



Topics

- What is Jenkins and CI/CD?
- Lab Setup & Installation
- Jenkins Dashboard Overview
- Some Basic Jenkins Job
- Scheduling The Job
- Working with GIT Repo
- Email Notification
- Job for action on Remote server
- Jenkins + Ansible Playbook
- Project - Website Update on Remote Server
- User Management (Role Based)
- Environment Variables
- Jenkins + MAVEN
- CI/CD Project (Jenkins PIPELINE)
- Bonus Tips



Jenkins

What is Jenkins?





**Jenkins is an open-source
automation server**



**Automate various parts of
software development.**

Building

Testing

Deploy



Continuous Integration and Continuous Delivery

As an extensible automation server, Jenkins can be used as a simple CI server or turned into the continuous delivery hub for any project.



Easy installation

Jenkins is a self-contained Java-based program, ready to run out-of-the-box, with packages for Windows, Linux, macOS and other Unix-like operating systems.



Easy configuration

Jenkins can be easily set up and configured via its web interface, which includes on-the-fly error checks and built-in help.



Plugins

With hundreds of plugins in the Update Center, Jenkins integrates with practically every tool in the continuous integration and continuous delivery toolchain.



Extensible

Jenkins can be extended via its plugin architecture, providing nearly infinite possibilities for what Jenkins can do.



Distributed

Jenkins can easily distribute work across multiple machines, helping drive builds, tests and deployments across multiple platforms faster.



What is CI/CD?

CI/CD is a method of frequently delivering apps to customers by introducing automation into the stages of app development.

CONTINUOUS INTEGRATION



CONTINUOUS DELIVERY



CONTINUOUS DEPLOYMENT



Lab Setup

Installation



```
sudo dnf upgrade
```

```
sudo wget -O /etc/yum.repos.d/jenkins.repo \  
https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

```
sudo rpm --import https://pkg.jenkins.io/redhat-  
stable/jenkins.io-2023.key
```

```
sudo yum install jenkins
```

```
sudo systemctl start/stop jenkins
```

To start the Jenkins from browser: IP:8080

To access it from browser, we need to enable it on firewall

