



La santé des écosystèmes logiciels à code source ouvert : une revue de littérature systématique

Mémoire

Aime Noutsa Fobang

Maîtrise en sciences de l'administration - avec mémoire
Maître ès sciences (M. Sc.)

Québec, Canada

La santé des écosystèmes logiciels à code source ouvert : une revue de littérature systématique

Mémoire

Aimé Noutsa Fobang

Sous la direction de :

Josianne Marsan, directrice de recherche

Résumé

Un logiciel à code source ouvert (LCSO) ou *Open Source Software* est un logiciel dont le code source est accessible et modifiable par quiconque, et dont la licence assure que ce code demeure dans le domaine public. Le développement des LCSO a connu une montée fulgurante au cours des deux dernières décennies. Étant considérés comme un élément clé de l'écosystème logiciel à code source ouvert (ECLSO), les projets de LCSO abondent et deviennent de plus en plus essentiels et indispensables pour les systèmes utilisés par plusieurs organisations à l'échelle mondiale. L'évolution rapide de la recherche sur les ECLSO a permis de nombreuses contributions significatives. Cependant, cette croissance suscite des interrogations et interpelle la communauté des chercheurs et praticiens en ce qui concerne la santé de ces ECLSO. Qu'advierait-il s'ils se dégradent ou cessaient d'exister ? Dans le but de mieux comprendre comment ce domaine a été abordé dans littérature, nous avons effectué une revue de littérature systématique descriptive afin d'avoir un aperçu des études qui ont été faites dans ce domaine. Au total, 98 articles ont été retenus dans notre échantillon et les résultats de notre analyse de ces articles montrent que la recherche dans ce domaine est encore immature. Il n'existe pas encore de consensus quant à la définition de la santé, ni à celle de l'écosystème logiciel, ni d'outils adéquats pour appuyer l'évaluation de la santé des ECLSO. Notre étude a révélé qu'aucun auteur n'a réellement défini ce que c'est que la « santé d'un ECLSO ». Néanmoins, certains aspects de la santé ont été étudiés pour divers composants d'un ECLSO. Notre revue de littérature permet de souligner que le domaine offre de réelles opportunités pour des recherches futures.

Abstract

Open Source Software (OSS) is software whose source code is accessible and modifiable by anyone, and whose license ensures that this code remains in the public domain. The development of OSSs has skyrocketed over the past two decades. As a key component of the Open Source Software (OSS) ecosystem, OSS projects abound and become increasingly essential and indispensable for various systems used most organizations worldwide. The rapid evolution of OSS ecosystem research has resulted in many significant contributions. However, this growth raises questions and challenges the community of researchers and practitioners as far as health of these OSS ecosystem is concerned. What would happen if they deteriorated or ceased to exist? To better understand how researchers addressed this area, we conducted a descriptive systematic literature review to gain insight into the studies that have been done in this area. In total, 98 articles were selected in our sample and the results of our analysis of these articles show that research in this area is still immature. There is still no consensus on the definition of health, the software ecosystem, nor adequate tools to support the health assessment of the ELCSO. Our study revealed that no author has really defined the "health of an OSS ecosystem". Although some aspects of health have been studied for various components of an OSS ecosystem, our literature review highlights that the field offers real opportunities for future research.

Table des matières

Résumé	iii
Abstract	iv
Table des matières	v
Liste des figures	vi
Liste des tableaux.....	vii
Dédicaces.....	viii
Remerciements	ix
Introduction	1
Chapitre 1 : Revue de la littérature	9
1.1 Logiciel à code source ouvert.....	9
1.2 Écosystèmes logiciels	12
1.3 Santé des écosystèmes logiciels	14
Chapitre 2 : Méthodologie.....	18
2.1 Recherche d'articles.....	18
2.2 Sélection des articles.....	19
2.3 Extraction des données	23
2.4 Analyse des données.....	25
Chapitre 3 : Résultats	28
3.1 Analyse descriptive.....	28
3.1.1 Caractéristiques des articles analysés.....	28
3.1.2 Caractéristiques de l'étude rapportée dans chaque article.....	32
3.2 Définitions de la santé des ELCSO	35
3.3 Santé des ELCSO	36
3.3.1 Aspects de la santé	37
3.3.2 Composants de l'écosystème	41
3.3.3 Aspects de la santé des composants de l'écosystème	43
Chapitre 4 : Discussion, limites et implications	48
4.1 Discussion	48
4.2 Limites de l'étude	53
4.3 Implications.....	54
Conclusion.....	56
Bibliographie	58
Annexes.....	67

Liste des figures

Figure 1 : Étapes du processus de sélection des articles	23
Figure 2 : Nombre d'articles par année de publication.....	30
Figure 3 : Classification en fonction du type de recherche	33
Figure 4 : Nombre d'articles en fonction de l'utilisation de la source de données.....	34

Liste des tableaux

Tableau 1 : Sélection des analyses de données.....	26
Tableau 2 : Nombres d'articles par domaine.....	29
Tableau 3 : Nombres d'articles par type de publication	29
Tableau 4 : Nombre d'articles par publication et par année	31
Tableau 5 : Classification en fonction de la méthode de recherche	32
Tableau 6 : Nombre d'articles ayant utilisé une théorie ou un modèle existant	34
Tableau 7 : Définition des aspects de santé issus des articles de notre échantillon.....	37
Tableau 8 : Définition des composants de l'écosystème issus des articles de notre échantillon	41
Tableau 9 : Aspects de santé et composants d'écosystème étudiés	44

Dédicaces

A

*Mes parents, M. **NOUTSA Maurice** & Mme **TIOTSIA Rose**,
pour l'amour, les encouragements, les marques d'attention, la
persévérance et surtout pour les sacrifices consentis ;*

*Ma tendre épouse, Mme **NANJOU Judith Mimosette**, pour sa
patience*

*Mes enfants, **Yann Evrard**, **Mauricette Herla**, **Hilary** et **Moïsha**
source d'inspiration et de motivation.*

Remerciements

Rendu au terme de mon travail d'étude, il s'appert opportun d'exprimer ma profonde gratitude à tous ceux qui ont contribué, chacun à sa manière et selon sa convenance, au parachèvement de ce modeste travail qui se veut un coup d'essai pour les recherches futures.

Tout d'abord, je tiens à exprimer ma profonde gratitude au corps professoral et administratif de la Faculté des sciences de l'administration à l'Université Laval pour leur formation adéquate, plus particulièrement le corps professoral du prestigieux département de Systèmes d'information organisationnels (SIO).

Sincères remerciements et très hautes déférences à mon encadreur, professeure **Josianne MARSAN** qui a cru en moi et m'a donné l'opportunité de participer au projet *SECOHealth*. Malgré ses multiples charges, ô combien très importantes et sensibles, elle a accepté de diriger ce travail. Nonobstant son rôle d'encadreur, elle a été pour moi un guide, une conseillère, que dis-je, « une mère », pour son sens élevé des relations humaines, sa capacité d'écoute, car elle n'a jamais ménagé aucun effort pour me faciliter une insertion dans le monde professionnel et de la recherche.

Je ne manquerai pas d'exprimer ma profonde gratitude au professeur **Jean Robert KALA KAMDJOUG** de l'Université Catholique d'Afrique Centrale (UCAC)-Institut Universitaire de Yaoundé-Cameroun, qui me soutient inlassablement à mille lieux de moi et ne cesse de m'encourager à poursuivre des études doctorales.

Une pensée très particulière revient aux membres de ma famille, mes frères et sœurs, **NDIFOR CHABIFOR, Clarisse MEKATSOP, Irène DONFOUET, Raoul KADJIO, Eugénie TEMGOUA, Martial TEDONKA, Sandrine NGUIMATSIA** et **Boris DONGMO**, qui sont toujours à mes côtés pour m'encourager dans ma vie quotidienne.

J'adresse également mes remerciements au Projet *SECOHealth* financé par le Programme bilatéral de recherche collaborative Québec – Communauté française de Belgique des Fonds de recherche du Québec (FRQ) et Fonds de la Recherche Scientifique (F.R.S.-FNRS).

Mes remerciements vont également à l'endroit des membres dudit projet, plus spécialement le professeur **Mathieu TEMPLIER**, pour ses conseils éclairés et le doctorant **Patrick MAROIS**, très bon compagnon de route au grand cœur. Une pensée à **Patrick TAVERA** et **Armstrong FOUNDJEM** qui ont participé à l'exercice d'inter-rater.

Un clin d'œil spécial à **Morvan NONGNI DONFACK, Harold NGUEGANG, Wilfried BAZOMANZA, Albert NYEBEL HIOL** et **Marius KEMAYOU** pour leur soutien tout au long de mon parcours.

Je témoigne ensuite ma déférence à tous les membres du jury pour avoir accepté de juger ce modeste travail académique, qui se veut être un pas vers la recherche.

Enfin, que tous ceux qui n'ont pas été cités nommément soient aussi rassurés de mon attachement indéfectible.

Introduction

De nos jours, les organisations sont confrontées au quotidien à de nombreux défis managériaux, compte tenu notamment de l'environnement ambiant, de la recherche de la performance et de la complexité des solutions technologiques (Fosso Wamba *et al.*, 2018). Afin de faire face à l'interdépendance grandissante des différentes technologies et à la complexité accrue de l'environnement dans lequel ces organisations évoluent, elles se doivent de coopérer avec d'autres organisations (Tidd et Bessant, 2018), et être prêtes à s'adapter aux changements organisationnels (Raymond, 2018). Jadis, les projets développés par ces organisations étaient basés sur des technologies avec des modèles commerciaux « fermés », c'est-à-dire des projets qui créent de la valeur à partir des connaissances et des technologies qu'ils développent et qui exploitent principalement des ressources internes (Colombo *et al.*, 2016). La concurrence croissante et les coûts de développement des logiciels, ainsi que la diffusion croissante de l'approche de l'innovation ouverte (Chesbrough, 2007), ont récemment conduit bon nombre d'organisations à adopter des modèles « ouverts » ou « *Open Source* » en anglais (Colombo *et al.*, 2016).

Un logiciel à code source ouvert (LCSO) ou *Open Source Software* est un logiciel dont le code source est accessible et modifiable par quiconque, et dont la licence assure que ce code demeure dans le domaine public (Marsan et Paré, 2013). Cette accessibilité et cette ouverture au code source permettent aux différentes communautés d'utilisateurs, de développeurs, de sponsors et de chercheurs de développer, de redéfinir et de valider de manière indépendante ledit logiciel, en apportant des améliorations sur sa fonctionnalité, sa valeur et sa qualité tout en le rendant librement accessible à tous (Macredie et Mijinyawa, 2011). À l'origine des LCSO existe un puissant mouvement à caractère idéologique dont la survie est maintenue par des développeurs bénévoles appelés « contributeurs » (Raymond, 2001).

Le développement de LCSO est devenu un modèle viable pour la création et la distribution de logiciels, transformant ainsi l'industrie du logiciel et servant d'inspiration pour l'adoption de pratiques plus ouvertes dans plusieurs autres domaines de l'industrie (Von Krogh et Hippel, 2006). En tant que tel, le LCSO a eu et continue d'avoir un impact significatif sur la stratégie des entreprises de l'industrie du logiciel (Morgan et Finnegan, 2014), et la gestion efficace de son utilisation et de son développement est cruciale et indispensable pour le

succès et la performance des entreprises (Daniel et Stewart, 2016). De nombreux LCSO sont aujourd'hui des leaders du marché, ce qui suggère que le développement des LCSO est la clé de la croissance du secteur des logiciels (Marsan *et al.*, 2019). Le développement de LCSO est une activité sociotechnique complexe, qui nécessite l'interaction des individus impliqués (Wu et Tang, 2007). Selon Ruff (2016), les LCSO aident à créer de l'innovation sur de nombreux segments de marchés. Il rappelle que le rapport *Hype Cycle* des LCSO, publié en 2016 par *Gartner*, explique comment de nombreux départements informatiques au sein des entreprises utilisent de nos jours les LCSO non seulement pour réaliser des économies, mais de plus en plus pour innover. Fort de ce constat, nous nous situons à l'apogée d'une époque où les LCSO étaient à la traîne par rapport aux logiciels propriétaires en termes de fonctionnalité et de qualité (Ruff, 2016). Percevant les LCSO comme des technologies de pointe et innovantes dans de nombreux domaines, tels que l'infonuagique ou *DevOps*, de nombreuses entreprises démarrent de nouveaux projets de développement ou d'utilisation de LCSO plutôt que d'utiliser des logiciels propriétaires (Ruff, 2016). Ainsi, l'adoption de technologies innovantes implique de travailler en permanence sur des projets LCSO, tels que l'écosystème *OpenStack* (Ruff, 2016). Depuis quelques années, l'adhésion et l'implication des entreprises dans des projets LCSO est un fait consolidé (Macredie et Mijinyawa, 2011; Capra *et al.*, 2009). À titre d'exemple, la neuvième enquête annuelle sur l'avenir du LCSO, effectuée par *Black Duck and North Bridge*, révèle que 64% des 1300 entreprises interrogées sont activement impliquées dans des projets LCSO dans le monde entier en 2015, contre 50% en 2014 (Vaughan-Nichols, 2015). Selon le même rapport, 88% prévoient augmenter leurs contributions aux projets LCSO au cours des prochaines années (Vaughan-Nichols, 2015). Une enquête réalisée par *Black Duck and North Bridge* en 2016 auprès de 3400 professionnels des technologies de l'information (TI), parmi lesquels 70% étaient des développeurs, a permis de faire ressortir quelques faits marquants liés aux LCSO, tels que leur valeur commerciale, leur rôle de moteur de l'innovation, la prolifération des modèles commerciaux basés sur les LCSO, l'augmentation du financement privé (quadruple accroissement en cinq ans) dans les projets de développement de LCSO, le développement de meilleures pratiques de sécurité et de gestion des LCSO ainsi que l'implication accrue de communautés d'entreprises actives et innovantes dans le développement de LCSO (LeClair, 2016). Pour créer de la valeur, Leclair (2016) relève que les répondants considèrent les LCSO

comme un élément essentiel dans leur stratégie de développement logiciel et les résultats de ladite enquête révèlent que plus de 65% des répondants s'appuient sur les LCSO pour accélérer leur développement logiciel, tandis que 65% des répondants contribuent à des projets de développement de LCSO. Étant donné que les LCSO intègrent les standards ouverts, leur adoption a permis à de nombreuses organisations de réaliser des économies de l'ordre de 60 milliards de dollars par an (Ghapanchi et Aurum, 2010). Reprenant un rapport de *The Standish Group*, Ghapanchi et Aurum (2010, p. 542) souligne « ... alors qu'ils [les LCSO] ne représentent que 6% des milliards de dollars estimés en technologies de l'information budgétisées par an, ils représentent une perte réelle de 60 milliards de dollars en revenus annuels pour les éditeurs de logiciels » (traduction libre).

Tout comme Marsan et Paré (2013), Crowston *et al.* (2012) relèvent que les LCSO sont des logiciels disponibles avec leur code source, à accès libre, et autorisant la modification et la redistribution de ce code source. Ils ont connu une montée fulgurante au cours des deux dernières décennies et, par conséquent, ont modifié la façon de les percevoir, développer et déployer (Ramanathan et Iyer, 2015). Étant considéré comme un élément clé de l'écosystème logiciel, les projets de LCSO abondent et deviennent de plus en plus essentiels et indispensables pour le fonctionnement Internet, ainsi que pour les systèmes utilisés par plusieurs entreprises dans le monde (Ahmed *et al.*, 2017). Bien qu'offrant des produits et des services gratuits ou à faible coût pour répondre aux besoins critiques, les projets de LCSO offrent également à la communauté de professionnels TI un moyen efficace et indépendant de résoudre un large éventail de problèmes communs, et de développer et partager des normes existantes dans le domaine (Ahmed *et al.*, 2017). Ahmed *et al.* (2017) soulignent que l'on trouve les LCSO dans le marché des navigateurs (p.ex. *Mozilla Firefox*), des logiciels de productivité (p.ex. *Open Office*) que des systèmes d'exploitation (p.ex. *Ubuntu*, l'une des nombreuses distributions de *Linux*).

Selon Mount et Fernandes (2013), les LCSO représentent un passage du paradigme économique « propriétaire » du développement de logiciels « fermés » à un paradigme plus « ouvert et principalement collaboratif » du développement de logiciels. Ils précisent que ce phénomène a donc joué un rôle de catalyseur dans l'évolution de l'industrie du logiciel, tant en ce qui concerne les processus de développement logiciel que la distribution et l'acquisition de logiciels aux utilisateurs. Certains auteurs (p.ex. Franco-Bedoya, 2017; Ruff, 2016)

soulignent que ce phénomène des LCSO continuera d'évoluer, de prospérer et de jouer un rôle significatif dans le paysage de solutions technologiques. Ruff (2016) précise que le rapport *Hype Cycle 2016* de *Gartner* sur les LCSO explique le caractère révolu des logiciels propriétaires au profit des LCSO qui sont de plus en plus utilisés par de nombreuses entreprises dans leurs nouveaux projets afin d'innover et obtenir un avantage concurrentiel, en travaillant notamment sur des écosystèmes logiciels ouverts tels que *Hadoop* ou *OpenStack*. Par ailleurs, de par leur nature ouverte et collaborative, des LCSO peuvent se développer à partir d'autres LCSO (Dijkers *et al.*, 2018). Le succès et la croissance des LCSO sont donc importants pour le bien-être général de l'écosystème logiciel (Ahmed *et al.*, 2017) et leur adoption est devenue une tendance pour les organisations, notamment dans les pays émergents, afin d'améliorer leur performance (Zhussupova et Rahman, 2011). Les projets de LCSO ont de plus en plus tendance à être intégrés à des projets à grande échelle ou à des "écosystèmes logiciels" afin de réduire les efforts et d'accélérer l'innovation (Marsan *et al.*, 2019).

Un écosystème logiciel se définit généralement comme un grand ensemble de projets logiciels interdépendants partageant une plateforme technologique et maintenus par de grandes communautés en ligne formées de contributeurs dans ces projets (Jansen et Cusumano, 2013). Mens *et al.* (2017) le définissent comme un réseau sociotechnique constitué d'un ensemble cohérent de projets logiciels interdépendants, développés et gérés par des groupes de contributeurs ou communautés faisant partie d'une communauté plus large. À titre d'exemple, un écosystème logiciel tel que *Ubuntu* contient plus de 70 000 packages binaires développés par près de 25 000 projets de développement de LCSO (Decan *et al.*, 2016). De même, l'écosystème logiciel *npm* pour les bibliothèques *javascript* contenait plus de 300 000 packages interdépendants en 2016 (Decan *et al.*, 2016). Nous constatons aussi que de nouveaux écosystèmes d'applications mobiles envahissent tous les aspects de notre vie quotidienne (Hyrynsalmi *et al.*, 2016). Selon Manikas et Hansen (2013c), en plus d'*iOS*, l'écosystème *Android* de *Google* est un exemple frappant d'écosystème logiciel et on peut soutenir que de tels écosystèmes gagnent en importance sur le plan commercial, car, selon les auteurs, il a été estimé en 2012 plus de ventes de smartphones que d'ordinateurs personnels. Qu'advierait-il si les écosystèmes logiciels à code source ouvert (ELCSO), dont les organisations et individus sont de plus en plus dépendants au quotidien, se

dégradaient ou cessaient d'exister ? Ils sont sujets au changement à un rythme toujours croissant et, souvent, aucun responsable n'est désigné pour assurer le monitoring de leur santé, de leur survie ou de leur longévité (Mens *et al.*, 2017). De plus, les outils existants pour supporter les responsables de ELCSO sont très insuffisants et il apparaît alors important pour les chercheurs et praticiens de s'intéresser à la santé des ELCSO, afin d'en maximiser la survie et l'évolution (Mens *et al.*, 2017). C'est justement l'objectif du projet *SECOHealth* dans lequel le présent mémoire s'inscrit. Considérée comme un facteur extrêmement important, Manikas et Hansen (2013a) soulignent que la santé d'un écosystème logiciel est « la capacité de l'écosystème à durer, à rester agile et productif au fil du temps » (p.2, traduction libre). Pour Iansiti et Levien (2004a), un écosystème « sain » est celui qui offre de réelles opportunités de croissance durable non seulement à ses membres, mais également à ceux qui en dépendent. Amorim *et al.* (2017b) considèrent que l'évaluation de la santé des écosystèmes logiciels est une activité clé pour les organisations, les partenaires ou même les développeurs, car il est indispensable pour l'ensemble des parties prenantes désireuses de s'engager dans un écosystème de connaître l'état de santé réel dudit écosystème.

Problématique

Depuis longtemps, le phénomène des LCSO attire une attention particulière et un engouement auprès des chercheurs du milieu académique dans différentes disciplines allant des sciences sociales et économiques à l'informatique et l'ingénierie (Amrollahi et Khansari, 2011). Ce phénomène complexe a complètement révolutionné la manière dont les organisations et les individus créent, distribuent, acquièrent et utilisent les SI (Crowston et Wade, 2010). Bien qu'il pose également des défis aux chercheurs, le phénomène des LCSO représente un sujet de plus en plus important pour les chercheurs en SI, car il nécessite une compréhension interdisciplinaire des dynamiques techniques, économiques, juridiques et socioculturelles (Crowston et Wade, 2010). En effet, ce phénomène inclut des communautés avec une grande diversité de développeurs bénévoles et de développeurs payés par des entreprises qui s'impliquent dans ces communautés, réunis autour d'un écosystème (Daniel *et al.*, 2018).

Un écosystème est le résultat d'un équilibre délicat et dynamique entre ses composants en interaction (Ostrom, 2009). Dans un écosystème naturel, le bien-être humain est considérablement menacé par des dommages croissants ou des pertes de ressources naturelles

(Ostrom, 2009). Ces ressources et leurs utilisateurs sont intégrés à des systèmes sociaux et écologiques complexes qui interagissent avec leur environnement (Ostrom 2009). Par analogie à l'écosystème naturel, l'on pourrait souligner que le bien-être ou la survie d'un écosystème logiciel dépend de chacun de ses composants.

Afin de développer des méthodes favorisant la durabilité des écosystèmes logiciels et préservant ainsi les moyens de survie de leurs composants, il s'avère nécessaire de développer des cadres qui offriraient une vue d'ensemble adéquate des problèmes, des causes associées et des effets qui en résultent (Marsan *et al.*, 2019; Marois *et al.*, 2018a, 2018b; Nassl et Löffler, 2015). Ces cadres constitueraient donc des outils utiles pour créer un vocabulaire commun de concepts et de termes liés à la santé des écosystèmes logiciels (Nassl et Löffler, 2015).

De nombreux chercheurs ont commencé à étudier les facteurs qui peuvent avoir un impact sur la santé des écosystèmes logiciels (p.ex. Jansen, 2014), leur succès (p.ex. Sen *et al.*, 2012) et leur survie (p.ex. Wang, 2012). Par ailleurs, Mens *et al.* (2017) relèvent qu'il n'existe pas de consensus et très peu d'outils pour la définition et le suivi de tels facteurs au niveau des écosystèmes logiciels en général, plus précisément au niveau des projets individuels. Le constat qui se dégage est que plusieurs auteurs s'intéressent principalement aux facteurs au niveau du projet ou de la communauté plutôt qu'au niveau de l'écosystème logiciel (Germonprez *et al.*, 2018). Ces derniers insistent sur le fait que la recherche devrait désormais prendre en compte des « écosystèmes complexes et larges » dans lesquels se situent les projets LCSO, car les études se focalisant uniquement sur les projets ou communautés ne peuvent pas apporter de résultats substantiels au niveau des ELCSO. Mens *et al.* (2017) expliquent cependant que l'une des raisons principales de ce non-consensus est dû au fait que la dynamique interne et les interactions d'un écosystème logiciel sont soumises à de fréquents changements, ou à l'évolution due à une combinaison d'événements internes et externes, tels que des changements technologiques ou environnementaux, ou l'apparition de systèmes concurrents à l'instar de *iOS* et *Android*. Selon Adams *et al.* (2016), la difficulté résiderait dans le fait que les ELCSO ne disposent pas en leur sein d'un management centralisé qui favoriserait la surveillance de la santé de l'écosystème.

Fort de ce constat et en dépit de l'intérêt que suscitent les LCSO et les écosystèmes logiciels, nous comprenons dès lors la raison pour laquelle peu d'efforts a été déployé jusqu'à date pour

organiser la majeure partie de la littérature relative à la santé des ELCSO, de manière à faciliter la recherche et à permettre une meilleure compréhension des aspects de « santé » des ELCSO. Manikas et Hansen (2013c) soulignent la contribution potentielle qu'une cartographie documentaire de la santé des LCSO pourrait apporter, notamment sur les aspects moins étudiés. Dans leurs travaux, ils se focalisent uniquement sur le niveau du LCSO. Or, un LCSO n'est qu'un composant d'un ELCSO où d'autres composants sont tout aussi importants. Considérant le nombre croissant d'ELCSO desquels les organisations et individus sont de plus en plus dépendants, il serait pertinent d'étendre leurs travaux en explorant la totalité de l'ELCSO.

Objectif et questions de recherche

Le présent mémoire fournit un aperçu des connaissances scientifiques sur la santé des ELCSO. L'objectif est de structurer et d'analyser une partie de la littérature existante sur la santé des ELCSO, soit les articles publiés dans les meilleures revues en Système d'information (SI) et les articles ayant cité les articles phares en SI concernant les LCSO. Par conséquent, nos questions de notre recherche sont les suivantes :

- Q1. Comment la santé des ELCSO est-elle définie dans la littérature ?
- Q2. Quels aspects de la santé des ELCSO, incluant leurs composants, ont été étudiés et, parmi ceux-ci, lesquels ont fait l'objet du plus grand nombre d'études ?

Nous répondons ainsi aux appels à la recherche de la part de Jansen (2014), Mens *et al.* (2017) et Marois *et al.* (2018a) concernant la santé des ELCSO et de Manikas et Hansen (2013a) concernant la santé des écosystèmes logiciels en général.

Intérêt de l'étude

Sur le plan théorique, nous espérons que les résultats de ce travail de revue de littérature systématique permettront d'organiser, de classier et de structurer la majeure partie de la littérature existante sur la santé des écosystèmes logiciels. Sur le plan pratique, nous pensons que la classification et la structuration des aspects de santé des ELCSO seraient une valeur ajoutée pour les parties prenantes directement impliquées dans les projets de développement de LCSO ou dans les ELCSO. Les résultats permettraient aux gestionnaires de communautés ou de ELCSO et aux contributeurs dans ces communautés ou ELCSO de mieux gérer et contrôler leurs écosystèmes en identifiant, grâce aux indicateurs de santé, des problèmes de

santé qui pourraient impacter sur la survie et la croissance dudit écosystème. Sur le plan des recherches futures, nous espérons que ce travail sera une boussole qui orientera les travaux futurs dans le domaine de la santé des ELCSO vers l'exploration de pistes non encore empruntées.

Ce mémoire s'inscrit dans le cadre du projet interdisciplinaire et collaboratif *SECOHealth* (subvention R.60.04.18.F du FRQ-FNRS et crédit de recherche JRS2323.16 « Analysis of Software Project Survival » du FNRS), impliquant des équipes de recherche de Polytechnique Montréal (Canada), de l'Université de Mons (Belgique) et de l'Université Laval (Canada), avec la participation de collaborateurs de Toulouse Business School (France) et de Politecnico di Milano (Italie). Le projet *SECOHealth* a pour objectif de contribuer à la recherche et à la pratique en proposant une méthodologie scientifique interdisciplinaire validée ainsi qu'un catalogue de directives et d'outils de recommandation pour améliorer la santé des écosystèmes logiciels (Mens *et al.*, 2017).

Le présent mémoire est structuré en quatre chapitres. Le chapitre premier présente les fondements conceptuels en lien avec la santé des ELCSO. Le chapitre deuxième présente la méthodologie de recherche utilisée pour mener la revue de littérature systématique. Le chapitre troisième présente les différents résultats obtenus. Enfin, la discussion des résultats présentés au chapitre précédent, les limites et les implications meublent le chapitre quatrième.

Chapitre 1 : Fondements conceptuels

Ce chapitre se propose de présenter, à partir de la littérature, les définitions des concepts clés en lien avec la santé des ELCSO. Afin de mieux cerner notre étude, il appert important d'analyser ces concepts qui nous éclaireront dans le cheminement de notre travail de recherche. Dans un premier temps, nous définirons les concepts clés liés à l'écosystème logiciel. Ensuite nous allons procéder à une présentation succincte du concept de santé des écosystèmes logiciels. Enfin, nous allons présenter sommairement quelques revues de littérature dans le domaine des systèmes d'information.

