

# HowTo ASIX Virtualization and installations

Curs 2021-2022

---

<b>Virtualization</b>	<b>4</b>
Conceptes clau	4
Instal·lació de les eines de virtualització Kvm/Qemu/Libvirt	5
Requeriments	5
Instal·lació del software	5
Configurar libvirtd per ser executat com un usuari no privilegiat	5
<b>Laboratori de pràctiques</b>	<b>8</b>
Per instal·lar via HTTP	8
Per instal·lar usant netinst	8
Per instal·lar usant un DVD complet	8
Imatges Cloud	9
Altres imatges	9
Tricks: users & passwords	10
<b>Eines virt</b>	<b>11</b>
Virt-manager	12
Crear una nova màquina virtual:	12
Opcions de creació d'una nova VM	12
Passos de creació de la VM:	13
Eliminar una màquina virtual	15
Configuració hardware d'una VM	16
Pràctica-1 Virt-manager	16
Virt-install	18
Descripció	18
Opcions de configuració	18
Exemples del Man	20
Pràctica-2 Virt-install	21
Virt-shell	24
Pràctica-3 Virsh	25
qemu	26
qemu / qemu-system-x86_64	26
qemu-img	27
Pràctica-4 qemu / qemu-img	27
Virt-clone	28

Virt-image	28
Virt-convert	28
Virt-top	28
Virt-viewer	28
Virt-xml-validate	28
<b>Configuracions de xarxa</b>	<b>29</b>
Default network (NAT)	30
Private (named)	30
Isolated	30
Bridge	31
Crear / Configurar bridge en Debian	31
<b>Instal·lacions</b>	<b>33</b>
Fedora-32	34
Layout per fer particions:	34
Instal·lació Fedora-32	34
Opcions de particionat de Fedora-32: automàtic	35
Opcions de particionat de Fedora-32: Custom	36
Exemple amb LVM + Create them automatically	37
Observar les opcions de configuració manual	38
Exemple amb Standard Partition + Create them automatically	38
Exemple de creació de particions manualment (gràfic)	39
Exemple de creació de particions Btrfs	40
Exemple d'espai insuficient per a la instal·lació	41
Debian-11 <pendent>	42
Ubuntu 20 <pendent>	43

---

---

Virt-Manager	kvm / qemu / libvirt
Vagrant	
Packer	
Ansible	
Cloud-init	

Docker	
Kubernetes	
Cloud Providers: AWS / Google	

- ☐ <https://wiki.debian.org/QEMU>
  - ☐ <https://wiki.debian.org/QEMU>
  - ☐ <https://www.linux-kvm.org/page/HOWTO>
  - ☐ <http://wiki.virtualsquare.org/#!/tutorials/vdebasics.md>
  - ☐ <https://wiki.debian.org/libvirt>
  - ☐ <https://libvirt.org/manpages/virsh.html>
  - ☐ [https://cloud-init.io/?\\_ga=2.115595270.1671114692.1642099236-687519757.1617810142](https://cloud-init.io/?_ga=2.115595270.1671114692.1642099236-687519757.1617810142)
  - ☐ <https://cloudinit.readthedocs.io/en/latest/>
  - ☐ Ansible: <https://elpuig.xeill.net/Members/vcarceler/asix-m06/a7>
-

---

---

# Virtualization

---

---

## Conceptes clau

- Para-virtualization.
- Full virtualization.
- Hypervisor
- Host / Guest
  
- Kvm
- Qemu
- Xen
- libvirt

Podeu consultar entre altres les següents fonts d'informació per conèixer el significat d'aquests conceptes clau:

- ☐ Wiki [Virtualization](#), [Hypervisor](#), [Kvm](#), [Qemu](#), [libvirt](#) ans [Xen](#).
- ☐ Publicació: Teoria\_Virtualizacio\_Contenedores.pdf, José Antonio Carrasco Díaz, I.E.S. Foco. Romero Vargas.

Conceptes clau:

- host: l'equip on està treballant l'usuari (si hi ha usuari), és a dir, el de l'alumne. En el host és on executem les ordres de virtualització i dins d'ell engegarem una a més màquines virtuals.
- guest: és la màquina virtual. El guest és la finestra (si es visualitza gràficament) on simulem tenir un ordinador executant un sistema operatiu.
- Tecla-sortida: usualment cal prémer ctr+alt per sortir de la finestra del guest i poder tornar a la del host. Una altra combinació usual és el control de la dreta.

Tipus d'imatges:

- Imatges iso.
- Imatges en pens USB.
- Imatges en particions físiques.
- Imatges .img fetes amb dd d'un device. De fet es poden muntar al loop i es tractaria del mateix que el cas anterior.
- Imatges residents en fitxers de imatges de disc tipus qcow2, ova, etc

# Instal·lació de les eines de virtualització Kvm/Qemu/Libvirt

## Requeriments

Verificar que la CPU té les extensions necessàries per fer la virtualització:

- Intel processor with the Intel VT and the Intel 64 extensions.
- AMD processor with the AMD-V and the AMD64 extensions.
- [svm](#) / [vmx](#)

```
$ grep -E "svm|vmx" /proc/cpuinfo  
$ lscpu | grep -E "svm|vmx"
```

## Instal·lació del software

1) Instal·lar el software de:

- @Virtualization
- libvirt
- qemu

Verificar que es disposa de virt-manager, qemu, libvirt

```
[fedora]$ rpm -qa | grep -E "qemu|libvirt|virt"  
[debian]$ dpkg -l | grep -E "qemu|virt"
```

```
$ sudo apt install qemu-system libvirt-daemon-system libvirt-clients
```

2) Activar el servei libvirt permanentment

```
$ sudo systemctl enable libvirt  
$ sudo systemctl start libvirt  
$ sudo systemctl is-enabled libvirt  
$ sudo is-active libvirt  
$ sudo status libvirt
```

## Configurar libvirt per ser executat com un usuari no privilegiat

Per tal de poder executar les eines de virtualització amb un usuari no privilegiat es configura libvirt fent:

- Cal assignar l'usuari al grup libvirt (creat en instal·lar el paquet)

- Modificar la configuració de libvirtd per permetre l'accés a través dels permisos de grup (/etc/libvirt/libvirtd.conf).
  - Cal descomentar la línia de la directiva: `unix_sock_group = "libvirt"`
  - Verificar la directiva: `unix_sock_rw_perms = "0770"`

Verificar l'existència del grup:

```
$ grep "libvirt" /etc/group
libvirt:x:984:

$ getent group libvirt
```

Assignar l'usuari al grup

```
$ sudo systemctl stop libvirtd

$ usermod -a -G libvirt $(whoami)
```

Modificar el fitxer de configuració `/etc/libvirt/libvirtd.conf`

```
unix_sock_group = "libvirt"

unix_sock_rw_perms = "0770"
```

Reiniciar el servei, pot ser que faci falta tancar la sessió gràfica i iniciar-la de nou per tal de que la pertinença al grup per part de l'usuari sigui efectiva:

```
$ systemctl start libvirtd

$ systemctl status libvirtd
```

Ara ja es pot iniciar virt-manager com un usuari no privilegiat:

```
$ virt-manager
```

Repàs de la configuració:

```
$ uname -a
Linux d02 5.10.0-9-amd64 #1 SMP Debian 5.10.70-1 (2021-09-30) x86_64 GNU/Linux

$ cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
```

```
$ dpkg -l virt*
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name                               Version                               Architecture
Description
+++-=====
```

Name	Version	Architecture	Description
ii virt-manager	1:3.2.0-3	all	desktop application for managing virtual machines
ii virt-viewer	7.0-2	amd64	Displaying the graphical console of a virtual machine
ii virtinst	1:3.2.0-3	all	utilities to create and edit virtual machines

```
$ dpkg -l | grep virt
```

```
ii gir1.2-libvirt-glib-1.0:amd64 3.0.0-1 amd64 GObject introspection files for the libvirt-glib library
ii gvfs:amd64 1.46.2-1 amd64 userspace virtual filesystem - GIO module
ii gvfs-backends 1.46.2-1 amd64 userspace virtual filesystem - backends
ii gvfs-common 1.46.2-1 all userspace virtual filesystem - common data files
ii gvfs-daemons 1.46.2-1 amd64 userspace virtual filesystem - servers
ii gvfs-fuse 1.46.2-1 amd64 userspace virtual filesystem - fuse server
ii gvfs-libs:amd64 1.46.2-1 amd64 userspace virtual filesystem - private libraries
ii libgovirt-common 0.3.7-2 all GObject-based library to access oVirt REST API (common files)
ii libgovirt2:amd64 0.3.7-2 amd64 GObject-based library to access oVirt REST API
ii libvirglrenderer1:amd64 0.8.2-5 amd64 Virtual GPU for KVM virtualization
ii libvirt-clients 7.0.0-3 amd64 Programs for the libvirt library
ii libvirt-daemon 7.0.0-3 amd64 Virtualization daemon
ii libvirt-daemon-config-network 7.0.0-3 all Libvirt daemon configuration files (default network)
ii libvirt-daemon-config-nwfilter 7.0.0-3 all Libvirt daemon configuration files (default network filters)
ii libvirt-daemon-driver-lxc 7.0.0-3 amd64 Virtualization daemon LXC connection driver
ii libvirt-daemon-driver-qemu 7.0.0-3 amd64 Virtualization daemon QEMU connection driver
ii libvirt-daemon-driver-vbox 7.0.0-3 amd64 Virtualization daemon VirtualBox connection driver
ii libvirt-daemon-driver-xen 7.0.0-3 amd64 Virtualization daemon Xen connection driver
ii libvirt-daemon-system 7.0.0-3 amd64 Libvirt daemon configuration files
ii libvirt-daemon-system-systemd 7.0.0-3 all Libvirt daemon configuration files (systemd)
ii libvirt-glib-1.0-0:amd64 3.0.0-1 amd64 libvirt GLib and GObject mapping library
ii libvirt0:amd64 7.0.0-3 amd64 library for interfacing with different virtualization systems
ii ovmf 2020.11-2 all UEFI firmware for 64-bit x86 virtual machines
ii python3-libvirt 7.0.0-2 amd64 libvirt Python 3 bindings
ii virt-manager 1:3.2.0-3 all desktop application for managing virtual machines
ii virt-viewer 7.0-2 amd64 Displaying the graphical console of a virtual machine
ii virtinst 1:3.2.0-3 all utilities to create and edit virtual machines
```

```
$ dpkg -l | grep qemu
```

```
ii ipxe-qemu 1.0.0+git-20190125.36a4c85-5.1 all PXE boot firmware - ROM images for qemu
ii libvirt-daemon-driver-qemu 7.0.0-3 amd64 Virtualization daemon QEMU connection driver
ii qemu 1:5.2+dfsg-11+deb11u1 amd64 fast processor emulator, dummy package
ii qemu-system-common 1:5.2+dfsg-11+deb11u1 amd64 QEMU full system emulation binaries (common files)
ii qemu-system-data 1:5.2+dfsg-11+deb11u1 all QEMU full system emulation (data files)
ii qemu-system-gui:amd64 1:5.2+dfsg-11+deb11u1 amd64 QEMU full system emulation binaries (user interface and audio support)
ii qemu-system-x86 1:5.2+dfsg-11+deb11u1 amd64 QEMU full system emulation binaries (x86)
ii qemu-utils 1:5.2+dfsg-11+deb11u1 amd64 QEMU utilities
```

---

---

# Laboratori de pràctiques

---

---

Per poder realitzar les pràctiques i exemples convé disposar localment i a través de xarxa d'imatges d'instal·lació, màquines virtuals del Clou i altres màquines.

