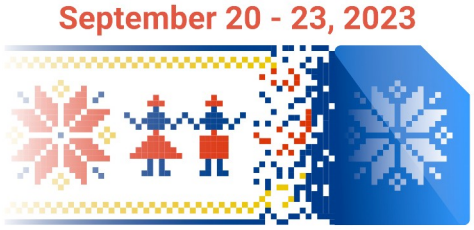
LibreOffice
The Document Foundation



Workshop: Introduction to LibreOffice Development

Office software, and the open source/free software development model

Hossein Nourikhah
Developer Community Architect at TDF

Self introduction

- **I am Hossein Nourikhah**
  - Ph.D. in information technology
  - Developer community architect for The Document Foundation (TDF)
  - Developer, former university lecturer and FOSS advocate
- **Any problems getting started with LibreOffice development?**
  - Contact me!
    - **Email:** hossein@libreoffice.org
    - **Wiki:** https://wiki.documentfoundation.org/User:Hossein
    - **IRC:** hossein at libreoffice-dev room in LiberaChat Network
      irc://irc.libera.chat/#libreoffice-dev

List of topics

- **Free / open source software (FOSS)**
  - What is FOSS?
  - Ideas and the history of FOSS
  - Problems with proprietary software
  - Importance and impact of FOSS throughout the world
  - FOSS development model
  - Different FOSS Licenses
- **Knowing LibreOffice** **hands on experience**
  - Multiple applications
  - Installing LibreOffice for different platform
  - Working with LibreOffice software

Free / open source software (FOSS)

What is Free / open source software (FOSS)?

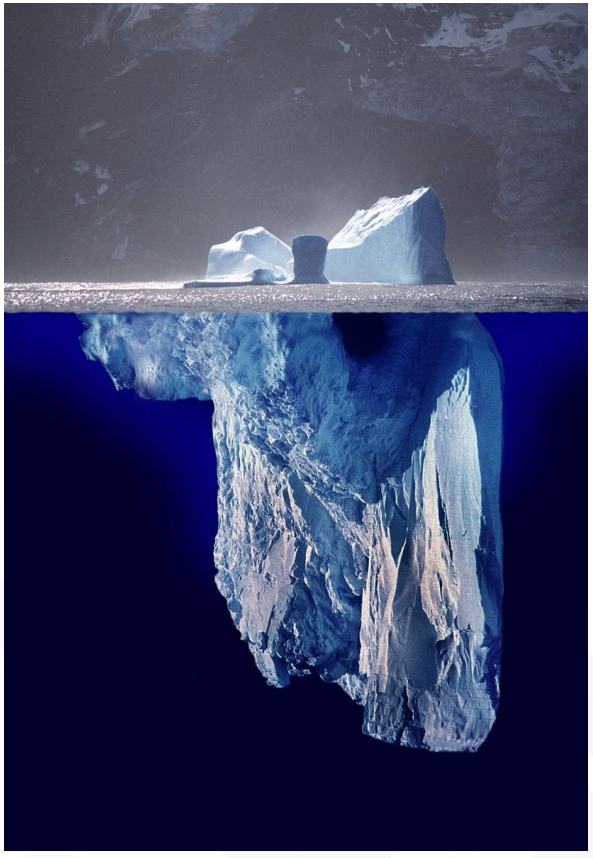
- **Free / open source software (FOSS)**
  - Software that anyone can
    - **Run the program** for any purpose; Not always, but usually is free of charge / low cost
      - Free as in freedom vs free goods / service
    - **See the source code** and rebuild the software using the available computer recipes
    - **Study and change the source**, if you want
    - Even **release it again**, with your changes!
  - Different licenses (will be discussed later)
    - Different interpretations
    - Different rights

