



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA - CCET
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA - PPGI

**POWER RELATIONS WITHIN
OPEN SOURCE SOFTWARE ECOSYSTEMS**

Victor Garcia Farias

Orientador

Rodrigo Pereira dos Santos

RIO DE JANEIRO, RJ - BRASIL

MARÇO 2022

POWER RELATIONS WITHIN
OPEN SOURCE SOFTWARE ECOSYSTEMS

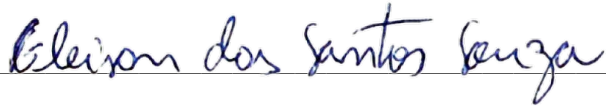
Victor Garcia Farias

DISSERTAÇÃO DE MESTRADO APRESENTADA COMO REQUISITO PARCIAL PARA OBTENÇÃO DO TÍTULO DE MESTRE PELO PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA DA UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO(UNIRIO). APROVADA PELA COMISSÃO EXAMINADORA ABAIXO ASSINADA.

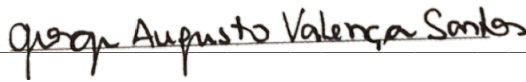
Aprovada por:



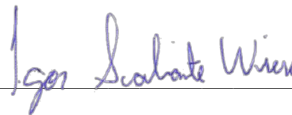
Rodrigo Pereira dos Santos, D.Sc. - UNIRIO



Gleison dos Santos Souza, D.Sc. - UNIRIO



George Augusto Valença Santos, D.Sc. - UFRPE



Igor Scalante Wiese, D.Sc. - UTFPR

RIO DE JANEIRO, RJ - BRASIL

MARÇO 2022

Catálogo informatizada pelo(a) autor(a)

F224	Farias, Victor Garcia Power Relations within Open Source Software Ecosystems / Victor Garcia Farias. -- Rio de Janeiro, 2022. 195 f. Orientador: Rodrigo Pereira dos Santos. Dissertação (Mestrado) - Universidade Federal do Estado do Rio de Janeiro, Programa de Pós-Graduação em Informática, 2022. 1. Power Relations. 2. Open Source Software. 3. Software Ecosystem. 4. Qualitative Analysis. 5. Systematic Mapping Study. I. Santos, Rodrigo Pereira dos, orient. II. Título.
------	--

Acknowledgements

First, I thank my family for their support not only during the period of this research but throughout my entire life. To my father Cláudio and to my mother Carmen who raised me along with my sister Júlia and my brother Gabriel and always encouraged us towards achieving our goals, specially when it comes to education.

To my advisor, Rodrigo Pereira dos Santos, who was always there to guide me with all of his patience. Thank you for the guidance, motivation and understanding. Also, thank you for tolerating me for so long (I know it is not an easy task).

To Igor Wiese, who, despite of role names or rules, I always considered as my co-advisor. Thank you for the guidance and the suggestions regarding the research and also for your patience.

Alexander Serebrenik and Eleni Constantinou, thank you for the collaboration in the questions' design and the very inspiring talks we had for the last months. I am looking forward to collaborate and learn a lot more from both of you.

To Alex Vaz and Roberto Dias who collaborated to the planning of the studies reported in Chapter 3 and Chapter 4. Thank you guys for the partnership in those moments.

To all of the Complex Systems Engineering Laboratory members. I learned a lot from all of you during your presentations throughout the period of this research. Thank you for sharing your knowledge.

Finally, to all my friends who I should not only thank for the support but also apologize for my absence during this research period. To Marcus Santos, Larissa Lima, Marina Braga, Carol Galli, Letícia Garcez, Daniela Rocha, Igor Balteiro, Rodolfo Gonçalves, Kaury Miranda, Gabriela Barboza and Alessandra Mattozinhos: thank you for your friendship.

To all of you: thank you so much!

FARIAS, Victor. **Relações de Poder em Ecossistemas de Software Aberto**. UNIRIO, 2022. 195 páginas. Dissertação de Mestrado. Programa de Pós-Graduação em Informática, UNIRIO.

Resumo

Relacionamentos em ecossistemas de software aberto (do inglês, *Open Source Software Ecosystems* ou OSSECO) emergem da colaboração em um ecossistema. Relacionamentos de poder (RP) estão presentes neste contexto sempre que uma entidade tem o poder de fazer outras entidades agirem como ela desejar. Sendo assim, essas RP poderiam afetar a colaboração em um OSSECO. Essa pesquisa visa investigar as RP, seus benefícios e desafios, e fornecer um entendimento sobre elas dentro do OSSECO. Adicionalmente, esta informação deve ser apresentada em um modelo conceitual. Um mapeamento sistemático da literatura (MSL) foi conduzido para extrair informações sobre RP de estudos existentes. Baseado nos resultados deste MSL, um estudo qualitativo baseado em entrevistas assíncronas foi conduzido com uma amostra de membros da comunidade do OSSECO do npm (Node Package Manager). Depois disso, um estudo de campo foi conduzido para refinar a informação levantada nos estudos anteriores. Ao final, usando os resultados obtidos, dois modelos conceituais foram construídos para representar as RP em OSSECO. Essa pesquisa define as diferentes formas de RP no contexto de OSSECO, um conjunto de 19 dinâmicas de RP, 14 entidades e 34 ações envolvidas com estas dinâmicas, um modelo conceitual baseado nas entidades (OSSECO-PREM) e um modelo conceitual baseado em formas de RP (OSSECO-PReFoM). A identificação de RP em um OSSECO habilitariam aqueles que estudam ou são membros da comunidade de um OSSECO a incentivar RP que estimulam a colaboração e evitar aquelas que podem levar desenvolvedores a abandonar o OSSECO.

Palavras-chave: Relacionamentos de Poder, Software Aberto, Ecossistema de Software, Análise Qualitativa, Mapeamento Sistemático da Literatura

FARIAS, Victor. **Power Relations within Open Source Software Ecosystems**. UNIRIO, 2022. 195 pages. Master's Thesis. Graduate Program in Informatics, UNIRIO.

Abstract

Relationships within open source software ecosystems (OSSECO) emerge from the collaboration within the ecosystem. Power relationships are present in this context whenever an entity has the power of making other entities act as it wants them to act. Therefore, these power relationships could affect collaboration within an OSSECO. This research aims at investigating power relationships, their benefits and challenges, and providing an understanding of them within OSSECO. In addition, this understanding should be presented in a conceptual model. A systematic mapping study (SMS) was conducted to extract information about power relationships from existing studies. Based on this SMS, a qualitative study based on asynchronous interviews with a sample of npm (Node Package Manager) OSSECO community members is conducted. Thereafter, a field study was conducted to refine the information gathered in previous studies. Finally, from the results of the previous studies, two conceptual models were constructed to represent the power relationships within an OSSECO. This research defines the different forms of power relationships within the OSSECO context, a set of 19 power relationship dynamics, 14 entities involved with these dynamics, 34 actions related to those dynamics, an entity-based conceptual model (OSSECO-PREM) and a power forms-based conceptual model (OSSECO-PReFoM). These results would enable those who study or are members of the ecosystem's community to enhance power relationships that support collaboration and to avoid those who can lead developers to leave the OSSECO.

Keywords: Power Relationships, Open Source Software, Software Ecosystem, Qualitative Analysis, Systematic Mapping Study

Contents

List of Figures	xi
List of Tables	xiv
1 Introduction	1
1.1 Context	1
1.2 Motivation	2
1.3 Problem	3
1.4 Objective	5
1.5 Organization	7
2 Research Methodology	9
2.1 Introduction	9
2.2 Systematic Mapping Study	9
2.3 Exploratory Study	10
2.4 Field Study	11
2.5 Model Construction	13
2.6 Final Remarks	14
3 Background	16
3.1 Introduction	16
3.2 Open Source Software Ecosystems	16
3.3 The npm and its Ecosystem	20
3.4 Power Relationships	21
3.5 Related Work	23
3.6 Glossary	25
3.7 Final Remarks	26

4	An Overview Power Relation within Open Source Software Ecosystems	28
4.1	Introduction	28
4.2	Planning	30
4.3	Execution	32
4.4	Results	32
4.5	Analysis	35
4.5.1	RQ1: What are the power relationships and their dynamics within an OSSECO?	36
4.5.2	RQ2: What are the benefits those power relationships can bring into an OSSECO?	38
4.5.3	RQ3: What are the challenges those power relationships can bring into an OSSECO?	40
4.6	Limitations	41
4.7	Final Remarks	41
5	Power Relations within the npm Open Source Software Ecosystem in Practice	43
5.1	Introduction	43
5.2	The Chosen OSSECO and Protocol Decisions	43
5.3	A Preliminary Study on the npm OSSECO	45
5.3.1	Introduction	45
5.3.2	Planning	46
5.3.3	Execution	50
5.3.4	Results	50
5.3.5	Analysis	54
5.3.6	Limitations	65
5.3.7	Conclusion	66
5.4	Refining the npm OSSECO Community's Opinion on Power Relationships	67
5.4.1	Introduction	67
5.4.2	Planning	69
5.4.3	Execution	73

5.4.4	Results	74
5.4.5	Analysis	88
5.4.6	Relationships Between Power Relationship Forms	108
5.4.7	Limitations and Threats to Credibility	111
5.4.8	Conclusion	112
5.5	Final Remarks	112
6	The OSSECO Power Relations Conceptual Models	114
6.1	Introduction	114
6.2	Planning	115
6.3	Describing the Power Relationship Dynamics	123
6.3.1	Entities Involved with Power Relationships Within an OSSECO	123
6.3.2	Actions that Could Be Performed by the Entities	125
6.3.3	Power sources	132
6.3.4	The Power Relationship Dynamics	133
6.4	OSSECO Power Relationships Entity Model (OSSECO-PREM) . . .	155
6.5	OSSECO Power Relationships Forms Model (OSSECO-PReFoM) . .	160
6.6	Limitations	162
6.7	Final Remarks	162
7	Discussion	164
7.1	Introduction	164
7.2	Power Relationships Within an OSSECO and the Literature	164
7.3	The Perspective on Power Within an OSSECO	165
7.4	Power and Automation in OSSECO	165
7.5	Final Remarks	166
8	Conclusion	167
8.1	Epilogue	167
8.2	Contributions and Implications	170
8.3	Publications	173
8.4	Threats to Validity	173
8.4.1	Internal Validity	173
8.4.2	External Validity	174

8.4.3 Reliability	174
8.5 Future Work	175
References	177
Appendices	182
I Exploratory Study	183
I.1 Invitation	183
I.2 Terms and Conditions	185
I.3 Demographic Questions	186
I.4 Technical Questions	187
II Field Study	189
II.1 Invitation	189
II.2 Terms and Conditions	191
II.3 Demographic Questions	193
II.4 Eindhoven University of Technology's Ethical Review Board Accep- tance Letter	194

List of Figures

Figure 2.1	Research methodology.	15
Figure 3.1	Software Ecosystem Scope Levels (BOUCHARAS et al., 2009).	17
Figure 3.2	Three dimensional view of SECO E&M (SANTOS et al., 2012).	17
Figure 3.3	An “onion” diagram representing a generic OSSD project or- ganizational hierarchy (JENSEN et al., 2007).	19
Figure 3.4	An OSSECO that has a programming language as its techno- logical platform.	19
Figure 3.5	Power Relationship according to Foucault (FOUCAULT, 1977).	21
Figure 3.6	Power Relationship forms conceptual model (LEONIDOU et al., 2014).	23
Figure 3.7	PRM-SECO coarse-grained view (VALENÇA et al., 2017).	24
Figure 4.1	Systematic mapping study methodology.	30
Figure 4.2	Studies selection process.	31
Figure 4.3	Selected studies timeline.	33
Figure 5.1	Exploratory study methodology.	46
Figure 5.2	Agreement with proposed definition in TQ1.	52
Figure 5.3	Motivations to contribute to an OSSECO.	54
Figure 5.4	The axial coding regarding the Dynamics.	57
Figure 5.5	The axial coding regarding the Entities.	61
Figure 5.6	The axial coding regarding the Actions.	65
Figure 5.7	Field study methodology.	68
Figure 5.8	The axial coding regarding the Dynamics.	90
Figure 5.9	The axial coding regarding the Entities.	97
Figure 5.10	The axial coding regarding the Actions.	101

Figure 5.11	The axial coding regarding the Factors.	104
Figure 6.1	Model construction methodology.	115
Figure 6.2	Entities within an OSSECO, part 1.	118
Figure 6.3	Entities within an OSSECO, part 2.	118
Figure 6.4	Entities within an OSSECO, part 3.	118
Figure 6.5	The technological platform entity symbol for this study: the npm logo.	119
Figure 6.6	Connectors used in the graphical representation.	119
Figure 6.7	An example of a graphical representation of a power relation- ship dynamic.	120
Figure 6.8	An example of a entity-based model.	121
Figure 6.9	The connectors for the power relationship forms model.	122
Figure 6.10	The example of how the power relationship forms model should look like.	122
Figure 6.11	Conforming to stakeholders' requirements refined.	135
Figure 6.12	Control over technological platform refined.	136
Figure 6.13	Dependency between projects refined.	137
Figure 6.14	Employment relation refined.	139
Figure 6.15	Learning reference refined.	140
Figure 6.16	Newcomers contribution rejection refined.	141
Figure 6.17	Promoting collaboration refined.	142
Figure 6.18	Pull request evaluation refined.	143
Figure 6.19	Role migration within the OSSECO refined.	145
Figure 6.20	Sponsoring refined.	146
Figure 6.21	Technical leadership refined.	147
Figure 6.22	Blocking refined.	149
Figure 6.23	Project forking.	150
Figure 6.24	Dependencies of dependency.	150
Figure 6.25	Disowning project.	152
Figure 6.26	Community building.	153
Figure 6.27	Conference organizing.	153
Figure 6.28	Foundation organizing.	154

Figure 6.29 Board organizing.	155
Figure 6.30 OSSECO Power Relationships Entity Model (PREM) dynam- ics view.	157
Figure 6.31 OSSECO Power Relationships Entity Model (OSSECO-PREM) triggers view.	158
Figure 6.32 OSSECO Power Relationships Entity Model (OSSECO-PREM) complete view.	159
Figure 6.33 OSSECO Power Relationship Forms Model (OSSECO-PReFoM).	161
Figure I.1 Asynchronous interview participation Terms and Conditions. .	185
Figure I.2 Asynchronous interview demographic questions (DQs). . . .	186
Figure I.3 Asynchronous interview technical questions (TQs) part 1. . .	187
Figure I.4 Asynchronous interview technical questions (TQs) part 2. . .	188
Figure II.1 Interview’s demographic questionnaire.	193
Figure II.2 Ethical Review Board acceptance letter, page 1.	194
Figure II.3 Ethical Review Board acceptance letter, page 2.	195

List of Tables

Table 3.1	Power relationships forms.	22
Table 4.1	Systematic mapping study inclusion criterion (IC).	31
Table 4.2	Systematic mapping study exclusion criteria (EC).	31
Table 4.3	Selected Studies.	34
Table 4.4	Power relationship forms and the adaption to the OSSECO context.	35
Table 4.5	Selected studies and the RQ they help to answer.	36
Table 4.6	Power relationship dynamics extracted from SMS.	36
Table 4.7	Power relationships benefits to the OSSECO.	39
Table 4.8	Power relationships challenges to the OSSECO.	40
Table 5.1	The GQM for the exploratory study goals.	49
Table 5.2	Respondents answers to the demographic questions.	51
Table 5.3	Codes identified during the open coding step.	55
Table 5.4	The identified power relationship dynamics and their power relationship forms.	60
Table 5.5	The entities identified within an OSSECO.	63
Table 5.6	Interviewees answers to the demographic questions, part 1. . .	74
Table 5.7	Interviewees answers to the demographic questions, part 2. . .	75
Table 5.8	Codes identified during the open coding step.	89
Table 5.9	The identified power relationship dynamics and their power relationship forms.	96
Table 5.10	The entities identified during the field study.	101
Table 5.11	The actions identified during the field study.	102
Table 5.12	The factors identified during the field study.	105

Table 5.13 The relationships between power relationship forms within an
OSSECO. 111

Table 6.1 The actions that can be performed within power relationship
dynamics. 129

Chapter 1. Introduction

This chapter presents the context, motivation, problem and objectives. We also present the structure of this dissertation.

1.1 Context

In Open Source Software (OSS) development, a group of developers who share a common problem or development goal gathers in a project to achieve those goals together (ALMARZOUQ et al., 2015). This group of developers is responsible not only for developing the software solution but also for making decisions about the development (e.g., which technology to use, rules for collaborating, which members would have more privileges) (MARSAN et al., 2019). These OSS community members organize themselves and interact with each other to achieve their objectives by their development (ALMARZOUQ et al., 2015). Although OSS development is open and collaborative, these communities have their hierarchy-like organization (JENSEN et al., 2007).

When a set of open source projects share the same technology platform and have interfaces between them, having a flow of developers through the projects to develop them, we have an Open Source Software Ecosystem (OSSECO) (FRANCO-BEDOYA et al., 2017). In this context, the OSS communities related to the standalone projects also start to present interfaces between them and to share members, creating an OSSECO community (FRANCO-BEDOYA et al., 2017). Given the size that some OSSECO can achieve (ANGEREN et al., 2011), there are cases in which organizations play the role of those ecosystems managers to help them evolve properly (PLAKIDAS et al., 2016).

In those OSSECO, there are different interactions and relationships that are important to the ecosystem's evolution (LINÅKER et al., 2020b). These interactions

are not only related to the technical matter but also the social one (DUCHENEAUT, 2005; TSAY et al., 2014; FARIAS et al., 2019). As different roles can be played by those who are part of an OSSECO, it is possible that power relationships directly affect (either positively or negatively) the interactions in the context and, therefore, OSSECO's evolution (FRANCO-BEDOYA et al., 2017).

Those interactions can be observed in the platforms in which the OSSECO members gather to develop their projects and discuss their next moves (FRANCO-BEDOYA et al., 2017). In the npm OSSECO (CONSTANTINO et al., 2017), developers use GitHub to host their source code and interact with each other and with their code. GitHub is an example of this type of platform. It is a platform where developers can share their codes and work together in their software solutions. However, it also supports social interactions such as commenting, following other developers, and opening issues (a technical problem alert in a format that is similar to a forum post) (FARIAS et al., 2019).

1.2 Motivation

In the early years of open source, the motivation for contributing to an ecosystem would be mostly voluntary (ALMARZOUQ et al., 2015). However, as OSSECO became popularized and organizations decided to be part of them, developers expected results other than simply having a software solution to satisfy their necessities. Therefore, other motivation reasons have been more common lately, representing a change in the developers' relation with the OSSECO as a whole (GEROSA et al., 2021).

The change in the developers' motivations for collaborating with the OSSECO could represent that the situations in which interactions were involved in OSSECO could also be somehow modified. Therefore, power relationships would be more frequent in this scenario, affecting the OSSECO positively and negatively (FARIAS et al., 2019). An example of how motivations are affected by the power relationships can be seen in the fact that employment relation (an example of power relationship) influences developers to collaborate in OSSECO (GEROSA et al., 2021). Another example can be identified in the fact that organizations that are part of the OSSECO attend to other organizations' requirements (acting as those wish) to achieve business

goals (LINÅKER et al., 2020a).

Moreover, understanding those power relationships could help organizations who are part of the OSSECO to achieve their business goals. Since those organizations usually have a financial power attached to them, they could use their power to support the OSSECO and the projects that are interesting to those organizations.

Additionally, those who maintain projects within the OSSECO or contribute with projects within the OSSECO could use the information about power to enhance their experience within the OSSECO and help them to achieve their goals. An example could be related to a maintainer that has the goal of working full time with his/her project. He/she could pursuit the sponsorship of an organization that uses his/her project since they have financial power and to benefit from this relationship.

Therefore, an OSSECO that has an active community and have public data that we could use to either confirm our results or to explore would be necessary for this research. The npm OSSECO (CONSTANTINO et al., 2017) is an OSSECO that fits this description and that can be used in this investigation.

1.3 Problem

Although power relationships may currently affect OSSECO, not many studies address this subject regarding the OSSECO context. Power relationships are relationships in which an actor that has something to offer exercises power over another actor that desires that something making the second actor act as the first desires (FOUCAULT, 1977). As an example of what that “something” would be in OSSECO, we can relate it to the role migration possibility that one actor can provide to another (JENSEN et al., 2007). The lack of an investigation about those power relationships, especially related to in which situations they appear and how they affect the developers within an OSSECO, leaves a set of topics yet to be discussed:

- *It is not clear what are the power relationships present within an OSSECO*

There are previous power relationships studies in different contexts. For example, in one of these contexts, an author suggested a categorization of power relationships, called power relationship forms (LEONIDOU et al., 2014). However, in an OSSECO context, there is no previous study identifying the power relationships in it.

- *It is not clear in what situations power relationships take place within an OSSECO*

Although knowing the power relationships in an OSSECO could help those who are part of it, identifying the situations in which those power relationships take place could help them even more. Moreover, identifying those situations could provide more detailed information, enabling those who are part of an OSSECO to understand them better and plan more accurate strategies (ALVES et al., 2019).

- *It is not clear who are the actors involved with power relationships within an OSSECO*

In an OSSECO, different actors can play different roles across the ecosystem. Usually, the roles are related to the level of privileges actors have in this context (JERGENSEN et al., 2011). Therefore, understanding how each role relates to the power relationships and their situations could provide more detailed information about how and why those relationships occur within an OSSECO.

- *It is not clear how power relationships would positively affect an OSSECO*

Since the power relationships could be directly related to the motivation for collaborating with an OSSECO, there is a possibility that those relationships could enhance collaboration and bring innovation into the ecosystem (LINÅKER et al., 2020a). Therefore, investigating how those power relationships positively affect the collaboration could enable those who are part of OSSECO to support those relationships, bringing more innovation and collaboration into the OSSECO.

- *It is not clear how power relationships would negatively affect an OSSECO*

Although the power relationships could positively affect an OSSECO, those could also represent a threat to the OSSECO. A negative impact that an OSSECO could suffer from the power relationships is if any of those relationships represent a barrier for collaboration and innovation that could lead developers to abandon an OSSECO (FARIAS et al., 2019). In those cases, developers

could feel motivated to seek collaboration with a different OSSECO. The OSSECO could maintain innovation and collaboration incoming by avoiding those power relationships.

The lack of information about those questions makes it more difficult for those part of an OSSECO to identify potential threatening relationships to the OSSECO. Moreover, they also have the problem of not identifying which relationships they should support to enhance collaboration and bring more innovation into an OSSECO.

1.4 Objective

This research aims at providing an understanding of power relationships within the OSSECO context. Therefore, it is a goal to identify: (i) the types of power relationships; (ii) the situations in which they take place; (iii) those who are involved with those situations; and (iv) how the power relationships affect the OSSECO (both positively and negatively).

A research question was defined to guide the research. In addition, were defined five secondary questions to help organize the information gathered in this research and have a more precise answer to our primary question. Our primary research question (PRQ) is:

PRQ. How do power relationships within an open source software ecosystem affect the projects' developers, artifacts, and relationships?

Power relationships are relationships in which one actor has something to offer, and the other actor desires that specific something, creating a dependency relationship between them (FOUCAULT, 1977). In the context of an OSSECO, in which interactions between those who are part of it are essential to maintain the OSSECO (LINÅKER et al., 2020b), power relationships could affect (either positively or negatively (ALVES et al., 2019)) those interactions and, therefore, affect the OSSECO's evolution.

Moreover, to help organize the information about power relationships within an OSSECO, the five secondary questions (SQ) are directly related to the topics reported in the Problem section:

- SQ1. What are the forms of power relationships within an OSSECO?

Rationale: Previously, studies investigated power relationships forms in different contexts, including other software ecosystems. There would be even relationships between the different power relationships forms, and they would affect each other (VALENÇA et al., 2017). The purpose is to identify what power relationships forms take place within an OSSECO and how they affect each other.

- SQ2. In what situations those power relationships take place within an OSSECO?

Rationale: Since OSSECO is an interactive and collaborative environment, it might be possible that different situations happen within it regarding either technical or social matters (FARIAS et al., 2019). Therefore, understanding those situations could give more details about how the power relationships occur within an OSSECO.

- SQ3. Who are the actors that participate in those power relationships within an OSSECO?

Rationale: Although OSSECO aims at having a collaborative hierarchy-free community environment, there are different roles within an OSSECO (JERGENSEN et al., 2011). Thus, understanding how those different actors who play different roles participate in the power relationships would give those interested in an OSSECO to identify the power holders in this context.

- SQ4. How those power relationships positively affect an OSSECO?

Rationale: Since the interactions and the collaboration is essential to the OSSECO maintenance (LINÅKER et al., 2020b), and since power relationships could affect those interactions, it would be interesting to understand how and what power relationships positively affect an OSSECO. Therefore, those interested in the OSSECO would support those power relationships that somehow bring benefits to the OSSECO.

- SQ5. How those power relationships negatively affect an OSSECO?

Rationale: Following the same rationale from SQ4, the opposite could also happen: power relationships could negatively affect the interactions and collaboration within an OSSECO. Therefore, knowing how this adverse effect takes place and the power relationships responsible for this effect would enable those interested in the OSSECO to avoid those power relationships.

Moreover, as a second objective, a model is built to combine all the information about power relationships within OSSECO gathered during this research. The expectation is that this understanding, combined with the model provided, helps those who are part of the OSSECO (both in a managerial position or members) to use the power relationships to support the OSSECO evolution.

1.5 Organization

This dissertation is organized as detailed next.

Chapter 2 presents this research's methodology, explaining all the phases and studies involved in each one of them.

Chapter 3 presents the concepts of OSSECO and power relationships. It also presents the npm ecosystem and previous power relationships investigations in Computer Science and in other areas. Finally, this chapter presents a glossary containing expressions related to this research and their definition.

Chapter 4 presents a systematic mapping study that investigates what exists in the literature about power relationships within OSSECO. It was possible to propose a power relationships forms categorization for the context as well as to identify 5 of those power relationships forms and 7 power relationships dynamics. We also identify the benefits and challenges that power relationships could bring into an OSSECO.

Chapter 5 presents two studies: (i) a exploratory study that aims to understand how an OSSECO community would perceive power relationships in their context; and (ii) a field study in which the objective was to refine the information gathered in this research until this point. In the exploratory study, it was possible to identify new power relationships dynamics, motivations for collaborating with the OSSECO, and the benefits and challenges of power relationships. Thereafter, in the field study,

it was possible to identify more details about power relationships dynamics and their effects in the OSSECO context.

Chapter 6 presents building a model that would combine all the information about power relationships gathered until this point of the research. It was possible to propose a description and graphic representation for each of the power relationships dynamics previously identified. We also proposed an entity-based model and a relationships between power relationships based model to help understand the relationships between the entities present in an OSSECO.

Chapter 7 presents a discussion considering this research results and the literature. This research implications are also discussed in this chapter.

Chapter 8 presents the conclusion and contributions. We also describe the research limitations and discuss the future work.

Appendices present artifacts and samples of artifacts that we used in the exploratory study and in the field study.

Chapter 2. Research Methodology

2.1 Introduction

In this chapter, we present the research methodology designed for this research and each one of its phases. We also present the studies that are part of each of the phases.

To achieve this research's objective, the research methodology was planned in five phases, as shown in Figure 2.1 and explained in the following sections.

The remainder of this chapter is structured as follows: Section 2.2 presents the systematic mapping study; Section 2.3 presents the exploratory study; Section 2.4 presents the field study; Section 2.5 presents the model construction; and Section 2.6 presents the final remarks.

2.2 Systematic Mapping Study

This study aims at identifying what power relationships forms were present within an OSSECO. It also has the objective of identifying which dynamics those power relationships occur. Finally, identifying the benefits and challenges that those power relationships would bring into an OSSECO is also considered an objective of this phase. To conduct this study, we follow Kitchenham & Charters' guidelines (2007).

Therefore, to help this study to achieve its objective, three research questions were defined (SMSRQ):

- *SMSRQ1: What are the power relationships and their dynamics within an OSSECO?*
- *SMSRQ2: What are the benefits those power relationships can bring into an OSSECO?*

- *SMSRQ3: What are the challenges those power relationships can bring into an OSSECO?*

Moreover, this study is composed by three phases:

- *Planning*: in this phase, the objective was to define the protocol used in this study. Therefore, we defined the digital libraries to run the search on, the search string and its terms, the exclusion criteria and inclusion criterion, and other protocol matters, following Kitchenham & Charters' guidelines (2007). At the end of this phase, we expect to have a defined protocol to run this search. The author was involved in this phase together with his mentor and two other students from the Scientific Methodology I course at the Graduate Program in Informatics, Federal University of the State of Rio de Janeiro (UNIRIO);
- *Execution*: in this phase, the objective was to run the search, select the studies that would answer the research questions, and extract the data from these studies. At the end of this phase, we expect to have a set of selected studies that could answer the research questions and to have the results and data extracted reported. The author is responsible for extracting the data and two experts are involved with the evaluation of the results;
- *Analysis*: in this phase, the objective was to analyze the data extracted from the selected studies and answer the research questions using the extracted data. At the end of this phase, the research questions should be addressed appropriately and answered. The author is responsible for analyzing the data and two experts are involved with the evaluation of this analysis.

2.3 Exploratory Study

This phase aims at identifying the opinion of a portion of developers who are part of the npm OSSECO regarding the power relationships within this ecosystem. Their motivations, perspectives about how the power relationships occur in the npm OSSECO and evaluation of a power relationship definition were collected. The planning and execution steps of this phase followed Kitchenham et al.'s guidelines (2015).

During the analysis, we use open and axial coding (STRAUSS et al., 2007) to identify the power relationships forms, power relationships dynamics, actors involved with the power relationships, and actions that those actors could perform in a power relationship dynamic.

Therefore, to help this study to achieve its objective, three research questions were defined (ESRQ):

- ESRQ1: What are the power relationship forms within an OSSECO?
- ESRQ2: In what situations those power relationships take place in an OSSECO?
- ESRQ3: Who are the entities that participate in those power relationships within an OSSECO?

Moreover, this study is composed by three phases:

- *Planning*: this phase had the objective of defining what would be the OSSECO that would be studied and what would be the interview protocol, following Kitchenham et al.'s guidelines (2015). The author is involved in this phase together with two experts who help design the protocol;
- *Execution*: this phase had the objective of running a pilot, randomly selecting the asynchronous interview potential respondents, inviting them to participate, and collecting their answers. The author is the only person involved with this phase;
- *Analysis*: this phase had the objective of answering the research questions for this study based on the results of the asynchronous interviews. Therefore, to extract the information from the asynchronous interview responses, we used open and axial coding (STRAUSS et al., 2007). The author is responsible for gathering the data and analyzing it while two experts evaluate the findings.

2.4 Field Study

This study aims at gathering information about power relationships within an OSSECO. Unlike the exploratory study phase, we conducted live interviews with the

npm OSSECO community members. Therefore, it was possible to ask follow-up questions and make a more specific investigation. The analysis used the same procedures as the previous interviewing phase.

Therefore, to help this study to achieve its objective, three research questions were defined (FSRQ):

- FSRQ1: What are the power relationship forms within an OSSECO?
- FSRQ2: In what situations those power relationships take place in an OSSECO?
- FSRQ3: Who are the entities that participate in those power relationships within an OSSECO?

Moreover, this study is composed by three phases:

- *Planning*: this phase had the objective of defining what would be the interview protocol, following Kitchenham et al.'s guidelines (2015). It was also defined how the potential interviewees would be selected. The npm OSSECO was already selected as this would be a refining study regarding what was previously discovered in Section 5.2. However, it was also necessary to define who would be the npm OSSECO community members that would be reached out to participate in this field study. The author was involved in this phase together with 4 experts who helped to design the protocol;
- *Execution*: this phase had the objective of executing a pilot and then executing the interviews with the npm OSSECO community members. After the pilot, possible adjustments in the interview protocol would be made prior to the interview with OSSECO community members execution. The author was the only person involved with this phase;
- *Analysis*: this phase had the objective of analyzing the data that was gathered in the interviews and extracting information from them. That information should be used to answer the research questions that were defined for this study. In this phase, the information gathered is also compared to the information that was gathered in the exploratory study reported in Section

5.2. The author is responsible for gathering and analyzing the data while the 4 experts evaluate the results and analysis.

2.5 Model Construction

The objective of this phase is to build a model that would represent all the information regarding power relationships, power relationships forms and power relationships dynamics. Since the previous studies identified information about power relationships within an OSSECO, in this phase, results of the previous studies are combined and used to construct a textual representation, a conditional representation and a graphical representation for the power relationships dynamics. After that, those power relationships dynamics are combined to construct a conceptual model representing power relationships between entities within an OSSECO, the OSSECO-PREM. A conceptual model based on the relationships between the different power relationships forms (OSSECO-PReFoM) is also presented;

Moreover, this study is composed by three phases:

- *Planning*: this phase had the objective of defining, inspired by Valença et al.'s (2017), the steps to be followed during this study model construction. Therefore, the steps to be followed during the next phases of this study and how those steps would lead to the fulfillment of their objectives are defined in this phase. The author and 4 experts are involved with this phase;
- *Describing the power relationship dynamics*: this phase had the objective of describing the power relationship dynamics that were previously identified in the studies reported in Chapter 4 and Chapter 5. Therefore, the steps defined in the planning phase would be followed and, at the end of this phase, all the power relationship dynamics would be described and a graphical representation, a textual representation and a logical representation for each power relationship dynamic are presented. The author is the only person involved with this phase;
- *Constructing the model*: this phase had the objective of constructing the model that would represent the power relationships within an OSSECO. Moreover, this model would have to represent the entities that were involved with the

power relationship dynamics previously identified and described. At the end of this phase, the OSSECO power relationships entity model (OSSECO-PREM) and the relationships between power relationships model (OSSECO-PReFoM) should be constructed. The author is the only person involved with this phase.

2.6 Final Remarks

In this chapter, we presented this research's methodology and all of its phases. We also presented the studies that are part of each of the methodology's phases.

In the next chapter, we present the concepts of OSSECO and power relationships. We also present the npm and its OSSECO, compare the present research with related work and present a glossary that contains the terms and expressions related to this research's scope.

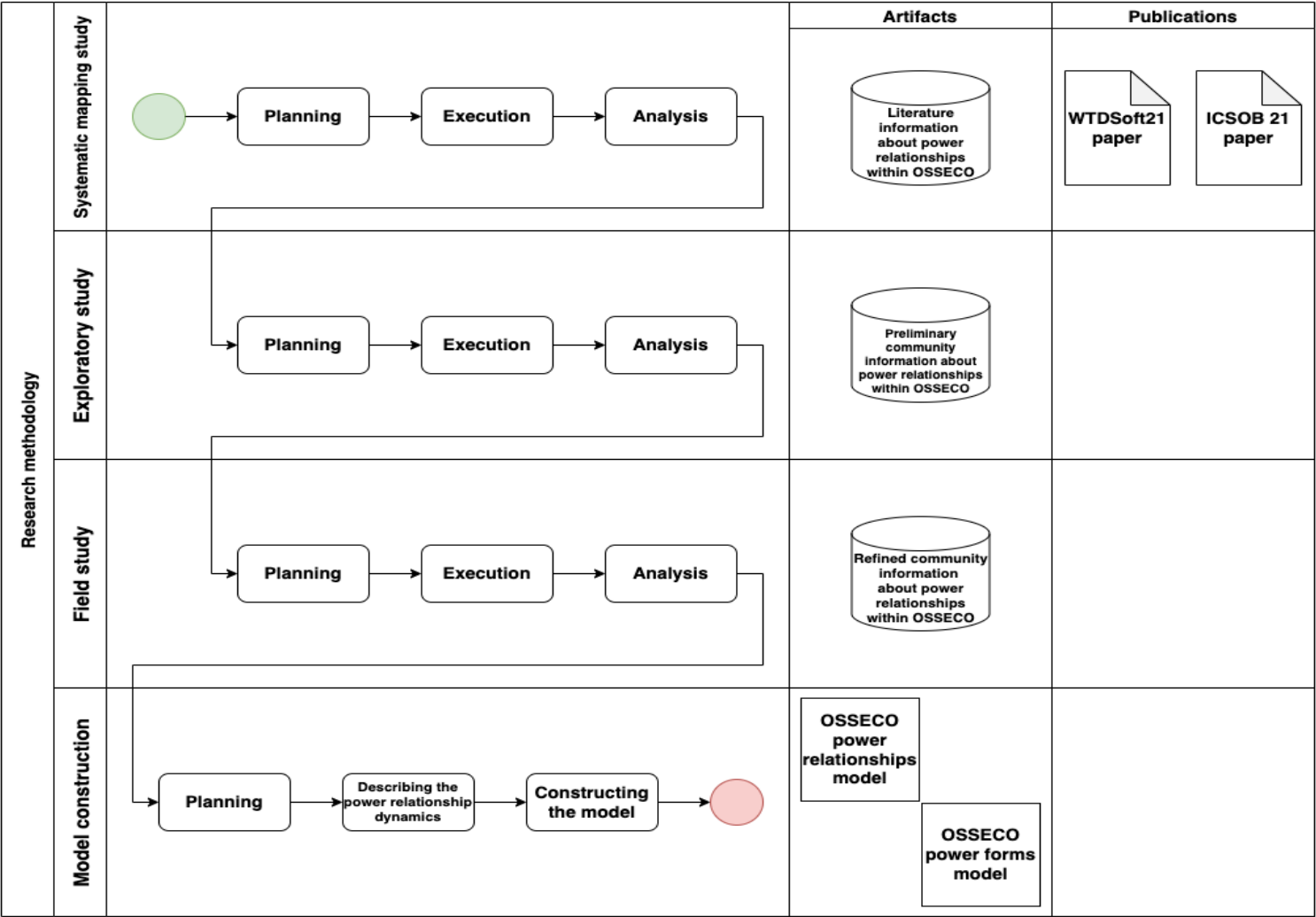


Figure 2.1: Research methodology.

Chapter 3. Background

3.1 Introduction

This chapter presents the basics of open source software ecosystems and power relationships. The related work presents studies that previously investigated power relationships in different areas, including but not limited to Computer Science and Software Ecosystems.

The remainder of this chapter is structured as follows: Section 3.2 presents the open source software ecosystems definition; Section 3.3 presents the npm package manager and its ecosystem; Section 3.4 presents the power relationships definition; Section 3.5 presents the related work; Section 3.6 presents the glossary containing the terms used in this research; and Section 3.7 the final remarks.

3.2 Open Source Software Ecosystems

In the early years of software engineering, software development would be performed by isolated teams, building monolithic software. Since then, software development has changed by replacing the isolated teams with developers that would flow between projects, collaborating, and the monolith by the development of modules that would integrate with each other (JANSEN et al., 2013).

This shift in software development allowed developers, organizations, and artifacts to interact, creating a network between them in which knowledge would flow. Moreover, those actors share a common technological platform (e.g., a programming language, digital store, operational system). Therefore, this gathering of actors, sharing a common technological platform and sharing knowledge could be defined as a software ecosystem (SECO) (BOSCH, 2009).

A model was proposed to represent the SECO and its networks in this context.

The Software Supply Network (SSN) (BOUCHARAS et al., 2009) considers three levels related to the SECO as seen in Figure 3.1: (a) the organizational scope level, in which are represented the actors that are part of the SECO and their relationships; (b) SSN and relationships level, in which are represented the relationships between software, hardware, and organizational services; and (c) the SECO level, which represents the relationships between different SECO.

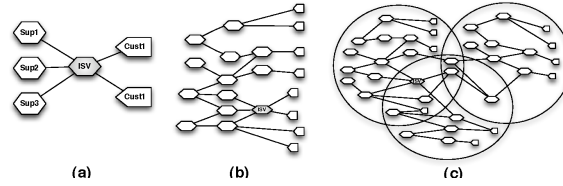


Figure 3.1: Software Ecosystem Scope Levels (BOUCHARAS et al., 2009).

Another representation refers to the dimensional perspective about SECO. The SECO would have three dimensions related to an ecosystem’s technical, social, and transactional aspects. It would also have a fourth dimension that would refer to the Engineering and Management (E&M) of a SECO, along with relationships for establishing a technological infrastructure (SANTOS et al., 2012), as seen in Figure 3.2.

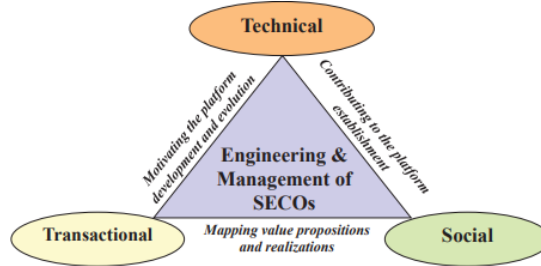


Figure 3.2: Three dimensional view of SECO E&M (SANTOS et al., 2012).

Therefore, in SECO, the actors are connected by symbiotic relationships in which each one of them has its own set of business goals, but all of them will collaborate to achieve their goals. Moreover, the actors will somehow benefit from their participation in SECO, and this benefit might vary according to the actor and the nature of the SECO (MANIKAS et al., 2013b). The nature of those SECO could be categorized by their business structure (MANIKAS, 2016):

- *Proprietary*: “Where value creation in the ecosystem is based on proprietary contributions. Ecosystems of this kind are typically protected by intellectual

property (IP) management processes and the value relates to monetary compensation”. An example of a proprietary SECO is the Danish telemedicine SECO, in which a set of different applications provide remote medical support to the Danish population (MANIKAS et al., 2013a);

- *Open Source*: “Where contributions are typically open to the rest of the actors or public. Value, in this ecosystem typically refers to other than monetary compensations, e.g. knowledge and experience or need satisfaction”. An example of an open source SECO is the npm SECO, in which developers gather to develop packages for the npm package manager (CONSTANTINOU et al., 2017);
- *Hybrid*: “The ecosystem that supports both proprietary and open source contributions”. An example of a hybrid SECO is the Android SECO, in which the Android development platform is openly distributed for developers to develop applications and thereafter commercialize those applications through an application store (EKLUND et al., 2014).

