Open source innovation platforms: business models and standard strategy

Charleux, Amel
Université de Montpellier
amel.charleux@umontpellier.fr

Mione, Anne
Université de Montpellier
Anne.mione@umontpellier.fr

Résumé :

Dans cette communication, nous explorons le management des plateformes d'innovation collaborative open source par les éditeurs de logiciels. En analysant 199 plateformes, nous identifions et décrivons quatre types de business models d'éditeurs. Nous soulignons la manière dont ces éditeurs gèrent les contributions collectives grâce aux standards et montrons que le choix d'un standard donné a un impact considérable sur la création et l'appropriation de la valeur dans l'innovation de type open source. Nous montrons que les éditeurs vont privilégier l'implémentation d'un standard ouvert lorsqu'ils cherchent à attirer des contributions d'experts, et qu'ils vont choisir des standards dominants lorsqu'ils visent la contribution de masse.

Mots-clés : standards, business models, innovation ouverte, open source

Open source innovation platforms: business models and standard strategy

Abstract :

In this contribution, we explore the editors' management of open source collaborative innovation platforms. Analysing 199 software platforms, we identify and describe four business models. We highlight the way the editors manage the collective contribution through standards and show that the choice of standard has a considerable impact on the value creation and capture in open source innovation. We find that the editors privilege implementing an open standard when they target contribution from experts and that they choose a dominant standard when they target mass contribution.

Key words: standards, business models, open innovation, open source

INTRODUCTION

The development of collaborative economy raises multiple questions to scholarship in management. There is no consensual definition, however, Botsman and Rogers (2010) previously defined collaborative economy as the sharing of resources and competences between individuals in order to decentralize - with new technology - the access to products and services. They highlight four drivers at the origin of this trend: technological innovations, values shift, economic realities and environmental pressures (Botsman, 2014). Among the researches on such phenomenon, some have focused on business models (Acquier, Carbone, & Masse, 2016; Bowman & Ambrosini, 2000; Lecocq, Demil, & Warnier, 2006). Acquier, Carbone and Massé (2016) have specifically described four collaborative ideal types: « annuitants », « collectivists », « altruists » and « matchmakers ». In this contribution, we aim at contributing to the knowledge of business models in the collaborative economy in considering the open source sector. Open source software development is a typical example of

collaborative economy (Benkler & Nissenbaum, 2006). It is based on pooling the developers' competences for programming, and corresponds to the peer economy which is a promising facet of the collaborative economy. It consists of sharing the access to the software code (the recipe of the software) in such a way that any participant is free to shape, transform, adapt and improve the software. The exact philosophy is based on the opportunity to access, modify and redistribute the software source code. The main characteristic is freedom: the contribution is based on the voluntary principle; the software is not patented therefore it is eligible for free distribution. We first raise the question of business models in such a situation. While many studies focus on individual benefits, we choose the organizational perspective and we specifically wonder how do firms engaged in such a process obtain a return on investment.

Secondly, we go deeper into the management of open sector activities. We observe that collaborative economy has also been associated to platform economy (Evans and Gawer, (2016). Defining a platform economy, Evans and Gawer described different forms (2016). One of them is innovation platforms. They consist of technological building blocks that are used as a foundation on top of which a large number of innovators who can develop complementary services or products can contribute. These complementary innovators can be anyone, anywhere in the world, and together they form what is called an innovation ecosystem around the platform. In the open source sector, we consider that any software development can be considered as a platform appealing for contributors. The software editor has to constitute a platform and design technical characteristics such as a development tools, technical standards, programming language and specific rules managing the flow of contributions. We are interested in such mechanisms and observe how the choice of a technical standard relates effectively to the business model.

Thus, our research question is twofold. **First, what are editors' business models for open innovation platforms? Second, how do they perform their strategy through standards?**

We adopt a mixed methodology. A quantitative analysis of 199 software editors is carried out. In order to better recognise different kinds of firm behaviour, we choose to classify them and deduce a taxonomy (Crombie, 1988). A qualitative analysis is then done in order to understand strategic implications of standards' choice on the business models identified.

Our proposition is that open source environment offers a variety of situations according to openness. Thanks to a classification of our observations (Baden-Fuller & Morgan, 2010), we identify and describe four open source business ideal-types. We then show that the choice of a

specific standard is one of the strategic mechanisms to manage this openness and attract or discourage the potential contributors to engage and defend a technology.

We choose the software sector. This sector is emblematic of a situation where private and public, proprietary and open systems coexist. It enables identifying diverse situations from pure players to hybrid profiles that create new business models to find ways of combining the advantages of both closed and open worlds.

In part one, we define the business model concept and give some taxonomies already identified in the literature. The specific role of standards is highlighted. In part two, we describe the method. In part three, we present our open source business models' taxonomy and the related standards strategies in three cases. In part four, we discuss our results and underline theoretical and practical implications.

1. BUSINESS MODEL IN OPEN SOURCE SECTOR AND THE STRATEGIC CHOICE OF STANDARD

We first consider business models in open source innovation before considering how standard choice may constitute strategic mechanisms to manage and develop a platform.

1.1. BUSINESS MODELS IN OPEN SOURCE INNOVATION

Open innovation (Chesbrough, 2003) has been associated to the image of a bazaar as opposed to a cathedral (Raymond, 1998). This openness suggests an acceleration of the innovation which raises the question of commercial viability (Chesbrough, 2006; Teece, 2010). In this context, firms have to review their policy of protecting new ideas: protecting them all induces a cost, while not all of them are likely to be suitable for the market (Chesbrough & Rosenbloom, 2002). In the open source innovation (Pénin, 2011), this acceleration is even more important regarding the huge number of collaborators involved in the innovation process. We thus question the value sharing between all the contributors. Why do firms engage in open source innovation? How do they generate value? Value creation and value appropriation are the two sides of the business model concept. We thus explore how do editors organize the contributions of different participants into the process of innovation. Collaborative economy raises questions and we adopt the concept of business model because solutions have to be found to manage the involvement of the contributors and to share ownership and the common innovation fruits (West, Salter, Vanhaverbeke, & Chesbrough, 2014).

