

Place de l'incertitude dans la phase de captation de la valeur par les développeurs des projets logiciels Open source

Martial KADJI NGASSAM

ESSEC – Université de Douala (Cameroun) LAREQUOI IAE de Lille

Jean BABEI

Université de Douala (Cameroun) LETA Université de Douala & CRETLOG Aix-Marseille
Université

Pour citer la référence

KADJI NGASSAM Martial & BABEI Jean (2024). « Place de l'incertitude dans la phase de captation de la valeur par les développeurs des projets Open Source », *Revue Psychanalyse & Management – Édition académique en Ligne* ISSN 2739-9656 - n° 01_2024, pp. 279_286

Introduction : Si vous ne savez pas ou allez, n'importe quel chemin peut vous s'y conduire, écrivait Talmude en 2021. Cette citation montre que toutes les activités humaines sont attachées à une incertitude. Aussi toute captation de valeur comporte un degré d'incertitude, y compris chez les développeurs de projet logiciel Open source. Les projets et communautés de logiciels libres comptent de multiples acteurs aux intérêts divers et variés. Dans cette diversité d'acteurs hétérogènes (Zune et *al.*, 2011), on peut citer les développeurs, les fondations, les éditeurs, les distributeurs, les clients mutualistes et les institutions publiques ou collectivités territoriales. Ces différents acteurs ont des raisons diverses d'agir et de s'impliquer dans des projets open source. L'enjeu pour eux est de définir et de mettre en place des conventions qui sont suffisamment partagées, anticipables et aussi efficaces, pour faciliter la coordination et l'action collective (Zune et *al.*, *Ibid*). On pourrait se poser la question de savoir quelle place occupe l'incertitude dans la phase de captation de la valeur par les développeurs des projets logiciels Open Source.

1. Les développeurs des projets logiciels Open source

1.1. Les logiciels Open source

La plupart des outils numériques sont caractérisés par deux composantes essentielles : la composante matérielle et la composante logiciel. La première composante est représentée par le processeur, le disque dur, le lecteur, la carte mère, etc. Alors que la seconde est un ensemble de programmes, des séquences d'instructions interprétables, et mise en œuvre par l'ordinateur.

Un logiciel open source ou logiciel libre est un logiciel dont l'accès aux codes sources contrairement aux logiciels dit « propriétaires » est ouvertement disponible. Les codes sources sont également modifiables par tous. Une fois publié, le code source du logiciel n'appartient plus à ses créateurs et son évolution future dépendra de ce que les autres usagers en feront (Meyer et Montage, 2007). Car ces créateurs de logiciel ont rendu le secret de conception librement accessible et réutilisable (Chevalier, 2005). D'après la *Free Software Foundation* (FSF), un logiciel pour être qualifié de libre ou open source doit respecter quatre libertés :

- Liberté d'exécuter le programme pour tous les usages,
- Liberté d'étudier le fonctionnement du programme et l'adapter selon son besoin (d'où la nécessité d'accès aux codes sources),
- Liberté de redistribuer des copies,
- Liberté d'améliorer le programme et de publier les améliorations au public

La construction de logiciel requiert des notions de programmation autrement de mise en œuvre de séquences d'instruction liées à des buts précis. Cette activité est la panacée des développeurs de logiciel. C'est une activité correspondant à une expression explicite du travail de programmation

ouverte à tous (Zimmerman et *al.*, 2011). L'activité du développeur dans l'open source consiste donc pour ces programmeurs à construire des bases de données et des logiciels.

1.2. Les différents développeurs de logiciels

Le développeur dans un projet open source est toute personne qui se manifeste spontanément pour apporter librement et volontairement en dehors de son temps professionnel et familial, une contribution individuelle non rémunérée, avec des intérêts qui lui sont propres (Pesqueux, 2010). Au départ dans les projets logiciel libre, les développeurs programmeurs étaient intéressés à construire et à échanger des programmes développés pour leurs propres besoins, ou du moins des logiciels dont ils avaient besoin dans leur quotidien (Lakhani et Wolf, 2006). Mais de nos jours, l'activité de bénévolat a suivi le développement philosophique et idéologique induit par les discussions autour financiarisation de cette activité et son introduction dans le monde marchand (Mouakhar et Tellier, 2013). Les développeurs de nos jours sont beaucoup plus intéressés et leur seule motivation n'est plus seulement la contribution à un projet passionnant. Les motivations ne sont donc pas restées intactes (Torre, 2011). Ces motivations peuvent être regroupées en deux catégories : les motivations intrinsèques (lorsque l'activité est exercée et appréciée pour elle-même et pour le plaisir) et les motivations extrinsèques (qui vont fournir aux bénévoles des récompenses indirectes pour les tâches à accomplir) (Ryan et Deci, 2000). Nous allons dans la suite de cet article analyser les différentes motivations des développeurs dans les communautés open source en vue d'identifier des figures hybrides associées à des légitimités variables (Demazière et *al.*, 2009) et dont les motivations réelles et l'éthique sont parfois floues (Hertel et *al.*, 2003).

Parmi les développeurs de logiciel libre, on distingue ceux qui agissent en tant que militants désintéressés, ceux qui sont des militants intéressés ou des développeurs rémunérés.

