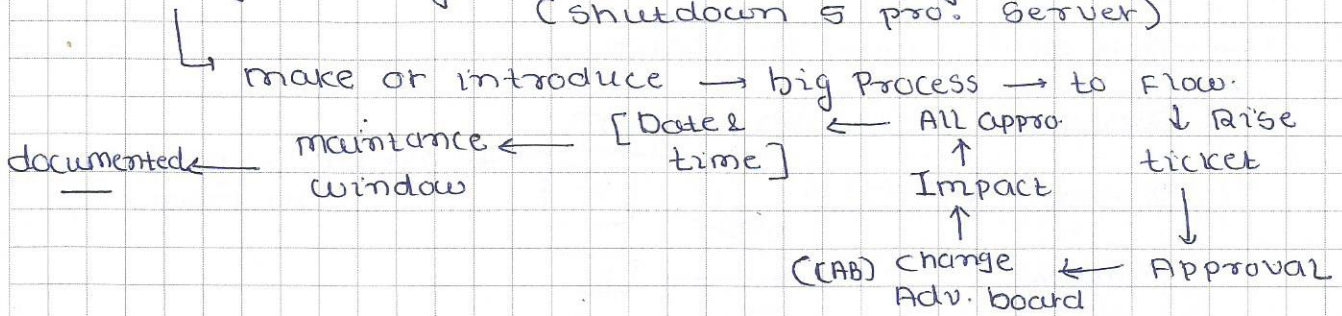
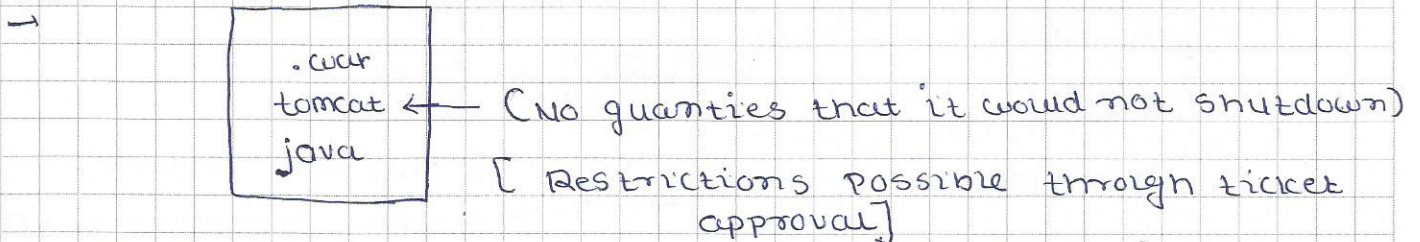


* Change management - maintaining App. (Prod. Env)
(Shutdown & prod. Server)



→ Instructions - assign to someone & start working on it

→ Systematic Proc. → Require approval & Everything.



→ making any change to Env. (or infrastructure)

↳ configuration change - gurdian. (system Related)
Change. Manage. (conf. safe & secure & work as a guardian).

* Configuration management -

- Install java
 - Install Tomcat
 - Copy .war file
 - Restart/start tomcat.
- } [500 production Server]

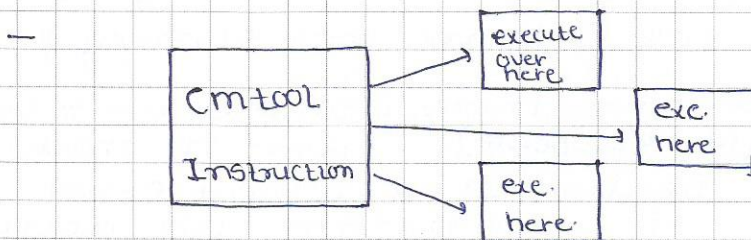
1. Manual Approach.

