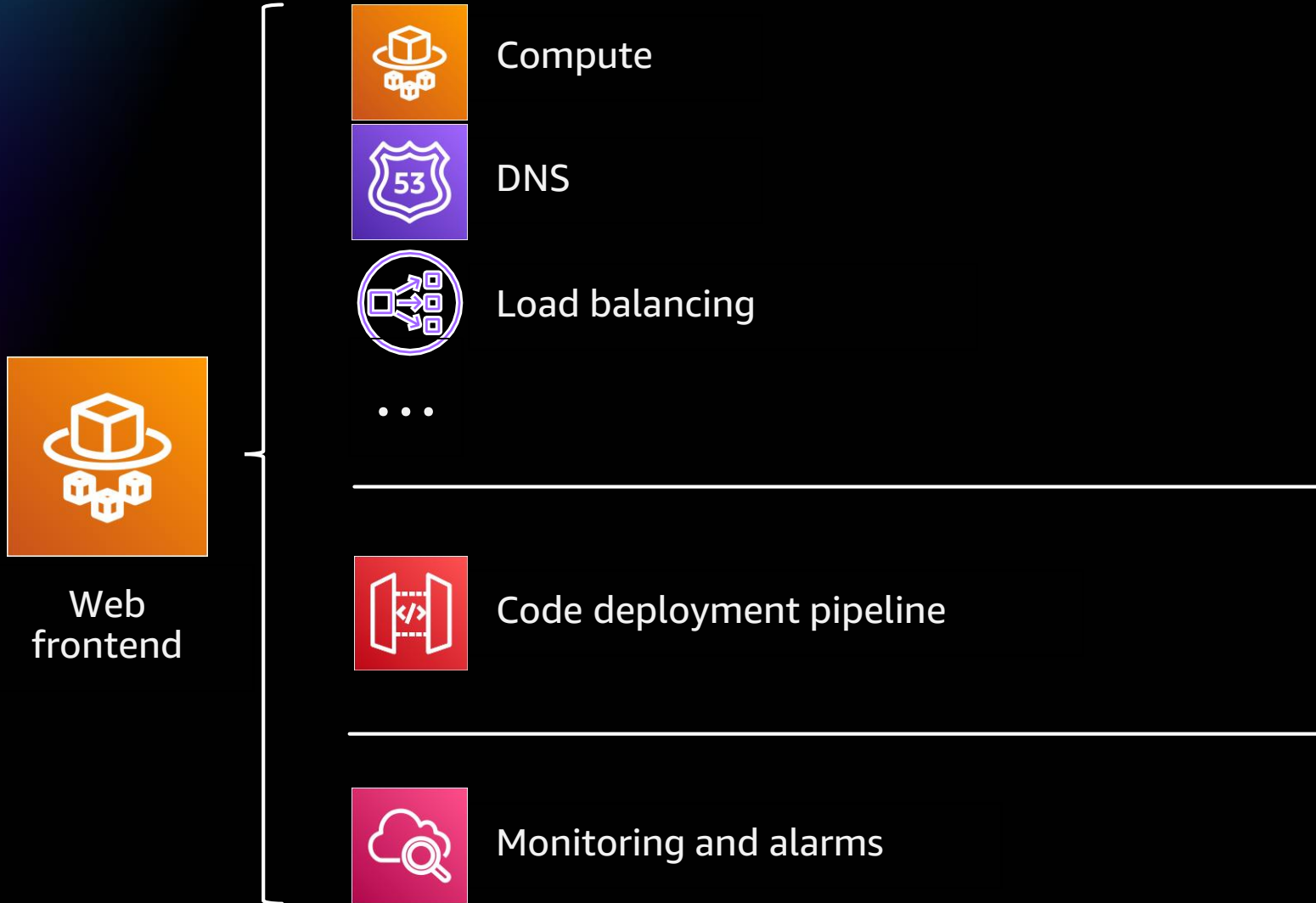# Why microservices ?