Les développeurs militants désintéressés se rassemblent le plus souvent autour de normes sociales fortes telle que le rejet de la hiérarchie, la promotion de la décentralisation, le partage de l'information et l'attachement à la communauté (Levy, 1984). Ils peuvent aussi être des chômeurs sans emploi et n'ayant aucun espoir d'en obtenir à cause de leurs contributions, expériences et engagement militant dans les projets de logiciels libres.

Le développeur militant intéressé quant à eux bien qu'ayant des revendications militantes et politiques espère tirer d'autres profits de son engagement. Il va en effet tout en respectant les « principes » et « l'éthique » fondateurs du logiciel libre, développer des stratégies individuelles pour s'approprier ou capter la valeur qu'il a contribué à créer dans le cadre de son activité de bénévole. Ces individus sont le plus souvent des étudiants en informatique, des chômeurs en recherche d'emplois, des personnes au foyer et des retraités. L'étudiant en informatique, lui, va par exemple essayer de s'impliquer dans des projets open source en vue de faire ses preuves, d'acquérir de l'expérience et de se faire une réputation qu'il pourra éventuellement capitaliser en se faisant recruter à la fin de ses études dans une entreprise travaillant sur des projets open source.

Enfin le développeur ici est rémunéré indirectement par son activité de contributeur bénévole. En effet, en participant activement aux projets de logiciels libres, certains bénévoles (salariés dans le domaine informatique, consultants et prestataires indépendants) accroissent leurs compétences, leur renommée qu'ils vont capitaliser financièrement. Comme la vente de services exige une maîtrise approfondie du logiciel (Demazière et *al.*, Ibid), le consultant ou prestataire de services indépendants va optimiser sa renommée et compétence dans le projet dans lequel il s'implique. Ensuite il pourra vendre des prestations de services (formation, SAV, installation). On distingue une autre catégorie de personnes rémunérées par des entreprises pour être bénévoles dans certaines communautés. Ils sont donc mis à disposition de la communauté ou d'un projet Open Source dans le but de le développer et surtout de le faire évoluer dans le sens souhaité et selon les intérêts de leur employeur. C'est le cas des grandes entreprises telles que RedHat et Microsoft qui emploient des dizaines d'ingénieurs dont le rôle principal est la participation au développement du noyau Linux. Dans certains cas, ce type bénévoles cachent aux autres membres de la communauté leurs statuts de « salariés missionnés pour contribuer dans le projet Open Source ». C'est ainsi que parfois des demandes pressantes en termes de délais et de fonctionnalités en contradiction avec l'activité de bénévolat pourraient amener des

bénévoles à se désengager, avec pour conséquence une communauté de plus en plus constituée d'une forte composante de salariés délégués par les entreprises (Julien et Zimmerman, 2005).

L'hybridation des logiques militantes et économiques constitue un enjeu central au sein des projets de logiciels libres. Elle accentue l'incertitude des différents acteurs à trouver satisfaction de leurs engagements. Elle engage la survie de ces groupes fragiles fondés sur des engagements volontaires et soumis à des obligations de résultats sous peine de décourager l'activité de création collaborative (Démazière, 2013).

2. Incertitude dans la captation de la valeur avec les projets Open source

2.1. Sur la phase de captation de la valeur

La phase de captation de la valeur est particulière dans l'open source dans la mesure où même les acteurs n'ayant pas participé à sa création ont la possibilité de la capter (Perens, 2005 ; Muselli, 2005 ; Dahlander *et al.*, 2008 ; Kadji, 2014b). Les acteurs de projets open source sont en effet multiples et variés. Et l'on peut citer les développeurs, les fondations, les éditeurs de logiciels, les intégrateurs, les distributeurs, les constructeurs de matériel et les clients utilisateurs (individus ou entreprises) ; en fin les pouvoirs publics ou collectivités territoriales. Il existe plusieurs modèles économiques de captation de la valeur dans l'open source.

Le tableau 1 ci-dessous présente les modèles les plus courants.

Tableau 1 : Les outils de captation de valeur

Type de modèle de valorisation	Exemple d'outils de captation de revenus	Acteurs concernés
Vente de licences	Double licence Licences décalées Vente module supplémentaire	Éditeurs Exemple : MsQl
Vente de prestations	Vente service de maintenance Vente service de formation Vente service de garantie et d'assurance	Prestataires, intégrateurs (SSLL) Exemple : Linagora, Nuxeo
Capter des revenus par la Mutualisation	Création du club de clients utilisateurs mutualistes Commande groupée	Editeur, Prestataire, Client entreprise, Collectivités Territoriales. Exemple : Adullact

Plusieurs travaux (Von Hippel et Von Krogh, 2003 ; Bonaccorsi *et al.*, 2006 ; Le Texier et Versailles, 2009 ; Torre, 2011 ; Benkeltoum, 2011) mettent en avant la stratégie de valorisation économique basée sur le type d'acteurs de la communauté du libre (les développeurs, les fondations, les distributeurs, les éditeurs open source et les prestations). Parmi la multitude de modèles économiques ou de captation de revenus nous retenons trois grandes catégories (Smiles, 2013) : La vente de licence, les prestations de services et la mutualisation.