Ideas and the history of FOSS

- Big picture of spreading and sharing knowledge
  - Well-known and well-respected throughout the history
  - Has evolved into different forms
- Specific to computer software
  - Collaborative advances in early years: 50s and 60s through knowledge sharing
  - Efforts by huge corporations in 70s to limit the sharing culture
  - Free software foundation (FSF) in 80s
  - Introduction of Linux in 90s
  - Free / open source companies in late 90 and 2000s
  - Android and other popular software since 2010s
  - Giant corporations try to use, not confront since then

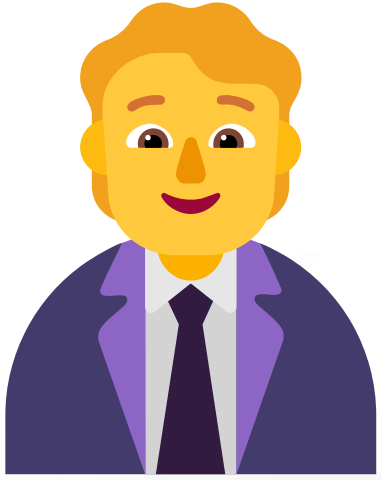
Problems with proprietary software

- **Ten issues with proprietary software**
  1. Security
  2. Privacy
  3. Data ownership
  4. Over-pricing
  5. Vendor lock-in
  6. Proprietary data formats
  7. DRM and its problems
  8. Poor customization
  9. Discouraging understanding, learning & teaching
  10. Bad for growing economies
- **These are only “tip of the iceberg”**



Importance of FOSS office suite

- Virtually in use by everyone
  - By Average people for their lives
  - By Small companies for their job
  - By Huge enterprises for their complex operations
  - In Schools for many of their tasks
  - In Academia for almost every task
  - In Governments for many of their tasks
- There are only a few big dominant proprietary players
  - MS Office on Windows / mac, Apple iWork on macOS
  - Google Docs and MS Office Live in cloud
  - MS Office, Apple iWork and Google docs, on mobile platforms



