AWS re:Invent

DOP326

Deploy your code, scale your application, and lower cloud costs using Elastic Beanstalk

Prashant Prahlad

Sr. Manager

Amazon Web Services

Sekhar Kutikuppala

Sr. Product Manager

Amazon Web Services

Productivity

Getting more results with less time and effort

Developer productivity

Harder to define and measure, but easier to know

- Quality of work is often more important
- New technologies or languages rarely change productivity levels
- There is no “one” trick to boost productivity

From Peopleware: - Productive Projects and Teams

Developer productivity: Common scenario

Deliver projects/applications:

- Under severe time pressure
- With changing requirements
- Minimal staffing
- Low operational costs

Knowing what tools are available, and how to use them effectively is P1

What you'll get out of this session

- AWS Elastic Beanstalk: a set of tools in your productivity toolkit
- [DEMO] Migrate your application + Try it at home kit
- [DEMO] Using Docker containers on Elastic Beanstalk
- Advanced use cases: CI/CD, Blue/Green
- [EXCLUSIVE]: Elastic Beanstalk feature launches NEW
- [DEMO] Machine Learning on Elastic Beanstalk

Elastic Beanstalk: Common use cases

- Set up infrastructure for your web application
- Create your application on a broad set of curated platforms
- Scale infrastructure for your app based on demand
- Automatically apply patches and security fixes
- Load-balance traffic for higher availability
- Monitor the health of your application

… at no additional cost

Not so common use cases

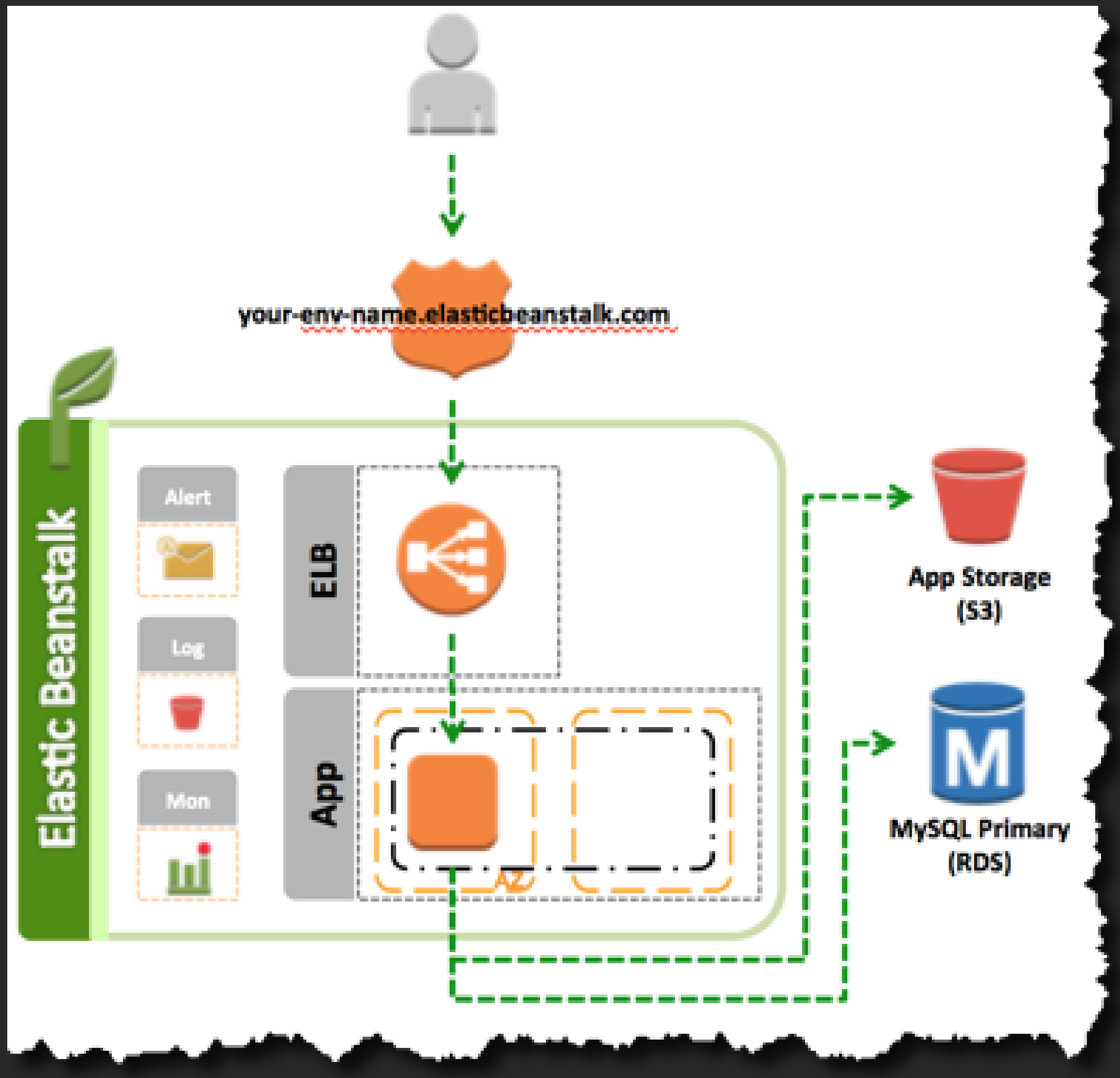
- Migrate your .NET application to managed infrastructure
- Modernize your application with multi-container Docker environments
- Serve and scale your machine learning model
- CI/CD for any application updates
- Blue/Green deployments
- Create your own development platform as a service

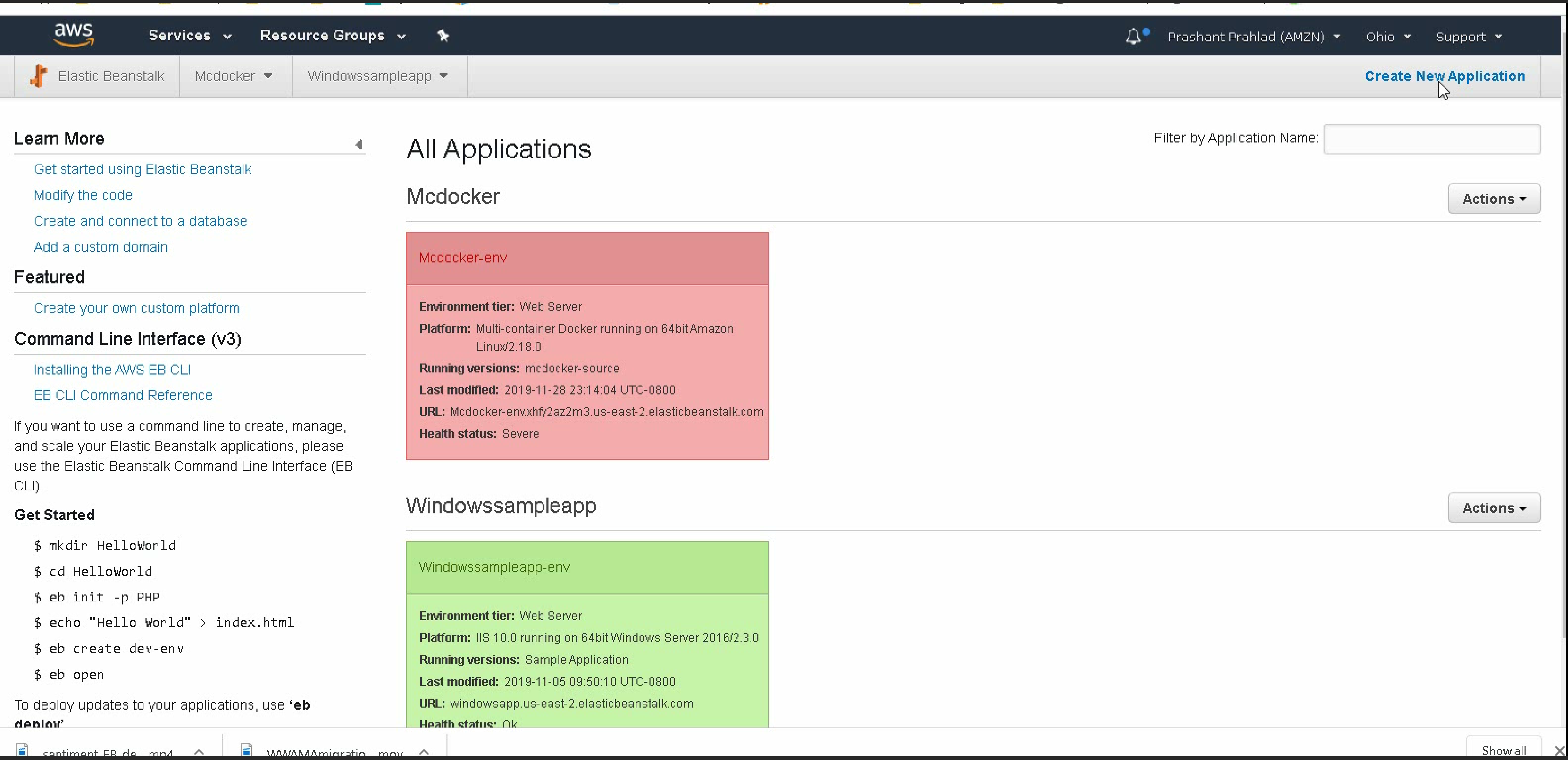
… and much more!

What is AWS Elastic Beanstalk?

Easy to begin, impossible to outgrow

- Upload your application
- Create resources required to launch the app
- Deploy updates





Learn More

- Get started using Elastic Beanstalk
- Modify the code
- Create and connect to a database
- Add a custom domain

Featured

- Create your own custom platform

Command Line Interface (v3)

- Installing the AWS EB CLI
- EB CLI Command Reference

If you want to use a command line to create, manage, and scale your Elastic Beanstalk applications, please use the Elastic Beanstalk Command Line Interface (EB CLI).

Get Started

```
$ mkdir HelloWorld
$ cd HelloWorld
$ eb init -p PHP
$ echo "Hello World" > index.html
$ eb create dev-env
$ eb open
```

To deploy updates to your applications, use **'eb deploy'**

All Applications

Filter by Application Name:

Mcdocker

Mcdocker-env

Environment tier: Web Server
Platform: Multi-container Docker running on 64bit Amazon Linux/2.18.0
Running versions: mcdocker-source
Last modified: 2019-11-28 23:14:04 UTC-0800
URL: Mcdocker-env.xhfy2az2m3.us-east-2.elasticbeanstalk.com
Health status: Severe

Windowssampleapp

Windowssampleapp-env

Environment tier: Web Server
Platform: IIS 10.0 running on 64bit Windows Server 2016/2.3.0
Running versions: Sample Application
Last modified: 2019-11-05 09:50:10 UTC-0800
URL: windowsapp.us-east-2.elasticbeanstalk.com
Health status: Ok

Too cool for console?

01

Create your Elastic Beanstalk app

```
$ eb init
```

02

Create the resources and launch the application

```
$ eb create
```

03

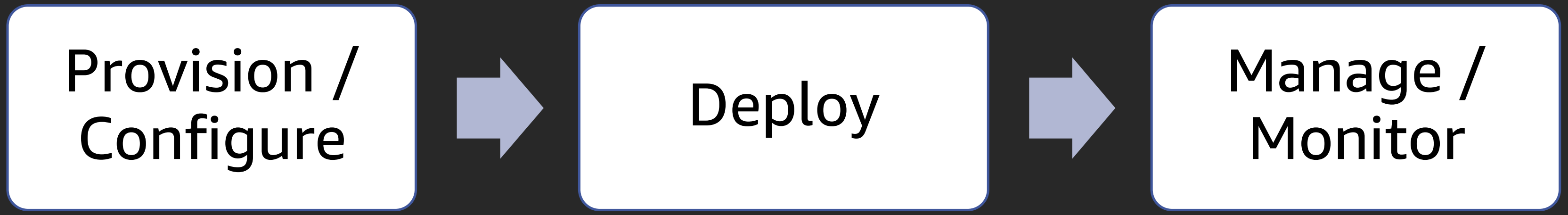
Deploy updates

```
$ eb deploy
```

What is AWS Elastic Beanstalk?

Automatically handles:

- Infrastructure provisioning and configuration
- Ongoing management of infrastructure
- Deployment
- Load balancing
- Auto scaling
- Health monitoring
- Automated Updates
- Analysis and debugging
- Logging



There is no additional charge for Elastic Beanstalk

What is AWS Elastic Beanstalk?

A set of tools:

- Infrastructure provisioning and configuration
- Ongoing management of infrastructure
- Deployment
- Load balancing
- Auto scaling
- Health monitoring
- Automated Updates
- Analysis and debugging
- Logging

There is no additional charge for Elastic Beanstalk

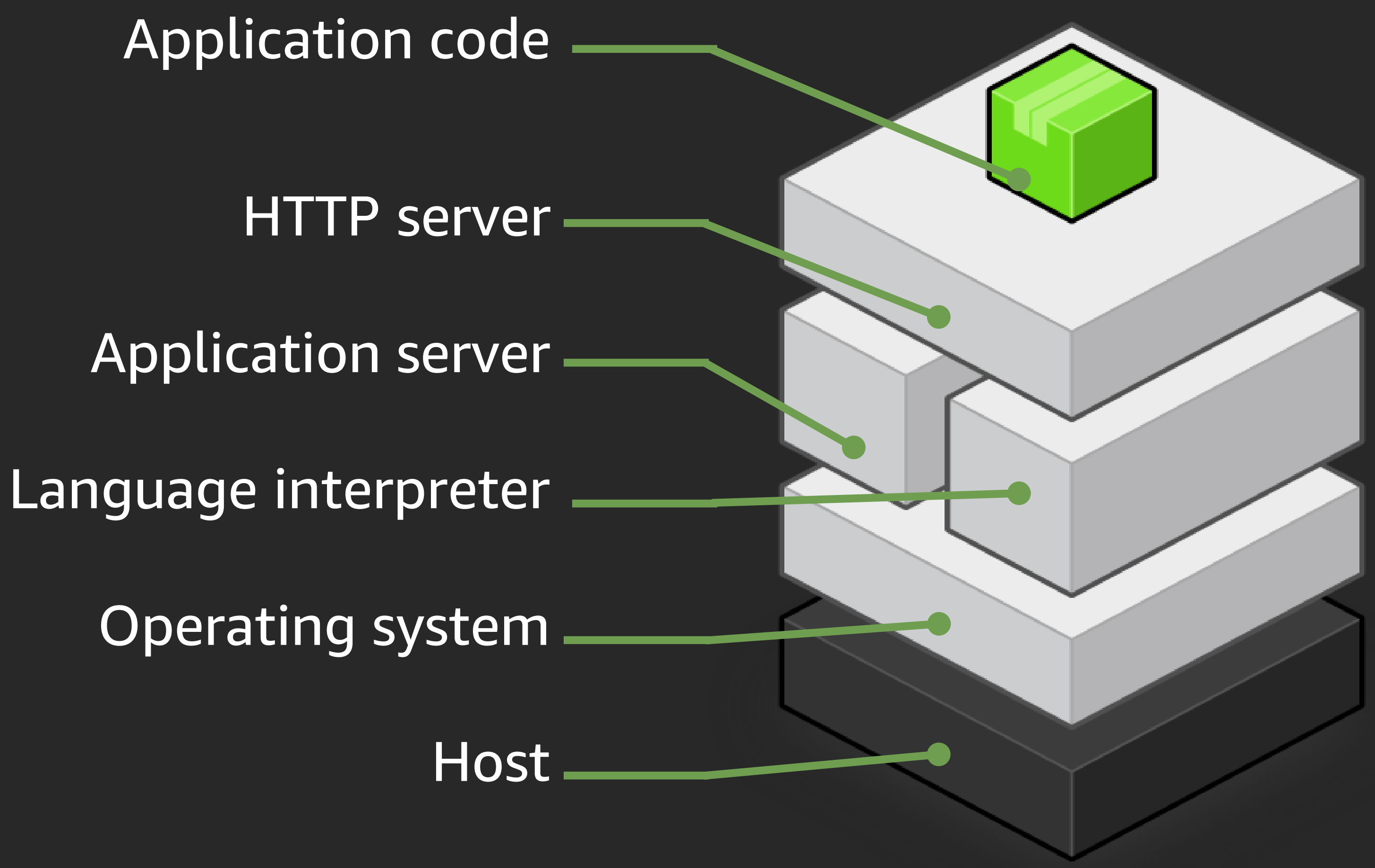
Terms you need to know

1. Application: “Container” that encapsulates code, infra, etc.
2. Application version: Code that you create and deploy
3. Platform: Language runtimes to run the application
4. Environment: Group of AWS infrastructure elements to support running the application

What does Elastic Beanstalk do?