Per instal·lar via HTTP

- ☐ Fedora32  
[https://dl.fedoraproject.org/pub/fedora/linux/releases/32/Everything/x86\\_64/os/](https://dl.fedoraproject.org/pub/fedora/linux/releases/32/Everything/x86_64/os/)
- ☐ Debian11  
<http://ftp.us.debian.org/debian/dists/Debian11.2/main/installer-amd64/>
- ☐ Ubuntu 20  
<http://archive.ubuntu.com/ubuntu/dists/focal/main/installer-amd64/>

Per instal·lar usant netinst

Es tracta d'imatges ISO d'instal·lació dels sistemes operatius que no contenen tot el software d'instal·lació i per tant caben en un CD ocupant al voltant de 500MB. El procediment d'instal·lació va a buscar el software als repositoris d'internet.

- ☐ Fedora 32 netinst 579M  
[https://archives.fedoraproject.org/pub/archive/fedora/linux/releases/30/Workstation/x86\\_64/iso/Fedora-Workstation-netinst-x86\\_64-30-1.2.iso](https://archives.fedoraproject.org/pub/archive/fedora/linux/releases/30/Workstation/x86_64/iso/Fedora-Workstation-netinst-x86_64-30-1.2.iso)
- ☐ Debian 11 Bullseye netinst amd64 396M  
<https://cdimage.debian.org/debian-cd/current/amd64/iso-cd/debian-11.1.0-amd64-netinst.iso>
- ☐ Centos 8.4 boot 723M  
[http://mirror.airenetworks.es/CentOS/8.4.2105/isos/x86\\_64/CentOS-8.4.2105-x86\\_64-boot.iso](http://mirror.airenetworks.es/CentOS/8.4.2105/isos/x86_64/CentOS-8.4.2105-x86_64-boot.iso)

Per instal·lar usant un DVD complert

Algunes versions de les distribucions encara es lliuren en format DVD complert que conté tot el software a instal·lar. Es tracta, però, del software en el moment de generar el DVD i



per tant queda desferrat respecte les actualitzacions. Un cop instal·lat el sistema és recomanable fer la actualització (per això és més pràctic usar les instal·lacions netinst).

- ☐ Ubuntu 20

<https://ubuntu.com/download/desktop/thank-you?version=20.04.3&architecture=amd64>

## Imatges Cloud

Màquines virtuals fetes per les propies distribucions, són petites i lleugeres per poder-les posar en funcionament directament. Poden proporcionar-se en varis formats com per exemple: qcow2 (quemu), raw (en cru), vagrant, per a vmware, per a virtualbox, etc.

- ☐ Fedora Cloud 32 Repo: /pub/fedora/linux/releases/32/Cloud/x86\_64/images  
[https://dl.fedoraproject.org/pub/fedora/linux/releases/32/Cloud/x86\\_64/images/Fedora-Cloud-Base-32-1.6.x86\\_64.qcow2](https://dl.fedoraproject.org/pub/fedora/linux/releases/32/Cloud/x86_64/images/Fedora-Cloud-Base-32-1.6.x86_64.qcow2)

Fedora images have no root/user access. Image should be modified by virt-sysprep to set the root password: virt-sysprep belongs to the libguestfs-tools package:

```
$ sudo virt-sysprep -a Fedora-name-image.qcow2 --root-password password:newpasswd
```

- ☐ Debian 11 Bullseye nocloud 297M  
<https://cloud.debian.org/images/cloud/bullseye/latest/debian-11-nocloud-amd64.qcow2>
- ☐ Ubuntu Cloud Images 20.10 (usb image .IMG 549M)  
<https://cloud-images.ubuntu.com/releases/groovy/release/ubuntu-20.10-server-cloudimg-amd64.img>

## Altres imatges

- ☐ Alpine  
<https://alpinelinux.org/downloads/>
- ☐ DSL Damn Small Linux  
Wiki en [https://en.wikipedia.org/wiki/Damn\\_Small\\_Linux](https://en.wikipedia.org/wiki/Damn_Small_Linux)  
[http://distro.ibiblio.org/damnsmall/release\\_candidate/dsl-4.11.rc1.iso](http://distro.ibiblio.org/damnsmall/release_candidate/dsl-4.11.rc1.iso)
- ☐ Gparted  
<https://downloads.sourceforge.net/gparted/gparted-live-1.3.1-1-amd64.iso>
- ☐ Super Grub Disk  
[https://sourceforge.net/projects/supergrub2/files/2.04s1/super\\_grub2\\_disk\\_2.04s1/super\\_grub2\\_disk\\_hybrid\\_2.04s1.iso/download](https://sourceforge.net/projects/supergrub2/files/2.04s1/super_grub2_disk_2.04s1/super_grub2_disk_hybrid_2.04s1.iso/download)

## Tricks: users & passwords

### Debian:

- ☐ Debian nocloud image qcow2: User [root](#), no password.
- ☐ Debian Live: [user / live](#)
- ☐ Debian AWS EC2:

### Fedora:

- ☐ Fedora Cloud images: usar virt-sysprep per assignar password.
- ☐ Fedora Live:
- ☐ Fedora AWS EC2

### Ubuntu:

- ☐ Ubuntu Cloud images:
- ☐ Ubuntu Live:
- ☐ Ubuntu AWS EC2

### AWS EC2 images

- ☐ Debian:
- ☐ Fedora:
- ☐ Centos:
- ☐ AWS AMI2: [ec2-user](#)
- ☐ Windows:

### Centos

- ☐
- ☐

### Windows

- ☐
- ☐

---

---

# Eines virt

---

---

- ☐ Virt-manager
- ☐ Virt-install
- ☐ Virt-shell
- ☐ qemu-xxxx
- ☐ Virt-clone
- ☐ Virt-image
- ☐ Virt-convert
- ☐ Virt-top
- ☐ Virt-viewer

virt-alignment-scan	virt-df	virt-index-validate	virt-make-fs
virt-qemu-run	virt-tar-in		
virt-builder	virt-diff	virt-inspector	virt-manager
virt-rescue	virt-tar-out		
virt-builder-repository	virt-edit	virt-install	virtnetworkd
virt-resize	virtualbox		
virt-cat	virt-filesystems	virtinterfaced	virtnodedevd
virtsecret	virtualboxvm		
virt-clone	virt-format	virtlockd	virtnwfilterd
virt-sparsify	virt-viewer		
virt-copy-in	virtfs-proxy-helper	virt-log	
virt-pki-validate	virtstoraged	virt-xml	
virt-copy-out	virt-get-kernel	virtlogd	virtproxyd
virt-sysprep	virt-xml-validate		
virt-customize	virt-host-validate	virt-ls	virtqemud
virt-tail			

qemu-edid	qemu-img	qemu-keymap	qemu-nbd
qemu-system-i386	qemu-trace-stap	qemu-ga	qemu-io
qemu-kvm	qemu-pr-helper	qemu-system-x86_64	

---

## Virt-manager

---

- ☐ Creació d'una nova màquina virtual
- ☐ Eliminar una màquina virtual
- ☐ Configuració hardware

### Crear una nova màquina virtual:

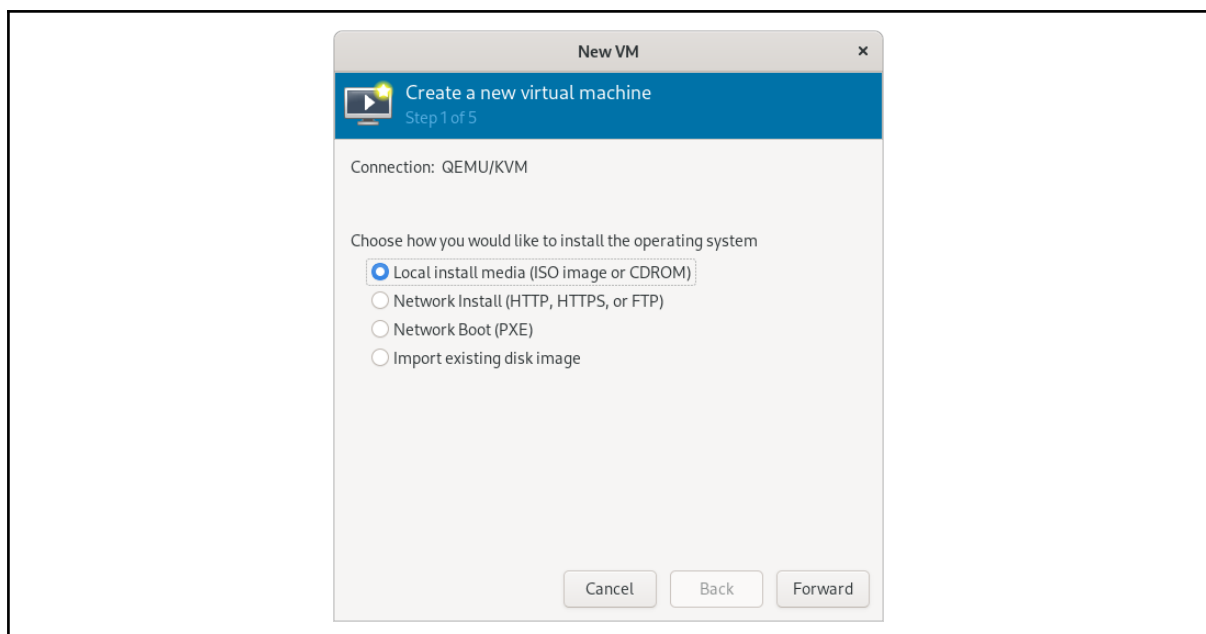
#### Opcions de creació d'una nova VM

##### Opcions (1)

- Instal·lar d'un CDROM
- Instal·lar de xarxa via HTTP/HTTPS/FTP
- Per xarxa via PXE
- Importar una imatge de disk existent

##### Opcions(2)

- Instal·lar d'un CDROM
- Instal·lar de xarxa via HTTP/HTTPS/FTP
- Importar una imatge de disk existent
- Manual Install

