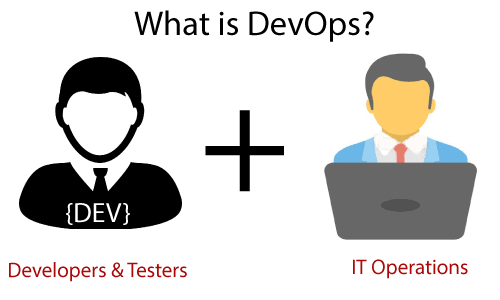
# DevOps

The DevOps is the combination of two words, one is ****Development**** and other is ****Operations****. It is a culture to promote the development and operation process collectively.

## What is DevOps?

The DevOps is a combination of two words, one is software Development, and second is Operations. This allows a single team to handle the entire application lifecycle, from development to ****testing, deployment****, and ****operations****. DevOps helps you to reduce the disconnection between software developers, quality assurance (QA) engineers, and system administrators.

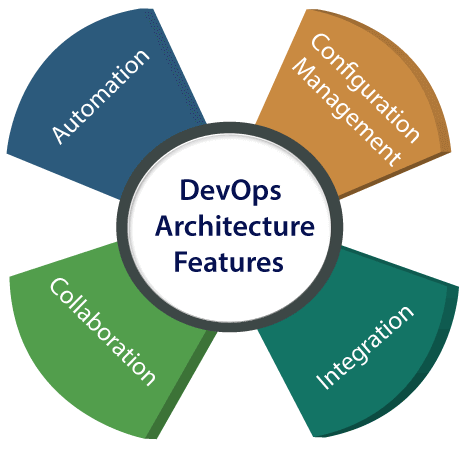


* DevOps promotes collaboration between Development and Operations team to deploy code to production faster in an automated & repeatable way.
* DevOps helps to increase organization speed to deliver applications and services.
* DevOps is nothing but a practice or methodology of making "****Developers****" and "****Operations****" folks work together.

## Why DevOps?

* The operation and development team worked in complete isolation.
* After the design-build, the testing and deployment are performed respectively. That's why they consumed more time than actual build cycles.
* Without the use of DevOps, the team members are spending a large amount of time on designing, testing, and deploying instead of building the project.
* Manual code deployment leads to human errors in production.
* Coding and operation teams have their separate timelines and are not in synch, causing further delays.

## DevOps Architecture Features



### 1) Automation

Automation can reduce time consumption, especially during the testing and deployment phase. The productivity increases, and releases are made quicker by automation.

### 2) Collaboration

The Development and Operations team collaborates as a DevOps team, which improves the cultural model as the teams become more productive with their productivity, which strengthens accountability and ownership.

### 3) Integration

Applications need to be integrated with other components in the environment. The integration phase is where the existing code is combined with new functionality and then tested.

### 4) Configuration management

It ensures the application to interact with only those resources that are concerned with the environment in which it runs. The configuration file can be written during deployment, or they can be loaded at the run time, depending on the environment in which it is running.

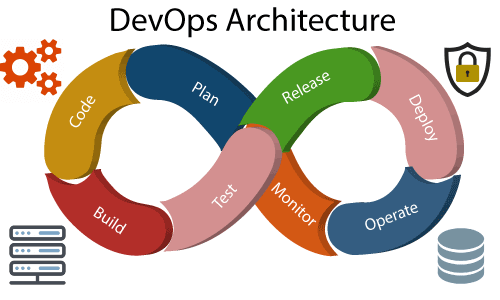
### Advantages

* DevOps is an excellent approach for quick development and deployment of applications.
* It responds faster to the market changes to improve business growth.
* DevOps escalate business profit by decreasing software delivery time and transportation costs.
* DevOps clears the descriptive process, which gives clarity on product development and delivery.
* It improves customer experience and satisfaction.
* DevOps simplifies collaboration and places all tools in the cloud for customers to access.
* DevOps means collective responsibility, which leads to better team engagement and productivity.

### Disadvantages

* DevOps professional or expert's developers are less available.
* Developing with DevOps is so expensive.
* Adopting new DevOps technology into the industries is hard to manage in short time.
* Lack of DevOps knowledge can be a problem in the continuous integration of automation projects.

