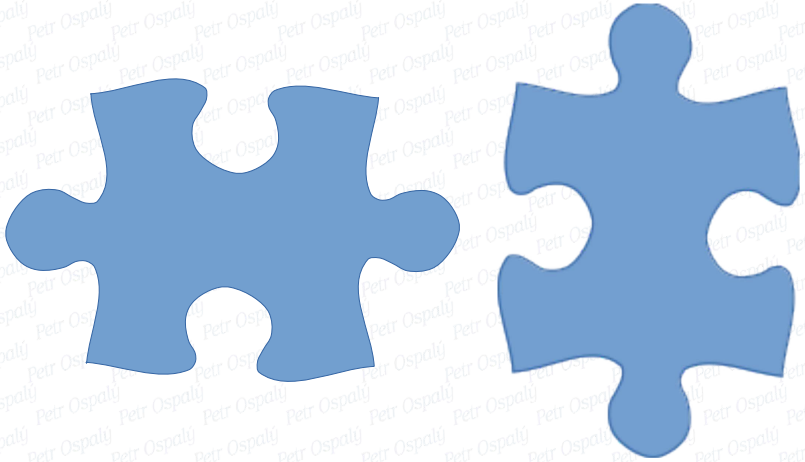


User Interfaces

Petr Ospalý

May 2025

User Interface

- A way to interact with the system, e.g.:
  - User to OS:
    - Inserting instructions via punch card
    - Typing on a terminal keyboard
    - Drawing on a touchscreen
  - OS to user:
    - Printing output on a paper
    - Displaying output on a screen

CLI – Command Line Interface

- Line-oriented
- Read-Evaluate-Print-Loop => REPL
- Typically running a command interpreter
- Effective and programmer friendly

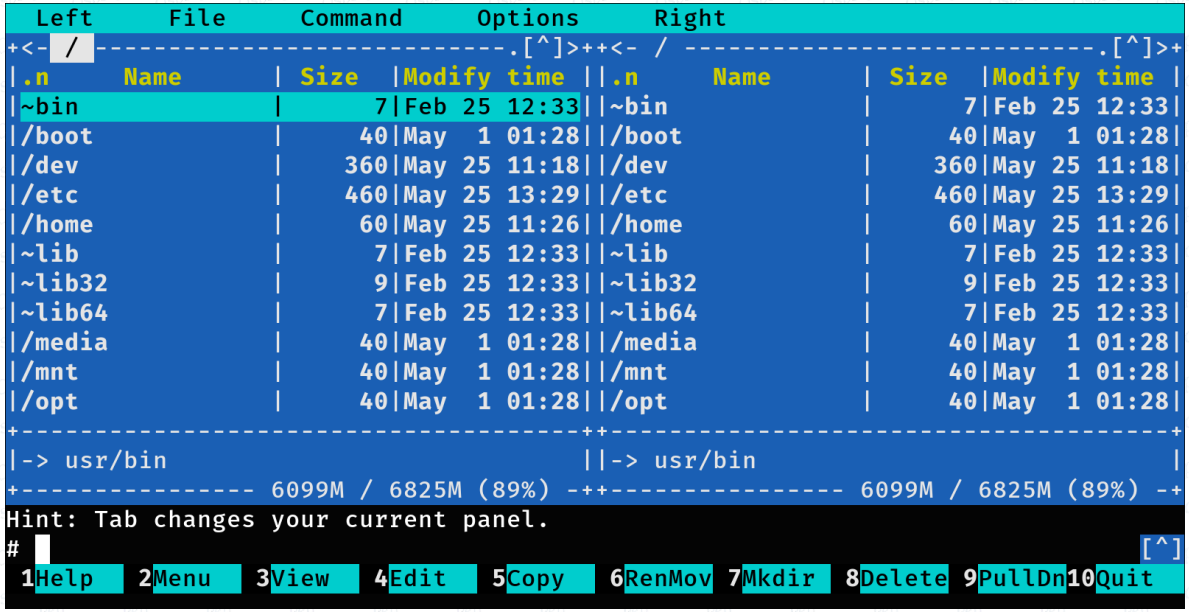
```
[friday@b9b278837505 ~]$ echo $SHELL
/bin/bash
[friday@b9b278837505 ~]$ printf "\n\nThis is 'Command Line Interface' running a 'bourne-again shell'\n\n\n"


This is 'Command Line Interface' running a 'bourne-again shell'


[friday@b9b278837505 ~]$
```