- La vente de licences (double licence ou hybride, licence décalée, modules complémentaires)

Il est provocateur de parler de vente de licences lorsqu'on parle des logiciels libres. En effet ici il ne s'agit pas de remettre en cause les principes d'ouverture des codes source et le respect des libertés fondamentales. Il est plutôt question pour nous ici de présenter les utilisations des licences comme outils permettant de transformer la valeur créée en revenu. Cette stratégie de valorisation par des licences est le plus souvent utilisée par des éditeurs de logiciels libres. Pour cela, ils ont mis au point plusieurs options dont les plus répandues sont la double licence, Open Core, la licence décalée et les licences hybrides (April, 2007, Smiles, 2013).

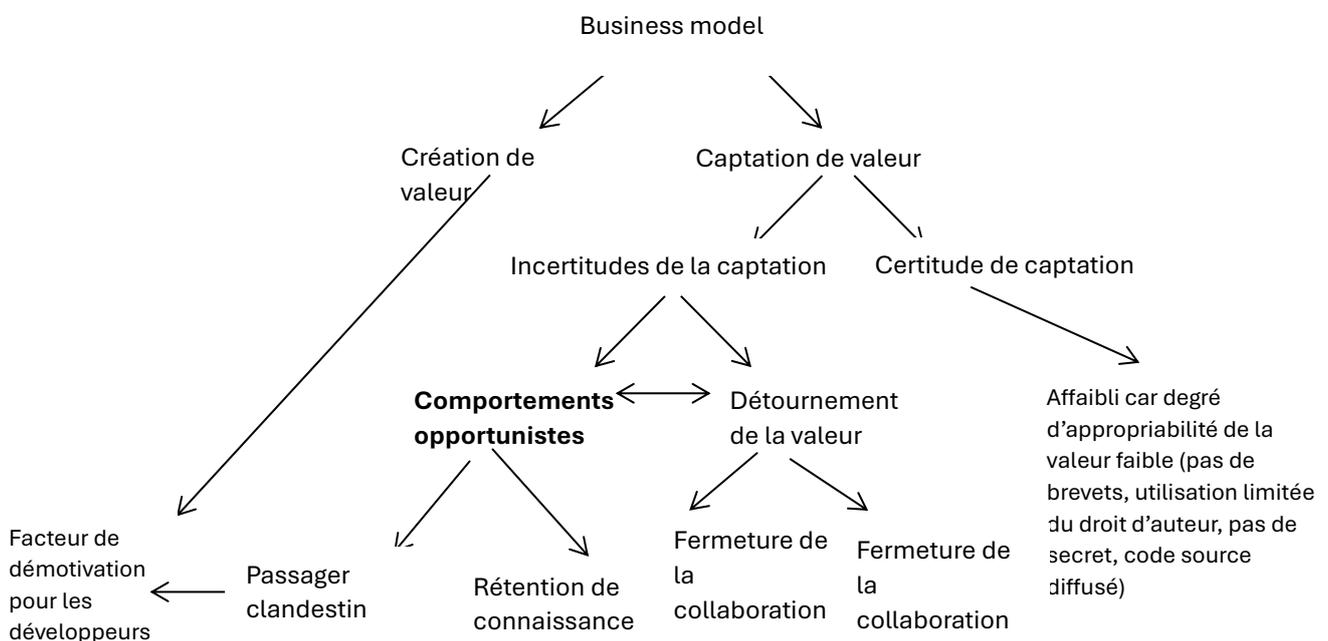
- La vente de prestations de services et de support

Ce type de modèle de captation de revenus est utilisé par les distributeurs, les prestataires et par certains membres des communautés et des fondateurs (Perens, 2005). Ces derniers valorisent en matière de services leurs expertises sur le projet Open Source dont ils ne sont pas éditeurs ou initiateur (Smiles, 2013). Toutefois ces derniers essaient le plus possible de contribuer et de participer à ces projets. Pour générer des revenus autour du code, ils mettent en place des contrats de vente de services de support, de vente de garantie et aussi dans certains cas de développement de modules spécifiques (Hecker, 2000)

- La captation de valeur basée sur la Mutualisation

Dans ce cas de figure, le participant au projet collaboratif ne génère pas de revenu par la vente de services ou de licences liés au logiciel open source dans lequel il collabore (Elie, 2009 ; Kadji, 2014a). La valeur créée se trouve plutôt au niveau du potentiel de réduction espéré des frais de R&D. Partant d'un business model, les articulations de la captation de valeur peuvent être éclairées par la figure1 suivante :

Figure 1 : Articulation entre Business model, développeur et incertitude



2.2. La captation de valeur et la psychologie des développeurs d'Open source

La psychologie est la science de la vie mentale. C'est la science du comportement et des processus mentaux. C'est la science des comportements des individus. Elle permet de comprendre et de prévoir leurs comportements. Le but ici est de comprendre la structure et le fonctionnement de l'activité mentale et des comportements qui lui sont associées. La littérature relève l'existence des troubles psychologiques qui sont abordés par la psychanalyse.

