Revista de Administração de Empresas
ISSN: 0034-7590
ISSN: 2178-938X
Fundação Getulio Vargas, Escola de Administração de Empresas de S.Paulo

Ferraz, Isabela Neves; Santos, Carlos Denner dos

TRANSFORMATION OF FREE AND OPEN SOURCE SOFTWARE DEVELOPMENT PROJECTS: GOVERNANCE BETWEEN THE CATHEDRAL AND BAZAAR

Revista de Administração de Empresas, vol. 62, no. 1, e2020-0482, 2022
Fundação Getulio Vargas, Escola de Administração de Empresas de S.Paulo

DOI: https://doi.org/10.1590/S0034-759020220104

Available in: https://www.redalyc.org/articulo.oa?id=155170214006

ARTICLES

Submitted 06.05.2020. Approved 01.05.2021

Evaluated through a double-blind review process. Guest Scientific Editor: José Carlos Freitas Junior

Original version | DOI: http://dx.doi.org/10.1590/S0034-759020220104

TRANSFORMATION OF FREE AND OPEN SOURCE SOFTWARE DEVELOPMENT PROJECTS: GOVERNANCE BETWEEN THE CATHEDRAL AND BAZAAR

Transformação de projetos de desenvolvimento de software livre: Uma governança entre a catedral e o bazar

Transformación de proyectos de desarrollo de software libre: Una gobernanza entre a catedral y o bazar

Isabela Neves Ferraz[1] | isabelanf.adm@gmail.com | ORCID: 0000-0001-8536-8418
Carlos Denner dos Santos[1,2] | carlosdenner@unb.br | ORCID: 0000-0002-4481-0115

[1]Universidade de Brasília, Programa de Pós-Graduação em Administração, SOCIE-DADOS, Brasília, DF, Brazil
[2]Universite du Quebec a Montreal, LATECE, Montreal, QC, Canada

ABSTRACT

It is common for community-based free software projects to be associated with an organizational scenario that resembles "a bazaar more than a cathedral," and to differ from the traditional, or 'bureaucratic' way of organizing work. This paper analyzes the governance of these organizations from the perspective of their structure and control, considering the development trajectory of three community-based free software projects in Brazil. Results show that the constant need to produce modern technologies gives rise to external pressures that promote change - albeit temporary - in the governance of these projects, making them resemble a cathedral more than a bazaar. Governance does not follow a cycle of sequential improvement; it changes depending on the external organizational actors present, such as sponsors. This suggests the need for strategic and flexible governance to deal with the acquisition and allocation of organizational resources. Governance of the projects described here varies along a spectrum of (in)formality that allows both production models - cathedral or bazaar - to exist in the same organization at different periods.

KEYWORDS | Community, free software projects, governance, structure, control.

RESUMO

Comumente, projetos de software livre comunitários são associados a um cenário organizacional que se assemelha mais "a um bazar do que a uma catedral", diferenciando-se da maneira tradicional ou "burocrática" de organizar o trabalho. Este artigo analisa a governança dessas organizações, pela perspectiva da estrutura e do controle, considerando a trajetória de desenvolvimento de três projetos de software livre comunitários brasileiros. Os resultados mostram que a constante necessidade de produzir tecnologias modernas gera pressões externas que promovem mudanças, especialmente temporárias, na governança desses projetos, fazendo-os parecer mais uma catedral do que um bazar. Além disso, a governança não segue um ciclo sequencial de aprimoramento, modificando-se na presença de atores organizacionais externos como patrocinadores. Isso sugere a necessidade de uma governança estratégica e flexível para lidar com a aquisição e alocação de recursos organizacionais. Teoricamente falando, a governança dos projetos aqui descritos varia ao longo de um espectro de (in)formalidade que permite que ambos os modelos de produção – catedral ou bazar – existam na mesma organização, em momentos distintos.

PALAVRAS-CHAVE */ Comunidade, projetos de software livre, governança, estrutura, controle.*

RESUMEN

Es común que los proyectos de software libre comunitario se asocien con un escenario organizacional que se asemeja "a un bazar más que a una catedral" y se diferencien de la forma tradicional o "burocrática" de organizar el trabajo. Este artículo analiza la gobernanza de estas organizaciones, desde la perspectiva de la estructura y el control, considerando la trayectoria de desarrollo de tres proyectos de software libre comunitario brasileños. Los resultados muestran que la necesidad constante de producir tecnología moderna genera presiones externas que promueven modificaciones, aunque temporales, en la gobernanza de estos proyectos, haciéndolos parecer más una catedral que un bazar. Además, la gobernanza no sigue un ciclo de mejora secuencial, cambia dependiendo de los actores organizacionales externos presentes, como los patrocinadores. Esto sugiere la necesidad de una gobernanza estratégica y flexible para hacer frente a la adquisición y asignación de recursos organizacionales. La gobernanza de los proyectos descritos aquí varía en un espectro de (in)formalidad que permite que ambos modelos de producción - bazar o catedral - existan en el mismo proyecto, en diferentes momentos.

PALABRAS CLAVE */ Comunidad, proyectos de software libre, gobernanza, estructura, control.*

INTRODUCTION

Advances in information and communication technology (ICT) and its diffusion, combined with increasing environmental complexity (Seidel & Stewart, 2011; Zebari, Zeebaree, Jacksi, & Shukur, 2019), have facilitated the emergence of modern and flexible organizations, which present themselves in a multitude of 'configurations' known as virtual communities (Arazy, Daxenberger, Lifshitz-Assaf, Nov, & Gurevych, 2016; Faraj, Jarvenpaa, & Majchrzak, 2011). A widespread example of a venture that is based on a virtual community format is free software projects (Li-Ying & Salomo, 2013; Seidel & Stewart, 2011). These communities are represented by projects that are created on the Internet, in which the program's source code is made publicly available to receive the external contributions of users and developers through the significant participation of volunteers (Eseryel, Wei, & Crowston, 2020; Santos, 2010; Seidel & Stewart, 2011; Shah, 2006).

It is common knowledge that virtual communities, such as free software projects, are organized in a manner that is different from traditional organizations (Arazy et al., 2016; Lee & Cole, 2003; Miscione, Ziolkowsk, Zavolokina, & Schwabe, 2018; Seidel & Stewart, 2011). In the metaphor used by Raymond (1999, 2005) with regard to the community model of production, free software projects are seen as bazaars, with their informal way of working, while traditional organizations are perceived as cathedrals, which function by way of a formalized structure and control processes.

Studies of free software need to detail the specificity of the operations (Bauer, Franke, & Tuertscher, 2016) in order to understand how, in the absence of formal means, or the minimal presence of the same, communities conduct their operations and achieve results (Arazy et al., 2016; Eseryel et al., 2020; Miscione et al., 2018; Panchal, 2010). A pivotal concept for verifying how work is performed in organizations is governance, which is multidimensional and involves formal and informal means for directing the actions of individuals (Li-Ying & Salomo, 2013; Markus, 2007).

