AWS Solution Architect

Container Service

A Day in the Life of a Cloud Architect

Mr. William works for an e-commerce company that is planning to launch its web application in a cloud-based environment so that it can run microservices on containers. The company wants Mr. William to propose a solution for the following requirements:

- The containers on which the microservices run should be monitored continuously to ensure the uptime of the server.
- The containers should automatically scale up and scale down depending upon the workload.

To achieve all the above, along with some other features, Mr. William will be learning concepts like AWS ECS, which will help him solve the above problem.

Learning Objectives

By the end of this lesson, you will be able to:

- Execute a Docker Container on EC2 instance
- Push and pull container images from Amazon ECR
- Create and manage container clusters using Amazon ECS
- Deploy container cluster using Amazon Fargate

TECHNOLOGY

Elastic Container Service

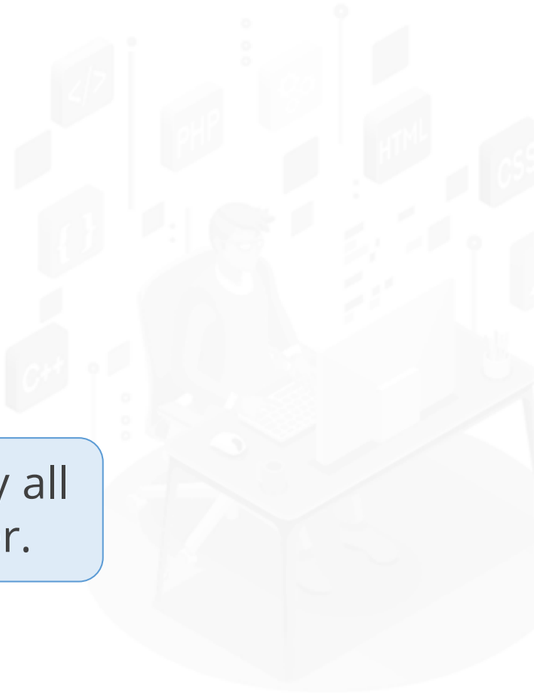
What Is a Container?

A container is a standardized unit of software development, containing everything that a software application needs to run, such as code, runtime, system tools, system libraries, and more.

More about containers:

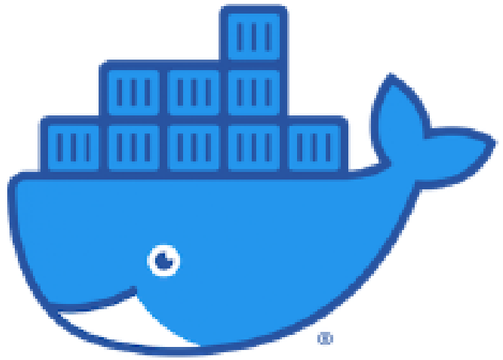
Containers are created from a read-only template called an image.

Images are text files that specify all the components of a container.



What Is a Docker?

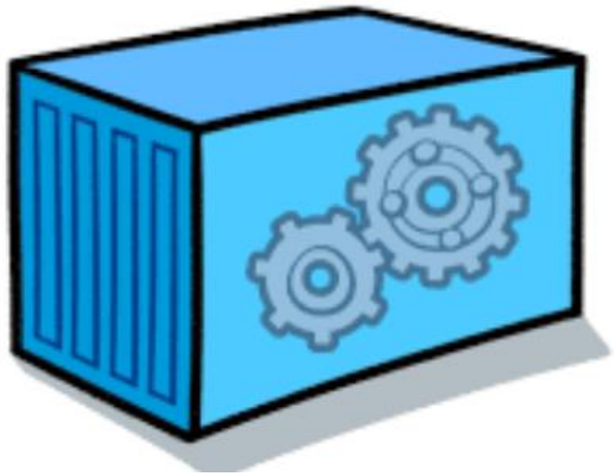
Docker is a tool to automate the deployment of an application as a lightweight container so that it can work efficiently in a different environment.



Docker container is a lightweight software package consisting of all the dependencies, such as **code**, **framework,** and **library**, required to run an application.