Importance of LibreOffice

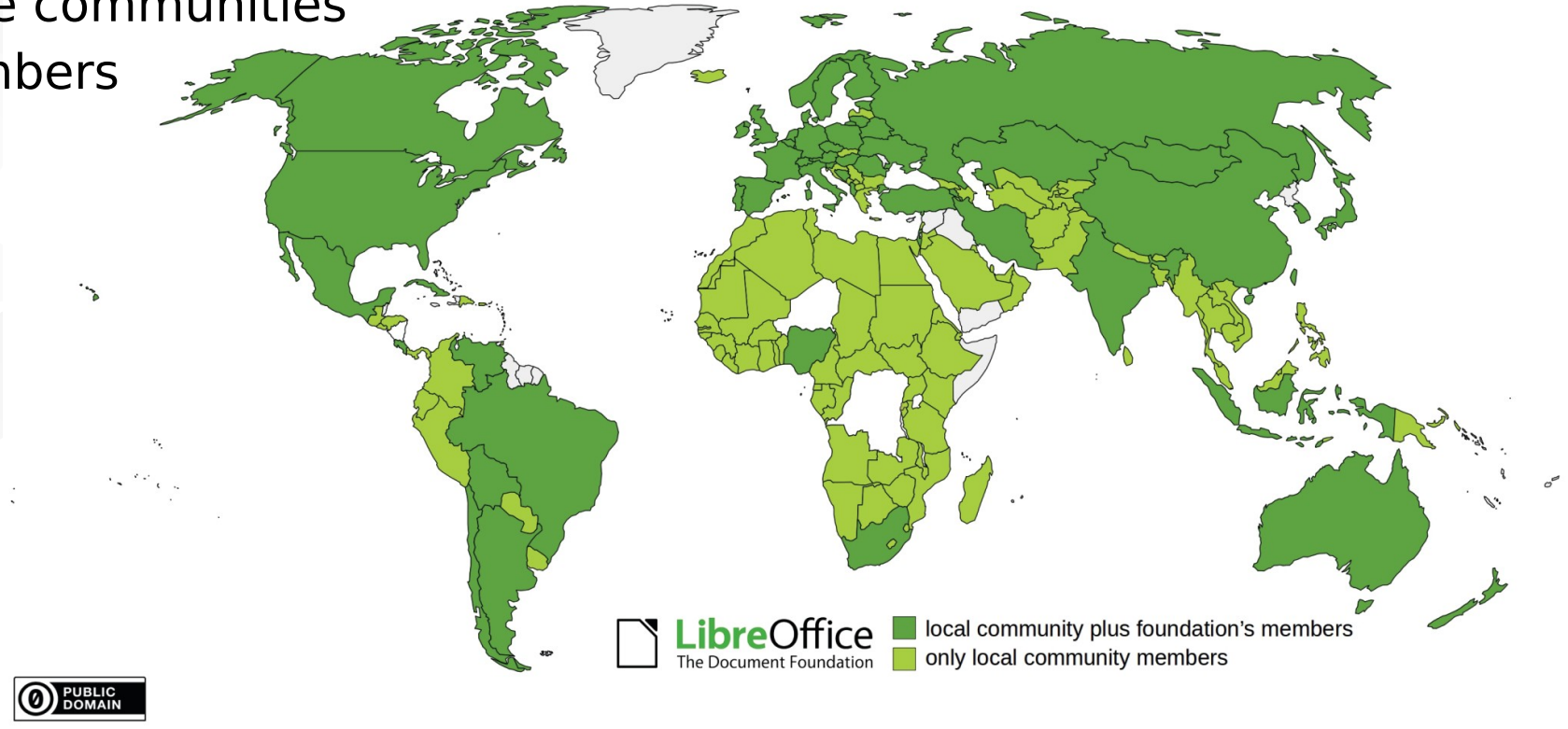
- LibreOffice: the leading open software office suite
  - Standard open file format called ODF (Open Document Format)
  - Compatibility with MS Office formats and many other formats
  - Feature rich
  - Easy to use
  - Different platforms (Linux, Windows, macOS, Android, ...)
  - Fast and efficient: written using mostly C++

Impact of FOSS throughout the world

- Multiple languages
  - Internationalization (i18n)
    - Ability to create output in different languages
  - Localization (l10n)
    - The UI is translated into different languages
- Both rich and poor countries benefit
  - For the rich: saving money to have more features
  - For the poor: avoiding expensive software

LibreOffice around the world

- 2022 annual report
  - Language communities
  - TDF members

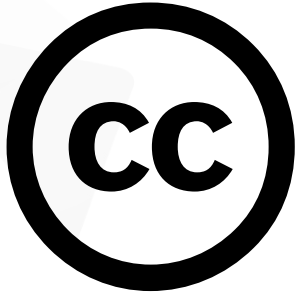


FOSS development model

- **FOSS development model**
  - Based on collaboration
  - Usually based on agile development
  - Development by contributors around the world
  - Extensive use of mailing lists and other communication channels
  - Both individuals and companies
  - Everyone has a voice
  - Those who do more and better, are chosen to decide
    - First, contribute and then get involved in decisin making

Different FOSS Licenses

- Difference licenses
  - GPL
  - LGPL
  - FDL
  - Apache license
  - Mozilla public license
  - Creative commons



Free as in Freedom

Free as in Freedom

Knowing LibreOffice

hands on experience

- If you are already a LibreOffice user, you won’t need this section

Installing LibreOffice

- **Get the binaries**
  - Go to https://libreoffice.org/download
  - Binaries available in the accompanied USB stick
  - Choose the right binary from:
    - OS Installation packages
    - Container installation
    - Portable application
  - Don’t forget SDK
- **Install the binaries**
  - Linux: Use .deb / .rpm / snap
  - Windows: Use the installer / unzip portable version
  - macOS: Use the installer