It is clear from the literature on free software that although it is usual to differentiate between the characteristics of communities and traditional organizations (Arazy et al., 2016; Lee & Cole, 2003; Seidel & Stewart, 2011), authors do not always consider the changes in projects and their development in this differentiation. As a result, mature, free software projects may be regarded as businesses that pursue an informal management logic instead of adopting a traditional organizational model (Demil & Lecocq, 2006; Markus, 2007; Raymond, 2005). This can be noted in studies such as those by Lee and Cole (2003), Shah (2006), Seidel and Stewart (2011), and Lindberg, Berente, Gaskin, and Lyytinen (2016). They cite successful projects, but without analyzing the changes that occurred in them over time. While works by Lattemann and Stieglitz (2005), Xu, Xu, and Lin (2005), and Panchal (2010) acknowledge the existence of different stages of development in free software initiatives, these authors do not conduct any empirical examinations, or reach any practical conclusions.

Given these considerations, a greater understanding of how free software projects conduct actions during their development is necessary (Forte & Lampe, 2013; Guimarães, Korn, Shin, & Eisner, 2013). This research aims to answer the following question: *In terms of structure and control, how does governance appear and transform itself in developing community-based free software projects?* It also analyzes these projects from the perspective of organizational theory and approaches that involve dependence on resources, stakeholders and temporary organizations. This qualitative study uses the case study strategy to focus on three community-based free software projects. The idea is to bring a dynamic view that enables researchers and professionals to understand the main drivers of change in governance during the development of free software businesses. This will enable verification of whether communities (bazaars) differ from traditional organizations (cathedrals) or are similar to them.

LITERATURE REVIEW

Governance in community-based free software projects

Free software communities retain all the essential features of a virtual community, where geographically dispersed members, who are mainly volunteers, work and collaborate in flexible ICT-mediated virtual spaces (Eseryel et al., 2020; Faraj et al., 2011; Seidel & Stewart, 2011; Shah, 2006). These communities differ from those that produce software commercially, because they are structures that promote the free use of software and make its source code freely available on the Internet (Miscione et al., 2018; Santos, Kuk, Kon, & Pearson, 2013; Seidel & Stewart, 2011). Such free software communities have been active in producing competitive and innovative products (Bauer et al., 2016; Viseur & Charleux, 2019).

According to Markus (2007), governance in the context of free software communities is represented by the formal and informal means that enable individuals, who are wholly or partially autonomous, to contribute jointly to a software development project. Despite the importance of governance in free software enterprises, studying it is still a controversial topic (Li-Ying & Salomo, 2013), mainly because a significant portion of research considers such communities to be arrangements that are marked by informal management, which differentiates them from the hierarchical and formal logic of traditional organizations (Eseryel et al., 2020; Seidel & Stewart, 2011). While recognizing such characteristics, however, other studies have argued that the governance mechanisms used by free software communities are quite flexible, so they can be modified over time (Lattemann & Stieglitz, 2005; Li-Ying & Salomo, 2013; Mäenpää, Munezero, Fagerholm, & Mikkonen, 2017; Viseur & Charleux, 2019; Xu et al., 2005).

This particular research began by assuming an understanding of governance to mean something that is carried out dynamically considering the development trajectory of community-based free software projects. Among the multiple prisms by which governance can be studied, two are analyzed in this study: structure (Lindberg et al., 2016; O'Mahony & Ferraro, 2007; Seidel & Stewart, 2011) and control processes (Lattemann & Stieglitz, 2005; Xu et al., 2005).

Structure and the control of community-based free software projects

From a structure perspective, free software communities differ from traditional organizations because of the absence of a central hierarchy and authority (Lee & Cole, 2003; Miscione et al., 2018; Panchal, 2010). These communities adopt a perspective in which power is established horizontally because individuals do not move vertically, but from the periphery towards the center of the project. Participants at the center, therefore, are regarded by others to be project connoisseurs and the most trusted decision makers (Dahlander & O'Mahony, 2011).

Roles in free software communities are not specialized, since exercising them is not previously designated, and occurs during the collaborative work process itself (Arazy et al., 2016; Lee & Cole, 2003). Overlaps in performing tasks are expected, and explain the contributors' participation in several functions within the project (Lee & Cole, 2003; Seidel & Stewart, 2011). Even though there are no deliberately specified roles in projects, the exercise of leadership is often well defined (Lee & Cole, 2003). Leadership is seen as a condition for a project's success as the leader assumes responsibility for the various subsystems and makes critical decisions regarding source codes and business directions (Forte & Lampe, 2013; Guimarães et al., 2013; Mäenpää et al., 2017).

With regard to the decision-making process, the literature on free software communities lists different approaches. Paradoxically, while assuming that the most impactful decisions focus on the collective's authority

(Lee & Cole, 2003; Seidel & Stewart, 2011; West & O'Mahony, 2005), it is suggested that community members have greater independence when carrying out their activities. This is not only because of their physical distance from each other, but also because communities are non-hierarchical groups mainly comprising volunteers (Panchal, 2010; Seidel & Stewart, 2011).

From the control perspective, although formal result- and behavior-oriented means can are found in free software communities (O'Mahony & Ferraro, 2007), the most common understanding is that it is the informal means, such as self-control, and social and peer control, that stand out (Jensen & Scacchi, 2010; Kolbjørnsrud, 2016; Lattemann & Stieglitz, 2005; Xu et al., 2005). Social control in free software projects seeks to strengthen the sense of belonging to the community that is based on intensive socialization and cultural assumptions. This control involves sharing the objectives and values that aim to achieve the behavioral orientation of individuals (Lattemann & Stieglitz, 2005; Xu et al., 2005).

With regard to self-control, those technology platforms that support community practices and allow individual contributions with high levels of collaboration emphasize autonomy and independence in doing the work, thus reducing coordination efforts (Crowston, Wei, Howison, & Wiggins, 2012; Lindberg et al., 2016). Project leadership plays an essential role in establishing self-control, especially when selecting members and the work to be done (Xu et al., 2005). Peer control is an alternative for guaranteeing the quality of the applications that are produced by free software communities (Kolbjørnsrud, 2016; Lattemann & Stieglitz, 2005). Its purpose is to enable other people to review the work done by an individual, which give rise to criticism of the work of others that is used for correcting mistakes, solving problems, and promoting improvements and an exchange of knowledge (Lee & Cole, 2003).

Structure and control transformations

Even though it is possible to describe the attributes of community enterprises in their organic form (Arazy et al., 2016; Seidel & Stewart, 2011), the fact that free software initiatives undergo successive transformations justifies contextualizing them in relation to the development they achieve (Faraj et al., 2011; Mäenpää et al., 2017). Authors such as Wynn (2003), Lattemann and Stieglitz (2005) and Xu et al. (2005) pointed out that such initiatives undergo successive phases of development (introduction, growth, maturity, and decline, or revival) that affect governance.

Although some authors, such as Raymond (1999, 2005), have claimed that the work of free software projects is similar to the logic of a "bazaar", in which participants are volunteers who are organized in a non-hierarchical and informal way (as opposed to the cathedral view of the organization, i.e., hierarchical and formal), such ventures are dynamic and may change over time. It is believed that the structure and control processes may arise from a more informal context and incorporate formal elements as these projects develop (Forte & Lampe, 2013; O'Mahony & Ferraro, 2007; Panchal, 2010).