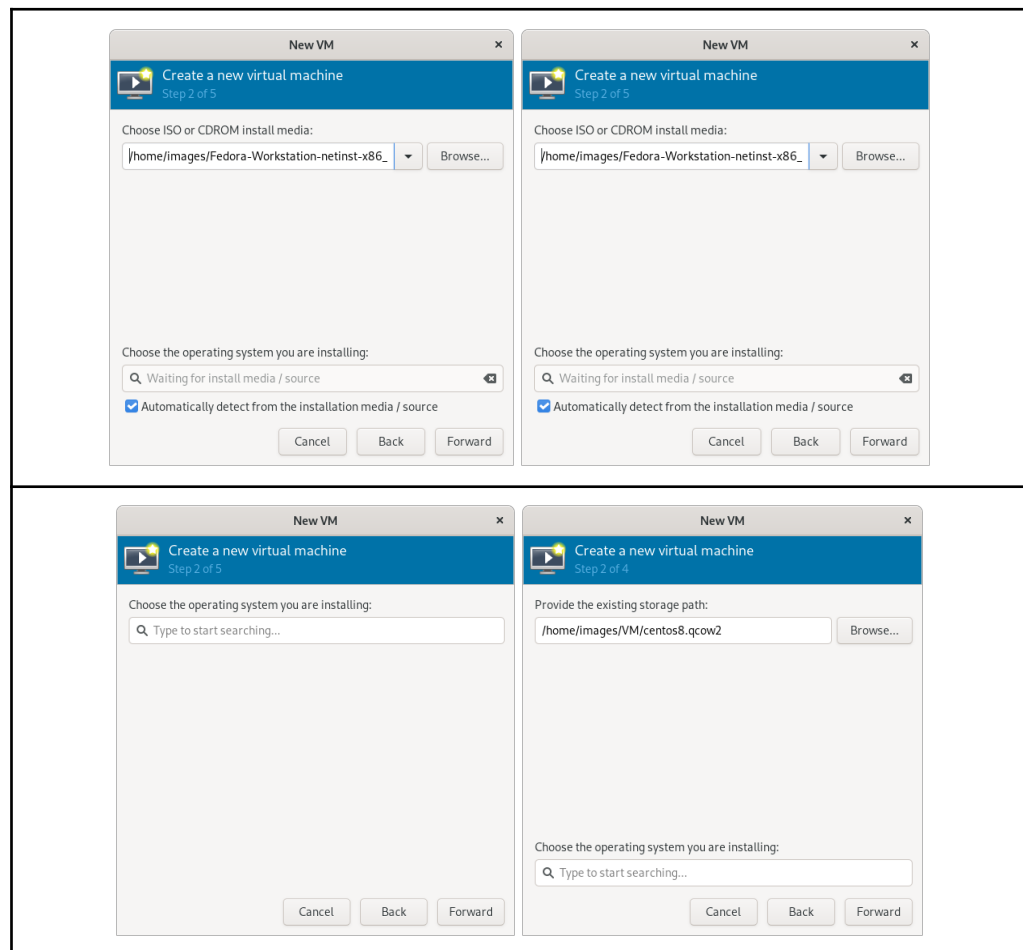


## Passos de creació de la VM:

### 1. New VM

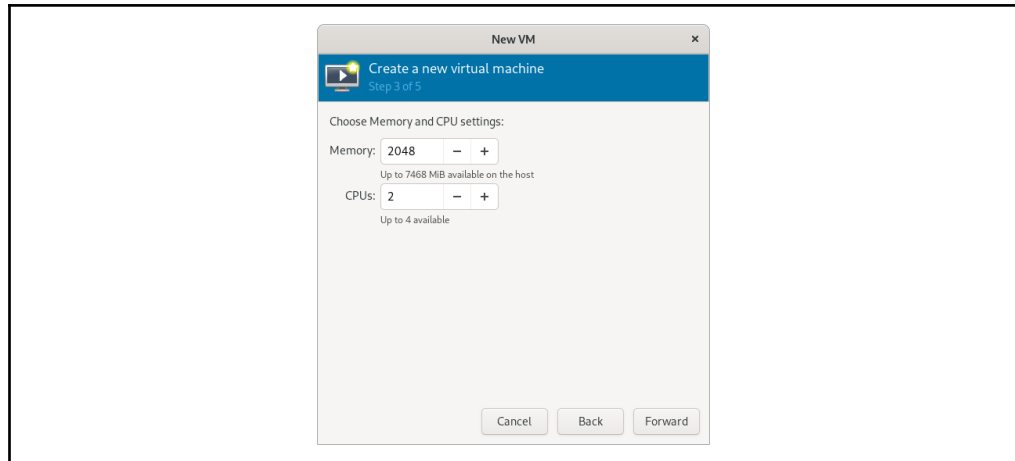
Segons el mètode d'instal·lació seleccionat cal indicar la iso a utilitzar o la URL, etc. Si no es detecta automàticament el tipus de sistema operatiu es pot indicar manualment seleccionant-ne un del desplegable.

- Si s'ha escollit instal·lar via ISO virt-manager lliga automàticament la ISO com a dispositiu CD/DVD únicament per a la primera arracada.
- <pendent> Network boot
- Si s'utilitza local install caldrà que sigui l'usuari qui configuri el mecanisme d'instal·lació, si és usant un CD/DVD caldrà que en fer el reboot pensi a modificar el Boot Order, també si ha triat instal·lar via PXE.



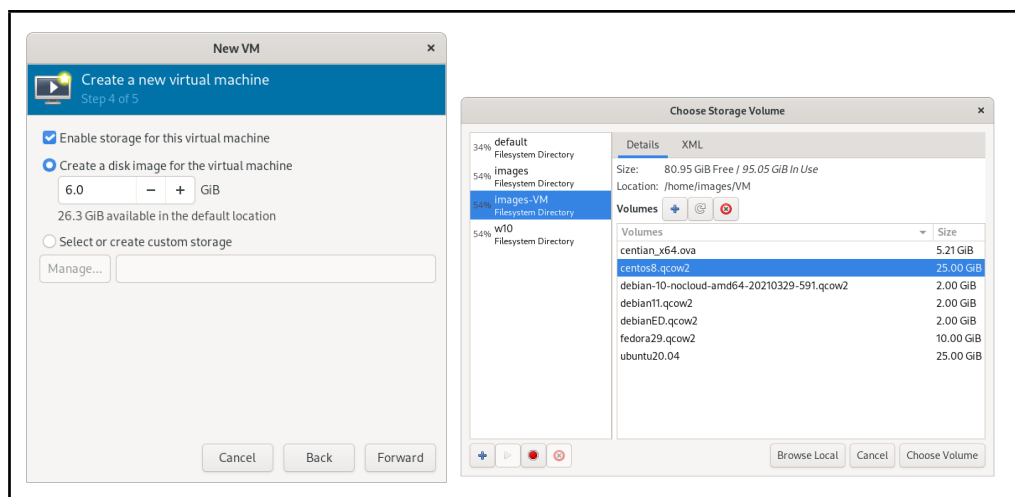
### 2. Memory and CPU settings

- Determinar la quantitat de memòria que tindrà la VM i quantes CPUs. No usar totes les disponibles o la màquina host es quedarà sense recursos.



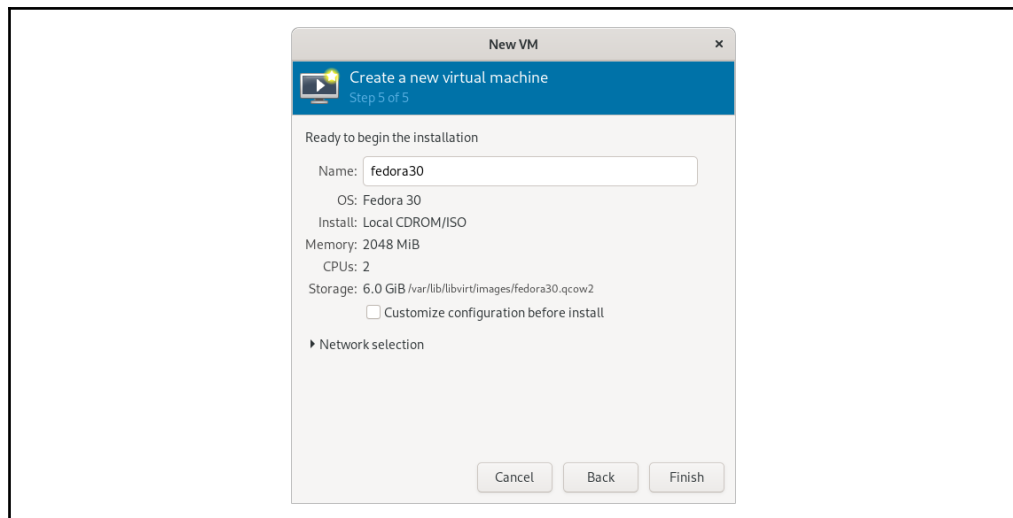
### 3. Storage

- Determinar si cal o no emmagatzemament per a la VM. Per exemple si es tracta d'una màquina que executa un sistema Live no requereix de HD.
- Si cal crear un (o més) HD d'emmagatzemament es pot:
  - Crear-ne un de nou indicant la mida en GiB.
  - Seleccionar una imatge o partició d'emmagatzemament ja existent.
- **qcow2**: aquest és el format per defecte que s'utilitza, un format Copy On Write (per capes) dinàmic (creix a mida que es necessita).
- **Atenció**: observar clarament on es genera el fitxer de la imatge (qcow2) per eliminar-lo quan s'elimini la VM, és important fer neteja o ràpidament el disc queda ple!. Usualment: `/var/lib/libvirt/images/nom.qcow2`.



### 4. Ready to begin

- Un cop seleccionades totes les opcions clicant a Finish es genera la VM.
- Observar la ruta on es genera el disc de emmagatzemament: `/var/lib/libvirt/images/nom.qcow2`.
- Observar l'opció de *"Custom configuration before install"* que permet configurar el hardware de la VM abans de crear-la / engegar-la.

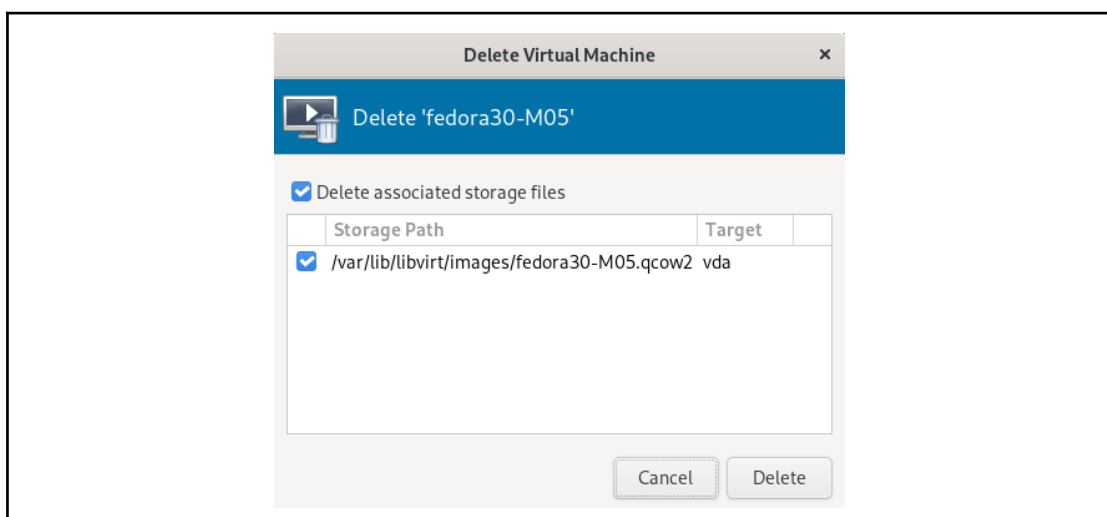


## 5. Executar la màquina virtual, engegar-la

- Un cop feta la creació s'engega automàticament.
- A vegades pot sol·licitar si es vol activar la xarxa virtual indicant que actualment està desactivada (yes).
- Si la màquina està parada es pot engegar amb el botó de play.
- Es pot configurar el hardware de la màquina amb la màquina parada prement el botó de configuració.

