



HAL
open science

L'évolution des licences libres et open source

Benjamin Jean

► **To cite this version:**

Benjamin Jean. L'évolution des licences libres et open source : critères, finalités et complétude ?. Camille Paloque-Berges; Christophe Masutti. Histoires et cultures du libre : des logiciels partagés aux licences échangées, 13, Framasoft; Inno cube; INRIA, pp.277-340, 2013, Framabook, 978-2-9539187-9-3. halshs-02077882

HAL Id: halshs-02077882

<https://shs.hal.science/halshs-02077882>

Submitted on 9 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

L'évolution des licences libres et *open source* : critères, finalités et complétude ?

Benjamin JEAN

Au premier regard, l'intérêt d'une étude historique ayant pour objet les licences libres n'est pas évident et l'on tend généralement, avec raison, à considérer que la véritable valeur du Libre réside dans les projets qui le composent (GNU-Linux, Firefox, LibreOffice, Wikipédia, etc.). Ainsi, complexes et hétérogènes, les documents contractuels associés ne seraient qu'une formalisation nécessaire, une attache obligatoire du Libre au système légal conséquente à la juridicisation de la société actuelle.

Une telle approche nierait la place toute particulière des licences au sein du mouvement du Libre en cristallisant les tensions constitutives du système, assurant l'équilibre d'une équation aux multiples variables. Dès le départ, les licences libres se positionnent en rupture avec les usages traditionnels attirant un nombre important d'« amateurs » (au regard de l'importance qu'ils y attachent et de leur grande connaissance du sujet, la plupart étant par ailleurs non juristes), une image qui dénote avec l'austérité souvent associée aux documents contractuels et remet en cause le monopole du juriste (lui-même souvent dépassé par la complexité d'un document aussi juridique que social)... L'usage du droit s'éloigne donc de la technicité qui le caractérise aujourd'hui pour renouer avec sa mission première d'organisation des relations sociales : formalisant les règles

relatives aux rapports entre les contributeurs d'un projet et entre les communautés, assurant sécurité et une pérennité à leur collaboration. Bien qu'il soit récent, le système est solide et on ne peut que saluer le nombre réduit de contentieux (la vocation du droit étant aussi d'éviter les litiges) : moins d'une dizaine à l'échelle internationale, si on se concentre sur ceux ayant réellement une spécificité attachée au Libre et que l'on écarte les litiges potentiels très souvent réglés par le biais de transactions¹. Toujours dans ce rapport qu'entretiennent les licences libres et le droit, il est important de remarquer que les rédacteurs des licences jouissent généralement d'une position « à part », relativement privilégiée : législateurs d'une communauté réduite, ils disposent de grands pouvoirs et donc de grandes responsabilités.

Cette situation n'est cependant que la résultante d'une lente évolution. Même s'ils ne le furent formellement que quelques années plus tard, les logiciels libres sont conceptuellement nés avec l'informatique : dès la massification des ordinateurs, les utilisateurs ont commencé à partager leurs programmes et à contribuer sur ceux de leurs pairs (de fait, la plupart des systèmes d'exploitation étaient ainsi libres jusqu'aux années 1970). Les logiciels étaient alors dénués de toute valeur économique intrinsèque et développés collectivement par un nombre restreint de personnes issues de la Recherche, de la Défense et des quelques entreprises vendant du matériel informatique. Aucune propriété n'était, pour ainsi dire, revendiquée sur ces derniers. Néanmoins, la démocratisation de l'informatique a favorisé le développement d'un marché du logiciel qui a motivé la transformation de ces produits immatériels en produits de consommation, à l'instar des livres ou des disques, par la reconnaissance d'un droit de propriété « intellectuelle » (en l'occurrence le droit d'auteur). Le Copyright Office américain ouvrit l'enregistrement (nécessaire pour bénéficier du copyright) d'œuvres logicielles dès les années 1960, sans préjuger du bénéfice de la protection par le copyright alors très incertaine. Mais ce n'est qu'en 1976, au travers du Copyright Act, et plus spécifiquement en 1980 par le Computer Software Copyright Act, qu'une telle protection fut reconnue par la Loi². Ce fut alors l'âge d'or des éditeurs de logiciels (qui dé-

1. L'initiative *gpl-violations.org* évalue à plus de 150 le nombre de dossiers traités en dehors des tribunaux.

2. La situation fut relativement similaire en France avec une doctrine qui accepta progressivement l'idée d'une protection du logiciel au même titre qu'une œuvre littéraire et la

mocratisèrent le *shrink wrap*, contrat associé à des logiciels vendus « sur étagère », et accepté par l'ouverture du produit), sur lequel reposent toujours des sociétés telles que Microsoft, Oracle, SAP, etc. Parallèlement au développement de cette industrie (intimement liée à l'extension des droits qu'elle entretient par un lobbying intensif), certains logiciels continuèrent néanmoins à être créés collectivement toujours au sein de la recherche mais aussi parmi une communauté croissante de hackers. Ces derniers, n'ayant pas les moyens ou ne souhaitant pas acheter de telles licences (alors généralement négociées par et pour les seules universités¹), mirent en place des règles de non-appropriation (non exclusivité) au travers des licences libres, rétablissant une confiance nécessaire au développement d'un projet commun.

S'appuyant sur le droit d'auteur (et sur un nombre croissant de nouveaux droits exclusifs) qui joue un rôle important et structurant, le Libre s'est étendu dans le secteur du logiciel et au-delà, en introduisant lorsqu'il le fallait de nouvelles problématiques, spécifiques ou générales à ces droits. Plus précisément, le système des licences libres a été bâti sur celui du copyright américain. Celui-ci possède en effet un statut de *license* par lequel le titulaire de droits renonce publiquement à certains de ces droits. Ces *licences* sont des engagements unilatéraux de détenteurs de droits soumis à un régime fédéral dans le cadre des lois sur le copyright alors que le *contract* (similaire à notre notion française) est soumis au régime général des contrats. Les conséquences pratiques sont de plusieurs ordres : concernant le formalisme requis (une licence ne pouvant être contestée par le bénéficiaire qui se retrouverait sans droit, elles sont généralement beaucoup plus courtes que les contrats), la juridiction compétente ou la loi applicable (régime fédéral ou loi d'un État) ou encore la réparation attachée à leur violation (interdiction d'exploiter ou exécution forcée en cas de licences, simple dédommagement en cas de contrat). C'est pour toutes ces raisons que certains revendiquent la qualification de

loi du 3 juillet 1985 (complétée plus tard par la directive 91/250/CEE du 14 mai 1991) qui entérina cette position.

1. Unix était ainsi disponible pour les universités à 150\$ et à 20 000\$ pour les autres. Voir S. Holmgren, « Network UNIX », [RFC] : <http://tools.ietf.org/html/rfc681>, consulté le 8 octobre 2012.

*licenses*¹, même si un tel statut n'existe pas dans notre régime de droit latin (ni même, plus largement, en Europe)².

Près de trente années plus tard, les licences libres font l'objet d'une véritable attention, de la part tant du monde professionnel que des utilisateurs et du secteur public. Elles remettent en cause un certain nombre de modèles économiques, de manière d'autant plus forte que de réelles politiques publiques commencent à se formaliser en faveur du libre (au point où la pratique du logiciel libre dans certains marchés publics devient le principe) – ajoutons qu'en réponse aux plaintes de certains éditeurs traditionnels, menacés par la généralisation du logiciel libre, plusieurs décisions sont venues reconnaître la compatibilité (et les bienfaits) du logiciel libre au regard du droit de la concurrence³. Grâce aux efforts d'une multitude d'acteurs, cette composante contractuelle d'un système alternatif a réussi à intégrer au fur et à mesure les évolutions du système classique afin de s'appuyer sur ces derniers – voire les contrer lorsqu'ils faisaient obstacle aux objectifs initiaux.

Pour autant, l'objet de cet article n'est pas tant d'étudier les licences libres comme objets juridiques⁴, mais de comprendre, au travers d'une perspective historique, le fonctionnement de leur processus d'édiction (la forme) ainsi que l'évolution de leur contenu (le fond). Ainsi, complétant les analyses plus juridiques qui évaluent chaque licence au regard d'un droit positif (ensemble de règles juridiques en vigueur à un moment

1. À noter que les deux qualifications (*contract* et *license*) ont été retenues par les juges de l'affaire *Jacobsen v. Katzner*, United States Court of Appeals for the Federal Circuit, 18/08/2008 (<http://www.cafc.uscourts.gov/images/stories/opinions-orders/08-1001.pdf>). En Europe, les décisions allemandes (en 2004 contre Sitecom Deutschland, en 2006 contre D-Link Deutschland ou encore contre Skype en 2008) reposent toutes sur la reconnaissance de la GNU GPL comme un contrat valide permettant la sanction du non-respect du formalisme qu'elle impose.

2. La distinction entre contrat et *license* n'existe pas en droit romain. Un juriste français ne verrait ainsi dans la licence qu'une offre d'un auteur (ou titulaire de droits) se transformant en contrat dès l'acceptation par l'utilisateur. L'article 1101 du Code civil définit le contrat comme « une convention par laquelle une ou plusieurs personnes s'obligent, envers une ou plusieurs autres, à donner, à faire ou à ne pas faire quelque chose. »

3. Voir *Court of Appeals for the Seventh Circuit*, *Wallace v. International Business Machines Corp. et al.* (2006) ; *Cour de Munich*, *Welte vs. Skype Technologies SA* (2007) ; *Décision numéro 350431 du Conseil d'État français* (2011) ou encore la *Décision de la Cour constitutionnelle italienne* (2011).

4. Sur le sujet, voir Mélanie CLÉMENT-FONTAINE, « Les oeuvres libres », Thèse de doctorat, Université de Montpellier, 2006, p. 99 ; Benjamin JEAN, *Option libre. Du bon usage des licences libres*, Paris : Framasoft/Framabook, 2012.

donné), il s'agira ici d'apporter un éclairage utile à la compréhension des clauses et des mécaniques des licences libres.

Notre étude portera donc sur l'évolution des licences libres et *open source*¹, de la première formalisation de la notion de logiciel libre au début des années 1980 à nos jours, en portant une attention particulière aux licences structurantes du mouvement. Eu égard à l'objet, les mutations évoquées seront essentiellement juridiques, mais nous adopterons une vision pratique tout en gardant à l'esprit les objectifs initiaux (avoués ou non) afin de situer l'évolution des licences dans leur contexte. Une attention particulière sera apportée aux finalités des licences, celles-ci permettant d'en comprendre l'apparition de nouvelles, leur renforcement pour répondre aux besoins et enfin leur structuration dans une vision globale supplantant chaque vision individuelle.

1. La distribution des cartes : le processus d'apparition des licences libres

Un processus itératif. Il n'est pas étonnant, au regard de leur objet, que les licences libres soient conçues dans le cadre d'une démarche itérative et progressive. Elles évoluent au fur et à mesure que la société se transforme et suivent plus ou moins rapidement les évolutions du droit et de la technique. Une première conceptualisation des licences trouve son origine dans les vingt dernières années du XX^e siècle (1.1), entraînant l'arrivée d'une vague de nouvelles licences les dix années suivantes (1.2).

1.1 La conceptualisation des licences libres(1984-1998)

1.1.1 Les premiers logiciels « libérés » par leur licence (1984-1989)

Entre domaine public et licence libre. Conditionné au respect d'un certain formalisme, le bénéfice du copyright américain n'était pas automatique au début des années 1980. Ainsi, de nombreux projets logiciels furent codéveloppés, de manière collaborative et ouverte, avant que ne soit

1. On oppose aujourd'hui licences libres à logiciels propriétaires de la même façon que l'on oppose *open source* à *closed source*, sans que les licences libres et *open source* soient véritablement superposables. Voir *infra*.

expressément apposée une licence explicitant les droits accordés. Ce n'est que plus tard et en réaction à l'élargissement des communautés d'utilisateurs et contributeurs qu'il fut jugé utile de formaliser les interdictions associées à la réutilisation du code, dans un premier temps, et de formaliser plus précisément les droits accordés, dans un second temps. Cet éloignement du domaine public se fit avec douceur. L'exemple de Tex, système logiciel de composition de documents créé à partir de 1977, est symptomatique. Selon les dires de son auteur Donald Knuth, Tex était ainsi dans le domaine public avec comme seule contrainte (ajouté en 1982 lors de la sortie de TeX82) que « TEX » ou « METAFONT » ne soient pas réutilisés dans des versions dérivées du logiciel¹. Plus restrictif, Larry Wall (futur développeur du langage de programmation Perl) autorisait sur le logiciel *trn* tout usage dès lors que l'utilisateur ne prétendait pas être l'auteur du code ou qu'il ne cherchait pas à le vendre.

L'apparition des premières licences *copyleft*. Auteur du logiciel GNU Emacs, Richard Stallman lui associa en 1985² une licence spécifique intitulée la GNU Emacs General Public License : sensiblement plus longue que les licences précédentes, elle définissait avec précision le cadre permettant la réutilisation de la licence et obligeait à conserver la licence sur toutes les versions dérivées (ou contenant une partie) du logiciel. Cette protection découlait essentiellement de déboires que connut Stallman lorsqu'il souhaita réutiliser une version modifiée de son logiciel Emacs (Gosling Emacs) et qu'il se vit opposer le copyright de l'auteur de cette dernière – situation qui le contraignit finalement à tout réécrire pour ne pas être inquiété par la société qui avait acheté les droits. Conditionnant la licence (au sens de permission) à une certaine réciprocité, il se servit ainsi de ses propres droits d'auteur pour limiter toute réappropriation par un autre. Ressentant comme une menace la combinaison de code libre et non libre, cette licence formalisa la notion de *copyleft* par laquelle l'ajout ou la jonction d'un élément quelconque à un programme sous une telle licence doit se faire de sorte que la version résultant de cette opération soit régie par les mêmes termes (et donc également libre).

1. Donald E. KNUTH, « Computers and Typesettings », dans : *TUGboat* 7, 1986, p. 95-98.

2. Sam WILLIAMS, Richard STALLMAN et Christophe MASUTTI, *Richard Stallman et la révolution du logiciel libre. Une biographie autorisée*, Paris : Eyrolles, 2010, p. 126.

Les premiers logiciels universitaires. Parallèlement, et en réaction aux premiers courants de l'appropriation logicielle, les universités s'émancipèrent progressivement des éditeurs traditionnels tout en optimisant la diffusion (et la réutilisation) de leur logiciel par l'usage de licences libres. Elles firent ainsi du code la même chose que ce qu'elles faisaient de la recherche : le rendre disponible et réutilisable par un maximum de personnes. Cela fut réalisé dès les années 1986 pour le MIT (lors de la sortie de la version X10R3 du logiciel X Window System) ou un peu plus tard pour Berkeley qui devait se séparer de toutes les contributions d'AT&T afin de distribuer son système d'exploitation librement (BSD-Tahoe fut publié en 1988 et Net/1 en 1989). Répondant ainsi aux besoins qui lui avaient été exprimés (Stallman joua notamment un rôle important dans cette évangélisation), le monde académique et universitaire américain offrit ainsi à la société de nombreux logiciels qui composent aujourd'hui encore l'infrastructure d'Internet (on peut aussi citer le système de nom de domaine BIND, le protocole TCP/IP et Sendmail). Tous ces logiciels eurent pour point commun d'être diffusés sous des licences qui étaient courtes, organisant une cession des droits très large tout en excluant toute garantie ou responsabilité. La principale condition concernait la mention de l'origine du code, dans des termes variables. Par commodité, ces licences ont rapidement été nommées en référence aux projets éponymes : BSD license (plus tard dite *4-clause license* ou *original BSD License*) et MIT license.

La formalisation de la Free Software Definition (FSD). Fort du succès de GNU Emacs, Stallman fonda en octobre 1985 la Free Software Foundation (FSF), organisation américaine à but non lucratif, pour aider au financement du projet GNU et de la communauté du logiciel libre¹. C'est dans ce cadre que fut rédigée la Free Software Definition en 1986², cette dernière définissant comme libre tout logiciel pour lequel l'utilisateur bénéficiait de trois libertés : la liberté d'étudier le fonctionnement du programme, et de l'adapter à ses besoins (l'accès au code source et donc nécessaire) (liberté 1) ; la liberté de redistribuer des copies du logiciel (liberté 2) ; la liberté d'améliorer le programme et de publier ses améliorations.

1. Le projet GNU est officiellement annoncé par un message de Richard Stallman en date du 27 septembre 1983 ayant comme sujet « *new Unix implementation* ».

2. Richard M. STALLMAN, « What is the Free Software Foundation ? », dans : *GNU's Bulletin* 1 (1986), p. 8.

rations (là encore, l'accès au code source est nécessaire) (liberté 3). La liberté 0, celle d'exécuter le programme pour tous les usages, n'apparut que plus tardivement, aux alentours des années 1996, lorsque Stallman s'aperçut qu'elle pouvait ne pas être déduite des trois autres. C'est au regard de cette définition finaliste (recherchant la liberté de l'utilisateur final) que furent comparées les licences *copyleft* et les licences permissives : les licences permissives assurent ces libertés pour les premiers utilisateurs du logiciel seulement alors que les licences *copyleft* les rendent persistantes au profit de tous les utilisateurs (premiers et subséquents) du logiciel¹, faisant reposer cette charge sur les licenciés eux-mêmes (les astreignant par exemple au maintien de la même licence sur leurs créations dérivées)². Depuis lors, ces deux courants cohabitèrent sans heurt majeur.

De la GNU Emacs Public License à la GNU General Public License.

La GNU Emacs General Public License fut la première de la longue série des licences *copyleft* rédigée par la FSF : suivirent ensuite des licences similaires associées à la sortie du GNU Debugger (GDB) en 1986, du GNU Compiler Collection (GCC) en 1987, du GNU Bison en 1988, etc. Le texte initial fut adapté à chaque logiciel et évolua au cours de ces quelques années en fonction des retours au sein de la communauté : la seconde version de la GNU Emacs GPL introduisit par exemple en 1988 la notion de « sphère privée » (les obligations de la licence ne doivent être respectées qu'à partir du moment où le logiciel est distribué, sortant de cette sphère privée). D'un statut simplement déclaratif, la licence était devenue l'instrument qui définissait les « règles de jeux » nécessaires pour encadrer la collaboration au sein d'une communauté d'auteurs, utilisateurs et contributeurs. Cette traduction juridique des libertés définies prit la forme de cessions de droits en intégrant certaines obligations pratiques nécessaires pour que l'utilisateur jouisse effectivement de ces libertés (accès au code source dans sa forme préférentielle, sa documentation ou, dans une certaine mesure, la qualité du code). Ce premier travail reçut un accueil très favorable de la communauté que réunissait le projet GNU et plus largement (attentif au travail mené par la FSF, certains projets commencèrent ainsi à modifier ces licences afin de les utiliser à l'intention de leur propre

1. Richard M. STALLMAN, « The GNU Operating System and the Free Software Movement », dans : *Open Sources : Voices from the Open Source Revolution*, sous la dir. de Chris DiBONA, Sam OCKMAN et Mark STONE, Sebastopol, CA : O'Reilly, 1999.

