

OPEN SOURCE IN 12 MINUTES

(hopefully)

MAIN CONCEPTS

Cyber Resilience Act (CRA)

Proposed EU Regulation
2022/0272/COD

«In order not to hamper innovation or research, **free and open-source software** (...) should not be covered by this Regulation».

«Free and open-source software» (FOSS)

- Software provided for free?
- And has its source code available?

not quite!

Main concepts:

- Source Code
- Free Software
- Open Source

FOSS EXAMPLES



Thunderbird



zotero



HISTORY





Richard Stallman

Programmer and hacker from Harvard and MIT in the 70's

In 1980, his lab at MIT got a new printer. It was not working as they intended, so Stallman wanted to fix it. He knew what the problem was and how to fix it.

Problem: **no access to the source code.**
Copyright...



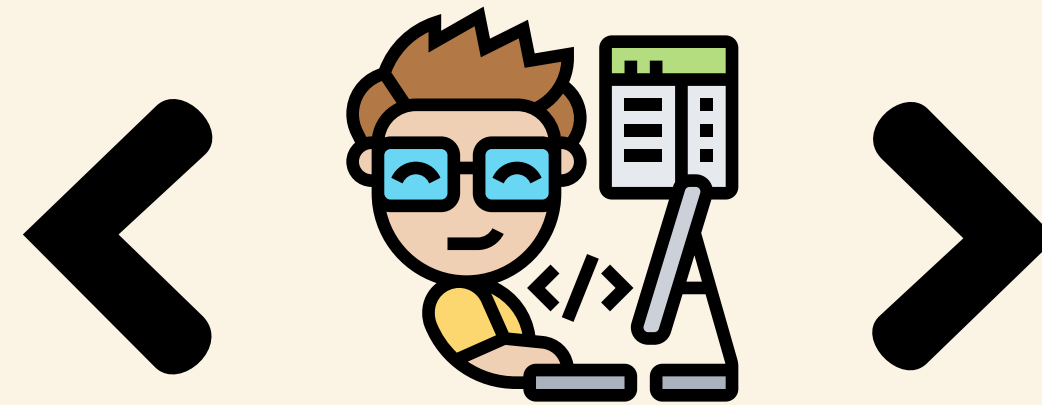
THE SOURCE CODE

What computers
understand:

01001001
01010000
01010011
01001001

Binary system

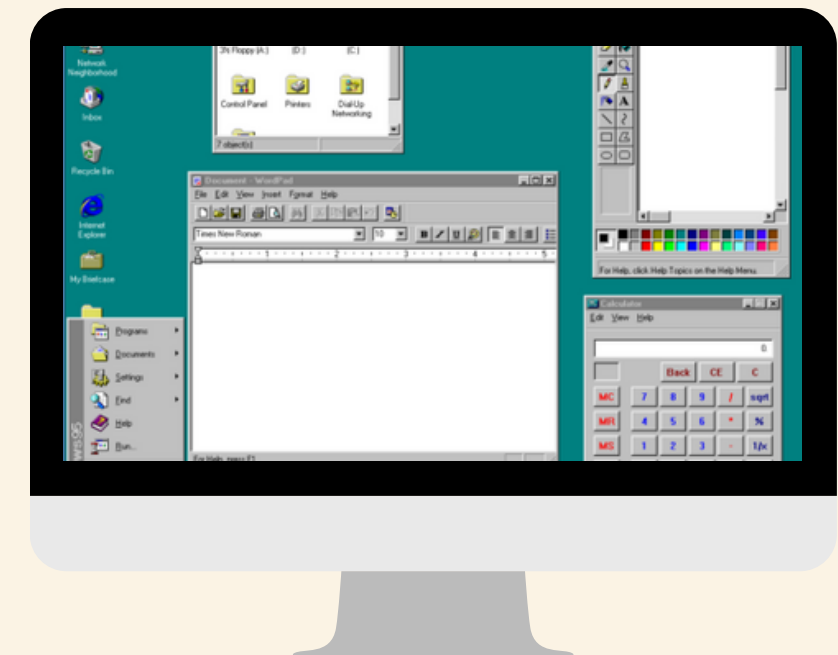
Programmers:



Programming languages

Python, C, C++, Java, Go,
R, PHP, Perl, Bash, Ruby, HTML*, etc.

