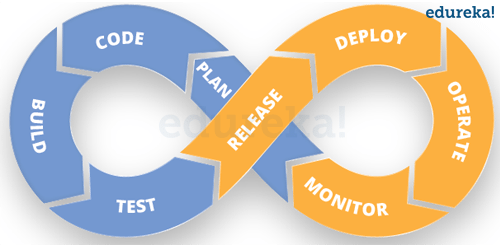
**What is DevOps?**

DevOps is a software development approach that involves Continuous Development, Continuous Testing, Continuous Integration, Continuous Deployment and Continuous Monitoring of the software throughout its [development life cycle](https://www.edureka.co/blog/devops-lifecycle/). These activities are possible only in DevOps, not Agile or waterfall, and this is why Facebook and other top companies have chosen DevOps as the way forward for their business goals. DevOps is the preferred approach to develop high-quality software in shorter development cycles which results in greater customer satisfaction.



**DevOps in Government**

**Profile**

The**National Aeronautics and Space Administration (NASA)** is the agency of the United States government that is responsible for the nation’s civilian space program and for aeronautics and aerospace research.

**Challenge**

NASA needed to move nearly 65 applications from a traditional hardware-based data center to a cloud-based environment for better agility and cost savings. The rapid timeline resulted in many applications being migrated ‘as-is’ to a cloud environment. This created an environment spanning multiple virtual private clouds (VPCs) and [AWS](https://www.edureka.co/blog/videos/aws-tutorial/) accounts that could not be easily managed. Even simple things, like ensuring every system administrator had access to every server, or simple patching, were extremely burdensome.

**Solution**

This problem was solved by leveraging [Ansible Tower](https://www.edureka.co/blog/ansible-tower/) to manage and schedule the cloud environment.

**Result**

As a result of implementing the Ansible Tower, NASA is better equipped to manage its AWS environment. Tower allowed NASA to provide better operations and security to its clients. It has also increased efficiency as a team. If you have a look at the numbers then:

* The time for updating nasa.gov was brought down from over 1 hour to under 5 minutes
* Process of patching came down from a few days to 45 minutes
* Achieving near real-time RAM and disk monitoring (accomplished without agents)
* Provisioning OS Accounts across the entire environment in under 10 minutes
* Baselining standard AMIs was brought down from 1 hour of manual configuration to becoming an invisible and seamless background process
* Application stack set up from 1-2 hours to under 10 minutes per stack

**DevOps in Banking**

**Profile**

**The Royal Bank of Scotland** commonly abbreviated as **RBS**, is one of the retail banking subsidiaries of The Royal Bank of Scotland Group plc, together with NatWest and Ulster Bank. The Royal Bank of Scotland has around 700 branches, mainly in Scotland, though there are branches in many larger towns and cities throughout England and Wales.

****

**Challenge**

The Royal Bank of Scotland was struggling to consolidate and re-architect its strategic payment transaction hub, which involved dozens of integration touchpoints. RBS had recently acquired another financial institution, so it needed to integrate that company’s systems. However, the RBS was already in the process of migrating its own systems to a new strategic transaction hub and delivering 43 functional enhancements with the new platform. The cost and complexity of maintaining four separate messaging hubs, in effect, quadrupled.

**Solution**

Finally, a solution was developed to help the bank tackle its integration challenges. It was necessary to assist the bank in identifying all the points across the software delivery life cycle at which it could use automated testing, agile development, and service virtualization to accelerate development, identify defects earlier and speed releases to production.

The solution included an integration and service virtualization tool. These software products helped create an end-to-end virtual environment for testing rather than having to create real test environments, which are expensive to build and difficult to maintain.

Using the solution, RBS was able to automate integration testing across more than 80 interfaces and sub-systems involved in executing its trading transactions. This enabled continuous testing and supported the team’s agile development process.

**Result**

In three years, the RBS realized substantial benefits from the integration and service virtualization solution. These benefits are as follows :

* Reduced system integration testing time from three weeks to half a day
* Reduced the number of production incidents from undetected defects by 99.6 percent
* Increased the project delivery capacity by 100 percent, growing from 40 to 80 projects completed annually
* Saved an estimated US$6 million in hardware, software and resource costs
* Accelerated the time from project inception to delivery by 44 percent, enabling the bank to bring products and services to customers faster and seize new market opportunities

**DevOps in Insurance**

**Profile**

NJM Insurance Group, headquartered in the West Trenton section of Ewing Township, Mercer County, New Jersey, United States, offers personal auto, commercial auto, workers’ compensation, homeowners, and umbrella insurance. The company was formed in 1913 by a group of factory owners seeking workers’ compensation coverage. NJM has grown to provide insurance to more than 850,000 policyholders living in New Jersey and Pennsylvania.

