BE-CSE-2, VII-Semester, Free and Open Source Software

Unit-II.b : Licenses, Patents, Opportunities, Internationalization

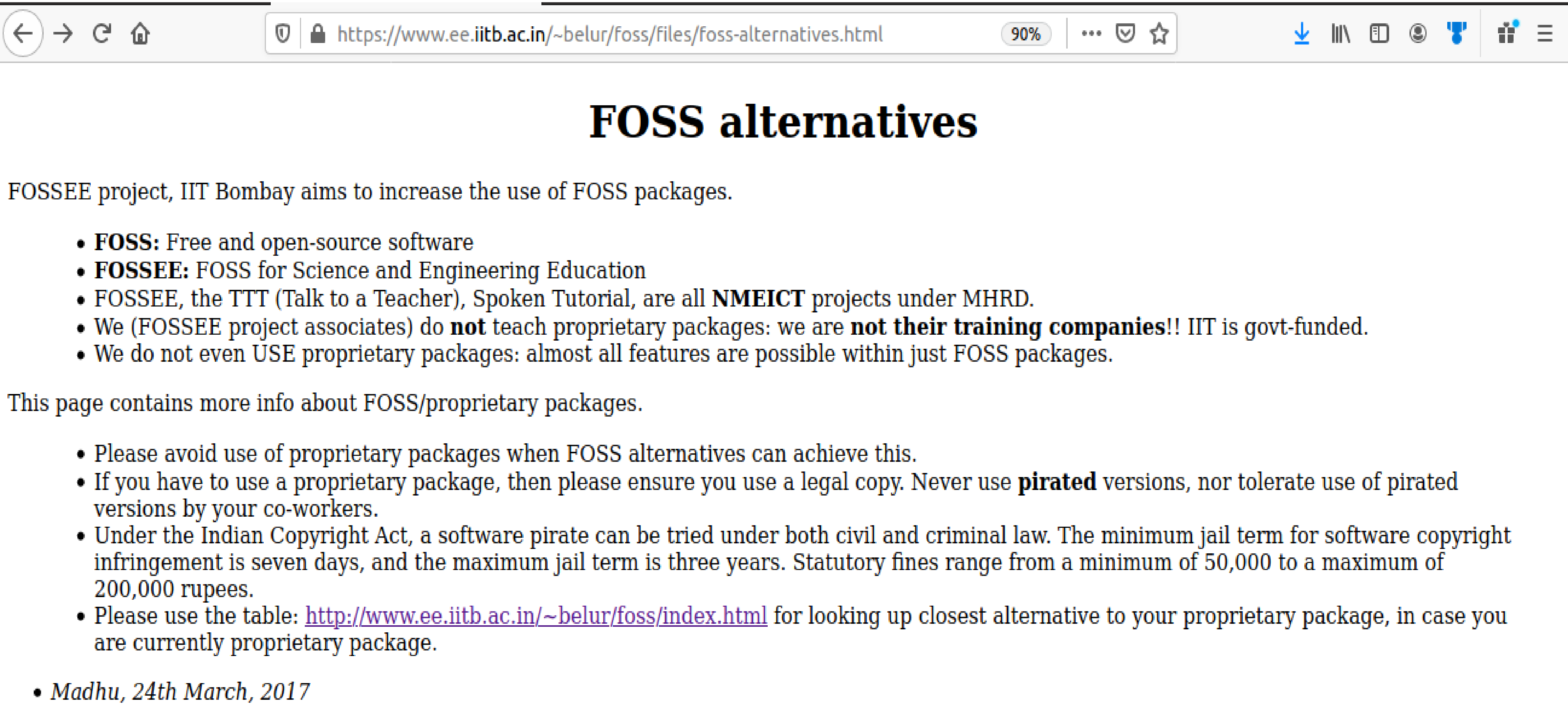
Outline

- Licenses
- Open Source Licenses
- Copyright Vs. Copyleft
- Patents
- Zero-marginal cost
- Income-generation opportunities
- Internationalization

Software Licenses

- What is a license?
- Do we require require license for software?
- What are the implications of unauthorized software usage?
- What are the different types of open source licenses?

https://www.ee.iitb.ac.in/~belur/foss/files/foss-alternatives.html



https://www.ee.iitb.ac.in/~belur/foss/files/foss-alternatives.html
90%

FOSS alternatives

FOSSEE project, IIT Bombay aims to increase the use of FOSS packages.

- **FOSS:** Free and open-source software
- **FOSSEE:** FOSS for Science and Engineering Education
- FOSSEE, the TTT (Talk to a Teacher), Spoken Tutorial, are all **NMEICT** projects under MHRD.
- We (FOSSEE project associates) do **not** teach proprietary packages: we are **not their training companies**!! IIT is govt-funded.
- We do not even USE proprietary packages: almost all features are possible within just FOSS packages.

This page contains more info about FOSS/proprietary packages.

- Please avoid use of proprietary packages when FOSS alternatives can achieve this.
- If you have to use a proprietary package, then please ensure you use a legal copy. Never use **pirated** versions, nor tolerate use of pirated versions by your co-workers.
- Under the Indian Copyright Act, a software pirate can be tried under both civil and criminal law. The minimum jail term for software copyright infringement is seven days, and the maximum jail term is three years. Statutory fines range from a minimum of 50,000 to a maximum of 200,000 rupees.
- Please use the table: http://www.ee.iitb.ac.in/~belur/foss/index.html for looking up closest alternative to your proprietary package, in case you are currently proprietary package.

- *Madhu, 24th March, 2017*

INTELLECTUAL PROPERTY
TM
LOGO
Trademark, Copyright or Patent
What should I opt to protect my intellectual property?
www.LegalWiz.in

https://www.legalwiz.in/blog/difference-between-trademark-copyright-patent-ipr-in-india

Licenses

- Is permission or right granted to engage in some act; Without a license the act might be otherwise unlawful
- It is a legal authority granted to a person or institution to do some thing
- It is also referred as a document that specifically describes these permissions and rights
- License must be automatic, no signature requirement
- Software licenses are considered as Copyright law
- **Open source licenses** define the **privileges** and **restrictions** a licensor must follow in order to use, modify or redistribute the open source software
- Open source software includes **software with source code** in the public domain and software distributed under an o**pen source license**

Open Source Licenses

- Example:
  - Apache License
  - BSD License
  - GNU General Public License
  - GNU Lesser General Public License
  - MIT License
  - Eclipse Public License
  - Mozilla Pubic License
- Proliferation of open source license is one of few -ve aspects of the open source movement as it is difficult to understand the legal implications of the different licenses

Public Domain

- All the free software and open source software is not public domain software
- They may be copyrighted and covered by a license, just a license that gives people more rights than they are used to
- Public domain program is one upon which the author has **deliberately surrendered his copyrights**
- One can treat the public domain public domain program as his own property and can do what he wants
- It can be re-licensed, remove the author’s name and treat it as your own work
- If you are doing **lot of work** with public domain software, then you can you apply for your own copyright to the program and re-license; you can keep **private** and **apply** for GPL or similar
- You can easily take a public domain program, private, by declaring a copyright and applying your own license to it simply declaring “All Rights Reserved”

Free Software Licenses in General

- Ownership of program always lies with the owner even you have purchased it
- Copyrighted programs are the property of copyright holder whether it is proprietary or open source license
- Program’s license grants some rights and have other rights under the definition of fair use in Copyright Law
- Author may issue software with different licenses (one GPL and other commercial; Ex: RedHat)
- One common feature of all open source software is, they disclaim all warranties; this is a protection for the authors and they contribute to the movement because of this.

GNU General Public License

- It is a political manifesto as well as software license and explain the rationale behind the license
- Created was by the assistance of some law professors, and is much better than the most of the other licenses
- We can also create our own license; but lot of care is to be taken other-wise there will be lot of legal complications. GPL helps to create our own license
- Text of GPL is not itself under GPL. Its license is simple. Every one is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
- Provisions of GPL satisfy the Open source definition, GPL does not require any of the provisions permitted by paragraph 4 of the OSD- ***Integrity of Author’s Source Code***
- GPL does not allow you to take “modifications private”, your modifications must be distributed under GPL which will lead to further improvements

GNU General Public License (cont.)

- GPL does not allow the incorporation of a GPL-ed program into a proprietary program- Ex: Debian
- Some loop holes with GPL; where the compilers and operating systems are linked with GPL-ed software, the result is a partially-free program. KDE author -Qt -LGPL
- GNU LGPL
  - Derivative of the GPL that was designed for software libraries
  - It allows GPL-ed programs to be incorporated into a proprietary program. Ex: C library provided with Linux systems,
  - LGPL-ed programs can be used to build proprietary programs
  - LGPL-ed program can be converted into a GPL-ed one at any time; once converted, that instance and its derivatives con’t be converted back to LGPL
  - Rest of the provisions of LGPL are similar to GPL

X, BSD and Apache Licenses

- There are different from GPL and LGPL
- X and BSD licenses are originally covered was funded by monetary grants off the US Govt (tax payers)
- X-licensed modifications can be private, allowed to sell binary versions without distributing source code, but still Open source where modifications need not require original license
- Many people adopted X-license and its variants including BSD and Aapche web server project
- BSD need a provision: “Software was developed at the University of California” as a foot notes- ad\vertising

Artistic, Netscape-Mozilla licenses

- Artistic license
  - Was developed for Perl; later used for software also
  - Section 5 of Artistic license prohibits sale of software, but allows aggregate of software distribution of more than one can be sold
  - Allows modifications and they it can be made private or pieces of software can be put in the public domain
- Netscape-Mozilla license
  - Netscape developed NPL when they made their product OSS and the open source version was Mozilla, Navigator was with Netscape's own product
  - NPL contains special privileges that apply for Netscape and nobody else
  - It gives the privilege of re-licensing modifications that you have made to their software;
  - Mozilla Public License (MPL) relaxed this clause

