

Pour une page web qui dure 10 ans ?

Des pages web légères et moins gourmandes en ressources, du « low-tech » c'est plus écologique probablement, mais c'est aussi une des conditions pour rendre durables des contenus qui ont une fâcheuse tendance à se volatiliser... Jeff Huang est professeur d'informatique et dans la page que Framalang a traduite pour vous, il fait le pari que son contenu sera encore accessible dans dix ans au moins, tout en proposant 7 recommandations pour créer des pages web pérennes.

Il ne s'agit pas de solutions miracles, mais plutôt d'incitations à l'action et au débat, comme il l'écrit :

Cet article est destiné à provoquer et à susciter une action individuelle, et non à proposer une solution complète pour soigner le Web en déclin

Donc il est clair que les conseils qu'il donne sont peut-être incomplets ou critiquables. Certain·e·s (comme un de nos traducteurs) vont par exemple sursauter de lire que les polices Google peuvent être utilisées car elles sont « probablement déjà dans le cache ». D'autres vont regretter que le recours à l'archivage ne soit pas assez mis en avant comme le fait cipherbliss dans cet article... bref, n'hésitez pas à ajouter des critiques constructives.

N'hésitez pas non plus à nous dire s'il existe vraiment des contenus web qui méritent selon vous d'être maintenus dix ans ou plus, et lesquels (créations et expressions personnelles, ressources des Communs, etc.), ou bien si vous acceptez avec fatalisme ou satisfaction que les pages web, comme tout le reste, finissent par disparaître...

Les commentaires, comme toujours sur ce blog, sont ouverts et

modérés.

Page originale : This Page is Designed to Last, A Manifesto for Preserving Content on the Web

Traduction Framalang : goofy, Côme, mo, wisi_eu, retrodev, tykayn

Un manifeste pour la pérennité des contenus sur le Web

Cette page est conçue pour durer

par Jeff Huang



Pour un professeur, la fin de l'année est l'occasion de faire le ménage et de se préparer pour le semestre à venir. Je me suis retrouvé à effacer de vieux marque-pages, eh oui, des marque-pages : cette fonctionnalité des navigateurs autrefois si appréciée qui paraît avoir perdu la bataille contre la « complétion automatique dans la barre d'adresse ». Mais ce geste nostalgique de nettoyage a fini par me déprimer.

Les uns après les autres, les marque-pages m'ont mené vers des liens morts. Disparus, de superbes écrits de kuroShin sur la culture technologique ainsi qu'une série de puzzles mathématiques et les discussions associées des universitaires que mon père m'avait présentés ; disparus aussi les tutoriels d'ingénierie inverse de Woodman qui datent de mes années d'étude, avec lesquels j'ai découvert le sentiment d'avoir le pouvoir sur les logiciels ; même mes plus récents marque-pages ont disparu, une série d'articles exposant sur Google+ la non-

conformité des chargeurs usb-c avec les normes...

Ce n'est pas seulement une question de liens fichus, c'est le problème de la difficulté croissante de maintenir en vie des contenus indépendants sur le Web, conduisant à une dépendance à des plateformes et à des formats de publication qui s'empilent de façon chronologique (blogs, fils, tweets...)

Bien sûr j'ai moi aussi contribué au problème. Un article que j'ai publié il y a sept ans comprend un résumé initial dans lequel un lien vers une démo a été remplacé par une page de spam avec une image de citrouille. C'est en partie à cause de la flemme de devoir renouveler et de maintenir une application web fonctionnelle année après année.

J'ai recommandé à mes étudiants de publier des sites web avec Heroku et des portfolios avec Wix. Mais toute plateforme au contenu irremplaçable meurt un jour. Geocities, LiveJournal, what.cd, maintenant Yahoo Groups. Un jour, Medium, Twitter et même les services d'hébergement comme GitHub Pages seront pillés puis jetés lorsqu'ils ne pourront plus se développer ou n'auront pas trouvé de modèle économique viable.

C'est un problème de nature plurielle.

Tout d'abord, maintenir du contenu demande du travail. Le contenu peut avoir besoin d'être mis à jour pour rester pertinent et devra éventuellement être ré-hébergé. Une grande partie du contenu – ce qui était autrefois la grande majorité du contenu – a été mise en place par des individus. Mais les particuliers (peut-être vous ?) finissent par se désintéresser, si bien qu'un jour, vous ne voudrez peut-être plus vous occuper de la migration d'un site web vers un nouvel hébergeur.

Deuxièmement, le nombre croissant de bibliothèques et de *frameworks* rend le Web plus sophistiqué, mais également plus complexe. Il y a d'abord eu jquery, puis bootstrap, npm, angular, grunt, webpack, et bien d'autres. Si vous êtes un développeur web qui se tient au courant des dernières

nouveautés, alors ce n'est pas un problème.

Mais dans le cas contraire, vous êtes peut-être programmeur de systèmes embarqués, directeur technique d'une start-up, un développeur Java d'entreprise ou un doctorant de chimie. Vous avez sans doute trouvé le moyen de mettre en place un serveur web et sa chaîne d'outils, mais continuerez-vous à le faire d'une année à l'autre, d'une décennie à l'autre ? Probablement pas ! Et lorsque, l'année suivante, vous rencontrerez un problème de dépendance à un paquet ou que vous découvrirez comment régénérer vos fichiers html, vous pourriez abandonner et zipper les fichiers pour vous en occuper « plus tard ».

Même les piles technologiques simples comme les générateurs de sites statiques (par exemple, Jekyll) nécessitent du travail et cesseront de fonctionner à un moment donné. Vous tombez dans l'enfer des dépendances aux paquets NPM, et vous oubliez la commande pour réaliser un déploiement. Et avoir un site web avec plusieurs pages html est complexe ; comment savoir comment chaque page est liée aux autres : « index.html.old », une copie de « about.html » « index.html (1) », « nav.html » ?

Troisièmement, et cela a déjà été rapporté par d'autres (et même réfuté), la disparition du Web « public » au profit du mobile et des applications web, des jardins clos (telles que les pages Facebook), du chargement en temps réel des WebSockets et de l'AMP diminue la proportion même de la toile dans la toile mondiale, qui ressemble désormais davantage à une toile continentale qu'à une « toile mondiale » (*World Wide Web*).

Alors, face à ces problèmes, que pouvons-nous faire ? Ce n'est pas un problème si simple qu'il puisse être résolu dans cet article. La Wayback Machine et archive.org permettent de conserver certains contenus plus longtemps. Et il arrive qu'un individu altruiste relocalise le contenu ailleurs.

Mais la solution se trouve sur plusieurs fronts. Comment créer

un contenu web qui puisse durer et être maintenu pendant au moins dix ans ? Étudiant l'interaction homme-machine, je pense naturellement aux parties prenantes que nous n'aidons pas : actuellement, la mise en ligne de contenu web est optimisée soit pour le développeur web professionnel (qui utilise les derniers frameworks et flux de travail), soit pour l'utilisateur non averti (qui utilise une plateforme).