2. JEAN, *op. cit.*, p. 201.

logiciel¹ et au-delà de la FSF – telle la Nethack General Public License dès 1989).

Genèse de la GNU GPL. Évoquée dès 1988, ce n'est qu'au début de l'année suivante (le 25 février 1989) qu'apparut la licence générique du projet GNU sous le nom de GNU General Public License (GNU GPL). Il s'agissait par ce texte de répondre aux besoins du projet GNU (notamment en termes d'harmonisation) et plus largement (puisque c'est aussi l'époque où d'autres communautés indépendantes commençaient à se former) avec pour volonté de favoriser la reprise de code entre les multiples projets libres qui se créaient. Cette décision fut stratégique vis-à-vis de l'utilisation et de la diffusion de la licence, la rendant rapidement centrale dans le paysage du Libre. Elle devenait ainsi plus simple à faire évoluer (puisque le rédacteur n'avait alors plus qu'un seul texte à modifier), plus simple aussi à comprendre (car plus utilisée – même si une FAQ et le *GNU manifesto*, datant de 1985, furent ajoutés pour mettre un terme aux divergences naissantes). Par la même occasion, un travail complémentaire fut aussi fourni afin de perfectionner la licence : en rallongeant le préambule de la licence ; en précisant les notions techniques (par exemple le code source qui fut défini comme la forme de création préférée pour faire des modifications sur celle-ci) ; en précisant les contraintes attachées au *copyleft* (notamment en formalisant l'interdiction d'ajouter des restrictions lors de la rediffusion). De multiples facteurs peuvent certainement expliquer son adoption rapide et large, mais la confiance particulière dont bénéficiait son rédacteur est certainement le plus important. Il n'est ainsi pas exagéré de considérer que ce que fit la FSF n'était à portée d'aucune autre organisation. Stallman, et avec lui la FSF, furent ainsi précurseurs dans la réunion de ce qui s'avéra plus tard être le terreau idéal pour le développement de logiciels libres : une communauté importante, des licences libres « génériques » et les bases d'une première gouvernance.

Une évolution parallèle à celle du copyright. L'histoire est une succession de coïncidences et d'intuitions. L'expansion d'Internet et du Web fut ainsi un accélérateur majeur dans la constitution des communautés de logiciels libres, de la même façon que le contexte politique est certainement

1. Stallman, et la FSF après lui, ont toujours été très permissifs à l'égard de leur licence en autorisant expressément toute réutilisation des clauses dans d'autres contrats dès lors que le nom et le préambule restaient inchangés.

corrélé à la rédaction de ces licences durant les années 1990. En effet, ces dates correspondent à l'époque où le régime du copyright américain subit une véritable rupture caractérisée par la ratification par les États-Unis de la convention de Berne le 16 novembre 1988¹, entraînant l'année suivante l'abandon du système du dépôt obligatoire au profit du régime de protection automatique (pratiqué depuis longtemps en Europe). Alors que la publication sans mention légale entraînait auparavant l'abandon de tout bénéfice du régime du copyright (l'enregistrement étant la condition pour défendre ses droits), l'ouverture et le partage de code devinrent dès cette date conditionnés à un comportement actif des auteurs se matérialisant par l'usage d'une licence (l'absence de licence étant un obstacle au partage et à la diffusion du code). Chacun put alors faire preuve d'originalité dans la rédaction de sa propre licence là où un comportement matériel (l'absence de mentions légales) suffisait auparavant. Il serait à cet égard intéressant de se demander si le logiciel libre aurait connu le même développement en l'absence de ce renversement de paradigme.

1.1.2 L'apparition des premières variations (1991-1998)

La réutilisation des licences. En raison de la grande liberté qu'elles accordaient, les premières licences BSD ou MIT se sont rapidement retrouvées dans une multitude de formes, fonctions de leurs utilisations : on repère ainsi des premières variations, qu'il s'agisse du projet Expat pour la licence MIT (par souci de précision, on distingue aujourd'hui chaque version de la licence initialement utilisée par le MIT : l'Expat License, la X10 et X11 License, etc.) ou des projets sous licence BSD (de la même façon, on distingue aujourd'hui les licences *BSD modified* ou *BSD simplified*). Par exemple, et pour faire un lien avec l'évolution d'Internet, Internet Systems Consortium (qui développe et maintient un certain nombre de projets essentiels au réseau : BIND, OpenReg, DHCP, etc.) distribua ses logiciels sous une licence ISC dérivée de la licence *BSD modified*, en s'appuyant sur les mécanismes introduits par la convention de Berne. Il s'agit d'une version simplifiée de cette dernière, et qui devint en 2003 la

1. Et son entrée en application aux États-Unis le 1^{er} mars 1989 par la « Berne Convention Implementation Act ». Voir *U.S. Copyright Office – Information Circular*, Copyright Notice 23.0811. Notons le vote en 1990 du Computer Software Rental Amendments Act, renforçant les prérogatives des ayants droit.

licence du projet FreeBSD. Cette tendance se confirma dans les années qui suivirent par l'adoption de telles licences par un certain nombre de projets sous le même nom où en s'appropriant parfois la licence (citons pour l'exemple la CWI License Agreement utilisée sur Python dès 1991 et directement inspirée de la licence type MIT, ou la Cryptix General License utilisée dès 1995 sur le projet Cryptix et directement inspirée de la BSD).

La déclinaison des licences GNU. S'appuyant sur les nombreux retours de la communauté croissante des hackers ayant adopté cette licence, une seconde version de la GNU GPL fut publiée en 1991 (introduisant pour la première fois une clause relative aux brevets) aux côtés d'une licence moins contraignante destinée à certains programmes qui gagnaient à être diffusés et utilisés en dépit d'un environnement non entièrement ouvert : la GNU Library General Public License (GNU LGPL). La GNU LGPL (immédiatement en version 2 aux fins d'harmonisation) autorise expressément la liaison de logiciels non libres avec la bibliothèque distribuée sous cette licence. À noter cependant que ces concessions n'étaient pas une renonciation aux libertés telles qu'elles avaient été formalisées au travers de la GNU GPL, mais un choix (initialement conçu pour la bibliothèque C) destiné à favoriser l'immixtion de logiciels libres stratégiques dans l'écosystème des logiciels propriétaires (dans les situations où, en présence d'alternatives, la licence aurait limité l'usage de ces logiciels) et, surtout, permettant aux éditeurs de développer des applications pour le système d'exploitation GNU sans les contraindre au choix de la licence GNU GPL (et donc rendre plus attirant ce dernier pour les développeurs)¹. Plus tard, en 1999, afin de freiner l'adoption systématique de la GNU LGPL pour les bibliothèques logicielles, la FSF modifia son nom au profit de GNU Lesser General Public License (la version devint alors 2.1) : une modification conforme aux objectifs qui avaient mené à la rédaction de la GNU LGPL, celle-ci n'étant stratégiquement intéressante qu'à l'égard de projets pour lesquels de grandes contraintes auraient eu

1. Par ailleurs, cette nouvelle licence répondait à la ligne de conduite que s'était fixée Stallman dès sa première licence libre. La GNU Emacs GPL précisant en effet dans son article 5 : « *If you wish to incorporate parts of GNU Emacs into other free programs whose distribution conditions are different, write to the Free Software Foundation. We have not yet worked out a simple rule that can be stated here, but we will often permit this. We will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software.* »

des effets trop néfastes sur leur diffusion (pour les autres, la GNU GPL devait être privilégiée). Les licences GNU GPL et GNU LGPL composèrent ainsi la première famille de licences : cohérente, complémentaire et solide juridiquement, celle-ci assurait la constitution d'un « pot commun » dans lequel toute personne était conviée à piocher dès lors qu'elle était prête à y reverser ses propres contributions. Encore peu juridiques¹, elles conservaient un langage très proche de celui des développeurs et étaient empreintes d'une intention forte qui se traduisait souvent par des termes complexes à définir (et donc à qualifier juridiquement).

Un système relativement complet. Le cumul des premières licences *copyleft* et des licences permissives donna un premier visage cohérent au système des licences libres. Ainsi, soit le contributeur souhaitait contribuer à un pot commun en usant de l'une des licences rédigées par la FSF soit il optait pour un partage plus large encore et se tournait vers l'une des licences rédigées par les universités. Dans les deux cas, les licences existantes pouvaient être utilisées en l'état – avec, bien heureusement pour les licences GNU, une aide communautaire concernant leur compréhension et usage. Les contraintes pour les développeurs étaient donc de trois ordres : tout reverser s'ils utilisaient un programme sous GNU GPL ; ne reverser que les modifications apportées à la bibliothèque sous GNU LGPL si leurs programmes dépendaient de celle-ci ; ou se limiter aux seules attributions obligatoires si le programme était sous une licence permissive (Apache, BSD ou MIT).

L'adoption de la GNU GPL par d'autres projets : Ghostscript, Perl et Linux. Qu'il s'agisse de L. Peter Deutsch, de Larry Wall ou Linus Torvald, ces trois développeurs de renom étaient convaincus de l'intérêt du Libre pour leurs projets (respectivement Ghostscript, Perl et Linux) : en termes de méthode de développement, de diffusion et extension. Approché et accompagné à ces débuts par Richard Stallman, L. Peter Deutsch avait logiquement choisi de diffuser Ghostscript sous GNU GPL dès son lancement dans les années 1988-1989. Une fois leurs projets matures, Larry Wall et Linus Torvald troquèrent plus tard leur propre licence par la GNU GPL : en 1989 pour le langage Perl (datant de 1987, et auparavant sous une licence interdisant de revendre le logiciel) et en 1992 pour

1. L'avocat et professeur de droit Eben Moglen ne rejoignit néanmoins cette cause qu'à partir des années 1993.

le noyau Linux (alors en version 0.99 – il était auparavant distribué sous une licence obligeant la fourniture des sources et interdisant l'usage commercial). Ces trois adoptions constituèrent un signal fort : représentatif de l'intérêt que présentait la GNU GPL à l'égard des projets collaboratifs, renforçant celui-ci du fait de leur légitimité propre. Cependant, elles illustrèrent tout aussi bien les premières frictions face aux positions très fortes que traduisait cette licence. Ainsi, Larry Wall et Linus Torvald ajoutèrent chacun une interprétation à la GNU GPL pour relativiser certains de ses effets en matière de combinaison de logiciels (cf. infra, section 2.1.1). Plus encore, Larry Wall rédigea en mars 1991 une nouvelle licence qu'il intitula Artistic License et L. Peter Deutsch créa en 1994 l'Aladdin Free Public License : l'Artistic License fut publiée à l'occasion de la sortie de la version 4 du langage Perl et s'ajouta à la licence GNU GPL (donc sous double licence) afin de « rassurer les industries du logiciel »¹ ; l'Aladdin Free Public License fut utilisée par la société Aladdin Entreprise pour interdire certains usages commerciaux, en complément de la version GNU Ghostscript (en retard de deux versions sur l'Aladdin Ghostscript) toujours diffusée sous licence GNU GPL.

De nouvelles licences permissives. En raison de leur simplicité de prise en main, de nombreuses nouvelles licences permissives apparurent durant les années 1990. La plupart étaient des variations plus ou moins fidèles des licences publiées par le MIT et Berkeley, mais d'autres se démarquèrent. La licence Zlib est apparue en 1995 en ajoutant quelques exigences quant à l'identification des modifications apportées. L'Apache Group, à l'origine du serveur web éponyme, basa ses développements sur le serveur httpd de la NCSA (sous une licence très libérale de type « *This code is in the public domain, and give us credit* ») et adopta en 1995 une première version de la licence Apache s'inspirant de la licence BSD originale (changeant le nom de l'organisation et ajoutant l'interdiction d'utiliser certains termes dans les produits dérivés).

1. Il tentait ainsi de répondre à un certain nombre de problèmes identifiés : en réduisant le *copyleft* (dès lors que le programme modifié n'entraînait pas en conflit avec le programme initial), en donnant la possibilité d'être partie d'un logiciel plus large (sans impact pour celui-ci quant à la licence) ; en supprimant tout impact du fait de codes laissés dans les logiciels générés.

Les complications de la clause de paternité. La première version de la licence BSD (dite *4-clause BSD*) contenait une clause qui devint rapidement polémique :

Tous les documents publicitaires mentionnant les caractéristiques ou l'utilisation de ce logiciel doivent afficher la mention suivante : « Ce produit contient un logiciel développé par l'Université de Californie, Berkeley et ses collaborateurs. »¹

La contrainte formalisée par cette troisième clause imposait donc un affichage proéminent qui assurait d'une part le respect la paternité de l'université de Berkeley et participait d'autre part à une valorisation importante de ses travaux grâce à cette « communication forcée ». Néanmoins, cette clause qui parut initialement anodine – voire légitime – devint rapidement une plaie lors de la réappropriation de cette licence pour d'autres logiciels : chacun requérant une mention spécifique lors de l'usage de sa propre contribution, il s'en est suivi des situations où le formalisme devint beaucoup trop lourd pour être raisonnable. S'appuyant sur des exemples de logiciels nécessitant la mention de plus d'une quarantaine d'auteurs, Stallman fit un lobbying important pour que cette clause (qu'il qualifia d'« odieuse clause ») soit supprimée. William Hoskins, directeur de l'Office of Technology Licensing for UC Berkeley fut sensible à ces arguments et publia une nouvelle version de la BSD (*new BSD* ou *modified BSD*) amputée de cette clause le 22 juin 1999. D'autres licences suivirent le mouvement : la Zope Public License (publiée en 1999) supprima une clause similaire en 2001 lors de son passage en version 2 ; la licence Apache fit de même en 2000 (passant ainsi en version 1.1). Malgré cela, de nombreux composants restèrent soumis à ces termes initiaux, obligeant le maintien de cette obligation au sein des programmes les réutilisant, telle la Sendmail License (2004) qui mêla un *copyleft* et des engagements hérités de la BSD originale (citation de l'université de Californie) au travers d'un article 5 qui y fait expressément référence ; ou encore la licence OpenSSL (1998) qui maintient les obligations héritées de la licence Apache initiale à laquelle s'ajoutait la licence SSLeay (*4-clause BSD*) sous la forme de ce qui fut appelé *dual-license* (qui, dans ce contexte, signifiait que l'utilisateur devait respecter les deux licences à la fois).

1. Traduction libre.

1.2 La ruée vers l'*open source* et la multiplication des licences (1998-2007)

L'apparition de nouvelles licences. Le XX^e siècle déboucha sur la reconnaissance du concept de logiciel libre en termes de réalité économique et technique. Il marqua l'entrée dans une courbe de croissance continue. L'arrivée de nouvelles licences fut peut être le corollaire imprévu, induit par la reconnaissance de nouveaux besoins, l'arrivée de nouveaux acteurs cherchant à se rattacher au mouvement en construction ou encore l'évolution des projets initiés dans les années 1980. Le premier lot de licences témoignait d'une faible présence européenne ou asiatique sur le sujet (même si de nombreux projets avaient déjà une dimension mondiale). Le début du XXI^e siècle, quant à lui, introduisit la diversité avec l'apparition de nouvelles licences, créées par des personnes publiques ou privées, plus seulement américaines, et dépassant le seul secteur du logiciel. Ce fut ainsi à la fois une réussite en termes d'expansion du phénomène (alors qu'à la même époque était voté le Digital Millennium Copyright Act¹ aux États-Unis), mais aussi un échec en termes d'harmonisation puisqu'on ne compta plus le nombre de nouvelles licences qui vinrent compléter les licences existantes (elles-mêmes souvent modifiées).

Des licences parfois inadaptées. Rapidement, les licences furent confrontées à leurs propres limites : l'effet relatif et la prévisibilité des contrats. L'effet relatif est la limite la plus importante du système : seules les personnes qui sont parties aux contrats sont engagées par ce dernier dans les limites de ce qu'elles ont convenu, de telle sorte que, sauf stipulations contraires, le rédacteur du contrat est tiers à cette relation et ne peut imposer d'aucune manière certains usages ou même certaines interprétations. Même s'ils peuvent parfaitement être utilisés pour encadrer des relations sur du très long terme, les contrats ne restent pertinents que s'ils réussissent à encadrer la relation telle qu'elle avait été prévue initialement, les procédures de révisions et d'adaptation des contrats étant alors généralement fastidieuses. Ainsi, la généricité des licences initiales, si elle

1. Votée en 1998, cette loi introduisit notamment la protection juridique au profit des mesures techniques de protections. Son équivalence européenne est l'EUCD (directive 2001/29/CE du Parlement européen et du Conseil du 22 mai 2001) et transposée en France par la loi DADVSI (Loi n°2006-961 du 1^{er} août 2006 relative au droit d'auteur et aux droits voisins dans la société de l'information).

a sans doute facilité leur réutilisation, n'a pas suffi pour empêcher l'apparition de nouvelles licences destinées à répondre à des situations particulières (communautés, domaine, contributeurs types, langage, voire objet de la licence lorsqu'on dépasse la sphère du logiciel). Fort heureusement, un rôle structurant essentiel à l'écosystème fut assuré par les organisations qui régulèrent et accompagnèrent l'apparition et l'usage des différentes licences libres. Ces organisations furent d'autant plus importantes dans la phase suivante que l'intérêt (et l'utilité) de nouvelles licences est très vite apparu comme relevant de deux conditions : 1) provenir d'une organisation bénéficiant d'une certaine légitimité (ou créées à cette fin et héritant de la légitimité de ses membres) et 2) répondre aux besoins d'une communauté (structurée ou non).

Deux mouvements parallèles caractérisent cette décennie : la structuration du secteur du logiciel avec l'industrialisation des licences libres (1.2.1) et l'extension du Libre à de nouveaux domaines (1.2.2).

1.2.1 L'arrivée de nouvelles licences libres industrielles : la phase de diffusion et d'émancipation

Le succès des licences *copyleft*. Concernant le type de licences, on remarque que la plupart des licences rédigées pour des projets d'origine privée (sociétés ou associations) étaient *copyleft* alors qu'à l'inverse les projets scientifiques ou communautaires privilégiaient les licences permissives, tel le W3C qui rédigea sa propre licence fin 2002 sous le nom de Software Notice and License. Les premiers cherchaient ainsi à maintenir un contrôle en empêchant la réappropriation de leur création alors que les projets tournés uniquement vers une optimisation de la diffusion s'orientaient vers des licences permissives. Tout à fait logiquement, compte tenu de la longueur et de la complexité des textes, les améliorations concernaient principalement les licences *copyleft* et étaient généralement destinées à renforcer la pérennité recherchée par l'usage de la licence : par une meilleure formulation de la licence relative aux brevets, par la recherche d'un élément déclencheur adapté, etc. À l'inverse, les seules améliorations constatées en matière de licences permissives consistèrent au raccourcissement des textes : telle la Fair License rédigée en 2004, qui est la plus courte des licences certifiées par l'OSI (composée de seulement avec les deux phrases), ou la Wide Open License (WOL).