Elastic Beanstalk:

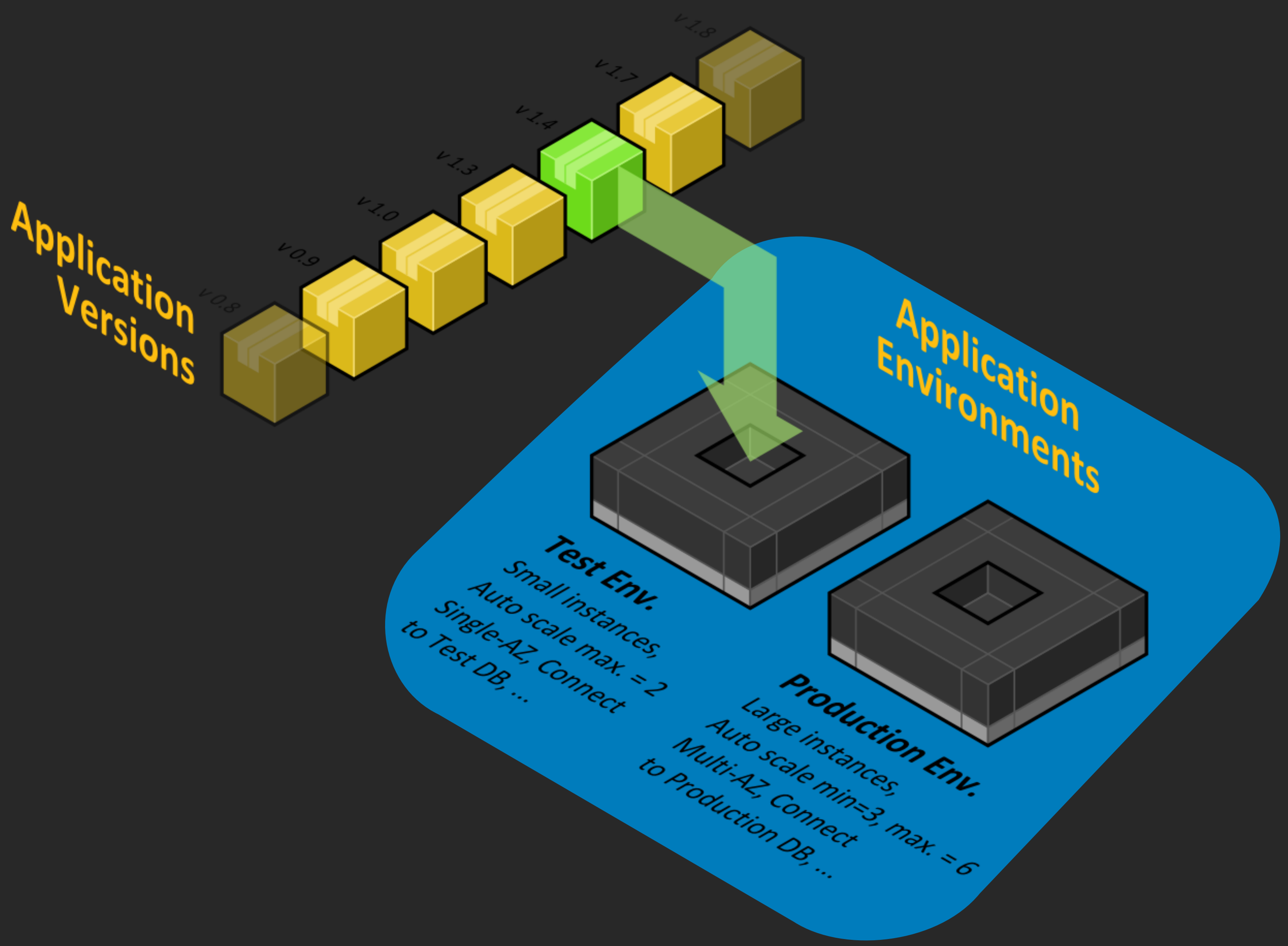
- Creates your application with the right runtime and platform components
- Ensures all of the components are working seamlessly together



AWS Elastic Beanstalk

Application versions

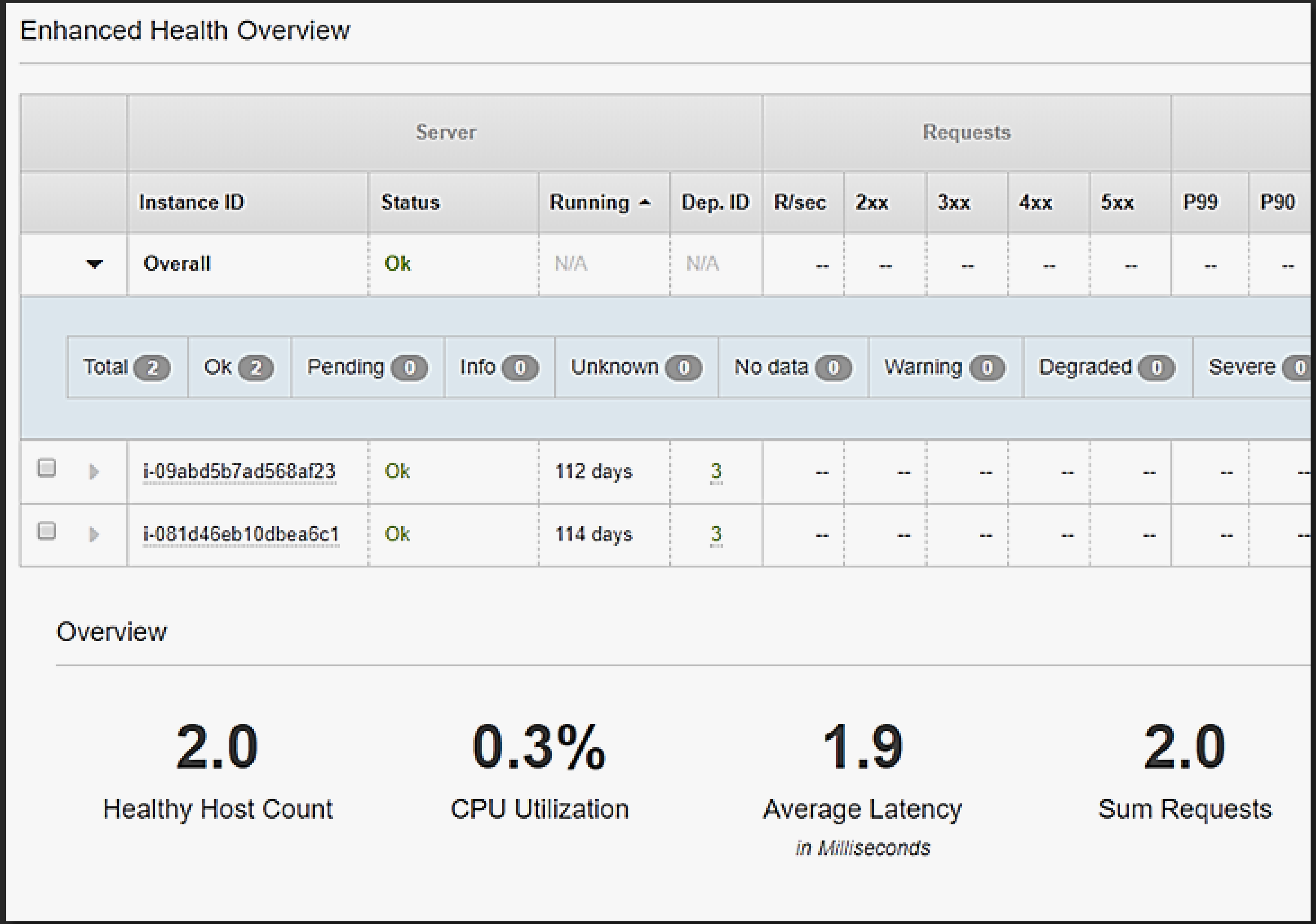
- Every deployment is its own unique version
- All versions stored in Amazon S3
- Allows easy rollback to a previous version without having to back out committed code changes
- Allows you to promote versions across different environments within an application



AWS Elastic Beanstalk

Health Monitoring

- Real-time health monitoring including load balancer checks, resource metrics, OS metrics, and web server logs
- Easily enable AWS X-Ray for additional analysis and debugging
- Visible through:
  - Console enhanced health overview
  - CloudWatch Logs streaming
  - Elastic Beanstalk CLI and API for easy integration into existing workflows



Enhanced Health Overview

		Server				Requests						
		Instance ID	Status	Running ▲	Dep. ID	R/sec	2xx	3xx	4xx	5xx	P99	P90
	▼	Overall	Ok	N/A	N/A	--	--	--	--	--	--	--
☐	▸	i-09abd5b7ad568af23	Ok	112 days	3	--	--	--	--	--	--	--
☐	▸	i-081d46eb10dbea6c1	Ok	114 days	3	--	--	--	--	--	--	--

Total 2 | Ok 2 | Pending 0 | Info 0 | Unknown 0 | No data 0 | Warning 0 | Degraded 0 | Severe 0

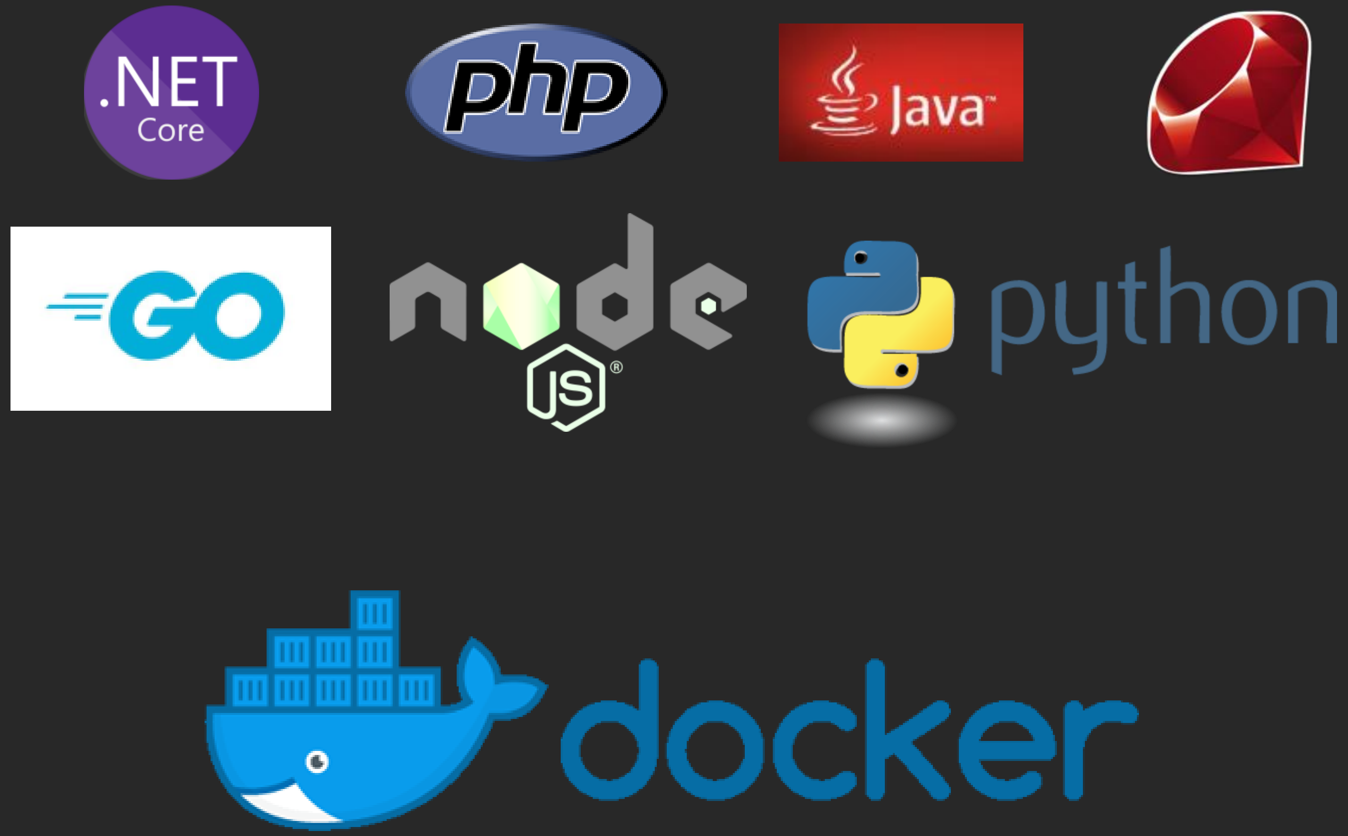
Overview

2.0	0.3%	1.9	2.0
Healthy Host Count	CPU Utilization	Average Latency *in Milliseconds*	Sum Requests

AWS Elastic Beanstalk supports

Platforms containing the most popular runtimes as well as Docker images, allowing you to bring your own

Elastic Beanstalk also supports custom platforms and custom images



Developer productivity: Task 1

Migrate a Windows .NET application running on IIS to AWS

Developer productivity: Task 1

Migrate a Windows .NET application running on IIS to AWS

- Create EC2 instances for Windows Server
- Set up IIS, .NET core and networking settings
- Migrate DB backend to RDS or other database services
- Deploy the web application to the new instance
- Operate the servers with system management tools

What is AWS Elastic Beanstalk?

A set of tools:

- Infrastructure provisioning and configuration
- Ongoing management of infrastructure
- Deployment
- Load balancing
- Auto scaling
- Health monitoring
- Automated Updates
- Analysis and debugging
- Logging

There is no additional charge for Elastic Beanstalk

What is AWS Elastic Beanstalk?

A set of tools:

- Infrastructure provisioning and configuration
- Ongoing management of infrastructure
- Deployment
- Load balancing
- Auto scaling
- Health monitoring
- Automated Updates
- Analysis and debugging
- Logging

There is no additional charge for Elastic Beanstalk

Developer productivity: Task 1

Migrate a Windows .NET application running on IIS to AWS

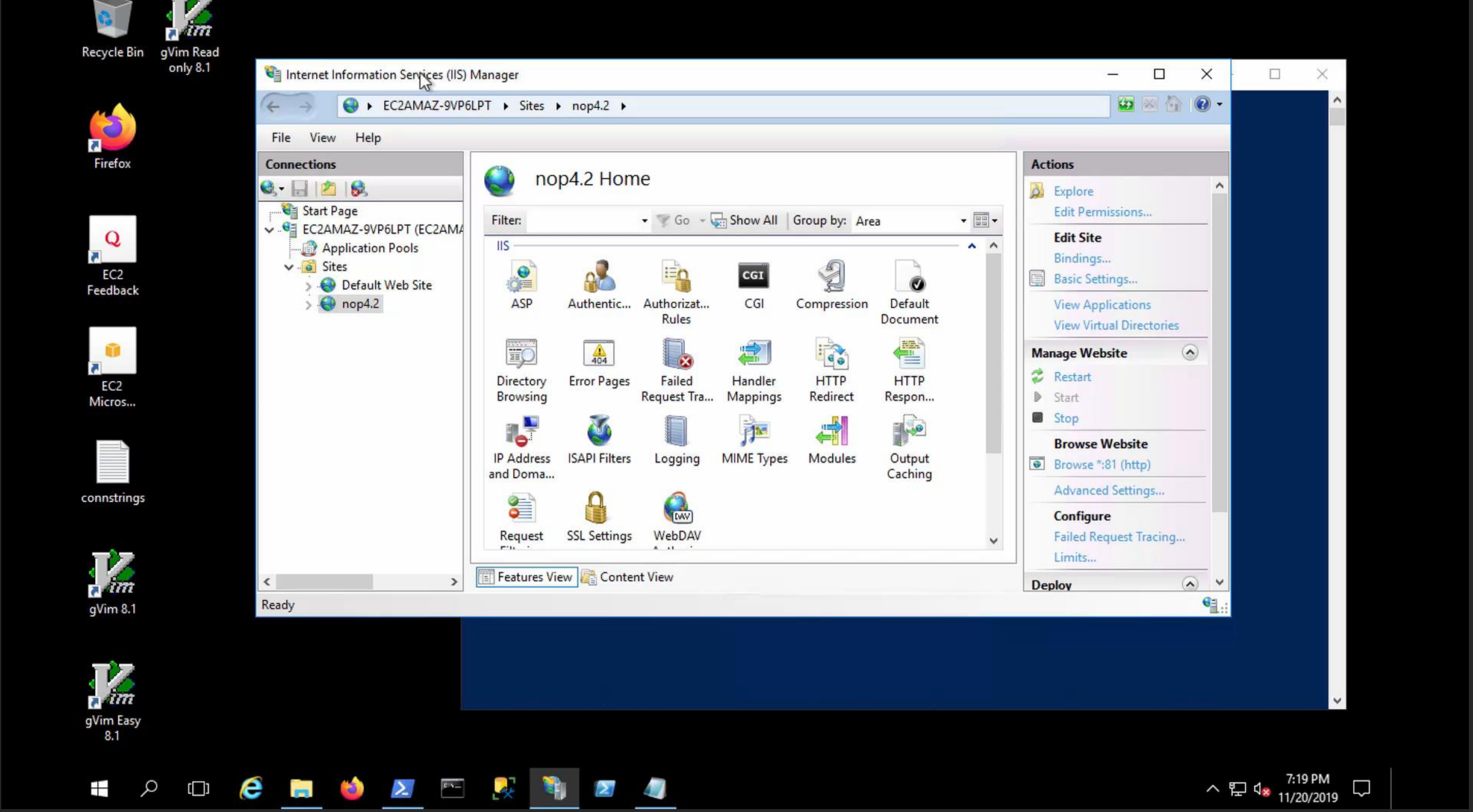
Toolkit:

- Elastic Beanstalk Environment with managed Windows .NET platform
- Migrate web application and associated SQL Server to AWS
- No infrastructure provisioning or ongoing management needed

EXCLUSIVE

But how do you actually perform the migration?

Launch: https://github.com/awslabs/windows-web-app-migration-assistant NEW


Recycle Bin
gVim Read only 8.1
Firefox
EC2 Feedback
EC2 Micros...
connstrings
gVim 8.1
gVim Easy 8.1
Internet Information Services (IIS) Manager
EC2AMAZ-9VP6LPT ▸ Sites ▸ nop4.2 ▸
File
View
Help
Connections
Start Page
EC2AMAZ-9VP6LPT (EC2AMA
Application Pools
Sites
Default Web Site
nop4.2
nop4.2 Home
Filter:
Go
Show All
Group by:
Area
IIS
ASP
Authentic...
Authorizat... Rules
CGI
Compression
Default Document
Directory Browsing
Error Pages
Failed Request Tra...
Handler Mappings
HTTP Redirect
HTTP Respon...
IP Address and Doma...
ISAPI Filters
Logging
MIME Types
Modules
Output Caching
Request Filt...
SSL Settings
WebDAV Auth...
Features View
Content View
Actions
Explore
Edit Permissions...
Edit Site
Bindings...
Basic Settings...
View Applications
View Virtual Directories
Manage Website
Restart
Start
Stop
Browse Website
Browse *:81 (http)
Advanced Settings...
Configure
Failed Request Tracing...
Limits...
Deploy
Ready
7:19 PM
11/20/2019

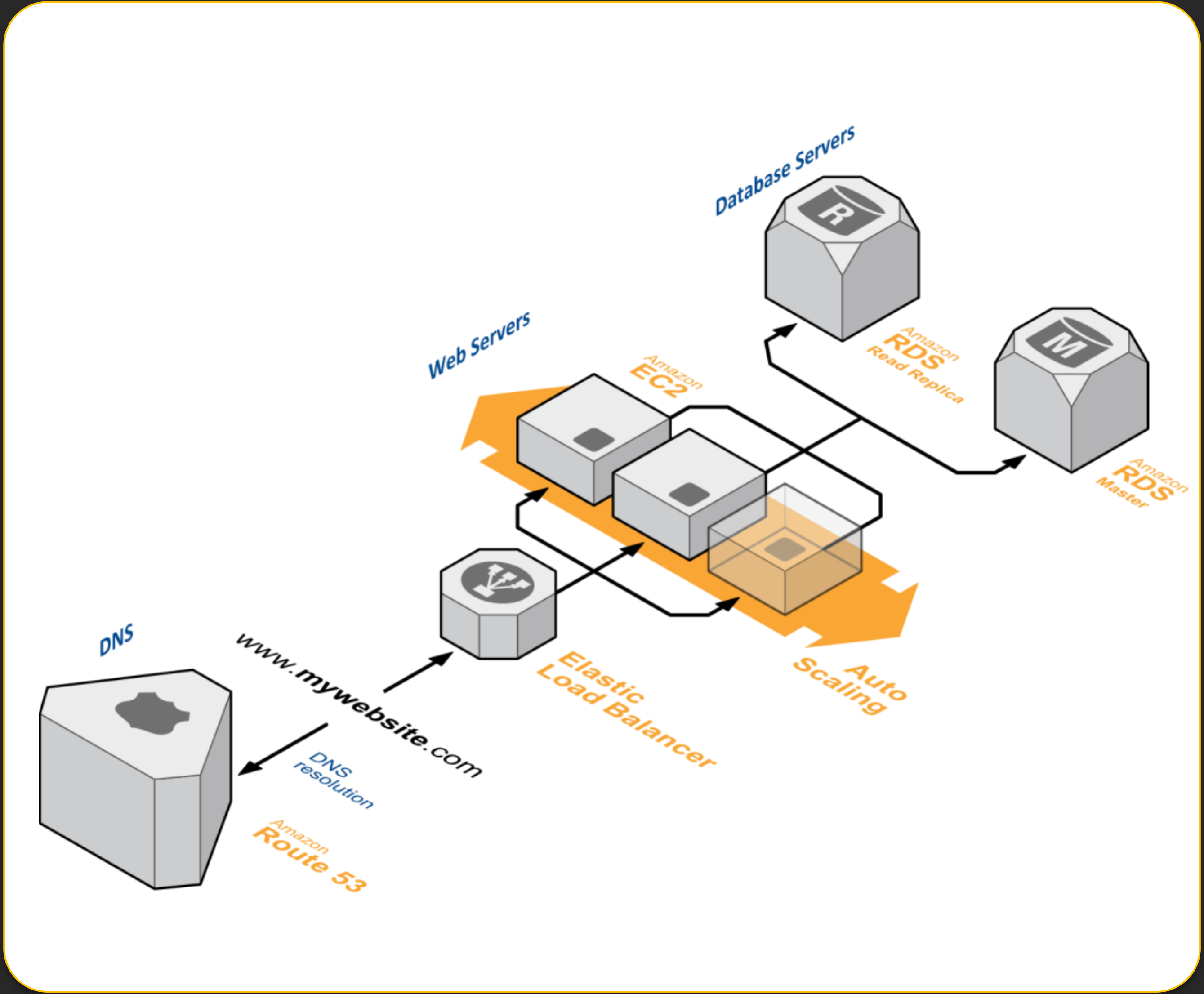
What just happened?

