

BLOC 2

# UTILISER

Les licences logicielles

1h30

*J'utilise une bibliothèque open source, ai-je le droit ?*

# Objectifs du Bloc 2

À l'issue de ce bloc, vous saurez :

- Comprendre ce qu'est une licence logicielle et pourquoi elle est nécessaire
- Distinguer les licences permissives des licences copyleft
- Identifier les principales licences (MIT, Apache, GPL, LGPL)
- Analyser un fichier LICENSE et en déduire vos obligations
- Choisir une licence adaptée pour votre propre projet

Ce bloc inclut un exercice pratique d'analyse de fichiers LICENSE sur GitHub

# Pourquoi les licences existent-elles ?

Rappel du Bloc 1 : Le code source est protégé par le droit d'auteur

Sans autorisation de l'auteur, vous ne pouvez PAS :

- Copier le code
- Modifier le code
- Distribuer le code (même gratuitement)
- Intégrer le code dans votre projet



## LA LICENCE

= L'autorisation donnée par l'auteur d'utiliser son code sous certaines conditions

Définition :

Une licence logicielle est un contrat par lequel le titulaire des droits accorde à un tiers le droit d'utiliser, copier, modifier et/ou distribuer un logiciel, sous certaines conditions.

# Formalisation : Puis-je utiliser ce code ?

Application de la méthode algébrique :

```
PEUT_UTILISER_CODE =  
    (A_LICENCE OR EST_DOMAINE_PUBLIC) AND  
    RESPECTE_CONDITIONS_LICENCE
```

Axiomes :

- A\_LICENCE : le code dispose d'une licence explicite
- EST\_DOMAINE\_PUBLIC : le code est explicitement placé dans le domaine public
- RESPECTE\_CONDITIONS\_LICENCE : vous respectez toutes les obligations de la licence

Et si le code n'a PAS de licence ?

Absence de licence ≠ Libre de droits

Sans licence explicite, le code est protégé par défaut et vous ne pouvez PAS l'utiliser !

# Deux grandes familles de licences

Toutes les licences open source se classent en deux catégories :

## LICENCES PERMISSIVES

Philosophie : "Faites ce que vous voulez"

- Peu de restrictions
- Utilisation commerciale libre
- Pas d'obligation de partager vos modifications
- Obligation minimale : attribution

Exemples : MIT, BSD, Apache 2.0

## LICENCES COPYLEFT

Philosophie : "Partagez à votre tour"

- Restrictions sur la redistribution
- Utilisation commerciale possible
- Obligation de partager vos modifications sous même licence
- "Effet viral" : se propage

Exemples : GPL, LGPL, AGPL

# La licence MIT

La licence la plus populaire sur GitHub (~45% des projets)

## MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software...

...subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

## En résumé :

- Vous POUVEZ : utiliser, copier, modifier, distribuer, vendre
- Vous DEVEZ : conserver la notice de copyright
- Vous N'AVEZ PAS À : partager vos modifications

# Formalisation : Licence MIT

```
UTILISATION_MIT_CONFORME =  
    CONSERVE_COPYRIGHT_NOTICE AND  
    CONSERVE_LICENCE_NOTICE
```

## Conditions très simples :

- CONSERVE\_COPYRIGHT\_NOTICE : garder la mention "Copyright (c) [année] [auteur]"
- CONSERVE\_LICENCE\_NOTICE : garder le texte de la licence dans votre distribution

### AUTORISÉ

- Usage commercial
- Modification
- Distribution
- Usage privé
- Sous-licence

### INTERDIT

- Retirer la notice copyright
- Tenir l'auteur responsable des bugs/dommages
- Utiliser le nom de l'auteur pour promouvoir votre produit

# La licence Apache 2.0

## Licence permissive avec protection supplémentaire

Similaire à MIT, mais avec des clauses supplémentaires :

