

Linux Applications and Software Licensing

Linux System Administration
COMP2018 Fall 2019

Linux Roles and Applications



Server vs. Desktop Roles

- Computers perform tasks for us
- The types of tasks they perform are determined by their roles; these determine how they are installed, configured, and maintained as well as what software they require
- Servers are unattended machines which provide resources to other machines using server applications
- Workstations, or desktops, are machines directly used by people to run desktop applications; those applications may or may not use services from servers

Desktops

- Desktop machines usually have a graphical interface and software designed to produce results for their users
- Office users typically use office productivity tools such as email, messaging, word processing, spreadsheets, and multimedia programs - they also use programs specific to their work, whether that is accounting software, legal document preparation, record-keeping, authoring books, magazines, or news articles, etc.
- Developers may use the same tools as office users, but also use programming tools including editors, compilers, debuggers, etc.
- Artists may add audio, video, or other graphic or 3d output programs and devices useful in their endeavours

General Purpose User Applications

- Most of the general purpose applications for users are part of the desktop environment, including
 - Menu system for launching applications and controlling them
 - Web Browsing (Firefox)
 - File management (Varies with desktop environment)
 - Word processing and simple text file handling (LibreOffice writer, gedit/atom/nano/vim/etc.)
 - Image viewing and manipulation (Image Viewer, Image Magick, Gimp, Various video)
 - Email, messaging (Thunderbird, Evolution)
 - Spreadsheets, presentations (LibreOffice calc, impress)
 - Calendar, contact manager, calculator, terminal, etc.
- A very large number of additional applications are available for specific types of users and are usually installed by end users using the graphical Software Center application

Servers

- Servers rarely support graphical user interfaces and instead provide resources using network protocols
- Servers may perform one or more of the following functions
 - be repositories for files accessed using file sharing protocols
 - offer web-based access to static or dynamic content
 - host email and messaging services
 - host databases, possibly with dataset manipulation and analysis
 - coordinate and provide printing or physical control services
 - provide infrastructure services including monitoring services
 - provide custom application services (e.g. twitter, netflix, gaming, etc.)
- Administration of servers is typically done remotely using a terminal emulator for command line access or web-based interfaces

File Sharing

- Various server protocols are supported, including Microsoft, Apple, and UNIX/Linux
- Microsoft, Apple, UNIX/Linux clients are included
- File shares can be accessed dynamically or statically configured to be always attached
- Servers provide access control and security for file sharing

Web Service

- Servers can provide static or dynamic content
- Many server-side technologies for dynamic content are supported
- Apache and Nginx are the most common web server software suites, both are enormously expandable and they are not mutually exclusive
- Linux provides a highly scalable platform for web services with support for every major open source programming language and toolset

Email and Messaging

- Linux support for email includes the sendmail, postfix, and exim MTA packages, and the dovecot and cyrus MDA packages
- Many other lesser known options are available but these are the major ones
- POP3 and IMAP4 as well as the SSL protected versions of them are supported, as well as direct mail spool access locally or via file sharing
- Facebook messenger, twitter, skype, telegram, pidgin, kopete, psi, etc. provide IM clients
- ejabberd, prosody, Openfire, etc. provide IM servers

Databases

- For the SQL crowd, MySQL, MariaDB, and PostgreSQL are ubiquitous
- Other options include MongoDB, SQLite, LucidDB, Firebird, Ingres, Drizzle, etc.
- Databases are typically used as backend services for other programs and service-providing software
- Databases can be accessed locally by applications running on the host holding the database, but can also be set up for network access, with application running on client systems

Printing and other devices

- Linux supports many print devices both traditional and 3d
- Linux also supports scanner and multifunction devices
- Surveillance systems are supported as Linux supports cameras, both attached and over the network
- Digitizing pads and other uncommon input devices are supported by Linux

