Acronym
ooo

History
ooooo

Licences
ooo

Advantages
ooooo

Challenges
oooo

Examples
ooo

Users/Developers
ooooooo

Meta
ooo

Free and Open Source Software

Alexander Schoch

SSC TheAlternative | ETHZ and UZH

October 6, 2020

Free

100 % unrelated to price:

- ▶ German: "frei"
- ▶ French: "libre"
- ▶ Italian: "libero"
- ▶ Romansh: "liber"

Open Source

Source code example:

```c
#include <stdio.h>
#define N 10 //Larger N for slower learners

int main()
{
    int i;

    puts("Repeat after me:");

    for (i = 0; i < N; i++)
    {
        puts("FOSS is also great for non-programmers!");
    }

    return 0;
}
```

Software

Binary file hexadecimal view:

```
                              [...]
00001660  43 00 5f 5f 69 6e 69 74 5f 61 72 72 61 79 5f 73  |C.__init_array_s|
00001670  74 61 72 74 00 5f 47 4c 4f 42 41 4c 5f 4f 46 46  |tart._GLOBAL_OFF|
00001680  53 45 54 5f 54 41 42 4c 45 5f 00 5f 5f 6c 69 62  |SET_TABLE_.__lib|
00001690  63 5f 63 73 75 5f 66 69 6e 69 00 5f 49 54 4d 5f  |c_csu_fini._ITM_|
000016a0  64 65 72 65 67 69 73 74 65 72 54 4d 43 6c 6f 6e  |deregisterTMClon|
000016b0  65 54 61 62 6c 65 2e 64 61 74 61 5f 73 74 61 72  |eTable.data_star|
000016c0  74 00 70 75 74 73 40 40 47 4c 49 42 43 5f 32 2e  |t.puts@@GLIBC_2.|
000016d0  32 2e 35 00 5f 65 64 61 74 61 00 5f 66 69 6e 69  |2.5._edata._fini|
000016e0  00 5f 5f 6c 69 62 63 5f 73 74 61 72 74 5f 6d 61  |.__libc_start_ma|
000016f0  69 6e 40 40 47 4c 49 42 43 5f 32 2e 32 2e 35 00  |in@@GLIBC_2.2.5.|
00001700  5f 5f 64 61 74 61 5f 73 74 61 72 74 00 5f 5f 67  |__data_start.__g|
00001710  6d 6f 6e 5f 73 74 61 72 74 5f 5f 00 5f 5f 64 73  |mon_start__.__ds|
00001720  6f 5f 68 61 6e 64 6c 65 00 5f 49 4f 5f 73 74 64  |o_handle._IO_std|
00001730  69 6e 5f 75 73 65 64 00 5f 5f 6c 69 62 63 5f 63  |in_used.__libc_c|
00001740  73 75 5f 69 6e 69 74 00 5f 65 6e 64 00 5f 73 74  |su_init._end._st|
00001750  61 72 74 00 5f 5f 62 73 73 5f 73 74 61 72 74 00  |art.__bss_start.|
00001760  6d 61 69 6e 00 5f 4a 76 5f 52 65 67 69 73 74 65  |main._Jv_Registe|
00001770  72 43 6c 61 73 73 65 73 00 5f 5f 54 4d 43 5f 45  |rClasses.__TMC_E|
00001780  4e 44 5f 5f 00 5f 49 54 4d 5f 72 65 67 69 73 74  |ND__._ITM_regist|
00001790  65 72 54 4d 43 6c 6f 6e 65 54 61 62 6c 65 2e 5f  |erTMCloneTable._|
000017a0  69 6e 69 74 00 00 00 00 00 00 00 00 00 00 00 00  |init............|
                              [...]
```

Acronym
○○○

History
●○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○○

Users/Developers
○○○○○○○

Meta
○○○

How did we get to free and open?

