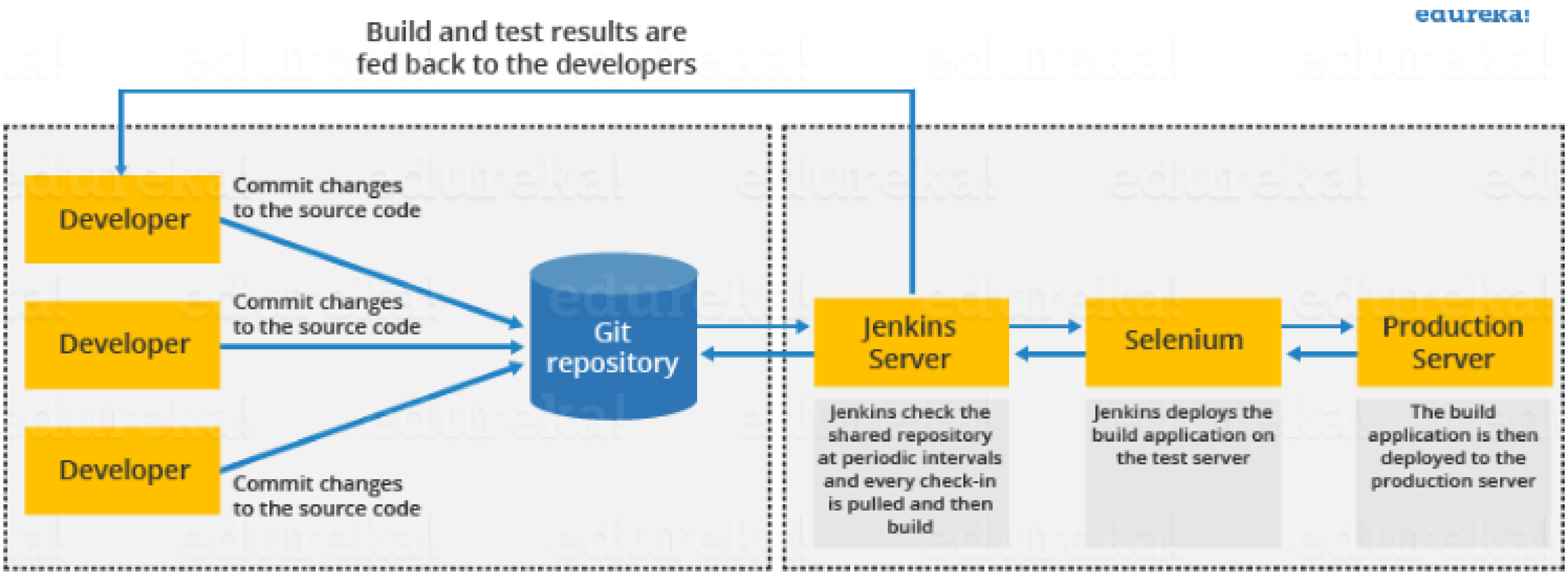
Topic 6: Distributed Jenkins

Simple Jenkins Flow



Build and test results are fed back to the developers

Commit changes to the source code

Commit changes to the source code

Commit changes to the source code

Jenkins check the shared repository at periodic intervals and every check-in is pulled and then build

Jenkins deploys the build application on the test server

The build application is then deployed to the production server

Problems ...

