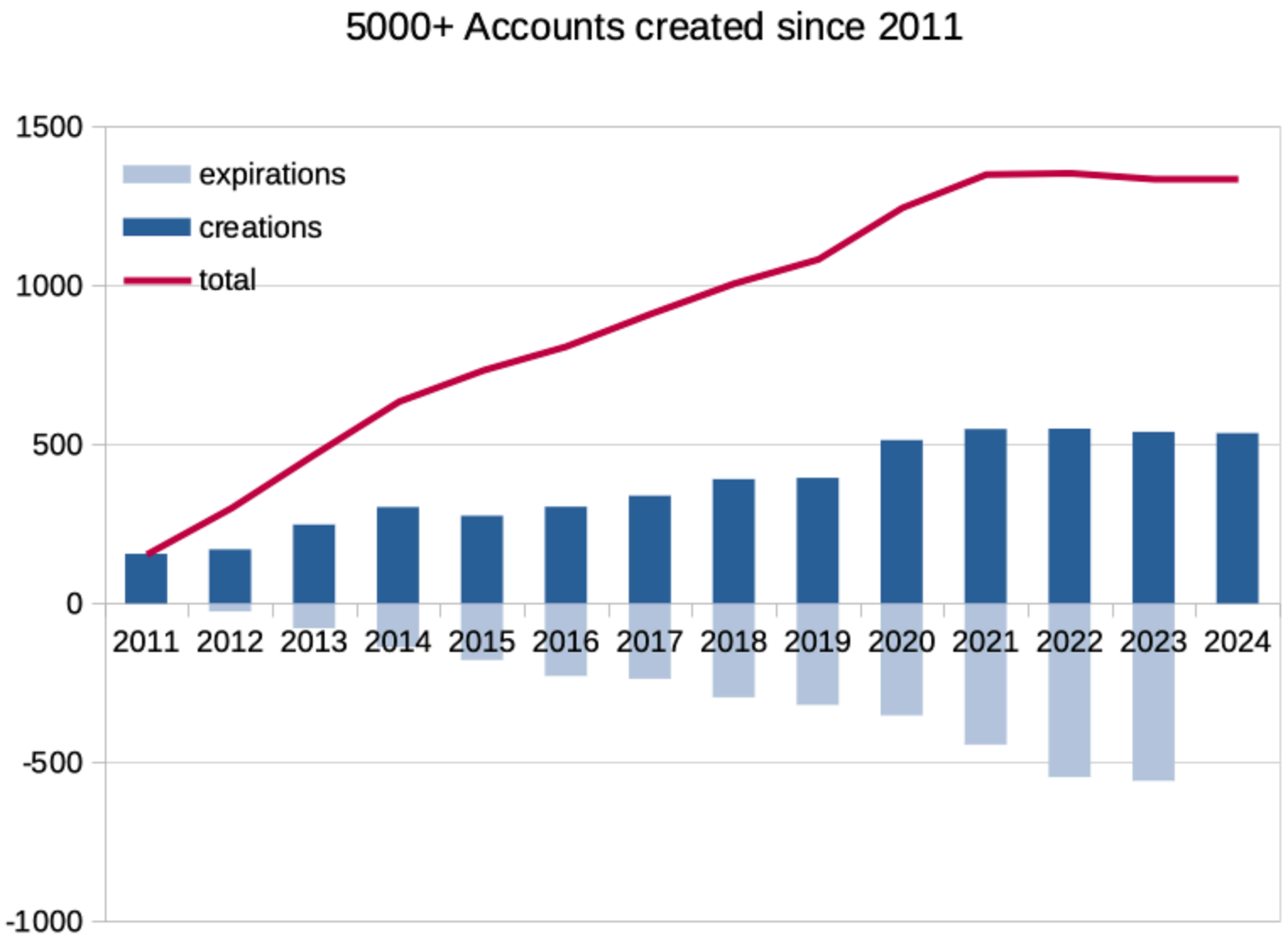
Help us help you

damien.francois@uclouvain.be

October 2024

You, the CÉCI users, are legion



We, the CÉCI administrators, are a small team

- UCLouvain: CISM (4.4)
- ULiège: nicadm (2.5)
- UNamur: PTCI (2)
- UMons: CMN (1)
- ULB: HPC team (2)

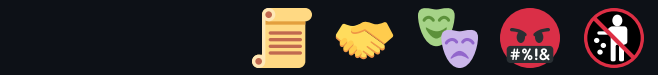
🎯 Our goals :

- 🚦 provide as much uptime as possible for the CÉCI infrastructure (a.o.)
- ⚖️ accommodate a wide spectrum of jobs
- 📶 maximize resource (CPUs, GPUs, memory) utilization
- 🕐 minimize turnaround for your jobs

How can you help us?

