



# Capítulo 5

---

LVM



# Conceptos sobre LVM

---

- LVM es un *conjunto* de herramientas que permiten la administración de unidades de almacenamiento denominadas *volúmenes lógicos*.
- Permite agrupar dispositivos físicos o particiones de manera tal que el sistema los considere como una unidad.
  - Cada uno de estos discos o particiones recibe el nombre de *volumen físico* (o **PV** por sus siglas en inglés, de **Physical Volume**). Cada PV está dividido en *extensiones físicas* (o **PEs**, de **Physical Extents**), que representan *trozos* de espacio de almacenamiento.
  - Utilizando una serie de PVs se crea un *grupo de volúmenes* (o **VG** por sus siglas en inglés, de **Volume Group**).
  - Finalmente, cada VG se divide en *volúmenes lógicos* (análogos a las particiones tradicionales) que pueden ser redimensionados utilizando el espacio disponible en el VG. Cuando un VG se queda sin espacio, simplemente necesitaremos crear un PV adicional, agregarlo al VG, para luego incrementar el tamaño del LV que lo necesite.
- Provee flexibilidad para administrar el almacenamiento del sistema de manera que pueda expandirse fácilmente cuando la necesidad así lo requiera.
- Una ventaja adicional de LVM es que permite tomar *instantáneas* de los volúmenes lógicos, las cuales pueden funcionar como copias de respaldo o ser utilizadas para migraciones, pruebas, restauraciones, etc., sin afectar el funcionamiento del sistema en su totalidad.



# Particionado tradicional vs. LVM

---

- Si bien es posible achicar una partición tradicional para darle espacio a otra que haya quedado pequeña, este procedimiento causa los siguientes inconvenientes:
  - **Provoca downtime.** Esto significa que mientras estemos redimensionando las particiones nadie más podrá utilizar el sistema. Si se trata de nuestro equipo personal, los únicos afectados seríamos nosotros. Pero si estamos hablando de un servidor al que necesiten acceder otros usuarios, esta alternativa puede no ser viable.
  - **Requiere copias de respaldo.** Si no se dispone de espacio extra en alguna partición, no quedará más remedio que agregar otro disco al equipo. Si queremos utilizar el nuevo dispositivo para **/home** (por ejemplo), esto implicará mover todos los archivos de los usuarios de la ubicación anterior a la nueva.
- Por otro lado, el redimensionado de los volúmenes lógicos bajo LVM se puede realizar sin interrumpir el funcionamiento normal del equipo. Además, el procedimiento es mucho más rápido que en el particionado tradicional. Los cambios quedan disponibles a los usuarios inmediatamente sin necesidad de reiniciar ningún servicio.



# LVM – Resumen de conceptos

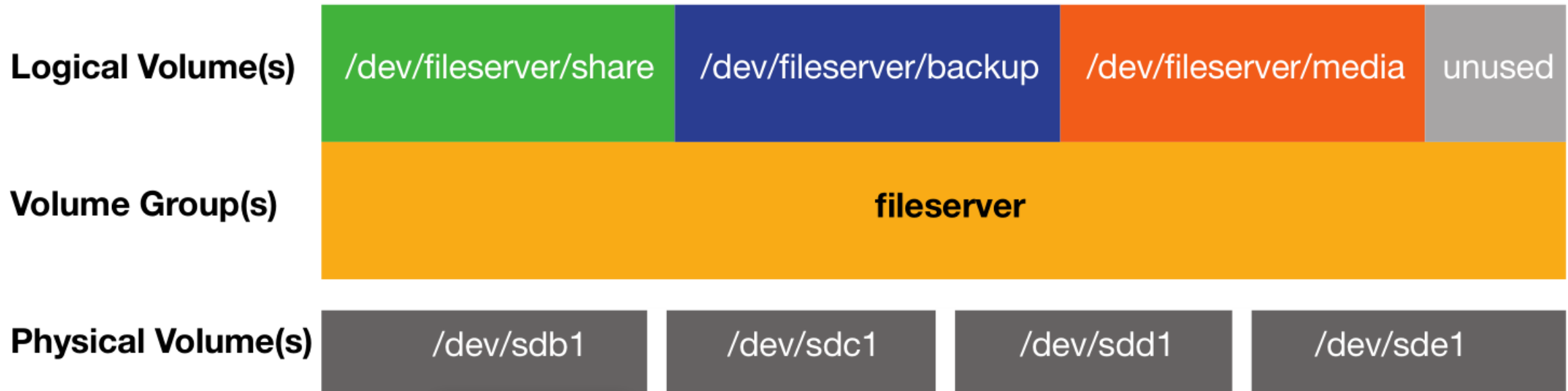
---

- RAID nos provee redundancia
- LVM brinda flexibilidad en la administración del almacenamiento
- Al contrario del particionado tradicional, LVM hace más simple el redimensionado de unidades de almacenamiento
- Si nos quedamos sin espacio en un volumen lógico, podemos tomar de alguno que tenga extra
- En el particionado tradicional, esta operación requiere downtime, en LVM no
- Cuando un nuevo disco se añade al sistema, no es necesario mover los datos de los usuarios
- Simplemente se añade el nuevo disco al grupo lógico correspondiente y se expanden los volúmenes lógicos