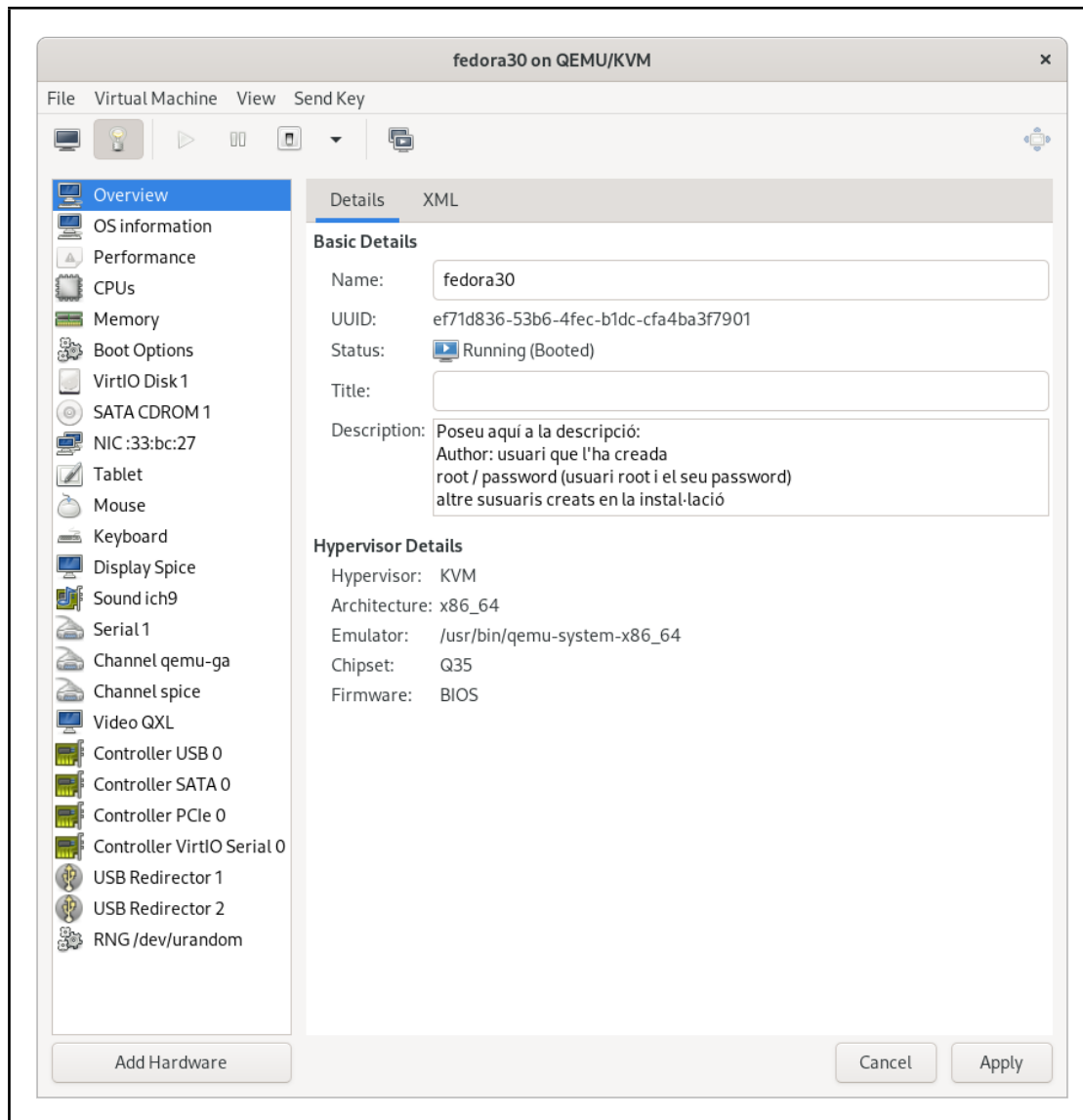
## Eliminar una màquina virtual

- Per eliminar una màquina virtual simplement seleccionar-la del llistat de màquines i escollir Delete.
- **Atenció:** decidir si es vol eliminar el storage o no. Per defecte a l'aula es recomana que si sinó es van deixant discs virtuals que omplen el disc dir i impedeixen de funcionar correctament el sistema operatiu.



## Configuració hardware d'una VM

Es pot modificar la configuració hardware d'una VM simulant la modificació que es faria en un ordinador si se li treuen, afageixen o modifiquen components, com per exemple targetes de xarxa, unitats de disc, de CD/DVD, etc.



## Pràctica-1 Virt-manager

- 01 Crear una VM amb un disc en blanc i una iso netinst de Debian i realitzar una instal·lació minimal.



- 02 Crear una VM amb un disc en blanc i instal·lar Fedora32 usant una url de HTTP com a mecanisme d'instal·lació.
- 03 Crear una VM amb un disc en blanc i instal·lar un Ubuntu complet partint de la url HTTP.
- 04 Crear una VM partint d'una imatge ja existent: debian qcow2. Assignar-li memòria, cpu, nom, descripció.
- 05 Crear una VM partint d'una imatge ja existent tipus DSL com per exemple un alpine. Assignar-li una unitat de disc extra.
- 06 Crear una VM que engega una Live i té assignat espai d'emmagatzemament amb un disc extra.
- 07 Crear una VM basada en la imatge Cloud de Fedora32. Modificar-la amb virt-sysprep per assignar-li a root el password habitual.
- 08 Generar amb dd una imatge raw de la partició 'minilinux' de l'aula. Generar una VM que utilitzi aquesta imatge raw.

---

## Virt-install

---

### Descripció

Amb l'eina virt-install es poden **crear màquines virtuals** des de l'entorn de comandes, sense usar l'eina gràfica virt-manager o usant-la. Crear la màquina tant pot voler dir crear una màquina partint d'una imatge ja existent com crear una màquina fent tot el procés d'instal·lació (interactivament o d manera desatesa).

#### Característiques:

- Permet crear màquines virtuals de manera desatesa, sense menús, des d'una sola comanda de text.
- La màquina virtual pot disposar d'un o més discs virtuals, devices usb i pci del host amfitrió, audio i interfícies de xarxa.
- La instal·lació del sistema operatiu en el guest es pot fer localment via cd, dvd, imatges iso o bé per xarxa via http, ftp, nfs, PXE, etc.
- Si el guest és una instal·lació desatesa (kickstart) permet crear el guest tot de manera desatesa.
- El guest s'executa com un procés en el host. No cal visualitzar-lo
- gràficament per tenir-lo engegat. Existeixen diversos mecanismes per 'visualitzar' una màquina virtual. Es pot fer via:
  - Hipervisors: com qemu i xen.
  - VNC (Virtual Network Computing), un visualitzador d'escriptoris remots.

### Opcions de configuració

Virt-install té multitud d'opcions de configuració que permeten configurar fins al mínim detall. Es recomana consultar la seva ajuda i els exemples de la pàgina man.

```
$ virt-install --help
```

#### Exemple d'opcions de configuració:

##### Opcions generals:

- --name=NAME permet posar un nom identificador que ha de ser únic al guest.
- --ram=MEM permet indicar la quantitat de memòria en Mbytes del guest.
- --vcpus=CPUS permet indicar la quantitat de cpus del guest.
- --os-type=OS-TYPE permet indicar el tipus de sistema operatiu del guest.
- --os-variant=OS-VARIANT permet indicar el tipus concret de sistema operatiu.
- --host-device=HOST\_DEV permet adjuntar un device físic del host en el guest.

Opcions referents al mètode d'instal·lació del sistema operatiu en el guest:

- `--cdrom=CDROM` indica que cal usar el device indicat (ex: `/dev/sr0`) o el fitxer indicat com a font de la instal·lació del sistema operatiu en el guest.
- `--location` permet indicar una URL d'on obtenir la font de la instal·lació.
- `--pxe` permet fer la instal·lació via PXE (requereix bridge de xarxa).
- `--import` obliga a saltar-se el pas de la instal·lació i utilitza directament el device o fitxer indicat com a virtual-disk (opcions `--disk` o `--file`) com a sistema operatiu fet.
- `--livecd` indica que el dispositiu d'instal·lació és un livecd.

• Opcions d'emmagatzemament:

- `--disk=DISKOPTIONS` permet descriure tot allò referent a un device de disc virtual que el guest ha de tenir.
  - El virtual-disk pot ser un fitxer o un dispositiu real. També pot ser local o compartit per xarxa.
  - Si el disc ja existeix s'utilitza. Si no existeix es crea de nou. En aquest cas és obligatori indicar la mida.
  - Els fitxers d'imatges virtuals de disc (virtual-disk) poden ser de diferents formats. Els identificats actualment són: raw, qcow2, qcow, vdi, vmsdk, etc.

Opcions de configuració de xarxa:

- `--network=NETWORK,opcions` permet descriure el o els dispositius de xarxa a usar per el guest. És a dir, descriu com serà la xarxa que lliga el guest amb el host i descriu el tipus d'interfície de xarxa virtual del guest.
- `NETWORK` pot prendre un dels tres valors següents que descriuen el 'tipus de xarxa' que s'estableix:
  - `bridge=BRIDGE` indica que el guest usarà un bridge conjuntament amb el host. Això significa que el guest té visibilitat a la xarxa externa del host, té una cara a la interfície de xarxa externa. Només es pot usar aquesta opció si el host té connexió de xarxa amb cable. Aquesta opció permet que els guest siguin servidors de xarxa i que s'hi accedeixi des de l'exterior. Per usar un bridge cal configurar-lo anteriorment en el host.
  - `network=NAME` crea una xarxa virtual de nom NAME. Les xarxes virtuals són xarxes internes entre el host i les màquines virtuals. Vàries màquines virtuals poden pertànyer a la mateixa xarxa interna (usar el mateix NAME).
  - Si no s'indica un nom de xarxa la xarxa usada s'anomena 'default'. El host pot fer NAT per proporcionar sortida a l'exterior dels guest de la xarxa interna. Els guest no són accessibles des de l'exterior. No poden fer de servidors. Cal usar aquesta opció si el host usa NetworkManager o usa wireless.
  - `user` permet usar opcions de xarxa molt limitades, s'utilitza amb `qemu` quan l'executa un usuari no privilegiat.
- Es poden indicar opcions que permetin descriure el tipus d'interfície de xarxa virtual a usar. ◦ Si el host té un bridge amb alguna interfície física adjunta s'utilitza el bridge. Si no s'utilitza la xarxa virtual 'default'.

Opcions de configuració gràfiques:

- `--vnc` activa en el guest un servidor vnc de manera que es pot visualitzar el guest a través d'una connexió vnc client des de qualsevol equip, sigui el host o un altre.
- `--vnc-port=xx` permet indicar el port per on escolta el servei vnc del guest.
- `--sdl` genera una consola virtual visible en el host. Si es tanca es tanca el guest.

Opcions específiques només per a Full-Virtualization:

- `--sound` permet disposar en el guest del device de so del host.

