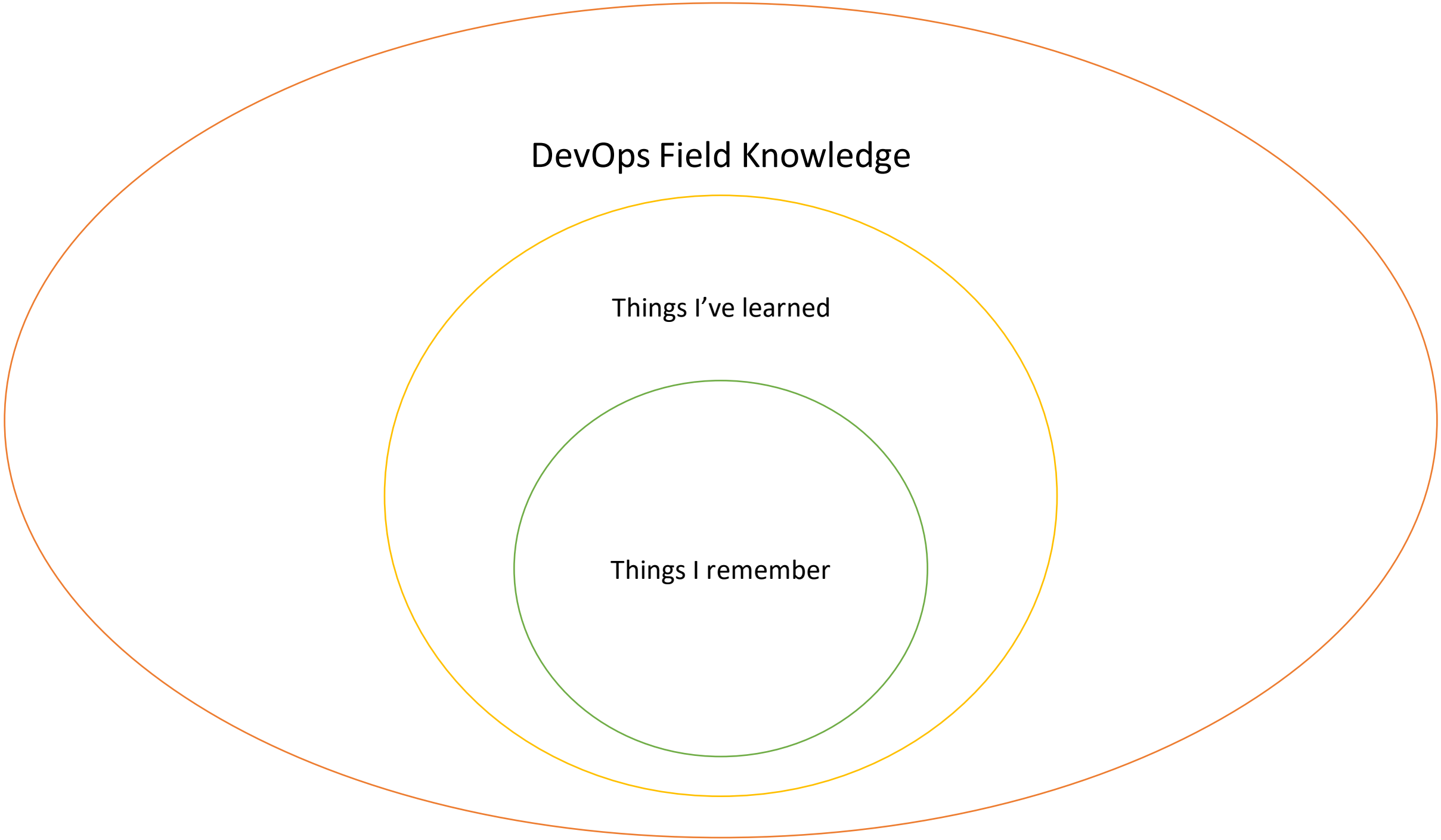
DevOps, Linux & Open Source

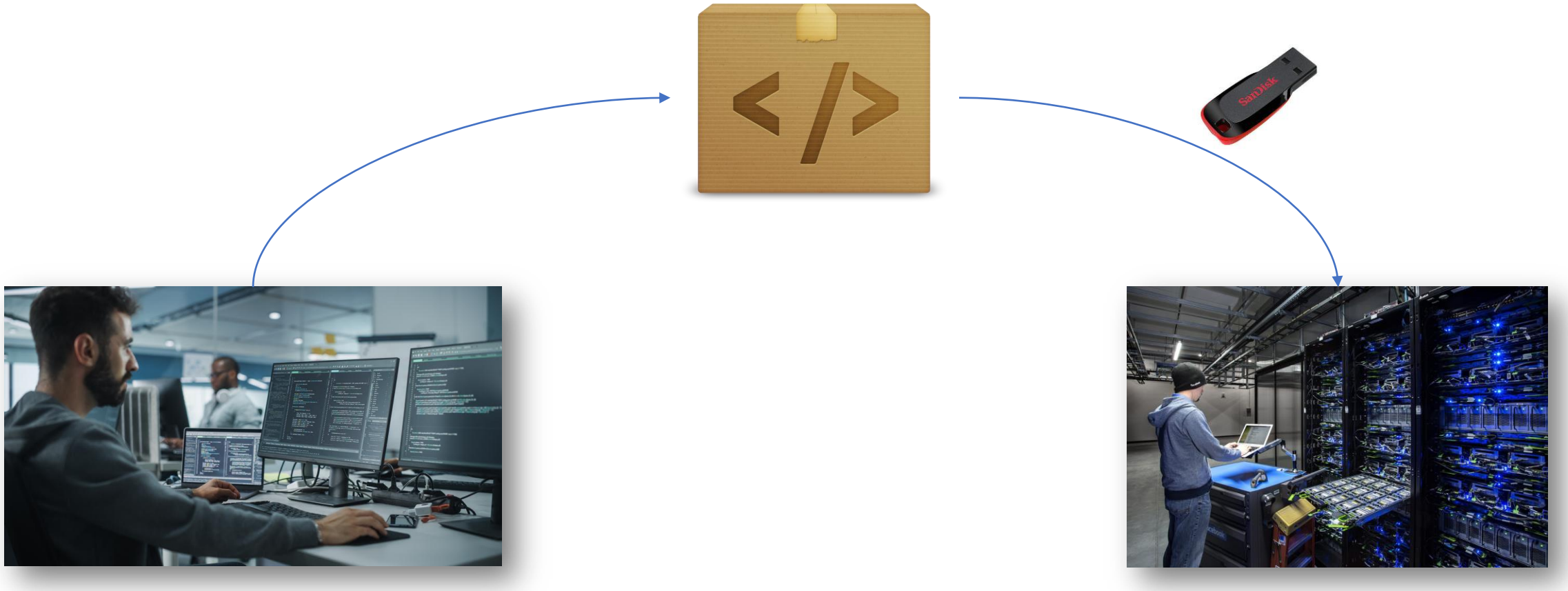
Srikar K.C.

July 20, 2024

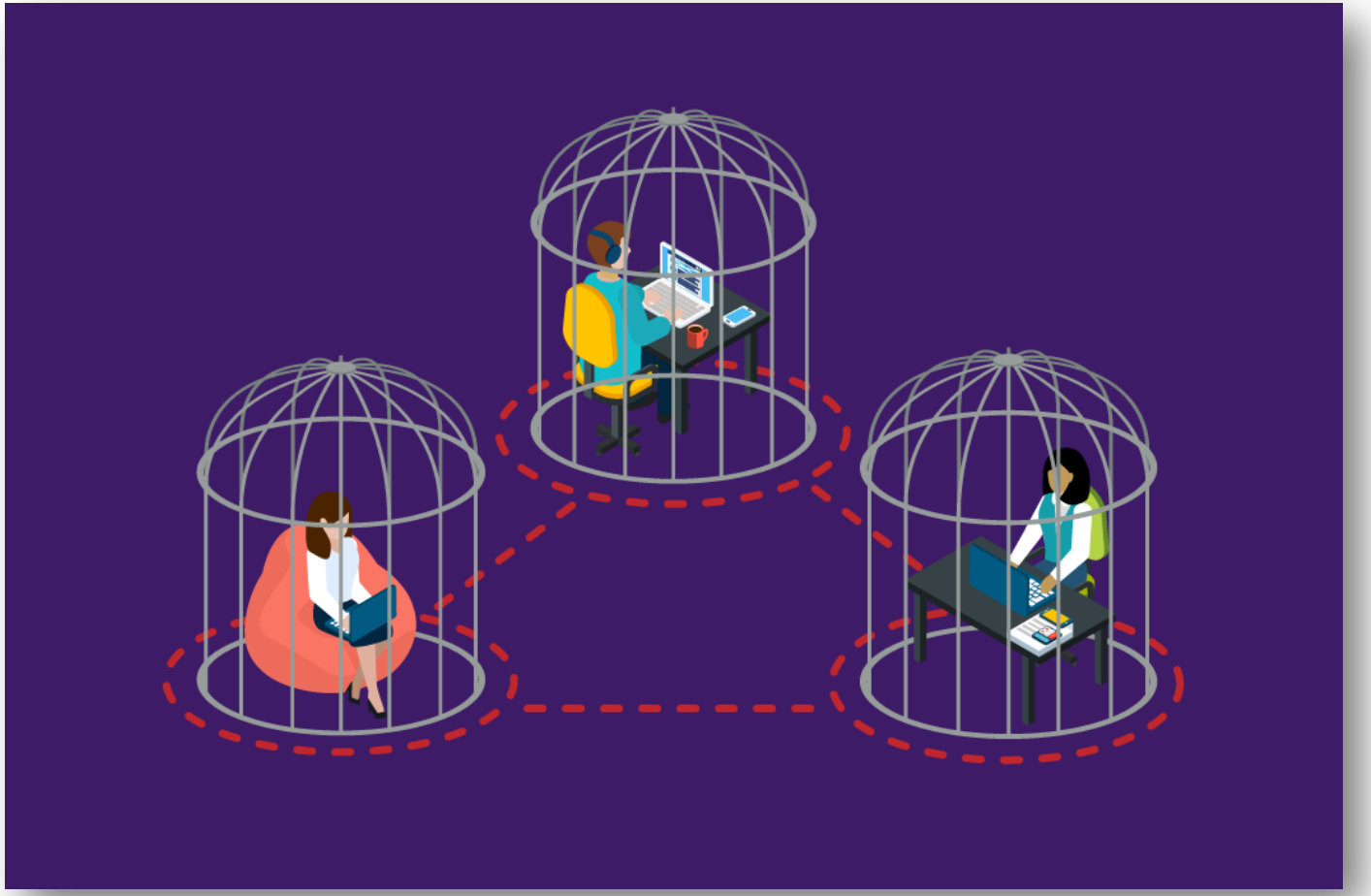
DISCLAIMER: All views & opinions are my own and DO NOT reflect those of my employer.