In this research, the focus is on the open source SECO. However, to understand this kind of SECO, it is necessary to understand the open source development before. The open source software (OSS) development emerged as an isolated, closed development alternative. In OSS, developers who share a development goal or a problem to be solved gather in an OSS project to achieve their goal or to solve their problem. In this context, the source code is open, and any developer who is interested in helping in the development can propose changes in an open and collaborative environment (ALMARZOUQ et al., 2015). Although in the early years only voluntary developers would join the OSS development, nowadays organizations also join OSS to achieve their business goals (PLAKIDAS et al., 2016).

Therefore, this group of developers and organizations perform the decision-making process in OSS. Together, they decide the best development practices, what technologies to use, changes in the main code, among other things. This group is known as OSS community (MARSAN et al., 2019). Although OSS development is open and collaborative, these communities have their hierarchy-like organization. This organization is role-based and it is possible to migrate from a role to another

inside of it (JENSEN et al., 2007), as seen in Figure 3.3. Moreover, in some cases, foundations are also founded to help organize the development and the decision-making process (LAAT, 2007).

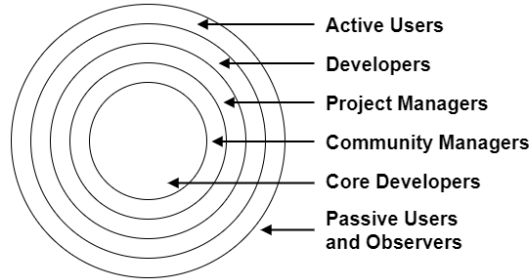


Figure 3.3: An “onion” diagram representing a generic OSSD project organizational hierarchy (JENSEN et al., 2007).

When a set of OSS projects share the same technology platform and their actors, organizations and artifacts interact with each other, flowing through the projects, we have an open source software ecosystem (OSSECO) (FRANCO-BEDOYA et al., 2017) as seen in Figure 3.4. Therefore, the OSS communities gather in an OSSECO community with the same goals as before but regarding the OSSECO context: to decide best practices, technologies to be used, development road-map, among others.

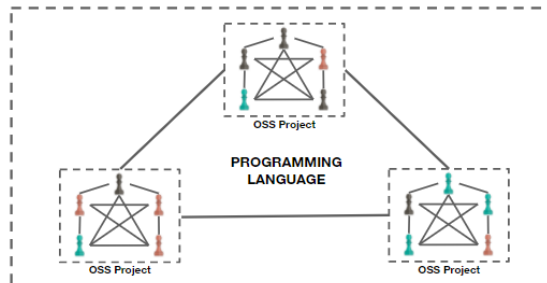


Figure 3.4: An OSSECO that has a programming language as its technological platform.

In some cases, those OSSECO communities can become so complex that foundations acting as OSSECO managers have to adopt governance methods. Governance methods directly affect the community, its interaction, and the OSSECO as a whole. Those methods vary from one OSSECO to another and from one foundation to another. However, one of the most determining aspects of the governance model is the decision-making process. There are autocratic and democratic governance models within an OSSECO. In the autocratic model, a foundation could consult the community regarding decisions before making them. However, a board composed of a

determined number of members who would consider that decision's impact on the OSSECO makes the final decision. In opposition to that, the democratic model relies on the community decision, which could be measured by the interactions within an OSSECO or by voting, to make a decision that could affect the OSSECO (LINÅKER et al., 2020b).

Moreover, the actors and organizations within an OSSECO have different motivations to participate. Although developers previously aimed at voluntary contribution considering the possibility of becoming a core member and being able to influence the decision-making process, nowadays they also consider the financial rewards that he/she could receive and the learning environment an OSSECO could be (GEROSA et al., 2021). The organizations' motivations, for example, tend to be related to business opportunities, their business model, or dependency on any of the OSSECO projects (LINÅKER et al., 2020a).

3.3 The npm and its Ecosystem

The npm is a package manager that helps developers with tasks related to the application's life cycle and dependency management (NPM, 2022). A registry, a collection of OSS projects, hosts the npm packages written in JavaScript and registered and uploaded by their owners, called packages.

The npm has an active community of developers, organizations, and users that collaborate to share JavaScript packages. Therefore, npm supports code reuse which developers can use to deliver code quickly. However, this possibility of reusing an external code could represent a security issue that has to be considered (ABDALKAREEM et al., 2017).

Since the npm packages are OSS projects that share a community of developers and organizations that interact to build artifacts, it is possible to say that the npm OSSECO exists. In addition to that, previous work has defined and investigated the npm OSSECO (CONSTANTINO et al., 2017).

The npm Inc., a company founded to manage the package manager itself, owns the npm. Although this company tends to leave decisions regarding the packages, their development, and their usage to the community members, in some cases, it makes decisions to preserve the npm and, therefore, the OSSECO.

In addition to that, the npm also has a portal that displays its packages numbers and collaborators and to what any logged-in user can have access. It also shows the package’s most recent versions published to the npm registry (NPM, 2022).

3.4 Power Relationships

It is possible to define power as the “ability to control people and events” or “an official or legal right to do something” (CAMBRIDGE, 2022). In one of the first studies regarding power, Dahl (1957) defines a situation in which power takes place as “A has power over B to the extent that he can get B to do something that B would not otherwise do”.

However, Emerson (1962) declared that power should be explored through the perspective of relationships and not to consider it a property of an actor or a group. This relationship perspective corroborates Foucault (1977) definition of power relationships: asymmetric relationships between two actors in which one actor has something and the other actor desires that something and the first actor does the second acts as he/she wants, as seen in Figure 3.5.

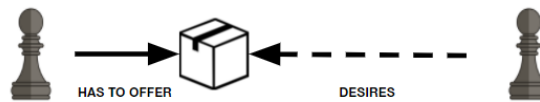


Figure 3.5: Power Relationship according to Foucault (FOUCAULT, 1977).

Those power relationships could be present in different situations, enabling them to be categorized. Leonidou et al. (2014) studies the buyer-seller relationship between firms to state that dependence is related to the need that one party has to maintain a relationship with another one to acquire resources or to accomplish business goals. This dependence would be the source of power for those relationships, that originate 6 power relationships forms that could lead to Satisfaction or Conflict, as seen in Figure 3.6. In this figure, the authors also represented the hypothesized associations between the power relationships forms, which are the arrows identified by H_{xy} . Those associations represent the relationships between the each power relationships form and dependence, conflict and satisfaction. Table 3.1 shows the power forms.

Areas other than Computer Science studied power relationships. In Sociology, power relationships have been studied through the perspective of how those would have effects on the relationships inside the society. Stehr and Adolf (2018) discuss the power through the knowledge perspective in different contexts within a society. In this scenario, power’s productiveness would be directly related to knowledge, which would be a social entity that would be used to exert power over someone.

Economics is another area that studied power relationships. Relationships between governments and organizations, especially related to the permission of operating in a country, are an example of power relationships. In this scenario, organizations desire to operate in a given country to achieve business goals or benefit from local policies or characteristics aligned to their businesses. In those cases, the local government would have to permit those organizations to operate in their country. Otherwise, the organizations could face legal procedures and even be unable to operate. Whenever this happens, organizations need to act as governments desires, either acting according to local working laws or complying with environmental rules (TELLERIA, 2017).

Table 3.1: Power relationships forms.

Power Relationship form	Definition (LEONIDOU et al., 2014)
Coercive power	Source of power based on a firm’s belief that another firm has the ability to punish the former if it fails to cooperate.
Expert power	Source of power based on the belief that one firm has specific knowledge needed by another firm.
Information power	Source of power based on the belief that one firm has valuable business data that could be shared with their business partners.
Legalistic power	Source of power based on the belief that one firm retains a legitimate right based on formal processes to influence another.
Referent power	Source of power based on the identification of one firm with another firm, or the former’s wish for being closely associated with the latter.
Reward power	Source of power based on a firm’s belief that another firm has the ability to mediate rewards and it will actually bring these rewards if the former complies.

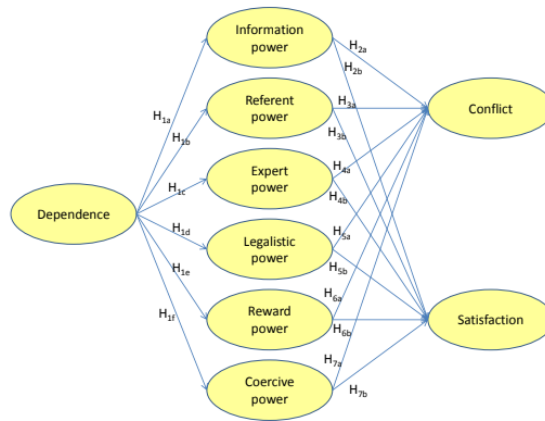


Figure 3.6: Power Relationship forms conceptual model (LEONIDOU et al., 2014).

3.5 Related Work

Previous research investigated power relationships within SECO. However, the focus of those researches or the contexts studied differed from those in the present research. We present those previous studies and their differences and similarities to the present research as follows.

Farias et al. (2019) aimed at studying the influencer characterization within the npm OSSECO. Using information gathered from a previous systematic mapping study, the authors assembled a questionnaire to identify which characteristics would be related to an influencer according to the npm OSSECO community. A survey was conducted to gather the OSSECO community's opinion about influencers' characterization and received 242 responses. As a result, 8 characteristics that a previous systematic mapping study has identified were ranked, and the most likely characteristics of an influencer (in the community opinion) were highlighted. Although this study investigates relationships within an OSSECO, it only focused on the influence perspective. The present research, on the opposite, focuses on the different power relationships and not only on the ones related to the influence.

In another related work, Linaker et al. (2020) also aimed at investigating influence within an OSSECO. However, its focus was on the stakeholders' influence in the requirements engineering (RE) process of the firms involved with an OSSECO. The objective was to propose a stakeholder influence analysis (SIA) method to help firms identify their stakeholders and their stakeholders' influence in the RE process within an OSSECO. The authors conducted a literature survey to gather informa-

tion about the stakeholders' influence in the RE process. The information gathered from the literature was used to construct a model to help in SIA. Although this study aimed to investigate relationships within an OSSECO, it only focused on the influence power and in a particular context (stakeholders' influence over the RE process). In the present research, not only the stakeholders' influence in the RE process within an OSSECO is considered but all the power relationships that might take place in this context.

Valena et al. (2017) aimed at investigating power relationships within a Small to Medium Enterprises (SME) SECO. The objective was to understand how power and dependence would affect the relationships within an SME SECO. The authors performed an exploratory study with 8 companies from two different emerging SME SECO, totalizing 27 interviews to achieve their objective. Using the information extracted from those interviews, the authors constructed power relationships models for each one of the companies. They then merged them into a power relationship model SECO (PRM-SECO), as seen in its coarse-grained view in Figure 3.7. The present research's objective is to gather information about power relationships within a SECO. The difference is that, in the present study, the relationships within an OSSECO are the objects of study and not within an SME SECO.

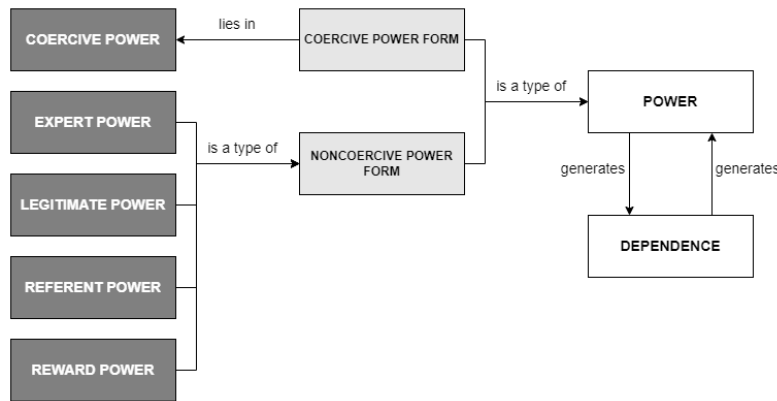


Figure 3.7: PRM-SECO coarse-grained view (VALENA et al., 2017).

Costa et al. (2020) aimed at investigating the proprietary SECO governance models by performing a literature review that would gather information to later survey and interview. During the analysis of the results, the authors observed that distributing power inside of a proprietary SECO would be directly related to bringing innovation into the proprietary SECO. Although this study investigates a different

type of SECO, it analyzes the effect of power over a SECO. In the present research, one of the objectives is to understand how power affects an OSSECO.

3.6 Glossary

During this research, there are some terms and expressions that are essential to the understanding of the research itself. Therefore, those terms and expressions must be defined since their definitions might differ depending on the context they are inserted. Moreover, some of those terms and expressions are directly related to the scope defined for this research and are key to understand the studies and their results. It is important to remark that the definitions presented in this section consider the defined scope of power within OSSECO, being the npm OSSECO the one chosen as study object of this research. The terms and expressions are defined as follows:

Action: in the present context, an action is an interaction between two entities (or groups of entities) within a power relationship dynamic.

Entity: in the present context, an entity represents a member of the OSSECO or an artifact that is part of the OSSECO. Therefore, an entity could be either a developer, an organization, or a package. It is also important to remark that there might be different types of entities. An example of types of entities is on the developers, that can be maintainers or collaborators.

Issue: in the GitHub platform, the issue is a feature used by those who are contributing with a project to communicate and register a necessary change. It has the format of a forum post, in which other users can also comment and debate those necessary changes. Those necessary changes can be either code evolution (new features) or corrections (bugs) (FARIAS et al., 2019).

Package: in the present context, it is the artifact created within the OSSECO. It is usually the product of a software project that is part of the OSSECO.

Power: it is the ability to make another power do things that he/she would not otherwise do (DAHL, 1957). Therefore, whenever an actor can make someone act differently from what he/she would usually do, this actor has power over he/she.

Power relationship: it is an asymmetric relationships between two actors in which one actor has something and the other actor desires that something and the

first actor does the second acts as he/she wants (FOUCAULT, 1977).

Power relationship dynamic: in the present context, a power relationship dynamic is a situation in which two groups of entities interact with each other and one of them exercises power over the other. It is composed by a set of actions that are performed by the entities groups during their interaction. An example of power relationship dynamic is the employment relation, in which an entity (an organization) exercises power over a group of entities (developers), by using its financial power to persuade them to work as it wishes.

Power relationship form: it is the type of power, according to the reason why a power relationship occurs (LEONIDOU et al., 2014). Therefore, depending on the motivation for the interaction to occur, we would have different power relationship forms.

Power source: it is the origin of power. It is the reason why an entity (or a group of entities) would have power over another entity (or another group of entities) (DAHL, 1957). An example of power source is the reputation that a developer holds within an OSSECO. Because of this reputation, he/she could be seen as an expert and then other developers would act as this expert desires just to have his/her technical help.

Pull request: in the GitHub platform, it is a code change request. However, differently from an issue, in which code changes are only asked, in the pull request the developers propose the code change that, if approved, are merged with the main code.

Trigger: in the present context, it is the side effect that an action has within an power relationship dynamic that ends up starting another power relationship dynamic.

3.7 Final Remarks

In this chapter, the concept of OSSECO was presented from the fundamental SECO perspective until the specific characteristics for this type of ecosystem. The npm OSSECO was also presented, as it is the OSSECO chosen to be studied in this research. Moreover, the concepts of power and power relationships were presented, as well as related work. In the last section, we presented a glossary containing the

terms and expressions related to this research's scope.

In the next chapter, a literature review is presented to understand what the literature already has on power relationships within OSSECO. This review was performed by conducting a systematic mapping study using the guidelines defined by Kitchenham & Charters (2007).

Chapter 4. An Overview Power Relation within Open Source Software Ecosystems

4.1 Introduction

This chapter presents a secondary study to gather information about power relationships within an OSSECO from the literature. We conducted a systematic mapping study (SMS) to investigate how the literature reports the power relationships within an OSSECO. Therefore, it was necessary to design a methodology to guide this study, and it was divided into three phases, as seen in Figure 4.1 and explained as follows:

- *Planning*: in this phase, the objective was to define the protocol used in this study. Therefore, we defined the digital libraries to run the search on, the search string and its terms, the exclusion criteria and inclusion criterion, and other protocol matters, following Kitchenham & Charters' guidelines (2007). At the end of this phase, we expect to have a defined protocol to run this search. The author was involved in this phase together with his mentor and two other students from the Scientific Methodology I course at the Graduate Program in Informatics, Federal University of the State of Rio de Janeiro (UNIRIO);
- *Execution*: in this phase, the objective was to run the search, select the studies that would answer the research questions, and extract the data from these studies. At the end of this phase, we expect to have a set of selected studies that could answer the research questions and to have the results and data

extracted reported. The author is responsible for extracting the data and two experts are involved with the evaluation of the results;

- *Analysis*: in this phase, the objective was to analyze the data extracted from the selected studies and answer the research questions using the extracted data. At the end of this phase, the research questions should be addressed appropriately and answered. The author is responsible for analyzing the data and two experts are involved with the evaluation of this analysis.

This work serves as a basis to help OSSECO community members and researchers in the field understand the power relationships within the OSSECO and their dynamics. Power relationships are mostly related to interactions, and keeping contributions happening is determinant to maintain the OSSECO development (LIAO et al., 2019). Therefore, the community could use information about the power relationship forms and their dynamics to enhance power relationships that support interactions maintenance and avoid power relationships that can lead developers to get away from the OSSECO. To guide this study and what it aimed at discovering, three research questions (RQ) were defined:

- *RQ1: What are the power relationships and their dynamics within an OSSECO?*
- *RQ2: What are the benefits those power relationships can bring into an OSSECO?*
- *RQ3: What are the challenges those power relationships can bring into an OSSECO?*

The remainder of this chapter is structured as follows: Section 4.2 presents the SMS planning phase; Section 4.3 presents the SMS execution phase; Section 4.4 presents the SMS results and the selected studies; Section 4.5 presents the SMS results analysis and answers for the RQ; Section 4.6 presents the limitations regarding this study; and Section 4.7 presents this study's final remarks.

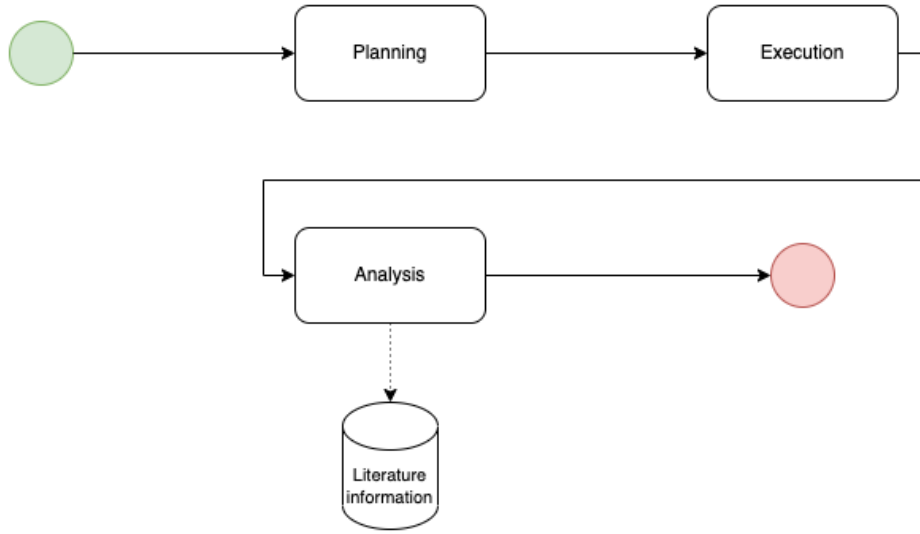


Figure 4.1: Systematic mapping study methodology.

4.2 Planning

Initially, we built the search string that would be used to perform the search using digital libraries. We combined research terms that guide this process in a search string during this process. It is important to remark that the search string also considered “software ecosystems” as many studies published before the definition of OSSECO called them only as “software ecosystems”. If we did not consider those, the search string would risk not retrieving valuable information. Also, the term “interaction” is added once power relationships influence interactions, and there might be some valuable information in the interaction study. The search string defined is:

((“open source software ecosystem” OR “software ecosystem”) AND
 (“power relation*” OR “relation*” OR “interaction*”))

After that, we were able to define the study selection process was divided into six steps, as seen in Figure 4.2 and listed as follows: (i) search execution; (ii) duplicate studies removal; (iii) first filtering: titles, abstracts and keywords reading; (iv) second filtering: introduction and conclusion reading; (v) third filtering: full text reading; and (vi) data extraction. The inclusion criterion (IC) defined for this study can be seen in Table 4.1 and exclusion criteria (EC) defined for this study can be seen in Table 4.2.

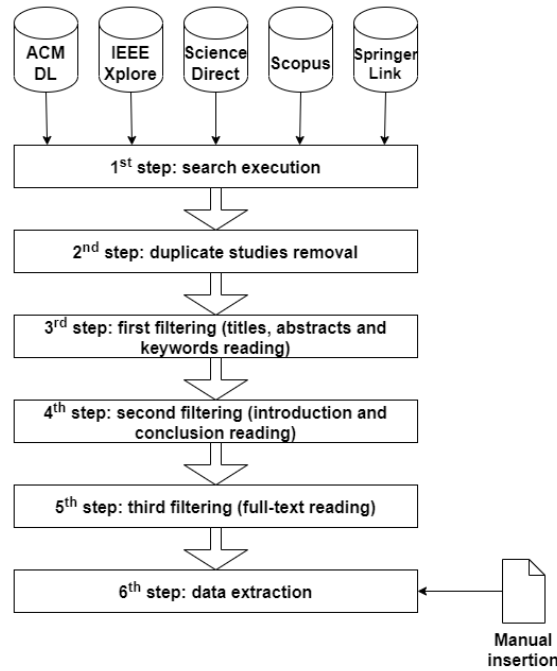


Figure 4.2: Studies selection process.

Table 4.1: Systematic mapping study inclusion criterion (IC).

ID	Inclusion criterion
IC1	The study reports power relationships in the OSSECO context.

Studies would be considered be adherent to the inclusion criterion whenever it reports a situation in which an organization, developer or other entity involved with an OSSECO acts as another entity desires. It is important to remark that this interpretation can be subjective and according to the researcher’s perspective since some studies might not have the focus on power relationships. Therefore, those studies might contain those power relationships but as part of their primary subject.

Table 4.2: Systematic mapping study exclusion criteria (EC).

ID	Exclusion criteria
EC1	The study was write in a language other than English.
EC2	The study is not available.
EC3	The study is similar to other study which is newer and reports the same results.
EC4	The study is a duplicate.

In the first step, we execute the search using a search string and search engines. In the second step, we remove the duplicated studies in the search results. In the third step, the filtering steps apply the IC and EC. In the first filter (step 3), we read

the titles, abstracts, and keywords and exclude the studies that do not comply with the IC or present some characteristics related to the EC. In the fourth step, we read the introduction and conclusion and exclude the studies that do not comply with the IC. Based on the full-text reading on step five, we exclude the studies that do not comply with the IC, and the ones left are considered the selected studies. Next, we execute a manual insertion of studies previously known as compliant to the IC and do not present any problems regarding EC. These studies were previously read by the author and they discuss influence within OSSECO (that is one situation in which an actor act as another one desires), and a study that was published after the search was performed but before the data extraction step begins. After the fifth step, the data extraction process aims at answering the research questions.

4.3 Execution

In the execution, the intention was to retrieve as many studies as possible reporting power relationships within OSSECO. The search engines used for the research were ACM¹, IEEE Xplore², ScienceDirect³, Scopus⁴, and SpringerLink⁵. In the first step, we found 1,068 studies. At the end of the fifth step, after all the exclusions, 8 studies were left. The manual insertion added 2 new studies to the list of selected studies. The researcher previously knew those 2 studies as sources of information about power relationships within an OSSECO. However, the digital libraries did not return those studies during the search execution step. Therefore, at the beginning of the data extraction phase, we had selected 10 studies to retrieve information about power relationships within OSSECO.

4.4 Results

Considering the studies that were selected during this SMS (Table 4.3), there were studies from 2007 until 2021, as can be seen in studies timeline (Figure 4.3). We had to exclude some studies that were not available online or written in English.

¹<https://dl.acm.org/>

²<https://ieeexplore.ieee.org/>

³<https://www.sciencedirect.com/>

⁴<https://www.scopus.com/>

⁵<https://link.springer.com/>

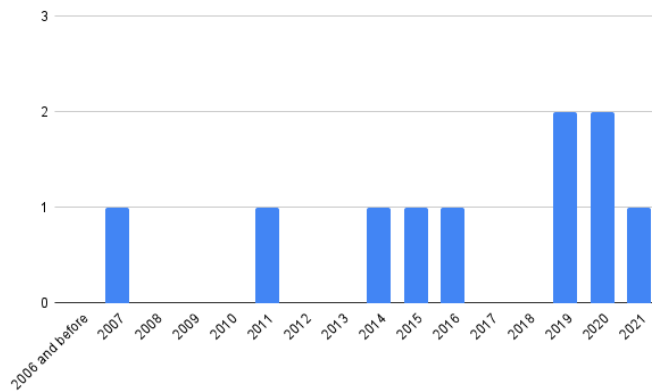


Figure 4.3: Selected studies timeline.

From the 10 studies that were selected (where SXX is the selected study ID), none of them cited “power relationships within OSSECO” directly. Were found studies that described situations in which it was possible to identify the occurrence of power relationships within OSSECO. Therefore, adding “interaction” as one of the search string’s terms was determinant to find those studies since power relationships happen mostly in interactions between actors.

We manually performed data extraction while reading the full text. As mentioned before, there was no study explicitly mentioning any power relationships within OSSECO. Therefore, it was essential to consider the studies’ implicitly reported power relationships. We were able to identify those power relationships primarily because situations - that can be called power relationships dynamics - were described in those studies along with the actors or artifacts that would have the power and those that would be affected by the power. Leonidou et al.’s (2014) power form classification was adapted to the OSSECO context to help in the identification of the forms of power relationships found in the SMS, as seen in Table 4.4. We also extracted the benefits that can be brought into the OSSECO by power relationships and the challenges that can occur because of the power relationships within an OSSECO. The benefits rely on the possibility that the power relationships can motivate the actors to keep contributing to the OSSECO and to maintain the OSSECO’s health - S03 (ECKHARDT et al., 2014), S04 (TEIXEIRA et al., 2015), and S06 (LIAO et al., 2019). However, the power relationships could bring some challenges, especially for newcomers who might have to transpose some barriers to accept their contribution and decide to abandon the OSSECO. Selected studies and

Table 4.3: Selected Studies.

ID	Title	Authors	Year
S01	Role Migration and Advancement Processes in OSSD Projects: A Comparative Case Study (JENSEN et al., 2007)	Jensen, C. and Scacchi, W.	2007
S02	The Onion Patch: Migration in Open Source Ecosystems (JERGENSEN et al., 2011)	Jergensen, C., Sarma, A. and Wagstrom, P.	2011
S03	The Merits of a Meritocracy in Open Source Software Ecosystems (ECKHARDT et al., 2014)	Eckhardt, E., Kaats, E., Jansen, S. and Alves, C.	2014
S04	Lessons Learned from Applying Social Network Analysis on an Industrial Free/Libre/Open Source Software Ecosystem (TEIXEIRA et al., 2015)	Teixeira, J., Robles, G. and González-Barahona, J.M.	2015
S05	How Do Software Ecosystems Evolve? A Quantitative Assessment of the R Ecosystem (PLAKIDAS et al., 2016)	Plakidas, K., Stevanetic, S., Schall, D., Ionescu, T. and Zdun, U.	2016
S06	Identification-method Research for Open-source Software Ecosystems (LIAO et al., 2019)	Liao, Z., Wang, N., Liu, S., Zhang, Y., Liu, H. and Zhang, Q.	2019
S07	What Characterizes an Influencer in Software Ecosystems? (FARIAS et al., 2019)	Farias, V., Wiese, I. and Santos, R.	2019
S08	A Method for Analyzing Stakeholders' Influence on an Open Source Software Ecosystem's Requirements Engineering Process (LINÅKER et al., 2020a)	Linåker, J., Regnell, B. and Damian, D.	2020
S09	Public Sector Platforms Going Open: Creating and Growing an Ecosystem with Open Collaborative Development (LINÅKER et al., 2020b)	Linåker, J. and Runeson, P.	2020
S10	The Shifting Sands of Motivation: Revisiting What Drives Contributors in Open Source (GEROSA et al., 2021)	Gerosa, M., Wiese, I., Trinkenreich, B., Link, G., Robles, G., Treude, C., Steinmacher, I. and Sarma, A.	2021

which RQ they help to answer can be seen in Table 4.5.

The power relationships, their dynamics and the source studies in which they were found can be seen in Table 4.6.

Table 4.4: Power relationship forms and the adaption to the OSSECO context.

Power Relationship form	Previous definition (LEONIDOU et al., 2014)	Definition within OSSECO context
Coercive power	Source of power based on a firm's belief that another firm has the ability to punish the former if it fails to cooperate.	The ability of an actor being able to punish other actor (e.g., by blocking him/her on the project, by preventing their changes to be merged into main code) if the second one does something the first one disapproves.
Expert power	Source of power based on the belief that one firm has specific knowledge needed by another firm.	The belief that an actor has an outstanding knowledge about something related to the OSSECO (e.g., he/she is a very experienced developer, he/she has skills that very few actors have) and that it might share this knowledge and help other actors if they act as the first one's desires.
Legalistic power	Source of power based on the belief that one firm retains a legitimate right based on formal processes to influence another.	The belief that an actor - due to legal processes and contracts - has the ability of making another actor act as the first one's desires.
Referent power	Source of power based on the identification of one firm with another firm, or the former's wish for being closely associated with the latter.	The belief that one actor - due to its identification with another actor - would be more likely to collaborate and act according to the another actor's desires.
Reward power	Source of power based on a firm's belief that another firm has the ability to mediate rewards and it will actually bring these rewards if the former complies.	The belief that one actor has the ability of rewarding other actor (e.g., with money, higher privileges) if the second one acts according to the first one's desire.

4.5 Analysis

During the analysis phase, it was possible to use data extracted from the selected studies to answer the RQ. These studies helped to identify the power relationships and their dynamics within an OSSECO but also helped to identify the benefits and challenges those power relationships can bring to the OSSECO.

Table 4.5: Selected studies and the RQ they help to answer.

ID	RQ1	RQ2	RQ3
S01	X		X
S02	X	X	X
S03	X		
S04	X		
S05	X	X	
S06			
S07	X	X	
S08	X		X
S09	X	X	
S10	X	X	

Table 4.6: Power relationship dynamics extracted from SMS.

Dynamic	Power form	Source studies
Conforming to stakeholders' requirements	Coercive	S08
Control over technological platform	Legalistic	S05
Employment relation	Reward	S09, S10
Learning reference	Referent	S10
Newcomers contribution rejection	Coercive	S01, S02
Role migration within the OSSECO	Reward	S02, S07
Technical leadership	Expert	S07

4.5.1 RQ1: What are the power relationships and their dynamics within an OSSECO?

The data extraction returned 7 power relationships dynamics that were divided into 5 categories adapted from Leonidou et al. (2014) (Table 4.6). The 5 categories represent the forms of power relationships present within an OSSECO. Their adapted definition to the OSSECO context are presented in Table 4.4. These dynamics help to understand who are the ones that hold power within the OSSECO in different situations and how this power affects the relationship between actors and artifacts. Therefore, it would be possible for OSSECO community members to make decisions according to their objective within the ecosystem by understanding how these relationships affect the OSSECO. It would also make it possible to understand why previous decisions or movements happened within the OSSECO based on the relationships that led to them. This work provides information that actors could use to take different actions depending on the goals and needs.

The *referent* power relationship form appeared in the “learning reference” power

relationship dynamic. In this dynamic, a project, an actor, or the experience of contributing to the OSSECO itself would make an actor play according to rules, desires, or expected behaviors to be able to remain in touch with the knowledge and, therefore, learn (GEROSA et al., 2021).

When analyzing the *coercive* power relationship form, we identified two dynamics: “conforming to stakeholders’ requirements”; and “newcomers’ contribution rejection”. Regarding the “conforming to stakeholders’ requirements”, it usually happens when a project tends to guide their development within the OSSECO according to some specific stakeholders’ requirements. This dynamic happens because, as more organizations are taking part in the OSSECO and business partnerships take place, there are some stakeholders whose requirements are a priority when compared to others members’ requirements. Therefore, actors that control the projects would be afraid of, if not conforming to their stakeholders’ requirements, being replaced by others who would (LINÅKER et al., 2020a). The stakeholders would have the power to punish the non-conforming projects. Regarding “newcomers’ contribution rejection”, in some cases, when a newcomer tries to contribute to a new project, he/she has his/her contribution rejected because the project’s owner does not know it yet. This dynamic happens because the nearly hierarchical structures that tend to appear within the OSSECO (JERGENSEN et al., 2011) acts as a barrier, where the ones that guide the projects’ decisions and development decide if a newcomer’s contribution is accepted or not (JENSEN et al., 2007).

We identified the *reward* power form in two power dynamics: “role migration within the OSSECO”; and “employment relation”. In the case of “role migration within the OSSECO”, the actors would play as other actors who have the power to give them higher privileges within the OSSECO, consequently, more power. This dynamic happens because having more privileges within an OSSECO means that the actor has the possibility of helping to make decisions about the ecosystem’s evolution and because his/her contributions are more likely to be accepted (JERGENSEN et al., 2011; FARIAS et al., 2019). Regarding the “employment relation”, it is known that the motivations for contributing with OSSECO have been changing, and there are actors that contribute because of their work (GEROSA et al., 2021). In this case, an organization employs an actor, and he/she acts as the organization desires

to keep his/her employment and receive his/her payment. It is a classic employment situation.

The *legalistic* power form takes place in the cases where an actor would have some rights over something within the OSSECO. A dynamic identified and linked with this power form was “control over technological platform”. In this dynamic, an organization owns a specific platform and can decide how to do things within this platform because of such ownership (PLAKIDAS et al., 2016).

We also identified the dynamic “technical leadership” and linked it to the *expert* power form. In this dynamic, an actor with considerable technical skill in a specific technology is a reference to other developers. Therefore, other actors will play as the expert desires so that they can count on his/her assistance in their projects or their problems (FARIAS et al., 2019).

Therefore, to properly answer RQ1, we were able to identify 5 power relationship forms within an OSSECO while extracting information from the selected studies: *coercive*, *expert*, *legalistic*, *reward*, and *referent*. In addition, 7 power relationship dynamics were identified: *control over technological platform*, *conforming to stakeholders’ requirements*, *employment relation*, *learning reference*, *newcomers’ contribution rejection*, *role migration within the OSSECO*, and *technical leadership*.

4.5.2 RQ2: What are the benefits those power relationships can bring into an OSSECO?

In order to answer RQ2, the analysis process focused on extracting the possible benefits associated with power relationships within an OSSECO. We identified three primary benefits during this investigation. Those benefits, the power relationship dynamics related to them, and their source studies can be seen in Table 4.7 and their explanations are presented as follows.

Motivations for collaborating with an OSSECO suffered a shift through time. One of the quoted motivations that was one of the top reasons for a developer to collaborate with an OSSECO was **learning** (GEROSA et al., 2021). Therefore, developers with new perspectives are attracted to OSSECO to acquire new skills and achieve learning objectives. Whenever this movement occurs, developers are possibly **bringing innovation** to the OSSECO. Since those developers are either newcomers

Table 4.7: Power relationships benefits to the OSSECO.

ID	Benefit	Power relationship dynamics	Source Studies
B01	Bringing innovation to the OSSECO	Learning reference	S10
B02	Organizing the OSSECO	Control over technological platform	S05
B03	Stable collaboration with the OSSECO	Employment relation, Role migration within the OSSECO	S02, S07, S09, S10

or developers that are improving their skills, it is possible that: (i) they have different perspectives and goals from those developers who were already part of the OSSECO; or (ii) they are improving their skills, probably enabling themselves to achieve goals that were not previously achievable for them. Given that innovation is essential to an OSSECO evolution (LINÅKER et al., 2020a), having power relationships that support innovation is a benefit that would help the OSSECO evolution. The power relationship dynamic related to this benefit is **learning reference**.

Moreover, the shift in motivations can be combined with early days motivations to assure **stable collaboration with the OSSECO**. Since organizations are more active within OSSECO to achieve their business goals than they were before, it is common to expect that their developers help develop projects that are part of an OSSECO (LINÅKER et al., 2020b). Whenever this movement occurs, those developers may make a regular contribution to the OSSECO, aiming at assuring quality and keeping projects updated to help organizations achieve their business goals (GEROSA et al., 2021). Unlike organizations' developers, developers who aim to take part in the OSSECO decision-making process also tend to make regular contributions. This regular contribution occurs since those regular contributions are a path to be followed by those who want to be noticed (FARIAS et al., 2019) and play decision-making roles within an OSSECO (JERGENSEN et al., 2011). The power relationship dynamics associated with this benefit are **employment relation** and **role migration within the OSSECO**.

Another benefit that was identified and is directly related to the power relationships is **organizing the OSSECO**. Whenever an organization has **control over a technological platform**, it can make decisions to protect the platform and

support its OSSECO evolution. That organization could adopt from simple best practices rules to complex governance methods. In addition to that, they could use a centralized decision-making process to mediate the OSSECO community’s desires with a roadmap that would support OSSECO evolution and prevent developers from abandoning the OSSECO (PLAKIDAS et al., 2016).

Therefore, we were able to identify 3 benefits of power relationships within an OSSECO to answer the RQ2: *bringing innovation to the OSSECO*, *organizing the OSSECO*, and *stable collaboration with the OSSECO*.

4.5.3 RQ3: What are the challenges those power relationships can bring into an OSSECO?

Although power relationships can bring benefits to the OSSECO, it is also possible that those relationships bring challenges. Analyzing the data extracted from the selected studies, we identified two challenges associated with power relationships. These challenges, the power relationships dynamics they are associated with, and their source studies can be seen in Table 4.8, and they are explained as follows.

Table 4.8: Power relationships challenges to the OSSECO.

ID	Challenge	Power relationship dynamics	Source studies
C01	Addressing private road-map	Conforming with stakeholders’ requirements	S08
C02	Blocking innovation	Newcomers’ contribution rejection	S01, S02

The decisions regarding the development and evolution of an OSSECO would be addressed directly by its community members, according to the community’s goals and problems to be solved. However, since organizations take part in an OSSECO and use its package as dependencies, they are expected to have their requirements. In addition to that, many package maintainers and other organizations realized that attending to those requirements could lead to business partnerships and even to financial rewards. Therefore, those organizations would influence the requirements engineering (RE) process, having their requirements prioritized when compared to the community’s requirements (LINÅKER et al., 2020a). This movement could lead to developers abandoning the OSSECO since their requirements would not

be attended to and could shift the OSSECO’s goals and problems to be solved, **addressing private roadmap**. The power relationship dynamic associated with this challenge is **conforming with stakeholders’ requirements**.

Another identified challenge related to the power relationships within an OSSECO is the **blocking innovation**. Although we know that bringing new perspectives, goals, and problems to be solved into an OSSECO is essential to its evolution, there are cases in which newcomers have their contributions blocked. The effects that this blocking can have gone from the newcomers deciding to abandon the OSSECO since they have their contributions rejected to having the OSSECO evolution slowed down or even stopped. The power relationship dynamic associated with this challenge is **newcomers’ contribution rejection**.

Moreover, based on the 2 identified challenges, we were able to answer RQ3. The challenges related to power relationships within an OSSECO are: *addressing private roadmap*, and *blocking innovation*.

4.6 Limitations

There are some limitations as follows: (i) there has been no deep investigation regarding power relationships within OSSECO yet, and it is possible that we did not use other terms related to them in this study’s search string and did not gather some information, but to minimize this risk, “interaction*” was added to the search string. Other terms, such as “platform” (that would be related to the ecosystems) could have been added to the search string; (ii) the power relationship dynamics were identified by a researcher and evaluated by two experts, both with experience of more than 10 years conducting SMS, and one of them is an OSS expert and the other one is a software ecosystem expert; (iii) since half of the power relationship dynamics were extracted from a single study, it was not possible to identify a pattern in the power relationship dynamics report; furthermore, (iv) other power relationship dynamics might not have been identified from the selected studies.

4.7 Final Remarks

In this chapter, we reported a literature investigation. It was possible to identify 7 power relationship dynamics within an OSSECO by conducting a systematic

mapping study. This chapter also presented a previous power relationship forms definition adapted to the OSSECO context, resulting in 5 power relationship forms that could occur within an OSSECO. We also categorized the identified power relationship dynamics according to the OSSECO power relationship forms definition.

Moreover, we identified 3 benefits and 2 challenges associated with the power relationships within an OSSECO. We explained how those benefits and challenges could occur within an OSSECO and presented the power relationship dynamics related to them and their source studies. We also presented their possible effects on an OSSECO.

In the next chapter, we report two studies with an OSSECO community: an exploratory study and a field study. The next chapter's goal is to explore the information that an OSSECO community has about power relationships since this chapter presented the information present in the literature regarding power relationships within OSSECO.

Chapter 5. Power Relations within the npm Open Source Software Ecosystem in Practice

5.1 Introduction

In this chapter, we present two studies that have the objective of gathering information about power relationships from an OSSECO community. The first study is an exploratory study based on an asynchronous interview with the objective of having a first contact with an OSSECO community, based on the literature investigation findings. The second study is a field study based on live interviews with the objective of refining the findings from the exploratory study and having more details on power relationships within an OSSECO. Both studies are conducted with npm OSSECO community members.

The remainder of this chapter is structured as follows: Section 5.2 presents the exploratory study; Section 5.3 presents the field study; and Section 5.4 presents the final remarks.

