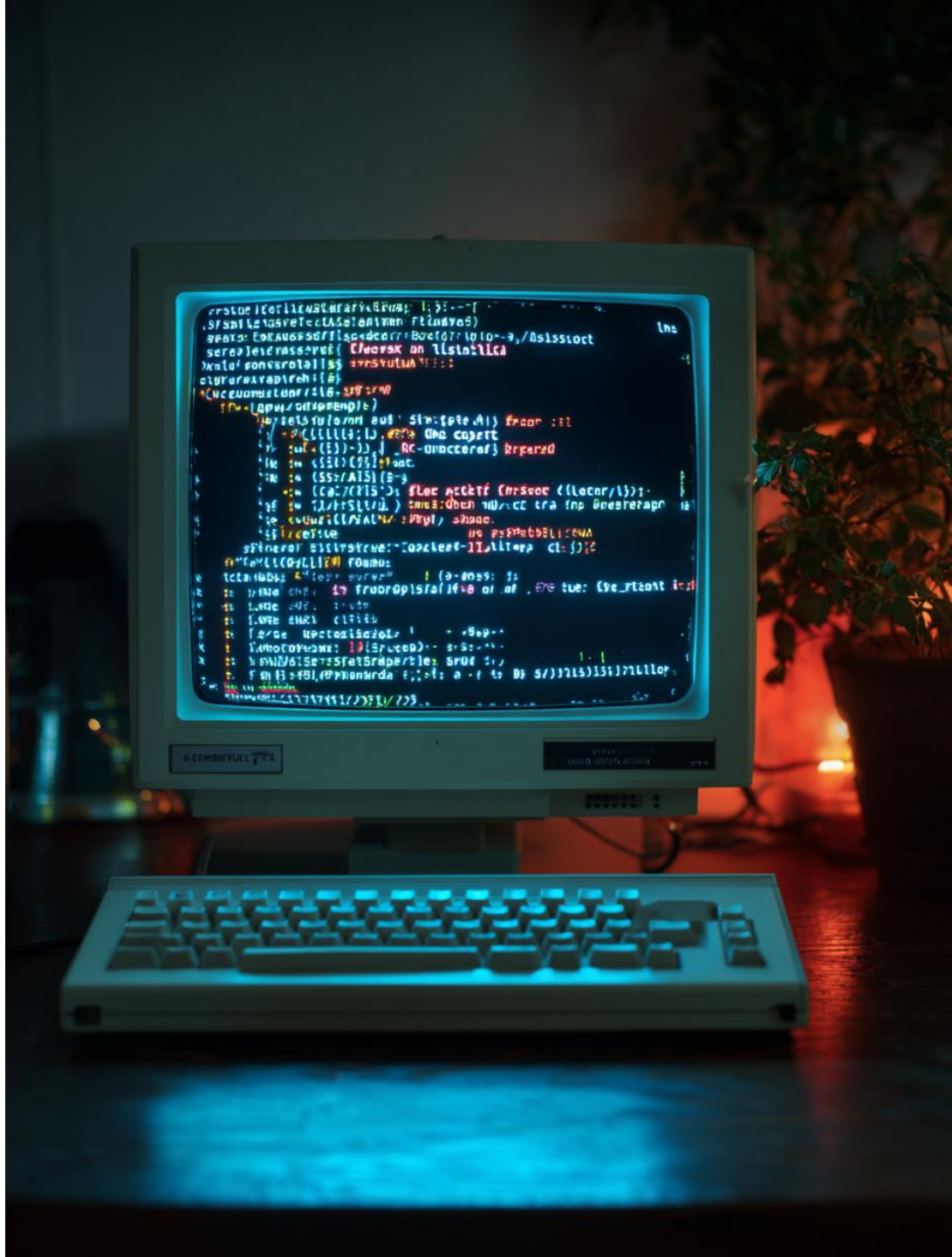
VŠB TECHNICKÁ UNIVERZITA OSTRAVA | FAKULTA ELEKTROTECHNIKY A INFORMATIKY | KATEDRA INFORMATIKY