DevOps Field Knowledge
Things I've learned
Things I remember

Before DevOps practices



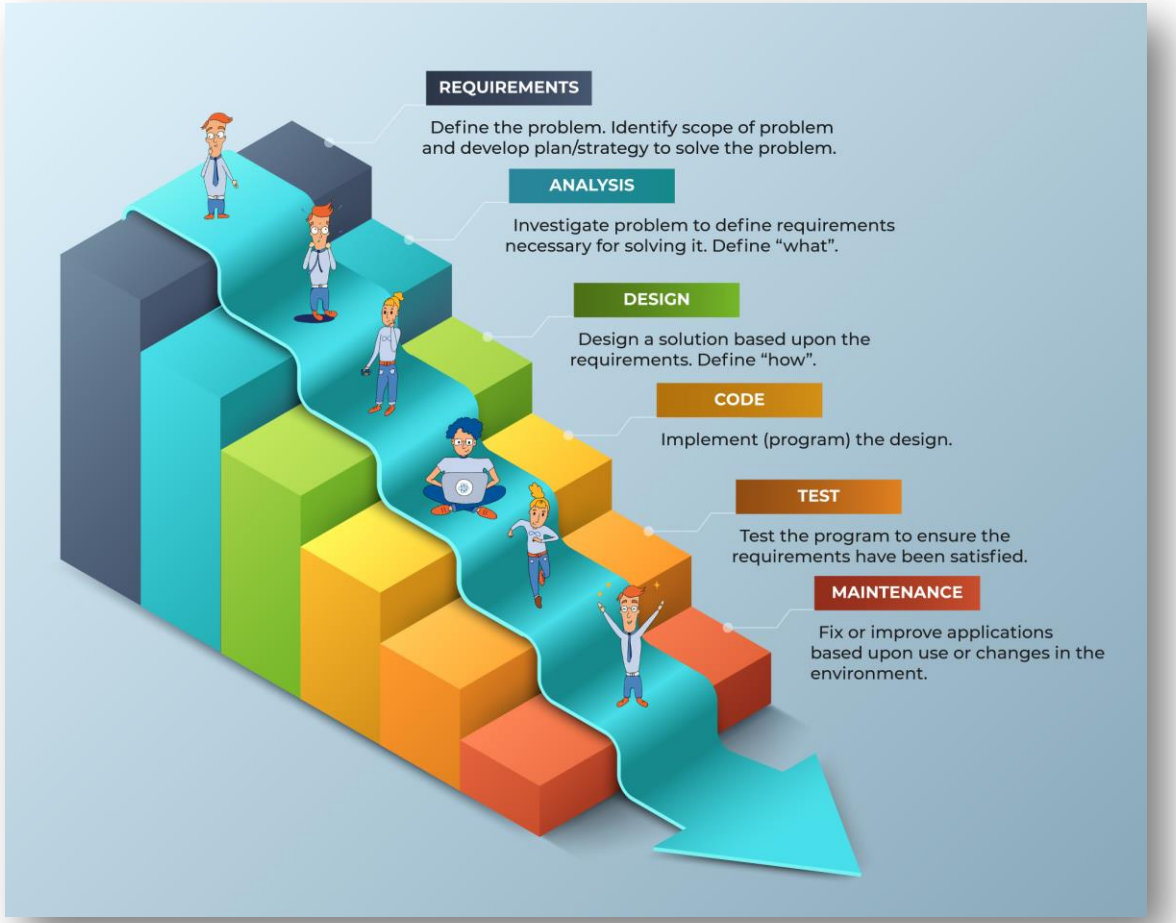
Issue 1 – Siloed teams



Issue 2 – Inconsistent environments

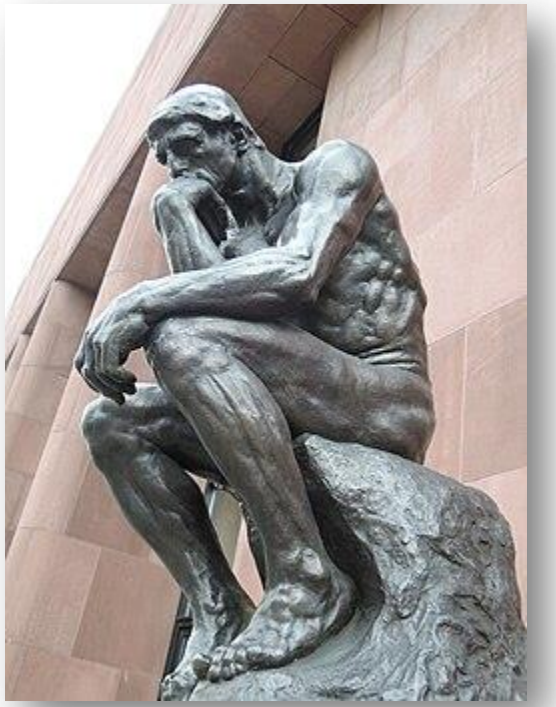
Issue 3 – Slow Deployment cycles

Waterfall SDLC

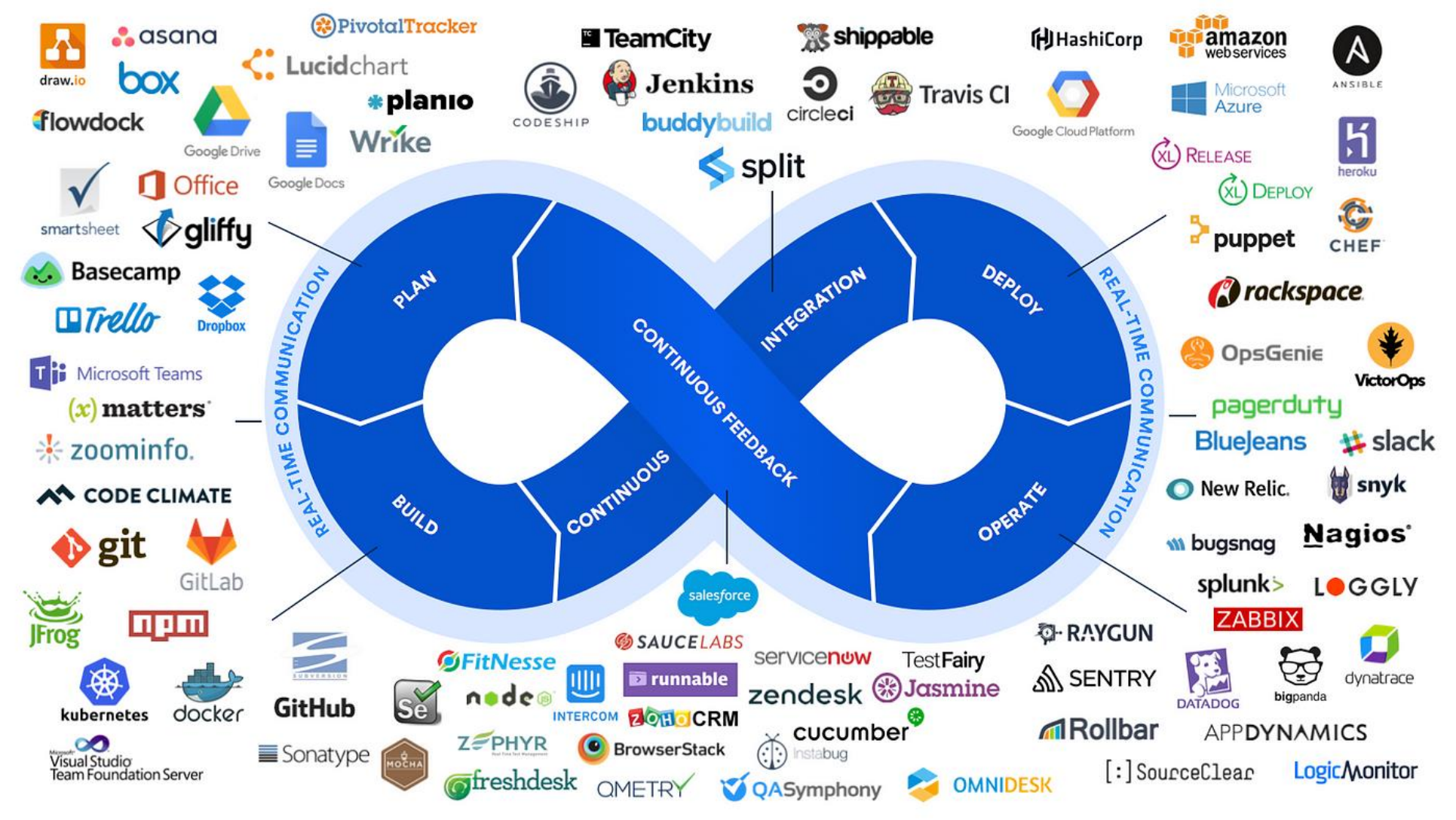


Issue 4 – Security Concerns

What is DevOps?



DevOps Tools



Agile Methodology

Agile Terminology

1. Sprint
2. Daily Stand-ups
3. Spring Planning
4. Sprint Review
5. Sprint Retrospective
6. User Stories
7. Epics
8. Backlog

Quick example

- **Project Overview**
  - **Project Name**: Online Bookstore Development
  - **Goal**: Develop an e-commerce platform for selling books online.
  - **Team**: 5 Developers, 1 SRE (DevOps), 1 Scrum Master, 1 Product Owner

Quick example

- **Sprint 1: Setting Up Basic Features**

- **Duration**: 2 weeks

- **Sprint Planning**:
  - **User Stories**:
    - As a user, I want to create an account so that I can make purchases.
    - As a user, I want to browse books by category so that I can find books easily.
    - As an admin, I want to add new books to the inventory so that users can buy them.

Quick example

- **Tasks**:
  - Implement user registration and login functionality.
  - Develop a basic UI for browsing books by category.
  - Create admin functionality for adding books to the inventory.

- **Daily Stand-Ups**:
  - Team members discuss progress, any blockers, and plan for the day.

Quick example

- **Sprint Review**:
  - Demonstrate the following:
    - User registration and login pages.
    - Book browsing interface.
    - Admin panel for adding books.

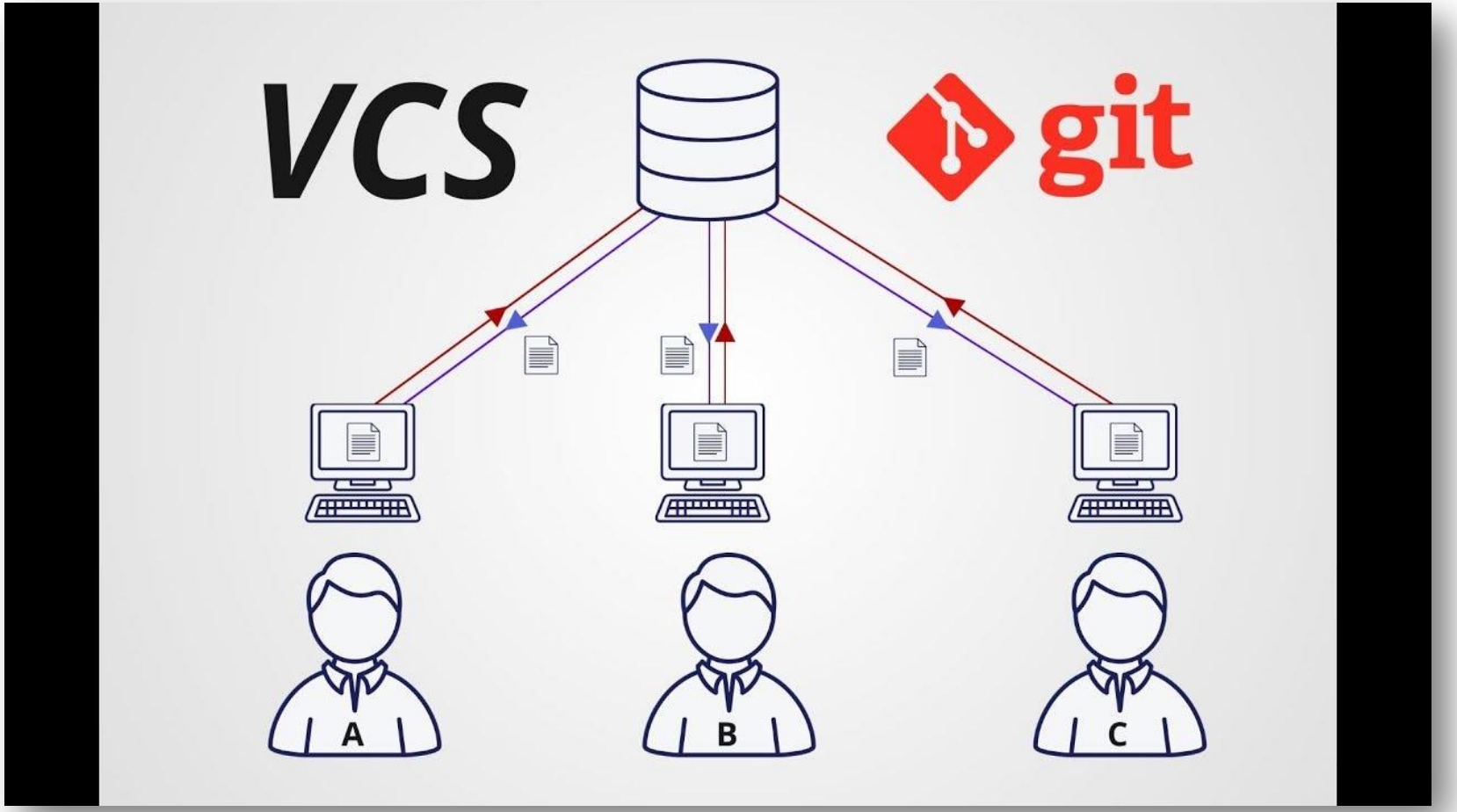
- Collect feedback from the Product Owner and stakeholders.

