



Chapter 01

Open Source Software

Open Source SW Development
CSE22300

Proprietary Software

Proprietary Software



iOS



Windows 10

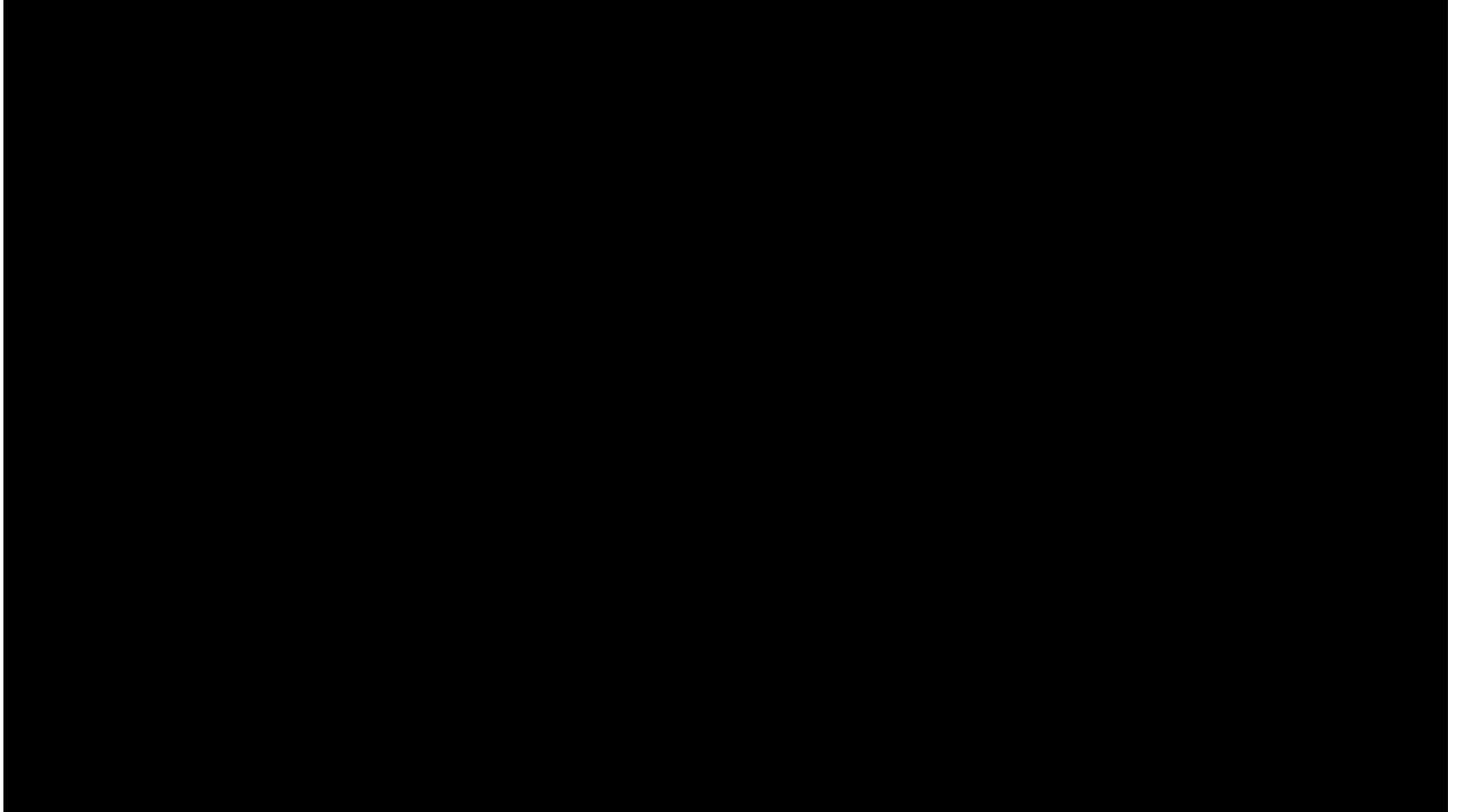


Microsoft Office

Definition

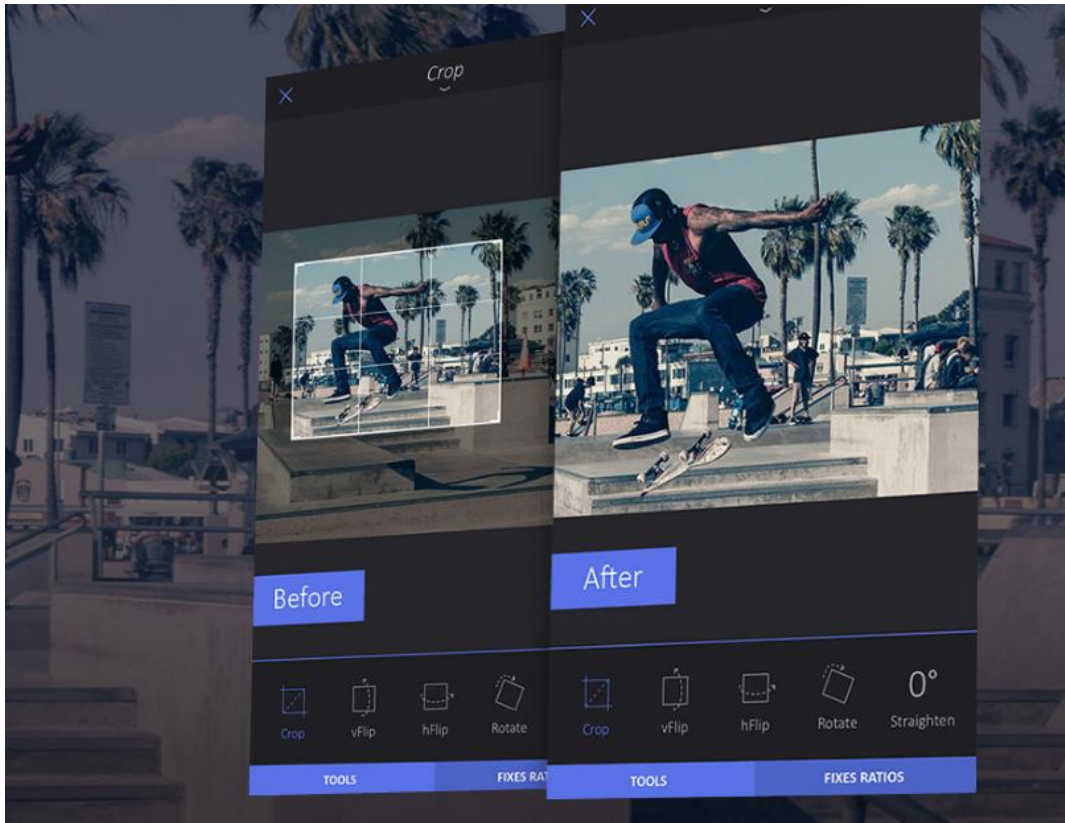
- **Definition**
 - Computer software licensed under exclusive legal right of the copyright holder
 - Restricted from other uses, such as modification, further distribution, or reverse engineering
- **Pros**
 - Consistent feature development
 - Stability
 - Responsibility for bugs, errors and updates
- **Cons**
 - High start-up costs
 - Dependency
 - Single company releasing patches

Enlight



Enlight

- How can we run Enlightenment on Android or other platforms?