Tipus de virtualització:

- `--hvm` indica que cal usar Full-Virtualization.
- `--paravirt` indica que cal usar paravirtualització en lloc de full-virtualization.

Indicar el hipervisor a usar: `--connect=HIPERVISOR` (només si volem usar un diferent del visor per defecte):

- xen
- qemu:///system
- qemu:///session

## Exemples del Man

Els següents exemples són extrets de la documentació de *man virt-install*.

```
#1
The simplest invocation to interactively install a Fedora 29 KVM VM with recommended
defaults. virt-viewer(1) will be launched to graphically interact with the VM install

# sudo virt-install --install fedora29
```

```
#2
Similar, but use libosinfo's unattended install support, which will perform the fedora29
install automatically without user intervention:

# sudo virt-install --install fedora29 --unattended
```

```
#3
Install a Windows 10 VM, using 40GiB storage in the default location and 4096MiB of ram,
and ensure we are connecting to the system libvirtd instance:

# virt-install \
    --connect qemu:///system \
    --name my-win10-vm \
    --memory 4096 \
    --disk size=40 \
    --os-variant win10 \
    --cdrom /path/to/my/win10.iso
```

```
#4
Install a CentOS 7 KVM from a URL, with recommended device defaults and default required
storage, but specifically request VNC graphics instead of the default SPICE, and request
8 virtual CPUs and 8192 MiB of memory:

# virt-install \
    --connect qemu:///system \
    --memory 8192 \
    --vcpus 8 \
    --graphics vnc \
    --os-variant centos7.0 \
    --location http://mirror.centos.org/centos-7/7/os/x86_64/
```

```
#5
Create a VM around an existing debian9 disk image:

# virt-install \
    --import \
    --memory 512 \
    --disk /home/user/VMs/my-debian9.img \
    --os-variant debian9
```

```
#6
Start serial QEMU ARM VM, which requires specifying a manual kernel.
```

```
# virt-install \
    --name armtest \
    --memory 1024 \
    --arch armv7l --machine vexpress-a9 \
    --disk /home/user/VMs/myarmdisk.img \
    --boot \
    --kernel=/tmp/my-arm-kernel,initrd=/tmp/my-arm-initrd,dtb=/tmp/my-arm-dtb,kernel_args="console=ttyAMA0 rw root=/dev/mmcblk0p3" \
    --graphics none
```

```
#7
Start an SEV launch security VM with 4GB RAM, 4GB+256MiB of hard_limit, with a couple of virtio devices:
Note: The IOMMU flag needs to be turned on with driver.iommu for virtio devices. Usage of --mementune is currently required
because of SEV limitations, refer to libvirt docs for a detailed explanation.
```

```
# virt-install \
    --name foo \
    --memory 4096 \
    --boot uefi \
    --machine q35 \
    --mementune hard_limit=4563402 \
    --disk size=15,target.bus=scsi \
    --import \
    --controller type=scsi,model=virtio-scsi,driver.iommu=on \
    --controller type=virtio-serial,driver.iommu=on \
    --network network=default,model=virtio,driver.iommu=on \
    --rng driver,iommu=on \
    --memballoon driver.iommu=on \
    --launchSecurity sev
```

## Pràctica-2 Virt-install

- 01 Crear una VM partint d'una imatge ja existent: debian qcow2. Assignar-li memòria, cpu, nom, descripció.
- 02 Crear una VM partint d'una imatge ja existent: debain o DSL o Alpine. Assignar-li també una unitat de disc extra.
- 03 Crear una VM que engega una Live i té assignat espai d'emmagatzemament amb un disc extra.
- 04 Generar amb dd una imatge raw de la partició 'minilinux' de l'aula. Generar una VM que utilitzi aquesta imatge raw.
- 05 Crear una VM amb un disc en blanc i una iso netinst de Debian i realitzar una instal·lació minimal.
- 06 Crear una VM amb un disc en blanc i una iso netinst de fedora i realitzar una instal·lació desatesa.

- 07 Generar un USB d'instal·lació de Debian planxant la so d'instal·lació al USB. Crear una VM amb un disc en blanc que utilitzi el USB real com a font de la instal·lació.
- 08 Crear una VM basada en la imatge Cloud de Fedora32. Modificar-la amb virt-sysprep per assignar-li a root el password habitual.
- 09 Crear una VM Windows partint de les màquines prefabricades de prova que proporciona el propi Windows.

## Exemples d'exercicis antics

```
#1
# Crear una màquina virtual d'un LiveCD de Fedora: imatge .iso
$ virt-install --name=Live01 --ram=512 --nodisks --livecd --cdrom
/fc9/vms/Fedora-14-i686-Live-Desktop.iso

#2
# Crear una màquina virtual d'un LiveCD de Fedora: imatge d'un device USB
$ virt-install --name=Live03 --ram=512 --import --disk path=/dev/sdb1 &

#3
# Crear una màquina virtual partint d'un virtual-device ja existent
$ virt-install --name=win01 --ram=512 --import --disk path=win2003s.img &
$ virt-install --name=lin01 --ram=512 --import --disk path=minif13-pxe-vi.img &

#4
# Crear una màquina virtual d'una imatge dd d'una partició
$ virt-install --name=lin02 --ram=512 --import --disk path=mini-linux.f13.img &

#5
# Crear una imatge virtual a partir d'un device
$ virt-install --name=dev01 --ram=512 --import --disk path=/dev/sda2 &
$ virt-install --name=dev02 --ram=512 --import --disk path=/dev/sda &
```

```
#6
# Crear un virtual-disk com a emmagatzemament per a un d'un Live CD
# virt-install --name=Live01 --ram=512 --livecd --cdrom Fedora-14-i686-Live-Desktop.iso
--disk path=free.img,size=1 &

#7
# Crear un virtual-disk nou on fer una instal·lació a partir d'una imatge Live
# virt-install --name=Live01 --ram=512 --livecd --cdrom Fedora-14-i686-Live-Desktop.iso
--disk path=fedora.img,size=2 &

#8
# Crear una màquina virtual amb un virtual-disk en blanc per instal·lar-hi un Fedora via
xarxa.
# virt-install --name Linux01 --ram 512 --disk path=fedora3g.img,size=3 --location
http://download.fedora.redhat.com/pub/fedora/linux/releases/14/Fedora/i386/os/ &

#9
# Crear una màquina virtual usant un virtual-disk ja existent
# virt-install --name Linux02 --ram 512 --disk path=fedora.img --import
```



---

## Virt-shell

---

```
$ virsh list [--all]
$ virsh dominfo vm01
$ virsh domstate vm01
$ virsh start vm01
$ virsh restart vm01
$ virsh shutdown vm01
$ virsh destroy vm01
```

```
$ virsh suspend vm01
$ virsh resume vm01
$ virsh screenshot
$ virsh vcpuinfo
$ virsh vncdisplay
```

```
$ virsh create xml.file
$ virsh define xml.file
$ virsh undefine vm01
$ virsh dumpxml xml.file
$ virsh save vm01
$ virsh restore vm01
```

```
$ virsh attach-device
$ virsh attach-disk
$ virsh attach-interface
$ virsh net-info
$ virsh net-list
$ virsh iface-list
$ virsh iface-bridge
```

```
$ virsh snapshot create
$ virsh snapshot list
```



```
$ virsh snapshot list
$ virsh snapshot-revert
$ virsh snapshot-delete
```

## Pràctica-3 Virsh

- 01 Proveu cada una de les ordres de les taules anteriors i anoteu quin és el seu significat.
- 02 Escriu el conjunt d'ordres necessaries per:
  - engegar una VM ja existent
  - Llistar la informació de la màquina
  - Llistar la configuració XML de la màquina
  - mostrar un screenshot de la seva pantalla
  - Accedir a la imatge usant VNC
  - Fer una pausa de l'execució de la màquina
  - Reanudar la màquina i fer-ne un restart
  - Llistar la informació de xarxa de la màquina.
  - Aturar la màquina
  - Destruir-la.
- 03 Engegar una VM i treballar amb snapshots:
  - Crear un snapshot
  - Fer modificacions a la màquina
  - Crear un segon snapshot.
  - Fer modificacions fatals a la màquina.
  - Revertir la màquina a l'estat corresponent al primer snapshot.

---

## qemu

---

Amb les eines de qemu es poden visualitzar les VM de manera ràpida sense engegar virt-manager.

qemu-edid	<a href="#">qemu-img</a>	qemu-keymap	qemu-nbd
qemu-system-i386	qemu-trace-stap		
qemu-ga	qemu-io	qemu-kvm	qemu-pr-helper
<a href="#">qemu-system-x86_64</a>			

### qemu / qemu-system-x86\_64

Les options de configuració més usals són:

- Memòria
- Numero de cpus
- Dispositius de disc i de cdrom
- Dispositius de xarxa
- Ordre de precedència de l'arrancada.

Exemples antics:

```
#1
$ qemu -cdrom brutalix_3.0_Beta.iso -vga std -m 512 &

#2
$ qemu -hda /dev/sdb1 -vga std -m 512 &

#3
$ qemu -hda /dev/sda6 -vga std -m 512 &

#4
$ qemu -hda /dev/sda -vga std -m 512 &

#5
$ qemu -hda mini-linux.img -vga std -m 512 &

#6
$ qemu -cdrom winXP_desatesa.iso -hda zero.img -vga std -m 512 &

#7
$ qemu -cdrom brutalix_3.0_Beta.iso -hda zero.img -hdb /dev/sda6 -vga std -m 512 &

#8
$ qemu -cdrom brutalix_3.0_Beta.iso -hda rawDisc1.img -hdb qcow2Disc1.img -vga std -m
```

## qemu-img

Eina per crear i gestionar imatges, permet entre altres:

- Crear imatges de disk.
- Modificar imatges.
- Convertir formats.
- Xequeig d'imatges.
- Mostrar informació de les imatges.
- Snapshots. Instantànies de les imatges.

Exemples antics:

```
#1
$ qemu-img create -f raw rawDisc1.img 500M

#2
$ qemu-img info rawDisc1.img

#3
$ qemu-img create -f qcow2 qcow2Disc1.img 500M
```

Documentació:

- Qemu Read the docs: [qemu-img](#)
- Man page ([man](#))
- Qemu-img for [windows](#)

## Pràctica-4 qemu / qemu-img

- 01 xx
- Xxx
- Xxx

---

Virt-clone

---

---

Virt-image

---

---

Virt-convert

---

---

Virt-top

---

---

Virt-viewer

---

---

Virt-xml-validate

---

---

---

## Configuracions de xarxa

---

---

☐ Default network (NAT)

☐ Private (named)

☐ Isolated

☐ Bridge

- KVM Networking [Documentation](#)

---

Default network (NAT)

---

---

Private (named)

---

---

Isolated

---

---

## Bridge

---

Funcionament d'un bridge:

- Crear una interfície bridge
- Afegir una (o més) interfície física al bridge
- Configurar el bridge, en especial perquè NetworkManager no fastiguegi!

