



# How DevOps as a Service Can Accelerate Your Software Development Lifecycle

---

WHITEPAPER

---

# Abstract

DevOps as a Service is the disruptive model in enhancing the flow and reducing complexity of the SDLC. DaaS contributes to the effective filling of the existing gaps between the development and operation teams, improves better collaboration activities, increases the speed of work, and time to market.

The white paper considers what DaaS is, main tenets of DaaS, how DaaS is more effective than other traditional DevOps models, and implementation guidelines. This means organizations adopting DaaS achieve lower costs, more elasticity and scalability, higher quality of software and contented clients.

# Introduction

**The today's digital environment** requires quicker and more effective ways of creating software. Hence, there is always interest in how different aspects of organizational work can be made to be more efficient in order to create and release good software products as fast as possible. DevOps is a practice that integrates the development process with the operational process of an organization, changing how the conventional software delivery occurred.

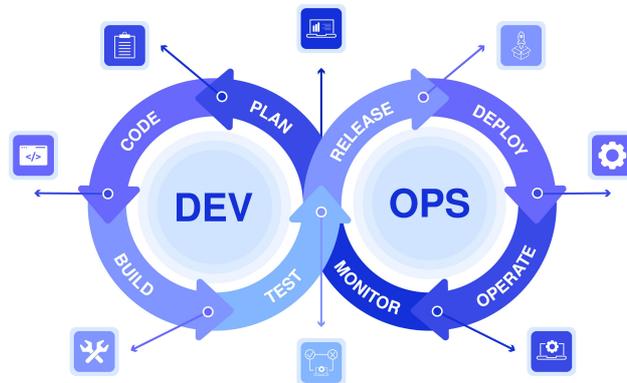
Now, with the advent of cloud technology, a different model has emerged that promises even more agility, scalability, and cost-effectiveness: DevOps as a Service, or DaaS.

DaaS extends the DevOps concept to the cloud from a box concept and then provides an organization with a scalable solution that can be activated on-demand as the organization scales. As rooted in the cloud, it erases the imperative to invest in expensive structures, cuts overhead costs, and invests in core business objectives.

This white paper provides more information about DaaS and its advantages, its application and its effect on SDLC.

# What is DevOps?

DevOps, in a nutshell, is a methodology that encourages collaboration between developers and operations personnel throughout the life cycle of a software development. The collaboration is achieved through practices and tools that automate and streamline processes, reducing errors and speeding up delivery.



It basically means the combination of people, process, and technology towards a continuous value creation for customers. DevOps integrates and automates the work of software development and IT operations as a way of achieving continuous improvement and shortening the systems development life cycle.