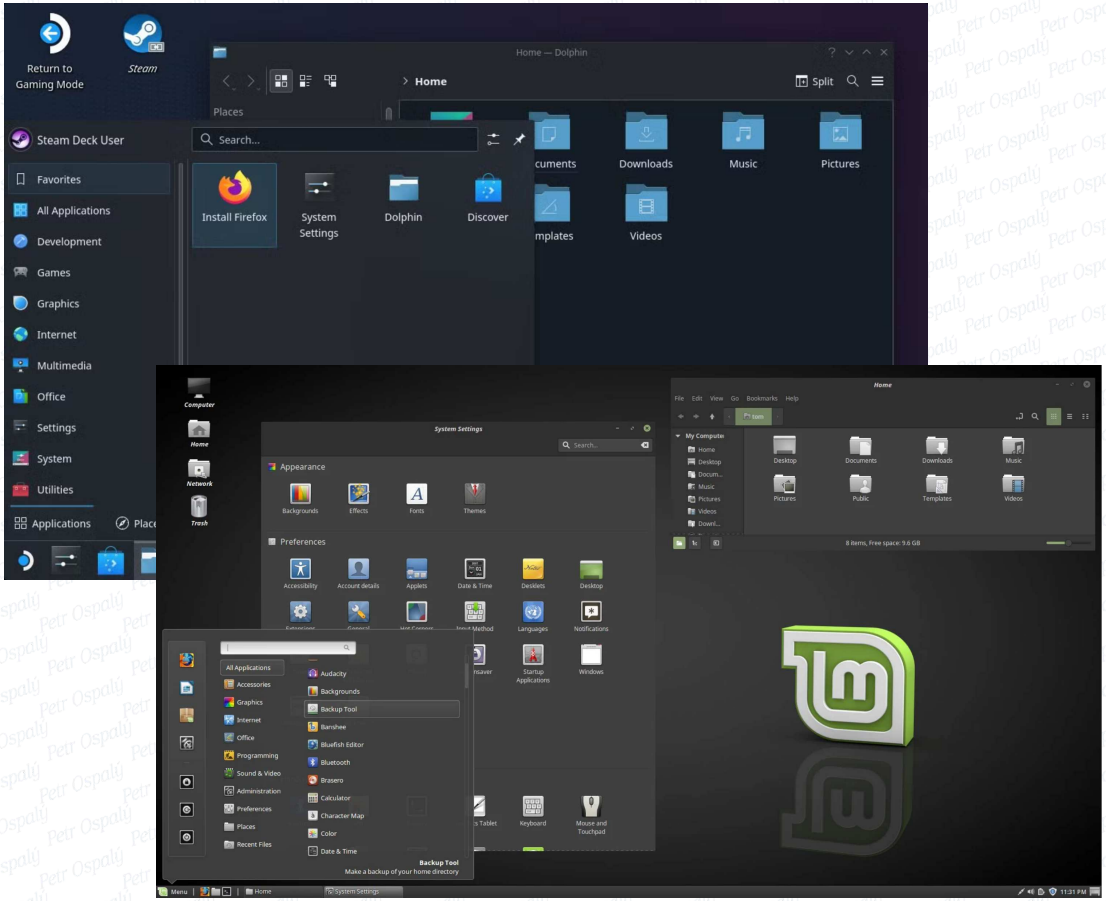
TUI – Text User Interface

- Precursor to GUI (next slide)
- Illusion of „widgets“ via „ascii art“
- Needs (n)curses library (or similar) to handle tricky terminal quirks