Il s'agit de l'investigation des processus psychiques profonds, de l'inconscient dont les fondements se trouvent dans les travaux de Freud. L'individu agissant de manière inconsciente

Au mépris du réel. Aussi, la participation des développeurs dans l'Open source pourrait se justifier selon deux courants : celui des motivations extrinsèques et celui des motifs intrinsèques. Pour les tenants du courant extrinsèques, les développeurs agissent en homo-economicus rationnels. Ainsi, pour Lerner et Tirole (2002), la participation des individus est avant tout guidée par des motivations extrinsèques. En utilisant son temps libre à l'écriture d'un code-source ouvert, le développeur se livre en vérité à un calcul économique. L'individu investit de son temps parce qu'il est sûr de retirer un bénéfice de sa participation dans l'Open source. Et la construction d'une réputation sur le marché du

travail et pas simplement à l'intérieur d'une communauté (Lahkani et von Hippel, 2003), constitue pour les auteurs une motivation importante. Ceci est d'autant plausible que pour les auteurs, l'ampleur et la qualité des contributions individuelles est facilement observable, non seulement par les membres de la communauté mais également par de nombreux observateurs extérieurs. De même, le besoin de se former et la recherche d'une solution à un problème strictement personnel sont généralement rangés dans ces motifs extrinsèques.

Alternativement à ce qui précède, certains développeurs ne sont pas attirés par des gratifications financières (ou même à des avantages y liés) lorsqu'ils coopèrent dans un projet Open source. La participation des individus serait fondamentalement motivée par le désir de se fondre dans une culture communautaire, le besoin de résoudre des difficultés techniques ou encore par pure altruisme, la culture du partage et le besoin de s'adonner à une activité créative en collaboration à l'Open source.

De même l'aspect communautaire de l'Open source est également valorisé chez les participants. Or, l'identification à une communauté implique généralement pour l'individu de se conforter à un ensemble de règles et de normes collectives plus ou moins explicites. Il s'agit par exemple de soumettre son travail aux jugements de ces pairs mais aussi d'obéir à des règles implicites : je participe au projet libre car j'ai moi-même profité des travaux de la communauté. Les motivations sont alors dites intrinsèques.

Au total, d'un point de vue psychanalytique la première attitude pourrait se justifier par une position matérielle confortable du développeur. Alors que la seconde se justifierait par une situation matérielle non confortable du développeur.

3. Méthodologie de recherche

3.1. Protocole de recherche et de collecte des données

Le présent travail de recherche résulte une étude de cas réalisée dans une situation d'observateur non participant. Ce choix de la méthode par étude de cas a permis de réaliser une enquête empirique permettant d'analyser en profondeur un phénomène contemporain dans son contexte de vie réelle (Yin, 1984). Notre cas concerne une entreprise privée ayant décidée de concevoir des logiciels avec des codes ouverts. Pour y parvenir et afficher son appartenance à la mouvance du logiciel libre cette dernière s'appuie sur une communauté de développeurs auprès de laquelle elle s'engage à partager la gouvernance du projet et de développer le code source dans le sens des intérêts de la communauté. Nous verrons dans ce cas précis que le non-respect de l'engagement de la part de cette entreprise et l'incertitude des développeurs à capter la valeur à l'issue de la phase de co-création va entraîner des tensions entre cette firme et sa communauté au point d'entacher la réputation de cette dernière et mettre en danger la pérennité de son modèle d'affaire.

En effet, dès le départ, les différents acteurs ont pris un engagement de création harmonieuse et optimale de la valeur sans toutefois s'accorder sur les modalités de captation à la fin de la phase collaborative. Comme nous allons le voir, les acteurs de ce projet n'ont pas tenu à leurs engagements en développant pour la grande partie des membres des comportements de *free riding* pendant le projet et en se battant à la fin de la phase collaborative sur les actifs ayant émergé de cette collaboration.

Une synthèse des caractéristiques de chacun de notre cas est résumée dans le tableau ci-dessous.

Tableau 2 : Caractéristiques et spécificités de notre étude de cas.

Caractéristiques	Types de financements	Nature des acteurs	États d'avancement et notoriété du projet	Gouvernance et pilotage du projet
Cas : Open Colab	Financement privé et public	Co-associés salariés. 15 développeurs (dont 10 sont salariés chez OPEN COLAB SA). Présence de quelques contributeurs non-développeurs.	Créé en 2003. L'entreprise OPEN COLAB est rentable avec ses activités open source. Elle a une certaine notoriété en France	Entreprise Dirigée par le PDG. Les développeurs de la communauté sont pour la plupart des salariés de l'entreprise OPEN COLAB SA.

Choix de triangulation méthodologique nous a conduits à l'utilisation de plusieurs sources de données empiriques (voir Tableau 4).

Tableau 3 : Sources et modalités de collecte des données

Types de données collectées	Descriptif
Données issues des entretiens semi-directifs (22 entretiens réalisés ; Durée moyenne 1h30)	Verbatims issus des retranscriptions (plus de 520 pages) Guides d'entretien
Données issues des Observations non participante	Comptes rendus des Workshops mensuels Prises des notes (cahier de bord) Données collectées lors d'événements
Données provenant des études documentaires et exploratoires	Publications liées aux projets (presses spécialisées, pôle de compétitivités, etc.)

3.2. Traitement des données

Ces interviews qui ont été systématiquement enregistrés ont fait par la suite l'objet d'une retranscription fidèle pour un total de près de 520 pages. A cela nous avons rajouté des données secondaires (issues des rapports d'étapes, publications des acteurs et du pôle de compétitivité Systematic) d'environ 170 pages. Cette triangulation des données avait pour but d'accroître la validité de nos résultats et de combler ainsi les limites et critiques envers la démarche qualitative. L'analyse de contenu que nous avons par la suite décidé de mettre en place, nous a permis, grâce à un ensemble de procédures structurées, de justifier les jugements portés sur l'émetteur du discours, sur le message lui-même, et le cas échéant, sur l'audience du message (Hlady-Rispal, 2002).