Enlight : “all-in-one” phone app runs on iOS exclusively

Porting to Other Platform

- **iOS Emulator**



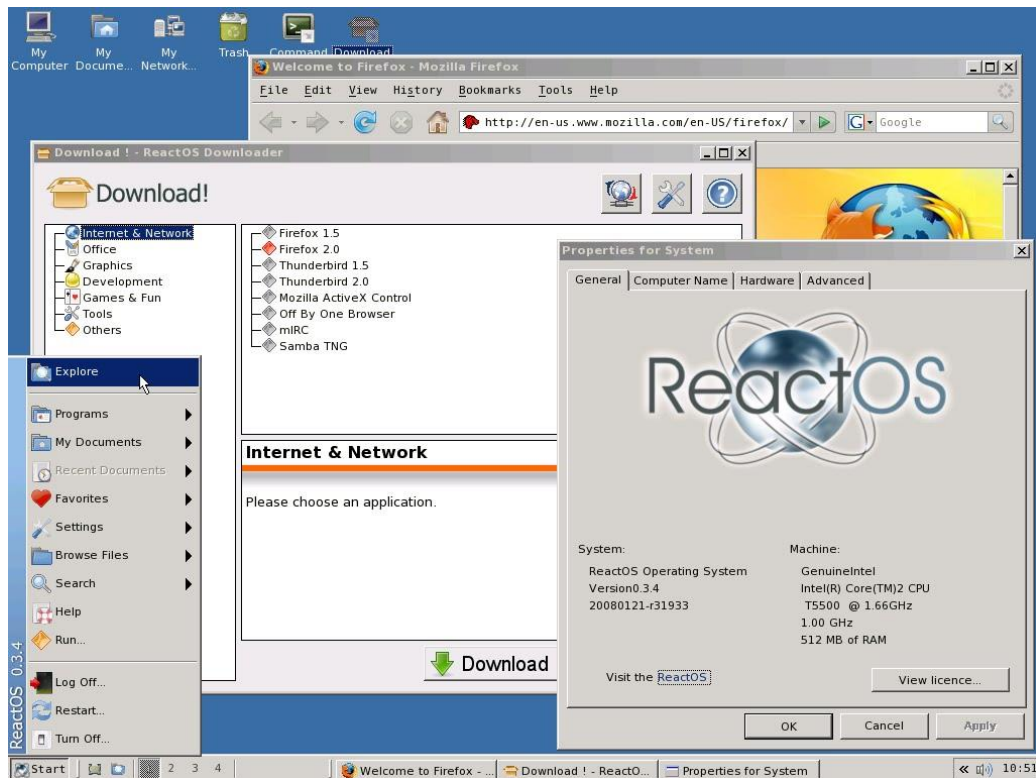
- Too slow, not working perfectly

- **Porting to Android**

- We need source codes!!!
- Source code of proprietary software is closed

React OS

- **Binary-compatible**
- **Start as Window 95 clone project in 1998**
- **Support A few software made for windows**



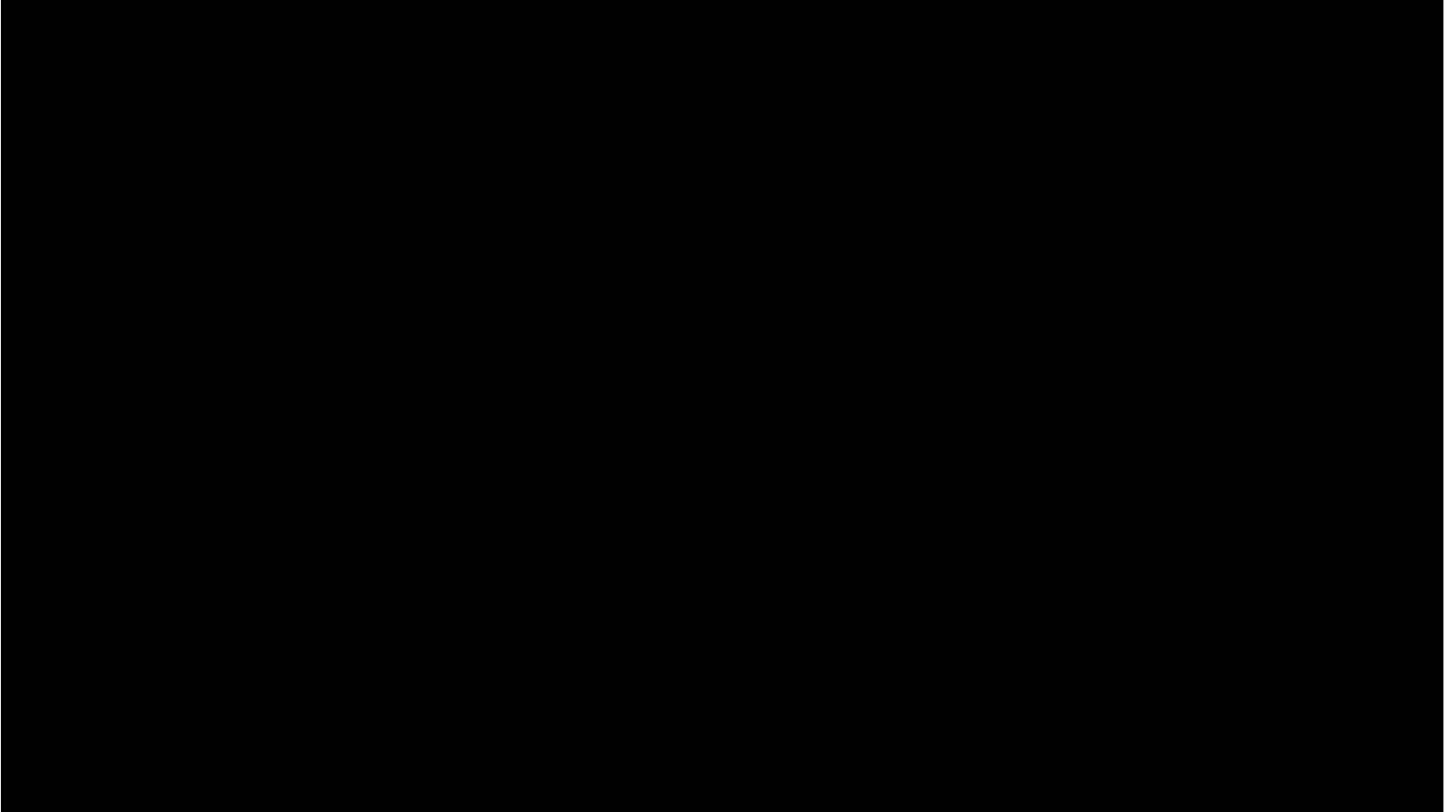
TmaxOS

- Developed by Tmax
- Compatible layer for MS Windows, Android and iOS



- Is it possible?

