Vagrant: Sync Directories

1 Display Global Vagrant Status and Running VM State

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant global-status
==============================================================================
id       name    provider   state   directory
------------------------------------------------------------------------------
------------------
de2e78f  default virtualbox running /home/tiago-paquete/Testdir/vagrant/
myvirtualmachines

The above shows information about all known Vagrant environments
on this machine. This data is cached and may not be completely
up-to-date (use "vagrant global-status --prune" to prune invalid
entries). To interact with any of the machines, you can go to that
directory and run Vagrant, or you can use the ID directly with
Vagrant commands from any directory. For example:
"vagrant destroy 1a2b3c4d”
==============================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant status
==============================================================================
Current machine states:

default                   running (virtualbox)

The VM is running. To stop this VM, you can run `vagrant halt` to
shut it down forcefully, or you can run `vagrant suspend` to simply
suspend the virtual machine. In either case, to restart it again,
simply run `vagrant up`.
==============================================================================
```

2 Inspect the Current `Vagrantfile` Configuration

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ ls
Vagrantfile
==============================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ cat Vagrantfile
==============================================================================
Vagrant.configure(2) do |config|
  config.vm.box = "ubuntu/jammy64"

  # Private network with static IP
  config.vm.network "private_network", ip: "192.168.56.10"

  # Public network (bridged), no static IP
  config.vm.network "public_network", bridge: "wlp0s20f3"

  config.vm.provider "virtualbox" do |vb|
    vb.memory = 1024
    vb.cpus = 2
  end
end
==============================================================================
```

3 SSH into the Running Ubuntu Guest

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant ssh
==========================================================================
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 5.15.0-140-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Jun  3 11:20:08 UTC 2025

  System load:             0.02
  Usage of /:              3.9% of 38.70GB
  Memory usage:            22%
  Swap usage:              0%
  Processes:               101
  Users logged in:         0
  IPv4 address for enp0s3: 10.0.2.15
  IPv4 address for enp0s9: 172.20.10.3
  IPv6 address for enp0s9: 2a00:20:51:7064:a00:27ff:fe67:32bb


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '24.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
==========================================================================
```

4 Switch to Root and Navigate to the Shared Vagrant Directory

```
vagrant@ubuntu-jammy:~$ sudo -i

root@ubuntu-jammy:~# cd /vagrant
==========================================================================
root@ubuntu-jammy:/vagrant# ls -la
total 16
drwxrwxr-x  1 vagrant vagrant 4096 May 22 17:22 .
drwxr-xr-x 20 root    root    4096 Jun  3 11:09 ..
drwxrwxr-x  1 vagrant vagrant 4096 May 22 17:37 .vagrant
-rw-rw-r--  1 vagrant vagrant  356 May 22 17:41 Vagrantfile
==========================================================================
```

5 Create Multiple Shell Script Files Inside the Synced Folder

```
root@ubuntu-jammy:/vagrant# touch file{1..10}.sh

root@ubuntu-jammy:/vagrant# ls -la
=============================================================================
total 16
drwxrwxr-x  1 vagrant vagrant 4096 Jun  3 11:22 .
drwxr-xr-x 20 root    root    4096 Jun  3 11:09 ..
drwxrwxr-x  1 vagrant vagrant 4096 May 22 17:37 .vagrant
-rw-rw-r--  1 vagrant vagrant  356 May 22 17:41 Vagrantfile
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file1.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file10.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file2.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file3.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file4.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file5.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file6.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file7.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file8.sh
-rw-r--r--  1 vagrant vagrant    0 Jun  3 11:22 file9.sh
=============================================================================
```

6 Clean Up Script Files Created Inside the Guest VM

```
root@ubuntu-jammy:/vagrant# rm file{1..10}.sh

root@ubuntu-jammy:/vagrant# ls -la
=============================================================================
total 16
drwxrwxr-x  1 vagrant vagrant 4096 Jun  3 11:24 .
drwxr-xr-x 20 root    root    4096 Jun  3 11:09 ..
drwxrwxr-x  1 vagrant vagrant 4096 May 22 17:37 .vagrant
-rw-rw-r--  1 vagrant vagrant  356 May 22 17:41 Vagrantfile
=============================================================================
```

7 Verify Global Vagrant Status and Local VM Running State

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant global-status
=============================================================================
id       name    provider   state   directory
-----------------------------------------------------------------------------
-----------------
de2e78f  default virtualbox running /home/tiago-paquete/Testdir/vagrant/
myvirtualmachines

The above shows information about all known Vagrant environments
```

```
on this machine. This data is cached and may not be completely
up-to-date (use "vagrant global-status --prune" to prune invalid
entries). To interact with any of the machines, you can go to that
directory and run Vagrant, or you can use the ID directly with
Vagrant commands from any directory. For example:
"vagrant destroy 1a2b3c4d”
==========================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant status
==========================================================================
Current machine states:

default                   running (virtualbox)

The VM is running. To stop this VM, you can run `vagrant halt` to
shut it down forcefully, or you can run `vagrant suspend` to simply
suspend the virtual machine. In either case, to restart it again,
simply run `vagrant up`.
==========================================================================
```