## DevOps adoption and performance

1

DevOps adoption has never been higher

74 %

Have introduced DevOps to at least some projects

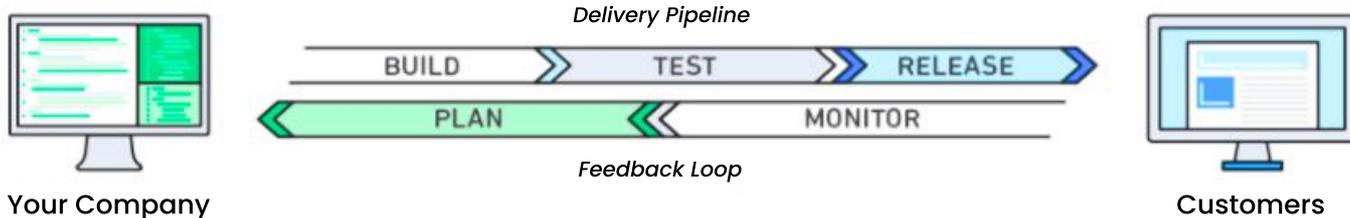
x2

Those who fully adopted DevOps have more than doubled in 5 years

82 %

Of large organizations (+1000 employees) have adopted some DevOps

# How Does DevOps Work



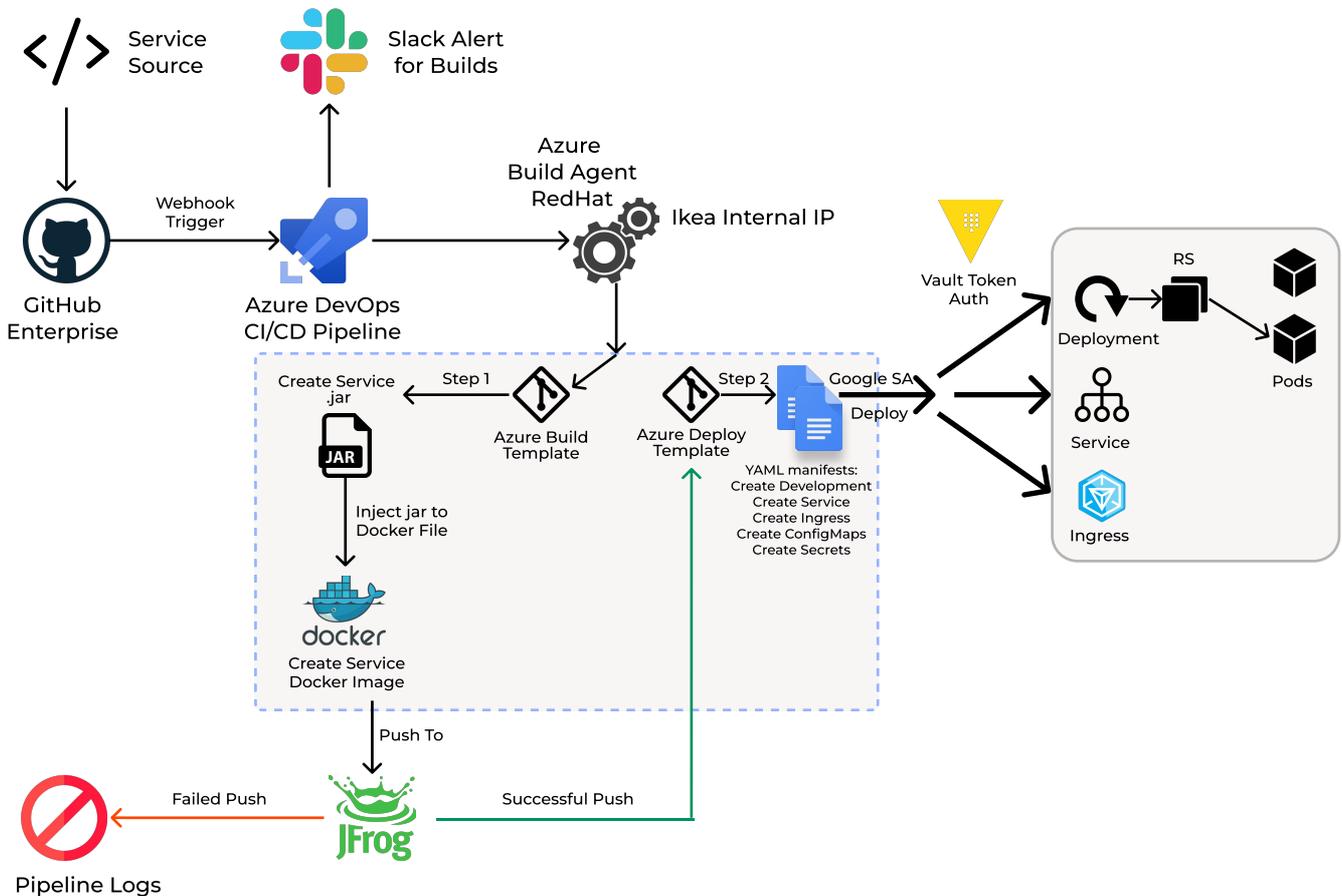
Graphically, the DevOps lifecycle can be shown as an infinite loop, which is symbolic of the cycle of development and delivery. Key phases are mentioned here:

- 1. Plan:** This involves planning the features, improvements, and bug fixes the team will work upon. Proper planning requires stakeholders, project managers, developers, and operations teams to ensure alignment to business goals is obtained.
- 2. Code:** This is the actual writing of the code of the software. Most teams would utilize version control systems like Git to manage changes efficiently and collaborate effectively with their team.
- 3. Build:** Once the code is written, it undergoes compilation and forms a build. In this step, the code written by different developers is integrated, ensuring that it works together in harmony.
- 4. Test:** Testing is one of the main components of automated and manual testing involved in DevOps. Testing ensures that new code does not introduce bugs or issues and also verifies that the software meets quality standards.
- 5. Release:** The release phase includes the deployment of the software into production environments. Through continuous integration, or the use of CI/CD pipelines, releases can be automated, thus making the deployment faster and more reliable.
- 6. Deploy:** In general, to deploy means to make the software available to its end-users. In DevOps, deployments are more frequent, incremental, and lower-risk, thus enabling quick feedback.
- 7. Operate:** Once deployed, the software has to be kept up by monitoring its performance, reliability, and availability. DevOps teams monitor their software with special tools that provide insight into its health and facilitate necessary changes.
- 8. Monitor:** By monitoring continuously, insights into application performance and user experience are gathered. The data from monitoring tools helps the team find areas that need improvements to be made and further updates.
- 9. Feedback:** Feedback collection is necessary for learning the needs of the users and the improvement of the product. It ensures feedback loops take user input into consideration during planning and development phases.



# DevOps as a Service

DevOps as a Service is one cloud-based model of DevOps that extends its principles to enabling continuous integration and continuous delivery, known as CI/CD. By automating the workflows and enabling smoother communication between the development and the operation teams, DaaS provides an uninterrupted path right from code commit to deployment. This service-oriented approach allows tapping into expertise and resources from external sources with no significant investments in-house.



According to a recent [Gartner report](#), over **85%** of organizations are expected to adopt a cloud-first approach by 2025.

# Key Principles of DevOps

DevOps practices vary, but several of the common ones that are easily adopted include:

## 1. Continuous Integration

CI is the practice of frequently integrating the code changes by multiple developers in a central repository. Several integrations occur daily, with automated tests being run in order for new code not to break already existing functionality. This minimizes much integration challenges and hence, allows teams to detect and resolve conflicts as early as possible.

## 2. Continuous Delivery

Taken one step further from CI, CD automatically completes the release process. In so doing, teams are able to deploy new features and updates to production rapidly and reliably, reducing the possibility of human error, which means that software can be released at any time.

## 3. Infrastructure as Code (IaC)

IaC treats the process of infrastructure management and provisioning as the process of managing machine-readable definition files rather than physically configuring hardware or using interactive tools that configure operating systems and middleware. This does indeed permit consistent, reproducible infrastructure deployments, reduces manual errors, and accelerates deployment.

**Businesses can also dedicate 33% more time to improving infrastructure thanks to DevOps.**

## 4. Collaboration and Communication Tools

Some of the essential tools used in DevOps are those of communication and collaboration. They include Slack, which provides messaging; Jira, a project management tool; and GitHub for version control. These tools help teams stay in sync and keep updated with the happenings in the course of development.

## How can DevOps as a Service Transform your Software Development Lifecycle?

DevOps as a Service is an outsourced model for deploying tooling, infrastructure, and expertise to enable organizations to perform and manage DevOps practices. While traditional DevOps requires organizations to provide resources and infrastructure to perform these functions themselves, DaaS is performed via the cloud for streamlined cost-effective delivery. Built on cloud platforms, DaaS applies automation of processes, integration of workflows, and application deployment at higher speed and reliability.

