

Deploy Operations Ready Containerized applications based on ECS and Fargate clusters using MontyCloud's DAY2™

Customers use containers for modern cloud applications because containers make it easy to develop, promote and deploy code consistently across different environments. But managing thousands of containers is no simple task.

Amazon Elastic Container Service and AWS Fargate help customers scale their container clusters efficiently. However, as applications grow, container environments become complex and hard to manage. For example, the infrastructure team at a large business conglomerate was spending 2-3 hours to scale up or down a cluster.

In this paper we share how MontyCloud helps infrastructure and application teams deliver highly available, scalable and operations ready container applications in just a few clicks. With MontyCloud DAY2™ customers can avoid cost overruns and have containerized applications running in hours to days.



Manage & automate your
Cloud operations like an expert

aws partner
network

Advanced
Technology
Partner

Cloud Management
Tools Competency
Public Sector Partner

Large business conglomerate was challenged to meet routine operational needs

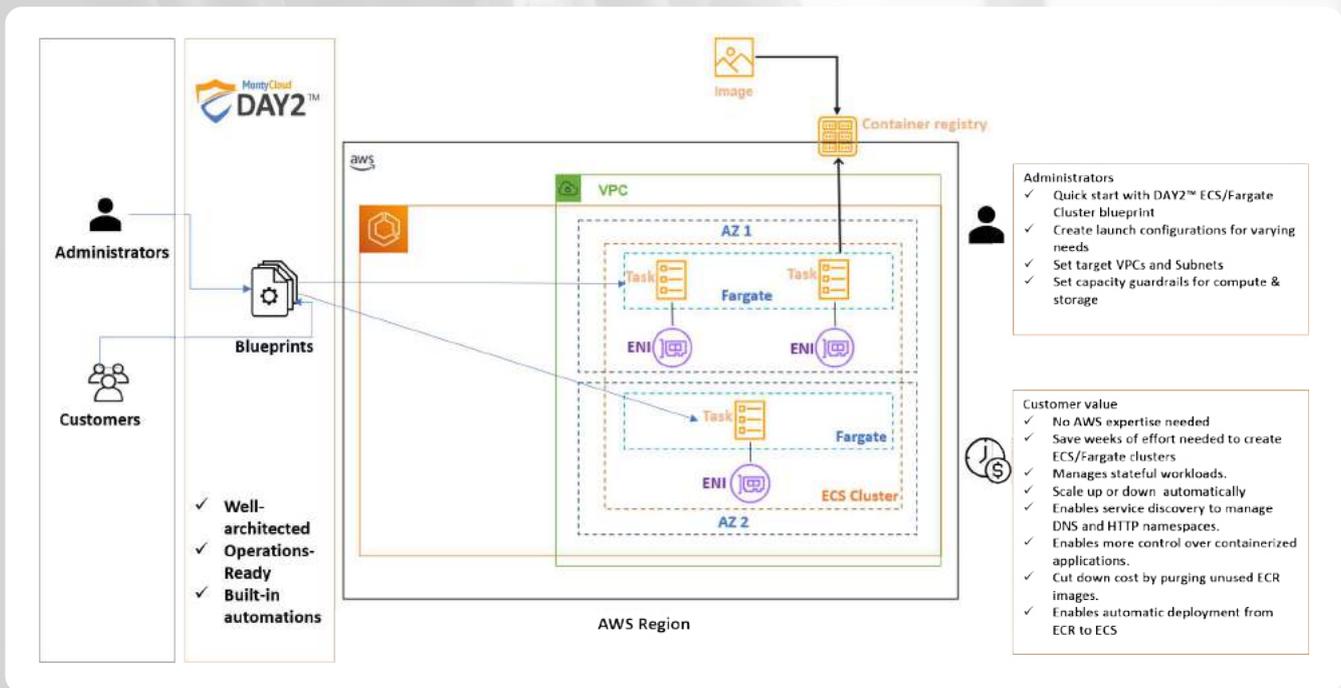
A large business conglomerate with multiple lines of business containerized two of their consumer brands. Both the applications are 3-tier web applications. The applications are written in Go with a React JS front end and a Cockroach DB database tier. Initially the customer used Docker compose to deploy and manage the application. As the application grew, the customer had to configure and run their own service discovery system and connect every service to a load balancer. Delivering highly available deployments required additional operations such as checking container health and replacing unhealthy containers with new ones, exposing select ports and load balancing. At one point the customer was using a number of distinct tools for their routine operations. The infrastructure team was always reacting to meet basic operational needs such as scaling up and down the cluster nodes or delivering persistent data storage.