A factor that may contribute to this formalization and affect governance has to do with the presence of other organizational actors in community activities, such as sponsors and foundations (Mäenpää et al., 2017; Viseur & Charleux, 2019; West & O'Mahony, 2005). This rather formal functioning of community-based free software initiatives, however, needs to be better understood, and it has led researchers to question some of the accepted ideas about how these communities work and organize their activities (Shaikh & Vaast, 2016).

Theoretical and conceptual organizational contribution

The stakeholder approach

According to Freeman (2010), stakeholders are individuals or groups of individuals who can influence the specific performance of organizations, including their management procedures and objectives. Since the concept of stakeholders was first presented in 1984 in Strategic Management: The Stakeholder Approach by the above author, it has been widely used in organizational studies. This approach has different definitions and forms of evaluative measurement.

Although the stakeholder approach has been widely used in studies involving private corporations, it is being increasingly used in analyses of public and nonprofit organizations, mainly because of concerns with regard to allocating resources and serving groups more effectively (Bryson, 2018). It is also common for the stakeholder approach to be used in conjunction with other organizational theories (Frooman, 1999).

Regardless of the context or the theory allocated to the stakeholder approach, the organization must be understood as a cluster of actors who focus on common problems that involve managing interests and points of view that are not always convergent (Friedman & Miles, 2006). This should be perceived as a two-way street, as stakeholders and organizations suffer reciprocal influences. This approach also deals with managing contradictions and conflicts that arise between individuals and/or groups, and seeks to solve them (Friedman & Miles, 2006; Frooman, 1999).

Resource dependence theory

Pfeffer and Salancik's 1978 seminal work, *The external control of organizational: a resource dependence*, consolidated the resource dependence theory (RDT). The RDT considers that external factors influence the behavior of organizations (Pfeffer & Salancik, 2003). But while it recognizes the interrelation between the organization and the context, the RDT produces a more proactive conception of the venture, as managers, employing various tactics, can reduce uncertainties, risks and dependencies (Hillman, Withers, & Collins, 2009).

Given the interdependence between organizations and the environments in which they operate, businesses can reconfigure their internal structure to meet current demands and use various strategies to secure the resources they need to survive and be competitive (Sharif & Yeoh, 2014). This role of providing and ensuring relevant resources usually rests with the person in charge of managing the organization (Sharif & Yeoh, 2014).

The wide range of formal or informal actions carried out by managers has the primary purpose of securing essential means, increasing the power in exchange relations, and reducing subordination and the uncertainties that enterprises face (Pfeffer & Salancik, 2003). According to Pfeffer (1988), one of the primary arguments of the RDT is that organizations are not autonomous. There is interdependency with other organizations, which means that the uncertain and survival-oriented actions of companies need to be managed. In short, the three ideas central to the RDT are that: social context is relevant; organizations need strategies in order to seek greater autonomy and to serve their own interests; and power is essential for understanding the internal and external actions of organizations (Davis & Cobb, 2010).

Temporary organizations

Temporary organizations (TOs) are represented by groups of interdependent actors formed to accomplish a complex task (Burke & Morley, 2016). Definitions of the concept converge when they state that TOs have a termination point that is fixed at a specific date, or they achieve a predefined state or condition (Bakker, 2011; Bakker, Cambré, & Provan, 2009).

TOs are vital-task-oriented (Bakker, 2011), especially those characterized by the complexity arising from their interdependencies, the ambiguity of their interests and the uncertainty of the outcomes (Bakker, 2011; Burke & Morley, 2016). Because of these characteristics, TO teamwork needs to be coordinated, and the leader's role is regarded as crucial (Bakker, 2011). TOs are intrinsically related to the external environment, which is represented by the organizational context and the broader social environment, encompassing relationships with other organizations. The primary resources necessary for the survival of TOs are found in the external environment (Grabher, 2004). Such intertwining with context may become a dilemma for TOs, because while such structures seek autonomy, they also need to acquire the resources necessary for operating (Burke & Morley, 2016).

Traditionally, TOs were understood as complementary arrangements to permanent organizational structures. In a later framework, alternative configurations, such as interorganizational projects and project-based organizations, were included as representative of TOs (Bakker, 2011; Burke & Morley, 2016). This type of organization may take many forms that surpassing intra- and interorganizational collectives, also involving groups that cooperate and integrate resources and expertise to achieve results (Sydow, Lindkvist, & DeFillippi, 2004).

METHOD

This qualitative paper used a case study strategy to examine the free software sector in three main phases. The first phase involved choosing the projects to be studied. The researchers mapped out competitive Brazilian projects that were indicated by researchers and professionals working with free software, and investigated projects that had participated in events in Brazil. Selection criteria were subsequently applied. Based on the works of Xu et al. (2005), Latterman and Stieglitz (2005), Panchal (2010), and Viseur and Charleux (2019), the intention was to identify Brazilian projects with possible variations in their governance and development profiles. These projects varied in terms of the following characteristics: the year they first appeared; the number of software versions; the number of committers (developers with the ability to modify the source code repository); the license used; and the presence of foundations or sponsors. This process led to three projects being selected: CoGrOO, Noosfero and Kytos.

The second phase covered data collection. Interviews were scheduled by e-mail and conducted in person, using Skype or Google Hangouts, when a previously tested semi-structured script was applied. This script was tested on five Brazilian participants who manage free software projects.

The selection of respondents was intentional and included leaders, developers, and stakeholders involved in the core group of projects. The first respondents were key informants with knowledge of the project's history, and the others were selected using the snowball technique. The research did not use theoretical saturation to establish the number of interviews. The concern was to have the history of the projects recounted by individuals who knew their reality and occupied different roles, as defined by structural sampling (Shah & Gorbatai, 2015). Fourteen interviews were carried out. In the case of CoGrOO, the history of the project at a given moment in time

had an interface with the global Apache OpenNLP project. This situation required complementary data to be collected. Any information that was not obtained from interviewing the key informants, or from documents was supplemented by questions in e-mail form that were sent to participants of the Apache OpenNLP project. Five forms were sent out, and two were answered.

The document search was used to map possible free software projects for inclusion in the study and to obtain complementary data to those collected in the interviews. The contents of e-mail lists, websites, and various documents (slides, reports, manuals, and news articles) were, accessed and organized in a spreadsheet.

Table 1. **Selection criteria of projects**

Project	Year of appearance	No. of versions	No. of committers	License	Foundation / Sponsorship	No. of interviews
Kytos	2014	3	6	MIT	Sponsorship	5
CoGrOO / Apache OpenNLP	2003 / 2000	17 / 34	3 / 23	Apache License 2.0	Foundation	3 + 2 forms
Noosfero	2007	191	12	GNU AGPLv3	Sponsorship	6

In the third phase, the data collected during the interviews and from the document search were transcribed and treated using the content analysis technique (Bardin, 2010), supported by Atlas.ti software. The text was coded in accordance with indicators that refer to each of the categories of analysis previously established, based on a review of the literature. Although the analysis categories were derived from the literature, this did not prevent us from making discoveries during the field research.