Regarding the question of business models, some authors have already explored the underlying processes of generating revenues and developed taxonomies. Lisein et al. (2009) propose three models adopted by open source companies: complexification, closed system model and intermediation. Complexification concerns specialized companies where human resource is the key competitive advantage. Closed system describes companies whose access to source codes reveals complicated and limit the relation with the community. Intermediation corresponds to free riders. They beneficiate from innovation without contributing to the community. Based on this work, Mouakhar and Albéric (2015) extend the research to a larger panel and propose a taxonomy of three open source sector strategic behaviours. Focusing on the legitimacy dimension (Suchman, 1995), they identify partners, diplomats and profiteers. Their description is based more on the respect of moral and philosophical values than on the implementation of a particular revenue model. However, this taxonomy does not allow a good understanding of the strategic motives of firms belonging to the open sector. Above the distinction between the firm's values in generating merchant activities in a non-merchant sphere, there is a need of clarifying their strategy.

Indeed, both taxonomies have focused on the financial, commercial or moral dimensions. In limiting the perspective on value appropriation, these studies ignore the issue of value creation. In our mind, the value creation represents one of the major component of business models (Johnson, Christensen, & Kadermann, 2008) and we argue that in open collaborative innovation, both sides, creation and appropriation, are interdependant and intricated. Precisely, the articulation between the two elements is essential (Zott, Amit, & Massa, 2011). The interdependence between the business model and the revenue model entails questions of value creation and value appropriation (Zott, Amit, & Massa, 2010). Teece's definition (2010) focuses on how a company delivers value to consumers, encouraging them to pay for that value and converts those payments in profits. It is nothing less than the organizational and financial architecture of a firm (Teece, 2010). Business model focuses on profitability that is not only a question of innovating but a question of finding ways to exploit innovation (Chesbrough & Rosenbloom, 2002). This is even more acute in the case of open source innovation (Chesbrough, Vanhaverbeke, & West, 2006; Pénin, 2011).

Moreover, regarding the specific open source field, the previous studies observe service companies in a large scope whereas it is necessary to distinguish between two major

distinctive activities: software editing and IT services. Positioned differently on the value chain, editors and IT service providers have distinct strategic objectives, which influences their business model choices.

Finally, literature on open source business models does not distinguish between free software and open source movements. Highly competitive in practice, actors of each movement endorse distinctive and specific values, rules and objectives that necessitate a separate analysis.

Studies on business models are increasingly paying attention to the interaction between stakeholders for efficiency and/or novelty (Benavent & Verstraete, 2000; Zott & Amit, 2008). In open environments, this interaction between complementors and between stakeholders is realized mainly by standardization (Almeida, Oliveira, & Cruz, 2011). It concerns all the contributors participating to the innovation process. We choose to examine the standard employed in order to deepen the knowledge of the strategic dimension of business models which requires further investigation (Wirtz, Pistoia, Ullrich, & Göttel, 2016).

In open source sector, the innovation process beneficiates from the contribution of free participants. This entails that the attractiveness of the project is important. In some cases, we think that the network effect may apply. This is due to the platform advantage. Platforms have unique characteristics, with a central feature being the presence of network effects. Network effects are prevalent in platforms, and they mean that more users beget more users, a dynamic which in turn triggers a self-reinforcing cycle of growth (Evans & Gawer, 2016).

The network effect raises the question of standard and standardization. Standards are technical designs enabling to achieve compatibility, interoperability, safety, repeatability, or quality standard. Standardization can be realized through negotiation in formal institutions (*de jure* standard) or through market dominance (*de facto* standard). In network technologies, standards play a key role because they organize interoperability that enables making the network bigger and thus increases the external value of the network. Interoperability achieves this primarily by increasing potential connections and secondarily by attracting new participants to the network. As observe Shapiro and Varian (1999), companies involved in fostering interoperability face a tension between cooperating with their competitors to grow the potential market for products and competing for market share.

In communication and information technologies, open standards and interfaces are often developed through the participation of multiple companies and are usually perceived to

provide mutual benefit. But, in cases where the relevant communication protocols or interfaces are closed standards, the network effect can give the company controlling those standards monopoly power. For example, the Microsoft Corporation is widely seen by computer professionals as maintaining its monopoly through these means. One method Microsoft uses to put the network effect to its advantage is called Embrace, Extend and Extinguish. Yami et al. (2015) examined how Microsoft participated into the *de jure* standardization process of the OOXML standard in order to obtain the institutionalization of its own standard and thus, preserve and strengthen their dominance of the market.

1.2. OPEN STRATEGY AS A SUBTLE OPENNESS OF STANDARDS

In the specific context of open source software, an open standard means that it can be freely adopted, implemented and extended (Simcoe, 2006). According to West (2004), the openness of a standard is graduated by the level of access to creation of specifications, the level of access to results of specifications and the level of access to implementation of specifications. Many specifications that are sometimes referred to as standards are proprietary and only available under restrictive contract terms (if they can be obtained at all) from the organization that owns the copyright on them. As such, these specifications are not considered to be fully opened. West has argued that "open" standards are not black or white but have many different levels of "openness" (West, 2007). In open source, these levels will impact the community contribution process which is at the heart of value creation. This is why we consider that the election of a standard relates to a strategic targeting of potential contributors. In order to measure this proposition, we measure the relation between standard and business model.

Our two objectives are to identify open source innovation business models and then to analyse the related choices in standards.

2. METHOD

Our first aim is to identify and describe the open source software editors' business models. We thus realise a quantitative study of 199 software. Our second objective is to analyse these different business models in the light of standard's choice. For that, we use a case study method and meticulously analyse secondary data coming from web sites, journals and blogs.

2.1. QUANTITATIVE ANALYSIS OF OPEN SOURCE EDITORS' PROFILES

In order to propose a taxonomy, we list 199 free software, designated by the specialized sites as the most used by IT developers[1]. We use Teece's framework (2010) to address the business model concept. Composed of two elements, the Teece's conception of business model allows to capture both the value creation dimension - through the organizational architecture – and the value appropriation dimension – through the financial architecture-. Moreover, it allows the operationalization of the concepts by pointing out concrete firm's elements. We thus consider the organizational and financial architectures as the two dimensions of a business model. We define organizational architecture by 1) the legal status of the entity that own the software and, 2) its organizational form that is more or less open and structured, described by terminology bazaar or cathedral (Raymond, 1998). The financial architecture, for its part, is described by the two components of 1) the intellectual property policy (IP) and, 2) the sources of income.

