

# ARTIGOS

Submetido 05.06.2020. Aprovado 05.01.2021

Avaliado pelo sistema *double blind review*. Editor Científico convidado: José Carlos Freitas Junior

Versão traduzida | DOI: <http://dx.doi.org/10.1590/S0034-759020220104x>

## TRANSFORMAÇÃO DE PROJETOS DE DESENVOLVIMENTO DE SOFTWARE LIVRE: UMA GOVERNANÇA ENTRE A CATEDRAL E O BAZAR

*Transformation of free and open source software development projects: Governance between the cathedral and bazaar*

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

Isabela Neves Ferraz<sup>1</sup> | [isabelanf.adm@gmail.com](mailto:isabelanf.adm@gmail.com) | ORCID: 0000-0001-8536-8418

Carlos Denner dos Santos<sup>1,2</sup> | [carlosdenner@unb.br](mailto:carlosdenner@unb.br) | ORCID: 0000-0002-4481-0115

<sup>1</sup>Universidade de Brasília, Programa de Pós-Graduação em Administração, SOCIE-DADOS, Brasília, DF, Brasil

<sup>2</sup>Universite du Quebec a Montreal, LATECE, Montreal, QC, Canadá

### 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.

### 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.

### 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.

## INTRODUÇÃO

Os avanços na tecnologia da informação e comunicação (TIC) e sua difusão, aliados à crescente complexidade ambiental (Seidel & Stewart, 2011; Zebari, Zeebaree, Jacksi, & Shukur, 2019), têm facilitado o surgimento de organizações modernas e flexíveis, que se apresentam em uma infinidade de “configurações” conhecidas como comunidades virtuais (Arazy, Daxenberger, Lifshitz-Assaf, Nov, & Gurevych, 2016; Faraj, Jarvenpaa, & Majchrzak, 2011). Um exemplo comum de empreendimento baseado no formato de comunidade virtual é o projeto de software livre (Li-Ying & Salomo, 2013; Seidel & Stewart, 2011). Essas comunidades são representadas por projetos criados na Internet, nos quais o código-fonte do software é disponibilizado publicamente para receber as contribuições externas de usuários e desenvolvedores por meio da significativa participação de voluntários (Eseryel, Wei, & Crowston, 2020; Santos, 2010; Seidel & Stewart, 2011; Shah, 2006).

É de conhecimento geral que as comunidades virtuais, como os projetos de software livre, são organizadas de maneira diferente das organizações tradicionais (Arazy et al., 2016; Lee & Cole, 2003; Miscione, Ziolkowski, Zavolokina, & Schwabe, 2018; Seidel & Stewart, 2011). Na metáfora utilizada por Raymond (1999, 2005) a respeito do modelo comunitário de produção, os projetos de software livre são vistos como bazares, com seu método de trabalho informal, enquanto as organizações tradicionais são percebidas como catedrais, que funcionam por meio de estrutura e processos de controle formalizados.

Os estudos de software livre precisam detalhar as especificidades das suas operações (Bauer, Franke, & Tuertscher, 2016), a fim de entender como, na ausência de meios formais ou na presença mínima deles, as comunidades conduzem ações para o alcance de resultados (Arazy et al., 2016; Eseryel et al., 2020; Miscione et al., 2018; Panchal, 2010). Um conceito central para verificar como o trabalho é realizado nas organizações é o de governança, que é multidimensional e envolve os meios formais e informais que direcionam as ações dos indivíduos (Li-Ying & Salomo, 2013; Markus, 2007).

É evidente a partir da literatura de software livre que, embora seja comum diferenciar entre as características das comunidades e das organizações tradicionais (Arazy et al., 2016; Lee & Cole, 2003; Seidel & Stewart, 2011), os autores nem sempre consideram as mudanças nos projetos e seu desenvolvimento nessa diferenciação. Como resultado, projetos maduros de software livre podem ser considerados como negócios que seguem uma lógica de gestão informal ao invés de adotar um modelo organizacional tradicional (Demil & Lecocq, 2006; Markus, 2007; Raymond, 2005). Isso pode ser observado em estudos como os de Lee e Cole (2003), Shah (2006), Seidel e Stewart (2011) e Lindberg, Berente, Gaskin e Lyytinen (2016). Eles citam projetos de sucesso, mas não analisam as mudanças que ocorreram neles ao longo do tempo. Embora os trabalhos de Lattemann e Stieglitz (2005), Xu, Xu e Lin (2005) e Panchal (2010) reconheçam a existência de diferentes estágios de desenvolvimento em iniciativas de software livre, esses autores não realizaram exames empíricos, nem chegaram a conclusões práticas.

Diante dessas considerações, é necessário um maior entendimento de como os projetos de software livre conduzem as ações durante o seu desenvolvimento (Forte & Lampe, 2013; Guimarães, Korn, Shin, & Eisner, 2013). Esta pesquisa pretende responder à seguinte questão: Em termos de estrutura e controle, como a governança aparece e se transforma no desenvolvimento de projetos de software livre comunitários? Ela também analisa esses projetos a partir da perspectiva da teoria organizacional e de abordagens que envolvem a dependência de recursos, os *stakeholders* e as organizações temporárias. Este estudo qualitativo usa a estratégia de estudo de caso com foco em três projetos de software livre comunitários. A ideia é trazer uma visão dinâmica que possibilite a pesquisadores e profissionais entender os principais vetores de mudança na governança durante o

desenvolvimento de negócios de software livre, o que permitirá verificar se as comunidades (bazares) diferem das organizações tradicionais (catedrais) ou são semelhantes a elas.

## REVISÃO DA LITERATURA

### Governança em projetos de software livre comunitários

As comunidades de software livre retêm todas as características essenciais de uma comunidade virtual, onde membros geograficamente dispersos, principalmente voluntários, trabalham e colaboram em espaços virtuais flexíveis mediados pela TIC (Eseryel et al., 2020; Faraj et al., 2011; Seidel & Stewart, 2011; Shah, 2006). Elas são diferentes das que produzem software comercialmente, pois são estruturas que promovem o uso livre do software e disponibilizam seu código-fonte gratuitamente na Internet (Miscione et al., 2018; Santos, Kuk, Kon, & Pearson, 2013; Seidel & Stewart, 2011). Além disso, essas comunidades têm ativamente produzido soluções competitivas e inovadoras (Bauer et al., 2016; Viseur & Charleux, 2019).

De acordo com Markus (2007), a governança no contexto das comunidades de software livre é representada pelos meios formais e informais que permitem que indivíduos, total ou parcialmente autônomos, contribuam conjuntamente para um projeto de desenvolvimento de software. Apesar da importância da governança nas iniciativas de software livre, estudá-la ainda é um tema controverso (Li-Ying & Salomo, 2013), principalmente porque uma parcela significativa das pesquisas considera tais comunidades como arranjos caracterizados pela gestão informal, diferente da lógica hierárquica e formal das organizações tradicionais (Eseryel et al., 2020; Seidel & Stewart, 2011). Embora reconhecendo tais características, no entanto, outros estudos argumentam que os mecanismos de governança usados pelas comunidades de software livre são bastante flexíveis, e portanto, podem ser modificados com o tempo (Lattemann & Stieglitz, 2005; Li-Ying & Salomo, 2013; Mäenpää, Munezero, Fagerholm, & Mikkonen, 2017; Viseur & Charleux, 2019; Xu et al., 2005).