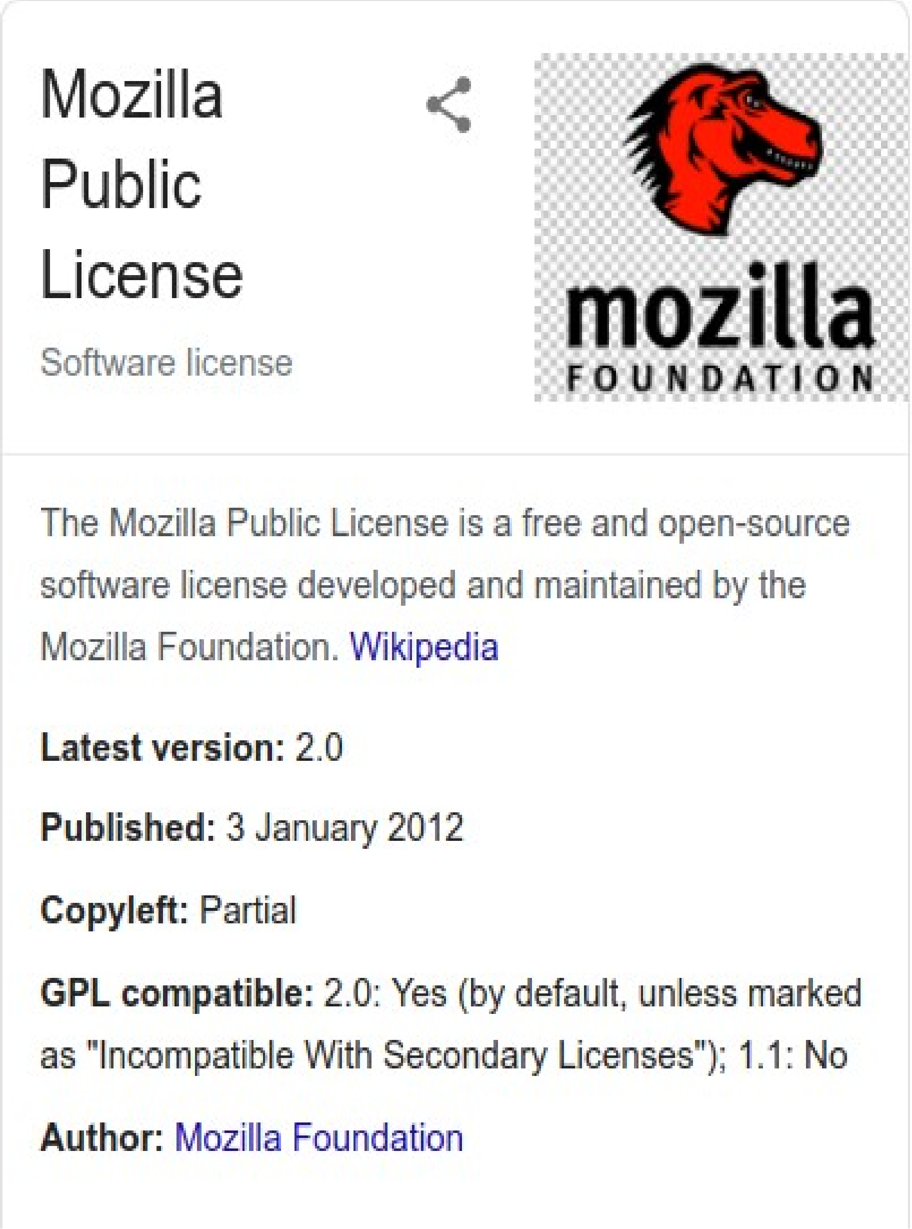
How to choosing a License?

License	Can be mixed with non-free SW	Modifications can be taken private and not returned to you	Can be re-licensed by anyone	Contains special privileges for the original Copyright holder over your modifications?
GPL				
LGPL	X			
BSD	X	X		
NPL	X	X		X
MPL	X	X		
Public Domain	X	X	X	

Apache License

- Apache Software Foundation is a free software license before version 2.0
- Requires the preservation of Copyright notice and disclaimer, but it is not a Copyleft license
- Allows source code for the development of free and open source software as well as proprietary and closed software
- All the software produced by the ASF or any of its projects or subjects is licensed according to the terms of the Apache License
- Some non-ASF software is licensed using Apache license
- ASF is a non-profit corporation to support Apache projects
- Apache projects are characterized by a collaboration, consensus based development process, an open and pragmatic software license
- It is a legal protection for its volunteers and projects

Apache Software License		“New” BSD
Author	ASF	Regent of UC
Version	2.0	N/A
Copyright	ASF	Public Domain
Published	Jan- 2004	1990
DFSG compatible	Yes	Yes
Free Software	Yes	Yes
OSI approved	Yes	Yes
GPL compatible	Yes + GPL v3	Yes
Copyleft	No	Yes
Linking from code with a different license	Yes	Yes



Mozilla Public License

Software license

The Mozilla Public License is a free and open-source software license developed and maintained by the Mozilla Foundation. Wikipedia

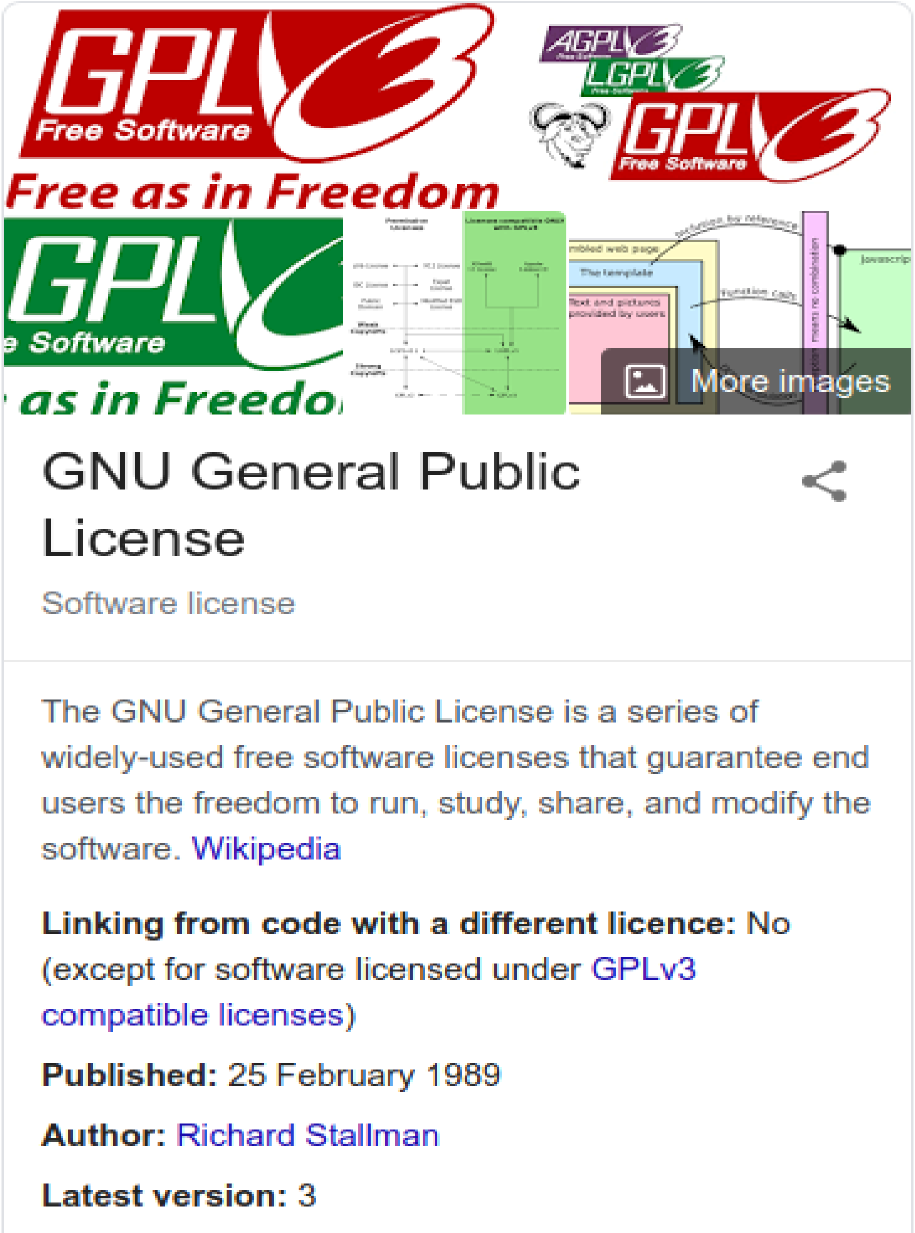
Latest version: 2.0

Published: 3 January 2012

Copyleft: Partial

GPL compatible: 2.0: Yes (by default, unless marked as "Incompatible With Secondary Licenses"); 1.1: No

Author: Mozilla Foundation



GNU General Public License

Software license

The GNU General Public License is a series of widely-used free software licenses that guarantee end users the freedom to run, study, share, and modify the software. Wikipedia

Linking from code with a different licence: No (except for software licensed under GPLv3 compatible licenses)

Published: 25 February 1989

Author: Richard Stallman

Latest version: 3

Debian Social Contract, Version 1.0

(July 5, 1997, superseded by Version 1.1, ratified on April 26,2004)

- Debian, the producers of the Debian GNU/Linux system, have created
- Debian-Free-Software Guidelines(DFSG) part of the contract, initially designed as a set of commitments that we agree to abide by, has been adopted by the free software community as the basis of the Open-Source Definition

“Social contract” with the Free Software Community

1. **Debian will Remain 100% Free Software**

   We promise to keep the Debian GNU/Linux Distribution *entirely free software*. As there are many definitions of free software, we include the guidelines we use to determine if software is “free” below. We will support our users who develop and run non-free software on Debian, but will never make the system depend on an item of non-free software

2. **We will give back to the Free Software Community**

   When we write new components of the Debian system, we will license them as free software. We will make the best system we can, so that free software will be widely distributed and used. We will feed back bug-fixes, improvements, user requests, etc. to the “upstream” authors of software included in our system

Debian Social Contract, Version 1.0 (cont.)

