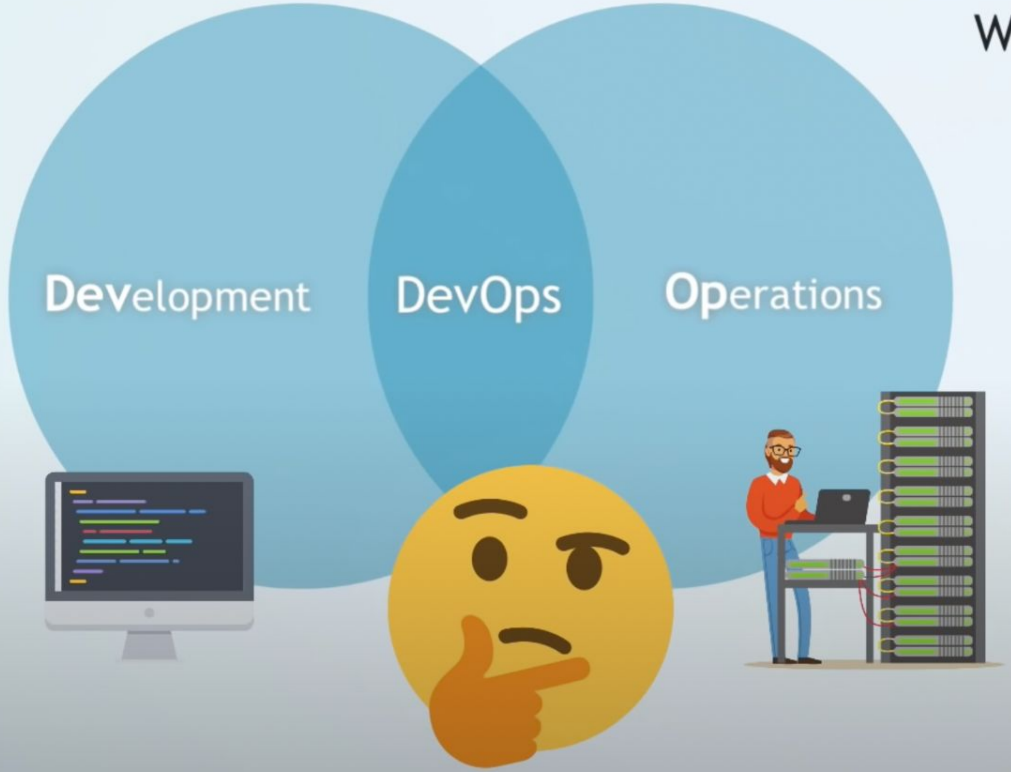
DevOps Basics

"DevOps"

Intersection of Dev & Ops ?

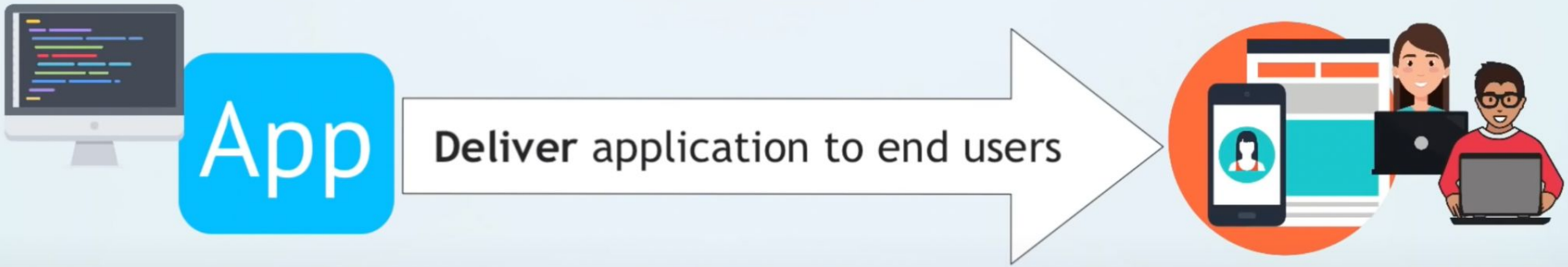


Where do **boundaries start and end?**

Which part of Dev is not DevOps?