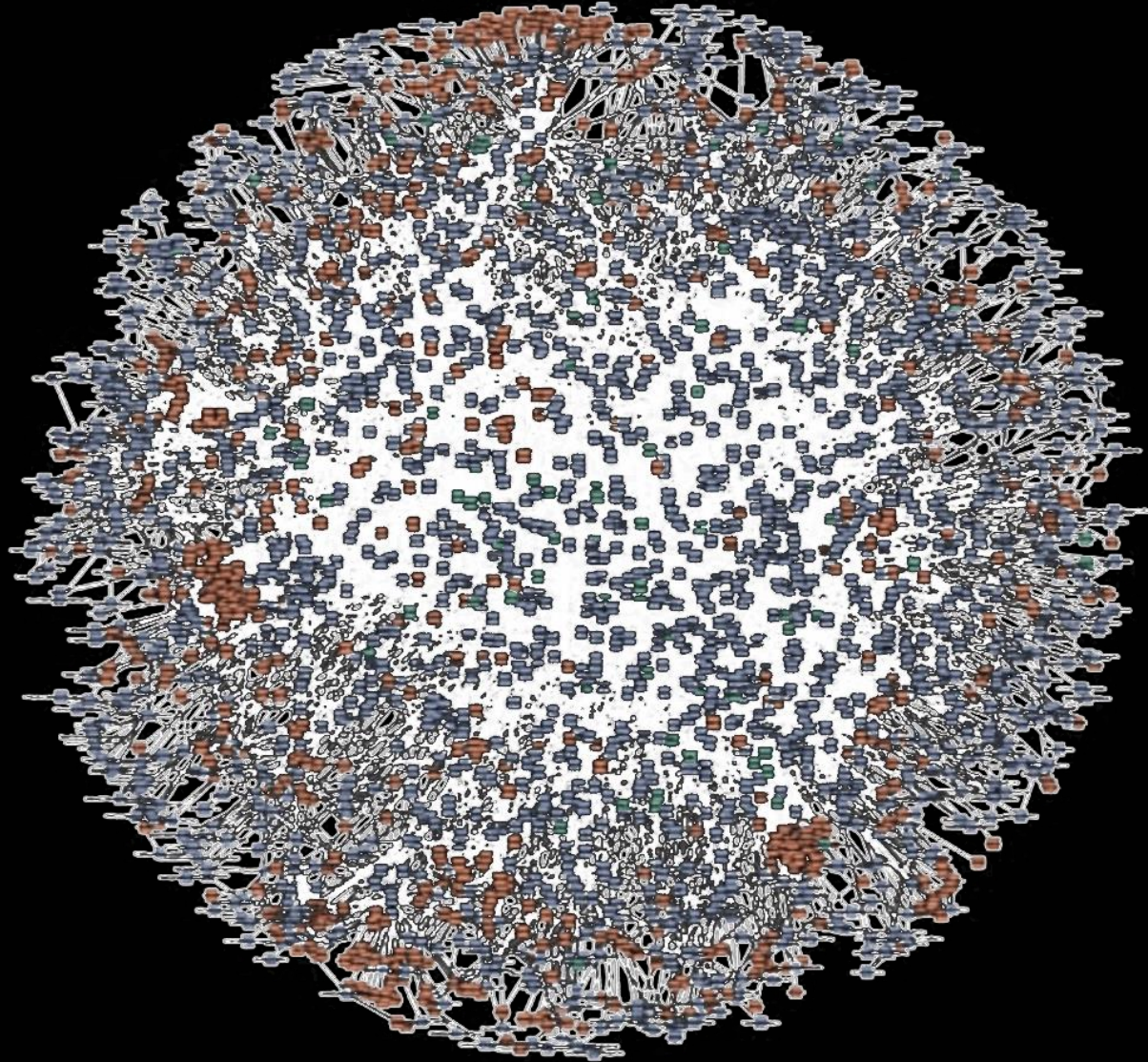
Agility