2. Scripts - (Shell scripts) → Instructions in single File

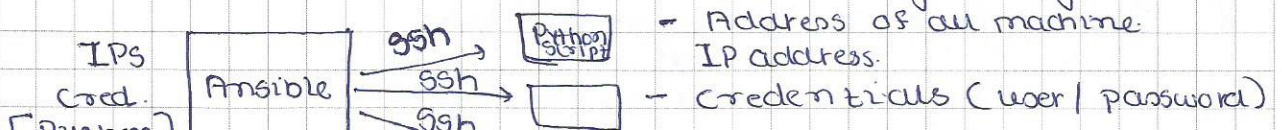
③ ↳ Login Required - Before Running somebody
Run java already.

- {tries to override again} - it blindly runs again

3. Configuration management tool. (Ansible)



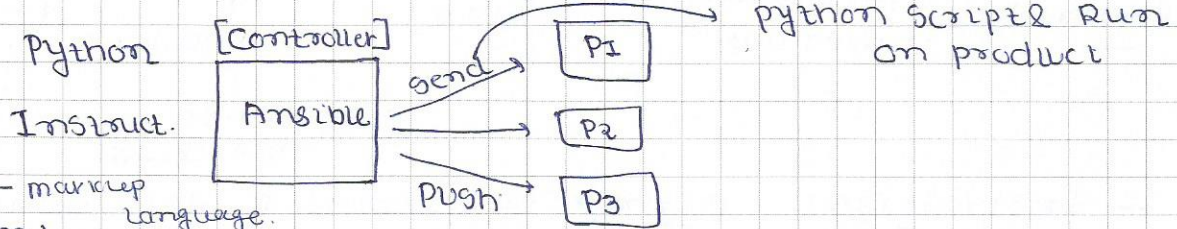
- change already there - I see skip it (Intelligent)
- don't disturb environment again again.



* Agent Ansible - Agentless (No overhead on prod. server) ②
Run your infrastructure.

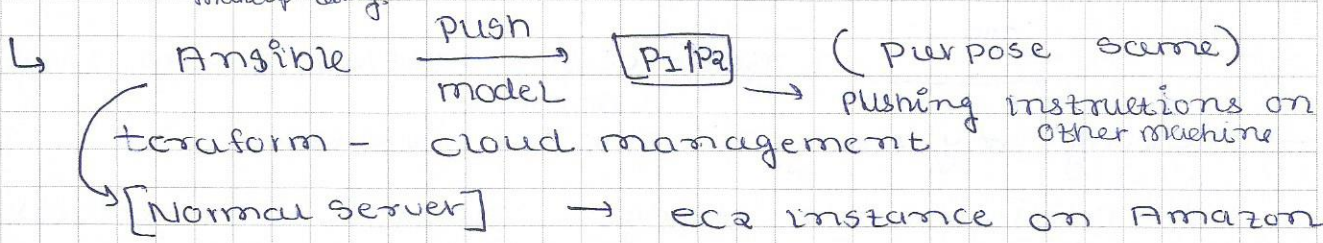
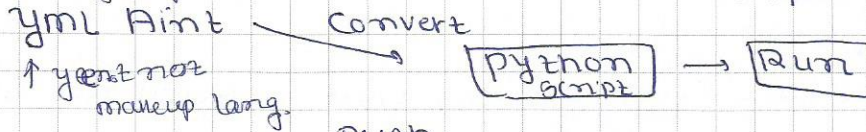
- you don't need to touch

- Ansible built on Python



design page, - markup
lang
file
color) → .yaml - language.
(Language)

YAML (yaml) - structure Serialization lang.
(spaces matter a lot)



* Ansible - one of the conf. - Puppet (needs Agents on machine)
- Salt

→ Config. change / operational change. (changing state)

- New Change (introducing new change)
→ state of server (Not introducing something New)

→ Process of configuration - change. mang. (what should be done)
↳ config. audit (it's safe, properly done).

→ Config. management - installing software.

