# RAID / LVM

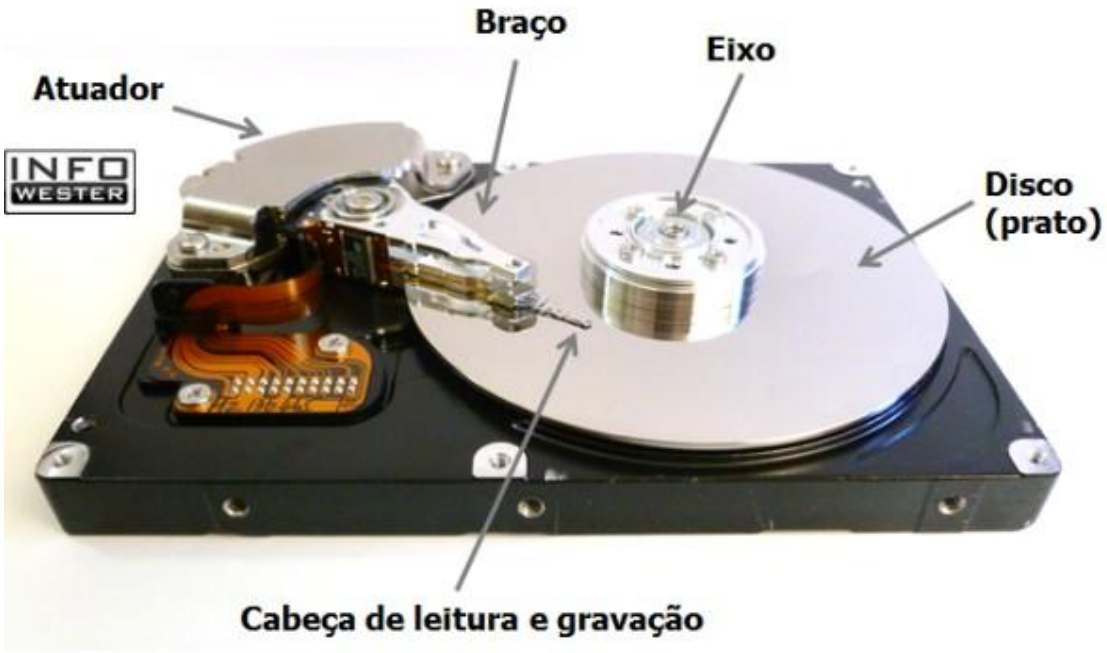
Fábio da Silva Lima
Analista de infraestrutura
www.arripio.com.br

