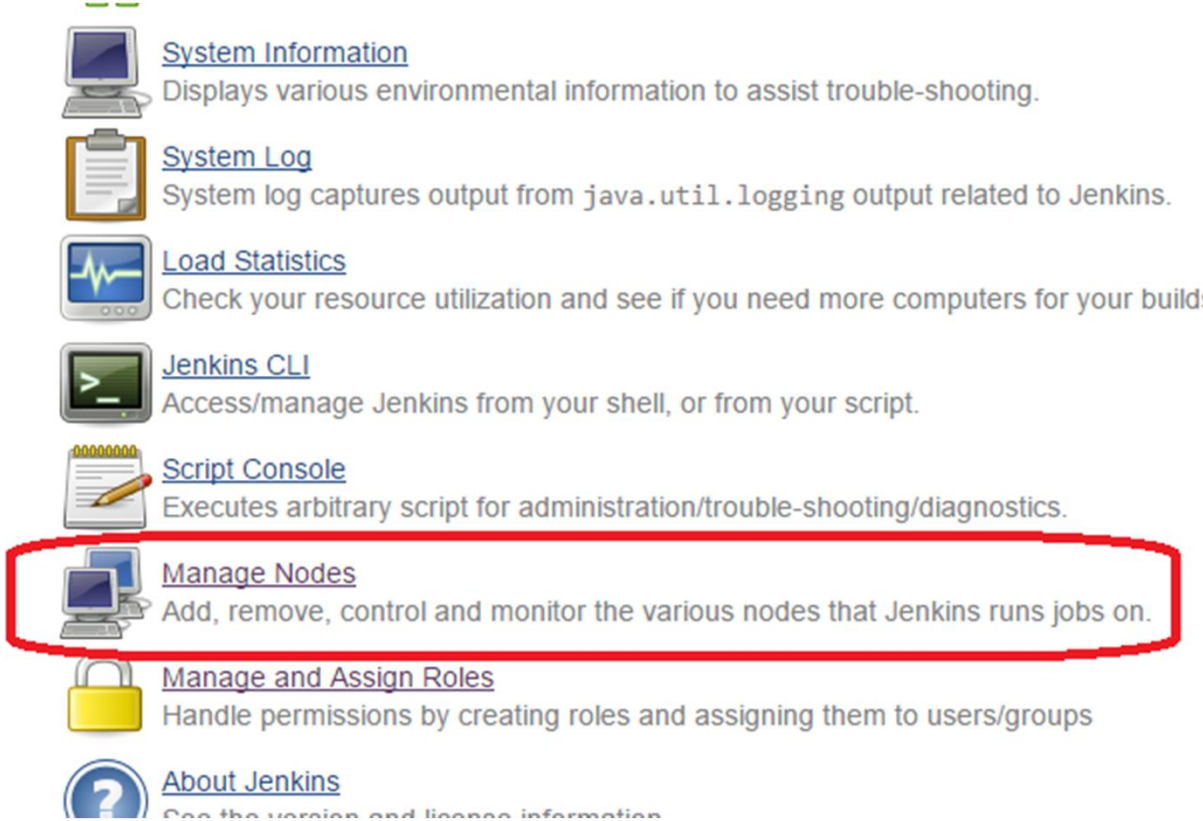
Jenkins – Slave Nodes

Jenkins – Master Configuration

- In this session we will look at using additional machines to build jenkins jobs
- The additional machines are needed to reduce work loads and are called ***Slaves***
- Go to Manage Jenkins → Configure Global Security
- Check *'TCP Port for JNLP Agent'* and **Save**
- You can keep a fixed port or randomly generate a port number