Ces 690 pages de données ont par la suite fait l'objet d'un codage en vue d'une analyse thématique. Ce codage a en partie été confirmé par un processus de double codage réalisé par un autre chercheur. Cela nous a aidé à réduire les biais et à garder un certain degré de rigueur tel que recommandé par Cole, C. et al (2011).

Une fois notre codage effectué, nous avons fait le choix du logiciel de traitement de contenu QSR NVivo 10. Le choix de ce logiciel vient du fait qu'il est l'un des plus récents en matière d'analyse de données qualitatives et il offre donc une multitude de possibilités de « faire parler » nos données.

1. Résultats et discussions

1.1 Résultats : études de cas

Cas Open Colab : Quand le non-respect des engagements et le flou dans les mécanismes de captation de la valeur entraîne une défection des contributeurs et le péril du projet collaboratif

Open Colab SAS édite une plateforme collaborative en *open source*. Ce projet d'entreprise est né dans le but de répondre à un besoin sur le marché en matière de complément ou d'alternative aux logiciels commerciaux. Le fonctionnement de ce projet et de l'entreprise associée correspond à la manifestation de la vision et des valeurs de son initiateur. Ce dernier croit en la collaboration comme levier de création logiciel comme le confirme l'extrait de l'interview suivant, il croit au mode de développement logiciel qu'offre l'open source : « *je suis convaincu que l'économie collaborative va prendre une part de plus en plus importante dans notre économie* ». Cette philosophie se retrouve notamment dans le choix d'appuyer le développement des activités de l'entreprise Open Colab SAS sur une communauté d'utilisateurs et de développeurs indépendants.

La base de code source produite dans la partie communautaire du projet Open Colab sert de socle aux produits et services vendus par l'entreprise Open Colab SAS. Cette dernière a plusieurs façons de capter la valeur produite par les membres de sa communauté *open source*. En effet, bien que le

code source soit ouvert et librement accessible à tous, l'entreprise Open Colab a mis en place des mécanismes liés à la gouvernance de la communauté qui ont pour but de lui permettre de s'accaparer l'essentiel de la valeur développée par cette dernière. C'est ainsi que le fait l'obtenir un « droit de *commit* »¹ dans le « cœur » du projet soit compliqué permet aux *committers* déjà présents, qui sont pour la plupart aussi salariés et/ou actionnaires d'Open Colab, d'intervenir, de développer des compétences autour du code et ainsi d'être cités massivement comme contributeurs principaux. La communauté est ainsi utilisée comme un levier moins onéreux de la création de la valeur comme l'affirme l'un de nos interviewés : « *Ce qui est bien est que dans la communauté, il ya des experts moins avarés de leur temps* ». Cela mettra l'entreprise Open Colab SAS en *pole position* lorsqu'il s'agira de profiter de cette notoriété pour vendre des services, du support et des développements spécifiques. De plus, l'entreprise s'arrange pour recruter la plupart du temps les développeurs de la communauté qui ont fait la preuve de leur talent. Cela permet ainsi à Open Colab SAS de s'assurer le contrôle des compétences sur le code du projet Open Colab.

De plus, aucun dispositif n'est mis en place en cas de nos respects des engagements pris par l'entreprise Open Colab vis-à-vis de la communauté de développeurs. Cette situation entraîne une anxiété des développeurs à ce projet. L'un d'eux affirme dans ce sens : « *l'absence de visibilité dans la valorisation de nos apports nous met dans une situation inconfortable. Nous donnons beaucoup de notre temps et de notre expertise sans savoir ce qu'on va gagner* ».

De plus, on note un non-respect par l'entreprise des engagements de gouvernance ouverte comme l'explique l'extrait suivant d'un ancien salarié de l'entreprise Open Colab : « [la communauté chez nous] *était gérée par des intervenant de la société qui géraient, car les développeurs étaient essentiellement ceux de la société, donc s'était clair que l'entreprise prenait pratiquement toutes les décisions sur la distribution, ce qui est logique car le système a été construit comme ça* ». Pour accorder ces droits de modification dans le « cœur » du projet à un contributeur, en plus des compétences de ce dernier, il faut un fort degré de confiance et de compatibilité de vision stratégique. Si malgré tous des blocages surviennent, l'ensemble des autres *committers* ont la possibilité de faire un vote pour exclure celui d'entre eux qui est la source du blocus de part son veto posé : « *Avant de mettre quelque chose d'important dans du code, ils le mettent, si quelqu'un est opposé par cet ajout, le reste de la communauté va revenir en arrière sur le code. Ils peuvent voter pour demander la suppression d'un code, à l'unanimité. Si on estime que ce code ne sert à rien dans le logiciel, il est enlevé, mais à l'unanimité* ». Ce verrouillage de la gouvernance et le non-respect des engagements pris vis-à-vis de la communauté ont entraîné une baisse des contributions des membres de la communauté du fait pour ces derniers d'une situation d'incertitude. Car si un développeur n'apporte pas des contributions conséquentes, ses citations (copywriting) et expertise dans le projet sont réduits et les perspectives de captation confortable de la valeur compromise. La figure ci-dessous nous en donne une illustration.