L'attrait de l'*open source* pour la diffusion des standards. Le développement du marché de l'édition logicielle a entraîné très rapidement des positions de monopoles et de domination par certains éditeurs. En réaction, l'industrie du logiciel a rapidement compris l'intérêt que présentait l'*open source* dans la (re)conquête d'un secteur ou d'un marché. Les sociétés s'accordèrent aussi sur l'intérêt crucial de la compatibilité et de l'interopérabilité. La Sun Industry Standards Source License (SISSL) fut utilisée par OpenOffice.org jusqu'en 2005. Elle permettait la réutilisation du code libre et sans contrainte dès lors que les modifications ne déviaient pas du standard (les APIs OpenOffice.org et le format de fichiers OpenOffice.org utilisé par OoO) et devenait *copyleft* en cas de déviation trop importante. L'adoption des libertés du logiciel libre ne fut pas instantanée et il est intéressant de constater l'évolution du positionnement des sociétés privées qui cherchèrent initialement à proposer des équilibres conçus à leur avantage (la Sun Community Source License publiée en 1998 différenciait ainsi les usages pour la recherche, l'utilisation/déploiement et l'utilisation commerciale) et optèrent progressivement pour des licences de plus en plus équitables, répondant aux définitions du logiciel libre et de l'*open source*. Les sociétés eurent donc tôt fait de comprendre et de se plier aux attentes des communautés qu'elles sollicitaient.

L'impact de l'industrialisation sur les licences. Bénéficiant ainsi de retours précieux (en termes d'utilisation et de développement) de ses utilisateurs, l'industrie du logiciel investit de plus en plus au profit de cette innovation partagée. Cela eut un impact fort sur le monde des licences, puisque l'immixtion de l'industrie passa par la mise à disposition des services juridiques des grandes sociétés de l'époque (IBM, Sun, Apple, etc.). La pratique juridique accessoire (et parfois amateur) devint ainsi une pratique stratégique (la gestion de la propriété intellectuelle étant une priorité pour ces sociétés éditrices ou intégratrices) et différentes modifications furent apportées : modifications accessoires (l'Intel Open Source License fut publiée en 2000 pour ajouter un article à la BSD quant aux lois relatives à l'export) ou plus importantes. À cet égard, la contribution prépondérante est sans aucun doute la rédaction de la Mozilla Public License (MPL) par la firme Netscape.

La Mozilla Public License (MPL). La licence Mozilla Public License (MPL) fut créée pour la diffusion du code source du Navigateur Netscape

Communicator de la société Netscape¹. La société souhaitait favoriser la création de solutions (même propriétaires) intégrant son logiciel tout en s'assurant que les modifications apportées à celui-ci resteraient accessibles et disponibles. En l'absence de licence adaptée, la rédaction d'une nouvelle licence fut décidée. Au confluent de l'industrie et des communautés, elle pouvait s'appuyer sur les licences existantes ainsi que sur les pratiques professionnelles des juristes de la société (et notamment Mitchell Baker, qui fut plus tard présidente de la Mozilla Foundation et de la Mozilla Corporation). La licence initialement soumise aux commentaires était la Netscape Public License (NPL), néanmoins, la prise en compte des multiples commentaires aboutit à la rédaction d'une nouvelle licence intitulée Mozilla Public License. En effet, la NPL garantissait à la société Netscape la faculté de relicencier sous toute autre licence (même commerciale) les contributions au code : une prérogative généralement conservée par le titulaire de droits et qui créait un déséquilibre au seul profit de Netscape, ce qui était mal accepté par les personnes consultées. Pour respecter l'ensemble des intérêts en présence, tout le code publié par Netscape le 31 mars 1998 fut publié sous NPL et tout développement subséquent sous MPL (avec une transition vers la MPL projetée en 2 ans).

La fondation de l'OSI. L'Open Source Initiative (OSI) fut imaginée le 3 février 1998 à Palo Alto, peu de temps après la libération du code de Netscape. Elle résultait du sentiment que Netscape ouvrait la porte à une dynamique plus grande, que se généralisait un mouvement d'ouverture aux communautés et qu'il devenait opportun de promouvoir cela en s'émancipant de la notion trop politique (et trop peu industrielle à leur goût) de logiciel libre. Le logiciel libre suivait encore les pas d'Internet : de la même manière qu'il avait imprégné l'infrastructure d'Internet lors de sa construction, il se banalisait en même temps que celui-ci. Fondée par Eric Steven Raymond et Bruce Perens, l'OSI est une *non profit public benefit corporation* (sorte d'association à but non lucratif) créée avec le soutien et l'étroite collaboration d'organisations de l'époque (telles que l'Internet Engineering Task Force ou l'Internet Systems Consortium qui voyaient dans l'*open source* un développement structurant pour Internet). Elle compta dès ses origines Todd Anderson, Chris Peterson (qui proposa l'appellation *open source*), John Maddog Hall, Larry Augustin, Sam

1. À ce sujet, voir l'article de Robert Viseur, dans cet ouvrage.

Ockman. Parmi ses premières contributions, l'OSI rédigea l'Open Source Definition (OSD) en s'appuyant sur les travaux récents réalisés en 1997 par Bruce Perens sur les Debian Free Software Guidelines (DFSG) du *Contrat Social Debian* (Debian est à cet égard le premier projet à avoir formalisé un « contrat social » assurant la pérennité du projet et une politique en matière de gestion de licence au sein de la distribution afin que ne soit inclus que des outils conformes à la philosophie GNU). Contrairement à la Free Software Definition, l'OSD s'intéresse essentiellement au contenu de la licence et les dix critères qu'elle énonce¹ déterminent les droits et conditions que doivent remplir les licences pour permettre un développement *open source*². Sur la base de l'OSD, l'OSI a accompagné de nombreuses entreprises et individus dans la rédaction de leurs licences, corrigeant leur rédaction et les accompagnant dans leur effort d'ouverture en certifiant leur licence d'*open source*³. Les conditions sont ainsi : la libre redistribution du logiciel (la licence ne peut, par exemple, exiger le paiement d'une redevance supplémentaire) ; la fourniture (ou l'accès) du code source ; la possibilité de créer des œuvres dérivées ; la préservation de l'intégrité du code (pour éviter l'appropriation par un tiers de contribution d'un autre – les modifications peuvent n'être éventuellement distribuées que sous forme de *patch*, séparément) ; l'absence de discrimination entre les personnes ou les groupes (toute personne détentricice d'une copie du logiciel bénéficie des termes de la licence tant qu'il s'y conforme

1. La dixième clause a été ajoutée en 2004 afin de s'assurer que les licences n'étaient pas spécifiques à un type de logiciel et de sortir de l'usage des *click-wrap*, le contrat n'étant conclu qu'à raison d'un clic de confirmation de l'utilisateur.

2. Il faut néanmoins reconnaître que les licences conformes à l'une des définitions le sont quasi systématiquement à l'autre, mais on peut trouver des exemples contraires, labellisées *open source*, mais non considérées comme libres, telles la Reciprocal Public License, l'Artistic licence, l'Apple Public Source License v1.x ou la Common Public Attribution License ; ou inversement telle la première licence de l'Affero GPL ou toutes les licences non encore labellisées telles que les CeCILL.

3. À noter que l'une des missions de l'organisation était de promouvoir la marque « Open Source ». Celle-ci ne fut néanmoins jamais détenue par l'OSI (une tentative d'enregistrement échoua en 1999 car le signe fut, déjà, considéré comme trop descriptif – pour éviter toute appropriation du langage, un signe ne peut en effet être accepté comme marque que s'il est suffisamment distinctif des produits ou services désignés : c'est-à-dire porteur d'un nouveau sens, ce qui n'était ici pas le cas puisque la notion d'*open source* était déjà largement reprise – contrairement aux autres marques : OSI, Open Source Initiative et les logos. Ce sont ces dernières qui permettent à l'OSI de contrôler l'usage « OSI Certified » par les éditeurs de logiciels *open source* et aussi sur cette base qu'a été conclu en 2012 un accord avec l'Open Source Hardware Association (OSHW) pour l'exploitation de son propre logo.

lui-même) ; l'absence de discrimination entre les domaines d'application (la licence se limite à la propriété intellectuelle : elle ne peut en aucun cas réguler d'autres champs de la société) ; l'absence de dépendance vis-à-vis d'autres contrats (par exemple un accord de confidentialité) ; l'absence de dépendance vis-à-vis d'un produit (la licence est attachée au code et non à un logiciel particulier) ; l'absence de contraintes vis-à-vis d'autres logiciels (ce qui ne s'oppose pas aux licences *copyleft* ayant un large *copyleft*, telle la GNU GPL qui ne s'étend qu'au programme envisagé comme un tout) ; la neutralité technologiquement. À l'instar de la FSD, l'OSD fut ensuite dérivée afin de transposer le courant du Libre et de l'*open source* dans d'autres domaines.

Les descendances de la MPL. Cette licence arriva rapidement aux oreilles des sociétés du secteur, attentives au mouvement naissant. La licence était spécifique au projet Mozilla, mais suggérait la création de licences dérivées : ce que firent très rapidement des sociétés sensibles à l'*open source* comme l'étaient Apple, Sun ou IBM. Bien d'autres licences s'en inspirèrent : telles la Nokia Open Source License (NOKOS License), l'Australian National University Open Source (ANUOS) License, CUA Office Public License, Interbase Public License, Ricoh Source Code Public License, Enhydra Public License, etc. Certaines y ajoutèrent des conditions supplémentaires telles la Yahoo ! Public License qui se réservait le droit de rompre le contrat si elle constatait une violation par l'utilisateur ou la Zimbra Public License qui étendit les obligations en termes de maintien des mentions légales aux mentions figurant dans l'interface utilisateur. L'université de Victoria publia en 2003 l'Adaptive Public License, une licence flexible basée sur la MPL qui donne au concédant le choix : de la juridiction, d'une licence de brevets, d'une extension aux usages internes, d'une formalisation des distributions de modification, et certaines attributions complémentaires. Très actif en matière d'*open source*, le NASA's Goddard Space Flight Center rédigea en 2004 sa propre licence en 2004 (NASA Open Source Agreement 1.3 – les versions précédentes n'ont jamais été publiques) basée sur la MPL et ses variantes.

L'apport d'IBM, Apple et Sun. Ces trois grands groupes surent très rapidement tirer profit de ce courant, mais ils y contribuèrent tout autant (reprenant et améliorant la MPL). Ainsi, IBM publia en 1999 l'IBM Public License en introduisant une notion de responsabilité à l'encontre de

l'entité (*publisher or distributor of the licensed software code*) qui prenait la charge de la distribution, en renforçant les clauses en matière de brevets (la licence étant annulée en cas de litiges de brevets) et en assouplissant le *copyleft* de la licence (afin de permettre l'ajout de liens). Elle inspira elle-même d'autres licences (notamment la Lucent Public License en 2000) et devint Common Public License en 2001 afin de faciliter sa réutilisation. La licence laissa la place à l'Eclipse Public License (EPL) en 2004, version relativement similaire modifiée en faveur de la fondation Eclipse. L'Apple Public Source License (APSL) aussi publiée en 1999 descendit d'un cran l'élément déclencheur : toute utilisation, même interne, emportait des obligations en termes de formalisme ; ajout d'une obligation de prévenir la société à chaque déploiement d'une version modifiée ; et une clause de résiliation arbitraire permettant à Apple de rompre la licence en cas d'accusation de violation de brevets ou de copyright. Après deux mises à jour en 1999 et 2001, toujours insuffisante aux yeux de la FSF, une version 2.0 a été publiée en 2003 avec le soutien de cette dernière. Sun publia en 1999 la Sun Public License (SPL) qui se contentait de remplacer les références à Mozilla par celles de Sun et, fin 2004, une licence plus aboutie sous le nom de Common Development and Distribution License (CDDL) qui complétait la MPL concernant la menace des brevets et, surtout, offrait la possibilité d'être réutilisée par tout autre projet (OpenSolaris, Glassfish).

La Qt Public License. L'évolution de modèles de licence autour de la bibliothèque Qt est intéressante en termes de dynamique. Co-fondateurs de TrollTech, Haavard Nord et Eirik Chambe-Eng développèrent en 1991 la bibliothèque Qt qu'ils commercialisèrent différemment selon qu'il s'agisse de la version Windows (Qt/Windows) ou Unix (Qt/X11). La version Unix était sous une licence Freeware (FreeQt licence) permettant le développement d'applications autour de celle-ci sur la plate-forme Unix alors que la version Windows ne pouvait être utilisée qu'à condition d'acheter une licence commerciale de la bibliothèque. Suite à la publication, en juillet 1998, de l'environnement de bureau K Desktop Environment (KDE) basé sur la bibliothèque Qt, la société Trolltech subit une forte pression de la communauté pour diffuser Qt sous une licence libre ¹.

1. Voir notamment « Position de Debian concernant les droits d'auteur et les problèmes de licence concernant KDE », 8 octobre 1998, URL : <http://www.debian.org/News/1998/19981008.fr.html>.

Elle prit celle-ci en compte par une double contribution : la rédaction de la licence Q Public License (QPL) pour la version Unix¹ et la création de la KDE Free Qt Foundation avec pour mission d'assurer la disponibilité et la pérennité de Qt (une convention lie l'éditeur à la communauté de telle sorte que s'il renonce à la publication de Qt sous sa licence *open source*, alors la fondation pourra diffuser Qt sous licence de type BSD ou toute autre licence *open source* de son choix). La licence Qt n'étant pas compatible avec la GNU GPL, la société Trolltech a modifié sa licence au profit de la GNU GPL v2 en 2000 (et GNU GPL v3 en 2008). En 2008, Trolltech est racheté par Nokia qui diffuse sous GNU LGPL 2.1 début 2009 et revend le projet à Digia le 9 août 2012.

L'Open Software License (OSL) est une licence issue des travaux de Lawrence E. Rosen, à l'époque conseiller juridique de l'OSI (avocat très impliqué dans l'*open source*, il fut directeur de la fondation Apache de 2011 à 2012). La première version de cette licence date de 2002. Modifiée la même année (version 1.1), Larry Rosen introduisit la notion d'*external deployment* (l'utilisation et les liaisons par réseau de logiciels) comme élément déclencheur afin de s'assurer que l'usage d'un logiciel par le réseau emporte les mêmes conséquences qu'une distribution de ce logiciel à l'utilisateur final. La version 1.1 fut certifiée *open source* par l'Open Source Initiative (OSI), mais fut déclarée incompatible avec la DFSG par la communauté Debian en raison de la clause de résiliation en matière de brevet. La version 2.0 date de 2003, et la dernière version, la troisième, date de 2005 (elle prend notamment en compte les remarques de la Commission européenne). Dès sa publication, la licence est expressément présentée comme un contrat (par opposition à la GNU GPL qui se présente comme une licence). Marginalement utilisée, elle est cependant très bien rédigée et a inspiré un certain nombre de licences (dont la *Jabber Open Source License*). Elle existe aussi dans une version permissive, sous l'appellation d'Academic Free License (aussi publiée en 2002). Ces licences furent un temps qualifiées de *best practice* par l'OSI (mi-2006) avant d'être catégorisées comme « redondante avec des licences plus populaires » quelques années plus tard.

1. La version Windows n'était toujours pas libre, le projet KDE sur Cygwin a travaillé en 2002 au portage de la bibliothèque Unix sur Windows et ce n'est qu'à partir de Qt/Windows 4, en juin 2005, que Trolltech a décidé d'étendre la GNU GPL à toutes les plates-formes (ce qui permit alors le développement d'applications multiplateformes).

L'évolution itérative des licences. Profitant de cette mouvance, le projet LaTeX s'est doté, début 1999, d'une nouvelle licence intitulée LaTeX Project Public License. Cette licence et ses versions 1.1 et 1.2 publiées la même année étaient tout à fait adaptées à la diffusion de la technologie LaTeX puisqu'elles permettaient la redistribution sans restriction de l'ensemble (mais d'une partie) des fichiers non modifiés, cependant elles encadraient beaucoup plus strictement la distribution de fichiers modifiés : celle-ci devant se faire, sauf mention contraire, dans l'en-tête des fichiers, à condition que ceux-ci soient renommés et que les auteurs initiaux soient clairement attribués. La licence fut modifiée en décembre 2003 avec le concours de Debian et de l'OSI, afin d'atténuer ce mécanisme en recommandant de tels changements de noms tout en permettant alternativement une identification « claire et non ambiguë » des versions modifiées (un mécanisme et une évolution pouvant ainsi être rapprochés de ceux des licences Artistic et Zope). L'Artistic License a aussi été modifiée à plusieurs reprises au profit de variations telles que la Simplified Artistic License, conçue en 2002 pour l'eNetwizard Content Application Server, ou encore la Clarified Artistic License, rédigée courant 2000 par Bradley Kunht (cette dernière sera en partie reprise lors de la publication en 2006 d'une version 2 de la licence).

Quelques licences particulières et non libres. L'année 2002, qui fut celle de la déclaration de guerre des USA contre le terrorisme, eut un impact sur les licences avec l'apparition de l'Hacktivismo Enhanced Source Software License Agreement (HESSLA – qui inspira le Torrifly Ethical Software License Agreement quelques années plus tard) ou encore la JSON License. La première est une licence relativement complète qui interdit les utilisations ou modifications qui violeraient les droits de l'homme ou entraîneraient l'espionnage des utilisateurs ; la seconde est une licence MIT à laquelle l'auteur (Douglas Crockford) ajouta « *The Software shall be used for Good, not Evil* ». Ces deux licences posèrent un certain nombre de difficultés au regard de leur subjectivité inhérente, de sorte qu'elles furent considérées comme « non libres » alors même que les logiciels ainsi développés présentaient un fort intérêt. Précisons qu'à la demande de la société, Douglas Crockford accepta d'accorder l'exception suivante : « *I give permission for IBM, its customers, partners, and minions, to use JSLint for evil.* »

Les licences « Badgeware ». Le terme *badgeware* apparut pour décrire ces licences qui imposaient l’affichage permanent de l’attribution des origines (telle la licence Zimbra) ou la Common Public Attribution License (publiée par la société Socialtext en 2007) qui permettait au développeur original d’imposer à chaque lancement du logiciel un affichage proéminent des attributions.

La seconde version de la licence Apache. En 2004, une refonte globale de la licence donna une version 2 beaucoup plus complète et rigoureuse répondant ainsi aux besoins de sécurité juridiques exprimés par les utilisateurs industriels. La fondation Apache considéra rapidement sa licence comme compatible avec la GNU GPL, mais cette compatibilité ne fut en réalité atteinte qu’à travers la version 3.0 de la GNU GPL.