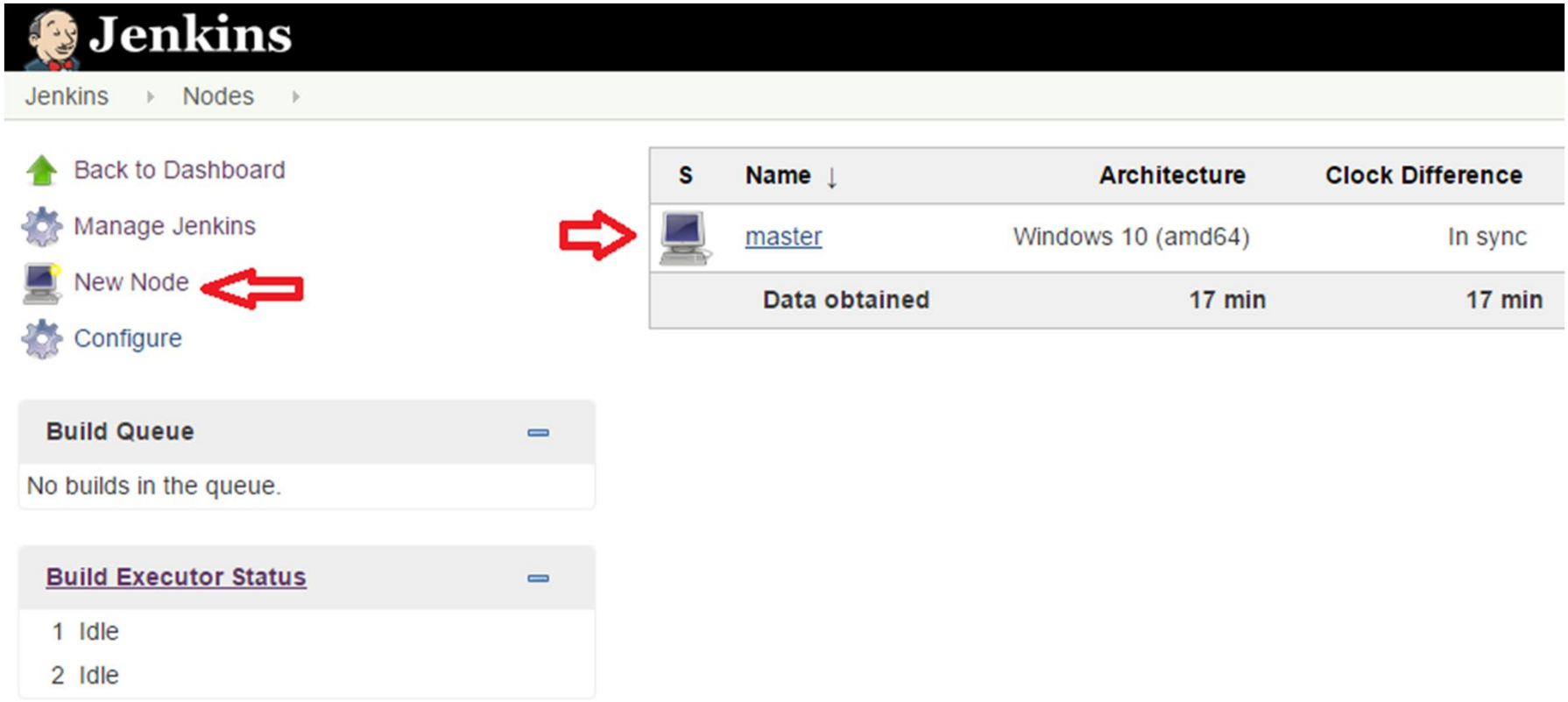
Jenkins – Create Slave Node

- In this exercise, we will utilize another windows machine as Slave
- Lets configure that as slave using Manage Jenkins → Manage Nodes



