sourcemind

Docker Compose and Introduction to CI/CD

Mocktar ISSA
Full-Stack Software Engineer

sourcemind

What is docker Compose?

Docker Compose is a tool that allows you to define and manage multi-container Docker applications. It's used to simplify the process of running multiple Docker containers together as a single application, enabling you to define the services, networks, and volumes required for your application in a simple YAML file. Docker Compose then uses this file to spin up and manage the containers based on the configurations you've specified.

sourcemind

What are the advantages of Docker Compose?

1.Simplified Multi-Container Deployment: Docker Compose allows you to define and manage all the containers required for your application in a single file. This makes it much easier to deploy complex applications that rely on multiple services and databases, as you can define all the necessary components in one place.
2.Declarative Syntax: Docker Compose uses a YAML file to define the configuration, making it easy to read and understand. You can specify the services, networks, volumes, and other configurations in a straightforward, human-readable format.
3.Container Orchestration: Docker Compose makes it easy to scale your application up or down by simply changing the number of container replicas in the YAML file. This is particularly useful for load balancing and ensuring high availability of your application.
4.Isolation of Environments: With Docker Compose, you can create separate environments for development, testing, and production using the same configuration file. This ensures consistency across different stages of your application's lifecycle.

sourcemind

What are the advantages of Docker Compose?

5. **Automated Container Linking:** Docker Compose automatically sets up communication between containers in the same network, making it easy for containers to interact with each other without needing to manually manage IP addresses or DNS configurations.

6.Service Discovery: Docker Compose provides service discovery out-of-the-box, meaning that each service within the application can be accessed by its service name within the same network. This makes it easier to connect different services without hardcoding IP addresses.

7.Version Control: As Docker Compose configurations are defined in a YAML file, you can easily version control it along with your source code. This ensures that the entire application stack, including its dependencies, can be versioned and tracked together.

8.Reproducibility: Docker Compose helps in ensuring the consistency and reproducibility of your application across different environments. The exact same configuration can be used to run the application on different machines without any manual changes.

In summary, Docker Compose simplifies the management of multi-container Docker applications, making it easier to deploy, scale, and maintain complex architectures. It offers a powerful and user-friendly way to define your application's infrastructure, allowing you to focus more on developing and less on managing the underlying setup.

sourcemind

Let’s create a simple Docker Compose File

In the Class Folder you can find a sample docker-compose.yml file.

To start all the containers

```
docker compose up
```

Press ctrl + c , ctrl + d To exit.

To detach the execution run:

```
docker compose up -d
```

When you make changes to the services you need to rebuild them:

```
docker compose up -d -build
```

To shut down the containers run:

docker compose down

sourcemind

Useful commands

To show all the services running in our current docker compose context run :
docker compose ps
To list all the running compose projects:
docker compose ls

We can also specify a compose file by using the -f flag:

docker compose -f docker-compose.dev.yml up -d -build

We can view the execution logs:
docker compose logs

We can follow the logs by using the -f flags
docker compose logs -f

There are many case specific commands we can use to get the job done and you can find the full list in the docker documentation at: Docker Compose Reference