1.1 Logiciel à code source ouvert

Bien que les chercheurs ne parviennent pas à être unanimes quant à une définition claire, précise et définitive (Crowston *et al.*, 2012), l'*Open Source Initiative* (www.opensource.org) énonce dix caractéristiques permettant de définir le LCSO parmi lesquelles les trois caractéristiques principales sont : la distribution gratuite du logiciel, la disponibilité du code source et les droits pour la modification de ce code (Open Source Initiative, 2007). Selon Marsan et Paré (2013), les LCSO sont des logiciels dont le code source est accessible et modifiable par quiconque, et dont la licence assure que ce code demeure dans le domaine public. Leur utilisation est donc régie par une charte des droits d'utilisateurs qui définit tous les droits spécifiques accordés par une licence logicielle appropriée (Saeed *et al.*, 2018). Depuis quelques années, leur adoption est devenue de plus en plus fréquente et leur nécessité s'est révélée dans les organisations (Marois *et al.*, 2018a). De nos jours, l'adoption des LCSO par les organisations est devenue un besoin stratégique dans de nombreux domaines d'application (Franco-Bedoya *et al.*, 2017). Cela a changé (et continue de changer) la façon dont les organisations acquièrent, utilisent, commercialisent et développent les logiciels (Angeren *et al.*, 2011).

Ainsi, les projets de développement de LCSO connaissent une popularité croissante depuis plus de trois décennies (Saeed, Fatima, et Iqbal, 2018). Le développement des LCSO a été reconnu comme un sujet de recherche dans le domaine des SI (p.ex. Niederman *et al.*, 2006; Fitzgerald, 2006), car il s'agit d'un phénomène qui a eu un impact significatif au cours de la dernière décennie, non seulement dans l'industrie du logiciel, mais également au sein des

organisations des secteurs public et privé (Hauge *et al.*, 2010). Au cours de cette dernière décennie, les travaux portant sur les LCSO ont considérablement évolué (Franco-Bedoya *et al.*, 2017), car il s'agit d'un phénomène récent (Aksulu et Wade, 2010), émergent (Fitzgerald, 2006) et ayant reçu une attention considérable dans la littérature scientifique, ainsi que chez les praticiens (Marsan *et al.*, 2012).

De plus en plus, des millions d'utilisateurs dépendent de systèmes LCSO tels que *Linux*, *Firefox* ou *Android*, et Internet, le plus vaste réseau mondial, dépend fortement d'outils LCSO comme *Apache* (Crowston et Wade, 2010). Le système d'exploitation *Linux* et le navigateur web de *Netscape* sont des exemples de réussite de projets de développement de LCSO (Perens, 1999). Les résultats d'une enquête réalisée par la communauté *Stack Overflow* en 2018 indiquent en effet que 48,3% des répondants utilisent *Linux* comme plateforme de développement tandis 76,5% la considèrent comme la plus appréciée. Une autre enquête portant sur les principales plateformes pour smartphones suggère que la plateforme LCSO *Android* développée par *Google* a vu la proportion de téléphones qui exploitaient *Linux* comme principal système, augmenter de 48,7% en quatre ans et demi, passant de 3,8% en novembre 2009 à 52,5% en avril 2014 (comScore MobiLens Service, 2010, 2014).

Étant donné que l'environnement de développement des LCSO est dispersé géographiquement à l'échelle mondiale, les interactions des différentes parties prenantes impliquées sont décentralisées et dynamiques (Saeed *et al.*, 2018). Par parties prenantes, il faut comprendre l'ensemble des contributeurs, généralement constitués d'un groupe de développeurs motivés, soutenu par un ensemble d'utilisateurs actifs organisés au sein d'une communauté (Manikas et Hansen, 2013a).

La motivation pour la création de projets de LCSO est souvent l'émanation d'un individu ou un groupe d'individus ayant un besoin spécifique à satisfaire (Franco-Bedoya *et al.*, 2017).et généralement réunis au sein des communautés (Roberts *et al.*, 2006). C'est le cas notamment d'*Apache* et des projets *Mozilla* dont le développement est contrôlé par des groupes de développeurs principaux et soutenu par de grandes communautés de contributeurs (Mockus *et al.*, 2002). Une telle communauté peut être considérée comme un réseau collaboratif de développeurs travaillant ensemble sur différentes équipes de projets (Hahn *et al.*, 2008). Les développeurs participent à une ou plusieurs équipes projets au sein d'une communauté (Xu *et al.*, 2006) et développent ainsi des liens avec les autres membres de l'équipe ou d'une autre

équipe appartenant à d'autres communautés (Hahn *et al.*, 2008). À cet effet, l'on associerait les LCSO à de nouvelles formes de socialisation, de pratiques de développement des projets, de réseautage communautaire dynamique ou de structure organisationnelle (Scacchi, 2007). En tant qu'acteurs clés des projets LCSO (Franco-Bedoya *et al.*, 2017), les communautés LCSO garantissent le développement, le support et la maintenance desdits logiciels (Foulonneau *et al.*, 2013). Dans leurs travaux, Franco-Bedoya *et al.* (2017) soulignent qu'une communauté de LCSO implique non seulement des individus mais également des organisations volontaires qui produisent et consomment des composants de LCSO. Dans la plupart des cas, les développeurs sont les premiers utilisateurs et consommateurs du LCSO produit (Fitzgerald, 2006; Scacchi, 2007).

D'autres récentes études ont permis d'examiner les rôles joués par les membres des communautés LCSO (Crowston et Howison, 2006; Xu *et al.*, 2006), car pour une meilleure compréhension de ces rôles, il est indispensable de maîtriser les complexités et les subtilités de l'innovation dans lesdits LCSO (von Krogh et von Hippel 2006). Franco-Bedoya *et al.* (2017) relèvent les nombreux rôles joués par les volontaires dans une communauté de LCSO avec différents niveaux de participation, à l'instar des utilisateurs, des réviseurs, des contributeurs, des administrateurs, des leaders, des partenaires et des développeurs (Squire et Williams, 2012). Comme l'ont noté plusieurs auteurs dans la littérature, la communauté des LCSO dans son ensemble forme un système complexe très particulier, doté d'une dynamique interne à court terme et d'une dynamique d'évolution à long terme (Franco-Bedoya *et al.*, 2017).

Une communauté fait partie intégrante d'un écosystème, constitué d'une ou de plusieurs communautés LCSO, lesquelles sont parfois conceptualisées comme des organismes vivants ou des sous-systèmes vivant dans un environnement précis (Carillo et Marsan, 2016). À cet effet, ces sous-systèmes (ou organes) constituant une organisation (Morgan, 2017) peuvent également être extrapolés en tant qu'éléments constitutifs de communautés de LCSO (ou de sous-communautés), considérées comme une espèce parmi tant d'autres entités sociales (Carillo et Marsan, 2016).

Selon Carillo et Marsan (2016), un « organisme » LCSO (c'est-à-dire une communauté de développement de LCSO) possède cinq organes, chacun décomposé en parties plus petites : artefacts informatiques, utilisateurs et contributeurs, gouvernance du projet, gestion du projet

et structure du projet. Tout d'abord, l'organe « utilisateurs et contributeurs » représente la partie sociale de l'écosystème logiciel et peut être décomposé en sous-groupes tels que les utilisateurs de base (par opposition aux rapporteurs de bogues), les nouveaux contributeurs du projet (par opposition aux contributeurs réguliers et permanents), les principaux contributeurs (par opposition aux périphériques) ou les contributeurs au code (par opposition aux non-codeurs, p.ex. les producteurs de la documentation accompagnant le logiciel) (Carillo et Marsan, 2016). L'organe « artefacts informatiques » est composé de plusieurs sous-systèmes, y compris le logiciel en cours de développement et ses modules, ainsi que des divers outils informatiques nécessaires au fonctionnement du processus de développement et de la communauté dans son ensemble (Carillo et Marsan, 2016). Les sous-systèmes de l'organe « gouvernance du projet » constituent tous les moyens stratégiques (portée du projet, les objectifs du projet et la résolution des conflits) permettant de diriger, contrôler et coordonner des individus et des groupes qui contribuent de manière collaborative au projet (Carillo et Marsan, 2016). Quant à l'organe « gestion de projet », il est composé de tous les moyens utiles pour gérer activement les tâches du projet et les développeurs (Carillo et Marsan, 2016). Finalement, l'organe « structure du projet » peut être divisé en différents sous-systèmes tels que la taille de l'équipe d'administrateurs du projet, le nombre de contributeurs et la taille de la communauté d'utilisateurs, aussi bien que le positionnement hiérarchique (Carillo et Marsan, 2016). Selon ces auteurs, les projets logiciels et les communautés qui les animent sont interdépendants. Cette interdépendance s'effectue au sein d'un environnement externe avec lequel des échanges permanents s'opèrent (Carillo *et al.*, 2017).

1.2 Écosystèmes logiciels

Les écosystèmes logiciels sont apparus au cours des dernières années comme un moyen novateur de comprendre les relations entre les projets logiciels, les produits qu'ils développent, les communautés et les organisations impliquées dans ces projets (Franco-Bedoya, 2014). Depuis l'avènement du concept d'écosystèmes logiciels (en anglais, *software ecosystem*) introduit en 2003 par Messerschmitt et Szyperski, on note que la majorité des études qui ont été menées dans la littérature sont spécialisées en génie logiciel (GL) (en anglais, *Software Engineering*) (p.ex. Jansen *et al.*, 2012; Jansen et Cusumano, 2013;

Manikas et Hansen, 2013c). En mettant à jour la revue de littérature systématique sur les écosystèmes logiciels de Manikas et Hansen (2013c), Manikas (2016) indique que la recherche sur ce concept a évolué en dehors des frontières des définitions existantes.

La première définition des écosystèmes logiciels a été proposée par Yu *et al.* (2007) qui les considèrent comme une collection de produits logiciels. Jansen *et al.* (2009b) considèrent les écosystèmes logiciels comme « un ensemble d'entreprises fonctionnant comme une unité et interagissant avec un marché partagé de logiciels et de services, ainsi que les relations entre ces entreprises. Ces relations reposent souvent sur une plate-forme technologique ou un marché commun et fonctionnent par le biais d'échanges d'informations, de ressources et d'artefacts » (p.187, traduction libre). D'après cette définition, il semble évident qu'on note l'existence d'une relation pertinente entre l'écosystème logiciel et la conceptualisation de l'écosystème d'entreprise (Teixeira et Hyrnsalmi, 2017). Lungu *et al.* (2010a) proposent une autre définition qui a été largement adoptée par les chercheurs. Ils considèrent un écosystème logiciel comme un groupe de projets logiciels développés et évoluant dans un environnement commun. Pour sa part, Manikas (2016) propose la définition suivante: « l'interaction logiciel et acteur en relation avec une infrastructure technologique commune, qui se traduit par un ensemble de contributions et qui influence directement ou indirectement l'écosystème » (p.93, traduction libre).

En rappelant que les concepts de LCSO et d'écosystème logiciel sont deux domaines de recherche consolidés en GL, Franco-Bedoya *et al.* (2017) constatent l'émergence du second concept au point de devenir un nouveau paradigme qui permettrait de mieux comprendre la dynamique et l'hétérogénéité dans le développement logiciel collaboratif en réseau. C'est ainsi qu'ils s'intéressent au concept d'*Écosystème logiciel à code source ouvert (ELCSO)* (en anglais, *Open source software ecosystem*), car ils ont pu observer que les études qui associent conjointement ces deux concepts sont rares. Pour ce faire, en faisant une jointure des définitions d'écosystème d'entreprise (*Business Ecosystem – BECO*), d'écosystème d'entreprise digitale (*Digital Business Ecosystem – DBECO*) et d'écosystème logiciel (*Software Ecosystem – SECO*) tirés dans la littérature, ils parviennent à opérer des différences entre les ELCSO et les autres types d'écosystèmes logiciels. C'est ainsi que Franco-Bedoya *et al.* (2017) arrivent à construire une définition générale de ELCSO: « un écosystème logiciel placé dans un environnement hétérogène, dont la frontière est un ensemble d'acteurs

de niche et dont le joueur clé est une communauté de LCSO autour d'un ensemble de projets sur une plate-forme ouverte commune » (p.179, traduction libre).

Étant donné qu'un écosystème logiciel est constitué de plusieurs projets interreliés (Carillo, Marsan, et Negoita, 2017), sa survie dépendra intrinsèquement du succès desdits projets. De plus, l'état du système ou de ses composants est matérialisé par leur « santé » (Amorim *et al.*, 2017a).

1.3 Santé des écosystèmes logiciels

Dans la littérature existante, plusieurs articles abordent le concept de « santé » sous des angles différents (Amorim *et al.*, 2017a). Sur la base de ces travaux, les auteurs ont pu identifier des facteurs qui ont un impact sur la santé des écosystèmes logiciels (par ex. Jansen, 2014), leur succès (par ex. Sen *et al.*, 2012), leur survie (par ex. Wang, 2012), et la santé des projets logiciels qui les composent (Amrit et Van Hillegersberg, 2010). De nos jours, de nombreux auteurs s'intéressent à la santé des écosystèmes logiciels de manière générale (Amorim *et al.*, 2017a) et spécifiquement à la santé des ELCSO (Jansen *et al.*, 2012; Jansen, 2014; Marois *et al.*, 2018a). Selon Manikas et Hansen (2013c), la garantie du bon fonctionnement d'un écosystème logiciel repose sur la nécessité de l'introduction de mesures adéquates qui offrent une vue d'ensemble de l'état de santé dudit écosystème logiciel. Selon eux, le concept de santé a été introduit par Iansiti et Levien (2004a) qui cherchaient une manière de mesurer la performance d'un BECO. En effet, Iansiti et Levien (2004a) ont mesuré le degré avec lequel un écosystème [au sens large du terme] offre de réelles opportunités de croissance durable, non seulement à ses membres, mais également à ceux qui en dépendent. Les approches d'évaluation de la santé des écosystèmes logiciels sont assez diverses et, au regard de la littérature, on constate que bon nombre d'auteurs se focalisent notamment sur la mesure des indicateurs de santé qui favoriseraient une bonne performance de l'ensemble de l'écosystème. S'inspirant de la métaphore de la santé d'un organisme vivant suggérée par Carillo *et al.* (2017), Marois *et al.* (2018, p.2) définissent la santé des ELCSO comme « un état de fonctionnement normal et continu des projets constituant l'écosystème logiciel ».

L'étude de cas menée par Gamalielsson, Lundell, et Mattsson (2011) a porté sur l'évaluation de la santé des écosystèmes sur deux projets de LCSO au moyen d'une analyse quantitative

sur des données accessibles au public et provenant de ces projets. Selon eux, la santé des écosystèmes est une condition préalable essentielle pour une chaîne d'outils LCSO durable à long terme qui peut faciliter la prise de décision stratégique en vue de son éventuelle adoption dans le contexte d'une entreprise. Ils ont établi le degré d'activités des développeurs et utilisateurs dans les deux écosystèmes, et ont identifié l'influence organisationnelle pour ces deux écosystèmes.

Pour Manikas et Hansen (2013a), la santé d'un écosystème logiciel est un indicateur de la qualité de son fonctionnement. La mesure de la santé d'un écosystème logiciel permet d'indiquer les causes plausibles, les problèmes de santé qui devraient être résolus dans l'écosystème et les zones à améliorer (Marois *et al.*, 2018a). Par ailleurs, Manikas et Hansen (2013a) relèvent un manque d'un moyen efficace pour mesurer et évaluer la santé des écosystèmes logiciels. Dans leurs travaux, ils ont passé en revue à la fois la littérature sur le concept de santé des écosystèmes logiciels ainsi que la littérature qui a inspiré la littérature sur la santé des écosystèmes logiciels. S'appuyant notamment sur 23 articles identifiés, les résultats de leurs travaux leur ont permis d'identifier que la santé des écosystèmes logiciels tire sa source de la santé des BECO, tout en étant également influencés par les théories des écosystèmes naturels et de LCSO. Ils ont également proposé un cadre conceptuel pour définir et mesurer la santé des écosystèmes logiciels.

S'agissant d'écosystèmes logiciels, l'essentiel des revues de littérature ont été menées principalement en GL (p.ex. Franco-Bedoya *et al.*, 2017; Manikas, 2016; Manikas et Hansen, 2013c) contrairement en SI où les revues de littérature sur les écosystèmes logiciels sont rares. La recherche sur la santé des ELCSO a récemment gagné en popularité bien que la littérature soit encore limitée. À notre connaissance, aucune revue de littérature systématique sur la santé des ECLSO existe. Les revues de littérature communément citées comme pertinentes comprennent le développement d'un cadre de classification de la recherche en LCSO (Nelson, Sen, et Subramaniam, 2006), une taxonomie et un cadre de référence du domaine de la recherche en LCSO (Aksulu et Wade, 2010), l'évaluation de l'état des connaissances dans le développement de LCSO (Crowston *et al.*, 2012) et une revue systématique de la recherche sur les écosystèmes logiciels dans le domaine du GL (Manikas et Hansen, 2013c). Les travaux antérieurs dans ce domaine portent généralement sur des projets de LCSO bien connus, plutôt que sur des ELCSO. Manikas et Hansen (2013c)

soulignent, d'une part, qu'il existe très peu de modèles analytiques d'écosystèmes logiciels et, d'autre part, le manque de consensus sur ce qui constitue effectivement un écosystème logiciel. Ils relèvent que très peu de recherches ont été menées dans le contexte d'écosystèmes réels. L'absence d'un échantillon empirique généralisable pourrait continuer à entraîner des discordances sur une vue globale de la santé des ELCSO.

Plus récemment, d'autres études ont commencé à émerger. Martinez-Torres et Diaz-Fernandez (2014) se sont intéressés à l'étude des communautés virtuelles de LCSO. Ils ont passé en revue les différents domaines et sujets de recherche liés aux communautés, tels que l'intelligence collective, leur structure et succès, leur motivation, le mécanisme de partage des connaissances, l'innovation et l'apprentissage. Margan et Candrljic (2015) ont présenté un aperçu de l'état de l'art de la recherche sur les LCSO dans différents domaines et disciplines des sciences sociales et de l'information. Ils ont relevé certains facteurs clés de succès qui ont contribué à l'essor du LCSO, notamment dans le domaine du GL et de l'économie. En procédant à une revue de littérature systématique sur les LCSO, Brandao Gomes da Silva *et al.* (2017) ont relevé neuf avantages qui caractérisent les approches de production de logiciels dans les projets de LCSO; quatre défis à relever; trois possibilités d'implémentation et deux motivations principales en faveur de l'adoption de nouveaux produits livrés; et enfin quatre principales stratégies pour les implémenter. Ainsi, leur étude a permis de fournir une meilleure compréhension structurée des approches de production de logiciels dans le contexte de projets de LCSO.

Malgré les progrès récents, à notre connaissance, aucune tentative, à l'exception de celle de Amorim *et al.* (2017b), n'a été menée pour rechercher systématiquement comment la santé des ECLSO est définie, a été évaluée, quels en sont les aspects qui sont les plus étudiés et quelles approches sont disponibles pour les étudier. Dans le domaine du GL, Amorim *et al.* (2017b) ont procédé à une revue systématique de la littérature afin d'examiner les méthodes d'évaluation de la santé des écosystèmes logiciels. Les résultats ont permis d'identifier 6 concepts différents de santé des écosystèmes logiciels, 211 indicateurs de santé, 4 domaines clés traités par la recherche, 19 pratiques de gouvernance et 9 outils de support utilisés par les études.

Bien que le concept de « santé » des écosystèmes ait été introduit en 2002, c'est tout récemment que ce sujet de recherche a attiré l'attention de la communauté des chercheurs

(Amorim *et al.*, 2017b). Comme nous pouvons le constater, les revues de littérature se sont davantage concentrées sur les LCSO et leurs communautés, plutôt que sur les ELCSO. Fort de ce constat et au regard de l'émergence de l'étude de la santé des écosystèmes de manière générale (Manikas et Hansen, 2013a; Amorim *et al.*, 2017b), l'objectif de notre recherche est d'évaluer l'état des connaissances sur la santé des ELCSO. Une revue de littérature systématique de type *descriptive* est appropriée à notre objectif, car elle permet non seulement de déterminer, par des analyses de fréquence, la couverture du domaine de recherche, mais aussi d'évaluer la taille et la portée de la littérature disponible sur un sujet et d'informer les chercheurs sur les axes de recherches futures (King et He, 2005; Templier et Paré, 2015, 2018).

Chapitre 2 : Méthodologie

La présente section fournit une brève description de la méthodologie appliquée qui nous a guidé dans le processus de collecte et d'analyse de données. Contrairement à un processus de revue non structuré, une revue de littérature dite « systématique » (Kitchenham et Charters, 2007; Templier et Paré, 2015, 2018; Paré *et al.*, 2015) réduit les biais et suit une séquence méthodologique précise et rigoureuse d'étapes de la revue de la littérature couverte (Brandao Gomes da Silva *et al.*, 2017). Les revues de littérature dite « systématique » reposent sur des protocoles d'examen bien définis et évalués pour extraire, analyser et documenter les résultats.

L'approche méthodologique adoptée pour cette étude est une revue de littérature systématique descriptive. Nous nous sommes appuyé sur les lignes directrices de Templier et Paré (2015, 2018) pour guider et évaluer les revues de littérature systématiques. Après avoir procédé à l'identification des sources d'information et des études pertinentes pour répondre à notre question de recherche, la sélection et l'analyse des articles ont été faites en suivant les recommandations associées à la réalisation d'une revue systématique pour classifier et structurer le domaine étudié, suggérées par Petersen *et al.* (2015).

Dans les sections suivantes, nous décrivons le protocole de notre revue de littérature. D'abord, nous présentons le processus de recherche des articles dans la base de données *Web of Science* et dans les huit (8) meilleures revues du domaine SI (*Basket of 8*). Ensuite, nous déroulons les quatre étapes consécutives pour la sélection des articles. Enfin, nous faisons mention de la phase d'extraction et d'analyse des données.

2.1 Recherche d'articles

La recherche de la littérature a été principalement faite dans la base de données *Web of science* et dans les huit (8) meilleures revues *AIS Senior Scholars' Basket of Journals*¹ (ou

¹ Selon le site de l'association pour les SI (en anglais, *AIS*), il s'agit d'un Collège de « chercheurs Senior » composé d'universitaires expérimentés en SI qui ont été rédacteurs en chef des revues répertoriées dans le *Basket of 8*, d'actuels et d'anciens présidents de l'*International Conference on Information Systems (ICIS – Conférence internationale sur les SI)*, d'actuels et anciens présidents de l'*AIS*, ainsi que tous les lauréats des « Leo Awards » et les boursiers *AIS*. Ces revues sont par ordre alphabétique : *European Journal of Information Systems*, *Information Systems Journal*, *Information Systems Research*, *Journal of AIS*, *Journal of Information Technology*, *Journal of MIS*, *Journal of Strategic Information Systems*, *MIS Quarterly*.
<https://aisnet.org/general/custom.asp?page=SeniorScholarBasket>

Basket of 8). Elles représentent, selon les bibliothécaires-conseils de la bibliothèque de l'Université Laval, les bases de données majeures pour la recherche d'articles en périodiques spécialisés en administration des affaires. Notre revue de littérature s'est penchée sur les articles publiés entre 2006 et 2018. L'année 2006 a été choisie à cause de l'article phare de Fitzgerald publié cette année-là dans *MIS Quarterly*, la revue la plus reconnue du domaine SI. Cet article, cité 289 fois selon la base de données *Web of Science*, marque la période où le phénomène des LCSO s'est transformé. Il s'agit du premier article en SI à parler explicitement de la transformation de l'*open source* en des projets qui impliquent non seulement des développeurs bénévoles mais de plus en plus d'entreprises qui souhaitent tirer avantage de l'*open source*. Fitzgerald (2006) propose le concept de « OSS 2.0 », c'est-à-dire de LCSO commercialement plus viables que ceux développés auparavant. Comme le souligne Fitzgerald (2006), « les avantages de l'*open source* sur le réseau résultent de la taille de la communauté et de l'écosystème. Ainsi, un réseau de parties intéressées aux capacités complémentaires peut former un écosystème pour offrir un produit et un service professionnels » (p.593, traduction libre). L'auteur précise que le développement de LCSO a été abondamment étudié par les chercheurs, notamment les caractéristiques des projets et la motivation des développeurs. Bien qu'utiles, ces recherches se focalisent sur des aspects individuels ou au niveau d'un projet spécifique, alors qu'il appert intéressant d'avoir une vue plus large du phénomène de l'*open source*, en particulier sur le phénomène « OSS 2.0 » (Fitzgerald, 2006). S'appuyant sur les travaux de Fitzgerald (2006), Iivari (2010) rappelle que le développement de LCSO a été reconnu comme un sujet de recherche actuel dans la recherche en SI.

2.2 Sélection des articles

La sélection d'articles s'est déroulée suivant un protocole de recherche (voir annexe1) conçu par l'équipe de recherche du projet international *SECOHealth* sur la santé des ECLSO dans lequel s'inscrit ce mémoire, et s'est articulée en quatre étapes :

1. Nous avons sélectionné des articles phares (en anglais, *importants papers*) qui doivent servir de point de départ pour conduire la revue de la littérature. Quinze (15) articles phares ont été présélectionnés en s'appuyant sur la connaissance de la littérature scientifique en SI et en sciences de la gestion plus généralement par les

experts du projet *SECOHealth*. Pour chaque article phare, nous avons dressé un tableau récapitulatif du nombre de fois qu'il a été cité par année (voir annexe 2), selon la base de données *Web of science*. En effet, cette base de données permet de générer un graphique du nombre annuel de sources qui citent un article donné. Nous avons donc procédé au calcul de la moyenne annuelle de sources ayant cité chaque article phare depuis son année de publication. Celle-ci variait de 1.33 à 17.50. Ce tableau nous a permis de sélectionner uniquement les articles phares qui ont eu une moyenne supérieure ou égale à 4,5. Ceci nous a permis de retenir définitivement huit (8) articles phares parmi les quinze ;

2. Nous avons effectué la recherche des articles citant les articles phares (*forward search* en anglais) sélectionnés à l'étape 1. Pour chaque article phare, nous avons considéré tous les articles l'ayant cité, selon *Web of science*. Tous ces articles ont été téléversés dans l'outil *Zotero* afin de constituer notre base de données. Après avoir extrait les doublons, nous avons procédé à une lecture rapide (*quick read* en anglais) du titre et du résumé de tous les articles en appliquant nos critères d'inclusion/exclusion. Les critères d'exclusion sont les suivants :
 - a. articles n'étant pas écrits en anglais;
 - b. éditoriaux;
 - c. revues de littérature;
 - d. indisponibilité du texte intégral

Les critères d'inclusion sont les suivants :

- a. articles de conférence ou de revue scientifique revus par les pairs;
- b. articles mentionnant au moins un aspect de la santé du LCSO ou de l'ELCSO.

Au début de la lecture rapide, nous avons effectué un test de fiabilité intercodeurs de notre processus d'inclusion/exclusion par le calcul du coefficient d'agrément ou Kappa de Cohen (1960). Ayant déjà procédé à la lecture rapide d'une centaine d'articles, il était question de s'assurer de la fiabilité du processus d'inclusion/exclusion. Tel que suggéré par Cohen (1988) et Faul *et al.* (2007), nous avons sélectionné de façon aléatoire 20% des articles que nous avons lus (parmi lesquels 30% étaient des articles que nous avons préalablement exclus après une lecture rapide) que nous avons transmis, accompagnés des critères d'inclusion/exclusion, à deux personnes (codeurs) qui ont fait l'exercice

d'inclusion/exclusion, de la même manière que le codeur principal et auteur du présent mémoire. Ces deux personnes sont respectivement étudiant à la maîtrise recherche en Systèmes d'information organisationnels (SIO) à l'Université Laval et étudiant au doctorat en GL à Polytechnique Montréal. Les résultats (voir annexe 3) dudit test démontrent une assez bonne force d'agrément (Landis et Koch, 1977) de 0,798 entre l'auteur et l'étudiant à la maîtrise, alors que celle avec l'étudiant au doctorat est plus faible, soit 0,579. Ceci peut s'expliquer par le fait que l'étudiant à la maîtrise évolue dans une faculté des sciences de l'administration tandis que l'étudiant au doctorat évolue dans une école d'ingénierie et est donc moins sensibilisé aux aspects sociaux et stratégiques des ELCSO. En effet, les éléments de désagrément avec l'étudiant au doctorat représentaient surtout les aspects sociaux évoqués dans les articles, alors que ce dernier a un background technique. Les désagréments avec l'étudiant de maîtrise ont été résolus de la manière suivante : une analyse manuelle a été réalisée à la suite d'une rencontre-atelier à l'effet non seulement de confirmer la corrélation des désagréments sur deux articles dont l'un était rejeté et l'autre accepté par le codeur, mais également d'arriver à une compréhension commune. À l'issue de cet exercice, nous nous sommes accordés sur l'article qui avait été rejeté initialement en l'acceptant au final ;

3. Nous avons procédé à une recherche par mots-clés dans les huit (8) meilleures revues scientifiques (*Basket of 8*) en SI (Lowry *et al.*, 2014) de tous les articles publiés entre 2006, année de la publication de l'article phare de Fitzgerald dans *MIS Quarterly*, et 2018. Cela nous a permis de collecter des articles plus spécifiques aux SI afin de s'assurer qu'aucun article pertinent du domaine SI n'a été omis. Cette recherche s'est faite exclusivement dans la base de données *ABI*, à l'exception de la revue *The Journal of Strategic Information Systems* qui s'est faite dans *Science Direct*, car cette revue n'est pas disponible dans *ABI*. Les critères d'inclusion/exclusion utilisées ont été identiques à ceux de l'étape précédente. La chaîne de mots-clés utilisée pour réaliser la recherche d'articles sur le texte intégral était la suivante :

“open source” OR “free software” OR “libre software” OR “OSS”
OR “FLOSS” OR “F/LOSS” OR “FOSS” OR “F/OSS”

Tous les articles obtenus à cette étape ont été également téléversés dans l'outil de gestion des références *Zotero*. Par la suite, nous avons fusionné les deux bases de données *Zotero* (étape 2 et étape 3) puis nous avons extrait les doublons pour obtenir une seule base de données de 314 articles;

4. Nous avons procédé à la lecture approfondie (*full read* en anglais) des 314 articles retenus, afin de retenir uniquement les articles qui mentionnent au moins un aspect de la santé du LCSO ou de l'ELCSO. Après une lecture aussi approfondie, 98 articles ont été sélectionnés pour être analysés (voir annexe 4).