What humans
understand:



User interfaces



THE SOURCE CODE

Code:

```
if ($(window).scrollTop() > header1_initialDistance) {  
  if (parseInt(header1.css('padding-top'), 10) >= header1_initialDistance) {  
    header1.css('padding-top', '' + $(window).scrollTop() - header1_initialDistance);  
  } else {  
    header1.css('padding-top', '' + header1_initialDistance);  
  }  
  if ($(window).scrollTop() > header2_initialDistance) {  
    if (parseInt(header2.css('padding-top'), 10) >= header2_initialDistance) {  
      header2.css('padding-top', '' + $(window).scrollTop() - header2_initialDistance);  
    } else {  
      header2.css('padding-top', '' + header2_initialDistance);  
    }  
  }  
}
```

Copyright exception:
Art 6 Directive
2009/24/EC

decompiling or
reverse engineering
(not that simple)

compiling

Executable file:



zoom-cloud-meetings-5-17-7-31859.exe
(binary)

When installing .exes:

- Black box: you can't access the code
- You don't know what you are actually installing
- You **trust** it is what they say it is

1/2



THE SOURCE CODE

Code:

```
if ($(window).scrollTop() > header1_initialDistance) {  
  if (parseInt(header1.css('padding-top'), 10) >= header1_initialDistance) {  
    header1.css('padding-top', '' + $(window).scrollTop() - header1_initialDistance);  
  } else {  
    header1.css('padding-top', '' + header1_initialDistance);  
  }  
  if ($(window).scrollTop() > header2_initialDistance) {  
    if (parseInt(header2.css('padding-top'), 10) >= header2_initialDistance) {  
      header2.css('padding-top', '' + $(window).scrollTop() - header2_initialDistance);  
    } else {  
      header2.css('padding-top', '' + header2_initialDistance);  
    }  
  }  
}
```

compiling

Executable file:



zoom-cloud-meetings-5-17-7-31859.exe
(binary)

decompiling or
reverse engineering
(not that simple)

Copyright exception:
Art 6 Directive
2009/24/EC

Source code required to:

- Know what the program actually does
- Modify the program

Having the code, anyone can compile it.

- In most software, the code is not provided to the user

2/2



HISTORY

Computers in the 70's:

Microship revolution, computers getting smaller and affordable to people, not only companies and institutions. First personal computers appear.

- Pre-Internet
- An expensive toy for hobbyists
- Software/code was created and shared among people ("IP negative space"); © required registration.

A shift: trend to professionalization

Copyright Act of 1976 (effective in 1978): protection since the creation.

Bill Gates: "An Open Letter to Hobbyists" (1976)

Copyright enforcement. Restrictive licenses. No more sharing of source code. Enclosure. Confidentiality, NDAs, trade secrets.

From freedom to restrictions (the copyright default).





The ideal software: Free Software

« Free as in 'free speech', not 'free beer' »

4 essentials freedoms:

- The freedom to **run the program** as you wish, for any purpose (freedom 0).
- The freedom to **study how the program works, and change it** so it does your computing as you wish (freedom 1).
- The freedom to **redistribute copies** so you can help others (freedom 2).
- The freedom to **distribute copies of your modified versions to others** (freedom 3).



A dream: a fully free operative system

- GNU Project (1983/84)
- GNU Manifesto (1985)
- Free Software Foundation (1985)



Objective: GNU operative system

But how?

Using copyright licenses!

GNU General
Public License
(1989)



GNU General Public License (1989)



Free Software

Granting users the freedoms to:

- use
- study
- share
- modify



Copyleft

(optional)

One restriction:


- if you modify this code and redistribute, you must make the new / changed work available under the same or equivalent license.
- Two categories: Permissive vs Restrictive (copyleft) free software licenses.





