

# What is DevOps

## What is DevOps ?

---

DevOps is the union of people, process, and products to enable continuous delivery of value to end users.

And all the following statements are TRUE about DevOps :

- DevOps is development and operations collaboration.
- DevOps is using automation.
- DevOps is small deployments.
- DevOps is feature switches.
- DevOps is treating your infrastructure as code.
- DevOps is decentralized.
- DevOps is a process, not a product.

DevOps culture nurtures better communication and collaboration between teams, thus breaking the silos formed among teams.

DevOps aims at:

- Developing high quality software
- Deploying in frequent cycles
- Reducing time to move from idea to implementation

## DevOps Foundational Aspects

---

Following are some of the foundational aspects and behaviors to be adopted, because they could drive DevOps Culture in your organization or team.



Hypothesis Driven  
Development

Evidence Gathered  
in  
Production

Live Site  
Culture

## Team Autonomy & Enterprise Alignment

Balance between Autonomy i.e., letting team what they want to do and Business Alignment.

## Rigorous Management of Technical Debt

Managing Technical Debt lets you deliver code and value effectively. Take time in your schedule to reduce technical debt.

## Focusing on Flow of Customer Value

This is a mine shift. It is a strong one for Development and Operations. The mind set of developer should not be on if his/her code is checked into version control, or for a tester if he has completed his test cases. The developer's or tester's mind should be thinking of the feedback the customer would be given on the feature that is developed, as the credit. This is the habit that drives the DevOps culture strong.

## Evidence gathered in Production

We are looking at production. And we figure out ways to gather insights on what are customers doing in production, to inform new ways of doing things or adding features. This leads to hypothesis driven development. Even in an agile model, we do not exactly know what we are building. Instead we build a hypothesis that we believe a piece of code would result in a specific beneficial behavior. This way we get patterns to estimate the next piece of code to be done.

## Live Site Culture

The goal or desire from the entire organization, for example say we need to get to production. Its this mindset that says there is no place like production, production is heaven, and we need to bring our application to production. This needs a shift of lot of people's attitude. The code that is not in production is no use.

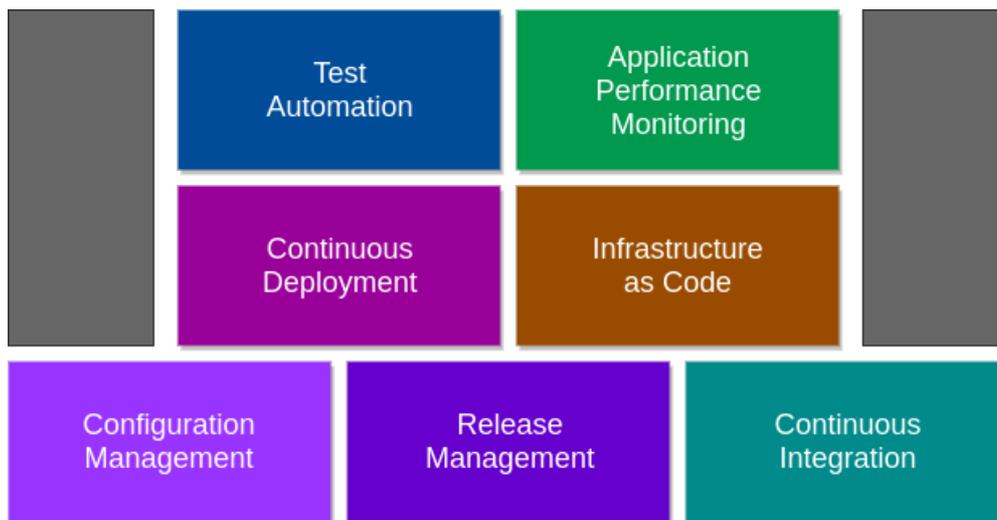
## Managing Infrastructure as a Flexible Resource

Its the idea that we need to treat our infrastructure as code.

# DevOps Practices

---

Following is a quick view of DevOps Practices.



## Configuration Management

Understanding what we are deploying, how we are deploying and what is the configuration that is going into production.

## Release Management

Understanding and controlling how we are building a release pipeline that we can trust

## Continuous Integration

The idea that we should be testing, compiling at every single check in and at every single version control commit.

## Continuous Deployment

Getting the application out to a test environment at very least and potentially all the way to production on a very rapid cadet.

## Application Performance Monitoring

Monitoring and watching our application in production to get performance, error information, usage information and insights on how we can evolve our application.

## Test Automation

---

Automating unit tests from developer, deployment tests, integration tests, user experience tests, UI tests, etc.

## Infrastructure as code

---

When we deploy our code, we have to get a related infrastructure that is checked into version control. Any change in infrastructure should be done in an automated fashion. In other words, we change the configuration and that is checked into code to change the infrastructure.

## Agile principles applicable to DevOps

---

Following are the principles of Agile methodology that apply to DevOps culture.

- Highest priority is to satisfy the customer through early and continuous delivery of software
- Deliver working software frequently
- Business and developers must work together daily throughout the project
- Working software is the primary measure of progress

## Agile Influence in DevOps

---

Processes and practices influenced by the Agility Movement:

- Source Control
- Small, frequent releases
- Automated Testing
- Continuous Integration
- Continuous Deployment
- Peer Review
- Immutable Infrastructure

It is not the strongest of the species that survive, nor the most intelligent, but the one most responsive to change. Get ready for DevOps.

### Software Modelling

- ◆ [UML Tutorial](#)

### Useful Resources

◆ [How to Learn Programming](#)

⇒ [What is DevOps](#)