



DEVOPS

Satya

Agenda:

- ▶ What is DevOps
- ▶ Where did DevOps come from
- ▶ What problems led to the creation of DevOps
- ▶ How is DevOps different from Traditional IT
- ▶ What are the Features of DevOps Implementation
- ▶ DevOps Lifecycle Phases and Measures
- ▶ DevOps tools

What is DevOps ?

- ❑ DevOps represents a change in the IT culture with a complete focus on rapid IT service delivery through the adoption of agile practices in the context of a system-oriented approach
- ❑ DevOps is a combination of two words Development and Operations. It is neither an application nor a tool, it is just a culture to promote development and Operation process collaboratively. As a result of DevOps implementation, the speed to deliver applications and services has increased.

Where did DevOps come from?

Two primary ancestors of DevOps are given below by expert IT professionals in several disciplines.

- ▶ **ESM (Enterprise Systems Management):** People involved in the initial phases of DevOps are system administrators. These experts brought the key ESM practices to DevOps like configuration management, automated provisioning, system monitoring, and the toolchain approach, etc.
- ▶ **Agile Development:** DevOps can be considered as an outgrowth of the agile. It is simply extending the Agile principles beyond boundaries of the code to the entire delivered services.

“When you are going agile without DevOps, it is like racing with a tractor instead of a car. You can do laps, but it will not move faster, and ultimately you are going to waste a lot of fuel without having any fun.”

What problems led to the creation of DevOps?

- ▶ DevOps implementation has increased the rate of software delivery and the revenue for business stakeholders.

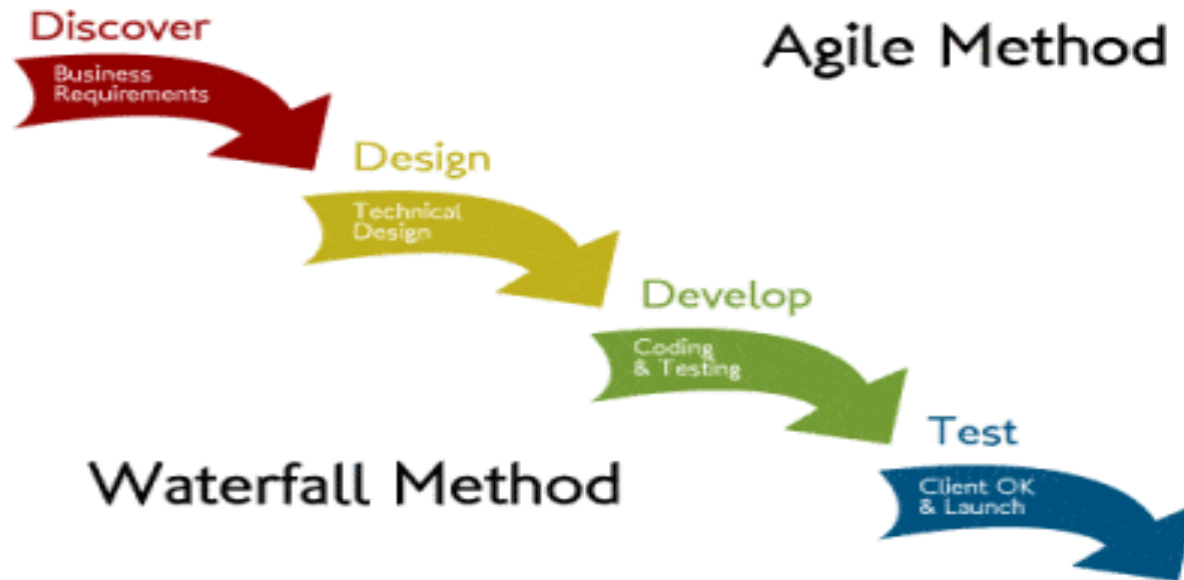
Following listed reasons are the most considerable ones that led to the creation of DevOps.

