

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

QUELLE STRUCTURE D'INCITATIFS POUR LES LOGICIELS LIBRES ?
UNE ANALYSE ÉCONOMIQUE DU MODÈLE DE DÉVELOPPEMENT
PARTICULIER DES LOGICIELS LIBRES

MÉMOIRE PRÉSENTÉ
COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN ÉCONOMIQUE

PAR
LOUIS-PHILIPPE VÉRONNEAU

JANVIER 2021

REMERCIEMENTS

Je souhaite remercier messieurs Steve Ambler et Louis Martin - respectivement professeur d'économie et professeur d'informatique à l'UQAM - pour avoir gracieusement accepté de diriger ce mémoire de maîtrise. Sans leurs judicieux conseils, je n'aurais pas été en mesure de terminer ce projet.

Merci à Delphine Labrecque-Synnott pour son amour et son support constant.

Merci à finalement à 陳昌倬 dont l'excellent thé vert m'a apporté force, concentration et vitalité tout au long de cet exercice ardu.

DÉDICACE

Ce mémoire est dédié à Debian, dont la communauté vibrante en fait une distribution formidable, tant au point de vue technique qu'au point de vue humain. Preuve indéniable qu'une communauté saine peut s'autogérer, croître et perdurer, Debian aura toujours une place particulière dans mon cœur.

TABLE DES MATIÈRES

Remerciements	ii
Dédicace	iii
Liste des tableaux	vi
Liste des abréviations, sigles et acronymes	vii
Abrégé	viii
Introduction	1
1 Qu'est-ce qu'un logiciel libre ?	3
1.1 Mouvement hackers	4
1.2 Institutionnalisation	6
1.3 Révolution <i>Open Source</i>	7
1.4 Massification des logiciels libres	8
2 Logiciels libres, un problème d'incitatifs économiques ?	11
2.1 « Tragédie des communs » et protection des communs numériques	11
2.2 Incitatifs économiques et logiciels libres	16
2.2.1 Lerner et Tirole : une affaire de réputation	17
2.2.2 Wu, Gerlach & Young : incitatifs et économie comportementale	18
2.2.3 Benkler : de l'importance des incitatifs sociaux	19
2.2.4 Rossi & Coleman : hétérogénéité des pratiques	20
3 Analyse empirique des incitatifs économiques liés aux logiciels libres	22
3.1 Description de la base de données	23
3.2 Méthodologie	26
3.2.1 Statistiques descriptives	31
3.2.2 Le salaire	31
3.2.3 Le plaisir d'écrire du code informatique	34
3.2.4 Le sentiment d'appartenance	36
3.2.5 L'altruisme	36
3.2.6 La satisfaction au travail	37
3.2.7 Choix des unités de mesure	37
3.2.8 Tests statistiques	38
3.3 Analyse des résultats	40

Conclusion	44
Annexe A : Quels modèles d'affaires pour financer les logiciels libres ?	46
Financement académique	46
Financement gouvernemental	47
Fondations	49
Socio-financement	49
Licences multiples	50
Vente de support	50
Logiciel en tant que service	51
Travail bénévole	52
Chasseurs de prime	52
Open Core	52
Mutualisation	53
Annexe B : Tableaux statistiques	54
Bibliographie	65

LISTE DES TABLEAUX

3.1	Description des variables retenues dans la base de données	25
3.2	Description des variables modifiées dans la nouvelle base de données . .	27
3.3	Statistiques descriptives des variables dichotomiques agrégées	54
3.4	Estimation de l'impact de la participation à un projet libre sur le salaire	55
3.5	Estimation de l'impact du degré de participation à la communauté libre sur le salaire	56
3.6	Estimation de l'impact de la participation à un projet libre sur le plaisir d'écrire du code informatique	57
3.7	Estimation de l'impact du degré de participation à la communauté libre sur le plaisir d'écrire du code informatique	58
3.8	Estimation de l'impact de la participation à un projet libre sur le sentiment d'appartenance à la communauté informatique	59
3.9	Estimation de l'impact du degré de participation à la communauté libre sur le sentiment d'appartenance à la communauté informatique	60
3.10	Estimation de l'impact de la participation à un projet libre sur l'altruisme	61
3.11	Estimation de l'impact du degré de participation à la communauté libre sur l'altruisme	62
3.12	Estimation de l'impact de la participation à un projet libre sur la satisfaction au travail	63
3.13	Estimation de l'impact du de degré de participation à la communauté libre sur la satisfaction au travail	64

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ADULLACT : Association des développeurs et utilisateurs de logiciels libres pour les administrations et les collectivités territoriales

BSD : Berkeley Software Distribution

CONTU : Commission on New Technological Uses of Copyrighted Works

FOSS : Free and Open Source Software

FSF : Free Software Foundation

GNU GPL : GNU General Public License

INRIA : Institut national de recherche en informatique et en automatique

MCO : Moindres carrés ordinaire

MIT : Massachusetts Institute of Technology

OSI : Open Source Initiative

SaaS : Software as a Service

UQAM : Université du Québec à Montréal

ABRÉGÉ

Ce mémoire traite des incitatifs économiques qui motivent un individu dans sa décision de publier du code informatique sous une licence de propriété intellectuelle libre. Ce code libre a parfois une grande valeur marchande : pourquoi le rendre accessible gratuitement ? Sujet populaire auprès de certain-e-s économistes dans les années 2000, peu semblent de nos jours s'intéresser à cette question, et ce, malgré la croissance fulgurante de cette branche de l'informatique dans la dernière décennie.

Pour y voir plus clair, nous revisitons certains modèles économiques traitant de ce sujet à l'aide d'une base de données récente et très populeuse, tirée du sondage *Stack Overflow Developers Survey 2018*. Si on constate qu'il est fort probable que l'attrait de la récompense monétaire soit un incitatif majeur à écrire du libre, d'autres incitatifs d'ordre social semblent également avoir un effet significatif.

Mots-clefs : logiciels libres, économie, incitatifs, modèles d'affaires, informatique, communs numériques, open source

INTRODUCTION

L'informatique prend une place exponentielle dans nos vies. Auparavant réservés au travail et à la sphère commerciale, les ordinateurs et autres appareils électroniques sont dorénavant partout, à un point tel que beaucoup ne pourraient plus envisager vivre sans eux.

Face aux multiples défis technologiques auxquels nos sociétés sont confrontées, l'informatique libre est de plus en plus privilégiée comme moyen d'assurer à la fois le respect de la vie privée, la sécurité informatique ainsi que l'excellence technique d'un projet.

Plus qu'un simple mouvement marginal, les logiciels libres sont maintenant adoptés par des entités gouvernementales partout à travers le monde et par des géants de l'informatique tels que Google, Apple, Facebook et Amazon. Chose difficile à envisager il y a quelques années et qui illustre bien ce changement de paradigme, Microsoft est depuis peu l'entreprise contribuant le plus aux logiciels libres dans le monde en terme de lignes de code (Asay, 2018).

Face à cette effervescence, tout porterait à croire que les logiciels libres se portent plutôt bien. Leur popularité n'est cependant pas forcément un signe de santé. En effet, dans les dernières années, des projets extrêmement importants et utilisés par l'écrasante majorité de l'infrastructure du web comme *OpenSSL* ou *GNU Privacy Guard* ont dévoilé des bogues sévères découlant d'un manque de financement (Eghbal, 2016a, p. 13 ; Edge, 2017).

La difficulté de trouver une structure pérenne pour financer adéquatement l'écriture et la maintenance du code libre est bien réelle et semble à première vue alimentée par la structure d'incitatifs économiques particulière aux logiciels libres. En effet, comme le code libre est la plupart du temps accessible gratuitement, les modèles d'affaires traditionnellement utilisés en informatique, comme la vente de licences, ne sont pas applicables.

Quels sont les incitatifs économiques et sociaux qui poussent les développeuses et les développeurs¹ à écrire du code libre ?

Avant de répondre en détails à cette question, nous consacrerons le premier chapitre de ce mémoire à un brève introduction historique et terminologique des logiciels libres. Monde complexe et parfois arcaïe, cette introduction est à notre avis nécessaire pour saisir la nature des débats économiques entourant les modèles d'incitatifs liés aux logiciels libres. Cet exercice nous permettra également de donner un sens aux débats portant sur les aspects éthiques de certains modèles d'affaires qui, encore de nos jours, animent la communauté libre.

Plongeant dans le vif du sujet, le second chapitre de ce mémoire portera sur l'analyse de la structure d'incitatifs économiques particulière aux logiciels libres. Nous tenterons ainsi de comprendre les raisons qui poussent certaines personnes à offrir gratuitement du code ayant une grande valeur économique.

Si plusieurs personnes se sont déjà attardées à l'analyse théorique des incitatifs liés aux logiciels libres, peu d'études chiffrées sur le sujet existent. Le troisième chapitre de ce mémoire sera donc dédié à une analyse empirique des modèles d'incitatifs économiques présentés au premier chapitre. Nous effectuerons cette analyse à l'aide de la base de données provenant du *Stack Overflow Developers Survey 2018*. Ce sondage donne un portrait de l'opinion de près de 100 000 développeuses et développeurs sur différents sujets informatiques en 2018. Nous tenterons ainsi d'établir la validité empirique de ces modèles avec des données récentes.

Finalement, comme complément de recherche, nous présenterons succinctement en annexe A les principaux modèles d'affaires utilisés pour financer les logiciels libres. Le financement de l'informatique libre et les modèles d'affaires qui en découlent ont en effet un impact important sur les possibilités de développer du code libre tout en étant rémunéré. Plus qu'une simple question d'administration des affaires et de plans stratégiques, le choix d'un modèle d'affaires est rendu complexe par l'aspect éthique et politique du mouvement libre.

1. Notre traduction du terme anglais *developers* couramment utilisé dans le milieu informatique pour regrouper les différentes catégories de professionnel-le-s qui travaillent directement avec du code informatique (administration système, programmation, génie logiciel, développement web, administration de réseaux, etc.)

CHAPITRE 1

QU'EST-CE QU'UN LOGICIEL LIBRE ?

S'il existe plusieurs définitions de ce qui constitue un logiciel libre, la définition la plus couramment acceptée est celle établie par Richard Stallman et la *Free Software Foundation* (FSF) (2001) au tournant des années 2000.

Ainsi, un logiciel est considéré libre s'il intègre à sa licence de propriété intellectuelle les quatre libertés suivantes :¹

0. La liberté d'exécuter le programme, pour tous les usages.
1. La liberté d'étudier le fonctionnement du programme et de l'adapter à ses besoins.
2. La liberté de redistribuer des copies du programme (ce qui implique la possibilité aussi bien de donner que de vendre des copies).
3. La liberté d'améliorer le programme et de distribuer ces améliorations au public, pour en faire profiter toute la communauté.

Ces quatre libertés sont encadrées dans des licences de propriété intellectuelle qui décrivent en détail les conditions d'utilisation, de modification et de redistribution des programmes libres. Plusieurs centaines de licences libres existent, mais la grande majorité des programmes utilisent des licences populaires comme la GNU GPL, la licence MIT ou encore la licence BSD.

La définition de la FSF — qui est en pratique l'organisme en charge de décider si une licence est libre ou non — diffère légèrement de celle utilisée par un autre organisme important, l'*Open Source Initiative* (OSI). Ainsi, il est possible qu'une licence soit approuvée par l'OSI (et soit donc dite *open source*), mais ne soit pas libre. En général les licences approuvées par la FSF le sont également par l'OSI.

1. Ces quatre libertés sont numérotées de 0 à 3 pour des raisons historiques ; jusqu'en 1990, il n'y avait que les libertés 1 à 3. Après réflexion, la FSF décide d'ajouter la liberté 0 avant toutes les autres pour marquer le fait que sans elle, les autres libertés ne peuvent s'appliquer.

Si le terme *open source* est plus connu du public, nous faisons plutôt ici le choix d'utiliser le terme « logiciel libre ». Ce terme a en effet une connotation forte et met l'accent sur l'aspect politique et éthique de ce mouvement plutôt que sur les modalités techniques du code informatique produit. Les logiciels libres les plus connus à l'extérieur des communautés informatiques sont le navigateur web *Firefox*, la plateforme de gestion de contenu en ligne *WordPress*, la suite de bureautique *LibreOffice* et le lecteur multimédia *VLC*.

Comme nous le verrons dans le second chapitre, la structure d'incitatifs économiques liés à la production de logiciels libres est complexe et est fortement influencée par l'histoire de l'informatique libre et du militantisme politique qui l'entoure. Avant de procéder à une analyse économique détaillée des particularités des logiciels libres, il est ainsi important d'effectuer un rappel historique des racines de ce mouvement.

1.1 Mouvement hackers

Avec l'émergence de l'informatique comme discipline universitaire dans les années 1960 apparaît une nouvelle génération d'étudiant-e-s à qui on donne accès aux ordinateurs de l'université. Cette génération forme la première vague de *hackers*, une communauté dotée d'un ethos promouvant le partage et la curiosité — parfois au mépris de la loi.

Le cadre juridique flou (voire inexistant) encadrant la propriété intellectuelle du code informatique faisant fonctionner les ordinateurs ainsi que les pratiques du milieu informatique de l'époque rendent possible la croissance de cette communauté qui met de l'avant la coopération comme moyen de production économique.

Les communautés *hackers* sont ainsi portées par un code d'éthique très fort qui leur permet de maintenir une cohésion sociale importante. Steven Levy (1984b, p. 26—36), journaliste ayant documenté les communautés *hackers* des années 70 et 80, résume l'ethos *hacker* en six points :

1. *Access to computers—and anything which might teach you something about the way the world works—should be unlimited and total. Always yield to the Hands-On Imperative!*
2. *All information should be free.*
3. *Mistrust Authority—Promote Decentralization.*
4. *Hackers should be judged by their hacking, not bogus criteria such as degrees, age, race or position.*
5. *You can create art and beauty on a computer.*
6. *Computers can change your life for the better.*

Plus qu'un simple code d'honneur bidon, cet ethos encadre la manière dont les *hackers* existent et respirent. Obsédés par les promesses d'un futur informatisé où les ordinateurs sont accessibles au commun des mortels, ces derniers travaillent jour et nuit sur les rares machines qui leur sont disponibles, allant même jusqu'à remplacer le cycle de vie de 24h typique par un cycle de 30h pour gagner en efficacité. Ces *hackers* — véritables pionniers de l'informatique moderne — ne sont pas motivés par le profit ou la gloire personnelle, mais bien par le pur plaisir d'écrire du code informatique et de réaliser le plein potentiel de leurs terminaux rudimentaires.

Levy décrit ainsi les motivations qui poussent Peter Samson, *hacker* au *MIT Artificial Intelligence lab* dans les années 70, à écrire le tout premier compilateur² de musique :

The question of royalties never came up. To Samson and the others, using the computer was such a joy that they would have paid to do it. The fact that they were getting paid the princely sum of \$1.60 an hour to work on the computer was a bonus. As for royalties, wasn't software more like a gift to the world, something that was reward in itself? The idea was to make a computer more usable, to make it more exciting to users, to make computers so interesting that people would be tempted to play with them, explore them, and eventually hack on them. When you wrote a fine program, you were building a community, not churning out a product.

Anyway, people shouldn't have to pay for software — information should be free! (Ibid., p. 43—44).

2. Un compilateur est un programme informatique qui transforme du code source — destiné à être lu par un être humain — en code binaire, destiné à être interprété par une machine.

1.2 Institutionnalisation

En parallèle avec l'émergence du mouvement *hackers*, la classe politique américaine profite dès 1967 des réflexions entourant la mise à jour *Copyright Act* pour aborder les questions entourant la propriété intellectuelle du code informatique. Après un rendez-vous manqué avec l'histoire en 1976, le Sénat américain adopte en 1980 les recommandations de la *Commission on New Technological Uses of Copyrighted Works* (CONTU) et encastre le code informatique dans le *Copyright Act*.

Une décennie après le début des travaux de la CONTU, les tribunaux américains établissent une jurisprudence forte sur cette question avec la cause *Apple v. Franklin* de 1983, affirmant que toutes les formes de code (code source, code binaire, code physiquement gravé dans des puces en silicone (*Read-only memory*)) sont sujettes au *Copyright Act* américain. Ces changements législatifs ont un impact majeur sur les communautés *hackers*, qui voient leur modèle de production économique gravement attaqué (Véronneau, 2018).

Rare was the program that didn't borrow source code from past programs, and yet, with a single stroke of the president's pen, the U.S. government had given programmers and companies the legal power to forbid such reuse. Copyright also injected a dose of formality into what had otherwise been an informal system. (Williams, 2002, p. 124).

Ces changements législatifs affectent durement les communautés *hackers*, qui voient leur accès au code source des logiciels écrits par les entreprises privées se réduire comme une peau de chagrin. La lente mort du modèle de création de code informatique propre aux années 60 et 70 basé sur le partage et la collaboration divise ces communautés holistiques. Partout à travers les États-Unis, les clubs de *hackers* se fractionnent en factions rivales et se font compétition pour les sommes faramineuses provenant de la montée en popularité des ordinateurs personnels et des consoles de jeux vidéos (Levy, 1984b, p. 395—434).

C'est dans ce contexte maussade que Richard Stallman, *hacker* au *MIT AI lab*, crée en 1985 la toute première licence de propriété intellectuelle libre, la *GNU Emacs General Public Licence*. Véritable perversion du copyright, cette licence est la première itération formelle de ce qui deviendra les 4 libertés fondamentales des logiciels libres. De plus, elle oblige la redistribution sous cette même licence de toutes modifications au programme, créant ainsi un effet viral. Dans les faits, cette licence encastre l'idéal *hacker*, les pratiques informelles de sa communauté et son ethos dans un texte légal protégé par le système juridique américain (Williams, 2002, p. 124—132).

Stallman fonde la FSF la même année, donnant naissance au mouvement des logiciels libres.