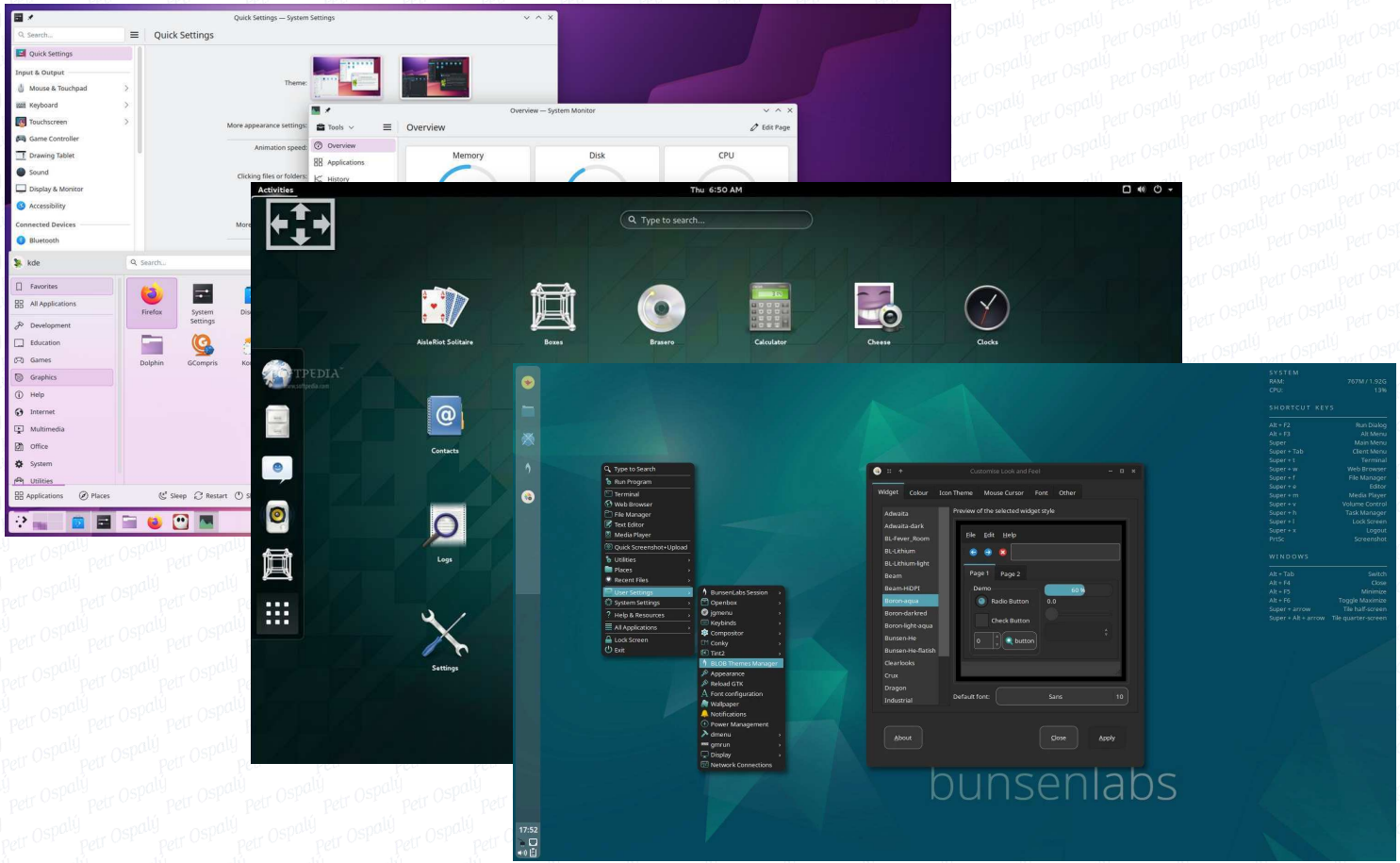
GUI – Graphical User Interface

- Utilizes vector or raster graphics (pixels on screen)
- Think of a desktop OS
- Linux/BSD has (too) many options
- Windows/Apple has one baked-in for each release