Migration Assistant

- Ran a set of PowerShell scripts to take a snapshot of the website, and connect it to a SQL server 2017 backend on Amazon EC2, and migrate the website to run on Beanstalk

Elastic Beanstalk

- Provisioned the necessary infrastructure resources including EC2 instances, security groups, load balancers, and auto scaling groups
- Configured Amazon Route 53 with a unique domain name that you can define



AWS Elastic Beanstalk for migrating workloads

Use Elastic Beanstalk as a landing zone for web workloads migrating to AWS. Offers:

- Automated provisioning and ongoing infrastructure management
- Patches and security fixes are automatically applied to the platform
- Application health is being monitored by Elastic Beanstalk
- Log streaming with appropriate log rotation policies
- Optional Elastic Load Balancer, Amazon RDS, and AWS Auto Scaling settings
- At no additional cost

Try it today – contribute and improve! Windows Web Application Migration Assistant

NEW

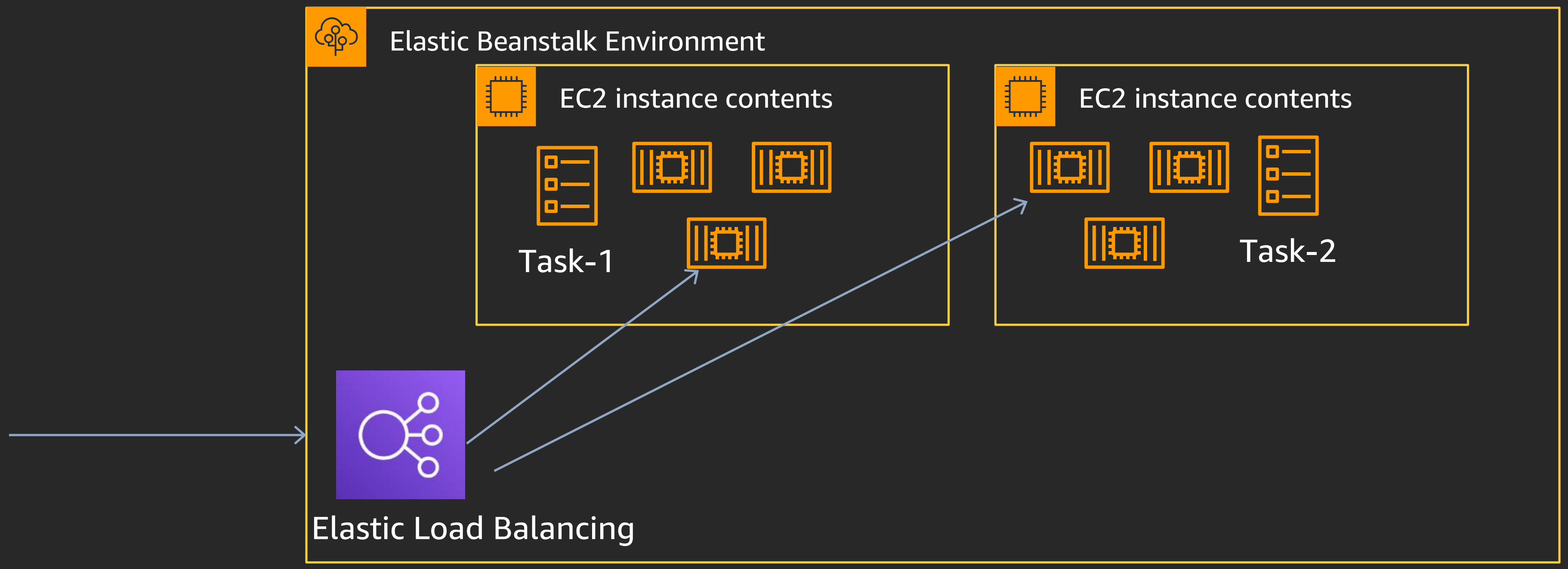
https://github.com/awslabs/windows-web-app-migration-assistant

Agenda

- Elastic Beanstalk: a set of tools in your productivity toolkit
- [DEMO] Migrate your application + Try it at home kit
- [DEMO] Using Docker containers on Elastic Beanstalk
- Advanced use cases: CI/CD, Blue/Green
- [EXCLUSIVE]: Elastic Beanstalk feature launches NEW
- [DEMO] Machine Learning on Beanstalk

Multi-container Docker support in Elastic Beanstalk

- Single Docker container: One container per environment
- Multi-container Docker runs multiple containers in one environment to realize cost savings



Developer productivity: Task 2

Build a PHP website with NGINX proxy and package as Docker containers

Toolkit:

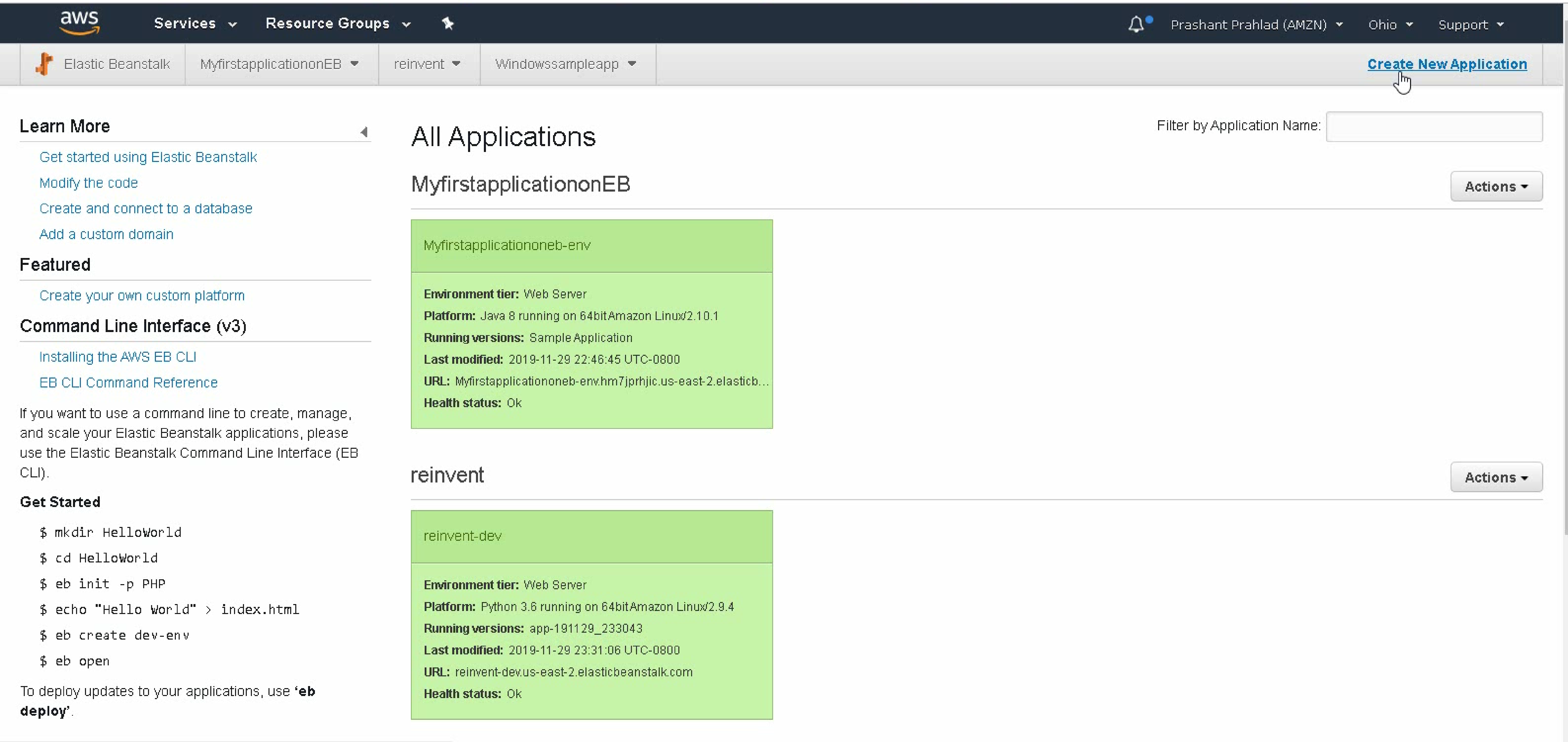
- Multi-container Docker on Elastic Beanstalk is implemented on Amazon Elastic Container Service (ECS)
- Automated provision and management, with load balancing, deployment options, and health monitoring

What is AWS Elastic Beanstalk?

A set of tools:

- Infrastructure provisioning and configuration
- Ongoing management of infrastructure
- Deployment
- Load balancing
- Auto scaling
- Health monitoring
- Automated Updates
- Analysis and debugging
- Logging

There is no additional charge for Elastic Beanstalk



Learn More

- Get started using Elastic Beanstalk
- Modify the code
- Create and connect to a database
- Add a custom domain

Featured

- Create your own custom platform

Command Line Interface (v3)

- Installing the AWS EB CLI
- EB CLI Command Reference

If you want to use a command line to create, manage, and scale your Elastic Beanstalk applications, please use the Elastic Beanstalk Command Line Interface (EB CLI).

Get Started

```
$ mkdir HelloWorld
$ cd HelloWorld
$ eb init -p PHP
$ echo "Hello World" > index.html
$ eb create dev-env
$ eb open
```

To deploy updates to your applications, use **'eb deploy'**.

All Applications

MyfirstapplicationonEB

Myfirstapplicationoneb-env

Environment tier: Web Server
Platform: Java 8 running on 64bit Amazon Linux/2.10.1
Running versions: Sample Application
Last modified: 2019-11-29 22:46:45 UTC-0800
URL: Myfirstapplicationoneb-env.hm7jprhjic.us-east-2.elasticb...
Health status: Ok

reinvent

reinvent-dev

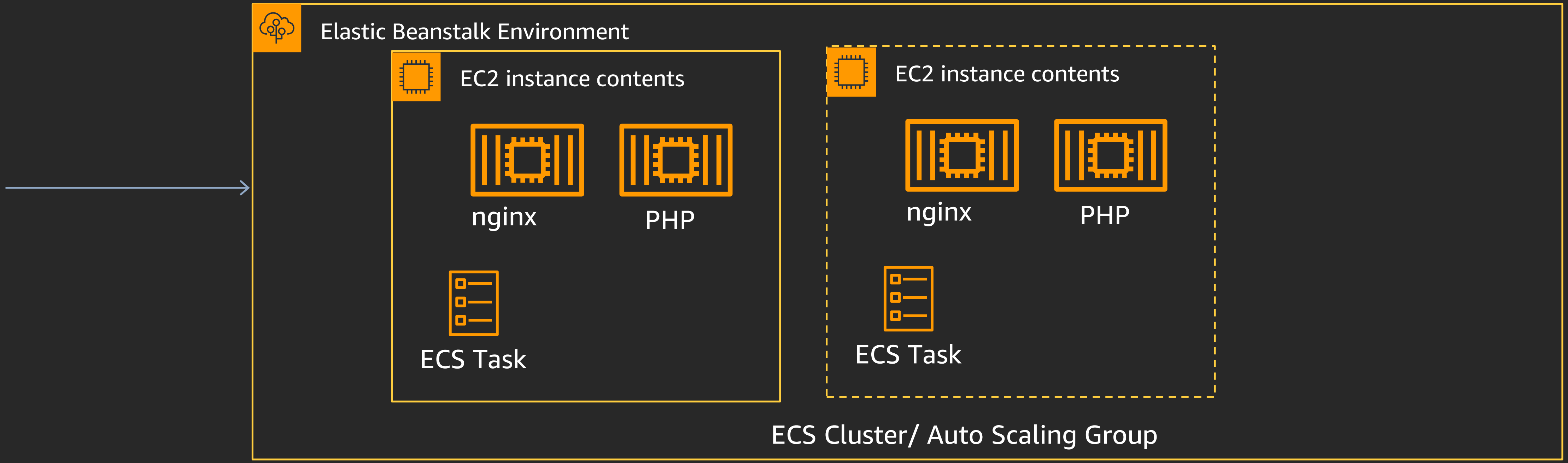
Environment tier: Web Server
Platform: Python 3.6 running on 64bit Amazon Linux/2.9.4
Running versions: app-191129_233043
Last modified: 2019-11-29 23:31:06 UTC-0800
URL: reinvent-dev.us-east-2.elasticbeanstalk.com
Health status: Ok

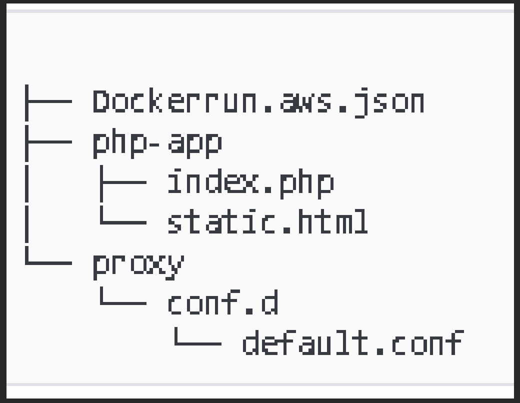
What just happened?

```
php-app/index.php

<h1>>Welcome to re:Invent 2019, now in PHP!!! </h1>

<h3>PHP Version <pre><?= phpversion()?></pre></h3>
```





```
├── Dockerrun.aws.json
├── php-app
│   ├── index.php
│   └── static.html
└── proxy
    └── conf.d
        └── default.conf
```

Go try it out:
https://github.com/aws-samples/eb-docker-nginx-proxy

But … it’s not just about getting started!

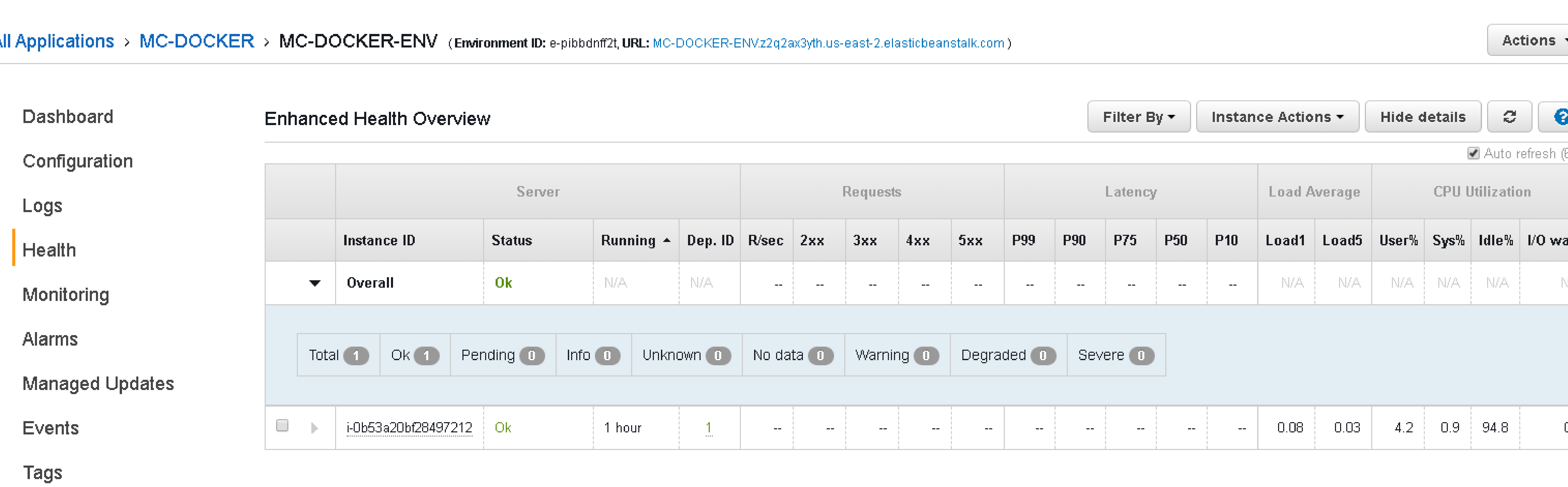
What is AWS Elastic Beanstalk?