1.3 Révolution *Open Source*

Blâmant Richard Stallman et la FSF pour leur approche dogmatique et leur faible ouverture vis-à-vis le monde commercial, Eric S. Raymond, Tim O'Reilly, Bruce Perens et une poignée d'autres figures montantes de l'informatique libre décident de se regrouper en 1998 pour tenter de changer l'image de marque des logiciels libres. Ces derniers désirent conserver le modèle de création particulier des logiciels libres basé sur la collaboration et la décentralisation, tout en laissant de côté l'aspect éthique et politique hérité du mouvement *hacker*. C'est l'acte de fondation de l'OSI (DiBona et al., 2008).

C'est dans cette optique que ce regroupement décide d'utiliser le terme *open source* plutôt que « logiciels libres ». En plus de briser la puissante image de marque établie par la FSF autour des logiciels libres, le terme *open source* a à leur avis l'avantage d'éviter l'ambiguïté apportée par le double sens du terme *free software* en anglais.³ En forgeant le terme *open source*, l'OSI souhaite en définitive créer un terme anhistorique et politiquement neutre :

The real conceptual breakthrough, though, was admitting to ourselves that what we needed to mount was in effect a marketing campaign—and that it would require marketing techniques (spin, image-building, and rebranding) to make it work.

[...]

3. Il n'est pas rare que la différence entre les deux significations de *free software* en anglais soit expliquée en déclarant : “*Think free as in free speech, not free beer*”.

It seemed clear in retrospect that the term ‘free software’ had done our movement tremendous damage over the years. [...] Most of the damage, though, came from something worse—the strong association of the term ‘free software’ with hostility to intellectual property rights, communism, and other ideas hardly likely to endear it to an MIS manager. (Raymond, 1999d, p. 175—176)

Les efforts de l’OSI pour promouvoir les logiciels libres dans les grandes entreprises et dans le monde commercial arrivent à point nommé. En effet, avec la montée de l’informatique dans la sphère commerciale, certains logiciels libres comme le serveur web *Apache* prennent littéralement leur envol. Entre la première version publique d’*Apache* en 1995 et l’incorporation de la *Apache Software Foundation* en 1999, le projet acquiert 61 % des parts du marché des serveurs webs. Le succès instantané de ce logiciel est en grande partie dû à son modèle de développement ouvert et à sa supériorité technique (*Ibid.*, p. 160).

Apache est cependant loin d’être une exception ; les années 2000 sont le cadre d’une véritable “révolution *open source*” avec la montée en force d’entreprises spécialisées dans l’offre commerciale de logiciels libres comme Red Hat, ou encore la consécration de *Perl*, fortement ancré dans l’écosystème libre, comme langage de programmation du web.

1.4 Massification des logiciels libres

Dans un domaine aussi changeant que l’informatique, vingt ans peuvent paraître une éternité. Chose certaine, depuis les années 2000, notre utilisation de l’internet et des ordinateurs a beaucoup changé. Si seulement 41 % des ménages américains avaient accès à l’internet chez eux en 2000, cette proportion s’élevait à 69 % en 2009 et à 82 % en 2016, preuve certaine de la nécessité grandissante d’être « connecté » de nos jours (U.S. Census Bureau, 2010 ; Ryan, 2018, p. 5).

Loin d’être le seul mouvement de fond ayant profondément marqué le mouvement libre ces vingt dernières années, l’augmentation rapide de la popularité des logiciels libres ainsi que de la quantité de code libre publié en ligne — phénomène que nous appellerons massification des logiciels libres — a très certainement eu un impact important sur l’écosystème libre dans son ensemble.

D’aucuns reconnaissent ainsi que, malgré les côtés positifs associés à la popularité croissante des logiciels libres, leur massification s’est faite à un certain prix. En effet, l’intérêt des grandes entreprises pour le libre s’est accompagné d’un phénomène de cooptation contraire à l’ethos si important de cette communauté.

Bradley M. Kuhn, *Policy Fellow* au *Software Freedom Conservancy*, un organisme de promotion des logiciels libres, affirmait ainsi lors d’une conférence d’informatique libre en février 2020 :

Everyday, more FOSS is being written that has ever existed in History. However, I feel that most for-profit companies, from my point of view, are basically writing the wrong free software. [...] They are focused on software that is going to increase their ability to move forward in their businesses. They are focused on software that they can use as a component or as a base system, open-core style, for some larger proprietary system that they are creating. [...] In some weird sense, because of the corporate interest in FOSS, companies have a lot more software freedom than individuals do each day.
(Sandler, Kuhn, 2020)

Un excellent exemple illustrant cette problématique est le système d’exploitation pour téléphones intelligents Android, basé sur le noyau Linux. La popularité d’Android — plus de 2.5 milliard d’appareils étaient en service en 2020 (Amadeo, 2020) — fait ainsi de Linux le système d’exploitation le plus populaire au monde, du moins sur papier.

En effet, si Android lui-même est libre, les téléphones intelligents nécessitent souvent une myriade de logiciels propriétaires pour fonctionner.⁴ Sans des applications propriétaires comme *Google Play*, permettant d’installer d’autres applications, ou encore *Google Maps*, un service de cartes et de positionnement en temps réel, ces téléphones ont en définitive peu de valeur. À quoi bon avoir un téléphone intelligent si on ne peut pas y installer de nouvelles applications ou utiliser le GPS pour se repérer ?

Comme Google avec Android, beaucoup d’entreprises utilisent dorénavant des logiciels libres comme base de leurs systèmes, mais se différencient par des interfaces ou des applications propriétaires. Le résultat est un système hybride qui ne respecte pas les quatre libertés fondamentales des logiciels libres, même si ces systèmes ne pourraient fonctionner sans eux.

4. Cela n’est généralement pas le cas pour les autres distributions Linux.

Loin d'être une panacée, la massification des logiciels libres observée depuis deux décennies n'a donc pas toujours été un phénomène positif, particulièrement au niveau du respect de l'éthos libre en ligne.

CHAPITRE 2

LOGICIELS LIBRES, UN PROBLÈME D'INCITATIFS ÉCONOMIQUES ?

2.1 « Tragédie des communs » et protection des communs numériques

Dans le célèbre papier de Garreth Hardin intitulé *The Tragedy of the Commons* (1968), ce dernier avance que face à un bien public, le gain marginal d'un individu à la surutilisation est positif, les profits étant individuels alors que les coûts sont partagés. Éventuellement, le bien public est détruit à force de le surutiliser.

Pour Hardin, ce qui rend ce cadre conceptuel aussi intéressant est son caractère systémique. Le terme « tragédie » des communs est ainsi utilisé pour illustrer l'aspect inévitable de ce phénomène. À l'instar d'une tragédie grecque, toutes les parties concernées sont conscientes qu'un problème existe, mais ne sont néanmoins pas en mesure de l'empêcher.

Le développement de logiciels libres semble être à première vue un cas classique de tragédie des communs. De par leur nature ouverte, les logiciels libres peuvent être classés comme des biens publics : n'importe qui peut — sans devoir payer un sou — télécharger et utiliser un logiciel libre. Les coûts associés à la conception et la maintenance d'un projet libre sont cependant bien réels ! Plus de 15 600 personnes différentes ont travaillé sur le noyau Linux¹ entre 2005 et 2016 (Corbet, Kroah-Hartman, 2017, p. 3), pour un coût total estimé à plusieurs milliards de dollars (McPherson et al., 2008).

1. Le noyau Linux est le cœur des systèmes Linux et fournit une interface entre les composantes physiques d'un ordinateur et les programmes informatiques qui souhaitent utiliser ces ressources.

Dans un rapport produit pour la *Ford Foundation* sur les communs numériques, Nadia Eghbal (2016a) en arrive à la conclusion que le code libre forme maintenant le cœur de l’infrastructure numérique contemporaine. Ces communs numériques — tout comme les routes et les ponts que nous empruntons quotidiennement — sont des biens publics essentiels sur lesquels nos sociétés modernes sont bâties. Contrairement aux infrastructures routières, en très grande majorité gérées et financées par les gouvernements, ces communs numériques n’ont pas forcément de modèles de financement pérennes. Tragédie des communs oblige, ils sont donc parfois laissés à l’abandon (*Ibid.*, p. 8). Ce manque de ressources financières peut avoir des impacts bien réels : une faille de sécurité dans un programme libre très utilisé peut permettre à certains acteurs malveillants de s’introduire dans des systèmes critiques et causer d’importants dégâts.

Par exemple, en 2014 une faille est trouvée dans le logiciel libre *OpenSSL*, utilisé par l’écrasante majorité des sites en ligne. Surnommée *Heartbleed*, cette faille de sécurité a permis à des acteurs malveillants de voler les clés de chiffrement des sites utilisant *OpenSSL* et d’intercepter les communications en ligne de centaines de milliers de personnes. Au Canada, *Heartbleed* a rendu possible le vol de 900 numéros d’assurance sociale et a forcé l’Agence du revenu du Canada à fermer son site plusieurs jours pendant la période des impôts (CBC News, 2014). Les problèmes d’*OpenSSL* ont vraisemblablement été causés par un manque de ressources financières (Eghbal, 2016a, p. 13).

Dans des cas extrêmes, ces bogues peuvent parfois même coûter des vies. En effet, avec la montée en puissance de l’informatique dans le domaine médical, de plus en plus d’implants, stimulateurs cardiaques et autres appareils médicaux utilisent du code libre. Le domaine aéronautique et l’industrie lourde sont également régulièrement affectés par ce genre de problématique.

Pour tenter de mitiger les problèmes créés par un phénomène de tragédie des communs, les économistes proposent traditionnellement d’établir des droits de propriétés clairs sur la ressource surexploitée (Ostrom, 1990c, p. 12). Les quotas de production de gaz à effet de serre et la bourse du carbone sont par exemple des solutions très en vue pour tenter de contrer les problèmes de réchauffement climatique. Cela n’est cependant pas une solution viable pour les logiciels libres : ceux-ci sont par définition des biens publics et il n’est pas possible d’en restreindre l’accès sans en faire des logiciels propriétaires.²

2. Récemment, certains projets comme *MongoDB* ou encore *Redis* ont modifié leurs licences pour tenter d’ajouter des restrictions additionnelles sur l’utilisation commerciale de leurs projets. Très mal vus par la communauté libre, ces efforts n’ont généralement pas eu l’effet escompté.

Contrairement à d'autres ressources victimes de tragédie des communs, comme les aquifères utilisés pour irriguer les cultures ou encore les pâturages destinés au bétail, les logiciels libres ont l'avantage d'être des ressources virtuelles. Le coût marginal d'utilisation d'une ressource supplémentaire d'un logiciel libre est donc nul. Il ne serait ainsi pas à proprement dit possible de « surutiliser » un logiciel libre. Qui plus est, les coûts de maintenance d'un logiciel sont généralement les mêmes, qu'il soit utilisé par une seule personne ou par dix mille individus. La particularité des projets libres est donc qu'ils font face à des coûts de maintenance fixes et ont des coûts variables soit extrêmement faibles, soit inexistants.

Si un financement adéquat est une condition essentielle à la réussite des logiciels libres, il serait erroné de réduire la tragédie des communs à une simple question d'argent. En effet, comme le fait remarquer Elinor Ostrom dans *Governing the Commons* (1990c) — un ouvrage qui lui a valu en 2009 le prix de la Banque de Suède en sciences économiques en mémoire d'Alfred Nobel — la tragédie des communs est d'abord et avant tout une question d'organisation sociale.

Rejetant le terme « tragédie des communs » pour sa connotation fataliste, Ostrom s'ancre plutôt dans un courant issu de l'économie institutionnelle pour analyser les réussites et les échecs de la gestion collective des communs, démontrant ainsi que la privatisation des communs n'est pas toujours une solution souhaitable. Selon Ostrom, l'organisation collective réussit souvent là où les problématiques de gestion sont abordées de manière holistique (*Ibid.*, p. 61).

Ostrom en arrive ainsi à identifier huit grands principes qui permettent aux collectivités de s'organiser pour gérer adéquatement les communs qui les entourent (*Ibid.*, p. 90) :

1. Il est nécessaire d'établir clairement les limites d'un projet, tant en termes de but final que de membres.
2. La collectivité doit avoir des règles bien adaptées aux besoins et aux conditions locales.
3. Il faut établir une structure de décision collective claire.
4. Il est nécessaire de mettre en place un système de surveillance, préférablement collectif.
5. La collectivité doit avoir un système de sanctions échelonnées et progressives.
6. Les mécanismes de résolution de conflits doivent être fonctionnels.
7. Les communs ne peuvent fonctionner sans la reconnaissance étatique du droit à l'organisation autonome.
8. Une organisation à plusieurs paliers est préférable à une organisation entièrement centralisée.

La gouvernance adéquate d'un projet semble donc être tout aussi importante que le choix d'un modèle d'affaires viable pour créer des écosystèmes libres sains et pérennes.

Malgré la pertinence du sujet, nous n'aborderons ici que le quatrième principe pour des raisons de brièveté. Un travail d'analyse plus complet pourrait en effet constituer à lui seul un mémoire de maîtrise des plus intéressants. Notons cependant qu'à première vue, comme le note avec brio Shauna Gordon-McKeon dans une conférence intitulée *Governing the Software Commons* (2019), les projets libres bien établis semblent mettre en application une majeure partie de ces principes.³

Selon le quatrième principe d'Ostrom, il est nécessaire de mettre en place un système de surveillance pour assurer le respect des règles établies collectivement. Typiquement, ces mécanismes ont comme but d'empêcher la surexploitation du commun pour empêcher sa dilapidation.

3. Par exemple, le projet Debian — véritable pilier de l'informatique libre depuis sa fondation en 1993 — aborde de manière explicite les points 1 à 6 à travers ses documents officiels, comme le *Debian Policy Manual* (2019).

Comme nous l'avons vu précédemment, les projets libres ne sont pas sujets à la surexploitation. Les stratégies de surveillance jouent cependant un rôle très important dans les communautés libres. En effet, de par la nature ouverte et collaborative des logiciels libres, il n'est pas rare que de parfaits inconnus collaborent sur un projet commun. Il est alors important d'avoir des mécanismes institutionnels pour préserver la qualité du code informatique. Ces procédés sont également utiles pour éviter l'ajout de code malveillant par des acteurs hostiles, ou encore de code de mauvaise qualité pouvant entraîner des failles de sécurité et des bogues.

Part entière du processus de maintenance du code informatique, les mécanismes de surveillance des logiciels sont très coûteux, non seulement pour les personnes qui les développent mais également pour les personnes qui les utilisent. Ainsi, il n'est pas possible de savoir si la nouvelle version d'un logiciel est réellement digne de confiance sans soumettre à un audit les modifications au code. S'il est possible de réduire ces coûts par des processus automatisés (intégration continue, tests unitaires, etc.), la force de la communauté libre passe notamment par la collectivisation des procédés de surveillance du code formant les communs libres.

Ainsi, certaines communautés comme le projet Debian ont entre autres été bâties dans le but de mutualiser le travail de surveillance des logiciels libres. Devenir membre à part entière du projet Debian est un processus ardu et nécessite de réussir une série de tests techniques et éthiques pointus. Il n'est pas rare que ce processus prenne plus d'un an à compléter. Les membres du projet Debian jouissent donc d'une excellente réputation et sont généralement considérés comme digne de confiance par la communauté libre.

Plutôt que d'effectuer un audit constant des logiciels qu'elle utilise, une entreprise peut ainsi se fier sur la qualité et la sécurité des logiciels redistribués par Debian. En retour, il est à son avantage de participer à ce processus d'audit collaboratif : en s'assurant elle-même que certains logiciels sont fiables dans le cadre du projet Debian, une entreprise redonne à la communauté et contribue à la pérennité du projet. Plusieurs personnes sont ainsi employées pour travailler sur Debian par des entreprises utilisant cette distribution.

En définitive, la popularité grandissante des logiciels libres depuis les années 90 témoigne de l'efficacité des mécanismes institutionnels mis en place par la communauté libre avec le temps. Consciente de la nécessité d'une saine gestion des communs libres, cette dernière a su contrer la « tragédie » des communs grâce à l'application des huit principes de gestion d'Ostrom. Certes imparfaite, la gestion collective et démocratique des communs libres a été gage de leur succès à grande échelle, et ce, malgré l'influence d'un cadre économique priorisant trop souvent la compétition à la collaboration.

2.2 Incitatifs économiques et logiciels libres

Si dans la section précédente nous avons abordé la pérennité d'un modèle de développement des logiciels libres décentralisé et démocratique, une question reste : pourquoi une quantité non négligeable d'individus travaillent-ils bénévolement sur des projets libres ?

En effet, si certaines personnes sont payées pour développer et maintenir des logiciels libres sur une base contractuelle ou salariée, une majeure partie de l'écosystème libre repose sur du travail bénévole. Un sondage auprès des membres du projet Debian en 2016 (de Blanc et al., 2017) démontre ainsi que seul 20 % des répondantes et des répondants sont payé-e-s pour travailler sur le projet, et qu'en moyenne, seul 36 % de leur travail sur cette distribution est réellement rémunéré. Ces chiffres sont en général assez représentatifs de la situation générale dans la communauté libre.

Et pourtant, comme le mentionne avec tant d'éloquence Eghbal (2016a), l'importance des logiciels libres dans l'économie contemporaine n'est plus à démontrer. On se retrouve donc dans une situation où un nombre très important de travailleuses et de travailleurs qualifié-e-s produisent collectivement un ensemble de communs libres ayant une valeur marchande considérable de manière tout à fait bénévole. À première vue, cela semble contraire à la théorie économique standard et plus précisément aux modèles de maximisation d'utilité par un-e agent-e économique rationnel-le. En effet, ces personnes pourraient très bien décider d'investir leur temps dans un projet informatique rémunéré, le marché du travail en informatique étant notoire pour sa compétitivité.

Force est donc de constater qu'il existe une structure d'incitatifs économiques particulière aux logiciels libres. Avec les années, plusieurs pistes de solutions pour tenter de comprendre ce phénomène ont été proposées par des économistes. Dans cette section, nous souhaitons présenter ces différents modèles.

2.2.1 Lerner et Tirole : une affaire de réputation

Une des explications les plus courantes du phénomène particulier d'incitatifs économiques propres aux logiciels libres est celui de la réputation. Reprise en 2001 par MM. Lerner et Tirole dans un papier très populaire, cette théorie soutient que la participation à un projet libre agit en quelque sorte comme une carte de visite pour les développeuses et les développeurs.