Les licences CeCILL. En France et à l’initiative des principaux centres de recherche publique, une première licence libre *copyleft* française, conforme au droit français et européen (la Loi Toubon contraint les personnes publiques à contracter en Français en France), et compatible avec la licence GNU GPL, fut publiée le 21 juin 2004 sous le nom de CeCILL pour Ce(a) C(nrs) I(nria) L(ogiciel) L(ibre)¹. Ce travail avait été soutenu par Renaud Dutreil, alors ministre de la Fonction publique et de la réforme de l’État, qui était allé jusqu’à demander à l’Agence pour le développement de l’administration électronique (ADAÉ) « en concertation avec les trois organismes à l’origine de CeCILL, de porter ce travail au niveau européen et de le soutenir auprès des pays francophones ». Cette première version essuya néanmoins un certain nombre de critiques de la part de la FSF et de l’APRIL², de sorte qu’une seconde version vint en remplacement le 21 mai 2005 afin d’étendre la cession en matière de brevets, préciser certaines notions techniques ou formulations, et accorder une pleine valeur juridique à la version anglaise. Depuis, deux variations, nommées CeCILL-B et CeCILL-C, vinrent s’ajouter en 2006 à la « famille des licences CeCILL » dorénavant composée d’une licence permissive (CeCILL-B), d’une licence *copyleft* à l’étendue inspirée de la MPL (CeCILL-C), et d’une licence *copyleft* inspirée de la GNU GPL (CeCILL ou CeCILL-A).

1. Initiative comparable à la German Free Software License (Deutsche Freie Software Lizenz), elle aussi publiée en 2004, compatible avec la GNU GPL et disponible en allemand et anglais.

2. Association pour la promotion et la recherche en informatique libre.

Les licences *shared source* de Microsoft. Microsoft a formalisé dès 2001 un programme *shared source* qui ouvre dans une certaine mesure (en fonction des produits et des règles d'éligibilité prédéfinies) du code de la société à titre de référence et d'interopérabilité, pour l'étude de la sécurité de certains systèmes ou pour le développement. Plusieurs licences ayant été rédigées dans ce cadre, certaines avaient été soumises à l'OSI dès 2005 sans l'accord de Microsoft qui, après avoir été contacté, préféra reporter cette décision. C'est donc en 2007 que Microsoft proposa à l'OSI deux des licences de son programme Shared Source Initiative. Elles furent labellisées *open source* le 12 octobre 2007 et changèrent de nom à cette occasion : la Microsoft Public License (Ms-PL, auparavant Microsoft Permissive License) et Microsoft Reciprocal License (Ms-RL, auparavant Microsoft Community License). Les licences restantes (Microsoft Reference Source License, Ms-RSL, Microsoft Limited Public License, Ms-LPL, Microsoft Limited Reciprocal License, Ms-LRL) ne peuvent néanmoins pas prétendre à la qualification d'*open source*.

L'European Union Public Licence (EURL). La licence de l'Union Européenne (EURL) est apparue sous forme de brouillon en 2005 (v.0.1) et 2006 (v.0.2) avant d'être approuvée par la Commission Européenne le 9 janvier 2007. Disponible en 22 langues, chacune de ses traductions ayant la même valeur contractuelle, l'European Union Public License est sans conteste la licence la plus internationale. Il est intéressant de noter qu'une étude relativement poussée des licences existantes a précédé la rédaction de celle-ci, surtout en raison du faible nombre de licences européennes et des garanties souhaitées pour la commission (la version 3 de l'OSL avait pour objectif de corriger les défauts identifiés par la Commission afin de la rendre éligible). Une version 2.0 est discutée depuis fin 2012 afin de mettre à jour les termes initiaux et agrandir la liste des licences *copyleft*.

La structuration de l'*open source*. En matière de licence, les actions combinées du projet Debian, de l'OSI et de la FSF favorisèrent autant qu'elles encadrèrent cette prolifération. Néanmoins, de nombreuses autres organisations apparurent dans les années 2000 et eurent un rôle déterminant dans l'adoption et la consolidation du mouvement : l'association Software in the Public Interest (1997) les fondations Apache (1999), GNOME (2000), Mozilla (2003) ou Eclipse (2004) l'Open Source Development Labs (OSDL) qui devint la Fondation Linux en 2007 (après la

fusion avec le Free Standards Group, FSG). La plupart de ces organisations étaient dédiées à un projet en particulier, mais avaient généralement d'autres objets plus larges (apporter le soutien à d'autres projets Libres pour la fondation Apache, « préserver le choix et l'innovation sur l'Internet »¹ pour la fondation Mozilla, etc.).

1.2.2 L'extension à de nouveaux domaines

Un contexte particulier². De façon plus ou moins liée au mouvement du logiciel, les autres domaines de la création furent progressivement touchés. La FSF ne s'est pas clairement positionnée sur le sujet³, mais elle accompagna – par la personne de Stallman pour un premier temps, mais ensuite avec d'autres de ses acteurs – certaines de ces démarches. En parallèle et pour contextualiser ce phénomène, il faut se rappeler que cette époque est aussi celle des lois poussées par l'industrie culturelle : le Traité de l'OMPI sur le droit d'auteur de 1996, les lois américaines tels que le Sonny Bono Copyright Term Extension Act (étendant la durée des droits à 90 ans pour les créations salariales – aussi appelé « Mickey Mouse Protection Act » – Lawrence Lessig s'était illustré en contestant cette Loi devant la Cour suprême des États-Unis⁴) et le Digital Millennium Copyright Act (DMCA) de 1998 ; la Directive européenne sur l'harmonisation de certains aspects du droit d'auteur et des droits voisins dans la société de l'information (2001) transposée en France par la Loi DADVSI (Droit d'Auteur et Droits Voisins dans la Société de l'Information).

Les licences destinées à la documentation. La documentation associée au logiciel fut le premier type de création non logicielle pour lequel fut ressenti le besoin d'utiliser aussi une licence libre. Les motifs, simples, étaient que la documentation est l'un des facteurs clés de la collaboration et qu'il est donc nécessaire qu'elle soit aussi libre que le logiciel sur lequel elle portait. Jusqu'aux années 1998, les licences des logiciels étaient

1. Voir *The Mozilla Manifesto*, URL : <http://www.mozilla.org/about/manifesto.html>.

2. Pour cette section, et notamment pour une présentation plus précise de l'appréhension du Libre dans le domaine de l'art et du contenu, voir l'article de Primavera De Filippi et Isabelle Ramade, dans cet ouvrage.

3. Voir « Licences pour les œuvres d'opinion et de jugement », URL : <http://www.gnu.org/licenses/license-list.fr.html>.

4. Lawrence LESSIG, « How I Lost the Big One », dans : *Stories, Legalaffairs* (2004), URL : http://www.legalaffairs.org/issues/March-April-2004/story_lessig_marapr04.msp.

simplement étendues à leur documentation et il existait donc, à l'instar du logiciel, des partisans de la GNU GPL ou de la licence BSD (Expat). Néanmoins, l'importance de la documentation s'accrut au fur et à mesure que les communautés s'élargissaient et c'est donc dans le même temps que le logiciel libre gagna ses lettres de noblesse dans l'industrie du logiciel et que les licences furent rédigées en faveur de l'édition de documentation libre et ouverte – la Free Document Dissemination License (FDDL) utilisée dès 1997 par l'INRIA favorisait la diffusion et l'amélioration (par le biais de notes et commentaires) sans faire de celle-ci une œuvre collaborative. Parallèlement, et conscient que la GNU GPL ne répondait pas parfaitement aux besoins, Stallman publia le 12 septembre 1999 une première version 0.9 « brouillon » de la GNU Free Documentation License, une licence dédiée à la documentation logicielle et conforme à la FSD. Après quelques révisions, une version 1.1 fut publiée en mars 2000 (la version 1.2 date de 2002). La documentation s'envisageant comme un tout, la licence devait étendre le *copyleft* initial à tout ajout ou modification réalisé sur la documentation initiale. Enfin, l'un des autres intérêts était le vecteur de communication que représentait la documentation : il était en effet possible d'expliquer, voire d'approfondir, au sein de cette documentation la démarche liée à l'utilisation d'une licence libre. Non satisfait de cette licence qu'elle trouvait trop complexe, Apple publia sa propre licence *copyleft* dédiée à la documentation un an plus tard : la Common Documentation License. Directement inspirée de la MPL, la Public Documentation License (PDL) est publiée en 2002 par Sun pour la documentation de certains de ses logiciels (notamment OpenOffice.org).

Le mouvement de l'Open Content. Dès juillet 1998, David A. Wiley publia l'Open Content License. Il s'agissait d'une licence *copyleft* relativement courte et permissive qui permettait à l'auteur d'interdire les créations dérivées (de sorte qu'il existait 4 variantes : OPL, OPL-A, OPL-B et OPL-AB : l'option A interdisait tout usage commercial de l'oeuvre, l'option B les modifications). La licence fut remplacée onze mois plus tard par l'Open Publication License, notamment utilisée par O'Reilly pour l'ouvrage d'Eric S. Raymond, *The Cathedral and the Bazaar*. Le projet *Open-Content* fut officiellement fermé le 30 juin 2003 lorsque David A. Wiley rejoignit Creative Commons comme *Director of Educational License* (il n'est plus aujourd'hui impliqué dans Creative Commons mais continue à

travailler dans le domaine des Open Educational Ressources, OER). En 1999, Michael Stutz publia une nouvelle licence, la Design Science License, qui se positionnait discrètement comme une licence *copyleft* générale adaptée à tout type d'œuvres. Elle préfigurait ce que seraient ensuite les licences libres non logicielles (même si les auteurs avaient jugé utile d'y intégrer quelques spécificités liées au changement d'environnement : chaque nouvelle version doit par exemple porter un nouveau nom, ce qui favorise la création à partir d'œuvres existantes, mais complexifie la création d'œuvres collaboratives).

La culture Libre. La « fibre du Libre » est très ancienne dans la musique¹. Les artistes se sont très rapidement approprié l'usage d'Internet pour diffuser leurs œuvres, les faire connaître et en créer de nouvelles grâce aux contributions venant de toute part. Un nombre important de licences écrites pour la musique s'inspirèrent de la GNU GPL : la Free Music Public License (FMPL) et la Choral Public Domain Library License (CPDLL) en 1998, la Licence Art Libre (LAL) et l'Ethymonics Free Music License (EFML) en 2000, les Open Music Licenses (OML) (Green, Yellow, Red et Rainbow License) de l'Open Music Project lancé en 2001, l'EFF Open Audio License en 2001, les onze licences Creative Commons en 2002 (la Music Sharing License, aussi appelée CC By-NC-ND, étant la plus utilisée), la Loca Public License en 2002, l'Open Source Music License (OSML) en 2003 ou encore la licence IANG Ain't No GNU ! (IANG) en 2004. Les Open Music Licenses étaient certainement les plus complexes, préfigurant aussi ce que seraient les Creative Commons puisqu'elles étaient rédigées sur un socle commun et permettaient d'ajouter plusieurs interdictions, dont l'usage commercial et l'interdiction de la modification – et d'ailleurs Laurence Lessig² y contribua avant de

1. De nombreux exemples pourraient être cités, mais le mouvement Death Pact International est peut être le plus illustre : même s'il n'y avait pas de licence à proprement parler, il s'agit d'un concept de projet musical créé par un groupe anglais, Grey Wolves, au début des années 1980. L'idée était que toute œuvre distribuée sous le nom de DPI était librement utilisable, modifiable et redistribuable dès lors que cela était fait sous le même nom du Death Pact International (il y avait donc cette idée de fond commun, mais sans la reconnaissance de paternité maintenue par le jeu des licences).

2. À l'initiative du projet *Copyright's Commons* en 1999 au sein duquel il promut l'usage du signe *counter-copyright* [cc] pour signaler le partage de ses œuvres. Voir <http://cyber.law.harvard.edu/cc>.

se consacrer à Creative Commons¹. Ram Samudrala (artiste à l'origine de la FMPL) et Antoine Moreau (artiste initiateur de la LAL) furent ainsi les principaux penseurs de ce mouvement. Aujourd'hui, à l'exception de la Licence Art Libre (en version 1.3), la plupart de ces licences ont été abandonnées, ou retirées au profit des licences Creative Commons. Notons que les licences Creative Commons ne constituent pas l'unique objet de la fondation et qu'un certain nombre de projets dépasse le seul domaine des licences : ainsi le projet Public Domain Dedication ou la Certification de « mise dans le domaine public » furent imaginés dès 2002 (aujourd'hui abandonnés au profit de la licence CC-0 publié en 2009 et du projet Public Domain Mark de 2010). Un autre projet nommé Founders' Copyright avait été lancé dès 2002 afin de permettre aux créations d'une personne ou d'une entreprise – O'Reilly Associates ayant été moteur sur cet aspect – de tomber dans le domaine public passée la durée originale de 14 ans.

Les licences pour Wiki. À l'instar du droit d'auteur classique, les licences libres consacrent une place très importante à chaque auteur et contributeur, souvent sous la forme d'une mise à jour continue (sur le site et/ou dans les sources) d'une liste des différents contributeurs. Fluidifiant à l'extrême les processus de contribution et de collaboration, les projets formalisés au travers de wikis donnent néanmoins une place nouvelle et importante au projet lui-même (celui-ci étant très souvent cité en remplacement des auteurs réellement impliqués dans le contenu réutilisé). Cet usage, compréhensible pour des motivations de commodité, mais qui exprime aussi des changements culturels (le particulier s'éclipsant d'une certaine manière au profit du général) dut aussi être pris en compte dans la rédaction des licences (notamment en termes de formalismes s'imposant lors de la réutilisation du contenu). C'est cette raison qui suscita la mise à jour des licences Creative Commons au profit d'une version 2.5 (abandonnant de ce fait le projet d'une CC-wiki dédié) afin de préciser que la mention de la paternité devait être effectuée au profit du titulaire de droit ou d'une entité qu'il aurait désignée. Notons aussi que la FSF avait deux projets, aujourd'hui abandonnés, qui auraient pu permettre un tel usage : la GNU Simpler Free Documentation License (SFDL) ou la GNU Wiki (auquel renvoyaient les anciens brouillons de la GNU GFDL 1.3).

1. Lawrence LESSIG, *Code and Other Laws of Cyberspace*, New York : Basic Books, 2000.

L'Open Gaming. L'Open Gaming license est associée en 2000 sur le jeu *Dungeons & Dragon* par Wizards of the Coast, introduisant la notion de Document de Référence du Système (DRS) établissant leur jeu de rôle comme standard (en l'espèce le *d20 system*). La licence distingue les Open Game Content (OGC) des Product Identity (PI) : seuls les premiers (mécanismes du jeu) sont soumis à la licence *copyleft* alors que les seconds (tout ce qui est lié à la marque, aux figurines, histoires, graphismes, etc.) restent l'entière propriété de leurs auteurs. La portée de cette licence est donc à différencier de la *Licence ludique générale*, elle aussi publiée en 2000, qui portait la GNU GPL dans le domaine des jeux vidéos (en ajoutant quelques spécificités comme la notion de « noyau du jeu » définissant l'étendue de la licence d'après la volonté de l'auteur).

L'Open Access ou le libre accès. Le mouvement de l'Open Access trouve ses racines dans le milieu du vingtième siècle, mais c'est dans les années 1980, avec l'arrivée d'Internet et de l'informatique, qu'il put réellement se diffuser : perçu comme l'évolution logique des pratiques scientifiques à l'ère du numérique (favorisant les échanges, évitant de dupliquer des recherches et des erreurs, offrant une vision plus large de l'état de la science, etc.). L'Open Access repose sur un certain nombre de normes communes : le Protocole OAI-PMH élaboré en 1999 par l'Open Archive Initiative afin de favoriser l'interopérabilité entre les archives ouvertes ; les déclarations de Budapest (Budapest Open Access Initiative) en février 2002, Bethesda (Bethesda Statement on Open Access Publishing) en juin 2003 et Berlin (Berlin Declaration on Open Access to Knowledge in the Sciences and Humanities) en octobre 2003. Ce concept englobe les publications électroniques en libre accès ou accès ouvert, c'est-à-dire qui offrent à tout utilisateur un droit d'accès immédiat, gratuit, irrévocable et universel des articles scientifiques en entier. Dans le mouvement d'Open Access, les auteurs autorisent un usage gratuit et sans restriction de leurs productions et se tournent ainsi naturellement vers les licences de type Creative Commons sans que des licences spécifiques n'aient été rédigées ni même ne soient recommandées. Par ailleurs, le principe repose sur la conservation par les auteurs de leur titularité lors de la publication, ce qui induit de changer les pratiques actuelles (le projet Sciences Commons recommande à cet effet l'ajout de certaines clauses afin d'atténuer les cessations de droits, au travers du Scholar's Copyright Addendum Engine).

L'Open Data. Les prémisses de l'Open Data sont plus lointaines qu'on ne l'imagine et il est ainsi possible de citer des projets comme l'Open Directory Project (ODP, base de données collaborative de sites web plus connue sous le nom de *dmoz1*) libéré par Netscape sous Open Directory License en 1999 (le projet est sous CC By-SA depuis juin 2011), Tela Botanica (base de données botanique) ou Open Street Map (base de données cartographique), respectivement lancées en 1999 et 2004 sous licence CC By-SA. Il apparut néanmoins quelques années plus tard que ces licences n'étaient pas nécessairement adaptées aux bases de données et que les projets nécessitaient de faire reposer leur efficacité sur une assiette plus large que le seul droit d'auteur, en incluant également le droit des contrats et le droit *sui generis* des bases de données. Au travers du projet Open Data Commons, l'OpenKnowledge Foundation fit ainsi rédiger spécialement des licences adaptées à ce besoin : la Public Domain Dedication and License (PDDL) – « Public Domain for data/databases » ; l'Attribution License (ODC-By) – « *Attribution for data/databases* » ; et l'Open Database License (ODC-ODbL) – « *Attribution Share-Alike for data/databases* ». Seule licence *copyleft* adaptée aux données, l'ODbL impose que toute base de données – modifiée ou non – soit distribuée sous la même licence. La première version « brouillon » (0.9) de la licence fut soumise à commentaires le 27 février 2009 (et publiée quelques mois plus tard). De nombreuses licences spécifiques existent aujourd'hui (telle la Licence Ouverte publiée en 2011 par la Commission Etalab en France pour les données de l'administration).