Este estudo foi iniciado a partir da compreensão da governança como algo conduzido de maneira dinâmica, considerando a trajetória do desenvolvimento dos projetos de software livre comunitários. Entre os múltiplos prismas pelos quais a governança pode ser estudada, dois são analisados aqui: estrutura (Lindberg et al., 2016; O'Mahony & Ferraro, 2007; Seidel & Stewart, 2011) e processos de controle (Lattemann & Stieglitz, 2005; Xu et al., 2005).

### Estrutura e controle de projetos de software livre comunitários

Do ponto de vista estrutural, as comunidades de software livre diferem das organizações tradicionais devido à ausência de uma hierarquia e autoridade centrais (Lee & Cole, 2003; Miscione et al., 2018; Panchal, 2010). Essas comunidades adotam uma perspectiva em que o poder é estabelecido horizontalmente, da periferia em direção ao centro do projeto. Os participantes do centro são considerados pelos outros como conhecedores do projeto e, portanto, os tomadores de decisão mais confiáveis (Dahlander & O'Mahony, 2011).

Os papéis nas comunidades de software livre não são especializados, de forma que o exercício das funções não é algo atribuído previamente e ocorre durante o próprio processo de trabalho colaborativo (Arazy et al., 2016; Lee & Cole, 2003). As sobreposições na execução de tarefas são esperadas e explicam a participação de colaboradores atuando em várias funções dentro do projeto (Lee & Cole, 2003; Seidel & Stewart, 2011). Mesmo que não existam funções deliberadamente especificadas nos projetos, a liderança é frequentemente bem defi-

nida (Lee & Cole, 2003). A liderança é vista como uma condição para o sucesso de um projeto, uma vez que líderes assumem a responsabilidade pelos vários subsistemas e tomam decisões críticas em relação aos códigos-fonte e direcionamentos de negócios (Forte & Lampe, 2013; Guimarães et al., 2013; Mäenpää et al., 2017).

No que diz respeito ao processo de tomada de decisão, a literatura sobre comunidades de software livre relaciona diferentes abordagens. Paradoxalmente, embora se presuma que as decisões mais impactantes se concentram na autoridade do coletivo (Lee & Cole, 2003; Seidel & Stewart, 2011; West & O'Mahony, 2005), sugere-se que os membros da comunidade tenham maior independência no desempenho de suas atividades. Isso não ocorre apenas devido à distância física entre si, mas também porque as comunidades são grupos não hierárquicos compostos principalmente por voluntários (Panchal, 2010; Seidel & Stewart, 2011).

Do ponto de vista do controle, embora os meios formais orientados por resultados e comportamentos possam ser encontrados em comunidades de software livre (O'Mahony & Ferraro, 2007), o entendimento mais comum é que são os meios informais que se destacam, como o autocontrole, o controle social e entre pares (Jensen & Scacchi, 2010; Kolbjørnsrud, 2016; Lattemann & Stieglitz, 2005; Xu et al., 2005). O controle social em projetos de software livre busca fortalecer o sentimento de pertencimento à comunidade, baseado em intensa socialização e premissas culturais. Esse controle envolve o compartilhamento de objetivos e valores que visam atingir a orientação comportamental dos indivíduos (Lattemann & Stieglitz, 2005; Xu et al., 2005).

Para o autocontrole, as plataformas de tecnologia que apoiam as práticas da comunidade e permitem contribuições individuais com altos níveis de colaboração, enfatizam a autonomia e independência na realização do trabalho, reduzindo assim os esforços de coordenação (Crowston, Wei, Howison, & Wiggins, 2012; Lindberg et al., 2016). A liderança do projeto desempenha um papel essencial no estabelecimento do autocontrole, especialmente na seleção dos membros e do trabalho a ser realizado (Xu et al., 2005). O controle entre pares é uma alternativa para garantir a qualidade dos aplicativos produzidos pelas comunidades de software livre (Kolbjørnsrud, 2016; Lattemann & Stieglitz, 2005). Sua finalidade é permitir que os indivíduos revisem e critiquem o trabalho uns dos outros. Esta crítica pode servir para corrigir erros, solucionar problemas e promover melhorias e trocas de conhecimentos (Lee & Cole, 2003)

### Transformações de estrutura e controle

Embora seja possível descrever os atributos dos empreendimentos comunitários em sua forma orgânica (Arazy et al., 2016; Seidel & Stewart, 2011), o fato de as iniciativas de software livre passarem por sucessivas transformações justifica contextualizá-las em relação ao desenvolvimento que alcançam (Faraj et al., 2011; Mäenpää et al., 2017). Autores como Wynn (2003), Lattemann e Stieglitz (2005) e Xu et al. (2005) apontaram que tais iniciativas passam por fases sucessivas de desenvolvimento (introdução, crescimento, maturidade e declínio, ou renascimento) que afetam a governança.

Apesar de alguns autores, como Raymond (1999, 2005), afirmarem que o trabalho em projetos de software livre se assemelha à lógica de um “bazar”, em que os participantes são voluntários e se organizam de forma não hierárquica e informal (uma lógica contrária à da visão de organização “catedral”, ou seja, hierárquica e formal), esse tipo de iniciativa é dinâmico e pode modificar-se com o tempo. Acredita-se que a estrutura e os processos de controle podem emergir de um contexto mais informal e incorporar elementos formais à medida que esses projetos se desenvolvem (Arazy et al., 2016; Forte & Lampe, 2013; O'Mahony & Ferraro, 2007; Panchal, 2010)

Um fator que pode contribuir para essa formalização e afetar a governança é a presença de outros atores organizacionais nas atividades da comunidade, como patrocinadores e fundações (Mäenpää et al., 2017; Viseur & Charleux, 2019; West & O'Mahony, 2005). No entanto, esse funcionamento formal das iniciativas de software livre comunitárias precisa ser melhor compreendido e tem levado pesquisadores a questionar algumas das ideias aceitas sobre como essas comunidades trabalham e organizam suas atividades (Shaikh & Vaast, 2016).

## Contribuição organizacional conceitual e teórica

### A abordagem stakeholder

Segundo Freeman (2010), *stakeholders* (ou partes interessadas) são indivíduos ou grupos de indivíduos que podem influenciar o desempenho específico das organizações, incluindo seus procedimentos e objetivos de gestão. O conceito tem sido amplamente utilizado em estudos organizacionais desde 1984, quando apresentado por Freeman pela primeira vez na obra *Strategic Management: The Stakeholder Approach*. A abordagem tem diferentes definições e formas de medição avaliativa.

Embora a abordagem do *stakeholder* tenha sido mais comumente observada em estudos envolvendo empresas privadas, ela vem sendo cada vez mais usada em análises de organizações públicas e sem fins lucrativos, principalmente pelas preocupações com relação à alocação de recursos e ao atendimento de grupos com maior eficiência (Bryson, 2018). Também é comum que a abordagem dos *stakeholders* seja usada em conjunto com outras teorias organizacionais (Frooman, 1999).

