

# Top 3 Challenges that can be Solved with DevOps

# Why Are DevOps Best Practices Important for Your Organization?

In today's IT world, software delivery innovators are moving beyond Agile and toward DevOps practices. When organizations adopt a DevOps methodology, the goal is usually to sync the Development and Operations team efforts, increase the quality of code produced, and accelerate application releases.

A DevOps strategy enables mechanisms that help coordinate, automate, and manages different tools and processes across the organization. Synchronizing Dev and Ops teams as well as automating thousands of processes is paramount when implementing and scaling DevOps.

Once these mechanisms are in place, organizations can achieve better visibility, faster software deployments, and take advantage of a centralized platform to manage tools and processes seamlessly. The ultimate goal organizations are seeking to achieve when implementing DevOps is to stay ahead of the competition by rapidly reacting to changes and removing the barriers that are in the way of innovation.

# Challenge #1

**Rapidly Scale to Meet Growing Market Demands and  
Focus on Releasing new Products and Features**



### The Challenge Overview

Over a period of 1 year, a major wireless communications company experienced a massive expansion of their international foot print. They needed to expand infrastructure capabilities to provide scalability and increased service levels of their marketing platform to meet these growing demands and focus on developing new product and features. The company needed an infrastructure that could be fully automated and is elastic, scalable, and more cost effective than their existing deployment, while avoiding needing to rebuild their applications. This basic deployment gave them insight into the AWS value proposition. The company needed to migrate their application to a cloud infrastructure designed to take full advantage of DevOps and AWS components.

### The ClearScale Solution

The wireless communications company partnered with ClearScale to optimize their architecture for AWS, automate it based on DevOps best practices, and migrate their entire platform from their existing environment to AWS. ClearScale worked closely with the company to develop an in depth understanding of their business and technical requirements including full review on their existing environment. After their review, ClearScale designed a solution fitting AWS best practices and the needs of the company. The company agreed on a complete cut-over to the new infrastructure versus a phased migration. Data synchronization was implemented between the existing deployments and AWS to lower downtime during the cut-over.

## Architecture

ClearScale designed a robust and highly scalable cloud architecture where Amazon CloudWatch service was leveraged to track performance and utilization of all AWS components to ensure that the solution was meeting specified requirements. The production Amazon Virtual Private Cloud (VPC) spans four AWS Availability Zones (AZs). Three subnets exist in each AZ; Public Subnet, Private Application Subnet, and Private Data Subnet. This provides the company with flexibility and redundancy within the AWS Region.

All external traffic is directed to AWS Elastic Load Balancers (ELBs) or HAProxy in the Public Subnet. The ELBs route whitelisted traffic to the company's application running on Amazon Elastic Compute Cloud Instances (EC2). HAProxy handles requests that are not whitelisted. The HAProxy EC2 instances are deployed across multiple AZs for load distribution and high availability (HA). Only infrastructure components that handle direct Internet connections are placed in the Public Subnet to limit security exposure.

Components in the public subnet communicate with two of the company's applications, RabbitMQ and Celery, in the Private Application Subnet, which do not allow any direct connections from the Internet. The company's application-hosting EC2 instances are set up as an Auto Scaling group across all Availability Zones.

## Challenge #1

This allows for automatic scaling of their applications depending on traffic load. Celery is used to run jobs and aggregate all event data that is written to Amazon ElastiCache (Memcache). RabbitMQ and Celery are set up as application clusters across multiple AZs for redundancy and availability. Amazon ElastiCache (Memcache & Redis) service, Amazon Relational Database Service (RDS) for MySQL, and MongoDB are deployed in the Private Data Subnet.

The Amazon RDS for MySQL implementation supports Content Management System and is deployed across all three AWS AZs. MongoDB, deployed on EC2 instances, is the primary database for all applications. Data in MongoDB is split into three shards across three replica sets, each in a different AZ. Components in the Private Data Subnet have been optimized for redundancy, performance, and scalability.