Exhibit 1. **Categories of analysis defined from the literature**

Concepts		Dimensions		Indicators
Governance	**Structure**	Hierarchy		Line of authority, subordinate relationships.
		Role specialization		Distribution of roles and responsibilities, division of labor.
		Centralization		Decision focused on a particular individual or organizational level.
	Control	Formal (behavior and results)		Standardization, rules, regulations, rewards based on individual results.
		Informal	Social	Shared culture and practices, socialization of individuals.
			Peer Control	Mutual adjustment, critical of peer work.
			Self-control	Independence in carrying out the work.
Organizational theories / approaches		Resource dependency		Search for resources in the environment.
		Stakeholders		Relationships with internal and external actors.
		Temporary organization		Task orientation, specific completion point.

ANALYSIS OF RESULTS

It is important to note that the CoGrOO project differs from other projects because its source code is coupled with the OpenNLP project, which is currently part of the Apache Foundation. This second project, therefore, was also considered in the analysis.

Case history

CoGrOO

The CoGrOO project is a grammar checker that can be coupled to OpenOffice.org. It assists in text writing by verifying whether the grammar is correct using natural language processing (NLP). The CoGrOO project started in 2003, when a university professor, who at the time worked at the National Business Learning Service (*Senac-SP*), noted the need to create a grammar checker for OpenOffice.org. Together with other faculty colleagues at the University of São Paulo (USP), this professor designed a project to develop software for this purpose, called CoGrOO.

Between 2004 and 2006, CoGrOO received funding from the Free Software grant facility of *Finep* (Studies and Project Funding Agency). The financial support provided by *Finep* made it possible to offer internship and undergraduate scholarships for developing the software, which was fundamental to the preparation of the initial versions of the grammar checker. When funding came to an end, there was a period in which CoGrOO kept working by way of volunteers and institutional support from the University of São Paulo (USP), which involved an interface with the university's research work and its academic courses. Some of the facts that occurred during this period are worth mentioning: grants were obtained from the Google *Summer of Code* program; the language code was supplied by Google; the CoGrOO community portal was developed on a graduate course at USP; and the software was incorporated by the FLOSS Competence Center at USP (CCSL/USP).

Even though CoGrOO has become a national benchmark, the software community is facing a challenge to stay active because of the loss of resources needed for sustaining the project, and the constant withdrawal of volunteers. One of the members who joined the community as an intern, however, and later volunteered was interested in continuing the project. Around 2011 this developer noted that another project in which he participated, the OpenNLP, had important interfaces with CoGrOO, even though it was not intended for end-users. As a result and on his initiative, about 90% of the CoGrOO code has been incorporated into OpenNLP by way of a patching process. By transferring the code, and because since 2011 OpenNLP has been an Apache project, the management of this code now has that foundation's infrastructure also, which continues today.

Noosfero

The Noosfero project is a web platform for social and solidarity economy networks. In addition to social networking's standard features, the project offers a content management system (CMS). The software originated in 2017 and was developed by the Cooperative of Free Technologies (*Colivre*), which aimed to meet the similar demands of the cooperative's two customers.

Although it arose in the context of commercial demand and was free software, it attracted the attention of external actors, which led to other organizations carrying out projects that used the software to develop their own websites. The existence of parallel projects has brought a specific dynamic for improving the Noosfero community.

In this sense, while the projects were carried out to meet a given organization's interests, free software supporters who voluntarily participated in these projects were concerned with returning the code that was developed to the Noosfero community. Among the institutions that acted in these initiatives, the following stood out: the Association of Free Software Brazil (*ASL*), the FLOSS Competence Center (CCSL/USP), the Federal Data Processing Service (*Serpro*), the Ministry of Planning, Development and Management (*MP*), and the University of Brasilia (UnB). In some of these projects there were partnerships between the abovementioned institutions and members of *Colivre*.

Even with these parallel initiatives, since the beginning of Noosfero, software development has always been linked to *Colivre* sponsorship. To date, the cooperative's contracts have enabled funds to be invested in the project, which means that *Colivre* continues to play a leading role in managing the project, launching new versions of the software, and reconciling the actors' multiple interests.

Kytos

Kytos is a free platform developed by the Scientific Computing Nucleus (*NCC*) team from the Universidade Estadual Paulista (*Unesp*) in partnership with the São Paulo Research and Analysis Center (SPRACE). This project aims to facilitate the development and implementation of software-defined networks (SDN). The *NCC* plays a significant role in instrumentation projects, hardware and computer networks. It has both Brazilian and international partnerships in the area, particularly with the following actors, to name but a few: National Education and Research Network (*RNP*), Academic Network at São Paulo (ANSP), California Institute of Technology (Caltech), Conseil Européen pour la Recherche Nucléaire (CERN), and Americas Lightpaths (AmLight). These relationships enable *NCC* members to undertake joint projects and participate in various events in Brazil and abroad.

Kytos came about when one of the *NCC* members participating in an overseas event had a chance to assist a demonstration, transfer data using free SDN controllers, and identify a number of limitations in these controllers. In 2014, by way of joint work between the *NCC* team and developers from Caltech, this member began developing a free library called python-openflow. The creation of this library was the first step towards the beginning of the Kytos project.

One of the key events that drove Kytos's development happened at the end of 2015 when the company received sponsorship from Huawei. This Chinese company has invested financial and material resources in Kytos through the Informatics Law, which has even enabled interns to be hired to assist with the development of the software. The initially created library has become a separate repository and is currently a subproject covered by the SDN Kytos platform macro-project. To date, Kytos has continued to be supported by *NCC* and Huawei, as well as by volunteer work being done by partner organizations.

Structure

With regard to physical and technological infrastructure, all projects began at some point to function within an institutional context. The existence of these organizations around projects shows that institutional support is a fundamental factor for making the basic physical and technological infrastructure available. According to the interviewees, the projects use ICT resources intensely due to the opening up of the software by the community and the geographic dispersion of individuals (Bauer et al., 2016; Shah, 2006; Seidel & Stewart, 2011). ICT includes source code hosting and version control platforms for software, such as GitHub, GitLab and Sourceforge; chat and video platforms, in particular Google Hangouts; team collaboration tools such as Slack; mailing lists; and home

pages used to disseminate project information. It is important to underline that mailing lists have proved to be the most relevant means of communication in all projects.

Concerning the hierarchy and the members' roles, the overall information indicated that throughout their development, projects had to deal with both formal and informal aspects. With regard to hierarchy, the projects formalized those individuals who were designated to exercise collective management, and adopted, in parallel, the figure of a lateral authority that transcends the established formal setting. With regard to this last aspect, there is convergence with what authors such as Dahlander and O'Mahony (2011) reported about free software projects, when they pointed out that these collectives admit a lateral authority based on the experience and contributions to the project of individuals.

In line with this model of formal and informal hierarchy are the roles played in the projects. In the list of formal roles associated with collective management, the following deserve special mention: in CoGrOO, formal management responsibility was concentrated in professors at several moments in time; in Kytos, follow-up was performed by the *NCC* director and the project coordinator or leader (*Unesp* employee); in Noosfero and OpenNLP, a role that was common to both projects was the release manager (RM). But while the RM in Noosfero released versions of the software and managed the project, in OpenNLP this function focused specifically on the version release process. The formal roles in OpenNLP were those of chair, vice president, and Project Management Committee (PMC) members.