Ordres relacionades:

```
# Debian

apt-get install bridge-utils
brctl show

ip link add
```

Documentació:

- How to use bridged networking with libvirt and KVM  
<https://linuxconfig.org/how-to-use-bridged-networking-with-libvirt-and-kvm>
- Create and Configure Bridge Networking For KVM in Linux ([computingforgeeks](https://www.computingforgeeks.com/create-and-configure-bridge-networking-for-kvm-in-linux/))
- [KVM Networking](https://www.computingforgeeks.com/kvm-networking/)

## Crear / Configurar bridge en Debian

Procediment:

- Crear la interfície bridge br0
- Afegir la interfície eth0 (eno1) al bridge. En aquest punt pot perdre la ip i perdre la configuració.
- Editar /etc/network/interfaces
  - Modificar l'entrada de eth0 (eno1) per que sigui configurada manualment i no a través de network manager (*iface eno1 inet manual*).
  - Afegir l'entrada del bridge, que s'activi automàticament en iniciar el sistema (*auto br0*).
  - També que rebi la configuració via dhcp (*iface br0 inet dhcp*)
  - I finalment indicar que el bridge està format per la interfície eno1 (*bridge\_ports eno1*). Reiniciar el servei de xarxa
- Verificar l'estat del bridge
- Reiniciar el host per verificar que en successives arrancades el sistema utilitza el bridge. **Atenció** que això generarà que a l'escriptori a la zona de notifikacions (dalt a la dreta) mostri que la xarxa està unmanaged, perquè el networkManager no controla la eno1.

```
apt-get install bridge-utils
```

```
ip link add br0 type bridge  
ip lnk set eno1 master br0  
ip link show master br0
```

```
cp /etc/network/interfaces /etc/network/myinterfaces.bk
```

```
cat /etc/network/interfaces  
    auto lo  
    iface lo inet loopback  
  
    iface eno1 inet manual  
  
    auto br0  
    iface br0 inet dhcp  
        bridge_ports eno1  
        bridge_stp off
```

```
systemctl restart networking  
systemctl status networking
```

```
brctl show
```



---

---

## Instal·lacions

---

---

☐ Fedora-32

☐ Debian-11

☐ Ubuntu-20

---

## Fedora-32

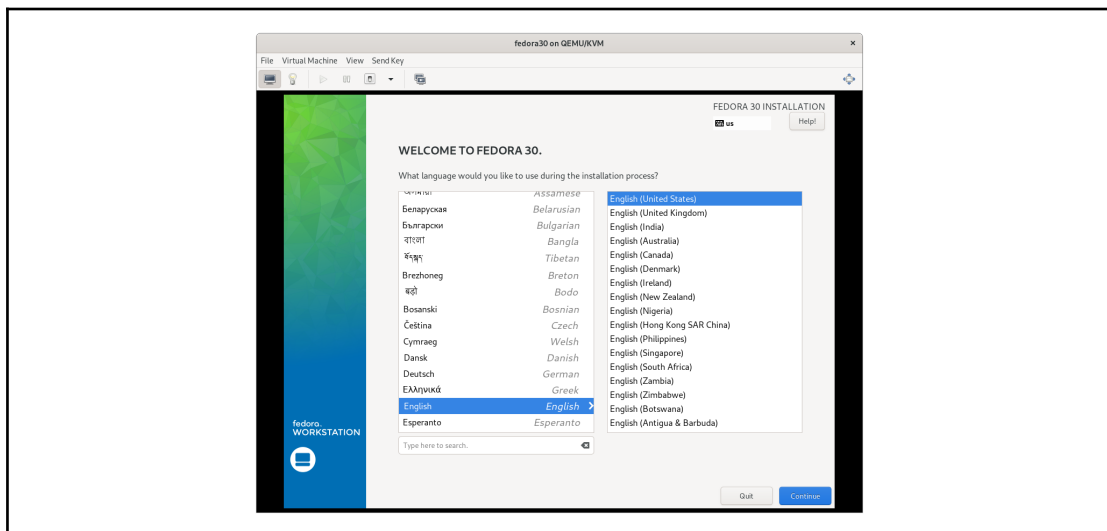
---

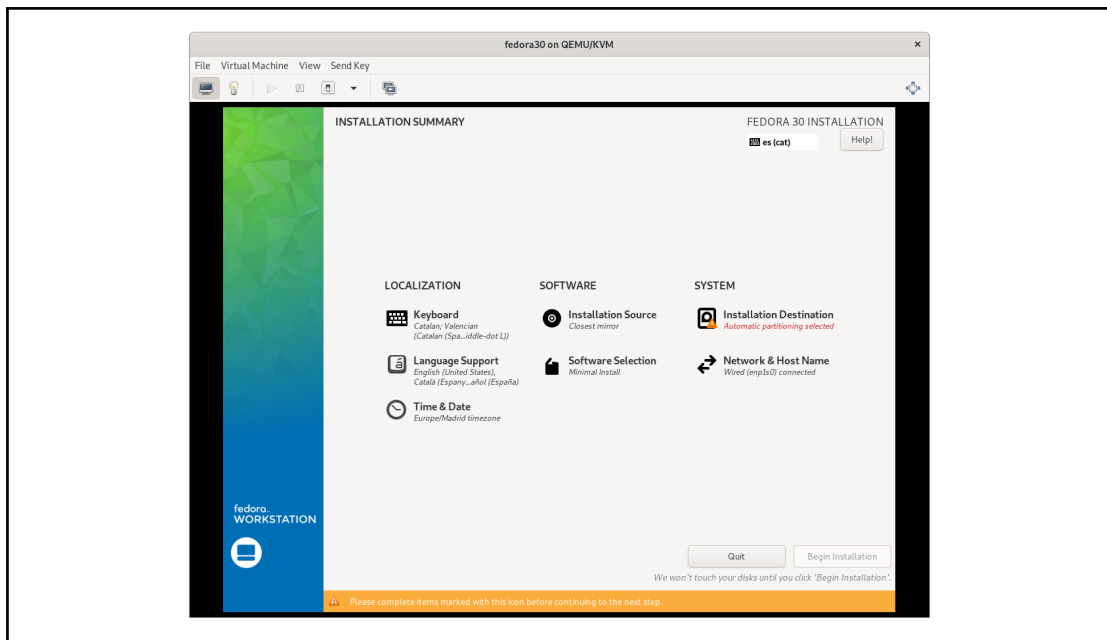
### Layout per fer particions:

- Tot en una sola, sense swap
- Partició arrel (/) i partició swap.
- Recomanat: /boot, / (arrel), swap
- Amb homes: /boot, / (arrel), /home, swap
- Altres punts de muntatge: /boot, /home, /var, /run, etc

### Instal·lació Fedora-32

- Components generals: installation-language, keyboard, language, source (closest mirror), software (minimal)
- Particionat: automàtic
- Característiques: 325 paquets, 5-10 min, 300MiB
- reboot, atenció a eliminar ara el device de CD, ja no cal, però no eliminar el fitxer .iso associat.
- Ordres a verificar: loadkeys es, df -h, lsbl, mount

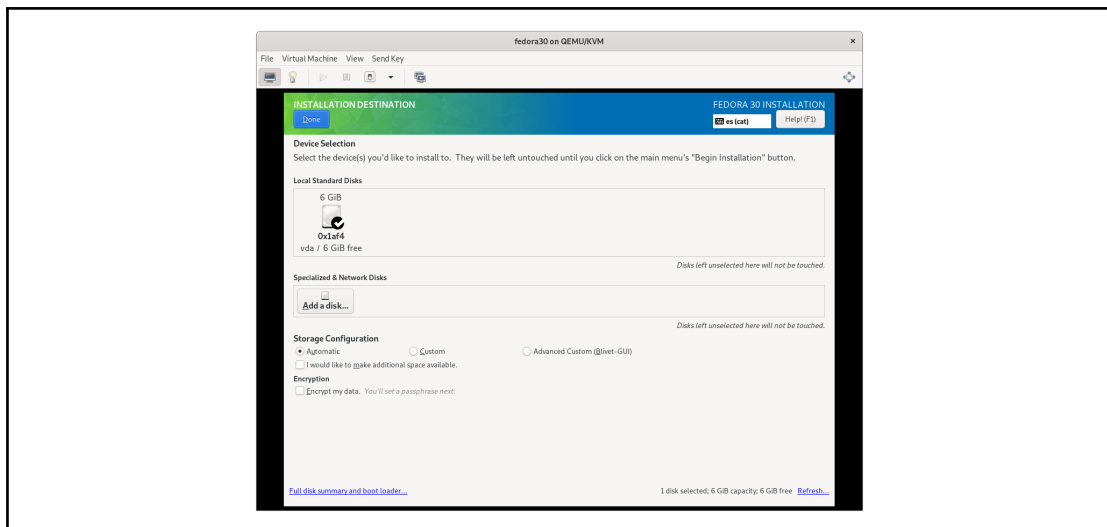




Opcions de particionat de Fedora-32: **automàtic**

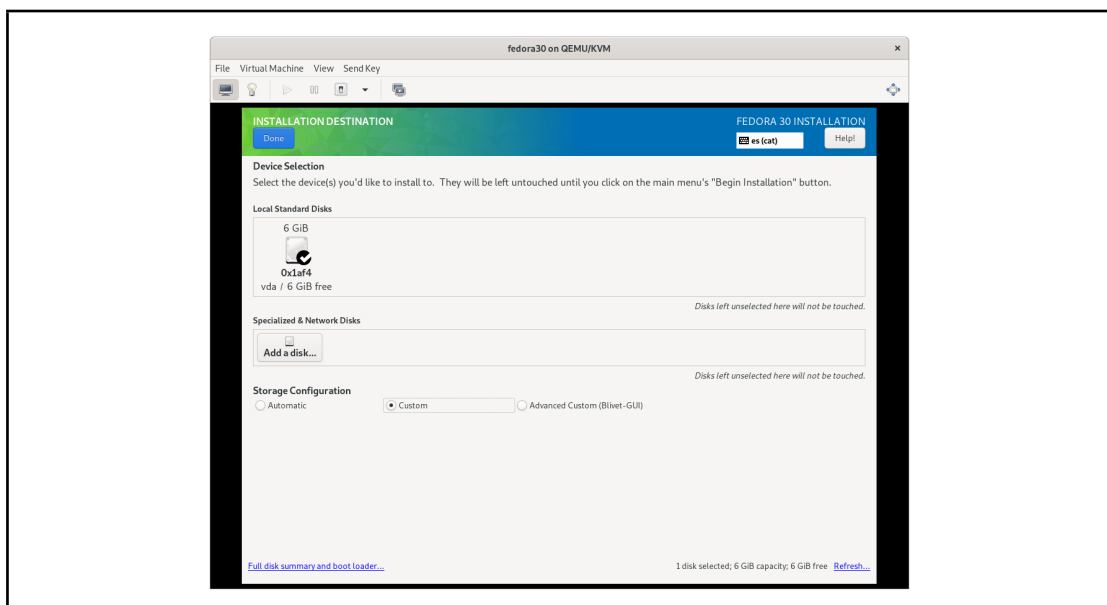
- automàtic
  - genera una partició /dev/vda1 de /boot
  - una partició /dev/vda de la que en fa un LVM un anomenat
    - fedora-root que és la / (arrel)
    - fedora-swap que és la partició swap