1. play along with others
2. prepare your interactions with us
3. be demanding with yourself

play along with others



📜 Know the rules

- the rules of your university
- the rules of the university hosting the cluster you are using
- the rules of CÉCI: https://www.ceci-hpc.be/faq.html#2.5
- do not harm security
    - **do not share secrets**
    - **do not give access to the clusters to others**

📜 The rules exist so that everyone's data are safe and secure

🤝 Be cooperative

- agree to be registered to mailing lists for announcements related to the infrastructure or events we organise (a dozen email per year) and **read the emails**
- participate in the ceci **user days**, take the **surveys**
- **acknowledge** the CÉCI and other providers in your publications that use their infrastructure

🤝 All this is important when requesting funding for the clusters

🎭 Do not game the system

- when we enforce something on users it is always to ensure fairness among users
- more constrains hinders performances and harms complex workflows
- do not stalk other users their science is as important as yours

🎭 Loss of trust ultimately results in loss of resources

🤜🤛 Be thoughtful of others

🤬 avoid problem-generating workflows:

- Running anything CPU-intensive on the head node
- Issuing many requests per unit of time to the scheduler
- Submitting large amounts of jobs without testing
- Performing excessive I/O on a global filesystem rather than on a local filesystem
- Storing large number of small files rather than consolidating them
- Leaving typos in the email options
- Leaving data on the scratch for ever

🤜🤛 Be thoughtful of others

🚳 do not waste resources, e.g. with

- jobs whose output is discarded because of misconfiguration
- jobs whose output is unsaved due to file manipulation error
- jobs under-using the resources requested because of misconfiguration
- jobs under-using resources due to bad scaling
- jobs sitting idle waiting for interactive commands
- jobs whose results are lost because of hardware failure

🤜🤛 Be thoughtful of others

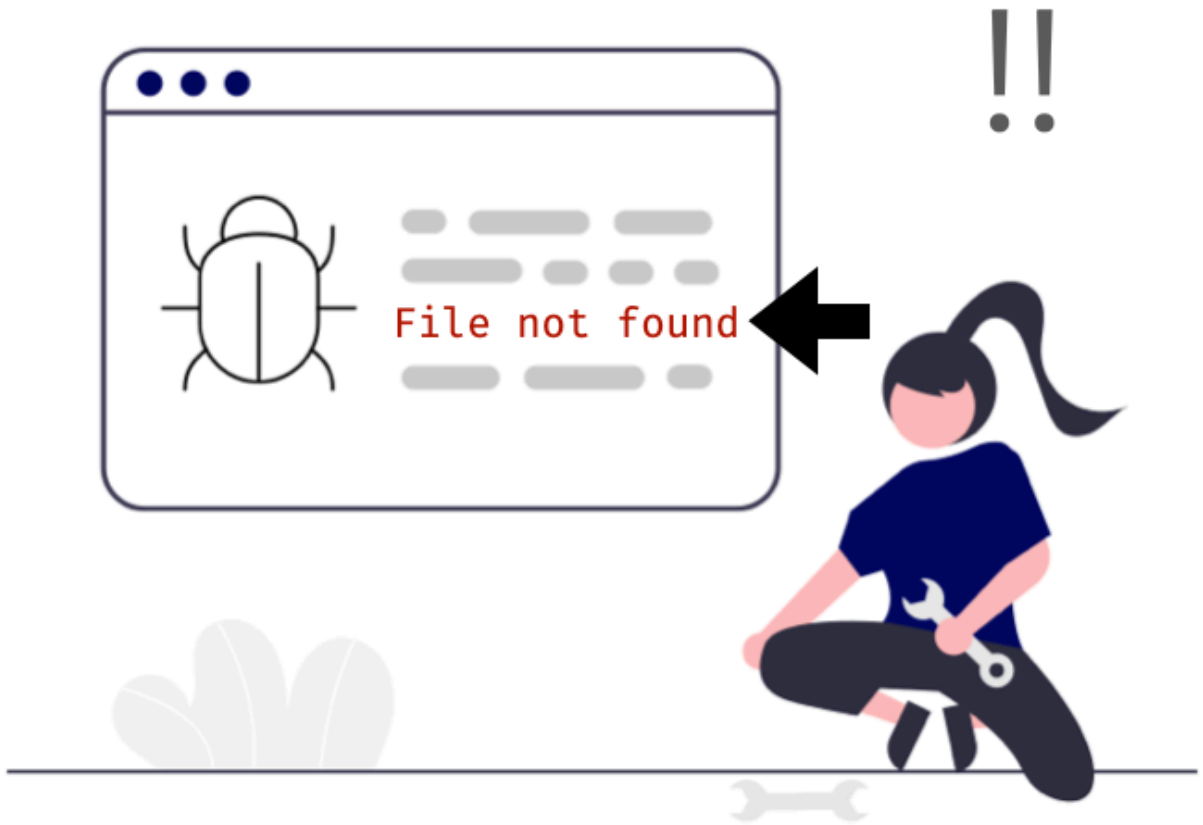
- avoid problem-generating workflows
- do not waste resources

