

Journée **LOOPS** 17 novembre 2015

Présentation sous licence Creative Commons
Attribution - Pas d'Utilisation Commerciale -
Partage dans les Mêmes Conditions 4.0
International :



LOGICIELS : TOUR D'HORIZON DES LICENCES ET APPLICATIONS

UNIVERSITÉ
S PARIS
SUD

Comprendre le monde,
construire l'avenir®



Chahrazed BELGACEM
chahrazed.belgacem@u-psud.fr - Juriste SAIC UPSud
Jean-Christophe SOUPLET
jcsouplet@lri.fr – LRI Pôle Développement Logiciel – LRI

Objectifs

- **Grandes familles de licences**
 - Distinction licences propriétaires/licences libres
 - Introduction à différentes licences libres
- **Problématiques de compatibilité**
 - Notion de composants
 - Enchaînement de licences
- **Comment l'appliquer dans son code ?**
 - Entête
 - Fichier
- **Outils d'analyse de code et lien avec la PI**
 - Aide au calcul des pourcentages de contribution
 - Analyse des dépendances

LES GRANDES FAMILLES DE LICENCES

Distinction licences propriétaires/licences libres
Introduction à différentes licences libres

Définitions

- **Contrat** = un accord de volonté entre deux ou plusieurs personnes
- **Contrat de Licence** = Ce contrat régit les conditions d'utilisation et de distribution d'un logiciel. Il est mis en place entre le titulaire des droits et un tiers intéressé
- **Logiciel** = « Ensemble des programmes, procédés et règles, et éventuellement de la documentation, relatifs au fonctionnement d'un ensemble de traitement de données. » (Arrêté du 22 décembre 1981 sur l'enrichissement de la langue française définit le terme « logiciel »)

Pourquoi une licence ?

Article L335-2 du Code de la propriété intellectuelle :

« Toute édition d'écrits, de composition musicale, de dessin, de peinture ou de toute autre production, imprimée ou gravée en entier ou en partie, au mépris des lois et règlements relatifs à la propriété des auteurs, est une contrefaçon et toute contrefaçon est un délit.

La contrefaçon en France d'ouvrages publiés en France ou à l'étranger est punie de trois ans d'emprisonnement et de 300 000 euros d'amende. »

- ✓ **S'il n'y a pas un droit explicitement donné, utiliser un logiciel relève de la contrefaçon. Les licences sont des contrats et protègent les auteurs, les utilisateurs et les éventuels collaborateurs au développement.**

Licences propriétaires

Définition de Licence Propriétaire :

Un logiciel propriétaire est défini comme étant un **logiciel non libre**, c'est-à-dire qu'au moins une de ces actions est interdite ou limitée par le propriétaire : utiliser, étudier, modifier (\Rightarrow **autorisation du titulaire des droits patrimoniaux**).

Licences propriétaires

Avantages/Inconvénients

Vis à vis du **titulaire des droits patrimoniaux**, le statut de Logiciel propriétaire apporte :

- Garantie du droit d'auteur
- Le titulaire des droits patrimoniaux garde contrôle de l'évolution, de la distribution et/ou de l'usage de son œuvre
- Le titulaire des droits patrimoniaux reste propriétaire de l'œuvre
- Garantie la protection du logiciel
- Permet une rétribution du titulaire des droits patrimoniaux

Vis à vis de **l'utilisateur**, le statut de logiciel propriétaire apporte :

- Clarté des versions
- Code source inaccessible ou accessible mais de manière restrictive
- Interdiction de réaliser certaines copies
- Utilisation soumise aux conditions du titulaire des droits patrimoniaux
- Difficulté d'adaptation aux besoins spécifiques des utilisateurs
- Risque d'abandon

Licences libres

- Définition:

Une licence est libre lorsqu'elle garantit à l'utilisateur du logiciel un certain nombre de libertés fondamentales. Selon la Free Software Foundation, voici ces **4 libertés** :