Basics of Information Technology

Ing. Michal Radecký, Ph.D. MBA

Operating Systems, Linux

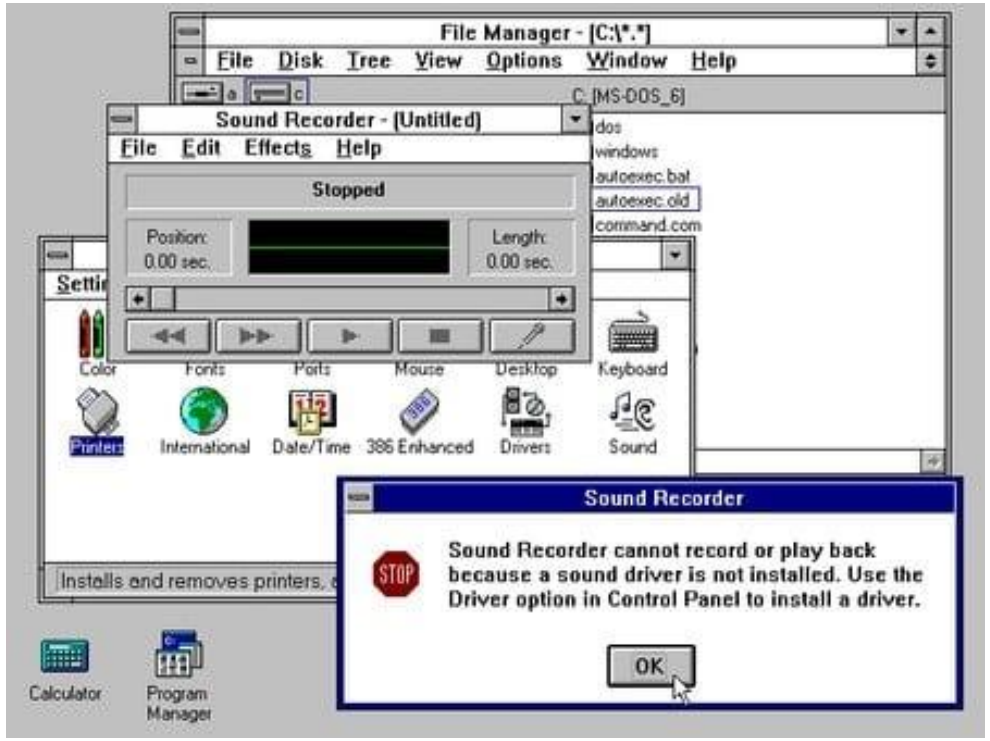


What is an operating system

- An operating system (OS) is the basic software that controls the operation of a computer or other device.
- It is an layer between hardware (processor, memory, disk, peripherals), applications and the user.

