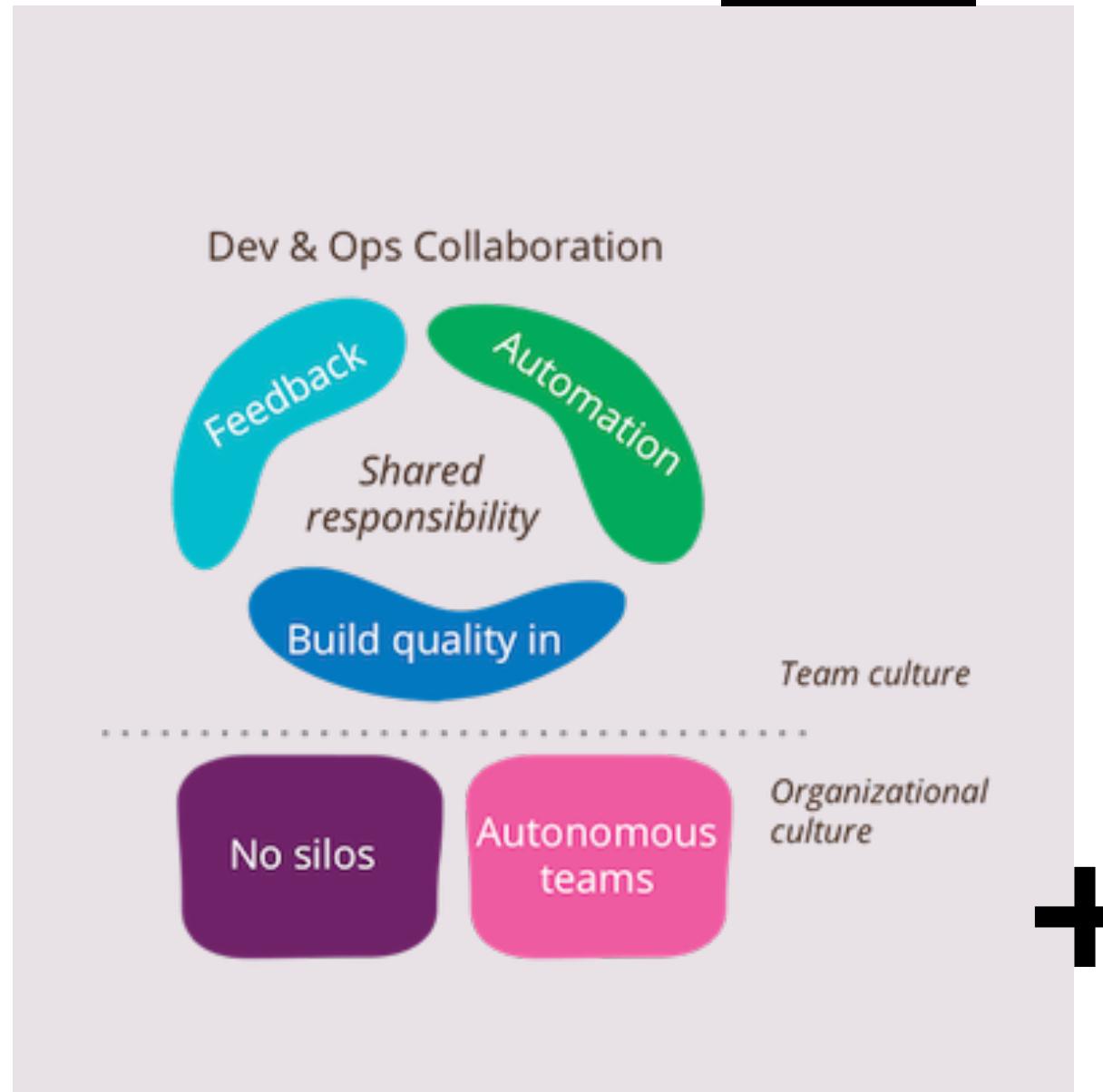


# Introduce DevOps pipeline

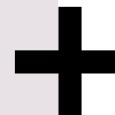
Thang Nguyen



# What is DevOps



# What is DevOps



# What is DevOps

## What's DevOps?

DevOps is a full lifecycle investment



