Cloud Computing Assignment 1

Objective:

The aim is to gain practical experience with setting up a QEMU/KVM hypervisor and performing basic virtualization tasks. This assignment will help understand the fundamentals of virtualization, virtual machine (VM) creation, and management using QEMU/KVM.

Task 1: Installation and Configuration

Objectives:

1. Hypervisor Installation:

   - Install the QEMU/KVM hypervisor on a Linux machine.

   - Verify that the KVM modules are loaded and functioning correctly.

2. Networking Setup:

   - Configure a bridged network on the host to allow communication between virtual machines and external networks.

   - Document all steps and commands used.

3. Management Tools:

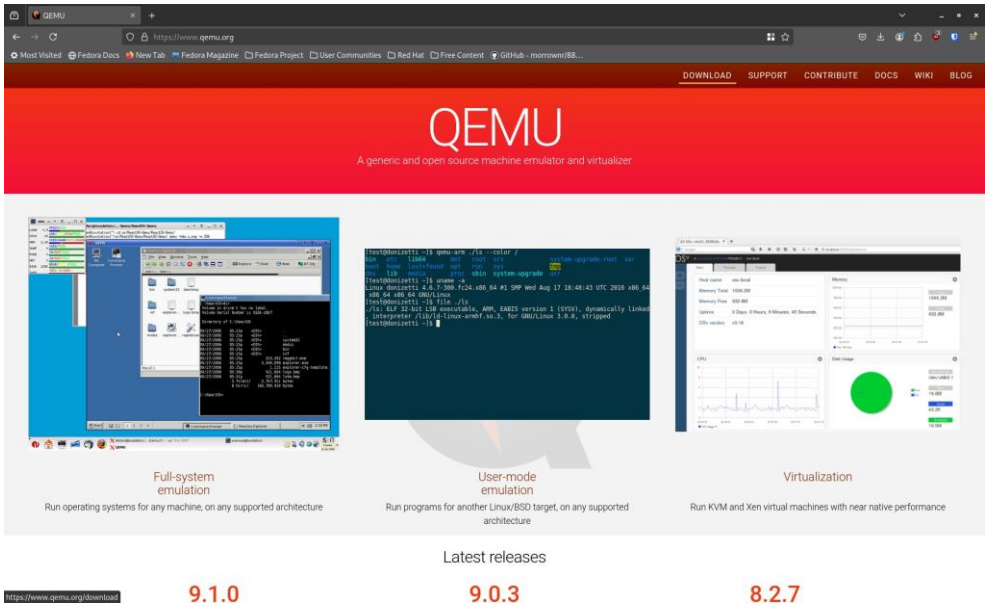
   - Install and configure `virt-manager` or `virsh` for VM management.

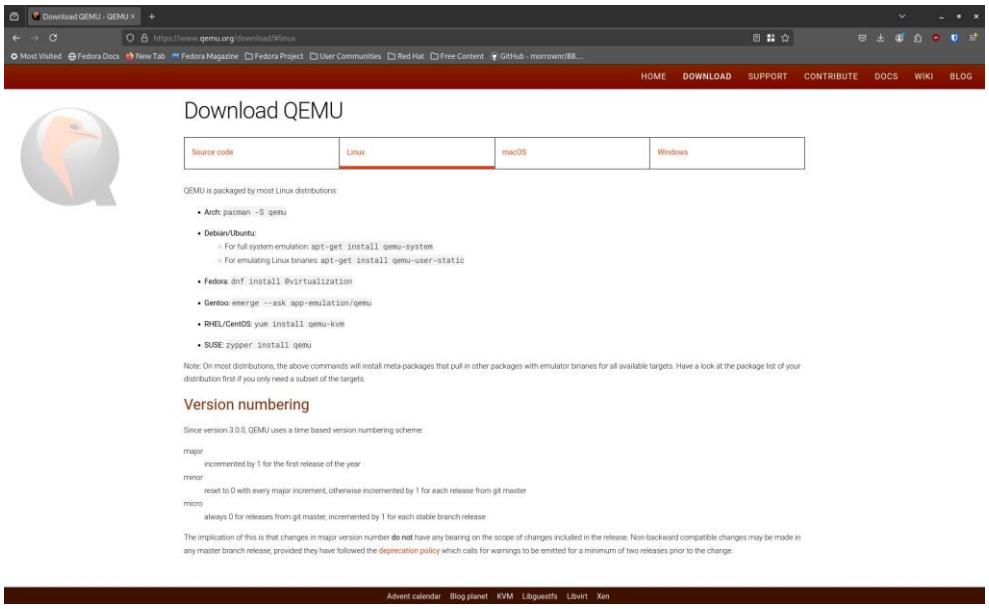
   - Get familiar with basic commands and GUI options for managing VMs.

Steps:

1. Hypervisor Installation:

   - Download the appropriate QEMU/KVM version for the Linux distribution.



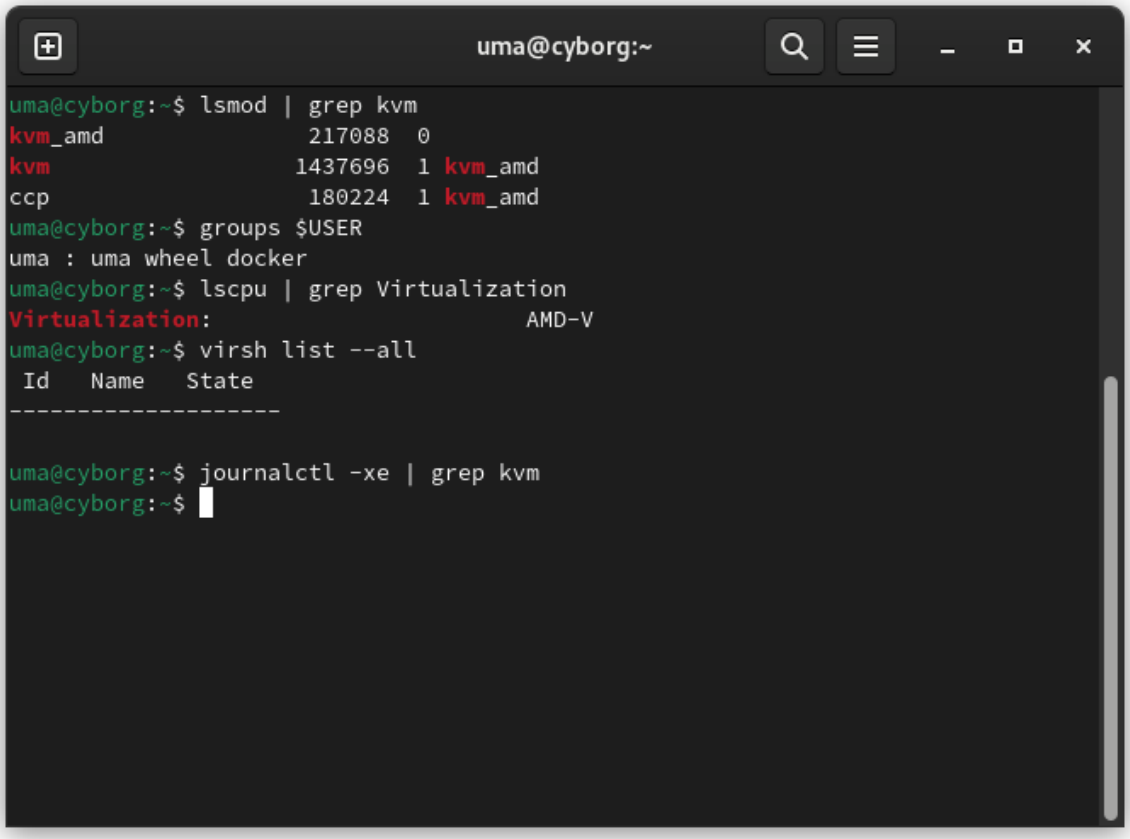


- Use a package manager to install (e.g., `dnf install @virtualization` for Fedora).

```
uma@cyborg:~$ sudo dnf install @virtualization
```

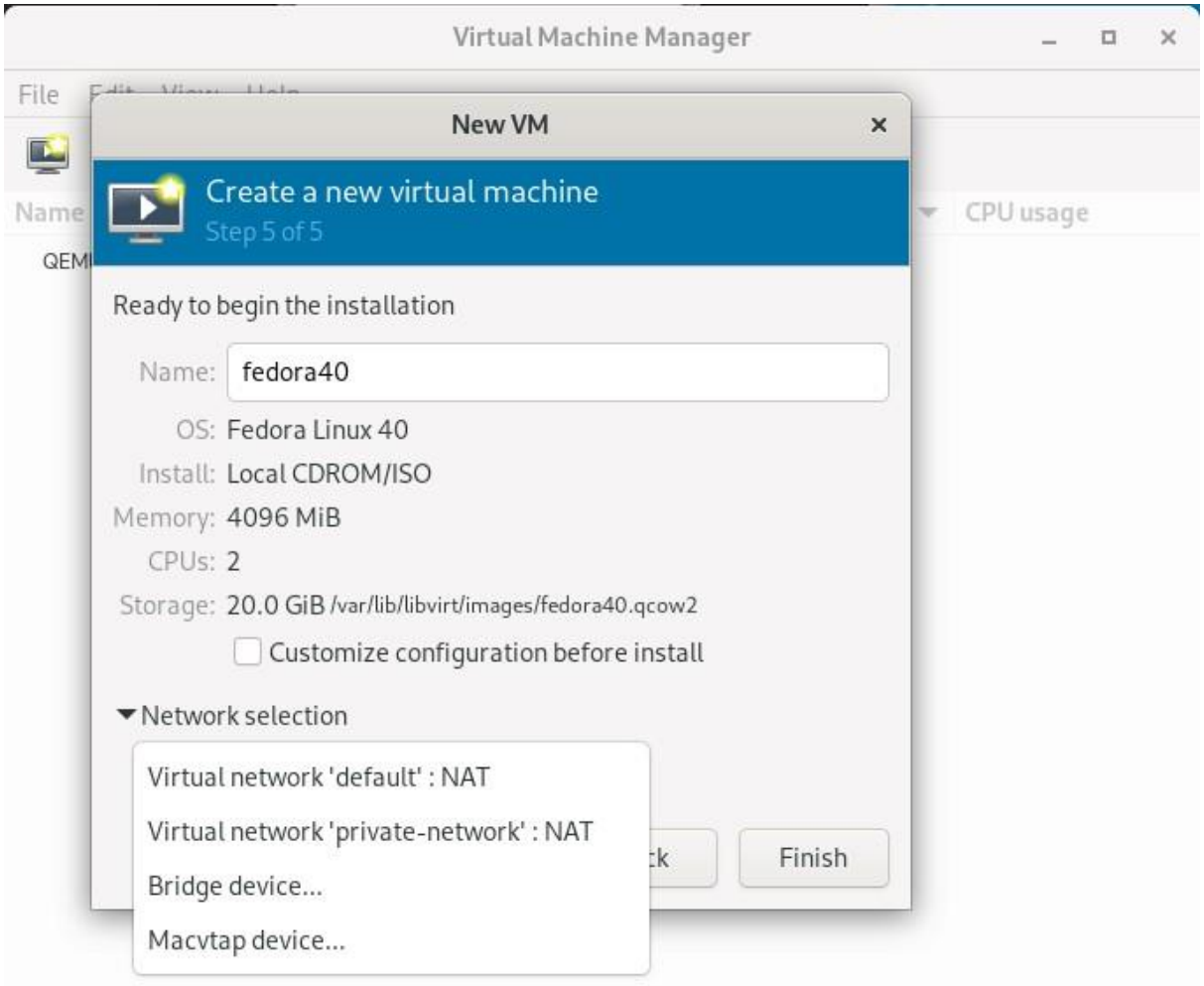
```
  Verifying        : virt-manager-4.1.0-3.fc     7/12
  Verifying        : virt-manager-common-4.1     8/12
  Verifying        : virt-viewer-11.0-7.fc39     9/12
  Verifying        : kf5-filesystem-5.116.0-    10/12
  Verifying        : libisoburn-1.5.6-5.fc39    11/12
  Verifying        : xorriso-1.5.6-5.fc39.x8    12/12

Installed:
  kde-filesystem-4-70.fc39.x86_64
  kf5-filesystem-5.116.0-1.fc39.x86_64
  libburn-1.5.6-2.fc39.x86_64
  libisoburn-1.5.6-5.fc39.x86_64
  libisofs-1.5.6-2.fc39.x86_64
  python3-libvirt-9.7.0-1.fc39.x86_64
  python3-libxml2-2.10.4-3.fc39.x86_64
  virt-install-4.1.0-3.fc39.noarch
  virt-manager-4.1.0-3.fc39.noarch
  virt-manager-common-4.1.0-3.fc39.noarch
  virt-viewer-11.0-7.fc39.x86_64
  xorriso-1.5.6-5.fc39.x86_64

Complete!
uma@cyborg:~$
```

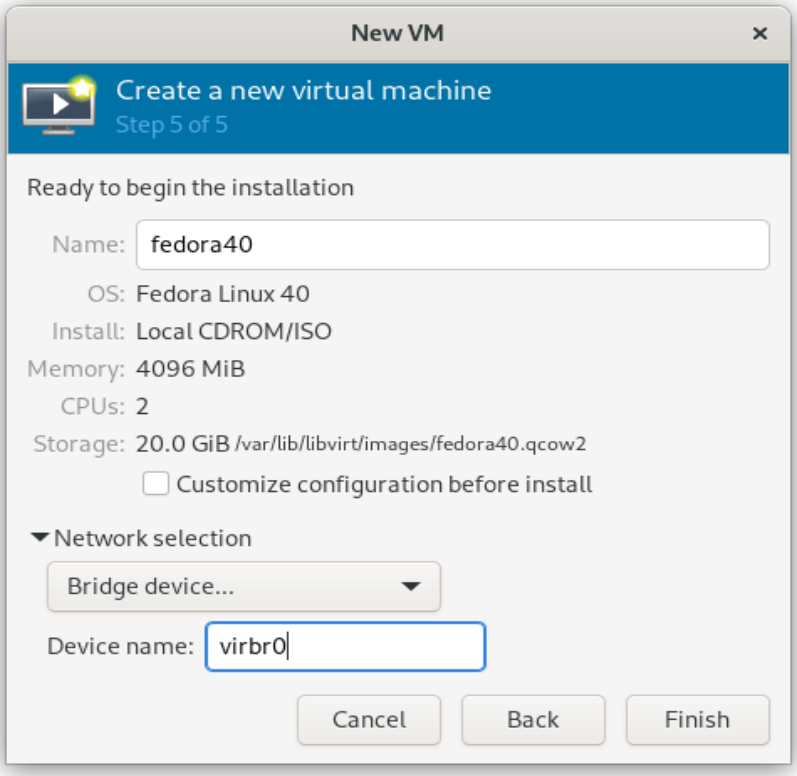
- Verify installation by checking the status of KVM modules (`lsmod | grep kvm`).



2. Networking Setup:

- Create a bridge network interface (`br0`) using network tools like `nmcli` or editing configuration files directly.


Virtual Machine Manager
File
New VM
Create a new virtual machine
Step 5 of 5
Name
CPU usage
Ready to begin the installation
Name: fedora40
OS: Fedora Linux 40
Install: Local CDROM/ISO
Memory: 4096 MiB
CPUs: 2
Storage: 20.0 GiB /var/lib/libvirt/images/fedora40.qcow2
Customize configuration before install
Network selection
Virtual network 'default' : NAT
Virtual network 'private-network' : NAT
Bridge device...
Macvtap device...
Finish


New VM
Create a new virtual machine
Step 5 of 5
Ready to begin the installation
Name: fedora40
OS: Fedora Linux 40
Install: Local CDROM/ISO
Memory: 4096 MiB
CPUs: 2
Storage: 20.0 GiB /var/lib/libvirt/images/fedora40.qcow2
Customize configuration before install
Network selection
Bridge device...
Device name: virbr0
Cancel
Back
Finish

- Verify that the bridge is functioning with `virsh domiflist <vm_name>` to check if VMs are using the bridged interface.

```
uma@cyborg:~$ sudo virsh domiflist fedora40
[sudo] password for uma:
 Interface   Type     Source   Model    MAC
-------------------------------------------------------
 vnet0       bridge   virbr0   virtio

uma@cyborg:~$
```

```
uma@cyborg:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp6s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether e0:d5:5e:a7:89:38 brd ff:ff:ff:ff:ff:ff
3: wlp13s0f3u1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 7c:c2:c6:11:e0:43 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.103/24 brd 192.168.0.255 scope global dynamic noprefixroute wlp13s0f3u1
       valid_lft 3957sec preferred_lft 3957sec
    inet6 fe80::ba3a:a946:d1a6:53c2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
4: virbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 52:54:00:41:fa:c6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.124.1/24 brd 192.168.124.255 scope global virbr0
       valid_lft forever preferred_lft forever
19: virbr1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:f6:a8:86 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/24 brd 10.0.0.255 scope global virbr1
       valid_lft forever preferred_lft forever
29: vnet0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master virbr0 state UNKNOWN group default
 qlen 1000
    link/ether fe:54:00:2f:cf:d3 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::fc54:ff:fe2f:cfd3/64 scope link proto kernel_ll
       valid_lft forever preferred_lft forever
uma@cyborg:~$
```

3. Management Tools:

- Install `virt-manager` for GUI-based management (`sudo dnf install virt-manager`).

- Get familiar with starting, stopping, and configuring VMs via both `virt-manager` and the command line (`virsh`).

Task 2: Virtual Machine Creation and Management

Objectives:

1. Create a Virtual Machine:

   - Use `virt-manager` or `virsh` to create a VM running a Linux distribution (e.g., Ubuntu).

   - Allocate appropriate resources (CPU, memory, disk space) based on the host's capacity.

2. Guest OS Installation:

   - Install the guest OS from an ISO image.

   - Document key installation steps with screenshots.

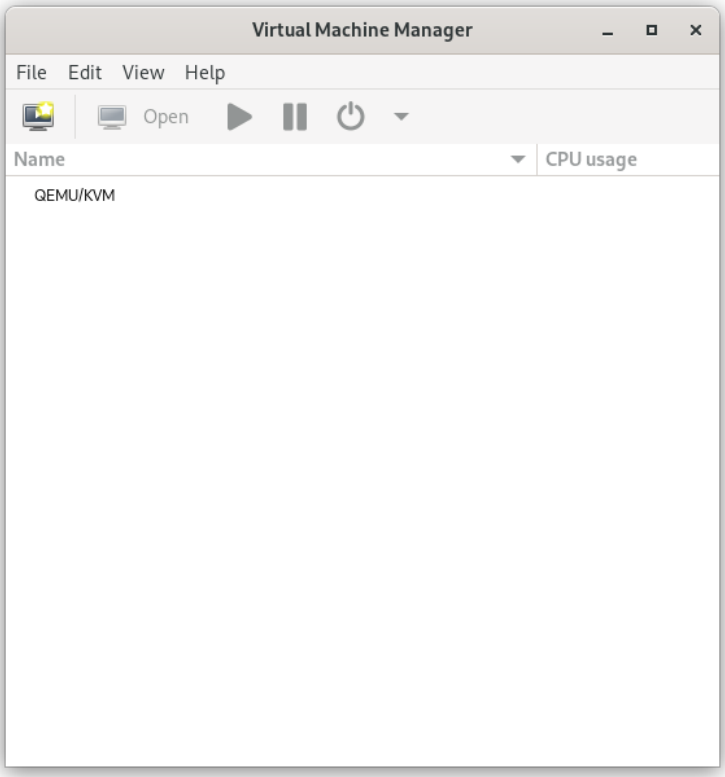
3. Basic VM Operations:

   - Perform basic operations: start, stop, pause, and resume the VM using both CLI (`virsh`) and GUI (`virt-manager`).