A set of tools:

- Infrastructure provisioning and configuration
- Ongoing management of infrastructure
- Deployment
- Load balancing
- Auto scaling
- Health monitoring
- Automated Updates
- Analysis and debugging
- Logging

There is no additional charge for Elastic Beanstalk

Enhanced Health



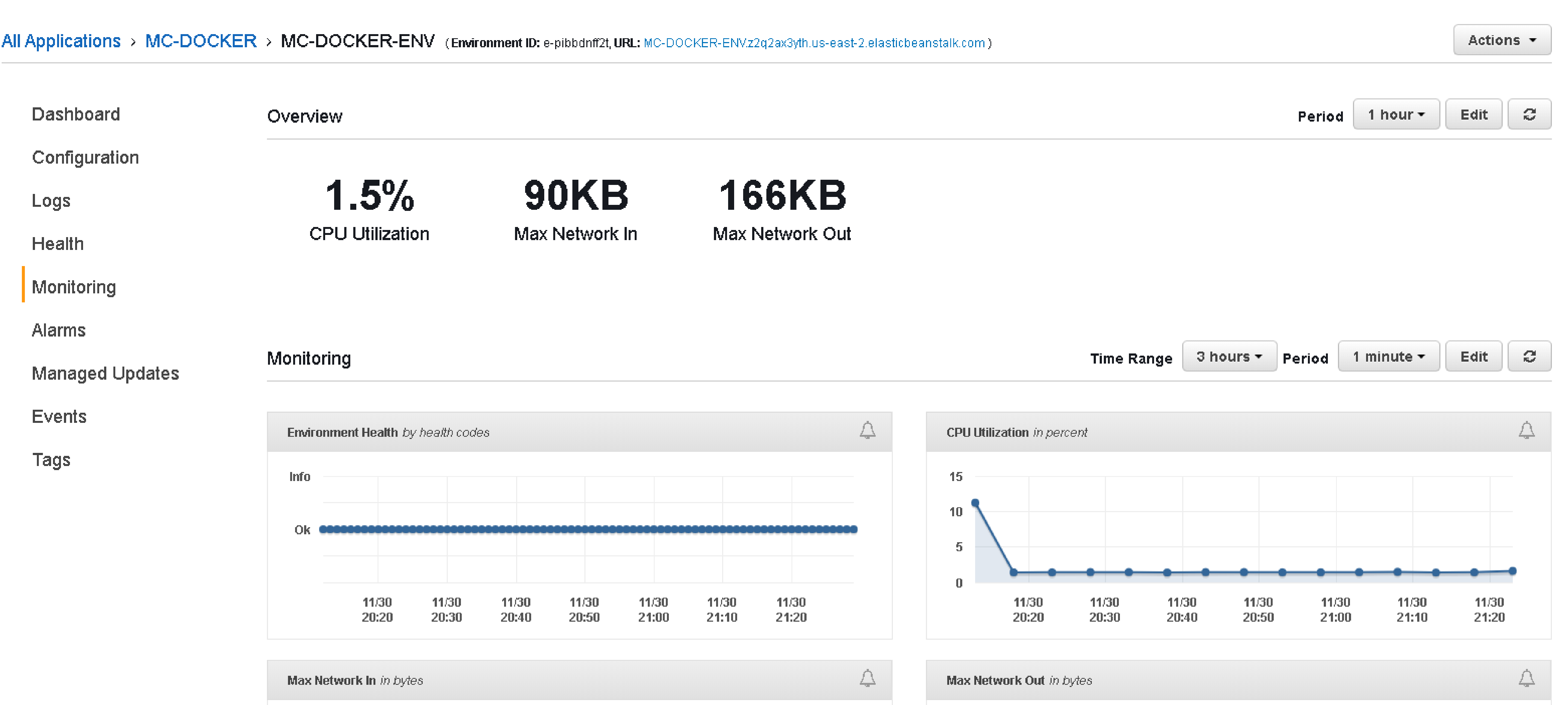
ll Applications › MC-DOCKER › MC-DOCKER-ENV (**Environment ID:** e-pibbdnff2t, **URL:** MC-DOCKER-ENV.z2q2ax3yth.us-east-2.elasticbeanstalk.com)

Enhanced Health Overview

	Server				Requests					Latency					Load Average		CPU Utilization			
	Instance ID	Status	Running ▲	Dep. ID	R/sec	2xx	3xx	4xx	5xx	P99	P90	P75	P50	P10	Load1	Load5	User%	Sys%	Idle%	I/O wa
▼	Overall	Ok	N/A	N/A	--	--	--	--	--	--	--	--	--	--	N/A	N/A	N/A	N/A	N/A	N
□ ▸	i-0b53a20bf28497212	Ok	1 hour	1	--	--	--	--	--	--	--	--	--	--	0.08	0.03	4.2	0.9	94.8	0

Total 1 | Ok 1 | Pending 0 | Info 0 | Unknown 0 | No data 0 | Warning 0 | Degraded 0 | Severe 0

Monitoring



All Applications › MC-DOCKER › MC-DOCKER-ENV (**Environment ID:** e-pibbdnff2t, **URL:** MC-DOCKER-ENV.z2q2ax3yth.us-east-2.elasticbeanstalk.com)

Overview

Period 1 hour ▾

1.5% CPU Utilization

90KB Max Network In

166KB Max Network Out

Monitoring

Time Range 3 hours ▾ **Period** 1 minute ▾

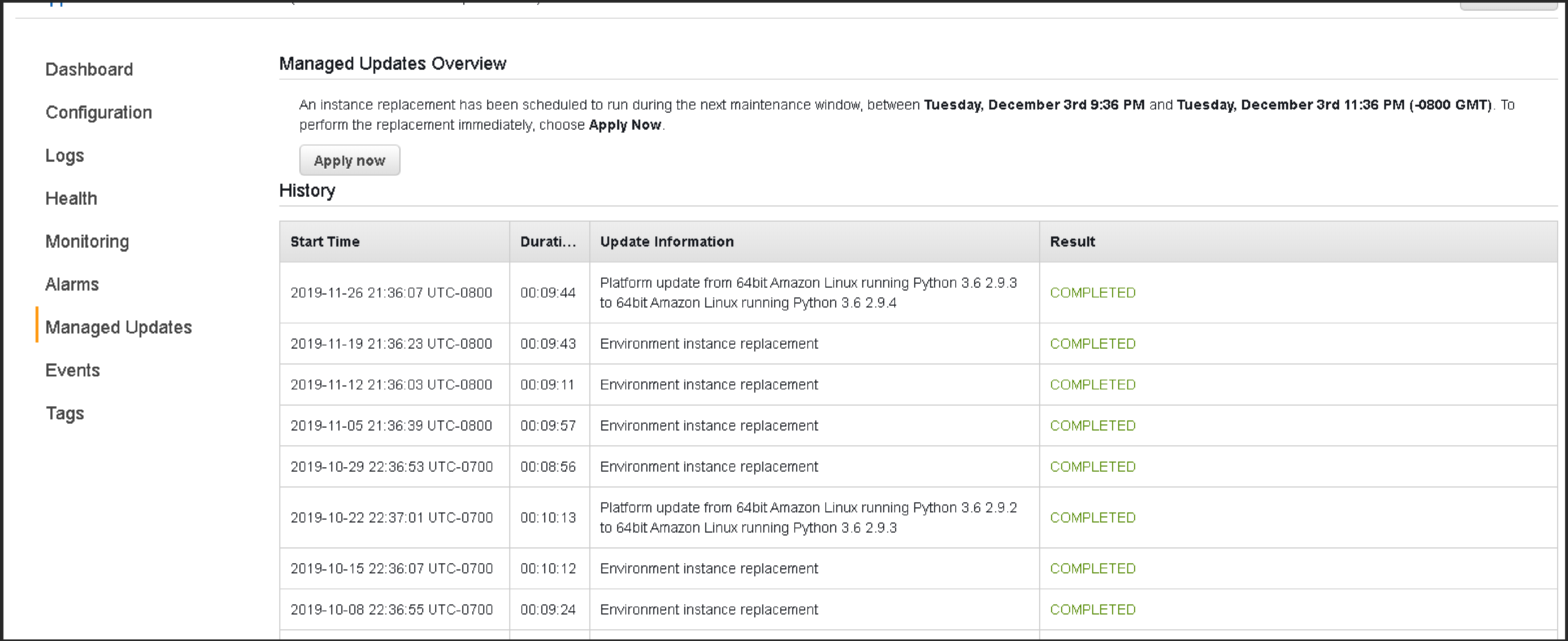
Environment Health *by health codes*

CPU Utilization *in percent*

Max Network In *in bytes*

Max Network Out *in bytes*

Automatic Updates



Managed Updates Overview

An instance replacement has been scheduled to run during the next maintenance window, between **Tuesday, December 3rd 9:36 PM** and **Tuesday, December 3rd 11:36 PM (-0800 GMT)**. To perform the replacement immediately, choose **Apply Now**.

History

Start Time	Durati...	Update Information	Result
2019-11-26 21:36:07 UTC-0800	00:09:44	Platform update from 64bit Amazon Linux running Python 3.6 2.9.3 to 64bit Amazon Linux running Python 3.6 2.9.4	COMPLETED
2019-11-19 21:36:23 UTC-0800	00:09:43	Environment instance replacement	COMPLETED
2019-11-12 21:36:03 UTC-0800	00:09:11	Environment instance replacement	COMPLETED
2019-11-05 21:36:39 UTC-0800	00:09:57	Environment instance replacement	COMPLETED
2019-10-29 22:36:53 UTC-0700	00:08:56	Environment instance replacement	COMPLETED
2019-10-22 22:37:01 UTC-0700	00:10:13	Platform update from 64bit Amazon Linux running Python 3.6 2.9.2 to 64bit Amazon Linux running Python 3.6 2.9.3	COMPLETED
2019-10-15 22:36:07 UTC-0700	00:10:12	Environment instance replacement	COMPLETED
2019-10-08 22:36:55 UTC-0700	00:09:24	Environment instance replacement	COMPLETED

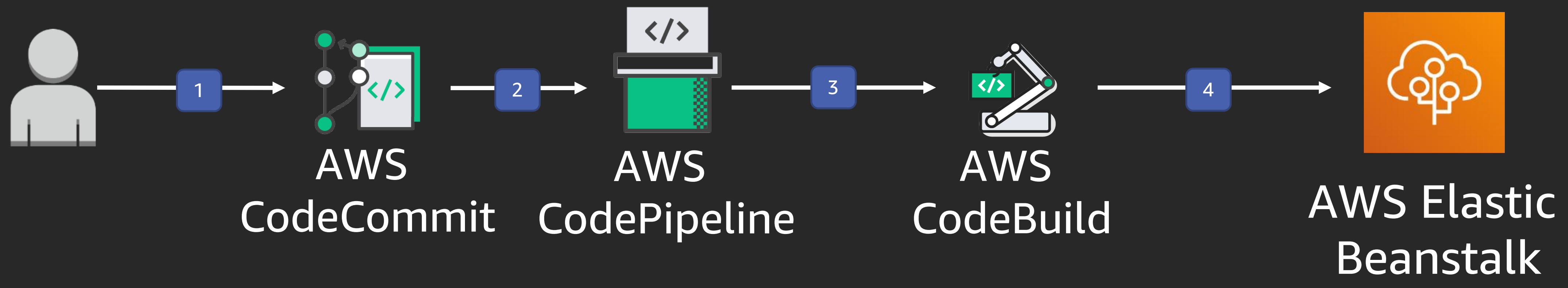
Advanced scenarios

AWS re:Invent

aws

Continuous integration and continuous deployment

How can I create robust but simple CI/CD across my release stages (dev, staging, production) with increasingly safer deployment status, deployment failure reporting, and ongoing health monitoring?



Extending to CI/CD – AWS developer tools

AWS CodeCommit

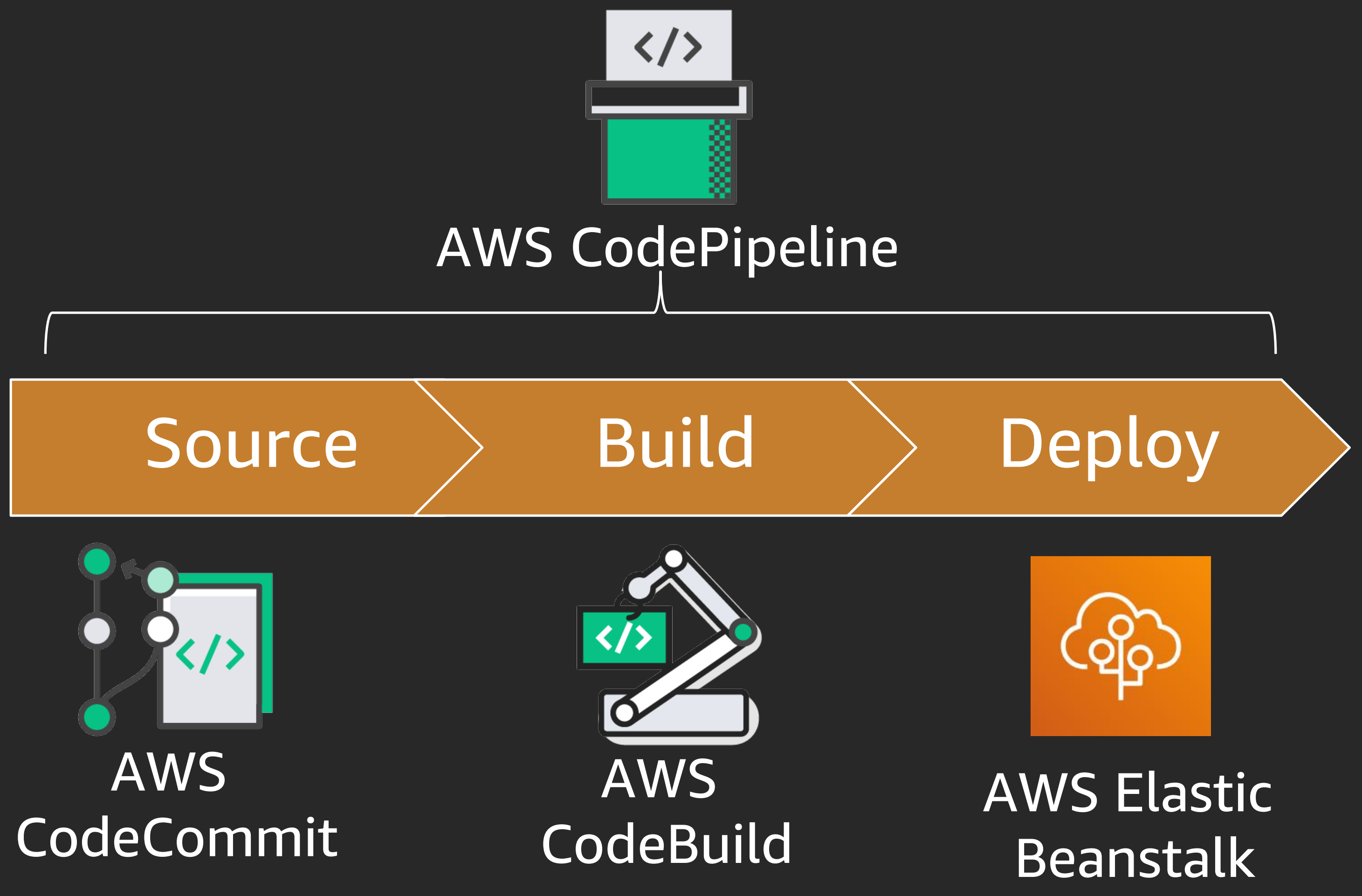
- Fully managed Git service

AWS CodeBuild

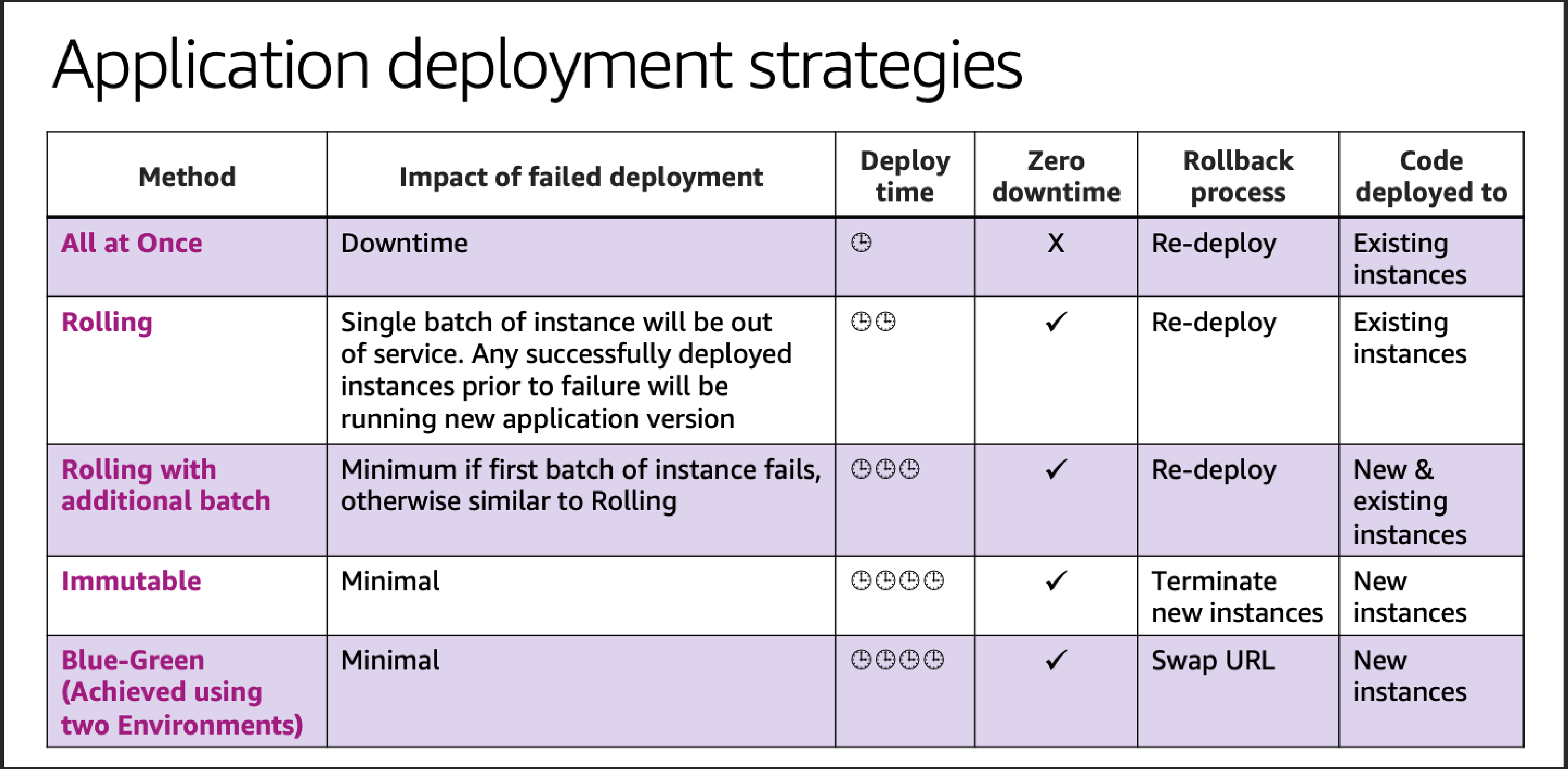
- Fully managed build service that compiles source code, runs tests, and produces software packages

