

# Compute Services



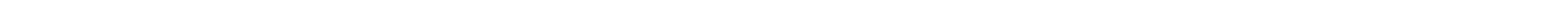
**David Tucker**

CTO Consultant

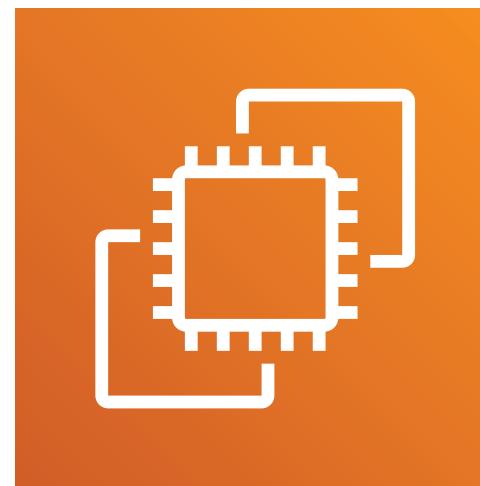
@\_davidtucker\_ | [davidtucker.net](http://davidtucker.net)

# Compute Services

A service that enables you to leverage cloud-based virtual machines for workloads. This could be serving web content to visitors, running a database, or calculating statistics from a data set.



# Compute Services on AWS



## Amazon EC2

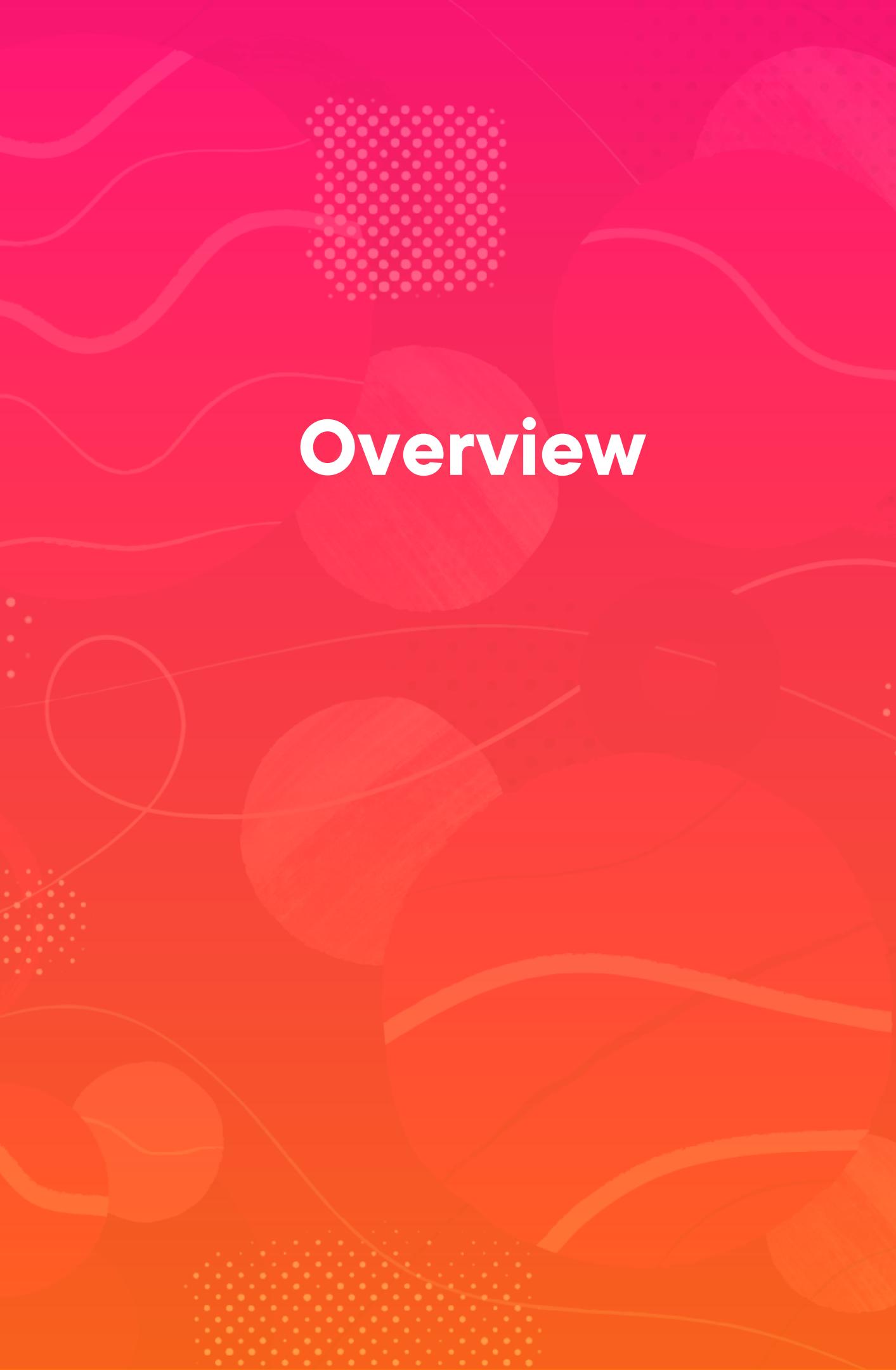
Provides secure and resizable virtual servers on AWS