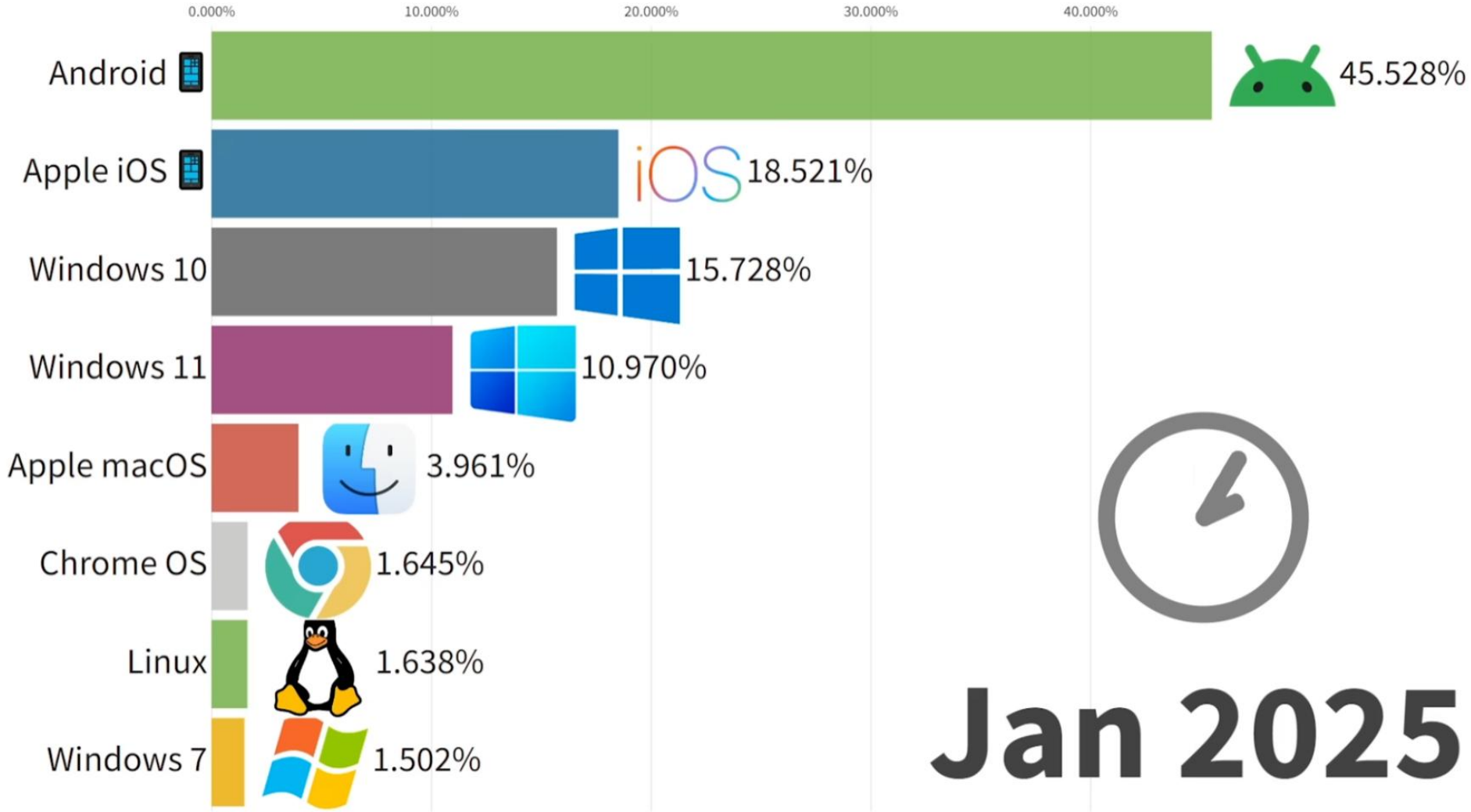
History

- 1950 – OS-free computing solution, firmly integrated (punch cards, relays, etc.)
- 1960 – batch processing, processing jobs in batches without direct user interaction
- 1970 – the first true operating systems, multiprogramming/multitasking, the first system with a graphical user interface (1973, Xerox)
  UNIX (1969, Bell Labs)
- 1980 – spread of computers among the public and individuals, various types of platforms incl. Graphical user interface(Xerox)
  MS-DOS (1981, Microsoft)
  Mac OS (1984, Apple)
  Windows 1.0 (1985, Microsoft)
- 1990 – focus on the general public, support for network connectivity (Internet), multimedia
  Linux (1991, Linus Torvalds)
  Windows 3.1 (1993, Microsoft)
- 2000+ - focus on performance, graphical interface, security, connectivity, the emergence of an OS for mobile platforms
  iOS (2007, Apple)
  Android (2008, Google)