- Grant de licence de brevet : l'auteur vous accorde une licence sur ses brevets
- Clause de rétention : si vous poursuivez pour violation de brevet, votre licence est révoquée
- Notice de modifications : vous devez documenter les changements que vous faites
- Fichier NOTICE : si présent, doit être conservé

```
UTILISATION_APACHE_CONFORME =
    CONSERVE_COPYRIGHT AND
    CONSERVE_LICENCE AND
    DOCUMENTER_MODIFICATIONS AND
    CONSERVE_NOTICE_SI_EXISTANT
```

*Utilisée par : Android, Kubernetes, Swift, TensorFlow...*

# La licence GPL (GNU General Public License)

La licence copyleft la plus connue - "L'effet viral"

Principe fondamental de la GPL :

**Si vous distribuez un logiciel basé sur du code GPL,  
vous DEVEZ distribuer TOUT votre code source sous licence GPL**

Conséquences pratiques :

- Utilisation interne : pas de problème (pas de distribution)
- Distribution (vente, publication) : obligation de fournir le code source
- Votre code devient également GPL : "effet viral"
- Incompatible avec certaines licences propriétaires

Utilisée par : Linux (kernel), WordPress, GIMP, GCC...

# Formalisation : Licence GPL

```
DOIS_PUBLIER_CODE_SOURCE =
    UTILISE_CODE_GPL AND
    DISTRIBUE_LOGICIEL
```

```
DISTRIBUTION_GPL_CONFORME =
    FOURNIT_CODE_SOURCE AND
    CODE_SOUS_GPL AND
    CONSERVE_NOTICES
```

## Le point clé : qu'est-ce que "distribuer" ?

### PAS de distribution =

- Utilisation personnelle
- Utilisation interne (entreprise)
- Tests et développement
- Exécution sur serveur (sauf AGPL)

### Distribution =

- Vente du logiciel
- Publication (téléchargement)
- Inclusion dans un produit
- Distribution à des tiers

*Important : exécuter du code GPL sur un serveur web n'est PAS une distribution (mais AGPL couvre ce cas)*

# La licence LGPL (Lesser GPL)

## GPL "allégée" pour les bibliothèques

Différence avec la GPL :

GPL : Tout code qui UTILISE du code GPL doit être GPL

LGPL : Seules les MODIFICATIONS de la bibliothèque doivent être LGPL

Votre code qui UTILISE la bibliothèque peut rester propriétaire

```
DOIS_PUBLIER_MODIFICATIONS =
    MODIFIE_BIBLIOTHEQUE_LGPL
```

```
PEUT_GARDER_PROPRIETAIRE =
    UTILISE_SANS_MODIFIER AND
    LIAISON_DYNAMIQUE
```

Cas d'usage typique : Vous utilisez une bibliothèque LGPL (ex: Qt) sans la modifier  
→ Votre application peut rester propriétaire

*Utilisée par : Qt, glibc, LGPL'd libraries...*

# Comparatif des licences

Licence	Famille	Commercial	Modifier	Distribuer modifs	Partager source
MIT	Permissive	✓	✓	✓	Non requis
Apache 2.0	Permissive	✓	✓	✓	Non requis
BSD	Permissive	✓	✓	✓	Non requis
LGPL	Copyleft faible	✓	✓	Sous LGPL	Modifs seulement
GPL	Copyleft fort	✓	✓	Sous GPL	Tout le code
AGPL	Copyleft fort+	✓	✓	Sous AGPL	Même serveur

Permissive = peu de contraintes | Copyleft = obligation de partage

Conseil : Dans le doute, préférez les licences permissives pour éviter les problèmes !

# Arbre de décision : Puis-je utiliser ce code ?



Astuce : Vérifiez TOUJOURS la licence AVANT d'intégrer du code dans votre projet

# Choisir une licence pour votre projet

Vous publiez votre code ? Choisissez une licence adaptée à vos objectifs :

## "Je veux une adoption maximale"

→ MIT ou Apache 2.0

Tout le monde peut utiliser,  
même dans des projets propriétaires

## "Je veux que ça reste libre"

→ GPL v3

Les améliorations doivent être  
partagées avec la communauté

## "C'est une bibliothèque"

→ LGPL ou MIT

Permet l'utilisation dans des  
projets propriétaires

## "Je veux aussi protéger les brevets"

→ Apache 2.0

Inclut une clause de rétorsion  
sur les brevets

# Exercice pratique : Analyser des licences

## GitHub

Objectif : Apprendre à lire et analyser un fichier LICENSE

Instructions :

- Allez sur GitHub.com
- Trouvez le fichier LICENSE d'un des projets suivants :
  - facebook/react
  - vuejs/vue
  - torvalds/linux
- Identifiez le type de licence et ses implications

Questions à vous poser :

- Puis-je utiliser ce code dans un projet commercial ?
- Dois-je publier mes modifications ?
- Quelles mentions dois-je conserver ?