Scalability

Resilience

Technological freedom

Complexity of operating microservices

Web frontend

- Compute
- DNS
- Load balancing
- . . .

- Code deployment pipeline

- Monitoring and alarms

What happens when the architecture grows?

Customers are building internal developer platforms on AWS to tie it all together
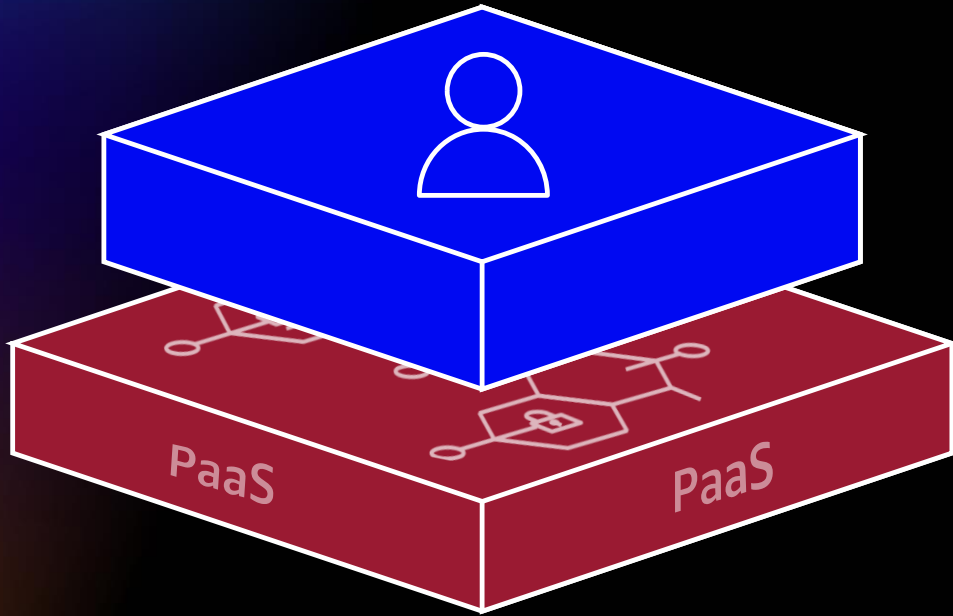
Intuit

Modern SaaS

Snapchat

Switchboard

Skyscanner

Slingshot

These platforms unlock innovation by increasing developer productivity and accelerating software delivery

Abstract away operational pain

Creating a developer platform is hard

Need to stand up a consistent stack to improve everyone's productivity

Also to provide better guardrails for protection

Managing and updating hundreds or thousands of deployed microservices is painful

Manage deployments with AWS Proton

- Deploy code pipelines with consistent standards and management

The first fully managed deployment service for
container and serverless applications

Infrastructure operators

Create application
infrastructure templates

AWS Proton

Monitor and update
deployments

Development teams

Find and deploy
application infrastructure

Core features – AWS Proton

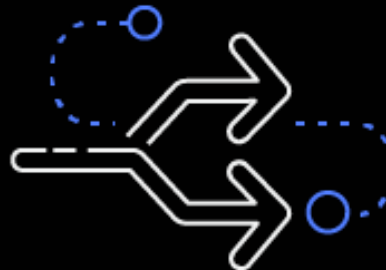Self-service interface
for developers

Centralized management
and visibility

Version control for
templates

One-click upgrades
to new versions

Infrastructure
pipelines

Third party
integrations

AWS Proton concepts

Environments

Shared resources in which services can run
Examples: a VPC, a cluster

Services

A workload running in an environment
Examples: an AWS Fargate web service, an
AWS Lambda data processor

Templates

Define a complete set of infrastructure resources, CI/CD
pipeline for environment and service