Quick example

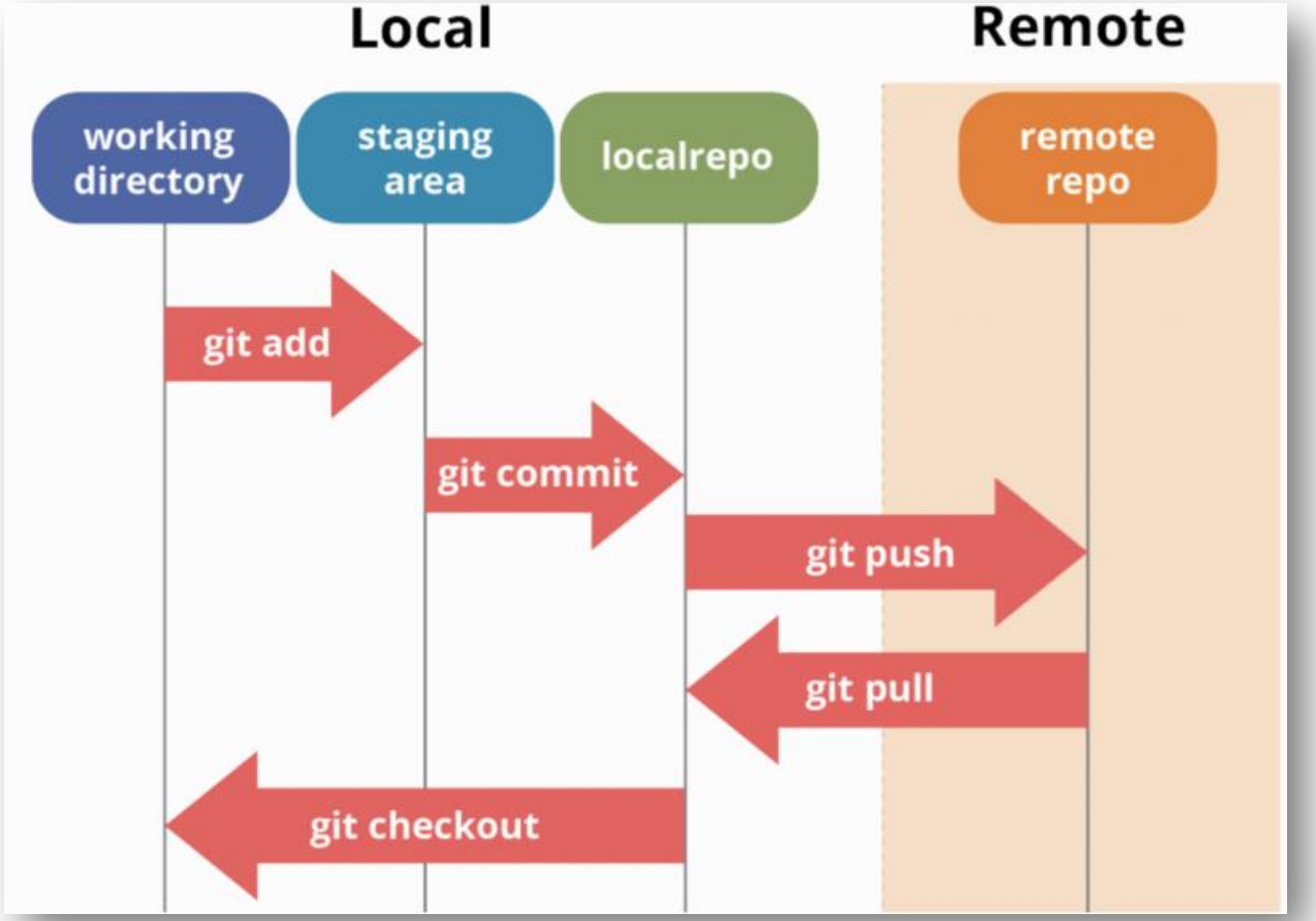
- **Sprint Retrospective**:
  - **What Went Well**:
    - Good collaboration among team members.
    - Completed all planned user stories.
  - **What Could Be Improved**:
    - Improve communication with stakeholders to gather more detailed requirements.
    - Allocate more time for testing.

Git

What is Git?



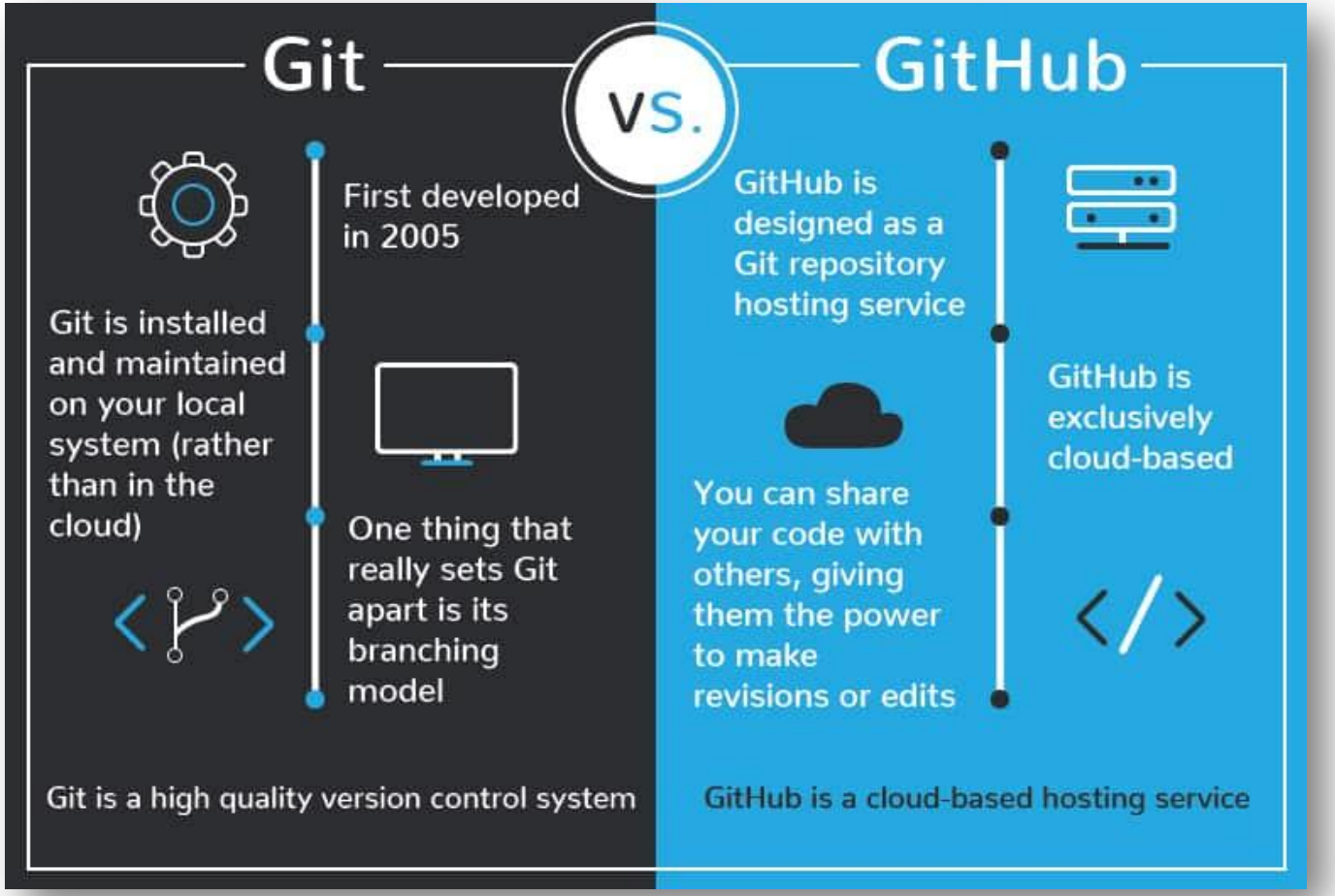
Git Workflow



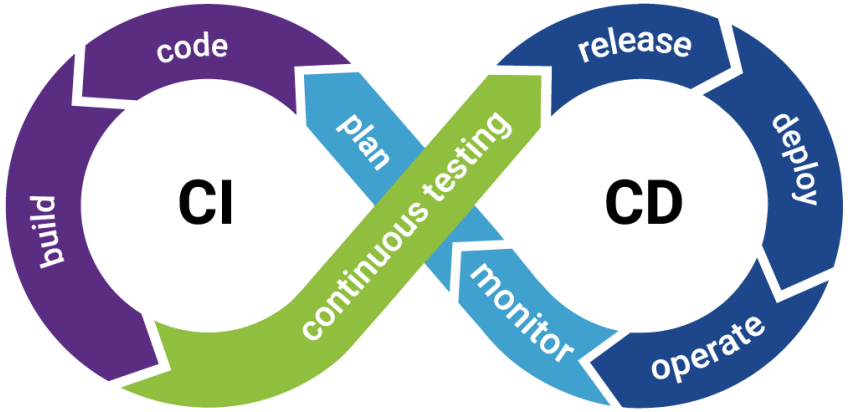
Basic Commands

- Clone: git clone [repository_url]
- Commit: git commit -m "commit message“
- Push: git push [remote] [branch]
- Pull: git pull [remote] [branch]
- Branch: git branch [branch_name]
- Merge: git merge [branch_name]
- Status: git status
- Log: git log