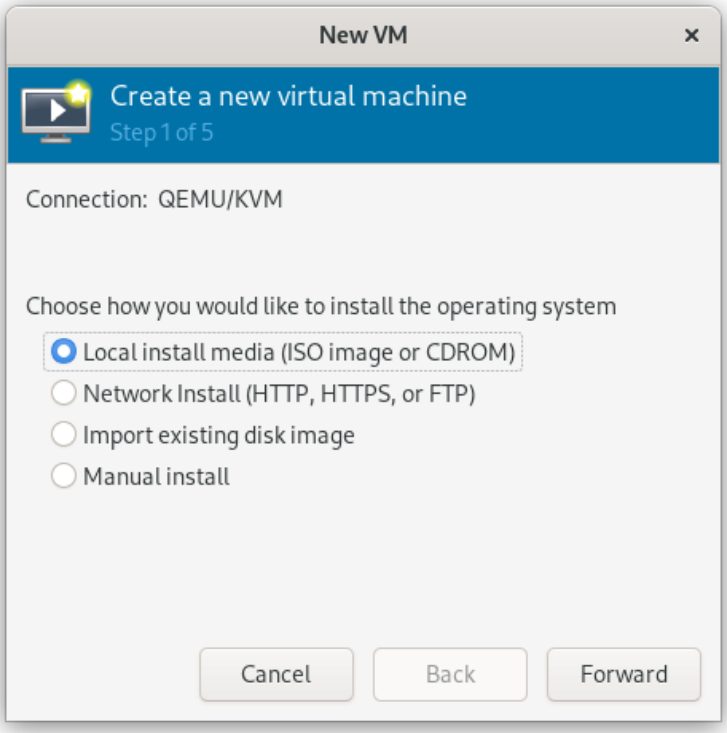
Steps:

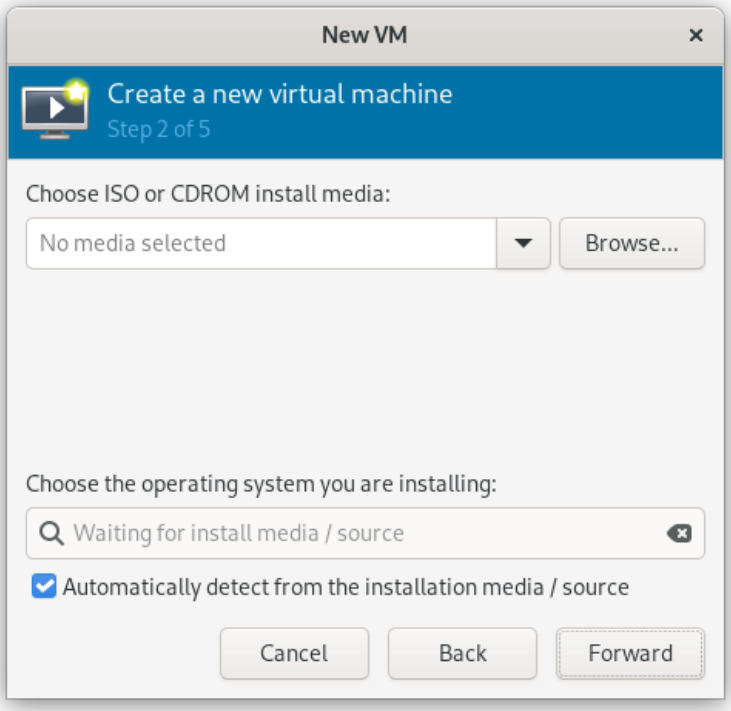
1. Creating the VM:

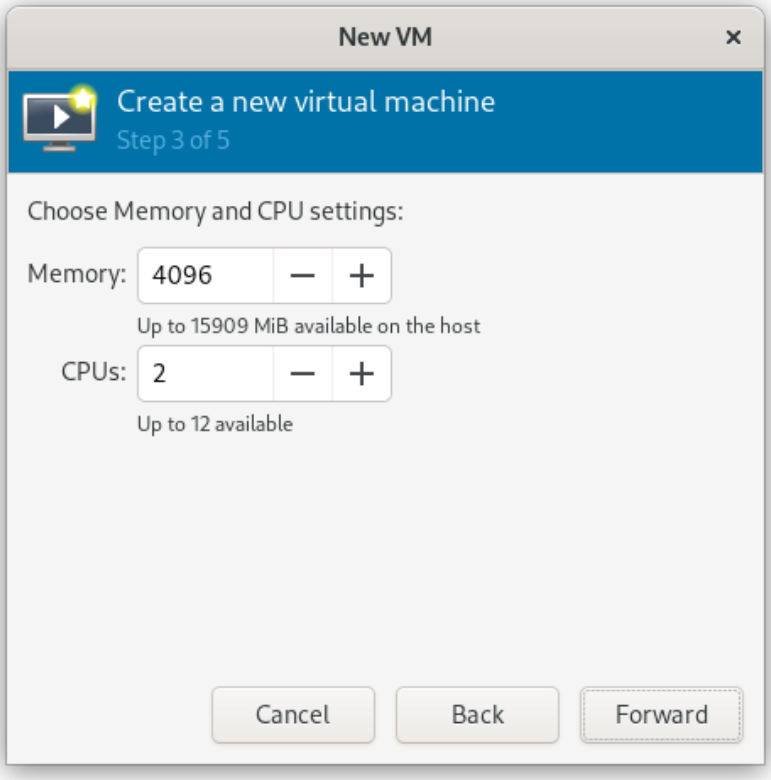
   - Open `virt-manager` and use the "Create new VM" wizard.



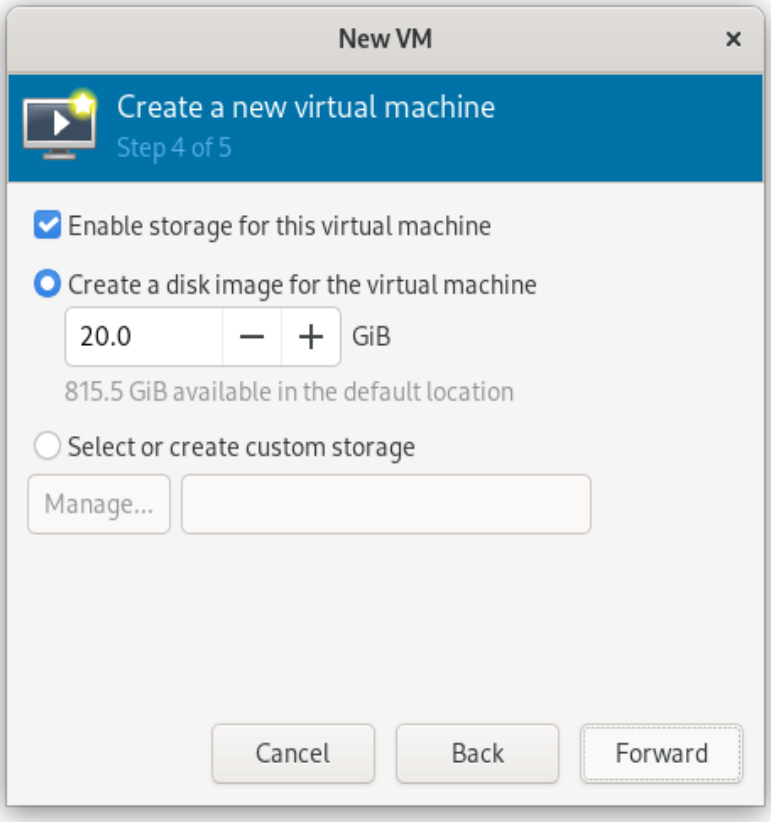
- Choose "Local install media" and provide the path to the downloaded ISO.

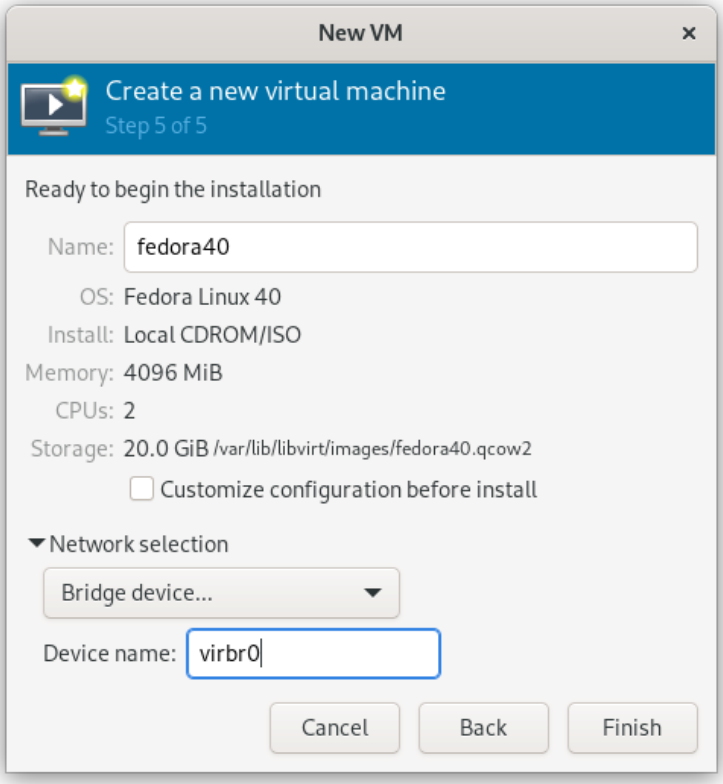

New VM
Create a new virtual machine
Step 1 of 5
Connection: QEMU/KVM
Choose how you would like to install the operating system
Local install media (ISO image or CDROM)
Network Install (HTTP, HTTPS, or FTP)
Import existing disk image
Manual install
Cancel
Back
Forward


New VM
Create a new virtual machine
Step 2 of 5
Choose ISO or CDROM install media:
No media selected
Browse...
Choose the operating system you are installing:
Waiting for install media / source
Automatically detect from the installation media / source
Cancel
Back
Forward



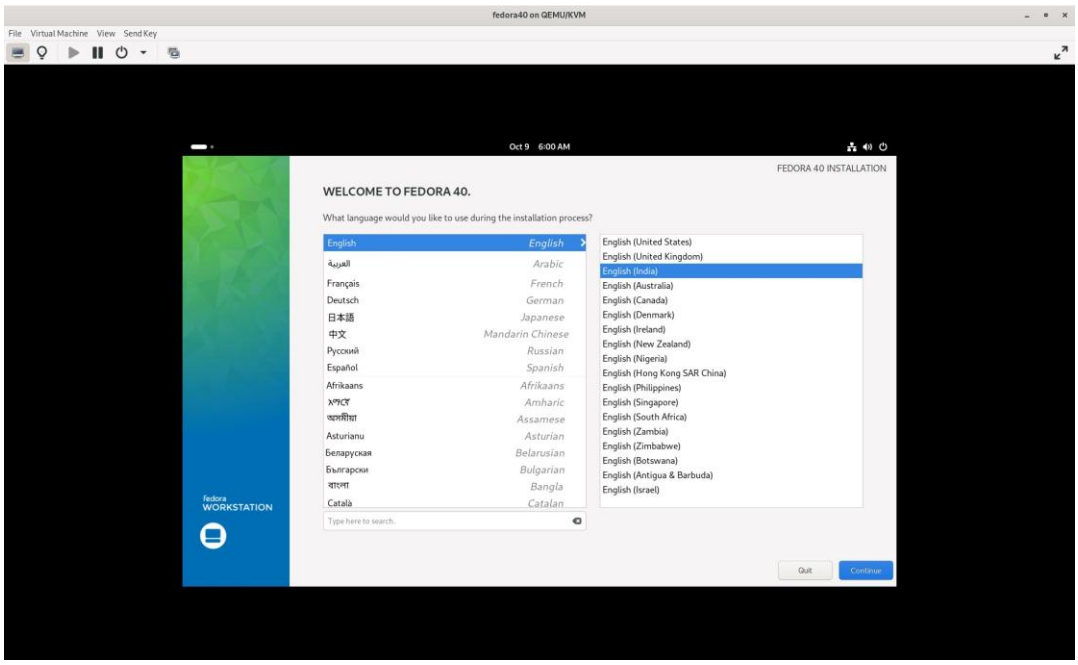
- Allocate resources (e.g., 2 CPUs, 4GB RAM, 20GB disk).

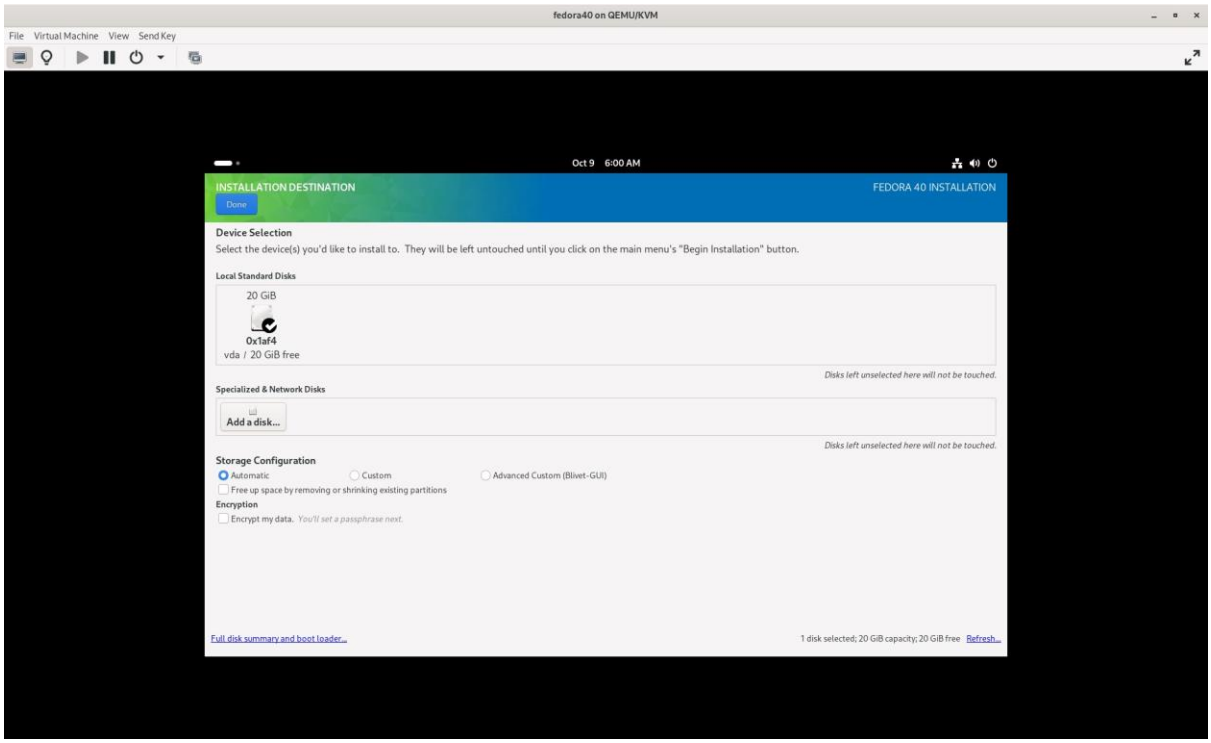


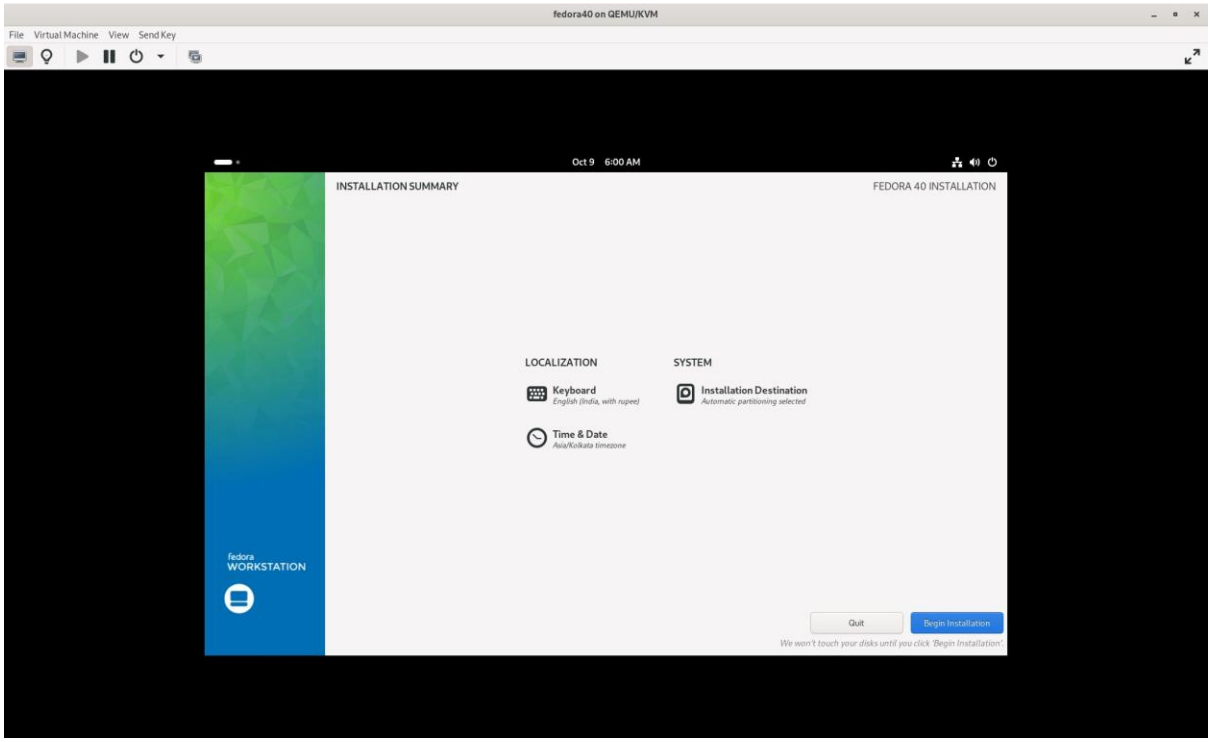


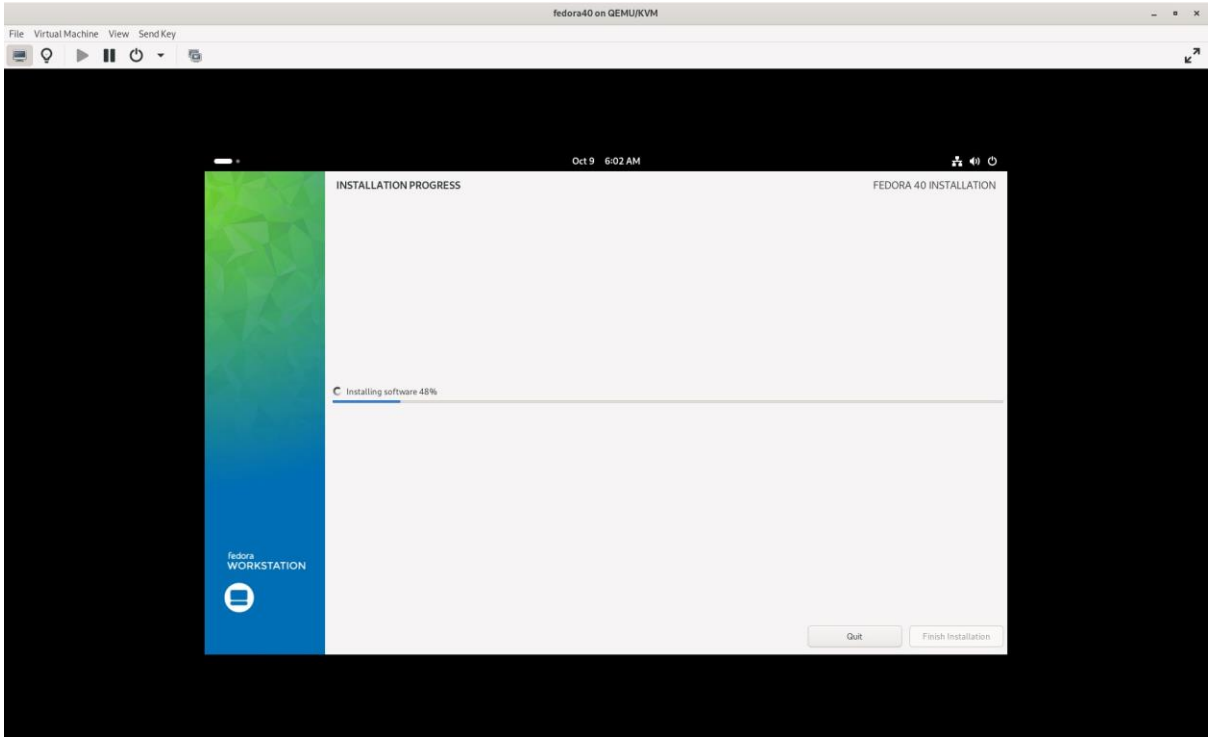
2. Installing the Guest OS:

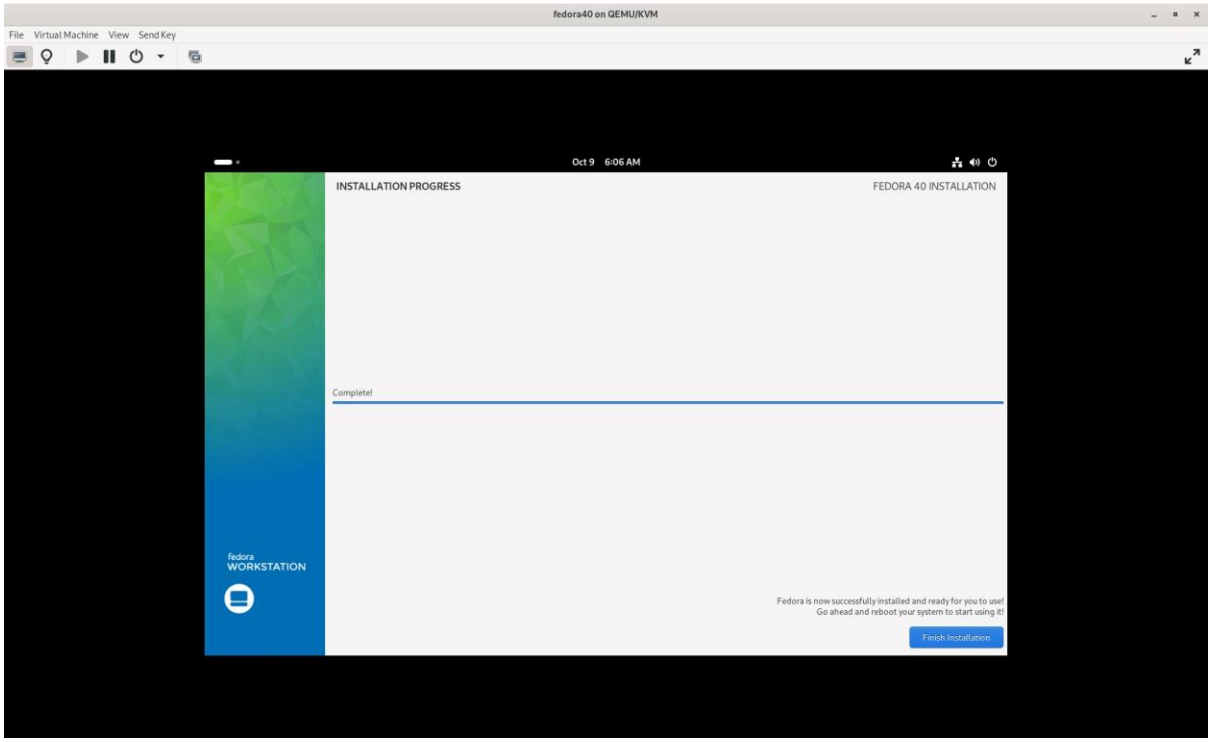
- Boot from the ISO and follow the on-screen instructions to install the chosen O




fedora40 on QEMU/KVM
File Virtual Machine View Send Key
Oct 9 6:00 AM
INSTALLATION DESTINATION
Done
FEDORA 40 INSTALLATION
Device Selection
Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.
Local Standard Disks
20 GiB
0x1af4
vda / 20 GiB free
Disks left unselected here will not be touched.
Specialized & Network Disks
Add a disk...
Disks left unselected here will not be touched.
Storage Configuration
Automatic
Custom
Advanced Custom (Blivet-GUI)
Free up space by removing or shrinking existing partitions
Encryption
Encrypt my data. You'll set a passphrase next.
Full disk summary and boot loader...
1 disk selected; 20 GiB capacity; 20 GiB free
Refresh...


fedora40 on QEMU/KVM
File Virtual Machine View Send Key
Oct 9 6:00 AM
INSTALLATION SUMMARY
FEDORA 40 INSTALLATION
LOCALIZATION
Keyboard
English (India, with rupee)
Time & Date
Asia/Kolkata timezone
SYSTEM
Installation Destination
Automatic partitioning selected
fedora
WORKSTATION
Quit
Begin Installation
We won't touch your disks until you click 'Begin Installation'



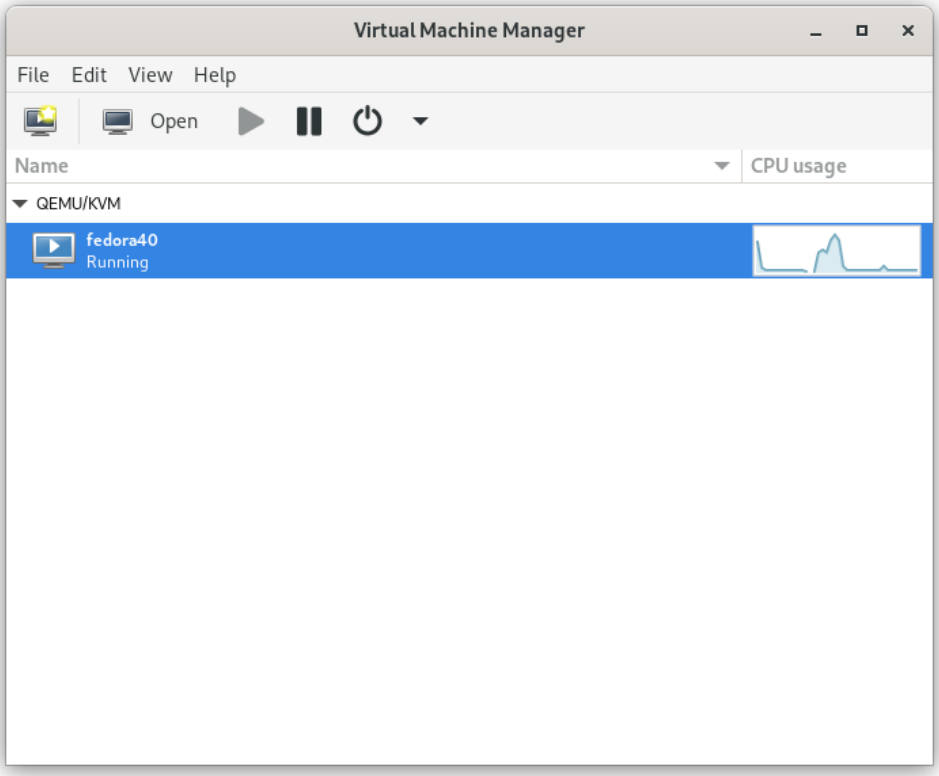


3. Basic VM Operations:

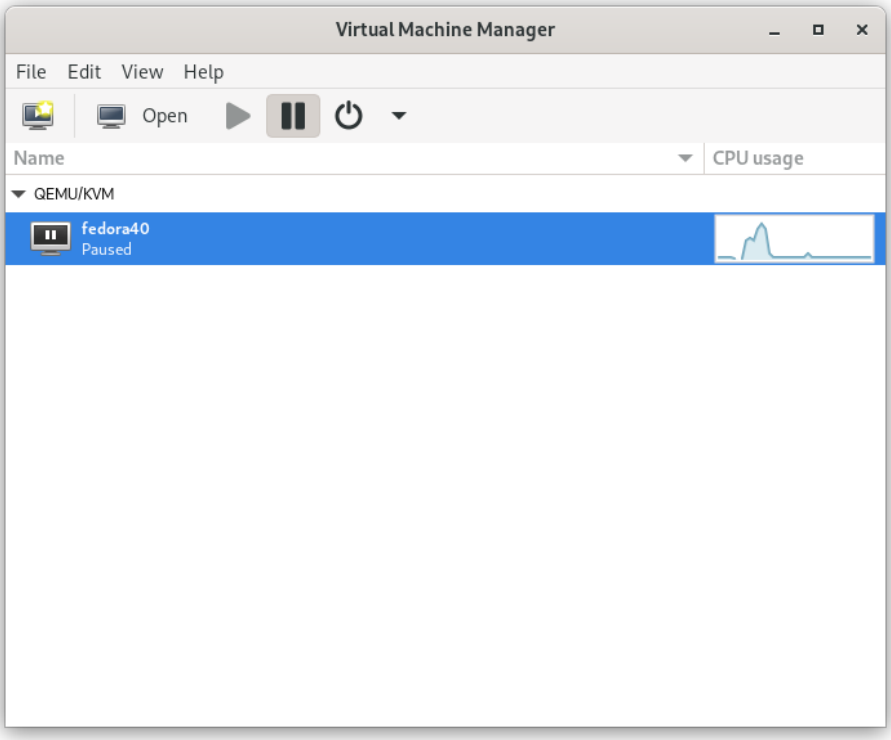
- Use `virsh` commands like `virsh start <vm_name>`, `virsh suspend <vm_name>`, and `virsh resume <vm_name>`.

```
uma@cyborg:~$ sudo virsh list --all
 Id   Name       State
-------------------------
 4    fedora40   running

uma@cyborg:~$ sudo virsh suspend 4
Domain '4' suspended

uma@cyborg:~$ sudo virsh resume 4
Domain '4' resumed

uma@cyborg:~$ sudo virsh shutdown 4
Domain '4' is being shutdown

uma@cyborg:~$ sudo virsh start fedora40
Domain 'fedora40' started

uma@cyborg:~$
```

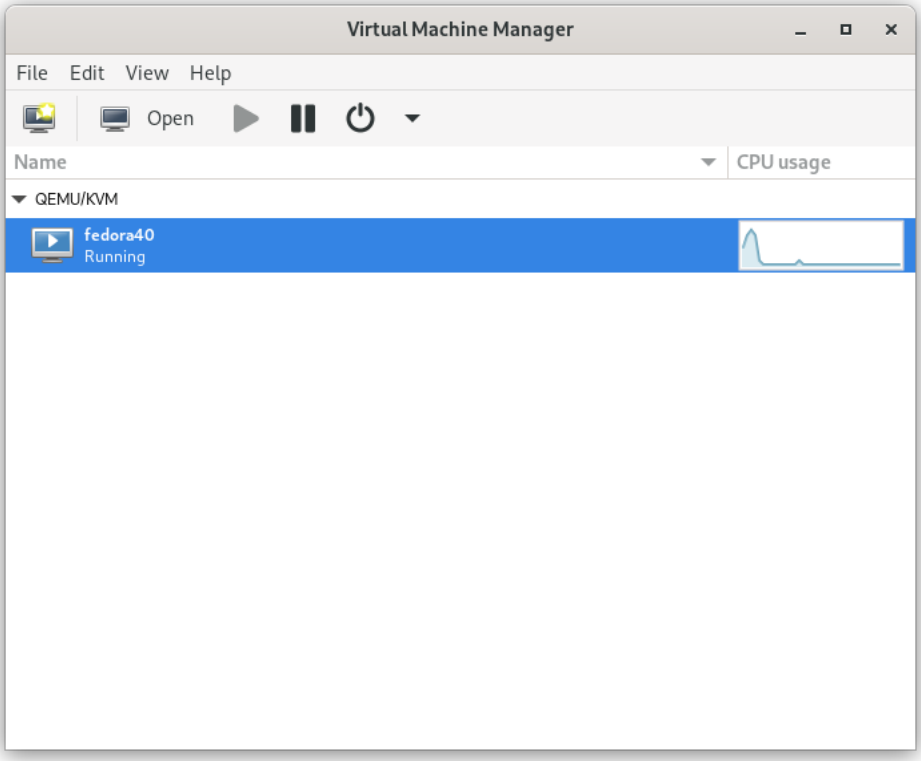
- Familiarize with equivalent GUI actions in `virt-manager`.



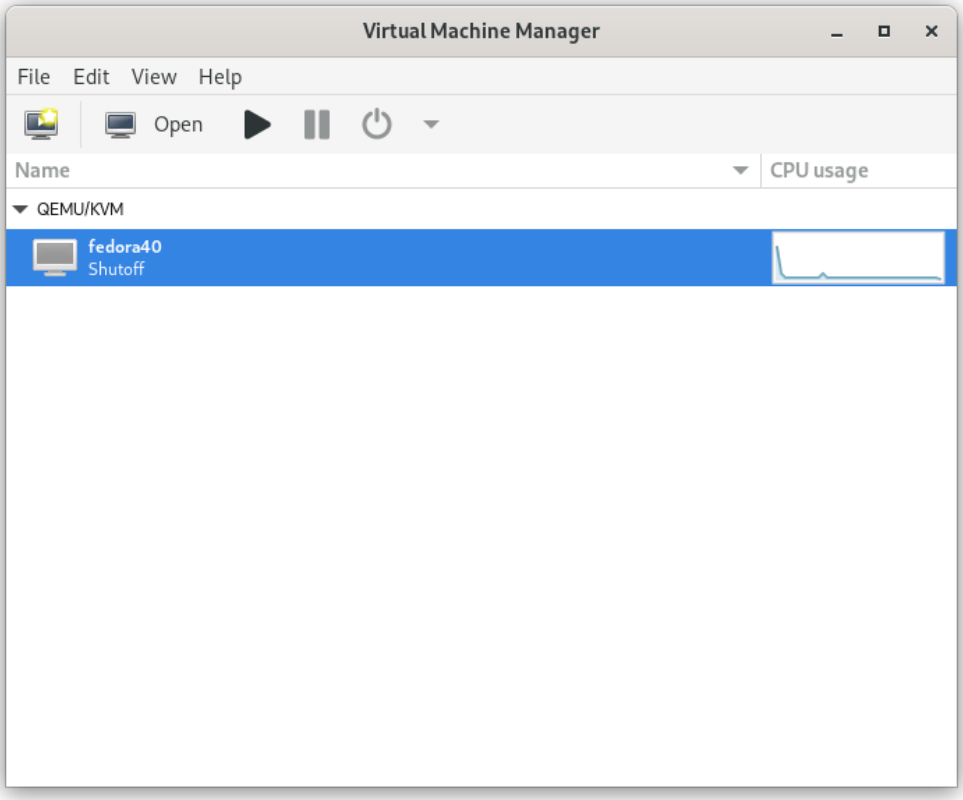
Paused:


Virtual Machine Manager
File
Edit
View
Help
Open
Name
CPU usage
QEMU/KVM
fedora40
Paused

Resume:


Virtual Machine Manager
File
Edit
View
Help
Open
Name
CPU usage
QEMU/KVM
fedora40
Running

Shutdown:



Task 3: Snapshot and Cloning

Objectives:

1. Snapshot Management:

   - Create a snapshot of a running VM.

   - Make changes (e.g., install software) and revert to the snapshot.

   - Discuss the advantages of snapshots.

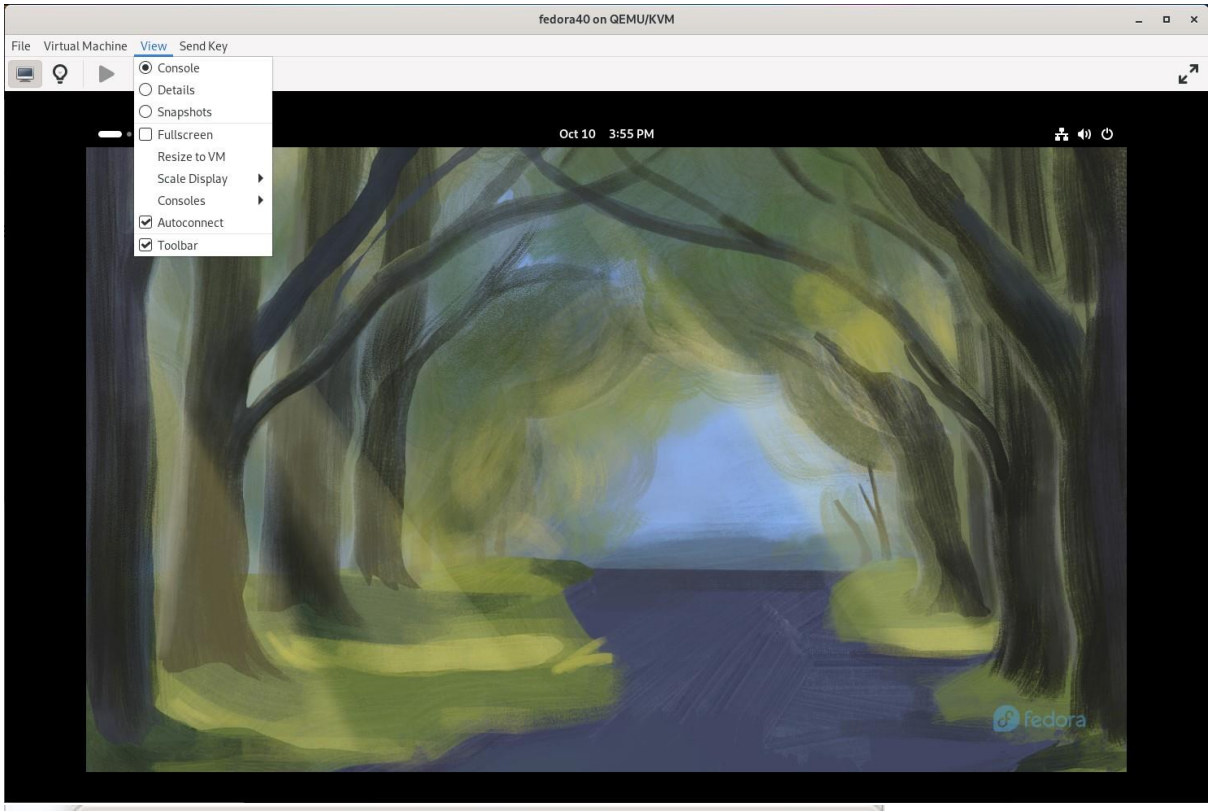
2. VM Cloning:

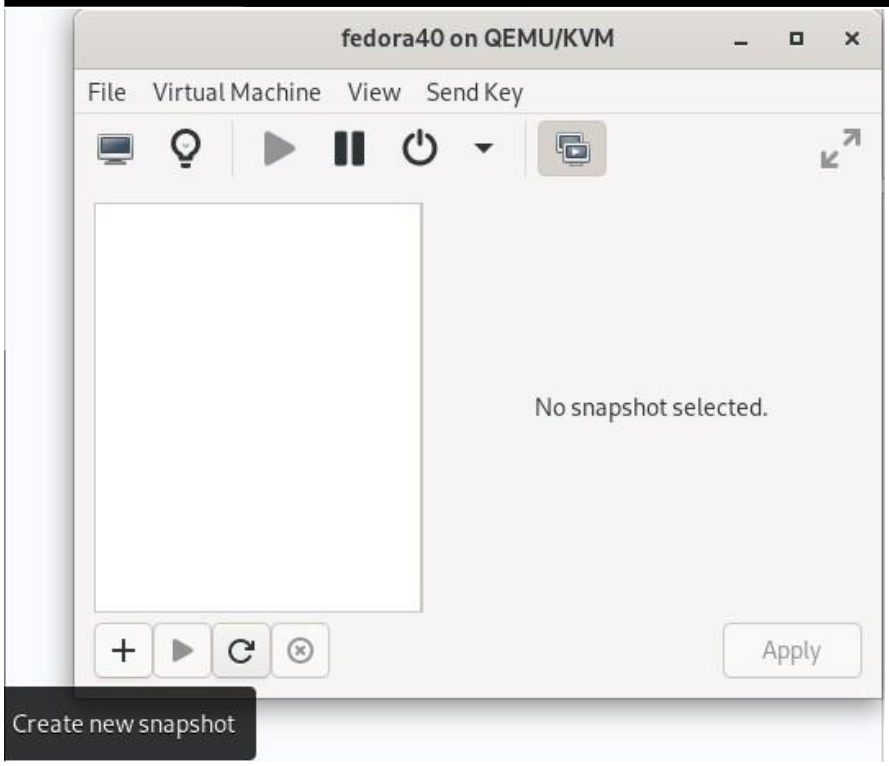
   - Clone the VM and modify the hostname and IP address to avoid conflicts.