# HDs

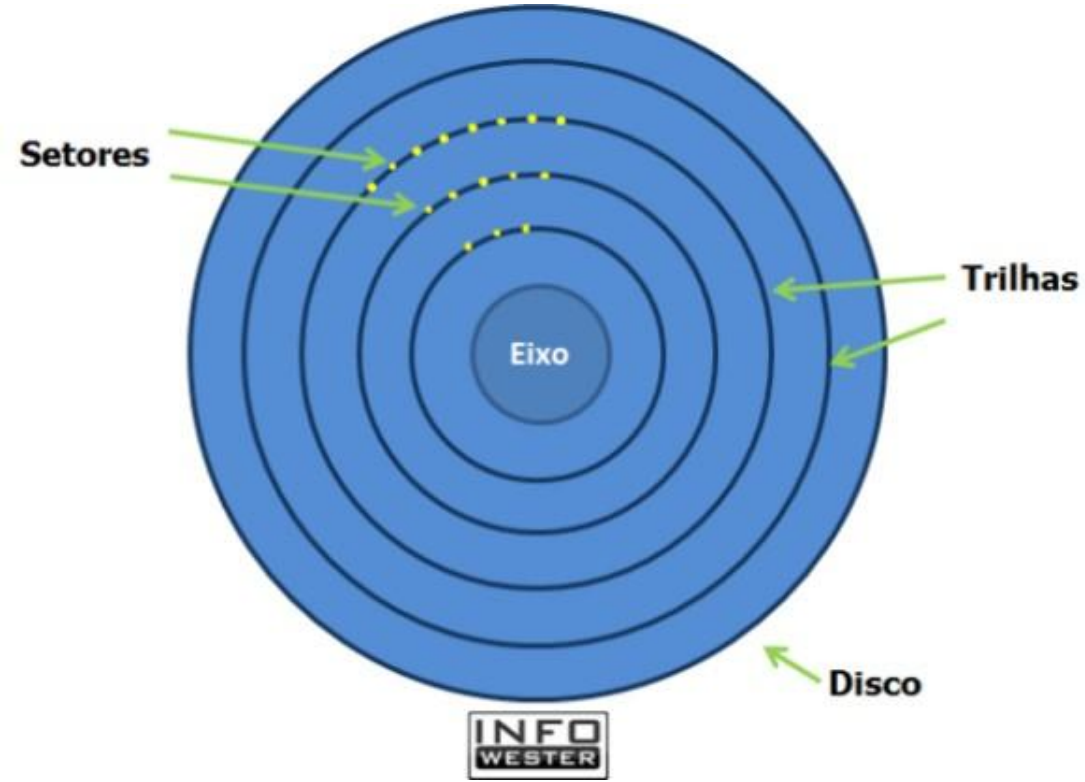
## Surgimento dos HDs

O disco rígido não é um tipo dispositivo de armazenamento novo, mas sim um aparelho que evoluiu - e muito - com o passar do tempo. Um dos primeiros HDs que se tem notícia é o *IBM 305 RAMAC*. Disponibilizado no ano de 1956, era capaz de armazenar até 5 MB de dados (um avanço para a época) e possuía dimensões enormes: 14 x 8 polegadas. Seu preço também não era nada convidativo: o 305 RAMAC custava cerca de 30 mil dólares.

