

# Filesystem Maintenance

René Serral-Gracià<sup>1</sup>

<sup>1</sup>Universitat Politècnica de Catalunya (UPC)

February 21, 2022

# Lectures

- 1 System administration introduction
- 2 Operating System installation
- 3 User management
- 4 Application management
- 5 System monitoring
- 6 **Filesystem Maintenance**
- 7 Local services
- 8 Network services
- 9 Security and Protection
- 10 Virtualization

# Outline

- 1 Introduction
  - Goals
- 2 Filesystems
- 3 Disk verification
- 4 Logical Volume Manager (LVM)
- 5 Backups

# Goals

## Knowledge

- Filesystems
- Backup tools
- Backup media

## Abilities

- Filesystem resizing
- Filesystem verification
- Perform and restore backups

# Outline

- 1 Introduction
- 2 Filesystems**
- 3 Disk verification
- 4 Logical Volume Manager (LVM)
- 5 Backups

# Filesystems (I)

- FAT (FAT16) → DOS
  - Small disks (< 4GB)
  - File names 8+3
- FAT32 (VFAT) → Win95
  - Larger disks
  - Long filenames
  - Partial definition of soft-links
  - No owner or file access privileges
- exFAT
  - FAT32 Extension
  - Theoretical maximum capacity of 64ZiB (512TiB real)
- NTFS → WinNT, XP, Vista, Windows 7
  - Integrates ownership and privileges (create, modify, access...)
  - Maps to Windows NT security model

# Filesystems (II)

- ext2
  - UNIX Filesystem
  - Soft/hard links
  - Access privileges
  - Long filenames
- ext3
  - Adds journaling (eases error recovery)
- reiserfs
  - Files and directories organized similarly to a database
  - Features journaling
  - Very efficient in small files
  - No internal block fragmentation

# Filesystems (III)

- xfs
  - journaling
  - Dynamic i-node management
  - ACLs
  - Very large disk sizes
  - Filesystem activity log
- jfs
  - journaling
  - Dynamic i-node management
  - ACLs and MAC (Mandatory Access Control)
  - Very large disk sizes



# Filesystems (i IV)

- ext4
  - 64 bits addressing, improvements in journaling
  - Delayed allocation
  - Extents
  - 1 exbibyte (EiB) maximum size
- btrfs
  - Extents
  - Online resizing
  - Online balancing
  - Online filesystem check

# Journal filesystems

- Journal: disk operation registry
  - Eases the recovery of the FS in case of crash or error
  - Slightly decrease in disk operations performance
- Journal outside the buffer cache
  - Journal can be stored in another disk or partition
- Ext3/4, reiserfs, JFS, XFS, NTFS, BTRFS have journal

# Outline

- 1 Introduction
- 2 Filesystems
- 3 Disk verification**
  - Filesystem ampliation
  - Disk quota management
- 4 Logical Volume Manager (LVM)
- 5 Backups

# Disk verification (I)

## Reason for errors

- Hardware errors
- Power shortage
- Operating system bugs
- Administration errors
  - Incorrect machine shutdown

## Never verify a filesystem while mounted

- High probability of disk corruption
- Verification access skips the buffer cache and acts directly on the device

# Disk verification (II)

## Logical verification

- Filesystem metadata
- Directory structure
- Lost data recovery
  - Directory `lost+found`

## Physical recovery

- Disk blocks with Input/output errors
- Command: `badblocks`

# Filesystem ampliation

- Install and configure the new disk
  - Partition
  - or recycle existing ones. . .
- Create filesystem
- Decide mountpoints
- Transfer the required data to the new partition
- Mount the partition
  - Update `/etc/fstab`
- Maybe you have to reorganize existing directories
  - `/home` → `/homeA` + `/homeB`
  - `/home` → `/home/students` + `/home/professors`

# Exercise

## Plan and issue a filesystem resizing for the partitions

- /home
- /var

# Disk quota (I)

## Quota

Ability to limit the amount of data a user (or user group) is able to use in a filesystem (partition)