Advantages of a Docker Container

The advantages of Docker containers are as follows:



01 Docker containers are highly scalable and efficient.

02 It gets booted up quickly.

03 Docker containers are highly portable.

Run Hello World Container on EC2 Instance

Problem Statement:

Duration: 15 mins

You have been asked to run the hello world container on an EC2 instance.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Install PuTTY and PuTTYgen
2. Launch an EC2 instance
3. Access the EC2 instance via putty
4. Install Docker on the EC2 instance
5. Run the hello world container

Create Container Registry Using AWS ECR

Duration: 15 mins

Problem Statement:

You have been asked to create a container image repository using AWS ECR and push images into it.

ASSISTED PRACTICE

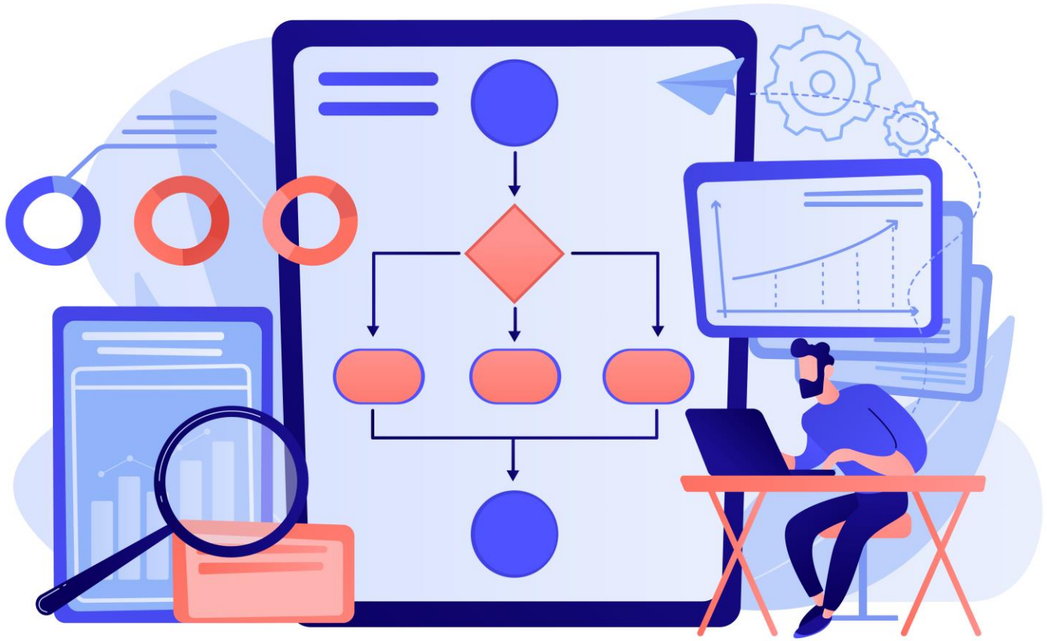
Assisted Practice: Guidelines

Steps to be followed:

1. Create ECR repository
2. Launch an EC2 Instance
3. Install docker on the EC2 instance
4. Push container image to the repository

What Is Container Orchestration?

Container orchestration automates the scheduling, development, networking, scaling, health monitoring, and management of the containers.



Container Orchestration Tools in AWS

AWS services that are used for container orchestration are as follows:

Amazon **ECS**

Amazon **EKS**

Source: docs.aws.amazon.com

Amazon Elastic Container Registry

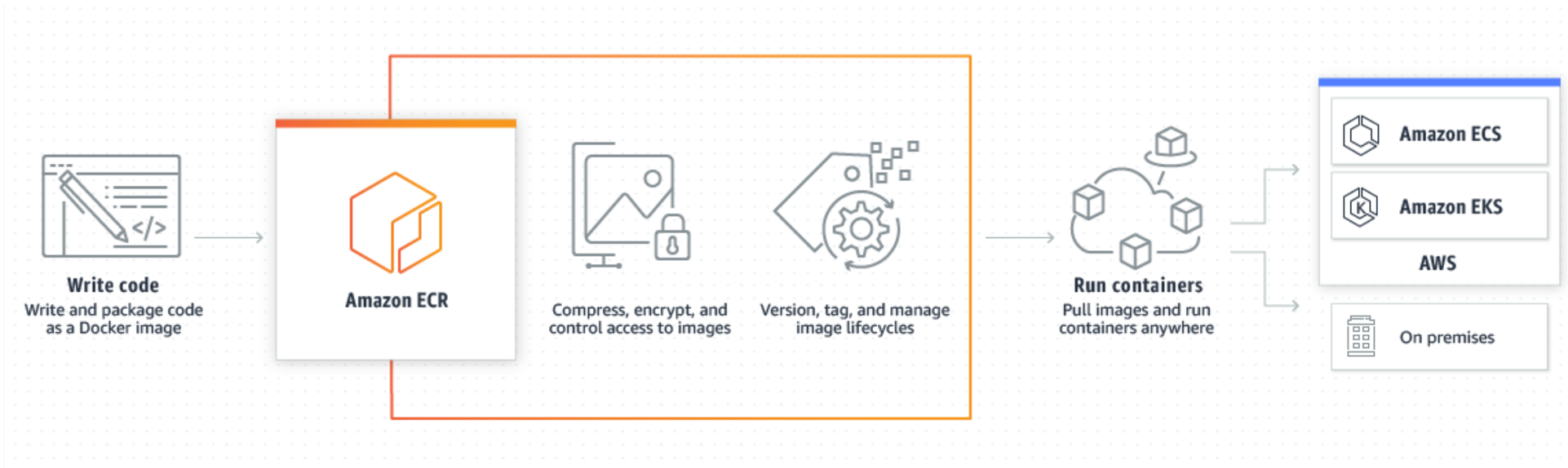
Amazon Elastic Container Registry (ECR) is a fully-managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images.