Linux

- Different distributions
  - Debian based
    - Extract the archive containing DEB files
    - Invoke **`sudo dpkg -i *.deb`**
    - Do similar for SDK
  - RPM based
    - Extract the archive containing RPM files
    - Invoke **`rpm -ivh *.rpm`**
    - Do similar for SDK
- Container based tools
  - Invoke: **`sudo snap install libreoffice`**
  - Alternatively, you can use Flatpak, AppImage

Ubuntu

Red Hat
Enterprise Linux

macOS

- macOS user: Which platform?
  - Mac OS X (Aarch64/Apple Silicon)
  - macOS x86_64 (10.14 or newer required)
- Use the .dmg installer to install
- Use the .dmg installer for SDK to install LibreOffic SDK

Windows

- Windows 7, 8, 10, 11
- Use appropriate installer (MSI)
  - 32 or 64 bit MSI?
  - These days, mostly 64 bit
  - Use the SDK installer
- Portable version
  - Unzip it (if it is not already done)
  - It is available in the accompanied USB stick
- Alternatively, you can use Chocolatey

Knowing LibreOffice

- Multiple applications
  - Writer → Word processor
  - Calc → Spreadsheet
  - Impress → Presentation
  - Draw → Vector image editing
  - Base → Database development
  - Math → Formula editor
- Native: Open Document Format (ODF)
- Compatibility: OOXML (DOCX, etc.)
- Old Compatibility: Office binary formats and various old document formats

Applications

- Writer
- Calc
- Impress
- Draw
- Base
- Math

Writer

- **LibreOffice Writer**
  - Word processing and desktop publishing
  - Alternative to MS Word
  - Support various document formats
    - Native format: ODT
    - Compatibility with MS Word: DOCX / DOC
  - Support various raster and vector image formats

hands on experience

- Create a small file with a sample text, 2 heading 1 titles, a table of contents. Then export it to PDF.

Calc

- **LibreOffice Calc**
  - Spreadsheet
  - Alternative to MS Excel
  - Support various document formats
    - Native format: ODS
    - Compatibility with MS Excel: XLSX / XLS
  - Support various data and spreadsheet formats
  - Supports import from CSV

hands on experience

- Create a simple Calc document, write 10 names, family names and GPA, 1 person each row, and use a formula to calculate the total GPA of all of them

Impress

- **LibreOffice Impress**
  - Presentation
  - Alternative to MS Powerpoint
  - Support various document formats
    - Native format: ODP
    - Compatibility with MS Powerpoint: PPTX / PPT
  - Support various presentation formats
  - Can be controlled via mobile phone
    - LibreOffice Impress remote

hands on experience

- Create an impress file containing the above text

Draw

- **LibreOffice Draw**
  - Vector graphics editing
  - Alternative to Corel Draw
  - Support various document formats
    - Native format: ODP
    - Compatibility with MS Excel: PPTX / PPT
  - Support various presentation formats
  - Ability to edit PDF files

hands on experience

- Draw a triangle and name its vertices A, B, C

Draw

- **LibreOffice Base**
  - Database management system (DBMS)
  - Alternative to Microsoft Access
  - Support different DBMSs for connection
    - Native format: ODB
    - Can import data from Access
    - No Compatibility with MS Access forms, code and reports
    - Two internal database engines: HSQLDB and Firebird

hands on experience

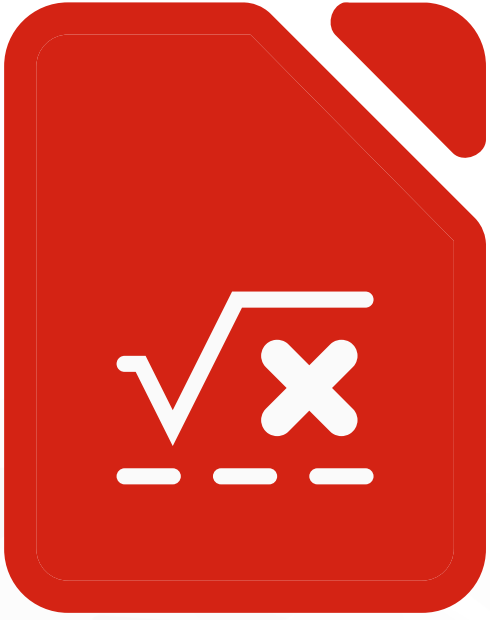
- Create a database with a table of code/name

