

Programme “Society and Future”

Final report - Research summary

PROJECT ACRONYM: **OSSPA**

TITLE : "LOGICIELS LIBRES ET POUVOIRS PUBLICS – OPEN SOURCE SOFTWARE AND PUBLIC AUTHORITIES"

RESEARCH CONTRACT: [TA/11/24](#)

RESEARCH TEAM (and institution) :

CO-ORDINATOR (if applicable) : —

PROMOTER(S): PROF. P. DESMAREZ, METICES-TEF, ULB

RESEARCHER(S): A. VANHEERSWYNGHEL, V. CORTESE, M. ZUNE

DATE : APRIL 2012



The purpose of this summary is to present the results of the research on the Internet.

Introduction

Community governance: from the bazaar model to hybridization of logics

According to de Laat (2007), the interest of human science researchers for open source software (OSS) has gone through a three-part evolution. The initial phase involved collecting works related to the apparent “spontaneous governance” of projects. Works were focused on questions related to the developers’ motivations (Ghosh et al. 2002), starting from the so-called “bazaar” hypothesis, (Raymond, 1998), according to which these practise communities, by short-circuiting classical institutional boundaries, would be based on forms of self-governance that are averse to the classical form of co-ordination and control, and would thus *spontaneously* create innovation (De Laat, 2007, p. 167). Several authors have tried to conceptualize this new form of co-operation, by contrasting it with the classical model of the software industry. In this context, Demil and Lecocq (2006), have demonstrated the viability of a production model as an alternative to co-ordination modes based on markets, hierarchy, or networks; von Hippel and von Krogh (2003) have evidenced that open source could be described as hybrid between individual and collective innovation models.

During the second phase, the basic premise of the self-governance of free software development communities was questioned, based on studies of important and sustainable projects (Apache, Debian, Mozilla, etc.). The study of these cases evidenced forms of explicit regulation: code modularization, specialization of roles, vertical division, decision structures, selection at entry, then training and socialization for newcomers, formalization of the commitments, etc. Confronted with this diversity of activity regulation mechanisms, a few authors have identified several organizational free software configurations. Starting from the classical works by Burns and Stalker (1961), De Laat (2007) identified two typical configurations: the “autocratic-mechanistic” structures (strong leader and progressive structuring towards centralization) and “democratic-organic” (egalitarian ethos, strong horizontal division, but weak on the vertical dimension, strong socialization. This distinction concurs with the opposition by O’Mahony and West (2005) between an “organic” form, driven by community logic, and a “synthetic” form, linked or controlled by firms, despite the fact that firms put the focus more on the importance of environment on structuring. After evidencing the links between community-based and private logic, several researchers studied free projects from the viewpoint of hybrid logic (Shah, 2006; Fosfuri et al. 2008). In this context, Dahlander and Magnusson (2005) have identified three strategies firms use to approach communities: the “access” strategy (remote support and/or code reuse), the “alignment” strategy (infiltration of communities and/or liberation of code that once was proprietary) and the “assimilation” strategy (making personnel available/or hiring contributors).

De Laat (2007) indicated that current research was entering a third development phase focusing on the relationships between communities and external actors in order to guarantee independence, either through the sophistication of user licenses, the management of intellectual property lawsuits, but also the development of legal forms like foundations in charge to manage and protect the interest of the projects. By extension, research is also interested in the issue of institutionalization of communities, i.e. their persistence over the long-term and their embeddedness in complex organizational arrangements.

However, while these works help understand the evolution of the organizational forms of communities and the intertwining of private and community-based logic, two limits should be set. First, these works do not provide significant information on the way these logics live together in the regulation activities of the communities. In other words, the analysis remains centred on the motivation of actors to participate, or their “business” strategies, with no attempt to understand the cooperation, or the elaboration of regulation mechanisms between

heterogeneous logics; in short, the construction of collective action. Secondly, public organizations¹ are absent from this view, despite the fact that they represent a significant percentage of the OSS user base, and that through various mechanisms, they contribute to their development.

