

Introduction to the Linux OS

Peter Huszár

KFA: DEPARTMENT OF ATMOSPHERIC PHYSICS

Pavel Řezníček

ÚČJF: INSTITUTE OF PARTICLE AND NUCLEAR PHYSICS

September 29, 2021

Overview and Organization

Introduction to the Operation system Linux, focus on the command line, scripting, basic services and tools used in (not only) physics: tasks automation in data processing and modeling

Organization

- Graded Assessment (KZ): attendance to the lectures, worked out homeworks

Literature

- C. Herborth: Unix a Linux - Názorný průvodce, Computer Press, Praha, 2006
- D. J. Barrett: Linux - Kapesní přehled, Computer Press, Praha, 2006
- M. Sobell: Mistrovství v RedHat a Fedora Linux, Computer Press, Praha, 2006
- M. Sobell: Linux - praktický průvodce, Computer Press, Praha, 2002
- E. Siever: Linux v kostce, Computer Press, Praha, 1999
- **Number of online sources...**

Study materials and homeworks

- <http://kfa.mff.cuni.cz/linux>



- 1 UNIX systems, history, installation, basic applications
- 2 Structure of the Linux OS, file systems, hierarchy of the file system
- 3 Command line, shells, remote access (ssh, ftp)
- 4 Processes and their administration, basic system commands, packages, printing
- 5 Users, file and directory permissions
- 6 Work with files and directories, file compression, links, partition
- 7 Text-file processing commands, redirection, pipeline
- 8 Regular expressions
- 9 Command line based text editors
- 10 User and system variables, output processing
- 11 Scripts: basic construction, conditionals, loops, functions, automation
- 12 Networking, server-client services: http, (s)ftp, scp, ssh, sshfs, nfs
- 13 Programming in Linux (examples of Fortran, C/C++, Python), version control systems, documents in Latex

Operating systems, two basic types:

- 1 **Windows**
- 2 **UNIX** (**Mac** and **Android** are also UNIX-based)
 - Windows historically more focused on user-PCs
 - UNIX systems used on servers (mail, web, computing, networking)
 - Nowadays the abilities of both systems are close each to other

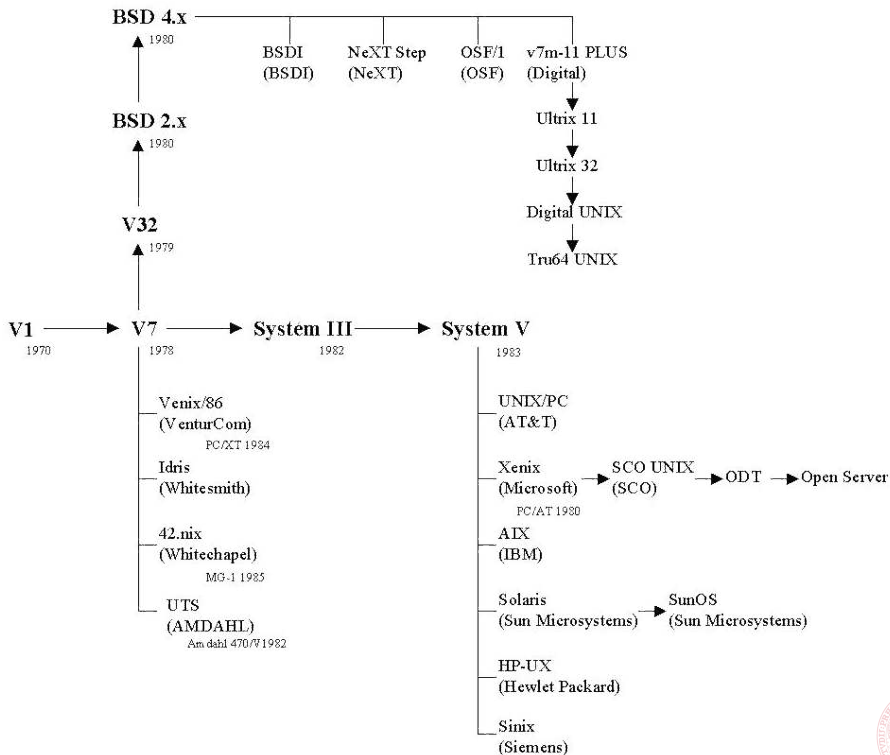
Why Linux ?