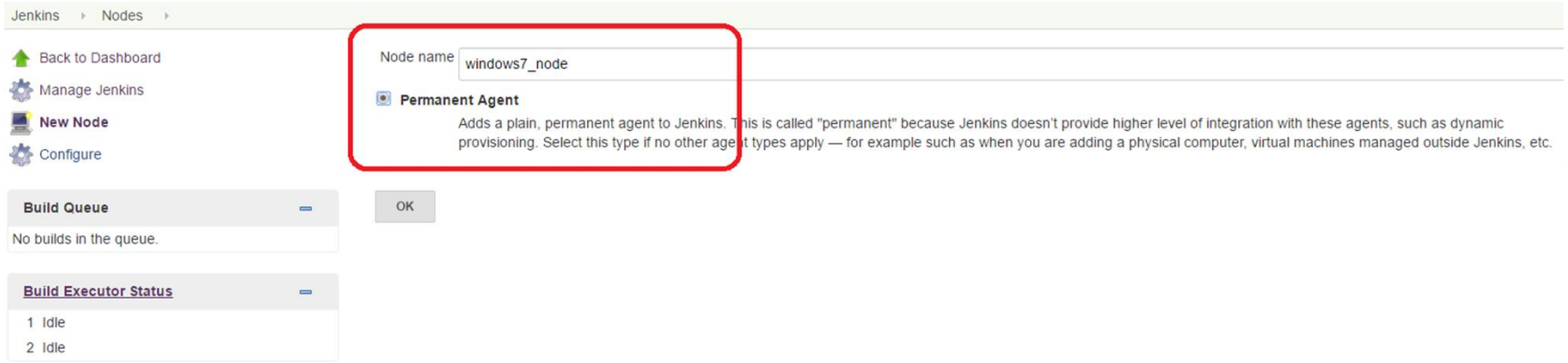
Jenkins – Create Slave Node contd..

- You can see only the master listed here
- Click on ***'New Node'*** link in Left Menu



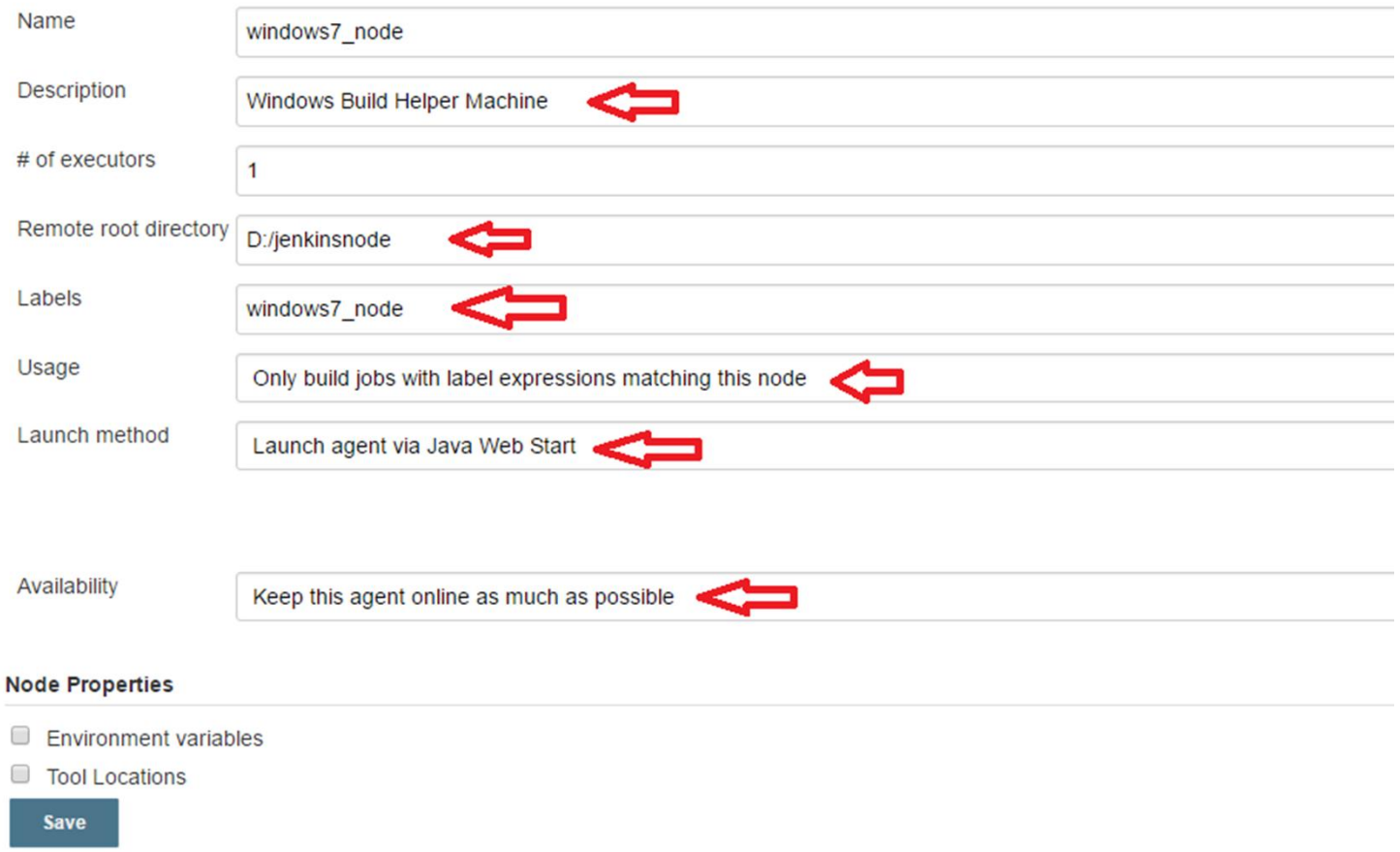
Jenkins – Create Slave Node contd..