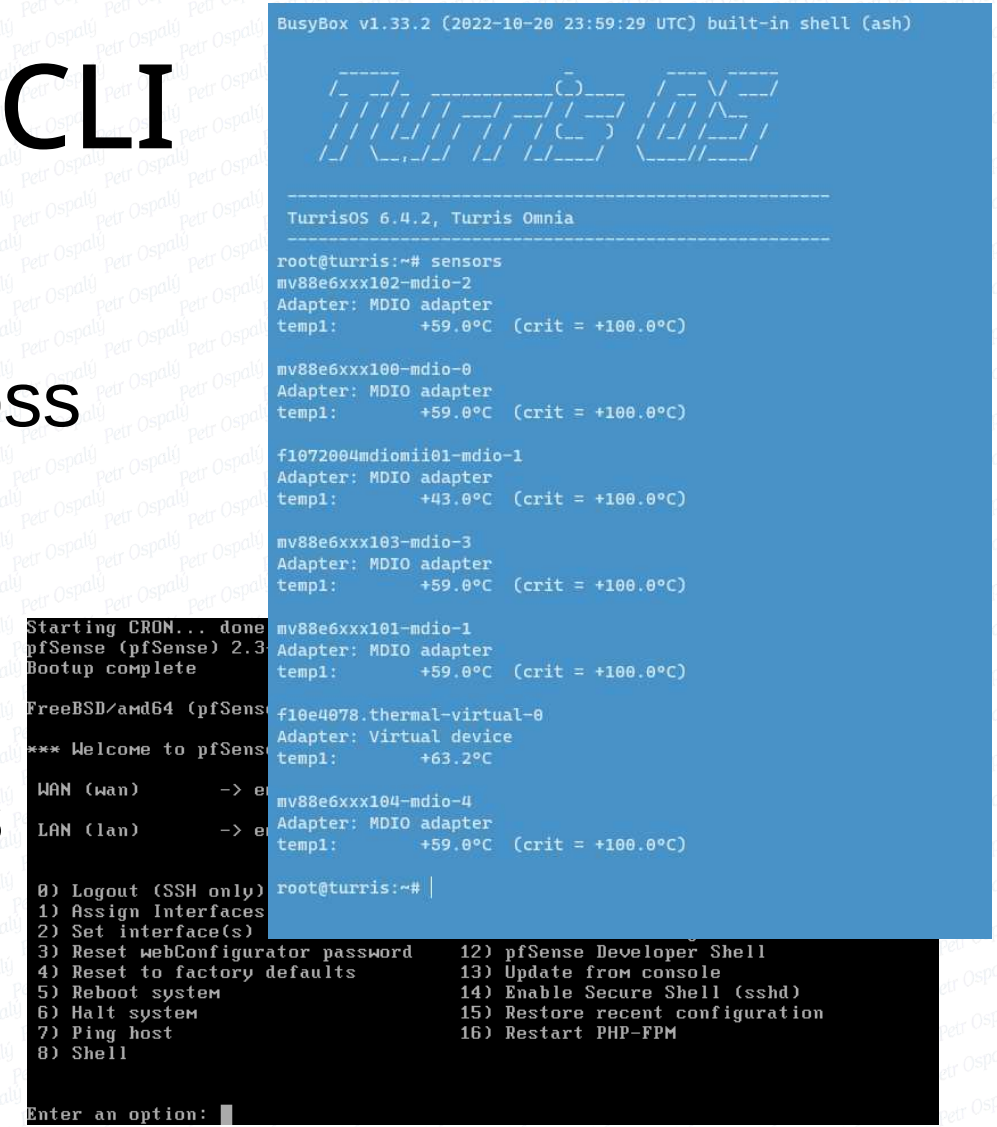
Linux Desktop Environments

- KDE
- MATE
- Gnome-shell
- Cinnamon
- XFCE
- LXDE
- WMs...