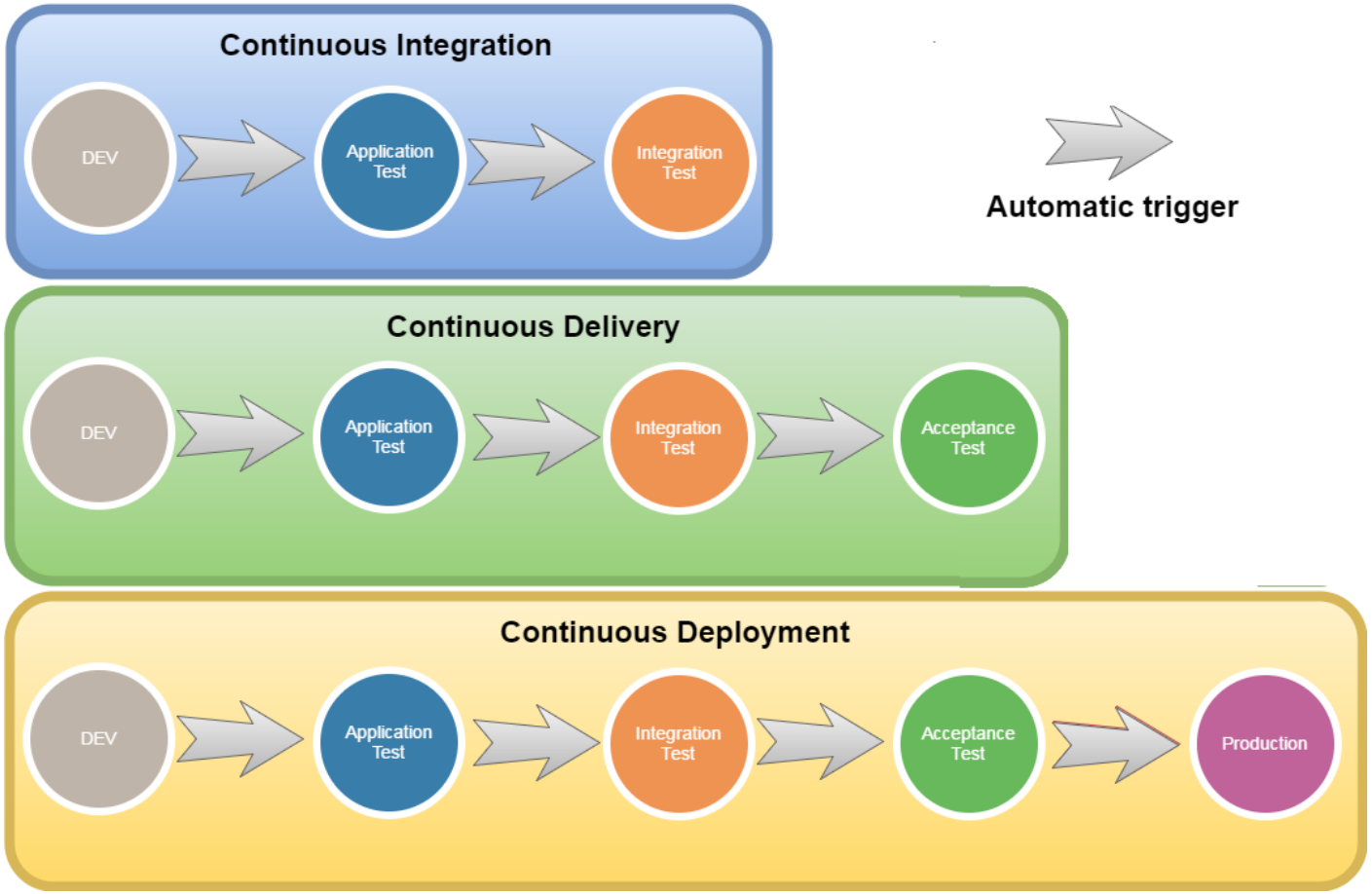
Git vs GitHub



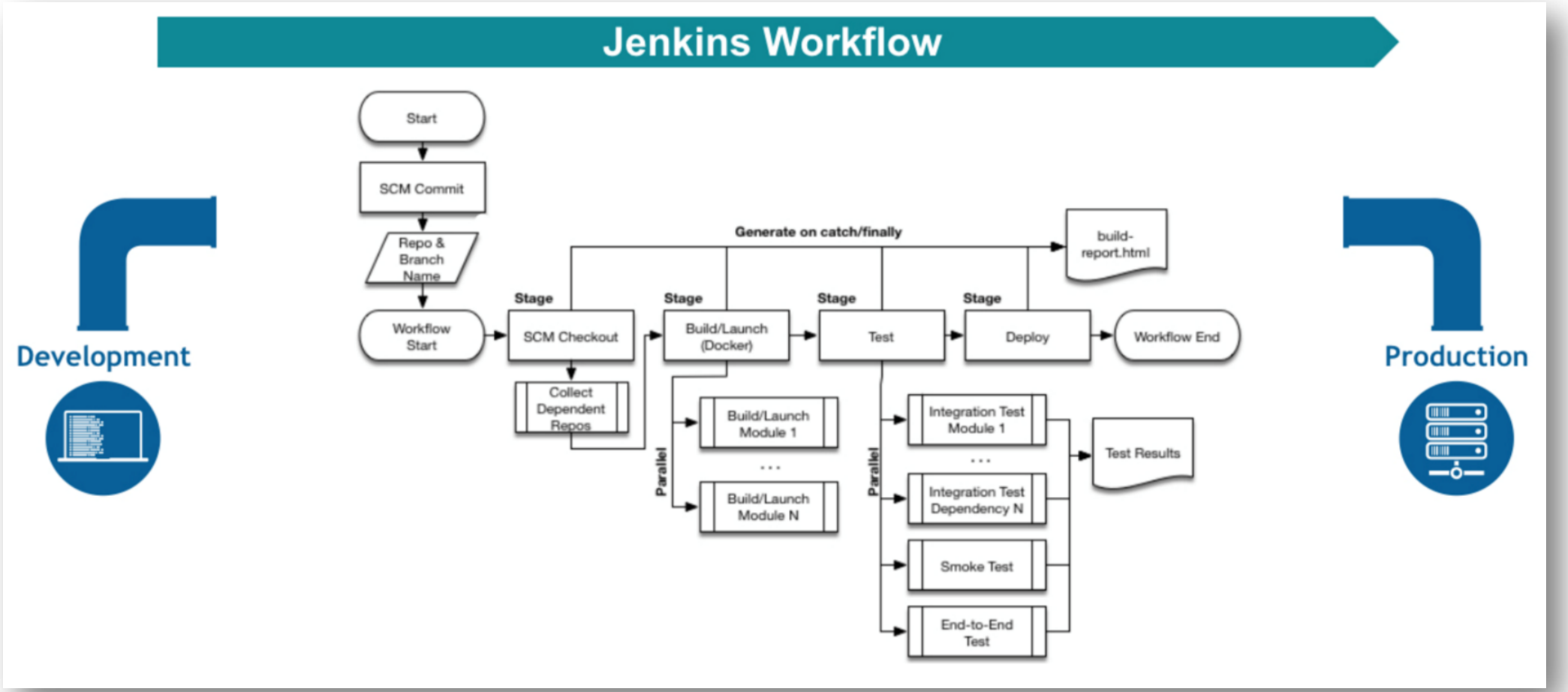
Ci/Cd


code
plan
build
CI
continuous testing
release
deploy
CD
monitor
operate

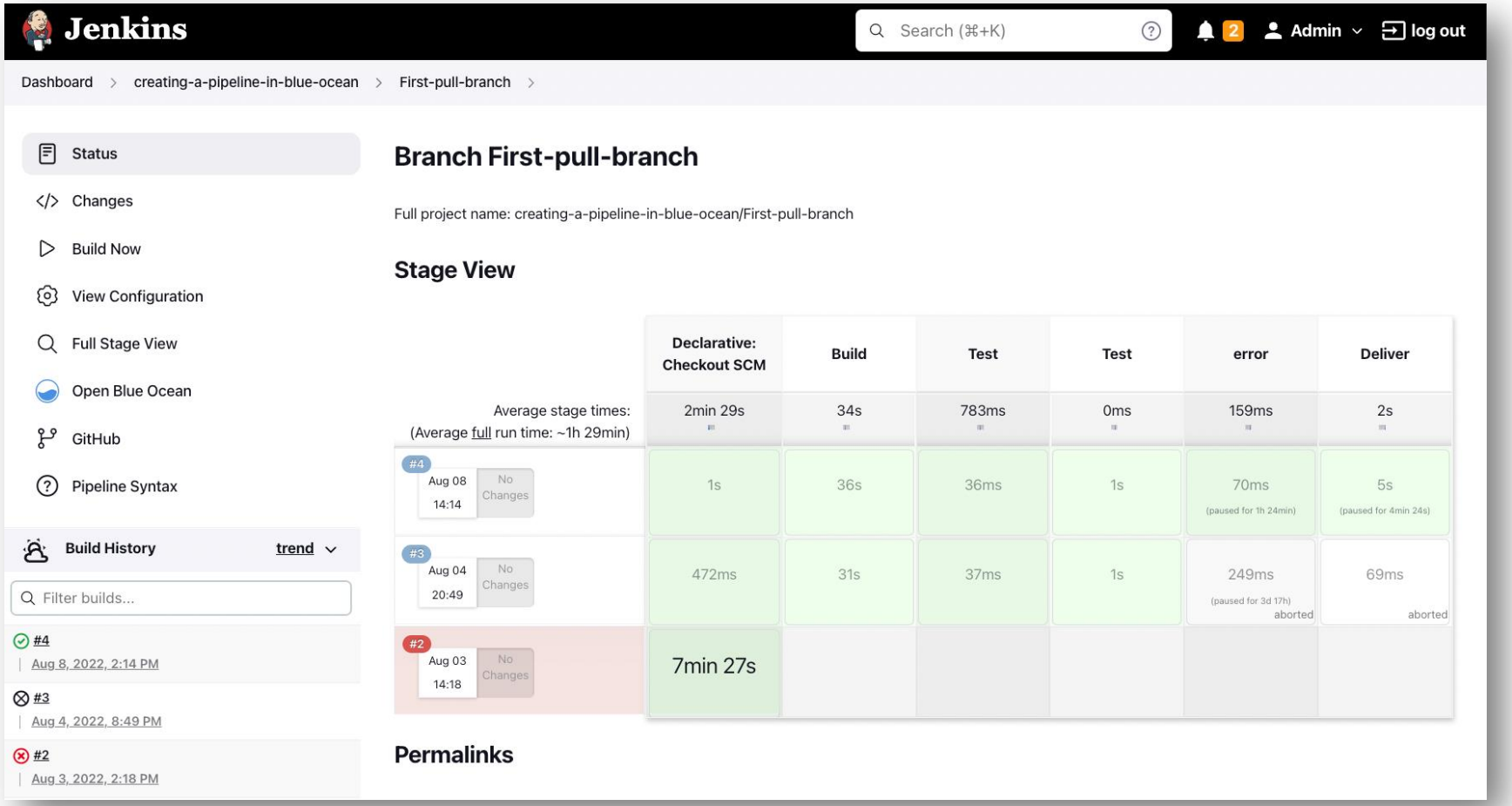
CI vs CD


Continuous Integration
DEV
Application Test
Integration Test
Automatic trigger
Continuous Delivery
DEV
Application Test
Integration Test
Acceptance Test
Continuous Deployment
DEV
Application Test
Integration Test
Acceptance Test
Production