Le processus de recherche et de sélection des articles est détaillé à la figure 1.

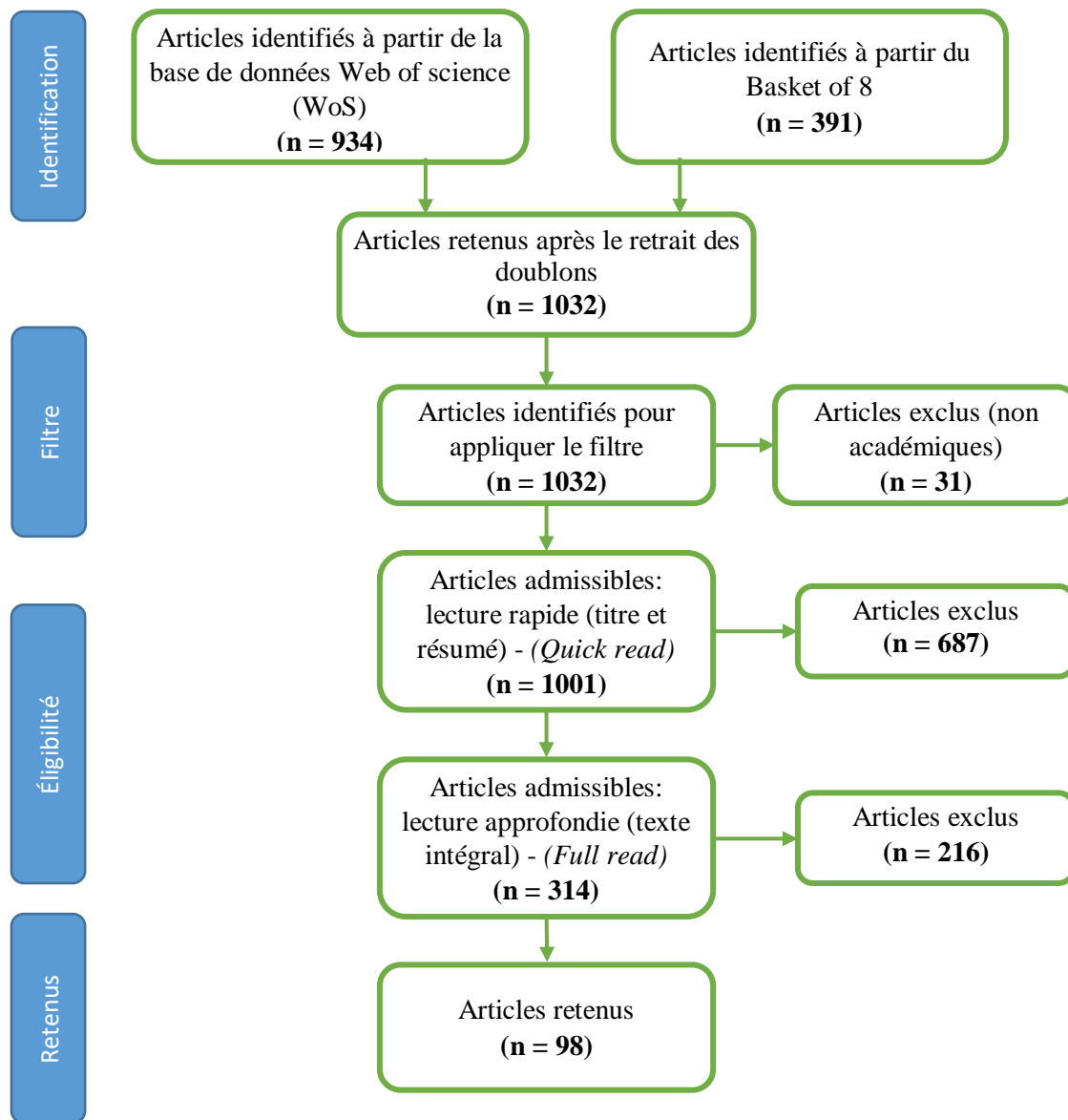


Figure 1 : Étapes du processus de sélection des articles

2.3 Extraction des données

Après la recherche et la sélection des articles, nous avons procédé à l'extraction des données à l'aide du logiciel *NVivo 12 Pro*. Les données extraites de chaque article sont subdivisées en trois (3) parties: 1) les caractéristiques de l'article, 2) les caractéristiques de l'étude rapportée dans l'article et 3) les caractéristiques de la santé des ECLSO que l'étude aura relevé. Selon King et He (2005), l'analyse descriptive implique entre autres la caractérisation d'un certain nombre d'articles, tels que l'année de publication, la méthode de recherche,

l'approche de recherche, les techniques de collecte de données et, dans une large mesure, la force des résultats de recherche de chaque article. Chacune des parties citées précédemment comporte des catégories d'informations (codes) et des subdivisions (sous-catégories), le cas échéant, à collecter. Les parties 2 et 3 portent sur le contenu des articles, alors que la partie 1 porte sur les métadonnées des articles. Les catégories ont été sélectionnées avec l'équipe d'experts du projet *SECOHealth*. Chacune de ces catégories est en fait représentée par des nœuds dans le logiciel *NVivo*. A titre d'exemple, pour les caractéristiques de l'article, nous avons des catégories (ou nœuds) telles que l'année de publication et le titre de l'article, pour ne citer que ces quelques-unes. Dans le logiciel *NVivo*, chaque nœud représente un thème, un concept, une idée ou expression caractéristique dudit nœud. Certains nœuds peuvent être subdivisés en sous-nœuds selon le niveau de granularité désiré.

Les catégories représentant les caractéristiques de l'article sont :

- l'année de publication;
- les noms des auteurs;
- le titre de l'article;
- la source où l'article a été publié;
- le domaine d'appartenance;
- le type de publication.

L'identification de l'appartenance d'un article à un domaine précis (SI ou GL) s'est faite sur la base de la ligne éditoriale de chaque revue. Quant au type de publication, il peut s'agir d'articles publiés soit dans une revue, présentés soit dans une conférence, un symposium ou un atelier.

S'agissant de la deuxième partie, elle s'intéresse aux caractéristiques de l'étude ayant été rapportée dans chaque article. Les catégories représentant les caractéristiques de l'étude sont :

- les buts et objectifs de l'étude;
- le type (p.ex. conceptuel ou empirique)
- l'approche de recherche;
- la méthode de recherche (p.ex. étude de cas, enquête, Delphi);
- l'utilisation des bases théoriques;
- le développement d'une nouvelle théorie;
- les disciplines de référence (c'est-à-dire les disciplines auxquelles des théories ou méthodes sont empruntées);
- le nom du ou des logiciels étudiés;
- le nom de ou des écosystèmes logiciels étudiés;
- le nom de la ou des communautés étudiées;
- le nom du ou des projets logiciels étudiés;

- le nom de la ou des organisations étudiées;
- le nom du ou des contributeurs étudiés.

Quant à la troisième partie, elle s'intéresse principalement aux caractéristiques de la santé des écosystèmes étudiés. C'est cette partie qui constitue l'essence de notre travail de recherche. Les catégories représentant les caractéristiques de la santé sont :

- la définition des concepts d'écosystème et de santé;
- les aspects étudiés de la santé;
- les indicateurs de santé;
- les problèmes de santé et leurs causes et impacts;
- les métriques utilisés pour mesurer la santé;
- la validation des dites métriques.

Ainsi, les caractéristiques de l'étude rapportée dans chaque article et les caractéristiques de la santé des ECLSO que chaque article aura relevé sont représentés par des extraits de texte des articles de notre échantillon.

L'extraction nous a permis d'analyser et de classer les articles définitivement retenus. Ceci nous permet d'évaluer avec exactitude et minutie la pertinence de nos articles retenus, et partant, de répondre à nos questions de recherche formulées à l'introduction. Bien que subjective, l'analyse et la classification desdits articles se sont faites de façon rigoureuse à l'aide d'une validation intercodeur. Ce processus de validation s'est fait avec un étudiant de doctorat en SIO de l'Université Laval, membre de l'équipe du projet *SECOHealth*. Pour y parvenir, nous avons sélectionné au hasard cinq articles que nous lui avons transmis pour effectuer le travail d'extraction des données, parmi les vingt articles que nous avons déjà lu intégralement et dont l'extraction des données avait été faite dans un premier temps sur une feuille de calcul Excel. Au terme de ce travail, nous avons comparé manuellement les résultats respectifs des deux codeurs. Avant de procéder à la phase d'analyse des données, bien que la fiabilité entre les évaluateurs soit élevée (un peu moins de 80%) (Landis et Koch, 1977), les désaccords concernant les résultats du codage ont été discutés jusqu'à ce que les deux parties parviennent à un consensus.

2.4 Analyse des données

L'analyse des données constitue la dernière phase de notre méthodologie de travail. Les 98 articles retenus à l'issue du processus de sélection des articles ont été codifiés à l'aide du logiciel *NVivo*. En effet, les codes représentent ce que nous avons appelé précédemment des

catégories (ou nœuds). Pour ce faire, nous nous sommes appuyés sur les recommandations de Miles et Huberman (1994) en matière d'analyse des résultats. Par la suite, nous avons procédé à l'analyse descriptive des données recueillies. En effet, l'analyse descriptive permet de quantifier un corpus de la recherche (King et He, 2005). Une analyse de fréquence considère un article retenu comme un seul enregistrement de données (unité individuelle d'analyse) et identifie les différentes tendances parmi les autres articles examinés faisant partie du corpus à l'étude (King et He, 2005).

Afin de pouvoir répondre à nos deux questions de recherche, nous avons analysé le contenu codifié dans chacune de ces catégories. Le Tableau 1 présente les analyses qui ont été sélectionnées dans le cadre de ce travail.

Tableau 1 : Sélection des analyses de données

Catégorie	Analyses sélectionnées
Caractéristiques de l'article	<ul style="list-style-type: none"> - Nombre d'articles publiés par année - Nombre d'articles publiés par source - Nombre d'articles publiés par domaine d'appartenance - Nombre d'articles publiés par type de source (p.ex. journal ou conférence ou atelier)
Caractéristiques de l'étude rapportée dans l'article	<ul style="list-style-type: none"> - Nombre d'articles publiés par méthode de recherche - Nombre d'articles publiés par type de recherche (p.ex. empirique) - Nombre d'articles publiés par source de données en ligne - Nombre d'articles publiés par base théorique (p.ex. théorie ou modèle)
Caractéristiques de la santé des ECLSO	<ul style="list-style-type: none"> - Définition de la santé dans les articles - Définition de l'écosystème dans les articles - Aspects étudiés de la santé - Composants étudiés de l'écosystème - Combinaisons étudiées d'aspects de la santé et de composants de l'écosystème

Pour analyser nos données et faciliter la présentation éventuelle des résultats, nous avons opté pour la tabulation et la synthèse des données. Il s'agit d'un exercice qui doit toujours précéder les statistiques inférentielles (Victor, 2018). Les tableaux croisés dynamiques que nous avons utilisés représentent l'une des fonctionnalités les plus puissantes d'Excel, avec des fonctions de tabulation qui facilitent énormément les statistiques descriptives (Victor,

2018) car ils permettent de faire les comparaisons et donner les tendances des données. Pour ce faire, certaines données extraites ont été consolidées puis exportées dans le logiciel Excel 2010 pour faciliter l'analyse, par la conception des tableaux et figures. Concernant la 3^e partie portant sur les caractéristiques de la santé, nous avons construit des nuages de mots autour des deux concepts principaux afin de savoir comment ils sont construits dans la littérature, c'est-à-dire les mots ou groupes de mots généralement associés (voir annexe 5).

Les figures et les tableaux ont été produits par l'auteur de ce mémoire. Tout en permettant aux lecteurs de gagner du temps et sauver de l'énergie, les graphiques, figures et tableaux sont utiles pour mieux comprendre les résultats, faciliter la transmission de la plus grande quantité d'informations de la manière la plus claire possible, compléter le contenu du texte et approfondir la compréhension (Franzblau et Chung, 2012). Ainsi, nous pouvons avoir une vue globale de la distribution des articles en fonction de l'année de publication ou du domaine d'appartenance, pour ne citer que ces deux exemples. S'agissant du domaine d'appartenance, il est intéressant de savoir si la majeure partie des articles ont été publiée dans des revues spécialisées en SI ou en GL, par exemple.

À l'issue du codage et de l'analyse qui s'en est suivi, nous avons été en mesure de répondre à nos deux questions de recherche.

Le présent chapitre nous a permis de passer en revue la méthodologie employée pour mener ce travail de recherche. Les résultats qui en résultent sont présentés dans le chapitre suivant.

Chapitre 3 : Résultats

Après un traitement des données extraites, la présente section de ce chapitre se consacre principalement à la présentation des résultats qui en découlent. L'utilisation des tableaux et graphiques permettent de les présenter de façon organisée. Ce chapitre s'articule autour de trois sections. La première section se consacrera notamment à une analyse descriptive des caractéristiques des articles de notre échantillon et de l'étude rapportée dans chaque article, tandis que les deux sections suivantes répondront chacune à nos deux questions de recherche formulées dans la partie Introduction du présent document :

- Q1. Comment la santé des ELCSO est-elle définie dans la littérature ?
- Q2. Quels aspects de la santé des ELCSO, incluant leurs composants, ont été étudiés et, parmi ceux-ci, lesquels ont fait l'objet du plus grand nombre d'études?

3.1 Analyse descriptive

Cette section est subdivisée en deux sous-sections. La première sous-section s'intéresse à l'analyse descriptive des caractéristiques des articles de notre échantillon tandis que la seconde présente les caractéristiques de l'étude rapportée dans chacun de ces articles.

3.1.1 Caractéristiques des articles analysés

Les Tableau 2 et Tableau 3 et la Figure 2 présentent respectivement le nombre d'articles par domaine d'étude, par type de publication et par année de publication.

Comme indiqué dans le Tableau 2, le domaine des SI enregistre le plus grand nombre d'articles, représentant presque la moitié des 98 articles de notre échantillon. La modalité « Autres » dans le Tableau 2 représente les articles qui ont été publiés dans des journaux n'appartenant ni au domaine des SI ni à celui du GL exclusivement. Il s'agit des revues du domaine des statistiques (*Statistical Science*) ou multidisciplinaires (p.ex. *Decision Sciences*), pour ne citer que ces deux (voir annexe 4).

Tableau 2 : Nombres d'articles par domaine

Domaine	Nombre d'articles	Pourcentage de l'échantillon
Systèmes d'information	50	52,04%
Génie logiciel	38	38,78%
Autres	9	9,18%
Total	98	100%

S'agissant du Tableau 3, il présente la répartition du nombre d'articles par type de publication. On peut se rendre compte à l'évidence que presque les trois quarts des articles retenus pour notre étude ont été publiés dans des revues.

Tableau 3 : Nombres d'articles par type de publication

Type de publication	Nombre d'articles	Pourcentage de l'échantillon
Revue	74	75,51%
Conférence	22	22,45%
Symposium	1	1,02%
Atelier	1	1,02%
Total	98	100%

La Figure 2 présente une courbe en forme de cloche avec le pic des publications enregistré en 2012 et 2013. On note une nette progression du nombre d'articles publiés de 2006 à 2012. Après 2013, le nombre de publications décroît en dents de scie jusqu'à atteindre en 2018, un nombre similaire à celui de 2006. On constate que presque plus du quart des articles retenus pour notre étude ont été publiés en 2012 et 2013.

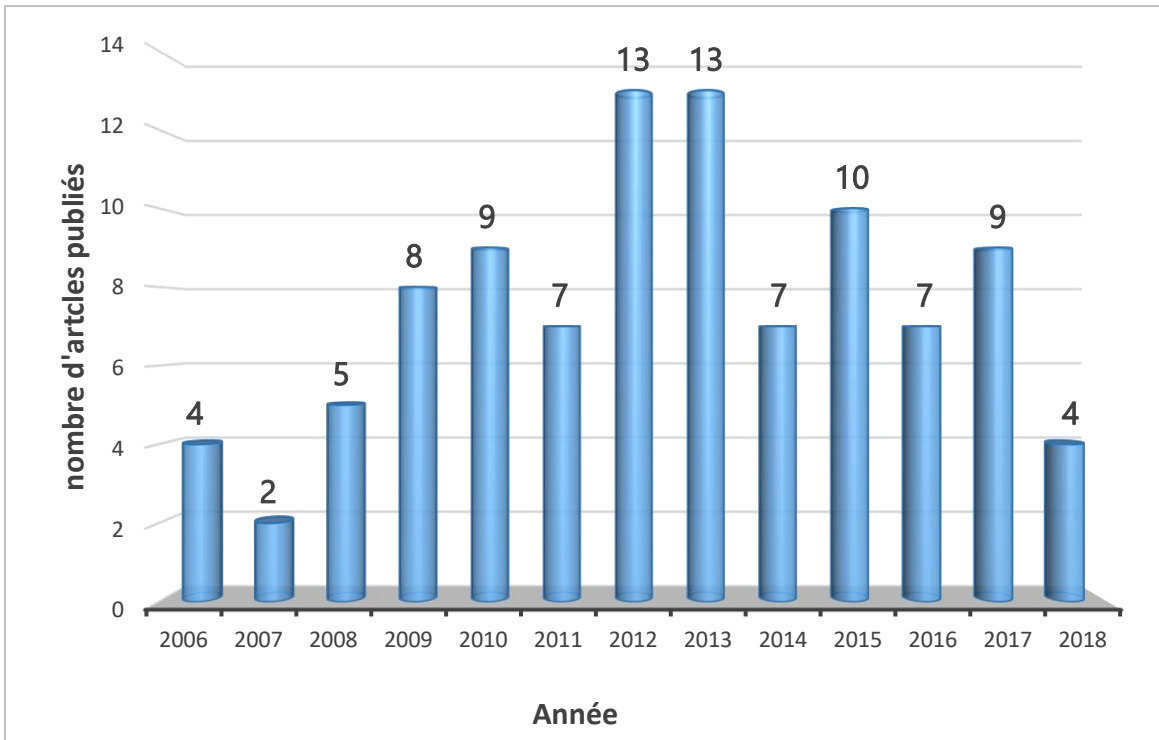


Figure 2 : Nombre d'articles par année de publication

Le Tableau 4 révèle que les quatre revues ayant enregistré le plus grand nombre d'articles sont respectivement, par ordre décroissant, *Information Systems Research*, *MIS Quarterly*, *Journal of Strategic Information Systems* et *Decision Support Systems*. Ces revues sont toutes issues du domaine SI. Environ le quart des articles de notre échantillon a été publié dans ces quatre revues. Le Tableau 4 présente uniquement les résultats pour les revues ayant plus d'un article publié parmi ceux de notre échantillon.

Tableau 4 : Nombre d'articles par publication et par année

Année de publication	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	Total
Nom de la publication														
Information Systems Research	1		1				1	2				2		7
Management Information Systems Quarterly	2		1			1	1		1				1	7
Journal of Strategic Information Systems							1	2			1	1	1	6
Decision Support Systems				1		1		1	1		1			5
Information and Software Technology					1		1						1	3
International Conference on Open Source Systems			1			1			1					3
Journal of the Association for Information Systems					2					1				3
Journal of Management Information Systems			1	1									1	3
The Journal of Systems and Software						1	1		1					3
ACM Conference on Computer-Supported Cooperative Work and Social Computing										1	1			2
European Journal of Information Systems				1		1								2
Information Technology & People				1								1		2
Information Systems Management							1			1				2
International Business Information Management Conference					2									2
Journal of Database Management			1			1								2
Total	4	2	5	8	9	7	13	13	7	10	7	9	4	98

3.1.2 Caractéristiques de l'étude rapportée dans chaque article

Comme le montre le Tableau 5, le plus grand nombre de publications portant sur la santé des ELCSO reposent sur une étude exploratoire, c'est-à-dire avec une collecte des données dans les sources de données concernant des projets LCSO en ligne (p.ex. *SourceForge.net*, *GitHub*) dans l'optique d'explorer un phénomène précis (p.ex. la relation entre variables). Les auteurs n'apportent aucune précision claire et précise sur la nature de la méthode employée; ils se contentent de souligner qu'ils ont collecté les données, puis énoncent les métriques utilisées. Les enquêtes sont la 2^e méthode de recherche qui enregistre le plus d'articles (presque le quart), suivies des études ayant adopté une approche moins conventionnelle (p.ex. ethnographie) et des études de cas qui enregistrent sensiblement le même nombre d'articles. Ensuite, très peu d'auteurs ont fait des expérimentations ou de la théorisation ancrée.

Tableau 5 : Classification en fonction de la méthode de recherche

Méthode de recherche	Nombre d'articles	Pourcentage de l'échantillon
Étude exploratoire	35	35,71%
Enquête	20	20,41%
Mixte (p.ex. étude cas et enquête)	15	15,31%
Étude de cas	13	13,27%
Expérimentation	4	4,08%
Théorisation ancrée (<i>Grounded theory</i>)	2	2,04%
Autres (p.ex. ethnographie)	8	8,16%
Total	97	100%

S'inspirant de la classification faite par Franco-Bedoya *et al.* (2017), nous avons classé les publications en deux catégories selon le type d'études. D'abord, les articles empiriques, dans lesquels les auteurs ont collecté des données pour procéder à la validation de leurs hypothèses et, par ricochet, répondre à leur question de recherche. Ensuite, les articles non

empiriques, qui comprennent entre autres des articles d'opinion. La Figure 3 présente le nombre d'articles pour chaque type. On constate que la très grande majorité des articles porte sur une étude empirique.

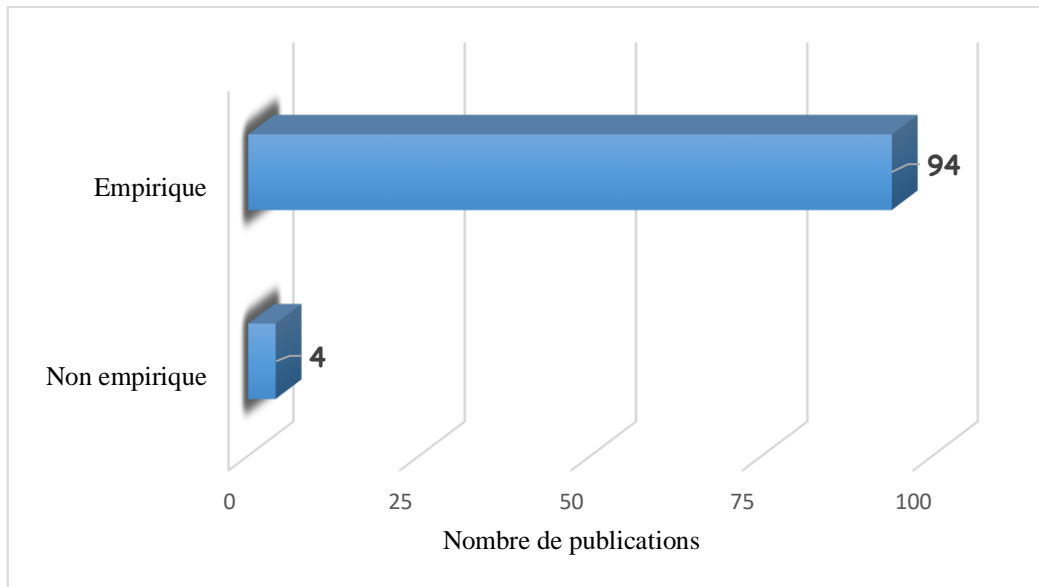


Figure 3 : Classification en fonction du type de recherche

La Figure 4 présente le nombre d'articles qui utilisent des sources de données de projets en ligne (Données en ligne), ceux qui utilisent des sources de données qui ne sont pas en ligne (Données hors ligne) et ceux qui n'utilisent aucune donnée (Pas de données). À la lecture de cette figure, on constate que sur les 98 articles retenus pour notre étude, près des deux tiers (61,22%) utilisent des sources de données de projets en ligne, tandis que 35,71% utilisent d'autres sources de données telles que les entrevues ou les enquêtes. Par ailleurs, 3,06% n'utilisent aucune donnée. Parmi les sources de données en ligne, nous avons trouvé que près des trois quarts des articles ont utilisé *SourceForge.net* comme source de données pour la validation des hypothèses de recherche énoncées. *GitHub* est le deuxième référentiel de données en ligne le plus utilisé.

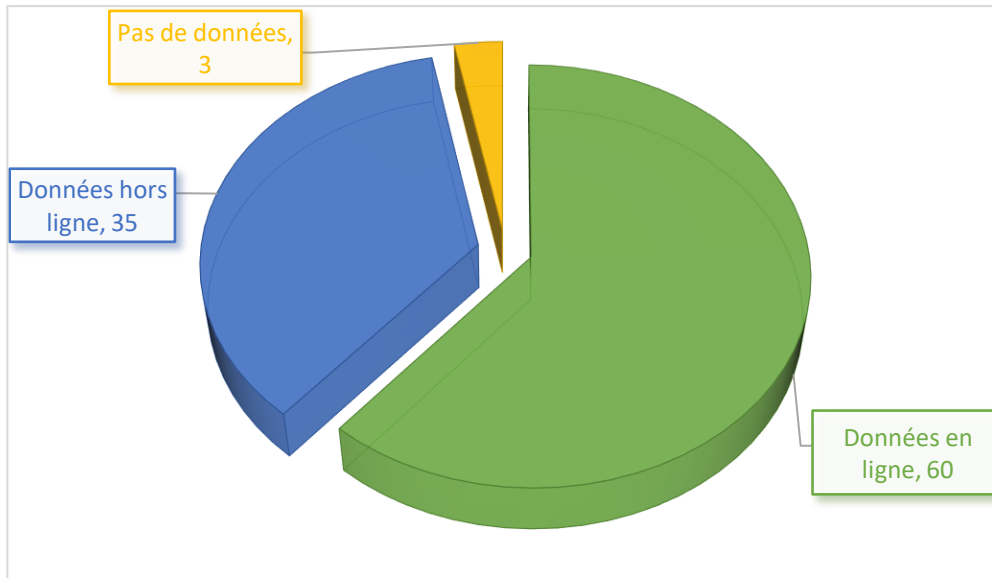


Figure 4 : Nombre d'articles en fonction de l'utilisation de la source de données

Le Tableau 6 présente les huit modèles et théories qui ont été le plus utilisés par les auteurs de notre échantillon. On constate que c'est la théorie de la Vision organisante (*Organizing Vision*) de Swanson et Ramiller (1997; 2004) qui a été le plus utilisée. Notons cependant que 37,76% des articles ont fait usage des théories et modèles. Par ailleurs, certains auteurs ont utilisé 2, voire 3 théories et/ou modèles. Le Tableau 6 présente uniquement les résultats pour les modèles et théories ayant été utilisés dans au moins deux articles parmi ceux de notre échantillon.