¹ Dans le langage *open source*, le « *committer* » est celui qui peut faire des apports en code source (*commit*) directement dans le cœur (ou noyau principal) du projet. Un développeur n'ayant pas le droit de *commit* ou le statut de *committer* ne peut faire que des propositions d'apports à un autre développeur qui est *committer* et qui, après évaluation, pourra l'intégrer dans la version principale (cœur) du projet. Toutefois, pour faciliter les contributions, les projets *open source* ont développé la possibilité de faire des extensions, « patch » ou « plugin ». Cela permet à tout développeur de contribuer au code source sans affecter le cœur du projet.

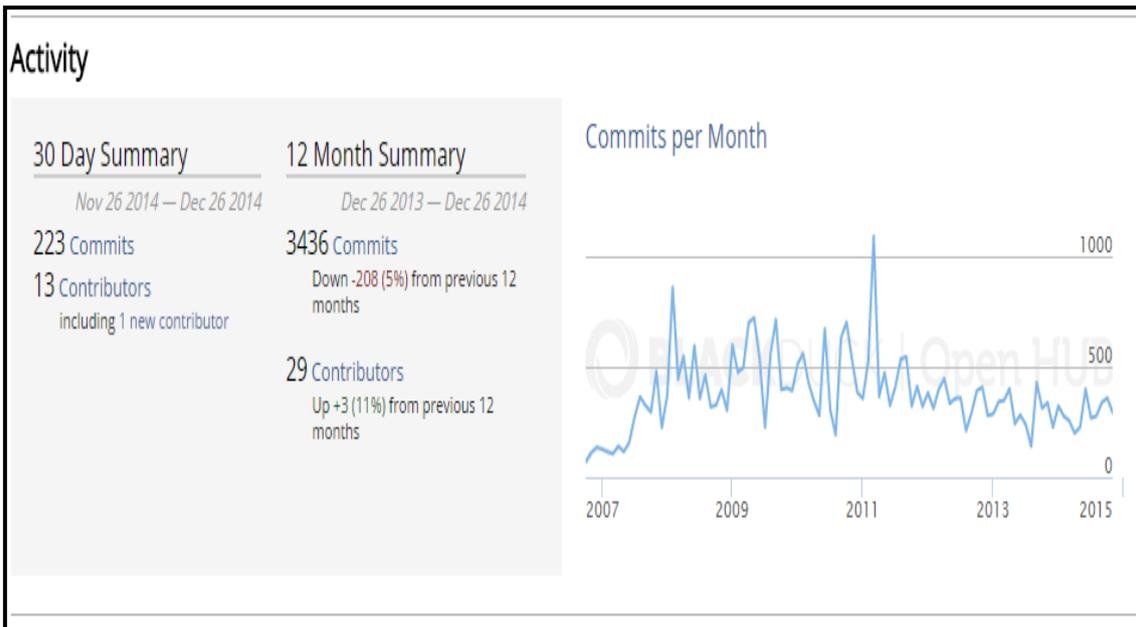


Figure 2 : évolution des contributions en code source Open Colab de 2007 à 2015
(Source : [https://www.openhub.net/p/Open Colab](https://www.openhub.net/p/Open%20Colab)).

Le graphique ci-dessus nous montre en effet que sur les 100 contributeurs habituels d’Open Colab il y’a que 13 qui ont effectué des contributions lors de période allant du 26 Nov 2014 au 26 DEC 2014. De plus, et en toute logique, le nombre de contributeurs mensuels a fortement diminué après un pic entre 2011 et 2012, pour passer aujourd’hui à moins 12 contributeurs mensuels (voir figure 3 ci-dessous). L’analyse de la composition et de l’origine de ces contributeurs actifs lors de 12 derniers mois nous permet d’affirmer qu’ils sont à plus de 90% membres de l’entreprise Open Colab SAS. Ce qui témoigne du contrôle que cette entreprise a sur la partie communautaire de son projet. Ce qui a pour effet de dissuader et de démotiver les contributeurs extérieurs.