TmaxOS

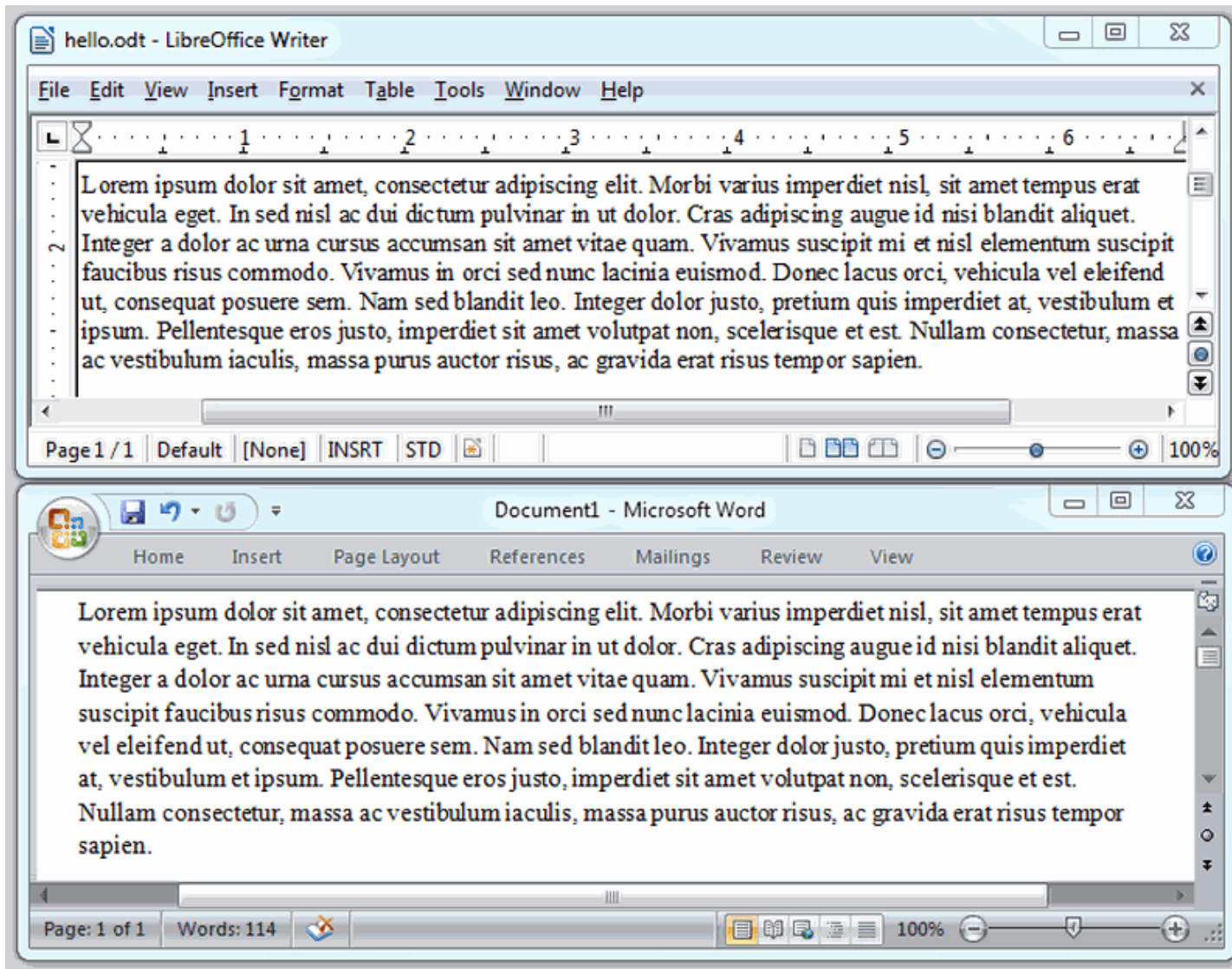


Proprietary Format

- **Used by individual programs**
- **Unique : cannot be read by other programs**
- **Huge barrier to rival**
- **Example**
 - **DWG : AutoCAD drawing**
 - **PSD : Adobe Photoshop's native image format**
- **Ownership of created content**



Word VS Writer



Open Source Software

Open Source Software



Open Source

- **Open Source Software**
 - Source code is available to the public
 - Anyone to copy, modify and redistribute the source code
 - GNU/Linux, Eclipse, Apache, Mozillar
- **Pros**
 - Low cost
 - Flexibility
 - Reliability and quality
 - No dependency
- **Cons**
 - Lack of personalized support
 - Restricted choice
 - Speed of change
 - No Warranty