Mais je pense que nous devrions considérer à la fois 1) le « responsable » occasionnel du contenu web, quelqu'un qui ne reste pas constamment à jour avec les dernières technologies web, ce qui signifie que le site web doit avoir de faibles besoins de maintenance ; 2) et les robots d'indexation qui préservent le contenu et les archiveurs personnels, « archiveur » implique que le site web doit être facile à sauvegarder et à interpréter.

Ma proposition consiste donc en sept lignes directrices non conventionnelles sur la manière dont nous traitons les sites web conçus pour être informatifs, pour les rendre faciles à entretenir et à préserver. Ma suggestion est que le responsable de la maintenance s'efforce de maintenir le site pendant au moins 10 ans, voire 20 ou 30 ans. Il ne s'agit pas nécessairement de points de vue controversés, mais d'aspirations qui ne sont pas courantes – un manifeste pour un site web durable.

1. Revenez à du HTML/CSS « Vanilla » (NdT : le plus simple, sans JavaScript) – Je pense que nous avons atteint le point où le html/css est plus puissant et plus agréable à utiliser que jamais. Au lieu de commencer avec un modèle obèse bourré de fichiers « .js », il est maintenant possible de simplement écrire en HTML, à partir de zéro. Les CSS « Flexbox » et « Grid », le canvas, les Selectors, le box-shadow, l'élément vidéo, le filtre, etc. éliminent une grande partie du besoin de bibliothèques JavaScript. Nous pouvons éviter le jQuery et le Bootstrap, car ils deviennent de toute façon moins pertinents. Plus il y a de bibliothèques intégrées au site

web, plus celui-ci devient fragile. Évitez les polyfills et les préfixes CSS, et n'utilisez que les attributs CSS qui fonctionnent sur tous les navigateurs. Et validez fréquemment votre HTML ; cela pourrait vous éviter un mal de tête à l'avenir lorsque vous rencontrez un bogue.

2. Ne réduisez pas ce HTML. Réduire (compresser) votre HTML et les CSS/JS associés vous semblera peut-être une précieuse économie de bande passante, et toutes les grandes entreprises le font. Pourquoi ne pas le faire ? Eh bien, vous n'économisez pas beaucoup parce que vos pages web doivent être compressées avant d'être envoyées sur le réseau, donc la réduction préventive de votre contenu compte probablement très peu dans l'économie de bande passante, voire pas du tout. Mais même si cela permettait d'économiser quelques octets (ce n'est après tout que du texte), vous devez maintenant avoir un processus de construction et l'ajouter à votre flux de travail, de sorte que la mise à jour d'un site web devient plus complexe. En cas de bogue ou d'incompatibilité future dans le HTML, le format minimisé est plus difficile à déboguer. De plus, il est peu convivial pour vos utilisateurs ; de nombreuses personnes commencent à utiliser le HTML en cliquant sur « Voir la source »¹, et la réduction de votre HTML empêche cet idéal d'apprentissage en regardant ce qui est fait. Réduire le HTML ne préserve pas sa qualité pédagogique, et ce qui est archivé n'est que le tas de code résultant.

3. Préférez maintenir une page plutôt que plusieurs. Plusieurs pages sont difficiles à maintenir. Vous pouvez oublier quelle page renvoient à quoi, et cela nécessite également la mise en place de modèles pour limiter les redondances. Combien de pages une personne peut-elle réellement maintenir ? Avoir un seul fichier, probablement juste un « index.html », est simple et facile à retenir. Profitez de ce défilement vertical infini. Vous n'aurez jamais besoin de fouiller dans vos fichiers ou de faire du grep pour voir où se trouve un certain contenu. Et comment gérer les multiples versions de ce

fichier ? Devriez-vous utiliser git ? Les placer dans un dossier « old/ » ? J'aime l'approche simple qui consiste à nommer les anciens fichiers avec la date à laquelle ils ont été retirés, comme index.20191213.html. L'utilisation du format ISO de la date permet de trier facilement, et il n'y a pas de confusion entre les formats de date américain et européen. Si j'ai plusieurs versions en une journée, j'utiliserais un style similaire à celui qui est habituel dans les fichiers journaux, index.20191213.1.html. Un effet secondaire agréable est que vous pouvez alors accéder à une version plus ancienne du fichier si vous vous souvenez de la date, sans vous connecter à la plateforme d'hébergement web.

4. Mettez fin à toutes les formes de liaison automatique (hotlinking). Ce mot d'avertissement semble avoir disparu du vocabulaire internet, mais c'est l'une des raisons pour lesquelles j'ai vu un site web parfaitement bon s'effondrer sans raison. Cessez d'inclure directement des images provenant d'autres sites web, cessez d'« emprunter » des feuilles de style en vous contentant de créer des liens vers celles-ci, et surtout cessez de créer des liens vers des fichiers JavaScript, même ceux qui sont hébergés par les développeurs d'origine. Les liaisons automatiques sont généralement considérées comme impolies car vos visiteurs utilisent la bande passante de quelqu'un d'autre, elles ralentissent l'expérience utilisateur, permettent un autre site web de pister vos utilisateurs et, pire encore, si l'endroit auquel vous vous connectez modifie la structure de ses dossiers ou se déconnecte tout simplement, la panne se répercute également sur votre site web. Google Analytics est inutile ; stockez vos propres journaux de serveur et configurez GoAccess ou découpez-les comme vous le souhaitez, ce qui vous donnera des statistiques plus détaillées. Ne donnez pas vos journaux à Google gratuitement.

5. N'utilisez que les 13 polices de caractères adaptées au Web – nous nous concentrons d'abord sur le contenu, comprenez :

les caractères décoratifs et inhabituels sont complètement inutiles. Maintenez un petit éventail et les 13 polices de caractères adaptées au Web. Peut-être avez-vous vraiment besoin d'une police de caractères néo-grotesque qui ne soit pas Arial/Helvetica, ou d'une police de caractères géométriques. Dans ce cas, utilisez le minimum nécessaire, comme Roboto (pour le néo-grotesque) et Open Sans (pour le géométrique) ; ce sont les deux polices les plus populaires de Google Fonts, il est donc probable qu'elles soient déjà en cache sur l'ordinateur de vos utilisateurs. Outre ces polices, votre objectif doit être de fournir le contenu à l'utilisateur de manière efficace et de faire en sorte que le choix de la police soit invisible, plutôt que de flatter votre ego en matière de conception. Même si vous utilisez les polices Google, elles n'ont pas besoin d'être liées automatiquement (hotlink). Téléchargez le sous-ensemble dont vous avez besoin et fournissez-les localement à partir de vos propres dossiers.