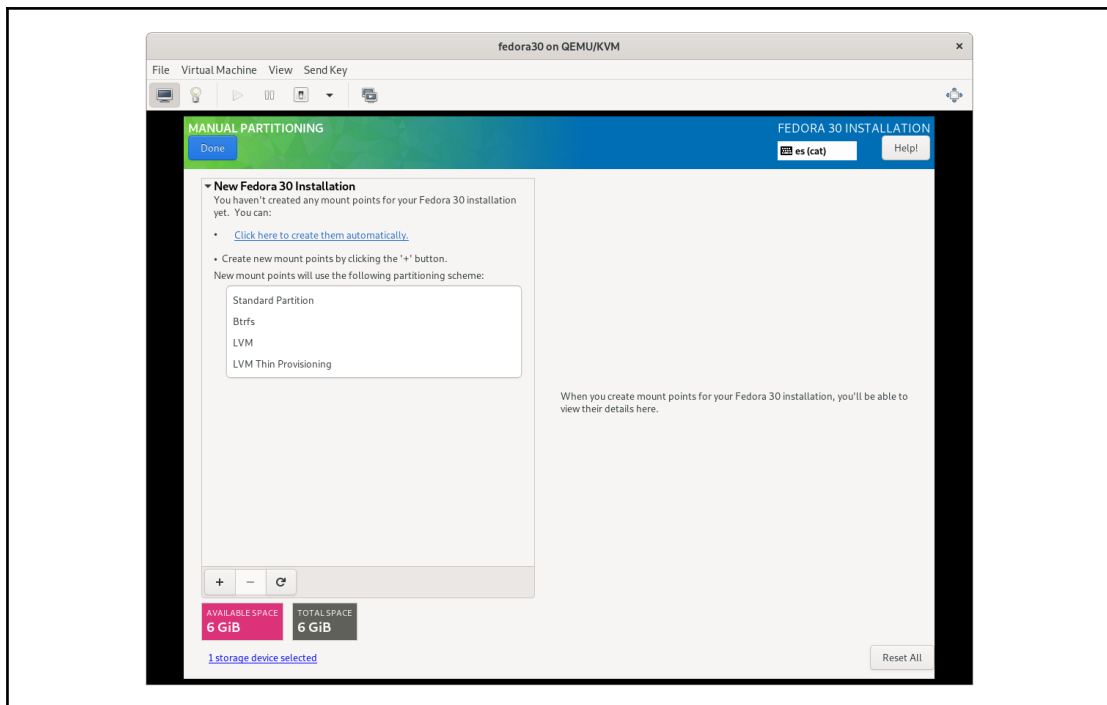
```
[root@localhost ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sr0                  11:0    1 1024M  0 rom
vda                  252:0    0    6G  0 disk
├─vda1               252:1    0    1G  0 part /boot
└─vda2               252:2    0    5G  0 part
   └─fedora-root      253:0    0  4.4G  0 lvm  /
      └─fedora-swap   253:1    0  616M  0 lvm  [SWAP]
[root@localhost ~]# _
```



## Opcions de particionat de Fedora-32: **Custom**

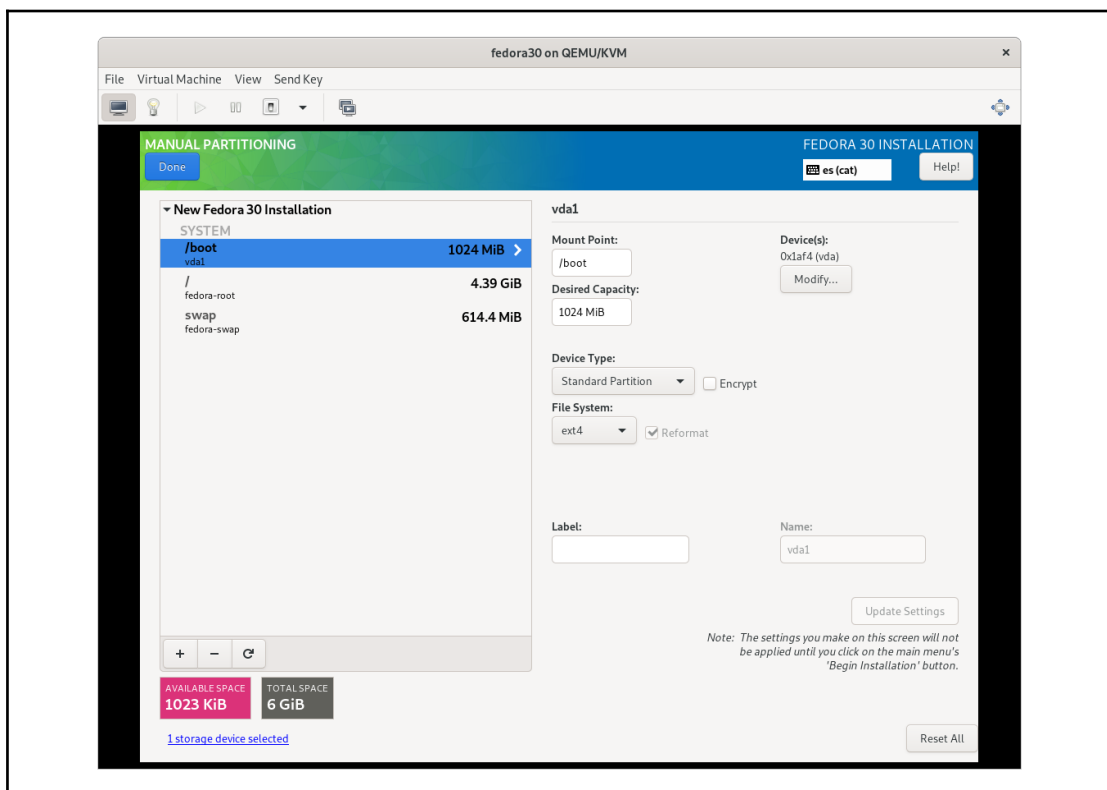
- Custom
  - Permet generar-les automàticament (clicar per observar què passa i estudiar quina és la configuració que genera)
  - Partition schema:
    - LVM Utilitza volums lògics
    - LVM Thin provisioning
    - Btrfs Sistema de fitxers Btrfs
    - Standard Partition (opció a usar per fer-les nosaltres manualment)





Exemple amb LVM + Create them automatically

- /boot (standard partition)
- / (arrel) un LVM anomenat fedora-root
- swap un LVM anomenat fedora-swap

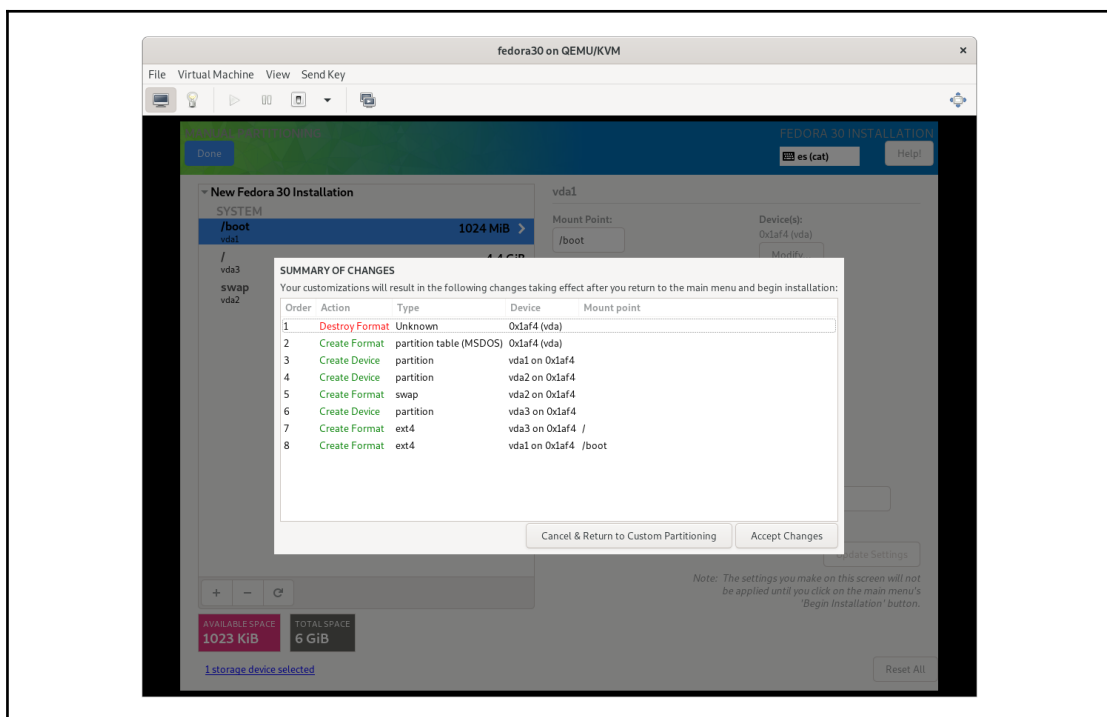
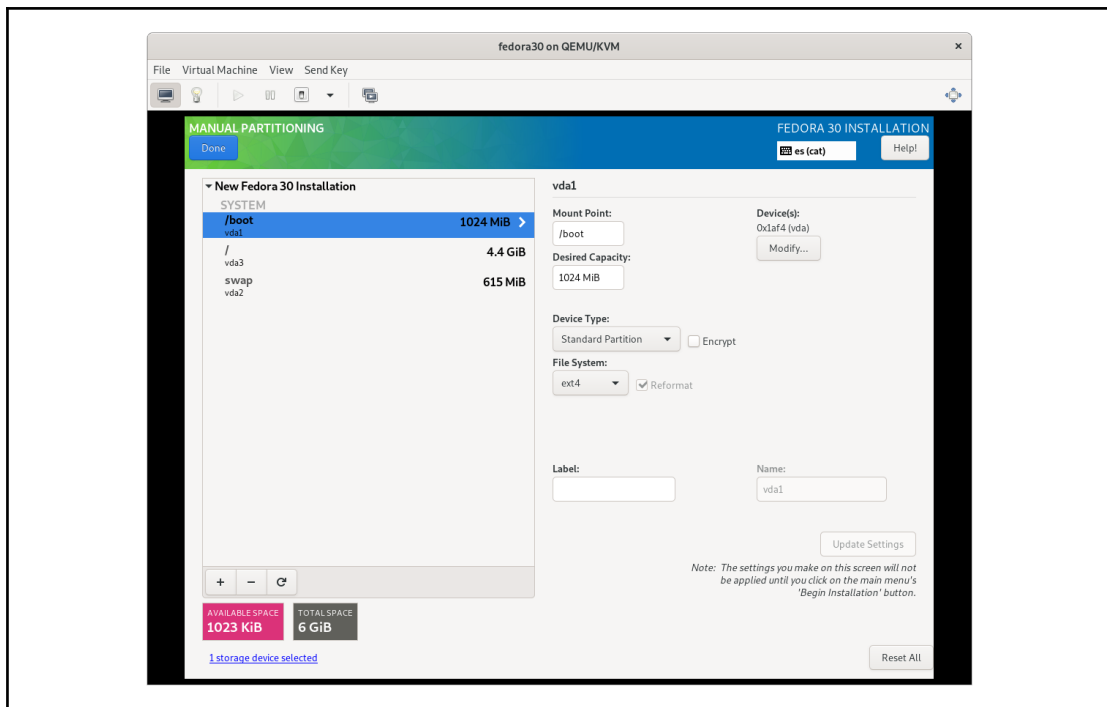


## Observar les opcions de configuració manual

- Globals:
  - Available espace
  - Total Space
  - + (per crear noves particions)
  - - (per eliminar una partició existent)
  - @ reload storage (per recarregar la configuració de disc)
  - Reset all (elimina tot el que s'ha configurat i comença de nou)
  - Storage device selected (mostra quins dispositius hi ha)
- vda1
  - Mount Point
  - Desired Capacity
  - Device Type ( + Encrypt)
  - File system
  - Label
- Update settings (desar els canvis)
- **Tricks:**
  - Observeu que cal modificar alguna cosa per poder fer el update settings.
  - En tot moment podeu fer el rescan i tornar a començar o el reset all.
  - Practiqueu diferents tipus de particionat sense fer la instal·lació realment. Fins i tot podeu desar-los formatant el disc (si és virtual, amb els reals de l'aula no!). Podem fer els accept changes i després NO fer la instal·lació. El disc dur no es formata realment fins al procés d'instal·lació.
  - Observar des de la consola (F2 de Send key) les particions amb les ordres fdisk -l, lsblk, blkid (F6 per tornar a la consola gràfica amb Send key).