AWS CodePipeline

- Continuous deployment service for fast and reliable application updates



Productivity booster: Deployment policies



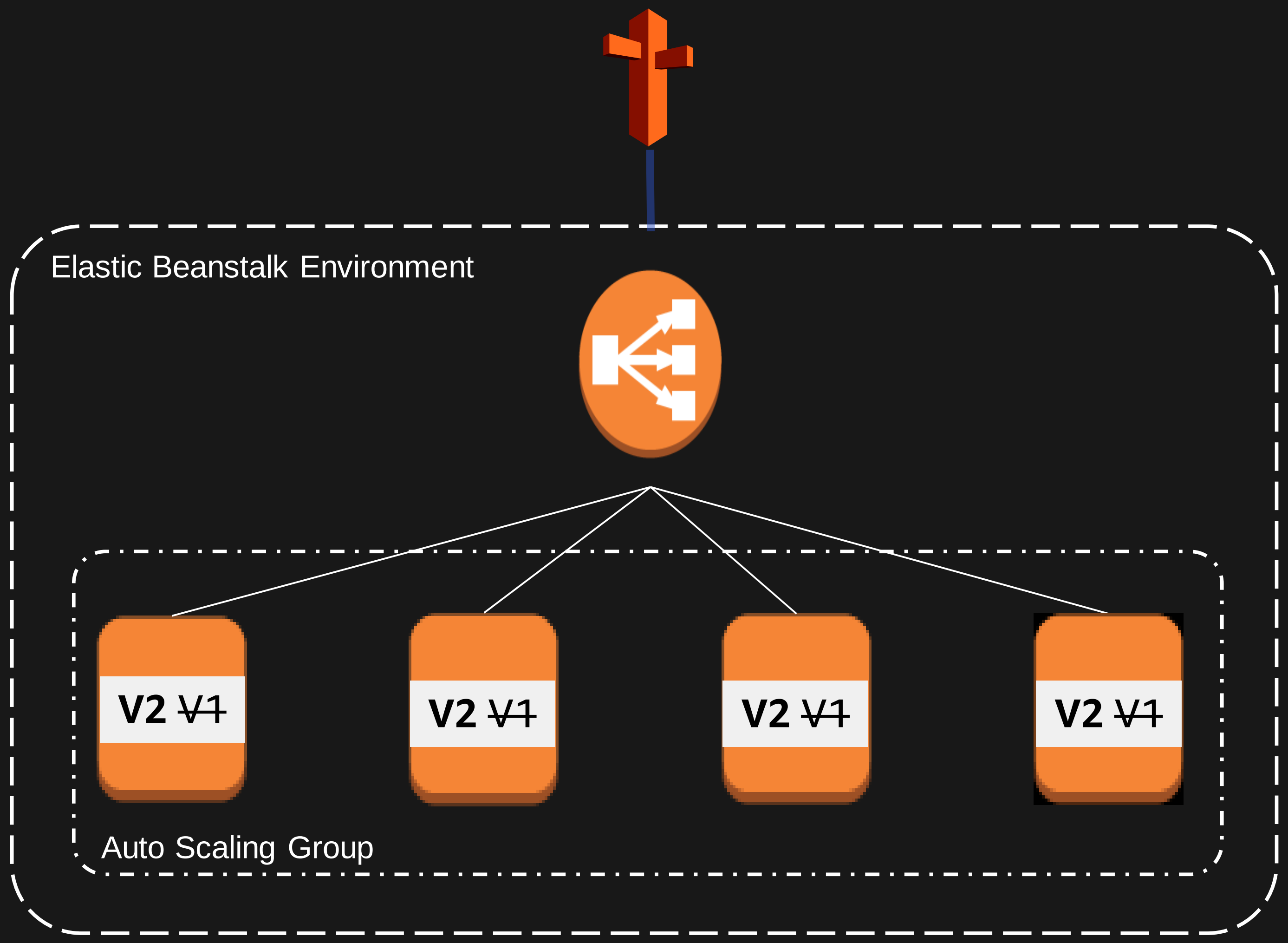
Application deployment strategies

Method	Impact of failed deployment	Deploy time	Zero downtime	Rollback process	Code deployed to
All at Once	Downtime	🕒	X	Re-deploy	Existing instances
Rolling	Single batch of instance will be out of service. Any successfully deployed instances prior to failure will be running new application version	🕒🕒	✓	Re-deploy	Existing instances
Rolling with additional batch	Minimum if first batch of instance fails, otherwise similar to Rolling	🕒🕒🕒	✓	Re-deploy	New & existing instances
Immutable	Minimal	🕒🕒🕒🕒	✓	Terminate new instances	New instances
Blue-Green (Achieved using two Environments)	Minimal	🕒🕒🕒🕒	✓	Swap URL	New instances