3. **We Won’t Hide Problems**

   We will keep our entries bug-report database open to public view at all times. Reports that users file on-line will immediately visible to others

4. **Our Priorities are Our Users and Free Software**

   We will be guided by the needs of our users and the free-software community. We will place their interests first in our priorities. We will support the needs of our users of operation in many different kinds of computing environment. We won’t object to commercial software that is intended to run on Debian systems, and we’ll allow others to create value-added distributions containing both Debian and commercial software, without any free fee from us. To support these goals, we will provide an integrated system of high-quality, 100% free software, with no legal restrictions that would prevent these kinds of use

5. **Programs That Don’t Meet Our Free-Software Standards**

   The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, may not restrict the program from being used in a business, or from being used for genetic research

Debian Free Software Guidelines (DFSG)

1. Free Redistribution- The License of Debian component may not restrict any party selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. This license may not require a royalty or other fee for such sale

2. Source Code – The program must include source code, and must allow distribution in source code as well as compiled form

3. Derived Works – The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software

4. Integrity of The Author’s Source Code – The license may restrict source-code from being distributed in modified form only if the license allows the distribution of “patch files” with the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

5. No Discrimination Against Persons or Groups – The license must not discriminate against any person or group of persons

Debian Free Software Guidelines (DFSG)

6. No Discrimination Against Fields of Endeavor – The license must not restrict anyone from making use of the program in a specific field of endeavor (ex. may not restrict from being used in a business, or from being used for genetic research)

7. Distribution of Licenses- The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties

8. License Must Not Be Specific to Debian – The rights attached to the program must not depend on the program’s being part of a Debian system. If the program is extracted from Debian and used or distributed without Debian but within the terms of the program’s license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the Debian system

9. license Must Not Contaminate Others Software - The license must not place restrictions on other software that is distributed along with the licensed software. (Ex., the license must not insist that all programs distributed on the same medium must be free software)

10. Examples Licenses - “GPL”, “BSD”, “Artistic” are examples that we consider “free”

Debian Social Contract (cont.)

- The concept of stating our "**social contract with the free software community**" was suggested by Ean Schuessler.
- This document was drafted by Bruce Perens, refined by the other Debian developers during a month-long e-mail conference in June 1997, and then accepted as the publicly stated policy of the Debian Project.
- Bruce Perens later removed the Debian-specific references from the Debian Free Software Guidelines to create "The Open Source Definition”
- Other organizations may derive from and build on this document. Please give credit to the Debian project if you do

https://opensource.org/docs/definition.php

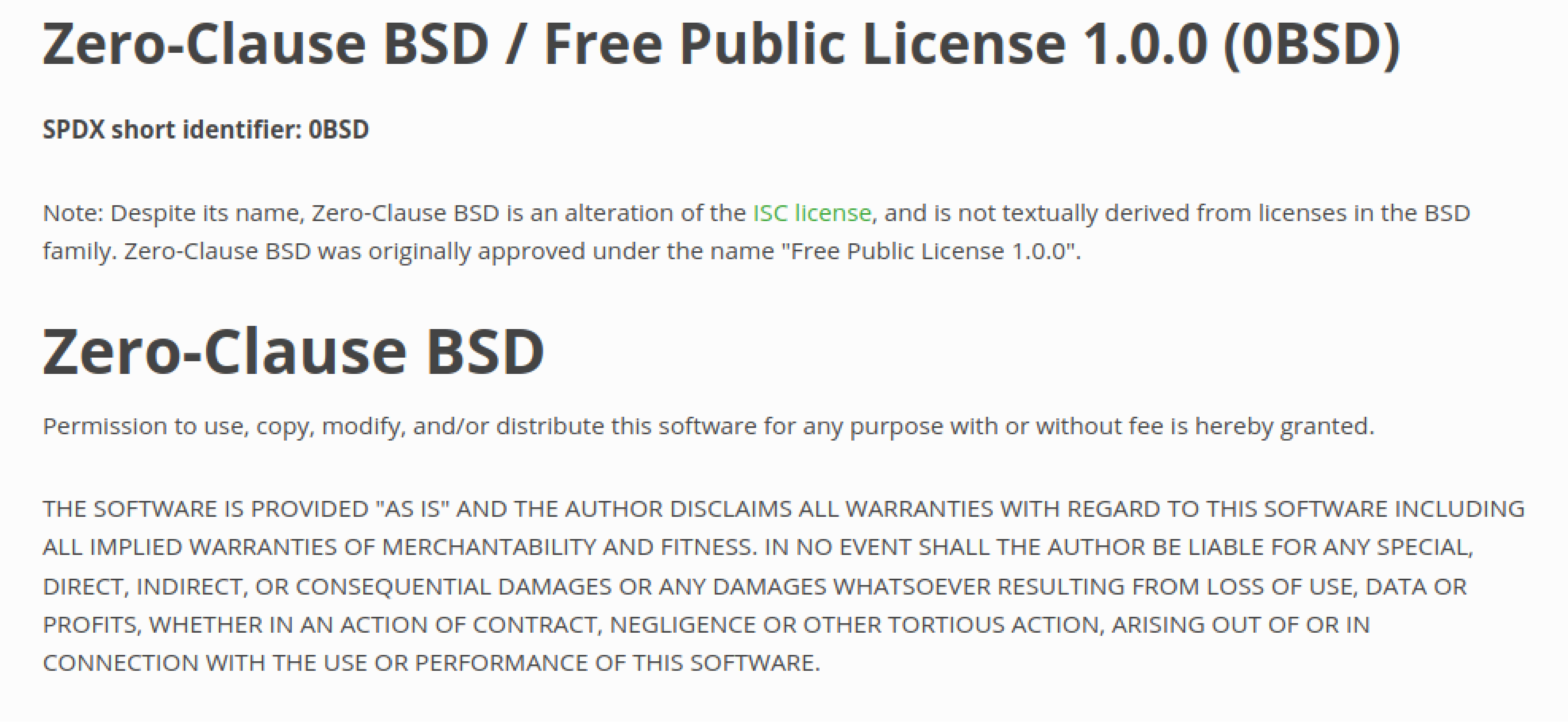
Open Source Definition - Criteria

1. Free distribution
2. Source Code (more elaborated statement)
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 Be Specific to a Product (must not be a part of some distribution), – in conjunction with the original software)
9. License Must Not Restrict Other Software (.....must be open source software)
10. License Must Be a Technology-Neutral – No provision of the license may be predicated on any individual technology or style of interface)

open source
initiative
Approved License®

Open Source Licenses & Standards

- Open Source license
  - Are licenses that comply with open source definition – in, brief, they allow software to be freely used, modified, and shared. To be approved by the Open Source Initiative(OSI), a license must go through the OSI review process
- Popular Open Source Licenses
  - Apache License 2.0
  - BSD 3-Clause “New” or “Revised” license,
  - BSD 2-Clause “Simplified” or “FreeBSD” license
  - GNU General Public License (GPL)
  - GNU Library or “Lesser” General Public License (LGPL)
  - MIT license
  - Mozilla Public License 2.0
  - Common Development and Distribution License
  - Eclipse Pubic License version 2.0
- All approved Licenses – Many other incenses are also OSI approved, but fall into other categories, such as special-purpose licenses superseded licenses, or retired licenses.   (list: https://opensource.org/licenses/category)



Zero-Clause BSD / Free Public License 1.0.0 (0BSD)

SPDX short identifier: 0BSD

Note: Despite its name, Zero-Clause BSD is an alteration of the ISC license, and is not textually derived from licenses in the BSD family. Zero-Clause BSD was originally approved under the name "Free Public License 1.0.0".

Zero-Clause BSD

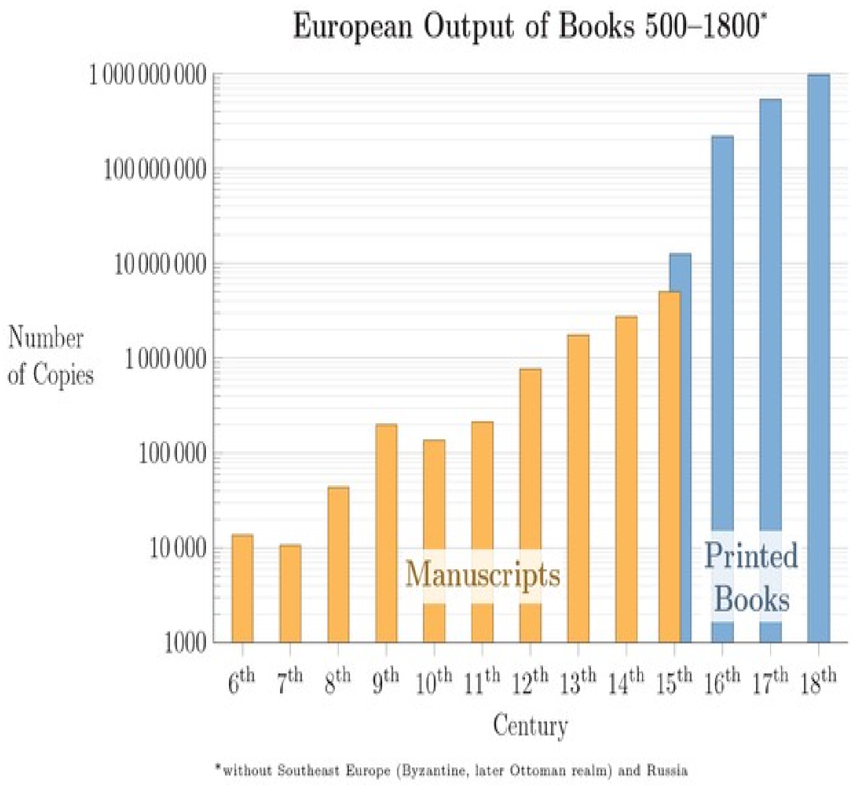
Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

https://opensource.org/licenses/0BSD

Copyright

- Relatively new invention in human history of legal concepts
- Copyright regulation reflects the social and technological transformation around human creative activity and distribution of the resultant profits
- Grants private rights to authors or copyright holders, encourages human creativity
- Also claims to recognize the larger public interest, particularly with respect to ***education, research, and access to information***
- Developed after printing press invention, 15-16th century



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

Copyright Acts

- Copyright law uses various means to balance public and private interests
- In the statute of Anne (1710), came into force, authorities are allowed to control the price of printed books according to the best judgment
- 1790, US constitution, authors are granted exclusive rights within a limited time
- 1886 international copyrights treaties, 1995 WTO TRIPS
- In copyright law, fair use of exceptions are specified to avoid the drawbacks of excessive assertions of exclusive rights and to attain a balance between conflicting interests
- 1957 Copyright act – India; 199, 2002, 2012

Copyright Office
Government of India

What can be Copyrighted?