sourcemind

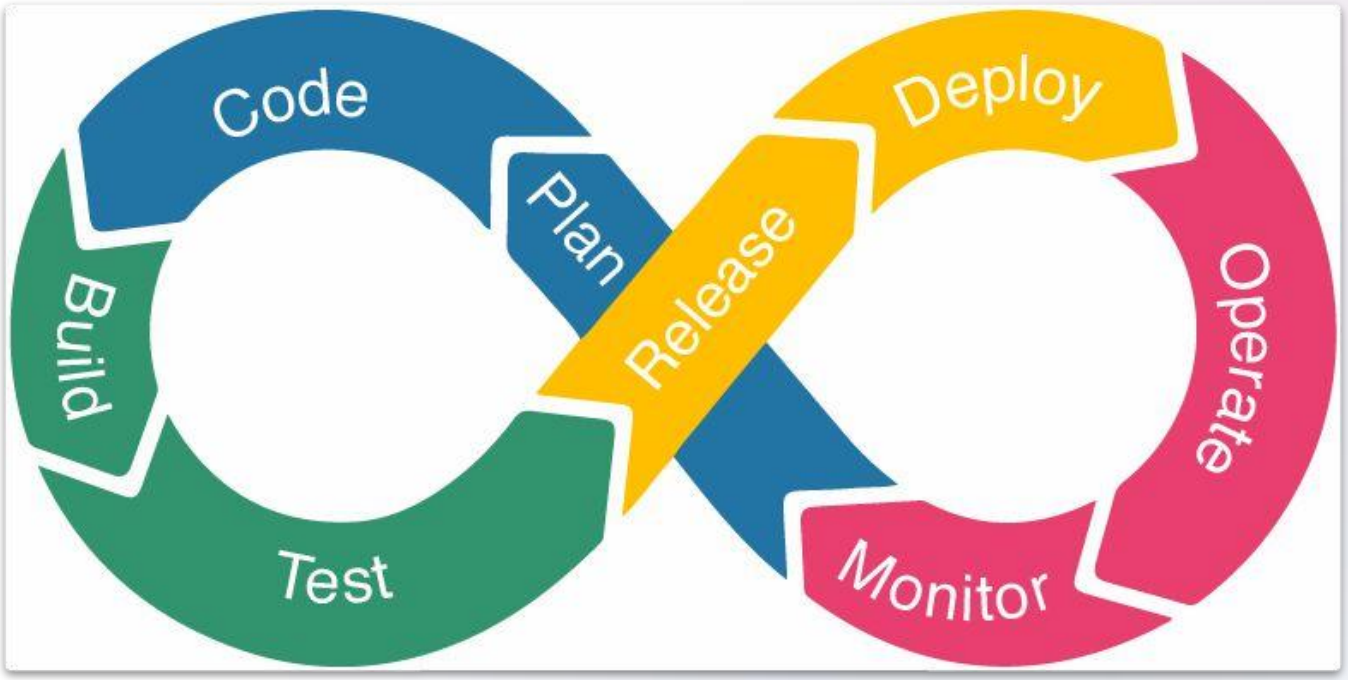
Key take aways

Docker compose enables us to run multi container environments in a simple and efficient manner. It can also be useful when running a configuration where we want to create multiple ressources at once Volume, Network, Services . And manage and link them from one source of truth.

sourcemind

What is DevOps ?

Devops is a software development methodology which improves the collaboration between developers and operations team using various automation tools. These automation tools are implemented using various stages which are a part of the Devops Lifecycle



sourcemind

Benefits of DevOps ?

Speed. DevOps practices let you move at the velocity you need to innovate faster, adapt to changing markets better, and become more efficient at driving business results.

Rapid delivery. When you increase the pace of releases, you can improve your product faster and build competitive advantage.

Reliability. DevOps practices like continuous integration and continuous delivery can ensure the quality of application updates and infrastructure changes so you can reliably deliver at a more rapid pace while maintaining an optimum experience for end users.

Improved collaboration. Under a DevOps model, developers and operations teams collaborate closely, share responsibilities, and combine their workflows. This reduces inefficiencies and saves time.

Security. You can adopt a DevOps model without sacrificing security by using automated, integrated security testing tools.

sourcemind

List of DevOps Practices

Continuous development. This practice spans the planning and coding phases of the DevOps lifecycle. Version-control mechanisms might be involved.

Continuous testing. This practice incorporates automated, prescheduled, continued code tests as application code is being written or updated. Such tests can speed the delivery of code to production.

Continuous integration (CI). This practice brings configuration management (CM) tools together with other test and development tools to track how much of the code being developed is ready for production. It involves rapid feedback between testing and development to quickly identify and resolve code issues.

Continuous delivery. This practice automates the delivery of code changes, after testing, to a pre production or staging environment. An staff member might then decide to promote such code changes into production.

