

Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

# Gestion des licences d'un projet informatique

Rémi Boulle mail@remiboulle.fr



Rémi Boulle  
mail@remiboulle.fr

## Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

# Contexte

## Contexte

- projet GNU en 1984 par Richard Stallman (RMS)
- Première GPL : 25 février 1989
- la technique est un moyen pour atteindre un but social
- 1991 : Système d'exploitation Minix (« *just a hobby, won't be big and professional like gnu* ») par Linus Torvalds.
- 1995 : création de Red Hat (au Nasdaq en 1999), licence Apache
- 1998 : libération de Netscape
- 1998 : fracture entre « libre » et « open-source » (OSI) autour du *copyleft* par Éric Raymond.



**Linus Torvalds**

GNU GENERAL PUBLIC LICENSE - Version 3,  
29 June 2007 - Copyright © 2007 Free  
Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and  
distribute verbatim copies of this  
license document, but changing it is not  
allowed.

Preamble. The GNU General Public License  
is a free, copyleft license for software  
and other kinds of works. The licenses  
for most software and other practical  
works are designed to take away your  
freedom to share and change the works.

By contrast, the GNU General Public  
License is intended to guarantee your  
freedom to share and change all versions  
of a program to make sure it remains  
free software for all its users. As the  
Free Software Foundation, use the GNU  
General Public License for most of our  
software; it applies also to any other  
work released this way by its authors.  
You can apply it to your programs, too.

When we speak of free software, we are  
referring to freedom, not price. Our  
General Public Licenses are designed to  
make sure that you have the freedom to  
distribute copies of free software (and  
charge for them if you wish), that you  
receive source code or can get it if you  
want it, that you can change the

software or use pieces of it in new free  
programs, and that you know you can do  
these things. To protect your rights, we  
need to prevent others from denying you  
these rights or asking you to surrender  
the rights. Therefore, you have certain  
responsibilities if you distribute  
copies of the software. Or if you modify  
it, responsibilities to respect the  
freedom of others. For example, if you  
distribute copies of such a program,  
whether gratis or for a fee, you must  
pass on to the recipients the same  
freedoms that you received. You must  
make sure that they, too, receive or can  
get the source code. And you must show  
them these terms so they know their  
rights.

```
boot __init();
page_address_init();
pr_notice("GNU Linux banner\n");
setup_arch(command_line);
mm_init_cache(&init_mm);
setup_command_line(command_line);
setup_nr_cpu_ids();
setup_nr_cpu_areas();
boot_cpu_state_init();
mm_freemem_boot_cpu();
build_all_zonelists(NULL, NULL);
page_init();
pr_notice("kernel command line: '%s'\n",
        parse_early_param());
after_forked = parse_args("Booting kernel\n",
        static_command_line, &start_page,
        &stop_page, &start_param,
        &end_param, &unknown_bootoption);
if (!IS_ERR_OR_NULL(after_forked))
    parse_args("Setting init args", after_forked,
        NULL, &init_arg);
just_label_init();
setup_io_huff();
pidmax_init();
vfs_cache_init_early();
soft_main_table();
tmp_init();
mm_init();
sched_init();
preempt_disable();
if (WARN(!irqs_disabled(),
        "Interrupts were disabled 'early' but\n",
        local_irq_disable());
ide_init_cache(&__init());
trace_init(); /* enables 'tracking' init */
early_irq_init(&__init_irq); task_init();
cpu_init_data(); init_timers();
hrtimers_init(); softirq_init();
clockchasing_init(); time_init();
sched_clock_postinit();
printk_mm_init(); perf_event_init();
profile_init(); call_function_init();
WARN(!irqs_disabled(), "Interrupts were\n",
        early_boot_irqs_disabled = false; local_irq_enable();
        init_irqs();
        kernel_init_freeable();
        async_synchronize_full();
        free_initmem();
        mark_readonly();
        kernel_state = SYSTEM_RUNNING;
```

Rémi Boulle  
mail@remibouille.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

## International

Free Software Foundation : <https://www.fsf.org/>

Open Source Initiative : <https://opensource.org/>

Linux Fondation : <https://www.linuxfoundation.org/>

Communautés de développeurs par projets (debian, Ubuntu, python...)