Math

- **LibreOffice Math**
  - Mathematical formula editing
  - Alternative to MathType and MS Office equation editor
    - Native format: ODF
    - Compatibility with MathType and formulas embedded in MS Office documents

hands on experience

- Write a simple formula: $x = -b \pm \frac{\sqrt{b^2 - 4ac}}{2a}$

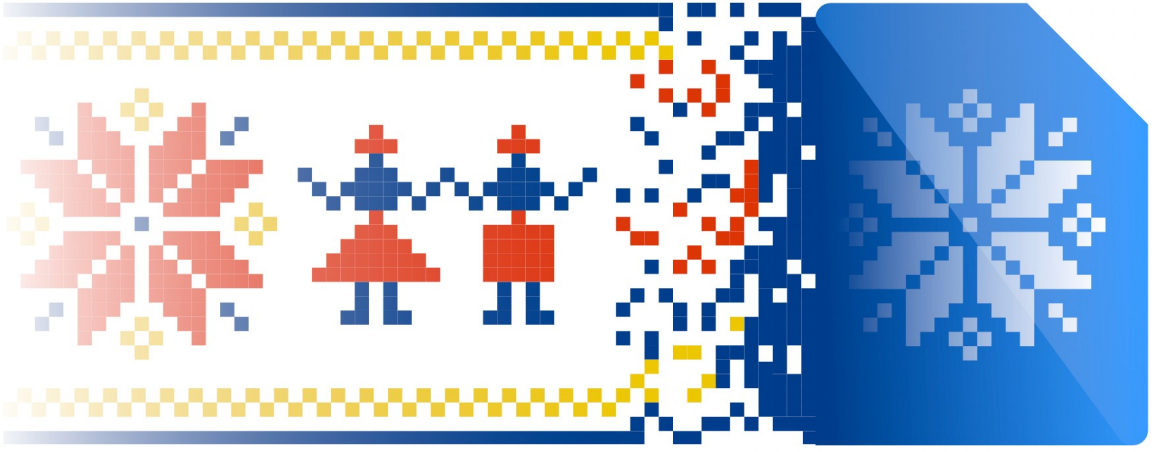


Thank you ...

- **Thank you for your patience!**

LibreOffice
The Document Foundation

September 20 - 23, 2023



1. Security (Proprietary)

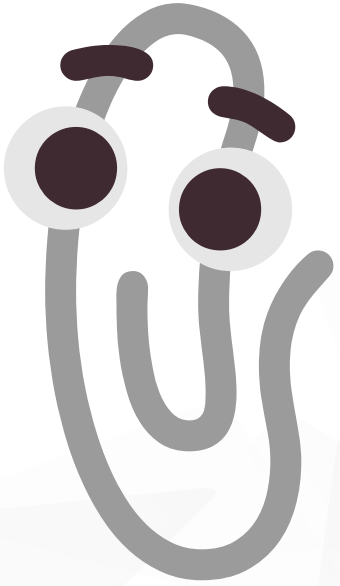
- **Proprietary software**
  - Black box, you don’t know the details of the code
  - Possible back doors
  - No way to ensure the security other than trusting the vendor
  - Uncertainty about the procurement, updates & changes
  - Only the company & affiliates are allowed to check the problems (white box) & fix them

1. Security (FOSS)

- **Open source / Free software**
  - Everything is open. From the code, to the metadata & assets
  - Any back doors would be visible
  - Every single details can be checked with automated tools
  - Certainty about the procurement, updates and changes with reproducible builds
  - Everyone is allowed to check the problems from a white box perspective and fix them