→ "snowflake system". - Unix - (other tools)

- Ansible - ~~greek~~ greek word (controlling space)

* Provisioning - EC2 (server provisioning)

* Orchestration - management / shutdown & server, (stop some server) run in order

* Automation - status, start/stop server perform all activity

Idempotence ≈ ~~no~~ skip changes

* Orchestration

task1	- sys1
task2	- sys2
task3	- sys3

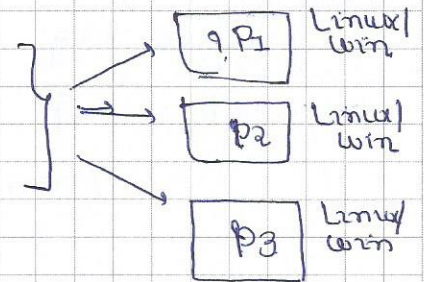
Yml → write instruction | human readable. ③

* Installation →

Authentication [1. SSH keys.
2. user/password] to connect to machine
Ansible uses

* Prerequisite

- ↳ Python (2.6+)
- ↳ SSH
- ↳ only for Linux based machine.
- ↳



Inst

terraform — provisioning
Ansible → for change.

(Windows Powershell)

(Controller machine)

> ssh devops@123.10.20.30

↳ ~/labs/ansible/install # (Ansible controller)

> ssh devops@52.15.64.90 (Node 1)

→ ansible --version
ansible 2.9.6
python - 2.7.2

- cd /etc/ansible → ansible.cfg (change config)
hosts (Remote server ips)
roles

> vi hosts
[webservers] groups } hosts file

node 52.15.64.149.
30.10.15.30

how to connect. Authentication. > ansible -m ping
↓
all } ansible all -m ping -u devops
> ansible all -m ping -u devops -k (ask password)

SSH Password:

- Host key checking (ok to connect or not)
↳ disable host key checking

> vi ansible.cfg

#3 line > host-key-checking = False. (enable first)

> ansible all -m ping -u devops -k [ask password]

> all should be on same network

(4)

* Key-base authentication. (ssh-key)

inmate> ssh-keygen
done

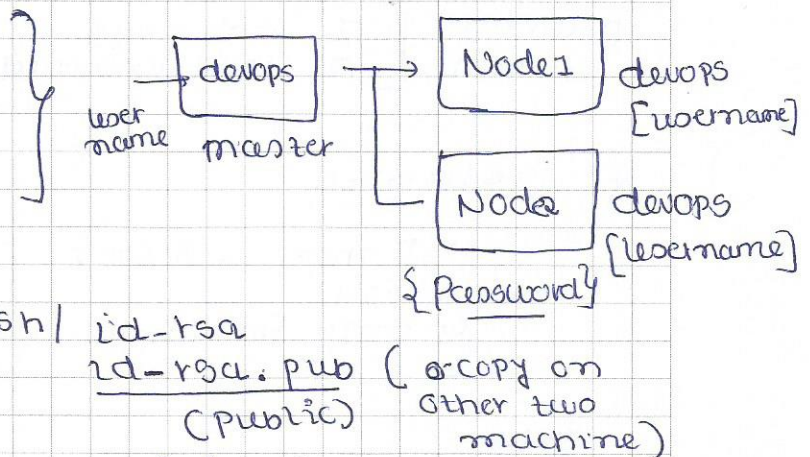
root@ --->
↑ create key for root user

devops@1:~\$ ssh-keygen

/home/devops/.ssh/id_rsa

id_rsa.pub
(public)

(or copy on other two machine)



> ssh-copy-id 172.31.7.9 (private ip)
password.

> ansible all -m ping (password based authentic. it's more secure)

(need to copy pub keys to all the machine)
(password -)

ansible all -m ping -u devops -k -vvvv (verbose)
↳ username of host machine.

detail Information

Background)

ping → ping.py (convert → execute a python script on Remote machine)

> cd /etc/ansible
hosts } many people working on same hosts.
(don't used) anywhere

> mkdir Ravi
> vi inv (hosts file)

[aws]

52.15.64.87 ansible-ss user = devops
3.15.7.87 ansible

password = test

> ansible all -m ping } → By default it goes then
etc/ansible/hosts.
(Remove it)

> ansible all -i /root/ravi/inv -m ping -u devops -k
Inventory File

ansible aws -i /root/ravi/inv -m ping -u devops -k

> vi ansible.cfg > 14 inventory = /.../ (global Level
No good idea)