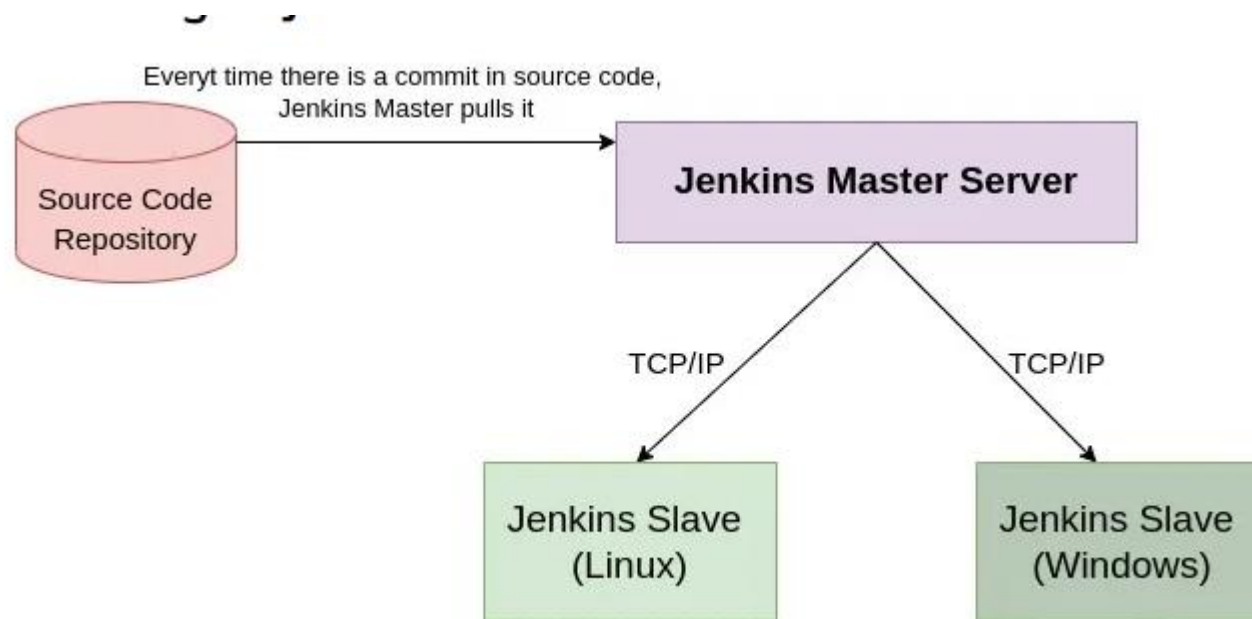


# Jenkins Architecture

OW

# Architecture

- Jenkins performs distributed build using a Master-Slave Architecture



• •

- **Source Code Repository-**

- A repository that contains the source code and all other required project-related files.

- **Master-**

- The main Jenkins Server is the Master Server of Jenkins. This acts as a control server that arranges the defined workflow of the pipeline. The jobs of this Master are as follows:

- Scheduling jobs
- sending builds to slaves to execute the job
- keeping track of the slaves states (online/offline)
- can also directly execute build jobs

- **Slave-**

- The slave is a part of the Jenkins environment that runs on a remote environment. The characteristics of slaves include:

- it listens to the Master instance
- it's job is to do as they are told to, that includes executing build jobs conveyed by the Master

# This architecture works in following steps

1. The developer commits the changes in the source code repository
2. The CI server of Jenkins then checks the repository and pulls the newly changed code at regular intervals.
3. The build server builds the pulled code into an executable file. Feedback is sent to the developers in case of failure of the build.
4. Jenkins deploys the build application on the test server. The developers are alerted if it fails.
5. If the tests are successful and the code is error-free, the tested application is deployed on the production server.

In some cases, files may have different code and require multiple builds and the Jenkins server cannot handle multiple builds simultaneously for this, the Master distributes the workload and allows us to run different builds on different environments each called a Slave.

• •

enough to run the subtask or main task of the Jenkins in a dedicated **executor**.