Independentemente do contexto ou da teoria sobre a abordagem dos *stakeholders*, a organização deve ser entendida como um grupo de atores que têm problemas comuns que envolvem a gestão de interesses e pontos de vista nem sempre convergentes (Friedman & Miles, 2006). Isso deve ser visto como uma via de mão dupla, já que os *stakeholders* e as organizações sofrem influências recíprocas. Essa abordagem também lida com o gerenciamento das contradições e conflitos que surgem entre indivíduos e/ou grupos, e busca resolvê-los (Friedman & Miles, 2006; Frooman, 1999).

### Teoria da dependência de recursos

O pioneiro trabalho de Pfeffer e Salancik em 1978, *The external control of organizational: a resource dependence*, consolidou a teoria da dependência de recursos (TDR). A TDR considera que fatores externos influenciam o comportamento das organizações (Pfeffer & Salancik, 2003). Mas enquanto reconhece a inter-relação entre a organização e o contexto, a TDR produz uma concepção mais proativa do empreendimento, pois os gestores, empregando várias táticas, podem reduzir incertezas, riscos e dependências (Hillman, Withers, & Collins, 2009).

Dada a interdependência entre as organizações e os ambientes em que operam, elas podem reconfigurar sua estrutura interna para atender às demandas atuais e usar várias estratégias para garantir os recursos de que precisam para sobreviver e ser competitivas (Sharif & Yeoh, 2014). Essa função de fornecer e garantir recursos relevantes geralmente recai sobre a pessoa responsável pela gestão da organização (Sharif & Yeoh, 2014).

A ampla gama de ações formais ou informais realizadas pelos gestores tem como objetivo primordial assegurar os meios essenciais, aumentar o poder nas relações de troca e reduzir a subordinação e as incertezas que as organizações enfrentam (Pfeffer & Salancik, 2003). Segundo Pfeffer (1988), um dos principais argumentos da TDR é que as organizações não são autônomas. Há interdependência com outras organizações, o que significa

que as incertezas e as ações orientadas para a sobrevivência das empresas precisam ser gerenciadas. Em suma, as três ideias centrais da TDR são: o contexto social é relevante; as organizações precisam de estratégias para buscar maior autonomia e atender aos seus próprios interesses; e o poder é essencial para a compreensão das ações internas e externas das organizações (Davis & Cobb, 2010).

## Organizações Temporárias (OTs)

As organizações temporárias (OTs) são representadas por grupos de atores interdependentes, estruturados para realizar uma tarefa complexa (Burke & Morley, 2016). As definições do conceito convergem quando afirmam que as OTs têm uma data definida para serem encerradas, ou que atingem um estado ou condição predefinida (Bakker, 2011; Bakker, Cambré, & Provan, 2009).

As OTs são orientadas para tarefas vitais (Bakker, 2011), especialmente aquelas caracterizadas pela complexidade decorrente de suas interdependências, a ambiguidade de seus interesses e a incerteza dos resultados (Bakker, 2011; Burke & Morley, 2016). Por essas características, o trabalho em equipe nas OTs precisa ser coordenado e o papel do líder é considerado crucial (Bakker, 2011). As OTs estão intrinsecamente relacionadas ao ambiente externo, representado pelo contexto organizacional e pelo ambiente social mais amplo, englobando relacionamentos com outras organizações. Os recursos essenciais necessários para a sobrevivência das OTs são encontrados no ambiente externo (Grabher, 2004). Esse entrelaçamento com o contexto pode se tornar um dilema para as OTs, pois enquanto tais estruturas buscam autonomia, elas também precisam adquirir os recursos necessários para operar (Burke & Morley, 2016).

Tradicionalmente, as OTs eram entendidas como arranjos complementares às estruturas organizacionais permanentes. Posteriormente, configurações alternativas como projetos interorganizacionais e organizações baseadas em projetos passaram a ser consideradas como OTs (Bakker, 2011; Burke & Morley, 2016). Este tipo de organização pode assumir muitas formas que ultrapassam os coletivos intra e interorganizacionais, envolvendo também grupos que cooperam e integram recursos e competências para alcançar resultados (Sydow, Lindkvist, & DeFillippi, 2004).

## MÉTODO

A presente pesquisa qualitativa adota a estratégia do estudo de caso para examinar o setor de software livre em três fases principais. A primeira fase envolveu a escolha dos projetos a serem estudados. Os pesquisadores mapearam projetos competitivos brasileiros indicados por pesquisadores e profissionais que trabalham com software livre, bem como projetos que participaram de eventos no Brasil. Os critérios de seleção foram posteriormente aplicados. Com base nos trabalhos de Xu et al. (2005), Latterman e Stieglitz (2005), Panchal (2010) e Viseur e Charleux (2019), a intenção foi identificar projetos brasileiros com possíveis variações em seus perfis de governança e desenvolvimento. Esses projetos variaram em função das seguintes características: o ano de criação; o número de versões de software; o número de *committers* (desenvolvedores com a capacidade de modificar o repositório de código-fonte); a licença usada; e a presença de fundações ou patrocinadores. Esse processo levou à seleção de três projetos: CoGrOO, Noosfero e Kytos.

A segunda fase abarcou a coleta de dados. As entrevistas foram agendadas por e-mail e realizadas presencialmente, por meio do Skype ou Google Hangouts, aplicando-se um roteiro semiestruturado, previamente testado com cinco participantes brasileiros que gerenciam projetos de software livre.

A seleção para as entrevistas foi intencional e incluiu líderes, desenvolvedores e *stakeholders* envolvidos no grupo central dos projetos. Os primeiros entrevistados eram informantes-chave com conhecimento da história do projeto e os demais selecionados pela técnica da bola de neve. Foram realizadas catorze entrevistas e a pesquisa não utilizou a saturação teórica para estabelecer esse número. A preocupação residiu em ter a história dos projetos contada por indivíduos que conheciam a sua realidade e que ocupavam diferentes papéis, conforme definido pela amostragem estrutural (Shah & Gorbatai, 2015). No caso do CoGrOO, a história do projeto, em determinado momento, teve uma conexão com o projeto global Apache OpenNLP, o que exigiu a coleta de dados complementares. As informações não obtidas na entrevista com o informante-chave ou nos documentos, foi complementada com perguntas em formulário enviado por e-mail aos participantes do projeto Apache OpenNLP, de modo que dos cinco formulários enviados, dois deles foram respondidos.

A pesquisa documental foi usada para mapear projetos de software livre que pudessem ser incluídos no estudo e também para obter dados complementares aos que foram coletados nas entrevistas. O conteúdo das listas de e-mail, sites e documentos diversos (slides, relatórios, manuais e notícias) foram acessados e organizados em uma planilha.