Très visible, beaucoup utilisé par la communauté et facilement consultable, le code libre produit par un individu devient ainsi une bonne manière de se distinguer pour obtenir une promotion, ou encore de démontrer de manière très concrète ses capacités à un futur employeur. En effet, les questions de licences et de secret commercial empêchent bien souvent une personne à la recherche d'un emploi de partager le travail qu'elle a effectué sur des logiciels propriétaires pour démontrer son mérite.

Lerner et Tirole argumentent cependant que malgré son importance, l'attrait de la réputation est trop faible pour expliquer le succès des logiciels libres et reconnaissent ainsi un rôle (mineur) à l'altruisme dans le développement des logiciels libres.

Ces derniers mentionnent finalement que beaucoup d'entreprises, sans avoir de politiques claires sur l'utilisation de projets libres, finissent en général par utiliser une part non négligeable de logiciels libres à l'interne. Cela se fait souvent à l'insu des cadres supérieurs, mais avec la bénédiction des directions intermédiaires. Par la force des choses, une partie des employé-e-s utilisant ces projets finirait par contribuer en retour. Lerner et Tirole parlent de ce phénomène comme d'une « subvention involontaire » aux projets libres, mais peinent à comprendre que ces compagnies gagnent beaucoup à utiliser ces projets libres plutôt que des logiciels propriétaires. À notre avis, le mot « subvention » n'est donc pas adéquat et il vaudrait mieux parler de « participation involontaire ».

Malgré le succès de leur papier, ces derniers ne présentent pas de modèle économique concret pour étayer leurs réflexions. Nous retenons cependant que si leurs hypothèses sur la réputation sont vraies, il devrait exister une corrélation positive entre le salaire et la participation à des projets libres.

2.2.2 Wu, Gerlach & Young : incitatifs et économie comportementale

Dans un papier publié en 2007, Wu, Gerlach & Young cherchent à en savoir plus sur ce qui motive les personnes qui développent des logiciels libres.

En se basant sur les travaux de Lerner et Tirole précédemment mentionnés, ces derniers développent un modèle théorique d'incitatifs inspiré de l'*Expectancy-Value Theory*, une variante de la théorie du choix rationnel utilisée en économie comportementale, et sondent 148 personnes pour le mettre à l'épreuve.

Ils avancent ainsi que les incitatifs principaux poussant les individus à s'adonner aux projets libres sont :

1. la motivation intrinsèque d'aider
2. le désir d'améliorer son capital humain
3. la perspective d'obtenir un meilleur emploi
4. les besoins personnels⁴

De surcroît, les auteurs divisent ces incitatifs entre incitatifs intrinsèques (1) incitatifs extrinsèques (2, 3 et 4).

Les résultats de leur sondage démontre ainsi que :

$$1 \succ 2 \sim 3 \succ 4 \tag{2.1}$$

En langage courant, l'équation 2.1 indique ainsi que la motivation intrinsèque d'aider tend à être la raison la plus souvent citée, suivie ex æquo par le désir d'améliorer son capital humain et la perspective d'obtenir un meilleur emploi. Finalement, combler un besoin personnel serait la raison la moins populaire pour écrire du code libre.

Le faible nombre de répondant-e-s au sondage effectué ne permet pas à notre avis de tirer de conclusions claires de cette étude, le risque de biais de sélection des participant-e-s étant relativement élevé. La distinction entre incitatifs intrinsèques et extrinsèques nous semble toutefois intéressante.

4. L'auteur de ce mémoire est par exemple le fier créateur de `rename-flac`, un programme libre très simple lui permettant d'ordonner automatiquement sa collection de musique numérique, une tâche manuelle auparavant bien pénible.

2.2.3 Benkler : de l'importance des incitatifs sociaux

Yochai Benkler — spécialiste de l'économie des réseaux et professeur à l'université Harvard — se penche sur les questions liées aux incitatifs économiques dans un papier intitulé *Coases's Penguin, or, Linux and the Nature of the Firm*, paru en 2002.

Partant de l'analyse de Ronald Coase sur la place des entreprises dans l'économie (1937), Benkler argumente que le travail collaboratif (*peer-production*) — une forme de travail endémique aux logiciels libres — est un nouveau modèle de production économique distinct des modèles traditionnels.

Ce dernier soutient de ce fait que le travail coopératif en informatique libre, contrairement au milieu académique, a la particularité de mettre en commun des contributions provenant d'horizons très différents. Ainsi, il n'est pas rare que se côtoient dans un même projet libre des employé-e-s de grandes multinationales en informatique, des personnes travaillant pour des organismes à but non lucratifs et des individus non rémunérés travaillant sur le projet de manière bénévole. Comme beaucoup (Mustonen, 2003 ; Lerner, Tirole, 2001 ; Broca, 2013), Benkler affirme que c'est l'aspect décentralisé et ouvert des projets libres qui réduit le coût d'opportunité de participer à un projet libre et qui permet l'existence de cette structure particulière.

Se penchant sur la question des incitatifs économiques liés à la production de logiciels libres, Benkler affirme que le modèle économique néoclassique, où l'utilité est réduite à un équivalent monétaire, ne permet pas d'expliquer ce phénomène convenablement. Il propose ainsi un modèle prenant en compte les trois aspects suivants :

1. l'utilité de la récompense monétaire
2. l'utilité de la récompense hédoniste intrinsèque
3. l'utilité de la récompense socio-psychologique

En reprenant la classification de Wu, Gerlach & Young, on peut ainsi classer les récompenses monétaires et socio-psychologiques comme des incitatifs extrinsèques et la récompense hédoniste intrinsèque comme un incitatif intrinsèque au développement de logiciels libres. Ce modèle est à notre avis bien plus complet que celui avancé par Lerner et Tirole, qui ne considère que la réputation comme un gain monétaire différé.

De manière plus formelle, on peut représenter la proposition de Benkler sous la forme d'une fonction d'utilité espérée de type Bernoulli, où U est l'utilité, M la récompense monétaire, H la récompense hédoniste intrinsèque et S la récompense socio-psychologique. On obtient ainsi l'équation 2.2.

$$U \sim f(M, H, S) \tag{2.2}$$

2.2.4 Rossi & Coleman : hétérogénéité des pratiques

Dans le cadre d'une revue étoffée des études abordant les problèmes économiques entourant l'informatique libre, Maria Alexandra Rossi (2004) conclut que les incitatifs économiques liés aux logiciels libres sont très hétérogènes. Reprenant elle aussi la distinction entre facteurs extrinsèques et intrinsèques, elle affirme que de par leur nature décentralisée et ouverte, les projets libres sont eux-mêmes très hétérogènes.

À son avis, cette hétérogénéité rend parfois difficile la comparaison des logiciels libres ayant une base institutionnelle forte (le noyau Linux, les projets de la fondation *Apache*, etc.) avec ceux développés sur une base moins formelle (projets individuels, scripts ouverts, etc.).

Les conclusions de Rossi rejoignent les propos de Gabriella Coleman, anthropologue spécialiste des relations sociales dans le milieu des logiciels libres et professeure à l'université McGill. D'avis que la grande hétérogénéité des logiciels libres est rarement comprise par les économistes, Coleman argue que tenter de restreindre un phénomène éminemment social à sa seule manifestation économique est une erreur :

Although a number of these studies tangentially discuss ethical questions (e.g. conflict resolution within F/OSS projects), they rarely address how developers commit themselves to an ethical vision through, rather than prior to, their participation in a F/OSS project. Much of the F/OSS literature, in other words, is heavily focused on the question of motivation or incentive mechanisms, and often fails to account for the plasticity of human motivations and ethical perceptions. — (Coleman, 2013, p. 123)

Pris ensemble, ces quatre visions des incitatifs économiques liés aux logiciels libres jouent à notre avis un rôle complémentaire et se renforcent mutuellement. Phénomène hétérogène ancré dans la diversité des individus et des communautés qui la compose, le monde des logiciels libres a très clairement à la fois un aspect économique et un aspect social.

De plus, la distinction entre incitatifs intrinsèques et extrinsèques nous semble très appropriée, apportant à la fois structure et clarté à un fait social parfois difficile à comprendre de par ses frontières floues. Grâce aux études abordées dans cette section, nous avons identifié deux incitatifs extrinsèques et deux incitatifs intrinsèques nous semblant jouer un rôle central dans la production de code informatique libre. Respectivement, ces derniers sont le gain monétaire espéré par l'écriture de code libre, le sentiment d'appartenance à la communauté libre, le plaisir d'écrire du code informatique libre et la gratification altruiste d'offrir du code libre.

Nous reviendrons à ces incitatifs dans le troisième chapitre de ce mémoire, où nous tenterons de valider empiriquement leur existence.

CHAPITRE 3

ANALYSE EMPIRIQUE DES INCITATIFS ÉCONOMIQUES LIÉS AUX LOGICIELS LIBRES

Dans le second chapitre de ce mémoire, nous avons effectué une analyse des différentes études portant sur les incitatifs économiques liés aux logiciels libres. Nous avons ainsi pu voir que malgré la grande hétérogénéité des facteurs poussant les individus à travailler sur des projets libres, il est possible de catégoriser les incitatifs économiques ayant une influence notable entre intrinsèques et extrinsèques.

Si plusieurs études précédemment mentionnées incluent des analyses empiriques pour appuyer les modèles développés, les données utilisées sont souvent assez anciennes et ont un échantillonnage relativement faible. Dans ce chapitre, nous proposons donc d'analyser ces modèles avec une base de données récente et ayant un large échantillonnage, dans le but de les valider ou les invalider empiriquement.

Plus précisément, nous commencerons par regarder si la participation à des projets libres a un effet sur le salaire annuel des développeuses et des développeurs, comme l'affirment Lerner, Tirole (2001), Mustonen (2003) et bien d'autres. Nous regarderons par la suite l'effet de la participation à des projets libres sur le plaisir personnel résultant de l'écriture de code informatique.

Nous procéderons finalement à l'analyse de divers facteurs comme le sentiment d'appartenance à la communauté informatique, le désir d'écrire du code pour des raisons altruistes et la satisfaction au travail.

3.1 Description de la base de données

Stack Overflow est un site web très populaire chez les développeurs et les développeuses qui permet de poser des questions en lien avec la programmation informatique et de recevoir des réponses et des conseils de la part des autres membres de la communauté. Les personnes utilisant le site peuvent voter pour les meilleures réponses, ce qui crée un classement très efficace. En juin 2019, le site se classait au 47^e rang des sites les plus consultés en ligne selon la compagnie *Alexa Internet, Inc.* (2019), qui se spécialise dans ce genre de palmarès.

À toutes les années depuis 2011, Stack Overflow demande aux membres de sa communauté de remplir un sondage en ligne comportant plus d'une centaine de questions portant sur divers aspects de la profession informatique. Par exemple, les répondant-e-s sont sondé-e-s sur leur participation à des projets libres, sur les technologies utilisées dans le cadre de leur travail, sur leur salaire, sur leurs perspectives de carrière ou encore sur le nombre et la taille de leurs écrans au travail.

Stack Overflow effectue ce sondage annuel pour promouvoir son image de marque et comme moyen de collecter une panoplie de données sur ses utilisatrices et utilisateurs. Ces données peuvent par la suite être utilisées à l'interne pour améliorer leurs services commerciaux, notamment leur service de recrutement professionnel.

Il est intéressant de noter qu'avec le temps, à l'instar du *CEO Confidence Survey* produit par le *Conference Board* américain, ce sondage est devenu une véritable institution de la communauté informatique. Ces données sont ainsi fréquemment reprises par des médias spécialisés pour analyser l'évolution des tendances lourdes dans l'industrie (Bright, 2018, 2019).

Les réponses à ce sondage sont mises en ligne et accessibles sous la licence libre *Open Database License*. Nous avons choisi de travailler avec les données issues du sondage effectué en 2018 (Stack Overflow, 2018) car les questions posées abordent des sujets connexes à notre sujet d'étude.

Le sondage a été effectué entre le 8 et le 28 janvier 2018, soit sur une période de 20 jours. Comme la base de donnée provient de ce sondage ponctuel et n'est pas le résultat d'une étude longitudinale, les réponses sont donc un portrait de l'opinion des répondant-e-s à cette date. Conséquemment, il ne nous est pas non plus possible d'utiliser les données de plusieurs sondages pour analyser l'évolution des réponses dans le temps.

Plus de cent mille développeuses et développeurs informatiques provenant de 183 pays différents ont répondu à ce sondage. Les réponses de 20 000 autres répondant-e-s n'ont pas été conservées par le service statistique de Stack Overflow, car elles et ils n'ont pas répondu à assez de questions pour que leur questionnaire soit considéré valide. La validité d'un questionnaire a été établie à partir du ratio entre le temps de réponse et le nombre de questions remplies. La plupart des questionnaires remplis en moins de 5 minutes n'ont pas été conservés. Il n'était pas possible pour une même personne de répondre au sondage plusieurs fois.

Sur les 101 592 personnes ayant des questionnaires considérés comme valides, 67 441 (66,4 %) ont complété le sondage dans son entièreté.

Les répondant-e-s ont été recruté-e-s en ligne. Les cinq moyens les plus efficaces pour recruter des répondant-e-s se sont avérés être :

1. L'achat de bannières publicitaires sur le web
2. La publication de messages sur des listes courriels
3. L'affichage de publicités sur le site `stackoverflow.com`
4. La publication d'entrées de blogue sur le site `stackoverflow.blog`
5. La publication de messages sur Twitter

Stack Overflow reconnaît que ses méthodes de recrutement des répondant-e-s ont favorisé la participation des membres les plus actifs de sa communauté, au détriment des personnes n'utilisant que rarement le site. De plus, les répondant-e-s qui ont terminé le sondage se sont vu attribuer un badge spécial dans leur profil Stack Overflow faisant mention dudit sondage.

Il est ainsi probable qu'il existe un biais de sélection dans les données finales, causé par une différence entre la population étant portée à répondre au sondage annuel de Stack Overflow et la population regroupant l'ensemble des développeuses et des développeurs. À notre avis, le nombre important de répondant-e-s tend toutefois à minimiser le risque d'un biais marqué.

À notre connaissance, aucun travail académique en économie n'a été réalisé à partir du sondage de 2018. Des études portant sur la participation des femmes en informatique ont toutefois utilisé les données de sondages précédents (Ford et al., 2016, 2017).

Stack Overflow met à notre disposition un total de 129 variables différentes après avoir retiré celles comportant des informations permettant d'identifier un individu. Sur ces 129 variables, seules les variables présentées à la table 3.1 ont été retenues.

Nous avons retenu ces variables pour leur lien spécifique avec notre sujet d'étude. La liste complète des variables de la base de données originelle ainsi que leur description est disponible en ligne sur le site de présentation de la base de données (Stack Overflow, 2018).

TABLEAU 3.1 – Description des variables retenues dans la base de données

Nom de la variable	Description
Age	L'âge
Gender	Le genre
RaceEthnicity	L'identification à un groupe ethnique
Country	Le pays de résidence
FormalEducation	Le niveau d'éducation formelle
EducationParents	Le niveau d'éducation le plus élevé des parents
Employment	Le statut d'emploi
ConvertedSalary	Le salaire annuel en USD basé sur le taux de change en vigueur le 2018-01-18, en prenant pour acquis 50 semaines travaillées par an
Hobby	Si les répondant-e-s écrivent du code informatique comme loisir
OpenSource	Si les répondant-e-s participent à des projets libres
YearsCoding	Le nombre d'années d'expérience d'écriture de code informatique
JobSatisfaction	La satisfaction au travail
CareerSatisfaction	La satisfaction du choix de carrière
EducationTypes	L'expérience non académique

suite à la page suivante

TABLEAU 3.1 – Description des variables retenues dans la base de donnée

AgreeDisagree1	Si les répondant-e-s éprouvent un lien d'appartenance avec la communauté informatique
EthicsChoice	Si les répondant-e-s écriraient du code informatique à des fins jugées profondément immorales
EthicsReport	Si les répondant-e-s dénonceraient du code informatique jugé profondément immoral
EthicalImplications	Si les répondant-e-s pensent avoir une obligation morale d'analyser les conséquence de l'écriture de code informatique
LanguageWorkedWith	Les langages de programmation fréquemment utilisés
DatabaseWorkedWith	Les logiciels de gestion de bases de données fréquemment utilisés
PlatformWorkedWith	Les plateformes informatiques fréquemment utilisées
IDE	Les environnements de programmation fréquemment utilisés
OperatingSystem	Les systèmes d'exploitation fréquemment utilisés

3.2 Méthodologie

À l'aide de base de données, nous effectuons une série de régressions à l'aide d'un estimateur de type « Moindres carrés ordinaire » (MCO). Plus précisément, nous souhaitons estimer l'effet de la participation à un projet libre sur :

1. le salaire
2. le plaisir ressenti à écrire du code informatique
3. le sentiment d'appartenance à la communauté informatique
4. l'altruisme
5. la satisfaction au travail

Pour des raisons techniques, nous modifions certaines variables provenant de la base de données originelle. En effet, plusieurs d'entre elles sont des variables catégoriques que nous traiterons plutôt comme des variables dichotomiques. Ces modifications sont présentées à la table 3.2.