6. Compressez vos images de manière obsessionnelle. Ce sera plus rapide pour vos utilisateurs, moins gourmand en espace d'archivage et plus facile à maintenir lorsque vous n'avez pas à sauvegarder un énorme dossier. Vos images peuvent avoir la même haute qualité, mais être plus petites. Minimisez vos SVG, compressez sans perte vos PNG, générez des JPEG pour qu'ils correspondent exactement à la largeur de l'image. Cela vaut la peine de passer du temps à trouver la meilleure façon de compresser et de réduire la taille de vos images sans perte de qualité. Et une fois que WebP sera pris en charge par Safari, passez à ce format. Réduisez impitoyablement la taille totale de votre site web et gardez-la aussi petite que possible. Chaque Mo peut coûter cher à quelqu'un ; en effet mon opérateur de téléphonie mobile (Google Fi) facture un centime (de dollar) par Mo de données. Ainsi, un site web de 25 Mo, assez courant de nos jours, coûte lui-même 25 centimes, soit à peu près autant qu'un journal quand j'étais enfant.

7. Éliminez le risque de rupture d'URL. Il existe des services

de contrôle qui vous indiqueront quand votre URL est en panne, ce qui vous évitera de réaliser un jour que votre page d'accueil ne charge plus depuis un mois et que les moteurs de recherche l'ont désindexée. Car 10 ans, c'est plus long que ce que la plupart des disques durs ou des systèmes d'exploitation sont censés durer. Mais pour éliminer le risque de panne totale d'une URL, mettez en place un second service de contrôle. En effet, si le premier s'arrête pour une raison quelconque (passage à un modèle payant, fermeture, oubli de renouvellement, etc.), vous recevrez toujours une notification lorsque votre URL est hors service, puis vous réaliserez que l'autre service de surveillance est hors service parce que vous n'avez pas reçu la deuxième notification. N'oubliez pas que nous essayons de maintenir un service pendant plus de 10 ans (idéalement bien plus longtemps, même 30 ans), donc beaucoup de services vont s'arrêter pendant cette période, donc deux services de surveillance, c'est plus sûr...

Après avoir fait cela, placez un texte dans le pied de page, « La page a été conçue pour durer », avec un lien vers cette page expliquant ce que cela signifie. Ces quelques mots attestent que le responsable fera de son mieux pour suivre les idées de ce manifeste.

Avant que vous ne protestiez, il est évident que cela n'est pas adapté pour les applications web. Si vous créez une application, alors créez votre application web ou mobile avec le flux de travail dont vous avez besoin. Je ne connais pas une seule application web qui ait fonctionné de manière identique pendant 10 ans (sauf le tutoriel python de Philip Guo, en raison de sa stratégie minimaliste de maintenance), donc cela semble être une cause perdue de toute façon. Ce n'est pas non plus adapté pour les sites web maintenus par une organisation comme Wikipédia ou Twitter. Vous faites votre truc, et le salaire d'une équipe informatique est probablement suffisant pour maintenir quelque chose en vie pendant un certain temps.

En fait, il n'est même pas si important de suivre strictement

les 7 « règles », car ce sont plus des incitations que des règles impératives.

Mais admettons qu'une petite partie du Web commence à concevoir des sites web dont le contenu est censé durer. Que se passe-t-il alors ? Eh bien, les gens préféreront peut-être créer des liens vers ces sites car leur accès est garanti à l'avenir. Plus généralement, les gens peuvent être plus soucieux de rendre leurs pages plus permanentes. Et les utilisateurs et utilisatrices ainsi que les robots qui archivent économisent de la bande passante lorsqu'ils visitent et stockent ces pages.

Les effets sont à long terme, mais les réalisations sont progressives et peuvent être mises en œuvre par les propriétaires de sites web sans dépendre de quiconque ni attendre un effet de réseau. Vous pouvez le faire dès maintenant pour votre site web, et ce serait déjà un résultat positif. C'est comme utiliser un sac de courses recyclé au lieu d'un sac en plastique, c'est une petite action individuelle.

Cet article est destiné à provoquer et à susciter une action individuelle, et non à proposer une solution complète au déclin de la Toile. Il s'agit d'un petit pas simple pour un système sociotechnique complexe. J'aimerais donc voir cela se produire. J'ai l'intention de maintenir cette page pendant au moins 10 ans.

Merci à mes étudiants en doctorat Shaun Wallace, Nediya Daskalova, Talie Massachi, Alexandra Papoutsaki, mes collègues James Tompkin, Stephen Bach, mon assistante d'enseignement Kathleen Chai et mon assistant de recherche Yusuf Karim pour leurs commentaires sur les versions précédentes.

Voir les discussions sur Hacker News et [reddit/r/programming](https://www.reddit.com/r/programming)

Cette page est conçue pour durer.



Image réalisée avec <https://framalab.org/gknd-creator/>

La réponse de l'hébergeur à la bergère

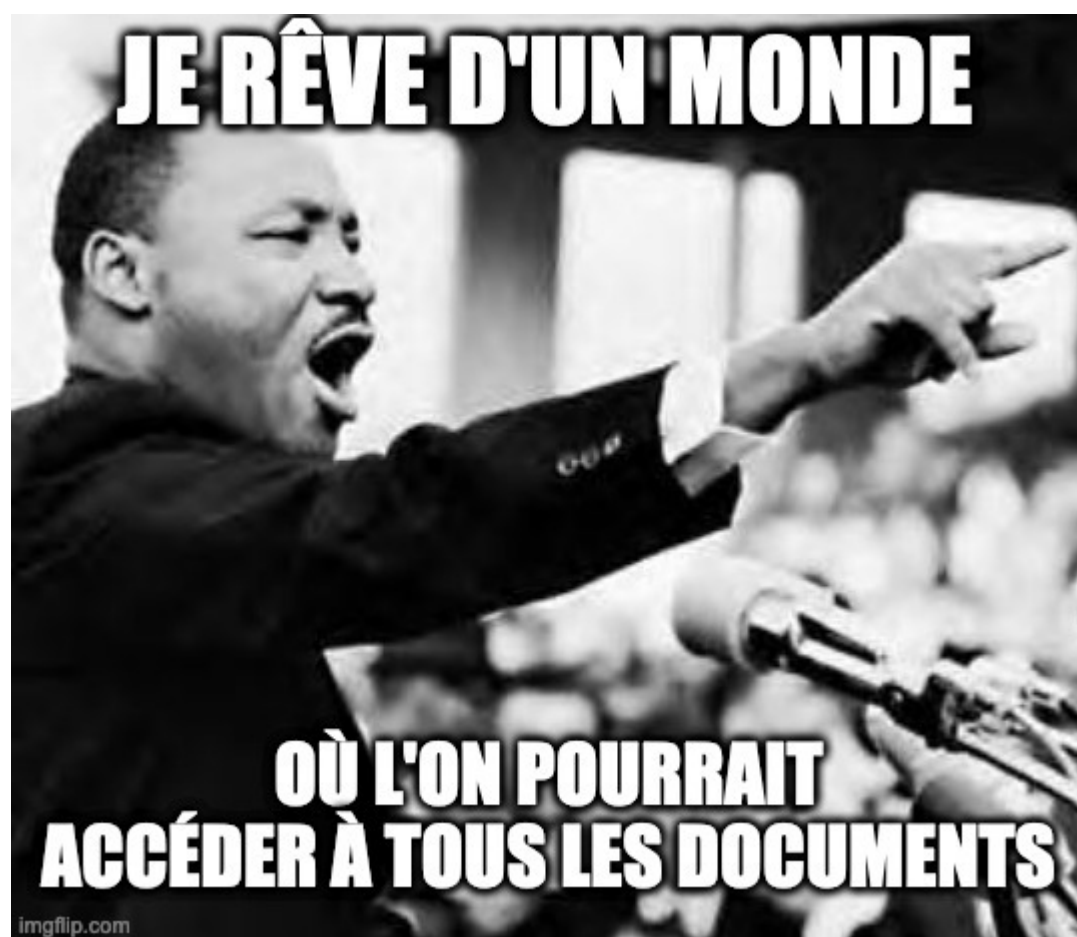
...ou considérations pratiques à l'attention des hébergeurs qui reçoivent une demande de retrait de contenu