Reasons to use CLI

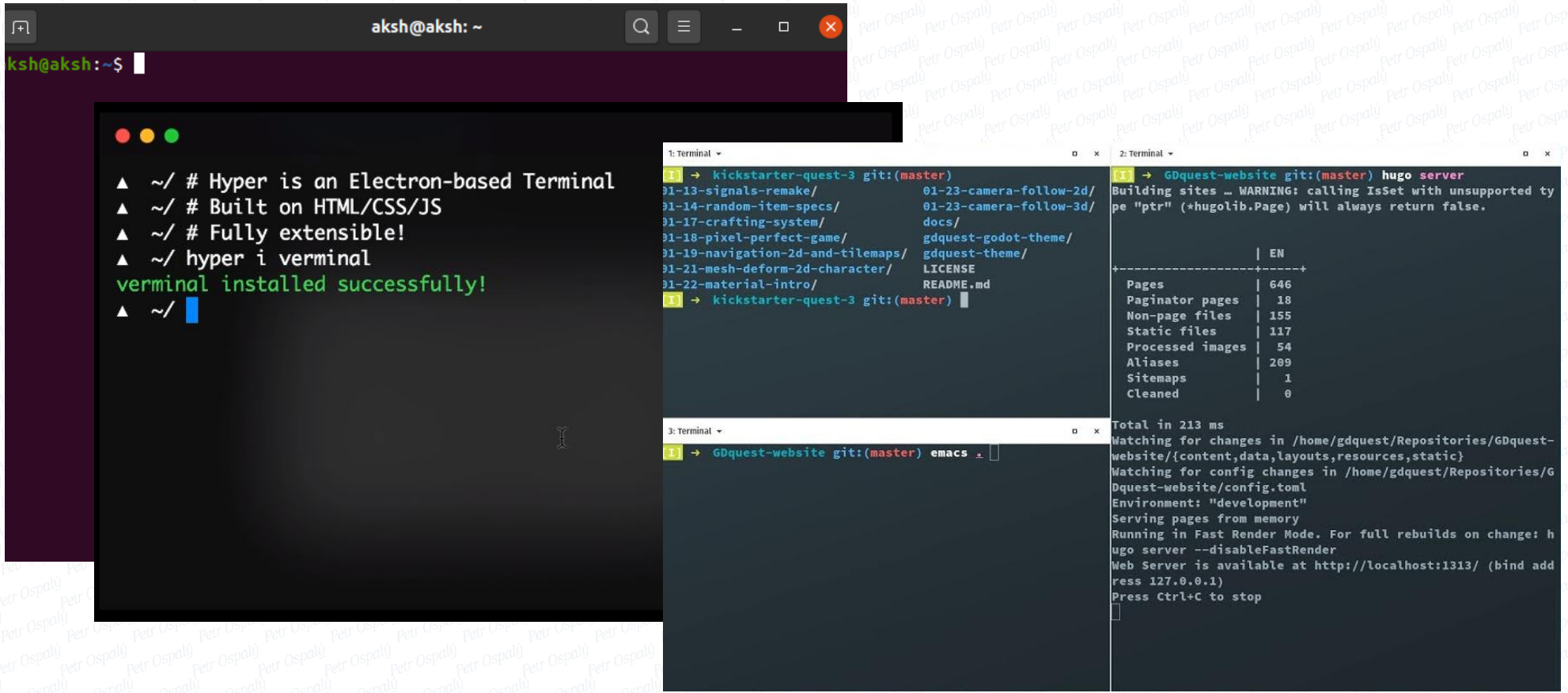
- Clean and minimalist
- Ubiquitous and same across environments
- Productive, programmer friendly, scriptable
- Low on resources – works over slow connections