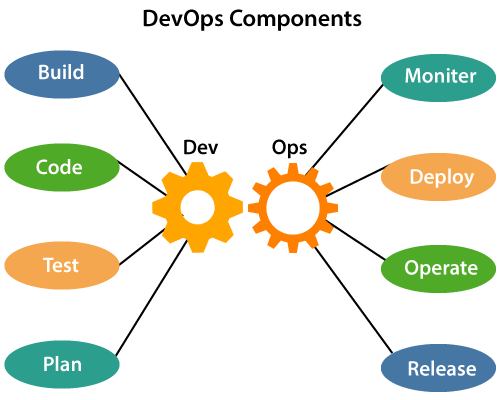
# DevOps Architecture



Development and operations both play essential roles in order to deliver applications. The deployment comprises analyzing the ****requirements, designing, developing****, and ****testing**** of the software components or frameworks. The operation consists of the administrative processes, services, and support for the software. When both the development and operations are combined with collaborating, then the DevOps architecture is the solution to fix the gap between deployment and operation terms; therefore, delivery can be faster.

DevOps architecture is used for the applications hosted on the cloud platform and large distributed applications. Agile Development is used in the DevOps architecture so that integration and delivery can be contiguous.

DevOps Components



# DevOps Tools

#### 1. Version Control tools

****GitHub****: Github is considered as one of the largest and most advanced development platforms in the world. Millions of developers and companies build, ship, and maintain their software on GitHub. Some of its’ salient features are:

* Collaborative Coding
* Automation / CI & CD
* Security including additional features for enterprise customers
* Project Management

****Bitbucket****: Bitbucket is a very popular platform, with over 10 million registered users. Although it is a platform for hosting code, it goes beyond just code management. Teams can plan projects, collaborate on code, test, and deploy from a single platform. Some of its features are:

* Tighter Jira and Trello integration.
* Integrated CI/CD to build, test and deploy.
* Pull requests and approve code review more efficiently.
* Keep your code secured in the Cloud with IP whitelisting and 2-step verification.

****GitLab****: It is an all-in-one DevOps Tool for rapid software delivery. It enables teams to perform all tasks right from Planning to SCM to Delivery to Monitoring and Security. Following are a few of its features:

* Single interface, single conversation thread, and single data store to manage projects effectively – Single source of truth.
* CI/CD for robust, scalable, and end-to-end automation to work together efficiently – Continuous Everything.
* Built-in functionality for Automated Security, Code Quality and Vulnerability management & with tight governance, your DevOps speed never slows down.

#### 2. Container Management tools

****Docker****:Docker is a light-weight tool that aims to simplify and accelerate various workflows in your SDLC with an integrated approach. A docker container image is a standalone, executable package that includes everything you need to run an application. Some of its primary features are that helped it become indispensable among DevOps tools are:

* Standardized Packaging format for diverse applications.
* Container runtime that runs on various Linux and Windows Server OSs.
* Developers uses Docker for build, test and collaborate.
* Docker Hub to explore millions of images from community and verified publishers.
* Package, Execute and Manage distributed applications with Docker App.

****Kubernetes****: Kubernetes is an open-source DevOps tool used to automate deployment and management of containerized applications & perhaps one of the most popular container orchestration tools. Features that differentiate it from other DevOps Tools include:

* Make changes to your application or its configuration and monitoring application health simultaneously – Automated rollouts & rollbacks.
* It offers own IP addresses and a single DNS name for a set of Pods – Service Delivery and load balancing.
* Automatically mount the storage system of your choice.
* Self-healing capability.

****Mesos****: Apache Mesos is a DevOps tool to manage computer clusters. It is a distributed systems kernel for resource management and scheduling across entire datacenter and cloud environments. Its features include:

* Offers native support to launch containers with Docker and AppC images.
* Supports cloud-native and legacy applications to run in the same cluster with pluggable scheduling policies.
* Runs on cross platforms like Linux, OSX and Windows.
* Scale to 10,000s of nodes easily.