Teams performed formal or informal functions in developing software. Among those in charge of formal roles were project professionals and interns who were hired by the projects (notably by CoGrOO and Kytos). In CoGrOO, hiring took place when the project received funds from *Finep*. In Kytos, hiring has been a reality since the project was first sponsored by Huawei, whose sponsorship remains to this day. Informal roles focused on volunteer developers who engaged in collaborative project activities. Also with respect to the development team, a common role in all collectives is the committer, who is a volunteer or employed developer, who has code writing power in the main project repository. In Noosfero and OpenNLP, committers appear in formally constituted groups. In Kytos, although the roles are pre-defined, becoming a committer has to do with the perception of the coordinator, who invites someone to perform the role when they consider that the individual deserves it. In CoGrOO, this role is informal.

In general, formal and informal roles coexisted throughout the development of the software in the analyzed projects. While research into free software communities shows that volunteers collaborate significantly in community projects (Seidel & Stewart, 2011; Shah, 2006), the projects that were identified here showed that this is not always the case. Particularly in specific projects, like Kytos and CoGrOO, hiring professionals was the solution for keeping the project running at certain times. Hiring and formalizing certain roles has shown that, while the literature states that roles are poorly defined in free software communities (Arazy et al., 2016; Eseryel et al., 2020; Lee & Cole, 2003), this is not always the reality in the projects studied.

Similarities with research into free software were also verified. For example, regardless of whether roles are formal or not, in all projects respondents reported that, as required, individuals could perform functions other than those formally assigned to them. This coincides with what Lee and Cole (2003) and Arazy et al. (2016) advocated, when they reported that individuals can perform multiple overlapping roles in community spaces. Another common issue in all projects relates to the role of leadership, which has been paramount in governance for promoting unity and the alignment of behaviors and objectives (Guimarães et al., 2013; Mäenpää et al., 2017; Seidel & Stewart, 2011). In Kytos, the coordinator is primarily responsible for leadership, while in CoGrOO the professors and the

main software developer exercised this role after removing the other members. In Noosfero, the RM assumed the project management and leadership roles. Although the chair was a type of leader in OpenNLP, in fact this person stood out because they were more heavily involved in bureaucratic activities, which led to other members of the group who had a leadership profile assuming this role.

In relation to decision-making, although all projects were open to participants expressing their opinions, centralization was visible in the absence of consensus, or when the decisions involved strategic questions about the direction of the project. This perception converges with the paradox of free software studies, according to which, although such projects are decentralized, relevant decisions are often centralized (Lee & Cole, 2003; Seidel & Stewart, 2011; West & O'Mahony, 2005). In CoGrOO, centralization happened because of the professors who oversaw the project at specific times, and the responsibility of the main project developer, who in several phases of the project was responsible for this function. In OpenNLP, a formalized consensus process, known as a consensus meeting, was notorious. Project members vote on a particular issue by way of email lists, and the solution resulting from the final vote count is the one that prevails. Despite this search for consensus, the representatives with voting power on the most relevant issues are those in the PMC. If there is a decision impasse in Noosfero, the individual formally assigned to take the final decision is the RM. In its decision-making structure Kytos stated that the most strategic choices are made by the director of the *NCC* and the project coordinator, although members are free to express their opinions.

Noosfero and OpenNLP have proved to be the projects with the most formal regulations on how decisions should be made. In the other projects, decisions are made according to the occasion, and without any previously defined protocol. Centralization of the most impactful decisions on the project, however, has been mostly entrusted to one or more individuals in the collective, either because of a formally delineated role, or because of the influence of more experienced collaborators in the project.

Control

Formal processes for controlling members' behavior and collective results were observed in some activity flows in the projects investigated. In this regard, the following processes stand out: choosing representatives and defining responsibilities for exercising a certain role; decision-making, in particular strategic decisions that set the objectives of the project; regulations of the contributions received, especially those regarding changes in the source code of the software; the routines required for releasing versions of the software; and reporting for accountability or project monitoring.

The analyzed projects that stood out because they had well-defined and formally delimited processes were OpenNLP, Kytos, and Noosfero. OpenNLP, as with other Apache projects, was characterized by following a number of procedures that were detailed on the foundation's website. Apache also has a code of conduct to guide participants and requires individuals who become committers to sign a document confirming that they follow the project license terms. Kytos is similar to the OpenNLP project, with individuals willing to collaborate needing to be aware that they will be governed by the software license terms set out on the project's webpage. Kytos also stood out with regard to formalizing certain roles in addition to those of the *NCC* director and the coordinator, such as hiring professionals with funds provided by the Huawei sponsorship agreement. Both Kytos and OpenNLP adhere to the rule whereby project tracking reports have to be delivered to the sponsoring company and the Apache foundation, respectively.

Having experienced a phase of conflict, in which the volunteers at one point demanded greater openness for the project and independence from *Colivre*, Noosfero drafted a document that contained a series of formalizations. The idea behind this document was that the requested openness should maintain the quality of the software. The document, which was made available virtually, provided guidance on the exercise of roles and the ways in which contributions involving the software should occur.

While CoGrOO had no well-defined processes, which only occurred when there was an interface with OpenNLP, it experienced times when formalized actions were needed. These periods occurred especially when the development of the grammar checker software received funding from *Finep*, and when the software was used in courses and research that was being conducted within the scope of USP. At such stages, the various existing task forces had limited commitments and delivery schedules that were supervised by professors.

When projects are considered during their development phase, the formalization of behaviors and results is evident in various situations. Thus, while research into free software indicates that these initiatives stand out because of their informality and poor process structures (Eseryel et al., 2020; Kolbjørnsrud, 2016; Lattemann & Stieglitz, 2005; Lindberg et al., 2016; Xu et al., 2005), these projects showed that, depending on the moment, formal processes were also adopted by the collectives. This happened mainly at times when projects established relationships with other formal organizations (Mäenpää et al., 2017; Viseur & Charleux, 2019; West & O'Mahony, 2005). Thus, projects such as CoGrOO, in its interface with OpenNLP, Noosfero, and Kytos, had more formalized processes in place. These are collectives that are part of foundations or receive sponsorship, which requires the expected conduct and deliveries of members to be clearly defined.

Social aspects were reported by the interviewees as informal control procedures. In promoting the socialization of project members, the use of ICT resources enabled a common virtual space to be created for collaborative work (Eseryel et al., 2020; Seidel & Stewart, 2011). This communication allows heterogeneous members to direct their efforts towards the results. Although most communication takes place virtually, face-to-face meetings to discuss project issues were also reported. Noosfero held face-to-face meetings, with most members living in the same city, especially in the beginning when the project was centered on *Colivre*. CoGrOO's developers studying at USP indicated that they had several moments of face-to-face interaction when the software was being developed in the university. Since Kytos uses *Unesp's* infrastructure, this enables some of its members to meet in person, which often involves those who were formally hired for the project. Opportunities for face-to-face meetings in OpenNLP are rare, and take place mainly at the ApacheCon event, which is an annual conference sponsored by the foundation.