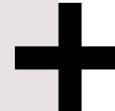
DevOps is a team undertaking



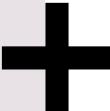
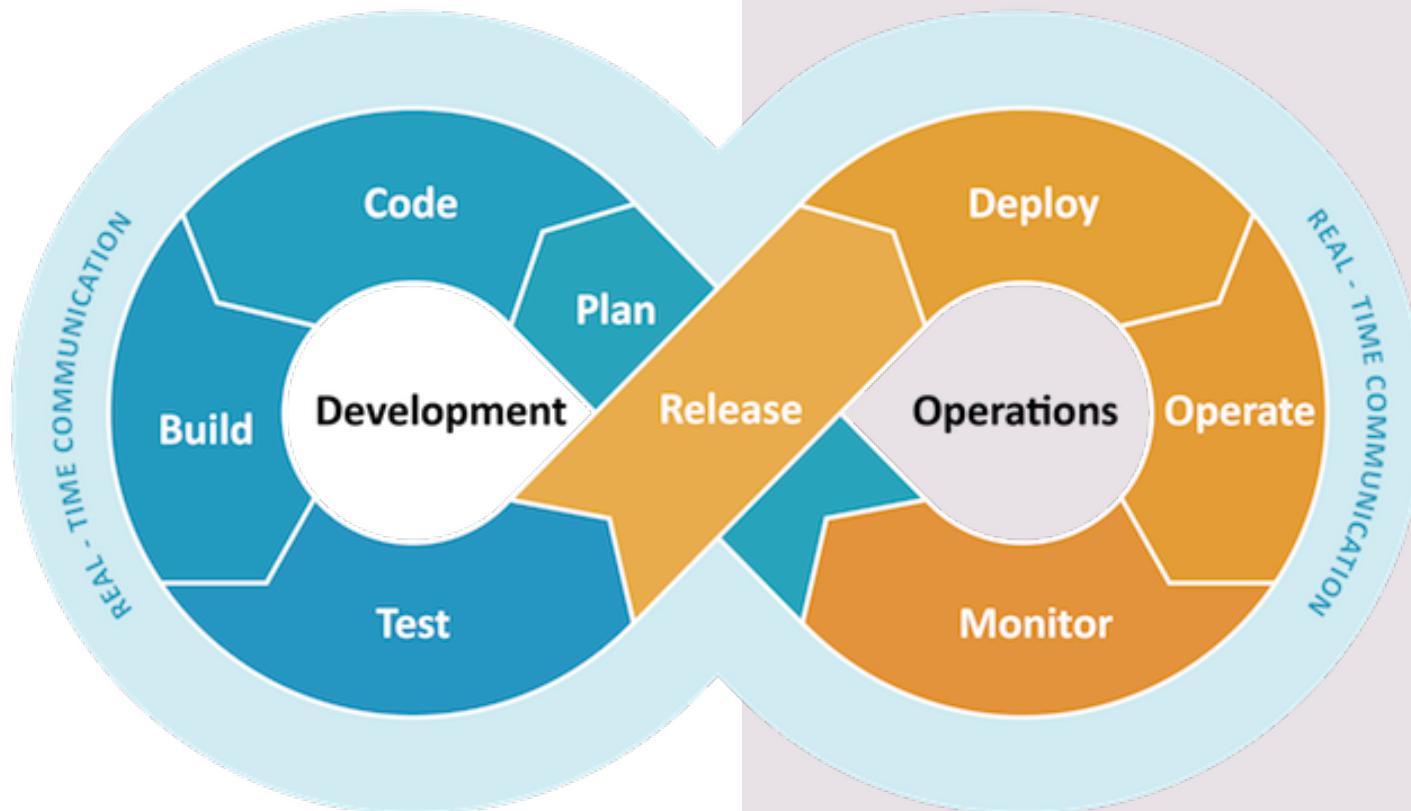
DevOps enables better software development & delivery practices