https://www.youtube.com/watch?v=kK7L2ISGucM

Today's market share



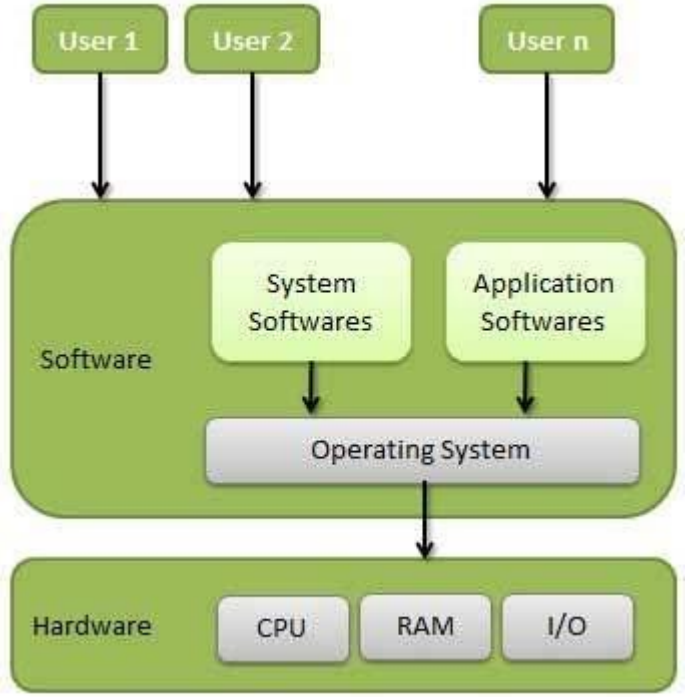
https://www.youtube.com/watch?v=29qnXTw0qr0, https://gs.statcounter.com/os-market-share

zdroj: https://www.hostragons.com/cs/blog/vyvoj-operacnich-systemu-od-unixu-k-modernim-systemum/

Basic functions of operating systems

- **Resource management**
  manages hardware - CPU, memory, storage, and I/O devices
- **Process control**
  starts and terminates programs (processes), allocates CPU time, scheduling tasks
- **Memory management**
  allocates and frees up memory to processes, sharing memory between different processes, managing memory
- **File system management**
  allows you to read, store and organize data on disk (files, directories)
- **User interface**
  provides an interface (GUI or command line), allows the user to interact with the system/applications
- **Securing**
  ensures the protection of the system and data against unauthorized access, user permissions

Types of operating systems



- **according to the way they interact with the user**
  text, graphic, batch, etc.
- **by number of users**
  single-user, multi-user
- **by how the tasks are processed**
  single-tasking, multitasking
- **by method of deployment**
  General, Industrial, Real-Time, Network, Specialized
- **by device**
  desktop, server, mobile, embedded (embedded)
- **by internal architecture (kernel)**
  monolithic core, microkernel, hybrid
- **by hardware binding**
  number of processors, processor architecture (ARM, x86), specialized (GPU), virtualization

Windows – proprietary, flexible hardware, hybrid, strong application support, simplicity

macOS - closed, hard-wired to hardware, stability, UNIX-based, hybrid core, Apple ecosystem

Linux - open, flexible, secure, monolithic core, used in servers and in the cloud

Important terms

- **Process**
  running program in memory, contains code, data, state (registers) and its own memory, mutually separated, OS controls the execution of processes (process scheduler)
- **Thread**
  smaller unit of running inside the process, shares resources within the process, possibility of simultaneous running (multithreading)
- **Multitasking**
  the ability of the OS to run multiple processes "simultaneously", the process scheduler decides on processor allocation and process switching, a technical solution dependent on the architecture
- **File system**
  the way in which the OS organizes and works with data (files, directories) on storage media, it also deals with access rights, various types: FAT32, NFS, exFAT, ex4, APFS
- **Driver**
  software that is part of the OS and allows communication with a physical device (I/O device), an interface between the OS and a specific device
- **Booting**
  the process of booting the computer (from initializing the HW to loading the OS), includes the HW test, firmware boot (BIOS/UEFI), booting the kernel using the bootloader, initial configuration, user login

Linux

- The inspiration was UNIX from 1969 (multi-user, multitasking)
- 1991 First version 0.01 of the Linux kernel
- By merging with the GNU project (compiler, shell, utility), a full-fledged GNU/Linux OS is created

- Linux is a **free and open source** operating system (open-source OS), a large community
- Linux is the kernel of the operating system
- Systems that use this kernel are referred to as Linux distributions (Ubuntu, Fedora, Debian) packaging systems, configurations, UI/GUI, etc.