- Provide a meaningful name for the slave node
- Click on ‘Permanent Agent’ and OK



Jenkins – Create Slave Node contd..

- Note – You need to create a folder D:/jenkinsnode on the Slave Machine
- Create a label as windows_node
- Launch method will be Java Web Start



Jenkins – Connect to slave

- You can see that the newly created slave node is marked as 'X'
- Indicating that the node is not accessible to master
- We will fix this by trying to connect to the slave node
- Click on the windows7_node

S	Name ↓	Architecture	Clock Difference	Free Disk Space
	master	Windows 10 (amd64)	In sync	266.14 GB
	windows7_node		N/A	N/A
	Data obtained	25 min	25 min	25 min

Jenkins – Connect to slave contd

- You can see that there are multiple options
- 1 -- is a Launch Agent from browser
- 2 -- Run agent from command line

- Lets use the option 2
- You need to have slave.jar to be available on Slave Machine
- Click on slave.jar and it will be download
- Copy slave.jar and also the jnlp url, secret key to slave machine

Jenkins – Slave Machine Configuration

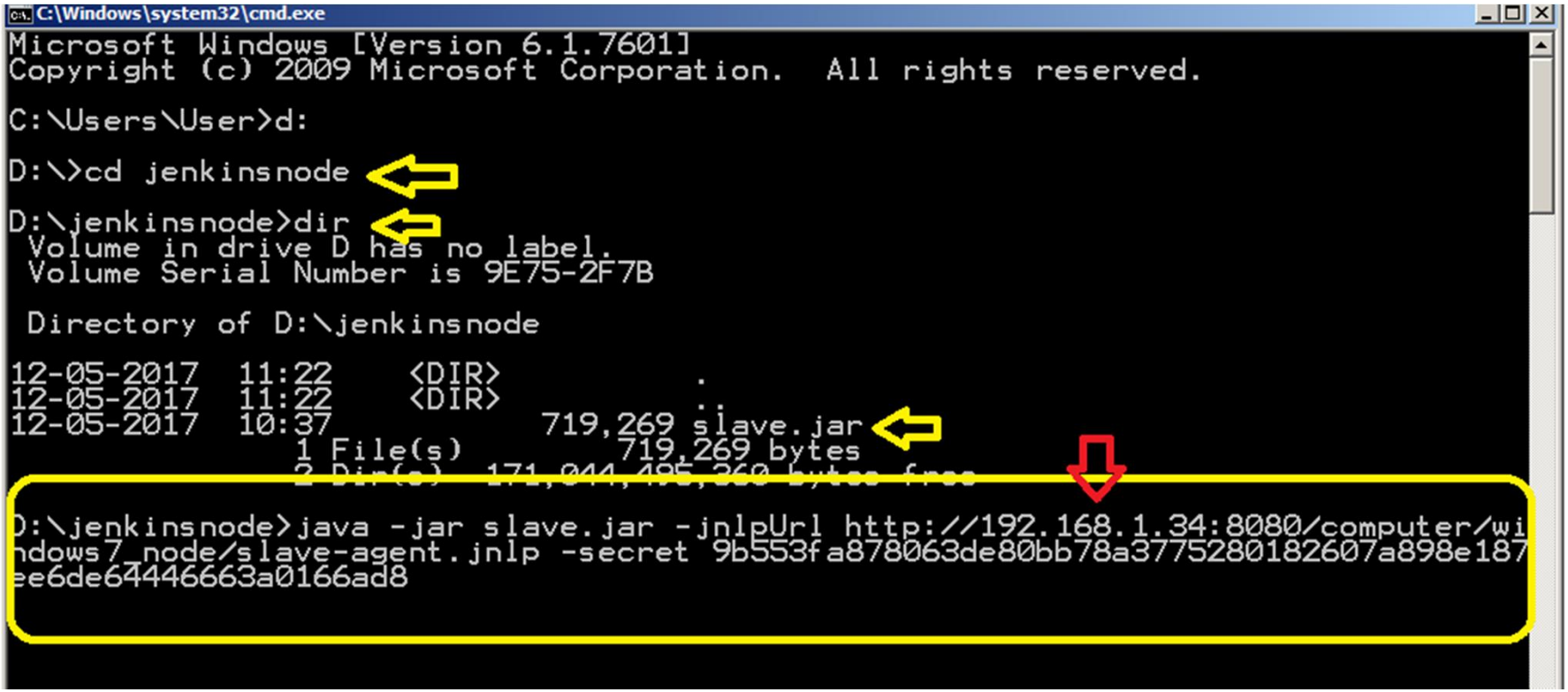
- NOTE : These commands are run on Slave Node

