OPEN SOURCE SOFTWARE

CHAPTER 1
INTRODUCTION TO OPEN-SOURCE

- Open Source, Need and Principles of OSS,
- Open-Source Standards, Requirements for Software,
- Free Software, Examples,
- Licensing, Free Vs. Proprietary Software,
- Free Software Vs. Open-Source Software,
- Public Domain.
- History of free software,
- Proprietary Vs Open-Source Licensing Model,
- use of Open- Source Software,
- FOSS does not mean no cost.
- History: BSD
- The Free Software Foundation and the GNU Project.

Open source software (OSS)

Open source software (OSS) refers to software that features freely available source code, which users may view, modify, adopt, and share for both commercial and noncommercial purposes.

Key principles of open source software development

Guided by the key principles of **transparency**, **collaboration**, and **decentralization**, the open source software model creates code from the community—and community from the code.

1.Collaboration

Unlike closed source software, OSS is not only available for anyone to use, but also to build on. This has resulted in a global network of contributors who work together on a project by collectively reviewing, testing, documenting, and patching code.

2.Transparency and security

Open source software development is public, which means all of the work, including the codebase and communication among users, is available for the **open source community** to see. Transparency instills trust among contributors, paving the way for innovation and play. And since the codebase is public, users are able to quickly identify and fix security vulnerabilities as they arise.

3.Decentralization

Since the community develops the code—and since no one person or company owns that code—open source is an inherently decentralized form of software development that comes with fewer silos, bottlenecks, and barriers to entry.

What are 'standards'?

ISO, the International Organization for Standardization, defines *standards* as "a document that provides requirements, specifications, guidelines or characteristics that can be used consistently to ensure that materials, products, processes and services are fit for their purpose."

Open standards example: The Internet

- ✓ In its explanation of open Internet standards, The Internet Society (a global organization that helps drive Internet policy and technology standards) says, "The Internet is allowing devices, services, and applications to work together across a wide and dispersed network of networks.

- ✓ " The page lists The Internet Engineering Task Force (IETF), The Internet Research Task Force (IRTF), and The Internet Architecture Board (IAB) as the core groups behind the development of the open Internet standards.

- ✓ "These organizations are all open, transparent, and rely on a bottom-up consensus-building process to develop standards. They help make sure open standards have freely accessible specifications, are unencumbered, have open development and are continuously evolving," the page explains.

Requirements for Software

1No Intentional Secrets: The standard MUST NOT withhold any detail necessary for interoperable implementation. As flaws are inevitable, the standard MUST define a process for fixing flaws identified during implementation and interoperability testing and to incorporate said changes into a revised version or superseding version of the standard to be released under terms that do not violate the OSR.

2**Availability:** The standard MUST be freely and publicly available (e.g., from a stable web site) under royalty-free terms at reasonable and non-discriminatory cost.

3.**Patents:** All patents essential to implementation of the standard MUST:

1. be licensed under royalty-free terms for unrestricted use, or
2. be covered by a promise of non-assertion when practiced by open source software

4.No Agreements: There MUST NOT be any requirement for execution of a license agreement, NDA, grant, click-through, or any other form of paperwork to deploy conforming implementations of the standard.

5.**No OSR-Incompatible Dependencies:** Implementation of the standard MUST NOT require any other technology that fails to meet the criteria of this Requirement.

EXAMPLES

Artificial intelligence[edit]

General AI[edit]

•OpenCog – A project that aims to build an artificial general intelligence (AGI) framework. OpenCog Prime is a specific set of interacting components designed to give rise to human-equivalent artificial general intelligence.

Computer vision[edit]

•AForge.NET – computer vision, artificial intelligence and robotics library for the .NET Framework

•OpenCV – computer vision library in C++

Machine learning[edit]

•See List of open-source machine learning software

•See Data Mining below

•See R programming language – packages of statistical learning and analysis tools

Planning[edit]

•TREX – Reactive planning

Robotics[edit]

•Robot Operating System (ROS)

•Webots – Robot Simulator

•YARP – Yet Another Robot Platform

1.Linux operating system: Linux is a widely supported operating system. It's common in computers, mainframes, smartphones, servers and embedded devices.

1.Apache web server application: Apache is a web server platform that's free and maintained by an open community of developers. The server stores data, processes requests and serves web assets in an easily queried format.

1.VCL media player: The VCL media player is a portable, open-source media player and streaming server. The program is compatible with mobile platforms and with various operating systems.

S/W LICENCE

A *software license* is a legal agreement that defines how a given piece of software can be used.