- ✓ Before DevOps, operation and development teams were working in an isolated environment.
- ✓ Testing and Deployment activities mostly were performed in an isolated manner after design-build step, and they took more time than actual project completion time.
- ✓ Team members usually spend a large amount of time in deploying, testing, designing, and building the projects
- ✓ Human production errors were deployed during manual code conduction.
- ✓ Operations and coding teams generally had different timelines and did not have proper synchronization that results in further delay.
- ▶ To avoid the hassles mentioned above and non-collaborative performance measures, there was an urgent need for robust IT technology like DevOps to satisfy business owners and stakeholders.

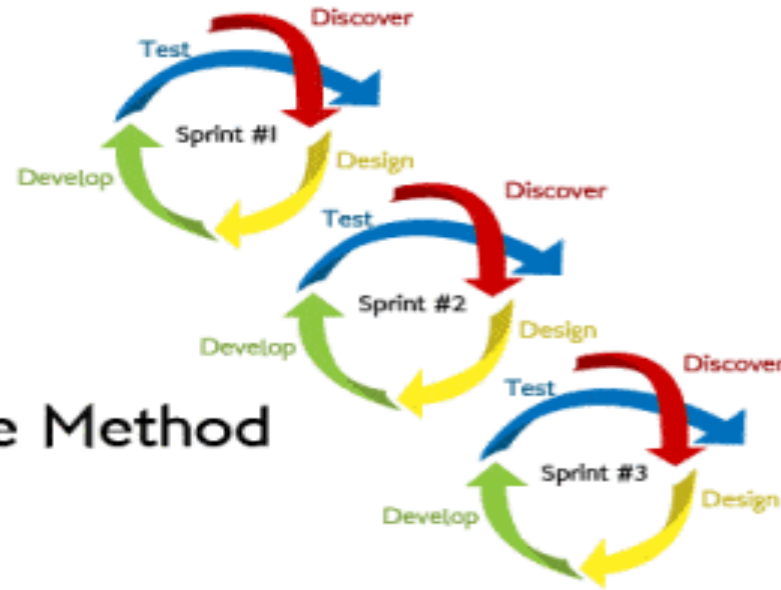
How is DevOps different from Traditional IT

- ❑ Let us compare the waterfall and Agile model with DevOps to see what changes DevOps brings to the IT.
- ▶ The two main development methodologies are agile and waterfall. They are commonly applied to software development and project management.
- ▶ Waterfall model, projects are completed sequentially. each phase should be completed before starting another phase. We cannot operate multiple phases at the same time.
- ▶ Agile model is related to a method of project management. Projects are completed iteratively in a cycle, that is splitting the tasks into shorter segments of work with frequent reviews and adaptation of plans.