2. Privacy (Proprietary)

- **Proprietary software**
  - Not caring/prioritizing privacy
  - Privacy usually contradicts corporations' business models
  - Selling user data directly
  - (Ab)using people data is a norm
  - You have no option to leave
  - You can not choose the platform, and you can not disagree with the vendor / service provider from a specific place



2. Privacy (FOSS)

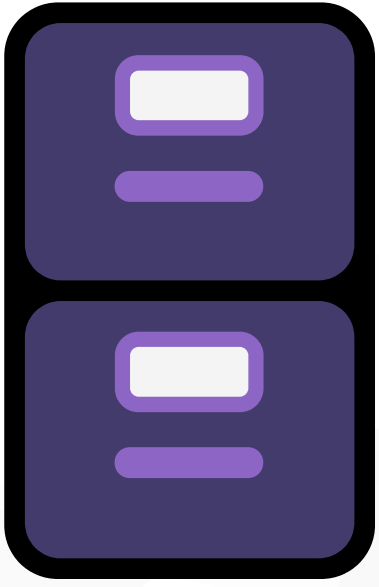
- **Open source / Free software**
  - Privacy is usually prioritized
  - Privacy is considered a reason for creating and using FOSS
  - Your data is at your hand
  - You can avoid data abuse
  - You can choose
  - You can run it on your device, self host, or choose among many service providers around the world

3. Data ownership (Proprietary)

- **Proprietary software**
  - You usually have to share data to be able to use software, both local and in the cloud
  - You don’t exactly know where your data ends up
  - Data protection policy can be there, but you don’t know what happens after a company is going bankrupt / get sold
  - Your data, your (bought/leased) service, but their rules

3. Data ownership (FOSS)

- **Open source / Free software**
  - Local: You retain your data
  - Cloud: You self host, or you choose your service provider
  - There may be some companies abusing the user data, but in that case, usually a new fork happens and it is legit and may dominate because of the advantage
  - Your data, your software/service, your rules



4. Pricing (Proprietary)

- **Proprietary software**
  - Usually overpriced
  - There is no reason for the price, companies charge as much as they can
  - Multiple million dollars from each country only for office software and related OS license
  - Having monopoly supports the over-pricing
  - Institutions / people should pay for a feature forever

4. Pricing (FOSS)

- **Open source / Free software**
  - Free of charge, or very low cost, even when buying enterprise licenses from companies
  - The pricing has reasonable justification, and is proportional to the time needed to fix the software or providing services
  - There is no monopoly, so competition can have positive impact on the prices
  - It is possible to sponsor a feature for cheaper price in mass

5. Vendor lock-in

- **Proprietary software**
  - Once you’re in, there is no easy way out!
  - Having incompatibility is a plus for them to keep the customers
  - Corporations do not like the markets that it is easy to enter, so they put barriers
  - Your data / archive may be locked physically or logically
  - Competitors are in many cases banned and forbidden

5. Vendor selection (FOSS)

- **Open source / Free software**
  - You can always migrate to other solutions
  - Compatibility with other FOSS and proprietary software is always a priority
  - FOSS facilitates the introduction of software using the same conventions and formats like ODF
  - Data export is a common option
  - Competitors are allowed, and this is the essence of FOSS

6. Data formats (FOSS)

- **Proprietary software**
  - Data formats may be closed, and need royalty
  - Bad for the interoperability
  - You usually have to buy the tools / libraries from the vendor