Police Libre. Les protections sur les polices de caractères ont très tôt été affirmées et il semblerait que leurs concepteurs puissent prétendre à des droits d'auteur lorsque toutes les conditions sont réunies, les polices étant assimilées à des logiciels. Quelle que soit la validité de ces droits, la présence de ces protections était susceptible d'empêcher l'utilisation de polices les plus répandues par les logiciels libres n'ayant pas souscrit les licences nécessaires. Le projet Free UCS Outline Font est certainement le premier à avoir diffusé des polices de caractères sous licences libres (GNU FreeFont). Ces polices sont sous GNU GPL avec la *font exception* qui empêche l'extension de la licence à tout document utilisant ces polices. L'association SIL (Summer Institute of Linguistics Inc.) International, aussi à l'origine de nombreuses polices libres (publiées pour répondre à ses ob-

jectifs d'étude et de développement des langages), est à l'origine de la licence SIL Open Font License (OFL). Anticipant tout frein à l'adoption de leur logiciel, de nombreuses industries ont ensuite diffusé des polices libres : les polices de caractères *Liberation font* par la société Red Hat en 2007, en version Sans, Serif et Mono (Oracle a contribué en 2010 la police *Liberation Sans Narrow* sous une licence légèrement modifiée) ; Google a aussi publié trois polices que sont *Arimo*, *Tinos* et *Cousine*, distribuées sous OFL, et deux autres polices destinées à Android sous licence Apache v2.

Open Hardware. Les produits de l'Open Hardware étant plus complexes, ils sont couverts par de multiples droits d'importance différente : les droits de propriété industrielle (brevets, dessins et modèles, topographies de semi-conducteurs) sont ici prédominants alors que le droit d'auteur est encore relativement peu utilisé (sauf pour ce qui concerne les logiciels embarqués, ainsi que la partie reprogrammable de certains composants). L'Open Source Hardware and Design Alliance a déjà joué un rôle structurant en proposant dès 2009 un label Open Source Hardware (fondé sur une définition dérivée de la FSD) ainsi qu'une nouvelle terminologie proposée en 2001 en remplacement de Free/Libre Open Source Hardware : *reables*. En pratique, chaque projet a adopté une politique différente en matière de licences : GNU GPL pour *FreeCores*, GNU LGPL pour le projet Opencores (1999), licences *copyleft* ou permissives pour le Linux-Fund.org (1999 – initialement Open Hardware Foundation) et l'Open Graphics Project (quelques licences propriétaires également pour ce dernier), ou encore la rédaction de licences spécifiques comme pour le projet Balloon ou Tucson Amateur Packet Radio (TAPR). Le projet Qi hardware, qui succède au projet OpenMoko, a pour ambition de créer un pot commun de matériels libres et recommande aujourd'hui la GNU GPL v3 (à la place de son ancienne politique par laquelle les plans étaient sous CC By-SA, le code sous GNU GPL v2 et les brevets sous une licence particulière). Le CERN a publié la version 1.1 de sa licence OHL (v1 en mars 2011 ; v1.1 en juillet 2011), qui encadre les plans et documentations d'équipements aux spécifications libres tout en imposant certaines obligations liées au partage des modifications des plans et à l'avertissement de toute mise en œuvre de ceux-ci. Dans un domaine aussi très technique et couvert par les brevets, citons le *BiOS Material Transfer Agreement*, outil juridique créé

en 2006 pour une plate-forme de partage encourageant la coopération et le développement de technologies (les brevets n'étant partagés qu'au sein de la communauté).

2. La consolidation et l'harmonisation des outils juridiques (depuis 2007)

Un besoin de mise à jour. Près de deux décennies après le lancement du mouvement du logiciel libre, la nécessité de moderniser et d'harmoniser les instruments juridiques s'est imposée. En effet, un certain nombre d'éléments ayant changé, pris en compte par de nouvelles licences ou non (le type de contributeur ou d'utilisateur, les utilisations et la « consommation » de l'informatique, le contexte législatif, etc.) et certaines utilisations ayant dépassé les objectifs initiaux des licences (surtout les plus utilisées, qu'il s'agisse des licences GNU ou des Creative Commons¹), il était temps de rééquilibrer l'équation à laquelle souhaitent répondre les licences libres et de prendre un peu de hauteur pour que toutes les initiatives opèrent de façon plus concertée. En cela, la multiplication des licences peut être vue positivement puisqu'elle permet globalement l'amélioration de certaines clauses (en termes de cession, de formalisme, etc.) ainsi que l'identification de besoins non nécessairement anticipés dans les licences initiales. À l'inverse, elle fut aussi vécue comme un frein, empêchant la combinaison de logiciels complémentaires sous licences libres.

Deux phénomènes suivirent cette première phase : une prise en main des licences au travers d'une pratique de plus en plus perfectionnée des utilisateurs (2.1.) ; une harmonisation des licences libres au sein de l'écosystème par les rédacteurs des licences (2.2).

2.1 Une évolution des pratiques entourant les licences

Le besoin de se recentrer sur les effets de la licence. Pour la majorité, les organisations qui rédigèrent une licence le firent initialement pour des besoins propres : s'assurer de l'accessibilité du code, réserver un contrôle

1. Voir Florian CRAMER, « The Creative Common Misunderstanding », dans : *FLOSS+ Art*, sous la dir. d'Améric MANSOUX et marloes DEVALK, OpenMute, 2008.

(et donc un modèle économique), encourager le développement de *plug-ins* complémentaires, nuire à un concurrent, etc. Inversement, la grande majorité des créateurs a pendant longtemps utilisé les licences libres sans véritablement lire, analyser, ni même comprendre les textes, se contentant au contraire de l'explication et de la philosophie qui s'en dégageait. Cela explique certainement que des licences très bien rédigées telles que l'OSL ne connurent qu'un succès mitigé alors que les licences GNU furent utilisées bien au-delà du cadre pour lequel elles avaient initialement été rédigées. Mais, passé l'engouement initial (ayant entraîné des utilisations contestables), il y eut un mouvement progressif de reprise en main par les utilisateurs. Le Libre a en effet eu pour conséquence de rendre l'utilisateur « acteur » et responsable du cadre dans lequel était utilisée la création.

Cela se traduit parallèlement par des situations nécessitant le changement de licences au sein de projets communautaires (2.1.1.) et l'établissement de pratiques permettant de modifier les licences utilisées (2.1.2.).

2.1.1 Les pratiques des utilisateurs : les changements de licences

Une nécessaire titularité pour le changement de licence. Afin de maintenir la confiance assurée par la licence et pouvoir valablement sanctionner des comportements déviants, un certain nombre de projets communautaires durent changer leurs licences. Cela pose rapidement une difficulté, car la plupart d'entre eux n'avaient pas anticipé une telle hypothèse lors du choix initial de la licence et n'avaient donc pas la capacité suffisante pour opérer une telle action. Pour rappel, les licences ne sont que des contrats de cession non exclusive de droits de propriété intellectuelle¹ et seuls les titulaires de droits (généralement les auteurs, sauf s'ils ont accepté de céder leur droit de façon exclusive ou consenti à déléguer cette opération) peuvent convenir d'une diffusion dans d'autres termes. C'est pour cette raison que de nombreux organismes, dont la FSF ou la SFC, demandent une cession pure et simple des droits relatifs aux projets qu'ils hébergent. Une telle pratique n'était néanmoins pas du goût de tous

1. « Les licences libres sont les licences par lesquelles l'auteur autorise la copie, la modification et la diffusion de l'œuvre modifiée ou non, de façon concurrente, sans transférer les droits d'auteur qui y sont attachés et sans que l'utilisateur ne puisse réduire ces libertés tant à l'égard de l'œuvre originale que de ses dérivés. Les œuvres ainsi diffusées sont qualifiées de libres. », CLÉMENT-FONTAINE, *loc. cit.*

(la création en 1991 du projet Lucid Emacs, *fork* de l'outil Emacs, a notamment été imputée au refus d'une cession systématiquement des droits sur le projet) et est restée longtemps mal comprise. De ce fait, quelques projets connurent des difficultés à changer leur licence, qu'il s'agisse de : documentations (l'exemple de Debian et de Fedora), logiciels (l'exemple de Mozilla, Squeak et du noyau Linux), « sites de collaboration multiauteurs » (Wikipédia) ou encore de bases de données (OpenStreetMap).

En matière de documentation : l'exemple de Debian (OPL à GNU GPL et MIT). La communauté Debian, notamment au travers des membres du groupe *debian-legal*, est particulièrement sensible aux enjeux de licences (systématiquement scrutées à l'aune du DFSG). Initialement rédigé à destination des logiciels, ce document n'a été appliqué que plus tardivement à tout contenu contribué (programme, documentation, image, son, etc.). Ainsi, en sa qualité de précurseur sur le sujet, la communauté avait naturellement mis le contenu de la documentation sous l'Open Publication License v1 lors de sa sortie en 1999. Quelques années plus tard, alors que d'autres licences fidèles au principe des quatre libertés avaient été publiées, la question se posa à nouveau avec d'autant plus d'acuité et de recul. Dès 2003, certains critiquèrent l'incompatibilité de l'OPL avec la GNU GPL, allant jusqu'à constater le non-respect de la DFSG le 14 mars 2004 (en raison d'un formalisme trop contraignant et susceptible de limiter les modifications, ainsi que d'une non-compatibilité vis-à-vis du « Dissident Test »). Depuis lors, et après plusieurs discussions animées sur les listes dédiées (notamment en 2006 et 2011), il fut choisi le 25 janvier 2012 de diffuser le contenu de la documentation sous GNU GPL (ou licence compatible) et licence Expat. Depuis cette date, toute nouvelle contribution peut être redistribuée et modifiée selon les termes de la licence Expat ou de la GNU GPL (version 2 ou ultérieure)¹. Près de 10 ans de procédure !

En matière de documentation : l'exemple inverse de Fedora (OPL à CC By-SA). Parallèlement, la communauté de Fedora avait entamé la même démarche de relicenciement du contenu sous OPL en 2009 au profit de la CC By-SA (conformément aux recommandations de David A.

1. Le contenu antérieur restant disponible sous OPL v1 dans l'attente de l'obtention d'un accord de tous les contributeurs (ceux-ci ont été contactés et la plupart ont accepté ce changement, la dernière phase de « consolidation » est programmée pour 2013).

Wiley). L'objectif étant de rejoindre les pratiques des autres communautés afin de favoriser le transfert de contenu (« *We can move from a tiny archipelago to the largest continent of open content under the flag of the CC BY SA* »¹). Cela fut néanmoins beaucoup plus rapide que pour Debian car Fedora possède une grande souplesse grâce au Contributors License Agreement imposé par Red Hat pour toute contribution (Debian s'étant tardivement rendu compte qu'en l'absence de cession à SPI, cette dernière n'était pas titulaire des droits comme la plupart l'avaient pensé). Cela permit d'opérer le changement en pas moins de trois années.

En matière de logiciel : l'exemple de Mozilla (MPL à la multilicence MPL/GPL/LGPL). Le navigateur Mozilla a aussi connu plusieurs changements de licence : lors de sa sortie en 1998 pour passer de la NPL à la MPL, puis en 2004 pour ajouter deux licences (la GNU GPL v2 et la GNU LGPL v2.1) au projet alors disponible à triple licence² à l'instar d'autres projets de Mozilla (tels que NSPR, NSS, PSM ou le moteur JavaScript). Ce relicenciement fut néanmoins relativement long puisque l'idée d'ajouter les deux licences GNU germa en août 2000 afin de conserver une licence adaptée au projet tout en rassurant les communautés du Libre (grâce à la compatibilité ainsi offerte). La formalisation du processus fut publiée le 19 septembre 2001 et l'achèvement de la procédure en janvier 2004. Même si la fondation s'était vu céder les droits que possédait Netscape en vertu de l'article 3 de la NPL (Alternative Licensing, permettant de choisir toute autre licence pour la redistribution du code), il était nécessaire d'obtenir l'accord de toutes les personnes ayant contribué au projet avant le 19 septembre 2001 (les contributions postérieures devant être faites sous triple licence). Ce fut un processus relativement long par lequel près de 450 contributeurs, personnes physiques ou morales, acceptèrent d'associer ces deux licences supplémentaires à leur code. Une fois ces autorisations récoltées, un script a effectué la mise à jour des en-têtes sur les fichiers affectés (plus précisément durant les cycles des versions 1.7a et 1.7b). Il fallut donc trois années pour corriger le statut de contributions apportées durant seulement deux années. Inversement, l'abandon

1. Propos rapportés par David A. WHEELER, « Make Your Open Source Software GPL-Compatible. Or Else. », 2002 (révisé en 2003), <http://www.dwheeler.com/essays/gpl-compatible.html>.

2. Une nouvelle version de la MPL a été publiée le 3 janvier 2012 et remplace dorénavant toutes les autres licences (elle est expressément compatible avec les licences GNU).

des licences GNU GPL et GNU LGPL lors de la diffusion sous MPL v2 (expressément compatible avec ces mêmes licences) fut immédiat.

En matière de logiciel : l'exemple de Squeak (APSL à MIT). Le logiciel Squeak était un projet collaboratif initialement diffusé par Apple sous licence Squeak. Devant la mise en cause de certaines des clauses de cette licence, la société a accepté de diffuser son code sous licence Apple Public Source License en 2006, puis licence Apache version 2 quelques mois plus tard. Cette cession ne concernait néanmoins pas les contributions individuelles versées sous la licence initiale du projet entre 1996 et 2006. Ainsi, avec l'aide de la Software Freedom Conservancy et la diffusion *a posteriori* d'un accord de relicencierement, une procédure relativement longue a permis de rendre le code disponible intégralement sous licence MIT (avec des portions sous licence Apache) depuis la version 3.10.2 sortie en mars 2010 (la procédure dura ainsi quatre années). Un exemple d'autant plus intéressant que le changement avait pour objectif, atteint, de permettre l'inclusion du logiciel au sein du projet Etoys du programme One Laptop Per Child, ce qui démontre une fois encore la diversité des stratégies ou leviers qui peuvent susciter le choix de certaines licences (et, en l'espèce, l'abandon d'une licence *ad hoc* au profit d'une licence plus utilisée et compatible avec la GNU GPL).

En matière de logiciel : l'hypothèse du noyau Linux (GNU GPLv2 à GNU GPLv3). À titre d'illustration, notons que la question du changement de licence (de la GNU GPL v2 seulement à GNU GPL v3) s'était aussi posée en 2007 lors de la publication de la GNU GPL v3. En effet, Linus Torvald étant clairement opposé à ce nouveau texte et refusant de revenir sur la limitation « *GNU GPL v2 only* » appliquée par lui et un certain nombre des contributeurs au noyau, empêchait Linux d'être distribué sous la troisième version de la GNU GPL, sauf à ce que leurs contributions soient réécrites (ou qu'ils changent d'avis et acceptent ce passage). La communauté de Linux et la renommée de Linus Torvald étant relativement fortes, cette position fut relativement bien suivie. Il fut néanmoins précisé que, compte tenu de la communauté dense et active du noyau, le changement serait tout à fait possible ultérieurement moyennant une substitution de tous les codes pour lesquels le changement de licence ne pourrait pas être obtenu.

En matière de « sites de collaboration multiauteurs » : l'exemple de Wikipédia. Lancée le 15 janvier 2001, l'encyclopédie en ligne Wikipédia utilisa la licence GNU FDL 1.2 sur son contenu durant de nombreuses années. Ainsi, tous les articles pouvaient librement être copiés, modifiés et distribués, ce qui permit à l'encyclopédie collaborative de grossir continuellement d'année en année, jusqu'à devenir une source de référence et une base de connaissance incontournable. Néanmoins, ce succès eut pour conséquence la professionnalisation de la gestion de l'encyclopédie ainsi que des usages qui en étaient faits. Dans ce cadre, le respect de la licence devint une condition indispensable au maintien de la confiance nécessaire aux contributions des milliers d'internautes. Or, le contexte dans lequel cette licence avait été rédigée justifia l'introduction d'une série de contraintes de formalisme lourdes et peu adaptées à l'usage effectif de Wikipédia : la GNU FDL imposait notamment 1) que la licence soit systématiquement recopiée au côté de la création, voire imprimée selon le nombre d'exemplaires tirés et 2) un *copyleft* très fort (calqué sur celui de la GNU GPL) et une série de stipulations complexes (avec notamment des sections invariantes ou encore des obligations différentes selon le format et le nombre de réutilisations du contenu). Le premier point posait certainement quelques difficultés pratiques lors de la reprise d'articles (le texte de la licence étant généralement plus long que l'article lui-même), mais le second emportait des effets juridiques (l'extension de la licence à toute œuvre contenant) non adaptés à l'effet souhaité de la licence.

C'est ce constat d'inadéquation qui motiva le changement de licence de Wikipédia en mai 2009. Ces conditions, dont la violation aurait normalement dû entraîner la résolution de la licence, étaient ainsi très rarement respectées et diminuaient d'autant la capacité des auteurs à faire respecter d'autres points de leur licence. Ainsi, l'ajout d'une Creative Commons CC By-SA 3.0 (plus claire, avec une étendue plus raisonnable et qui impose seulement des mentions légales succinctes ainsi qu'un lien vers la licence) eut pour effet de sécuriser la réutilisation du contenu de Wikipédia. Un tel changement aurait dû passer, à l'instar des autres projets, par une période transitoire de relicencement permettant à chaque contributeur d'accepter ces nouveaux termes. Mais deux écueils ajoutèrent à la difficulté de la démarche : le nombre inégalable de contributeurs à l'en-

cyclopédie (à l'époque à peu près 80 000 contributeurs par mois¹) et la simplification des modalités de contributions (beaucoup de contributions étant par exemple identifiables par la seule adresse IP – temporaire – du contributeur).

La solution différente et originale fut trouvée par l'ajout *a posteriori* d'une compatibilité *ad hoc* au sein de la GNU FDL utilisée. En effet, après avoir envisagé une licence adaptée au Wiki, la FSF accepta, malgré le profond désaccord de Stallman², l'idée de rédiger une nouvelle version transitoire de sa GNU FDL. Publiée le 3 novembre 2008, cette version 1.3 permettait une compatibilité expresse et limitée au profit de la CC-By-SA 3.0, par ailleurs spécialement limitée au seul bénéfice des « sites de collaboration multiauteurs » et pour tout changement opéré avant le 1^{er} août 2009. C'est sur cette base qu'a été modifiée la licence de Wikipédia, sans se passer néanmoins d'une procédure de validation par ses utilisateurs (étant juridiquement insuffisante – il aurait fallu une unanimité pour songer à un autre statut – celle-ci était néanmoins nécessaire d'un point de vue de gouvernance et de communication, pour intégrer la communauté dans ce processus). Aujourd'hui, le contenu est globalement disponible sous CC By-SA et, pour partie, cumulativement sous GNU FDL (un changement de licence qui put se faire, avec l'aide de la FSF, en moins de deux années).