Early Computing

~1950-1960: Public-domain software, academic "hacking" culture

Bendix G-15 photo by Gah4 - CC BY-SA 4.0 via Commons

Birth of the Proprietary Software Industry

∼1970-1980:

► Increasingly easy to withold control (e.g. binary-only distribution)

► United States vs. IBM (1969, dropped 1982): "Bundled Software is anticompetitive"

► Computer Software Copyright Act (1980): "Software is patentable"

Acronym History Licences Advantages Challenges Examples Users/Developers Meta
○○○   ○○○●○○   ○○○○○   ○○○○○   ○○○○   ○○○○   ○○○○○○○   ○○○

How did we get to free and open?

Confronting the Software Industry

- ▶ Promotes 4 universal freedoms: study, distribute, create, and modify computer software.
- ▶ Promotes moral obligation to use free software.
- ▶ "Ethical" (dogmatic?)

Acronym
○○○

History
○○○●○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

How did we get to free and open?

Working with the Software Industry

open source
initiative

2017 top Linux kernel
contributor organizations [1]

Company	Changes	%
Intel	10,833	13.1%
none	6,819	8.2%
Red Hat	5,965	7.2%
Linaro	4,636	5.6%

▶ Promotes open-source principles [2].

▶ "Rebranded" the free software movement.

▶ Google, Facebook, etc. produce open-source software.

Acronym
○○○

History
○○○○○●

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

How did we get to free and open?

Sticking together

FOSS

Acronym	History	Licences	Advantages	Challenges	Examples	Users/Developers	Meta
○○○	○○○○○	●○○	○○○○○	○○○○	○○○	○○○○○○○	○○○

Key ideas of FOSS

GNU Public License - based on the FSF's "four freedoms"

▶ The freedom to run the program as you wish, for any purpose

▶ The freedom to study how the program works, and change it so it does your computing as you wish

▶ The freedom to redistribute copies so you can help your neighbor

▶ The freedom to distribute copies of your modified versions to others

Acronym ○○○ | History ○○○○○ | **Licences** ●○● ○○ | Advantages ○○○○○ | Challenges ○○○○ | Examples ○○○ | Users/Developers ○○○○○○○ | Meta ○○○
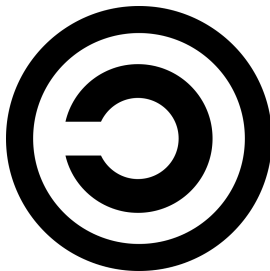
Key ideas of FOSS

BSD License - very permissive 3 clauses

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

▶ Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

▶ Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

▶ Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Acronym
○○○

History
○○○○○

Licences
○○●

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

Key ideas of FOSS

Copyleft

- ▶ All derivatives must for ever stay free
- ▶ May restrict interaction with non-free programs (e.g. library linking)
- ▶ "Viral Licensing"

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
●○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

Why is FOSS awesome?

Performance

Linus' Law: "Given enough eyeballs, all bugs are shallow."

You:

▶ experience less crashes.

▶ increase performance (especially for older hardware).

▶ get better support - if ever it still fails.

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○●○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

Why is FOSS awesome?

Education

You can:

▶ better manifest your intelligence and creativity.

▶ learn valuable, transferable skills.

▶ get better software habits

  ▶ Open Formats
  ▶ LaTeX
  ▶ Git
  ▶ CLI

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○●○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

Why is FOSS awesome?

Empowerment

You:

▶ free yourself from vendor lock-in.

▶ get control over any service running on your system.

▶ innovate faster, never having to ask for permission.

▶ have more freedom, even than on the open market.

Acronym
ooo

History
ooooo

Licences
ooo

Advantages
oooo●o

Challenges
oooo

Examples
ooo

Users/Developers
ooooooo

Meta
ooo

Why is FOSS awesome?

Security

You:

▶ benefit from more privacy.

▶ benefit from more safety.

▶ use software which is more transparent (as for government - i.e. trustworthy).

▶ have access to better encryption.

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○●

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

Why is FOSS awesome?

Sustainability

"the quality of being able to continue over a period of time"

The Software:

▶ depends on no single entity for its continuity.