Amazon ECS helps manage containers at scale but requires deep expertise

The customer moved to Amazon Elastic Container Service (ECS) and AWS Fargate (Fargate) to scale more efficiently. To use ECS and Fargate customers also need deep expertise in several AWS native services including AWS Identity and Access Management (IAM), Amazon Key Management Service (KMS), Amazon Virtual Private Cloud (VPC), ECS task definitions, AWS load balancer and AWS security services. In addition, customers also have to enable different components such as service discovery, choosing appropriate networking mode in a container cluster for ECS tasks to communicate with one another. It can take several days to weeks for teams to build and use ECS and Fargate clusters.

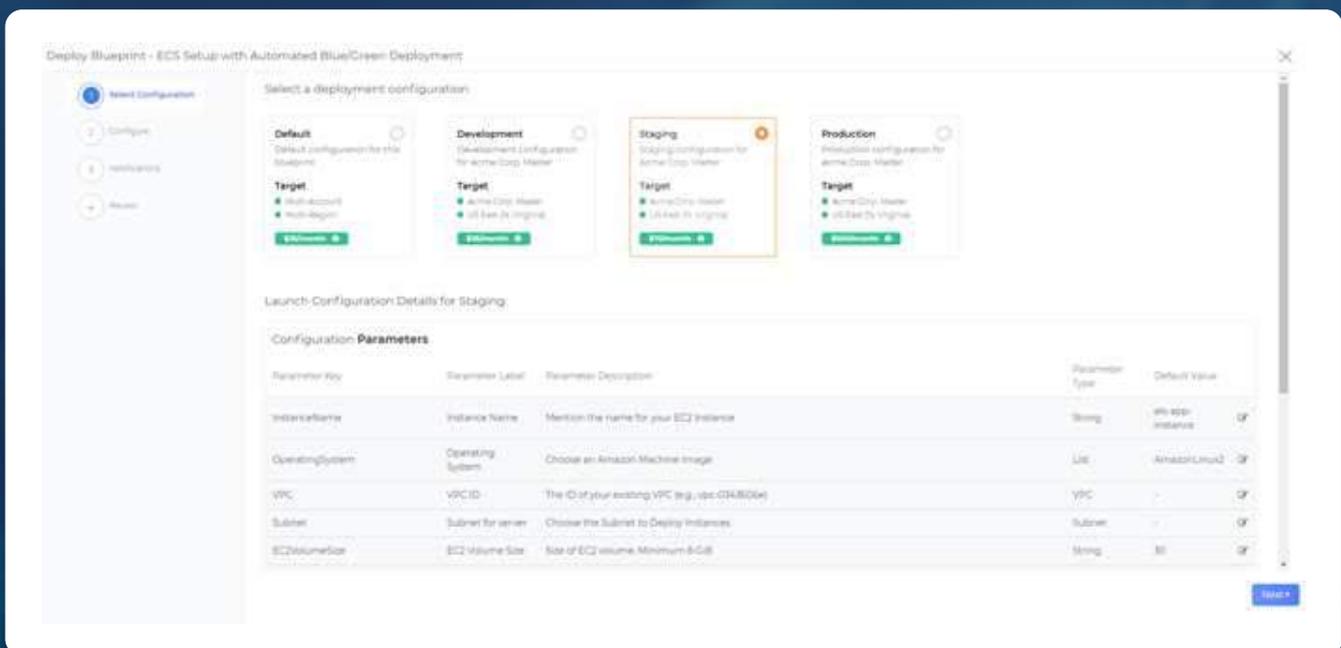
From 3 hours per container node to a few minutes with MontyCloud DAY2™