# DevOps as a Service: Benefits

Quick overview of the incredible benefits of implementing DevOps as a Service approach:

### 1. Effective Collaboration and Communication

One of the key concepts in DevOps is to break silos between the development and operational teams. DaaS provides support for culture by applying those tools and practices which help promote communication. The team will apply it and align objectives more concretely, hence providing faster resolution of issues and at the same time making the workflow smooth. Teams will better collaborate, misunderstandings can be avoided, and it ensures everyone is on the same page.

### 2. Accelerated Development and Deployment

Integration and delivery become automated with DevOps as a Service, increasing the development and deployment speed manifold, so to say. Developers are continuously integrating their code into the common repository because of Continuous Integration. It offers the capability to release code in production at any given moment in time through Continuous Delivery. Having such acceleration allows quicker releases whereby organizations can immediately act upon any market demands and user feedback.

**94% of organizations say platform engineering plays a key role in maximizing DevOps benefits.**

### 3. Better Quality and Reliability

Automated testing and continuous monitoring form part of DaaS. This therefore, provides the organization with the ability to detect and fix issues at an early stage in software products' development on matters regarding quality and reliability. With automated tests, problems that were caused by code changes are avoided and continuous monitoring shows insight into the working of systems in real-time and thus helps find solutions to problems way ahead of time.

### 4. Scalability and Flexibility

DaaS providers offer scalable solutions to meet the ever-changing demands of organizations. Scaling up or down, with increased demand for resources, can be easily done without having to make major investments in infrastructure. This flexibility will ultimately enable the organization to manage workloads with efficiency, optimize resource utilization, and ensure operational efficiency and cost-effectiveness.

### 5. Reduced Costs and Resource Optimization

The reasons being that by outsourcing DevOps, organizations save on the costs involved in maintaining an in-house DevOps team and the required infrastructure. Providers of DaaS give access to advanced tools and technologies. So, there is no need to invest in costly software licenses and hardware, and automation reduces manual intervention so resources can be utilized to focus on strategic initiatives and innovation.

**DevOps can help companies cut down the time spent on handling support cases by 60% – good news for those looking to streamline processes next year.**

### 6. Improved Security and Compliance

Security and compliance are major concerns in software development. DaaS integrates robust security practices at the core of the development lifecycle to ensure applications are built with security in mind from the very beginning. This is possible through continuous monitoring and automated security testing, which allow you to discover vulnerabilities at an early stage and diminish the risk of a security breach. What's more, DaaS providers stay up-to-date regarding changes in compliance standards and make sure applications adhere to all regulatory requirements. DevOps simplifies development, and DevSecOps adds a highly required layer of security. Explore- [DevOps vs DevSecOps](#) for better understanding.

# Best Practices for Implementing DevOps as a Service Approach

## 1. Evaluate Present Capabilities

It is an excellent idea to grasp the current condition of your organization's development and operations prior to trying to venture into DaaS. This typically includes the following activities.

- **Locate Bottlenecks:** Find the points where processes are slowing or not happening properly. Are there steps in a process that need to be automated? Are teams failing to communicate properly?
- **Assess Current Tools:** Have critical reviews of your current development and operations tools. Are they outdated or lacking some functionality you desperately need?
- **Skill Assessment:** Assess what skills and experience your development and operations teams have. Would they be in a position to teach DevOps practices?

Once an organization is fully informed of its standing, the same can then gauge where DaaS would bring the best value and can mold the implementation accordingly.

## 2. Clearly Determine Your Objectives

Setting clear-cut objectives remains very vital in any DaaS implementation process. Ask yourself these kinds of questions:

- **What do I want to accomplish with DevOps:**  
time-to-market, quality of the software or bridge the gaps between teams?
- **How does DevOps enable the business objectives?**  
Does DevOps let you innovate more, move faster according to customers needs or reduce costs?
- **What are the metrics of success?**  
Identify the key performance indicators, best known as KPIs, in order to track the progress your DevOps implementation has made.