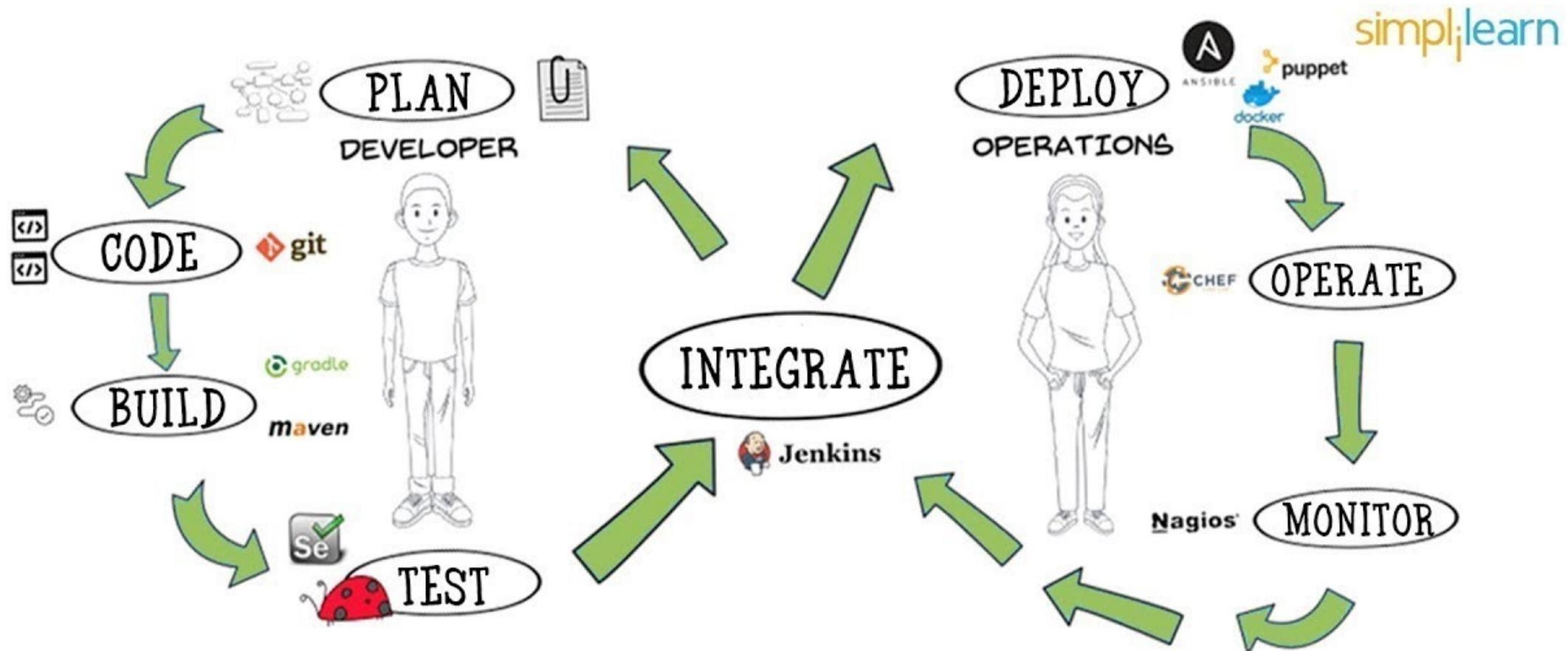
DevOps accelerates the last mile of continuous delivery