Rémi Boulle  
mail@remibouille.fr

## Contexte

TP1

TP2

TP3 et TP4

## International

Free Software Fondation : <https://www.fsf.org/>

Open Source Initiative : <https://opensource.org/>

Linux Fondation : <https://www.linuxfoundation.org/>

Communautés de développeurs par projets (debian, Ubuntu, python...)

# France

CNLL : <http://cnll.fr/>

April : <https://april.org/>

Framasoft : <https://framasoftware.org/>

Toulibre : <http://toulibre.org/>

etc...

Rémi Boulle

mail@remiboulle.fr

## Contexte

## Droit d'auteur

## Libre

## TP1

## TP2

## Libre

Mouvement social, question de liberté et de communauté.

*"Toutes les libertés dépendent de la liberté informatique, elle n'est pas plus importante que les autres libertés fondamentales mais, au fur et à mesure que les pratiques de la vie basculent sur l'ordinateur, on en aura besoin pour maintenir les autres libertés". (RMS)*

# Open-source

Open-source : code ouvert, méthodologie de développement, *"l'idéologie, c'est nul"* (selon LT), pragmatisme

## On est toujours l'idéologue de quelqu'un... Débat dépassé ?

Rémi Boulle  
mail@remiboulle.fr

## Vocabulaire

Libre

TP1

TP2

TP3 et TP4

# Vocabulaire

- **Logiciels privatifs** et non propriétaires (tous le sont !)
- **Copyleft** : *droit laissé*, obligation de garantir que les libertés offertes par la licence soient préservées lors de la redistribution.
- **Copyright** : pas de valeur juridique particulière en France mais permet d'identifier les auteurs...
- **Oeuvre** : création originale, pas vraiment de sens dans le monde numérique
- **Originalité** : expression juridique de la créativité de l'auteur, empreinte de sa personnalité. **Création intellectuelle** dans le monde numérique

Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

**Droit d'auteur**

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

# Droit d'auteur

- Droits patrimoniaux
  - Droit de reproduction, de représentation
  - Droit de suite (pas pour les logiciels)
  - Droits voisins
- Droits extra-patrimoniaux
  - perpétuels et inaliénables
  - paternité, divulgation respect de l'oeuvre, repentir, retrait (quasi éradication dans le numérique).

## Droit d'auteur adapté

Libre et privatif sont régis par le *droit d'auteur adaptés au logiciels*.

## Gauche d'auteur

Le libre est adossé au droit d'auteur.

Les idées sont de libre parcours.

Rémi Boulle  
mail@remiboulle.fr

## Privatif

Libre

TP1

TP3 et TP4

# Privatif

Principe général : réserver l'intégralité des droits au titulaire.  
Droits patrimoniaux organisés pour assurer un monopole d'exploitation économique.

## Définition : logiciel privatif

Logiciel sous licence privative, c'est-à-dire n'offrant pas au moins une des quatre libertés des licences libres.





Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

*Si on ne peut pas rentrer par la porte, rentrons par la fenêtre !*  
Qui gouverne quel projet de logiciel libre ? Posez-vous juste la question.

Rémi Boulle  
mail@remibouille.fr

Libre

TP1

TP3 et TP4

## Libre

## Principe général

Toutes les licences entendent que le *code source* reste libre.

Elles ne divergent principalement que sur les modalités de redistribution :

- pas d'obligation de redistribuer le code
- obligation de redistribuer le code sous les mêmes termes : empêcher de tirer bénéfice du logiciel sans reverser en retour sa propre oeuvre dérivée.

## Définition : logiciel libre

Logiciel sous licence libre, c'est-à-dire offrant les quatre libertés des licences libres.

- Absolue liberté d'utilisation sans aucune restriction
- Oeuvre produite en utilisant un logiciel libre n'est pas soumise à sa licence
- Résultats produits par un logiciel : parfois des oeuvres dérivées...
- Quid de l'utilisation sans distribution ?

## Principe

Dans l'intérêt du licencié.

Question rapide : Est-ce que la licence de JSON est libre ?

<http://www.json.org/license.html>

- Liberté d'étude : aucun obstacle juridique ou technique.
- Pas obligatoire de livrer le source avec le binaire (=pragmatisme)
- Mise à disposition des sources à cout nul ou faible
- Code source mis à disposition au minium 3 ans (GPLv3)

# Principe

Dans l'intérêt du licencié encore.