L'association Scenari dont je suis membre a reçu un mail de la directrice juridique d'un important éditeur de manuels scolaires (que j'appellerai Éditions X), intitulé « contenus non autorisés ». Ce mail nous informait qu'avaient été découverts « des contenus non autorisés sur votre site et notamment « `<<Relation Client à Distance et Digitalisation >>` » et nous demandait « de supprimer tous les contenus non autorisés par nos maisons d'édition ».

On pourrait s'étonner que certains éditeurs de manuels scolaires jugent opportun en ce moment de chasser les copies illicites sur le Web. On pourrait préférer qu'ils concentrent

l'ensemble de leurs forces pour chercher comment mettre à disposition leurs ressources au plus grand nombre. Mais ce n'est pas le sujet de cet article.

On pourrait aussi avoir envie de rappeler que les contenus pédagogiques devraient être sous licences libres, a fortiori quand ils ont été largement financés par l'argent public. Cela permettrait aux enseignants de se les réapproprier plutôt que de recréer des ressources à côté. Cela permettrait de favoriser des processus contributifs. Cela permettrait leur capacitation numérique, cela améliorerait leur autonomie quand il s'agit de mettre à disposition du contenu en ligne. Mais ce n'est toujours pas le sujet de cet article.



À la réception de cette demande, nous avons réagi *promptement* (on verra que c'est le terme employé dans la loi), mais avec un peu de recul, je me dis que nous avons réagi trop

promptement. Le sujet de cet article est d'étudier comment un petit hébergeur associatif doit réagir en face d'une telle demande.

Rappel du contexte

L'association Scenari est hébergeur de contenus créés avec MyScenari, un logiciel libre combiné à un hébergement offert à ses membres, qui permet notamment de créer des sites et de les mettre en ligne. Pendant la crise Covid-19 l'association a lancé une « Action solidaire Scenari » permettant à tout enseignant non membre de l'association de bénéficier de cet hébergement (scenari.org).

C'était la première fois que nous recevions une demande de retrait de contenus.

Des contenus non autorisés... par quelle autorité ?

Nous avons donc réagi promptement, c'est à dire que nous avons immédiatement répondu au mail reçu que nous allions « éliminer les contenus non autorisés » et nous avons presque aussi rapidement signifié à l'auteur que nous avons reçu cette demande. Celui-ci a retiré ses contenus aussitôt le message reçu, reconnaissant que c'était « *borderline* », mais regrettant que l'éditeur n'ait pas été « un peu plus compréhensif » dans le contexte actuel où il lui faut bien chercher des solutions pour maintenir la fameuse « continuité pédagogique », et s'étonnant que les Éditions X avec qui il est en contact ne l'aient pas interpellé directement.

Donc les contenus ont été éliminés. Grâce à nous.

Grâce à nous un éditeur a pu faire valoir son bon droit. Grâce à nous le travail d'un enseignant et de ses étudiants a été compliqué. Le travail d'un enseignant qui avait fait l'effort de chercher des solutions, par lui-même, d'en trouver, de les

mettre en œuvre. Pas pour spolier des ayants droit, mais pour inventer des solutions à ces problèmes. Il a créé du contenu, nous l'avons détruit.

Pourquoi diable avons-nous fait cela ?

Parce que nous avons réagi, presque mécaniquement, à un argument d'autorité. Un argument d'autorité c'est un argument qui « consiste à faire appel à une autorité plutôt qu'à la raison », nous dit Arthur Schopenhauer dans *L'Art d'avoir toujours raison* (ouvrage que je recommande par ailleurs). Donc, la Directrice juridique (avec une majuscule) des Éditions X nous remercie de. Notez qu'il n'y a pas d'accent à Édition dans le mail reçu, qu'un éditeur devrait pourtant savoir que les majuscules s'accentuent, et qu'en faisant remarquer cela j'use également d'un stratagème rhétorique, l'attaque personnelle (*argumentum ad personam*) qui permet d'attaquer la personne plutôt que le discours. Je me dispense habituellement de le faire lorsque je m'en rends compte, c'est un des avantages de s'intéresser à la rhétorique. Disons que j'ai laissé celui-ci pour illustration de mon propos, et montrer que la rhétorique est une arme à double tranchant.

Soyons procéduriers avec Édith Sillon

Nous vous demandons donc de supprimer les contenus non autorisés, car nous défendons les intérêts des ayants-droits



Pas question
parce que le pluriel
c'est « ayants droit »
donc votre demande
n'est pas recevable



et toc

Argument d'autorité donc. Je vois pourtant au moins trois bonnes raisons de faire fonctionner sa raison, justement, et de ne pas répondre à une telle demande à moins, soit d'y être obligé légalement, soit de s'être construit son propre avis sur la question.

En répondant positivement à la demande sans y être légalement obligé, on fait le choix de léser celui qui est ciblé par la demande. Or nous n'avons pas forcément les éléments pour savoir qui est dans son droit de l'un ou de l'autre. D'une part au sens légal, qu'est-ce qui me prouve que ?... On verra que la loi actuelle va en ce sens et que la preuve est à la charge du demandeur. D'autre part au sens éthique : les ayants droit ont-ils vraiment raison d'interdire coûte que coûte l'accès à leurs contenus, en toutes circonstances ? Aaron Swartz est mort d'avoir refusé une réponse simpliste à cette question. On ne nous en demande pas tant, mais on peut au moins s'arrêter un peu et réfléchir.