- Applies to the expression of ideas in different forms including
  - Literary (poems, theses, fictional characters, plays)
  - Dramatic, motion pictures, choreography
  - Musical compositions, sound recordings,
  - Paintings, drawings, sculptures, photographs,
  - Computer Software, radio and Television broadcasts
  - Industrial designs
- Graphic designs and industrial designs have separate or overlapping laws (Patents, Trademarks)
- Copyright does not cover ideas and information themselves, only the form or manner in which they are expressed

How to Copyright our work?

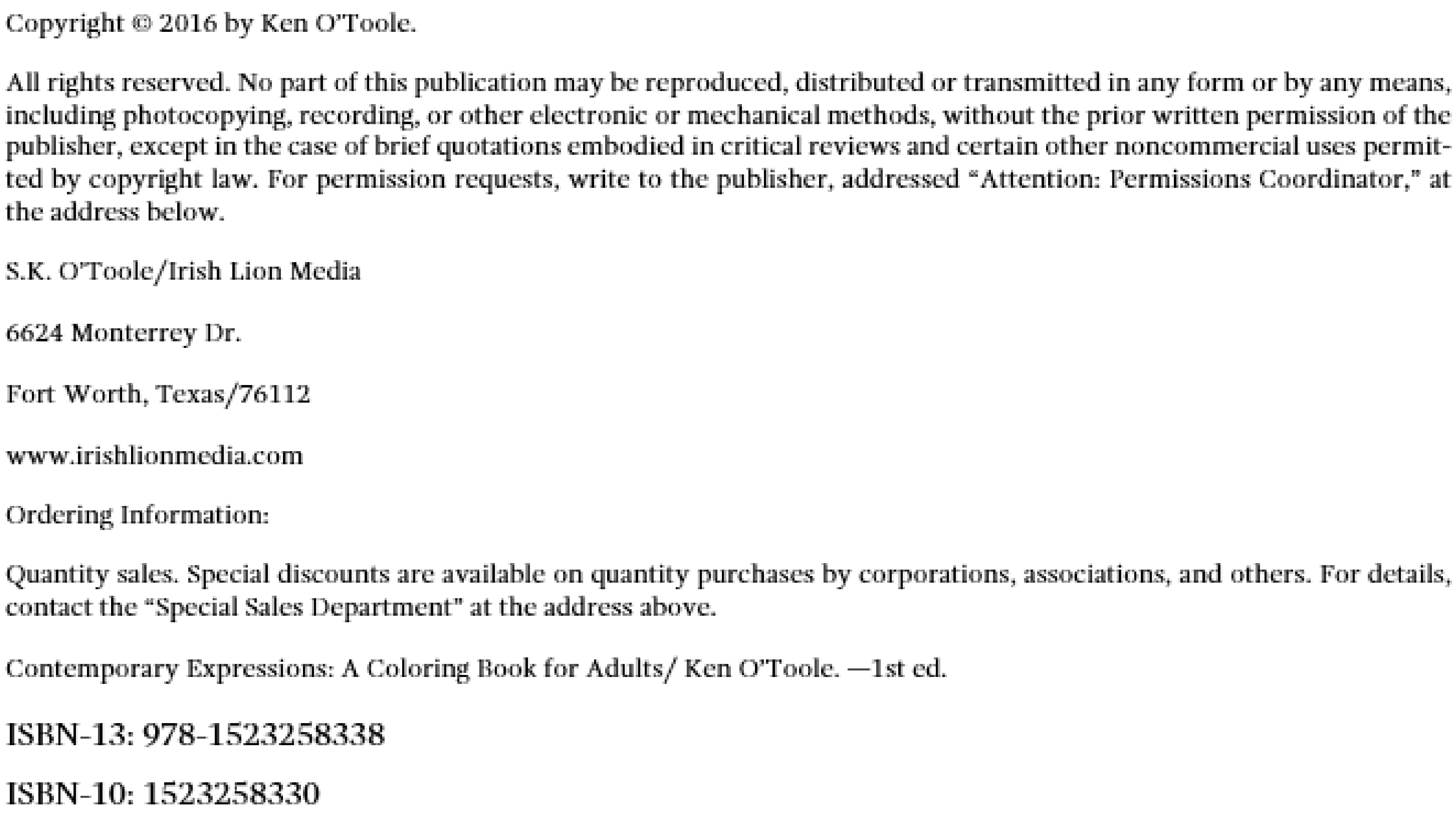
- Now a days, copyright does not require formalities
- Author need not publish, register, pay a registration fee of any kind, nor attach a copyright notice to his/her work
- Copyright is automatically applied to a work once it is created and the creator of the work automatically becomes the copyright holder
- Copyright is a legal concept enacted by most of the national Govts., gives exclusive rights to the creator of the original work for a limited period
- The rights may be use, copy, distribution, ownership, financial benefit etc.

Copyrights exceptions

- Copyright does not prohibit all copying or replication
- In US, the fair use doctrine (Copyright Act 1976) permits some copying and distribution without he permission of the copyright holder or payment to same
- The statute does not clearly define fair use, but instead gives four non-exclusive factors to consider in fair use analysis:

1. The purpose and character of one’s use
2. The nature of the copyrighted work
3. What amount and proportion of the while work was taken and
4. The effect of the use upon the potential market for or value of the copyrighted work

Sample Copyright of a textbook



Copyright © 2016 by Ken O'Toole.

All rights reserved. No part of this publication may be reproduced, distributed or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to the publisher, addressed "Attention: Permissions Coordinator," at the address below.

S.K. O'Toole/Irish Lion Media

6624 Monterrey Dr.

Fort Worth, Texas/76112

www.irishlionmedia.com

Ordering Information:

Quantity sales. Special discounts are available on quantity purchases by corporations, associations, and others. For details, contact the "Special Sales Department" at the address above.

Contemporary Expressions: A Coloring Book for Adults/ Ken O'Toole. —1st ed.

ISBN-13: 978-1523258338

ISBN-10: 1523258330

Economic Rights

- Owner may decide how it is to be used, and others can use it lawfully only if they have owner’s permission, often through a license
- Owner’s use of the property must, respect the legally recognized rights and interests of other members of the society
- Right owners can protect or prohibit
  - Reproduction of the work in various forms, such as printed publications or sound recordings
  - Distribution of copies of the work;
  - Public performance of the work;
  - Broadcasting or other communication of the work to public;
  - Translation of the work into other languages; and
  - Adaptation of the work, such as turning a novel into a screenplay

Moral rights

- Concerned with the non-economic rights of a creator
- Protects the creator’s connection with a work as well as the integrity of the work
- Accorded to individual authors and in any national laws they remain with the authors even after the authors have transferred their economic rights; may go for renewal
- US Copyright Law, several exclusive rights are grated to the copyright holder
  - Protection of the work
  - to determine and decide how and under what circumstances, the work may be marked, publicly displayed, reproduced, distributed etc
  - To produce copies or reproductions of the work and sell copies Including electronic copies)to import or export the work, to create derivatives, to perform or display the work publicly, to sell the rights to others, to transmit or display by radio, video or internet

Copyleft

- Copyleft is a method for making a program or other work free, and requiring all modified and extended versions of the program to be free as well
- Is a form of licensing and may be used to modify copyrights for works such as computer software, documents, music and art and scientific discoveries
- Most popular copyleft license is GPL. CC-SA(share-alike)
- It is also characterized as a copyright licensing scheme in which an author surrenders some but not all rights under copyright law

Copyleft cont.)

- Simplest way is to make the program free software is to put in the public domain, un-copyrighted, which allows people to share the program and their improvements
- With public domain software, un-coperated people convert the program into proprietary software. They can make changes, many or few and distribute the result as proprietary product.
- People who receive the program in that modified form, do not have the freedom that the original author gave them; middleman has stripped it away
- Copyleft also provides an incentive for other programmers to add to free software. Ex: **GNU C++ compiler** exist only because of this
- Copyleft also helps programmers to contribute improvements to free software get permissions to to do that

History

- Tiny Basic project – started in the newsletter “People’s Computer Company” in 1975
- Dennis Allenson wrote the specification for a simple version of Basic PL
- News letter became Dr. Dobb’s Journal of Tiny BASIC,
- Several hobbyists began writing BASIC language interpreters for their microprocessor based home computers and sending the source code
- In 1976 middle, several Tiny BASIC interpreter available for Intel 8080, Motorola 6800, and MOS Technology 6502. This was a very successful open source project
- 1976 May issue of Dr. Dobb’s Journal had Li-Chen Wang’s Palo Alto Tiny BASIC for Intel 8080 microprocessor with “@ALL WRONG RESERVED”
- Authors in other journals also included “COPYLEFT” Notice

History (cont.)