TABLEAU 3.2 – Description des variables modifiées dans la nouvelle base de données

Nom de la variable	Description
Age0017	Si l'individu a entre 0 et 17 ans
Age1824	Si l'individu a entre 18 et 24 ans
Age2534	Si l'individu a entre 25 et 34 ans
Age3544	Si l'individu a entre 35 et 44 ans
Age4454	Si l'individu a entre 45 et 54 ans
Age5564	Si l'individu a entre 55 et 64 ans
Age65More	Si l'individu a 65 ans ou plus
GenderDummy	Si l'individu est un homme
RaceEthnicityDummy	Si l'individu est de descendance européenne
CountryHDI	L'indice de développement humain du pays
FormalEducationDummy	Si l'individu détient un diplôme universitaire
EducationParentsDummy	Si un parent de l'individu détient un diplôme universitaire
EmploymentFull	Si l'individu travaille à temps plein
EmploymentPartSelf	Si l'individu travaille à temps partiel ou travaille à son compte
EmploymentOther	Si l'individu est à la retraite, ne travaille pas ou a un statut d'emploi autre

suite à la page suivante

TABLEAU 3.2 – Description des variables modifiées dans la nouvelle base de données

More8YearsCoding	Si l'individu écrit du code informatique depuis plus de 8 ans
OpenSourceEducation	Si l'individu a déjà participé à un projet libre à des fins éducatives
LanguageProprietary	Si l'individu travaille avec des langages de programmation propriétaires
DatabaseProprietary	Si l'individu travaille avec des logiciels de gestion de base de données propriétaires
PlatformProprietary	Si l'individu travaille avec des plateformes informatiques propriétaires
IDEProprietary	Si l'individu utilise un environnement de programmation propriétaire
OSProprietary	Si l'individu utilise un système d'exploitation propriétaire
Altruism	Le degré d'altruisme
Satisfaction	Le degré de satisfaction au travail
OpenSourceComposite	Le degré de participation à la communauté libre

La variable `CountryHDI` est obtenue en croisant la variable `Country` avec l'indice de développement humain du pays en 2018 (UNDP, 2019). Cela nous permet en effet d'obtenir de l'information sur le niveau de développement du pays de résidence du répondant ou de la répondante lors de l'année du sondage.

Pour étudier l'appartenance à la communauté informatique, nous renommons la variable `AgreeDisagree1` par `Belonging` et la transformons en variable continue allant de 0 à 1. Ainsi, plus les répondant-e-s éprouvent un lien d'appartenance fort à la communauté, plus la variable `Belonging` s'approche de 1.

L'altruisme, de par sa nature morale et philosophique, est une chose complexe à quantifier. Comme il ne nous est malheureusement pas possible de mesurer cette variable directement à partir de notre base de données, nous avons choisi de retenir les variables `EthicsChoice`, `EthicsReport` et `EthicalImplications` comme indicateurs d'altruisme.

En effet, il y a à notre avis un lien fort entre le refus d'écrire du code informatique pour des raisons morales et le niveau d'altruisme chez une personne. En général, lorsqu'une personne refuse d'écrire du code qu'elle juge immoral, c'est parce qu'elle se préoccupe des répercussions possibles sur ses pairs et sur la société en général.

On forme donc la variable `Altruism`, allant de 0 à 1, pour étudier le niveau d'altruisme des individus sondés, où la valeur de la variable est donnée par l'équation 3.1.

$$Altruism = \frac{EthicsChoice + EthicsReport + EthicalImplications}{3} \quad (3.1)$$

Ainsi, si une personne refuserait d'écrire du code informatique à des fins jugées profondément immorales, si elle dénoncerait publiquement le fait de se faire demander une telle chose et qu'elle pense avoir l'obligation d'analyser les conséquences morales d'écrire du code informatique, sa variable `Altruism` serait de 1.

Contrairement à l'altruisme, la satisfaction au travail est nettement plus simple à évaluer ; on dispose en effet de deux variables appropriées, `JobSatisfaction` et `CareerSatisfaction`. On agrège de fait ces deux variables via l'équation 3.2. La variable résultante, `Satisfaction`, va de 0 à 1 : plus une personne est satisfaite au travail plus son score se rapproche de 1.

$$Satisfaction = \frac{JobSatisfaction + CareerSatisfaction}{2} \quad (3.2)$$

Finalement, on crée la variable `OpenSourceComposite` pour tenter de quantifier le degré de participation à la communauté libre d'un répondant ou d'une répondante. Cette variable composite est continue allant de 0 à 1 et est définie par l'équation 3.3, où :

- `OpenSourceComposite` = OS_C
- `OpenSource` = OS
- `OpenSourceEducation` = OS_E
- `LanguageProprietary` = L_P
- `DatabaseProprietary` = D_P
- `PlatformProprietary` = P_P
- `IDEProprietary` = I_P
- `OSProprietary` = O_P

$$\begin{aligned}
 OS_C &= \frac{1}{3} \cdot OS + \frac{1}{9} \cdot (OS_E + \neg L_P + \neg D_P + \neg P_P + \neg I_P + \neg O_P) \\
 &= \frac{1}{3} \cdot OS + \frac{1}{9} \cdot (OS_E + 1 - L_P + 1 - D_P + 1 - P_P + 1 - I_P + 1 - O_P) \\
 &= \frac{1}{3} \cdot OS + \frac{1}{9} \cdot OS_E - \frac{1}{9} \cdot (L_P + D_P + P_P + I_P + O_P) + \frac{5}{9}
 \end{aligned} \tag{3.3}$$

Par exemple, si une personne participe à des projets libres, a déjà participé à un projet libre à des fins éducatives, programme avec un langage dont le compilateur est libre, travaille avec des bases de données libres, utilise un environnement de programmation libre ainsi que Linux, sa variable `OpenSourceComposite` serait de 1. À l'inverse, une personne affirmant ne pas participer à la communauté libre et utilisant exclusivement des technologies propriétaires aurait un score de 0.

Dans l'équation 3.3, nous attribuons des poids différents aux variables qui composent `OpenSourceComposite`. Si le rapport entre la variable `OpenSource` ($1/3$) et les autres variables utilisées ($1/9$) est en définitive arbitraire, il est à notre avis important d'attribuer plus de poids à `OpenSource` pour mettre l'accent sur son importance. En définitive, une développeuse ou un développeur qui utilise des logiciels libres mais qui affirme ne pas contribuer à ces derniers ne participe pas réellement à cette communauté.¹

1. Contrairement à ce qu'une personne n'évoluant pas dans la communauté libre pourrait être portée à croire, il n'est pas nécessaire d'écrire du code informatique ou même de savoir coder pour contribuer aux logiciels libres. En effet, de nombreuses formes de contributions non techniques sont nécessaires à la pérennité d'un projet. Par exemple, une personne peut soumettre un rapport de bogue en cas de problèmes, participer à la traduction de la documentation ou de l'interface graphique d'un programme ou encore participer à la gestion des affaires administratives d'un projet.

3.2.1 Statistiques descriptives

À tous les ans, Stack Overflow (2018) rend disponible en ligne un rapport statistique détaillé préparé par son équipe de statisticien-ne-s pour accompagner la publication des données de son sondage annuel.

Nous encourageons le lecteur ou la lectrice intéressé-e par la distribution statistique des variables utilisées dans ce travail à consulter ce rapport ; interactif et très détaillé, il saura mettre en valeur notre base de données bien mieux que ce nous aurions pu présenter ici.

Comme les variables dichotomiques agrégées présentées précédemment ne font bien évidemment pas partie des statistiques descriptives publiées par Stack Overflow, nous présentons ces données à la table 3.3.

3.2.2 Le salaire

On souhaite vérifier l'effet de la participation à un projet libre sur le salaire de l'individu. On pose ainsi l'équation 3.4.

$$\textit{ConvertedSalary} = \beta_0 + \beta X_1 + \delta D_1 + \epsilon \quad (3.4)$$

ϵ est un vecteur de termes d'erreur. Les matrices lignes β et δ contiennent respectivement les vecteurs de coefficient de régression des vecteurs colonnes X_1 et D_1 . On définit ces derniers ainsi :

$$X_1 = \left[\text{CountryHDI} \right] \quad (3.5)$$

$$D_1 = \begin{bmatrix} \text{OpenSource} \\ \text{Age0017} \\ \text{Age1824} \\ \text{Age3544} \\ \text{Age4554} \\ \text{Age5564} \\ \text{Age65More} \\ \text{GenderDummy} \\ \text{RaceEthnicityDummy} \\ \text{FormalEducationDummy} \\ \text{EducationParentsDummy} \\ \text{EmploymentPartSelf} \\ \text{EmploymentOther} \\ \text{More8YearsCoding} \\ \text{Hobby} \end{bmatrix}$$

Les termes de la matrice X_1 sont des variables continues, alors que ceux de la matrice D_1 sont des variables dichotomiques.

De manière similaire, on calcule l'effet de la variable continue `OpenSourceComposite` sur le salaire par l'équation 3.6.

$$\text{ConvertedSalary}_c = \beta_0 + \beta X_2 + \delta D_2 + \epsilon \quad (3.6)$$

$$X_2 = \begin{bmatrix} \textit{OpenSourceComposite} \\ \textit{CountryHDI} \end{bmatrix} \quad (3.7)$$

$$D_2 = \begin{bmatrix} \textit{Age0017} \\ \textit{Age1824} \\ \textit{Age3544} \\ \textit{Age4554} \\ \textit{Age5564} \\ \textit{Age65More} \\ \textit{GenderDummy} \\ \textit{RaceEthnicityDummy} \\ \textit{FormalEducationDummy} \\ \textit{EducationParentsDummy} \\ \textit{EmploymentPartSelf} \\ \textit{EmploymentOther} \\ \textit{More8YearsCoding} \\ \textit{Hobby} \end{bmatrix}$$

Les résultats des équations 3.4 et 3.6 sont respectivement présentés aux tableaux 3.4 et 3.5.

3.2.3 Le plaisir d'écrire du code informatique

De manière similaire, on cherche à établir l'effet de la participation à un projet libre sur le plaisir d'écrire du code informatique, représenté par la variable *Hobby*. On prend ainsi pour acquis que si une personne écrit du code informatique comme hobby, elle tire de cette activité un certain plaisir. On pose ainsi l'équation 3.8.

$$Hobby = \beta_0 + \beta X_3 + \delta D_3 + \epsilon \quad (3.8)$$

$$X_3 = \begin{bmatrix} CountryHDI \\ ConvertedSalary \end{bmatrix} \quad (3.9)$$

$$D_3 = \begin{bmatrix} OpenSource \\ Age0017 \\ Age1824 \\ Age3544 \\ Age4554 \\ Age5564 \\ Age65More \\ GenderDummy \\ RaceEthnicityDummy \\ FormalEducationDummy \\ EducationParentsDummy \\ EmploymentPartSelf \\ EmploymentOther \\ More8YearsCoding \end{bmatrix}$$

De manière similaire, on calcule l'effet de la variable continue `OpenSourceComposite` sur le plaisir d'écrire du code informatique par l'équation 3.10.

$$Hobby_c = \beta_0 + \beta X_4 + \delta D_4 + \epsilon \quad (3.10)$$

$$X_4 = \begin{bmatrix} OpenSourceComposite \\ CountryHDI \\ ConvertedSalary \end{bmatrix} \quad (3.11)$$

$$D_4 = \begin{bmatrix} Age0017 \\ Age1824 \\ Age3544 \\ Age4554 \\ Age5564 \\ Age65More \\ GenderDummy \\ RaceEthnicityDummy \\ FormalEducationDummy \\ EducationParentsDummy \\ EmploymentPartSelf \\ EmploymentOther \\ More8YearsCoding \end{bmatrix}$$

Les résultats des équations 3.8 et 3.10 sont respectivement présentés aux tableaux 3.6 et 3.7.

3.2.4 Le sentiment d'appartenance

On souhaite maintenant établir l'effet de la participation à un projet libre sur le sentiment d'appartenance à la communauté informatique. On pose ainsi l'équation 3.12.

$$\textit{Belonging} = \beta_0 + \beta X_3 + \delta D_1 + \epsilon \quad (3.12)$$

De manière similaire, on calcule l'effet de la variable continue `OpenSourceComposite` sur ce même sentiment d'appartenance par l'équation 3.13.

$$\textit{Belonging}_c = \beta_0 + \beta X_4 + \delta D_2 + \epsilon \quad (3.13)$$

Les résultats des équations 3.12 et 3.13 sont respectivement présentés aux tableaux 3.8 et 3.9.

3.2.5 L'altruisme

On souhaite établir l'effet de la participation à un projet libre sur l'altruisme des personnes sondées. On pose ainsi l'équation 3.14.

$$\textit{Altruism} = \beta_0 + \beta X_3 + \delta D_1 + \epsilon \quad (3.14)$$

De manière similaire, on calcule l'effet de la variable continue `OpenSourceComposite` sur l'altruisme grâce à l'équation 3.15.

$$\textit{Altruism}_c = \beta_0 + \beta X_4 + \delta D_2 + \epsilon \quad (3.15)$$

Les résultats des équations 3.14 et 3.15 sont respectivement présentés aux tableaux 3.10 et 3.11.

3.2.6 La satisfaction au travail

On souhaite établir l'effet de la participation à un projet libre sur la satisfaction au travail des personnes sondées. On pose ainsi l'équation 3.16.

$$Satisfaction = \beta_0 + \beta X_3 + \delta D_1 + \epsilon \quad (3.16)$$

De manière similaire, on calcule l'effet de la variable continue `OpenSourceComposite` sur la satisfaction au travail grâce à l'équation 3.17.

$$Satisfaction_c = \beta_0 + \beta X_4 + \delta D_2 + \epsilon \quad (3.17)$$

Les résultats des équations 3.16 et 3.17 sont respectivement présentés aux tableaux 3.12 et 3.13.

3.2.7 Choix des unités de mesure

Dans le cadre de ce travail, nous avons choisi d'effectuer des régressions en niveau plutôt que de procéder à une transformation logarithmique. Si on aurait pu s'attendre à ce que ce genre de transformation soit effectuée — particulièrement pour les équations 3.4 et 3.6² — la particularité de notre base de données nous a poussé à faire autrement.

En effet, la majorité de nos variables sont dichotomiques ; s'il est possible de les traiter pour les inclure dans une fonction logarithmique, cela aurait nécessité un traitement statistique supplémentaire et aurait complexifié l'interprétation des résultats.

De plus, d'aucuns apprécient utiliser des fonctions logarithmiques car elles permettent d'interpréter les coefficients de régression en termes de pourcentages. Dans le cas des équations 3.8, 3.10, 3.12, 3.13, 3.14, 3.15, 3.16 et 3.17, la valeur de la variable dépendante analysée est par construction entre 0 et 1. On obtient ainsi le même résultat.

2. Il est courant de voir des variables comme le salaire représentées sous la forme d'une fonction logarithmique du niveau d'éducation et de l'expérience sur le marché du travail (équation de Mincer) (Heckman et al., 2003).

3.2.8 Tests statistiques

Colinéarité parfaite

Pour éviter des problèmes de colinéarité parfaite, on exclut des matrices D_1 D_2 , D_3 , D_4 , D_5 , D_6 , D_7 , D_8 , D_9 et D_{10} les variables `Age2534` et `EmploymentFull`, que l'on peut ainsi considérer comme des situations de référence. La valeur du coefficient de régression de ces variables peut être obtenue par le théorème de Frisch-Waugh.

Endogénéité

Comme nous l'expliquons en détail à la section 2.3, les résultats que nous obtenons démontrent une corrélation entre les variables dépendantes étudiées et les deux variables indépendantes qui nous intéressent (`OpenSource` et `OpenSourceComposite`). Il ne nous est cependant pas possible de parler de relations causales, notamment pour des questions d'endogénéité.

Dans *A Guide to Modern Econometrics*, Marno Verbeek affirme en effet :

[The problem of endogeneity] also arises if there are unobservable omitted factors in the model that happen to be correlated with one or more of the explanatory variables. This bias is of particular concern when we wish to attach a causal interpretation to our model coefficients, in which case the ceteris paribus condition includes all other factors that have an impact on the outcome variable y_i , whether observed or unobserved. — (Verbeek, 2017, p. 146)

Il est ainsi pertinent de se demander si nos deux variables d'intérêts sont exogènes, une condition nécessaire pour conclure l'existence d'une relation causale. Sans nous avancer dans l'élaboration d'un modèle complexe pour expliquer la variable `OpenSource`, on peut raisonnablement conclure que cette dernière est le résultat d'interactions complexes et variées et que nous avons un problème de variables omises. Il ne serait pas non plus surprenant qu'un phénomène de causalité inverse (*reverse causality*) existe entre les variables `OpenSource` et `ConvertedSalary`, tel qu'avancé dans la section d'analyse des résultats.

Par construction, la présence d'endogénéité dans la variable `OpenSource` implique la présence d'endogénéité dans la variable `OpenSourceComposite`. Notons toutefois que plusieurs variables explicatives dans nos équations utilisées à fin de contrôle, comme `Age`, `GenderDummy` et `RaceEthnicityDummy`, sont effectivement exogènes.

Plusieurs techniques existent pour prendre en compte de la présence d'endogénéité dans nos données, la plus courante étant l'élaboration d'un modèle à variables instrumentales. Ce genre de modèle permet d'effectuer un test de Durbin-Wu-Hausman pour vérifier la présence d'endogénéité. Malheureusement, il nous semble très difficile, voire impossible de procéder de la sorte : la base de données avec laquelle nous travaillons ne fournit pas les instruments nécessaires pour réaliser un tel exercice.

Le lecteur ou la lectrice devrait donc être conscient-e de cette problématique probable lors de la lecture de ce travail et analyser les résultats fournis en toute connaissance de cause.

Si nous avons pu participer à la collecte de données, il aurait été intéressant de poser des questions supplémentaires aux répondant-e-s et d'obtenir des variables permettant d'effectuer les tests précédemment mentionnés.

Pour obtenir un instrument corrélé avec la participation à un projet libre sans l'être avec le salaire ou la satisfaction au travail, on aurait ainsi pu leur demander : « Dans la dernière année, avez-vous recommandé à vos ami-e-s ou à vos proches l'utilisation de logiciels libres comme alternatives à des logiciels propriétaires ? ».

L'intuition derrière cette question est que recommander un logiciel libre à son entourage est un bon indicateur que l'on participe à un projet libre ou que l'on fait parti de la communauté libre. On peut de plus raisonnablement supposer que ce comportement est indépendant du salaire, de la satisfaction au travail et des autres variables explicatives utilisées aux équations 3.4 et 3.16.

Pour le modèle présenté à l'équation 3.12, tentant d'évaluer le lien entre la participation à un projet libre et le sentiment d'appartenance à la communauté informatique, ou encore celui à l'équation 3.14, traitant de la propension à l'altruisme, on aurait pu suivre un raisonnement similaire et poser la question suivante : « Avez-vous déjà utilisé vos contributions à un ou des projets libres (par exemple sur Gitlab ou Github) comme argument en votre faveur lors d'un processus d'embauche ? ».