En répondant positivement à la demande sans y avoir mûrement réfléchi, nous agissons comme les robots qui suppriment des contenus, parfois de façon tout à fait stupide et injustifiée. Or la majorité des hébergeurs, a fortiori petits, a fortiori libristes, sont contre les systèmes de filtrage automatisés et ont combattu leur systématisation prévue par la proposition de directive européenne sur le droit d'auteur.

En répondant positivement à la demande sans y être obligé, nous consommons de l'énergie qui n'est pas investie ailleurs. Les hébergeurs associatifs, sans orientation commerciale, ont mieux à faire que de s'occuper des intérêts des détenteurs de droits patrimoniaux. Faire tourner les services, les sécuriser, les faire connaître, les documenter, répondre aux utilisateurs, modérer les propos inappropriés, haineux ou discriminatoires... À tout cela on s'est engagé. Mais supprimer des corrigés d'exercices pour un BTS ? Vraiment ? Est-ce que les éditeurs ne peuvent pas se débrouiller pour cela ? (on verra que c'est également à peu près ce que dit la loi, pour

le moment).

La loi qui s'applique actuellement est la loi n° 2004-575 du 21 juin 2004 pour la confiance dans l'économie numérique, dites LCEN.

L'article qui nous intéresse est en particulier l'article 6.

En voici quelques points saillants :

Les hébergeurs sont désignés par la périphrase : « Les personnes dont l'activité est d'offrir un accès à des services de communication au public en ligne », je garderai le terme hébergeur pour mon exégèse (Section I-1).

Les hébergeurs doivent être en mesure de prévenir le « téléchargement et la mise à disposition illicite d'œuvres et d'objets protégés par un droit d'auteur ou un droit voisin » (Section I-1 et référence à l'article L. 336-3 du CPI).

Les hébergeurs ne peuvent pas voir leur responsabilité engagée s'ils « n'avaient pas effectivement connaissance » du caractère illicite des données stockées ou s'ils « ont agi promptement pour retirer ces données ou en rendre l'accès impossible » » (Section I-2 et I.3).

Les hébergeurs ne sont pas obligés de « surveiller les informations qu'[ils] transmettent ou stockent », ni de « rechercher des faits ou des circonstances révélant des activités illicites » » (sauf « surveillance ciblée et temporaire demandée par l'autorité judiciaire ») (Section I-7).

Les hébergeurs ont l'obligation de « mettre en place un dispositif facilement accessible et visible permettant à toute personne de porter à leur connaissance » des données permettant « la répression de l'apologie des crimes contre l'humanité, de la provocation à la commission d'actes de terrorisme et de leur apologie, de l'incitation à la haine

raciale, à la haine à l'égard de personnes à raison de leur sexe, de leur orientation ou identité sexuelle ou de leur handicap ainsi que de la pornographie infantine, de l'incitation à la violence, notamment l'incitation aux violences sexuelles et sexistes, ainsi que des atteintes à la dignité humaine », « des activités illégales de jeux d'argent », des opérations liées au « tabac manufacturé dans le cadre d'une vente à distance » (l'atteinte au droit d'auteur n'est, logiquement, pas mentionné dans cette liste) (Section I-7).

Et la section I.5 nous précise que la connaissance des faits litigieux est présumée acquise lorsqu'il leur est notifié (je reproduis intégralement cette partie) :

- la date de la notification ;
- si le notifiant est une personne physique : ses nom, prénoms, profession, domicile, nationalité, date et lieu de naissance ; si le requérant est une personne morale : sa forme, sa dénomination, son siège social et l'organe qui la représente légalement ;
- les nom et domicile du destinataire ou, s'il s'agit d'une personne morale, sa dénomination et son siège social ;
- la description des faits litigieux et leur localisation précise ;
- les motifs pour lesquels le contenu doit être retiré, comprenant la mention des dispositions légales et des justifications de faits ;
- la copie de la correspondance adressée à l'auteur ou à l'éditeur des informations ou activités litigieuses demandant leur interruption, leur retrait ou leur modification, ou la justification de ce que l'auteur ou l'éditeur n'a pu être contacté.

On comprend donc que celui ou celle qui veut faire retirer du contenu par un hébergeur doit :

- au moins avoir essayé de contacter directement l'auteur

de l'infraction qu'elle pointe, avant de s'adresser à l'hébergeur,

- fournir une motivation qui prouve les faits, ce n'est pas à l'hébergeur de mener l'enquête.

Évolutions attendues à moyen terme (directive européenne)

La directive européenne 2019/790 sur le « droit d'auteur et les droits voisins dans le marché unique numérique » a été adoptée le 17 avril 2019. Il s'agit d'une directive, c'est donc un texte qui ne s'applique pas encore mais qui doit être transposé dans la loi française.

L'article 13 du projet de directive, devenu l'article 17 de la directive adoptée, a été combattu, notamment par les hébergeurs et les défenseurs des libertés individuelles, parce qu'il renverse la charge des ayants droit vers les hébergeurs :

« Si aucune autorisation n'est accordée, les fournisseurs de services de partage de contenus en ligne sont responsables des actes non autorisés de communication au public, y compris la mise à la disposition du public, d'œuvres protégées par le droit d'auteur et d'autres objets protégés, à moins qu'ils ne démontrent que [...] ».

L'hébergeur qui n'aura pas d'accord avec les ayants droit et/ou qui ne sera pas en mesure de filtrer a priori les contenus sera responsable. Ce qui implique la nécessité pour les hébergeurs de passer de tels contrats et de mettre en place des dispositifs automatisés de filtrage.

Je ne m'étends pas sur cette évolution à venir, pour le moment, le régime qui s'applique est celui décrit précédemment.

« La directive passe par deux étapes avant de produire ses

effets : une fois votée par les institutions européennes, elle doit ensuite être transposée par les États membres dans leur droit national, à la différence du règlement, qui s'applique directement. »

Guide dont vous êtes le héros à l'usage des petits hébergeurs

1

Vérifiez que le sujet de la demande n'est relatif qu'au droit d'auteur.

Si on est bien uniquement dans le cas du droit d'auteur, **allez en 2.**

Si on est dans le cas d'un autre signalement portant sur une répression d'intérêt général tel que mentionné par la loi (crimes, haine, terrorisme, discrimination... cf. supra), **allez directement en 6.**

2

Vérifiez que la demande reçue est conforme à la forme prescrite par la LCEN, article 6, section I.5 (cf. supra).

Si oui, il faut la considérer, **allez en 6.**

Sinon, **allez en 3.**

3

La demande reçue n'est pas complète :

S'il y a des menaces associées à une demande incomplète, **allez en 4.**

Si la demande est presque complète, il ne manque qu'une information par exemple, **allez en 5.**

Sinon, **allez en 9.**

4

Vous avez été menacé alors que la demande de signalement n'est pas conforme à la loi :

- signalez à votre tour la menace reçue au procureur de la république avec un mot pour lui expliquer la situation et mettre en évidence votre statut de petit hébergeur (ce sera utile notamment si le demandeur est un habitué des démarches cavalières) ;
- si vous pouvez joindre l'auteur mis en cause, transmettez-lui la demande pour information, informez-le de votre démarche.