- Create a folder D:/jenkinsnode
- Copy the slave.jar into D:/jenkinsnode
- Open a command prompt and navigate to D:/jenkinsnode directory
- Execute the command which you copied in previous step
- **NOTE:** Change the url from localhost to IP Address of Jenkins Master server

java -jar slave.jar -jnlpUrl http://<IP Address of Jenkins Master>:8080/computer/windows7_node/slave-agent.jnlp -secret 9b553fa878063de80bb78a3775280182607a898e187ee6de64446663a0166ad8

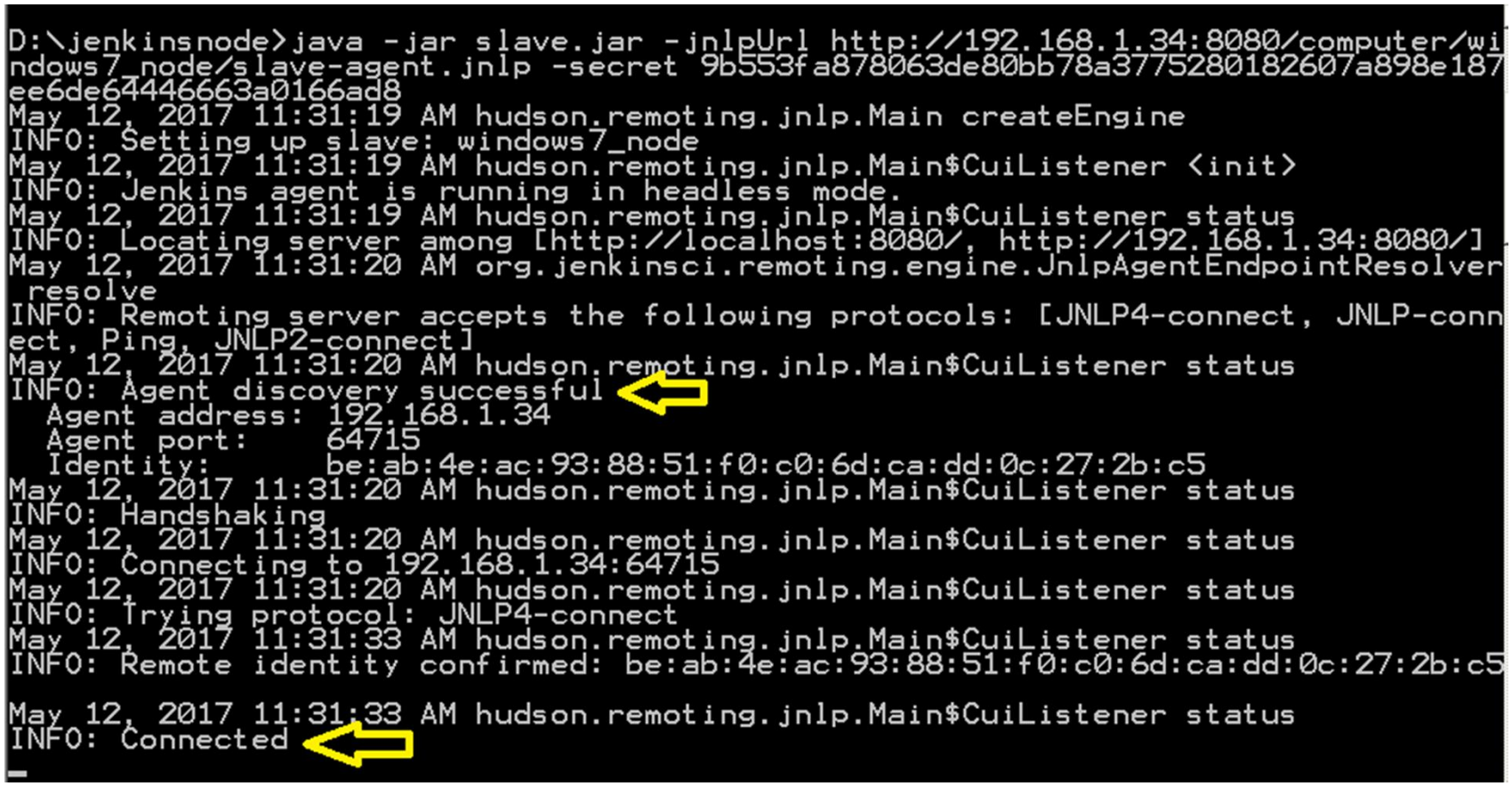
Jenkins – Slave Machine Configuration contd

- In the screen shot below, you can see that the command is run from D:/jenkinsnode