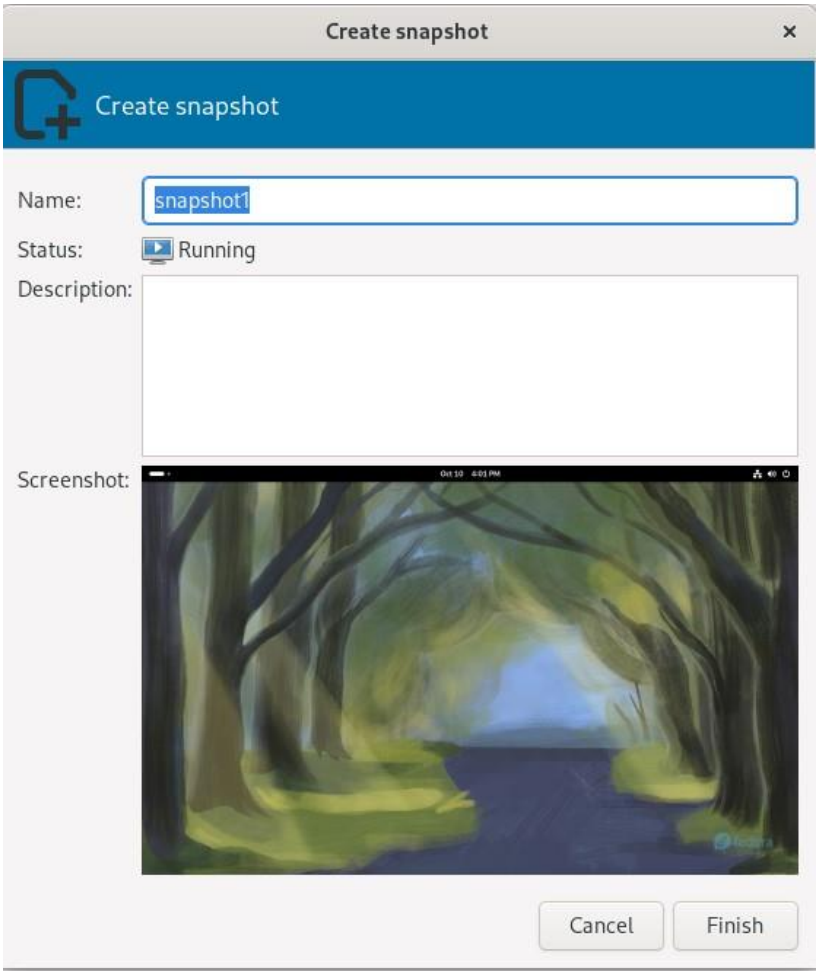
   - Confirm the cloned VM functions as expected.

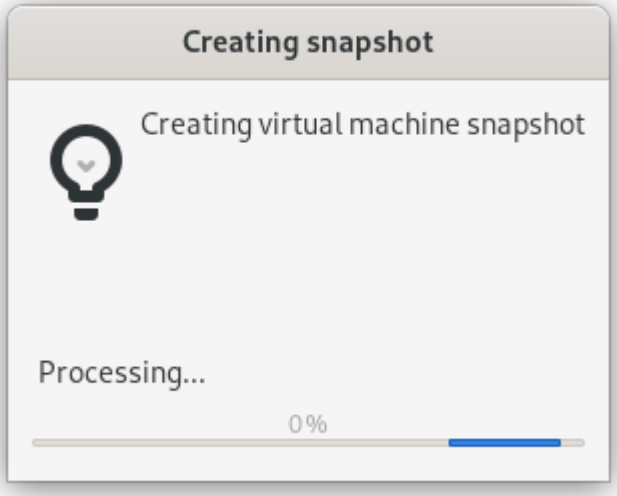
Steps:

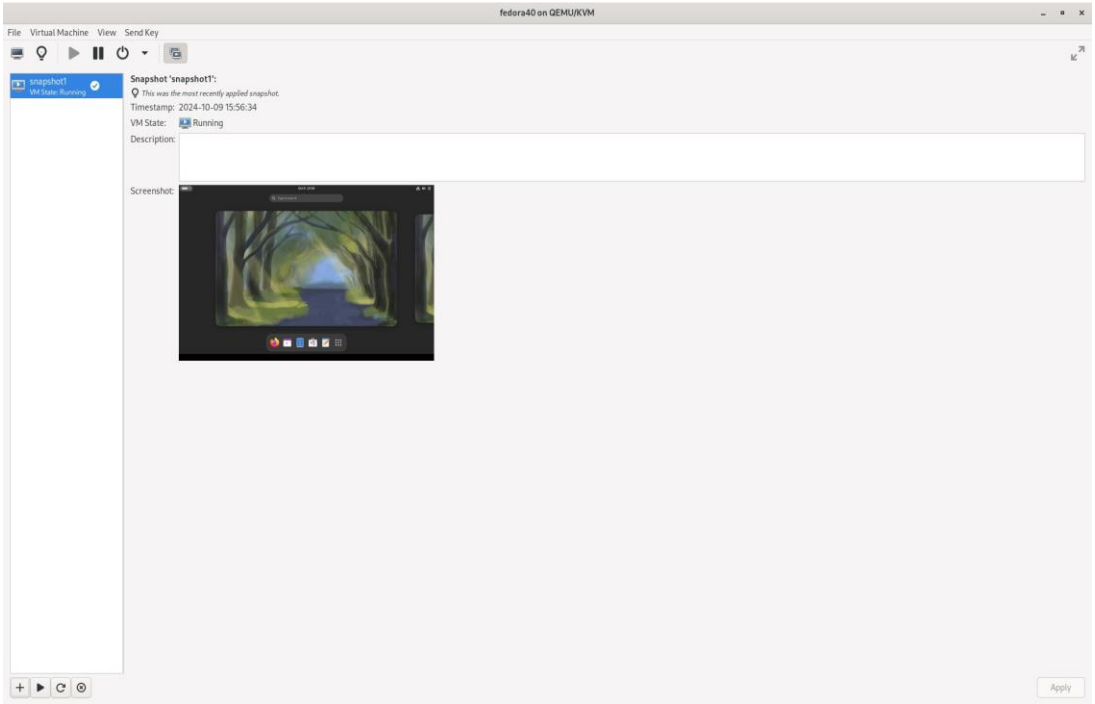
1. Snapshot Management:



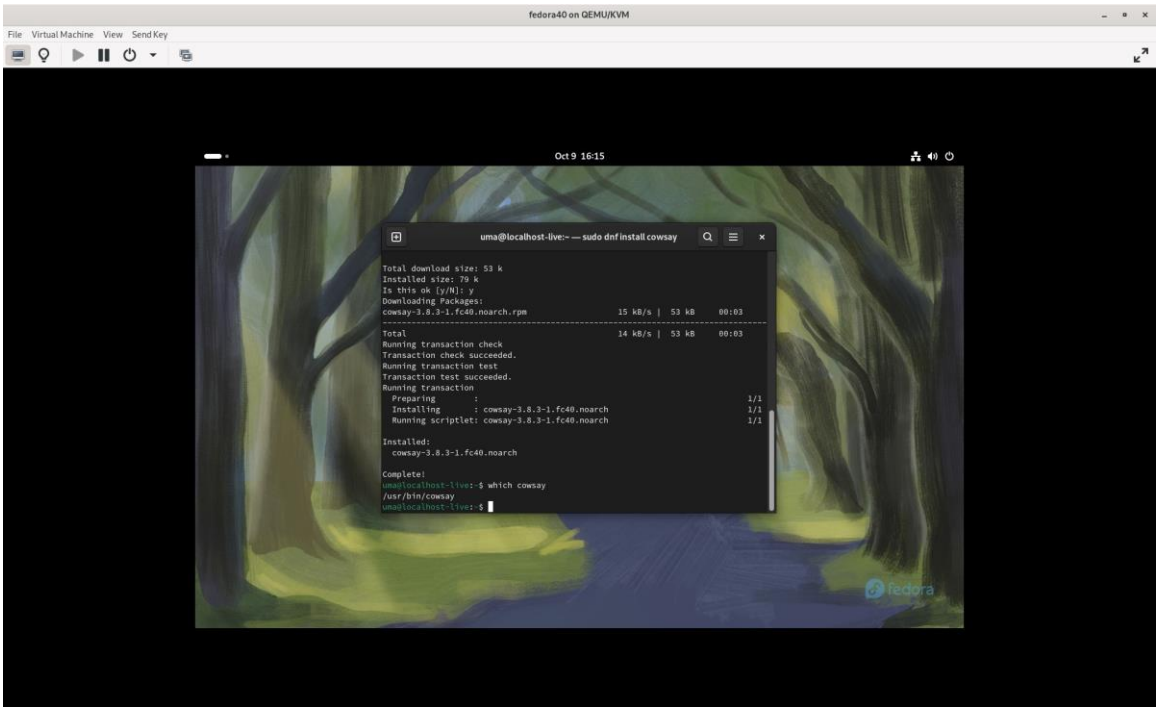



Create snapshot
Create snapshot
Name:
snapshot1
Status:
Running
Description:
Screenshot:
Cancel
Finish

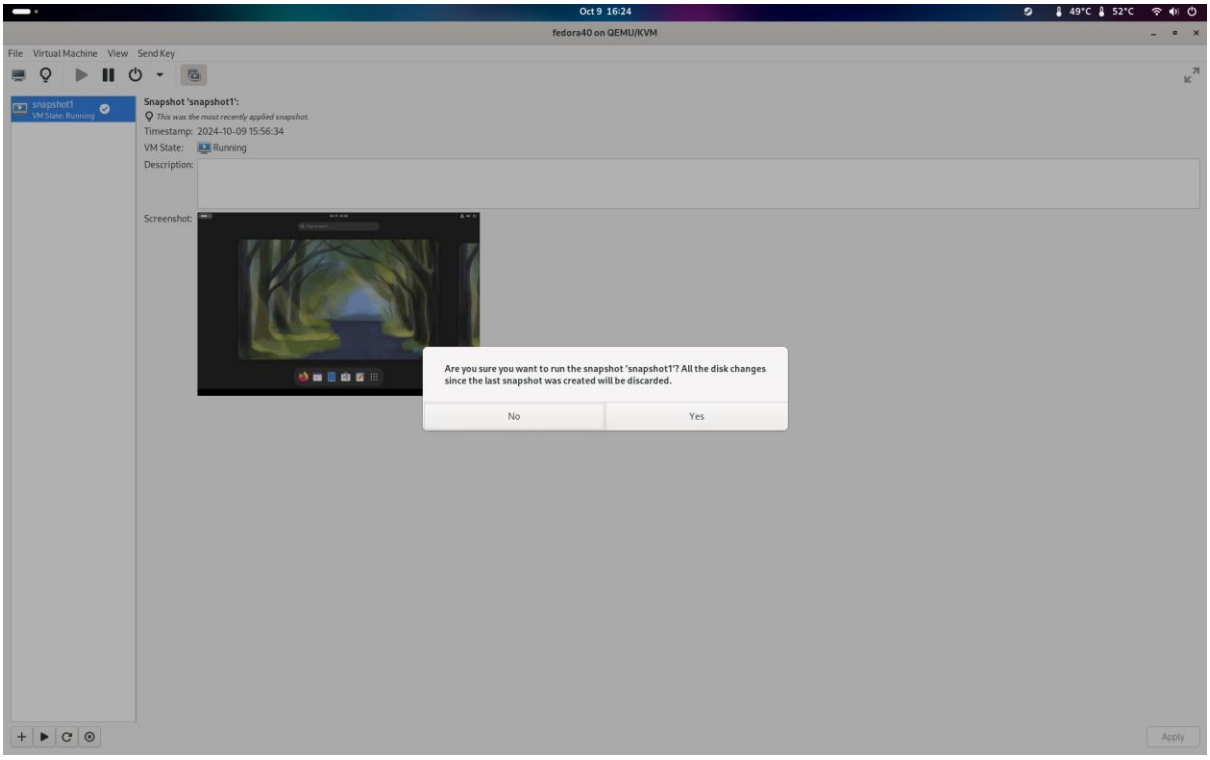

Creating snapshot
Creating virtual machine snapshot
Processing...
0 %



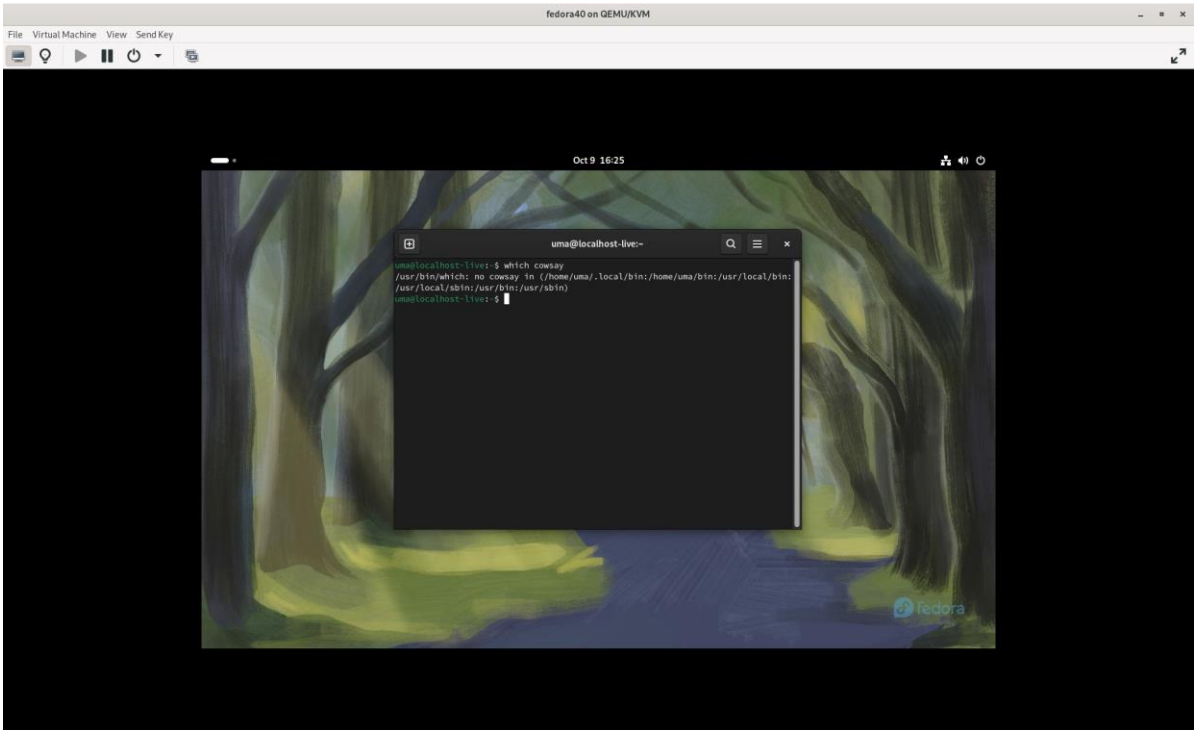
Modified the VM:



Reverting back to the Snapshot:

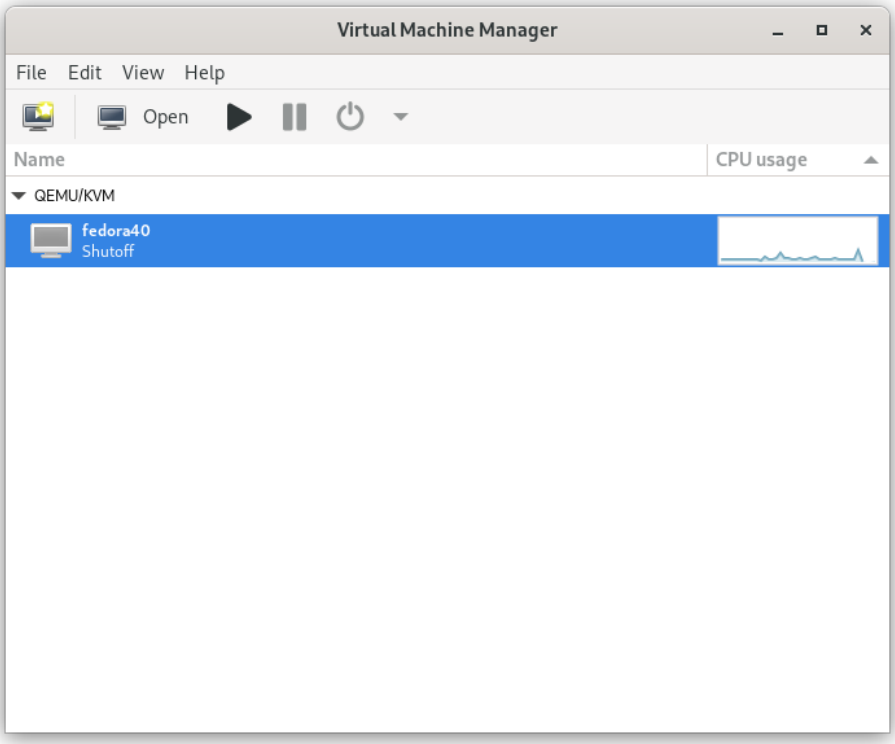


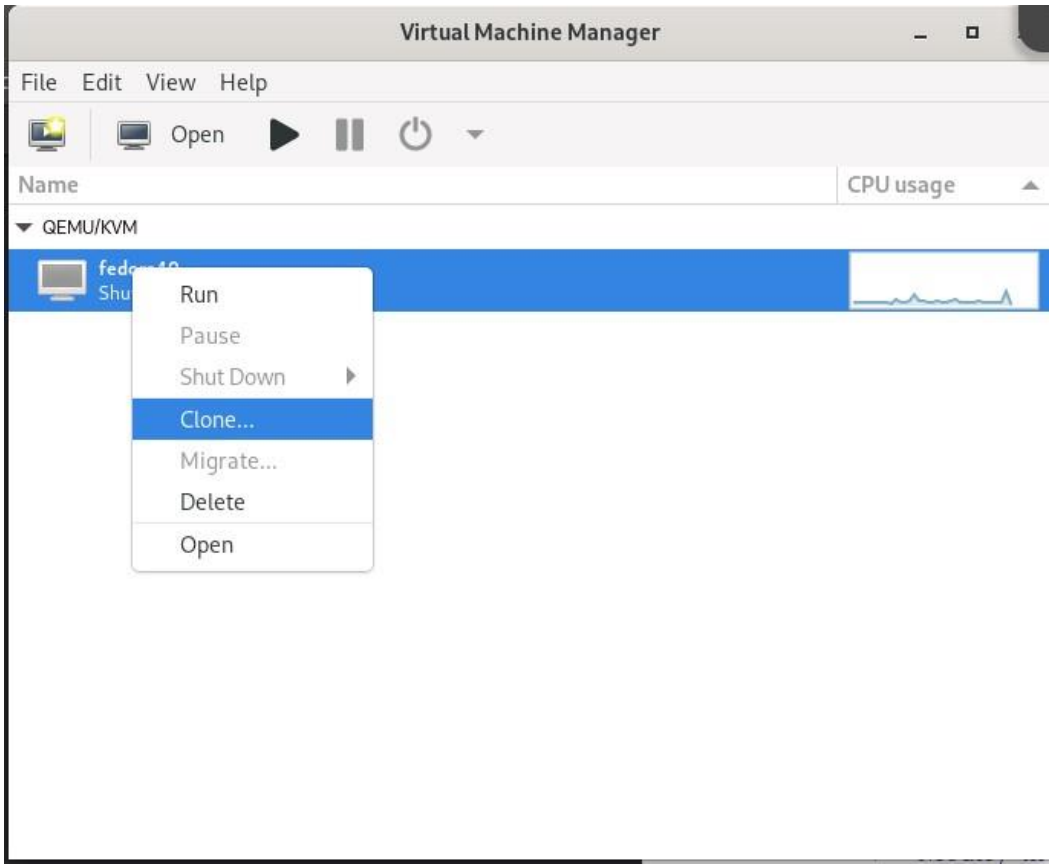
Verifying:

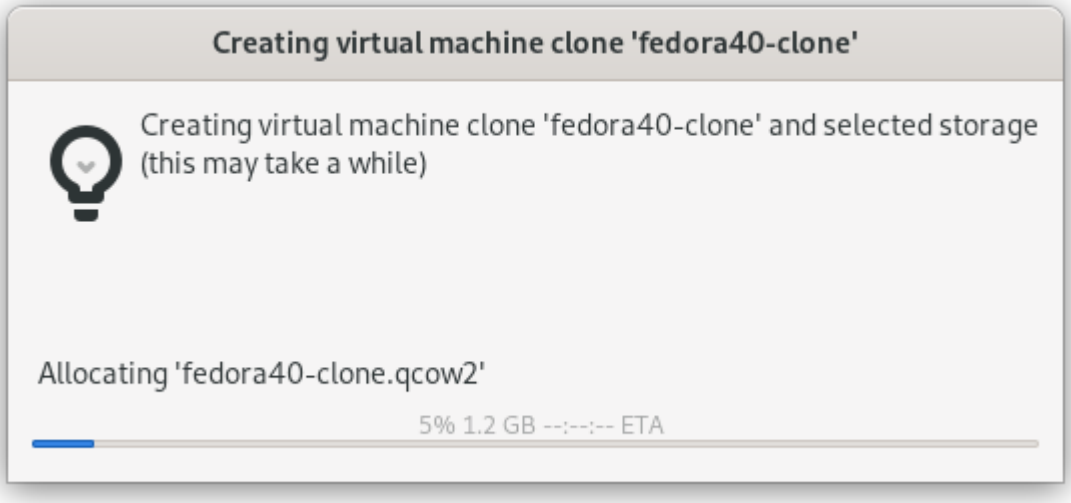


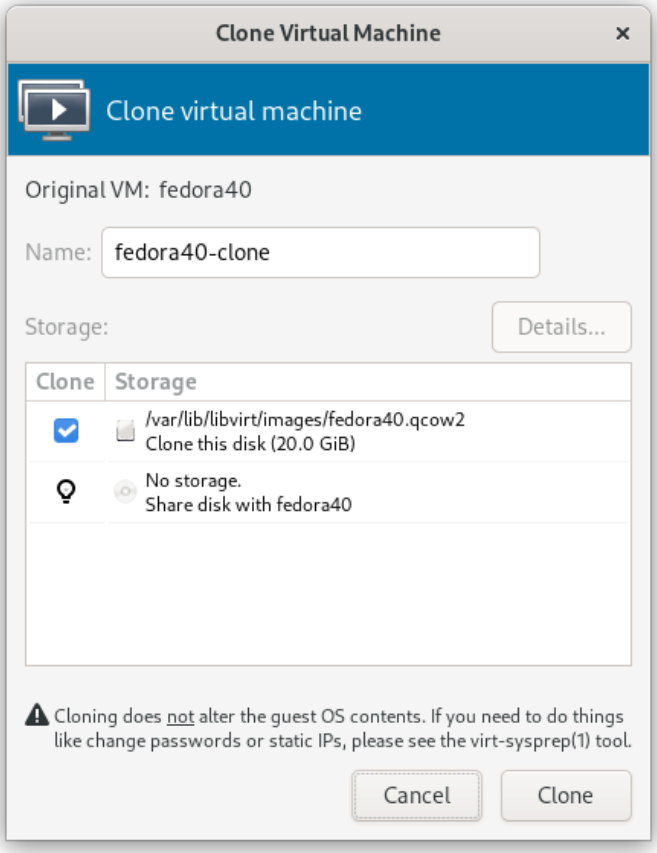
2. VM Cloning:

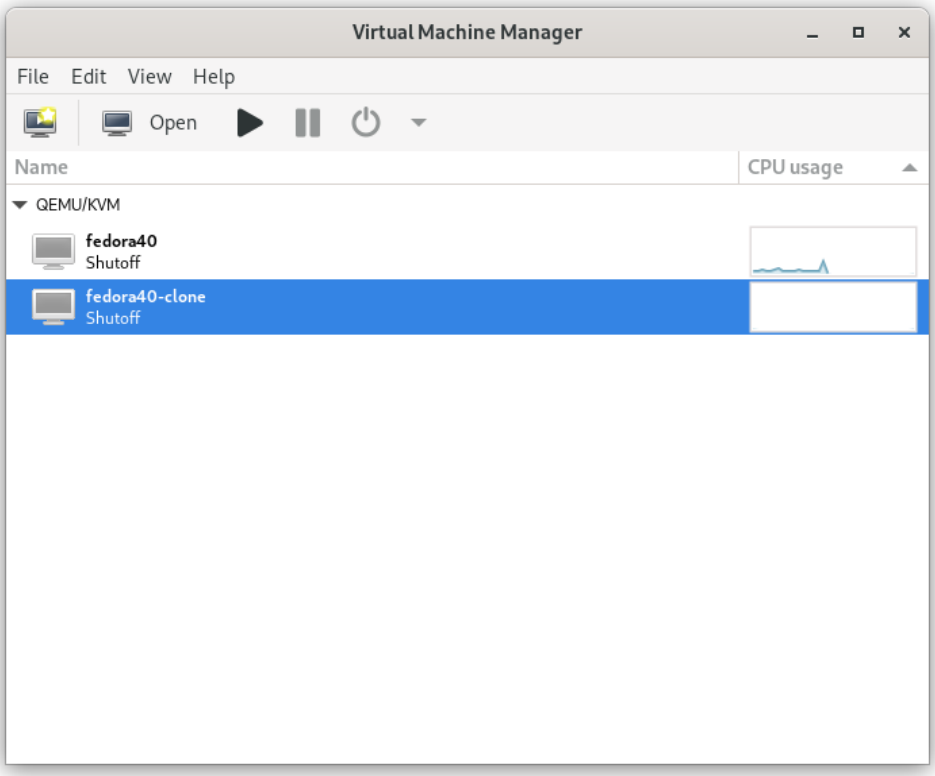
- Shut down the VM, then clone it using the "Clone" option in `virt-manager`.


Virtual Machine Manager
File
Edit
View
Help
Open
Name
CPU usage
QEMU/KVM
fedora40
Shutoff

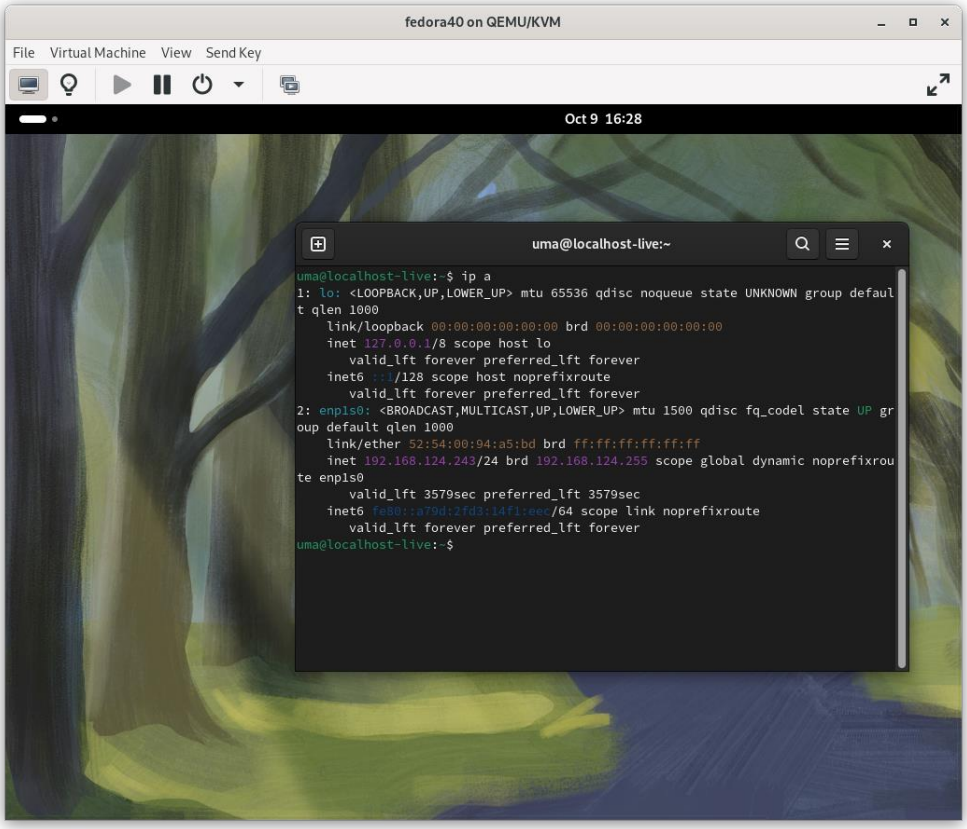

Virtual Machine Manager
File
Edit
View
Help
Open
Name
CPU usage
QEMU/KVM
Run
Pause
Shut Down
Clone...
Migrate...
Delete
Open

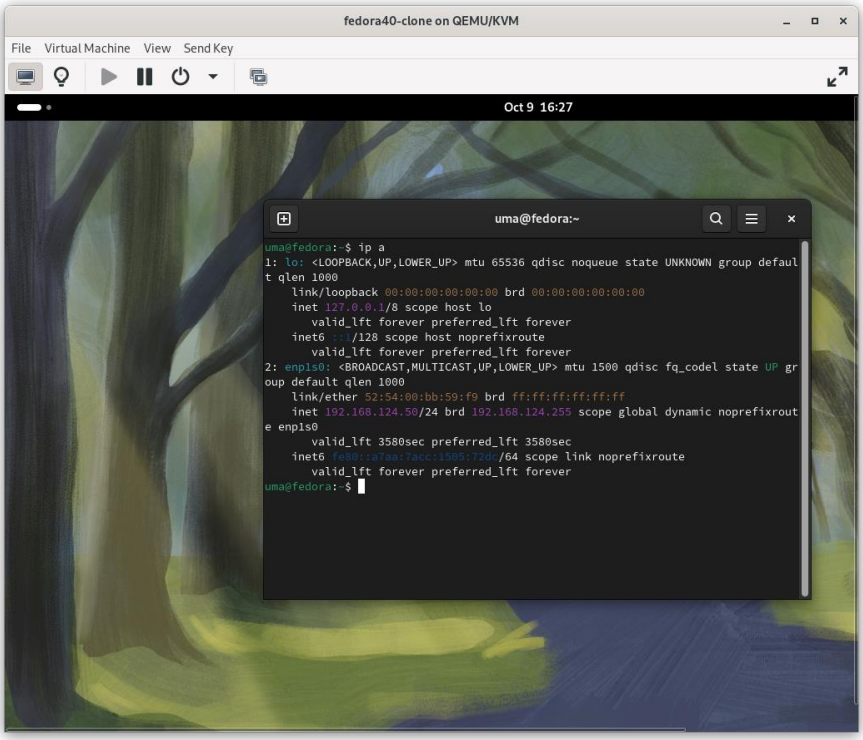

Creating virtual machine clone 'fedora40-clone'
Creating virtual machine clone 'fedora40-clone' and selected storage (this may take a while)
Allocating 'fedora40-clone.qcow2'
5% 1.2 GB --:--:-- ETA





- Verify changes by checking the new VM's hostname and IP.


fedora40 on QEMU/KVM
File Virtual Machine View Send Key
Oct 9 16:28
uma@localhost-live:~
uma@localhost-live:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host noprefixroute
valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
link/ether 52:54:00:94:a5:bd brd ff:ff:ff:ff:ff:ff
inet 192.168.124.243/24 brd 192.168.124.255 scope global dynamic noprefixrou
te enp1s0
valid_lft 3579sec preferred_lft 3579sec
inet6 fe80::a79d:2fd3:14f1:eec/64 scope link noprefixroute
valid_lft forever preferred_lft forever
uma@localhost-live:~$


fedora40-clone on QEMU/KVM
File Virtual Machine View Send Key
Oct 9 16:27
uma@fedora:~
uma@fedora:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host noprefixroute
valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
link/ether 52:54:00:bb:59:f9 brd ff:ff:ff:ff:ff:ff
inet 192.168.124.50/24 brd 192.168.124.255 scope global dynamic noprefixrout
e enp1s0
valid_lft 3580sec preferred_lft 3580sec
inet6 fe80::a7aa:7acc:1505:72d2/64 scope link noprefixroute
valid_lft forever preferred_lft forever
uma@fedora:~$

Task 4: Resource Allocation and Monitoring

Objectives:

1. Resource Allocation:

   - Adjust the VM's CPU and memory settings.

   - Explain how these changes impact performance.

2. Performance Monitoring:

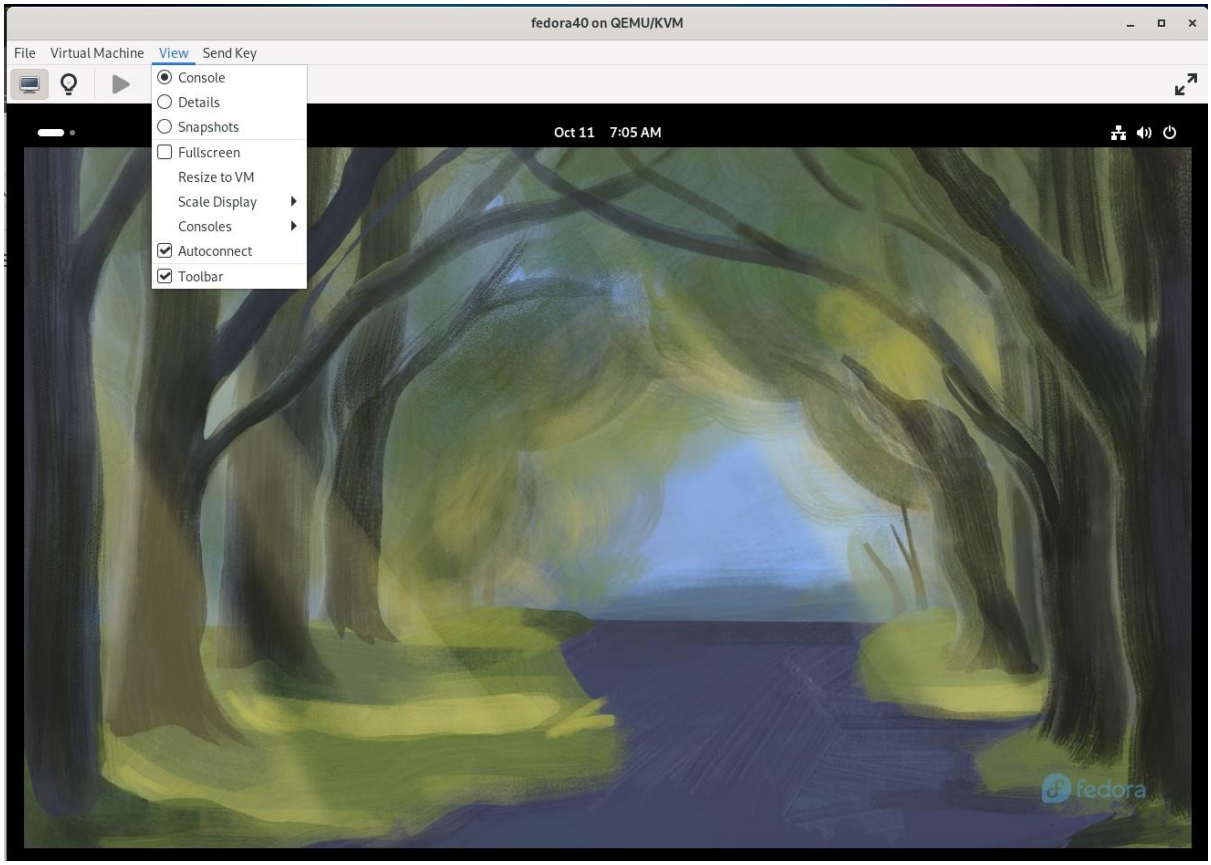
   - Monitor performance using tools like `htop`, `vmstat`, or `virt-top`.

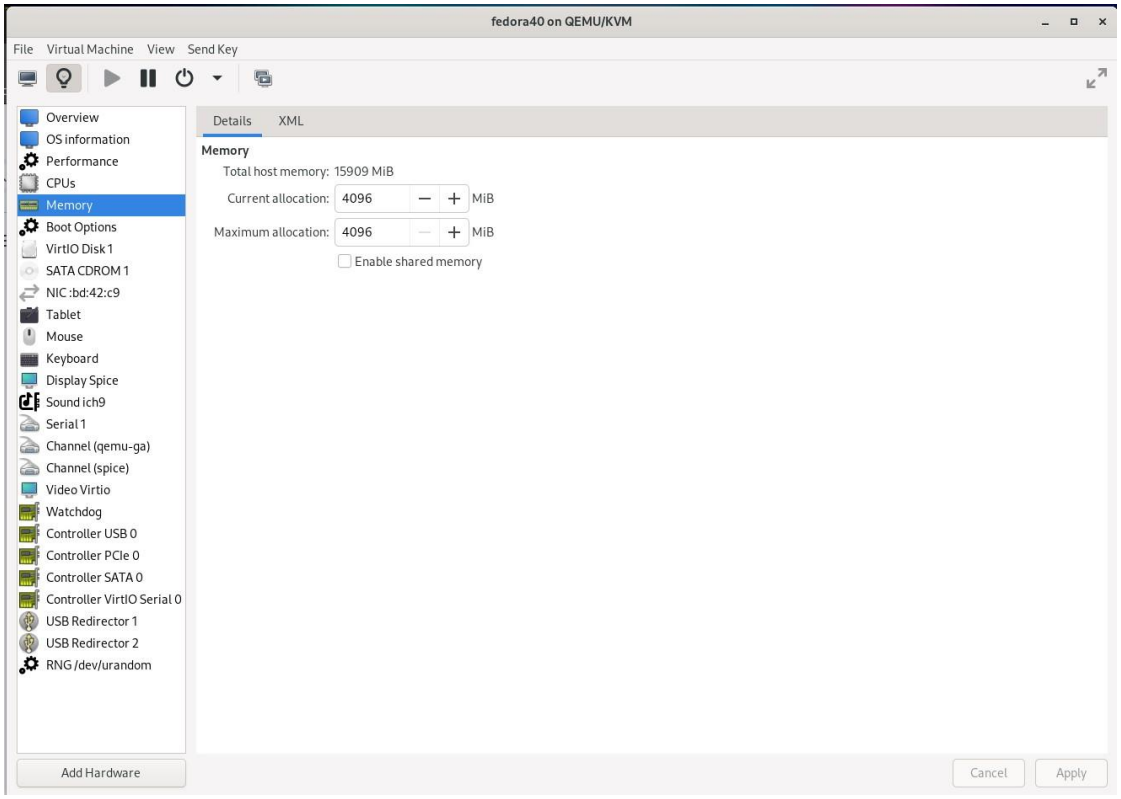
   - Analyze the resource usage (CPU, memory, disk I/O).