➔ la liberté d'**utiliser** le logiciel pour tous les usages

➔ la liberté de **redistribuer** des copies du logiciel

➔ la liberté d'**étudier** le fonctionnement du logiciel, et de pouvoir l'adapter à ses besoins

➔ la liberté de **modifier** le logiciel et de publier ses modifications

Licences libres

**Pour résumer on peut
dire qu'un Logiciel
Libre est un logiciel
dont le propriétaire a
décidé de soumettre
son exploitation à des
Conditions
particulières**

Licences libres

Attention :

Le logiciel libre (différent du logiciel propriétaire) ne doit donc Pas être confondu avec :

- le logiciel dans le **domaine public** : logiciel non soumis au droit d'auteur (= l'auteur a renoncé à tous ses droits patrimoniaux ou 70 ans après la mort de l'auteur;
- le **freeware** : logiciel gratuit mais pas nécessairement libre ;
- le **shareware** : logiciel dont l'utilisation est soumise à rétribution du titulaire des droits patrimoniaux, après une période d'essai gratuite où les fonctionnalités sont souvent limitées.

Licences libres

Sans Copyleft

- Licence initiale ne s'impose pas
- Permission de le redistribuer et de le modifier, mais aussi d'y ajouter des restrictions.

Copyleft faible

- Licence initiale reste
- Ajouts peuvent avoir autre licence

Copyleft fort

- Licence initiale s'impose sur tout
- Licence « dite » contaminante

Quelques exemples

Type	Licences
Sans Copyleft	BSD license Apache License 2.0 CeCILL-B
Copyleft faible	GNU library or « Lesser » General Public License (LGPL) CeCILL-C
Copyleft fort	CeCILL V2 GNU General Public License

Licence GNU-GPL

- **La licence la plus connue et la plus répandue dans le monde du libre**
- **Cette licence autorise plusieurs actes sans l'accord de l'auteur et sans crainte d'une action en contrefaçon :**
 - l'utilisation du logiciel
 - l'étude du fonctionnement du logiciel,
 - l'adaptation du logiciel aux besoins de l'utilisateur,
 - la copie et la diffusion auprès d'amis ou de collègues,
 - l'amélioration du logiciel par l'utilisateur et la distribution du logiciel modifié au public.

Licence GNU-GPL

ATTENTION: il s'agit d'une Licence dite
contaminante !!

Elle impose à l'utilisateur de rediffuser ces modifications
sous licence GPL => Ainsi, **tout programme réutilisant un
bout de code GPL est contaminé par la GPL.**

Licences CeCILL

- **CeCILL** : Acronyme de « *CEa Cnrs Inria Logiciel Libre* » désignant une licence de **logiciel libre** établie par trois organismes de recherche français (le CEA, le CNRS et l'INRIA).
- Il existe **3** types de licences CeCILL:
 - **CeCILL** : redistribution uniquement sous CeCILL + obligation distribution code source à la demande.
 - **CeCILL B** : autorise la réutilisation de son logiciel sans aucune autre contrainte que la citation
 - **CeCILL C** : adaptée à la distribution des bibliothèques et aux composants logiciels. Elle impose au développeur de soumettre à cette licence les modifications apportées à leur code source tout en gardant la liberté de choisir une autre licence pour le reste de son application.

Licence BSD

BSD = Berkeley System Distribution license

⇒ Licence très peu restrictive

⇒ Les logiciels diffusés sous licence BSD peuvent être **librement copiés ou modifiés**. Une seule contrainte existe, faire figurer sur tous les travaux dérivés, les documentations et les publicités relatives à ces travaux une mention apparente faisant référence à la licence elle-même, et mentionnant les auteurs du logiciel original.

Licence BSD

- Note :

Dans le cas de logiciels très populaires, le nombre des auteurs est souvent très important. La mention de l'intégralité des contributeurs a abouti à la création de notices légales longues et illisibles. Pour supprimer ce désagrément, la **version 2 de la licence BSD supprime les obligations de mentions des auteurs.**

➔ La version 2 de la licence BSD est donc une des moins restrictives qui soit.