▶ can be improved by all and degraded by none.

▶ is reproducible and transparent (as for science - i.e. comprehensible).

You:

▶ Can pool your resources with the world.

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○

Challenges
●○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

What to look out for now and in the near future.

Software as a Service

- ▶ Also known as SaaS or "the cloud".
- ▶ May be based on FOSS.
- ▶ Usage paradigm is unfree, closed, and unsustainable.

Regulation - from without

E.g. via laws and regulations:

▶ "Privacy" threatens admissible information sources.
▶ "Copyright" threatens admissible information matter.
   ▶ Embrace, Extend, Extinguish as a business model.
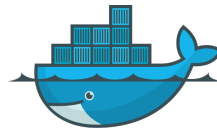▶ "Consumer Protection" threatens admissible information destinations.

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○●○

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

What to look out for now and in the near future.

Regulation - from within

E.g. via political "Codes of Conduct", which can:

▶ make participation contingent on ideology (less free).

▶ encourage exclusion from the community (less open).

▶ reduce resource pooling (less sustainable).

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
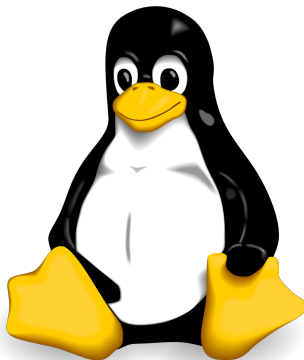○○○●

Examples
○○○

Users/Developers
○○○○○○○

Meta
○○○

What to look out for now and in the near future.

App-ification / Containerization

- ▶ Possibly FOSS
- ▶ Centralized certification and/or distribution
- ▶ Every application as an island:
  - ▶ Reduces user control (less free).
  - ▶ Reduces system transparency (less open).
  - ▶ Reduces system resource pooling (less sustainable).

| Acronym | History | Licences | Advantages | Challenges | Examples | Users/Developers | Meta |
| ooo | ooooo | ooo | ooooo | oooo | ●oo | ooooooo | ooo |

What kind of software is FOSS?

Linux Kernel

"Tux" by Larry Ewing, Simon Budig, Anja Gerwinski Licensed under Attribution via Commons

As of 2010 only 2 % of the Linux Kernel was written by Linus Torvalds.

Acronym	History	Licences	Advantages	Challenges	Examples	Users/Developers	Meta
○○○	○○○○○	○○○○○	○○○○○	○○○○	○●○	○○○○○○○	○○○

What kind of software is FOSS?

The GNU/Linux Operating System

Developed by the FSF:

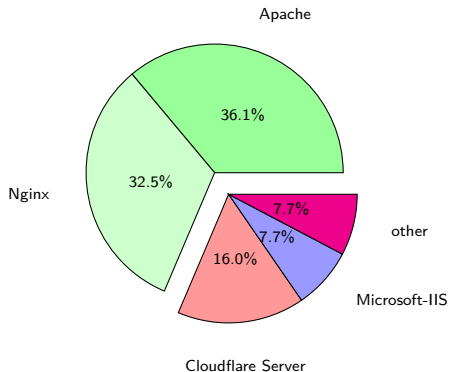▶ GNU Compiler Collection

▶ GNOME

▶ GNU Octave, GnuCash, etc.

"Gnu-and-penguin-color" by FSF - CC BY-SA 3.0 via Commons

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○●

Users/Developers
○○○○○○○

Meta
○○○

What kind of software is FOSS?

So Much More
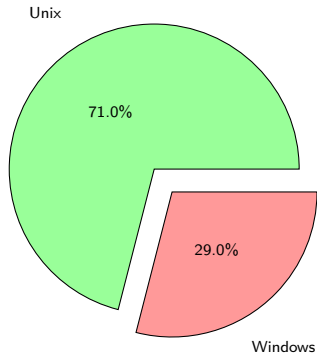
▶ Desktop: Chromium, Firefox, LibreOffice, LATEX, etc.