Finalement, comme instrument permettant d'utiliser un modèle à variables instrumentales pour traiter la possibilité d'endogénéité dans la relation entre la participation à un projet libre et le fait d'écrire du code informatique comme hobby (équation 3.8), nous aurions pu demander aux répondant-e-s : « Écrivez-vous du code libre dans le cadre de votre travail à la demande de votre employeur ? ».

Hétéroscédasticité

Pour chacune des régressions décrites dans la section 2.2, nous avons effectué un test de Breusch-Pagan pour vérifier que la variance des termes d'erreurs est constante. Nous constatons ainsi la présence d'hétéroscédasticité dans nos données.

Pour pallier à ce problème, les écarts types présentés à l'annexe B ont été générés à l'aide d'un estimateur de White (1980), où la matrice de variance-covariance prend la forme suivante :

$$(X'X)^{-1}X'\Omega X(X'X)^{-1}$$

3.3 Analyse des résultats

Dans ce chapitre, nous avons comme but de tester les différentes hypothèses avancées par le panel d'expert-e-s présenté au second chapitre. Plus précisément, nous souhaitons vérifier le lien entre la participation à un projet libre et le salaire, le plaisir ressenti à écrire du code informatique, le sentiment d'appartenance à la communauté informatique, l'altruisme et la satisfaction au travail. Peut-on considérer ces facteurs comme des incitatifs au libre ?

En analysant les résultats obtenus au tableau 3.4, il semble qu'il existe bel et bien un lien entre le salaire d'une personne travaillant en informatique et sa préférence pour les logiciels libres. En effet, le coefficient de régression estimé pour la variable `OpenSource` à l'équation 3.4 est de 9 101 USD, ce qui représente environ 17 % du salaire annuel médian gagné par les individus ayant répondu au sondage. Qui plus est, cette relation a une valeur-p quasi-nulle et est donc très significative.

On ne peut cependant pas en dire autant pour le coefficient de régression estimé pour la variable `OpenSourceComposite` à l'équation 3.6, présenté dans le tableau 3.5, ce dernier ayant une valeur-p supérieure à 0.05, seuil minimal que nous sommes prêts à accepter dans le cadre de ce mémoire.

Peut-on donc conclure, à l'instar de Lerner et Tirole, qu'en définitive, les développeuses et les développeurs écrivent des logiciels libres pour gagner un plus gros salaire ? Cela serait à notre avis présomptueux.

En effet, si nous avons prouvé qu'il existe une corrélation forte et significative entre le salaire et la participation à un projet libre, il ne nous est pas possible de nous prononcer sur la causalité de ce lien. Tel qu'argumenté par Angrist et Krueger (1999), une analyse statistique plus poussée nous permettrait sûrement d'aller plus loin et de s'aventurer sur le terrain des relations causales ; cela dépasse cependant le cadre de ce mémoire. Lerner et Tirole ont sûrement partiellement raison, mais il se pourrait par exemple que la relation soit inverse.

Les gens qui gagnent un salaire plus élevé ont généralement plus de latitude financière et sont plus à même de négocier des clauses dans leur contrat d'embauche quant à la propriété intellectuelle du code qu'elles et ils écrivent ou encore de prendre le temps de choisir un emploi qui leur permet d'écrire du libre au travail. Chose certaine, nos résultats n'infirmement pas leurs hypothèses.

Qu'en est-il des hypothèses de Wu, Gerlarch & Young, résumées par l'équation 2.1 ? Malheureusement pour nous, il semble que notre base de données soit peu adaptée à valider ou invalider leur modèle.

En effet, s'il est possible d'affirmer que « le désir d'améliorer son capital humain » et « la perspective d'obtenir un meilleur emploi » peuvent être représentés par l'instrument `ConvertedSalary`, on commence à tirer l'élastique en tentant de représenter « la motivation intrinsèque d'aider » par notre variable `Altruism`. Qui plus est, nous n'avons pas de données pouvant prétendre mesurer le nombre de répondant-e-s qui écrivent du code libre pour combler leurs « besoins personnels ».

Attardons-nous tout de même aux résultats des tableaux 3.10 et 3.11, qui présentent respectivement l'effet de la participation à un projet libre et l'effet du degré de participation à la communauté libre sur l'altruisme. On constate de fait que les coefficients de régression estimés pour les variables `OpenSource` (équation 3.14) et `OpenSourceComposite` (équation 3.15) sont fortement significatifs.

Ces coefficients sont toutefois assez faibles, marquant une différence d'environ 2 points de pourcentage chez les individus affirmant écrire du code libre et de 4 points de pourcentage pour une personne étant très impliquée dans la communauté libre. Plutôt que d'en conclure que l'altruisme n'est pas un incitatif important à l'écriture de code libre, nous préférons plutôt retenir que la construction de la variable `Altruism` à l'équation 3.1 semble problématique.

En effet, comme abordé dans la section dédiée à la construction de cette variable, l'altruisme est une chose difficile à quantifier empiriquement. Si la relation mise de l'avant par ces deux équations est certainement réelle, nous ne sommes toutefois pas prêts à affirmer que la corrélation mesurée représente l'altruisme comme phénomène social dans toute sa complexité. Pourrait-on plutôt parler d'une propension éthique marginalement supérieure chez les gens qui s'impliquent activement dans les communautés libres ?

À l'instar de l'altruisme, les coefficients de régression estimés exprimant la relation entre la participation à un projet libre et le sentiment d'appartenance à la communauté informatique (tableaux 3.8 et 3.9) sont faibles et très significatifs. Ainsi, le coefficient estimé pour la variable `OpenSource` (équation 3.12) est de 0.03 et celui pour la variable `OpenSourceComposite` (équation 3.13) est de 0.04.

Ces résultats sont surprenants. À première vue, on aurait pu s'attendre à ce que la réputation de la communauté libre pour son fort esprit de corps, ainsi que son attachement à un ethos souvent contraignant, se traduise par un sentiment d'appartenance élevé. En poussant l'analyse plus loin, on constate toutefois que seul 8 % des répondant-es affirmaient ne pas ressentir de tel sentiment. À notre avis, la faiblesse des coefficients estimés peut être expliquée par le fort sentiment d'appartenance présent dans l'ensemble de la profession informatique, libre ou non.

Le troisième modèle que nous avons analysé au chapitre deux est celui de Yochai Benkler, que nous avons résumé à l'équation 2.2. En reprenant les termes utilisés par Benkler, on peut ainsi associer « l'utilité de la récompense monétaire » à la variable `ConvertedSalary`, « l'utilité de la récompense hédoniste intrinsèque » aux variables `Hobby` et `Altruism`, et finalement, « l'utilité de la récompense socio-psychologique » à la variable `Belonging`. Seuls les résultats concernant la variable `Hobby` n'ont pas déjà été analysés plus haut.

Comme on peut l'observer aux tableaux 3.6 et 3.7, les coefficients de régression estimés décrivant la relation entre la participation à un projet libre et le plaisir d'écrire du code informatique sont très significatifs. Il semble ainsi qu'une personne qui participe à un projet libre (variable `OpenSource` dans l'équation 3.8) voit ses chances d'écrire du code informatique comme hobby augmenter de 17 points de pourcentage. Pour une personne très fortement impliquée dans la communauté libre (variable `OpenSourceComposite` dans l'équation 3.10), cette augmentation se chiffre à 23 points de pourcentage.

Nos résultats empiriques semblent donc supporter la relation décrite à l'équation 2.2. Pour en dire plus, il serait à notre avis nécessaire d'effectuer une analyse plus profonde des implications de ce modèle. Il serait ainsi intéressant de bâtir dans un travail subséquent un vrai modèle expérimental en tentant d'identifier les variables composant M , H et S et de tester son pouvoir de prédiction.

La seule relation que nous n'avons pas encore analysée est celle reliant la participation à la communauté libre et la satisfaction au travail. Présentés aux tableaux 3.12 et 3.13, les résultats obtenus sont relativement décevants. En effet, si les coefficients de régression estimés pour les variables `OpenSource` (équation 3.16) et `OpenSourceComposite` (équation 3.17) sont fortement significatifs, les valeurs obtenues sont très faibles.

Avec un peu de recul, ces résultats ont du sens. En effet, si nous connaissons le degré de satisfaction au travail des répondant-e-s, nous ne savons pas si elles et ils écrivent du code libre au travail. Il est ainsi peu probable qu'une personne fortement impliquée dans la communauté libre mais forcée d'écrire du code propriétaire pour gagner sa vie soit très satisfaite de son emploi. À l'inverse, on peut facilement imaginer la frustration d'une personne qui, obligée par son employeur à écrire du libre, n'en voit pas l'intérêt. Il nous aurait malheureusement fallu une variable supplémentaire pour être en mesure de tirer des conclusions intéressantes de cette relation.

CONCLUSION

Au cours de ce travail, nous avons abordé plusieurs questions importantes quant à la structure d'incitatifs économiques liés aux logiciels libres. Si au premier abord la maintenance collective de communs numériques libres peut sembler – du point de vue de l'économie néoclassique – vouée à l'échec, nous avons démontré que d'autres approches sont à même d'expliquer ce phénomène. Ainsi, les travaux d'Elinor Ostrom sur la gestion collective des communs semblent à notre avis décrire le modèle de développement propre aux logiciels libres avec beaucoup d'acuité.

Nous nous sommes par la suite attardés à l'étude de travaux économiques tentant de modéliser les incitatifs à l'écriture de code libre. Nous avons ainsi pu constater que ces derniers sont généralement peu complexes et ont rarement été vérifiés à l'aide de données empiriques valides. Si certains modèles, comme celui présenté par Lerner et Tirole, mettent l'accent sur l'espérance de gain monétaire comme incitatif principal, nous leur avons préféré des modèles plus complexes, prenant également en compte des facteurs socio-psychologiques et faisant la distinction entre incitatifs intrinsèques et extrinsèques. En définitive, la nature très hétérogène des projets libres complexifie l'élaboration d'un modèle unique et détaillé.

Pour tenter d'y voir plus clair, nous avons finalement analysé de manière empirique les hypothèses à la base de ces modèles grâce aux données du *Stack Overflow Developers Survey 2018*, un sondage exhaustif ayant été rempli par plus de 100 000 développeuses et développeurs. Nous avons de fait découvert une corrélation forte et significative entre la participation à un projet libre et un salaire plus élevé. De même, il semble exister une corrélation forte et significative entre la participation à un projet libre et le plaisir d'écrire du code informatique. Nous avons également trouvé qu'à un plus faible degré, la participation à un projet libre est corrélée positivement avec le sentiment d'appartenance à la communauté informatique, l'altruisme et la satisfaction au travail. Ces résultats doivent cependant être analysés en prenant en compte la présence possible d'endogénéité dans nos équations.

Comme nous l'avons présenté dans le second chapitre, la question des incitatifs économiques au libre est très souvent associée à la nécessité de trouver des sources de financement pérennes pour développer les communs numériques. Nous avons donc choisi de présenter différents modèles d'affaires populaires dans l'écosystème libre à l'annexe A en guise de complément.

Sans être révolutionnaire, nous sommes d'avis que ce travail apporte certaines réponses à la question « Qu'est-ce qui motive une personne à écrire des logiciels libres ? ». Comme abordé à la section 2.3, la prochaine étape serait de bâtir un vrai modèle détaillé d'incitatifs économiques et sociaux.

Malheureusement, le désintérêt des économistes pour les rouages de l'écosystème libre depuis la fin des années 2000 est marqué. Cela est regrettable, car comme nous avons pu le prouver dans ce travail, les données pour traiter de cette question existent et sont de qualité acceptable. Si le sondage annuel de Stack Overflow n'a pas la prétention d'être une étude longitudinale, les questions qui nous intéressent le plus comme le salaire annuel et la participation à un projet libre reviennent années après années. Un-e économiste s'intéressant au libre trouverait ici de quoi s'occuper pour nombreuses années à venir.

ANNEXE A : QUELS MODÈLES D’AFFAIRES POUR FINANCER LES LOGICIELS LIBRES ?

Comme nous l’avons abordé dans le premier chapitre de ce mémoire, l’informatique libre est très hétérogène et répond à des besoins vastes et variés. Ainsi, un logiciel comme le serveur web *Apache* — dont dépend une bonne partie du trafic en ligne — n’a pas la même structure de développement qu’une application libre destinée aux téléphones intelligents, ou encore qu’un script libre utilisé par une personne faisant de l’administration système.

Il existe conséquemment de nombreux modèles d’affaires qui permettent de financer le développement de logiciels libres. Dans cette annexe, nous souhaitons recenser les plus communs et donner quelques exemples pertinents de réussites (et parfois d’échecs) de ces modèles.

Financement académique

Avant l’apparition des premières licences libres, une partie importante des logiciels distribués librement l’étaient par les communautés *hackers*, liées de près ou de loin au milieu académique. Les logiciels écrits par ces communautés sont les précurseurs des logiciels libres que l’on connaît de nos jours.

Le financement académique joue encore un rôle important dans les communautés libres. Si plusieurs universités supportent activement le développement de certains logiciels au niveau institutionnel (dons, contrats de support, contributions explicites par des employé-e-s), une partie non négligeable de l’apport du milieu académique aux logiciels libres provient du travail effectué par des chercheuses, des chercheurs et des professeur-e-s.

En effet, l'indépendance académique et la structure salariale très stable des professeur-e-s leur permet de contribuer plus facilement aux communs libres que d'autres corps de métiers. Il existe de surcroît d'importants rapprochements entre l'éthos scientifique (Merton, 1973) et l'éthos du mouvement libre. Cela s'exprime entre autre par la synergie de ces deux mouvements dans la Science ouverte (Benkler, 2002, p. 381).

Certains langages de programmation comme *OCaml* sont même réputés pour être très « académiques » et ne sont que faiblement utilisés en dehors des laboratoires et centres de recherche universitaires. Sans le financement et le support du milieu académique, la viabilité économique d'importants projets à faible valeur commerciale comme le *Software Heritage* de l'INRIA (Di Cosmo, Zacchiroli, 2017) — un projet d'archive universelle des logiciels libres — serait carrément impossible.

La popularité de certains outils libres dans la communauté académique n'est toutefois pas forcément gage de leur soutenabilité. En effet, plusieurs programmes et bibliothèques (*libraries*) à caractère scientifique très utilisées comme *Matplotlib* ont d'importantes difficultés à lever des fonds académiques pour assurer leur avenir :

On 10 April [2019], astrophysicists announced that they had captured the first ever image of a black hole. This was exhilarating news, but none of the giddy headlines mentioned that the image would have been impossible without open-source software. The image was created using Matplotlib, a Python library for graphing data, as well as other components of the open-source Python ecosystem. Just five days later, the US National Science Foundation (NSF) rejected a grant proposal to support that ecosystem, saying that the software lacked sufficient impact. — (Nowogrodzki, 2019)

Financement gouvernemental

Comme toutes les grandes organisations, les gouvernements ont des besoins technologiques qui peuvent parfois être complexes. De plus — contrairement à certaines entreprises — les États doivent typiquement respecter des protocoles de sécurité très contraignants tout en évitant de se placer à la merci d'entreprises privées. Cela est particulièrement vrai dans l'armée.

Beaucoup de gouvernements se tournent donc vers des solutions libres, car ces dernières ont l'avantage de ne pas être susceptibles à l'enfermement propriétaire (*Vendor Lock-in*), de permettre la mutualisation des coûts de développement avec d'autres organismes publics et de faciliter les audits de sécurité indépendants. C'est par exemple le choix qu'a fait la France, où la Gendarmerie nationale a décidé en 2013 de migrer plus de 30 000 postes informatiques à Linux (Elie, 2019). De manière similaire, le gouvernement de Taïwan a décidé de se tourner vers les logiciels libres pour appuyer ses démarches de renouveau démocratique (O'Flaherty, 2018 ; Tang, 2018).

Des logiciels libres comme le projet Tor, permettant de s'anonymiser en ligne, ou encore *LibreOffice*, une suite de logiciels de bureautique libre rivalisant *Microsoft Office*, sont ainsi partiellement financés par différents organismes gouvernementaux à travers le monde (Open Technology Fund, 2017). Certains pays financent même directement des projets libres à partir de fonds spéciaux, comme le fait l'Allemagne avec son *Prototype Fund*.

L'adoption des logiciels libres dans la fonction publique — et donc leur financement par ces organismes — est cependant liée de très près à la volonté des gouvernements élus. Un changement de gouvernement municipal en 2017 dans la ville de Munich a ainsi engendré la décommission des 13 000 postes de travail Linux au profit de machines Windows (Tomazeli, 2019). Munich était pourtant considérée depuis 2004 comme une pionnière des logiciels libres dans la fonction publique et contribuait activement à certains projets libres.

Face à ces défis, plusieurs groupes de pression organisent des campagnes de mobilisation pour défendre la place des logiciels libres dans les organismes publics. Ainsi, la campagne européenne *Public Money, Public Code*, visant à s'assurer que les logiciels développés avec de l'argent public soient systématiquement publiés sous des licences libres, joue actuellement un rôle important dans la promotion du libre en Europe (Moody, 2019). Fait surprenant, depuis 1895 le travail produit par les agences gouvernementales fédérales aux États-Unis (code informatique, photos, œuvres littéraires, etc.) ne sont pas protégées par le copyright et appartiennent au domaine public (United States Senate, 1961 ; Institute of Electrical and Electronics Engineers, 1977).

Fondations

Parfois, certains projets n'ayant pas de visées commerciales deviennent tellement importants qu'il leur est nécessaire d'engager du personnel pour faire du travail administratif. En général, ces projets établissent une fondation qui s'incorpore comme organisme de bienfaisance, lui permettant ainsi de récolter des dons et d'émettre des reçus déductibles d'impôt. C'est par exemple le cas de la *Document Foundation*, qui maintient la suite *LibreOffice*, la fondation *Apache*, qui maintient entre autres le serveur web *Apache*, et la fondation Linux, qui travaille entre autres sur le noyau Linux.