- D:/jenkinsnode folder also containes **slave.jar**
- The IP address of Jenkins Server (Master) has replace 'localhost'
- Note: This command can be run from anywhere as long as 'slave.jar' is available on Java Class Path



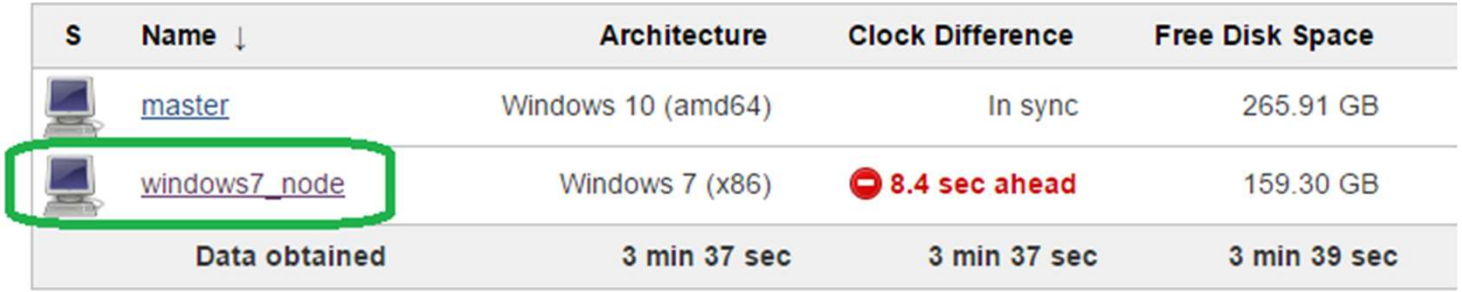
Jenkins – Slave Machine Configuration contd

- The connection should be successful
- Now go back and Check the status of node in Jenkins Server



Jenkins – Back to Master

- Go to Manage Jenkins → Manage Nodes
- You can see that the JNLP agent is successfully running on Slave machine
- Master and Slave are able to communicate
- Next lets configure a simple job to check if remote build is working