🤜🤛 Cooperation is what keeps the cluster running

prepare your interactions with us

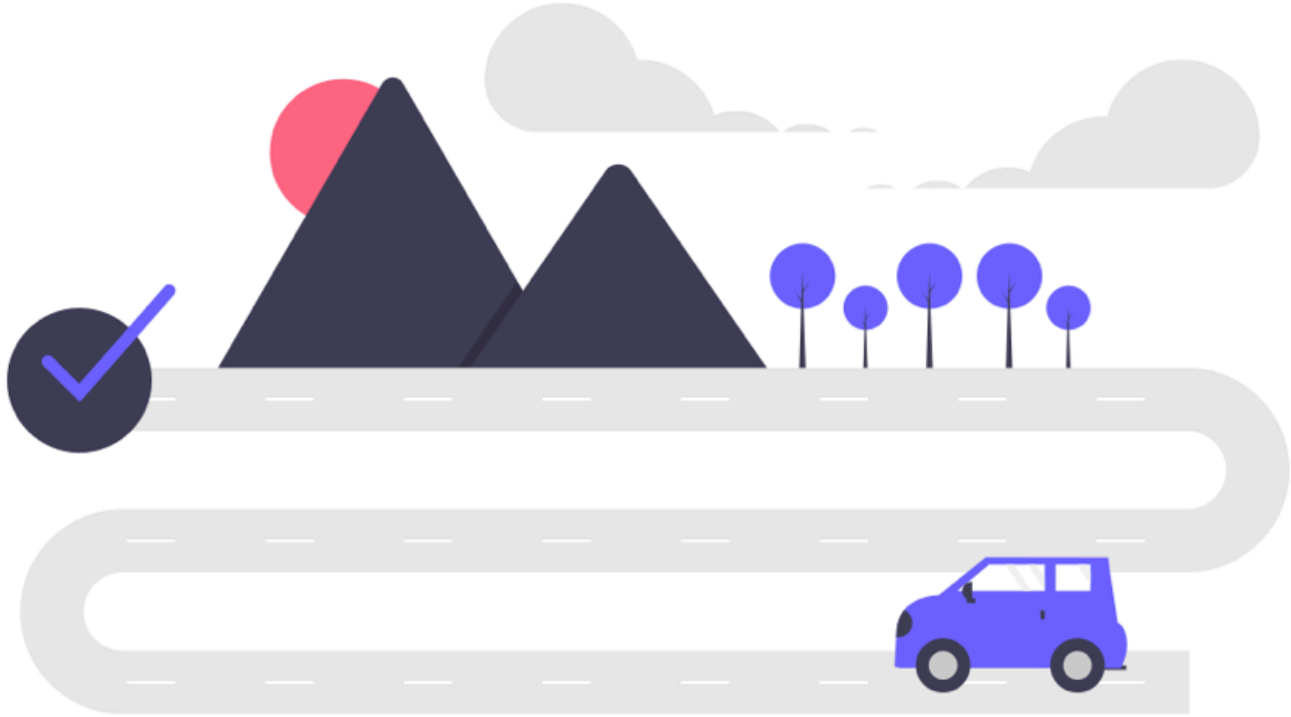
🤔 📡 🎉 ℹ️ 💬

🤔 1. Start with your due diligence.