Amazon ECR is integrated with Amazon Elastic Container Service (ECS) to simplify the container deployment process.

Source: docs.aws.amazon.com

Amazon ECR Workflow

This image shows where the Amazon ECR is used in container deployment process:



Source: docs.aws.amazon.com

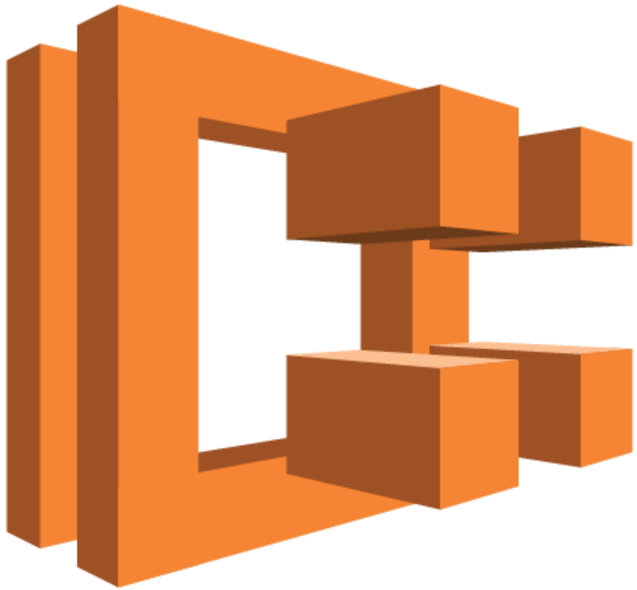
TECHNOLOGY

Introduction to Elastic Container Service

simplilearn

Amazon Elastic Container Service

Amazon Elastic Container Service **Amazon ECS** is a highly scalable and fast container management service, which is used to deploy and manage containers on a cluster.



ECS is deeply integrated with the rest of the AWS platform to provide a secure and easy-to-use solution for running container workloads in the cloud.

Source: docs.aws.amazon.com

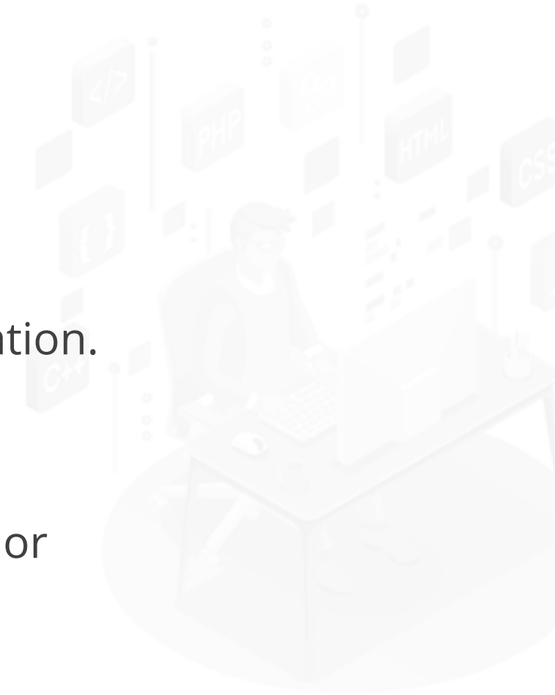
Features of ECS

01 **Docker Support**: Amazon ECS supports Docker, which enables users to run and manage Docker containers.