Jenkins Pipeline



Jenkins UI


Jenkins
Search (⌘+K)
2
Admin
log out
Dashboard
creating-a-pipeline-in-blue-ocean
First-pull-branch
Status
Changes
Build Now
View Configuration
Full Stage View
Open Blue Ocean
GitHub
Pipeline Syntax
Build History
trend
Filter builds...
#4
Aug 8, 2022, 2:14 PM
#3
Aug 4, 2022, 8:49 PM
#2
Aug 3, 2022, 2:18 PM
Branch First-pull-branch
Full project name: creating-a-pipeline-in-blue-ocean/First-pull-branch
Stage View
Declarative: Checkout SCM
Build
Test
Test
error
Deliver
Average stage times:
(Average full run time: ~1h 29min)
2min 29s
34s
783ms
0ms
159ms
2s
#4
Aug 08 14:14
No Changes
1s
36s
36ms
1s
70ms
(paused for 1h 24min)
5s
(paused for 4min 24s)
#3
Aug 04 20:49
No Changes
472ms
31s
37ms
1s
249ms
(paused for 3d 17h)
aborted
69ms
aborted
#2
Aug 03 14:18
No Changes
7min 27s
Permalinks

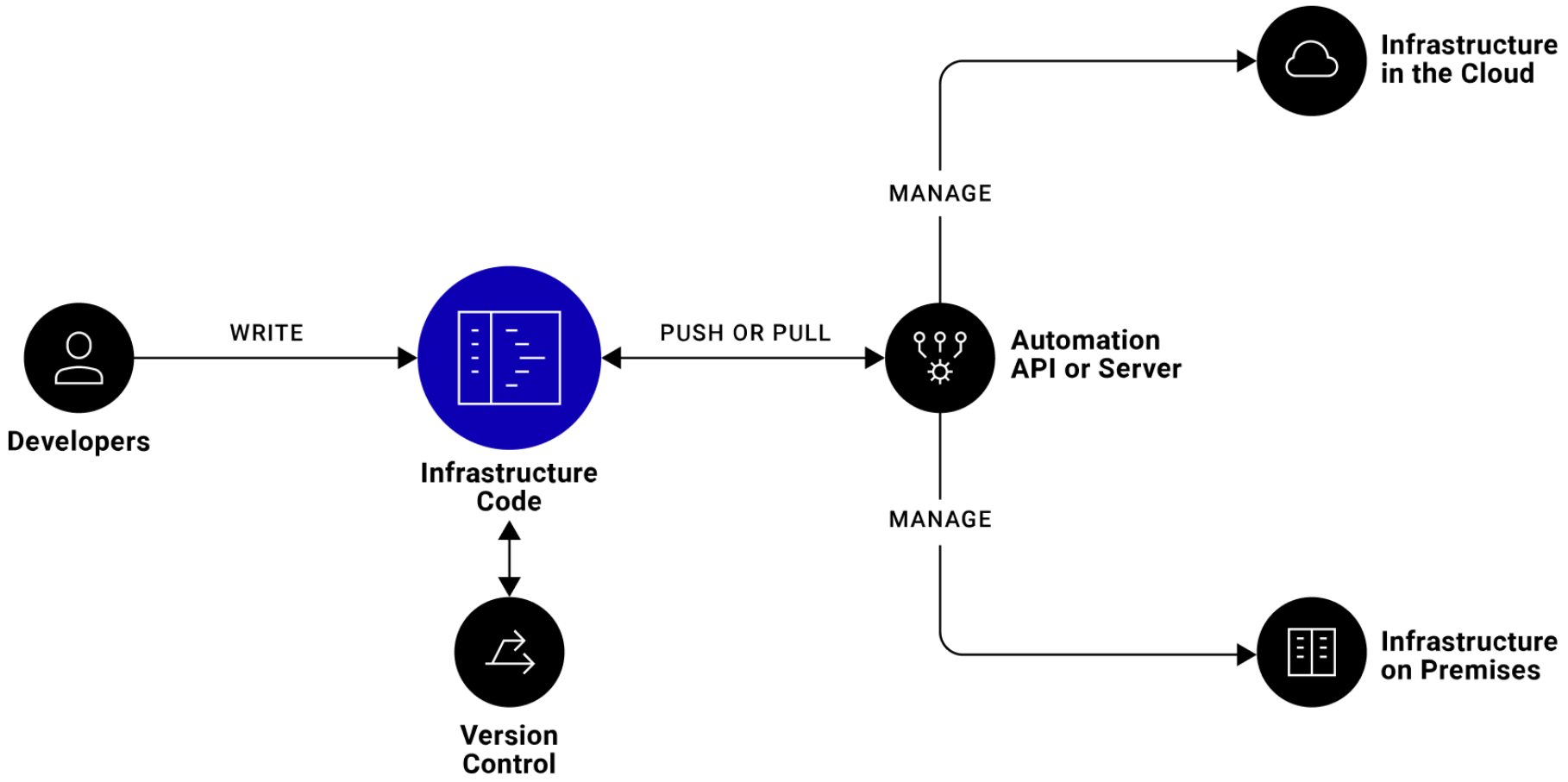
Jenkinsfile

Jenkinsfile (Declarative Pipeline)

```
pipeline { (1)
    agent any (2)
    options {
        skipStagesAfterUnstable()
    }
    stages {
        stage('Build') { (3)
            steps { (4)
                sh 'make' (5)
            }
        }
        stage('Test'){
            steps {
                sh 'make check'
                junit 'reports/**/*.xml' (6)
            }
        }
        stage('Deploy') {
            steps {
                sh 'make publish'
            }
        }
    }
}
```

Infrastructure As Code (IaC)

IaS Workflow

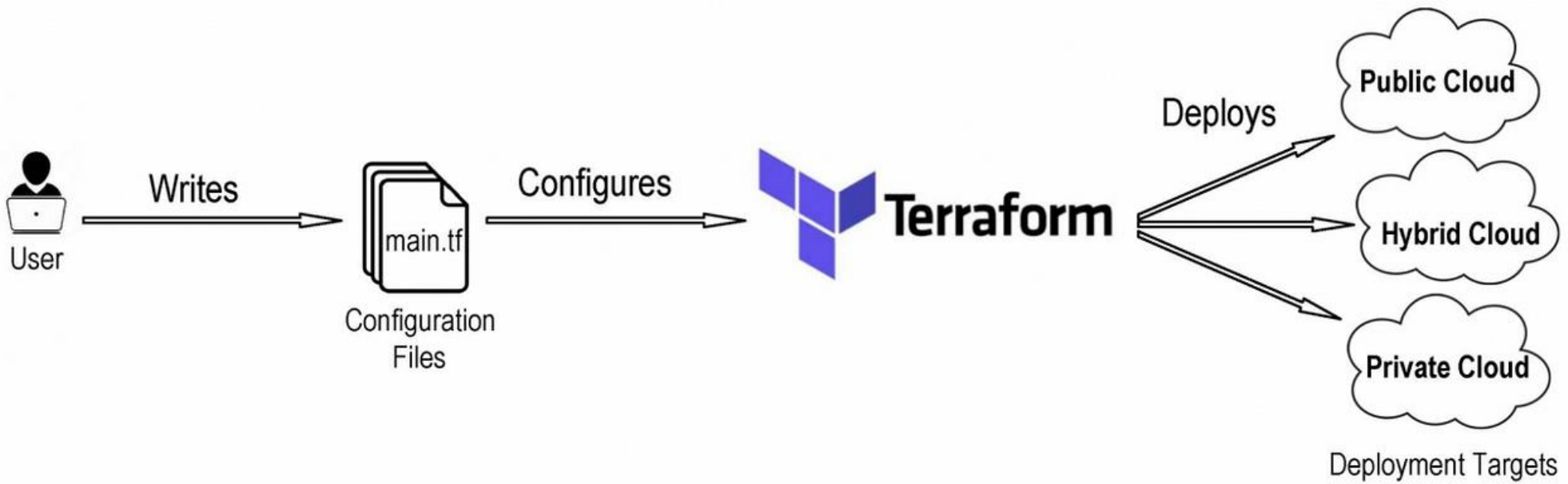


Terraform Code

```
# Configure the Google Cloud provider
provider "google" {
  project = "your-gcp-project-id"
  region  = "us-central1"
  zone    = "us-central1-a"
}
```


```
resource "google_compute_instance" "vm_instance" {
 name = "terraform-instance"
 machine_type = "f1-micro"
 initial_node_count = “3”
 boot_disk {
    initialize_params {
        image = "debian-cloud/debian-9"
        }
 }
 network_interface {
    network = google_compute_network.vpc_network.name
    access_config {
    }
 }
 }
```