- Powerful command-line (*cmd*)
 - Applications started from *cmd*
 - System control via *cmd*
 - Programs control, compilation, modifications and debugging via *cmd*
 - Scripting / programming using shell
- Natural remote access (including graphic windows), remote administration
- Open source & free applications, support through wide-community

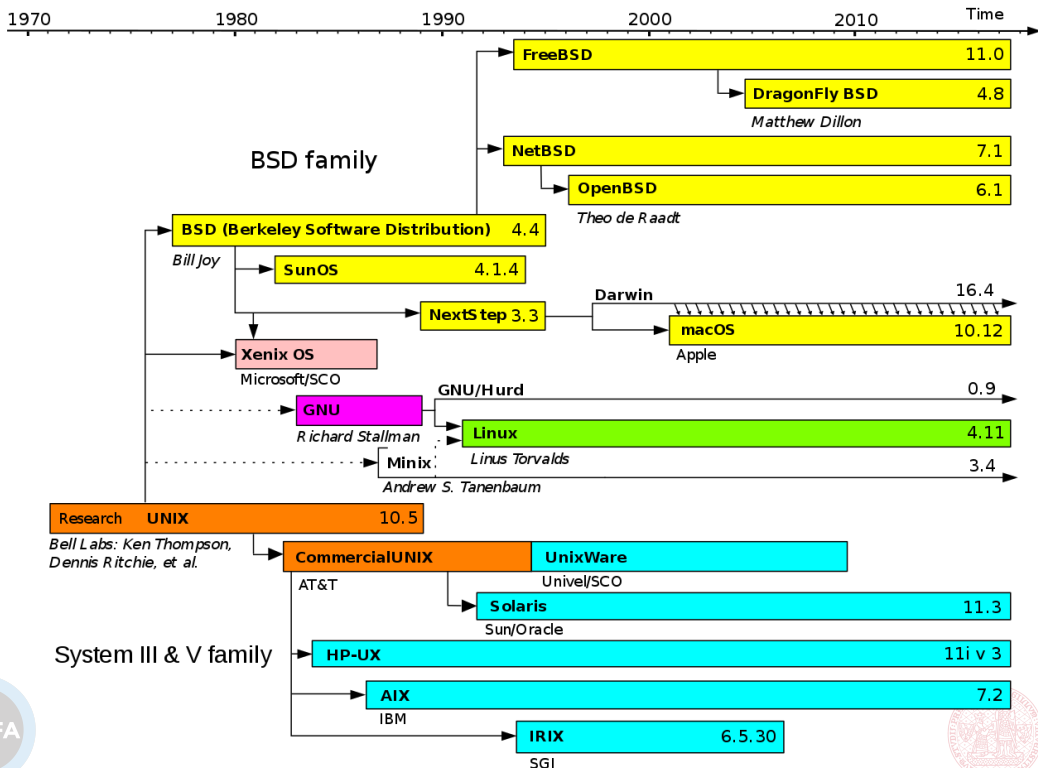
UNIX - trademark of operating system created in Bell Laboratories of AT&T company in 1965

Path to UNIX

- **1964:** Bell Telephone Laboratories, General Electric and MIT develops OS Multics (MULTIplexed Information and Computing System)
 - Unfinished
 - *Kenn Thompson, Dennis Ritchie, Brian Kernighan*
- **1969:** Bell Labs. withdraws from Multics project, *Thompson* writes basic OS (kernel, shell), editor and assembler for the PDP-7 computer
- **1970:** *Kernighan* suggests the name of the OS UNIX (firstly UNICS)
- **1971:** *Thompson* asks for new PDP-11 computer for further development (rejected); *Thompson* pretends development of automatized office → computer assigned for text processing
- **1973:** *Thompson* rewrites Fortran language - languages B, *Richie* rewrites UNIX into C language for better portability
- **1978:** UNIX v7 released for Universities (Berkeley) - UNIX divided into two parts:
 - AT&T (System III, System V)
 - Berkeley (BSD 3.0)



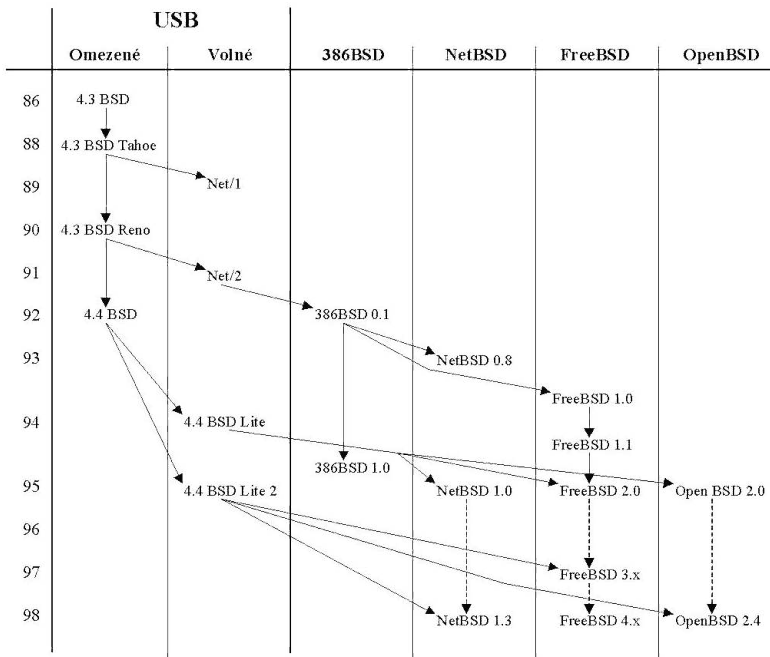
UNIX Systems



- SUN: Sun OS, Solaris
- Silicon Graphics: Irix
- DEC: Ultrix, Digital Unix
- IBM: AIX
- HP: HP-UX
- Siemens Nixdorf: SINIX
- Novell: UNIXware
- SCO: SCO Unix
- FreeBSD, NetBSD, OpenBSD, ...
- **Linux**, Mac OS X
- Android OS