5

La demande est presque dans les formes, mais qu'il manque au moins une information :

- accusez réception de la demande en répondant que vous êtes un prestataire technique dont l'activité est d'offrir un accès à des services de communication au public en ligne tel que défini par la loi n° 2004-575 du 21 juin 2004 pour la confiance dans l'économie numérique ;
- n'entreprenez aucune autre démarche (au demandeur de rendre conforme sa demande s'il le souhaite) ;
- si vous pouvez joindre l'auteur mis en cause, transmettez-lui la demande pour information, informez-le de votre démarche.

6

Vous avez reçu une demande conforme à la LCEN, vous êtes tenu d'y répondre, vérifiez les informations transmises.

Si vous êtes convaincu que les informations sont fausses, **allez en 7.**

Si vous avez un doute sur la véracité des informations, **allez en 8.**

Si les informations vous semblent vraies, **allez en 10.**

7

Vous avez reçu une demande conforme à la LCEN, mais vous êtes convaincu qu'elle est abusive :

- accusez réception et informez le demandeur que vous pensez les informations transmises fausses et signalez-lui que l'article 6 de la loi n° 2004-575 du 21 juin 2004 pour la confiance dans l'économie numérique, prévoit que le fait de demander sciemment un retrait sur des bases inexactes est puni d'une peine d'un an d'emprisonnement et de 15 000 Euros d'amende ;
- si vous pouvez joindre l'auteur mis en cause, transmettez-lui la demande pour information, informez-le de votre démarche.

8

Vous avez reçu une demande conforme à la LCEN, mais avez un doute sur la véracité des informations :

- □accusez réception et demandez un complément d'information au demandeur ;□
- si vous pouvez joindre l'auteur mis en cause, transmettez-lui la demande pour information, informez-le de votre démarche, demandez-lui éventuellement son avis ;
- une fois recueillies les informations complémentaires, décidez de vous rendre en 7 ou en 10.

9

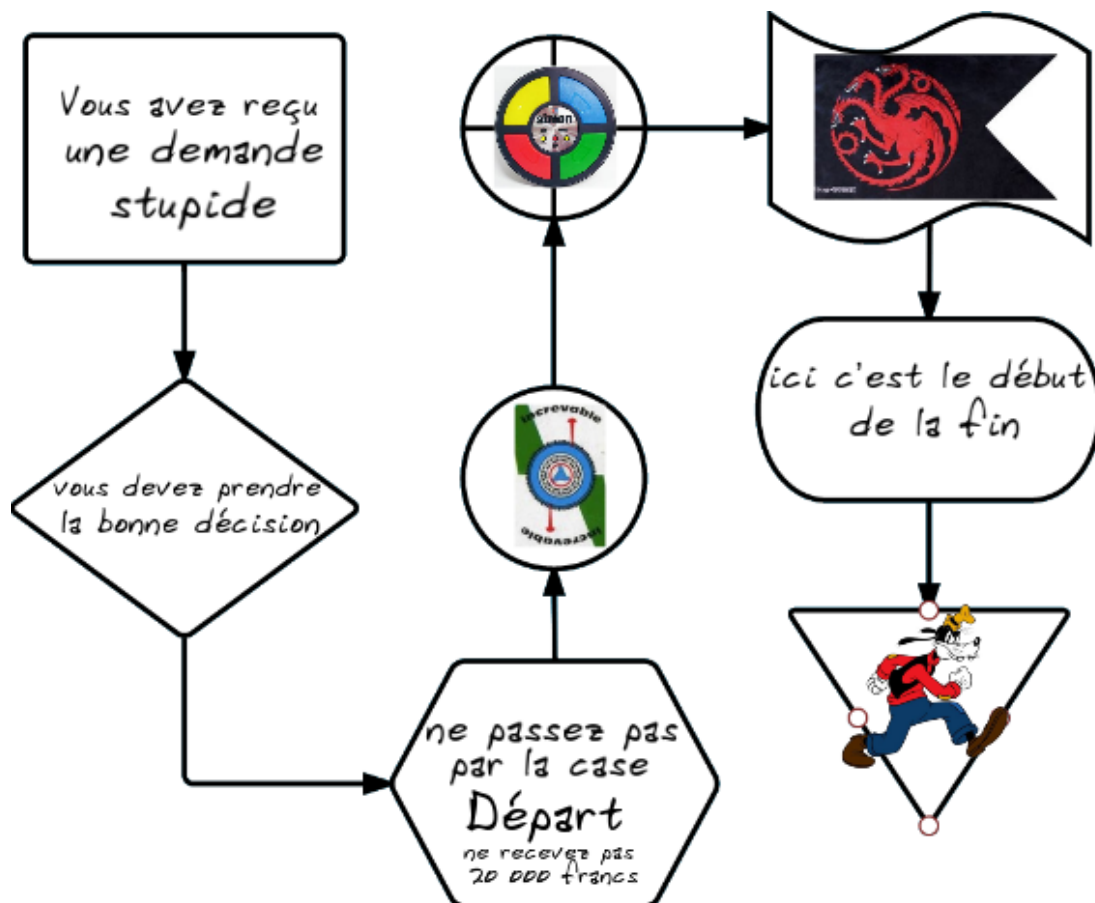
La demande est incomplète, sans menace :

Ignorez la demande et ignorez les relances si elles ne sont pas plus circonstanciées (ou menaçantes).

10

Vous avez reçu une demande conforme à la LCEN et les informations vous semblent vraies :

- procédez au retrait immédiat des données ou rendez-les inaccessibles (en cas de manipulation informatique présentant un risque de perte de données, procéder à une copie préalable) ;
- si vous pouvez joindre l'auteur mis en cause, transmettez-lui la demande et informez-le de votre décision.



Le trajet idéal vous est fourni dans une représentation simplifiée grâce à ce logigramme. On est comme ça chez Framasoft.

La responsabilité de l'hébergeur

Je précise que ce guide est juste un guide, chaque hébergeur est invité à l'adapter selon son éthique et les situations

rencontrées. En particulier, la jurisprudence a déjà considéré que la responsabilité de l'hébergeur pouvait être engagée même si la demande était incomplète dès lors que la description des faits était suffisamment précise pour permettre le retrait. Donc, suivre ce guide comporte une part de risque, notamment si vous ne répondez pas ou peu (points 5 et 9 du guide).

Mais d'un autre côté le législateur a confié de facto à l'hébergeur la responsabilité de garantir l'équilibre entre liberté d'expression d'une part et le préjudice aux tiers d'autre part. Si l'hébergeur ne tient pas son rôle, en arbitrant systématiquement en faveur des retraits, de peur d'un jugement défavorable, y compris lorsque les faits ne sont pas avérés ou que les procédures ne sont pas respectées, alors il œuvre de fait contre la liberté d'expression. Le statut juridique de l'hébergeur ne lui permet pas d'être neutre, il doit prendre ses responsabilités.