Terraform Workflow



Terraform Commands

- Init
- Plan
- Apply
- Show
- Destroy

Benefits of Terraform

- Consistency
- Version Control
- Scalability
- Automation
- Collaboration

Configuration Management

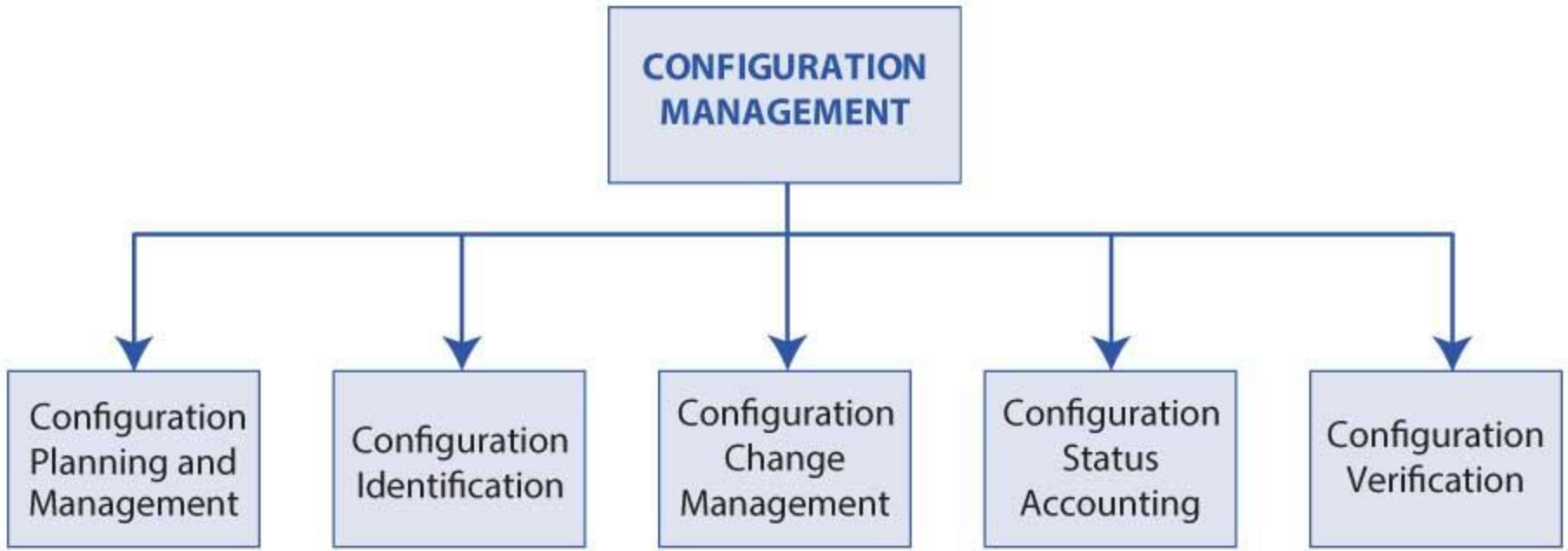
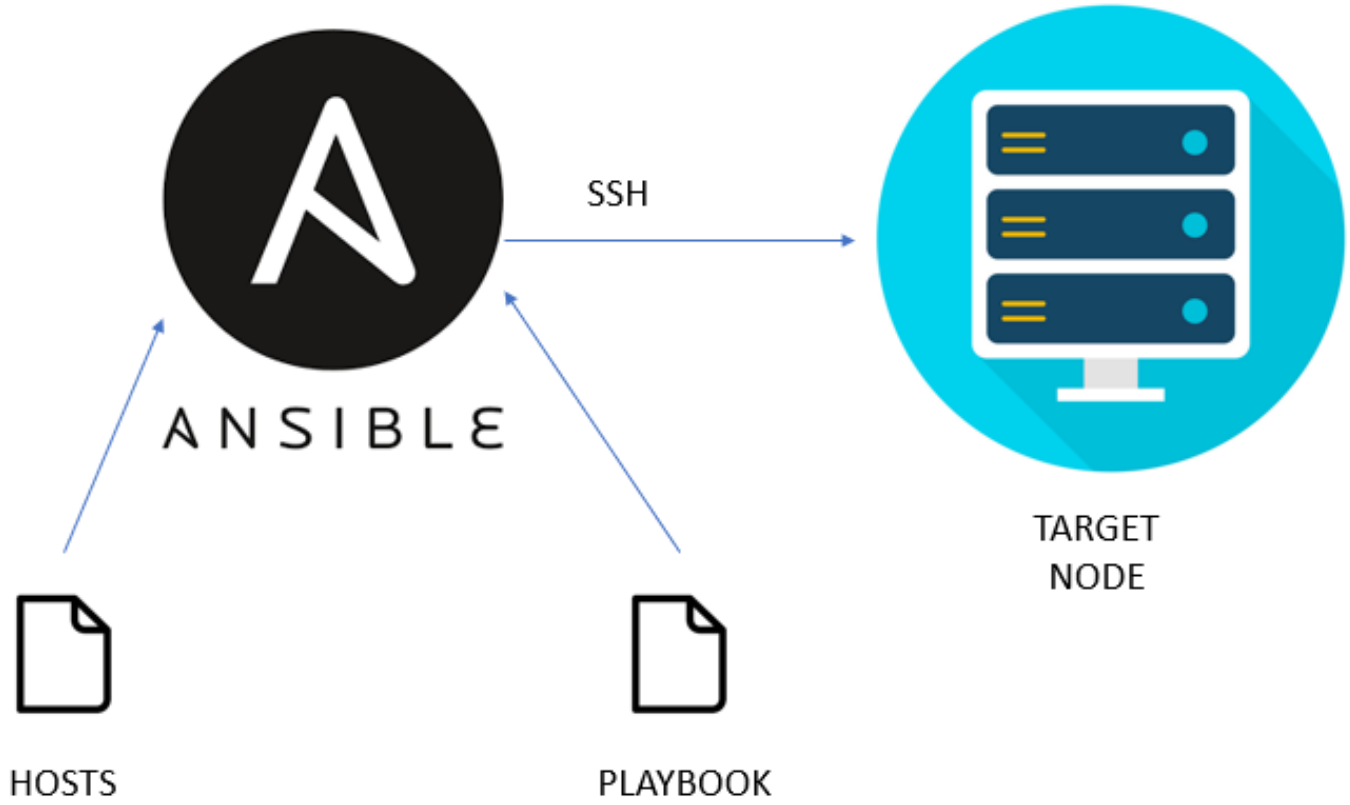


FIGURE 6.5-2 Five Elements of Configuration Management

Ansible



Ansible Hosts File

```
[webservers]
    webserver1.example.com ansible_host=192.168.1.1 ansible_user=root ansible_ssh_private_key_file=~/.ssh/id_rsa
    webserver2.example.com ansible_host=192.168.1.2 ansible_user=root ansible_ssh_private_key_file=~/.ssh/id_rsa
    webserver3.example.com ansible_host=192.168.1.3 ansible_user=root ansible_ssh_private_key_file=~/.ssh/id_rsa

[webservers:vars]
    http_port=80
    max_clients=200
```

Ansible playbook – webserver_setup.yml

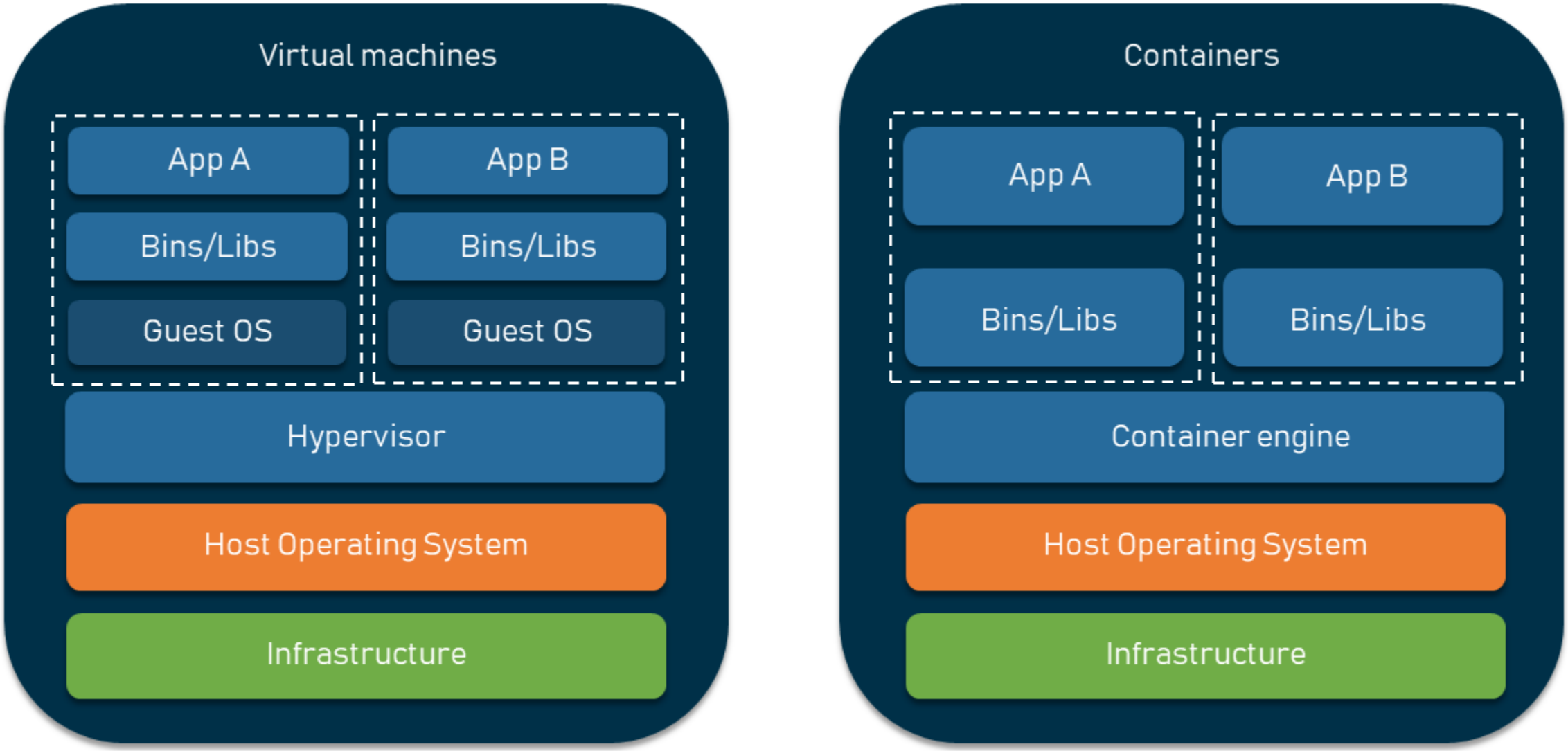
```yaml
---
- name: Setup web servers
  hosts: webservers
  become: yes

  tasks:
    - name: Ensure Apache is installed
      apt:
        name: apache2
        state: present
        update_cache: yes

    - name: Ensure Apache is started and enabled
      service:
        name: apache2
        state: started
        enabled: yes

    - name: Configure Apache
      template:
        src: templates/apache.conf.j2
        dest: /etc/apache2/apache2.conf
```

Command to execute the Playbook

```
$ ansible-playbook -i hosts webserver_setup.yml
```