Term

- **Free Software Movement**
 - Freedom to change the source code to whatever you need to do
 - Social movement
 - Negative image to business
- **“Open Source” label**
 - Created at a strategy session on Feb. 3rd, 1998 in Palo Alto
 - In reaction to Netscape’s source code release
 - By Tim O’Reilly, Eric Raymond and Larry Augustin
 - Methodology
 - Open Source Definition

Open Source Initiative

- **Open Source Initiative**
 - Organization dedicated to promoting open source software
 - Founded in Feb. 1998 by Bruce Perens and Eric Raymond
- **Open Source Definition**
 - Document published by the Open Source Initiative
 - Based on the Debian Free Software Guidelines
 - Written by Bruce Perens
 - Does not mean access to the source code
 - Redistribution!



Open Source Definition

- **Free Redistribution**
 - License shall not require a royalty or other fee for such sale
- **Source Code**
 - Must include source code & allow distribution
- **Derived works**
 - Must allow modification & allow them to be distributed



Replicant

A fully free
Android distribution

Open Source Definition

- **Integrity of Author's Source Code**
 - License must permit distribution of software built from modified source code
 - License may require derived works to carry a different name or version number from the original software
 - LibreOffice is forked from OpenOffice



Open Source Definition

- **No discrimination against**
 - **Persons, groups**
- **Distribution of license**
 - **Rights to program must apply to all without the need for execution of additional license**
- **License must not be specific to a product**
 - **The rights attached to a program must not depend on the program's being part of a particular software distribution**

Open Source Definition

- **License must not restrict other software**
 - The license must not place restrictions on other software that is distributed along with the licensed software
- **License must be technology-neutral**
 - No provision of the license may be predicated on any individual technology or style of interface