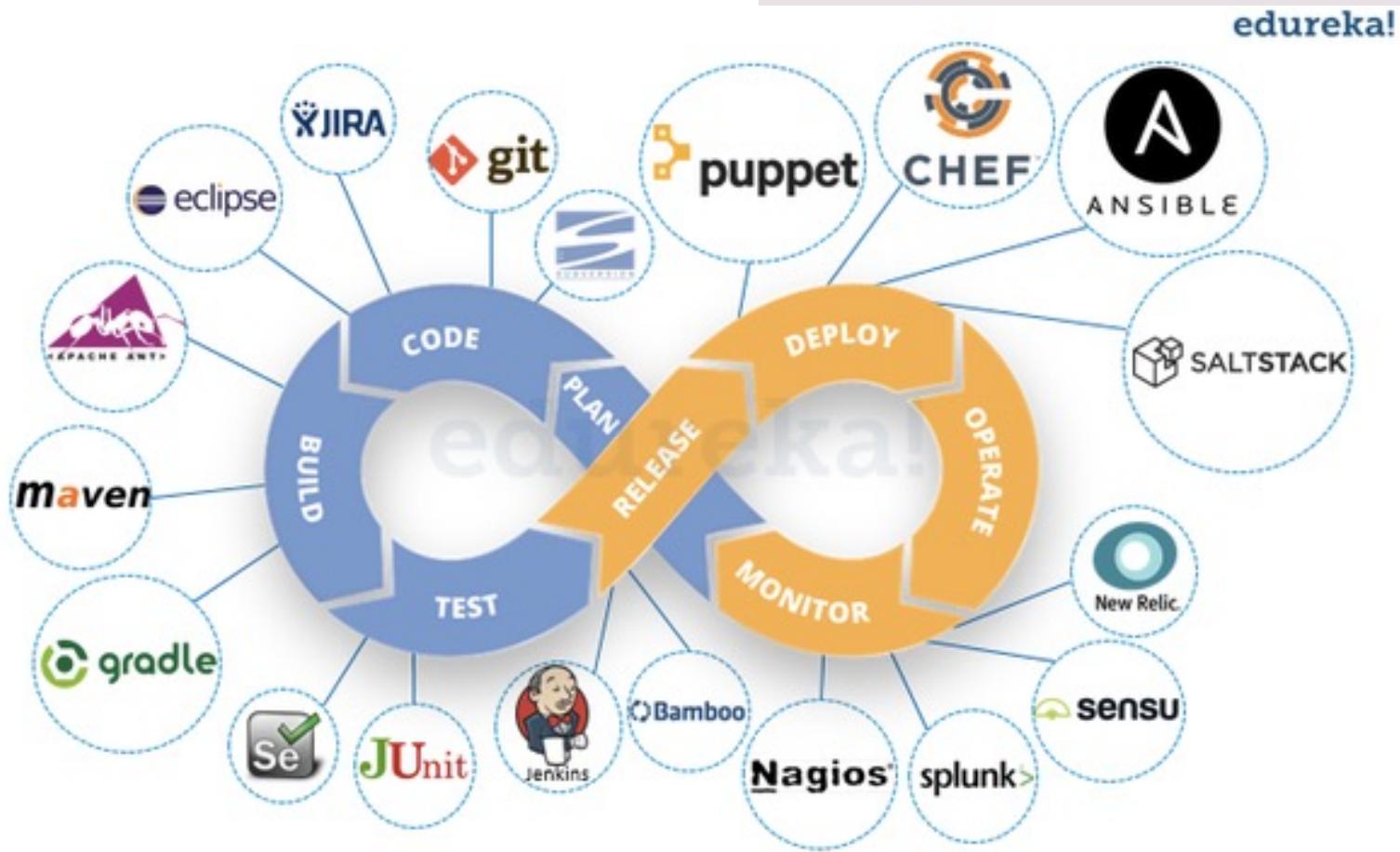
# What is DevOps



# What is DevOps

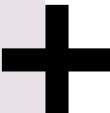


# What is DevOps



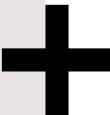
# What we need?

- **Learn the culture**
  - Concept
  - Motivation
  - Factor Influencing
  - Barriers



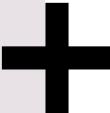
# What we need?

- **Learn Server Administrator**
  - Linux OS
  - Linux Commands
  - File, permission, network
  - ...



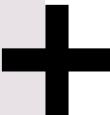
# What we need?

- **Learn Webserver, Network, Protocol**
  - TCP/IP
  - SSL, SSH, FTP...
  - Firewall, Load Balancer
  - Apache, Tomcat, IIS,...



# What we need?

- **Learn Scripting & Programming language**
  - IaC
  - PaC
  - Other languages



# What we need?

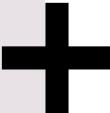
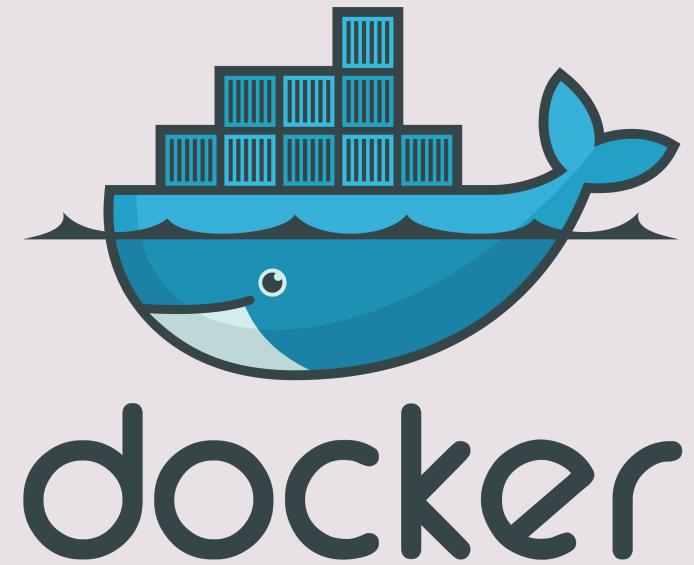
- **Learn Git**
  - Best of SCM



+

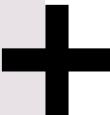
# What we need?