Table 1. Business model framework

Business Model (Teece, 2010)	Organizational architecture	Legal status
		Organizational form
	Financial architecture	Intellectual property right
		Revenue streams

The composition of our variables is the following;

1. Legal status:

· Community: the software is created by a community and there is no legal status

· Association: the software is published by a non lucrative association

· Foundation: the software is published by a foundation

· Private company: the software is edited by a private lucrative company

2. Organizational form:

· (B): « bazaar » (when there is a community);

· (C): « cathedral » (when there is no community around the software).

3. Intellectual property:

· (CR): Copyright;

· (CLC): Controlled copyright (when copyright grants a certain number of freedoms) ;

· (CLT): Copyleft.

[1] www.open-source-guide.com

4. Incomes:

· (D): Donations;

· (Sp): Sponsoring;

· (S): Services;

· (R): Rents.

Additional variables are considered illustrative variables: the nature of the software defined by the code's status (open or closed) that allows to qualify the innovation process, the governance according to the typology of Demil and Lecocq (2006), inspired by the work of Williamson (1996), and finally the organization of the ecosystem within the meaning of Boudreau and Lakhani (2009) (market or community). To enrich our database, we also completed it with the category of the software, the official website, the license(s) used, the year of software creation, the number of lines of code, the number of contributors, the number of downloads and the programming language used.

The study aims to define profiles of publishers. To do this, we use the TwoStep Cluster method available in SPSS. This method allows revealing natural groups in large databases. We exclude the classification method in dynamic clusters because this method requires to enter a priori a defined number of groups. The selected method imposes certain constraints and assumptions: independence of variables and sorting biases.

2.2. THE CHOICE OF THE STANDARD'S REFERENCE

In the software sector, different elements may be considered as standards. De Vries et al. (2008) consider the program as a standard (internet browser). However, Krechmer (2006) signals that a software or a technological solution is difficult to consider as a standard because of its permanent enhancement. Some others consider the MS Office Word as a standard while it is a program. In this contribution, we consider the language chosen to develop the software. We study the language instead of the software because it is more stable. Also, we look at the standard interface function to constitute a common reference enabling to communicate and interact. The contributors of software share a programming language as a common vocabulary. The programming language have already been studied as a standard, in the ISO *de jure* standardization context (Vion, Diaz, Dudouet, & Graz, 2013; Yami, Chappert, Mione, 2015). In this contribution, we consider the programming language in the *de facto* standardization (Foray, 2002) and we will also observe its spread on the market.

When its installed user basis develops (Farrell & Saloner, 1986), the standard diffuses and gains dominance. We will consider the programming language status regarding its diffusion on the market by examining the annual classification of the Tiobe Index[2]. This index is a reference in software programming. It annually ranks more than 240 programming languages[3]. A language can be considered as a dominant standard when it is widely and largely implemented. A standard is considered as emerging when it is "hype" and growing in use.

2.3. CASES SELECTION AND DATA COLLECTION

The first stage of our work was to select a category of programs. In order to have recent and innovative software, we selected the Cloud Computing category in a list of 350 software established by the Open Source Guide[4] (annex). We obtained a sample of five competing solutions. For each one, we identified the original editor and its open source business model according to the framework of our quantitative analysis. For each program, we reported 1) the year of the first commit (can be considered as a year of market introduction); 2) the programming languages up to three, classified by the % of using in the program[5] and 3) the rank of the main development language, one year before the first commit.

This letter information is key because we consider that the choice of the programming language is made at the beginning of the project, when this one is still under development and not yet opened to the market. When the software is open sourced, often it means that there is a "minimum viable product" that can show the essential features of the software. Technical components are demand side oriented, the choice about them occurs long before disclosure.

3. RESULTS

Our results are twofold. We firstly set up a publishers' business model taxonomy in open source sector regarding to the openness strategy. Then, we highlight the technical standard choice in each model according to its popularity and openness.

3.1. OPEN SOURCE BUSINESS MODELS

3.1.1. Natural open source clusters

[2] http://www.tiobe.com/tiobe_index?page=index
[3] For details on the calculation model, see:
http://www.tiobe.com/tiobe_index?page=programminglanguages_definition
[4] www.open-source-guide.com
[5] www.openhub.net

Chi-square test results confirm variables dependency[6]. Thus, the nature of the company behind the software has an influence on its licensing policy (var. IPR), its organizational model (var. orga form) but also its appropriation of value (var. source of income). The business model represents a comprehensive approach where the choices adopted for each parameter affect the entire model. Despite this variables dependency, the analysis by the TwoStep Cluster method is allowed because it is "strong enough to violations of the independence assumptions."[7]

Our first result shows that it is possible to identify three types of groups among our 199 software and the quality of our clusters is correct (Figure 1).

The weight of each of the four variables in determining the clusters is shown in the figure 2. We observe that the groups were strongly influenced by the variable "income" and the variables "IP". Variables of "legal status" and "organizational form" were less decisive. This was due in large part to the largest number of responses available to the variable "source of income" (7 types of responses against 3 and 4 for the other three variables).

Figure 1. Quality and summery of the 1st TwoStep Cluster

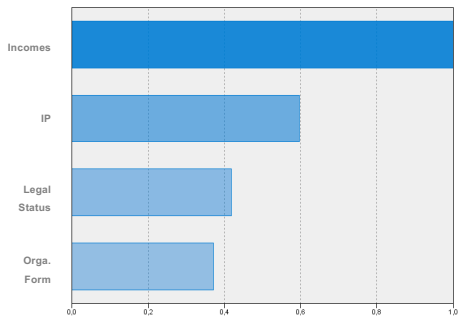


Figure 2. Weight of variables

Table 2. Clusters' distribution

Cluster	N	% of combined	% of total
1	61	30.7%	30.7%
2	66	33.2%	33.2%
3	72	36.2%	36.2%
Combined	199	100%	100%
Total	199		100%

[6] Asymptotic significance less than 0.05
[7] Extract from the SPSS Help section

The cluster size is homogeneous and the size ratio of the largest cluster size on the smallest cluster is less than 3 (1.18) (Table 2).

