DevOps

In practice...

Jayanta Dutta

What it is actually...

- Development + Operation = DevOps
- DevOps combines cultural philosophies, collaborative practices, and tools to accelerate software delivery through automation, fast feedback, and iterative improvement, enabling organizations to deliver applications and services at high velocity.

Core DevOps principles:

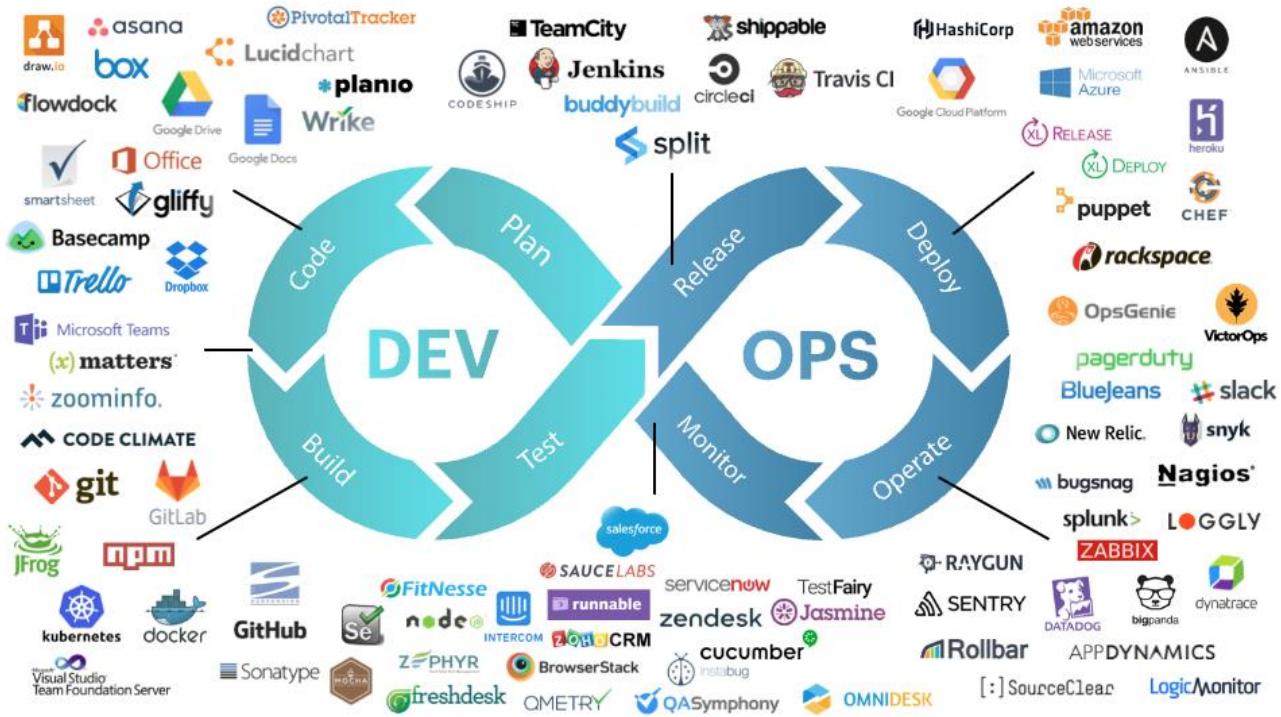
- Collaboration and shared responsibility
- Automation
- Continious Improvement

DevOps Practices

The following are DevOps best practices:

- **Continuous Integration (CI):** Automate the process of integrating code changes from multiple developers into a shared repository frequently.
- **Continuous Delivery (CD):** Automate the process of deploying code changes to staging upto production environments.
- **Configuration Management/Infrastructure as Code (IaC):** Manage and automate the configuration of infrastructure and application components.
- **Microservices Architecture:** Design applications as a collection of loosely coupled, independently deployable services.
- **Containerization:** Package applications and their dependencies into lightweight, portable containers.
- **Automated Testing:** Automate the execution of various types of tests, including unit tests, integration tests, and acceptance tests.
- **Continuous Monitoring/Logging:** Monitor application and infrastructure performance, availability, and security in real-time.
- **Collaborative Culture:** Foster collaboration and communication between development, operations, and other stakeholders.
- **Feedback Loops:** Establish feedback mechanisms throughout the development and operations lifecycle. Gather insights from users, automate feedback loops, and continuously improve processes.