Participants also stressed situations in which they had the opportunity to interact with actors outside the projects. Among these situations, members' participation in technical or academic events in the free software area, or in some project activity were mentioned. Noosfero participants had the opportunity to give talks at meetings on free software. Kytos, CoGrOO and OpenNLP all emphasized events to present technical or academic papers on the projects, and Kytos, in particular, even promoted editions of its Kytos Summit, with talks for participants or individuals interested in the project.

Another perceived form of social control is associated with recognition of the work performed on projects, a practice that is associated with meritocracy. In this respect, experienced members who made relevant contributions and were recognized by their peers distinguished themselves by exerting significant influence on decisions, as highlighted by Lattemann and Stieglitz (2005) and O'Mahony and Ferraro (2007). Respondents also reported external opportunities for recognition. Situations such as reports, interviews, awards, and even moments when

significant resources could be captured were mentioned as being crucial, especially with regard to the project and its main representatives being recognized. This view of recognition as a form of social control converges with what is expected in community settings, as it is a form of motivation that differs by not using formal rewards or private benefits (Shaikh & Vaast, 2016).

These experiences, by way of internal or external interactions, were essential when it came to individuals creating a sense of belonging with the projects (Lattemann & Stieglitz, 2005; Xu et al., 2005), and even involved those who were remote or less active at the time. Having this reference and identity influences self-control, especially among participants who have made, or who still make contributions as volunteers. By allowing autonomy and independence in the execution of work, technological resources also provide support for self-control, thereby reducing coordination efforts in the collective (Crowston et al., 2012).

With regard to reviewing contributions made to the software, although the projects have rules requiring external collaboration to be reviewed by committers, when these contributions come from committers themselves, the analysis takes place informally by way of a peer control process. In these cases, while committers have writing power in the repository, a common practice has been to have the code reviewed by another committer in the community, especially a more experienced developer. Noosfero is the only project that has tried to create explicit rules in this regard, yet with some flexibility. If after one week the code has not been reviewed by any peer in the collective, the committer that proposed the code has the autonomy to add it to the repository. The projects investigated converged with free software research in the sense that criticism of others' work and the exchange of experiences that is made possible by reviews help ensure the quality of the software that is produced (Kolbjørnsrud, 2016; Lee & Cole, 2003).

Analysis based on the conceptual and theoretical organizational contribution

Besides the projects studied having in common the fact that they work in the context of a provider of physical and technological infrastructure, throughout their history they also had relationships with different organizations. Understanding that stakeholders are groups or people that affect or may be affected by an organization (Freeman, 2010; Frooman, 1999), those actors with whom projects have established relationships are stakeholders. The CoGrOO's main stakeholders are USP, *Finep*, Google and the Apache Foundation's OpenNLP project. In addition to *Colivre*, Noosfero had several stakeholders, such as USP, *Serpro*, UnB, the Federal Government, among others. Kytos's stakeholder were *Unesp*, Huawei, partner institutions and other communities.

This understanding, that the agents with whom projects interact constitute stakeholders, shows that that there is a mutual influence between these projects and their interest groups (Frooman, 1999). This logic of reciprocal influence was seen in Noosfero, when parallel organizations emerged that started executing projects using the software to develop websites in line with their interests. The individuals who participated in these projects and were sympathetic to the free software movement voluntarily returned the code they developed to the Noosfero community. Another example is the fact that *Colivre*, the main sponsor and executor of commercial contracts using Noosfero, invests in maintaining the project community. Stakeholders were also influential when *Colivre* had to manage conflicts with external actors, who demanded greater openness from the project community.

Since the interaction between the projects and the stakeholders is a reality in the analyzed collectives, context plays a crucial role in altering project behaviors (Pfeffer & Salancik, 2003). In this scenario, the resource dependency theory (RDT) can be applied to community-based free software projects. The dynamics perceived

during the development of the investigated projects revealed that the main driving force of the relationships with stakeholders is the need to obtain resources for business survival. Resources had to be sought to ensure that projects continued to function over the years, either via the infrastructure provided by an institution, or by support from a foundation, sponsor, and/or funding agency. An example of this is Kytos, which in addition to the support offered by *Unesp's NCC* via contacts made by the director of the *NCC*, obtained sponsorship from Huawei, which also made it possible for it to obtain the financial and technological resources needed for developing the software.

Considering the RDT, it is clear that projects are not only influenced by the environment, but also have a proactive stance towards it (Hillman et al., 2009). Project leaders have the autonomy to use various management tactics to provide the means they need to survive in the environment (Sharif & Yeoh, 2014). CoGrOO's constant search for resources took place at various times. In addition to USP's support and the funding provided by *Finep*, the organization had to look for ways to secure more funds during its trajectory, which included receiving funds from the Google *Summer of Code* program. Later, when one of its developers eventually became solely responsible for the project's operations, he found OpenNLP to be a way for the CoGrOO code to continue receiving the necessary maintenance. Since OpenNLP became part of Apache, the project has also had the indirect support of this foundation.

Another admissible association with community-based free software initiatives refers to their similarity to the concept of temporary organizations (TOs). As the definition of TOs expresses it (Burke & Morley, 2016), the projects were characterized as interdependent actors who directed their actions towards accomplishing complex tasks. The release of software versions is the termination point, which occurs as a result of the enterprise setting a specific date, or reaching a previously defined state or condition (Bakker, 2011; Bakker et al., 2009). Another way of perceiving free software initiatives like TOs is through the association of projects with different stakeholders. In these associations, structural adjustments and control processes took place in the projects, marking the outlines of a temporary organizational arrangement that changed as new relationships with stakeholders emerged. Another characteristic that makes the investigated projects more like TOs is the fact that leadership has proved to be a function of fundamental relevance in directing collaborative work (Bakker, 2011).