#### 3. Application Performance Monitoring tools

****Prometheus****:Prometheus is an open-source and community driven performance monitoring solution. It also supports container monitoring and creates alerts based on time series data. Solution includes the following features:

* Scaling with the help of functional sharding and federation.
* Numerous client libraries allow easy service instrumentation.
* Powerful reporting capabilities through PromQL.

****Dynatrace****: Covers all monitoring needs such as application performance, digital experience, business analytics, AIOps, and infrastructure monitoring. Its features are:

* Automate orchestration with open APIs.
* Provides extensive cloud support and compatible with all major db technologies.
* Dynatrace APM solution provides automatic quality checks and KPIs.
* AI Driven problem detection and resolution.

****AppDynamics****: AppDynamics facilitates real-time insights into application performance. This DevOps tool monitors and reports on the performance of all transactions flowing through your application. Its features are:

* Agents are Intelligent and know when to capture the details of transactions.
* Solves performance problems through an analytics-driven approach.
* Automatically finds out normal performance and stop false alarms.
* Smart Analytics enables to find and fix issues from the very beginning.
* Enables full system-wide data recording.

#### 4. Deployment & Server Monitoring tools

****Splunk****: Splunk is a monitoring and exploring tool that is used on SaaS and on-premises. It has features like:

* Monitor and troubleshoot across your infrastructure, including physical, virtual, or in the cloud.
* Modernize applications for better customer experiences through accelerated innovation.
* AIOps with Machine Learning for predictive alerting and auto-remediation.
* Improved Efficiency in MTTA with mobile-first, automated incident response.

****Datadog****:Datadog is a SaaS-based DevOps tool for server and app monitoring having hybrid cloud environments. It facilitates monitoring of Docker containers as well. Some of its salient features:

* Seamlessly aggregates metrics and events across the full DevOps stack.
* Offers end-to-end user experience visibility in a single platform.
* Prioritizes business and engineering decisions with user experience metrics.
* Built to give visibility across teams.

****Sensu****: Sensu is an open-source devops tool for monitoring cloud environments. It is easily deployable through Puppet and Chef. Following are its features:

* The Sensu Observability Pipeline is integrated, secure and scalable. Collaboration between development and operations relies on self-service workflows with integrated authentication solutions.
* Declarative configurations and a service-based approach to monitoring let you define the monitoring insights that matter most, automating your workflows so you can focus on what matters.

#### 5. Configuration Management tools

****Chef****: Chef is an open-source DevOps tool for automation and configuration management built by Erlang and Ruby. Its features are:

* “Cookbooks” which facilitates infrastructure coding in languages specific to domains.
* Easily integrated with cloud platforms like Amazon AWS, MS Azure, GCP etc.
* Configuration as code.

****Puppet****:Puppet is responsible for managing and automating your infrastructure and complex workflows in a simplistic manner. Features of this DevOps Tool are:

* Automate and simplify critical manual tasks by extracting configuration details across various operating systems and platforms.
* When you have 100, 1000 of severs or a mixed environment or when you have to plans to scale your infrastructure, it becomes difficult to maintain all servers in a certain state - Puppet helps you in saving time and money, scaling effectively and do this effectively.

****Ansible****: Ansible delivers simple IT automation that ends repetitive tasks and frees up teams for more strategic work. Focusing on two key use-cases:

* Configuration management – Aim to be the simplest solution and designed to be minimal in nature, consistent, secure and highly reliable, with a focus on getting started quickly for administrators, developers and IT managers.
* Orchestration - Ansible’s library of modules and easy extensibility, makes it simple to orchestrate different conductors in different environments, all using one simple language.

#### 6. CI / Deployment Automation tools

****Bamboo****:It is a DevOps tool to help you practice Continuous Delivery, from code to deployment. It gives the ability to tie automated builds, tests, and releases together in a single workflow. Some of its salient features are:

* Allows users to create multi-stage build plans and set up triggers to start builds upon commits.
* Parallel automated tests unleash the power of Agile Development and make catching bugs easier and faster.
* Tighter integration with Jira, Bitbucket.

