



Mahavir Education Trust's  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
Chembur, Mumbai - 400 088  
**UG Program in Artificial Intelligence and Data Science**

## Experiment No. 6

**Aim:** To understand Jenkins Master-Slave Architecture & scale your Jenkins standalone implementation by implementing slave nodes.

**Theory:**

**Jenkins:**

Jenkins is an open-source automation tool written in Java with plugins built for Continuous Integration purposes. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.

If you are working on multiple projects, you may run multiple jobs on each project. Some projects need to run on some nodes, and in this process, we need to configure slaves. Jenkins slaves connect to the Jenkins master using the Java Network Launch Protocol.

**Jenkins Distributed Architecture:**

Jenkins uses a Master-Slave architecture to manage distributed builds. In this architecture, Master and Slave communicate through TCP/IP protocol.

**Jenkins Master:**

- Your main Jenkins server is the Master. The Master's job is to handle:
- Scheduling build jobs.
- Dispatching builds to the slaves for the actual execution.
- Monitor the slaves (possibly taking them online and offline as required).
- Recording and presenting the build results.
- A Master instance of Jenkins can also execute build jobs directly.



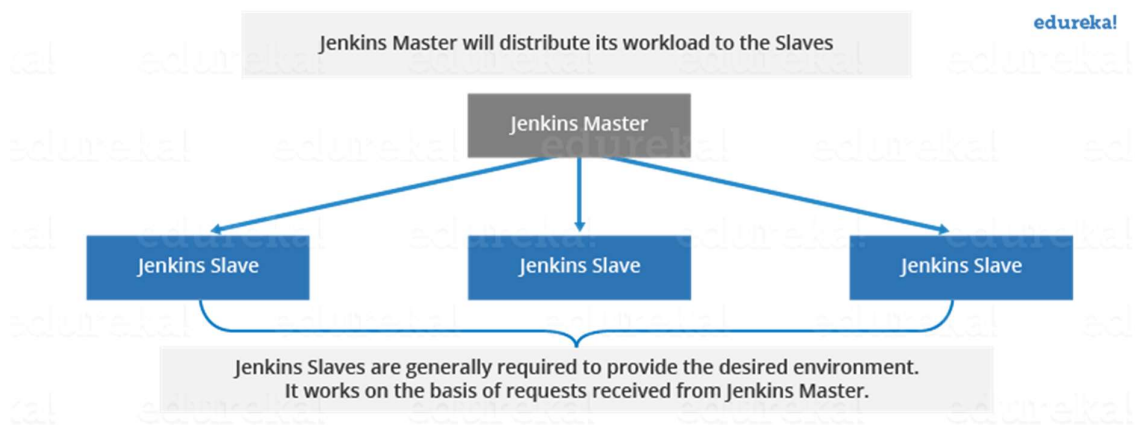
Mahavir Education Trust's  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
Chembur, Mumbai - 400 088  
**UG Program in Artificial Intelligence and Data Science**

Jenkins Slave:

A Slave is a Java executable that runs on a remote machine. Following are the characteristics of Jenkins Slaves:

- It hears requests from the Jenkins Master instance.
- Slaves can run on a variety of operating systems.
- The job of a Slave is to do as they are told to, which involves executing build jobs dispatched by the Master.
- You can configure a project to always run on a particular Slave machine or a particular type of Slave machine, or simply let Jenkins pick the next available Slave.

The diagram below is self-explanatory. It consists of a Jenkins Master which is managing three Jenkins Slave.



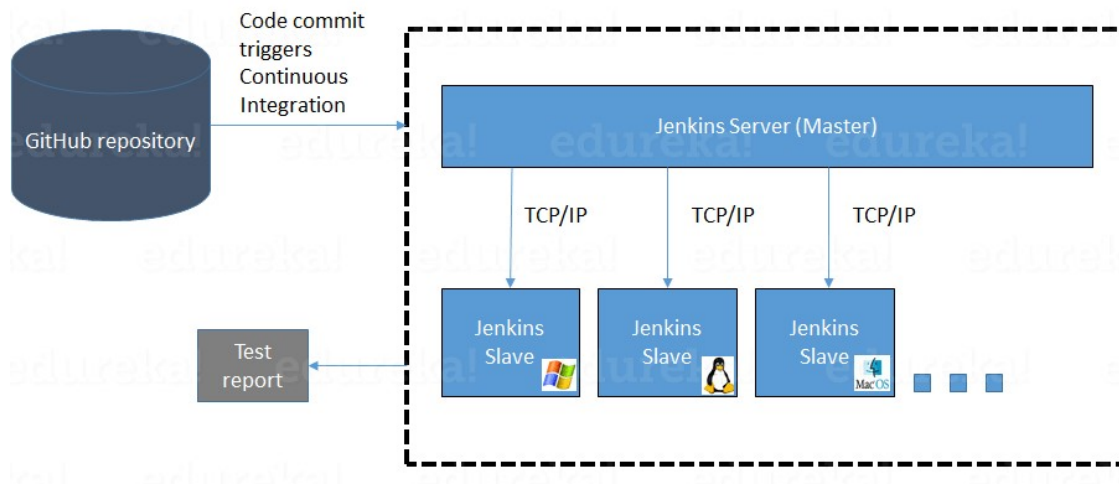
How Jenkins Master and Slave Architecture works?