Note concernant l'identité de l'éditeur

L'association Scenari a préféré ne pas divulguer le nom de l'éditeur, j'ai respecté ce choix, le propos de l'article étant moins de porter l'attention sur l'attitude de celui-ci que de proposer une réflexion pratique.

Note concernant le mail reçu

Ressentant peut-être une légère honte à faire cette demande en plein confinement, la directrice juridique a assorti sa demande du commentaire suivant : « En effet, nous nous opposons à la mise en ligne de corrigés de nos ouvrages et la majorité des enseignants nous demande de lutter contre ces pratiques qui perturbent leur enseignement. ».

J'ai choisi de ne pas considérer cet angle car :

- Sa demande portait bien sur « tous les contenus non autorisés par nos maisons d'édition » et non pas sur tel ou tel corrigé.

- Je souhaiterais une preuve que la préoccupation de la majorité des enseignants soient en ce moment de lutter contre ces pratiques et pas plutôt de lutter pour trouver des solutions.
- Une recherche web circonstanciée ne faisait pas ressortir ces contenus, donc seuls les étudiants ayant déjà l'adresse fournie par l'enseignant pouvait en pratique accéder au contenu.

Remerciements

Merci aux membres de l'association Scenari, de Framasoft et du CHATONS de m'avoir aidé dans cette recherche, et en particulier à Denis Dordoigne à qui j'ai emprunté une part significative du guide proposé, à Stéphane Poinart de m'avoir pointé la référence du JournalDuNet concernant la jurisprudence, à Christelle pour ses précieux compléments, à Benjamin pour m'avoir relu et rappelé que les hébergeurs avaient leur rôle à jouer, à tous les autres qui ont contribué anonymement.

La réforme européenne du droit d'auteur ? – une menace pour le logiciel libre selon Glyn Moody

Comme vous l'avez peut-être lu dans l'appel de Julia Reda que nous avons publié hier, l'heure est à la mobilisation contre une proposition de directive européenne qui pourrait avoir des

effets dévastateurs.

Non, les GAFAM ne seraient pas les premiers impactés, mais plutôt des entreprises moins bien armées et aussi des sites comme Wikipédia, ainsi que des plateformes de dépôt et partage de code qui constituent des ressources précieuses pour la communauté du logiciel libre.

De tels sites risquent d'être contraints à des dispositifs coûteux et difficiles à mettre en œuvre pour filtrer les contenus sous droits. C'est ce que détaille aujourd'hui Glyn Moody à propos des effets de l'article 13 de cette proposition de directive, contre laquelle a déjà alerté l'April depuis septembre dernier.

Aujourd'hui, la mobilisation de plus de 80 organisations et la mise en place du site <https://savethememe.net/en> constituent des formes d'action militante auxquelles le plus grand nombre doit contribuer. Diffusons largement la traduction des articles de Julia Reda et de Glyn Moody. Signons la lettre ouverte *Save Code Share!* Opposons le groupe de pression de la communauté du libre au lobby du droit d'auteur qui est sans cesse à la manœuvre dans les institutions européennes.

*Ce texte est une traduction d'un article rédigé par le journaliste **Glyn Moody** et publié sur le site Linux Journal le 3 avril 2018. Il a également été publié sur le site de l'April. Nous souhaitons contribuer à lui donner davantage encore de visibilité.*

Traduction : etienne, goofy, mo, Lumi, wyatt, Alby, glyn moody, Fred, April.

Le logiciel libre subit l'offensive des nouvelles lois européennes sur le droit d'auteur



Glyn Moody, photo
Zaizi Ltd (CC BY-SA
2.0)

Le logiciel libre et le droit d'auteur sont étroitement liés. C'est grâce au détournement (hack) habile de la loi sur le droit d'auteur par Richard Stallman qu'a pu être créée la General Public License (GPL) et, par conséquent, le logiciel libre. La GPL requiert de la part des personnes copiant ou modifiant un logiciel publié sous cette licence qu'elles préservent les quatre libertés. Si cela n'est pas le cas, elles enfreignent alors les clauses de la GPL et perdent ainsi toute protection juridique de leurs copies et modifications. En d'autres termes, la sévérité des sanctions pour une violation du droit d'auteur est ce qui permet d'assurer la liberté de partage.

Malgré l'utilisation du droit d'auteur pour faire respecter la GPL et toutes les autres licences du logiciel libre ou *open source*, le droit d'auteur n'est généralement pas si inoffensif. Ce n'est pas étonnant : le droit d'auteur est un monopole intellectuel. En règle générale, il cherche à

empêcher le partage – pas à le promouvoir. Ainsi, les ambitions de l'industrie du droit d'auteur vont généralement à l'encontre des aspirations du monde du logiciel libre.

C'est en Europe que l'on retrouve l'une des preuves les plus évidentes du désintérêt du monde du droit d'auteur envers les préoccupations de la communauté du logiciel libre. Les propositions actuelles de réforme du droit d'auteur au niveau de l'Union européenne contiennent un élément qui aurait des effets dévastateurs pour le codage du logiciel libre. L'article 13 de la pompeusement titrée « Directive du Parlement européen et du Conseil sur le droit d'auteur dans le marché unique numérique » contient la disposition clef suivante :

Les prestataires de services de la société de l'information qui stockent un grand nombre d'œuvres ou d'autres objets protégés chargés par leurs utilisateurs et qui donnent accès à ces œuvres et autres objets prennent, en coopération avec les titulaires de droits, des mesures destinées à assurer le bon fonctionnement des accords conclus avec les titulaires de droits en ce qui concerne l'utilisation de leurs œuvres ou autres objets protégés ou destinées à empêcher la mise à disposition, par leurs services, d'œuvres ou d'autres objets protégés identifiés par les titulaires de droits en coopération avec les prestataires de services. Ces mesures, telles que le recours à des techniques efficaces de reconnaissance des contenus, doivent être appropriées et proportionnées.

Cela signifie concrètement que les sites détenant une (très) importante base de fichiers téléversés par les utilisateurs et utilisatrices seront forcés de filtrer tous les fichiers avant de les publier. Les problèmes posés par cette proposition sont clairs. Une surveillance constante de l'activité des internautes sur lesdits sites, avec tout ce que cela induit en termes de perte de vie privée. Les faux-positifs sont

inévitables, particulièrement parce que les complexités du droit d'auteur ne peuvent pas se voir réduites à de simples algorithmes qui pourront être appliqués automatiquement. Ajouté à l'effet dissuasif que cela aura sur la volonté des internautes de publier des contenus, cela impactera négativement la liberté d'expression et affaiblira le domaine public (article en anglais).