# Exemple : Licence de React (MIT)

## Projet : facebook/react

### MIT License

Copyright (c) Meta Platforms, Inc. and affiliates.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

## Analyse :

- Type : MIT (permissive)
- Vous pouvez : utiliser, modifier, distribuer, vendre, intégrer dans projet propriétaire
- Vous devez : conserver la notice copyright et la licence
- Implications : Aucune obligation de partager votre code

# Exemple : Licence de Linux (GPL v2)

## Projet : torvalds/linux

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

[...]

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

## Analyse :

- Type : GPL v2 (copyleft fort)
- Vous pouvez : utiliser, modifier, distribuer
- Vous devez : fournir le code source, garder sous GPL, documenter les modifications
- Implications : Si vous distribuez un dérivé, TOUT votre code doit être GPL

C'est pourquoi les drivers propriétaires pour Linux sont controversés !

# Exercice Wooclap : Quiz sur les licences

Testez vos connaissances !

5 questions rapides sur les licences logicielles

Q1. Un code sans licence est-il libre de droits ?

Q2. Quelle licence oblige à publier votre code source ?

Q3. Pouvez-vous vendre un logiciel sous licence MIT ?

Q4. Quelle licence choisir pour une adoption maximale ?

Q5. Qu'est-ce que l'"effet viral" de la GPL ?

Répondez sur Wooclap

# Correction du quiz

**Q1. Un code sans licence est-il libre de droits ?**

NON - Sans licence = droits réservés par défaut

**Q2. Quelle licence oblige à publier votre code ?**

GPL (et AGPL, LGPL pour les modifications)

**Q3. Pouvez-vous vendre un logiciel sous licence MIT ?**

OUI - La licence MIT autorise l'usage commercial

**Q4. Quelle licence pour une adoption maximale ?**

MIT ou Apache 2.0 (permissives)

**Q5. L'"effet viral" de la GPL ?**

Le code dérivé doit aussi être sous GPL

# Compatibilité des licences

Attention : toutes les licences ne sont pas compatibles entre elles !

Règle générale de compatibilité :



Le code sous licence permissive peut être intégré dans du GPL (mais pas l'inverse)

Exemples de combinaisons :

MIT + MIT	✓	MIT
MIT + Apache	✓	Apache ou MIT
MIT + GPL	✓	GPL (le tout devient GPL)
GPL + Propriétaire	✗	INCOMPATIBLE
Apache + GPLv2	✗	INCOMPATIBLE (clauses brevets)

*En cas de doute : consultez les conditions des deux licences ou évitez la combinaison*

# Ce qu'il faut retenir - Bloc 2

## Principe fondamental :

```
PEUT_UTILISER_CODE = (A_LICENCE OR DOMAINE_PUBLIC) AND RESPECTE_CONDITIONS
```

## Deux familles de licences :

- Permissives (MIT, Apache, BSD) : peu de contraintes, usage commercial libre
- Copyleft (GPL, LGPL) : obligation de partager le code source si distribution

## Points clés :

- Sans licence = droits réservés (ne pas utiliser)
- MIT/Apache = attribution suffit
- GPL = partage obligatoire si distribution
- Vérifiez TOUJOURS avant d'intégrer du code tiers

Ressource utile : <https://choosealicense.com>

**En entreprise : vérifiez les licences de TOUTES vos dépendances avant de livrer !**

# **Fin du Bloc 2**

UTILISER - Les licences logicielles

---

Prochaine étape :

## **Bloc 3 : TRAITER**

Le RGPD et la protection des données