Security groups are configured for each application, based on their unique access and communication requirements. Security groups also follow strict naming conventions to accommodate DevOps automation practices. This has proven to be an effective approach, balancing granularity of control with manageability and automation. AWS CloudTrail service is used to record every AWS operation performed in the environment for audit logging and compliance.

## Automation

DevOps best practices are a big factor to the overall scalability of an environment. Automation and managing cloud infrastructure as code are integral components of these practices. AWS CloudFormation service was used to automate infrastructure deployment. Chef was chosen as the configuration management tool for this infrastructure because its an ideal platform to create versioned, repeatable configurations.

With AWS CloudFormation and Chef, automation takes only a few clicks to provision a new infrastructure block or migrate/redeploy an existing one. Chef is also a key to scalability of the solution, providing means to deploy new instances quickly and efficiently. Every instance deployed is fully automated through Chef recipes and cookbooks.

AWS CloudFormation templates were segregated into 4 functional areas:

**VPC Template** - where networking and security components are defined

**Apps Template** - where all application instances and load balancers are defined

**RDS DB and ElastiCache Template** - where MySQL RDS and caching layers are defined

**MongoDB Template** - where MongoDB clustering components are defined

### Chef Automation

Chef is a powerful automation platform that transforms complex infrastructure into code. By leveraging Chef, you can automate how applications are configured, deployed, and managed across the network. Chef is built around these simple concepts: achieving a desired state, centralized modeling of IT infrastructure, and resource primitives that serve as building blocks. Hosted Chef is the SaaS offering of this powerful platform.

Numerous Chef recipes and cookbooks were produced to automate server and application configuration. Some were shared cookbooks for common tasks, such as updating Route53 DNS records and managing EBS storage volumes, and some were individual cookbooks that were developed for the following applications: HAProxy, RabbitMQ, Celery and the MongoDB cluster. Here is an overview of one of the many Chef cookbooks (comprised of several recipes) developed for this wireless communications company. This particular cookbook automates the deployment of MongoDB clusters.

**Recipe for adding MongoDB repository**

**Recipe to setup MongoDB as configuration server for the cluster**

**Recipe to setup MongoDB as data server in the cluster**

**Recipe to create backup of cluster**

**Recipe to schedule daily backups**



Cloud automation is a huge factor when it comes to scalability and efficiency of infrastructure operations. Ultimately, successful automation practices equate to better availability and resiliency of the infrastructure and services - a winning combination when you combine the ability to deploy AWS infrastructure through CloudFormation, and the power to configure services with Chef.

### **A Successful Migration Methodology**

ClearScale used a repeatable cloud migration methodology developed over years of experience. An infrastructure audit was performed and based on the company's requirements, the architecture was redesigned with AWS components in place. The infrastructure was converted to code and full automation of infrastructure and server configuration was developed.

AWS CloudFormation and Chef automation was used to deploy the staging environment. As the staging environment was tested, all changes required were integrated back into the automation tools. When the staging environment received a stamp of approval, the same automation was used to deploy the production environment. Data replication was set up from their existing environment to AWS in order to keep MySQL and Mongo databases in sync.

A “mock” cutover was performed to further reduce migration risks. Every component and process (with exception of propagating DNS changes) had been tested thoroughly before the actual migration took place. This iterative testing practice is extremely effective in delivering successful migrations.

### **The Benefits**

The wireless communications company was able to meet the growing needs of their customers on demand, and with the flexibility and power of the AWS Cloud. With DevOps best practices and auto scaling in place, engineers were able to focus more on developing new products and features, and less on operations and capacity build-outs. The company has achieved the operational efficiencies provided by the AWS Cloud, bringing down their total cost of ownership (TCO). The company avoided the additional complexity of needing to rebuild their application to fit in AWS native environment by leveraging the chosen automation tools and processes. More importantly, they have increased the availability and robustness of services provided to their customers.