## Requires

- Support from the filesystem
- Support from the kernel



# Quota management (II)

## Partition preparation

- Mounted using options 'usrquota' and/or 'grpquota'
- It can be done from /etc/fstab

```
/dev/sda9 /home ext4 defaults,usrquota,grpquota 1 1
```

- quotacheck command to create the quota files

```
quotacheck -v -a -g -u -m  
verbose all group user no-remount
```

- Creates
  - /aquota.user
  - /aquota.group

# Quota management (III)

- Quota enabling

```
quotaon -v      -a  -g  -u
           verbose all group user
```

- Activates quota mechanisms, usually from `/etc/init.d/`

- Quota disabling

- `/sbin/quotaoff`

- Quota editing (`edquota`)

```
Disk quotas for user xavim (uid 500):
Filesystem  blocks      soft      hard    inodes      soft      hard
/dev/sdb1   3           16       32         2           0         0
```

- Data blocs and i-nodes quota
- It is not possible to edit the used blocks/inodes, but the limits of the quota can be changed

# Quota management (and IV)

- Visualize quotas: `quota -v`

```
Disk quotas for user xavim (uid 500):
```

Filesystem	blocks	quota	limit	grace	files	quota	limit	grace
/dev/sdb1	32*	16	32	6days	2	0	0	-

\* We are above the quotas, within the "hard" limit!!

- "Grace period"
  - Grace time where the user can reach the hard limit, it only raises warnings
  - If the grace period expires, then the system does not allow to go above the soft limit

# Other maintenance tasks

## Monitoring

- Free space (`df`)
  - Most systems reserve a (5%) of the space to be exclusively used by root
- Occupied space (`du`)

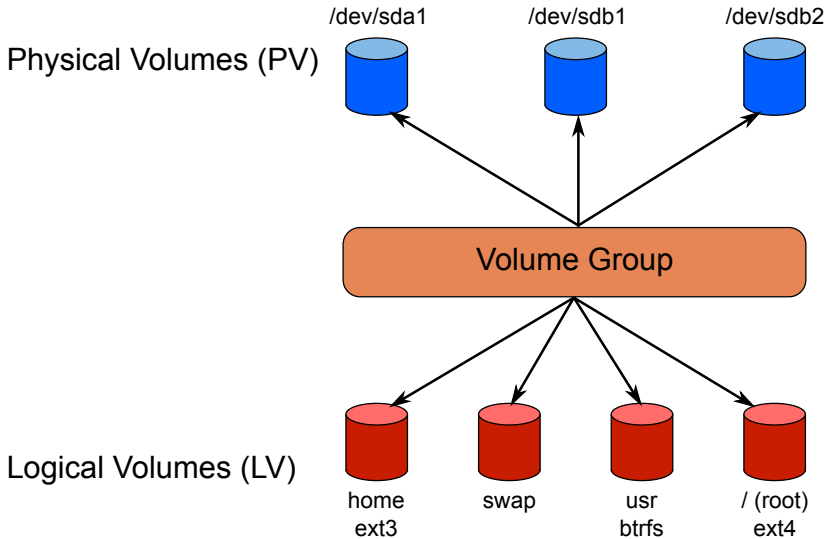
## Synchronization

- Write to disk the modified buffers
  - `sync`
  - Update daemon

# Outline

- 1 Introduction
- 2 Filesystems
- 3 Disk verification
- 4 Logical Volume Manager (LVM)**
- 5 Backups

# Logical Volume Manager (LVM) (I)



# Logical Volume Manager (and II)

- High level abstraction of the disk space
- Aggregates multiple physical partitions/disks
  - Allows to add more partitions to the volume
- It allows logical partitions within the volume
  - They can be assigned logical names
  - Customized distribution among the physical volumes
    - Resizing
    - Move
- Example: `/etc/fstab`

```
/boot      /dev/sda1    ...  
swap      /dev/vg00/swap ...  
/          /dev/vg00/root ...  
/home     /dev/vg00/home ...  
/usr      /dev/vg00/usr ...
```