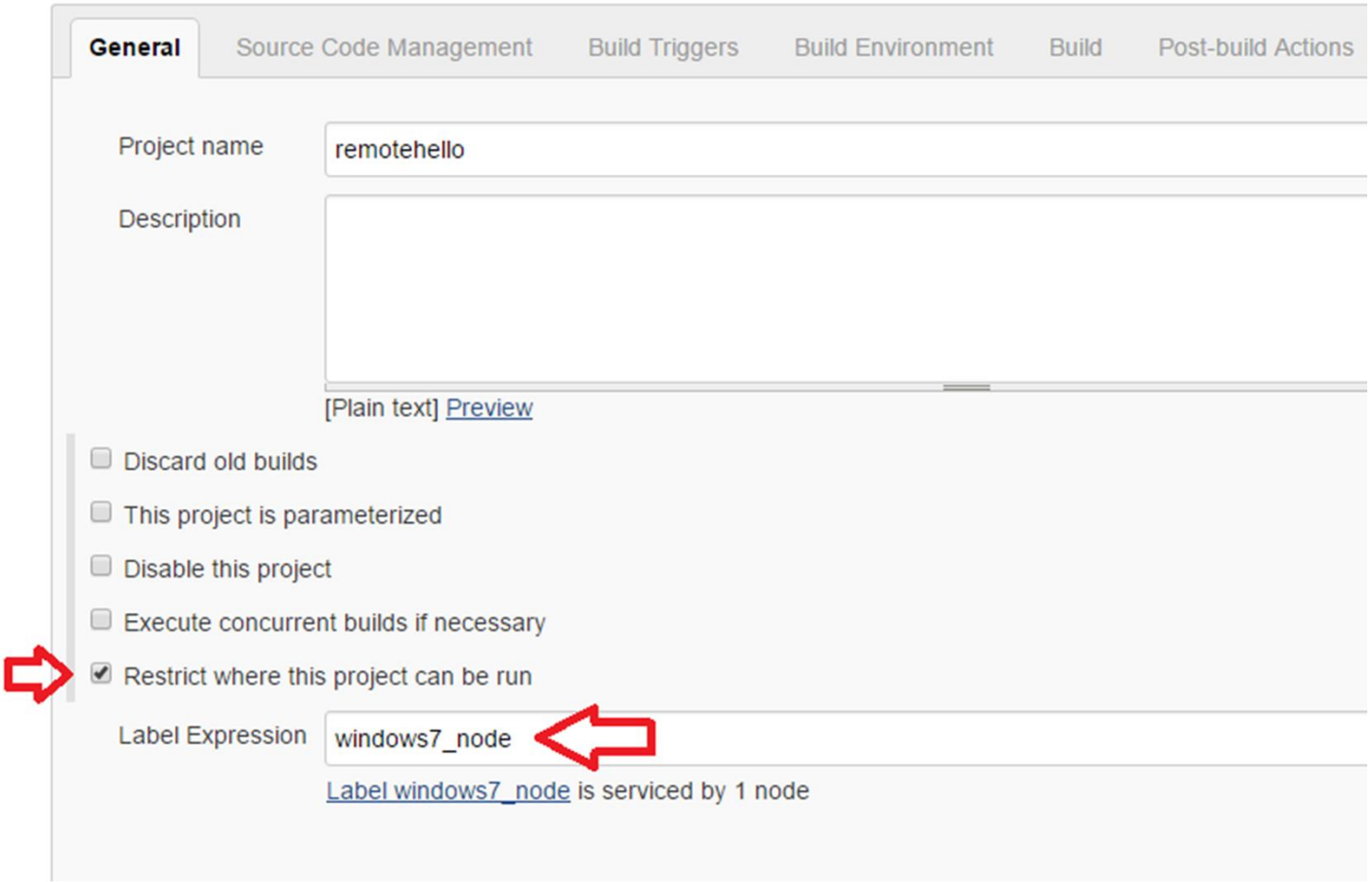
S	Name ↓	Architecture	Clock Difference	Free Disk Space
	master	Windows 10 (amd64)	In sync	265.91 GB
	windows7_node	Windows 7 (x86)	8.4 sec ahead	159.30 GB
	Data obtained	3 min 37 sec	3 min 37 sec	3 min 39 sec