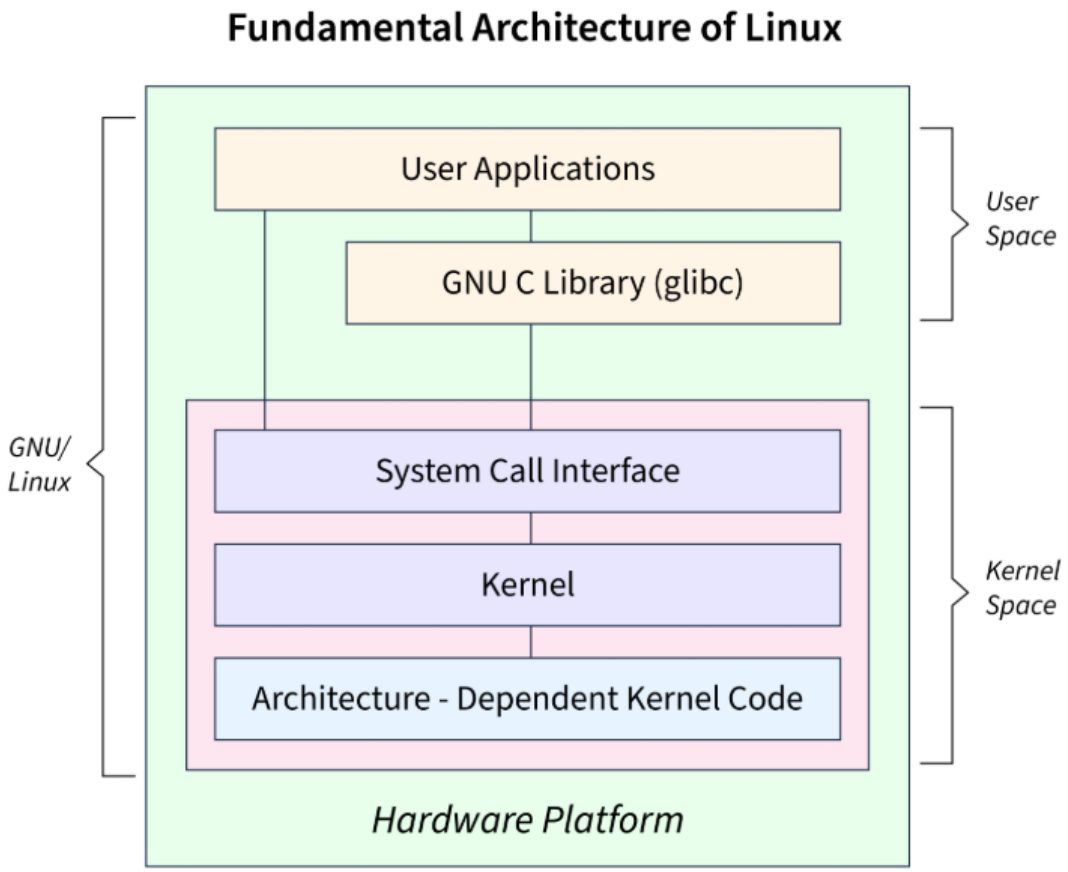
Servers, supercomputers (95+% TOP500), Android, Embedded/IoT (smart TVs, network devices, cars, ...)