# Outline

- 1 Introduction
- 2 Filesystems
- 3 Disk verification
- 4 Logical Volume Manager (LVM)
- 5 Backups**
  - Full Backup
  - Incremental Backup
  - Reverse Incremental Backup

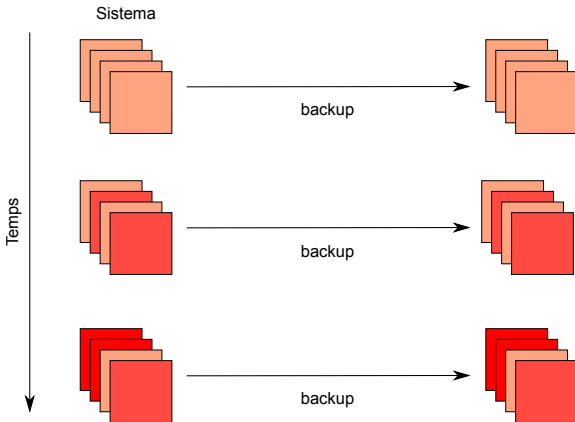


# Backups

- Data to copy
  - User data (home, mail, ... )
  - Program data (BBDDs, CVS, web, ...)
  - System configuration
  - Binary?
- Backup frequency
  - Data confidence
  - Data importance
- Backup types
  - Full Backup (all)
  - Incremental Backup (only changes)
  - Reverse Incremental Backup (only changes)

# Full Backup

- Always copy all the data
  - Fast to restore
  - Large size



# Incremental Backup

- Only backup the changed files

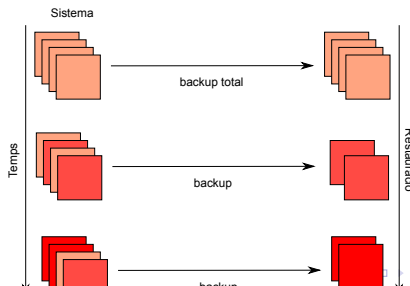
## Advantages

- Small size
- It can be done in any media

## Inconveniences

- Slower to restore
- The first one is equal to a full backup

Do not create a large backup sequence



# Reverse Incremental Backup

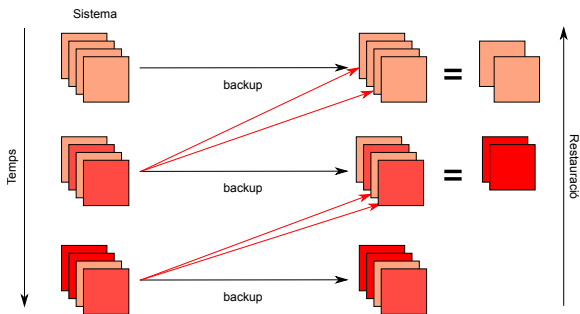
- Everything is copied but in the former backup only the changes are kept

## Advantages

- Fast to restore
- Little space

## Inconveniences

- Only possible in random access media



# Backups

## Physical support

- Floppy, disc, CD, tape, network. . .
- To consider:
  - Cost size ratio
  - Reliability
  - Availability
  - Usability
  - Speed

## Copy location

- Accident protection
- Fireproof boxes
- Keep some backups outside the company premises
- Stealing protection

# Exercise

Define a backup policy (data to backup, backup type, frequency, device, compression, . . . ) for a multi-user server within a company with:

- 500 Gb. disk and 80 users
- Mail
  - 50Mb per user
- Web pages
  - 20 Mb per user
  - 100 Mb company web
- Code repository
  - 10 GB distributed among 20 projects
  - Only 5 active projects

# Other considerations

- When having different servers it is recommended to
  - Define specific backup machines
    - cheaper
    - easy to administer

Tools: `tar+rsync/ssh`, `amanda`, `bacula`

# Personal Homework

- Task automation
  - Programming language: `bash`, `perl`
  - Information search: `find`, `grep`...