- <http://www.logidee.com/lib/exe/fetch.php/licences.pdf>
- <http://lelogiciellibre.net/tutoriaux/licence-logiciel.php>
- <http://www.cecill.info/faq.fr.html#differences>

PROBLÉMATIQUES DE COMPATIBILITÉ

Notion de composants

Enchaînement de licences

Constat

- Le développement d'un logiciel \neq développer à partir de zéro
 - Nombreux codes accessibles (internes, externes) et très souvent réutilisables
- Evitons de réinventer la roue

Les composants logiciels

- Éléments logiciels destinés à être inclus dans des applications. Le lien entre l'application et le composant se fait en passant par une interface définie.
- Exemples
 - Les bibliothèques logicielles, les exécutables, les fichiers (e.g. données, configuration, scripts, ...), les bases de données, les drivers, les codecs, ...
- Objectifs
 - Réutilisation du code
 - Collaborations – chacun responsable de sa brique logicielle
 - Mise sur le marché / Valoriser
- Toutefois
 - Pas de consensus strict sur la définition
 - Ici : cadre plus large → e.g. réutilisation de code source

→ Programme informatique encapsulé dans une interface

Pascal Fradet, Alain Girault, Gregor Goessler - INRIA Rhône-Alpes



Ce que doit être un composant

- **Documenté** : la fonction rendue par le composant **ET** son interface d'utilisation doivent être connues et décrites
- **Fiable et robuste** : le comportement doit être celui décrit (on attend aucun bogue)
- **Générique** : fonction rendue suffisamment fine et programmable (e.g. choix de paramètres) pour une adaptation rapide
- **Abstrait** : on ne doit pas avoir besoin d'accéder au code du composant

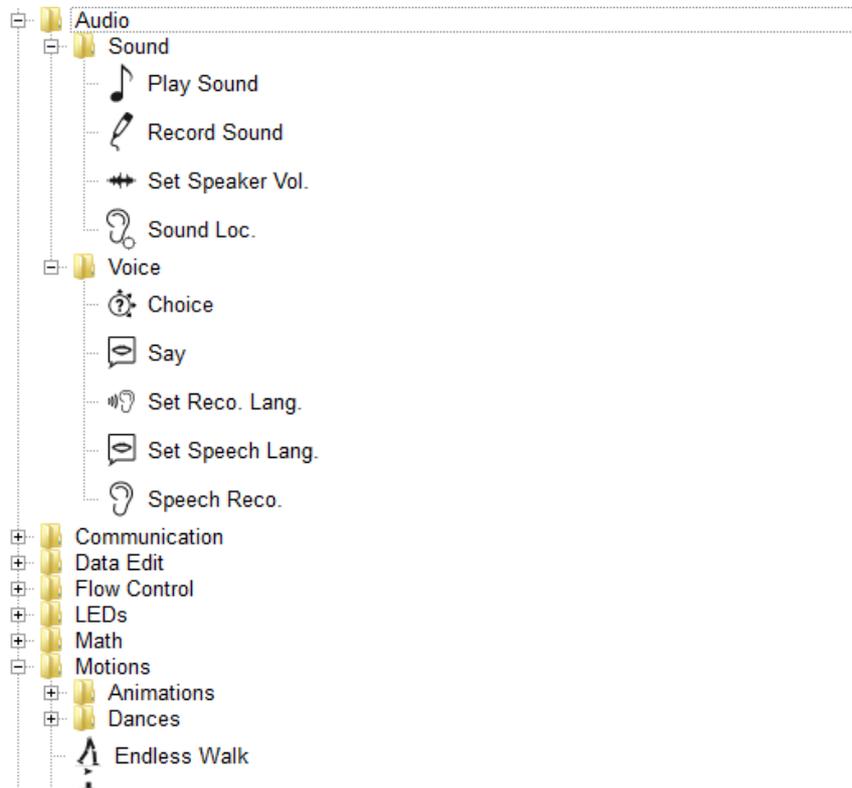
→ Il doit être Réutilisable

Les composants, 1 langage ?