Main features

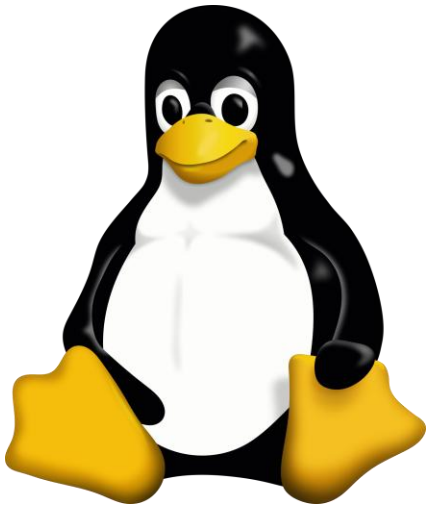
- Monolithic kernel – all basic services run in the kernel, extensibility by drivers
- Multi-user – multiple users at the same time, permission system
- Multitasking – multiple processes at the same time, control is determined by the OS
- File-oriented – everything (processes, devices, etc.) is a "file", a hierarchical structure
- Cross-platform – high portability to different devices and environments

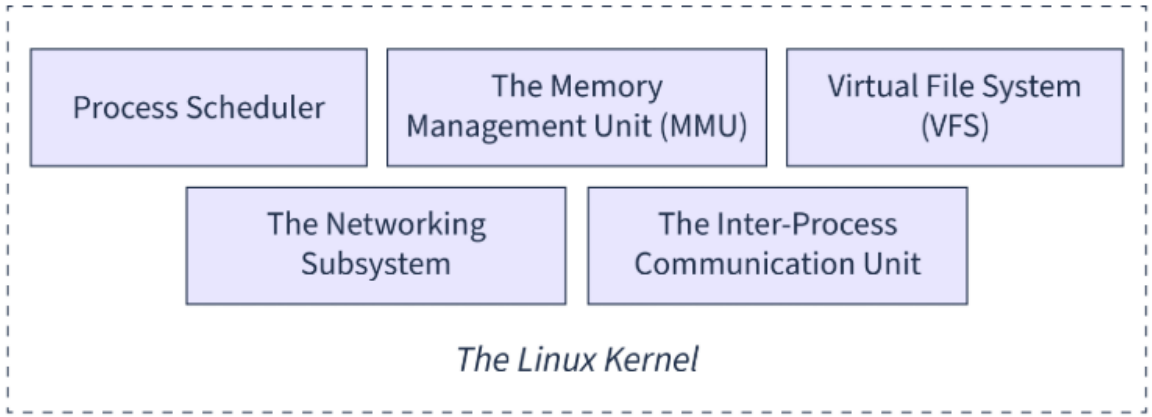
zdroj: https://www.scaler.com/topics/linux-kernel-architecture/

Linux



SCALER Topics





SCALER Topics

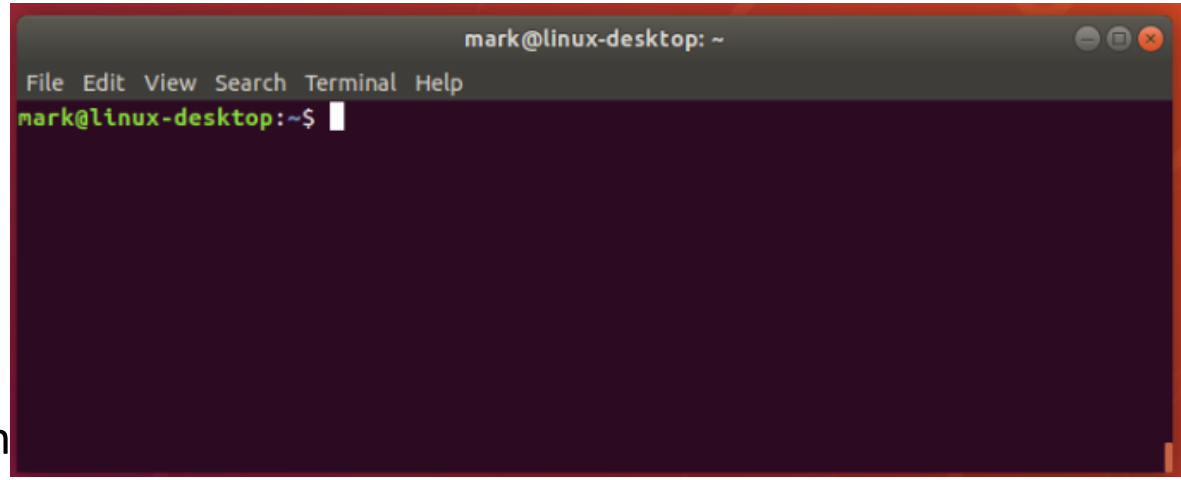
zdroj: https://ubuntu.com/tutorials/command-line-for-beginners