2. Use the correct channel

🎉 3. State the general goal

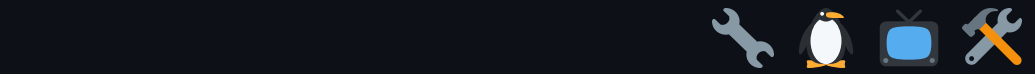


ℹ️ 4. Provide all important information ("five W")

- **Who**: what is your login?
- **What**: what job ID, what file, what modules?
- **When**: on which date, at what time?
- **Where**: which cluster, which directory?
- **Why**: what is the problem?

💬 5. Give the exact error (copy/paste from terminal)

be demanding with yourself



🔧 There are certain tasks you can do by yourself

- install software (modules included)
- change permissions back after an error, sharing files among users
- remove a failing `.bashrc`
- change group ownership
- join a Tier-1 project
- fix quota exceedance

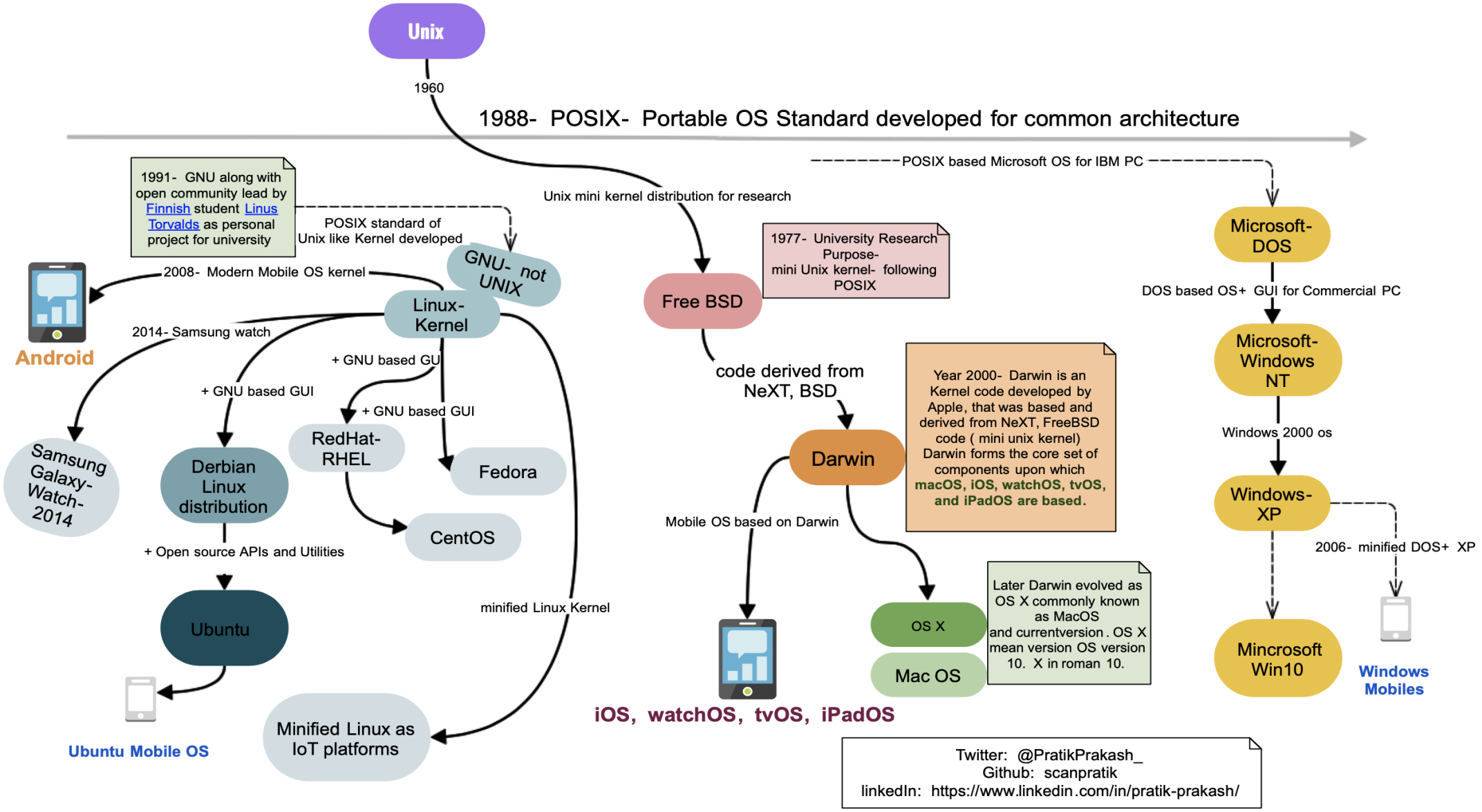
Some you MUST do

- take responsibility for backups of your files 💾

Learn the basics before you use the clusters

- 🐧 Basics of **Operating systems**
- 📺 Basics of **Text user interfaces and the Terminal**
- 🛠️ Basics of **Programming**