- Les composants peuvent être regroupés dans un catalogue
- La programmation devient
 - Prendre un composant et le relier aux autres
 - On fait de « l'assemblage »
- Eventuellement,
 - Développement de nouveaux composants
 - Adaptation des anciens à la problématique visée

Les composants, 1 langage ?

- Robot Nao : logiciel
Choregraphe



Impact de la licence du composant

- Pour un composant libre
 - Voir schéma de Thierry Aimé,
<https://www.projet-plume.org/ressource/guide-logiciels-libres-administrations>
- Pour un composant propriétaire
 - Cf. licence. Si besoin → demander une licence particulière

Impact de la licence du composant

- Sans Copyleft → Modification et/ou Agrégation
 - Copyleft faible
 - Copyleft fort
 - Propriétaire

→ Interaction possible d'un composant sans copyleft vers un autre composant

- Copyleft faible → Modification Redistribution sous même licence
 - Agrégation
 - Distribution de l'ajout sous licence de son choix

→ Interaction possible d'un composant copyleft faible vers un autre composant



Impact de la licence du composant

- Copyleft fort Modification et/ou Agrégation → Redistribution sous même licence
- Propriétaire Modification et/ou Agrégation → Voir licence ou demander une licence particulière

→ Dans ces deux cas, l'interaction du composant avec d'autres composants peut présenter des impacts/incompatibilités sur la licence de ces composants tiers

→ se rapprocher des services valorisation



COMMENT APPLIQUER UNE LICENCE À SON CODE ?

Entête

Fichier



Comprendre le monde,
construire l'avenir®



Préconisations résumées par TGD

- Avant la diffusion du logiciel
- Attention aux cahiers des charges et aux contrats (clauses PI, licences)
- En-tête pour tous les fichiers
 - Nom du fichier, nom du logiciel, n° version
 - Copyright (©, Droits patrimoniaux), année(s),
 - Auteur(s)
 - Licence
 - Utiles : date de création, une adresse de contact

Source : T. Gomez-Diaz (CNRS - LIGM),

<http://igm.univ-mlv.fr/~teresa/logicielsLIGM/documents/Seminaires/2014janvIPSLParis.pdf>

Préconisations résumées par TGD

- Et en plus
 - Ajouter un fichier de licence au package des fichiers source avec le texte complet ou une URL
 - Indiquer les briques logicielles utilisées, leurs licences et leurs auteurs
 - Indiquer la licence du logiciel (et ses auteurs) dans la documentation et sur le site Web
 - Donner des licences aux documentations (CC ou autres...).

Source : T. Gomez-Diaz (CNRS - LIGM),

<http://igm.univ-mlv.fr/~teresa/logicielsLIGM/documents/Seminaires/2014janvIPSLParis.pdf>



Et pour les documents ?

- Tout type de licence reste possible
- Connaître les licences creative commons
 - <http://creativecommons.org/>
 - Aide à la diffusion de documents
 - Fixe un cadre de licence
 - Pas besoin de s'enregistrer ou payer : juste d'utiliser une des 6 licences
 - Outils d'aide au choix d'une licence
 - <http://creativecommons.org/choose/>



Licences Creative Commons



**Attribution
CC BY**



**Attribution - Partage
dans les Mêmes
Conditions
CC BY-SA**



**Attribution - Pas
de Modification
CC BY-ND**



**Attribution -
Pas
d'Utilisation
Commerciale
CC BY-NC**



**Attribution - Pas
d'Utilisation
Commerciale -
Partage dans les
Mêmes Conditions
CC BY-NC-SA**



**Attribution - Pas
d'Utilisation
Commerciale -
Pas de
Modification
CC BY-NC-ND**



Exemple d'application d'une Cecill

- <http://www.cecill.info/placer.fr.html> :
 1. Détermination les titulaires des droits sur le logiciel
 2. Inventaire des morceaux de code ou bibliothèques que vous utilisez (et que vous n'avez pas écrit)
 3. Inventaires des éventuelles contraintes d'origines contractuelles sur le logiciel
 4. Préparation du code source et de la documentation (avec exemples d'entête de fichiers)
 5. Organisation de la distribution
- Avant la distribution : dépôt des sources du logiciel auprès d'un tiers de séquestre