# Estructura de LVM

Un LVM se descompone en **tres partes**:





# Estructura de LVM – PVs y VGs

---

- Volúmenes físicos (Physical Volumes, o PVs): Particiones o discos enteros
  - Crear volúmenes físicos: **`pvcreate /dev/sdf /dev/sdg`**
  - Mostrar atributos de los PVs: **`pvdisplay`**
  - Información sobre PVs: **`pvs`**
  - Quitar una partición o disco entero: **`pvremove /dev/sdg`**
- Grupos de volúmenes (Volume Groups, o VGs): Las "cajas" donde residen los PVs
  - Mientras un PV no se agregue al VG, no podemos usarlo
  - Crear un VG con los PVs existentes: **`vgcreate cursoadm /dev/sdf /dev/sdg`**
  - Mostrar atributos de un VG: **`vgdisplay cursoadm`**
  - Información sobre un VG: **`vgs cursoadm`**
  - Agregar un nuevo PV al VG: **`vgextend cursoadm /dev/sdh`** (hay que acordarse de crear un PV en /dev/sdh primero!)
  - Si quisiéramos quitar un PV del VG: **`vgremove /dev/sdh`**
  - Si queremos renombrar un VG: **`vgrename cursoadm cursoadm nuevo`**



# Estructura de LVM – PVs y VGs

---

- Volúmenes físicos (Physical Volumes, o PVs): Particiones o discos enteros
  - Crear volúmenes físicos: **`pvcreate /dev/sdf /dev/sdg`**
  - Mostrar atributos de los PVs: **`pvdisplay`**
  - Información sobre PVs: **`pvs`**
  - Quitar una partición o disco entero: **`pvremove /dev/sdg`**
- Grupos de volúmenes (Volume Groups, o VGs): Las "cajas" donde residen los PVs
  - Mientras un PV no se agregue al VG, no podemos usarlo
  - Crear un VG con los PVs existentes: **`vgcreate cursoadm /dev/sdf /dev/sdg`**
  - Mostrar atributos de un VG: **`vgdisplay cursoadm`**
  - Información sobre un VG: **`vgs cursoadm`**
  - Agregar un nuevo PV al VG: **`vgextend cursoadm /dev/sdh`** (hay que acordarse de crear un PV en /dev/sdh primero!)
  - Si quisiéramos quitar un PV del VG: **`vgremove /dev/sdh`**
  - Si queremos renombrar un VG: **`vgrename cursoadm cursoadm nuevo`**



# Estructura de LVM – LVs

---

- Es el equivalente a una partición en un sistema tradicional
- Cada LV es visible como un dispositivo estándar de bloques, por lo que puede contener un sistema de archivos (por ejemplo /home)
- Crear LVs dentro de un VG:  
`lvcreate --name operador --size 300M cursoadm`  
`lvcreate --name admin --size 500M cursoadm`  
`lvcreate --name redes --extents 100%FREE cursoadm`
- Mostrar atributos de un LV: `lvdisplay operador`
- Información sobre un LV: `lvs cursoadm/operador`
- Renombrar un LV: `lvrename cursoadm admin nivelador`
- Quitar un LV de un VG: `lvremove cursoadm/nivelador`





# Estructura de LVM – LVs (Cont.)

---

- Quitar un LV de un VG: **lvremove cursoadm/nivelador**
- Redimensionar un LV:
  - Reducir: **lvreduce --size 200M cursoadm/operador** (¡CUIDADO! No reducir por debajo de la capacidad utilizada actualmente!)
  - Extender: **lvextend --extents +100%FREE cursoadm/redes**
- Los "extents" en el contexto de LVM son los bloques básicos en los que se divide un PV
- Crear un sistema de archivos en un LV: **mkfs.ext4 /dev/cursoadm/redes**
- Reducir o extender un LV: **lvresize --resizefs --size 150M /dev/cursoadm/redes**
- Usar **blkid /dev/cursoadm/redes** para averiguar el UUID del dispositivo y usarlo en /etc/fstab
  - UUID significa "Universally Unique Identifier"