Evolution of Modern Operating Platforms



If Linux is entirely new to you, please read

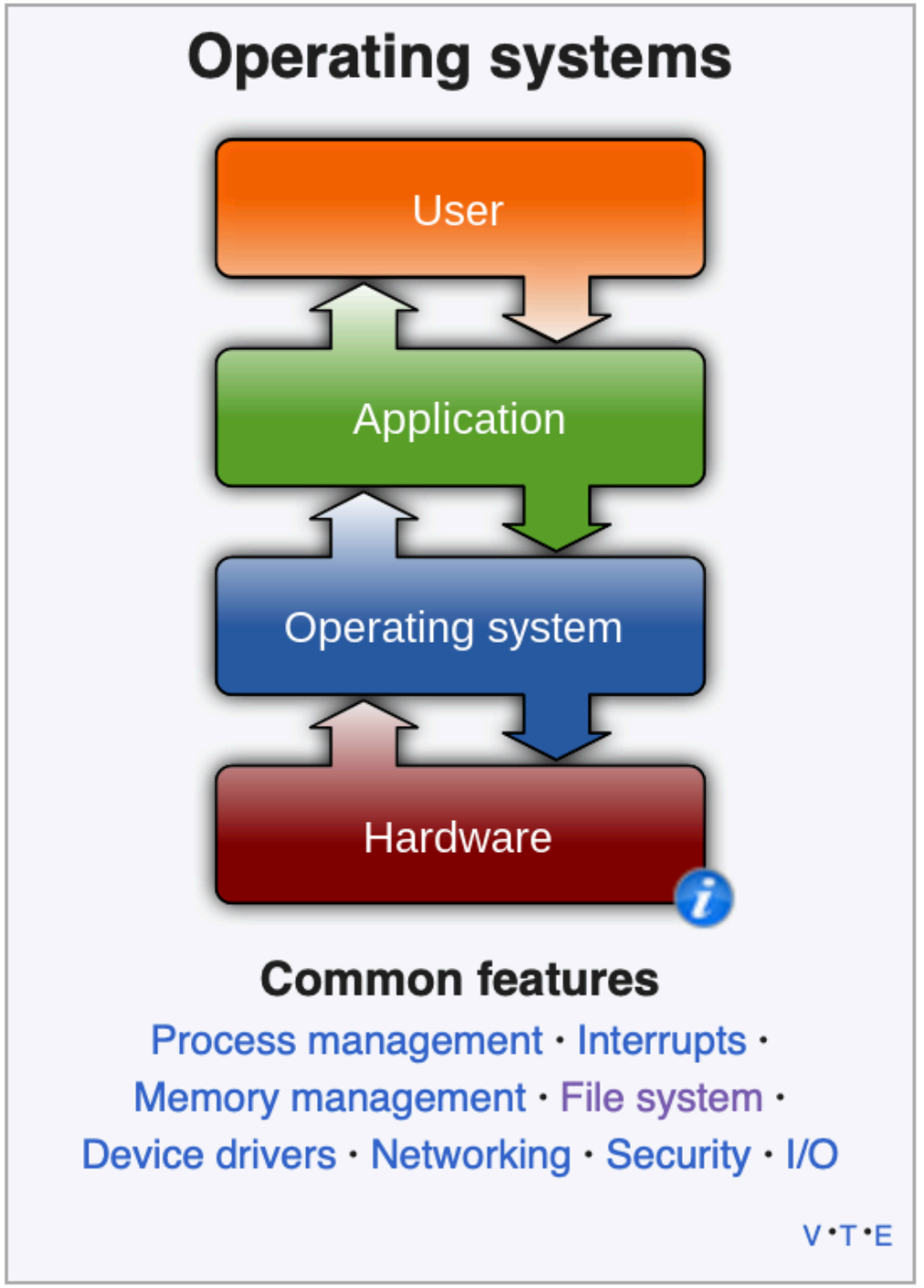
https://www.linux.com/what-is-linux/

- 5-minutes read
- addresses the basics
- offers links to further documentation

And attend the next sessions.

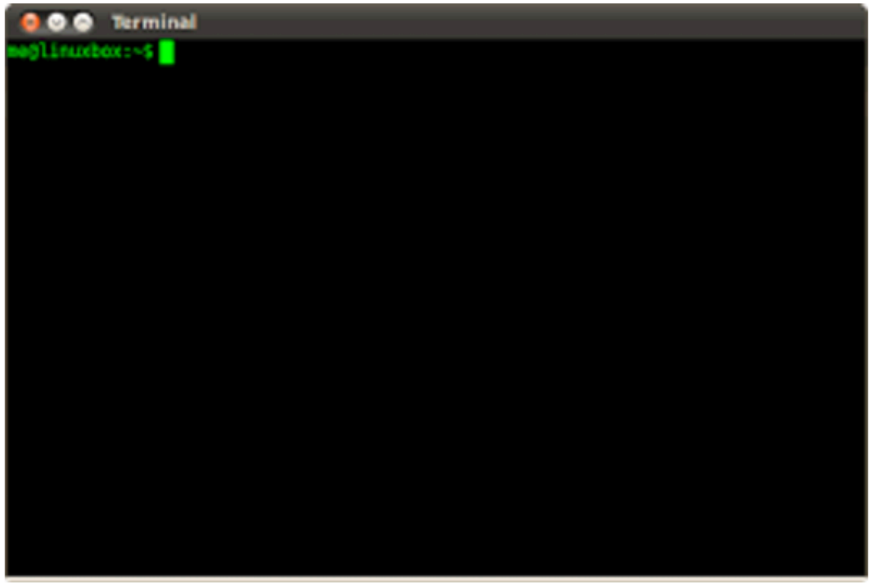
Operating system, *OS*

- organises access to resources inside a computer
- allows starting programs, accessing files, etc.
- interacts with the user through:
  - the *desktop* (GUI)
  - the *terminal* (CLI)



https://en.wikipedia.org/wiki/Operating_system

The *desktop* vs the *terminal*



Interacting with the *shell* in the *terminal*:

- we type "simplified" sentences after the *prompt* in the *terminal*
- we press the `Enter` key when we have finished our sentence
- the *shell* interprets the sentence and act accordingly
- we wait for the response, which is then displayed in the *terminal*
- and we start again with another sentence ⤴️

On the compute servers, the shell is called `bash`

The Bash *shell* gives access to

- a set of basic shell commands
- a set of command-line tools (`GNU`)
- a rudimentary programming syntax

If the terminal is entirely new to you, please read

https://frontend.turing.edu/lessons/module-1/getting-around-in-the-terminal.html

and

https://www.makeuseof.com/tag/a-beginners-guide-to-the-windows-command-line/

- 15-minutes read in total
- addresses the basics
- offers links to further documentation

And then attend the next sessions.

About programming

Concept

> Writing instructions for the computer in a human-comprehensible language that can be univoquely translated into a computer-understandable language.

"write 'hello' on the screen" -> `echo hello` -> `10100101011101011010010 11...`

About programming

Basic constructs

- Variables: type, scope, assignment, and operators
- Conditionals: if-then, if-then-else
- Loops: for, while, etc.
- Functions: call, parameter, return value, side effects

If programming is new to you, please read

https://www.geeksforgeeks.org/basics-of-computer-programming-for-beginners/

- 10-minutes read
- addresses the basics
- offers links to further documentation

before you attend the next sessions.

Help us help you

play along others for maximum science per Watt

prepare your interactions with us for most efficient support

be demanding with yourself for getting the most of the resources

😄 We thank you in advance! 😄