The Cathedral and the Bazaar

The Cathedral and The Bazaar



Sagrada Família



The Grand Bazaar

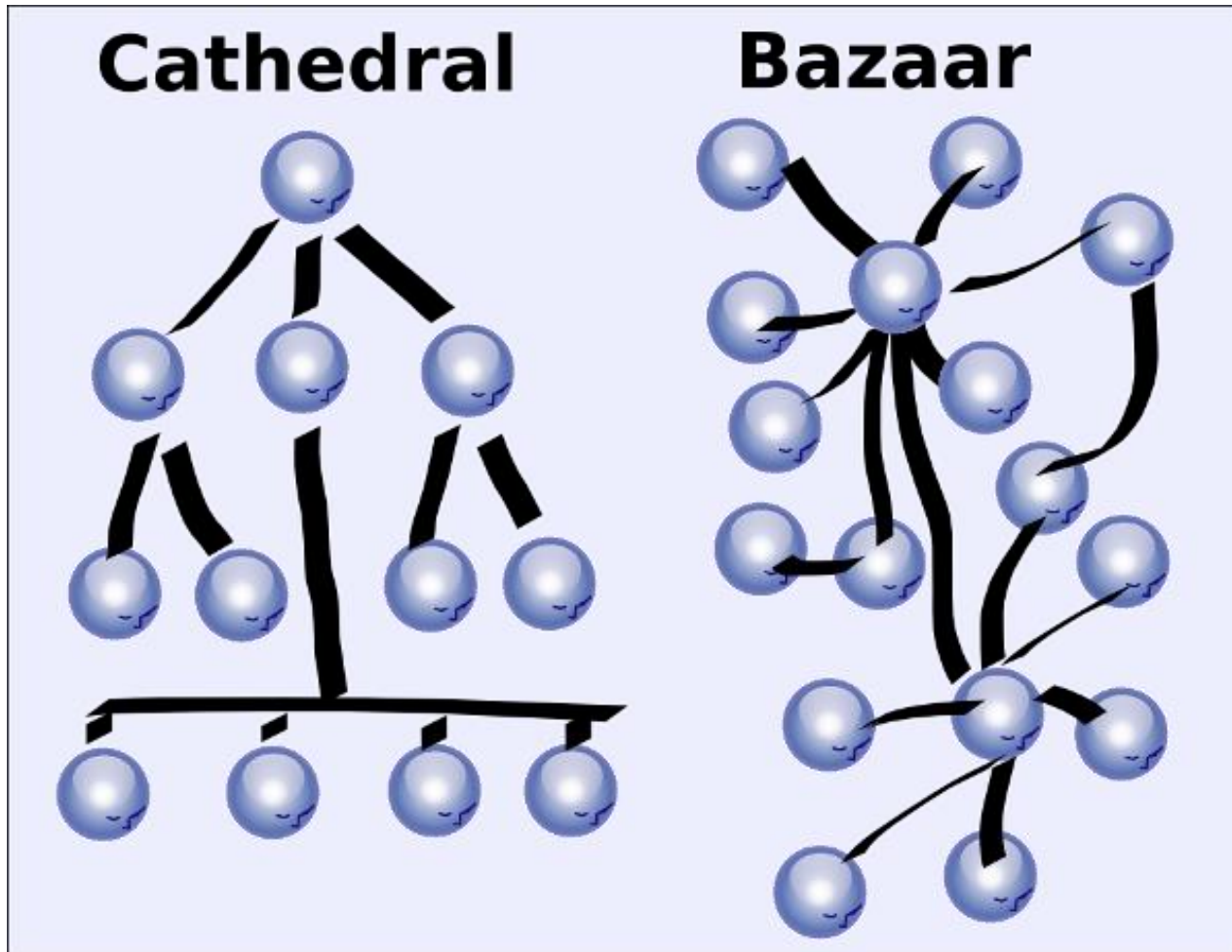
Cathedral Approach

- **Top-down**
- **Command-and-control**
- **Source code only available after release**
- **Leads to large complex programs**
 - IBM, Microsoft
- **Well-organized, full time development team**
- **Followed by corporate teams**
- **Worked on by teams of “high-priests/cathedral builders”**
- **Release far between (not until it’s “perfect”)**

Bazaar Approach

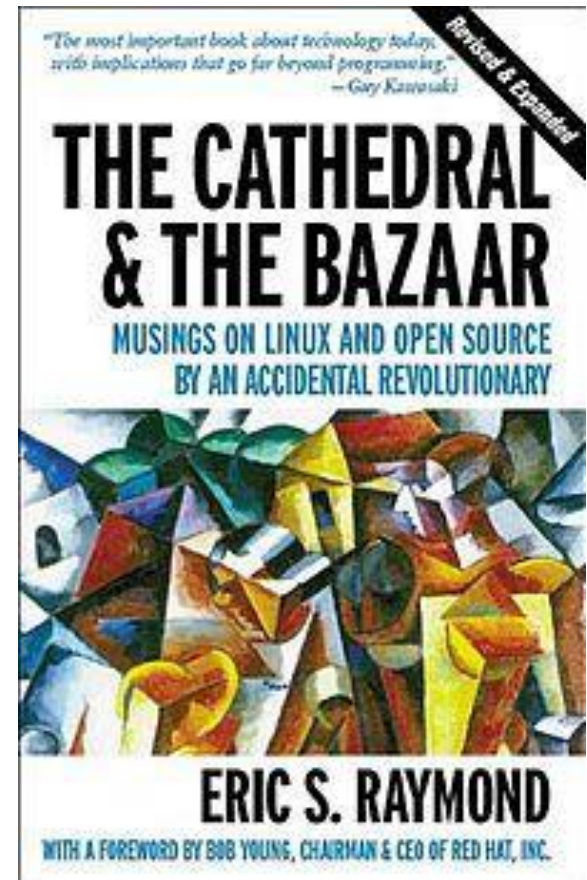
- **Bottom-up**
- **Decentralized and Cooperative**
- **Source code available for public testing**
- **A large network of communicating developers**
- **Loosely-organized volunteer (Hackers)**
- **Release made frequently to attain feedback**

Software Development Model



The Cathedral and The Bazaar

- **Written by Eric Raymond**
- **Collection of Essays**
- **Explores open source development, hacker culture, economics of open source**
- **First complete and commercially distributed book published under an open source document license**
- **Contains Fetchmail Story**