SPSS offers us three groups defined as follows:

The first group includes private publishing companies whose intellectual property policy is mixed: open license combined with a closed *entreprise* license. Their main source of income comes from the *entreprise* license rents combined with other possible sources that are service and sponsorship. The organization model of their various information flows is that of bazaar model (existence of a community) as well as that of cathedral model (flows inside the firm).

The second group also consists of private publishers but also communities which, unlike the first group, adopted exclusively an open IP policy (GPL and other open source licenses) with a business strategy focused on service and sponsorship. The organization is hybrid with a strong dominance of the bazaar structure.

The third group is made up of communities, foundations and associations. This group has a completely open IP policy. Its revenues come from the donations and sponsorship. Information flows are in the form of bazaar. We notice that this group is more heterogeneous than the previous two where private publishers were highly dominant. In this group, we have three types of actors that are represented at more than 70% each. This group deserves more detailed analysis.

To refine our analysis and have a number of clusters with good representativeness (previously, it was "correct"), we perform a second analysis of our sample by forcing for 4 groups.

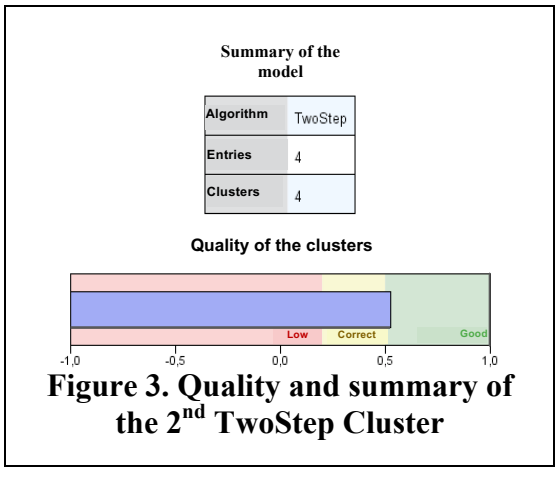


Figure 3. Quality and summary of the 2nd TwoStep Cluster

With this new operation, we improve the quality of our clusters (Figure 3). The distribution between the clusters is also satisfactory as the largest cluster ratio on the smallest cluster is always less than 3 (1.85) (Table 3).

Table 3. Clusters' distribution

Cluster	N	% of combined	% of total
1	61	30,7%	30,7%
2	60	30,2%	30,2%
3	33	16,6%	16,6%
4	45	22,6%	22,6%
Combined	199	100,0%	100,0%
Total	199		100,0%

In the new treatment, the distribution is the same as in the first test and we can identify four types of groups. The first group is the same as identified during the first treatment. The second is also very close to the second cluster identified during the first treatment. Regarding the third group, this one was split in two, which confirms the need of a better refinement.

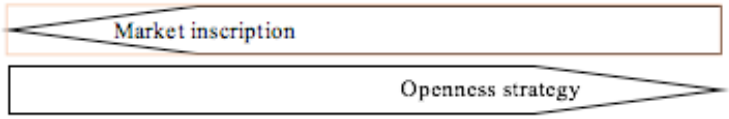
So, the new third group is mainly represented by the foundations, and to a lesser extent by some communities. The license policy adopted by this group is only "open source". Sources of revenue are mainly due to the couple "donations and sponsorship."

Group four is, for its part, made up of communities and associations organized in a bazaar model. Unlike the group 3, the license used is exclusively the "GPL" and the sources of income are strongly related to pure donations.

3.2. THE TAXONOMY

The descriptive analysis of 199 software allows us to identify four types of business models where three are specific to software developed in open source innovation. We propose to call them by separating the business model of engagement, that is specific to pure free software, from the other three business models that we call exploration business model, expertise and optimization. The taxonomy is presented in the following table (Table 4). We can notice at the left the Teece's two business model dimensions. In the last three lines, we can find complementary information that permits clarifying the models. Moreover, the four business models are classified by 1) their commercial and market interest; 2) their openness strategy.

Table 4. Open source editors' business models: open strategy vs. market concerns



		OPTIMIZATION	EXPERTISE	EXPLORATION	ENGAGEMENT
Organizational architecture	Legal status	Private firm	Private firm	Foundation	Non-profit organisation
	Organizational form	Hybrid (Cathedral & Bazaar)	Hybrid to Bazar	Bazaar	Bazaar
Financial architecture	Intellectual property right	Copyright + Copyleft	Controlled copyleft	Controlled copyleft	Full copyleft
	Revenue streams	Rents + Services + Sponsoring	Services	Donations + Sponsoring	Dons
Product	Status of software	Hybrid (open & closed)	Open	Open	Open
Firm	Governance model	Hierarchical + Network	Hierarchical + Bazar	Network	Bazaar
Ecosystem	Ecosystem organization	Market + Communal	Market + Communal	Market + Communal	Communal

Engagement business model: pure free software logic

The business model of engagement is not considered as open source model because its characteristics are those of free software movement. Without any market considerations, this model is exclusively based on voluntary donations that support minimal requirements for software development. Organized under non-profit organizations, actors that initiate projects are fully committed to free software values and engaged actively in a non-profit logic of software editing. The source code of the software is fully open under a GPL license.

Exploration business model: open to innovate.

This business model concerns mainly editors that are registered as foundations. Projects that they develop are supported by a strong network of partners that are involved financially but also in terms of availability of resources. The organizational form is that of bazaar: in one hand, the community is free to contribute to the source code of the software without selective entry; on the other hand, a certain internal control is established for official partners' contributions in order to ensure the quality and stability of the software. This one is offered under an open source license. The income of the foundation comes from various donations but

above all, from an elaborate sponsorship program: the partners engage with amounts that can be very high (for the OpenStack project, for example, the Platinum Partnership is $ 500 000 / year and the provision of two full-time developers dedicated to the project of the foundation). This business model is specific to editors that develop highly innovative technological projects. Their objective is the technological development and research. They are born in favour of projects that could be described as exploratory and are keen to maintain their objectives out of the market constraints. In order to keep a great freedom for their research project, founders rely often on a legal structure that is in charge of finding revenues, hire experts, communicate effectively and invest in infrastructure that maintain a qualitative research environment. Finally, the governance of these explorers is structured in a network with relationships based primarily on exchange and reciprocal incentives (Demil & Lecocq, 2006). These incentives are of average intensity: the best developers can be hired by the partners, invited to events or become ambassadors; professional partners benefit from a valuation system that lead to certification as "expert" and from a better visibility on the project's website.