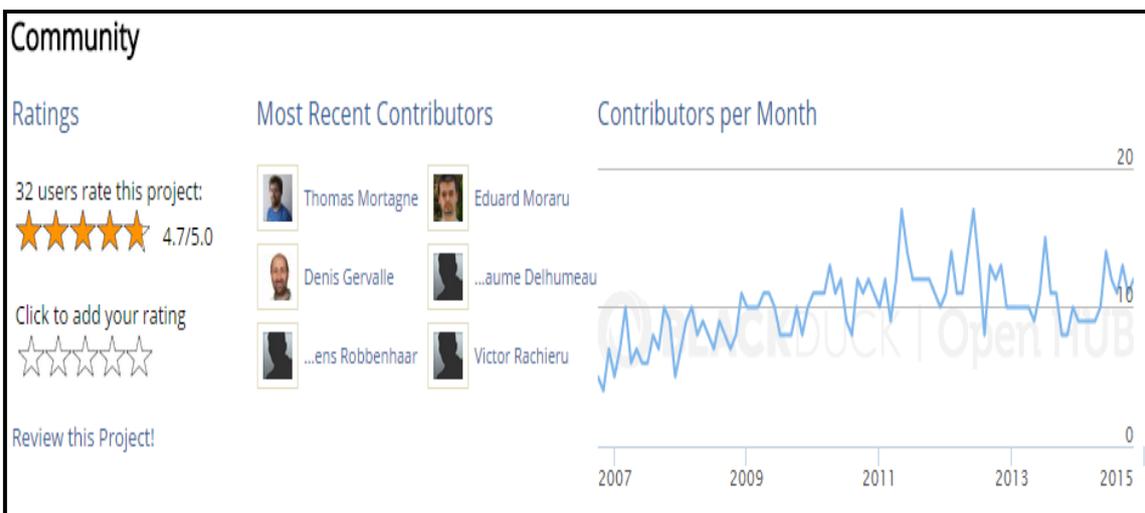


Figure 3 : Évolution mensuel du nombre de contributeur au projet Open Colab de 2007 à 2020
(Source : [https://www.openhub.net/p/Open Colab](https://www.openhub.net/p/Open%20Colab)).

Nous voyons donc comment une situation d’incertitude pousse l’entreprise Open Colab à verrouiller les contributions et à les réserver majoritaire à ces développeurs internes. Ceci pour s’assurer en exclusivité de l’expertise dans le projet et être ainsi en bonne position lors la phase de captation via la vente de service ou le développement des modules spécifiques. Dans ce projet, nous avons constaté

que les autres développeurs se sentant lésés ont adopté des comportements opportunistes notamment la rétention de connaissance et l'étude de la perspective de faire un fork² du projet dans lequel ils espèrent appliquer rigoureusement les principes et valeurs de l'Open source en matière de libre captation et partage de la valeur.

1.2. Discussions

Dans les projets collaboratifs ou en réseaux, il est nécessaire de créer, soit de provoquer ou de préserver un contexte favorable à la confiance et au respect des engagements ne matière de création et de captation de la valeur. Car, un environnement de confiance rassure les parties prenantes et incitent ces dernières à respecter leurs engagements, notamment par le fait d'une réduction des incertitudes. En effet, dans les processus de création collaborative de la valeur, pour la préserver, il est donc nécessaire de disposer d'outils de résolution de conflits, de dispositifs de sanction, de définitions d'engagements en vue de dissuader les comportements de non-confiance (Loilier et Tellier, 2004). Les auteurs proposent, en plus du mode coordination en réseau dans les innovations ouvertes, de mettre en place des dispositifs de sanctions et de règlements de situation de crise. Ils affirment que « pour limiter les risques d'appropriation unilatérale, plus largement les comportements opportunistes, les acteurs de réseau doivent socialiser leurs échanges, c'est-à-dire les inscrire dans un groupe social qui a ses règles de fonctionnement, ses coutumes, ses rites ». Il faut selon eux développer un fort degré de socialisation qui d'une part « accroît le coût des trahisons en augmentant le coût économique (exclusion des projets futurs) de coûts symboliques et sociaux (exclusion des manifestations propres au groupes sociale) ; d'autre part, elle assure l'optimalité de la régulation par le don puisque le comportement opportuniste sera sanctionné même en l'absence de contrat formel ». Ces pratiques si elles sont mises en place vont dissuader les envies de détournement de la valeur (Belketoum, 2011) et ainsi équilibrer les relations entre les acteurs. De plus, comme le dit Ferrary (2002), le fait d'exposer les tricheurs est une double sanction ; car, elle va inciter chacun des membres à refuser toute nouvelle collaboration avec ce dernier et ainsi favoriser le respect des engagements. En plus de cette coordination en réseau, certains auteurs proposent une sorte d'enchevêtrement avec la coordination par le marché et la hiérarchie (Bradach et Eccles, 1989 ; Brulhart et Favoreu, 2003 ; Loilier et Tellier, 2004). En effet, quelle que soit la bonne foi des acteurs présents, il est nécessaire de mettre en place des dispositifs de gouvernance et de réciprocité pour garantir l'équité des échanges et éviter les mauvaises surprises. Pour cela, il faut mettre en place en plus des ajustements informels des clauses contractuelles (Poppo et Zenger, 2002). Ces clauses et règles vont régir au sein de la communauté les trois phases qui sont : donner, recevoir et rendre (Perroux, 1960). C'est l'incertitude de la réciprocité qui fait planer les doutes sur l'équilibre des relations et le maintien de la confiance. Il faut donc trouver des mécanismes de confiance. Loilier et Tellier (2004) proposent une synthèse de neuf conditions de production de la confiance dans le cas particulier des réseaux distants. Nous utilisons cette grille pour l'analyse de la pérennité de l'Open Source, dans la mesure où la plupart des projets de logiciels libres se font en partie par des collaborations à distance via le réseau internet. Toutefois, pour certains projets, des réunions physiques et régulières sont organisées afin de permettre aux acteurs de se rencontrer et pour résoudre certains problèmes liés à la collaboration difficile à gérer à distance.