CONCLUSIONS

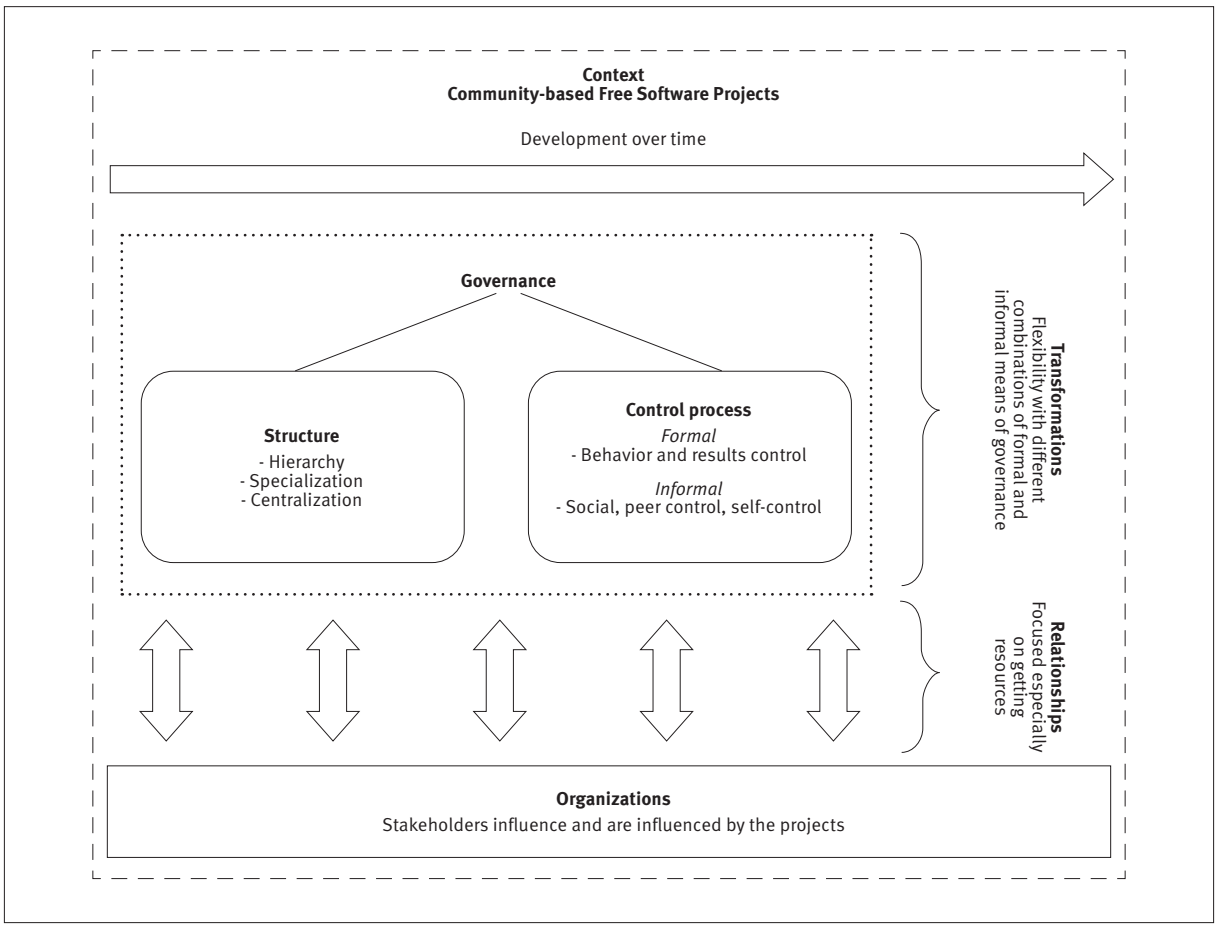
The difficulty of separating communities from the formal initiatives developed within the scope of the projects investigated was evident in this study. Governance research must consider the entire context in which community projects take place and the relationships established by these projects. Governance (the structure and control processes) proved to be a flexible and changing concept, which adapted to the development paths taken by these projects. Throughout these changes, formal and informal governance procedures took place.

Although community traits persisted during the development of these projects, management was formalized, mainly due to relationships established between organizational actors. An analysis of these projects indicates that free software ventures need to establish contacts with stakeholders in order to obtain the financial, material, and human resources necessary for their survival. Leadership has proved to be crucial in finding these resources in the environment and for guiding these businesses.

Even though the literature specifies the developmental phases that free software projects pass through and valid features that differentiate them from traditional organizations, in practice these situations did not occur in a

well-defined manner. Although the informal characteristics of the community were retained, the transformations that led to the development of these projects were associated with relationships for obtaining resources that were developed with other organizations. Formal governance in these relationships was necessary at times, and coexisted with the informality of community spaces. Instead of the development phases occurring in a continuous growth perspective, the projects experienced moments of greater and lesser activity that were governed by the resources they were able to obtain from formal organizations. These projects did not fit entirely within either of the concepts in the bazaar vs. cathedral discussion. They shared similarities with traditional organizations, while maintaining community characteristics. Thus, they proved to be intricate mixtures of both types of organization.

Figure 1. Core research idea



This paper's limitations include the choice of units of analysis based on the availability of projects and the fact that the results reflect the investigated context of free software communities. In future research, it is recommended that quantitative or mixed methodological approaches be used, as well as investigations of other types of community organization. In terms of its contributions, this study provides researchers and professionals with a more realistic understanding of the governance of community businesses and the tactics that leaders use to obtain the resources they need to ensure their projects survive.

REFERENCES

Arazy, O., Daxenberger, J., Lifshitz-Assaf, H., Nov, O., & Gurevych, I. (2016). Turbulent stability of emergent roles: The dualistic nature of self-organizing knowledge coproduction. *Information Systems Research*, *27*(4), 792-812. doi: 10.1287/isre.2016.0647

Bakker, R. M. (2011). *It's only temporary: Time and learning in inter-organizational projects*. Tilberg: University Tilberg.

Bakker, R. M., Cambré, B., & Provan, K. G. (2009). The resource dilemma of temporary organizations: A dynamic perspective on temporal embeddedness and resource discretion. In Bakker, R. M., Cambré, B., & Provan, K. G. (Eds.), *Temporary organizations: Prevalence, logic and effectiveness* (pp. 201-219). United Kingdom: Edward Elgar Publishing. doi: 10.4337/9781849802154.00017

Bardin, L. (2010). *Análise de conteúdo*. Lisboa, Portugal: Edições 70.

Bauer, J., Franke, N., & Tuertscher, P. (2016). Intellectual property norms in online communities: How user-organized intellectual property regulation supports innovation. *Information Systems Research*, *27*(4), 724-750. doi: 10.1287/isre.2016.0649

Bryson, J. M. (2018). *Strategic planning for public and nonprofit organizations: A guide to strengthening and sustaining organizational achievement*. Hoboken, New Jersey. John Wiley & Sons.

Burke, C. M., & Morley, M. J. (2016). On temporary organizations: A review, synthesis and research agenda. *Human Relations*, *69*(6), 1235-1258. doi: 10.1177/0018726715610809

Crowston, K., Wei, K., Howison, J., & Wiggins, A. (2012). Free/Libre open-source software development: What we know and what we do not know. *ACM Computing Surveys*, *44*(2), Article 7. doi: 10.1145/2089125.2089127

Dahlander, L., & O'Mahony, S. (2011). Progressing to the center: Coordinating project work. *Organization Science*, *22*(4), 961-979. doi: 10.1287/orsc.1100.0571

Davis, G. F., & Cobb, J. A. (2010). Resource dependence theory: Past and future. In S. B. Bacharach (Ed.), *Research in the sociology of organizations*. London, UK: Elsevier.

Demil, B., & Lecocq, X. (2006). Neither market nor hierarchy nor network: The emergence of bazaar governance. *Organization Studies*, *27*(10), 1447-1466. doi: 10.1177/0170840606067250

Eseryel, U. Y., Wei, K., & Crowston, K. (2020). Decision-making processes in community-based free/libre open source software-development teams with internal governance: An extension to decision-making theory. *Communications of the Association for Information Systems*, *46*(1), 20. doi: 10.17705/1cais.04620

Faraj, S., Jarvenpaa, S. L., & Majchrzak, A. (2011). Knowledge collaboration in online communities. *Organization Science*, *22*(5), 1224-1239. doi: 10.1287/orsc.1100.0614

Forte, A., & Lampe, C. (2013). Defining, understanding, and supporting open collaboration: Lessons from the literature. *American Behavioral Scientist*, *57*(5), 535-547. doi: 10.1177/0002764212469362

Freeman, R. E. (2010). *Strategic management: A stakeholder approach*. Cambridge, United Kingdom: Cambridge University Press.