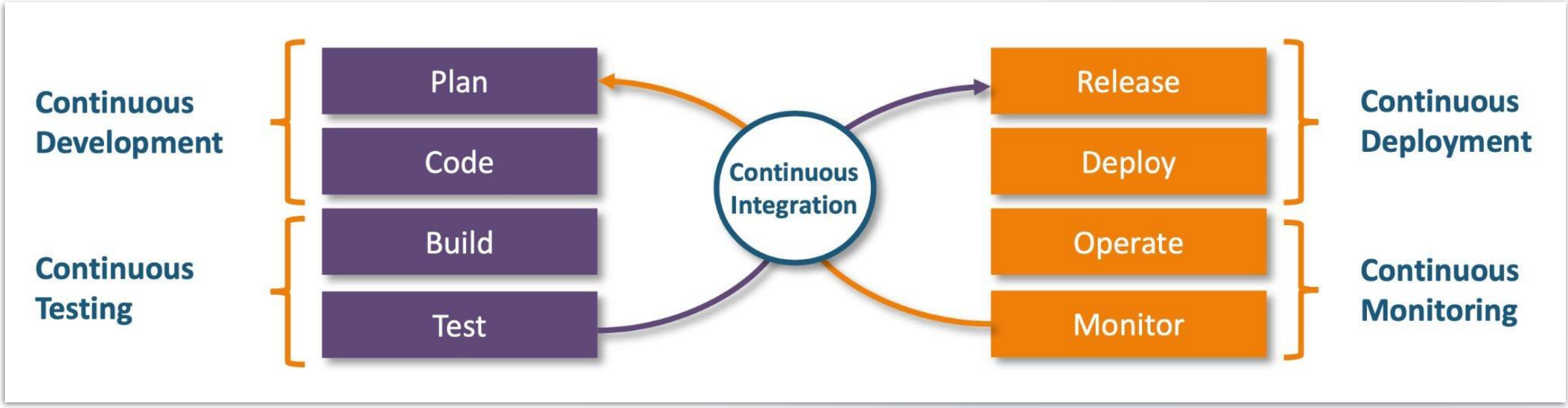
Continuous deployment (CD). Similar to continuous delivery, this practice automates the release of new or changed code into production. A company doing continuous deployment might release code or feature changes several times per day.

Continuous monitoring. This practice involves ongoing monitoring of both the code in operation and the underlying infrastructure that supports it. A feedback loop that reports on bugs or issues then makes its way back to development.

Infrastructure as code. This practice can be used during various DevOps phases to automate the provisioning of infrastructure required for a software release. Developers add infrastructure “code” from within their existing development tools.

sourcemind

How DevOps works ?



sourcemind

How DevOps works ?

- **Continuous Development**
- **Continuous Integration**
- **Continuous Deployment**
- **Continuous Testing**
- **Continuous Monitoring**

This stage involves committing code to version control tools such as Git or SVN for maintaining the different versions of the code, and tools like Ant, Maven, Gradle for building/ packaging the code into an executable file that can be forwarded to the QAs for testing.

sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

The stage is a critical point in the whole Devops Lifecycle. It deals with integrating the different stages of the devops lifecycle, and is therefore the key in automating the whole Devops Process

sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

In this stage the code is built, the environment or the application is containerized and is pushed on to the desired server. The key processes in this stage are Configuration Management, Virtualization and Containerization

sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

The stage deals with automated testing of the application pushed by the developer. If there is an error, the message is sent back to the integration tool, this tool in turn notifies the developer of the error. If the test was a success, the message is sent to Integration tool which pushes the build on the production server