Tabela 1. Critério de seleção dos projetos

| Projeto                 | Ano de Criação | Nº de Versões | Nº de <i>committers</i> | Licença            | Fundação / Patrocinador | Nº de entrevistados |
|-------------------------|----------------|---------------|-------------------------|--------------------|-------------------------|---------------------|
| Kytos                   | 2014           | 3             | 6                       | MIT                | Patrocinador            | 5                   |
| CoGrOO / Apache OpenNLP | 2003 / 2000    | 17 / 34       | 3 / 23                  | Apache Licença 2.0 | Fundação                | 3 + 2 formulários   |
| Noosfero                | 2007           | 191           | 12                      | GNU AGPLv3         | Patrocinador            | 6                   |

Na terceira fase, os dados coletados nas entrevistas e na pesquisa documental foram transcritos e tratados por meio da técnica de análise de conteúdo (Bardin, 2010), com apoio do software Atlas.ti. O texto foi codificado de acordo com indicadores que se referem a cada uma das categorias de análise previamente estabelecidas, a partir da revisão da literatura. Embora as categorias de análise tenham derivado da literatura, não impediram descobertas durante a pesquisa de campo.

Quadro 1. Categorias de análise definidas pela literatura

| Conceitos                          | Dimensões               |   | Indicadores   |   |
|------------------------------------|-------------------------|---|---|---|
| Governança                         | Estrutura               | Hierarquia  | Cadeia de comando, relações subordinadas.   |   |
|                                    |                         | Função especializada                                  | Distribuição das funções e responsabilidades, divisão do trabalho.                  |   |
|                                    |                         | Centralização   | Decisão focada em um determinado nível individual e organizacional.                 |   |
|                                    | Controle                | Formal (Comportamento e resultados)                   | Padronização, regras, regulamentos, recompensas baseadas em resultados individuais. |   |
|                                    |                         | Informal  | Social  | Compartilhamento de culturas e práticas, socialização dos indivíduos. |
|                                    |                         |   | Controle entre pares  | Ajuste mútuo, pares fazem a crítica ao trabalho.                      |
| Teorias organizacionais/ Abordagem | Autocontrole            |   | Independência na execução do trabalho.  |   |
|                                    | Dependência de Recursos |   | Busca por recursos no ambiente.   |   |
|                                    | Stakeholders            |   | Relacionamento com atores internos e externos.                                      |   |
| Organizações Temporárias           |                         | Orientação por tarefa, data de encerramento definida. |   |   |

## ANÁLISE DE RESULTADOS

É importante notar que o projeto CoGrOO difere de outros, pois seu código-fonte está integrado ao projeto OpenNLP, que atualmente faz parte da Fundação Apache. O OpenNLP, portanto, também foi incluído na análise.

### Histórico dos casos

#### CoGrOO

O projeto CoGrOO é um corretor gramatical que pode ser acoplado ao OpenOffice.org. Ele auxilia na escrita do texto, verificando se a gramática está correta usando o processamento de linguagem natural (PLN). O projeto CoGrOO teve início em 2003, quando um professor universitário, que na época trabalhava no Serviço Nacional de Aprendizagem Comercial (Senac-SP), observou a necessidade de criar um corretor gramatical para o OpenOffice.org. Juntamente com outros colegas do corpo docente da Universidade de São Paulo (USP), ele elaborou um projeto de desenvolvimento de software para essa finalidade.

Entre 2004 e 2006, o CoGrOO recebeu recursos do programa de financiamento de software livre da Financiadora de Estudos e Projetos (Finep). O apoio financeiro da Finep possibilitou a oferta de bolsas de estágio e graduação para o desenvolvimento do software, um recurso fundamental para a preparação das versões iniciais do corretor gramatical. Ao fim do financiamento, o CoGrOO continuou sendo desenvolvido por meio de voluntários e apoio institucional da Universidade de São Paulo (USP), interagindo com trabalhos de pesquisa da universidade e seus cursos acadêmicos. Alguns fatos ocorridos nesse período merecem destaque: a iniciativa obteve bolsas do programa Google *Summer of Code*; o Google forneceu ao projeto o seu código linguístico; o portal da comunidade CoGrOO foi desenvolvido em uma disciplina de pós-graduação da USP; e o software foi incorporado pelo Centro de Competência em Software Livre da USP (CCSL/USP).

Embora o CoGrOO tenha se tornado uma referência nacional, ele enfrenta o desafio de manter-se em operação devido à perda de recursos e a constante evasão de voluntários. No entanto, um membro que ingressou na comunidade como estagiário e posteriormente tornou-se voluntário, permaneceu interessado em dar continuidade ao projeto. Por volta de 2011 esse desenvolvedor notou que outro projeto do qual participava, o OpenNLP, possuía similaridades com o CoGrOO, embora não se destinasse a usuários finais. Como resultado e por iniciativa dele, cerca de 90% do código do CoGrOO foi incorporado ao OpenNLP por meio de um processo de correções conhecido como *patching*. Por transferir o código e porque desde 2011 o OpenNLP é um projeto Apache, o gerenciamento desse código passou a ser feito na infraestrutura da Fundação Apache.

#### Noosfero

O projeto Noosfero é uma plataforma *web* para redes de economia social e solidária. Além dos recursos padrão das redes sociais, o projeto oferece um sistema de gerenciamento de conteúdo (SGC). O software teve origem em 2017 e foi desenvolvido pela Cooperativa de Tecnologias Livres (Colivre), que visava atender a demandas semelhantes de dois clientes da cooperativa.

Surgido em um contexto de demanda comercial e sendo um software livre, a plataforma atraiu a atenção de atores externos, o que levou outras organizações a realizarem projetos utilizando o software no desenvolvimento de seus próprios *websites*. A existência de projetos paralelos trouxe uma dinâmica própria para o aprimoramento da comunidade Noosfero. Nesse sentido, enquanto os projetos eram realizados para atender aos interesses de



uma determinada organização, os apoiadores do software livre que participavam voluntariamente preocupavam-se em devolver o código desenvolvido à comunidade Noosfero. Dentre as instituições que atuaram nessas iniciativas, destacam-se: a Associação de Software Livre (ASL), o Centro de Competência em Software Livre da USP (CCSL/USP), o Serviço Federal de Processamento de Dados (Serpro), o Ministério do Planejamento, Desenvolvimento e Gestão (MP) e a Universidade de Brasília (UnB). Em alguns destes projetos as instituições mencionadas e membros da Colivre atuaram em parceria.

Mesmo com essas iniciativas paralelas, desde o início do Noosfero, o desenvolvimento de software sempre esteve atrelado ao patrocínio da Colivre. Até o momento, os contratos da cooperativa permitiram o aporte de investimentos, o que significa que a Colivre continua desempenhando um papel de liderança na gestão do projeto, lançando novas versões do software e conciliando os múltiplos interesses dos atores.

