

Service Name

Amazon Elastic Container Service (ECS)

Service Overview

Amazon Elastic Container Service (Amazon ECS) is a fully managed container orchestration service that helps you easily deploy, manage, and scale containerized applications. As a fully managed service, Amazon ECS comes with AWS configuration and operational best practices built-in. It's integrated with both AWS tools, such as Amazon Elastic Container Registry, and third-party tools, such as Docker. This integration makes it easier for teams to focus on building the applications, not the environment. You can run and scale your container workloads across AWS Regions in the cloud, and on-premises, without the complexity of managing a control plane.

Key Use Cases

- **Web applications and APIs:** Long-running stateless applications that require high availability and automatic scaling
- **Batch processing:** Running containerized batch jobs for data processing, ETL operations, and computational workloads
- **Microservices architecture:** Breaking down applications into smaller, independently deployable services
- **Hybrid deployments:** Running containerized workloads across AWS and on-premises infrastructure using ECS Anywhere
- **Modernizing legacy applications:** Containerizing existing applications to improve portability and scalability
- **Machine learning workloads:** Running ML training and inference workloads with GPU acceleration support

Features

- **Task Definitions:** Blueprint for applications that specifies container configurations, resource requirements, and networking
- **Clusters:** Logical grouping of compute resources where containers run
- **Services:** Long-running stateless applications with automatic scaling and health monitoring
- **Tasks:** Single instances of applications such as batch jobs that perform work and then stop
- **Auto Scaling:** Automatic scaling of both container instances and services based on demand
- **Service Connect:** Service discovery and connectivity with traffic monitoring for inter-service communication

- **Blue/Green Deployments:** Built-in deployment strategy for safer software updates with zero downtime
- **ECS Anywhere:** Extends ECS to run containers on on-premises infrastructure
- **Express Mode:** Simplified deployment that automatically provisions infrastructure components
- **Multiple Launch Types:** Support for Fargate serverless, EC2 instances, and ECS Managed Instances

Value Proposition

Amazon ECS provides a fully managed container orchestration experience that eliminates infrastructure management overhead while maintaining flexibility and control. Unlike self-managed Kubernetes, ECS integrates seamlessly with AWS services and provides built-in security, monitoring, and scaling capabilities:

- It offers multiple compute options from serverless Fargate to EC2 instances, allowing customers to optimize for cost and performance
- ECS reduces operational complexity through features like Express Mode for rapid deployment, automated infrastructure optimization with Managed Instances, and native blue/green deployments for safe software updates

Service Integration

Amazon ECS integrates deeply with core AWS services to provide a comprehensive container platform. It works with Amazon ECR for container image storage, Elastic Load Balancing for traffic distribution, and AWS Cloud Map for service discovery:

- VPC integration provides secure networking, while IAM enables fine-grained access control
- CloudWatch offers monitoring and logging capabilities, and AWS Systems Manager supports configuration management
- ECS also integrates with AWS Fargate for serverless compute, Amazon EC2 for traditional instances, and supports hybrid deployments through integration with AWS Outposts and on-premises infrastructure via ECS Anywhere

Onboarding and Offboarding

Customers can get started with Amazon ECS through multiple approaches. The Express Mode provides the fastest path by automatically provisioning all required infrastructure with a single command:

- Traditional setup involves creating clusters, defining task definitions, and configuring services through the AWS Management Console, CLI, or Infrastructure as Code tools like CloudFormation and CDK

- AWS provides comprehensive tutorials, including hands-on labs with AWS Copilot for simplified container deployment
- The service supports gradual adoption, allowing teams to start with simple applications and expand to complex microservices architectures as their expertise grows

Getting Started Guide:

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/Welcome.html>

Data Removal

To offboard from Amazon ECS, customers delete services, task definitions, and clusters through the console or API. When deleting services, ECS automatically scales them to zero tasks before removal. Container instances are deregistered, and external instances (ECS Anywhere) require cleanup of AWS resources including stopping ECS and Systems Manager services. Customers should remove ECR repositories, load balancers, and related CloudFormation stacks. ECS provides automated cleanup features for stopped tasks and unused Docker images on container instances.

Backup/Restore and Disaster Recovery

Amazon ECS supports disaster recovery through multi-AZ deployments and cross-region replication strategies. Container images stored in Amazon ECR can be replicated across regions. Task definitions and service configurations can be backed up using Infrastructure as Code. For stateful applications, customers should implement backup strategies for persistent data using EBS snapshots or database replication. ECS integrates with AWS Backup for centralized backup management of associated resources.

Backup Capabilities

Amazon ECS does not have native backup capabilities as it primarily manages stateless containers. However, it integrates with AWS Backup for backing up associated resources like EBS volumes attached to container instances:

- Container images are stored in Amazon ECR with cross-region replication support
- Service configurations and task definitions can be version-controlled and backed up through Infrastructure as Code tools like CloudFormation and CDK

Disaster Recovery

Amazon ECS supports disaster recovery through architectural patterns rather than direct integration with AWS Elastic Disaster Recovery. Customers implement DR using multi-region deployments, where ECS services can be replicated across regions:

- ECS works with Application Load Balancers and Route 53 for traffic failover
- The service's stateless nature facilitates quick recovery by launching new tasks in healthy regions when failures occur

Recovery Objectives

Amazon ECS helps achieve low RPO and RTO objectives through its distributed architecture and integration with AWS services. Multi-AZ deployments provide sub-minute RTO for availability zone failures. Cross-region deployments can achieve RTOs of minutes with proper automation. RPO depends on data replication strategies for persistent storage, with ECS supporting real-time replication through database services and storage solutions. Auto Scaling and health checks enable rapid detection and recovery from failures.

Service Constraints

Service Quotas:

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/service-quotas.html>

FAQ: <https://aws.amazon.com/ecs/faqs/>

Technical Requirements

Service Documentation:

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/>

User Guide:

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/Welcome.html>

API Reference: <https://docs.aws.amazon.com/AmazonECS/latest/APIReference/>

Pricing Overview

Please see the AWS UK G Cloud 15 Pricing Document accompanying this service in the Digital Marketplace.

Public Pricing: <https://aws.amazon.com/ecs/pricing/>

Cost Optimization

Amazon ECS offers several unique cost optimization features including AWS Fargate Spot for up to 70% savings on interruptible workloads, ECS Managed Instances that automatically optimize infrastructure by right-sizing and consolidating tasks, and intelligent capacity management that removes idle instances. Compute Optimizer provides recommendations for ECS services running on Fargate:

- Customers can mix Fargate and EC2 launch types within the same cluster, use Savings Plans for predictable workloads, and leverage Auto Scaling to match capacity with demand
- Express Mode automatically shares Application Load Balancers across multiple services to reduce costs

Savings Considerations

Primary cost savings come from choosing the right compute option for workloads: Fargate for variable workloads eliminates unused capacity costs, while EC2 instances with Reserved Instances or Savings Plans provide savings for steady-state workloads. ECS Managed Instances optimize costs through automatic instance right-sizing and task consolidation:

- Using Fargate Spot can reduce costs by up to 70% for fault-tolerant applications
- Proper resource allocation through CPU and memory optimization prevents over-provisioning
- Auto Scaling prevents paying for unused capacity during low-demand periods, while shared load balancers in Express Mode distribute infrastructure costs across multiple services