- LISP interpreter - another instance of Copyleft
- Richard Stallman – Symbolics – public domain – dispute –
- Symbolics improved the LISP and when Stallman asked, then it refused to give back
- Proceed to work towards eradicating the emerging behavior and culture of proprietary, which he named software hoarding
- He deemed it impractical in the short term to eliminate current copyright law and the wrongs he perceived it perpetuating, he decided to work within the framework of existing law
- In 1985, Stallman created his own copyright license, the Emacs GPL, first copyleft license, later evolved as GPL

Applying Copyleft

- Common practice for using copyleft is codify the copying terms for a work with a license
- Gives each person to posses a copy of the work the same freedoms as the author, including (from Free Software Definition)
  1. Freedom to use and study the work
  2. Freedom to copy and share the work with others
  3. Freedom to modify the work
  4. Freedom to distribute modified and therefore derivative works
- Freedoms do not ensure that a derivative work will be distributed under the same libera terms
- In order for the work for truly copyleft, the license has to ensure that the derived work can only distribute such works made under the same or equivalent license
- In addition to restrictions on copying, copyleft license address other possible impediments, including the rights cannot be later revoked and requiring the work and the derivatives are provided in a form that facilitates modifications; some case source code of the derived work is to be made available with the software itself

Applying Copyleft (cont.)

- Copyleft licenses necessarily make creative use of relevant rules and laws. Ex: when using copyright law. Those who contribute to a work under copyleft, usually must gain, defer or assign copyright holder status
- Some laws used for copyleft vary country to country. Ex: accepting to sell a software product without warranty.
- In standard GPL European countries, won’t permit software distributors to waive all warranties regarding a sold product

Types of Copyleft and other licenses

- Strong and Weak Copyleft
  - Strong: Copyleft provisions are to be efficiently imposed on all kinds of derived works. Ex: ***GPL***
  - **Weak**: not all derived works inherent the copyleft license, depends on the way they derived; Ex: ***LGPL***
  - **Weak**: copyleft is used for the creation of software libraries, to allow other software to link to the library, then be redistributed without the legal requirement for the work to be distributed under the library's copyleft Ex: **G Lib C**
  - Examples of non-copyleft, free software licenses: *X11, Apache, BSD, MIT etc.*
- Full and Partial Copyleft
  - **Full**: All parts of a work can be modified by consecutive authors
  - **Partial:** some parts of the work from the copyleft provisions; permitting unrestricted modifications; or not imposing all of the principles of copyleft on the work; Ex: artistic copyleft, full is not possible or desirable
  - Share-alike

Copyright Vs. Copyleft

Copyright	Copyleft
Restrictions on use, distribution, modification of his work	removing restrictions on distributing copies and derived works
Allows an author to prohibit others from reproducing, adapting, distributing copies	Encourages others to use, copy, distribute , modify and redistribute; but under same licensing as original
Allows authors to make the work as proprietary	Prohibits to make a work proprietary
Take away users’ freedoms	Guarantees the freedoms on derivatives
Doe not allow re licensing or sub-licensing; stops further development	Progressive development
Perpetual licensing	Viral; reciprocal licenses, self-perpetual
Source code is not available;	Licenses text is available in text, html, in the source code and manuals
EULA, contracts, MoUs	GPL, LGPL, FDL

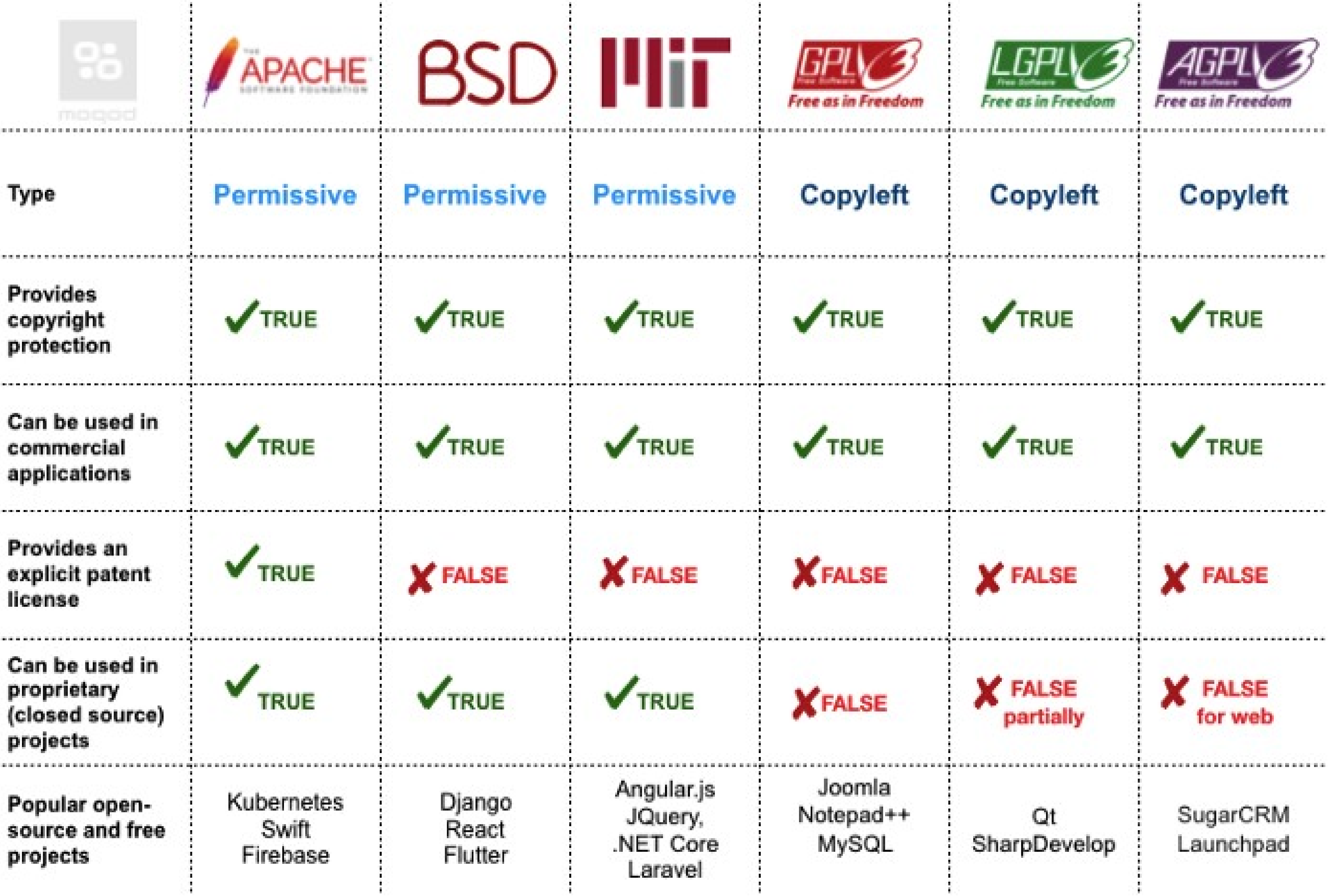
moqod

	Free software	Open-source software	Freeware	Public-domain software
Definition	**"FREE" is a matter of liberty, not price**	**"OPEN" doesn't just mean access to the source code**	**"FREE" refers to price, while freedom of the use is restricted by creator**	**"PUBLIC DOMAIN" belongs to the public as a whole**
Ground philosophy	Social movement	Development methodology	Marketing goals	Copyright disclamation
Ground rules	Four Freedoms https://www.gnu.org/philosophy/free-sw.html	Open Software initiative https://opensource.org/osd		Creative Common Organization https://creativecommons.org
Free of charge	Not necessary	Not necessary	✓YES	✓YES
Covered by copyright law	✓YES	✓YES	✓YES	✗ NO
Examples	Linux ubuntu	MySQL Apache	Skype Adobe Acrobat	SQLite