En matière de bases de données : l'exemple d'Open Street Map (de la CC By-SA vers l'ODbL). En 2004, lorsque le projet OpenStreetMap fut lancé, le choix de la licence se porta naturellement sur la variante *copyleft* des licences Creative Commons (la CC By-SA) alors très en vogue. Cependant cette licence, essentiellement basée sur le droit d'auteur, montra rapidement ses limites dans une application à l'égard de données et/ou de base de données, la rendant difficile, voire impossible à mettre en œuvre selon les contextes : un flou juridique nuisible pour la réutilisation des bases de données (notamment par les acteurs économiques) et pour la communauté (puisqu'il faisait courir le risque d'une démotivation consécutive à l'utilisation de ces failles au profit d'un acteur ou projet tiers à la communauté). Il fallut donc s'assurer que la condition essentielle de

1. Voir <http://stats.wikimedia.org/EN/Tables/WikipediansEditsGt5.htm>.

2. Voir Lawrence LESSIG, « Enormously Important News From the Free Software Foundation », dans : *Blog de L. Lessig* (2008).

la licence, son *copyleft*, impose que toute base de données, modifiée ou non, soit distribuée sous la même licence. Cette prise de conscience remonte aux années 2008/2009 et conduisit à la mise en place d'un processus de changement de licence au profit de l'ODbL, soumis à un vote des membres de la fondation OSM (une décision favorable au changement de licence fut actée 26 décembre 2009).

Une procédure de transition en quatre phases fut alors établie par un *License working group* : la phase 1 (10 mai 2010) consista en l'association d'une double licence, CC By-SA 2.0 et OdbL 1.0, aux nouvelles contributions ; la phase 2 (10 août 2010) fut celle de la mise en place d'une procédure permettant aux anciens contributeurs d'accepter la nouvelle licence pour leurs anciennes contributions ; la phase 3 (17 avril 2011) fut l'étape à partir de laquelle la connexion au site et l'utilisation de l'API de la plate-forme imposèrent l'acceptation des termes de l'ODbL ; et la phase 4 (19 juin 2011) consista en la suppression des comptes et des données des utilisateurs n'ayant pas accepté la nouvelle licence. Un robot de nettoyage et de migration vint à cette fin éliminer les données des contributeurs n'ayant pas accepté le changement de licence : ce qui représenta moins de 1% des données à l'échelle mondiale et environ dix fois moins sur le territoire français (une cinquantaine de contributeurs ayant refusé le changement de licence – d'autres communautés locales telles que la Pologne ou la Macédoine furent plus touchées). La dernière étape de cette migration au profit de la licence ODbL 1.0 fut franchie par le projet OpenStreetMap le 12 septembre 2012 (date de la première publication sous OdbL seulement du fichier *planet* contenant un instantané des données – il fallut donc moins de trois années). Ainsi, alors qu'avec l'ancienne licence toute fabrication d'une carte devait être diffusée sous la même licence (sans nécessairement partager les données utilisées pour constituer sa carte ni même encadrer la conception de bases de données dérivées), aujourd'hui les cartes peuvent être produites sans contrainte de licence dès lors que sont mises à disposition toutes les améliorations apportées aux données OSM lors de la fabrication de sa carte.

2.1.2 Les pratiques des utilisateurs : les ajouts aux licences

L'intérêt des ajouts aux licences. Parallèlement, nombre de projets choisirent d'utiliser des licences bien connues en y associant des termes

adaptés à leurs besoins spécifiques. Cela présente un intérêt pour le projet – bénéficiant d'une licence déjà fortement éprouvée et à la sécurité juridique plus forte – ainsi que vis-à-vis de sa communauté puisqu'on bénéficie ainsi de l'image (que l'on souhaite généralement bonne) de la licence ainsi que des efforts de vulgarisation et de formation déjà réalisés sur cette dernière. Sans surprise, car elles restent les licences les plus symboliques, ce sont donc les licences GNU qui bénéficièrent le plus de ces exceptions (la licence MPL initiale étant rédigée pour favoriser l'apparition de nouvelles licences, les ajouts au texte se firent au travers de ces dernières).

Interprétations et exceptions. Deux types d'ajouts sont généralement distingués : l'*interprétation* qui permet de préciser une clause imprécise ou équivoque afin de figer la portée de la licence tenant les parties et le juge ; l'*exception* qui modifie une licence préexistante en ajoutant une spécificité qui déroge aux termes initiaux et qui rend ainsi la licence finale plus ou moins contraignante¹. L'une ou l'autre trouve généralement sa place dans une clause jointe à la licence ou insérée dans les en-têtes des fichiers. Elles sont d'autant plus intéressantes qu'elles témoignent très souvent des effets recherchés par les utilisateurs des licences, divergeant parfois de ceux souhaités par les rédacteurs de ces licences. La majorité d'entre elles concerne la GNU GPL et avaient pour objectif d'adapter (et de réduire) ses effets à un certain type de logiciel, de langage ou encore de modèle économique.

La première interprétation de la GNU GPL. Lors de son utilisation de la licence GNU GPL, Linus Torvald ajouta une interprétation de la notion de *derived work* dans une précision apportée sur chaque en-tête de ses fichiers : « NOTE ! Ce copyright ne s'étend *pas* aux programmes utilisateurs qui utilisent les services du noyau par des appels systèmes normaux – cela est considéré comme une utilisation normale du noyau, et n'est donc *pas* soumis au régime des créations dérivées »². Une précision du même genre fut rédigée par Larry Wall lorsqu'il adopta la GNU GPL v2 dès sa publication en 1989 : « Pour ceux d'entre vous qui choisissez d'utiliser la GNU General Public License, mon interprétation de la

1. À noter que, dans la pratique de la FSF, la notion d'exception se limite à l'ajout de permissions additionnelles (et non de termes potentiellement plus contraignants).

2. Traduction libre, voir <https://www.kernel.org/pub/linux/kernel/COPYING>.

GNU GPL est qu'aucun script Perl tombe sous les termes de la GPL sauf mention explicite de votre part mettant ledit script sous les termes de la licence GPL »¹. L'une et l'autre sont célèbres et toujours d'actualité. En pratique, elles favorisent le développement d'applications accessoires ou dépendantes et sont considérées comme intimement liées au succès de ces logiciels. Cet usage semble néanmoins témoigner d'une époque révolue, la légitimité de ces auteurs égalait alors celle des organisations telles que la FSF et de telles interprétations étaient non seulement acceptées, mais aussi difficilement contestables. Dorénavant, ce type de précisions tendrait à prendre la forme d'exception (la doctrine s'étant en effet développée de telle sorte qu'il serait alors plus complexe d'expliquer les raisons d'une interprétation divergente plutôt que de simplement modifier la portée de la licence). L'interprétation sera donc limitée aux seules zones grises de la licence, le cas échéant, et dans l'absence d'interprétation publiée par le rédacteur de la licence lui-même.

Les exceptions associées aux « générateurs de codes ». Certains logiciels susceptibles d'intégrer leurs propres codes au sein des créations générées utilisèrent la technique des exceptions pour encadrer l'étendue de la licence afin que tout produit généré par ces outils puisse être utilisé conformément aux choix de leurs auteurs en termes de licence. Ainsi, certains fichiers du compilateur GCC (libgcc², libstdc++, libfortran, libgomp, libdecnumber, libgcov, etc.) se voient associer une exception *Runtime library exception* qui permet de produire des programmes sous d'autres licences que la seule GNU GPL (l'analyseur syntaxique GNU Bison s'est vu ajouté une exception similaire en 1996). À l'inverse, et cela est révélateur de l'aspect stratégique de la licence, le studio de développement EiffelStudio intègre des fichiers sous GNU GPL qui imposent aux utilisateurs l'utilisation de la même licence (sauf à ce qu'ils se tournent vers la licence commerciale).

La limitation de l'étendue de la GNU GPL au travers de la *GPL linking exception*. La *GPL linking exception* est aujourd'hui une exception « générique » réutilisée à l'identique, ou dans une forme similaire, par

1. Traduction libre, voir <http://dev.perl.org/licenses>.

2. Il s'agit peut être du premier composant pour lequel Stallman décida d'utiliser une exception : libgcc.c permettant d'implémenter des opérations primitives arithmétiques sur des machines qui n'avait pas d'instructions pour ces opérations, il considéra que cela ne devait pas emporter d'effet à l'encontre des programmes utilisateurs.

de nombreux autres projets. Ainsi, la *GPL linking exception* est utilisée sur des bibliothèques désirant une licence proche de la GNU LGPL en ce qu'elle permet expressément les liaisons entre programmes, mais sans requérir de conditions particulières, la GNU LGPL étant en effet relativement exigeante quant à la fourniture des informations nécessaires (même si non soumis au copyleft, certaines informations du code utilisant le composant sous GNU LGPL devront être fournies afin de permettre à l'utilisateur de modifier et tester des versions modifiées de ce dernier). Elle est actuellement utilisée par des projets comme GNU Guile, GNAT ou GNU Classpath – l'*Exception classpath* utilisée sur le projet GNU Glasspath (plus tard réutilisée sur Java par Sun) est sans doute la meilleure illustration de l'utilité de l'exception puisqu'elle permet de lier du code à la bibliothèque sans étendre la GNU GPL au produit final. Dans le même ordre d'idée, on peut évoquer la wxWindows Library License publiée en 1998 qui, même si elle porte un nom, consista en l'ajout d'une permission à la GNU Library General Public License pour autoriser la distribution sous forme propriétaire le code objet de logiciels ou bibliothèques basés sur la licence (s'inspirant en cela de la MPL). Plus généralement, les noyaux logiciels semblent majoritairement avoir été diffusés – à l'instar du noyau Linux – sous des licences permettant expressément l'utilisation ou l'interaction avec des applications : qu'il s'agisse de l'eCos License 2.0 qui remplaça la Red Hat eCos Public License (ou RHEPL, une licence dérivée de la MPL) pour limiter l'effet de la licence GNU GPL aux seuls fichiers d'origine ou la licence du système d'exploitation Real-Time Executive for Multiprocessor Systems (ou OS RTEMS) qui se compose de la GNU GPL (version 2 ou supérieure) avec une exception particulière inspirée de l'exception GNAT Modified General Public licence (permettant ainsi la liaison avec d'autres composants). Enfin, notons que la GNU LGPL est elle-même une exception à la GNU GPL depuis leur version 3.

La limitation de l'étendue de la GNU GPL au travers de la FOSS exception. La *FOSS exception* (qui recoupe aussi différentes variations) permet de lever les problématiques d'incompatibilité entre licences dans un contexte *open source* seulement. Les composants *open source*, pour autant que leur licence figure dans la liste annexée à cette exception, peuvent ainsi continuer à être régis par leur licence d'origine même s'ils sont combinés ou intégrés au logiciel sous licence GNU GPL. Son utilisation fut

banalisée par MySQL, la licence GNU GPL + *FOSS Exception* remplaçant en 2004 la LGPL auparavant utilisée. L'intérêt d'une telle démarche (poussée en interne par Zak Greant) est de maintenir le bénéfice d'une diffusion et réutilisation dans un mode *open source* tout en conservant les contraintes de la GNU GPL pour une utilisation en dehors de l'*open source* (MySQL reste utilisé par de nombreux programmes sous licences propriétaires/commerciale). Même si les idées initiales restent les mêmes, on retrouve de nombreuses variantes à cette exception, se distinguant généralement par le nombre et le type de licences compatibles : par exemple, DigiRule Solutions's Free and Open Source Software (FOSS) License Exception, Kyoto Products FOSS License Exception (qui liste 11 licences), Hamsterdb FOSS License Exceptions (qui liste 8 licences), Stanford University FOSS License Exception v1.0 (qui en liste 12), etc.

Les exceptions dédiées aux langages. De nombreux concepteurs de langage ont reproché à la GNU GPL d'être trop inspirée des mécanismes du langage C, un héritage qui empêcherait de prendre en compte les spécificités des autres langages. Certains langages adoptèrent ainsi leur propre licence (la licence Artistic pour Perl en 1991, la licence Ruby de 1995, la licence PHP en 1995 ou encore la licence Falcon en 2003), mais la plupart se tournèrent vers les licences les plus connues (GNU GPL en tête, mais aussi MPL, BSD, EPL, Affero GPL, etc.). Parmi ceux qui adoptèrent une licence GNU (AGPL, GPL ou LGPL), plusieurs ajoutèrent une exception : telle la L LGPL (Lisp Lesser General Public Licence) rédigée dès 2000 pour le langage Lisp, la GNU Classpath Exception utilisée pour Java (la Enhydra Public License fut rédigée avec les mêmes objectifs), la GNAT Modified General Public License pour Ada (faisant suite à la création de l'entreprise par les fondateurs du projet en 1994), etc.

L'ajout de licences aux licences. Dans la même idée, beaucoup d'utilisateurs de licence s'aperçurent qu'il serait plus radical, et parfois plus simple, d'ajouter une licence réputée aux côtés de leur licence spécifique. On parle alors de multilicence (une seule et même création est simultanément disponible sous différentes licences) ou « double licence » lorsque la licence libre n'est qu'une option au côté de licences propriétaires classiques. Une telle solution présente un certain nombre d'aspects négatifs et positifs. Parmi les désavantages, il convient de préciser : la complexité qu'un tel schéma peut revêtir pour un utilisateur non averti (il est cepen-

dant tout à fait possible de ne communiquer que sur l'une seule des licences et de n'évoquer les autres licences qu'en présence de personnes ou projets susceptibles d'y être attentifs); la nécessité de distribuer sous les mêmes licences toutes les contributions souhaitant être réutilisées dans le projet; le risque accru concernant les *forks* (division de la communauté en deux projets distincts) qui deviennent encore plus dangereux et contreproductifs puisque les différents projets sont susceptibles d'être diffusés sous des licences incompatibles (au détriment du seul projet d'origine ou des deux). La solution a aussi son lot d'avantages : une meilleure compatibilité *downstream* (c'est-à-dire permettant la réutilisation du code au sein d'autres projets), le bénéfice de la renommée de toutes les licences, une consolidation des cessions de droits et, surtout, le maintien d'une mécanique propre à la licence initiale tout en permettant à d'autres communautés de profiter de ces développements. Ainsi, le langage de programmation Falcon, initialement sous licence Falcon Programming Language License (une licence proche de la licence Apache v2 optimisée pour Falcon, le langage de programmation de scripts, modifiée en 2008 pour empêcher l'extension de la licence aux scripts) a ensuite été cumulativement diffusé sous GNU GPL version 2 dès la version 0.8.10; la licence Artistic a doublé la distribution du langage Perl sous licence GNU GPL; la MPL s'est vu adjoindre les GNU GPL et LGPL dans les années 2000.

L'action des Commercial Open Source Software (COSS). Ce terme est parfois utilisé pour désigner des projets édités par des éditeurs commerciaux qui cherchent à diffuser une version *open source* tout en valorisant de façon classique des fonctionnalités (logicielles ou non) plus évoluées pour ses clients (créant des *vendor lock-in* – des mécanismes rendant dépendant d'un éditeur – en tout point similaires à ceux des logiciels traditionnels). On observe que, depuis quelques années, le modèle de double licence poussé par MySQL, Talend, Trolltech, etc. (qui proposaient une licence commerciale alternativement à une licence *open source*) perd en intérêt. En effet, si les clients industriels « achètent » volontiers pendant quelques années la sécurité juridique par ce biais (agissant ainsi sur recommandation de leurs conseils juridiques), ils se tournent aujourd'hui de plus en plus vers les versions communautaires dont ils perçoivent mieux les avantages ainsi que les conditions. Néanmoins, les éditeurs perfectionnent et maximisent aussi les marges de manœuvre dont ils dis-

posent du fait des licences libres, s’approchant parfois dangereusement de la frontière entre libre et fermé (ces mêmes éditeurs associent par exemple à la plupart de leurs contrats – services, assistance et support, avec les distributeurs, etc. – des clauses par lesquelles les cocontractants s’empêchent de bénéficier ou promouvoir la version communautaire libre).

2.2 Une harmonisation par la modification et consolidation des licences

2.2.1 Une consolidation par la GNU GPL

Le rôle complexe de la GNU GPL. La GNU GPL joue un rôle compliqué dans la consolidation de l’écosystème des licences, alliant deux paradoxes. Le premier concerne le rôle (et la légitimité conséquente) de la GNU GPL : à la fois fondatrice et « leader historique » qui tend à réunir les autres licences sous sa coupelle et « extrémiste du libre » qui, par sa position souvent inconditionnelle, détourne parfois certains utilisateurs (raison pour laquelle Linus Torvald rejeta la dernière version de la GNU GPL). Le second paradoxe concerne l’étendue très forte de la GNU GPL qui, contrairement aux licences de type MPL, emportait l’extension de la GNU GPL sur des composants initialement soumis à d’autres licences (ce qui était relativement agressif ou contraignant pour les développeurs). Cette étendue forçait la compatibilité sans laisser de réelle marge de manœuvre aux rédacteurs de nouvelles licences (beaucoup de licences furent ainsi modifiées pour devenir compatibles avec la GNU GPL : la BSD en 1999, l’Eiffel Forum License v2 en 2008, etc.). Au fur et à mesure de son adoption, la licence s’est néanmoins détachée graduellement du contrôle de la seule FSF et le juriste Eben Moglen assista Richard Stallman dans la rédaction des licences qui suivirent la GNU GPL v2 (personnellement dans un premier temps et ensuite par l’intermédiaire du Software Freedom Law Center qu’il dirige). La troisième version de la GNU GPL vient alléger les deux paradoxes évoqués.

Une conservation de l’esprit initial. Il est ainsi intéressant d’écouter Eben Moglen expliquer qu’en dépit des modifications importantes que connut la GNU GPL, la dernière version conserve l’esprit initial qu’elle replace dans l’époque actuelle :

La GPLv3 transpose la constitution de RMS pour le logiciel libre dans le nouveau siècle. Elle rend les principes juridiques du logiciel libre véritablement mondiaux, non centré sur le seul droit américain. Elle a renforcé les défenses de logiciels libres contre les attaques en matière de brevets, juste à temps pour la guerre des brevets que nous avons depuis longtemps prédit en cas d'acceptation du brevet logiciel. Elle empêche aussi l'emprisonnement des logiciels libres dans des produits matériels qui ne permettent pas à leurs propriétaires de modifier le logiciel incorporé. La GPL v3 a anticipé les enjeux d'aujourd'hui et nous aidera à relever les défis de demain¹

En pratique, cette nouvelle version opère des changements considérables afin de maintenir l'objectif initial (la liberté des utilisateurs dans une société qui a évolué), tout au plus pourrait-on faire remarquer que ces évolutions dépassent parfois les attentes des utilisateurs de la licence, mais force est de constater que le travail fourni par la FSF répond à la mission dont elle a, plus ou moins implicitement, la charge : plus que de faire évoluer un texte juridique, elle a pour mission de protéger les libertés des utilisateurs.