## AWS Elastic Beanstalk

Platform for scaling and deploying web apps and services

## AWS Lambda

Enables compute without managing servers



## Overview

**Introducing Amazon EC2 capabilities**

**Exploring pricing approaches for EC2 instances**

**Introducing the capabilities of AWS Elastic Beanstalk**

**Reviewing use cases for Elastic Beanstalk**

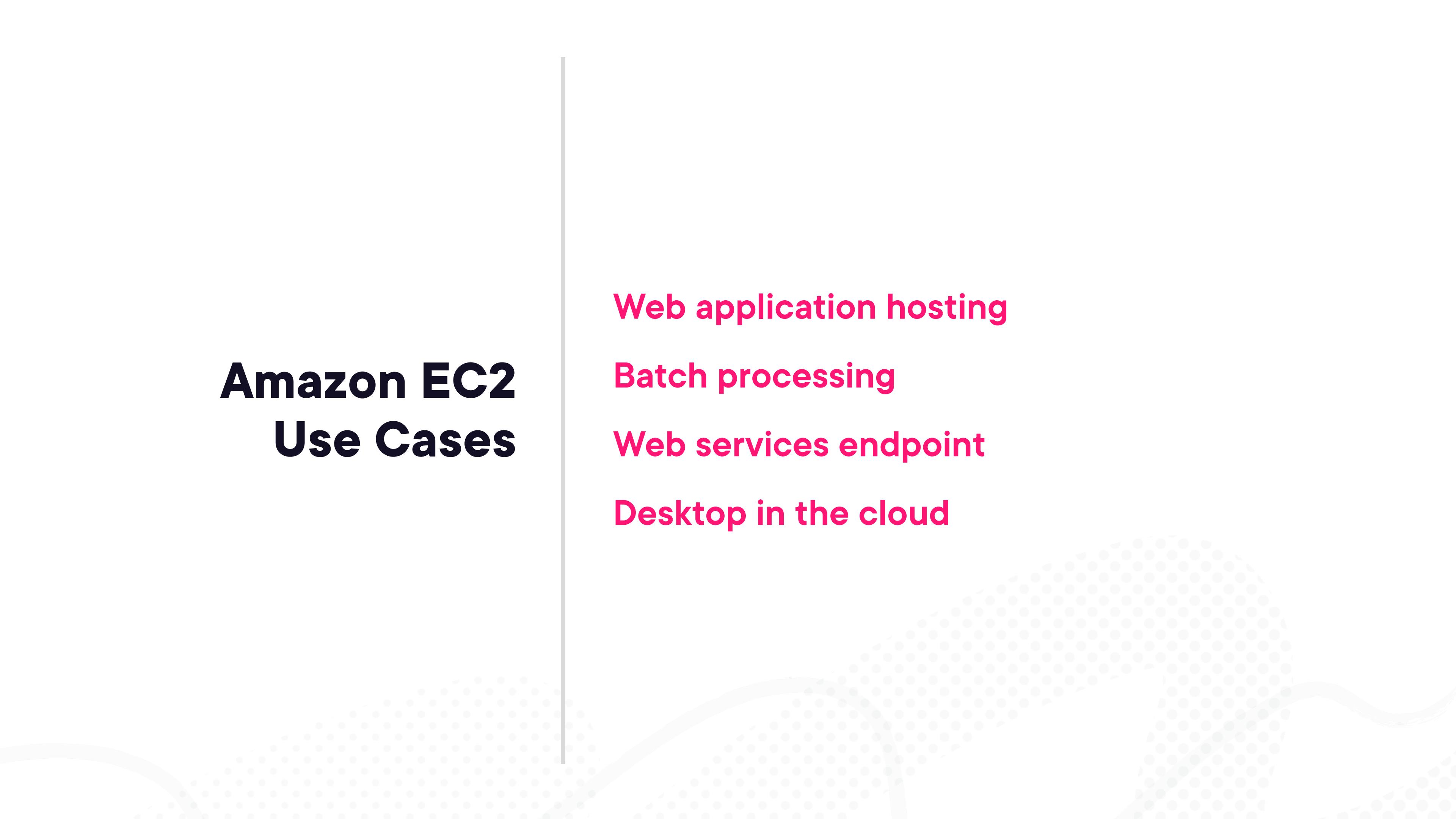
**Introducing AWS Lambda**



# Amazon EC2 Overview

**“Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.”**

# **Amazon EC2 Use Cases**



**Web application hosting**

**Batch processing**

**Web services endpoint**

**Desktop in the cloud**

# Amazon EC2 Concepts

**Instance Types**

**Root Device Type**

**Amazon Machine Image (AMI)**

**Purchase Options**

# Amazon EC2 Instance Types

**Defines the processor, memory, and storage type**

**Cannot be changed without downtime**

**Provided in the following categories**

- General purpose
- Compute, memory, and storage optimized
- Accelerated computing

**Pricing is based on instance type**

**Some instance types have unique capabilities**

# Example EC2 Instance Type Pricing

	vCPU	Memory	Linux Pricing
<b>t3.medium</b>	2	4 GiB	\$0.0416 per Hour
<b>m5.large</b>	4	16 GiB	\$0.096 per Hour
<b>c5d.24xlarge</b>	96	375 GiB	\$4.608 per Hour
<b>p3.16xlarge</b>	64	488 GiB	\$24.48 per Hour
<b>i3.16xlarge</b>	64	488 GiB	\$4.992 per Hour

# Root Device Type

## Instance Store

Ephemeral storage that is physically attached to the host the virtual server is running on

## Elastic Block Store (EBS)

Persistent storage that exists separately from the host the virtual server is running on