Development Methodologies



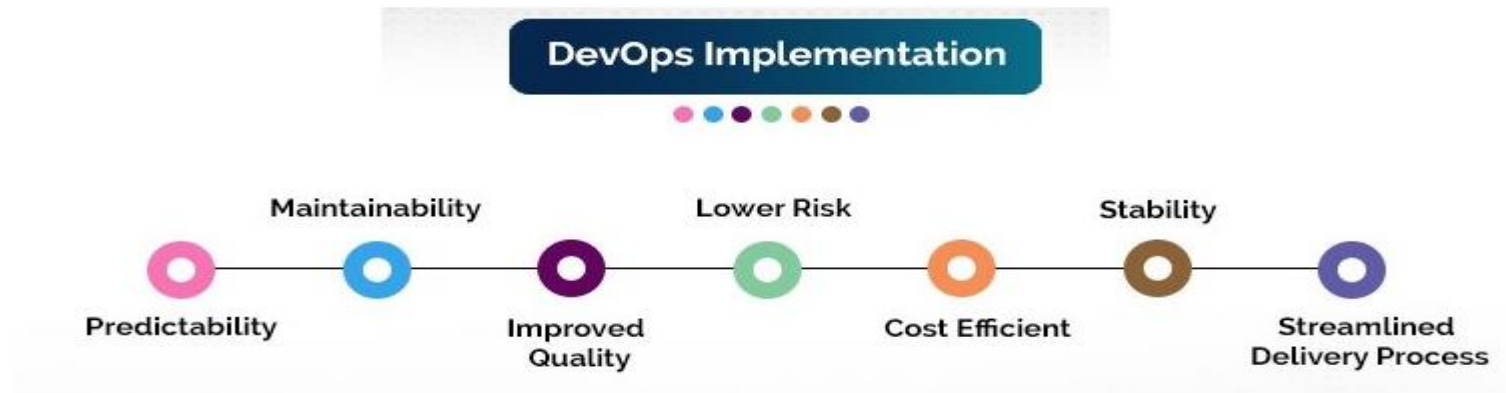
Agile Method



Traditional IT	DevOps
Once the order for new servers is placed, the development team starts working on testing. The development team has to continue with heavy paperwork as required by enterprises to deploy the infrastructure.	Once the order for new servers is placed, the development team and operations team start the paperwork to set up new servers that result in better visibility of infrastructure equipment.
Projections about failover, data center locations, redundancy, and storage requirements are not clear as no inputs are available from the development team even if they have the depth knowledge of the application.	Projections about failover, data center locations, redundancy, and storage requirements are 100 percent clear because of accurate inputs given from the development team.
In old software development processes, the operations team has no idea of the progress of the development team. Operation team has to prepare a monitoring plan as per their own understanding.	In DevOps, the operations team have a complete idea of the progress of development. Operations team and development team work together to develop a monitoring plan that caters to the current business, and IT needs.
Before go-live, the load testing may crash the application, and the release may get delayed. It affects the overall cost of the project and project delivery deadline.	Before go-live, the load testing makes the application a little slow. The development team quickly fixes bottlenecks, and the application is released on time.

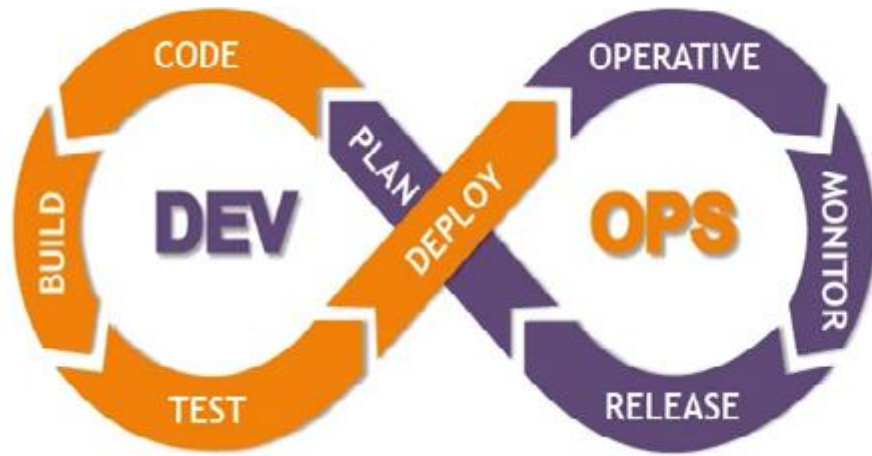
What are the Features of DevOps Implementation?

“DevOps is not a goal but a never-ending process of continual improvement.”