****

**Challenge**

As New Jersey Manufacturers Insurance Group (NJM) upgraded core technology platforms, they wanted to accelerate software delivery in order to align delivery practices with business demands for upgraded solutions. With their volume and cadence of work increasing day-by-day, they quickly reached their limits as they tried to scale their entirely manual software deployment processes.

Maintaining and demonstrating compliance without introducing more painful manual processes was also a topmost concern for this large insurance company. NJM needed a more reliable way to streamline software deployments and at the same time ensure compliance controls were maintained. Their ultimate goal was to automate the release of high-quality software and at the same time reducing complexities and minimizing overhead.

**Solution**

NJM successfully implemented deployment automation which accelerated and standardized software delivery processes while helping them more easily demonstrate compliance. This initiative significantly reduced operational overhead and streamlined deployment processes so NJM could scale.

**Result**

* Reduced deployment times dramatically Non-production deployment – from days to minutes Production deployments – 30%-50% faster
* 1,000 to 1,500+ deployments per month through automation. A few years ago, the company manually handled its releases
* The company started doing Continuous, scheduled, and self-service deployments. Eliminated reliance on specially-skilled technicians to deploy software. QA, developers, and trainees are now handling deployments
* No-hassle compliance Highly visible, a zero-touch process that is fully traceable and auditable

**DevOps in Retail**

**Profile**

This specialty retailer is a well-known household name whose stock is traded on the NYSE. With about 100 stores across the US and a strong eCommerce presence, this retailer is known for its inspiring merchandise and is also consistently ranked as one of the best places to work.

**Challenge**

As the retail landscape is changing rapidly, this organization was looking to enable its in-house development team to stay nimble and one step ahead of the competition. Tasked with servicing the organization’s eCommerce site and in-store systems, the company decided to create an IT modernization plan that was broken into discrete projects. The first such project was to grow developer and IT automation, increasing their productivity and the ability to quickly iterate on innovation.

**Solution**

* Docker container microservices
* DevOps based cloud computing
* Fully automated CI/CD coupled with an immutable infrastructure

**Result**

As a development shop that wears multiple hats, this retailer’s development team was keen to standardize on a single platform. Its environment had grown complex and difficult to manage; yet, with new Docker container-based microservices working in conjunction with tools like Ansible, Docker Swarm and Hashicorp Consul and Vault, the organization has gained a higher level of automation. This automation has, in turn, allowed developers to get to work faster, iterate with agility, and create a more strategic impact on the organization.

**DevOps in Travel**

**Profile**

***Amadeus***is one of the largest travel operators worldwide. Their systems interact with **90%** of all travel-related transactions, serving more than **700** airline companies and around **600,000** hotels, processing more than **55,000** operations per second at peak loads — and the numbers are constantly growing.

****

**Problem**

The company used a private cloud with virtual machines served by Vagrant and vSphere. However, the number of computing resources spend on maintaining the hypervisor layer of the infrastructure was too high, and the speed of processing was not optimal, while even several seconds of delay can result in huge losses for a travel operator.

**Solution**

The company chose [**Docker**](https://www.docker.com/)instead of Vagrant and decided to move to an on-prem cloud running OpenShift, Docker, and Kubernetes. By using a proprietary DevOps management system they were able to efficiently utilize their whole IT infrastructure, taking the resources previously used by hypervisors. This accounted for nearly **20%** of their computing power.

**Result**

The company got **several million worths of computing resources** by simply utilizing their IT infrastructure efficiently. In addition, Docker containers running in Kubernetes clusters allow processing the workloads in real-time, as there is no delay due to the absence of the hypervisor layer.

As this list shows, industries can benefit from using DevOps even if they are not tech-centric. One of the main goals of DevOps is to stop departments from working in compartmentalized ways. Instead, it encourages them to communicate with each other. That aim helps to achieve goals faster with less friction.