OUTILS D'ANALYSE DE CODE ET LIEN AVEC LA PI

Aide au calcul des pourcentages de contribution
Analyse des dépendances

Damian Bursztyn

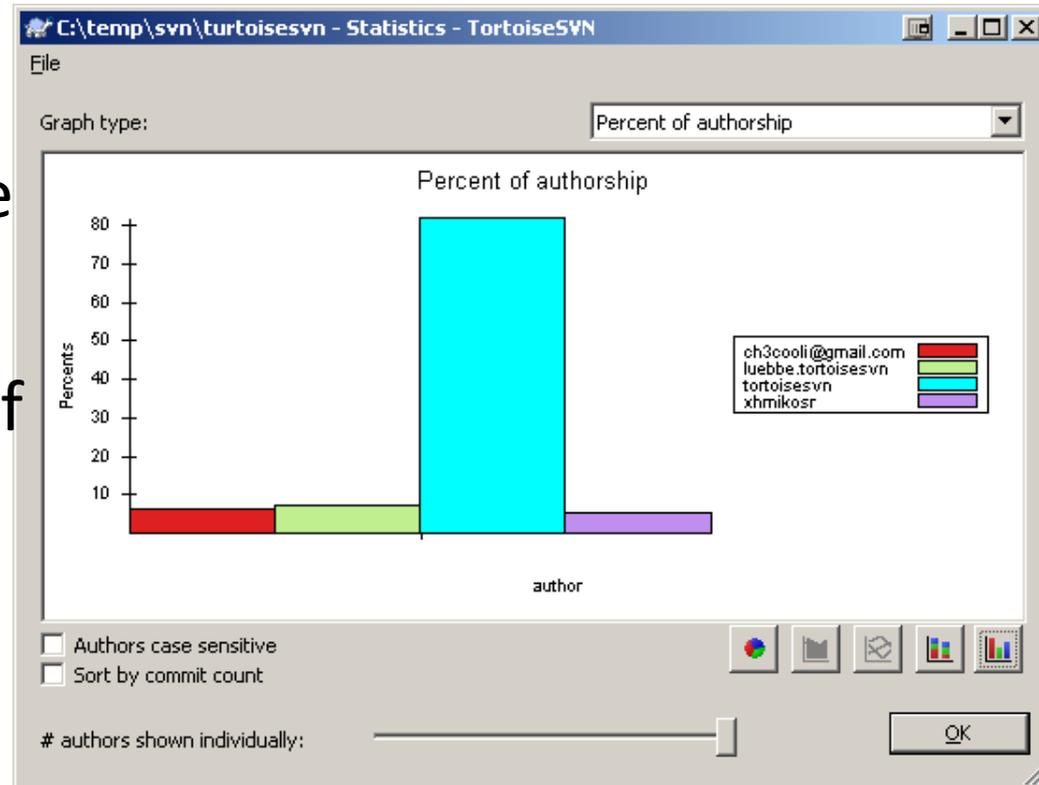
Mission doctorale Valorisation - Pôle Développement Logiciel – LRI

- Calcul des pourcentages de contribution des auteurs
- Lister toutes les dépendances / briques utilisées et s'assurer de l'absence d'incompatibilité de licences entre elles

Statistiques d'un dépôt de gestionnaire de code source

Attention : Indicateur relatif au contenu du dépôt (code, documentation, conception, architecture, dépendances, etc.)

=> Indication mais pas vérité



Exemple de statistiques via TortoiseSVN

Analyse des dépendances

Il y a t-il dans mon code des fichiers externes connus et sous licences ? Si oui lesquels et lesquelles ?

Plusieurs outils disponibles : Antepedia reporter, Blackduck, FOSSology, OSLC, ...