Rémi Boulle

## Vocabulaire

Droit d'auteur

Libre

TP1

TP2

## Classification

## Périmètre

TP3 et TP4

## Futur ?

- Liberté de modification absolue mais restriction si distribution
- Préserver les mentions de titularité du droit d'auteur
- Droit au nom : identifier les contributeurs (L.121-1 du CPI) et même convention de Berne (internationalement reconnu)
- Droit au respect de la réputation mis en oeuvre par certaines licences : identifier les contributions.

# Principe

Dans l'intérêt du licencié toujours.

# liberté 3 : liberté de redistribution

Rémi Boulle  
mail@remibouille.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

- Modalités de redistribution : principale différence entre licences libres :
  - **Copyleft fort**
  - **Copyleft faible**
  - **Copyfree**
- Droit au nom : identifier les contributeurs (L.121-1 du CPI et convention de Berne)
- Droit au respect de la réputation mis en oeuvre par certaines licences : identifier les contributions.

## Principe

Dans l'intérêt des autres utilisateurs. Cette liberté est surtout une condition nécessaire aux 3 précédentes.

## Quand la licence se déclenchent-elles ?

### Redistribuer déclenche la licence

Exception : AGPL où dès qu'il y a interaction avec le service, les obligation de la licence se déclenchent. (art. 13)



Rémi Boulle  
mail@remibouille.fr

Libre

TP1

TP2

TP3 et TP4

# TP1

En 30 minutes et par groupe de 3 :

- ouvrir un pad collaboratif
- analyser pour chacune des 4 libertés du logiciel libre leur transposition dans les logiciels privatifs
- la mise en commun des travaux des groupes se fera sur un pad commun

## Consigne générale

## Toujours sourcer vos informations

Rémi Boulle  
mail@remiboulle.fr

Libre

TP1

TP2

TP3 et TP4

## TP2

# Le libre dans vos entreprises ?

Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

- Y-a-t-il une "gouvernance" autour du libre dans vos entreprises ?
- Utilisez-vous des briques libres dans vos développements ? Précisez.

Mise en commun sur : <https://frama.link/entrepriseLL>

Rémi Boulle  
mail@remibouille.fr

Libre

TP1

TP2

## Classification

TP3 et TP4

# Classification

# Principe général

Empêcher de tirer bénéfice auprès de tiers sans reverser en retour sa propre contribution

### Définition (Pellegrini, Canevet)

Licence libre **avec copyleft** imposant que toute nouvelle contribution "s'appuyant" sur du code source placé sous cette licence soit également placé sous les termes de cette licence

On parle de *copyleft fort*.

Autre termes rencontrés : "licences virales", "contaminantes" (à éviter, idéologie embarquée).

Exemples : GNU GPL, CeCILL(A)....

## Stratégiquement **expansionnistes.**

## Principe général

Empêcher toute privatisation de l'oeuvre sans s'appliquer aux oeuvres dérivées.

### Définition (Pellegrini, Canevet)

Licence libre **avec copyleft** n'empêchant pas l'usage des  
oeuvres placées sous leur régime au sein d'oeuvres placées sous  
d'autres licences (y compris privatives)

On parle de **copyleft faible**, licences "pérennes". Plutôt pour des librairies.

Exemples : GNU LGPL, CeCILL-C (Composant)

## Stratégiquement **défensives**.

## Principe général

Liberté totale !

### Définition (Pellegrini, Canevet)

Licence libre **sans copyleft** autorisant la redistribution du logiciel sans son code source.

## On parle de *copyfree*

Autre terme moins rigoureux : licences permissives

Exemples : BSD (BSD 4 clauses, BSD 3, BSD 2), MIT, Apache, CeCILL-B...

## Stratégiquement **prosélytes**.



Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

**Périmètre**

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

# Périmètre



La diffusivité n'est pas du tout absolue.