# Challenge #2

**Enable Continuous Integration and Continuous Deployment  
with Cloud Independent Strategy and Micro Services  
Architecture to Stay Ahead of the Competition**



### The Challenge Overview

An IT enterprise needed to undergo a challenging transformation to support a new SaaS product they were developing which required Continuous Integration & Deployment (CI/CD). It was also mandated that a vendor-independent and Cloud-independent strategy should be taken even though AWS was the primary delivery platform, driving the architecture decisions around automation and deployment. With a complex development environment that had already been manually deployed to AWS, our client needed a much more refined and easily-deployable approach since what they had lacked redundancy and automation.

### The ClearScale Solution

Needing a production-grade Cloud architecture to support their needs, ClearScale reviewed the client's requirements and identified a number of ways we could implement a robust solution that provided the scalability of AWS with the architectural management toolsets that existed elsewhere in the market; namely AWS CloudFormation, Chef, Marathon, Mesos, Artifactory and Zookeeper. By taking this approach, it provided end-to-end automation when it was implemented.

### Architecture

Our client's development teams use Docker to manage their development efforts within containers that allow them to pre-define the various runtime environments and the programming language the team has decided to use. Management of these Docker containers was architected to be hosted by Apache Mesos. To ensure that these containers would be properly deployed and managed, however, ClearScale implemented a framework called Marathon - a production-grade orchestration platform that pulled Docker elements directly from Artifactory.

This approach provides constant uptime by proactively monitoring activity within the environment and immediately seeking out other candidate containers if a failure occurs. With no impact to the application, Marathon will immediately scale and recover using Marathon REST APIs to identify where spare capacity is located and then dynamically shift the containers to nodes that meet the criteria needed to keep the application available.

ClearScale coupled the Marathon framework with an additional service called Zookeeper which manages failover of clustered nodes. By implementing Zookeeper, we were able to have it monitor the remaining nodes during an outage event and select a new master node to allow the application to continue running unabated.

## Challenge #2

Once these pieces of infrastructure were in place, we worked on migrating the services that had been deployed in development and test environments within AWS Virtual Private Cloud (VPC) to a production ready VPC. To do this, ClearScale leveraged other AWS services such as AWS Identity and Access Management (IAM) to provide a secure environment within AWS that adhered to our client's corporate security procedures and requirements. ClearScale also invoked Amazon Elastic Search service, which provides a clustered indexed data storage for use in the client's product.

### Automation

Within the development, staging and production environments in AWS, ClearScale also implemented Chef, a service that helps manage the deployment of code to the various environments using smaller packets of code called recipes. This allows for Continuous Integration (CI) and Continuous Deployment (CD) strategies that our client required. Chef was instantiated to set up recipes for configuring the base infrastructure components; in addition to Mesos and Marthon these included a data warehouse, HBase, RabbitMQ, SMTP servers, HAProxy for both internal and external proxy routing, ELK stack for application logging and alerting, and Active Directory. Chef was then used to configure and deploy Marathon templates.

## Challenge #2

The customized solution that ClearScale deployed to AWS for our client's SaaS product and continuous integration development strategy demonstrates the flexibility of AWS services and solutions. AWS does not limit a customer's ability to deploy a customized solution to fill specific needs. Rather, the teams at AWS recognize that customers have a variety of needs and requirements that Amazon needs to support regardless of whether or not the services that are implemented are entirely AWS offered solutions.

This is the true power behind utilizing AWS; by leveraging many of the services they provide, such as scalability, event logging and monitoring, and S3 buckets to store static files, and then integrating other services available in the development space that are widely used and supported, ClearScale can not only devise a solution that meets a client's immediate need, but also a solution that will scale to meet future needs.

### The Benefits

Designing an entirely new Continuous Integration and Continuous Deployment (CI/CD) approach while insuring constant uptime with minimal failover faults is a daunting task even under the best circumstances. Leveraging DevOps best practices, AWS components, scalable architecture and embedding a client required solution made the job easier.