La plupart du temps, les décisions techniques du projet et les décisions financières de la fondation sont indépendantes. Ainsi, un projet s'assure de sa pérennité financière tout en conservant sa liberté technique. La fondation est alors responsable d'engager des contributrices et des contributeurs à contrat pour travailler sur le projet.

Ce modèle d'affaires repose en très grande partie sur des dons effectués par des entreprises, des gouvernements et des particuliers. Si ces groupes ont souvent à cœur le développement des programmes chapeautés par ces fondations, les entreprises voient également en ces dons une manière d'améliorer leur image de marque et de payer moins d'impôt.

Plusieurs grandes entreprises disposent également de fondations privées qui financent parfois des projets libres. L'impact de ces fondations est cependant relativement limité et ne pourrait pas réellement être considéré comme un modèle d'affaires à proprement parler.

Socio-financement

Phénomène relativement récent — particulièrement dans le monde du libre — le socio-financement se distingue d'autres modèles d'affaires par sa volonté de faire reposer les coûts du développement d'un logiciel sur les individus qui l'utilisent.

Les campagnes de socio-financement fructueuses visent souvent à soutenir une personne travaillant sur de multiples projets libres plutôt qu'un projet en particulier. Les revenus tirés du socio-financement sont typiquement faibles et ont habituellement comme but premier la diversification des sources de revenu d'un développeur ou d'une développeuse.

Si quelques individus ayant une certaine crédibilité dans leurs communautés respectives comme Joey Hess (2018) ou encore Drew DeVault (2019) s'en tirent somme toute relativement bien, ces cas font figures d'exception. Plusieurs, comme François Élie (2019), considèrent cependant le socio-financement comme une source de financement peu efficace et parfois même dommageable, préférant plutôt que les coûts de développement des logiciels libres reposent sur des institutions et non sur des individus.

Licences multiples

Comme nous l'avons vu dans l'introduction de ce mémoire, ce qui différencie le code libre du code propriétaire est sa licence de propriété intellectuelle. Il arrive cependant qu'un programme soit simultanément publié sous une licence libre et une licence propriétaire. On parle alors de projet à licences multiples (*Multi-licensing*).

La licence d'un projet libre peut parfois être considérée comme un frein à son adoption par une grande entreprise pour des motifs légaux, les services de conseils juridiques étant reconnus pour leur faible enthousiasme envers les logiciels libres. D'aucuns souhaiteraient également intégrer à des projets propriétaires du code libre normalement publié sous une licence restrictive comme la GNU GPL.

Utiliser de multiples licences permet donc à ces projets de vendre des licences propriétaires à des prix leur permettant de financer leurs activités. En général, les projets ayant des licences multiples ont des licences principales très restrictives pour inciter les entreprises à acheter une licence propriétaire. C'est par exemple le cas pour l'infrastructure de développement *Qt* ou encore pour le serveur de téléphonie numérique *Asterisk*.

Vente de support

La vente de support commercial pour financer un projet libre est un modèle populaire et ayant fait ses preuves, comme peut en témoigner l'énorme succès de Red Hat (Levine, 2014). Certaines entreprises sont attirées par les différentes vertus des logiciels libres (dont le fait d'éviter l'enfermement propriétaire), mais n'ont pas forcément les ressources à l'interne pour s'assurer du bon fonctionnement du logiciel en question. D'autres préfèrent tout simplement contracter de l'aide extérieure et se concentrer sur leur mission principale.

Ces entreprises font donc affaire avec un partenaire externe offrant du support commercial pour le logiciel qu'elles décident d'utiliser. Souvent, ce partenaire est l'entité principale qui développe le logiciel. Néanmoins, la nature ouverte des logiciels libres fait en sorte qu'il existe parfois plusieurs entités commerciales collaborant au développement du même logiciel.

Ce type de modèle fonctionne relativement bien car il constitue un modèle d'affaires clair, qui permet de monter un plan d'affaires traditionnel. De plus, comme ce type d'offre existe déjà pour de nombreux logiciels propriétaires, les entreprises sont généralement à l'aise avec ce genre de dépenses.

Certains affirment toutefois qu'il peut être difficile de convaincre des entreprises de payer pour du support lorsqu'un logiciel leur est offert à un coût monétaire nul (Janke, 2010a, 2010b, 2010c ; Akhmechet, 2017).

Logiciel en tant que service

Grâce à l'émergence de plateformes d'infonuagique (*Cloud Computing*) compétitives, robustes et peu onéreuses, la vente de logiciels en tant que service (*Software as a Service* — SaaS) est un modèle d'affaires de plus en plus populaire. Poussant le concept de vente de support un cran plus loin, ce modèle d'affaires propose aux entreprises d'héberger et de gérer un logiciel libre pour elles, en échange d'un montant mensuel ou annuel.

Historiquement, ce modèle était réservé aux grands projets libres bien établis : pour être en mesure d'offrir un produit en SaaS compétitif dans les années 2000, il était nécessaire d'acheter et de maintenir des serveur physiques dans un centre de données — une entreprise complexe, coûteuse et risquée.

Avec l'arrivée d'offres de location de ressources en centre de données très compétitives (*Amazon Web Services, Microsoft Azure, Google Cloud Platform, etc.*), les entreprises souhaitant offrir du SaaS n'ont plus besoin de faire des investissements en capitaux importants et peuvent moduler leurs coûts fixes en fonction de la demande, parfois de manière très agressive.

Comme pour la vente de support, le SaaS n'est pas un modèle d'affaires unique aux logiciels libres et est également très populaire dans le monde des logiciels propriétaires, créant ainsi un sentiment de familiarité pour les entreprises qui décident de retenir ce genre de service.

Plusieurs projets de technologies web libres comme *Wordpress* (Mullenweg, 2014) tirent une majorité de leurs revenus du SaaS.

Travail bénévole

Le travail bénévole est de loin le « modèle d'affaires » le plus populaire pour le développement de logiciels libres. L'écrasante majorité des projets libres reposent d'une manière ou d'une autre sur une certaine proportion de travail non rémunéré, que ce soit par la contribution directe de code ou encore par l'écriture de rapports de bogues.

Chasseurs de prime

Populaire surtout dans le milieu de la sécurité informatique, certaines grandes organisations offrent des primes (parfois importantes) pour la découverte de problèmes de sécurité majeurs pour certains logiciels libres jugés critiques (Reda, 2019).

Il est rare qu'un projet ne survive que par les gains faits à travers des primes de ce genre, mais certaines personnes sont en mesure de financer leur travail sur des projets libres en récoltant ces primes.

Open Core

Le modèle *Open Core* consiste à financer le développement d'un logiciel libre (le *core*) en vendant des fonctionnalités propriétaires additionnelles. Souvent ces fonctionnalités sont nécessaires pour déployer le logiciel en entreprise. Historiquement très répandu dans la communauté libre, ce modèle a perdu en popularité au fil des années au profit de modèles jugés « plus modernes » (Dix, 2017), comme le SaaS. Malgré tout, de nombreux géants du libre, comme Gitlab, Puppet ou encore ElasticSearch, utilisent ce modèle d'affaires avec beaucoup de succès.

De nombreuses personnes soutiennent cependant que le modèle *Open Core* est contraire à l'ethos des logiciels libres (Prentice, 2010 ; Kuhn, 2009 ; Phipps, 2010). Ces personnes affirment en effet que financer le développement de ces logiciels grâce à l'érection de barrières artificielles visant à restreindre l'accès à certaines parties du code informatique n'est pas conséquent avec les valeurs à la base du mouvement libre.

Mutualisation

Modèle plus large englobant souvent de nombreuses sources de financement diverses, la mutualisation permet à différentes entités de se regrouper et de diviser les coûts de maintenance et de développement d'un projet.

Plusieurs types d'entités utilisent la mutualisation comme moyen de financer des projets complexes. Par exemple, le projet *Civil Infrastructure Platform* regroupe des partenaires comme Siemens, Toshiba et Renesas Electronics dans le but de supporter la maintenance de versions spécifiques du noyau Linux sur le très long terme³ pour assurer la pérennité d'infrastructures civiles critiques dans le secteur du transport et de la production d'énergie (Kobayashi, 2017).

En France, la mutualisation permet à de nombreuses municipalités de petite taille de s'associer pour développer des logiciels adaptés à leurs réalités. Fondée en 2002, l'association des développeurs et utilisateurs de logiciels libres pour les administrations et les collectivités territoriales (ADULACT) offre ainsi à ses membres un marché commun ainsi qu'une forge logicielle (Elie, 2019).

Au Canada la seule initiative majeure de mutualisation des infrastructures informatiques libres est le projet *Open Resource Exchange* — géré par le gouvernement fédéral — et qui vise à mettre en lien différents acteurs gouvernementaux (agences fédérales, provinces et municipalités) à travers le pays pour développer et maintenir collectivement des solutions informatiques partagées (Tomazeli, 2019).

3. Les projets d'ingénierie civile dont les membres de la *Civil Infrastructure Platform* ont la charge ont une durée de vie typique allant de 25 à 50 ans — une véritable éternité en informatique.

ANNEXE B : TABLEAUX STATISTIQUES

TABLEAU 3.3 – Statistiques descriptives des variables dichotomiques agrégées

	Q1	Médiane	Q3	Moyenne	Écart type
Belonging	0.50	0.75	0.75	0.69	0.22
Altruism	0.61	0.78	0.89	0.72	0.19
Satisfaction	0.50	0.75	0.83	0.69	0.25
OpenSourceComposite	0.11	0.44	0.56	0.39	0.26

TABLEAU 3.4 – Estimation de l'impact de la participation à un projet libre sur le salaire

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	-259 611.30	11 786.13	-22.03	0.00
OpenSource	9 101.46	2 116.98	4.30	0.00
CountryHDI	377 857.15	12 086.24	31.26	0.00
Age0017	-37 669.50	8 052.75	-4.68	0.00
Age1824	-18 091.75	2 712.45	-6.67	0.00
Age3544	20 894.35	2 847.67	7.34	0.00
Age4554	46 558.46	5 715.04	8.15	0.00
Age5564	78 491.38	13 885.52	5.65	0.00
Age65More	29 295.16	35 556.22	0.82	0.41
GenderDummy	7 511.60	3 747.17	2.00	0.05
RaceEthnicityDummy	16 302.09	3 091.48	5.27	0.00
FormalEducationDummy	9 460.22	2 453.95	3.86	0.00
EducationParentsDummy	3 412.34	2 065.93	1.65	0.10
EmploymentPartSelf	-15 091.55	2 762.70	-5.46	0.00
EmploymentOther	-38 184.74	3 979.54	-9.60	0.00
More8YearsCoding	9 613.97	2 397.56	4.01	0.00
Hobby	-866.28	2 731.89	-0.32	0.75

Nombre d'observations : 39 149

TABLEAU 3.5 – Estimation de l’impact du degré de participation à la communauté libre sur le salaire

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	-264 950.97	14 358.66	-18.45	0.00
OpenSourceComposite	7 862.50	4 453.93	1.77	0.08
CountryHDI	378 097.94	14 477.93	26.12	0.00
Age0017	-49 171.87	7 159.88	-6.87	0.00
Age1824	-19 565.58	3 123.77	-6.26	0.00
Age3544	21 437.04	3 301.77	6.49	0.00
Age4554	40 457.42	6 158.49	6.57	0.00
Age5564	97 368.68	18 883.93	5.16	0.00
Age65More	43 279.52	51 941.60	0.83	0.40
GenderDummy	8 862.96	4 478.50	1.98	0.05
RaceEthnicityDummy	16 234.29	3 667.02	4.43	0.00
FormalEducationDummy	11 950.30	2 752.48	4.34	0.00
EducationParentsDummy	4 197.52	2 398.25	1.75	0.08
EmploymentPartSelf	-12 866.59	3 096.47	-4.16	0.00
EmploymentOther	-35 474.44	5 042.76	-7.03	0.00
More8YearsCoding	9 775.98	2 757.30	3.55	0.00
Hobby	3 635.35	3 157.83	1.15	0.25

Nombre d’observations : 29 179

TABLEAU 3.6 – Estimation de l’impact de la participation à un projet libre sur le plaisir d’écrire du code informatique

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.54	0.03	20.52	0.00
OpenSource	0.17	0.00	46.34	0.00
CountryHDI	0.14	0.03	5.27	0.00
Age0017	0.10	0.01	7.25	0.00
Age1824	0.04	0.01	7.99	0.00
Age3544	-0.05	0.01	-8.34	0.00
Age4554	-0.08	0.01	-7.92	0.00
Age5564	-0.15	0.02	-7.26	0.00
Age65More	-0.11	0.06	-1.90	0.06
GenderDummy	0.12	0.01	14.04	0.00
RaceEthnicityDummy	-0.02	0.01	-3.46	0.00
FormalEducationDummy	-0.04	0.00	-8.38	0.00
EducationParentsDummy	-0.00	0.00	-0.31	0.75
EmploymentPartSelf	-0.00	0.01	-0.48	0.63
EmploymentOther	0.03	0.01	2.82	0.00
More8YearsCoding	0.01	0.00	2.13	0.03
ConvertedSalary	-0.00	0.00	-0.32	0.75

Nombre d’observations : 39 149

TABLEAU 3.7 – Estimation de l’impact du degré de participation à la communauté libre sur le plaisir d’écrire du code informatique

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.51	0.03	17.04	0.00
OpenSourceComposite	0.23	0.01	29.97	0.00
CountryHDI	0.19	0.03	6.43	0.00
Age0017	0.08	0.02	4.77	0.00
Age1824	0.04	0.01	6.50	0.00
Age3544	-0.04	0.01	-7.32	0.00
Age4554	-0.08	0.01	-7.02	0.00
Age5564	-0.16	0.02	-6.43	0.00
Age65More	-0.07	0.07	-1.09	0.28
GenderDummy	0.11	0.01	10.57	0.00
RaceEthnicityDummy	-0.02	0.01	-2.55	0.01
FormalEducationDummy	-0.03	0.00	-6.54	0.00
EducationParentsDummy	-0.00	0.00	-0.22	0.82
EmploymentPartSelf	-0.01	0.01	-1.33	0.18
EmploymentOther	0.01	0.01	0.79	0.43
More8YearsCoding	0.01	0.01	1.74	0.08
ConvertedSalary	0.00	0.00	1.15	0.25

Nombre d’observations : 29 179

TABLEAU 3.8 – Estimation de l’impact de la participation à un projet libre sur le sentiment d’appartenance à la communauté informatique

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.62	0.01	42.12	0.00
OpenSource	0.03	0.00	13.50	0.00
CountryHDI	0.02	0.01	1.38	0.17
Age0017	0.04	0.01	2.65	0.01
Age1824	0.00	0.00	1.49	0.14
Age3544	0.00	0.00	0.39	0.70
Age4554	0.02	0.00	3.66	0.00
Age5564	0.03	0.01	3.05	0.00
Age65More	0.03	0.03	1.22	0.22
GenderDummy	0.01	0.00	2.40	0.02
RaceEthnicityDummy	0.00	0.00	1.08	0.28
FormalEducationDummy	0.00	0.00	1.84	0.07
EducationParentsDummy	0.01	0.00	4.57	0.00
EmploymentPartSelf	-0.01	0.00	-3.49	0.00
EmploymentOther	-0.03	0.01	-4.90	0.00
More8YearsCoding	0.00	0.00	1.53	0.13
ConvertedSalary	0.00	0.00	1.20	0.23
Hobby	0.04	0.00	15.40	0.00

Nombre d’observations : 38 978

TABLEAU 3.9 – Estimation de l’impact du degré de participation à la communauté libre sur le sentiment d’appartenance à la communauté informatique

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.62	0.02	36.74	0.00
OpenSourceComposite	0.04	0.00	8.10	0.00
CountryHDI	0.02	0.02	1.34	0.18
Age0017	0.03	0.02	1.80	0.07
Age1824	0.00	0.00	1.35	0.18
Age3544	0.00	0.00	0.13	0.90
Age4554	0.02	0.01	2.96	0.00
Age5564	0.04	0.01	4.00	0.00
Age65More	0.02	0.04	0.61	0.54
GenderDummy	0.01	0.01	2.00	0.05
RaceEthnicityDummy	0.00	0.00	0.83	0.41
FormalEducationDummy	0.01	0.00	2.36	0.02
EducationParentsDummy	0.01	0.00	3.94	0.00
EmploymentPartSelf	-0.01	0.00	-3.35	0.00
EmploymentOther	-0.03	0.01	-3.42	0.00
More8YearsCoding	0.00	0.00	1.09	0.28
ConvertedSalary	0.00	0.00	1.65	0.10
Hobby	0.05	0.00	13.66	0.00

Nombre d’observations : 29 058

TABLEAU 3.10 – Estimation de l’impact de la participation à un projet libre sur l’altruisme

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.53	0.01	41.86	0.00
OpenSource	0.02	0.00	12.36	0.00
CountryHDI	0.22	0.01	17.05	0.00
Age0017	-0.03	0.01	-2.21	0.03
Age1824	-0.02	0.00	-7.95	0.00
Age3544	0.02	0.00	10.06	0.00
Age4554	0.05	0.00	12.35	0.00
Age5564	0.06	0.01	7.48	0.00
Age65More	0.07	0.03	2.52	0.01
GenderDummy	-0.04	0.00	-12.53	0.00
RaceEthnicityDummy	0.01	0.00	4.47	0.00
FormalEducationDummy	0.01	0.00	5.38	0.00
EducationParentsDummy	0.01	0.00	5.41	0.00
EmploymentPartSelf	-0.00	0.00	-0.78	0.44
EmploymentOther	0.00	0.01	0.11	0.92
More8YearsCoding	0.01	0.00	5.24	0.00
ConvertedSalary	0.00	0.00	2.19	0.03
Hobby	0.01	0.00	4.40	0.00

Nombre d’observations : 38 880

TABLEAU 3.11 – Estimation de l’impact du degré de participation à la communauté libre sur l’altruisme