6. Data formats (FOSS)

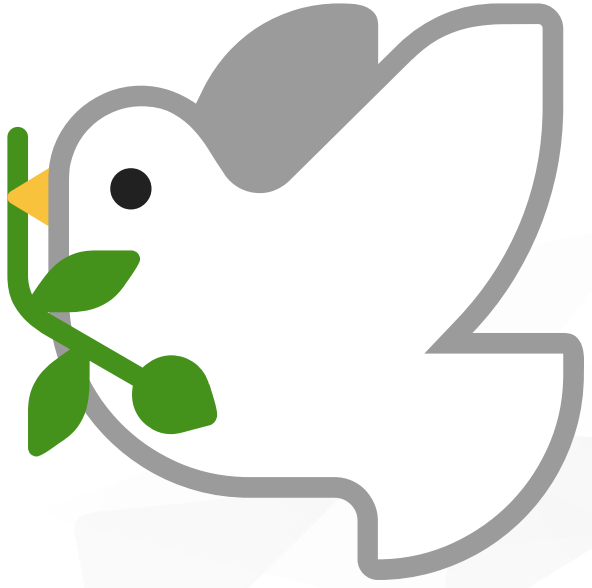
- **Open source / Free software**
  - Data formats are usually open, and royalty-free
  - Good for the interoperability
  - Free / open source tools and libraries readily available

7. DRM (Proprietary)

- **Proprietary software**
  - Restricting the people from using their legitimately bought contents
  - Brings another set of problems
    - More control over users
    - Discrimination based on software / hardware / location
    - Harder to use
    - Preventing format conversion
    - Limited platform support
    - Reduced security

7. DRM (FOSS)

- **Open source / Free software**
  - No DRM, which is bad by design
  - User is responsible for the legitimate usage of the contents
  - Brings more benefits
    - More freedom
    - Lets user choose their software / hardware / location
    - Easier to use
    - Ability of format conversion
    - Ability to use on different platforms



8. Poor customization

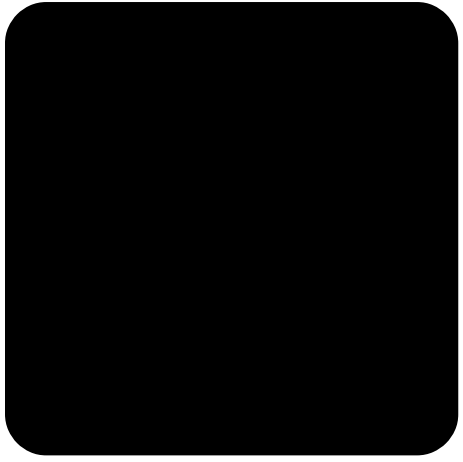
- **Proprietary software**
  - You are limited to vendor decision
  - The port is as best limited to major platforms
  - If Linux port is in contradiction to corporate policies, it won’t happen. It may be even removed!
  - Not every customization is desirable for the vendor

8. Good customization

- **Open source / Free software**
  - There is no limitation
  - It is common to see the software is ported to many platforms beyond some major ones
  - Linux port is usually among the first and prioritized things
  - It is the matter of the choice of the people to do the customization

9. Discouraging understand, learn and teach

- **Proprietary software**
  - Reverse-engineering is prohibited
  - Discussing about the details of the software internals is usually prohibited by the NDAs
  - Using the architectures, algorithms and methods similar to the other proprietary software may be prohibited by the patents



9. Encouraging understand, learn and teach

- **Open source / Free software**
  - Not only reverse engineering, but all sorts of analysis of code is usually done and encouraged
  - Discussing the details of the software internals is encouraged
  - Not only the architectures, algorithms and methods can be used elsewhere, but the whole code can be used directly, changed or unchanged

10. Bad for growing economies

- **Proprietary software**
  - Good places are already taken
  - Companies pay what the vendor decides
  - Expensive infrastructure for growth
    - Usually include royalty
    - Issues with copyright, patent, etc.

10. Good for growing economies

- **Open source / Free software**
  - There is always place for providing innovative solutions
  - Companies pay for the development, and maintenance
  - Flexible, customize-able Infrastructure
    - Usually royalty free
    - Copyleft or compatible, patent protection in many cases
    - Future improvements will be accessible to all