8 Gracefully Shut Down the Virtual Machine with vagrant halt

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant halt
==========================================================================
==> default: Attempting graceful shutdown of VM…
==========================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant status
==========================================================================
Current machine states:

default                   poweroff (virtualbox)

The VM is powered off. To restart the VM, simply run `vagrant up`
==========================================================================
```

9 Destroy the Virtual Machine and Remove its Resources

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant destroy –force
==========================================================================
==> default: Destroying VM and associated drives…
==========================================================================
```

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant status
==========================================================================
Current machine states:

default                   not created (virtualbox)

The environment has not yet been created. Run `vagrant up` to
create the environment. If a machine is not created, only the
default provider will be shown. So if a provider is not listed,
then the machine is not created for that environment.
==========================================================================
```

10 Prune Stale Vagrant Environment Entries

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant global-status prune
==========================================================================
id       name   provider state  directory
--------------------------------------------------------------------------
There are no active Vagrant environments on this computer! Or,
you haven't destroyed and recreated Vagrant environments that were
started with an older version of Vagrant.
==========================================================================
```

🔍 `vagrant global-status`

This part **lists all Vagrant environments** that have been created on your system, regardless of which directory you're currently in. For each environment, it shows:

- **id**: a unique identifier for the VM
- **name**: the name of the VM
- **provider**: e.g., virtualbox, vmware, etc.
- **state**: running, poweroff, saved, etc.
- **directory**: the path where the Vagrantfile is located

🧹 `--prune`

This option **removes entries** for Vagrant environments that **no longer exist**, i.e., those where:

- The Vagrantfile directory was deleted.
- The `.vagrant` folder is gone or invalid.
- The virtual machine was deleted outside of Vagrant (e.g., directly via VirtualBox UI).

11 Manually Delete the Existing Vagrantfile

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ ls
Vagrantfile

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ rm Vagrantfile

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ ls -la
==========================================================================
total 12
drwxrwxr-x 3 tiago-paquete tiago-paquete 4096 Jun  3 13:31 .
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 19:37 .vagrant
==========================================================================
```

12 Remove Vagrant Metadata Directory (.vagrant)

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ rm -rf .vagrant

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ ls -la
==========================================================================
total 8
drwxrwxr-x 2 tiago-paquete tiago-paquete 4096 Jun  3 13:32 .
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
==========================================================================
```

13 Re-initialize a Fresh Vagrant Environment (`vagrant init`)

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant init ubuntu/jammy64
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
==========================================================================
A `Vagrantfile` has been placed in this directory. You are now
ready to `vagrant up` your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
`vagrantup.com` for more information on using Vagrant.
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
==========================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant up
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
==========================================================================
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Importing base box 'ubuntu/jammy64'...

    default: /home/tiago-paquete/Testdir/vagrant/myvirtualmachines => /
vagrant
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
==========================================================================
```

14 Check Vagrantfile

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ ls -la
=========================================================================
total 16
drwxrwxr-x 3 tiago-paquete tiago-paquete 4096 Jun  3 13:38 .
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 May 22 15:05 ..
drwxrwxr-x 4 tiago-paquete tiago-paquete 4096 Jun  3 13:38 .vagrant
-rw-rw-r-- 1 tiago-paquete tiago-paquete 3388 Jun  3 13:38 Vagrantfile
=========================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ cat Vagrantfile
=========================================================================
# -*- mode: ruby -*-
# vi: set ft=ruby :

# All Vagrant configuration is done below. The "2" in Vagrant.configure
# configures the configuration version (we support older styles for
# backwards compatibility). Please don't change it unless you know what
# you're doing.
Vagrant.configure("2") do |config|
  # The most common configuration options are documented and commented
below.
  # For a complete reference, please see the online documentation at
  # https://docs.vagrantup.com.

  # Every Vagrant development environment requires a box. You can search
for
  # boxes at https://vagrantcloud.com/search.
  config.vm.box = "ubuntu/jammy64"

  # Disable automatic box update checking. If you disable this, then
  # boxes will only be checked for updates when the user runs
  # `vagrant box outdated`. This is not recommended.
  # config.vm.box_check_update = false

  # Create a forwarded port mapping which allows access to a specific
port
  # within the machine from a port on the host machine. In the example
below,
  # accessing "localhost:8080" will access port 80 on the guest machine.
  # NOTE: This will enable public access to the opened port
  # config.vm.network "forwarded_port", guest: 80, host: 8080

  # Create a forwarded port mapping which allows access to a specific
port
  # within the machine from a port on the host machine and only allow
access
  # via 127.0.0.1 to disable public access
  # config.vm.network "forwarded_port", guest: 80, host: 8080, host_ip:
"127.0.0.1"

  # Create a private network, which allows host-only access to the
machine
  # using a specific IP.
```

```
  # config.vm.network "private_network", ip: "192.168.33.10"

  # Create a public network, which generally matched to bridged network.
  # Bridged networks make the machine appear as another physical device
on
  # your network.
  # config.vm.network "public_network"

  # Share an additional folder to the guest VM. The first argument is
  # the path on the host to the actual folder. The second argument is
  # the path on the guest to mount the folder. And the optional third
  # argument is a set of non-required options.
  # config.vm.synced_folder "../data", "/vagrant_data"

  # Disable the default share of the current code directory. Doing this
  # provides improved isolation between the vagrant box and your host
  # by making sure your Vagrantfile isn't accessible to the vagrant box.
  # If you use this you may want to enable additional shared subfolders
as
  # shown above.
  # config.vm.synced_folder ".", "/vagrant", disabled: true

  # Provider-specific configuration so you can fine-tune various
  # backing providers for Vagrant. These expose provider-specific
options.
  # Example for VirtualBox:
  #
  # config.vm.provider "virtualbox" do |vb|
  #   # Display the VirtualBox GUI when booting the machine
  #   vb.gui = true
  #
  #   # Customize the amount of memory on the VM:
  #   vb.memory = "1024"
  # end
  #
  # View the documentation for the provider you are using for more
  # information on available options.

  # Enable provisioning with a shell script. Additional provisioners
such as
  # Ansible, Chef, Docker, Puppet and Salt are also available. Please
see the
  # documentation for more information about their specific syntax and
use.
  # config.vm.provision "shell", inline: <<-SHELL
  #   apt-get update
  #   apt-get install -y apache2
  # SHELL
end
=====================================================================
```