Copyright Copyleft Creative Commons Public Domain

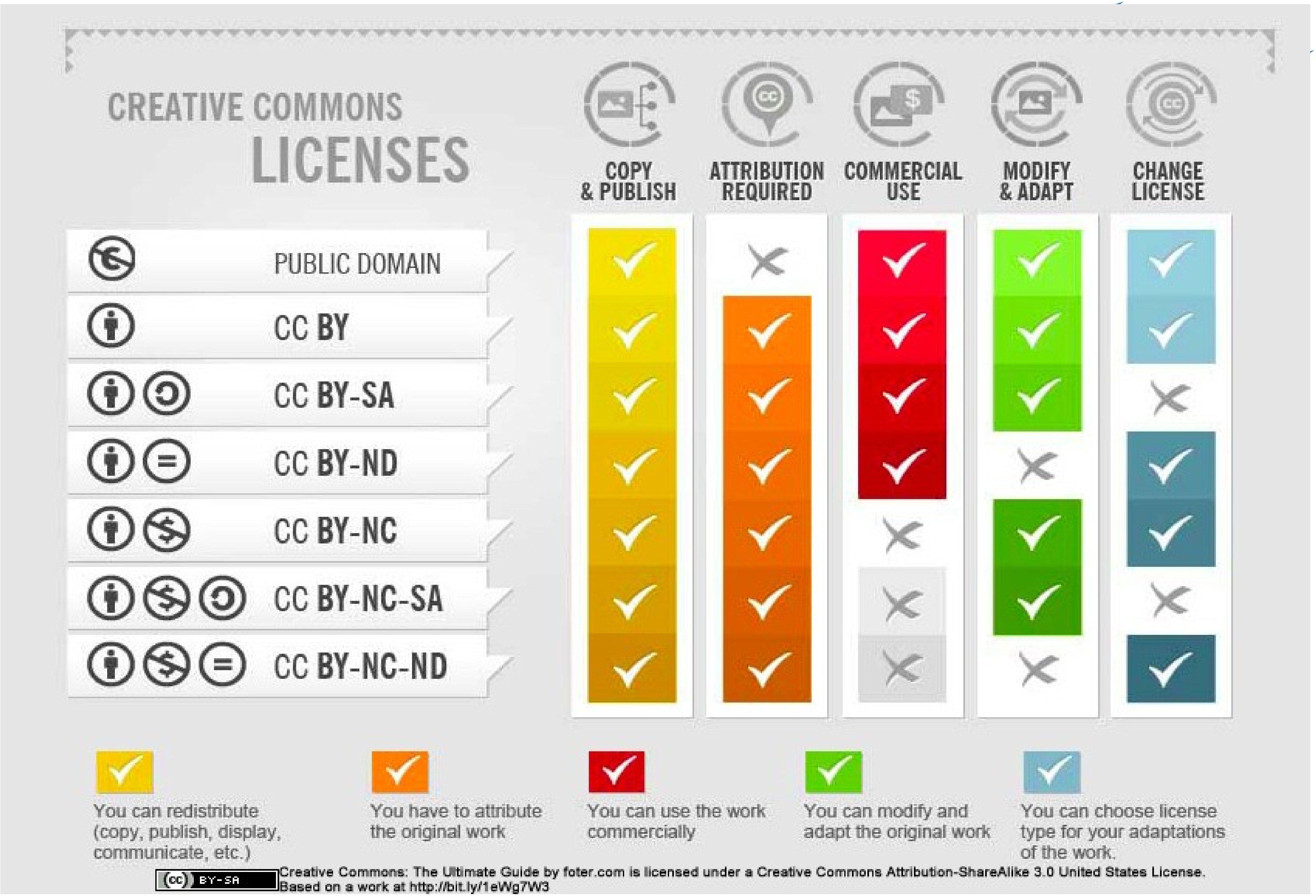
moqod

	Copyright	Copyleft	Permissive	Creative Commons
What is a user allowed to do with the code?	What creator dictates	What user wants under certain rules	What user wants with a few restrictions	What user wants without restrictions
Clause of the use	As creator dictates	Derivative work must be attributed to creator, open-source and copyleft	Derivative work must be attributed to a creator	Derivative work must be attributed to a creator
Source code	As creator dictates	Must be open	Don't have to be open	No specific terms about the distribution of source code
Is creator liable for bugs?	✓ YES	✓ YES	✗ NO	✗ NO
Re-licensing	As creator dictates	Derivative work cannot be released as proprietary software	Derivative work can be released under another license or as proprietary software	Derivative work can be released under another license or as proprietary software
Commercial restrictions	As creator dictates	Permitted	Permitted	Permitted



moqod	THE APACHE SOFTWARE FOUNDATION	BSD	MIT	GPLv3 Free Software Free as in Freedom	LGPLv3 Free Software Free as in Freedom	AGPLv3 Free Software Free as in Freedom
Type	Permissive	Permissive	Permissive	Copyleft	Copyleft	Copyleft
Provides copyright protection	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE
Can be used in commercial applications	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE
Provides an explicit patent license	✓ TRUE	✗ FALSE	✗ FALSE	✗ FALSE	✗ FALSE	✗ FALSE
Can be used in proprietary (closed source) projects	✓ TRUE	✓ TRUE	✓ TRUE	✗ FALSE	✗ FALSE partially	✗ FALSE for web
Popular open-source and free projects	Kubernetes Swift Firebase	Django React Flutter	Angular.js JQuery, .NET Core Laravel	Joomla Notepad++ MySQL	Qt SharpDevelop	SugarCRM Launchpad

https://medium.com/@moqod_development/understanding-open-source-and-free-software-licensing-c0fa600106c9



CREATIVE COMMONS LICENSES

	COPY & PUBLISH	ATTRIBUTION REQUIRED	COMMERCIAL USE	MODIFY & ADAPT	CHANGE LICENSE
PUBLIC DOMAIN	✓	✗	✓	✓	✓
CC BY	✓	✓	✓	✓	✓
CC BY-SA	✓	✓	✓	✓	✗
CC BY-ND	✓	✓	✓	✗	✓
CC BY-NC	✓	✓	✗	✓	✓
CC BY-NC-SA	✓	✓	✗	✓	✗
CC BY-NC-ND	✓	✓	✗	✗	✓

- ✓ You can redistribute (copy, publish, display, communicate, etc.)
- ✓ You have to attribute the original work
- ✓ You can use the work commercially
- ✓ You can modify and adapt the original work
- ✓ You can choose license type for your adaptations of the work.

(cc) BY-SA Creative Commons: The Ultimate Guide by foter.com is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Based on a work at http://bit.ly/1eWg7W3

Patent

INTELLECTUAL PROPERTY **INDIA**

- A patent is set of exclusive rights granted by a state to an inventor or his assignee for a fixed period of time in exchange for a disclosure of an invention
- Procedures and requirements may vary county to country according to the regional laws and international agreements
- Application must include one or more claims defining the invention which must be new, inventive and useful or industrial applicable
- Patents may be granted to people who invents or discovers any new and useful **process**, **machine, article of manufacture** or **composition of matter**, or any new and improvement
- Utility patents – includes processes, composition of matter, machines and manufacturers. Ex:biological, business methods, chemical and software patents
- Design patents –protects "ornamentation of an object"- protects only appearance
- Plant patents – protects new and distinctive plant

Patent (cont.)

- Patent is not a right to practice or use the invention, rather provides the rights to exclude others from making, using, selling, offering, for sale or importing the patented invention for the term of the patent, which usually 20 years from the date of filing
- Patent is a limited property right that a Govt. offer inventors to share their inventions with the public
- Patents may be sold, licensed, mortgaged, assigned or transferred, given away, or simply abandoned
- **Enforcement:**
  - Enforced through civil law-suites and vary country/territory- *civil action, criminal penalties, monetary compensation for past infringement and seek for injunction prohibiting the defendant from engaging in future for acts of patent infringement*
  - Patent owner has to prove the infringement
  - Majority of patent rights are not determined by litigation, but are resolved privately through patent licensing

Patent (cont.)

- Patent licensing agreements are effectively contracts in which the patent owner agrees not to sue the licensee for infringement of the lencensor’s patent rights, usually in return for a royalty or other payment
- US has more number of patents followed by EU and Japan
- **Ownership:**
  - **Natural persons** or **corporate entities** may apply for a patent and become the owner (inventor/inventors)
  - In US only natural person(s) may apply. If two persons are there both will be listed as inventor owns the patent separately
  - Inventors may **assign** ownership rights to a corporate entity (as a condition of their company)
  - The ability to assign ownership rights increases the liquidity of a patent as property

Patent (cont.)

- **Governing Laws:**
  - **Grant and enforcement** of patents are governed by national laws, and also by i**nternational treaties** (territorial nature)
  - Patent office, infringement in the national courts
  - Global patent harmony- **WTO active** – TRIPS agreement
  - Organize conventions
  - Authority for patent statues in different countries varies. In US, the Constitution empowers Congress to make laws to “**promote the Progress of Science and useful Arts**” , Patent & Trademark Office
  - There are international treaty procedures

Patent (cont.)

- **Application**
  - Filing a written application at relevant patent office
  - Contain the description of **how to make** and **use the invention**, and under some legislation, usefulness
  - Must also comprise “**claims**” – about the invention to provide for the public notice -what a patent covers
  - Patent application will go to legal process to check whether it meets the legal requirements related to patent-ability
  - Patent office examine the application for compliance with the requirements of the relevant patent law (objections communicated)
  - In most countries, there is no requirement to build a prototype to obtain a patent
  - Once granted, the patent is subject to renewal fees to keep the patent in force in each year

Patent (cont.)

- **Economics:**
  - Four primary incentives embodied in the patent system: To invent in the 1st place; to disclose the invention once made; to invest the sums necessary to experiment, produce and market the invention; and to design around and improve upon earlier patents

1. Patents provide incentives for for economically efficient R&D
2. Patents facilitate and encourage disclosure of innovations into the pubic domain for common good
3. In many industries, once an inventions exists, the cost of commercialization is far more than the initial conception cost
4. Patent rights create an incentive for companies to develop workaround the patented inventions

   small-time inventor can use the exclusive right status to become a licensor. Inventor can accumulate capital quickly from licensing the invention and may allow rapid innovation to occur

Software Patents

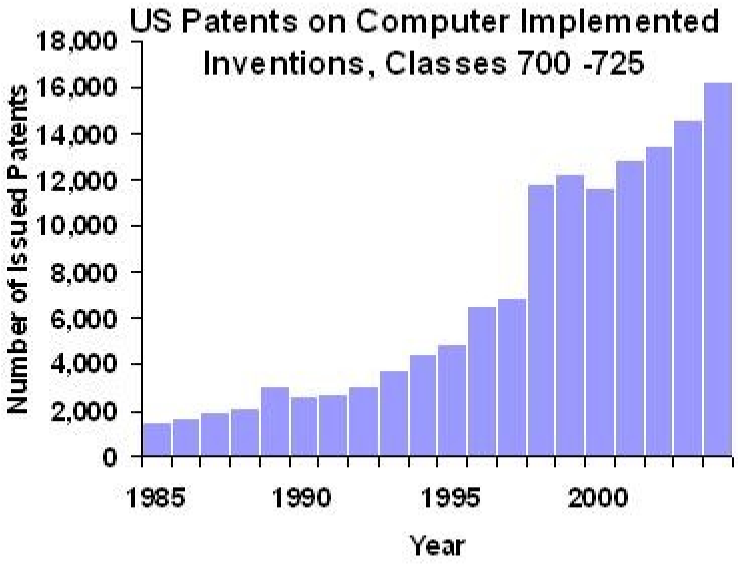
- Software patents in US include *virus detection, web interfaces, content licensing, video compression software and more*
- Some of the categories of USPTO Software patents
  - Computer speed, Computer security, Word processing, Spreadsheets
  - Business management systems, Graphics. Operating systems
- Kinds of recent Patents approved by US courts are: Networking technology, User interface, Machine control interface, Encryption, Databases,Internet search
- Not all software; but should meet certain standards
- USPTO created documents: 2014 Interim Guidance on Patent Matter Eligibility, 2015 Update: Subject Matter Eligibility
- Standards for software patents:
  - Is the software an abstract idea?
  - Does the process turn that abstract idea into an “inventive concept”?
- If the software is “abstract idea”, that will work on nonspecific computer, then it can’t be patented. Ex: Math formula, Algorithm, Economic Practice