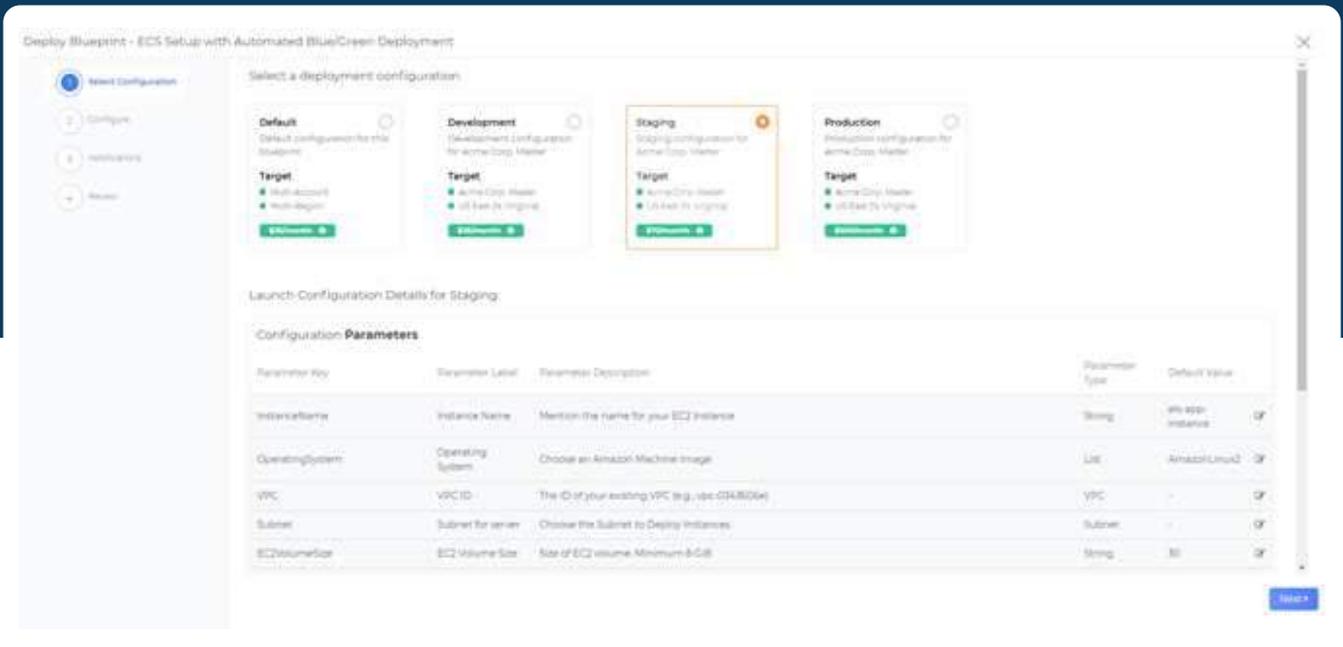
The customer's infrastructure team now uses the DAY2™ ECS and Fargate Blueprints to provision ECS clusters. The blueprints help the infrastructure team to self-service provision consistent operations ready container environments. Further, DAY2™'s No-Code Autonomous CloudOps automatically takes care of routine operations such as discovery of containerized services, and maintenance of persistent volumes. As a result, the customer reduced 2-3 weeks of work per node deployed to under few minutes.



Introducing MontyCloud DAY2™ ECS and Fargate Blueprints

DAY2™ ECS and Fargate Blueprints are operations ready well-architected templates that helps infrastructure teams deploy and instantly manage highly scalable, fast, serverless application environments.