Tableau 6 : Nombre d'articles ayant utilisé une théorie ou un modèle existant

Modèles et théories	Nombre d'articles
Théorie de la vision organisante	3
Théorie des capacités dynamiques (DCT)	2
Théorie institutionnelle	2
Écologie organisationnelle	2
Théorie de l'autodétermination	2
Théorie des réseaux sociaux	2
Modèle d'acceptation de la technologie	2
Cadre Technologie – Organisation – Environnement (TOE)	2

3.2 Définitions de la santé des ELCSO

Parmi tous les articles retenus pour l'extraction et l'analyse des données, aucun ne définit de façon explicite le concept de « santé » à l'exception de Lucassen *et al.* (2013) qui, reprenant la définition de Hartigh *et al.* (2006), définissent la santé des écosystèmes comme « le bien-être financier à long terme de l'écosystème commercial et la force à long terme du réseau » (p.184, traduction libre). De même, les auteurs de 29,59% des articles (29) emploient par endroit le terme « santé » (*health* ou *healthiness* en anglais) ou un de ses dérivés tels que « sain » (*healthy* ou *healthier* en anglais), « malsain » (*unhealthy* en anglais) et « bien-être » (*well-being* en anglais). Parmi ces 29 articles, nous avons trouvé deux articles évaluant la santé d'un écosystème particulier d'une part, et un article évaluant la santé d'un projet LCSO d'autre part, mais nous ne pouvons pas identifier exactement ce que le terme « santé » signifie pour eux.

Cette diversité de concepts et le flou qui entoure leurs définitions influent sur la disposition des éléments qui composent les approches d'évaluation de l'ELCSO. Les auteurs de certains articles de notre échantillon décrivent les indicateurs de santé d'un projet comme suit : « [...] la qualité du code, les évaluations des utilisateurs, le nombre d'utilisateurs/téléchargements et la réutilisation de code, entre autres indicateurs de la santé et du succès d'un projet LCSO. » (Amrit et Van Hillegersberg, 2010, p.226, traduction libre) ou soulignent du moins la nécessité de sélectionner des projets « sains » en rappelant les critères indispensables à la santé d'un projet de LCSO :

Pour concevoir un moyen de sélectionner uniquement les projets de LCSO "sains", nous nous sommes concentrés sur les projets hébergés dans *SourceForge* qui répondaient à trois critères: ils doivent (a) être développés en collégialité (b) être maintenus en collégialité et (c) fournir des données d'activités accessibles au public. (Colazo et Fang, 2009, p.1002, traduction libre)

Certains auteurs (p.ex. Amrit et Van Hillegersberg, 2010) rapportent que la santé d'un projet, d'une communauté ou d'un écosystème est une condition *sine qua none* pour le succès d'un LCSO. Autrement dit, la santé ou l'échec dudit logiciel dépend intrinsèquement de la santé du projet, de la communauté ou de l'écosystème au sein duquel il est développé et maintenu. Selon Amrit et Van Hillegersberg (2010), le modèle de changement de position d'un développeur, du noyau à la périphérie, pourrait être considéré comme un autre

indicateur de la santé des projets LCSO. Ils soulignent que le noyau est l'épicentre du projet. Il est généralement constitué des leaders de projets et des principaux développeurs chargés de guider et de coordonner le développement de projets LCSO, qui participent généralement au projet depuis longtemps (parfois depuis son lancement) et y apportent une contribution significative (Amrit et Van Hillegersberg, 2010). Par ailleurs, la périphérie implique des développeurs qui apportent occasionnellement de nouvelles fonctionnalités au projet et leurs codes sont généralement revus par les développeurs principaux avant l'inclusion dans la base de code (Amrit et Van Hillegersberg, 2010). Ainsi, si un développeur passe de la périphérie au noyau, cela est bénéfique pour le projet, contrairement à un mouvement inverse qui pourrait avoir un impact sur la santé du logiciel (Amrit et Van Hillegersberg, 2010). De plus, ils insistent sur le fait que des mouvements oscillatoires de va-et-vient vers et en dehors du noyau peuvent signaler une instabilité du projet (Amrit et Van Hillegersberg, 2010).

Le constat concernant la définition du concept de santé est presque le même en ce qui concerne la définition du concept d'écosystème. En effet, aucun auteur n'a proposé une définition propre à lui. Ils empruntent des définitions à d'autres auteurs. A titre d'exemple, Hanssen (2012) fait une énumération de quelques définitions du concept « écosystème logiciel » issues de la littérature tandis que Lucassen *et al.* (2013) empruntent la définition proposée par Jansen *et al.* (2009b) : « un ensemble d'acteurs fonctionnant comme une unité et interagissant avec un marché partagé de logiciels et de services, ainsi que leurs interrelations. Ces interrelations reposent souvent sur une plate-forme technologique ou un marché commun, et sur l'échange d'informations, de ressources et d'artefacts. » (p.184, traduction libre). Pour certains auteurs des articles de notre échantillon (p. ex. Shaikh, 2016 et Gamalielsson *et al.*, 2011), un écosystème sain renvoie à un écosystème durable et une croissance à long terme des projets LCSO. En général, le concept d'écosystème est combiné à un autre terme, p.ex. *software ecosystem* ou *ecosystem health* (voir nuage de mots en annexe 5).

3.3 Santé des ELCSO

Après codification, les données recueillies des 98 articles retenus dans notre échantillon ont été analysées pour comprendre comment la santé des ELCSO a été étudiée (voir annexe 6 et 7). Les résultats obtenus sont présentés dans les prochaines sections.

3.3.1 Aspects de la santé

Sur la base des concepts présentés dans les études de notre échantillon, nous pouvons identifier les principaux aspects de la santé des écosystèmes logiciels. Le Tableau 7 présente les définitions de chaque aspect de la santé selon les auteurs qui les ont utilisés.

Tableau 7 : Définition des aspects de santé issus des articles de notre échantillon

Aspect de santé	Définition	Articles [R..]*
Santé	<p><i>La santé des écosystèmes se définit comme « le bien-être financier à long terme de l'écosystème commercial et la force à long terme du réseau ». Dans le contexte de la santé de l'écosystème PaaS, les utilisateurs directs de la technologie PaaS sont les contributeurs de la force à long terme de l'écosystème, à savoir les développeurs. Si un groupe de développeurs contribue activement au développement de la PaaS ou de ses extensions, la PaaS lui-même a plus de chances de réussir à long terme [R25].</i></p> <p><i>Nous nous attendons à ce que les projets ayant un plus grand nombre de lignes de code soient associés à une communauté plus large et aient ainsi une meilleure santé [R47].</i></p> <p><i>L'activité dans les écosystèmes est un facteur important qui reflète la santé et la durabilité à long terme des projets de [LCSO]. Plus précisément, trois aspects de la santé des écosystèmes ont été jugés importants: l'activité dans les écosystèmes; influence de l'entreprise dans les écosystèmes; et l'interaction entre les écosystèmes. [R65].</i></p>	[R25] ; [R47] ; [R65]
Adoption	<p><i>Il est à la fois pertinent et important d'examiner de plus près l'adoption des [LCSO] dans les organisations et, plus spécifiquement, leur institutionnalisation. Une pratique ou une innovation est dite institutionnalisée lorsqu'elle est considérée comme acquise et que son utilisation devient la norme. L'adoption d'une innovation dite institutionnalisée n'est pas le comportement d'un acteur rationnel cherchant à maximiser son efficacité et son efficacité, mais plutôt le comportement d'un acteur institutionnel qui accepte et suit les normes sociales, sans s'interroger ni même réellement penser à leur sujet, afin de garantir la légitimité de ses actions [R50].</i></p> <p><i>Nous assimilons la présence de génériques paramétrés à l'adoption de génériques et la présence de génériques bruts à la non-adoption. Pour les annotations, nous avons compté le nombre d'annotations dans le projet. Notez que ces mesures ne fournissent qu'une vision très large de l'adoption [R23].</i></p>	[R50] ; [R23]
Attractivité	<p><i>La capacité d'attirer des pairs pour co-créeer avec le fondateur est comprise comme l'attractivité du projet. L'attractivité est une cause commune du nombre de visiteurs qu'un site Web de projet reçoit, du nombre d'utilisateurs qu'il a ou du nombre de téléchargements et du nombre de contributeurs dont il</i></p>	[R42] ; [R80]

Aspect de santé	Définition	Articles [R..]*
	<p><i>dispose. Une plus grande attractivité conduit à plus d'intention d'adopter (télécharger) et de contribuer (devenir membre), motivant et justifiant les activités de production et les investissements consacrés au logiciel pour améliorer la qualité et générer de l'innovation [R42].</i></p> <p><i>L'attractivité est définie comme un ensemble de valeurs de projet perçues par les visiteurs, utilisateurs et développeurs potentiels et réels [R80].</i></p>	
Collaboration	Pas défini	[R17]
Conflit	<p><i>Le conflit est un processus d'équipe important; Cet article cherche à comprendre comment la structure unique des équipes "permanentes" (équipes composées de membres dynamiques pouvant évoluer pendant plusieurs années) affecte les processus d'équipe tels que les conflits.</i></p>	[R59]
Convivialité	<p><i>La convivialité désigne « le degré avec lequel un produit peut être utilisé par des utilisateurs spécifiés pour atteindre des objectifs spécifiés avec efficacité, efficience et satisfaction dans un contexte d'utilisation spécifié »</i></p>	[R95]
Défection	<p><i>La densité de défauts est le nombre de défauts par taille du logiciel. Une défection est définie comme un bogue signalé sur l'entrepôt de données SourceForge.</i></p>	[R53]
Durabilité	<p><i>L'activité dans les écosystèmes est un facteur important qui reflète la santé et la durabilité à long terme des projets de [LCSO],</i></p> <p><i>L'influence de la société dans le temps peut avoir une incidence sur la durabilité à long terme d'un projet.</i></p>	[R65]
Efficacité	<p><i>La recherche considère deux types d'efficacité dans les équipes de [LCSO]: l'attraction et la rétention de l'apport des développeurs et la génération des résultats du projet;</i></p>	[R87]
Émergence	<p><i>Le concept d'émergence et sa référence implicite à l'écologie impliquent un changement d'orientation des structures internes de l'organisation (organisme individuel) vers son environnement et les relations et actions au sein de (l'écosystème). Considérer l'industrie du logiciel et son marché comme un écosystème peut introduire un ensemble de nouveaux défis et opportunités, par exemple de nouveaux modèles commerciaux, l'innovation ouverte, le développement collaboratif, les questions de propriété, la planification et gestion de la variabilité.</i></p>	[R13]
Évolutivité	<p><i>Les informations pertinentes sur l'évolution des logiciels indiquent que la taille et les fonctionnalités des logiciels ont tendance à croître d'une version à l'autre afin de rester utiles aux utilisateurs. De même, la complexité du logiciel tendra à augmenter au cours de la vie d'un projet, à moins qu'il ne soit activement géré [R66].</i></p> <p><i>Pour comprendre l'évolutivité de l'écosystème Linux, nous étudions comment la productivité du mainteneur évolue</i></p>	[R66] ; [R63]

Aspect de santé	Définition	Articles [R..]*
	<i>lorsque plusieurs mainteneurs sont ajoutés à un ensemble de fichiers maintenus [R63].</i>	
Maintenance	<i>Le cycle de vie du développement logiciel montre que, lorsque le logiciel est livré aux utilisateurs, tout type d'activité de support est pris en charge dans le cadre de la maintenance logicielle. En réalité, la maintenance des logiciels a une portée beaucoup plus large et couvre de nombreux aspects tels que la perfection, les correctifs et les améliorations.</i>	[R58]
Participation (continue) / Engagement	<i>La participation volontaire est cruciale pour les projets de [LCSO], car une participation volontaire inadéquate entraîne l'annulation de 80% des projets [LCSO] pendant le développement [R15]. L'engagement est un lien psychologique qui stabilise le comportement individuel dans des conditions où celui-ci pourrait être enclin à changer. La forme affective d'engagement organisationnel souligne l'attachement émotionnel entre un individu et son organisation [R86].</i>	[R15] ; [R86]
Performance	<i>La performance est fonction de la capacité et de l'effort, effort qui peut être affecté par les perceptions, les états psychologiques et d'autres facteurs tels que le temps disponible [R22]. L'achèvement des tâches est considéré comme une mesure de la performance du projet et nous avons calculé l'achèvement de la tâche en tant que pourcentage des tâches terminées, c'est-à-dire $\text{achèvement de la tâche} = (\text{total des requêtes} - \text{requêtes ouvertes}) / \text{total des requêtes} \times 100$ [R2].</i>	[R22] ; [R2]
Productivité	<i>Nous quantifions la productivité des développeurs à partir de valeurs abstraites en métriques mesurables afin de comprendre la quantité de contribution et d'engagement de chaque développeur dans un projet [R39]. La productivité est comprise comme la quantité de travail accomplie par chaque communauté de développeurs au fil du temps [R48].</i>	[R39] ; [R48]
Qualité (contrôle)	<i>« Le contrôle de la qualité (CQ) est un processus de vérification et de validation, généralement intégré à un processus de test structuré, qui utilise des plans de haut niveau et des scripts de test détaillés pour documenter et gérer le processus de test ». Nous dimensionnons le CQ d'un projet [LCSO] en utilisant l'efficacité de la gestion des bogues, des demandes d'aide, l'efficacité du support, l'efficacité de la gestion des tâches et de la gestion des artefacts [R45]. Nous considérons la qualité d'un projet comme comprenant deux aspects: le premier est le nombre "d'étoiles" qu'un projet peut recevoir d'utilisateurs, et le second est la réponse de l'équipe de projet aux problèmes signalés pour un référentiel donné. Dans le cas du premier indicateur, nous pensons que la réaction de la communauté au projet est une mesure appropriée de sa qualité. Tout utilisateur est capable d'apprécier un quelconque avec une "étoile" [R49].</i>	[R45] ; [R49]

Aspect de santé	Définition	Articles [R..]*
Succès	<p><i>En termes simples, le succès est la réalisation de quelque chose que l'on souhaite. Mesurer le succès d'un [LCSO] peut être difficile car subjectif. Parmi les aspects de succès des [LCSO] étudiés, la recherche s'est axée sur l'intérêt des utilisateurs, car son intérêt peut avoir une incidence sur les résultats du projet, tels que sa vitalité et son activité [R32]. Nous considérons le succès du « côté de l'offre » et du « côté de la demande ». Les deux parties ont des critères différents pour évaluer le succès du projet. Pour les développeurs, l'activité de développement est un indicateur clé de la réussite du projet: une activité de développement élevée indique que les développeurs du projet contribuent en permanence au projet; le projet mourra jusqu'à ce qu'il n'y ait plus d'activité en développement. Pour les utilisateurs, la popularité du projet est une mesure essentielle de la réussite du projet: sa grande popularité montre que de nombreux utilisateurs s'intéressent et ont tendance à utiliser ce projet; le projet mourra si personne ne s'en sert pendant un certain temps [R38].</i></p> <p><i>Il est clair que, si la théorie du capital social et la perspective de l'agence humaine représentent chacune une compréhension nuancée des facteurs de succès des logiciels libres, aucune ne parvient à elle seule à résoudre ce problème [R52].</i></p> <p><i>Le choix de la licence d'un logiciel est l'un des premiers facteurs de configuration de projet que les chefs de projet doivent prendre en compte dès le début de tout projet de [LCSO]. Par conséquent, son implication stratégique dans le succès des projets est une préoccupation cruciale pour ceux-ci [R53].</i></p>	[R32] ; [R38] ; [R52] ; [R53]
Survie (vitalité)	<p><i>La « capacité de survie du [LCSO] » est la « capacité d'un projet à fournir des fonctionnalités et un soutien à ses utilisateurs ».</i></p> <p><i>La viabilité du projet est une mesure robuste et valide de la survie d'un projet [LCSO] pouvant être utilisée pour prédire avec précision l'échec ou la survie d'un projet [LCSO]. Nous définissons la viabilité d'un projet comme la capacité d'un projet à se développer et à maintenir sa structure lors de la survenance de perturbations. Les trois dimensions de la viabilité du projet sont la vigueur, la résilience et l'organisation.</i></p>	[R20]
Utilisation (popularité)	<p><i>Une communauté efficace d'assistance en ligne aux utilisateurs est également nécessaire pour induire l'utilisation du [LCSO] et la popularité, où la popularité est généralement mesurée en termes de nombre d'utilisateurs actifs ou de téléchargements.</i></p>	[R92]

* [R..] représente un article de notre échantillon, numéroté de R1 à R98

3.3.2 Composants de l'écosystème

Le Tableau 8 présente les définitions de chaque composant de l'écosystème logiciel selon les auteurs qui les ont employés dans les articles de notre échantillon.

Tableau 8 : Définition des composants de l'écosystème issus des articles de notre échantillon

Composant de l'écosystème	Définition	Articles[R.]*
Écosystème	<p><i>Considérer l'environnement organisationnel comme un écosystème signifie qu'il est considéré comme un système ouvert et non fermé, les frontières organisationnelles sont perméables et les organisations entretiennent des relations dynamiques avec d'autres organisations du même domaine. Un écosystème qui fonctionne bien est, en résumé, une configuration complexe avec une collaboration à travers des frontières organisationnelles traditionnellement fermées [R13]. Ensemble d'acteurs fonctionnant comme une unité et interagissant avec un marché partagé de logiciels et de services, ainsi que leurs interrelations. Ces relations reposent souvent sur une plate-forme technologique ou un marché commun et reposent sur l'échange d'informations, de ressources et d'artefacts [R25]</i></p> <p><i>[R65] considèrent qu'un ELCSO est en fait un projet de développement de LCSO. Toutefois, ils ne définissent pas ce qu'est un projet de développement de LCSO.</i></p>	[R13] ; [R25]
Code de conduite	<p><i>Le concept de « code de conduite » est le moyen le plus courant utilisé par des projets. Un tel code de conduite établit essentiellement des règles de base pour les communications entre les participants, décrit les mécanismes d'application des violations et tente de codifier l'esprit d'une communauté, de sorte que tout le monde puisse contribuer confortablement sans distinction de sexe, d'appartenance ethnique ou d'orientation sexuelle.</i></p> <p><i>Un code de conduite comprend des principes, des valeurs, des normes et des règles qui agissent en tant que lignes directrices qui tiennent compte des avantages globaux des parties prenantes tout en respectant les droits des membres.</i></p>	[R43]
Communautés	<p><i>Nous considérons ici [...] des communautés d'individus souvent bien établies et de grande réputation, grâce en grande partie à sa forte "main-d'œuvre" et ses qualités très connues [R9].</i></p> <p><i>Les communautés de développement de [LCSO] sont formées par des personnes dispersées géographiquement qui collaborent au développement de logiciels à distribuer sous une licence qui garantit un accès ouvert au code source du logiciel [R90].</i></p>	[R9] ; [R90]

Composant de l'écosystème	Définition	Articles[R.]*
Contributeur	<i>Nous soutenons que les contributeurs, y compris les visiteurs, les utilisateurs et les développeurs, contribuent de différentes manières au processus d'innovation partagée et qu'en tant que collectif, ils favorisent l'adoption à grande échelle ainsi que le développement et la maintenance de logiciels.</i>	[R80]
Développeurs	<i>Un développeur est un individu qui code pour le plaisir et est intrinsèquement motivé. Il contribue dans un projet [LCSO] et travaille avec d'autres développeurs pour produire un logiciel [R86]. Les produits [LCSO] se développent grâce aux contributions de deux types de développeurs distincts: les développeurs principaux et les développeurs périphériques. Les développeurs principaux sont les fondateurs ou les initiateurs du produit et dirigent le développement de l'architecture du produit et des modules principaux. En revanche, les développeurs périphériques travaillent sur l'amélioration des produits [R51].</i>	[R86] ; [R51]
Équipe (structure sociotechnique)	<i>Une équipe est considéré comme un nombre de développeurs qui s'engagent dans le développement d'un projet en le divisant en parties gérables et répartiraient le travail entre plusieurs équipes de développeurs.</i>	[R48] ; [R59] ; [R74] ;
Mainteneur	<i>Le noyau Linux a une organisation hiérarchique assez sophistiquée au sein de ce groupe de contributeurs appelés mainteneurs. « Les mainteneurs des sous-systèmes sont chargés de collecter les correctifs acceptés, d'effectuer une dernière révision et de les soumettre [...]au principal mainteneur » « Les mainteneurs sont comme des éditeurs de presse » avec des responsabilités lourdes, telles que 487 correctifs en deux semaines et des centaines de courriels par jour. « Être un mainteneur signifie que vous lisez les correctifs des émetteurs, répondez aux questions des développeurs et des utilisateurs sur les éléments liés au sous-système (généralement des rapports de bogues). Si un correctif semble acceptable, testez-le si possible, appliquez-le à l'arborescence appropriée, transmettez-le publiquement et informez l'auteur qu'il a été accepté. Chaque jour de la semaine, ces arbres sont fusionnés dans la prochaine version de linux, et inévitablement, des problèmes sont signalés et il incombe au mainteneur de les résoudre lorsqu'ils affectent leur portion du noyau »</i>	[R63]
Organisation / Institution / Corporation	<i>Pas défini</i>	[R15] ; [R31]
Ouverture (opensourcing)	<i>Le terme d'ouverture (opensourcing) consiste à utiliser le modèle de développement de [LCSO] en tant que stratégie globale d'approvisionnement pour le processus de développement de logiciels d'une organisation. L'une des caractéristiques novatrices de l'étude réside dans le fait qu'elle prend en compte les deux aspects du processus, c'est-à-dire l'organisation qui « commande » l'opensourcing et, à son tour, la communauté des [LCSO] qui s'occupe du développement.</i>	[R67]

Composant de l'écosystème	Définition	Articles[R.]*
Projets	<p><i>Les développeurs sur les [LCSO] contribuent aux projets de différentes manières. Écrire un morceau de code, réviser un code écrit par d'autres développeurs, signaler un problème dans un projet et résoudre un problème signalé sont des types courants de contribution.</i></p> <p><i>Les projets [LCSO] comprennent des groupes de développeurs et d'utilisateurs dispersés géographiquement mais reliés entre eux par des valeurs partagées et Internet.</i></p>	[R80]
Utilisateurs	<p><i>Les utilisateurs stipulent des problèmes pertinents à résoudre, tels que des rapports de bogues et activent les externalités de réseau qui non seulement accroissent la popularité du projet, mais attirent également de nouveaux utilisateurs et maintiennent la contribution des développeurs.</i></p>	[R80]
Visiteur	<p><i>Les visiteurs d'une page Web du projet, représentant la visibilité et le succès commercial de la marque peuvent contribuer aux externalités du réseau en améliorant le classement du projet visité. De plus, les visiteurs peuvent contribuer indirectement à l'amélioration du projet en signalant les liens rompus ou en participant à des activités de recherche et développement.</i></p>	[R80]
Volontaire	<p><i>La durée du mandat des volontaires a été mesurée à l'aide d'un indicateur à élément unique indiquant le nombre d'années pendant lesquelles un volontaire a travaillé sur le projet en cours. Cette mesure est significative pour tous les rôles de bénévole (p. ex., programmeur, concepteur, emballeur et correcteur de bogues) et est également cohérente avec la façon dont le travail passé a mesuré la durée du mandat.</i></p> <p><i>Les membres des projets [LCSO] étant généralement des volontaires, leurs motivations impliquent généralement des objectifs / intérêts personnels, par exemple, aiguiser leurs compétences en programmation et acquérir de l'expérience, suivre des collègues, créer des réseaux avec les membres de la communauté [LCSO] ou simplement soutenir des projets [LCSO].</i></p>	[R15]

* [R..] représente un article de notre échantillon, numéroté de R1 à R98

3.3.3 Aspects de la santé des composants de l'écosystème

Le Tableau 9 présente les aspects de santé et les composants d'écosystème étudiés ainsi que les articles associés.

Tableau 9 : Aspects de santé et composants d'écosystème étudiés

<i>Aspect de santé</i> <i>Composant d'écosystème</i>	Santé	Succès	Performance	Adoption	Durabilité	Qualité	Survie	Convivialité	Attractivité
Écosystème	[R25]; [R65]				[R65]				
Utilisateurs			[R22]						[R80]
Organisation / Institution Corporation									
Projets (logiciels)	[R47]	[R2]; [R5]; [R7]; [R11]; [R14]; [R28]; [R21]; [R29]; [R32]; [R38]; [R52]; [R53]; [R54]; [R55]; [R56]; [R58]; [R61]; [R62]; [R64]; [R68]; [R73]; [R79]; [R82]; [R83]; [R85]; [R89]; [R96];	[R2]; [R8]; [R26]; [R44]; [R71]; [R74]	[R3]; [R4]; [R19]; [R23]; [R24]; [R27]; [R34]; [R35]; [R36]; [R50]; [R57]; [R60]; [R88]	[R6]; [R12]; [R70]; [R77]; [R93]	[R15]; [R37]; [R45]; [R49]; [R51]	[R20]; [R49]; [R72]; [R75]; [R76]	[R1]; [R10]; [R18]; [R94]; [95]	[R42]
Développeurs			[R46]; [R81]; [R84]; [R97]			[R51]			[R80]
Communautés			[R46]		[R9]; [R78];	[R51]	[R90]	[R95]	
Équipe (structure sociotechnique)			[R74]						
Code de conduite				[R43]					
Contributeur			[R44]; [R98];						[R80]
Mainteneur									
Ouverture (open sourcing)		[R67]							
Visiteur									[R80]
Volontaire									

Tableau 9 : suite

<i>Aspect de santé</i> <i>Composant d'écosystème</i>	Participation (continue) / Engagement	Utilisation (popularité)	Productivité	Évolutivité	Émergence	Collaboration	Défection	Maintenance	Conflit	Efficacité
Écosystème					[R13]					
Utilisateurs										
Organisation / Institution / Corporation	[R31] ; [R15]									
Projets (logiciels)		[R16] ; [R69] ; [R92]		[R41] ; [R66]	[R91]		[R40]	[R58]		
Développeurs	[R86]		[R39]			[R17]				
Communautés	[R33]		[R39]			[R9] ;				
Équipe (structure sociotechnique)			[R48]						[R59]	[R87]
Code de conduite										
Contributeur										
Mainteneur				[R63]						
Ouverture (<i>opensourcing</i>)										
Visiteur										
Volontaire	[R15]									

* [R..] représente un article de notre échantillon, numéroté de R1 à R98

Le Tableau 9 révèle que seulement 3,06% des articles ont étudié précisément la « santé » parmi lesquels 2 articles ont traité de la santé d'un écosystème. Il s'agit des articles de Lucassen *et al.* (2013) et Gamalielsson *et al.* (2011). Le troisième article, celui d'Amrit et Van Hillegersberg (2010), s'est penché sur l'étude de la santé d'un seul projet LCSO.

Lucassen *et al.* (2013) ont mené une recherche exploratoire sur huit écosystèmes, fournisseurs de plateforme en tant que service (*Platform As A Service – PaaS*) à l'aide des données recueillies sur *GitHub*. Les fournisseurs de *PaaS* inclus dans leur recherche sont: *Azure*, *Cloud Foundry*, *Cloud*, *Engine Yard*, *Google App Engine*, *Heroku*, *Nodejitsu* et *OpenShift*. Étant donné que les fournisseurs de *PaaS* sont principalement des entreprises de logiciels, les auteurs ont exploité les connaissances du domaine universitaire des écosystèmes logiciels pour étudier la santé de ces écosystèmes. C'est ainsi qu'ils adoptent la définition proposée par Jansen *et al.* (2009b) mentionnée précédemment. Afin d'évaluer la santé d'un écosystème *PaaS*, quatre indicateurs issus de la littérature ont été considérés comme appropriés dans le contexte *PaaS*: le nombre de développeurs actifs, les *spin-offs*², l'intérêt pour le projet et le nombre de téléchargements du logiciel développé. Sur tous ces indicateurs, *Heroku* a été dominant, le positionnant comme le leader du *PaaS*. L'intérêt porté à ce fournisseur ne se limite pas uniquement aux développeurs actifs, mais prend en compte un grand nombre d'utilisateurs passifs intéressés par l'avancement de ses projets. Cependant, la taille absolue n'est pas un indicateur d'un grand écosystème en soi. Par ailleurs, la répartition généralement positive des développeurs actifs par semaine implique des prévisions de croissance futures positives pour l'ensemble du secteur des fournisseurs *PaaS*. Les résultats présentent un aperçu général de l'état de chaque écosystème basé sur ces quatre indicateurs. Pour ce faire, les auteurs estiment que ces résultats aideront les entreprises et particuliers pour une meilleure prise de décision éclairée sur le choix des fournisseurs de services *PaaS*. De plus, ils donnent aux fournisseurs de *PaaS* des informations sur l'état de leur écosystème par rapport à leurs concurrents.

² Selon le dictionnaire Cambridge (<https://dictionary.cambridge.org/>), il s'agit d'un produit qui se développe à partir d'un autre produit plus important. Autrement dit, dans le contexte de LCSO, il peut s'agir d'un projet (ou produit logiciel) qui se développe à partir d'un autre projet (ou autre produit). Ce phénomène pourrait s'étendre aux communautés. Ce qui entraînerait le départ de principaux développeurs engagés dans le projet initial.