Linus Torvalds

21 year old student from Helsinki, Finland (1991)

 **Linus Benedict Torvalds**

★

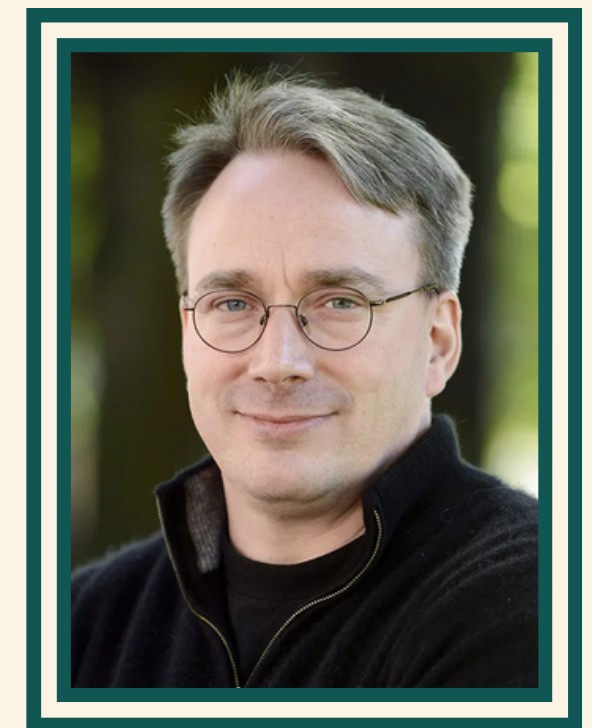
Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

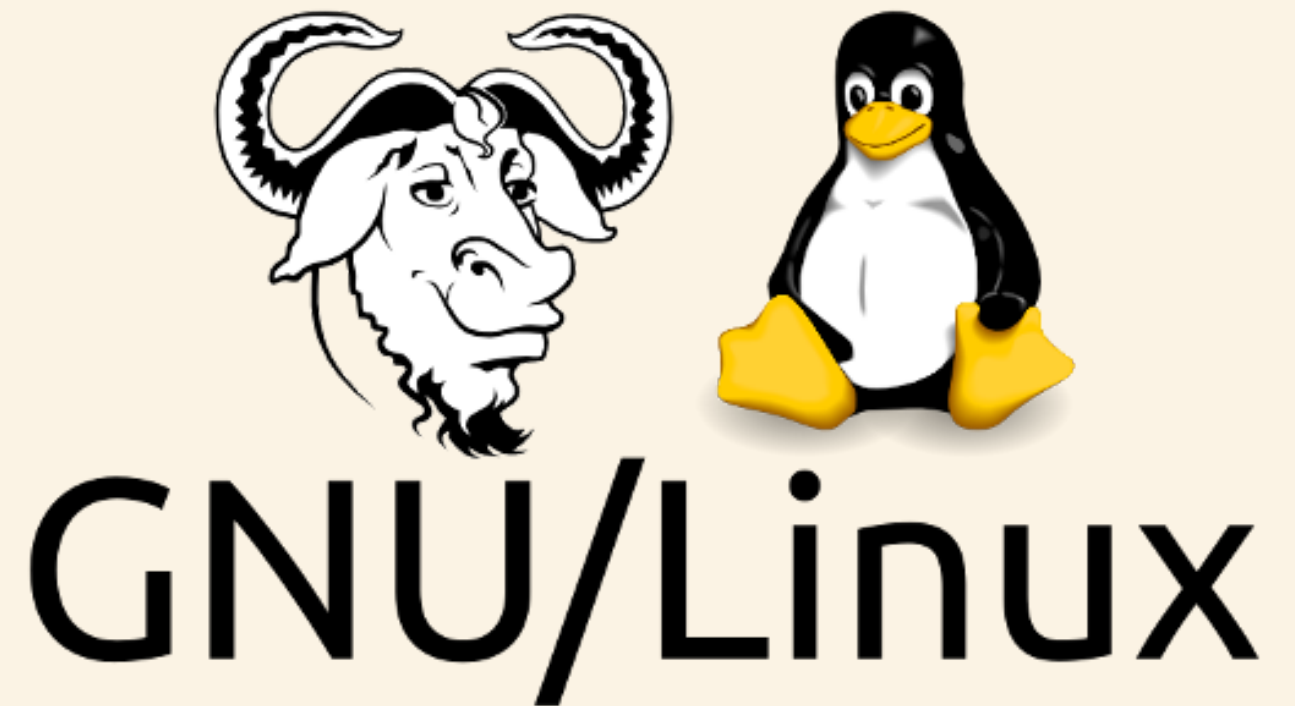
Linus (torv...@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).