Au pôle Dev, deux outils sont utilisés

- **Antepedia reporter**: comparaison des fichiers à ceux d'une base de connaissances (+2 millions de projets open source licences) (→ la licence pour un fichier donné)
- **FOSSology**: scan du contenu des fichiers en recherchant des informations de licences ou copyright



Exemple d'analyse par FOSSology

Résumé de l'analyse (FOSSology dashboard)

Count	Files	License Name
3309	Show	No_license_found
3	Show	EPL-1.0
3	Show	LGPL-2.1+
2	Show	Dual-license
1	Show	LGPL
1	Show	MIT-style

Hint: Click on the license name to **highlight** where the license is found in the file listing.

6 Unique licenses

10 Licenses found

3309 Files with no licenses

3635 Files

Détail de l'analyse

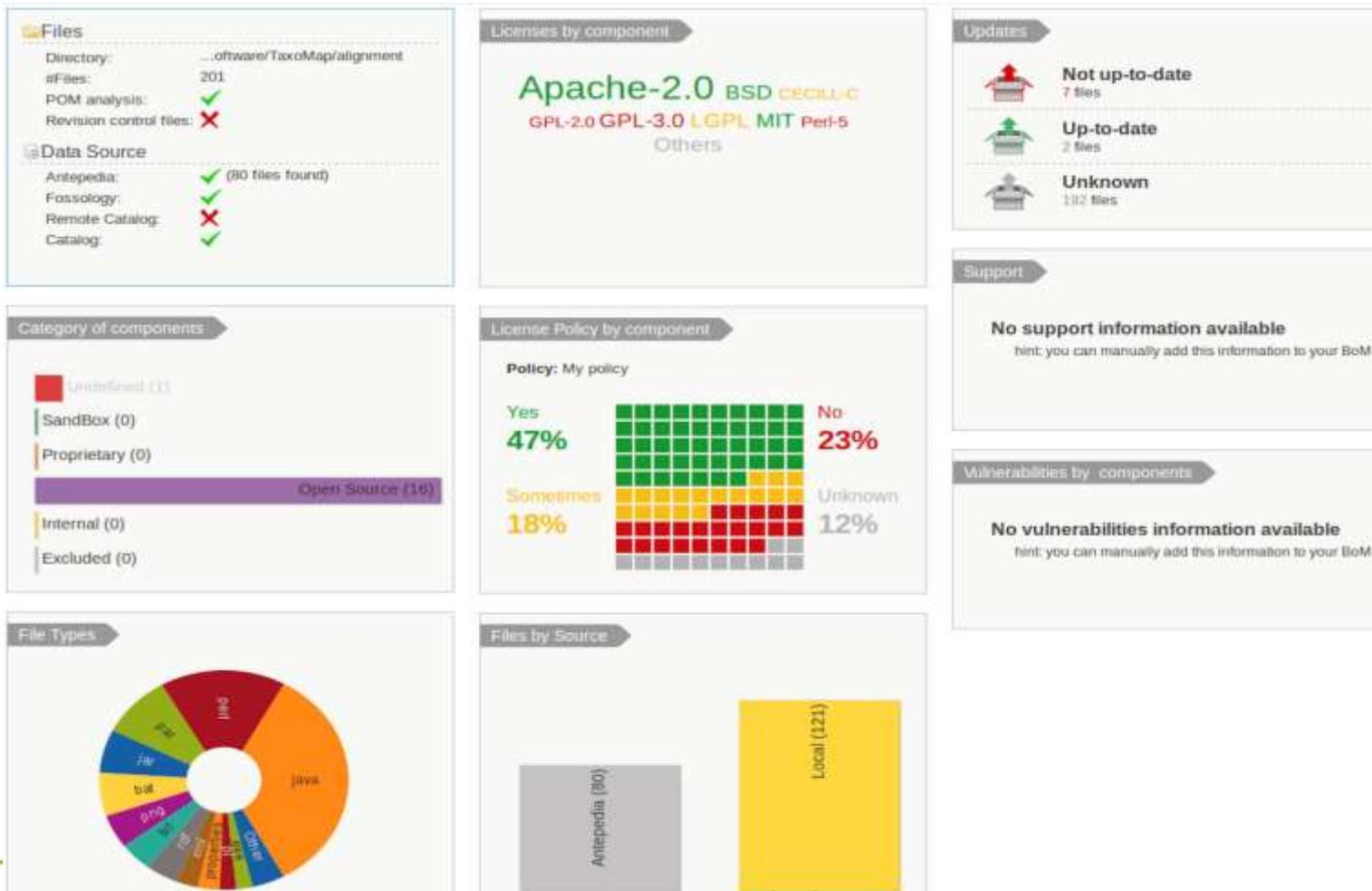
3 files found (3 unique) with license **LGPL-2.1+**