- **Learn Container Architect**
  - Docker
  - K8s
  - Rancher



# What we need?

- **Learn Configuration Mgmt**
  - Ansible
  - Chef
  - Puppet



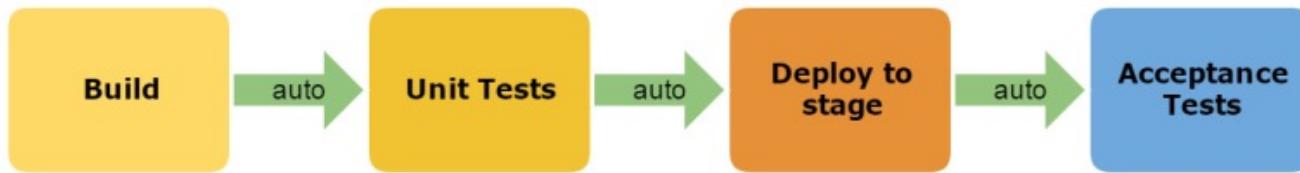
# What we need?

- Learn Cloud IaC
  - Google Cloud
  - AWS



# CI/CD

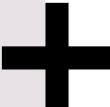
## Continuous Integration



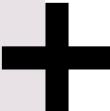
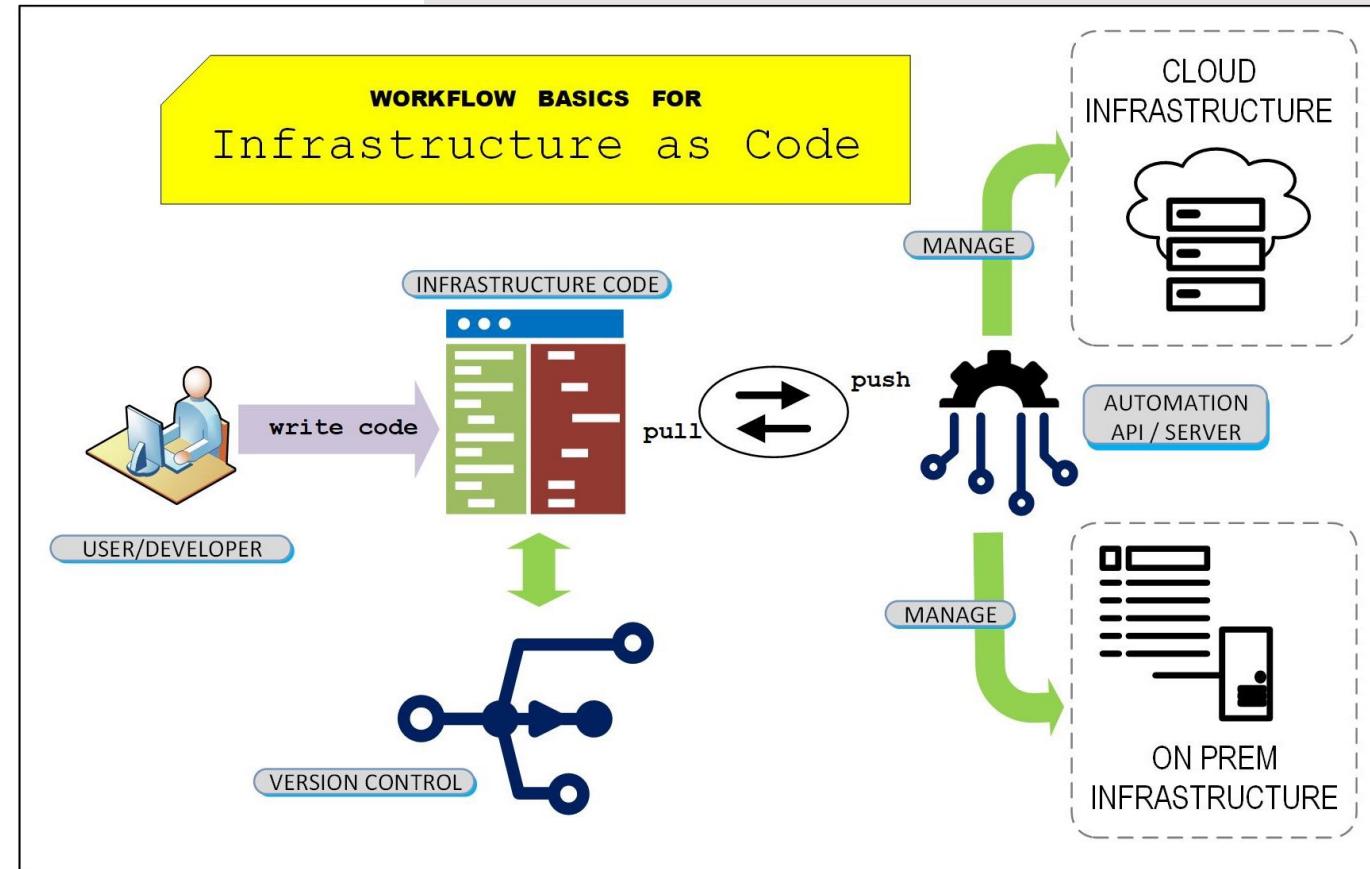
## Continuous Delivery