Steps:

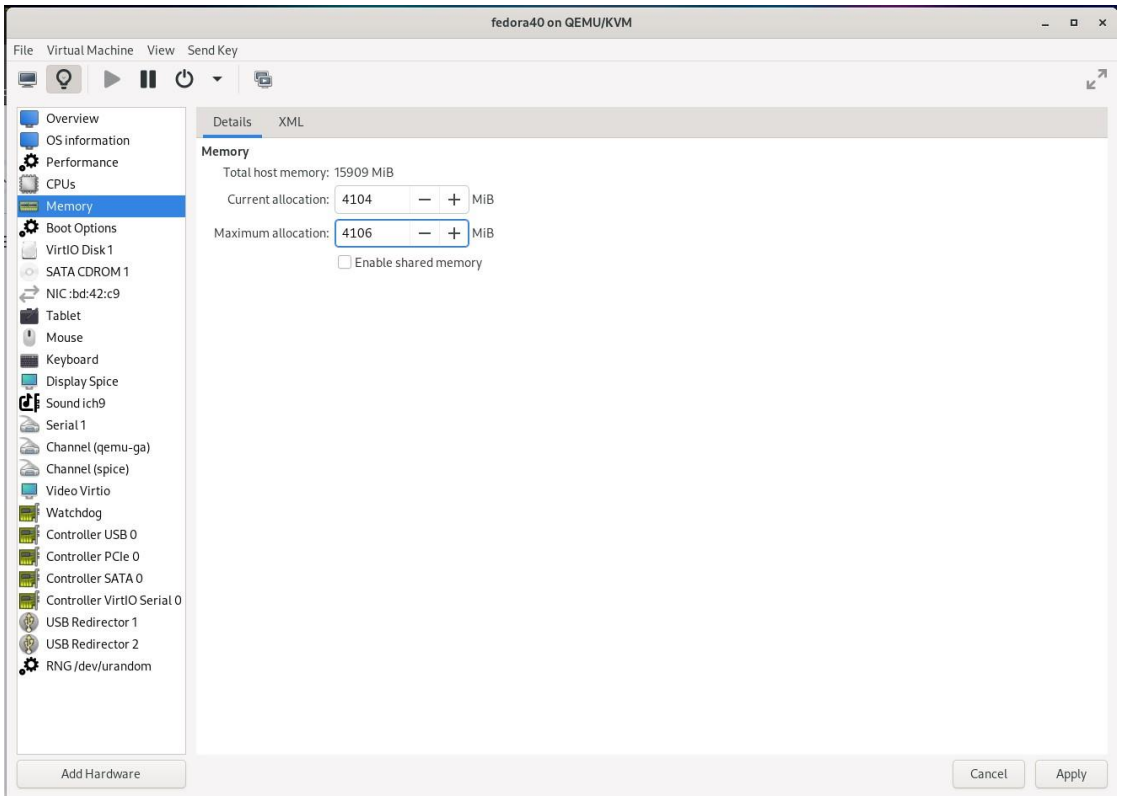
1. Modify Resource Allocation:

   - Adjust settings in `virt-manager` under "View > Details".





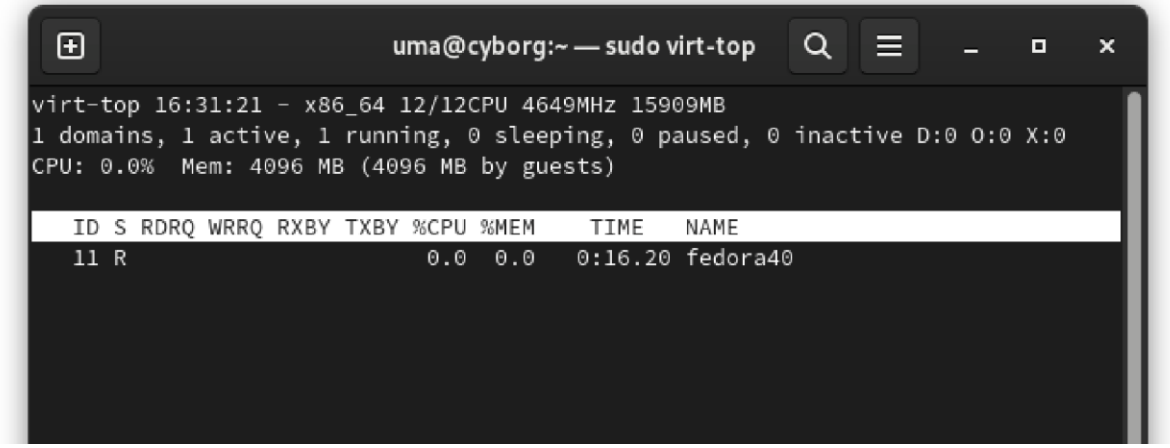
Modify the Memory and Click on Apply



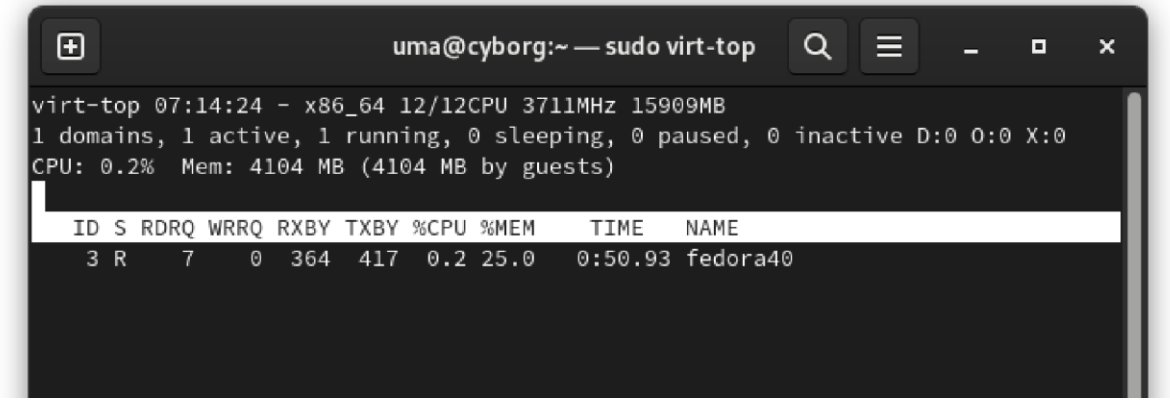
- Shutdown and restart the VM to apply changes.

2. Monitor Performance:

- Use commands like `virt-top` to observe CPU and memory usage.



- Document any performance changes based on resource allocation.



Task 5: Networking and Storage Management

Objectives:

1. Network Interface Management:

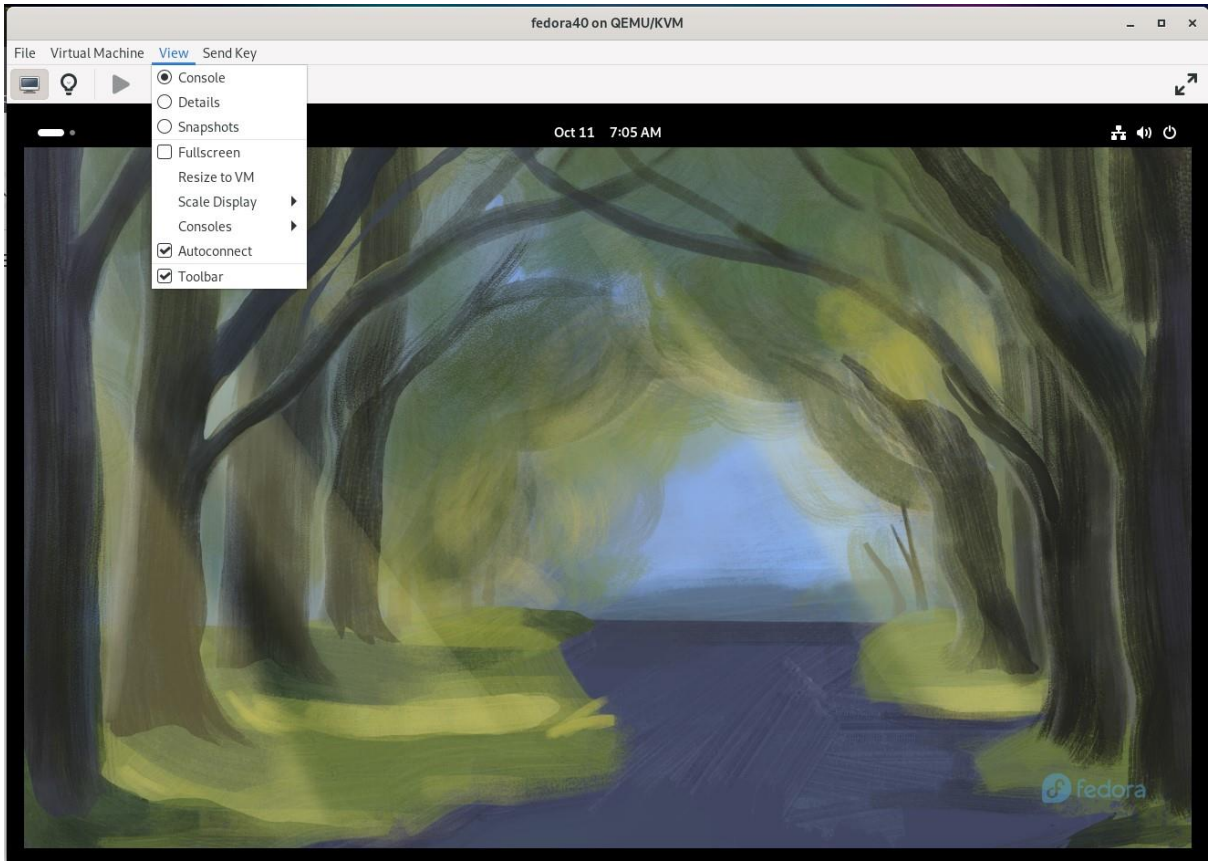
   - Add a virtual network interface for internal communication.

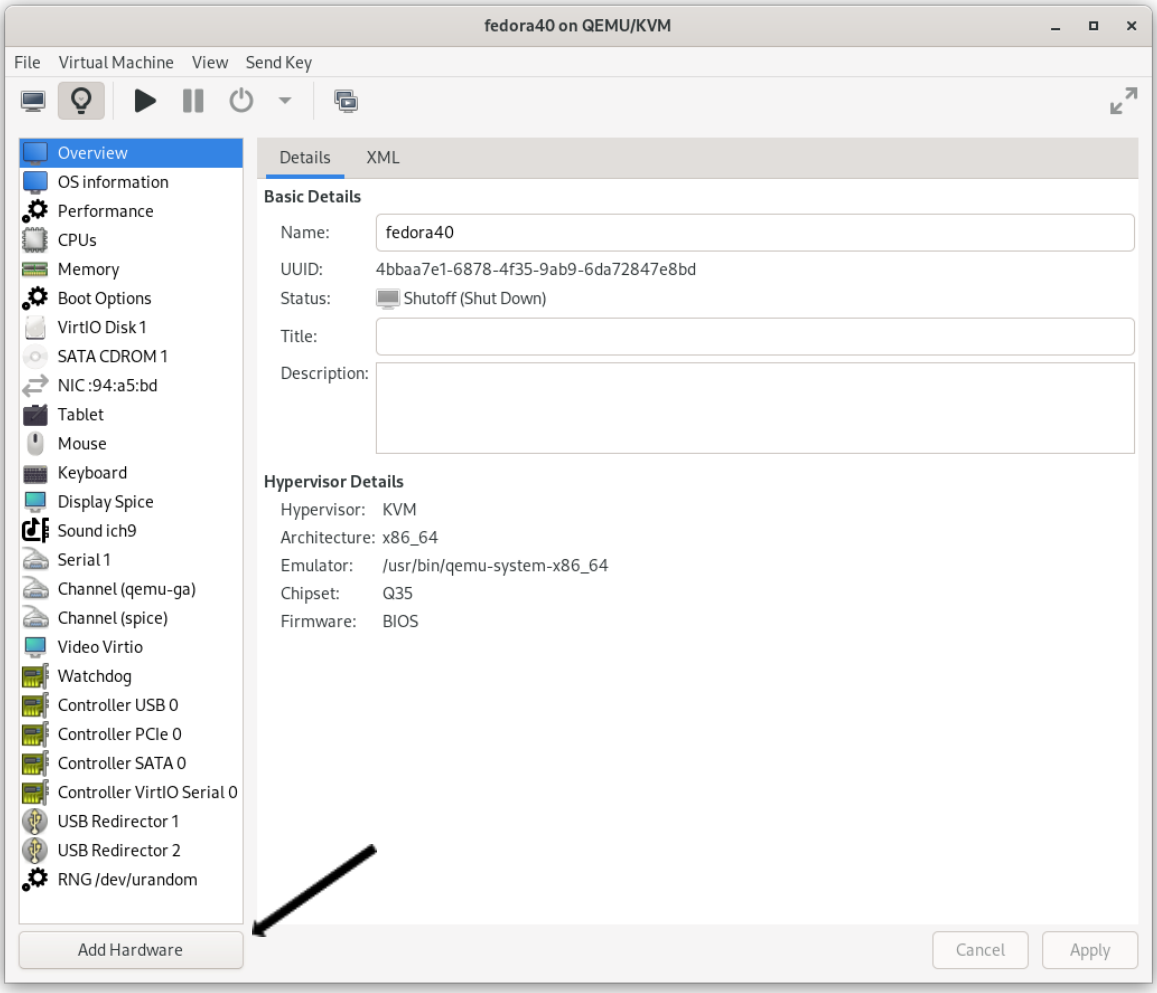
2. Storage Management:

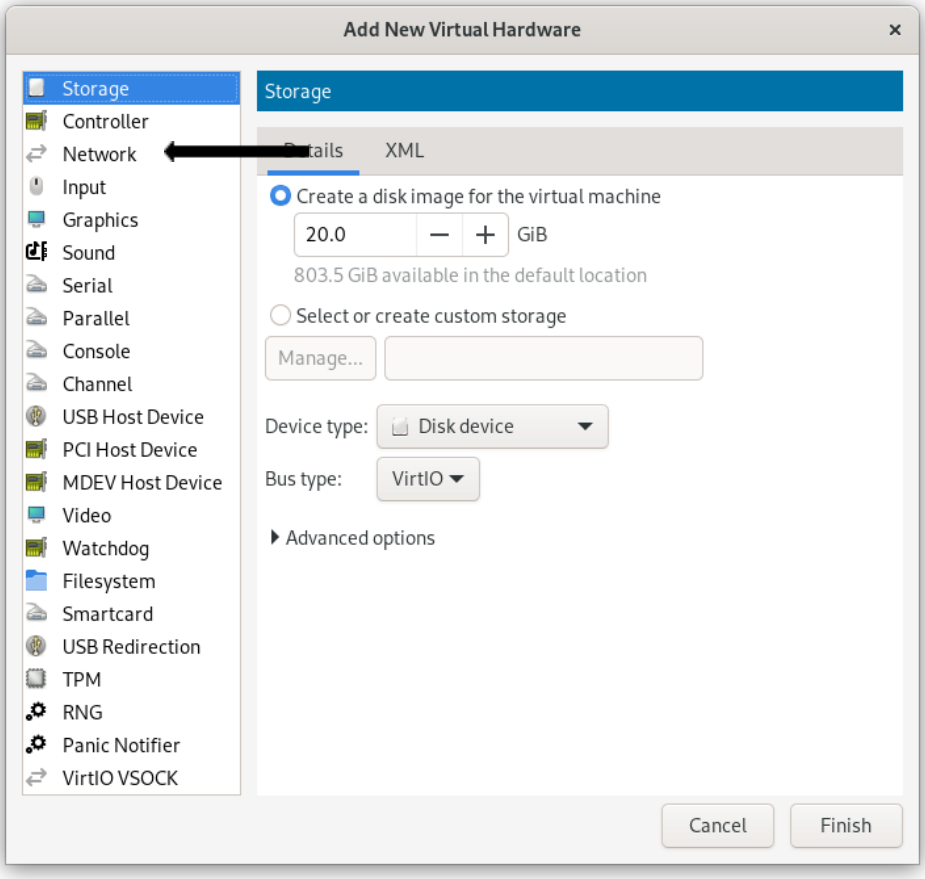
   - Attach a virtual disk and configure it for use in the guest OS.

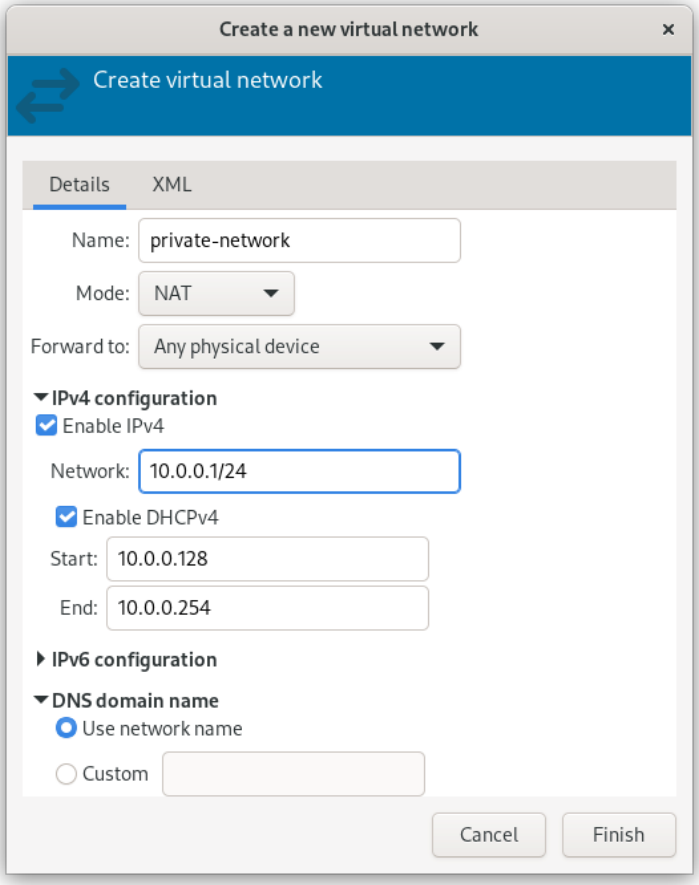
Steps:

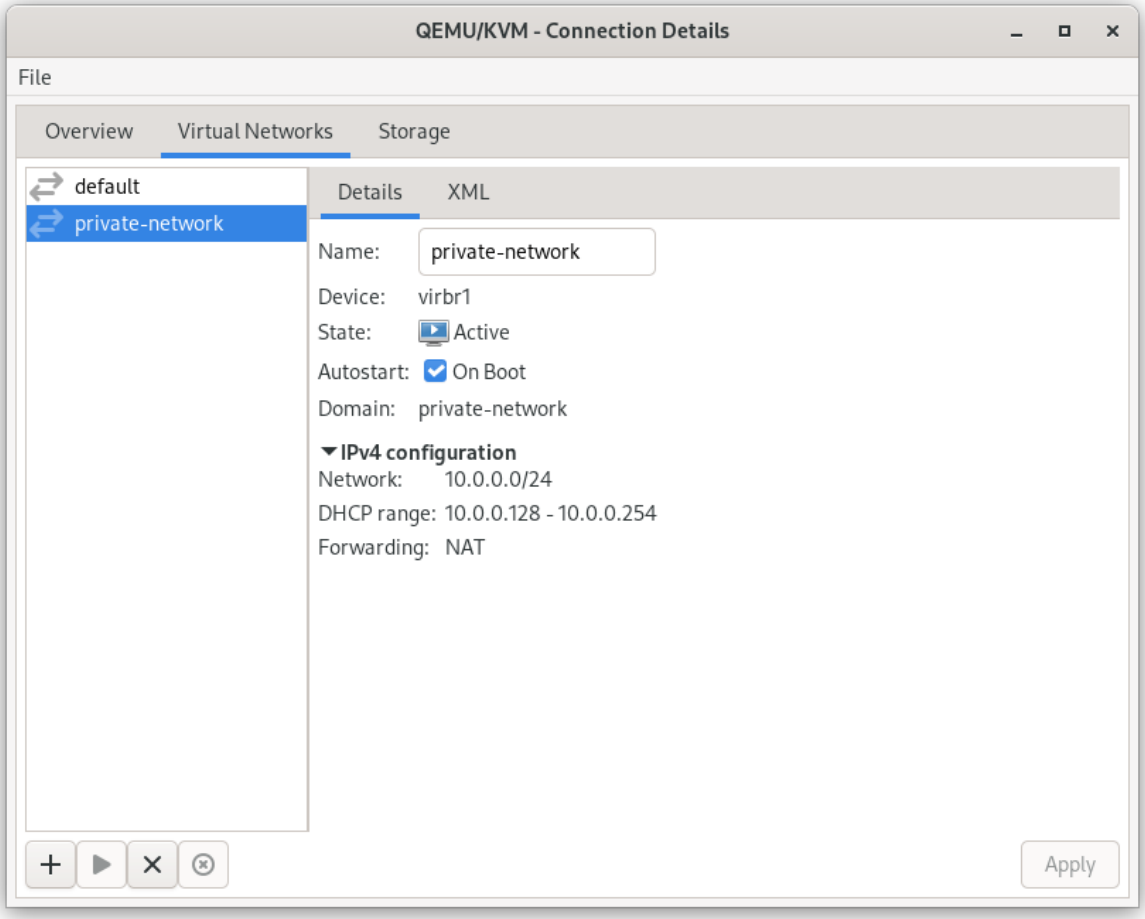
1. Add a Network Interface:

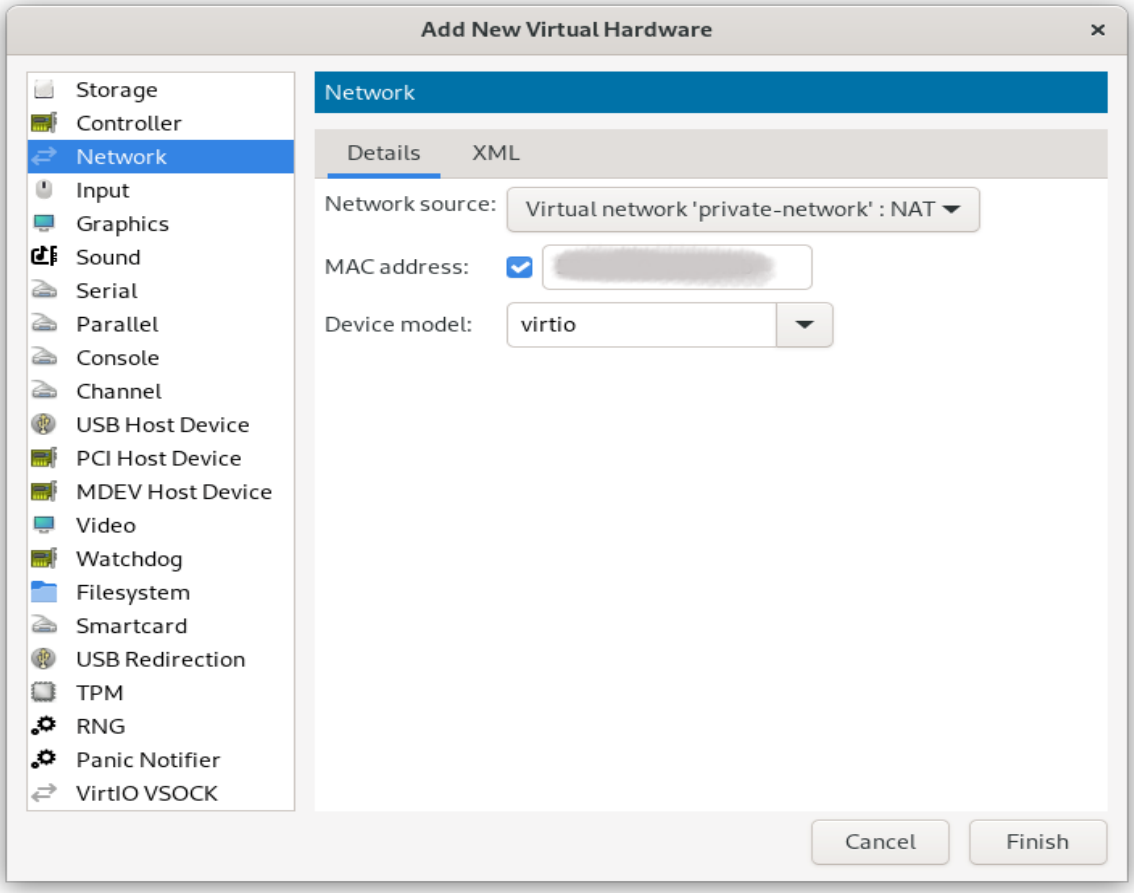


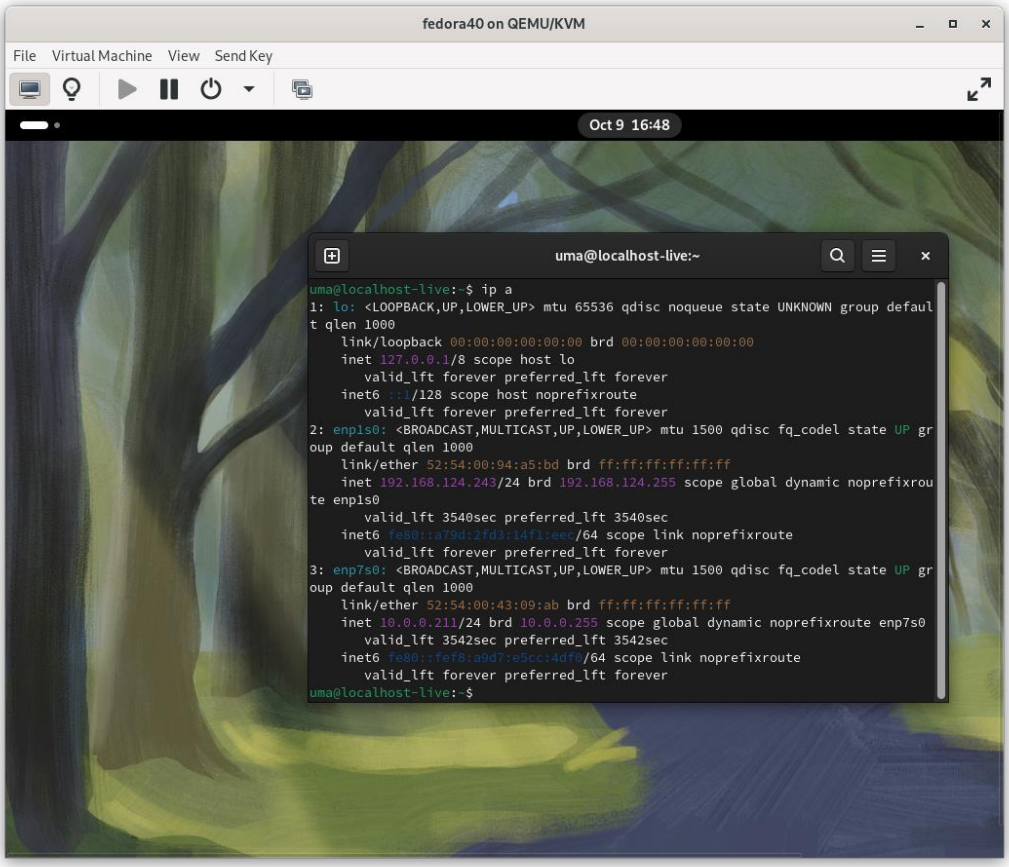

fedora40 on QEMU/KVM
File
Virtual Machine
View
Send Key
Overview
OS information
Performance
CPUs
Memory
Boot Options
VirtIO Disk 1
SATA CDROM 1
NIC :94:a5:bd
Tablet
Mouse
Keyboard
Display Spice
Sound ich9
Serial 1
Channel (qemu-ga)
Channel (spice)
Video Virtio
Watchdog
Controller USB 0
Controller PCIe 0
Controller SATA 0
Controller VirtIO Serial 0
USB Redirector 1
USB Redirector 2
RNG /dev/urandom
Add Hardware
Details
XML
Basic Details
Name:
fedora40
UUID:
4bbaa7e1-6878-4f35-9ab9-6da72847e8bd
Status:
Shutoff (Shut Down)
Title:
Description:
Hypervisor Details
Hypervisor:
KVM
Architecture:
x86_64
Emulator:
/usr/bin/qemu-system-x86_64
Chipset:
Q35
Firmware:
BIOS
Cancel
Apply

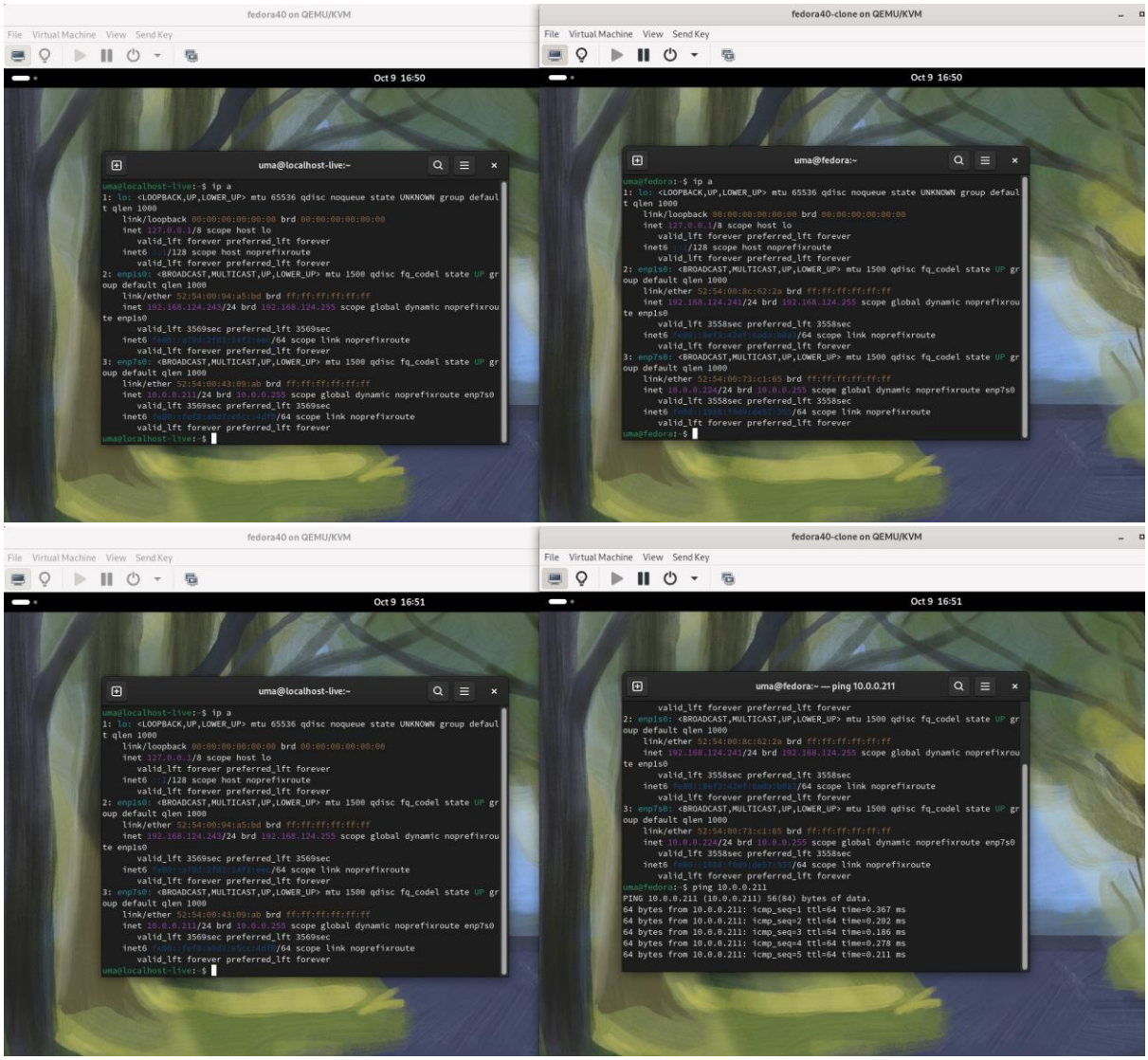

Add New Virtual Hardware
Storage
Controller
Network
Input
Graphics
Sound
Serial
Parallel
Console
Channel
USB Host Device
PCI Host Device
MDEV Host Device
Video
Watchdog
Filesystem
Smartcard
USB Redirection
TPM
RNG
Panic Notifier
VirtIO VSOCK
Storage
Details
XML
Create a disk image for the virtual machine
20.0
GiB
803.5 GiB available in the default location
Select or create custom storage
Manage...
Device type:
Disk device
Bus type:
VirtIO
Advanced options
Cancel
Finish


Create a new virtual network
Create virtual network
Details
XML
Name:
private-network
Mode:
NAT
Forward to:
Any physical device
IPv4 configuration
Enable IPv4
Network:
10.0.0.1/24
Enable DHCPv4
Start:
10.0.0.128
End:
10.0.0.254
IPv6 configuration
DNS domain name
Use network name
Custom
Cancel
Finish


QEMU/KVM - Connection Details
File
Overview
Virtual Networks
Storage
default
private-network
Details
XML
Name:
private-network
Device:
virbr1
State:
Active
Autostart:
On Boot
Domain:
private-network
IPv4 configuration
Network:
10.0.0.0/24
DHCP range:
10.0.0.128 - 10.0.0.254
Forwarding:
NAT
Apply

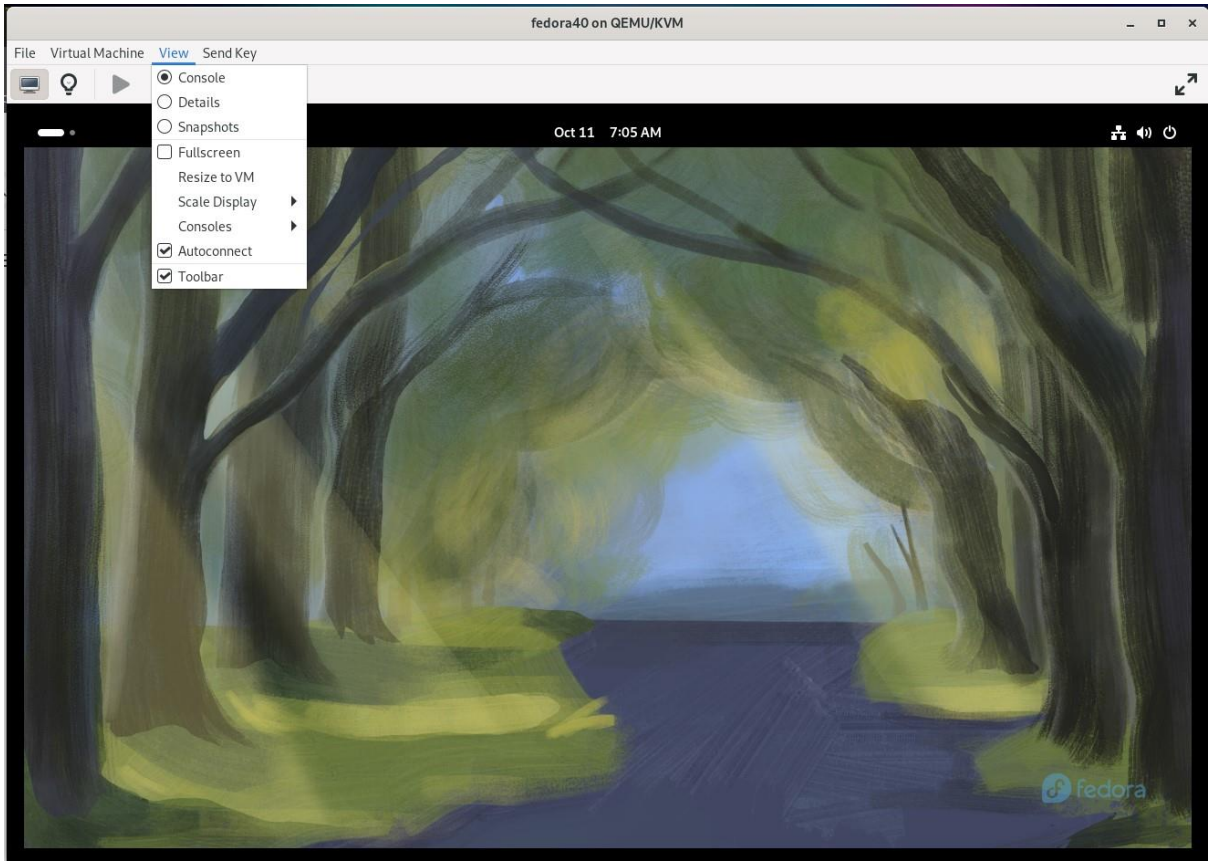

Add New Virtual Hardware
Storage
Controller
Network
Input
Graphics
Sound
Serial
Parallel
Console
Channel
USB Host Device
PCI Host Device
MDEV Host Device
Video
Watchdog
Filesystem
Smartcard
USB Redirection
TPM
RNG
Panic Notifier
VirtIO VSOCK
Network
Details
XML
Network source:
Virtual network 'private-network' : NAT
MAC address:
Device model:
virtio
Cancel
Finish

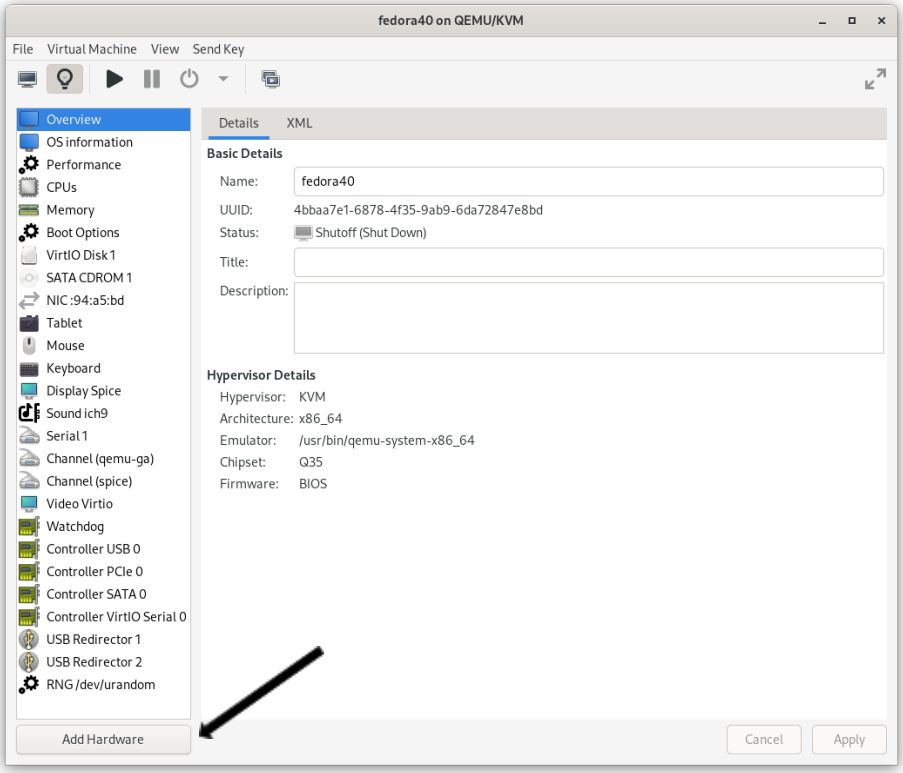

fedora40 on QEMU/KVM
File
Virtual Machine
View
Send Key
Oct 9 16:48
uma@localhost-live:~
uma@localhost-live:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host noprefixroute
valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
link/ether 52:54:00:94:a5:bd brd ff:ff:ff:ff:ff:ff
inet 192.168.124.243/24 brd 192.168.124.255 scope global dynamic noprefixrou
te enp1s0
valid_lft 3540sec preferred_lft 3540sec
inet6 fe80::a79d:2fd3:14f1:eec/64 scope link noprefixroute
valid_lft forever preferred_lft forever
3: enp7s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
link/ether 52:54:00:43:09:ab brd ff:ff:ff:ff:ff:ff
inet 10.0.0.211/24 brd 10.0.0.255 scope global dynamic noprefixroute enp7s0
valid_lft 3542sec preferred_lft 3542sec
inet6 fe80::fef8:a9d7:e5cc:4df0/64 scope link noprefixroute
valid_lft forever preferred_lft forever
uma@localhost-live:~$