Shell

- Shell is the interface between the user and the operating system.
- It allows the user to enter commands (which are interpreted and passed on to the execution of the OS), including the execution of programs.
- CLI vs. GUI, typically understood as a text-based environment (CLI – Command Line Interface)

Various implementations

- sh – original UNIX shell
- bash – most commonly used as the default in distributions
- zsh, fish, csh



Main options

- Interpreting Commands (Parameters) and Executing Com
- Command history
- Command chaining
- Scripting
- Work with inputs and outputs (redirecting)

syntaxe:

```
command [params] [arguments]
        -x --xx
```

zdroj: https://blog.fourninecloud.com/linux-file-system-hierarchy-explained-1d80b2cee03c

Directory structure

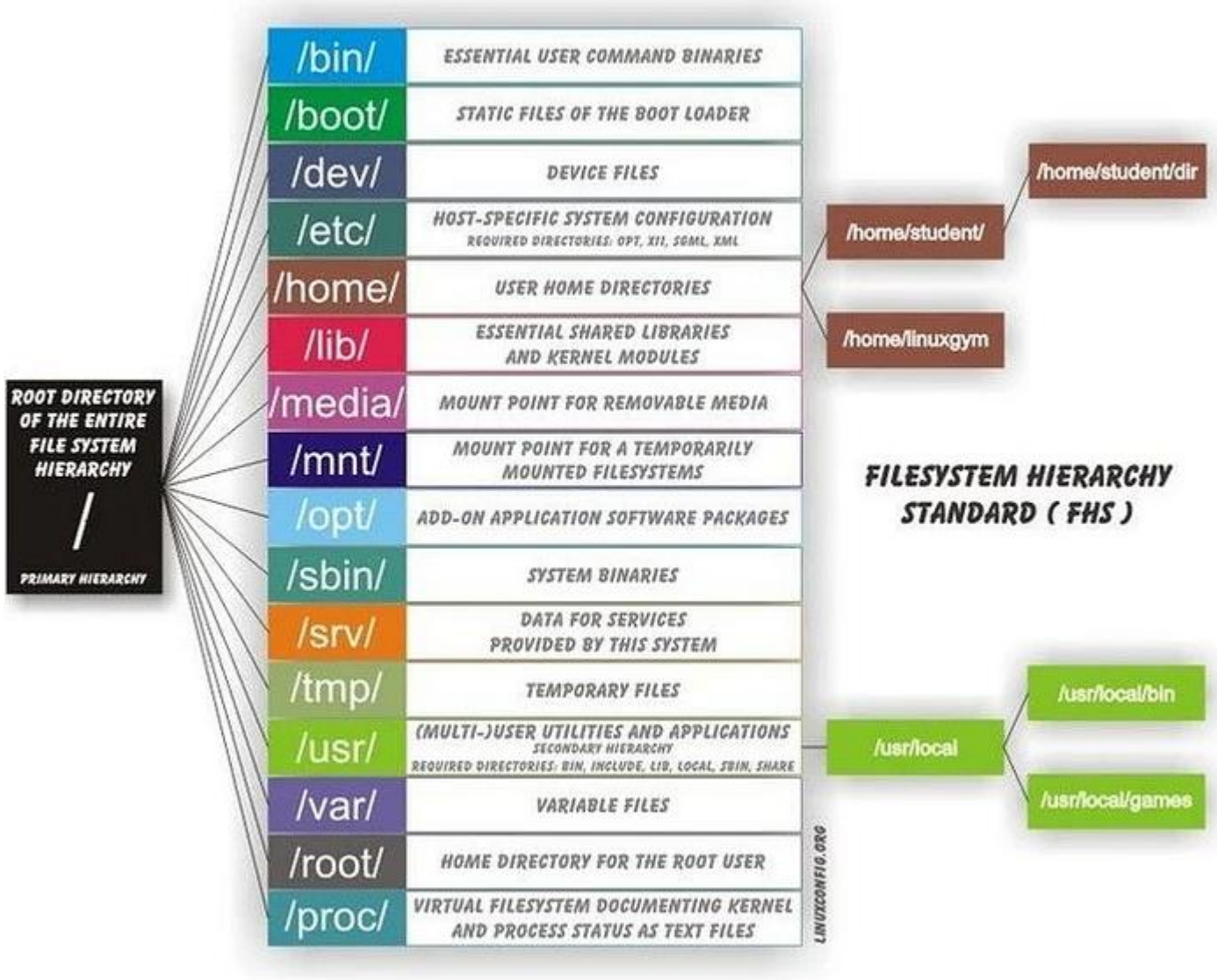
pwd
Print Working Directory

cd
Moving in a directory structure
No parameter - home direktory (~)

cd ..
cd /
cd home
Moving in a directory structure

- Absolute and relative paths
- Home Directory (~)
- Hidden files (.x)

mkdir, rmdir



Working with files/directories

`ls`
File list (-la)

`mv`
Move/rename a file

`rm`
Remove a file

`cp`
Copying a File

`chmod, chown, chgrp`
Change the permissions, owner, and group of a file

- Wildcards: *, ?, [..], [!..], [^..], {..}
- Understanding permissions
  - Owner, Group, Others
  - Read, write, execute
  - rwxr-xr-- = 754

`cat`
Listing the contents of the file

`less`
File content viewer

`head`
Listing the beginning of the file

`tail`
File end dump

`grep`
Search in file contents
Returns rows with result
Supports regular expressions

`vi, vim, nano`
File content editors

`echo`
Creates output

`sort`
Sorts the input lines

Output redirection and command chaining

Each process works with 3 "streams":

- **stdin (0)** - Standard input (keyboard, file)