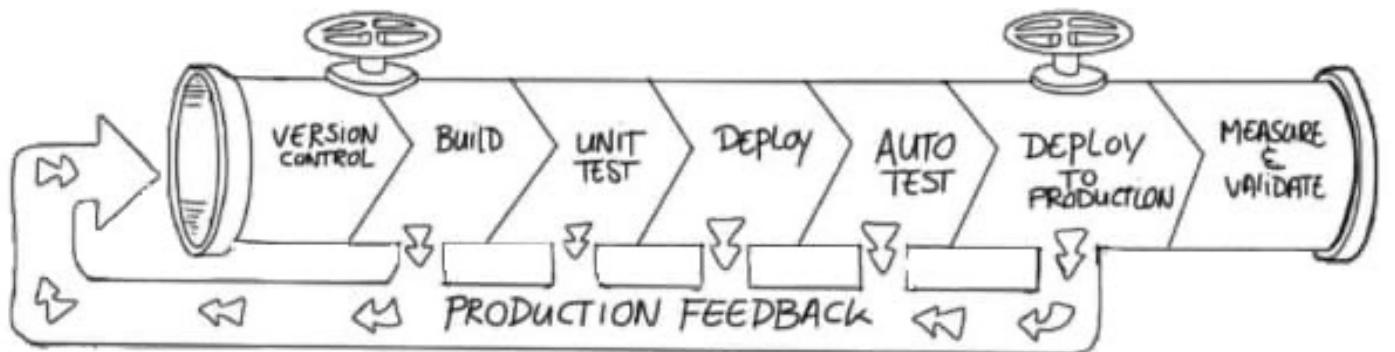
## Continuous Deployment



# IaC Concepts



# PaC Concepts



[Extension Development Host] - Jenkinsfile - Visual Studio Code

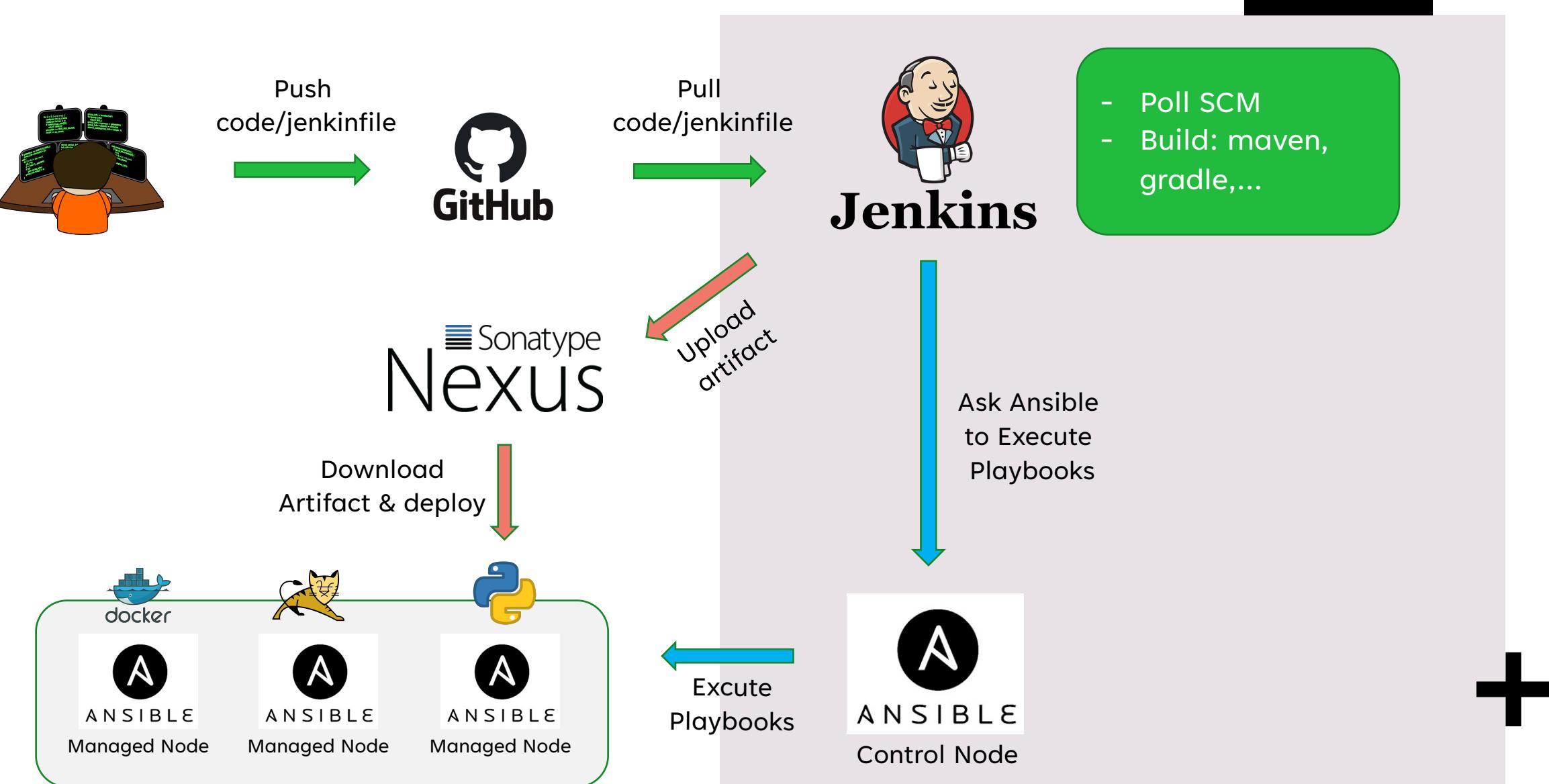
```
File Edit Selection View Go Debug Tasks Help
Jenkinsfile x
1 pipeline {
2     agent any
3
4     stages {
5         stage('Build') {
6             steps {
7                 echo 'Building..'
8             }
9         }
10        stage('Test') {
11            steps {
12                echo 'Testing..'
13            }
14        }
15    }
16}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Jenkins Pipeline Linter

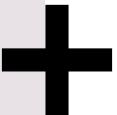
Ln 1, Col 11 Spaces: 4 UTF-8 CRLF Groovy

# Chosen pipeline



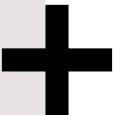
# Demo Projects

- Simple Flask Python
- Jenkinfile



# Installation steps

- Setup 4 Ubuntu servers
  - Virtual Box
  - Ubuntu 18.04



# Installation steps

## – Install Jenkins

```
wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
sudo apt update
sudo apt install jenkins
sudo systemctl start jenkins
```

# Installation steps

Getting Started

## Customize Jenkins

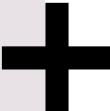
Plugins extend Jenkins with additional features to support many different needs.

### Install suggested plugins

Install plugins the Jenkins community finds most useful.

### Select plugins to install

Select and install plugins most suitable for your needs.



# Installation steps

## – Config Jenkins

- Initial Password: `sudo cat /var/lib/jenkins/secrets/initialAdminPassword`
- Install Plugins: Nexus Artifact Uploader, Github Plugin, Git Plugin, Publish Over SSH
- Config Plugins: Publish Over SSH (Config System)
- New Item > Pipeline Items
  - Config Pipeline from SCM
  - Poll SCM
- Change Jenkins build Step
- Test by Build Now

# Installation steps

## – Install Nexus