The use and the involvement by public authorities: studies in search of cases

The public actor is indeed often implicitly linked to the development of OSS: universities, research and public studies centres, public services requiring a large-scale IT infrastructure, are all environments in which developers start dedicating part of their time to free software projects. This involvement is however reinforced by political incentives to use free software, based on concerns in terms of interoperability, containment of public spending, even by a mission of public interest development that would encounter, through an effect of value convergence, the development communities.

Several studies, among which the FLOSSPOLS study (2005) conducted by the U. of Maastricht following the FLOSS survey on developers' motivations (Ghosh et al., 2002), evidenced the significant progression of OSS in public organizations, due to the growing use of IT in public services. With the analysis of the answers supplied by the 955 respondents from public organizations in 13 European countries, researchers shown that most of the organizations declare that they use free tools, even if they noted a relative lack of knowledge regarding software specificity. National differences were pinpointed, with significant variations between countries: Austria (which has a very high use rate, but paradoxically, a strong lack of knowledge regarding the differences between free software and proprietary software), Germany, Belgium, Sweden, France, Spain and Italy have the highest use rate, while the United Kingdom, Greece, and to a certain extent the Netherlands seem to be the most refractory.

These usage-focused surveys provide interesting contextual data, but do not help the detailed understanding of the dynamics behind the introduction of OSS in public administrations and the practice of the freedom of code modification and articulation with the initial communities. So, other researchers tackled the paradox consisting in the fact that, while the definite advantages of OSS solutions (no more licence costs, independence, influence of public organizations on local development, flexibility and internet innovation, collaboration between public organizations, see Titterton, 2003; Simon, 2005; Waring & Maddocks, 2005 ; Ghering, 2006 ; Weber, 2004) should logically plead for a "natural" diffusion of these solutions in public organizations, their use actually remains relatively modest and confined.

Surveys based on case study analysis with public organizations that have chosen OSS provide contrasting results. Waring and Maddocks (2005) conducted a study on the relatively low use of OSS in 8 English local and national administrations, which had benefited from government financing. Motivations related to cost reduction were the first evoked by the promoters of these projects (which confirms other results, see Applewhite, 2003; Banahan & Taylor, 2003), but also underlined the difficulty to maintain these practises over time, due to the lack of internal competence (which evokes the trend towards IT externalization policy to private service providers over the past twenty years). Meanwhile, Rentocchini and Tartari (2008) have conducted a similar study in the Emilia-Romagna region: here these practices are more widespread and supported over time, and the independence towards suppliers seems to be the main factor of adhesion. The objective of the works by Huysmans et al.

¹ By Public Organization we mean every organization that is publicly funded, whether it be an administration, a university, a hospital, etc. This definition is a bit broader than the first perimeter of study which was related to Public Authorities.

(2008) and Morgado et al. (2007), based on case studies of (non-) adoption of OSS by administrations, was to go behind the simple enumeration of favourable or unfavourable factors declared by those responsible of the project, to take the context of migration projects (technical, administrative, financial, legal), by more precisely identifying the dynamics that are favourable or not for a change. They thus shifted the question by showing that beyond a generic evaluation of OSS solutions in terms of advantages and disadvantages, case studies are enhanced when tool introduction processes are identified, as well as the way in which the actors – IT departments, decision-makers, users, opponents, etc. – are involved and contribute to the change. The purpose of the case studies conducted by Cassell (2008, 2009) in four German and Austrian cities was to study the issue of the influence of the political and administrative context on the migration processes towards OSS. More specifically, the author described the way the arguments presented by the various parties involved in these projects are bound to evolve and are re-interpreted over time. For example, the issue of cost is subject to many different interpretations, since the effects of cost cutting can only be assessed over the long term. This is why the TCO indicators – Total Cost of Ownership – are developed in such a way that they can not be considered as evaluation tools *ex post*. In other words, while the argument of cost cutting is a premise of “innovation by necessity” (Kotter, 1996), and allows the adhesion of all the parties involved, new considerations emerge over the course of the project, progressively shifting the attention and contributing to the definition of new objectives, notably regarding the design of new tools, the development of new relationships with more local or emergent suppliers, but also the co-operation between services located in different cities in the development of common features.