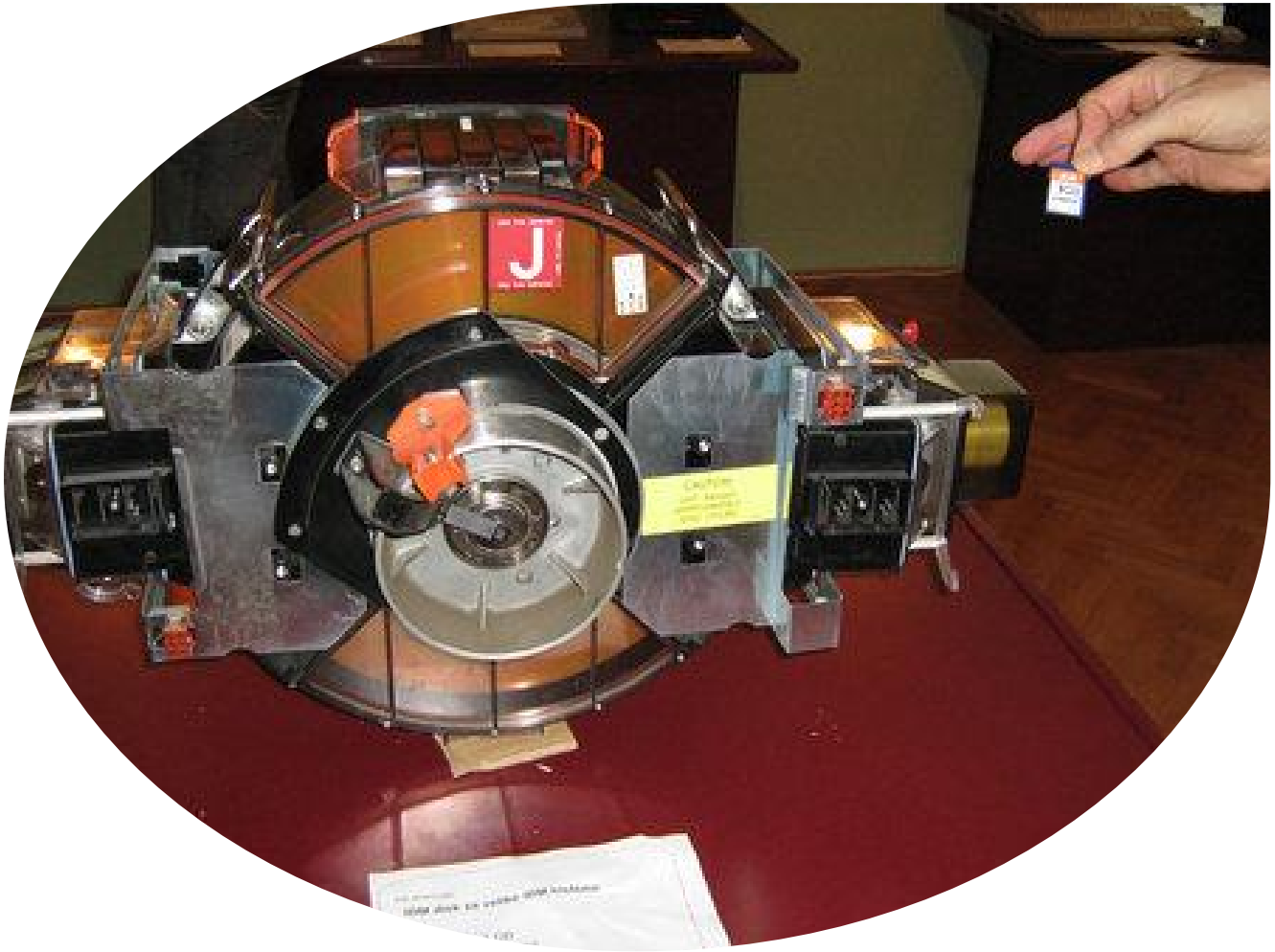
# Funcionamento HD



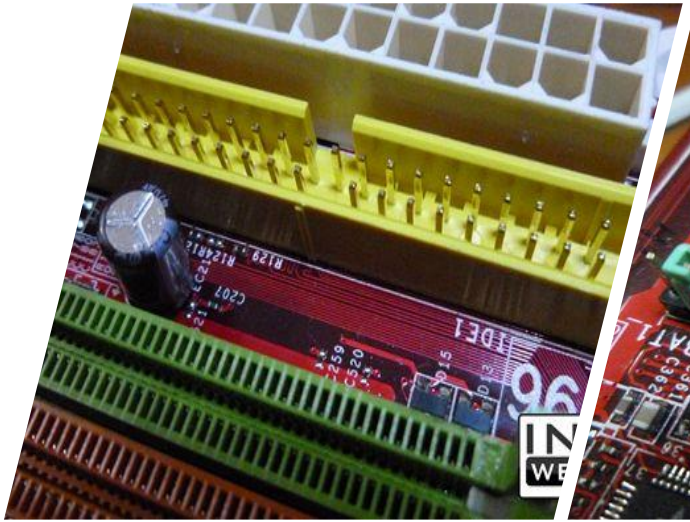
# Organização



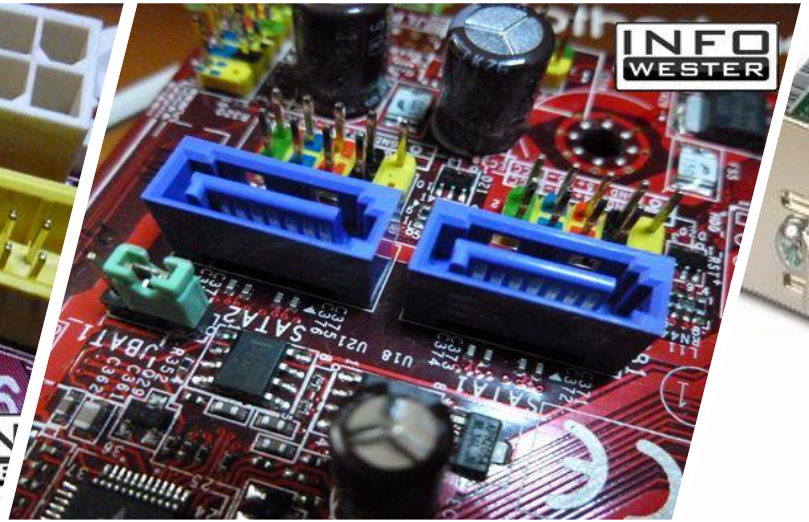
# Evolução



# Tipos de HDs



*IDE*




*SATA*

*ISCSI*

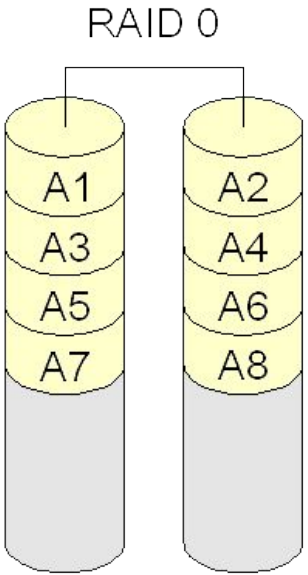
# RAID

RAID (*Redundant Array of Independent Disks*) se destina a combinar discos de forma a incrementar o desempenho de entrada e saída e, principalmente, segurança dos dados contra defeitos em discos.