5.2 The Chosen OSSECO and Protocol Decisions

Although we decided that we would study the power relationships within an OSSECO according to an OSSECO community experience, we did not define which OSSECO and its community would be the studies' object. It was necessary that this OSSECO was active, having its projects currently developed by the community, and that it was already defined. It was also necessary to access the OSSECO's projects and their information, including developers' contact information. The OSSECO chosen for this study was the npm OSSECO and the reasons are as follows:

- the npm OSSECO is a previously well-defined OSSECO. Constantinou et al. (2017) defined the npm OSSECO when investigating developers retention in this OSSECO;
- the npm OSSECO is an active OSSECO. It is an OSSECO that is growing in terms of the number of projects. In 2017, the npm OSSECO had about 380,000 packages. In 2022, the npm OSSECO had about 1,855,007 packages. This represents a growth of 488% over 5 years (NPM, 2022);
- the npm OSSECO has its projects publicly hosted at GitHub ¹, a platform for collaborating technically and socially over software development.

Since these studies would require interaction with people, we would have to define the terms and conditions for those npm OSSECO community members to participate in the study. It was important to define those terms and conditions considering 5 main subjects that would protect the participants identity, data and well-being. Although each of the studies would have their own Terms and Conditions document, the 5 main subjects involved in their design are as follows:

1. confidentiality, assuring that the respondent would not be identified by any information provided and that all data would be anonymously used;
2. voluntary participation, informing the respondent that his/her participation is entirely voluntary and that declining to participate would not involve any penalty or benefit loss. It also informed that the participation could be withdrawn at any time by the respondent;
3. data storage, informing the respondent for how long the researchers would store the data gathered. It also informs that the researchers can be contacted at any time regarding interviews matter.
4. consent, informing that when the respondent accepts the Terms and Conditions, he/she gives him/her consent to the usage of all the information provided according to the conditions above mentioned;

¹<https://github.com/>

5. confirmation, informing that, by clicking on the button to move forward to the questionnaire, the respondent accepts the participation Terms and Conditions.

Thereafter, the OSSECO community that would have members invited to the studies was defined - the npm OSSECO -, and the main subjects that should be covered by the terms and conditions for participating in the studies was defined as well. Therefore, we can now present the exploratory study with the goal of getting to know better what is the npm OSSECO community members perspective on power relationships within their context.

5.3 A Preliminary Study on the npm OSSECO

5.3.1 Introduction

This section presents an exploratory study. Considering the information gathered in the systematic mapping study reported in Chapter 4, we set up a questionnaire to extract information from an OSSECO community.

This study's methodology involves a structured asynchronous interview with developers from an OSSECO. Three phases were defined for this study, as seen in Figure 5.1 and as explained as follows:

- *Planning*: this phase had the objective of defining what would be the OSSECO that would be studied and what would be the interview protocol, following Kitchenham et al.'s guidelines (2015). The author is involved in this phase together with two experts who help design the protocol;
- *Execution*: this phase had the objective of running a pilot, randomly selecting the asynchronous interview potential respondents, inviting them to participate, and collecting their answers. The author is the only person involved with this phase;
- *Analysis*: this phase had the objective of answering the research questions for this study based on the results of the asynchronous interviews. Therefore, to extract the information from the asynchronous interview responses, we applied open and axial coding (STRAUSS et al., 2007). The author is responsible for gathering the data and analyzing it while two experts evaluate the findings.

This study had the objective of understanding an OSSECO community’s opinion about the power relationships in their context, regarding what power relationship forms are present in their context, what power relationship dynamics are present in their context, and who are the entities involved with those power relationships. Additionally, respondents shared their motivations to be part of the OSSECO and evaluated a power relationship definition regarding their experience within the OSSECO. The data gathered was analyzed to help answer the research questions (RQ) for this study that were defined as follows:

- RQ1: What are the power relationship forms within an OSSECO?
- RQ2: In what situations those power relationships take place in an OSSECO?
- RQ3: Who are the entities that participate in those power relationships within an OSSECO?

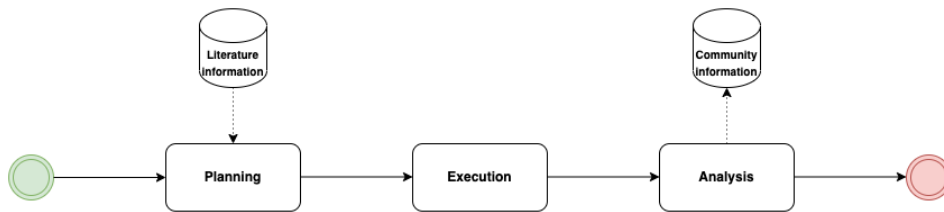


Figure 5.1: Exploratory study methodology.

The remainder of this section is structured as follows: Subsection 5.2.2 presents the exploratory study planning phase; Subsection 5.2.3 presents the exploratory study execution phase; Subsection 5.2.4 presents the exploratory study results obtained from the execution phase; Subsection 5.2.5 presents the exploratory study results analysis and the research questions answers; Subsection 5.2.6 presents the limitations regarding this study; and Subsection 5.2.7 presents this study’s conclusions.

5.3.2 Planning

The first step in the planning phase of this study was to define which method would be used to interact with the npm OSSECO community members. It was decided that asynchronous interviews would be conducted in this study. The decision of

using an asynchronous interview format during this study was related to the fact that this study would be a first contact with an OSSECO community. Therefore, we would not like to overwhelm the OSSECO community with a large scale survey or with a long time duration synchronous interview.

The second step in the planning phase for this study was the protocol definition. In this step, we should define three artifacts: (i) the potential respondent invitation; (ii) the participation Terms and Conditions; and (iii) the questionnaire (both demographic and technical questions). We wrote the three artifacts in English.

The method chosen to send the invitations was e-mail messaging. We chose the e-mail invitation as GitHub publicly hosted developers' e-mails addresses. Developers who would have contributed with npm projects hosted in GitHub would be eligible to respond to the asynchronous interview questions and, therefore, participate in the study. Contribution, in this context, is considered to be either committing code, opening pull requests, opening issues, or commenting on an open issue/pull request. This condition is related to the fact that this study aimed at investigating the opinion of those who are active members of the OSSECO. Following Kitchenham's et al. guidelines (2015), we designed the invitations to present the researchers involved with it, to be customized according to the potential respondent, and to present the study and its objective briefly. An invitation sample can be seen in Appendix I.1.

Thereafter, the participation Terms and Conditions should be defined. We defined that we would host the Terms and Conditions and the questionnaire in a Google Forms form. We decided that the Terms and Conditions would have a brief researchers presentation, an introduction to the study and the conditions. The participation Terms and Conditions for this study can be seen in Appendix I.2.

We set up the questionnaire using the information gathered from the study reported in Chapter 4, and the estimated time to participate was from 5 to 7 minutes. We divided the questionnaire into two parts: the Demographic Questions (DQ) and the Technical Questions (TQ). All the questions were obligatory. We designed the 3 DQ to characterize the respondents and are as follows:

- *DQ1: What is your gender?*
- *DQ2: In npm ecosystem, Are you a...*

- *DQ3: How many projects do you contribute to?*

The DQ can be seen, as we present them to the respondents, in Appendix I.3. It is interesting to remark that in DQ2 a set of possible roles that an actor can play in an OSSECO is presented. Therefore, the idea would be for the respondent to complete with his/her role.

We designed the TQ according to the information gathered in the study reported in Chapter 4 and the RQ for the present study. The TQ and their rationales are as follows:

- *TQ1: “Power relations are relations between an individual who has something to offer and another individual who desires that something. In open source software ecosystems, these relations are directly related to roles and motivation for contribution. Understanding and managing those relations is fundamental for an open source software ecosystem do evolve, because they can affect both collaboration (by improving and encouraging it) and conflicts (by avoiding them).” Do you agree with what is stated?*

Rationale: Considering power relationships have not been broadly explored in the OSSECO context, the first question presented a statement for power relationships within the npm OSSECO and asked the respondents to which extent they agree. The objective of this question was to potentially find a definition of what would be a power relationship within an OSSECO and why it would be interesting to know them. Another objective was to get the respondents warmed up about the topic to respond to the following questions.

- *TQ2: Justify your previous answer.*

Rationale: We also asked for them to justify their level of agreement. The objective was to verify if the community considered the proposed statement adequate to address power relationships within the OSSECO. It also had the objective of gathering more information about the power relationships within the OSSECO.

- *TQ3: What power relationships do you identify inside the npm ecosystem?*

Rationale: This question was a direct question that aimed to retrieve what the npm OSSECO community members considered as power relationships in their context. There was a possibility of identifying power relationship forms, dynamics, and entities involved using the answers for this question.

- *TQ4: Which of the motives below relate to your participation within the npm ecosystem?*

Rationale: This question presented some possible motivations for respondents to contribute with the npm OSSECO and asked them to select those who would apply to them. In addition, the respondents could also add other motivations that were not present in the options. This question’s objective was to understand what would be the most recurrent motivations among npm OSSECO community members as the motivation for contribution has been changing (GEROSA et al., 2021). Therefore, it would be possible to understand some power relationship dynamics within the OSSECO, knowing the community’s top motivations.

The TQ can be seen as we presented them to the respondents in Appendix I.4. The GQM (Goal-Question-Metric) model (BASILI, 1992) for this exploratory study clarifies the goal of this study. It can be seen in Table 5.1.

Table 5.1: The GQM for the exploratory study goals.

Analyze With the purpose of With respect to The point of view of In the context of	power relationships and their dynamics identifying impact OSSECO community members npm OSSECO
---	---

Thereafter, we present the results, dividing them into the respondents’ characterization and the technical questions. Considering the characterization results, it is possible to know our population sample. Since we would expect that the technical questions would assist in answering the RQ, we first present their answers individually before analyzing them.

During the analysis, we would use open and axial coding (STRAUSS et al., 2007) to categorize quotes within the respondents’ answers and then group those categories according to relationships among them. Those procedures help extract information

from the interviews responses that we can use to understand the power relationships within an OSSECO. During this analysis, we relied on ATLAS.ti² tool in its web version to manage quotations and codes.

5.3.3 Execution

Before reaching the population sample, we conducted a pilot in which 2 experts contributed with suggestions that resulted in the questionnaire's adjustments regarding the questions' formulation. As a suggestion, a question regarding if the respondent was a practitioner or if the respondent was an academic was replaced by a question regarding the role this respondent plays in the npm OSSECO. Although knowing if the respondent is a practitioner or an academic would not be necessarily related to his/her experience with power relationships within an OSSECO, knowing which role he/she plays would. Since different roles can have different levels of permissions and can hold power in different situations.

Next, 14 developers participated in the interviews responding to the questionnaire in an asynchronous format. We invited those developers as they are active members amongst the group of active members of a set of 310 projects that are part of the npm OSSECO. This questionnaire received responses from January 30th, 2021 to February 28th, 2021.

5.3.4 Results

Characterization

According to the demographic questions section of this asynchronous interview, it was possible to characterize the respondents' group of this exploratory study. Table 5.2 shows each respondent (RXX, as ID) answers to the demographic questions. Regarding gender identification, 71.4% of the respondents identified themselves as male, while 28.6% preferred not to answer the question. It is important to remark that, from the 28.6% of respondents who preferred not to answer the question, half of them (14.3%) expressed dissatisfaction regarding the question itself and the options.

Considering the possible roles within the npm OSSECO, 85.7% of them declared themselves project owners within the npm OSSECO. In contrast, 7.1% declared

²<https://web.atlasti.com/>

Table 5.2: Respondents answers to the demographic questions.

ID	DQ1	DQ2	DQ3
R01	Male	Project owner	2 projects
R02	Prefer not to answer	Project owner	5+ projects
R03	Male	Project owner	5+ projects
R04	Male	Project owner	2 projects
R05	Male	Project owner	5+ projects
R06	Prefer not to answer	Project owner	2 projects
R07	Male	Project owner	5+ projects
R08	Prefer not to answer	All of the above	5+ projects
R09	Male	Project owner	2 projects
R10	Male	Project owner	5+ projects
R11	Male	Project owner	5+ projects
R12	Prefer not to answer	Project owner	1 project
R13	Male	User	1 project
R14	Male	Project owner	3 projects

themselves as users, and other 7.1% declared themselves as playing all of the roles (project owner, project contributor, and user). It is interesting to remark that identifying as playing all of the roles within the OSSECO was a suggestion sent by one of the respondents. This answer reminds us of the possibility of playing different roles while collaborating and interacting within the OSSECO.

Half of the respondents (50%) selected 5 or more projects option when asked about how many projects they contribute to. At the same time, 28.6% said that contributed to 2 projects, 14.3% said that contributed to 1 project only, and 7.1% said that contributed to 2 projects.

Technical Questions

TQ1 presents a power relationship definition and asks the respondents to evaluate it. The portion of respondents who agreed with the definition was 28.6%, those who partially agreed were 42.9%, and those who disagree were 28.6%, as seen in Figure 5.2.

However, to understand the TQ1 results, respondents justified their answers in TQ2. Those who agreed with the definition quoted the situation in which an entity A desires a given behavior and has something to offer. An entity B desires that something and behaves as entity A desires expecting that entity A would reward entity B with that something. An example is the answer of Respondent 10, who

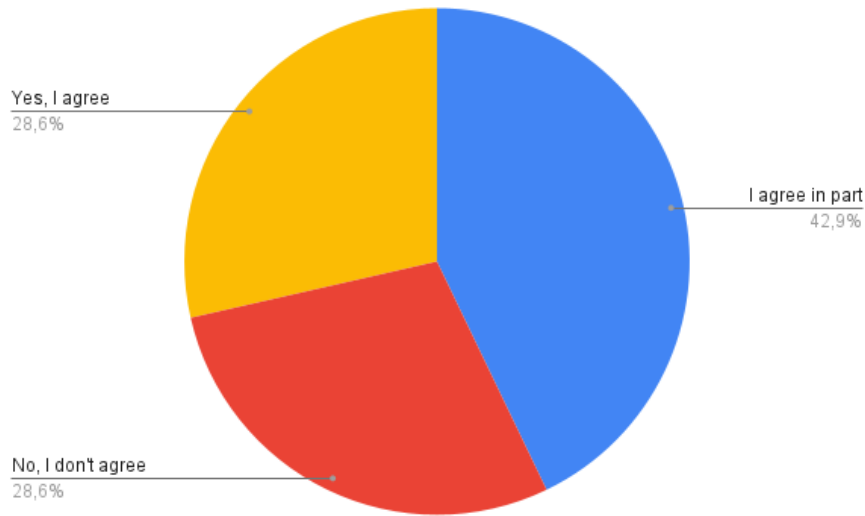


Figure 5.2: Agreement with proposed definition in TQ1.

claimed to have some trouble answering the question but defined a power relationship in his/her opinion:

Respondent 10

“...when A can do some things and B can’t but B needs A to do something (such as publish an update) then A has the ability to with-hold something, that is, they have power over B.”

In the group of those who partially agreed with the proposed definition, two main topics were referred to by the respondents: the need for a definition that would picture power relationships beyond a transactional perspective and the power relationships in collaborations motivated for other reasons than rewards. As an example of those necessary improvements to the definition, Respondent 02 and Respondent 14 justified their answers as follows:

Respondent 02

“It’s a very simplified description as stated, missing out on positions and dynamics of power beyond the want/need relation.”

Respondent 14

“For me it is more satisfaction of sharing with others.”

The group of those who disagreed with the proposed definition referred to the fact that, in their opinion, OSSECO communities would not have roles. An example of this opinion is the answer of Respondent 04:

Respondent 04

“Open Source community must be open, no ‘roles’ but just proximity to the specific task.”

In the TQ3, respondents were asked about the power relationships that they identify within the OSSECO. We further explore the results for this question in the Analysis section of this study. However, we can remark that only the respondents who did not believe in the existence of power relationships within the OSSECO were not able to identify those. All the other respondents identified at least entities involved with power relationships within the OSSECO, and some even gave examples of those relationships. The examples of both situations are as follows:

Respondent 07

“Non. There are none. Its all contributions. No power. Just added value.”

Respondent 02

“I will highlight one for you: negligent code ownership. I’ve personally experienced the power of denying collaboration.”

The last technical question, TQ4, presented some possible reasons that could motivate the respondents to participate in the OSSECO. It is important to remark that the respondents could also add motivations not displayed in the default list of options. The motivations and the number of times respondents chose each one of them can be seen in Figure 5.3.

It is interesting to remark that “Development for professional purpose” is the second most chosen motivation in TQ4. This high number of motivation choices is probably related to the fact that there is a shift in the motivation for collaborating within OSSECO, as stated by Gerosa et al. (2021).

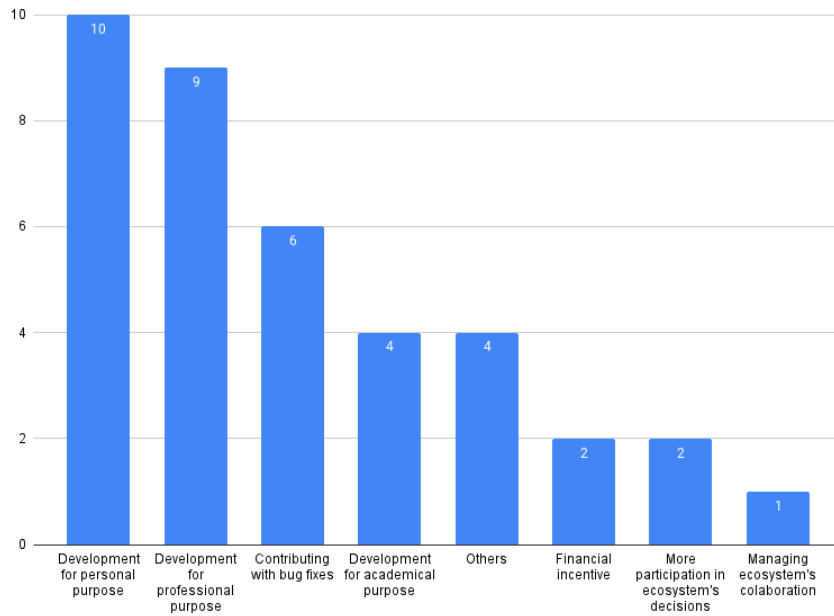


Figure 5.3: Motivations to contribute to an OSSECO.

5.3.5 Analysis

In the analysis phase, we analyze the results shown in the last section to extract the answers to our study RQ. Therefore, it was needed to define a method to analyze those results. We chose the open and axial coding (STRAUSS et al., 2007) to be executed in this study as the activity of categorizing sentences in the responses and then identifying the relationships between those categories. Therefore, information would be adequately extracted from the responses and used to answer our RQ.

During the analysis phase of this study, it was not possible to determine a relationship between different roles and different perspectives about how the power relationships occur within an OSSECO.

Open and Axial Coding

In the open coding, we read the responses for TQ2 and TQ3, and then codes were linked to sentences contained in those responses. Those codes would represent the ideas extracted from those sentences and would be used to categorize the sentences. In this procedure, we did not define a set of terms that we would use during the coding procedure, which was completely free.

During the open coding step, we identified 27 codes. The 2 codes that were the

most quoted in the responses for TQ2 and TQ3 combined had 10 quotes. After finishing the coding process, while reading the codes, it was possible to identify three codes that were mergeable: **pull request acceptance**, **pull request rejection**, and **pull request evaluation**. Therefore, we chose the **pull request evaluation** code to be maintained as the evaluation process of a pull request can involve either acceptance or rejection. The codes and their quotation number of times can be seen in Table 5.3. The number of quotations ordered the codes in a descendant order. In the cases of codes that had the same quantity of quotations, they were alphabetically ordered. It is possible to make some analyses considering the identified codes. Interestingly, all the codes were obtained directly from the responses reading process; therefore, we did not search for previously defined codes in the sentences.

Table 5.3: Codes identified during the open coding step.

Code	Quantity of quotations
Collaborator	10
Maintainer	10
Pull request evaluation	5
Employment relation	4
Make code contribution	4
Package	4
Technical leadership	4
Make a decision	3
Open an issue	3
Organization	3
Reference to others	3
Right to make decisions	3
Sponsoring	3
User	3
Dependency between projects	2
Fork the project	2
Give money	2
Learn	2
Blocking	1
Breaking changes	1
Conflicts	1
Control over technological platform	1
Package ownership	1
Promoting collaboration	1
Reputation	1
Share knowledge	1
Technological platform	1

In the axial coding, the codes gathered in the open coding step would have their relationships identified, and then they would be grouped. In this study, we defined two relationships by which we should group the codes: Dynamics and Entities. However, during the axial coding process, a third relationship was identified: Actions. This study defined that we could only link a code to one relationship. The information extracted was used to answer our RQ.

RQ1 and RQ2: What are the power relationship forms within an OSSECO? In what situations those power relationships take place in an OSSECO?

Although RQ1 aims to identify what power relationships occur within an OSSECO, we need first to identify the situations, also called dynamics, in which power relationships occur. Thereafter, based on the power relationship forms definition (as seen in Table 4.4), it is possible to identify which of those power relationship forms take place within an OSSECO according to the respondents.

The identified dynamics within an OSSECO according to the respondents are shown in Figure 5.4. The quantity of quotations related to each dynamic is shown along with their code. The dynamic that was the most quoted in this study was **pull request evaluation**. In this dynamic, those who want to collaborate with code must comply with specific rules and behaviors in their pull requests. In the meantime, those who have the power to evaluate the pull requests decide if that contribution is accepted or rejected. Therefore, those who decide have power over those who want to collaborate. This dynamic was related to the *coercive* form since it is expected that those who want to have their collaboration merged act as the maintainers want to avoid punishment by rejection. Respondent 06 states:

Respondent 06

“Largely the power resides with the maintainers, those who wish to see change done must approach the core team via issues and PR requests...”

Another dynamic identified in this study is **employment relation**. This dynamic is directly related to the classical employment relationship in which an organization employs a person. From the perspective related by the respondents, this

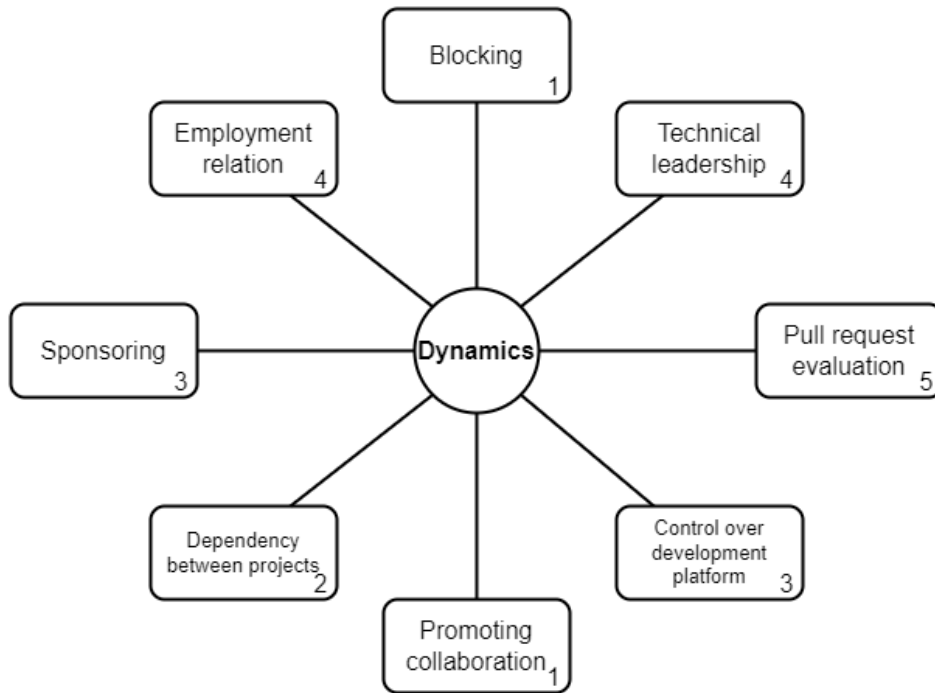


Figure 5.4: The axial coding regarding the Dynamics.

employment would be a reward for the technical skills a developer could have. An organization would hire, and therefore pay, developers who have the technical skills to help develop a project that is part of the OSSECO and essential for that organization to keep developing its projects. Therefore, this dynamic would be directly related to the *reward* form. Respondent 08 reported this power relationship dynamic as follows:

Respondent 08

“...there is a large portion of force or implied force in many power relationships, particularly around employment, and that is deeply intertwined with the concept of open source.”

Whenever developers have a recognized set of skills within the OSSECO, other developers and organizations may consider this skilled developer a reference. Therefore, they will act as the skilled developer desires in order to have him/her share his/her knowledge and technical skills with them. This dynamic is called **technical leadership**, and it is an example of a *expert* power form. Respondent 10 mentioned this power relationship dynamic:

Respondent 10

“...when A can do some things and B can’t but B needs A to do something (such as publish an update) then A has the ability to with-hold something, that is, they have power over B.”

In some situations, organizations can own technological platforms and make decisions that affect those platforms. Whenever this happens, we have the **control over technological platform** dynamic. In this dynamic, it is expected that organizations that own those platforms have the right to decide what development roadmap to be followed. This dynamic is an example of a *legalistic* form due to the right to decide. Respondent 05 mentioned this power relationship dynamic:

Respondent 05

“Microsoft, through owning GitHub, definitely can move open source efforts towards its projects more than other companies.”

Another dynamic in which organizations act is **sponsoring**. In this dynamic, organizations that depend on a given project sponsor that project to guarantee that developers involved with the project keep its maintenance. Since an organization sponsors the projects according to their importance to that organization and their maintenance, it is clear that this is a reward to those projects. Therefore, it is an example of a *reward* power form. Respondent 05 reported this power relationship dynamic:

Respondent 05

“If Microsoft is sponsoring a given open source project, it’ll likely have more people interested in working on it...”

However, the projects are not only environments in which power relationships can occur but also have relationships, and therefore, power relationships between them. Within the npm OSSECO, it is expected that projects have other projects as dependencies. Usually, those projects chosen to be dependencies are known as projects that can solve specific development problems and have a satisfactory level of quality and security involved. In those cases, we have a **dependency between projects**

power dynamic in which a project could even consider development changes in order to keep using another project as a dependency. This dynamic is directly related to the *referent* power form. Respondent 10 reported this power relationship dynamic as follows:

Respondent 10

“Owning an npm module means you can publish new patch versions, which will be installed automatically when people update. So you have the power to push updates to people’s applications, almost.”

On some occasions, conflicts happen between a developer who maintains a project and another who wants to collaborate or depends on that project. Although these conflicts are often resolved, and the collaboration/dependency is maintained, in some cases, the developer who maintains the project may decide to block the other user because of the conflict. This punishment happens according to the behavior of the blocked developer whose actions were considered inappropriate by the maintainer. Therefore, this **blocking** dynamics is also a *coercive* power form example. Respondent 02 mentioned this power relationship dynamic as follows:

Respondent 02

“I wanted a bug fixed that caused issues downstream, but the maintainer of a repository did purposefully abandon their code and blocked me after I inquired.”

Promoting collaboration emerges as a *referent* power form example. Whenever an entity has enough reputation so that other entities identify themselves with it, that entity can influence others’ decisions. It would be expected, in those cases, that the entity that influences others has a higher hierarchy position than the influenced ones. Respondent 05 reported this power relationship dynamic:

Respondent 05

“Microsoft, through owning GitHub, definitely can move open source efforts towards its projects more than other companies.”

Therefore, we are able to answer both RQ1 and RQ2. Regarding RQ1, we were able to identify 5 power relationship forms that occur within an OSSECO: *coercive*, *expert*, *legalistic*, *referent* and *reward*. In addition, considering RQ2, we were able to identify 8 power relationship dynamics that occur within an OSSECO: *blocking*, *control over technological platform*, *dependency between projects*, *employment relation*, *pull request evaluation*, *promoting collaboration*, *sponsoring* and *technical leadership*. The power relationship dynamics, their power relationship forms and the respondents (Rxx, where xx is a number) who reported them can be seen in Table 5.4.

Table 5.4: The identified power relationship dynamics and their power relationship forms.

Power relationship dynamic	Power relationship form	Respondent
Blocking	Coercive	R02
Control over technological platform	Legalistic	R05
Dependency between projects	Referent	R10, R13
Employment relation	Reward	R06, R08, R10
Pull request evaluation	Coercive	R01, R02, R03, R06, R10
Promoting collaboration	Referent	R05
Sponsoring	Reward	R05, R06
Technical leadership	Expert	R06, R10, R14

RQ3: Who are the entities that participate in those power relationships within an OSSECO?

In the process of coding the sentences, it was possible to identify the entities who are part of the power relationships that occur within the OSSECO. Those entities can either exercise power or be affected by the power. The codes that were related to the entities relationship can be seen in Figure 5.5 and are explained as follows.

The **maintainer** is the entity that has committer privileges within projects that are part of the OSSECO. They are core developers within the OSSECO, and they help make decisions that could potentially affect a lot of other entities and projects within the OSSECO. Respondent 09 directly reported this entity:

Respondent 09

“Maintainers and users.”

Meanwhile, **collaborator** is the entity that contributes to projects within the OSSECO. This entity opens pull requests, open issues, and comments on both pull requests and issues. It can also be responsible for bringing new requirements into the projects or identifying bugs that need fixing. Respondent 01 mentioned this entity:

Respondent 01

“PRs for fixing bugs or adding features to the modules published to npm.”

The **user** is the entity that has projects that are part of the OSSECO. It usually works with projects that have a project that is part of the OSSECO as a dependency. It can also open issues or comment on other issues. Sometimes, it is possible that this entity can request a new feature or fix. Respondent 09 also directly mentioned this entity:

Respondent 09

“Maintainers and users.”

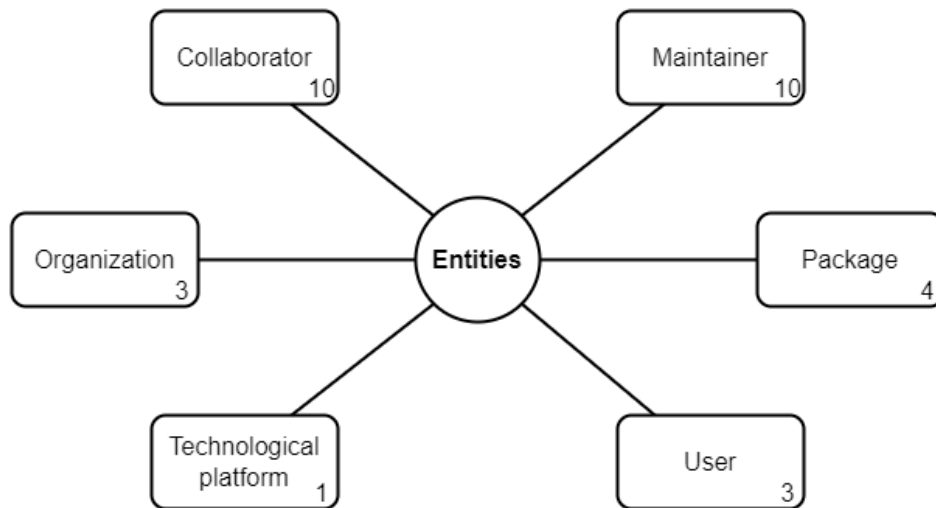


Figure 5.5: The axial coding regarding the Entities.

Organization is an entity that represents a company that takes part in the OSSECO. This entity is usually responsible for making decisions regarding the **Technological platform** (in the case of owning it), the projects it might own, and for sponsoring projects and hiring developers. Respondent 05 reported this entity while mentioning Microsoft:

Respondent 05

“If Microsoft is sponsoring a given open source project...”

The **package** is an artifact, but, in this power relationships context, it acts as an entity. In some cases, they can be referred to as “modules”. However, in this study, we chosen to call them “packages” since this name is used more frequently within the npm OSSECO. This entity usually interacts with other **Package** entities but can also suffer effects from power relationships because of the actions of other entities. Respondent 10 reported this entity:

Respondent 10

“npm modules are structured as property.”

Technological platform is the central technology around which the OSSECO exists. In some cases, an **organization** owns this entity and decides the technological platform development and evolution. Respondent 05 reported this entity in a indirect way. It is important to remark that he/she mentioned the fact of Microsoft owning GitHub, which is not the technological platform in this case. However, GitHub owns npm Inc, which is the organization that runs the npm package manager, which is the technological platform:

Respondent 05

“Microsoft, through owning GitHub...”

Therefore, we are now able to answer RQ3. The 6 entities identified as being involved with power relationships within an OSSECO are: *collaborator*, *maintainer*, *organization*, *package*, *technological platform*, and *user*. The entities and the respondents who reported them can be seen in Table 5.5.

The Actions

Although the main objective was to answer our RQ it was possible to identify one more relationship between codes during the coding: Actions. The Actions represent some of the possible actions that the entities could take within an OSSECO related

Table 5.5: The entities identified within an OSSECO.

Entity	Source respondent
Collaborator	R01, R02, R03, R05, R06, R08, R10
Maintainer	R01, R02, R03, R05, R06, R08, R09, R10
Organization	R05, R06, R08
Package	R05, R10, R13
Technological platform	R05
User	R01, R09, R10

to power relationships. Those actions may assist in understanding how the entities interact in the power relationship dynamics.

Make code contribution refers to the act that both **maintainer** and **collaborator** can perform to have changes desired by them merged into the main code. Respondent 01 reported this action:

Respondent 01

“PRs for fixing bugs or adding features to the modules published to npm.”

Regarding the **open an issue**, it is an action that is commonly performed by **collaborator** and **user**, but can also be performed by **maintainer**. Issues are opened in projects whenever a problem is identified or the community requests a new requirement. Respondent 06 mentioned this action:

Respondent 06

“Largely the power resides with the maintainers, those who wish to see change done must approach the core team via issues and PR requests...”

Make a decision is the action of choosing, considering what would be best for the community what development paths should be followed. This actions can be performed by **maintainer** or **organization** and can affect directly **technological platform** or **package** and indirectly the entire OSSECO. Respondent 05 reported this action:

Respondent 05

“Microsoft, through owning GitHub, definitely can move open source efforts towards its projects more than other companies.”

Give money is usually performed by **organization** that uses financial motivation to guide developers and projects and, therefore, achieve their business objectives. Respondent 05 also mentioned this action as follows:

Respondent 05

“If Microsoft is sponsoring a given open source project...”

Learn is an action that is directly related not only to a power relationship form (*expert*) but also to motivations to contribute within an OSSECO. Respondent 14 reported this action:

Respondent 14

“For me it is more satisfaction of sharing with others.”

Whenever conflicts reach a stage in which the **maintainer** ignores the **collaborator** and rejects his/her pull requests, there is the possibility of **fork the project**. This action involves creating a clone of the code in a new project and having the desired changes merged into this clone. Respondent 06 mentioned this action:

Respondent 06

“...disagreements I have noticed result in a forking of efforts which rarely get merged back into the original project.”

Share knowledge is the act of, motivated either for a financial reason or for the interest of helping other developers, guiding other developers through learning within an OSSECO. It can be performed by **maintainer** or **collaborator**. Respondent 14 also reported this action as follows:

Respondent 14

“For me it is more satisfaction of sharing with others.”

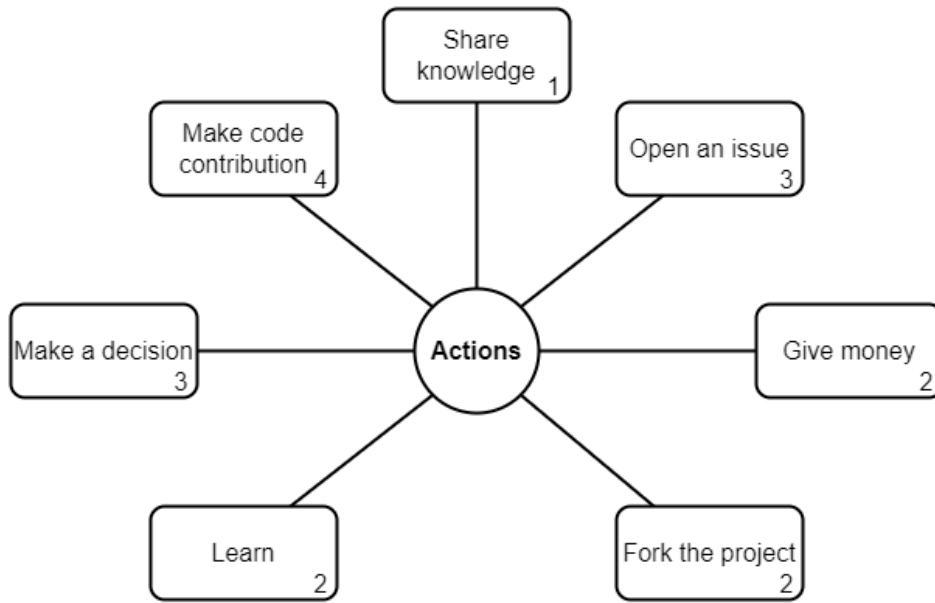


Figure 5.6: The axial coding regarding the Actions.

5.3.6 Limitations

During this study, we identified some limitations and threats to validity. Considering that the majority of the respondents identified themselves as male and as project owners, it is possible that the results are not representative of the population. This could represent that the study results are biased according to those groups opinion about power relationships within their context. In addition to this fact, the number of respondents concerning the population sample can also represent a limitation of this survey's results.

Another limitation is related to the format of this study. In an asynchronous interview format, an electronic questionnaire was used to reach a higher quantity of respondents. However, a live interview could extract more information about the respondents' experiences with power relationships within the npm OSSECO. One of the reasons why this could happen is the possibility of asking follow-up questions, which is not possible in an asynchronous format.

Moreover, this study only considered the npm OSSECO community. Therefore, the results might not be fully applicable to other OSSECO. Hence, to achieve saturation on OSSECO scope as a whole, it would be necessary to investigate other OSSECO communities.

Finally, this study had its results evaluated by two experts: one of them is an

expert with more than 10 years of experience regarding software ecosystems and the other one is an expert with more than 10 years of experience regarding open source software. However, the results have not been validated by other any OSSECO community.

5.3.7 Conclusion

In this section, a exploratory study based on asynchronous interview study was reported. This study aimed at investigating an OSSECO community and its opinion about the power relationships within their context. We used open and axial coding (STRAUSS et al., 2007) to conduct a qualitative analysis and to extract information from the respondents responses.

Therefore, we were able to identify 8 power relationship dynamics, in which 5 them have not been previously identified during the literature investigation in Chapter 4. Those 5 power relationship dynamics were categorized according to the power relationship forms definition presented in Chapter 4.

In addition, we identified 6 entities that take part in power relationship dynamics within an OSSECO. Those entities represent either actors, artifacts or the technological platform of an OSSECO.

Moreover, we were able to identify 7 actions that can be performed by entities within those power relationship dynamics. These actions can be either reactions to a previous action or an initial action to an interaction and, therefore, a power relationship dynamic to occur.

The implications this study has to practitioners is related to the possibility of, using the information provided, understand and take advantage of those power relationship dynamics. Those who want to migrate from a role with less privileges to a role with more privileges could use the information about power relationships to understand how they affect this migration and, therefore, achieve his/her goals. The implications this study has to researchers is related to the information that could be used in further studies. Those who study the OSSECO, could use the identified power relationships to investigate

5.4 Refining the npm OSSECO Community's Opinion on Power Relationships

5.4.1 Introduction

This section presents a field study based on an interview that had its questionnaire set up to confirm and gather new information considering all the information that was gathered in Chapter 4 and in the exploratory study reported in Section 5.2.

This study was structured in three phases that can be seen in Figure 5.7 and are explained as follows:

- *Planning*: this phase had the objective of defining what would be the interview protocol, following Kitchenham et al.'s guidelines (KITCHENHAM et al., 2015). It was also defined how the potential interviewees would be selected. The npm OSSECO was already selected as this would be a refining study regarding what was previously discovered in Section 5.2. However, it was also necessary to define who would be the npm OSSECO community members that would be reached out to participate in this field study. The author was involved in this phase together with 4 experts who helped to design the protocol;
- *Execution*: this phase had the objective of executing a pilot and then executing the interviews with the npm OSSECO community members. After the pilot, possible adjustments in the interview protocol would be made prior to the interview with OSSECO community members execution. The author was the only person involved with this phase;
- *Analysis*: this phase had the objective of analyzing the data that was gathered in the interviews and extracting information from them. That information should be used to answer the research questions that were defined for this study. In this phase the information gathered is also compared to the information that was gathered in the exploratory study reported in Section 5.2. The author is responsible for gathering and analyzing the data while the 4 experts evaluate the results and analysis.

Since this study aims at refining the information gathered in the exploratory study reported in Section 5.2, the same research questions (RQ) defined for that study were also defined for the present study. They are as follows:

- RQ1: What are the power relationship forms within an OSSECO?
- RQ2: In what situations those power relationships take place in an OSSECO?
- RQ3: Who are the entities that participate in those power relationships within an OSSECO?

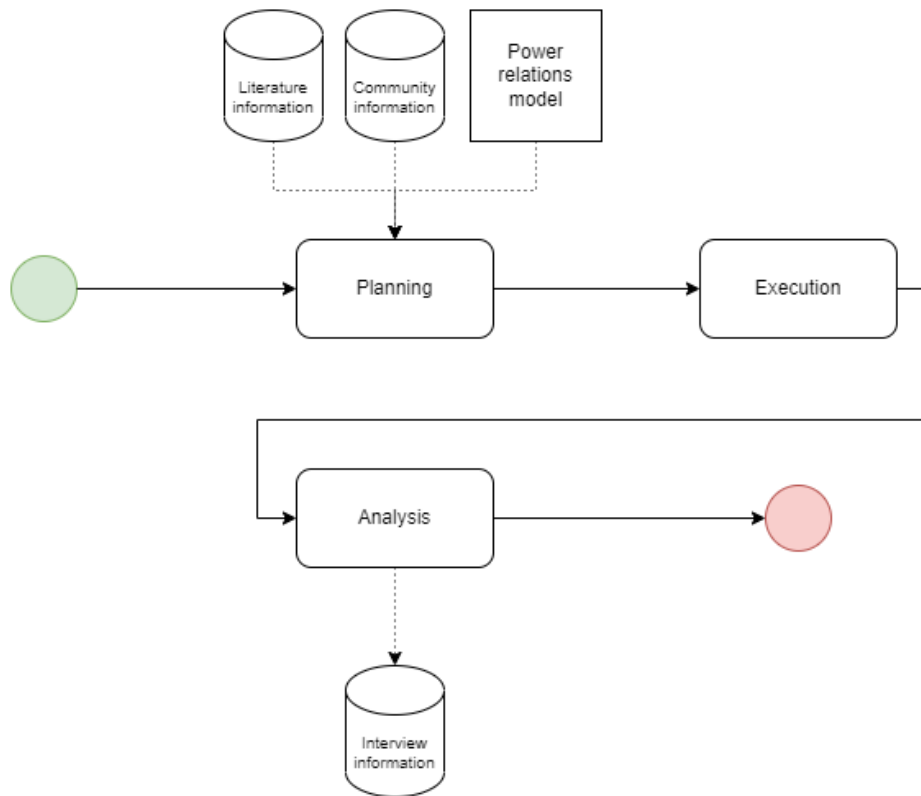


Figure 5.7: Field study methodology.