# **Amazon Machine Image (AMI)**

**Template for an EC2 instance including configuration, operating system, and data**

**AWS provides many AMI's that can be leveraged**

**AMI's can be shared across AWS accounts**

**Custom AMI's can be created based on your configuration**

**Commercial AMI's are available in the AWS Marketplace**



# Amazon EC2 Purchase Types

# Amazon EC2 Purchase Options

**On-Demand**

**Reserved**

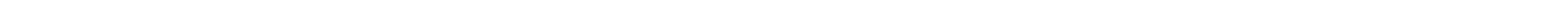
**Savings Plan**

**Spot**

**Dedicated**

# Reserved Instances

**Provides discounts over the on-demand model when you can commit to a specific period of time. In addition, it provides a capacity reservation for the specific instance type that you specify.**



# EC2 Reserved Instance Types

## Standard

Highest discount, works for steady workloads

## Convertible

Enables the conversion of attributes, works for steady workloads

## Scheduled

Works for a time window you reserve, good for a predictable workload

# Standard Reserved Instance Cost Models

## All Upfront

Entire cost for the 1 or 3 year period is paid upfront

## Partial Upfront

Part of 1 or 3 year cost is paid upfront with a reduced monthly cost

## No Upfront

No upfront payment is made, but there is a reduced monthly cost

Maximum  
Savings

Minimum  
Upfront Cost

## Savings Plans

- Similar in concept to reserved instances
- Supports compute with EC2, Fargate, and Lambda
- Unlike Reserved Instances, it does not reserve capacity
- Provide savings of up to 72%
- Comes in 1 or 3 year terms