```
sudo apt-get update  
sudo apt install openjdk-8-jre-headless  
cd /opt  
sudo wget https://download.sonatype.com/nexus/3/latest-unix.tar.gz  
tar -zxvf latest-unix.tar.gz  
sudo mv /opt/nexus-3.30.1-01 /opt/nexus
```

# Installation steps

## – Install Nexus

```
sudo adduser nexus
```

```
sudo visudo
```

Add new line: “nexus ALL=(ALL) NOPASSWD: ALL” and save

```
sudo chown -R nexus:nexus /opt/nexus
```

```
sudo chown -R nexus:nexus /opt/sonatype-work
```

```
sudo nano /opt/nexus/bin/nexus.rc
```

Uncomment the line and put run\_as\_user="nexus"

# Installation steps

- Install Nexus

Change JVM heap size in `/opt/nexus/bin/nexus.vmoptions`

```
-Xms1024m  
-Xmx1024m  
-XX:MaxDirectMemorySize=1024m  
-XX:LogFile=./sonatype-work/nexus3/log/jvm.log  
-XX:-OmitStackTraceInFastThrow  
-Djava.net.preferIPv4Stack=true  
-Dkaraf.home=.  
-Dkaraf.base=.  
-Dkaraf.etc=etc/karaf  
-Djava.util.logging.config.file=/etc/karaf/java.util.logging.properties  
-Dkaraf.data=./sonatype-work/nexus3  
-Dkaraf.log=./sonatype-work/nexus3/log  
-Djava.io.tmpdir=./sonatype-work/nexus3/tmp
```

# Installation steps

- Install Nexus

Run Nexus as service:

`sudo nano /etc/systemd/system/nexus.service` and put content:

```
[Unit]
Description=nexus service
After=network.target
[Service]
Type=forking
LimitNOFILE=65536
ExecStart=/opt/nexus/bin/nexus start
ExecStop=/opt/nexus/bin/nexus stop
User=nexus
Restart=on-abort
[Install]
WantedBy=multi-user.target
```

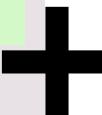
# Installation steps

## – Start Nexus

```
sudo systemctl start nexus
```

```
sudo systemctl enable nexus
```

```
sudo systemctl status nexus
```



# Installation steps

- Config Nexus

- Access via Web , port: 8081
- Initial password : cat /opt/nexus/sonatype-work/nexus3/admin.password
- Add repo: DeployPython (raw)

- Config Jenkins:

- Change Jenkins Upload Step using Syntax Generator

# Installation steps

- Install Ansible on Controller Node

```
sudo apt-add-repository ppa:ansible/ansible
```

```
sudo apt update
```

```
sudo apt install ansible
```

```
sudo nano /etc/ansible/hosts
```

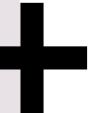
Change file content:

```
[servers]
```

```
docker ansible_host=192.168.1.141
```

```
[all:vars]
```

```
ansible_python_interpreter=/usr/bin/python3
```



# Installation steps

- Config Ansible – add ansibleadmin

```
sudo useradd ansibleadmin
```

```
sudo visudo and put line
```

```
"ansibleadmin    ALL=(ALL)    NOPASSWD: ALL"
```

```
sudo nano/etc/ssh/sshd_config change to PasswordAuthentication yes
```

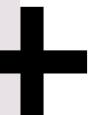
```
(also add user ansibleadmin on Managed node)
```

- Generate keys:

```
ssh-keygen
```

```
ssh-copy-id <ip managed node>
```

```
Test by ssh without password.
```



# Installation steps

- Install docker on Managed Node

```
sudo apt update
```

```
sudo apt install apt-transport-https ca-certificates curl software-properties-common
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable"
```

```
sudo apt update
```

```
apt-cache policy docker-ce
```

```
sudo apt install docker-ce
```

```
sudo systemctl status docker
```

```
Add ansibleadmin to docker group: sudo usermod -aG docker ansibleadmin
```

```
sudo apt-get install python-docker
```

# Installation steps

- Change Jenkinfile to add Deploy
  - Write: DeployDocker.yaml on Ansible Controller Node
  - Change Jenkin to sshUpload to control Ansible Controller Node to deploy on Managed Node
- Test by Build Now

# Q&A