Now let us look at an example in which we use Jenkins for testing in different environments like Ubuntu, MAC, Windows, etc.



**Mahavir Education Trust's**  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
**Chembur, Mumbai - 400 088**  
**UG Program in Artificial Intelligence and Data Science**

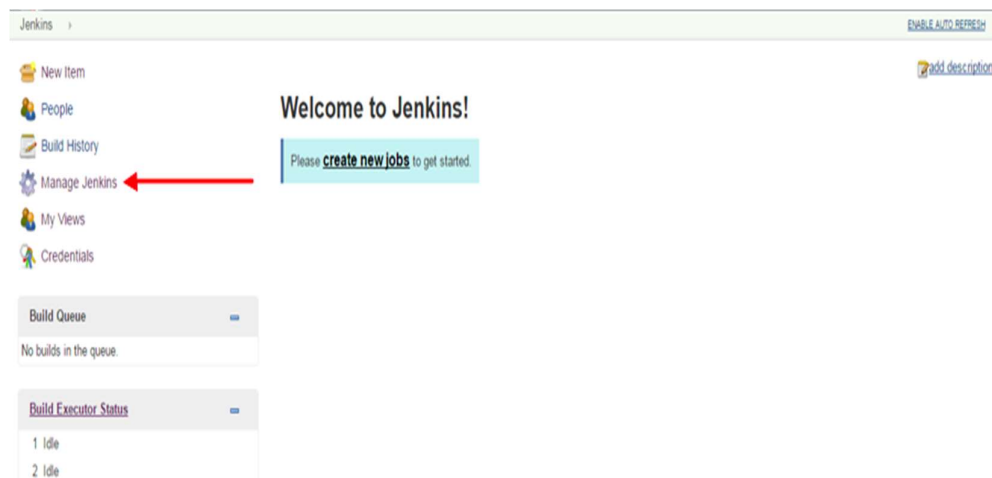
The diagram below represents the same:



The above image represents the following functions:

- Jenkins checks the Git repository at periodic intervals for any changes made in the source code.
- Each builds requires a different testing environment which is not possible for a single Jenkins server. In order to perform testing in different environments, Jenkins uses various Slaves as shown in the diagram.
- Jenkins Master requests these Slaves to perform testing and to generate test reports.

**Output:**





**Mahavir Education Trust's**  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
**Chembur, Mumbai - 400 088**  
**UG Program in Artificial Intelligence and Data Science**



[Load Statistics](#)

Check your resource utilization and see if you need more computers for your builds.



[Jenkins CLI](#)

Access/manage Jenkins from your shell, or from your script.



[Script Console](#)

Executes arbitrary script for administration/trouble-shooting/diagnostics.



[Manage Nodes](#)

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.



[About Jenkins](#)

See the version and license information.

Jenkins > Nodes >

Back to Dashboard

Manage Jenkins

New Node

Configure

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Node name

Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

OK



**Mahavir Education Trust's**  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
**Chembur, Mumbai - 400 088**  
**UG Program in Artificial Intelligence and Data Science**

Jenkins > Nodes >

[Back to Dashboard](#)

[Manage Jenkins](#)

[New Node](#)

[Configure](#)

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Name

Slave1

Description

# of executors

1

Remote root directory

Remote directory is mandatory

Labels

Usage

Use this node as much as possible

Launch method

Launch agent via execution of command on the master

Launch command

No launch command specified

Availability

Keep this agent online as much as possible

Node Properties

☐ Environment variables

☐ Tool Locations

Save



Mahavir Education Trust's  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
Chembur, Mumbai - 400 088  
**UG Program in Artificial Intelligence and Data Science**

Jenkins > Configure Global Security

## Configure Global Security

☒ Enable security

TCP port for JNLP agents ☐ Fixed :  ☒ Random ☐ Disable

Disable remember me ☐

Access Control

### Security Realm

☐ Delegate to servlet container

☒ Jenkins' own user database

☐ Allow users to sign up

☐ LDAP

### Authorization

☐ Anyone can do anything

☐ Legacy mode

☒ Logged-in users can do anything

☐ Allow anonymous read access

Markup Formatter

Treats all input as plain text. HTML unsafe characters like < and & are escaped to their respective character entities.