When all was said and done, ClearScale was able to deliver a quality development and production environment to our client with a very unique architecture that met their strategy. By leveraging tools like Chef, Marathon, Mesos, Artifactory and Zookeeper the client was able to achieve a vendor independent and cloud independent solution that could be launched anywhere.



# Challenge #3

**Enable Automated Approach for Coordinating and Managing Various Sets of Tools and Processes Across the Organization**



### The Challenge Overview

The need to develop and deploy software solutions when financial information is involved requires a fair amount of diligence and foresight to insure that the applications that handle personal identifiable information, monetary transactional information, and the transmission of said data are secure. When working with a large financial institution, ClearScale knew that in order to meet the client's needs, these concerns needed to be taken into account.

Our client approached us and asked that we create a cloud infrastructure for their financial platform. It needed to support application development that leveraged existing micro-services and allowed the application to aid consumers with their financial investments and other financial services, connect to bank accounts and purchase bonds. The client wanted to leverage AWS services for ease of management and scalability.

Having the application interact with the micro-services was the easier issue, but designing an end-to-end fully automated solution was a complex undertaking. Based on our extensive experience with AWS, ClearScale knew that there were many different ways we could approach this effort and the solution that we ultimately decided on was robust and scalable enough to exceed our client's requirements.

### The ClearScale Solution

In this particular instance, we needed to use Docker containers in our solution design along with Amazon Web Services to support our client's development efforts. This would allow them to configure their own runtime environments, including choosing which platform to operate on, being able to pick the programming language their development team wanted, and definition of other application dependencies.

This solution is managed through Amazon EC2 Container Service (ECS) which enables running Docker containers on EC2 instances. This allowed their developers to maintain control over their Docker instances while allowing a clean integration with other AWS services such as AWS Elastic Beanstalk, a service that allows for cluster creation, task definition, and execution.

### Architecture & Automation

To fully build out and maintain the development environment while allowing access to financial and consumer information, ClearScale implemented the Amazon API Gateway service to enable the micro-services the client was interested in. Using either the web app that had already been provided by the client which managed authentication and authorization information using SAML, or by directly hitting the RESTful API endpoints using the AWS IAM service, the API Gateway allowed for an exchange of information from the micro-services directly into the development environment all while being encrypted using SSL and leveraging AWS API Keys.

## Challenge #3

ClearScale has implemented AWS Lambda to aid in the deployment pipeline that the client's Jenkins Continuous Integration (CI) servers needed as well as the AWS Simple Notification Service (SNS). The entire infrastructure environment was implemented in AWS Virtual Private Cloud (VPC) to allow for ease of access behind robust authentication and distribution across multiple physical regional locations for redundancy purposes. In addition, AWS VPC allows for private connections with other AWS services without having to go over the internet using a NAT device, a VPN connection or AWS Direct Connect. It further allows communication between AWS VPC and other AWS services without impacting network traffic. We configured the Virtual Private Gateway in such a way that allows the investment platform admin gateway our client chose to initiate the VPN request.

All client static file information was stored in a dedicated Amazon Simple Storage Service (S3) bucket and the environment set up to utilize AWS Elastic Load Balancing (ELB), a load balancer for application and file servers. As the infrastructure environment was built out further, ClearScale also deployed AWS CloudWatch to monitor activity within the environment, as well as AWS CloudTrail to log and audit all transactional information. By implementing and deploying all of this within AWS, we were able to take further advantage of other services, such as Amazon CloudFront and Amazon ElastiCache which, when utilized together, allowed for more scalability.

## Challenge #3

Using AWS Relational Database Service (RDS) with a MySQL engine, ClearScale was able to configure encrypted database replication manually through a VPN tunnel. This allowed for a configured RDS MySQL instance in the primary region to be copied over to a similarly configured RDS MySQL instance in a secondary region using the VPN tunnel, thus allowing for data redundancy and recover should it ever be required. This paradigm allows for a “hot-swap” situation should disaster recovery procedures be initiated with minimal downtime.