The remainder of this Section is structured as follows: Subsection 5.3.2 presents the field study planning phase; Subsection 5.3.3 presents the field study execution phase; Subsection 5.3.4 presents the field study results obtained from the execution phase; Subsection 5.3.5 presents the field study results analysis and the research questions answers; Subsection 5.3.6 presents the limitations regarding this study; and Subsection 5.3.7 presents this study’s conclusions.

5.4.2 Planning

The first step of the field study planning was to define which kind of method would be used for interacting with the npm OSSECO community members. It was decided that in this study live interviews would be conducted to try to extract more information that it was possible to do with the exploratory study reported in Section 5.2. In addition, differently from the exploratory study, we had more information about how OSSECO community members consider power relationships within their context and thus it was possible to design a questionnaire that would help us extract more information about power relationships within an OSSECO.

However, in this study, the process of finding the potential interviewees and their contact information for inviting them to participate was different from the previous study. Potential interviewees selection was made through the information contained in boards that are related to the OSSECO, nearby developers and users and developers and users who were indicated by other participants. Those who had a previous experience within the OSSECO or are active members of it were considered able to participate in this study.

The second step in the planning phase for this study was the interview protocol definition. In this step, four artifacts should be defined: (i) the potential interviewees invitation; (ii) the participation Terms and Conditions; (iii) the questionnaire (both demographic and technical questions); and (iv) the interview guidelines. The four artifacts were written in English.

Although the method chosen to send the invitations was email messaging, it was possible that a first contact were made using other communication tools, such as messaging applications. Email invitation was chosen as developers' emails addresses were public information hosted in some boards contact information. Developers who would have contributed with npm projects hosted in GitHub and users who use npm packages in their projects would be considered able to respond the interview questions. Contribution, in this context, is considered to be either committing code, opening pull requests, opening issues or commenting on an open issue/pull request. To use a npm package, in this context, is considered to have a package that is part of the npm OSSECO as a dependency for the project the user works with. Following Kitchenham's et al. guidelines (2015), the invitations were designed to present the

researches involved with it, to be customized according to the potential interviewees and to briefly present the research and its objective. An invitation sample can be seen in Appendix II.1.

Thereafter, the Terms and Conditions should be defined. It was defined that the Terms and Conditions and the demographic questionnaire would be hosted in a Google Forms form. The interview itself would be performed using the Google Meet platform, being recorded and would be stored in a private repository within Google Drive. It was decided that the Terms and Conditions would have a brief researches presentation, an introduction to the research and the conditions. The participation Terms and Conditions for this study can be seen in Appendix II.2.

The questionnaire was set up using the information gathered from the studies reported in Chapter 4 and in Section 5.2, and the estimated time to participate was from 30 to 45 minutes. The questionnaire was divided in two parts: the Demographic Questions (DQ) and the Technical Questions (TQ). All the questions were required. The 3 DQ were designed with the objective of characterizing the interviewees. The DQ can be seen, as they were presented to the interviewees, in Appendix II.3 and are as follows:

- *DQ1: Which gender do you most identify with?*
- *DQ2: Which role do you most identify with within the npm ecosystem?*
- *DQ3: How long have you been contributing with code within the npm ecosystem?*
- *DQ4: How long have you been commenting or opening issues within the npm ecosystem?*
- *DQ5: How long have you been contributing (code, comments, issues) with open source software in general?*

The TQ were designed according to the information gathered in the study reported in Section 5.2 and the RQ for the present study. The TQ and their rationales are as follows:

- *TQ 1: In an open source software ecosystem, projects may depend on other projects or have interfaces between each other. In those cases, could you tell*

a bit about how this would affect relationships between projects when breaking changes happen?

Rationale: Since dependency between projects is a common characteristic of the npm OSSECO, this questions aims at understanding how the need for a code maintenance could affect this dependency relationship.

- *TQ 2: In an open source software ecosystem, interactions between developers, users, organizations and artifacts are very important to the ecosystem development. Can you give an example of how those interactions are affected by power relations?*

Rationale: This question aims at, based on the interactions interviewee witness within the OSSECO, identifying power relationship dynamics that might have not been identified so far.

- *TQ 3: In an open source software ecosystem, developers, organizations, and artifacts might be known by their reputation (either positive or negative). How would this reputation affect its lead?*

Rationale: Since the reputation was identified in the previous exploratory study presented in Section 5.2 as a factor, this question aims at understanding how it could influence the entity participation within an OSSECO.

- *TQ 4: How did you experience power relations within the open source software ecosystem? Could you please share your experience?*

Rationale: This question aims at, based on the interactions interviewee experience within the OSSECO, identifying power relationship dynamics that might have not been identified so far.

- *TQ 5: In those relationships, how do the actors benefit from the power relations effect? What are the challenges power relations bring into the relationships?*

Rationale: Since the effects that power relationships bring into an OSSECO were identified in the literature but not in the exploratory study presented in Section 5.2, this question aims at determining the positive and negative effects power relationships bring into an OSSECO.

- *TQ 6: In an open source software ecosystem, projects tend to have their own rules and best practices defined. How are the rules and the code best practices defined and how do they affect the relationships within the ecosystem?*

Rationale: This question aims at understanding how the rules that organize development and collaboration within an OSSECO are defined and what power relationships could be related to them.

Thereafter, the interview guidelines were defined. Those guidelines would refer to important steps to be taken during the interview with the objective of informing the interviewees about the research and of having a standardized execution of the interview. The guidelines are as follows:

1. Remind the interviewee that the interview will be recorded and ask him/her for permission;
2. Send the interview terms and conditions and the demographic questions;
3. Introduce the research and the interviewer;
4. Thank for the participation and ask the interviewee to introduce himself/herself;
5. Ask the technical questions;
6. Ask for the indication of other potential participants;
7. Thank for the participation.

The execution phase is planned to be conducted in two steps: pilot and interview with the npm OSSECO community. However, between those two steps, this study's protocol, containing updates regarding the pilot, was submitted to the Eindhoven University of Technology's Ethical Review Board. This submission was necessary since two researchers from this University collaborated in the questions' design and our study involves interacting with people, which is a sensitive situation. It is important to remark that these two researchers did not have access to the responses of the interviewees. The Eindhoven University of Technology's Ethical Review Board acceptance letter for this study protocol can be seen in Appendix II.4.

During the analysis, we would use open and axial coding (STRAUSS et al., 2007) to categorize quotes within the interviewees' answers and then group those categories according to relationships among them. Those procedures help extract information that we can use to understand the power relationships within an OSSECO. During this analysis, we relied on ATLAS.ti³ tool in its web version to manage quotations and codes.

5.4.3 Execution

The interview process was conducted in two phases: in the first phase, a pilot was conducted with a npm OSSECO community member. In the second phase, the interview containing the adjustments suggested during the pilot was conducted with a group of npm OSSECO community members.

During the pilot, it was possible to verify that we should decrease the maximum estimated time for the interview from 45 minutes to 25 minutes and the minimum estimated time from 30 minutes to 20 minutes. The pilot interviewee was a npm OSSECO community member with a vast experience within the OSSECO, to which he/she collaborated with for over 7 years. Therefore, he/she was able not only to answer the interview questions and the follow-up questions, but also to give detailed examples of situations that could relate to power relationships. However, in some moments, examples and comments would fade into completely different subjects and the interviewer had to make the power relationships subject center in the discussion again. The decision of decreasing the time for the interview conduction would allow the interviewer and the interviewee to focus on the power relationships subject.

Regarding the usage of information gathered in the pilot, it was decided to use those information. Since the interviewee was a npm OSSECO member with a vast experience within the OSSECO, he/she was able to answer all of the questions and provide valuable information regarding the power relationships within OSSECO. Considering that the change in estimated time was the only change in the interview protocol, therefore, not affecting the technical questions and their possible answers, it would be possible to use this information.

In total, we interviewed 13 npm OSSECO community members from June, 2021

³<https://web.atlasti.com/>

to January, 2022. Although we wrote the artifacts used in this interview in English, we gave the interviewees the option of being interviewed in Portuguese. From the 13 interviews, we conducted 4 of them in English and 9 of them in Portuguese.

5.4.4 Results

Characterization

Considering the responses received from interviewees regarding the demographic questions section of this interview, it is possible to characterize the group of interviewees for this field study. Table 5.6 and Table 5.7 show the answers of each interviewee to the demographic questions. Regarding the DQ1, 23.1% of the interviewees identified themselves as female while 76.9% of the interviewees identified themselves as male.

Table 5.6: Interviewees answers to the demographic questions, part 1.

ID	DQ1	DQ2	DQ3
I01	Male	User	Never contributed with code
I02	Female	User	3-5 years
I03	Male	User	7+ years
I04	Male	User	3-5 years
I05	Male	All of the above	1-3 years
I06	Male	All of the above	5-7 years
I07	Female	User	3-5 years
I08	Male	User	Never contributed with code
I09	Male	Maintainer	7+ years
I10	Male	Maintainer	7+years
I11	Female	User	Never contributed with code
I12	Male	All of the above	3-5 years
I13	Male	User	Never contributed with code

In DQ2, we asked interviewees about which role did they most identify with within the npm OSSECO. A remarkable information is that when interacting within the npm OSSECO, a person could play different roles across different packages and through the time. However, we would expect that the interviewees would identify themselves with their current role and the role they play related to the package they most interact with. Therefore, 61.5% of the interviewees identified themselves as users, while 23.1% identified themselves as playing all of the possible roles (maintainer, collaborator and user), and 15.4% identified themselves as maintainers.

Table 5.7: Interviewees answers to the demographic questions, part 2.

ID	DQ4	DQ5
I01	Never contributed with comments or opening issues	Never contributed with open source software
I02	Never contributed with comments or opening issues	1-3 years
I03	7+ years	7+ years
I04	Never contributed with comments or opening issues	Never contributed with open source software
I05	3-5 years	3-5 years
I06	5-7 years	7+ years
I07	Never contributed with comments or opening issues	Never contributed with open source software
I08	Never contributed with comments or opening issues	Never contributed with open source software
I09	7+ years	7+ years
I10	7+ years	7+ years
I11	Never contributed with comments or opening issues	Never contributed with open source software
I12	1-3 years	1-3 years
I13	3-5 years	3-5 years

Considering the contribution time, we asked interviewees to indicate, according to the option in DQ3, for how long they have been contributing with code within the npm OSSECO. From the interviewees, 30.8% of them indicated that they contributed with code for 3 to 5 years, while 23.1% responded that contributed for more than 7 years, 7.7% responded that contributed for 1 to 3 years and another 7.7% responded that contributed for 5 to 7 years. On the other hand, 30.8% of the interviewees indicated that have never contributed with code within the npm OSSECO.

We can observe that the total percentage of interviewees who have already contributed with code within the npm OSSECO (69.3%) is bigger than the percentage of interviewees that identified themselves as playing all of the roles or the maintainer role in DQ2 (38.5%). Therefore, this could indicate that there is a portion of the interviewees who were not contributing with code by the time of the interview and thus identified themselves as users, but had the contribution experience within the npm OSSECO.

Since code contribution is not the only contribution that a person can make within the npm OSSECO, DQ4 asked interviewees to indicate for how long they

would be making (and if they have ever made) comment contributions or opening issues. From the interviewees, 46.2% claimed that never contributed with comments or opened any issues. In addition, 23.1% indicated that made this kind of contribution for more than 7 years, 15.4% indicated that contributed for 3 to 5 years, 7.7% for 1 to 3 years and 7.7% for 5 to 7 years.

Considering DQ5, which asked about for how long interviewees have been contributing with open source in general, 38.5% claimed that have never contributed with open source, 30.8% claimed to have contributed with open source for more than 7 years, 15.4% claimed that contributed with open source for 1 to 3 years and 15.4% claimed that contributed with open source for 5 to 7 years.

Technical Questions

TQ 1: In an open source software ecosystem, projects may depend on other projects or have interfaces between each other. In those cases, could you tell a bit about how this would affect relationships between projects when breaking changes happen? In this question, interviewees had different opinions regarding what would happen in the case of a breaking change happening. All of them recognized the fact that those changes happen from time to time in packages that are part of the npm OSSECO. However, most of them said that what they would do to address this change would depend on two major factors: (i) the impact this dependency has on their project; and (ii) their resource availability to address this change. Interviewee 13 stated the following:

Interviewee 13

*“I think it depends on the project scenario and its moment. If a dependency, I do not know, was updated and caused an issue in the code and it is something that can be quickly solved, you fix your project and that is it. **The problem is that some updates basically break all of the project’s code.**”*

Therefore, to understand the first factor, we would need to understand from the interviewees what would be the possible impacts they identify that a dependency can have in their projects. Interviewee 01 gave the example of a situation in which

a framework received an update. Because of this update, anyone interested on using this framework's latest version would have to update a portion of their code. Since his/her project was based on this framework, he/she had to decide along with his/her coworkers if they would update it or not. Therefore, they analyzed their needs and identified that a feature that was present in the new version, and that was not available on the old version, was necessary for them.

Interviewee 01

*“Recently I had to do this: update a framework. **We needed new feature that was not available in the previous (version).** We had to update it and did not have much choice since it would be update the framework or build another application using another framework.”*

Moreover, there are other situations that can occurs and influence the evaluation regarding the impact those breaking changes can have on a dependant project. Occasionally, dependency packages can be deprecated and their development can be completely discontinued. Interviewee 02 gave the example of a situation in which, since a package was discontinued, his/her project had to replace this dependency for another one which would have its maintenance guaranteed.

Interviewee 02

*“What we can do is to change the dependency in case the update does not satisfy our needs... **We are currently replacing momentJS because it was discontinued.**”*

When asked in a follow-up question, interviewees stated that security issues would also be an impact. Since dependencies are packages developed by other developers and projects usually have several dependencies, it is not always possible for a project team to verify all of the dependencies codes to search for vulnerabilities. Therefore, Interviewee 05 raised a concern not only about the possible vulnerabilities in dependencies, but also with the dependency cascade. The dependency cascade would happen when a package A, depending on a package B, would also have package C as its dependency since package B depends on C. In this case, regardless of package A not being directly dependant on package C, package A could also suffer

from security problems since package C's code is embedded in one of package A's dependencies.

Interviewee 05

“What worries me the most in this dependency cascade is malicious software because you do not have control over what other people are doing despite of npm active participation in malicious software detection... And another thing that really worries me in this dependency matter is about the maintenance of these dependencies of dependency packages. For example, package X that has package Y as its dependency but package Y is not maintained anymore.”

Considering the impact this dependency has on their project, interviewees would opt out for one of three possible approaches: (i) replacing the dependency; (ii) keeping an old dependency version; or (iii) proceeding with code changes in order to update the dependency. Interviewee 02 gave the example of a situation in which he/she and his/her coworkers decided to replace a dependency. Interviewee 01 stated that the most common reaction to the breaking changes is to keep an old dependency version until the moment maintainer considers the update necessary.

Interviewee 01

*“So, we **only do it (update it) when it is necessary**. If an old dependency is working fine, with no performance issues and our needs are fulfilled, I tend to keep it in the current version since I am not refactoring the code at the moment.”*

Considering the code changes, it would be also possible that package teams consider those changes acceptable and proceed with them. Interviewee 12 gives an example of a dependency that he/she would have to update his/her code in order to use this dependency's new version. Moreover, he/she discovered some issues in the new version and even reported them to the package maintainers who addressed those issues, providing a new version containing the fixes.

Interviewee 12

*“I have been through a problem like this recently, around five months ago. I say it was recent because it was even an issue that I had while exporting reports. What happened was: **a framework that I use, it was updated**, and, therefore, I would have to implement an update in my code to export a report. Until this moment, it was everything ok. The framework was updated and the code regarding report exportation was also updated. **However, there was an internal problem, within this report exportation package in which its dependencies were not updated, which is related to the fact that this package depends on other packages**. So, at the same time this is good, it might also be bad... We opened an issue on GitHub, we reported where the problem was. We could also correct it but I had no time to do this at that moment. **It was quickly fixed by the developers, it took three day, I think.**”*

TQ 2: In an open source software ecosystem, interactions between developers, users, organizations and artifacts are very important to the ecosystem development. Can you give an example of how those interactions are affected by power relations? In this question, interviewees gave examples of interactions that they identified to have been affected power relationships within the npm OSSECO. Interviewee 09 reported a situation that would be common within this OSSECO. According to him/her, in the npm OSSECO, there is what he/she called an inverse power relationships situation. This situation would occur because of the fact that the maintainers, on the opposite that we could consider at first, suffer a lot of pressure from the users and collaborators regarding code changes in packages. Interviewee 09 even compared those who would put pressure on maintainers to “vampires”:

Interviewee 09

*“Let me give you the the typical relationship that there is in the npm ecosystem. Okay, **it’s the inverse of a power relations**. So, I use these metaphors, okay. So **the vast majority of the people that consumes npm modules are vampires. They suck blood out of***

maintainers.** So the power dynamics is that once you expose yourself on npm, you basically need to fend off vampires at every corner, you become very attractive to vampires. And **if you don't treat yourself very carefully, and start to say no to a lot of people, they will write your code."

Therefore, the interactions between the maintainers and those who use or collaborate with the package would be directly affected by this power relationship. However, this was not the only situation reported by the interviewees.

Interviewees also reported how the power relationships could affect the maintainers act of addressing changes and issues. Since the maintainers are "the owners" of the packages, it would be possible that those maintainers ignore community requests to follow a particular road-map. However, interviewees consider this behavior to be damageable to the OSSECO community and claimed that this behavior would probably be punished. This OSSECO community punishment would probably come through the package abandonment. Interviewee 02 says as follows:

Interviewee 02

*"Since they developed it and did it for the community, **I think they would have the decision-making power about the package's directions.** It is an interesting question since when a given open source tool gets so big that becomes even more important than the organization itself. **In this case, it brings the question about if they would have the right to make decisions ignoring the community... In this situation, I would not agree with this.**"*

TQ 3: In an open source software ecosystem, developers, organizations, and artifacts might be known by their reputation (either positive or negative). How would this reputation affect its lead? In TQ3 interviewees were asked about the developers, organizations and artifacts reputation and how this would affect the relationships within the OSSECO. Therefore, the answers for this questions were different depending on if the interviewees were referring to developers, organizations or artifacts.

Considering the developers, interviewees were asked in follow-up questions how the reputation would affect the developers in different situations. However, those different situations were related to contributing with packages. In the case that a developer had a great reputation, being known by his/her technical skills and prior contributions, interviewees stated that it would matter, but would not define if a contribution would be accepted or not. Interviewee 05 stated that if a well known open source contributor open a pull request to his/her project, he would evaluate the contribution similarly to how would do if an unknown developers opened the pull request. However, he/she also clarified that he/she would probably take more time to explain to a well known developer why his/her pull request would not be merged to the main code.

Interviewee 05

*“It is possible that this happens because, for example: a popular developer opened a pull request to my package. **I will evaluate what he/she did and it is like, the developer is a technical reference, so I am going to be more tempted to accept his/her contribution that I would be in the case of an unknown developer.** So... yes, reputation counts but I try to evaluate if that pull request makes sense, for example: a popular developer opened a pull request to my package but that contribution does not make any sense for the package scope. So I would reject it. Maybe not reject it in a rude way. I would explain ‘oh, this contribution does not make sense for package’s scope at this moment so I am going to reject it’.”*

However, the interviewees also mentioned that having a good reputation could help for the pull requests to be evaluated. According to them, considering that in some cases there are many opened pull requests, pull requests opened by known developers would be evaluated first. Occasionally, the fact that a maintainer would personally know those developers would also influence the pull request evaluation. Interviewee 03 said the following:

Interviewee 03

*“Popular, reputable, authors are quite likely to get invited to know each other. You get like invited to speak at conferences. **So that you meet people face to face. So you have built a sort of a network of like, much more personal relations than if you would just interacting online.**”*

However, interviewees could also have a very low reputation, not being known yet by package maintainers. An example of situation in which this occurs is when newcomers try to contribute with a package within the npm OSSECO. Since they have never contributed before, it would be possible that their pull requests would have a higher chance to be rejected. This might be true, but according to the interviewees this rejection would not be related to the fact of the developer being a newcomer. Actually, it would be related to the fact that he/she might not know the contribution rules yet and, therefore, his/her code would be rejected by not respecting one of those rules. Interviewee 04 said the following about the newcomers contribution:

Interviewee 04

*“I think that this contribution being authored by a newcomer, or not necessarily a newcomer since maybe a newcomer would not be capable of making a contribution, but an unknown developer and a tech lead. Since their contribution would help to improve or evolve code, but not going away from the package’s scope, **I would accept the pull request from whoever it was since this pull request made sense to the package. If it did not make sense, I would reject it no matter who the author was.**”*

Regarding the organizations, interviewees considered that the organizations reputation could affect decisions regarding using packages or not. Occasionally, organizations either sponsor a given package that is important to their business goals achievement or make packages developed by internal teams available to the public. Whenever this occurs, developers would be more motivated to use those packages than others that could provide the same features. According to the interviewees,

this preference is related to the fact that while those organizations sponsor packages or maintain their own public packages, these packages are more likely to have an active community contribution and a regular maintenance routine. Therefore, those packages would be more reliable than those who are only community-backed and that could be discontinued at any moment. Interviewee 05 states:

Interviewee 05

*“These organizations are participating. They are putting money and people to maintain that package. That is important for me since **I feel more confident to use a package that organizations maintain or at least sponsor**, that they can possibly not be maintaining, but are paying developers to do that...”*

The artifacts would have their reputation affected either by the organizations that sponsor them or by their history. We previously reported what interviewees said about the organizations effect on the artifacts reputation. However, artifacts can also have their reputations affected by their issue and fixing history. According to the interviewees, packages that constantly present security issues would be candidates to be replaced as package dependencies. In addition, interviewees would only consider this replacement in case those packages that presented some kind of security issues do not have an active and regular maintenance history. Otherwise, interviewees would even consider packages that, despite of having security issues, regularly provide fixes and have an active development community, a package that should be maintainer as a dependency. Interviewee 11 states:

Interviewee 11

*“I do not know. It is like: when a lock is broken, we put a better lock. **So this package is likely more protected now that this security breach was discovered since they would have more resources to make it secure. Of course, at a first moment our trust in this package get shaken up:** ‘how am I going to trust in something that has this security issue? How can I know if my code is really secure?’ **However, as time passes, we get back to use that package if there is no other package to replace it.**”*

TQ 4: How did you experience power relations within the open source software ecosystem? Could you please share your experience? In this question, interviewees were asked to report their experience with power relationships within the npm OSSECO. It is essential to remark that OSSECO community members are more likely to perceive situations related to coercive power relationships. This occurs because of the fact that the punishment is usually something that the OSSECO community members tend to observe as being different from the usual open contribution environment related to an OSSECO.

Interviewee 03 reported a situation in which, to avoid legal penalties, the npm Inc. renamed a package. This package's maintainer decided to name the package with the same name a messaging application had by the time. However, the company that owned this messaging application was alerted about this package name and requested a name change. Since the package maintainer refused to change his/her package name in time, npm Inc. renamed the package as it wished. He/she stated the following:

Interviewee 03

*“So what happened is, there was this developer, he’d written loads modules. And **he/she had this module called a given name**. And then this, which wasn’t like a particularly big, it was just a small thing. Like, I mean, it was something he used, I don’t think it was like, hugely popular. **But then there was this quite well known chat app with the same name**. Like, not as big chat app or something, but it’s like this was a few years ago, you know, there was like a minor chat app. **But it had a trademark on its name. And contacted it tried to, like, tell him not to use it. And contacted npm** and then npm sided with the cop, the trademark holder. I mean, I think it could have been maybe resolved more amicably, but that maintain it was just like, ‘no’, like, ‘I’m not giving it’ like ‘no fuck you’, like ‘go away’, like ‘you can’t take my module’. **And npm, like, overruled us, and it gave the module another name.**”*

TQ 5: In those relationships, how do the actors benefit from the power relations effect? What are the challenges power relations bring into the relationships? In this TQ, interviewees mentioned the benefits and challenges they were able to identify regarding the power relationships within an OSSECO. Regarding benefits, since most of the interviewees focused on the dependency between packages, the benefits brought by those interviewees also had connections with this dependency. Interviewee 07 identified the development benefit of having this dependency relationship, in which those who develop would have their coding time reduced by their dependencies:

Interviewee 07

*“There is a huge benefit because it (dependency) kind of encapsulate everything and **sometimes you need to code something that if you would do it by yourself, by not using a dependency, you would spend much more time on coding it and would not be optimized.** Because usually those codes, when they are uploaded to the registry, they are optimized already.”*

However, we also had an interviewee who was not able to identify any benefit. Interviewee 05 stated that power relationships within an OSSECO do not bring any benefit since they would create a strong dependency and the relationship would be no longer of collaboration:

Interviewee 05

*“In my opinion, no power relations would benefit anything since people would become strong dependant on others and, in my opinion, in collaborative development the people can not feel hostages of their dependencies. **So I only recognize the power relations as harmful relations.**”*

Considering the challenges, Interviewee 08 reported a situation in which the ownership of a package could affect the dependants. He/she states that, if the package owners would start to try to dictate how a package would be used and even interfere in the dependants package, this would be an issue:

Interviewee 08

“If a package maintainers would start to somehow dictate how the package should be used. So, instead of just suggesting it, that would be no problem at all, it is common, they would dictate and even interfere in dependant projects, so **there would be a huge problem.**”

However, other challenge has been identified by Interviewee 06, who stated that it is really difficult for people who are new to the OSSECO to have an influential voice. This challenge is mostly related to the fact that maintainers who contribute to the OSSECO for a long time might have a great reputation and, therefore, in the case of an argument regarding some development, their voice would be more likely to be heard. He/she stated:

Interviewee 06

“It’s really difficult for somebody brand new to break in and have an influential voice. Right? Especially the more opinionated and vocal, the existing maintainers are, right? It becomes really difficult for somebody to come and say, well, well, no, I think it, I think it should be done this other way. And here’s why. **You’re having these, these top maintainers come in, and they’ll just like, make a statement, and it just sucks all the air out of the room. Right? It just, you know, it’s no more conversation, you know, they have spoken.**”

TQ 6: In an open source software ecosystem, projects tend to have their own rules and best practices defined. How are the rules and the code best practices defined and how do they affect the relationships within the ecosystem? In this question, interviewees mentioned that the definition of rules within the npm OSSECO may vary from one project to another. They also mentioned that these rules definition is a process that is not usually open and public. However, they have mentioned that projects that have defined contribution rules would be more interesting to contribute with. This would happen since communication models and the rules of how the pull requests should be opened would be clear to anyone who would like to contribute. Since those rules are open and defined, pull

request evaluation would be less subjective. Regarding the process of defining those rules which is not public, Interviewee 12 stated the following:

Interviewee 12

“It vary from a developer to another, just like it vary from a package to another. What is the unit test percentage? This is defined by the project owner. It is very specific, each package has its own rules.”

Regarding the fact that those rules would help contributors, Interviewee 06 even said that when newcomers ask him/her what packages they could contribute with, he/she would indicate those who have defined rules and best practices for contributing:

Interviewee 06

“For me, a well defined contributors guide that kind of really spells out, you know, here’s what you need to do to contribute, you know, here’s what we expect in commits. Here’s what you know, how we review our, you know, review pull requests and things that lay out the criteria for the this is what we will accept, this is what we won’t accept. In my experience, having that well defined in those rules, kind of well established, and laid out objectively, rather than very subjectively, actually makes it easier for more people to get involved in the project. It the ones that have been, the more the more difficult to get involved with are the ones that do not have that those rules laid out, that aren’t documenting them. And the ones where I’ve seen that contribution process go sour, had been ones where those rules in that have been very subjective because they weren’t written down. There was nothing you could point to, to say, Well, wait, I thought this is how it worked. Right. And a lot of the, you know, for me, you know, as I’ve been mentoring new, you know, new developers coming into it, I’ve always, you know, encourage folks go look for the projects that have a very well defined contribution guide.”

5.4.5 Analysis

In the analysis phase, the results shown in the last section were analyzed to extract the answers to our study RQ. Therefore, it was needed to define a method to analyze those results. The open and axial coding (STRAUSS et al., 2007) were chosen to be executed in this study as the activity of categorizing sentences in the responses and then identifying the relationships between those categories. Therefore, information would be properly extracted from the responses and used to answer our RQ.

During the analysis phase of this study, it was not possible to determine a relationship between different roles and different perspectives about how the power relationships occur within an OSSECO.

Open and Axial coding

In the open coding, the responses for TQ2 and TQ3 were read and then codes were linked to sentences contained in those responses. Those codes would represent the ideas that could be extracted from those sentences and would be used to categorize the sentences. In this procedure, we did not define a set of terms that could be coded, the coding procedure was completely free.

During the open coding step, 34 codes were identified. **Dependency update frequency** and **reputation** codes were the most quoted with 23 quotes. The codes and the quantity of times they were quoted can be seen in Table 5.8. The codes were ordered by the quantity of quotations in a descendant order. In the cases of codes that had the same quantity of quotations, they were alphabetically ordered. It is possible to make some analyses considering the identified codes. It is interesting to remark that all the codes were obtained directly from the responses reading process, therefore, we did not searched for previously defined codes in the sentences.

In the axial coding, the codes gathered in the open coding step would have their relationships identified and then they would be grouped. In this study, we defined three relationships that we should group the codes by: Dynamics, Entities, and Actions. However, during the axial coding process, there was identified one more relationships: Factors. In this study, it was defined that a code could only be linked to one relationship. The information extracted was used to answer our RQ.

Table 5.8: Codes identified during the open coding step.

Code	Quantity of quotations
Dependency update frequency	23
Reputation	23
Dependency between projects	19
Contribution quality	14
Change package dependency	13
Dependencies of dependency	13
Technical leadership	11
Employment relation	10
Maintainer	9
Organization	9
Organization-backed package	8
Fork the project	6
Project forking	6
User	6
Breaking change	5
Project best practices	5
Communication	4
Online course	4
Influence collaboration	3
Role migration	3
Sponsoring	3
Collaborator	2
Conference head	2
Foundation	2
Power burden	2
Board	1
Board member	1
Community building	1
Conference	1
Disowning project	1
Foundation member	1
Give money	1
Share knowledge	1
User block	1

RQ1 and RQ2: What are the power relationship forms within an OSSECO? In what situations those power relationships take place in an OSSECO?

During the analysis phase, it was possible to identify situations in which the power relationships would occur within an OSSECO. The codes that represent those situations are part of the Dynamics relationship defined during the axial coding pro-

cedure and can be seen in Figure 5.8. Therefore, it was possible to identify and describe those situations using the information provided by the interview interviewees. Moreover, from the 9 identified power relationship dynamics, the studies reported in Chapter 4 and Section 5.2 had already identified and described 5 of them. However, the other 4 power relationship dynamics have not been identified and described previously. The power relationship dynamics identified using the interviews' information are as follows:

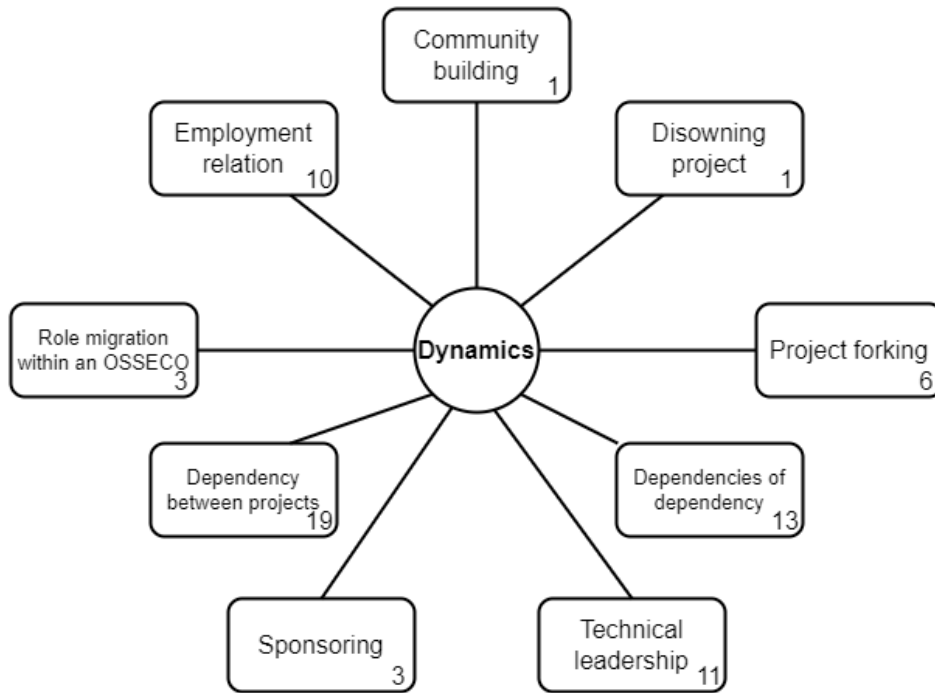


Figure 5.8: The axial coding regarding the Dynamics.

In an OSSECO, its community and the interactions within its context are essential to the OSSECO's evolution. However, when an OSSECO does not have an active community to keep maintaining its development, it is possible that some entities act as community builders to attract developers interested in maintaining the OSSECO. This power relationship dynamics is the **community building** dynamic. Entities (that can be either foundations, boards, organizations, or maintainers) use their reputation to attract developers who have never contributed with the OSSECO or to bring back those who abandoned it. Since those entities use their reputation to exercise power over other entities, this power relationship dynamic is a *referent* power form dynamic. Interviewee 10 reports a situation in which he/she contributed to the community building:

Interviewee 10

*“...I put our governance model there, **I went to work building community and getting promotion around it and getting people involved.** I built like an internationalization community and a website community to make a nice website.”*

Since in an OSSECO projects tend to have interfaces between each other, having dependencies on projects that address specific development goals, this dependency between projects is always present in this context. However, occasionally, those dependencies can represent a situation in which projects depend on projects that depend on other projects. Therefore, this projects inter-dependency could create a dependency cascade that could affect the power within an OSSECO since a project that is an indirect dependency to another project could also affects its development. This **dependencies of dependency** power relationship dynamic is a *referent* power form dynamic. Since a project would consider that another project not only addresses the development goal, but also attends to the same quality demands, creating an identification between those projects.

That same situation also refers to the previously identified **dependency between projects** power relationship dynamic. The difference between those two power relationship dynamics reside in the fact that **dependencies of dependency** refers exclusively to the indirect effect that the dependency cascade has on the OSSECO. In contrast, **dependency between projects** refers to the direct effect this dependency relationship can have on the OSSECO. Interviewee 03 reported how dependencies can scale up with **dependencies of dependency** while Interviewee 11 states that dependencies have the power to insert or remove code whenever they want in **dependency between projects**:

Interviewee 03

*“But then **when things start depending on the things that depend on**, so when it gets like two or three layers deep, then the chance that one of those things becomes really popular. **And now you have, like 10s, or hundreds of 1000s of, or you could have huge number of people that are using your thing.** But most of them*

have no idea what their thing is.”

Interviewee 11

*“...in the project, the dependencies can’t order anything to anyone, **but they have the power to insert or remove features. So we become dependant on it.** It is like we were really dependant on it but it can’t tell us what to do. But we can’t do more than it offers.”*

However, the projects dependency and usage is also related to the fact that a group of developers, usually organized by the maintainers, have to develop those projects. Occasionally, those projects are developed by a small group of maintainers who receive a lot of change requests. Therefore, they have an OSSECO community pressure on them to solve problems or to evolve their project accordingly to the OSSECO community goals. However, this pressure can make maintainers abandon their projects and even the OSSECO. This **disowning project** power relationship dynamic is a *coercive* form of exercising power in which the maintainers punish the OSSECO community by abandoning their project development. This power relationship dynamic can also occur whenever the technological platform, for some reason that could be a legal issue, removes the ownership of a project that is considered essential by the OSSECO community. Therefore, the technological platform would guarantee that this essential project would remain available for the OSSECO community. Interviewee 03 shares the moment he/she decided that it would not be possible to keep maintaining his/her projects and disowned them all:

Interviewee 03

*“I’m just like, no, like, or it’s quite like, you know, like some stranger somewhere who is trying to do something and they need help with something that I wrote. And it’s like, I’m shouldn’t be on the firing line for like helping every single person. **So I realized what I needed to do was just disown all of them all of those modules. So so what I did is I transferred all of the modules, I made another npm account, and called no persons modules.** And I transferred all of these modules to that account, and then I forgot the password to that account”*

In the **employment relation** power relationship dynamic, an organization hires a collaborator or a maintainer in order to have dedicated developers to help updating community-backed projects. Occasionally, those developers could also be responsible for maintaining organization-backed projects publicly available within the npm OSSECO. Therefore, this a *reward* form of exercising power within an OSSECO since developers identified as technically able to update or maintain projects are rewarded with money to help the organization. Interviewee 09 reports his/her experience regarding contribution with open source and employment relationships:

Interviewee 09

*“So you need to you have some of your motivation to be to be there. Okay? Okay. So once you have that motivation, then this is a massive jumpstart for you. **If it wasn’t for open source, I would not have the career that I had. Like, I’m totally grateful for my career, because of my open source activity.** I would not had the chance to do what I’ve done so far.”*

However, in npm OSSECO projects it is possible that users or collaborators desire to have code changes to address their needs. Whenever this occurs, there are two approaches maintainers can have regarding those requests: (i) addressing the requests, either by proceeding with a code change or responding the comments; or (ii) ignoring the requests since it is not a change considered adherent to the project roadmap. Occasionally, users or collaborators can be unsatisfied considering the maintainers’ response to their requests. Therefore, those users or collaborators could proceed with the **project forking** power relationship dynamic, in which a project is forked (which means it is cloned), and then the users or collaborators assume the forked project’s development decisions. Since those users or collaborators now have this power of making decisions within this forked project, they punish the original project maintainers by removing some of their power. Therefore, this is an example of a *coercive* form of exercising power within an OSSECO. Interviewee 10 shares a situation in which, because of a conflict, his/her group decided to fork a project:

Interviewee 10

*“They removed two committers from the project. And they without speaking with anybody announced their new governance process, which they called Pure consensus, but included nobody but their own people. And, and we were like, oh, okay, I think they just told us to go fuck off. **So we, we, at that point, did fork.**”*

Since many collaborators within an OSSECO seek the role migration to play decision-making roles, those who can provide this role ascension can exercise power on those who desire it. Therefore, contributions are evaluated and since they are considered not only valid, but also essential to the project, they can lead to this role ascension. However, there is also the possibility of a maintainer who performed a change that was not desired by the other maintainers of a project to lose his/her privileges and be demoted. Whenever this code change is not only not desired but considered harmful in anyway, the maintainers can demote another maintainer by removing his/her privileges. Therefore, **role migration within an OSSECO** is a power relationship dynamic in which we can have entities exercising power either in a *reward* or in a *coercive* form. Interviewee 03 reports how he/she decided to promote a collaborator to maintainer since he/she was not interested in maintaining a given package alone anymore:

Interviewee 03

*“And there was like this. So this, this guy turned up. And he’s like: ‘hey, it doesn’t look like yours Event Stream anymore. I still find it quite useful. **Do you mind making me the maintainer?**’ And I’m like: ‘Oh, great. Someone wants to help me for once. **So it’s yours’.** I give it to him. And I’ve done this a couple of times, I’ve done this a couple of other times as well. It’s like a pretty, it was a pretty widespread practice and the like early.”*

Occasionally, organizations depend on a community-based project to achieve their business goals. Therefore, they have a high interest in this project’s maintenance and update frequency. However, those organizations might not be interested in hiring developers to work on those projects but would not consider taking the risk

of having this project discontinued. Therefore, those organizations would sponsor those projects by giving them money and having their guarantee that the project is not on risk of being abandoned. Since organizations are rewarding projects that are essential to their business goal achievement process, this **sponsoring** power relationship dynamic is an example of a **reward** form of exercising power within an OSSECO. Interviewee 05 states that not only the organizations are sponsoring community-backed packages but that he/she considers this to be important:

Interviewee 05

*“The **organizations** are **putting money and people to maintain that package**. And that is very important to me.”*

Whenever a developer shows technical skills that very few developers have within an OSSECO, it is expected that this developer at some point perform a **technical leadership** power relationship dynamic. Since this developer, usually a collaborator, has the specific technical skills that other might need, those in need developers would request the technical reference developer help. Occasionally, this developer could consider helping those in need developers and, therefore, share his technical skills. However, this developer could also completely ignore the in need developers’ request. Considering that this developer provided the requested help, he/she exercised his/her power in an **expert** form. Interviewee 06 states that because of his/her reputation as an technical reference, it is easier for his/her contribution to be accepted:

Interviewee 06

“My reputation as being a board member, core contributor is, you know, I’m going to get folks attention and, and kind of make it easier for me to contribute.”

Therefore, it was possible to answer RQ1 and RQ2 since we identified 9 power relationship dynamics within an OSSECO considering the interviewees’ responses: *community building, dependencies of dependency, dependency between projects, disowning project, employment relation, project forking, role migration within the OSSECO, sponsoring and technical leadership*. In addition, it was possible to identify

that those 9 power relationship dynamics can be related to 4 power relationship forms: *coercive*, *expert*, *referent* and *reward*. Regarding the role migration within the OSSECO power relationship dynamic, it was the first dynamic in this research in which more than one power relationship form could be exercised in the same power relationship dynamic, depending on the maintainers' evaluation. The identified power relationship dynamics, the power forms they can be related to and the interviewees who reported them can be seen in Table 5.9.