## Kytos

O Kytos é uma plataforma gratuita desenvolvida pela equipe do Núcleo de Computação Científica (NCC) da Universidade Estadual Paulista (Unesp) em parceria com o São Paulo Research and Analysis Center (SPRACE). Este projeto visa facilitar o desenvolvimento e implementação de redes definidas por software (*software defined networks* – SDN). O NCC desempenha um papel importante em projetos de instrumentação, hardware e redes de computadores. Ele possui parcerias nacionais e internacionais na área, a exemplo dos seguintes atores: Rede Nacional de Ensino e Pesquisa (RNP), Academic Network at São Paulo (ANSP), California Institute of Technology (Caltech), Conseil Européen pour la Recherche Nucléaire (CERN) e Americas Lightpaths (AmLight). Essas relações permitem aos membros da NCC realizar projetos conjuntos e participar de eventos no Brasil e no exterior.

A ideia do projeto surgiu quando, ao participar de um evento fora do país, um dos membros do NCC assistiu uma apresentação envolvendo transferência de dados usando controladores SDN gratuitos e observou algumas limitações nesses controladores. Em 2014, em um trabalho conjunto entre a equipe do NCC e desenvolvedores do Caltech, esse membro iniciou o desenvolvimento de uma biblioteca gratuita, chamada *python-openflow*. A criação desta biblioteca foi o embrião do projeto Kytos.

Um dos principais eventos que impulsionou o desenvolvimento do Kytos aconteceu no final de 2015, quando recebeu o patrocínio da Huawei. Essa empresa chinesa tem investido recursos financeiros e materiais no Kytos por meio da Lei de Informática, que possibilitou inclusive a contratação de estagiários para auxiliar no desenvolvimento do software. A biblioteca criada inicialmente tornou-se um repositório separado e atualmente é um subprojeto que integra o macroprojeto da plataforma SDN Kytos. Até o momento, o Kytos continua a ser apoiado pelo NCC e pela Huawei, bem como pelo trabalho voluntário realizado por organizações parceiras.

## Estrutura

Sobre a infraestrutura física e tecnológica dos projetos, todos eles começaram, em algum momento, a funcionar dentro de um contexto institucional. A existência de organizações em torno dos projetos mostra que o apoio institucional tem papel fundamental disponibilizando infraestrutura física e tecnológica básica. Segundo os entrevistados, os projetos envolvem o uso intensivo de recursos de TIC devido a abertura que a comunidade oferece ao software e a dispersão geográfica dos indivíduos (Bauer et al., 2016; Shah, 2006; Seidel & Stewart, 2011). Os recursos de TIC incluem: plataformas de hospedagem do código-fonte e controle de versões do software, como GitHub, GitLab e Sourceforge; plataformas de chat e vídeo, em particular Google Hangouts; fer-

ramentas para trabalho colaborativo em equipe, como Slack; listas de e-mail; e websites usados para divulgar informações sobre o projeto. É importante salientar que as listas de e-mail têm sido o meio de comunicação mais relevante em todas as iniciativas.

Quanto à hierarquia e às funções dos membros, no geral, as informações obtidas sugerem que, ao longo de seu desenvolvimento, os projetos tiveram que lidar com aspectos formais e informais. Sobre a hierarquia, os projetos indicaram formalmente os indivíduos designados para a gestão e adotaram, simultaneamente, a figura de uma autoridade paralela que transcende a formalidade estabelecida. Com relação ao último aspecto, o observado neste estudo converge com os resultados de autores como [Dahlander e O'Mahony \(2011\)](#) que, ao pesquisar projetos de software livre, apontaram que esses coletivos admitem o papel de uma autoridade lateral, alicerçada na experiência e contribuições dadas pelos indivíduos ao projeto.

Em consonância com este modelo de hierarquia formal e informal estão os papéis desempenhados nos projetos. Na lista das funções formais associadas à gestão coletiva, merecem destaque as seguintes: no CoGrOO, a responsabilidade formal da gestão concentrou-se nos docentes em diversos momentos ao longo do desenvolvimento do projeto; no Kytos, o acompanhamento era realizado pelo diretor do NCC e pelo coordenador ou líder do projeto (um funcionário da Unesp); no Noosfero e no OpenNLP, uma função comum a ambos os projetos era a de *release manager* (RM), ou gerente de lançamento (porém, enquanto o RM no Noosfero lançava versões do software e gerenciava o projeto, no OpenNLP essa função se concentrava especificamente no processo de lançamento da versão). As funções formais no OpenNLP eram as de *chair* (presidente), vice-presidente e membros do Comitê de Gerenciamento do Projeto (*Project Management Committee* – PMC).

As equipes desempenharam funções formais ou informais no desenvolvimento do software. Entre os responsáveis por funções formais estavam profissionais e estagiários contratados pelos projetos (especificamente nos casos do CoGrOO e do Kytos). No CoGrOO, a contratação ocorreu com o aporte de recursos da Finep, enquanto no Kytos a contratação aconteceu desde a entrada do patrocínio da Huawei (que continua até os dias de hoje). As funções informais ficaram a cargo de desenvolvedores voluntários que se engajaram nas atividades colaborativas. Também com relação à equipe de desenvolvimento, um agente comum em todos os coletivos é o *committer*, um desenvolvedor voluntário ou contratado que tem autoridade para escrever o código do software diretamente no repositório principal do projeto. No Noosfero e no OpenNLP, os *committers* aparecem em grupos formalmente constituídos. No Kytos, embora os papéis sejam pré-definidos, é o coordenador que convida um indivíduo para a função de *committer*, quando considera que o indivíduo é capacitado. No CoGrOO, essa função é informal.

Em geral, papéis formais e informais coexistiram ao longo do desenvolvimento do software nos projetos analisados. Enquanto a pesquisa em comunidades de software livre mostra que os voluntários colaboram significativamente em projetos comunitários ([Seidel & Stewart, 2011](#); [Shah, 2006](#)), os casos estudados aqui mostraram que essa afirmação nem sempre corresponde à realidade já que, principalmente em projetos específicos como Kytos e CoGrOO, a contratação de profissionais foi a solução para a manutenção do projeto em alguns momentos. A contratação e formalização de certas funções têm mostrado que, embora a literatura afirme que as funções são pouco definidas nas comunidades de software livre ([Arazy et al., 2016](#); [Eseryel et al., 2020](#); [Lee & Cole, 2003](#)), nem sempre essa foi a realidade dos projetos estudados.

Também foram verificadas semelhanças entre o observado nos casos e as descobertas de pesquisas anteriores sobre software livre. Por exemplo, em todos os projetos os entrevistados relataram que, conforme necessário, os indivíduos poderiam desempenhar funções diferentes daquelas primeiramente atribuídas a eles, independentemente de serem funções formais ou informais. Isso coincide com o que [Lee e Cole \(2003\)](#) e [Arazy et al. \(2016\)](#) defenderam ao relatar que os indivíduos podem desempenhar papéis múltiplos e sobrepostos em

espaços comunitários. Outra questão comum a todos os projetos é o papel da liderança, que tem sido crucial na governança para promover a unidade e o alinhamento de comportamentos e objetivos (Guimarães et al., 2013; Mäenpää et al., 2017; Seidel & Stewart, 2011). No Kytos, o coordenador é o principal responsável pela liderança, enquanto no CoGrOO os docentes e o principal desenvolvedor do software, após o afastamento dos demais membros, foram as pessoas que exerceram essa função. No Noosfero, o RM assumiu as funções de gerenciamento e liderança do projeto. Embora a função de *chair* (presidente) seja considerada um papel de liderança no OpenNLP, esse cargo destacou-se pela sua relação com questões burocráticas. Assim, outros membros do grupo com perfil de liderança tiveram a oportunidade de assumir papéis de comando dentro do projeto.