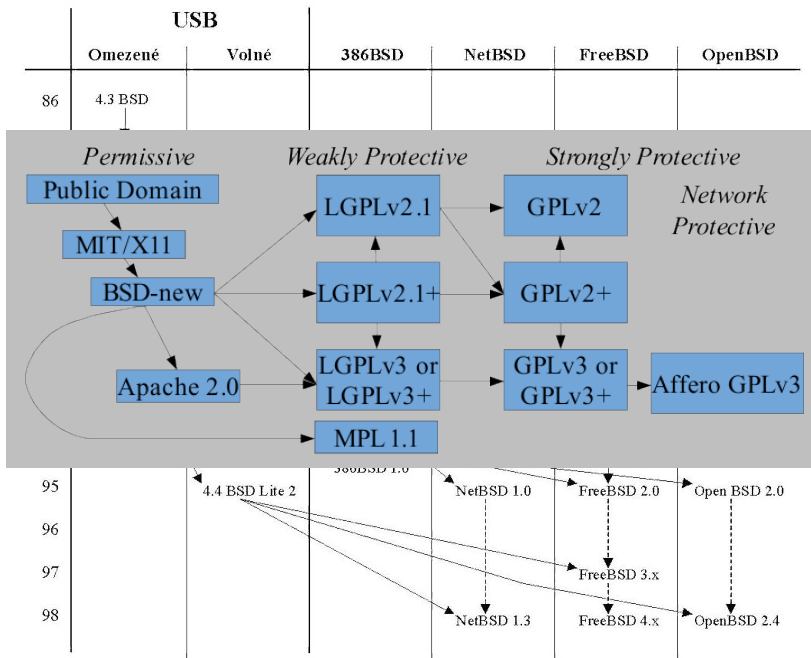
Non-free source codes → release of open and free versions

- Development through free versions



Non-free source codes → release of open and free versions

- Development through free versions



Linux foundation

- **1984:** *Richard Stallman* creating GNU (GNU's not UNIX)
 - Attempt to create free (license) system
 - Creating General Public License (GPL)
 - *Freedom to run, study, share and modify the software*
- **1991:** *Linus Torvalds* trying to create freely available system
 - Based on Minix
 - Writing kernel of the system

GNU Linux - Linux kernel, tools and GNU libraries

Two SW development models:

Bazaar

- Source code developed over the Internet in view of the public - *Linus Torvalds*



Tux

Cathedral

- Source code available with each software release, but code developed between releases is restricted to an exclusive group of software developers

Heckert



ÚČJF
MFF UK

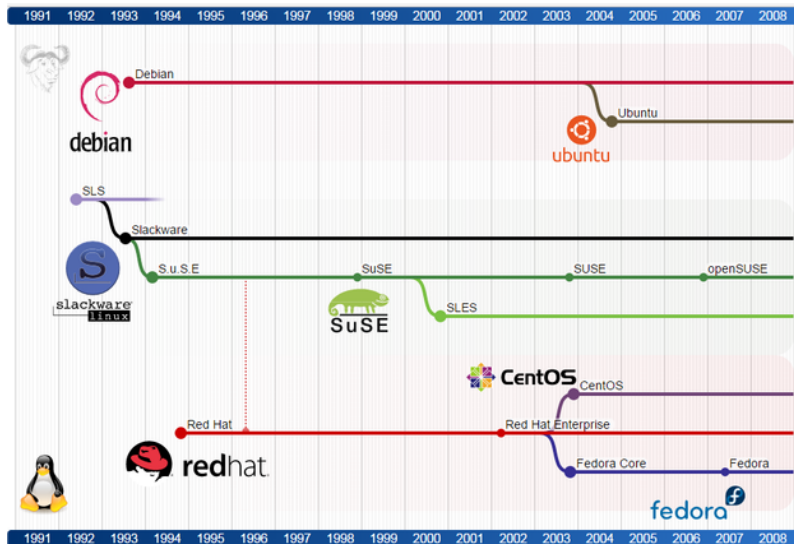
- **Kernel** of the system: drivers, processes, memory, filesystem management
- **Filesystem**: different from Windows
- **Console**: text-based interface
- **Graphical interface**:
 - Management allowing remote transfer or graphics windows or even whole screen
 - Terminal (*cmd*)
 - Common applications: office, web-browsers, multimedia, file-browsers, coding editors, ...
- **Software repositories**
- Natural multi-user and multitasking