Anatomy of an AWS Proton template

AWS Proton schema

Parametrised infrastructure as code templates

Parametrised CI/CD template

Manifest

AWS Proton sample templates

Well-Architected

Built using AWS
Well-Architected best practices

Production ready

Templates can be used to deploy
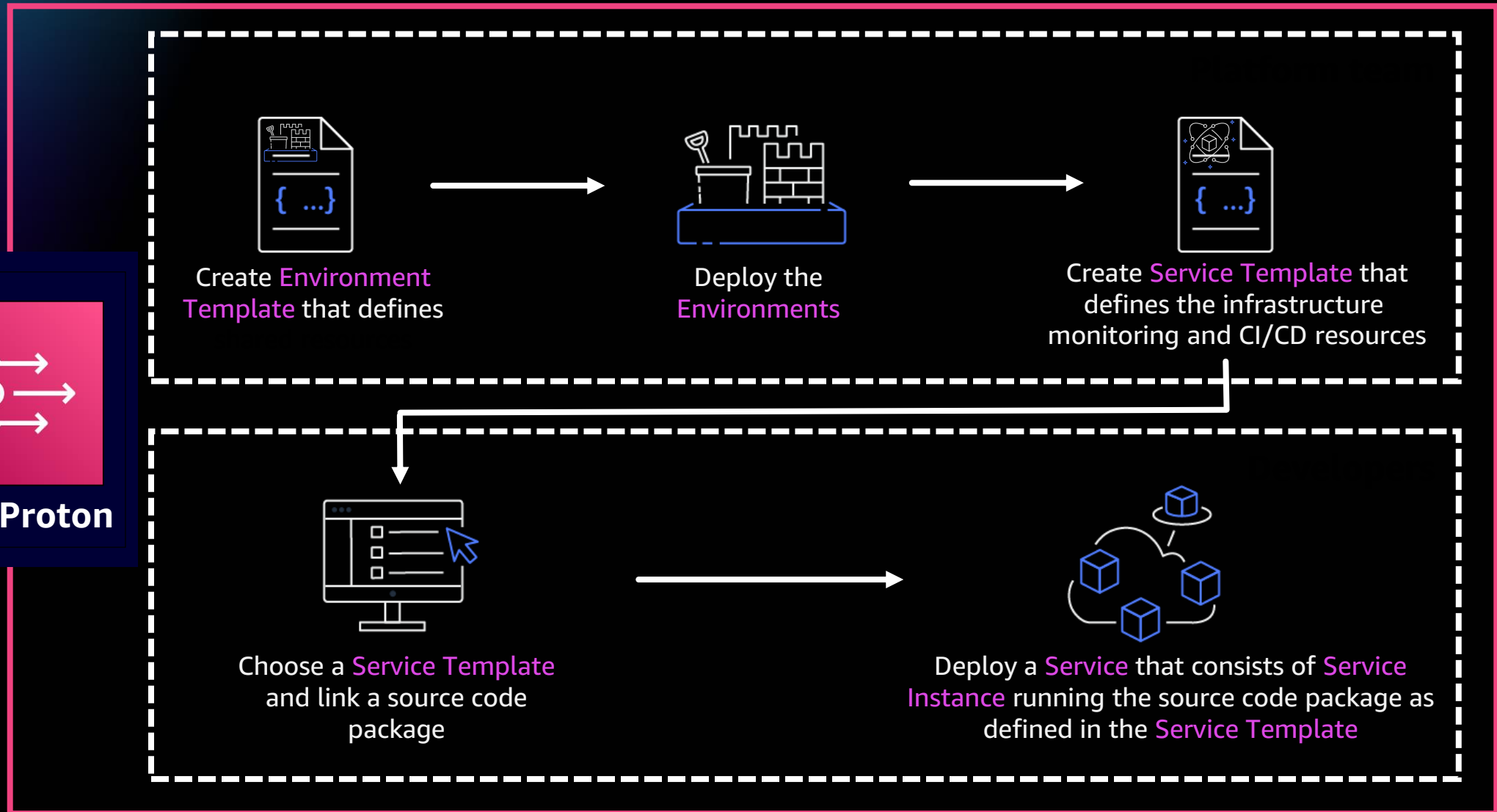production-grade infrastructure