sourcemind

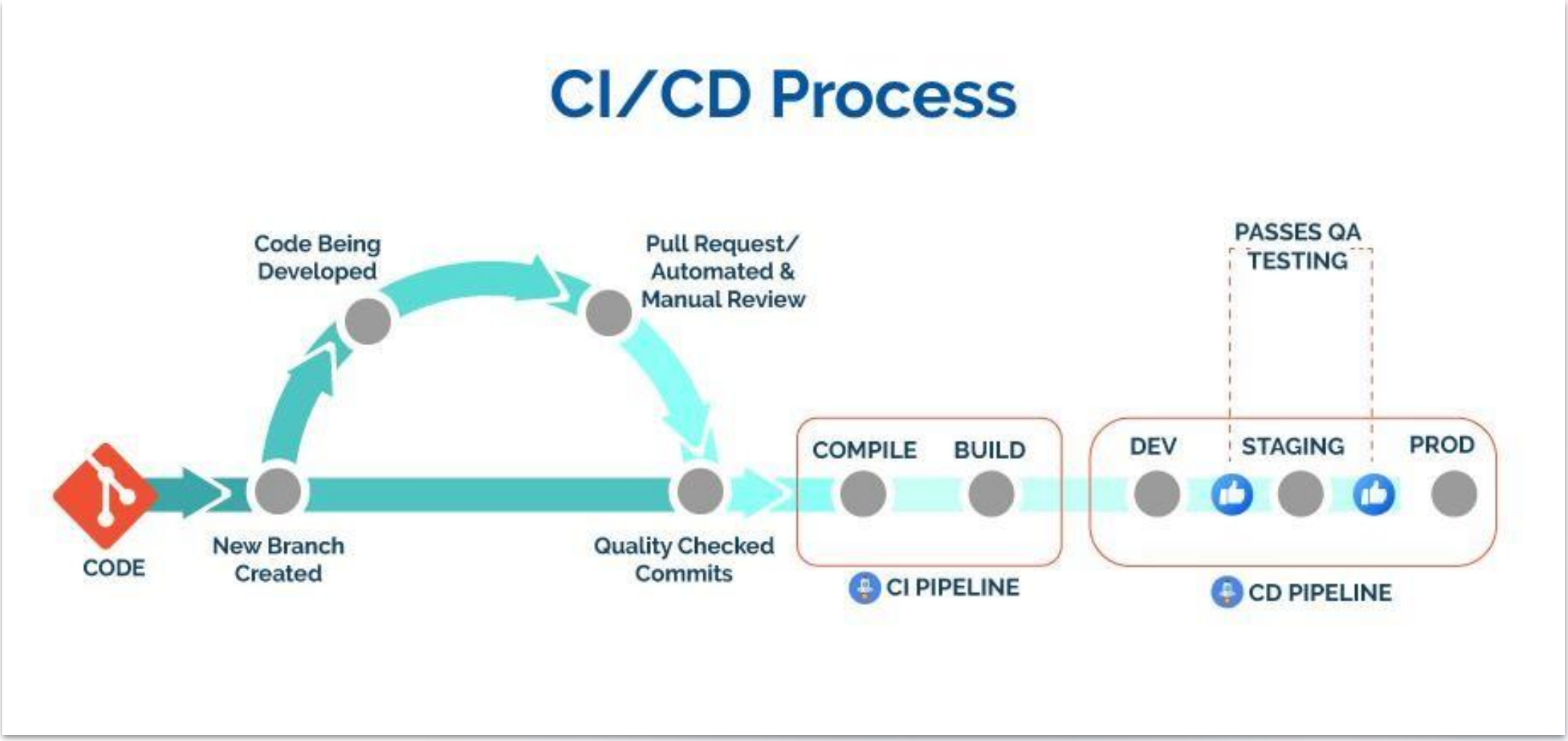
How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

The stage continuously monitors the deployed application for bugs or crashes. It can also be setup to collect user feedback. The collected data is then sent to the developers to improve the application