- **stdout (1)** - Standard Output (Screen Output)
- **stderr (2)** - Error Output(s)

>
Output redirection (new file)

>>
Output redirection (append)

<
Input redirection

2>
Redirecting Error Output

&>
Redirecting stdout and stderr

;
Sequential order of commands

&&, ||
Execution of AND and OR statements

|
(pipeline)
The output is passed as input to the next command

&
Running a command in the background

Examples

```
cd /tmp
mkdir testdir
cd ..
rmdir testdir/data

touch file1.txt
echo „first line" > file1.txt
echo „second line" >> file1.txt
cp file1.txt file2.txt
mv file2.txt new.txt
rm new.txt

ls -la
chmod 644 file1.txt
chmod u+x file1.txt
chown user:group file1.txt

head -n 1 file1.txt
tail -n 1 file1.txt
```

```
ls > list.txt
echo "New line" >> list.txt
sort < list.txt
ls /nodir 2> chyba.txt
ls /nodir &> vystup.txt

echo „First command příkaz" ; echo „Second command"
mkdir new && cd new
cd nodir || echo „Directory not found"

ls -l | grep ".txt"
ps aux | grep firefox

sleep 60 &

dmesg | grep usb > usb_log.txt
cat file.txt | sort | uniq -c > stats.txt
command1 && command2 | tee output.txt
```

Other commands

`man, --help`
Command documentation

`ps (ps aux)`
Listing of currently running processes

`top, htop`
Listing of currently running processes

`kill`
Terminates a specific process

`sudo`
Runs an individual command as a superuser

`whoami`
Login of the logged-in user

`id`
User Information

`shutdown`
Shuts down or restarts your computer

`fg, bg, jobs`
Switching and overview of stopped processes

`lsblk`
View mountable storage devices

`mount, umount`
Connecting/disconnecting the storage media to the directory

`ip addr (ifconfig)`
Information/setting of network interface parameters

`wget, curl`
Commands for getting content online (URL)

`Ctrl+C, Ctrl+Z, …`
Interaction with active/running processes and the shell