Expertise business model: open to create value

This business model concerns editors that are registered as private companies. They rely on their software development competences and not on the proprietary rents that protection of this one can bring. In this model, the software is open sourced thanks to a copyleft licence. All the community can participate to its development. However, this participation is framed with road maps and steering committees, and the organization is considered as hybrid bazaar. Unlike the model of exploration, developments should remain close to the market needs and not only driven by the technological research. The main incomes of these editors comes from the services added to the software. That is why software should be close to consumers' need to promote additional services. Also, the quality of contributions is a key factor for the editors' reputation and notoriety. Again, because their incomes are linked to the diffusion of the software and because this diffusion is conditioned by the quality of the code and features, the community has to create true value with its contribution. Regarding the governance, editors can deploy a hierarchical model within their organization and a bazaar model with their community. The expertise business model is highly adapted for those who wants to fulfil with open source innovation without giving up the economic concerns.

Optimization business model: open to make business

This model refers to what practitioners call "open core editors". It concerns private organizations that propose their software under two formulas: an open and community based version which is the core of the software; and a closed and paying version that includes all the features developed internally. The first version is proposed under an open source license which is compatible with commercial concerns (copyleft), and the second improved version is proposed under an *enterprise* licence (copyright). The letter version relies heavily on the community version but is maintained only by the employees of the private organization in a "cathedral" way. The free of the first version is allowed by the rents that the second one can bring. This two speed model attract more and more editors and is at the cross of proprietary and open models. Editors that adopt this business model benefit both the community emulation and creativity, and the secure of a more controlled version that can be sold. In addition to proprietary rents, there is an important part of these editors' incomes that is linked to the services they can provide.

The open source business models described above show that there are specific logics in open worlds. They illustrate how open source actors beneficiate from their investment in this specific innovation model, based as we see previously, on the principle of platforms. Our next result highlights the particular role of standards in managing these platforms.

We consider the choice of the programming language as technical standard. It is qualified according to its statute of dominant, emergent or nascent. We expect to see how the choice of a technical standard relates effectively to the business model.

3.2.1. Balancing between openness and popularity

The table 5 below summarizes our selection of five case studies from the Cloud Computing category, and the business models corresponding. We will then identify the programming languages that are used for each case to define the software.

Table 5. Software and editors' business model in Cloud Computing category

Software	Editor	Characteristics	BM
OpenStack	OpenStack Foundation	**Organizational Architecture** Legal Status: Foundation Organizational form: Bazar **Financial Architecture** IP: Controlled copyleft Income sources: sponsorship and donations	Explorer
OpenNebula	OpenNebula Systems	**Organizational Architecture** Legal Status: Private firm Organizational form: Hybrid **Financial Architecture** IP: Controlled copyleft	Expert

		Income sources: services	
CloudStack	Apache Foundation	**Organizational Architecture** Legal Status: Foundation Organizational form: Bazar **Financial Architecture** IP: Controlled copyleft Income sources: sponsorship and donations	Explorer
Eucalyptus	Hawlett Packard	**Organizational Architecture** Legal Status: Private firm Organizational form: Hybrid **Financial Architecture** IP: Hybrid Income sources: rents, services	Optimiser
Nimbus	University of Chicago	**Organizational Architecture** Legal Status: University Organizational form: Bazar **Financial Architecture** IP: Controlled copyleft Income sources: sponsorship and donations	Explorer

In this list, we excluded CloudStack because it is supported by the Apache Foundation that supports many other programs. Thus, it was not relevant to observe strategic implications of technical standard choice of only one program among tens of others that constitute the Apache Foundation activity. We also excluded Nimbus edited by the University of Chicago because it represents an academic exploration that imply specific considerations.

Our final sample (Table 6) is made up of OpenStack, edited by the OpenStack Foundation; OpenNebula, edited by OpenNebula Systems and Eucalyptus, edited by Hawlett Packard. We focus on the major programming language used for the development (Lang.1).

Table 6. Case studies and their programming languages

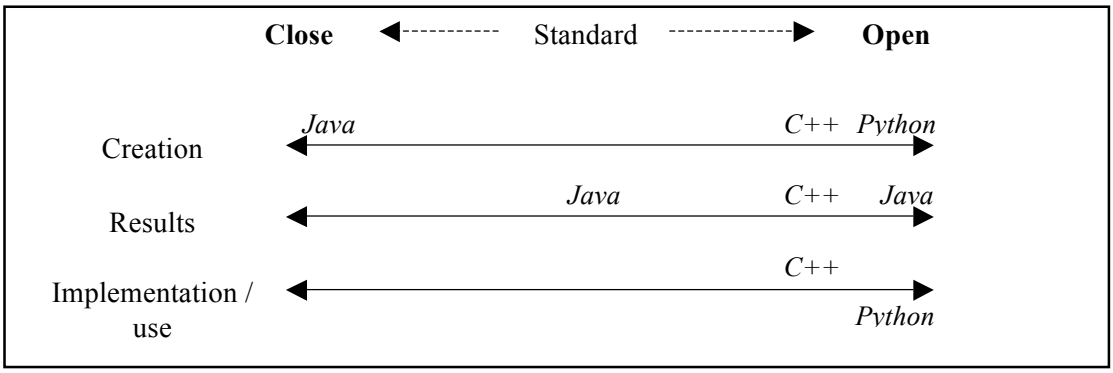
Program	Editor	BM	Year of creation	Lang. 1	Lang. 2	Lang.3	Rank Y-1 Lang. 1	% of user base
OpenStack	OpenStack Foundation	Explorer	2006	Python (77 %)	XML (13 %)	Otheres (10 %)	7	2,506%
OpenNebula	OpenNebula Systems	Expert	2008	C++ (48 %)	Ruby (36 %)	Shell (11 %)	3	10,425%
Eucalyptus	Hawlett Packard	Optimiser	2009	Java (54 %)	C (14 %)	Groovy (6 %)	1	20,849%

One year before the first commit of OpenStack in 2005, Python, the main programming language of this software had 2,506% of the user base and thus, was ranked in the 7th place of the top programming classification. For its part, C++, the language adopted by OpenNebula was in 3rd position with 10,425% of the user base. Finally, in 2008, one year before the first

commit of Eucalyptus, Java, its main programming language was ranked as number one in the programming languages classification with 20,849% of the user base.