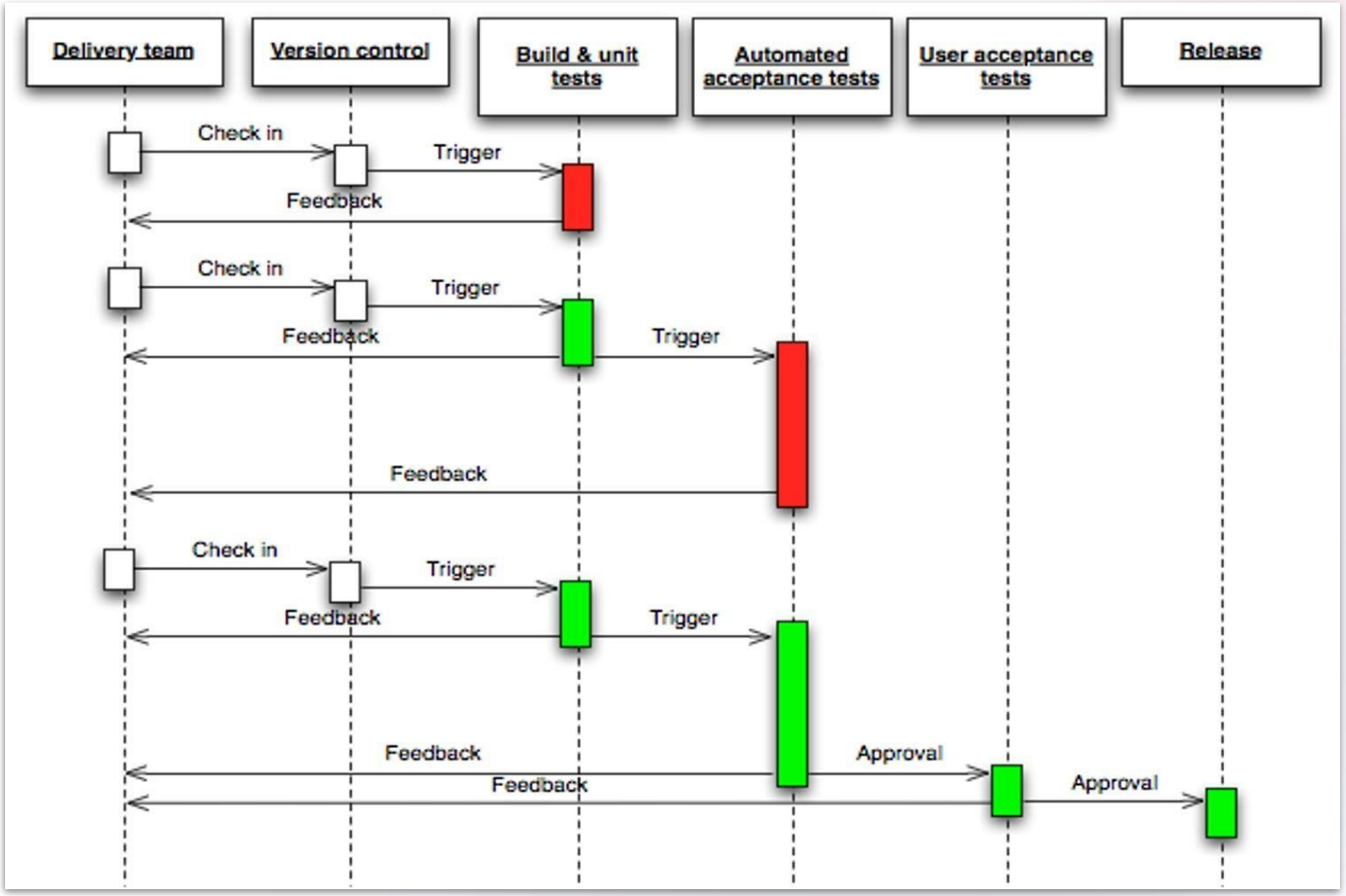
sourcemind

Automated CI/CD Pipeline



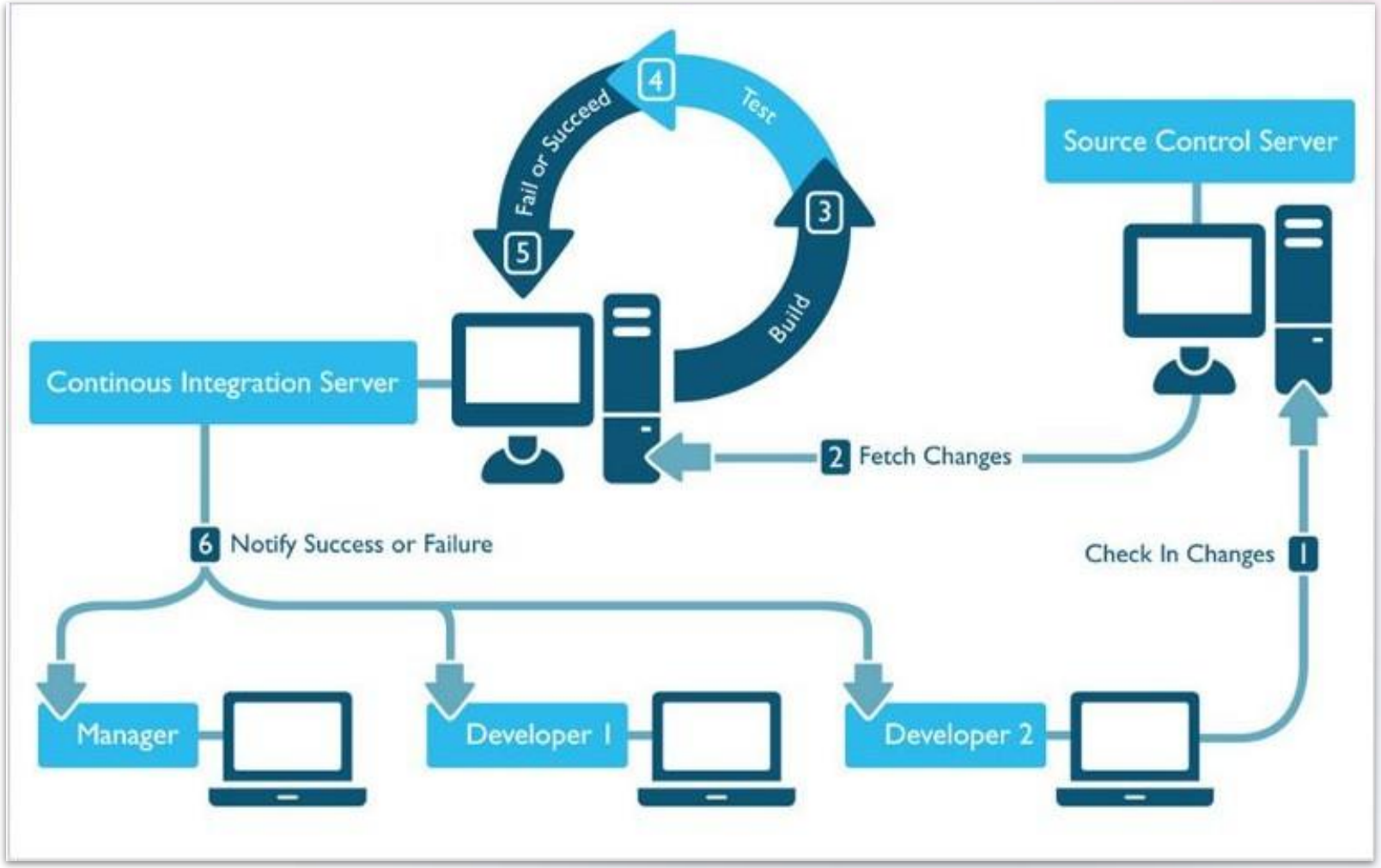
sourcemind

CI Evaluation



sourcemind

Automated CI/CD Pipeline



sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

Git is a distributed version-control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source-code management in software development, but it can be used to keep track of changes in any set of files

sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

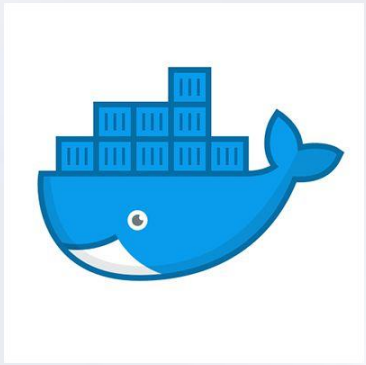
Jenkins is an open source general-purpose CI/CD solution that allows the most flexibility for all sorts of workflows and development practices. It comes bundled with both vendor specific and community maintained plugins.

sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

In this stage the code is built, the environment or the application is containerized and is pushed on to the desired server. The key processes in this stage are Configuration Management, Virtualization and Containerization



sourcemind

How DevOps works ?

- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