Les problèmes posés par la faille des logiciels utilisés en tant que service (ASP loophole). Conçues à l'époque où les logiciels s'échangeaient encore principalement sur support magnétique, les premières licences libres associaient l'acte de distribution à celui de la sortie de la sphère privée (emportant les contraintes nécessaires pour assurer les libertés des utilisateurs subséquents). Néanmoins, avec le développement du réseau, le nombre de logiciels libres utilisé en « Software as a Service » (Saas), c'est-à-dire comme un service², devint de plus en plus important. En l'absence de distribution du code, de nombreux géants du web ont pu construire leurs services sur la base de composants de logiciels libres sans contribuer directement à leur propre modification, cassant la confiance de certains développeurs pour qui un tel usage aurait dû être couvert par la GNU GPL. Au début des années 2000, plusieurs nouvelles licences vinrent encadrer cet usage : l'Apple Public Source License (APSL) publiée en 1999, la Reciprocal Public License publiée en 2001 par Scott

1. Traduction libre des propos retranscrits par Josh Gay le 29 juin 2012, <http://www.fsf.org/blogs/licensing/gnu-gplv3-turns-5>.

2. Installés sur les serveurs d'une entreprise, ces logiciels permettent d'offrir des services à une multitude des clients de cette dernière sans être matériellement accessibles aux utilisateurs finaux.

Shattuck (de la société Technical Pursuit Inc.) et mise à jour (version 1.5) en 2007, l’Affero Public License publiée en mars 2002 par la société Affero avec l’aide de la FSF ou encore l’Open Software License en 2002 (d’autres suivirent et reprirent ensuite ces clauses, telles que la Common Public Attribution License (CPAL) en 2007 ou l’European Union Public Licence lors du passage en version 1.1 en 2009). Renonçant à son idée de l’intégrer directement au sein de sa nouvelle GPL (mais le projet figurait encore dans l’article 7b.4 de son second brouillon¹), la FSF conçut une nouvelle licence, la GNU Affero GPL, qui contient un article 13 obligeant le licencié qui utilise le logiciel pour fournir un service à des utilisateurs via le réseau à redistribuer les modifications qu’il aurait lui-même apportées à ce dernier.

L’encadrement de mesures techniques de protection et la Tivoïzation. Le terme Tivoïzation est un néologisme définissant la création d’un système qui inclut des logiciels libres, mais utilise le matériel électronique pour interdire aux utilisateurs d’y exécuter des versions modifiées ne disposant pas des signatures électroniques du constructeur (technique initialement mise en œuvre par TiVo sur ses enregistreurs vidéos numériques, mais ensuite réutilisés pas d’autres constructeurs tels que Tomtom pour les GPS). La GNU GPL v3 intègre et encadre ce dispositif en interdisant expressément de limiter l’usage d’un produit embarquant du code sous GNU GPL v.3 à la seule version fournie par le constructeur (sous peine de perdre le bénéfice de la licence). Cette clause est strictement réservée aux « produits de consommation » (donc non applicable à un usage interne ou entre professionnels), mais a un impact concret dans le domaine de l’embarqué (de nombreux constructeurs refusant de permettre une telle mise à jour par l’utilisateur). Torvalds s’est catégoriquement opposé à cette clause qui limite certaines utilisations du logiciel (alors qu’à

1. Prévue pour la version 3 de la GNU GPL, le rejet de la clause Affero (qui aurait eu un impact important et difficilement maîtrisable sur les logiciels utilisés en Saas puisque cet usage serait devenu « déclencheur » de la licence, entraînant l’obligation de mettre à disposition les sources ainsi que toute modification apportée au logiciel) entraîna de violentes accusations (certains n’hésitant pas à dire que « la GPL est la nouvelle BSD », voire à lui affubler l’appellation de « SHING GPL » en référence à « Sun HP IBM Nokia Google », principaux bénéficiaires de cette clause).

ses yeux la licence doit au contraire favoriser les usages) et refuse depuis lors de changer de licence au profit de la dernière version¹.

La résolution immédiate. Comme évoqué précédemment, la sanction initiale à toute violation de licence libre était initialement très radicale puisqu'elle entraînait la résolution pure et simple de la licence, rendant contrefacteur l'utilisateur quant à tous les usages ultérieurs. Utile pour faire respecter les licences, cette sanction était néanmoins parfois trop sévère et faisait courir un risque important au secteur industriel susceptible (comme tout autre²) de violer une licence par ignorance. Ainsi, un mécanisme d'atténuation à progressivement été introduit au sein des licences quant aux effets d'un tel non-respect, permettant au licencié de corriger son comportement, voire d'être automatiquement réintroduit dans ses droits, à certaines conditions. La FSF a intégré cette attente en organisant au sein de la GNU GPL v.3 un système de « rétablissement des droits » lorsqu'il s'agit d'une première violation et que celle-ci est rapidement corrigée. Par la suite, la FSF communiqua sur l'importance de ce changement face aux nouveaux usages industriels réalisés à partir du noyau Linux (Android en premier lieu) afin d'inciter le projet Linux à utiliser la dernière version de la licence³. Sans succès.

La recherche d'une meilleure sécurité juridique et l'enjeu en matière de brevets. Bénéfice sans doute le plus important de l'apport des licences industrielles, la recherche d'une meilleure sécurité juridique est commune à la majorité des licences qui se succédèrent dans les années 2000 : internationalisation des licences, meilleure définition de l'œuvre dérivée pour les licences *copyleft*, généralisation des licences en matière de brevets, stipulation relative à la loi compétente, encadrement des mesures techniques de protection, etc. L'accélération de la brevetabilité des logiciels (de façon explicite aux États-Unis et Japon, de façon masquée en Europe où les logiciels ne peuvent être breveté en tant que tels) fut certainement vu comme le plus grand danger des dernières années : beaucoup

1. Voir à cet égard Linus TORVALDS, « Re : Dual-Licensing Linux Kernel with GPL V2 and GPL V3 », message sur la liste de diffusion *Linux Kernel Mailing*, 13 juin 2007.

2. Citons à cet égard Richard Stallman qui, en 2011, admis que plusieurs versions de son logiciel GNU Emacs ne respectaient pas la GNU GPL (« *Emacs distributions are not GPL-compliant* ») <http://lists.gnu.org/archive/html/emacs-devel/2011-07/msg01155.html>. Il est toujours bon de se rappeler que même les meilleurs sont faillibles.

3. Voir Brett SMITH, « Android GPLv2 termination worries : one more reason to upgrade to GPLv3 », 18 août 2011, <http://www.fsf.org/news/android-termination-upgrade-gplv3>.

plus difficiles à contourner, voire même à identifier, ils laissent planer une menace constante qui nuit à la pérennité (et donc l'adoption) des logiciels libres. Les sociétés peuvent se protéger contre un tel risque en développant elles-mêmes des portefeuilles de brevets défensifs, mais une telle pratique n'est pas possible pour un projet communautaire (même s'il est possible de se positionner contre de tels brevets ainsi que l'a publiquement fait Mozilla en 2004) qui peut au mieux bénéficier de la protection (plus précaire) de grands groupes (il s'agit d'initiatives de regroupements de brevets à titre défensif – *patent pools* individuels¹ ou collectifs² – avec pour objectif d'immobiliser toute action tierce en contrefaçon de brevets logiciels : on les intitule *patent angels*, *anti-patent trolls* ou encore *reverse patent troll*). Ainsi, si la FSF s'est très tôt positionnée contre les brevets logiciels (dès 1990, intégré l'année suivante dans les GNU GPL et LGPL), ce n'est que dans les années 2000 que les clauses des licences relatives aux brevets furent améliorées de sorte à encadrer le risque afférent (pour les industries comme les communautés) et mettre un terme à la campagne de FUD (*Fear, uncertainty and doubt*) menée contre les logiciels libres par certains éditeurs. Un effort similaire fut donc prévu dans le cadre de la nouvelle GNU GPL, d'autant plus qu'un nouvel artifice fut révélé dans l'accord conclu entre Microsoft et Novell puisque, Novell ne pouvant bénéficier d'une licence sur les brevets de Microsoft sans partager ce bénéfice, une situation préférentielle fut organisée afin qu'elle puisse valoriser (commercialement et en termes de communication) la promesse faite par Microsoft de ne pas agir à l'encontre des clients de sa distribution Suse. L'article 11 de la GNU GPL v3 vient donc améliorer la portée de la licence en matière de brevets (relative à toute invention contenue dans la version contributive) et étendant ses effets aux bénéficiaires de promesses de non-agression (qui doivent alors choisir entre rendre le code source public, renoncer au bénéfice de la licence ou étendre celui-ci).

1. Plusieurs milliers de brevets sont ainsi regroupés en garantie par des sociétés comme IBM, Nokia, Sun, etc., notamment l'IBM Statement of Non-Assertion of Named Patents Against OSS, le Sun Patent Program ou encore le Novell Statement on Patents and Open Source Software.

2. Notamment l'Open Source Development Laboratory (OSDL), l'Electronic Frontier Foundation (EFF, avec notamment l'initiative Patent Busting Project) ou l'Open Invention Network (OIN) auquel ont souscrit des sociétés comme Sony, IBM, NEC, Red Hat, Philips et Novell. La Foundation for a Free Information Infrastructure (FFII) et l'initiative *no software patents* affirmant que la constitution de regroupements de brevets à titre défensif est une mauvaise solution par rapport à la lutte pour imposer la non-brevetabilité des logiciels.

Une prise en compte des nécessités d'une meilleure compatibilité. Si l'on en croit les études publiées par les sociétés éditrices d'outils d'audit de code, il y aurait plusieurs milliers de licences différentes actuellement utilisées. En pratique, il s'agit de la centaine de licences répertoriées par l'Open Source Initiative, auxquelles s'ajoutent les versions antérieures et les modifications apportées par le biais d'interprétation ou d'exception. La licence GNU GPL est en tête en termes de couverture de code, mais le résultat est plus équilibré, notamment au profit de licences telle que la licence Apache, lorsqu'on se focalise sur les logiciels les plus importants. Cette multiplication de licences destinées à libérer l'utilisateur du contrôle des éditeurs aboutit néanmoins à une difficulté inhérente à la technique utilisée (le contrat). D'autant plus paradoxale au regard des objectifs des différentes licences, la notion d'incompatibilité décrit la situation où les termes contradictoires de licences libres entraînent l'impossibilité de les respecter simultanément et, conséquemment, de combiner plusieurs composants libres (soit en raison de leurs spécificités, soit parce qu'elles sont elles-mêmes *copyleft*). À l'instar de l'ASP *loophole*, la problématique de l'incompatibilité a longtemps été laissée dans une zone grise. Des licences comme l'Apache License ou l'Artistic License étaient considérées comme compatibles avec la GNU GPL par leurs rédacteurs respectifs alors qu'elles n'avaient pas été rédigées comme telles (l'incompatibilité la plus symptomatique, et longtemps problématique, fut par exemple entre la licence GNU GPL version 2 et l'Apache version 2, cette dernière contenant une clause plus protectrice en matière de brevet qui n'aurait pas été respectée si le composant s'était retrouvé couvert « seulement » diffusé sous GNU GPL). La nécessité de trouver une issue à ces enjeux n'est finalement que du bon sens et les rédacteurs de licences se sont aussi emparés du sujet (les exceptions FOSS ou *linking* sus-citées avaient une intention similaire), considérant que « le point important n'est pas les différences, mais au contraire le but commun ¹. Aujourd'hui, la compatibilité est assurée au sein de plusieurs licences par différents mécanismes : l'existence d'une clause expresse (la GNU LGPL v2.1 permet, par exemple, une diffusion sous GNU GPL v2), la compatibilité conditionnée (telles la CeCILL ou l'EURL qui permettent une rediffusion sous GNU GPL en cas de

1. Tiré d'un courrier de Lawrence Lessig, disponible sur <http://creativecommons.org/weblog/entry/5709>.

combinaison avec des composants sous cette licence). Depuis sa dernière version, la GNU GPL v3 contient un article spécifique destiné à renforcer une compatibilité à son profit, c'est-à-dire la capacité à respecter un nombre accru de licence en diffusant sous GNU GPL v3. Parmi les autres licences remarquables quant à cet aspect, il est possible d'évoquer l'European Union Public License v1.1 compatible avec cinq licences libres *copyleft* (l'OSL v.2.1 et v.3.0, la CPL v1.0, l'EPL v.1.0, la CeCILL v2.0 et la GNU GPL v.2.0), la nouvelle version de la Mozilla Public License publiée en 2012 et rédigée pour être compatible avec la licence Apache, ainsi que les licences GNU (Affero, LGPL ou GPL). La seconde version de l'EURL a pour objectif principal d'étendre la liste des licences compatibles. De la même façon, Creative Commons travaille aussi sur une compatibilité avec des licences proches de la CC By-SA (notamment la Licence Art Libre ouverte à une telle compatibilité depuis sa version 1.3 à condition d'une réciprocité). Malheureusement, la notion de compatibilité elle-même diverge selon que la licence soit très utilisée (telle la GNU GPL ou les Creative Commons qui cherchent surtout à favoriser l'usage de leur licence par ce biais) ou plus spécifique (telles que l'EURL ou la CeCILL qui souhaitent surtout que le code puisse être repris par d'autres projets, quitte à ce que la licence soit changée au profit d'une autre compatible).

2.2.2 Un processus de rationalisation : abandon et sélection

Le rôle des organisations du secteur. Les principales organisations du secteur ont joué un rôle essentiel : que ce soit la FSF, le SFLC et la SFC en accompagnant l'usage de licence GNU ; l'OSI en entamant une campagne contre la prolifération des licences, mais aussi toutes les autres organisations qui, par leur structuration, ont favorisé la diffusion de « bonnes pratiques » (telles que les Fondations Apache ou Eclipse qui conditionnent l'adhésion au respect d'une certaine politique en matière de licence). Dans les années qui suivirent, d'autres entreprises ont aussi joué le jeu : Sun a retiré sa licence SISSL en 2005 et SPL en 2006, IBM fit de même pour la licence CPL en 2009 (ainsi que l'Intel Open Source License, la Jabber Open Source License, la MITRE Collaborative Virtual Workspace License).

Le travail mené au sein de la fondation Creative Commons. Alors que le projet comptait à son lancement onze licences différentes en 2002,

ce nombre a été réduit dès 2004 avec la publication d'une seconde version des licences *Creative Commons* utilisée sans l'option d'attribution, répondant d'une pierre deux coups au faible attrait constaté pour ces licences ainsi qu'aux difficultés rencontrées lors de l'internationalisation des licences puisque l'option ne pouvait être retenue dans les pays accordant une place aux droits moraux de l'auteur. Parallèlement, certainement plus par naïveté que calcul, Creative Commons a incité et parfois accompagné de nombreuses demandes en rédigeant un certain nombre de licences spécifiques au *sampling* fin 2003 (*Sampling*, *NonCommercial-Sampling+*, *Sampling+*), aux pays en voie de développement (*DevNations*, une *CC By* limitée aux pays en voie de développement) en septembre 2004, ainsi qu'un projet non abouti en matière d'éducation en 2003. Les licences *DevNations* et la *Sampling License* furent retirées en 2007, ne répondant pas aux libertés minimales de diffusion non commerciale souhaitée par le projet. Les autres furent abandonnées en 2011 (la *PDD* fut remplacée par la *CC-zero* ainsi que la *Public Domain Mark*) en raison de leur faible utilisation, d'une volonté de clarté et de la recherche d'une meilleure compatibilité entre les licences. La *Creative Archive Licence* a ainsi permis la publication, de 2005 à 2006, d'un certain nombre de créations en provenance d'établissements publics britanniques (*BBC*, *British Film Institute*, *Université Ouverte*, etc.) afin de permettre une réutilisation non commerciale particulièrement encadrée (ainsi, celle-ci était limitée au seul territoire du Royaume-Uni et interdisait tout usage promotionnel ou politique).

Le projet de lutte contre la prolifération lancé par l'OSI. Fin 2007, l'*Open Source Initiative* a lancé un nouveau projet (complémentaire à celui de labellisation) de lutte contre la prolifération des licences. À cette fin, les licences sont depuis cette date classées en plusieurs catégories : 1) les licences qui sont populaires et largement utilisées ou avec une large communauté ; 2) les licences répondant à des besoins spécifiques/particuliers ; 3) les licences qui sont redondantes avec les licences les plus populaires ; 4) les licences qui ne sont pas réutilisables ; 5) les licences autres/diverses. Ce projet répond à une demande formulée dès 2004 au sein des administrateurs de l'OSI et qui avait donné naissance à un rapport publié en 2006 (*Report of License Proliferation Committee and draft FAQ*). Ce travail a permis d'identifier neuf licences « populaires » (*Apache License 2.0* ; *BSD 3-Clause New or Revised License* ; *BSD 2-Clause Simplified or*

FreeBSD License ; GNU General Public License (GPL) ; GNU Library or *Lesser* General Public License (LGPL) ; MIT license ; Mozilla Public License 2.0 : Common Development and Distribution License ; Eclipse Public License). L'OSI a ainsi contacté de nombreuses organisations afin de procéder à cette réduction du nombre de licences actives. Cette nouvelle mission que s'est attribuée l'OSI, tout à fait louable, a néanmoins froissé un certain nombre de personnes puisque la sélection « arbitraire » du comité en charge avait ensuite un impact sur l'usage de la licence (ainsi Lawrence Rosen, alors qu'il n'était plus conseil juridique de l'OSI, s'était clairement manifesté contre la classification de ses deux licences dans les catégories des licences « redondantes » ou « autres »).

Une obsolescence de fait. L'évolution des usages et des technologies ne cesse de remettre en cause nos paradigmes. De ce fait, un nombre croissant des licences utilisées devient obsolète sans que leurs rédacteurs de l'époque n'aient de réel contrôle sur celles-ci (la licence Artistic est à cet égard un exemple éclairant puisqu'en l'absence de clause permettant la mise à jour de la licence, deux versions cohabitent sans que le passage de l'une à l'autre ne puisse se faire aisément¹). Une « sélection naturelle » a amené les utilisateurs à privilégier le choix de licences mieux connues et assurant une meilleure compatibilité avec les projets existants.

2.3 L'importance du perfectionnement des processus

L'introduction d'une édition collaborative des licences. La majorité des licences utilisées par des communautés furent discutées avec celles-ci avant d'être définitivement publiées². Ainsi Stallman partagea-t-il sa GNU GPL avant de la publier ou Netscape fit circuler des brouillons avant toute publication. De nombreuses autres tombèrent néanmoins dans

1. Notons à cet égard que la fondation Perl considère, sans nécessairement le justifier pleinement, la seconde version de l'Artistic comme étant compatible avec la première et que Larry Wall accepte volontiers de relicencier ses contributions sous Artistic v2 (mais cela ne couvre malheureusement que ses propres contributions).