Based on free SW

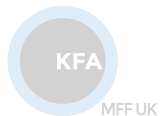
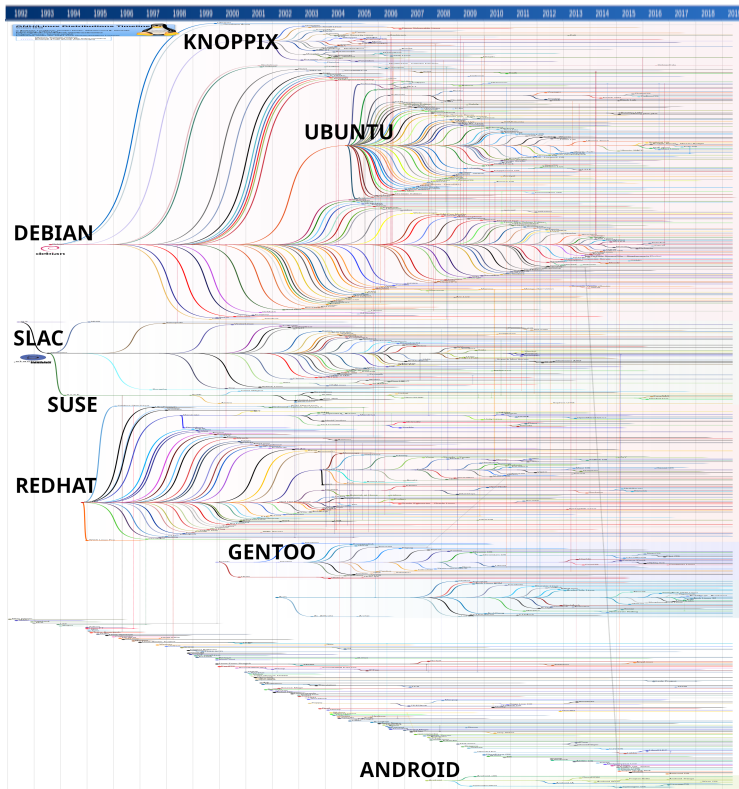
- Practically all parts of the system have number of variants
- Number of Linux distributions:
 - Various graphical interfaces
 - Various choices of default applications
 - Various program and library versions (stable vs. bleeding edge)
 - Various SW repository types

Linux Distributions

- List of components: kernel of the operating system GNU/Linux, libraries and other GNU tools
- Have its own installation image, packages and their repositories
- Historically 3 distributions: *Debian*, *Red Hat* and *Slackware* expanded into several hundreds, including *Ubuntu*, *openSUSE*, *Fedora*, *Mandriva*, *Gentoo*, ...
- Most distributions have their own Live versions: run from USB stick or CD/DVD



Linux Distributions Chart (2019)

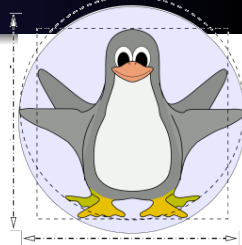




- *Debian* is one of the **most extensive** distributions
- Fully **developed by the community**
- Supports 11 platforms:
 - amd64, i386, Arm, PowerPC, mips, s390, ...
- More than 50000 packages
- Includes even non-Linux kernels (FreeBSD, NetBSD)
- Very **specific development cycle**:
 - Deployment of **stable** release (getting only minor security updates) once per ~ two years
 - Very stable and secure Linux, often used for servers
 - In the mean time new packages and package versions are tested in **experimental**, **unstable** and **testing** branches
 - After certain time the **testing** release becomes **stable**
 - For impatient users, there are also **backports** for the **stable** release
 - Using **testing** distribution can serve as relatively stable "rolling distribution" (continuously getting updates without and "release" dates)
 - Using **unstable** release as "rolling" is not that comfortable as it can contain serious incompatibilities
- Debian (and most of its derivatives) uses package system based on ***.deb** packages and system **apt**
 - Automatic resolution of package dependencies, conflicts, diversions, alternatives
 - Pre/post-installation scripts
 - Complex system to create or build own packages
 - Widely considered as best packaging system



- *Ubuntu* is a system based on *Debian*
- Free, with both community and professional support
- Ubuntu community is based on principles described in the [Ubuntu Manifesto](#):
 - Should have the freedom to download, run, copy, distribute, study, share, change and improve their software for any purpose, without paying licensing fees.
 - Should be able to use their software in the language of their choice.
 - Should be able to use all software regardless of disability.
 - *Southern African philosophy of ubuntu (literally, "human-ness")*
- Suitable both as user-PC as well as for servers
- Supports **most common architectures**: PC 32bit (i386), PC 64bit (amd64) and PowerPC (older Apple iBook, Powerbook, G4, G5)
- Half-year release, long-term supported release every 2 years
- **Unity** is the default GUI (Graphic User Interface)
 - MAC-like, touchscreen friendly
 - However, number of variants with different GUI exists (*Kubuntu, Xubuntu, Lubuntu, ...*),
 - As well as specific-focus derivatives (*Edubuntu, ...*),
 - or enhanced distributions (*Linux Mint, ...*)