Table 5.9: The identified power relationship dynamics and their power relationship forms.

Power relationship dynamic	Power relationship form	Source interviewee
Community building	Referent	I10
Dependencies of dependency	Referent	I01, I02, I03, I04, I05, I06, I07, I11, I12, I13
Dependency between projects	Referent	I01, I02, I03, I07, I11, I12, I13
Disowning project	Coercive	I03
Employment relation	Reward	I04, I05, I07, I08, I09, I11, I13
Project forking	Coercive	I04, I10
Role migration within the OSSECO	Coercive, Reward	I03, I12
Sponsoring	Reward	I02, I05
Technical leadership	Expert	I03, I05, I06, I07, I08, I09, I10, I11, I12

RQ3: Who are the entities that participate in those power relationships within an OSSECO?

Considering the interviewees answers, we were able to identify the entities that are part of the OSSECO and that are involved with the power relationship dynamics that occur in this context. Those entities could exercise power over other entities within an OSSECO or suffer the effects of a power relationship dynamic. Therefore, to understand those entities, we present a brief description of each one of them, based on the interviewees' answers. Moreover, those entities can be seen in Figure 5.9.

A **board** is an entity that represents a group of developers, usually highly skilled, that gather to help organize the OSSECO. It can regard technical decisions, patterns and best practices but can also be involved with collaboration promotion within the

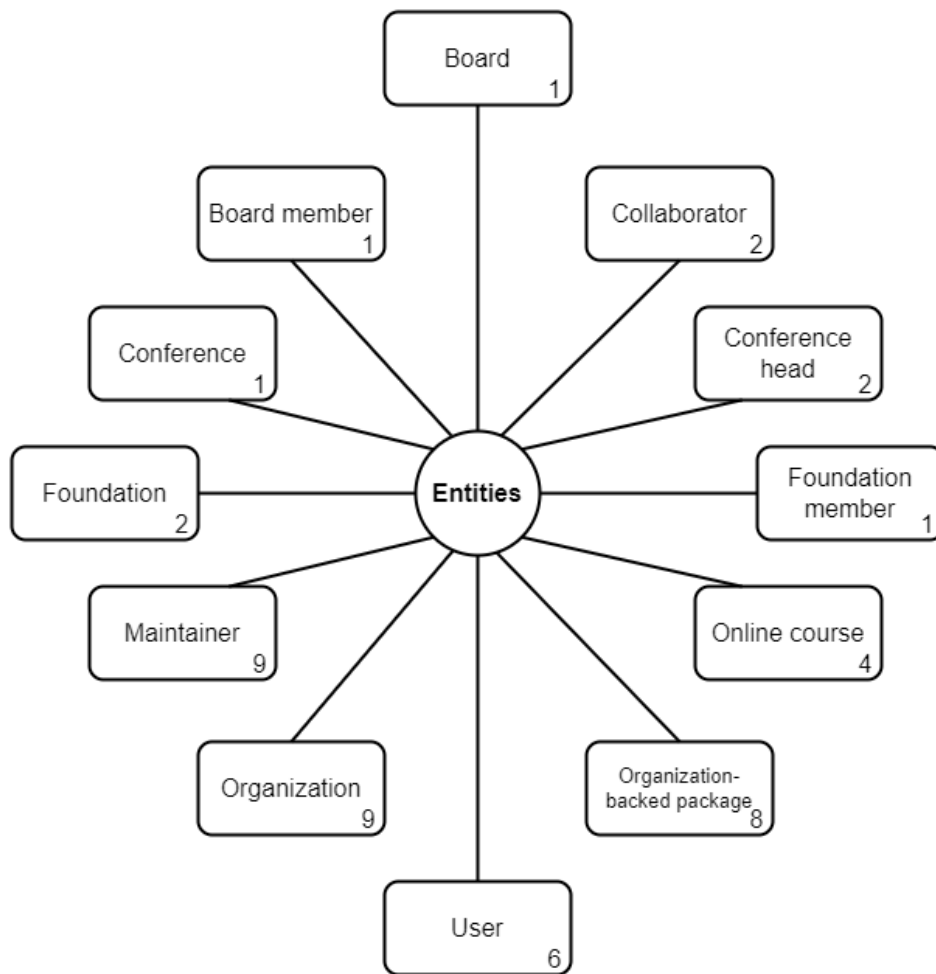


Figure 5.9: The axial coding regarding the Entities.

OSSECO. The **board member** is an entity that represents a person that is part of a **Board**. He/she can be chosen to be part of it considering his/her reputation, technical skills or even past contributions. He/she has influence over decisions that are made in a board and that can influence community building and best practices definition. Interviewee 06 reported what he/she considers to be an effect of being a board member. It is important to stress that Interviewee is a member of a board related to the npm OSSECO:

Interviewee 06

“Happens all the time. Just my own experience: you know, I’ve been around Node JS long enough and contributed enough to it that pretty much any, if I went to any of the repositories, you know, for any project on npm, and said: ‘Hey, you should do it like this. Right?’ My reputation as being board member, core contributors is, you know, going to get folks

attention and, and kind of make it easier for me to contribute”

The **collaborator** is an entity that contributes with code, comments or both, and that does not have privileges within the project. He/she can also fork a project if that project’s maintainer does not address his/her requests or ignores his/her pull requests.

A **conference** is an entity that represents an event in which OSSECO community members gather to share information and to discuss about the OSSECO development. It can influence and promote collaboration within the OSSECO, depending on how its program is defined. The **conference head** is an entity that represents a person that has influence over a **conference** entity. He/she has influence over the conference program, being able to define what is discussed in a conference. Therefore, he/she would be able to influence how the collaboration is promoted by a conference within an OSSECO. Interviewee 10 shares that he/she started a conference within the OSSECO and created conference conference communities. By creating them, he/she was also one of the **conference heads**:

Interviewee 10

“...I started a conference, and I ran a lot of the early Node JS conferences and created a lot of conference communities. I also helped start Node School as well. A couple of things...”

A **foundation** is an entity that represents a non-profit organization which is created to help organize the OSSECO. This entity can help define best practices, development patterns and even the technological platform development roadmap. It can influence the contribution promotion process. The **foundation member** is an entity that represents a person that takes part in a **foundation**. He/she can influence the decisions make within the foundation and, therefore, influence best practices definition, development patterns and the technological platform development. Interviewee 10 also remembered the situation in which a foundation was created and two of them would almost fight against each other:

Interviewee 10

“...they almost immediately decided to join a foundation, but for quite a while held out hopes that they could basically control how that

process was going and could continue to control the project inside of the foundation. Like basically there would be a Node JS foundation and an IO JS foundation that they were kind of fighting against.”

The **maintainer** is the entity responsible for maintaining a project. Therefore, he/she has the privileges to define the code changes that are merged with the main code. However, this entity can suffer a lot of pressure from project’s stakeholders since there would be a desire to please those stakeholders. Interviewee 06 mentioned the maintainers and how the first impression seems to be important to them:

Interviewee 06

*“You know **how the maintainers react to that first PR, or that first issue you open, that decides whether or not you’re going, you’re going to be actively involved in the project, continue or not, because it’s that first impression of how the how the maintainers respond to me.**”*

The **online course** is an entity that represents a learning platform that shares knowledge with those who subscribe it. It can influence the contribution an user or collaborator is planning to make. Interviewee 08, when asked about how did he/she proceed when a technical issue is identified, stated:

Interviewee 08

“(I check) some courses that exists on the Internet, they help me a lot.”

An **organization** is an entity which represents a company which has business goals that participates in the OSSECO desiring to achieve those goals. This entity can be involved with different power relationship dynamics, but it is the only entity that uses money to exercise its power. Interviewee 08 talked about a big organization and how he/she is more confident in using their package:

Interviewee 08

“And it also counts that it (package) was developed by Facebook developers. Because of that and because it is widely used, having a large community using it I feel more confident in using it).”

Organization-backed package is an entity that represents a package that is maintained by an organization. Therefore, this package usually has a private roadmap to follow, which could lead to OSSECO community's requests being ignored and even rejected in case they do not follow this roadmap. It is important to remark that those packages might be created within organizations, as private packages, and then be published in the registry and had their code publicly available on GitHub, becoming an OSSECO project. The previous reported interviewee 08 with a Facebook package portrays the idea of an organization-backed package.

An **user** is an entity that does not contribute with code. His/her contributions are limited to the comments he/she can make and the issues he/she can open. He/she can also fork a project if that project's maintainer does not address his/her requests. Interviewee 02 identifies itself as having a "only user" experience:

Interviewee 02

"I'm a software developer since 2016. I started as a front-end developer with JavaScript and now I'm a back-end developer, also using JavaScript. My experience with npm is an only user experience."

Therefore, to answer RQ3, we were able to identify 12 entities that are involved with power relationships within an OSSECO: *board*, *board member*, *collaborator*, *conference*, *conference head*, *foundation*, *foundation member*, *maintainer*, *online course*, *organization*, *organization-backed package*, and *user*. Those entities and the interviewees who reported them can be seen in Table 5.10.

Actions in Power Relationships within an OSSECO

During the open coding procedure, we identified some quotations that could be related to actions in power relationship dynamics within an OSSECO. Therefore, in the axial coding procedure, a relationship called Actions was created and used to group those codes. Considering the interviewees' answers, we were able to identify and briefly describe 6 of those actions. The actions happen within the power relationship dynamics and between two or more entities and can be seen in Figure 5.10. These actions and the interviewees who reported them can be seen in Table 5.11 and are as follows:

Table 5.10: The entities identified during the field study.

Entity	Interviewee source
Board	I06
Board member	I06
Collaborator	I02, I05
Conference	I10
Conference head	I10
Foundation	I10
Foundation member	I10
Maintainer	I03, I05, I06, I08, I09, I11, I12, I13
Online course	I08, I10, I11
Organization	I01, I03, I05, I08, I10, I13
Organization-backed package	I02, I04, I05, I06, I08, I09, I13
User	I01, I02, I07, I08, I11, I13

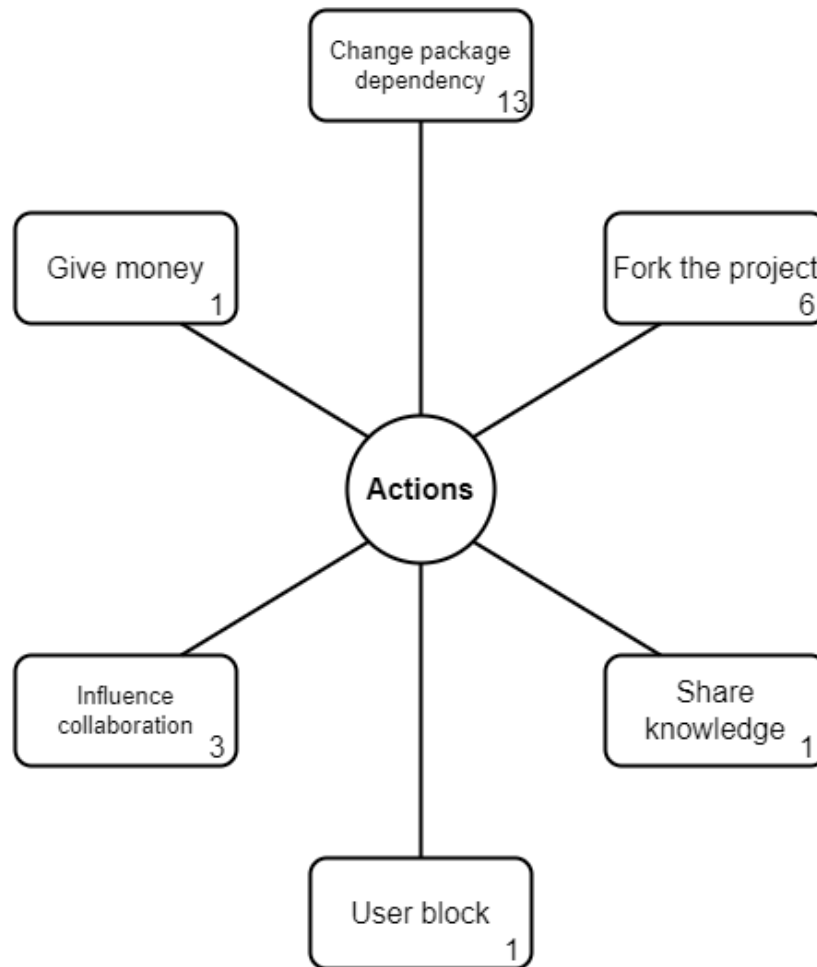


Figure 5.10: The axial coding regarding the Actions.

Change package dependency: it is an action in which an entity, usually a community-backed package or an organization-backed package, decides to change a

Table 5.11: The actions identified during the field study.

Action	Source interviewee
Change package dependency	I01, I02, I03, I05, I07, I08, I11, I13
Fork the project	I04, I10
Give money	I03
Influence collaboration	I06, I10
Share knowledge	I06
User block	I10

dependency and replace it. Interviewee 02 reports this action that is taken when a dependency does not satisfy his/her needs anymore:

Interviewee 02

“What we can do is to change the dependency if the update does not satisfy us.”

Fork the project: it is an action in which an entity, usually an user or a collaborator, since his/her requests and comments have not been addressed or responded adequately, decide to clone the project and develop this forked project to attend his/her needs. Interviewee 10 states that the possibility of forking a project is also related to freedom:

Interviewee 10

*“The primary freedom of open source that is somewhat unique to open source, like as a technology and is like a community and a movement is this freedom to fork, right? **Like the idea that there there is a limit to control and power that anyone has with a public license, because at the end of the day, you can always fork it.**”*

Give money: it is an action in which an entity, usually an organization, rewards another entity, that can be either a maintainer, a collaborator or even a community-backed package. Interviewee 05, as we previously seen in the sponsoring power relationship dynamic identification, mentioned this action:

Interviewee 05

*“The **organizations are putting money** and people to maintain that package. And that is very important to me.”*

Influence collaboration: it is an action in which an entity, that can be either a foundation, an organization, a collaborator or a maintainer, uses its/his/her reputation to make another entity, that is usually a collaborator or a maintainer, to act as it/he/she desires. Interviewee 06 reports an example of this action:

Interviewee 06

*“So that was... and that it’s clear example of how people are using their kind of influence over that ecosystem, to kind of push the entire ecosystem in a particular direction, **and kind of really having kind of undue influence over where things are gonna go.**”*

Share knowledge: it is an action in which an entity, since another entity wants to learn his/her technical skills, decides to teach those skills. Interviewee 06 reports:

Interviewee 06

“And what is common, if you look at all those crafts, what is common is just sharing skills, sharing techniques and people learning from each other...”

User block: it is an action in which an entity, usually a maintainer, because of an harmful or inappropriate request from an entity, that can be either an user or a collaborator, decides to block this entity. Interviewee 10 reports a situation in which users were removed (blocked) from a project:

Interviewee 10

“They removed two committers from the project.”

The Factors Involved with Power Relationships within an OSSECO

During the coding phase, we were able to identify quotations in which factors related to the power relationship dynamics were mentioned. Those factors can be either a prerequisite for a power relationship dynamic to happen, which we can call a power capability, an entity characteristic or a power relationship dynamic effect. There were identified 7 factors related to the power relationship dynamics within an

OSSECO, as seen in Figure 5.11. These factors and the interviewees who reported them can be seen in Table 5.12 and are as follows:

Breaking change: it is a factor related to a change in code that introduces significant changes to features. The significant changes are changes that alter the usage or the results of some features. This change might even mean that a dependant package has to adjust its code to keep using the dependency's new version. It is a prerequisite for the **Dependency between projects** power relationship dynamic. Interviewee 06 states that at some moment, because of the possibility of breaking dependants project a breaking change was not coded:

Interviewee 06

*“We’ve had cases where things that **things that we needed to do in Node JS, code changes and stuff we could not do**, because (it is) one of the most popular modules out on npm. It would break them (dependants), and the **author of the module, the maintainer of the module would just refuse to change things...**”*

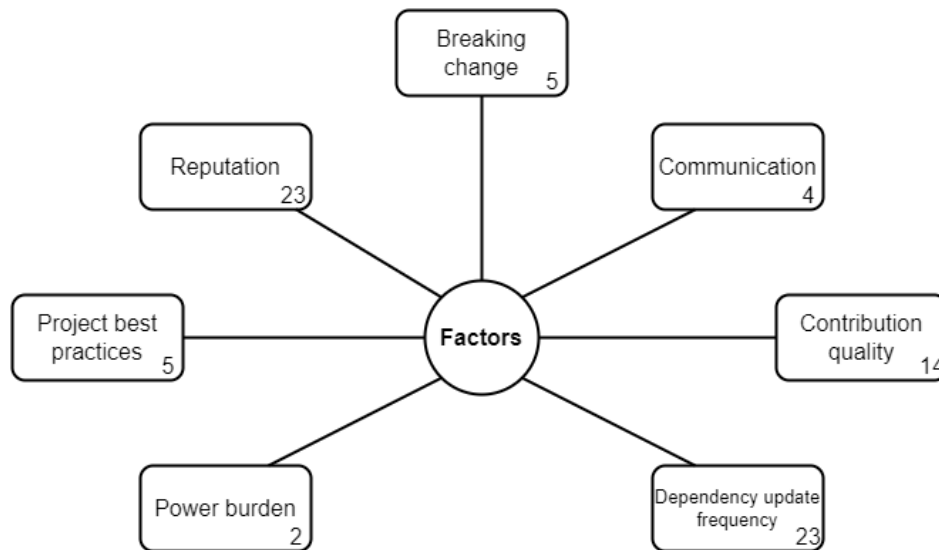


Figure 5.11: The axial coding regarding the Factors.

Communication: it is a factor related to the act of communicating within the OSSECO. This factor is prerequisite for some power relationship dynamics in which the maintainers choose not to communicate with other OSSECO community

Table 5.12: The factors identified during the field study.

Factor	Source interviewee
Breaking change	I03, I06, I07
Communication	I03, I09, I10
Contribution quality	I04, I06, I09, I10, I12, I13
Dependency update frequency	I01, I02, I04, I05, I06, I07, I08, I11, I12, I13
Power burden	I03
Project best practices	I05, I06, I09, I11
Reputation	I02, I04, I05, I06, I07, I08, I09, I12, I13

members and, therefore, are punished by those OSSECO community members. Interviewee 09 even states that experienced OSSECO community members have their own communication approach:

Interviewee 09

*“Because it’s... you will have that most **experienced people in the community have a certain way of communicating on GitHub or online as well.**”*

Contribution quality: it is a factor related to value a contribution can add to a project within an OSSECO. This factor is evaluated and defined according to the maintainers considerations and their opinion regarding if the contribution is adherent to the project roadmap or not. This evaluation can also be affected by the project’s best practices guidelines. Interviewee 04 states, as previously presented, that the contribution quality is even more important for him/her than reputation when evaluating a pull request:

Interviewee 04

*“I think that this contribution being authored by a newcomer, or not necessarily a newcomer since maybe a newcomer would not be capable of making a contribution, but an unknown developer and a tech lead. Since their contribution would help to improve or evolve code, but not going away from the package’s scope, **I would accept the pull request from whoever it was since this pull request made sense to the package. If it did not make sense, I would reject it no matter who the author was.**”*

Dependency update frequency: it is a factor related to the number of times a package that is a dependency is updated. Whenever a project has a very low update frequency, it means that if this project has security issues or bugs identified, they probably are taking long to be solved. However, if a project is regularly updated, it means that any issues could be quickly addressed and solved. Since updates can lead to breaking changes, it is also important to analyze with what frequency breaking changes are introduced within a project. Interviewee 04 reports 2 different situations: a negative situation in which a package is not being updated frequently and a positive situation in which the OSSECO community involvement supports the package update in a frequent basis:

Interviewee 04

*“The **maintenance frequency**, how long since the last pull request opened to this package, and **if the maintainers update the package...**”*

*“...there is also the community participation. **If the community is using it (the package) and commenting on it.** As I see it, OS-SECO community is based on collaboration.”*

Power burden: it is a factor related to the negative effect that holding power can have on an entity. Therefore, it would be related to entities that can hold power in some occasions. Considering the interviewees' answers, the entity identified as suffering this effect was the maintainer. Occasionally, the pressure related to making decisions regarding a project's development can be so burdening that a maintainer abandons the project and even the OSSECO. Interviewee 03 compares having power on a package to the responsibility with children:

Interviewee 03

“It's more like: when you create a package it's more like it owns you, like you've created a responsibility to maintain it. Like having children and like you can't sell your children which is quite like and if you make have a lot of children, it's like is that a problem?”

Project best practices: it is a factor related to a set of project’s contribution guidelines within an OSSECO. Since these guidelines are defined and able to all those who want to contribute with the project, collaborators and maintainers must follow the rules to have their contribution accepted. It can also provide guidance regarding how issues should be reported, affecting also the users who report those issues. Interviewee 06 claims that well defined best practices make a package more attractive to developers:

Interviewee 06

“well defined contributors guide that kind of really spells out: ‘here’s what you need to do to contribute, here’s what we expect in commits. Here’s how we review our pull requests and things that lay out the criteria for the this is what we will accept, this is what we won’t accept’. In my experience, having that well defined in those rules, kind of well established, and laid out objectively, rather than very subjectively, actually makes it easier for more people to get involved in the project”

Reputation: it is a factor related to how recognized a project, a collaborator, a maintainer or an organization can be within an OSSECO. Depending on the entity, this factor can have different meanings. Regarding a project, reputation can refer to the number of projects that depend on it and to a clean history considering security breaches. Whenever the entity is a collaborator or a maintainer, the reputation resides in his/her technical skills and past contributions. In addition, when considering organizations, reputation can refer to owned projects, financial support, past contributions or technical reference. Regarding the package reputation, interviewee 06 claims that a package that has a lot of people using it might be considered to have a great reputation:

Interviewee 06

“This module has been downloaded 30 million times in the past week, then the automatic assumption is: it must be good. Let’s go use that!”

5.4.6 Relationships Between Power Relationship Forms

During the analysis phase of this study, it was possible to identify, in the interviewees quotes, not only the power relationship dynamics, the entities, the actions and the factors, but also relationships between power relationship forms. Therefore, it would be possible to present interactions between those power relationship forms in which one of them would maximize the other or minimize it. Identifying those interactions would be essential to understand how taking determined actions could lead to an entity power being maximized or minimized. In addition, since these interactions are between power relationship forms, they represent a wider perspective regarding the power relationships within an OSSECO, not being limited to the identified power relationship dynamics.

Interviewee 10 reports a situation in which a developer, previously unknown to the OSSECO community members makes a contribution. However, this contribution was essential at the OSSECO moment and started contributing regularly, showing relevant technical skills. Thereafter, this developer became a technical committee lead within the OSSECO. Therefore, using his technical skills, which can be related to the *expert power form*, he was capable of achieving a higher role within the OSSECO and building a reputation, which can be related to the *referent power form*. Since this technical behavior enhances the reputation, we could claim that **expert power form positively affect referent power form**. The interviewee 10 quote is as follows:

Interviewee 10

*“I think that was some of the earliest people maybe... I think that a very good example, is a guy named CENSORED. **He came in and built out the whole build system community, he ended up becoming the TSC lead.** Before IO JS, he had never contributed with Node Core. And he was somebody who definitely had the technical chops to conduct a Node Core. He had built, like, C libraries for Node.”*

Whenever an entity achieves a high reputation within the OSSECO, it is possible that the other entities start to see this entity as a reference. These entities can develop a trust relationship with each other. Interviewee 09 states, when asked about

packages that he/she would choose as dependencies, that he/she usually knows the package maintainers so he/she trust them:

Interviewee 09

*“Usually there is somebody I trust in the chain. I’ve been in the campaign in the community enough that I know who to trust... It’s a tight community. **The person that you can actually trust, like, probably a few hundreds.** But there is a few 100s of people, maybe 1000 I don’t know, that support the whole npm ecosystem.”*

However, trusting in this “tight” group of developers could create a dependency on them, what could lead into a bigger problem in case one of those developers decided to abandon the OSSECO. This abandonment situation was reported by interviewee 02 regarding the MomentJS framework, which was discontinued:

Interviewee 02

*“What we can do is to change the dependency in case the update does not satisfy our needs... **We are currently replacing momentJS because it was discontinued.**”*

Therefore, this trust that come from the reputation, the *referent power form*, could enhance some kind of behavior or action that could represent a punishment to the OSSECO, which could be related to the *coercive power form*. Thus, **referent power form positively affect coercive power form**.

A common relationship between power relationship forms is the one between reputation, which can be related to *referent power form*, and technical skills, which can be related to *expert power form*. When a reputable OSSECO community member tries to contribute with a package, other OSSECO community members relate this reputation to a contribution that would contain a high technical value. Therefore, they are more tempted to accept this contribution. Thus, **referent power form positively affect expert power form**. Interviewee 06 states:

Interviewee 06

*“My **reputation** as being a board member, core contributor is, you know, I’m going to get folks attention and, and **kind of make it easier for me to contribute.**”*

Considering the financial support that is given in a *sponsoring power relationship dynamic*, that is related to the *reward power form*, this support could maximize the reputation, which could be related to the *referent power form* of a community-backed package. This happens because when an organization gives money to a community-backed package, the other OSSECO community members consider that this package will be maintained and, therefore, is more reliable. Thus, **reward power form positively affect referent power form**. Interviewee 05 reports:

Interviewee 05

*“The **organizations are putting money and people to maintain that package.** And that is very important to me.”*

Whenever an entity has to punish another entity, which can be related to the *coercive power form*, its reputation, which can be related to the *referent power form*, suffers an impact and is minimized. Therefore, **coercive power form negatively affect referent power form**. This happens because the *referent power form* is related to the identification with another entity. Since we are talking about OSSECO, which is a collaborative environment, whenever a punishment happen, the punishing entity is seen as acting in disagreement with the OSSECO. Interviewee 06 reports the situation in which collaboration rules are not properly defined and a subjective pull request evaluation would make it more difficult to collaborate:

Interviewee 06

*“In my experience, **having that well defined in those rules, kind of well established, and laid out objectively, rather than very subjectively, actually makes it easier for more people to get involved in the project.** It the ones that have been, the more the more difficult to get involved with are the ones that do not have that those rules laid out, that aren’t documenting them.”*

In the case of an entity that exercises its *legalistic power form* by using its right to proceed with a change against the OSSECO community agreement, the reputation of this entity would be negatively affected. Therefore, **legalistic power form negatively affect referent power form**. When asked about if a maintainer, by owning a package, would decide to dictate how the package development and usage should be done, interviewee 08 claims:

Interviewee 08

*“If a package maintainers would start to somehow dictate how the package should be used. So, instead of just suggesting it, that would be no problem at all, it is common, they would dictate and even interfere in dependant projects, so **there would be a huge problem.**”*

Finally, the power relationship forms, their interactions and the effect they have on each other can be seen in Table 5.13.

Table 5.13: The relationships between power relationship forms within an OSSECO.

ID	Power relationship form source	Power relationship form destination	Effect
PRFI01	Expert	Referent	Positive
PRFI02	Referent	Coercive	Positive
PRFI03	Referent	Expert	Positive
PRFI04	Reward	Referent	Positive
PRFI05	Coercive	Referent	Negative
PRFI06	Legalistic	Referent	Negative

5.4.7 Limitations and Threats to Credibility

In this field study, we performed a qualitative analysis during its analysis phase. When considering a qualitative study, we have to consider not only its limitation, but also its threats to credibility. Those threats to credibility are related to how careful thorough and honest the researchers were during the data extraction and analysis process (ROBSON, 2002). Therefore, it is necessary to understand that the reality is constructed, multi-dimensional and ever-changing. Therefore, researches in a exploratory study would provide an interpretation of someone’s interpretation of a real situation. In the present study, we interpret the interpretation of the npm OSSECO community members regarding power relationships within their OSSECO.

In order to address this threat to credibility, we used different sources of information, inviting potential participants from different countries, who contribute with different packages within the npm OSSECO, with different experience time and skill sets. This approach is what Merriam (MERRIAM et al., 2015) suggests to confirm the emerging results. It is also important to remark that, after the last interview was conducted and comparing the transcripts from the last 5 interviews, there was no new information provided by interviewees. Therefore, we considered to have achieved theoretical saturation about power relationships within the npm OSSECO. This theoretical saturation is related to the scope defined for this study and the goals and questions defined in it.

Finally, as a limitation, the results of these studies were evaluated by two experts: one of them is an expert with more than 10 years of experience regarding software ecosystems and the other one is an expert with more than 10 years regarding open source software. However, these results have not been validated by any OSSECO community.

5.4.8 Conclusion

In this section, it was reported a field study which had the objective of refining the information already gathered in Chapter 4 and Section 5.2.

As a result, we identified 9 power relationship dynamics using this field study results. From those 9 power relationship dynamics, 4 of them were new dynamics that the previous studies did not identify. In addition, we also identified 14 entities, 8 new entities and one entity that was renamed. Moreover, we identified 6 actions in this study. Finally, we identified factors, that would be prerequisites, characteristics or effects associated to the power relationship dynamics. There are 7 factors identified by this study. We were also able to identify 6 relationships between power relationship forms.

5.5 Final Remarks

In this chapter, we reported an exploratory study and a field study with the objective of extracting information from the npm OSSECO community members about power relationships within their context. At the end of both studies, we were able to

identify 9 new power relationship dynamics and to relate them with the 5 power relationship forms. We also identified 14 entities that would be directly related to the power relationship dynamics that occur within an OSSECO. A set of actions that would be performed during the power relationship dynamics and factors that would affect those dynamics were also identified. Finally, we identified a set of relationships between power relationship forms.

In the next chapter, based on the information from this chapter combined with information gathered in Chapter 4, a conceptual model is developed to represent the power relationships within an OSSECO.

Chapter 6. The OSSECO Power Relations Conceptual Models

6.1 Introduction

This chapter reports the process of constructing a conceptual model to represent the information that was gathered in Chapter 4 and Chapter 5. Considering all the information that was previously extracted from both literature and community, a conceptual model based in Valença et al.'s (2017), PRM-SECO model, was built. The methodology designed to guide this study was divided in three phases, as seen in Figure 6.1 and explained as follows:

- *Planning*: this phase had the objective of defining, inspired by Valença et al.'s (2017), the steps to be followed during this study model construction. Therefore, the steps to be followed during the next phases of this study and how those steps would lead to the fulfillment of their objectives are defined in this phase. The author and 4 experts are involved with this phase;
- *Describing the power relationship dynamics*: this phase had the objective of describing the power relationship dynamics that were previously identified in the studies reported in Chapter 4 and Chapter 5. Therefore, the steps defined in the planning phase would be followed and, at the end of this phase, all the power relationship dynamics would be described and a graphical representation, a textual representation and a logical representation for each power relationship dynamic are presented. The author is the only person involved with this phase;
- *Constructing the model*: this phase had the objective of constructing the model that would represent the power relationships within an OSSECO. Moreover,

this model would have to represent the entities that were involved with the power relationship dynamics previously identified and described. At the end of this phase, the OSSECO power relationships entity model (OSSECO-PREM) and the relationships between power relationships model (OSSECO-PReFoM) should be constructed. The author is the only person involved with this phase.

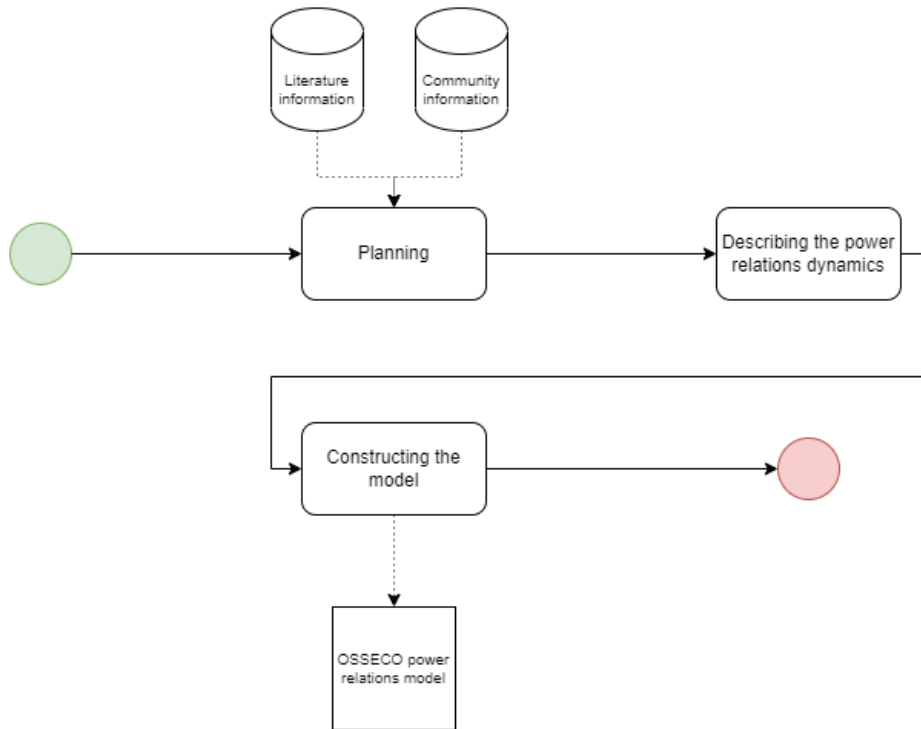


Figure 6.1: Model construction methodology.

This remainder of this chapter is structured as follows: Section 6.2 presents the planning phase for the power relationship dynamics description and model construction; Section 6.3 presents the power relationship dynamics description process; Section 6.4 presents the construction of the OSSECO-PREM; Section 6.5 presents the construction of the OSSECO-PReFoM; Section 6.6 presents the limitations regarding this study; and Section 6.7 presents this study’s final remarks.

6.2 Planning

Prior to the phase of power relationship dynamics description, it was necessary to define the steps that would be followed. Since power relationship dynamic definition was previously provided, we only refined this definition to add a new information: power source.

A power relationship dynamic would be a situation in which power relationship take place. Those situations would have a power source, which would be a prerequisite for that power relationship dynamic to happen. Considering the interactions that are common in the OSSECO context, power relationship dynamics would contain those interactions between the different entities that are part of an OSSECO. In a comparison with Valença et al.'s (2017) power relationship conceptual model, the power relationship dynamics would be directly related to the power capabilities described in that study.

In the case of power relationship dynamics, three elements would always be present: entities, actions and power sources. Regarding the entities, it would always be necessary to have at least two entities, in which one of them holds the power. There would be no maximum limit to the quantity of entities involved with a power relationship dynamic. Regarding the actions, at least one action must be present in a power relationship dynamic, otherwise there would be no interaction and no power relationship. Once again, there is no maximum limit to the quantity of actions that are part of a power relationship dynamic. Those actions could trigger other power relationship dynamics, since they could affect power sources and other interactions. In this description, we would also relate the power relationship dynamics to the benefits and challenges identified in the systematic mapping study reported in Chapter 4. Those benefits can be seen in Table 4.7 while the challenges can be seen in Table 4.8. The power relationship dynamics would have three representations:

- a textual description in which the power relationship dynamic would be explained. In this description, it would also be referenced the community and literature information used as source of information for this power relationship dynamic description. In the cases in which both literature and community were the sources of information, the description should remark each source contribution;
- a conditional representation in which the power relationship dynamics, the entities that are part of them and the actions performed within them are represented using conditional sentences;
- a graphical representation in which the power relationship dynamics, the en-

tities that are part of them and the actions performed within them are represented in a figure. This representation contains the same information as the conditional representation.

Although the textual description is quite simple to be explained in few lines, both conditional and graphical representation would need to be defined in more details. In the conditional representation, conditional sentences are used to link the entities and the actions that they can perform in order to represent the conditions within a power relationship dynamic. The conditional sentence operators used in this study are as follows:

- **WHEN:** operator that initiates a sentence in the conditional representation. It should always precede entities whose action(s) would be responsible for triggering the power relationship dynamic;
- **IF:** conditional operator that indicates that a power relationship dynamic has come to a point of decision. Depending on how a group of entities responds to the previous action determines which happens next in this power relationship dynamic;
- **THEN:** operator that connects the condition expressed by the IF operator and this condition's consequences (actions performed by a group of entities). It can also serve as a connector between an two actions when the second action is a consequence of the first action and a WHEN introduced the first action;
- **OR:** operator that indicates that either one entity or another entity or both entities (but at least one of them) are involved in part of a power relationship dynamic. It can also indicate that more than one action might occur at the same time or generate a similar effect within the power relationship dynamic;
- **AND:** operator that indicates that more than one action must occur at the same time within a power relationship dynamic to generate a given effect. It could also be used similarly to the OR operator to relate entities. However, in this study, there was identified no power relationship dynamic in which more than one entity must be performing an action at the same time for an effect to take place.

Meanwhile, the graphical representation would also need to have defined entities representations and connectors. Moreover, there should be a defined model of how to use those entities and connectors in order to properly represent the power relationship dynamics. Since we now have information about relationships between power relationship forms, there would also be defined a model regarding those relationships. Therefore, the symbols that represent the entities previously identified in Chapter 4 and Chapter 5 can be seen in Figure 6.2, Figure 6.3, and Figure 6.4. Since we are investigating the npm OSSECO, for the entity called technological platform, npm logo is used as the entity symbol (Figure 6.5).



Figure 6.2: Entities within an OSSECO, part 1.

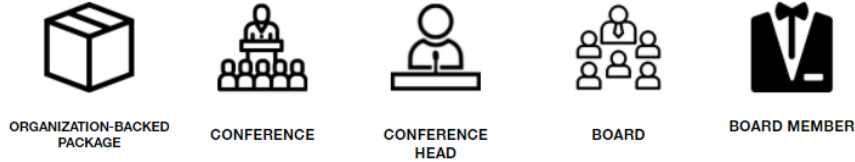


Figure 6.3: Entities within an OSSECO, part 2.

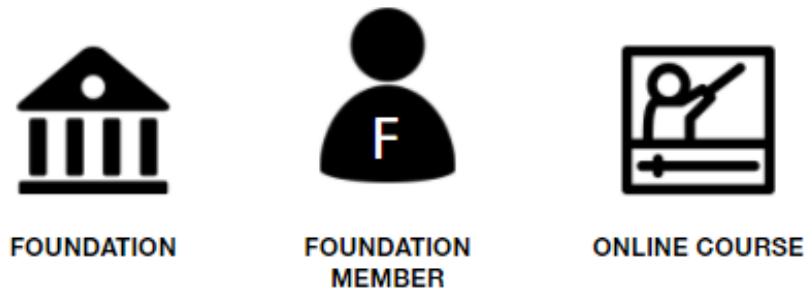


Figure 6.4: Entities within an OSSECO, part 3.

In addition, the connector symbols used in this study can be seen in Figure 6.6. The arrow connector represents the source entity group of an action and which entity group would suffer the effects of that action. An arrow can represent one or more actions that could occur at the same time within a power relationship dynamic and



Figure 6.5: The technological platform entity symbol for this study: the npm logo.

that could lead to a similar effect. Whenever an arrow represents more than one action, we use the conditional operators OR and AND to inform if the actions have to happen together or if only one of them is necessary. The dashed arrow symbolizes an action triggers or can possibly trigger another power relationship dynamic. The decision symbol represents action effects that can be different according to an entity group perspective. Depending on an entity group perspective, the power relationship dynamic could flow through different paths, and the power would produce different effects on the relationships.

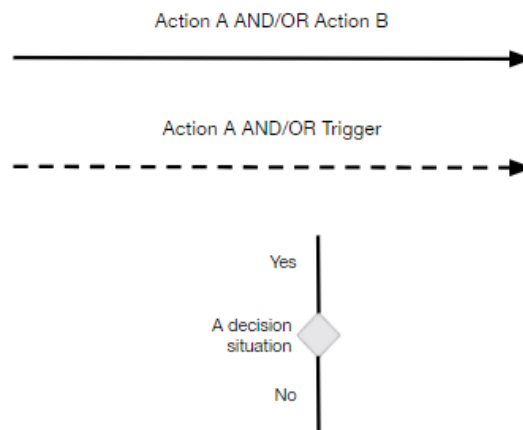


Figure 6.6: Connectors used in the graphical representation.

Since we presented the entities involved and the connectors used in the graphical representation, we are now able to discuss the power relationship dynamic representation. In this representation, we firstly need to define where the entities are grouped according to their power holding. The defined organization is that the group of entities in the left side of the graphical representations is the group that holds the power in the power relationship dynamic. The group of entities in the right side is the group that suffers the effects of the power relationship dynamic.

Those groups would have their relationship represented by the arrows (representing the actions) and the decision symbols (representing decisions that would be made depending on a group perspective). An example of a power relationship dynamic graphically represented and containing all of the possible symbols and their usage can be seen in Figure 6.7.

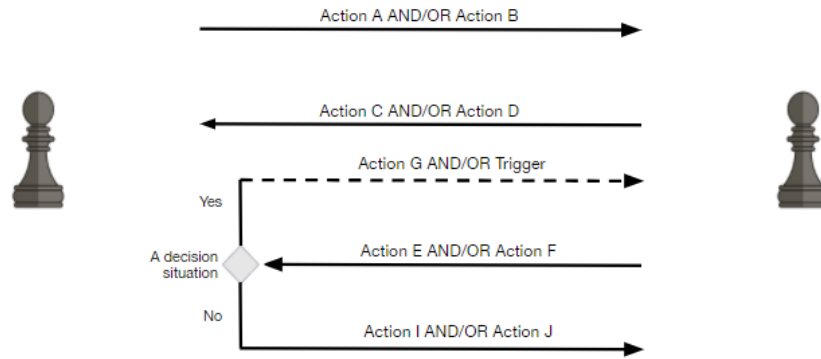


Figure 6.7: An example of a graphical representation of a power relationship dynamic.

Thereafter, we would have the power relationship dynamics ready to have their graphical representation presented. It is important to remark that each power relationship dynamic has its own id (Dxx , where xx is a number), its name, the entity group that holds the power, the entity group affected by the power and the actions performed within it. Those information are presented at the beginning of each power relationship dynamic description process.

