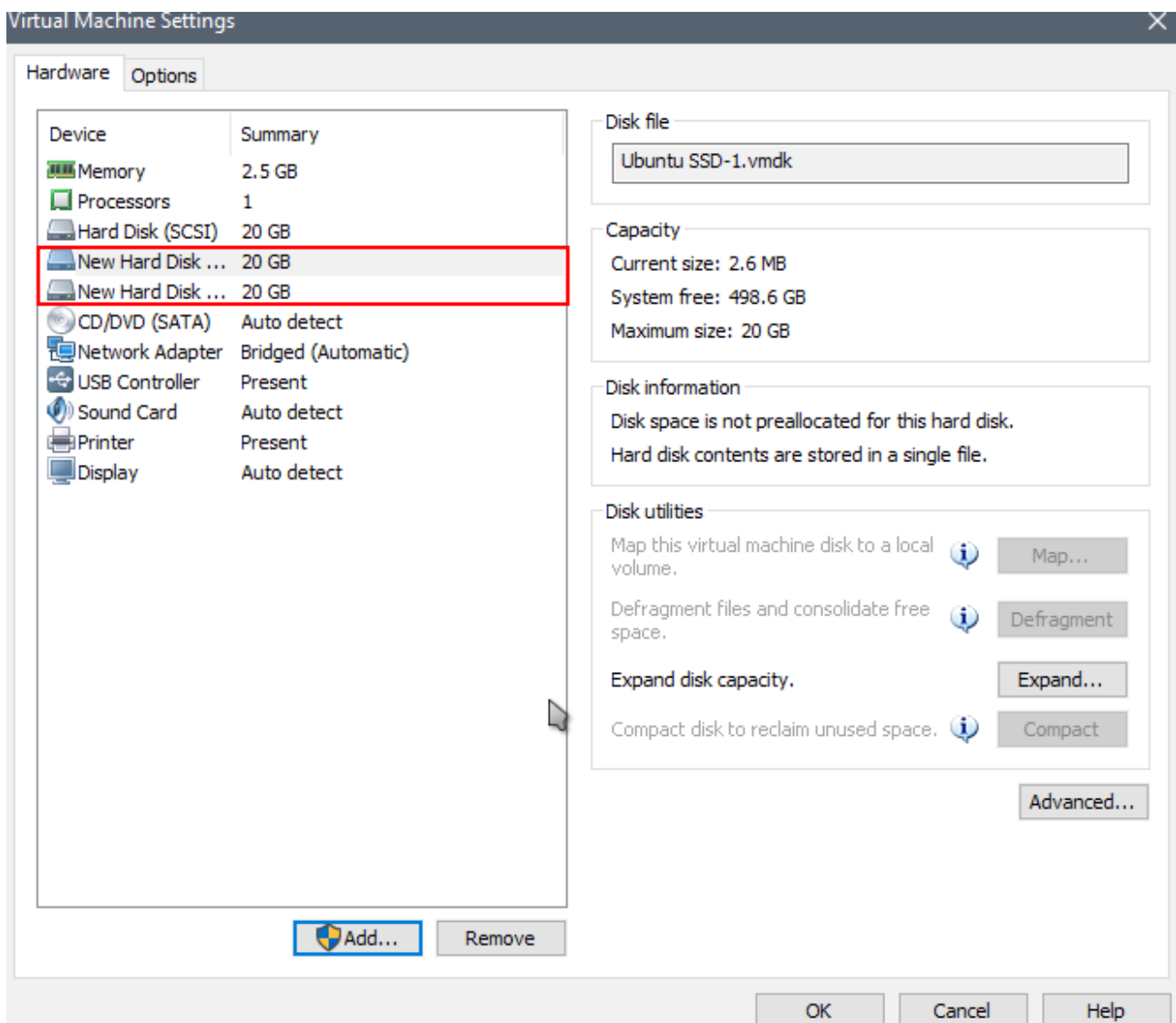


Práctica SSD Caché

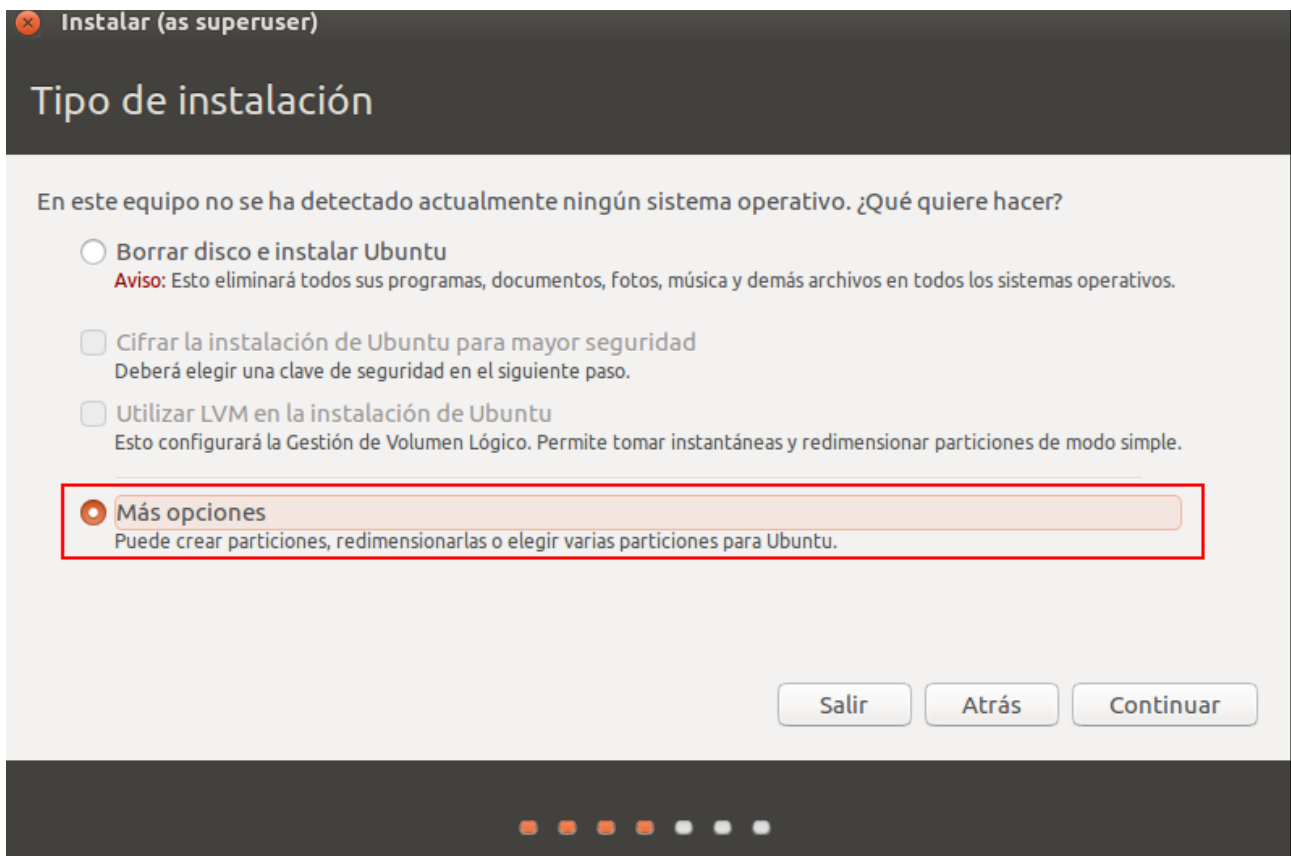


Esta práctica va a tratar sobre la instalación de una caché en SSD que trabaje en conjunto con un disco HDD convencional. La instalación se realizará en Linux Ubuntu 16.04 utilizando el comando **bcache** para llevar a cabo las operaciones requeridas.