Easily enabled through the Elastic Beanstalk console or command line

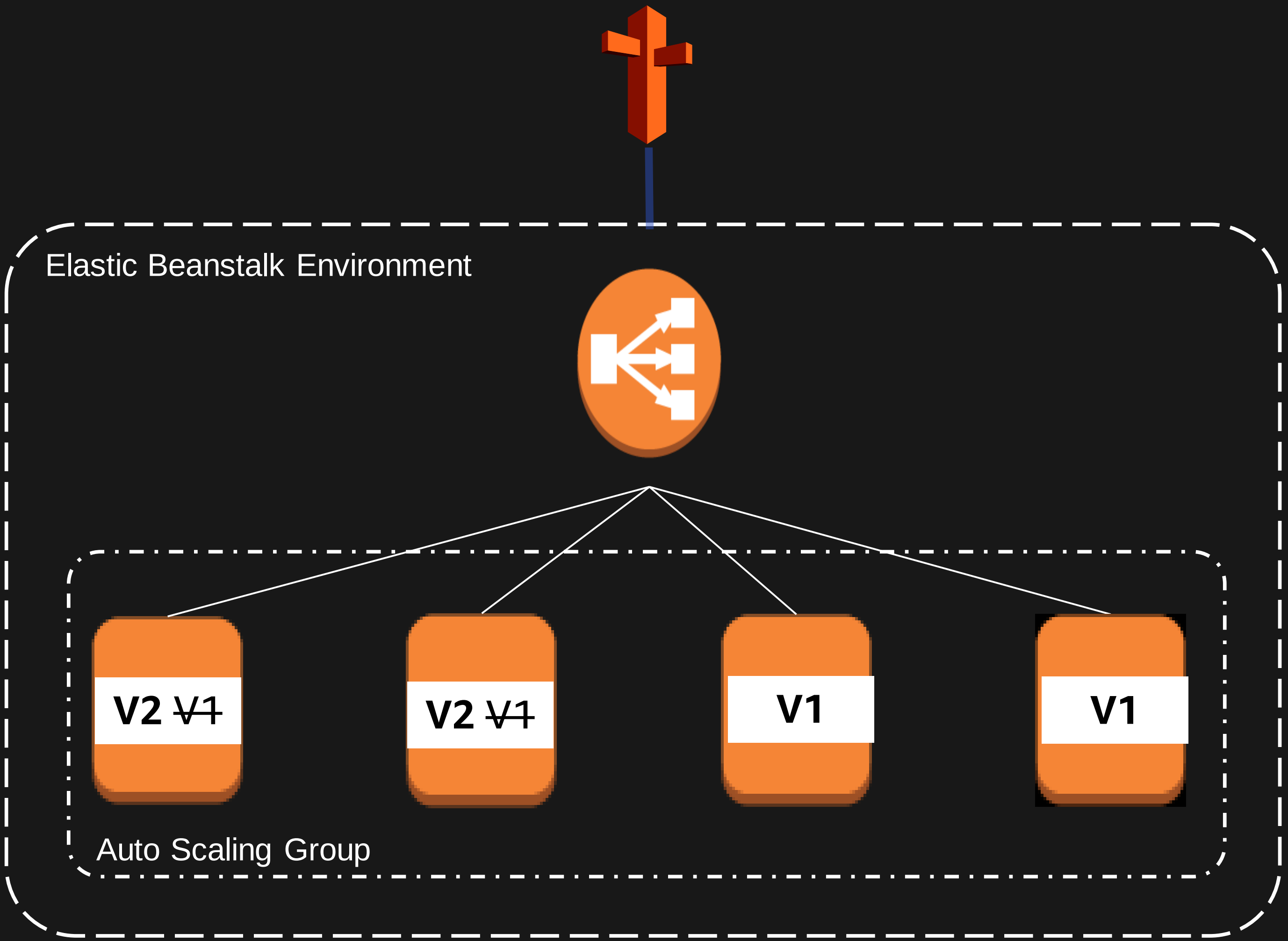
Deployment option: All at once

myapp.us-east-1.elasticbeanstalk.com



Deployment option: Rolling

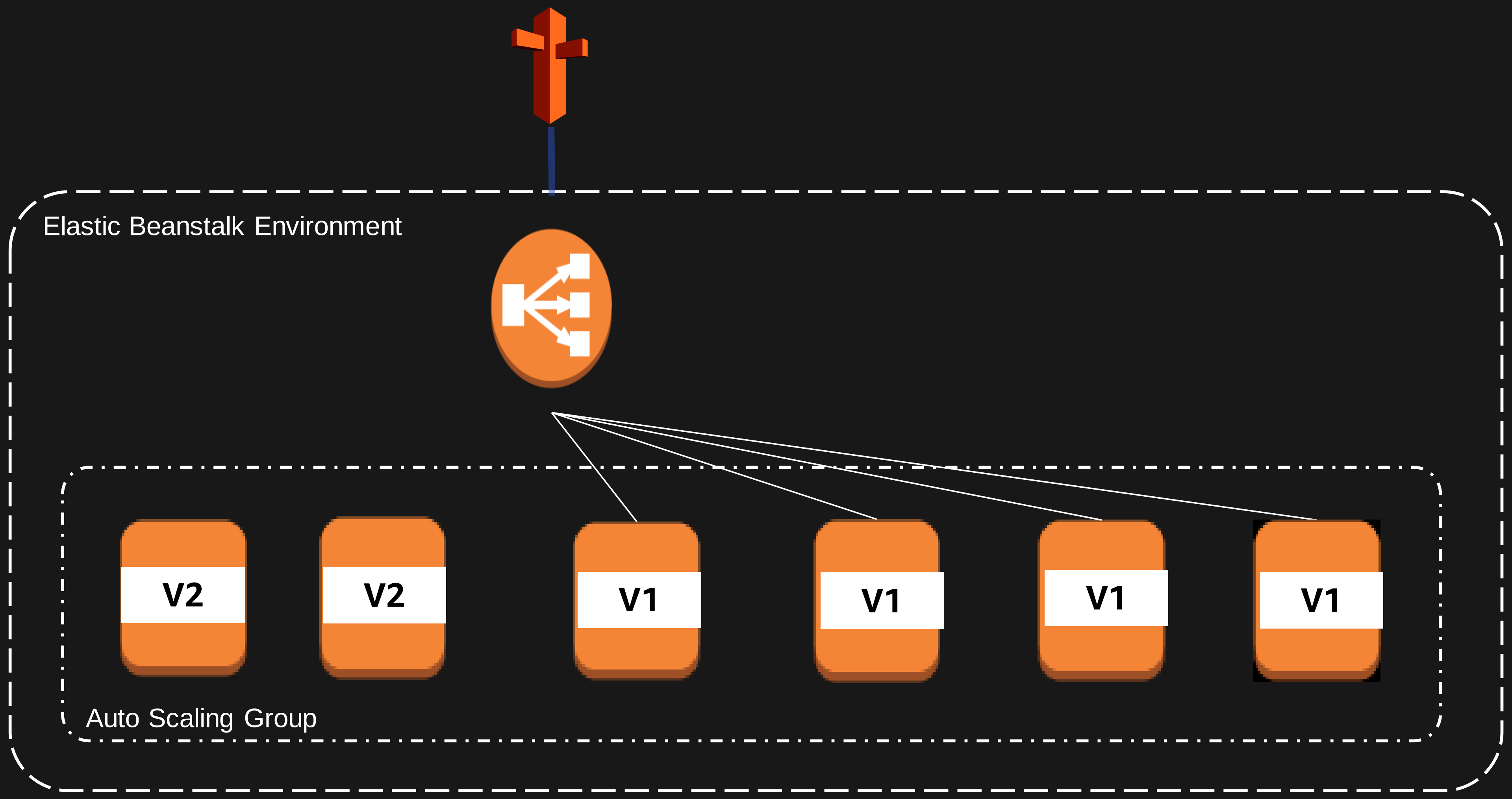
myapp.us-east-1.elasticbeanstalk.com



Batch size = 2

Deployment option: Rolling with additional batch

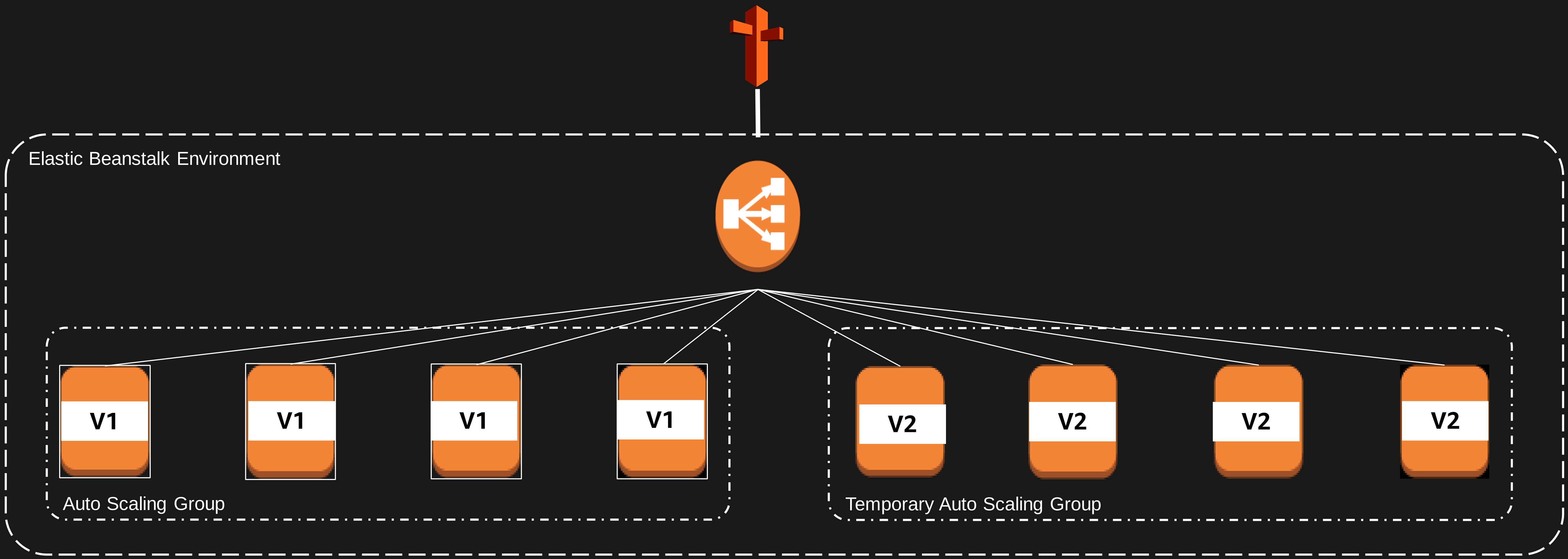
myapp.us-east-1.elasticbeanstalk.com



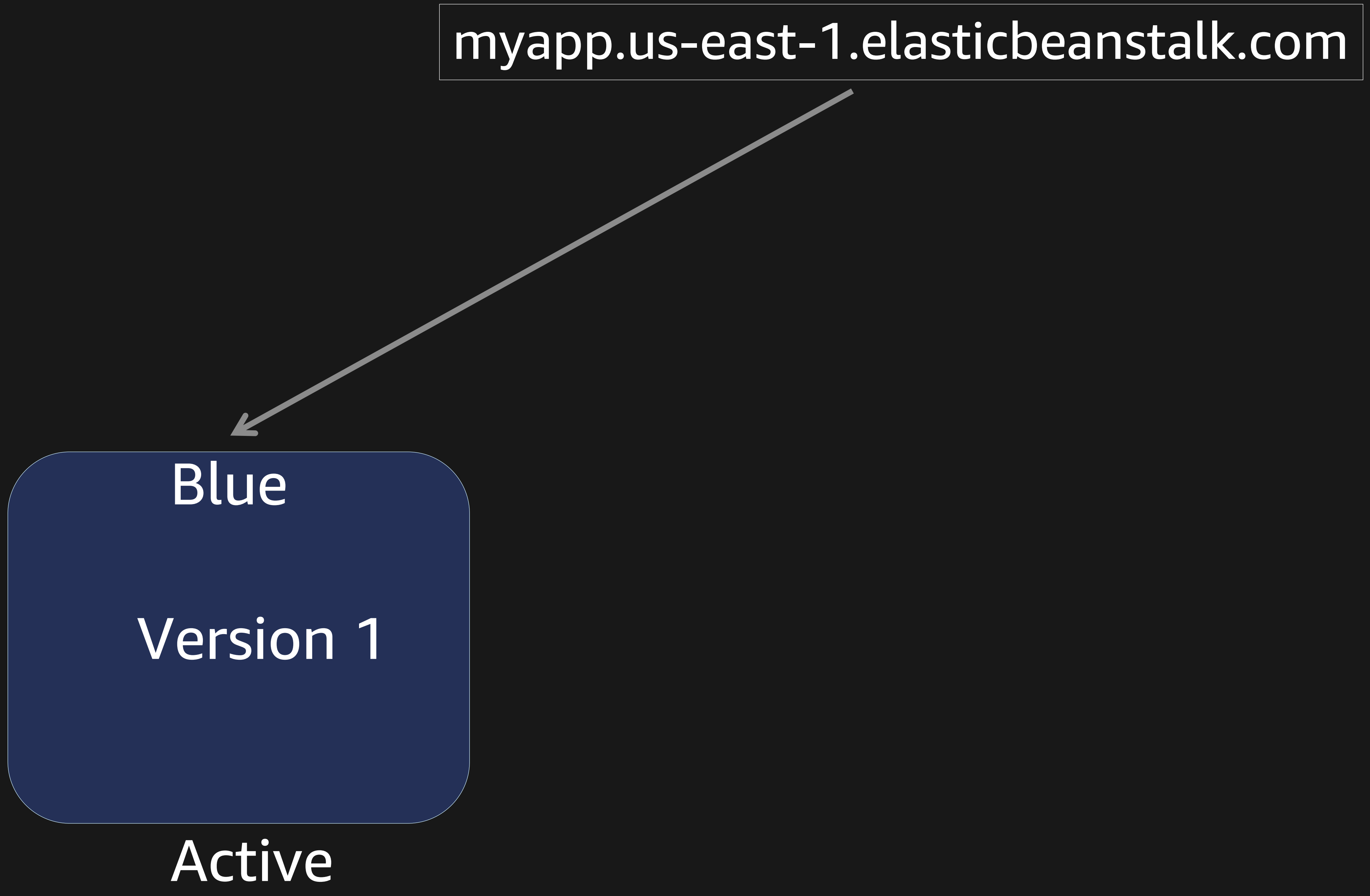
Batch size = 2

Deployment option: Immutable

myapp.us-east-1.elasticbeanstalk.com



Blue-Green deployments



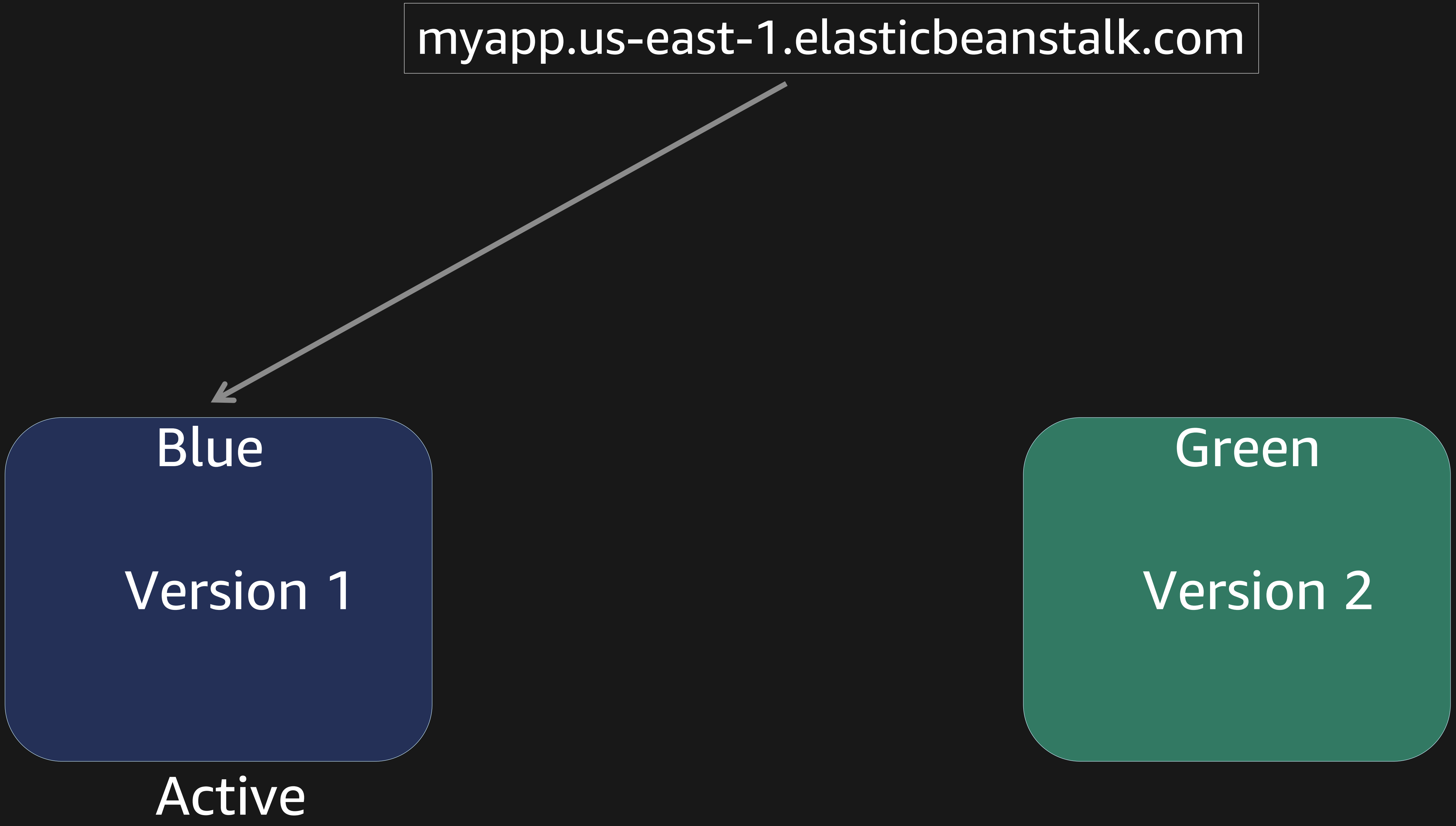
myapp.us-east-1.elasticbeanstalk.com

Blue

Version 1

Active

Blue-Green deployments



myapp.us-east-1.elasticbeanstalk.com

Blue

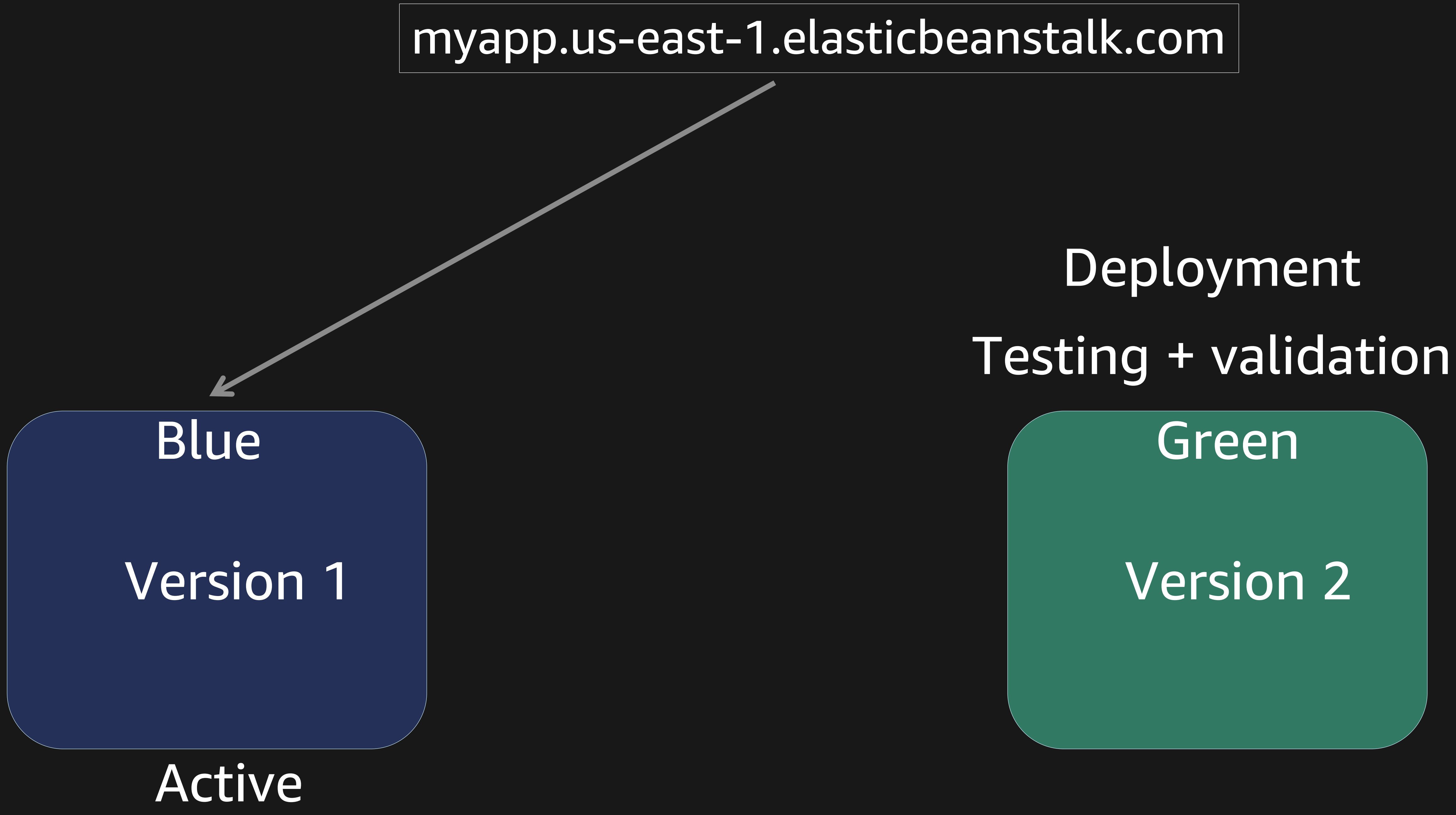
Version 1

Active

Green

Version 2

Blue-Green deployments



myapp.us-east-1.elasticbeanstalk.com

Deployment

Testing + validation

Blue

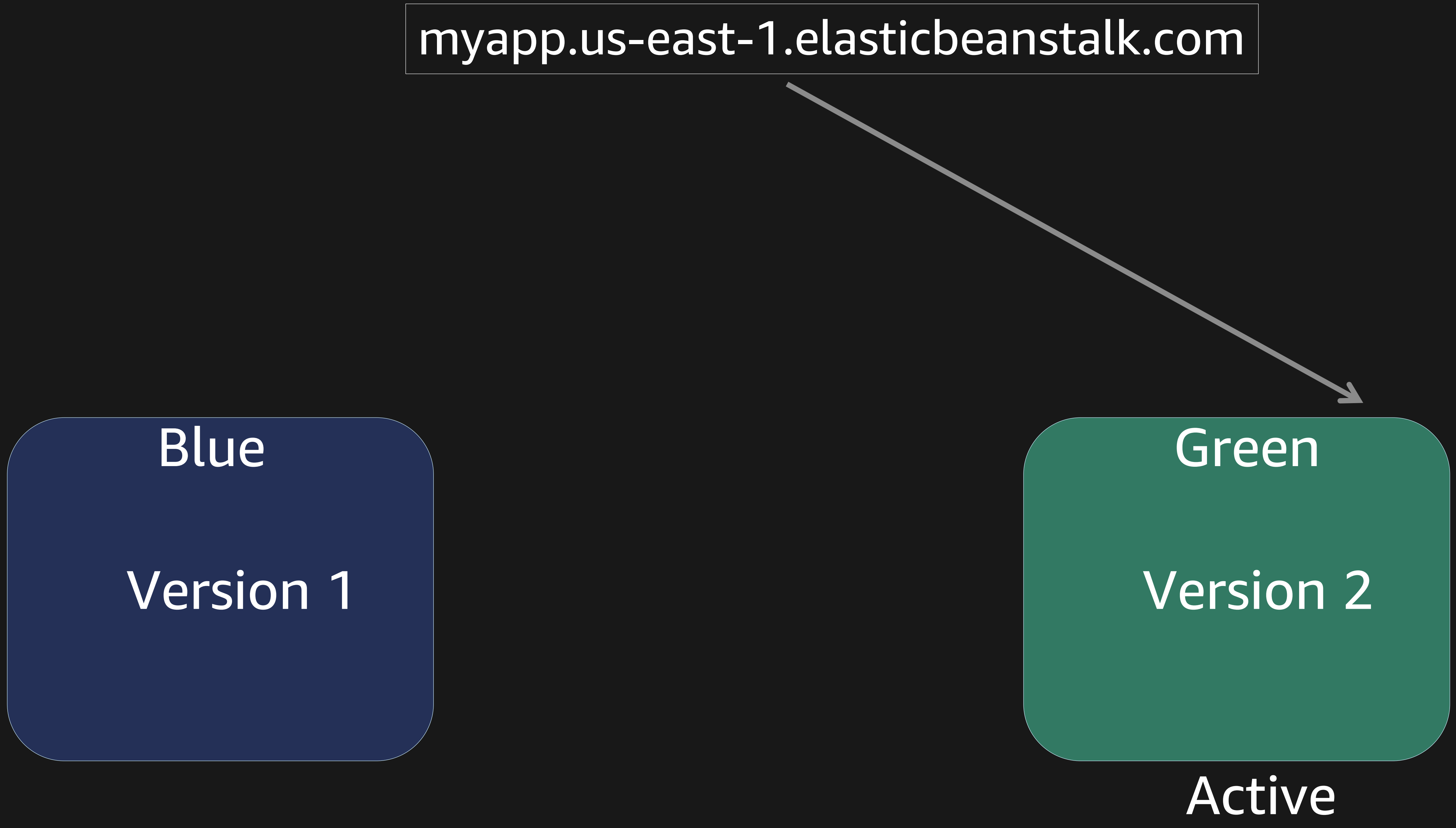
Version 1

Active

Green

Version 2

Blue-Green deployments



myapp.us-east-1.elasticbeanstalk.com

Blue

Version 1

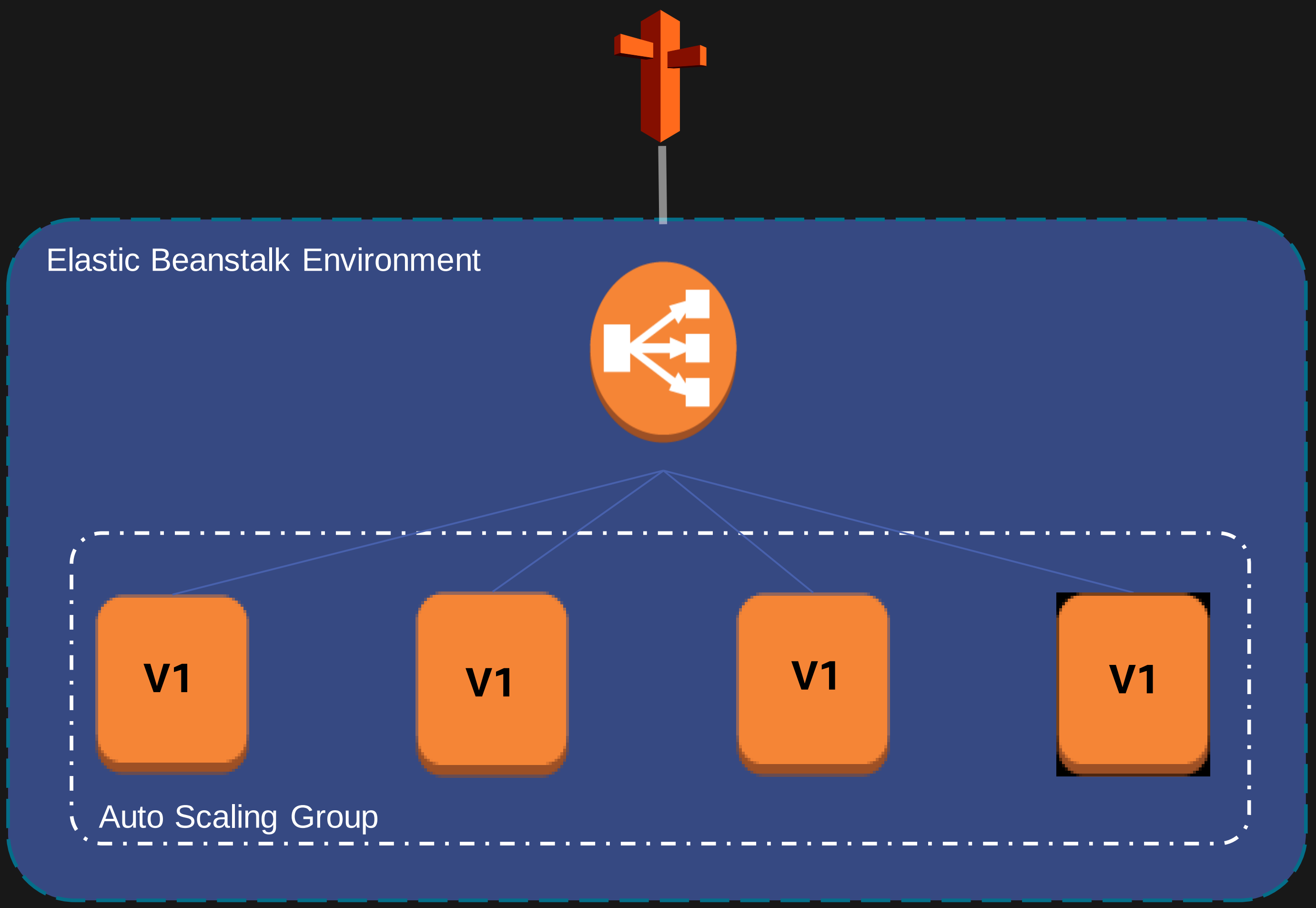
Green

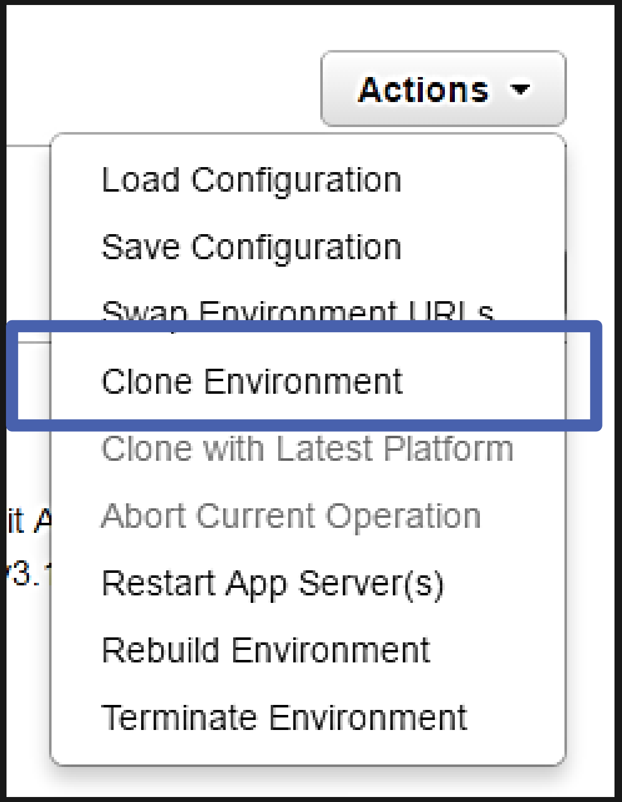
Version 2

Active

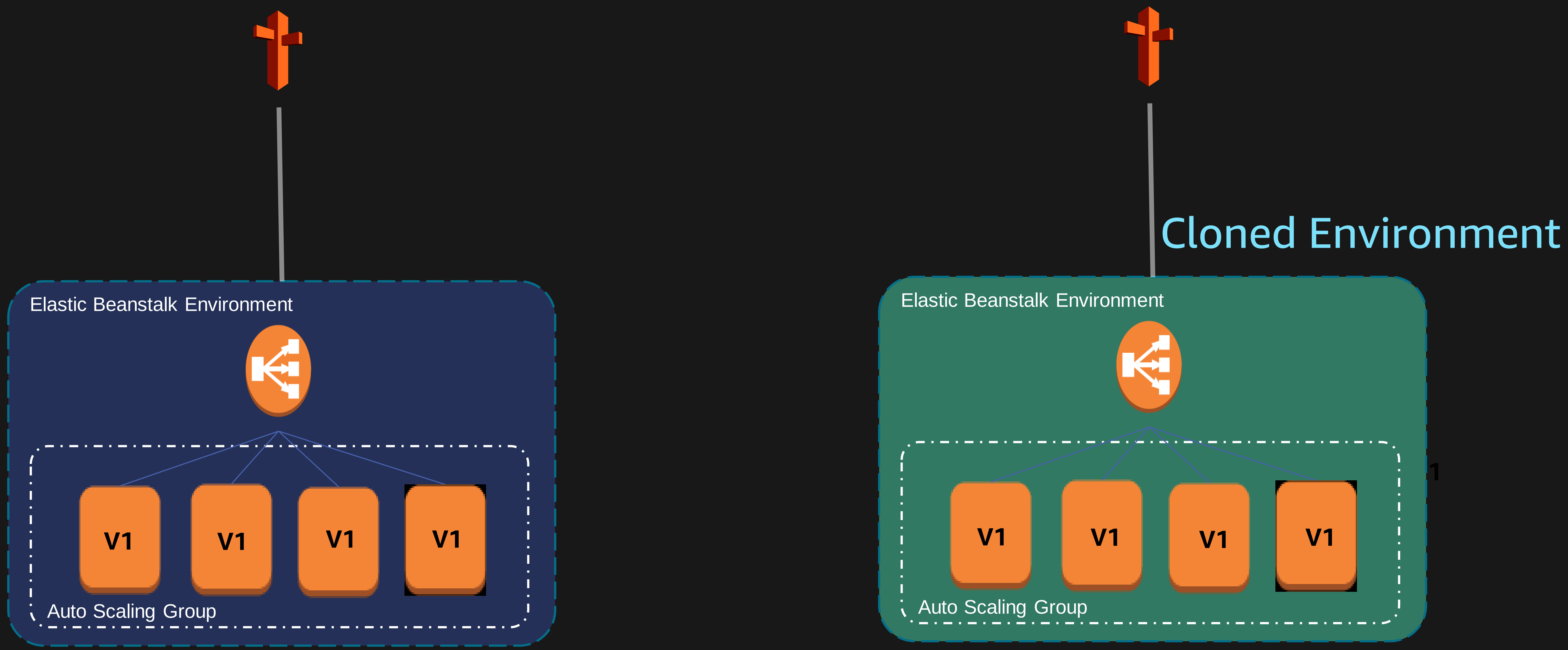
Blue-Green deployments – Step1: Clone the Environment