DevOps lifecycle and tools

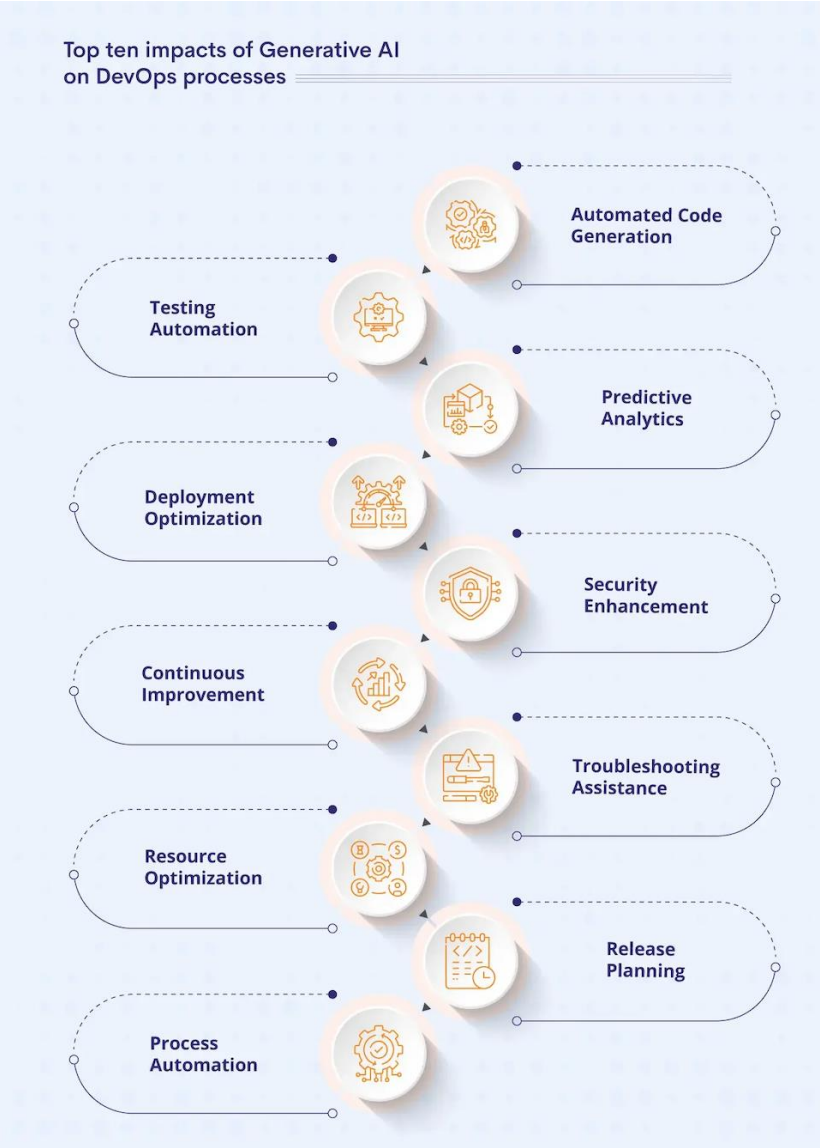


DevOps Metrics: **How Google/DORA Measures Software Delivery Performance**

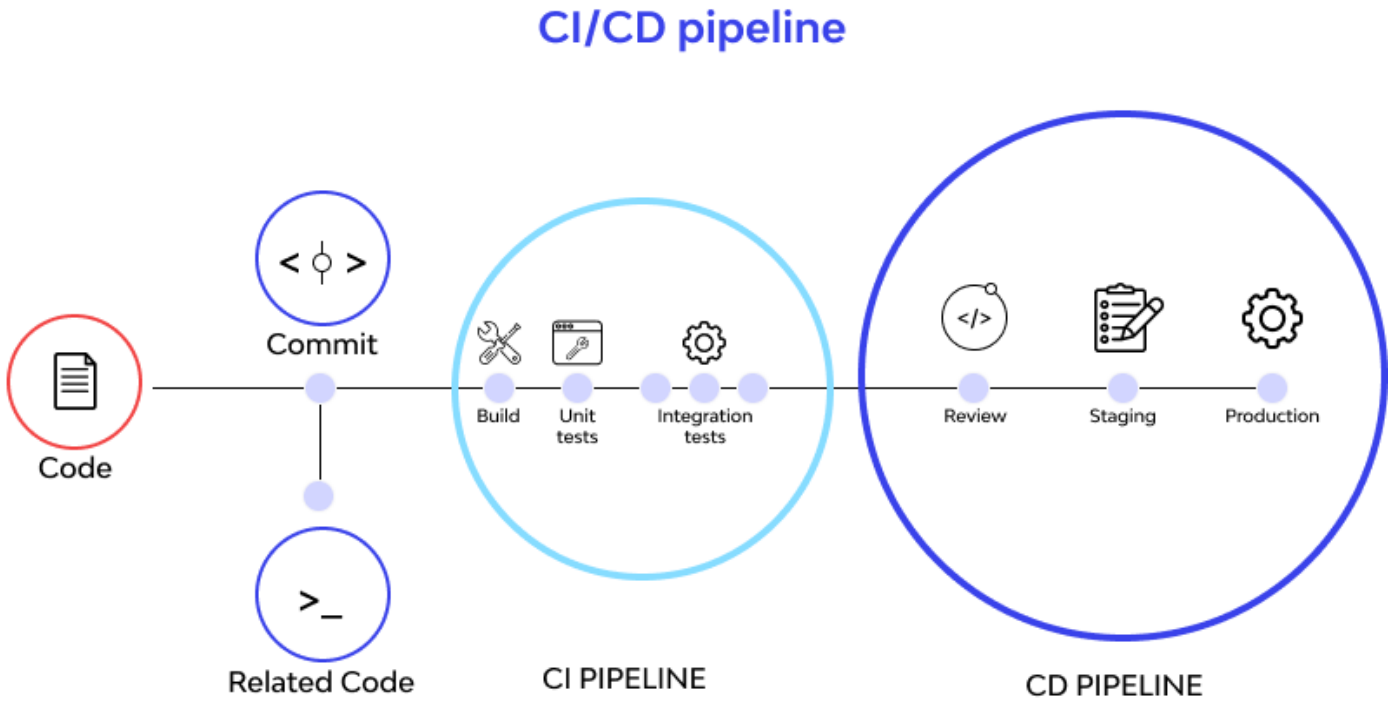
A team can achieve significantly better business outcomes by following these..

Software delivery performance metric	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per week and once per month	Between once per month and once every 6 months	Fewer than once per six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one hour	Between one day and one week	Between one month and six months	More than six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day	Between one day and one week	More than six months
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0%-15%	16%-30%	16%-30%	16%-30%

Gen AI on DevOps



CICD pipeline



Demo time

Deploy an application using DevOps practice into Cloud

Secure way of accessing application

Testing

Monitoring

Metrics

Configure alert if required

Q & A

Thank you !