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.53	0.01	36.21	0.00
OpenSourceComposite	0.04	0.00	10.89	0.00
CountryHDI	0.21	0.01	14.22	0.00
Age0017	-0.02	0.02	-1.07	0.29
Age1824	-0.02	0.00	-6.21	0.00
Age3544	0.02	0.00	8.81	0.00
Age4554	0.05	0.00	11.46	0.00
Age5564	0.06	0.01	7.10	0.00
Age65More	0.06	0.04	1.60	0.11
GenderDummy	-0.04	0.00	-9.39	0.00
RaceEthnicityDummy	0.01	0.00	4.39	0.00
FormalEducationDummy	0.01	0.00	4.61	0.00
EducationParentsDummy	0.01	0.00	4.56	0.00
EmploymentPartSelf	-0.00	0.00	-0.01	0.99
EmploymentOther	-0.00	0.01	-0.33	0.74
More8YearsCoding	0.01	0.00	4.64	0.00
ConvertedSalary	0.00	0.00	1.80	0.07
Hobby	0.01	0.00	3.04	0.00

Nombre d’observations : 29 012

TABLEAU 3.12 – Estimation de l'impact de la participation à un projet libre sur la satisfaction au travail

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.48	0.02	27.97	0.00
OpenSource	0.03	0.00	10.90	0.00
CountryHDI	0.21	0.02	12.22	0.00
Age0017	0.05	0.03	1.92	0.06
Age1824	0.02	0.00	5.34	0.00
Age3544	-0.01	0.00	-1.73	0.08
Age4554	-0.02	0.01	-3.44	0.00
Age5564	0.00	0.01	0.21	0.83
Age65More	0.06	0.03	1.98	0.05
GenderDummy	-0.00	0.00	-0.59	0.56
RaceEthnicityDummy	-0.03	0.00	-9.17	0.00
FormalEducationDummy	0.01	0.00	2.45	0.01
EducationParentsDummy	0.02	0.00	6.51	0.00
EmploymentPartSelf	0.00	0.00	1.03	0.30
More8YearsCoding	0.01	0.00	2.85	0.00
ConvertedSalary	0.00	0.00	2.10	0.04
Hobby	0.01	0.00	3.82	0.00

Nombre d'observations : 37 400

TABLEAU 3.13 – Estimation de l'impact du de degré de participation à la communauté libre sur la satisfaction au travail

	Coef. estimé	Écart type	Statistique t	valeur-p
(Intercept)	0.47	0.02	24.03	0.00
OpenSourceComposite	0.03	0.01	6.19	0.00
CountryHDI	0.21	0.02	10.58	0.00
Age0017	0.05	0.03	1.43	0.15
Age1824	0.02	0.00	4.50	0.00
Age3544	-0.00	0.00	-0.30	0.76
Age4554	-0.02	0.01	-2.85	0.00
Age5564	0.00	0.01	0.09	0.93
Age65More	0.10	0.04	2.75	0.01
GenderDummy	0.00	0.01	0.67	0.51
RaceEthnicityDummy	-0.03	0.00	-7.21	0.00
FormalEducationDummy	0.01	0.00	1.92	0.05
EducationParentsDummy	0.02	0.00	5.59	0.00
EmploymentPartSelf	0.01	0.00	1.27	0.21
More8YearsCoding	0.01	0.00	3.05	0.00
ConvertedSalary	0.00	0.00	2.26	0.02
Hobby	0.02	0.00	4.71	0.00

Nombre d'observations : 28 136

BIBLIOGRAPHIE

AKHMECHET, Slava, 2017. *RethinkDB : why we failed* [en ligne]. janvier 2017. Disponible à l'adresse : <https://web.archive.org/web/20170721173319/http://www.defmacro.org/2017/01/18/why-rethinkdb-failed.html>.

ALEXA INTERNET, INC., 2019. *stackoverflow.com Competitive Analysis, Marketing Mix and Traffic* [en ligne]. juin 2019. Disponible à l'adresse : <https://www.alexa.com/siteinfo/stackoverflow.com>.

ALLBERY, Russ et WHITTON, Sean, 2019. *Debian Policy Manual version 4.4.1.1* [en ligne]. octobre 2019. Disponible à l'adresse : <https://www.debian.org/doc/debian-policy/>.

AMADEO, Ron, 2020. Android 10 has the fastest update rate ever, hits 16% of users in 10 months. In : *Ars Technica* [en ligne]. juillet 2020. Disponible à l'adresse : <https://arstechnica.com/gadgets/2020/07/android-10-has-the-fastest-update-rate-ever-hits-16-of-users-in-10-months/>.

ANGRIST, Joshua et KRUEGER, Alan, 1999. Empirical Strategies in Labor Economics. In : *Handbook of Labor Economics, Volume 3* [en ligne]. Elsevier Science. 1999. p. 1277—1364. Disponible à l'adresse : https://uh.edu/~adkugler/Angrist/&Krueger_HLE.pdf.

APRIL, 2007. *Les modèles économiques du Logiciel Libre* [en ligne]. April. Disponible à l'adresse : <https://www.april.org/files/documents/200712-modeles-economiques.pdf>.

ASAY, Matt, 2017. Open source innovation is now all about vendor on-ramps. In : *InfoWorld* [en ligne]. novembre 2017. Disponible à l'adresse : <https://www.infoworld.com/article/3238491/open-source-tools/open-source-innovation-is-now-all-about-vendor-on-ramps.html>.

ASAY, Matt, 2018. Microsoft may be the world's largest open source contributor, but developers don't care, yet. In : *TechRepublic* [en ligne]. novembre 2018. Disponible à l'adresse : <https://www.techrepublic.com/article/microsoft-may-be-the-worlds-largest-open-source-contributor-but-developers-dont-yet-care/>.

BEAUPRÉ, Antoine, 2016a. Django debates user tracking. In : *LWN.net* [en ligne]. novembre 2016. Disponible à l'adresse : <https://lwn.net/Articles/707443/>.

BEAUPRÉ, Antoine, 2016b. *My free software activities, October 2016* [en ligne]. octobre 2016. Disponible à l'adresse : <https://anarc.at/blog/2016-10-31-free-software-activities-october-2016/>.

BENKLER, Yochai, 2002. Coase's Penguin, or, Linux and The Nature of the Firm. In : *The Yale Law Journal* [en ligne]. 2002. Vol. 112, n° 3, p. 371—399. Disponible à l'adresse : https://www.yalelawjournal.org/pdf/354_t5aih5i1.pdf.

BOTTOMLEY, James, 2018. *GPL as the best licence – Community, Code and Licensing* [en ligne]. janvier 2018. Disponible à l'adresse : <https://blog.hansenpartnership.com/gpl-as-the-best-licence-community-code-and-licensing/>.

BOYER, Marcel et ROBERT, Jacques, 2006. *The Economics of Free and Open Source Software : Contributions to a Government Policy on Open Source Software* [en ligne]. CIRANO. Disponible à l'adresse : <http://www.cirano.qc.ca/pdf/publication/2006RP-03.pdf>.

BRESSERS, Josh, 2019. *Misguided misguidings over the EU bug bounty* [en ligne]. janvier 2019. Disponible à l'adresse : <https://opensourcsecurity.io/2018/12/30/misguided-misguidings-over-the-eu-bug-bounty/>.

BRIGHT, Peter, 2018. Developers love trendy new languages but earn more with functional programming. In : *Ars Technica* [en ligne]. mars 2018. Disponible à l'adresse : <https://arstechnica.com/gadgets/2018/03/developers-love-trendy-new-languages-but-earn-more-with-functional-programming/>.

BRIGHT, Peter, 2019. Developers love Python and TypeScript, get paid for Clojure, and aren't using blockchain. In : *Ars Technica* [en ligne]. avril 2019. Disponible à l'adresse : <https://arstechnica.com/gadgets/2019/04/developers-love-python-and-typescript-get-paid-for-clojure-and-arent-using-blockchain/>.

BROCA, Sébastien, 2013. *Utopie du logiciel libre, du bricolage informatique à la réinvention sociale*. Neuvy-en-Champagne : Le Passager clandestin.

CANNON, Brett, 2016. *Why I took October off from OSS volunteering* [en ligne]. novembre 2016. Disponible à l'adresse : <https://snarky.ca/why-i-took-october-off-from-oss-volunteering/>.

CBC NEWS, 2014. Stephen Arthuro Solis-Reyes charged in Heartbleed-related SIN theft. In : *CBC News* [en ligne]. avril 2014. Disponible à l'adresse : <https://www.cbc.ca/news/politics/stephen-arthuro-solis-reyes-charged-in-heartbleed-related-sin-theft-1.2612526>.

CIGREF, 2008. *L'Open source en entreprise* [en ligne]. Cigref. Disponible à l'adresse : <https://www.cigref.fr/lopen-source-en-entreprise>.

CIGREF, 2011. *Maturité et gouvernance de l'Open Source : la vision des grandes entreprises* [en ligne]. Cigref. Disponible à l'adresse : <https://www.cigref.fr/maturite-et-gouvernance-de-lopen-source-la-vision-des-grandes-entreprises>.

CIGREF, 2017. *L'open innovation, une réponse aux challenges de l'entreprise* [en ligne]. Cigref. Disponible à l'adresse : <https://www.cigref.fr/etude-innovation-cigref-2017-1-open-innovation>.

CIGREF, 2018a. *Open source : une alternative aux grands fournisseurs du numérique* [en ligne]. Cigref. Disponible à l'adresse : <https://www.cigref.fr/publication-open-source-une-alternative-aux-grands-fournisseurs-du-numerique>.

CIGREF, 2018b. *Un modèle d'analyse et de benchmarking des coûts informatiques pour piloter sa performance économique* [en ligne]. Cigref. Disponible à l'adresse : <https://www.cigref.fr/benchmarking-analyse-couts-informatiques-modele-piloter-performance-economique-it-mise-a-jour-2018>.

CLABURN, Thomas, 2018a. Fed up with cloud giants ripping off its database, MongoDB forks new 'open-source license'. In : *The Register* [en ligne]. octobre 2018. Disponible à l'adresse : https://www.theregister.co.uk/2018/10/16/mongodb_licensning_change/.

CLABURN, Thomas, 2018b. Redis has a license to kill : Open-source database maker takes some code proprietary. In : *The Register* [en ligne]. août 2018. Disponible à l'adresse : https://www.theregister.co.uk/2018/08/23/redis_database_license_change/.

COASE, Ronald H., 1937. The Nature of the Firm. In : *Economica* [en ligne]. 1937. Vol. 4, n° 16, p. 386—405. Disponible à l'adresse : <http://www.jstor.org/stable/2626876>.

COLEMAN, E. Gabriella, 2013. *Coding Freedom : The Ethics and Aesthetics of Hacking*. Princeton : Princeton University Press.

- COLEMAN, Gabriella, 2017. From Internet Farming to Weapons of the Geek. In : *Current Anthropology* [en ligne]. 2017. Vol. 58, n° S15, p. S91—S102. Disponible à l'adresse : <https://www.journals.uchicago.edu/action/showCitFormats?doi=10.1086/688697>.
- CORBET, Jonathan, 2016. Funding Qubes OS. In : *LWN.net* [en ligne]. novembre 2016. Disponible à l'adresse : <https://lwn.net/Articles/707713/>.
- CORBET, Jonathan, 2018. Protecting the open-source license commons. In : *LWN.net* [en ligne]. novembre 2018. Disponible à l'adresse : <https://lwn.net/Articles/769982/>.
- CORBET, Jonathan et KROAH-HARTMAN, Greg, 2017. *2017 Linux Kernel Development Report* [en ligne]. The Linux Foundation. Disponible à l'adresse : <https://www.linuxfoundation.org/2017-linux-kernel-report-landing-page/>.
- DAFFARA, Carlo, 2007. *Business models in FLOSS-based companies* [en ligne]. Disponible à l'adresse : <https://flosshub.org/sites/flosshub.org/files/OSSEMP07-daffara.pdf>.
- DARCHE, Marc-Aurèle et OTHERS, 2013. *Modèles économiques liés aux logiciels libres* [en ligne]. Disponible à l'adresse : <https://aful.org/media/document/modeles-economiques.pdf>.
- DARVELL, James, 2016. Open-Source Project Secretly Funded by CIA. In : *Linux Journal* [en ligne]. avril 2016. Disponible à l'adresse : <https://www.linuxjournal.com/content/open-source-project-secretly-funded-cia>.
- DE BLANC, Molly, ONEIL, Mathieu, RAISSI, Mahin et ZACCHIROLI, Stefano, 2017. *Debian Contributors Survey 2016 — Preliminary Analysis* [en ligne]. Disponible à l'adresse : <https://upsilon.cc/debian-survey-2016/preliminary-analysis.html>.
- DEVVAULT, Drew, 2019. *I'm going to work full-time on free software* [en ligne]. janvier 2019. Disponible à l'adresse : <https://drewdevault.com/2019/01/15/Im-doing-FOSS-full-time.html>.
- DIBONA, Chris, OCKMAN, Sam et STONE, Mark, 2008. Introduction. In : *Open Sources : Voices from the Open Source Revolution*. Sebastopol : O'Reilly Media.
- DI COSMO, Roberto et ZACCHIROLI, Stefano, 2017. Software Heritage : Why and How to Preserve Software Source Code. In : *Proceedings of the 14th International Conference on Digital Preservation* [en ligne]. New York : ACM. 2017. Disponible à l'adresse : <https://ipres2017.jp/wp-content/uploads/19Roberto-Di-Cosmo.pdf>.

DINGLEDINE, Roger, 2018. *Transparency, Openness, and Our 2016 and 2017 Financials* [en ligne]. décembre 2018. Disponible à l'adresse : <https://blog.torproject.org/transparency-openness-and-our-2016-and-2017-financials>.

DIX, Paul, 2017. *The Open Source Business Model is Under Siege* [en ligne]. mai 2017. Disponible à l'adresse : <https://www.influxdata.com/blog/the-open-source-database-business-model-is-under-siege/>.

DIX, Paul, 2018. *It's Time for the Open Source Community to Get Real* [en ligne]. août 2018. Disponible à l'adresse : <https://www.influxdata.com/blog/its-time-for-the-open-source-community-to-get-real/>.

DRYDEN, Ashe, 2013. *The Ethics of Unpaid Labor and the OSS Community* [en ligne]. novembre 2013. Disponible à l'adresse : <https://www.ashedryden.com/blog/the-ethics-of-unpaid-labor-and-the-oss-community>.

EDGE, Jake, 2017. Breaking Libgcrypt RSA via a side channel. In : *LWN.net* [en ligne]. juillet 2017. Disponible à l'adresse : <https://lwn.net/Articles/727179/>.

EDGE, Jake, 2019a. Defining 'sustainable' for an open-source project. In : *LWN.net* [en ligne]. mars 2019. Disponible à l'adresse : <https://lwn.net/SubscriberLink/783169/910bdd575799ed63/>.

EDGE, Jake, 2019b. The state of the OSU Open Source Lab. In : *LWN.net* [en ligne]. mars 2019. Disponible à l'adresse : <https://lwn.net/SubscriberLink/783580/b19af29e210fc01c/>.

EGHBAL, Nadia, 2016a. *Roads and Bridges : The Unseen Labor Behind Our Digital Infrastructure* [en ligne]. Disponible à l'adresse : <https://www.fordfoundation.org/about/library/reports-and-studies/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure/>.

EGHBAL, Nadia, 2017. *Consider the maintainer - LCA 2017 keynote* [en ligne]. Disponible à l'adresse : <https://www.youtube.com/watch?v=W2AR1owgOao>.

ELIE, François, 2009. *Économie du logiciel libre*. Paris : Eyrolles.

ELIE, François, 2019. *Logiciel libre et le secteur public* [en ligne]. Disponible à l'adresse : <https://agendadulibre.qc.ca/events/2017>.

EVÄKALLIO, Jani, 2019. *Paying Cold, Hard Cash for Open Source Contributions* [en ligne]. mai 2019. Disponible à l'adresse : <https://formidable.com/blog/2019/sauce-program/>.

FINK, Martin, 2015. *Martin Fink on Open Source Community Collaboration at Linux-Con Europe 2015* [en ligne]. Disponible à l'adresse : <https://www.youtube.com/watch?v=zxIEDNyZ0kA>.

FORD, Danae, HARKINS, Alisse et PARNIN, Chris, 2017. Someone like me : How does peer parity influence participation of women on stack overflow? In : *2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)* [en ligne]. Piscataway : IEEE. 2017. p. 239—243. Disponible à l'adresse : <https://ieeexplore.ieee.org/abstract/document/8103473>.

FORD, Danae, SMITH, Justin, GUO, Philip J. et PARNIN, Chris, 2016. Paradise Unplugged : Identifying Barriers for Female Participation on Stack Overflow. In : *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering* [en ligne]. New York : ACM. 2016. p. 846—857. Disponible à l'adresse : <https://dl.acm.org/citation.cfm?id=2950331>.

FREE SOFTWARE FOUNDATION, 2001. *What is free software ?* [en ligne]. février 2001. Disponible à l'adresse : <https://www.gnu.org/philosophy/free-sw.html>.

GARRETT, Matthew, 2018. *The Commons Clause doesn't help the commons* [en ligne]. septembre 2018. Disponible à l'adresse : <https://mjg59.dreamwidth.org/51177.html>.

GOODIN, Dan, 2018. Widely used open source software contained bitcoin-stealing backdoor. In : *Ars Technica* [en ligne]. novembre 2018. Disponible à l'adresse : <https://arstechnica.com/information-technology/2018/11/hacker-backdoors-widely-used-open-source-software-to-steal-bitcoin/>.

GORDON-MCKEON, Shauna, 2019. *Governing the software commons*.

HARDIN, Garrett, 1968. The Tragedy of the Commons. In : *Science* [en ligne]. décembre 1968. Vol. 162, n° 3859, p. 1243—1248. Disponible à l'adresse : https://pages.mtu.edu/~asmayer/rural_sustain/governance/Hardin%201968.pdf.

HECKMAN, James, LOCHNER, Lance et TODD, Petra, 2003. Fifty Years of Mincer Earnings Regressions. In : *NBER Working Paper* [en ligne]. 2003. n° 9732. Disponible à l'adresse : <https://doi.org/10.3386/w9732>.

HESS, Joey, 2016. *Patreon* [en ligne]. août 2016. Disponible à l'adresse : <https://joeyh.name/blog/entry/Patreon/>.