The starting point of these case studies is the migration practises of internal IT in administrations towards OSS use. Except for the works by Cassell, research performed has a tendency of considering OSS solutions as technical alternatives, and migrations under the angle of common technological change processes. The focus is on public administrations as users of existing products. It's only indirectly that new development capacities that would be generated by these migration processes are evoked. In this context, Cassell noted that several cities studied had started to share their internal developments with other entities, but had not gone deeper into the issue of cooperation between administrations, and of course nor between administrations and communities. Finally, the relationships with the new service providers are not studied, only relatively vague conclusions are drawn, regarding the independence towards *vendors* and the avoidance of *lock-in* effects, without getting deeper into the nature of the configurations in terms of re-constructed relations between actors.

The overall objectives

This literature review helps identify the two current limits of the knowledge. First, studying the way communities function requires further analysis work to better grasp the nature and the specificity of this collective action mode in OSS communities of development, based on in-depth empirical data and detailed case studies. Secondly, the issue of the influence of public authorities in the production of OSS remains relatively obscure.

Conceptualizing the functioning of community-oriented collective action

The first objective of our research is to contribute to the understanding of the functioning of OSS development collectives, by not focusing so much on their technical or legal characteristics, which could constitute specific products, but rather on the production processes based on community-based functioning modes. In fact, the initial enigma at the centre of our questioning was the nature of the organizational forms that developed in this manner. Indeed these production communities do not show any characteristics usually linked to productive organizations: contract-based commitments, remunerated contributions,

workers' co-presence, hierarchical control, etc. Considering this, how can multiple, scattered and independent contributions be produced and put together to obtain a coherent and valuable product? How can ideas and production of participants distant one from another be coordinated, combined, sorted? Are there rules or systems ensuring the management functions, which usually structure productive activity? How can this regulation materialize and how can it function? How is the tension between co-ordination and evaluation constraints originating from the production objective and the exigencies in terms of autonomy and initiative generated by the voluntary and benevolent character of the individual participations handled, managed?

Understanding the logics behind the involvement and cooperation of public authorities in the projects

The second objective of this research – closely linked to the first – is to achieve a more insightful analysis into projects characterized by a certain complexity due to the heterogeneity of the contributors' implication logics, including contributors from public organizations. Several sources identified the cases of projects promoted or supported by public logics (IDA, 2001; PS-OSS, 2006, OSOR.EU). They show that multiple levels and actors are involved in such interactions: within and between public organizations (notably for pooling or joint procurement initiatives), with intermediaries such as IT consultants, and communities as such. Strategies and economic models of collaboration varies also widely: public organizations can be in position of use of an existing OSS software (e.g. Linux, Apache, Dokeos) or stimulate its development (e.g. Aplaws), or even support specific forks or specifications (Spip-Agora, CommunesPlone, etc.). The creation of specific markets for OSS consultants and the organisation of their collaboration and sharing of information towards the development of a common –and public– good, is an additional issue affecting the sustainability of such projects. Numerous questions are thus raised: What are the current types of relationships existing between PA and OSS communities? On what basis can successful interactions be attained between PA and the communities? / What are the possible obstacles for cooperation? Are consultant' strategies and interests compatible with OSS development ? More generally, on what economic models is the use of OSS based?

The methodology used

To answer these questions, we favoured an empirical approach by case study, starting from earlier empirical material, previously collected but not yet utilized, and bringing them up to date with new surveys.