The power relationship sources, in Valença et al. (2017) study, are the factors that contribute to an actor being a power holder. Those power relationship sources would be comprised within the power capability (PC), that would be the ability of a power holder to exercise his/her power. In the present research, we also consider the power relationship sources. However, instead of considering PC, we consider the power relationship dynamics. In this context, power sources would be prerequisites for a power relationship dynamic to occur in an OSSECO. Those prerequisites are also presented in the power relationship dynamics description phase and, thereafter, used to construct the model to represent relationships between power relationship forms.

Since we have defined power relationship dynamics, it is already possible to plan the model. In their work, Valença et al. (2017) combined the experiences and

perspectives from different small and medium enterprises members to extract information from and propose a SECO power relationships model (PRM-SECO). During the phase of constructing the model, the author decided to focus on the relationships between the different power relationship forms and how those relationships were affected by the power relationship sources.

In the present study, considering that interactions between entities are essential to the OSSECO and combining the new information gathered by the studies reported in Chapter 5 and the information gathered by the literature interview reported in Chapter 4, we propose a new version of the OSSECO-PREM. In this model, the entities are the core concepts and the main goal is to represent the interactions between them that are affected by power relationships. In order to simplify the model's visualization, we replaced the entities symbols previously used in the power relationship dynamics graphical representation by boxes with the entities' names on them. However, we maintained one of the connectors used in the power relationship dynamics graphical representation: the arrows.

Although we also use arrows in the model, it is essential to understand that there are slight changes in what those arrows represent. In the power relationship dynamics graphical representations, arrows would represent that a group of entities would perform an action that would affect another group of entities. The dashed arrows follow the arrows logic but represent triggers. Different from that representation, in the model arrows represent that an entity exercises power, through a power relationship dynamic identified by its ID, over another entity. Figure 6.8 is an example of a model in which Entity A exercises power through power relationship dynamic D_{xx} over the Entity B. In the same example, Entity C exercises power through power relationship dynamic D_{xy} over the Entity B and triggers D_{xz} power relationship dynamic.

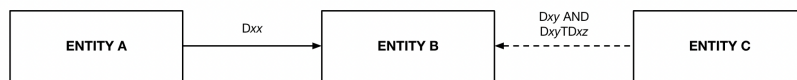


Figure 6.8: An example of an entity-based model.

In this study, we also consider the relationship between the power relationship

forms. Considering findings in the field study reported in Chapter 4, where 6 relationships between power relationship forms could be identified using the interviewees answers to the questions, it was also possible to develop an OSSECO power relationship forms model (OSSECO-PReFoM). OSSECO-PReFoM is inspired by PRM-SECO (VALENÇA et al., 2017) in its fine-grained view. In addition, a power relationship form would have a power source, which is a factor that represents from where a power comes. Therefore, we have to define the connectors to be used in the model that represents these OSSECO power relationship forms interactions. A power relationship form would be capable of either maximize or minimize another power relationship form's effect. Therefore, we have two connectors that can be seen in Figure 6.9. The arrow with a plus sign above means that the source power relationship form maximizes the destination power relationship form's effect. In contrast, the arrow with a minus sign above means that the source power relationship form minimizes the destination power relationship form's effect. The power sources related to each power relationship form should also be represented in this model. Figure 6.10 shows an example of the idea for the model.

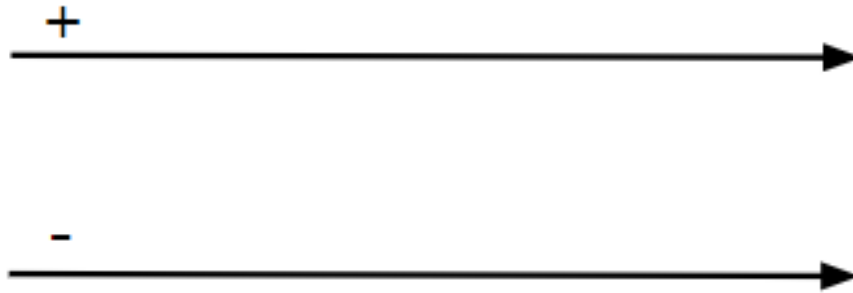


Figure 6.9: The connectors for the power relationship forms model.

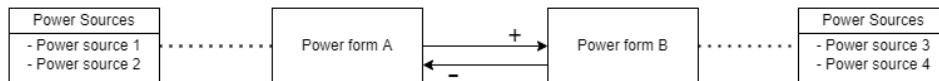


Figure 6.10: The example of how the power relationship forms model should look like.

Therefore, we defined how the power relationship dynamics have to be defined and what symbols, operators and connectors should be used in the different representations, and how we intend to combine those power relationship dynamics to

construct a model. It is now possible to proceed to the power relationship dynamics description phase and, thereafter, the OSSECO-PREM construct phase and a new model that represents the relationships between the power relationship forms, the OSSECO-PReFoM.

6.3 Describing the Power Relationship Dynamics

Since we have defined how the power relationship dynamics should be described and which descriptions should be used to represent each one of them, we can proceed with the descriptions phase. However, we did not described one of the components that are part of a power relationship dynamic: the actions. Actions are essential to the power relationship dynamics as we are describing them since they represent the interactions in which power is exercised within a dynamic. The studies reported in Chapter 5 identified some of those actions. From those the already identified actions and the literature, a set of 34 actions was identified. Based on the information gathered by the studies reported in Chapter 5, it was also possible to identify 14 entities that are involved with power relationship dynamics within an OSSECO. We were also able to identify 6 power sources related to the power relationships within an OSSECO. The sources from where those entities, actions and power sources came from are also presented. Those sources can be selected studies from the systematic mapping study reported in Chapter 4 (*Sxx*, as seen in Table 4.3), respondents from the exploratory study reported in Chapter 5 (*Rxx*, as seen in Table 5.2) or interviewees from the field study also reported in Chapter 5 (*Ixx*, as seen in Table 5.6).

6.3.1 Entities Involved with Power Relationships Within an OSSECO

Considering the information gathered in Chapter 4 and Chapter 5, it was possible to identify the entities involved with the power relationship dynamics within an OSSECO. It is important to remark that entities, in this context, are not only actors, but also artifacts and organizations that somehow are part of the npm OSSECO. Therefore we describe the entities as follows:

Board: it is an entity that represents an organization, that usually addresses technical subjects related to an OSSECO. This entity was reported by I06;

Board member: it is an entity that represents an board member, which is a

person that has the power to make decisions regarding the subjects addressed by the board. This entity was reported by I06;

Collaborator: it is the entity that contributes to projects within the OSSECO. This entity opens pull requests, open issues, and comments on both pull requests and issues. It can also be responsible for bringing new requirements into the projects or identifying bugs that need fixing. This entity was reported by R01;

Community-backed package: is an entity that represents a package that is maintained by the OSSECO community. Its roadmap is defined according to the package scope and the OSSECO community needs. This entity was previously comprised in the Package entity. However, since there are packages maintained by the OSSECO community and others maintained by organizations, two new entities were defined. This entity was reported by R05, R10 and R13 as “package”. However, it was renamed because of the identification of a package that would be guided by an organization roadmap, the *organization-backed package*;

Conference: it is an entity that represents an event in which OSSECO community members gather to discuss OSSECO related subjects. This entity was reported by I10;

Conference head: it is an entity that represents a conference member that has to make decisions regarding the conference program and its content. This entity was reported by I10;

Foundation: it is an entity that represents a non-profit organization that help to organize the OSSECO. This entity was reported by S09 and I10, but was only identified combining both reports;

Foundation member: it is an entity that represents a member of a foundation, who is a person that can make decision regarding the OSSECO organization. This entity was reported by S09 and I10, but was only identified combining both reports;

Maintainer: it is the entity that has committer privileges within projects that are part of the OSSECO. They are core developers within the OSSECO, and they help make decisions that could potentially affect a lot of other entities and projects within the OSSECO. This entity was reported by R09.

Online Course: it is an entity that represents an online learning environment. It can not only share knowledge with others but also influence decisions. This entity

was reported by I08, I10 and I11;

Organization: it is an entity that represents a company that takes part in the OSSECO. This entity is usually responsible for making decisions regarding the **technological platform** (in the case of owning it), the projects it might own, and for sponsoring projects and hiring developers. This entity was reported by R05;

Organization-backed package: is an entity that represents a package that is maintained by an organization. Its roadmap is defined according to the organization's business goals. This entity was reported by I02, I04, I05, I06, I08, I09, I13;

Technological platform: it is the central technology around which the OSSECO exists. In some cases, an **organization** owns this entity and decides the technological platform development and evolution. This entity was reported by R05;

User: it is the entity that has projects that are part of the OSSECO. It usually works with projects that have a project that is part of the OSSECO as a dependency. It can also open issues or comment on other issues. Sometimes, it is possible that this entity can request a new feature or fix. This entity was reported by R09.

6.3.2 Actions that Could Be Performed by the Entities

Although to identify those actions helps to understand the power relationship dynamics, it is essential that those actions are properly described. This description provides information that can be used to assist in the power relationship dynamics description and understanding. Table 6.1 contains information about the identified actions and we describe them as follows:

Accept a pull request: whenever a collaborator or an user who wants to become a collaborator want to have a code change authored by them merged into the package main code, they need to open a pull request. This pull request is then evaluated by the package maintainers, who essentially evaluate the code quality, if the code follows defined best practices and if the changes match the package goals. If the code is acceptable according to these topics, the pull request is accepted. This actions was reported by S03, selected in Chapter 4;

Address an issue: however, in some cases, collaborators or users do not have

the time or the technical skills required to propose a code change but still need that change. When this happens, it is common that those collaborators or users open an issue, which would be a public message in a forum-like format informing about the necessary change. This issue can also be reporting a problem to the package maintainers. Therefore, package maintainers should either proceed with the code changes, delegate this change to another maintainer or contributor or explain why this change would not be interesting to the package and reject it. This action was reported by S01 and S02;

Ask for technical help: whenever a package maintainer do not have the technical skills for making a necessary or desired code change, it is possible that he/she asks for a collaborator's help. Since there are collaborators known by their technical skills, OSSECO community members consider those collaborators to be the ones that are able to help whenever a technical block appears. Therefore, when a package maintainer needs help to achieve a goal that himself/herself is not technically capable of, it requires those collaborators' help. This action was reported by S07;

Ask for technical knowledge: however, in some cases the goal of a collaborator, maintainer or user is not to achieve a specific goal, but to learn a technical skill. Therefore, those collaborators who have known technical skills that could help in this learning process could be required to help. In this case, it is possible that the help is related to a specific issue or development goal. However, the main goal for those who require this help is to learn how a new technical skill or improve an already existing one. This action was reported by R14 as "Learn", but was renamed;

Block user: whenever a collaborator or an user needs a code change or an issue to be solved, they will interact with the package maintainers, either opening an issue or a pull request. Since it is possible that: (i) package maintainers are not able to immediately respond to those requests; or (ii) those requests might not be adherent to the package's goals; it is possible that this interaction cause conflicts. Whenever those conflicts emerge, package maintainers and the requester discuss whether this change should or should not be addressed. In some cases, those discussions can cause more conflicts that might result in the package maintainers blocking the requester. This action was reported by S01 and S02;

Change package dependency: whenever a dependency package changes their

code and this change is a breaking change that requires code changes from the dependant package, there might occur a dependency replacement. This change occurs in order to avoid the need of changing the own code or to avoid possible unwanted features, security breaches and the adding of more dependencies. This actions was reported by S08. Whenever a stakeholder is not satisfied by its dependency anymore, it could proceed with the change of this dependency for another which satisfies he/she/it the most;

Contribute for the first time: occasionally, users can try to become collaborators by opening pull requests to have their desired code changes merged into the package's main code. This action can happen since previous issues would not be addressed by package maintainers because they would not have time to do it. This action was reported by S01 and S02 since one of the processes of role migration within an OSSECO would be directly related to the first contribution of a newcomer, that would become a collaborator;

Define program: it is an action that represents a decision regarding the content of a conference program. This actions was reported by S10 and it was possible to identify after the mentioning of *conference* entity;

Fork the project: it is an action that represents the act of cloning a repository. This actions was reported by I04 and I10;

Give higher privileges: during their contributions, developers might seek to play roles in which their opinions could influence the decision-making process. Therefore, those who have the power to promote those developers (maintainers) would occasionally grant some of those developers a higher set of permissions and influence within a package. That could happen if the contribution is considered to have a high value to the package and it is part of a regular contribution. This action was reported by S01 and S02 since, considering that a collaborator contributes regularly and with a high quality to the OSSECO, this collaborator would be a candidate to have more permissions within the OSSECO;

Give money: it is the simple act of rewarding someone with a financial value. It is usually performed by organizations. This action was reported by R05;

Have its (package) code changed: whenever a package has a pull request accepted and merged by its maintainers or even one of the maintainers make a code

change by himself/herself, we have this action. Since the package code changed, it is possible that this change affects the packages that dependent on it. This action was reported by S08;

Hire: directly related to the employment relation, this action is the classical hiring action. This action is exclusively performed by organizations and it usually affects collaborators and maintainers. This action was reported by S10;

Ignore the request: whenever a maintainer receives a request (that can be an opened issue or opened pull request), he/she has to decide if this request should or should not be addressed. Since some requests might not be a priority for the package or might not even be adherent with the package's goals, it is possible that those are ignored. This action could generate conflicts between maintainers and those who requested the changes. This action was reported by S04;

Influence a decision: occasionally, entities might have some influence over other entities decisions. This is usually related to the entities that somehow have a reputation that makes other entities consider them as references. Therefore, other entities should consider their opinion regarding the decision-making process. This action was reported by S07;

Influence collaboration: similarly to the previous action, this action is also performed by entities that considers that other entities' opinions should be considered. However, in this action, the entities consider their opinion regarding the development process. This action was reported by S07;

Invite to contribute: it is an action that represents an invitation made for collaborators to contribute with an OSSECO project. This action is an extension of *influence collaboration* action. It was created just to differentiate the act of influencing an OSSECO community member to contribute from the act of influencing an outsider to contribute;

Keep package dependencies: whenever a package has its code changed, packages that depend on it have to decide either if they are replacing the dependency or keeping it. This action refers to the act of keeping the dependency. This action was reported by S08;

Make a comment: it is the simple act of commenting on an issue or in a pull request. This action was reported by S07;

Table 6.1: The actions that can be performed within power relationship dynamics.

ID	Action name	Entities who can perform it
A01	Accept a pull request	Maintainer
A02	Address an issue	Maintainer
A03	Ask for technical help	Maintainer
A04	Ask for technical knowledge	Collaborator, Maintainer, User
A05	Block user	Maintainer
A06	Change package dependency	Collaborator, Organization, Package, User
A07	Contribute for the first time	User
A08	Define program	Conference head
A09	Fork the project	Collaborator, User
A10	Give higher privileges	Maintainer
A11	Give money	Organization
A12	Have its code changed	Package
A13	Hire	Organization
A14	Ignore the request	Collaborator
A15	Influence a decision	Collaborator, Maintainer
A16	Influence collaboration	Collaborator, Maintainer, Organization
A17	Invite to contribute	
A18	Keep package dependencies	Collaborator, Organization, Package, User
A19	Make a comment	Collaborator, Maintainer, Organization, User
A20	Make a decision	Organization
A21	Make code contribution	Collaborator, Maintainer
A22	Open a pull request	Collaborator, Maintainer, User
A23	Open an issue	Collaborator, Maintainer, Organization, User
A24	Provide technical help	Collaborator
A25	Reject a pull request	Maintainer
A26	Remove higher privileges	Maintainer
A27	Remove ownership	Maintainer, Technological platform
A28	Request a code change	
A29	Respond to a comment	Maintainer
A30	Share knowledge	Collaborator, Maintainer
A31	Show technical skills	Collaborator, Maintainer
A32	Stop to give money	Organization
A33	Terminate employment	Organization
A34	Transform user into collaborator	Maintainer

Make a decision: it is the simple act of, given a situation, make a decision in order to address the pros and cons of this situation properly. This action was reported by R05;

Make code contribution: it is the simple act of making a code contribution to a package code. This action was reported by R01;

Open a pull request: whenever a collaborator or an user that wants to collaborate want to have their code changes merged into the main code, they need to open a pull request. This action refers to the act of opening a pull request that would signalize to the maintainers the intention of a code change. This action was reported by S01 and S02;

Open an issue: occasionally, an user or a collaborator might need some code change or issue fixing, but are not able to technically perform those changes or fixes. In those cases, they open issues that alert the maintainers that some code changes or bug fixes were requested. This action was reported by R06;

Provide technical help: in some cases, maintainers might request help to a collaborator who has specific technical skills. Therefore, this collaborator would evaluate if he/she is interested in making this contribution. If the answer is “Yes” he/she provides the technical help maintainer was asking for. This action was reported by S07;

Reject a pull request: whenever a collaborator or an user who wants to become a collaborator want to have a code change authored by them merged into the package main code, they need to open a pull request. This pull request is then evaluated by the package maintainers, who essentially evaluate the code quality, if the code follows defined best practices and if the changes match the package goals. If the code is not acceptable according to these topics, the pull request is rejected. This action was reported by S01, S02 and S07;

Remove higher privileges: during their contributions, developers might seek to play roles in which their opinions could influence the decision-making process. Therefore, those who have the power to demote those developers (maintainers) would occasionally remove some of those developers set of permissions and influence within a package. That could happen if the contribution is considered anyway harmful. This action was reported by S01 and S02;

Remove ownership: it is an action that represents the decision of removing another entity’s ownership of an OSSECO project. This action emerged based on I03 report of ownership being removed within the npm OSSECO;

Request a code change: it is an action that represents that an entity desires to have a code changed within an OSSECO project. This action was reported by S08;

Respond to a comment: it is the simple act of responding to a previous comment on an issue or in a pull request. This action was reported by S07;

Share knowledge: occasionally, collaborators, maintainers or users might request developers who have essential and specific technical skills to share their knowledge with them. This action refers to the act of those developers sharing their skills in order to collaborators, maintainers and user to learn from them. This action was reported by R14;

Show technical skills: it is the act of, through regular contributions, proving their technical skills. It can be performed by collaborators across different packages within the OSSECO. This action was reported by S07 since a developer could use his/her technical skills to influence others within an OSSECO;

Stop to give money: it is the simple act of stopping to give money to another entity since their actions are not conforming with the organization's desires. As stated, this action is exclusively performed by organizations. This action emerges from the combination of information reported by S10 and R05. S10 states that an organization participates in an OSSECO to achieve its business goals. R05 states that, whenever an organization identifies a package within the OSSECO to be essential for its business goals to be achieved, it sponsors this package to reward its development and guarantee its maintenance. Therefore, in case this package is no longer considered to be essential for an organization to achieve its business goals, it would stop sponsoring it. In other words, it would stop giving money to the package;

Terminate employment: similarly to the previous action, but related to the employment relation. Whenever an organization considers that a collaborator or maintainer is not performing as desired, the organization could terminate the employment relation. This action was reported by S10;

Transform user into collaborator: it is an action that represents that an entity, that has powers to alter other entity's role, uses this power to proceed with this change. This action was reported by S01 and S02.

6.3.3 Power sources

Considering the results from the studies reported in Chapter 4 and Chapter 5, it was possible to identify and describe the power sources for the power relationship dynamics. Power sources would be a factor that an entity uses to affect another entity's behavior (DAHL, 1957). Therefore, those power sources would be essential to understand where does the exercised power came from in a power relationship. We were able to identify 8 power sources as follows:

Breaking change: it is a power source related to the insertion of code in a package that require that packages that depend on it update their code. This power source would enable a package to insert significant code change into another package's code. Therefore, this code change would make the dependant package behave as the first package desires. It was reported by I03, I06, and I07;

Financial reward: it is a power source related to payment in exchange of a contribution. Therefore, this financial motivation would make the entity that is being paid to behave as the paying entity desires. It was reported by S10;

Knowledge: it is a power source related to the knowledge that an entity has and that it could use to make another entity behaves as it desires. For example, in the case of an entity that wants to learn something, this entity would behave as another entity that has the desired knowledge wants in exchange for the knowledge itself. It was reported by S10;

Ownership: it is related to the legal right that an entity has over something (e.g., a package). Since this entity has the legal right, it could make other entities behave as it wants. It was reported by I03;

Partnership opportunity: it is related to an entity, because of its desire to have business partner, that behaves as another entity desires. Since partnerships can help entities to achieve their goals within the OSSECO, those entities would behave differently to have those partnerships. It was reported by S08;

Reputation: it is related to the status an entity have within an OSSECO. Because of this status, this entity is considered as having a high value for the OSSECO. This high value can be related to either social or technical aspects. Thus, this entity would have other entities behaving as it wants in order to be maintained satisfied. It was reported by I02, I04, I05, I06, I07, I08, I09, I12, and I13;

Role: it is related to the role that an entity plays in the OSSECO. Different roles have different permissions within the OSSECO. Therefore, what a maintainer and a collaborator can do within the OSSECO is different. This difference of permissions would make entities behave as others want just to achieve higher roles with higher privileges within the OSSECO. It was reported by S01 and S02;

Technical skills: it is related to the technical skills that an entity It was reported by S07.

6.3.4 The Power Relationship Dynamics

Since we have already defined all the entities, symbols, operators and actions that we might use during the power relationship dynamics description, it is time to present the descriptions. We present each power relationship dynamic separately and organize the descriptions in the following order: (i) textual description; (ii) conditional description; and (iii) graphical description. In addition, we indicate if the power relationship dynamic came from the systematic mapping study reported in Chapter 4, from the exploratory study reported in Chapter 5, or from the field study also reported in Chapter 5. We also indicate from which selected studies (*Sxx*, as seen in Table 4.3), respondents (*Rxx*, as seen in Table 5.2) or interviewees (*Ixx*, as seen in Table 5.6) they were extracted.

Conforming to stakeholders' requirements (D01). This power relationship dynamic occurs whenever an *entity*, that can be a *collaborator*, an *organization* or an *user*, that use a given package, makes a request to this package's *maintainer*. This request can be sent through opening an issue or making a comment in an already opened issue or pull request.

In some cases, *maintainers* consider it to be necessary to keep those stakeholders satisfied since their usage can help the package to achieve their goals (either business or development goals). It is also possible that *maintainers* consider the requirements adherent to the package's goal and, therefore, are interesting to the package. However, it is also possible that *maintainers* do not consider the requirements to be part of the package's scope.

Therefore, the *maintainers* can either respond to a comment or to address the

request. Whenever this happens, the requesting *entity* evaluates the response or the proposal to address the request. In case this response is satisfactory to this *entity*, which means that this *entity* either has its needs fulfilled or understands and agrees with the reasons given for the changes being rejected, this *entity* keeps the package dependency. Otherwise, in case this response is not satisfactory to this *entity*, which means that this *entity* did not have its needs fulfilled and did not agree with the reasons given by the *maintainers* to reject the changes.

Whenever this power relationship dynamic ends with the *entity's* decision to change the dependency, this coercive power relationship dynamic directly affects the *maintainer*. Therefore, the *maintainer* possibly loses their stakeholders who could help in the package development and achievement of business goals. Since in the npm OSSECO there usually many packages that provide the same features, the *entity* that dropped the package is probably replacing it with no trouble. In case the *entity's* decision is to keep the dependency the power relationship dynamic does not affect negatively affect either *entity* or *maintainer* until the next interaction between them. It is possible that this power relationship dynamic triggers **dependency between projects power relationship dynamic (D03)**.

The power source for this power relationship dynamic is **partnership opportunity**. This power relationship dynamic was also reported by S08. Since in this power relationship dynamic, entities would be pressured to follow a private roadmap to avoid the punishment of being replaced as dependencies, we can relate this dynamic to the *addressing private roadmap* challenge. Figure 6.11 contains this power relationship dynamic graphical representation and its conditional representations is presented as follows:

WHEN Collaborator OR Organization OR User *Open an issue* OR *Make a comment* to **Community-backed package** OR **Organization-backed package**, THEN **Community-backed package** OR **Organization-backed package** *Request a code change* to **Maintainer**, THEN **Maintainer** *Make code contribution*, THEN **Community-backed package** OR **Organization-backed package** *Address an issue* OR *Respond to a comment*, THEN Collaborator OR Organization OR User *Trigger D03*

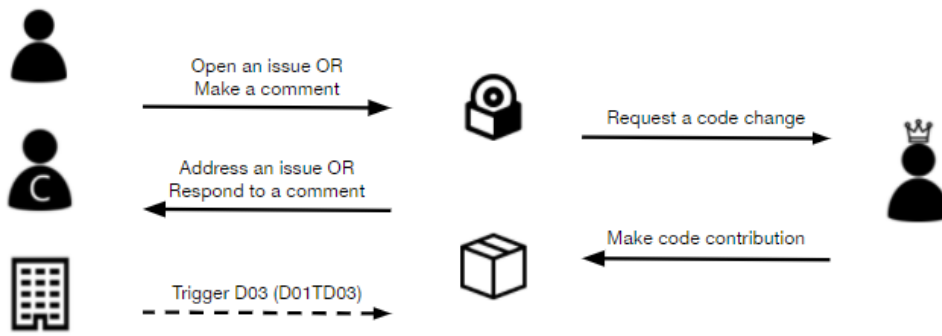


Figure 6.11: Conforming to stakeholders' requirements refined.

Control over technological platform (D02). In this power relationship dynamic, an *organization* has the power of making decisions regarding a *technological platform*. Although OSSECO community members usually made decisions regarding the *technological platform* development, this is not what happens in all cases.

In some cases, *organizations* assist in the OSSECO management and decision-making process. Those *organizations* can adopt different governance methods to help to organize the OSSECO. Since OSSECO community members consider it essential to take part in the decision-making process, many *organizations* adopt democratic approaches that consider the OSSECO community's opinion during the decision-making process. However, in some cases, there are *organizations* that prefer to adopt an autocratic approach in which the OSSECO community members in which members' opinion is only considered in situations that are not considered essential do the *technological platform* development. Although it is common that one of those approaches is adopted, methods can also be mixed depending on the situations. As an example, in case of situations that could represent a legal risk to the *technological platform*, an *organization* that usually adopts a democratic approach can use an autocratic approach to make a decision to minimize that legal risk and avoid legal penalties. In this case, this power relationship dynamic might trigger **disowning project power relationship dynamic (D15)**.

The power source for this power relationship dynamic is **ownership**. This power relationship dynamic was reported by S05 and R05. Since this power relationship dynamic is related to the technological platform being owned and organized by an organization, we can relate this dynamic to the benefit *organizing the OSSECO*. Figure 6.12 shows this power relationship dynamic graphical representation and its

conditional representations is presented as follows:

WHEN **Organization** *Make a decision*, **Technological platform** *Have its code changed*, THEN **Technological platform** *Trigger D15*.



Figure 6.12: Control over technological platform refined.

Dependency between projects (D03). This power relationship dynamic occurs whenever an *entity A*, that can be either a *community-backed package* or an *organization-backed package*, that is a dependency to an *entity B*, that can be either a *community-backed package* or an *organization-backed package*, has its code changed. Since *packages* are dependant to each other within the npm OSSECO, it is common that *packages* have lots of dependencies. Those dependencies could be used to address trivial tasks or specific features related to tasks that are not related to the *package's* core business.

Therefore, *entity B* and its maintainers need to be aware of changes in their dependencies since they should evaluate whether those changes negatively affect the *package* or not. Moreover, it is possible that changes in *entity A* require *entity B* to also change its code to keep using *entity A's* latest version.

Whenever *entity B* decides that the changes do not negatively affect it or considers own code changes to be reasonable in order to keep using *entity A's* latest version, *entity B* keeps the dependency. This act of keeping the dependency does not affect the OSSECO. Otherwise, when *entity B* decides that the changes could negatively affect it or considers own code changes to be unnecessary or undesired, *entity B* changes the dependency and drops *entity A*. This act of changing the dependency affects the OSSECO since *entity B* replaces *entity A* and, therefore, *entity A* can suffer a reputation loss and possible collaborators to help in its development.

It is important to remark that this power relationship dynamic was not refined. The power source for this power relationship dynamic is **breaking change**. This power relationship dynamic was reported by R10, R13, I01, I02, I03, I07, I11, I12, and I13. Since this power relationship dynamic does not seem to match any of the benefits and challenges identified in Chapter 4, we consider this power relationship dynamic to be neutral. Figure 6.13 shows this power relationship dynamic and its conditional representation is presented as follows:

WHEN Community-backed package A OR Organization-backed package A *Have its code changed*, IF the change needed was satisfactory to Community-backed package B OR Organization-backed package B, THEN Community-backed package B OR Organization-backed package B *Keep package dependencies*. IF the change needed was not satisfactory to Community-backed package B OR Organization-backed package B, THEN Community-backed package B OR Organization-backed package B *Change package dependency*.

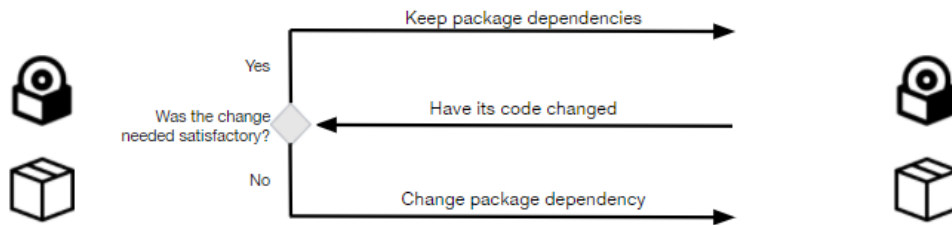


Figure 6.13: Dependency between projects refined.

Employment relation (D04). In this power relationship dynamic, a classical employment relation occurs. Whenever an *organization* hires an *entity*, that can be either a *collaborator* or a *maintainer*, this *organization* expects that those new hired employees are able to perform tasks in order to help this *organization* to achieve its business goals. Since *organizations* are part of the OSSECO community in order to achieve business goals, they consider that *maintainers* and *collaborators* that already contribute with the OSSECO could help them to achieve those business goals.

Therefore, the hired *entity* contributes to packages that it already contributed before or to packages that it had never contributed before. These contributions are

opened issues, comments made in issues or pull requests, and opened pull requests. Since this *entity* has a promised financial reward to contribute, it concentrates its efforts into collaborating to the packages the hiring *organization* desires.

Whenever the hired *entity* makes its contribution, the *organization* evaluates this contribution to determine if the contribution is satisfactory or not. This evaluation might differ from an *organization* to another and, therefore, it is a very specific process for each *organization*. However, the results of this evaluation affect the employment relation. In case the *organization* considers the *entity's* contribution to be unsatisfactory, it can terminate the employment to both punish the *entity* and to be able to employ another *entity*. A contribution is considered unsatisfactory either if it did not achieve defined quality levels, if it did not meet defined deadlines or if it did not help in achieving *organization's* business goals. Otherwise, in case the *organization* consider the *entity's* contribution to be satisfactory, the *organization* proceed with the hired *entity* payment and the employment relation is maintained.

It is important to remark that this power relationship dynamic was not refined. The power source for this power relationship dynamic is **financial reward**. This power relationship dynamic was reported by S10, R06, R08, R10, I04, I05, I07, I08, I09, I11, and I13. Since the entities are employed to satisfy an organization roadmap, we can relate this power relationship dynamic to the *addressing private roadmap* challenge. Figure 6.14 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Organization** *Hire* **Collaborator** OR **Maintainer**, **Collaborator** OR **Maintainer** *Make a comment* OR *Open an issue* OR *Open a pull request*. IF the contribution was satisfactory to **Organization**, THEN **Organization** *Give money* to **Collaborator** OR **Maintainer**. IF the contribution was not satisfactory to **Organization**, THEN **Organization** *Terminate employment*.

Learning reference (D05). In this power relationship dynamic, an *entity A*, that can be either a *collaborator* or a *maintainer*, shows its technical skills and achieves a high reputation regarding those skills. Therefore, another *entity B*, that can be

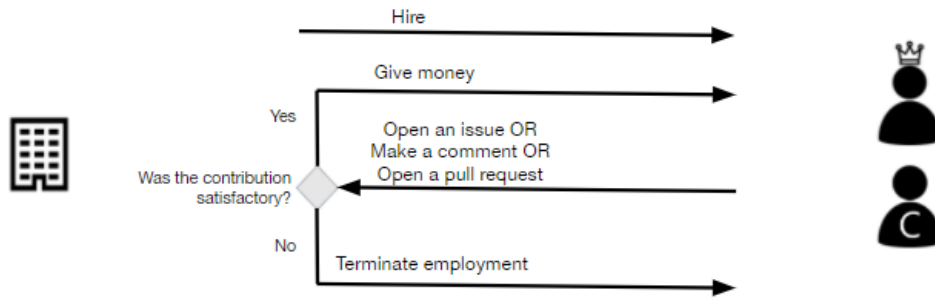


Figure 6.14: Employment relation refined.

either a *collaborator*, a *maintainer* or an *user*, consider *entity A* to be a technical reference.

Since the motivations for collaborating within an OSSECO have shifted and learning is a common motivation nowadays, *entity B* would be interested in learning new technical skills or improving existing ones. Therefore, *entity B* would request *entity A* to share its technical knowledge so that *entity A* can learn from it.

Whenever *entity A* considers this request to be interesting to it, it can be because of its finality or by being a subject that interests *entity A*, *entity A* responds to this request. In this response, it is possible that *entity A* not only share its knowledge to help *entity B* learning a new skill or improving an existing one, but also influences *entity B* decisions. This influence can occur either by sharing best practices and coding styles or by giving advice regarding a different approach to solve a given problem. Otherwise, if *entity A* considers the request not to be interesting, it simply ignores the request and *entity B* does not learn new skills or improve existing ones.

The effects to the OSSECO depend on the *entity A* response to *entity B* request. In case *entity A* shares its knowledge, not only *entity B* become more skilled and able to help in the OSSECO development but also *entity A* achieves a higher reputation within the OSSECO. This reputation enhancement could lead to more entities requesting *entity A* help and even newcomers joining the OSSECO as they would know that *entity A* could help them learn new skills and, therefore, help to develop within the OSSECO.

It is important to remark that this power relationship dynamic was not refined. The power source for this power relationship dynamic is **knowledge**. This power relationship dynamic was reported by S10. Since this power relationship dynamic

reports a situation in which entities want to learn to be able to collaborate, it can be related to *bringing innovation to the OSSECO* and *stable collaboration with the OSSECO* benefits. Figure 6.15 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN Collaborator OR Conference OR Maintainer OR Online course *Show technical skills*, Collaborator OR Maintainer OR User *Ask for technical knowledge*. IF the request was interesting to Collaborator OR Maintainer, THEN Collaborator OR Maintainer *Share knowledge*. IF the request was not interesting to Collaborator OR Maintainer, THEN Collaborator OR Maintainer *Ignore the request*.

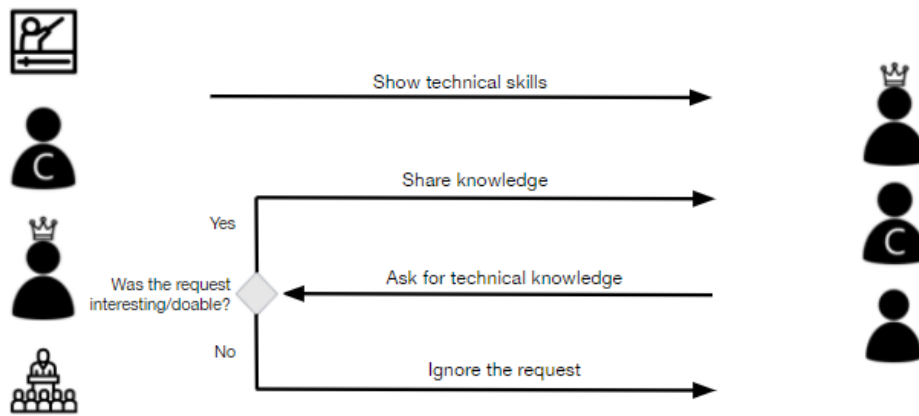


Figure 6.15: Learning reference refined.

Newcomers contribution rejection (D06). In this power relationship dynamic, an *user* that desires to become a *collaborator* opens a pull request to contribute with code for the first time. Thereafter, a *maintainer* evaluates *user's* contribution to decide if the contribution is either valid or not.

This evaluation might differ from case to case depending on the package this contribution was sent to and the *maintainer* that is executing this evaluation. *Maintainer* considers during the evaluation if the contribution is adherent to the package road-map and if the code meets best practices and quality demands.

Whenever *maintainer* evaluates this contribution as valid, the pull request related to it is accepted and merged into the main code. In this case, an *user* becomes

a *collaborator*, which can bring new perspectives, goals and innovation into the OSSECO. Otherwise, if the *maintainer* evaluates this contribution as not valid, the pull request related to it is rejected. In this case, the *user* can either try to adjust his/her contribution and submit it for evaluation again or drop this contribution, which could lead to an *user* not migrating to a collaborator role and to blocking innovation income.

The power source for this power relationship dynamic is **role**. This power relationship dynamic was reported by S01 and S02. Since this power relationship dynamic presents a situation in which newcomers have their contribution blocked and those newcomers are responsible for bringing innovation to the OSSECO, we can relate this dynamic to the *blocking innovation* challenge. Figure 6.16 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **User** *Contribute for the first time*, IF the contribution was valid to **Maintainer**, THEN **Maintainer** *Accept pull request AND Transform user into collaborator*. IF the contribution was not valid to **Maintainer**, THEN **Maintainer** *Reject pull request*.

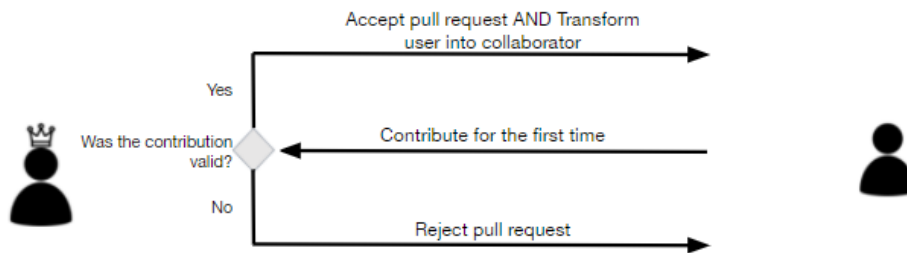


Figure 6.16: Newcomers contribution rejection refined.

Promoting collaboration (D07). In this power relationship dynamic, an *entity*, that can be either a *collaborator*, a *maintainer* or an *organization*, uses its reputation to influence a *collaborator's* collaboration within the OSSECO. This situation usually happens when this *entity* has a road-map that requires a contribution that it might not be able to make at that point. Therefore, it uses its reputation to encourage a *collaborator* to work on that desired contribution and help it to achieve its

road-map goals. In this case, this power relationship dynamic triggers **Dependency between projects power relationship dynamic (D03)**.

Although this power relationship dynamic usually works in an active influence from *entity* to the *collaborator*, it can also happen in a passive form. Whenever a *collaborator* knows that an *entity* is interested in a given package, this *collaborator* could make contributions to this package in order to draw this *entity* attention.

Moreover, this power relationship dynamic could affect the OSSECO regarding the packages which have higher interest from the community perspective. Therefore, it could be related to the *stable collaboration with the OSSECO* benefit. The power source for this power relationship dynamic is **reputation**. This power relationship dynamic was reported by R05. Figure 6.17 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN Collaborator OR Conference OR Maintainer OR Organization *Influence collaboration* AND *Trigger D03*, Collaborator *Make code contribution*.

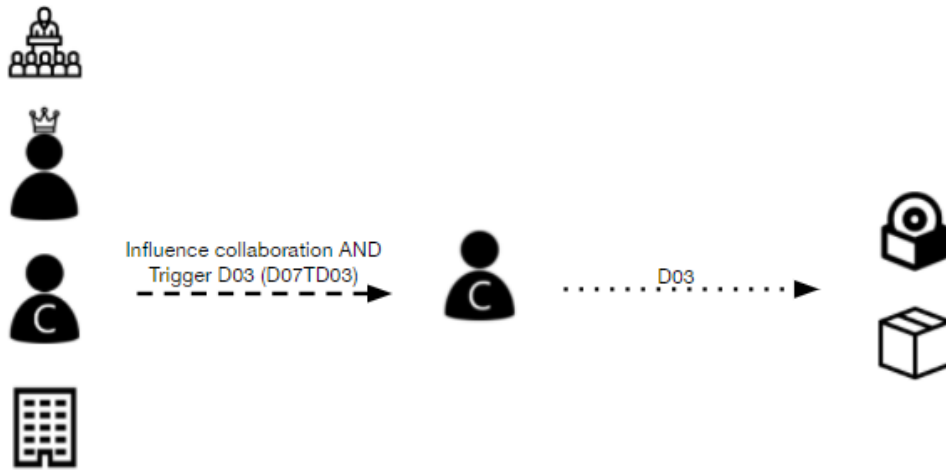


Figure 6.17: Promoting collaboration refined.

Pull request evaluation (D08). In this power relationship dynamic, a *collaborator* who intends to propose a code change to a package's main code opens a pull request. In this case, this power relationship dynamic triggers **role migration within the OSSECO power relationship dynamic (D09)** and **blocking power relationship dynamic (D12)**. Thereafter, a *maintainer* evaluates *collaborator's* contribution to decide if the contribution is either valid or not.

This evaluation might differ from case to case depending on the package this contribution was sent to and the *maintainer* that is executing this evaluation. *Maintainer* considers during the evaluation if the contribution is adherent to the package road-map and if the code meets best practices and quality demands.

Whenever the *maintainer* evaluates the contribution and consider it valid, the pull request is accepted and merged to the main code. Otherwise, if the *maintainer* evaluates the contributions and consider it not valid, the pull request is rejected. In this case, this power relationship dynamic triggers **project forking power relationship dynamic (D13)**.