**Spot instances enable you  
to leverage excess EC2  
compute capacity.**

# Spot Instances

**Can provide up to 90% discount over on-demand pricing**

**There is a market price for instance types per availability zone called the Spot price**

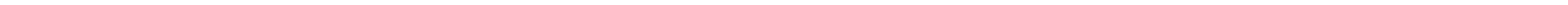
**When you request instances, if your bid is higher than Spot price they will launch**

**If the Spot price grows to exceed your bid, the instances will be terminated**

**Spot instances can be notified 2 minutes prior to termination**

# Dedicated Host

The dedicated host pricing model gives you a dedicated physical server. It will be the most expensive option, but it may be required for either server software licensing or due to a compliance requirement.



# Amazon EC2 Purchase Options



If you have an instance that is consistent and always needed, you should purchase a Standard or Convertible Reserved Instance.



If you have batch processing where the process can start and stop without affecting the job, you should leverage Spot Instances.

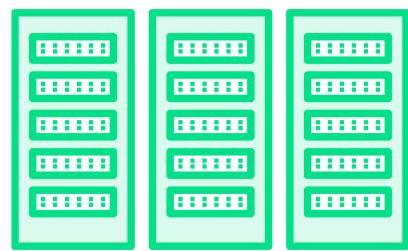


If you have an inconsistent need for instances that cannot be stopped without affecting the job, leverage On-Demand Instances.

# Amazon EC2 Purchase Options



If you have specific per-server licensing or if you have a compliance requirement for a dedicated server, you should use Dedicated Host.



If you are leveraging Lambda and/or Fargate alongside EC2 and want to achieve discounts for 1 or 3 years, choose a Savings Plan.



If you have a predictable but not steady workload in EC2, you should purchase a Scheduled Reserved Instance.

# Reserved Instance EC2 Pricing Example

	<b>On-Demand</b>	<b>All Upfront</b>	<b>Effective Hourly</b>	<b>Savings</b>
<b>t3.medium</b>	\$0.0416 per Hour	\$213.00 (1 Year) \$412.00 (3 Years)	\$0.024 (1 Year) \$0.015 (3 Years)	\$151.42 (1 Year) \$681.25 (3 Years)
<b>c5d.24xlarge</b>	<b>On-Demand</b>  \$4.608 per Hour	<b>Partial Upfront (1 Yr)</b>  \$12,124 (Upfront) \$1,010.32 (Monthly)	<b>Effective Hourly</b>  \$2.768	<b>Savings</b>  \$16118.40 (40%)
<b>i3.16xlarge</b>	<b>On-Demand</b>  \$4.992 per Hour	<b>No Upfront (3 Yr)</b>  \$1,765.87 (Monthly)	<b>Effective Hourly</b>  \$2.419	<b>Savings</b>  \$22539.48 (52%)

# Spot Instance EC2 Pricing Example

	On-Demand	Spot Pricing	Percentage Savings
<b>t3.medium</b>	\$0.0416 per Hour	\$0.0125 per Hour	70%
<b>c5d.24xlarge</b>	\$4.608 per Hour	\$0.9122 per Hour	80%
<b>i3.16xlarge</b>	\$4.992 per Hour	\$1.4976 per Hour	70%



# Launching EC2 Instances

# Demo

**Launching a new EC2 instance based on an AWS AMI**

**Exploring the EC2 launch wizard in the AWS Console**

**Configuring EC2 instance to be used as a web server**

**Terminating an EC2 instance**



# AWS Elastic Beanstalk Overview

# AWS Elastic Beanstalk

Automates the process of deploying and scaling workloads on EC2 (PaaS)