Once we observed the popularity of the languages used by our three editors, we observe their openness. For each one, we graduate it according to West's (2004) definition of open standard (Figure 4).

Figure 4. Programming language openness through West's graduation



Java is one of the most popular programming language created by Sun Microsystems that was bought by Oracle in 2009. According to West's graduation, Java is not a full open standard. It is a dominant standard because it is widely used (according to the Java website, there is more than 9 billion developers using it all around the world), but it can't be considered as totally open because it is owned and maintained by a specific vendor which is Oracle. The process of creation is furthermore closed to a restricted membership and conducted by Oracle. The results of the language creation and updates are publicly available and downloadable whereas some elements of the language specifications are patented. The implementation and the use are fully opened without any selection.

The first version of C++ was published in 1983. This language was standardised by the ISO in 1998. Since that, three updates were published and the last one was published in 2014. The next standard is planned for 2017. This standard can be considered as fully open because from the beginning, it was open sourced by its creator, Bjarne Stroustrup. The C++ language belongs to nobody. The standard creation process is also completely open and all the drafts are proposed to the community through the GitHub platform[8]. Access to results and implementation are free too. The C++ community is supported by a non profit foundation that has a mission of promoting and supporting financially the standardization work.

[8] GitHub is the main collaborative platform for open source projects: https://github.com/cplusplus/draft

Python was created in 1990 and publicly open sourced for its first version in 1991. It is supported by the Python Foundation that was created for managing the trademark and the financial supports of the community. Python is a technical *de facto* standard. The specifications are documented, collegially decided through the PEP program (Python Enhancement Proposal) and freely available for implementation to everyone.

Our analysis shows that Java is a dominant programming language but not fully open. C++ is opened and structured through Standard Development Organisations (SDOs). The language is widely used without being dominant (3rd position). Finally, Python is a fully open language but used by few developers.

3.2.2. Technical standards and open source business models: a positive relation

While observing the popularity of the languages and their openness, regarding to the projects that they allow to develop, we can grab strategic considerations of editors: there is a link between the technical standard popularity and the market orientation of the business model, and at the same time, the openness of the standard and the openness of the editor's strategy (Table 7). Thus, standards constitute a strategic mechanism to manage and develop an innovative software platform.

Table 7. Relation between technical standards and business models

Project *Editor*	Eucalyptus *Hawlett Packard*	OpenNebula *OpenNubula Systems*	OpenStack *OpenStack Foundation*
Technical standard choice	Java	C++	Python
Standard popularity	Dominant ←	Emergent	→ Nascent
Standard openness	→		
Business Model strategy	Optimiser	Expert	Explorer
BM market orientation	←		
BM openness strategy	→		

Dominant standard for a wide platform to make business, the case of optimisers:

As the main objective of editors of this category is to optimise their investments in relying both on community advantages and proprietary potential rents, they naturally tend to use large dominant technical standards to develop their programs. This strategic decision promotes the development of a broader platform for community contributions. Because the standard is

dominant, they can expect to interest a larger base of developers that knows and uses this standard and thus, can contribute to the improvement of the solution. The community version is an important driver for this business model because it is a pool of inspiration for the proprietary version and because it legitimates the services offered around the software.

Some literature focused on the negative effects of the standardization on creativity and team effectiveness (David & Rothwell, 1996; Thompson, 1965). In our case, editors count on the standard to aggregate and motivate contribution around the platform and thus, innovation and creativity for their products. In line with conclusions of Rosenberg (1976), the presence of a standard is able to reduce uncertainty and grant a widespread diffusion of a technology. In our case, an open standard that is dominant ensures not only the diffusion of a technology but moreover the expansion of innovation and thus, the expansion of the business. The technical standard choice is a key component of the optimisers' business model construction because it impacts the two essential concerns of the concept: the value creation (the standard allows the aggregation of a community), and the value appropriation (if the software attracts massive contributions, this will allow a more precise development of the proprietary version that will fit with the real needs of the market and thus, grant revenues for the editor).

Emergent standard to create value around the platform, the case of experts:

Developing a specific expertise on a dominant open standard is difficult. There are many firms that can do it and the market is quickly saturated. This situation led to intensive competition and the falling price of the services that can be proposed. In contrary, choosing to built a software - and thus a service offers - on an emergent technical open standard, that is not yet dominant but in a process of growing adoption, can procure interesting opportunities. Be among the first to propose solutions relying on this kind of standards and to be recognized for that is a great manner to ensure revenues from sophisticated services. In this case, the notoriety is crucial. The capacity of the editor to aggregate a qualitative community around its platform using an emergent standard is a positive technical signal to the market. It positions the firms as an expert in its field.

The experts' business model is closely and positively linked to the emergent standard adoption. In open worlds, the notoriety is a competitive advantage that can be built on technical skills of a firm. Thus, technical choices are components of this competitive advantage. To be capable of managing and developing a community around a solution that uses emergent open standard is a key component of the expertise building.

Nascent technology to a highly selective and innovative platform, the case of explorers:

The explorers' business model is the ideal type of open source business model. It is totally communal, without any commercial versions and it focuses on technological excellence. Financial and market concerns are committed to organisation (often a foundation) that acts as a sponsor of the software. This edition model allows founders to concentrate on exploration aspects and to have a very close relation with a more restricted but highly qualified community. The objective of this model is not to make big profits but just to have enough revenues to continue exploration. Thus, the language of software development adopted can be totally new and nascent. It is even better if the technical standard is practiced by few insiders because the value of their contribution is even more precious and valuable. For editors of this kind, exploring new features is the main driver and often, for doing so, new technical tools are needed. But because we are in open source sector, these new tools need to be also well documented and freely available for allowing the platform to emerge and community to contribute. That is why the core strategy of explorers are closely related to their technical standard choices.

4. DISCUSSION

4.1. IMPLICATIONS FOR RESEARCH

The present study intends to better understand the business models of collaborative economy and analyse the standard's role in managing platforms that support them. We observe open source software sector and show that contrary to what might have been expected, open models do not necessary require the implementation of open standards. We show that this relation depends on business concerns.