2. Étonnamment, l'importance des communautés dès l'édiction des licences n'a pas été aussi naturelle que l'on pourrait le penser. Voir notamment, au sujet des licences CC Sampling, Victor STONE, *ccMixer : A Memoir OR How I Learned to Stop Worrying about the RIAA and Love the Unexpected Collaborations of Distributed Creativity During the First Four Years of Running ccMixer*, février 2009, URL : <http://fourstones.net/ccmixer-a-memoir>.

le premier travers du droit qui est d'en faire une discipline réservée (quitte à ensuite connaître plusieurs révisions successives pour répondre aux attentes de chacun). Cependant, l'implication des communautés tend à se développer de plus en plus tôt : passant d'une discussion du texte final à une discussion de brouillon et enfin une consultation ouverte. Au fur et à mesure que les projets gagnent en importance et en contributeurs, il devient de plus en plus difficile, ou pour le moins très critiquable, d'imposer tout changement sans consultation des auteurs/utilisateurs de ces licences. En effet, le rédacteur se trouve très souvent dans la situation très particulière où il dispose de la faculté de changer un contrat utilisé par d'autres, en totale contradiction avec l'effet relatif des contrats selon lequel la convention à force de loi entre les parties (dès lors qu'il ne porte pas atteinte aux bonnes mœurs ou à l'ordre public) et n'est donc opposable qu'aux seules parties au contrat (seule une personne ayant consenti à un contrat peut se le voir opposer). D'une rédaction individuelle, le processus est devenu collectif, renforçant d'autant la légitimité et la force de ces licences comme outils de gouvernance des communautés. Par ailleurs, notons que même pour les organisations ayant anticipé contractuellement une telle faculté (par une cession exclusive des droits des contributeurs à leur profit ou grâce à une clause de la licence prévoyant une mise à jour de la licence à la discrétion du rédacteur de la licence), elles n'en usent généralement qu'après consultation et accord de leurs communautés. Les processus les plus aboutis furent certainement ceux mis en place pour la troisième version de la GNU GPL. La Free Software Foundation initia en 2006 une consultation des auteurs et communautés afin de comprendre les enjeux auxquels devait répondre la GNU GPL et les attentes des utilisateurs (notamment au regard d'un premier brouillon fourni en janvier 2006). Grâce à des outils dédiés permettant de récupérer, discuter et consolider tous les commentaires, un certain nombre d'attentes furent prises en compte : concernant les évolutions législatives, techniques, pratiques et industrielles. Après quatre itérations (chacune donnant lieu à une version consolidée et revue par Richard Stallman, Eben Moglen et Richard Fontana), la version finale de la GNU GPL v3 fut publiée le 29 juin 2007 et présentée comme la licence conforme à l'esprit d'origine, mais formalisant 16 ans de changements et acceptant un certain nombre de compromis pour répondre aux attentes des utilisateurs industriels comme

communautaires (la clause Affero fut certainement la plus polémique). Ce procédé fut repris dans des conditions assez semblables lors de la mise à jour de la Mozilla Public License en 2011. Ce processus collaboratif est aujourd'hui poussé encore plus loin au travers d'une initiative nommée *GPL.next* par Richard Fontana sur GitHub ¹.

Le processus de certification des licences a lui aussi grandement évolué. Aujourd'hui, quatre principes fondamentaux permettent de vérifier l'adéquation avec l'Open Source Definition : identifier la catégorie parmi celles identifiées par le projet de *License Proliferation* et essayer de limiter la duplication de licences (voir *infra*) ; et assurer une procédure transparente et pérenne d'examen des licences (dans les 60 jours en moyenne) et fournir une traçabilité du processus d'examen. De la même façon, cette structuration renforce l'intérêt de cette certification. La requête – en principe, faite par le rédacteur de la licence, mais le respect de cette règle est relatif – doit identifier le type de soumission, vérifier que tous les critères sont réunis pour cette requête et être complète (sujet clair, texte de la licence, demande d'inscription au système de suivi et toute autre information utile). Le demandeur doit lui-même s'inscrire à la liste de diffusion correspondante (*license-review*). Il existe plusieurs types de soumission d'une licence : lors de la mise à jour d'une licence (il faut alors inclure un lien vers l'ancienne, la nouvelle, et les modifications apportées à la licence), pour l'ajout dans une nouvelle catégorie, pour retirer une version, pour l'approbation d'une version déjà existante (il faut indiquer la catégorie et les justificatifs) ou pour l'approbation d'une nouvelle licence – ou licence utilisée par une seule entité (il faut soumettre la préversion, justifier, comparer vis-à-vis des licences existantes, recommander une catégorie). Par la suite, la certification de la licence est discutée sur la liste dédiée au moins une trentaine de jours (l'outil de suivi de l'OSI fournissant le statut approprié de chaque tâche) et une synthèse des recommandations est présentée au bureau de l'OSI (avec copie à la liste) par la personne en charge de la labellisation des licences. Le bureau de l'OSI dispose alors du pouvoir de prendre une décision finale ou de demander des informations complémentaires. Pour finir, la personne en charge des

1. Service web d'hébergement et gestion de versions décentralisée permettant à chacun de créer sa propre version tout en suggérant (poussant) les modifications que l'on souhaiterait voir reprendre dans la version principale.

licences informe la liste et, si approuvée, la licence est ajoutée au site de l'OSI.

Une diffusion des bonnes pratiques au sein des structures. S'apercevant de l'importance de la dimension sociale de ces licences, les différentes structures qui forment le monde du Libre se sont par ailleurs mieux organisées, notamment dans la répartition de leurs tâches en privilégiant la complémentarité et la transparence. Au regard de son âge et de ses missions, la Free Software Foundation s'est ainsi rapidement appuyée sur un certain nombre de « structures sœurs » qui complétaient ses tâches : tels le Software Freedom Law Center (SFLC) créé en 2005 (réellement opérationnel à partir de 2007) qui assure, sous la direction d'Eben Moglen, l'assistance juridique aux organisations à but non lucratif et la diffusion de connaissances juridiques au travers d'une activité de publication, ou encore la Software Freedom Conservancy créée en 2006 pour aider à promouvoir, améliorer, développer et défendre des projets de logiciels libres et *open source* (tels des projets réputés comme Amarok, BusyBox, Git, Inkscape, JQuery, Samba ou Wine) sous la direction de Bradley Kuhn. Dans la même orientation, l'Apache Software Foundation héberge depuis 2002 au sein de l'Apache Incubator un nombre croissant de projets qui lui sont confiés pour bénéficier de ses processus de développement et de gestion (de projet et de licences) nécessaires à leur développement ouvert et collaboratif. De manière plus informelle, la Mozilla Foundation a lancé en 2011 l'initiative Web FWD (pour Web Forward) afin d'aider les projets orientés en faveur de technologies ouvertes du Web et la fondation Linux ouvre progressivement des sous-groupes industriels. Plus loin et conséquemment à l'introduction progressive des acteurs traditionnels, un nombre croissant d'organisations industrielles a vu le jour afin de mutualiser différentes missions avec des partenaires (clients ou fournisseurs) et concurrents au sein de fondations, associations, consortium ou alliances : tels l'association ObjectWeb, l'Eclipse Foundation, l'Open Handset Alliance, L'Alliance GENIVI, la Foundation LiMo.

La Fondation Eclipse. Initialement consortium formé par IBM lors de l'ouverture de la plate-forme Eclipse en *open source*, la structure est devenue Foundation Eclipse en février 2004 (avec, comme membres fondateurs des sociétés telles que Ericsson, HP, IBM, Intel, MontaVista Software, QNX, SAP et Serena Software). La fondation supervise le déve-

loppement de l'IDE *open source* Eclipse et aide à cultiver une communauté *open source* et un écosystème de produits et de services complémentaires autour d'Eclipse. Aujourd'hui, presque deux cents membres financent la fondation par leurs cotisations, ce qui lui confère toute son autonomie même si l'implication d'IBM et des autres membres stratégiques reste très forte. Chacun des membres stratégiques possède un représentant au conseil d'administration de la fondation Eclipse, ce qui leur permet d'avoir une influence directe sur la direction stratégique de la fondation. Les membres stratégiques possèdent aussi un siège dans les différents conseils de la fondation, fournissant ainsi leur avis sur les thèmes et les priorités de l'écosystème Eclipse. La fondation met à disposition de ses membres un certain nombre de documents juridiques (statuts, contrat d'adhésion, politique en matière de PI et de respect des dispositions antitrust) et techniques (processus de développement essentiellement) ainsi que la licence Eclipse déjà évoquée.

L'Alliance GENIVI. Il s'agit d'une alliance industrielle à but non lucratif et ayant pour objet de promouvoir l'adoption générale d'une plateforme *In-vehicle infotainment* (IVI, infodivertissement pour véhicule) de référence basée sur des solutions libres, avec un développement *open source*. Elle a été fondée le 2 mars 2009 par le Groupe BMW, Delphi, GM, Intel, Magneti-Marelli, PSA Peugeot Citroen, Visteon, et Wind River Systems et compte aujourd'hui plus d'une centaine de membres (des constructeurs automobiles, des fabricants de semi-conducteurs, des fournisseurs d'équipements et des fournisseurs de services et de logiciels). L'alliance a lancé en août 2011 son programme de certification qui permet aux membres de l'alliance de valider la conformité des plates-formes logicielles présentées à une version des spécifications d'architecture. Le 19 septembre 2012, GENIVI a ouvert officiellement son propre projet (relatif aux logiciels automobiles spécifiques développés par l'alliance) *open source* hébergé par la Fondation Linux.

ObjectWeb. ObjectWeb est une communauté créée fin 1999 par Bull, France Télécom R&D et l'INRIA, autour du développement mutualisé de *middleware* (intergiciel) libres et *open source* et qui a connu un certain nombre de phases dans son développement : transformation en un consortium international en 2002, fusion avec Orientware en 2007 (ce qui donna l'actuel OW2) ainsi qu'avec l'Open Solutions Alliance en 2009.

L'association Tizen. Anciennement LiMo Foundation, elle fut créée en janvier 2007 et est composée de plus de 50 membres et organisations. La gouvernance du projet est « ouverture et transparence », le leadership et la prise de décision sont entièrement partagés. Les membres fondateurs sont Motorola, NEC, NTT DoCoMo, Panasonic Mobile Communications, Samsung Electronics, et Vodafone. Abandonnant son propre projet Meego (provenant de la jonction de Maemo de Nokia et Moblin d'Intel), Intel a rejoint la structure en 2011. Devenu Tizen Association, le projet est maintenant hébergé par la Fondation Linux. Le fonctionnement de ce projet est relativement intéressant puisqu'il repose à la fois sur la Fondation Linux qui assure la gestion du projet et sur une association dédiée au sein de laquelle les industriels se réunissent pour mutualiser leurs besoins, identifier les modèles économiques et travailler sur les services annexes (marketing et éducation notamment). Le choix de la Flora licence sur certains composants de la plate-forme est néanmoins critiqué au regard des limitations qu'apporte cette licence (conditionnant le bénéfice de certaine clause, notamment la licence en matière de brevets, au respect de certaine procédure ou test spécifique à la plate-forme).

L'Open Handset Alliance. Pour favoriser le développement du projet Android (système d'exploitation pour mobile et *framework* pour développer de nouveaux logiciels) diffusé sous licence Apache v2, la société Google a formé l'Open Handset Alliance¹ qui réunit les sociétés désireuses de participer à ce courant d'*open innovation* dans le secteur du mobile (elle est composée de 79 membres dont des opérateurs mobiles, des constructeurs de téléphones, de microprocesseurs, des éditeurs de logiciel et des distributeurs).

Conclusion

Le monde d'aujourd'hui n'est plus celui des années 1980. Il est évident que le logiciel libre est infiniment plus mature aujourd'hui qu'il ne l'était à ses débuts et que l'évolution des différents projets de logiciels libres a entraîné le perfectionnement de leur gouvernance. Les licences

1. L'Open Handset Alliance (OHA) est une alliance commerciale de 79 sociétés réunies pour développer des standards ouverts pour les terminaux mobiles. Voir <http://www.openhandsetalliance.com>.

libres eurent la double tâche de formaliser en interne les règles de contribution et d'agir en externe pour prévenir les risques identifiés au travers l'évolution des lois ou des pratiques (brevets, DRM, etc.).

L'apogée des licences libres, une ère révolue ? La multiplication de licences a permis de pérenniser et de renforcer ce courant en favorisant son adoption par des acteurs hétérogènes. En contrepartie, elle a engendré un certain nombre de problèmes (dont l'introduction de la notion d'incompatibilité et, plus généralement, une complexification du système) qui gênent la diffusion du mouvement, mais qui semblent néanmoins en passe d'être résolus par la mise à jour progressive des différentes licences (et l'arrivée de nouveaux schémas de licences telle que l'initiative *GPL.Next* de Richard Fontana), et par l'interconnexion croissante entre les différents projets (la circulation des contributions et des personnes favorisant une harmonisation des licences). Aujourd'hui, les licences suivent de plus en plus un processus d'édiction communautaire et leur nombre diminue, de sorte qu'il semble possible d'espérer que les préoccupations futures concernent d'avantages l'encadrement des communautés que le seul contrat assurant la gestion des droits. Elles resteraient ainsi tout aussi essentielles, mais en devenant mieux comprises, mieux admises et plus sécurisées. Le risque afférent s'en trouvera diminué, le sujet deviendra moins central et les obstacles à l'adoption moins importants (ce qui est d'autant plus dommage que les juristes commencent seulement à s'y intéresser). Au contraire, du côté des pratiques, des outils et de la gestion des communautés, les prochaines décennies seront certainement riches et cruciales. Relativement récents (tous deux lancés en 2010), les projets *Harmony Agreements* (qui permet la création de contrats de cession de contributions génériques) et le projet *Software Package Data Exchange (SPDX)*, sont de nouvelles réponses susceptibles de favoriser la diffusion du logiciel libre.

Une vigilance toujours nécessaire : les risques emportés par l'industrialisation et la nécessité de définition plus large. Grâce au Libre (et aux multiples courants associés), le logiciel devient ainsi plus proche d'une connaissance qu'on partage que d'un produit que l'on vend. Cela favorise un rapport d'égal à égal entre les individus, les communautés, les entreprises et les institutions publiques. Le prochain défi auquel devra peut être faire face le Libre concerne d'une part les pratiques indus-

rielles, de plus en plus fines, qui se développent autour d'une industrialisation de projets majeurs diffusés sous licences *open source* et, d'autre part, des évolutions techniques qui tendent à rendre le logiciel *in fine* beaucoup moins visible pour l'utilisateur final. Concernant l'industrialisation du Libre/de l'*open source*, il convient en effet de constater que, si les licences sont généralement tout à fait conformes aux définitions FSD ou OSD, l'utilisateur se retrouve de moins en moins bénéficiaire des libertés dont il devrait disposer. Il conviendrait même de se demander si la structuration de l'écosystème par la réunion d'entreprises autour de projets majeurs peut aboutir à des situations d'ententes mais aussi à des abus de position dominante (les deux étant condamnables sur le plan du droit de la concurrence, mais il nous intéresse essentiellement de percevoir l'impact sur les utilisateurs et sur les logiciels eux-mêmes).

L'exemple le plus symptomatique concerne sans aucun doute l'écosystème construit autour du projet Android qui réunit près d'une centaine de sociétés au sein de l'Open Handset Alliance et confère une place prééminente à Google en tant que titulaire des marques, mais surtout en qualité d'éditeur du logiciel et de fournisseurs des services (dont l'accès au Market ou encore un support de niveau très élevé au bénéfice des fabricants). Ainsi, l'éditeur a pu faire renoncer (au motif d'une compatibilité dont il définit les termes) aux libertés assurées par les licences du logiciel à un certain nombre de grands constructeurs, tel Acer qui dû renoncer en 2012 à la distribution en Asie de terminaux fonctionnant grâce à Aliyun OS (*fork* chinois d'Android développé par AliCloud, une filiale du moteur de recherche Alibaba) ou encore Motorola et Samsung qui souhaitaient en 2010 intégrer le service de localisation de la société Skyhook à la place de celui offert par Google. Les libertés deviennent dès lors relatives, organisant un statut *open until it matters*¹ bien loin des objectifs des millions de contributeurs adoptant sciemment des licences comme la GNU GPL. De telles situations réduisent le bénéfice des libertés assurées par les licences et il serait important de conserver un regard critique sur ces comportements.

L'un des chantiers des prochaines années sera ainsi peut-être de repenser les libertés des utilisateurs à une autre échelle que le seul logiciel, premièrement, en étendant ces définitions (l'Open Definition vient par exemple

1. Que l'on pourrait traduire par « ouvert jusqu'à que cela compte (économiquement) ».

généraliser les principes pour les appliquer aux œuvres non logicielles et aux bases de données) et, deuxièmement, en les adaptant au changement de notre société (par exemple, pour ce qui concerne le *cloud*, les libertés des utilisateurs de tels services imposent une réflexion plus large¹). Les libertés d'aujourd'hui ne sont plus nécessairement tout à fait les mêmes que celles d'hier.

Références

- CLÉMENT-FONTAINE, Mélanie, « Les oeuvres libres », Thèse de doctorat, Université de Montpellier, 2006.
- CRAMER, Florian, « The Creative Common Misundestanding », dans : *FLOSS+ Art*, sous la dir. d'Aymeric MANSOUX et marloes DEVALK, OpenMute, 2008.
- JEAN, Benjamin, *Option libre. Du bon usage des licences libres*, Paris : Framasoft/Framabook, 2012.
- LESSIG, Lawrence, *Code and Other Laws of Cyberspace*, New York : Basic Books, 2000.
- « Enormously Important News From the Free Software Foundation », dans : *Blog de L. Lessig* (2008).
 - « How I Lost the Big One », dans : *Stories, Legalaffairs* (2004), URL : http://www.legalaffairs.org/issues/March-April-2004/story_lessig_marapr04.msp.
- STALLMAN, Richard M., « The GNU Operating System and the Free Software Movement », dans : *Open Sources : Voices from the Open Source Revolution*, sous la dir. de Chris DIBONA, Sam OCKMAN et Mark STONE, Sebastopol, CA : O'Reilly, 1999.
- « What is the Free Software Foundation ? », dans : *GNU's Bulletin 1* (1986), p. 8.

1. Sans être mature, le mouvement de l'Open Cloud n'est pas récent. Citons par exemple la fondation OpenWeb qui a publié plusieurs licences qui s'étendent tant au droit des brevets qu'au droit d'auteur ; l'Open Web Final Specification Agreement (OWFa 1.0) utilisé par le projet Open Compute, lancé en 2011 par Facebook ; etc. Il nous semble important d'intégrer à cette définition les problématiques d'accès aux données, de logiciels libres et d'interopérabilité.

WILLIAMS, Sam, Richard STALLMAN et Christophe MASUTTI, *Richard Stallman et la révolution du logiciel libre. Une biographie autorisée*, Paris : Eyrolles, 2010.