Jenkins – Create a job

- Lets Create a job and build it on remote slave
- Go to Dashboard → New Item
- Give the name as remotehello and select ‘Free Style Project’ and OK

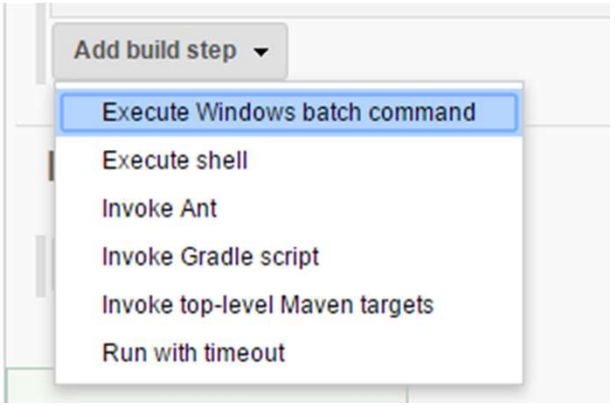
Jenkins – Create a job contd

- Click on ‘Restrict where this Project can be Run’
- Provide the Label of the Slave Machine which was created during Slave machine configuration



Jenkins – Create a job contd

- Lets try with a simple build here, Click on Add Build Step
- Select 'Execute windows batch command'
- Just put a simple echo "Hello" and Save



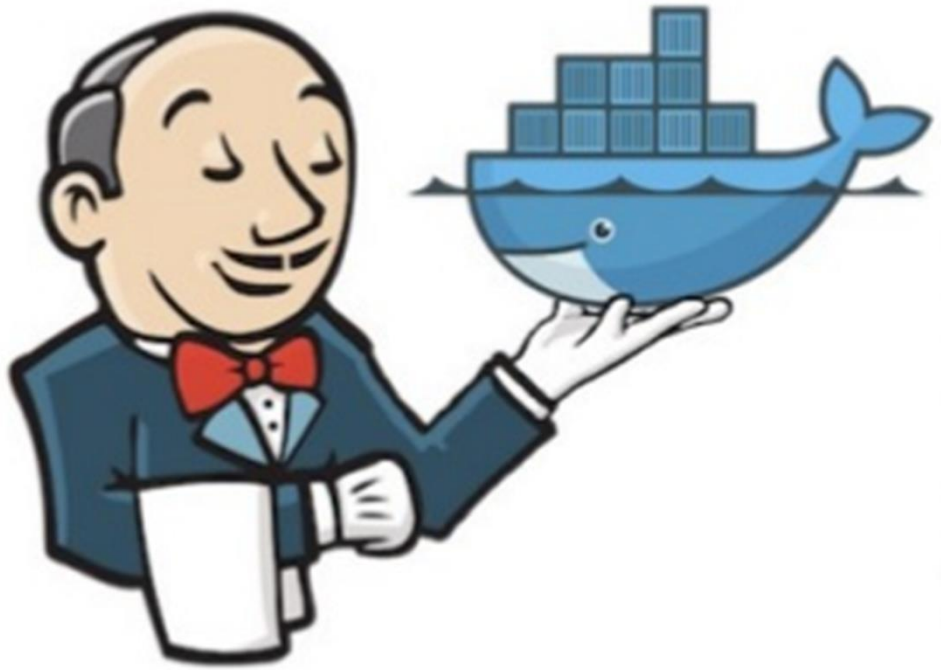
Jenkins – Build the remote job

- Build the job and check the Console
- As you can see, the job is built on remote slave node

Jenkins & Docker

- Docker is a Open source container management platform
- Light weight & can bring up any application stack (open source or free to use) within seconds

- Docker Plugin
  - Dynamically provision slaves
  - Run build
  - Tear-down slave
- Slaves in the Cloud



Jenkins & Docker

- Jenkins can bring up Slaves (Docker containers) on Demand
- Master can only be built on Docker