Open source

Customers can customise
templates to suit their needs

View on GitHub - https://github.com/aws-samples/aws-proton-sample-templates

Working with AWS Proton

AWS Proton

Create Environment Template that defines

Deploy the Environments

Create Service Template that defines the infrastructure monitoring and CI/CD resources

Choose a Service Template and link a source code package

Deploy a Service that consists of Service Instance running the source code package as defined in the Service Template

Operator experience – Demo

Developer experience – Demo

aws

Recap

Platform operator

- Deployed best practice environment and service templates

- Updated environment and service templates as a major version

- Created service roles for AWS Proton

- Connected GitHub repository

- Deployed an environment including a CI/CD pipeline

Developer

- Deployed service instance based on best practice templates

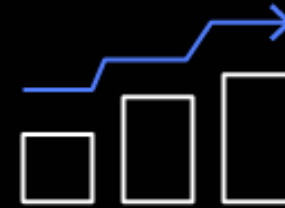- Updated code on GitHub which auto-updated the web app

Benefits of AWS Proton

Set guardrails

Enforce best practices

Increase developer productivity

Product features – launched

- Deploy in multi-account environments

- PrivateLink support based on templates

- API Support for boto3

- Limit permissions via IAM condition keys

Visit our AWS Proton documentation
https://docs.aws.amazon.com/proton/index.html

Product features – roadmap

- Support for third-party tooling

  - Infrastructure as code, CI/CD, observability

- Arbitary resources for service creation

- Self-service CI/CD integration

Visit our public roadmap on GitHub
https://github.com/aws/aws-proton-public-roadmap/projects/1

What's next ?

✓ Get started https://aws.amazon.com/proton/

✓ Checkout out the sample templates on GitHub
https://github.com/aws-samples/aws-proton-sample-templates

✓ Read the AWS Proton Blog
https://aws.amazon.com/blogs/aws/preview-aws-proton-automated-management-for-container-and-serverless-deployments/

aws

Visit the Modern Applications Resource Hub for more resources

Dive deeper with these resources to help you develop an effective plan for your modernization journey.

- Build modern applications on AWS e-book
- Build mobile and web apps faster e-book
- Modernize today with containers on AWS e-book
- Adopting a modern Dev+Ops model e-book
- Modern apps need modern ops e-book
- Determining the total cost of ownership: Comparing Serverless and Server-based technologies paper
- Continuous learning, continuous modernization e-book
- … and more!

https://bit.ly/3yfOvbK

Visit resource hub »

aws

AWS Training and Certification
Accelerate modernization with continuous learning

Free digital courses, including:
Architecting serverless solutions
Getting started with DevOps on AWS

Earn an industry-recognized credential:
AWS Certified Developer – Associate
AWS Certified DevOps – Professional

Hands-on classroom training
(available virtually) including:
Running containers on Amazon Elastic
Kubernetes Service (Amazon EKS)
Advanced developing on AWS

Create a self-paced learning roadmap
AWS ramp-up guide - Developer
AWS ramp-up guide - DevOps

Take Developer
and DevOps training
today

Learn more about
Modernization training for you
and your team

Thank you for attending AWS Innovate Modern Applications Edition

We hope you found it interesting! A kind reminder to **complete the survey.**
Let us know what you thought of today's event and how we can improve the event experience for you in the future.

📦 aws-apj-marketing@amazon.com

🐦 twitter.com/AWSCloud

f facebook.com/AmazonWebServices

▶ youtube.com/user/AmazonWebServices

slideshare.net/AmazonWebServices

twitch.tv/aws

aws

Thank you!