- **KNOPPIX** is a primarily **Live OS**
- Already after installation contains wide range of SW, automatic HW detection and HW support (sound, graphics, peripherals)
 - The CD contains up to 2 GB of compressed SW
 - The DVD version up to 8 GB of compressed SW
- Can serve as "rescue" Live system
- Can be installed as ordinary user-PC as well
- Derivatives: localized and enhanced system *Danix*



- *Red Hat* is a commercial Linux
- **Enterprise-level support**
- Packages system uses **.rpm* files (2nd widely-used package system)
- Number of free derivatives with community-support only



- *Fedora* is a **non-commercial Red Hat derivative**
- Development (community) **supported by the Red Hat** company
- Serves as testing platform for the commercial *Red Hat* releases
- **Progressive**, implementing new features very soon
- Focused on user-PC, suitable for beginners
- Wide range of SW in the distribution
- Supports **most common architectures**: i386, amd64 and PowerPC
- Very **strict in licensing**, e.g. yet recently missing ***.mp3** support



- *CentOS* is another *Red Hat* derivative
- Originally started as independent distribution, but transferred under *Red Hat* (developers from the *Red Hat* open-source team)
- Free and community supported
- Often **used on servers**
 - Used in CERN (European Organization for Nuclear Research)



- *Mandriva* is originally based on *Red Hat*
 - Started in 1998 started by *Geld Welb*, attempting to make more comfortable GUI experience
 - The modified distribution, called *Mandrake*, was put on servers and *Geld Welb* left for holidays
 - Strong response from users after return, offers to contribute from developers and testers
 - Nowadays, *Mandriva* has number of developers in France and USA
- Focusing on **office-PC and multimedia-PC**
- Wide range of SW in the distribution
- Easy maintenance
- Strong **involvement of users** on the final shape of the distribution
 - Beta-testers
 - Suggestions for improvements and changes



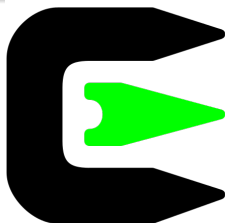
- *Slackware* was the **first widely used Linux distribution**, created in 1992 by *Patrick Volkerding*, who remains its only official developer
- Although many contributors bring new packages, the existence of the only one developer makes the concept of the distribution unified and development trouble-free
- *Slackware* is known for **simplicity and clarity**:
 - Using easy to understand text configuration files
 - Transparency in every process
- **Simple packaging system**, able to only install and remove packages:
 - No dependency resolutions
 - No automatic updates
 - → Very flexible in what is being installed on the system
 - Still there are enhancements that overcomes the simplifications above



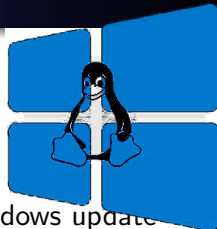
- *SUSE* was created in 1992 in Germany
 - First release in 1994
 - 2003/2004 bought by **Novell**
- Release of boxed version with manual first (together with Live DVD for preview only), online after several months
- Possibility to buy professional support
- Later project *openSUSE*:
 - Community based program sponsored by Novell
 - Easier and free access to *SUSE* Linux



- **Gentoo** uses system similar to the *BSD* ports, called **Portage**
 - **Portage** is a very flexible packaging system (flexible in the installation and maintenance of the SW)
 - SW is being built from source at the time of installation
 - The **Portage** system allows to set various installation configurations via use-flags
 - Includes build-dependencies
 - Safe installation and deinstallation of packages (via so called sandbox)
 - Protection of config files etc.
- **Very flexible installation**, e.g. one can built system from source-code with selected optimization → **Gentoo** is often considered as **metadistribution**
- Active development, rapid fixes and updates
- Half-year releases



- *Cygwin* is a Linux system that is compiled and runs natively on Windows
 - No emulation as virtual machine
- Settings partly bound to windows (users)
- Possible to launch windows programs
- Nowadays owned by *Red Hat*



- *Windows Subsystem for Linux* (WSL) introduced in 2016 Windows update
 - Shell on Windows / *Ubuntu* on Windows
 - But not really limited to one distribution
 - Enable in "Windows Features" settings
- Allows to run linux binaries (64bit) natively on Windows 10
- Can also run graphical applications (after some tuning...)
- Superuser (root) privileges limited (can't change Windows config - network etc.)
- Underlying principle: Linux system calls translated to Windows system calls
 - Performance worse than on native Linux
- Special filesystem



- *VirtualBox* and *VMware* create virtual machine, in which another OS can run (Windows, Linux, MAC)
- Commercial SW, *VirtualBox* is open-source and free for personal or educational use
- Makes use of Intel/AMD hardware-assisted virtualization
- Can run several Linux installations on Windows in parallel (if enough resources on PC)

Linux Installation Notes

Install Linux Inside Windows

- Not genuine Linux experience: administration and hardware management may be limited by what Windows allows...
- No need for disk repartitioning
- Easy to uninstall

Cygwin

- Simply follow installation instructions, installs as an ordinary windows application

Windows Subsystem for Linux

- Enable in "Windows Features" settings
- Install *Ubuntu* via Microsoft Store: search for "Run Linux on Windows" and choose distribution