1 2 ... [Next]

File	
<p>Exclude this file type. Exclude files with license: EPL-1.0 ,LGPL-2.1+ ,MIT-style.</p> <p>1: Folder: Software Repository/ CQaPri.zip/ CQaPri/ dependencies/ org.sat4j.core.jar/ about.html</p>	EPL-1.0 ,LGPL-2.1+ ,MIT-style
<p>Exclude this file type. Exclude files with license: Dual-license ,EPL-1.0 ,LGPL-2.1+.</p> <p>2: Folder: Software Repository/ CQaPri.zip/ CQaPri/ dependencies/ org.sat4j.core.jar/ overview.html</p>	Dual-license ,EPL-1.0 ,LGPL-2.1+
<p>Exclude this file type. Exclude files with license: Dual-license ,EPL-1.0 ,LGPL-2.1+.</p> <p>3: Folder: Software Repository/ CQaPri.zip/ CQaPri/ dependencies/ org.sat4j.core.jar/ plugin.properties</p>	Dual-license ,EPL-1.0 ,LGPL-2.1+

1 2 ... [Next]

Antepedia Reporter dashboard



Détail de l'analyse

Categories		Components For Category Open Source									
Name	Version	Snippets	Snippets Found	Licenses	Antepedia Lic	Fossology Lic	Copyright	Fossology Cop	File / Path	Files	Comments
Open Source (16)											
Proprietary (0)											
Internal (0)											
Excluded (0)											
Undefined (1)											
SandBox (0)											
SLF4J LO...	1.5.8			MIT	MIT				slf4j-log4j12-1.5.8.jar	1	
SQLite JDBC	3.6.16			Apache-2.0	Apache-2.0				sqlite-jdbc-3.6.16.jar	1	
Site SVN de...				GPL-3.0	GPL-3.0	No_license...			wintreetagger.exe	1	
Textometrie				GPL-3.0	GPL-3.0				fr.par	1	
This is a m...				LGPL	LGPL				french-par-linux-3.2.bin	1	
Xerces2 Ja...	2.7.1			Apache-2.0	Apache-2.0				xercesimpl-2.7.1.jar	1	
aligne				CECIL-C	CECIL-C	No_license...			/trunk/treetagger/	34	
cognatex pr...				GPL-2.0	GPL-2.0	No_license...			/trunk/treetagger/	29	
et projects				LGPL	LGPL	No_license...			/trunk/treetagger/treet...	3	
nestedvm	1.0			Apache-2.0	Apache-2.0				nestedvm-1.0.jar	1	

File Name	Extens	Version	Snippets	Snippets F	Licenses	Antepedia l	Fossology	Copyright	Fossology	Relative Path	Size	File Comments	Comments
xercesimpl-2.7.1.jar	jar	2.7.1			Apac...	Apache...				/trunk/pom.xml/xercesimpl-2.7.1.jar	0.0...		

Possibilité d'analyse conjointe avec Fossology et d'utiliser en plus de la base Antepedia de référence, sa propre base de fichiers

Résumé de la démarche et limites

- Nous recherchons des **fichiers** de code source et / ou des **extraits** ainsi que les **dépendances** de projets qui ne sont pas licence compatible ou qui **peuvent causer un conflit de licence**.
- Une fois les résultats obtenus, nous les vérifions par une recherche sur les sites des projets → présence de faux positifs
- Il peut aussi y avoir des faux négatifs (e.g. projets open source non inclus dans la base de référence)

Nous vous remercions de votre attention