An empirical research based on project ethnography

Analyses of virtual groups are rarely based on insightful and longitudinal research conducted in an ethnographic perspective. In a general way, investigation at a distance is widely preferred to in situ research, and researchers would rather use documents generated by collective action, than having to negotiate positions and roles allowing a prolonged immersion in the heart of this collective action (Wellman & Gulia, 1999). This research strategy is facilitated by the abundance of information produced by the groups organized on the Internet (O'Mahony, 2003; Lee & Cole, 2003; Spaeth & al., 2008). In fact, these "virtual" groups leave many material traces of their activity, which appear as ways to have a grasp of this activity: web sites of the projects acting as vectors of information and communication, discussion lists and instant messaging services used as exchange and co-ordination media, forums or blogs listing the points of view and opinions of the participants. Furthermore, the researcher is often overwhelmed and fascinated by the richness of the material accessible, so much that his priority is to make drastic selections in order to be able to use it (Lazaro,

2008). These sources must not be neglected, as they constitute ingredients for the development of a virtual ethnography (Hine, 2000) and for experimentation with methods mobilizing new technologies (Hine, 2005 ; Murthy, 2008; Garcia & al., 2009). However, these sources alone are not enough to analyze the functioning of groups like free software development collectives (Pudelko & alii, 2006). The main reason for this is that the digital showcase of these groups only supplies an incomplete point of view on the activities that are performed there, on the exchanges between members, on the behaviour of the participants, on the regulations supporting the projects.

Here, instead, we based our analysis on the ethnographic study of software development projects with varied configurations, but who are all characterized by their intermediary status: they are beyond the stage of emergence and confidentiality because they have been able to federate several significant contributors and generate long-term practise by users, without, however, reaching a size that would allow the formalization of functioning rules and to contemplate a certain level of sustainability. Because of their intermediate size these projects are exposed to dual requirements of attractiveness and co-ordination. On one hand, new members must be attracted in permanence, on the other, co-ordination must be reinforced between participants. To reconcile these two objectives, the functioning must be both supple enough in order not to discourage participation and strong enough not to harm the coherence of the software produced (Demazière & al., 2007a, b).

The monographs collected are based on in-depth surveys. According to the cases, we involved a multiplicity of methods, combining the collection and the use of traces left on the Internet, with on-site *in vivo* inquiry based on a progressive insertion in the group. At the beginning of the study, we were able to use research based on interviews and recorded digital exchanges (notably on IRC) previously collected regarding four cases, but only partially analyzed. Thanks to this project, we were able to complete this first corpus by new empirical data, often collected from cases we already knew, in order to understand the structuring processes of collectives over time temps. The objective was to analyze, since the start of the project, the work organization modes, co-operation mechanisms, exchange forms and regulation processes, which support the functioning of the community. At the end, the analysis has been based mainly on the three projects explored among the five selected. This choice can be explained by the specificities of the ethnographic methods : as Ragin puts it (Ragin, 1992), despite any rational selection of cases, it's rather at the end of the ethnographic exploration that the researcher is able to assess the richness and interest of a case, all the more when conducting it in a comparative way.

Empiric base

Our research approach is based on the collection of case studies of projects evidencing either central or more peripheral aspects of the work dynamics involved when public organizations – of very diverse nature: administration, education organizations, public services, etc. – want to collaborate in the development of OSS. The analysis and the collection of empirical data originating from cases were conducted in a specific way during the years 2008 and 2009 of the project. Five cases of projects were investigated. Their choice had been determined as a function of an equilibrium between the opportunity of empirical access to the fields – implying a certain proximity – and differentiation of the cases.

1. Spip and Spip-Agora

Spip, an Internet publishing software is an interesting project to analyze for several reasons. First of all, it corresponds to a typical case of development based on voluntary commitments and in a totally distributed environment. Medium-sized, (a few dozen active developers), it is also characterized by the fact that it was initially supported by – and designed for – the French left-wing associative world, thus with a strong underlying political

and ideological orientation. In this context, the objective of the project is in tension with the necessary openness of the community to newcomers with more heterogeneous motivations, and notably web-developers working for web-agencies. In 2004, the French Government's Information Service (SIG) announced a new version of the software developed by the administration (project called Spip-Agora), the code of which would be transferred to the "Spip community". But this attempt to mix code and to sustain collaboration between project managers from the administration, private consultants hired for code development, and the initial Spip community, proved to be unsuccessful. Therefore the Spip-Agora became a fork, but in 2008 was finally abandoned.