HISTORY - GNU/LINUX

«Oh, look! A kernel!»



- Lots of enthusiasts join in and help to improve it, a community is created.
- Huge success: the development rhythm is unmatched.
- From the niches to mainstream.



"The Cathedral and the Bazaar" (1997)

Eric S. Raymond

Analysing two different development models: Cathedral vs Bazaar. Counter intuitively, the bazaar works and it can be extremely competitive.

1998: Netscape releases its source code (Mozilla)

"Free software" does not work as a name.
Let's call it **open source**.

« Can we make this model more business-friendly? »



open source
initiative[®]
(1998)



FREE SOFTWARE VS OPEN SOURCE

"Libre Software"

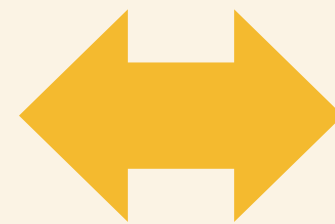
FREE SOFTWARE

Ethical
Political
Focus on users' freedom



OPEN SOURCE

Business oriented
Politically neutral
Practical
Economic and competitive
benefits of the open model



Rivalry



open source
initiative®



FREE SOFTWARE VS OPEN SOURCE

“Libre Software”

FREE SOFTWARE

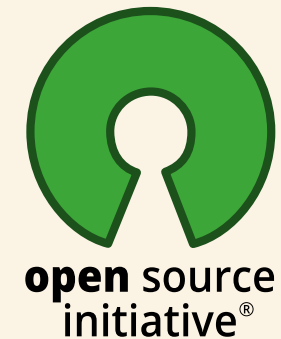


Four freedoms to:

- use
- study
- share
- modify

OPEN SOURCE

1. Free Redistribution
2. Source Code
3. Derived Works
4. Integrity of The Author's Source Code
5. No Discrimination Against Persons or Groups
6. No Discrimination Against Fields of Endeavor
7. Distribution of License
8. License Must Not Be Specific to a Product
9. License Must Not Restrict Other Software
10. License Must Be Technology-Neutral



FREE SOFTWARE VS OPEN SOURCE

“Libre Software”

FREE SOFTWARE

- GNU General Public License (GPL)
- Berkeley Software Distribution (BSD)
- MIT License
- European Union Public License (EUPL)
- Mozilla Public License

...



OPEN SOURCE

- GNU General Public License (GPL)
- Berkeley Software Distribution (BSD)
- MIT License
- European Union Public License (EUPL)
- Mozilla Public License

...



open source
initiative®

←→
Same licenses!



No legal definition of FOSS, but...

European Union Public Licence (EUPL)



- Published in 2007
- Written and approved by the European Commission
- For software made available by the EU, but anyone can reuse the license
- Approved by FSF and OSI
- Copyleft



Cyber Resilience Act (CRA)

Proposed EU Regulation
2022/0272/COD

«In order not to hamper innovation or research, free and open-source software (...) should not be covered by this Regulation. This is in particular the case for software, including its **source code** and modified versions, that is **openly shared and freely accessible, usable, modifiable and redistributable**.



Cyber Resilience Act (CRA)

Proposed EU Regulation
2022/0272/COD

In order not to hamper innovation or research, free and open-source software **developed or supplied outside the course of a commercial activity** should not be covered by this Regulation. This is in particular the case for software, including its source code and modified versions, that is openly shared and freely accessible, usable, modifiable and redistributable.