Containerization

VIRTUAL MACHINES VS CONTAINERS

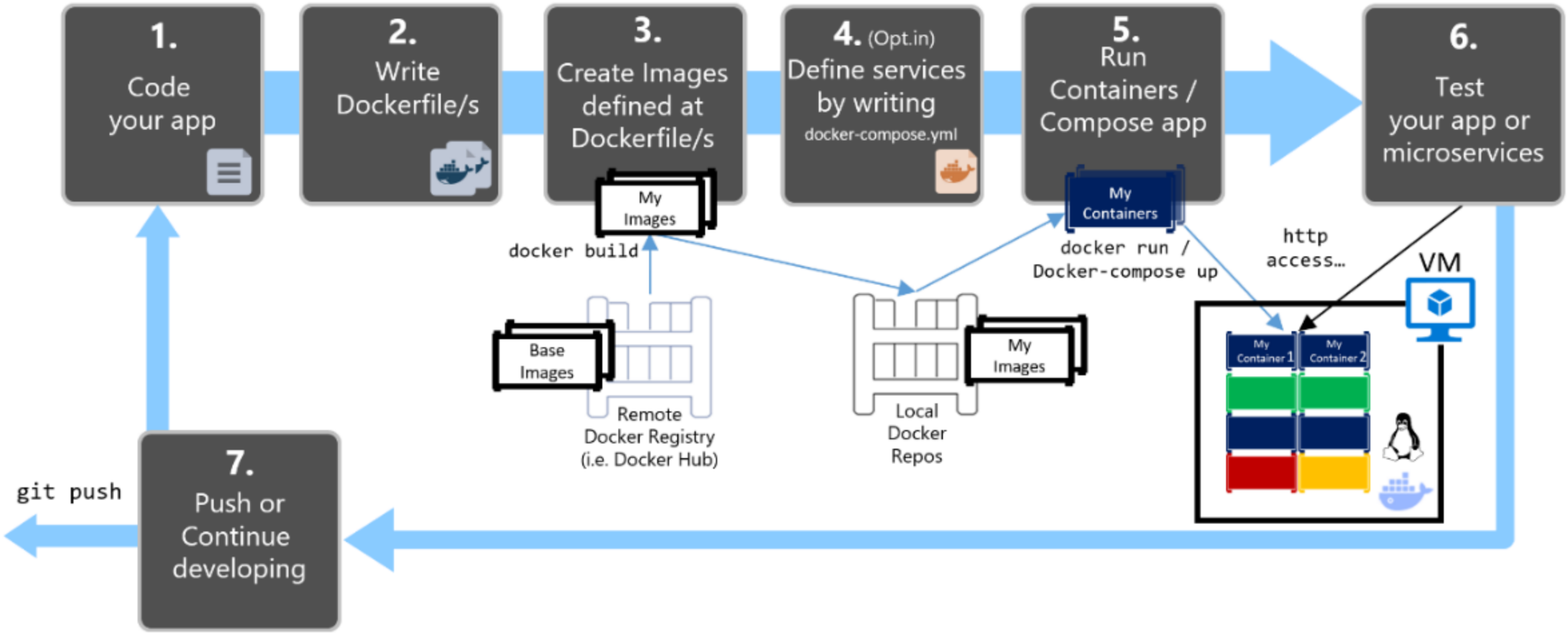


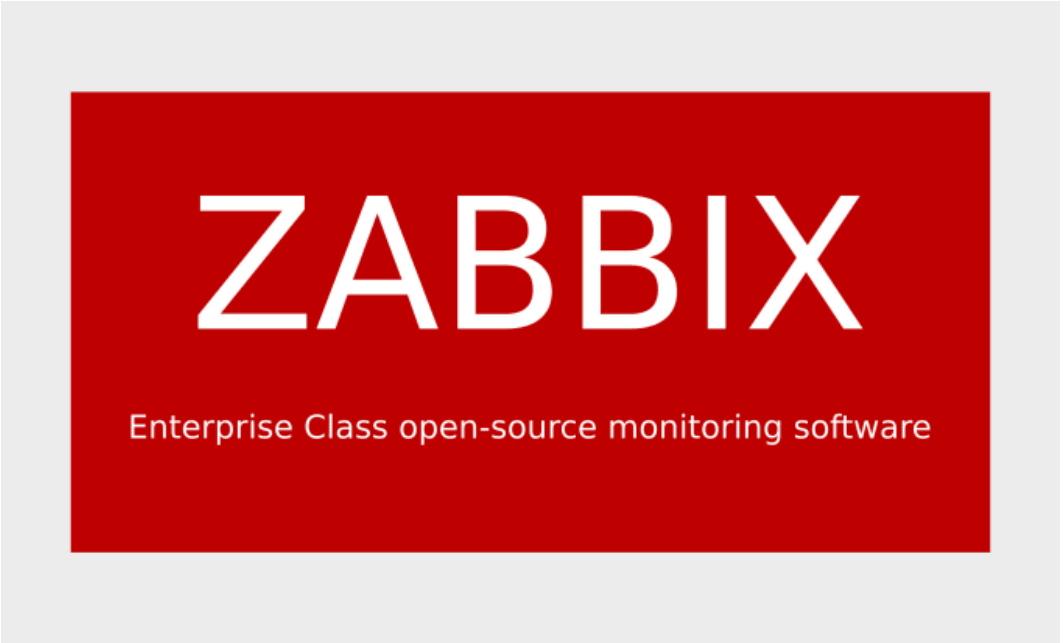
altexsoft

backend > Dockerfile > ...

```dockerfile
FROM node:14.16.0-alpine3.13

RUN addgroup app && adduser -S -G app app
RUN ["chmod", "+x", "/usr/local/bin/docker-entrypoint.sh"]
USER app

WORKDIR /app

COPY package*.json ./
RUN npm install
COPY . .

USER root
RUN ["chmod", "+x", "./docker-entrypoint.sh"]
RUN ["chmod", "+x", "./wait-for"]
USER app

EXPOSE 3001

CMD ["npm", "start"]
```

Inner-Loop development workflow for Docker apps





Monitoring

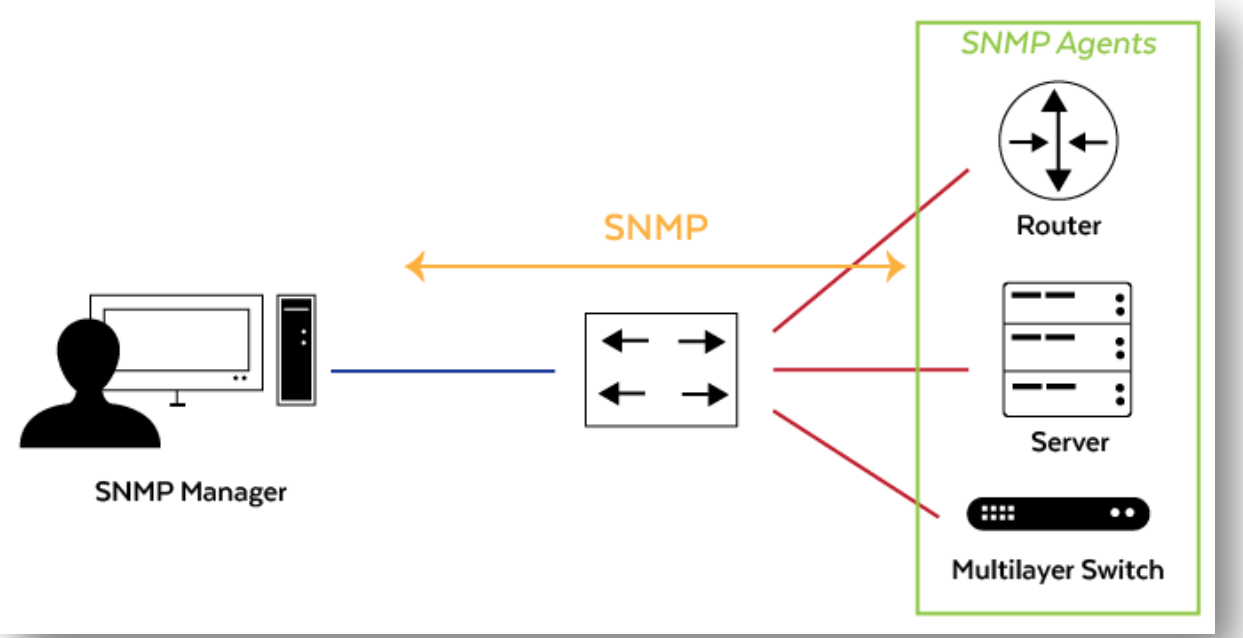
What is Zabbix?