L'étude de cas des deux écosystèmes distincts constitués respectivement des projets *TopCased* et *Papyrus* (Gamalielsson *et al.*, 2011) a porté sur la durabilité et l'évolution des communautés au sein de ces écosystèmes au fil du temps, en mettant un accent particulier sur l'influence organisationnelle, à l'aide d'une analyse quantitative sur des données recueillies sur des bases de données publiques. Plus précisément, les auteurs ont étudié trois aspects de la santé des écosystèmes : l'activité dans les écosystèmes, mesurée par le nombre de contributions des contributeurs au logiciel; l'influence des entreprises dans les écosystèmes, mesurée par le degré avec lequel différents contributeurs sont affiliés à différentes entreprises; et l'interaction entre les écosystèmes pour les deux projets, évaluée en étudiant les contributeurs actifs dans les deux projets à la fois. Sans toutefois définir ce qu'est la santé d'un écosystème, les auteurs ont établi des informations détaillées sur le degré d'activité des développeurs et des utilisateurs et l'identification de l'influence organisationnelle sur les deux écosystèmes. C'est ainsi que leur analyse a suggéré que les deux projets ont des écosystèmes sains, avec une base active de développeurs et d'utilisateurs.

Le Tableau 9 révèle que, parmi les aspects de la santé, les trois qui sont les plus étudiés sont, dans l'ordre, « succès », « performance » et « adoption ». Par ailleurs, ceux qui ont reçu moins d'attention sont, *ex aequo*, « défection », « maintenance », « conflit », et « efficacité ». Le tableau 9 révèle aussi que le composant de l'écosystème qui est le plus étudié est « projet » avec 87,76% des articles. On note également que certains auteurs ont utilisé les termes de « visiteur » ou « mainteneur » pour désigner certains types de contributeurs ou d'utilisateurs et le terme « ouverture » (*opensourcing*) considéré comme un modèle de développement de logiciel. Semblable à l'externalisation, ce modèle permet aux entreprises de sous-traiter certaines activités de développement, sinon toutes, à une autre partie (Agerfalk et Fitzgerald, 2008). Chacun de ces termes apparaissent dans un seul article. Nous pouvons constater que seulement 3,06% des articles ont explicitement étudié l'écosystème. De plus, plus d'un quart des articles se sont intéressés au succès d'un projet tandis que 13,27% des articles se sont penchés sur l'adoption, par des utilisateurs, du logiciel développé dans un projet.

La discussion des résultats présentés dans le présent chapitre, ainsi que les limites et implications de notre étude, feront l'objet du chapitre suivant.

Chapitre 4 : Discussion, limites et implications

Après avoir présenté les différents résultats au chapitre précédent, le présent chapitre vise d'une part, à élaborer une discussion afin d'interpréter lesdits résultats et, d'autre part, à relever les limites et implications de notre étude.

4.1 Discussion

Compte tenu de la nouveauté relative du concept de « santé » des ELCSO, il n'est pas surprenant qu'il n'existe pas jusqu'à date une base de recherche solide sur ce phénomène. Gardant cela à l'esprit, notre étude visait à mieux le comprendre. L'objectif de ce travail de recherche était de structurer et d'analyser la littérature existante afin de fournir un aperçu des connaissances scientifiques sur la santé des ELCSO en identifiant l'état actuel de la recherche. Au cours des dernières années, les LCSO ont suscité l'attention des chercheurs, des praticiens et des membres du monde universitaire et cette forte attention est le fruit d'une croissance remarquable de leur nombre (Ghapanchi, 2015). Considérés comme des solutions innovantes, bon nombre d'organisations démarrent, développent et maintiennent de nouveaux projets de développement ou d'utilisation de LCSO au détriment des logiciels propriétaires (Ruff, 2016). Maillon essentiel et indispensable de l'écosystème logiciel mondial, les projets de LCSO s'y intègrent dans l'optique de réduire les coûts et les efforts de développement et, par ricochet, faciliter l'accélération de l'innovation (Marsan *et al.*, 2019). Au quotidien, notre environnement est complètement envahi par des écosystèmes d'applications mobiles (Hyrynsalmi *et al.*, 2016) tels que l'écosystème *Android* de *Google* présent dans bon nombre de téléphones intelligents vendus aux utilisateurs (Manikas et Hansen, 2013c). Il est bien logique de se demander ce qu'il adviendrait à ces ELCSO, dont les organisations et individus sont de plus en plus dépendants au quotidien, s'ils se dégradaient ou cessaient de fonctionner? Mens *et al.* (2017) font le constat selon lequel il n'y a véritablement pas de management centralisé qui puisse faciliter le monitoring de la santé des ELCSO car les logiciels sont basés sur des technologies ouvertes sujettes à des mutations exponentielles. C'est ce qui a motivé cette revue de littérature systématique de la santé des ELCSO.

Les données collectées nous ont permis de nous rendre compte, à l'évidence, que le concept de « santé » n'est pas encore bien appréhendé par la communauté des chercheurs. Ceci rejoint les conclusions de Amorim *et al.* (2017b) qui ont constaté plusieurs divergences en ce qui concerne le concept approprié de la santé des écosystèmes logiciels. Il n'y a aucun consensus à propos de la définition de la santé, mais une définition largement acceptée est qu'un écosystème en santé est stable et durable (Jansen, 2014). Autrement dit, un écosystème en santé devrait pouvoir conserver sa structure organisationnelle et maintenir une stabilité dans la durée, avoir de bonnes capacités d'auto-ajustement et une certaine capacité de gestion de son environnement externe (Liao *et al.*, 2019). L'aspect de santé « popularité » ne fait pas l'unanimité car Stewart *et al.* (2006) s'inquiètent du fait qu'il adviendrait que des projets ayant des structures à complexité croissante ne soient plus en mesure d'apporter des améliorations qui puissent répondre efficacement aux besoins sans cesse croissants et changeants des utilisateurs et par conséquent, parvenir à perdre ainsi en popularité. Ils estiment que cette tendance de considérer la popularité comme un aspect de la santé ne serait pas souhaitable à long terme pour certains projets de LCSO. S'il n'y a aucun consensus pour certains aspects de santé ou composants de l'écosystème, il n'en demeure pas moins qu'il serait difficile d'avoir un consensus quant à la définition du concept de « santé » ou d'écosystème.

On constate dans nos résultats, d'une part que la plupart des articles, sinon la totalité, ont mené une étude empirique, et d'autre part que très peu d'études ont porté sur des expérimentations ou de la théorisation ancrée comme méthodes de recherche. Compte tenu du manque de consensus sur la définition des concepts sus-évoqués, il serait bénéfique qu'il y ait plus d'études non-empiriques, i.e. des études de développement conceptuel ou théorique. En effet, ce sont ces types d'étude qui permettent de définir et de clarifier des concepts (p.ex. santé, écosystème) et des liens entre ces concepts. Certes la découverte de nouveaux concepts est une avancée significative dans la recherche en général, mais il est tout aussi bénéfique de mieux conceptualiser ceux déjà existants (Barki, 2008). Soulignant l'utilité d'une meilleure conceptualisation des concepts, Barki (2008) précise que leur découverte et leur développement sont loin d'être anodins. Faisant le constat qu'il existe un écart dans la littérature quant aux directives concernant les premières étapes du développement d'un concept, l'auteur propose quatre approches pour conceptualiser les

concepts : 1) la formulation d'une définition claire, condition *sine qua none* notamment en SI compte tenu de la prolifération de nouvelles terminologies et de nouveaux concepts constamment développés en raison de l'évolution rapide des TI; 2) la spécification des dimensions d'un concept et ses relations, utile lorsque des concepts multidimensionnels sont conceptualisés de différentes manières dans la littérature et qu'il n'existe aucun consensus pour concilier ces différences, dans la mesure où les phénomènes étudiés sont complexes et font appel à des vues multidisciplinaires; 3) l'exploration de l'application d'un concept à plusieurs contextes, qui consiste à explorer la manière dont un concept quelconque pourrait s'appliquer dans plusieurs contextes tels que technologiques, organisationnels ou individuels; 4) l'extension de la conceptualisation d'un concept, qui consiste à élargir la conceptualisation des concepts existants afin qu'ils puissent aider à mieux expliquer et comprendre les réalités organisationnelles complexes aux facettes multiples. On pourrait comprendre pourquoi plusieurs auteurs optent pour des études empiriques, car bon nombre d'entre eux se préoccupent davantage à la validité du concept et au développement de mesures dudit concept qu'à leur identification et leur spécification (Barki, 2008). Dès lors, il est important d'attirer l'attention de la communauté de

chercheurs qui croient que la conceptualisation de nouveaux concepts ne devrait pas être une préoccupation dans des domaines appliqués tels que les SI, que l'emprunt des concepts développés dans des disciplines de référence en les adaptant au contexte des TI est susceptible d'être suffisant [...] parce que les concepts sont la pierre angulaire des théories. [...] Cette activité [de conceptualisation du concept] peut fournir d'importantes retombées en termes de contribution et d'impact (Barki, 2008, p.17, traduction libre).

À la lumière de nos résultats, nous suggérons de repenser la conceptualisation des concepts d'écosystème logiciel et de santé des écosystèmes logiciels.

Nos résultats ont montré que bon nombre d'auteurs utilisent des sources de données disponibles en ligne pour vérifier leurs hypothèses. Bien qu'elles soient des sources importantes de données qui aident les chercheurs à mieux explorer les ELCSO, et conscient de leurs avantages, il n'en demeure pas moins que les avis sur l'utilisation de ces sources sont divergents. Selon les auteurs utilisant *SourceForge.net*, il s'agit de l'un des plus grands référentiels de projets LCSO, bien que certains (p.ex. Fitzgerald, 2006; Colazo et Fang, 2009) estiment qu'une grande partie des données provenant de grands référentiels de données en ligne (p. ex. *SourceForge.net*, *GitHub.net*, *GHTorrent*) n'a peut-être que peu de

valeur ou peut être trompeuse. Fitzgerald (2006) prévoyait, il y a déjà presque 14 ans, que la recherche sur les LCSO pourrait être source de problème dans la mesure où les « chercheurs tirent parti de la disponibilité immédiate de grands référentiels de données en ligne (où une grande partie des données peuvent être de peu de valeur réelle), et continuent à concentrer des efforts de recherche sur le phénomène [naissant du LCSO] pour étudier à plusieurs reprises les caractéristiques des projets et la motivation des développeurs, par exemple » (p.597, traduction libre). C'est exactement le constat que nous avons pu faire dans notre revue de littérature. Les limitations de ces référentiels de données ont été également formulées par Sutanto *et al.* (2014). Ils citent notamment l'obsolescence de certains projets, l'inclusion des données de systèmes antérieurs et l'hébergement de certains projets dans plusieurs autres sources, différentes de celles sur lesquelles porte l'étude, ce qui poserait un problème de validité des résultats, car ils ne reflèteraient pas très exactement l'image réel des caractéristiques des projets étudiés.

Nous avons pu constater que la courbe des articles publiés par année présente une forme en cloche avec le pic des publications enregistré en 2012 et 2013. En effet, bien avant 2012, l'environnement commercial des LCSO a connu d'importants mouvements, qui non seulement ont engendré la fusion de plusieurs firmes mais aussi ont emmené certaines organisations telles que *Microsoft* à s'y intéresser. En effet, la firme *Sun Microsystems* a acquis en 2008 *MySQL AB*, propriétaire de la base de données à code source ouvert *MySQL* (« Sun to Acquire MySQL », 2008, paragr.1). En 2010, *Oracle* achète *Sun Microsystems* en acquérant ses droits d'auteur, brevets et marques commerciales pour devenir le propriétaire de la base de données « propriétaire » et de la base de données « ouverte » les plus populaires. On note cependant que les tentatives d'*Oracle* de commercialiser la base de données *MySQL* à code source ouvert ont suscité des inquiétudes au sein de la communauté LCSO (Thomson, 2011). S'agissant de *Microsoft*, elle crée une filiale en 2012, *Microsoft Open Technologies Inc.*, dont le but est de réduire l'écart entre les technologies « propriétaires » de *Microsoft* et les LCSO (Ovide, 2012). Cette filiale s'est vue replier très rapidement dans *Microsoft* lorsque la position de la compagnie sur les plates-formes LCSO et non *Windows* devenait de plus en plus favorable. Ces événements, pour ne prendre que ces deux exemples, auraient pu susciter auprès de la communauté de chercheurs la nécessité

d'explorer ces phénomènes, d'où le pic prononcé en 2012 et 2013 dans notre échantillon d'articles.

Le Tableau 9 a permis de recenser tous les aspects de santé et les composants de l'écosystème qui ont été étudiés dans notre échantillon d'articles. L'aspect de la santé et le composant le plus étudiés sont respectivement le succès et le projet. Ceci pourrait se comprendre dans la mesure où les projets sont le cœur des écosystèmes et en tant que solutions technologiques, le premier objectif recherché est la réussite ou le succès car les facteurs de succès font référence aux domaines clés sur lesquels les dirigeants se focalisent davantage pour atteindre les objectifs de performance de leurs projets (Fosso Wamba *et al.*, 2018). Notre revue de littérature a pu faire ressortir certains termes qui ne sont pas, à notre connaissance, abondamment étudiés par la communauté des chercheurs, par exemple, le terme « émergence » comme aspect de santé et le terme « ouverture (*opensourcing*) » comme composant de l'écosystème. Hahn *et al.* (2008) estiment que l'émergence d'une équipe de projet issue de communautés de développement du LCSO est un phénomène « naturel » qui suscite un intérêt pour étudier la manière dont les individus sont influencés par le réseau dans lequel ils sont intégrés. S'agissant des écosystèmes logiciels, leur émergence fait naître de nouveaux défis ou challenges pour les organisations qui s'y engagent; ce qui implique inexorablement de nouveaux modèles d'affaires qui auront un impact sur la structure organisationnelle et la concurrence ou encore affecteront les droits de propriété intellectuelle (Hanssen, 2012), pour ne citer que ces quelques exemples. Selon Hanssen (2012), le passage des organisations de processus fermés à des structures plus « ouvertes » (Mount et Fernandes, 2013), faisant intervenir la participation d'acteurs externes à l'organisation au développement de logiciels, est principalement dû à l'émergence des écosystèmes de logiciels dans l'industrie du logiciel. Laquelle « ouverture », devenue une tendance émergente, appelle les organisations à « sous-traiter » le développement de solutions innovantes à une communauté globale mais largement inconnue de développeurs de LCSO (Agerfalk et Fitzgerald, 2008). Ces derniers considèrent ce phénomène, qui fait usage du modèle de développement LCSO, comme une stratégie globale d'approvisionnement pour le processus de développement logiciel d'une organisation. Nous recommandons donc aux chercheurs d'étendre et de poursuivre les

études sur les aspects de la santé et les composants de l'écosystème qui n'ont pas été suffisamment étudiés, en particulier la santé générale au niveau de l'écosystème même.

Nos résultats ont relevé que certains des articles retenus dans notre échantillon ont été publiés dans des revues ne faisant pas partie du domaine SI, ni du domaine GL. Jadis inféodée au domaine technique, la recherche portant sur les LCSO intéresse aussi les chercheurs d'autres domaines moins techniques tels que Management et SI. Bien que l'évaluation de la santé puisse être envisagée sous différentes perspectives, il est à noter que les projets de développement de logiciels sont non seulement un processus technique, mais également un processus social impliquant de multiples parties prenantes dont les comportements ont une influence sur la performance desdits projets (Xu *et al.*, 2011; Vasilescu *et al.*, 2015). Nous recommandons dès lors à la communauté de chercheurs et de praticiens à développer davantage de collaborations entre eux sur des projets multidisciplinaires qui aideront à développer des cadres de référence et des outils pour l'évaluation de la santé des ECLSO.

L'analyse des données a fait émerger la théorie de la Vision organisante (*Organizing Vision*) comme étant la théorie qui a été la plus utilisée dans notre échantillon. Selon Swanson et Ramiller (1997), une vision organisante est définie comme le discours public véhiculé par une communauté focale sur l'application d'une technologie dans les organisations. Les LCSO sont des technologies innovantes qui envahissent notre quotidien et se positionnent comme un véritable moteur de l'innovation (Leclair, 2016; Marsan *et al.*, 2019). On constate que les trois articles qui ont utilisé cette théorie proviennent du même premier auteur, Marsan, J., dont la thèse a produit ces trois articles sur le rôle de la vision organisante du LCSO dans l'adoption et la diffusion de cette famille particulière de logiciels dans les organisations. Il serait intéressant de creuser davantage cette théorie ainsi que d'autres théories dans des recherches futures, si ce n'est de développer de nouvelles théories qui pourraient éclairer les chercheurs et les praticiens sur une meilleure compréhension du concept de santé des ECLSO.

4.2 Limites de l'étude

Notre travail de recherche présente des limites qui offrent des pistes pour des recherches futures. En dehors de la non-garantie de l'exhaustivité des données empiriques recueillies

et l'accessibilité à toutes les données possibles (Egger *et al.*, 2008), notre recherche s'est principalement appuyée sur la base de données *Web of Science*. Elle est considérée comme l'une des bases de données majeures pour la recherche d'articles et périodiques spécialisés en administration des affaires. L'utilisation de mots-clés dans les huit meilleures revues en SI (*Basket of 8*) nous a permis de recueillir d'autres articles pertinents. L'usage de ces deux sources de données ne nous a pas permis d'explorer avec exhaustivité l'ensemble de la littérature pertinente dans le domaine de la santé des ELCSO. L'ensemble des journaux et conférences spécialisés en administration des affaires n'y est pas indexé. De plus, utiliser des mots-clés présente également quelques limites dans la mesure où certains articles pertinents qui auraient pu apparaître dans nos résultats peuvent ne pas y être à cause du choix des mots-clés qui peut être différent d'un auteur à l'autre.

Une autre limite à relever réside dans la méthodologie employée pour la recherche des articles ayant cités les articles phares (*Forward search*). Bien que l'usage d'articles phares est une bonne pratique méthodologique lorsqu'on fait des revues de littérature systématique, il n'en demeure pas moins qu'il présente quelques biais en ce sens que des articles très pertinents pourraient avoir été exclus de la recherche parce que n'ayant pas cité un de nos articles phares.

4.3 Implications

Ce travail de recherche n'étant qu'un pas dans le domaine de la santé des ELCSO, nous estimons que les résultats de ce travail de revue de littérature systématique pourront inspirer les chercheurs. Plusieurs axes d'analyse sont à envisager car les résultats obtenus viennent enrichir la littérature dans un domaine immature où il existe encore un vide qu'il faille combler. Par ailleurs, nous pensons que la classification et la structuration des aspects de santé des ELCSO seraient une valeur ajoutée pour les parties prenantes directement impliquées dans les projets de développement de LCSO ou dans les ELCSO. En effet, Jansen (2014) souligne l'importance de la survie des ELCSO pour les différents acteurs de l'écosystème (programmeurs, utilisateurs finaux, investisseurs, participants) qui souhaitent savoir si leur écosystème est sain et performant. Ainsi, la santé des ELCSO ne préoccupe pas seulement les chercheurs mais également les praticiens. Les résultats pourraient donc orienter non seulement les gestionnaires de communautés ou des ELCSO, mais aussi les

contributeurs dans ces communautés ou ELCSO, de la nécessité de mieux gérer et contrôler leurs écosystèmes en identifiant des indicateurs de santé fiables et consensuels qui pourraient avoir un impact sur la santé, la survie et/ou la croissance dudit écosystème.

Sur le plan des recherches futures, nous espérons que ce travail sera une boussole qui orientera les travaux futurs dans le domaine de la santé des ELCSO vers l'exploration d'autres aspects de la santé et des composants de l'ELCSO jusque-là très peu étudiés. Plus précisément, ces résultats peuvent sensibiliser les chercheurs sur la nécessité de combler cet écart observé dans la littérature. Il serait pertinent que la revue de littérature soit étendue à d'autres bases de données et mots-clés. Cette revue pourrait être couplée à une étude qualitative auprès des experts et praticiens du domaine à l'effet d'avoir un aperçu de leur perception des concepts clés de la présente étude.

Conclusion

Un écosystème logiciel est constitué de plusieurs projets interreliés (Carillo, Marsan, et Negoita, 2017) dont la survie dépend inéluctablement du succès desdits projets. Selon plusieurs auteurs, l'état de l'écosystème ou de ses composants est matérialisé par leur « santé » (Amorim *et al.*, 2017b). Afin de fournir un aperçu des connaissances scientifiques sur la santé des ELCSO, l'objectif de ce travail était de structurer et d'analyser la littérature existante en identifiant l'état actuel de la recherche sur le domaine en question. Pour ce faire, nous avons élaboré deux questions de notre recherche qui nous ont orienté dans l'atteinte de notre objectif :

- Q1. Comment la santé des ELCSO est-elle définie dans la littérature ?
- Q2. Quels aspects de la santé des ELCSO, incluant leurs composants, ont été étudiés et, parmi ceux-ci, lesquels ont fait l'objet du plus grand nombre d'études ?

En effet, ce travail de recherche s'inscrit dans le cadre du projet interdisciplinaire et collaboratif *SECOHealth* auquel ont été impliquées plusieurs équipes de recherche du Canada, de la Belgique, de la France et d'Italie. Ledit projet avait pour objectif de contribuer à la recherche et à la pratique en proposant une méthodologie scientifique interdisciplinaire validée ainsi qu'un catalogue de directives et d'outils de recommandation pour améliorer la santé des écosystèmes logiciels (Mens *et al.*, 2017). Pour ce faire, nous avons réalisé une revue de littérature systématique descriptive en suivant scrupuleusement les étapes proposées par Templier et Paré (2015, 2018) et en s'inspirant des recommandations suggérées par Petersen, Vakkalanka, et Kuzniarz (2015). Ainsi, nous répondions aux appels à la recherche de la part de Jansen (2014), Mens *et al.* (2017) et Marois *et al.* (2018a) concernant la santé des ELCSO et de Manikas et Hansen (2013a) concernant la santé des écosystèmes logiciels en général. Ce faisant, nous pensons qu'une analyse descriptive peut affirmer que ses conclusions représentent les faits probants ou l'état de connaissances d'un domaine de recherche (King et He, 2005).

Pour mener ce travail, nous nous sommes appuyés sur 15 articles phares méthodiquement et minutieusement sélectionnés, sur lequel nous avons fait un *forward search*. Parallèlement nous avons procédé à une recherche par mots-clés dans les huit (8) meilleures revues

scientifiques (*Basket of 8*) en SI (Lowry *et al.*, 2014). Au terme de ce processus, nous avons retenu 98 articles sur lesquels les données étaient extraites et codifiées.

Les résultats ce travail de recherche nous informent qu'aucun article n'a défini explicitement les concepts de « santé » et « écosystème ». Ce qui corrobore les idées de Mens *et al.* (2017) qui relèvent que l'une des principales raisons du non-consensus est dû aux fréquents changements auxquels sont confrontées la dynamique interne et les interactions des composants d'un écosystème, ou tout au moins à l'évolution due à une combinaison d'événements internes et externes, tels que des changements technologiques ou environnementaux, ou l'apparition de systèmes concurrents mobiles. Étant un système libre et ouvert, cette difficulté résiderait également du fait que les ELCSO ne disposent pas en leur sein d'un management centralisé qui favoriserait la surveillance de la santé de l'écosystème (Adams *et al.*, 2016). Nos investigations ont montré que les aspects de santé et les composants d'écosystème les plus étudiés dans la littérature sont respectivement le « succès » et le « projet ». Cette recherche révèle effectivement que la littérature portant sur le domaine de la santé des ECLSO reste encore embryonnaire et qu'il est indispensable que la communauté des chercheurs s'y intéresse davantage. Il s'agit d'une sonnette d'alarme car

de nos jours, 95% des entreprises exploitent des ressources de LCSO non triviales dans le cadre de leurs charges de travail informatiques critiques, qu'elles le sachent ou non (p. ex., indirectement via un logiciel propriétaire commercial utilisant des bibliothèques LCSO). [...] De plus, les innovations technologiques émergent continuellement avec de fortes synergies avec le LCSO et continuent de jouer un rôle moteur dans l'innovation (Gartner, 2019, paragr.1, traduction libre).

Nos résultats doivent être interprétés avec prudence en raison de certaines limitations ne permettant pas une généralisation. Premièrement, nous avons choisi de nous limiter à deux sources de données pour l'identification des articles. Deuxièmement, la méthode du *forward search* ne permet pas d'identifier tous les articles car certains peuvent ne pas avoir cité nos articles phares choisis. Sur le plan théorique, notre étude permet d'enrichir la littérature existante en posant certaines bases pour des recherches futures. Sur le plan pratique, la classification et la structuration des aspects de santé des ELCSO apporterait une valeur ajoutée pour les parties prenantes directement impliquées dans les projets de développement de LCSO ou dans les ELCSO.

Bibliographie

- Agerfalk, P. J., & Fitzgerald, B. (2008). Outsourcing to an unknown workforce: Exploring open sourcing as a global sourcing strategy. *MIS Quarterly*, 32(2), 385-409.
- Ahmed, I., Forrest, D., & Jensen, C. (2017). A Case Study of Motivations for Corporate Contribution to FOSS. *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 223-231.
- Aksulu, A., & Wade, M. (2010). A Comprehensive Review and Synthesis of Open Source Research. *Journal of Association for Information Systems*, 11(Special issue), 576-656.
- Amorim, S. d., McGregor, J. D., Almeida, E. S., & Chavez, C. v. (2017a). Software ecosystems' architectural health: another view. *JSOS '17 Proceedings of the Joint 5th International Workshop on Software Engineering for Systems-of-Systems and 11th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems* (pp. 66-69). Buenos Aires, Argentina: IEEE Press Piscataway.
- Amorim, S. d., Neto, F. S., McGregor, J. D., Santana de Almeida, E., & Chavez, C. G. (2017b). How Has the Health of Software Ecosystems Been Evaluated? A Systematic Review. *31st Brazilian Symposium on Software Engineering (SBES)* (pp. 14-23). Fortaleza: Association for Computing Machinery (ACM). doi:10.1145/3131151.3131174
- Amrit, C., & Hillegersberg, J. v. (2010). Exploring the impact of socio-technical core-periphery structures in open source software development. *Journal of Information Technology*, 25, 216–229. doi:10.1057/jit.2010.7
- Amrollahi, A., & Khansari, M. (2011). A Framework for Open Source Research Categorization based on Stakeholder Analysis. *International Conference on Software and Information Engineering (ICSIE 2011)*, 625-630.
- Angeren, J. v., Kabbedijk, J., Jansen, S., & Popp, K. (2011). A survey of associate models used within large software ecosystems. In: Jansen, Bosch, Ahmed, & Campell (Éds.) *Proceedings of the 3rd CEUR-WS*, pp. 27–39.
- Barki, H. (2008). Thar's Gold in Them Thar Constructs. *The Database for Advances in Information Systems*, 39(3), 9-20. doi:10.1145/1390673.1390677
- Brandao Gomes da Silva, A. C., Carneiro, G. d., Brito e Abreu, F., & Monteiro, M. P. (2017). Frequent Releases in Open Source Software: A Systematic Review. *Information*, 8(3), 1-18.
- Capra, E., Francalanci, C., Merlo, F., & Lamastra, C. R. (2009). A Survey on Firms' Participation in Open Source Community Projects. In: Boldyreff, Crowston, Lundell, & Wasserman (Éds.) *Open Source Ecosystems: Diverse Communities Interacting. OSS 2009. IFIP Advances in Information and Communication Technology*, 299, pp. 225–236. doi:10.1007/978-3-642-02032-2_20
- Carillo, K. D., & Marsan, J. (2016). The Dose makes the Poison - Exploring the Toxicity Phenomenon in Online Communities. *International Conference Information Systems*, 1-18.