## But général des diffusives

- Contrepartie
- Coexistence

- Diffusivité ascendante si oeuvre dérivée (**basée** sur celui-ci au sens du droit d'auteur). Bibliothèque et appels systèmes (libc LGPL et Linux GPLv2) ?
- Diffusivité descendante : pilotes = oeuvre dérivée ? (wrapper)

Oeuvre dérivée dès qu'il n'y pas substituabilité (selon si le couplage avec le code diffusif est spécifique ou générique)

## Périmètre

Le périmètre de la licence est défini par la substituabilité..

Trois cas :

- oeuvre utilisant... : Si "oeuvre considérée comme utilisant simplement la bibliothèque", pas d'obligations (sauf liaison statique, voir licence)
- oeuvres combinées.... avec la bibliothèque
- ajout de fonctionnalités

Préserver le périmètre fonctionnel du logiciel couvert pour éviter subversion... (cf MPL)

## Périmètre défini de façon abstraite et fonctionnelle en termes de modules

# Périmètre

Le périmètre de la licence est l'interface.

## But général des évanescences

Aucune obligation de distribuer le source avec le code objet.

## Périmètre

Le périmètre de la licence est le fichier lui-même.

Suivre les indications des licences...

Rémi Boulle  
mail@remibouille.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

**Compatibilité**

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

# Compatibilité



## Définition : licences incompatibles

## Licences qui imposent des obligations contradictoires..

Licence B est compatible avec licence A si :

- les droits de B sont inclus dans ceux de A
- les obligations de A sont incluses dans celles de B

*nemo plus juris*

*Personne ne peut transférer à autrui plus de droits qu'il n'en a lui-même.*

Les licences diffusives sont par nature incompatibles entre elles...



# La compatibilité en un tableau

Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

Votre code = du code + A + B. Pouvez-vous le distribuer ?

Code B \ Code A	Diffusive	Persistante	Evanescente
Diffusive	<b>Impossible</b> sauf si clause de compatibilité	<b>Impossible</b> sauf si clause de compatibilité	<b>Possible</b> P peut être distribué sous la licence de A
Persistante	<b>Impossible</b> sauf si clause de compatibilité	<b>Possible.</b> Distribution sans que les licences A et B ne se diffusent et à condition que chaque module soit distribué sous sa propre licence	<b>Possible.</b> Distribution sans que la licence de A ne se diffuse à la partie hachurée et au code B et à condition que A soit distribué sous sa licence.
Evanescente	<b>Possible.</b> P peut être distribué sous la licence B	<b>Possible.</b> P peut être distribué sans que la licence B ne se diffuse à la partie hachurée et le module A à condition que B soit distribué sous la licence B	<b>Possible.</b> Aucune diffusivité.

Rémi Boulle  
mail@remibouille.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

**TP3 et TP4**

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

## TP3 et TP4

Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

Indispensable pour tout développeur au risque de rater sa vie.

*The SMOG grade is a measure of readability that estimates the years of education needed to understand a piece of writing.*

*SMOG is an acronym for Simple Measure of Gobbledygook.*

Selon Alexios Zavras, le SMOG grade moyen est de 17.

Certaines licences dépassent 25.

Par groupe de 3, dans le document fourni, lire et commenter la GPLv3 (durée 1h) :

- reformuler clairement sans paraphraser
- expliquer en détail certaines notions : pourquoi sont-elles introduites, approfondir (brevets, MTP, diffusivité...)
- comparer même brièvement avec d'autres licences libres (CeCILL, MIT, LGPL, AGPL...)



Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

**TP3 et TP4**

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

Par groupe de 3, répondez aux 3 questions tirées au sort

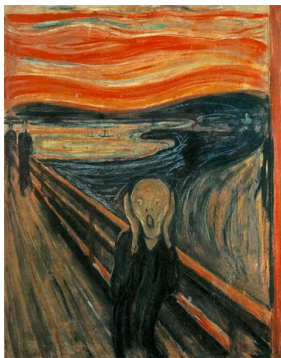
- Chaque groupe présentera ces réponses en 10 minutes questions de la promo comprises
- Notation en 7, 10, 12, 14, 18, 20.





Selon *Black Duck Software* :

- 30% de la base de code des entreprises serait libre
- 98% des entreprises ne le sauraient pas...



Marketing de la *peur* mais :  
Double contrainte :

- juridique : compatibilité des licences (privatives, libres, libres entre elles)
- technique : failles de sécurités de n'importe quel composant externes (HeartBleed dans OpenSSL, ...)