02 **Programmatic Control**: Amazon ECS allows users to integrate and extend their service through API calls.

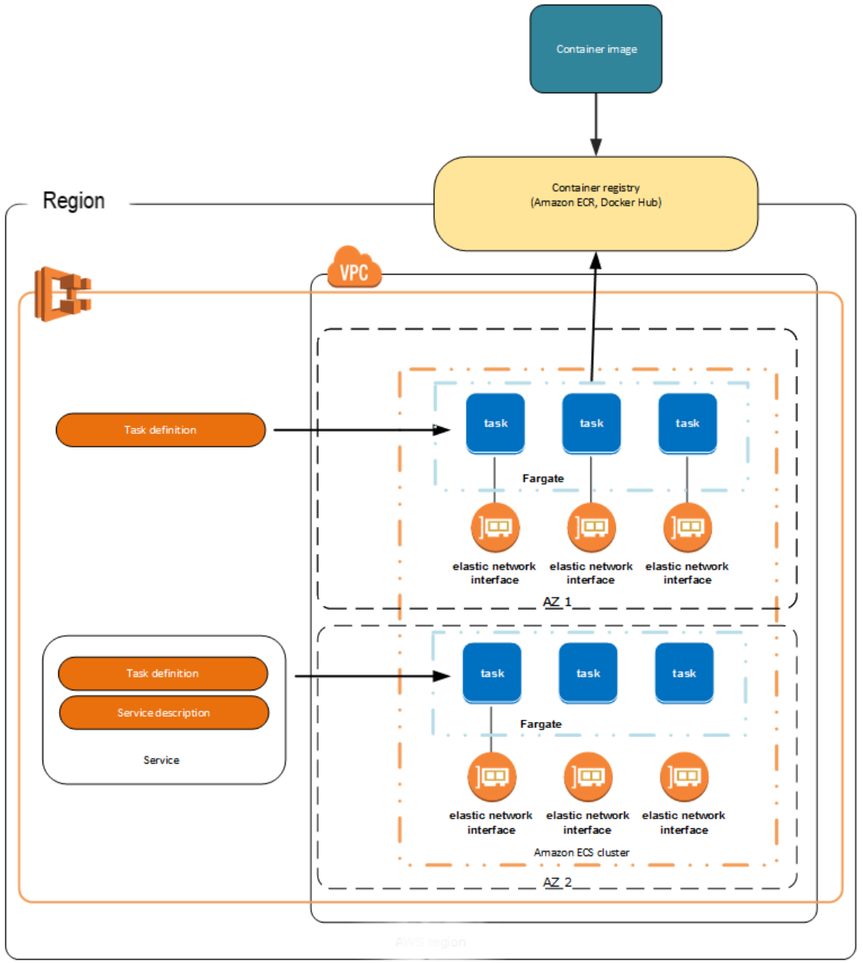
03 **Container Auto-Recovery**: Amazon ECS automatically recovers failed containers to maintain the availability and dependability of the application.

04 **Scheduling**: Amazon ECS can schedule clusters based on a variety of needs, such as configuring clusters based on resource needs like CPU or RAM.



ECS Architecture

The image below shows how Amazon ECS uses AWS Fargate to run containers:



Source: docs.aws.amazon.com

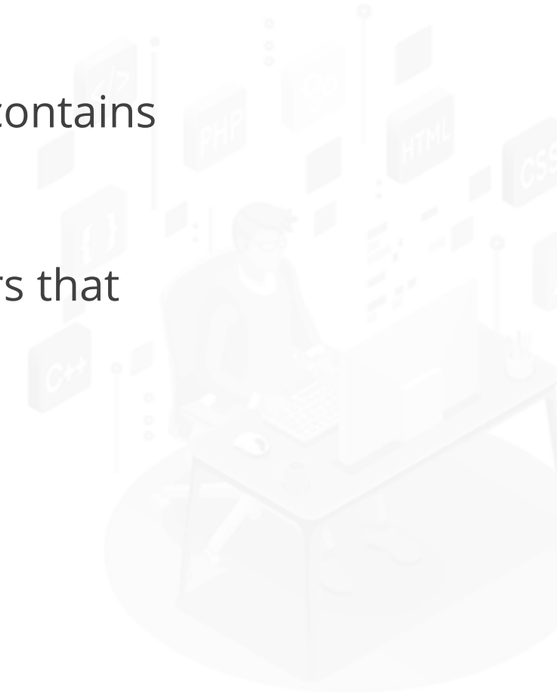
Components of ECS

01 **Cluster**: Logical grouping of tasks or services is known as ECS cluster.