▶ Multimedia: mpv, mpd, Gmpc, VLC etc.

▶ Scientific: NumPy, matplotlib, NiPype, JabRef, R, etc.

▶ Server/Cloud: Apache, Openstack, WordPress, etc.

▶ Graphics: RawTherapee, GIMP, Inkscape, etc.

▶ etc.

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
●○○○○○○

Meta
○○○

Who benefits from FOSS?

The Internet

Web Server Market Share 2020-10-06 via w3techs

Website OS Market Share 2020-10-06 via w3techs

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○●○○○○○○

Meta
○○○

Who benefits from FOSS?

Public Institutions

► Governments (benefits: strategic, economic)
► ETH
  ► Linux cluster running CentOS
  ► Fedora/Windows dual boot on public computers
► UZH
  ► OpenStack-based cluster (ScienceCloud)

Acronym
○○○

History
○○○○○

Licences
○○○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○●○○○○○

Meta
○○○

Who benefits from FOSS?

Businesses

"Chromium Material Icon" by The Chromium Project - CC BY 2.5 via Commons

"Google Chrome Material Icon" by Google, Google Play. - BSD License via Commons

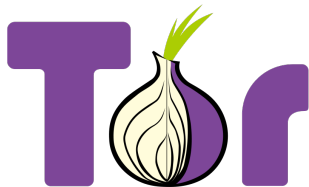| Acronym	History	Licences	Advantages	Challenges	Examples	Users/Developers	Meta
○○○	○○○○○	○○○	○○○○○	○○○○	○○○	○○○○●○○○	○○○

Who benefits from FOSS?

Businesses

©Red Hat, Inc.

- ▶ Founded 1993, subsidiary of IBM since 2019
- ▶ 2015–2018 stock price $\beta_{\mathrm{RHT}} = 1.16$ (e.g. $\beta_{\mathrm{MSFT}} = 1.22$)
- ▶ Very diverse open source products, e.g. OpenStack, RHEL, CloudForms

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○●○○

Meta
○○○

Who benefits from FOSS?

The Persecuted

"Tor-logo-2011-flat" by Tor Project - CC BY 3.0 via Commons

Free and Open by design, not by convention.

▶ Whistleblowers ▶ Marginalized Gvmts. ▶ (Alleged) Criminals

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○●○

Meta
○○○

Who benefits from FOSS?

The Busy and Creative

▶ Students

▶ Scientists

▶ Engineers

▶ Analysts

▶ Artists

*Get the opportunity to learn **the right way**, keep the freedom to do it **your way**.*

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○●

Meta
○○○

Who benefits from FOSS?

Most of all: You

▶ The sooner the better
▶ Less time-intensive than 1KP
▶ You have already started!

Acronym
○○○

History
○○○○○

Licences
○○○

Advantages
○○○○○

Challenges
○○○○

Examples
○○○

Users/Developers
○○○○○○○

Meta
●○○

About this presentation

What now?

▶ Q&A round
in a few seconds

▶ Come to the next "Linux Days" events: `http://thealternative.ch/`

▶ Join the next "Stammtisch":
This will not take place due to the coronavirus

▶ Give us feedback!
`https://feedback.thealternative.ch/`

These Slides

- ▶ Latest Slides:
  https://gitlab.ethz.ch/thealternative/courses/blob/master/FOSS_
  course/slides.pdf
- ▶ Source:
  https:
  //gitlab.ethz.ch/thealternative/courses/blob/master/FOSS_course/
- ▶ License: Slides by Alexander Schoch/Christian Horea, CC BY-SA 4.0

References

[1] J. Corbet and G. Kroah-Hartman.
    2017 Linux Kernel Development Report.
    www.linuxfoundation.org/2017-linux-kernel-report-landing-page/ (via the Linux
    Foundation).

[2] Open Source Initiative.
    The Open Source Definition.
    opensource.org/docs/osd (via the Internet Archive).