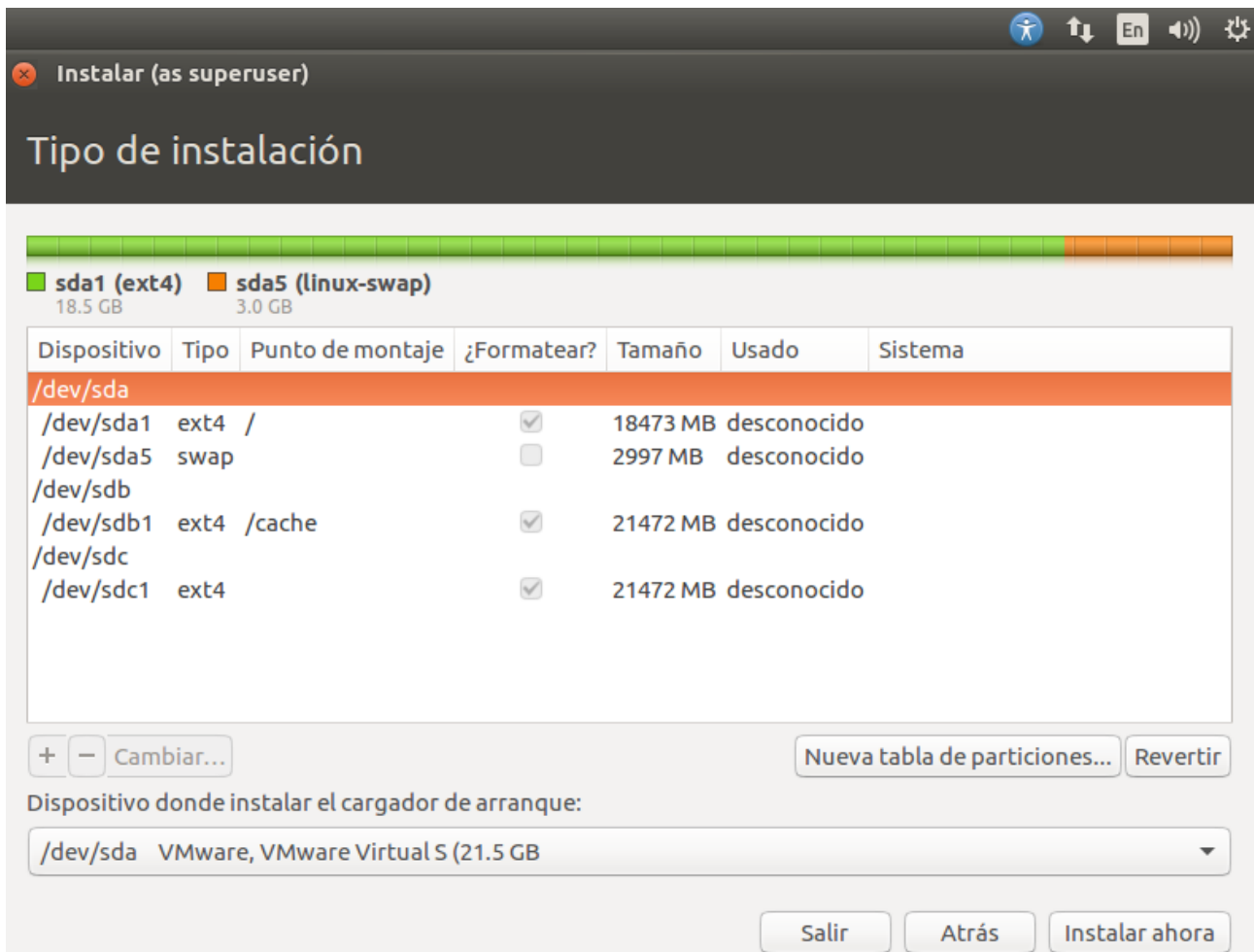
Para mayor facilidad en el proceso de configuración de la caché no vamos a usar el disco principal (donde tenemos instalado Ubuntu) como HDD. En su lugar instalamos dos discos en VMWare, uno hará de HDD y el otro simulará ser el SSD que hará de caché. Así nos quedamos con un total de tres discos, sda, sdb y sdc.



Usando el instalador de Ubuntu vamos a configurar manualmente el particionado de los discos para tenerlos listos a la hora de programar la caché.



En sda vamos a instalar el sistema operativo en sí, con punto de montaje en raíz y una parte de swap, como en cualquier instalación común. Sdb va a tener su propio punto de montaje en una carpeta creada por nosotros mismos, /cache. Finalmente, a sdc, nuestro disco SSD, vamos a darle formato, pero no punto de montaje porque no lo necesita.



Completamos la instalación de manera normal. Cuando nuestro Ubuntu esté listo para ser utilizado, accedemos a una terminal de comandos y ejecutamos `#sudo apt-get update`.

```
presen@presen-virtual-machine:~$ sudo apt-get update
```

Vamos a instalar **bcache**, esencial para este proceso. Podemos hacerlo con el comando `#sudo apt-get install bcache-tools`.

```
presen@presen-virtual-machine:~$ sudo apt-get install bcache-tools
```

Ya disponemos de **bcache**, pero antes de entrar en el proceso de asignar los discos y programar la caché, necesitamos hacer dos cosas. En primer lugar, desmontar el disco sdb, puesto que si no lo hacemos, el sistema nos indicará que está ocupado a la hora de hacer cualquier tipo de modificación en él. No es necesario hacer esto con sdc, pues recuerda que no le dimos punto de montaje durante el particionado.

```
presen@presen-virtual-machine:~$ sudo umount /dev/sdb1
```