Same Process of: **Deliver** application to the end user

- This is always the **main goal**!



Source Code Repository: Git/GitHub/ BitBucket/ SVN/ GitLab/ CodeCommit(AWS)
Jenkins: Bamboo/Code Pipeline(AWS)/ GitHub Actions
Configuration Management Tools: Ansible/Chef/ Puppet/ SaltStack
Build Tools: Maven/ NPM/ Gradle
Test: Selenium/ Junit
Containerisation: Dockers/Rocket
Orchastrators: K8s/Docker Swarm/ EKS(AWS)/ ECS(AWS)
Monitoring: Grafana/Prometheus/Nagios/DataDog/ELK/Splunk/GrayLog
Infra Provision Tools: Terraform/ CloudFormation(AWS)

Typical Software Release Process

Idea

Requirements

Code it

Test it

Build &
Package

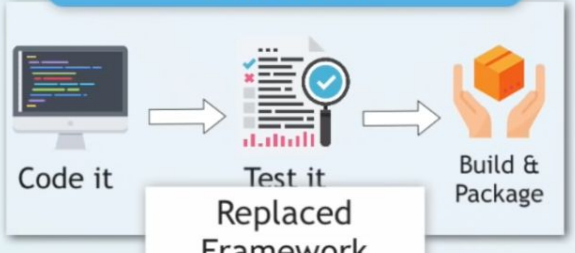
Deploy it