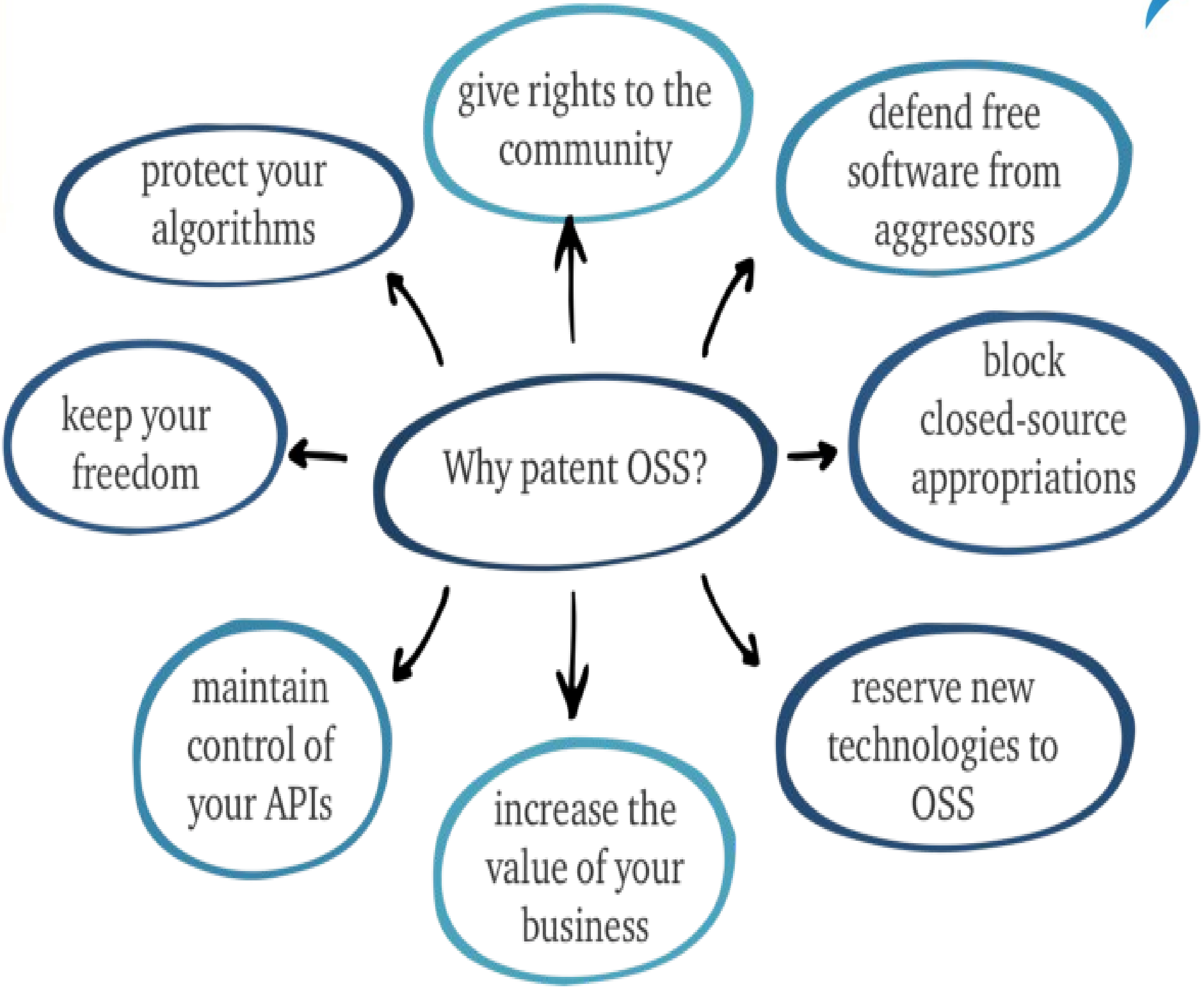
Patents - Criticism

- Many critics/criticisms on patents – several groups oppose / lobby
- Patents are always been criticized for being granted on already known innovations -1938 R B Fuller an inventor
- Criticized for conferring a “negative right” upon a patent owner, permitting them to exclude competitors from using or exploiting the invention, even the competitor subsequently develops the same invention independently
- Patents may hinder innovations as well
- Michael Heller & Rebecca Sue Eisenberg- theory of anticomons-IPR may become so fragmented that, effectively, no one can take advantage of them as to do so would require an agreement between the owners of all of the fragments
- In 1980’s, patent officers accepted computer program subject to patent law
- Economics of FOSS – OSI non—profit organization

Patents – history

- **History**
  - Some thing like patents was used in **Greek cities**
  - Creator of a new recipe was granted an exclusive right to make the food for one year and similar practice existed in some **Rome cities**
  - Modern stage patents originated in Italy in **1474**; England followed with the status of monopolies in 1623, for only “projects of new invention”, **US in 1790** (1st patent for making Potash
  - Economics of FOSS OSI dedicated to manage and promote the Open Source definition for the god of the community through OSI certified Open Source Software certification mark and the program
  - “Open source standard is more than just a specification. The principles behind standard and the **practice of offering and operating the standard**, are what make the standard Open”




Why patent OSS?
protect your algorithms
give rights to the community
defend free software from aggressors
keep your freedom
block closed-source appropriations
maintain control of your APIs
increase the value of your business
reserve new technologies to OSS

https://medium.com/novalto/why-patent-open-source-software-b17ca6eb89d9

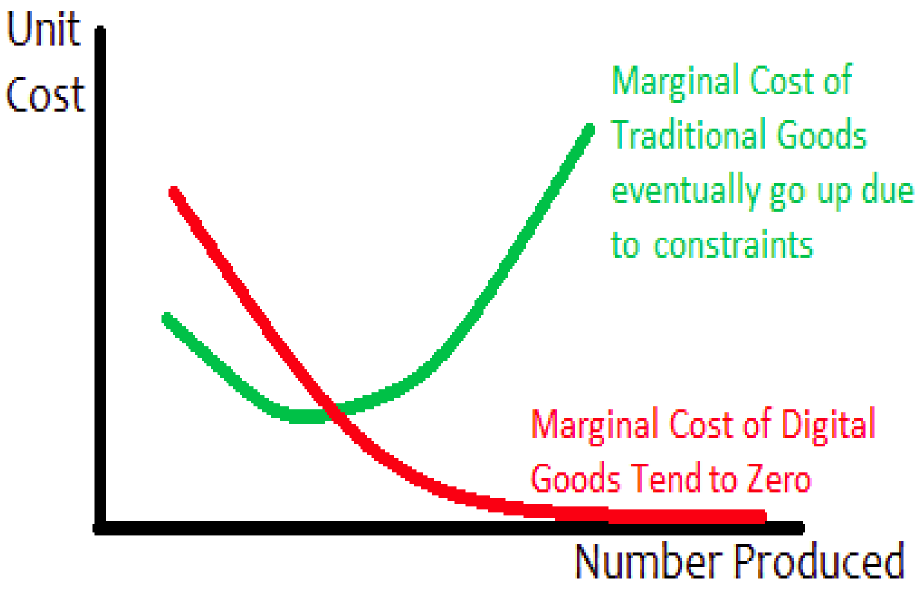
Zero marginal cost

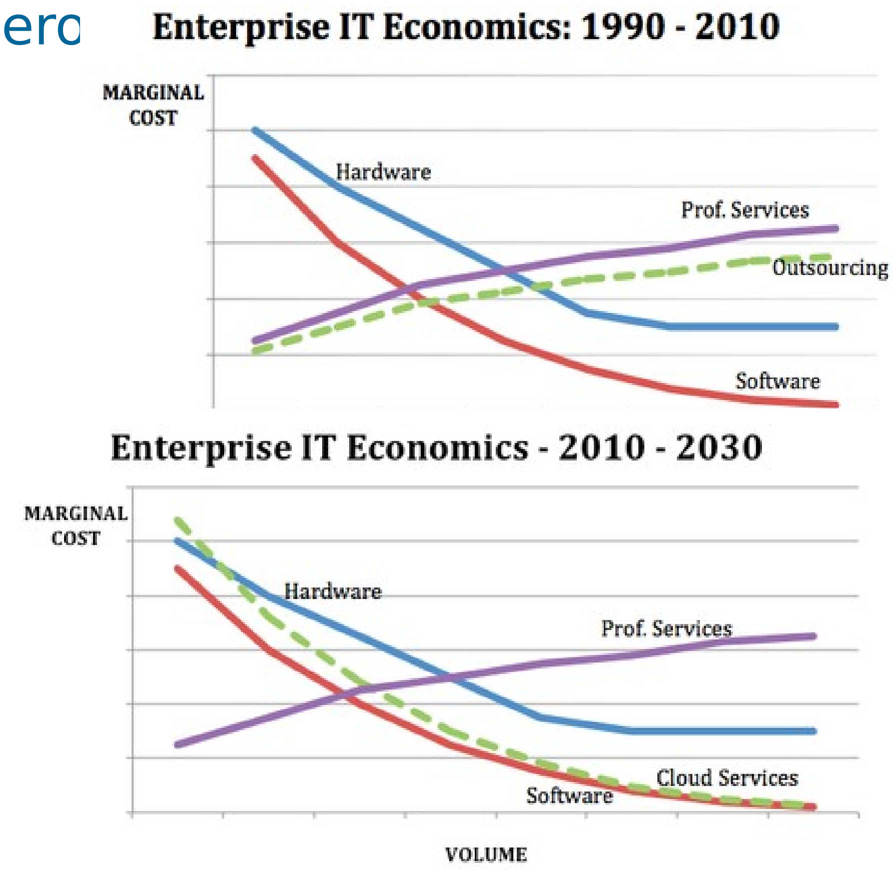
$$\text{Marginal Cost} = \frac{\text{Change in Total Cost}}{\text{Change in Quantity}}$$

- Units:
  - Marginal cost = the cost to produce the next unit
  - Fixed cost = costs that are not reducible in the short time(lease, factory cost)
  - Variable costs = costs that scale with production
- Marginal costs include both fixed and variable costs
  - Cost of software production is primarily fixed and won’t be changed with the scale. For example, $10 million to build MSOffice, there is not much additional cost to produce 1000th unit sold. Once break-even is crossed, then it will be profit
  - **Production of Jeans**: fixed costs(plant, equipment and some labor) and variable cost(*materials, some labor, storage, shipping, sales, marketing* etc.)