Supports a specific set of technologies

Leverages existing AWS services

Only pay for the other services you leverage

Handles provisioning, load balancing, scaling, and monitoring

Java  
.NET  
PHP  
Node.js  
Python  
Ruby  
Go  
Docker

## Supported Application Platforms

# Elastic Beanstalk Features

**Monitoring**

**Deployment**

**Scaling**

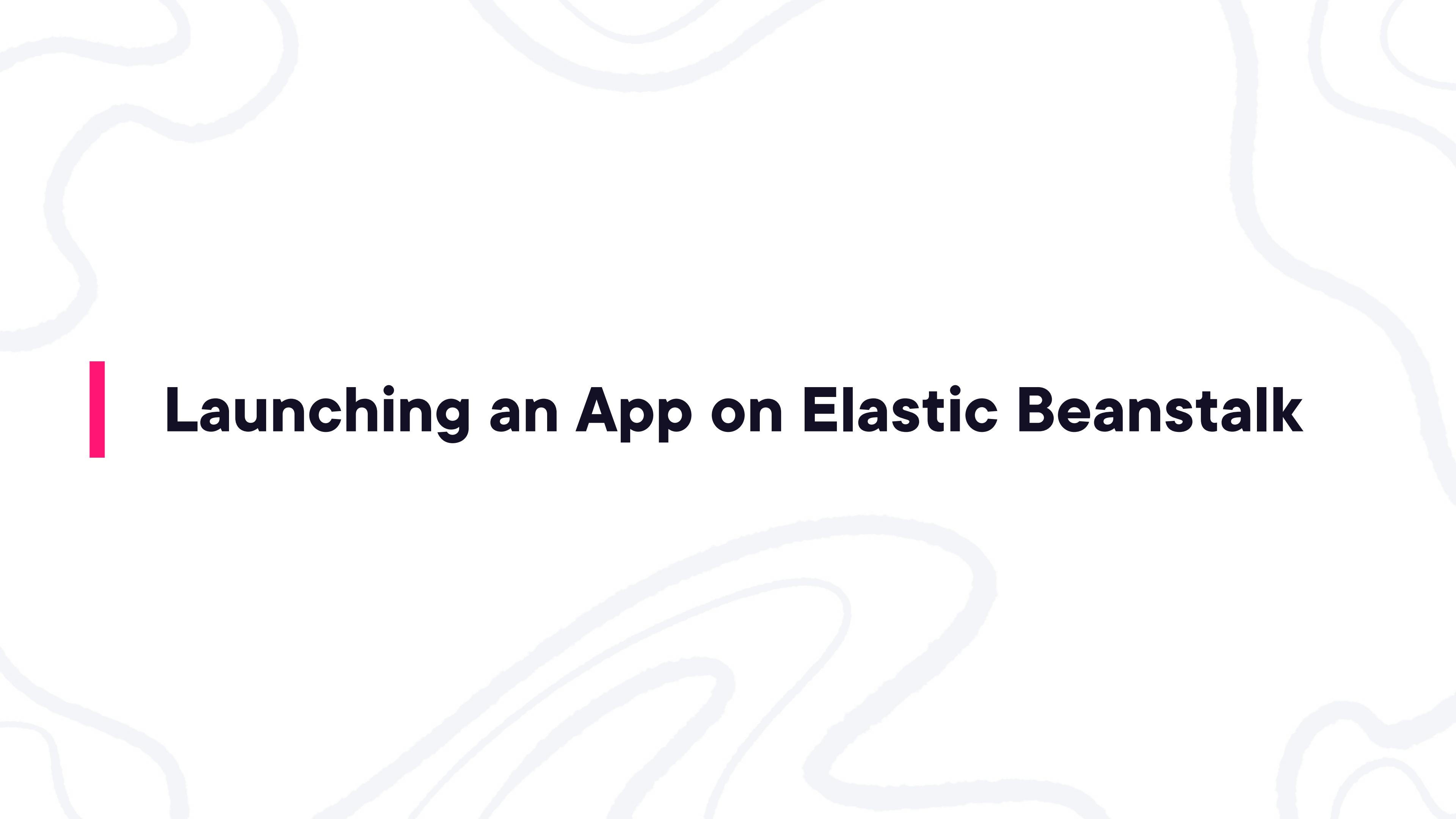
**EC2 Customization**

## **Use Cases**

**Deploy an application with minimal knowledge of other services**

**Reduce the overall maintenance needed for the application**

**Few customizations are required**



# Launching an App on Elastic Beanstalk

## Demo

**Accessing the sample Elastic Beanstalk applications**

**Launching a sample application on Elastic Beanstalk**

**Deleting a deployed Elastic Beanstalk application**



# AWS Lambda Overview

Amazon Web Services

**“AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume. You can run code for virtually any type of application or backend service - all with zero administration.”**