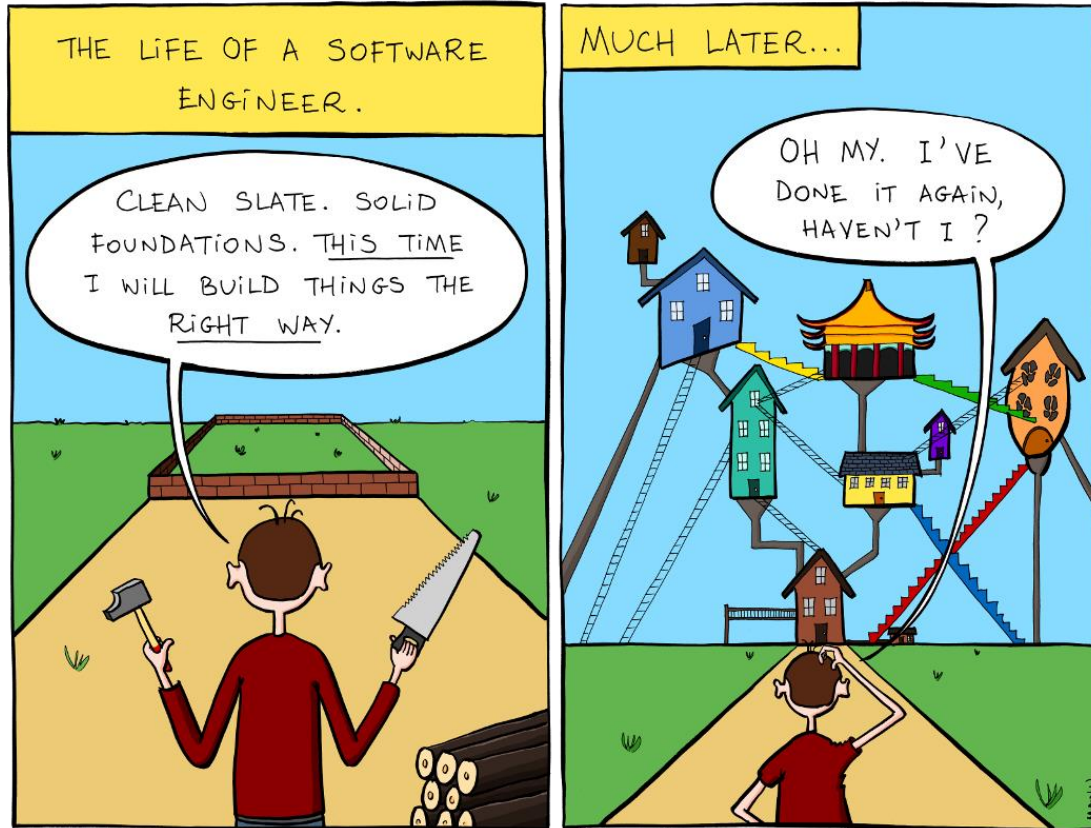
Fetchmail

- **A new and fancy mail client program is needed**
- **Needs**
 - Access mail from stand-alone computer that's not always on-line
 - Can't simply forward
- **POP3 Server available, need POP3 client**
- **Missing feature**
 - Replying to email from stand-alone computer
 - Need to tack on end of email address
- **Every good work of software starts by scratching a developer's personal itch**
 - Necessity is the mother of invention

Don't reinvent the wheel

- **Candidates**
 - 9 existing programs were close
 - Selected 'fetchpop' first
- **Example**
 - Linux Torvalds started, not from scratch, but with Minix
- **Good programmers know what to write. Great ones know what to rewrite (and reuse)**

Don't reinvent the wheel



Never fear refactoring

- **Fetchpop**
 - Consists of amateur codes
 - Handles only POP3
- **Popclient**
 - Written by Carl Harris
 - It was more cleanly written
 - However lacked some key, difficult to implement features
 - Stay or switch?
- **Plan to throw one away, you will, anyhow**
 - Fetchpop → popclient

Be kind and share

- **Popclient**
 - Carl Harris had basically lost interest in popclient
 - The code was a bit dusty, with minor bugs hanging
 - Popclient is taken over to Eric Raymond
- **When you lose interest in a program, your last duty is to hand it off to a competent successor**
 - Carl Harris handed ‘popclient’ code to Eric Raymond

Users as developers

- **Users can become co-developers.**
 - Source code is available, they can be effective hackers.
 - This can be tremendously useful for shortening debugging time
- **Treating your users as co-developers is your least-hassle route to rapid code improvement and effective debugging**
 - Power was underestimated until Linux
- **“I’m basically a very lazy person who likes to get credit for things other people actually do – Linus Torvalds**

Release early, release often

- **Cathedral**
 - Release occasionally → buggy code → alienate users
- **Bazaar**
 - Release often → motivate developers → maximize development time

Linus' Law

- **“Given enough eyeballs, all bugs are shallow.”**
- **“Given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix obvious to someone.”**
- **“Debugging is parallelizable.”**
- **“Delphi effect” – averaged opinion of a mass of observers (equally expert or ignorant) is more reliable than opinion of random observer**

Open Source Software License

Most slides from <https://www.blackducksoftware.com>

Types Of Open Source Licenses

- **Restrictive**
 - Requires licensor to make improvements or enhancements available under similar terms
 - Licenses establish a specific trigger for sharing obligation
 - Example is the GPLv2
- **Permissive**
 - Modifications/enhancements may remain proprietary
 - Distribution in source code or object code permitted provided copyright notice & liability disclaimer are included and contributors' names are not used to endorse products
 - Examples: Berkeley Software Distribution (BSD), Apache Software License

Types Of Open Source Licenses

Capabilities (Without Application Licensing Restriction)	GPL (Linux)	Dual-GPL (MySQL)	LGPL/MPL (OpenOffice, Firefox)	Apache/BSD (Apache, FreeBST)
1) Download	✓	✓	✓	✓
2) Evaluate	✓	✓	✓	✓
3) Deploy	✓	✓	✓	✓
4) Redistribute	✗ ¹	✓ ³	✓	✓
5) Modify	✗ ²	✗ ²	✗ ²	✓ ⁴

1) Application needs to be licensed under GPL if redistributed with the GPL asset.