Na tomada de decisões, embora todos os projetos fossem abertos para que os participantes expressassem suas opiniões, na ausência de consenso ou quando essas decisões envolveram questões estratégicas sobre o rumo dos projetos, a centralização foi algo percebido. Essa percepção converge com o paradoxo dos estudos de software livre, segundo os quais, embora tais projetos sejam descentralizados, as decisões relevantes são frequentemente centralizadas (Lee & Cole, 2003; Seidel & Stewart, 2011; West & O'Mahony, 2005). No CoGrOO, a centralização aconteceu nos professores que supervisionavam o projeto em momentos específicos e no principal desenvolvedor do projeto. No OpenNLP, é adotado um processo formalizado, conhecido como reunião de consenso. Os membros do projeto votam em um determinado assunto por meio de listas de e-mail, e a solução resultante da contagem final dos votos é a que prevalece. Apesar dessa busca do consenso, os representantes com poder de voto nas questões mais relevantes são os membros do PMC. Se houver um impasse de decisão no projeto Noosfero, o indivíduo formalmente designado para tomar a decisão final é o RM. No caso do Kytos, as escolhas mais estratégicas são feitas pelo diretor do NCC e pelo coordenador do projeto, embora os membros sejam livres para expressar suas opiniões.

Noosfero e OpenNLP provaram ser os projetos com as regulamentações mais formais sobre como as decisões devem ser tomadas. Nos demais, as decisões são tomadas de acordo com a ocasião e sem obedecer a nenhum protocolo previamente definido. A centralização das decisões mais impactantes, no entanto, tem sido confiada principalmente a um ou mais indivíduos do coletivo, seja por um papel formalmente delineado, seja pela influência dos colaboradores mais experientes.

## Controle

Processos formais de controle de comportamento e resultados foram observados em alguns fluxos de atividades dos projetos investigados. Nesse sentido, destacam-se os seguintes processos: a escolha dos representantes e definição das responsabilidades para o exercício de determinada função; a tomada de decisões, em particular, daquelas que definem as estratégias e os objetivos do projeto; a regulamentação das contribuições recebidas, especialmente as relativas a alterações no código-fonte do software; as rotinas necessárias para lançar versões do software; a elaboração de relatórios para prestação de contas ou monitoramento dos projetos.

Os projetos analisados que se destacaram por possuírem processos bem definidos e formalmente delimitados foram o OpenNLP, Kytos e o Noosfero. O OpenNLP, assim como outros projetos Apache, caracterizou-se por seguir uma série de procedimentos detalhados no website da fundação. A Apache mantém um código de conduta que orienta os participantes e exige que os indivíduos que se tornam *committers* assinem um documento confirmando que seguem os termos de licença do projeto. O Kytos é semelhante ao projeto OpenNLP. Os indivíduos dispostos a colaborar precisam estar cientes de que serão regidos pelos termos de licença de software definidos no website do projeto. O Kytos também se destacou na formalização de algumas funções (além

do diretor do NCC e do coordenador), como a contratação de profissionais utilizando recursos do patrocínio da Huawei. Tanto o Kytos quanto o OpenNLP aderem à regra na qual os relatórios de acompanhamento do projeto devem ser entregues à empresa patrocinadora e à Fundação Apache, respectivamente.

Após vivenciar uma fase de conflito, na qual os voluntários passaram a exigir maior acesso ao projeto e independência da Colivre, o Noosfero redigiu um documento que continha uma série de formalizações. A ideia por trás dessa medida foi garantir que, apesar de concedido o acesso solicitado, a qualidade do software deveria ser mantida. O documento disponibilizado virtualmente orienta sobre o exercício das funções e sobre as formas como devem ocorrer as contribuições que envolvem o software.

Enquanto o CoGrOO não tinha processos bem definidos – o que só ocorreu após sua conexão ao OpenNLP – foram necessárias ações formalizadas em alguns momentos, especialmente quando o desenvolvimento do software corretor gramatical recebeu financiamento da *Finep* e quando foi utilizado em cursos e pesquisas realizadas no âmbito da USP. Nessas etapas, as várias forças-tarefa existentes tinham compromissos e cronogramas de entrega delimitados, supervisionados por professores.

Quando os projetos são considerados ao longo das suas fases de desenvolvimento, a formalização de comportamentos e resultados torna-se evidente em diversas situações. Assim, enquanto a pesquisa em software livre indica que essas iniciativas se destacam pela sua informalidade e pela pouca estruturação dos processos (Eseryel et al., 2020; Kolbjørnsrud, 2016; Lattemann & Stieglitz, 2005; Lindberg et al., 2016; Xu et al., 2005), esses projetos mostraram que, em alguns momentos, processos formais também foram adotados pelos coletivos, principalmente quando os projetos estabeleceram relacionamentos com outras organizações formais (Mäenpää et al., 2017; Viseur & Charleux, 2019; West & O’Mahony, 2005). Assim, projetos como o CoGrOO, em sua conexão com OpenNLP, Noosfero e Kytos, tiveram processos mais formalizados. São coletivos que integram fundações ou recebem patrocínios, o que exige uma definição clara de condutas e cronograma de entrega dos associados.

Os aspectos sociais foram relatados pelos entrevistados como procedimentos informais de controle. Ao promover a socialização dos membros do projeto, o uso de recursos de TIC possibilitou a criação de um espaço virtual comum para o trabalho colaborativo (Eseryel et al., 2020; Seidel & Stewart, 2011). Essa comunicação permite que membros heterogêneos direcionem seus esforços para os resultados. Embora a maior parte da comunicação ocorra virtualmente, reuniões presenciais para discussão de problemas do projeto também foram relatadas. O Noosfero realizou reuniões presenciais, especialmente no início, quando a maioria dos membros residiam na mesma cidade e o projeto era centrado na Colivre. Os desenvolvedores do CoGrOO que estudaram na USP, mencionaram vários momentos de interação face a face quando o software estava sendo desenvolvido na universidade. Como o Kytos utiliza a infraestrutura da Unesp, permite que alguns de seus integrantes se encontrem pessoalmente, o que muitas vezes envolve aqueles que foram contratados formalmente para o projeto. Oportunidades para reuniões presenciais no OpenNLP são raras e ocorrem principalmente no evento ApacheCon, uma conferência anual patrocinada pela Fundação Apache.

Os participantes também destacaram situações em que tiveram a oportunidade de interagir com atores externos aos projetos. Dentre elas, citaram a participação de membros em eventos técnicos e acadêmicos na área de software livre ou de atuação do projeto. Os participantes do Noosfero tiveram a oportunidade de dar palestras em reuniões sobre software livre. Os membros do Kytos, CoGrOO e OpenNLP destacaram participar de eventos para apresentação de trabalhos técnicos ou acadêmicos sobre os projetos. O Kytos, em particular, até promoveu edições de um evento denominado “Kytos Summit”, com palestras para participantes ou interessados no projeto.