# AWS Lambda

**Enables the running of code without provisioning infrastructure**

**Only charged for usage based on execution time**

**Can configure available memory from 128 MB to 3008 MB**

**Integrates with many AWS services**

**Enables event-driven workflows**

**Primary service for serverless architecture**

**Reduced maintenance requirements**

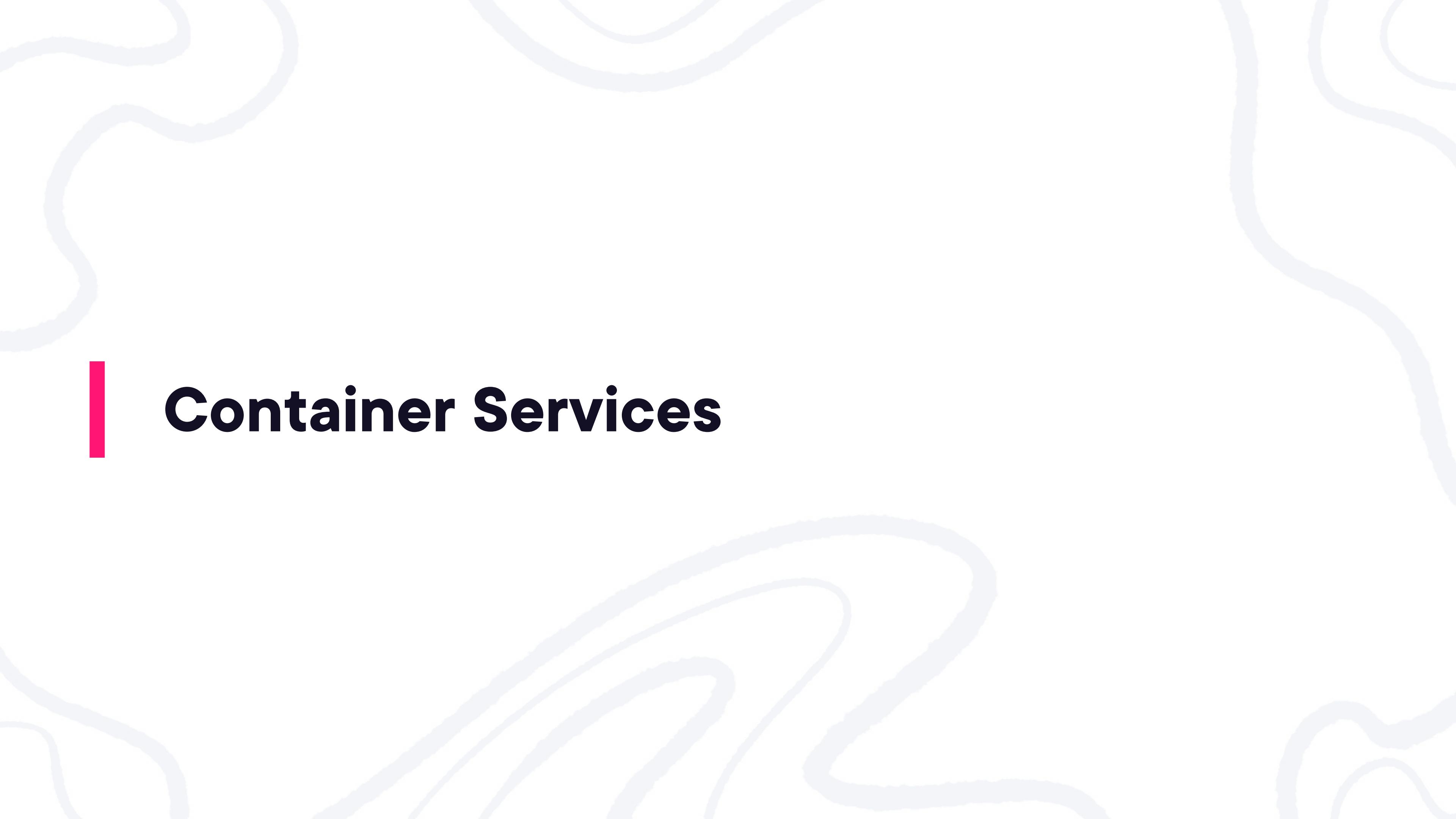
**Enables fault tolerance without additional work**