```
sudo firewall-cmd --permanent --zone=public --add-port=8080/tcp  
sudo firewall-cmd --reload
```

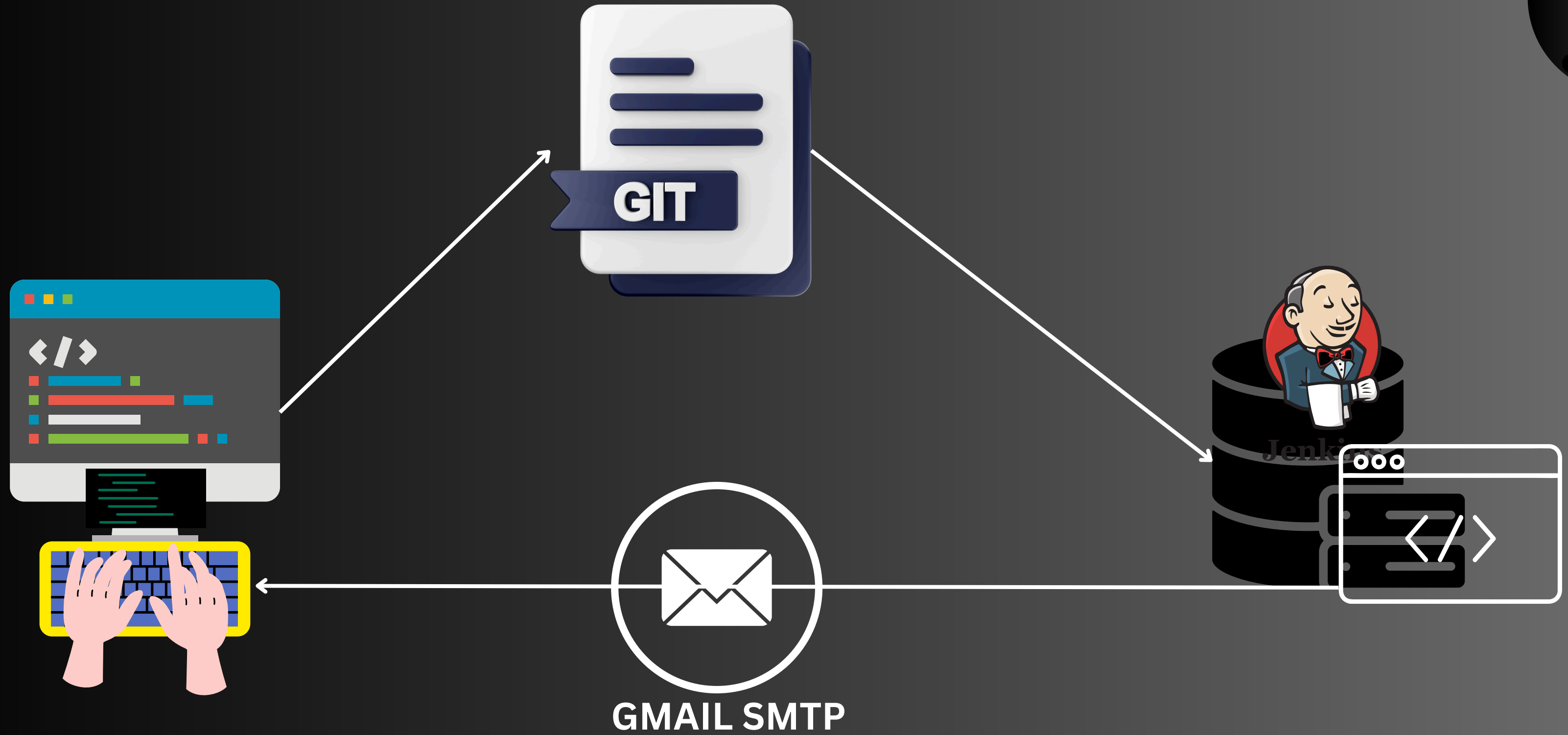


Jenkins

Overview Of DASHBOARD



[Git hub link - https://github.com/paulphilip/pythoncode](https://github.com/paulphilip/pythoncode)



[Git hub link - https://github.com/paulphilip/pythoncode](https://github.com/paulphilip/pythoncode)

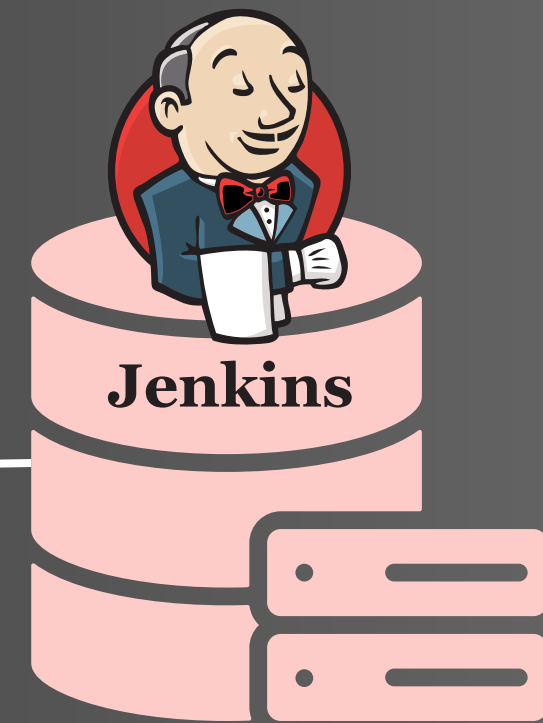


Task like

Transfer file
Execute Commands



SERVER B



SERVER A