First, open source software sector represents an emblematic and specific case of collaborative economy. Relying on the free and communal contribution, it supposes new business strategies (Chesbrough & Rosenbloom, 2002). In this contribution, we explored the nature of standards implemented by the editors while building their collaboration platforms, according to the business models. This relation has never been observed before. Our first contribution enriches the literature on the business model by proposing a new taxonomy of open source software editors. Our findings reveal three business orientations that oscillate between two continuums: the openness strategy vs. the market inscription. Openness strategy is used for the purpose of value creation, which represents the first component of a business model (Chesbrough & Rosenbloom, 2002; Mansfield & Fourie, 2004). Market inscription is motivated by the more

or less value capture interest, which represents the second major business model component. Editors using open source innovation to create value have to balance between the attraction of a potentially unlimited source of contributions and the need of being profitable that suggests a certain closure. We characterize these three editors types as optimiser, expert and explorer.

Second, we contribute to the literature on standards by highlighting the business implications of the standard choice in the platform construction strategy. Our study shows that orientations of a firm in terms of value creation and value capture are tightly related to the nature of the technical standard chosen in the beginning of the project, i.e. in the supply phase (Anderson & Tushman, 1990; Clark, 1985; Tassey, 2000; Weiss & Birnbaum, 1989). The results of our exploratory case study confirm that in open worlds like open source software, the choice of technical standard is related to the choice of business model. To do so, we had to distinguish between technological and technical standards and to analyse open standards in their complexity: 1) regarding to the degree of openness and 2) to the market success of these technical standards. Relying on the percentage of market share, we differentiated dominant, emergent and nascent standards. We showed that the more the editors' concerns are market oriented, the more the standard is dominant for developing wide and easily accessible collaborative platforms. And in the same time, the more the editors' interest for openness and value creation is high, the more the standard is open - according to West's graduation. Finally, open standards are preferably chosen even when this openness is not as pure as it could be expected, which is mostly the case in open source innovation.

4.2. IMPLICATIONS FOR PRACTITIONNERS

We pointed out the importance of the standard's choice in open source software industry. Open environments are complex. They are regulated through specific rules. Management requires a delicate balance between the compliance with the openness duty and the need of profitable structure. In this context, the role of the supply side standard is determinant. We showed how the two dimensions of openness and market success of a technical standard had to be arbitrated since the early beginning of the software development. Later, as some scholars showed (Cusumano, Mylonadis, & Rosenbloom, 1992b; de Vries et al., 2008), the business model itself impacts the standardization strategy of the firm on the demand side (what kind of sponsoring, standardization in committees or consortiums etc.).

4.3. LIMITATIONS AND FUTURE RESEARCH DIRECTIONS

Our exploratory research opens up new research paths because there are many aspects to observe and to complete in order to better understand the link between standards, openness and strategic firms' choices. First of all, our cases are exclusively observed in open source sector while it would be important to introduce in the study cases that belong to proprietary software sector. Even if our optimiser case includes a proprietary dimension, it is not a pure proprietary player. Moreover, it is necessary to observe the role of standards in other open fields like cultural sector, automotive or pharmaceutical industries where more and more open projects are experienced. Further in depth cases studies are necessary to understand the causal relation between the use of standards and strategic orientations of the firms' following an open strategy. In the present paper we observe a relation between the technical standard choices and the business model construction without addressing the exact role of the standard in the construction of the business model. Is it a determent or just a contingent context? Do managers really and practically select the technical standard they want to use in investigating its business consequences or is it just a natural orientation? These questions are in line with Astley and Van de Van's (1983) ontological approach of the standards when they suspect that the most difficult dimension to address in standardization literature is that of deterministic – voluntaristic behaviour. We suggested that the standard's choice is related to the business model and the nature of innovation platform desired. Our first results bring support to this hypothesis. Further researches on standard's adoption choice would enrich this contribution. One new perspective could be showing the consequence of the adopters' decision in platform competition.

REFERENCES

Acquier, A., Carbone, V., & Masse, D. (2016). L'économie collaborative : Fondements théoriques et agenda de recherche. *International Workshop on the Sharing Economy 2*.

Almeida, F., Oliveira, J., & Cruz, J. (2011). Open Standards and Open Source: Enabling Interoperability. *International Journal of Software Engineering & Applications*, *2*(1).

Anderson, P., & Tushman, M. L. (1990). Technological discontinuities and dominant designs: A cyclical model of technological change. *Administrative Science Quarterly*, *35*.

Astley, W. G., & Van de Ven, A. H. (1983). Central Perspectives and Debates in Organization Theory. *Administrative Science Quarterly*, *28*.

Baden-Fuller, C., & Morgan, M. S. (2010). Business Models as Models. *Long Range Planning*, *43*(2–3).

Benavent, C., & Verstraete, T. (2000). Entrepreneuriat et NTIC: construction et régénération des Business-model. In *Histoire d'entreprendre – les réalités de l'entrepreneuriat*. Caen.

Benkler, Y., & Nissenbaum, H. (2006). Commons-based Peer Production and Virtue. *Journal*