VirtualBox

- 1 Install *VirtualBox*
- 2 Create new virtual system: allocate part of RAM and disk space (VDI type)
- 3 Downloaded *Ubuntu* ISO image and install the system in *VirtualBox* via "Start" button

- 1 Free disk space in Windows
 - May need to switch off (temporarily) disk encrypting, otherwise disk repartitioning won't work
 - Disk cleanup, including cleanup of system files (downloaded updates, old win versions, log files from upgrades, ...)
 - Compress system files
 - Disable hibernation (removes large hiberfile.sys file)
 - Make small memory swap file (shrinks large pagefile.sys file)
 - Make small space for restore points
 - Make small shadow storage space
 - Cleanup C:\SWSETUP which contains driver updates installation files
 - Backup notebook recovery partition to USB stick and then delete the partition; not the $\sim (1 - 2)$ GB at disk beginning, but the $(\sim (10 - 20))$ GB at the end !
 - → **Can shrink Windows partition down to ~ 30 GB (but will need USB for larger Windows updates)**
- 2 Most distributions have (Live) CD/DVD/USB to start the installation
 - Install the *.iso image on USB using Rufus program on Windows
- 3 No complications when following instructions...
- 4 However, it might be good to have partitioning of disk under control, instead of relying on the partitioning the Live Linux performs



Disk Repartitioning

- 1 Use Live Linux, e.g. specialized *System Rescue CD* using *GParted* program
- 2 Follow instruction to install on USB:
 - With *Rufus* on Windows
 - With `dd if=name.iso of=/dev/sdX ; sync` on Linux
- 3 May need to play with Boot/Startup sequence and with UEFI/SecureBoot in BIOS to allow boot from the USB stick
- 4 Most likely will have to create logical partition, since the default partition table does allow more than 4 primary partitions
 - Windows, Windows recovery, Linux root, Linux home, swap

Partitions to create

- Root partition '/' of type *ext4* and size $\sim (20 - 50)$ GB
- Home partition '/home' of type *ext4*
- Swap partition of type *swap*
 - $> 2 \times$ RAM for small RAM systems (< 4 GB)
 - $> 1 \times$ RAM for middle-size RAM systems (< 8 GB)
 - $0.5 \times$ RAM for large RAM systems (> 8 GB) and when hibernation-to-disk is not needed
 - Ubuntu recommendations: $SWAP = \sqrt{RAM}$ when hibernation-to-disk is not needed, add $1 \times$ RAM when it is needed

When touching (resizing) Windows partition, always make only one step at a time and reboot to Windows for automatic check of the newly-repartitioned disk. Otherwise Windows may become unbootable ! (although there is a fix...)

Dual-boot features

- May need to re-point WinRE (Windows Recovery Environment) to the correct partition on disk (usually is at the end of the disk)
 - `reagentc /info`
 - `reagentc /setreimage /path \\?\GLOBALROOT\device\harddisk0\partition2\Recovery\WindowsRE`
 - `reagentc /enable`
- Windows recovery to notebook factory settings may not work anymore
- Putting PC to hibernate on disk in both Windows and Linux at the same time may lead to loss of data if Windows disks are mounted in Linux

Linux boot recovery (Grub)

- May happen if / after reinstalling Windows
- Follow instructions at [System Rescue CD](#)

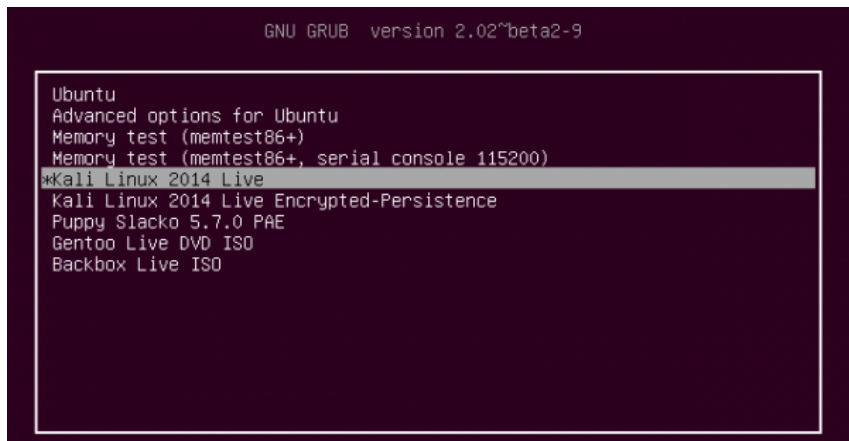
- 1 Do not forget any data on your Linux disks :)
- 2 Set bootable flag (e.g. in *GParted*) program back to the Windows system partition
- 3 Install back the Windows *Master Boot Record* from within Linux (or Live Linux); several ways are possible, depending on what SW is available on your Linux (commands below to be run as root):
 - A. `dd bs=440 count=1 conv=notrunc if=/usr/lib/syslinux/mbr.bin of=/dev/sdX`
(using `syslinux` package, `mbr.bin` may be located at different path)
 - B. `lilo -M /dev/sdX mbr` (using `lilo` package)
 - C. `install-mbr -i n -p D -t 0 /dev/sdX` (using `mbr` package)
 - D. `apt install boot-repair` from `ppa:yannubuntu/boot-repair` repository
- 4 Boot to Live Linux (*System Rescue CD*)
- 5 Wipe data (rewrite with random contents) using `schred` / `wipe` commands
 - `shred -v -z -n 10 /dev/sdXY`
- 6 Delete the Linux partitions and resize back the Windows one(s)

Again, the deletion and resizing should be done in single steps, followed by reboots to Windows !