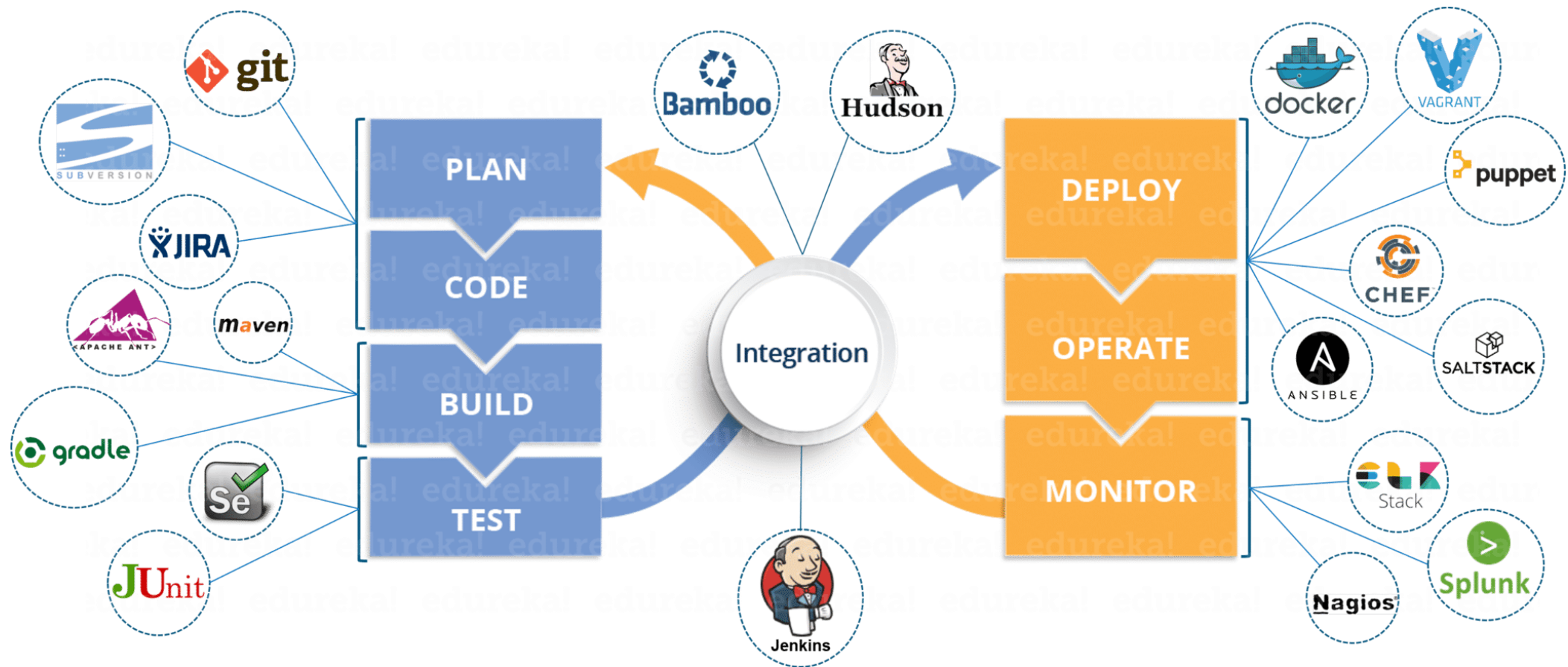


- ▶ **Predictability:** DevOps decreases the failure rate of new product releases.
- ▶ **Maintainability:** The process improves the overall recovery rate at the time of the release event.
- ▶ **Improved Quality:** DevOps improves the quality of product development by incorporating infrastructure issues.
- ▶ **Lower Risk:** Number of defects gets decreased across the product
- ▶ **Cost Efficient:** Cost efficiency is improved due to DevOps that is always an aspiration of every business organization.
- ▶ **Stability:** DevOps implementation offers a stable and secure operational state.
- ▶ **Streamlined Delivery Process:** As DevOps provides streamlined software delivery

DevOps Lifecycle Phases and Measures



It is a continuous process – continuous development, testing, integration, deployment and monitoring.



