

What's in a license

European Parliament A Free Software Licensing Crash Course!

Or

All You Need to Know About Free and Open
Source Software and You Didn't Dare Asking

Carlo Piana, Lawyer
Counsel, Free Software Foundation Europe
License Creative Commons by-sa 3.0

An historic perspective

- Software was free
- Then ISV came along
- Need to “protect” and monetize work producing Software
- How?

Various possibilities

- Copyright (eventually prevailed)
- Patents
- “*sui generis*”

Copyright, how it works

- No need to apply, just create it
- Automatically author gets exclusionary rights
 - Make copies
 - Adaptations/modifications
 - Translations
 - Rent, borrow
- Disposing of these rights requires a positive act
- Or wait until you're dead plus 70 years!

So you always need a license?

- Not needed to “buy” a copy: EULA are unnecessary
 - Bonus: you can resell copies, make backups, reverse engineer your copy (under some conditions)
- But licenses needed to do non-statutory acts
- Like releasing software as Free Software

What is Free Software (open source)

- Software is licensed in a way that grant the 4 Freedoms:
 - #0 to use the software
 - #1 to study the software, how it's made
 - #2 to pass it onto others
 - #3 to make and pass onto others modified versions of it

How?

- DIY?
- Use an already existing license? Which? How many are there?
- Is license proliferation a bad thing?

Yes, proliferation == Bad

- Legal language is not deterministic, and it is neither mathematics
- Consequences of legal language difficult to predict
- Only ex post some clarity is possible
- Main issue: licence incompatibility

How can licenses be incompatible?

- Free Software Not a new concept
- BSD was born mid sixties
- Very low requirements (conditions), mix and match software under many different conditions
- Including proprietary
- Unix, born in universities, spawned a generation of incompatible proprietary Unices

Copyright



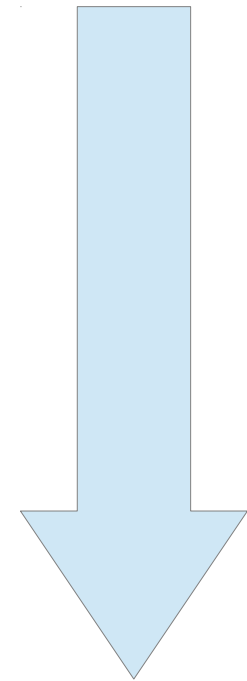
Copyleft

- The “contrary” of copyright (left the opposite of right)
- Uses copyright to let software to remain free (left as in to leave)
- Condition of reciprocity
 - License in == license out

How many copyleft do we have?

- Strong copyleft
 - “product level” copyleft
 - No linking or inclusion in incompatibly licensed software
 - High resilience to proprietarization
- Weak copyleft
 - “file level” copyleft
 - Permits linking into otherwise licensed works (including proprietary)
 - Low resilience to proprietarization
- No copyleft (permissive)
 - No resistance to proprietarization

Max incompatibility



Min incompatibility

Checklist

- Do we want to license it?
- Do we want to license it under Free Software?
- Permissive, weak or strong copyleft?

Strong copyleft options

- GNU (A)GPL
 - v2: old, popular, but not up with new threats (e.g., patents)
 - v3: new but adversed by large companies *because* “too” powerful
- EUPL
 - Conceived for releasing software of the European institutions

Why not just EUPL

- EUPL created proliferation
- Ostensibly for a cause
- But created incompatibility
- Solved with a compatibility clause...
- But was it?

EUPL incompatible with GPL v.3

- Why? Because of non reciprocity of compatibility?
- No such a thing, strong copyleft by design must be incompatible
 - GPL v.3 made by design incompatible with v.2 (but not with v.2+)
- Especially, EUPL is recessive
- GPL → EUPL → weak copyleft loophole

What is the proposal

- Not fixating on one single license
- Double licensing: EUPL+ and AGPL v3+
- What “+” means, then?

+ is shorthand for “any later version”

- Once software is licensed only the owner(s) of copyright can change licensing
- But with many owners, agreement is impossible
- “or any later version” = permits upgrading to an officially released newer version issued by the steward of the license
- Not automatic, decision of any downstream recipient
- Guarantee it remains strong copyleft

Final: what's in a “A”

- “A” comes from “Affero”, an additional condition to GPL v.2
- Closing the “ASP loophole”, software modified and not distributed, but made available through offering services (“Cloud”)
- No distribution == no requirement to make source code available
- Affero requires so
- In v.3 it was decided to accept the clause and make a new license, the AGPL, with an additional section

Conclusion

- EUPL OK-ish, but with a + provision
- AGPL v.3+ is a natural choice
- The recipient will decide which course to take in further downstream distribution
- Purpose: reap contributions/enhancements/community: possible
- Share: for sure, on a legally sound and considerate way

That's all folks!

Open to questions