**Scales based on demand**

**Pricing is based on usage**

## **AWS Lambda Advantages**



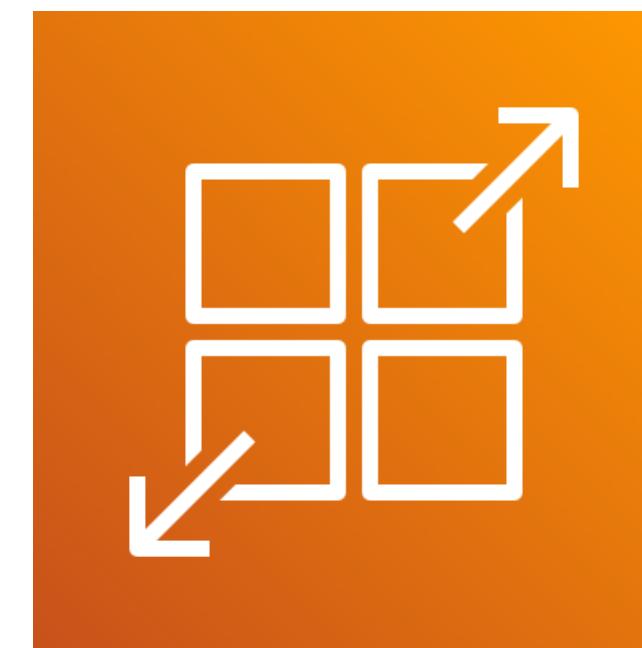


# **Container Services**

Google

**“Containers are packages of software that contain all of the necessary elements to run in any environment... containers virtualize the operating system and run anywhere, from a private data center to the public cloud or even on a developer’s personal laptop.”**

# Getting Started with Containers



**AWS App Runner**  
**Deploy web application or API  
containers without prior knowledge of  
container infrastructure**