Les travaux de Loilier et Tellier (2004) sur les conditions de production et de maintien de la confiance dans les réseaux distants mettent en évidence le rôle des communautés et des écosystèmes d'acteurs. Ces écosystèmes sont des lieux où se regroupent des acteurs aux intérêts plus ou moins convergents. Ces derniers ont des tentations à l'adoption des comportements opportunistes et au non-respect des engagements se comportant comme des passagers clandestins. Dans ce contexte, les différentes dimensions de confiance que nous avons abordées (Zuker, 1986) seront déterminantes dans la démarche de co-construction de la valeur entre les différentes parties prenantes.

² *Un fork* est une copie d'un projet logiciel existant à un moment donné pour ... dans le développement des logiciels libres et *open source*

Conclusion

Dans un contexte de création collaborative et ouverte de la valeur, notre étude de cas montre que dans l'open source, la situation d'incertitude vient du non-respect des engagements par les parties prenantes. En effet, c'est le respect des engagements qui découle de la confiance, eu égard à la réciprocité des parties en présence. L'incertitude entraîne des comportements de détournement de la valeur, de free riding et des risques de fork. Remettant ainsi en question la pérennité des projets Open source

L'espoir que nous avons au travers de cet article est de poser des bases de réflexions sur le lien incertitudes et succès des projets open source.

Dès lors, on peut se poser la question de savoir si la situation de flou et d'incertitude dans la captation de la valeur n'est pas aussi une conséquence du fort degré d'ouverture des projets open source ?

Bibliographie

- April, (2005), *Les Modèles économiques du Logiciel Libre*, Livre Blanc.
- Benkeltoum, N. (2011), Regards sur les stratégies de détournement dans l'industrie open source, *Vie & sciences de l'entreprise*, 187, 72-94.
- Bonaccorsi, A. et C. Rossi (2003), Why Open Source can succeed, *Research Policy*, 32:7, 1243-1258.
- Chevalier, B. (2005), *Logiciels libres Open Source : Qu'est-ce que c'est ?* Paris : H&K.
- Dahlander, L. et M. Magnusson (2008), How do Firms Make Use of Open Source Communities? *Long Range Planning*, 41 : 6, 629-49.
- Demazière, D. (2013), Concilier projet militant et réussite économique du produit : le cas des logiciels libre, *La Découverte Réseaux*, 5 : 181, 25-50.
- Demazière, D., F. Horn, M. Zune (2009b), La socialisation dans les « communautés » de développement de logiciels libres, *Sociologie et Société*, 41 : 1, 217-238.
- Elie, F. (2009), *Economie du Logiciel Libre*, (ed) Eyrolles, Paris.
- Hlady-Rispal, M. (2002), *La méthode des cas, application à la recherche en gestion*, Bruxelles : De Boek.
- Hertel, G., S. Niednet et S. Herrmann (2003), Motivation of Software developers in open Source projects: an Internet-based survey of contributors to the Linux kernel, *Research policy*, 32, 1159-1177.
- Julien, N., et J.B. Zimmermann (2011), Floss in an industrialeconomics perspective, *Revue d'économie industrielle*, 136, 37-64.
- Levy, S. (1984), *Hackers: Heroes of the computer Revolution*, New York: Anchor Press/Double day.
- Meyer, M. et F. Montagne (2007), Le logiciel libre et la communauté autorégulée, *Revue d'Economie Politique*, 117 : 3, 387-405.
- Mouakhar, K. et A. Tellier (2013), Comment concilier marchand et non-marchand : une taxonomie empirique des comportements stratégiques des SSSL, *Revue Système d'Information et Management*, 3: 18, 5-36.
- Muselli, L. (2005), Open Source, création et captation de valeur, *14ème Conférence de l'AIMS*, Pays de la Loire (Angers-Nantes), Disponible sur <http://www.strategieaims.com/events/conferences/9-xiveme-conference-de-l-aims/communications/635-opensource-creation-et-captation-de-valeur/download>, consulté le 22 novembre 2013.
- Lerner, J. et J. Tirole (2003), Some simple economics of Open Source, *Journal of Industrial Economics*, 50: 2, 197-234.
- Ryan, R. M. et E.L. Deci (2000), Self-determination theory and the facilitation of intrinsic

motivation, social development, and well-being, *American Psychologist*, 55, 68-78.

Smiles (2014), *Comprendre l'open source et les logiciels libres*, Livre Blanc, SMILE, disponible au <http://www.smile.fr/Livres-blancs/Culture-du-web/Introduction-a-l-open-source>

Torre, D. (2011), Le modèle économique du logiciel Open Source : viabilité et compétitivité, *Revue d'économie industrielle*, 136 : 4, 11-16.

Pesqueux, Y. (2010), Bénévole, volontaire, militant, élu, *Archives Ouvertes*, Disponible sur http://halshs.archives_ouvertes.fr/docs/00/50/96/96/PDF/bA_nA_vole_volontaire_militant_A_lu.pdf, Consulté le 10 décembre 2013.

Perens, B. (2005), The Emerging Economics of Open-Source Software, Disponible sur www.firstmonday.org/ojs/index.php/fm/article/view/1470/1385. Consulté le 22 février 2011.

Von Hippel, E. et R. Katz (2003), Shifting Innovation to Users via Toolkits, *Management Science*, 48:7, 821-833.

Zune, M., D. Demazière, F. Horn et A. Vanheerswyngheles (2011), L'implication des pouvoirs publics dans les projets de logiciel libre, *Rapport Final du projet OSSPA*, Université Libre de Bruxelles.