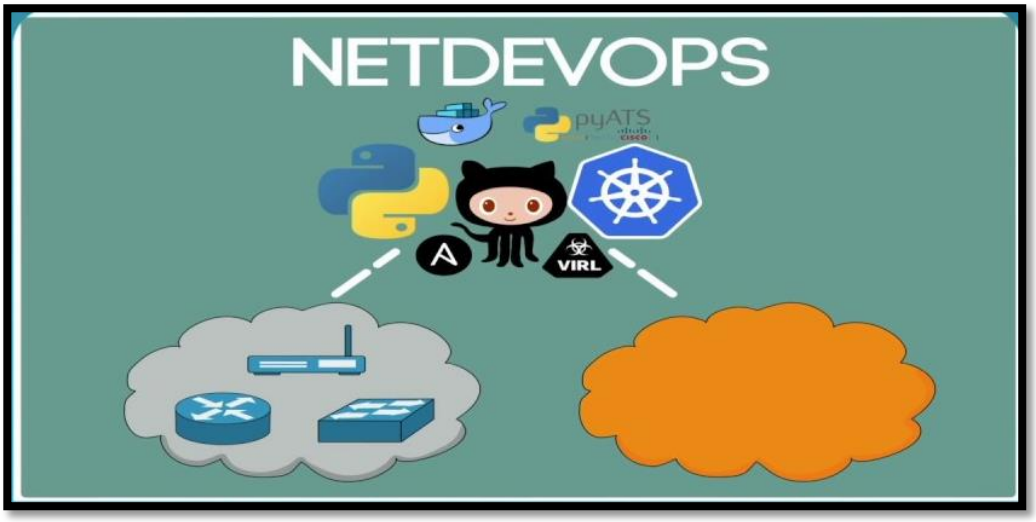


- # What is Network DevOps?

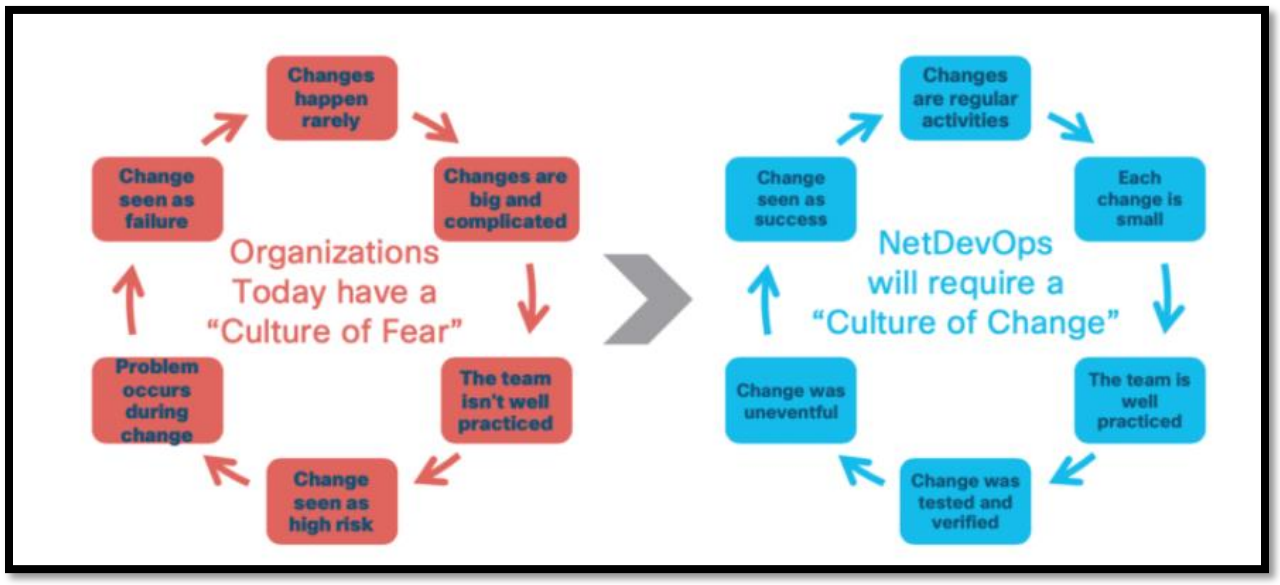
Network DevOps refers to the acceptance of DevOps culture and principles in network engineering and operations. It embodies a preference for network automation including "network as API" concepts, along with continuous development, integration and deployment of new networking technologies.