Outra forma percebida de controle social está associada ao reconhecimento do trabalho realizado nos projetos, prática que está associada à meritocracia. Nesse aspecto, membros experientes que deram contribuições relevantes foram reconhecidos por seus pares e se destacaram por exercerem influência significativa nas decisões, conforme destacado por [Lattemann e Stieglitz \(2005\)](#) e [O'Mahony e Ferraro \(2007\)](#). Os entrevistados também relataram oportunidades externas de reconhecimento. Situações como reportagens, entrevistas, premiações e até momentos em que recursos significativos puderam ser captados foram citados como cruciais no reconhecimento do projeto e de seus principais representantes. Essa visão do reconhecimento como forma de controle social converge com o que é esperado em ambientes comunitários, pois é uma forma de motivação diferente por não utilizar recompensas formais ou benefícios privados ([Shaikh & Vaast, 2016](#)).

Essas experiências, por meio de interações internas ou externas, foram essenciais para a criação de um sentimento de pertencimento aos projetos ([Lattemann & Stieglitz, 2005](#); [Xu et al., 2005](#)), mesmo entre os indivíduos que atualmente estão afastados ou menos ativos. Ter essa referência e identidade influencia o autocontrole, especialmente entre participantes que fizeram ou ainda fazem contribuições como voluntários. Ao permitir autonomia e independência na execução do trabalho, os recursos tecnológicos também fornecem suporte para o autocontrole, reduzindo assim os esforços de coordenação no coletivo ([Crowston et al., 2012](#)).

Na revisão das contribuições feitas ao software, embora os projetos tenham regras para que as colaborações externas sejam revisadas pelos *committers*, quando essas contribuições vêm dos próprios *committers* a análise ocorre informalmente por meio de um processo de controle entre pares. Nesses casos, embora os *committers* tenham autonomia para escrita no repositório, uma prática comum tem sido fazer com que o código seja revisado por outro *committer* na comunidade, especialmente um desenvolvedor mais experiente. O Noosfero é o único projeto que tentou criar regras explícitas nesse sentido, mas com alguma flexibilidade. Se após uma semana o código não tiver sido revisado por nenhum par do coletivo, o *committer* que propôs o código tem autorização para adicioná-lo ao repositório. Os projetos investigados convergiram com a pesquisa em software livre no sentido de que a crítica ao trabalho alheio e a troca de experiências possibilitada pelas revisões ajudam a garantir a qualidade do software que é produzido ([Kolbjørnsrud, 2016](#); [Lee & Cole, 2003](#)).

## Análise baseada na contribuição organizacional conceitual e teórica

Além dos projetos estudados terem em comum o fato de trabalharem no contexto de uma instituição que oferece infraestrutura física e tecnológica, ao longo da sua história também tiveram relações com diferentes organizações, consideradas como partes interessadas ou *stakeholders* – entendendo que *stakeholders* são grupos ou pessoas que afetam ou podem ser afetados por uma organização ([Freeman, 2010](#); [Frooman, 1999](#)). Como principais *stakeholders* do CoGrOO estão a USP, a Finep, o Google e o projeto OpenNLP da Fundação Apache. Além da Colivre, o Noosfero contou com diversos *stakeholders*, como a USP, o Serpro, a UnB, o Governo Federal, entre outros. Os *stakeholders* do Kytos foram a Unesp, a Huawei, as instituições parceiras e outras comunidades.

Ao considerar que os agentes com os quais os projetos interagem constituem seus *stakeholders*, evidencia-se que há uma relação de influência mútua ([Frooman, 1999](#)). Essa influência recíproca foi observada no Noosfero, quando surgiram organizações paralelas que passaram a executar projetos utilizando o software para desenvolver websites alinhados aos seus interesses. Os indivíduos que participaram desses projetos e simpatizaram com o movimento do software livre devolveram voluntariamente o código que desenvolveram à comunidade Noosfero. Outro exemplo é o fato da Colivre, principal patrocinadora e executora dos contratos comerciais que utiliza o Noos-

fero, investir na manutenção da comunidade do projeto. As partes interessadas também foram influentes quando a Colivre teve que gerenciar conflitos com atores externos, que exigiam maior acesso da comunidade ao projeto.

Uma vez que a interação entre os projetos e as partes interessadas é uma realidade nos coletivos analisados, o contexto desempenha um papel crucial na alteração das condutas dos projetos (Pfeffer & Salancik, 2003). Nesse cenário, a TDR pode ser aplicada a projetos de software livre comunitários. A dinâmica percebida durante o desenvolvimento dos projetos investigados revelou que o principal motor do relacionamento com os *stakeholders* é a necessidade de obtenção de recursos para a sobrevivência do negócio. Foi necessário buscar recursos para garantir que os projetos continuassem a funcionar ao longo dos anos, seja por meio da infraestrutura fornecida por uma instituição, seja pelo apoio de uma fundação, patrocinador e /ou agência de financiamento. Exemplo disso é o Kytos, que além do apoio oferecido pelo NCC da Unesp, por meio dos contatos feitos pelo diretor do NCC, obteve patrocínio da Huawei, permitindo obter os recursos financeiros e tecnológicos necessários ao desenvolvimento do software.

Considerando a TDR, fica claro que os projetos não são apenas influenciados pelo ambiente, mas também têm uma postura proativa em relação a ele (Hillman et al., 2009). Os líderes dos projetos têm autonomia para usar várias táticas gerenciais com a intenção de garantir no ambiente os meios de que precisam para sobreviver (Sharif & Yeoh, 2014). A busca constante do CoGrOO por recursos ocorreu em vários momentos. Além do apoio da USP e do financiamento da Finep, a organização teve que buscar formas de garantir mais recursos ao longo de sua trajetória, o que incluiu o recebimento de recursos do programa Google *Summer of Code*. Mais tarde, quando um de seus desenvolvedores acabou se tornando o único responsável pelas operações do projeto, o OpenNLP se revelou como uma forma de o código do CoGrOO continuar recebendo a manutenção necessária. Como o OpenNLP era parte da Apache, o projeto obteve apoio indireto dessa fundação.

Outra associação possível entre iniciativas de software livre comunitárias é sua semelhança com o conceito de OTs. Conforme expressa a definição de OTs (Burke & Morley, 2016), os projetos caracterizaram-se por serem formados por atores interdependentes que direcionaram suas ações para a realização de tarefas complexas. O lançamento de versões de software é considerado o ponto de encerramento da OT, que ocorre em consequência de uma definição de data específica pela organização, ou quando se atinge um estado ou condição previamente definidos (Bakker, 2011; Bakker et al., 2009). Outra forma de perceber as iniciativas de software livre como OTs é por meio da associação dos projetos com diferentes *stakeholders*. Nessas associações, ocorreram ajustes estruturais e dos processos de controle nos projetos, delineando um arranjo organizacional temporário que se modificava conforme surgiam novos relacionamentos com *stakeholders*. Outra característica que torna os projetos investigados mais parecidos com OTs é o fato de a liderança ter se mostrado uma função de fundamental relevância no direcionamento do trabalho colaborativo (Bakker, 2011).