Well-architected infrastructure as code deployment templates

1

DAY2™ ECS/Fargate Blueprint manages HTTP and DNS namespaces for their Amazon ECS/Fargate services. Now, new services are registered and old/unhealthy services are deregistered automatically and this enables ECS tasks to communicate with each other without any additional configuration. Customers no longer have to configure, run and maintain separate discovery systems such as Consul, etcd and ZooKeeper. Depending on the cluster size and complexity, this can save customers tens of hours of effort or more every month.

2

DAY2™ Blueprint comes with CNCF recommended container-native “aws-vpc” network mode to simplify container networking where tasks are allocated its own elastic network interface (ENI) and a primary private IPv4 address. Customers now have more control over how containerized applications (ECS tasks) communicate within their VPCs.

3

Infrastructure teams can now optionally enable Application Load Balancer (internet-facing or internal) to distribute traffic evenly across ECS tasks. Docker based deployments on ECS/Fargate are now simple and automatically enabled with enterprise class capabilities like service discovery, health checks, and load balancing.

4

With DAY2™ ECS/Fargate Blueprints, customers can pass environment variables to specify the startup command for a docker image. Most of the time applications requires some environment variables such as connection strings and sensitive API keys to be passed in the right order. The DAY2™ ECS/Fargate Blueprints enables injects the variables in the right format and sequence. IT teams no longer have to write custom code to handle the environment variables.

Data persistence for serverless applications

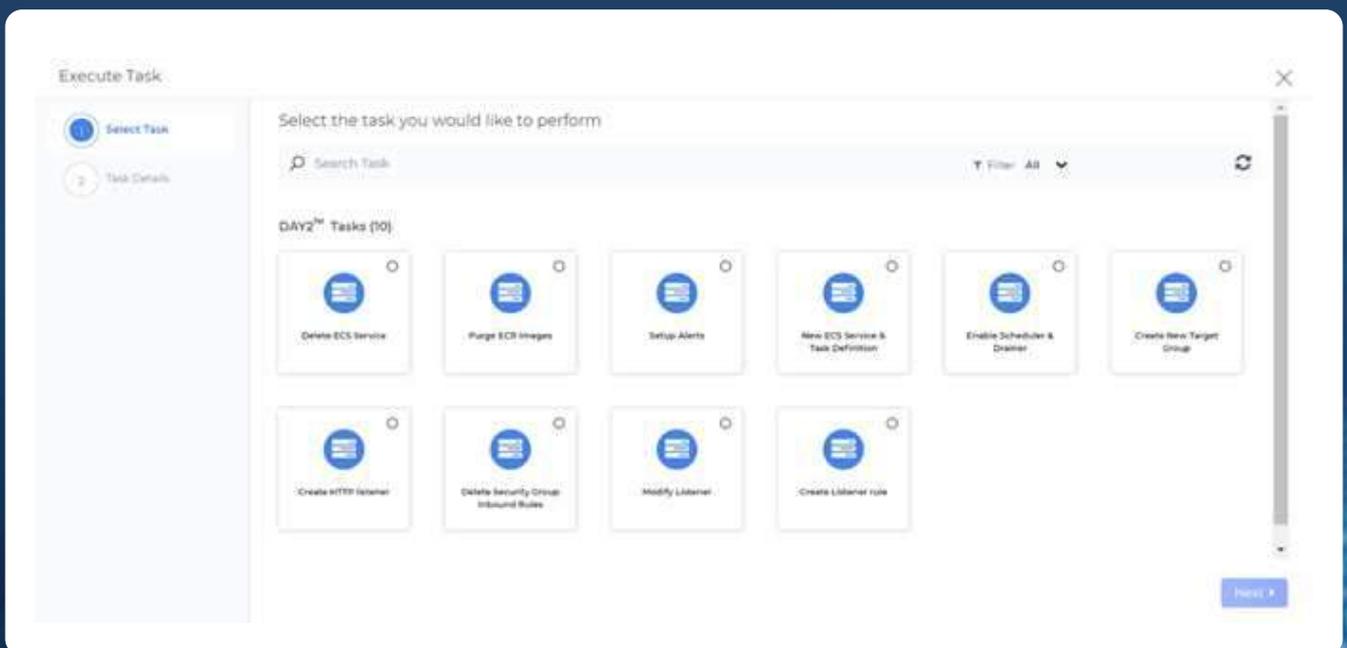
DAY2™ ECS/Fargate Blueprint provides the option to create stateful services within Fargate (or EC2) by integrating with Amazon Elastic File System (EFS). Infrastructure admins no longer have to manually mount the volumes. All they have to do is simply point to the EFS volume in the task definition. Infrastructure admins also have the option to Bind Mount and provide their application teams with persistent data volumes for use with containers.