Key Features

- Data Collection
- Alerting & Notification
- Visualization

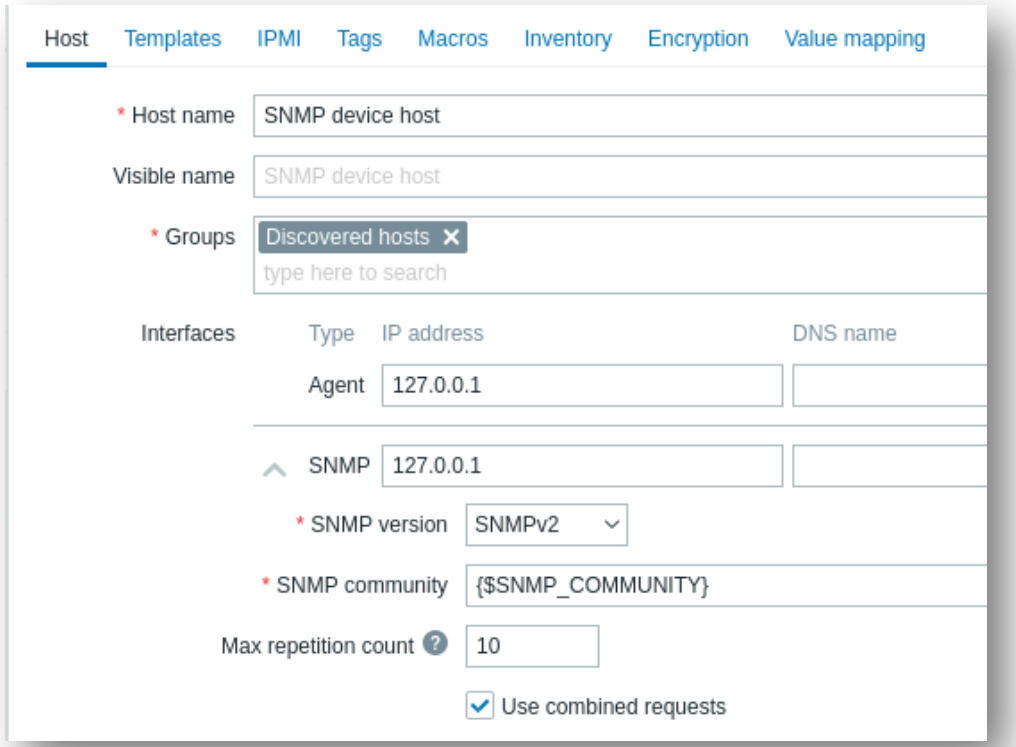
- Auto-Discovery
- Templates
- APIs

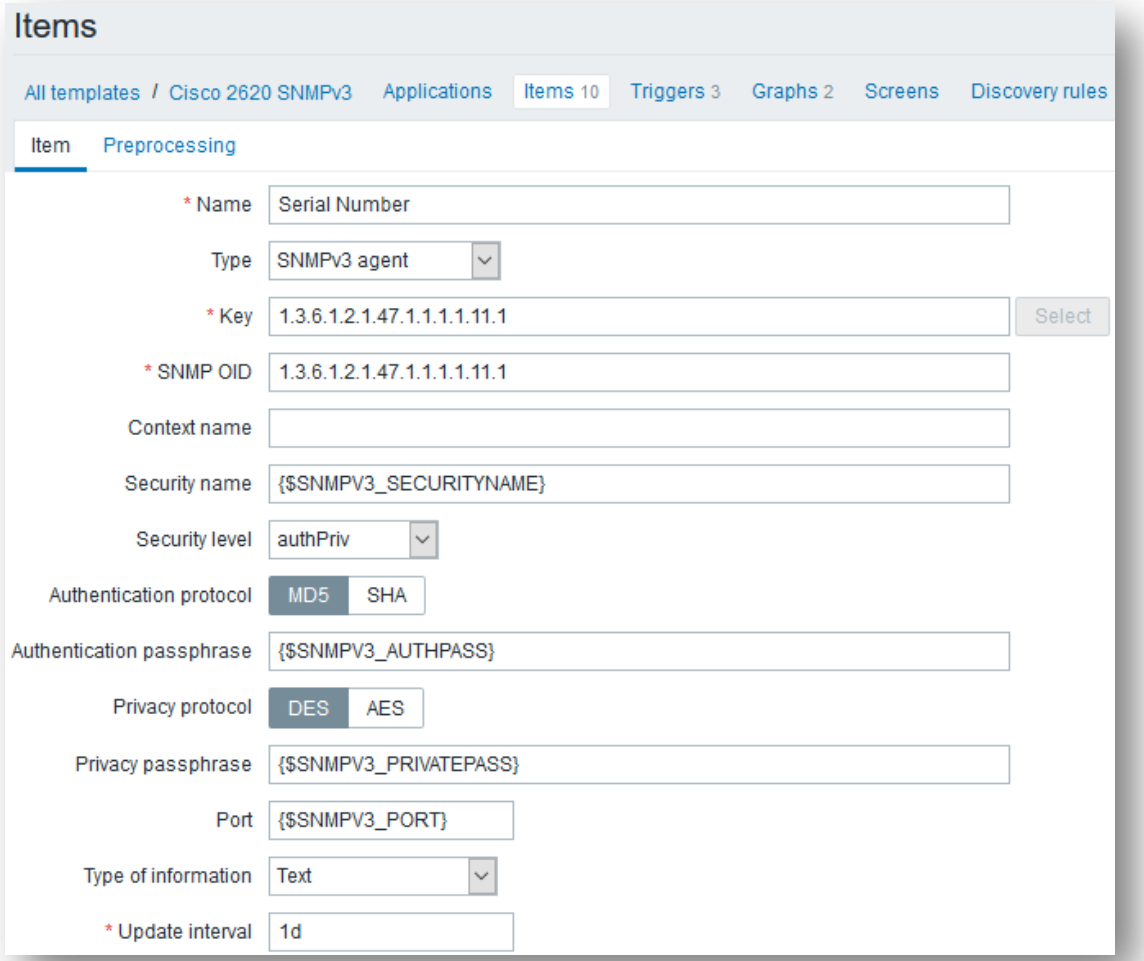
SNMP



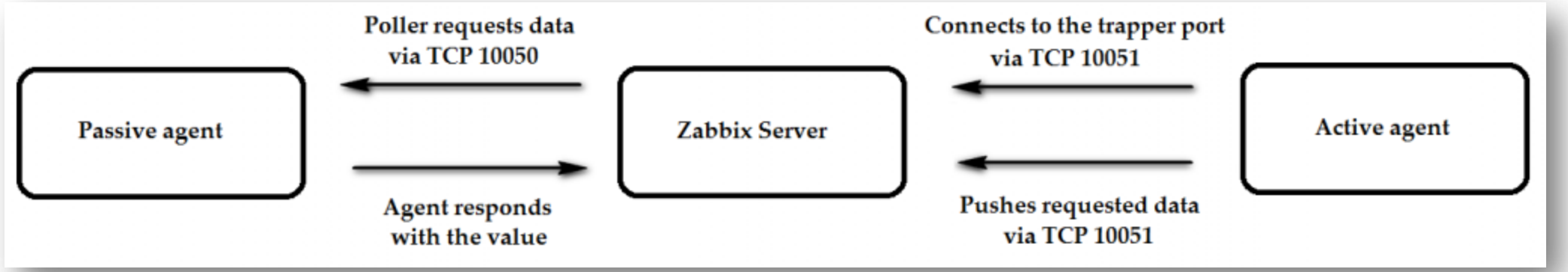
SNMP OID	Description
fgIntfBcQPackets 1.3.6.1.4.1.12356.101.7.5.4.1.1	Packets conform by shaping in the interface, policy, and class.
fgIntfBcQBytes 1.3.6.1.4.1.12356.101.7.5.4.1.2	Bytes conform by shaping in the interface, policy, and class.
fgIntfBcQPDrops 1.3.6.1.4.1.12356.101.7.5.4.1.3	Packets discard by shaping in the interface, policy, and class.
fgIntfBcQBDrops 1.3.6.1.4.1.12356.101.7.5.4.1.4	Bytes discard by shaping in the interface, policy, and class.

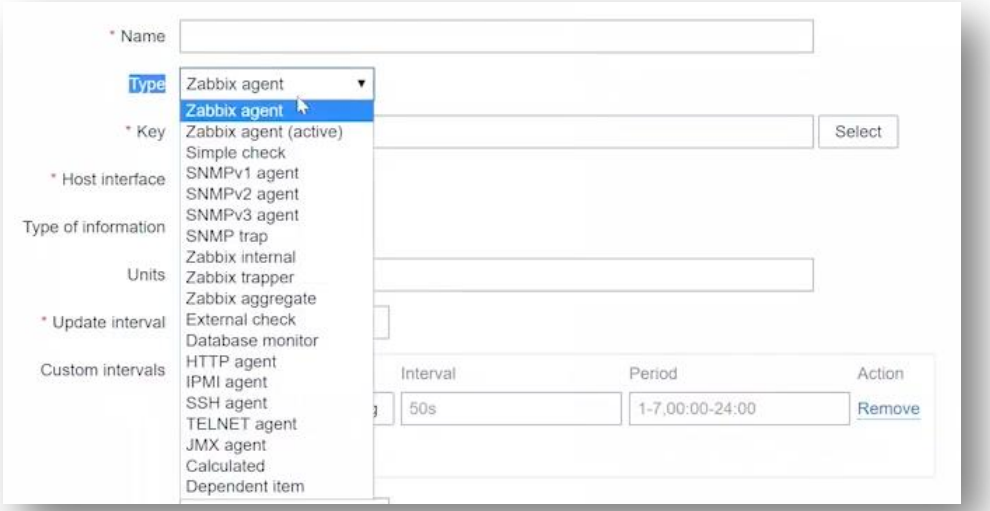
SNMP


Host
Templates
IPMI
Tags
Macros
Inventory
Encryption
Value mapping
* Host name
SNMP device host
Visible name
SNMP device host
* Groups
Discovered hosts
type here to search
Interfaces
Type
IP address
DNS name
Agent
127.0.0.1
SNMP
127.0.0.1
* SNMP version
SNMPv2
* SNMP community
{$SNMP_COMMUNITY}
Max repetition count
10
Use combined requests

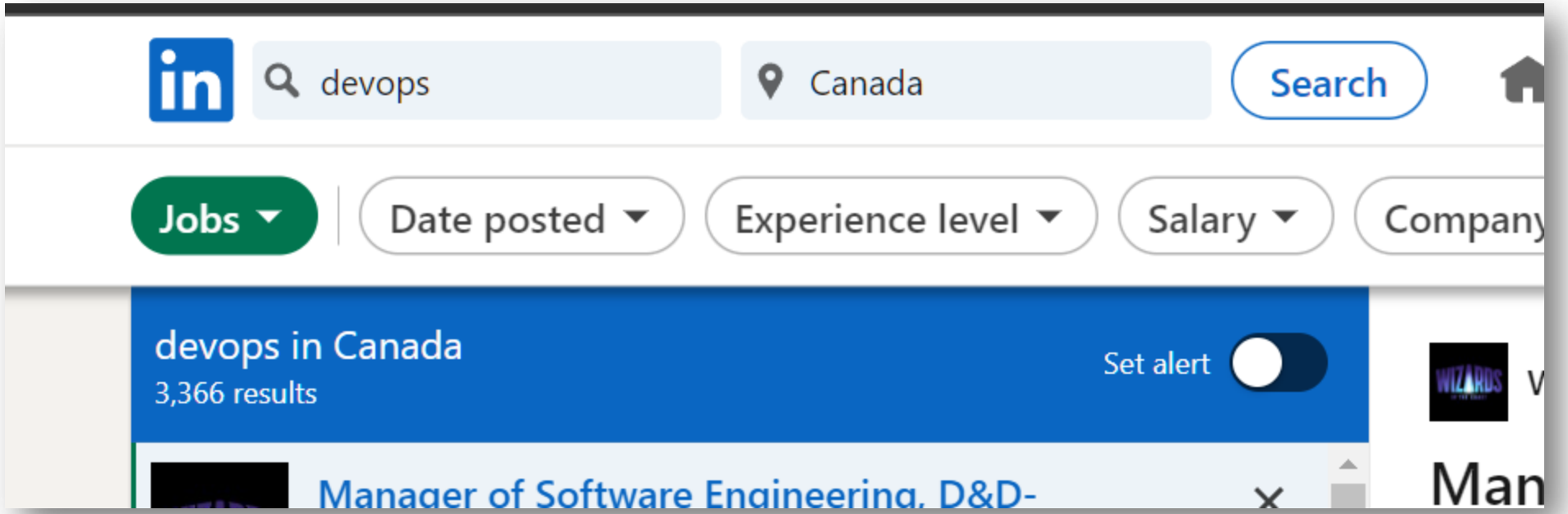

Items
All templates / Cisco 2620 SNMPv3
Applications
Items 10
Triggers 3
Graphs 2
Screens
Discovery rules
Item
Preprocessing
* Name
Serial Number
Type
SNMPv3 agent
* Key
1.3.6.1.2.1.47.1.1.1.1.11.1
Select
* SNMP OID
1.3.6.1.2.1.47.1.1.1.1.11.1
Context name
Security name
{$SNMPV3_SECURITYNAME}
Security level
authPriv
Authentication protocol
MD5
SHA
Authentication passphrase
{$SNMPV3_AUTHPASS}
Privacy protocol
DES
AES
Privacy passphrase
{$SNMPV3_PRIVATEPASS}
Port
{$SNMPV3_PORT}
Type of information
Text
* Update interval
1d

Zabbix Agent

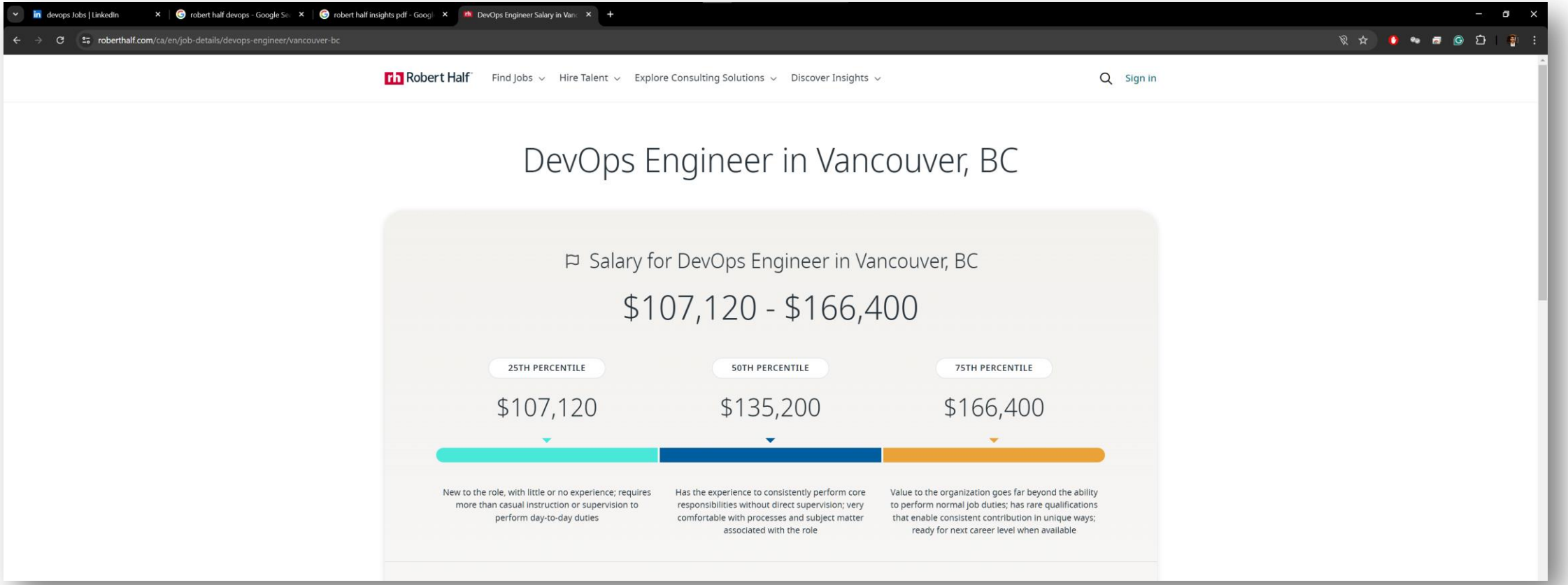




Need in the market



Decent pay


devops Jobs | LinkedIn
robert half devops - Google Se
robert half insights pdf - Googl
DevOps Engineer Salary in Vanc
roberthalf.com/ca/en/job-details/devops-engineer/vancouver-bc
Robert Half
Find Jobs
Hire Talent
Explore Consulting Solutions
Discover Insights
Sign in
DevOps Engineer in Vancouver, BC
Salary for DevOps Engineer in Vancouver, BC
$107,120 - $166,400
25TH PERCENTILE
$107,120
50TH PERCENTILE
$135,200
75TH PERCENTILE
$166,400
New to the role, with little or no experience; requires more than casual instruction or supervision to perform day-to-day duties
Has the experience to consistently perform core responsibilities without direct supervision; very comfortable with processes and subject matter associated with the role
Value to the organization goes far beyond the ability to perform normal job duties; has rare qualifications that enable consistent contribution in unique ways; ready for next career level when available

How to break in?

- Technical Skills

1. https://roadmap.sh/devops
2. Projects / personal portfolio site

- Job search

1. Attend DevOps conferences
2. Practice interviews
3. Reach out on LinkedIn


RECAP
TIME IT IS
makeameme.org


"What you do makes a difference,
and you have to decide
what kind of difference
you want to make."
—Jane Goodall
GoalHabits.com