2. CommunesPlone/Gov

This CMS represent the situation where a small group of developers coming from local authorities (Communes) decide to develop specific tools for local administrations. The project is based on Plone, but actors decide to adopt the standards, methods and culture of the Plone community, and step by step, the project is succeeding in federating various public actors at local, regional, federal and international levels. This case thus represents a situation where developers from various public organizations set up a relatively autonomous community, with public sector leadership, but which however necessitates coping with bureaucratic logics based on control and formalization, and with commercial logics brought in by IT service providers.

3. Claroline/Dokeos

The Claroline e-learning software was initially developed at the University of Louvain, Belgium, by a scientist who then established his own company to propose services around the software. This change of position led to the development of a new version named Dokeos, and by a fork of the community of other developers and users, with a part of them deciding to follow him, another part, staying with Claroline. The interest with this case is to show the potential difficulty to establish an economic activity (market-driven logic, customer oriented) on top of a project initially supported by a community composed of public education contributors, whose project is to develop an alternative to proprietary e-learning software (public logic oriented towards the making of a generic software).

4. PMB

PMB is a ILS – Integrated Library Management Software. At the origin, the software was created and developed by a librarian working at a small local library, who wanted a free software solution for the management of his library. He was joined afterwards by other contributors who will finally decided to establish a company for selling services around the use of the software. The founder then left the project, which is now maintained by PMB Services. Customers of the company are mainly big corporations and large-scale French libraries. However a community of very active users, coming on majority from local French-speaking Belgian libraries, is proposing support, translations and communication. Those contributors have on majority chosen to use PMB software, even if the administration does not support it and prefers to push its own build-in solution.

5. Aplaws

Aplaws is a CMS that was developed initially in England in the frame of an e-government plan developed by the Blair administration in early 2000. Like CommunesPlone, it is aimed at serving the interests of local administration, and developing e-gov tools. The business model is however based on joint-procurement and networking private consultants in the making of the software. This project ended in 2009, but is trying to restart through a new international consortium of public organizations.

Summary of empirical data collected (interviews)

Case	Empirical base (interviews)
------	-----------------------------

Spip/Agora	(24) + 14 = 38
CommunesPlone/Gov	(10) + 29 = 39
Claroline/Dokeos	(7) + 6 = 13
Aplaws	(4) + 3 = 7
PMB	6

The results obtained

The three case studies selected for the analysis have all revealed empirical elements suggesting difficulties in terms of articulation of the three logics – community-based, public and private– co-existing in the projects.

The Spip/Spip-Agora case illustrates a situation of failure involving the development of integration between several work environments that were unable to find common grounds to confer a shared meaning to collaboration. Since software is free, the administration was able to use the existing code and modify it according to its own needs. But the real interest of such strategy is not to develop a new autonomous version of the software, because users lose all the advantages of being certain that the efforts for the maintenance of the code produced will last over time, as well as the evolution of this code as a function of the improvements brought to other software components. This is actually the trap that Spip-Agora fell prey to, because the “code release” approach had failed. The “Spip-Agora” community had become de facto promoted by service providers in charge of developing the Internet sites of the various ministries, and encouraged to “play the community game” in order to support the common development of the software. This second approach also proved to be a fiasco. The case clearly shows that the coordination failure was not the result of technical constraints, lack of resources, or outside events, but certainly by a lack of trust and co-ordination between the partners who had placed very different ambitions in the Spip project than in the Spip-Agora project.