- Carillo, K., Marsan, J., & Negoita, B. (2017). Exploring the biosphere - Towards a conceptualization of peer production communities as living organisms. *SIGOPEN 2017 Developmental Workshop for Openness Research, ICIS*, (pp. 1-12). Seoul.
- Chesbrough, H. W. (2007). 'Why companies should have open business models'. *MIT Sloan Management Review*, 48(2), 22–27.
- Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1), 37-46. doi:10.1177/001316446002000104
- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (éd. 2nd). Hillsdale, NJ: Erlbaum.
- Colazo, J., & Fang, Y. (2009). Impact of License Choice on Open Source Software Development Activity. *Journal of the American Society for Information Science and Technology*, 60(5), 997-1011.
- Colombo, M. G., Cumming, D., Mohammadi, A., Rossi-Lamastra, C., & Wadhwa, A. (2016). Open business models and venture capital finance. *Industrial and Corporate Change*, 25(2), 353–370. doi:10.1093/icc/dtw001
- Comscore MobiLens service. (2010). *comScore-Reports-February-2010-US-Mobile-Subscriber-Market-Share*. Récupéré sur Site web ComScore, Inc.: <https://www.comscore.com>
- Comscore MobiLens service. (2014). *Comscore Reports April 2014 U.S. Smartphone Subscriber Market Share*. Récupéré sur Site Web Comscore, Inc.: <https://www.comscore.com/>
- Crowston, K., & Howison, J. (2006). Assessing the health of open source communities. *IEEE Comput.*, 39(5), 89–91.
- Crowston, K., & Wade, M. (2010). Introduction to JAIS Special Issue on Empirical Research on Free/Libre Open Source Software. *Journal of the Association for Information Systems (JAIS)*, 11(numéro spécial), i-v.
- Crowston, K., Wei, K., Howison, J., & Wiggins, A. (2012). Free/Libre open-source software development: What we know and what we do not know. *ACM Computing Surveys (CSUR)*, 44(2), 1-35.
- Daniel, S. L., Maruping, L. M., Cataldo, M., & Herbsleb, J. (2018). The Impact of Ideology Misfit on OSS Communities and Companies. *MIS Quarterly*, 42(4), 1069-1096. doi:10.25300/MISQ/2018/14242
- Daniel, S., & Stewart, K. (2016). Open source project success: Resource access, flow, and integration. *Journal of Strategic Information Systems*, 159–176.
- Dijkers, J., Sincic, R., Wasankhasit, N., & Jansen, S. (2018). Exploring the Effect of Software Ecosystem Health on the Financial Performance of the Open Source Companies. (SoHeal'18, Éd.) *SoHeal'18:IEEE/ACM 1st International Workshop on Software Health*, pp. 48-55. doi:10.1145/3194124.3194126
- Egger, M., Dickersin, K., & Smith, G. D. (2008). Problems and Limitations in Conducting Systematic Reviews. In: Egger, Smith & Altman (Éds.), *Systematic Reviews in Health Care: Meta-Analysis in Context, Second Edition* (pp. 43-68). Wiley Blackwell. doi:10.1002/9780470693926.ch3

- Faul, F., Erdfelder, E., Lang, A.-G., & Buchner, A. (2007). G*Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior Research Methods*, 39(2), 175-191.
- Fitzgerald, B. (2006). The Transformation of Open Source Software. *MIS Quarterly*, 30(3), 587-598.
- Fosso Wamba, S., Kala Kamdjoug, J. R., Akter, S., & Carillo, K. (2018). ERP Adoption and Use in Production Research: An Archival Analysis and Future Research Directions. *I3E 2018*, 1-18.
- Foulonneau, M., Pawelzik, R., Grégoire, B., & Donak, O. (2013). Analyzing the open source communities' lifecycle with communication data. *Proceedings of the 5th MEDES, ACM*, pp. 340–344.
- Franco-Bedoya, O., Ameller, D., Costal, D., & Franch, X. (2014). QuESo: modèle de qualité pour les écosystèmes de logiciels open source. *Actes de la neuvième ICSoft-EA*, pp. 209–221.
- Franco-Bedoya, O., Ameller, D., Costal, D., & Franch, X. (2017). Open source software ecosystems: A Systematic mapping. *Information and Software Technology*, 91, 160–185.
- Franzblau, L. E., & Chung, K. C. (2012). Graphs, Tables, and Figures in Scientific Publications: The Good, the Bad, and How Not to Be the Latter. *The Journal of Hand Surgery*, 37(3), 591-596.
- Gamalielsson, J., Lundell, B., & Mattsson, A. (2011). Open source software for model driven development: a case study. Dans S. Hissam, B. Russo, M. de Mendonça Neto, & F. Kon (Éd.), *Open Source Systems: Grounding Research. IFIP Advances in Information and Communication Technology*. 365, pp. 348–367. Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-24418-6_30
- Gartner. (2019). *Hype Cycle for Open-Source Software, 2019*. Récupéré sur Site web de Gartner: <https://www.gartner.com>
- Germonprez, M., Link, G. J., Lombard, K., & Goggins, S. (2018). Eight Observations and 24 Research Questions About Open Source Projects: Illuminating New Realities. *Proceedings of the ACM on Human-Computer Interaction - CSCW*. 2, pp. 1-22. New York, NY: ACM. doi:10.1145/3274326
- Ghapanchi, A. H. (2015). Investigating the Interrelationships among Success Measures of Open Source Software Projects. *Journal of Organizational Computing and Electronic Commerce*, 25(1), 28-46. doi:10.1080/10919392.2015.990775
- Ghapanchi, A. H., & Aurum, A. (2010). A Framework to Represent Antecedents of User Interest in Open-Source Software Projects. *International Business Information Management Conference (IBIMA)*, 542-553.
- Hahn, J., Moon, J. Y., & Zhang, C. (2008). Emergence of New Project Teams from Open Source Software Developer Networks : Impact of Prior Collaboration Ties. *Information Systems Research*, 19(3), 369-391. doi:10.1287/isre.1080.0192
- Hanssen, G. K. (2012). A longitudinal case study of an emerging software ecosystem. Implications for practice and theory. *The Journal of Systems and Software*, 85, 1455–1466.

- Hartigh, E. d., Tol, M., & Visscher, W. (2006). The Health Measurement of a Business Ecosystem. *In: Proceedings of the European Network on Chaos and Complexity Research and Management Practice Meeting*, (pp. 1-39). Bergen aan Zee.
- Hauge, Ø., Ayala, C., & Conradi, R. (2010). Adoption of open source software in software-intensive organizations – A systematic literature review. *Information and Software Technology*, 52, 1133–1154.
- Hill, G., Datta, P., & Vander Weerd, C. (2017). Developers, Quality Control and Download Volume in Open Source Software (OSS) Projects. *Journal of Organizational and End User Computing*, 29(2), 43-66. doi:10.4018/JOEUC.2017040103
- Hyrnsalmi, S., Suominen, A., Mäkilä, T., & Knuutila, T. (2016). Mobile Application Ecosystems: An Analysis of Android Ecosystem. . *Encyclopedia of E-Commerce Development, Implementation, and Management*, 1418-1434.
- Iansiti, M., & Levien, R. (2002). Keystones and dominators: Framing operating and technology strategy in a business ecosystem. *Working paper /Division of Research*. Harvard Business School.
- Iansiti, M., & Levien, R. (2004a). *The Keystone Advantage: What the New Dynamics of Business Ecosystems Mean for Strategy Innovation and Sustainability*. Boston: Harvard Business Press.
- Iivari, N. (2010). Discursive construction of 'user innovations' in the open source software development context. *Information and organization*, 20(2), 111-132. doi:10.1016/j.infoandorg.2010.03.002
- Jabeur, L., Tamine, L., & Boughanem, M. (2010). A social model for literature access: towards a weighted social network of authors. *Proceedings of 9th RIAO (Recherche d'Information Assistée par Ordinateur)* (pp. 32–39). Paris: Bibliotheque Nationale de France.
- Jabeur, L., Tamine, L., & Boughanem, M. (2010). A social model for literature access: towards a weighted social network of authors. *International Conference on Adaptivity, Personalization and Fusion of Heterogeneous Information (RIAO 2010)* (pp. 32–39). Paris, France: Bibliotheque Nationale de France.
- Jansen, S. (2014). Measuring the health of open source software ecosystems: Beyond the scope of project health. *Information and Software Technology*, 56(11), 1508–1519.
- Jansen, S., & Cusumano, M. (2013). Defining software ecosystems: a survey of software platforms and business network governance. *Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry*, 41-58.
- Jansen, S., Brinkkemper, S., Souer, J., & Luinenburg, L. (2012). Shades of gray: opening up a software producing organization with the open software enterprise model. *Journal of Systems and Software*, 85(7), 1495–1510.
- Jansen, S., Finkelstein, A., & Brinkkemper, S. (2009b). A sense of community: A research agenda for software ecosystems. *31st International Conference on Software Engineering - Companion Volume* (pp. 187–190). Vancouver : IEEE.
- Kamei, Y., Matsumoto, S., Maeshima, H., Onishi, Y., Ohira, M., & Matsumoto, K.-i. (2008). Analysis of coordination between developers and users in the apache community. *4th International Conference on Open Source Systems held at the 20th World Computer Congress* (pp. 81–92). Milan: IFIP.

- King, W. R., & He, J. (2005). Understanding the role and methods of meta-analysis in IS research. *Communications of the Association for Information Systems*, 16, 665-686.
- Kitchenham, B., & Charters, S. (2007). *Guidelines for Performing Systematic Literature Reviews in Software Engineering, Version 2.3*. EBSE Technical Report, Keele University et University of Durham, Keele, UK; Durham, UK.
- Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1), 159-174.
- LeClair, H. (2016). *Future-open-source-survey*. Récupéré sur Site web Opensource.com: <https://opensource.com>
- Liao, Z., Yi, M., Wang, Y., Liu, S., Liu, H., Zhang, Y., & Zhou, Y. (2019). Healthy or Not: A Way to Predict Ecosystem Health in GitHub. *Symmetry-Basel*, 11(2), 1-16. doi:10.3390/sym11020144
- Lowry, P. B., Moody, G. D., Gaskin, J., Galletta, D. F., Humpherys, S., Barlow, J. B., & and Wilson, D. W. (2014). Evaluating Journal Quality and the Association for Information Systems Senior Scholars' Journal Basket Via Bibliometric Measures: Do Expert Journal Assessments Add Value? *MIS Quarterly*, 37(4), 993-1012.
- Lucassen, G., van Rooij, K., & Jansen, S. (2013). Ecosystem Health of Cloud PaaS Providers. In: G. Herzwurm, & T. Margaria (Éds.), *4th International Conference on Software Business (ICSOB)*. 150, pp. 183-194. Univ Potsdam, Campus Griebnitzsee, Potsdam, GERMANY: Springer-Verlag.
- Lundella, B., Lings, B., & Syberfeldt, A. (2011). Practitioner perceptions of Open Source software in the embedded systems area. *The Journal of Systems and Software*, 84, 1540-1549.
- Lungu, M., Lanza, M., Gîrba, T., & Robbes, R. (2010a). The small project observatory: Visualizing software ecosystems. *Science of Computer Programming*, 75(4), 264-275.
- Macredie, R. D., & Mijinyawa, K. (2011). A theory-grounded framework of Open Source Software adoption in SMEs. *European Journal of Information Systems*, 20, 237-250. doi:10.1057/ejis.2010.60;
- Manikas, K. (2016). Revisiting software ecosystems Research: A longitudinal literature study. *The Journal of Systems and Software*, 117, 84-103.
- Manikas, K., & Hansen, K. M. (2013a). Reviewing the Health of Software Ecosystems – A Conceptual Framework Proposal. *5th Workshop on Software Ecosystems (IWSECO)*, (pp. 33-44). Postdam, Germany.
- Manikas, K., & Hansen, K. M. (2013c). Software ecosystems – A systematic literature review. *Journal of Systems and Software*, 86(5), 1294-1306. doi:10.1016/j.jss.2012.12.026
- Margan, D., & Candric, S. (2015). The success of open source software: A review. *38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1463-1468). Opatija, Croatia: IEEE. doi:10.1109/MIPRO.2015.7160503
- Marois, P., Marsan, J., Mens, T., & Templier, M. (2018b). L'implication des entreprises dans les écosystèmes logiciels open source : Une analyse par les concepts des

- écosystèmes naturels. *Journée Pré-ICIS Francophone de l'Association Information & Management (AIM)*. San Francisco.
- Marois, P., Mopenza, G. L., Marsan, J., & Templier, M. (2018a). La santé des écosystèmes logiciels ouverts : Une enquête qualitative auprès d'experts (recherche en cours). *23ème Conférence de l'Association Information & Management (AIM)*, (pp. 1-7). Montréal.
- Marsan, J., & Paré, G. (2013). Antecedents of Open Source Software Adoption in Health Care Organizations: A Qualitative Survey of Experts in Canada. *International Journal of Medical Informatics*, 82(8), 731-741.
- Marsan, J., Paré, G., & Beaudry, A. (2012). Adoption of open source software in organizations: A socio-cognitive perspective. *Journal of Strategic Information Systems*, 21, 257-273.
- Marsan, J., Templier, M., Marois, P., Adams, B., Carillo, K., & Mopenza, G. L. (2019). Towards solving social and technical problems in open source software ecosystems : using cause-and-effect analysis to disentangle the causes of complex problems. *IEEE Software*, 36(1), 34 - 41. doi:10.1109/MS.2018.2874323
- Martinez-Torres, M., & Diaz-Fernandez, M. (2014). Current issues and research trends on open- source software communities. *Technology Analysis & Strategic Management*, 26(1), 55-68.
- Mens, T., Adams, B., & Marsan, J. (2017). Towards an interdisciplinary, socio-technical analysis of software ecosystem health. *BElgian-NEtherlands software eVOLution (BENEVOL) symposium*. Antwerp, Belgique.
- Messerschmitt, D., & Szyperski, C. (2003). *Software Ecosystem: Understanding an Indispensable Technology and Industry* (éd. 1ère, Vol. 1). Cambridge, MA, USA: MIT Press Books.
- Miles, M. B., & Huberman, A. M. (1994). *Qualitative Data Analysis: An Expanded Sourcebook*. Thousand Oaks, California, USA: SAGE Publications.
- Mockus, A., Fielding, R., & Herbsleb, J. (2002). Two case studies of open source software development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology*, 11(3), 309–346. doi:10.1145/567793.567795
- Morgan, B. (2017). *what-is-customer-experience?* Récupéré sur Site web Forbes Media LLC: <https://www.forbes.com>
- Morgan, L., & Finnegan, P. (2014). Beyond free software: an exploration of the business value of strategic open source. *The Journal of Strategic Information Systems*, 23(3), 226–238.
- Mount, M. P., & Fernandes, K. (2013). Adoption of free and open source software within high-velocity firms. *Behaviour & Information Technology*, 32(3), 231–246. doi:10.1080/0144929X.2011.596995
- Nassl, M., & Löffler, J. (2015). Ecosystem services in coupled social–ecological systems: Closing the cycle of service provision and societal feedback. *Ambio*, 44(8), 737–749. doi:10.1007/s13280-015-0651-y
- Nelson, M., Sen, R., & Subramaniam, C. (2006). Understanding Open Source Software: A Research Classification Framework. *Communications of the Association for Information Systems*, 17, 266-287.

- Niederman, F., Davis, A., Greiner, M. E., Wynn, D., & York, P. T. (2006). A Multi-Level Research Agenda for Studying Open Source I: A Framework. *Communications of the Association for Information Systems*, 18, 129-149.
- Open Source Initiative. (2007). *The Open Source Definition*. Récupéré sur Site web Opensource.org: <https://opensource.org>
- Ostrom, E. (2009). A general framework for analyzing sustainability of social-ecological systems. *Science*, 325, 419–422. doi:10.1126/science.1172133
- Ovide, S. (2012). *Microsoft Dips Further Into Open-Source Software*. Récupéré sur Site web The World street journal: <https://www.wsj.com/>
- Paré, G., Trudel, M. C., Jaana, M., & Kitsiou, S. (2015). Synthesizing information systems knowledge: A typology of literature reviews. *Information & Management*, 52(2), 183-199.
- Perens, B. (1999). The Open Source Definition. In: C. DiBona, S. Ockman, & M. Stone (Éds.) *Open Sources: Voices from the Open Source Revolution*, pp. 171-188.
- Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for Conducting Systematic Mapping Studies in Software Engineering: An Update. *Information and Software Technology*, 64, 1-18.
- Ramanathan, L., & Iyer, S. K. (2015). A Qualitative Study on the Adoption of Open Source Software in Information Technology Outsourcing Organizations. *11th International Conference on Open Source Systems (OSS)*, 103-113. doi:10.1007/978-3-319-17837-0_10
- Raymond, B. (2018). L'influence des artéfacts TI sur la conception et la performance des routines organisationnelles. *Actes de la 23e Conférence de l'Association Information & Management (AIM 2018)* (pp. 1-18). Montréal, Canada: AIM.
- Raymond, E. S. (2001). *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, Revised Edition*. (T. O'Reilly, Éd.) Sebastopol, CA, United States of America: O'Reilly & Associates, Inc.
- Roberts, J., Hann, I.-H., & Slaughter, S. (2006). Understanding the motivations, participation, and performance of open source software developers: a longitudinal study of the apache projects. *Management Science*, 52(7), 984–999.
- Ruff, N. A. (2016). *Trends in Corporate Open Source Engagement*. Récupéré sur Site web Open Health Marketplace: <http://www.openhealthnews.com>
- Saeed, S., Fatima, U., & Iqbal, F. (2018). A review of Requirement Elicitation techniques in OSSD. *IJCSNS International Journal of Computer Science and Network Security*, 18(3), 86-92.
- Scacchi, W. (2007). Free/open source software development. *ACM*, (pp. 459-468). Dubrovnik.
- Sen, R., Singh, S. S., & Borl, S. (2012). Open source software success: Measures and analysis. *Journal of Decision Support Systems*, 52(2), 364-372.
- Shaikh, M. (2016). Negotiating open source software adoption in the UK public sector. *Government Information Quarterly*, 33, 115–132.

- Shaikh, M., & Cornford, T. (2010). 'Letting go of Control' to Embrace Open Source: Implications for Company and Community. *43rd Hawaii International Conference on Systems Sciences (HICSS 2010)*, (pp. 2837-2846). Honolulu.
- Squire, M., & Williams, D. (2012). Describing the software forge ecosystem. *Proceedings of the 45th HICSS*, pp. 3416–3425.
- Stackoverflow. (2018). *Developer Survey Results 2018*. Récupéré sur Site web Stackoverflow: <https://insights.stackoverflow.com>
- Stewart, K. J., Darcy, D. P., & Daniel, S. L. (2006). Opportunities and Challenges Applying Functional Data Analysis to the Study of Open Source Software Evolution. *Statistical Science*, *21*(2), 167–178. doi:10.1214/088342306000000141
- Sun to Acquire MySQL. (2008). Récupéré sur Site Web MySQL, Archives de <http://mysql.com/news-and-events/sun-to-acquire-mysql.html>: <https://web.archive.org/>
- Sutanto, J., Kankanhalli, A., & Tan, B. C. (2014). Uncovering the relationship between OSS user support networks and OSS popularity. *Decision Support Systems*, *64*, 142-151.
- Swanson, E., & Ramiller, N. (1997). The organizing vision in information systems innovation. *Organization Science*, *8*, 458–474.
- Swanson, E., & Ramiller, N. (2004). Innovating mindfully with information technology. *MIS Quarterly*, *28*, 553–583.
- Teixeira, J. A., & Hyrynsalmi, S. (2017). How do Software Ecosystems Co-Evolve? A view from OpenStack and beyond (Research working paper). *8th International Conference on Software Business (ICSOB 2017)*, (pp. 1-18). Essen.
- Templier, M., & Paré, G. (2015). A Framework for Guiding and Evaluating Literature Reviews. *Communications of the Association for Information Systems*, *37*, 112-137.
- Templier, M., & Pare, G. (2018). Transparency in literature reviews: an assessment of reporting practices across review types and genres in top IS journals. *European Journal of Information Systems*, *27*(5), 503-550. doi:10.1080/0960085X.2017.1398880
- Thomson, I. (2011). *Oracle offers commercial extensions to MySQL*. Récupéré sur Site web The Register: <https://www.theregister.co.uk/>
- Tidd, J., & Bessant, J. (2018). *Managing Innovation: Integrating Technological, Market and Organizational Change*. 5th ed. Chichester: Wiley. (éd. 6th). Hoboken, NJ, USA: Wiley.
- Toivanen, T., Mazhelis, O., & Luoma, E. (2015). Network Analysis of Platform Ecosystems: The Case of Internet of Things Ecosystem. *6th International Conference on Software Business (ICSOB)*. 210, pp. 30-44. Braga, Portugal: Lecture Notes in Business Information Processing.
- Vasilescu, B., Filkov, V., & Serebrenik, A. (2015). Perceptions of Diversity on Git Hub: A User Survey. *roceedings of the Eighth International Workshop on Cooperative and Human Aspects of Software Engineering* (pp. 50-56). Florence: IEEE Press. doi:10.1109/CHASE.2015.14

- Vaughan-Nichols, S. J. (2015). *It's an open-source world: 78 percent of companies run open-source software*. Consulté le octobre 20, 2019, sur Site web ZDNet: <https://www.zdnet.com/>
- Victor, G. (2018). WASP (Write a Scientific Paper) using Excel – 2: Pivot tables. *Early Human Development*, 117, 104-109.
- von Krogh, G., & von Hippel, E. (2006). The promise of research on open source software. *Organisation Science*, 52(7), 975–983.
- Wang, J. (2012). Survival factors for Free Open Source Software projects: A multi-stage perspective. *European Management Journal*, 30(4), 352-371.
- Wu, J., & Tang, Q. (2007). Analysis of Survival of Open Source Projects: a Social Network Perspective. *Pacific Asia Conference on Information Systems (PACIS) 2007 Proceedings*, pp. 55-68.
- Xu, B., Lin, Z., & Xu, Y. (2011). A Study of Open Source Software Development from Control Perspective. *Journal of Database Management*, 22(1), 26-42.
- Xu, J., Christley, S., & Madey, G. (2006). Application of social network analysis to the study of open source software. In: J. Bitzer, & P. J. Schröder (Éds.) *The economics of open source software development*, pp. 247–269.
- Yu, L., Ramaswamy, S., & Bush, J. (2007). Software evolvability: an ecosystem point of view. *Proceedings of the 3rd International IEEE Workshop on Software Evolvability*, pp. 75–80. doi:10.1109/SE.2007.8
- Yu, L., Schach, S., Chen, K., & Offutt, J. (2004). Categorization of common coupling and its application to the maintainability of the Linux kernel. *IEEE Transactions on Software Engineering*, 30(10), 694-706. doi:10.1109/TSE.2004.58
- Zhussupova, A., & Rahman, A. A. (2011). Open source software adoption in public organizations of Kazakhstan. *IEEE Conference on Open Systems (ICOS2011)*, 417-422.

Annexes

Annexe 1 : Protocole de recherche pour les articles en système d'information

(dans le cadre du projet *SECOHealth* dans lequel s'inscrit ce mémoire)

Étape 1 : Citations des « papiers phares » :

- **Fichier à consulter** : « papiers phares ».
- **Portée** : les articles sélectionnés par les experts du projet *SECOHealth*.
- **Base de données à utiliser** : Web of science (WoS)
- **Instruction** : pour chaque article, indiquer le nombre total de citations et aussi par année (sous forme de graphique si possible).

Étape 2 : Recherche des articles ayant cités les articles phares (*Forward search*)

- **Portée** : les articles phares sélectionnés
- **Base de données à utiliser** : Web of science (WoS)
- **Instruction** : pour chaque article, considérer tous les articles ayant été publiés après chaque article phare et l'ayant cité.
 - **Zotero** : supprimer les doublons
 - **Quick read (titre et résumé)** : pour inclusion / exclusion
 - **critères d'exclusion** :
 - a. articles n'étant pas écrits en anglais;
 - b. éditoriaux;
 - c. revues de littérature;
 - d. indisponibilité du texte intégral.
 - **critères d'inclusion**
 - a. articles de conférence ou de revue scientifique revus par les pairs;
 - b. articles mentionnant au moins un aspect de la santé du LCSO/ELCSO.
 - **test de fiabilité intercodeurs** : validation du processus d'inclusion/exclusion
 - **Raison** : Accepté ou rejeté

Étape 3 : Recherche par mots-clés dans les huit meilleures revues en SI (*Basket of 8*)

- **Bases de données** :
 - ABI/INFORM Global et Web of Science
- **Chaîne de mots-clés**

“open source” OR “free software” OR “libre software” OR “OSS” OR
“FLOSS” OR “F/LOSS” OR “FOSS” OR “F/OSS”

- **Instruction** : considérer tous les articles ayant été publiés entre 2006 et 2018
 - **Zotero** : supprimer les doublons
 - **Quick read (titre et résumé)** : pour inclusion / exclusion
 - **critères d'exclusion** :
 - a. éditoriaux;
 - b. articles n'étant pas écrits en anglais;
 - c. revues de littérature;
 - d. indisponibilité du texte intégral.
 - **critères d'inclusion**
 - a. articles de conférence ou de revue scientifique revus par les pairs;

b. articles mentionnant au moins un aspect de la santé du LCSO ou du ELCSO.