Fast roll out of global updates in minutes

With the DAY2™ ECS/Fargate Blueprint application teams can setup complete, end-to-end continuous deployment (CD) pipeline for Amazon ECS/Fargate by integrating it with AWS CodePipeline. Now your applications is automatically deployed each time a new ECR image is pushed. This enables applications teams to quickly get new features to their users faster through a continuous delivery pipeline.

Performance at scale

With the DAY2™ ECS/Fargate Blueprint infrastructure teams can now pre-configure to auto scale services based on AWS CloudWatch alarms. This enables application teams to scale-in or scale-out based cluster nodes that conform to IT standards on demand.



The DAY2™ ECS/Fargate Blueprint comes with a built-in task library. You can customize and pre-configure your ECS cluster deployment and automate routine management and operations tasks. These tasks include:

- 1 — Create ECS Service or tasks**

Create different ECS services such as a web application consisting of a frontend, a search service, caching service or any other type of services based on your application's requirement with just a few clicks. DAY2™ automatically creates the ECS Task definition and AWS CodePipeline as part of service creation.
- 2 — Delete an ECS Service**

You can delete a ECS service or can schedule it at a certain time. All you have to do is mention the ECS service name. The service is automatically scaled down to zero. If you have a load balancer or service discovery resources associated with the service, they are not affected by the service deletion.
- 3 — Manage and purge ECR Images**

The DAY2™ ECS/Fargate Blueprint enables you to keep a predetermined number of images in your ECR repositories. You can also automatically remove stale or unused images using lifecycle policies. For example, you can create a policy to keep your latest 10 images and automatically lifecycle delete the older images. Your ECR repositories to be better organized, making it easier to find the code revisions that matter, and lowers storage costs.
- 4 — Enable Scheduler & Drainer**

During a cluster update, the Auto Scaling group terminates all old instances. If any ECS tasks are running on the deleted instances, they are all abruptly stopped, causing a micro-outage. DAY2™ uses Chaos lambda to deliberately inject faults into the container stack to make it resilient at deployment. In addition, you can also pre-configure your container cluster to first set the state to DRAINING and inject a lag for orderly shifting of active tasks to other containers using the ECS scheduler and the shutdown of other services, before an update.
- 5 — Configure AWS CloudWatch events**

You can setup alarms for your Amazon ECS resources using Amazon CloudWatch. You can get notifications via email when important metrics of the resources in this application go over the recommended thresholds. For Example: You can measure the CPU utilization and memory reservation across your cluster, and the CPU and automatically add/purge nodes.

Conclusion

With DAY2™ you can simplify Docker based deployments for ECS or Fargate clusters and automate routine operations with AWS native services, with No-Code and No-Agents. With DAY2™ you can save on both deployment and on-going operations costs.

The MontyCloud's DAY2™ ECS/Fargate Blueprint is available through DAY2™ Well-Managed Applications .

You can get started with a **14 day no-obligation free trial**.