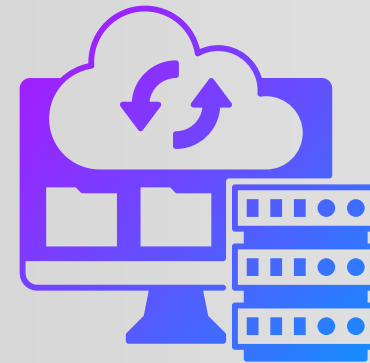
Jenkins



ServerA



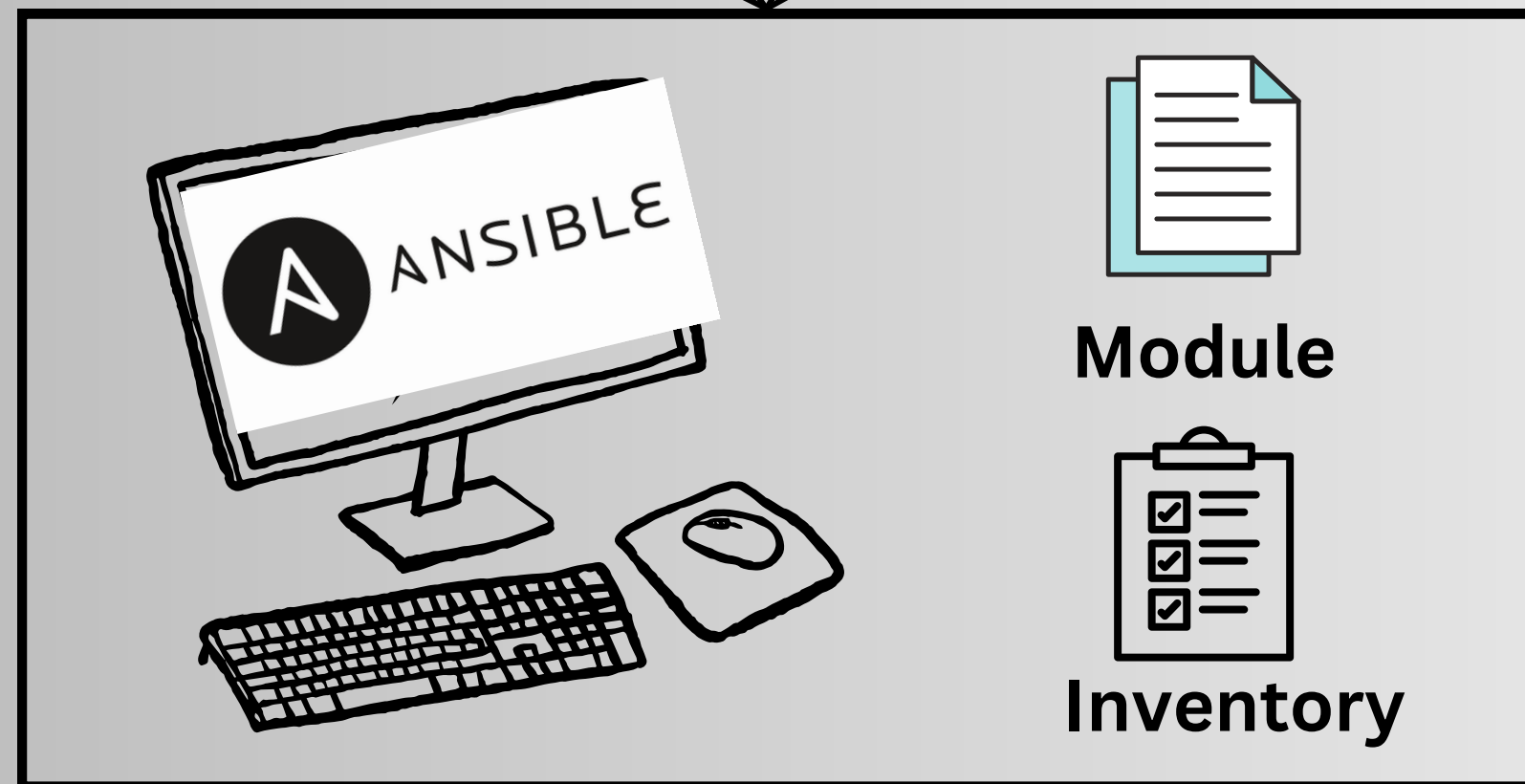
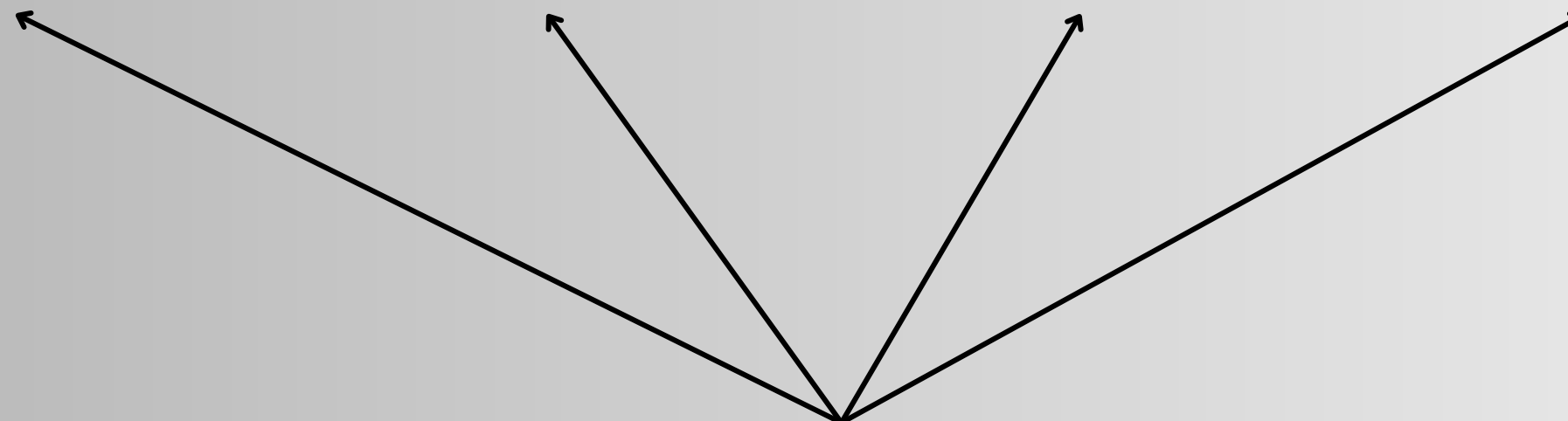
ServerB



ServerC



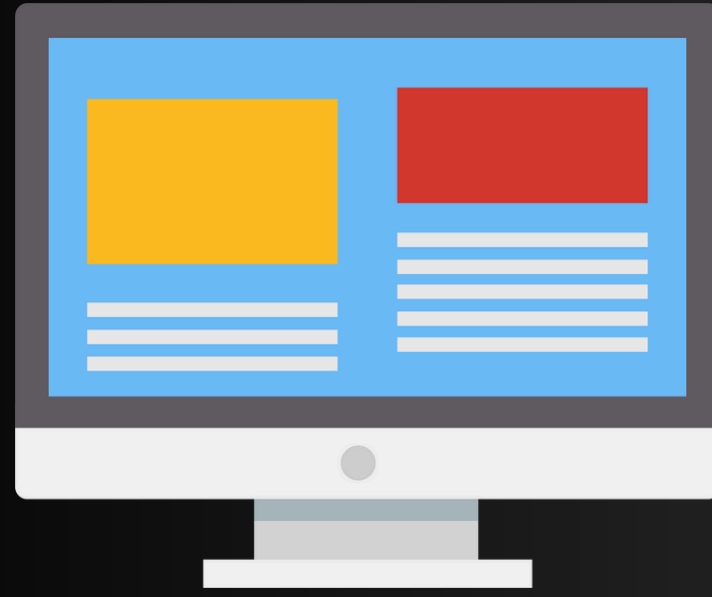
ServerD





Jenkins

SERVER A



Apache Webserver



SERVER B

Summary of Jenkins + Ansible Section

- Quick overview of Ansible
- Installing Ansible plugin
- Jenkins job with 'Invoke Ansible Playbook'
- Perform task on Local vs Remote host
- Ansible job execute as 'jenkins' user