02 **Container**: It is a standardized piece of software development that contains all the components needed for a software application to function.

03 **Task definition:** It is a text file that describes one or more containers that make up an application, and it is in JSON format.

04 **Task:** It is the instantiation of a task definition within a cluster.



Components of ECS

05 **Service:** It is to run and maintain desired number of tasks simultaneously in an Amazon ECS cluster.

06 **Container agent:** The container agent runs on each container instance within an Amazon ECS cluster, and it sends information about the current running tasks and resource utilization of the container to Amazon ECS.

Elastic Container Service

ECS can be launched in the following modes:

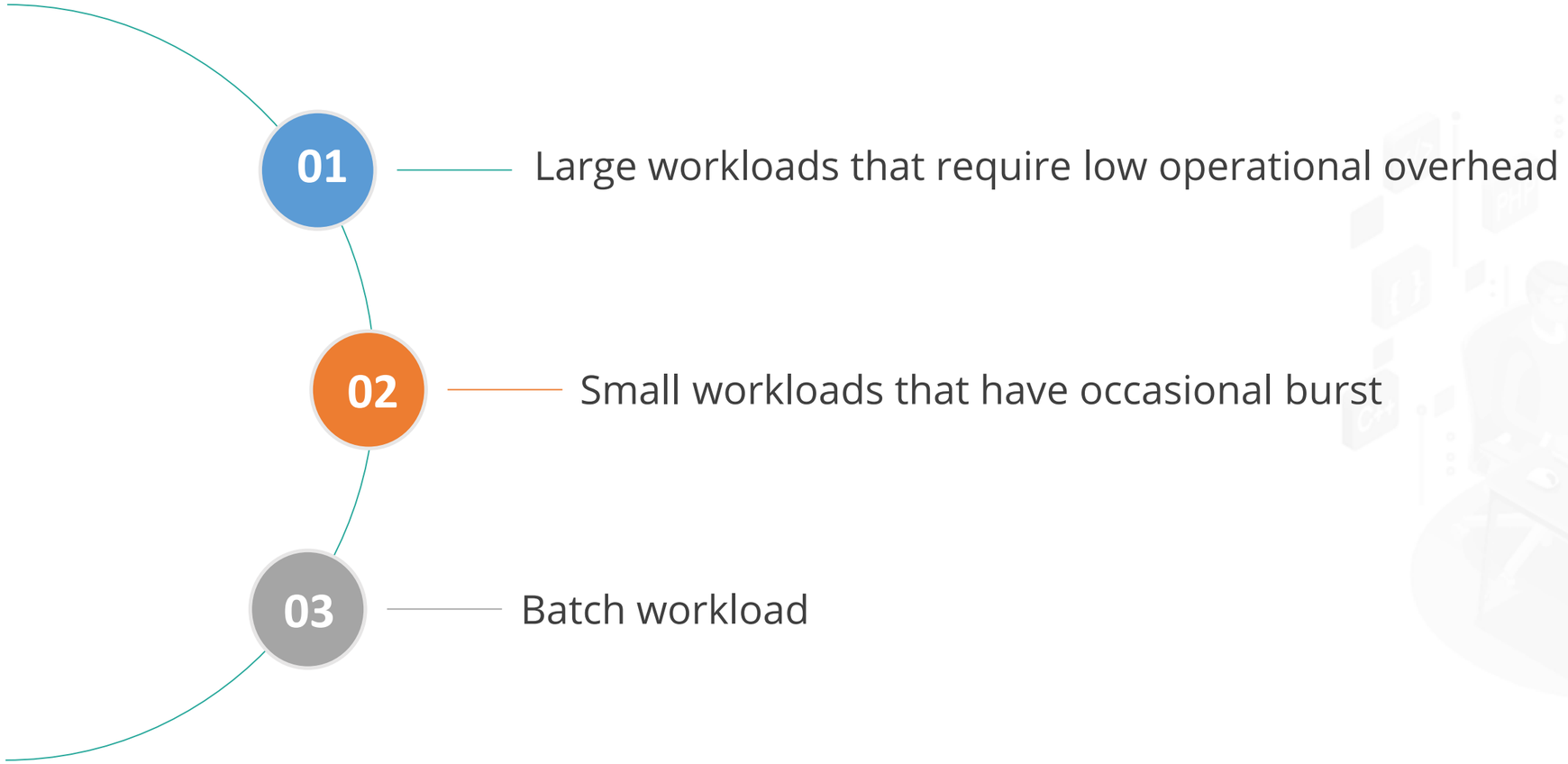
Fargate launch

EC2 launch

Source: docs.aws.amazon.com

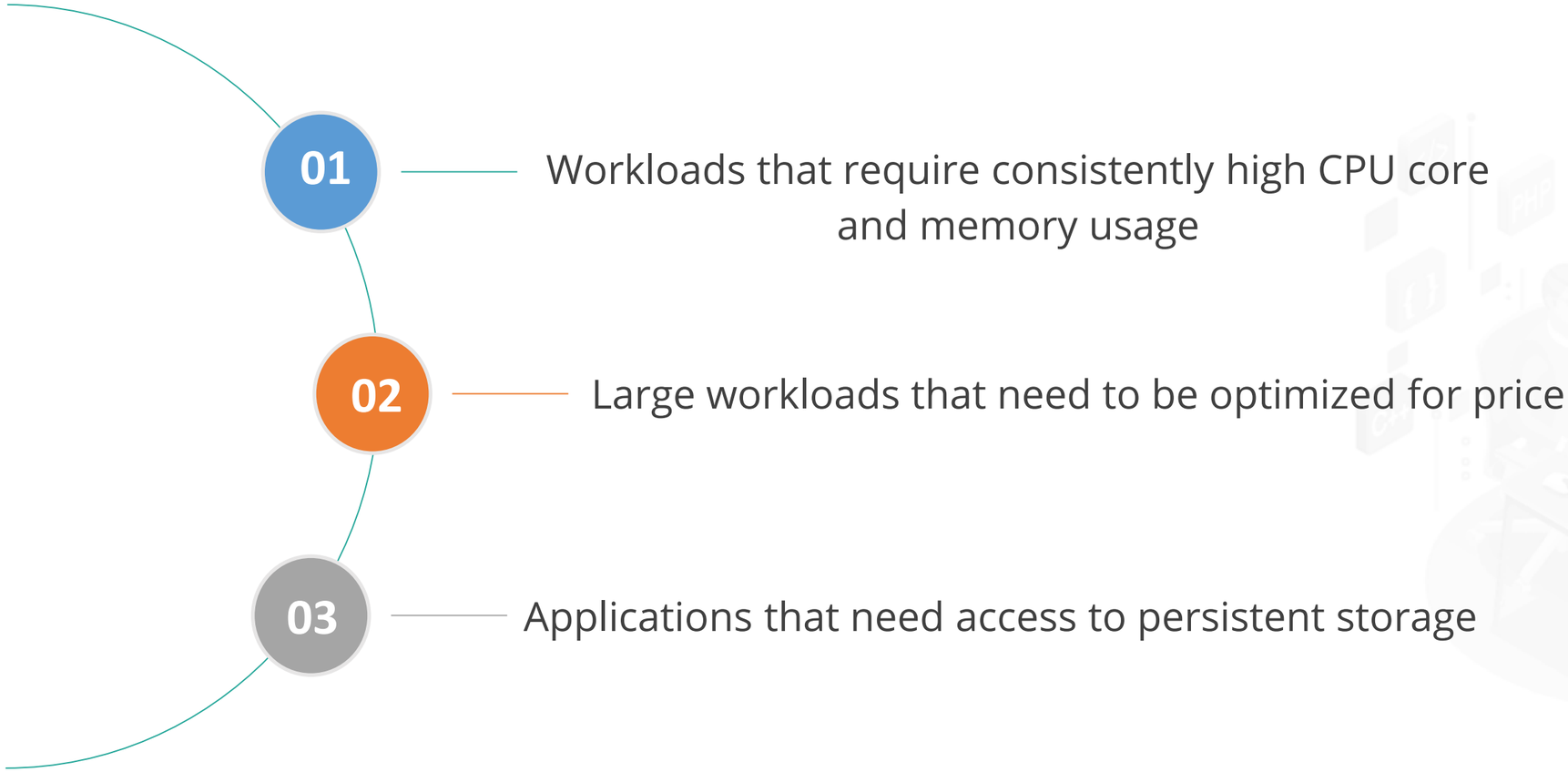
When to Use Fargate Launch Type?