Selenium is a portable software-testing framework used for web applications. It is an open source tool which is used for automating the tests carried out on web browsers (Web applications are tested using any web browser).

sourcemind

How DevOps works ?

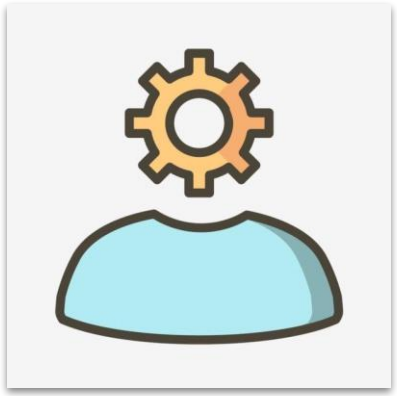
- Continuous Development
- Continuous Integration
- Continuous Deployment
- Continuous Testing
- Continuous Monitoring

Prometheus is a free software application used for event monitoring and alerting. It records real-time metrics in a time series database built using a HTTP pull model, with flexible queries and real-time alerting. Grafana is a multi-platform open source analytics and interactive visualization web application. It provides charts, graphs, and alerts for the web when connected to supported data sources.

sourcemind

Jenkins

Admins



TeamCity

Advantages

- Easy Setup & Administration
- Simple Configuration
- Plugins Integration With tools such as Docker , Bitbucket, Github etc…

sourcemind

Lab lauching a Jenkins Instance

Follow the instructions to install a Jenkins Instance on your Machine

Duration: 1 H

sourcemind

Thank you for your attention !

Q&A