## Exemple amb Standard Partition + Create them automatically

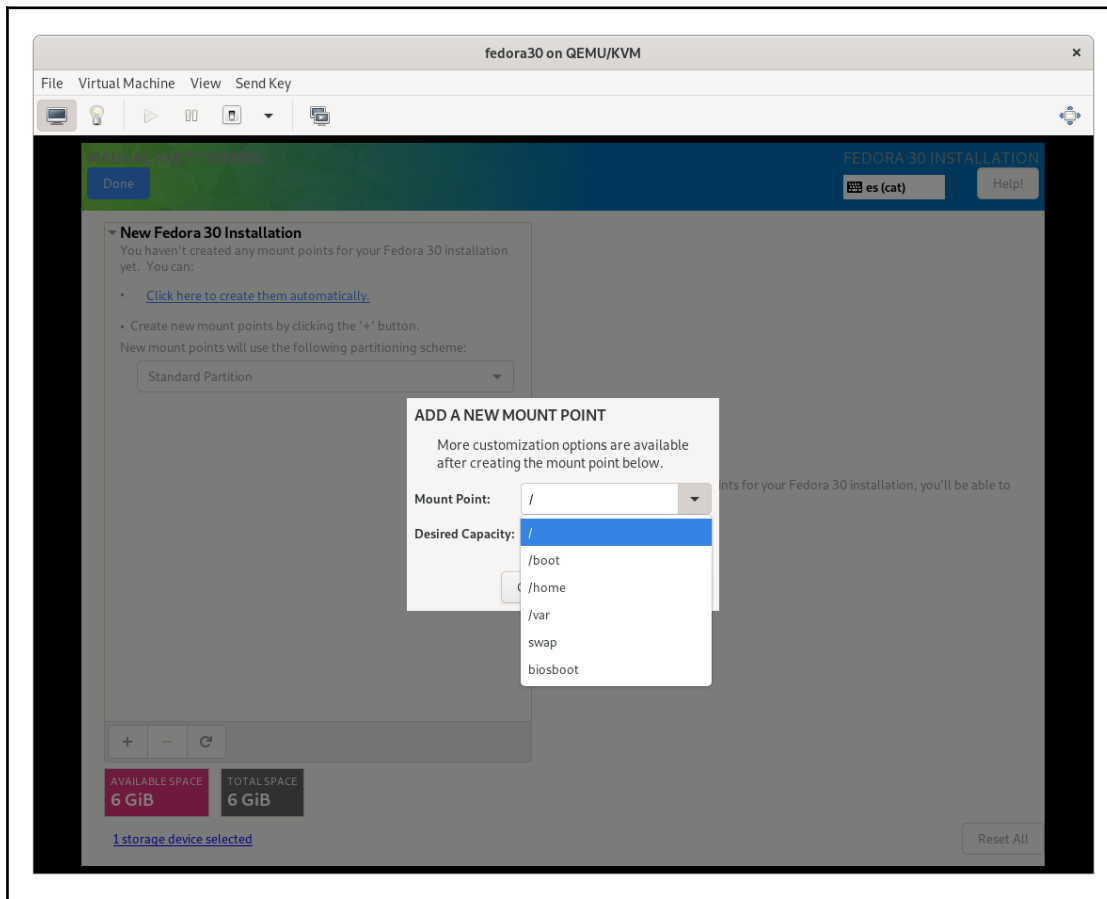
- /boot (/dev/vda1)
- / (arrel) (/dev/vda3)
- swap (/dev/vda2)



### Exemple de creació de particions manualment (gràfic)

- Si es selecciona que es vol fer una configuració Custom i les particions de tipus Standard Partition es poden usar els botons + i – per crear particions.
- Observar que per crear una nova partició cal indicar:
  - Mount point
  - Capacity
- El Mount Point es pot indicar manualment però n'hi ha uns quants de ja predefinits:
  - /

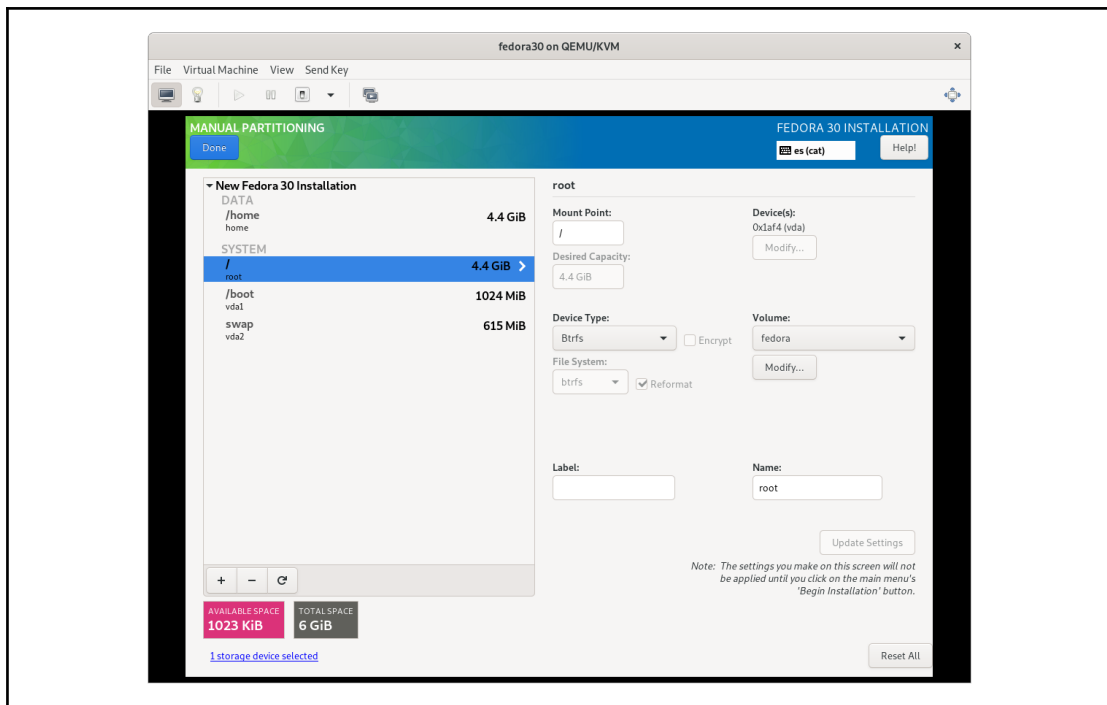
- /boot
- /home
- /var
- swap
- Biosboot
- També es pot anar a la consola (F2 amb Send key) i crear allà manualment les particions. Un cop fetes en tornar a la sessió gràfica (F6) cal fer el Rescan del disc.



### Exemple de creació de particions Btrfs

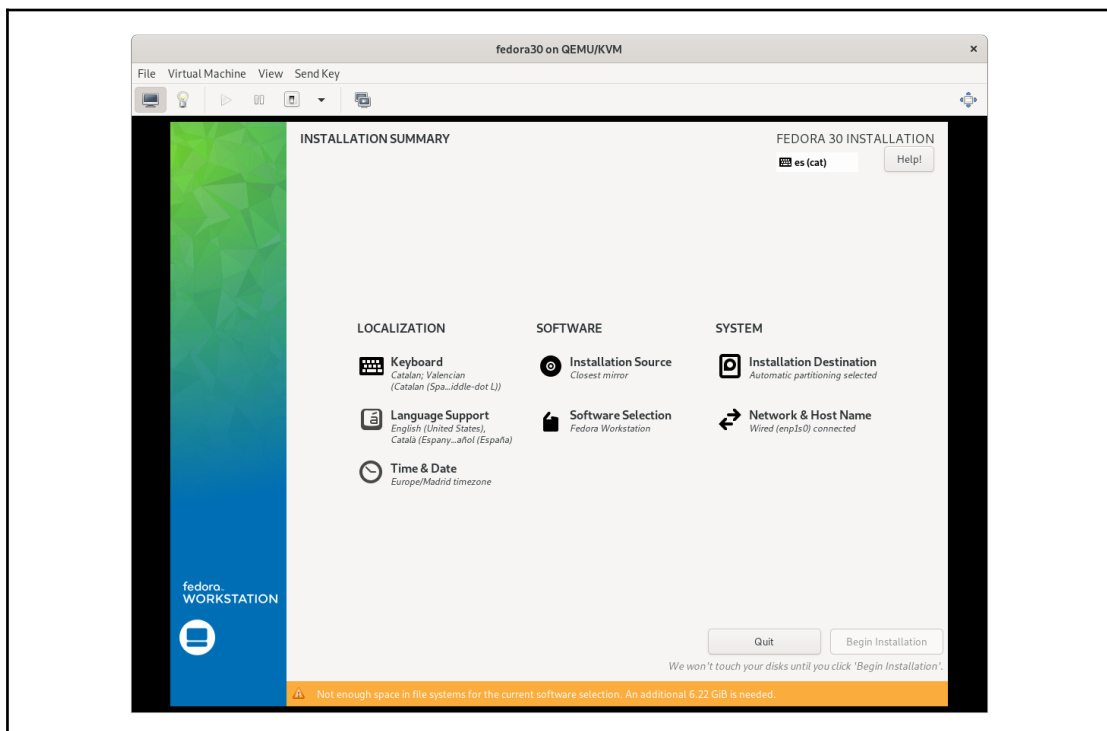
- /home de tipus Btrfs
- / (arrel) de tipus Btrfs
- /boot (standard partition)
- swap (standard partition)





## Exemple d'espai insuficient per a la instal·lació

- En aquest exemple podem veure que en intentar fer una instal·lació Workstation en lloc de minimal es queixa de que li falten 6 GiB més (en total en requereix +12 GiB).



---

## Debian-11 <pendent>

---

### Debian-11 Bullseye

- Practicar el procediment d'instal·lació de Debian-11
- Observar les opcions de particionat automatiques i manuals.

---

Ubuntu 20 <pendent>

---