The following workloads are suitable for the **Fargate** launch type:



When to Use EC2 Launch Type?

The following workloads are suitable for the **EC2** launch type:



How to Access ECS?

ECS can be accessed from the following interfaces:

01 AWS management console

02 AWS Command Line Interface (AWS CLI)

03 AWS Copilot

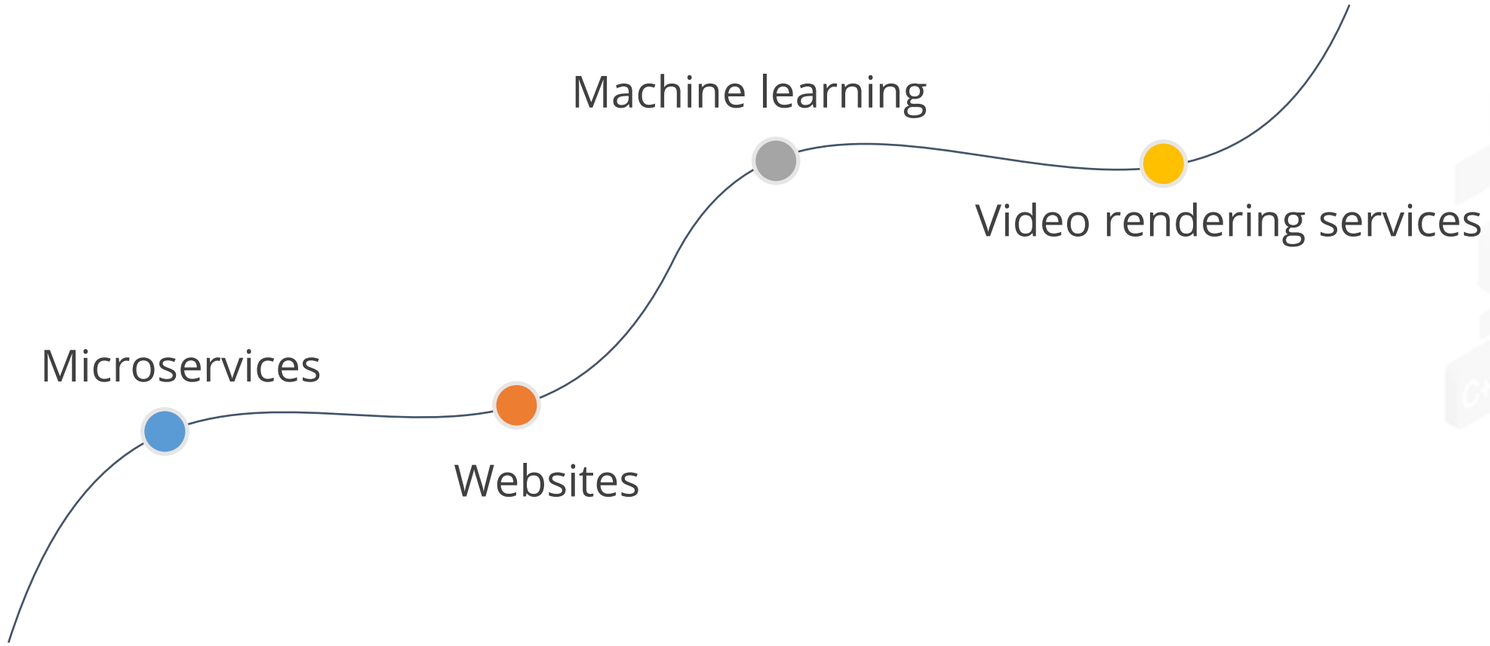
04 Amazon ECS CLI

05 AWS CDK

06 AWS SDK

Use Cases

ECS use cases are as follows:



Companies Using Amazon ECS

Amazon ECS is used by the following companies:

Source: docs.aws.amazon.com

Create an EC2 cluster and a task definition then run the task

Duration: 15 mins

Problem Statement:

You have been asked to create an EC2 cluster and a task definition then run that task using Amazon ECS.