Defining Copyright and Licensing

In the U.S. and many countries, there are certain legal protections you are automatically granted for any creative work you produce, one of those being *copyright*. The U.S. Copyright Office defines copyright as “a type of intellectual property that protects original works of authorship,” specifically when the “author fixes the work in a tangible form of expression.” This means with copyright you are not the owner of the idea, but rather the material expression of the idea. If a copyright owner desires stricter legal protection over their work, this can be achieved through patents, trademarks, and intellectual property law

Proprietary Software, Free Software, and Open-Source Software

- *Proprietary software* is any software with a license that restricts how it can be used, modified, or shared. Video games are a common example of proprietary software.
- If you purchase a video game (whether as a cartridge, disc, or digital download), you aren't allowed to make a copy of that game to share with friends or sell for profit.
- It's also likely you aren't permitted to modify the game's code to run it on a different platform than the one you originally bought it for.

The Differences between Proprietary and Open Source Software

Open Software (Linux Ubuntu, OpenOffice.org Write, GIMP)	Proprietary Software (Windows Vista, Microsoft Word 2007, Adobe Photoshop CS3)
S Purchased **with** its source code	s Purchased **without** its source code
S User can get open software for **free** of charge	s User **must pay** to get the proprietary software
n Users **can** modify the software	w Users **cannot** modify the software
n Users can install software freely into any computer	w User must **have a license** from vendor before install into computer
c No one is responsible to the software	s Full support from vendor if anything happened to the software

S.No.	FS Philosophy	OSS Philosophy
1.	It was coined by the Free Software Foundation in the 1980s.	In response to the restrictions of free software, the phrase “open source” was coined in the late 1990s.
2.	Software is an important part of people’s lives.	Software is just software. There are no ethics associated directly with it.
3.	Software freedom translates to social freedom.	Ethics are to be associated with the people not with the software.
4.	Freedom is a value that is more important than any economical advantage.	Freedom is not an absolute concept. Freedom should be allowed, not imposed.
5.	Every free software is open source.	Every open-source software is not free software.

Public Domain

Public-domain software is a type of software that you can use without the permission of or payment to the author. Such software has no legal, copyright, or editing restrictions, so no one owns the right to restrict or control its use. A few examples of public-domain software are SQLite, I2P, and SHA-3.