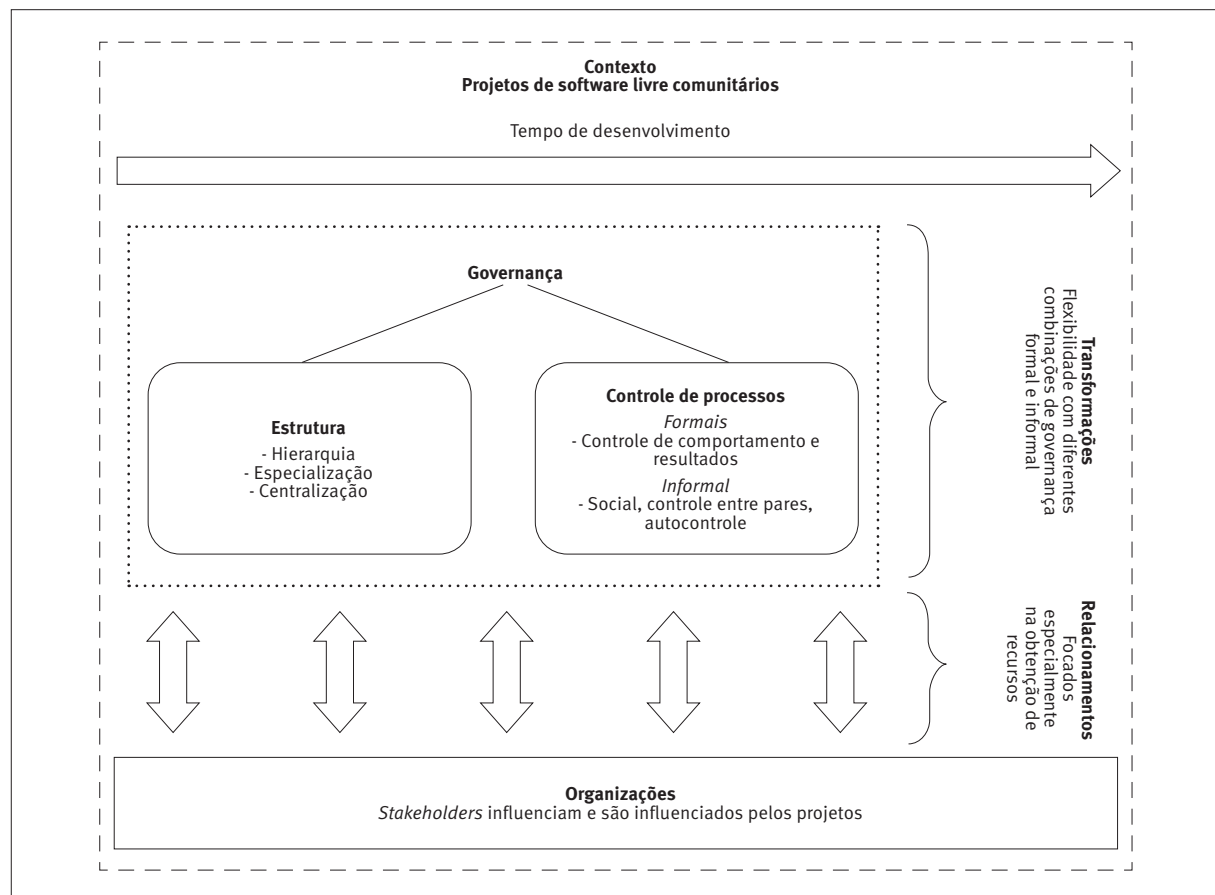
## CONCLUSÕES

Ficou evidente neste estudo a dificuldade de fazer a separação entre as comunidades e as iniciativas formais desenvolvidas por essas estruturas no âmbito dos projetos. Por esse motivo, é relevante que a pesquisa sobre governança considere todo o contexto e os relacionamentos estabelecidos pelos projetos comunitários. A governança (estrutura e processos de controle) mostrou-se um conceito flexível e mutante, que se adaptou aos caminhos de desenvolvimento percorridos pelos projetos investigados. Ao longo dessas mudanças, ocorreram procedimentos de governança formais e informais.

Mesmo que os traços comunitários tenham persistido durante o desenvolvimento dos projetos, a gestão foi formalizada em determinados momentos, principalmente devido às relações estabelecidas com atores organizacionais externos. A análise desses projetos indicou que os empreendimentos de software livre precisaram estabelecer contatos com *stakeholders* a fim de obter os recursos financeiros, materiais e humanos necessários para sua sobrevivência. A liderança demonstrou ser fundamental para encontrar os recursos no ambiente e orientar os negócios.

Ainda que a literatura especifique as fases de desenvolvimento pelas quais passam os projetos de software livre e defina características válidas que diferenciam esses projetos das organizações tradicionais, na prática essas situações não ocorreram de forma bem definida. Embora as facetas informais da comunidade tenham sido mantidas, as transformações que levaram ao desenvolvimento dos projetos foram associadas aos relacionamentos para obtenção de recursos desenvolvidos com outras organizações. A governança formal nessas relações às vezes era necessária e coexistia com a informalidade dos espaços comunitários. Ao invés das fases de desenvolvimento ocorrerem numa perspectiva de crescimento contínuo, os projetos vivenciaram momentos de maior e menor atividade que foram regidos pelos recursos obtidos de organizações formais. Esses projetos não se encaixaram inteiramente em nenhum dos conceitos da discussão bazar versus catedral. Eles apresentaram semelhanças com organizações tradicionais, mantendo as características da comunidade. Assim, se revelaram misturas intrincadas dos dois tipos de organização.

Figura 1. Ideia central da pesquisa



As limitações deste artigo incluem a escolha de unidades de análise com base na disponibilidade dos projetos e o fato de que os resultados refletem o contexto investigado das comunidades de software livre. Em pesquisas futuras, recomenda-se o uso de abordagens metodológicas quantitativas ou mistas, bem como o estudo de outros tipos de organização comunitária. Em relação às suas contribuições, este estudo oferece aos pesquisadores e profissionais uma compreensão mais realista da governança dos negócios comunitários e das táticas que os líderes usam para obter os recursos necessários a garantia da sobrevivência de seus projetos.

## REFERÊNCIAS

- 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. Recuperado de <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. Mizuchi & 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. Recuperado de <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.voio.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. Recuperado de [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=2822679](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. Recuperado de [https://ideas.repec.org/a/cai/inndbu/inno\\_058\\_0071.html](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. Recuperado de <https://1library.net/document/ye8v26ry-e-business-requirements-flexibility-implementation-enterprise-review.html>

## CONTRIBUIÇÃO DE AUTORIA

Os autores declaram que participaram de todas as etapas do desenvolvimento do artigo. A Isabela Neves Ferraz ajudou na preparação da pesquisa e na análise dos dados, enquanto o Carlos Denner dos Santos participou na orientação de todo o processo de pesquisa, revisão do texto e fazendo as adequações necessárias.