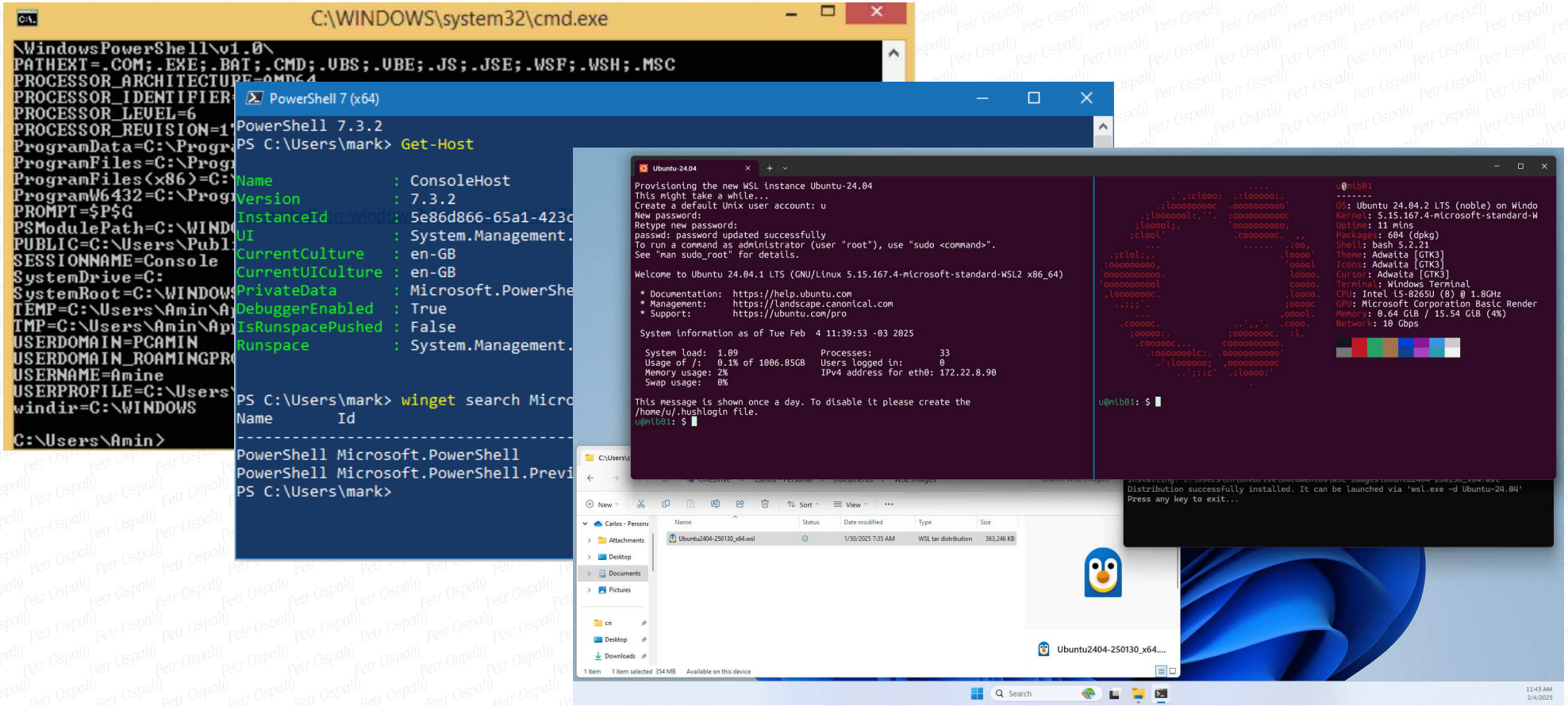
Terminal

- **Originally a simple physical device with no local processing**
- **Think of a “typewriter” that uses a video screen instead of a paper**
- There are no longer used in physical form but they survived emulated
- Running terminal window with a shell (command interpreter) is an idiomatic way to work on a *nix machine like Linux or Apple OSes
- Even Windows provide their old DOS era *cmd.exe* and modern *powershell* – these are not actually implementation of the original terminal devices but they are **still text-based command line interfaces**
- Recently Linux based *WSL terminal* is also available on Windows