HESS, Joey, 2018. *git-annex and funding update* [en ligne]. décembre 2018. Disponible à l'adresse : https://joeyh.name/blog/entry/git-annex_and_funding_update/.

HILL, Benjamin Mako, 2018. *How markets coopted free software's most powerful weapon* [en ligne]. Disponible à l'adresse : <https://mako.cc/copyrighteous/libreplanet-2018-keynote>.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, 1977. Works of the U.S. government. In : *IEEE Transactions on Professional Communication* [en ligne]. novembre 1977. Vol. PC-20, n° 3, p. 188—189. Disponible à l'adresse : <https://ieeexplore.ieee.org/document/6591616>.

IRVING, Alanna, 2017. *Preact : Shattering the Perception that Open Source Must be Free* [en ligne]. mars 2017. Disponible à l'adresse : <https://medium.com/open-collective/why-and-how-to-fund-the-open-source-projects-you-depend-on-da62a582307>.

JANKE, Jorg, 2010a. *Compiere Open Source failed ?* [en ligne]. juin 2010. Disponible à l'adresse : <https://www.compieresource.com/2010/06/compiere-open-source-failed.html>.

JANKE, Jorg, 2010b. *Open Source Business Models (for Compiere) (Compiere from the Source)* [en ligne]. juillet 2010. Disponible à l'adresse : <https://www.compieresource.com/2010/07/open-source-business-models.html>.

JANKE, Jorg, 2010c. *Open Source Contributions (Compiere ERP) (Compiere from the Source)* [en ligne]. juillet 2010. Disponible à l'adresse : <https://www.compieresource.com/2010/07/open-source-contributions.html>.

JULLIEN, Nicolas et ZIMMERMANN, Jean-Benoît, 2011. FLOSS in an Industrial Economics Perspective. In : *Revue d'économie industrielle* [en ligne]. décembre 2011. n° 136, p. 39—64. Disponible à l'adresse : <https://journals.openedition.org/rei/pdf/5177>.

KOBAYASHI, Yoshitake, 2017. *Debian on Civil Infrastructure Systems* [en ligne]. août 2017. Disponible à l'adresse : <https://debconf17.debconf.org/talks/101/>.

KUHN, Bradley M., 2009. « *Open Core* » *Is the New Shareware* [en ligne]. octobre 2009. Disponible à l'adresse : <http://ebb.org/bkuhn/blog/2009/10/16/open-core-shareware.html>.

KUHN, Bradley M., 2016. *Very long wait for funds transfer* [en ligne]. décembre 2016. Disponible à l'adresse : <http://lists.spi-inc.org/pipermail/spi-general/2016-December/003659.html>.

LARDINOIS, Frederic, 2019. AWS gives open source the middle finger. In : *TechCrunch* [en ligne]. janvier 2019. Disponible à l'adresse : <https://techcrunch.com/2019/01/09/aws-gives-open-source-the-middle-finger/>.

LAUTER, Kristin, 2018. *The Microsoft Simple Encrypted Arithmetic Library goes open source* [en ligne]. décembre 2018. Disponible à l'adresse : <https://www.microsoft.com/en-us/research/blog/the-microsoft-simple-encrypted-arithmetic-library-goes-open-source/>.

LEECH, David P., FERRIS, Stacey et SCOTT, John T., 2018. *The Economic Impacts of the Advanced Encryption Standards, 1996-2017* [en ligne]. Disponible à l'adresse : <https://nvlpubs.nist.gov/nistpubs/gcr/2018/NIST.GCR.18-017.pdf>.

LERNER, Josh et TIROLE, Jean, 2001. The open source movement : Key research questions. In : *European Economic Review* [en ligne]. mai 2001. Vol. 45, n° 4—6, p. 819—826. Disponible à l'adresse : <https://www.sciencedirect.com/science/article/pii/S0014292101001246>.

LERNER, Josh et TIROLE, Jean, 2002. Some Simple Economics of Open Source. In : *The Journal of Industrial Economics* [en ligne]. 2002. Vol. 50, n° 2, p. 197—234. Disponible à l'adresse : <https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-6451.00174>.

LEVINE, Peter, 2014. Why There Will Never Be Another RedHat : The Economics Of Open Source. In : *TechCrunch* [en ligne]. février 2014. Disponible à l'adresse : <https://techcrunch.com/2014/02/13/please-dont-tell-me-you-want-to-be-the-next-red-hat/>.

LEVY, Steven, 1984b. *Hackers : Heroes of the Computer Revolution*. New York : Anchor Press.

LICQUIA, Jeff et MCPHERSON, Amanda, 2015. *A \$5 Billion Value : Estimating the Total Development Cost of Linux Foundation's Collaborative Projects* [en ligne]. Linux Foundation. Disponible à l'adresse : https://www.static.linuxfound.org/sites/main/files/lfpub_cp_cost_estimate2015.pdf.

MATTIS, Peter, DARNELL, Ben et KIMBALL, Spencer, 2019. Why We're Relicensing CockroachDB. In : [en ligne]. juin 2019. Disponible à l'adresse : <https://www.cockroachlabs.com/blog/oss-relicensing-cockroachdb/>.

- MCPHERSON, Amanda, PROFFITT, Brian et HALE-EVANS, Ron, 2008. *Estimating the Total Cost of a Linux Distribution* [en ligne]. The Linux Foundation. Disponible à l'adresse : <https://www.linux.com/publications/estimating-total-cost-linux-distribution>.
- MEIER, Trevor, 2016. *A New Economy* [en ligne]. 2016. Disponible à l'adresse : <http://www.aneweconomy.ca/>.
- MERTON, Robert K., 1973. The Normative Structure of Science. In : *The Sociology of Science : Theoretical and Empirical Investigations*. Chicago : Chicago University Press.
- MITCHELL, Kyle E., 2017. *The License Zero Manifesto : Sustainable Software in the Open* [en ligne]. septembre 2017. Disponible à l'adresse : <https://writing.kemitchell.com/2017/09/12/The-License-Zero-Manifesto.html>.
- MOODY, Glyn, 2019. If Software Is Funded from a Public Source, Its Code Should Be Open Source. In : *Linux Journal* [en ligne]. février 2019. Disponible à l'adresse : <https://www.linuxjournal.com/content/if-software-funded-public-source-its-code-should-be-open-source>.
- MULLENWEG, Matt, 2014. *Five for the Future* [en ligne]. septembre 2014. Disponible à l'adresse : <https://ma.tt/2014/09/five-for-the-future/>.
- MUSTONEN, Mikko, 2003. Copyleft—the economics of Linux and other open source software. In : *Information Economics and Policy* [en ligne]. 2003. Vol. 15, n° 1, p. 99—121. Disponible à l'adresse : <http://www.sciencedirect.com/science/article/pii/S0167624502000902>.
- MUSTONEN, Mikko, 2005. When Does a Firm Support Substitute Open Source Programming? In : *Journal of Economics & Management Strategy* [en ligne]. 2005. Vol. 14, n° 1, p. 121—139. Disponible à l'adresse : <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1430-9134.2005.00036.x>.
- NICHOLSON, Deb, 2018. *Matthew Garrett : Software Freedom Activist* [en ligne]. décembre 2018. Disponible à l'adresse : <https://sfconservancy.org/blog/2018/dec/13/mjgInterview/>.
- NOWOGRODZKI, Anna, 2019. Tips for Open-Source Software Support. In : *Nature* [en ligne]. juillet 2019. Vol. 571, p. 133—134. Disponible à l'adresse : <https://www.nature.com/magazine-assets/d41586-019-02046-0/d41586-019-02046-0.pdf>.

O'FLAHERTY, Kate, 2018. Taiwan's revolutionary hackers are forking the government. In : *Wired* [en ligne]. avril 2018. Disponible à l'adresse : <https://www.wired.co.uk/article/taiwan-sunflower-revolution-audrey-tang-g0v>.

O'GRADY, Stephen, 2015. *The Software Paradox : The Rise and Fall of the Commercial Software Market*. Sebastopol : O'Reilly Media.

OPEN TECHNOLOGY FUND, 2017. *FY 2017 Annual Report* [en ligne]. Open Technology Fund. Disponible à l'adresse : https://www.opentech.fund/documents/8/FY2017_OTF_Annual_Report.pdf.

ORAM, Andy, 2018. Making institutional free software successful. In : *LWN.net* [en ligne]. avril 2018. Disponible à l'adresse : <https://lwn.net/Articles/750781/>.

ORLAND, Kyle, 2018. Sony using open source emulator for PlayStation Classic plug-and-play. In : *Ars Technica* [en ligne]. novembre 2018. Disponible à l'adresse : <https://arstechnica.com/gaming/2018/11/sony-using-open-source-emulator-for-playstation-classic-plug-and-play/>.

OSTROM, Elinor, 1990c. *Governing the commons : The evolution of institutions for collective action*. Cambridge : Cambridge University Press.

PERRET, Florian et NIEDDU, Martino, 2007. Une régulation de l'hybridation entre marchand et non-marchand. Les cas des formes de production de logiciels libres. In : *Revue internationale de l'économie sociale* [en ligne]. mai 2007. n° 304, p. 63—75. Disponible à l'adresse : <https://www.erudit.org/fr/revues/recma/2007-n304-recma01090/1021532ar/>.

PHIPPS, Simon, 2010. Open Core Is Bad For You. In : *Computer World UK* [en ligne]. juin 2010. Disponible à l'adresse : <https://web.archive.org/web/20110128014410/http://blogs.computerworlduk.com/simon-says/2010/06/open-core-is-bad-for-you>.

PRENTICE, Brian, 2010. *Open-Core : The Emperor's New Clothes* [en ligne]. mars 2010. Disponible à l'adresse : http://blogs.gartner.com/brian_prentice/2010/03/31/open-core-the-emperors-new-clothes/.

QUOSHI, Elio, 2015. *ura - Design for Free & Open Source projects* [en ligne]. mai 2015. Disponible à l'adresse : <https://elioqoshi.me/2016/05/ura/>.

RASHID, Fahmida Y., 2016. Time is running out for NTP. In : *InfoWorld* [en ligne]. novembre 2016. Disponible à l'adresse : <https://www.infoworld.com/article/3144546/security/time-is-running-out-for-ntp.html>.

RAYMOND, Eric S., 1999d. *The Cathedral and the Bazaar : Musings on Linux and Open Source by an Accidental Revolutionary*. Sebastopol : O'Reilly Media.

REDA, Julia, 2019. *In January, the EU starts running Bug Bounties on Free and Open Source Software* [en ligne]. janvier 2019. Disponible à l'adresse : <https://juliareda.eu/2018/12/eu-fossa-bug-bounties/>.

RED HAT INC., 2018. [en ligne]. octobre 2018. Disponible à l'adresse : <https://www.redhat.com/en/about/press-releases/ibm-acquire-red-hat-completely-changing-cloud-landscape-and-becoming-worlds-1-hybrid-cloud-provider>.

RIEHLE, Dirk, 2012. The Single-Vendor Commercial Open Source Business Model. In : *Information Systems and e-Business Management* [en ligne]. 2012. Vol. 10, n° 1, p. 5—17. Disponible à l'adresse : <https://dirkriehle.com/wp-content/uploads/2010/11/svcos-isebm-v05-final-final.pdf>.

ROSSI, Maria Alexandra, 2004. Decoding the 'Free/Open Source (F/OSS) Software Puzzle' : a survey of theoretical and empirical contributions. In : *The Economics of Open Source Software Development : Analyzing Motivation, Organization, Innovation and Competition in the Open Source Software Revolution*. Amsterdam : Elsevier Press. p. 15—55.

RYAN, Camille, 2018. *Computer and Internet Use in the United States : 2016* [en ligne]. U.S. Census Bureau. Disponible à l'adresse : <https://www.census.gov/library/publications/2018/acs/acs-39.html>.

SANDLER, Karen et KUHN, Bradley M., 2020. *Open Source Won, but Software Freedom Hasn't Yet* [en ligne]. Disponible à l'adresse : https://fosdem.org/2020/schedule/event/open_source_won.

SCHOLZ, Trebor, 2017. *How Platform Cooperativism Can Unleash The Network* [en ligne]. février 2017. Disponible à l'adresse : <https://novaramedia.com/2017/02/19/how-platform-cooperativism-can-unleash-the-network/>.

SCHOLZ, Trebor et SCHNEIDER, Nathan, 2016. *Ours to Hack and to Own. The Rise of the Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. New York : OR Books.

SOFTWARE FREEDOM CONSERVANCY INC., 2015. *The Principles of Community-Oriented GPL Enforcement* [en ligne]. 2015. Disponible à l'adresse : <https://sfconservancy.org/copyleft-compliance/principles.html>.

STACK OVERFLOW, 2018. *Stack Overflow Annual Developer Survey 2018* [en ligne]. mai 2018. Disponible à l'adresse : <https://insights.stackoverflow.com/survey/2018/>.

STEVENSON, Sarah, 2019. *Strength in Numbers : The Final Count Is In* [en ligne]. janvier 2019. Disponible à l'adresse : <https://blog.torproject.org/strength-numbers-final-count>.

SULLIVAN, John, 2018. *Free Software Foundation receives \$1 million from Handshake* [en ligne]. décembre 2018. Disponible à l'adresse : <https://www.fsf.org/news/free-software-foundation-receives-1-million-from-handshake>.

TANG, Audrey, 2018. *Q&A Session with Minister Tang* [en ligne]. Disponible à l'adresse : <https://saimei.ftp.acc.umu.se/pub/debian-meetings/2018/DebConf18/2018-07-28/q-a-session-with-minister-tang.webm>.

TERREL, Andy, 2017. *Why Is NumPy Only Now Getting Funded?* [en ligne]. juin 2017. Disponible à l'adresse : <https://numfocus.org/blog/why-is-numpy-only-now-getting-funded>.

THE TREASURY BOARD OF CANADA SECRETARIAT et THE DEPARTMENT OF DIGITAL, CULTURE, MEDIA AND SPORT OF THE UNITED KINGDOM OF GREAT BRITAIN AND NORTHERN IRELAND, 2017. *Memorandum of Understanding between the Treasury Board of Canada Secretariat and the Department of Digital, Culture, Media and Sport, of the United Kingdom of Great Britain and Northern Ireland Concerning Digital Government* [en ligne]. octobre 2017. Disponible à l'adresse : <https://www.canada.ca/en/treasury-board-secretariat/services/innovation/memorandum-understanding-concerning-digital-government.html>.

THOMPSON, Ben, 2019. *AWS, MongoDB, and the Economic Realities of Open Source*. In : [en ligne]. janvier 2019. Disponible à l'adresse : <https://stratechery.com/2019/aws-mongodb-and-the-economic-realities-of-open-source/>.

TOMAZELI, Bianca, 2019. *Comment utiliser les logiciels libres pour faciliter l'innovation digitale pour les municipalités* [en ligne]. Disponible à l'adresse : <https://agendadulibre.qc.ca/events/2017>.

UNDP, 2019. *Human Development Report 2019 : Beyond Income, Beyond Averages, Beyond Today - Inequalities in Human Development in the 21st Century* [en ligne]. United Nations. Disponible à l'adresse : <http://hdr.undp.org/en/content/human-development-report-2019>.

UNITED STATES SENATE, 1961. Copyright in Government Publications. In : *Copyright Law Revision : Studies Prepared for the Subcomm. on Patents, Trademarks, and Copyrights of the Senate Comm. on the Judiciary* [en ligne]. 1961. p. 23—42. Disponible à l'adresse : <http://www.copyright.gov/history/studies/study33.pdf>.

U.S. CENSUS BUREAU, 2010. *Computer and Internet Use in the United States : 1984 to 2009* [en ligne]. U.S. Census Bureau. Disponible à l'adresse : <https://www.census.gov/data/tables/time-series/demo/computer-internet/computer-use-1984-2009.html>.

VARDA, Kenton, 2017. *Introducing Cloudflare Workers : Run JavaScript Service Workers at the Edge* [en ligne]. septembre 2017. Disponible à l'adresse : <https://blog.cloudflare.com/introducing-cloudflare-workers/>.

VAUGHAN-NICOLS, Steven J., 2018. Microsoft open-sources its patent portfolio. In : *ZD-Net* [en ligne]. octobre 2018. Disponible à l'adresse : <https://www.zdnet.com/article/microsoft-open-sources-its-entire-patent-portfolio/>.

VERBEEK, Marno, 2017. *A Guide to Modern Econometrics*. Hoboken : John Wiley & Sons.

VÉRONNEAU, Louis-Philippe, 2018. *De la compétition à la coopération ou Comment le code informatique est devenu propriétaire* [en ligne]. Disponible à l'adresse : <https://veronneau.org/media/blog/2018-12-29/paper.pdf>.

WEBER, Steven, 2000. *The Political Economy of Open Source Software* [en ligne]. Disponible à l'adresse : <https://econpapers.repec.org/paper/cdlucbrie/qt3hq916dc.htm>.

WHITE, Halbert, 1980. A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity. In : *Econometrica* [en ligne]. mai 1980. Vol. 48, n° 4, p. 817—838. Disponible à l'adresse : <http://www.jstor.org/stable/1912934>.

WILLIAMS, Sam, 2002. *Free as in Freedom : Richard Stallman and the Free Software Revolution*. Sebastopol : O'Reilly Media.

WU, Chong-Guang, GERLACH, James H. et YOUNG, Clifford E, 2007. An empirical analysis of open source software developers' motivations and continuance intentions. In : *Information & Management*. janvier 2007. Vol. 44, p. 253—262.

YOSHIDA, Junko, 2018. MIPS Goes Open Source. In : *EE Times* [en ligne]. décembre 2018. Disponible à l'adresse : https://www.eetimes.com/document.asp?doc_id=1334087.

ZUEGEL, Devon, 2019. *Let's talk about open source sustainability* [en ligne]. janvier 2019. Disponible à l'adresse : <https://github.blog/2019-01-17-lets-talk-about-open-source-sustainability/>.

ZUKIN, Sharon et PAPADANTONAKIS, Max, 2017. Hackathons as Co-optation Ritual : Socializing Workers and Institutionalizing Innovation in the 'New' Economy. In : *Research in the Sociology of Work*. 2017. Vol. 31, p. 157-181.