# Container Orchestration Services

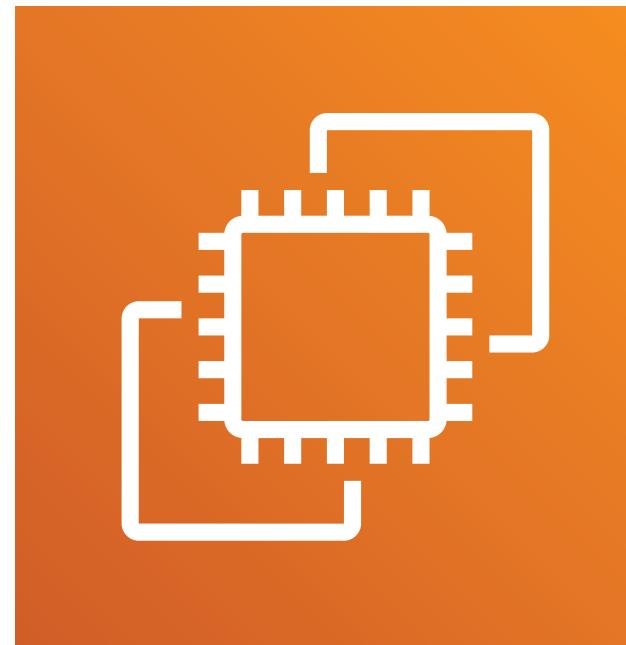


**Amazon ECS**  
**Full-featured container  
orchestration service**

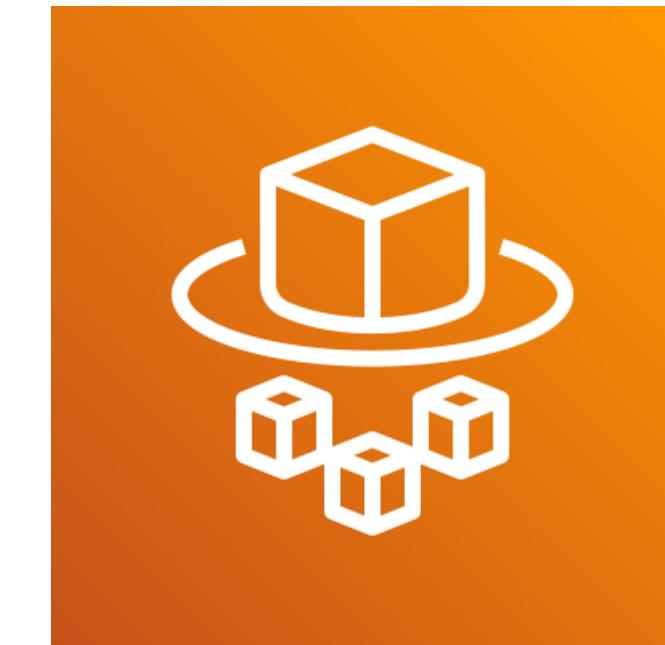


**Amazon EKS**  
**Kubernetes-based  
full-featured container  
orchestration service**

# Container Orchestration Compute Engines



**Amazon EC2**  
Container orchestration  
services can manage EC2  
instances directly



**AWS Fargate**  
Serverless compute engine for  
use with the container  
orchestration services

# Selecting a Container Service

**App Runner requires no prerequisite knowledge of containers or infrastructure**

**EKS will provide an easier starting point for Kubernetes in the cloud**

**ECS natively integrates with many AWS services**

**Fargate reduces the amount you have manage for container orchestration**

**EC2 provides maximum control for configuration of scaling with containers**



# **Scenario Review**

# Scenario 1



**Sylvia's company is in the process of moving multiple workloads into AWS**

**One workload is an application that will be leveraged for at least 5 more years**

**The organization is looking to be as cost efficient as possible for its EC2 usage**

**What EC2 purchase option should be chosen for this application?**

## Scenario 2



**Edward is looking to deploy his PHP web application to a virtual server**

**He doesn't have experience managing EC2 instances on AWS**

**He needs the ability to scale this application to meet user demand**

**What is the best compute option for Edward based on this criteria?**

## Scenario 3



**Cindy's company is transitioning to the cloud for its data processing workloads**

**These workloads happen daily and can start or stop without a problem**

**This workload will be leveraged for at least one year**

**What EC2 purchase option would be the most cost efficient choice?**



# Summary



## Summary

- Introduced Amazon EC2 capabilities
- Explored pricing approaches for EC2 instances
- Introduced the capabilities of AWS Elastic Beanstalk
- Reviewed use cases for Elastic Beanstalk
- Introduced AWS Lambda

# Scenario 1



**Sylvia's company is in the process of moving multiple workloads into AWS**

**One workload is an application that will be leveraged for at least 5 more years**

**The organization is looking to be as cost efficient as possible for its EC2 usage**

**What EC2 purchase option should be chosen for this application?**

**Solution:** All Upfront Reserved - 3 Years

## Scenario 2



**Edward is looking to deploy his PHP web application to a virtual server**

**He doesn't have experience managing EC2 instances on AWS**

**He needs the ability to scale this application to meet user demand**

**What is the best compute option for Edward based on this criteria?**

**Solution:** AWS Elastic Beanstalk

## Scenario 3



**Cindy's company is transitioning to the cloud for its data processing workloads**

**These workloads happen daily and can start or stop without a problem**

**This workload will be leveraged for at least one year**

**What EC2 purchase option would be the most cost efficient choice?**

**Solution:** Spot Instances