Terminal Emulators

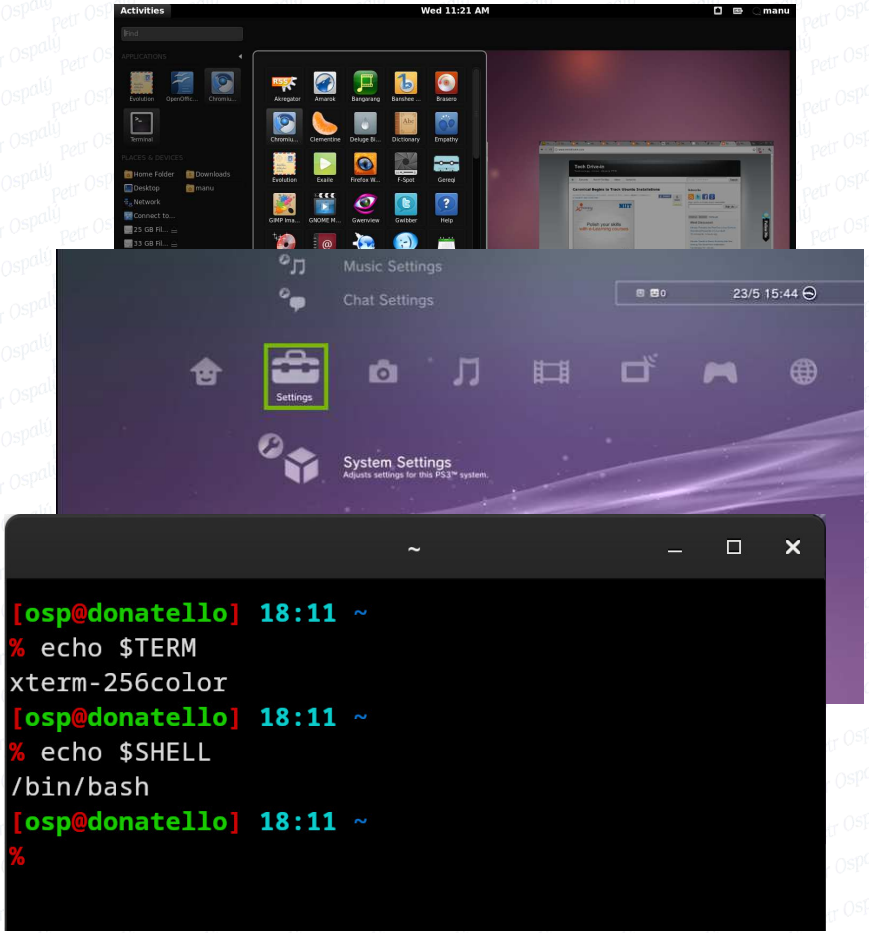


CLI/Terminals on Windows



Shell

- **Shell is the actual program interfacing with the user**
- Varies on context, e.g.:
  - Gnome 3 is a desktop “shell”
  - Console launcher is also a shell (PS, Xbox)
  - Terminals are running shell programs too – otherwise they would not be much useful
- Shell in our context is the command interpreter
- Many variants of shells…:
  - POSIX shell / bourne shell
  - BASH (Bourne Again Shell)
  - KSH, ASH, DASH, ZSH etc...

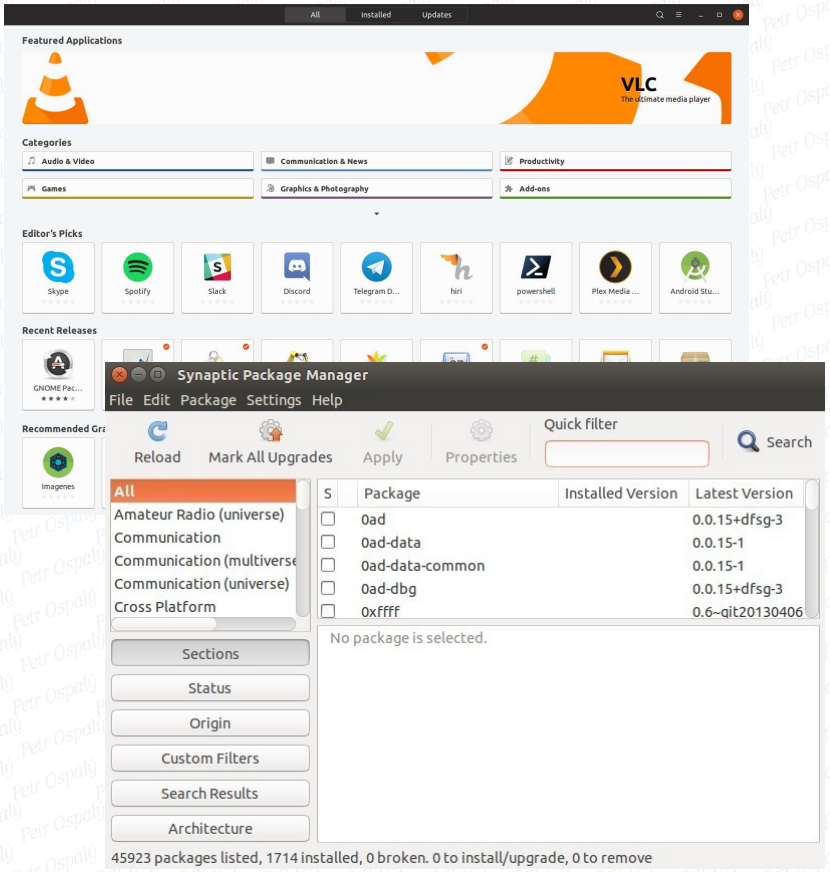


Comparison CLI vs GUI (1/4)

- Go-to tool to manage software on Linux is some package manager
- Let’s take example from Ubuntu
- You can either use command in the terminal as **aptitude** or use some graphical frontend (synaptic)
- The fact is that GUI application will never do more than the CLI tool will