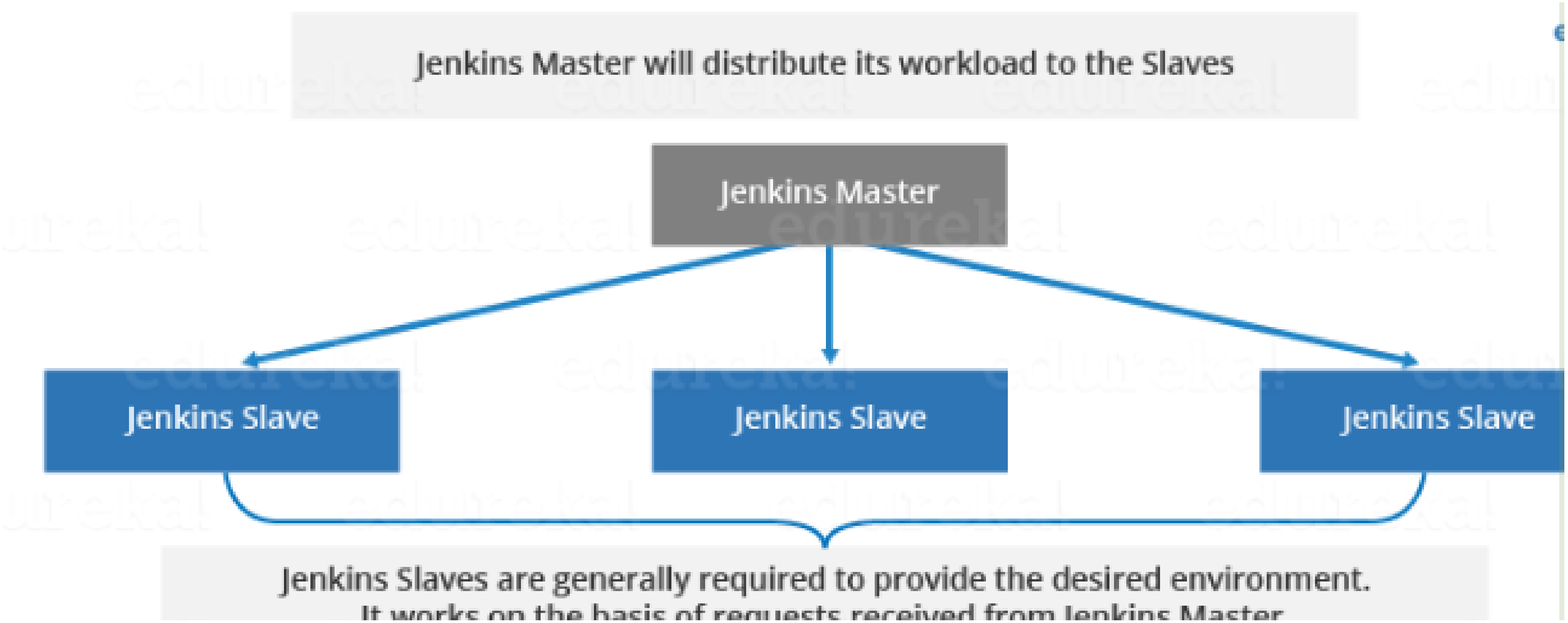
This single Jenkins server was not enough to meet certain requirements like:

* Sometimes you might need several different environments to test your builds. This cannot be done by a single Jenkins server.

* If larger and heavier projects get built on a regular basis then a single Jenkins server cannot simply handle the entire load.

To address the above-stated needs, Jenkins distributed architecture came into the picture.

Jenkins Architecture



Jenkins Master will distribute its workload to the Slaves

Jenkins Slaves are generally required to provide the desired environment.
It works on the basis of requests received from Jenkins Master

Add Slave Nodes

Step 1: Install the necessary packages.

You will need to install some packages on the agent node, such as Java, as well as any tools that you may require to run your builds (Git, Maven, Ant, Gradle, etc.).

Add Slave Nodes

Step 2: Create user on slave node

Now we need to create a user on the agent. The Jenkins master will log into the agent as this user, and all build jobs will execute as this user.

> User should exists (with password/key)

> The remote directory to be created and should have proper permissions

Add Slave Nodes

Step 3: Add node on Master

Go to Manage Jenkins – Manage nodes – New node to add the node

Fill the required details:

Name: Name of the slave node

Remote root directory: Path of the root directory

Labels: give any label

Usage: select use this node as much as possible

Add Slave Nodes

Step 3: Add node on Master

Go to Manage Jenkins – Manage nodes – New node to add the node

Fill the required details:

Name: Name of the slave node

Remote root directory: Path of the root directory

Labels: give any label

Usage: select use this node as much as possible

Launch Method

> Launch agent by connecting it to the master

> Launch agent via execution of command on the controller

> Launch agents via SSH