



Introduction To devops.

Theory:- Over the years, software development has gone to great lengths in evolution, starting from brute waterfall model we now have advanced methods like Evolutionary prototyping, agile which not only allow effective development but also parallel working and multiple project handling with very decent precision and good result.

But these have lead to troubles as well, Imagine using Agile model where in a CD model of delivery is required, every new iteration, fix, build must be deployed and tested.

Earlier that was not the case as the final product was the only thing that required delivery was the end product.

Devops is basically the core integration of development with operation, which basically means that Devops is the operational model that takes care of delivery & fixing the delivery related bugs, this simply allows developers to focus solely on creation (of bugs :-p) of software.

It allows tackling the following challenges:-

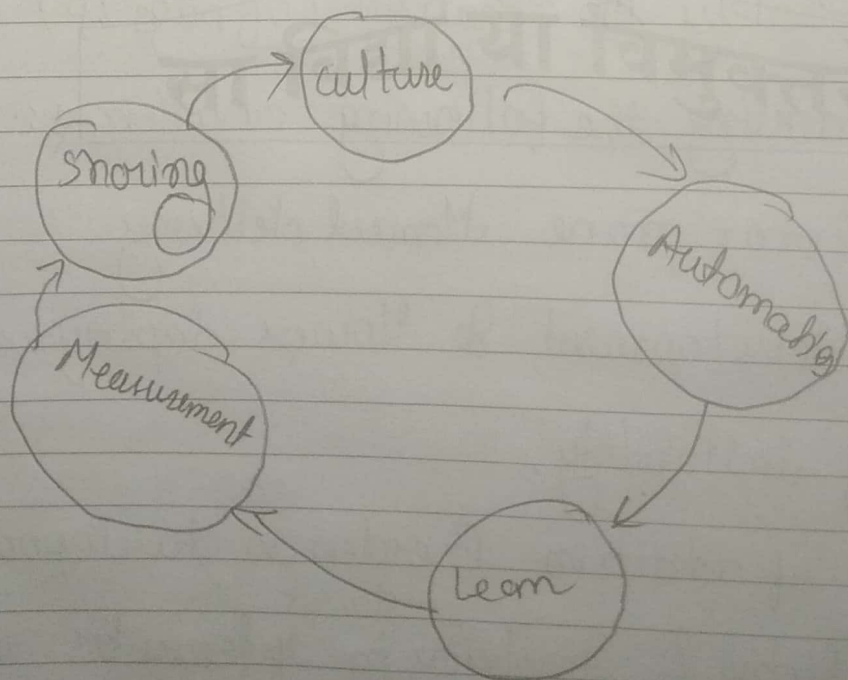
- i) Costly Error prone Manual delivery
- ii) Slow development & slower deployment
- iii) Risk of instability.
- iv) Removal of addition Burden on developers.
- v) Error fixing & resolving in seconds as well as in software.

In technical terms, Devops is combination of cultural philosophy, practices and tools that increases an organization's ability to deliver applications and service at high velocity, evolving and improving products at a faster pace than organization using traditional software development and infrastructure management process.

Gains with Devops:-

- i) Collaboration and trust:- Devops allows Developers to work and operations to operate collaborating flawlessly and effectively.
- ii) Faster Releases & Smarter Works.
- iii) Accelerate time to resolution.
- iv) Better Manage unplanned work.

AMS Framework for Devops:-



→ Culture:-

- (i) Develops culture in one word, it'd be "collaboration".
- (ii) Sharing a common goal and having
- (iii) Helps in focusing attention on specific problem.

→ Automation:- Time

- (i) Don't waste doing things manually
- (ii) Have a CI/CD mindset.
- (iii) Automation should be smooth and easy to replicate.

→ Learn

- (i) The time gap between development to deployment must be brought down.
- (ii) Decrease the count of recurring bugs or failures
- (iii) Team must focus on acquiring data on the increase or decrease of count.

→ Measurement & Sharing.

- (i) Pair developers and operators in each phase of application lifecycle.
- (ii) Slogging through the tough patches together makes celebration success all the more sweet.
- (iii) Positive feedback from peers motivates as much as our paychecks and career ambitions.

Develops practices:-

- (i) CI/CD
- (ii) Microservices
- (iii) Infrastructure as Code
 - ↳ Configuration Management
 - ↳ Policy as a Code.
- (iv) Monitoring and logging
- (v) Communication and Collaboration.

* Benefits of Devops:-

- Speed
- Rapid Delivery
- Reliability
- Scale
- Improved Collaboration
- Effective Scaling & deployment Monitoring.

* Barriers to Devops:-

- Culture.
- Legacy
- Application Complexity
- Continuous Testing
- Managing Environment
- Budget.
- Fragmented Tools.

Tools:-

* Configuration Management:-

- Chef
- Salt Stack
- puppet.
- Ansible

* Continuous Integration.

- Jenkins
- TeamCity
- CodeShip
- Circle ci

* Microservices:-

- Docker.
- Mesos
- Elastic Box.

* Collaboration

- JIRA
- Slack
- HipChat
- PaaS.

* Monitoring

- Monit
- Ganglia
- S.Nort
- Cacti

* Development

- Visual Studio
- Apache Active MQ.
- Vagrant
- Microsoft Azure.