In its simplest terms, NetDevOps can be defined as an intersection of Networking and DevOps. It is open communication, done through automation and using Infrastructure as Code (IaC). It is the process that provisions and manages data center resources (compute and connectivity) through machine-readable definition files instead of using physical hardware or interactive configuration tools.

- # Network DevOps Requires Automation

DevOps for networking has developed in the context of IT and production teams adopting DevOps and agile development utilizing various programming languages. As software development organizations look to work more collaboratively and in a continuous process from the development stage onward, network teams that are locked in a more manual and siloed way of functioning become chokepoints. Any network operator or service provider trying to maintain that sort of agility can only do so through the systematic introduction of automation principles, including provisioning, configuration, testing, version control and deployment of application and network services.

- **Bringing a Change in Culture**

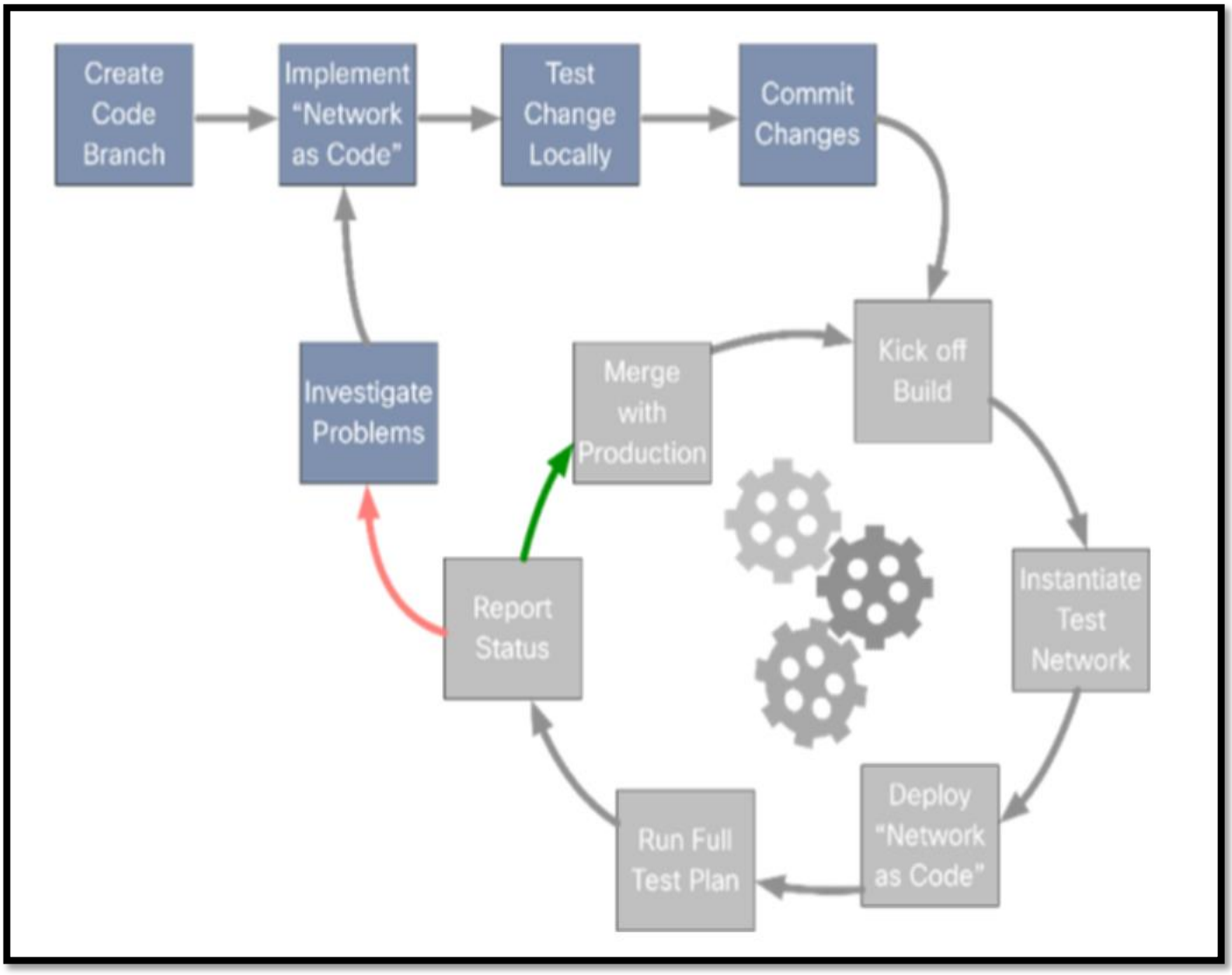


For the purpose of achieving success similar to DevOps in software development, NetDevOps must be transformational. Setting up the culture right is the first step. In order to eliminate cultural fear from the organization, the following cultural changes need to be done.

- Accept failure and create learning from the same for a better future.
- Understand, 'change is the only constant' and is good for further development.
- Establish active collaboration and coordination between developers and operations teams.
- Encourage teams for ownership and responsibility.
- Continuous feedback systems which will escalate iteration and improvement of processes.
- End to end automation for complete change life-cycle.

This shift in the culture will eventually change the way networks are managed, automated and scaled up in an organization. It will ensure effective team collaboration, focusing on improving the flexibility and reliability of the network. Let's explore how it happens.

- **Steps to Embrace NetDevOps**



When Infrastructure as Code approach is applied with network device configurations, it is termed as Network as Code. The idea of Network as Code is to store network configurations of the whole network in a Version Control System (VCS), which manages and tracks changes in the network. The VCS is considered as a Single Source of Truth for all-things network configuration. The below bullets describe the complete network-configuration change loop.

Network configuration changes are projected in the code branches, the place where network developers work on their proposed configurations, without affecting the master branch, having the master configurations.