ASSISTED PRACTICE

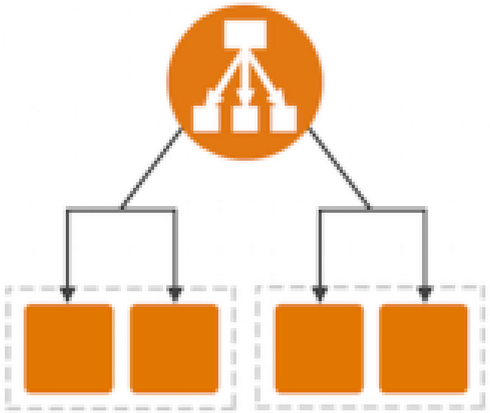
Assisted Practice: Guidelines

Steps to be followed:

1. Create a cluster
2. Create a task definition
3. Run a task from that task definition on the cluster

What Is Dynamic Port Mapping?

Dynamic port mapping is a feature of an application load balancer through which users can run various tasks from a single service on the same container instance.



Dynamic port mapping can also be done with network load balancers.

Source: docs.aws.amazon.com

Setting Up Dynamic Port Mapping

Steps to set up dynamic port mapping are as follows:

Step 1 Create an **Application Load Balancer** and a **target group**

Step 2 Open the Amazon ECS console and then set the host port to 0 for the task definition that needs to be created or updated

Note: Ensure that the container port mappings is set for the application

Step 3 Add a rule to allow inbound traffic from the load balancer to the container instances

Note: The security group and **network access control list network ACL** must allow traffic from the load balancer to the instances over the ephemeral port range.

Setting Up Dynamic Port Mapping

Steps to setup dynamic port mapping are as follows:

Step 4	Go to **ECS console** and then configure the service to use the Application Load Balancer created
Step 5	Go to the **EC2 console** and then select **Target Groups** from the navigation pane
Step 5	Choose the hyperlinked name for the target group and from the **Targets tab**, users can see what port is used for the task in the service that is created

If dynamic port mapping is set up correctly, the registered targets will be visible in the target group and the assigned port for the task.

Dynamic Port Integration with Application Load Balancer

Duration: 15 mins

Problem Statement:

You have been asked to enable dynamic port mapping in a container.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Create a custom VPC and enable DNS hostname
2. Create Internet Gateway
3. Create three subnets
4. Create a Route table and attach it to three subnets
5. Create a Target group
6. Create an application load balancer
7. Create a Cluster
8. Create a Task definition
9. Run the task on the cluster

TECHNOLOGY

Introduction to AWS Fargate

simplilearn

AWS Fargate

Fargate is a service provided by AWS that serves as the compute engine for ECS.

AWS Fargate is compatible with both **Amazon ECS** and **Amazon EKS**.

Source: docs.aws.amazon.com

Advantages of AWS Fargate

01 **Easy management:** Fargate eliminates the requirement for server management.

02 **Auto-scaling:** It can automatically scale the cluster up or down in response to changing demand.

03 **Health checks:** It can monitor the health of the containers and automatically replace the faulty one.

04 **Task definitions**: It allows users to define the tasks they want to run and the environment in which they should run.

Use Cases of AWS Fargate

AWS Fargate

01 Fargate is used in **web apps**, **APIs**, and **microservices**.

02 It is used in **running** and **scaling** of **container workload.**

03 It is used in **AI** and **ML** applications.

Run Task on a Fargate Cluster

Duration: 15 mins

Problem Statement:

You have been asked to run a task on a Fargate Cluster.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Create a Fargate cluster
2. Create a task definition
3. Run its service

Fargate with an Application Load Balancer

Duration: 15 mins

Problem Statement:

You have been asked to deploy tomcat by using Fargate cluster with an application load balancer.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Create a Fargate cluster
2. Create a task definition
3. Set up Load Balancer for the task
4. Run a new service on the cluster

Key Takeaways

- Containers are software packages that come with everything needed to execute an application in any environment.
- Docker is a containerization platform that enables users to package applications into containers.
- The deployment, management, scaling, and networking of containers are all automated through container orchestration.
- Amazon ECS is a container orchestration service that supports Docker containers and allows users to easily run and scale containerized applications on AWS.
- Amazon Fargate is a serverless container platform for running Docker containers.

Monitoring Insights for ECS Cluster Using CloudWatch

Duration: 30 mins

Project agenda: To monitor and gain container insights of an Amazon ECS Cluster using CloudWatch

Description:

Your company is going to deploy a container application using ECS Clusters. As a Cloud architect, you should be able to monitor the cluster's health by measuring various metrics such as CPU utilization, memory utilization, task count, and so on.

Perform the following:

1. Monitor the Cluster using CloudWatch container insights

LESSON-END PROJECT

TECHNOLOGY

Thank You

simplilearn