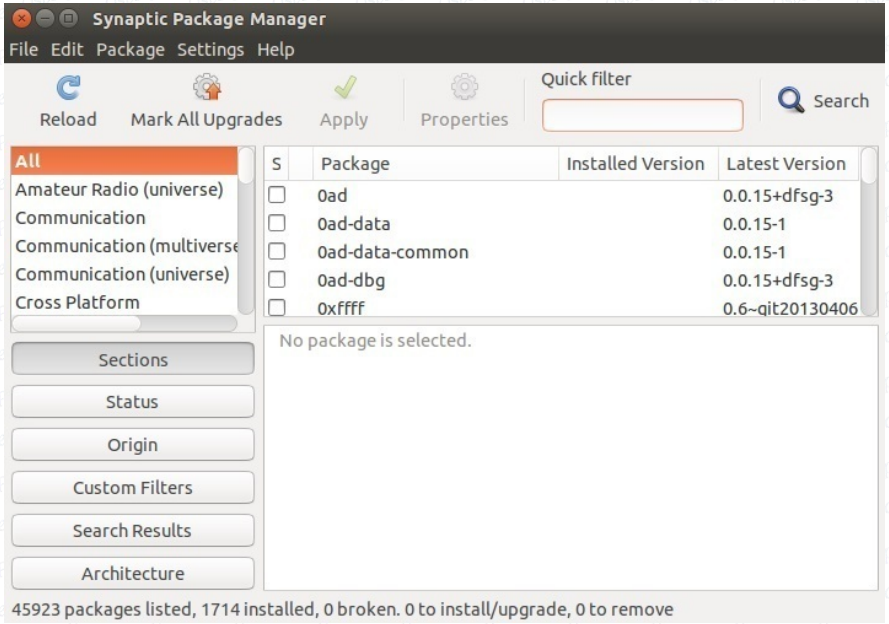
Comparison CLI vs GUI (2/4)

- Maybe we want to fully upgrade our system and install extra package (vlc)
- Graphical tool will require typing in search fields and clicking on buttons
- We even have more GUI frontends to do so



Comparison CLI vs GUI (3/4)

- In synaptic we have to:
  - Click on **reload** button
  - **Mark All Upgrades** button
  - **Apply** button
  - Typing “vlc“ in the quick filter
  - Ticking box in the main section for vlc package
  - **Apply** again



Comparison CLI vs GUI (4/4)

- In the terminal with the CLI:
  - **DEBIAN_FRONTEND=noninteractive **

    **apt-get update **

    **&& apt-get dist-upgrade **

    && apt-get install vlc
  - This is one long line which will not prompt the user for any input and will do all the tasks in one go
- Easily embedded in scripts

Terminal in practice

- Pictures of logos
- Joining into one
- Converting from PNG to JPG
- Result is below

```
~/tmp/pics
[osp@donatello] 16:20 ~/tmp/pics
% ls
arch.png    fedora.png  mint.png    slackware.png
debian.png  gentoo.png  redhat.png  ubuntu.png
[osp@donatello] 16:20 ~/tmp/pics
% magick *.png +append all.png
[osp@donatello] 16:20 ~/tmp/pics
% ls all.png
all.png
[osp@donatello] 16:20 ~/tmp/pics
% convert all.png -background white -alpha remove -alpha off all.jpg
[osp@donatello] 16:20 ~/tmp/pics
% ls all.jpg
all.jpg
[osp@donatello] 16:20 ~/tmp/pics
% file all.jpg
all.jpg: JPEG image data, JFIF standard 1.01, aspect ratio, density 0x0, segment leng
th 16, comment: "Created by DistroWatch.com", baseline, precision 8, 724x114, compone
nts 3
[osp@donatello] 16:20 ~/tmp/pics
%
```

Final words

- **Terminal and shell are often lumped together**
- **Linux distributions traditionally package all software in their repositories**
- GUI and CLI both has its places
- Graphical/Creative software is better served by GUI conveniences
- Outputs of many of these GUI applications are text files (e.g. XML)
- Many CLI tools exists to handle, parse and modify files (especially text) or they are backends to their GUI extensions
- Modern systems usually talk to each other via text interfaces (e.g. REST API + JSON)
- Many services, clients, servers and tools have no graphical representation or the CLI version is more convenient and/or more powerful