Software versioning

1 . 4 . 2

Major Minor Patch

Version 1



1.0.0

Version 2

1.1.0

Version 3

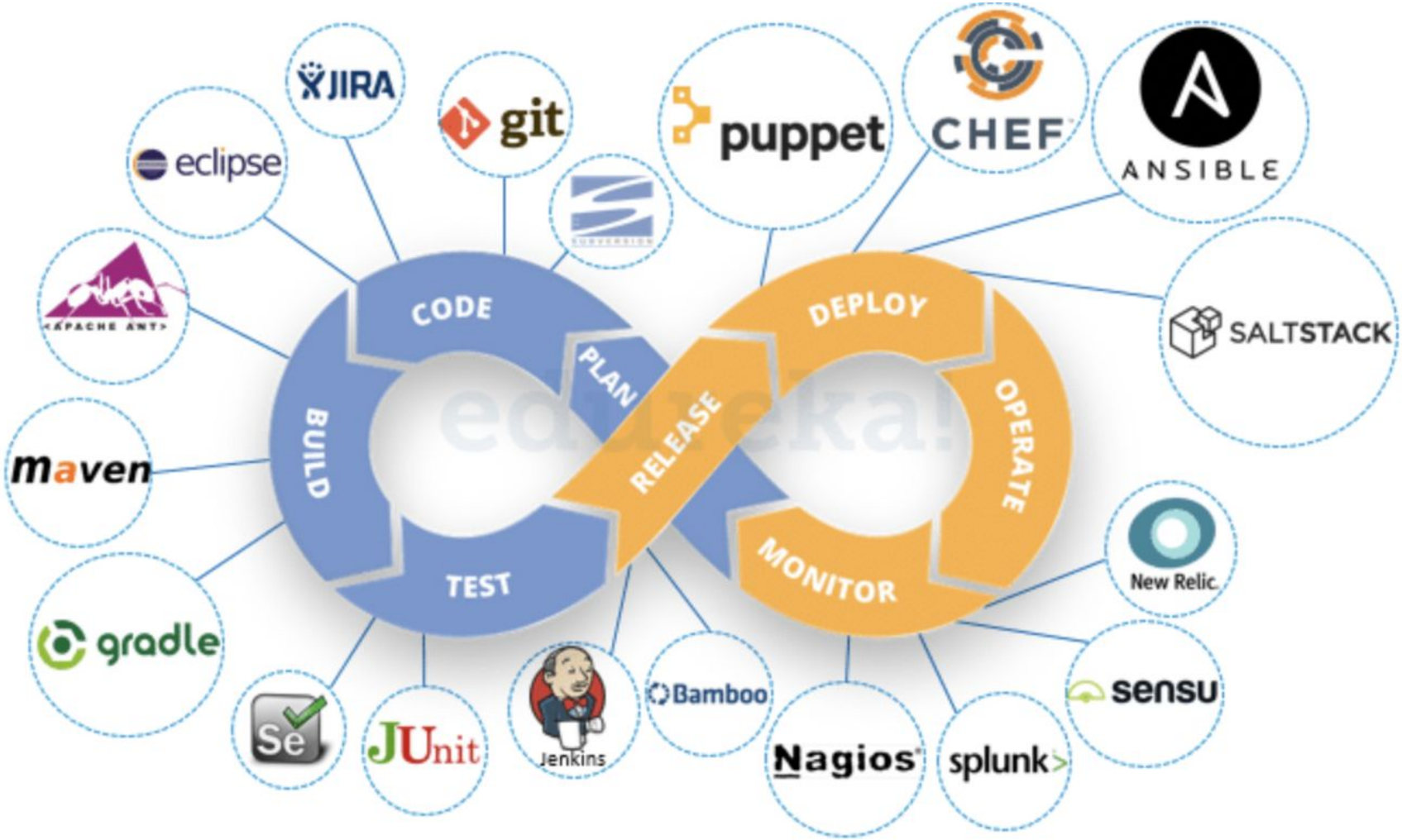
1.1.1

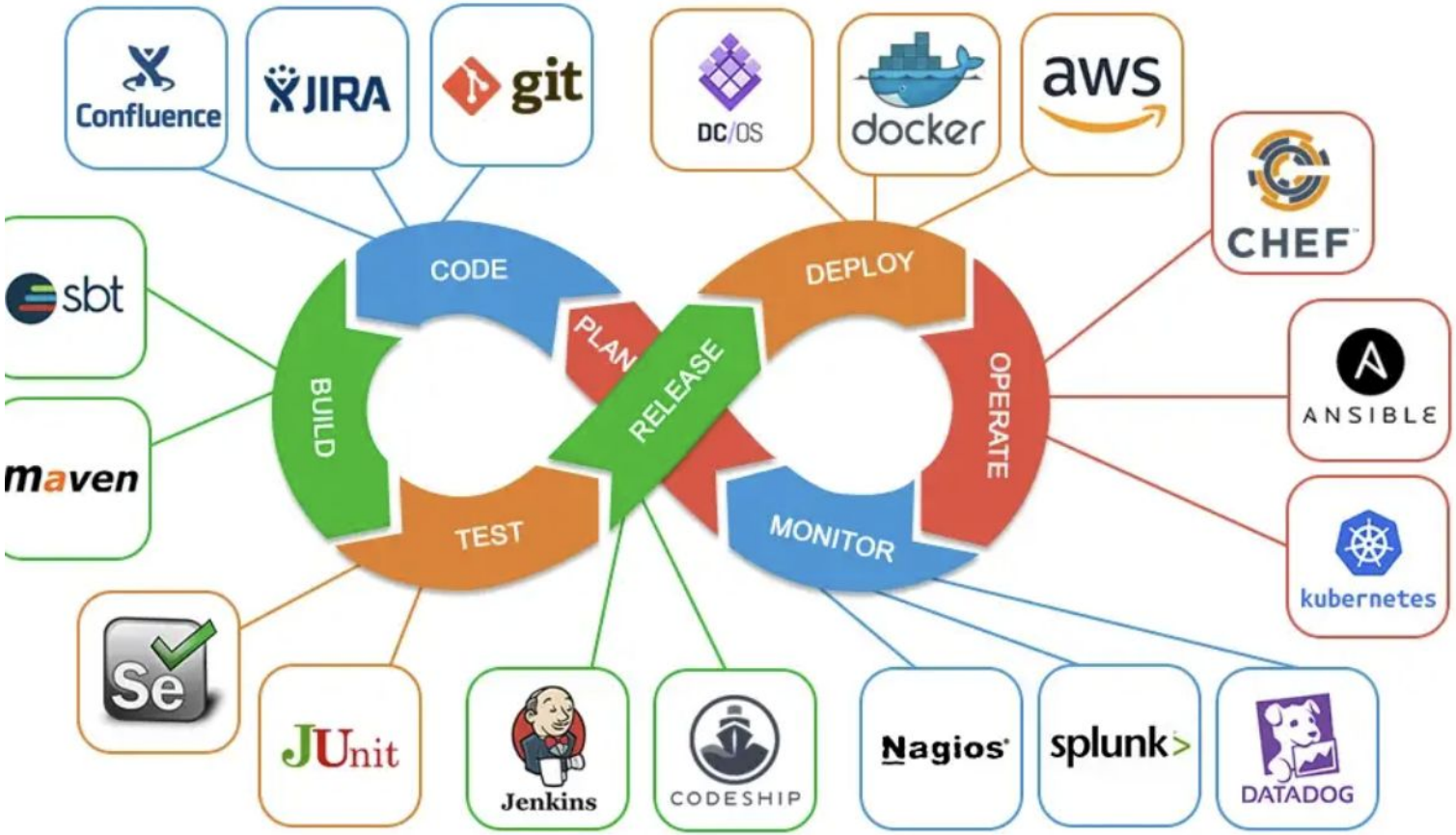
Bugfix

Version 4

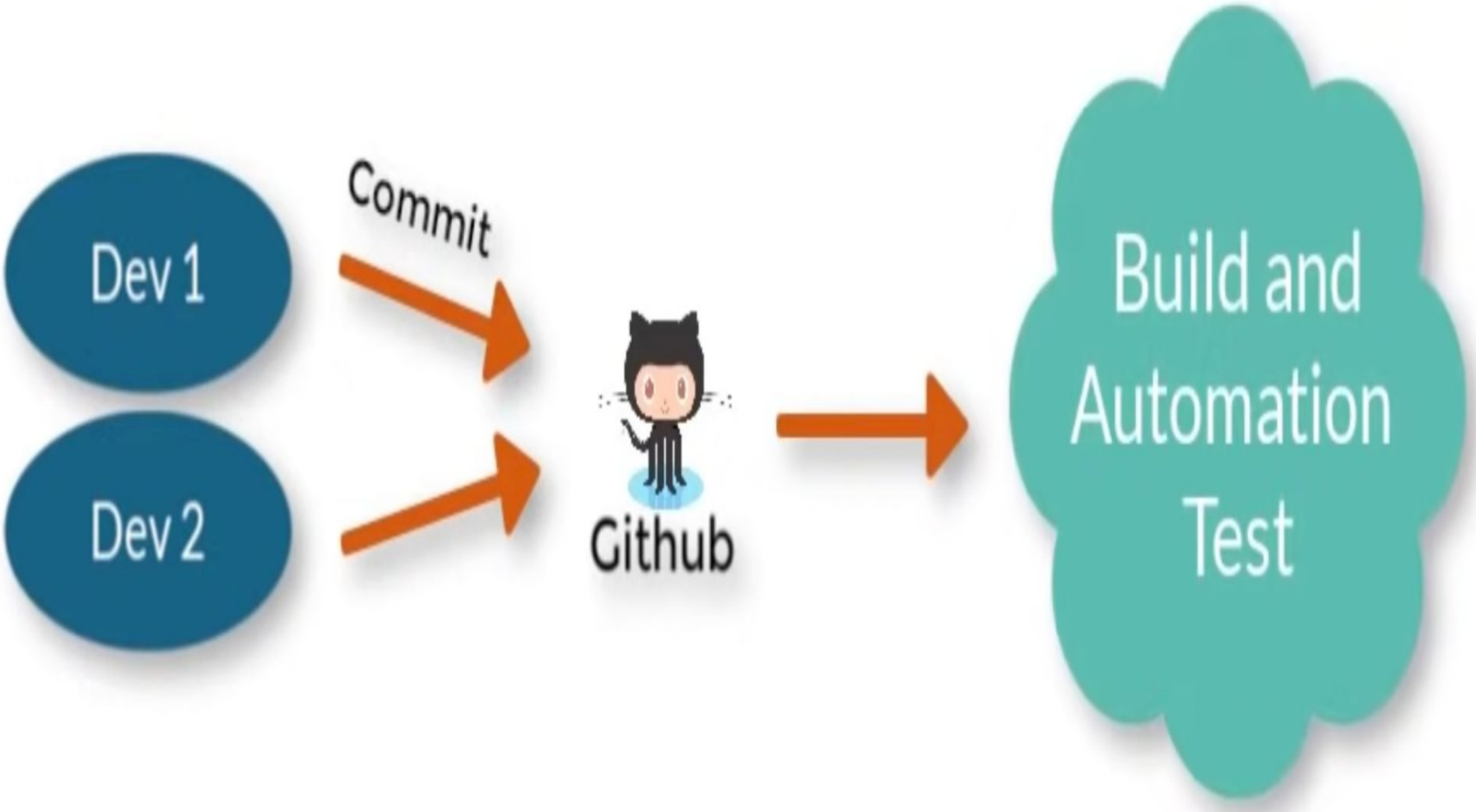
1.1.2

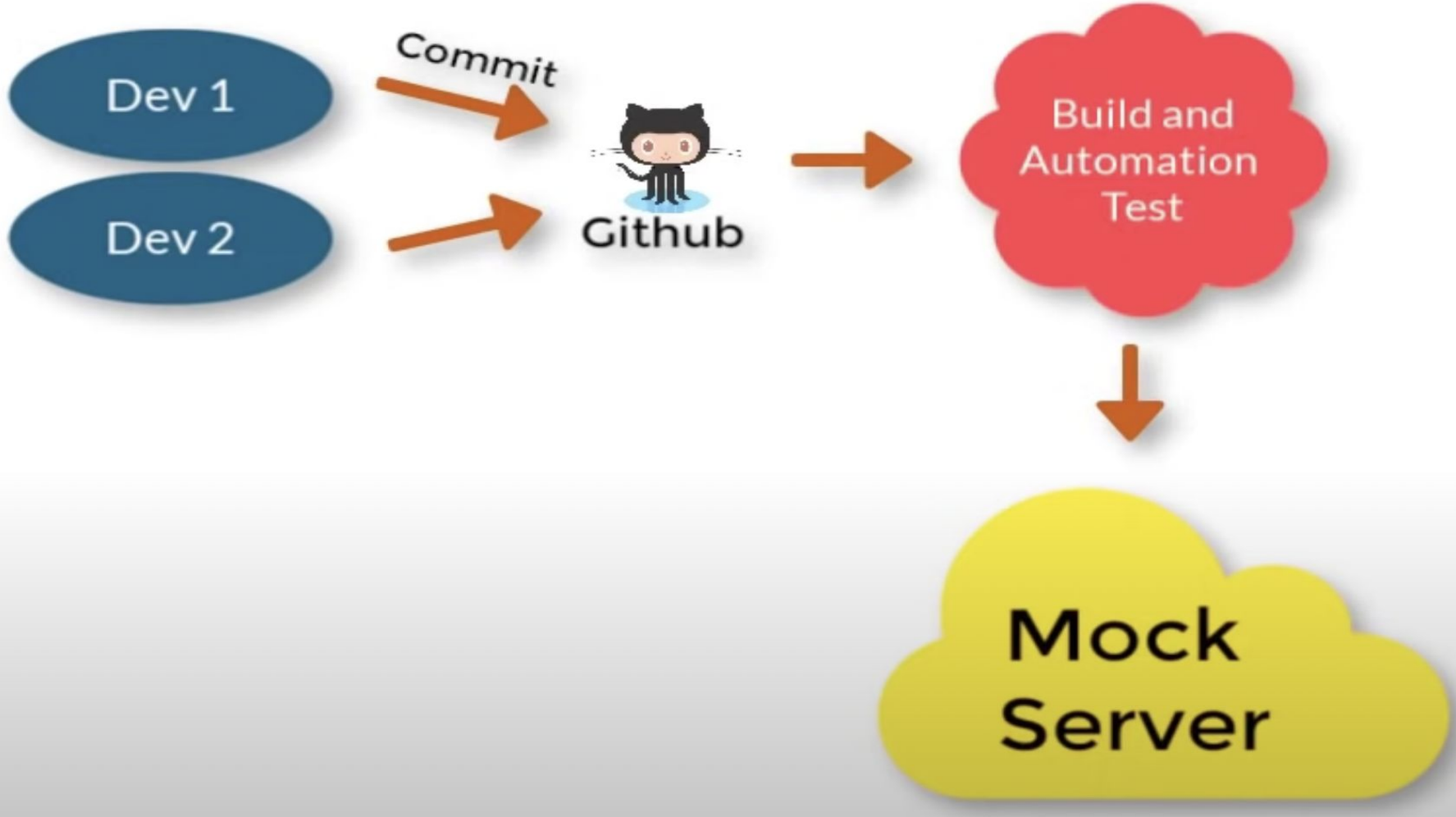
Bugfix

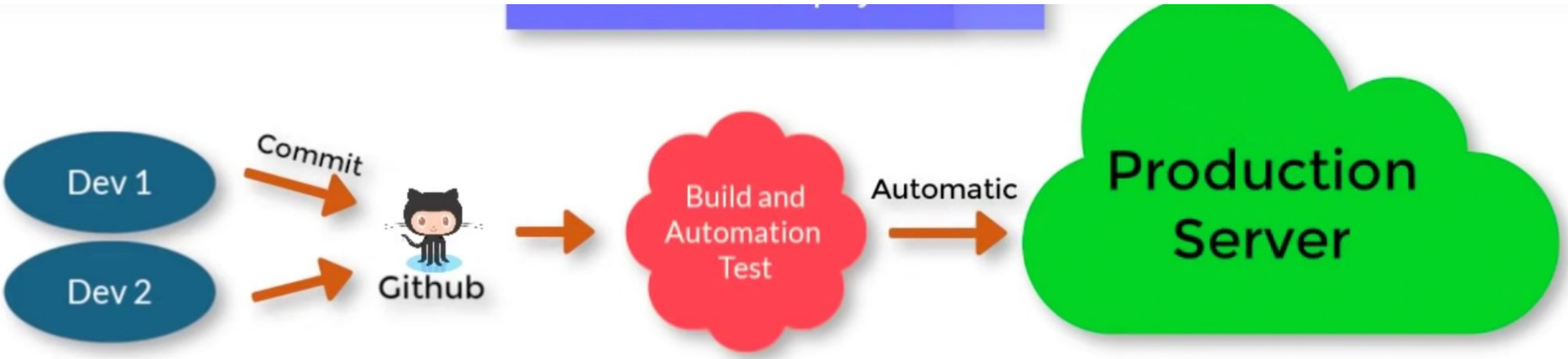

JIRA
git
eclipse
puppet
CHEF
ANSIBLE
SALTSTACK
CODE
PLAN
BUILD
TEST
RELEASE
DEPLOY
OPERATE
MONITOR
maven
gradle
New Relic
sensu
Bamboo
Jenkins
JUnit
Nagios
splunk
edureka!



Devops Lifecycle Phases & Related Tools.


Dev 1
Dev 2
Commit
Github
Build and
Automation
Test


Dev 1
Dev 2
Commit
Github
Build and Automation Test
Mock Server


Dev 1
Dev 2
Commit
Github
Build and Automation Test
Automatic
Production Server

Development vs Staging vs Production environment

Pipeline integration

Assignment

Create ec2 instance ubuntu

Install git

Fork existing repository and clone it

benefits of using DevOps with cloud

1. **Easy Automation**
2. **Effective Monitoring**
3. **Rapid Deployment**