myapp.us-east-1.elasticbeanstalk.com





Blue-Green deployments - Step1: Clone the Environment



myapp.us-east-1.elasticbeanstalk.com

Elastic Beanstalk Environment

V1 V1 V1 V1

Auto Scaling Group

myapp_new.us-east-1.elasticbeanstalk.com

Cloned Environment

Elastic Beanstalk Environment

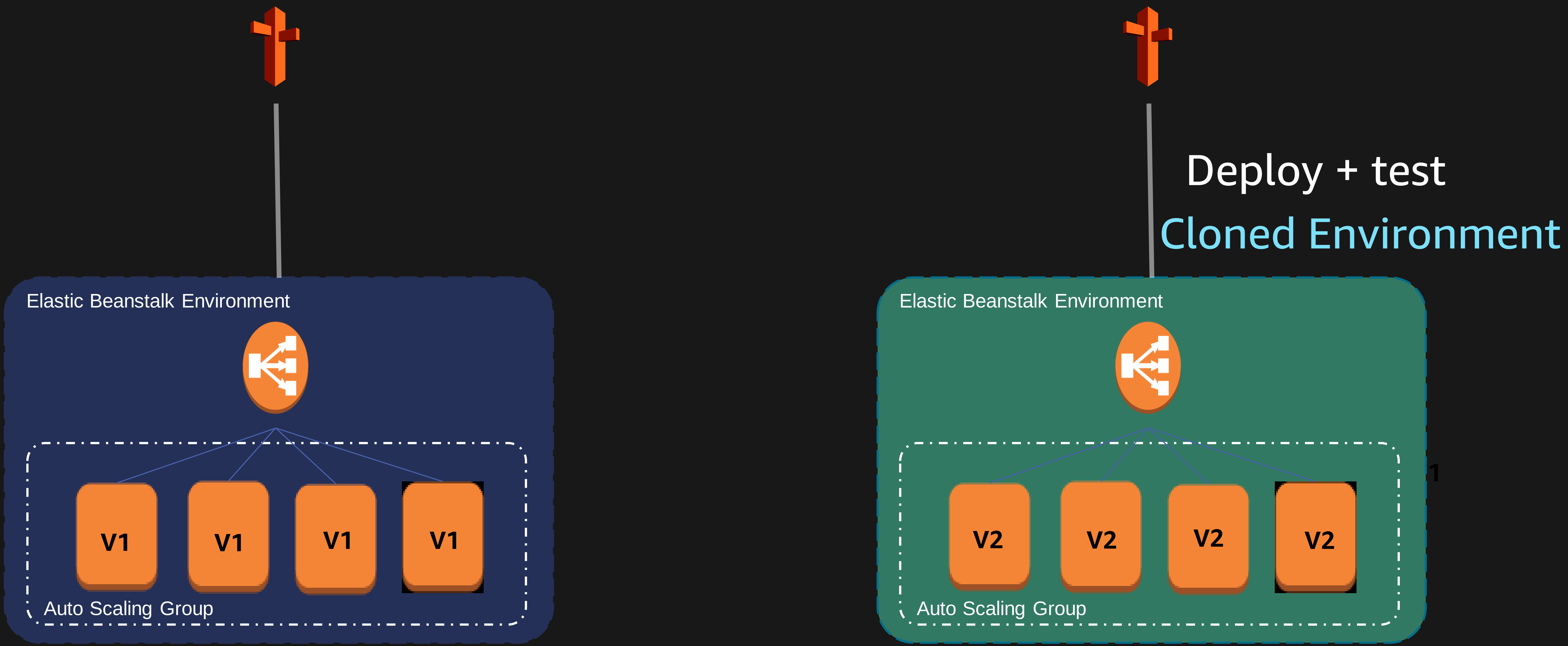
V1 V1 V1 V1

Auto Scaling Group

Blue-Green deployments – Step 2: Deploy changes to cloned Environment

myapp.us-east-1.elasticbeanstalk.com

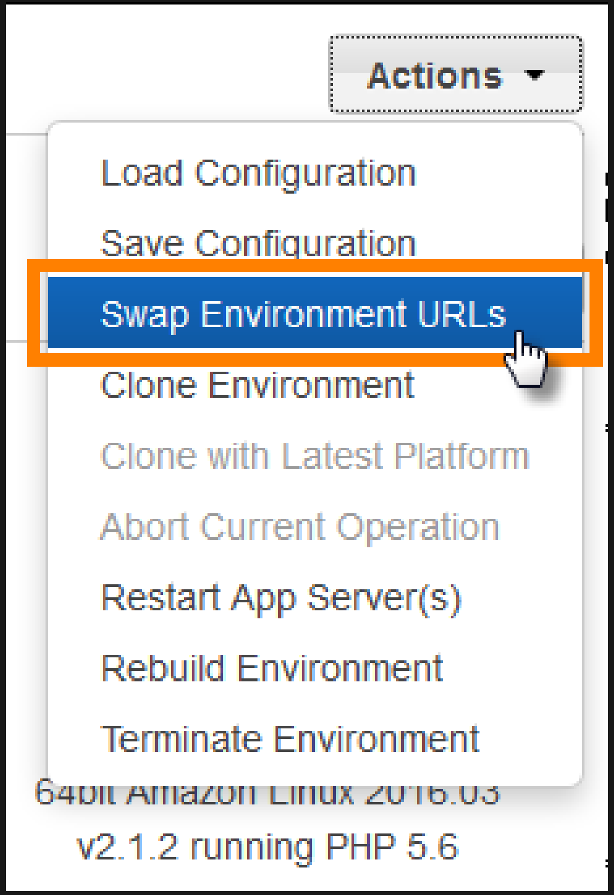
myapp_new.us-east-1.elasticbeanstalk.com

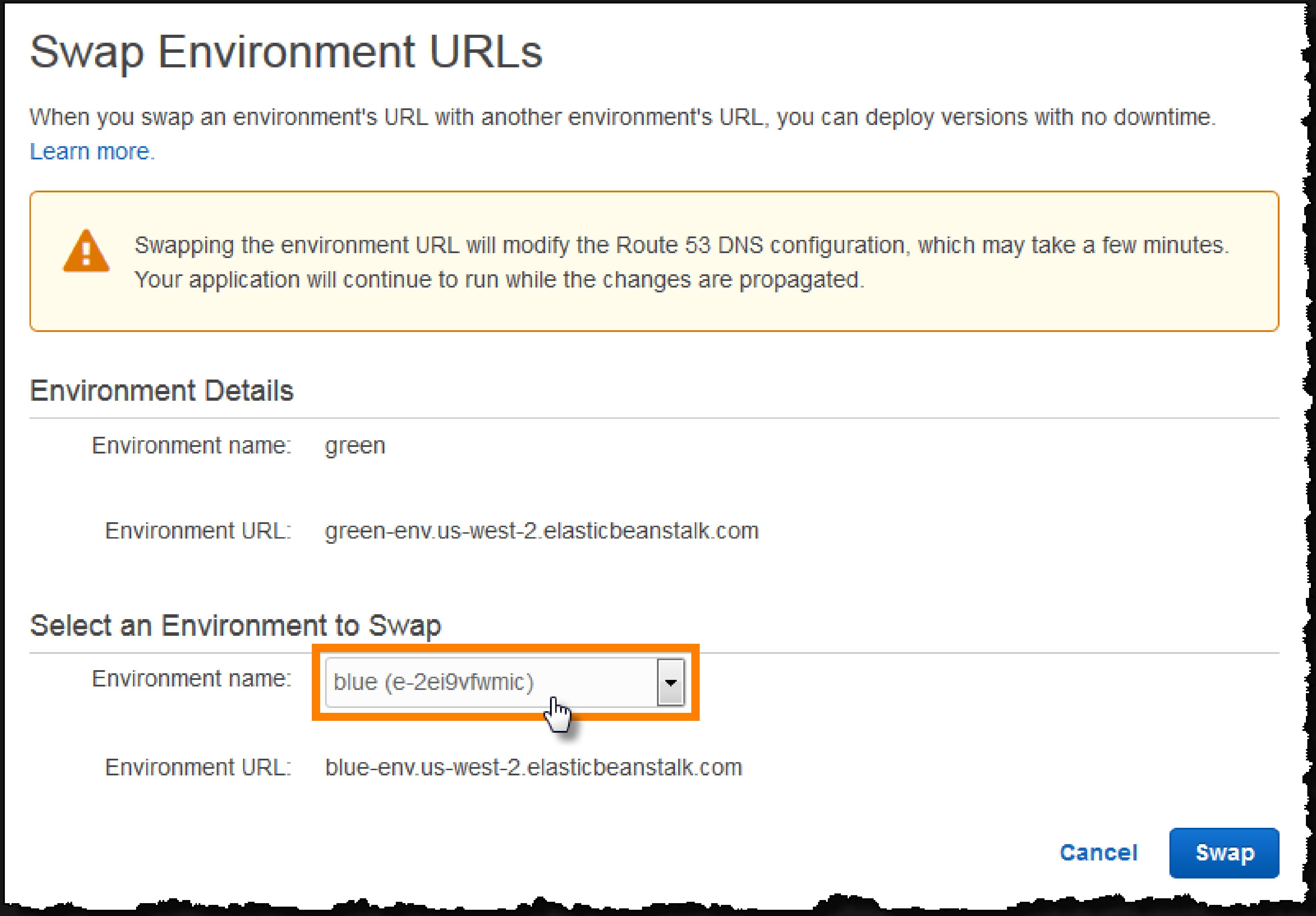


Deploy + test

Cloned Environment

Blue-Green deployments – Step 3: Swap Environment URL





Swap Environment URLs

When you swap an environment's URL with another environment's URL, you can deploy versions with no downtime. Learn more.

Swapping the environment URL will modify the Route 53 DNS configuration, which may take a few minutes. Your application will continue to run while the changes are propagated.

Environment Details

Environment name: green

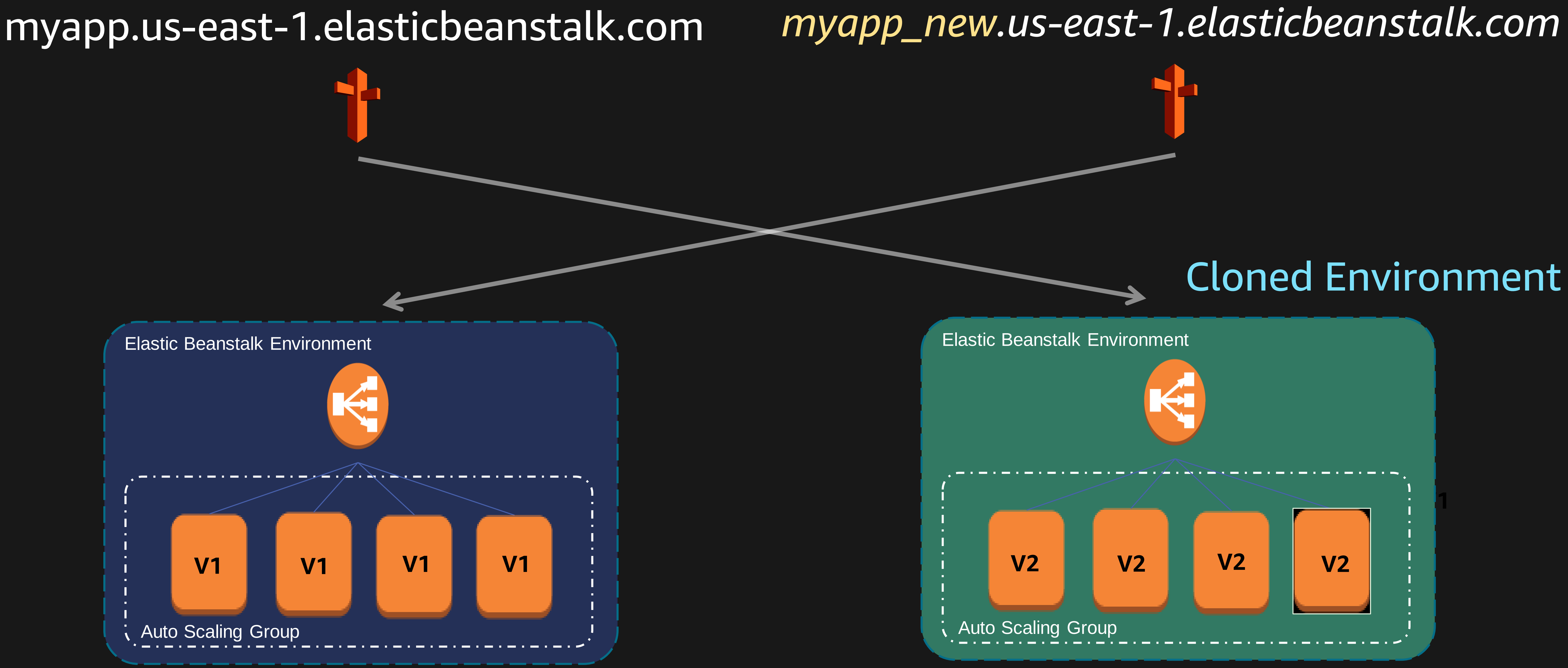
Environment URL: green-env.us-west-2.elasticbeanstalk.com

Select an Environment to Swap

Environment name: blue (e-2ei9vfwmic)

Environment URL: blue-env.us-west-2.elasticbeanstalk.com

Blue-Green deployments – Step 3: Swap Environment URL


myapp.us-east-1.elasticbeanstalk.com
myapp_new.us-east-1.elasticbeanstalk.com
Cloned Environment
Elastic Beanstalk Environment
V1
V1
V1
V1
Auto Scaling Group
Elastic Beanstalk Environment
V2
V2
V2
V2
Auto Scaling Group
1

Build your own enterprise platform using Elastic Beanstalk

- Develop a front-end catalog of applications, some provisioned using Elastic Beanstalk, others directly on EC2 instances or container services such as AWS Fargate or Amazon EKS
- Apply best practices to applications that are allowed and prevent configuration drift
- Seamless enterprise integration (single-sign-on, logging, DNS, etc.)

Example: Qualcomm Forge, check DEV 323: PaaS – From code to running application using AWS Elastic Beanstalk: Link

Agenda

- Elastic Beanstalk: a set of tools in your productivity toolkit
- [DEMO] Migrate your application + Try it at home kit
- [DEMO] Using Docker containers on Elastic Beanstalk
- Advanced use cases: CI/CD, Blue/Green
- [EXCLUSIVE]: Elastic Beanstalk feature launches NEW
- [DEMO] Machine Learning on Beanstalk

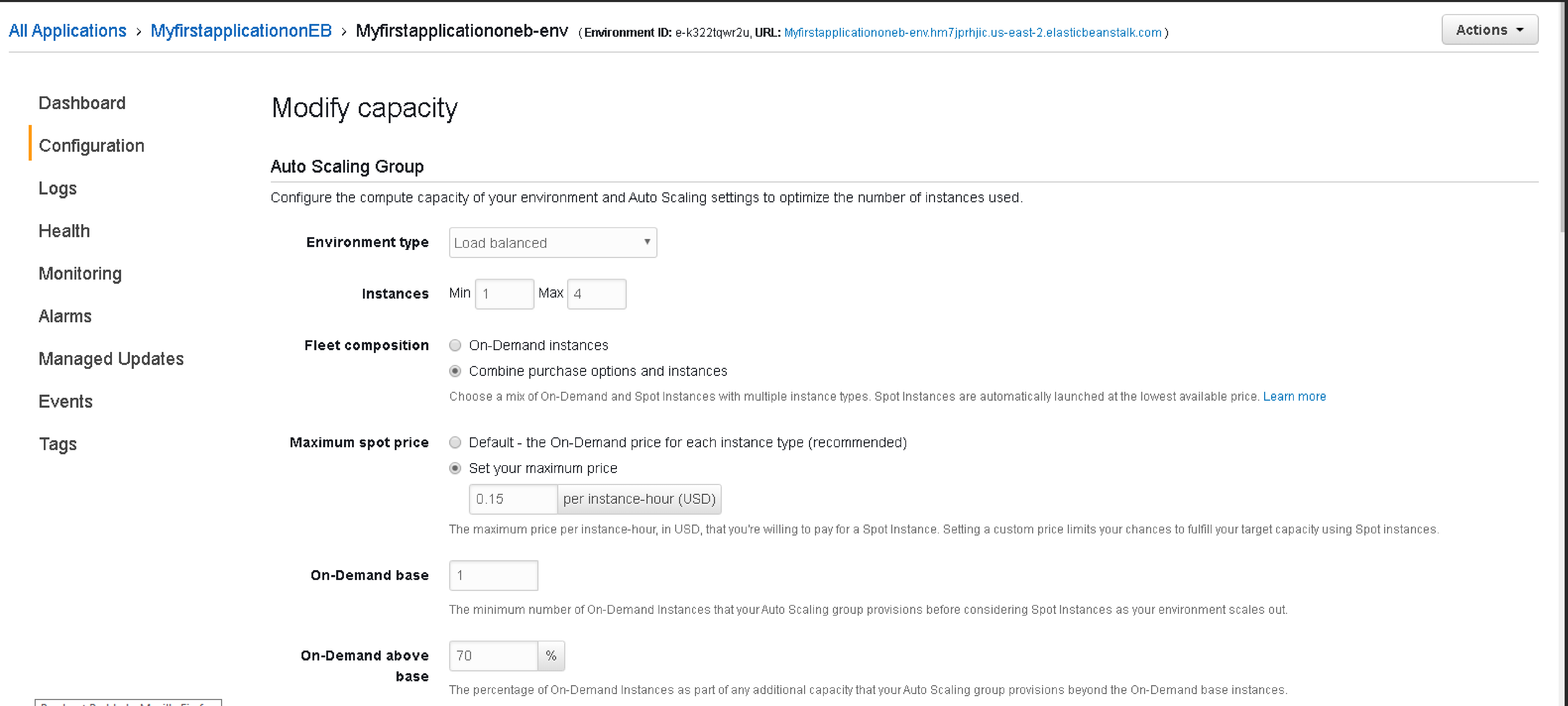
Elastic Beanstalk: New feature launches

NEW

#1: Enable scaling with EC2 Spot Instances

- Scale your applications with Spot Instances and save!
- No additional ongoing operational effort needed
- Available on EB CLI, management console, and APIs

Add EC2 Spot Instances to your mix



(Environment ID: e-k322tqwr2u, URL: Myfirstapplicationoneb-env.hm7jprhjic.us-east-2.elasticbeanstalk.com)

Modify capacity

Auto Scaling Group

Configure the compute capacity of your environment and Auto Scaling settings to optimize the number of instances used.