Finalement, une chance que les composants impactés soient des logiciels libres...



Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

**Bonnes  
pratiques et  
audits**

Futur ?

Bibliographie  
et crédits

Essentiellement trois solutions...

Rémi Boulle  
mail@remibouille.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

**Bonnes  
pratiques et  
audits**

Futur ?

Bibliographie  
et crédits

Fondé par un ancien de Microsoft.

- Open Source Application Security
- Open Source Compliance and Management

<https://www.blackducksoftware.com/>

## FOSSology

*License compliance software system and toolkit.*

- *As a toolkit you can run license, copyright and export control scans from the command line.*
- *textitAs a system, a database and web ui are provided to give you a compliance workflow. License, copyright and export scanners are tools available to help with your compliance activities.*

<https://github.com/fossology/fossology>

49/58

Rémi Boulle  
mail@remibouille.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

Bibliographie  
et crédits

*ScanCode is a suite of utilities used to scan a codebase for license, copyright, and other interesting information that can be discovered in files.*

*A typical software project often reuses hundreds of third-party components. License and origin information is often scattered and not easy to find: ScanCode discovers this data for you.*

<https://github.com/nexB/scancode-toolkit>

ffmpeg
compat
doc
libavcodec
libavdevice
libavfilter
tests
x86
af_volume.asm
af_volume_init.c
avf_showcqt.asm
avf_showcqt_init.c
colospacecsp.asm
colospacecsp_init.c
Makefile
vf_blend.asm
vf_blend_init.c
vf_bwdif.asm
vf_bwdif_init.c
vf_eq.c
vf_fspp.asm
vf_fspp_init.c
vf_gradfun.asm
vf_gradfun_init.c
vf_hqdn3d.asm
vf_hqdn3d_init.c
vf_idet.asm
vf_idet_init.c
vf_interlace.asm
vf_interlace_init.c
vf_maskedmerae.asm

## License Summary

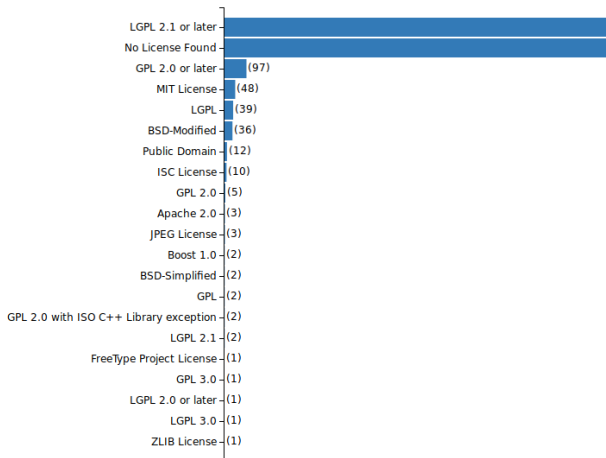
## Copyright Summary

## License &amp; Copyright Details

## File Details

## Packages

Total Files Scanned: 6471



Rémi Boulle  
mail@remibouille.fr

Libre

TP1

TP3 et TP4

## Futur ?

# Futur ?

Préliminaire : avoir des juristes compétents sur ces thèmes.

## Problèmes

- 70% des contributions à Linux proviennent de quelques entreprises. Lesquelles ?
- Wikipédia dépend fortement de Google.
- ....
- Gouvernance communautaire ?

## Pistes

- "Peer Production Licence" (*copyfarleft*)
- "Commons reciprocity licence"
- "Fair source"
- Licence = mauvaise piste ? Piste légale plutôt ?

Voir conférence de Calimaq :

<https://scinfolex.com/tag/capitole-du-libre/>



Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

**Bibliographie  
et crédits**

# Bibliographie et crédits







Rémi Boulle  
mail@remiboulle.fr

Contexte

Vocabulaire

Droit d'auteur

Privatif

Libre

TP1

TP2

Classification

Périmètre

Compatibilité

TP3 et TP4

Bonnes  
pratiques et  
audits

Futur ?

**Bibliographie  
et crédits**

Document sous licence GFDL1.3+, CC-BY-SA+, LAL+.