By defining your goals, you are aligning your DaaS implementation with the overall business strategy.

### 3. Choose the Right Provider

The most important step in choosing the right DevOps service provider will make or break your implementation. Based upon your needs consider the following:

- **Expertise:** Does the supplier have relevant experience within your particular industry or with similar projects?
- **Toolsets:** Does the supplier provide a full suite of DevOps tools that will meet your needs?
- **Scalability:** Will the supplier be in a position to scale their services to meet your growing needs?
- **Support:** Does the supplier fully support their services in terms of support and maintenance?
- **Security:** How does the supplier ensure data and application security?
- **Cost:** What is the total cost of DaaS, and is that cost justified by the benefit to be accrued?

These factors must be taken properly into consideration when choosing your provider for your DevOps goals.

### 4. Creating a Collaborative Culture

DevOps is a cultural shift in which collaboration and communication across diverse development and operation teams are required. Foster a culture for:

- **Shared responsibility:** Break silos by providing a sense of shared ownership in the whole software delivery life cycle.
- **Open communication:** Share frank and transparent communication across teams to increase knowledge sharing and problem-solving.
- **Continuous improvement:** Implement continuous learning and improvement to find new ways of improving DevOps practices.

### 5. Automation to Implement

Automation is core to DevOps. Automate processes where practical that could reduce the human effort in performing a task to improve efficiency and reduce errors. The automation of areas such as:

- **Testing:** Automation will ensure that unit, integration, and regression testing provide quality code through early defect detection within the development phase of a product.
- **Deployment:** Automated continuous integration and deployment pipelines for seamless and smooth code movements from development to production.
- **Infrastructure Provisioning:** Among others, this will include the usage of IaC for the automation of infrastructure resource creation and management.
- **Monitoring:** Set up automated monitoring of your applications and infrastructure concerning performance and proactively conduct incident detection and resolutions.

## 6. Setup CI/CD Pipelines

Continuous Integration and Continuous Delivery form the bedrock of DevOps. The idea behind CI is that each developer integrates code changes into a central repository; automated tests take place. Continuous delivery automatically deploys the code changes to production. By creating a CI/CD pipeline, the following can be attained:

- Squashing defects earlier in development because of better code quality can be ensured through automated testing.
- **Reduce the risk of deployment:** one can automate deployment steps by removing the risks of human errors or inconsistencies.
- **Faster time-to-market:** Provide new features and updates faster to your end customers.

## 7. Monitor and Optimize

Monitor the performance of your DevOps processes and tools continuously to identify further opportunities for improvement. Metrics and feedback shall help to:

- **Measure success:** Track your KPIs to assess the effectiveness of the DevOps implementation.
- **Find bottlenecks:** Identify those points in processes which are slow or inefficient.
- **Enhance performance:** Tune your tools, processes, or team structure for better performance and efficiency.
- **Gather feedback:** Obtain feedback from your development and operations teams on areas that need improvement and pose challenges.

That is to say, by following the steps described above, and from the benefits available on DevOps as a Service, get your software delivery in order, smoothen the collaboration process, and deliver quality software quicker.

## Different DevOps tools used for software development lifecycle:

### 1. Planning



- **Jira Product Discovery:** A tool to help you prioritize and manage your product backlog.
- **Mural:** A collaborative mind mapping tool.
- **Miro:** A collaborative whiteboard platform.

### 2. Plan

- **Jira:** A project management tool for agile teams.
- **Confluence:** A team wiki and knowledge base tool.
- **Slack:** A communication platform for teams.

### 3. Build



- **Kubernetes:** An open-source system for automating deployment, scaling, and management of containerized applications.
- **Docker:** A platform for developing, deploying, and running applications in containers.
- **Puppet:** An infrastructure automation and configuration management tool.
- **Chef:** An infrastructure automation and configuration management tool.
- **Ansible:** An open-source software provisioning, configuration management, and application deployment tool.
- **Terraform:** An open-source infrastructure as code tool that lets you define and provision your cloud infrastructure using declarative configuration files.

