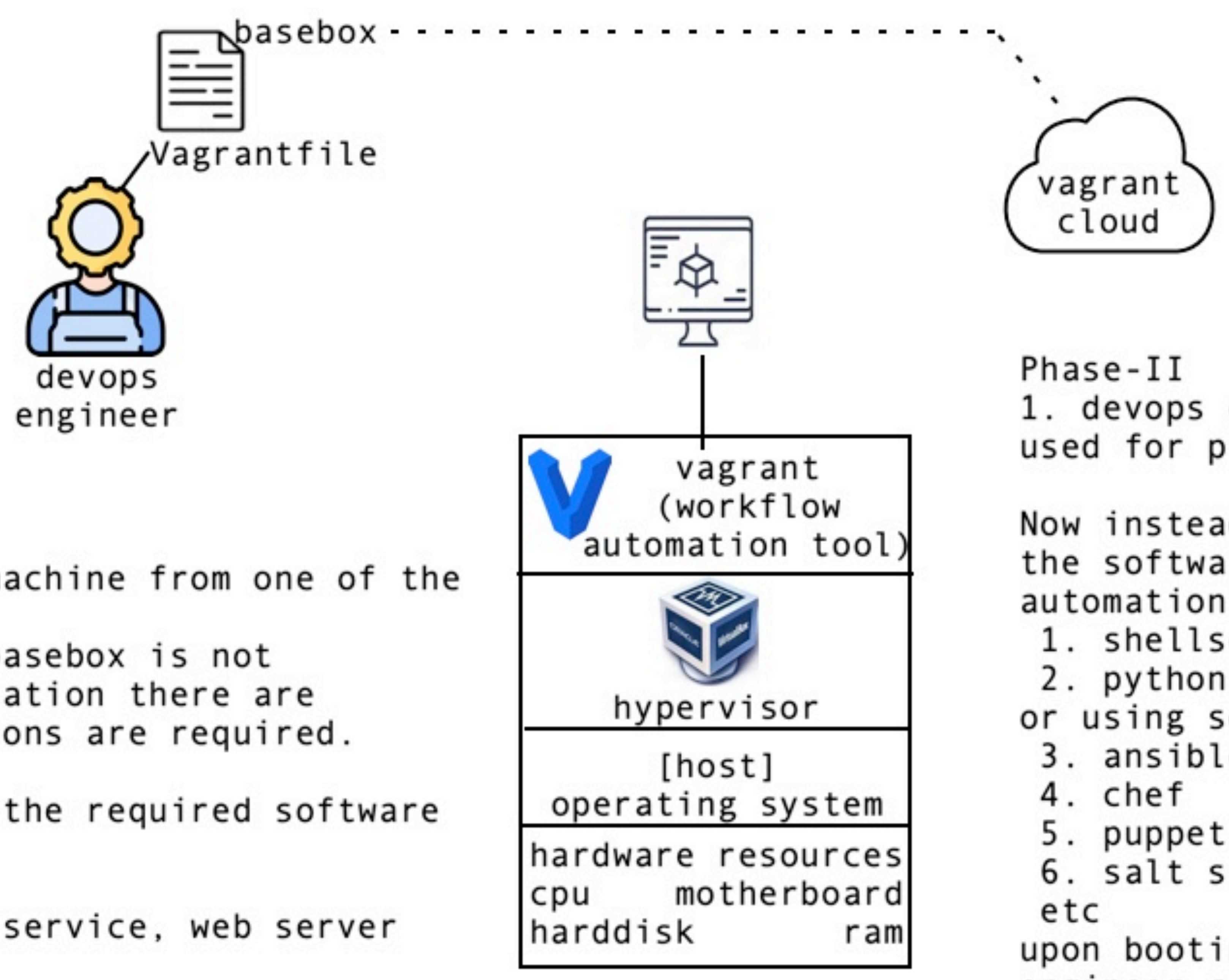
```
netbanking
|-src
  |-main
    |-java
    |-resources
    |-webapp
      |-WEB-INF
        |-web.xml
|-pom.xml
|-Vagrantfile
```

The Virtual Machine is provisioned for developing/testing or deploying the software application, so usually the machine configuration file will be written aspart of the project directory for whom we are creating the machine for



Phase-I
1. devops engineer provisioned the vagrant machine from one of the basebox by writing Vagrantfile
The vagrant machine that is created out of basebox is not sufficient to deploy/run the software application there are additional software packages and configurations are required.

2. devops engineer has to manually download the required software packages/libraries
3. install them on the vagrant machine
4. configure these software packages (initd service, web server configurations …)
so that the vagrant machine will be ready for usage.

since multiple people in the team requires same vagrant machine environment, the devops engineer can expose this machine as an image and distribute to the others in the team.

This approach has few dis-advantages:
1. the time required for setting up the virtual machine by installing software packages, libraries and configurations will be more
2. it is an repeatitive process in setting up the virtual machine when there is a change in software stack or patches or upgrades are available
3. since it is manual process in setting up the env, it is prone to errors also

Phase-II
1. devops engineer writes the Vagrantfile with basebox to be used for provisioning

Now instead of manually downloading, installing and configuring the software packages the devops engineer can rely on automation scripts in setting up the environment like
 1. shellscripting
 2. python
or using software configuration management tools like:
 3. ansible
 4. chef
 5. puppet
 6. salt stack
 etc
upon booting up the virtual machine using vagrant, the devops engineer can manually run these configuration management scripts on the virtual machine that takes of installing & configuring the required software packages/libraries so that the env will be ready for usage

The automation scripts that are written by the devops engineers might have to executed by others as well in team inorder to setup the virtual machine environment like devopers, qa etc. But there are challenges involved in running those scripts upon first-boot of the virtual machine like
 1. there can be many scripts that needs to be executed in an specific order, that may not be aware by everyone in the team
 2. the specific details of how to run those or commands needs to be used in running these scripts may not be known to everyone in the team

Instead of devops engineers or anyone who brings up the virtual machine to run these automation scripts manually in settingup the virtual machine environment, if vagrant itself can run these automation scripts upon the first-boot of the virtual machine then the vm will be available always for usage without any difficulties

The devops engineer aspart of the Vagrantfile should declare automation scripts that needs to be executed upon the first-boot machine as provisioners so that vagrant can help in executing them, so that all the above challenges we discussed will be resolved.