Friedman, A. L., & Miles, S. (2006). *Stakeholders: Theory and practice*. Oxford: Oxford University Press on Demand.

Frooman, J. (1999). Stakeholder influence strategies. *Academy of Management Review*, *24*(2), 191-205. doi: 10.5465/amr.1999.1893928

Grabher, G. (2004). Temporary architectures of learning: Knowledge governance in project ecologies. *Organization Studies*, *25*(9), 1491-1514. doi: 10.1177/0170840604047996

Guimarães, A. L., Korn, H. J., Shin, N., & Eisner, A. B. (2013). The life cycle of open source software development communities. *Journal of Electronic Commerce Research, 14*(2), 167. Retrieved from http://www.jecr.org/node/25

Hillman, A. J., Withers, M. C., & Collins, B. J. (2009). Resource dependence theory: A review. *Journal of Management*, *35*(6), 1404-1427. doi: 10.1177/0149206309343469

Jensen, C., & Scacchi, W. (2010). *Governance in free software development projects: A comparative multi-level analysis*. International Conference on Free Systems, Springer, Heidelberg.

Kolbjørnsrud, V. (2016). Agency problems and governance mechanisms in collaborative communities. *Strategic Organization, Special Issue Article*, 1-36. doi:10.1177/1476127016653727

Lattemann, C., & Stieglitz, S. (2005). Framework for governance in free communities. *Proceedings of the HICSS'05*, Hawaii.

Lee, G. K., & Cole, R. E. (2003). From a firm-based to a community-based model of knowledge creation: The case of the Linux kernel development. *Organization Science*, *14*(6), 633-649. doi: 10.1287/orsc.14.6.633.24866

Lindberg, A., Berente, N., Gaskin, J., & Lyytinen, K. (2016). Coordinating interdependencies in online communities: A study of an free software project. *Information Systems Research*, *27*(4), 751-772. doi: 10.1287/isre.2016.0673

Li-Ying, J., & Salomo, S. R. (2013). Design of governance in virtual communities: Definition, mechanisms, and variation patterns. *International Journal of Collaborative Enterprise*, *3*(4), 225-251. doi: 10.1504/IJCENT.2013.058619

Mäenpää, H., Munezero, M., Fagerholm, F., & Mikkonen, T. (2017, August). The many hats and the broken binoculars: State of the practice in developer community management. *Proceedings of the OpenSym*, Galway, 1-9.

Markus, M. L. (2007). The governance of free/free software projects: Monolithic, multidimensional, or configurational? *Journal of Management & Governance*, *11*(2), 151-163. doi 10.1007/s10997-007-9021-x

Miscione, G., Ziolkowski, R., Zavolokina, L., & Schwabe, G. (2018, September). Tribal governance: The business of blockchain authentication. *Proceedings of the HICSS*, Hawaii, 1-10.

O'Mahony, S., & Ferraro, F. (2007). The emergence of governance in an free community. *Academy of Management Journal*, *50*(5), 1079-1106. doi: 10.5465/amj.2007.27169153

Panchal, J. H. (2010). Coordination in collective product innovation. *Proceedings of the ASME*, Vancouver, Canada.

Pfeffer, J. (1988). A resource dependence perspective on intercorporate relations. In M. Mizruchi & M. Schwartz (Eds.), Intercorporate Relations: The Structural Analysis of Business (Structural Analysis in the Social Sciences, pp. 25-55). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511570841.002

Pfeffer, J., & Salancik, G. R. (2003). *The external control of organizations: A resource dependence perspective*. Stanford, California: Stanford University Press.

Raymond, E. (1999). The cathedral and the bazaar. *Knowledge, Technology & Policy*, *12*(3), 23-49. Retrieved from https://link.springer.com/content/pdf/10.1007/s12130-999-1026-0.pdf

Raymond, E. (2005, October 3). The Cathedral and the Bazaar (Originally Published in Volume 3, Number 3, March 1998) [Special Issue]. *First Monday*. doi:10.5210/fm.v0i0.1472

Santos, C. D. (2010). Atratividade de projetos de software livre: importância teórica e estratégias para administração. *RAE-Revista de Administração de Empresas*, *50*, 424-438. doi: 10.1590/S0034-75902010000400007

Santos, C. D., Kuk, G., Kon, F., & Pearson, J. (2013). The attraction of contributors in free and open source software projects. *Journal of Strategic Information Systems, 22*(1), 26-45. doi: 10.1016/j.jsis.2012.07.004

Seidel, M. D. L., & Stewart, K. J. (2011). An initial description of the C-form. In C. Marquis, M. Lounsbury, & R. Greenwood (Eds.), *Communities and organizations, Research in the sociology of organizations* (pp. 37-72). Bingley: Emerald Group Publishing Limited.

Shah, S. K. (2006). Motivation, governance, and the viability of hybrid forms in free software development. *Management Science*, *52*(7), 1000-1014. doi: 10.1287/mnsc.1060.0553

Shah, S. K., & Gorbatai, A. D. (2015). Structural sampling: A technique for illuminating social systems. In Elsbach, K., D. Kramer, R. *Handbook of Qualitative Organizational Research*, (pp 251-261). New York: Routledge

Shaikh, M., & Vaast, E. (2016). Folding and unfolding: Balancing openness and transparency in free communities. *Information Systems Research*, *27*(4), 813-833. doi: 10.1287/isre.2016.0646

Sharif, S. P., & Yeoh, K. K. (2014). Independent directors' resource provision capability in publicly-listed companies in Malaysia. *Corporate Ownership and Control*, *11*(3), 113-121. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2822679

Sydow, J., Lindkvist, L., & DeFillippi, R. (2004). Project-based organizations, embeddedness and repositories of knowledge: Editorial. *Organization Studies*, *25*(9), 1475-1489. doi: 10.1177/0170840604048162

Viseur, R., & Charleux, A. (2019). Changement de gouvernance et communautés open source: Le cas du logiciel Claroline. *Innovations*, (1), 71-104. Retrieved from https://ideas.repec.org/a/cai/inndbu/inno_058_0071.html

West, J., & O'Mahony, S. (2005). Contrasting community building in sponsored and community founded free projects. *Proceedings of the HICSS'05*, Hawaii.

Wynn, D. E. (2003). Organizational structure of free projects: A life cycle approach. *Proceedings of the Annual Conference of the Southern AIS*, Georgia.

Xu, B., Xu, Y., & Lin, Z. (2005). Control in free software development. *Proceedings of the Americas Conference on Information Systems (AMCIS)*, Omaha.

Zebari, R. R., Zeebaree, S. R., Jacksi, K., & Shukur, H. M. (2019). E-business requirements for flexibility and implementation enterprise system: A review. *International Journal of Scientific and Technology Research*, *8*(11), 655-660. Retrieved from https://1library.net/document/ye8v26ry-e-business-requirements-flexibility-implementation-enterprise-review.html

AUTHORS' CONTRIBUTIONS

The authors declare that they participated in all stages of the paper's development. Isabela Neves Ferraz helped prepare the research and in data analysis. Carlos Denner dos Santos participated in guiding the entire process involved in searching and revising the text, and making the necessary adjustments.