Le coût élevé de la mise en place des filtres de contenus – le système ContentID de Google a nécessité 50 000 heures de codage (page en anglais) et coûté 60 millions de dollars – signifie qu'un nombre restreint de sociétés finiront par contrôler le marché des systèmes de censure. Leur pouvoir d'oligopole leur donne la possibilité de faire payer très cher leurs services, ce qui imposerait de lourdes charges aux entreprises de l'Union européenne et conduirait à une diminution des *startups* dans la région. Un autre problème, parmi d'autres, avec cette idée : le fait non-négligeable que cela pourrait être contraire au droit de l'UE en vigueur (en anglais).

L'article 13 a été spécifiquement rédigé pour satisfaire le désir à peine déguisé de l'industrie européenne du droit d'auteur d'attaquer des entreprises états-uniennes prospères comme Google ou Facebook. Mais le filtrage des contenus mis en ligne est une arme grossière et va en affecter beaucoup d'autres qui, ironiquement, vont être moins capables que les géants d'Internet de se conformer aux onéreuses exigences de la censure. Par exemple, il est fort probable que Wikipédia rentrera dans le périmètre de la nouvelle règle. Après tout, le projet héberge un grand nombre d' « objets protégés » chargés par des utilisateurs et utilisatrices. Comme le montre un billet sur le blog de Wikimedia (en anglais), « il serait absurde de demander à la Fondation Wikimedia de mettre en place des systèmes automatisés coûteux et technologiquement défectueux pour détecter des violations de droit d'auteur. »

Le point de l'article 13 qui est peut être le plus inquiétant

pour les lecteurs du *Linux Journal* concerne les conséquences pour le logiciel libre. Une autre catégorie de sites web qui donnent accès « à des œuvres ou d'autres objets protégés chargés par leurs utilisateurs » : les plateformes de développement collaboratif et les dépôts de code [informatique]. Comme l'explique le site Savecodeshare.eu (Note de traduction : en anglais, voir l'article de l'April qui soutient cette campagne), créé par la Free Software Foundation Europe et OpenForum Europe (note de transparence : je suis un membre bénévole de OpenForum Academy) :

Si cette réforme du droit d'auteur devait être votée, chaque utilisateur ou utilisatrice d'une plateforme de partage de code, qu'il ou elle soit une personne physique, une entreprise ou une administration publique, serait traité comme un potentiel contrevenant au droit d'auteur : tous ses contenus, incluant des dépôts entiers de code, seraient contrôlés et empêchés d'être partagés en ligne à n'importe quel moment. Cela restreindrait la liberté des développeurs et développeuses d'utiliser des composants et outils logiciels spécifiques, ce qui, en retour, conduirait à moins de compétition et d'innovation. Finalement, cela pourrait conduire à des logiciels moins fiables et à une infrastructure logicielle moins résiliente pour tout le monde.

Comme l'explique en détail un livre blanc (PDF en anglais) de la même organisation, des sites web majeurs tels que Software Heritage, GitHub, GitLab, GNU Savannah et SourceForge sont menacés. Il met en exergue le fait que si cette proposition de loi venait à être adoptée, ces sites seraient directement responsables d'un grand nombre d'actions de leurs utilisateurs et utilisatrices, ce qui inclut le téléversement de copies non autorisées de logiciels et l'utilisation de code contraire à la licence initiale. Du fait de la difficulté, voire de l'impossibilité pour la plupart de ces sites d'éviter que cela

se produise, il est fort probable que certains d'entre eux bloquent l'accès à leur site aux utilisateurs et utilisatrices européens et arrêtent leur activité au sein de l'UE. Les projets de logiciels libres pourraient alors être forcés de faire la même chose. Dans le livre blanc, Thomas Pfeiffer, membre du conseil d'administration de KDE, est cité ainsi :

Pour savoir à quel point KDE est directement touché par le règlement proposé, il faut se demander si la configuration ci-dessus ferait de nous des « fournisseurs de services de la société de l'information stockant et distribuant de grandes quantités d'œuvres ou d'autres objets téléchargés par leurs utilisateurs ». Si c'est le cas, nous devrions probablement déplacer la majeure partie de notre infrastructure et de notre organisation en dehors de l'UE.



Figure 1 : Les sociétés et services concernés par l'Article 13 (Image fournie par EDiMA : <http://edima-eu.org/wp-content/uploads/2018/01/Services-affect>

ed-by-Article-13-Infographic.jpg)

Le potentiel impact de l'Article 13 de la directive sur le droit d'auteur sur la façon dont le logiciel libre est créé tout autour du globe est clairement considérable. La bonne nouvelle est que cette loi européenne n'a pas encore été finalisée et est toujours amenée à évoluer ; la mauvaise est qu'elle n'évolue actuellement pas dans le bon sens.

Par exemple, le responsable politique supervisant l'adoption de cette nouvelle loi par le Parlement européen a récemment proposé un amendement qui exonérerait les grands sites de filtrer les contenus mis en ligne à condition que ceux-ci signent des accords avec l'ensemble des titulaires de droits d'auteur sur les contenus qu'ils hébergent. Ce qui est impossible pour les dépôts de code [informatique], étant donné le volume de fichiers téléversés – il n'existe pas de sociétés de perception pour les développeurs et développeuses, comme on peut en trouver chez les musicien·nes ou auteur·es, pouvant par exemple accorder une licence globale sur l'ensemble du contenu hébergé. L'article 13, qu'il s'agisse de filtre automatisé ou d'accord de licence obligatoire, est inconciliable avec la manière de fonctionner des principaux sites dans le domaine du logiciel libre.

Ses défauts fondamentaux impliquent que l'article 13 doit être retiré de la directive sur le droit d'auteur. Un site multilingue baptisé Savethememe.net (en anglais) a été mis en place pour faciliter la prise de contact direct avec les membres du Parlement européen, qui auront un vote décisif sur les propositions. Un autre moyen d'action est de faire connaître la nocivité de l'article 13 pour l'ensemble de l'écosystème du logiciel libre, et sur les effets secondaires négatifs que cela aurait pour l'innovation en général et pour la société.

Plus les programmeurs et programmeuses sont conscient·es du

problème et le font savoir partout, plus les dépôts de code les plus affectés se joignent à la vague globale d'inquiétude grandissante sur les conséquences des filtres de contenus, plus grand sera l'impact de leur appel à complètement abandonner l'article 13.

À propos de l'auteur

Glyn Moody écrit au sujet d'Internet depuis 1994, et au sujet des logiciels libres depuis 1995. En 1997, il a écrit le premier article grand public sur GNU/Linux et le logiciel libre, paru dans *Wired* (en anglais). En 2001, son livre *Rebel Code: Linux And The Open Source Revolution* a été publié. Depuis, il a très souvent écrit sur le logiciel libre et les libertés informatiques. Il a un blog et il est actif sur des réseaux sociaux : @glynmoody sur Twitter ou identi.ca, et +glynmoody sur Google+.