Linux Graphical User Interface and Software

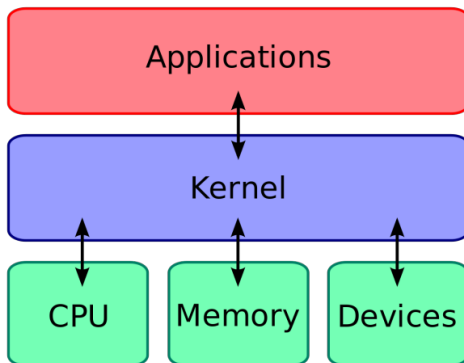
Loader of the Linux kernel

- Most current Linux distributions use *GRUB* as the default bootloader
- It is loaded into the *Master Boot Record* of the disk
- Allows to load other systems too → dual-boot
 - The installed systems are usually automatically recognized and added to the boot menu
- Allows to perform also memory tests
- Allows to add parameters to the kernel loading
- Allows rescue mode system load (e.g. when a disk fails to mount)



Linux kernel is the core interface between a computer's hardware and its processes

- Controls processes: create, stop, suspend, communicate, access to peripherals, ...
- Manages filesystems: allocation, freeing, protection, keeping consistency, ...
- Memory management: allocation, protection, swapping of unused memory to disk, ...
- Management of processes CPU time sharing: planning algorithm, assignment of time quantum, priorities, ...
- Can have installed several kernel versions for the same system; GRUB allows to choose the one to boot



Text-based interface to the system

- Provides a way for the kernel and other processes to output text-based messages to the user, and to receive text-based input from the user
- Works in parallel to the GUI
- Especially useful when GUI is not starting or not installed (servers)
- Limited support for localization, mouse and copy & paste; can be enhanced by SW packages as *consolation* or *gpm*

Switch between console and GUI (X-server)

- Virtual terminals: **Ctrl+Alt+F1** (F1-F6)
- GUI: **Ctrl+Alt+F7** (and above, if more X-sessions are running)

X display manager is a graphical login manager which starts a login session on an X server from the same or another computer.

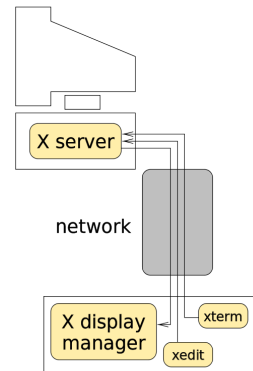
- Natural use of graphical applications on remote server
- Number of alternatives, distributions typically binds them with the used *Window Manager*, resp. *Desktop Environment*
- Some distributions prefer to use light *Display Managers*
- Choose user to login, *Desktop Environment* to start, switch to another user, reboot/shutdown/sleep menu
- Analogy of the Windows login screen

Display Managers

- *LightDM*, *SDDM*, *SLiM*, *LXDM*, *MDM* (light display managers)
 - Not all support remote logins
- *KDM* (KDE Display Manager)
- *GDM* (Gnome Display Manager)



user



Window Manager (WM) is a system software that controls the placement and appearance of windows within a windowing system in a graphical user interface (GUI). It can be part of a *Desktop Environment* (DE) or be used standalone.

One can install several Desktop Environments in parallel and test them. KDE and Gnome are the heaviest and most feature-full DEs

- Gnome3 / Unity - MAC like look
 - Cinnamon - Gnome3 port with Windows-like look
 - KDE - Windows-like look
 - Mate - Gnome2 based
 - Xfce4, LXDE/Qt - light environment, but still feature-full
 - IceWM - extremely light Window Manager
-
- Native support for multiple desktops
 - Applications can use different libraries for graphics decoration, typically unified for a given DE. However, the applications written for specific DE can still be used in other DE or WM.