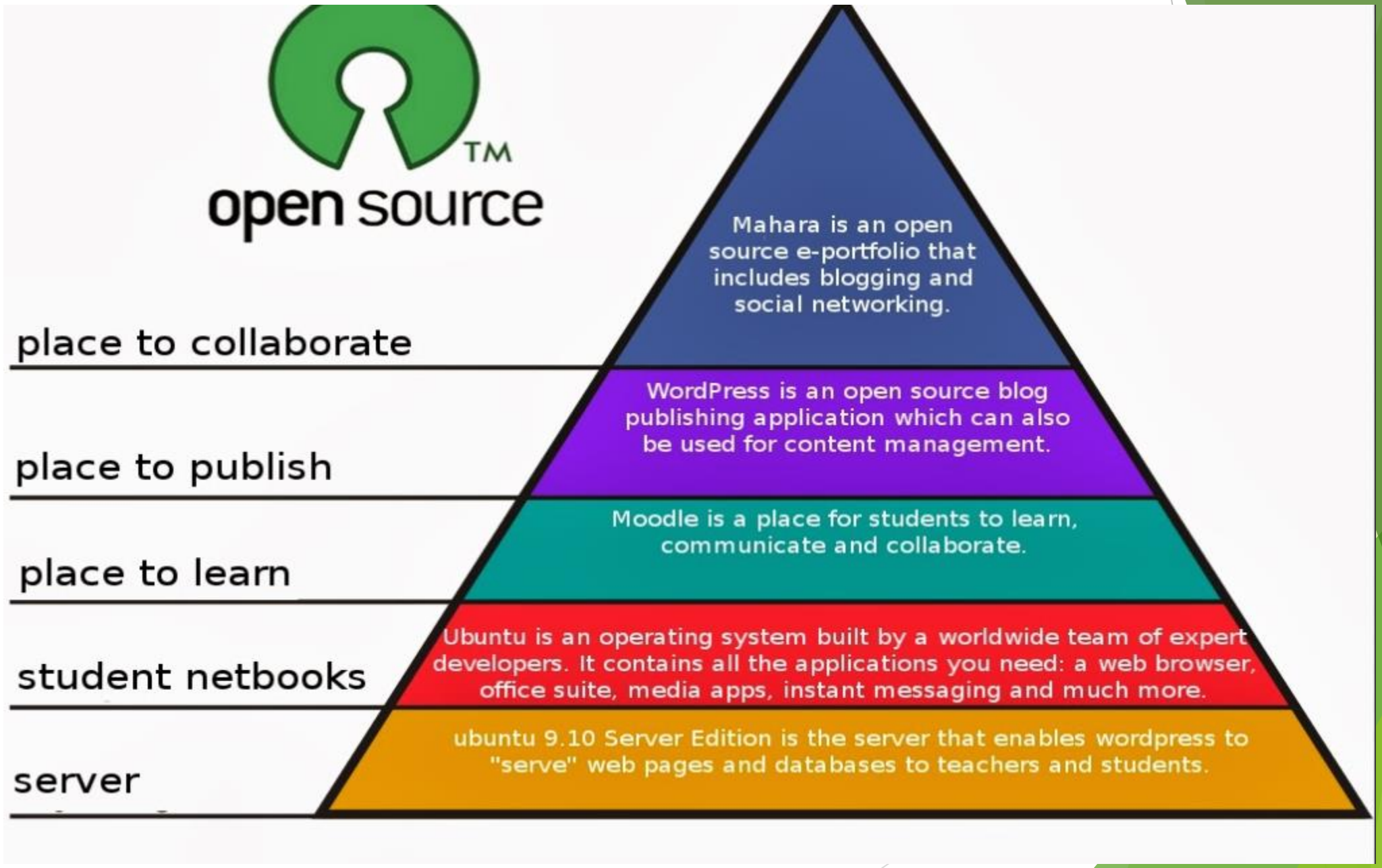
PROPRIRTARY MODEL

- ✓ The traditional licensing model of proprietary software is inherently based on the implementation of certain restrictions. These restrictions could be manifold.
- ✓ They could apply to the number of end users that can use the software, the number of processors or processor cores which can run the software or the number of permitted installations.
- ✓ Under the proprietary licensing model, if the end user would want to increase this fixed capacity that is being served by the software the only solution would be to purchase additional licenses.
- ✓ the customer has no access whatsoever to the source code.

OSS MODEL

- ✓ the open source software (OSS) licensing model presents some profound differences as it is based on certain basic freedoms that are transmitted to the end user together with the software
- ✓ 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).

USE OF OSS


open source
Mahara is an open source e-portfolio that includes blogging and social networking.
place to collaborate
WordPress is an open source blog publishing application which can also be used for content management.
place to publish
Moodle is a place for students to learn, communicate and collaborate.
place to learn
Ubuntu is an operating system built by a worldwide team of expert developers. It contains all the applications you need: a web browser, office suite, media apps, instant messaging and much more.
student netbooks
ubuntu 9.10 Server Edition is the server that enables wordpress to "serve" web pages and databases to teachers and students.
server

What is FOSS?

FOSS means Free and Open Source Software. It doesn’t mean the software is free of cost. It means that the software's source code is open for all and anyone is free to use, study and modify the code. This principle allows others to contribute to developing and improving a software like a community.

- ✓ Any user should be able to run the software for any purpose.
- ✓ Users should be free to see the source code of the software and if need be, users should be allowed to modify the code as well.
- ✓ Users should be free to distribute copies of the software to others.
- ✓ If a user modifies the code, she/he should be free to distribute the modified code to others. The modified code must have the source code open.

HISTORY OF BSD

- ✓ FreeBSD, a free and open-source Unix-like operating system has been around since 1993. However, its origins are directly linked to that of BSD, and further back, those of Unix. During this History of FreeBSD series, we will talk about how Unix came to be, and how Berkeley's Unix developed at Bell Labs.

- ✓ **Before FreeBSD, there was Unix**
- ✓ The story of Unix starts back in the mid-1960s with Multics. MIT, AT&T Bell Labs, and GE started jointly developing Multics as an experimental operating system for the GE-645 mainframe.

- ✓ Multics had many new ideas that paved the way for all modern Operating Systems. Some of those ideas are still in use today such as, dynamic linking, a hierarchical file system, and memory mapped files. In 1969, Bell Labs pulled out of the project due to dissatisfaction with the projects progress.

- ✓ Later in 1992 386BSD 0.1 was released and set the stage for the formation of FreeBSD and NetBSD. A group of users began releasing an unofficial patchkit, by collecting bug fixes and enhancements. This group disagreed with the future direction and release schedule of 386BSD, so in 1993 they founded The FreeBSD Project.

.....THANK YOU.....