☒ Prevent Cross Site Request Forgery exploits

Crumbs

### Crumb Algorithm

☒ Default Crumb Issuer

☐ Enable proxy compatibility

### Plugin Manager

☐ Use browser for metadata download

☒ Enable Slave → Master Access Control

Rules can be tweaked [here](#)



**Mahavir Education Trust's**  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
**Chembur, Mumbai - 400 088**  
**UG Program in Artificial Intelligence and Data Science**

Jenkins

Jenkins > Nodes > Slave1

Back to List  
Status  
Delete Agent  
Configure  
Build History  
Load Statistics  
Log

Build Executor Status

### Agent Slave1

Connect agent to Jenkins one of these ways:

- Launch agent from browser
- Run from agent command line:

```
java -jar slave.jar -jnlpUrl http://localhost:8080/computer/Slave1/slave-agent.jnlp -secret 19385e54b9539f6c6272df525dc9e7687efea162e44a83d918ca918294eb9f8e
```

Projects tied to Slave1

None

Mark this node temporarily offline

Jenkins Location

Jenkins URL

System Admin e-mail address

Provide machine name

http://localhost:8080/

Please set a valid host name, instead of localhost

address not configured yet <nobody@nowhere>

Jenkins > Nodes >

Back to Dashboard  
Manage Jenkins  
New Node  
Configure

Build Queue

No builds in the queue

Build Executor Status

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Windows 7 (x86)	In sync	11.11 GB	20.14 GB	11.11 GB	0ms
	Slave1		N/A	N/A	N/A	N/A	Time out for last 1 try
	Data obtained	32 min	32 min	32 min	32 min	32 min	32 min

Refresh status

master

1 Idle

2 Idle

Slave1 (offline)



**Mahavir Education Trust's**  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
**Chembur, Mumbai - 400 088**  
**UG Program in Artificial Intelligence and Data Science**

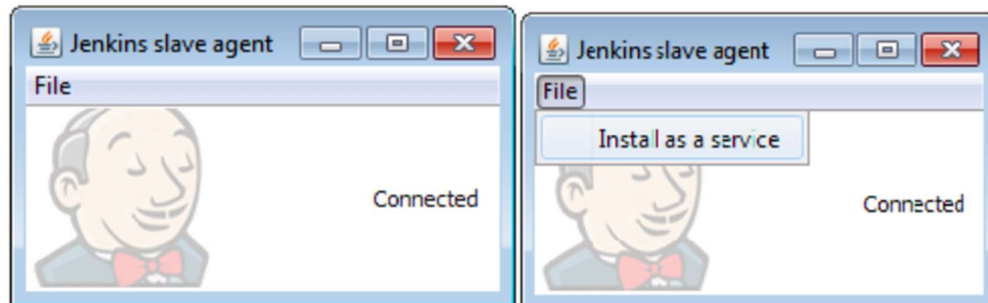
The screenshot shows the Jenkins web interface. At the top, there's a navigation bar with 'Jenkins' and a search bar. Below it, a breadcrumb trail shows 'Jenkins > Nodes > Slave1'. On the left, there's a sidebar with links: 'Back to List', 'Status', 'Delete Agent', 'Configure', 'Build History', 'Load Statistics', and 'Log'. The main content area is titled 'Agent Slave1' and contains instructions on how to connect the agent to Jenkins. It lists two methods: 'Launch agent from browser' (with a 'Launch' button) and 'Run from agent command line'. The command line is: `java -jar slave.jar -jnlpUrl http://localhost:8080/computer/Slave1/slave-agent.jnlp -secret 19385e54b9539f6c6272df525dc9e7687efa162e44a83d918ca918294eb9f8e`. Below the command line, it says 'Projects tied to Slave1' and 'None'. There's also a 'Mark this node temporarily offline' button in the top right corner.

The screenshot shows a Windows security warning dialog box. The title is 'Do you want to run this application?'. It features the Java logo (a cup of coffee) and the following information: **Name:** Jenkins Remoting Agent, **Publisher:** Infradna Inc (Kohsuke Kawaguchi), and **Locations:** http://localhost:8080. Below this, it says 'Launched from downloaded JNLP file'. A warning message states: 'This application will run with unrestricted access which may put your computer and personal information at risk. Run this application only if you trust the locations and publisher above.' At the bottom, there's a 'More Information' link with an information icon, and two buttons: 'Run' and 'Cancel'.





Mahavir Education Trust's  
**SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE**  
Chembur, Mumbai - 400 088  
**UG Program in Artificial Intelligence and Data Science**



- Back to List
- Status
- Delete Agent
- Configure
- Build History
- Load Statistics
- Script Console
- Log
- System Information
- Disconnect

**Build Executor Status**

1 Idle



## Agent slave1

Connected via JNLP agent.

### Projects tied to slave1

None

### **Conclusion:**

Hence, we have successfully understood & configured Master-Slave Architecture & Implemented slave nodes for standalone Jenkins program.