The power source for this power relationship dynamic is **role**. This power relationship dynamic was reported by R01, R02, R03, R06 and R10. Since this power relationship dynamic is part of a routine within an OSSECO and does not blocks innovation or supports any collaboration, we could not relate it to any challenge or benefit. Therefore, this power relationship dynamic is neutral. Figure 6.18 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Collaborator** *Open a pull request* AND *Trigger D09* AND *Trigger D12*, IF the contribution was valid to **Maintainer**, THEN **Maintainer** *Accept pull request*. IF the contribution was not valid to **Maintainer**, THEN **Maintainer** *Reject pull request* AND *Trigger D13*.

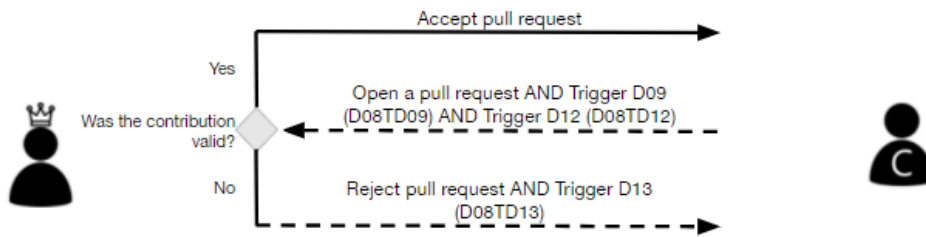


Figure 6.18: Pull request evaluation refined.

Role migration within the OSSECO (D09). In this power relationship dynamic, an *entity*, that can be either a *collaborator* or an *user*, make a contribution to a package. Thereafter, a *maintainer* evaluates *entity's* contribution to decide if the contribution is either valid or not.

This evaluation might differ from case to case depending on the package this contribution was sent to and the *maintainer* that is executing this evaluation. *Maintainer* considers during the evaluation if the contribution is adherent to the package road-map and if the code meets best practices and quality demands.

Whenever *maintainer* considers the contribution to be valid, the *maintainer* also evaluates the contribution regarding how was its impact on the package. If the impact on the package is not remarkable, *maintainer* only accept the pull request or respond to the comment. If the impact on the package is remarkable, *maintainer* accept the pull request or respond to the comment and grant *entity* new permissions and, therefore, a higher privilege.

However if *maintainer* considers the contribution not to be valid, the *maintainer* also evaluates the contribution regarding contribution's harm potential. If the *maintainer* considers the contribution not to be harmful, *maintainer* only rejects the pull request or ignores the request. In this case, this power relationship dynamic triggers **project forking power relationship dynamic (D13)**. If the *maintainer* considers the contribution to be harmful, *maintainer* rejects the pull request or ignores the request and demote the *entity*. Therefore, this *entity* would lose their privileges and roles within the OSSECO.

The power source for this power relationship dynamic is **role**. This power relationship dynamic was reported by S01, S02, I03, and I12. This power relationship dynamic represent another situation that we are not able to relate to any of the benefits and challenges. Therefore, it is a neutral power relationship dynamic. Figure 6.19 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Collaborator** OR **User** *Open a pull request* OR *Make a comment*, IF the contribution was valid to **Maintainer** AND the contribution was outstanding to **Maintainer**, THEN **Maintainer** *Accept pull request* OR *Respond to a comment* AND *Give higher privileges*. IF the contribution was valid to **Maintainer** AND the contribution was not outstanding to **Maintainer**, THEN **Maintainer** *Accept pull request* OR *Respond to a comment*. IF the contribution was not valid to **Maintainer** AND the contribution was not harmful in any way, THEN **Maintainer**

*Reject pull request OR Ignore the request AND Trigger D13. IF the contribution was not valid to **Maintainer** AND the contribution was harmful in any way, THEN **Maintainer** Reject pull request OR Ignore the request AND Remove higher privileges.*

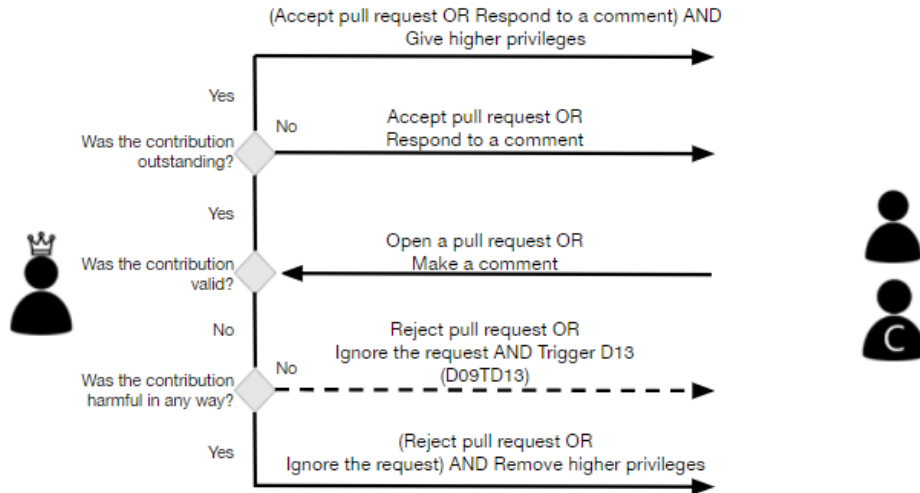


Figure 6.19: Role migration within the OSSECO refined.

Sponsoring (D10). In this power relationship dynamic, an *organization* decides to sponsor a *community-backed package* by sending it money and influencing its decisions. Since *organizations* joined OSSECO to achieve their business goals, it is possible that some *community-backed packages* are used as dependencies of this *organization's* projects. Therefore, it is essential for this *organization* to assure this dependencies maintenance and evolution according to its business goals.

Whenever this *community-backed package* accepts this sponsorship, it addresses *organization's* issues or responds to *organizations* comments. Thereafter, the *organization* evaluates the *community-backed package's* response to *organization's* request. If the *organization* considers the response as satisfactory, it maintains the sponsorship relationship, giving money and influencing *community-backed package's* decisions. Otherwise, if the *organization* considers the response as unsatisfactory, it interrupts the sponsorship relationship and stops to give money to the *community-backed package*. In this case, this power relationship dynamic also triggers, **dependency between projects power relationship dynamic (D04)**.

The power source for this power relationship dynamic is **financial reward**. This

power relationship dynamic was reported by R05, R06, I02, and I05. Since packages receive money to be maintained and occasionally suffer some kind of external influence from the sponsoring organization, we could relate this dynamic to the *addressing private roadmap* challenge. Figure 6.20 shows this power relationship dynamic graphical representation and its conditional representations is presented as follows:

WHEN **Organization** *Give money* OR *Influence a decision*, **Package** *Address an issue* OR *Respond to a comment*. IF the response was satisfactory to **Organization**, THEN **Organization** *Give money* OR *Influence a decision*. IF the response was not satisfactory to **Organization**, THEN **Organization** *Stop to give money* AND *Trigger D04*.

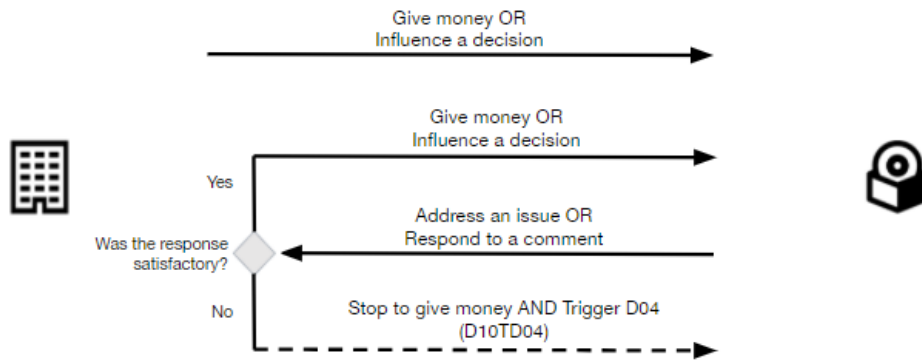


Figure 6.20: Sponsoring refined.

Technical leadership (D11). In this power relationship dynamic, a *collaborator* shows its technical skills and achieves a high reputation regarding those skills. Therefore, a *maintainer* consider the *collaborator* to be a technical reference. Moreover, in this dynamic, the *maintainer* has a development goal that he/she is not able to achieve or an issue that the *maintainer* does not have the technical skills to solve, he/she requests *collaborator's* help.

Whenever the *collaborator* considers this request to be interesting to it, it can be because of its finality or by being a subject that interests the *collaborator*, *collaborator* responds to this request. In this response, it is possible that the *collaborator* not only provides technical help to the *maintainer*, but also influences the *maintainer's*

decisions. This influence can occur either by sharing best practices and coding styles or by giving advice regarding a different approach to solve a given problem. Otherwise, if *collaborator* considers the request not to be interesting, it simply ignores the request and *maintainer* needs to request help from another *collaborator*.

The effects to the OSSECO depend on the *collaborator* response to *maintainer* request. In case *collaborator* provides technical help, not only *maintainer* become more skilled and able to achieve his/her goals but also *collaborator* achieves a higher reputation within the OSSECO. This reputation enhancement could lead to more entities requesting *collaborator's* help.

It is important to remark that this power relationship dynamic was not refined. The power source for this power relationship dynamic is **technical skills**. This power relationship dynamic was reported by S07, R06, R10, R14, I03, I05, I06, I07, I08, I09, I10, I11, and I12. Since this power relationship dynamic can help OSSECO community members to achieve their goals and the OSSECO to maintain its evolution, we can relate this dynamic to the *stable collaboration with the OSSECO* benefit. Figure 6.21 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Collaborator** *Show technical skills*, **Maintainer** *Ask for technical help*. IF the request is interesting to **Collaborator**, THEN **Collaborator** *Provide technical help* OR *Influence a decision*. IF the request is not interesting to **Collaborator**, THEN **Collaborator** *Ignore the request*.

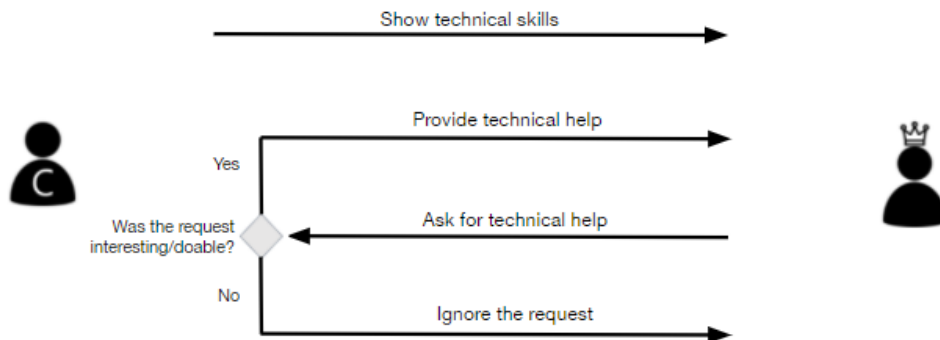


Figure 6.21: Technical leadership refined.

Blocking (D12). In this power relationship dynamic, an *entity*, that can be a *collaborator* or an *user*, tries to contribute with a package. Thereafter, a *maintainer* evaluates the *entity's* contribution to decide if the contribution is either valid or not.

This evaluation might differ from case to case depending on the package this contribution was sent to and the *maintainer* that is executing this evaluation. *Maintainer* considers during the evaluation if the contribution is adherent to the package road-map and if the code meets best practices and quality demands.

Whenever *maintainer* considers the contribution valid, *maintainer* accepts the pull request or responds the comment. In this case, this power relationship dynamic also triggers **role migration within the OSSECO power relationship dynamic (D09)**. However, if *maintainer* consider the contribution not valid, *maintainer* evaluates if the contribution is harmful to the package or not. An example of a harmful contribution is a contribution that contains security breaches in its code or that encourage other developers to abandon best practices and to ignore quality demands. If *maintainer* considers the contribution not to be harmful, he/she only ignores the request. However, if the *maintainers* considers the contribution to be harmful, he/she blocks the contribution's author. In this case, this power relationship dynamic also triggers **project forking power relationship dynamic (D13)**.

The power source for this power relationship dynamic is **role**. This power relationship dynamic was reported by R02. Since this power relationship dynamic is not necessarily blocking newcomers or supporting innovation, we could not relate it to any of the benefits and challenges previously identified. Therefore, this power relationship dynamic is neutral. Figure 6.22 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Collaborator** OR **User** *Open a pull request* OR *Make a comment*, IF the contribution was valid to **Maintainer**, THEN **Maintainer** *Accept pull request* OR *Respond to a comment* AND *Trigger D09*. IF the contribution was not valid to **Maintainer** AND the contribution was not harmful in any way, THEN **Maintainer** *Ignore the request* AND *Trigger D13*. IF the contribution was not valid to **Maintainer** AND the contribution was harmful in any way, THEN **Maintainer** *Block user*.

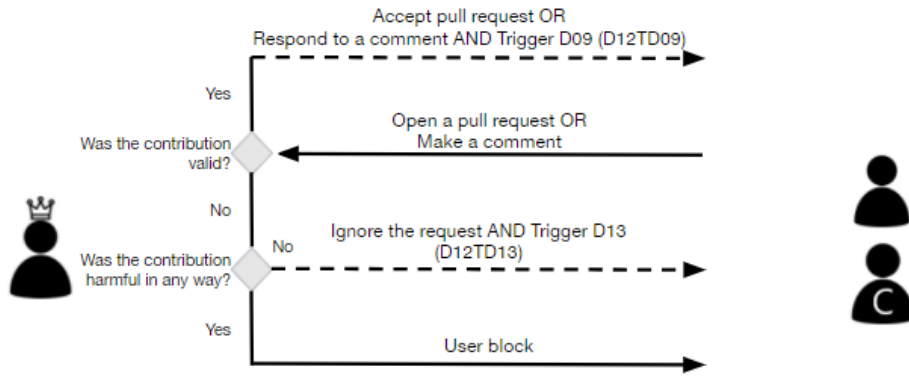


Figure 6.22: Blocking refined.

Project forking (D13). In this power relationship dynamic, an *entity A*, that can be either a *collaborator* or an *user*, make a contribution to an *entity B*, that can be a *community-backed package* or an *organization-backed package*, and then a *maintainer* evaluates this contribution as valid or not. Whenever the *maintainer* considers this contribution not valid, he/she rejects or ignores the contribution. Since *entity A* has needs that he/she depends on to continue developing his/her project accordingly, he/she could simply fork the project. Therefore, *entity A* would be able to proceed with the necessary changes in a new forked project and fulfill his/her needs.

The power source for this power relationship dynamic is the **role**. This power relationship dynamic was reported by I04 and I10. It was not possible to relate this power relationship dynamic to any of the previously identified benefits or challenges. Therefore, this is a neutral power relationship dynamic. Figure 6.23 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN Collaborator OR User *Open a pull request OR Make a comment* to **Community-backed package OR Organization-backed package**, THEN **Community-backed package OR Organization-backed package** *Request change* to **Maintainer**. IF the contribution was not valid to **Maintainer**, THEN **Maintainer** *Reject pull request OR Ignore the request*, THEN **Community-backed package OR Organization-backed package** *Reject pull request OR Ignore the request* from **Collaborator OR User** THEN **Collaborator OR User** *Fork the project*.



Figure 6.23: Project forking.

Dependencies of dependency (D14). In this power relationship dynamic, an *entity A*, that can be either a *community-based package* or an *organization-backed package*, has its code changed, affecting an *entity B*, that can be a *community-based package* or an *organization-backed package*. This code changed in *entity A* triggers the **dependency between projects power relationship dynamic (D03)**. The specific characteristic in this power relationship dynamic is that it represents a situation in which a package is indirectly affected by a code change in a package that is dependency of one of its dependencies.

The power source of this power relationship dynamic is the **breaking change**. This power relationship dynamic was reported by I01, I02, I03, I04, I05, I06, I07, I11, I12, and I13. Since this power relationship dynamic is a variation of the *dependency between projects* power relationship dynamic, it is also a neutral dynamic. Figure 6.24 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN Community-backed package OR Organization-backed package *Have its code changed*, THEN Community-backed package OR Organization-backed package *Have its code changed* AND Trigger *D03*.



Figure 6.24: Dependencies of dependency.

Disowning project (D15). In this power relationship dynamic, an *entity A*, that can be either a *maintainer* or a *technological platform*, decides to remove the ownership of a *maintainer* from a project. This situation can happen based on two different motivations: (i) self-preservation; and (ii) punishment.

Whenever a *maintainer* owns a project which is highly used by the OSSECO community, having a high number of dependants, it is possible that this project has also a high number of requests. Those requests can be related to bug that should be fixed or to code changes that might be done to achieve new goals and to provide new features. However, if this number of requests is too high (and this is a metric that would be completely personal and would depend on the *maintainer* perspective), it is possible that the *maintainer* is not able to respond to all of them in time. Moreover, the OSSECO community members, because of their needs, might put pressure on the *maintainer* for their changes to be done. Therefore, the *maintainer*, to avoid this pressure, would remove his/her own ownership in order to punish the OSSECO community members, since he/she would have no power to approve the changes anymore, and preserve his/her from the pressure.

In addition, occasionally *technological platforms* identify, or get notified, about a situation that might lead to a legal issue. Therefore, this *technological platform* would contact those who maintain a package that is involved with this possible legal issue. However, it is not guaranteed that those *maintainers* would attend to the *technological platform* requests. Since the *technological platform* has to protect itself and its community, it would remove the *maintainer's* ownership from a project and assume some of the tasks.

This power relationship dynamic power source is the **role**. This power relationship dynamic was reported by I03. When this action is taken by a technological platform, we can relate it to *organizing the OSSECO* and *stable collaboration with the OSSECO* benefits. However, when this action is taken by a maintainer, we are not able to relate it to any benefit or challenge. Figure 6.25 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN Maintainer OR Technological platform Remove ownership
from Maintainer.

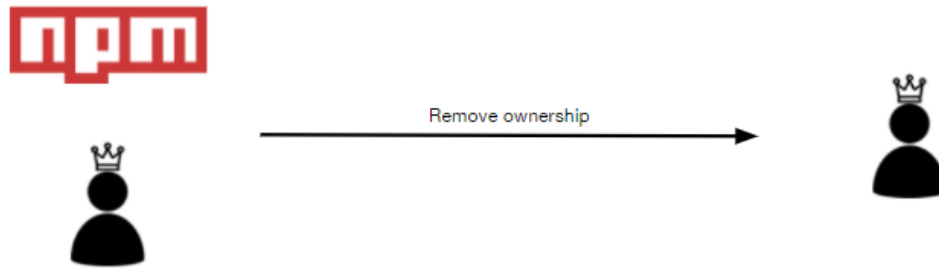


Figure 6.25: Disowning project.

Community building (D16). In this power relationship dynamic, an *entity A*, that can be either an *organization*, a *foundation*, a *conference*, a *collaborator* or a *maintainer*, invite another *entity B*, that can be an *user*, a *collaborator* or a *maintainer*, to contribute with the OSSECO. This situation aims to build an OSSECO community around the *technological platform*. Therefore, *entity A* which have a high reputation try to attract another *entity B* to contribute to the OSSECO.

The power source for this power relationship dynamic is **reputation**. This power relationship dynamic was reported by I10. Since this power relationship dynamic is directly related to inviting people to be part of the OSSECO, we can relate it to the *bringing innovation to the OSSECO* benefit. Figure 6.26 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Collaborator** OR **Conference** OR **Foundation** OR **Maintainer** OR **Organization** *Invite to contribute to* **Collaborator** OR **Maintainer** OR **User**.

Conference organizing (D17). In this power relationship dynamic, a *conference head* defines a conference program and, therefore, affects the *conference* and the information that the *conference* is planning on distributing. Whenever this power relationship dynamic take place, it is possible that it triggers **promoting collaboration power relationship dynamic (D07)**.

The power source for this power relationship dynamic is **reputation**. Since reputation is involved, dynamic is related to the *referent* power form. This power relationship dynamic was reported by I10, but was only identified combined with information reported from S05. Since this power relationship dynamic is related

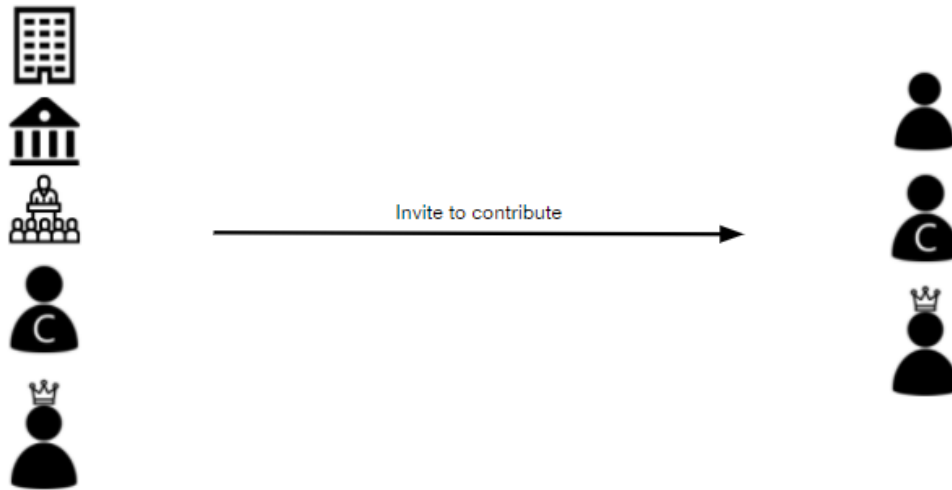


Figure 6.26: Community building.

to the organizing of a conference, we can relate it to the *organizing the OSSECO* benefit. Figure 6.27 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Conference head** *Define program* to **Conference**, THEN **Conference** *Trigger D07*.

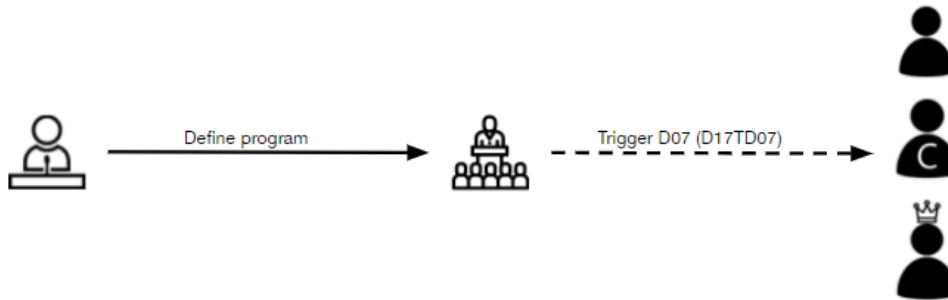


Figure 6.27: Conference organizing.

Foundation organizing (D18). In this power relationship dynamic, a *foundation member* makes a decision that affects the *foundation*. *Foundations* can help OSSECO to organize development, support the community building process and promote collaboration. Therefore, this power relationship dynamic could possibly trigger the **community building power relationship dynamic (D16)**.

The power source for this power relationship dynamic is **reputation**. Since reputation is involved, dynamic is related to the *referent* power form. This power

relationship dynamic was reported by S05. Since this power relationship dynamic is not only related to the organization of a foundation, but also to promoting collaboration, we can relate it to 2 benefits: *bringing innovation to the OSSECO* and *organizing the OSSECO*. Figure 6.28 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Foundation member** *Make a decision* to **Foundation**, THEN **Foundation** *Trigger D16*.

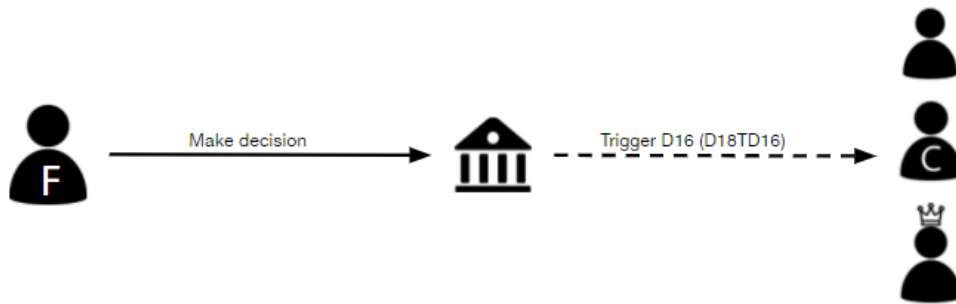


Figure 6.28: Foundation organizing.

Board organizing (D19). In this power relationship dynamic, a *board member* makes a decisions that affects a *board*. *Boards* can help OSSECO to organize development, support the community building process and promote collaboration. It is possible that this power relationship dynamic triggers the **community building power relationship dynamic (D16)**. The power source for this power relationship dynamic is **reputation**. Since reputation is involved, dynamic is related to the *referent* power form. This power relationship dynamic can be related to 2 benefits: *bringing innovation to the OSSECO* and *organizing the OSSECO*. This happens because this dynamic not only relates to the organization of board, but also with promoting collaboration within the OSSECO. This power relationship dynamic was reported by S05. Figure 6.29 shows this power relationship dynamic graphical representation and its conditional representation is presented as follows:

WHEN **Board member** *Make a decision* to **Board**, THEN **Board** *Trigger D16*.



Figure 6.29: Board organizing.

6.4 OSSECO Power Relationships Entity Model (OSSECO-PREM)

Considering the previously defined power relationship dynamics, we can now construct the OSSECO power relationships entity model (OSSECO-PREM) and present it. In order to construct this model, we chose the entities as the core concepts of it. Since interactions between the entities are essential for the OSSECO evolution and power relationships, OSSECO-PREM represents those interactions. Figure 6.30 shows the OSSECO-PREM dynamics view. This view only represents the power relationship dynamics between the entities.

The model represents the 19 power relationship dynamics between the 14 entities. It contains 60 interactions that can happen between those entities and that would be affected by the power relationships. Occasionally, entities can have more than one interaction happening between them. Whenever these interactions have the same source entity and the same destination entity, a single arrow containing the power relationship dynamics IDs connects both entities.

Since we present an entity-based model, it is possible to analyze how those power relationships affect the entities within an OSSECO. It is interesting to remark that the *organization* entity is the only entity that only exercises power over other entities. This behavior is possibly related to the fact that organizations join OSSECO communities to achieve their business goals, but have no interest in adapting to the OSSECO itself. In contrast, *technological platform* is the only entity that does not exercise power over any other entity.

The *maintainer* is the entity involved with more interactions in the represented model. It is involved with 27 interactions in total (both as source and destination entity). On the other hand, *online course*, *foundation member*, *board member* and

conference head are the entities involved with less interactions represented in this model. Each one of them is only involved with one interaction.

The entity that most exercises power over other entities is the *maintainer*, interacting 14 times as the source entity. However, *maintainer* is also the entity that other entities most exercise power over, with 15 interactions as destination entity. Therefore, it is possible to remark that *maintainer* is a central entity regarding power relationships within an OSSECO.

In addition, in this model we should also observe the triggers that occur between the power relationships dynamics. Those triggers affect entities that respond with an action triggering another power relationships dynamics. Figure 6.31 shows the OSSECO-PREM triggers view. In this model, we present 28 interactions that could trigger power relationships dynamics within an OSSECO.

Finally, Figure 6.32 shows the OSSECO-PREM complete view. This view represents both power relationships dynamics between entities and triggers.

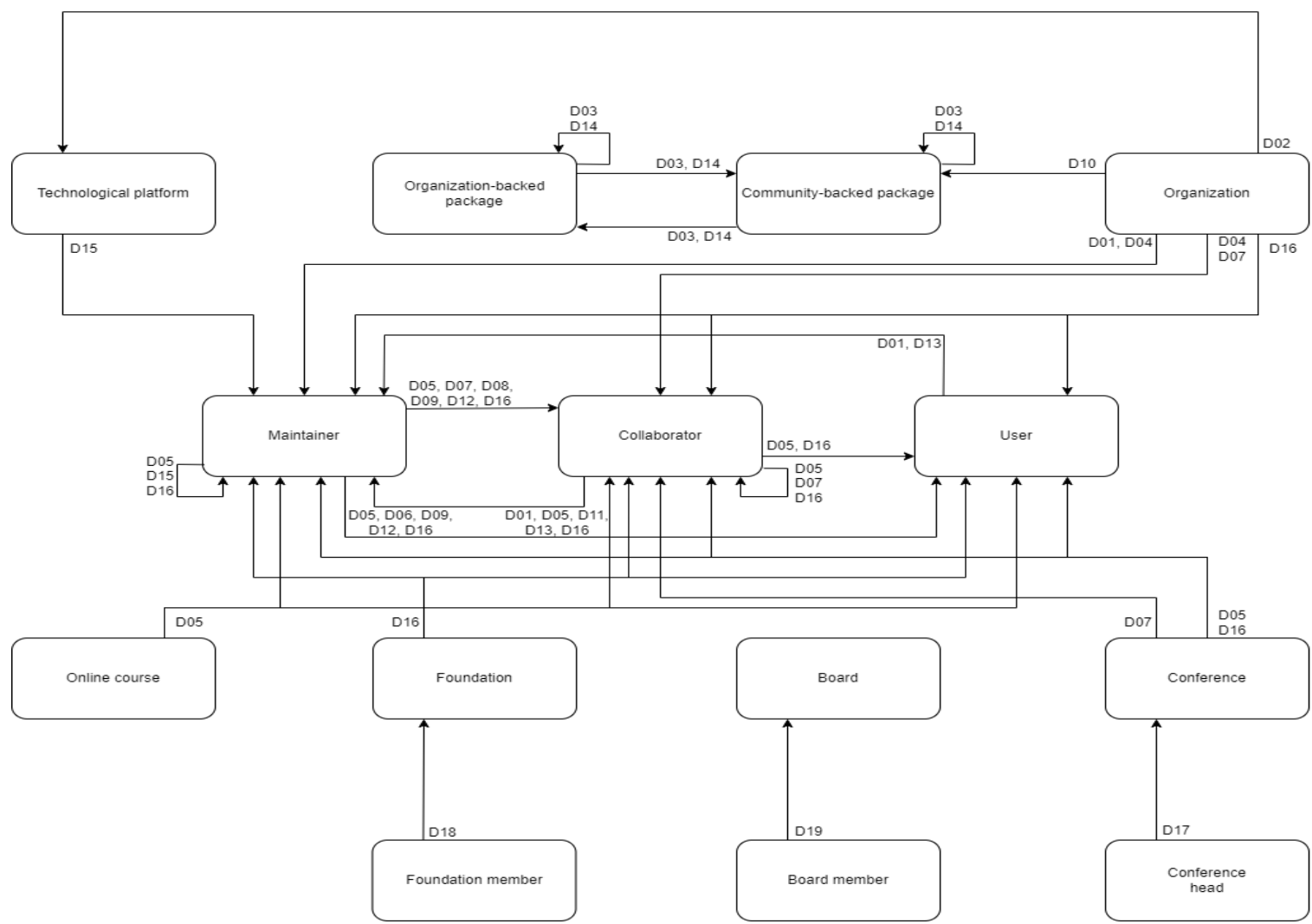


Figure 6.30: OSSECO Power Relationships Entity Model (PREM) dynamics view.

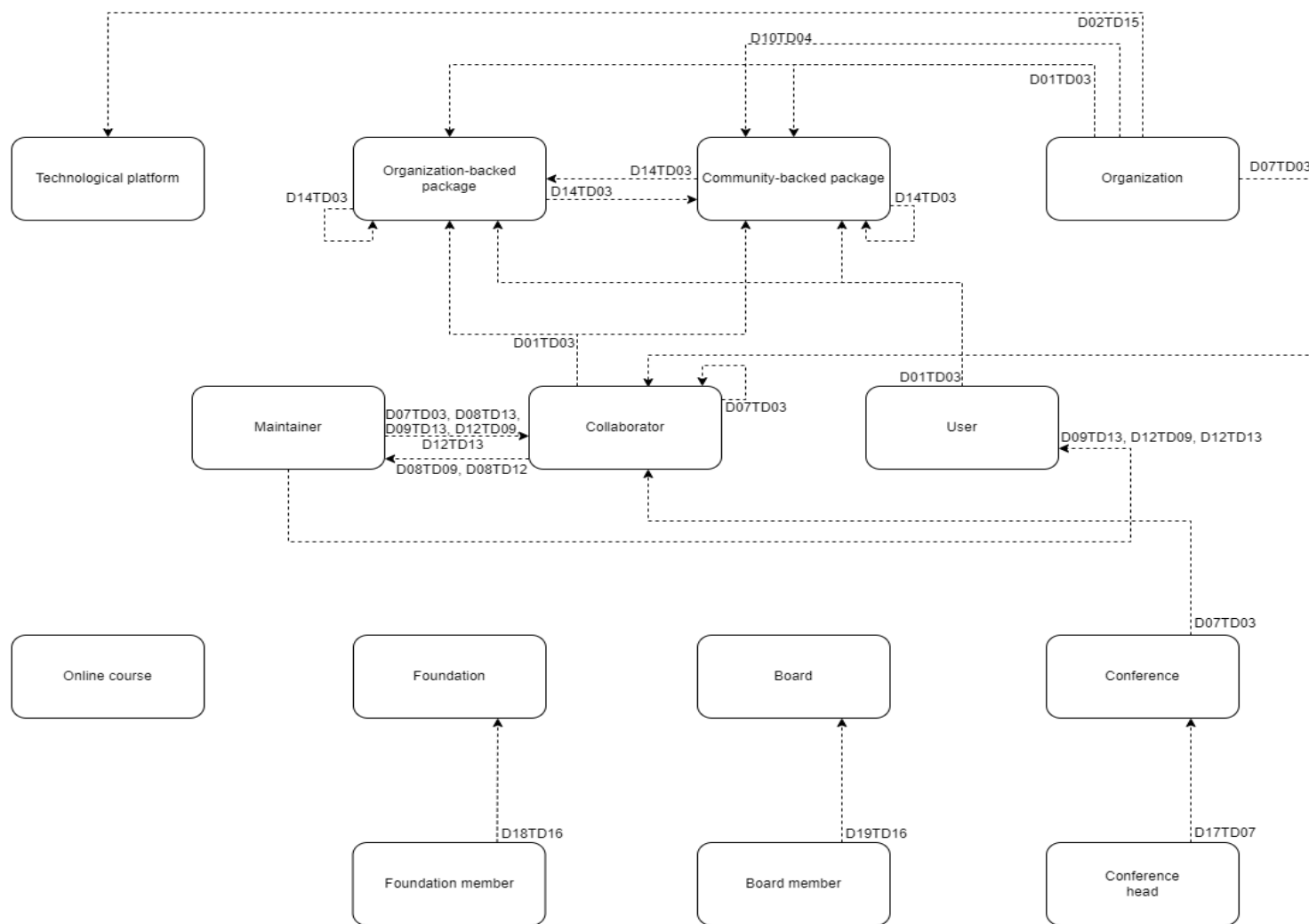


Figure 6.31: OSSECO Power Relationships Entity Model (OSSECO-PREM) triggers view.

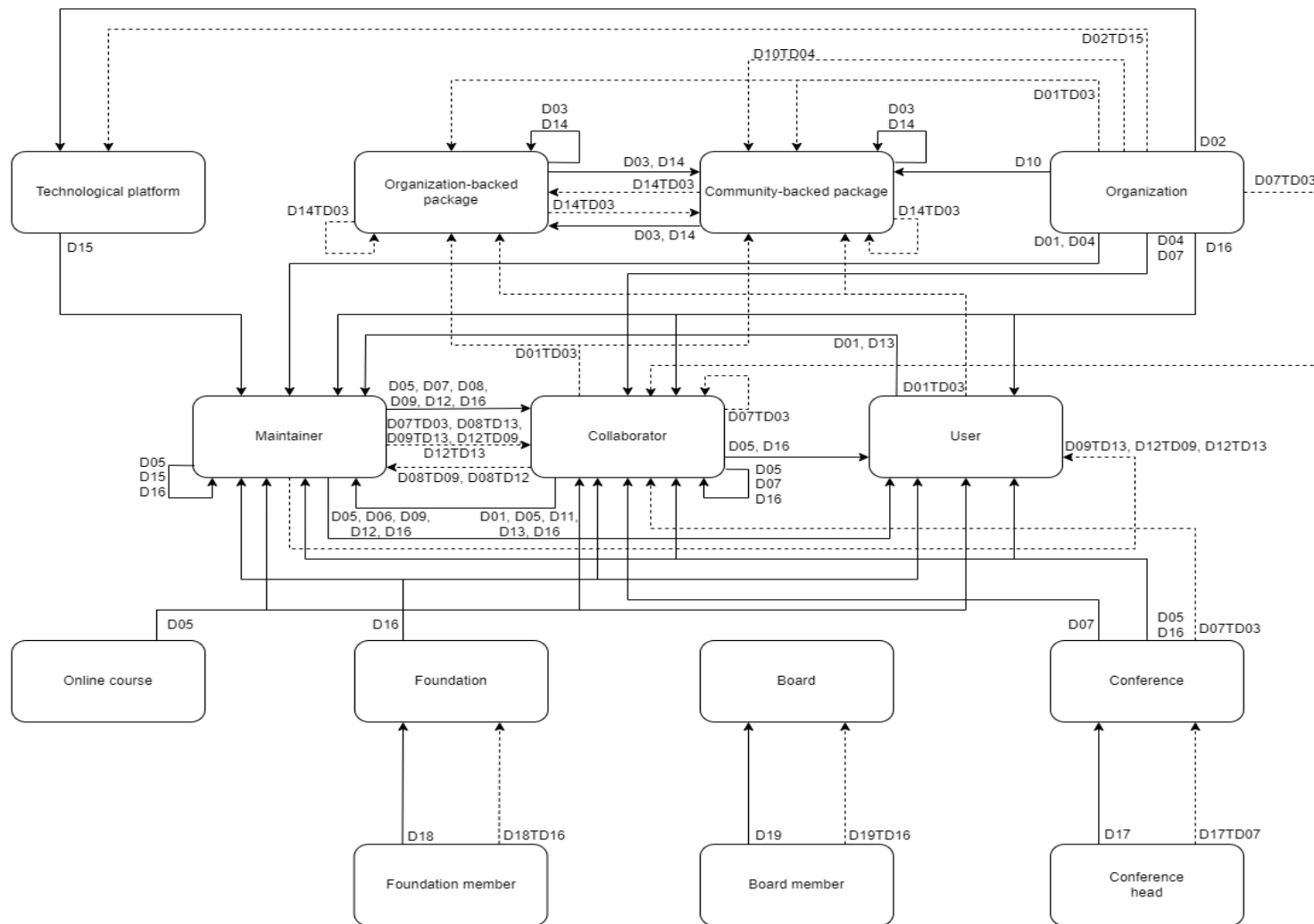


Figure 6.32: OSSECO Power Relationships Entity Model (OSSECO-PREM) complete view.

6.5 OSSECO Power Relationships Forms Model (OSSECO-PReFoM)

Considering the relationships between power relationship forms identified in Chapter 5, we were able to propose a conceptual model inspired by PRM-SECO (VALENÇA et al., 2017). This conceptual model We were able to identify 6 interactions between the 5 power relationship forms previously identified within an OSSECO. In addition, 8 power sources related to those power relationship forms were also identified during this information gathering.

Therefore, we are also able to make suggestions based on the respondents information. One of those suggestions regards the usage of ownership to justify changes or rejections within an OSSECO. In an OSSECO, community members tend to consider that their needs, either goals or problems to be solved, should be addressed. Therefore, whenever an entity decides to punish another, this entity might even achieve its goals within an OSSECO, however, its reputation would suffer a damage. In contrast, whenever an entity shows its technical skills, it maximizes its Expert and Referent power. Figure 6.33 shows the OSSECO-PReFoM.

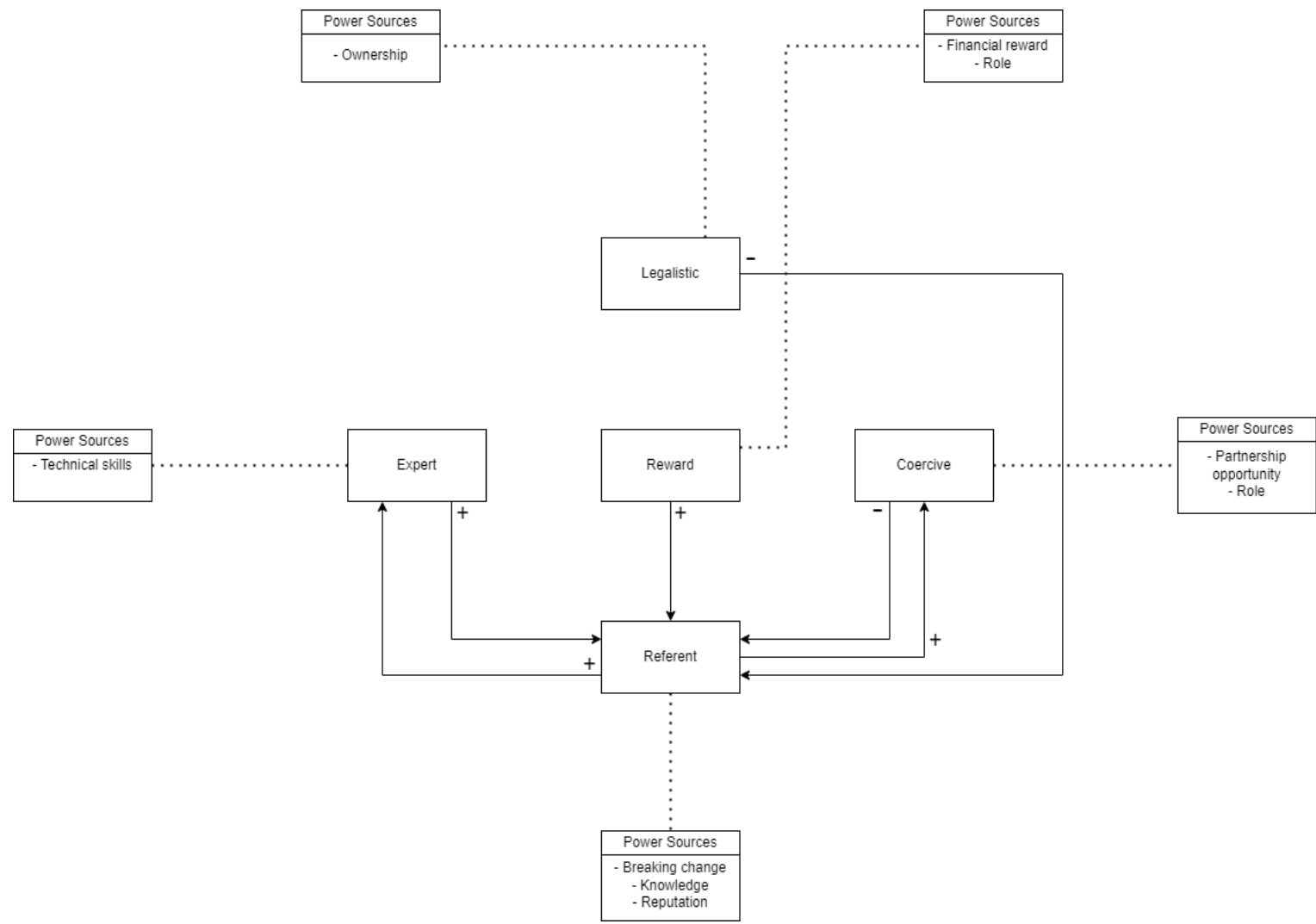


Figure 6.33: OSSECO Power Relationship Forms Model (OSSECO-PReFoM).