En segundo lugar, para que **bcache** pueda escribir en la configuración de los discos a utilizar, necesitamos borrar sus super bloques. Para ello ejecutamos `#sudo wipefs /dev/sdb1`. Repetimos el proceso pero añadimos la opción `-a` (`#sudo wipefs -a /dev/sdb1`). Tenemos que repetir este proceso también con sdc.

```
presen@presen-virtual-machine:~$ sudo wipefs /dev/sdb1
offset          type
-----
0x438          ext4   [filesystem]
                UUID:   e993d6ce-a5b5-4c7f-a8fa-04f1dfaa977f
```

```
presen@presen-virtual-machine:~$ sudo wipefs -a /dev/sdb1
/dev/sdb1: 2 bytes were erased at offset 0x00000438 (ext4): 53 ef
```

```
presen@presen-virtual-machine:~$ sudo wipefs /dev/sdc1
offset          type
-----
0x438          ext4   [filesystem]
                UUID:   bf47e0fb-55d8-44d0-9209-1dfec65ba253
```

```
presen@presen-virtual-machine:~$ sudo wipefs -a /dev/sdc1
/dev/sdc1: 2 bytes were erased at offset 0x00000438 (ext4): 53 ef
presen@presen-virtual-machine:~$
```

Ahora nuestros discos están listos para ser configurados. En primer lugar tenemos que indicarle a **bcache** cual es nuestro dispositivo de respaldo, es decir, nuestro HDD. Ejecutamos `#sudo make-bcache -B /dev/sdb1`.

```
presen@presen-virtual-machine:~$ sudo make-bcache -B /dev/sdb1
UUID:          4411a822-4ea0-4c75-8e53-778b32581b4e
Set UUID:      ef0eb22a-9bde-4875-9b0d-f2fbec444db2
version:       1
block_size:    1
data_offset:   16
presen@presen-virtual-machine:~$
```

A continuación, cual es el dispositivo de caché, el SSD. Repetimos el proceso, pero cambiamos la opción `-B` por `-C` y la ruta de nuestro dispositivo.

```
presen@presen-virtual-machine:~$ sudo make-bcache -C /dev/sdc1
UUID:          2103baa3-e212-48f2-a5f3-2d1a87b3c4ef
Set UUID:      fcffbe36-cb88-4066-a7b0-e0f143f5b540
version:       0
nbuckets:      40956
block_size:    1
bucket_size:   1024
nr_in_set:     1
nr_this_dev:   0
first_bucket:  1
presen@presen-virtual-machine:~$
```

Ahora tenemos que relacionar el dispositivo de respaldo con el de caché en la configuración de **bcache**. Para ello necesitamos saber el UUID del dispositivo SSD. Para averiguarlo ejecutamos el comando `# bcache-super-show /dev/sdc1 | grep cset.uuid`.

```
presen@presen-virtual-machine:~$ sudo bcache-super-show /dev/sdc1 | grep cset.uuid
cset.uuid          fcffbe36-cb88-4066-a7b0-e0f143f5b540
presen@presen-virtual-machine:~$
```

Copiamos el código (o lo memorizas, si tienes memoria fotográfica). Usamos el comando `# echo [UUID] > /sys/block/bcache0/bcache/attach`. **Nota:** Este comando solo funcionará si estamos como root.

```
presen@presen-virtual-machine:~$ sudo echo fcffbe36-cb88-4066-a7b0-e0f143f5b540
> /sys/block/bcache0/bcache/attach
bash: /sys/block/bcache0/bcache/attach: Permiso denegado
presen@presen-virtual-machine:~$ sudo su
root@presen-virtual-machine:/home/presen# sudo echo fcffbe36-cb88-4066-a7b0-e0f1
43f5b540 > /sys/block/bcache0/bcache/attach
```

¡Y ya está! Para comprobar que nuestra caché está correctamente configurada **bcache** tiene un archivo que nos indica el estado de la susodicha. Ejecutamos `# cat /sys/block/bcache0/bcache/state`.

Nos puede dar varios resultados:

- **no cache:** no tenemos ninguna caché configurada.
- **clean:** la caché está configurada y limpia, este es el estado que queremos que nos salga.
- **dirty:** que todo está configurado correctamente, pero la caché está sucia.
- **inconsistent:** el dispositivo de respaldo y el de caché no están sincronizados adecuadamente, en algún paso lo has hecho mal.

```
root@presen-virtual-machine:/home/presen# cat /sys/block/bcache0/bcache/state
clean
```