****Jenkins****:Written in Java, Jenkins is an open-source platform for continuous integration and continuous delivery that is used to automate your end-end release management lifecycle. Jenkins has emerged as one of the essential DevOps Tools because of its features:

* Used as a simple CI server or turned into the CD hub for any project.
* Easily set up and configured via its web interface, which includes on-the-fly error checks and built-in help.
* Easily distribute work across multiple machines, helping drive builds, tests and deployments across multiple platforms faster.

****IBM UrbanCode****: As a deployment automation and release management solution, IBM UrbanCode enables uninterrupted delivery for any combination of on-premises, mainframe and cloud applications. Some of its features are:

* Uses an enterprise-optimized solution along with development, testing and release tools to enhance build management.
* Automates application development, middleware configuration, and database changes.

#### 7. Test Automation tools

****Test.ai****: It is an AI-powered automation testing tool to release apps faster and with better quality. Its AI-Bots:

* Build tests without coding or scripting.
* Accelerate testing to the speed of DevOps.
* Scale testing to any platform, any app.
* Maintain tests automatically and improve quality everywhere.

****Ranorex****: An all-in-one solution for any type of automated testing, whether it is cross-browser testing or cross-device testing. Its features include:

* All the tools you need for test automation comes in a single license.
* Test on real devices or simulators/emulators.
* Allows simple integration with CI servers, issue tracking tools, and more

****Selenium****:Primarily used to automate web applications for testing purposes but can, also, be used to automate other web-based admin tasks. Three components:

* Selenium WebDriver allows you to create robust, browser-based regression automation suites and tests and helps you in scaling and distributing scripts across many environments.
* Selenium IDE is a chrome and Firefox add-on that helps in simple record-and-playback of interactions with the browser.
* Selenium Grid for scaling your testing efforts by running tests on several machines and manage multiple environments from a central point

#### 8. Artifact Management tools

****Sonatype NEXUS****: Claims to be the world’s #1 repository manager, Sonatype efficiently distributes parts and containers to developers acting as a single source of truth for all of your components, binaries and build artifacts. Its features are:

* Offers universal support for all popular build tools.
* Efficiency and flexibility to empower development teams.

****JFRog Artifactory****: Functions as the single source of truth for all container images, packages, and Helm charts, as they move across the entire DevOps pipeline. Its features are:

* Scale with active/active clustering and multi-site replication for your DevOps setup.
* Allows you to choose the tool stack and integrates with your environment.
* Release faster and automate your pipeline via powerful REST APIs.

****CloudRepo****: Used for managing, sharing and distributing private Maven and Python repositories.

* Stores the repositories across multiple servers to ensure high availability.
* Easily provide or restrict access to your clients.
* Integrates with all major CI tools.

#### 9. Codeless Test Automation tools

****AccelQ****: AcceIQ is leads the codeless test automation space among the DevOps Tools. It is a powerful codeless test automation tool, which allows the tester to freely develop test logic without having concern about the programming syntax:

* Follows a design-first approach and enforces modularity and reusability in the development of test assets effortlessly.
* Handle iframes and other dynamic controls.
* Supports advanced interactions and logic development capabilities.

****QuerySurge****: QuerySurge is the smart data testing solution that is the first-of-its-kind full DevOps solution for continuous data testing.

* Robust API with 60+ calls
* Seamlessly integrates into the DevOps pipeline for continuous testing
* Verifies large amounts of data quickly
* Validates difficult transformation rules between multiple source and target systems
* Detects requirements and code changes, updates tests accordingly and alerts team members of said changes
* Provides detailed data intelligence & data analytics

****Appvance****: This Al and ML-powered autonomous testing platform performs end-to-end testing along with ML assisted codeless scripting. Its features are:

* A test automation system with level-5 autonomy.
* Self-healing scripts & AI-generated tests for complete application coverage and validations with 90% less effort.
* Powers continuous testing in your DevOps environment.

****Testim.io****: AI-based UI Testing to help you run tests that deliver super-fast authoring that increases coverage and quality. It helps in your DevOps Journey by:

* Integrating with tools like Saucelabs, Jira, and Github.
* Eliminating flaky tests and reduces maintenance.
* Pinpointing root cause to fix bugs and release faster.
* Efficiently expanding testing operations with control, management, and insights.