15 Verify Guest Login and System Information via `vagrant ssh`

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant ssh
==============================================================================
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 5.15.0-140-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Jun  3 11:41:48 UTC 2025

  System load:  0.05              Processes:               107
  Usage of /:   3.9% of 38.70GB   Users logged in:         0
  Memory usage: 21%               IPv4 address for enp0s3: 10.0.2.15
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '24.04.2 LTS' available.
==============================================================================
```

16 Review the Custom `Vagrantfile` (Networking, Resources, Synced Folder)

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ cat Vagrantfile
==============================================================================
Vagrant.configure("2") do |config|
  # Set the base box to Ubuntu 22.04 (Jammy Jellyfish) 64-bit
  config.vm.box = "ubuntu/jammy64"

  # Configure a private network with a static IP address
  config.vm.network "private_network", ip: "192.168.56.10"

  # Configure a public (bridged) network using the specified network
interface
  config.vm.network "public_network", bridge: "wlp0s20f3"

  # Set VirtualBox provider-specific options
  config.vm.provider "virtualbox" do |vb|
    # Allocate 1024 MB of RAM to the VM
```

```
    vb.memory = 1024
    # Allocate 2 CPU cores to the VM
    vb.cpus = 2
  end

  # Sync the local ./scripts folder to /opt/scripts in the VM
  config.vm.synced_folder "./scripts", "/opt/scripts"
end
```

17 Reload the VM to Apply Updated Configuration (`vagrant reload`)

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant reload
==> default: Attempting graceful shutdown of VM...

…

==> default: Configuring and enabling network interfaces...
==> default: Mounting shared folders...
    default: /home/tiago-paquete/Testdir/vagrant/myvirtualmachines => /vagrant
    default: /home/tiago-paquete/Testdir/vagrant/myvirtualmachines/scripts => /opt/scripts
==> default: Machine already provisioned. Run `vagrant provision` or use the `--provision`
==> default: flag to force provisioning. Provisioners marked to run always will still run.
```

18 Create/delete new files in /scripts

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ cd scripts

@Ubuntu:~/Testdir/vagrant/myvirtualmachines/scripts$ ls

@Ubuntu:~/Testdir/vagrant/myvirtualmachines/scripts$ touch
file{1..10}

@Ubuntu:~/Testdir/vagrant/myvirtualmachines/scripts$ ls
file1  file10  file2  file3  file4  file5  file6  file7  file8  file9

@Ubuntu:~/Testdir/vagrant/myvirtualmachines/scripts$ rm file{1..10}

@Ubuntu:~/Testdir/vagrant/myvirtualmachines/scripts$ ls
```

19 Gracefully Halt the Virtual Machine (`vagrant halt`)

```
@Ubuntu:~/Testdir/vagrant/myvirtualmachines/scripts$ cd ..

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant status
==========================================================================
Current machine states:

default                   running (virtualbox)

The VM is running. To stop this VM, you can run `vagrant halt` to
shut it down forcefully, or you can run `vagrant suspend` to simply
suspend the virtual machine. In either case, to restart it again,
simply run `vagrant up`.
==========================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant halt
==========================================================================
==> default: Attempting graceful shutdown of VM…
==========================================================================

@Ubuntu:~/Testdir/vagrant/myvirtualmachines$ vagrant status
==========================================================================
Current machine states:

default                   poweroff (virtualbox)

The VM is powered off. To restart the VM, simply run `vagrant up`
==========================================================================
```