- **Fusion : les deux Zotero et extraction des doublons**

Étape 4 : Lecture complète (*Full read*)

- **Portée** : Tous les articles sélectionnés
- **Instruction** : Pour chaque article, considérer UNIQUEMENT les articles qui mentionnent au moins un aspect de la santé du LCSO ou de l'ELCSO
- **Extraction et analyse** : données issues des articles retenus après l'instruction du *full read*

Annexe 3 : Résultats des tests de fiabilité intercodeurs

1er test					
		Auteur de ce mémoire			Number of observed agreements: 18 (90.00% of the observations)
		OUI	NON		Number of agreements expected by chance: 10.1 (50.50% of the observations)
Intercodeur 1	OUI	11	1	11	Kappa= 0.798
	NON	0	8	9	SE of kappa = 0.135
		11	9		95% confidence interval: From 0.532 to 1.000
					The strength of agreement is considered to be 'good'.
2e test					
		Auteur de ce mémoire			Number of observed agreements: 16 (80.00% of the observations)
		OUI	NON		Number of agreements expected by chance: 10.5 (52.50% of the observations)
Intercodeur 2	OUI	11	4	15	Kappa= 0.579
	NON	0	5	5	SE of kappa = 0.171
		11	9		95% confidence interval: From 0.244 to 0.914
					The strength of agreement is considered to be 'moderate'

Annexe 4: Articles retenus dans notre échantillon

Domaine SI

Conférence

Pacific Asia Conference on Information Systems (PACIS)

Wu, J. et Tang, Q. (2007)

Scandinavian Conference on Information Systems (SCIS)

Iivari, N. (2013)

Journal

Behaviour & Information Technology

Mount, M.P. et Fernetes, K. (2013)

Creativity and Innovation Management

Chan, J. et Husted, K. (2010)

Decision Support Systems

Cai, Y. et Zhu, D. (2016)

Gwebu, K.L. et Wang, J. (2011)

Li, J.P., Chen, R., Lee, J. et Rao, H.R. (2013)

Subramaniam, C., Sen, R. et Nelson, M.L. (2009)

Sutanto, J., Kankanhalli, A. et Tan, B.C.Y. (2014)

European Journal of Information Systems

Macredie, R. D. et Mijinyawa, K. (2011)

Vitari, C. et Ravarini, A. (2009)

European Management Journal

Wang, J. (2012)

Government Information Quarterly

Shaikh, M. (2016)

Information & Management

Xu, B., Jones, D. R. et Shao, B. (2009)

Information Systems

Ghapanchi, A.H. (2015)

Information systems Research

Daniel, S., Agarwal, R. et Stewart, K. J. (2013)

Hahn, J., Moon, J. Y. et Zhang, C. (2008)

Ho, S. Y. et Rai, A. (2017)

Setia, P., Rajagopalan, B., Sambamurthy, V. et Calantone, R. (2012)

Stewart, K.J., Ammeter, A.P. et Maruping, L.M. (2006)

Wen, W., Forman, C. et Graham, S. J. H. (2013)

Zhang, C., Hahn, J. et De, P. (2013)

Information Technology & People

Iivari, N. (2009)

Nevo, S. et Chengalur-Smith, I. (2017)

Information Systems Management

Ghapanchi, A. H. et Tavana, M. (2015)

Ven, K. et Verelst, J. (2012)

International Journal of Medical Informatics

Marsan, J. et Paré, G. (2013)

Journal of the Association for Information Systems (JAIS)

Chengalur-Smith, I., Sidorova, A. et Daniel, S. (2010)

Choi, N., Chengalur-Smith, I. et Nevo, S. (2015)

Ke, W. et Zhang, P. (2010)

Journal of Business Research

Wang, J., Hu, M. Y. et Shanker, M. (2012)

Journal of Information Technology

Amrit, C. et van Hillegersberg, J. (2010)

Journal of Management Information Systems

Fang, Y. et Neufeld, D. (2009)

Moqri, M., Mei, X., Qiu, L. et Betyopadhyay, S. (2018)

Sen, R., Subramaniam, C. et Nelson, M. L. (2008)

Journal of Strategic Information Systems

Carillo, K., Huff, S. et Chawner, B. (2017)

Daniel, S. et Stewart, K. J. (2016)

Daniel, S., Midha, V., Bhattacharjee, A. et Singh, S. P. (2018)

Marsan, J., Paré, G. et Beaudry, A. (2012)

Peng, G., Wan, Y. et Woodlock, P. (2013)

Santos, C., Kuk, G., Kon, F. et Pearson, J. (2013)

Journal of The American Society for Information Science and Technology

Colazo, J. et Fang, Y. (2009)

Library Hi Tech

Choi, N. (2014)

MIS Quarterly

Ågerfalk, J. et Fitzgerald, B. (2008)

Daniel, S. L., Maruping, L. M., Cataldo, M. et Herbsleb, J. (2018)

Fitzgerald, B. (2006)

Howison, J. et Crowston, K. (2014)

Singh, P. V. Tan, Y. et Mookerjee, V. (2011)

Stewart, K. J. et Gosain, S. (2006)

von Krogh, G., Haefliger, S., Spaeth S. et Wallin, M. W. (2012)

Program

Cherukodan, S. et Kabir S. H. (2016)

Domaine SI/GL

Conférence

ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW)

Filippova, A. et Cho, H. (2015)

Filippova, A. et Cho, H. (2016)

Journal

Journal of Database Management

Ven, K. et Verelst, J. (2008)

Xu, B., Lin Z. et Xu, Y. (2011)

Domaine GL

Conférence

IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)

Tourani, P., Adams, B., Serebrenik, A. (2017)

Annual CHI Conference on Human Factors in Computing Systems (CHI)

Vasilescu, B., Posnett, D., Ray, B. et van den Bret, M. G. J. (2015)

Conference on Computers and People Research

Teixeira, J. et Lin, T. (2014)

European Software Engineering Conference (ESEC) / ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE)

Zhou, M., Chen, Q., Mockus, A. et Wu, F. (2017)

IEEE Computer society

Shaikh, M. et Cornford, T. (2010)

IEEE International Conference on Computer Science and IT

Ahmed, F., Campbell, P., Jaffar, A. et Capretz, L. F. (2009)

IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)

Low, J. F., Yathog, T. et Svetinovic, D. (2015)

IEEE International Conference on Software Maintenance

Karus, S. (2013)

IEEE/ACM 37th IEEE International Conference on Software Engineering

Rastogi, A. (2015)

IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining

Allaho, M. Y. et Lee, W.-C. (2013)

International Business Information Management Conference

Ghapanchi, A. H. et Aurum, A. (2010)

International Conference on Computational Science and Its Application (ICCSA)

Anjos, E., Brasileiro, J., Silva, D. et M´ario Z.-R. (2016)

International Conference on Open Source Systems (OSS)

- Ahmed, I., Ghorashi, S. et Jensen, C. (2014)
Gamalielsson, J., Lundell, B. et Mattsson, A. (2011)
Iivari, N., Hedberg, H. et Kirves, T. (2008)

International Conference on Social Informatics (SocInfo)

- Jarczyk, O., Gruszka, B., Jaroszewicz, S., Bukowski, L. et Wierzbicki, A. (2014)

International Conference on Software Business (ICSOB)

- Lucassen, G., van Rooij, K., Jansen, S. (2013)

Journal

Electron Markets

- Ghapanchi, A. H. et Aurum, A. (2012)

Empirical Software Engineering

- Parnin, C., Bird, C. et Murphy-Hill, E. (2013)

Engineering with Computers

- Raza, A. et Capretz, L. F. (2012)

IEEE Software

- Dueñas, J. C., Parade G. H. A., Cuadrado, F., Santillán, M. et Ruiz, J. L. (2007)

IEEE Transactions on Engineering Management

- Ke, W. et Zhang, P. (2011)

IEEE Transactions on Software Engineering

- Raja U. et Tretter, M. J. (2012)

IET software

- Raza, A., Capretz, L. F. et Ahmed, F. (2012)

Information and Software Technology

- Jarczyk, O., Jaroszewicz, S., Wierzbicki, A., Pawlak, K. *et al.* (2018)
Marsan, J., Paré, G. et Wybo, M. D. (2012)
Samoladas, I., Angelis, L. et Stamelos, I. (2010)

Information Technology and Management

- Raja, U. et Tretter, M. J. (2009)

International Journal of Project Management

- Ghapanchi, A. H. et Aurum, A. (2012)

IT Professionnal

- Wang, H. J., Blue, J. et Plourde, M. (2010)

Journal of Global Information Management

- Wu, J., Goh, K.-Y., Li, H., Luo, C. et Zheng, H. (2016)

Journal of Organizational and End User Computing

- Hill, G., Datta, P. et Veter Weerdt, C. (2017)

Journal of Organizational Computing and Electronic Commerce

- Ghapanchi, A. H. (2015)

The Journal of Systems and Software

Gamalielsson, J. et Lundell, B. (2014)

Hanssen, G. K. (2012)

Lundell, B., Lings, B. et Syberfeldt, A. (2011)

Symposium

IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)

Ahmed, I., Forrest, D. et Jensen, C. (2017)

Atelier

IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)

Vasilescu, B., Filkov, V. et Serebrenik, A. (2015)

Domaine Autres/e-Commerce

Journal

Journal of Electronic Commerce Research

Guimaraes, A. L. S., Korn, H. J., Shin, N. et Eisner, A. B. (2015)

Domaine Autres/Management

Conférence

International Business Information Management Conference

Ghapanchi, A. H. et Aurum, A. (2010)

Domaine Autres/Maths-Statistique

Journal

Statistical science

Stewart, K. J., Darcy, D. P. et Daniel, S. L. (2006)

Domaine Autres/Multidisciplinaire

Journal

Decision Sciences

Peng, G., Mu, J. et Di Benedetto, C. A. (2013)

Journal of Internet Services and Applications

dos Santos, C. D. Jr. (2017)

Annexe 6: Les occurrences du mot « HEALTH » et ses dérivés dans les 98 articles

[<Files\\AIME\\Data_Extraction_Nov_2019\\101. HEALTH Developers-Quality-Control-and-Download-Volume-in-Open-Source-Software-\(OSS\)-Projects>](#) - § 1 reference coded [0,07% Coverage]

Reference 1 - 0,07% Coverage

As evidenced by our results, CVS Churn Control can serve as a successful litmus test for the **health** of an OSS project.

[<Files\\AIME\\Data_Extraction_Nov_2019\\121. \(HEALTH\) Exploring the impact of socio-technical core-periphery structures in open source software development>](#) - § 24 references coded [3,12% Coverage]

Reference 1 - 0,10% Coverage

It has been observed that success or failure of Open Source software depends largely on the **health** of their Open Source community

Reference 2 - 0,18% Coverage

Although there are a handful of papers discussing how one can assess if an Open Source project is a success (Lee *et al.*, 2009, Subramaniam *et al.*, 2009), there are relatively few recent papers discussing the **health** of an Open Source project.

Reference 3 - 0,10% Coverage

measures that reflect the **health** of the community's social structure but do not consider the socio-technical structure of the community.

Reference 4 - 0,12% Coverage

We propose that an analysis of the socio-technical structure of an Open Source project can provide a better understanding of the **health** of the project.

Reference 5 - 0,15% Coverage

We show that when such information is integrated into Open Source project portals such as Sourceforge, one can obtain considerable information on the socio-technical **health** of a particular project.

Reference 6 - 0,08% Coverage

We then see the relationship between the movement across this structure and the **health** of the project.

Reference 7 - 0,13% Coverage

This research adds to the literature on the socio-technical core-periphery shift pattern, while providing another way of assessing the **health** of an Open Source project.

Reference 8 - 0,10% Coverage

Using the average core-periphery shift metric we build on the notion of how one can determine the **health** of an Open Source project

Reference 9 - 0,12% Coverage

They focus more on how developers can successfully contribute to an Open Source project, rather than try and

determine the **health** of the Open Source project.

Reference 10 - 0,10% Coverage

So if a developer shifts from the core to the periphery it need not necessarily have an impact on the **health** of the software.

Reference 11 - 0,10% Coverage

Furthermore, we expect projects with large code (more LOC) to be associated with a larger community and as a result have a better **health**.

Reference 12 - 0,25% Coverage

We studied the average CPDM of different projects from Sourceforge.net (Sourceforge, Retrieved 1st March 2009) selected based on the following criteria: (i) size of the project, in terms of number of developers and LOC and (ii) based on the **health** of the project according to the status of the project on Sourceforge.net

Reference 13 - 0,14% Coverage

While JAIM has had very little activity, Eclipse Plugin Profiler is formally inactive and has poor **health**

Reference 14 - 0,14% Coverage

Although ivy-ssh and Megameknet are declared inactive and have poor **health** (Appendix B), JBoss is Production/Stable and as seen earlier is considered to be a successful Open Source project

Reference 15 - 0,09% Coverage

That is, a project that is inactive or whose **health** is waning need not have a core-periphery shift away from the core.

Reference 16 - 0,15% Coverage

Furthermore, an oscillatory shift towards and from the core need not indicate poor **health** of the project especially as the average CPDM never touches zero (as in the case of Megameknet and ivy-ssh).

Reference 17 - 0,20% Coverage

In this paper, we have discussed how we applied the core periphery concept from the field of social networks to identify problematic socio-technical core-periphery shifts in Open Source projects, which can provide another indicator for the **health** of the project.

Reference 18 - 0,14% Coverage

Crowston *et al.* (2006a) describe code quality, user ratings, number of users/downloads and code reuse among other indicators for the **health** and success of an Open Source project.

Reference 19 - 0,07% Coverage

The core-periphery shift pattern could give us another indicator of Open Source project **health**.

Reference 20 - 0,17% Coverage

Through the identification of core periphery shift patterns, we plan to provide the project leader (of JAIM for example) and potentially interested developers with one more indicator for the **health** of the Open Source

project.

Reference 21 - 0,14% Coverage

An oscillatory shift away and towards the core with a CPDM of zero in-between, as in the case of the Megameknet project, could also be considered as unstable for the **health** of the project.

Reference 22 - 0,14% Coverage

In this paper we claim that the trend of the average CPDM is only an indicator that the **health** of the project may be deteriorating and need not always imply that the project is **unhealthy**.

Reference 23 - 0,09% Coverage

We propose and demonstrate that this pattern can help in measuring and predicting the **health** of an Open Source project.

Reference 24 - 0,10% Coverage

Comparison along with a ranking of the different factors that affect a project's **health** can also be considered for future research.

[<Files\\AIME\\Data_Extraction_Nov_2019\\144. Gender and Tenure Diversity in GitHub Teams>](#) - § 1 reference coded [0,09% Coverage]

Reference 1 - 0,09% Coverage

In a **healthy** project, some rate of turnover is in fact desirable, as “new blood” brings in new abilities and ideas

[<Files\\AIME\\Data_Extraction_Nov_2019\\161. Impact of license choice on Open Source Software development activity>](#) - § 4 references coded [0,49% Coverage]

Reference 1 - 0,17% Coverage

While *Linux* and Apache are clearly success cases, it also was observed that over 80% of the OSS projects fail not because their products had no appeal to users but because their development process could not sustain a **healthy** level of activity and failed to attract much needed help

Reference 2 - 0,11% Coverage

In OSS development, there is a strong tendency to “release early and release often,” which implies that an active, frequent release cycle is considered a sign of a **healthy** development process

Reference 3 - 0,07% Coverage

Given this situation, we needed to devise a way to select only “**healthy**” OSS projects that have tractable activity data.

Reference 4 - 0,13% Coverage

To devise a way to select only “**healthy**” OSS projects, we focused on projects hosted in SF that met three criteria: They must be collaboratively (a) developed and (b) ported, and (c) had activity data publicly available.

[<Files\\AIME\\Data_Extraction_Nov_2019\\174. Learning and Open Source Software License Choice>](#) - § 1 reference coded [0,16% Coverage]

Reference 1 - 0,16% Coverage

While proprietary software is usually licensed to profit, OSS is licensed for the **healthy** development of the OSS community and usually is non-profit (von Hippel & von Krogh, 2003). Open Source Initiative (OSI), the governing body

[<Files\\AIME\\Data_Extraction_Nov_2019\\185. \(HEALTH\) Mudslinging and Manners Unpacking Conflict in Free and Open Source Software>](#) - § 2 references coded [0,26% Coverage]

Reference 1 - 0,18% Coverage

Given the long-term focus of the projects, developers recognize that it is better for the overall **health** of the team and their relationships with authority figures to accommodate the individuals in charge.

Reference 2 - 0,08% Coverage

Normative conflict therefore has potential to be generative to community **health** and function.

[<Files\\AIME\\Data_Extraction_Nov_2019\\186. Negotiating open source software adoption in the UK public sector>](#) - § 1 reference coded [0,10% Coverage]

Reference 1 - 0,10% Coverage

To build a sustainable ecosystem and long-term growth for open source products in the public sector it is essential that a **healthy** ecosystem of small and medium sized firms are stimulated to service public sector open source products.

[<Files\\AIME\\Data_Extraction_Nov_2019\\194. \(HEALTH\) On the Scalability of Linux Kernel Maintainers' Work>](#) - § 1 reference coded [0,10% Coverage]

Reference 1 - 0,10% Coverage

The most obvious and compelling reason is commercial interest of large (and rich) companies in the continued robust **health** of *Linux*

[<Files\\AIME\\Data_Extraction_Nov_2019\\207. \(EcoHealth\) Open Source Software for Model Driven Development. A Case Study>](#) - § 11 references coded [2,42% Coverage]

Reference 1 - 0,26% Coverage

Before an organisation adopts an Open Source project it is important to evaluate its ecosystem in order to make sure that it is **healthy** and that the project is likely to be sustainable and maintained for a long time

Reference 2 - 0,14% Coverage

One important means in such an evaluation is to quantitatively assess the **health** of an Open Source community

Reference 3 - 0,23% Coverage

Specifically, the aim is to reveal insights concerning the **health** of the ecosystems of the two OSS projects as this constitutes an important basis for strategic decision making within the company.

Reference 4 - 0,26% Coverage

Specifically, three aspects of ecosystem **health** were considered important: activity in ecosystems; company influence in ecosystems; and interaction between ecosystems for the two OSS projects (Topcased and Papyrus).

Reference 5 - 0,12% Coverage

The company interest in these three aspects of ecosystem **health** is in line with previous research

Reference 6 - 0,32% Coverage

The activity in ecosystems is an important factor that reflects the **health** and long-term sustainability of OSS projects, and we therefore studied each project separately in terms of the extent of contributions to SCM repositories, mailing lists, and forums over time.

Reference 7 - 0,15% Coverage

In summary, our analysis suggests that both projects have **healthy** ecosystems, with an active base of developers and users.

Reference 8 - 0,18% Coverage

Our specific assessment of the **health** for two Open Source communities may have broader implications for evaluation and assessment of software systems.

Reference 9 - 0,28% Coverage

Our specific strategy used for assessing **health** of Open Source ecosystems has certain similarities with, and may contribute to, previously proposed approaches (e.g. OpenBRR, QSOS, OMM) for evaluation and assessment of Open Source projects

Reference 10 - 0,19% Coverage

From this we note that both Open Source projects have promising **health** in their respective ecosystems

Reference 11 - 0,28% Coverage

Since the company perceives the **health** of the Topcased ecosystem to be promising and based on earlier positive experience of the tool, such a pilot study involving Topcased usage within the company context is seen as a natural next step.

[<Files\\AIME\\Data_Extraction_Nov_2019\\211. \(HEALTH\) Opportunities and Challenges Applying Functional Data Analysis to the Study of Open Source Software Evolution>](#) - § 1 reference coded [0,19% Coverage]

Reference 1 - 0,19% Coverage

Given that popularity has been considered a facet of OSS project success (Stewart, Ammeter and Maruping, 2006), such a trend would be undesirable for the longer-term **health** of an OSS project.

[<Files\\AIME\\Data_Extraction_Nov_2019\\217. \(HEALTH\) Perceptions of Diversity on Git Hub A User Survey>](#) - § 1 reference coded [0,21% Coverage]

Reference 1 - 0,21% Coverage

c) Negative Effects: Related to programming experience, examples include: improper contributions by newcomers (“sometimes newer team members have trouble with code **health** and best practices”

[<Files\\AIME\\Data_Extraction_Nov_2019\\219. \(HEALTH\) Practitioner perceptions of Open Source software in the embedded systems area>](#) - § 7 references coded [0,78% Coverage]

Reference 1 - 0,16% Coverage

that the company that initially developed the software is trying to hand over too much work to the community, “few people will bother contributing” (Engelfriet, 2010). However, in a **healthy** open source community this is not an issue.

Reference 2 - 0,15% Coverage

Hence, a **healthy** evolution of the open business ecosystem around that platform is important as it creates new opportunities for collaboration between Nokia and a range of other organisations and individual stakeholders

Reference 3 - 0,05% Coverage

Both stress the importance of the **health** of the OS community around a project

Reference 4 - 0,11% Coverage

To assess the **health** of a community, the number of users of the software and the activity in the development project are the indicators primarily considered

Reference 5 - 0,07% Coverage

Long-term maintenance, including reduced cost: community maintenance can be expected in a **healthy** project.

Reference 6 - 0,15% Coverage

This corresponds with the first point in Abedour’s (2007) “important guidelines” for highquality Open Source software and with the findings of Crowston and Howison (2006) on the **health** of Open Source communities.

Reference 7 - 0,09% Coverage

Our results also reveal the importance attached to the **health** and activity of communities around an Open Source software project.

[<Files\\AIME\\Data_Extraction_Nov_2019\\220. Predicting software future sustainability A longitudinal perspective>](#) - § 1 reference coded [0,12% Coverage]

Reference 1 - 0,12% Coverage

High level of development sustainability is an indication to community users that a given project is alive, **healthy** and potentially responsive.

[<Files\\AIME\\Data_Extraction_Nov_2019\\235. \(HEALTH\) Software Analytics Study of Open-Source System Survivability Through Social Contagion>](#) - § 3 references coded [0,99% Coverage]

Reference 1 - 0,19% Coverage

In order to retrieve the dominant features that influence the **health** of a FLOSS project, we employed machine learning.

Reference 2 - 0,23% Coverage

Software development methods that allow for more flexibility, e.g., agile methods [21], are far more conducive to the **health** of a FLOSS project.

Reference 3 - 0,56% Coverage

The design of our study, namely the manner in which we used the most recent commit activity timestamp as an indicator of the **health** of a project, does not allow us to observe the pattern of commits, such as when commits dropped to zero and when the number of commits picked up again, that is necessary to study the phenomenon of project revival.

[<Files\\AIME\\Data_Extraction_Nov_2019\\245. \(HEALTH\) Surgical teams on GitHub Modeling performance of GitHub project development processes>](#) - § 1 reference coded [0,05% Coverage]

Reference 1 - 0,05% Coverage

Bug fixing times have also been proposed as a factor in the measurement of **health** of OSS communities

[<Files\\AIME\\Data_Extraction_Nov_2019\\246. Survival analysis on the duration of open source projects>](#) - § 1 reference coded [0,13% Coverage]

Reference 1 - 0,13% Coverage

Healthy projects often have hundreds of commits per month even per week, so it is fair to assume that a project with one commit per month has ended.

[<Files\\AIME\\Data_Extraction_Nov_2019\\250. Sustainability of Open Source software communities beyond a fork. How and why has the LibreOffice project evolved>](#) - § 2 references coded [0,16% Coverage]

Reference 1 - 0,05% Coverage

Earlier research also suggests that an effective structure of governance is a basis for **healthy** and sustainable OSS communities (

Reference 2 - 0,11% Coverage

Further, a number of comments also revealed that the evolution for the LO project seemed to have exceeded the expectations of the respondents, as illustrated by the following comments: “while this is a young project, I am surprised by how diverse it is and how **healthy** it is”,

[<Files\\AIME\\Data_Extraction_Nov_2019\\253. The application profiles and development characteristics of library Open Source Software projects>](#) - § 1 reference coded [0,12% Coverage]

Reference 1 - 0,12% Coverage

A good count of projects exist in the other development modes, suggesting **healthy** communities that attract and sustain projects.

[<Files\\AIME\\Data_Extraction_Nov_2019\\275. THE LIFE CYCLE OF OPEN SOURCE SOFTWARE DEVELOPMENT COMMUNITIES>](#) - § 1 reference coded [0,30% Coverage]

Reference 1 - 0,30% Coverage

But coordinators should also be responsible for identifying important issues in their domain, planning and facilitating events and communication; linking members and other communities; managing community boundaries; fostering membership development; stimulating the setup and maintenance of a knowledge base; documenting best practices, lessons learned, tools, and methods; and assessing the **health** of the community.

[<Files\\AIME\\Data_Extraction_Nov_2019\\297. Understanding Sustained Participation in Open Source Software Projects>](#) - § 2 references coded [0,16% Coverage]

Reference 1 - 0,05% Coverage

According to IPP theory, situated learning and identity construction are essential elements of **healthy** OSS communities

Reference 2 - 0,11% Coverage

The project easily met the second criterion; phpMyadmin was a **healthy** and busy project with a large volume of mailing list messages and cVS commits, and was being steadily contributed to over time with no major interruptions and structural changes in progress.

[<Files\\AIME\\Data_Extraction_Nov_2019\\304. Using Classification Methods to Reinforce the Impact of Social Factors on Software Success>](#) - § 1 reference coded [0,19% Coverage]

Reference 1 - 0,19% Coverage

We also found other studies where authors applied further concepts of machine learning to predict how **healthy** a project were in a given moment of time.

[<Files\\AIME\\Data_Extraction_Nov_2019\\310. \(HEALTH\) What makes a good contributor Understanding contributor behavior within large Free Open Source Software projects–A socialization perspective>](#) - § 2 references coded [0,09% Coverage]

Reference 1 - 0,05% Coverage

This research implies that more attention shall be paid by firms to allocate employees that engage in behaviors that aim at maintaining communities into a sound and **healthy** state

Reference 2 - 0,04% Coverage

It emphasizes the need to adopt a community centric approach that places the **health** and well-being of communities at the core of the debate.

[<Files\\AIME\\Data_Extraction_Nov_2019\\4. A Case Study of Motivations for Corporate Contribution to FOSS>](#) - § 1 reference coded [0,16% Coverage]

Reference 1 - 0,16% Coverage

Understanding how corporate sponsorship affects the social dynamics and evolution of Free/Open Source code and community is critical to fostering **healthy** communities.

[<Files\\AIME\\Data_Extraction_Nov_2019\\47. An Exploration of Code Quality in FOSS Projects>](#) - § 1 reference coded [0,71% Coverage]

Reference 1 - 0,71% Coverage

In our analysis we have used the total number of coding standard violations and code smells as the indicator of FOSS projects quality. Identified regression models indicate that the number of core developers is correlated with quality of the FOSS project. This is in line with the findings of Sen *et al.*, who found that number of developers reflect the “**healthiness**” of a FOSS project

[<Files\\AIME\\Data_Extraction_Nov_2019\\61. \(HEALTH\) Automatic Means of Identifying Evolutionary Events in Software Development>](#) - § 1 reference coded [0,33% Coverage]

Reference 1 - 0,33% Coverage

Choosing the open source project to implement in business scenario – evolving source code is deemed to signify good **health** of a software project.

[<Files\\AIME\\Data Extraction Nov 2019\\OK.109. Dual Allegiance and Knowledge Sharing in Open Source Software Firms>](#) - § 1 reference coded [0,17% Coverage]

Reference 1 - 0,17% Coverage

Because gatekeepers typically establish and nurture a good, **healthy** equilibrium between allegiance towards the community and the firm, they are highly desirable in the context of open source software firms.

[<Files\\AIME\\Data Extraction Nov 2019\\OK.111. Ecosystem Health of Cloud PaaS Providers>](#) - § 17 references coded [3,87% Coverage]

Reference 1 - 0,26% Coverage

Despite this explosive growth in both academic and business relevance, no publicly accessible information is available on the ecosystems **health** of public cloud PaaS providers

Reference 2 - 0,20% Coverage

Ecosystem health is defined as "long-term financial well-being of the business ecosystem and the long-term strength of the network" [4]

Reference 3 - 0,24% Coverage

Access to knowledge regarding ecosystem **health** is crucial for businesses researching the possibility to move their software to the cloud for two reasons [5].

Reference 4 - 0,25% Coverage

Because PaaS providers are primarily software businesses, this paper leverages knowledge on the academic field of software ecosystems to study their ecosystem **health**.

Reference 5 - 0,24% Coverage

In the context of PaaS ecosystem **health**, the contributors to the long-term strength of the ecosystem are the direct users of the PaaS technology: developers.

Reference 6 - 0,20% Coverage

This paper uses data of the open source code hosting service GitHub to analyze the ecosystem **health** of eight different PaaS providers.

Reference 7 - 0,22% Coverage

Next, specific data fields are aggregated into ecosystem **health** results based on open source ecosystem **health** measures introduced by Crowston *et al.*

Reference 8 - 0,21% Coverage

Finally, statistical analysis extracts relevant insights on the current growth expectations of a PaaS and expected **health** of their ecosystem.

Reference 9 - 0,13% Coverage

Relevant comparative measures are necessary to evaluate the **health** of a PaaS ecosystem.

Reference 10 - 0,18% Coverage

In 2006, a multitude of indicators were collected by Crowston *et al.* to measure the **health** of open source ecosystems

Reference 11 - 0,37% Coverage

We consider four indicators from this research based on three earlier publications [13,14,15] to be appropriate in the context of PaaS ecosystem **health**:

1. Number of active developers 2. Spin offs 3. Interest in the project 4. Download count

Reference 12 - 0,20% Coverage

On the other hand, if developers are not interested in developing for that platform, this indicates that the ecosystem is **unhealthy**.

Reference 13 - 0,31% Coverage

This paper is based on the assumption that developer interest has a direct correlation to the customer size of the PaaS provider and subsequently the long term **health** of the ecosystem that belongs to it.

Reference 14 - 0,18% Coverage

As a result, it is impossible to assess PaaS ecosystem **health** with the download count indicator of Crowston et al [11].

Reference 15 - 0,25% Coverage

Although e.g. Engine Yard scored low, they reported revenues of 28\$ million in 2011. This shows an obvious market interest and a potential for **healthy** future growth

Reference 16 - 0,31% Coverage

As a consequence, we are unable to evaluate the long-term financial well-being of the business ecosystem and are restricted to the long-term strength aspect of the ecosystem **health** definition on page 184.

Reference 17 - 0,13% Coverage

The current ecosystem **health** is skewed to two major players, with Heroku far ahead.

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.2. 'Letting go of Control' to Embrace Open Source. Implications for Company and Community>](#) - § 2 references coded [0,60% Coverage]

Reference 1 - 0,20% Coverage

One feature identified by our respondents, which has not been focused on in the literature, was the concept of a '**healthy** community' established around an open source product.

Reference 2 - 0,40% Coverage

Another manager was happy to define the characteristics considered when assessing a **healthy** community; “It’s how many people are in the community, how many people contribute, how often do they release, how many bugs do they have on a given release. How many days does it typically take for a bug to be resolved [by] the community, how has it grown or shrunk overtime”

[<Files\AIME\Data_Extraction_Nov_2019\OK.54_Apache_and_Eclipse_Comparing_Open_Source_Project_Incubators>](#) - § 4 references coded [1,00% Coverage]

Reference 1 - 0,20% Coverage

To nurture **healthy** communities, deliver stable releases, and manage initial risks, organizations follow various incubation processes and approaches.

Reference 2 - 0,29% Coverage

Both incubators require candidates to have a focused set of objectives that must be formally stated in a proposal. This helps create **healthy** communities that aim at solving specific problems, promoting effectiveness

Reference 3 - 0,18% Coverage

Both emphasize the importance of the community for a project’s **health** by using the number of committers as a criterion for approval.

Reference 4 - 0,33% Coverage

In this sense, although the EMO acknowledges that the project community’s size is a key aspect for community **health** and the project’s technical impact, the minimum number of committers to pass the creation review isn’t clearly established.

Annexe 7 : Les occurrences du mot « ECOSYSTEM.S » dans les 98 articles

[<Files\\AIME\\Data_Extraction_Nov_2019\\144. Gender and Tenure Diversity in GitHub Teams>](#) - § 1 reference coded [0,06% Coverage]

Reference 1 - 0,06% Coverage

We chose GITHUB for our study because it: (i) is the largest **ecosystem** of its kind;

[<Files\\AIME\\Data_Extraction_Nov_2019\\167. Investigating the Interrelationships among Success Measures of Open Source Software Projects>](#) - § 1 reference coded [0,13% Coverage]

Reference 1 - 0,13% Coverage

ChengalurSmith and colleagues (2010) defined sustainability as “the ability of an organism or an **ecosystem** to maintain its activity and productivity over time” (p. 660)

[<Files\\AIME\\Data_Extraction_Nov_2019\\186. Negotiating open source software adoption in the UK public sector>](#) - § 6 references coded [0,48% Coverage]

s

Reference 1 - 0,11% Coverage

Camden Council faced numerous challenges with its decision to adopt and co-create open source software in the form of community management, limited funding, a dying **ecosystem** of SME vendor support, and lack of uptake of the CMS by as many councils as expected

Reference 2 - 0,07% Coverage

Our data shows that the becoming of adoption can be both constrained and precipitated by various forms of materiality (of the assemblage of the open source **ecosystem**)

Reference 3 - 0,04% Coverage

However, a shrinking **ecosystem** of vendor support over time made a change in code less possible.