Environment type Load balanced

Instances Min 1 Max 4

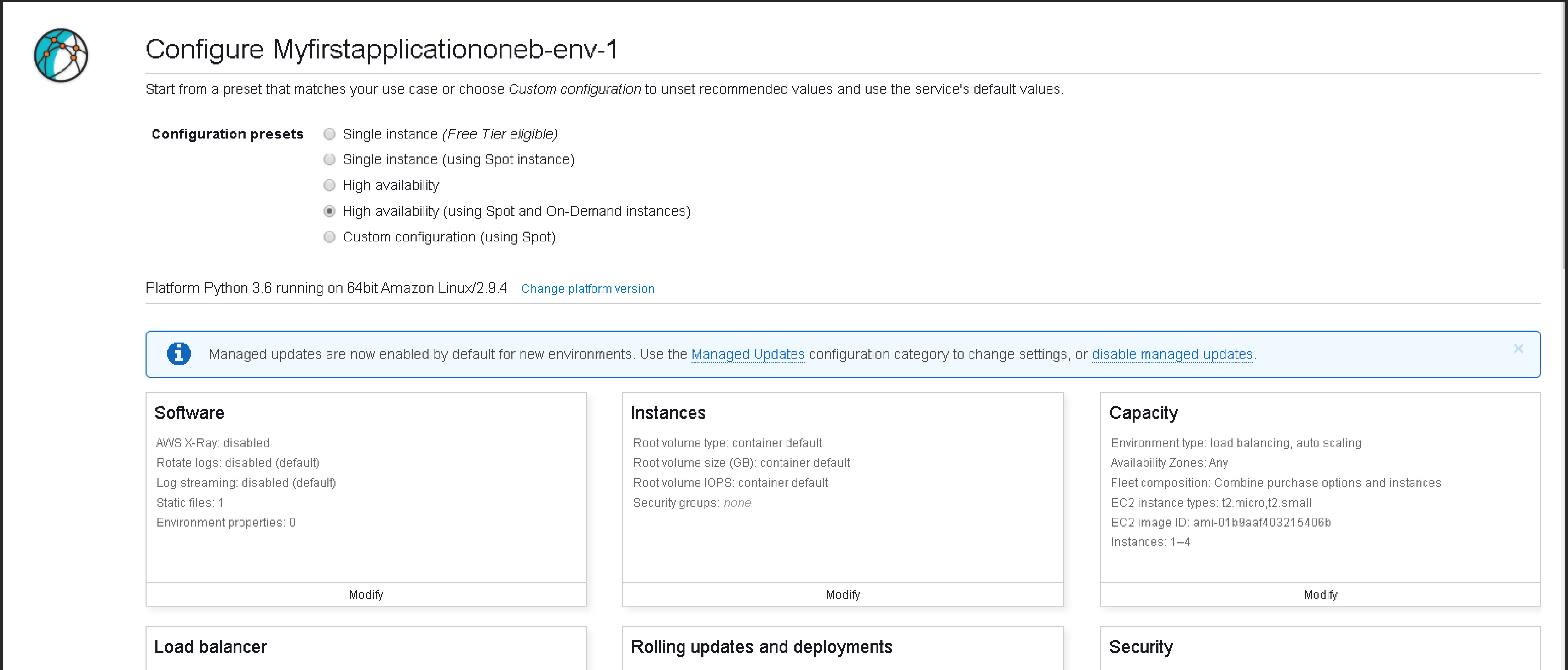
Fleet composition
○ On-Demand instances
● Combine purchase options and instances
Choose a mix of On-Demand and Spot Instances with multiple instance types. Spot Instances are automatically launched at the lowest available price. Learn more

Maximum spot price
○ Default - the On-Demand price for each instance type (recommended)
● Set your maximum price
0.15 per instance-hour (USD)
The maximum price per instance-hour, in USD, that you're willing to pay for a Spot Instance. Setting a custom price limits your chances to fulfill your target capacity using Spot instances.

On-Demand base 1
The minimum number of On-Demand Instances that your Auto Scaling group provisions before considering Spot Instances as your environment scales out.

On-Demand above base 70 %
The percentage of On-Demand Instances as part of any additional capacity that your Auto Scaling group provisions beyond the On-Demand base instances.

Or create new Environments with Spot/On-Demand Instances



Configure Myfirstapplicationoneb-env-1

Start from a preset that matches your use case or choose *Custom configuration* to unset recommended values and use the service's default values.

Configuration presets

○ Single instance *(Free Tier eligible)*
○ Single instance (using Spot instance)
○ High availability
● High availability (using Spot and On-Demand instances)
○ Custom configuration (using Spot)

Platform Python 3.6 running on 64bit Amazon Linux/2.9.4 Change platform version

Managed updates are now enabled by default for new environments. Use the Managed Updates configuration category to change settings, or disable managed updates.

Software

AWS X-Ray: disabled
Rotate logs: disabled (default)
Log streaming: disabled (default)
Static files: 1
Environment properties: 0

Modify

Instances

Root volume type: container default
Root volume size (GB): container default
Root volume IOPS: container default
Security groups: *none*

Modify

Capacity

Environment type: load balancing, auto scaling
Availability Zones: Any
Fleet composition: Combine purchase options and instances
EC2 instance types: t2.micro,t2.small
EC2 image ID: ami-01b9aaf403215406b
Instances: 1–4

Modify

Load balancer

Rolling updates and deployments

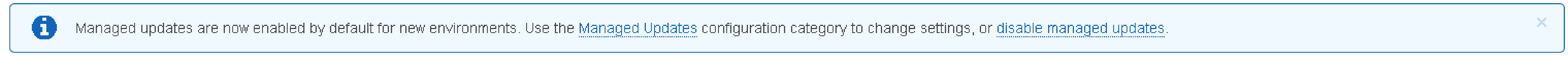
Security

Elastic Beanstalk: New feature launches

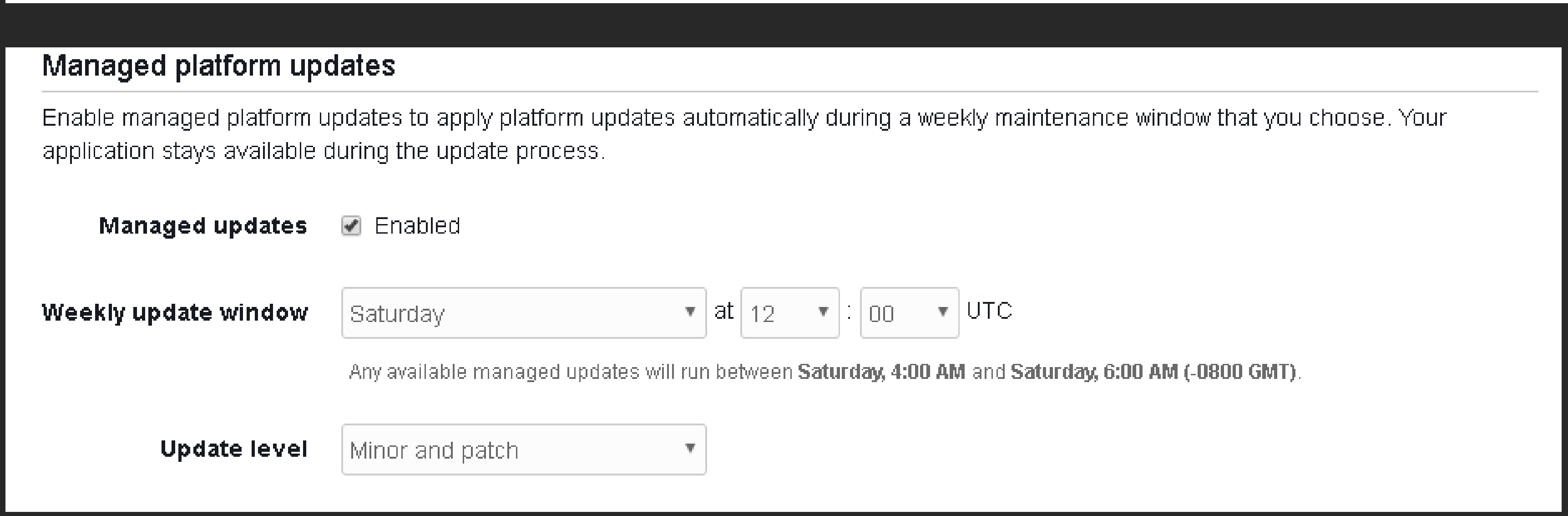
NEW

#2: Managed Updates now turned on by default

- Automatically applies security fixes, patches, and minor updates
- Update window chosen at random
- Turning on will not change application availability



Managed updates are now enabled by default for new environments. Use the Managed Updates configuration category to change settings, or disable managed updates.



Managed platform updates

Enable managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

Managed updates ☑ Enabled

Weekly update window Saturday at 12 : 00 UTC

Any available managed updates will run between **Saturday, 4:00 AM** and **Saturday, 6:00 AM (-0800 GMT)**.

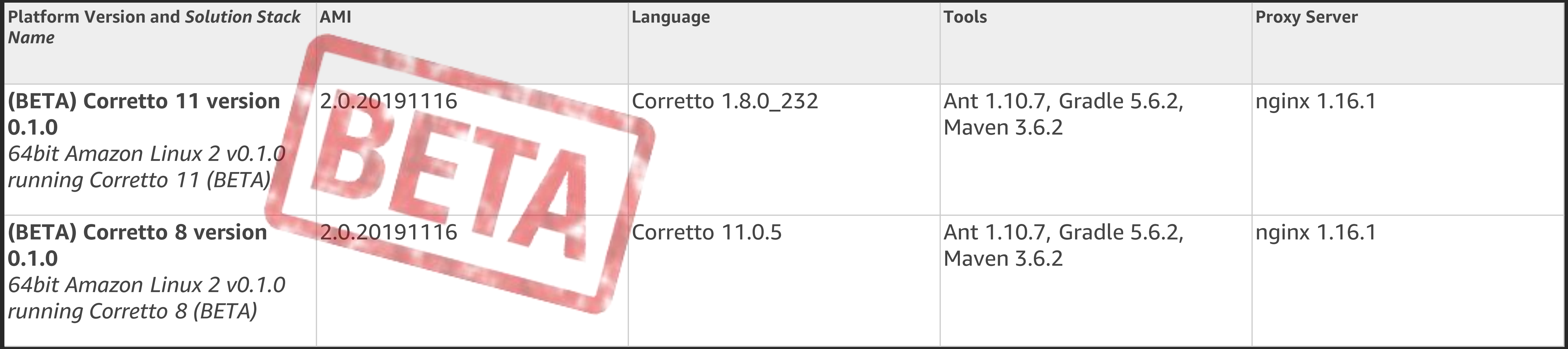
Update level Minor and patch

Elastic Beanstalk: New feature launches

NEW

#3: New platform: Amazon Linux 2 with Open JDK Corretto 8, 11 (BETA)

Amazon Corretto is a no-cost, multiplatform, production-ready distribution of the Open Java Development Kit (OpenJDK)



Platform Version and *Solution Stack Name*	AMI	Language	Tools	Proxy Server
(BETA) Corretto 11 version 0.1.0 *64bit Amazon Linux 2 v0.1.0 running Corretto 11 (BETA)*	2.0.20191116	Corretto 1.8.0_232	Ant 1.10.7, Gradle 5.6.2, Maven 3.6.2	nginx 1.16.1
(BETA) Corretto 8 version 0.1.0 *64bit Amazon Linux 2 v0.1.0 running Corretto 8 (BETA)*	2.0.20191116	Corretto 11.0.5	Ant 1.10.7, Gradle 5.6.2, Maven 3.6.2	nginx 1.16.1

BETA

Agenda

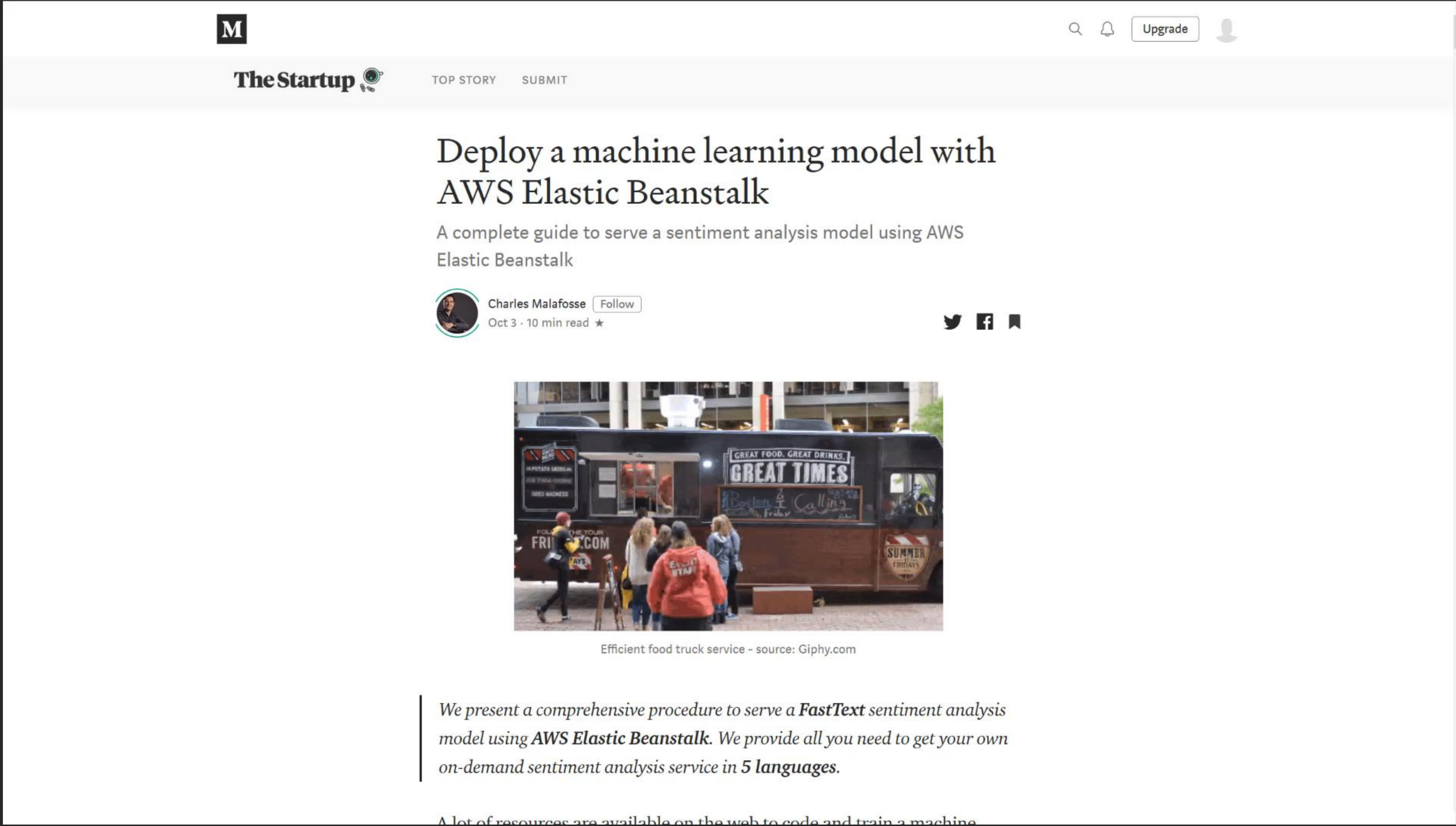
- Elastic Beanstalk: a set of tools in your productivity toolkit
- [DEMO] Migrate your application + Try it at home kit
- [DEMO] Using Docker containers on Elastic Beanstalk
- Advanced use cases: CI/CD, Blue/Green
- [EXCLUSIVE]: Elastic Beanstalk feature launches NEW
- [DEMO] Machine Learning on Beanstalk

Developer productivity: Task 3

Set up a machine learning app for sentiment analysis, and scale it

Toolkit:

- Elastic Beanstalk Environment with managed Python
- Scale with multi-instance platform
- Reduce costs by using Spot Instances
- No infrastructure provisioning or ongoing management needed



Deploy a machine learning model with AWS Elastic Beanstalk

A complete guide to serve a sentiment analysis model using AWS Elastic Beanstalk

Charles Malafosse
Oct 3 · 10 min read ★

Efficient food truck service - source: Giphy.com

> *We present a comprehensive procedure to serve a **FastText** sentiment analysis model using **AWS Elastic Beanstalk**. We provide all you need to get your own on-demand sentiment analysis service in **5 languages**.*

A lot of resources are available on the web to code and train a machine

Elastic Beanstalk: Your productivity toolkit

- Get started with web applications using one of 12 curated platforms
- Perform lift-and-shift migration of Windows .NET workloads into a managed platform with health monitoring, automated managed updates, and more

https://github.com/awslabs/windows-web-app-migration-assistant

- Modernize your application using multi-container Docker images on Elastic Beanstalk

https://github.com/aws-samples/eb-docker-nginx-proxy

- Not just about getting started: Monitor health, apply updates, debug, and analyze
- Advanced scenarios: Deployment options, CI/CD, Blue-Green, build a PaaS
- Feature: Scale with EC2 Spot Instances, at no additional cost
- Feature: Managed updates now applied automatically
- Feature: Amazon Linux 2 with Corretto 8 and Corretto 11 now available in BETA
- Try running a Machine Learning model on Elastic Beanstalk

https://github.com/charlesmalafosse/aws-elasticbeanstalk-ml-server

Learn DevOps with AWS Training and Certification

Resources created by the experts at AWS to propel your organization and career forward

- Take free digital training to learn best practices for developing, deploying, and maintaining applications
- Classroom offerings, like DevOps Engineering on AWS, feature AWS expert instructors and hands-on activities
- Validate expertise with the **AWS Certified DevOps Engineer - Professional** or **AWS Certified Developer - Associate** exams

Visit aws.amazon.com/training/path-developing/

aws training and certification

Thank you!

Prashant Prahlad

pprahlad@amazon.com

Please complete the session survey in the mobile app.