of Political Philosophy, *14*(4).
Botsman, R. (2014). The Sharing Economy Lacks A Shared Definition. Retrieved from w.fastcoexist. com/3022028/the-sharing-economy-lacks-a-shared- definition Botsman
Botsman, R., & Rogers, R. (2010). *What's mine is yours How collaborative consumption is changing the way we live* (Collins). London.
Boudreau, K. J., & Lakhani, K. R. (2009). How to manage outside innovation. *MIT Sloan Management Review*, *50*(4).
Bowman, C., & Ambrosini, V. (2000). Value creation versus value capture : towards a coherent definition of value in strategy - an exploratory study. *British Journal of Management*, *11*.
Chesbrough, H. (2003). *Open Innovation, The New Imperative for Creating and Profiting from Technology*. Boston: Harvard Business School Publishing Corporation.
Chesbrough, H., & Rosenbloom, R. S. (2002). The role of the business model in capturing value from innovation: evidence from Xerox Corporation's technology spin-off companies. *Industrial and Corporate Change*, *11*(3).
Chesbrough, H., Vanhaverbeke, W., & West, J. (2006). *Open Innovation : Researching A New Paradigm*. Oxford University Press.
Clark, K. B. (1985). The Interaction of Design Hierarchies and Market Concepts in Technological Evolution. *Research Policy*, *14*.
Crombie, A. C. (1988). Designed in the mind: western visions of science, nature and humankind. *History of Science*, *26*.
David, P. A., & Rothwell, G. S. (1996). Standardization, diversity and learning: Strategies for the coevolution of technology and industrial capacity. *International Journal of Industrial Organization*, *14*(2).
de Vries, H., de Vries, H., & Oshri, I. (2008). *Standards-Battles in Open Source Software*. Basingstoke, UK: Palgrave Macmillan.
Demil, B., & Lecocq, X. (2006). Neither Market nor Hierarchy or Network : The Emerging Bazaar Governance. *Organization Studies*, *27*(10).
Evans, P. C., & Gawer, A. (2016). The Rise of the Platform Enterprise A Global Survey. *The Center for Global Enterprise*, (January).
Farrell, J., & Saloner, G. (1986). Installed Base and Compatibility: Innovation, Product Preannouncements, and Predation. *The American Economic Review*, *76*(5).
Foray, D. (2002). Innovation et concurrence dans les industries de réseau. *Revue Française de Gestion*, *3*(139).
Johnson, M. W., Christensen, C., & Kadermann, H. (2008). Reinventing business model. *Harvard Business Review*, *86*(12).
Krechmer, K. (2006). Open Standards Requirements. *International Journal of IT Standards and Standardization Research*, *4*(1).
Lecocq, X., Demil, B., & Warnier, V. (2006). Le business model, un outil d'analyse stratégique. *L'Expansion Management Review*, *123*.
Lisein, O., Pichault, F., & Desmecht, J. (2009). Les business models des sociétés de services actives dans le secteur Open Source.
Mansfield, G. M., & Fourie, L. C. H. (2004). Strategy and business models—strange bedfellows? A case for conver- gence and its evolution into strategic architecture. *South African Journal of Business Management*, *35*(1).
Mouakhar, K., & Hachard, V. (2015). Quels Business Models pour les organisations hybrides : une opérationnalisation de l'approche RCOV au sein du secteur de l'open source. In *AIMS*.

Pénin, J. (2011). Open Source Innovation : Towards a generalization of the open source model beyond software. *Revue D'économie Industrielle*, *136*.

Raymond, E. (1998). La cathédrale et le bazar. Retrieved from http://www.linux-france.org/article/these/cathedrale-bazar/cathedrale-bazar_monoblock.html

Rosenberg, N. (1976). *Perspectives on Technology*. Cambridge University Press.

Shapiro, C., & Varian, H. R. (1999). *Information rules: A Stragegic Guide to The Network Economy*. *Harvard Business School Press* (Vol. 32).

Simcoe, T. (2006). Open Standards and Intellectual Property Rights. In *Open Innovation: Researching a New Paradigm*. Oxford University Press.

Suchman, M. C. (1995). Managing Legitimacy: Strategic and Institutional Approaches. *Academy of Management Review*, *20*(3).

Tassey, G. (2000). Standardization in technology-based markets. *Research Policy*, *29.*

Teece, D. J. (2010). Business Models, Business Strategy and Innovation. *Long Range Planning*, *43*(2–3).

Thompson, V. A. (1965). Bureaucracy and Innovation. *Administrative Science Quarterly*, *10*(1).

Vion, A., Diaz, P., Dudouet, F.-X., & Graz, J.-C. (2013). Normalisation informatique et marchés de services : le cas du XML. In *Services sans frontières : mondialisation, normalisation et régulation de l'économie des services*.

Weiss, A. R., & Birnbaum, P. H. (1989). Technological Infrastructure and the Implementation of Technological Strategies. *Management Science*, *35*.

West, J. (2004). What are Open Standards? Implications for Adoption, Competition and Policy. In *Standards and Public Policy Conference, Federal Reserve Bank of Chicago*.

West, J. (2007). The Economic Realities of Open Standards: Black, White and many Shades of Gray. In *Standards and public policy*. Cambridge University Press.

West, J., Salter, A., Vanhaverbeke, W., & Chesbrough, H. (2014). Open innovation: The next decade. *Research Policy*, *43*(5).

Williamson, O. E. (1996). *The Mecanisms of Governance*. New York: Oxford University Press.

Wirtz, B. W., Pistoia, A., Ullrich, S., & Göttel, V. (2016). Business Models: Origin, Development and Future Research Perspectives. *Long Range Planning*, *49*(1).

Yami, S., Chappert, H., & Mione, A. (2015). Séquences stratégiques relationnelles : le jeu coopetitif de Microsoft dans le processus de normalisation OOXML. *M@nagement*.

Zott, C., & Amit, R. (2008). The fit between product market strategy and business model: implications for firm performance. *Strategic Management Journal*, *29*(1).

Zott, C., Amit, R., & Massa, L. (2010). *The Business Model: Theoretical Roots, Recent Developments, and Future Research*.

Zott, C., Amit, R., & Massa, L. (2011). The business model: Recent developments and future research. *Journal of Management*, *37*(4).

Annex

Software categories

Infrastructure: 100 software

http Accelerator (2)	Cloud Computing (5)	Remote control (2)
Deployment and backup (11)	Park management (5)	High availability (4)
Security (8)	Virtual private network (2)	Firewall (9)
Supervision and metrology (10)	OS Linux & BSD (9)	Virtualisation (6)
VoIP / Telephony (10)	Emailing & GroupWare (13)	Others (4)

Development and intermediate layers (133)

Company directory (4)	Data base (17)	Big Data (4)
BPM / Workflow (3)	Development tools (18)	Testing & continuous integration(12)
Enterprise service bus (9)	Library (17)	Frameworks mobiles (2)
Authentication, federation and identity management (8)	Load testing tools (6)	Public key infrastructure (4)
Extract Transform Load (2)	HTTP server and application servers (10)	Search engine (9)
Others (8)		

Applications (117)

Customer Relation Management (6)	Reporting (3)	Content Management System (23)
E-commerce (10)	Enterprise Resource Planning (4)	Portal (8)
GED & EDM (8)	Library and documentation (4)	e-Learning (5)
Social networks (9)	Blogs, wiki et forums (11)	Product Information Management (2)
Audience tracking (2)	Document Sharing Tools (7)	Digital Asset Management (7)
Decision-making (4)	Others (3)	