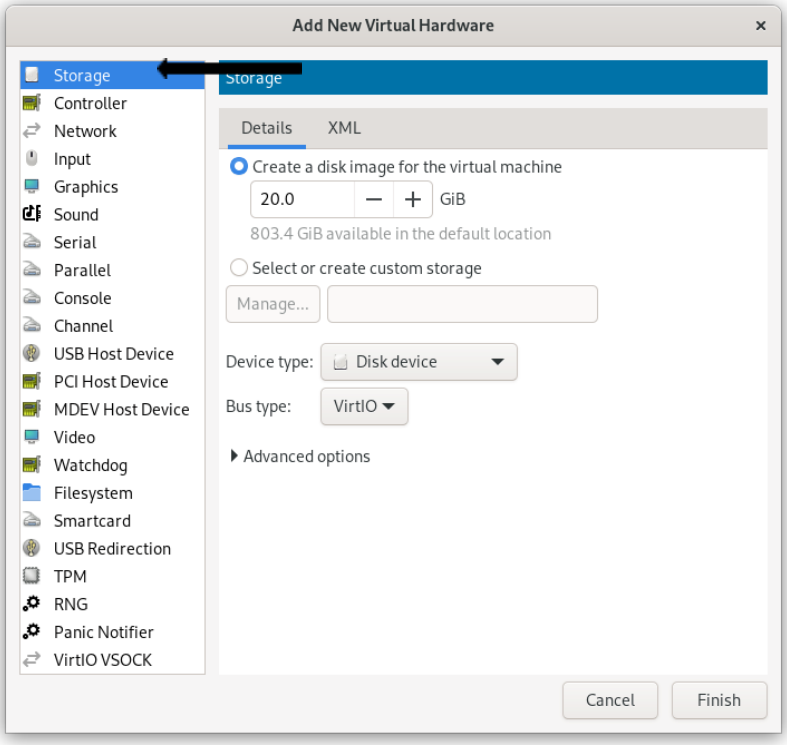


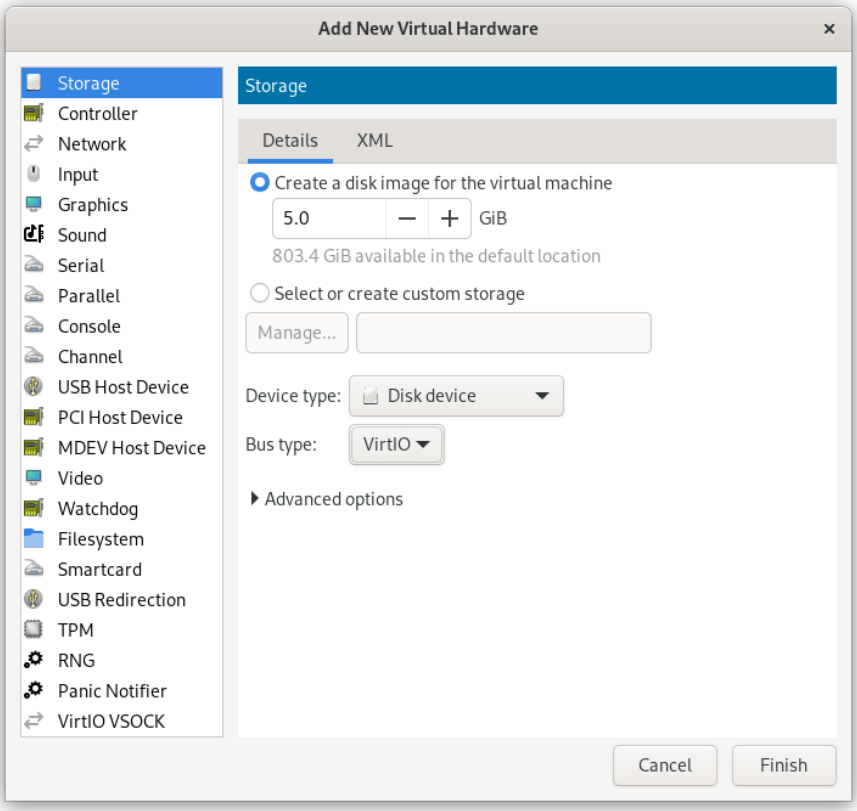
2. Attach and Configure Storage:

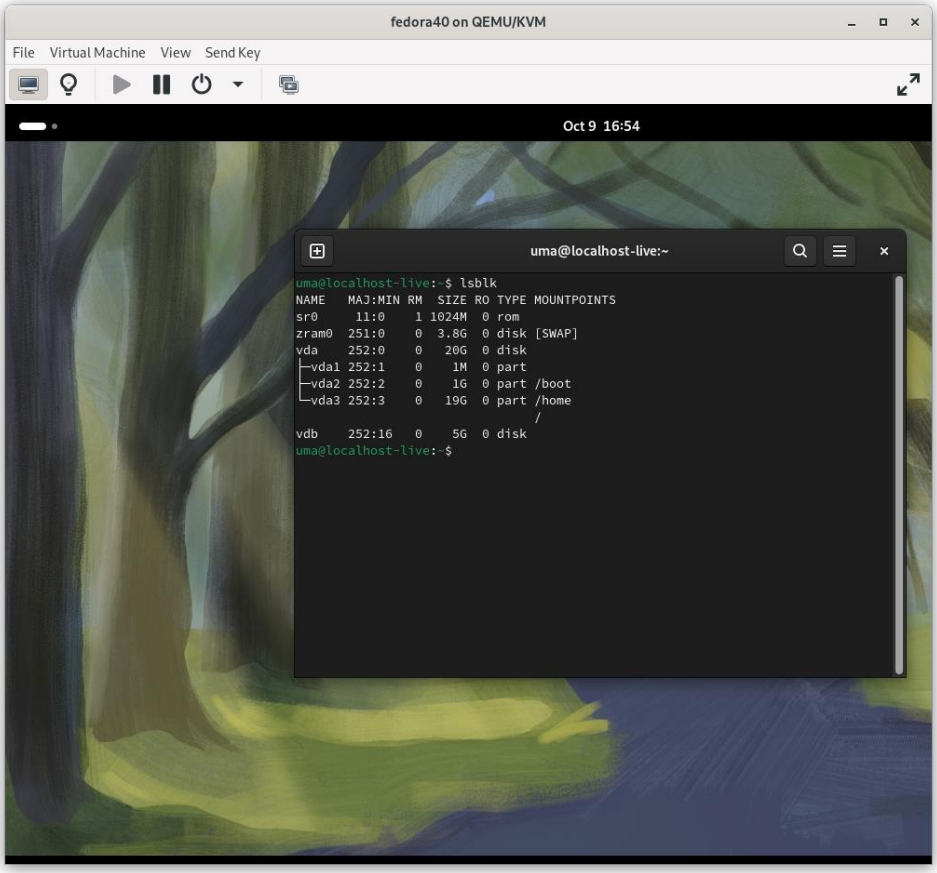
   - Add a new disk from "View > Details" in `virt-manager`.


fedora40 on QEMU/KVM
File
Virtual Machine
View
Send Key
Console
Details
Snapshots
Fullscreen
Resize to VM
Scale Display
Consoles
Autoconnect
Toolbar
Oct 11 7:05 AM
fedora

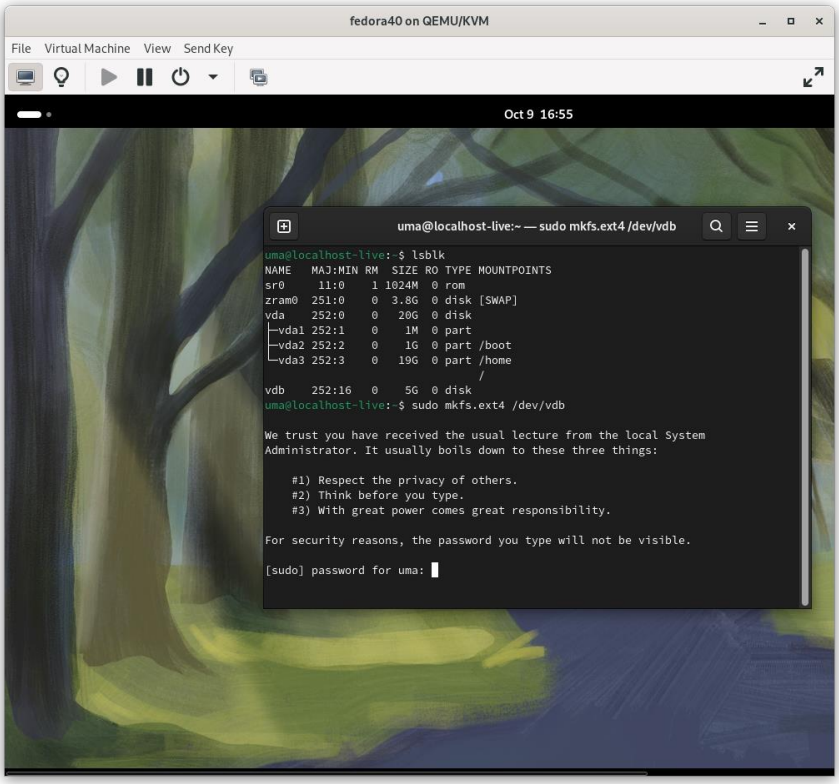

fedora40 on QEMU/KVM
File
Virtual Machine
View
Send Key
Overview
OS information
Performance
CPUs
Memory
Boot Options
VirtIO Disk 1
SATA CDROM 1
NIC :94:a5:bd
Tablet
Mouse
Keyboard
Display Spice
Sound ich9
Serial 1
Channel (qemu-ga)
Channel (spice)
Video Virtio
Watchdog
Controller USB 0
Controller PCIe 0
Controller SATA 0
Controller VirtIO Serial 0
USB Redirector 1
USB Redirector 2
RNG /dev/urandom
Add Hardware
Details
XML
Basic Details
Name: fedora40
UUID: 4bbaa7e1-6878-4f35-9ab9-6da72847e8bd
Status: Shutoff (Shut Down)
Title:
Description:
Hypervisor Details
Hypervisor: KVM
Architecture: x86_64
Emulator: /usr/bin/qemu-system-x86_64
Chipset: Q35
Firmware: BIOS
Cancel
Apply

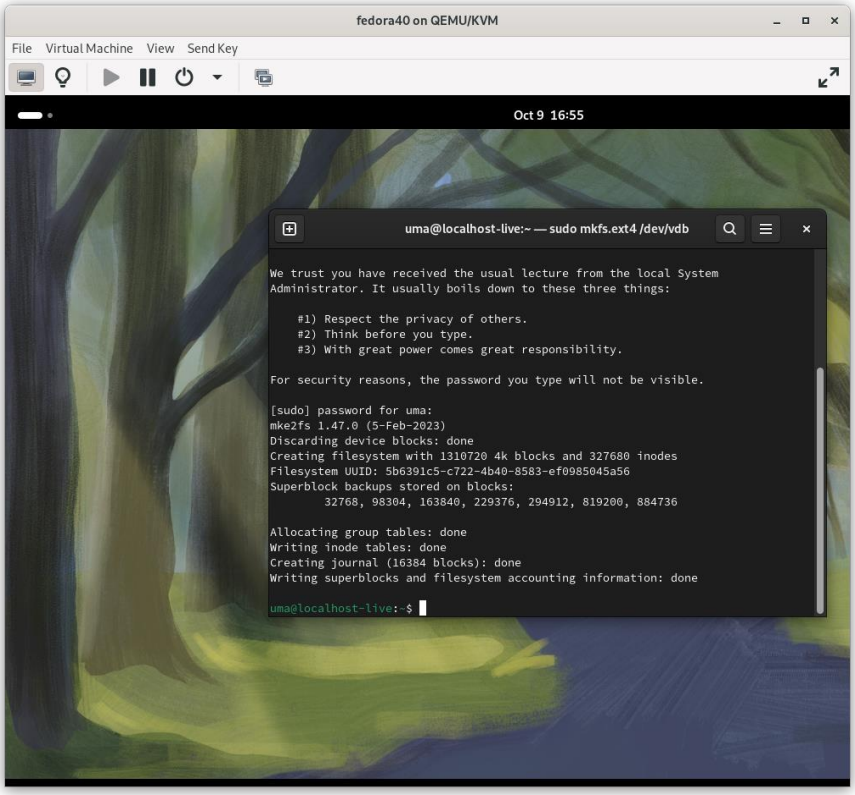

Add New Virtual Hardware
Storage
Controller
Network
Input
Graphics
Sound
Serial
Parallel
Console
Channel
USB Host Device
PCI Host Device
MDEV Host Device
Video
Watchdog
Filesystem
Smartcard
USB Redirection
TPM
RNG
Panic Notifier
VirtIO VSOCK
Storage
Details
XML
Create a disk image for the virtual machine
20.0
GiB
803.4 GiB available in the default location
Select or create custom storage
Manage...
Device type:
Disk device
Bus type:
VirtIO
Advanced options
Cancel
Finish

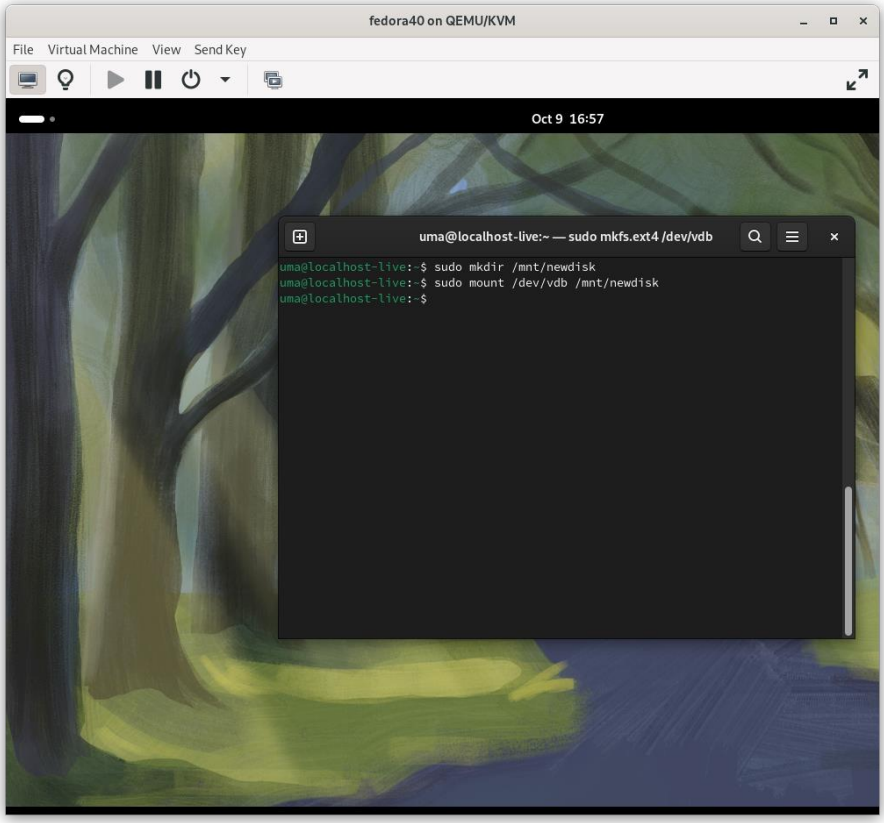

Add New Virtual Hardware
Storage
Controller
Network
Input
Graphics
Sound
Serial
Parallel
Console
Channel
USB Host Device
PCI Host Device
MDEV Host Device
Video
Watchdog
Filesystem
Smartcard
USB Redirection
TPM
RNG
Panic Notifier
VirtIO VSOCK
Storage
Details
XML
Create a disk image for the virtual machine
5.0
GiB
803.4 GiB available in the default location
Select or create custom storage
Manage...
Device type:
Disk device
Bus type:
VirtIO
Advanced options
Cancel
Finish



- Format the disk with `mkfs.ext4` and mount it (`/mnt/newdisk`).




fedora40 on QEMU/KVM
File Virtual Machine View Send Key
Oct 9 16:55
uma@localhost-live:~ — sudo mkfs.ext4 /dev/vdb
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:
#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.
For security reasons, the password you type will not be visible.
[sudo] password for uma:
mke2fs 1.47.0 (5-Feb-2023)
Discarding device blocks: done
Creating filesystem with 1310720 4k blocks and 327680 inodes
Filesystem UUID: 5b6391c5-c722-4b40-8583-ef0985045a56
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736
Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
uma@localhost-live:~$


fedora40 on QEMU/KVM
File Virtual Machine View Send Key
Oct 9 16:57
uma@localhost-live:~ — sudo mkfs.ext4 /dev/vdb
uma@localhost-live:~$ sudo mkdir /mnt/newdisk
uma@localhost-live:~$ sudo mount /dev/vdb /mnt/newdisk
uma@localhost-live:~$

Task 6: Automation with Scripts

Objectives:

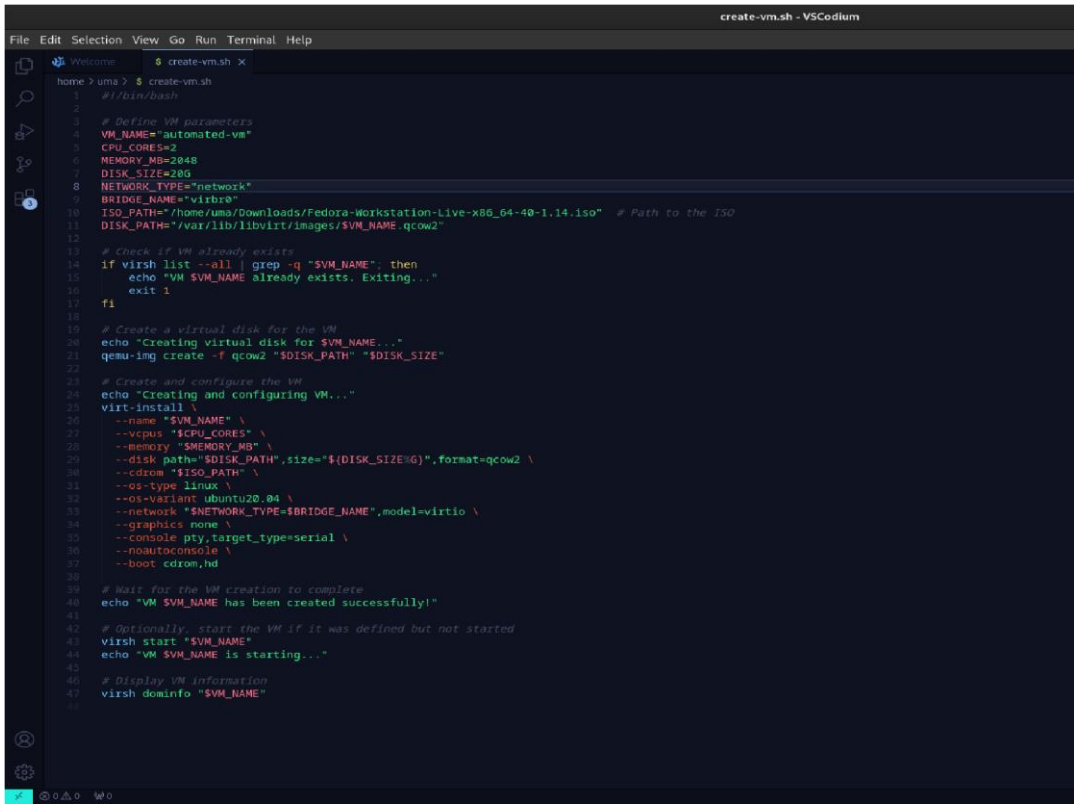
1. Automate VM Creation:

- Create a bash script for automating VM creation.

- Test the script and validate the VM's setup.

Steps:

1. Write a Bash Script:



2. Execute the Script:

```
uma@cyborg:~$ sudo ./create-vm.sh
Creating virtual disk for automated-vm...
Formatting '/var/lib/libvirt/images/automated-vm.qcow2', fmt=qcow2 cluster_size=6
5536 extended_l2=off compression_type=zlib size=21474836480 lazy_refcounts=off re
fcount_bits=16
Creating and configuring VM...

Starting install...
Creating domain...                              |    0 B  00:00

Domain is still running. Installation may be in progress.
You can reconnect to the console to complete the installation process.
VM automated-vm has been created successfully!
error: Domain is already active

VM automated-vm is starting...
Id:             18
Name:           automated-vm
UUID:           e644eac1-3a99-4e50-8227-3e4466585dcf
OS Type:        hvm
State:          running
CPU(s):         2
CPU time:       0.4s
Max memory:     2097152 KiB
Used memory:    2097152 KiB
Persistent:     yes
Autostart:      disable
Managed save:   no
Security model: selinux
Security DOI:   0
Security label: system_u:system_r:svirt_t:s0:c11,c175 (enforcing)

uma@cyborg:~$
```