## 4. Test

- **Xray:** A test management tool for Jira.
- **mabl:** A mobile testing platform.
- **Zephyr:** A test management tool for Jira.

## 5. Deploy

- **Jenkins:** An open-source automation server.
- **AWS:** Amazon Web Services, a comprehensive suite of cloud computing services.
- **Bitbucket:** A Git repository hosting service.
- **CircleCI:** A continuous integration and continuous delivery (CI/CD) platform.
- **Sonarsource:** A platform for continuous code inspection and automatic code reviews.

## 6. Operate

- **Jira Service Management:** A service desk and IT service management tool.
- **Jira Software:** A project management tool for agile teams.
- **Opsgenie:** An alerting and incident management tool.
- **Statuspage:** A status page platform.

## 7. Observe

- **AppDynamics:** A business application performance monitoring tool.
- **DataDog:** A monitoring and analytics platform for cloud applications, servers, databases, tools, and logs.
- **Slack:** A communication platform for teams.
- **Splunk:** A software platform for searching, analyzing, and visualizing machine-generated data from a wide variety of sources.
- **New Relic:** A cloud-based observability platform built for software engineers.
- **Opsgenie:** An alerting and incident management tool.
- **Pingdom:** A real user monitoring tool.

- **Nagios:** An open-source monitoring system.
- **Dynatrace:** A digital intelligence platform for cloud and enterprise environments.
- **Hosted Graphite:** A monitoring platform for time-series data.
- **Sumo Logic:** A cloud-based log management and analytics platform.

## 8. Continuous feedback

- **Slack:** A communication platform for teams.
- **Jira Service Management:** A service desk and IT service management tool.
- **Pendo:** A product feedback and user insights platform.

## 10. Cloud-Based Development Environments

- **AWS Cloud9:** Web-based IDE for writing, running, and debugging code.
- **Microsoft Visual Studio Online:** Customizable cloud-hosted IDE for coding and collaboration.
- **Google Cloud Shell:** Browser-based CLI for managing Google Cloud resources.

# ENDING WORDS

As the DevOps space continues to evolve, staying on top of trends and responding to challenges will make all the difference in how organizations maximize the benefits of DaaS. Continuous learning, investment in skilled talent, and leveraging innovative technologies-these are what will position businesses for real success in the long term in the digital era.

In other words, DevOps as a Service acts as the powerful framework which lets your organization reinvent software development and channelizes innovation. With ScalaCode, you will be able to unlock full potential DaaS and drive business success in competitive marketplaces.



Partner with ScalaCode to ensure the best support for the implementation of AI in software development.

# Get in Touch →



## About ScalaCode™

ScalaCode is a globally acclaimed software development and IT consulting services provider with 13 years of experience. We help businesses of all sizes with custom software solutions that meet their specific requirements and drive business growth.



### Headquarter & Delivery Center

#### India

Suite # 16, Ground Floor  
Tower A, Stellar IT Park, C 25  
Sec - 62, Noida, Uttar Pradesh

+91 8527981600

sales@scalacode.com

### Our Global Presence

#### United States

P.O Box 1286,  
Layton, Utah 84041

+1 801 618 5791

+1 409 995 3910

#### United Kingdom

Rowan House, Culmhead  
Nr Taunton TA3 7DU  
Somerset, UK

+1 801 618 5791

#### Sweden

Fagerlidsvägen 15D  
566 92 HABO

+46 76 310 62 86

#### Canada

2433 Lakeshore Road, Burlington  
Ontario Canada L7R 1B9

#### Australia

16-18 Beverley Avenue  
Rosedale South Brisbane  
Queensland 4123

+61423661345