RAID pode ser provido via software ou hardware (melhor este último).

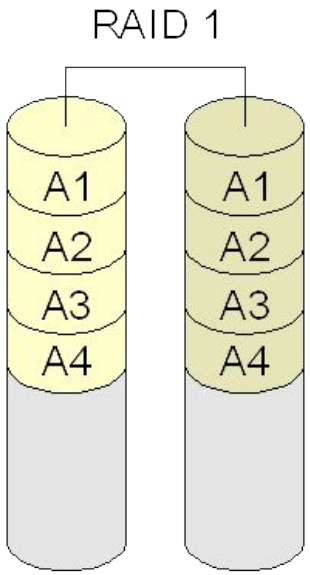
O Linux possui implementação por software em seu kernel

# Tipos RAID-0



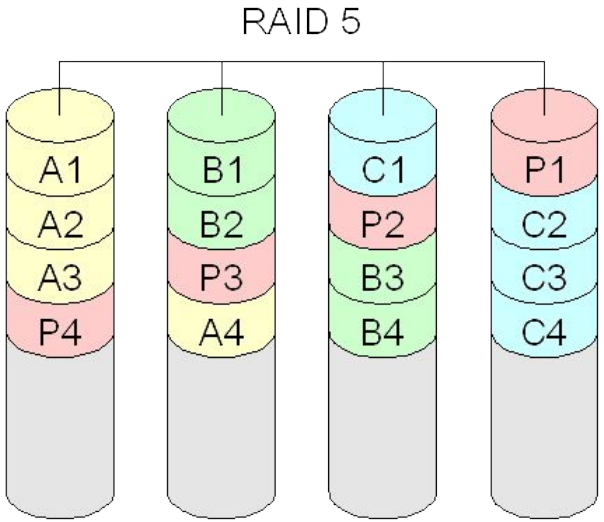

- ***RAID 0 (ou striping)***: combina discos ou partições de forma alternada, para distribuir os acessos entre eles (aumentar desempenho). Porém, se um disco falhar perdem-se todos os dados. Requer um mínimo de dois discos.

# Tipos RAID-1




- ***RAID 1 (ou mirroring)***: combina discos ou partições para espelhar dados (segurança). Requer o dobro de discos necessários para guardar os dados (ex: se há dois discos com dados, são necessários outros dois para espelhamento). Se todos os discos falharem, é possível continuar a operar usando os discos espelhados. Requer no mínimo dois discos.