Reference 4 - 0,07% Coverage

Encourage the growth of vendors and suppliers that are in a position to service open source products to create a necessary **ecosystem** of support for products.

Reference 5 - 0,10% Coverage

To build a sustainable **ecosystem** and long-term growth for open source products in the public sector it is essential that a healthy **ecosystem** of small and medium sized firms are stimulated to service public sector open source products.

Reference 6 - 0,08% Coverage

The theoretical idea is that the becoming of adoption can be both constrained and precipitated by various forms of software materiality (of the assemblage of the open source **ecosystem**)

[<Files\\AIME\\Data_Extraction_Nov_2019\\194. \(HEALTH\) On the Scalability of Linux Kernel Maintainers' Work>](#) - § 17 references coded [2,20% Coverage]

Reference 1 - 0,12% Coverage

We expect that our proposed framework to quantify maintainer practices will help clarify the factors that allow rapidly growing **ecosystems** to be sustainable.

Reference 2 - 0,14% Coverage

Free/libre open source (FLOSS) **ecosystems**, in particular large **ecosystems** such as the *Linux* kernel, OpenStack, Docker, or *Android*, represent critical computing infrastructure for our society.

Reference 3 - 0,11% Coverage

Such **ecosystems** involve contributions from various participants who are distributed over the world, and are diverse in skills, interests, and needs.

Reference 4 - 0,07% Coverage

Understanding how such core groups function can shed light on *Linux* and other large **ecosystems**.

Reference 5 - 0,19% Coverage

We, therefore, assume that the coping mechanisms that make **ecosystem** effective allow for adaptations that both are caused by and affect software architecture (represented by module structure), change over time, and react to commercial involvement strategies.

Reference 6 - 0,14% Coverage

These differences, which are explored [...], may reflect how the **ecosystem** adapts to the variations in architecture, commercial involvement, and growing workloads.

Reference 7 - 0,15% Coverage

To understand the scalability of *Linux* **ecosystem**, we investigate how maintainer productivity scales when more maintainers are added to a set of maintained files

Reference 8 - 0,15% Coverage

We explore how the *Linux* **ecosystem**, particularly the central parts (represented by the seven modules defined in Section 2.2.2), grows over time in terms of commits, authors, files, maintainers, and new joiners.

Reference 9 - 0,10% Coverage

An average maintainer in the *Linux* kernel does not appear to have an increasing workload despite rapid expansion of the **ecosystem**.

Reference 10 - 0,11% Coverage

If that relationship does apply, it may suggest that the *Linux* kernel **ecosystem** has self-similarity [28], i.e., the system has invariants that are preserved

Reference 11 - 0,15% Coverage

Such self-similarity, as in biological **ecosystems**, would suggest that similar mechanisms must be at play in the context of *Linux* kernel maintainers as at the larger scale of core/peripheral contributors.

Reference 12 - 0,15% Coverage

The **ecosystem** uses community support for central tasks and central code, but the modules of the architectural

periphery are most effectively supported by commercial entities (except for the kernel module).

Reference 13 - 0,11% Coverage

The **ecosystem** grows primarily because commercial entities add code to the drivers module; the community resolves conflicts in core modules such as mm.

Reference 14 - 0,18% Coverage

They characterized a community as a series of concentric circles; each circle is occupied by people playing a particular role in the development process. In this study we focus on the central circle of the *Linux* kernel **ecosystem**: maintainers.

Reference 15 - 0,07% Coverage

Evolutionary studies of long-lived, large-scale FLOSS **ecosystems** have attracted some attention

Reference 16 - 0,18% Coverage

We investigated maintainer activities in the *Linux* **ecosystem** and quantified maintainers' work based on the files they maintain, the change activity in the maintained files, and the number and churn of external contributors they must deal with.

Reference 17 - 0,10% Coverage

The mechanisms underlying such large-scale complicated production remain to be explored because the **ecosystem** never stops evolving.

[<Files\\AIME\\Data_Extraction_Nov_2019\\207. \(EcoHealth\) Open Source Software for Model Driven Development A Case Study>](#) - § 19 references coded [4,26% Coverage]

Reference 1 - 0,26% Coverage

Before an organisation adopts an Open Source project it is important to evaluate its **ecosystem** in order to make sure that it is healthy and that the project is likely to be sustainable and maintained for a long time

Reference 2 - 0,23% Coverage

Specifically, the aim is to reveal insights concerning the health of the **ecosystems** of the two OSS projects as this constitutes an important basis for strategic decision making within the company.

Reference 3 - 0,24% Coverage

To guide the decision on whether to engage in any of the **ecosystems** for the two projects, Combitech wanted to establish that the projects are likely to be sustainable and maintained for a long time

Reference 4 - 0,26% Coverage

Specifically, three aspects of **ecosystem** health were considered important: activity in **ecosystems**; company influence in **ecosystems**; and interaction between **ecosystems** for the two OSS projects (Topcased and Papyrus).

Reference 5 - 0,12% Coverage

The company interest in these three aspects of **ecosystem** health is in line with previous research

Reference 6 - 0,18% Coverage

To investigate the activity in **ecosystems**, we analysed the contributions in terms of committed SCM artefacts of the Open Source projects over time

Reference 7 - 0,19% Coverage

To investigate the company influence in **ecosystems**, we analysed over time the extent to which different contributors are affiliated with different companies.

Reference 8 - 0,20% Coverage

The interaction between **ecosystems** was assessed by studying contributors that are active in both projects.

Reference 9 - 0,32% Coverage

The activity in **ecosystems** is an important factor that reflects the health and long term sustainability of OSS projects, and we therefore studied each project separately in terms of the extent of contributions to SCM repositories, mailing lists, and forums over time.

Reference 10 - 0,22% Coverage

The results indicate that there is considerable interaction between **ecosystems**, as a large proportion of the commits each month is provided by developers active in both projects.

Reference 11 - 0,13% Coverage

Organisations that want to use/adopt OSS-based MDD tools need to engage in the **ecosystems** of the projects

Reference 12 - 0,17% Coverage

In order to investigate the collaborative climate in each **ecosystem**, it would necessitate genuine interaction with each project in a pilot study

Reference 13 - 0,27% Coverage

Without such it would be difficult to assess whether the collaborative climate in the Topcased and Papyrus **ecosystems** is congruent with existing work practices in the company

Reference 14 - 0,25% Coverage

In summary, our analysis suggests that both projects have healthy **ecosystems**, with an active base of developers and users. Despite a seemingly commercial drive, both projects seem to have developers representing

Reference 15 - 0,38% Coverage

Our specific strategy used for assessing health of Open Source **ecosystems** has certain similarities with, and may contribute to, previously proposed approaches (e.g. OpenBRR, QSOS, OMM) for evaluation and assessment of Open Source projects

Reference 16 - 0,17% Coverage

The focus for this case study has been to explore the sustainability of the **ecosystems** for the two Open Source projects Topcased and Papyrus.

Reference 17 - 0,19% Coverage

From this we note that both Open Source projects have promising health in their respective **ecosystems**.

Reference 18 - 0,28% Coverage

Since the company perceives the health of the Topcased **ecosystem** to be promising and based on earlier positive experience of the tool, such a pilot study involving Topcased usage within the company context is seen as a natural next step.

Reference 19 - 0,21% Coverage

From this, the company will be able to further investigate the collaborative climate through genuine interaction between the company context and the broader Topcased **ecosystem**.

[<Files\\AIME\\Data_Extraction_Nov_2019\\219. \(HEALTH\) Practitioner perceptions of Open Source software in the embedded systems area>](#) - § 6 references coded [0,81% Coverage]

Reference 1 - 0,18% Coverage

For electronic archiving the OSS licence GPL has been recommended for software that handles data that needs to be kept for (very) long life-cycles as it creates an “**ecosystem**” around the software that is independent of the organisation that generated it.

Reference 2 - 0,11% Coverage

From these, we have seen novel infrastructure result, as well as knowledge on sociotechnical issues related to the broader business **ecosystem** around OSS projects.

Reference 3 - 0,10% Coverage

The idea is to establish a business **ecosystem** involving other organisations and stakeholders that may have different interests in the software.

Reference 4 - 0,08% Coverage

Different companies in a business **ecosystem** may have different interests in the success of an OSS-project.

Reference 5 - 0,19% Coverage

Obviously, depending on the role a company has in the value-chain of an existing business relationship, a move by one company to make a system open source can fundamentally change (both positively and negatively) the conditions for making business in such a business **ecosystem**.

Reference 6 - 0,15% Coverage

Hence, a healthy evolution of the open business **ecosystem** around that platform is important as it creates new opportunities for collaboration between Nokia and a range of other organisations and individual stakeholders.

[<Files\\AIME\\Data_Extraction_Nov_2019\\220. Predicting software future sustainability A longitudinal perspective>](#) - § 1 reference coded [0,12% Coverage]

Reference 1 - 0,12% Coverage

Chengalur-Smith *et al.* [4] define sustainability as “... the ability of an organism or an **ecosystem** to maintain its activity and productivity over

[<Files\\AIME\\Data_Extraction_Nov_2019\\235. \(HEALTH\) Software Analytics Study of Open-Source System Survivability Through Social Contagion>](#) - § 1 reference coded [0,27% Coverage]

Reference 1 - 0,27% Coverage

Not only is GitHub data's size a testimony to FLOSS development's popularity, it also means that a rich **ecosystem** of software projects exists for us to examine.

[<Files\\AIME\\Data_Extraction_Nov_2019\\249. Sustainability of Free Libre Open Source Projects A Longitudinal Study>](#) - § 2 references coded [0,13% Coverage]

Reference 1 - 0,06% Coverage

Sustainability is generally defined as the ability of an organism or an **ecosystem** to maintain its activity and productivity over time

Reference 2 - 0,07% Coverage

Definitions of organizational sustainability vary from narrowly organizational to those that include social, economic and natural **ecosystems**

[<Files\\AIME\\Data_Extraction_Nov_2019\\250. Sustainability of Open Source software communities beyond a fork How and why has the LibreOffice project evolved>](#) - § 2 references coded [0,15% Coverage]

Reference 1 - 0,07% Coverage

Another argument is that code forking can positively impact on both governance and sustainability of OSS projects at the levels of the software, its community and business **ecosystem**

Reference 2 - 0,07% Coverage

Further, the importance of a vibrant community was also stressed by one respondent as follows: "the more rich and diverse and compelling we make our **ecosystem**, the stronger it is".

[<Files\\AIME\\Data_Extraction_Nov_2019\\254. The attraction of contributors in free and open source software projects>](#) - § 5 references coded [0,44% Coverage]

Reference 1 - 0,12% Coverage

This paper seeks to examine the set of project conditions and the underlying dynamics of what makes an open source project attractive, aiming to inform practice and generate a theory designed to the open source **ecosystem**, a kind of theory that, being specific, has been overlooked

Reference 2 - 0,11% Coverage

The success of open source software has been attributed to many developers working under the Bazaar paradigm. However, only a few FOSP managed to build virtuous and productive Bazaar **ecosystems** (Krishnamurthy, 2002), giving rise to a scale-free network

Reference 3 - 0,07% Coverage

Free software projects, as creative enterprises, have agents embedded in an open **ecosystem** that can inspire and evaluate their contributions and resulting products

Reference 4 - 0,05% Coverage

The richer this **ecosystem**, or larger the social milieu, the better for the project, as it becomes more diverse, fostering innovation

Reference 5 - 0,10% Coverage

This process that FOSS follow to accumulate resources, together with the limited availability of resources in the market and the high competition for them, generates a disparate **ecosystem** with the scale-free characteristic observed

[<Files\\AIME\\Data_Extraction_Nov_2019\\270. The Impact of Intellectual Property Rights Enforcement on Open Source Software Project Success>](#) - § 2 references coded [0,29% Coverage]

Reference 1 - 0,14% Coverage

Lawsuit attracted significant publicity, partly because it was the first major IPR enforcement suit concerning OSS usage and development and partly because it related to the *Linux* platform and so had important implications for the entire *Linux ecosystem*

Reference 2 - 0,14% Coverage

This is surprising, given that the large number of recent IPR enforcement actions targeting entities that use or develop OSS (e.g., surrounding mobile phone technology and the *Android* platform) suggests great salience of these issues to the OSS **ecosystem**

[<Files\\AIME\\Data_Extraction_Nov_2019\\286. THE TRANSFORMATION OF OPEN SOURCE SOFTWARE>](#) - § 3 references coded [0,46% Coverage]

Reference 1 - 0,17% Coverage

Small software companies can become part of an **ecosystem** offering consultancy, service, and support of open source products

Reference 2 - 0,15% Coverage

Indeed, the network benefits of open source arise as a result of the size of the overall community and **ecosystem**.

Reference 3 - 0,15% Coverage

Thus, a network of interested parties with complementary capabilities can form an **ecosystem** to offer a professional product and service in an agile, bazaar-friendly manner.

[<Files\\AIME\\Data_Extraction_Nov_2019\\4. A Case Study of Motivations for Corporate Contribution to FOSS>](#) - § 2 references coded [0,21% Coverage]

Reference 1 - 0,09% Coverage

Free/Open Source Software (FOSS) is a key component of our current computing **ecosystem**.

Reference 2 - 0,12% Coverage

The success and growth of the FOSS movement important to the is overall well-being of the therefore computing **ecosystem**.

[<Files\\AIME\\Data_Extraction_Nov_2019\\70. Changes in free and open source software licenses managerial interventions and variations on project attractiveness>](#) - § 1 reference coded [0,22% Coverage]

Reference 1 - 0,22% Coverage

Nowadays, there are hundreds of thousands of free software projects online, each representing a computer supported cooperative work opportunity for generating an active and growing **ecosystem** of users and contributors capable of joint development at an unprecedented scale

[<Files\\AIME\\Data_Extraction_Nov_2019\\75. Code of Conduct in Open Source Projects>](#) - § 4 references coded [0,40% Coverage]

Reference 1 - 0,15% Coverage

In addition, we noticed that many Google results correspond to **ecosystems** instead of to individual projects, while the GitHub results mostly correspond to individual projects.

Reference 2 - 0,11% Coverage

As such, the Google **ecosystem** results actually imply that a larger number of existing projects are using the same code of conduct

Reference 3 - 0,06% Coverage

Finally, Ubuntu and Python are highly popular open source **ecosystems**.

Reference 4 - 0,07% Coverage

Such leadership-related clauses seem specific to **ecosystem**-related codes of conduct.

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.111. Ecosystem Health of Cloud PaaS Providers>](#) - § 24 references coded [5,61% Coverage]

Reference 1 - 0,26% Coverage

Despite this explosive growth in both academic and business relevance, no publicly accessible information is available on the **ecosystems** health of public cloud PaaS providers

Reference 2 - 0,20% Coverage

Ecosystem health is defined as "long-term financial well-being of the business **ecosystem** and the long-term strength of the network"

Reference 3 - 0,23% Coverage

Access to knowledge regarding **ecosystem** health is crucial for businesses researching the possibility to move their software to the cloud for two reasons

Reference 4 - 0,12% Coverage

To begin, businesses attempt to seek partners with a robust business **ecosystem**

Reference 5 - 0,25% Coverage

Because PaaS providers are primarily software businesses, this paper leverages knowledge on the academic

field of software **ecosystems** to study their **ecosystem** health

Reference 6 - 0,78% Coverage

Several authors have defined the term software **ecosystem** [7,8], but this research applies the following definition by Jansen *et al.*, as it builds upon earlier definitions and further abstracts the concept [9]: "a set of factors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts."

Reference 7 - 0,09% Coverage

A high profile example of a strong software **ecosystem** is *iOS*.

Reference 8 - 0,24% Coverage

Apple benefits from the popular App Store, due to which the *iOS* **ecosystem** remains the market leader in terms of app availability, app sales and profit margins [10]

Reference 9 - 0,24% Coverage

In the context of PaaS **ecosystem** health, the contributors to the long-term strength of the **ecosystem** are the direct users of the PaaS technology: developers.

Reference 10 - 0,20% Coverage

This paper uses data of the open source code hosting service GitHub to analyze the **ecosystem** health of eight different PaaS providers

Reference 11 - 0,19% Coverage

First, metadata of all software projects or repositories that contribute to the **ecosystems** of each PaaS providers is collected.

Reference 12 - 0,22% Coverage

Next, specific data fields are aggregated into **ecosystem** health results based on open source **ecosystem** health measures introduced by Crowston *et al.*

Reference 13 - 0,21% Coverage

Finally, statistical analysis extracts relevant insights on the current growth expectations of a PaaS and expected health of their **ecosystem**.

Reference 14 - 0,16% Coverage

Moreover, the PaaS providers are presented with a method to gain insight into the state of their **ecosystem**

Reference 15 - 0,13% Coverage

Relevant comparative measures are necessary to evaluate the health of a PaaS **ecosystem**.

Reference 16 - 0,18% Coverage

In 2006, a multitude of indicators were collected by Crowston *et al.* to measure the health of open source **ecosystems**

Reference 17 - 0,37% Coverage

We consider four indicators from this research based on three earlier publications [13,14,15] to be appropriate in the context of PaaS **ecosystem** health: 1. Number of active developers 2. Spin offs 3. Interest in the project 4. Download count

Reference 18 - 0,11% Coverage

If developers create applications for a PaaS, its **ecosystem** will flourish

Reference 19 - 0,51% Coverage

On the other hand, if developers are not interested in developing for that platform, this indicates that the **ecosystem** is unhealthy. This paper is based on the assumption that developer interest has a direct correlation to the customer size of the PaaS provider and subsequently the long term health of the **ecosystem** that belongs to it

Reference 20 - 0,17% Coverage

As a result, it is impossible to assess PaaS **ecosystem** health with the download count indicator of Crowston et al

Reference 21 - 0,32% Coverage

Based on the low percentage of false positives found in the sample, we are confident that the collected dataset provides an accurate representation of the contemporary **ecosystems** of the selected PaaS providers.

Reference 22 - 0,11% Coverage

However, absolute size is not an indicator of a great **ecosystem** per se.

Reference 23 - 0,12% Coverage

The current **ecosystem** health is skewed to two major players, with Heroku far ahead

Reference 24 - 0,19% Coverage

Additionally, PaaS providers can evaluate the current state of their **ecosystem** and adjust their corporate strategy accordingly.

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.14. A longitudinal case study of an emerging software ecosystem. Implications for practice and theory>](#) - § 35 references coded [7,73% Coverage]

Reference 1 - 0,06% Coverage

A recent development within software engineering is the emergence of software **ecosystems**

Reference 2 - 0,15% Coverage

This new concept and its implicit reference to ecology imply a shift of focus from the internals of the software organization (the individual organism) towards its environment and the relations and actions within (the **ecosystem**).

Reference 3 - 0,20% Coverage

Viewing the software industry and the market it serves as an **ecosystem**, may introduce a set of new challenges and opportunities (Jansen *et al.*, 2009a,b), for example new business models, open innovation, collaborative development, issues of ownership, strategic planning, and variability management.

Reference 4 - 0,10% Coverage

Moved from a closed and plan-driven approach to a practice of agile software product line engineering (SPLE) and now further towards a software **ecosystem**.

Reference 5 - 0,13% Coverage

Inspired by our detailed insight into a real industrial case and with the ambition to try to understand some of the details of how a concrete **ecosystem** develops, we have define research question 1

Reference 6 - 0,13% Coverage

We have found that this knowledge is relevant, applicable and beneficial to the software engineering domain, and that existing literature on software **ecosystems** lacks this theoretical connection.

Reference 7 - 0,08% Coverage

We believe that this field of research can benefit from the development of an empirically grounded theory of software **ecosystems**

Reference 8 - 0,11% Coverage

Section 2 provides relevant background information about the concepts being discussed, such as socio-technical theory, organizational ecology and software **ecosystems**.

Reference 9 - 0,16% Coverage

Viewing the organizational environment as an **ecosystem** means that it is considered to be an open system as opposed to a closed one, organizational borders are permeable, and organizations relate dynamically to other organizations in the same field.

Reference 10 - 0,15% Coverage

Unlike the micro-level (the single organization) and the macro-level systems (society), the intermediate level systems (organizational **ecosystems**) are hard to see, understand, and describe due to their weak structuring.

Reference 11 - 0,12% Coverage

They are also the most recent type, so there is less experience with them. This relates especially to software engineering **ecosystems**, which is a new but rapidly advancing concept

Reference 12 - 0,13% Coverage

Software **ecosystems** is a more recent term, that refers to a networked community of organizations, which base their relations to each other on a common interest in a central software technology.

Reference 13 - 0,22% Coverage

Well-known examples of communities that may be seen as software **ecosystems** are Apples iPhone/Appstore platform, and the open-source development environment Eclipse. The first is an example of a partially closed and controlled **ecosystem**, and the latter is an example of an open **ecosystem** allowing more flexibility in use and development

Reference 14 - 0,69% Coverage

This simply illustrates that the **ecosystem** concept may refer to a wide range of configurations. Yet, they all involve two fundamental concepts: (1) a network of organizations and (2) a common interest in central software technology. These organizations may have different relations to the central software technology, and for this reason, different roles in the **ecosystem**. There are three key role types:

- First, one organization (or a small group) acts as the keystone organization, and is in some way leading the development of the central software technology.
- The second key organizational role is the end-users of the central technology, who need it to carry out their business, whatever that might be.
- The third key role is third party organizations that use the central technology as a platform for producing related solutions or services. In addition to these key roles, various other related roles might be part of the **ecosystem** (Jansen *et al.*, 2009a), for example standardization organizations, resellers, and operators.

Reference 15 - 0,12% Coverage

Bosch (2009) proposed a Software **Ecosystem** Taxonomy that identifies nine potential classes of the central software technology, according to classification within two dimensions.

Reference 16 - 0,15% Coverage

These responsibilities come in addition to activities specific to **ecosystems** such as enabling efficient external extensibility, provision of insight into planning and development, and supporting **ecosystem** partners in various other ways.

Reference 17 - 0,11% Coverage

Also, the proximity between the organizations in an **ecosystem** may enable active engagement of various stakeholders in the development of the central software technology

Reference 18 - 0,19% Coverage

The ultimate objective for investing in and working towards an **ecosystem** is that all members will gain more benefits from being a part of it, as compared to the more traditional approach for software product development with segregated roles, a low level of collaboration, and closed processes

Reference 19 - 0,09% Coverage

A well functioning **ecosystem** is, in summary, a complex configuration with collaboration across traditionally closed organizational borders.

Reference 20 - 0,07% Coverage

Together, these results show how CSoft relates to its external environment, constituting a software **ecosystem**.

Reference 21 - 0,06% Coverage

This shows the networked character of the **ecosystem** that is shaped around the product line.

Reference 22 - 0,09% Coverage

Software **ecosystems**, as a concept, have the potential of becoming an important field of practice and research in the years to come.

Reference 23 - 0,12% Coverage

Just like the taxonomy suggested by Bosch (2009), a theory of software **ecosystems** is valuable and useful to generalize the concept and bring together results from more empirical studies.

Reference 24 - 0,29% Coverage

It concerns organizations operating in complex and unstable domains, in principle a suitable description of software **ecosystems** – and certainly of CSoft. Using this general theory of organizational ecology, we derive a set of theoretical propositions suitable to software **ecosystems**:

Reference 25 - 1,33% Coverage

1. Member organizations in a software **ecosystem** are linked to a key organization among them, which acts as a central referent organization, doing so even though many of them are only partially under its control or linked to it only through interface relations. [...] However, all activity in the **ecosystem** is related to the product line, which is controlled by CSoft.
2. Software **ecosystems** promote self-regulation.
3. Software **ecosystems** have a networked character.
4. Software **ecosystems** exist through the use of technology. The **ecosystem**, which CSoft is a part of, relies on the use of technology to enable collaboration.
5. Software **ecosystems** have shared values. In the CSoft **ecosystem** the software (product line) is this shared value. [...] These five propositions constitute a start of a theory for software **ecosystems**. In

Reference 26 - 0,55% Coverage

6. The shared value of a software **ecosystem** is both the software product and the business domain.
7. As a software **ecosystem** emerges, the control of and influence on its development becomes a shared responsibility between the supplier and the external environment. [...]. This affects the motivation to collaborate, and is of benefit to all members of the **ecosystem**.

Reference 27 - 0,14% Coverage

Establishing and benefiting from a software **ecosystem** takes time. A successful development relies on repeated cycles of experimentation and learning. This learning process needs to involve all types of actors.

Reference 28 - 0,09% Coverage

The shared interest in the product line (the shared value) is a key enabler for driving the collaboration between the actors in the **ecosystem**.

Reference 29 - 0,14% Coverage

To summarize our study we propose a simple conceptual model of a software **ecosystem**. This model represents the case we have studied, and could serve as a basis to reflect on and guide other similar cases

Reference 30 - 0,10% Coverage

The model illustrates the main actors in a software ecosystem of the type that CSoft is part of – we define this **ecosystem** as a “keystone-centric” type.

Reference 31 - 0,08% Coverage

Note that there are several other potential variations, with more loosely coupled communities controlling the **ecosystem**.

Reference 32 - 0,05% Coverage

Being a part of an **ecosystem** means that these actors learn about each other.

Reference 33 - 0,09% Coverage

According to Yin, such a study may act as a prelude to further studies of a relatively new topic, such as software **ecosystems** in this case.

Reference 34 - 0,12% Coverage

Over a period of approximately five years we have studied a software product line organization and its external environment, showing and explaining an emerging software **ecosystem**.

Reference 35 - 1,32% Coverage

Based on our study we propose the following seven directions for future research:

(1) As Jansen *et al.* also point out (2009a) we need to see more empirical studies of various types of software **ecosystems**, how they develop, and the effects they produce. [...]. In particular it would be valuable to get a better understanding of **ecosystems** as seen from the point of view of external actors.

(2) To build a common understanding of software **ecosystems**: how they shape, how they work, and what their effects are, we advise a further refining of a theory of software **ecosystems**, [...]. One way of developing these concepts would be to apply them to existing well known **ecosystems** such as iPhone app store, MS CRM, Eclipse, *Android* and others.

(3) The emergence of software **ecosystems** comes with new business models affecting intellectual property rights, economic models, competition, etc. We need to see more dedicated studies of these issues to realize the potential of **ecosystems**.

(4) Software **ecosystems** are closely related to the more mature concept of open source software development. We need to better understand the similarities and the differences in order to transfer knowledge between these two related domains (Fitzgerald, 2006).

(5) The engine of a software **ecosystem** is the collaboration with external actors. We have showed some examples through our studies, but this is abroad topic that needs further investigation.

(6) The study of software **ecosystems** potentially relates to several disciplines such as business strategy, sociology, technology and innovation management, economy, and others.

(7) Software **ecosystems** affect the shape of control structures.

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.54. Apache and Eclipse Comparing Open Source Project Incubators>](#) - § 1 reference coded [0,27% Coverage]

Reference 1 - 0,27% Coverage

Recent studies propose the emergence of a new organizational model for F/OSS, called OSS 2.0, based on aggregating communities into **ecosystems** at some point between the “cathedral” and the “bazaar.”

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.77. Collaboration in the open-source arena the webkit case>](#) - § 1 reference coded [0,19% Coverage]

Reference 1 - 0,19% Coverage

Firms relying on WebKit source of innovation, kept differentiating both while porting it to their own architectures and in other areas of their computer-based platform/**ecosystem**.

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.82. Community Source Software in Higher Education>](#) - § 2 references coded [1,31% Coverage]

Reference 1 - 0,35% Coverage

Free and open source software (FOSS) has indeed dramatically changed the software industry and significantly impacted the behavior of different stakeholders in the software **ecosystem**

Reference 2 - 0,96% Coverage

The Sakai educational community license lets any company or organization offer the Sakai Collaboration and Learning Environment (CLE) as a part of their for-profit service offering. The Sakai Foundation encourages the growth of this **ecosystem** through its commercial affiliate program's voluntary annual subscription model (US\$2,000 per \$1,000,000 in company revenues) that provides funding for Foundation activities and visibility to those business partners.

[<Files\\AIME\\Data_Extraction_Nov_2019\\OK.93. Defining and Evaluating a Measure of Open Source Project Survivability>](#) - § 1 reference coded [0,12% Coverage]

Reference 1 - 0,12% Coverage

We derive the measure under the influence of ecology literature [51], [52] where the survivability of natural **ecosystems** is measured in terms of their viability.