ClearScale was able to deliver an automated infrastructure solution that allowed our client’s developers to rapidly build and test their code on local Docker instances. They then are able to push their code into the development pipeline using AWS CodeCommit which triggers a notification using AWS SNS. Since AWS CodeCommit cannot directly activate AWS Lambda functions, the AWS SNS trigger acts as the intermediary in this instance and performs this action, activating the Jenkins build deployed within the AWS VPC. Jenkins pulls the fresh code in from AWS CodeCommit, creates the application build, and runs the necessary tests before finally building the Docker image. Once complete, the Docker image is pushed to the AWS ECR repository and Jenkins sends a notification to AWS Elastic Beanstalk to deploy the new application build.

## Challenge #3

ClearScale has implemented AWS Lambda to aid in the deployment pipeline that the client's Jenkins Continuous Integration (CI) servers needed as well as the AWS Simple Notification Service (SNS). The entire infrastructure environment was implemented in AWS Virtual Private Cloud (VPC) to allow for ease of access behind robust authentication and distribution across multiple physical regional locations for redundancy purposes. In addition, AWS VPC allows for private connections with other AWS services without having to go over the internet using a NAT device, a VPN connection or AWS Direct Connect. It further allows communication between AWS VPC and other AWS services without impacting network traffic. We configured the Virtual Private Gateway in such a way that allows the investment platform admin gateway our client chose to initiate the VPN request.

All client static file information was stored in a dedicated Amazon Simple Storage Service (S3) bucket and the environment set up to utilize AWS Elastic Load Balancing (ELB), a load balancer for application and file servers. As the infrastructure environment was built out further, ClearScale also deployed AWS CloudWatch to monitor activity within the environment, as well as AWS CloudTrail to log and audit all transactional information. By implementing and deploying all of this within AWS, we were able to take further advantage of other services, such as Amazon CloudFront and Amazon ElastiCache which, when utilized together, allowed for more scalability.

### The Benefits

The customer was able to go live with a highly secure and automated architecture fitting for a financial organization. They also ended up with a very elegant and easy to manage infrastructure that leveraged many AWS native services. The CI/CD systems simplified the code release process, an Achilles heel of any system, reducing downtime and speeding up time to market. The result was a highly automated, repeatable micro services oriented architecture that meets all regulatory requirements of a financial technology company.

# About ClearScale

An AWS Premier Consulting Partner in North America, ClearScale has the proven capability to build, deploy, automate and manage complex cloud architectures on AWS. Our core competency is delivering custom cloud projects and services for clients who have limited cloud expertise on staff or who need additional resources to execute better and faster. We leverage the best cloud technology available to provide a solution that is unique to your project requirements.

Based in the San Francisco Bay Area, we built a born-in-the-cloud consulting practice exclusive to AWS, helping companies like Samsung, Shinola Detroit, GoPro, Medallia, The Globe and Mail, Easton and Conde Nast to deploy or migrate core business applications onto AWS. Our long track record of successful DevOps projects, cloud application development, and infrastructure migrations spans hundreds of client engagements. Whether this is your first project in the cloud or one of many, ClearScale's team has the expertise to handle your most complex requirements.

**Contact ClearScale to Discuss your Cloud Initiative**

**Tel (800) 591-0442 | Email: [sales@clearscale.net](mailto:sales@clearscale.net) | [www.ClearScale.com](http://www.ClearScale.com)**



# About Amazon Web Services (AWS)

For 10 years, Amazon Web Services has been the world's most comprehensive and broadly adopted cloud platform. AWS offers over 70 fully featured services for compute, storage, databases, analytics, mobile, Internet of Things (IoT) and enterprise applications from 35 Availability Zones (AZs) across 13 geographic regions in the U.S., Australia, Brazil, China, Germany, Ireland, Japan, Korea, Singapore, and India. AWS services are trusted by more than a million active customers around the world – including fast growing startups, large enterprises, and leading government agencies – to power their infrastructure, make them more agile, and lower costs.

**To learn more about AWS visit:**

<http://aws.amazon.com>.