# Tipos RAID-5

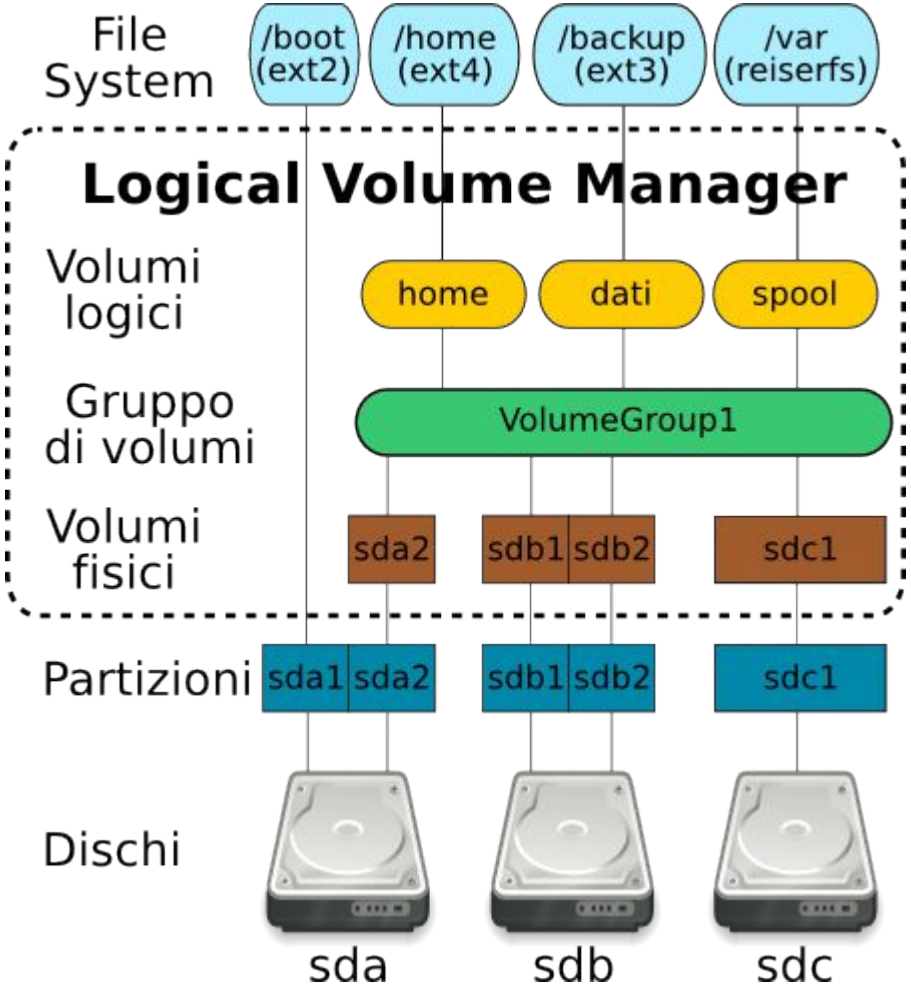


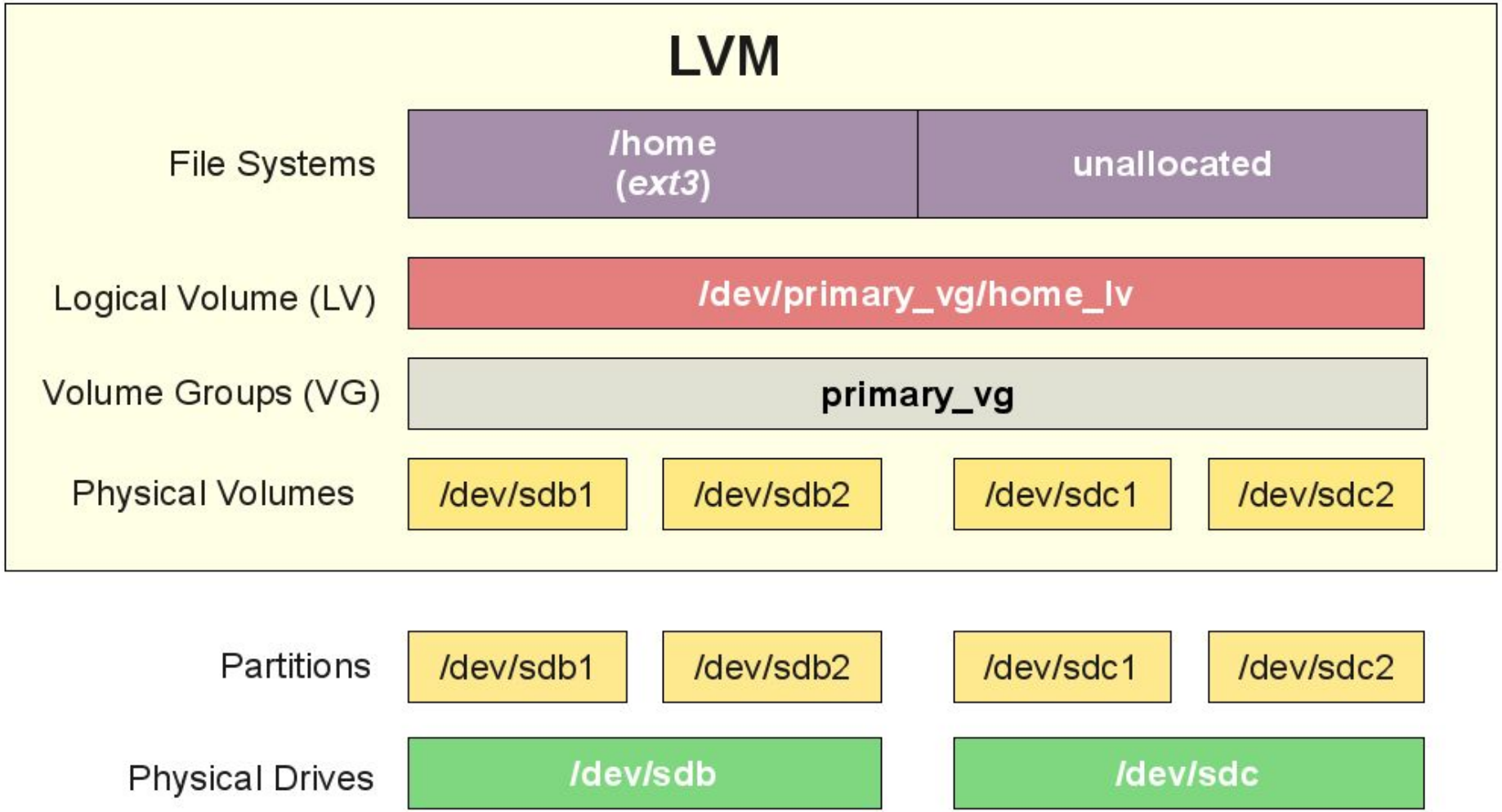

- ***RAID 5*** combina discos ou partições para ter redundância de dados (segurança), usando um esquema baseado em paridade. Se um disco falhar, é capaz de continuar operando (porém com desempenho reduzido até que esse disco seja reposto).

# LVM

Em Linux, **Logical Volume Manager** (**LVM**), em português **Gerenciador de Volume Lógico**, é um mapeador de dispositivos que fornece gerenciamento de volume lógico para o kernel do Linux

# LVM


File System
/boot (ext2)
/home (ext4)
/backup (ext3)
/var (reiserfs)
Logical Volume Manager
Volumi logici
home
dati
spool
Gruppo di volumi
VolumeGroup1
Volumi fisici
sda2
sdb1
sdb2
sdc1
Partizioni
sda1
sda2
sdb1
sdb2
sdc1
Dischi
sda
sdb
sdc


LVM
File Systems
/home
(ext3)
unallocated
Logical Volume (LV)
/dev/primary_vg/home_lv
Volume Groups (VG)
primary_vg
Physical Volumes
/dev/sdb1
/dev/sdb2
/dev/sdc1
/dev/sdc2
Partitions
/dev/sdb1
/dev/sdb2
/dev/sdc1
/dev/sdc2
Physical Drives
/dev/sdb
/dev/sdc

# Laboratório 1

## Criação do LVM no Linux

A sequência de criação de um VG e seus LV é a seguinte:

1. Criar partições físicas do tipo 8E (Linux LVM), que serão usadas para serem os PV
2. Preparar essas partições para serem usadas como PV, usando o comando *lvm pvcreate caminho_partição (ex: lvm pvcreate /dev/sdb1)*
3. Criar o VG, usando o comando *lvm vgcreate nome_vg pv1 [pv2 ...] (ex: lvm vgcreate meu_vg /dev/sdb1 /dev/sdb2)*
4. Criar os LV, com o comando *lvm lvcreate nome_vg -L tamanho_LV -n nome_LV (ex: lvm lvcreate meu_vg -L 512M -n teste)*
5. Formatar os LV (ex: *mke4fs -j /dev/meu_vg/teste*, para formatar com sistema de arquivos ext4)

Abaixo segue um exemplo de uma sequência de comandos relacionados com LVM, desde o particionamento de um disco até o redimensionamento de um LV existente:

# LVM

```
# Prepara as partições (devem ser do tipo 8E (Linux LVM))
fdisk /dev/sdb

# Prepara essas duas partições para serem usadas como volumes físicos
lvm pvcreate /dev/sdb1
lvm pvcreate /dev/sdb2

# Cria o volume group "vg"
lvm vgcreate vg /dev/sdb1 /dev/sdb2

# Cria dentro do volume group "vg" um volume lógico "dados" com 512 MB iniciais
lvm lvcreate vg -L 512M -n dados

# Cria dentro do volume group "vg" um volume lógico "teste" com 256 MB iniciais
lvm lvcreate vg -L 256M -n teste

# Mostra informações sobre todos os volumes lógicos
lvm lvs

# Mostra detalhes sobre o volume lógico "dados", que pertence ao volume group "vg"
lvm lvdisplay /dev/vg/dados

# Formata o volume lógico "dados" com sistema de arquivos do tipo "ext4"
mkfs.ext4 -j /dev/vg/dados

# Formata o volume lógico "teste" com sistema de arquivos do tipo "xfs"
mkfs.xfs /dev/vg/teste

# Aumenta em 512 MB o tamanho do volume lógico "dados"
lvm lvresize -L +512M /dev/vg/dados

# Aumenta o sistema de arquivos contido no volume lógico "dados", para adaptá-lo ao seu novo tamanho
resize2fs /dev/vg/dados
```

# Laboratório 2

1. Com o fdisk crie três novas partições, no início do espaço livre do disco, uma de tamanho de 512 MB, outra de 1GB e a terceira com 1.4 GB. Formate-as com sistema de arquivos ext4.
2. Monte estas partições em /dados, /soft e /outra.
3. Configure o sistema para que faça a montagem automaticamente, ou seja, em todo reinício da máquina.
4. Desfaça os itens 2 e 3, para dar prosseguimento ao exercício.
5. Crie um grupo de volume LVM (VG) com nome GerVg, contendo as duas partições criadas no item 1. Esse VG deverá ter tamanho total de 1512 MB.
6. Crie 4 volumes lógicos, "dados", "home", "teste", "softwares", respectivamente com 300 , 400, 100 e 500 MB, dentro do VG.
7. Formate os volumes lógicos.
8. Monte as novas partições em /dados, /usuarios, /nada e /soft, respectivamente.
9. Aumente o tamanho de "home" em 500 MB, redimensionando o sistema de arquivos apropriadamente (e sem desmontá-lo).
10. Com o fdisk remova completamente as partições criadas, para não deixar o hd “bagunçado”. Remova também todos os diretórios criados.

# Comandos

- **lvmchange** — Change attributes of the Logical Volume Manager.
- **lvmdiskscan** — Scan for all devices visible to LVM2.
- **lvmdump** — Create lvm2 information dumps fo
- **pvchange** — Change attributes of a Physical Volume.
- **pvck** — Check Physical Volume metadata.
- **pvcreate** — Initialize a disk or partition for use by LVM.
- **pvdisplay** — Display attributes of a Physical Volume.
- **pvmove** — Move Physical Extents.
- **pvremove** — Remove a Physical Volume.
- **pvresize** — Resize a disk or partition in use by LVM2.
- **pvs** — Report information about Physical Volumes.
- **pvscan** — Scan all disks for Physical Volumes.

# Comandos

- **vgcfgbackup** — Backup Volume Group descriptor area.
- **vgcfgrestore** — Restore Volume Group descriptor area.
- **vgchange** — Change attributes of a Volume Group.
- **vgck** — Check Volume Group metadata.
- **vgconvert** — Convert Volume Group metadata format.
- **vgcreate** — Create a Volume Group.
- **vgdisplay** — Display attributes of Volume Groups.
- **vgexport** — Make volume Groups unknown to the system.
- **vgextend** — Add Physical Volumes to a Volume Group.
- **vgimport** — Make exported Volume Groups known to the system.
- **vgimportclone** — Import and rename duplicated Volume Group (e.g. a hardware snapshot).
- **vgmerge** — Merge two Volume Groups.
- **vgmknodes** — Recreate Volume Group directory and Logical Volume special files
- **vgreduce** — Reduce a Volume Group by removing one or more Physical Volumes.
- **vgremove** — Remove a Volume Group.
- **vgrename** — Rename a Volume Group.
- **vgs** — Report information about Volume Groups.
- **vgscan** — Scan all disks for Volume Groups and rebuild caches.
- **vgsplit** — Split a Volume Group into two, moving any logical volumes from one Volume Group to another by moving entire Physical Volumes.

# Fábio da Silva Lima

Analista de Infraestrutura

www.arripio.com.br

Não espere por outra pessoa o que só você pode fazer (Atitude palavra de  ordem)