6.6 Limitations

In this study, we described a set of power relationship dynamics, representing interactions between entities, that would be related to power relationships within an OSSECO. Thereafter, we proposed a conceptual model to represent those power relationship dynamics between the different entities within an OSSECO.

The information used to describe those power relationship dynamics and, therefore, to construct this model was obtained from the studies reported in Chapter 4 and Chapter 5. Therefore, it comprises the snapshot of a moment considering what the literature has on power relationships within OSSECO and what members of the npm OSSECO experience regarding power relationships in their context.

Moreover, the power relationship dynamics and both conceptual models (OSSECO-PREM and OSSECO-PReFoM) have been evaluated by two experts: one of them with more than 10 years of experience regarding software ecosystems and the other one with more than 10 years of experience regarding open source software. However, the power relationship dynamics and the conceptual models have not been validated with any OSSECO community.

6.7 Final Remarks

In this chapter, we presented the process of planning and constructing the OSSECO-PREM, a power relationships entity-based model to represent the power relationships within an OSSECO. We also describe the OSSECO-PReFoM, a power relationship forms based model to represent the relationships between power relationship forms. Moreover, we described the power relationship dynamics that occur within an OSSECO.

As a result, we described 19 power relationship dynamics in which 14 entities are involved within an OSSECO. Those entities would also perform action when taking part in those power relationships. Therefore, we also identified 34 actions that those entities could perform within an OSSECO. Finally, we propose a model to represent those power relationships within an OSSECO.

In the next chapter, a discussion considering this research results and the literature is presented. We also discuss a possible change in the perspective on power

considering the scope.

Chapter 7. Discussion

7.1 Introduction

In this chapter, we present a discussion considering the results obtained by the present research and the literature related to power relationships and to OSSECO. We also present the perspective on power, in the proposed scope, and how it might have been changed considering those results.

The remainder of this chapter is structured as follows: Section 7.2 presents a discussion considering the results obtained and the literature; Section 7.3 presents a discussion regarding the perspective on power; Section 7.4 presents a discussion regarding power and automation; and Section 7.5 presents the final remarks.

7.2 Power Relationships Within an OSSECO and the Literature

In the OSSECO context, at a first moment, it might seem awkward to talk about power. Since these ecosystems rely on the classical view of open source development, in which the collaboration is mostly voluntary, the idea of having power relationships within an OSSECO would be strange (ALMARZOUQ et al., 2015). However, since it was possible to identify 19 power relationships dynamics and 6 power sources within an OSSECO during the investigation, it seems clear that power is present in OSSECO.

Since the motivations shifted in OSSECO in the last few years and other entities, such as organizations (GEROSA et al., 2021), are now being part of it, it could be the case in which these changes were responsible for the appearance of power and power relationships within an OSSECO. However, one of the identified power relationship dynamics is related to the role migration within the OSSECO. This role migration process has been identified by Jergensen et al. (2011). Therefore, the power that

arises from the role power source was already present in the OSSECO for some time, not being possible to claim that power is a new matter in this context. It is possible that since this power was treated as something natural by the OSSECO community for so long, it was not perceived as having power involved with it. That possibility is enhanced by the vision that, since the contribution is voluntary, power should not be involved with the OSSECO, only added value. Moreover, it explains the reason why power is seen as a negative aspect within an OSSECO. However, in the next section, we bring some results that indicate that power is not necessarily negative within an OSSECO.

7.3 The Perspective on Power Within an OSSECO

Considering what we discussed in the end of last section, it is necessary to also discuss the perspective on power within an OSSECO. In an environment such as an OSSECO that supports free and voluntary contribution, hierarchy and power tend to be rejected by the communities and seen as negative aspects. However, not only previous studies identified nearly hierarchical structures within an OSSECO but also, in the present research, we had some results that indicate that power can be positive for an OSSECO.

During the systematic mapping study reported in Chapter 4, we were able to identify some benefits and challenges that power relationships would have on OSSECO. Although it is true that we identified 2 challenges related to the power relationships, we were able to identify more benefits: 3. Moreover, 9 of the power relationship dynamics were related to benefits while 4 power relationship dynamics were related to challenges. Therefore, we were able to identify more positive power relationship dynamics than negative ones.

7.4 Power and Automation in OSSECO

The pull request evaluation process in the last few years received a new feature that helps maintainers to do their evaluation with more reliability: the automation. Platforms, such as GitHub, provide tasks automation in the pull request evaluation. Those tasks automation are mostly related to project building and testing, but might relate to other tasks that must be done before a pull request can be considered good

to be merged.

However, how does power relate to these automation? Maintainers have the power to approve or reject pull requests according to their evaluation. In their evaluation, maintainers usually evaluate the scope of the changes and the technical part of it. In the technical evaluation, the build and testing of the application as long as the technical aspects regarding code patterns and security. While some of those aspects need to be evaluated by the maintainers, building and testing can be automated using new features on GitHub. Therefore, the platform has the power to block a pull request and prevent it from being merged with the main code. Since the platform has this power, a question that comes in mind is: could we say that the platform has power over the pull request evaluation process? It would be possible to say that the platform holds some power over the process, but this power is not owned by the platform. It is giving to it by the maintainer, who can, at any moment, remove the automation and, hence, remove the power from the platform.

7.5 Final Remarks

In this chapter, we presented a discussion considering the results obtained by the present research and the literature related to power relationships and to OSSECO. We also presented how the perspective on power, in this scope, might have been changed by the results.

In the next chapter, we present the conclusion for this research.

Chapter 8. Conclusion

This chapter presents the conclusions of this dissertation and is structured in 5 sections: Section 8.1 presents the epilogue, in which we summarize this research; Section 8.2 presents the contributions, in which we present this research findings and their implication to practitioners and researchers; Section 8.3 presents the publications, in which we present the publications that were made during this research period; Section 8.4 presents the threats to validity, in which we present the research limitations; and Section 8.5 presents the future work, in which we present further studies to conduct in order to address other topics related to power relationships within an OSSECO.

8.1 Epilogue

Open source software ecosystems have always been an alternative to isolated and proprietary software development. Therefore, in its early days, contribution was mostly voluntary, in which developer would gather to achieve common goals or to solve common issues. Moreover, the power relationships between those developers were mostly related to the ones that had the power to approve changes and those who wanted changes to be approved.

Since motivations to contribute with the OSSECO shifted through the years, the relationships within the OSSECO have also been affected. Learning motivation and employment relationships brought different relationships into the OSSECO. Therefore, it is possible that those relationships are being affected for different power relationships other than the contribution approval/rejection. Moreover, those new power relationships could affect the interactions within the OSSECO and, therefore, its development.

In order to avoid those power relationships negatively affecting the OSSECO

and to benefit from them, it would be necessary to understand how those power relationships occur within an OSSECO and how they could affect an OSSECO. To guide this investigation about what are the power relationships and how they might affect the OSSECO, we defined a main research question (PRQ). We also defined five secondary research questions (SQ) to help detail which information about power relationships within an OSSECO we would investigate. In addition, a methodology was planned and conducted, involving methods such as systematic mapping study, exploratory study based on asynchronous interviews with an OSSECO community members, models construction and a field study based on semi-structured live interviews with an OSSECO community members.

During the research, the systematic mapping study embraced 10 studies from which to extract information about power relationships within an OSSECO. In the execution phase of the exploratory study, 14 members of the npm OSSECO responded to the asynchronous interview and provided an initial understanding of how power relationships are perceived by an OSSECO community. Finally, in the execution phase of field study, 13 members of the npm OSSECO were interviewed and provided information about the power relationships within their context.

Based on a previous power relationship forms definition proposed by Leonidou et al. (2014), we were able to propose an adapted version of this categorization to the OSSECO context. Thereafter, based on this new definition and in the information extracted from the studies selected in the systematic mapping study, we were able to identify 5 power relationship forms involved with an OSSECO. Therefore, we were able to answer SQ1.

In addition, we also identified 7 power relationship dynamics within this systematic mapping study results, which we used to answer SQ2. During exploratory study we were able to identify 5 more power relationship dynamics that were reported by the respondents. Finally, during the field study, we were able to identify 7 more power relationship dynamics which were reported by the interviewees. In the end, we were able to identify a total of 19 power relationship dynamics to answer SQ2. It is important to remark that some power relationship dynamics were mentioned in more than one study. Whenever this happened, these power relationship dynamics would be refined with the new information provided when necessary.

In order to answer SQ3, we used the results from both exploratory study and field study. During the exploratory study, we were able to identify 6 entities that would be involved with power relationships within an OSSECO. Considering the field study, we were able to identify 9 entities that would be involved with power relationships within an OSSECO. However, 2 of the entities identified in the field study were, actually, a refining on an entity that was previously identified in the exploratory study. The package entity was divided into community-backed package and organization-backed package, since their roadmaps and motivations within the OSSECO would be different between each other. Therefore, we were able to identify a total of 14 entities involved with power relationships within an OSSECO.

Regarding SQ4, we answered this question by using systematic mapping study results, which identified that an OSSECO could be positively affected by power relationships regarding OSSECO community members motivation to contribute. We were also able to answer this question considering the reports from interviewees in the field study, who claimed that power relationships could bring benefits regarding coding time reduction.

However, we were also able to identify negative effects and answer the SQ5. From the systematic mapping study results, we were able to extract negative effects related to the fact that barriers for newcomers and innovation that happen because of specific power relationships could lead to community members abandoning the OSSECO. Based on the information gathered during the field study, we were also able to identify a negative effect regarding maintainers that would start to dictate development goals within the OSSECO instead of considering the OSSECO community requirements. It is essential to remark that the barrier for a newcomer to have his/her voice heard was also mentioned during the field study as a negative effect.

Thereafter, the information gathered during this investigation was combined in the construction of two models that represent the power relationships within an OSSECO: OSSECO power relationships entity model (OSSECO-PREM) and OSSECO power relationship forms model (OSSECO-PReFoM). OSSECO-PREM was constructed based on the information from the systematic mapping study and the exploratory study. However, OSSECO-PREM was refined based on the new

information gathered in the field study. OSSECO-PreFoM was constructed based on the information gathered from systematic mapping study, exploratory study, and field study.

8.2 Contributions and Implications

This dissertation contributed to the construction of an understanding regarding power relationships within an OSSECO and how those power relationships would affect the interactions within the OSSECO and OSSECO development. The contributions provided by this research are as follows:

- *a power relationship forms definition for the OSSECO context*, that we constructed considering a previous definition to, considering the OSSECO context characteristics, define the power forms using terms related to this context;
- *a set of power relationship dynamics*, that contain the power relationship dynamics, its descriptions, entities involved with them, actions that can be performed within those dynamics, their power source and if they are positive, neutral or negative;
- *OSSECO-PREM*, which is a model constructed using the information gathered about the power relationship dynamics to represent those dynamics between the entities within an OSSECO;
- *OSSECO-PreFoM*, which is a model constructed using the information gathered about the power relationship forms to represent the relationships between those power relationship forms within an OSSECO.

Those who are involved with an OSSECO could use those contributions differently to achieve their goals. Therefore, the implications of this research could be divided into implications for practitioners and implications for researchers. Regarding the practitioners, who are involved with an OSSECO playing different roles and having different motivations to contribute, the implications are as follows:

- *to those who maintain packages within an OSSECO*: the power relationship dynamics identified indicate that it is expected that maintainers suffer the effects of those power relationships. These effects can be related to the pressure

of having changes merged into the package's main code. Since the number of simultaneous requests and the frequency of these requests can overwhelm maintainers, those should consider sharing package maintenance tasks with other maintainers. Considering that role migration, in which a developer aims to have more privileges within an OSSECO package, is an expected desire among OSSECO community members, it would not be difficult to recruit new maintainers. However, it is necessary to previously know this maintainer candidate since he would be given power that could affect not only this OSSECO package, but also those packages that are dependant on it;

- *to those who contribute with packages within an OSSECO or want to become collaborators (newcomers)*: whenever a collaborator wants to have its contribution accepted or an issue addressed, there are some points of attention to be considered. First, packages occasionally have best practices and contribution rules defined. Check those rules before proceeding with a pull request or an issue opening since this could avoid unnecessary or unwanted interaction with maintainers. Second, consider the package scope when suggesting a change. Helping to solve problems that maintainers might not be able because of tasks overload might give you power to request your desired changes. However, even when considering those points of attention, it is possible that your changes keep being rejected. Therefore, collaborator should remember that also holds power. It is possible to fork this package and develop the forked package as he/she wishes, or to simply replace the dependency for another package;
- *to those who use packages that are part of an OSSECO*: the power relationships between packages within an OSSECO can affect those who depend on those packages. Therefore, those who use OSSECO packages and depend on them should be aware that dependencies might have power of his/her project. This happens because package maintainers have the power to update their code and, indirectly, update dependants codes as well. Since security breaches are a main concern in software development, an user should be aware that this power relationships between projects could results in a malicious code being indirectly merged into his/her code;

- *to the organizations that are part of an OSSECO*: the power relationships affect not only the interactions within an OSSECO but also its development. Organizations who depend on OSSECO community-backed packages should be aware of all the risks previously indicated for those who are users within an OSSECO. Organization who maintain their own OSSECO packages should consider that OSSECO community has its own goals and problems to be solved. Therefore, the OSSECO community members require changes that address those goals and solve those problems. Since organizations usually have their own private roadmap, containing private business goals and problems to be solved, it would be essential to balance those different needs in order to have the OSSECO community help.

Moreover, we identified the 9 positive power relationship dynamics that should be supported to enhance collaboration within the OSSECO. We were also able to identify the 4 negative power relationship dynamics that should be avoided to prevent developers from abandoning the OSSECO. However, it is essential to remark that this information should be carefully used since this does not mean that the negative power relationship dynamics should not occur within an OSSECO or that only positive power relationship dynamics should occur within an OSSECO. Those power relationship dynamics should balance each other to achieve a healthy open collaborative environment within the OSSECO. Therefore, there might be situations in which the negative power relationship dynamics should be used to protect the OSSECO.

An example of a situation of that kind is when a newcomer proposes a change that might contain malicious code and should be rejected. His/her intention might not be inflict damage to the OSSECO with his/her contribution, but if his/her contribution were not rejected, that could happen. In this case, the OSSECO would exchange the innovation that the newcomer could bring into OSSECO for the OSSECO security.

Considering researchers, this research contributions can help in the understanding of social, technical and business interactions within an OSSECO. Researchers interested in investigating power relationships within an OSSECO also are provided with a power relationship forms definition, a set of described power relationship dynamics and the group of entities that take part in those power relationships within

an OSSECO.

8.3 Publications

- **Investigating Power Relation in Open Source Software Ecosystems:** this paper was produced during the definition phase of this research proposal. This paper was published in the 11th CBSoft’s Workshop on Theses and Dissertations (WTDSOft) (FARIAS et al., 2021a);
- **Power Relations Within an Open Source Software Ecosystem:** this paper was produced during the second year of this research when it was necessary to conduct a systematic mapping study to investigate the literature regarding power relations within an OSSECO. This paper was published in the 12th International Conference on Software Business (ICSOB21) (FARIAS et al., 2021b).

8.4 Threats to Validity

This section discusses this research limitations and threats to validity. Based on Merriam’s work (2015), we present our internal, external and reliability threats to validity. We also present the adopted strategies to minimize those threats.

8.4.1 Internal Validity

Regarding internal validity, we are concerned of how adherent to reality this research findings are. In a qualitative research, there is an understanding that the reality is not fixed and constant, but ever-changing. Therefore, in a qualitative research the reality is not measured but interpreted. Since the researcher provides an interpretation of reality, to guarantee the internal validity means that the interpretation provided is valid according to the qualitative paradigm’s notion of reality.

In order to guarantee the internal validity of this research, we use some of the strategies suggested by Merriam (2015). One of those strategies regards the *members check*. To proceed with this strategy, we conducted a field study based on interview with npm OSSECO community members. This field study considered the findings

in a previous exploratory study. Therefore, the field study provided a validation regarding the findings on the exploratory study.

Another strategy used to address the internal validity was obtaining information from different sources. During this research, we gathered information from the literature and from npm OSSECO community members that contribute to different packages within this OSSECO. Moreover, the investigation with the npm OSSECO community members not only comprised contributors from different packages, but also maintainers, collaborators and users from different countries. In addition, the respondents also had different levels of experience within the npm OSSECO and contributing with open source software as a whole.

8.4.2 External Validity

Regarding the external validity, which is related to the extent to which the research findings would be applicable to other situations. It is essential to remark that this research studies the power relationships within an OSSECO by focusing on the npm OSSECO context. In a exploratory study, the user of the research would discuss how the results would be applicable in different contexts, and not the researcher. Therefore, the interpretation of the research user would be determinant to whether the results of a exploratory study can be generalized or not.

Following Merriam's (2015) suggestion of strategies, an approach to address external validity is to provide *rich descriptions*. Therefore, the research user would have enough information to use this interpretation of reality to apply the findings to another context. Since we focus on the power relationships in this research, the power relationship dynamics provided in Chapter 6 present details regarding those power relationship dynamics, including the entities involved with them, the actions that can be performed and even graphical descriptions.

8.4.3 Reliability

Finally, the reliability regards the probability of finding a research results again. The more a study can be replicated and its results can be found again, the more stable a study is. Since a qualitative research aims to understand an interpretation of a reality, the replication might generate different results. However, the replication

should generate results that are different but feasible interpretations of a situation.

In order to guarantee the reliability of this research, we used different strategies across the different research phases. In the literature review reported in Chapter 4, we used *peer examination*, in which two experts, both with more than 10 years of experience conducting systematic mapping studies, one of them an expert in software ecosystems and the other an expert in open source software development, evaluated the obtained results. In addition, during the community study, we adopted the *multiple sources of data* strategy, by interviewing OSSECO community members from different projects, from different countries, who play different roles and that have different experiences contributing with the OSSECO. These strategies are suggested by Merriam (2015).

8.5 Future Work

Considering future work that could be conducted regarding power relationships within an OSSECO and related to the present research, there are three possible investigations:

- an updated systematic mapping study, considering new terms that we did not know when we planned and conducted the systematic mapping study reported in this dissertation. An example of a term that could be added to a search string is “dynamic” since it could refer to situations in which power relationships occur. Another term that could be added to the search string is “platform” since this term can be related to ecosystems;
- an exploratory study based on interviews with members of an OSSECO (or a set of OSSECO) other than npm could be conducted. Since we only interviewed members of the npm OSSECO, it would be interesting to gather information from different OSSECO. Moreover, it would be possible to later compare the results in different OSSECO;
- a field study based on interviews with members from other OSSECO to evaluate the power relationship dynamics and the models proposed in the present research. Therefore, it would be possible to verify if those artifacts could also

be used to understand power relationships in different OSSECO. Moreover, those artifacts could be refined to be adherent to other OSSECO.

References

- ABDALKAREEM, Rabe; NOURRY, Olivier; WEHAIBI, Sultan; MUJAHID, Suhaib; SHIHAB, Emad. Why Do Developers Use Trivial Packages? An Empirical Case Study on Npm. In: PROCEEDINGS of the 2017 11th Joint Meeting on Foundations of Software Engineering. New York, NY, USA: [s.n.], 2017.
- ALMARZOUQ, Mohammad; GROVER, Varun; THATCHER, Jason Bennett. Taxing the development structure of open source communities: An information processing view. **Decision Support Systems**, v. 80, p. 27–41, 2015.
- ALVES, Carina; VALENÇA, George; FRANCH, Xavier. Exercising Power in Software Ecosystems. **IEEE Software**, v. 36, n. 3, p. 50–54, 2019.
- ANGEREN, Joey van; KABBEDIJK, Jaap; JANSEN, Slinger; POPP, Karl. A Survey of Associate Models used within Large Software Ecosystems. **Computing**, v. 746, p. 27–39, Jan. 2011.
- BASILI, Victor. **Software Modeling and Measurement: The Goal/Question/Metric Paradigm**. USA, 1992.
- BOSCH, Jan. From software product lines to software ecosystem. In: p. 111–119.
- BOUCHARAS, Vasilis; JANSEN, Slinger; BRINKKEMPER, Sjaak. Formalizing Software Ecosystem Modeling. In: PROCEEDINGS of the 1st International Workshop on Open Component Ecosystems. Amsterdam, The Netherlands: Association for Computing Machinery, 2009. (IWOCE '09), p. 41–50.
- CAMBRIDGE. **POWER** — meaning in the Cambridge English Dictionary. Jan. 2022. Available from: <https://dictionary.cambridge.org/dictionary/english/power>.

- CONSTANTINO, Eleni; MENS, Tom. An Empirical Comparison of Developer Retention in the RubyGems and npm Software Ecosystems. **Innovations in Systems and Software Engineering**, v. 13, p. 101–115, Sept. 2017.
- DAHL, Robert. The concept of power. **Behavioral Science**, v. 2, n. 3, p. 201–215, 1957.
- DUCHENEAUT, Nicolas. Socialization in an Open Source Software Community: A Socio-Technical Analysis. **Computer Supported Cooperative Work**, v. 14, p. 323–368, 2005.
- ECKHARDT, Evert; KAATS, Erwin; JANSEN, Slinger; ALVES, Carina. The Merits of a Meritocracy in Open Source Software Ecosystems. In: PROCEEDINGS of the 2014 European Conference on Software Architecture Workshops. Vienna, Austria: Association for Computing Machinery, 2014. (ECSAW '14).
- EKLUND, Ulrik; BOSCH, Jan. Architecture for embedded open software ecosystems. **Journal of Systems and Software**, v. 92, p. 128–142, 2014.
- FARIAS, Victor; SANTOS, Rodrigo; WIESE, Igor; SEREBRENIK, Alexander; CONSTANTINO, Eleni. Investigating Power Relations in Open Source Software Ecosystems. In: PROCEEDINGS of the 12th Brazilian Conference on Software: Theory and Practice. Joinville: SBC, 2021a. p. 53–59.
- FARIAS, Victor; WIESE, Igor; SANTOS, Rodrigo. Power Relations Within an Open Source Software Ecosystem. In: WANG, Xiaofeng; MARTINI, Antonio; NGUYEN-DUC, Anh; STRAY, Viktoria (Eds.). **Software Business**. Cham: Springer International Publishing, 2021b. p. 187–193.
- FARIAS, Victor; WIESE, Igor; SANTOS, Rodrigo. What Characterizes an Influencer in Software Ecosystems? **IEEE Software**, v. 36, n. 1, p. 42–47, 2019.
- FOUCAULT, Michel. **Discipline And Punish: the Birth of the Prison**. New York: Pantheon Books, 1977.
- FRANCO-BEDOYA, Oscar; AMELLER, David; COSTAL, Dolors; FRANCH, Xavier. Open source software ecosystems: A Systematic mapping. **Information and Software Technology**, v. 91, p. 160–185, 2017.

- GEROSA, Marco; WIESE, Igor; TRINKENREICH, Bianca; LINK, Georg; ROBLES, Gregorio; TREUDE, Christoph; STEINMACHER, Igor; SARMA, Anita. The Shifting Sands of Motivation: Revisiting What Drives Contributors in Open Source. In: PROCEEDINGS of the 43rd International Conference on Software Engineering. [S.l.]: IEEE Press, 2021. p. 1046–1058.
- JANSEN, Slinger; CUSUMANO, Michael. Defining Software Ecosystems: A Survey of Software Platforms and Business Network Governance. In: ANALYZING and Managing Business Networks in the Software Industry. [S.l.: s.n.], 2013. chap. 1.
- JENSEN, Chris; SCACCHI, Walt. Role Migration and Advancement Processes in OSSD Projects: A Comparative Case Study. In: 29TH International Conference on Software Engineering (ICSE'07). [S.l.: s.n.], 2007. p. 364–374.
- JERGENSEN, Corey; SARMA, Anita; WAGSTROM, Patrick. The Onion Patch: Migration in Open Source Ecosystems. In: PROCEEDINGS of the 19th ACM SIGSOFT Symposium and the 13th European Conference on Foundations of Software Engineering. Szeged, Hungary: Association for Computing Machinery, 2011. (ESEC/FSE '11), p. 70–80.
- KITCHENHAM, B.; BUDGEN, D.; BRERETON, P. **Evidence-Based Software Engineering and Systematic Reviews**. [S.l.]: Chapman & Hall/CRC, 2015.
- LAAT, Paul de. Governance of open source software: State of the Art. **Journal of Management and Governance**, v. 11, p. 165–177, Feb. 2007.
- LEONIDOU, Leonidas; AYKOL, Bilge; LINDSAY, Valerie; KATSIKEAS, Constantine; TALIAS, Michael. Drivers and outcomes of exercised power in buyer-seller relationships: A meta-analysis, 2014.
- LIAO, Zhifang; WANG, Ningwei; LIU, Shengzong; ZHANG, Yan; LIU, Hui; ZHANG, Qi. Identification-Method Research for Open-Source Software Ecosystems. **Symmetry**, v. 11, p. 182, Feb. 2019.
- LINÅKER, Johan; REGNELL, Björn; DAMIAN, Daniela. A method for analyzing stakeholders' influence on an open source software ecosystem's requirements

- engineering process. **Requirements Engineering**, v. 25, p. 115–130, Mar. 2020a.
- LINÅKER, Johan; RUNESON, Per. Public Sector Platforms Going Open: Creating and Growing an Ecosystem with Open Collaborative Development. In: PROCEEDINGS of the 16th International Symposium on Open Collaboration. Virtual conference, Spain: Association for Computing Machinery, 2020b.
- MANIKAS, Konstantinos. Revisiting software ecosystems Research: A longitudinal literature study. **Journal of Systems and Software**, v. 117, p. 84–103, 2016.
- MANIKAS, Konstantinos; HANSEN, Klaus Marius. Characterizing the Danish Telemedicine Ecosystem: Making Sense of Actor Relationships. In: PROCEEDINGS of the Fifth International Conference on Management of Emergent Digital EcoSystems. Luxembourg, Luxembourg: Association for Computing Machinery, 2013a. (MEDES '13), p. 211–218.
- MANIKAS, Konstantinos; HANSEN, Klaus Marius. Software ecosystems – A systematic literature review. **Journal of Systems and Software**, v. 86, n. 5, p. 1294–1306, 2013b.
- MARSAN, Josianne; TEMPLIER, Mathieu; MAROIS, Patrick; ADAMS, Bram; CARILLO, Kevin; MOPENZA, Georgia Leida. Toward Solving Social and Technical Problems in Open Source Software Ecosystems: Using Cause-and-Effect Analysis to Disentangle the Causes of Complex Problems. **IEEE Software**, v. 36, n. 1, p. 34–41, 2019.
- MERRIAM, Sharan; TISDELL, Elizabeth. **Qualitative Research: A Guide to Design and Implementation**. [S.l.]: Wiley, 2015. (The Jossey-Bass higher and adult education series).
- NPM. **npm About**. Jan. 2022. Available from: <https://www.npmjs.com/about>.
- PLAKIDAS, Konstantinos; STEVANETIC, Srdjan; SCHALL, Daniel; IONESCU, Tudor; ZDUN, Uwe. How Do Software Ecosystems Evolve? A Quantitative Assessment of the r Ecosystem. In: PROCEEDINGS of the 20th International Systems and Software Product Line Conference. Beijing, China: Association for Computing Machinery, 2016. (SPLC '16), p. 89–98.

- ROBSON, Colin. Real World Research : A Resource for Social Scientists and Practitioner-Researchers / C. Robson., Jan. 2002.
- SANTOS, Rodrigo Pereira dos; WERNER, Cláudia Maria Lima. ReuseECOS: An Approach to Support Global Software Development through Software Ecosystems. In: 2012 IEEE Seventh International Conference on Global Software Engineering Workshops. [S.l.: s.n.], 2012. p. 60–65.
- STRAUSS, Anselm; CORBIN, Juliet. **Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory**. Thousand Oaks, CA: SAGE Publications, 2007.
- TEIXEIRA, Jose; ROBLES, Gregorio; GONZALEZ-BARAHONA, Jesus. Lessons Learned from Applying Social Network Analysis on an Industrial Free/Libre/Open Source Software Ecosystem. **Journal of Internet Services and Applications**, v. 6, July 2015.
- TELLERIA, Juan. Power relations? What power relations? The de-politicising conceptualisation of development of the UNDP. **Third World Quarterly**, Routledge, v. 38, n. 9, p. 2143–2158, 2017.
- TSAY, Jason; DABBISH, Laura; HERBSLEB, James. Influence of Social and Technical Factors for Evaluating Contribution in GitHub. In: PROCEEDINGS of the 36th International Conference on Software Engineering. Hyderabad, India: Association for Computing Machinery, 2014. (ICSE 2014), p. 356–366.
- VALENÇA, George; ALVES, Carina. A theory of power in emerging software ecosystems formed by small-to-medium enterprises. **Journal of Systems and Software**, v. 134, p. 76–104, 2017.

Appendices

Appendix I. Exploratory Study

I.1 Invitation

Email subject: Assistance for determining what is your feeling about power relations in the npm ecosystem

Email body:

Dear [NAME_PARTICIPANT],

My name is Victor Farias, and I am a researcher from Federal University of the State of Rio de Janeiro, Brazil. Along with Alex Vaz (Federal University of the State of Rio de Janeiro, Brazil), Roberto Dias (Federal University of the State of Rio de Janeiro, Brazil) and Rodrigo Santos (Federal University of the State of Rio de Janeiro, Brazil), we are studying power relations within projects in the npm open source software ecosystem at GitHub repositories.

This interview is aimed at determining what is the perspective of the GitHub community on the power relations within the npm ecosystem. While mining information from some GitHub repositories activities, we found your activity in the npm ecosystem. Therefore, it would be of great help if you could spend some minutes to answer the following questionnaire: [QUESTIONNAIRE_LINK]

The estimated time to complete the questionnaire is 5-7 minutes.

Your participation in this interview is greatly appreciated, but in no way required. Information will be used to support our ongoing research. Your individual responses or any personal information about you will not be identifiable in any published reports. Your IP address is not

collected. Thank you so much for your time, and please do not hesitate in contacting us if you have any questions.

Thanks in advance,

Victor Farias, Alex Vaz, Roberto Dias and Rodrigo Santos

I.2 Terms and Conditions

Understanding community opinion about power relations in open source software ecosystems

This asynchronous interview is conducted by Alex Vaz (UNIRIO), Roberto Dias (UNIRIO), Victor Farias (UNIRIO) and Rodrigo Santos (UNIRIO).

We are researches from Brazil studying power relations in Open Source Software Ecosystems. This asynchronous interview is aimed at determining what is the perspective of the community on npm ecosystem about power relations within this ecosystem.

The estimated time to complete the questionnaire is 3-5 minutes.

STATEMENT OF INFORMED CONSENT

By answering this asynchronous interview, you agree to permit the researcher to obtain, use and disclose the anonymous information provided as described below

CONDITIONS AND STIPULATIONS

1. I understand that all information is confidential. I will not be personally identified. I agree to complete the survey for research purposes and that the data derived from this anonymous survey may be published in journals, conferences and blog posts.
2. I understand that my participation in this research survey is totally voluntary, and that declining to participate will involve no penalty or loss of benefits. If I choose, I may withdraw my participation at any time. I also understand that if I choose to participate, I may decline to answer any question that I am not comfortable answering.
3. I understand that I can contact the researcher if I have any questions about the research survey. I am aware that my consent will not directly benefit me. I am also aware that the author will maintain the data collected in perpetuity and may utilize data for future academic work.
4. By clicking the button below I freely provide consent and acknowledge my rights as a voluntary research participant as outlined above and provide consent to the researcher to use my information in conducting research on the areas mentioned above.
5. By clicking "Next" you agree with the terms above. You can withdraw your participation at any time.

In case of any questions or suggestions, please contact us at alex.vaz@edu.unirio.br, roberto.dias@uniriotec.br and victor.farias@uniriotec.br

Faça login no Google para salvar o que você já preencheu. [Saiba mais](#)

Próxima

Página 1 de 3

Limpar formulário

Figure I.1: Asynchronous interview participation Terms and Conditions.

I.3 Demographic Questions

Understanding community opinion about power relations in open source software ecosystems

[Faça login no Google](#) para salvar o que você já preencheu. [Saiba mais](#)

*Obrigatório

Demographic Questions

What is your gender? *

☐ Male

☐ Female

☐ Transgender

☐ Gender Variant / Non-Conforming

☐ Prefer not to answer

☐ Outro: _____

In npm ecosystem, Are you a *

☐ Project owner

☐ Project committer (has merge permission and contributes more regularly)

☐ Project contributor (has no merge permission and contributes less regularly)

☐ Outro: _____

How many projects do you contribute to? *

☐ 1 project

☐ 2 projects

☐ 3 projects

☐ 4 projects

☐ 5+ projects

[Voltar](#) [Próxima](#) Progress bar Página 2 de 3 [Limpar formulário](#)

Figure I.2: Asynchronous interview demographic questions (DQs).

I.4 Technical Questions

Understanding community opinion about power relations in open source software ecosystems

[Faça login no Google](#) para salvar o que você já preencheu. [Saiba mais](#)

*Obrigatório

Questionary

"Power relations are relations between an individual who has something to offer and another individual who desires that something. In open source software ecosystems, these relations are directly related to roles and motivation for contribution. Understanding and managing those relations is fundamental for an open source software ecosystem do evolve, because they can affect both collaboration (by improving and encouraging it) and conflicts (by avoiding them)." Do you agree with what is stated? *

☐ Yes, I agree

☐ No, I don't agree

☐ I agree in part

Justify your previous answer. *

Sua resposta

What power relations do you identify inside the npm ecosystem? *

Sua resposta

Figure I.3: Asynchronous interview technical questions (TQs) part 1.

Which of the motives below relate to your participation within the npm ecosystem? *

- ☐ Development for personal puporse
- ☐ Financial incentive
- ☐ More participation in ecosystem's decisions
- ☐ Development for professional purpose
- ☐ Contributing with bug fixes
- ☐ Development for academical purpose
- ☐ Managing ecosystem's colaboration
- ☐ None of the above
- ☐ Outro: _____

Would you agree to be interviewed later about this matter? If so, please leave your e-mail for contact.

Sua resposta _____

[Voltar](#) [Enviar](#)[Página 3 de 3](#) [Limpar formulário](#)

Figure I.4: Asynchronous interview technical questions (TQs) part 2.

Appendix II. Field Study

II.1 Invitation

Subject: Invitation to interview about power relations within the npm ecosystem!

Text:

Hi [NAME_PARTICIPANT],

I hope this message finds you well.

My name is Victor Farias (UNIRIO, Brazil) and I'm a researcher studying power relations within the npm ecosystem. Together with Igor Wiese (UTFPR, Brazil) and Rodrigo Santos (UNIRIO, Brazil), we aim at understanding how power relations take place in the npm ecosystem and how they could influence the ecosystem evolution.

To do so, we need to interview some members of the npm ecosystem. While searching for potential respondents, we found your profile because of your participation in the [PARTICIPATION_WITHIN_NPM_ECOSYSTEM].

We firmly believe that understanding the power relations within an ecosystem like npm could help to promote collaboration within the ecosystem and to prevent developers from abandoning it.

It would be great if you could answer some questions for us. It would take 20-25 minutes and we could schedule it according to your disponibility. The interview would be entirely in English and we would have to record it. If you are not comfortable with the language or with this format of live interview, let us know and we will arrange an alternative way. Your participation means a lot to us!

If you have any questions regarding our study or the interview before

making a decision, we would be more than happy to provide additional information.

Looking forward to hearing from you,

II.2 Terms and Conditions

This interview is conducted by Victor Farias (UNIRIO) for his Masters dissertation. It is supervised by Igor Wiese (UTFPR) and Rodrigo Santos (UNIRIO). The questions' design also had the collaboration of Alexander Serebrenik (Eindhoven University of Technology) and Eleni Constantinou (Eindhoven University of Technology), who will not have access to your responses.

Victor Farias is studying power relations in Open Source Software Ecosystems. Power relations have been widely investigated in other areas of interest such as psychology and sociology, and even in other types of software ecosystems, such as the small and medium enterprises ecosystem. Those power relations could influence the relationships within the ecosystem and their decisions and evolution. In the context of the open source software ecosystem, where the community holds the power in a decentralized format, it would be harder but even more important to understand the power relations within the ecosystem.

The estimated time to complete the questionnaire is 20-25 minutes.

STATEMENT OF INFORMED CONSENT

By agreeing on being interviewed, you agree to permit the researcher to obtain, use and disclose the anonymous information provided as described below

CONDITIONS AND STIPULATIONS

1. I understand that all information is confidential. I will not be personally identified. I agree to complete the interview for research purposes and that the data derived from this anonymous interview may be published in journals, conferences and blog posts.
2. I understand that my participation in this research interview is totally voluntary, and that declining to participate will involve no penalty or loss of benefits. If I choose, I may withdraw my participation at any time during the interview. I also understand that if I choose to participate, I

may decline to answer any question that I am not comfortable answering.

3. I understand that I can contact the researcher if I have any questions about the research interview. I am aware that my consent will not directly benefit me. I am also aware that the author will maintain the data collected during 3 years.

4. I freely provide consent and acknowledge my rights as a voluntary research participant as outlined above and provide consent to the researcher to use my information in conducting research on the areas mentioned above.

5. I state that I am of legal age according to my local law.

6. I can withdraw my participation at any time during the interview.

In case of any questions or suggestions, please contact me at victor.farias@edu.unirio.br

II.3 Demographic Questions

The image shows a web-based questionnaire titled "Investigating Power Relations in Open Source Software Ecosystems". Below the title is a Google login link: "Faça login no Google para salvar o que você já preencheu. Saiba mais". The questionnaire is divided into a section titled "Demographic Questions" which contains five questions, each with radio button options:

- Which gender do you most identify with?
 - ☐ Male
 - ☐ Female
 - ☐ None of the above
 - ☐ Prefer not to say
- Which role do you most identify with within the npm ecosystem?
 - ☐ Maintainer
 - ☐ Contributor
 - ☐ User
 - ☐ All of the above
- How long have you been contributing with code within the npm ecosystem?
 - ☐ 1-3 years
 - ☐ 3-5 years
 - ☐ 5-7 years
 - ☐ more than 7 years
 - ☐ never contributed with code
- How long have you been commenting or opening issues within the npm ecosystem?
 - ☐ 1-3 years
 - ☐ 3-5 years
 - ☐ 5-7 years
 - ☐ more than 7 years
 - ☐ never contributed with comments or opening issues
- How long have you been contributing (code, comments, issues) with open source software in general?
 - ☐ 1-3 years
 - ☐ 3-5 years
 - ☐ 5-7 years
 - ☐ more than 7 years
 - ☐ never contributed with open source software

At the bottom, there are navigation buttons: "Voltar" and "Próxima", a progress bar, and the text "Página 2 de 3" and "Limpar formulário".

Figure II.1: Interview's demographic questionnaire.

II.4 Eindhoven University of Technology's Ethical Review Board Acceptance Letter

Ms. Eleni Constantinou
e.constantinou@tue.nl



Date
December 2, 2021

Reference
ERB2021MCS16

Ethical Review Board TU/e

T +31 (0)40 247 6259
ethics@tue.nl

intranet.tue.nl/ethics

Ethical review research proposal

Dear Ms. Constantinou,

It is a pleasure to inform you that the local Ethical Review Board (ERB) from the department of Mathematics and Computer Science has discussed and approved your application "Investigating Power Relations within Open Source Software Ecosystems", under the condition you change the terms and conditions from "I am a researcher studying ..." to "Victor Farias is studying ...".

The Board wants to draw your attention to the terms and conditions in the appendix.

Success with your research!

Sincerely,

A handwritten signature in black ink, appearing to be 'J.F. Groote', written over a white background.

Prof.dr.ir. J.F. Groote

Enclosures
1

The ERB retains the right to revise its decision regarding the implementation and the WMO¹/WMH² status of any research study in response to changing regulations, research activities, or other unforeseen circumstances that are relevant to reviewing any such study. The ERB shall notify the principal researcher of its revised decision and of the reasons for having revised its decision.

¹WMO: Law on Medical Scientific Research involving Human Beings (in Dutch: Wet medisch-wetenschappelijk onderzoek met mensen)

²WMH: Medical Device Directive (in Dutch: Wet op de medische hulpmiddelen)

Figure II.2: Ethical Review Board acceptance letter, page 1.

Date
December 2, 2021

TU/e

APPENDIX 1

Terms and conditions

Amendments

When considerable amendments are made to the design of the study or educational activity, or when the time period between ERB approval and start of the study is longer than one year, please consult the ERB.

Privacy and research data management

The ERB would like to point out that collecting, handling and storing personal information is subject to the General Data Protection Regulation. Please visit TU/e intranet for the latest information and regulations on www.tue.nl/rdm

Page
2 from 2

Figure II.3: Ethical Review Board acceptance letter, page 2.