- Execute task which required SUDO access
- Basic project of Updating the Webpage of remote server and publish it.

User Management

- Plugin: Role-based Authorization Strategy
- Create a new user

Manage Jenkins:

- Security -> Authorization -> Role-based
- Manage and Assign Roles
 - Add a new read-only role and assign the permission
 - Assign Roles -> Add new user (nick) and assign read-only role

Environment Variables

We can use pre-defined variables by jenkins like

BUILD_NUMBER, WORKSPACE etc

<https://wiki.jenkins.io/display/JENKINS/Building+a+software+project#Buildingasoftwarproject-belowJenkinsSetEnvironmentVariables>



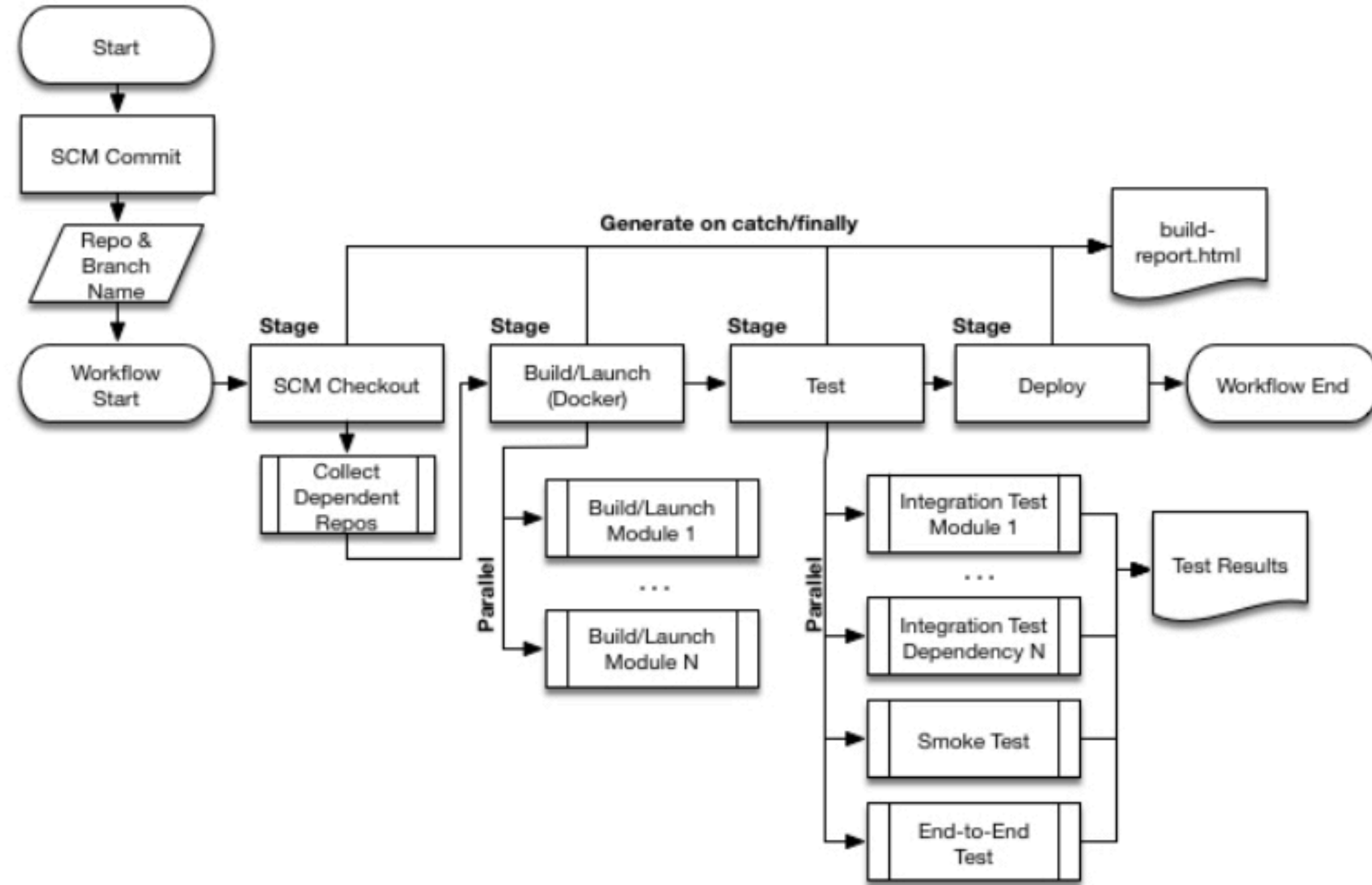
Jenkins

What is maven?

Maven is a build automation tool used primarily for Java projects.

- **Working with Git and Maven**
- **Build jar using Maven**
- **Testing**
- **Deploy jar locally**
- **Graphical representation of results**
- **Send Email notification**

Jenkins Pipeline



<https://www.jenkins.io/doc/book/pipeline/>