The same situation prevails in the Claroline/Dokeos case and in the PloneGov case. In the first case, the disagreements within the developers’ group were generated by the progressive and increasingly obvious divergence between several motivations and ways of being involved in the development of the software. Here too, the grounds for this disagreement were not generated by exogenous forces. No loss of users, no lack of interest, no obsolescence, no loss of involvement in terms of development force or financing: the objective causes of this failure originate exclusively from a disagreement between the parties regarding the forms of participation and animation of the project. The fork situation leading to a new project can be explained by a divergence in terms of contributors’ implication logics. And, in the case of PloneGov, while it does not present a *fork* situation as the others, the creation and upkeep of a group identity depends on a series of difficulties, surmounted each time at the cost of a reconfiguration of the collective action, including notably a differentiation of the logics (public, business-oriented) that were initially entwined.

Heterogeneity of the implication modes

A first analysis conducted on the cases focused on the characterization of the different principles that are structuring actor implication. In a first phase, using the ideal-type approach, we tried to characterize the “worlds” (Salais & Storper, 1993) on which the implication in OSS development was based, by differentiating the worlds of “techno-activist” mobilization, “public action”, and “business-oriented rationality”, and by situating several implication logics present in these worlds.

In the first world (“techno-activist”), OSS symbolizes the stake of the collective appropriation

of Internet relationship development potentialities, the distance taken from commercial constraints and technological dependency on the proprietary model, and of creative expression freed from deadlines and efficiency constraints. It is at the base of two logics “politico-ideological activist action” in which the project becomes a citizen-driven *cause* to support, and a “technological fun attitude” in which the project is seen as an unrestricted exercise area for technical skills, at the service of innovation.

The second world evidenced, the one focused on “public action”, is articulated around the prevalence of rule and hierarchy-based organization along with the management of common resources. Implication follows the same decisional scheme as any other IT project (choice of a provider after a call for tender) or administrative justifications. Two implication logics are identified: one is organized around the requirement of rational management of public resources and public infrastructure control; the other is focused on the political role of public organizations in the production of public resources.

Finally, the world of “business rationality”, includes the contributors who present themselves as “professionals”, motivated by service profitability, and looking for direct or indirect profits from the development of OSS. The implication of contributors involved in this logic is evaluated through reasoning in terms of profitability and client service and satisfaction, following however two logics. The first considers the interest of OSS as an extension of an existing commercial offer, also including proprietary solutions. The second considers free software as a new business model, and intend to act as pure players.

This analysis evidences different implication argument logics developed specifically according to the three initial worlds identified, but which can also be differentiated, on the horizontal axis by a dual matrix. The first mode is based on the justification of the implication through arguments linked to the *use of the product*, while the second mode relies on justifications based on the *society-related meaning* of implication in free software.

Argument logics for implication according to the different worlds

	“Techno-activist” mobilization	Public action	Business-oriented rationality
<i>Use of the product</i>	Technological challenges	Rationalization of public spending	New commercial opportunities
<i>Society-related meaning</i>	New militant terrain	Production of public resources	Driving force of innovative ecosystem

Through this analysis matrix, we can then re-read the cases and evidence the way in which requirements and characteristics of each “world” have an impact on the possibility to develop collective action, and the fact that articulation still constitutes a problem. Not only are projects characterized by structuring initial identities that operate as interpretative modes of more-or-less legitimate implication logics but, in turn, these same implication logics are not immutable rationalities which would involve stabilized and committed involvement, but instead they are bound to evolve, in parallel or prior to the evolutions of roles and positions in the projects.

Project regulation

However, although the tensions generated by the confrontation of these heterogeneous worlds constitute a structural component of these projects, and while it appears illusory to identify stabilized organizational forms, these collectives are nevertheless regulated. To analyze the regulation activity of the collectives, we will use the proposed analysis framework

developed by Ahrne & Brunsson (2010) regarding what they call “partial organizations”. The authors consider the organizations as *decided orders*, which included one or more elements among the five constitutive dimensions: entry selection, development of a hierarchy, implementation of a system of rules, elaboration of control and steering tools, and definition of sanctions. In contrast with the “complete” organizations, partial organizations are structured around only some of these five dimensions, especially by alternating between several sources of mechanisms, notably involving market or network mechanisms. In the case studies:

- in contrast with the “complete organizations” in which entry generates the creation of a member identity different from the non-members, this difference here is very vague: most of the contributors are situated in zones in which the question of group membership is vague. This is essentially due to the absence of formal contract defining the framework of mutual expectations.
- decision processes are not based on prerogatives linked positions of authority: the legitimacy of the decision remains indexed in time on the value of individual contributions. Therefore, the implementation of decisions is based on rallying, adhesion, incitation and leadership processes.
- functional rules are based not so much on explicit formalization, but rather on the predominance of diffuse norms. Again, this characteristic is related to the absence of a contract system, and to the importance of equilibrium between flexible rules in order not to discourage implication while remaining sufficiently operant to organize the group.
- steering systems for contributor’s activity are inexistent from a formal point of view, however, steering is based on other mechanisms involving two elements at the source of the implication: the fact that the project is more or less close to free software ideology and the more or less predominant character of the influence of the project contributors’ membership organizations.
- sanction mechanisms do not include the exclusion of members, or the withdrawal of the benefit of certain resources : the sanction here is based on the differentiation of status within the projects.

These five mechanisms do not appear in the same way in the three projects: Dokeos and PloneGov are still in a development process of the collective, with occasional moments of crisis in which problems crystallize. In contrast, Spip presents a configuration in which the five organizational mechanisms seem to have taken a more stabilized and regular form. Three factors can explain those divergences:

- the maturity of the projects, understood as the collective learning process allowing to successfully overcome the difficulties generated by growth,
- the structural importance of a common reference which can operate as a powerful regulatory principle. This reference corresponds to the power of the project identity and to its index on the way the group appropriates the political stake of free software.
- the impact of the institutional anchoring of the participants, or to the contrary, of the autonomy of the project from the environment.

In the light of this analysis, we can discard the hypothesis according to which these communities would be structured according to specific modes, which can be interpreted as imperfect or incomplete variants of the standard organizational model, which would lead to consider them as semi-organizations, without true specificity. To the contrary, the cases show that the functioning of these collectives takes the form of continuous, iterative and processual structuring, and that it must be considered as the combination of incremental regulations, partially stabilized and widely informal. This organizational configuration is based on the co-existence of two joint regulation mechanisms: the formalization of norms and their mobilization as control tools; the diffuse socialisation and its influence distributed through the interactions. And the balance maintained between these two regulation forms is necessarily fragile, unstable, and therefore they are likely to evolve.

In effect, in a context of remote relationships between contributors who do not necessarily share norms that would usually be acquired by an intellectual community in which prior common experiences, socialization of the members can only be limited. This implies that regulations are actualized necessarily through adjustments, confrontations, and compromises made throughout the interactions. So, even the formal rules are applied in a flexible way and negotiable way, as a function of the context of the transgressions. The articulation of those two regulation forms is thus not just a question of the amount of one or the other, but reveals a close interdependency: socialization is not sufficient enough for the conducts to adjust and co-ordinate automatically and formal regulation is insufficient to ensure that norms are systematically applied. The two types of regulations, formal and informal, thus go through continuous negotiation activities between participants. This constitutes a fundamental characteristic of the singular organizational form supporting free software development communities, which we propose to theorize under the term of “alter-organization”.

Therefore, analysing the collaboration between public organizations and OSS communities necessitates to depict the rationality of each stakeholder in presence (including IT providers selling OSS services), and understand the mechanism through which communities are established. It is this lack of understanding that can explain the failure of the Spip-Agora merge with Spip, as well as the current fork within the Dokeos project, while the CommunesPlone project shows a collective action that has been able to cope with various logics through a constant reconfiguration of the regulation mechanisms. But this analysis shows also how different the “complete” use of OSS is compared to traditional software procurement, and that the success of OSS migration depends greatly on new perspective in project and innovation management in public organizations, that cross the usual rules and norms of the public action.

*

* *