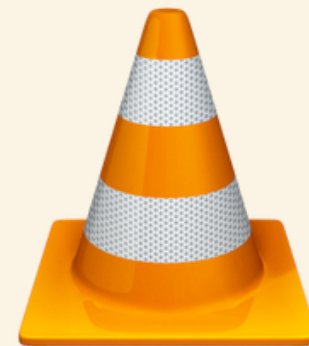
Recital 10 - **“definition” & exclusion... in recitals.**



FOSS EXAMPLES



Thunderbird



VLC

zotero



«Free and Open Source Software» (FOSS)

- Not a type of software
- It's **not** about the programming languages used, nor the purpose of software
- It's **not** about the price. Most business models provide the software gratis, but it's not mandatory
- **It's about the copyright license!**
- Source code must be available + licensed under a FOSS license



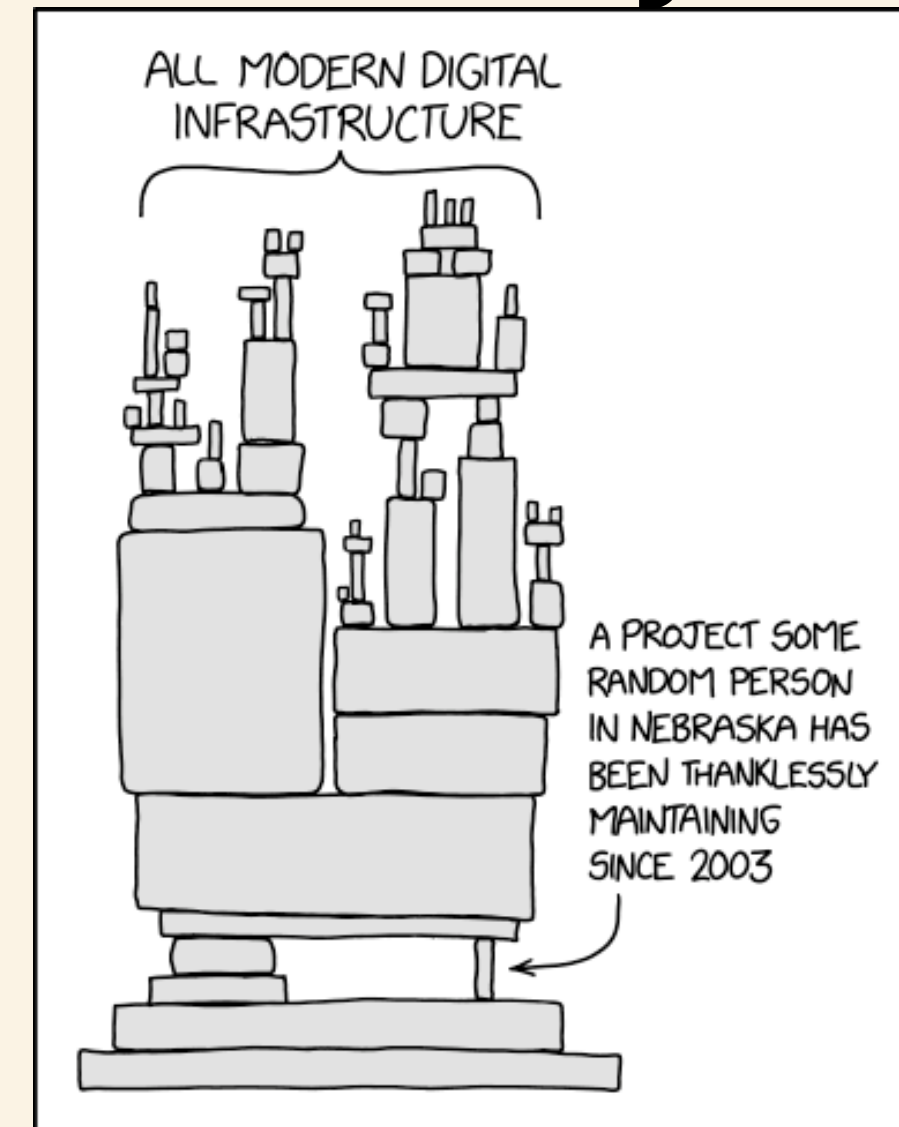
FOSS & sustainability

- inclusive
- better flexibility
- avoids vendor lock-in
- more secure
- competitive
- transparent

9 INDUSTRY, INNOVATION
AND INFRASTRUCTURE



Sustainability of FOSS



xkcd.com/2347