Once the new configurations are available, developers request their branch to be merged with the master configurations and pass through an approval process so as to verify that no issues are present while incorporating changes.

Further, CI/CD build servers are used for the automatic deployment and examination of the proposed configurations in testing, staging and production environments.

The newly created configurations which successfully pass the tests get deployed into the production environment. Just in case of the occurrence of any failure during the final deployment, the system rollbacks itself automatically with the proposed changes, bringing the production network into its original state.

In place of old, slow and error-prone command-line interfaces, modern interfaces and APIs are used for the deployment of the configuration. With this approach, an environment is established supporting automatic deployment and test configuration changes across the network.

The above process creates a complete automated environment that can deploy and test configuration changes across the network. Though the Dev and Ops teams work to provide a rapid and stable release, a communication gap and tooling differences have been seen between the Networking and DevOps teams.

• Advantages of NetDevOps

▪ Breaking Human Silos:

Much like DevOps, communication is the key to a successful NetDevOps implementation. Consistent communication will encourage effective collaboration and will increase the efficiency of the team members across the enterprise network. When an issue pops up, it is much more practical to have multiple teams working towards the solution of the problem rather than just sitting back and turning a blind eye thinking of it as the other department's problem. This results in tooling expansion between the application, server and networking space. Finding and applying the best solutions across the entire enterprise establishes consistency in between silos.

▪ Less Manual Involvement:

The repetitive logins into the server and making manual changes every time, leading to a change in the network is quite a struggle. Consequently, it wastes a lot of time and is error-prone. IaC leads to a shift from manual mode to automation ensuring automation scripts to be error-free and can be redeployed on multiple servers. Additionally, can be rolled back and is easily accessed by all the teams.

• Conclusion

DevOps has brought cultural transformation in the software development industry. Much like DevOps, a culture of fear is seen inside the organization affecting the capability of the networking teams to deliver services quickly and efficiently. The efficiency of an organization will enhance when networking and DevOps principles are integrated together for similar goals Network automation and IaC is the key for NetDevOps implementation.