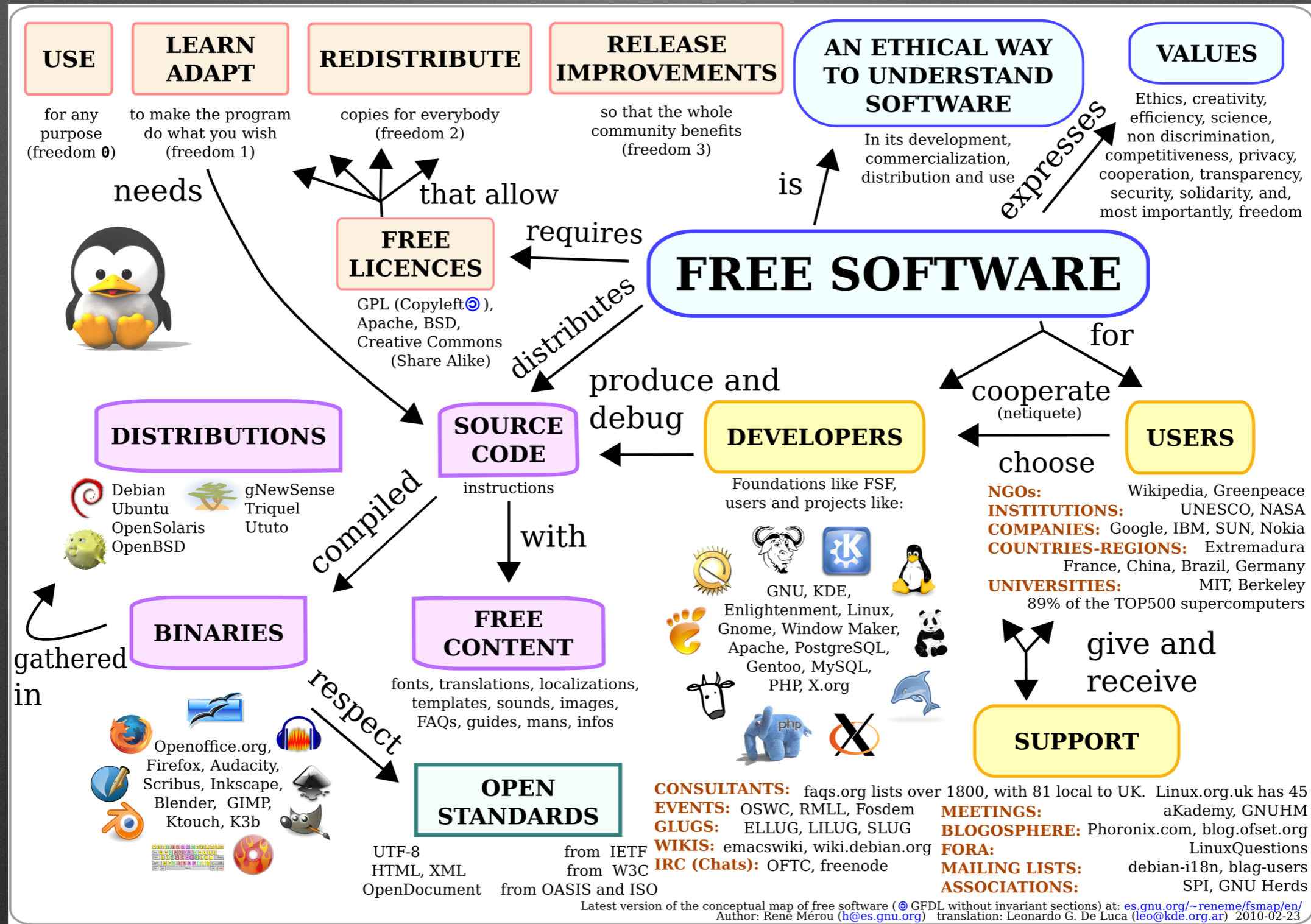
Infrastructure Services

- Linux and UNIX systems have provided the core of the internet for many years
- DHCP, DNS, LDAP, routing, whois, and many other infrastructure services are reliably provided using Linux servers
- More sophisticated infrastructure devices such as IPS, IDS, UTMs, and proxy servers are also often built on Linux

Server Administration Interfaces

- Servers are traditionally administered using a shell accessed with a terminal or pseudo-terminal, typically remotely over the network
- Some server applications provide a web interface for administration
- Software packages like cockpit and webmin provide an integrated web interface for managing an entire system
- There are also underlying remote administration tools and protocols such as SNMP to implement monitoring and control of large numbers of systems in an enterprise

Software Licensing



Software Licensing

- When software is created, it is considered in most countries to be owned by the creator (copyright applies)
- The creator then decides whether to keep it private and secret, or allow others to use it and what the rules are for that usage
- Putting software into the public domain means relinquishing control over the software and allowing anyone to do anything with it, with or without the creator's awareness or permission
- Any other means of permitting use is known as giving the user license to use the software and a contract is written to describe how it can be used

Open Source Licensing

- Open source software is provided with the source code included
- Open source does not mean free of charge, nor does it mean the user can do whatever they want with the source - it just means they can read it and use it to understand how the software works
- Several common contracts are used to expand on this license

Free Software Foundation

- Richard Stallman created the FSF in 1985 to promote the concept of free software, meaning free of charge, not public domain
- He created the GNU Public License (GPL) , GPLv2 and GPLv3, a viral license that requires you to apply the GPL to any derived works you create
- The GPL allows you to read source code, GPLv2 and GPLv3 allow you to modify source code, and redistribute source code as long as you do not claim credit for the original code, or charge more than nominal fees for duplication and distribution
- It has a copyleft provision that requires you to contribute any modifications made for other than purely private use back to the author so it can be integrated into the original source if the author so chooses
- GPLv3 prevents "tivoization", the use of proprietary hardware to circumvent GPLv2 clauses, LGPL (lesser GPL) allows linking to non-GPL libraries, AGPL (Affero GPL) adds restrictions on distributing code over networks
- Some commercial organizations claimed the FSF was too political and extreme

Open Software Initiative

- OSI was formed in 1998 to review licensing and endorse licenses as their version of "free"
- OSI endorsed licenses must allow software to be modified, redistributed, and used for any purpose whatsoever without requiring anything in return to the originator of the software
- OSI endorsed licenses allow a company to take open source software and modify it, then use it in proprietary products and keep it secret
- FSF licenses are OSI endorsed, not all OSI endorsed licenses are FSF approved
- The difference is primarily a philosophical difference, FSF does not approve of modifying open source and then placing restrictions on it, OSI is fine with that

Creative Commons

- A licensing approach for creative and written works
- Several types of licenses are available
 - Attribution - must acknowledge the author
 - Sharealike - similar to GPL with copyleft
 - No-Derivs - cannot change the content
 - Non-Commercial - no commercial use
 - No Rights Reserved - public domain
- Combinations are allowed (CC-A, CC-ND, CC-NC)

Making Money With Open Source Software

- Selling adjunct services such as installation, support, consulting is the primary way to earn an income with open source software
- Some licenses permit you to have proprietary components which you can sell
- Use open source software to perform paid services, think ISP or cloud services provider
- Get paid by interested parties or sponsors to work on open source projects
- Build backend services for open source client software which are paid or subscription-based or use proprietary paid data
- Add open source work to your CV as a way to do public service and get credit for it, build your reputation or personal brand on it
- Sell training or speaking on open source software or topics

Lab 2

Applications

Identify, Install, and Run Applications

Software can be chaotic, but we make it work



Expert

Trying Stuff
Until it Works

O RLY?

The Practical Developer
@ThePracticalDev