PERIODIC TABLE OF DEVOPS TOOLS (V3)																			
<div><div><div>Os</div><div>Fr</div><div>Fm</div><div>Pd</div><div>En</div></div><div><div>Open Source</div><div>Free</div><div>Freemium</div><div>Paid</div><div>Enterprise</div></div><div><div></div><div></div><div></div><div></div><div></div></div><div><div>Source Control Mgmt.</div><div>Database Automation</div><div>Continuous Integration</div><div>Testing</div><div>Configuration</div></div><div><div></div><div></div><div></div><div></div><div></div></div><div><div>Deployment</div><div>Containers</div><div>Release Orchestration</div><div>Cloud</div><div>AIOps</div></div><div><div></div><div></div><div></div><div></div><div></div></div><div><div>Analytics</div><div>Monitoring</div><div>Security</div><div>Collaboration</div></div></div>																			
1 Os GI GitLab																	2 En Sp Splunk		
3 Fm Gh GitHub	4 En Dt Datical																		
11 Os Sv Subversion	12 En Db DBMaestro																		
19 En Cw ISPW	20 En Dp Delphix	21 Os Jn Jenkins	22 Fm Cs Codeship	23 Os Fn FitNesse	24 Fr Ju JUnit	25 Fr Ka Karma	26 Fm Su SoapUI	27 En Ch Chef	28 Fr Tf Terraform	29 En XLd XebiaLabs XL Deploy	30 En Ud UrbanCode Deploy	31 Os Ku Kubernetes	32 Fm Cc CA CD Director	33 En Pr Plutora Release	34 Pd Al Alibaba Cloud	35 Os Os OpenStack	36 Os Ps Prometheus		
37 Pd At Artifactory	38 Fm Rg Redgate	39 Pd Ba Bamboo	40 Fm Vs VSTS	41 Fr Se Selenium	42 Fr Jm JMeter	43 Os Ja Jasmine	44 Pd Sl Sauce Labs	45 En An Ansible	46 Os Ru Rudder	47 En Oc Octopus Deploy	48 Os Go GoCD	49 Os Ms Mesos	50 Pd Gke GKE	51 Fm Om OpenMake	52 Pd Cp AWS CodePipeline	53 Pd Cy Cloud Foundry	54 En It ITRS		
55 Pd Nx Nexus	56 Os Fw Flyway	57 Os Tr Travis CI	58 Fm Tc TeamCity	59 Os Ga Gatling	60 Fr Tn TestNG	61 Fm Tt Tricentis Tosca	62 Pd Pe Perfecto	63 En Pu Puppet	64 Os Pa Packer	65 Fm Cd AWS CodeDeploy	66 En Ec ElectricCloud	67 Os Ra Rancher	68 Pd Aks AKS	69 Os Rk Rkt	70 Os Sp Spinnaker	71 Pd Ir Iron.io	72 En Mg Moogsoft		
73 Fm Bb BitBucket	74 En Pf Perforce	75 Fm Cr Circle CI	76 Pd Cb AWS CodeBuild	77 Fr Cu Cucumber	78 Os Mc Mocha	79 Os Lo Locust.io	80 En Mf Micro Focus UFT	81 Os Sa Salt	82 Os Ce CFEngine	83 En Eb ElasticBox	84 En Ca CA Automic	85 En De Docker Enterprise	86 Pd Ae AWS ECS	87 Fm Cf Codefresh	88 Os Hm Helm	89 Os Aw Apache OpenWhisk	90 En Ls Logstash		



91 En XLi Xebialabs XL Impact	92 Os Ki Kibana	93 Fm Nr New Relic	94 En Dt Dynatrace	95 En Dd Datadog	96 Fm Ad AppDynamics	97 Os El ElasticSearch	98 Os Ni Nagios	99 Os Zb Zabbix	100 En Zn Zenoss	101 En Cm Checkmarx SAST	102 En Wp Signal Sciences WPP	103 En Bd BlackDuck	104 Os Sr SonarQube	105 Os Hv HashiCorp Vault
106 En Sw ServiceNow	107 Pd Jr Jira	108 Fm Tl Trello	109 Fm Sk Slack	110 Fm St Stride	111 En Cn CollabNet VersionOne	112 En Ry Remedy	113 En Ac Agile Central	114 Pd Og OpsGenie	115 Pd Pd Pagerduty	116 Os Sn Snort	117 Fm Tw Tripwire	118 En Ck CyberArk	119 En Vc Veracode	120 En Ff Fortify SCA

Periodic Table Of DevOps Tools - A Revolutionary Breakthrough

<https://xebialabs.com/periodic-table-of-devops-tools/>

Q&A