Zero marginal cost (cont.)

- The marginal cost of
  - providing support approaches to zero,
  - maintenance & improvements approaches to zero
  - infrastructure and hardware approaches to zero
- The marginal cost never become zerc





https://ckluis.com/the-marginal-cost-of-software-approaches-zero-7fda166f21

http://wikibon.org/wiki/v/Both_Buyers_and_Sellers_Must_Learn_to_Compete_in_the_Amazon_Economy

Zero Marginal Cost

- Economics of digital products are odd because once they are produced, the product cost of making copies is nearly zero
- Zero marginal costs- music, movies, newspapers, software
- The markets will drive prices to zero and so companies try to play tricks to make money even though the companies economics tend do push prices ever downward

Ways to deal Zero marginal cost (cont.)

- Give up/ecosystem
  - Give up and make the product free. Ex: GPS devices for cars -business; free for Google and Apple on phones to enhance the value to their larger smartphone ecosystem.; Jio -free vice calls and paid data
- Bundling -
  - Bundle products as a package with very low marginal cost items and sell them together to make money. Ex: Netflix
- Freemium/tiered pricing
  - Give away some core product free, then extra to hard core users for extras. Tiered pricing has been around a long times of course. Coupons are the classic example. NY Time is another example
- Minimal pricing
  - Music industry is doing this with 99 cents. Most people prefer pay minimal cost better than piracy
- Advertising
  - Classic, used by radio and TV for decades. Product is free, but you have to listen to adds. Newspapers used to make 80% of their money from adds, now it is fallen to 20%.

Income generation Opportunities

- Enthusiastic and generous developers spend their days and nights creating software for free.
- For a few decades, “open source software” was synonymous with “free software”
- Developing with open code through group effort excluded the opportunity for profit, but today there are ways to profit from building OSS
- Lately, developers have started thinking about how to make monetize their OSS
- Software as Service(OpenSaas), Paid Support, Dual Licensing, Pain extra-features or functionalities, paid Certification

https://rubygarage.org/blog/how-make-money-with-open-source-projects#article_title_0

Common ways to make profits from OSS

- Software as a Service (**OpenSaas**) model
  - When you have a full-fledged application of generating demand,
  - Popular way to license software because it’s flexible and rapid deployment and decreased costs
  - Cloud made this model attractive, where users need a web browser to access an application
  - This is popular business model for vendors that build tools for *HR, collaboration, content management, and project management*
  - SaaS solution with a free codebase reduce development costs and eliminate the need to build redundant functionality
  - Accordingly, a vibrant OSS community will gladly promote quality open source products With an OpneSaaS model, software is produced via subscripts, which can after varying levels of service. Ex: WordPress and Share tribe

Common ways to make profits from OSS (cont.)

- WordPress as OpenSaaS
  - Serves as a free SaaS product. It’s an open platform, and it offers subscriptions plans with extra features like unlimited storage, automatic backup, customization themes, and custom domain names
  - Can be installed on your own web server and maintain independently.
  - SaaS version of WordPress can be at wordpress.com. Since there is no vendor lock-in, a customer can easily switch to a self-hosted WordPress any time
- Sharetribe as OpenSaaS
  - A marketplace builder. It comes with two versions: a self-hosted free version found on GitHub and a cloud-based hosted version at sharetribe.com
  - Provide support for hosting,maintenance, backups, updations , domains, removal of Sharetribe branding, personalizing your marketplace filters and change colors, images, and block position in the UI
  - Open source version of Sharetribe deploy Sharetribe on server, run updates and backups

Common ways to make profits from OSS (cont.)

- WordPress as OpenSaaS
  - Serves as a free SaaS product. It’s an open platform, and it offers subscriptions plans with extra features like unlimited storage, automatic backup, customization themes, and custom domain names
  - Can be installed on your own web server and maintain independently.
  - SaaS version of WordPress can be at wordpress.com. Since there is no vendor lock-in, a customer can easily switch to a self-hosted WordPress any time
- Sharetribe as OpenSaaS
  - A marketplace builder. It comes with two versions: a self-hosted free version found on GitHub and a cloud-based hosted version at sharetribe.com
  - Provide support for hosting,maintenance, backups, updations , domains, removal of Sharetribe branding, personalizing your marketplace filters and change colors, images, and block position in the UI
  - Open source version of Sharetribe deploy Sharetribe on server, run updates and backups

Common ways to make profits from OSS (cont.)

- **Paid support**
  - Many OSS companies succeed by providing extra services like technical support, certifications, and training
  - Most professional open source companies including **RedHat, JBoss**, and **MySQL**- have their entire business by providing free solutions. These companies generate profits only from additional services which are paid
  - RedHat earned $2.05 billion in 2016 with RedHat Enterprise Linux and also sells yearly subscriptions for user and technical support
  - Customers can choose a subscription plan based on number of requests, severity of of requests, support channels and hours of coverage.
  - MySQL, derives revenue from selling support subscriptions for their product
  - Paid support is an effective tool for making profit from open source for few reasons
  - Enterprise owners can save money on their payroll. Instead of hiring in-house specialists, enterprise owners can have access to certified support specialists on a less expensive basis
  - Enterprise can have peace of mind knowing that they can cal in the pros whenever a problem arises

Few common ways to make profits from OSS (cont.)

- **Dual licensing**
  - Dual licensing allows to release commercial software (commercial license) derived from free OSS commonly distributed under the GNU GPL license
  - Dual licensing can be implemented two ways. The company
    1. Releases identical product under a commercial license and under a free license
    2. Release different versions under different licenses
  - Why do we need dual license?
    - GPL allows the users to run OSS, redistribute that software and modify it, but you can’t embed OSS solutions in proprietary (commercial) software and make profit under a GPL license. So, commercial version of open source product to sell the commercial software
    - Ex: MySQL releases MySQL Enterprise Edition, MySQL Cluster COE, and MySQL Standard Edition under commercial licenses while still offering two other products – MySQLClassic Edition, MySQL Community Edition
    - Commercial versions receive enhanced security features, backups, 24x7 support and more

Few common ways to make profits from OSS (cont.)

- **Paid extra features or functionaries**
  - Companies can make money with open source by distributing their software free, but charge money for additional features, functionalities or updates
  - Customers feel most comfortable paying for only the services they utilize. Charging money extra functionality is attractive for a number of reasons
  - It’s clear what extra functionality customers are paying for
    - Customers can save money on deployment and troubleshooting when these services are included in paid packages
    - Ex: GitLab distributes their developer tools in 3 editions with premium support, file locking and advanced solutions for remote teams and is billed per user
  - Charging money for additional functionality works well for software designed for blogging, HR, collaboration, customer management, and other similar needs
  - WordPress provides Premium and Enterprise customers, for insurance, with additional websites and an additional workforce to maintain their Wordpress sites. Customers receive certified professionals assigned to specific technologies and business, but charge $5K per month

Few common ways to make profits from OSS (cont.)

- **Paid certification**
  - If software is popular enough, companies can offer certification opportunities for specialists who want to validate their knowledge and skills
  - Getting certified, as a developer, is quiet useful for a number of reasons
  - It’s a great opportunity to differentiate yourself among other specialists with the same skills
  - Developers realize the importance of networking with mentors and group-mates
  - A developer’s certification leads lends them additional professional credibility and even promotions, and can boost a company’s image
  - Open source giants including Magento and RedHat offer a variety of certification options
  - RedHat offers around 30 different kinds of certifications in RedHat’s products as well as PaaS, business processes, deployment and system management.
  - Magento issue certificates for skills related to Magento development and solutions

Success of Open-source Projects

- Open source software is highly regarded to its security, realiability, and vibrant communities support
- Moreover, OSS helps companies avoid reinventing the wheel and can even be used within proprietary software
- People are willing to pay for open source software to ensure legal protection, technology support, and professional services
- With OSS companies can offer the best of both worlds, transparent technology with thee support and features of commercial software

Internationalization

- OSS allows for extensible internationalization
- OSS can be modified for any language
- Many FOSS packages are available for several languages
- Adapting software for their native languages nations, cultures, regions etc.
- Localization- process of adapting software for a specific region or language by adding specific components and translating text

Focal points of internationalization

- Language – computer encode text
- alphabet/scripts – keyboards, L-R, R-L
- Spelling – multiple dictionaries; audio-video
- Date/time format including calendar
- Formatting number (decimal points, text etc.
- Time zones (UTS)
- Currency
- Images and colors (comprehensibility & cultural appropriateness)
- Names and titles
- Telephone numbers
- Weights and measures
- Paper sizes

Focal points of internationalization

- Regulatory compliance
- Language translation
- National varieties of languages
- Special support for certain languages
- Local customs
- Local content
- Symbols
- Order of sorting
- Aesthetics
- Cultural values an social context
- Methods – UNICODE
- Rationalization

Example License

- G-Suite: https://gsuite.google.com/terms/2013/1/premier_terms.html?_ga=2.111247565.878899665.1599894661-2128516781.1599894661
- https://gsuite.google.com/intl/en/terms/additional_services.html
-

References

- https://www.gnu.org/philosophy/free-sw.html
- https://opensource.com/open-source-way
- https://www.tricentis.com/
- https://praxtime.com/2013/01/06/digital-economics-the-zero-marginal-cost-economy/
- https://smallbusiness.findlaw.com/intellectual-property/types-of-patents.html