2) Library code modifications need to be licensed under the same license as the originating asset.

3) Usually requires a commercial license from the copyright holder.

4) Although much more permissive than an OSI license, some BSD based licenses, such as Apache V2, still have some copyleft materials.

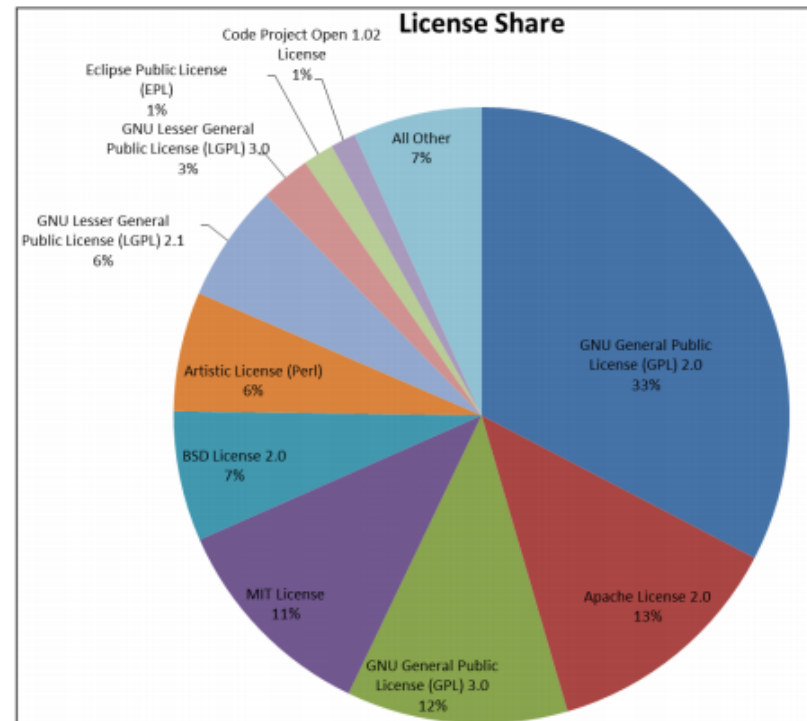
Top 20 Open Source License

Rank	License
1	GNU General Public License (GPL) 2.0
2	Apache License 2.0
3	GNU General Public License (GPL) 3.0
4	MIT License
5	BSD License 2.0
6	Artistic License (Perl)
7	GNU Lesser General Public License (LGPL) 2.1
8	GNU Lesser General Public License (LGPL) 3.0
9	Eclipse Public License (EPL)
10	Code Project Open 1.02 License
11	Microsoft Public License
12	Mozilla Public License (MPL) 1.1
13	Common Development and Distribution License (CDDL)
14	BSD 2-clause "Simplified" or "FreeBSD" License
15	Common Public License (CPL)
16	zlib/libpng License
17	Academic Free License
18	GNU Affero GPL v3
19	Microsoft Reciprocal License (Ms-RL)
20	Open Software License (OSL)

Source: [//www.blackducksoftware.com/oss/licenses#top20](http://www.blackducksoftware.com/oss/licenses#top20)
July 2013

Ranked according to number of open source projects using the license:

- Top 10 licenses account for **93%**
- Top 20 licenses account for **97%**
- GPL family of licenses account for **53%**
- Apache+BSD+MIT licenses account for **31%**



General Public License

- **GPL is a copyleft license**
 - Requires derivative works to be released under the same license
 - Programs linked with a library released under the GPL must also be released under the GPL.
- **GPLv3**
 - Requires that the user must be able to run the modified versions on the device as long as the device is targeted to consumer.
 - Stronger Protection Against Patent Threats

Lesser GPL License

- **Can be linked to by non-GPL compatible software**
 - **GPL license force to link to only GPL software**
- **The license allows developers and companies to use and integrate software released under the LGPL**
 - **Proprietary software can use GPL software**
- **If you change LGPL software which link to yours, you have to distribute your software under LGPL**

Apache License

- **Permissive free software license**
 - **Written by Apache Software Foundation**
 - **Requires preservation of the copyright notice and disclaimer**
- **Apache License allows the user of the software the freedom to use the software for any purpose, to distribute it, to modify it, and to distribute modified versions of the software**

End