Copy & paste in/from a clipboard

- Highlighting automatically copies highlighted text into clipboard
- Paste action is triggered by mouse middle button
- Most GUI applications support in addition Windows-like Ctrl+c & Ctrl+v
 - Some applications (terminals, where Ctrl+c is reserved) support Ctrl+Shift+c & Ctrl+Shift+v
- Number of enhanced clipboard applications allowing to keep in memory multiple texts at one time

Drag & drop

- Drag and drop works for most graphical applications, but may not be available for the most simple *Window Managers*

Recycle bin

- The most complex *Desktop Environments* support *Recycle Bin*, but only for files deleted in their specific file-browser.
- Other file-browsers and especial deletion in terminal does not put files into the *Recycle Bin* → the files are instantly deleted.
- Moreover, the system may be configured from start not to use *Recycle Bin* at all

Quick list of Applications

Microsoft **Office** not available on Linux. However, there are number of alternatives:

- *LibreOffice* (former *OpenOffice*) fully included
- Gnome Office: *Abiword*, *Gnumeric*, *Gnucash*
- *Calligra Suite* (KDE focused office package)

PDF viewers:

- *Okular*: KDE PDF viewer, capable of adding annotations
- *Evince*: Gnome PDF viewer
- Old *Adobe reader* (needs special package repository)
- *Foxit reader*: not in package repositories, needs to be downloaded and installed manually

Latex packages naturally included in all distributions, with number of graphical editors allowing to debug Latex compilation and preview generated PDFs.

- *Texstudio*
- *Texmaker*
- *Kile* (KDE)
- *Visual Studio Code* + Latex plugin: free Microsoft product

The most **common browsers** are included:

- *Firefox*
- *Chrome* / Chromium
- *Opera*

There are also DE specific ones:

- *Konqueror* (KDE)
- *Epiphany* (Gnome)

As well as **terminal-based browsers** (limited use only, no javascripts etc.):

- *Lynx* / *Links* / *Elinks*
- *curl* / *wget*
- **Useful for scripts to extract information from web**

Number of **file browsers**, usually specific to DE:

- *Dolphin* (KDE)
- *Nautilus* (Gnome)
- *Thunar* (XFce4)
- *Krusader*: KDE-based browser with *Total Commander* like look and features
- *Midnight commander*: terminal based 2-panel file browser (*Norton/Total Commander* like); very useful when working on remote machines as it does not need to transfer graphics

Number of **terminal emulators**, usually specific to DE:

- *xterm*: basic terminal emulator
- *rxvt-unicode*: basic terminal emulator with unicode support
- *konsole*: KDE-based terminal emulator, support for tabs, complex configuration including predefined profiles etc.
- *gnome-terminal*: Gnome-based terminal emulator, support for tabs, complex configuration including predefined profiles etc.
- *xfce4-terminal*: Xfce4-based terminal emulator, support for tabs, complex configuration

Editors **for programmers** with various level of IDE:

- *Kate*: KDE based
- *KDevelop*: KDE based
- *Geany*: Gnome based
- *Eclipse*: heavy, Java based
- *Gedit*: light
- *Visual Studio Code*: Microsoft product, but open-source and free use (both home and commercial)

Console editors (useful when working on remote machines):

- *nano*: simple
- *jed*: simple
- *vim*: very powerful, but difficult for beginners
- *emacs*: very powerful, but difficult for beginners

Figures:

- *Gwenview* / *Gthumb* as picture viewers
- *Gimp* for bitmap drawings editing
- *Inkscape* for vector drawings editing

Video:

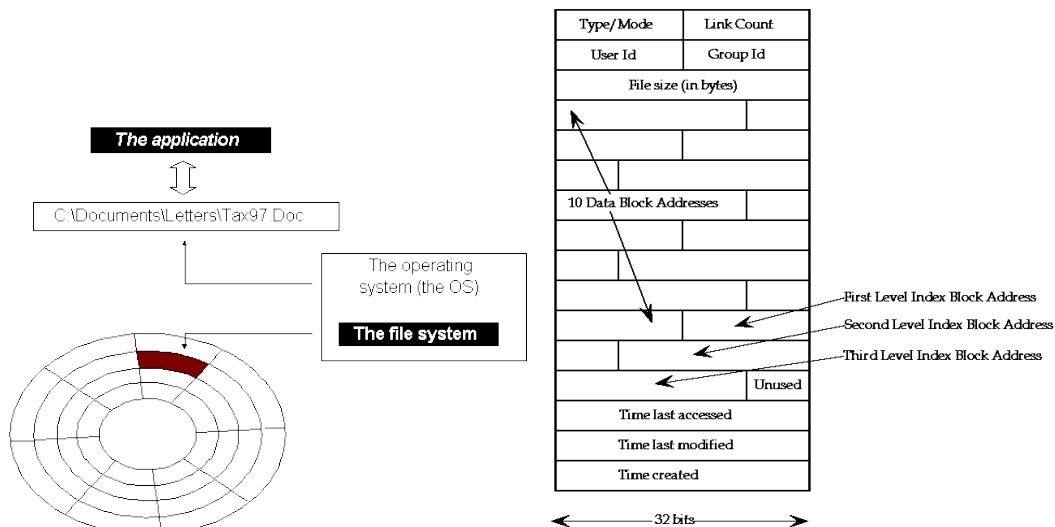
- *VLC*
- *SMplayer*
- *Xine*

Number of **command-line programs** for figures/audio/video conversions and processing

- *convert* for figures manipulations
- *ffmpeg* for audio/video manimulation

Filesystems

System defining the way files and directories are stored on the physical disk



Command to show this file information: **stat**